A MULTILEVEL HYBRID CLASSIFIER USING VARIANT FEATURE SETS FOR INTRUSION DETECTION

by

Aslıhan (ÖZKAYA) AKYOL

Submitted to the Institute of Graduate Studies in Science and Engineering
in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Mevlana (Rumi) University
2016

A MULTILEVEL HYBRID CLASSIFIER USING VARIANT FEATURE SETS FOR INTRUSION DETECTION

Submitted by **Aslıhan (ÖZKAYA) AKYOL** in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical and Computer Engineering Department, Mevlana (Rumi) University

APPROVED BY:

Examining Committee Members:

Prof. Dr. Bekir KARLIK . . . . . .
(Thesis Supervisor)

Prof. Dr. Ahmet ARSLAN . . . . . .

Prof. Dr. Novruz ALLAHVERDİ . . . . . .

Assist. Prof. Dr. Armağan ÖZKAYA . . . . . .

Assist. Prof. Dr. Mesut GÜNDÜZ . . . . . .

Assist. Prof. Dr. Mehmet ARGIN . . . . . .
Head of Department, Electrical and Computer Engineering

Prof. Dr. Ali SEBETCİ . . . . . .
Director, Institute of Graduate Studies in Science and Engineering

DATE OF APPROVAL (Day/Month/Year)

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Aslıhan (ÖZKAYA) AKYOL

Signature :

ABSTRACT

A MULTILEVEL HYBRID CLASSIFIER USING VARIANT FEATURE SETS FOR INTRUSION DETECTION

Aslıhan (ÖZKAYA) AKYOL

Ph.D. Thesis, 2016

Thesis Supervisor: Prof. Dr. Bekir KARLIK

Keywords*:* Artificial Intelligence, Intrusion Detection, Discernibility Function, Feature Selection, Artificial Neural Networks, Anomaly Based Intrusion Detection, Anomaly Detection, Network Based Intrusion Detection Systems, Transfer Learning, Network Security

With the broad usage of the Internet, people have been experiencing more security issues than ever before. Therefore internet users are employing software such as firewalls, antivirus, anti-spam, and anti-malware to protect their personal information. But these software programs are not enough to guard systems from various network attacks such as Denial of Service (DoS). Therefore Intrusion Detection Systems (IDS) have been developed to detect the security violations. Commercial IDSs are usually misuse-based which can only detect known attacks. On the other hand, anomaly-based IDSs, which have the capability of detecting unknown attacks, have been under research. However, IDSs are still immature since they create high false alarms, and have to be trained with up-do-date network communication data packets.

This thesis examines various anomaly based intrusion detection techniques and offers a solution for frequently updated datasets to facilitate and accelerate the learning process. It focuses on achieving better detection rates and lower false alarm rates. Meanwhile, many additional issues come into the picture when it comes to real-time applications. In real-time applications, it is vital to train the intrusion detection systems fast to not miss any communication packets. In order to speed up the detection, feature selection is conducted

which helps to utilize only related features. Moreover, it is widely known that unrelated features negatively impact the learning system. Thus, removing unrelated features increases the learning performance. As a result, Feature Selection (FS) improves the classification rate and lightens the IDS. Therefore, FS methods are widely used in artificial intelligence and machine learning, particularly when dealing with large datasets. FS has also become widespread in IDS. This thesis focuses also on various effects of pre-processing intrusion detection dataset, and applies a wrapper based FS using the Discernibility Function as the search algorithm to create candidate feature subsets. A hybrid method is proposed as well and conducted on a benchmark intrusion detection dataset as well as on a real intrusion detection dataset. Moreover, this thesis studies transfer learning which helps to utilize the previously known information, and applies the previously gained knowledge to the learning system with an updated dataset.

ÖZET

SALDIRI TESPİTİ İÇİN FARKLI ÖZELLİK SETLERİ KULLANAN ÇOK DÜZEYLİ MELEZ SINIFLANDIRICI

Aslıhan (ÖZKAYA) AKYOL

Doktora Tezi, 2016

Tez Danışmanı: Prof. Dr. Bekir KARLIK

Anahtar Kelimeler: Yapay Zeka, Saldırı Tespiti, Ayırtedilebilirlik Foksiyonu, Özellik Seçimi, Yapay Sinir Ağları, Anomaly Tabanlı Saldırı Tespit Sistemi, Anomaly Tespiti, Ağ Tabanlı Saldırı Tespit Sistemi, Transfer Öğrenme, Ağ Güvenliği

İnternetin geniş kullanımı ile, insanlar her zamankinden daha fazla güvenlik sorunları yaşıyor. Bu nedenle internet kullanıcıları kişisel bilgilerini korumak için güvenlik duvarları, antivirüs, anti-spam ve anti-malware gibi yazılım kullanıyor. Fakat bu yazılımlar, "Denial of Service" (DoS) gibi birçok ağ saldırılarına karşı sistemleri korumak için yeterli değildir. Bu nedenle, güvenlik ihlallerini tespit edebilmek için Saldırı Tespit Sistemleri (STS) geliştirilmiştir. Ticari STS'ler genellikle imza tabanlıdır ve sadece bilinen saldırıları algılayabilmektedir. Diğer yandan, bilinmeyen saldırıları tespit yeteneğine olan sahip anomali tabanlı STS'ler üzerine araştırma yapılmaya devam edilmektedir. Ancak STS'ler hala olgunlaşmamışır, çünkü yüksek oranda yanlış alarm (false alarm) vermekte ve sürekli güncel ağ iletişim paketleri ile eğitilmek zorundadırlar.

Bu tez, çeşitli anomali tabanlı saldırı tespit sistemlerini inceler ve sık güncellenen veriler için öğrenme işlemini hızlandırmak ve kolaylaştırmak için çözüm sunuyor. Aynı zamanda daha yüksek tespit oranları ile daha düşük yanlış alarm oranlarına ulaşmaya odaklanır. Diğer yandan, gerçek zamanlı uygulamalar söz konusu olduğunda pek çok ek sorunlar ortaya çıkmaktadır. Gerçek zamanlı uygulamalarda, herhangi bir network iletişim paketinin gözden kaçırılmaması için saldırı tespit sistemlerinin eğitiminin hızlı olması hayati önem taşımaktadır. Saldırıların tespitini hızlandırmak amacıyla sadece ilgili özelliklerin kullanılmasını sağlayan özellik seçimi yapılır. Ayrıca, yaygın olarak bilinmektedir ki, ilgisiz özellikler öğrenme sistemini olumsuz yönde etkilemektedir. Bu nedenle, ilgisiz özellikler kaldırılarak öğrenme performansı artırılır. Sonuç olarak özellik seçimi sınıflandırma oranını yükseltir ve STS'yi hafifleştirir. Bu nedenle özellik seçme, özellikle büyük veri setleri ile çalışırken yapay zeka sistemleri ile makine öğrenmede yaygın olarak kullanılmaktadır. Özellik seçme STS'lerde de yaygın olarak uygulanmaktadır. Bu tez saldırı tespit veri setlerinde üzerlerinde uygulanan ön işlemenin çeşitli etkileri üzerinde de durmaktadır. Ayrıca özellik aday alt kümelerini oluşturmak için arama algoritması olarak ayırt edilebilirlik (discernibility) fonksyonunu kullanarak sarma tabanlı özellik seçimini uygular. Buna ek olarak melez bir yöntem sunulmuş ve önerilen bu yöntem hem benchmark saldırı tespit veri kümesi üzerinde hem de gerçek bir saldırı tespit veri kümesi üzerinde uygulanmıştır. Bunlara ek olarak bu tezde önceden öğrenilmiş bilgiyi değerlendiren ve güncellenmiş veri seti ile öğrenme sistemine önceden kazanılmış bilgiyi uygulayan transfer öğrenme çalışılmıştır.

This thesis work is dedicated to all members of my family,

Especially to my parents,

And to my beloved husband

For their unconditional and continuous support and encouragement.

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor Prof. Dr. Bekir Karlık for his continuous support of my Ph.D study, for his patience, motivation, and immense knowledge. Throughout my research, I hugely benefited from his constructive guidance.

Besides my advisor, I would like to thank the rest of my thesis committee: Assit. Prof. Dr. Armağan Özkaya and Assit. Prof. Dr. Mesut Gündüz for their insightful comments and encouragement, but also for the hard question which incented me to widen my research from various perspectives. I also would like to express my sincere appreciation to Assist. Prof. Dr. Mustafa Kaiiali who generously helped me to improve my research and academic writing skills.

Finally, special thanks to my family for all their love and encouragement, to my grandmother, uncle, aunts, brother, sisters and their families for supporting me spiritually, especially to my parents who supported me in all my pursuits. I would like to thank for all of the sacrifices that they've made on my behalf. Their prayer for me was what sustained me thus far. And most of all, I would like to thank my loving, supportive, encouraging, and patient husband, Assit. Prof. Dr. Harun Akyol whose faithful support during the final stages of this Ph.D. is sincerely appreciated. Thank you!

Aslıhan (Özkaya) Akyol
Mevlana University
March 2016

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Explanation
A-IDS	Anomaly based Intrusion Detection System
ANN	Artificial Neural Network
BN	Bayesian Networks
BP	Backpropagation
C4.5	C4.5 Decision Tree
C5.0	Improved C4.5
CGI	Common Gateway Interface
CNF	Conjunctive Normal Form
CPE	Cost Per Example
DARPA	Defense Advanced Research Projects Agency
DF	Discernibility Function
DFBFS	Discernibility Function Based Feature Selection
DNF	Disjunctive Normal Form
DoS	Denial of Service
DR	Detection Rate
FAR	False Positive Rate
GA	Genetic Algorithm
HIDS	Host-based IDS
HYP	Hypersphere Algorithm
ICMP	Internet Control Message Protocol
ID3	Iterative Dichotomiser 3
IDEVAL	Intrusion Detection EVALuation
IDPS	Intrusion Detection and Prevention System
IDS	Intrusion Detection Systems
IP	Internet Protocol
IPv6	Internet Protocol version 6
K-M	K-Means Clustering
KDD	Knowledge Discovery and Data Mining
MHCVF	Multilevel Hybrid Classifier with Variant Feature Sets
MLP	Multi-Layer Perception
NIDS	Network-based IDS

NSL-KDD	A new version of the KDD dataset
R2L	Remote to Local
RBF	Radial Basis Function
RMS	Root Mean Square
RP	Resilient Back Propagation
SVM	Support Vector Machine
TCP	Transmission Control Protocol
U2R	User to Root
UDP	User Datagram Protocol

1. INTRODUCTION

In recent years, IT security has received increasing attention. The main reason is the enormous growth of the usage of Internet and network systems. Almost all types of organizations have a network system or at least a connection to the Internet. This wide usage comes with its security risks. According to a research survey [1] 90% of organizations have at least one successful security breach observed, and 41% of the organizations had to spend $500,000 or more over the past year because of the attacks. Not only organizations but also individual users have been experiencing cyber-attacks. Another survey [2] reported that 43% of daily internet users have experienced spam emails and 8% of internet users have experienced or been a victim of identity theft. Even a person who has never used the Internet is in risk; because governments which have all the personal information of their citizens use the Internet. All these risks put researcher's attentions on Intrusion Detection Systems (IDS) which aim to detect intrusions (security violations) on networks or computer systems [3].

Intrusion detection methods fall into two main categories according their method of detection [4]. These categories are signature-based detection (also known as knowledge-based detection or misuse detection) and anomaly-based detection (also known as behavior-based detection). In the signature-based detection method, known attacks are analyzed to extract the discriminating characteristics and patterns (named as signatures). These signatures are used to compare with the captured network traffic and to detect intrusions. Signature based detection has a very low False Alarm Rate (FAR), thus this method is being widely used by the commercial IDS vendors. However, signature-based detection methods are not able to detect any new or unknown attacks. For this reason, researchers focus on anomaly-based detection which is a powerful method in detecting unknown and new (zero-day) attacks. It analyses the patterns of normal network and system activities, and classifies them as anomalous if they differ from normal patterns.

Many anomaly based intrusion detection systems (A-IDSs) have been studied. Most of these studies have focused on obtaining higher detection rates (DRs) and lower FARs. A considerable portion of these studies has used the Knowledge Discovery and Data Mining (KDD) 99 dataset which is created in 1999. This dataset has two parts: one for training and the other one for testing. The test dataset includes same attacks that exist in the train dataset. It also includes new attacks that do not exist in the train dataset.

1.1. **Related Works**

There are several studies which use machine learning algorithms on intrusion detection. Most studies train their IDSs with the train dataset, and then test their trained systems with the test dataset and then compare the obtained Detection Rates (DRs) and FARs (or other metrics) with previous similar studies. For instance, in [5] the authors propose a Hierarchical Gaussian Mixture Model for IDSs. They apply the model on the KDD'99 train dataset and test the proposed model on the KDD'99 test dataset. The DRs and FARs of the proposed method are presented and compared with previous studies.

The KDD'99 dataset is derived from the DARPA IDEVAL dataset, which is in a TCP dump format, by transforming the TCP packets into TCP connections. Some criticisms have been done about the transformation of DARPA 98 to KDD'99 in [6]. The authors declared that the attributes in the KDD'99 dataset, which is obtained through a transformation of the DARPA 98 dataset, are not enough to detect some attacks especially Remote to Local (R2L) attacks. They propose to add additional attributes in to the KDD'99 dataset to improve the detection rate of novel attacks. They apply the C4.5 algorithm to train and test their system with the train and test datasets which are the modified versions of the KDD'99 Train and Test datasets.

Another IDS study has been done in [7]. The authors randomly selected data from the KDD'99 dataset for their Train and Test sets. They used Rough Set Theory (RST) to select features from the KDD'99 dataset and applied the support vector machines on the data with selected features. They also applied SVM (Support Vector Machine) on the whole data (without feature selection) and on part of the dataset with entropy based feature selection. The DRs and FARs of the three different methods have been compared. It is presented that the method with RST provides higher accuracy than without feature selection and with entropy based feature selection.

Hu et al. [8] proposed an IDS based on the AdaBoost algorithm which is applied on the KDD'99 Train and Test datasets. They have presented the DRs and FARs and compared them with previous studies where they have shown that their method has low computational complexity and error rates.

Lee et al. [9] have used the DARPA 98 TCPDump training and testing files as the train and test datasets, respectively. They have used the ID3 (Iterative Dichotomizer 3) algorithm as the data mining method. The steps of generating the Decision Tree have been described in detail.

The DRs of each attack category have been presented and compared with a previous study. Their method showed an improvement in detecting new kinds of attacks.

Mahoney [10] has proposed an anomaly detection model named as NETAD (Network Traffic Anomaly Detector). This model first filters the traffic and let to pass only the packets of most interest and then models the commonly observed protocols to flag packets that have not been observed for a long time. The author used the DARPA dataset of week 1-3, which do not have any attacks, as the train dataset; and the dataset of week 4 and 5, which have 185 attacks, as the test dataset. The performance of proposed model has been compared with similar studies participated in the original 1999 DARPA IDS evaluation competition.

Powers et al. [11] have proposed a hybrid IDS model that includes artificial immune system and Kohonen Self Organizing Maps (SOM) together. The artificial immune system has been trained with normal connections and is assigned to filter the connections that differ from normal connections. In the second stage, the connections, filtered as anomalous, are passed through the SOM which is trained to classify the attacks. The authors have used the KDD'99 10% dataset on their research and compared the DRs of each attack category with previous studies.

Sabri et al. [12] have used the IDA (Intelligent Data Analysis) software to reduce the FARs of IDSs. They have randomly selected 7000 samples from the KDD'99 dataset and presented the FARs and DRs of each attack category. They showed that their technique has reduced the FAR and increased the accuracy for the randomly selected samples.

A multiple-level hybrid classifier which combines the supervised tree classifiers and unsupervised Bayesian clustering for IDSs has been proposed by Xiang et al. in [13]. They have used the KDD'99 dataset in their experiments. After feature selection and removing unrelated features from the dataset they applied their hybrid method and compared the DRs and FARs with the results of the KDD'99 Cup competition winner. The proposed method showed high DR and low FAR.

Feng et al. [14] have proposed a new classification method named as Combining Support Vectors with Ant Colony (CSVAC), that combines the modified versions of both Support Vector Machines (SVM) and Clustering based on Self-Organized Ant Colony Network (CSOACN). They have applied the proposed method on the KDD'99 dataset and presented the DRs, FARs and training times. They have also compared their results with the results of

IDSs which used the SVM and CSOACN independently. They also compared the results with the results of the KDD'99 competition winner. The proposed method outperformed SVM and CSOACN, which are applied independently, in terms of classification rate and run-time efficiency.

It has been evident that there are numerous related works. However it should be noted that there are still open issues in IDSs, which are presented in the following section.

1.2. **Open Issues in IDSs**

The most important challenge in IDSs is to increase the system effectiveness which can be done with a system that detects close to 100% of attacks with minimal FAR [15]. Currently this goal has still not been achieved [15].

Another open issue is the need to apply the IDS algorithms on real world dataset, stated in [16]. It claims that currently most of the IDS algorithms are conducted on benchmark datasets, whereas real world environments are more complicated.

Another research states that, high FARs for unknown attacks, long training and testing times, low detection accuracies, and high computational costs are still the challenges of IDSs [17].

It is also stated that most of the IDSs are tested with KDD'99 dataset which is already outdated [18]. They emphasize that the number of experiments with new and real datasets is still low. Moreover, minimizing FARs is still an ongoing challenge. It is also pointed out that the volumes of datasets have grown rapidly in recent years. This increase causes some problems in hybrid approaches. They also claim that data preparation and feature selection are currently challenging tasks.

Pradhan et.al [19] have stated that; due to technical reasons, current anomaly intrusion detection approaches usually suffer from high FARs.

There are some studies that analyze and/or criticize the datasets used in IDSs. A study, related with generating synthetic data in order to be used in IDS evaluation, is presented in [20]. This study has focused on HTTP traffic which is generated from the logs of a web server. They have compared the generated data with the KDD'99 dataset and with a real dataset. They have concluded that the generated data is more similar to the real dataset than the KDD'99 dataset.

Chandola et al. have been discussing the challenges of anomaly detection in [21]. They mention that normal behaviors keep changing; therefore, a current normal behavior may not represent normal behaviors sufficiently in the future. They have also mentioned about the difficulties regarding the availability of labeled datasets for training and testing IDSs.

Wu and Banzhaf have been discussing the drawbacks of the KDD'99 dataset, which is failing to realistically simulating real network [16]. They have also believed that a new (up to date) dataset should be produced. Moreover, it is highlighted that not only intrusive behavior is changing over time but also legitimate user behavior is shifting over time; therefore, the IDSs should able to adapt itself to constantly changing environments. They also mention about the volume of the IDS datasets as they are usually too large for building an effective IDS.

It is discussed in [22] that IDSs are challenging high FARs. It is also pointed out that the IDSs do not guard themselves against attacks. They also present that IDS benchmarking is insufficient. The need for adaptive IDS is also mentioned in this study as well as in [16].

As a result, current anomaly-based detection systems are immature since they create high false alarms and have to be trained constantly with up-to-date datasets. Furthermore, generating and labeling network packets to create an up-to-date dataset is a costly process in terms of time and resource consumption. The IDSs should be trained with an up-to-date dataset in order to have the IDSs work in real environments. But after some time this dataset may become outdated, because legitimate network behaviors may change or new attack behavior may appear [8], [21], [23]. For instance, the remote desktop connection protocol (RDP) can be used to demonstrate how the legitimate network behavior can change. Let's assume that the train dataset was created before 2001 and was not trained for RDP packets since the RDP was started to be used with Windows XP in 2001. Therefore, the IDS that is trained with this train dataset will not be aware of this new technology. Consequently, it is possible that the IDS is going to label the RDP packets as intrusions. This will cause a raise in FAR. Many other examples can be given since new technologies show up continuously. For that reason the IDS will always turn out to be outdated after some time [23]. On the other hand, the availability of labeled data used to train IDS is usually a major issue [21]. Collecting and labeling new data is a costly process; and throwing old data away is a waste [24], [25]. Therefore to train the IDS continuously all over again and keep it updated is difficult.

As a result, there are lots of open issues in IDSs. It can be stated that, research in IDSs is still mature. There are lots of issues that need to be improved. This thesis focuses on some of the open issues and suggests some solutions which are presented in Section 6. The contributions of this thesis are summarized in the following section.

1.3. Contributions

The effect of the pre-processing stage that influences the performance of IDSs is investigated in this thesis. Moreover, a fast and less expensive way of feature selection is studied and conducted. Additionally a hybrid method is proposed which archives high detection rates and low FAR. This thesis also explore transferring previous knowledge by using transfer learning so that the need and effort to recollect the train data could be reduced; the time to train the system could be decreased; and higher detection rates could be obtained.

This thesis provides several contributions to the research in A-IDSs:

1. It demonstrates the effect of pre-processing that influence the performance of IDSs.
2. It applies a fast feature selection algorithm on the large KDD intrusion detection dataset.
3. According to a comprehensive research, this is the first research that applies discernibility function based feature selection on IDSs.
4. A hybrid intrusion detection model with variant feature sets is studied and proposed which provides high detection rates and low false alarm rate.
5. According to a comprehensive research, this is the first research that applies Transfer Learning on IDSs.
6. It emphasizes on the practicability of IDS such as easily updating the IDS with only newly appeared data and forwards the attentions to keeping the IDS up-to-date.

1.4. Organization of the Thesis

This thesis is organized as in follows: Section 2 presents the state of art in IDSs, such as popular datasets that has been used in IDSs, types of intrusion detection methods, feature selection methods that have been widely used on large datasets and current challenges in IDSs. The most popular machine learning algorithms applied on A-IDSs such as the Backpropagation algorithm, Decision Tree classifier and Radial Basis Function have been

presented in Section 3. A feature selection method used in this thesis has been explained in Section 4 while the descriptions of the attacks found in the KDD Cup and ISCX datasets are stated in Section 5. Three different implementations and their results have been presented in Section 6. It concludes and presents the future work in Section 7.

2. STATE OF THE ART IN INTRUSION DETECTION

This chapter presents the state of art in IDSs. The most popular datasets, used in IDS evaluation, the types of IDSs, and the types of feature selection methods have been described in this chapter by presenting previous related studies. The challenges in network based IDSs which are being faced and are still challenging researchers who are studying network based IDS is also described in this chapter.

2.1. Popular Datasets

In intrusion detection, generally three types of datasets are collected for research: 1- network packets, 2- command sequences from user input, and 3- low-level system information such as log files and CPU/memory usage [16]. Some of the most common used datasets can be seen in Table 1.

Table 1 Commonly used datasets

Dataset Name	**Abbreviation**
Network Packets	
1998 DARPA Intrusion Detection Evaluation Dataset - TCPDump Files [26]	DARPA98 IDEVAL
1999 DARPA Intrusion Detection Evaluation Dataset - TCPDump Files [26]	DARPA99 IDEVAL
KDD Cup 1999 Dataset [27]	KDD'99 or KDD Cup'99
Information Exploration Shootout [28]	IES
The Kyoto Dataset [29]	Kyoto
Information Security Centre of Excellence 2012 IDS Dataset [30]	ISCX 2012 IDS
Center for Applied Internet Data Analysis dataset [31]	CAIDA
Lawrence Berkeley National Laboratory (LBNL) dataset [32]	LBNL
DefCon dataset [33]	DEFCON
University of Brescia dataset [34]	UNIBS
Tezpur University IDS dataset [35]	TUIDS
Command Sequences, User Behaviors and System Information	
1998 DARPA Intrusion Detection Evaluation Data Set - BSM Files [26]	BSM98
1999 DARPA Intrusion Detection Evaluation Data Set - BSM Files [26]	BSM99
UNIX User Data [36]	UNIXDS

Some researchers also prefer to generate their own datasets by using real network traffic, even though it is difficult to give assurance that the network data is intrusion free. Therefore artificial traffic generators are used by some researchers [16].

2.1.1. The KDD Cup 1999 Data

The most popular dataset is the KDD'99 which is derived from the DARPA98 IDEVAL dataset by assembling TCP (Transmission Control Protocol) packets into TCP connections. This dataset is used in the International Knowledge Discovery and Data Mining Tools competition in 1999. KDD'99 has 41 features (see Table 2) plus one class type which indicates the attack names (Figure 1). The full train dataset contains about five million connection records while the 10% of it contains 494,021 records and the test dataset contains 311,029 connection records. The 10% Train dataset contains 24 attack types, and the test dataset contains 38 types of attacks. The link in [27] provides the KDD Cup (the full dataset), KDD Cup 10% (a 10% subset) and KDD Cup test data.

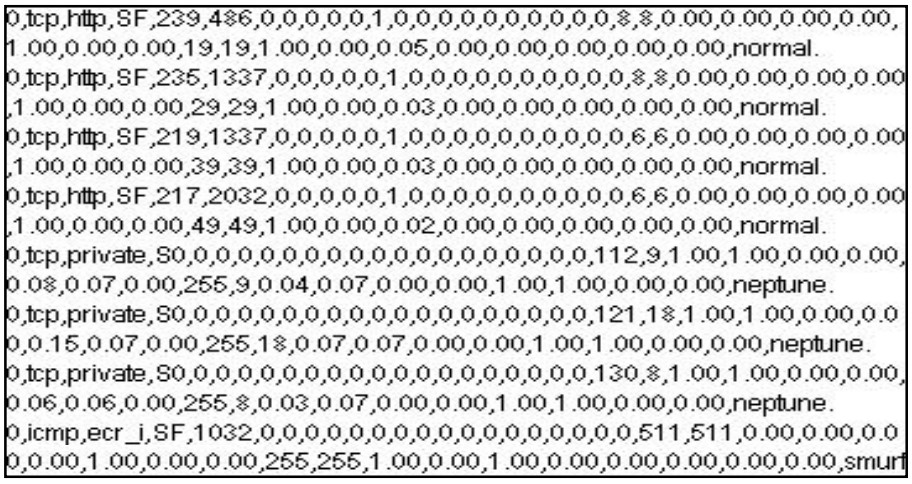

Figure 1 Sample data of KDD Cup

DARPA has grouped the attacks into 4 main groups: Denial of Service (DoS), Remote to Local (R2L), User to Root (U2R), and Probing (or Probe, Surveillance) [37]. In *DoS attacks,* attackers aim to interrupt the network service by loading high volume traffic, creating malformed packets or taking advantage of software bugs. In *R2L attacks,* attackers send packet to a victim machine to gain local access. They do this by using buffer overflows, weak security policies or guessing passwords. Obtaining root privileges without having the right is called *U2R attack*. These attacks can be done by using buffer overflows, weaknesses in path

name verifications or some software bugs. *Probe attacks* means scanning the network to exploit vulnerabilities. Probe usually is used before applying an attack to a system [37].

Table 2 Attribute name and index number of each record of the KDD Cup dataset

Index	**Attribute Name**	**Index**	**Attribute Name**
A1	duration	A22	is_guest_login
A2	protocol_type	A23	count
A3	service	A24	srv_count
A4	flag	A25	serror_rate
A5	src_bytes	A26	srv_serror_rate
A6	dst_bytes	A27	rerror_rate
A7	land	A28	srv_rerror_rate
A8	wrong_fragment	A29	same_srv_rate
A9	urgent	A30	diff_srv_rate
A10	hot	A31	srv_diff_host_rate
A11	num_failed_logins	A32	dst_host_count
A12	logged_in	A33	dst_host_srv_count
A13	num_compromised	A34	dst_host_same_srv_rate
A14	root_shell	A35	dst_host_diff_srv_rate
A15	su_attempted	A36	dst_host_same_src_port_rate
A16	num_root	A37	dst_host_srv_diff_host_rate
A17	num_file_creations	A38	dst_host_serror_rate
A18	num_shells	A39	dst_host_srv_serror_rate
A19	num_access_files	A40	dst_host_rerror_rate
A20	num_outbound_cmds	A41	dst_host_srv_rerror_rate
A21	is_host_login	A42	Attack

The number of attacks presented in the KDD Cup train and test sets and their corresponding class names are shown in Table 3.

Table 3 Number of attacks appeared in KDD Cup'99 Train and Test sets

Category	Attack Name	Train Set	Test Set
Normal	normal.	595,797	60,593
Remote to Local (R2L)	ftp_write.	8	3
	guess_passwd.	53	4,367
	httptunnel.	0	158
	imap.	12	1
	multihop.	6	18
	named.	0	17
	phf.	3	2
	sendmail.	0	17
	snmpgetattack.	0	7,741
	snmpguess.	0	2,406
	warezmaster.	20	1,602
	worm.	0	2
	xlock.	0	9
	xsnoop.	0	4
Probing	ipsweep.	7,579	306
	mscan.	0	1,053
	nmap.	2,316	84
	portsweep.	2,782	354
	saint.	0	736
	satan.	5,393	1,633
Denial of Service (DoS)	apache2.	0	794
	back.	2,002	1,098
	land.	17	9
	mailbomb.	0	5
	neptune.	204,815	58,001
	pod.	40	87
	processtable.	0	759
	smurf.	227,524	164,091
	teardrop.	199	12
	udpstorm.	0	2
User to Root (U2R)	buffer_overflow.	5	22
	loadmodule.	2	2
	perl.	2	2
	ps.	0	16
	rootkit.	0	13
	sqlattack.	0	2
	xterm.	0	13
	Total	1,048,575	311.029

2.1.2. NSL-KDD Dataset

KDD'99 Cup is created based on the data captured in DARPA'98. Tavallaee et al. [38] have analyzed the KDD'99 Cup and stated many inherent problems, such as:

1. The workload of the synthesized data is not likely to be same with the traffic in real networks.
2. There is no check mechanism whether any packet is dropped during traffic collection.
3. The definitions of the attacks are unclear.

Table 4 Statistics of randomly selected records from KDDTrain^{+} set [38]

Difficulty Level	Distinct Records	Percentage	Selected Records
0-5	407	0.04	407
6-10	768	0.07	767
11-15	6,525	0.61	6,485
16-20	58,995	5.49	55,757
21	1,008,297	93.80	62,557
Total	1,074,992	100.00	125,973

Table 5 Statistics of randomly selected records from KDDTest^{+} set [38]

Difficulty Level	Distinct Records	Percentage	Selected Records
0-5	589	0.76	585
6-10	847	1.10	838
11-15	3,540	4.58	3,378
16-20	7,845	10.15	7,049
21	64,468	83.41	10,694
Total	77,289	100.00	22,544

A new dataset called as NSL-KDD is created by revising the KDD'99 Cup dataset in [38]. Tavallaee et al. mentioned about problems that cause the evaluation results to be unreliable. For instance there are many redundant records in both KDD train and test sets. In NSL-KDD, all redundant records are deleted. They have also used seven different machine learning algorithms (each algorithm is applied three times) on the KDD'99 Cup dataset. A new column is added to the dataset and named as #successfulPrediction which is a number between 0 and 21. This column represents the number of machine learning algorithms that has correctly detected the corresponding record (the current row). In other words it reflects the difficulty level of detection in which zero is the most difficult one. The KDD'99 Cup train and test

datasets are resampled to new distributions with respect to the difficulty level and named as KDDTrain^{+} (see Table 4) and KDDTest^{+} (see Table 5), whereas both new datasets in general are called as NSL-KDD.

2.1.3. The ISCX 2012 IDS Dataset

The ISCX dataset [30] has been prepared based on different user profiles at the Information Security Centre of Excellence, University of New Brunswick, Canada. The attribute descriptions of the ISCX dataset are shown in Table 6. Over two millions real traffic packets consisting of HTTP, SMTP, SSH, IMAP, POP3, and FTP protocols were generated in seven days. Four different attack types named as Brute Force SSH, Infiltrating, HTTP DoS, and DDoS are conducted on different days as shown in Table 7. The percentage of attacks is 2.8% which makes it close to real world in where the percentage of attacks is observed around 1% [39].

Table 6 Descriptions of each attribute of the ISCX dataset

Attribute Name	**Type**	***Description***
generated	Date & Time	*Generation date and time of connection*
appName	Text	*Application Name*
totalSourceBytes	Integer	*Number of bytes send by source*
totalDestinationBytes	Integer	*Number of bytes send by destination*
totalDestinationPackets	Integer	*Number of packets send by destination*
totalSourcePackets	Integer	*Number of packets send by source*
sourcePayloadAsBase64	Text	*Payload as base64 send by source*
sourcePayloadAsUTF	Text	*Payload as UTF send by source*
destinationPayloadAsBase64	Text	*Payload as base64 send by destination*
destinationPayloadAsUTF	Text	*Payload as UTF send by destination*
direction	Text	*Direction of communication*
sourceTCPFlagsDescription	Text	*Active flags in TCP packets send by source*
destinationTCPFlagsDescription	Text	*Active flags in TCP packets send by destination*
source	IP Address	*Source IP Address*
protocolName	Text	*Name of Protocol*
sourcePort	Integer	*Port number in Source*
destination	IP Address	*Destination IP Address*
destinationPort	Integer	*Port number in destination*
startDateTime	Date & Time	*Start time of communication*
stopDateTime	Date & Time	*Stop time of communication*
Tag	Text	*Label (Attack, Normal)*

Even though minor disadvantages mentioned in [40] exist the ISCX dataset is still used to evaluate the proposed method, because it is the most applicable one compared to the other explored datasets.

Table 7 ISCX dataset distribution

Date	Normal	Attack	Name of Attack
11/6/2010	378,667	0	N/A
12/6/2010	131,111	2,082	Brute Force SSH
13/6/2010	255,170	20,358	Infiltrating
14/6/2010	167,609	3,771	HTTP DoS
15/6/2010	534,320	37,378	DDoS
16/6/2010	522,263	0	N/A
17/6/2010	392,392	5,203	Brute Force SSH
TOTAL	2,381,532	68,792	

2.1.4. The Kyoto Dataset

The Kyoto dataset [29] is a traffic data obtained from the Kyoto University's Honeypots from November 2006 to August 2009. Most of the traffic captured from the honeypots are observed as attack data, while only small amount of traffic was undefined. Therefore, all of the honeypot data are considered as attack data. On the other hand, normal traffic is generated through mail servers which were deployed into the same network with honeypots. Even though the mail server received small amount of attacks, they were considered as normal traffic [39]. The Kyoto dataset is considered a worthwhile dataset for research community [41]–[43]. However the dataset contains large amount of attacks and isn't labeled manually by human experts.

2.1.5. The CAIDA Dataset

CAIDA (Center for Applied Internet Data Analysis) collects various types of data and serves them to the research community. Most of the CAIDA security datasets contain only a particular attack. For instance, "DDoS Attack 2007", "Backscatter" and "Telescope Sipscan" datasets contain only DDoS, DoS, and UDP (User Datagram Protocol) Probing, respectively [31].

2.1.6. The LBNL Dataset

Lawrence Berkeley National Laboratory (LBNL) [32] released 11 GB of anonymized packet traces of LBNL's enterprise traffic. The LBNL, which are used to research on anonymizing traffic traces, are just basic captured network traces and don't include any attack.

2.1.7. The DEFCON Dataset

DefCon [33] contains only attacks that are created for competitions, which are conducted yearly. Since DefCon only contains attacks, it is not similar to real world network traffic.

2.1.8. The UNIBS Dataset

The UNIBS dataset [34] consists of network traces which are collected on the edge router of the campus network of the University of Brescia on 09/30/2009, 10/01/2009, and 10/02/2009. The dataset includes TCP and UDP traffic which composes of various protocols, such as HTTP, HTTPS, POP3, IMAP4, SMTP, FTP, SSH, and MSN. This dataset is not similar to real world network traffic, because it does not include any attack.

2.1.9. The TUIDS Dataset

Another publicly available dataset is the TUIDS [35] prepared at the Network Security Lab, Tezpur University, India. The dataset is based on different attack scenarios. However, it is different from real world network traffic since over 40% of the dataset are attacks whereas in real world the ratio of attacks are estimated about 1% [39].

2.2. **Feature Selection in IDS**

Intrusion detection systems usually deal with large amount of data with lots of features. However, some of the features may not represent an attack, therefore using these features will increase the training time and decrease the detection rate of the attacks [44]. Hence, different feature selection methods have been applied so far. Some common feature selection methods are Information Gain and Mutual Information [45].

In the literature, feature selection methods fall into two main categories, which are wrapper-based and filter-based feature selection.

2.2.1. Wrapper-Based Feature Selection

Wrapper method uses a search algorithm to create subsets of the features and evaluates each subset by running a model on it [46].

Sung and Mukkamala [47] used SVM and NN to rank the importance of the features of the KDD dataset. They did this by removing a feature at each step and starting the classifier to train and test with the reminder features. If the testing result increased the removed feature is insignificant, if the result decreased they considered that the removed feature is significant.

Another wrapper approach is applied by Hofmann et al. [48] to select features from the TCP/IP data of the 1998 DARPA IDS evaluation set. They have been focused on seven attacks (back, dictionary, guest, ipsweep, nmap, portsweep and warezclient) and used an evolutionary algorithm for architecture optimization of RBF networks to select the best network and the most related features for each attack type. They have been reduced the number of features to a range of 1-8 out of 137 features [48]. Another wrapper method is performed by Li et al. [49], where modified Random Mutation Hill Climbing (RMHC) is used to create feature subsets and modified linear Support Vector Machines (SVMs) is used to evaluate the created subsets for obtaining the optimum one.

2.2.2. Filter-Based Feature Selection

Filter methods are applied by using prior knowledge, such as the correlation between the feature and the target class [46].

Chebroul et al. [50] have applied Bayesian networks (BN), Classification and Regression Trees (CART), and an ensemble of BN and CART to select features.

Another work with feature selection has been done by Amiri et al. [51]. They presented two filter-based FS algorithms which are Linear Correlation-based Feature Selection (LCFS) and Modified Mutual Information-based Feature Selection (MMIFS). They compared their results with several other FS algorithms that had been previously presented in some other research papers. To classify the data, a reformulated Least Square SVM (LSSVM) algorithm is proposed. Another filter-based FS has been applied by Olusola et al. [52]. They selected features from the KDD Cup'99 dataset using rough set degree of dependency and dependency ratio.

Zargari and Voorhis [53] have used CfsSubsetEval for attribute evaluator and GreedyStepwise for search method. In their second experiment, they used InfoGainAttributeEval for attribute evaluator and Ranker for search methods which are tools in WEKA [54] for feature selection.

Tsang et al. [45] have applied both wrapper based and filter based FS methods and obtained over 20 subsets of features. Using these subsets, they have applied common classification algorithms (C4.5, NB, k-NN, SVM). Feature ranking algorithms such as Information Gain (IG), Gain Ratio (GR), Chi-Square (CS), and Relief-F have been applied in the filter approaches while Best First (BF), Forward Sequential Selection (FSS), Backward Sequential Selection (BSS), and Genetic Algorithm (GA) have been applied separately in the wrapper approaches. They also have proposed a Multi-Objective Genetic Fuzzy Intrusion Detection System (MOGFIDS) as a wrapper approach and compared it with the baseline wrappers.

2.3. **Types of IDS**

According to the detection model, IDSs can be separated into two major types: misuse-based IDSs (also called as signature-based) and anomaly-based IDSs [44], [55]. On the other hand, the IDS can be divided into two main types again this time according to the source of audit data: Host-based IDS (HIDS) and Network-based IDS (NIDS) [44]. NIDS is usually located between host and firewall, while HIDS is installed on the server or main computer which will be protected [7].

2.3.1. Misuse-Based IDS

Kumar has described a generic model of matching, based on CP-Nets for pattern matching in misuse intrusion detection approach [56].

In [57], a comprehensive set of pattern recognition machine learning algorithms (MLP, GAU, K-M, NEA, IRBF, LEA, HYP, ART, C4.5) have been performed on the KDD dataset. They have proposed a multi-classifier model derived from combining the algorithms that have the best result of each class. In this model; MLP, K-M, and GAU are suggested for Probe, DoS&U2R, and R2L attacks, respectively.

Kreibich and Crowcroft have presented a system that automatically generated signatures for the network attacks [58]. They called this system as Honeycomb which applies pattern-

matching techniques and protocol conformance checks the network traffics. The network packets are collected from a honeypot system.

2.3.2. Anomaly-Based IDS

Peddabachigari et al. [59] have randomly selected data from the KDD'99 dataset and divided it into two parts for training and testing. SVM and Decision Tree are used as classifiers and they have concluded with that Decision Tree gives better result than SVM.

In [60], three different algorithms have been applied on a randomly selected KDD'99 dataset. The algorithms that have been used are as follows: SVM, MARS (Multivariate Adaptive Regression) and ANN (Artificial Neural Network). The ANN algorithms that have been used in the experiments are Resilient Back-propagation (RP), Scaled Conjugate Gradient alg. (SCG), and One-Step-Secant alg. (OSS). They proposed an ensemble of ANN, SVM and MARS which gave higher overall accuracy than individually applied.

C4.5 is applied and enhanced by Bouzida et.al. [61]. The enhanced C4.5 says that if the data does not match any rule it is considered as a new class, with this approach they improved the prediction of new attacks.

Tavallaee et al. [38] have discussed the deficiencies of the KDD'99 dataset and proposed a new dataset (NSL-KDD) which is generated from the original KDD dataset and is publicly available. They have trained and tested both original and proposed datasets with the following algorithms; J48, Naive Bayes, NB Tree, Random Forest, Random Tree, Multilayer Perceptron (MLP), and Support Vector Machine (SVM).

Su et al. [62] have designed a real-time NIDS using incremental mining for fuzzy association rules. They first have trained their system using attack free data records and then tested the system with 30 DoS attacks in a real-time environment. The proposed system shows high accuracy and does not generate any false alarm.

Palomo et al. [63] have used Growing Hierarchical Self-Organizing Maps (GHSOMs) for IDS. They have proposed a new metric for GHSOMs in order to deal with both numerical and symbolic data thus to improve classification.

Another work with SVM is applied by Chen et al. in [7]. They have randomly selected the train and test data from the KDD'99 dataset and used the Rough Set Theory to reduce the features. They have applied SVM to train and test the system.

Horng et al. [64] have used BIRCH hierarchical clustering, which uses CF (Clustering Feature) Trees to reduce the KDD dataset, and then applied SVM. Their method have been able to shorten the training time and increase the classification rate of SVM based classification.

In [65], Bae et al. have applied the Artificial Bee Colony (ABC) Algorithm on IDS and compared their results with five other algorithms which are Naive Bayes, SVM, Classification Tree, kNN, and C4.5. They have used a randomly selected dataset from KDD'99 for their experiments. They showed that ABC algorithm is able to classify network intrusion datasets and outperforms other five popular benchmark classifies.

Wattanapongsakorn et al. [66] have proposed a network-based Intrusion Detection and Prevention System (IDPS) where online network data is classified using machine learning methods such as Decision Tree, Ripple Rule, Random Forest, and Bayesian Network. The dataset was captured from an online network data which consists of normal network activity and network attacks (DoS and Probing) created with some attack tools in a laboratory. Their experiments with the Decision Tree classification showed high detection rates. The proposed system is able to protect the system by blocking the corresponding IP (Internet Protocol) address or port number.

Another significant work was done by Waizumi at.al. [67]. They have reclassified the attacks based on the type of anomalies they create. The types of their new classifications are: 1- Anomaly in the amount of traffic and the range of communication, 2- Anomaly in communication procedures and 3- Anomaly in content of communication. They applied PCA (Principal Component Analysis) on the IDEAL dataset to classify the data according the proposed classes.

Lee et al. [68] have applied feature selection to improve the accuracy. They have discussed approaches to improve the efficiency of the real-time IDS model and reduce computational cost.

In [69], Janakiramanand and Vasudevan have introduced a distributed intrusion detection system architecture which uses Ant Colony Optimization (ACO) for scalability.

Anuar et al. [70] made a comparison of misuse-based and anomaly-based IDSs. They have used only 20 attributes out of 41 attributes from the KDD dataset and classified them with C5.0 (Decision Tree). They concluded that the Decision Tree is better at classifying Normal, DoS and R2L classes while the rule-based approach gives better results on Probe and U2R classes.

Tajbakhsh at.al. [71] have used fuzzy association rules for intrusion detection. They have proposed and implemented an intrusion detection framework. Both misuse and anomaly based detection are applied and compared with fuzzy association rules. Their method did not show promising detection rate for unseen attacks. However total detection rate and DRs for known attacks is significant while FAR is kept low.

2.3.3. Hybrid IDS

Horng et al. [64] have proposed an SVM based IDS. The hierarchical clustering algorithm is used to cluster the KDD Cup dataset. They have created separate SVM classifiers for each attack classes while each of the classifier works with its own feature set and CF (Clustering Feature) tree. The classifiers then are combined to build the IDS.

- Feature Selection: A basic feature selection method is applied. A classifier is applied first using the whole feature set. Then a feature is removed from the feature set and the classifier is applied again. If the performance of the classifier increases, the removed feature is considered as unimportant and vice versa.
- Pros: Their system is fast since the size of the dataset is decreased with hierarchical clustering. They used the whole KDD Cup dataset, that's why it is possible to compare their results with other studies. Separate feature sets are used for each attack classifier.
- Cons: The authors did not mention how they combined the classifiers, therefore it is not clear how a test dataset is tested. On the other hand the system still provides low detection rates for both R2L and U2R attacks.

Another hybrid classification method for IDS was introduced by Feng et al. [14]. They have proposed an algorithm that combines SVM method with clustering based on Self-Organized Ant Colony Network (CSOACN).

- Feature Selection: Not applied.

- Pros: SVM classifier can update itself with new dataset. It is shown that the new method outperforms pure SVM and CSOACN.
- Cons: In this study the full KDD 10% dataset is used as the test set while 20% of the same dataset is used as the train set. However the KDD 10% dataset is widely used only as the train set while a test set , named as "corrected", is already available in [27], and contains new attacks that are not included in the train set. Therefore the results of this study cannot be compared with other studies as well as with the study conducted in this thesis since it would be unfair as the results belong to different test sets.

Govindarajan and Chandrasekaran presented two classification methods (MLP and RBF), and an ensemble of both classifiers. A hybrid method which has been combined with the bagging technique has been proposed [72].

- Feature Selection: Not applied.
- Pros: They showed that an ensemble of MLP and RBF gives higher accuracy rates than individual classifiers. It has been also proven that ensemble of MLP is superior to ensemble of RBF for normal records while it is vice versa for abnormal records.
- Cons: The dataset that has been used is very narrow and created from a limited set of programs in a single environment. Therefore it is difficult to generalize and compare the proposed method with other studies. Two ensemble models are proposed but none of them is giving high detection rate in detecting both normal and abnormal records. Ensemble of MLP shows high accuracy for normal and low for abnormal and the ensemble of RBF shows high accuracy for abnormal and low for normal records.

A multi-level hybrid classifier with four stages, similar to [73] with three stages, has been proposed by Xiang et al. [13]. Their classifier combines Decision Tree and Bayesian clustering. Bayesian clustering is used in stage two while Decision Tree is used in other three stages. The KDD Cup'99 dataset is used for the experiments.

- Feature Selection: Feature selection is applied using Information Gain, AutoClass algorithm, and Bayesian clustering.
- Pros: Their study is similar to [73], a three level classifier, and can be considered as an improvement of [73]. Feature selection is applied for each level separately. The results provide relatively high detection rates and low false alarm rate of 3.2%.
- Cons: The detection rate of R2L attacks is very low.

An ensemble of ANNs, SVM, and MARS has been proposed by Mukkamala et al. [60]. They have created five classifiers and used them simultaneously. A majority voting approach, in which the detected class is the one where most of the classifiers agreed, is used.

- Feature Selection: Not applied.

- Pros: It has been showed that the hybrid classifier gives higher detection rates than the individual classifiers.
- Cons: The data used in the experiment is randomly generated from the KDD Cup'99 dataset, therefore it is not possible to compare the results with other studies.

Chebrolu et al. [50] have proposed an ensemble IDS model which includes Bayesian Network (BN) and Classification and Regression Trees (CART) classifiers. Each classifier is given a weight according their accuracies. If both classifiers agree then the decision is given accordingly. If there is a conflict then the decision is given by the classifier with the highest weight. They have also used BN and CART to select significant features.

- Feature Selection: BN and CART is used for feature selection.
- Pros: The performance of two feature selection methods are investigated. A basic ensemble model with high accuracy is proposed.
- The dataset is randomly selected from the KDD Cup'99. That's why it is not possible to compare the result with other studies.

Lastly, Aydın et al. [74] have combined Packet Header Anomaly Detector (PHAD) and Network Traffic Anomaly Detector (NETAD), which are anomaly-based approaches. Then, they have applied them as a pre-processor to Snort which functions as a signature-based IDS. DARPA Intrusion Detection Evaluation Dataset (IDEVAL) dataset is used to evaluate the hybrid method while no feature selection method is applied.

- Feature Selection: Not applied.
- Pros: Signature-based IDS provides high detection rates for known attacks with low FARs while Anomaly-based IDS is able to detect unknown attacks. Since this study combines Anomaly-based and Signature-based IDSs, it gains the advantages of both methods.
- Cons: No feature selection is applied, however it is possible that some of the features are unrelated. This can cause longer training time and decrease the classification performance.

In the previous studies, none of the hybrid IDSs have applied a wrapper based FS method while few of them applied filter based FS. Also, only a few of them have applied the entire KDD Cup '99 10% and the entire test set which are important for comparing the results with other studies.

2.4. Challenges of NIDS

Since most of the research is based on anomaly detection and, moreover, the proposed work is in anomaly-based NIDS, this thesis will only focus on the challenges related to anomaly detection.

2.4.1. False Positive Alarms

One of the important drawbacks is the high false positive rates, which is the rate of falsely raised security alarms [18], [75]–[77]. FARs make it difficult to manage the security issues on a network. It is mentioned that for an effective system the maximum requirement of FAR is 0.001% [78].

2.4.2. Dataset

The KDD'99, the most popular dataset used for evaluating IDSs, is more than 15 years old. On the other hand, every day new attacks are launched and many previous attacks are getting outdated. Therefore it is difficult to tell that a proposed system which shows high performance on the KDD'99 dataset shows the same performance on current attacks [16]. For this reason it is very important to frequently update the datasets for accurate evaluations [18], [76]. There is also a need of test-beds to provide reliable metrics for evaluating NIDSs [77].

2.4.3. Performance Issues

Since the audit data is very large it is difficult to train and test the system, thus this makes it difficult to build an effective IDS [16]. Therefore some researchers do not use the whole dataset, instead they randomly select data from the train and test datasets to use them on their system. Some others use divide-and-conquer algorithms or utilize distributed environments. On the other hand, since the system has to be trained again for every updated dataset, the long training time and the difficulty of preparing a new dataset makes the Anomaly-based IDS impractical in real environments.

2.4.4. IDSs Defending Themselves

Most of the Intrusion Detection Systems have issues in defending themselves from attacks [77], [79]. This is another challenge which is still under research.

2.4.5. Evaluation Methodologies

Most of the evaluations are done with Receiver Operating Characteristic (ROC) curve; however there are lots of critics of the opinion that ROC curve is misleading. Therefore, the lack of a global standard and metric for evaluation of IDSs is another important problem in this area [76], [77].

3. MACHINE LEARNING ALGORITHMS

Machine learning is a field of scientific study that focuses on algorithms to "learn" in order to make predictions on data [80]. In [81], machine learning algorithms are categorized as five types: Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, Reinforcement Learning, and Inductive Learning (or Transfer Learning).

In this chapter, Supervised Learning and Transfer Learning have been described and brief overviews of the related algorithms used in this thesis are given.

3.1. **Supervised Learning**

In supervised learning the method provides the system to be trained with some inputs and target outputs. The system creates an activation function that gives the relation of inputs and outputs and predicts unseen data by using this function.

3.1.1. Backpropagation Algorithm

Backpropagation (BP) algorithm is a widely used supervised method in Artificial Neural Networks (ANN). This algorithm requires desired output(s) for each set of input in order to update and obtain the optimal weights and optimal function that can produce the desired output(s). A common example [82], used to describe this algorithm, is the XOR problem as shown in Table 8.

Table 8 XOR input and outputs

x_1	x_2	f(x)
0	0	0
0	1	1
1	0	1
1	1	0

As it can be seen in Table 8, the XOR problem has two inputs (x_1 and x_2) and one output (f(x)). For that reason the ANN structure has two inputs at the input layer and one output at the output layer. The number of nodes used in the hidden layer can be chosen to be any value, while in this illustration two nodes are chosen in the hidden layer. The ANN structure of the XOR problem is shown in Figure 2 and the steps of the BP are as follows:

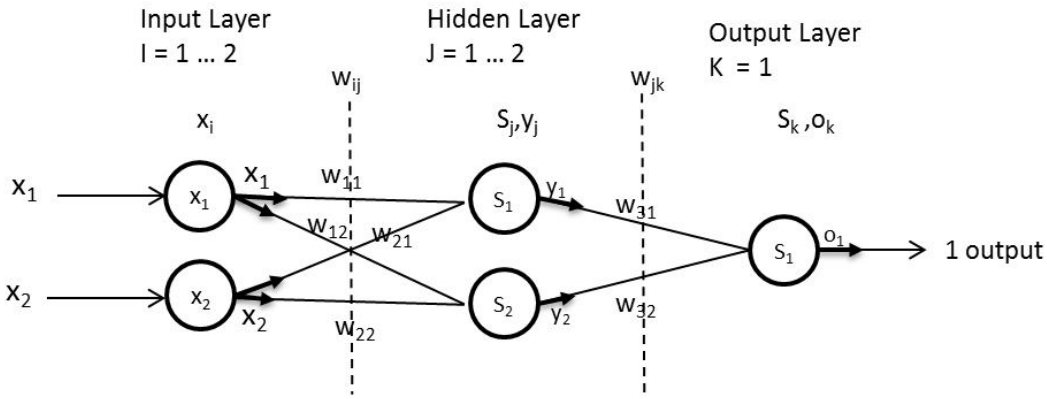

Figure 2 The ANN structure for the XOR problem

Step 1: Random initial values are assigned to weights (w_{ij}).

Step 2: Calculate S_j and y_j values for each node (where S is the sum of input×weight)

$$S_j = \sum_{i=1}^{I} x_i \times w_{ij} \quad (1)$$

$$y_j = f(S_j) = \frac{1}{1+e^{-S_j}} \quad (2)$$

Where;
I: number of nodes in the input layer
f(S_j): Sigmoid function

Step 3: Calculate y_k and S_k values (where o_k is the calculated or predicted output)

$$S_k = \sum_{j=1}^{J} y_j \times w_{jk} \quad (3)$$

$$o_k = f(S_k) \quad (4)$$

Where;
J: number of nodes in the hidden layer
f(S_k): Sigmoid function

Step 4: The Root Mean Square (RMS) error is calculated.

$$E = \frac{1}{2}\sum_{k=1}^{K}(o_k - d_k)^2 \quad (5)$$

Where;
K: number of nodes in the output layer
d_k: desired output
o_k: calculated output

Step 5: In order to obtain optimal output values (close to desired outputs) the RMS need to be minimized by updating the weights. The minimum point of the RMS (see Figure 3) can be found with the following equation:

$$\Delta w_{\mathrm{jk}} = \varepsilon \frac{\partial E}{\partial w_{jk}} \tag{6}$$

Where;

ε: learning rate (usually selected 0.01 – 0.9) which defines the step length of each iteration [82].

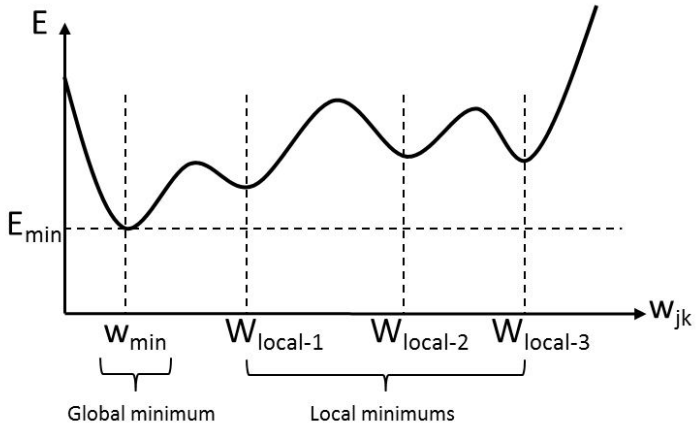

Figure 3 A sample RMS vs weight graph

By applying equation (6) the weights are going to be updated. The equation is simplified to the following equations; more detailed information can be found in [82]–[84]:

$$\delta_{\mathrm{o}} = (d_{\mathrm{k}} - y_{\mathrm{k}}) y_{\mathrm{k}} (1 - y_{\mathrm{k}}) \tag{7}$$

$$\delta_{\mathrm{y}} = y_{\mathrm{j}} (1 - y_{\mathrm{j}}) \sum_k \delta_{\mathrm{o}} w_{\mathrm{jk}} \tag{8}$$

$$\Delta w_{\mathrm{ij}}(t+1) = \varepsilon\, \delta_{\mathrm{y}}\, x_{\mathrm{i}} + \propto \Delta w_{\mathrm{ij}}(t) \tag{9}$$

$$\Delta w_{\mathrm{jk}}(t+1) = \varepsilon\, \delta_{\mathrm{o}}\, y_{\mathrm{j}} + \propto \Delta w_{\mathrm{kj}}(t) \tag{10}$$

α: exponential decay factor (btw. 0-1) that determines the relative contribution of the current gradient and earlier gradients to the weight change [85].

These steps are going to be repeated until it reaches to a pre-set mean square error or a selected iteration value (details are in [85]). In this thesis, artificial neural networks which use the BP are written in the C programming language.

3.1.2. Decision Tree Classifier

Decision Trees are one of the most used tools for classification and prediction. It consists of a tree structure where each node is either a leaf node or a decision node. A leaf node indicates the decision of the target class while a decision node contains a condition statement with sub-trees for some attributes [86], [87]. In this section both ID3 and C4.5, which are the most widely used Decision Tree algorithms and introduced by Quinlan, are summarized.

3.1.2.1. The ID3 Algorithm

Let's consider a train set given in Table 9 to demonstrate how the ID3 algorithm works. The dataset has four attributes (Outlook, Temperature, Humidity and Windy) and one output (Class) which is either N (not play) or P (play).

Table 9 Train data of playing tennis

ID	Attributes				Class
	Outlook	**Temperature**	**Humidity**	**Windy**	
1	sunny	hot	high	false	N
2	sunny	hot	high	true	N
3	overcast	hot	high	false	P
4	rain	mild	high	false	P
5	rain	cool	normal	false	P
6	rain	cool	normal	true	N
7	overcast	cool	normal	true	P
8	sunny	mild	high	false	N
9	sunny	cool	normal	false	P
10	rain	mild	normal	false	P
11	sunny	mild	normal	true	P
12	overcast	mild	high	true	P
13	overcast	hot	normal	false	P
14	rain	mild	high	true	N

The Decision Tree learned with the train set in Table 9 and is demonstrated in Figure 4. This tree classifies Saturday mornings whether playing tennis is suitable or not [87]. This part is called the *induction task*. It is possible to obtain different Decision Trees that are all true for the train set. But the Decision Tree is expected to also work for unseen inputs (test set).

Therefore it is recommended to choose the simpler one which is likely to have meaningful relationship between the class value and the attributes. When there are many attributes and many instances, calculating each possible Decision Tree and selecting the simplest one is a costly process in terms of computation. For that reason, the ID3 algorithm is designed to obtain a reasonably good Decision Tree without much computation [88].

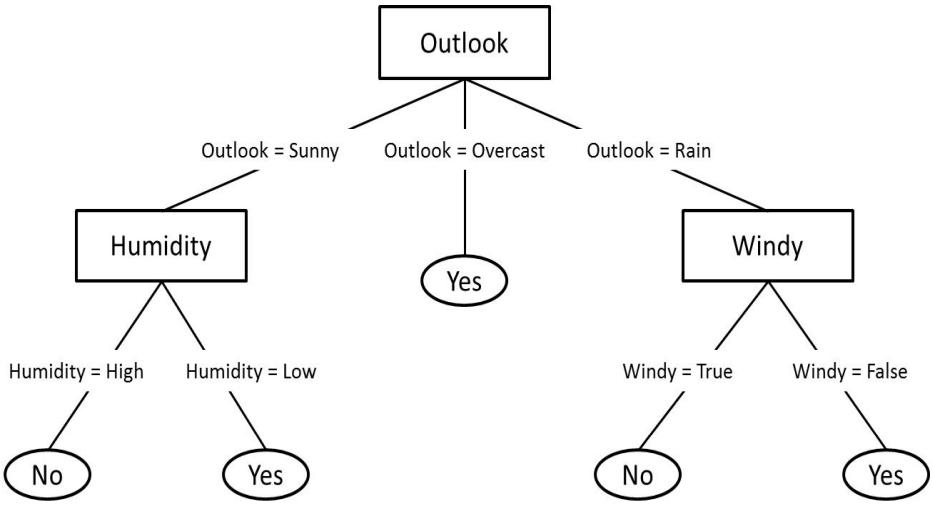

Figure 4 A Decision Tree for play tennis data [87]

The ID3 algorithm is an iterative algorithm (Figure 5) in which a random subset of the train set is used to build the Decision Tree. If the Decision Tree works correctly for the rest of the train set this Decision Tree is considered as the correct Decision Tree and the process is terminated. Otherwise the incorrectly classified train set is added to the subset of the train set and a new Decision Tree is built by using the new subset of train set [88]. With this method, the correct Decision Tree is obtained only after few iterations. But it should be noted that ID3 may misclassify data.

In ID3, *information gain* is calculated (at each step) to select the best attribute that is most useful for classifying data. For that, first the Entropy is calculated that defines the purity of a collection of data [89].

$$Entropy\ (S) = \sum_{i=1}^{c} -p_i \log_2 p_i \tag{11}$$

where p_i is the proportion of S belonging to class i and S is the train set. The Information Gain is calculated by using the Entropy as shown below:

$$Gain(S, A) = Entropy(S) - \sum_{v=values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (12)$$

where *values(A)* is the set of all possible values of for attribute A, and S_v is the subset of S where the attribute A has the value of v.

```
function ID3
Input:   (R: a set of non-target attributes,
         C: the target attribute,
         S: a train set) returns a Decision Tree;
begin
If S is empty, return a single node with value Failure;
If S consists of records all with the same value for the target attribute,
return a single leaf node with that value;
If R is empty, then return a single node with the value of the most frequent of the values of
the target attribute that are found in records of S; [in that case there may be be errors,
examples that will be improperly classified];
Let A be the attribute with largest Gain(A,S) among attributes in R;
Let {aj| j=1,2, .., m} be the values of attribute A;
Let {Sj| j=1,2, .., m} be the subsets of S consisting respectively of records with value aj for
A;
Return a tree with root labeled A and arcs labeled a1, a2, .., am going respectively to the
trees (ID3(R-{A}, C, S1), ID3(R-{A}, C, S2),.....,ID3(R-{A}, C, Sm);
Recursively apply ID3 to subsets {Sj| j=1,2, .., m} until they are empty
End
```

Figure 5 ID3 Decision Tree algorithm [90]

3.1.2.2. The C4.5 Algorithm

The standard ID3 algorithm has some deficiencies such as over-fitting in the trees, not able to handling with continuous valued attributes, and missing attribute values. The C4.5 algorithm is an extension of ID3 that deals with the aforementioned deficiencies [89].

The C4.5 algorithm avoids over-fitting in Decision Trees by applying post-pruning on the tree. To test the pruning, some examples are separated from the training examples and used on the pruned tree to evaluate it.

The algorithm also deals with continuously valued attributes by converting them into distinct values by using threshold values. The threshold value is usually selected by sorting the data according to the continuous valued attribute and then finding adjacent examples that differ in

the target classification by calculating the information gain. The candidate threshold that gives the highest information gain is selected.

Lastly the C4.5 deals with missing values in the attributes by calculating the probability of each possible values of A and replacing the missing values with the highest probable values [89].

In this thesis, the WEKA 3.6 software [54] has been used to apply the C4.5 algorithm.

3.1.3. Radial Basis Function (RBF)

Radial Basis Function (RBF) (see Figure 6) is a type of ANN classification method of supervised learning [91]. A typical RBF has three layers: input layer, hidden layer, and output layer.

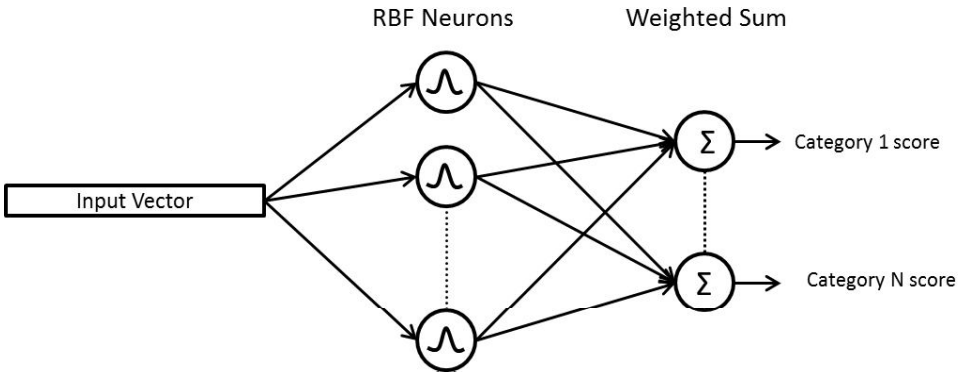

Figure 6 RBF network architecture

The hidden layer has a non-linear radial basis activation function. Each node in the hidden layer contains a prototype vector and calculates the distance between the inputs and the prototype. The node then outputs either 1 or 0 according to the distance; if the input is close to the prototype then the output will be 1 and if the distance grows the output falls off exponentially towards 0 [92]. In other words if the input is equal to the prototype the RBF neuron will result with its highest value.

The output layer has multiple nodes each for one category. It performs linear regression to predict the outputs as shown in Equation (13).

$$h(\mathrm{x}) = \sum_{n=0}^{N} w_n \exp\left(-\gamma \|x - x_n\|^2\right) \tag{13}$$

where $x_n, x, \gamma,$ and w_n are the prototype vector, input vector, a coefficient, and the weights respectively. The γ coefficient controls the width of the bell curve while the w_n controls the height of the Gaussian and applied by the output nodes.

In this thesis MATLAB 2010b software has been used for the RBF functions.

3.1.4. Support Vector Machines (SVM)

Support Vector Machines (SVM) is another supervised learning model which does classification and regression by a separating hyperplane. It was introduced by Boser, Guyon, and Vapnik at the Computational Learning Theory (COLT) conference in 1992. The SVM basically reads the train data and outputs an optimal hyperplane which categorizes new examples.

A 2-D example of a dataset can be shown in Figure 7. As it can be seen in this figure, there are multiple possible solutions to separate the examples. SVM tries to find the optimal solution (separating line).

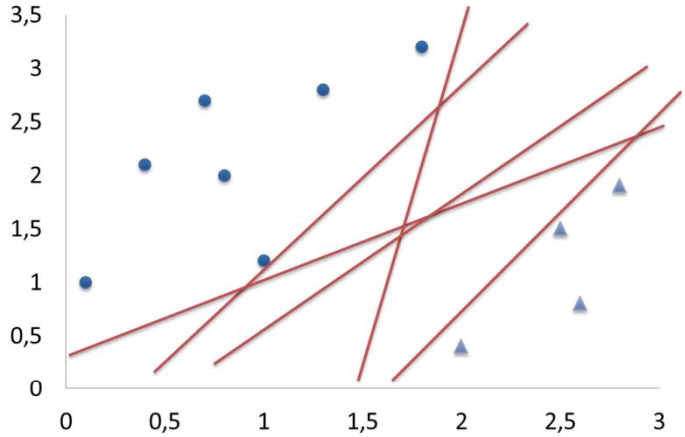

Figure 7 A 2-D example of many hyperplanes separating the data in 2-D

The optimal separating line is the one that has the largest distance to all points. This distance is also called ***margin*** which is an important variable in SVM theory. Therefore having the maximal margin gives the optimal hyperplane.

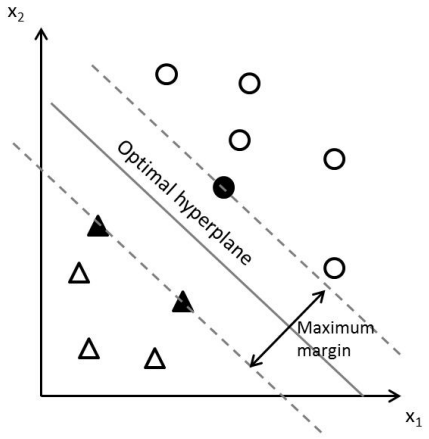

Figure 8 Demonstration of an optimal hyperplane in 2-D

The training examples that are closest to the hyperplane (see the filled vectors in Figure 8) are generally called ***support vectors***. The hyperplane is calculated as the following function:

$$w^T x + b = 0 \tag{14}$$

where *w*, *b* and *x* are the weight, weight vector and the training example, respectively. The boundaries (the dashed lines on Figure 8) on where the support vectors are intersecting should have the maximum distance in between, in order to have the optimal hyperplane. The maximum distance is obtained by minimizing the following equation [93]:

$$Maximum\ distance = \frac{w^T w}{2} \tag{15}$$

After obtaining the optimal hyperplane the testing vectors will be tested with the obtained function. Anything above and below the hyperplane will be classified as 1 and -1, respectively. This method is used when the dimension of the examples is higher than two.

3.1.5. Naive Bayes Classifier

Naive Bayes is a probabilistic supervised classifier that is based on the Bayes Theorem [94] and can be efficiently used in supervised learning. The Bayes Theorem is shown below:

$$P(C|F) = \frac{P(F|C) \times P(C)}{P(F)} \tag{16}$$

where *C* is the class and *F* is the feature variable. This equation calculates the probability of feature variable *F* being in class *C*.

In Naive Bayes, it is assumed that all features are independent from each other [94], therefore the Bayes rule for multiple features is written as below:

$$P(C|F_1, F_2, \ldots, F_n) = \frac{P(F_1|C) \times P(F_2|C) \times \ldots \times P(F_n|C) \times P(C)}{P(F_1, F_2, \ldots, F_n)} \quad (17)$$

which calculates the probability of class C when it is known that the features are $F_1, F_2, \ldots, F_n$. Naive Bayes classifies the data according their probabilities under a given feature set.

3.1.6. Genetic Algorithms

Genetic algorithm (GA) is a search algorithm that simulates a natural selection to optimize a problem [95], [96]. GA is widely applied on diverse areas such as machine learning, chemistry, economy, algebra, music generation, and strategy planning [97], [98].

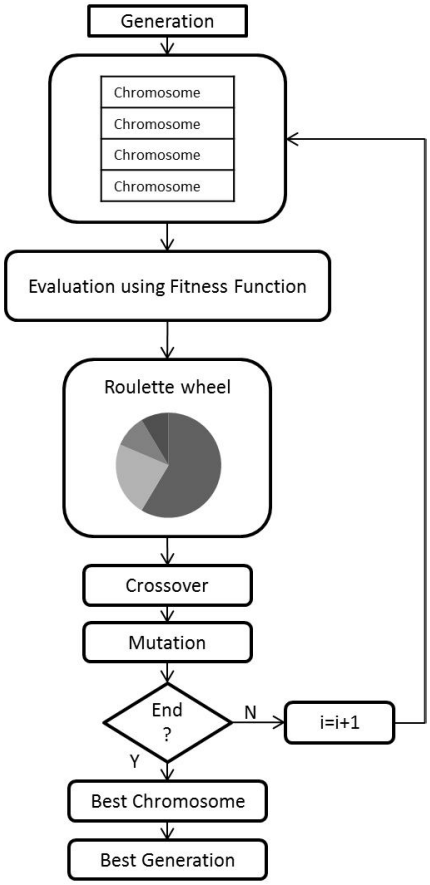

Figure 9 Genetic algorithm flowchart [99]

In GA a population of candidate subsets is evolved to obtain candidate solutions also called as individuals. Each individual consists of genes that can be either numerical or binary values

[99]. A fitness function is used to measure the suitability of the solutions. The solutions with the best fitness values have higher probability to be selected with the roulette wheel for the next generation [99]. Then crossover is applied on some individuals of next generation where each individual is selected under a pre-determined probability value (or *crossover rate*) [98]–[100]. Also the crossover point is determined randomly for each crossover pair. Then, mutation, in which a selected gene is replaced by a random value, is applied on some genes determined by a pre-determined *mutation rate parameter*. As a result, a new generation is created. The whole process (creating a new generation) is going to be repeated until a pre-determined iteration or fitness value is met, see Figure 9.

3.2. **Transfer Learning**

Traditional data mining and machine learning algorithms use labeled or unlabeled data to train the system and perform predictions on new data that has unknown class labels [24], [100], [101]. It is assumed that both test and train data are in the same distribution. However in many real-time applications it is observed that the distribution of test sets differ from the distribution of the train sets [24], [25], [102], [103]. Transfer learning allows using different domains, tasks, and distributions to be used in training and testing [24], [25]. The motivation of transfer learning comes from the fact that people can solve new problems faster or better by making use of previously learned knowledge [24].

In real time applications once the data is outdated, new data should be re-collected and the system should be retrained from scratch using a new dataset [103]. This process is a costly process and throwing old data is a waste of time. However transfer learning provides the system to make use of previous knowledge [24], [25], [104]. This helps to make use of previous dataset and the system does not have to be trained all over again from scratch. Moreover the system can be trained only with few up-to-date train data by making use of the previous knowledge [24].

3.2.1. Types of Transfer Learning

Transfer learning is summarized into three main categories according to the relations between the source and target domains and tasks [24]; inductive transfer learning, transductive transfer learning, and unsupervised transfer learning.

Inductive transfer learning is applied to systems where the source and target tasks are different. In this system the source and target domains may be either same or different while the data in the source domain is either labeled or unlabeled. Few labeled data in the target domain are required [24]. Web mining, where the web pages are classified, can be given as an example. In web mining, collecting and labeling new data is a costly process, while the collected datasets can be easily outdated because of the frequent change in the content of the web [25]. This causes a change in the distribution of the train and test data. In [25], a small amount of labeled data from the target domain is used with the old train data from the source domain. This method aims to transfer knowledge from the old data to the new one.

In *transductive transfer learning*, the source and target domains are the same while the tasks are different. There is also no labeled data in the target domain while lots of labeled data is available in the source domain [24]. A sample transductive transfer learning problem is presented in [105]. In this example, they applied name entity recognition in which the source domain is a corpus of encyclopedia articles with many labeled personal names and the target domain is the instant messenger data. The target source domain does not have any labeled data, which is also called as unlabeled test data, but the unlabeled test data is made available during training phase to be used with the train data [105].

In *unsupervised transfer learning*, no labeled data is available in both source and target domains. The source and target tasks are different but related. The main focus of this transfer learning is to solve unsupervised learning tasks such as clustering images [24].

3.2.2. Genetic Transfer Learning with ANN

In genetic transfer learning the first generation is usually created randomly. In this thesis, the ANN is used in the genetic transfer learning. Thus the first generation is created from the weights which are obtained during the training phase in ANN. Furthermore, the fitness function is the backpropagation algorithm, in which the best fitness value is zero representing the error rate.

For instance let's assume that the structure of the ANN is as shown in Figure 2 and it is trained with 10 iterations. Six weights (w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32}) are obtained during each iteration. If the weights of each iteration are saved; the dimensions of the generation will be 10 x 6 where each set of 6 weights is called as *individual* (see Figure 10).

w^1_{11}	w^1_{12}	w^1_{21}	w^1_{22}	w^1_{31}	w^1_{32}	individual
w^2_{11}	w^2_{12}	w^2_{21}	w^2_{22}	w^2_{31}	w^2_{32}	
w^3_{11}	w^3_{12}	w^3_{21}	w^3_{22}	w^3_{31}	w^3_{32}	
w^4_{11}	w^4_{12}	w^4_{21}	w^4_{22}	w^4_{31}	w^4_{32}	
w^5_{11}	w^5_{12}	w^5_{21}	w^5_{22}	w^5_{31}	w^5_{32}	
w^6_{11}	w^6_{12}	w^6_{21}	w^6_{22}	w^6_{31}	w^6_{32}	
w^7_{11}	w^7_{12}	w^7_{21}	w^7_{22}	w^7_{31}	w^7_{32}	
w^8_{11}	w^8_{12}	w^8_{21}	w^8_{22}	w^8_{31}	w^8_{32}	
w^9_{11}	w^9_{12}	w^9_{21}	w^9_{22}	w^9_{31}	w^9_{32}	
w^{10}_{11}	w^{10}_{12}	w^{10}_{21}	w^{10}_{22}	w^{10}_{31}	w^{10}_{32}	

Figure 10 One generation obtained from the ANN training with 10 iterations ($w^x_{y,}$ x: iteration number, y: index number of weight)

The rest of the process is the same with genetic transfer. The weights are going to be used as the initial generation. This generation will be passed through the genetic algorithm process (Figure 9) and a new generation is going to be created. Before killing the old generation individuals with the best, median and worst fitness values will be saved in the solution pool.

4. DISCERNIBILITY FUNCTION BASED FEATURE SELECTION

Some of the purposes of this thesis are to improve the classification accuracy of the KDD Cup dataset, to decrease the classification cost of the machine learning algorithms and to lighten the IDS. Therefore feature selection (FS), which finds the optimal minimum feature subset that also represents the original dataset, is applied. Typically, one dataset has a lot of Minimal Subsets (MSs). The discernibility matrix-based approach is capable of generating all MSs of features. By using a discernibility-matrix based program, for instance the Rough Set Exploration System (RSES) exhaustive search approach, 2N-1 subsets need to be generated and verified in order to obtain all MSs for a dataset with N features. Generating 2N-1 subsets is a problem of an exponential complexity of N. This causes to consume huge memory and time, even for a medium sized dataset [106].

Rough Set Theory based FS is previously used in other IDS related studies [107], [108] in which the minimal subsets are obtained with various programs without computing all MSs. Therefore it is possible to overlook the optimal ones. One of the goals of this study is to compute all MSs and find the best MSs without the risk of losing the optimal ones. Since this is a problem with high computational complexity, a new method, called Decision Relative Discernibility Function-Based FS and proposed by Kahramanli et.al. [106], is applied which obtains the same results faster and with less memory need than the regular Rough Set method. In this thesis, this method is called as the Discernibility Function Based Feature Selection (DFBFS) method. The DFBFS is used as a wrapper based FS. This is the first study that applies the DFBFS on intrusion detection datasets. Applying the DFBFS will generate all possible MSs of features in which unrelated features of intrusion detection datasets are discarded. After that, optimal feature subsets will be evaluated and selected with BP algorithm. The comprehensive description of the DFBFS method that is applied in this thesis can be found in [106], while the brief description is given in this section:

A dataset is represented as $S = \{E, C \cup D\}$, where $E = \{E_a\}_{a=1}^{K}$ is a set of examples, $C = \{C_b\}_{b=1}^{L}$ is a set of condition features, and $D = \{D_c\}_{c=1}^{M}$ is a decision feature. Table 10 shows an example of a dataset with eight examples $E = \{E_1, E_2, \ldots, E_8\}$, four condition features $C = \{C_1, C_2, C_3, C_4\}$, and one decision feature $D = \{D\}$.

Table 10 An example dataset

Examples	C1	C2	C3	C4	D
E1	1	0	2	2	0
E2	0	1	1	1	2
E3	2	0	0	1	1
E4	1	1	0	2	2
E5	1	0	2	0	1
E6	2	2	0	1	1
E7	2	1	1	1	2
E8	0	1	1	0	1

DFBFS method selects the subsets of features with minimal size that contain relevant features for the dataset. The DFBFS method, which is explained in detail in [106], has three steps, as shown in Figure 11.

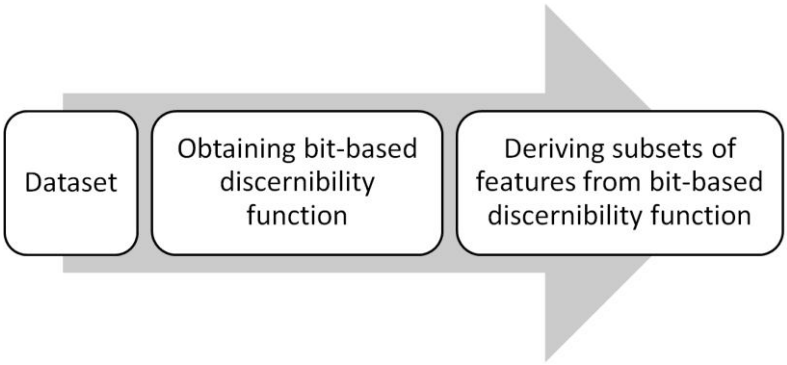

Figure 11 The flow-chart of the DFBFS method

4.1. **Obtaining Bit-Based Discernibility Function**

A discernibility matrix is a symmetric $K \times K$ matrix with entries h_{jk} [109]. Each entry consists of a set of attributes upon which examples E_j and E_k differ.

$$h_{mn} = \{C_i \in C \mid C_i(E_m) \neq C_i(E_n)\},$$

$$i \in \{1,2,\dots,L\} \text{ and } m,n \in \{1,2,\dots,K\} \quad (18)$$

Each h_{mn} contains attributes that differ between examples E_m and E_n. A discernibility function f_D is a boolean function of m boolean variables $C_1^*,\dots,C_Z^*$, defined as below [110]:

$$f_D(C_1^*,\dots,C_Z^*) = \wedge\{\vee h_{mn} \mid 1 \leq m \leq n \leq |S|, h_{mn} \neq \emptyset \quad (19)$$

The code-based form of discernibility function of the dataset shown in Table 10 is:

$$f_D(C_1, C_2, C_3, C_4) = \{C_1 \vee C_2 \vee C_3 \vee C_4\} \wedge \{C_1 \vee C_3 \vee C_4\} \wedge \{C_4\} \wedge \{C_1 \vee C_2 \vee C_3\} \wedge \{C_1 \vee C_2 \vee C_4\} \wedge \{C_2 \vee C_3 \vee C_4\} \wedge \{C_1 \vee C_4\} \quad (20)$$

In this study, the bit-based form of the discernibility function $B_b f_D$, defined as below, is used.

$$b_{mnj} = 1 \; if \; C_i(E_m) \neq C_i(E_n), \; b_{mnj} = 0 \; if \; C_i(E_m) = C_i(E_n) \quad (21)$$

$$B_b f_D = \{b_{mn1}, b_{mn2}, \dots, b_{mnn}\} \quad (22)$$

The $B_b f_D$ of the dataset, shown in Table 10, is obtained as follows:

$$B_b f_D = \{1111, 1011, 0110, 0001, 1110, 1101, 0111, 1001\} \quad (23)$$

$B_b f_D$ may have some redundant elements and these elements slow down the feature selection process. For accelerating this process, the $B_b f_D$ needs to be simplified by deleting redundant elements from $B_b f_D$. The *"AND"* Boolean operator has been used between all elements of $B_b f_D$ for determining the elements to be deleted [111]. This deletion is performed according to the equation shown below:

$$if \; b_{mnj} \& \; b_{mnj+1} = b_{mnj} \; then \; delete \; b_{mnj+1}, \; j = \{1,2, \dots, n\}$$

$$if \; b_{mnj} \& \; b_{mnj+1} = b_{mnj+1} \; then \; delete \; b_{mnj}, \; j = \{1,2, \dots, n\} \quad (24)$$

As a result, the minimized form of the $B_b f_D$ is obtained as follows:

$$B_b f_D = \{0110, 0001\} \quad (25)$$

4.2. **Deriving Subset of Features**

The subsets of the datasets can be obtained by converting the Equation (25) from conjunctive normal form (CNF) to disjunctive normal form (DNF). Therefore, each element in the $B_b f_{D_{min}}$ should be expanded. Each element of $B_b f_{D_{min}}$ preserves all information of the dataset because each bit is associated with one condition feature in the dataset shown in Table 10. Each element of the $B_b f_{D_{min}}$ can be expanded as follows:

$$E(0110) = \{0100, 0010\} \quad (26)$$

$$E(0001) = \{0001\} \tag{27}$$

$$E\left(B_b f_{D_{min}}\right) = \{\{0100, 0010\}, \{0001\}\} \tag{28}$$

For creating the subsets of features in DNF, *"OR"* boolean operator is used between the elements of $E\left(B_b f_{D_{min}}\right)$ as the following expression:

$$DNF(S) = \mathop{|}_{q=1}^{Q} E\left(B_b f_{D_{min}}\right) \tag{29}$$

For Equation (28), the DNF of the dataset given in Table 10 can be obtained as following:

$$DNF(S) = \{0100, 0010\} \mid \{0001\} = \{0101, 0011\} \tag{30}$$

$DNF(S)$ has subsets of features of the dataset given in Table 10. One of these subsets can be used instead of the original datasets because these subsets define the original dataset and do not contain irrelevant or redundant features. To show the subsets in a code-based form, the 1's positions can be associated with the condition features indicated in the Table 10 as the following:

$$Subsets = \{0101, 0011\} \rightarrow \{(C_2, C_4), (C_3, C_4)\}$$

It can be seen that instead of using four features for the given example dataset, two feature subsets each with two features are obtained. Each feature subset given above is able to represent the original dataset.

5. TYPES OF ATTACKS

In this chapter, descriptions of the attacks that appear in the KDD Cup and ISCX datasets are presented in detail.

5.1. Description of Attacks that Appear in the KDD Cup Dataset

The Table 11 shows the exploits of three different operating systems used in 1998 DARPA test-bed and groups them into four categories. These categories are Denial of Service (DoS), Remote to Local (R2L), User to Root (U2R) and Probing.

Table 11 Attacks that are implemented in 1998 DARPA test-bed [37]

	Solaris	**SunOS**	**Linux**
Denial Of Service (DoS)	Apache2 Back Mailbomb Neptune Ping Of Death Process Table Smurf Syslogd UDP Storm	Apache2 Back Land Mailbomb Neptune Ping of death Process Table Smurf UDP Storm	Apache2 back Mailbomb Neptune Ping of death Process Table Smurf Teardrop UDP Storm
Remote to Local (R2L)	dictionary ftp-write guest phf xlock xsnoop	dictionary ftp-write guest phf xlock xsnoop	dictionary ftp-write guest imap named phf sendmail xlock xsnoop
User to Root (U2R)	eject ffbconfig fdformat ps	loadmodule ps	perl xterm
Surveillance/ Probing	ip sweep mscan nmap saint satan	ip sweep mscan nmap saint satan	ip sweep mscan nmap saint satan

5.1.1. Denial of Service (DoS)

The DoS attack aims to disable a service or a process by making the system too busy. The attacker takes advantage of software bugs or misconfigurations and causes the system to be unable to respond legitimate requests.

Table 12 DoS attacks [60]

Attack Type	**Mechanism**	**Effect of attack**
Apache2	Abuse	Crashes httpd
Back	Abuse/Bug	Slows down server response
Land	Bug	Freezes the machine
Mailbomb	Abuse	Annoyance
SYN Flood	Abuse	Denies service on one or more ports
Ping of Death	Bug	None
Process Table	Abuse	Denies new processes
Smurf	Abuse	Slows down the network
Syslogd	Bug	Kills the Syslogd
Teardrop	Bug	Reboots the machine
Udpstorm	Abuse	Slows down the network

5.1.1.1. Apache2

The Apache2 attack is a DoS attack in which the attacker sends requests with too many HTTP headers too the Apache web server. After receiving to many of these requests the web server runs out of memory and eventually crashes [112].

5.1.1.2. Back

The back attack is another DoS attack that causes the Apache server to become unavailable. The attacker submits URL requests that contain many frontslashes. The frontslashes causes excessive consumption of the server's CPU and eventually the server becomes unable to respond any other system activities [112].

5.1.1.3. Land

The land attack takes advantage of some older TCP/IP implementations. It sends a spoofed SYN packet in which the source and destination addresses are the same. When the PC receives this packet it enters into an infinite loop and is not able to respond to any other system activities. The PC has to be restarted in order to recover [112].

5.1.1.4. Mailbomb

In a mailbomb attack the server receives many messages that causes the server's system to overflow [112].

5.1.1.5. SYN Flood (Neptune)

Each TCP/IP implementation is vulnerable to SYN flood therefore this attack is still today's one of existing attacks. Each half-open TCP connection (SYN request) is stored in TCP/IP stack to track the incoming connections. This stack has a finite number of storages where the incoming connections (SYN requests) are stored in. The attacker takes advantage of this limitation and sends many SYN requests with spoofed non-existent source addresses. The victim's system will send a SYN/ACK packet to the non-existent addresses and wait for the ACK responses. This connection is a half-open connection and will take relatively long time to time out. The attacker will flood the victim system with many SYN packets and eventually the systems TCP/IP stack will overflow and new connections will be no longer accepted [113].

5.1.1.6. The Ping of Death (Pod)

Every ICMP (Internet Control Message Protocol) echo has a limit of 2^{16} bytes for its data part. An attacker can easily send an ICMP echo packet that exceeds its data limit just by using the command line in Windows95 or simple programs in other operating systems [112]. Several operating systems crash if they receive an oversized ICMP echo packet. But currently almost all systems have patches against this attack and are not affected. However this vulnerability can be seen in new technology such as Bluetooth protocol which has a similar ping packet on the L2CAP layer [113].

5.1.1.7. Process Table

This attack is created for the DARPA evaluation experiment. The goal of this attack is to completely fill the victim's process table with multiple processes and eventually to shut down the system [112].

5.1.1.8. Smurf

A smurf attack is an ICMP echo request that has the broadcast addresses from remote locations as the destination and the victim's address as the source (see Figure 12). The packet pretends as if it is sent by the victim's machine. Since the packet is sent to the broadcast addresses of many remote networks, the victim receives too many ICMP echo reply packets

from all addresses that receive the ICMP echo. By generating such malformed ping flood, the victim's bandwidth is used up and legitimate traffic is not able to get through [112], [113].

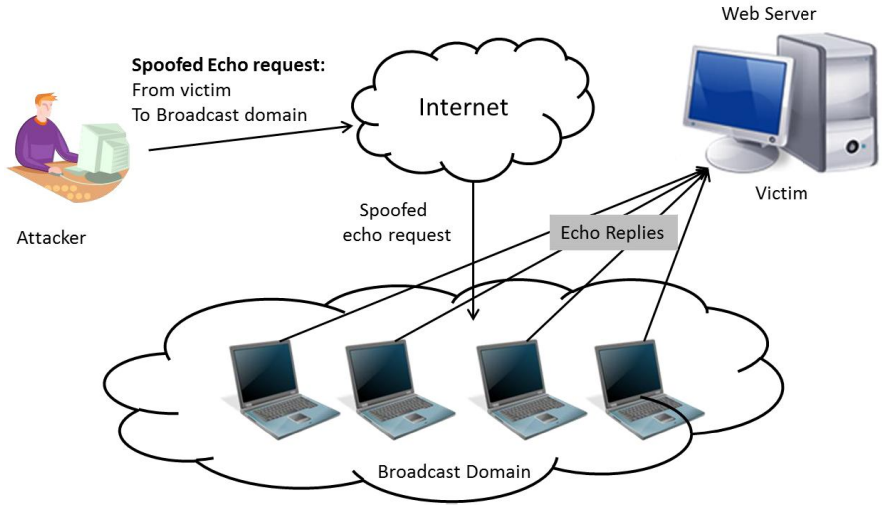

Figure 12 A demonstration of a smurf attack

5.1.1.9. Syslogd

The syslogd is an exploit that allows the attacker to kill remotely the syslogd service on the Solaris server. After receiving a remote message, the solaris server tries to attempt a DNS lookup on the source IP address which is the attacker's fake address. When there is no match in the DNS result, the syslogd will crash [112].

5.1.1.10. Teardrop

The teardrop attack takes advantage of older TCP/IP implementations. In reassembly of fragmented IP packets usually the fragmented packets lineup correctly because of correct offset numbers. But teardrop sends packets with wrong offset numbers that cause the fragmented packets to overlap and crash the system. This exploit is appeared in the early implementations of IPv6 as well [113].

5.1.1.11. Udpstorm

The Udpstorm attack creates a connection between two UDP services by simply sending a spoofed packet to a victim's machine that has a UDP service and pretends as if it is sent by another victim's machine that also has a UDP service. Those two UDP services start to blindly reply echo request to each other and this continues as an infinite loop that causes the system to be unresponsive to any other legitimate request. This infinite loop can only be stopped by restarting the daemon [112].

5.1.2. Remote to Local (R2L)

R2L attacks tries to gain local access on a machine without having permission by taking advantage of misconfigurations and/or software bugs [37]. The R2L attacks that have been included in the DARPA test bed are described in this section.

5.1.2.1. Dictionary

In the dictionary attack the attacker applies multiple username and password guesses to find out a valid username and password. In this attack the attacker usually knows a username and tries to guess multiple passwords to gain access. Dictionary attack can be applied using different services which provide authentication such as telnet, ftp, pop, rlogin, and imap. Frequent failed login attempts may be observed during this attack [112].

Table 13 R2L attacks [60]

Attack Type	**Mechanism**	**Effect of attack**
Dictionary	Abuse feature	Gains user access
Ftp-write	Misconfig.	Gains user access
Guest	Misconfig.	Gains user access
Imap	Bug	Gains user access
Named	Bug	Gains user access
Phf	Bug	Executes commands as HTTP user
Sendmail	Bug	Executes commands as root
Xlock	Misconfig.	Spoof user to obtain password
Xnsoop	Misconfig.	Monitor key stokes remotely

5.1.2.2. Ftp-write

This attack takes advantage of the anonymous FTP misconfiguration. One of these misconfigurations is missing write protection on the directories. The intruder may be able to add files such as viruses to gain local access to the system [112].

5.1.2.3. Guest

Another R2L attack is the guest attack. The attacker takes advantage of the default guest accounts which are often left without password or have simple passwords. A guest attack is similar with the dictionary attack. The only difference is in the dictionary attack the attacker tries hundreds/thousands username/password combinations while in guest attack the attacker only tries couple of username/password combinations such as guest/<none> or guest/guest [112].

5.1.2.4. Imap

Imap is a R2L attack that takes advantage of vulnerable versions of servers or inadequate configuration in the inputs of IMAP servers. Inadequate configurations or vulnerable versions of servers cause bugs such as buffer overflow. The attacker sends a wisely created text to the IMAP server and gains root access [112]. In DARPA evaluation, the imap attack has been applied on the IMAP server of Redhat Linux 4.2.

5.1.2.5. Named

Named is another R2L attack that makes use of the buffer overflow bug in the early releases of BIND. This attack either crashes the system or obtains root privileges [112].

5.1.2.6. Phf

Common Gateway Interface (CGI) protocol is mostly used in web servers to produce dynamic web pages. On the other hand, the cgi-bin is a folder where the CGI scripts such as "phf" are located. A badly written phf script may be vulnerable to attacks. For instance, the behavior of the script changes if "0a" is appended to the URL. The following example is a phf attack that reads the password file [114]:

http://victim.com/cgi-bin/phf?Qalias=x%0a/bin/cat%20/etc/passwd

5.1.2.7. Sendmail

Sendmail is an e-mail transfer software that routes messages between networks. In a sendmail attack the attacker uses the bug in the sendmail software. In 1998 DARPA intrusion detection evaluation, the buffer overflow bug found in sendmail version 8.8.3 is used for the sendmail attack. With this attack the remote intruder is able to execute root commands on the server [112].

5.1.2.8. Xlock

In the xlock attack, the intruder uses a modified xlock program. The original xlock program is a screensaver that locks the computer. The modified (trojan) xlock program acts as if it is the screensaver but sends the users password and username to the attacker. This trojan program is displayed on the victims machine that has left the x display open [112].

5.1.2.9. Xsnoop

In the xsnoop attack the intruder monitors the keystrokes of the user who has left the x display unprotected. With this attack the intruder can obtain confidential information such as victim's username and password and can gain access into the system.

5.1.3. User to Root (U2R)

In an U2R attack the intruder has already local access and obtains privileges that is reserved for UNIX root or super-user by taking advantage of software bugs or misconfigurations [37]. Each U2R attack and their effects are summarized in Table 14. The U2R attacks applied in the DARPA intrusion detection evaluation are described in this section.

Table 14 U2R attacks [60]

Attack Type	**Mechanism**	**Effect of attack**
Eject	Buffer overflow	Gains root shell
Ffbconfig	Buffer overflow	Gains root shell
Fdformat	Buffer overflow	Gains root shell
Loadmodule	Poor environment sanitation	Gains root shell
Perl	Poor environment sanitation	Gains root shell
Ps	Poor temp file management	Gains root shell
Xterm	Buffer overflow	Gains root shell

5.1.3.1. Eject

This attack exploits a buffer overflow in the eject program that comes with Solaris 2.5. This software bug can be used by an intruder to gain root access on the system [112].

5.1.3.2. Ffbconfig

The ffbconfig attack uses the software bug found in ffbconfig program that comes with the Solaris 2.5. This program has a buffer overflow problem that help the intruder to overwrite the internal stack space of the ffbconfig program [112].

5.1.3.3. Fdformat

Fdformat attack takes advantage of a buffer overflow in the fdformat program that is distributed with Solaris 2.5 [112].

5.1.3.4. Loadmodule

In DARPA intrusion detection evaluation, the loadmodule is applied against SunOS 4.1 systems which use the xnews window system server to load two dynamically loadable kernel drivers into the system and to create special devices in the /dev directory to use those modules. The bug in the loadmodule program allows an intruder to gain root access on the local system [112], [115], [116].

5.1.3.5. Perl

In a perl attack anyone who has local access can gain root access on systems that has suidperl or sperl installed. Suidperl is a version of perl that supports saved set-user-ID and saved set-group-ID. This program has an implementation bug that does not correctly drop its root privileges when changing the accounts user and group ID [116].

5.1.3.6. Ps

In a ps attack the intruder with local access on the system gains root access through accessing temporary files. This attack can occur if the permissions for the temporary files /temp and /car/temp are set incorrectly in the version of ps distributed with Solaris 5.2 [116].

5.1.3.7. Xterm

In an xterm attack the intruder with local access on the system gains root access through a buffer overflow in the Xaw library that comes with different operating systems. In DARPA evaluation the xterm attack is applied on the Redhat Linux 5.0 operating system. The bug is a buffer overflow caused by a user supplied data in both the xterm program and any program that uses the Xaw library [116].

5.1.4. Probing (Probe)

The probe attack is a scanning mechanism that scans the network to gather information and vulnerabilities [37], [60]. Probing attacks used in DARPA evaluation and their effects are summarized in Table 15. Probing attacks applied in the DARPA intrusion detection evaluation are described in this section.

5.1.4.1. Ipsweep

The ipsweep is looking for hosts which are listening to the network. It commonly sends an ICMP ping packet to every possible host in networks. The hosts that send replies are recorded

as active hosts that are listening the network. The attacker uses this information to perform attack and/or to search for vulnerabilities [116].

Table 15 Probe attacks [60]

Attack Type	Mechanism	Effect of attack
Ipsweep	Abuse of feature	Identifies active machines
Mscan	Abuse of feature	Looks for known vulnerabilities
Nmap	Abuse of feature	Identifies active ports on a machine
Saint	Abuse of feature	Looks for known vulnerabilities
Satan	Abuse of feature	Looks for known vulnerabilities

5.1.4.2. Mscan

Mscan is a scanning tool that locates machines and looks for their vulnerabilities through DNS zone transfers and/or scanning the IP addresses. By using mscan an entire domain can be scanned for specific vulnerabilities [112].

5.1.4.3. Nmap

Nmap is a free port scanner available for both Linux and Windows. It scans the network for a variety of options such as SYN, FIN and ACK with TCP, UDP and ICMP scanning. The nmap tool provides different scanning options such as specifying the ports that are being scanned, setting time intervals between each scan, and setting sequentially or random scanning [112].

5.1.4.4. Saint

Saint (Security Administrator's Integrated Network Tool) is a security tool that is normally not intended to be an attack tool. It helps the network administrator to check the network. However it is a useful tool for attackers to gather variety of information from the network. Saint examines the network services, such as finger, NFS, FTP, and TFTP, to gather information from remote hosts. Some examples of the collected information are: presence of various network information and security flaws such as incorrectly setup network services, well known bugs in system or network utilities, and poor policy decisions [112].

5.1.4.5. Satan

Satan is an early predecessor of saint described in the previous section. Satan has a similar design and similar purpose as saint. The minor difference is that Satan scans for different vulnerabilities.

5.2. Description of Attacks that Appear in the ISCX dataset

5.2.1. Infiltrating the Network from the Inside (Infiltrating)

This attack scenario first collects information about the target such as network IP ranges, mail servers, and user email accounts. This information is usually obtained through DNS queries, such as nslookup and dig. However the results show that only NAT server is accessible from outside network. Therefore client side attack techniques are more useful. In order to conduct a client side attack, potential email addresses that are required to penetrate into the system are guessed through the mail server [30].

In this attack scenario the Adobe Reader util.printf() buffer overflow vulnerability was used as a starting point. A malicious PDF sent on behalf of admin@[...] to all 21 users of the testbed as a system upgrade email. By clicking on the PDF file, Adobe opens but shows only a gray window. At the background, it makes a reverse TCP connection back to the attacking computer which is listening on port 5555. Once full access on a local machine is gained, potential targets such as internal servers and databases are investigated and their vulnerabilities are exploited. In this attack a backdoor is created which can provide flow of desired information from the inside network to the attacker within a pre-set interval [30].

5.2.2. HTTP Denial of Service (HTTPDoS)

The second attack is conducted with *Slowloris* tool which has proven to cause webservers totally inaccessible by just using one PC. This tool creates full TCP connection with the server and sends valid but incomplete HTTP requests at regular intervals. After a short time all of sockets of the webserver will become busy and the webserver will no longer be able to respond to any additional HTTP requests [30]. To conduct this attack the attacker first finds a vulnerable host by starting the TCP listener on the attacking PC. The TCP listener waits the operating system of a previously exploited user from inside network to run the backdoor scheduled task which is configured during the infiltrating attack. When the connection is established the whole subnet of the local machine is scanned to find vulnerable hosts. When a vulnerable host is found the attacker exploits its vulnerability and tunnel back a remote desktop connection between the user and attacker. The user's PC is then used to attack the main Apache Web server using *Slowloris*. The web server becomes inaccessible about after 10 minutes of attack [30].

5.2.3. Distributed Denial of Service using an IRC Botnet (DDoS)

In this attack scenario, a distributed attack is performed using infected hosts on a testbed. A malicious update email that contains Internet Relay Chat (IRC) bot in the attachment is sent to users. The bot downloads a Denial of Service (DoS) program, written by the research team, from a remote server and executes it with user privileges. The DoS program performs an HTTP GET attack on a pre-defined target. Heavy multithreading is performed by each bot and applied at the same time. This attack is applied for about 60 minutes on the main Apache server. This attack caused the server to be inaccessible or to slow down [30].

5.2.4. Brute Force SSH (BruteForce)

Brute force attack breaks into accounts which have weak username and passwords. In this scenario, an SSH account is acquired with a dictionary brute force attack against the main server. The dictionary composes of over 5000 alphanumerical passwords. The attack is executed for 30 minutes and the superuser account credentials are obtained. The credentials were used to login into the server and download sensitive information such as */etc/passwd* and */etc/shadow* files [30].

6. EXPERIMENTS ON INTRUSION DETECTION

The implementations of all subjects that have been covered in the previous chapters are explained in this chapter. The experiments are conducted in three phases. In the first phase the RBF Neural Network is applied on the KDD dataset which are divided into three parts according to the protocol type. The second phase is focusing on adapting IDSs on new datasets. In this phase, transfer learning is applied on IDSs which has never been implemented before according to a comprehensive research. In the third phase, various popular machine learning algorithms described in Chapter 3 and a feature selection method described in Chapter 4 are applied on intrusion detection datasets. This phase focused on obtaining higher DRs and lower FARs using hybrid detection methods.

To evaluate the experiments the Detection Rates (DRs), accuracy, and False Alarm Rate (FAR) which are commonly used in IDS related papers [45], [46], [51], [117] are calculated. Table 16 describes the True Positive, True Negative, False Positive, and False Negative values which are used in the following equations. Equations (31, 32, and 34) describe the DR, FAR, and accuracy, respectively.

Table 16 Confusion Matrix

		Predicted Class	
		Positive Class	**Negative Class**
Actual Class	**Positive Class**	True Positive (TP)	False Negative (FN)
	Negative Class	False Positive (FP)	True Negative (TN)

$$\text{Detection Rate (DR)} = \begin{cases} \frac{TP}{TP+FN}, & \text{for positive classes (attacks)} \\ \frac{TN}{TP+FP}, & \text{for negative classes (normal)} \end{cases} \quad (31)$$

$$\text{False Alarm Rate (FAR)} = \frac{FP}{FP+TN} \quad (32)$$

$$\text{Overall Accuracy} = \frac{TP+TN}{TP+TN+FN+FP} \quad (33)$$

6.1. Phase I: Protocol Type Based Intrusion Detection Using RBF Neural Network

6.1.1. Dataset

In this phase experiments with RBF Neural Network are conducted on the KDD Cup'99 dataset. MATLAB is used for the experiments on a PC with 4 GB of memory and 2.27 GHz of CPU. Since the memory and speed of the PC is limited, the size of the train set is decreased by deleting repeated data. After deleting repeated data, 614,450 of train and 77,290 of test data are left. On the other hand, the dataset is divided into three parts, according to the protocol names: TCP, UDP and ICMP. Table 17 shows the number of remaining test and train data grouped by the protocols. It can be seen that the size of data of ICMP protocol groups is very less. This is because of its nature. ICMP protocol differs from UDP and TCP protocols. It is not regularly employed by end-user network applications, instead it is usually used for diagnostic or control purposes such as sending error messages. Therefore the size of ICMP protocols is usually expected to be lower than the sizes of UDP and TCP protocols.

Table 17 Dataset description after deleting repeated data and grouping by protocol types

Protocol Name	TCP		UDP		ICMP	
	Train	Test	Train	Test	Train	Test
Normal	529,517	43,908	28,435	3,770	1,324	233
Attack	50,499	27,214	866	922	3,809	1,242
Total	580,016	71,122	29,301	4,692	5,133	1,475

The data sizes are still not small enough, thus some randomly selected data is deleted. Since the dataset includes too many "normal" data, most of the deleted data is chosen from "normal" labeled data.

Some of the attacks only appear in test set. Since RBF is a supervised learning technique, the training should be done with all attacks. For this reason, some of these attacks are copied to the train set. But the test set is kept the same (see Table 18).

Table 18 Data description after deleting some more data randomly and copying some attacks from test to train dataset

Protocol Name	TCP		UDP		ICMP	
	Train	Test	Train	Test	Train	Test
Normal	2,698	43,908	5,134	3,770	1,325	233
Attack	3,302	27,214	942	922	3,838	1,242
Total	6,000	71,122	6,076	4,692	5,163	1,475

In pre-processing, the string type attributes of the KDD Cup'99 dataset are converted to numeric values in three different ways called as Type-A, Type-B and Type-C, with respect to their frequency in the test set. In Type-B, the highest number is given to the attribute with most frequency and 1 with less frequency. It is done in the opposite way for Type-A, and random numerical values were given in Type-C (see Table 19, Table 20, Table 21, and Table 22).

Table 19 Converting flag names to numerical values (for TCP data)

Flag	Frequency	Type-A	Type-B	Type-C
SF	6,765	1	11	10
S0	3,986	2	10	6
REJ	1,488	3	9	2
RSTR	633	4	8	5
RSTO	307	5	7	3
S3	272	6	6	9
SH	182	7	5	11
S1	58	8	4	7
S2	29	9	3	8
RSTOS0	25	10	2	4
OTH	4	11	1	1

The flag attribute in UDP and ICMP parts contains only one instance. Therefore this attribute (the entire column) is deleted for UDP and ICMP test and train sets. Some other attributes with only one value are also observed. These attributes (columns) are deleted since they don't affect the learning. The final number of input and output of each dataset can be found in Table 23. Each dataset is then normalized to the [0-1] range.

Table 20 Converting service names to numerical values (for TCP data)

Service Name	Frequency	Type-A	Type-B	Type-C
private	3,156	1	57	40
http	3,012	2	56	17
telnet	1,669	3	55	50
ftp	910	4	54	13
other	864	5	53	35
ftp_data	821	6	52	14
smtp	765	7	51	44
finger	507	8	50	12
pop_3	401	9	49	38
imap4	227	10	48	19
auth	177	11	47	1
sunrpc	113	12	46	47
IRC	110	13	45	20
time	88	14	44	51
domain	52	15	43	8
remote_job	40	16	42	41
sql_net	39	17	40	45
ssh	39	18	41	46
X11	32	19	39	56
discard	29	20	36	7
echo	29	21	37	9
systat	29	22	38	49
gopher	28	23	34	15
link	28	24	35	25
iso_tsap	26	25	28	21
mtp	26	26	29	27
netbios_ns	26	27	30	30
netstat	26	28	31	32
pop_2	26	29	32	37
rje	26	30	33	42
daytime	25	31	24	6
netbios_dgm	25	32	25	29
supdup	25	33	26	48
uucp_path	25	34	27	53
bgp	24	35	20	2
ctf	24	36	21	5
netbios_ssn	24	37	22	31
whois	24	38	23	55
csnet_ns	23	39	17	4
name	23	40	18	28

vmnet	23	41	19	54
hostnames	22	42	15	16
Z39_50	22	43	16	57
nntp	18	44	13	34
pm_dump	18	45	14	36
ldap	15	46	12	24
uucp_path	10	47	11	52
login	9	48	10	26
nnsp	7	49	7	33
printer	7	50	8	39
shell	7	51	9	43
kshell	6	52	6	23
courier	5	53	3	3
exec	5	54	4	11
http_443	5	55	5	18
efs	4	56	2	10
klogin	3	57	1	22

Table 21 Converting service names to numerical values (for UDP data)

Service Name	Frequency	Type-A	Type-B	Type-C
domain_u	3,679	1	5	1
ntp_u	1,373	2	4	3
private	795	3	3	5
other	152	4	2	2
tftp_u	-	5	1	4

Table 22 Converting service names to numerical values (for ICMP data)

Service Name	Frequency	Type-A	Type-B	Type-C
eco_i	2,990	5	1	1
ecr_i	1,727	4	2	5
urp_i	270	3	3	2
urh_i	146	2	4	4
tim_i	-	1	5	3

Table 23 Output counts and input counts after pre-processing

Protocol Name	TCP	UDP	ICMP
Input counts	31	20	18
Output counts	1	1	1

6.1.2. Experiments and Results

The experiments are conducted on Matlab and the maximum number of neurons is set to 1000. Number of neurons to add between displays is set to 25 and spread of RBF is set to one. The training and testing is repeated for all three data types in which string attributes are converted to numericals in three different ways and named as Type-A, Type-B, and Type-C.

Table 24 MSE values obtained from training the TCP dataset

Number of Neurons	Type-A	Type-B	Type-C
50	0.02702	0.02718	0.02985
100	0.01540	0.01575	0.01648
150	0.01127	0.01097	0.01275
200	0.00900	0.00869	0.00927
250	0.00772	0.00722	0.00680
500	0.00321	0.00335	0.00295
750	0.00165	0.00157	0.00151
1,000	0.00101	0.00097	0.00089

Table 25 MSE values obtained from training the UDP dataset

Number of Neurons	Type-A	Type-B	Type-C
50	0.00410	0.00410	0.00259
100	0.00175	0.00175	0.00138
150	0.00108	0.00108	0.00085
200	0.00079	0.00079	0.00065
250	0.00062	0.00062	0.00057
500	0.00033	0.00034	0.00032
750	0.00023	0.00022	0.00022
1,000	0.00018	0.00020	0.00021

In these experiments the mean squared error (MSE), represents the performance, is used to compare them. It can be seen from Table 24, Table 25 and Table 26 that the best training results for all three types of converted data are different. For instance, the best result for TCP dataset is Type-C conversion while the UDP dataset's best result is obtained with Type-A and ICMP dataset's best result is obtained with Type-B conversion.

Table 26 MSE values obtained from training the ICMP dataset

Number of Neurons	Type-A	Type-B	Type-C
50	0.00617	0.00062	0.00625
100	0.00382	0.00382	0.00264
150	0.00183	0.00184	0.00211
200	0.00183	0.00182	0.00210
250	0.00183	0.00182	0.00208
500	0.00099	0.00053	0.00080
750	0.00087	0.00036	0.00052
1,000	0.00073	0.00030	0.00043

The training performances are plotted in Figure 13, Figure 14, and Figure 15 to see the results of each type of conversion against the other types of conversion. The plotted results show that the learning performances for each type are similar to each other.

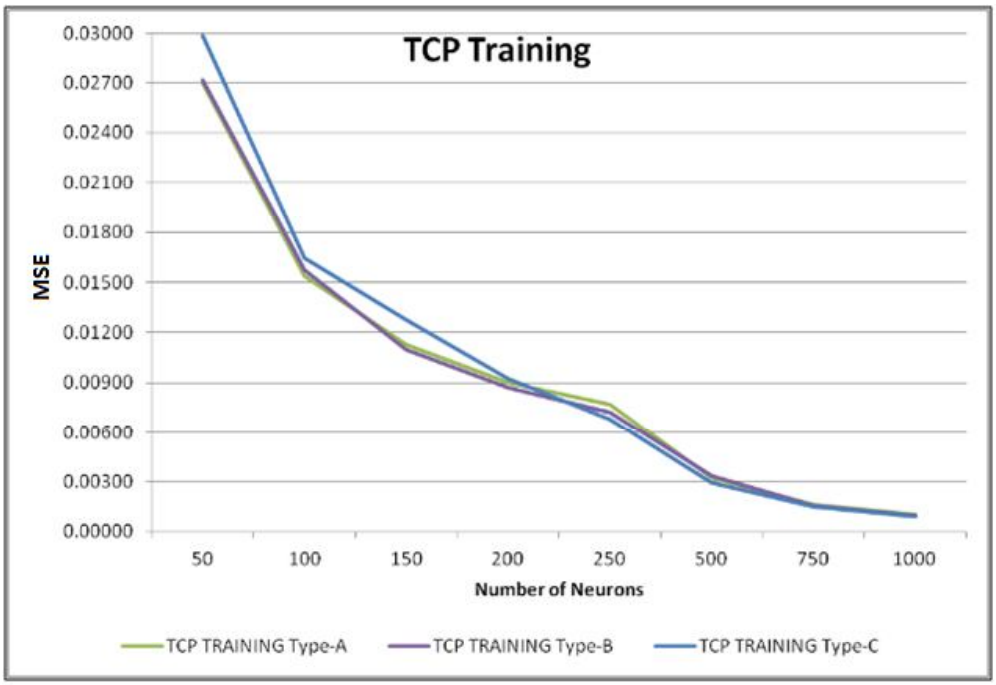

Figure 13 Training results of each conversion type for TCP dataset

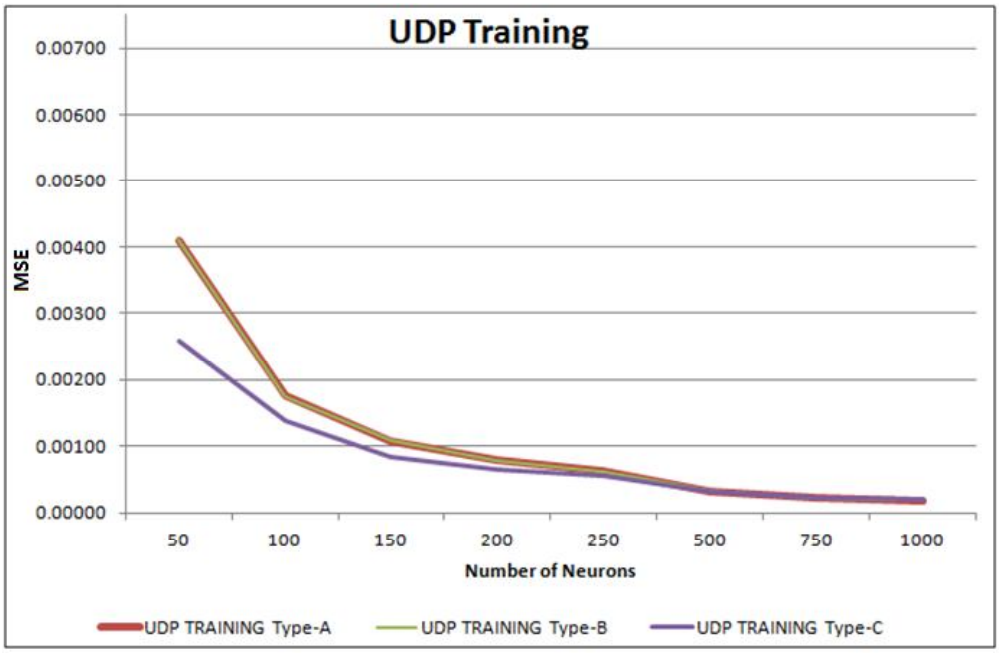

Figure 14 Training results of each conversion type for UDP dataset

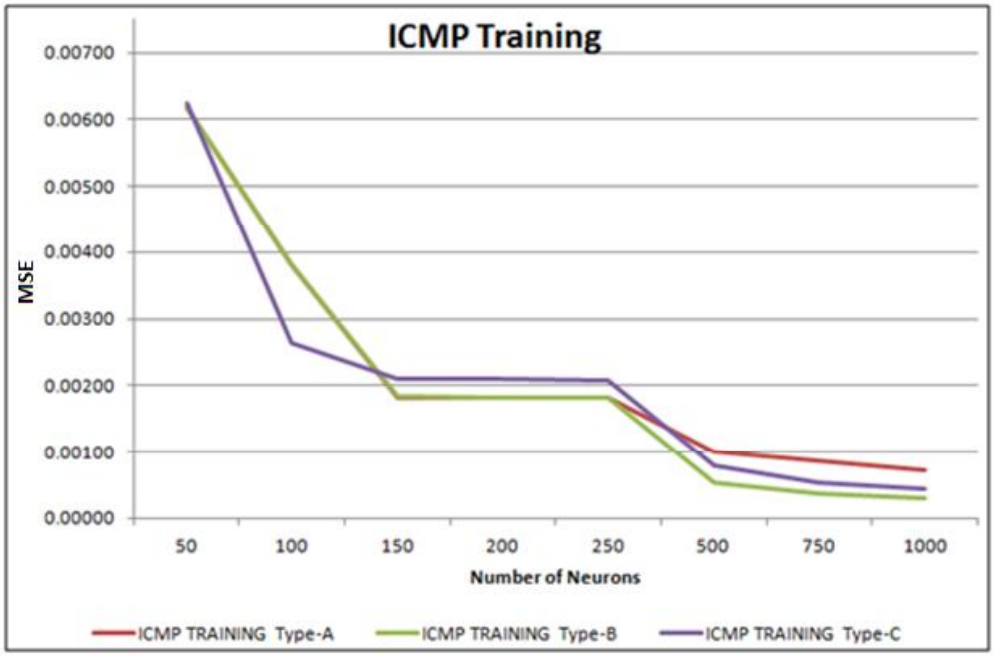

Figure 15 Training results of each conversion type for ICMP dataset

The best accuracies and FAR values of test set are obtained as 95.65% and 2.6%, 63.96% and 7.85%, 79.39% and 4.72% for TCP, UDP, and ICMP, respectively. For better visualization, the same results are shown in graphic format in Figure 16 and Figure 17.

Table 27 Testing results in terms of accuracy and FAR

		Type-A	Type-B	Type-C
TCP	Accuracy	90.86%	94.28%	**95.65%**
	FAR	3.45%	3.38%	**2.60%**
UDP	Accuracy	61.42%	**65.09%**	63.96%
	FAR	8.78%	10.29%	**7.85%**
ICMP	Accuracy	**88.95%**	83.46%	79.39%
	FAR	16.31%	15.88%	**4.72%**

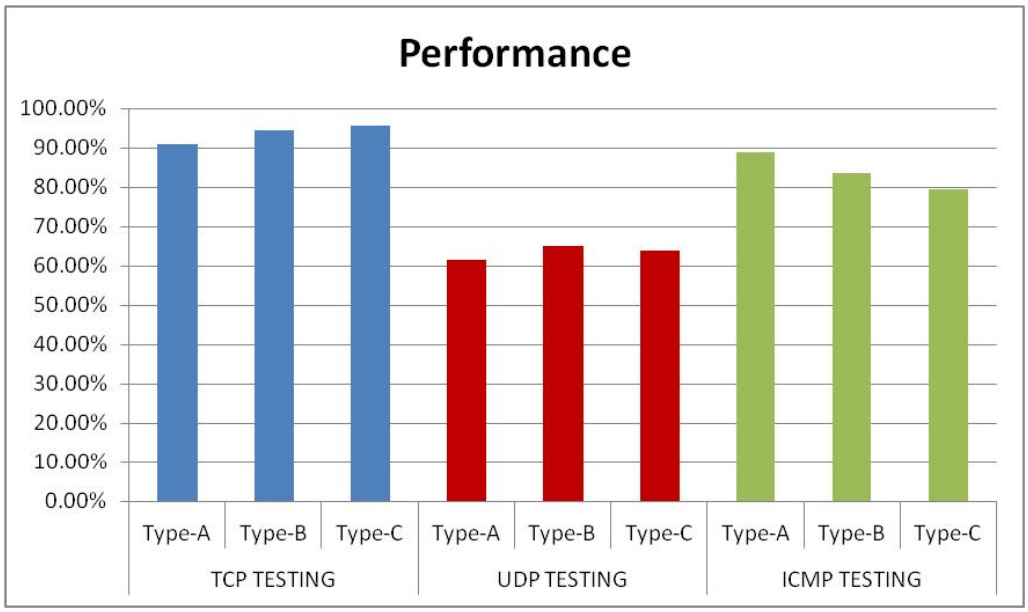

Figure 16 Accuracies obtained with the test dataset

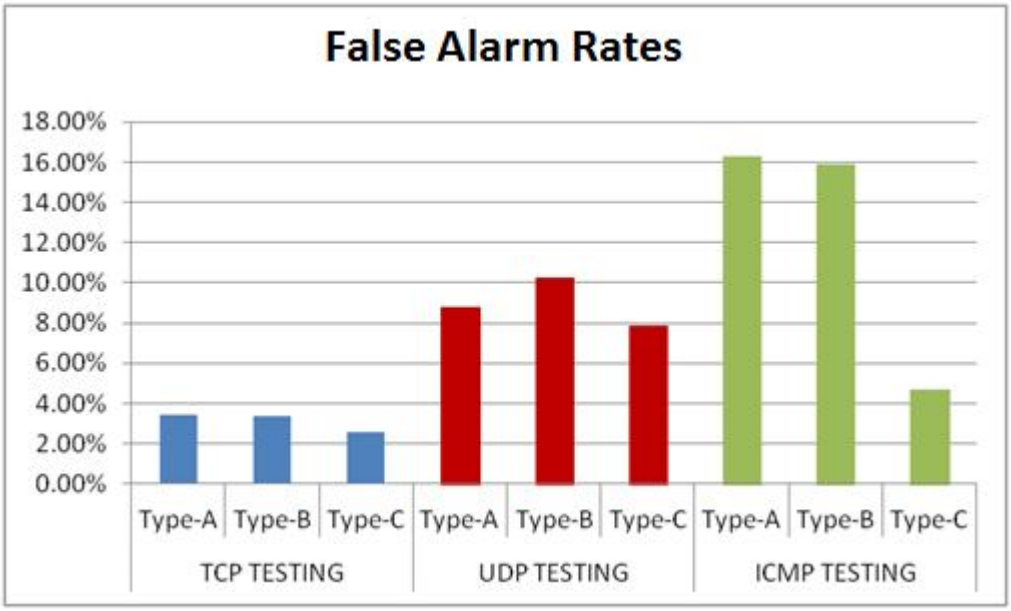

Figure 17 FARs obtained with the test dataset

The FAR values of all types of conversions are observed to be similar for both TCP and UDP test datasets. The FAR values for ICMP test dataset have considerable amount of differences. It can be seen that FARs are more than 15% in Type-A and Type-B while it is less than 5% in Type-C.

These experimental results show that FAR values are higher for Type-A and Type-B datasets. This concludes that before normalization, assigning numbers to string attributes according to their frequencies lead to higher FAR values therefore may not be preferred as a conversation technique.

Training and testing the TCP dataset shows acceptable results, but still needs to be improved. On the other hand, the results for both UDP and ICMP datasets are very poor. Increasing the number of train data or more related attributes, or applying feature selection may improve the results.

The overall accuracy and FAR values are calculated as 93.42% and 2.95% respectively. In [38] the overall accuracy values are 81.66%, 92.79%, 92.59%, 92.26%, and 65.01% with Naive Bayes, Random Forest, Random Tree, Multi-Layer Perceptron, and SVM, respectively. In the same study the overall accuracy values of J48 and NB Tree are very close to the overall accuracies which are 93.82% and 93.51% respectively. In [118], the overall accuracy is 89% and FAR is 11% using RBF neural network. Since those studies are applied in different environments and the training dataset in this experiment is slightly modified, the results are not compared with other studies.

6.2. **Phase II: Applying Transfer Learning on IDSs**

6.2.1. Evaluation

In phase II, the cumulative errors, which reflect the sum of the differences of each predicted value from the desired value, as shown in Equation (34), are compared.

$$\text{Cumulative Error} = \sum_{n=1}^{N}(d - o) \tag{34}$$

Where *d*, *o,* and *N* are desired output, obtained output, and number of inputs, respectively. If the cumulative error becomes lower, the IDS will provide more precise predictions. It can also be stated that the lower the cumulative error becomes the higher DR is obtained.

6.2.2. Dataset

The KDD Cup 10% dataset is used in phase II. Many works are only interested in classifying the data just as normal and anomalous [63]. Therefore the main focus of the experiments conducted with transfer learning is; detecting whether there is any attack (anomalous) or not (normal), rather than detecting the type of attacks such as DoS, U2R, Probing, or R2L.

Table 44, which is in Section 6.3.3.3, shows the best results of each category, with different feature subsets obtained with DFBFS described in Section 4. It shows that, the best result for Normal records is obtained with the feature subset ID 413. For this reason feature subset ID 413, that gave the highest DR for normal records, has been chosen for the experiments. The attributes of this subset can be seen in Table 28. The output is a binary value which represents the record as normal or attack.

Table 28 The attributes in feature subset ID 413

Index	**Title**	**Index**	**Title**
A2	protocol_type	A33	dst_host_srv_count
A4	flag	A35	dst_host_diff_srv_rate
A5	src_bytes	A37	dst_host_srv_diff_host_rate
A6	dst_bytes	A38	dst_host_serror_rate
A10	hot	A39	dst_host_srv_serror_rate
A12	logged_in	A40	dst_host_rerror_rate
A23	count	A41	dst_host_srv_rerror_rate
A24	srv_count		

Table 29 Attack names and number of occurrences

Name	**Count**	**Name**	**Count**
back.	968	perl.	3
buffer_overflow.	30	phf.	4
ftp_write.	8	pod.	206
guess_passwd.	53	portsweep.	416
imap.	12	rootkit.	10
ipsweep.	651	satan.	906
land.	19	smurf.	641
loadmodule.	9	spy.	2
multihop.	7	teardrop.	918
neptune.	51820	warezclient.	893
nmap.	158	warezmaster.	20

Besides having benign (normal) records, the KDD Cup 10% dataset has 22 different attack types. The names and number of occurrences of these attacks are shown in Table 29.

In this experiment, the KDD Test dataset has not been used. Instead, the KDD Cup 10% dataset has been used in two forms; as outdated and updated. The outdated dataset has one of the following attacks (back, ipsweep, neptune, nmap, pod, portsweep) missing while the updated dataset has all attacks included (see Table 30). In another experiment the outdated dataset has missing all records that used the ICMP protocol. The updated dataset is created in three different ways (see Table 30). The first one has included benign (normal) packets that use the ICMP protocol, the second has included the attack packets that use ICMP protocol, and the third one has included all packets that use ICMP protocol.

Table 30 The distinction between outdated and updated datasets used in Experiments 1-6

Experiment	**The attack name that is extracted from the outdated and included the updated dataset**
Experiment 1	back attack
Experiment 2	ipsweep attack
Experiment 3	neptune attack
Experiment 4	nmap attack
Experiment 5	pod attack
Experiment 6	portsweep attack

The main idea of adding the removed data is to simulate a real-time ANN training where a newly collected dataset becomes outdated as soon as a new benign network behavior or a new attack appears in the network. It has shown that the training time decreases if the knowledge is transferred from the gained knowledge (training) with the outdated dataset.

Table 31 The distinction between outdated and updated datasets used in Experiments 7-9

Experiment	**Outdated dataset does not have**	**Updated dataset has**
Experiment 7	any packet that use ICMP protocol	normal (benign) packets that use ICMP protocol
Experiment 8		attacks that use ICMP protocol
Experiment 9		normal (benign) & attack packets that use ICMP protocol

6.2.3. Experiments & Results

The dataset, used in the experiments, has 15 attributes and one output. The output is a binary value (as either attack or normal). According to the dataset there are 15 nodes in the input layer and one node in the output layer of the ANN, whereas the number of nodes in the hidden layer is 30 (Figure 18). The ANN is a fully connected network, therefore there are 15×30=450 weights between the input and hidden layer and 30×1=30 weights between the hidden and output layer. As a result there are a total of 480 weights. In other words there are 480 genes in each individual for the genetic algorithm.

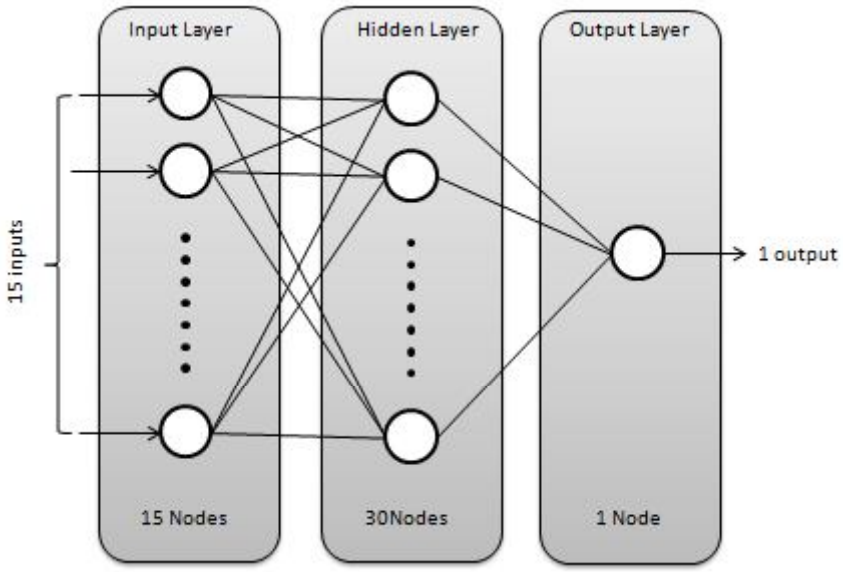

Figure 18 The structure of the artificial neural network used in the experiments

The process of applying transfer learning is done as following: The ANN is trained on the outdated dataset with 100 iterations and the weights obtained from each iteration are stored. At the end of this process the first generation is completed. This generation is passed through genetic algorithms to create new generations. Before killing the old generation, two individuals (with the best and median fitness values) from the old generation are saved into the solution pool. Then again the new generation is passed through genetic algorithms. This process is repeated 100 times. As a result the solution pool has 200 individuals each with 480 weights (genes). The crossover rate and the mutation rate parameters are selected as 0.7 and 0.01, respectively, and are chosen empirically.

The solution pool is used when there is a new (updated) dataset which is similar to the outdated dataset. Each individual is applied on the updated dataset with ANN. The individual with the fittest result is used as the initial weights at the ANN training process. The cumulative errors (equation (34)) of the Genetic & ANN hybrid transfer learning and of the Classical ANN are compared. Lower cumulative error is better because lower cumulative error provides higher detection rates. This proved that transfer learning helps the system to benefit from the previously obtained knowledge.

It can be clearly seen that in all experiments (Figure 19 - Figure 27) transfer learning method started with better cumulative error values. Additionally, even the beginning error values obtained with transfer learning method in experiments 4-7 (Figure 22 - Figure 25) were better than or very close to the error values obtained after 100 iterations with the Classical ANN. This proves that Genetic & ANN hybrid transfer learning decreases the time to train the system and provides better detection rates. In Experiment-1 (Figure 19), the Classical ANN showed slightly better result but still very similar with Genetic & ANN hybrid method. While in Experiment-3 (Figure 21) Genetic & ANN transfer learning started with a clear advantage, but after 100 iterations Classical ANN showed slightly better result than Genetic & ANN Hybrid transfer learning method. In all other Experiments (# 2, 4-9) the Genetic & ANN hybrid transfer learning method shows evidently better results than the Classical ANN. These results make it obvious that the transfer learning method helps to utilize previously obtained knowledge and improves the detection rate. It can also reduce the need to recollect the whole dataset, since only new attacks are needed for training.

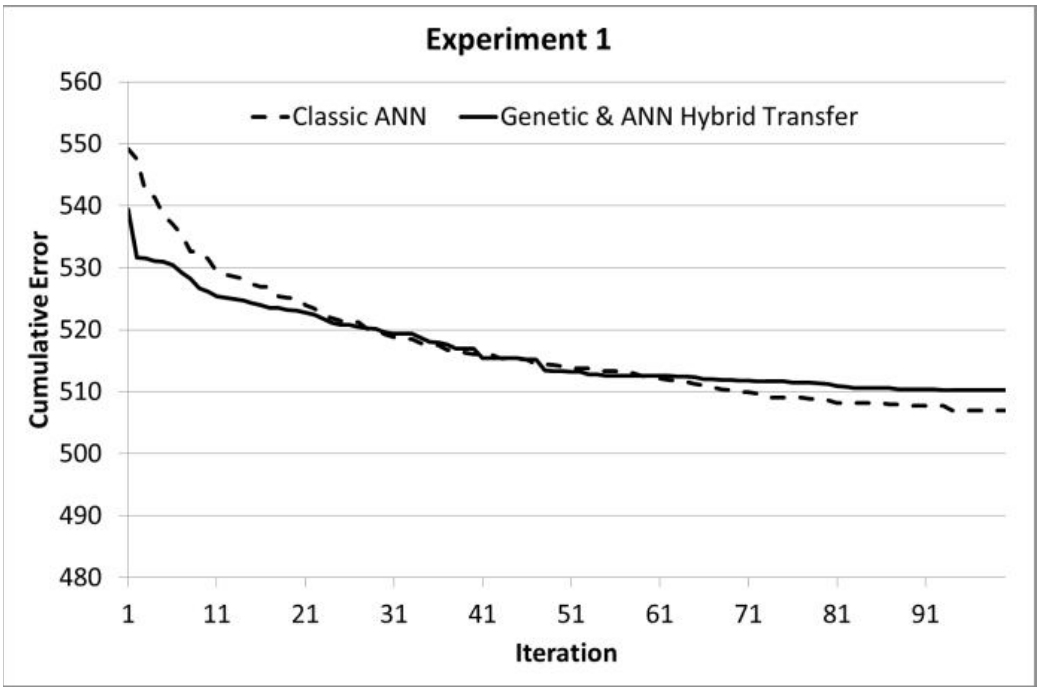

Figure 19 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any back attack while updated dataset has 968 back attacks)

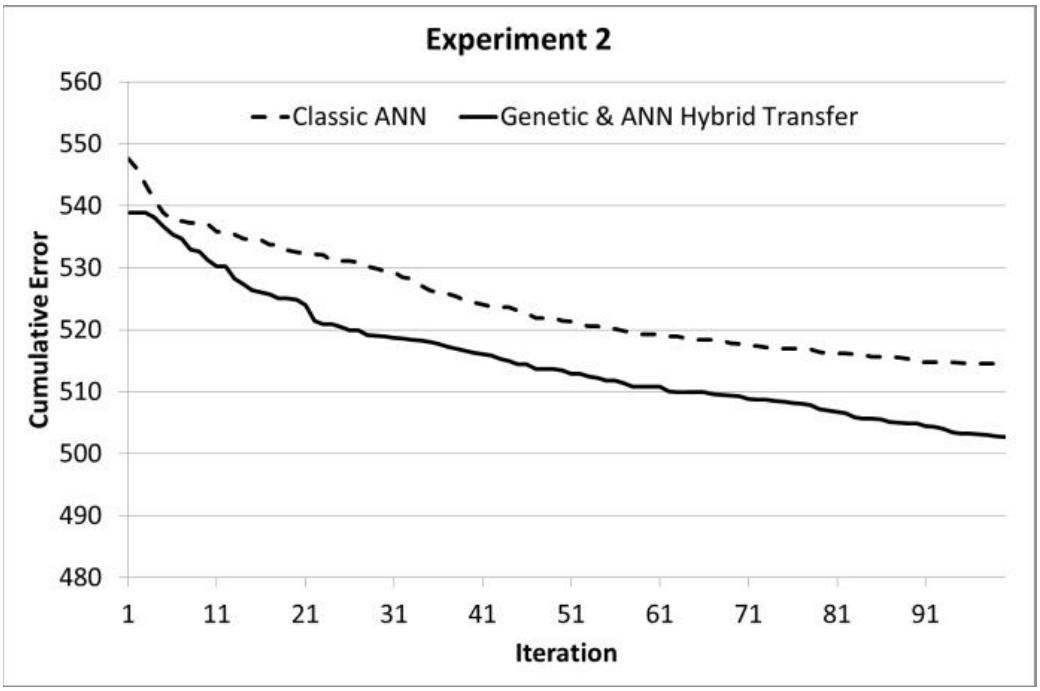

Figure 20 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any ipsweep attack while updated dataset has 651 ipsweep attacks)

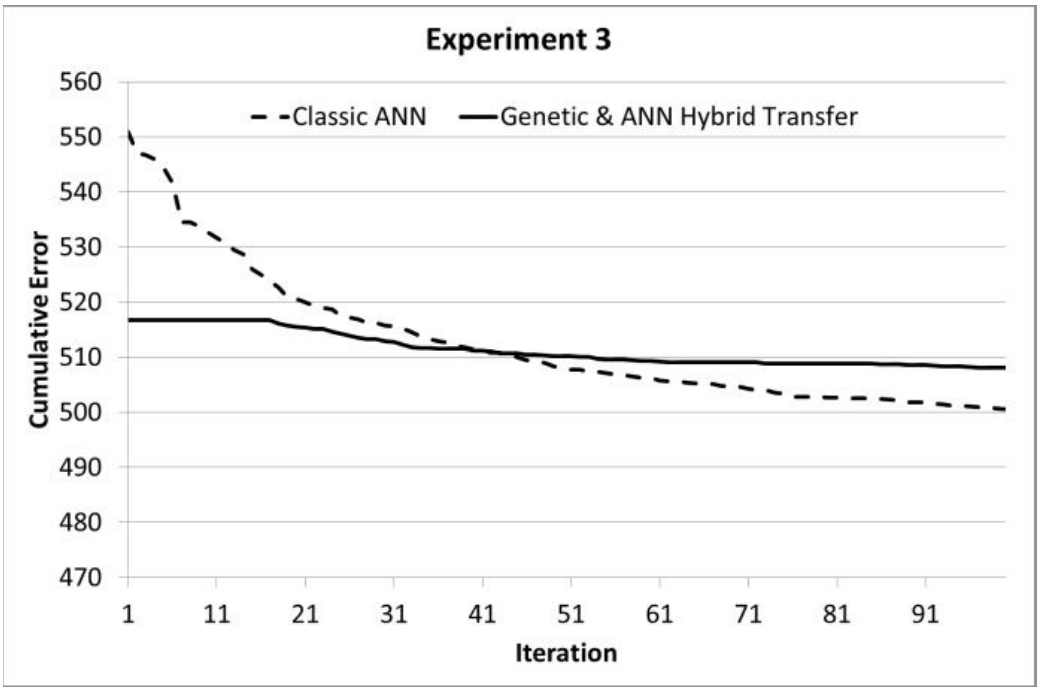

Figure 21 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any neptune attack while updated dataset has 51820 neptune attacks)

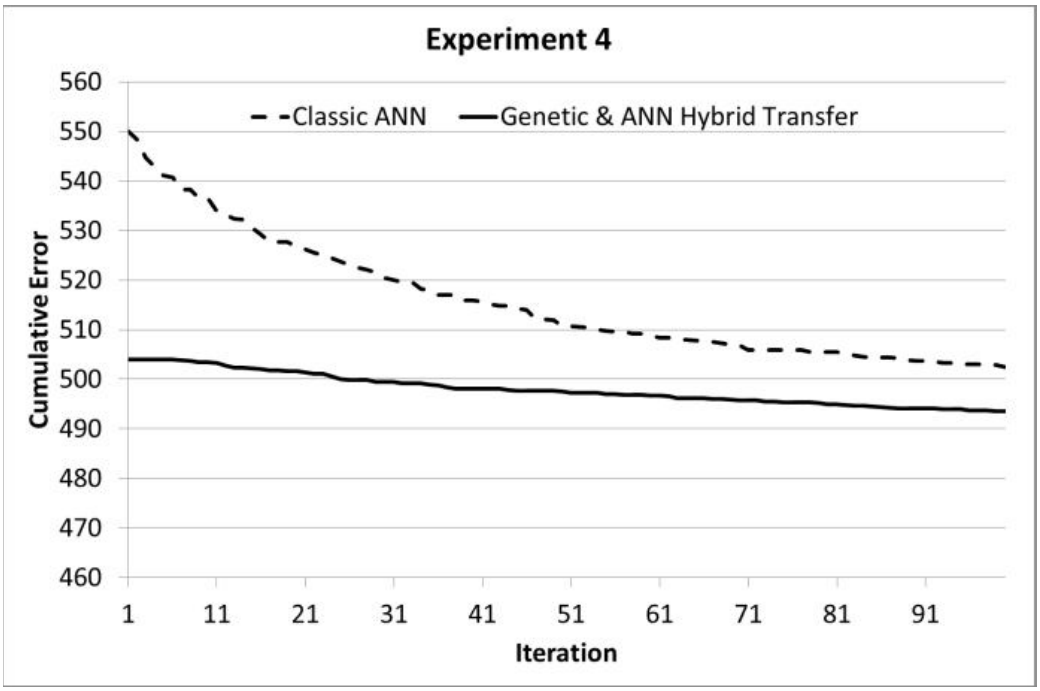

Figure 22 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any nmap attack while updated dataset has 158 nmap attacks)

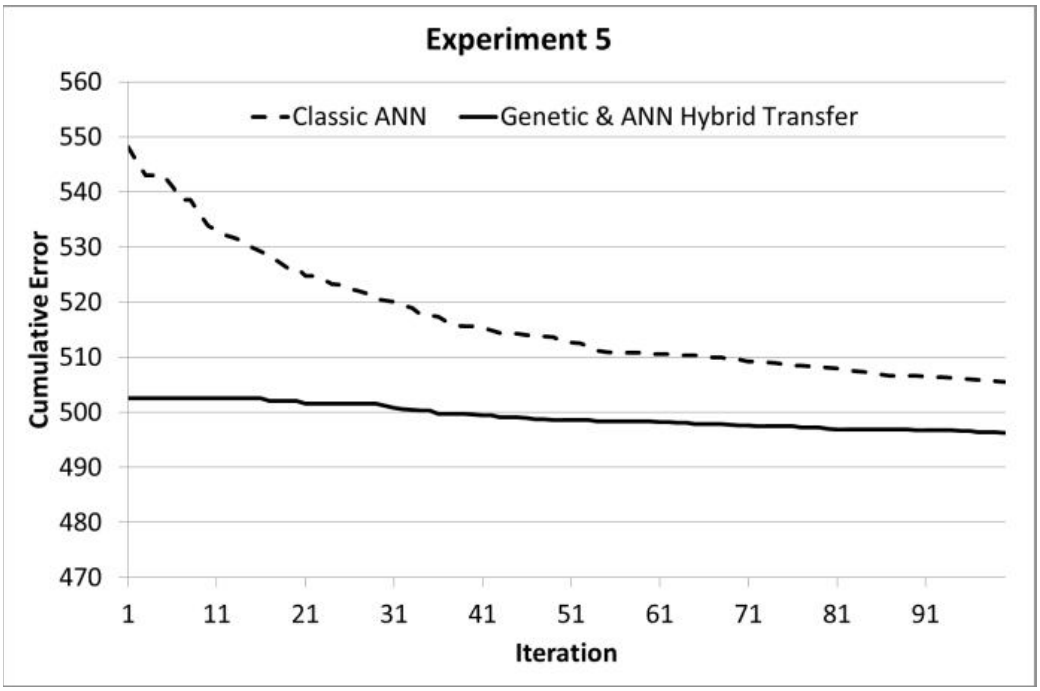

Figure 23 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any pod attack while updated dataset has 206 pod attacks)

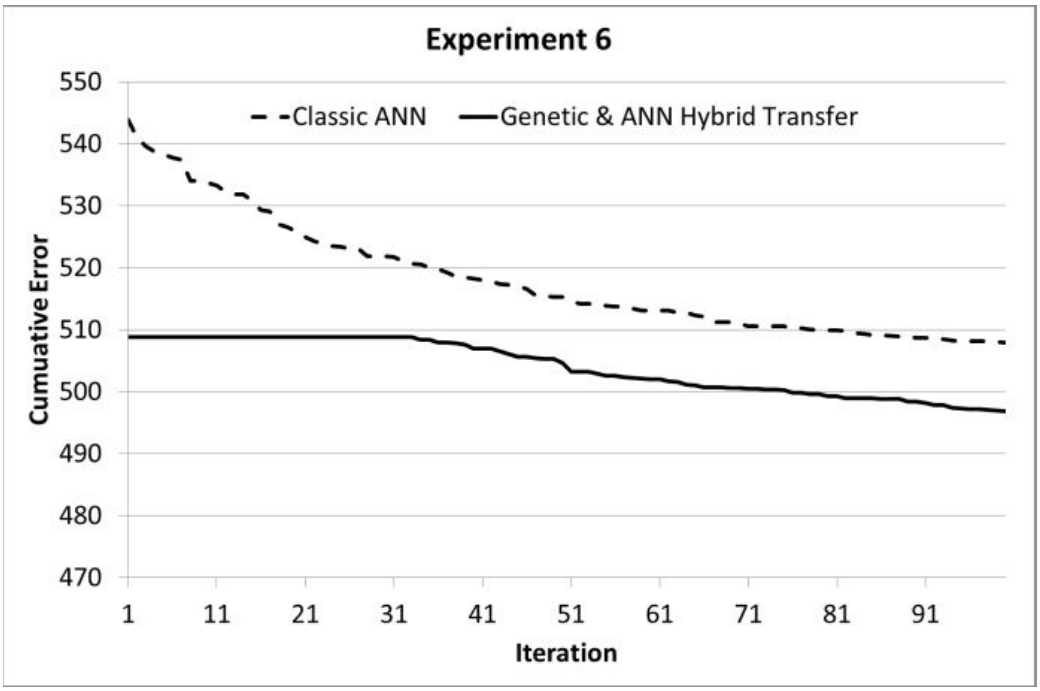

Figure 24 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any portsweep attack while updated dataset has 416 portsweep attacks)

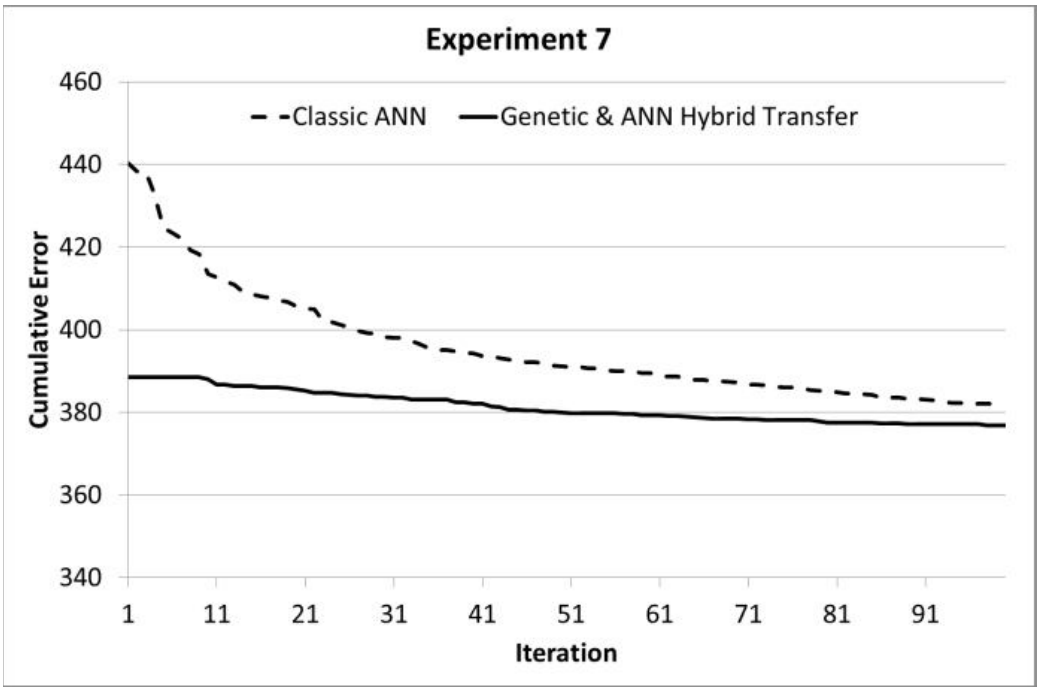

Figure 25 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any packets with ICMP protocol while updated dataset has normal packets with ICMP protocol)

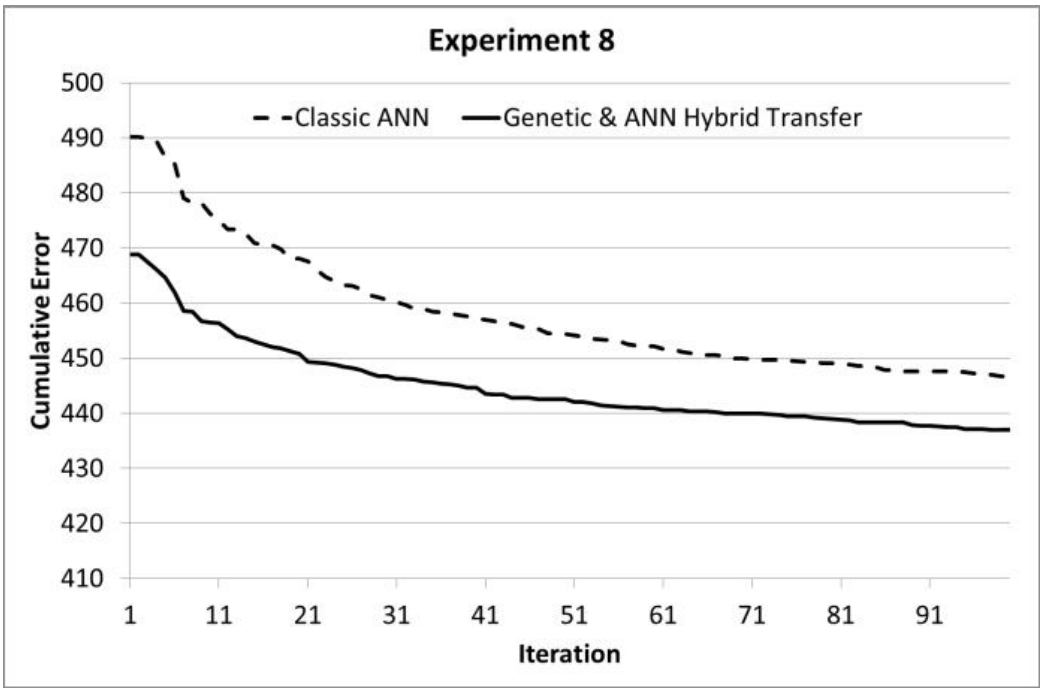

Figure 26 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any packets with ICMP protocol while updated dataset has attack packets with ICMP protocol)

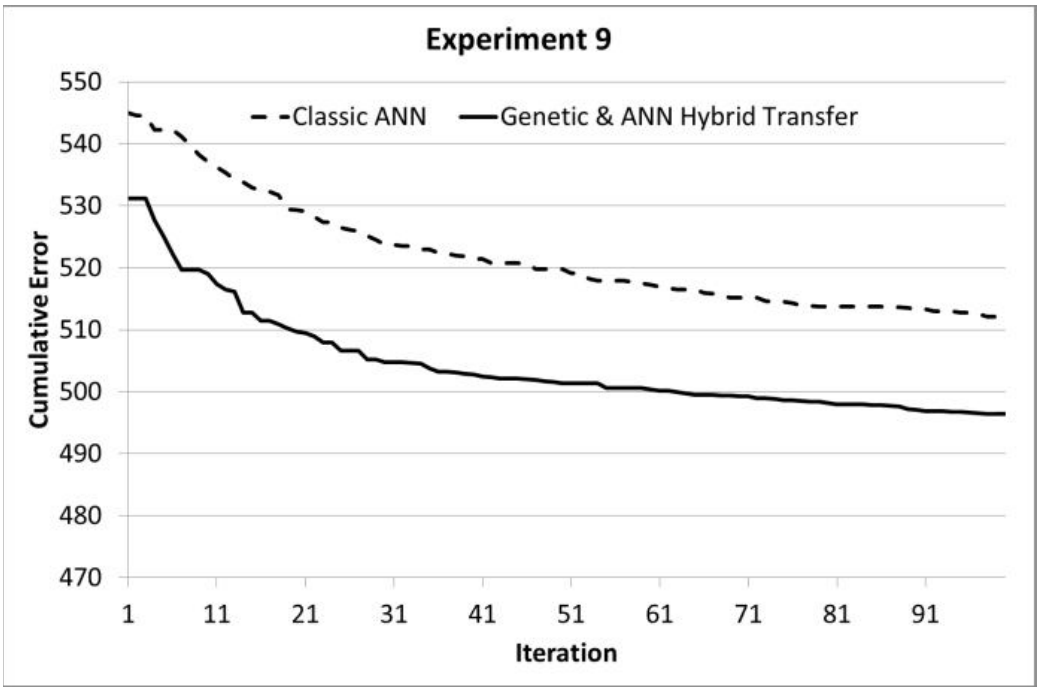

Figure 27 Difference between Classical ANN and Genetic & ANN Hybrid Transfer (outdated dataset don't have any packets with ICMP protocol while updated dataset has normal & attack packets with ICMP protocol)

6.3. **Phase III: Multilevel Hybrid Classifier for IDSs**

This phase proposes a multilevel hybrid classifier that uses different feature sets on each classifier. It uses the Discernibility Function based Feature Selection method, defined in Section 4, and two classifiers involving multilayer perceptron (MLP) and decision tree (C4.5). Experiments, defined in Section 3, are conducted on the KDD Cup and ISCX datasets. The proposal demonstrates better performance than individual classifiers and other proposed hybrid classifiers. The proposed method provides significant improvement in the detection rates of attack classes and Cost Per Example (CPE) which was the primary evaluation method in the KDD'99 Cup competition.

6.3.1. Evaluation

The cost matrix (Table 32) used in the KDD Cup competition is taken to calculate the Cost Per Example (CPE). CPE was mainly used in the KDD Cup contest for evaluation. The equation for CPE can be seen in Equation-35:

$$\text{Cost per Example (CPE)} = \frac{1}{N} \sum Mij \; x \; Cij \tag{35}$$

In this equation, N represents the total number of instances, while M_{ij} is the number of samples in class i that are classified as class j. C_{ij} is the corresponding cost in the cost matrix (Table 32). The goal is to have the lowest CPE value, whereas zero is the best possible CPU value.

Table 32 Cost matrix used in the KDD'99 Cup competition

		Predicted				
		Normal	**Probe**	**DoS**	**U2R**	**R2L**
Actual	**Normal**	0	1	2	2	2
	Probe	1	0	2	2	2
	DoS	2	1	0	2	2
	U2R	3	2	2	0	2
	R2L	4	2	2	2	0

6.3.2. Dataset

In this study, the KDD Cup'99 dataset is used for the experiments. Even though there are some limitations mentioned in [38] this dataset is still the most used dataset in IDS related research [14], [115], [119] and is considered as a classic challenge for IDS [120]. Because of the wide use of the KDD Cup'99, it brings the opportunity to compare the results with many other studies.

In order to verify the effectiveness of the proposed method it should be conducted on a real dataset as well as KDD Cup'99. Therefore publicly available datasets were explored (see Section 2.1). Since real world network communications usually contain less than 1% attack [39], only the datasets whose attack rates are close to 1% are considered. Therefore the ISCX dataset is chosen since the percentage of attacks is 2.8% which makes it close to real world datasets. Even though minor disadvantages mentioned in [40] exist, the ISCX dataset is still considered and used to evaluate the proposed method, because it is the most applicable one compared to the other explored datasets in Section 2.1.

6.3.2.1. Pre-processing KDD Cup Dataset

In the experiments, the KDD Cup 10% is used for training, while the KDD Test data is used for the testing stage. To make the results possible to compare with other related studies, no random selection process on the test set and train set are performed. The duplicated data is removed only from the train dataset to lighten the training process. No data from the KDD Test data (see Table 33) are deleted.

Table 33 Data sets with their number of records and number of attacks

Data set	Number of records (original)	Number of records (when duplicates are deleted)	Number of attacks + normal
KDD Cup'99 10%	494,021	154,585	23
KDD Cup Test	311,029	N/A*	38

* Duplicates in the KDD Cup test set are not removed

The KDD Cup 10% and KDD Cup test sets have 494,021 and 311,029 instances respectively. Each instance has one label (output). The KDD Cup 10% has 22 attack types plus one normal record, the test set has 37 attack types plus one normal record. The attacks presented in the KDD Cup dataset and their corresponding class names are shown in Table 3 which is in Section 2.1.1. In the experiments, the dataset is used in two different forms; in the first form the output records are the class names/categories while in the second form the output records are the attack names. Both forms of datasets are actually the same but their output types are converted from attack names to class names or vice versa as shown in Figure 28.

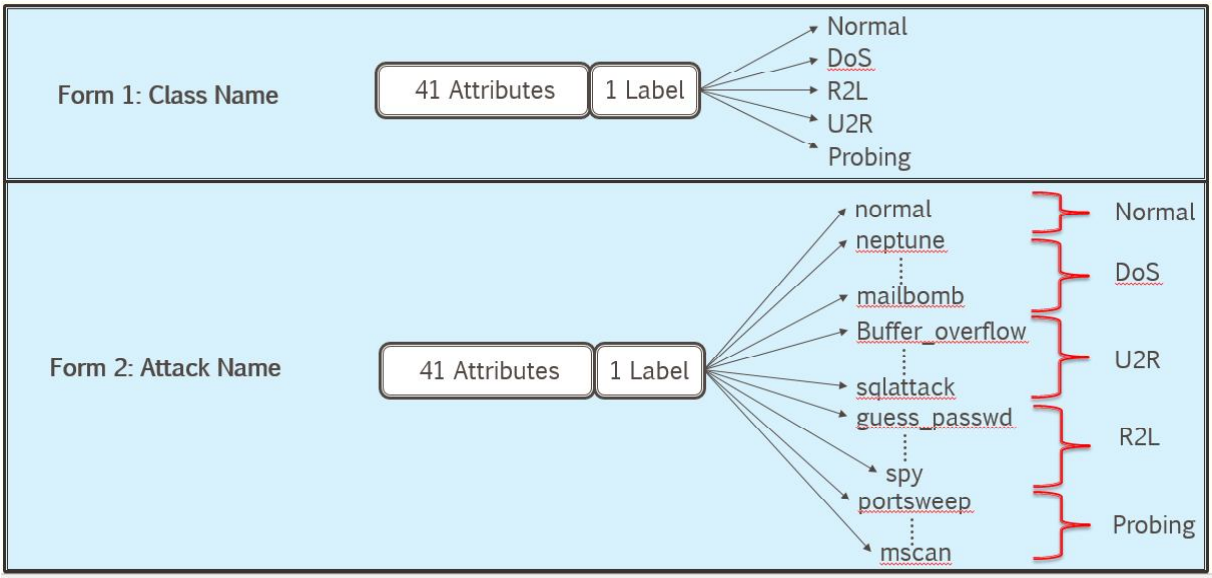

Figure 28 Description of both Form 1 and Form 2 datasets

6.3.2.2. Pre-processing ISCX Dataset

The original ISCX dataset has binary labels (Normal and Attack). The attack labels are changed to its corresponding name. As it can be seen in Table 7, which can be found in Section 2.1.3, different attacks are applied each day. The labels named as "attack" are replaced with their corresponding attack names which are obtained from Table 7. For instance

during 12/6/2010 "BruteForce" attack is conducted; therefore, the "attack" labels are replaced with "BruteForce" .

The dataset consists of 16 features (number of bytes, packets, payloads, TCP flag descriptions, IP addresses, and port numbers for both source and destination, time stamp, application name, protocol, and direction) and Tag (label). The IP addresses and Port numbers can be different for the same attack in different networks. Thus IP addresses and Port numbers cannot generalize the characteristic of attacks. Therefore the IP addresses, Port numbers and payloads are removed for the experiments. As a result the following 10 features remained in the ISCX dataset: application name (a1), total source bytes (a2), total destination bytes (a3), total destination packets (a4), total source packets (a5), direction (a6), source TCP flag description (a7), destination TCP flag description (a8), protocol name (a9), and duration (a10). Then repeated records are removed. The remaining dataset (Table 34) has 1.21% attack rate which is similar to the ratio of real network communication [39]. Train and test datasets are formed from the ISCX dataset in the following ratio; 60% and 40%, respectively. Both train and test datasets have preserved the normal/attack ratios which are 98.79% / 1.21% .

Table 34 Attack distribution of ISCX dataset (without repeated data)

	Number of packets	**Percentage**
Normal	911,206	98.79%
BruteForce	1,064	0.12%
DDoS	8,507	0.92%
HTTPDoS	818	0.09%
Infiltrating	725	0.08%
Total	922,320	100.00%

Both train datasets (KDD Cup 10% and ISCX) are pre-processed for the DFBFS and for the BP algorithm independently. Before applying DFBFS, continuous values are converted into discrete numbers by using the Entropy based discretization method [121]. This method recursively splits the attribute at a cut giving the maximal information gain. All continuous features are discretized. For instance feature A1 of KDD Cup dataset ranges between 0-58329 and is shown in Table 35. On the other hand for the Backpropagation (BP) algorithm strings are converted into discrete numerical values and then numeric values are normalized to the range of [0,1] to avoid attributes with higher values dominating attributes with smaller values.

Table 35 The discretized values of A1

Attribute: A1					
1	<=0.00	8	(30.00, 1031.00]	15	(10134.00, 12705.00]
2	(0.00, 4.00]	9	(1031.00, 4990.00]	16	(12705.00, 14682.00]
3	(4.00, 8.00]	10	(4990.00, 5025.00]	17	(14682.00, 15127.00]
4	(8.00, 11.00]	11	(5025.00, 5064.00]	18	(15127.00, 15168.00]
5	(11.00, 12.00]	12	(5064.00, 5085.00]	19	(15168.00, 20940.00]
6	(12.00, 14.00]	13	(5085.00, 10039.00]	20	(20940.00, 30190.00]
7	(14.00, 30.00]	14	(10039.00, 10134.00]	21	>30190.00

6.3.2.3. Feature Selection with DFBFS

The DFBFS is applied on the discretized KDD Cup 10% data as shown in Figure 29. In total 930 subsets each with 12-16 features are obtained. The complete list of the candidate subsets can be found in Appendix A. The backpropagation algorithm, written in C, is applied on all 930 subsets in order to find the subsets that provide the highest DRs for each class as shown in Figure 30. The KDD Cup 10% has 23 types of outputs; therefore, the BP algorithm is designed with 23 outputs and 23 hidden layers. The number of inputs is changed according to the number of features of each subset.

Algorithm 1: Wrapper based Feature Selection with DFBFS and BP

Input:
TS1, KDD Cup'99 10% with Attack Names (Form 2);
Output:
C*FSub*, Candidate Feature Subsets*; FSub,* selected Feature Subsets*,*
begin
CFSub =Apply DFBFS on *TS1*
for each CFSub
apply BP on *TS1*
save overall Accuracy and DRs for each class.
end
FSub = 10 CFSub that give the highest two values of DRs of each class (5 classes) and 2 CFSub that give the highest two values of overall Accuracy
end

Figure 29 Algorithm for applying wrapper based FS with DFBFS & BP on the KDD Cup'99 dataset

The BP algorithm for each training (154585 records with 1000 iterations) took about 8 hours on a Windows virtual server machine with 20 GB RAM and 8 CPUs each with 2.39 GHz speed. Technically the elapsed time for the whole training would be about 310 days (930 training x 8 hours/training). To shorten the elapsed time, the experiments are conducted on 4

virtual servers with parallel processing. Each machine did 7 trainings simultaneously. As a result the trainings for all 930 subsets took about 10 days.

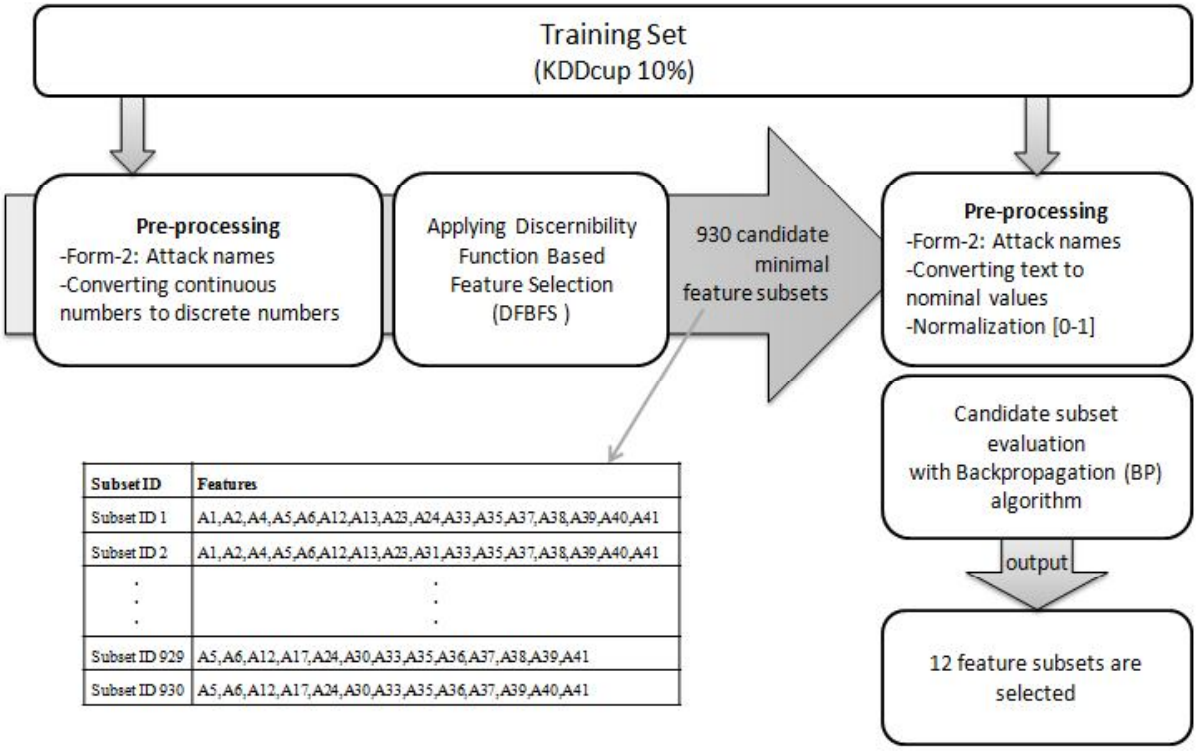

Figure 30 Flowchart of applying the DFBFS and evaluating the subsets obtained by the DFBFS.

Table 36 Highest two DRs of each class, overall accuracy and their corresponding subset numbers

	Feature Subsets	Detection Rates (DRs)					Accuracy	FAR
		Normal	DoS Attacks	U2R Attacks	R2L Attacks	Probing Attacks		
1	Subset ID 413	**0.997013**	0.965974	0.014286	0.002875	0.635862	0.919641	0.002987
2	Subset ID 3	**0.996930**	0.965961	0.000000	0.000000	0.565290	0.919335	0.003070
3	Subset ID 92	0.983364	**0.972108**	0.014286	0.000734	0.664906	0.921454	0.016636
4	Subset ID 912	0.984041	**0.971538**	0.014286	0.000122	0.716275	0.923682	0.015959
5	Subset ID 149	0.984140	0.966061	**0.342857**	0.006179	0.236678	0.919692	0.015860
6	Subset ID 155	0.983183	0.965861	**0.300000**	0.000000	0.434470	0.920425	0.016817
7	Subset ID 708	0.984685	0.827094	0.000000	**0.052181**	0.723236	0.818371	0.015315
8	Subset ID 843	0.983595	0.966566	0.000000	**0.049856**	0.697552	0.921695	0.016405
9	Subset ID 670	0.983463	0.968062	0.000000	0.000000	**0.821171**	0.920638	0.016537
10	Subset ID 352	0.983794	0.82518	0.000000	0.007524	**0.81517**	0.816226	0.016206
11	Subset ID 121	0.984305	0.969093	0.228571	0.001835	0.463514	**0.924756**	0.015695
12	Subset ID 633	0.994966	0.970037	0.042857	0.00263	0.43447	**0.92438**	0.005034

The DFBFS has computed 930 candidate feature subsets in a less time compared to other discernibility matrix-based approaches; however, finding the optimal feature subsets is computationally costly.

The DR for each class, overall accuracy, and FAR values are calculated with each subset that is evaluated with the BP algorithm. Then the subsets with the best two DRs and accuracy are adopted for future experiments. Table 36 describes the best two DRs for each class and overall accuracy. The results for all 930 feature subsets can be found in Appendix B. The attributes for the adopted subsets is presented in Table 37.

Table 37 The 12 selected feature subsets and their attributes

Subset ID	**Number of attributes**	**Attributes**
3	16	A1,A2,A4, A5, A6, A12, A13, A24, A29, A33, A35, A37, A38, A39, A40, A41
92	14	A1,A3,A4,A5,A6,A12,A17,A24,A30,A33, A35,A36,A37,A39
121	14	A1,A3,A5,A6,A10,A12,A24,A29,A33,A36,A37,A38,A39,A40
149	14	A1,A3,A5,A6,A12,A14,A24,A25,A29,A33,A36,A37,A39,A40
155	14	A1,A3,A5,A6,A12,A14,A24,A29,A33,A36,A37,A38,A39,A40
352	15	A1,A5,A6,A12,A13,A24,A29,A30,A33,A34,A35,A37,A39,A40,A41
413	15	A2,A4,A5,A6,A10,A12,A23,A24,A33,A35,A37,A38,A39,A40,A41
633	14	A3,A5,A6,A10,A12,A24,A29,A33,A36,A37,A38,A39,A40,A41
670	14	A3,A5,A6,A12,A13,A24,A29,A30,A33,A34,A35,A37,A39,A40
708	14	A3,A5,A6,A12,A14,A24,A27,A30,A33,A34,A35,A37,A39,A40
843	14	A3,A5,A6,A12,A18,A24,A29,A32,A33,A35,A37,A38,A39,A40
912	13	A5,A6,A10,A12,A24,A30,A33,A35,A36,A37,A39,A40,A41

DFBFS is applied on the ISCX dataset as well. Since the ISCX dataset has only 10 features and 4 attack types, only 3 minimal feature subsets are obtained. Therefore there is no need to apply a wrapper based selection to decrease the feature subsets.

6.3.3. Experiments and Results

6.3.3.1. IDS with Back-propagation Algorithm

This study used the Back-propagation (BP) algorithm, written in the C programming language. The structure of the Artificial Neural Networks (ANN) for the ISCX dataset is chosen as N inputs where N is equal to the number of attributes, N+1 hidden layers and 5 outputs (Normal, BruteForce, Infiltrating, HTTPDoS and DDoS). Since the KDD Cup dataset

is used in two different forms and has many feature subsets, applying BP algorithm took more procedures than applying BP on the ISCX dataset. The procedures are explained as follows: Firstly, the structure of Artificial Neural Networks (ANN) for the KDD Cup dataset is changed according to the train data and its output types. If the class names (Normal, DoS, Probing, U2R, and R2L) are used as the output type, the number of nodes in the output layer is set to 5; otherwise, the number of nodes is equal to the number of attacks presented in the train data. Secondly, the input layer depends on the feature subsets. Some subsets have 13 features, in this case the number of input nodes is set to 13. Lastly, the number of nodes in the hidden layer is chosen as X or X+1 where X is equal to MAX (number of nodes in Input Layer, number of nodes in Output Layer).

Table 38 The number of nodes in the input, hidden and output layer for the ANN structure for each data type

Data Type	**Number of nodes in**		
	Input Layer	**Hidden Layer**	**Output Layer**
Form 2: Attack names	13 - 16	23	23
Form 1:Class names	13 - 16	17	5

Algorithm 2: Applying BP algorithm for the KDD Cup datasets

Input:
TS1, KDD Cup dataset with Attack Names (Form 2); *TS2*, KDD Cup dataset with Category Names (Form 1); *TS*, KDD Cup Test set; *FSub*, Feature Subsets

Output:
DRs, Detection Rates; *FAR*, False Alarm Rate; *AC*, Accuracy

```
begin
    for trainset= TS1, and trainset= TS2,
      if trainset= TS1, Create 23 hidden and 23 output nodes
       if trainset= TS2, Create 17 hidden and 5 output nodes
       while any FSub is left do
            Select next FSub
            Create Classifier, Train with BP algorithm
            Test TS
            Save DRs, FAR and AC
       end
    end
end
```

Figure 31 Algorithm for testing the selected 12 feature subsets with BP

At the beginning of the algorithm the structure of the ANN is set according to the output type as shown in Table 38. Then, all 12 subsets are evaluated with the BP algorithm. Afterward,

the weights, obtained from the training phase, are used in the testing phase where the KDD Cup Test set is used. As shown in Figure 31 the outputs are saved for each subset and the process is repeated with the next output type.

The results of this algorithm (Figure 31) are given in Table 39 and Table 40 applied on KDD Cup (the full) dataset and KDD Cup 10% dataset, respectively. It can be seen that using the KDD Cup 10% in Form 2 (attack names) gives good results. This might be because the DFBFS is applied on the same dataset with the same data form.

Table 39 The results of the BP algorithm applied on the KDD Test set, train set is full KDD Cup dataset

Output number *	Subset ID	Detection Rates					Accuracy	FAR
		Normal	DoS	U2R	R2L	Probing		
5	3	0.989025	0.790188	0	0	0.220115	0.784891	0.010975
5	92	0.985015	0.790592	0	0.000061	0.108737	0.803311	0.014985
5	121	0.987391	0.647057	0	0	0.482477	0.808767	0.012609
5	149	0.993696	0.075287	0	0.002325	0.036966	0.388414	0.006304
5	155	0.988167	0.789796	0	0	0.454153	**0.914092**	0.011833
5	352	0.986319	0.646792	0	0.000245	0.035526	0.731025	0.013681
5	413	0.997425	**0.852723**	0	0	0.068171	0.832225	**0.002575**
5	633	0.987820	0.789561	0	0	0.598896	0.913349	0.012180
5	670	**0.998333**	0.644112	0	0	0.561690	0.809420	0.001667
5	708	0.983381	0.645413	0	0.000061	0.041527	0.678921	0.016619
5	843	0.988381	0.788426	0	0	0.107297	0.777992	0.011619
5	912	0.988695	0.646531	0	0	0.000960	0.672217	0.011305
20	3	0.986154	0.787625	0	0	0.621219	0.782499	0.013846
20	92	0.983810	0.646505	0	0.000245	0.764282	0.679683	0.016190
20	121	0.983711	0.644812	0	0.000184	0.813970	0.679075	0.016289
20	149	0.982110	0.780769	0	**0.030097**	0.804129	0.780676	0.017890
20	155	0.995363	0.786946	0	0.000061	0.782525	0.785956	0.004637
20	352	0.984503	0.645665	0	0	**0.872780**	0.680637	0.015497
20	413	0.986368	0.788278	0	0	0.630101	0.783142	0.013632
20	633	0.983249	0.648336	0	0.000122	0.762362	0.680895	0.016751
20	670	0.984866	0.645961	0	0	0.832933	0.680393	0.015134
20	708	0.984800	0.642485	0	0	0.805089	0.677438	0.015200
20	843	0.983959	0.074922	0	0.018658	0.748920	0.258069	0.016041
20	912	0.991534	0.644481	0	0.000306	0.605377	0.677567	0.008466

*5= Form 1: Class Names; 20= Form 2: Attack Names

The results of the BP algorithm applied on the ISCX dataset is shown in Table 41. It shows that the BP algorithm is not able to detect any attack other than BruteForce, the BP algorithm used with 4 features (a1,a3,a7, and a10) / tester id number 8 detected normal 100%. However

this classifier is insignificant, because it doesn't classify any attack. Thus tester id number 8 is discarded.

Table 40 The results of the BP algorithm applied on the KDD Test set, train set is KDD Cup 10% dataset

Output number*	Subset ID	Detection Rates					Accuracy	FAR
		Normal	DoS	U2R	R2L	Probing		
5	3	0.999290	0.253684	0	0	0	0.384700	0.000710
5	92	0.793276	0.794286	0	0	0	0.776911	0.206724
5	121	0.988299	0.964460	0	0	0.266203	0.918316	0.011701
5	149	0.989949	0.967392	0	0	0.070811	0.918580	0.010051
5	155	0.985312	0.965861	0	0	0.136822	0.917496	0.014688
5	352	0.984899	0.826894	0	0	0.231637	0.814744	0.015101
5	413	**0.999488**	0.791262	0	0	0.093135	0.782171	**0.000512**
5	633	0.974320	0.790462	0	0	0	0.775082	0.025680
5	670	0.969584	0.827681	0	0	0	0.811355	0.030416
5	708	0.987721	0.969968	0	0	0.039606	0.915448	0.012279
5	843	0.997293	0.253819	0	0	0.128421	0.814352	0.002707
5	912	0.985163	0.790940	0	0	0.534325	0.788550	0.014837
23	3	0.996930	0.965961	0.000000	0.000000	0.565290	0.919335	0.003070
23	92	0.983364	**0.972108**	0.014286	0.000734	0.664906	0.921454	0.016636
23	121	0.984305	0.969093	0.228571	0.001835	0.463514	**0.924756**	0.015695
23	149	0.984140	0.966061	**0.342857**	0.006179	0.236678	0.919692	0.015860
23	155	0.983183	0.965861	0.300000	0.000000	0.434470	0.920425	0.016817
23	352	0.983794	0.825180	0	0.007524	0.815170	0.816226	0.016206
23	413	0.997013	0.965974	0.014286	0.002875	0.635862	0.919641	0.002987
23	633	0.994966	0.970037	0.042857	0.002630	0.434470	0.924380	0.005034
23	670	0.983463	0.968062	0	0	**0.821171**	0.920638	0.016537
23	708	0.984685	0.827094	0	**0.052181**	0.723236	0.818371	0.015315
23	843	0.983595	0.966566	0.000000	0.049856	0.697552	0.921695	0.016405
23	912	0.984041	0.971538	0.014286	0.000122	0.716275	0.923682	0.015959

*5= Form 1: Class Names; 23= Form 2: Attack Names

Table 41 The results of the BP algorithm applied on the ISCX Test set

Classification Algorithm	Attributes	ID	Overall Accuracy	Normal	BruteForce	Infiltrating	HTTPDoS	DDoS
BP	All	5	**0.984864**	**0.995780**	**0.932084**	0	0	0
	a1,a2,a3,a4,a5,a6,a7,a8,a10	6	0.953991	0.964668	0.814988	0	0	0
	a1,a6,a7,a10	7	0.981972	0.992990	0.814988	0	0	0
	a1,a3,a7,a10	8	0.987936	1.000000	0	0	0	0

6.3.3.2. IDS with J48 Algorithm

The J48 is the implementation of the C4.5 Decision Tree learner, written for the WEKA software. It is taken as a separate algorithm to evaluate the selected 12 feature subsets of the KDD Cup dataset. The steps are shown in Figure 32.

Algorithm 3: Applying C4.5 Algorithm for the KDD Cup dataset

Input:
TS1, KDD Cup'99 10% with Attack Names; *TS2*, KDD Cup'99 10% with Category Names; *TS*, KDD Cup Test set; *FSub*, Feature Subset

Output:
DRs, Detection Rates; *FAR*, False Alarm Rate; *AC*, Accuracy

```
begin
    for trainset= TS1, and trainset= TS2,
        while FSub is left do
            Select next FSub
            Create Classifier, Train with C4.5 algorithm
            Test TS
            Save DRs, FAR and AC
        end
    end
end
```

Figure 32 Algorithm for testing the selected 12 feature subsets with C4.5

The Decision Tree that is obtained with WEKA J48 algorithm is used on another program written in C to apply the tree on the Test set. Excellent results are obtained with the J48 algorithm as shown in Table 42. Especially 81% of U2R attacks, 84% of R2L attacks and over 99% of Probing attacks are detected which are much higher than results obtained for other related studies. Normal data is detected with an average of 94% rate while the FAR is experienced at approximately 5%.

Table 39 and Table 40 show that using the dataset in different forms (different output types) reveals various DRs. Some classes are detected better with the dataset that uses the attack names as output, while some classes are better detected with the dataset that uses class names as output.

Table 42 The results of the J48 algorithm applied on the KDD Test set, train set is KDD Cup 10% dataset

Output number*	Subset ID	Detection Rates					Accuracy	FAR
		Normal	Probing	DoS	U2R	R2L		
23	3	0.942701	0.991599	**0.999935**	0.657143	0.835689	0.979963	0.057299
23	92	**0.949368**	0.993039	0.999930	**0.814286**	0.827552	0.980886	0.050632
23	121	0.949104	0.990639	0.999843	0.700000	0.825044	0.980581	0.050896
23	149	0.949104	0.990158	0.999869	0.642857	0.825901	0.980626	0.050896
23	155	0.951381	0.986558	0.999839	0.700000	0.785710	0.978899	0.048619
23	352	0.942948	0.993759	0.999930	0.542857	0.836606	0.980060	0.057052
23	413	0.942750	0.990639	0.999930	0.457143	0.836912	0.979976	0.057250
23	633	0.949219	0.990398	0.999852	0.542857	0.824433	0.980539	0.050781
23	670	0.943559	**0.994479**	0.999917	0.642857	0.840154	0.980388	0.056441
23	708	0.943575	0.993999	0.999922	0.671429	0.838870	0.980327	0.056425
23	843	0.943295	0.988958	0.999900	0.657143	0.838625	0.980172	0.056705
23	912	0.948922	0.991839	0.999917	0.571429	0.824371	0.980552	0.051078
5	3	0.942734	0.990639	0.999926	0.728571	0.836790	0.980024	0.057266
5	92	0.949054	0.993519	0.999917	0.757143	0.828837	0.980876	0.050946
5	121	0.948790	0.987998	0.999830	0.814286	0.826207	0.980561	0.051210
5	149	0.948790	0.988238	0.999835	0.814286	0.827063	0.980613	0.051210
5	155	0.948757	0.988478	0.999830	0.814286	0.827063	0.980606	0.051243
5	352	0.942007	0.994239	0.999926	0.742857	0.839971	0.980102	0.057993
5	413	0.942404	0.992079	0.999939	0.585714	0.836728	0.979954	0.057596
5	633	0.948906	0.988478	0.999830	0.800000	0.825778	0.980565	0.051094
5	670	0.943575	0.993519	0.999917	0.728571	0.839604	0.980368	0.056425
5	708	0.943278	0.993039	0.999909	0.757143	**0.840521**	0.980352	0.056722
5	843	0.943344	0.993039	0.999939	0.900000	0.839359	0.980359	0.056656
5	912	0.948889	0.993759	0.999917	0.728571	0.826268	0.980706	0.051111

*5= Form 1: Class Names; 23= Form 2: Attack Names

Table 43 The results of the J48 algorithm applied on the ISCX Test set

Attributes	ID	Overall Accuracy	Normal	BruteForce	Infiltrating	HTTPDoS	DDoS
All	1	0.**995679**	**0.999734**	0.950820	0.133803	0.750760	**0.662750**
a1,a2,a3,a4,a5, a6,a7,a8,a10	2	0.995332	0.999506	0.950820	0.133803	0.750760	0.649530
a1,a6,a7,a10	3	0.992831	0.999561	**0.981265**	**0.197183**	**0.790274**	0.359577
a1,a3,a7,a10	4	0.994121	0.999641	0.913349	0.109155	0.644377	0.520858

Same J48 algorithm is applied on the ISCX dataset with WEKA to evaluate the feature subsets. Results of J48 for each feature subset are shown in Table 43. It is noted that J48 is

able to detect all attack types while BP (see Table 41) only detects BruteForce attacks and Normal packets. In the experiments with J48, J48, the highest DRs of Normal and DDoS are obtained through tester id number 1 with 99.9% and 66.3% rates, respectively. The highest DRs of BruteForce, Infiltrating and HTTPDoS are obtained through tester id number 3 with 98.1%, 19.7%, and 79%, respectively. The DR of Infiltrating attacks is very low because during infiltrating the attacker only gathers network information; therefore, this attack type shows similar characteristics with Normal behaviors. For this reason Infiltrating attacks are usually detected as Normal packets. BP is able to detect only Normal packets and BruteForce attacks which are best detected through tester id number 5 as 99.5% and 93.2%, respectively. Even though tester id number 8 has 100% DR for Normal packets, this tester is discarded because it is not classifying any attack.

6.3.3.3. A Proposed Hybrid IDS - Multilevel Hybrid Classifier with Variant Feature Sets (MHCVF)

According to the experiments, which are conducted on the KDD Cup dataset Normal connections are best detected with the BP algorithm which is trained with the KDD 10% dataset that has the class names as the output label. On the other hand, the highest DRs of DoS, Probing, and U2R attacks are obtained with the C4.5 (J48) algorithm while each of them is better with different feature subsets and dataset forms as shown in Table 44. R2L attacks are also best detected with the C4.5 (J48) algorithm with Form-1 dataset (class name taken as the output type). The building time of each tester is also given in Table 44.

Table 44 The algorithm, dataset form and feature subset ID that give the best result for each class

Tester Name	Class	Algorithm	Dataset Form	Feature Subset ID	Detection Rate (DR)	Building Time (sec)
Normal Tester	Normal	BP	Form 1: Class Names	413	0.99949	170
DoS Tester	DoS	J48	Form 2: Attack Names	3	0.999935	3355
Probing Tester	Probing	J48	Form 2: Attack Names	670	0.994479	2692
U2R Tester	U2R	J48	Form 2: Attack Names	92	0.814286	2303
R2L Tester	R2L	J48	Form 1: Class Names	708	0.840521	6132

Since each class is detected with different methods, a hybrid architecture is proposed to detect intrusions. In this model the KDD Cup test data is evaluated as shown in Figure 33. This model first applies the R2L tester, and it labels the record as R2L if it is predicted as R2L by

the tester. If the record is not predicted as R2L, then it is passed to the next tester and so on. The rest of the records are predicted for all classes at the last tester, which is the Normal tester.

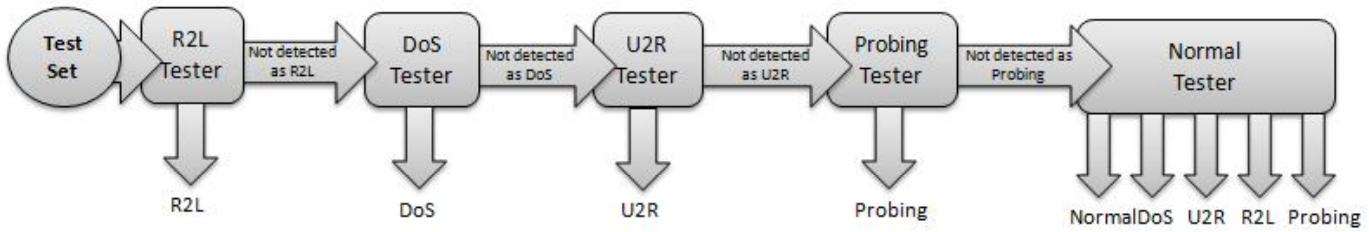

Figure 33 The model of MHCVF for KDD

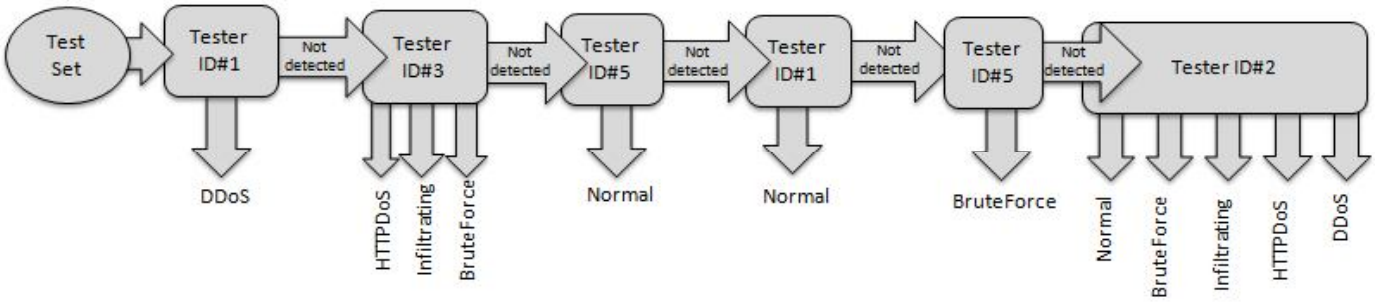

Figure 34 The model of MHCVF for ISCX

Table 45 Comparison of the hybrid model with individual classifiers (KDD dataset)

Given Name	Algorithm	Detection Rates					Accuracy	FAR
		Normal	DoS Attacks	Probing Attacks	U2R Attacks	R2L Attacks		
Normal Tester	BP	0.9995	0.7913	0.0931	0.0000	0.0000	0.7822	0.0005
DoS Tester	C4.5	0.9427	0.9999	0.9916	0.6571	0.8357	0.9800	0.0573
Probing Tester	C4.5	0.9436	0.9999	0.9945	0.6429	0.8402	0.9804	0.0564
U2R Tester	C4.5	0.9494	0.9999	0.9930	0.8143	0.8276	0.9809	0.0506
R2L Tester	C4.5	0.9433	0.9999	0.9930	0.7571	0.8405	0.9804	0.0567
Hybrid	BP & C4.5	0.9429	0.9999	0.9939	0.8	0.8405	0.9803	0.0571

The best DR for each algorithm is chosen in the hybrid model for ISCX dataset. For instance, a Normal tester is chosen both from J48 algorithm (id number 1) and from the BP algorithm (id number 5). The same way classifier id number 3 and id number 5 are chosen for BruteForce tester. The remaining attacks were only detected with the J48 algorithm; therefore one tester for each of these attacks is selected. In the hybrid model, the ISCX dataset is

evaluated as shown in Figure 34. Some of attacks and normal packets are detected by multiple testers. If the first one does not identify the attack then it may be detected by the next corresponding tester, which improves the DR.

Table 46 Comparison of the hybrid model with individual classifiers (ISCX dataset)

Given Name	Algorithm	Detection Rates					Accuracy	FAR
		Normal	BruteForce Attacks	Infiltrating Attack	HTTPDoS Attacks	DDoS Attacks		
Tester ID number 1	C4.5	0.999734	0.95082	0.133803	0.75076	0.66275	0.995679	0.00026613
Tester ID number 2	C4.5	0.999506	0.95082	0.133803	0.75076	0.64953	0.995332	0.000493849
Tester ID number 3	C4.5	0.999561	0.981265	0.197183	0.790274	0.359577	0.992831	0.000438977
Tester ID number 5	BP	0.99578	0.932084	0	0	0	0.984864	0.002965837
Hybrid	BP & C4.5	0.99969	0.990632	0.197183	0.790274	0.66275	0.995766	0.00031

Table 47 Confusion matrix obtained with the MHCVF for test dataset

		Predicted				
		Normal	Probing	DoS	U2R	R2L
Actual	Normal	57134	23	42	0	3395
	Probing	15	4141	7	1	2
	DoS	12	5	229835	0	1
	U2R	12	0	0	56	2
	R2L	2607	0	0	0	13740

Table 48 Confusion matrix obtained with MHCVF for ISCX Test set

		Predicted				
		Normal	BruteForce	Infiltrating	HTTPDoS	DDoS
Actual	Normal	364371	40	9	14	50
	BruteForce	4	423	0	0	0
	Infiltrating	214	4	56	2	8
	HTTPDoS	46	0	23	260	0
	DDoS	1148	0	0	0	2256

The orders of the testers in the hybrid method of KDD and ISCX are obtained empirically and are chosen according to their CPE values and DR rates of the testers. The comparisons of the hybrid models with individual classifiers are shown in Table 45 and Table 46 for KDD and

ISCX, respectively. Both results show that hybrid methods outperform base classifiers for most of the DRs of attacks.

The confusion matrixes of the hybrid method for both datasets are presented in Table 47 and Table 48.

6.3.3.4. Results

The performance metrics of the proposed IDS are compared with other methods that are presented on the literature (Table 49 and Table 50). It can be noted that there are some papers that use the KDD Test set with their proposed methods [51], [64], [122], [123], but those studies include the http-tunnel attack in the U2R class [64], [122]. In fact, the http-tunnel attack is a Remote-to-Local (R2L) attack [37] which intends to gain local access from a remote machine. The http-tunnel attack creates a covert channel between the victim machine and the remote attacker that looks like normal web browsing connections. By using the covert channel the attacker is able to install files in the victim machine and run UNIX commands remotely [37]. The total number of U2R attacks is 228 in [6], [7], [45], [46], [51], [61], [64], [122], [123] where the http-tunnel attack is considered as a U2R attack, while in this study the http-tunnel attack is taken as a R2L attack and the number of U2R attacks is 70. One must note that, for a meaningful comparison attack categories must include the same data.

Table 49 compares the proposed method with other studies which used the corrected KDD Cup Test set. It shows that all methods tested on the KDD Test set offered an acceptable level of detection rates for Normal records. In terms of Probe, DoS, U2R, and overall accuracy it can be seen that the proposed MHCVF model demonstrates better performance. The method gives especially high DR (99.99%) in DoS attacks. One should note that the KDD Test dataset includes 18,729 new kinds of attacks. A total of 6,555 of these are new DoS attacks (apache2, mailbomb, processtable, and udpstorm) that never appear in the train set. While the total number of DoS attacks in Test set is 229,853, the percentage of new DoS attacks is only 2.85%. Even though they are new attacks, they show similar characteristics with existing DoS attacks. Therefore the method is able to detect new DoS attacks as “neptune" and “back” which are known attacks. “Clustering feature [64]” and “Multiple-level hybrid classifier [13]” are also able to detect these new attacks with 99.53% and 98.66% detection rates, respectively. Most particularly, the DR for U2R has a noticeable difference with other methods. Moreover, extremely good DR for R2L which is very close to the rate presented in [51] is obtained. With respect to CPE, the proposed method achieved as low as 0.056 being

remarkably better than 0.233, 0.181, and 0.222 which are the CPE's reported in [122], [51], and [120], respectively. But it does not demonstrate an acceptable False Alarm Rate (FAR). The FAR found to be 5.71% which is considered high.

Table 49 Comparisons of the proposed method with other studies (KDD Cup Test set)

Method	Normal %	Probe %	DoS %	U2R %	R2L %	Accuracy %	CPE
MHCVF (proposed method of this thesis)	94.29	99.39	99.99	80.00	84.05	98.03	0.056
Clustering feature [64]	99.29	97.55	99.53	19.73	28.81	95.72	not reported
PLSSVM [51]	95.69	86.46	78.76	30.70	84.85	not reported	0.181
Misuse detection approach [71]	91.00	88.50	78.90	68.60	6.20	91.00	not reported
Multiple-level hybrid classifier [13]	96.80	93.40	98.66	71.43	46.97	96.78	not reported
KDD'99 winner [122]	99.50	97.10	83.30	13.20	8.40	not reported	0.233

The results obtained with the ISCX dataset were compared against the literature shown in Table 50. The method gives promising results in Accuracy and DR of Normal packets. Moreover the FAR is lower than the FARs of other studies. The DR of attacks is 68% which is very low according to [71], [124], [125] where the DRs are over 90%. However, it should be noted that the numbers in Table 50 are not comparable since they are not conducted on the same part of the ISCX dataset. The ISCX dataset is not divided as train and test sets by the provider; therefore, each study has selected part of the ISCX dataset and divided it into as train and test sets in its own way.

Table 50 Comparisons of the proposed method with other studies (ISCX dataset)

Method	Normal %	Attack %	Accuracy %	FAR %
MHCVF (proposed method of this thesis)	99.9	68.2	99.5	0.03
ALL-AGL [124]	99.5	93.2	95.4	0.30
KMC+NBC [125]	97.7	99.7	99.0	2.2
AMGA2-NB [126]	95.2	92.7	94.5	7.0

Since many of researchers only use a part of the train and test sets, it is difficult to compare the training and testing time. The studies that are listed in Table 49, used the entire test dataset but only two of them report the execution time of training and testing, which are shown in Table 51. Experiments were conducted on a PC with 2.39 GHz CPU, 16 GB of RAM and Windows 7 operating system. The training took longer time than in PLSSVM [51]. But in PLSSVM [51] the training experiments were not done on the whole dataset; therefore, one cannot claim that it is faster. In general, it can be stated that the training and testing times are not comparable since they all are executed on PCs with different specifications. On the other hand, the Multiple-level hybrid classifier method [13] is superior to MHCVF in terms of testing time which is as low as 4 minutes, even with a slower PC. In order to detect effectively in real-time the testing time of MHCVF should be improved.

Table 51 Comparisons of training and testing time

Method	**Size of Training Dataset**	**Total Training Time**	**Size of Testing Dataset**	**Total Testing Time**	**Environment of Experiments**
MHCVF (proposed method of this thesis)	154585 (after duplicates are deleted)	240 min	311029 (Entire tes set)	35 min	2.39 GHz CPU, 16 GB RAM
PLSSVM [51]	13183 (after random selection)	107 min	311029 (Entire tes set)	46 min	2.49 GHz CPU, 3 GB RAM
Multiple-level hybrid classifier [13]	13184 (after random selection)	1443 min	311029 (Entire tes set)	4 min	1.86 GHz CPU, 512MB RAM

It can be noted that changing the output type (converting the attack names to class names or vice versa) affects the detection rates of the testers. Additionally, according to the knowledge of the author, this study is the first one that uses the DFBFS on an intrusion detection dataset. The results show that different feature subsets provide different detection rates for each class. Therefore, each class is tested with its own optimum feature subset. Moreover, the results show that using the hybrid method of BP and C4.5 gives better detection rates than using the classifiers individually, as shown in Table 45. Consequently, the main contribution of this thesis is the MHCVF model that uses different learners and feature subsets for each class. The ISCX dataset provides high rates on Accuracy and FAR, whereas the DR for attacks is low and should be improved. On the other hand, the results for the KDD dataset demonstrate

approximately 12%, 2%, and 1% better accuracy rates on U2R, Probe, and DoS attack classes, respectively. Furthermore, a better Cost per Example (CPE) value which was the primary evaluation method in the KDD'99 Cup competition is obtained. So it can be stated that the MHCVF is superior to the existing IDS methods in terms of four attack classes and CPE value. However, the FAR value, obtained with the KDD Cup dataset, is computed as 5.71% which is considered to be too high. Therefore, MHCVF should be improved to decrease the FAR while preventing any decrease in the DRs of the four attack classes. This problem is left as a future work.

7. CONCLUSIONS AND FUTURE WORK

This thesis aimed on critically studying different machine learning methods for IDSs. In order to achieve this objective this thesis utilized feature selection and machine learning algorithms, and compared them to find a new solution. Additionally, this thesis has focused on adapting the IDS to new environments.

7.1. **Conclusions**

Phase I describes a study that emphasizes the importance of pre-processing. In this study the RBF is used to classify the dataset as attack or normal. String attributes such as Flag Name and Service Name are converted to numeric and then the whole dataset which only contains numbers is normalized to the range of [0-1]. This study showed that the process of assigning numbers to text data during pre-processing stage plays a role in accuracy. The conversion is done in three different ways and three different datasets are created. Even though they are basically the same dataset, each of them showed different detection rates and different mean square error (MSE) values.

In Phase II, this study applied transfer learning on IDSs to adapt them to new environments. According to the author's best knowledge, transfer learning method has not been applied before by any researcher to adapt IDSs to continuously changing environments. It has been proved that transfer learning is a very effective method to update the IDS for new environments. The experiments show that this method helps the learner to utilize previously learned knowledge. Since it uses the previous knowledge, it can be trained again by just adding the new-coming dataset to adapt itself to the new environment. This gives the opportunity to just being updated by a partial dataset and to not necessarily collect an extensive dataset again. It also helps to improve the speed of training. These utilities provide quicker way of training the system. It is assumed that collecting the dataset partially is possible. However it must be noted that to be able to detect and collect the data that express new network behavior is not the main aim of this thesis. In a nutshell, this part of the thesis is only limited with applying transfer learning on IDS.

Phase III presented improving the detection rates of attack classes and lightening the IDSs. One of the contributions of this thesis is the use of DFBFS method. This method is different

from traditional discernibility matrix-based approaches which generate all minimal subsets of features. However, the traditional discernibility matrix-based approaches consume huge memory and time even for a mid-sized dataset since the subsets are computed with an exponential complexity. Therefore usually by using this method, researcher only compute part of minimal feature subsets, not all of them. This may cause to overlook the most optimal feature subsets. The DFBFS method is able to compute all minimal feature subsets, since this method is faster and needs less memory than the traditional discernibility matrix-based approaches.

The DFBFS method is applied on the KDD Cup 10% dataset and 930 candidate feature subsets that represent the whole dataset are obtained. Then, the candidate subsets are evaluated using the BP algorithm to find 12 optimum feature subsets which are adopted for the rest of the experiments. Following this, the train dataset is converted into two forms with different output labels; Attack names and Class names (Table 39). The BP and C4.5 learners are applied on both forms of the train set with each adopted feature subset. Afterwards, the testers are applied on the corrected KDD Test set for performance comparisons.

Other than KDD Cup'99 dataset, the ISCX dataset which is a real intrusion detection dataset is also used in Phase III. The same way, DFBFS is applied on the ISCX dataset and the minimal feature subset were adopted. Since the ISCX dataset has only 10 features and just four attacks, the number of minimal subset were three. Therefore, there is no need to apply another algorithm as it is done for the KDD Cup'99 dataset to decrease the number of candidate feature subsets. Then, the BP and Decision Tree algorithms are applied on the ISCX dataset using all minimal feature subset.

One of the contributions of this study is that it shows that changing the output type (converting attack names to class names or vice versa) affects the detection rates of the testers. For instance in Table 39 it can be seen that the accuracies obtained with Form-1 and Form-2 using the subset ID 843 are about 77.8% and 25.9%, respectively. Again in the same table, the accuracies obtained with Form-1 and Form-2 using the subset ID 670 are about 80.9% and 68%, respectively. Table 39 is summarized in Table 52 to see how the output type (Form-1 and Form-2) affects the accuracy.

Additionally, according to the knowledge of the authors, this study is the first one that uses the DFBFS on an intrusion detection dataset. The results show that different feature subsets

provide different detection rates for each class. Therefore, each class is classified with its own optimum feature subset as shown in Table 44. Moreover, the results show that using the hybrid method of BP and C4.5 gives better detection rates than using the classifiers individually (see Table 45). Consequently, a hybrid MHCVF that uses different learners and feature subsets for each class is proposed. This is the main contribution of this research into the existing literature.

Table 52 The results of BP algorithm for KDD Test set

Train Dataset	Output Type	Subset ID	Accuracy	Subset ID	Accuracy
KDD Cup dataset	Form 1: Class Names	3	0.784891	413	0.832225
	Form 2: Attack Names		0.782499		0.783142
	Form 1: Class Names	92	0.803311	633	0.913349
	Form 2: Attack Names		0.679683		0.680895
	Form 1: Class Names	121	0.808767	670	0.809420
	Form 2: Attack Names		0.679075		0.680393
	Form 1: Class Names	149	0.388414	708	0.678921
	Form 2: Attack Names		0.780676		0.677438
	Form 1: Class Names	155	0.914092	843	0.777992
	Form 2: Attack Names		0.785956		0.258069
	Form 1: Class Names	352	0.731025	912	0.672217
	Form 2: Attack Names		0.680637		0.677567
KDD Cup 10% dataset	Form 1: Class Names	3	0.384700	413	0.782171
	Form 2: Attack Names		0.919335		0.919641
	Form 1: Class Names	92	0.776911	633	0.775082
	Form 2: Attack Names		0.921454		0.924380
	Form 1: Class Names	121	0.918316	670	0.811355
	Form 2: Attack Names		0.924756		0.920638
	Form 1: Class Names	149	0.918580	708	0.915448
	Form 2: Attack Names		0.919692		0.818371
	Form 1: Class Names	155	0.917496	843	0.814352
	Form 2: Attack Names		0.920425		0.921695
	Form 1: Class Names	352	0.814744	912	0.788550
	Form 2: Attack Names		0.816226		0.923682

The results compared in Table 49 with five different studies of IDSs that are applied on the full KDD Test set without any random selection. The highest detection rate of Normal connections was obtained by KDD99 classification cup winner [122] with 99.5% while the detection rate obtained with MHCVF is 94.29%. In Table 49, the highest DRs of Probe, DoS and U2R attacks were obtained with the proposed MHCVF method as 99.39%, 99.99%, and

80%, respectively, which are approximately 2%, 1% and 12% higher than other studies. The R2L detection rate was best obtained in PLSSVM [51] as 84.85%, while it is obtained as 84.05% with the proposed MHCVF method. It can be stated that the MHCVF method gives very close result compared to [51] in terms of R2L detection rate. Moreover the proposed MHCVF method demonstrates a better Cost per Example (CPE) value which was the primary evaluation method in the KDD'99 Cup competition. The CPE value is calculated as low as 0.056 whereas the ideal value is zero. On the other hand, by PLSSVM [51] and by KDD99 classification cup winner [122] the CPE values are 0.181 and 0.233 which are poorer than the CPU value in MHCVF.

So it is clear that the MHCVF is superior to the existing IDS methods in terms of four attack classes and CPE value. While the MHCVF produced 5.71% FAR value with the KDD Cup'99 dataset, it is %0.03 with the ISCX dataset. This difference is significant enough for future research studies.

7.2. **Future Work**

The testing time of the proposed MHCVF method should be improved to efficiently detect in real-time environments. The FAR value obtained with the KDD Cup dataset and attack DR obtained with the ISCX dataset should also be improved.

There are open-source IDS datasets that are recently served online such as: traffic traces from the WIDE backbone presented by the MAWI Working Group [127], packet-, flow- and http-traces presented by the MOME database [128], ADFA [129], CAIDA Datasets [31], and Waikato Internet Traffic Storage [130]. The proposed methods will be applied on these new datasets and the results can be compared with the results obtained with the KDD'99 and ISCX datasets to measure the effectiveness of the proposed methods.

Another future work is to generate a new dataset. The new dataset can be generated in a lab environment with synthetic data. The task can be split into sub tasks by focusing first on specific protocols such as HTTP, as it is already done in [20]. Other protocols can be added one by one to keep the job less complicated and to grow the network in a more controlled way. Another starting point can be: limiting the attacks in the network to one or to a small number; then incrementing the number of attacks. Obtaining a new dataset at a real network, where real communication packets flow, is much more complicated. However it will reflect

real network traffic, whereas evaluating the IDS with a real network traffic is always more reliable. However it is a challenging task since one cannot be 100% sure that the network is attack free.

Adapting the method on real-time detection instead of offline detection is left as future work. In real-time detection, the IDS has to be fast, lightweight, and reliable; meaning low FAR and high Accuracy. The IDS also should able to adapt itself to new environments where the network behaviors continuously change.

The proposed methods may also be modified and improved to detect attacks conducted on interconnection power electricity. Moreover, Big Data Mining Methods should be researched as well, because network datasets usually have millions of records which make it huge. Moreover, feature extraction using Principal Component Analysis (PCA) or Fuzzy Clustering would be another future work.

REFERENCES

[1] Ponemon Institute, "Perceptions About Network Security Survey of IT & IT security practitioners in the U . S," Ponemon Institute, U.S., Research Report, 2011.

[2] TNS Opinion & Social, "Cyber Security," General Home Affairs, Report, 2012.

[3] D. E. Denning, "An Intrusion-Detection Model," *IEEE Transactions on Software Engineering*, vol. 13, no. 2, pp. 222–232, Feb. 1987.

[4] G. Vigna, W. Robertson, V. K. V. Kher, and R. a. A. Kemmerer, "A stateful intrusion detection system for world-wide web servers," in *Proceedings of the 19th Annual Computer Security Applications Conference*, 2003, vol. 08–12, pp. 34–43.

[5] M. Bahrololum and M. Khaleghi, "Anomaly Intrusion Detection System Using Hierarchical Gaussian Mixture Model," *IJCSNS International Journal of Computer Science and Network Security*, vol. 8, no. 8, pp. 264–271, 2008.

[6] A. Bsila, S. Gombault, and A. Belghith, "Improving traffic transformation function to detect novel attacks," in *4th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications*, 2007, pp. 1–8.

[7] R. C. Chen, K. F. Cheng, Y. H. Chen, and C. F. Hsieh, "Using Rough Set and Support Vector Machine for Network Intrusion Detection System," in *2009 First Asian Conference on Intelligent Information and Database Systems*, 2009, pp. 465–470.

[8] W. Hu, W. Hu, and S. Maybank, "AdaBoost-Based Algorithm for Network," *IEEE Transactions on Systems, MAN, and Cybernetics—Part B: Cybernetics*, vol. 38, no. 2, pp. 577–583, 2008.

[9] J.-H. Lee, J.-H. Lee, S.-G. Sohn, J.-H. Ryu, and T.-M. Chung, "Effective Value of Decision Tree with KDD 99 Intrusion Detection Datasets for Intrusion Detection System," in *The 10th International Conference on Advanced Communication Technology (ICACT 2008)*, 2008, vol. 2, pp. 1170–1175.

[10] M. V. Mahoney, "Network Traffic Anomaly Detection Based on Packet Bytes," in *Proceedings of the 2003 ACM Symposium on Applied Computing - SAC '03*, 2003, pp. 346–350.

[11] S. T. Powers and J. He, "A hybrid artificial immune system and Self Organising Map for network intrusion detection," *Information Sciences*, vol. 178, no. 15, pp. 3024–3042, Aug. 2008.

[12] F. N. M. Sabri, N. M. Norwawi, and K. Seman, "Identifying False Alarm Rates for Intrusion Detection System with Data Mining," *IJCSNS International Journal of Computer Science and Network Security*, vol. 11, no. 4, pp. 95–99, 2011.

[13] C. Xiang, P. C. Yong, and L. S. Meng, "Design of multiple-level hybrid classifier for intrusion detection system using Bayesian clustering and decision trees," *Pattern Recognition Letters*, vol. 29, no. 7, pp. 918–924, May 2008.

[14] W. Feng, Q. Zhang, G. Hu, and J. X. Huang, "Mining network data for intrusion detection through combining SVMs with ant colony networks," *Future Generation Computer Systems*, vol. 37, pp. 127–140, Jul. 2014.

[15] R. a. Kemmerer and G. Vigna, "Intrusion Detection: A Brief History and Overview," *Security and Privacy a Supplement to IEEE Computer Magazine*, vol. 35, no. 4, pp. 27–30, Apr. 2002.

[16] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," *Applied Soft Computing*, vol. 10, no. 1, pp. 1–35, Jan. 2010.

[17] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in Cloud," *Journal of Network and Computer*

Applications, vol. 36, no. 1, pp. 42–57, Jan. 2013.
[18] D. Stiawan, A. Y. I. Shakhatreh, M. Y. Idris, K. Abu Bakar, and A. H. Abdullah, "Intrusion Prevention System : A Survey," *Journal of Theoretical and Applied Information Technology*, vol. 40, no. 1, pp. 44–54, 2012.
[19] M. Pradhan, S. K. Pradhan, and S. K. Sahu, "A Survey on Detection Methods in Intrusion Detection System," *International Journal of Computer Application*, vol. 3, no. 2, pp. 81–90, 2012.
[20] H. G. Kayacik and N. Zincir-Heywood, "Generating Representative Traffic for Intrusion Detection System Benchmarking," in *Proceedings of the 3rd Annual Communication Networks and Services Research Conference*, 2005, pp. 112 – 117.
[21] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection : A Survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, pp. 1–72, 2009.
[22] Y. Bai and H. Kobayashi, "Intrusion Detection Systems: technology and development," in *17th International Conference on Advanced Information Networking and Applications (AINA)*, 2003, pp. 710–715.
[23] H. Bensefia and N. Ghoualmi, "A new approach for adaptive intrusion detection," in *2011 Seventh International Conference on Computational Intelligence and Security*, 2011, pp. 983–987.
[24] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[25] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proceedings of the 24th International Conference on Machine Learning (ICML'07)*, 2007, pp. 193–200.
[26] mit.edu, "DARPA Intrusion Detection Data Sets," *Lincoln Labratory*. [Online]. Available: http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/index.html. [Accessed: 01-Jan-2013].
[27] uci.edu, "KDD Cup 1999 Data," *The UCI KDD Archive*, 1999. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. [Accessed: 05-Jul-2013].
[28] uml.edu, "Information Exploration Shootout," *Institute for Visualization and Perception Research, University of Massachusetts Lowell*, 2001. .
[29] takakura.com, "Kyoto Dataset," 2009. [Online]. Available: http://www.takakura.com/Kyoto_data/. [Accessed: 01-Jan-2016].
[30] A. Shiravi, H. Shiravi, M. Tavallaee, and A. a. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Computers and Security*, vol. 31, no. 3, pp. 357–374, 2012.
[31] Center for Applied Internet Data Analysis (caida.org), "CAIDA Data." [Online]. Available: http://www.caida.org/data/overview/. [Accessed: 24-Nov-2015].
[32] icir.org, "LBNL Enterprise Trace Repository," 2005. [Online]. Available: http://www.icir.org/enterprise-tracing/. [Accessed: 31-Dec-2015].
[33] shmoo.com, "DefCon," 2015. [Online]. Available: http://cctf.shmoo.com/. [Accessed: 30-Dec-2015].
[34] unibs.it, "UNIBS-2009," 2009. [Online]. Available: http://www.ing.unibs.it/ntw/tools/traces/. [Accessed: 30-Dec-2015].
[35] P. Gogoi, M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Packet and Flow Based Network Intrusion Dataset," in *Communications in Computer and Information Science*, vol. 306, no. August, M. Parashar, D. Kaushik, O. F. Rana, R. Samtaney, Y. Yang, and A. Zomaya, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 322–334.
[36] uci.edu, "UNIX User Data," *The UCI KDD Archive Information and Computer Science*

University of California, Irvine, 1998. [Online]. Available: http://kdd.ics.uci.edu/databases/UNIX_user_data/UNIX_user_data.html. [Accessed: 01-Aug-2013].

[37] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall, D. McClung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham, and M. A. Zissman, "Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation," in *DARPA Information Survivability Conference and Exposition (DISCEX'00)*, 2000, vol. 2, pp. 12–26.

[38] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proceedings of the Second IEEE Symposium on Computational Intelligence in Security and Defence Applications (CISDA)*, 2009, pp. 53–58.

[39] J. Song, H. Takakura, Y. Okabe, and Y. Kwo, "Correlation Analysis Between Honeypot Data and IDS Alerts Using One-class SVM," in *Intrusion Detection Systems*, P. Skrobanek, Ed. Shanghai: InTech Open Access Publisher, 2011, pp. 173–192.

[40] Y.-D. Lin, P.-C. Lin, S.-H. Wang, I.-W. Chen, and Y.-C. Lai, "PCAPLib: A System of Extracting, Classifying, and Anonymizing Real Packet Traces," *IEEE Systems Journal*, vol. PP, no. 99, pp. 1–12, 2014.

[41] S. Novakov, C.-H. Lung, I. Lambadaris, and N. Seddigh, "A Hybrid Technique Using PCA and Wavelets in Network Traffic Anomaly Detection," *International Journal of Mobile Computing and Multimedia Communications*, vol. 6, no. 1, pp. 17–53, Jan. 2014.

[42] J. Song, H. Takakura, Y. Okabe, D. Inoue, M. Fto, and K. Nakao, "A comparative study of unsupervised anomaly detection techniques using honeypot data," *IEICE Transactions on Information and Systems*, vol. E93-D, no. 9, pp. 2544–2554, 2010.

[43] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, and K. Nakao, "Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation," in *Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security - BADGERS '11*, 2011, pp. 29–36.

[44] S. Lee, D. Kim, and J. Park, "A survey and taxonomy of lightweight intrusion detection systems," *Journal of Internet Services and Information Security*, vol. 2, no. 1/2, pp. 119–13, 2012.

[45] C. H. Tsang, S. Kwong, and H. Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," *Pattern Recognition*, vol. 40, no. 9, pp. 2373–2391, Sep. 2007.

[46] T. S. Chou, K. K. Yen, and J. Luo, "Network intrusion detection design using feature selection of soft computing paradigms," *International journal of computational Intelligence*, vol. 4, no. 3, pp. 196–208, 2008.

[47] A. H. Sung and S. Mukkamala, "Identifying Important Features for Intrusion Detection Using Support Vector Machines and Neural Networks," in *Proceedings of the Symposium on Applications and the Internet (SAINT'03)*, 2003, vol. 1, no. 1, pp. 209–216.

[48] A. Hofmann, T. Horeis, and B. Sick, "Feature selection for intrusion detection: an evolutionary wrapper approach," in *IEEE International Joint Conference on Neural Networks*, 2004, vol. 2, pp. 1563–1568.

[49] Y. Li, J. L. Wang, Z. H. Tian, T. B. Lu, and C. Young, "Building lightweight intrusion detection system using wrapper-based feature selection mechanisms," *Computers & Security*, vol. 28, no. 6, pp. 466–475, Sep. 2009.

[50] S. Chebrolu, A. Abraham, and J. P. Thomas, "Feature deduction and ensemble design of intrusion detection systems," *Computers & Security*, vol. 24, no. 4, pp. 295–307, Jun. 2005.

[51] F. Amiri, M. R. Yousefi, C. Lucas, A. Shakery, and N. Yazdani, "Mutual information-based feature selection for intrusion detection systems," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1184–1199, Jul. 2011.
[52] A. A. Olusola, A. S. Oladele, and D. O. Abosede, "Analysis of KDD'99 Intrusion Detection Dataset for Selection of Relevance Features," in *Proceedings of the World Congress on Engineering and Computer Science WCECS 2010*, 2010, vol. 1.
[53] S. Zargari and D. Voorhis, "Feature Selection in the Corrected KDD-dataset," in *Third International Conference on Emerging Intelligent Data and Web Technologies*, 2012, pp. 174–180.
[54] Machine Learning Group (ac.nz), "WEKA software," *at the University of Waikato*. [Online]. Available: http://www.cs.waikato.ac.nz/ml/index.html.
[55] H. Debar, M. Dacier, and A. Wespi, "Towards a taxonomy of intrusion-detection systems," *Computer Networks*, vol. 31, no. 8, pp. 805–822, Apr. 1999.
[56] S. Kumar and E. H. Spafford, "A Pattern Matching Model for Misuse Intrusion Detection," Purdue University, Technical Report, 1994.
[57] M. Sabhnani and G. Serpen, "Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context," in *Proceedings of International Conference on Machine Learning: Models, Technologies, and Applications (MLMTA)*, 2003, pp. 209–215.
[58] C. Kreibich and J. Crowcroft, "Honeycomb: creating intrusion detection signatures using honeypots," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 51–56, 2004.
[59] S. Peddabachigari, A. Abraham, and J. Thomas, "Intrusion Detection Systems Using Decision Trees and Support Vector Machines," *International Journal of Applied Science and Computations*, vol. 11, no. 3, pp. 118–134, 2004.
[60] S. Mukkamala, A. H. Sung, and A. Abraham, "Intrusion detection using an ensemble of intelligent paradigms," *Journal of Network and Computer Applications*, vol. 28, no. 2, pp. 167–182, Apr. 2005.
[61] Y. Bouzida and F. Cuppens, "Neural networks vs. decision trees for intrusion detection," in *IEEE/IST Workshop on Monitoring, Attack Detection and Mitigation (MonAM)*, 2006, pp. 81–88.
[62] M.-Y. Su, G.-J. Yu, and C.-Y. Lin, "A real-time network intrusion detection system for large-scale attacks based on an incremental mining approach," *Computers & Security*, vol. 28, no. 5, pp. 301–309, Jul. 2009.
[63] E. J. Palomo, E. Domínguez, R. M. Luque, and J. Muñoz, "An Intrusion Detection System Based on Hierarchical," in *Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems CISIS'08*, 2009, pp. 139–146.
[64] S. J. Horng, M. Y. Su, Y. H. Chen, T. W. Kao, R. J. Chen, J. L. Lai, and C. D. Perkasa, "A novel intrusion detection system based on hierarchical clustering and support vector machines," *Expert Systems with Applications*, vol. 38, no. 1, pp. 306–313, Jan. 2011.
[65] C. Bae, W.-C. Yeh, M. A. M. Shukran, Y. Y. Chung, and T.-J. Hsieh, "A Novel Anomaly-Network Intrusion Detection System Using ABC Algorithms," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 12, pp. 8231–8248, 2012.
[66] N. Wattanapongsakorn, S. Srakaew, E. Wonghirunsombat, C. Sribavonmongkol, T. Junhom, P. Jongsubsook, and C. Charnsripinyo, "A Practical Network-Based Intrusion Detection and Prevention System," in *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, 2012, pp. 209–214.
[67] Y. Waizumi, Y. Sato, and Y. Nemoto, "A Network-Based Anomaly Detection System

Based on Three Different Network Traffic Characteristics," *Journal of Communication and Computer*, vol. 9, no. 7, pp. 805–812, 2012.
[68] W. Lee, S. J. Stolfo, P. K. Chan, E. Eskin, W. Fan, M. Miller, S. Hershkop, and J. Zhang, "Real Time Data Mining-based Intrusion Detection," in *DARPA Information Survivability Conference & Exposition II, DISCEX'01*, 2001, vol. 1, pp. 89–100.
[69] S. Janakiraman, "ACO based Distributed Intrusion Detection System," *JDCTA: International Journal of Digital Content Technology and its Applications*, vol. 3, no. 1, pp. 66–72, Mar. 2009.
[70] N. B. Anuar, H. Sallehudin, A. Gani, and O. Zakari, "Identifying False Alarm for Network Intrusion Detection System using Hybrid Data Mining a and Decision Tree," *Malaysian journal of computer science*, vol. 21, no. 2, pp. 101–115, 2008.
[71] A. Tajbakhsh, M. Rahmati, and A. Mirzaei, "Intrusion detection using fuzzy association rules," *Applied Soft Computing*, vol. 9, no. 2, pp. 462–469, Mar. 2009.
[72] M. Govindarajan and R. Chandrasekaran, "Intrusion detection using neural based hybrid classification methods," *Computer Networks*, vol. 55, no. 8, pp. 1662–1671, Jun. 2011.
[73] C. Xiang, M. Y. Chong, and H. L. Zhu, "Design of multiple-level tree classifiers for intrusion detection system," in *IEEE Conference on Cybernetics and Intelligent Systems*, 2004, vol. 2, pp. 873–878.
[74] M. A. Aydın, A. H. Zaim, and K. G. Ceylan, "A hybrid intrusion detection system design for computer network security," *Computers & Electrical Engineering*, vol. 35, no. 3, pp. 517–526, May 2009.
[75] S. Kumar, "Survey of Current Network Intrusion Detection Techniques," 2007. [Online]. Available: http://www.cse.wustl.edu/~jain/cse571-07/ftp/ids/. [Accessed: 04-Jul-2013].
[76] A. Patcha and J. Park, "An overview of anomaly detection techniques : Existing solutions and latest technological trends," *Computer Networks*, vol. 51, pp. 3448–3470, 2007.
[77] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Computers & Security*, vol. 28, no. 1, pp. 18–28, Feb. 2009.
[78] S. Axelsson, "The base-rate fallacy and its implications for the difficulty of intrusion detection," in *Proceedings of the 6th ACM Conference on Computer and Communications Security (CCS '99)*, 1999, pp. 1–7.
[79] S. Axelsson, "Intrusion detection systems: A survey and taxonomy," Chalmers University of Technology, Göteborg, Sweden, Technical Report, 2000.
[80] R. Kohavi and F. Provost, "Glossary of Terms," *Special Issue on Applications of Machine Learning and the Knowledge Discovery Process*, vol. 30, no. 2/3, pp. 271–274, 1998.
[81] B. Karlık, "Machine Learning Algorithms for Characterization of EMG Signals," *International Journal of Information and Electronics Engineering*, vol. 4, no. 3, pp. 189–194, 2014.
[82] B. Karlık, "Artificial neural networks," Mevlana University, Konya, Lecture Notes, 2012.
[83] B. Karlik, M. O. Tokhi, and M. Alci, "A fuzzy clustering neural network architecture for multifunction upper-limb prosthesis.," *IEEE transactions on bio-medical engineering*, vol. 50, no. 11, pp. 1255–61, Nov. 2003.
[84] B. Karlik, "Differentiating Type of Muscle Movement via AR Modeling and Neural Network Classification," *Turkish Journal of Electrical Engineering*, vol. 7, no. 1–3, pp. 45–52, 1999.

[85] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.
[86] H. Hamilton, E. Gurak, L. Findlater, and W. Olive, "Decision Trees," *University of Regina*, 2014. [Online]. Available: http://dms.irb.hr/tutorial/tut_dtrees.php.
[87] T. M. Mitchell, "Decision Tree Learning," in *Machine Learning*, McGraw Hill, 1997, pp. 52–80.
[88] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, Mar. 1986.
[89] D. Gamberger, T. Šmuc, and I. Marić, "Data mining server (DMS)," *Rudjer Boskovic Institute*, 2014. [Online]. Available: http://dms.irb.hr/tutorial/tut_dtrees.php.
[90] G. Xu, Y. Zhang, and L. Li, *Web Mining and Social Networking: Techniques and Applications*. Springer Science & Business Media, 2010.
[91] Ma. J. L. Orr and B. Place, "Introduction to radial basis function networks," Center for Cognitive Science, University of Edinburgh, Technical Report, 1996.
[92] C. McCormick, "Radial Basis Function Network (RBFN) Tutorial," 2013. [Online]. Available: http://chrisjmccormick.wordpress.com/2013/08/15/radial-basis-function-network-rbfn-tutorial/.
[93] Z. Gavrilov, "SVM Tutorial," MIT, Cambridge, MA., 2012.
[94] C. Haruechaiyasak, "A Tutorial on Naive Bayes Classification," King Mongkut's University of Technology, North Bangkok, Online Tutorial, 2008.
[95] U. Maulik and S. Bandyopadhyay, "Genetic algorithm-based clustering technique," *Pattern Recognition*, vol. 33, no. 9, pp. 1455–1465, Sep. 2000.
[96] H.-T. Lin, Y.-Y. Lin, and J.-W. Chiang, "Genetic-based real-time fast-flux service networks detection," *Computer Networks*, vol. 57, no. 2, pp. 501–513, Feb. 2013.
[97] M. Srinivas and L. M. Patnaik, "Genetic algorithms: A survey," *Computer*, vol. 27, no. 6, pp. 17–26, Jun. 1994.
[98] B. Koçer, "New Approaches in Transfer Learning," PhD Thesis, Selçuk University, 2012.
[99] D. Hermawanto, "Genetic algorithm for solving simple mathematical equality problem," Indonesian Institute of Sciences (LIPI), Online Tutorial, 2013.
[100] B. Koçer and A. Arslan, "Genetic transfer learning," *Expert Systems with Applications*, vol. 37, no. 10, pp. 6997–7002, Oct. 2010.
[101] E. Baralis, S. Chiusano, and P. Garza, "A lazy approach to associative classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 2, pp. 156–171, Feb. 2008.
[102] H. Daum and D. Marcu, "Domain Adaptation for Statistical Classifiers," *Journal of Artificial Intelligence Research*, vol. 26, pp. 101–126, 2006.
[103] S. Gou, Y. Wang, L. Jiao, J. Feng, and Y. Yao, "Distributed transfer network learning based intrusion detection," in *IEEE International Symposium on Parallel and Distributed Processing with Applications*, 2009, pp. 511–515.
[104] A. J. Storkey, "When training and test sets are different: Characterising learning transfer," Institute of Adaptive and Neural Computation, School of Informatics, University of Edinburg, Report, 2013.
[105] A. Arnold, R. Nallapati, and W. W. Cohen, "A Comparative Study of Methods for Transductive Transfer Learning," in *The Seventh IEEE International Conference on Data Mining Workshops (ICDMW)*, 2007, pp. 77–82.
[106] S. Kahramanli, M. Hacibeyoglu, and A. Arslan, "A Boolean function approach to feature selection in consistent decision information systems," *Expert Systems with Applications*, vol. 38, no. 7, pp. 8229–8239, Jul. 2011.
[107] A. S. Raut and K. R. Singh, "Feature Selection for Anomaly-Based Intrusion Detection

using Rough Set theory," in *International conference on industrial Automation and computing (ICIAC)*, 2014, no. April.
[108] V. Rampure and A. Tiwari, "A Rough Set Based Feature Selection on KDD CUP 99 Data Set," *International Journal of Database Theory and Application*, vol. 8, no. 1, pp. 149–156, 2015.
[109] A. Skowron and C. Rauszer, "The Discernibility Matrices and Functions in Information Systems," in *Intelligent Decision Support*, 1st ed., vol. 11, R. Słowiński, Ed. Dordrecht: Springer Netherlands, 1992, pp. 331–362.
[110] R. Jensen and Q. Shen, "Rough Set-based Feature Selection : A Review," in *Rough Computing: Theories, Technologies and Applications*, 2007, pp. 70–107.
[111] S. Kahramanli, M. Hacibeyoglu, and A. Arslan, "Attribute Reduction By Partitioning The Minimized Discernibility Function," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 5, pp. 2167–2186, 2011.
[112] K. Kendall and A. C. Smith, "A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems," Master Thesis, Massachusetts Institute of Technology, 1999.
[113] J. Erickson, *Hacking: The art of exploitation*, 2nd ed. No Starch Press, 2008.
[114] T. Aulakh, "Intrusion detection and prevention system : CGI," Master Thesis, San Jose State University, 2009.
[115] M. T. M. Win and K. T. Khaing, "Detection and Classification of Attacks in Unauthorized Accesses," in *International Conference on Advances in Engineering and Technology (ICAET'2014)*, 2014, pp. 345–349.
[116] MIT Lincoln Labs, "DARPA Intrusion Detection Evaluation." [Online]. Available: http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/docs/attackDB.html. [Accessed: 10-Dec-2012].
[117] P. Tang, R. Jiang, and M. Zhao, "Feature selection and design olintrusion detection system based on k-means and triangle area support vector machine," in *Second International Conference on Future Networks ICFN'10*, 2010, pp. 144–148.
[118] J. Bi, K. Zhang, and X. Cheng, "Intrusion Detection Based on RBF Neural Network," *2009 International Symposium on Information Engineering and Electronic Commerce*, pp. 357–360, May 2009.
[119] S. Devaraju and S. Ramakrishnan, "Performance Comparison for Intrusion Detection System using Neural Network with KDD Dataset," *ICTACT Journal on Soft Computing*, vol. 04, no. 03, pp. 743–752, 2014.
[120] L. Koc, T. a. Mazzuchi, and S. Sarkani, "A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier," *Expert Systems with Applications*, vol. 39, no. 18, pp. 13492–13500, 2012.
[121] U. M. Fayyad and K. B. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning," in *UAI*, 1993, pp. 1022–1027.
[122] B. Pfahringer, "Winning the KDD99 classification cup," *ACM SIGKDD Explorations Newsletter*, vol. 1, no. 2, p. 65, Jan. 2000.
[123] I. Levin, "KDD-99 classifier learning contest LLSoft's results overview," *ACM SIGKDD Explorations Newsletter*, vol. 1, no. 2, p. 67, Jan. 2000.
[124] H. Sallay, A. Ammar, M. Ben Saad, and S. Bourouis, "A real time adaptive intrusion detection alert classifier for high speed networks," in *IEEE 12th International Symposium on Network Computing and Applications (NCA)*, 2013, pp. 73–80.
[125] W. Yassin, N. I. Udzir, and Z. Muda, "Anomaly-Based Intrusion Detection Through K-Means Clustering and Naives Bayes Classification," in *Proceedings of the 4th International Conference on Computing and Informatics (ICOCI)*, 2013, no. 049, pp. 298–303.

[126] Z. Tan, "Detection of Denial-of-Service Attacks Based on Computer Vision Techniques," *IEEE Transactions on Computers*, no. December, pp. 1–14, 2014.
[127] MAWI Working Group, "WIDE backbone traffic traces," 2006. [Online]. Available: http://mawi.wide.ad.jp/mawi/. [Accessed: 24-Nov-2015].
[128] MOME Cluster of Europian Project, "MOME Database," 2005. [Online]. Available: https://www.ist-mome.org/database/MeasurementData/index8db7.html. [Accessed: 24-Nov-2015].
[129] J. Hu, "The ADFA Intrusion Detection Datasets," 2012. [Online]. Available: https://www.unsw.adfa.edu.au/australian-centre-for-cyber-security/cybersecurity/ADFA-IDS-Datasets/. [Accessed: 24-Nov-2015].
[130] WAND Network Research Group, "Waikato Internet Traffic Storage (WITS)," 2013. [Online]. Available: http://wand.net.nz/wits/. [Accessed: 24-Nov-2015].

APPENDIX A

The complete list of 930 candidate feature subsets obtained with the DFBFS using the KDD Cup 10% dataset in Section 6.3.2.3.

Subset ID	**Features**
Subset ID 1	A1,A2,A4,A5,A6,A12,A13,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 2	A1,A2,A4,A5,A6,A12,A13,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 3	A1,A2,A4,A5,A6,A12,A13,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 4	A1,A2,A4,A5,A6,A12,A14,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 5	A1,A2,A4,A5,A6,A12,A14,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 6	A1,A2,A4,A5,A6,A12,A14,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 7	A1,A2,A4,A5,A6,A12,A16,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 8	A1,A2,A4,A5,A6,A12,A16,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 9	A1,A2,A4,A5,A6,A12,A16,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 10	A1,A2,A4,A5,A6,A12,A18,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 11	A1,A2,A4,A5,A6,A12,A18,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 12	A1,A2,A4,A5,A6,A12,A18,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 13	A1,A3,A4,A5,A6,A10,A12,A23,A24,A25,A33,A34,A35,A37,A39
Subset ID 14	A1,A3,A4,A5,A6,A10,A12,A23,A24,A27,A33,A34,A35,A37,A39
Subset ID 15	A1,A3,A4,A5,A6,A10,A12,A23,A24,A30,A33,A34,A35,A37,A39
Subset ID 16	A1,A3,A4,A5,A6,A10,A12,A23,A24,A33,A35,A36,A37,A39
Subset ID 17	A1,A3,A4,A5,A6,A10,A12,A23,A24,A33,A35,A37,A38,A39,A40
Subset ID 18	A1,A3,A4,A5,A6,A10,A12,A23,A25,A31,A33,A34,A35,A37,A39
Subset ID 19	A1,A3,A4,A5,A6,A10,A12,A23,A27,A31,A33,A34,A35,A37,A39
Subset ID 20	A1,A3,A4,A5,A6,A10,A12,A23,A30,A31,A33,A34,A35,A37,A39
Subset ID 21	A1,A3,A4,A5,A6,A10,A12,A23,A31,A33,A35,A36,A37,A39
Subset ID 22	A1,A3,A4,A5,A6,A10,A12,A23,A31,A33,A35,A37,A38,A39,A40
Subset ID 23	A1,A3,A4,A5,A6,A10,A12,A24,A27,A33,A35,A36,A37,A39
Subset ID 24	A1,A3,A4,A5,A6,A10,A12,A24,A27,A33,A35,A37,A38,A39,A40
Subset ID 25	A1,A3,A4,A5,A6,A10,A12,A24,A29,A30,A33,A34,A35,A37,A39
Subset ID 26	A1,A3,A4,A5,A6,A10,A12,A24,A29,A33,A35,A36,A37,A39
Subset ID 27	A1,A3,A4,A5,A6,A10,A12,A24,A29,A33,A35,A37,A38,A39,A40
Subset ID 28	A1,A3,A4,A5,A6,A10,A12,A24,A30,A33,A35,A36,A37,A39
Subset ID 29	A1,A3,A4,A5,A6,A12,A13,A23,A24,A25,A33,A34,A35,A37,A39
Subset ID 30	A1,A3,A4,A5,A6,A12,A13,A23,A24,A27,A33,A34,A35,A37,A39
Subset ID 31	A1,A3,A4,A5,A6,A12,A13,A23,A24,A30,A33,A34,A35,A37,A39
Subset ID 32	A1,A3,A4,A5,A6,A12,A13,A23,A24,A33,A35,A36,A37,A39
Subset ID 33	A1,A3,A4,A5,A6,A12,A13,A23,A24,A33,A35,A37,A38,A39,A40
Subset ID 34	A1,A3,A4,A5,A6,A12,A13,A23,A25,A31,A33,A34,A35,A37,A39
Subset ID 35	A1,A3,A4,A5,A6,A12,A13,A23,A27,A31,A33,A34,A35,A37,A39
Subset ID 36	A1,A3,A4,A5,A6,A12,A13,A23,A30,A31,A33,A34,A35,A37,A39
Subset ID 37	A1,A3,A4,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A39
Subset ID 38	A1,A3,A4,A5,A6,A12,A13,A23,A31,A33,A35,A37,A38,A39,A40

Subset ID 39	A1,A3,A4,A5,A6,A12,A13,A24,A27,A33,A35,A36,A37,A39
Subset ID 40	A1,A3,A4,A5,A6,A12,A13,A24,A27,A33,A35,A37,A38,A39,A40
Subset ID 41	A1,A3,A4,A5,A6,A12,A13,A24,A29,A30,A33,A34,A35,A37,A39
Subset ID 42	A1,A3,A4,A5,A6,A12,A13,A24,A29,A33,A35,A36,A37,A39
Subset ID 43	A1,A3,A4,A5,A6,A12,A13,A24,A29,A33,A35,A37,A38,A39,A40
Subset ID 44	A1,A3,A4,A5,A6,A12,A13,A24,A30,A33,A35,A36,A37,A39
Subset ID 45	A1,A3,A4,A5,A6,A12,A14,A23,A24,A25,A33,A34,A35,A37,A39
Subset ID 46	A1,A3,A4,A5,A6,A12,A14,A23,A24,A27,A33,A34,A35,A37,A39
Subset ID 47	A1,A3,A4,A5,A6,A12,A14,A23,A24,A30,A33,A34,A35,A37,A39
Subset ID 48	A1,A3,A4,A5,A6,A12,A14,A23,A24,A33,A35,A36,A37,A39
Subset ID 49	A1,A3,A4,A5,A6,A12,A14,A23,A24,A33,A35,A37,A38,A39,A40
Subset ID 50	A1,A3,A4,A5,A6,A12,A14,A23,A25,A31,A33,A34,A35,A37,A39
Subset ID 51	A1,A3,A4,A5,A6,A12,A14,A23,A27,A31,A33,A34,A35,A37,A39
Subset ID 52	A1,A3,A4,A5,A6,A12,A14,A23,A30,A31,A33,A34,A35,A37,A39
Subset ID 53	A1,A3,A4,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A39
Subset ID 54	A1,A3,A4,A5,A6,A12,A14,A23,A31,A33,A35,A37,A38,A39,A40
Subset ID 55	A1,A3,A4,A5,A6,A12,A14,A24,A27,A33,A35,A36,A37,A39
Subset ID 56	A1,A3,A4,A5,A6,A12,A14,A24,A27,A33,A35,A37,A38,A39,A40
Subset ID 57	A1,A3,A4,A5,A6,A12,A14,A24,A29,A30,A33,A34,A35,A37,A39
Subset ID 58	A1,A3,A4,A5,A6,A12,A14,A24,A29,A33,A35,A36,A37,A39
Subset ID 59	A1,A3,A4,A5,A6,A12,A14,A24,A29,A33,A35,A37,A38,A39,A40
Subset ID 60	A1,A3,A4,A5,A6,A12,A14,A24,A30,A33,A35,A36,A37,A39
Subset ID 61	A1,A3,A4,A5,A6,A12,A16,A23,A24,A25,A33,A34,A35,A37,A39
Subset ID 62	A1,A3,A4,A5,A6,A12,A16,A23,A24,A27,A33,A34,A35,A37,A39
Subset ID 63	A1,A3,A4,A5,A6,A12,A16,A23,A24,A30,A33,A34,A35,A37,A39
Subset ID 64	A1,A3,A4,A5,A6,A12,A16,A23,A24,A33,A35,A36,A37,A39
Subset ID 65	A1,A3,A4,A5,A6,A12,A16,A23,A24,A33,A35,A37,A38,A39,A40
Subset ID 66	A1,A3,A4,A5,A6,A12,A16,A23,A25,A31,A33,A34,A35,A37,A39
Subset ID 67	A1,A3,A4,A5,A6,A12,A16,A23,A27,A31,A33,A34,A35,A37,A39
Subset ID 68	A1,A3,A4,A5,A6,A12,A16,A23,A30,A31,A33,A34,A35,A37,A39
Subset ID 69	A1,A3,A4,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A39
Subset ID 70	A1,A3,A4,A5,A6,A12,A16,A23,A31,A33,A35,A37,A38,A39,A40
Subset ID 71	A1,A3,A4,A5,A6,A12,A16,A24,A27,A33,A35,A36,A37,A39
Subset ID 72	A1,A3,A4,A5,A6,A12,A16,A24,A27,A33,A35,A37,A38,A39,A40
Subset ID 73	A1,A3,A4,A5,A6,A12,A16,A24,A29,A30,A33,A34,A35,A37,A39
Subset ID 74	A1,A3,A4,A5,A6,A12,A16,A24,A29,A33,A35,A36,A37,A39
Subset ID 75	A1,A3,A4,A5,A6,A12,A16,A24,A29,A33,A35,A37,A38,A39,A40
Subset ID 76	A1,A3,A4,A5,A6,A12,A16,A24,A30,A33,A35,A36,A37,A39
Subset ID 77	A1,A3,A4,A5,A6,A12,A17,A23,A24,A25,A33,A34,A35,A37,A39
Subset ID 78	A1,A3,A4,A5,A6,A12,A17,A23,A24,A27,A33,A34,A35,A37,A39
Subset ID 79	A1,A3,A4,A5,A6,A12,A17,A23,A24,A30,A33,A34,A35,A37,A39
Subset ID 80	A1,A3,A4,A5,A6,A12,A17,A23,A24,A33,A35,A36,A37,A39
Subset ID 81	A1,A3,A4,A5,A6,A12,A17,A23,A24,A33,A35,A37,A38,A39,A40
Subset ID 82	A1,A3,A4,A5,A6,A12,A17,A23,A25,A31,A33,A34,A35,A37,A39
Subset ID 83	A1,A3,A4,A5,A6,A12,A17,A23,A27,A31,A33,A34,A35,A37,A39

Subset ID 84	A1,A3,A4,A5,A6,A12,A17,A23,A30,A31,A33,A34,A35,A37,A39
Subset ID 85	A1,A3,A4,A5,A6,A12,A17,A23,A31,A33,A35,A36,A37,A39
Subset ID 86	A1,A3,A4,A5,A6,A12,A17,A23,A31,A33,A35,A37,A38,A39,A40
Subset ID 87	A1,A3,A4,A5,A6,A12,A17,A24,A27,A33,A35,A36,A37,A39
Subset ID 88	A1,A3,A4,A5,A6,A12,A17,A24,A27,A33,A35,A37,A38,A39,A40
Subset ID 89	A1,A3,A4,A5,A6,A12,A17,A24,A29,A30,A33,A34,A35,A37,A39
Subset ID 90	A1,A3,A4,A5,A6,A12,A17,A24,A29,A33,A35,A36,A37,A39
Subset ID 91	A1,A3,A4,A5,A6,A12,A17,A24,A29,A33,A35,A37,A38,A39,A40
Subset ID 92	A1,A3,A4,A5,A6,A12,A17,A24,A30,A33,A35,A36,A37,A39
Subset ID 93	A1,A3,A4,A5,A6,A12,A18,A23,A24,A25,A33,A34,A35,A37,A39
Subset ID 94	A1,A3,A4,A5,A6,A12,A18,A23,A24,A27,A33,A34,A35,A37,A39
Subset ID 95	A1,A3,A4,A5,A6,A12,A18,A23,A24,A30,A33,A34,A35,A37,A39
Subset ID 96	A1,A3,A4,A5,A6,A12,A18,A23,A24,A33,A35,A36,A37,A39
Subset ID 97	A1,A3,A4,A5,A6,A12,A18,A23,A24,A33,A35,A37,A38,A39,A40
Subset ID 98	A1,A3,A4,A5,A6,A12,A18,A23,A25,A31,A33,A34,A35,A37,A39
Subset ID 99	A1,A3,A4,A5,A6,A12,A18,A23,A27,A31,A33,A34,A35,A37,A39
Subset ID 100	A1,A3,A4,A5,A6,A12,A18,A23,A30,A31,A33,A34,A35,A37,A39
Subset ID 101	A1,A3,A4,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A39
Subset ID 102	A1,A3,A4,A5,A6,A12,A18,A23,A31,A33,A35,A37,A38,A39,A40
Subset ID 103	A1,A3,A4,A5,A6,A12,A18,A24,A27,A33,A35,A36,A37,A39
Subset ID 104	A1,A3,A4,A5,A6,A12,A18,A24,A27,A33,A35,A37,A38,A39,A40
Subset ID 105	A1,A3,A4,A5,A6,A12,A18,A24,A29,A30,A33,A34,A35,A37,A39
Subset ID 106	A1,A3,A4,A5,A6,A12,A18,A24,A29,A33,A35,A36,A37,A39
Subset ID 107	A1,A3,A4,A5,A6,A12,A18,A24,A29,A33,A35,A37,A38,A39,A40
Subset ID 108	A1,A3,A4,A5,A6,A12,A18,A24,A30,A33,A35,A36,A37,A39
Subset ID 109	A1,A3,A5,A6,A10,A12,A23,A24,A33,A34,A35,A37,A38,A39
Subset ID 110	A1,A3,A5,A6,A10,A12,A23,A24,A33,A35,A36,A37,A38,A39
Subset ID 111	A1,A3,A5,A6,A10,A12,A23,A24,A33,A36,A37,A39,A40
Subset ID 112	A1,A3,A5,A6,A10,A12,A23,A31,A33,A34,A35,A37,A38,A39
Subset ID 113	A1,A3,A5,A6,A10,A12,A23,A31,A33,A35,A36,A37,A38,A39
Subset ID 114	A1,A3,A5,A6,A10,A12,A23,A31,A33,A36,A37,A39,A40
Subset ID 115	A1,A3,A5,A6,A10,A12,A24,A25,A29,A33,A36,A37,A39,A40
Subset ID 116	A1,A3,A5,A6,A10,A12,A24,A25,A30,A33,A36,A37,A39,A40
Subset ID 117	A1,A3,A5,A6,A10,A12,A24,A27,A33,A35,A36,A37,A38,A39
Subset ID 118	A1,A3,A5,A6,A10,A12,A24,A29,A30,A33,A34,A35,A37,A38,A39
Subset ID 119	A1,A3,A5,A6,A10,A12,A24,A29,A31,A33,A36,A37,A39,A40
Subset ID 120	A1,A3,A5,A6,A10,A12,A24,A29,A33,A35,A36,A37,A38,A39
Subset ID 121	A1,A3,A5,A6,A10,A12,A24,A29,A33,A36,A37,A38,A39,A40
Subset ID 122	A1,A3,A5,A6,A10,A12,A24,A30,A31,A33,A36,A37,A39,A40
Subset ID 123	A1,A3,A5,A6,A10,A12,A24,A30,A33,A35,A36,A37,A38,A39
Subset ID 124	A1,A3,A5,A6,A10,A12,A24,A30,A33,A36,A37,A38,A39,A40
Subset ID 125	A1,A3,A5,A6,A10,A12,A24,A33,A35,A36,A37,A39,A40
Subset ID 126	A1,A3,A5,A6,A12,A13,A23,A24,A33,A34,A35,A37,A38,A39
Subset ID 127	A1,A3,A5,A6,A12,A13,A23,A24,A33,A35,A36,A37,A38,A39
Subset ID 128	A1,A3,A5,A6,A12,A13,A23,A24,A33,A36,A37,A39,A40

Subset ID 129	A1,A3,A5,A6,A12,A13,A23,A31,A33,A34,A35,A37,A38,A39
Subset ID 130	A1,A3,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A38,A39
Subset ID 131	A1,A3,A5,A6,A12,A13,A23,A31,A33,A36,A37,A39,A40
Subset ID 132	A1,A3,A5,A6,A12,A13,A24,A25,A29,A33,A36,A37,A39,A40
Subset ID 133	A1,A3,A5,A6,A12,A13,A24,A25,A30,A33,A36,A37,A39,A40
Subset ID 134	A1,A3,A5,A6,A12,A13,A24,A27,A33,A35,A36,A37,A38,A39
Subset ID 135	A1,A3,A5,A6,A12,A13,A24,A29,A30,A33,A34,A35,A37,A38,A39
Subset ID 136	A1,A3,A5,A6,A12,A13,A24,A29,A31,A33,A36,A37,A39,A40
Subset ID 137	A1,A3,A5,A6,A12,A13,A24,A29,A33,A35,A36,A37,A38,A39
Subset ID 138	A1,A3,A5,A6,A12,A13,A24,A29,A33,A36,A37,A38,A39,A40
Subset ID 139	A1,A3,A5,A6,A12,A13,A24,A30,A31,A33,A36,A37,A39,A40
Subset ID 140	A1,A3,A5,A6,A12,A13,A24,A30,A33,A35,A36,A37,A38,A39
Subset ID 141	A1,A3,A5,A6,A12,A13,A24,A30,A33,A36,A37,A38,A39,A40
Subset ID 142	A1,A3,A5,A6,A12,A13,A24,A33,A35,A36,A37,A39,A40
Subset ID 143	A1,A3,A5,A6,A12,A14,A23,A24,A33,A34,A35,A37,A38,A39
Subset ID 144	A1,A3,A5,A6,A12,A14,A23,A24,A33,A35,A36,A37,A38,A39
Subset ID 145	A1,A3,A5,A6,A12,A14,A23,A24,A33,A36,A37,A39,A40
Subset ID 146	A1,A3,A5,A6,A12,A14,A23,A31,A33,A34,A35,A37,A38,A39
Subset ID 147	A1,A3,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A38,A39
Subset ID 148	A1,A3,A5,A6,A12,A14,A23,A31,A33,A36,A37,A39,A40
Subset ID 149	A1,A3,A5,A6,A12,A14,A24,A25,A29,A33,A36,A37,A39,A40
Subset ID 150	A1,A3,A5,A6,A12,A14,A24,A25,A30,A33,A36,A37,A39,A40
Subset ID 151	A1,A3,A5,A6,A12,A14,A24,A27,A33,A35,A36,A37,A38,A39
Subset ID 152	A1,A3,A5,A6,A12,A14,A24,A29,A30,A33,A34,A35,A37,A38,A39
Subset ID 153	A1,A3,A5,A6,A12,A14,A24,A29,A31,A33,A36,A37,A39,A40
Subset ID 154	A1,A3,A5,A6,A12,A14,A24,A29,A33,A35,A36,A37,A38,A39
Subset ID 155	A1,A3,A5,A6,A12,A14,A24,A29,A33,A36,A37,A38,A39,A40
Subset ID 156	A1,A3,A5,A6,A12,A14,A24,A30,A31,A33,A36,A37,A39,A40
Subset ID 157	A1,A3,A5,A6,A12,A14,A24,A30,A33,A35,A36,A37,A38,A39
Subset ID 158	A1,A3,A5,A6,A12,A14,A24,A30,A33,A36,A37,A38,A39,A40
Subset ID 159	A1,A3,A5,A6,A12,A14,A24,A33,A35,A36,A37,A39,A40
Subset ID 160	A1,A3,A5,A6,A12,A16,A23,A24,A33,A34,A35,A37,A38,A39
Subset ID 161	A1,A3,A5,A6,A12,A16,A23,A24,A33,A35,A36,A37,A38,A39
Subset ID 162	A1,A3,A5,A6,A12,A16,A23,A24,A33,A36,A37,A39,A40
Subset ID 163	A1,A3,A5,A6,A12,A16,A23,A31,A33,A34,A35,A37,A38,A39
Subset ID 164	A1,A3,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A38,A39
Subset ID 165	A1,A3,A5,A6,A12,A16,A23,A31,A33,A36,A37,A39,A40
Subset ID 166	A1,A3,A5,A6,A12,A16,A24,A25,A29,A33,A36,A37,A39,A40
Subset ID 167	A1,A3,A5,A6,A12,A16,A24,A25,A30,A33,A36,A37,A39,A40
Subset ID 168	A1,A3,A5,A6,A12,A16,A24,A27,A33,A35,A36,A37,A38,A39
Subset ID 169	A1,A3,A5,A6,A12,A16,A24,A29,A30,A33,A34,A35,A37,A38,A39
Subset ID 170	A1,A3,A5,A6,A12,A16,A24,A29,A31,A33,A36,A37,A39,A40
Subset ID 171	A1,A3,A5,A6,A12,A16,A24,A29,A33,A35,A36,A37,A38,A39
Subset ID 172	A1,A3,A5,A6,A12,A16,A24,A29,A33,A36,A37,A38,A39,A40
Subset ID 173	A1,A3,A5,A6,A12,A16,A24,A30,A31,A33,A36,A37,A39,A40

Subset ID 174	A1,A3,A5,A6,A12,A16,A24,A30,A33,A35,A36,A37,A38,A39
Subset ID 175	A1,A3,A5,A6,A12,A16,A24,A30,A33,A36,A37,A38,A39,A40
Subset ID 176	A1,A3,A5,A6,A12,A16,A24,A33,A35,A36,A37,A39,A40
Subset ID 177	A1,A3,A5,A6,A12,A17,A23,A24,A33,A34,A35,A37,A38,A39
Subset ID 178	A1,A3,A5,A6,A12,A17,A23,A24,A33,A35,A36,A37,A38,A39
Subset ID 179	A1,A3,A5,A6,A12,A17,A23,A24,A33,A36,A37,A39,A40
Subset ID 180	A1,A3,A5,A6,A12,A17,A23,A31,A33,A34,A35,A37,A38,A39
Subset ID 181	A1,A3,A5,A6,A12,A17,A23,A31,A33,A35,A36,A37,A38,A39
Subset ID 182	A1,A3,A5,A6,A12,A17,A23,A31,A33,A36,A37,A39,A40
Subset ID 183	A1,A3,A5,A6,A12,A17,A24,A25,A29,A33,A36,A37,A39,A40
Subset ID 184	A1,A3,A5,A6,A12,A17,A24,A25,A30,A33,A36,A37,A39,A40
Subset ID 185	A1,A3,A5,A6,A12,A17,A24,A27,A33,A35,A36,A37,A38,A39
Subset ID 186	A1,A3,A5,A6,A12,A17,A24,A29,A30,A33,A34,A35,A37,A38,A39
Subset ID 187	A1,A3,A5,A6,A12,A17,A24,A29,A31,A33,A36,A37,A39,A40
Subset ID 188	A1,A3,A5,A6,A12,A17,A24,A29,A33,A35,A36,A37,A38,A39
Subset ID 189	A1,A3,A5,A6,A12,A17,A24,A29,A33,A36,A37,A38,A39,A40
Subset ID 190	A1,A3,A5,A6,A12,A17,A24,A30,A31,A33,A36,A37,A39,A40
Subset ID 191	A1,A3,A5,A6,A12,A17,A24,A30,A33,A35,A36,A37,A38,A39
Subset ID 192	A1,A3,A5,A6,A12,A17,A24,A30,A33,A36,A37,A38,A39,A40
Subset ID 193	A1,A3,A5,A6,A12,A17,A24,A33,A35,A36,A37,A39,A40
Subset ID 194	A1,A3,A5,A6,A12,A18,A23,A24,A33,A34,A35,A37,A38,A39
Subset ID 195	A1,A3,A5,A6,A12,A18,A23,A24,A33,A35,A36,A37,A38,A39
Subset ID 196	A1,A3,A5,A6,A12,A18,A23,A24,A33,A36,A37,A39,A40
Subset ID 197	A1,A3,A5,A6,A12,A18,A23,A31,A33,A34,A35,A37,A38,A39
Subset ID 198	A1,A3,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A38,A39
Subset ID 199	A1,A3,A5,A6,A12,A18,A23,A31,A33,A36,A37,A39,A40
Subset ID 200	A1,A3,A5,A6,A12,A18,A24,A25,A29,A33,A36,A37,A39,A40
Subset ID 201	A1,A3,A5,A6,A12,A18,A24,A25,A30,A33,A36,A37,A39,A40
Subset ID 202	A1,A3,A5,A6,A12,A18,A24,A27,A33,A35,A36,A37,A38,A39
Subset ID 203	A1,A3,A5,A6,A12,A18,A24,A29,A30,A33,A34,A35,A37,A38,A39
Subset ID 204	A1,A3,A5,A6,A12,A18,A24,A29,A31,A33,A36,A37,A39,A40
Subset ID 205	A1,A3,A5,A6,A12,A18,A24,A29,A33,A35,A36,A37,A38,A39
Subset ID 206	A1,A3,A5,A6,A12,A18,A24,A29,A33,A36,A37,A38,A39,A40
Subset ID 207	A1,A3,A5,A6,A12,A18,A24,A30,A31,A33,A36,A37,A39,A40
Subset ID 208	A1,A3,A5,A6,A12,A18,A24,A30,A33,A35,A36,A37,A38,A39
Subset ID 209	A1,A3,A5,A6,A12,A18,A24,A30,A33,A36,A37,A38,A39,A40
Subset ID 210	A1,A3,A5,A6,A12,A18,A24,A33,A35,A36,A37,A39,A40
Subset ID 211	A1,A4,A5,A6,A7,A12,A13,A24,A27,A32,A33,A35,A36,A37,A39,A41
Subset ID 212	A1,A4,A5,A6,A7,A12,A13,A24,A27,A33,A34,A35,A36,A37,A39,A41
Subset ID 213	A1,A4,A5,A6,A7,A12,A14,A24,A27,A32,A33,A35,A36,A37,A39,A41
Subset ID 214	A1,A4,A5,A6,A7,A12,A14,A24,A27,A33,A34,A35,A36,A37,A39,A41
Subset ID 215	A1,A4,A5,A6,A7,A12,A16,A24,A27,A32,A33,A35,A36,A37,A39,A41
Subset ID 216	A1,A4,A5,A6,A7,A12,A16,A24,A27,A33,A34,A35,A36,A37,A39,A41
Subset ID 217	A1,A4,A5,A6,A7,A12,A18,A24,A27,A32,A33,A35,A36,A37,A39,A41
Subset ID 218	A1,A4,A5,A6,A7,A12,A18,A24,A27,A33,A34,A35,A36,A37,A39,A41

Subset ID 219	A1,A4,A5,A6,A10,A12,A23,A24,A25,A33,A34,A35,A37,A39,A41
Subset ID 220	A1,A4,A5,A6,A10,A12,A23,A24,A27,A33,A34,A35,A37,A39,A41
Subset ID 221	A1,A4,A5,A6,A10,A12,A23,A24,A30,A33,A34,A35,A37,A39,A41
Subset ID 222	A1,A4,A5,A6,A10,A12,A23,A24,A33,A34,A35,A37,A38,A39,A41
Subset ID 223	A1,A4,A5,A6,A10,A12,A23,A25,A31,A33,A34,A35,A37,A39,A41
Subset ID 224	A1,A4,A5,A6,A10,A12,A23,A27,A31,A33,A34,A35,A37,A39,A41
Subset ID 225	A1,A4,A5,A6,A10,A12,A23,A30,A31,A33,A34,A35,A37,A39,A41
Subset ID 226	A1,A4,A5,A6,A10,A12,A23,A31,A33,A34,A35,A37,A38,A39,A41
Subset ID 227	A1,A4,A5,A6,A10,A12,A24,A29,A30,A33,A34,A35,A37,A39,A41
Subset ID 228	A1,A4,A5,A6,A12,A13,A23,A24,A25,A32,A33,A35,A37,A39,A41
Subset ID 229	A1,A4,A5,A6,A12,A13,A23,A24,A25,A33,A34,A35,A37,A39,A41
Subset ID 230	A1,A4,A5,A6,A12,A13,A23,A24,A27,A32,A33,A35,A37,A39,A41
Subset ID 231	A1,A4,A5,A6,A12,A13,A23,A24,A27,A33,A34,A35,A37,A39,A41
Subset ID 232	A1,A4,A5,A6,A12,A13,A23,A24,A30,A32,A33,A35,A37,A39,A41
Subset ID 233	A1,A4,A5,A6,A12,A13,A23,A24,A30,A33,A34,A35,A37,A39,A41
Subset ID 234	A1,A4,A5,A6,A12,A13,A23,A24,A32,A33,A35,A37,A38,A39,A41
Subset ID 235	A1,A4,A5,A6,A12,A13,A23,A24,A33,A34,A35,A37,A38,A39,A41
Subset ID 236	A1,A4,A5,A6,A12,A13,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 237	A1,A4,A5,A6,A12,A13,A23,A25,A31,A32,A33,A35,A37,A39,A41
Subset ID 238	A1,A4,A5,A6,A12,A13,A23,A25,A31,A33,A34,A35,A37,A39,A41
Subset ID 239	A1,A4,A5,A6,A12,A13,A23,A27,A31,A32,A33,A35,A37,A39,A41
Subset ID 240	A1,A4,A5,A6,A12,A13,A23,A27,A31,A33,A34,A35,A37,A39,A41
Subset ID 241	A1,A4,A5,A6,A12,A13,A23,A30,A31,A32,A33,A35,A37,A39,A41
Subset ID 242	A1,A4,A5,A6,A12,A13,A23,A30,A31,A33,A34,A35,A37,A39,A41
Subset ID 243	A1,A4,A5,A6,A12,A13,A23,A31,A32,A33,A35,A37,A38,A39,A41
Subset ID 244	A1,A4,A5,A6,A12,A13,A23,A31,A33,A34,A35,A37,A38,A39,A41
Subset ID 245	A1,A4,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 246	A1,A4,A5,A6,A12,A13,A24,A29,A30,A32,A33,A35,A37,A39,A41
Subset ID 247	A1,A4,A5,A6,A12,A13,A24,A29,A30,A33,A34,A35,A37,A39,A41
Subset ID 248	A1,A4,A5,A6,A12,A13,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 249	A1,A4,A5,A6,A12,A13,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 250	A1,A4,A5,A6,A12,A14,A23,A24,A25,A32,A33,A35,A37,A39,A41
Subset ID 251	A1,A4,A5,A6,A12,A14,A23,A24,A25,A33,A34,A35,A37,A39,A41
Subset ID 252	A1,A4,A5,A6,A12,A14,A23,A24,A27,A32,A33,A35,A37,A39,A41
Subset ID 253	A1,A4,A5,A6,A12,A14,A23,A24,A27,A33,A34,A35,A37,A39,A41
Subset ID 254	A1,A4,A5,A6,A12,A14,A23,A24,A30,A32,A33,A35,A37,A39,A41
Subset ID 255	A1,A4,A5,A6,A12,A14,A23,A24,A30,A33,A34,A35,A37,A39,A41
Subset ID 256	A1,A4,A5,A6,A12,A14,A23,A24,A32,A33,A35,A37,A38,A39,A41
Subset ID 257	A1,A4,A5,A6,A12,A14,A23,A24,A33,A34,A35,A37,A38,A39,A41
Subset ID 258	A1,A4,A5,A6,A12,A14,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 259	A1,A4,A5,A6,A12,A14,A23,A25,A31,A32,A33,A35,A37,A39,A41
Subset ID 260	A1,A4,A5,A6,A12,A14,A23,A25,A31,A33,A34,A35,A37,A39,A41
Subset ID 261	A1,A4,A5,A6,A12,A14,A23,A27,A31,A32,A33,A35,A37,A39,A41
Subset ID 262	A1,A4,A5,A6,A12,A14,A23,A27,A31,A33,A34,A35,A37,A39,A41
Subset ID 263	A1,A4,A5,A6,A12,A14,A23,A30,A31,A32,A33,A35,A37,A39,A41

Subset ID 264	A1,A4,A5,A6,A12,A14,A23,A30,A31,A33,A34,A35,A37,A39,A41
Subset ID 265	A1,A4,A5,A6,A12,A14,A23,A31,A32,A33,A35,A37,A38,A39,A41
Subset ID 266	A1,A4,A5,A6,A12,A14,A23,A31,A33,A34,A35,A37,A38,A39,A41
Subset ID 267	A1,A4,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 268	A1,A4,A5,A6,A12,A14,A24,A29,A30,A32,A33,A35,A37,A39,A41
Subset ID 269	A1,A4,A5,A6,A12,A14,A24,A29,A30,A33,A34,A35,A37,A39,A41
Subset ID 270	A1,A4,A5,A6,A12,A14,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 271	A1,A4,A5,A6,A12,A14,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 272	A1,A4,A5,A6,A12,A16,A23,A24,A25,A32,A33,A35,A37,A39,A41
Subset ID 273	A1,A4,A5,A6,A12,A16,A23,A24,A25,A33,A34,A35,A37,A39,A41
Subset ID 274	A1,A4,A5,A6,A12,A16,A23,A24,A27,A32,A33,A35,A37,A39,A41
Subset ID 275	A1,A4,A5,A6,A12,A16,A23,A24,A27,A33,A34,A35,A37,A39,A41
Subset ID 276	A1,A4,A5,A6,A12,A16,A23,A24,A30,A32,A33,A35,A37,A39,A41
Subset ID 277	A1,A4,A5,A6,A12,A16,A23,A24,A30,A33,A34,A35,A37,A39,A41
Subset ID 278	A1,A4,A5,A6,A12,A16,A23,A24,A32,A33,A35,A37,A38,A39,A41
Subset ID 279	A1,A4,A5,A6,A12,A16,A23,A24,A33,A34,A35,A37,A38,A39,A41
Subset ID 280	A1,A4,A5,A6,A12,A16,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 281	A1,A4,A5,A6,A12,A16,A23,A25,A31,A32,A33,A35,A37,A39,A41
Subset ID 282	A1,A4,A5,A6,A12,A16,A23,A25,A31,A33,A34,A35,A37,A39,A41
Subset ID 283	A1,A4,A5,A6,A12,A16,A23,A27,A31,A32,A33,A35,A37,A39,A41
Subset ID 284	A1,A4,A5,A6,A12,A16,A23,A27,A31,A33,A34,A35,A37,A39,A41
Subset ID 285	A1,A4,A5,A6,A12,A16,A23,A30,A31,A32,A33,A35,A37,A39,A41
Subset ID 286	A1,A4,A5,A6,A12,A16,A23,A30,A31,A33,A34,A35,A37,A39,A41
Subset ID 287	A1,A4,A5,A6,A12,A16,A23,A31,A32,A33,A35,A37,A38,A39,A41
Subset ID 288	A1,A4,A5,A6,A12,A16,A23,A31,A33,A34,A35,A37,A38,A39,A41
Subset ID 289	A1,A4,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 290	A1,A4,A5,A6,A12,A16,A24,A29,A30,A32,A33,A35,A37,A39,A41
Subset ID 291	A1,A4,A5,A6,A12,A16,A24,A29,A30,A33,A34,A35,A37,A39,A41
Subset ID 292	A1,A4,A5,A6,A12,A16,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 293	A1,A4,A5,A6,A12,A16,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 294	A1,A4,A5,A6,A12,A17,A23,A24,A25,A33,A34,A35,A37,A39,A41
Subset ID 295	A1,A4,A5,A6,A12,A17,A23,A24,A27,A33,A34,A35,A37,A39,A41
Subset ID 296	A1,A4,A5,A6,A12,A17,A23,A24,A30,A33,A34,A35,A37,A39,A41
Subset ID 297	A1,A4,A5,A6,A12,A17,A23,A24,A33,A34,A35,A37,A38,A39,A41
Subset ID 298	A1,A4,A5,A6,A12,A17,A23,A25,A31,A33,A34,A35,A37,A39,A41
Subset ID 299	A1,A4,A5,A6,A12,A17,A23,A27,A31,A33,A34,A35,A37,A39,A41
Subset ID 300	A1,A4,A5,A6,A12,A17,A23,A30,A31,A33,A34,A35,A37,A39,A41
Subset ID 301	A1,A4,A5,A6,A12,A17,A23,A31,A33,A34,A35,A37,A38,A39,A41
Subset ID 302	A1,A4,A5,A6,A12,A17,A24,A29,A30,A33,A34,A35,A37,A39,A41
Subset ID 303	A1,A4,A5,A6,A12,A18,A23,A24,A25,A32,A33,A35,A37,A39,A41
Subset ID 304	A1,A4,A5,A6,A12,A18,A23,A24,A25,A33,A34,A35,A37,A39,A41
Subset ID 305	A1,A4,A5,A6,A12,A18,A23,A24,A27,A32,A33,A35,A37,A39,A41
Subset ID 306	A1,A4,A5,A6,A12,A18,A23,A24,A27,A33,A34,A35,A37,A39,A41
Subset ID 307	A1,A4,A5,A6,A12,A18,A23,A24,A30,A32,A33,A35,A37,A39,A41
Subset ID 308	A1,A4,A5,A6,A12,A18,A23,A24,A30,A33,A34,A35,A37,A39,A41

Subset ID 309	A1,A4,A5,A6,A12,A18,A23,A24,A32,A33,A35,A37,A38,A39,A41
Subset ID 310	A1,A4,A5,A6,A12,A18,A23,A24,A33,A34,A35,A37,A38,A39,A41
Subset ID 311	A1,A4,A5,A6,A12,A18,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 312	A1,A4,A5,A6,A12,A18,A23,A25,A31,A32,A33,A35,A37,A39,A41
Subset ID 313	A1,A4,A5,A6,A12,A18,A23,A25,A31,A33,A34,A35,A37,A39,A41
Subset ID 314	A1,A4,A5,A6,A12,A18,A23,A27,A31,A32,A33,A35,A37,A39,A41
Subset ID 315	A1,A4,A5,A6,A12,A18,A23,A27,A31,A33,A34,A35,A37,A39,A41
Subset ID 316	A1,A4,A5,A6,A12,A18,A23,A30,A31,A32,A33,A35,A37,A39,A41
Subset ID 317	A1,A4,A5,A6,A12,A18,A23,A30,A31,A33,A34,A35,A37,A39,A41
Subset ID 318	A1,A4,A5,A6,A12,A18,A23,A31,A32,A33,A35,A37,A38,A39,A41
Subset ID 319	A1,A4,A5,A6,A12,A18,A23,A31,A33,A34,A35,A37,A38,A39,A41
Subset ID 320	A1,A4,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 321	A1,A4,A5,A6,A12,A18,A24,A29,A30,A32,A33,A35,A37,A39,A41
Subset ID 322	A1,A4,A5,A6,A12,A18,A24,A29,A30,A33,A34,A35,A37,A39,A41
Subset ID 323	A1,A4,A5,A6,A12,A18,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 324	A1,A4,A5,A6,A12,A18,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 325	A1,A5,A6,A7,A12,A13,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 326	A1,A5,A6,A7,A12,A13,A24,A32,A33,A35,A36,A37,A39,A40,A41
Subset ID 327	A1,A5,A6,A7,A12,A13,A24,A33,A34,A35,A36,A37,A39,A40,A41
Subset ID 328	A1,A5,A6,A7,A12,A13,A24,A33,A35,A36,A37,A38,A39,A40,A41
Subset ID 329	A1,A5,A6,A7,A12,A14,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 330	A1,A5,A6,A7,A12,A14,A24,A32,A33,A35,A36,A37,A39,A40,A41
Subset ID 331	A1,A5,A6,A7,A12,A14,A24,A33,A34,A35,A36,A37,A39,A40,A41
Subset ID 332	A1,A5,A6,A7,A12,A14,A24,A33,A35,A36,A37,A38,A39,A40,A41
Subset ID 333	A1,A5,A6,A7,A12,A16,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 334	A1,A5,A6,A7,A12,A16,A24,A32,A33,A35,A36,A37,A39,A40,A41
Subset ID 335	A1,A5,A6,A7,A12,A16,A24,A33,A34,A35,A36,A37,A39,A40,A41
Subset ID 336	A1,A5,A6,A7,A12,A16,A24,A33,A35,A36,A37,A38,A39,A40,A41
Subset ID 337	A1,A5,A6,A7,A12,A18,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 338	A1,A5,A6,A7,A12,A18,A24,A32,A33,A35,A36,A37,A39,A40,A41
Subset ID 339	A1,A5,A6,A7,A12,A18,A24,A33,A34,A35,A36,A37,A39,A40,A41
Subset ID 340	A1,A5,A6,A7,A12,A18,A24,A33,A35,A36,A37,A38,A39,A40,A41
Subset ID 341	A1,A5,A6,A12,A13,A23,A24,A32,A33,A35,A37,A39,A40,A41
Subset ID 342	A1,A5,A6,A12,A13,A23,A24,A33,A34,A35,A37,A39,A40,A41
Subset ID 343	A1,A5,A6,A12,A13,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 344	A1,A5,A6,A12,A13,A23,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 345	A1,A5,A6,A12,A13,A23,A31,A32,A33,A35,A37,A39,A40,A41
Subset ID 346	A1,A5,A6,A12,A13,A23,A31,A33,A34,A35,A37,A39,A40,A41
Subset ID 347	A1,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 348	A1,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 349	A1,A5,A6,A12,A13,A24,A25,A29,A32,A33,A35,A37,A39,A40,A41
Subset ID 350	A1,A5,A6,A12,A13,A24,A25,A29,A33,A34,A35,A37,A39,A40,A41
Subset ID 351	A1,A5,A6,A12,A13,A24,A29,A30,A32,A33,A35,A37,A39,A40,A41
Subset ID 352	A1,A5,A6,A12,A13,A24,A29,A30,A33,A34,A35,A37,A39,A40,A41
Subset ID 353	A1,A5,A6,A12,A13,A24,A29,A32,A33,A35,A37,A38,A39,A40,A41

Subset ID 354	A1,A5,A6,A12,A13,A24,A29,A33,A34,A35,A37,A38,A39,A40,A41
Subset ID 355	A1,A5,A6,A12,A13,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 356	A1,A5,A6,A12,A13,A24,A29,A33,A35,A36,A37,A39,A40,A41
Subset ID 357	A1,A5,A6,A12,A13,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 358	A1,A5,A6,A12,A13,A24,A30,A33,A35,A36,A37,A39,A40,A41
Subset ID 359	A1,A5,A6,A12,A14,A23,A24,A32,A33,A35,A37,A39,A40,A41
Subset ID 360	A1,A5,A6,A12,A14,A23,A24,A33,A34,A35,A37,A39,A40,A41
Subset ID 361	A1,A5,A6,A12,A14,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 362	A1,A5,A6,A12,A14,A23,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 363	A1,A5,A6,A12,A14,A23,A31,A32,A33,A35,A37,A39,A40,A41
Subset ID 364	A1,A5,A6,A12,A14,A23,A31,A33,A34,A35,A37,A39,A40,A41
Subset ID 365	A1,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 366	A1,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 367	A1,A5,A6,A12,A14,A24,A25,A29,A32,A33,A35,A37,A39,A40,A41
Subset ID 368	A1,A5,A6,A12,A14,A24,A25,A29,A33,A34,A35,A37,A39,A40,A41
Subset ID 369	A1,A5,A6,A12,A14,A24,A29,A30,A32,A33,A35,A37,A39,A40,A41
Subset ID 370	A1,A5,A6,A12,A14,A24,A29,A30,A33,A34,A35,A37,A39,A40,A41
Subset ID 371	A1,A5,A6,A12,A14,A24,A29,A32,A33,A35,A37,A38,A39,A40,A41
Subset ID 372	A1,A5,A6,A12,A14,A24,A29,A33,A34,A35,A37,A38,A39,A40,A41
Subset ID 373	A1,A5,A6,A12,A14,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 374	A1,A5,A6,A12,A14,A24,A29,A33,A35,A36,A37,A39,A40,A41
Subset ID 375	A1,A5,A6,A12,A14,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 376	A1,A5,A6,A12,A14,A24,A30,A33,A35,A36,A37,A39,A40,A41
Subset ID 377	A1,A5,A6,A12,A16,A23,A24,A32,A33,A35,A37,A39,A40,A41
Subset ID 378	A1,A5,A6,A12,A16,A23,A24,A33,A34,A35,A37,A39,A40,A41
Subset ID 379	A1,A5,A6,A12,A16,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 380	A1,A5,A6,A12,A16,A23,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 381	A1,A5,A6,A12,A16,A23,A31,A32,A33,A35,A37,A39,A40,A41
Subset ID 382	A1,A5,A6,A12,A16,A23,A31,A33,A34,A35,A37,A39,A40,A41
Subset ID 383	A1,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 384	A1,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 385	A1,A5,A6,A12,A16,A24,A25,A29,A32,A33,A35,A37,A39,A40,A41
Subset ID 386	A1,A5,A6,A12,A16,A24,A25,A29,A33,A34,A35,A37,A39,A40,A41
Subset ID 387	A1,A5,A6,A12,A16,A24,A29,A30,A32,A33,A35,A37,A39,A40,A41
Subset ID 388	A1,A5,A6,A12,A16,A24,A29,A30,A33,A34,A35,A37,A39,A40,A41
Subset ID 389	A1,A5,A6,A12,A16,A24,A29,A32,A33,A35,A37,A38,A39,A40,A41
Subset ID 390	A1,A5,A6,A12,A16,A24,A29,A33,A34,A35,A37,A38,A39,A40,A41
Subset ID 391	A1,A5,A6,A12,A16,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 392	A1,A5,A6,A12,A16,A24,A29,A33,A35,A36,A37,A39,A40,A41
Subset ID 393	A1,A5,A6,A12,A16,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 394	A1,A5,A6,A12,A16,A24,A30,A33,A35,A36,A37,A39,A40,A41
Subset ID 395	A1,A5,A6,A12,A18,A23,A24,A32,A33,A35,A37,A39,A40,A41
Subset ID 396	A1,A5,A6,A12,A18,A23,A24,A33,A34,A35,A37,A39,A40,A41
Subset ID 397	A1,A5,A6,A12,A18,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 398	A1,A5,A6,A12,A18,A23,A24,A33,A35,A36,A37,A39,A40,A41

Subset ID 399	A1,A5,A6,A12,A18,A23,A31,A32,A33,A35,A37,A39,A40,A41
Subset ID 400	A1,A5,A6,A12,A18,A23,A31,A33,A34,A35,A37,A39,A40,A41
Subset ID 401	A1,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 402	A1,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 403	A1,A5,A6,A12,A18,A24,A25,A29,A32,A33,A35,A37,A39,A40,A41
Subset ID 404	A1,A5,A6,A12,A18,A24,A25,A29,A33,A34,A35,A37,A39,A40,A41
Subset ID 405	A1,A5,A6,A12,A18,A24,A29,A30,A32,A33,A35,A37,A39,A40,A41
Subset ID 406	A1,A5,A6,A12,A18,A24,A29,A30,A33,A34,A35,A37,A39,A40,A41
Subset ID 407	A1,A5,A6,A12,A18,A24,A29,A32,A33,A35,A37,A38,A39,A40,A41
Subset ID 408	A1,A5,A6,A12,A18,A24,A29,A33,A34,A35,A37,A38,A39,A40,A41
Subset ID 409	A1,A5,A6,A12,A18,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 410	A1,A5,A6,A12,A18,A24,A29,A33,A35,A36,A37,A39,A40,A41
Subset ID 411	A1,A5,A6,A12,A18,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 412	A1,A5,A6,A12,A18,A24,A30,A33,A35,A36,A37,A39,A40,A41
Subset ID 413	A2,A4,A5,A6,A10,A12,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 414	A2,A4,A5,A6,A10,A12,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 415	A2,A4,A5,A6,A10,A12,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 416	A2,A4,A5,A6,A12,A17,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 417	A2,A4,A5,A6,A12,A17,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 418	A2,A4,A5,A6,A12,A17,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 419	A3,A4,A5,A6,A10,A12,A23,A24,A25,A32,A33,A35,A37,A39
Subset ID 420	A3,A4,A5,A6,A10,A12,A23,A24,A27,A32,A33,A35,A37,A39
Subset ID 421	A3,A4,A5,A6,A10,A12,A23,A24,A30,A32,A33,A35,A37,A39
Subset ID 422	A3,A4,A5,A6,A10,A12,A23,A24,A32,A33,A35,A36,A37,A39
Subset ID 423	A3,A4,A5,A6,A10,A12,A23,A24,A33,A34,A35,A36,A37,A39
Subset ID 424	A3,A4,A5,A6,A10,A12,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 425	A3,A4,A5,A6,A10,A12,A23,A25,A31,A32,A33,A35,A37,A39
Subset ID 426	A3,A4,A5,A6,A10,A12,A23,A27,A31,A32,A33,A35,A37,A39
Subset ID 427	A3,A4,A5,A6,A10,A12,A23,A30,A31,A32,A33,A35,A37,A39
Subset ID 428	A3,A4,A5,A6,A10,A12,A23,A31,A32,A33,A35,A36,A37,A39
Subset ID 429	A3,A4,A5,A6,A10,A12,A23,A31,A33,A34,A35,A36,A37,A39
Subset ID 430	A3,A4,A5,A6,A10,A12,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 431	A3,A4,A5,A6,A10,A12,A24,A25,A27,A32,A33,A35,A37,A39,A40
Subset ID 432	A3,A4,A5,A6,A10,A12,A24,A25,A27,A33,A34,A35,A37,A39,A40
Subset ID 433	A3,A4,A5,A6,A10,A12,A24,A25,A32,A33,A35,A37,A38,A39,A40
Subset ID 434	A3,A4,A5,A6,A10,A12,A24,A25,A33,A34,A35,A37,A38,A39,A40
Subset ID 435	A3,A4,A5,A6,A10,A12,A24,A27,A32,A33,A35,A36,A37,A39
Subset ID 436	A3,A4,A5,A6,A10,A12,A24,A27,A32,A33,A35,A37,A38,A39,A40
Subset ID 437	A3,A4,A5,A6,A10,A12,A24,A27,A33,A34,A35,A36,A37,A39
Subset ID 438	A3,A4,A5,A6,A10,A12,A24,A27,A33,A34,A35,A37,A38,A39,A40
Subset ID 439	A3,A4,A5,A6,A10,A12,A24,A27,A33,A35,A36,A37,A39,A41
Subset ID 440	A3,A4,A5,A6,A10,A12,A24,A27,A33,A35,A37,A38,A39,A40,A41
Subset ID 441	A3,A4,A5,A6,A10,A12,A24,A29,A30,A32,A33,A35,A37,A39
Subset ID 442	A3,A4,A5,A6,A10,A12,A24,A29,A32,A33,A35,A36,A37,A39
Subset ID 443	A3,A4,A5,A6,A10,A12,A24,A29,A33,A34,A35,A36,A37,A39

Subset ID 444	A3,A4,A5,A6,A10,A12,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 445	A3,A4,A5,A6,A10,A12,A24,A30,A32,A33,A35,A36,A37,A39
Subset ID 446	A3,A4,A5,A6,A10,A12,A24,A30,A33,A34,A35,A36,A37,A39
Subset ID 447	A3,A4,A5,A6,A12,A13,A23,A24,A25,A32,A33,A35,A37,A39
Subset ID 448	A3,A4,A5,A6,A12,A13,A23,A24,A27,A32,A33,A35,A37,A39
Subset ID 449	A3,A4,A5,A6,A12,A13,A23,A24,A30,A32,A33,A35,A37,A39
Subset ID 450	A3,A4,A5,A6,A12,A13,A23,A24,A32,A33,A35,A36,A37,A39
Subset ID 451	A3,A4,A5,A6,A12,A13,A23,A24,A33,A34,A35,A36,A37,A39
Subset ID 452	A3,A4,A5,A6,A12,A13,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 453	A3,A4,A5,A6,A12,A13,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 454	A3,A4,A5,A6,A12,A13,A23,A25,A31,A32,A33,A35,A37,A39
Subset ID 455	A3,A4,A5,A6,A12,A13,A23,A27,A31,A32,A33,A35,A37,A39
Subset ID 456	A3,A4,A5,A6,A12,A13,A23,A30,A31,A32,A33,A35,A37,A39
Subset ID 457	A3,A4,A5,A6,A12,A13,A23,A31,A32,A33,A35,A36,A37,A39
Subset ID 458	A3,A4,A5,A6,A12,A13,A23,A31,A33,A34,A35,A36,A37,A39
Subset ID 459	A3,A4,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 460	A3,A4,A5,A6,A12,A13,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 461	A3,A4,A5,A6,A12,A13,A24,A25,A27,A32,A33,A35,A37,A39,A40
Subset ID 462	A3,A4,A5,A6,A12,A13,A24,A25,A27,A33,A34,A35,A37,A39,A40
Subset ID 463	A3,A4,A5,A6,A12,A13,A24,A25,A32,A33,A35,A37,A38,A39,A40
Subset ID 464	A3,A4,A5,A6,A12,A13,A24,A25,A33,A34,A35,A37,A38,A39,A40
Subset ID 465	A3,A4,A5,A6,A12,A13,A24,A27,A32,A33,A35,A36,A37,A39
Subset ID 466	A3,A4,A5,A6,A12,A13,A24,A27,A32,A33,A35,A37,A38,A39,A40
Subset ID 467	A3,A4,A5,A6,A12,A13,A24,A27,A33,A34,A35,A36,A37,A39
Subset ID 468	A3,A4,A5,A6,A12,A13,A24,A27,A33,A34,A35,A37,A38,A39,A40
Subset ID 469	A3,A4,A5,A6,A12,A13,A24,A27,A33,A35,A36,A37,A39,A41
Subset ID 470	A3,A4,A5,A6,A12,A13,A24,A27,A33,A35,A37,A38,A39,A40,A41
Subset ID 471	A3,A4,A5,A6,A12,A13,A24,A29,A30,A32,A33,A35,A37,A39
Subset ID 472	A3,A4,A5,A6,A12,A13,A24,A29,A32,A33,A35,A36,A37,A39
Subset ID 473	A3,A4,A5,A6,A12,A13,A24,A29,A33,A34,A35,A36,A37,A39
Subset ID 474	A3,A4,A5,A6,A12,A13,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 475	A3,A4,A5,A6,A12,A13,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 476	A3,A4,A5,A6,A12,A13,A24,A30,A32,A33,A35,A36,A37,A39
Subset ID 477	A3,A4,A5,A6,A12,A13,A24,A30,A33,A34,A35,A36,A37,A39
Subset ID 478	A3,A4,A5,A6,A12,A13,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 479	A3,A4,A5,A6,A12,A14,A23,A24,A25,A32,A33,A35,A37,A39
Subset ID 480	A3,A4,A5,A6,A12,A14,A23,A24,A27,A32,A33,A35,A37,A39
Subset ID 481	A3,A4,A5,A6,A12,A14,A23,A24,A30,A32,A33,A35,A37,A39
Subset ID 482	A3,A4,A5,A6,A12,A14,A23,A24,A32,A33,A35,A36,A37,A39
Subset ID 483	A3,A4,A5,A6,A12,A14,A23,A24,A33,A34,A35,A36,A37,A39
Subset ID 484	A3,A4,A5,A6,A12,A14,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 485	A3,A4,A5,A6,A12,A14,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 486	A3,A4,A5,A6,A12,A14,A23,A25,A31,A32,A33,A35,A37,A39
Subset ID 487	A3,A4,A5,A6,A12,A14,A23,A27,A31,A32,A33,A35,A37,A39
Subset ID 488	A3,A4,A5,A6,A12,A14,A23,A30,A31,A32,A33,A35,A37,A39

Subset ID 489	A3,A4,A5,A6,A12,A14,A23,A31,A32,A33,A35,A36,A37,A39
Subset ID 490	A3,A4,A5,A6,A12,A14,A23,A31,A33,A34,A35,A36,A37,A39
Subset ID 491	A3,A4,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 492	A3,A4,A5,A6,A12,A14,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 493	A3,A4,A5,A6,A12,A14,A24,A25,A27,A32,A33,A35,A37,A39,A40
Subset ID 494	A3,A4,A5,A6,A12,A14,A24,A25,A27,A33,A34,A35,A37,A39,A40
Subset ID 495	A3,A4,A5,A6,A12,A14,A24,A25,A32,A33,A35,A37,A38,A39,A40
Subset ID 496	A3,A4,A5,A6,A12,A14,A24,A25,A33,A34,A35,A37,A38,A39,A40
Subset ID 497	A3,A4,A5,A6,A12,A14,A24,A27,A32,A33,A35,A36,A37,A39
Subset ID 498	A3,A4,A5,A6,A12,A14,A24,A27,A32,A33,A35,A37,A38,A39,A40
Subset ID 499	A3,A4,A5,A6,A12,A14,A24,A27,A33,A34,A35,A36,A37,A39
Subset ID 500	A3,A4,A5,A6,A12,A14,A24,A27,A33,A34,A35,A37,A38,A39,A40
Subset ID 501	A3,A4,A5,A6,A12,A14,A24,A27,A33,A35,A36,A37,A39,A41
Subset ID 502	A3,A4,A5,A6,A12,A14,A24,A27,A33,A35,A37,A38,A39,A40,A41
Subset ID 503	A3,A4,A5,A6,A12,A14,A24,A29,A30,A32,A33,A35,A37,A39
Subset ID 504	A3,A4,A5,A6,A12,A14,A24,A29,A32,A33,A35,A36,A37,A39
Subset ID 505	A3,A4,A5,A6,A12,A14,A24,A29,A33,A34,A35,A36,A37,A39
Subset ID 506	A3,A4,A5,A6,A12,A14,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 507	A3,A4,A5,A6,A12,A14,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 508	A3,A4,A5,A6,A12,A14,A24,A30,A32,A33,A35,A36,A37,A39
Subset ID 509	A3,A4,A5,A6,A12,A14,A24,A30,A33,A34,A35,A36,A37,A39
Subset ID 510	A3,A4,A5,A6,A12,A14,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 511	A3,A4,A5,A6,A12,A16,A23,A24,A25,A32,A33,A35,A37,A39
Subset ID 512	A3,A4,A5,A6,A12,A16,A23,A24,A27,A32,A33,A35,A37,A39
Subset ID 513	A3,A4,A5,A6,A12,A16,A23,A24,A30,A32,A33,A35,A37,A39
Subset ID 514	A3,A4,A5,A6,A12,A16,A23,A24,A32,A33,A35,A36,A37,A39
Subset ID 515	A3,A4,A5,A6,A12,A16,A23,A24,A33,A34,A35,A36,A37,A39
Subset ID 516	A3,A4,A5,A6,A12,A16,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 517	A3,A4,A5,A6,A12,A16,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 518	A3,A4,A5,A6,A12,A16,A23,A25,A31,A32,A33,A35,A37,A39
Subset ID 519	A3,A4,A5,A6,A12,A16,A23,A27,A31,A32,A33,A35,A37,A39
Subset ID 520	A3,A4,A5,A6,A12,A16,A23,A30,A31,A32,A33,A35,A37,A39
Subset ID 521	A3,A4,A5,A6,A12,A16,A23,A31,A32,A33,A35,A36,A37,A39
Subset ID 522	A3,A4,A5,A6,A12,A16,A23,A31,A33,A34,A35,A36,A37,A39
Subset ID 523	A3,A4,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 524	A3,A4,A5,A6,A12,A16,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 525	A3,A4,A5,A6,A12,A16,A24,A25,A27,A32,A33,A35,A37,A39,A40
Subset ID 526	A3,A4,A5,A6,A12,A16,A24,A25,A27,A33,A34,A35,A37,A39,A40
Subset ID 527	A3,A4,A5,A6,A12,A16,A24,A25,A32,A33,A35,A37,A38,A39,A40
Subset ID 528	A3,A4,A5,A6,A12,A16,A24,A25,A33,A34,A35,A37,A38,A39,A40
Subset ID 529	A3,A4,A5,A6,A12,A16,A24,A27,A32,A33,A35,A36,A37,A39
Subset ID 530	A3,A4,A5,A6,A12,A16,A24,A27,A32,A33,A35,A37,A38,A39,A40
Subset ID 531	A3,A4,A5,A6,A12,A16,A24,A27,A33,A34,A35,A36,A37,A39
Subset ID 532	A3,A4,A5,A6,A12,A16,A24,A27,A33,A34,A35,A37,A38,A39,A40
Subset ID 533	A3,A4,A5,A6,A12,A16,A24,A27,A33,A35,A36,A37,A39,A41

Subset ID 534	A3,A4,A5,A6,A12,A16,A24,A27,A33,A35,A37,A38,A39,A40,A41
Subset ID 535	A3,A4,A5,A6,A12,A16,A24,A29,A30,A32,A33,A35,A37,A39
Subset ID 536	A3,A4,A5,A6,A12,A16,A24,A29,A32,A33,A35,A36,A37,A39
Subset ID 537	A3,A4,A5,A6,A12,A16,A24,A29,A33,A34,A35,A36,A37,A39
Subset ID 538	A3,A4,A5,A6,A12,A16,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 539	A3,A4,A5,A6,A12,A16,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 540	A3,A4,A5,A6,A12,A16,A24,A30,A32,A33,A35,A36,A37,A39
Subset ID 541	A3,A4,A5,A6,A12,A16,A24,A30,A33,A34,A35,A36,A37,A39
Subset ID 542	A3,A4,A5,A6,A12,A16,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 543	A3,A4,A5,A6,A12,A17,A23,A24,A25,A32,A33,A35,A37,A39
Subset ID 544	A3,A4,A5,A6,A12,A17,A23,A24,A27,A32,A33,A35,A37,A39
Subset ID 545	A3,A4,A5,A6,A12,A17,A23,A24,A30,A32,A33,A35,A37,A39
Subset ID 546	A3,A4,A5,A6,A12,A17,A23,A24,A32,A33,A35,A36,A37,A39
Subset ID 547	A3,A4,A5,A6,A12,A17,A23,A24,A33,A34,A35,A36,A37,A39
Subset ID 548	A3,A4,A5,A6,A12,A17,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 549	A3,A4,A5,A6,A12,A17,A23,A25,A31,A32,A33,A35,A37,A39
Subset ID 550	A3,A4,A5,A6,A12,A17,A23,A27,A31,A32,A33,A35,A37,A39
Subset ID 551	A3,A4,A5,A6,A12,A17,A23,A30,A31,A32,A33,A35,A37,A39
Subset ID 552	A3,A4,A5,A6,A12,A17,A23,A31,A32,A33,A35,A36,A37,A39
Subset ID 553	A3,A4,A5,A6,A12,A17,A23,A31,A33,A34,A35,A36,A37,A39
Subset ID 554	A3,A4,A5,A6,A12,A17,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 555	A3,A4,A5,A6,A12,A17,A24,A25,A27,A32,A33,A35,A37,A39,A40
Subset ID 556	A3,A4,A5,A6,A12,A17,A24,A25,A27,A33,A34,A35,A37,A39,A40
Subset ID 557	A3,A4,A5,A6,A12,A17,A24,A25,A32,A33,A35,A37,A38,A39,A40
Subset ID 558	A3,A4,A5,A6,A12,A17,A24,A25,A33,A34,A35,A37,A38,A39,A40
Subset ID 559	A3,A4,A5,A6,A12,A17,A24,A27,A32,A33,A35,A36,A37,A39
Subset ID 560	A3,A4,A5,A6,A12,A17,A24,A27,A32,A33,A35,A37,A38,A39,A40
Subset ID 561	A3,A4,A5,A6,A12,A17,A24,A27,A33,A34,A35,A36,A37,A39
Subset ID 562	A3,A4,A5,A6,A12,A17,A24,A27,A33,A34,A35,A37,A38,A39,A40
Subset ID 563	A3,A4,A5,A6,A12,A17,A24,A27,A33,A35,A36,A37,A39,A41
Subset ID 564	A3,A4,A5,A6,A12,A17,A24,A27,A33,A35,A37,A38,A39,A40,A41
Subset ID 565	A3,A4,A5,A6,A12,A17,A24,A29,A30,A32,A33,A35,A37,A39
Subset ID 566	A3,A4,A5,A6,A12,A17,A24,A29,A32,A33,A35,A36,A37,A39
Subset ID 567	A3,A4,A5,A6,A12,A17,A24,A29,A33,A34,A35,A36,A37,A39
Subset ID 568	A3,A4,A5,A6,A12,A17,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 569	A3,A4,A5,A6,A12,A17,A24,A30,A32,A33,A35,A36,A37,A39
Subset ID 570	A3,A4,A5,A6,A12,A17,A24,A30,A33,A34,A35,A36,A37,A39
Subset ID 571	A3,A4,A5,A6,A12,A18,A23,A24,A25,A32,A33,A35,A37,A39
Subset ID 572	A3,A4,A5,A6,A12,A18,A23,A24,A27,A32,A33,A35,A37,A39
Subset ID 573	A3,A4,A5,A6,A12,A18,A23,A24,A30,A32,A33,A35,A37,A39
Subset ID 574	A3,A4,A5,A6,A12,A18,A23,A24,A32,A33,A35,A36,A37,A39
Subset ID 575	A3,A4,A5,A6,A12,A18,A23,A24,A33,A34,A35,A36,A37,A39
Subset ID 576	A3,A4,A5,A6,A12,A18,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 577	A3,A4,A5,A6,A12,A18,A23,A24,A33,A35,A37,A38,A39,A40,A41
Subset ID 578	A3,A4,A5,A6,A12,A18,A23,A25,A31,A32,A33,A35,A37,A39

Subset ID 579	A3,A4,A5,A6,A12,A18,A23,A27,A31,A32,A33,A35,A37,A39
Subset ID 580	A3,A4,A5,A6,A12,A18,A23,A30,A31,A32,A33,A35,A37,A39
Subset ID 581	A3,A4,A5,A6,A12,A18,A23,A31,A32,A33,A35,A36,A37,A39
Subset ID 582	A3,A4,A5,A6,A12,A18,A23,A31,A33,A34,A35,A36,A37,A39
Subset ID 583	A3,A4,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 584	A3,A4,A5,A6,A12,A18,A23,A31,A33,A35,A37,A38,A39,A40,A41
Subset ID 585	A3,A4,A5,A6,A12,A18,A24,A25,A27,A32,A33,A35,A37,A39,A40
Subset ID 586	A3,A4,A5,A6,A12,A18,A24,A25,A27,A33,A34,A35,A37,A39,A40
Subset ID 587	A3,A4,A5,A6,A12,A18,A24,A25,A32,A33,A35,A37,A38,A39,A40
Subset ID 588	A3,A4,A5,A6,A12,A18,A24,A25,A33,A34,A35,A37,A38,A39,A40
Subset ID 589	A3,A4,A5,A6,A12,A18,A24,A27,A32,A33,A35,A36,A37,A39
Subset ID 590	A3,A4,A5,A6,A12,A18,A24,A27,A32,A33,A35,A37,A38,A39,A40
Subset ID 591	A3,A4,A5,A6,A12,A18,A24,A27,A33,A34,A35,A36,A37,A39
Subset ID 592	A3,A4,A5,A6,A12,A18,A24,A27,A33,A34,A35,A37,A38,A39,A40
Subset ID 593	A3,A4,A5,A6,A12,A18,A24,A27,A33,A35,A36,A37,A39,A41
Subset ID 594	A3,A4,A5,A6,A12,A18,A24,A27,A33,A35,A37,A38,A39,A40,A41
Subset ID 595	A3,A4,A5,A6,A12,A18,A24,A29,A30,A32,A33,A35,A37,A39
Subset ID 596	A3,A4,A5,A6,A12,A18,A24,A29,A32,A33,A35,A36,A37,A39
Subset ID 597	A3,A4,A5,A6,A12,A18,A24,A29,A33,A34,A35,A36,A37,A39
Subset ID 598	A3,A4,A5,A6,A12,A18,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 599	A3,A4,A5,A6,A12,A18,A24,A29,A33,A35,A37,A38,A39,A40,A41
Subset ID 600	A3,A4,A5,A6,A12,A18,A24,A30,A32,A33,A35,A36,A37,A39
Subset ID 601	A3,A4,A5,A6,A12,A18,A24,A30,A33,A34,A35,A36,A37,A39
Subset ID 602	A3,A4,A5,A6,A12,A18,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 603	A3,A5,A6,A10,A12,A23,A24,A32,A33,A35,A37,A38,A39
Subset ID 604	A3,A5,A6,A10,A12,A23,A24,A32,A33,A35,A37,A39,A40
Subset ID 605	A3,A5,A6,A10,A12,A23,A24,A32,A33,A36,A37,A39,A40
Subset ID 606	A3,A5,A6,A10,A12,A23,A24,A33,A34,A35,A36,A37,A38,A39
Subset ID 607	A3,A5,A6,A10,A12,A23,A24,A33,A34,A35,A37,A39,A40
Subset ID 608	A3,A5,A6,A10,A12,A23,A24,A33,A36,A37,A38,A39,A40,A41
Subset ID 609	A3,A5,A6,A10,A12,A23,A31,A32,A33,A35,A37,A38,A39
Subset ID 610	A3,A5,A6,A10,A12,A23,A31,A32,A33,A35,A37,A39,A40
Subset ID 611	A3,A5,A6,A10,A12,A23,A31,A32,A33,A36,A37,A39,A40
Subset ID 612	A3,A5,A6,A10,A12,A23,A31,A33,A34,A35,A36,A37,A38,A39
Subset ID 613	A3,A5,A6,A10,A12,A23,A31,A33,A34,A35,A37,A39,A40
Subset ID 614	A3,A5,A6,A10,A12,A23,A31,A33,A34,A36,A37,A39,A40
Subset ID 615	A3,A5,A6,A10,A12,A23,A31,A33,A36,A37,A38,A39,A40,A41
Subset ID 616	A3,A5,A6,A10,A12,A24,A25,A29,A32,A33,A35,A37,A39,A40
Subset ID 617	A3,A5,A6,A10,A12,A24,A25,A29,A33,A34,A35,A37,A39,A40
Subset ID 618	A3,A5,A6,A10,A12,A24,A25,A30,A32,A33,A35,A37,A38,A39,A40
Subset ID 619	A3,A5,A6,A10,A12,A24,A25,A30,A33,A34,A35,A37,A38,A39,A40
Subset ID 620	A3,A5,A6,A10,A12,A24,A25,A32,A33,A36,A37,A39,A40
Subset ID 621	A3,A5,A6,A10,A12,A24,A27,A30,A32,A33,A35,A37,A39,A40
Subset ID 622	A3,A5,A6,A10,A12,A24,A27,A30,A33,A34,A35,A37,A39,A40
Subset ID 623	A3,A5,A6,A10,A12,A24,A27,A32,A33,A35,A36,A37,A38,A39

Subset ID 624	A3,A5,A6,A10,A12,A24,A27,A33,A34,A35,A36,A37,A38,A39
Subset ID 625	A3,A5,A6,A10,A12,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 626	A3,A5,A6,A10,A12,A24,A29,A30,A32,A33,A35,A37,A38,A39
Subset ID 627	A3,A5,A6,A10,A12,A24,A29,A30,A32,A33,A35,A37,A39,A40
Subset ID 628	A3,A5,A6,A10,A12,A24,A29,A30,A33,A34,A35,A37,A39,A40
Subset ID 629	A3,A5,A6,A10,A12,A24,A29,A32,A33,A35,A36,A37,A38,A39
Subset ID 630	A3,A5,A6,A10,A12,A24,A29,A32,A33,A35,A37,A38,A39,A40
Subset ID 631	A3,A5,A6,A10,A12,A24,A29,A33,A34,A35,A36,A37,A38,A39
Subset ID 632	A3,A5,A6,A10,A12,A24,A29,A33,A34,A35,A37,A38,A39,A40
Subset ID 633	A3,A5,A6,A10,A12,A24,A29,A33,A36,A37,A38,A39,A40,A41
Subset ID 634	A3,A5,A6,A10,A12,A24,A30,A32,A33,A35,A36,A37,A38,A39
Subset ID 635	A3,A5,A6,A10,A12,A24,A30,A33,A34,A35,A36,A37,A38,A39
Subset ID 636	A3,A5,A6,A10,A12,A24,A30,A33,A36,A37,A38,A39,A40,A41
Subset ID 637	A3,A5,A6,A10,A12,A24,A31,A32,A33,A36,A37,A39,A40
Subset ID 638	A3,A5,A6,A10,A12,A24,A32,A33,A35,A36,A37,A39,A40
Subset ID 639	A3,A5,A6,A10,A12,A24,A32,A33,A36,A37,A38,A39,A40
Subset ID 640	A3,A5,A6,A10,A12,A24,A33,A34,A36,A37,A39,A40
Subset ID 641	A3,A5,A6,A10,A12,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 642	A3,A5,A6,A12,A13,A23,A24,A32,A33,A35,A37,A38,A39
Subset ID 643	A3,A5,A6,A12,A13,A23,A24,A32,A33,A35,A37,A39,A40
Subset ID 644	A3,A5,A6,A12,A13,A23,A24,A32,A33,A36,A37,A39,A40
Subset ID 645	A3,A5,A6,A12,A13,A23,A24,A33,A34,A35,A36,A37,A38,A39
Subset ID 646	A3,A5,A6,A12,A13,A23,A24,A33,A34,A35,A37,A39,A40
Subset ID 647	A3,A5,A6,A12,A13,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 648	A3,A5,A6,A12,A13,A23,A24,A33,A36,A37,A38,A39,A40,A41
Subset ID 649	A3,A5,A6,A12,A13,A23,A31,A32,A33,A35,A37,A38,A39
Subset ID 650	A3,A5,A6,A12,A13,A23,A31,A32,A33,A35,A37,A39,A40
Subset ID 651	A3,A5,A6,A12,A13,A23,A31,A32,A33,A36,A37,A39,A40
Subset ID 652	A3,A5,A6,A12,A13,A23,A31,A33,A34,A35,A36,A37,A38,A39
Subset ID 653	A3,A5,A6,A12,A13,A23,A31,A33,A34,A35,A37,A39,A40
Subset ID 654	A3,A5,A6,A12,A13,A23,A31,A33,A34,A36,A37,A39,A40
Subset ID 655	A3,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 656	A3,A5,A6,A12,A13,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 657	A3,A5,A6,A12,A13,A23,A31,A33,A36,A37,A38,A39,A40,A41
Subset ID 658	A3,A5,A6,A12,A13,A24,A25,A29,A32,A33,A35,A37,A39,A40
Subset ID 659	A3,A5,A6,A12,A13,A24,A25,A29,A33,A34,A35,A37,A39,A40
Subset ID 660	A3,A5,A6,A12,A13,A24,A25,A30,A32,A33,A35,A37,A38,A39,A40
Subset ID 661	A3,A5,A6,A12,A13,A24,A25,A30,A33,A34,A35,A37,A38,A39,A40
Subset ID 662	A3,A5,A6,A12,A13,A24,A25,A32,A33,A36,A37,A39,A40
Subset ID 663	A3,A5,A6,A12,A13,A24,A27,A30,A32,A33,A35,A37,A39,A40
Subset ID 664	A3,A5,A6,A12,A13,A24,A27,A30,A33,A34,A35,A37,A39,A40
Subset ID 665	A3,A5,A6,A12,A13,A24,A27,A32,A33,A35,A36,A37,A38,A39
Subset ID 666	A3,A5,A6,A12,A13,A24,A27,A33,A34,A35,A36,A37,A38,A39
Subset ID 667	A3,A5,A6,A12,A13,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 668	A3,A5,A6,A12,A13,A24,A29,A30,A32,A33,A35,A37,A38,A39

Subset ID 669	A3,A5,A6,A12,A13,A24,A29,A30,A32,A33,A35,A37,A39,A40
Subset ID 670	A3,A5,A6,A12,A13,A24,A29,A30,A33,A34,A35,A37,A39,A40
Subset ID 671	A3,A5,A6,A12,A13,A24,A29,A32,A33,A35,A36,A37,A38,A39
Subset ID 672	A3,A5,A6,A12,A13,A24,A29,A32,A33,A35,A37,A38,A39,A40
Subset ID 673	A3,A5,A6,A12,A13,A24,A29,A33,A34,A35,A36,A37,A38,A39
Subset ID 674	A3,A5,A6,A12,A13,A24,A29,A33,A34,A35,A37,A38,A39,A40
Subset ID 675	A3,A5,A6,A12,A13,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 676	A3,A5,A6,A12,A13,A24,A29,A33,A36,A37,A38,A39,A40,A41
Subset ID 677	A3,A5,A6,A12,A13,A24,A30,A32,A33,A35,A36,A37,A38,A39
Subset ID 678	A3,A5,A6,A12,A13,A24,A30,A33,A34,A35,A36,A37,A38,A39
Subset ID 679	A3,A5,A6,A12,A13,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 680	A3,A5,A6,A12,A13,A24,A30,A33,A36,A37,A38,A39,A40,A41
Subset ID 681	A3,A5,A6,A12,A13,A24,A31,A32,A33,A36,A37,A39,A40
Subset ID 682	A3,A5,A6,A12,A13,A24,A32,A33,A35,A36,A37,A39,A40
Subset ID 683	A3,A5,A6,A12,A13,A24,A32,A33,A36,A37,A38,A39,A40
Subset ID 684	A3,A5,A6,A12,A13,A24,A33,A34,A36,A37,A39,A40
Subset ID 685	A3,A5,A6,A12,A13,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 686	A3,A5,A6,A12,A14,A23,A24,A32,A33,A35,A37,A38,A39
Subset ID 687	A3,A5,A6,A12,A14,A23,A24,A32,A33,A35,A37,A39,A40
Subset ID 688	A3,A5,A6,A12,A14,A23,A24,A32,A33,A36,A37,A39,A40
Subset ID 689	A3,A5,A6,A12,A14,A23,A24,A33,A34,A35,A36,A37,A38,A39
Subset ID 690	A3,A5,A6,A12,A14,A23,A24,A33,A34,A35,A37,A39,A40
Subset ID 691	A3,A5,A6,A12,A14,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 692	A3,A5,A6,A12,A14,A23,A24,A33,A36,A37,A38,A39,A40,A41
Subset ID 693	A3,A5,A6,A12,A14,A23,A31,A32,A33,A35,A37,A38,A39
Subset ID 694	A3,A5,A6,A12,A14,A23,A31,A32,A33,A35,A37,A39,A40
Subset ID 695	A3,A5,A6,A12,A14,A23,A31,A32,A33,A36,A37,A39,A40
Subset ID 696	A3,A5,A6,A12,A14,A23,A31,A33,A34,A35,A36,A37,A38,A39
Subset ID 697	A3,A5,A6,A12,A14,A23,A31,A33,A34,A35,A37,A39,A40
Subset ID 698	A3,A5,A6,A12,A14,A23,A31,A33,A34,A36,A37,A39,A40
Subset ID 699	A3,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 700	A3,A5,A6,A12,A14,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 701	A3,A5,A6,A12,A14,A23,A31,A33,A36,A37,A38,A39,A40,A41
Subset ID 702	A3,A5,A6,A12,A14,A24,A25,A29,A32,A33,A35,A37,A39,A40
Subset ID 703	A3,A5,A6,A12,A14,A24,A25,A29,A33,A34,A35,A37,A39,A40
Subset ID 704	A3,A5,A6,A12,A14,A24,A25,A30,A32,A33,A35,A37,A38,A39,A40
Subset ID 705	A3,A5,A6,A12,A14,A24,A25,A30,A33,A34,A35,A37,A38,A39,A40
Subset ID 706	A3,A5,A6,A12,A14,A24,A25,A32,A33,A36,A37,A39,A40
Subset ID 707	A3,A5,A6,A12,A14,A24,A27,A30,A32,A33,A35,A37,A39,A40
Subset ID 708	A3,A5,A6,A12,A14,A24,A27,A30,A33,A34,A35,A37,A39,A40
Subset ID 709	A3,A5,A6,A12,A14,A24,A27,A32,A33,A35,A36,A37,A38,A39
Subset ID 710	A3,A5,A6,A12,A14,A24,A27,A33,A34,A35,A36,A37,A38,A39
Subset ID 711	A3,A5,A6,A12,A14,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 712	A3,A5,A6,A12,A14,A24,A29,A30,A32,A33,A35,A37,A38,A39
Subset ID 713	A3,A5,A6,A12,A14,A24,A29,A30,A32,A33,A35,A37,A39,A40

Subset ID 714	A3,A5,A6,A12,A14,A24,A29,A30,A33,A34,A35,A37,A39,A40
Subset ID 715	A3,A5,A6,A12,A14,A24,A29,A32,A33,A35,A36,A37,A38,A39
Subset ID 716	A3,A5,A6,A12,A14,A24,A29,A32,A33,A35,A37,A38,A39,A40
Subset ID 717	A3,A5,A6,A12,A14,A24,A29,A33,A34,A35,A36,A37,A38,A39
Subset ID 718	A3,A5,A6,A12,A14,A24,A29,A33,A34,A35,A37,A38,A39,A40
Subset ID 719	A3,A5,A6,A12,A14,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 720	A3,A5,A6,A12,A14,A24,A29,A33,A36,A37,A38,A39,A40,A41
Subset ID 721	A3,A5,A6,A12,A14,A24,A30,A32,A33,A35,A36,A37,A38,A39
Subset ID 722	A3,A5,A6,A12,A14,A24,A30,A33,A34,A35,A36,A37,A38,A39
Subset ID 723	A3,A5,A6,A12,A14,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 724	A3,A5,A6,A12,A14,A24,A30,A33,A36,A37,A38,A39,A40,A41
Subset ID 725	A3,A5,A6,A12,A14,A24,A31,A32,A33,A36,A37,A39,A40
Subset ID 726	A3,A5,A6,A12,A14,A24,A32,A33,A35,A36,A37,A39,A40
Subset ID 727	A3,A5,A6,A12,A14,A24,A32,A33,A36,A37,A38,A39,A40
Subset ID 728	A3,A5,A6,A12,A14,A24,A33,A34,A36,A37,A39,A40
Subset ID 729	A3,A5,A6,A12,A14,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 730	A3,A5,A6,A12,A16,A23,A24,A32,A33,A35,A37,A38,A39
Subset ID 731	A3,A5,A6,A12,A16,A23,A24,A32,A33,A35,A37,A39,A40
Subset ID 732	A3,A5,A6,A12,A16,A23,A24,A32,A33,A36,A37,A39,A40
Subset ID 733	A3,A5,A6,A12,A16,A23,A24,A33,A34,A35,A36,A37,A38,A39
Subset ID 734	A3,A5,A6,A12,A16,A23,A24,A33,A34,A35,A37,A39,A40
Subset ID 735	A3,A5,A6,A12,A16,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 736	A3,A5,A6,A12,A16,A23,A24,A33,A36,A37,A38,A39,A40,A41
Subset ID 737	A3,A5,A6,A12,A16,A23,A31,A32,A33,A35,A37,A38,A39
Subset ID 738	A3,A5,A6,A12,A16,A23,A31,A32,A33,A35,A37,A39,A40
Subset ID 739	A3,A5,A6,A12,A16,A23,A31,A32,A33,A36,A37,A39,A40
Subset ID 740	A3,A5,A6,A12,A16,A23,A31,A33,A34,A35,A36,A37,A38,A39
Subset ID 741	A3,A5,A6,A12,A16,A23,A31,A33,A34,A35,A37,A39,A40
Subset ID 742	A3,A5,A6,A12,A16,A23,A31,A33,A34,A36,A37,A39,A40
Subset ID 743	A3,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 744	A3,A5,A6,A12,A16,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 745	A3,A5,A6,A12,A16,A23,A31,A33,A36,A37,A38,A39,A40,A41
Subset ID 746	A3,A5,A6,A12,A16,A24,A25,A29,A32,A33,A35,A37,A39,A40
Subset ID 747	A3,A5,A6,A12,A16,A24,A25,A29,A33,A34,A35,A37,A39,A40
Subset ID 748	A3,A5,A6,A12,A16,A24,A25,A30,A32,A33,A35,A37,A38,A39,A40
Subset ID 749	A3,A5,A6,A12,A16,A24,A25,A30,A33,A34,A35,A37,A38,A39,A40
Subset ID 750	A3,A5,A6,A12,A16,A24,A25,A32,A33,A36,A37,A39,A40
Subset ID 751	A3,A5,A6,A12,A16,A24,A27,A30,A32,A33,A35,A37,A39,A40
Subset ID 752	A3,A5,A6,A12,A16,A24,A27,A30,A33,A34,A35,A37,A39,A40
Subset ID 753	A3,A5,A6,A12,A16,A24,A27,A32,A33,A35,A36,A37,A38,A39
Subset ID 754	A3,A5,A6,A12,A16,A24,A27,A33,A34,A35,A36,A37,A38,A39
Subset ID 755	A3,A5,A6,A12,A16,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 756	A3,A5,A6,A12,A16,A24,A29,A30,A32,A33,A35,A37,A38,A39
Subset ID 757	A3,A5,A6,A12,A16,A24,A29,A30,A32,A33,A35,A37,A39,A40
Subset ID 758	A3,A5,A6,A12,A16,A24,A29,A30,A33,A34,A35,A37,A39,A40

Subset ID 759	A3,A5,A6,A12,A16,A24,A29,A32,A33,A35,A36,A37,A38,A39
Subset ID 760	A3,A5,A6,A12,A16,A24,A29,A32,A33,A35,A37,A38,A39,A40
Subset ID 761	A3,A5,A6,A12,A16,A24,A29,A33,A34,A35,A36,A37,A38,A39
Subset ID 762	A3,A5,A6,A12,A16,A24,A29,A33,A34,A35,A37,A38,A39,A40
Subset ID 763	A3,A5,A6,A12,A16,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 764	A3,A5,A6,A12,A16,A24,A29,A33,A36,A37,A38,A39,A40,A41
Subset ID 765	A3,A5,A6,A12,A16,A24,A30,A32,A33,A35,A36,A37,A38,A39
Subset ID 766	A3,A5,A6,A12,A16,A24,A30,A33,A34,A35,A36,A37,A38,A39
Subset ID 767	A3,A5,A6,A12,A16,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 768	A3,A5,A6,A12,A16,A24,A30,A33,A36,A37,A38,A39,A40,A41
Subset ID 769	A3,A5,A6,A12,A16,A24,A31,A32,A33,A36,A37,A39,A40
Subset ID 770	A3,A5,A6,A12,A16,A24,A32,A33,A35,A36,A37,A39,A40
Subset ID 771	A3,A5,A6,A12,A16,A24,A32,A33,A36,A37,A38,A39,A40
Subset ID 772	A3,A5,A6,A12,A16,A24,A33,A34,A36,A37,A39,A40
Subset ID 773	A3,A5,A6,A12,A16,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 774	A3,A5,A6,A12,A17,A23,A24,A32,A33,A35,A37,A38,A39
Subset ID 775	A3,A5,A6,A12,A17,A23,A24,A32,A33,A35,A37,A39,A40
Subset ID 776	A3,A5,A6,A12,A17,A23,A24,A32,A33,A36,A37,A39,A40
Subset ID 777	A3,A5,A6,A12,A17,A23,A24,A33,A34,A35,A36,A37,A38,A39
Subset ID 778	A3,A5,A6,A12,A17,A23,A24,A33,A34,A35,A37,A39,A40
Subset ID 779	A3,A5,A6,A12,A17,A23,A24,A33,A36,A37,A38,A39,A40,A41
Subset ID 780	A3,A5,A6,A12,A17,A23,A31,A32,A33,A35,A37,A38,A39
Subset ID 781	A3,A5,A6,A12,A17,A23,A31,A32,A33,A35,A37,A39,A40
Subset ID 782	A3,A5,A6,A12,A17,A23,A31,A32,A33,A36,A37,A39,A40
Subset ID 783	A3,A5,A6,A12,A17,A23,A31,A33,A34,A35,A36,A37,A38,A39
Subset ID 784	A3,A5,A6,A12,A17,A23,A31,A33,A34,A35,A37,A39,A40
Subset ID 785	A3,A5,A6,A12,A17,A23,A31,A33,A34,A36,A37,A39,A40
Subset ID 786	A3,A5,A6,A12,A17,A23,A31,A33,A36,A37,A38,A39,A40,A41
Subset ID 787	A3,A5,A6,A12,A17,A24,A25,A29,A32,A33,A35,A37,A39,A40
Subset ID 788	A3,A5,A6,A12,A17,A24,A25,A29,A33,A34,A35,A37,A39,A40
Subset ID 789	A3,A5,A6,A12,A17,A24,A25,A30,A32,A33,A35,A37,A38,A39,A40
Subset ID 790	A3,A5,A6,A12,A17,A24,A25,A30,A33,A34,A35,A37,A38,A39,A40
Subset ID 791	A3,A5,A6,A12,A17,A24,A25,A32,A33,A36,A37,A39,A40
Subset ID 792	A3,A5,A6,A12,A17,A24,A27,A30,A32,A33,A35,A37,A39,A40
Subset ID 793	A3,A5,A6,A12,A17,A24,A27,A30,A33,A34,A35,A37,A39,A40
Subset ID 794	A3,A5,A6,A12,A17,A24,A27,A32,A33,A35,A36,A37,A38,A39
Subset ID 795	A3,A5,A6,A12,A17,A24,A27,A33,A34,A35,A36,A37,A38,A39
Subset ID 796	A3,A5,A6,A12,A17,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 797	A3,A5,A6,A12,A17,A24,A29,A30,A32,A33,A35,A37,A38,A39
Subset ID 798	A3,A5,A6,A12,A17,A24,A29,A30,A32,A33,A35,A37,A39,A40
Subset ID 799	A3,A5,A6,A12,A17,A24,A29,A30,A33,A34,A35,A37,A39,A40
Subset ID 800	A3,A5,A6,A12,A17,A24,A29,A32,A33,A35,A36,A37,A38,A39
Subset ID 801	A3,A5,A6,A12,A17,A24,A29,A32,A33,A35,A37,A38,A39,A40
Subset ID 802	A3,A5,A6,A12,A17,A24,A29,A33,A34,A35,A36,A37,A38,A39
Subset ID 803	A3,A5,A6,A12,A17,A24,A29,A33,A34,A35,A37,A38,A39,A40

Subset ID 804	A3,A5,A6,A12,A17,A24,A29,A33,A36,A37,A38,A39,A40,A41
Subset ID 805	A3,A5,A6,A12,A17,A24,A30,A32,A33,A35,A36,A37,A38,A39
Subset ID 806	A3,A5,A6,A12,A17,A24,A30,A33,A34,A35,A36,A37,A38,A39
Subset ID 807	A3,A5,A6,A12,A17,A24,A30,A33,A36,A37,A38,A39,A40,A41
Subset ID 808	A3,A5,A6,A12,A17,A24,A31,A32,A33,A36,A37,A39,A40
Subset ID 809	A3,A5,A6,A12,A17,A24,A32,A33,A35,A36,A37,A39,A40
Subset ID 810	A3,A5,A6,A12,A17,A24,A32,A33,A36,A37,A38,A39,A40
Subset ID 811	A3,A5,A6,A12,A17,A24,A33,A34,A36,A37,A39,A40
Subset ID 812	A3,A5,A6,A12,A17,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 813	A3,A5,A6,A12,A18,A23,A24,A32,A33,A35,A37,A38,A39
Subset ID 814	A3,A5,A6,A12,A18,A23,A24,A32,A33,A35,A37,A39,A40
Subset ID 815	A3,A5,A6,A12,A18,A23,A24,A32,A33,A36,A37,A39,A40
Subset ID 816	A3,A5,A6,A12,A18,A23,A24,A33,A34,A35,A36,A37,A38,A39
Subset ID 817	A3,A5,A6,A12,A18,A23,A24,A33,A34,A35,A37,A39,A40
Subset ID 818	A3,A5,A6,A12,A18,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 819	A3,A5,A6,A12,A18,A23,A24,A33,A36,A37,A38,A39,A40,A41
Subset ID 820	A3,A5,A6,A12,A18,A23,A31,A32,A33,A35,A37,A38,A39
Subset ID 821	A3,A5,A6,A12,A18,A23,A31,A32,A33,A35,A37,A39,A40
Subset ID 822	A3,A5,A6,A12,A18,A23,A31,A32,A33,A36,A37,A39,A40
Subset ID 823	A3,A5,A6,A12,A18,A23,A31,A33,A34,A35,A36,A37,A38,A39
Subset ID 824	A3,A5,A6,A12,A18,A23,A31,A33,A34,A35,A37,A39,A40
Subset ID 825	A3,A5,A6,A12,A18,A23,A31,A33,A34,A36,A37,A39,A40
Subset ID 826	A3,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 827	A3,A5,A6,A12,A18,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 828	A3,A5,A6,A12,A18,A23,A31,A33,A36,A37,A38,A39,A40,A41
Subset ID 829	A3,A5,A6,A12,A18,A24,A25,A29,A32,A33,A35,A37,A39,A40
Subset ID 830	A3,A5,A6,A12,A18,A24,A25,A29,A33,A34,A35,A37,A39,A40
Subset ID 831	A3,A5,A6,A12,A18,A24,A25,A30,A32,A33,A35,A37,A38,A39,A40
Subset ID 832	A3,A5,A6,A12,A18,A24,A25,A30,A33,A34,A35,A37,A38,A39,A40
Subset ID 833	A3,A5,A6,A12,A18,A24,A25,A32,A33,A36,A37,A39,A40
Subset ID 834	A3,A5,A6,A12,A18,A24,A27,A30,A32,A33,A35,A37,A39,A40
Subset ID 835	A3,A5,A6,A12,A18,A24,A27,A30,A33,A34,A35,A37,A39,A40
Subset ID 836	A3,A5,A6,A12,A18,A24,A27,A32,A33,A35,A36,A37,A38,A39
Subset ID 837	A3,A5,A6,A12,A18,A24,A27,A33,A34,A35,A36,A37,A38,A39
Subset ID 838	A3,A5,A6,A12,A18,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 839	A3,A5,A6,A12,A18,A24,A29,A30,A32,A33,A35,A37,A38,A39
Subset ID 840	A3,A5,A6,A12,A18,A24,A29,A30,A32,A33,A35,A37,A39,A40
Subset ID 841	A3,A5,A6,A12,A18,A24,A29,A30,A33,A34,A35,A37,A39,A40
Subset ID 842	A3,A5,A6,A12,A18,A24,A29,A32,A33,A35,A36,A37,A38,A39
Subset ID 843	A3,A5,A6,A12,A18,A24,A29,A32,A33,A35,A37,A38,A39,A40
Subset ID 844	A3,A5,A6,A12,A18,A24,A29,A33,A34,A35,A36,A37,A38,A39
Subset ID 845	A3,A5,A6,A12,A18,A24,A29,A33,A34,A35,A37,A38,A39,A40
Subset ID 846	A3,A5,A6,A12,A18,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 847	A3,A5,A6,A12,A18,A24,A29,A33,A36,A37,A38,A39,A40,A41
Subset ID 848	A3,A5,A6,A12,A18,A24,A30,A32,A33,A35,A36,A37,A38,A39

Subset ID 849	A3,A5,A6,A12,A18,A24,A30,A33,A34,A35,A36,A37,A38,A39
Subset ID 850	A3,A5,A6,A12,A18,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 851	A3,A5,A6,A12,A18,A24,A30,A33,A36,A37,A38,A39,A40,A41
Subset ID 852	A3,A5,A6,A12,A18,A24,A31,A32,A33,A36,A37,A39,A40
Subset ID 853	A3,A5,A6,A12,A18,A24,A32,A33,A35,A36,A37,A39,A40
Subset ID 854	A3,A5,A6,A12,A18,A24,A32,A33,A36,A37,A38,A39,A40
Subset ID 855	A3,A5,A6,A12,A18,A24,A33,A34,A36,A37,A39,A40
Subset ID 856	A3,A5,A6,A12,A18,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 857	A4,A5,A6,A7,A10,A12,A24,A27,A32,A33,A35,A36,A37,A39,A41
Subset ID 858	A4,A5,A6,A7,A10,A12,A24,A27,A33,A34,A35,A36,A37,A39,A41
Subset ID 859	A4,A5,A6,A7,A12,A17,A24,A27,A32,A33,A35,A36,A37,A39,A41
Subset ID 860	A4,A5,A6,A7,A12,A17,A24,A27,A33,A34,A35,A36,A37,A39,A41
Subset ID 861	A4,A5,A6,A10,A12,A23,A24,A25,A32,A33,A35,A37,A39,A41
Subset ID 862	A4,A5,A6,A10,A12,A23,A24,A27,A32,A33,A35,A37,A39,A41
Subset ID 863	A4,A5,A6,A10,A12,A23,A24,A30,A32,A33,A35,A37,A39,A41
Subset ID 864	A4,A5,A6,A10,A12,A23,A24,A32,A33,A35,A37,A38,A39,A41
Subset ID 865	A4,A5,A6,A10,A12,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 866	A4,A5,A6,A10,A12,A23,A25,A31,A32,A33,A35,A37,A39,A41
Subset ID 867	A4,A5,A6,A10,A12,A23,A27,A31,A32,A33,A35,A37,A39,A41
Subset ID 868	A4,A5,A6,A10,A12,A23,A30,A31,A32,A33,A35,A37,A39,A41
Subset ID 869	A4,A5,A6,A10,A12,A23,A31,A32,A33,A35,A37,A38,A39,A41
Subset ID 870	A4,A5,A6,A10,A12,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 871	A4,A5,A6,A10,A12,A24,A29,A30,A32,A33,A35,A37,A39,A41
Subset ID 872	A4,A5,A6,A10,A12,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 873	A4,A5,A6,A10,A12,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 874	A4,A5,A6,A12,A17,A23,A24,A25,A32,A33,A35,A37,A39,A41
Subset ID 875	A4,A5,A6,A12,A17,A23,A24,A27,A32,A33,A35,A37,A39,A41
Subset ID 876	A4,A5,A6,A12,A17,A23,A24,A30,A32,A33,A35,A37,A39,A41
Subset ID 877	A4,A5,A6,A12,A17,A23,A24,A32,A33,A35,A37,A38,A39,A41
Subset ID 878	A4,A5,A6,A12,A17,A23,A24,A33,A35,A36,A37,A39,A41
Subset ID 879	A4,A5,A6,A12,A17,A23,A25,A31,A32,A33,A35,A37,A39,A41
Subset ID 880	A4,A5,A6,A12,A17,A23,A27,A31,A32,A33,A35,A37,A39,A41
Subset ID 881	A4,A5,A6,A12,A17,A23,A30,A31,A32,A33,A35,A37,A39,A41
Subset ID 882	A4,A5,A6,A12,A17,A23,A31,A32,A33,A35,A37,A38,A39,A41
Subset ID 883	A4,A5,A6,A12,A17,A23,A31,A33,A35,A36,A37,A39,A41
Subset ID 884	A4,A5,A6,A12,A17,A24,A29,A30,A32,A33,A35,A37,A39,A41
Subset ID 885	A4,A5,A6,A12,A17,A24,A29,A33,A35,A36,A37,A39,A41
Subset ID 886	A4,A5,A6,A12,A17,A24,A30,A33,A35,A36,A37,A39,A41
Subset ID 887	A5,A6,A7,A10,A12,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 888	A5,A6,A7,A10,A12,A24,A32,A33,A35,A36,A37,A39,A40,A41
Subset ID 889	A5,A6,A7,A10,A12,A24,A33,A34,A35,A36,A37,A39,A40,A41
Subset ID 890	A5,A6,A7,A10,A12,A24,A33,A35,A36,A37,A38,A39,A40,A41
Subset ID 891	A5,A6,A7,A12,A17,A24,A27,A33,A35,A36,A37,A38,A39,A41
Subset ID 892	A5,A6,A7,A12,A17,A24,A32,A33,A35,A36,A37,A39,A40,A41
Subset ID 893	A5,A6,A7,A12,A17,A24,A33,A34,A35,A36,A37,A39,A40,A41

Subset ID 894	A5,A6,A7,A12,A17,A24,A33,A35,A36,A37,A38,A39,A40,A41
Subset ID 895	A5,A6,A10,A12,A23,A24,A32,A33,A35,A37,A39,A40,A41
Subset ID 896	A5,A6,A10,A12,A23,A24,A33,A34,A35,A37,A39,A40,A41
Subset ID 897	A5,A6,A10,A12,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 898	A5,A6,A10,A12,A23,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 899	A5,A6,A10,A12,A23,A31,A32,A33,A35,A37,A39,A40,A41
Subset ID 900	A5,A6,A10,A12,A23,A31,A33,A34,A35,A37,A39,A40,A41
Subset ID 901	A5,A6,A10,A12,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 902	A5,A6,A10,A12,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 903	A5,A6,A10,A12,A24,A25,A29,A32,A33,A35,A37,A39,A40,A41
Subset ID 904	A5,A6,A10,A12,A24,A25,A29,A33,A34,A35,A37,A39,A40,A41
Subset ID 905	A5,A6,A10,A12,A24,A29,A30,A32,A33,A35,A37,A39,A40,A41
Subset ID 906	A5,A6,A10,A12,A24,A29,A30,A33,A34,A35,A37,A39,A40,A41
Subset ID 907	A5,A6,A10,A12,A24,A29,A32,A33,A35,A37,A38,A39,A40,A41
Subset ID 908	A5,A6,A10,A12,A24,A29,A33,A34,A35,A37,A38,A39,A40,A41
Subset ID 909	A5,A6,A10,A12,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 910	A5,A6,A10,A12,A24,A29,A33,A35,A36,A37,A39,A40,A41
Subset ID 911	A5,A6,A10,A12,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 912	A5,A6,A10,A12,A24,A30,A33,A35,A36,A37,A39,A40,A41
Subset ID 913	A5,A6,A12,A17,A23,A24,A32,A33,A35,A37,A39,A40,A41
Subset ID 914	A5,A6,A12,A17,A23,A24,A33,A34,A35,A37,A39,A40,A41
Subset ID 915	A5,A6,A12,A17,A23,A24,A33,A35,A36,A37,A38,A39,A41
Subset ID 916	A5,A6,A12,A17,A23,A24,A33,A35,A36,A37,A39,A40,A41
Subset ID 917	A5,A6,A12,A17,A23,A31,A32,A33,A35,A37,A39,A40,A41
Subset ID 918	A5,A6,A12,A17,A23,A31,A33,A34,A35,A37,A39,A40,A41
Subset ID 919	A5,A6,A12,A17,A23,A31,A33,A35,A36,A37,A38,A39,A41
Subset ID 920	A5,A6,A12,A17,A23,A31,A33,A35,A36,A37,A39,A40,A41
Subset ID 921	A5,A6,A12,A17,A24,A25,A29,A32,A33,A35,A37,A39,A40,A41
Subset ID 922	A5,A6,A12,A17,A24,A25,A29,A33,A34,A35,A37,A39,A40,A41
Subset ID 923	A5,A6,A12,A17,A24,A29,A30,A32,A33,A35,A37,A39,A40,A41
Subset ID 924	A5,A6,A12,A17,A24,A29,A30,A33,A34,A35,A37,A39,A40,A41
Subset ID 925	A5,A6,A12,A17,A24,A29,A32,A33,A35,A37,A38,A39,A40,A41
Subset ID 926	A5,A6,A12,A17,A24,A29,A33,A34,A35,A37,A38,A39,A40,A41
Subset ID 927	A5,A6,A12,A17,A24,A29,A33,A35,A36,A37,A38,A39,A41
Subset ID 928	A5,A6,A12,A17,A24,A29,A33,A35,A36,A37,A39,A40,A41
Subset ID 929	A5,A6,A12,A17,A24,A30,A33,A35,A36,A37,A38,A39,A41
Subset ID 930	A5,A6,A12,A17,A24,A30,A33,A35,A36,A37,A39,A40,A41

APPENDIX B

The complete results of the experiment given in Section 6.3.2.3 and Table 36.

	Normal (Detection Rate)	DoS Attacks (Detection Rate)	U2R Attacks (Detection Rate)	R2L Attacks (Detection Rate)	Probing Attacks (Detection Rate)	ACCURACY	FALSE ALARM RATE
Subset:1	0,987127	0,966239	0,000000	0,000000	0,644503	0,918557	0,012873
Subset:2	0,986187	0,965178	0,000000	0,001346	0,642343	0,917857	0,013813
Subset:3	0,996930	0,965961	0,000000	0,000000	0,565290	0,919335	0,003070
Subset:4	0,996716	0,963651	0,185714	0,001040	0,621699	0,918426	0,003284
Subset:5	0,996501	0,965630	0,085714	0,000000	0,639462	0,919287	0,003499
Subset:6	0,986946	0,964290	0,000000	0,000000	0,657225	0,918310	0,013054
Subset:7	0,987952	0,966209	0,000000	0,000000	0,631541	0,917792	0,012048
Subset:8	0,996650	0,965622	0,000000	0,000061	0,636582	0,919544	0,003350
Subset:9	0,996815	0,964112	0,000000	0,000122	0,648344	0,920461	0,003185
Subset:10	0,986797	0,964573	0,000000	0,000122	0,686990	0,919008	0,013203
Subset:11	0,995478	0,963999	0,000000	0,000000	0,648104	0,918715	0,004522
Subset:12	0,986236	0,964134	0,000000	0,000306	0,594575	0,918419	0,013764
Subset:13	0,983727	0,966979	0,000000	0,043494	0,690350	0,920998	0,016273
Subset:14	0,981731	0,968958	0,042857	0,000245	0,666827	0,922168	0,018269
Subset:15	0,981995	0,968258	0,000000	0,000367	0,724196	0,919053	0,018005
Subset:16	0,983595	0,971447	0,000000	0,000367	0,716755	0,921564	0,016405
Subset:17	0,982770	0,964804	0,000000	0,000061	0,683869	0,918516	0,017230
Subset:18	0,983298	0,967397	0,000000	0,019392	0,707393	0,919892	0,016702
Subset:19	0,983166	0,967971	0,000000	0,000061	0,644983	0,919866	0,016834
Subset:20	0,983546	0,968441	0,000000	0,003242	0,731877	0,920766	0,016454
Subset:21	0,983546	0,969889	0,085714	0,000856	0,695631	0,920165	0,016454
Subset:22	0,983678	0,965030	0,000000	0,000122	0,624820	0,917844	0,016322
Subset:23	0,983480	0,967845	0,171429	0,001468	0,692031	0,921403	0,016520
Subset:24	0,984124	0,963825	0,000000	0,000000	0,628181	0,916670	0,015876
Subset:25	0,983298	0,967884	0,000000	0,000551	0,745799	0,919940	0,016702
Subset:26	0,983628	0,970799	0,000000	0,000795	0,677628	0,924145	0,016372
Subset:27	0,972654	0,966635	0,000000	0,000000	0,656505	0,917021	0,027346
Subset:28	0,983827	0,969715	0,028571	0,000367	0,717235	0,921435	0,016173
Subset:29	0,983496	0,968193	0,000000	0,006607	0,713394	0,920773	0,016504
Subset:30	0,983612	0,968162	0,000000	0,016823	0,650024	0,920377	0,016388
Subset:31	0,983876	0,968254	0,000000	0,000000	0,688670	0,920075	0,016124
Subset:32	0,982358	0,968919	0,000000	0,007218	0,678829	0,919519	0,017642
Subset:33	0,982935	0,966431	0,000000	0,000000	0,656985	0,917024	0,017065
Subset:34	0,981285	0,968415	0,000000	0,017496	0,649784	0,920204	0,018715
Subset:35	0,981813	0,967723	0,000000	0,009054	0,621459	0,918532	0,018187
Subset:36	0,983562	0,959287	0,000000	0,010338	0,668507	0,915728	0,016438
Subset:37	0,983414	0,969850	0,000000	0,000245	0,582333	0,918779	0,016586
Subset:38	0,982011	0,966191	0,000000	0,000061	0,689870	0,917561	0,017989
Subset:39	0,982721	0,967971	0,000000	0,003793	0,703553	0,920625	0,017279
Subset:40	0,982655	0,964791	0,000000	0,000000	0,646183	0,918066	0,017345
Subset:41	0,982754	0,969807	0,000000	0,000000	0,677628	0,920358	0,017246
Subset:42	0,983397	0,970877	0,000000	0,000551	0,690590	0,922499	0,016603
Subset:43	0,984916	0,964121	0,000000	0,000061	0,622660	0,917654	0,015084
Subset:44	0,982886	0,968884	0,028571	0,003303	0,699232	0,919583	0,017114
Subset:45	0,978793	0,965913	0,000000	0,011868	0,717235	0,917435	0,021207
Subset:46	0,983628	0,967314	0,200000	0,000979	0,679789	0,920387	0,016372
Subset:47	0,983876	0,969585	0,000000	0,007708	0,703072	0,920223	0,016124
Subset:48	0,983628	0,971103	0,000000	0,001162	0,649304	0,919824	0,016372
Subset:49	0,978875	0,964295	0,128571	0,000489	0,684590	0,918252	0,021125
Subset:50	0,982770	0,967618	0,000000	0,000000	0,699712	0,919705	0,017230
Subset:51	0,983480	0,967453	0,014286	0,000000	0,634902	0,920126	0,016520
Subset:52	0,984074	0,966474	0,000000	0,028262	0,668267	0,921956	0,015926
Subset:53	0,984025	0,970977	0,000000	0,000000	0,698512	0,923004	0,015975
Subset:54	0,982952	0,966287	0,000000	0,000122	0,645223	0,917432	0,017048

Subset:55	0,984173	0,969959	0,000000	0,000612	0,686750	0,920953	0,015827
Subset:56	0,983546	0,965482	0,028571	0,000184	0,644983	0,918509	0,016454
Subset:57	0,983876	0,968793	0,100000	0,000428	0,785886	0,920573	0,016124
Subset:58	0,984602	0,968719	0,028571	0,000673	0,673548	0,919335	0,015398
Subset:59	0,978974	0,965622	0,000000	0,000367	0,704273	0,918313	0,021026
Subset:60	0,984289	0,969524	0,057143	0,003303	0,764762	0,920152	0,015711
Subset:61	0,983496	0,968441	0,000000	0,000184	0,702592	0,920380	0,016504
Subset:62	0,984025	0,967384	0,000000	0,000245	0,664186	0,919908	0,015975
Subset:63	0,983876	0,968502	0,000000	0,000367	0,736438	0,919467	0,016124
Subset:64	0,983744	0,969776	0,000000	0,013275	0,705473	0,920692	0,016256
Subset:65	0,982886	0,964382	0,000000	0,000000	0,713634	0,917017	0,017114
Subset:66	0,984140	0,967966	0,000000	0,004527	0,666347	0,919564	0,015860
Subset:67	0,983051	0,968985	0,000000	0,000000	0,676908	0,921348	0,016949
Subset:68	0,984388	0,968732	0,000000	0,000551	0,660586	0,920618	0,015612
Subset:69	0,983117	0,969728	0,000000	0,000061	0,708593	0,919506	0,016883
Subset:70	0,984206	0,822565	0,000000	0,000061	0,635622	0,811583	0,015794
Subset:71	0,983860	0,967988	0,000000	0,003854	0,690110	0,920191	0,016140
Subset:72	0,982176	0,964221	0,000000	0,000000	0,550408	0,916021	0,017824
Subset:73	0,984569	0,968388	0,000000	0,000918	0,706673	0,919879	0,015431
Subset:74	0,984322	0,970999	0,000000	0,000673	0,671867	0,921756	0,015678
Subset:75	0,977654	0,966714	0,000000	0,000000	0,648104	0,916947	0,022346
Subset:76	0,983513	0,968545	0,014286	0,006484	0,725876	0,919120	0,016487
Subset:77	0,983678	0,966948	0,000000	0,022940	0,723236	0,919670	0,016322
Subset:78	0,981136	0,966178	0,000000	0,007708	0,664186	0,917956	0,018864
Subset:79	0,983926	0,967640	0,000000	0,000061	0,678349	0,919339	0,016074
Subset:80	0,983529	0,969863	0,000000	0,000673	0,719155	0,920101	0,016471
Subset:81	0,982457	0,964308	0,000000	0,000000	0,689630	0,917397	0,017543
Subset:82	0,984140	0,966757	0,028571	0,002019	0,715795	0,918805	0,015860
Subset:83	0,978512	0,967993	0,000000	0,000000	0,652664	0,919635	0,021488
Subset:84	0,981450	0,968284	0,000000	0,000245	0,721315	0,920413	0,018550
Subset:85	0,983414	0,970960	0,000000	0,000000	0,699712	0,920522	0,016586
Subset:86	0,983810	0,966365	0,042857	0,000122	0,635862	0,917943	0,016190
Subset:87	0,983744	0,968410	0,000000	0,000245	0,680749	0,919146	0,016256
Subset:88	0,984520	0,964756	0,000000	0,000000	0,669467	0,917120	0,015480
Subset:89	0,983926	0,968845	0,000000	0,000000	0,753961	0,920930	0,016074
Subset:90	0,983612	0,968241	0,000000	0,000367	0,683629	0,919705	0,016388
Subset:91	0,984223	0,966213	0,000000	0,000184	0,691551	0,918963	0,015777
Subset:92	0,983364	0,972108	0,014286	0,000734	0,664906	0,921454	0,016636
Subset:93	0,983777	0,966439	0,000000	0,003915	0,699952	0,919046	0,016223
Subset:94	0,983727	0,967101	0,000000	0,009176	0,654345	0,919712	0,016273
Subset:95	0,979783	0,967788	0,000000	0,000000	0,725156	0,918828	0,020217
Subset:96	0,983744	0,968541	0,000000	0,003609	0,711234	0,918824	0,016256
Subset:97	0,983150	0,965465	0,000000	0,000000	0,716995	0,918133	0,016850
Subset:98	0,985081	0,967040	0,000000	0,029118	0,692031	0,921451	0,014919
Subset:99	0,983348	0,968084	0,000000	0,000306	0,640903	0,921126	0,016652
Subset:100	0,983183	0,969119	0,000000	0,000000	0,711234	0,921593	0,016817
Subset:101	0,984124	0,969524	0,000000	0,000000	0,729237	0,919712	0,015876
Subset:102	0,982589	0,965421	0,014286	0,000000	0,672348	0,916837	0,017411
Subset:103	0,983282	0,968576	0,014286	0,046736	0,685310	0,922387	0,016718
Subset:104	0,859258	0,075766	0,000000	0,000000	0,812050	0,314299	0,140742
Subset:105	0,982242	0,968280	0,000000	0,003670	0,681949	0,919795	0,017758
Subset:106	0,983860	0,967384	0,100000	0,002508	0,700192	0,919030	0,016140
Subset:107	0,983711	0,966418	0,014286	0,000061	0,683389	0,918143	0,016289
Subset:108	0,983959	0,968880	0,014286	0,000734	0,715074	0,919127	0,016041
Subset:109	0,983744	0,895812	0,000000	0,000000	0,680029	0,864945	0,016256
Subset:110	0,984916	0,966322	0,071429	0,001346	0,699472	0,916702	0,015084
Subset:111	0,985592	0,966757	0,014286	0,001958	0,578493	0,918966	0,014408
Subset:112	0,984635	0,966196	0,000000	0,000000	0,635142	0,916786	0,015365
Subset:113	0,994174	0,966270	0,000000	0,000000	0,706913	0,917882	0,005826
Subset:114	0,983694	0,968053	0,071429	0,000122	0,655065	0,921679	0,016306
Subset:115	0,984685	0,969480	0,000000	0,002263	0,322612	0,923284	0,015315
Subset:116	0,985081	0,963742	0,071429	0,001774	0,708113	0,918207	0,014919
Subset:117	0,985460	0,964651	0,128571	0,000551	0,644983	0,917705	0,014540
Subset:118	0,981582	0,967475	0,000000	0,000122	0,706673	0,917416	0,018418

Subset:119	0,984223	0,967897	0,000000	0,000122	0,219875	0,918824	0,015777
Subset:120	0,985130	0,967723	0,000000	0,000122	0,650504	0,917451	0,014870
Subset:121	0,984305	0,969093	0,228571	0,001835	0,463514	0,924756	0,015695
Subset:122	0,984668	0,966940	0,157143	0,000428	0,721075	0,919335	0,015332
Subset:123	0,984800	0,926566	0,000000	0,000061	0,683629	0,886985	0,015200
Subset:124	0,984025	0,911226	0,000000	0,002630	0,768843	0,880259	0,015975
Subset:125	0,984074	0,967240	0,028571	0,008014	0,580413	0,917117	0,015926
Subset:126	0,984536	0,966687	0,000000	0,000000	0,684349	0,918059	0,015464
Subset:127	0,985279	0,832832	0,000000	0,000000	0,698752	0,817782	0,014721
Subset:128	0,990543	0,966635	0,028571	0,000367	0,628661	0,920503	0,009457
Subset:129	0,985031	0,967083	0,000000	0,000061	0,691311	0,917513	0,014969
Subset:130	0,985064	0,968310	0,000000	0,000061	0,666347	0,917667	0,014936
Subset:131	0,984784	0,964434	0,000000	0,000367	0,650984	0,918821	0,015216
Subset:132	0,984619	0,965391	0,000000	0,000061	0,649544	0,919615	0,015381
Subset:133	0,984784	0,965722	0,057143	0,008381	0,740759	0,918969	0,015216
Subset:134	0,983843	0,838097	0,042857	0,000918	0,651464	0,823068	0,016157
Subset:135	0,980146	0,968410	0,000000	0,000000	0,692511	0,918467	0,019854
Subset:136	0,983678	0,965991	0,057143	0,009604	0,220115	0,918525	0,016322
Subset:137	0,984652	0,967162	0,000000	0,000122	0,682189	0,917426	0,015348
Subset:138	0,984058	0,964891	0,000000	0,000245	0,550168	0,919744	0,015942
Subset:139	0,983893	0,967244	0,014286	0,000856	0,737878	0,918937	0,016107
Subset:140	0,987078	0,966257	0,000000	0,001652	0,546087	0,915217	0,012922
Subset:141	0,990989	0,902029	0,000000	0,000367	0,716995	0,873462	0,009011
Subset:142	0,985097	0,966640	0,000000	0,000000	0,608017	0,916561	0,014903
Subset:143	0,984751	0,831331	0,000000	0,000000	0,638022	0,816712	0,015249
Subset:144	0,985081	0,965856	0,000000	0,000122	0,676188	0,915963	0,014919
Subset:145	0,985774	0,966583	0,000000	0,000245	0,692271	0,920329	0,014226
Subset:146	0,984982	0,829957	0,000000	0,000000	0,688190	0,816229	0,015018
Subset:147	0,985988	0,830935	0,000000	0,000000	0,667787	0,816146	0,014012
Subset:148	0,993267	0,965748	0,185714	0,000122	0,645703	0,919078	0,006733
Subset:149	0,984140	0,966061	0,342857	0,006179	0,236678	0,919692	0,015860
Subset:150	0,984982	0,964020	0,185714	0,002997	0,708593	0,916754	0,015018
Subset:151	0,983843	0,963729	0,057143	0,002386	0,711234	0,916625	0,016157
Subset:152	0,984850	0,967884	0,000000	0,000122	0,695631	0,917779	0,015150
Subset:153	0,982292	0,966526	0,242857	0,000122	0,245079	0,919377	0,017708
Subset:154	0,984932	0,966796	0,014286	0,000000	0,702352	0,917291	0,015068
Subset:155	0,983183	0,965861	0,300000	0,000000	0,434470	0,920425	0,016817
Subset:156	0,983942	0,967592	0,071429	0,000612	0,727316	0,918937	0,016058
Subset:157	0,986566	0,863382	0,014286	0,001652	0,657465	0,840153	0,013434
Subset:158	0,984883	0,963933	0,114286	0,011011	0,760682	0,917763	0,015117
Subset:159	0,984701	0,965052	0,142857	0,000122	0,618099	0,916927	0,015299
Subset:160	0,984223	0,966518	0,000000	0,001101	0,695391	0,916866	0,015777
Subset:161	0,985427	0,966030	0,000000	0,000000	0,664426	0,916580	0,014573
Subset:162	0,984404	0,966283	0,014286	0,000122	0,686990	0,920213	0,015596
Subset:163	0,983893	0,966644	0,000000	0,000000	0,673788	0,916686	0,016107
Subset:164	0,985675	0,845902	0,000000	0,000000	0,697312	0,827476	0,014325
Subset:165	0,984602	0,964930	0,000000	0,000061	0,632021	0,918201	0,015398
Subset:166	0,984239	0,965099	0,000000	0,002202	0,655545	0,919393	0,015761
Subset:167	0,983463	0,966004	0,014286	0,005873	0,789006	0,918709	0,016537
Subset:168	0,984734	0,966422	0,085714	0,005016	0,654825	0,918107	0,015266
Subset:169	0,982754	0,968341	0,000000	0,000000	0,723716	0,919133	0,017246
Subset:170	0,983117	0,968741	0,000000	0,000734	0,228997	0,918818	0,016883
Subset:171	0,983876	0,967610	0,000000	0,000428	0,661306	0,917969	0,016124
Subset:172	0,984487	0,964442	0,014286	0,000306	0,474796	0,918792	0,015513
Subset:173	0,984668	0,919435	0,071429	0,002019	0,652184	0,883156	0,015332
Subset:174	0,985411	0,965552	0,042857	0,000061	0,652184	0,915500	0,014589
Subset:175	0,984206	0,967031	0,014286	0,003670	0,756361	0,919091	0,015794
Subset:176	0,984784	0,966709	0,000000	0,000306	0,614738	0,916551	0,015216
Subset:177	0,983992	0,967645	0,000000	0,000000	0,687710	0,918278	0,016008
Subset:178	0,985724	0,824557	0,000000	0,000000	0,687710	0,811667	0,014276
Subset:179	0,982803	0,965909	0,042857	0,007096	0,240278	0,919403	0,017197
Subset:180	0,985724	0,965922	0,000000	0,000000	0,660346	0,916982	0,014276
Subset:181	0,992458	0,954601	0,000000	0,000061	0,608737	0,908301	0,007542
Subset:182	0,983051	0,965217	0,000000	0,000367	0,645463	0,918307	0,016949

Subset:183	0,984289	0,965347	0,000000	0,000061	0,541047	0,918416	0,015711
Subset:184	0,984883	0,966283	0,000000	0,000367	0,649784	0,917107	0,015117
Subset:185	0,984338	0,964682	0,000000	0,002936	0,694671	0,918657	0,015662
Subset:186	0,985262	0,968149	0,000000	0,000061	0,683149	0,918535	0,014738
Subset:187	0,983529	0,965252	0,042857	0,000245	0,232837	0,918111	0,016471
Subset:188	0,984140	0,966535	0,000000	0,000000	0,693231	0,917416	0,015860
Subset:189	0,984421	0,962607	0,000000	0,000428	0,499280	0,917686	0,015579
Subset:190	0,986352	0,968758	0,028571	0,000184	0,691791	0,919278	0,013648
Subset:191	0,985856	0,965352	0,014286	0,000061	0,658425	0,915452	0,014144
Subset:192	0,984437	0,963137	0,028571	0,000673	0,720835	0,917795	0,015563
Subset:193	0,985147	0,966283	0,014286	0,011439	0,577532	0,917320	0,014853
Subset:194	0,984619	0,966435	0,000000	0,000000	0,682669	0,917210	0,015381
Subset:195	0,984338	0,966309	0,000000	0,000061	0,710034	0,917548	0,015662
Subset:196	0,984058	0,965308	0,171429	0,004833	0,575132	0,918561	0,015942
Subset:197	0,985229	0,876834	0,000000	0,000000	0,655545	0,850689	0,014771
Subset:198	0,985064	0,966687	0,000000	0,000000	0,700912	0,916532	0,014936
Subset:199	0,988282	0,965443	0,014286	0,001835	0,611618	0,919245	0,011718
Subset:200	0,984668	0,965025	0,000000	0,000367	0,412386	0,919130	0,015332
Subset:201	0,984322	0,822617	0,157143	0,008503	0,75228	0,814429	0,015678
Subset:202	0,984767	0,928067	0	0,000612	0,654105	0,888888	0,015233
Subset:203	0,984701	0,968249	0	0,000061	0,706913	0,918442	0,015299
Subset:204	0,983067	0,96868	0	0,000673	0,253961	0,919364	0,016933
Subset:205	0,98386	0,967357	0	0,003548	0,722516	0,918043	0,01614
Subset:206	0,984817	0,965408	0,185714	0,000306	0,468555	0,919744	0,015183
Subset:207	0,985015	0,965726	0,014286	0,000122	0,718435	0,918786	0,014985
Subset:208	0,985708	0,823687	0	0,00104	0,362458	0,806658	0,014292
Subset:209	0,985064	0,964791	0,128571	0,000551	0,661306	0,917133	0,014936
Subset:210	0,984569	0,966057	0	0	0,597696	0,917554	0,015431
Subset:211	0,985625	0,967196	0	0,000979	0,606097	0,91867	0,014375
Subset:212	0,992144	0,967083	0	0,000061	0,62362	0,92139	0,007856
Subset:213	0,986253	0,965521	0,014286	0,000918	0,62794	0,918844	0,013747
Subset:214	0,986104	0,896451	0,114286	0,000306	0,638262	0,867495	0,013896
Subset:215	0,986352	0,966557	0	0,000245	0,638982	0,918577	0,013648
Subset:216	0,98584	0,968593	0	0,011562	0,621699	0,920396	0,01416
Subset:217	0,985394	0,967414	0	0,000306	0,628421	0,919307	0,014606
Subset:218	0,986418	0,943077	0	0	0,647384	0,901479	0,013582
Subset:219	0,984322	0,820372	0	0	0,717475	0,813294	0,015678
Subset:220	0,985048	0,827712	0	0,000428	0,652904	0,816564	0,014952
Subset:221	0,984949	0,826328	0	0,000061	0,713154	0,813783	0,015051
Subset:222	0,984883	0,823426	0,028571	0	0,705473	0,814168	0,015117
Subset:223	0,985163	0,823966	0	0,003976	0,722756	0,813824	0,014837
Subset:224	0,984404	0,826198	0	0,000122	0,62794	0,814963	0,015596
Subset:225	0,984256	0,826624	0	0	0,7494	0,815567	0,015744
Subset:226	0,982308	0,82575	0,028571	0	0,708353	0,815336	0,017692
Subset:227	0,985295	0,82524	0,014286	0,000184	0,795007	0,814963	0,014705
Subset:228	0,984916	0,822835	0	0,001591	0,698992	0,812606	0,015084
Subset:229	0,979816	0,963303	0	0,005077	0,727076	0,917538	0,020184
Subset:230	0,986451	0,823835	0	0,000061	0,62602	0,81504	0,013549
Subset:231	0,985312	0,825615	0	0	0,629381	0,815091	0,014688
Subset:232	0,985477	0,822495	0	0	0,715795	0,813596	0,014523
Subset:233	0,98452	0,826824	0	0	0,713874	0,815393	0,01548
Subset:234	0,983678	0,822234	0	0,002263	0,693951	0,813741	0,016322
Subset:235	0,984817	0,823592	0	0	0,675468	0,814654	0,015183
Subset:236	0,984404	0,968689	0	0,000061	0,657705	0,919663	0,015596
Subset:237	0,996468	0,254032	0	0	0,672348	0,393356	0,003532
Subset:238	0,984701	0,824118	0	0,000122	0,721795	0,81422	0,015299
Subset:239	0,984998	0,824331	0	0,000061	0,62794	0,815136	0,015002
Subset:240	0,985873	0,825554	0	0	0,614018	0,81441	0,014127
Subset:241	0,984322	0,825084	0	0,00104	0,722276	0,814612	0,015678
Subset:242	0,984173	0,826946	0	0	0,712674	0,815342	0,015827
Subset:243	0,984586	0,8229	0	0,000734	0,675228	0,813529	0,015414
Subset:244	0,978908	0,954393	0	0,004955	0,708353	0,908571	0,021092
Subset:245	0,9831	0,968036	0	0,00263	0,714114	0,9185	0,0169
Subset:246	0,983232	0,825784	0	0	0,715314	0,814609	0,016768

Subset:247	0,985015	0,82176	0	0	0,782285	0,815403	0,014985
Subset:248	0,984932	0,968036	0	0,000979	0,669707	0,919998	0,015068
Subset:249	0,988134	0,968145	0	0,000122	0,708353	0,91948	0,011866
Subset:250	0,985576	0,823557	0,042857	0,000489	0,721795	0,813879	0,014424
Subset:251	0,984767	0,823974	0	0,004649	0,724676	0,814085	0,015233
Subset:252	0,984157	0,822678	0	0,000122	0,638742	0,814467	0,015843
Subset:253	0,984289	0,82524	0,071429	0,000306	0,731877	0,815824	0,015711
Subset:254	0,98249	0,823178	0,028571	0,000367	0,719635	0,815766	0,01751
Subset:255	0,984305	0,825832	0	0	0,701632	0,814985	0,015695
Subset:256	0,979569	0,964068	0,057143	0	0,682909	0,916805	0,020431
Subset:257	0,985031	0,823883	0	0	0,709554	0,813722	0,014969
Subset:258	0,984503	0,966692	0,128571	0	0,673548	0,917667	0,015497
Subset:259	0,98452	0,823352	0,128571	0,000061	0,641863	0,813091	0,01548
Subset:260	0,982952	0,824805	0,085714	0	0,703313	0,813853	0,017048
Subset:261	0,985031	0,824336	0,028571	0,000367	0,635142	0,815667	0,014969
Subset:262	0,984388	0,824745	0,242857	0,000856	0,646183	0,814728	0,015612
Subset:263	0,984718	0,823148	0	0	0,703553	0,81359	0,015282
Subset:264	0,98419	0,826067	0	0,000306	0,728997	0,814394	0,01581
Subset:265	0,984536	0,819424	0,014286	0	0,665867	0,813747	0,015464
Subset:266	0,985312	0,823448	0	0,000061	0,658425	0,814185	0,014688
Subset:267	0,991814	0,910369	0,014286	0,000673	0,612098	0,877384	0,008186
Subset:268	0,98485	0,824801	0,014286	0	0,794767	0,815078	0,01515
Subset:269	0,985312	0,825214	0	0	0,760202	0,814744	0,014688
Subset:270	0,992606	0,969163	0,014286	0,000489	0,657225	0,921859	0,007394
Subset:271	0,987061	0,896516	0,114286	0,000184	0,650984	0,866495	0,012939
Subset:272	0,985856	0,823274	0	0,000367	0,719635	0,812403	0,014144
Subset:273	0,985922	0,823331	0	0	0,701152	0,814657	0,014078
Subset:274	0,980014	0,965252	0	0	0,663946	0,917982	0,019986
Subset:275	0,985394	0,824309	0	0,000061	0,62674	0,814519	0,014606
Subset:276	0,984454	0,824301	0	0,000061	0,755641	0,816162	0,015546
Subset:277	0,981912	0,828638	0	0	0,790446	0,818033	0,018088
Subset:278	0,98579	0,820633	0	0,000122	0,659866	0,812921	0,01421
Subset:279	0,984041	0,823931	0	0	0,68987	0,813448	0,015959
Subset:280	0,984206	0,968863	0	0,002936	0,718675	0,919519	0,015794
Subset:281	0,977291	0,96389	0	0,000061	0,665867	0,915808	0,022709
Subset:282	0,985097	0,823161	0	0	0,715074	0,813554	0,014903
Subset:283	0,996122	0,252283	0	0,000306	0,646183	0,394767	0,003878
Subset:284	0,98183	0,838427	0	0,000489	0,663946	0,823225	0,01817
Subset:285	0,981334	0,869604	0	0	0,697312	0,846847	0,018666
Subset:286	0,982572	0,826485	0	0	0,717235	0,815361	0,017428
Subset:287	0,984437	0,822339	0	0,000061	0,640182	0,812992	0,015563
Subset:288	0,986748	0,82119	0	0	0,656745	0,812362	0,013252
Subset:289	0,983942	0,969185	0	0,000245	0,708113	0,919117	0,016058
Subset:290	0,984503	0,825541	0	0	0,785886	0,816323	0,015497
Subset:291	0,984157	0,826298	0	0,002508	0,783245	0,815226	0,015843
Subset:292	0,986253	0,96808	0,014286	0,000489	0,664186	0,919567	0,013747
Subset:293	0,986566	0,850148	0	0,000122	0,720355	0,831903	0,013434
Subset:294	0,982853	0,940162	0	0,000061	0,707153	0,898987	0,017147
Subset:295	0,985345	0,826115	0	0	0,662746	0,815252	0,014655
Subset:296	0,984569	0,824057	0	0,000428	0,714834	0,814927	0,015431
Subset:297	0,986649	0,82146	0	0	0,677388	0,813246	0,013351
Subset:298	0,984503	0,823731	0	0	0,699472	0,814123	0,015497
Subset:299	0,984652	0,826067	0,028571	0,000061	0,670427	0,815014	0,015348
Subset:300	0,984635	0,826989	0	0	0,68507	0,814863	0,015365
Subset:301	0,982044	0,875299	0	0	0,695391	0,851162	0,017956
Subset:302	0,985889	0,825497	0	0	0,760682	0,814975	0,014111
Subset:303	0,983084	0,942159	0	0,000061	0,671147	0,901861	0,016916
Subset:304	0,986302	0,824623	0	0	0,68939	0,813953	0,013698
Subset:305	0,985576	0,822795	0	0,000856	0,638502	0,814696	0,014424
Subset:306	0,985427	0,825676	0	0,000489	0,69059	0,81485	0,014573
Subset:307	0,985229	0,824827	0	0	0,720355	0,814361	0,014771
Subset:308	0,984173	0,827503	0,071429	0,000061	0,756121	0,816673	0,015827
Subset:309	0,980988	0,937347	0	0,000918	0,68963	0,897293	0,019012
Subset:310	0,985724	0,822234	0	0	0,674028	0,812693	0,014276

Subset:311	0,985048	0,968393	0	0,000979	0,642823	0,919123	0,014952
Subset:312	0,984734	0,823039	0,028571	0,000061	0,673548	0,814262	0,015266
Subset:313	0,984883	0,825045	0	0,000061	0,700192	0,813394	0,015117
Subset:314	0,985774	0,823809	0	0,000734	0,633461	0,815201	0,014226
Subset:315	0,984751	0,824349	0	0	0,632981	0,81494	0,015249
Subset:316	0,985988	0,820337	0	0	0,651704	0,812657	0,014012
Subset:317	0,984157	0,827024	0	0	0,710034	0,814435	0,015843
Subset:318	0,98386	0,823239	0	0,000122	0,683149	0,813114	0,01614
Subset:319	0,984586	0,823135	0	0,000122	0,691791	0,813329	0,015414
Subset:320	0,984371	0,827799	0,014286	0,001162	0,650504	0,815017	0,015629
Subset:321	0,984041	0,825863	0	0,004466	0,75012	0,815316	0,015959
Subset:322	0,984487	0,828029	0	0	0,783965	0,816525	0,015513
Subset:323	0,990477	0,969972	0	0,000184	0,647144	0,921374	0,009523
Subset:324	0,991468	0,96908	0	0,000061	0,647864	0,920007	0,008532
Subset:325	0,986319	0,949259	0,014286	0,000367	0,616179	0,906575	0,013681
Subset:326	0,989091	0,96553	0	0,000184	0,637782	0,918609	0,010909
Subset:327	0,98749	0,966261	0	0	0,661066	0,920274	0,01251
Subset:328	0,984701	0,964025	0	0,000122	0,620979	0,916853	0,015299
Subset:329	0,990989	0,826942	0,057143	0,001896	0,638022	0,817245	0,009011
Subset:330	0,984635	0,963886	0,114286	0,004343	0,650504	0,916706	0,015365
Subset:331	0,98622	0,9664	0	0,001407	0,647384	0,919419	0,01378
Subset:332	0,984734	0,963851	0,171429	0,000122	0,633941	0,917233	0,015266
Subset:333	0,985576	0,964042	0	0,007035	0,62482	0,916841	0,014424
Subset:334	0,99155	0,966196	0,042857	0,000367	0,658185	0,919705	0,00845
Subset:335	0,98655	0,96804	0	0	0,642103	0,92084	0,01345
Subset:336	0,988002	0,966874	0	0,008809	0,6229	0,91893	0,011998
Subset:337	0,985411	0,965909	0,085714	0,002753	0,646663	0,918438	0,014589
Subset:338	0,991319	0,965917	0	0,002263	0,635622	0,920278	0,008681
Subset:339	0,990131	0,82923	0	0	0,632741	0,818821	0,009869
Subset:340	0,985279	0,966035	0	0,000122	0,6241	0,916828	0,014721
Subset:341	0,981136	0,92711	0,028571	0,000061	0,659386	0,891595	0,018864
Subset:342	0,984569	0,822569	0	0	0,741719	0,81432	0,015431
Subset:343	0,984635	0,967714	0	0	0,642343	0,91976	0,015365
Subset:344	0,984734	0,967497	0,014286	0,000122	0,642583	0,919252	0,015266
Subset:345	0,985163	0,817592	0	0,000061	0,631061	0,81366	0,014837
Subset:346	0,984652	0,821904	0	0	0,692991	0,813792	0,015348
Subset:347	0,985081	0,968258	0	0	0,671147	0,918712	0,014919
Subset:348	0,983133	0,968067	0,014286	0,001468	0,671147	0,918339	0,016867
Subset:349	0,984734	0,823052	0	0,000306	0,669947	0,814185	0,015266
Subset:350	0,985295	0,823874	0,014286	0	0,663466	0,814795	0,014705
Subset:351	0,982952	0,824466	0	0,000245	0,776284	0,815802	0,017048
Subset:352	0,983794	0,82518	0	0,007524	0,81517	0,816226	0,016206
Subset:353	0,985741	0,823531	0,014286	0	0,675468	0,813725	0,014259
Subset:354	0,982622	0,824871	0	0,000122	0,692991	0,813808	0,017378
Subset:355	0,985543	0,963781	0	0,000061	0,675468	0,918262	0,014457
Subset:356	0,985196	0,967845	0	0,000061	0,646663	0,919593	0,014804
Subset:357	0,987919	0,919096	0,014286	0,000306	0,68483	0,88538	0,012081
Subset:358	0,983827	0,965704	0	0,001529	0,667067	0,918737	0,016173
Subset:359	0,983051	0,823252	0,014286	0,000122	0,683629	0,815201	0,016949
Subset:360	0,984091	0,822769	0	0	0,792127	0,815689	0,015909
Subset:361	0,98386	0,835525	0,085714	0,00312	0,702112	0,821538	0,01614
Subset:362	0,984404	0,966418	0	0,000061	0,62794	0,916654	0,015596
Subset:363	0,98655	0,820624	0	0,000061	0,614498	0,812632	0,01345
Subset:364	0,983414	0,823583	0	0	0,703072	0,814451	0,016586
Subset:365	0,985361	0,906714	0	0,000245	0,723476	0,873931	0,014639
Subset:366	0,984586	0,9661	0,171429	0,000245	0,615939	0,918612	0,015414
Subset:367	0,983992	0,823357	0	0	0,718195	0,81439	0,016008
Subset:368	0,984635	0,822987	0,085714	0,000612	0,681949	0,814085	0,015365
Subset:369	0,985625	0,824953	0,014286	0	0,649064	0,815577	0,014375
Subset:370	0,985361	0,825667	0,071429	0,000184	0,782285	0,815721	0,014639
Subset:371	0,984751	0,822795	0,042857	0,000918	0,681469	0,814937	0,015249
Subset:372	0,979354	0,828712	0,028571	0,000306	0,737398	0,817345	0,020646
Subset:373	0,985394	0,964904	0,157143	0,000061	0,679309	0,918191	0,014606
Subset:374	0,983562	0,825275	0,171429	0,000918	0,661306	0,813632	0,016438

Subset:375	0,989702	0,965291	0,085714	0,006974	0,620739	0,917998	0,010298
Subset:376	0,983265	0,964973	0,171429	0,003242	0,706433	0,918744	0,016735
Subset:377	0,985147	0,820825	0	0,000367	0,675228	0,814622	0,014853
Subset:378	0,984487	0,820986	0	0	0,680269	0,814699	0,015513
Subset:379	0,984998	0,967584	0	0,002386	0,695391	0,918557	0,015002
Subset:380	0,986863	0,966605	0	0,003365	0,659626	0,919191	0,013137
Subset:381	0,984734	0,821786	0	0	0,615939	0,813519	0,015266
Subset:382	0,977555	0,966204	0	0	0,695151	0,918194	0,022445
Subset:383	0,98452	0,96821	0	0,003609	0,693711	0,919522	0,01548
Subset:384	0,983744	0,965474	0,014286	0,000245	0,636342	0,918101	0,016256
Subset:385	0,983628	0,823661	0	0,004527	0,693951	0,814885	0,016372
Subset:386	0,985658	0,822735	0	0,000734	0,677388	0,81459	0,014342
Subset:387	0,983232	0,826106	0,028571	0,001529	0,641143	0,814329	0,016768
Subset:388	0,982176	0,826198	0	0,000612	0,728036	0,815631	0,017824
Subset:389	0,985213	0,82173	0	0,017496	0,668507	0,815062	0,014787
Subset:390	0,986847	0,822043	0	0,000184	0,668267	0,812178	0,013153
Subset:391	0,98513	0,852393	0	0,000184	0,708353	0,835732	0,01487
Subset:392	0,990296	0,964573	0	0,000061	0,654585	0,920229	0,009704
Subset:393	0,98622	0,966422	0,014286	0,002019	0,650744	0,917657	0,01378
Subset:394	0,98419	0,966109	0	0,006668	0,731397	0,918066	0,01581
Subset:395	0,984536	0,821677	0	0,000673	0,661786	0,813066	0,015464
Subset:396	0,984718	0,823413	0	0	0,732837	0,815451	0,015282
Subset:397	0,985658	0,824514	0	0,001285	0,682189	0,813455	0,014342
Subset:398	0,984206	0,966622	0	0,000428	0,62218	0,917159	0,015794
Subset:399	0,984817	0,817731	0,042857	0,000612	0,628661	0,813358	0,015183
Subset:400	0,984652	0,823478	0	0,000061	0,68771	0,814905	0,015348
Subset:401	0,984685	0,967636	0,014286	0,000184	0,692991	0,917628	0,015315
Subset:402	0,984355	0,96697	0,014286	0,000184	0,617379	0,917442	0,015645
Subset:403	0,985279	0,820446	0,014286	0,000061	0,693471	0,812873	0,014721
Subset:404	0,984272	0,822047	0	0,000306	0,696831	0,813349	0,015728
Subset:405	0,981301	0,825693	0	0,000184	0,68675	0,813712	0,018699
Subset:406	0,984437	0,82561	0,014286	0,000306	0,81157	0,817133	0,015563
Subset:407	0,98518	0,823696	0,057143	0,000734	0,692991	0,813979	0,01482
Subset:408	0,984883	0,824301	0	0,000184	0,68963	0,814204	0,015117
Subset:409	0,98622	0,966383	0	0,000061	0,680269	0,920001	0,01378
Subset:410	0,992722	0,869116	0	0,001407	0,645943	0,847452	0,007278
Subset:411	0,985526	0,88487	0,014286	0,001285	0,636582	0,857943	0,014474
Subset:412	0,98546	0,967523	0	0,000184	0,710754	0,919387	0,01454
Subset:413	0,997013	0,965974	0,014286	0,002875	0,635862	0,919641	0,002987
Subset:414	0,996171	0,965008	0,1	0	0,609217	0,921412	0,003829
Subset:415	0,987028	0,965991	0	0,004894	0,638262	0,918506	0,012972
Subset:416	0,986731	0,966104	0	0,000122	0,628661	0,918516	0,013269
Subset:417	0,986715	0,965395	0	0	0,618579	0,917484	0,013285
Subset:418	0,98655	0,963855	0	0	0,658665	0,91732	0,01345
Subset:419	0,977902	0,967083	0	0,000673	0,675708	0,918165	0,022098
Subset:420	0,981797	0,966918	0	0,016945	0,668027	0,919798	0,018203
Subset:421	0,983199	0,827194	0	0,025265	0,756841	0,819551	0,016801
Subset:422	0,9848	0,968154	0	0,002569	0,640663	0,918483	0,0152
Subset:423	0,983546	0,969372	0	0,001958	0,712434	0,920625	0,016454
Subset:424	0,98046	0,966331	0	0,000061	0,702592	0,918056	0,01954
Subset:425	0,982473	0,966509	0,028571	0,000245	0,645703	0,917085	0,017527
Subset:426	0,984701	0,965221	0	0,000122	0,628661	0,918641	0,015299
Subset:427	0,979503	0,964943	0,028571	0,000122	0,648824	0,919451	0,020497
Subset:428	0,983562	0,967131	0,028571	0,001285	0,691311	0,918554	0,016438
Subset:429	0,984883	0,968689	0	0,000551	0,727316	0,920506	0,015117
Subset:430	0,984817	0,963533	0,071429	0,001591	0,530005	0,916677	0,015183
Subset:431	0,983117	0,964777	0	0,00832	0,680989	0,919827	0,016883
Subset:432	0,948245	0,966879	0,014286	0,00263	0,691311	0,914268	0,051755
Subset:433	0,983249	0,964012	0	0	0,694911	0,918606	0,016751
Subset:434	0,983744	0,966209	0	0,012418	0,735238	0,919943	0,016256
Subset:435	0,984751	0,96473	0	0,004588	0,662506	0,917738	0,015249
Subset:436	0,982374	0,963768	0	0,000122	0,681229	0,917779	0,017626
Subset:437	0,983744	0,969089	0	0,000306	0,673068	0,920413	0,016256
Subset:438	0,981929	0,824166	0,014286	0,000245	0,690831	0,816741	0,018071

Subset:439	0,98348	0,968088	0	0,017373	0,68723	0,920917	0,01652
Subset:440	0,981896	0,967592	0	0	0,68483	0,921261	0,018104
Subset:441	0,983661	0,963459	0	0,000061	0,739078	0,918027	0,016339
Subset:442	0,982473	0,969576	0	0,000918	0,669707	0,919991	0,017527
Subset:443	0,983893	0,969333	0	0,000184	0,665386	0,91993	0,016107
Subset:444	0,984239	0,966392	0,157143	0,000184	0,668747	0,919975	0,015761
Subset:445	0,984998	0,967723	0,028571	0,002692	0,675708	0,917847	0,015002
Subset:446	0,983992	0,96804	0	0,001407	0,768843	0,920406	0,016008
Subset:447	0,979899	0,966126	0	0,00263	0,706913	0,919197	0,020101
Subset:448	0,983183	0,965709	0	0,000122	0,651224	0,918622	0,016817
Subset:449	0,982886	0,824357	0	0,000122	0,724676	0,8143	0,017114
Subset:450	0,98343	0,967105	0,014286	0,003242	0,683629	0,917705	0,01657
Subset:451	0,984107	0,968045	0	0	0,733557	0,919332	0,015893
Subset:452	0,983315	0,969067	0,014286	0	0,691791	0,919043	0,016685
Subset:453	0,983249	0,96503	0,014286	0,001101	0,68531	0,918271	0,016751
Subset:454	0,982572	0,966818	0,014286	0,010583	0,655305	0,918329	0,017428
Subset:455	0,981103	0,966727	0	0,005995	0,615218	0,917911	0,018897
Subset:456	0,981037	0,966722	0,014286	0,029241	0,681949	0,922014	0,018963
Subset:457	0,983447	0,968967	0	0	0,699472	0,919528	0,016553
Subset:458	0,984124	0,968654	0	0	0,713394	0,919721	0,015876
Subset:459	0,983711	0,969276	0	0,000489	0,593135	0,918895	0,016289
Subset:460	0,983513	0,963598	0	0	0,645223	0,917484	0,016487
Subset:461	0,981995	0,96486	0	0,019025	0,69011	0,920634	0,018005
Subset:462	0,983628	0,964286	0	0,00728	0,716515	0,919789	0,016372
Subset:463	0,983893	0,965669	0	0,000122	0,677868	0,917754	0,016107
Subset:464	0,983067	0,966218	0	0	0,736198	0,919313	0,016933
Subset:465	0,983794	0,967131	0	0,000306	0,663706	0,919008	0,016206
Subset:466	0,97267	0,965974	0	0,000061	0,712914	0,916596	0,02733
Subset:467	0,983529	0,96831	0	0,005322	0,69011	0,920718	0,016471
Subset:468	0,972604	0,967723	0	0,000612	0,68915	0,918178	0,027396
Subset:469	0,983117	0,967966	0	0,011439	0,68891	0,920171	0,016883
Subset:470	0,984355	0,965095	0	0	0,662746	0,917635	0,015645
Subset:471	0,983744	0,960044	0	0	0,704033	0,913117	0,016256
Subset:472	0,983364	0,970777	0	0	0,672588	0,922049	0,016636
Subset:473	0,984404	0,968549	0	0,038967	0,675948	0,921959	0,015596
Subset:474	0,983628	0,967327	0,028571	0,000184	0,707873	0,919008	0,016372
Subset:475	0,982358	0,964277	0	0	0,707153	0,919988	0,017642
Subset:476	0,983744	0,966866	0,014286	0,000306	0,727076	0,918824	0,016256
Subset:477	0,984074	0,969124	0	0,009482	0,69035	0,919236	0,015926
Subset:478	0,982655	0,969306	0,028571	0,000245	0,806769	0,92056	0,017345
Subset:479	0,984322	0,966422	0,014286	0,020982	0,719395	0,91994	0,015678
Subset:480	0,981615	0,965565	0	0,039212	0,704273	0,921326	0,018385
Subset:481	0,983034	0,954802	0,014286	0,000428	0,711954	0,90979	0,016966
Subset:482	0,983744	0,967814	0,057143	0,002263	0,664186	0,918959	0,016256
Subset:483	0,983876	0,968397	0,042857	0	0,715795	0,92038	0,016124
Subset:484	0,983249	0,967906	0	0,000856	0,62218	0,91848	0,016751
Subset:485	0,984157	0,965891	0,128571	0,000122	0,732117	0,917808	0,015843
Subset:486	0,983034	0,967575	0,157143	0,000061	0,662506	0,918094	0,016966
Subset:487	0,983893	0,966026	0	0,000306	0,651464	0,920213	0,016107
Subset:488	0,982028	0,966835	0,157143	0,001529	0,679549	0,920233	0,017972
Subset:489	0,983942	0,967092	0,057143	0,000245	0,682189	0,918078	0,016058
Subset:490	0,983562	0,969132	0	0	0,725636	0,920602	0,016438
Subset:491	0,983761	0,968254	0,185714	0,001101	0,680029	0,920088	0,016239
Subset:492	0,983711	0,966139	0,085714	0,001285	0,650024	0,917345	0,016289
Subset:493	0,983678	0,963903	0	0,000061	0,673308	0,919654	0,016322
Subset:494	0,983034	0,964991	0,071429	0,00471	0,680509	0,920998	0,016966
Subset:495	0,982572	0,962785	0,042857	0,000061	0,677388	0,918924	0,017428
Subset:496	0,983827	0,966992	0,114286	0,002814	0,695631	0,920001	0,016173
Subset:497	0,984074	0,968158	0,014286	0,003242	0,693471	0,919747	0,015926
Subset:498	0,98211	0,965456	0,028571	0,00416	0,669227	0,919712	0,01789
Subset:499	0,983265	0,967166	0,1	0,00367	0,704273	0,920033	0,016735
Subset:500	0,980361	0,966931	0,085714	0,000122	0,708353	0,919368	0,019639
Subset:501	0,983117	0,967527	0,014286	0,008809	0,704513	0,921647	0,016883
Subset:502	0,98447	0,96553	0,128571	0	0,56169	0,917117	0,01553

Subset:503	0,983447	0,968489	0	0,000245	0,68579	0,919082	0,016553
Subset:504	0,983628	0,96838	0,042857	0,000551	0,611138	0,920348	0,016372
Subset:505	0,984371	0,968719	0,114286	0	0,656745	0,919181	0,015629
Subset:506	0,983513	0,968915	0,257143	0,008381	0,677148	0,920165	0,016487
Subset:507	0,983265	0,963659	0,014286	0,0052	0,683149	0,919027	0,016735
Subset:508	0,981747	0,969058	0,028571	0,007402	0,693231	0,919773	0,018253
Subset:509	0,983298	0,969137	0	0,002202	0,684109	0,920879	0,016702
Subset:510	0,984173	0,968206	0,028571	0,000551	0,741719	0,919802	0,015827
Subset:511	0,979453	0,967122	0,014286	0,000122	0,610418	0,91632	0,020547
Subset:512	0,983282	0,963568	0	0	0,652424	0,917043	0,016718
Subset:513	0,983265	0,967114	0	0,007463	0,68795	0,920741	0,016735
Subset:514	0,983447	0,967392	0,014286	0,029241	0,659145	0,919262	0,016553
Subset:515	0,984124	0,967357	0,071429	0	0,724676	0,918657	0,015876
Subset:516	0,983447	0,968985	0	0,000184	0,671147	0,919631	0,016553
Subset:517	0,982391	0,964238	0	0	0,744359	0,917606	0,017609
Subset:518	0,984041	0,824988	0,014286	0,000061	0,637782	0,813554	0,015959
Subset:519	0,983265	0,966144	0	0,045513	0,621459	0,920014	0,016735
Subset:520	0,983694	0,968088	0	0,005628	0,700432	0,918982	0,016306
Subset:521	0,984437	0,967779	0	0,004099	0,713874	0,918676	0,015563
Subset:522	0,984272	0,969393	0	0,000061	0,715074	0,919397	0,015728
Subset:523	0,983166	0,968989	0	0	0,657945	0,918741	0,016834
Subset:524	0,980179	0,964482	0	0	0,62482	0,915542	0,019821
Subset:525	0,983529	0,8233	0	0,000061	0,659145	0,814889	0,016471
Subset:526	0,983084	0,964704	0	0	0,718195	0,919818	0,016916
Subset:527	0,969749	0,964229	0	0	0,721075	0,916034	0,030251
Subset:528	0,983546	0,966731	0	0,001346	0,69011	0,920126	0,016454
Subset:529	0,983959	0,966979	0	0,000856	0,640182	0,919731	0,016041
Subset:530	0,985147	0,962807	0	0,002569	0,668027	0,917902	0,014853
Subset:531	0,983249	0,968071	0	0,004588	0,662506	0,922589	0,016751
Subset:532	0,984289	0,968545	0	0,011623	0,656265	0,921438	0,015711
Subset:533	0,982968	0,968449	0	0,00104	0,674508	0,919985	0,017032
Subset:534	0,985559	0,964699	0,028571	0	0,62218	0,918564	0,014441
Subset:535	0,984404	0,966644	0	0,008381	0,771003	0,919924	0,015596
Subset:536	0,983001	0,9664	0,028571	0,003548	0,619059	0,918027	0,016999
Subset:537	0,983447	0,968388	0	0,000367	0,68603	0,919474	0,016553
Subset:538	0,983546	0,970268	0,085714	0,000673	0,700192	0,92067	0,016454
Subset:539	0,984718	0,965796	0	0	0,684349	0,917095	0,015282
Subset:540	0,983249	0,96962	0,014286	0,006668	0,736198	0,921551	0,016751
Subset:541	0,983744	0,969437	0,014286	0,000184	0,733557	0,920499	0,016256
Subset:542	0,983298	0,969524	0,014286	0,011868	0,740518	0,921708	0,016702
Subset:543	0,982671	0,967436	0	0,002508	0,615939	0,918689	0,017329
Subset:544	0,982836	0,967231	0,028571	0,000306	0,610418	0,919419	0,017164
Subset:545	0,982985	0,968484	0,014286	0,000061	0,738358	0,919638	0,017015
Subset:546	0,98414	0,967296	0	0,000122	0,704273	0,91866	0,01586
Subset:547	0,984239	0,968419	0	0,000184	0,726116	0,919946	0,015761
Subset:548	0,980988	0,966039	0	0,000184	0,716755	0,918168	0,019012
Subset:549	0,983727	0,966592	0,014286	0,002814	0,683149	0,918229	0,016273
Subset:550	0,983595	0,968493	0,014286	0,042699	0,656985	0,922419	0,016405
Subset:551	0,983496	0,967301	0	0,002753	0,704513	0,919657	0,016504
Subset:552	0,983249	0,969093	0,057143	0,000367	0,68987	0,919888	0,016751
Subset:553	0,983628	0,968841	0	0,000061	0,68651	0,918824	0,016372
Subset:554	0,981285	0,964904	0	0,00367	0,669227	0,918856	0,018715
Subset:555	0,98282	0,964656	0,014286	0,011806	0,746039	0,920663	0,01718
Subset:556	0,980542	0,965595	0,014286	0	0,728997	0,920419	0,019458
Subset:557	0,983265	0,963877	0	0,008931	0,709313	0,917808	0,016735
Subset:558	0,982787	0,96697	0	0	0,681949	0,919869	0,017213
Subset:559	0,984041	0,965204	0	0,02502	0,664426	0,918702	0,015959
Subset:560	0,981665	0,837705	0	0,036888	0,714114	0,828643	0,018335
Subset:561	0,983744	0,968228	0	0,000061	0,692271	0,919429	0,016256
Subset:562	0,982457	0,965308	0	0	0,75036	0,920152	0,017543
Subset:563	0,983414	0,967675	0	0,001101	0,68627	0,920535	0,016586
Subset:564	0,984883	0,965761	0	0	0,675708	0,918593	0,015117
Subset:565	0,983463	0,940484	0	0,000489	0,680029	0,898833	0,016537
Subset:566	0,984569	0,967405	0,014286	0,000795	0,664666	0,919782	0,015431

Subset:567	0,983744	0,969424	0,014286	0,000122	0,715554	0,922621	0,016256
Subset:568	0,98414	0,966879	0	0	0,68555	0,919036	0,01586
Subset:569	0,984305	0,96791	0	0,000184	0,670427	0,918754	0,015695
Subset:570	0,983529	0,96888	0	0,00367	0,682909	0,920294	0,016471
Subset:571	0,983199	0,967649	0	0,005873	0,703793	0,920361	0,016801
Subset:572	0,982457	0,966635	0	0,000306	0,68723	0,917718	0,017543
Subset:573	0,979701	0,967279	0,028571	0,000428	0,730917	0,918866	0,020299
Subset:574	0,984025	0,968171	0	0	0,651704	0,920062	0,015975
Subset:575	0,983348	0,968806	0	0,000122	0,724436	0,920393	0,016652
Subset:576	0,982902	0,968706	0	0,001713	0,636342	0,918483	0,017098
Subset:577	0,98074	0,964029	0	0	0,755881	0,917394	0,01926
Subset:578	0,983678	0,966983	0,057143	0,00104	0,591695	0,917168	0,016322
Subset:579	0,983381	0,964094	0	0	0,658425	0,917918	0,016619
Subset:580	0,983827	0,96463	0	0,001346	0,720115	0,919818	0,016173
Subset:581	0,983777	0,967301	0,014286	0,042393	0,660346	0,919853	0,016223
Subset:582	0,98381	0,969528	0	0	0,704993	0,921184	0,01619
Subset:583	0,983381	0,968067	0	0,000428	0,712194	0,918197	0,016619
Subset:584	0,983232	0,968149	0	0,001958	0,664426	0,919914	0,016768
Subset:585	0,981533	0,963281	0	0,033829	0,728757	0,920268	0,018467
Subset:586	0,984899	0,964264	0,028571	0	0,728036	0,920535	0,015101
Subset:587	0,983117	0,822121	0	0,004527	0,727316	0,813773	0,016883
Subset:588	0,98419	0,965504	0	0	0,674268	0,919863	0,01581
Subset:589	0,990808	0,967166	0	0,000184	0,676188	0,920532	0,009192
Subset:590	0,983496	0,969128	0	0,003365	0,695391	0,921094	0,016504
Subset:591	0,983496	0,969024	0	0,03126	0,68867	0,922708	0,016504
Subset:592	0,983661	0,96868	0	0	0,7518	0,922165	0,016339
Subset:593	0,983216	0,966531	0	0,000673	0,693951	0,918506	0,016784
Subset:594	0,984338	0,966013	0,042857	0,000061	0,612338	0,917197	0,015662
Subset:595	0,983794	0,968541	0	0,000673	0,698272	0,919294	0,016206
Subset:596	0,984025	0,966648	0	0,008992	0,632501	0,918371	0,015975
Subset:597	0,983744	0,969463	0	0,000184	0,664666	0,920265	0,016256
Subset:598	0,984206	0,970094	0,028571	0,000979	0,691551	0,921377	0,015794
Subset:599	0,964699	0,966126	0	0	0,663466	0,915204	0,035301
Subset:600	0,982589	0,970881	0,057143	0,026733	0,68651	0,922464	0,017411
Subset:601	0,98414	0,968284	0	0,000061	0,68723	0,920351	0,01586
Subset:602	0,982325	0,969768	0,042857	0,000428	0,783005	0,921744	0,017675
Subset:603	0,984619	0,967027	0	0	0,663946	0,917143	0,015381
Subset:604	0,984487	0,966104	0	0	0,635622	0,91704	0,015513
Subset:605	0,984272	0,96493	0,014286	0,000306	0,705713	0,919442	0,015728
Subset:606	0,985097	0,966801	0	0	0,694431	0,917204	0,014903
Subset:607	0,98216	0,968767	0,014286	0,034808	0,726356	0,922168	0,01784
Subset:608	0,986088	0,963851	0,071429	0	0,694191	0,917397	0,013912
Subset:609	0,985823	0,966679	0	0	0,639222	0,916853	0,014177
Subset:610	0,984025	0,966152	0,014286	0	0,62626	0,916956	0,015975
Subset:611	0,987193	0,963085	0	0	0,68579	0,919692	0,012807
Subset:612	0,984817	0,967414	0	0	0,691071	0,917172	0,015183
Subset:613	0,983876	0,96416	0	0,001896	0,699472	0,919451	0,016124
Subset:614	0,983909	0,967788	0	0,000122	0,715554	0,920541	0,016091
Subset:615	0,984256	0,965452	0	0,004282	0,675948	0,918072	0,015744
Subset:616	0,983612	0,931791	0	0,000122	0,720595	0,89429	0,016388
Subset:617	0,98315	0,968232	0	0,001162	0,681469	0,921165	0,01685
Subset:618	0,984305	0,964086	0	0,000061	0,62794	0,916914	0,015695
Subset:619	0,984305	0,966222	0	0,002875	0,719875	0,918956	0,015695
Subset:620	0,984701	0,970285	0,071429	0,008136	0,431349	0,92002	0,015299
Subset:621	0,983348	0,8428	0	0,000184	0,766683	0,827875	0,016652
Subset:622	0,983678	0,967101	0	0,003609	0,770763	0,920705	0,016322
Subset:623	0,984074	0,964764	0,085714	0,032116	0,658185	0,918496	0,015926
Subset:624	0,984751	0,966626	0	0,034563	0,693471	0,919962	0,015249
Subset:625	0,984767	0,965695	0	0,004037	0,617139	0,917429	0,015233
Subset:626	0,984569	0,966835	0	0,009054	0,719155	0,918676	0,015431
Subset:627	0,984239	0,966022	0	0,009788	0,713154	0,918506	0,015761
Subset:628	0,980031	0,9667	0,014286	0,000061	0,779405	0,919422	0,019969
Subset:629	0,984833	0,967653	0	0	0,640422	0,917606	0,015167
Subset:630	0,984322	0,967014	0	0,008075	0,642823	0,920959	0,015678

Subset:631	0,984487	0,967349	0,042857	0,004466	0,680509	0,91857	0,015513
Subset:632	0,98216	0,966787	0	0,000122	0,694191	0,91912	0,01784
Subset:633	0,994966	0,970037	0,042857	0,00263	0,43447	0,92438	0,005034
Subset:634	0,9848	0,96335	0	0	0,653385	0,914815	0,0152
Subset:635	0,984239	0,833206	0	0,000122	0,703313	0,818406	0,015761
Subset:636	0,98452	0,96617	0,071429	0,001958	0,733317	0,918995	0,01548
Subset:637	0,992441	0,963851	0,085714	0,001162	0,43903	0,918683	0,007559
Subset:638	0,991319	0,966379	0,042857	0	0,611618	0,91875	0,008681
Subset:639	0,986533	0,965587	0	0,002997	0,431829	0,918828	0,013467
Subset:640	0,984454	0,967101	0	0,000061	0,265002	0,919811	0,015546
Subset:641	0,983364	0,967801	0,114286	0,001285	0,647624	0,918979	0,016636
Subset:642	0,985444	0,966444	0	0	0,600816	0,917355	0,014556
Subset:643	0,98584	0,840046	0,042857	0,000122	0,656985	0,829826	0,01416
Subset:644	0,983001	0,963107	0,085714	0,000061	0,678108	0,91713	0,016999
Subset:645	0,984503	0,967166	0	0,000367	0,721555	0,918313	0,015497
Subset:646	0,983331	0,965543	0	0,002141	0,728277	0,920223	0,016669
Subset:647	0,983711	0,965874	0,014286	0,000551	0,68603	0,918056	0,016289
Subset:648	0,984487	0,965147	0	0,000122	0,693951	0,91841	0,015513
Subset:649	0,985345	0,965769	0	0	0,634662	0,918525	0,014655
Subset:650	0,985213	0,96503	0	0	0,616659	0,918111	0,014787
Subset:651	0,983876	0,963829	0,042857	0,000184	0,681949	0,920136	0,016124
Subset:652	0,985708	0,825406	0	0	0,682909	0,812792	0,014292
Subset:653	0,979602	0,94131	0	0,017129	0,695871	0,902099	0,020398
Subset:654	0,984008	0,966814	0	0,018597	0,68507	0,921162	0,015992
Subset:655	0,991072	0,966187	0	0	0,696111	0,917676	0,008928
Subset:656	0,983711	0,96764	0,028571	0,002141	0,657945	0,918201	0,016289
Subset:657	0,984305	0,965661	0	0	0,692271	0,918149	0,015695
Subset:658	0,983909	0,967353	0	0,011378	0,676428	0,919573	0,016091
Subset:659	0,983463	0,96597	0	0,001958	0,715314	0,919696	0,016537
Subset:660	0,996254	0,965134	0	0,000061	0,649064	0,919387	0,003746
Subset:661	0,983133	0,965656	0	0,0052	0,699952	0,918091	0,016867
Subset:662	0,984784	0,965504	0	0,000367	0,273644	0,918281	0,015216
Subset:663	0,990791	0,966044	0	0,000061	0,663946	0,920895	0,009209
Subset:664	0,984124	0,969002	0	0,002875	0,697552	0,920802	0,015876
Subset:665	0,994273	0,964608	0	0,047593	0,667307	0,921345	0,005727
Subset:666	0,983761	0,964669	0	0,002814	0,683869	0,918037	0,016239
Subset:667	0,984833	0,964247	0	0,001652	0,664186	0,916201	0,015167
Subset:668	0,985411	0,966831	0,028571	0,000061	0,714114	0,917657	0,014589
Subset:669	0,9831	0,968397	0	0	0,679549	0,918972	0,0169
Subset:670	0,983463	0,968062	0	0	0,821171	0,920638	0,016537
Subset:671	0,984355	0,966322	0	0,000122	0,668027	0,916866	0,015645
Subset:672	0,983612	0,966557	0	0	0,680269	0,919477	0,016388
Subset:673	0,984718	0,96898	0	0	0,68699	0,919262	0,015282
Subset:674	0,982853	0,967397	0	0,000979	0,646183	0,920011	0,017147
Subset:675	0,98419	0,966548	0,014286	0,005077	0,672588	0,918101	0,01581
Subset:676	0,994323	0,963755	0,057143	0,002386	0,538646	0,921101	0,005677
Subset:677	0,989075	0,965835	0	0	0,661066	0,916558	0,010925
Subset:678	0,984503	0,967958	0	0,000122	0,386462	0,913323	0,015497
Subset:679	0,984586	0,966239	0	0,006668	0,699952	0,916699	0,015414
Subset:680	0,984553	0,966674	0,028571	0,003487	0,731877	0,919265	0,015447
Subset:681	0,984916	0,964251	0,014286	0,014559	0,442631	0,917786	0,015084
Subset:682	0,98546	0,966122	0	0,000061	0,593855	0,917596	0,01454
Subset:683	0,985048	0,96483	0,028571	0,004649	0,476476	0,916587	0,014952
Subset:684	0,984586	0,968706	0	0,001652	0,205233	0,918995	0,015414
Subset:685	0,983216	0,965761	0,014286	0,001101	0,671147	0,919075	0,016784
Subset:686	0,986071	0,966535	0	0,000122	0,676668	0,917541	0,013929
Subset:687	0,977885	0,961745	0,042857	0,000061	0,638742	0,918037	0,022115
Subset:688	0,98381	0,963255	0,028571	0	0,679309	0,917683	0,01619
Subset:689	0,985163	0,968593	0	0	0,691071	0,918397	0,014837
Subset:690	0,984124	0,968214	0	0,000061	0,598896	0,920142	0,015876
Subset:691	0,984239	0,95645	0,071429	0	0,699952	0,909954	0,015761
Subset:692	0,983959	0,963459	0,114286	0,004282	0,714354	0,917284	0,016041
Subset:693	0,98579	0,966035	0,014286	0	0,635862	0,917246	0,01421
Subset:694	0,984685	0,962915	0,028571	0	0,619539	0,917214	0,015315

Subset:695	0,984173	0,963738	0	0,000122	0,665146	0,918091	0,015827
Subset:696	0,98452	0,96734	0	0,001346	0,655545	0,917406	0,01548
Subset:697	0,982094	0,966931	0,014286	0,003242	0,703793	0,919371	0,017906
Subset:698	0,983529	0,968876	0	0,000061	0,68819	0,919368	0,016471
Subset:699	0,984767	0,824362	0,042857	0,002569	0,68915	0,811973	0,015233
Subset:700	0,984025	0,966331	0,071429	0,000489	0,654585	0,918091	0,015975
Subset:701	0,984437	0,964651	0,071429	0,009237	0,700192	0,917181	0,015563
Subset:702	0,984586	0,9657	0,028571	0	0,631061	0,918265	0,015414
Subset:703	0,983975	0,967853	0	0,000122	0,701872	0,919786	0,016025
Subset:704	0,996353	0,966209	0,071429	0,000061	0,650504	0,919361	0,003647
Subset:705	0,985625	0,964447	0,042857	0,000184	0,658185	0,918901	0,014375
Subset:706	0,988695	0,963938	0	0,000061	0,578493	0,918149	0,011305
Subset:707	0,985559	0,964251	0,057143	0,006301	0,652664	0,918178	0,014441
Subset:708	0,984685	0,827094	0	0,052181	0,723236	0,818371	0,015315
Subset:709	0,988976	0,965848	0,028571	0,008442	0,633461	0,917779	0,011024
Subset:710	0,984041	0,965782	0,085714	0,002019	0,696591	0,917975	0,015959
Subset:711	0,983018	0,968267	0	0,007341	0,661546	0,919053	0,016982
Subset:712	0,985642	0,966783	0	0,000061	0,716515	0,918574	0,014358
Subset:713	0,983546	0,965804	0	0,000367	0,74844	0,919329	0,016454
Subset:714	0,983331	0,968188	0	0,000734	0,714114	0,92049	0,016669
Subset:715	0,984635	0,967192	0	0,000061	0,606577	0,917342	0,015365
Subset:716	0,984784	0,9661	0,028571	0,000061	0,604897	0,918159	0,015216
Subset:717	0,984701	0,966801	0	0	0,671387	0,917702	0,015299
Subset:718	0,98419	0,968241	0	0	0,676908	0,92076	0,01581
Subset:719	0,983893	0,965291	0,042857	0,000122	0,677868	0,91766	0,016107
Subset:720	0,984041	0,965169	0,214286	0,007341	0,616179	0,920843	0,015959
Subset:721	0,985015	0,961932	0	0	0,656025	0,912931	0,014985
Subset:722	0,983513	0,966613	0	0,000367	0,665386	0,916821	0,016487
Subset:723	0,986731	0,837318	0,071429	0,00416	0,675228	0,822161	0,013269
Subset:724	0,984371	0,965404	0,028571	0,007769	0,760922	0,919544	0,015629
Subset:725	0,985328	0,967914	0	0,000428	0,43567	0,916969	0,014672
Subset:726	0,9848	0,96543	0,028571	0,000122	0,62746	0,917226	0,0152
Subset:727	0,991319	0,963551	0	0,010094	0,476956	0,9165	0,008681
Subset:728	0,98386	0,968619	0	0,009115	0,25084	0,920538	0,01614
Subset:729	0,985246	0,964255	0	0,000734	0,631781	0,916503	0,014754
Subset:730	0,984998	0,967523	0	0	0,656025	0,918004	0,015002
Subset:731	0,984206	0,963124	0	0,000061	0,641383	0,91995	0,015794
Subset:732	0,990378	0,962737	0,014286	0,000122	0,704033	0,919535	0,009622
Subset:733	0,983959	0,966474	0	0,001958	0,716515	0,916895	0,016041
Subset:734	0,983843	0,964369	0	0	0,681949	0,920975	0,016157
Subset:735	0,984916	0,966809	0	0,000245	0,660826	0,917738	0,015084
Subset:736	0,991402	0,963873	0,014286	0,00416	0,726356	0,918307	0,008598
Subset:737	0,98485	0,966278	0	0	0,665627	0,916779	0,01515
Subset:738	0,984239	0,963646	0	0,000489	0,634902	0,918213	0,015761
Subset:739	0,984074	0,965108	0,028571	0,012235	0,652664	0,918792	0,015926
Subset:740	0,984058	0,966174	0	0	0,721555	0,916336	0,015942
Subset:741	0,983183	0,912696	0	0,003059	0,710274	0,879188	0,016817
Subset:742	0,983827	0,968893	0	0	0,695391	0,920695	0,016173
Subset:743	0,984404	0,966774	0	0,000061	0,658185	0,916927	0,015596
Subset:744	0,984503	0,968728	0	0,005139	0,672108	0,918709	0,015497
Subset:745	0,98381	0,964638	0,042857	0,029914	0,671867	0,918348	0,01619
Subset:746	0,982869	0,966309	0	0,012969	0,692751	0,919847	0,017131
Subset:747	0,983876	0,967044	0	0	0,683629	0,919538	0,016124
Subset:748	0,983529	0,909198	0	0,005383	0,427028	0,874086	0,016471
Subset:749	0,983084	0,966339	0	0,000061	0,724196	0,919599	0,016916
Subset:750	0,985559	0,96506	0,014286	0,013519	0,49856	0,918702	0,014441
Subset:751	0,995478	0,831801	0,000000	0,000184	0,687470	0,822258	0,004522
Subset:752	0,983315	0,966318	0,000000	0,000000	0,776044	0,920821	0,016685
Subset:753	0,995495	0,962624	0,014286	0,018658	0,665627	0,917738	0,004505
Subset:754	0,984289	0,824410	0,057143	0,004955	0,651464	0,813538	0,015711
Subset:755	0,985378	0,964838	0,014286	0,006790	0,617859	0,916207	0,014622
Subset:756	0,978710	0,966183	0,000000	0,000000	0,689870	0,916689	0,021290
Subset:757	0,978991	0,966787	0,000000	0,000734	0,661306	0,916275	0,021009
Subset:758	0,980674	0,966822	0,000000	0,000000	0,767883	0,920278	0,019326

Subset:759	0,984949	0,966661	0,000000	0,000000	0,702832	0,917390	0,015051
Subset:760	0,983529	0,966209	0,014286	0,003854	0,644263	0,917892	0,016471
Subset:761	0,984635	0,966909	0,000000	0,004955	0,686030	0,917767	0,015365
Subset:762	0,983117	0,880989	0,000000	0,000000	0,663226	0,856685	0,016883
Subset:763	0,982886	0,967893	0,071429	0,002141	0,719875	0,919843	0,017114
Subset:764	0,983496	0,964938	0,000000	0,002263	0,504801	0,918384	0,016504
Subset:765	0,992078	0,965669	0,000000	0,000612	0,658905	0,917056	0,007922
Subset:766	0,985972	0,966518	0,000000	0,000184	0,678349	0,916445	0,014028
Subset:767	0,984107	0,844279	0,042857	0,000551	0,681949	0,826733	0,015893
Subset:768	0,984041	0,965073	0,028571	0,008197	0,719155	0,917419	0,015959
Subset:769	0,985493	0,965465	0,000000	0,003670	0,329573	0,916824	0,014507
Subset:770	0,992903	0,967214	0,057143	0,000489	0,481997	0,918384	0,007097
Subset:771	0,990808	0,966109	0,014286	0,004527	0,335814	0,918667	0,009192
Subset:772	0,983992	0,969141	0,000000	0,000000	0,261882	0,920612	0,016008
Subset:773	0,984833	0,966296	0,000000	0,002753	0,625780	0,918435	0,015167
Subset:774	0,985015	0,964817	0,000000	0,000061	0,644263	0,917175	0,014985
Subset:775	0,983397	0,964125	0,000000	0,000061	0,618819	0,918258	0,016603
Subset:776	0,991385	0,964316	0,000000	0,000122	0,693231	0,919879	0,008615
Subset:777	0,985130	0,966531	0,014286	0,000061	0,684349	0,917464	0,014870
Subset:778	0,983067	0,967249	0,000000	0,000184	0,712674	0,920898	0,016933
Subset:779	0,988728	0,963920	0,128571	0,000000	0,637782	0,917249	0,011272
Subset:780	0,995676	0,966522	0,000000	0,000000	0,660346	0,919316	0,004324
Subset:781	0,983084	0,965300	0,000000	0,000000	0,608977	0,917281	0,016916
Subset:782	0,982803	0,963037	0,085714	0,000673	0,680029	0,917220	0,017197
Subset:783	0,984635	0,967971	0,000000	0,000000	0,688430	0,917223	0,015365
Subset:784	0,983727	0,966331	0,000000	0,000000	0,703313	0,919879	0,016273
Subset:785	0,983430	0,967571	0,014286	0,005811	0,772204	0,921599	0,016570
Subset:786	0,983447	0,964760	0,000000	0,001162	0,715795	0,918107	0,016553
Subset:787	0,983777	0,964595	0,000000	0,000000	0,714114	0,919702	0,016223
Subset:788	0,983529	0,968741	0,000000	0,000000	0,678589	0,920962	0,016471
Subset:789	0,985378	0,966648	0,000000	0,005934	0,655065	0,918545	0,014622
Subset:790	0,983579	0,965504	0,000000	0,000000	0,715554	0,919053	0,016421
Subset:791	0,983777	0,964634	0,000000	0,000918	0,355257	0,917689	0,016223
Subset:792	0,979998	0,946779	0,000000	0,000122	0,752760	0,905115	0,020002
Subset:793	0,983628	0,826089	0,000000	0,043005	0,729237	0,818101	0,016372
Subset:794	0,984751	0,965717	0,000000	0,012418	0,656025	0,918027	0,015249
Subset:795	0,984058	0,965482	0,000000	0,002875	0,707633	0,918072	0,015942
Subset:796	0,984025	0,947149	0,028571	0,005200	0,644263	0,904048	0,015975
Subset:797	0,984437	0,964708	0,000000	0,000000	0,711474	0,916111	0,015563
Subset:798	0,980064	0,964690	0,000000	0,000061	0,682669	0,915175	0,019936
Subset:799	0,983694	0,969428	0,000000	0,000000	0,752520	0,920538	0,016306
Subset:800	0,985559	0,966422	0,000000	0,001101	0,657945	0,916728	0,014441
Subset:801	0,985361	0,964978	0,000000	0,005261	0,656745	0,918557	0,014639
Subset:802	0,983975	0,967449	0,014286	0,004404	0,642343	0,918072	0,016025
Subset:803	0,980856	0,969459	0,000000	0,000061	0,666827	0,920265	0,019144
Subset:804	0,983612	0,964760	0,000000	0,002875	0,529525	0,920557	0,016388
Subset:805	0,986748	0,962128	0,014286	0,000184	0,652424	0,913500	0,013252
Subset:806	0,984998	0,965317	0,000000	0,000000	0,692511	0,915497	0,015002
Subset:807	0,984223	0,966922	0,028571	0,002569	0,746759	0,918078	0,015777
Subset:808	0,985312	0,964116	0,042857	0,001223	0,365578	0,916731	0,014688
Subset:809	0,985427	0,965904	0,000000	0,003487	0,579693	0,918500	0,014573
Subset:810	0,989042	0,962237	0,014286	0,000061	0,487998	0,916828	0,010958
Subset:811	0,984454	0,968549	0,000000	0,003915	0,228517	0,919744	0,015546
Subset:812	0,989817	0,967035	0,014286	0,000367	0,620739	0,918754	0,010183
Subset:813	0,985774	0,967544	0,000000	0,000061	0,664426	0,918455	0,014226
Subset:814	0,984041	0,963172	0,014286	0,000306	0,633221	0,917702	0,015959
Subset:815	0,984998	0,963098	0,000000	0,000306	0,708353	0,918033	0,015002
Subset:816	0,984388	0,966074	0,000000	0,000061	0,694671	0,916336	0,015612
Subset:817	0,982275	0,966204	0,000000	0,010461	0,702832	0,921284	0,017725
Subset:818	0,984718	0,966039	0,000000	0,000061	0,699952	0,916863	0,015282
Subset:819	0,988910	0,964595	0,000000	0,002692	0,646904	0,918628	0,011090
Subset:820	0,984338	0,825114	0,000000	0,000000	0,683149	0,812442	0,015662
Subset:821	0,984883	0,966030	0,000000	0,000061	0,618099	0,917358	0,015117
Subset:822	0,984784	0,963233	0,000000	0,000000	0,687710	0,916988	0,015216

Subset:823	0,985229	0,966613	0,000000	0,000000	0,694671	0,916686	0,014771
Subset:824	0,984553	0,966365	0,000000	0,031382	0,629381	0,920705	0,015447
Subset:825	0,984899	0,966418	0,000000	0,000000	0,634422	0,921149	0,015101
Subset:826	0,984817	0,834464	0,000000	0,000306	0,696591	0,819792	0,015183
Subset:827	0,985114	0,967906	0,057143	0,000306	0,644023	0,917808	0,014886
Subset:828	0,984619	0,963511	0,000000	0,000428	0,656025	0,917078	0,015381
Subset:829	0,981004	0,964982	0,000000	0,003854	0,713874	0,918143	0,018996
Subset:830	0,984371	0,966727	0,014286	0,000061	0,687710	0,919252	0,015629
Subset:831	0,985774	0,965869	0,000000	0,000000	0,634422	0,918776	0,014226
Subset:832	0,983562	0,967183	0,000000	0,000122	0,665386	0,918651	0,016438
Subset:833	0,990576	0,965134	0,028571	0,000061	0,565050	0,918860	0,009424
Subset:834	0,979222	0,966883	0,000000	0,001958	0,727556	0,918345	0,020778
Subset:835	0,984140	0,844057	0,000000	0,001529	0,734518	0,829283	0,015860
Subset:836	0,988679	0,964538	0,028571	0,004588	0,663946	0,917754	0,011321
Subset:837	0,984173	0,964995	0,000000	0,006607	0,677868	0,918956	0,015827
Subset:838	0,983463	0,852319	0,142857	0,000061	0,663226	0,832803	0,016537
Subset:839	0,983595	0,967536	0,000000	0,000795	0,717475	0,919242	0,016405
Subset:840	0,982919	0,965169	0,000000	0,000061	0,740999	0,919303	0,017081
Subset:841	0,983579	0,968014	0,000000	0,020371	0,757081	0,921654	0,016421
Subset:842	0,984421	0,966157	0,000000	0,000061	0,662746	0,917194	0,015579
Subset:843	0,983595	0,966566	0,000000	0,049856	0,697552	0,921695	0,016405
Subset:844	0,984817	0,967001	0,000000	0,004894	0,654585	0,917885	0,015183
Subset:845	0,984734	0,966796	0,000000	0,000184	0,630821	0,919480	0,015266
Subset:846	0,983480	0,965743	0,000000	0,000612	0,661306	0,917750	0,016520
Subset:847	0,983546	0,964821	0,000000	0,002508	0,602976	0,919503	0,016454
Subset:848	0,992276	0,965343	0,000000	0,000000	0,663226	0,916693	0,007724
Subset:849	0,985130	0,860746	0,000000	0,000122	0,680269	0,838616	0,014870
Subset:850	0,984091	0,842151	0,014286	0,005873	0,708353	0,825942	0,015909
Subset:851	0,988183	0,966209	0,014286	0,004466	0,764282	0,919409	0,011817
Subset:852	0,987111	0,964151	0,000000	0,000306	0,458713	0,916715	0,012889
Subset:853	0,996369	0,964634	0,000000	0,000612	0,580893	0,919319	0,003631
Subset:854	0,983579	0,963494	0,071429	0,010644	0,519203	0,917914	0,016421
Subset:855	0,984223	0,968532	0,000000	0,010461	0,251560	0,920560	0,015777
Subset:856	0,983364	0,966387	0,042857	0,002202	0,619539	0,917654	0,016636
Subset:857	0,985543	0,967501	0,014286	0,007524	0,631301	0,920004	0,014457
Subset:858	0,986368	0,967966	0,000000	0,000306	0,639462	0,920239	0,013632
Subset:859	0,984800	0,968175	0,042857	0,002263	0,637782	0,919866	0,015200
Subset:860	0,985840	0,966466	0,000000	0,002814	0,629381	0,919098	0,014160
Subset:861	0,983579	0,823130	0,000000	0,004221	0,704033	0,814683	0,016421
Subset:862	0,983166	0,824331	0,000000	0,000061	0,631061	0,814146	0,016834
Subset:863	0,985873	0,824235	0,000000	0,000122	0,716275	0,813947	0,014127
Subset:864	0,980526	0,944960	0,028571	0,004037	0,697072	0,902491	0,019474
Subset:865	0,985939	0,968523	0,000000	0,000000	0,658185	0,919043	0,014061
Subset:866	0,985064	0,821164	0,028571	0,000061	0,732597	0,814586	0,014936
Subset:867	0,985262	0,822356	0,028571	0,000061	0,635142	0,813397	0,014738
Subset:868	0,984206	0,822822	0,028571	0,000673	0,748920	0,813892	0,015794
Subset:869	0,984503	0,822204	0,042857	0,001591	0,706913	0,814850	0,015497
Subset:870	0,985130	0,826328	0,000000	0,000856	0,698032	0,814207	0,014870
Subset:871	0,983992	0,826411	0,000000	0,000000	0,725396	0,816162	0,016008
Subset:872	0,986616	0,969150	0,000000	0,000673	0,702592	0,920769	0,013384
Subset:873	0,988299	0,912622	0,000000	0,000061	0,651464	0,878851	0,011701
Subset:874	0,980608	0,963672	0,000000	0,000061	0,690831	0,916233	0,019392
Subset:875	0,985510	0,822922	0,000000	0,000000	0,667067	0,814056	0,014490
Subset:876	0,985180	0,825106	0,000000	0,000000	0,705233	0,814847	0,014820
Subset:877	0,984652	0,822582	0,000000	0,000000	0,676908	0,813413	0,015348
Subset:878	0,985279	0,967136	0,000000	0,008136	0,614498	0,918117	0,014721
Subset:879	0,985724	0,823039	0,000000	0,000184	0,690831	0,813455	0,014276
Subset:880	0,985906	0,821930	0,000000	0,000000	0,658425	0,815606	0,014094
Subset:881	0,984305	0,821834	0,000000	0,001040	0,690831	0,813754	0,015695
Subset:882	0,985312	0,821551	0,000000	0,000795	0,624100	0,812149	0,014688
Subset:883	0,984272	0,912461	0,014286	0,000795	0,640422	0,876767	0,015728
Subset:884	0,985114	0,824570	0,000000	0,000184	0,741239	0,815352	0,014886
Subset:885	0,986302	0,899031	0,000000	0,000673	0,687950	0,869469	0,013698
Subset:886	0,985955	0,968676	0,000000	0,000367	0,703313	0,919155	0,014045

Subset:887	0,985526	0,899070	0,042857	0,000061	0,604177	0,867643	0,014474
Subset:888	0,991930	0,966557	0,014286	0,000918	0,634181	0,920361	0,008070
Subset:889	0,984503	0,969154	0,000000	0,000000	0,618099	0,920519	0,015497
Subset:890	0,984058	0,970033	0,028571	0,002447	0,615218	0,919927	0,015942
Subset:891	0,993217	0,825958	0,000000	0,000795	0,619299	0,815175	0,006783
Subset:892	0,993052	0,966600	0,000000	0,000000	0,642583	0,919503	0,006948
Subset:893	0,985394	0,965930	0,000000	0,000000	0,645223	0,919570	0,014606
Subset:894	0,988761	0,964395	0,000000	0,000245	0,619299	0,918162	0,011239
Subset:895	0,978908	0,964751	0,028571	0,000061	0,668987	0,918705	0,021092
Subset:896	0,981714	0,846237	0,014286	0,000489	0,686510	0,830688	0,018286
Subset:897	0,984520	0,965321	0,000000	0,000000	0,702112	0,917172	0,015480
Subset:898	0,984470	0,969802	0,057143	0,000551	0,654825	0,921557	0,015530
Subset:899	0,985609	0,821094	0,000000	0,000000	0,616659	0,814525	0,014391
Subset:900	0,982523	0,825545	0,000000	0,000000	0,701632	0,815104	0,017477
Subset:901	0,983959	0,967923	0,000000	0,000856	0,710994	0,918307	0,016041
Subset:902	0,984751	0,969189	0,057143	0,001101	0,618819	0,920573	0,015249
Subset:903	0,986682	0,822286	0,000000	0,000061	0,677388	0,813027	0,013318
Subset:904	0,984800	0,826067	0,028571	0,000000	0,651464	0,814975	0,015200
Subset:905	0,985147	0,824879	0,028571	0,000979	0,700192	0,817615	0,014853
Subset:906	0,985988	0,825445	0,028571	0,001713	0,751560	0,816281	0,014012
Subset:907	0,984734	0,826019	0,042857	0,004649	0,677868	0,815869	0,015266
Subset:908	0,984025	0,822012	0,042857	0,000061	0,753241	0,814403	0,015975
Subset:909	0,987688	0,965695	0,000000	0,000061	0,690590	0,918310	0,012312
Subset:910	0,992771	0,968362	0,042857	0,000428	0,630101	0,921506	0,007229
Subset:911	0,987061	0,839384	0,028571	0,000122	0,623380	0,824467	0,012939
Subset:912	0,984041	0,971538	0,014286	0,000122	0,716275	0,923682	0,015959
Subset:913	0,984850	0,820994	0,014286	0,000367	0,692271	0,813792	0,015150
Subset:914	0,984520	0,822025	0,000000	0,000184	0,811810	0,816448	0,015480
Subset:915	0,984520	0,965413	0,000000	0,001529	0,685550	0,916648	0,015480
Subset:916	0,983727	0,966409	0,000000	0,000122	0,662506	0,916837	0,016273
Subset:917	0,984388	0,821847	0,014286	0,000061	0,632021	0,813963	0,015612
Subset:918	0,985510	0,823996	0,000000	0,000000	0,668747	0,813889	0,014490
Subset:919	0,984718	0,966783	0,000000	0,000061	0,707393	0,917869	0,015282
Subset:920	0,983876	0,966104	0,000000	0,000306	0,661066	0,919143	0,016124
Subset:921	0,984322	0,825175	0,014286	0,002447	0,692991	0,815975	0,015678
Subset:922	0,985295	0,821207	0,014286	0,000184	0,683389	0,814795	0,014705
Subset:923	0,983348	0,823822	0,014286	0,000551	0,780125	0,816493	0,016652
Subset:924	0,974700	0,965974	0,000000	0,000061	0,758041	0,918737	0,025300
Subset:925	0,981731	0,824057	0,000000	0,000000	0,678589	0,815201	0,018269
Subset:926	0,984817	0,823635	0,014286	0,000000	0,701632	0,815734	0,015183
Subset:927	0,985955	0,966218	0,000000	0,000306	0,684109	0,919194	0,014045
Subset:928	0,991418	0,827011	0,000000	0,000734	0,650744	0,818120	0,008582
Subset:929	0,985823	0,965661	0,000000	0,002569	0,695391	0,917210	0,014177
Subset:930	0,985097	0,966622	0,000000	0,003303	0,715314	0,918946	0,014903

APPENDIX C

The table below shows the numerical results of Experiment num. 1 depicted in Figure 19.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	549,18533	539,378179
2	547,53379	531,6488246
3	542,14759	531,4964739
4	541,38796	531,0893482
5	538,55625	530,9855062
6	537,01202	530,4234268
7	535,40527	529,1710726
8	532,63438	528,2177784
9	532,55794	526,63864
10	531,45348	526,1207985
11	529,36451	525,4002119
12	528,90576	525,1314979
13	528,5156	524,927737
14	528,22739	524,6795533
15	527,33594	524,3058107
16	526,89077	523,9008585
17	526,88883	523,5454787
18	525,38445	523,4559396
19	525,1357	523,1765008
20	525,08961	523,0235066
21	523,95245	522,7902653
22	523,36503	522,3964968
23	522,27514	521,7538164
24	521,82267	521,0980248
25	521,44429	520,8158282
26	521,22376	520,8158282
27	521,22376	520,4735046
28	520,11083	520,1700627
29	520,02348	520,0666451
30	519,20274	519,5362043
31	518,80595	519,3885741
32	518,52742	519,3885741
33	518,48528	519,3806349
34	517,84578	518,7012561
35	517,52085	518,0613995
36	517,52085	517,9674678
37	516,75751	517,5706135
38	516,70663	516,9392
39	516,29251	516,9392
40	516,07348	516,9392
41	515,90985	515,366782
42	515,89576	515,366782
43	515,43436	515,366782
44	515,27454	515,366782
45	515,27454	515,366782
46	515,14582	515,1682729
47	514,41404	515,1682729
48	514,41404	513,4747537
49	514,33967	513,2795484
50	514,11847	513,2651971
51	513,7517	513,2569819
52	513,7517	513,1531448
53	513,7165	512,7656728
54	513,32934	512,7656728
55	513,31832	512,5235157
56	513,29697	512,5235157
57	513,21667	512,5235157
58	512,93775	512,5235157
59	512,56095	512,5235157
60	512,3234	512,5008623
61	512,07341	512,4929237
62	511,87928	512,4929237
63	511,73552	512,4300197
64	511,52741	512,392617
65	511,25437	512,2835603
66	510,9678	512,0077182
67	510,65253	512,007016
68	510,30842	511,9256331
69	510,27356	511,8811755
70	509,96669	511,7991987
71	509,96669	511,729465
72	509,70235	511,6679456
73	509,08533	511,6320695
74	509,08533	511,6320695
75	509,08533	511,6320695
76	509,08533	511,4021212
77	509,08533	511,4021212
78	508,86474	511,4021212
79	508,86474	511,3091644
80	508,64764	511,2391844
81	508,19528	510,9277072
82	508,19528	510,8412284
83	508,19528	510,6210422
84	508,19528	510,5602897
85	508,19528	510,5468819
86	508,12872	510,525256
87	507,96585	510,525256
88	507,96585	510,3052773
89	507,76086	510,3052773
90	507,75629	510,3052773
91	507,70043	510,3052773
92	507,70043	510,3052773
93	507,70043	510,297145
94	506,95179	510,2712125
95	506,95179	510,2712125
96	506,91038	510,2712125
97	506,91038	510,2712125
98	506,91038	510,2712125
99	506,91038	510,2712125
100	506,91038	510,2712125
101	506,91038	510,2712125

The table below shows the numerical results of Experiment num. 2 depicted in Figure 20.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	547,6264997	538,8384192
2	545,9897195	538,8384192
3	543,3818338	538,8384192
4	540,9365633	538,130787
5	538,8179893	536,7072249
6	537,6790016	535,3169403
7	537,493951	534,68159
8	537,2448886	532,9931059
9	537,1896356	532,5588553
10	537,0402181	531,1813689
11	535,7719391	530,2299485
12	535,7719391	530,2299485
13	535,3568316	528,2621071
14	534,697709	527,340743
15	534,5208758	526,3675252
16	534,4892829	525,9892148
17	533,6701169	525,7074305
18	533,6701169	525,0409116
19	532,792333	525,0409116
20	532,5402501	524,8676016
21	532,250034	523,9131935
22	532,1203133	521,4166718
23	532,0919835	520,9226762
24	531,0452001	520,901859
25	531,0452001	520,4731315
26	531,0452001	519,8609073
27	530,8436935	519,8609073
28	530,1921679	519,0891184
29	529,8507046	519,0202521
30	529,4142268	518,8701079
31	529,3150517	518,7394455
32	528,4697084	518,5598444
33	528,1971738	518,3168456
34	527,1361663	518,20241
35	526,4020946	518,0366619
36	525,8967075	517,7500577
37	525,810078	517,2765565
38	525,3361851	516,9854868
39	524,7622774	516,6260454
40	524,4303853	516,3240682
41	524,0763576	516,0975328
42	523,731964	515,8469844
43	523,6667844	515,2666617
44	523,6108878	514,9336733
45	523,0345598	514,4052793
46	522,6692514	514,3939383
47	521,9116775	513,6645172
48	521,9116775	513,6587454
49	521,8231088	513,6164178
50	521,3725497	513,4606
51	521,2999565	512,8529392
52	520,9139677	512,8529392
53	520,4960887	512,4517504
54	520,4960887	512,2690729
55	520,0959408	511,7689567
56	520,0959408	511,7689567
57	519,8299552	511,3010463
58	519,4826721	510,8296053
59	519,2346446	510,8296053
60	519,2346446	510,8296053
61	519,2346446	510,7567444
62	518,9093324	510,065392
63	518,9093324	509,9533018
64	518,6056336	509,9533018
65	518,4042532	509,9533018
66	518,4042532	509,9533018
67	518,4042532	509,5478379
68	518,280354	509,447355
69	517,8460545	509,390131
70	517,7141181	509,2860778
71	517,3843634	508,8602857
72	517,3335391	508,7375779
73	517,1752164	508,7375779
74	516,9444867	508,4896936
75	516,9444867	508,3660899
76	516,9444867	508,1520891
77	516,9128737	508,0610248
78	516,7755417	507,8640306
79	516,4261307	507,2057571
80	516,1240386	506,9338376
81	516,1240386	506,6995753
82	516,1240386	506,512865
83	516,0389981	505,8828997
84	515,9259872	505,5850092
85	515,5616913	505,5850092
86	515,5616913	505,5665183
87	515,5616913	505,1334079
88	515,5507679	504,9696065
89	515,2576796	504,9148512
90	514,7193552	504,8491114
91	514,7193552	504,400372
92	514,7193552	504,3441095
93	514,7193552	503,9826048
94	514,7193552	503,4222131
95	514,653817	503,1713471
96	514,4798642	503,1713471
97	514,4798642	503,068925
98	514,4798642	503,0566501
99	514,4798642	502,7996317
100	514,4798642	502,6557762

The table below shows the numerical results of Experiment num. 3 depicted in **Figure 21**.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	550,8999	516,7388
2	547,0285	516,7388
3	546,7555	516,7388
4	545,9427	516,7388
5	544,4368	516,7388
6	541,5815	516,7388
7	534,4551	516,7388
8	534,4551	516,7388
9	533,7392	516,7388
10	532,6223	516,7388
11	531,7056	516,7388
12	530,6325	516,7388
13	529,4855	516,7388
14	528,6999	516,7388
15	525,9357	516,7388
16	524,9578	516,7388
17	523,9139	516,7388
18	522,6313	516,1422
19	520,9472	515,771
20	520,5228	515,463
21	519,9503	515,4117
22	519,3756	515,1363
23	518,9042	515,1363
24	518,6479	514,604
25	517,4739	514,2552
26	517,1689	513,8336
27	516,8891	513,5342
28	516,2162	513,3263
29	516,2162	513,258
30	515,7806	512,9412
31	515,6174	512,7626
32	515,2319	512,2443
33	514,5207	511,7793
34	513,7921	511,661
35	513,269	511,661
36	512,8493	511,5285
37	512,6991	511,5285
38	512,0749	511,5285
39	511,7638	511,5285
40	511,4114	511,1738
41	511,1324	511,1738
42	510,8546	511,0282
43	510,7541	510,7301
44	510,6798	510,7301
45	509,9889	510,6904
46	509,449	510,3952
47	509,3414	510,3952
48	508,9702	510,3277
49	508,3246	510,1752
50	507,9415	510,1752
51	507,7634	510,1724
52	507,754	510,1208
53	507,3273	510,026
54	507,3273	509,7564
55	507,0739	509,6229
56	506,9087	509,6229
57	506,7102	509,6229
58	506,4896	509,503
59	506,2008	509,3732
60	506,2008	509,3732
61	505,7608	509,1555
62	505,6385	509,1379
63	505,5901	509,1379
64	505,3561	509,1379
65	505,2453	509,1379
66	505,2453	509,125
67	505,1319	509,125
68	504,8108	509,1107
69	504,8108	509,0825
70	504,666	509,0591
71	504,2179	509,0534
72	504,0685	509,0534
73	504,0621	508,8929
74	503,5303	508,8929
75	503,4556	508,8929
76	502,8022	508,8929
77	502,7614	508,8929
78	502,7614	508,8929
79	502,7614	508,8929
80	502,7236	508,8645
81	502,6487	508,8645
82	502,5604	508,8645
83	502,5604	508,7793
84	502,5604	508,7793
85	502,5604	508,7793
86	502,4455	508,7615
87	502,2798	508,6977
88	502,1634	508,6568
89	501,8533	508,5623
90	501,8395	508,5623
91	501,7633	508,5623
92	501,521	508,4223
93	501,3812	508,3388
94	501,1279	508,3286
95	501,1279	508,3286
96	501,0648	508,1759
97	500,9298	508,1197
98	500,8981	508,0965
99	500,6693	508,0965
100	500,5175	508,0929

The table below shows the numerical results of Experiment num. 4 depicted in Figure 22.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	550,0859	504,0384
2	548,3118	504,0384
3	544,6352	504,0163
4	542,9037	504,0163
5	541,0581	504,0163
6	540,6884	503,9471
7	538,1675	503,795
8	538,1675	503,656
9	536,4462	503,3924
10	536,4462	503,3924
11	533,7883	503,2384
12	533,115	502,7016
13	532,2943	502,3299
14	532,1637	502,2635
15	530,5678	502,1482
16	529,166	502,0168
17	527,6194	501,7799
18	527,6194	501,7799
19	527,6194	501,6936
20	526,7497	501,6936
21	526,1718	501,3951
22	525,4895	501,0624
23	525,0339	501,0624
24	524,3811	500,5276
25	523,6768	500,0113
26	523,0476	499,8468
27	522,4819	499,8466
28	522,0042	499,8466
29	521,4575	499,5129
30	520,3954	499,5129
31	520,024	499,5129
32	519,5374	499,1511
33	519,5374	499,1511
34	518,2898	499,1185
35	517,9957	498,8667
36	516,9705	498,784
37	516,9705	498,3142
38	516,9705	498,1121
39	515,9152	498,1121
40	515,8509	498,1121
41	515,5568	498,0153
42	515,0625	498,0153
43	514,783	498,0143
44	514,783	497,8204
45	514,2157	497,6539
46	514,0329	497,6474
47	512,0011	497,6474
48	512,0011	497,6474
49	511,9141	497,6474
50	510,6772	497,5016
51	510,6772	497,294
52	510,5906	497,294
53	510,4382	497,294
54	510,0469	497,294
55	509,7965	497,0367
56	509,6488	496,9291
57	509,5329	496,906
58	509,1757	496,876
59	509,1757	496,8233
60	508,8315	496,7037
61	508,3229	496,6637
62	508,3178	496,5387
63	508,1459	496,0986
64	507,8952	496,0986
65	507,78	496,0986
66	507,6664	496,0986
67	507,6023	496,0175
68	507,3229	495,9601
69	506,9955	495,8447
70	506,835	495,6874
71	505,9523	495,6874
72	505,9523	495,6874
73	505,9523	495,5311
74	505,9523	495,5311
75	505,9523	495,3898
76	505,9523	495,3898
77	505,9523	495,3898
78	505,5335	495,3898
79	505,4997	495,2075
80	505,465	494,9494
81	505,4443	494,9494
82	505,1287	494,8344
83	504,8622	494,6741
84	504,5106	494,6741
85	504,4415	494,5095
86	504,3998	494,3238
87	504,3976	494,2712
88	504,1791	494,1583
89	503,9008	494,1583
90	503,7444	494,1583
91	503,7444	494,1583
92	503,6191	494,105
93	503,2928	494,007
94	503,2928	493,9336
95	503,0564	493,9269
96	503,0564	493,7434
97	502,9771	493,6865
98	502,8946	493,6865
99	502,8946	493,5338
100	502,4722	493,5338

The table below shows the numerical results of Experiment num. 5 depicted in Figure 23.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	548,1640667	502,5983
2	545,5222011	502,5983
3	543,018425	502,5983
4	543,018425	502,5983
5	543,018425	502,5983
6	540,9624798	502,5983
7	538,5845525	502,5983
8	538,531748	502,5983
9	536,074969	502,5983
10	533,8311774	502,5983
11	533,0515247	502,5983
12	532,2025861	502,5983
13	531,7058799	502,5983
14	531,0172114	502,5983
15	529,9819503	502,5983
16	529,2306317	502,5983
17	528,272074	501,9981
18	527,3003977	501,9981
19	526,2749774	501,9981
20	526,1845308	501,9981
21	524,7740284	501,6089
22	524,7167237	501,6089
23	524,1913705	501,6089
24	523,2571567	501,4999
25	523,1183028	501,4999
26	522,3958253	501,4999
27	522,0162735	501,4999
28	521,51072	501,4999
29	520,5276607	501,4999
30	520,257806	501,163
31	520,012675	500,8546
32	519,4896947	500,5983
33	518,8903753	500,4466
34	517,7213452	500,354
35	517,5319971	500,3398
36	517,3342009	499,7417
37	516,4169693	499,7376
38	515,7442568	499,7074
39	515,5833384	499,6827
40	515,5833384	499,6173
41	515,3689961	499,4467
42	514,8348718	499,4467
43	514,3987181	499,1255
44	514,3070745	499,0845
45	514,3070745	499,0845
46	514,0348746	498,9661
47	513,9084864	498,6999
48	513,7491618	498,6904
49	513,6345982	498,6206
50	512,8071849	498,6161
51	512,6572827	498,6161
52	512,5892106	498,6161
53	511,9382189	498,554
54	511,1416075	498,3136
55	510,8800381	498,3136
56	510,8172753	498,3136
57	510,8172753	498,3136
58	510,8172753	498,3136
59	510,8172753	498,3136
60	510,5787881	498,3122
61	510,5787881	498,2343
62	510,5237895	498,2243
63	510,3701967	498,1023
64	510,3066812	498,1023
65	510,2601729	497,8535
66	510,1387151	497,8535
67	509,9723053	497,82
68	509,8939844	497,82
69	509,6231014	497,7033
70	509,5499104	497,5615
71	509,1470661	497,5511
72	509,1470661	497,5205
73	509,095502	497,4465
74	508,964326	497,4465
75	508,7551122	497,4465
76	508,5240571	497,4465
77	508,5240571	497,2746
78	508,2851484	497,2325
79	508,2014512	497,1849
80	508,0536185	496,9639
81	507,9487437	496,8681
82	507,7169577	496,867
83	507,5203757	496,867
84	507,3816356	496,867
85	507,1153295	496,867
86	506,8956715	496,867
87	506,6742519	496,867
88	506,6705438	496,867
89	506,6705438	496,867
90	506,5621192	496,7998
91	506,4882237	496,7998
92	506,3151129	496,7998
93	506,3151129	496,7353
94	506,1928955	496,7183
95	506,1529958	496,583
96	506,0109161	496,5751
97	505,923792	496,3971
98	505,8962405	496,3971
99	505,5720138	496,3472
100	505,486213	496,2994

The table below shows the numerical results of Experiment num. 6 depicted in **Figure 24**.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	543,909	508,7454
2	541,0001	508,7454
3	539,6356	508,7454
4	538,6935	508,7454
5	538,2928	508,7454
6	537,7013	508,7454
7	537,3648	508,7454
8	533,9815	508,7454
9	533,9815	508,7454
10	533,7005	508,7454
11	533,21	508,7454
12	532,2501	508,7454
13	531,8271	508,7454
14	531,8271	508,7454
15	530,8144	508,7454
16	529,3392	508,7454
17	529,0375	508,7454
18	526,9753	508,7454
19	526,5664	508,7454
20	525,7415	508,7454
21	524,9397	508,7454
22	524,2219	508,7454
23	523,9534	508,7454
24	523,4616	508,7454
25	523,3798	508,7454
26	523,058	508,7454
27	523,058	508,7454
28	521,8137	508,7454
29	521,8137	508,7454
30	521,8137	508,7454
31	521,6792	508,7454
32	521,0356	508,7454
33	520,6713	508,7454
34	520,5403	508,4013
35	520,0328	508,4013
36	519,7127	507,9276
37	519,1646	507,9276
38	518,5814	507,7954
39	518,4428	507,5519
40	518,2339	506,9548
41	518,0199	506,9548
42	517,7591	506,913
43	517,3251	506,4699
44	517,2109	506,0686
45	516,9712	505,6104
46	516,6003	505,6104
47	515,4758	505,3854
48	515,4758	505,2398
49	515,2895	505,2398
50	515,2557	504,5907
51	514,6211	503,2228
52	514,1312	503,2228
53	514,1312	503,1975
54	514,1312	502,8741
55	513,8519	502,5961
56	513,711	502,5961
57	513,675	502,356
58	513,3716	502,2676
59	513,1139	502,0655
60	513,1139	501,9806
61	513,1139	501,9806
62	513,1139	501,6157
63	512,6947	501,5106
64	512,6947	501,0796
65	512,3007	500,9645
66	512,1121	500,6285
67	511,2078	500,6285
68	511,2078	500,6285
69	511,2078	500,6102
70	510,9872	500,5957
71	510,5459	500,4965
72	510,5459	500,4965
73	510,5459	500,3788
74	510,5459	500,3214
75	510,5398	500,2915
76	510,2717	499,8413
77	510,2717	499,8413
78	510,0477	499,5541
79	510,0239	499,5541
80	509,9324	499,2957
81	509,8571	499,2719
82	509,7763	498,8989
83	509,4853	498,8989
84	509,303	498,8989
85	509,153	498,8989
86	509,0904	498,8074
87	508,982	498,8074
88	508,9344	498,8074
89	508,8472	498,3737
90	508,7245	498,353
91	508,6926	498,2012
92	508,5633	497,8684
93	508,5085	497,7981
94	508,274	497,3994
95	508,177	497,2525
96	508,177	497,2046
97	508,1051	497,1938
98	508,0961	497,11
99	508,0072	497,0028
100	507,9314	496,8116

The table below shows the numerical results of Experiment num. 7 depicted in **Figure 25**.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	440,3075735	388,4537
2	438,3791223	388,4537
3	438,3791223	388,4537
4	432,6858399	388,4537
5	424,6130699	388,4537
6	423,3801278	388,4537
7	422,0388912	388,4537
8	419,2492929	388,4537
9	418,2281969	388,4537
10	413,5676547	388,0079
11	412,7129907	386,7176
12	411,6879669	386,7176
13	410,9063443	386,3181
14	409,0389382	386,3181
15	408,5229504	386,3181
16	408,0990877	386,0788
17	407,7437353	386,0373
18	407,0424498	386,0373
19	406,7825455	385,7976
20	405,5804026	385,606
21	405,1282224	385,2698
22	404,9263325	384,7673
23	402,0248505	384,6709
24	401,813757	384,6472
25	401,0886318	384,4044
26	400,5281844	384,2522
27	399,6005298	384,02
28	399,2313776	384,02
29	399,0748592	383,757
30	398,1678188	383,751
31	397,9918166	383,5934
32	397,9534089	383,5543
33	397,2094454	383,1444
34	396,3575674	383,0644
35	395,4647666	383,0644
36	395,1327285	382,9916
37	395,0286756	382,9916
38	394,7250293	382,4411
39	394,4273885	382,4411
40	394,1866108	382,0915
41	393,5677223	382,0915
42	393,5160365	381,3999
43	393,1180783	381,2216
44	392,7800835	380,6315
45	392,4169811	380,6315
46	392,0565065	380,4272
47	392,0565065	380,4272
48	391,7275365	380,1412
49	391,3476112	380,0842
50	391,0507174	379,9657
51	390,9453453	379,8194
52	390,8775027	379,8194
53	390,5937382	379,8048
54	390,5937382	379,8048
55	390,1883263	379,7478
56	389,9802499	379,7223
57	389,8958146	379,6382
58	389,891697	379,5917
59	389,4824362	379,3102
60	389,4472477	379,2196
61	388,6680059	379,2196
62	388,6680059	379,1258
63	388,6680059	379,0721
64	388,2230403	378,8712
65	387,7598066	378,7391
66	387,7598066	378,6262
67	387,5848904	378,5277
68	387,4598246	378,5277
69	387,3064647	378,3876
70	387,0992155	378,3876
71	386,6651332	378,3215
72	386,6651332	378,3215
73	386,476455	378,0657
74	386,2617905	378,0657
75	386,0751014	378,0657
76	386,0564558	378,0657
77	385,8766698	378,0657
78	385,4473569	378,0657
79	385,1815658	377,8263
80	385,1622173	377,5361
81	384,8565533	377,5361
82	384,4696804	377,5361
83	384,4696804	377,5361
84	384,3929538	377,5361
85	384,2073331	377,4643
86	383,6338462	377,3246
87	383,5807215	377,2687
88	383,4830835	377,2687
89	383,2782397	377,2203
90	383,161275	377,185
91	383,0040818	377,185
92	382,9395789	377,185
93	382,5842778	377,185
94	382,1632055	377,185
95	382,1632055	377,185
96	382,1360027	377,185
97	382,1360027	377,0869
98	382,1360027	376,756
99	382,1195298	376,756
100	381,8086746	376,756

The table below shows the numerical results of Experiment num. 8 depicted in **Figure 26**.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	490,1484	468,8607
2	490,1484	468,8607
3	489,9574	467,445
4	489,3867	466,0658
5	486,4297	464,6089
6	485,227	461,9179
7	479,1111	458,5338
8	478,1736	458,4074
9	478,1736	456,6773
10	476,2078	456,5293
11	475,0186	456,3831
12	473,3972	455,2503
13	473,3972	454,0568
14	472,6943	453,67
15	470,9293	453,0401
16	470,5481	452,5716
17	470,5481	452,0054
18	469,7971	451,8555
19	468,0441	451,3165
20	468,0441	450,8349
21	467,6428	449,3234
22	465,977	449,2293
23	464,7659	449,0257
24	464,0387	448,7877
25	463,2924	448,4864
26	463,1003	448,2536
27	462,5531	447,8105
28	461,3558	447,2709
29	460,9875	446,7765
30	460,5718	446,7765
31	460,1299	446,3048
32	459,7204	446,3048
33	458,9836	446,139
34	458,9113	445,764
35	458,4742	445,6129
36	458,3816	445,3399
37	458,1232	445,2208
38	457,8774	444,9884
39	457,6023	444,6931
40	457,221	444,6236
41	456,9236	443,5883
42	456,7821	443,4392
43	456,2561	443,4392
44	456,207	442,8507
45	455,7668	442,8507
46	455,3507	442,8507
47	455,2319	442,5548
48	454,5251	442,5548
49	454,5251	442,5548
50	454,4017	442,5548
51	454,1262	442,0266
52	453,8319	442,0266
53	453,5652	441,8073
54	453,4442	441,4097
55	453,2246	441,308
56	453,2047	441,2135
57	452,5443	441,0715
58	452,302	441,0715
59	452,2147	440,9795
60	452,1258	440,883
61	451,7137	440,6313
62	451,6595	440,6313
63	451,1725	440,6271
64	450,9881	440,3684
65	450,591	440,3684
66	450,591	440,3684
67	450,591	440,2389
68	450,3593	439,9163
69	449,9753	439,9163
70	449,9722	439,9163
71	449,7862	439,9163
72	449,7382	439,9163
73	449,7382	439,8731
74	449,7382	439,6889
75	449,5574	439,4363
76	449,486	439,4363
77	449,3718	439,4254
78	449,2867	439,1703
79	449,1162	439,1233
80	449,0496	438,9306
81	449,0482	438,8588
82	448,8435	438,6752
83	448,6364	438,3566
84	448,6013	438,3566
85	448,2839	438,2902
86	447,8858	438,2902
87	447,8858	438,2902
88	447,5494	438,2902
89	447,5494	437,8394
90	447,5494	437,7874
91	447,5494	437,7064
92	447,5494	437,6023
93	447,5494	437,4661
94	447,5494	437,4554
95	447,4938	437,1396
96	447,2253	437,1396
97	447,2073	437,1396
98	446,9687	437,0354
99	446,7758	437,0354
100	446,5807	437,0067

The table below shows the numerical results of Experiment num. 9 in depicted Figure 27.

	Classic ANN	Genetic & ANN Hybrid Transfer
1	545,01	531,2085
2	544,5422	531,2085
3	544,5422	531,2085
4	542,209	527,6092
5	542,209	525,1448
6	542,209	522,139
7	541,176	519,6629
8	539,7486	519,6349
9	538,1465	519,6349
10	537,0691	519,0153
11	536,1694	517,3064
12	535,3687	516,437
13	534,231	516,1097
14	533,8463	512,7757
15	532,8721	512,7121
16	532,4962	511,4354
17	532,2314	511,3917
18	531,6582	510,8381
19	529,4604	510,1842
20	529,3144	509,6956
21	529,0463	509,4582
22	528,182	508,919
23	527,2778	507,9049
24	527,2778	507,9049
25	526,4831	506,6563
26	526,139	506,6563
27	525,8647	506,6563
28	525,1071	505,2283
29	524,4894	505,1297
30	523,7056	504,7233
31	523,7056	504,7233
32	523,4988	504,7233
33	523,4988	504,5814
34	522,9893	504,4996
35	522,9138	503,772
36	522,4092	503,2152
37	522,3318	503,2152
38	521,9899	503,1263
39	521,8808	502,8463
40	521,3948	502,78
41	521,36	502,4715
42	520,7455	502,272
43	520,7455	502,1099
44	520,7455	502,1099
45	520,7455	502,1099
46	520,5855	502,0488
47	519,7571	501,8784
48	519,7571	501,7064
49	519,7571	501,6129
50	519,7571	501,291
51	519,126	501,291
52	519,0775	501,291
53	518,2081	501,291
54	517,941	501,291
55	517,941	500,5912
56	517,941	500,5912
57	517,941	500,5912
58	517,6748	500,5912
59	517,4202	500,5912
60	517,2448	500,2992
61	516,963	500,095
62	516,7065	500,095
63	516,4379	499,9004
64	516,4379	499,7302
65	516,4379	499,491
66	515,8914	499,491
67	515,8676	499,4821
68	515,4815	499,3778
69	515,1837	499,3577
70	515,1754	499,2306
71	515,1754	499,2188
72	515,1754	498,9686
73	514,5562	498,9686
74	514,5203	498,8684
75	514,4475	498,5809
76	514,2394	498,5776
77	513,8464	498,4711
78	513,825	498,3677
79	513,757	498,3677
80	513,757	498,1717
81	513,757	497,9951
82	513,7215	497,9728
83	513,7215	497,9126
84	513,7109	497,9094
85	513,7109	497,878
86	513,6774	497,812
87	513,5795	497,7405
88	513,5795	497,662
89	513,4629	497,1955
90	513,3016	497,0138
91	513,3016	496,8854
92	513,0122	496,8854
93	512,9962	496,8251
94	512,9962	496,7321
95	512,7096	496,7321
96	512,7096	496,5913
97	512,5394	496,5587
98	512,0473	496,4499
99	512,0473	496,4499
100	512,0473	496,4499