ISTANBUL TECHNICAL UNIVERSITY ★ ENERGY INSTITUTE

COMPARISON OF ENERGY PRODUCTION BETWEEN OBSERVED AND PREDICTED WIND SPEED SERIES WITH ARTIFICIAL NEURAL NETWORKS

M.Sc. THESIS

Coşkun YILDIZ

Energy Science and Technology Division

Energy Science and Technology Programme

MAY, 2016

ISTANBUL TECHNICAL UNIVERSITY ★ ENERGY INSTITUTE

COMPARISON OF ENERGY PRODUCTION BETWEEN OBSERVED AND PREDICTED WIND SPEED SERIES WITH ARTIFICIAL NEURAL NETWORKS

M.Sc. THESIS

Coşkun YILDIZ
(301131007)

Energy Science and Technology Division

Energy Science and Technology Programme

Thesis Advisor: Assist. Prof. Dr. Burak BARUTCU

MAY, 2016

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ ENERJİ ENSTİTÜSÜ

YAPAY SİNİR AĞLARIYLA YAPILMIŞ RÜZGAR HIZI TAHMİN VE GÖZLEM SERİLERİNDEN ENERJİ ÜRETİM HESABI KARŞILAŞTIRMASI

YÜKSEK LİSANS TEZİ

Coşkun YILDIZ
(301131007)

Enerji Bilim ve Teknoloji Anabilim Dalı

Enerji Bilim ve Teknoloji Programı

Tez Danışmanı: Yrd. Doç. Dr. Burak BARUTÇU

MAYIS, 2016

Coşkun YILDIZ, a **M.Sc.** student of **ITU Energy Institute** student ID **301131007**, successfully defended the **thesis** entitled "**COMPARISON OF ENERGY PRODUCTION BETWEEN OBSERVED AND PREDICTED WIND SPEED SERIES WITH ARTIFICIAL NEURAL NETWORKS"** which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Assist. Prof. Dr. Burak BARUTCU** ....
Istanbul Technical University

Jury Members : **Prof. Dr. Kasım KOCAK** ....
Istanbul Technical University

Assoc. Prof. Dr. Ozan ERDINC ....
Yıldız Technical University

Date of Submission: 02 May 2016
Date of Defense : 09 June 2016

To my lovely family,

FOREWORD

All the effort put into this master thesis would not be enough, if I did not have the support of my loving family and friends around me. Therefore, I would like to thank my sister, Nilgün Alageyik, for being strict when I needed it and her priceless advice and support they was always ready to give. My friends made sure I had the necessary distraction during the process of research and always came up with handy ideas.

Most of all, I would like to show my appreciation and gratitude for the involvement of my supervisor, Assist. Prof. Dr. Burak Barutçu, who kept an eye on the shaping of the thesis. With his thoroughness and excellent pedagogic skills, writing of this thesis would never be possible, nor would it be pleasant.

Moreover, I recognize and appreciate mr. Ömer Duyar owner of Duyar Valve made to encourage me to strive for more and show the incentive to work independently. This study would not be possible without the courtesy and understanding of Levent Günaydın, my director in Duyar Valve.

Thereby I would like to close one chapter of my life and open up a different one, hopefully as interesting and rewarding as the three years of studying at the Istanbul Technical University and working at Duyar Valve.

May 2016

Coşkun Yıldız
(Mechanical Engineer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

ADALINE	: Adaptive Linear Neuron
ANN	: Artificial Neural Networks
AR	: Autoregressive
ARCH	: Autoregressive Conditional Heteroskedasticity
ARIMA	: Autoregressive Integrated Moving-Average
ARMA	: Autoregressive Moving-Average
GARCH	: Generalized Autoregressive Conditional Heteroskedasticity
IEEE	: Institute of Electrical and Electronics Engineers
INNS	: International Neural Network Society
LMS	: Least Mean Square
MA	: Moving-Average
MADALINE	: Many Adaptive Linear Neurons
MFNN	: Multi-layered Feed-forward Neural Network
MLR	: Multiple Layer Regression
NRMSE	: Normalized Root Mean Square Error
NWP	: Numerical Weather Prediction
RMSE	: Root Mean Square Error
SLR	: Single Layer Regression
TAR	: Threshould Auto Regressive
VARMA	: Vector Autoregressive Moving-Average
VARMAX	: Vector Autoregressive Moving-Average with Exogenous Variable

LIST OF TABLES

Page

LIST OF FIGURES

Page

COMPARISON OF ENERGY PRODUCTION BETWEEN OBSERVED AND PREDICTED WIND SPEED SERIES WITH ARTIFICIAL NEURAL NETWORKS

SUMMARY

Transportation and mining costs of nuclear power systems and fossil fuel cells are very changeable and electricity cost from these power sources often fluctuates. Because wind is fixed and free, many of the experts assume that wind power sources are a viable alternative in order to decrease this cost.In order to have cost-effective and renewable energy production, the usage of wind-generated electiricity is an option.

There are many advantages of using wing energy sources. One of the advantages is that small businesses and local landowners could operate a single wind turbine or cluster of wind turbine because wind energy is homegrown. Secondly, wind energy does not emit contaminants into the streams and lakes, and wind energy is not a reason for hazardous airborne pollutants. Thirdly, wind energy does not cause acid rains and contribution to global climate change and this problem is currently continuous to be withstanded. Another important problem for the world is greenhouse gases. A greenhouse gas (sometimes abbreviated GHG) is a gas in an atmosphere that absorbs and emits radiation within the thermal infrared range. This process is the fundamental cause of the greenhouse effect. Most other power sources, such as coal and natural gas, causes greenhouse gases, on the other hand wind energy produce none. Alhough wind farm affects livestock grazing and crop production, this impact is minimal because wind farms do not always founded in crop prediction areas and it does not cover a big area of the land.

On the other hand, renewable energy prediction and dispatching has priority with respect to other kinds of energy. Because, transmission and dispatching of electricity into the grid are submitted to very strong constraints, in order to maintain continuous and instantaneous balancing between energy entered and requested and to guarantee energy frequency and voltage into narrow range. Respecting these constraints is awkward because of the variability of production, transmission and demand and due to the absence of storage systems. A program of injection and withdrawal is defined for each dispatching zone with 24 hours of advance: fees are assigned to the producers by the electric service provider due to mismatch between energy forecasted and actually entered into the grid. An error on the forecasted power is allowed but after the first 6 months of 2013 such limit is set to 10%. The accuracy on wind power production forecast on 24 hours basis is therefore fundamental in order to limit charges due to the unbalance.

Wind energy prediction starts with analyzing of time series. Time series analyzing methods separated two parts of linear and non-linear methods. As most of the other problem in nature, wind speed could be analyzed by using non-linear methods. Nonlinear and data-analytic nonparametric methods have greatly advanced along

seemingly unrelated paths, among very important developments in the fileds of statistic over the last two decades. One of the important and developing nonlinear methods is the usage of artificial neural networks.

Within the scope of the thesis, performance analysis of ANN and future prediction of wind speed will be made by using Matlab ANN codes. One more prediction for wind energy is also calculated. Being 1-hour later, 6-hour later, 12-hour later and 24-hour later, four different calculations are going to be made in order to see the performance of the ANN. In addition to this, a next-year prediction is going to be made by using wind speed data in 2001 and making a prediction for wind speed data in 2002. Lastly, three different scenarios will be applied for wind energy prediction. A comparison will be made by using predicted and onserved wind speed data in WAsP and the effect of wind direction will also be observed by using different wind direction data. In conclusion, both wind speed and wind energy prediction will be made. Although, wind speed prediction will be made by using only Matlab ANN codes, energy prediction will be made by using a hybrid system of using both Matlab ANN codes and WAsP software.

YAPAY SİNİR AĞLARIYLA YAPILMIŞ RÜZGAR HIZI TAHMİN VE GÖZLEM SERİLERİNDEN ENERJİ ÜRETİM HESABI KARŞILAŞTIRMASI

ÖZET

Günümüzde kullanımı oldukça yaygın olan nükleer güç sistemlerinin ve fossil yakıtların ulaştırma ve işletme masrafları oldukça değişkendir. Bu kaynkalardan elde edilen elektirik üretim maliyetleri de bu nedenden dolayı dalagalanmalar göstermektedir. Birçok uzman, rüzgârın elektirik enerjisi üretim maliyetlerini azaltmak için güçlü bir alternatif olduğunu iddia etmektedir. Yenilenebilir enerji kaynaklarının kullanımı ve düşük maliyetli enerji üretimi adına rüzgâr kaynaklı güç santrallerinin kurulması bir seçenek olarak karşımıza çıkmaktadır.

Rüzgâr enerjisi kullanımını günlük hayatımız ve doğa için birçok avantaj sağlamaktadır. Rüzgârın yerel bir enerji kaynağı olması sebebiyle, küçük işletmeler ve küçük toprak sahipleri kendi enerjilerilerini karşılayabilmek adına rüzgâr enerjisi kullanabilmektedirler. Rüzgâr enerjisi dereler ve nehirler için kirleticilere neden olmamaktadır ve havada uçuşan thelikeli maddeler yaymamaktadır. Günümüzde, doğanın karşı karşıya kaldığı en büyük problemlerden biri olarak asit yağmurları gelmektedir. Asit yağmurları zararlı enerji kaynaklarının kullanımı sonucu ortaya çıkmaktadır. Ayrıca, kömür ve doğal gaz gibi zararlı enerji kaynaklarının kullanılması iklim değişikliklerine ve doğal dengenin bozulmasına neden olmaktadır. Enerji elde etmek için rüzgâr enerjisin kullanılması bu bakımdan da önem kazanmaktadır.

Rüzgâr enerjisinin stokastik doğası gereği, karmaşık elektirik şebekelerinde, programlanabilir kaynaklarla beraber enerji üretimininindeki dengeyi sağlamak amacıyla, güç üretim tahmini yapmak oldukça önemlidir.

Enerji frekans ve voltajını dengeleyebilmek amacıyla, anlık ve devamlı olarak enerji girdisinin ve talep miktarının iyi bir şekilde ayarlanması gerekir. Bu sebeple elektiriğin iletilmesi ve sevki sırasında güçlü yaptırımlarla karşılaşılır. Yenilenebilir enerjinin sevkiyatı, diğer enerji kaynaklarının sevkiyatına göre bazı zorluklar taşımaktadırlar.

Üretimim değişkenliği, iletim, talep ve depolama sistemlerinin olmaması nedeniyle bazı yaptırımların uygulanmaktadır. Her bir şebeke için yük atma ve yük çekme programı 24 saatlik süre boyunca uygulanır. Ayrıca, elektirik sağlayıcıları tarafından, elektirik üreticilerine tahmin edilen enerjini ve sağlanan enerji arasındaki dengesizlikler nedeniyle ücretlendirme politikası uygulaktadır. 2013 yılının ilk yarısı itibariyle, tahmin edilen enerji ile gerçekte üretilen enerji arasındaki fark 10% değeriyle sınırlandırmıştır. Bu sebeple, enerji üretimiyle ilgili yapılacak öngörülür, dengesizlikler sonucu uygulanacak para cezaları nedeniyle, önem taşımaktadır.

Türkiye'de bu konu incelenirse; sistem işletmecileri elektrik üretiminin tüketimden fazla olduğu durumlarda santrallerden yük attırmak, üretimin tüketimden az olduğu

durumlarda ise santrallerle yük aldırmak suretiyle sistemin dengesini sağlar. Bu durumda gün öncesi planlama kapsamında piyasa işletmecisi, dengeleme güç piyasası kapsamında ise sistem işletmecisi tarafindan üretim azalması için verilen talimatlara Yük-atma (YAT), üretim artışı için verilen talimatlara da Yük-al (YAL) talimatları denilmektedir. Piyasa katılımcısı gün öncesinde Piyasa İşletmecisine şu girdileri vermek zorundadır: günlük üretim programı, ikili anlaşmlar ve serbest tüketicilerin tüketimi. Dengeleme Güç Piyasası'nda Saat 14:00 ile 16:00 arasında Piyasa Katılımcıları kesinleşmiş günlük üretim programlarını, emreamade kapasitelerini, yük alma ve yük atma miktar ve fiyat setlerini UYTM'ye bildirirler.

Rüzgâr enerjisi üzerine yapılacak tahminler yukarıda belirtilen nedenler dışında da önem taşımaktadır. 3-7 gün arası yapılan bir tahmin ile beraber bakım-onarım üzerine çalışmalar yapılabilirken; şebekedeki güç dengesinin sağlanabilmesi için genellikle 1-72 saatlik tahminlere ihtiyaç duyulmaktadır.

Rüzgâr gücü tahmin yöntemleri iki ana başlık altında incelenebilir. Bunlardan biri uzun dönemlik tahminler için kullanılan ve istatistiksel bir yaklaşım olan rüzgâr zaman serileri analiz metodudur. Diğer yöntem ise kısa dönemlik tahminler için kullanılan hava tahmin yöntemlerinin girdi olarak kullanıldığı ve enerjinin tahmin edildiği yönemlerdir. Tahmin yöntemleri genel olarak ayrıca fiziksel yöntemler, istatistiksel yöntemler (kara kutu yöntemi) ve öğrenen yöntemler olamak üzere 3 farklı gruba ayrılabilir. Öğrenen yöntemlerin ve istatistiksel yöntemlerin kullanılabilmesi için genellikle büyük sayıda ve doğrulukta kaydedilmiş geçmiş datalara ihtiyaç duyulmaktadır.

İstatistiksel bir yöntem olarak yapay sinir ağları çok katmanlı bir algılayıcı olarak olarak rüzgâr hızı tahminlerinde kullanılabilirler. Yapay sinir ağları, insan beyninin sinir sistemine ve çalışma prensibine dayanan elektriksel bir modeldir. Bir anlamda insan beyninin ufak bir kopyası gibidir. İnsan beyninin öğrenme yoluyla yeni bilgiler üretebilme, keşfedebilme, düşünme ve gözlemlemeye yönelik yeteneklerini, yardım almadan yapabilen sistemler geliştirmek için tasarlanmışlardır, çünkü bu özellikleri, geleneksel programlama algoritmaları ile yaratabilmek imkânsızdır.

İlk yapay nöron,1943 yılında nöropsikiyatrist Warren McCulloch ve bilim adamı Walter Pits tarafindan üretilmiştir. Ancak dönemin kısıtlı olanakları nedeniyle, bu alanda çok gelişme sağlanamamıştır. Bundan sonra 1969'da Minsky ve Papert bir kitap yayınlayarak, yapay sinir ağları alanında duyulan etik kaygıları da ortadan kaldırmış ve bu yeni teknolojiye giden yolu açmışlardır. İlk gözle görülür gelişmeler ise 1990'lı yıllara dayanmaktadır.

Yapay sinir ağları insan beynini taklit edebilmek için birbirleriyle uyum içinde çalışan yoğun şekilde bağlanmış bilgi işleme merkezlerinden, yani nöronlardan, oluşmaktadır. İşlem birimleri aslında bir transfer denklemi gibidir. Bilgiyi alır, transfer fonksiyonunu uygulayarak işleme sokar ve bir çıktı oluşturur. İnsan beynindeki sinir hücresinde de bu görevi sırasıyla dentrit, hücre gövdesi ve akson üstlenir. Bir yapının bilgiyi nasıl işleyeceği, transfer fonksiyonuna, diğer ağlarla birbirlerine nasıl bağlandıklarına ve kendi sinaptik ağırlıklarına bağlıdır.

Bir yapay sinir ağı, belirli bir amaç için oluşturulur ve insanlar gibi örnekler sayesinde öğrenir. Nasıl insanın öğrenmesi sinaptik boşluklardaki (2 sinir hücresi arasındaki boşluklar) bazı elektriksel ayarlamalar sayesinde oluyorsa, aynı şekilde, yapay sinir ağları da tekrarlanan girdiler sayesinde kendi yapısını ve ağırlığını değiştirir. Yapay sinir ağları, canlıların sinir sistemi gibi adapte olabilen bir yapıya sahiptir. Karar verme aşamasında bağlantı ağırlıkları da devreye girer. İşlem birimleri her ne kadar tek başlarına çalışıyor gibi gözükse de, aslında birçok yapay sinir ağı aynı anda çalışır ve dağınık, paralel hesaplama örneği gösterir.

Yapay sinir ağlarıyla regresyon analizi, tahmin yürütme, karar verme, zaman serisi analizi, sınıflandırma, control benzetimi, optimizasyon, doğrusal olmayan sinyal işleme ve doğrusal olmayan system modellemesi yapılabilir. Yapay sinir ağları bu sebeple; elektronik, havacılık, robotic, tıp, savunma sanayi, dil işleme, otomotiv ve savunma sistemleri alanlarında kullanılabilmektedir.

Yapay sinir ağlarının zaman serilerinin analizinde ve yahmin yürütmede kullanılabiliyor olması, bu ağların rüzgâr nerjisi tahmini için kullanılabileceği anlamını taşımaktadır. Çünkü, uzun bir dönem için (örneğin 1 yıl) kaydedilen rüzgâr hızı datai bir zaman serisi örneğidir. Yapay sinir ağlarının öğrenme ve tahmin yürütme özellikleri yardımıyla, kaydedilen bu verilerle ileriye dönük tahminler yapılabilmektedir.

Zaman serisi analizi doğrusal zaman serisi analizi ve doğrusal olmayan zaman serisi analizi olmak üzere iki bölüme ayrılmıştır. Doğanın etkili olduğu olaylar, doğrusal olmayan zaman serisi analizi kullanılarak analiz edilebilir. Rüzgâr, doğa olaylarından doğrudan etkilendiği için, rüzgâr hız verileri de doğrusal olmayan yöntemlerle analiz edilebilecek bir zaman serisini oluşturmaktadır. Doğrusal özellik göstermediği için, rüzgâr hız verilerinin analiz edilmesinde parametrik yöntemlerin kullanılması, parametrik olmayan yöntemlerin kullanımına oranla daha fazla hata verebilmektedir.

Yapay sinir ağlarına rüzgâr hızı verileri beslenildiği takdirde sonuç olarak yine tahmin edeceği veriler rüzgâr hızı olacaktır. Aynı şekilde, yapay sinir ağları girişine rüzgâr enerjisi verileri girildiğinde ise sonuç olarak enerji tahmini elde edilcektir. Bu sebeple ağa beslenecek olan veri önem taşımaktadır. Ancak yapay sinir ağına beslenen bir rüzgâr hızı verisinden elde edilecek verilerle rüzgâr enerjisi hesabı yapılabilir. Rüzgâr enerjisi hesabı enerji denkleminin kullanılması, Matlab programının kullanılması ve WAsP gibi analiz programının kullanılması gibi çeşitli yöntemlerle yapılabilmektedir.

WAsP, Danimarka Meteoroloji Teşkilatı'nın Riso Meteoroloji Laboratuvarında hazırlanmış ve geliştirilmiş olan ve Avrupa Rüzgâr Atlasının hazırlanmasında da kullanılan bir paket programdır. WAsP paket programı, veri analizlerini, rüzgâr hız verilerinin 2 parametreli Weibull dağılımına uygun bir dağılım gösterdiğini varsayarak yapmaktadır. Bu program, dört değişik girdi bilgisini kendi alt modellerinde değerlendirerek, bölgesel rüzgâr atlası istatistiklerini hesaplamaktadır. WAsP'ın kullandığı temel bilgiler saatlik rüzgâr verisi, bölge pürüzlülük bilgileri, yakın çevre engel bilgisi ve bölgenin topoğrafyasıdır. WAsP programı bir bütün olmakla birlikte, bilgilerin değerlendirilmesinde alt modeller kullanmaktadır.

Bu tez çalışmasında, iki ayrı ana başlık altında inceleme yapılacaktır. İlk yapılacak çalışmada, İstanbul Teknik Üniversitesi (İTÜ) Maslak/Ayazağa kampüsünde ölçülen 2001-2002 yıllarına ait saatlik rüzgâr hızı verilerinin kullanılmasıyla, yapay sinir ağlarının performans analizi yapılacaktır. İlk çalışmada 2001 yılına ait rüzgâr hız verileri, Matlab (R2015a) programında hazırlanan yapay sinir ağına beslenerek en iyi performans değerini sağlayan yapay sinir ağı elde edilmeye çalışılacaktır. En iyi ağı elde etmek için, yapay sinir ağının bazı parametrelerinde (giriş sayısı, gizli tabaka sayısı, vs.) değişiklikler yapıpacaktır. 2001 yılına ait gözlemlenen rüzgâr hızı verileri ve yapay sinir ağı kullanılarak tahmin edilen rüzgâr hız verileri WAsP programına aktarılarak rüzgâr enerjisi hesabı yapılacaktır. WAsP'ta hesaplama yapılırken, diğer tüm değişkenler aynı kalmak şartıyla (topoğrafya, kullanılan rüzgâr türbini, yön bilgisi, vs.) sadece rügzar hız verileri değiştirilecektir. Elde edilen rüzgâr enerji değerleri karşılaştırılacaktır.

İkinci olarak ise, yapay sinir ağlarının ileriye dönük rüzgâr hızı tahmini üzerinde başarısı araştırılacaktır. Çalışmanın bu aşamasında, yapay sinir ağına beslenen 2001 rüzgâr hızı verilerinden ileriye dönük elde edilen çıktılar (tahmin edilen); 2002 yılına ait verilerle (gözlemlenen) karşılaştırılacaktır.

Genel olarak bu tez çalışmasında, yapay sinir ağlarının performans analizi ve ileriye dönük tahminlerdeki başarısı üzerine çalışılacaktır. Tahmin edilen ve gözlemlenen rüzgâr hızlarındaki datalar WAsP programına aktarılarak enerji hesabı karşılaştırması yapılacaktır. Yapay sinir ağlarıyla tahmin edilen rüzgar hızlanırının WAsP yazılımı kullanılarak rüzgar enerjisine dönüştürülmesi, hibrit bir tahmin sisteme örnek teşkil etmektedir.

1. INTRODUCTION

1.1 Purpose of Thesis

Within the scope of this master thesis, how neural networks are designed by using the Matlab program to forecast wind speed and wind energy forecasting by WAsP software will be shown. Artificial neural networks (which will be mentioned ANN shortly) will be designed in order to predict the future value of wind speed by using older values of the related wind speed data. Furthermore, a complete design of a wind farm using the WAsP program of ITU campus located in Maslak in Istanbul will be done. The reason is to use WAsP program will be explained later but the main purpose of the usage of the program is to calculate a possible wind energy calculation by using the predicted data from the created ANN. The figure (see Figure 1.1) shows the related topics and subjects which are included in the study.

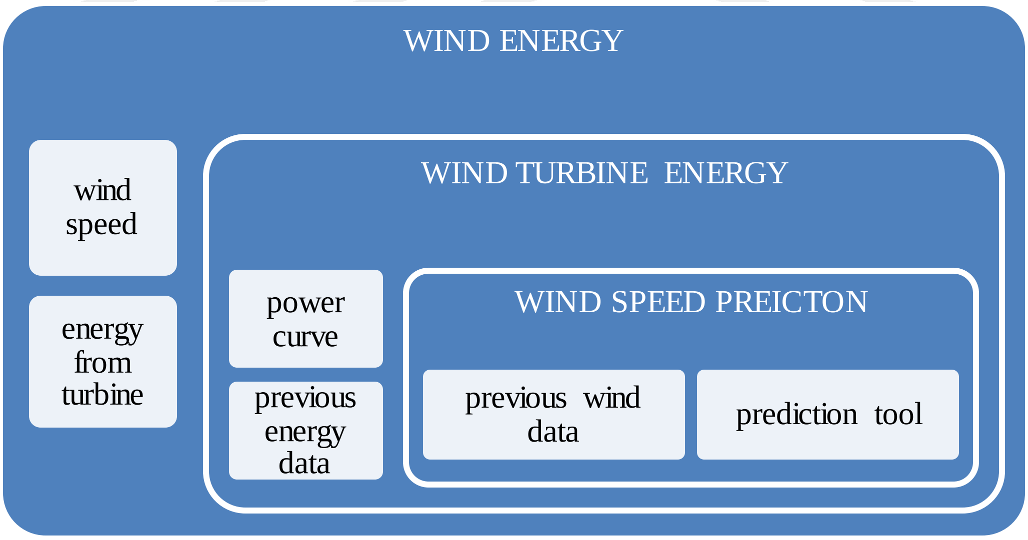

Figure 1.1 : Related topics and subjects that will be in project.

1.2 Literature Review

Because wind sources has a stochastic nature, wind speed forecast is fundamental in order to balance energy production with programmable sources in a complex electric grid, especially with the increase of wind farm installations. In order to calculate a possible wind energy capacity, first of all to do is making a prediction on wind speed. After making essantial prediction, a prediction for wind energy calculation also possible (Castellani et all., 2013).

Transmission and dispatching of electricity into the grid are submitted to very strong constraints, in order to maintain continuous and instantaneous balancing between energy entered and requested and to guarantee energy frequency and voltage into narrow range. Renewable energy dispatching has priority with respect to other kinds of energy. Respecting these constraints is awkward because of the variability of production, transmission and demand and due to the absence of storage systems. A program of injection and withdrawal is defined for each dispatching zone with 24 hours of advance: fees are assigned to the producers by the electric service provider due to mismatch between energy forecasted and actually entered into the grid. An error on the forecasted power is allowed but after the first 6 months of 2013 such limit is set to 10%. The accuracy on wind power production forecast on 24 hours basis is therefore fundamental in order to limit charges due to the unbalance (Castellani et all., 2013).

There are different approaches for wind speed and power forecasting. Wind speed and power forecasting are separated into three different approaches of physical forecasting approaches, statistical forecasting approaches and combination approaches. Moreover, statictical forecasting approaches can be separated into four different categories of conventional statistical approaches, artificial neural networks approaches (see Figure 1.2), ANN-fuzzy approaches and a Gaussian-Process-based method for forecasting (Jung and Broadwater, 2014).

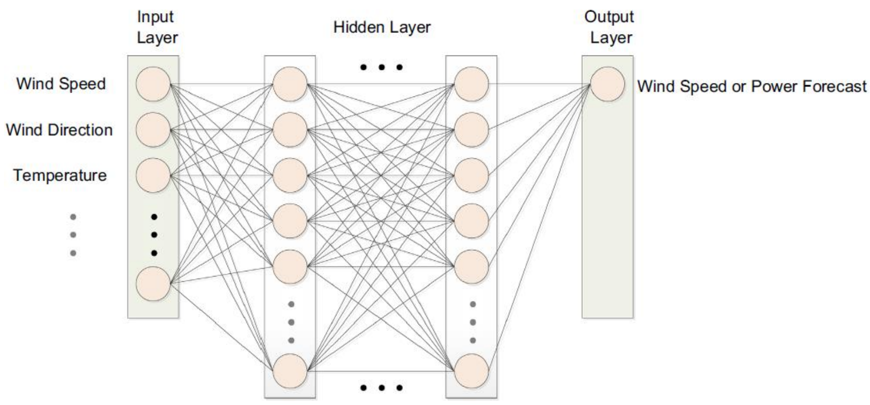

Figure 1.2 : ANN for wind forecasting (Jung and Broadwater, 2014).

In addition to this, prediction methods could also be divided into three different general models of physical methods, traditional statistical methods which is named also black box methods and learning approaches (see Figure 1.3). Different usages of these prediction methods basically lie in the large amount and the quality of historical data needed, which might not be available.

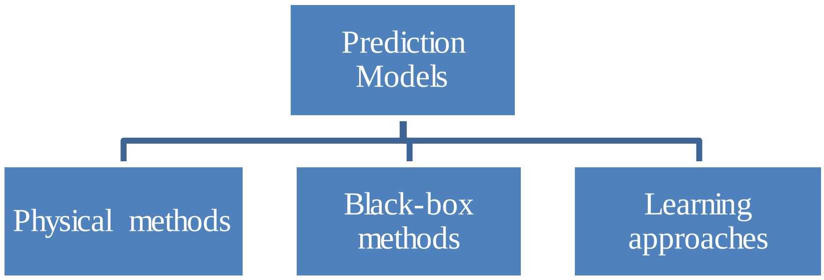

Figure 1.3 : Different types of prediction models (Castellani et all., 2013).

Prediction models could also be used together in order to make a prediction for wind speed and wind energy. Some important conventional wind power forecasting methods are one-two-three equation method, Kiranoudus method, polynomial modelling of wind turbine power curve and random number generation method (M. Jafarin, 2010). Being a hybrid model, fuzzy modelling techniques and artificial neural networks were used to developed a model to estimate annual energy output for S-47 wind turbine in different regions by M. Jafarin and A.M. Ranjbar. Another example for a hybrid system is combining different methods to forecast wind power. Lingling Li, Minghui Wang, Fenfen Zhu and Chengshan Wang have used ARMA model to forecast wind speed and artificiasl neural Networks to forecast wind power (see Figure 1.4).

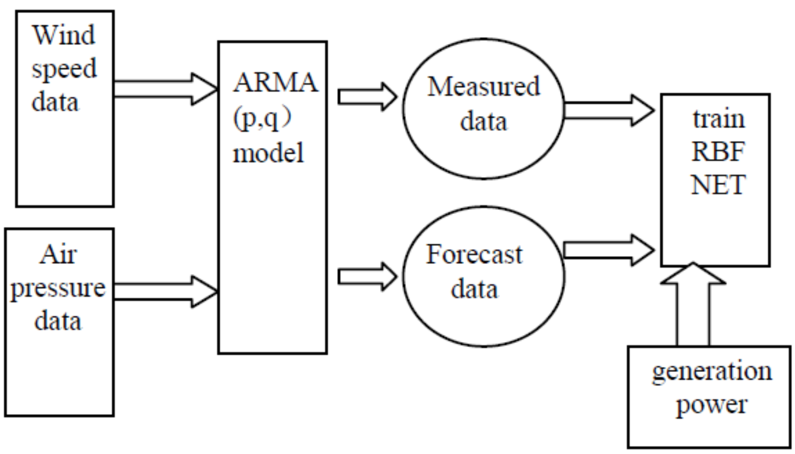

Figure 1.4 : Algorithm of forecasting wind speed and power (Li et all., 2009).

As mentioned before, artificial neural networks could be applied in order to make estimation for wind speed data and different types of typical neural networks can be applied. Gong Li and Jing Shi has investigated three artificial neural networks for wind speed forecasting by using adaptive linear element, back propagation and radial basis function (see Figure 1.5). The performances of these three models were evaluated based on three metrics, namely, mean absolute error, root mean square error, and mean absolute percentage error. According to the results they reached, no single neural network model outperforms others universally in terms of all evaluations metrics. Moreover, the selection of the neural networks type for best performance is also dependent upon the data sources (Li and Shi, 2009).

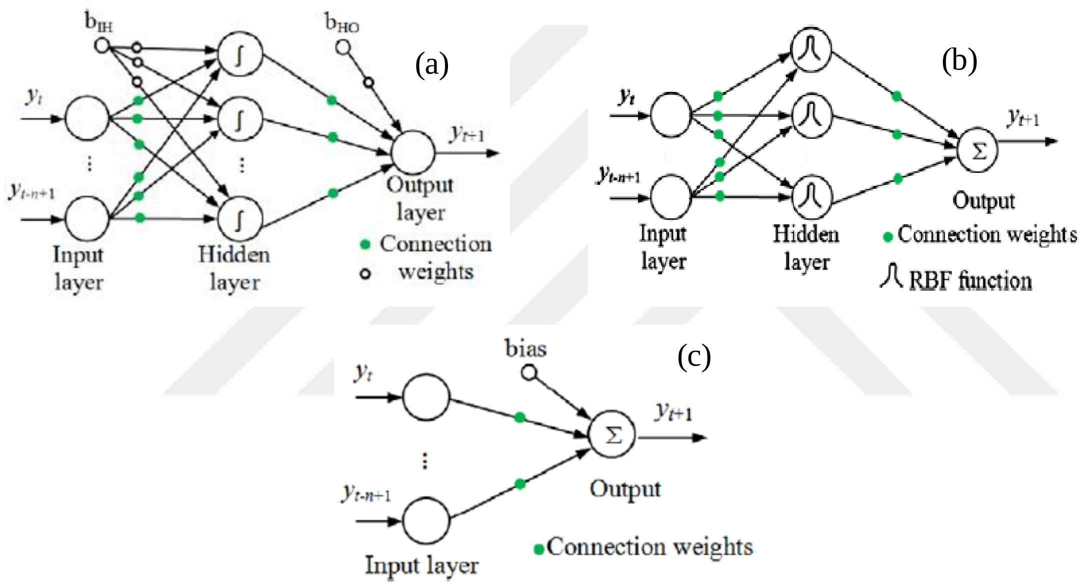

Figure 1.5 : Back propagation (a), RBF NN (b) and ADALINE network (c)

Artificial neural networks can use historical wind speed data in order to predict wind speed. On the other hand, if wind speed data are not avaliable for the related field, other informations could be used as an input for ANN like temperature, air pressure, solar radiation and altitude. Ramasamy, Chandel and Kumar Yadav have developed ANN model to predict daily wind speed for different location by using temperature, air pressure, solar radiation and altitude as inputs for ANN model. They have tested their ANN model by using mean absolute percentage error and they validated their ANN model by predicting wind speed for Gurgoan city for which measured data are avaliable with MAPE 6.489% and correlation coefficient 0.99 showing high prediction accuracy of the developed ANN model (see Figure 1.6).

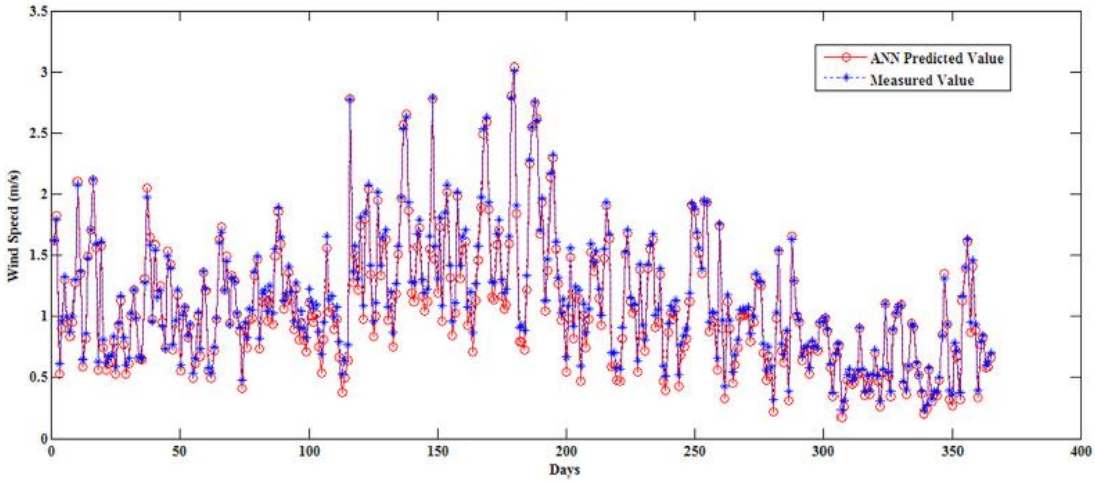

Figure 1.6 : Measured vs predicted wind speed (Ramasamy et all., 2015).

All in all, wind prediction consists of wind speed prediction and wind energy prediction. Within the purpose of providing a new hybrid wind prediction model, ANN model is applied by using Matlab ANN codes to forecast wind speed and WAsP software is applied to predict possible wind energy. At the end, some comparisons are made between predicted and observed wind speed data. Because wind energy is not avaliable for the related region, a comparison is made by using different wind speed data (observed, predicted) as input to WAsP.

2. TIME SERIES

2.1 Examples of Time Series

Analysing time series is dealing with records which are collected during time and these time orders are an important specification of time series. The collected records generally depend on each other. Time series examples will be given next but mainly, time series data can be collected hourly, daily, weekly, monthly, or yearly, and so on. Time series application can classified very different areas by the means of their usage. To give an example, the notification of $\{X_t\}$ or $\{Y_t\}$ $(t = 1, \cdot\cdot\cdot, T)$ can be used to denote a time series of length T. Following examples includes some real data which are generally used in the literature to illustrate time series modelling and forecasting (see Figure 2.1).

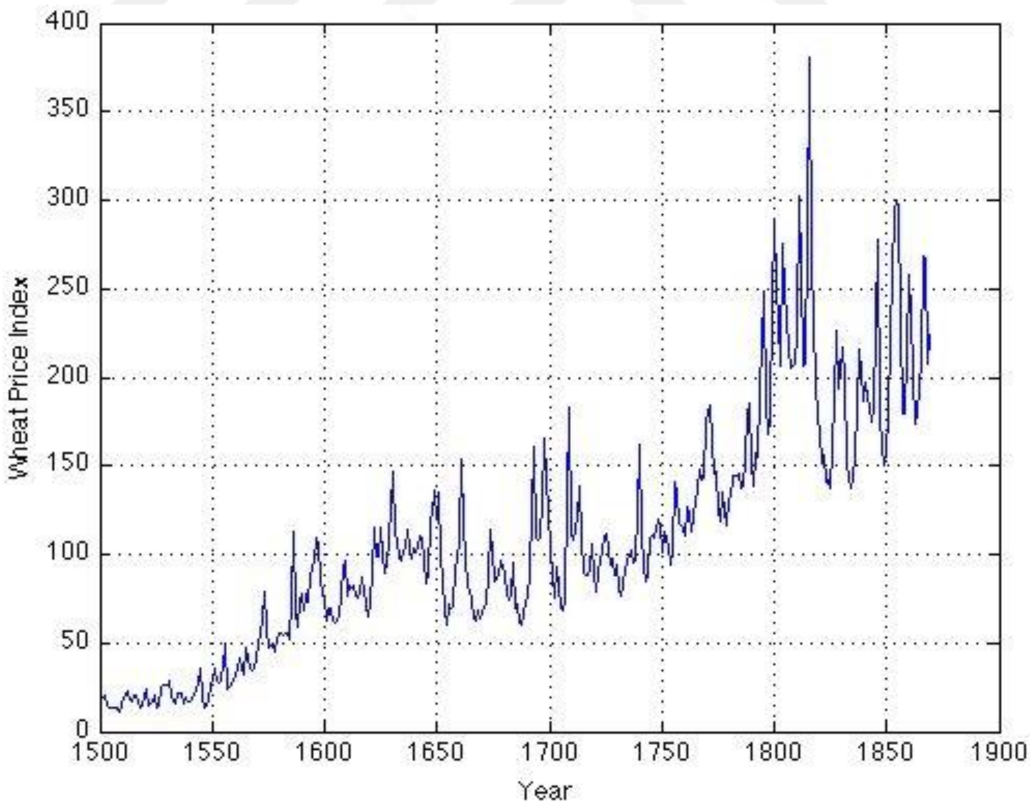

Figure 2.1 : Part of the Beveridge wheat price index series(Chatfield, 1995).

2.1.1 Sunspot data

The recording of sunspots dates back as far as 28 B.C., during the Western Han Dynasty. Let's X_t be Wolf's sunspot number's annual means or the number of sunspots in year 1770+t (see Figure 2.2). In the figure the sunspots numbers from 1770 to 1994 are plotted according to the time. Horizontal axis has the index of time t and vertical axis shows the the collected value of X_t during the time t. This is an simple example of time series and such a plot is called as time series plot.

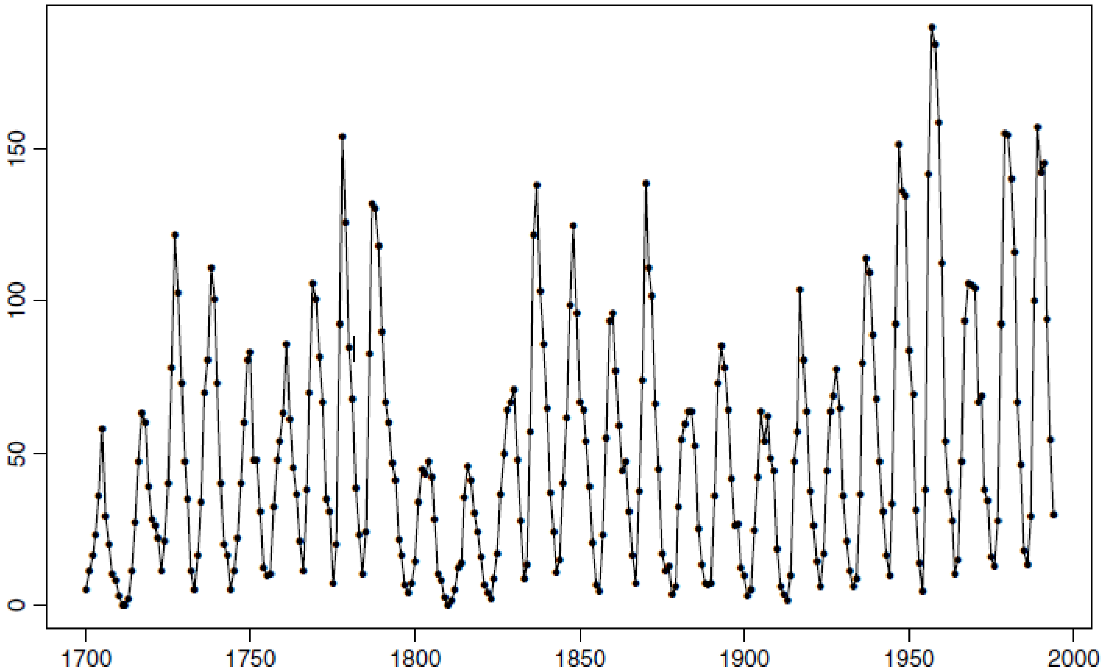

Figure 2.2 : Annual means of Wolf's sunspots from 1700 to 1994(Tong, 1990).

2.1.2 Canadian Lynx Data

The related data set consist of the annual fur returns of lynx at auction in London by the Hudson Bay Company for a period of 1821-1934 (see Figure 2.3). It is sign of the annual numbers of the Canadian lynx trapped in the Mackenzie River district of northwest Canada and reflects to some extent the population size of the lynx in the Mackenzie River district. The plot give us some ecological information about the related field. One of the background information about the data is if the proportion of the number of lynx being caught to the population size remains approximately constant, after logarithmic transforms, the differences between the observed data and the population sizes remain approximately constant (Fan and Yao, 2003).

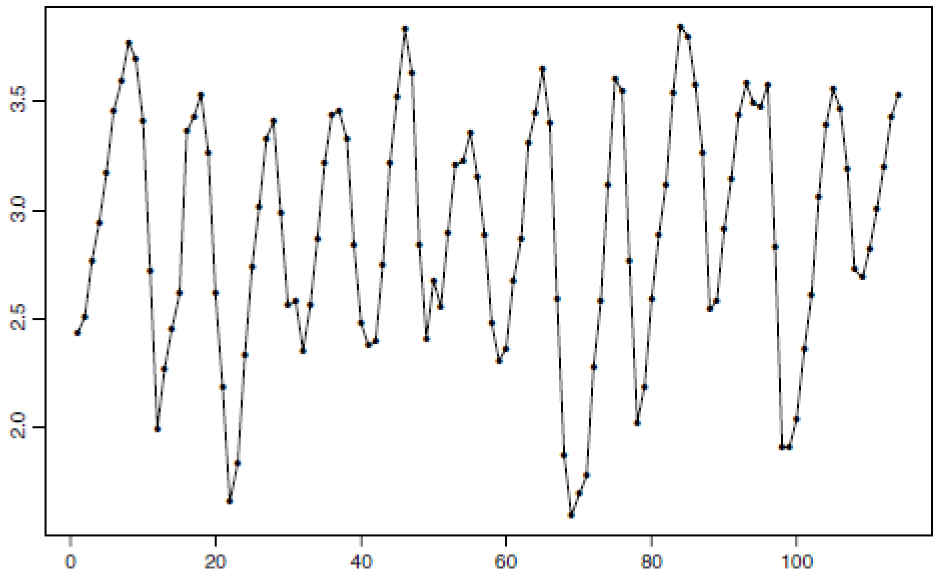

Figure 2.3 : Lynx trapped over the period of 1821-1934 (Fan and Yao, 2003).

2.1.3 Signal processing-decelaration during car crashes

One of the most appearing areas of time series is in signal processing. Vehicle crash signals are considered as an example. A microprocessor-based controller performing an algorithm on the digitized output of an accelerometer accomplishes airbag deployment during a crash. Generally, accelerometer is mounted in the passenger compartment of the vehicle. It experiences decelerations of varying magnitude as the vehicle structure collapses during a crash impact.

Acceleration time series of the vehicle could be seen at 1.25 milliseconds per samples (see Figure 2.4). On the other hand, readings of acceleration are very small when driving is going on normally. On opposite to this, readings are very high according to the severity of the crashes when vehicles are crashed or driven on very rough and bumpy road.

An airbag is deployed according to the standards. Because of this, not all crashes result with an airbag deployed. Federal standards define minimum requirement for crashing situations, which include speed and barrier types. In addition to these standards, automobile manufacturers also provide more requirements for the airbag systems according to the experiments, which are applied during manufacturing process. According to the experiment, manufacturers decide whether a crash needs to trigger an airbag, according to the severity of injuries. According to the previous reading and current reading, dynamical decisions are made for deploying airbag or not.

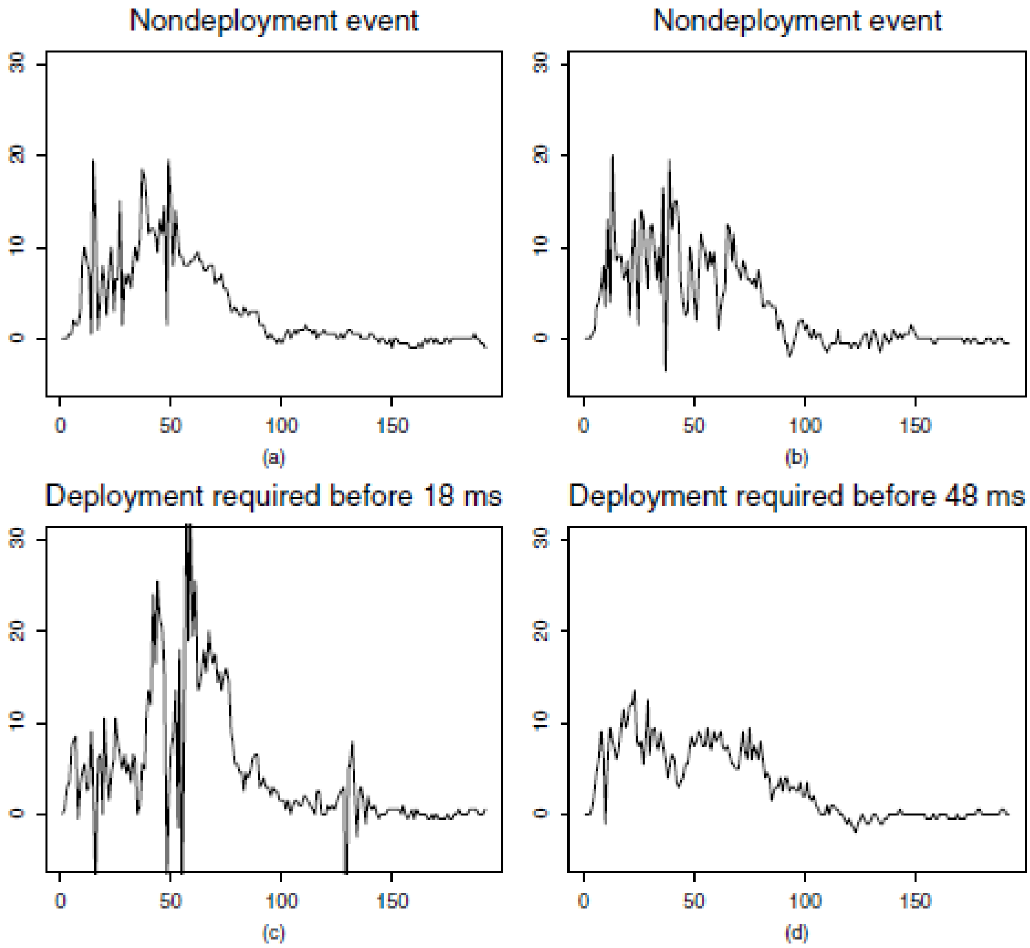

Figure 2.4 : Signals recorded during crashes of four vehicles (Fan and Yao, 2003).

The acceleration (in *a*) is plotted against time (in milliseconds) after crashes. The top panels are the events that require no airbag deployments. The bottom panels are the events that need the airbag triggered before the required time (see Figure 2.4).

Given examples above are just a few of multiple time series. Time series could be observed in very different areas of astronomy, biology, economy, environmental studies, finance and engineering area (Fan and Yao, 2003).

The purpose of using time series is depending on the backgroung of the applications. As an example, statisticians usually view a time series as a realiziation from stochastic process. A fundamental task is to unveil the probability law that governs the observed time series. With such a probability law, we can understand the underlying dynamics, forecast future events, and control future events via intervention. Those are the three main objectives of time series analysis (Fan and Yao, 2003).

In this study wind speed data recorded over time will be used in order to make predicted data by using artifical neural networks. The lenght of the time series data and specification of the data will be given later.

2.2 Linear Time Series Models

Including purely autoregressive model (AR) and purely moving-average model (MA), an autoregressive moving average models (ARMA) is the most popular class of linear time series models. In order to model linear dynamic structures, to depict linear relationships and for linear forcasting, ARMA models are mostly used.

In addition to the model of ARMA, a different model of autoregressive integrated moving average model, shortly ARIMA model, is also used. This model includes *stationary* ARMA-processes as a subclass (Fan & Yao, 2003).

White noise processes, AR models, MA models, ARMA models and ARIMA models are classified under the models of linear time series. A basic information about these different models are given below.

2.2.1 White noise processes

A stationary time series model of $\mathcal{E}$ is said to be white noise if $Corr\ (\mathcal{E}_t, \mathcal{E}_s) = 0$ for all *all* $t \neq s$.

Where:

$\mathcal{E}_t$ = sequence of uncorrelated random variables with constant variance and constant mean.

One of the assuming is that constant mean value is zero. At the end of the plotting a white noise series example, it can be shown that a very erratic, unpredictable and jumpy behavior. In order to forecast future values of t, $\mathcal{E}$ does not help because $\mathcal{E}$ is uncorrelated (see Figure 2.5).

Spins of a roulette wheel could be accepted as an example for white noise series. White noise series themselves are quite uninteresting from a forecasting standpoint (they are no linearly forecastable), but they form the building blocks for more general models.

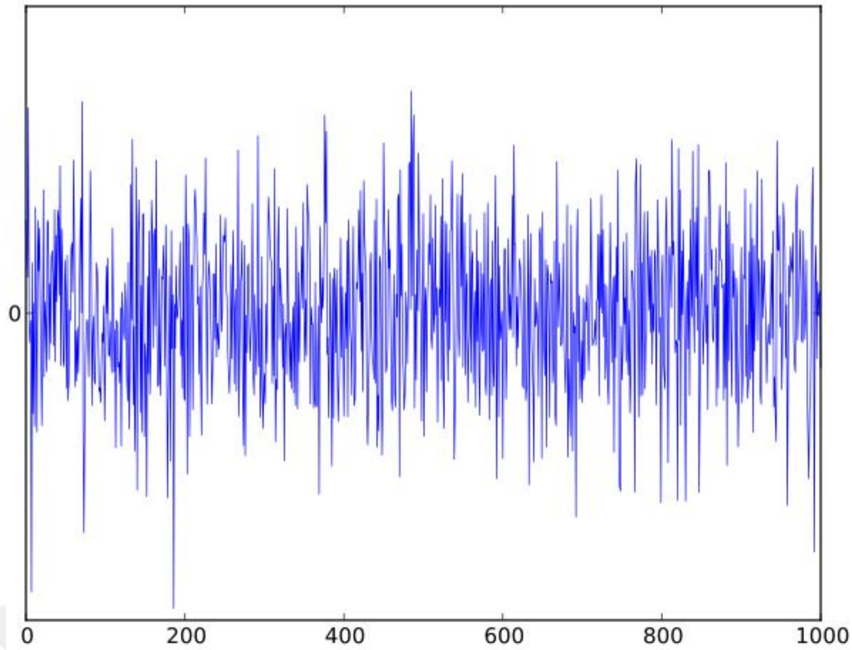

Figure 2.5 : Plot of Gaussian white noise signal.

2.2.2 AR models

An auto-regressive model is when a value from a time series is regressed on previous values from that same time series. As an example, y_t value on y_{t-1}.

In this type of models, previous time period's response variable become the predictor and the errors have our usual assumptions about errors in a simple linear regression model.

AR models has some orders values. The order of an autoregression is the number of immediately preceding values in the series that are used to predict the value at the present time. According to the expression, the preceding model is a first-order autoregression, written as AR(1).

Furhermore, let's y_t is current predicted value of this year. If y_{t-1} and y_{t-2} values of previous two years are used in order for making current year prediction, then the autoregressive model for doing so would be: $y_t=\beta_0+ \beta_1 y_{t-1}+ \beta_2 y_{2-1}$. This model is a second-order autoregression, written as AR(2), since the value at time t is predicted from the values at times $t-1$ and $t-2$ (see Figure 2.6).

More generally, a k^{t-h} order autoregression, written as AR(k), is a multiple linear regression in which the value of the series at any time t is a (linear) function of the values at times $t-1, t-2, \ldots, t-k$.

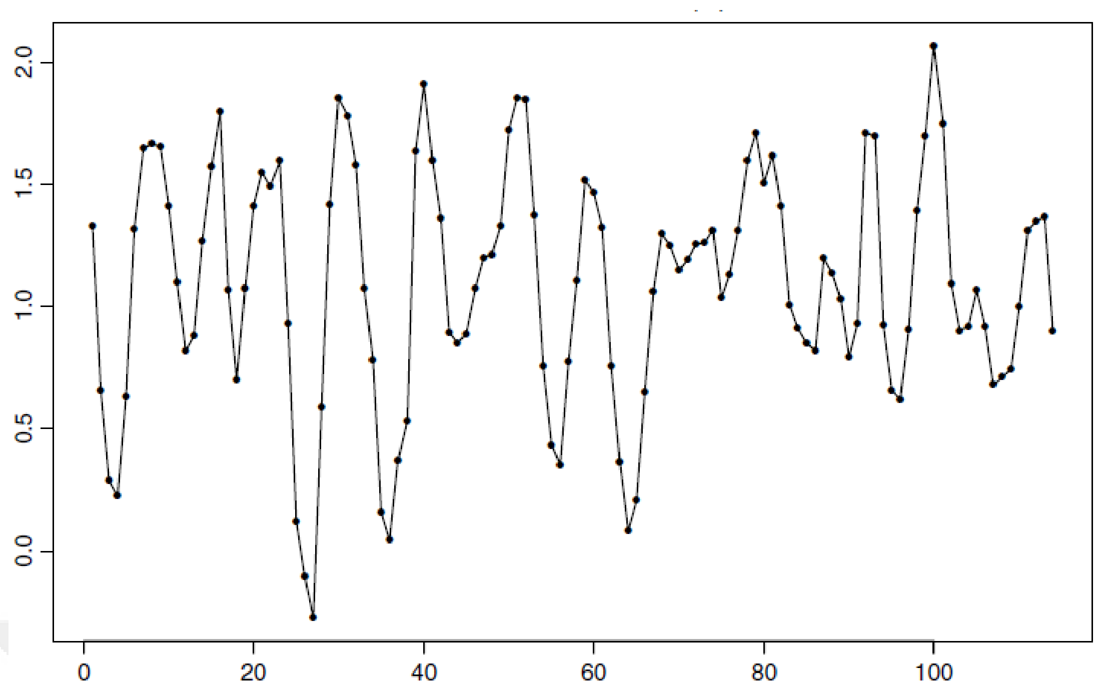

Figure 2.6 : AR (2) model X_t=1.07+1.35X_t-1-0.72X_{t-2}+$\mathcal{E}_t$ (Fan and Yao, 2003).

2.2.3 MA models

This third type of linear time series model of MA expresses as moving average of a white noise process. A white noise is difficult to observe, therefore the application of MA models are more difficult than AR models.

Rather than use past values of the forecast variable in a regression, a moving-average model uses past forecast errors in a regression-like model.

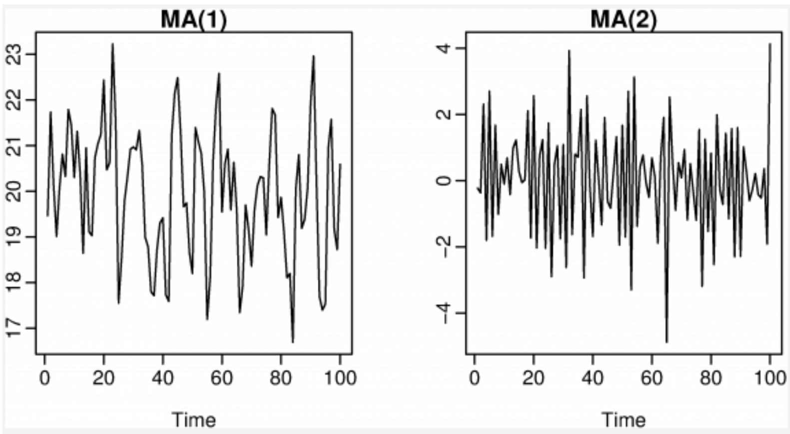

Figure 2.7 : MA(1) with y_t=20+e_t+0.8e_{t-1}. MA(2) y_t=e_t-e_{t-1}+0.8e_{t-2}.

Figure 2.7 shows some data from an MA(1) model and a MA(2) model. Changing the parameters' results in different time series patterns. As with autoregressive models, the variance of the error term e_t will only change the scale of the series, not the patterns.

2.2.4 ARMA and ARIMA models

Combining AR model and MA model forms together, a more complicated dynamic time series models of autoregressive moving average model (ARMA) is obtained. In general, this form of combined ARMA model can be used to model a time series with fewer terms overall than either an MA or an AR model by themselves.

Both ARMA and ARIMA models assume that time series is stationary but it is not a general case. Before appliying such a model, it is essantial to remove trends and periodicity. Because they exists in many case. This removing operation should be carried out until there is no obviouse trends or periodicities. Different processes are described thanks to the order of 1, 2, 3, … etc. ARIMA models include AR, MA and I. The letter in the ARIMA model refers to the fact that the dataet have been initially differenced (cf. differentiation) and when the modelling is complete the results then have to be summed or integrated to produce the final estimations and forecasts (Box, M. Jenkins, & Reinsel, 2008).

2.3 Nonlineer Time Series Model

Gaussian time series models have developed and dominated both practical usage and theoritical explorations thanks to the leading work of Yule on AR modelling of the sunspots number in 1927 and the study of Box and Jenkins on ARMA modelling in terms of theory and methodology in 1970.

Although some different models of ARMA has been developed like long-range dependence with fractionally integrated ARMA (Granger and Joyeux 1980, Hosking 1981), multivariate VARMA and VARMAX models (Hannan and Deistler 1988), and random walk nonstationarity via cointegration (Engle and Granger 1987), original ARMA model has been guarded its popularity for nearly four decades.

Because original ARMA model and other models developed depending on original ARMA model have simplicity, feasibility and flexibility, they all could be used in the analyzing of time series by a continuous usage.

As mentioned previous chapter, one important paper written by P.A.P Moran in 1950's on Canadian lynx data shows boundary of linear models. He drew attention to the "curious feature" that the residuals for the sample points greater than the mean were significantly smaller than those for the sample points smaller than the mean. This, as we now know, can be well-explained in terms of the so-called "regime effect" at different stages of population fluctuation (Tong, 1990).

Gaussian time series models have not covered modelling the regime effect or other non-standard features. It is important that a stationary purely nondeterministic Gaussian process is always linear. In other words, those nonstandard features are accepted as nonlinear features. Some examples for nonstandard features are nonnormality, asymmetric cycles, bimodality, nonlinear relationship between lagged variables, variation of prediction performance over the state-space, time irreversibility, sensitivity to initial conditions, and others. Many real time series data include nonlinearity (Tong, 1990).

Nonlienar time series analysis is seperated into three different subtitles of ARCH model, threshold model and nonparametric autoregressive model. A simple model for nonlinear time series could also be categorised but in this thesis project it does not covered. Background information for other three susbtitles are given below.

2.3.1 ARCH model

First example for nonlinear time series models is an *autoregressive conditional heteroscedastic model* which is know as *ARCH model*. This model is defined as in the equation **(2.1)**.

$$X_t = \sigma_t \mathcal{E}_t \quad \text{and} \quad \sigma_t^2 = a_0 + b_1 X_{t-1}^2 + \cdots + b_q X_{t-q}^2 \qquad \textbf{(2.1)}$$

Where:

$a_0 \geq 0$, $b_j \geq 0$ and $\{\mathcal{E}_t\}$ ~IID (0, 1).

In order to model conditional variance or volatility, Engle has developed ARCH model in 1982. This type of time series models are mostly used in financial applications because financial time series models have larger quantities and larger quantities mean larger variances.

In addition to this development, Boolerslev has developed a *generalized autoregressive conditional heteroscedastic* model, which is shortly GARCH model (see Figure 2.8). Therefore, equation **(2.1)** is replaced by equation **(2.2)** which is shown below.

$$\sigma_t^2 = a_0 + a_1\sigma_{t-1}^2 + \cdots + a_p\sigma_{t-p}^2 + b_1X_{t-1}^2 + \cdots + b_qX_{t-q}^2 \tag{2.2}$$

Where:

$a_j \geq 0$, $b_j \geq 0$.

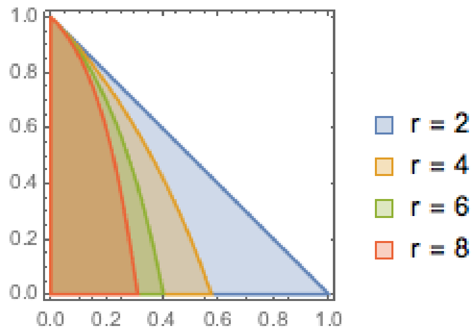

Figure 2.8 : First few even moments for a weakly stationary GARCH process.

2.3.2 Threshold model

Second important type of nonlinear time series model is threshold model. Threshold autoregrssive model is shortly known as *TAR* model. TAR model has been developed by H. Tong who assumes different linear forms in different regions of the state-space (Fan & Yao, 2003). This type of time series model is in the form of equation **(2.3)** below.

$$X_t = b_0^{(i)} + b_1^{(i)}X_{t-1} + \cdots + b_p^{(i)}X_{t-p} + \varepsilon_t^{(i)}, \ \ if\ X_{t-d}\epsilon\Omega_i \tag{2.3}$$

For i=*1,...k*, where $\{\Omega_i\}$ forms a *nonoverlapping* partition of the real line.

And $\{\varepsilon_t^{(i)}\}$ ~ IID $(0,\sigma_i^2)$.

On the other hand, there is a more simplier model of thresholding model. This model is the two-regime (i.e. k = 2) TAR model with $\Omega_1 = \{X_{t-d} \leq \tau\}$, where the threshold τ is unknown.

Two regime TAR(2)-model has been illustrated in the equation **(2.4)** below. (ref.Tong, 1990)

$$X_t = \begin{cases} 0.62 + 1.25X_{t-1} - 0.43X_{t-2} + \mathcal{E}_t, & X_{t-2} \leq 3.25 \\ 2.25 + 1.52X_{t-1} - 1.24X_{t-2} + \mathcal{E}_t, & X_{t-2} > 3.25 \end{cases} \qquad \textbf{(2.4)}$$

Where:

$\mathcal{E}_t \sim N\,(0, 0.2^2)$

And

$\mathcal{E}'_t \sim N\,(0, 0.25^2)$.

2.3.3 Nonparametric autoregressive model

The last form of nonlinear time series for this thesis is nonparametric ayroregressive model. Nonliner time series models could be infinite possible forms. Because of that reason, it is not possible to use one specific form of model to all data. At this point a natural model form could be used by nonpaarmetric approach and this assuming could be shown in the equation **(2.5)** below.

$$X_t = f_1\left(X_{t-1}, \dots, X_{t-p}\right) + \sigma\left(X_{t-1}, \dots, X_{t-p}\right)\mathcal{E}_t \qquad \textbf{(2.5)}$$

Where:

$f(.)$ and $\sigma(.)$ are unknown fractions.

3. WIND PREDICTION

3.1 State of Art Wind Prediction

There are two important prediction subjects about wind prediction. These two predictions are wind speed prediction and wind power prediction. From these two predictions, wind speed prediction is important for weather forecasting and working on meteorology. Many numerical weather predictions (shortly NWP) tools are already developed and in used for prediction of wind speeds for changing locations. Following figure (see Figure 3.1) shows different possibilities in wind prediction. A very important criterion for the assessment of the accuracy in the energy yield from a wind farm is the used prediction tool (Lange et all., 2009).

3.1.1 Wind speed prediction

Prediction on wind speed's importance has mentioned before. As prediction of wind speed, the prediction methods of wind speed are also an important situation. The more detailed input we have the more accurate results from the choosed prediction method could be obtained. Some of the important inputs that lead to the formation of wind speeds are atmosphoric temperature, pressure and humidity for a prediction tools in order to predict something in the future.

One of the specification of wind speed is it is a non-linear fluctuating function. This causes, normal prediction methods cannot be applied for forecasting in wind speed. At this point, a nonlinear method must be applied to make prediction about wind speed in future. The next figure included one of the nonlinear prediction methods of artificial neural networks. The technique of solving a nonlinear problem is a method of using the intelligent engineering represented by a neural network, a genetic algorithm, a chaos fractal, etc. These techniques are already adopted as numerical prediction, prediction of the weather, etc.

One of the fact about the wind energy fields that most of the wind farm locations are always not in the same location of the meteorological measurement station. This is called the *horizontal interpolation* "from the grid points to the coordinate of the turbine"). On the other hand, the height of the used wind turbine is not the same height of the meteorological measurement station and this measurement is called *vertical interpolation*.

Because of the fact that mentioned above, the role of the prediction tools could help for transferring the measured data (may including the predicted data from NWP) to the studying area of interest. This prediction method is shown in the following Figure 3.1 within the order of constant time range/vertical and/or horizontal and which lead to the fact that essantial predicted data (which would be measured or predicted) at the related area of interest (Rabunal and Dorado, 2005).

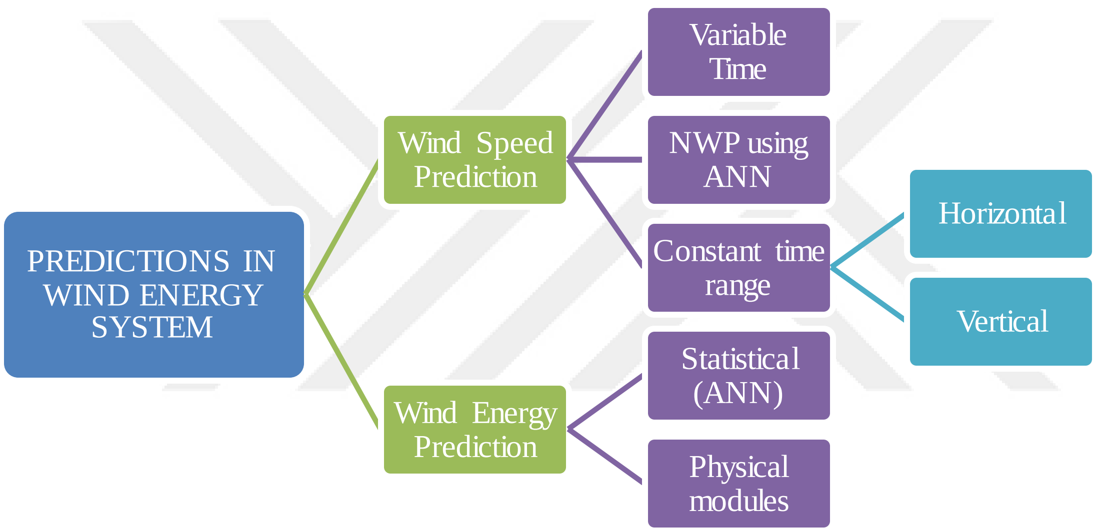

Figure 3.1 : Different methods of prediction in wind energy(Khatib, 2011).

Figure 3.1 shows the different possibility of using prediction in wind energy with the different prediction tools that can be used for every situations. A very important criterian for the assessment of the accuracy in the energy yield from a wind farm is the usage prediction tool. There are different methods which are used for wind speed and energy prediction, in this chapter we will talk about the strength and weakness of the different prediction methods used in this field.

3.1.2 Wind power prediction

A furthere prediction about wind energy is prediction of wind power. This is because of a possible energy output for a related area could be obtained by using some prediction methods. Energy output of wind is not only obtained by using some nonlinear prediction methods as ANN (which is also called a statistical method), but also some physical system.

3.1.2.1 Statistical model of wind power prediction

Statistical models are mainly based on training with the recorded measurement data from a related field of area. The idea is to derive a statistical relation between the given input from the weather prediction and the measured power output of wind turbine (farms). Because of that, statistical systems completely rely on data analysis ignoring the meteorological details. One of the statisticial model of ANN model could be used for analysing avaliable measurement data.

Many different methods for determining the relation between forecast and power output have been developed and one of the very prominent example is the system WPPT by the Danish Technical University. Additional example for statistical model is the system developed by ISET from Germany which provides forecasts for a number of German TSOs. This system works on artificial neural networks (ANN) which are trained with either historical wind farm data or measurements from transformer stations where a number of wind farms are connected. The advantage of that system of German is that providing an online estimation of the wind power that is currently fed into the electrical grid based on extrapolating measurements at representative wind farms (Khatib, 2011).

Main advantage of statistical system is clearly that the predictions are inherently adapted to the location of the wind farm such that systematical errors are automatically reduced. On opposite to this advantage, the need for long-term measurement data and an additional effort for the training is a problem. Another disadvantage of this system is to predict correctly rare atmospheric conditions if they appear too seldom during the training period. Unfortunately, a correct prediction of these rare situations is rather important and can otherwise lead to large forecast errors (Khatib, 2011).

3.1.2.2 Physical model of wind power prediction

Physical systems use some equations to obtain wind power. Additionally, a detailed physical description of the lower atmosphere's parameterizations is used in this prediction systems. Equation **(3.5)** shows the power outputs of a wind turbine under the related atmospheric conditions (Burton et all, 2011).

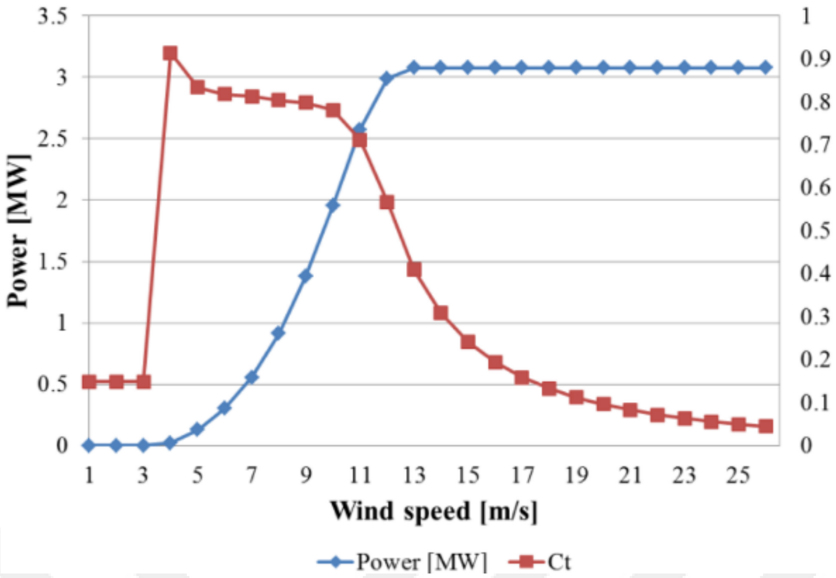

Figure 3.2 : C_p and C_f examples of a wind turbine (Castellani et all., 2013).

According to the Figure 3.2, there is a defination of thrust coefficient. In order to calculate thrust coeefficient, equation **(1.1)** is used (Sanderse, 2016).

$$C_T = \frac{T}{\frac{1}{2}\rho u_\infty^2 A_d} \tag{3.1}$$

In a non-dimensional form:

$$C_T = 1 - b^2 = 4a(1-a) \tag{3.2}$$

Where:

a is the axial induction factor,

$$a = 1 - \frac{u_d}{u_\infty} \tag{3.3}$$

and b is:

$$b = \frac{u_d}{u_\infty} \tag{3.4}$$

$$P = \frac{1}{2} c_p \rho v^3 A \qquad \textbf{(3.5)}$$

In the equation (3.1):

P: Wind power,

c_p: Coeffient of performance,

ρ: Density of air,

v: Wind speed,

A: Swept area of the wind turbine.

Because of the equation above, data is about the density of air in the related location, wind speed in that location and swept area of the choosen turbine have to known to calculate related wind power at related field.

Another important knowledge about wind is to know the related energy yield per month or year. For this purpose, wind power multiplication with time is essential which is given by equation **(3.2)**.

$$E = P \; x \; t \qquad \textbf{(3.6)}$$

Where:

E: Energy yield (per month/year, etc.),

P: Power output

t: Time (month, year, etc.)

For a capability of the calculation of resulted power output of a wind turbine in the future for a specific time, it is essential to know related wind speed value (at the same excat time) in the future. For example NWP, a prediction tool could be used for that purpose.

3.2 Electricity Marketting Law and Importance of Wind Prediction

A law about electricity marketing has been announced in 20th February in 2001 with the name of Electricity Marketting Law (law no.: 4628) to ensure the development of a financially sound and transparent electricity market operating in a competitive environment under provisions of civil law and the delivery of sufficient, good quality, low cost and environment friendly electricity to consumers and to ensure the autonomous regulation and supervision of this market (Electricity Marketting Law, 2001).

The scope of this law covers generation, transmission, distribution, wholesale, retailing and retailing services, import, export of electricity; rights and obligations of all real persons and legal entities (see Figure 3.3) directly involved in these activities; establishment of Electricity Market Regulatory Authority and determination of operating principles of this authority; and the methods to be employed for privatization of electricity generation and distribution assets (Electricity Marketting Law, 2001).

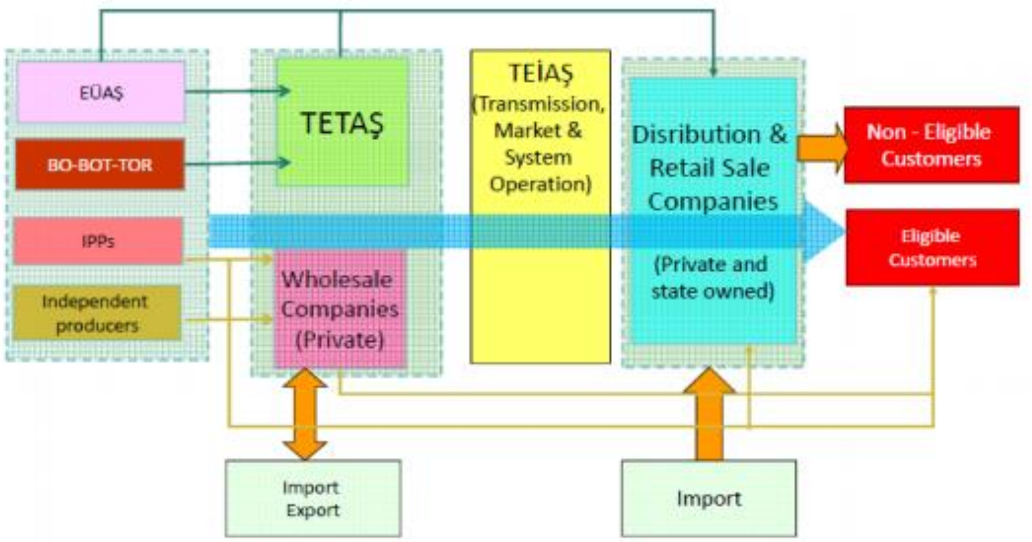

Figure 3.3 : Marketting structure.

In the years of 2000’s, two big problems have been faced during the distrubiton of electricity to customer. These are leakage of electricity and high rate of operational costs. Companies, which produce electricity, must overcome the problems mentioned above; these companies must have targets. Taking the reference of these companies’ targets, they could have profit or they lose money because of not reaching their targets.

Electricity marketting tariffs can be classified six different categories of retail energy sale tariffs, distrubition system usage tariffs, retail service tariffs, transmission fee, counter reading tariffs and illegal usage of tariffs. Retail energy sale tariff is the sale cost of the unit eletrical power purchased from producers in the unit of (kr/kWh) according to Turkey electricity market.

Electricity marketting tariff could be divided into two separate categories. These categories are one-period electricity marketting tariffs and multiple periods' marketting tariffs.

In the one period electricity marketting tariffs, customers pay same money during day hours. To put it other way, producers pay same amount of money per kWh electricity during 24 hours a day.

On the other hand, customers like business centers and industrial buildings pay different amount money according to the day periods. Because electricity cost changes during daytime, these tariffs are named multiple periods' marketting tariffs. Mutiple tariffs mean of three distinguish periods during a day. These periods are

Day period 06:00-17:00

Peak period 17:00-22:00

Night period 22:00-06:00 and customer pay different amount of money according to these sperated periods (Büyükyıldız, 2012).

3.3 Market Financial Settlement Center and Cost Analysis

All producers within the scope of interconnection have to give information about their production capacity for the following day. Additionally system operators have to be give information about essantial amount of electrical energy.These two essentiallity are necessary in order to provide a balance between producers and customers. Following figure shows supply and demand logic between producers and customers.

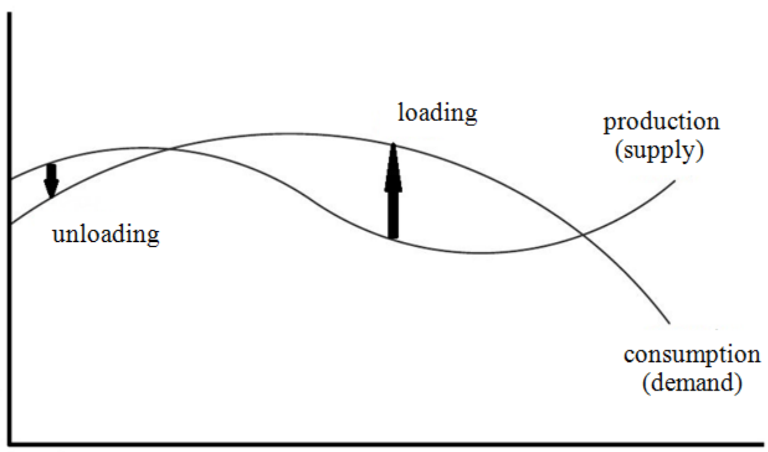

Figure 3.4 : Main logic of electrical energy marketting.

The main purpose of the market is to make a balance between supply and demand by covering supply and demand curve on each other. All these activities are generally named by balancing activities.

Market operator within the scope of the planning before the following day and system operator within the scope of the balancing power market, instructions that are given to decrease the production of energy is named by unloading power and instructions that are given to increase the power production is named by loading power (see Figure 3.4).

Market participations have to give the following informations before the following day:

- Daily production plan
- Bilateral agreements
- Customers consumption

Market participations have to give informations about their final values of daily production plan, their capacity, amount of loading and unloading power and their cost values (see Figure 3.5).

Making a good prediction on production capacity is very important for companies in order not to have loss after all. Therefore choosing a prediction is very mportant subject for supply companies. In order to make a prediction, collected time series of wind speed values could be used on artificial neural networks to have predicted values of wind speed. Furhermore, these predicted values could be used in either manual calculations or WAsP pragram.

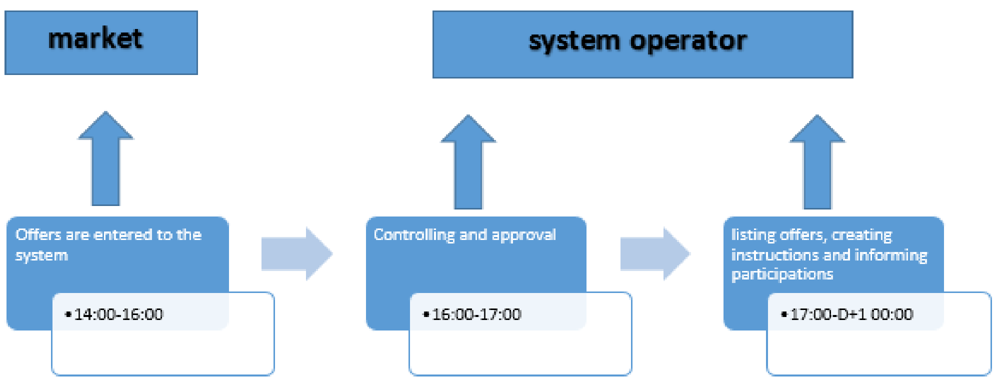

Figure 3.5 : Stabilizer Power marketting system.

4. ARTIFICIAL NEURAL NETWORKS

4.1 What are Artificial Neural Networks

Artificial neural networks will shortly named ANN during the explanations. After recognizing the structure of human brain computation, a new studying area of artificial neural networks has been conducted. Currently, modern computers compute complex problems. Opposite to the modern computers, human brain has a processing system that solve very complex, nonlinear and parallel information in a different way from conventional computer systems (see Figure 4.1).

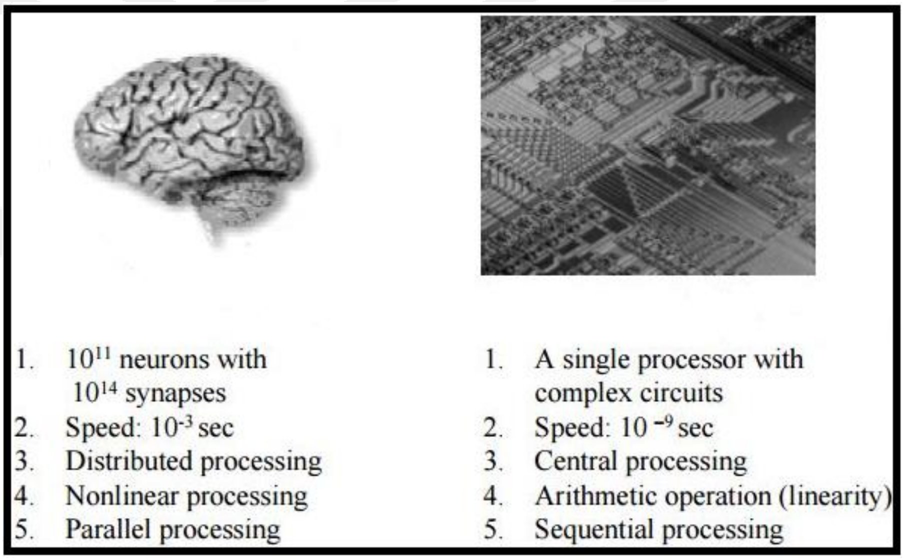

Figure 4.1 : Human brain vs Computer (Zhang, 2001).

Human brain has some structural compenents of neurons. Existing of these neorons provide a capability of performing a certain problems more faster than any of the conventional computer system today for human brain.

One of the main tasks of the human brain is accomplishing perceptual recognition tasks routinely. Being an example from life, human can recognize familiar face embedded in an unfamiliar scene.

A neural network is a structure, which is modelled according to the brain performing system on a particular task to solve and neural network can be applied using either some electronic parts or designening a model thanks to using of software on a computer. During this thesis study, a software named by Matlab will be used in order to make some prediction for future by using historical data. Neural network has an important property of storing studied experiments and making it available. A neural network is a massively parallel-distributed processor, which make up of simple processing units.

Artificial neural networks have also some properties in which they resemble to human brain. Firstly, through a learning mechanism knowledge is acquired. Secondly, similar property is interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge.

The term of learning algorithm is the methodology applied to perfom learning process (Khatib, 2011).

4.2 History and Benefits of ANN

However artificial neural networks modelling seems like a recent investigation, this studying field was established before the advent of computers, and has survived at least one major setback in several eras. Developments on artficial neural networks started in 1943's.

In 1943, neurophysiologist Warren McCulloch and mathematician Walter Pitts wrote a paper on how neurons might work including the subjects how neurons in the brain might work.

In 1949, Donald Hebb wrote *The Organization of Behavior*, which provides information that neural pathways are strengthened each time they are used. If two nerves fire at the same time, he argued, the connection between them is increased.

Thanks to the development of computers in 1950's, Nathanial Rochester from the IBM has developed a hypothetical neural network but the studying was failed.

In 1959, ADALINE and MADALINE models were discovered by Bernard Widrow and Marcian Hoff . The name of MADALINE comes from the usage of Multiple ADAptive LINear Elements.

ADALINE was developed to recognize binary patterns so that if it was reading streaming bits from a phone line, it could predict the next bit. The first artificial neural networks that have been used for a real problem is MADALINE model which has used an adaptive filter for eliminate echoes on phone lines. This developed system is still in use commercially although it is as ancient as earlier traffic control system.

Furheremore, in 1962, a learning procedure has been developed that check the value before the weight adjust it (i.e. 0 or 1) according to the rule before:

Weight Change = (Pre-Weight line value) * (Error / Number of Inputs).

Despite the later success of the neural network, traditional von Neumann architecture took over the computing scene, and neural research was left behind. Ironically, John von Neumann himself suggested the imitation of neural functions by using telegraph relays or vacuum tubes.

The idea of there could not be an extension from single-layered neural network to a multiple-layered neural network has been written at the similar period. Because most of the scientists in these studying fields were applied similar learning functions, research and funding went drastically down. Because, that learning function was flawed.

In addition to this problem, at that time, early successes of some neural networks led to an exaggeration of the potential of neural networks, especially considering the practical technology. As a result, promises have been unfulfilled.

One of the very appeling idea during development is that a computer would program itself. Thousands of bugs that the programming staff made could be repaired if Microsoft's Windows 2000 could reprogram itself.

In addition to the development above, Kohonen and Anderson have developed a network by using matrix mathematics in order to describe their ideas in 1972. The problem they both face is that what they were doing was creating an array of analog ADALINE circuits.

In 1975, first of the multilayered network has been created. Additionally, John Hopfield has written two highly readable papers on neural networks in 1982 and 1984 and these, together with his many lectures all over the world. Highly qualified scientists, mathematicians and technologists have persuaded these two papers to join the emerging field of neural networks.

The most important development about artificial neural networks has been faced by the publication of *Paralel Distributed Processing, Volumes 1 and 2* (PDP books) which have been edited by Rumelhart and McClelland (22. Referans) in 1986. After that publication, the first open conference was held in San Diego in the IEEE International Conference on Neural Networks about field of artificial networks in modern time series. Following this development, International Neural Network Society (INNS) was formed and the INNS journal Neural Networks was founded in 1988, followed by Neural Computation in 1989 and the IEEE Transactions on Neural Networks in 1990 (Yadav, 2015).

Table 4.1 shows a brief history about the development of artificial neural networks from 1890 to 1986.

Table 4.1 : Brief history of ANN (Zhang, 2001).

Date	Scientist	Contribution for Artificial neural networks
1890	William James	Describes (in words and figures) simple distributed networks and Hebbian Learning
1943	McCulloch & Pitts	Binary threshold units that perform logical operations (they prove universal computations!)
1949	Hebb	Formulation of a physiological (local) learning rule
1958	Rosenblatt	The Perceptron - a first real learning machine
1960	Widrow & Hoff	ADALINE and the Windrow-Hoff supervised learning rule
1969	Minsky & Papert	The limitations of perceptron – the beginning of the “Neural Winter”
1973	v.d.Malsburg	Self-organizing Maps
1980	Grossberg	Adaptive Resonance Theory
1982-84	Hopfield	Attractor Networks: A clean theory of pattern association and memory
1982	Kohonen	Self-organizing maps
1986	Rumelhart, Hinton, & Williams	Backprop

As computers, artificial neural networks have many important advantages for humanity. ANN can be used for different purposes such as classification, forecasting, control systems, optimization and decision-making, etc. Figure 4.2 shows the fields where artificial neural networks could be applied.

Forecasting	• weather and market trends prediction • electrical and thermal load predictions
Control systems	• adaptive control • robotic control
Optimisation and decision	• engineering systems management • analysis of electromyographs and other
Classification	• identification of mlitary target • identification of explosives passengers
Recognation	• pattern recognation • sound and speech recognation

Figure 4.2 : Different usage of artificial neural networks.

Other benefits of ANN are below:

- They are extremely powerful computational devices,
- They can learn and generalize from training data. Therefore, there is no need for enormous feats of programming,
- They are particularly fault tolerant, so this is equivalent to the "graceful degradation" found in biological systems,
- They are very noise tolerant, so they can cope with situations where normal symbolic systems would have difficulty,
- In principle, they can do anything a symbolic/logic system can do, and more.

4.3 Biological and Mathematical Model of ANN

Human nervous system can be divided into three different stages of receptors, neural network and effectors (see Figure 4.3). The response is given after passing all these stages from a human nervous system.

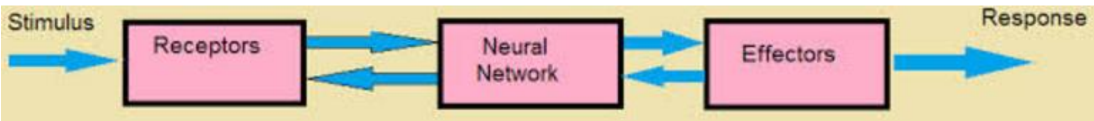

Figure 4.3 : Human nervous system's block diagram.

The first place of the human nervous system is receptor. The purpose of a recopter is to collect related information from the related environment. The last stage is effectors and at this stage interaction generations is happen. For example, activating muscles is a generation of interaction at the stage of effectors. In order to have an output form human nervous system there has to be a hierarchy (Kumar, 2009). Figure 4.4 shows the hierarchy of interwoven levels of organizations.

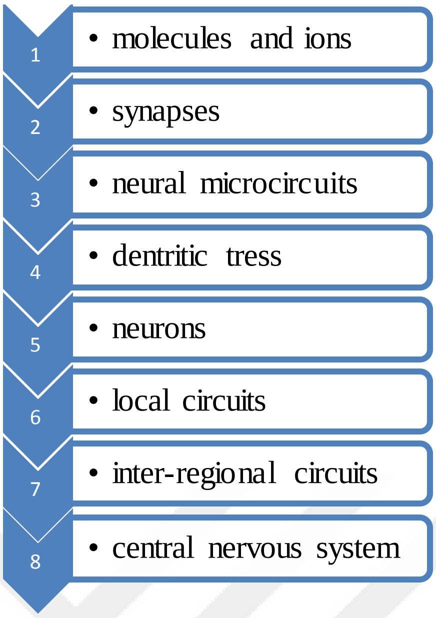

Figure 4.4 : The hierarchy of organization.

Approximately there are 10 billion neurons in the human brain and each of these neurons is connceted to many thousands of other neurons. Additionally, biological neurons typical speed is measured in milliseconds.

Activations and outputs are encoded by many of the neurons like a series of brief electirical pulses. There is an incoming activation to the neurons and the neurons take this incoming activation and convert it to a output activation by processing. Most types of the cell of neurons' nucleus contains the genetic material in the form of DNA (deoxyribonucleic acid).

Dendrites are fibers which emanate from the cell body and provide the receptive zones that receive activation from other neurons. In order to send activations to the other neurons, axons and fibers acting like tranmission lines. Synapses are the junctions which allow signal transmission between dentrites and axons.

Figure 4.5 shows a detailed information about complete neuron cell diagram that is created in 2007.

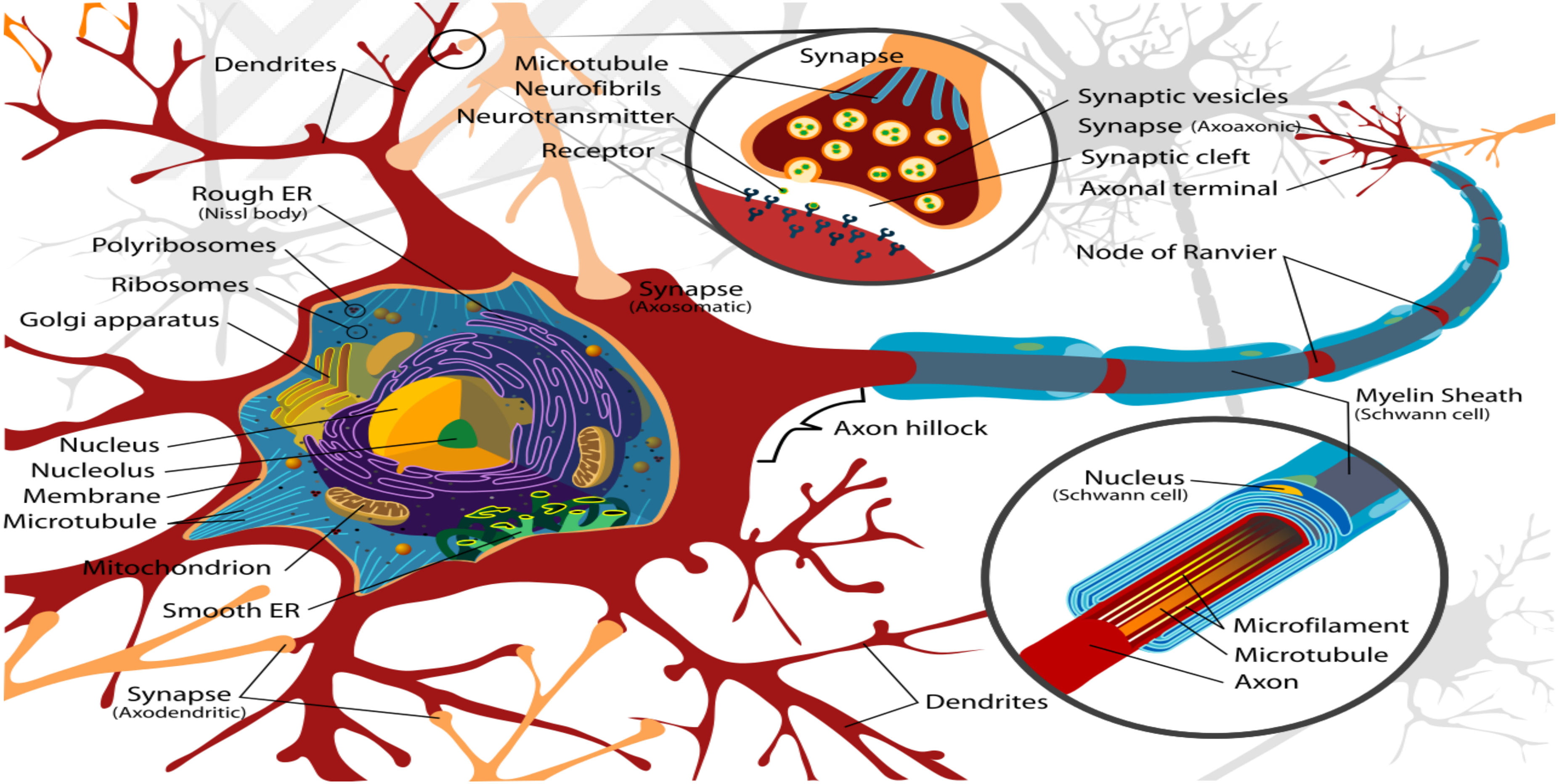

Figure 4.5 : Complete neuron cell diagram(LadyofHats, 2007).

Neural networks include neurons and a neuron is a fundamental information-processing unit for networks. Neural network's neuron has three important components. Three basic elements of a neuron are given below.

1. Weights: each of weights is separated by a strength of its own. The components are given below:

   x_p: signal

   k: neuron

   w_{kp}: weight

   The signal x_p connected to the neuron k and which is multiplied by weight w_{kp}. The weight of artificial neural networks could vary between ranges, which have both negative and positive values.

2. Summation of input signals: After calculating input signals, which are already weighted by weights of the neurons, an adder is essential to calculate the owerall summation of input signal.

3. Activation function: Neuron outputs have amplitude but this amplitude has to be limited by an activation function. An activation function could also referred to as squashing function. Activation functions squash the amplitude range of the signal output to some finite value. More detailed information about activation functions will be given next parts.

Figure 4.6 shows the mathematical modelling of artificial neural networks including three different parts.

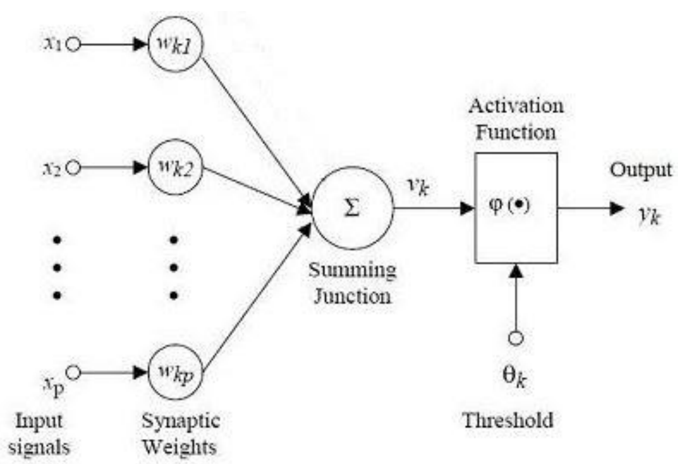

Figure 4.6 : Mathematical modeling of ANN (Martinez-Alvarez et all., 2015)

$$v_k = \Sigma_{\mathrm{j}}^{\mathrm{p}} w_{kj} \times x_j \quad \textbf{(4.1)}$$

Additionally,

$$y_k = \varphi(v_k + \theta_k) \quad \textbf{(4.2)}$$

According to the equations **(4.1)** and **(4.2)**, y_k is a function of v_k and θ_k. These equations sumerize the mathematical model of an artificial neural networks.

4.4 Network Architecture

As mentioned in the main title, there are three fundamental classes of network architectures. These are single layer feed forwards networks, multilayer feed forward networks and recurrent networks (Freeman and Skapura, 1991).

4.4.1 Single layer Feed forward networks

A feed forward type of network is that having an input layer of source nodes that projects onto an output layer of neurons, but not vice versa.

In a single layer feed forwards network, for example a simple perceptron, has one input layer and one output layer of processing units (see Figure 4.7). Because no mathematical calculations take place, input layer is not accpeted as an layer. In other words, in a single layer feed forwards networks, entering informations flows through input layer to output layer by crossing a hidden layer as well.

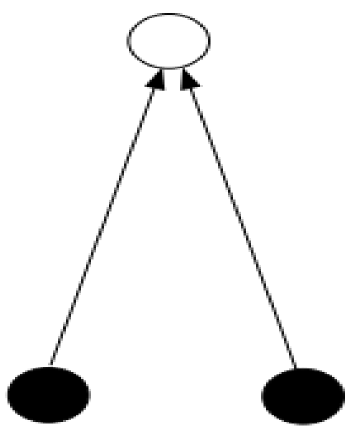

Figure 4.7 : Single-layer preceptron.

4.4.2 Multilayer Feed forward network

Being different from a single layer feed forward network, multilayer feed forward network has one input layer, one output layer and one or more hidden layers of processing units.

In this type networks, there is no feed-back connections as well. The hidden layers sits in between the input layers and output layers, and are thus hidden from the outside world. A multilayer perceptron could be an example for multilayer feed forward networks (see Figure 4.8).

One of the important steps of designing an artificial neural networks is choosing number of hidden neurons. Having more hiddens neurons provides for network a capability to perceive more input data and extract higher order statistics. The input signal is applied to the neurons in the second layer. The output signal of second layer is used as inputs to the third layer, and so on for the rest of the network.

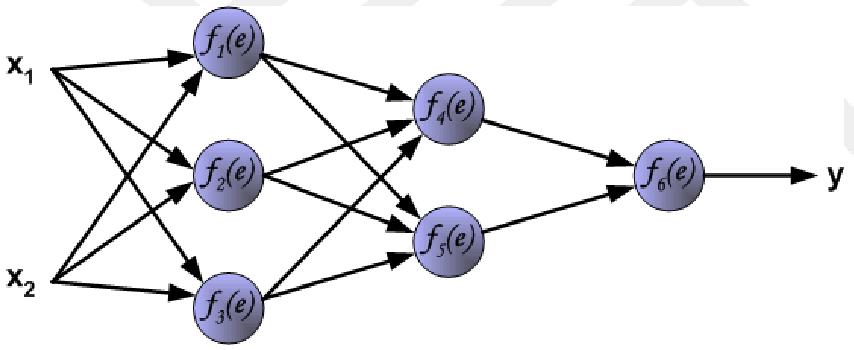

Figure 4.8 : Two inputs and one output three-layered neural network.

Figure 4.8 shows a three-layered network which has two inputs and one output. Additionally, the network has three different layer.

Compenent of the neural network shown in the Figure 4.8 is given below.

e: adder output signal

y=f(e): output signal of nonlinear

x_1 , x_2: input signals

y: output signal of the neuron

y': desired output

wmn: weight of connection between

yn: output signal of neuron *n*

Training data set consists of input signals of x_1, x_2 and corresponding target value (desired value) of *y'*. Training process of the network is made an iterative process. By using new data from training data sets, each iteration weights coefficient of nodes.

In addition to this, sysmbol *wmn* shows the weights of the connections between input neuron of *n* and output neuron of *m* in the next layer. Another sysmbol of *yn* shows the output signal of neuron *n*.

Following equations shows the connection in the neural network. Firstly output signal's equations from **(4.3)** to **(4.8)** are given.

$$y_1 = f_1(w_{11}x_1 + w_{21}x_2) \tag{4.3}$$

$$y_2 = f_2(w_{12}x_1 + w_{22}x_2) \tag{4.4}$$

$$y_3 = f_3(w_{13}x_1 + w_{23}x_2) \tag{4.5}$$

$$y_4 = f_4(w_{14}y_1 + w_{24}y_2 + w_{34}y_3) \tag{4.6}$$

$$y_5 = f_4(w_{15}y_1 + w_{25}y_2 + w_{35}y_3) \tag{4.7}$$

$$y_6 = f_6(w_{46}y_4 + w_{56}y_5) \tag{4.8}$$

The difference between desired output and the output from the network is called error signal and which has the symbol of δ. Error signal is an important value for networks, because it shows the performance of the designed artificial neural network. Following equations from **(4.9)** to **(4.14)** give the δ values.

$$\delta = y' - y \tag{4.9}$$

$$\delta_4 = w_{46}\delta \tag{4.10}$$

$$\delta_5 = w_{56}\delta \tag{4.11}$$

$$\delta_3 = w_{34}\delta_4 + w_{35}\delta_5 \quad \textbf{(4.12)}$$

$$\delta_2 = w_{24}\delta_4 + w_{25}\delta_5 \quad \textbf{(4.13)}$$

$$\delta_1 = w_{14}\delta_4 + w_{15}\delta_5 \quad \textbf{(4.14)}$$

There are three main important specification of a multilayer perceptron.

- Multilayer perceptrons contain one or more hidden neuron layer in order to learn and solve more complex problem.
- There is a high degree connection between network layers. This means that, a change in the connectivity results in a change in the network weights.
- Each neuron in a network has a nonlinear acvitation function. One of the commonly used functions is *sigmoid function* (see Figure 4.9) is given in the equation **(4.15)**.

$$y = \frac{1}{1 + e^{-x}} \quad \textbf{(4.15)}$$

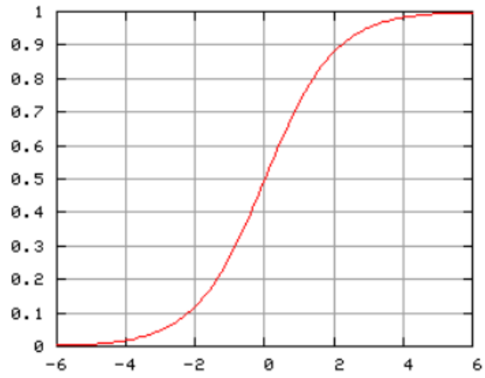

Figure 4.9 : Sigmouid function produce with an 'S' shape.

The other commonly used function is hyperbolic tangent function (see Figure 4.10) . A hyperbolic tangent function is given in the equation **(4.16)**.

$$y = \frac{1 - e^{-x}}{1 + e^{-x}} \quad \textbf{(4.16)}$$

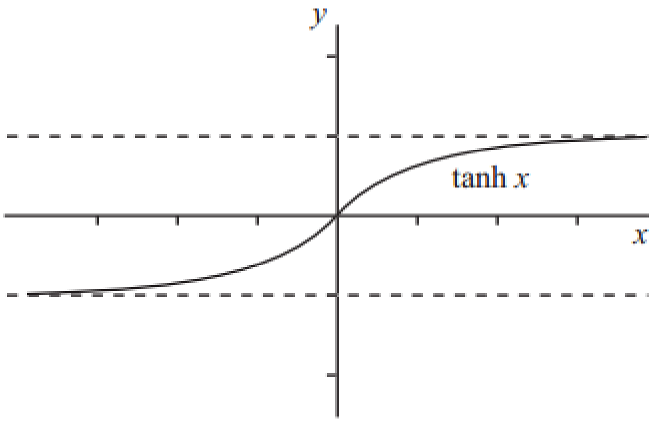

Figure 4.10 : Hyperbolic tangent function.

4.4.3 Recurrent networks

The last one of the ANN structures (or architectures/topologies) is a recurrent networks. These networks are any networks with at least one feed-back connection. The main different this type network from others is having feed-back connections (see Figure 4.11).

On the other hand, a recurrent network could have a hidden neuron or not. A simple recurrent network is an example for recurrent networks.

Recurrent networks could contain a single layer of neurons with each neuron feeding its output signal back to the inputs of all the other neurons. The output of a neuron is fed back to its own input and which is known as *self-feedback*.

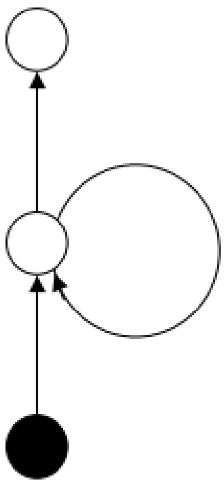

Figure 4.11 : Simple recurrent network.

The performance and learning capability of the networks are affected by the existence of feedback loops in a very important manner. To put it other words, the activation values of the units undergo a relaxation process such that the network will evolve to a stable state in which these activations do not change anymore.

Some of other interesting forms include; short cut connections, partial connectivity, time-delayed connections, Elman networks (see Figure 4.12), Jordan networks, moving windovs, etc.

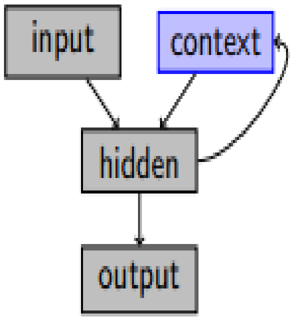

Figure 4.12 : Elman networks are MFNNs with an extra context layer.

4.5 Learning Process

The main purpose of an ANN is to provide a desired output according to the input data. An example is that having wind speed outputs according to the given data of historical wind speed data or pressure.

There are different methods in order to arrange the streghths of the connections. One of the method to adjust the weights explicity is using a prior knowledge. Other method is a training of neural networks. This training would be made by feeding neural networks teaching patterns and letting neural network to change its weights according to the some learning specifications.

To put it other way, a learning process is a procedure in order for modifying the weights and biases of a network. The main advantages of a learning procedure is to train a neural network for having good results from network.

Learning processes could be divided into three main headings of supervised learning, reinforcement learning and unsupervised learning. These there types of learning methods are generally used. On the other hand, learning algorithms could be divided into more different categories.

Figure 4.13 shows different types of learning algorithms. According to the figure, learning processes divided into four different parts including stochastic learning method of Boltzman Learning Rule.

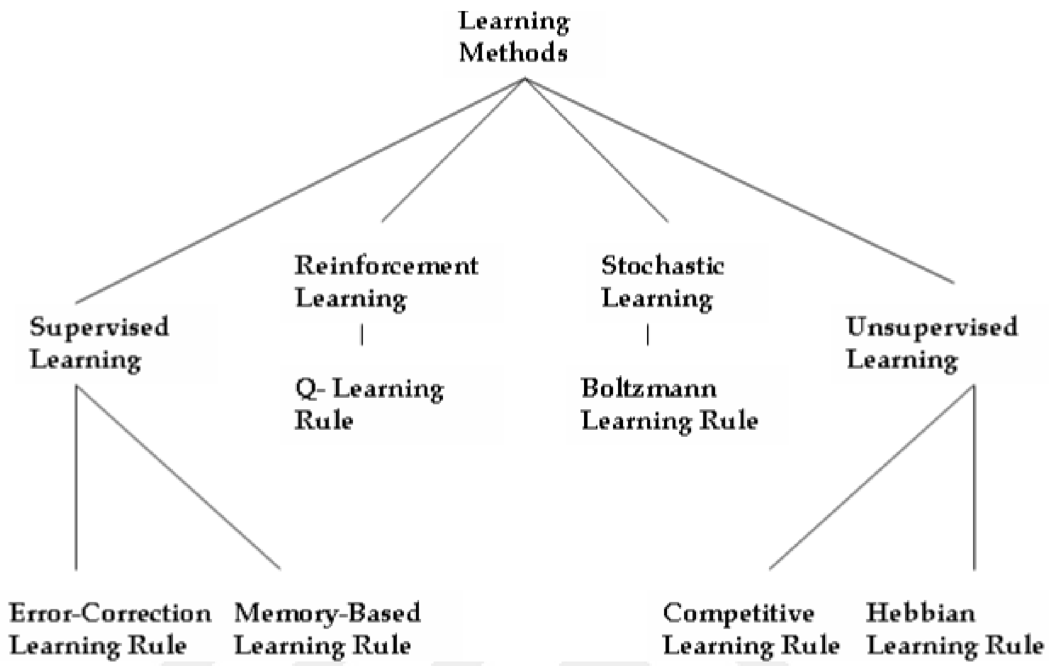

Figure 4.13 : Different learning methods of ANN (Sathya and Abraham, 2013).

4.6 Experimental Working Steps

According to the background information given before, wind speed prediction will be made by using arficial neural networks. Later, energy calculation will be experienced from the predicted value of wind speed. In order to build a neural network, some of the steps will be applied (see Figure 4.14).

1. Recording data and pre-processing
2. Conversion of the data and normalization process
3. Statistical analysis of ANN
4. Artificial neural networks designing
5. Traning the ANN
6. Testing the ANN

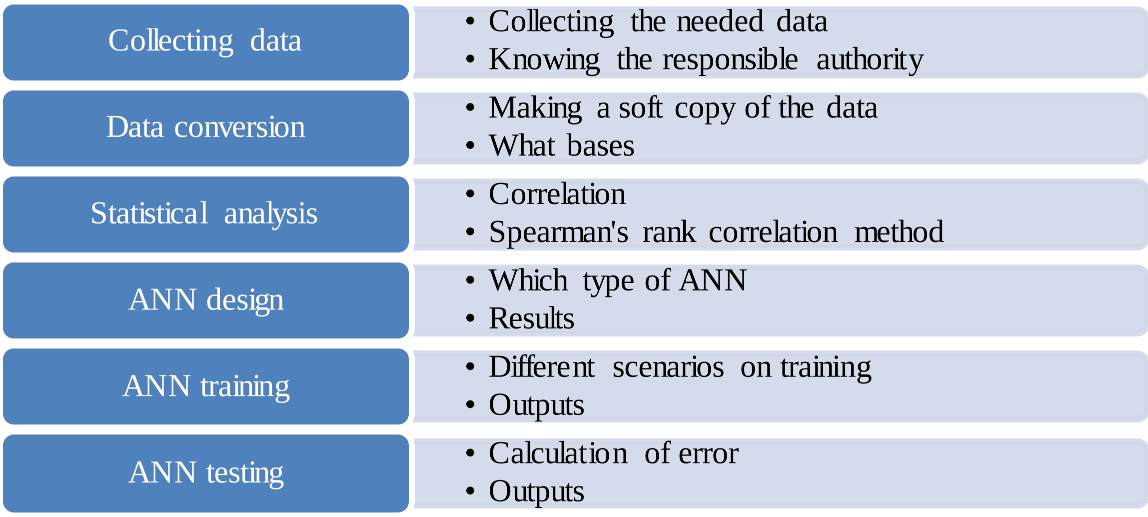

Figure 4.14 : Working steps on artifcial neural networks.

4.6.1 Recording data and pre-processing

This is the first level of the building an ANN. The main purpose of this stage is to collect essential data. For example, observed wind speed of related area was measured before; therefore, there is no need to wait to collect data information again for this study (see Table 4.2).

Table 4.2 : Observed wind speed and direction data from 2001 to 2004.

HOUR	2001 ITU MASLAK		2002 ITU MASLAK		2003 ITU MASLAK		2004 ITU MASLAK	
	speed (m/s)	direction(°)	speed (m/s)	direction(°)	speed (m/s)	direction(°)	speed (m/s)	direction(°)
1:00 AM	3.414	32.66	4.278	218.4	0.545	118.6	9.86	215.8
2:00 AM	3.486	33.7	5.305	215.5	0.2	143.3	8.21	213.4
3:00 AM	3.527	32.51	5.997	221	0.2	135.5	8.5	213.6
4:00 AM	3.062	26	6.569	214.9	0.2	111.9	5.299	205.1
5:00 AM	1.497	340.7	6.378	210.2	0.937	166.3	6.573	216.9
6:00 AM	1.464	318.6	5.794	206.5	0.68	147.2	5.89	218.5
7:00 AM	1.256	14.99	6.468	207.9	1.711	159.5	6.334	215
8:00 AM	1.932	59.07	5.381	217.2	0.324	112.5	4.944	223.4
9:00 AM	2.029	50.16	6.768	212.2	0.372	99.6	3.581	12.27

4.6.2 Conversion of data and normalization

After collecting (or finding) essential data, the second stage is to make a software copy of the data. If necessary, collected data are divided by its maximum value. One of the important subjects is the quantity of the data. This is because; data could be lost during the making of a software copy. For example, excel sheet has a maximum capacity in coloums and rows. On condition that the available data is bigger than the capacity of the excel sheet, then some of the data would be lost.

Total number of reading along one year is given in equatin **(4.17)**.

$$\frac{\#\, of\ reading}{10\, min} \times \frac{60\ min}{1h} \times \frac{24\ h}{1\ day} \times \frac{365\, day}{1\, year} = \frac{52560\ \#\ of\ reading}{1\ year} \qquad \textbf{(4.17)}$$

4.6.3 Statistical analysis

It is important to analysis the available data as the input data and target data. Target data is a measured data (for this project target data is measured wind speed) which would be wind speed, pressure and direction of the wind.

Without any normalization, using directly row data may result in good results. Nevertheless, using row data directly, cannot overcome the problems of incorrect measurement or unexpected wind changes in direction or speed. Because the effect of the problems is invisible (this is because of dealing with data between different years), row data could be used directly. The main subject is to abtain good results from the ANN.

On the other hand, the problems could be more visible when using the data of the same year with different heights for example.

4.6.4 Correlation analysis

Correlation analysis is a statistical tool that studies the connection between two variables (K. Sreelakshmi, 2008). It can be determined the degree of association or relationship between two or more variables by a statistical procedure of *correlation analysis*. The quantity of correlation in a sample of data is obtained by the sample coefficient of correlation, which is, generally, denoted by r or by ρ.

There are some different types of methods for correlation:

- Scatter diagram method or Dot diagram method.
- Karl Perason's coefficient of correlation method.
- Spearman's rank correlation method.
- Concurrent deviation.
- Two-way frequency table method.

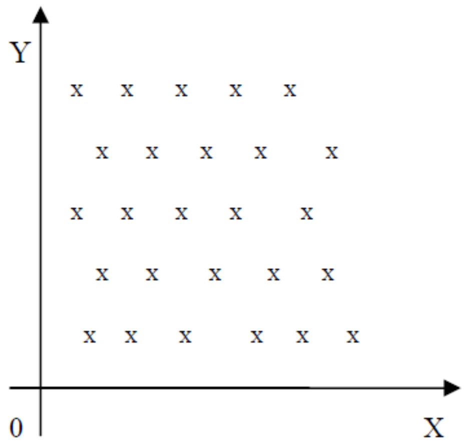

Figure 4.15 : An example for dot diagram method.

Figure 4.15 shows application of dot diagram method for correlation. There is no correlation on the graph.

Another method is two way frequency table. According to the Figure 4.16, the sum of the row total equals the sum of the column totals.

Joint Frequency

Marginal Frequency

	Sport Utility Vehicle (SUV)	Sports Car	Totals
male	21	39	60
female	135	45	180
Totals	156	84	240

Figure 4.16 : An example for two way frequancy table.

4.6.5 What is best for ANN tool

Designing artificial neural networks requires knowledges about training data, sizing the network and weightining-learning paarmeters. The performance of an ANN is directly related with its designer. Some backgroung information about these criteria will be given in this part.

4.6.5.1 Training data in ANN

1. Experience is one of the most important and essential properties because there is no a single rule to apply.
2. As many as data should be used during traning the network although not all of them are necessary. From the available data, a small subset is often used, remaning data can be used to test the network whether it can perform desired mapping on input vector has never encountered during training.
3. Training data should contain the entire expected input space. During the training process, training vector pairs are randomly selected from the set of inputs that is why in any event don't train the network completely with input vectors of one class, and then switch to another class because the network will forget the original training.
4. Output values have to be scaled when output function is *Sigmoidal*. The outputs of the network never reach 0 or 1 because of the specification of *Sigmoidal function*. As a result, 0,1 and 0,9 should be used as maximum and minimum values.
5. There are many such possibilities that depend largely on the problem being solved Sigmoid are often called "squashing" functions, because they compress an infinite input range into a finite output range.
6. As the input gets larger, Sigmoid functions are characterized by the fact that their slopes must approach. This causes a problem when you use steepest descent to train a multilayer network with sigmoid functions, because the gradient can have a very small magnitude and, therefore, cause small changes in the weights and biases, even though the weights and biases are far from their optimal values. The purpose of the resilient backpropagation (Rprop) training algorithm is to eliminate these harmful effects of the magnitudes of the partial derivatives (Khatib, 2011).

4.6.5.2 Network sizing of ANN

In general, three layers are sufficient to solve a problem. On the other hand, some of the problems could be seen easier (learning faster) than others. According to the nature of the application, the size of the input layer is generally dictated. By deciding analog values or binary values for output units, the number of nodes could be determined.

In addition, it could be also possible not to use any hidden nodes in the networks, which are superfluous. The weight on certain nodes changes very little from their initial values when the weights on the hidden nodes are checked periodically during training. It could be said that this certain nodes cannot be join the training any more, so fewer hidden units may suffice (Khatib, 2011).

Rumelhart has developed an otomatic method in order to separate unnecessary nodes from the neural networks.

4.6.5.3 Weightining and learning of ANN

The last but least important subject is weightining and learning of artificial neural networks. Weights should be initialized to small, random values (between +-0.5) as for the bias it is common to treat it as another weight. For learining rate parameter η, selection of value has very important effect for the performance of ANN. Usually η must be a small number (about 0.05 to 0.25) to ensure that the network will settle to a solution.

The small values of η is that ANN will have to make many of iterations and such a situation is important to be thinking about. It is often possible to increase the size of η as learning proceeds. Increasing η as the network error decreases will often help to speed convergence by increasing the step size as the error reaches a minimum, but the network may bounce around too far from the actual minimum value if η gets too large. To solve such a problem, momentum technique (to increase the speed of the convergence and to protect the right curse of the errors) is applied (Khatib, 2011).

Second solution for that problem is local minimum. According to the figure below, if the weightining changes to the minimum at a local point is different that global minimum, unacceptable high error is obtained. For example, Matlab software could calculate similar result to each other during the application of ANN, but these result could obtain high error from the exact result. This means that the global minimum point is not closed to the results which are obtained from the software.

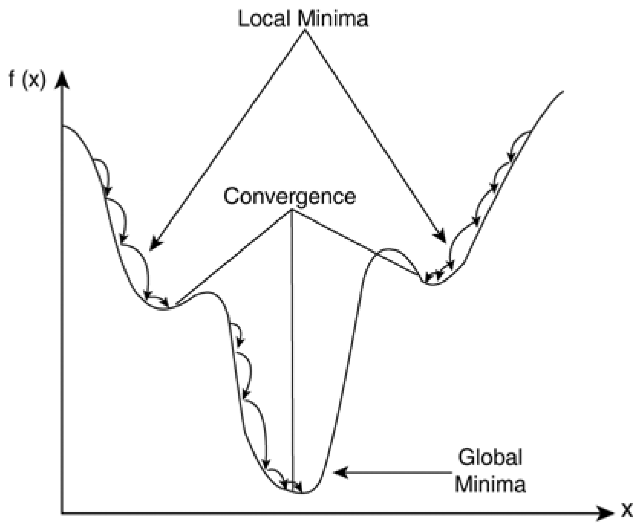

Figure 4.17 : Multiple local minima and one global minimum (Beck, 2014).

A network could stop learning before reaching a good solution. Changing number of hidden nodes or the learning parameters could fix the problem. On the other hand, when a network gives an acceptable result there is no guarantee that the network has reached the global minimum rather than a local one (see Figure 4.17).

On condition that the network gives a solution which is acceptable from an error standpoint, it does not matter whether the minimum is global or local, or even whether the training was halted at some point before a true minimum was reached.

4.6.6 Regression analysis

Let us consider there are two variables of x and y. On condition that y is influenced by x (changes on x affect the variable y), then a simple regression or linear regression is obtained between x and y. This simple regression is observed with the equation of y on x. There, y is known as dependent variables (or regressinon or explained variable) and x is known as independent variable (or predictor or explanatory).

In this project, future value of wind speed is a dependent variable and previos value of wind speed data is independent variable. Because, previous values of wind speed data affect the future value of the wind speed data.

Because regression shows a relationship between two variable’s average values, it is very favorable for estimating the average value of one variable from a given value of other variable. Best average value of one variable according to the other variable could be obtained by using of regression equation.

On the other hand, the estimation for one value can be made by using of regression line. Therefore, regression line of x and y is the line that provides the best estimation for the values of x for a given value of y (in this case, line of regression of x on y; otherwise, line of regression of y on x).

Some advantages of using regression is given below.

- Regression establishes the rate of change in one variable in terms of other one.
- Thanks to regression analysis, cause and effect relations are recorded.
- It could estimate the values of unknown quantities.
- Coefficient of regression could be determined by regression
- Nature of relation between variables could be studied tanks to regression analysis.
- Regression analysis could be used natural, social and physical science in which the data are in a fuctional relationship.

Figure 4.18 shows different types of regression models. Regression models contain two different categories of simple regression and multiple regression.

Simple linear regression (SLR) is a method that enables you to determine the relationship between a continuous process output (Y) and one factor (X).

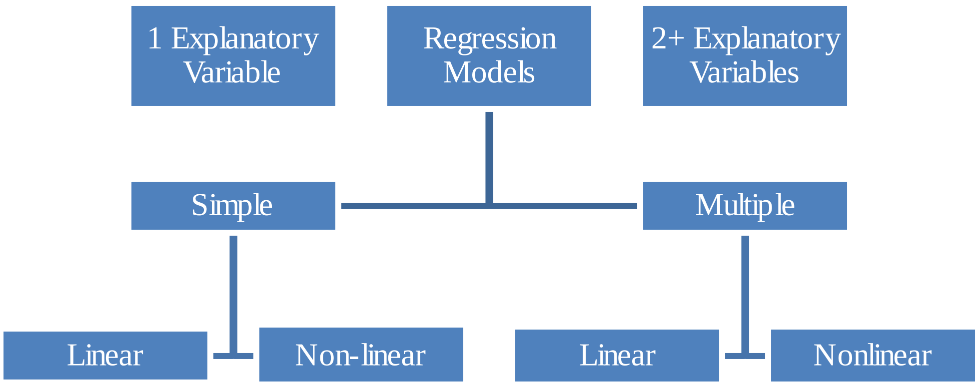

Figure 4.18 : Different types of regresssion.

The mathematical model of an SLR (see Figure 4.19) is given in the equation **(4.18)**.

$$E(y) = A + Bx + e \tag{4.18}$$

Where;

y: dependent variable,

x: independent variable,

A and B: regression coefficients,

e: random error.

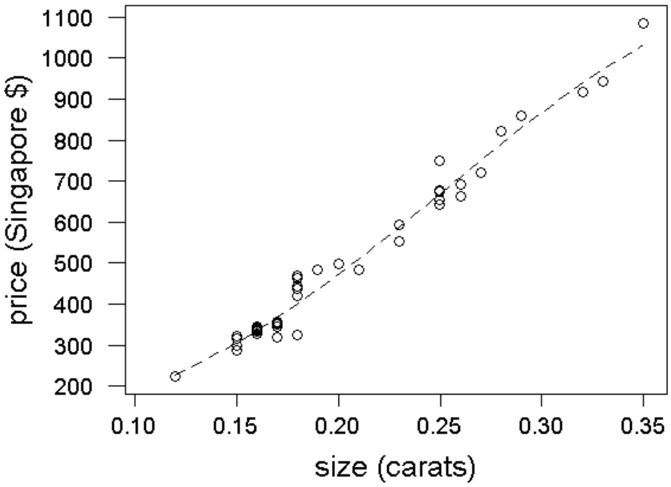

Figure 4.19 : Simple linear regression (SLR) mode.

On the other hand multiple linear regression (MLR) is in the form of given quation **(4.19)**.

$$E(y) = \beta_1 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + e \qquad \textbf{(4.19)}$$

Where;

y : dependent variable,

x_1, x_2, …, x_k : independent variables,

β_1, β_2,…, β_k: regression coefficients,

e: random error.

4.6.7 Training of ANN: Levenberg-Marquardt (*trainlm*)

One of the big disadvantages of using artificial neural networks is could be face in this step of traning of ANN. This is because of not having a direct low or formula in order to find exact number of hidden neurons. Additionally, right transfer function of the neural network could not be found without making any test.

A network's training occurs according to training parameters of *trainlm* and Table 4.3 shows default values of training parameters (Matlab-R2015a).

Table 4.3 : Default values of *trainlm* training parameters(Matlab-R2015a).

Parameter	Value	Explanation
net.trainParam.epochs	1000	Maximum number of epochs
net.trainParam.goal	0	Performance goal
net.trainParam.max_fail	6	Maximum validation failures
net.trainParam.min_grad	1e-7	Minimum performance gradient
net.trainParam.mu	0.001	Initial mu
net.trainParam.mu_dec	0.1	mu decrease factor
net.trainParam.mu_inc	10	mu increase factor
net.trainParam.mu_max	1e10	Maximum mu
net.trainParam.show	25	Epochs between displays
net.trainParam.showCommandLine	false	Generate command-line output
net.trainParam.showWindow	true	Show training GUI
net.trainParam.time	inf	Maximum time to train in seconds

According to the Levenberg-Marquardt optimization, *trainlm* (a network training function) updates weights and biases values. Although *trainlm* requires more memory than other algorithms, which is often the fastest backpropagation algorithm in the Matlab neural network toolbox, and is generally apllied as the first-choice supervised algorithm.

Training stops when any of these conditions occurs:

- The maximum number of epochs (repetitions) is reached.
- The maximum amount of time is exceeded.
- Performance is minimized to the goal.
- The performance gradient falls below *min_grad*.
- *mu* exceeds *mu_max*.
- Validation performance has increased more than *max_fail* times since the last time it decreased (when using validation).

Levenberg-Marquardt algorithm (like the quasi-Newton methods) was designed to approach second order training speed without having to compute the Hessian matrix. When the performance function has the form of a sum of squares (as is typical in training feedforward networks), then the Hessian matrix can be approximated as **(4.20).**

$$H = \mathrm{J}^T\mathrm{J} \tag{4.20}$$

The gradient could be computed as **(4.21)**.

$$g = \mathrm{J}^T\mathrm{e} \tag{4.21}$$

Where Jacobian matrix J contains first derivatives of the network errors according to the biases and weights, and e is a network error vector. Through a standard backpropagation technique, the Jacobian matrix could be computed and that is much easier than computing the Hessian matrix.

The Levenberg-Marquardt algorithm uses this approximation to the Hessian matrix in the following equation **(4.22)** Newton-like update:

$$x_{k+1} = x_k - [\mathrm{J}^T\mathrm{J} + \mu\mathrm{I}]^{-1}\mathrm{J}^T\mathrm{e} \tag{4.22}$$

When the μ is zero in the equation **(4.22)**, that is just Newton's method, using the approximate Hessian matrix. On the other hand, when the μ is large, then gradient descent becomes with a small step size. Newton's method is faster and more accurate near an error minimum, so the aim is to shift towards Newton's method as quickly as possible. Thus, μ is decreased after each successful step (reduction in performance function) and is increased only when a tentative step would increase the performance function. In this way, the performance function will always be reduced at each iteration of the algorithm (Matlab-R2015a).

4.6.8 Testing of ANN

In this step, the performance of ANN is tested. Spearman's rank correlation and regression analysis are essential to give a good idea about expected result of the used ANN tool.

In addition to this, one more method for accuracy of the ANN is also needed. Accuracy means to calculate the error of the prediction and be able to compare results for different prediction tools.

Error calculation is mostly studied by using root mean square deviation (RMSD) method. The next part will provide more detailed information about RMSD.

4.6.8.1 Root mean square deviation method

The Root Mean Square Error (RMSE) (also called the root mean square deviation, RMSD) is a frequently used measure of the difference between values predicted by a model and the values actually observed from the environment that is being modelled (see Table 4.4). These individual differences are also called residuals, and the RMSE serves to aggregate them into a single measure of predictive power (T. Chai, 2014).

The RMSE of a model prediction with respect to the estimated variable X_{model} is defined as the square root of the mean squared error which is given in the equation **(6.23)**.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(x_{obs,i}-x_{model,i})^2}{n}} \quad \textbf{(6.23)}$$

where X_{obs} is observed values and X_{model} is modelled values at time/place i.

RMSE's equation could also be shown in the equation **(4.24)**.

$$RMSE = \sqrt{MSE(x')} = \sqrt{\frac{\sum_{i=1}^{n}(x_i'-x_i)^2}{n}} \quad \textbf{(4.24)}$$

Where:

- n is the total number of reading (fort his Project, the number of measurement for instance)
- x_i' is the value of predicted parameters
- x_i is the value of predicted parameters

At the end of the step of testing with root mean square deviation, more accurate understanding of the output results are obtained, not only a general over view like before (T. Chai, 2014). Next Table 4.4 shows some RMES values.

Table 4.4 : Five sets-errors of size n (T. Chai, 2014).

n	***RMESs***
4	0.92, 0.65, 1.48, 1.02, 0.79
10	0.81, 1.10, 0.83, 0.95, 1.01
100	1.05, 1.03, 1.03, 1.00, 1.04
1000	1.04, 0.98, 1.01, 1.00, 1.00
10000	1.00, 0.98, 1.01, 1.00, 1.00
100000	1.00, 1.00, 1.00, 1.00, 1.00
1000000	1.00, 1.00, 1.00, 1.00, 1.00

4.6.8.2 Normalized root mean square error (NRMSE)

RMSE's could also be used in non-dimensional form. This is because, non-dimensional form of root mean square error provides comparisons in different units. There are two approaches for normalized data.

Normalize RMSE to the range of the observed data given in the equation **(4.25)**.

$$NRMSE = \frac{RMSE}{x_{obs,max} - x_{obs,min}} \tag{4.25}$$

Normalize to the mean of the observed data given in the equation **(4.26)**.

$$NRMSE = \frac{RMSE}{\bar{x}_{obs}} \tag{4.26}$$

Figure 4.20 shows plot of normalized root mean square error (NRMSE) against consistency threshold for different number of time intervals in a set (W. Karlen, 2014)

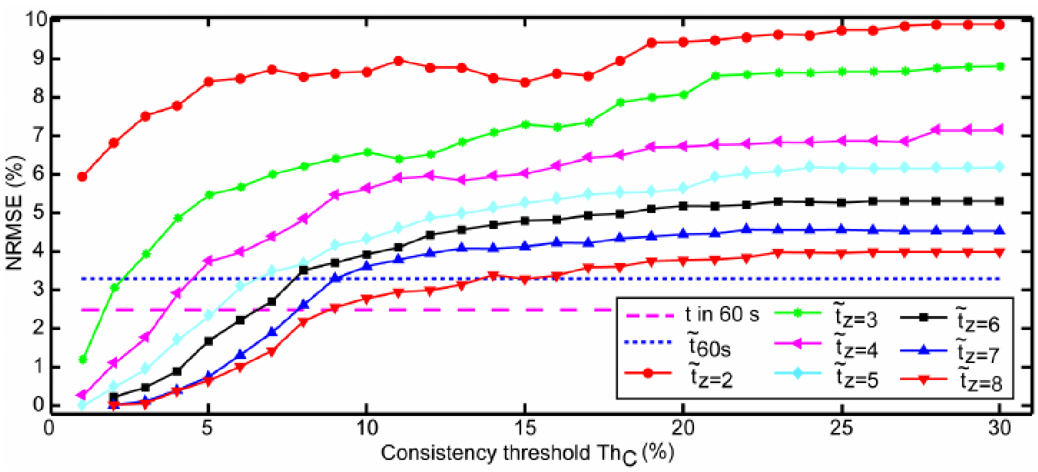

Figure 4.20 : Plot of NRMSE against consistency threshold.

4.7 Calculation of wind energy

At the end of the prediction of wind speed, then energy could also predicted by using different methods. Within the scope of this study, three different methods for wind energy prediction will be given. As mentioned earlier, the output of this wind speed prediction tool can be an input to an energy prediction tool (to predict the amount of energy generated by a wind farm).

4.7.1 Manual calculation

In order to make a manual calculation, the specific equation for wind energy yield of a turbine is used. The methodology of this method is to calculate total energy yield a wind turbine. After that, total output of a wind energy field could be calculated by using the output of one turbine. The equation for using in this methodology is given in the equation **(4.27)**.

$$P = \frac{1}{2} c_p \rho v^3 A \qquad \textbf{(4.27)}$$

In the equation (3.1):

P: Wind power,

c_p: Coeffient of performance,

ρ: Density of air,

v: Wind speed,

A: Swept area of the wind turbine.

4.7.2 Calculation with WAsP

Another method to calculate owerall energy output is to use WAsP software program. More detailed information about WAsP will be given next chapter. A brief information about the program is that WAsP uses wind speed and direction data in order to calculate energy output. Because it is essantial to choose a wind turbine in the program, a backgroung information about wind turbines is also necessary.

4.7.3 Calculation with Matlab

Essantial data for Matlab is ordered below:

- Related field wind speed data,

- Related wind turbines power curve,
- Other parameters from selected data.

After providing the inputs ordered above, Matlab could calculate energy output of a wind turbine (Khatib, 2011).

5. TECHNICAL STUDY OF THE PROJECT

This chapter of the study will be divided into two different parts. In the first part, artificial neural networks and WAsP software will be used. For second part, artificial neural networks are going to be applied again in order make a future prediction.

5.1 Performance Analysis of ANN and Energy Calculation by WAsP

5.1.1 Performance analysis of ANN

The purpose of this part is to make calculation on Matlab by using ANN codes to find best performanced networks. After finding the network, its results will be used to calculate wind energy. Because we will have both observed value and calculated value, we could make a comparison on energy output on WAsP by using these different wind speed values.

On Matlab, either ANN tool or ANN Matlab code could be used in order to design a network. For this study, Matlab ANN codes are used.

In order to find best result (which is the best value of perform), number of inputs and number of hidden layers are adjusted.

Input number of the network is adjusted from n=1 to n=7. For any number of the input value, three different numbers of hidden layer h (starting by h1=2xn+1) are calculated. The results of ANN on Matlab change when running the code. In other words, with same input and hidden layer values, Matlab code gives different results for perform. Therefore, the calculation is made for five times to find best result. Additionally, number of epoch of the networks is fixed 500 for calculations.

Table 5.1 shows the perform results of the different structures of artificial neural networks for 1-hour later prediction. The best result is obtained when input number is 2 and hidden layer number is 5.

Table 5.1 : Performance results of ANN for 1-hour later prediction.

input	hidden layer	epoch	perfom1	perfom2	perfom3	perfom4	perfom5	best perform
1	3	500	0.4495	0.4400	0.4431	0.4402	0.4428	0.4400
1	4	500	0.4414	14.7927	0.5663	0.4404	0.4393	0.4393
1	5	500	0.4418	0.4401	0.4411	0.4428	0.4458	0.4401
2	5	500	0.4418	0.4382	0.4438	0.4423	0.4459	0.4382
2	6	500	0.4455	0.4463	0.4477	0.4499	0.4529	0.4455
2	7	500	0.4431	0.4390	0.4447	0.4463	0.4516	0.4390
3	7	500	0.4442	0.4627	0.4459	0.4445	0.4406	0.4406
3	8	500	0.4457	0.4429	0.4531	0.4467	0.4420	0.4420
3	9	500	0.4472	0.4420	0.4457	0.4422	0.4448	0.4420
4	9	500	0.4526	0.4683	0.4509	0.4630	0.4595	0.4509
4	10	500	0.4631	0.4811	0.4535	0.4507	0.4562	0.4507
4	11	500	0.4712	0.4679	0.4718	0.4730	0.4563	0.4563
5	11	500	0.4829	0.4568	0.4521	0.4563	0.4615	0.4521
5	12	500	0.5053	0.4482	0.4709	0.4486	0.4835	0.4482
5	13	500	0.4675	0.4633	0.5036	0.4472	0.4733	0.4472
6	13	500	0.4724	0.4538	0.4709	0.4477	0.4540	0.4477
6	14	500	0.4796	0.4730	0.4523	0.4835	0.5069	0.4523
6	15	500	0.5632	0.5097	0.5059	0.4976	0.4626	0.4626
7	15	500	0.4964	0.4975	0.4741	0.4795	0.4491	0.4491
7	16	500	0.4739	0.4587	0.5872	0.4471	0.4693	0.4471
7	17	500	0.5002	0.4916	0.5371	0.4837	0.4856	0.4837

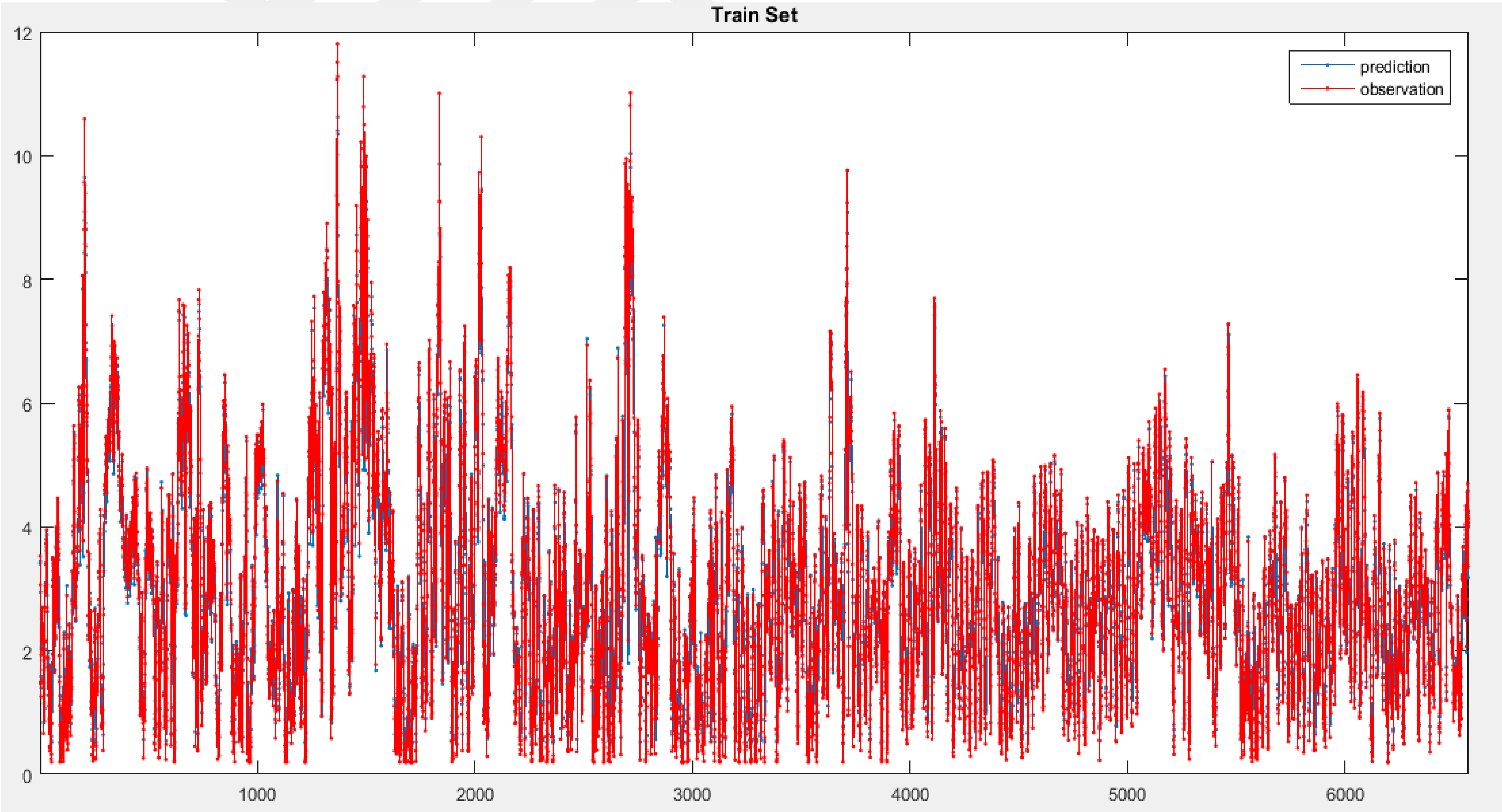

Figure 5.1 : 1-hours later predictions vs observed wind speed data for train set.

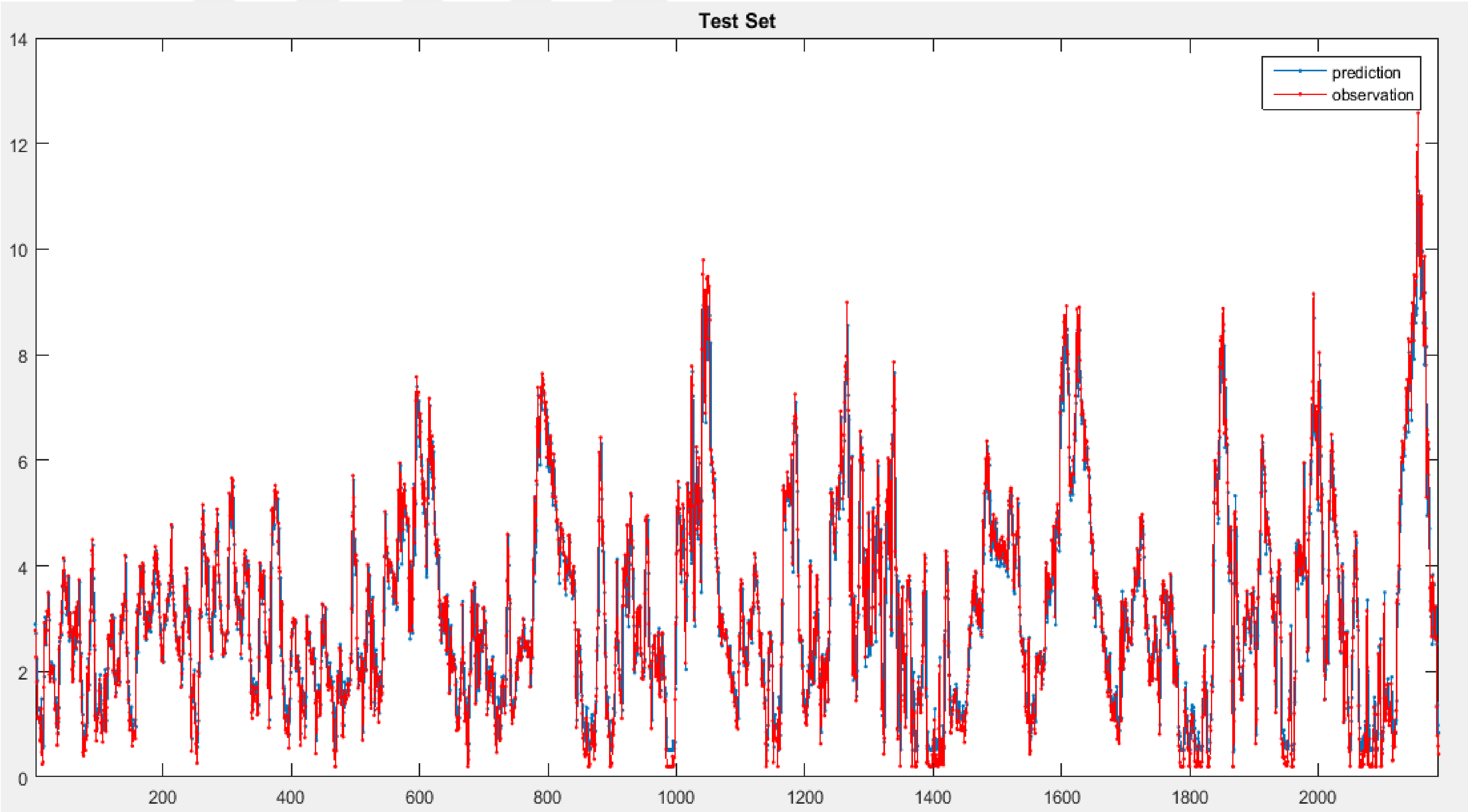

Figure 5.2 : 1-hour later predictions vs observed wind speed data for test set.

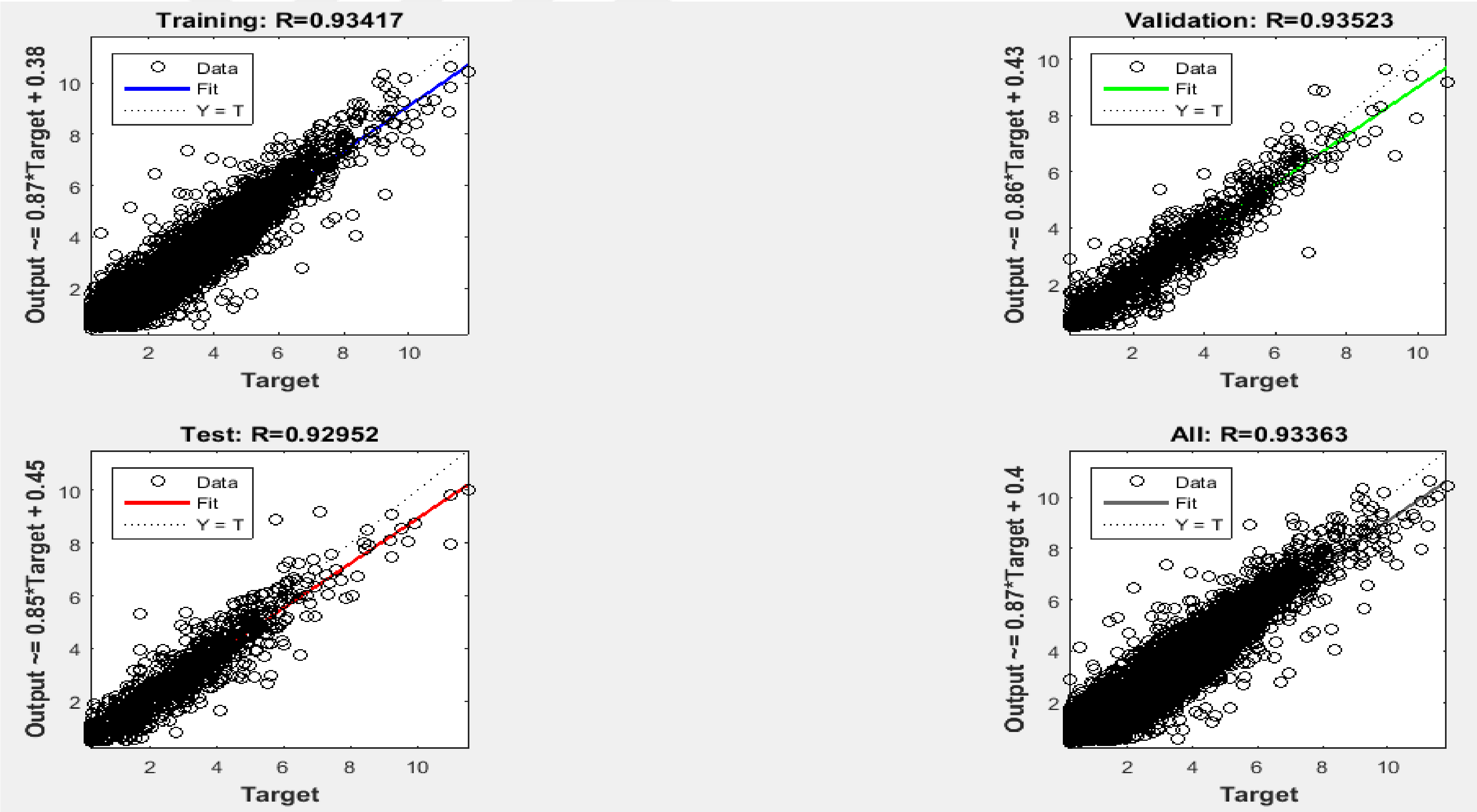

Figure 5.3 : Regression analysis for 1-hour later prediction.

Figures given in the previous pages (see Figure 5.1, Figure 5.2) were obtained by using best ANN on Matlab. There, best means to have the smaller perform value for the network. As mentioned, in order to have the best perform value, number of input and number of hidden layer is adjusted.

It is clear that number of input and number of hidden layer have an impact on ANN. On the other hand, the value of epoch could also affect the result of the network. In the following tables, the effect of number of input, number of hidden layer and number of epoch are tried to be examined. For this purpose, when changing a parameter, the others are fixed. When Table 5.2 shows number of epoch's effect on ANN, Figure 5.4 shows the graph of number of epoch versus performance of ANN. Similarly, Table 5.3 and Figure 5.5 examine hidden layer effect. Furthermore Table 5.4 and Figure 5.6 examine effect of number of input on ANN performance.

Table 5.2 : Number of epoch effect on an ANN.

input	hidden layer	epoch	duration (seconds)	perform
2	5	1000	197.7	0.4410
2	5	700	101.7	0.4451
2	5	400	62.2	0.4396
2	5	100	15.5	0.4389
2	5	50	8.7	0.4401
2	5	20	3.4	0.4429
2	5	10	2.4	0.4413
2	5	5	1.8	0.5358
2	5	2	1.4	0.5502
2	5	1	1.3	0.4724

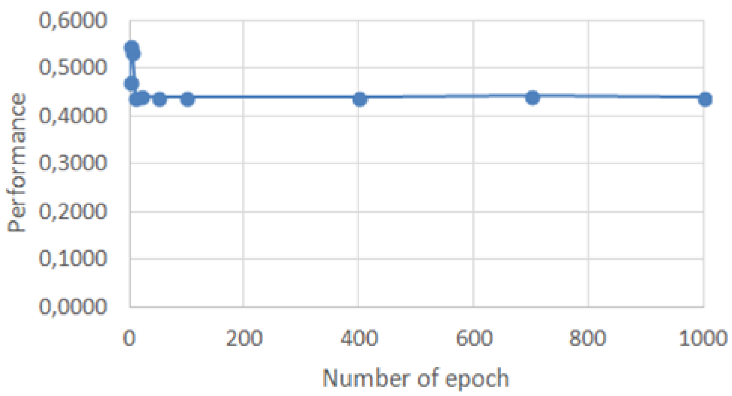

Figure 5.4 : Number of epoch versus performance of ANN.

Table 5.3 : Number of hidden layer effect on an ANN.

input	hidden layer	epoch	duration (seconds)	perform
2	1	100	3.0	0.440
2	2	100	13.7	0.442
2	3	100	13.2	0.443
2	4	100	14.5	0.442
2	5	100	24.7	0.443
2	6	100	16.2	0.448
2	7	100	14.5	0.443
2	8	100	14.2	0.446
2	9	100	14.9	0.445
2	10	100	16.2	0.445

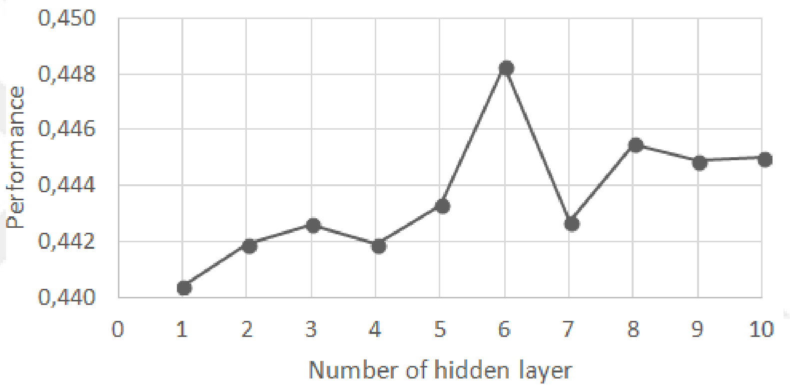

Figure 5.5 : Number of hidden layer versus performance of ANN.

Table 5.4 : Effect of number of input on an ANN.

input	hidden layer	epoch	duration (sec)	perform
1	5	100	12.5	0.4428
2	5	100	12.5	0.4432
3	5	100	12.9	0.4392
4	5	100	13.1	0.4559
5	5	100	13.8	0.4597
6	5	100	13.5	0.4582
7	5	100	17.2	0.4665
8	5	100	18.2	0.5577
9	5	100	18.6	0.4724
10	5	100	16.8	0.4574

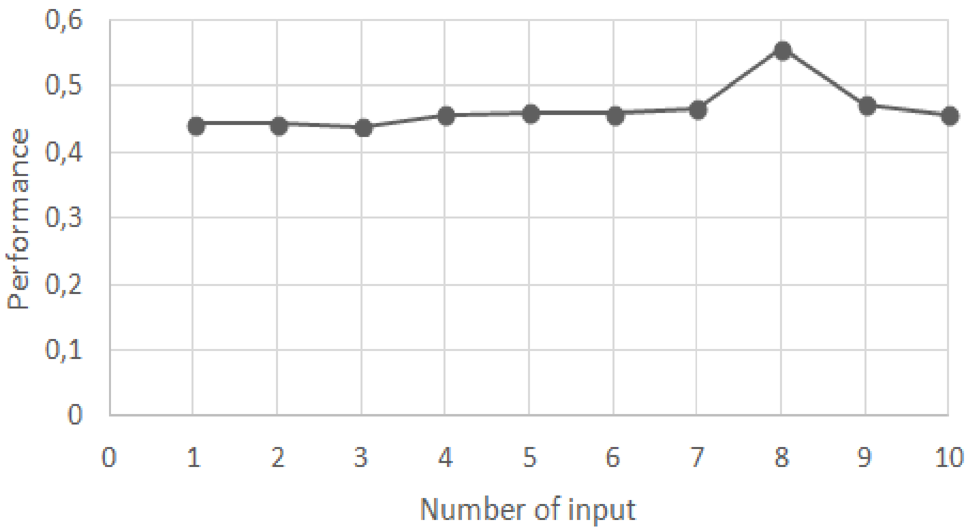

Figure 5.6 : Number of input versus performance of ANN.

5.1.2 WAsP project report for 'ITU_2001_Observed_Data': Scenario-1

(Produced on 24-Apr-16 at 12:57:19 PM by licenced user: csknyldz using WAsP version: 9.00.0153.)

Project parameters:

All of the parameters in the project have default values.

Reference conditions:

The wind atlas contains data for 4 reference roughness lengths (0.000 m, 0.030 m, 0.100 m, 0.400 m) and 5 reference heights (10 m, 25 m, 50 m, 100 m, 200 m) above ground level. The roses of Weibull parameters have 12 sectors each.

In the project, a Vestas V80-2MW wind turbine is used. The Vestas V80-2.0 MW turbine is a pitch-regulated turbine with an 80 meter diameter three bladed rotor. The new turbine is a further development of the well known technology from the V66-1.65 MW.

The speed of revolution of the rotor of the V80-2.0 MW turbine varies between 9 and 19 rpm, and this flexibility is one of the reasons why this turbine is ideal for installation in areas with modest wind speeds. Figure 5.7 shows the power curve of Vestas V80-2MW wind turbine. The wind turbine has an operating tenperature range of -20° to 40° C but there is also more lower temperature wind turbine. The wind turbine have 80 m rotor diameter, 5,027 m^2 swept area and air brake full blade feathering with 3 pitch cylinders

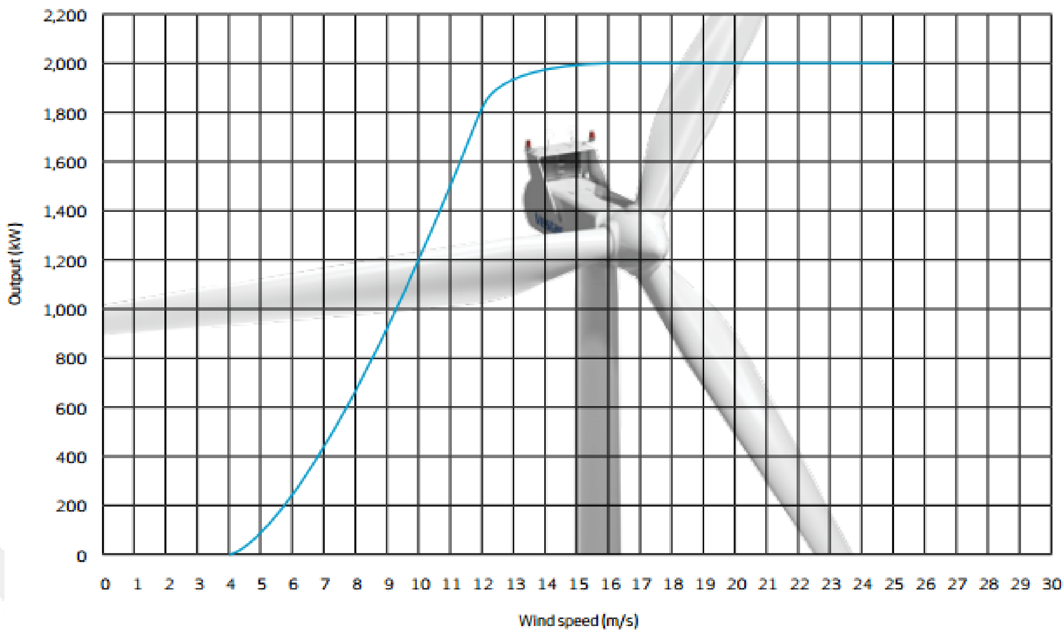

Figure 5.7 : Power curve of V80-2MW (50/60 Hz).

Annual energy production of V80-2MW turbine is given in the Figure 5.8. Note that the graphical result (see Figure 5.8) could only possible when given condition below are occurred.

- One wind turbine, 100% availability
- 0% loses, k factor =2
- Standard air density= 1.225
- Wind speed at hub height

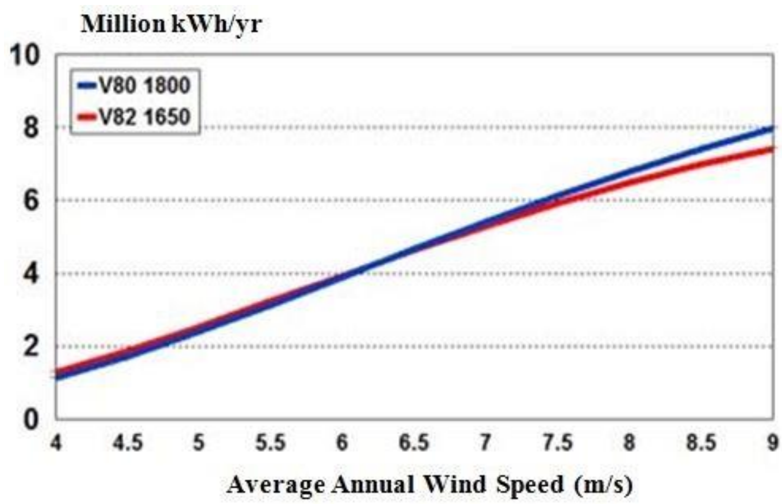

Figure 5.8 : Annual energy production of V80 and V82

Power Regulation:

Pitch regulated with variable speeds

Operating Data:

Rated power 2,000 kW (50Hz)

Cut-in wind speed 4 m/s

Rated wind speed 14 m/s

Cut-out wind speed 25 m/s

Wind class IEC IA

Operating temperature range:

Standard: -20° to 40° C

Low T turbine: -30° to 40° C

Sound Power:

Max105 dB, Mode 0,

10 m above ground,

Hub height 80 m,

Air density 1.225 kg/m3

Rotor:

Rotor diameter 80 m

Swept area 5,027 m^2

Air brake full blade feathering with 3 pitch cylinders

Electrical:

Frequency 50 Hz

Generator type 4-pole (50Hz) doubly fed generator

Slip rings

Blade dimesions:

Lenght 39 m

Maximum chord 3.4 m

Tower:

Tubalar stell tower

Hub height 80 m

Hub Dimensions:

Maximum transport height 3.4 m

Maximum transport width 4 m

Maximum transport lenght 4.2 m

According to the different roughness lenght of 0.00 m, 0.03 m , 0.10 m and 0.40 m Weibull parameter *A*, Weibul parameter *k*, mean speed *U* and power density *E* are shown in Table 5.5. The corresponding heights changes from 10.0 m to 200.0 m.

Table 5.5 : Regional wind climate summary.

Height	Parameter	0.00 m	0.03 m	0.10 m	0.40 m
10.0 m	Weibull A [m/s]	3.76	2.62	2.29	1.81
	Weibull k	1.67	1.48	1.48	1.47
	Mean speed U [m/s]	3.35	2.37	2.07	1.64
	Power density E [W/m²]	54	23	15	8
25.0 m	Weibull A [m/s]	4.12	3.15	2.84	2.40
	Weibull k	1.72	1.57	1.56	1.54
	Mean speed U [m/s]	3.67	2.83	2.55	2.16
	Power density E [W/m²]	69	35	26	16
50.0 m	Weibull A [m/s]	4.43	3.67	3.35	2.91
	Weibull k	1.75	1.71	1.68	1.65
	Mean speed U [m/s]	3.95	3.28	2.99	2.60
	Power density E [W/m²]	83	49	38	26
100.0 m	Weibull A [m/s]	4.80	4.37	4.01	3.54
	Weibull k	1.71	1.79	1.79	1.82
	Mean speed U [m/s]	4.28	3.89	3.57	3.14
	Power density E [W/m²]	109	78	60	40
200.0 m	Weibull A [m/s]	5.28	5.43	4.94	4.31
	Weibull k	1.64	1.73	1.74	1.77
	Mean speed U [m/s]	4.73	4.84	4.40	3.84
	Power density E [W/m²]	156	156	116	76

Figure 5.9 shows the wind rose for roughness length 0.00 m. According to the figure, the sector is seperated 12 different region. Prevailing wind direction is north-north-east (NNE). Figure Figure 5.10, Figure 5.11 and Figure 5.12 show wind rose for roughnes length of 0.03 m, 0.10 m and 0.40 m.

In addition, table Table 5.6, Table 5.7, Table 5.8 and Table 5.9 show sector frequences for these four different roughness lenghts. As mentioned before, sector is divided into 12 equal region by the angle of 30°. It could also be seen from these tables that prevailing wind direction is NNE, because wind frequency for 30° is 24.4% and this is the highest frequency acoording to the other sectors. This table is important in order to set the wind turbine on the related area in order to have efficient energy output. Morever, these sector could also be divided into more region. In this study, sector are divided into twelve different region, which is the default value of WAsP.

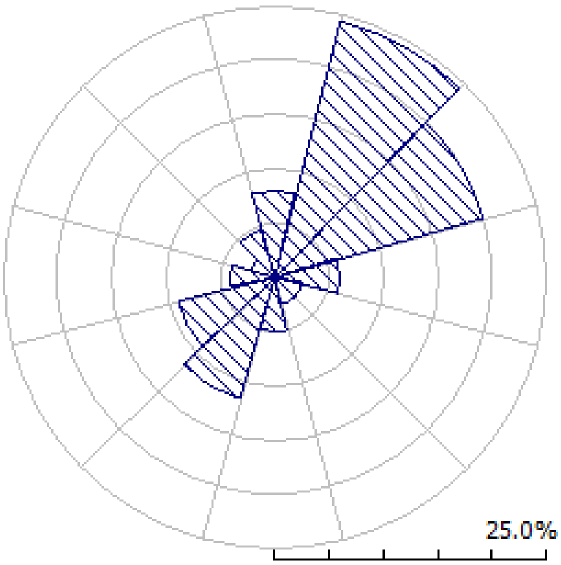

Figure 5.9 : Wind rose for roughness length 0.00 m.

Table 5.6 : Sector frequencies for roughness length 0.00 m.

Sector index	1	2	3	4	5	6	7	8	9	10	11	12
Sector centre angle [°]	0	30	60	90	120	150	180	210	240	270	300	330
Frequency [%]	8.0	24.4	19.8	6.0	2.5	2.6	5.0	11.6	9.1	4.1	2.4	4.5

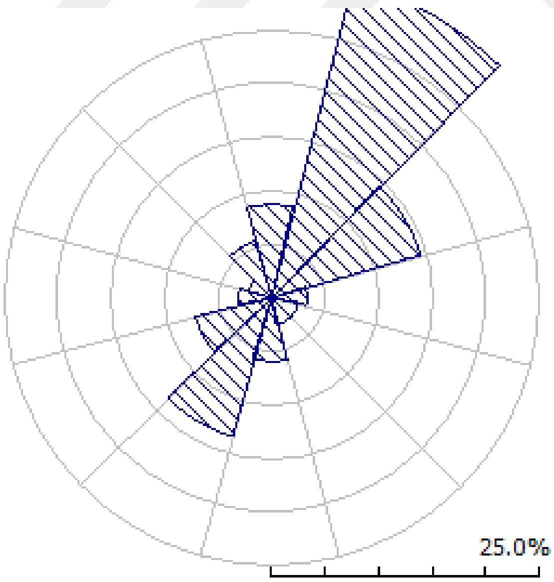

Figure 5.10 : Wind rose for roughness length 0.03 m.

Table 5.7 : Sector frequencies for roughness length 0.03 m.

Sector index	1	2	3	4	5	6	7	8	9	10	11	12
Sector centre angle [°]	0	30	60	90	120	150	180	210	240	270	300	330
Frequency [%]	8.9	30.4	14.5	3.4	2.6	2.5	6.0	13.4	7.3	3.1	2.2	5.6

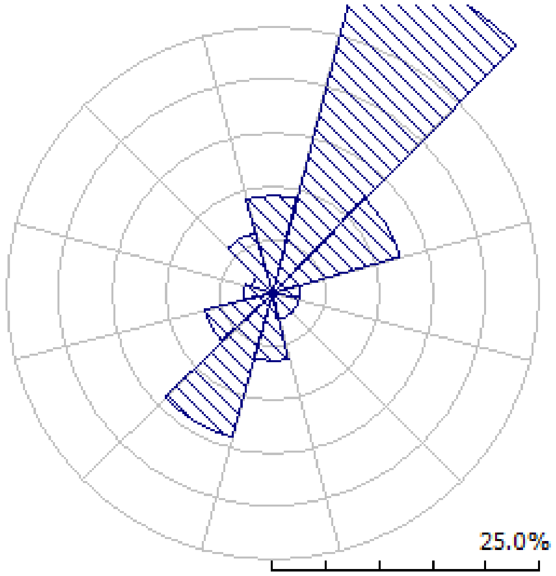

Figure 5.11 : Wind rose for roughness length 0.10 m.

Table 5.8 : Sector frequencies for roughness length 0.10 m.

Sector index	1	2	3	4	5	6	7	8	9	10	11	12
Sector centre angle [°]	0	30	60	90	120	150	180	210	240	270	300	330
Frequency [%]	9.2	32.6	12.5	2.5	2.6	2.5	6.4	14.1	6.6	2.7	2.2	6.0

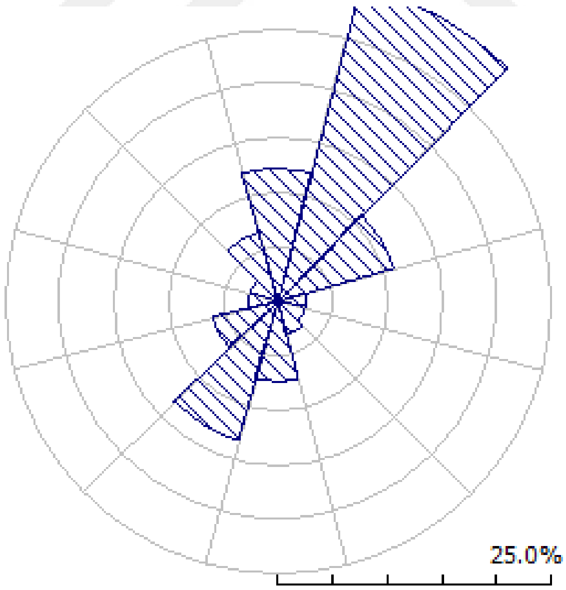

Figure 5.12 : Wind rose for roughness length 0.40 m.

Table 5.9 : Sector frequencies for roughness length 0.40 m.

Sector index	1	2	3	4	5	6	7	8	9	10	11	12
Sector centre angle [°]	0	30	60	90	120	150	180	210	240	270	300	330
Frequency [%]	12.3	30.0	11.0	2.5	2.5	3.1	7.3	13.2	6.1	2.6	2.8	6.5

Given information below shows the setup of resource grid in WAsP. Boundary conditions and nodes are adjusted according to the covering best energy side on the vector map.

Structure: 14 columns and 16 rows at 600 resolution gives 224 calculation sites.

Boundary: (411700, 4547700) to (420100, 4557300)

Nodes: (412000, 4548000) to (419800, 4557000)

Height a.g.l.: 67m

Figure 5.13 shows the mean speed of the vector map on WAsP. Similarly, Figure 5.14 shows the resource grid's mean speed. In the following figures, maximum value, minimum value and mean value are given. Additionally, Figure 5.15 and Figure 5.16 show the power density of vector map and resource grid on WAsP. These figures are important to set the location of the wind turbine on the map in order to have the best wind power output.

Mean Speed [m/s]

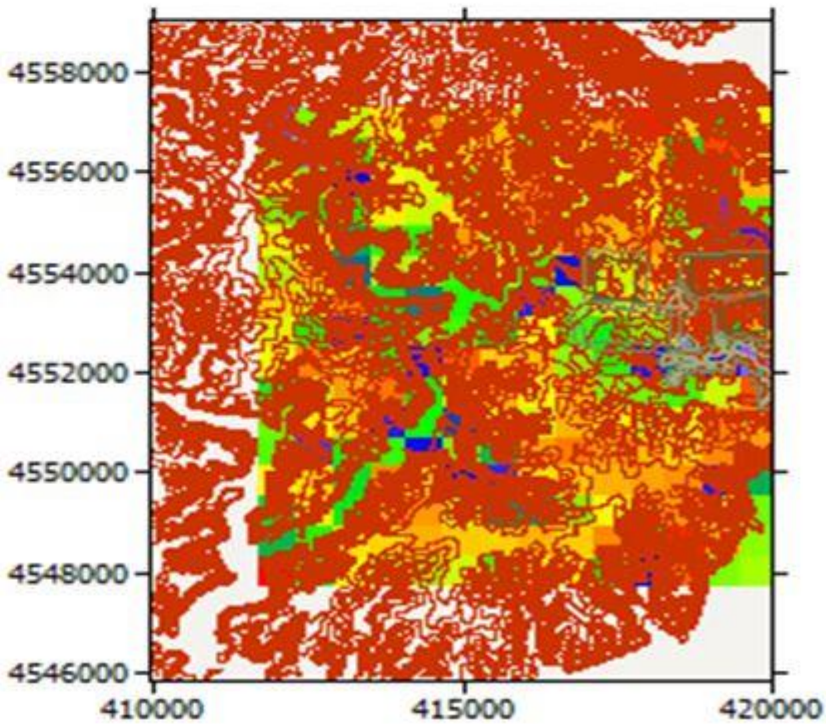

Figure 5.13 : Mean speed of vector map on WAsP.

Maximum Value: 3.78 m/s at (412000.0,4548000.0)

Minimum Valeu: 2.64 m/s at (412600.0,4556400.0)

Mean Value: 3.26 m/s at

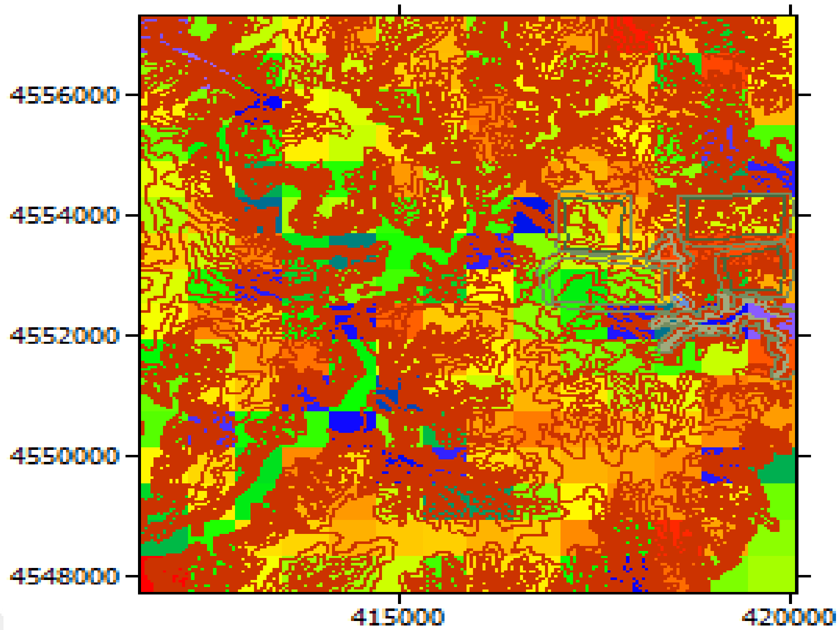

Figure 5.14 : Mean speed of the resource grid on WAsP.

Power Density [W/m²]

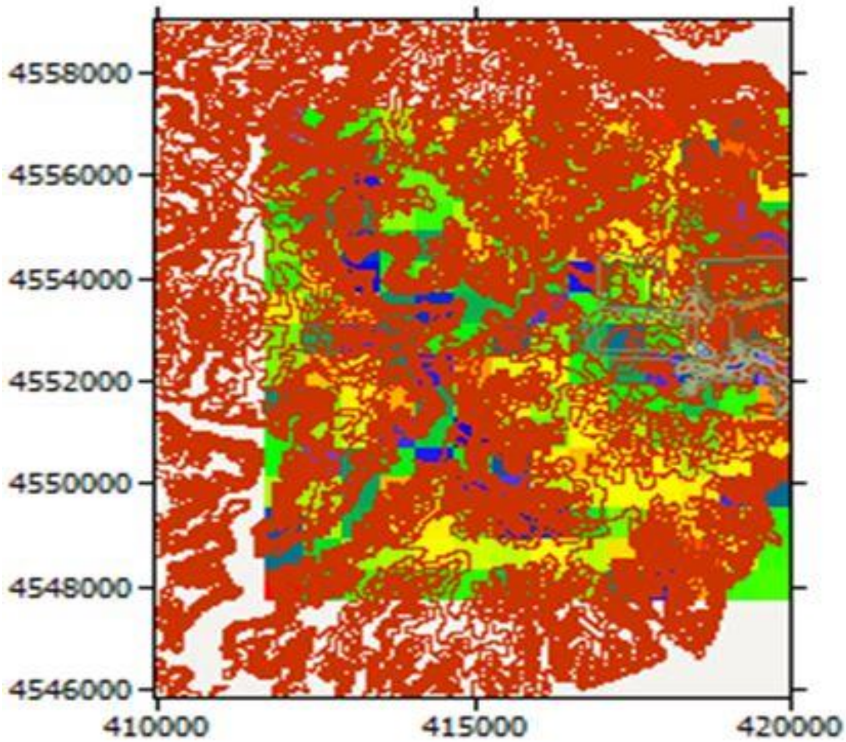

Figure 5.15 : Power density of vector map on WAsP.

Maximum Value: 78 W/m² at (412000.0,4548000.0)

Minimum Value: 23 W/m² at (412600.0,4556400.0)

Mean Value: 48 W/m²

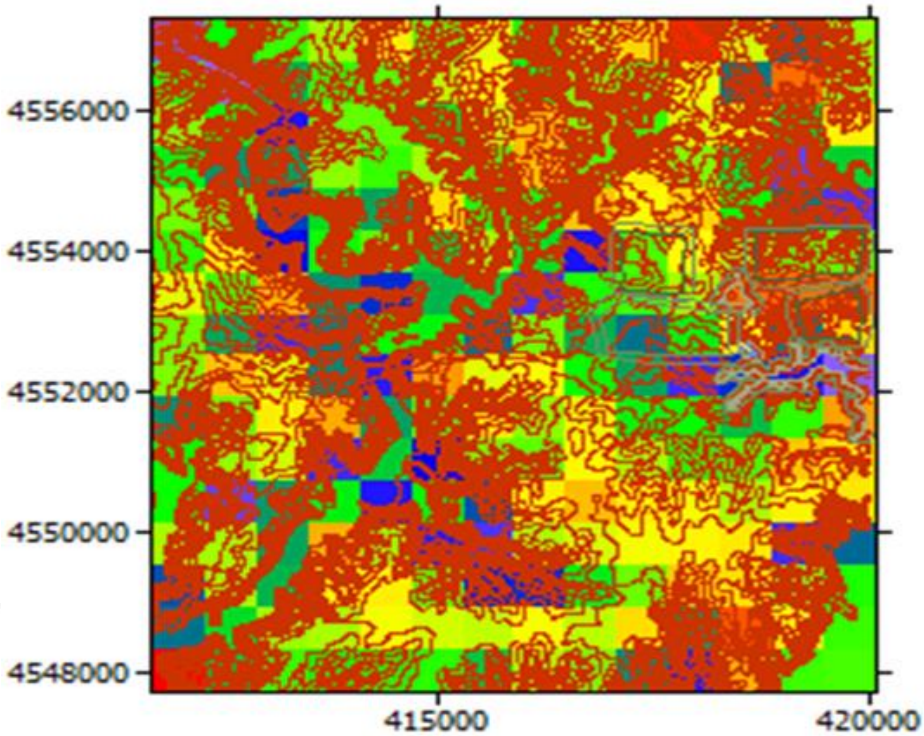

Figure 5.16 : Power density of the resource grid on WAsP.

Nearly same coloured graphs for mean speed and power density are obtained by WAsP for both resourced grid and full of the vector map. Maximum values for mean speed and power speed are seen at the point of (412000.0,4548000.0). Additionally, minimum value for them is obtained at the point of (412600.0,4556400.0).

The use of the ruggedness index (RIX) belongs to the advanced use of WAsP. RIX is an objective measure of the extent of steep slopes in an area. Large RIX differences indicate the size of the wind speed prediction error. In principle, RIX could eb used to make production correction to be applied to the final WAsP prediction – however, the accuracy of such a procedure would completely be designer responsibility.

The RIX value is first of all a tool by which it can be judged whether WAsP is working within or outside its performance envelope, whether prediction errors may be expected, and the sign and approximate magnitude of such errors. Figure 5.17 and Figure 5.18 shows RIX [%] and below the figures detailed information about RIX [%] values are given. All of the figures which are shown in taken from the WAsP software are important in order to make decision about wind turbines' location in the related area. Thanks to the usage of the whole information from the software, the best energy output result could be obtained. Therefore, one of the main important steps to apply this software is to know the meaning of the figures.

RIX [%]

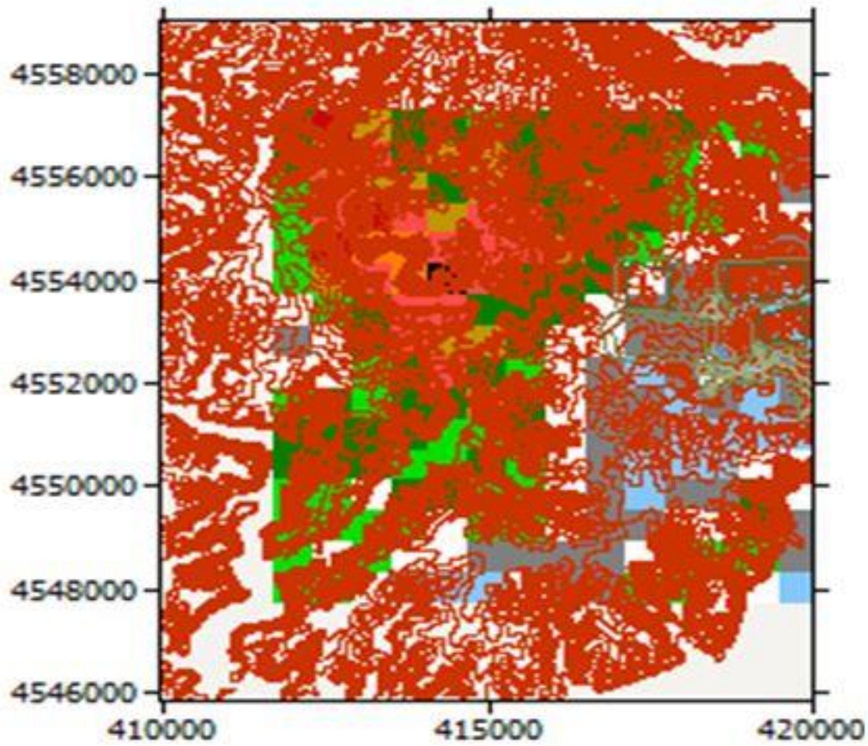

Figure 5.17 : RIX [%] of vector map on WAsP.

Maximum Value: 7.7% at (414400.0,4554000.0)

Minimum Value: 1.5% at (419800.0,4553400.0)

Mean Value: 3.8%

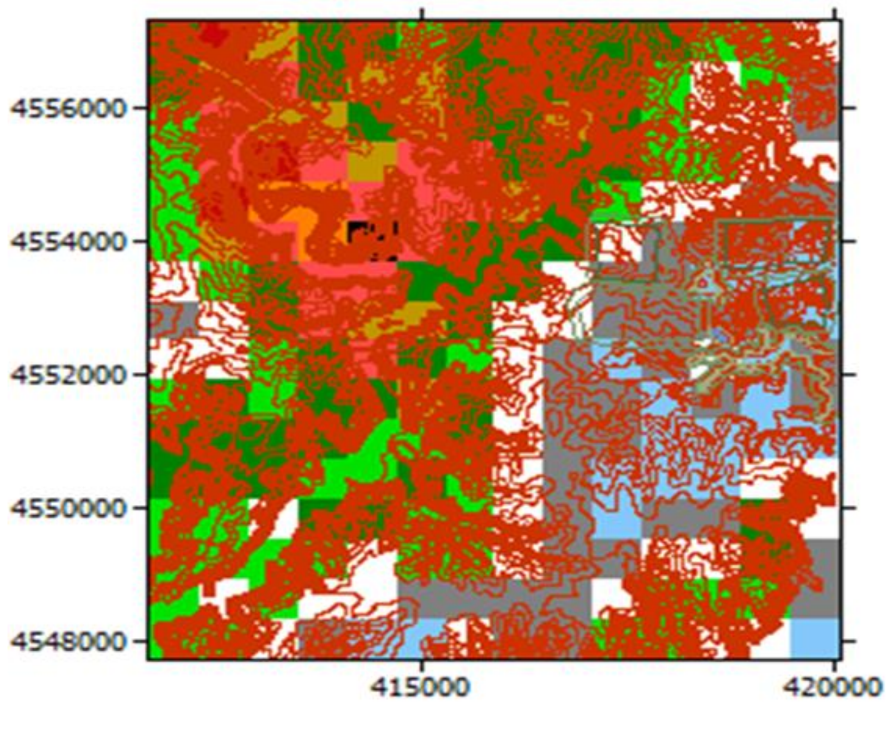

Figure 5.18 : RIX [%] of the resource grid on WAsP.

Wind farm: 'Turbine cluster 1'

The wind farm lies in a map called ITU_VECTOR_MAP. Figure 5.19 shows turbine cluster on İTU vector map in WAsP.

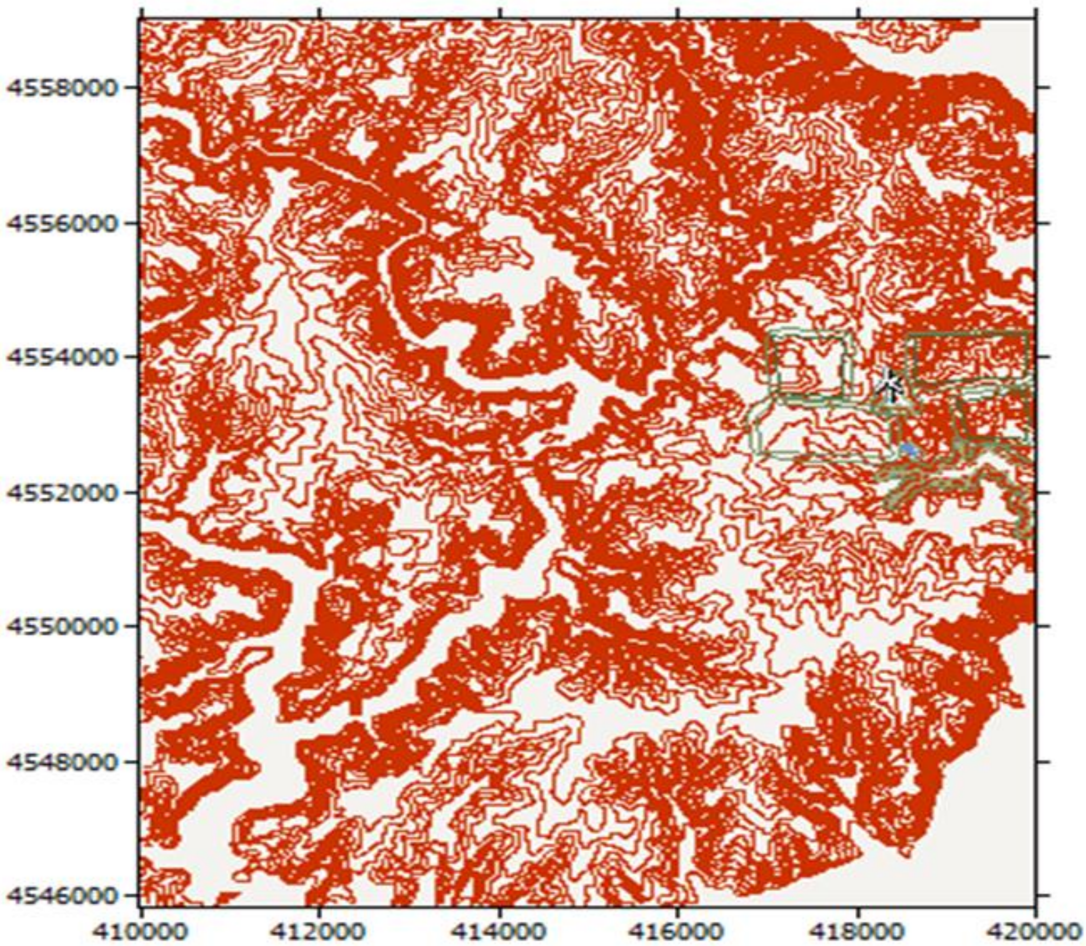

Figure 5.19 : Turbine cluster on ITU_VECTOR_MAP.

After setting the location of V-80 wind turbine on the vector map by considering the power density figure, total wind power output is calculated. Because only one wind turbine is used on the map, average value, minimum value and maximum value for turbine cluster is same. Table 5.10, Table 5.11 and Table 5.12 give informations about turbine cluster, site results and site wind climates.

Table 5.10 : Summary result of turbine cluster for observed wind data.

Parameter	Total	Average	Minimum	Maximum
Net AEP [GWh]	1.368	1.368	1.368	1.368
Gross AEP [GWh]	1.368	1.368	1.368	1.368
Wake loss [%]	0.0	-	-	-

Table 5.11 : Site results for observed wind data.

Site	Location [m]	Turbine	Elevation [m] a.s.l.	Height [m] a.g.l.	Net AEP [GWh]	Wake loss [%]
Turbine site 001	(418388.6,4553379.0)	Vestas V80 (2.0 MW)	140	67	1.368	0.0

Table 5.12 : Site wind climates for observed wind data.

Site	Location [m]	H [m]	A [m/s]	k	U [m/s]	E [W/m²]	RIX [%]
Turbine site 001	(418388.6,4553379.0)	67	4.4	1.74	3.96	84	2.2

5.1.3 WAsP Project Report for 'ITU_2001_ANN Sim._Data': Scenario-2

All of the parameters for *Scenario-2* are same with *Scenario-1* except wind speed data.

Mean Speed [m/s]

Maximum Value: 3.73 m/s at (412000.0,4548000.0)

Minimum Value: 2.61 m/s at (412600.0,4556400.0)

Mean Value: 3.22 m/s

Power Density [W/m²]

Maximum Value: 71 W/m² at (412000.0,4548000.0)

Minimum Value: 22 W/m² at (412600.0,4556400.0)

Mean Value: 44 W/m²

RIX [%]

Maximum Value: 7.7% at (414400.0,4554000.0)

Minimum Value: 1.5% at (419800.0,4553400.0)

Mean Value: 3.8%

Table 5.13 : Summary results of turbine cluster for simulated wind data.

Parameter	Total	Average	Minimum	Maximum
Net AEP [GWh]	1.257	1.257	1.257	1.257
Gross AEP [GWh]	1.257	1.257	1.257	1.257
Wake loss [%]	0.0	-	-	-

Table 5.14 : Site results for simulated wind data.

Site	Location [m]	Turbine	Elevation [m] a.s.l.	Height [m] a.g.l.	Net AEP [GWh]	Wake loss [%]
Turbine site 001	(418388.6,4553379.0)	Vestas V80 (2.0 MW)	140	67	1.257	0.0

Table 5.15 : Site wind climates for simulated wind data.

Site	Location [m]	H [m]	A [m/s]	k	U [m/s]	E [W/m²]	RIX [%]
Turbine site 001	(418388.6,4553379.0)	67	4.4	1.81	3.90	77	2.2

5.1.4 WAsP Project Report for 'ITU_2001_ANN Sim._Data': Scenario-3

(Produced on 29-Apr-16 at 8:54:48 AM by licenced user: csknyldz using WAsP version: 9.00.0153.)

All of the parameters for Scenario-3 are same with Scenario-2 except wind speed direction data. The purpose of Scenario-3 is to see the effect of wind direction on energy production on WAsP software. Therefore, wind direction column used in scenario-2 is changed by wind direction column which was observed in Biga (Canakkale) in 2009. After all, given results below are obtained.Table 5.16, Table 5.17 and Table 5.18 show different results for simulated wind data and direction column of Biga_2009.

Mean speed [m/s]:

Maximum Value: 3.53 m/s at (418600.0,4548600.0)

Minimum Value: 2.70 m/s at (419200.0,4554600.0)

Mean Value: 3.09 m/s

Power density:

Maximum Value: 59 W/m² at (418600.0,4548600.0)

Minimum Value: 26 W/m² at (414400.0,4550400.0)

Mean Value: 40 W/m²

RIX [%]:

Maximum Value: 7.7% at (414400.0,4554000.0)

Minimum Value: 1.5% at (419800.0,4553400.0)

Mean Value: 3.8%

Table 5.16 : Summary result of turbine cluster.

Parameter	Total	Average	Minimum	Maximum
Net AEP [GWh]	1.109	1.109	1.109	1.109
Gross AEP [GWh]	1.109	1.109	1.109	1.109
Wake loss [%]	0.0	-	-	-

Table 5.17 : Site results of the study.

Site	Location [m]	Turbine	Elevation [m] a.s.l.	Height [m] a.g.l.	Net AEP [GWh]	Wake loss [%]
Turbine site 001	(418388.6,4553379.0)	Vestas V80 (2.0 MW)	140	67	1.109	0.0

Table 5.18 : Site wind climates.

Site	Location [m]	H [m]	A [m/s]	k	U [m/s]	E [W/m²]	RIX [%]
Turbine site 001	(418388.6,4553379.0)	67	4.2	1.81	3.75	69	2.2

5.2 Future Prediction by ANN

In this part of the study, ANN will be used to make a future prediction. Performance analysis of ANN is applied by using ITU Maslak 2001 data only. On the other hand, to make a future prediction, ITU Maslak the year of 2001 data are used to make wind speed prediction for ITU Maslak 2002.

Table 5.19 shows the performance of the ANN. As previous studies, number of input values n is set from 1 to 7 and three different number of hidden layers h are used for each number of input starting with $n = 2xh+1$. The best result of the ANN is obtained when number of input $n=5$ and number of hidden layer is $h=11$. (see Table 5.19).

Figure 5.20 gives a comparison between observed and predicted wind speed data for train set. Predicted wind speed data are obtained by using the previous wind speed data of the related area (ITU Maslak 2001 observed wind speed data). Additionally, Figure 5.21 shows the comparison of predictied and observed wind speed data for test set. Because, train set include three times more data from test set, the figures are different from each other by the mean of time steps' number.

Table 5.19 : Performance of ANN for future prediction from 2001 to 2002 data.

input	hidden layer	epoch	perform1	perfom2	perfom3	perfom4	perfom5	best perform
1	3	500	2.9809	2.9792	2.9780	2.9821	2.9922	2.9780
1	4	500	2.9916	2.9809	2.9938	2.9750	2.9945	2.9750
1	5	500	3.0006	2.9998	2.9894	3.0088	3.0050	2.9894
2	5	500	2.9752	2.9804	2.9972	2.9892	2.9757	2.9752
2	6	500	2.9905	2.9662	2.9840	2.9804	2.9726	2.9662
2	7	500	2.9747	2.9782	2.9765	3.0065	3.0323	2.9747
3	7	500	2.9632	2.9980	2.9603	2.9713	3.0018	2.9603
3	8	500	3.0008	2.9954	2.9698	2.9725	2.9879	2.9698
3	9	500	2.9838	2.9837	2.9700	2.9812	2.9795	2.9700
4	9	500	2.9736	3.0253	2.9635	2.9729	3.0146	2.9635
4	10	500	2.9688	2.9632	2.9688	2.9876	3.0119	2.9632
4	11	500	2.9832	2.9676	2.9923	2.9733	2.9974	2.9676
5	11	500	3.0935	3.0309	2.9495	2.9616	2.9735	2.9495
5	12	500	2.9946	3.0185	3.0036	2.9708	3.1390	2.9708
5	13	500	2.9667	2.9855	2.9957	3.1634	2.9604	2.9604
6	13	500	2.9930	3.1263	3.0095	2.9611	2.9547	2.9547
6	14	500	2.9657	3.0480	3.1957	2.9976	3.0241	2.9657
6	15	500	2.9767	2.9852	2.9519	3.0217	3.0071	2.9519
7	15	500	2.9699	2.9827	2.9627	2.9633	3.0715	2.9627
7	16	500	2.9505	2.9555	2.9782	2.9948	2.9985	2.9505
7	17	500	3.1525	2.9711	2.9910	2.9848	3.0042	2.9711

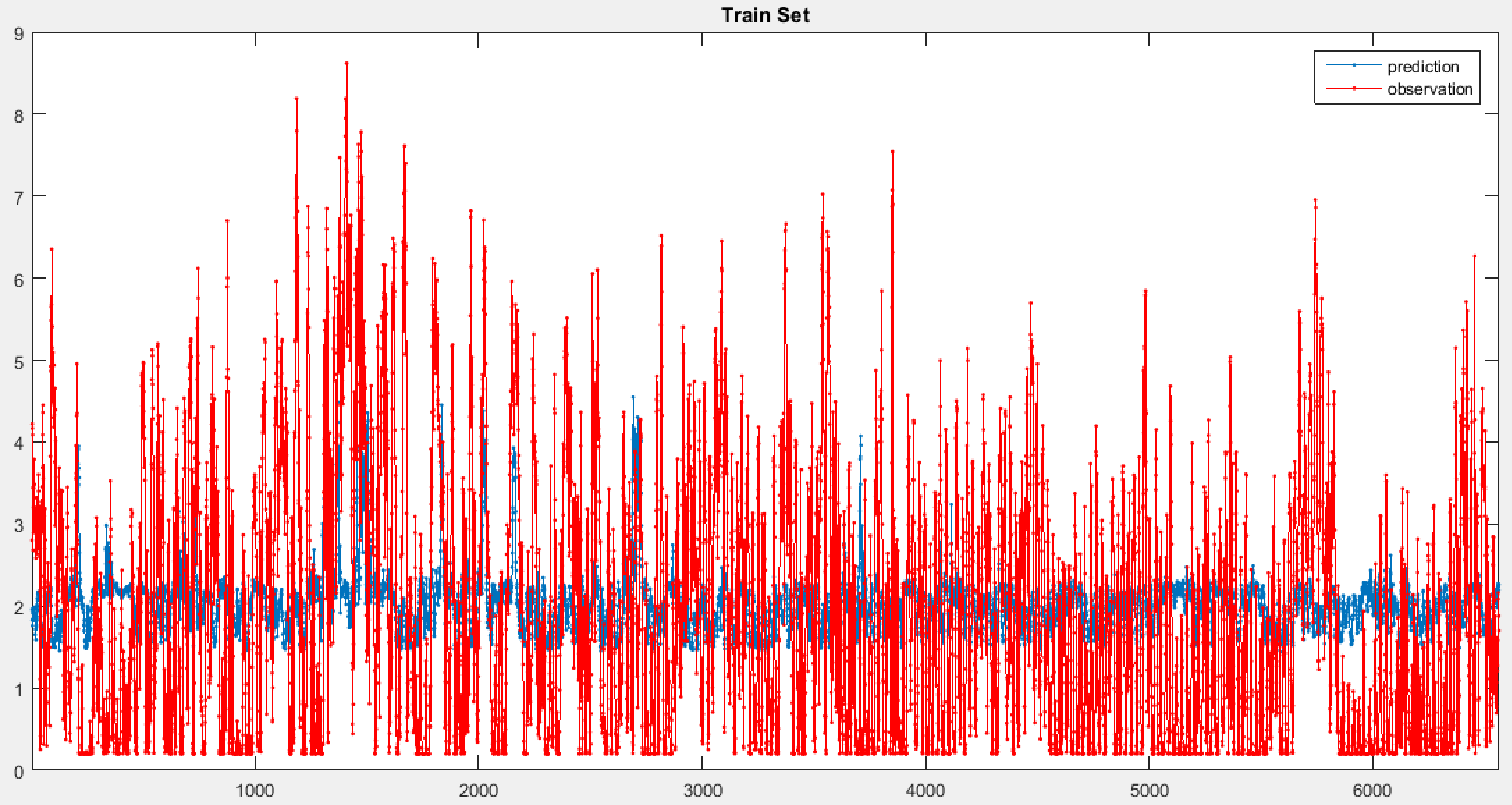

Figure 5.20 : Observed vs trained data for train set of future prediction.

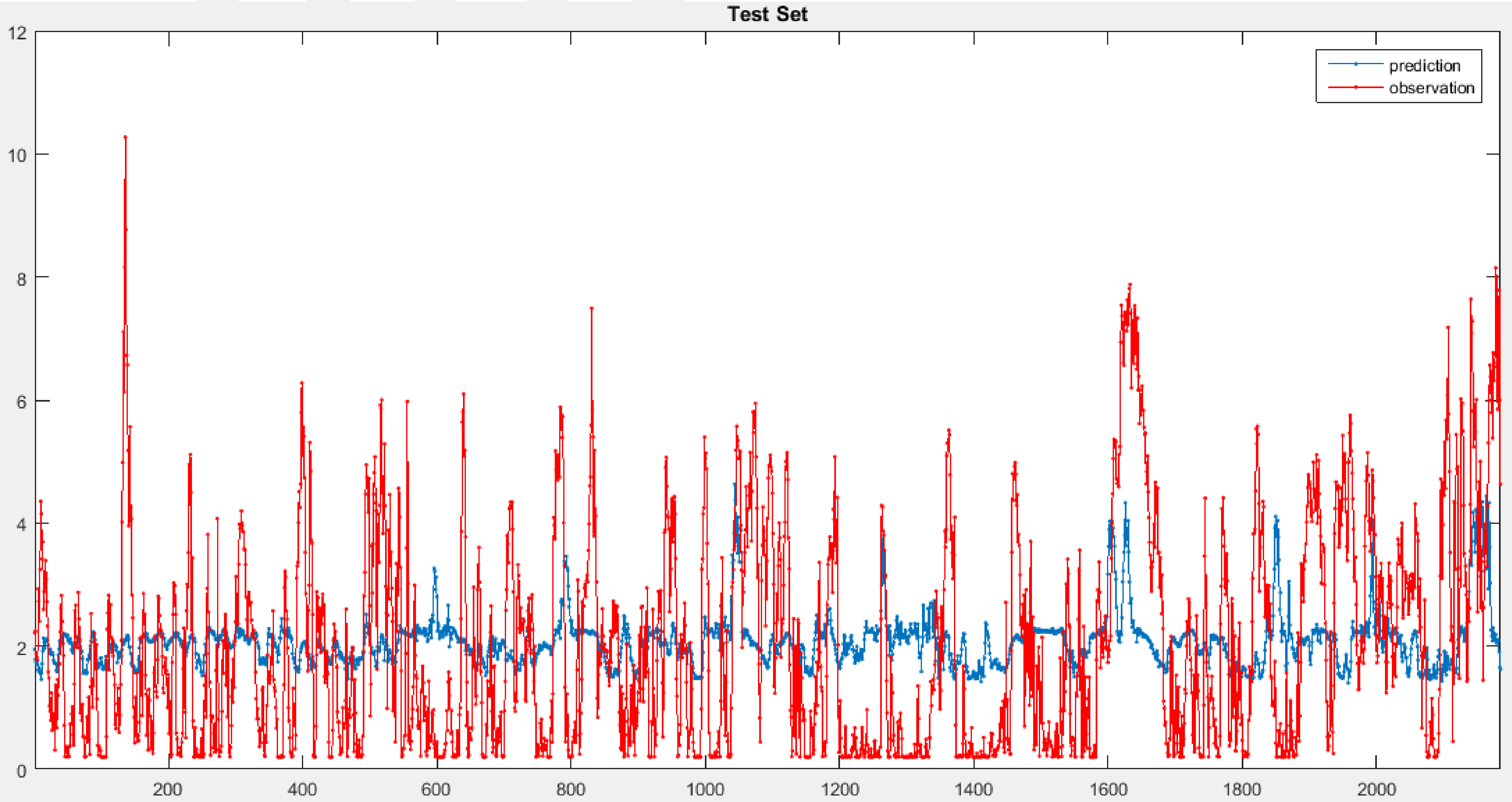

Figure 5.21 : Observed vs trained data for test set of future prediction.

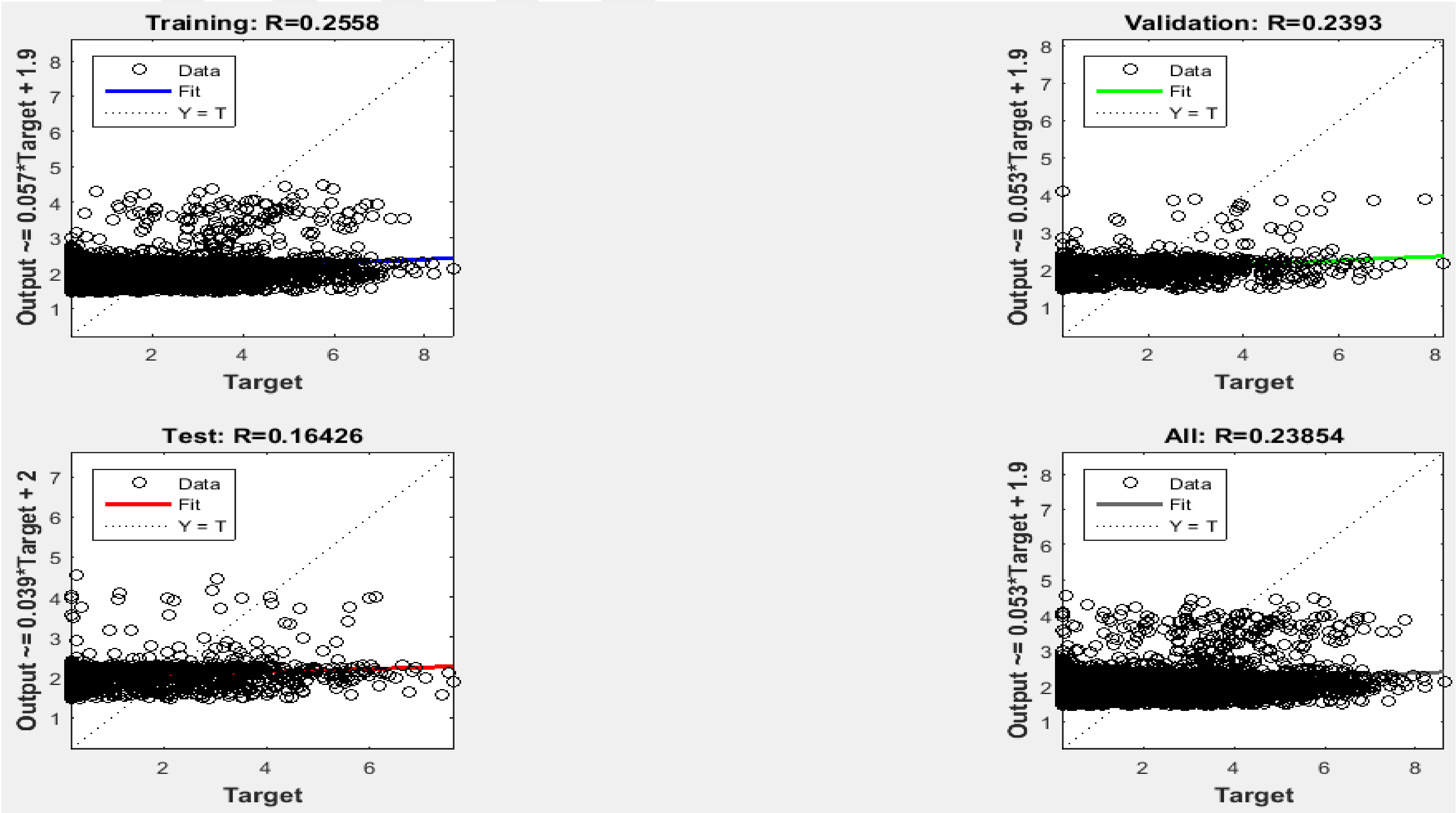

Figure 5.22 : Regression analysis of best network for future prediction.

6. RESULT AND DISCUSSION

6.1 Results for Performance analysis

One of the purposes of this study was to analize the performance of artificial neural networks.

The most important criteria during this study are given below:

- Number of input,
- Number of hidden layer.

Table 6.1 : Performance results of different scenarios.

scenario	input	hidden layer	epoch	best perform
1-hour later	2	5	500	0.4382
6-hours later	1	3	500	2.1501
12-hours later	1	3	500	3.1513
24-hours later	6	13	500	3.7032

Table 6.1 gives the performance results of different scenarios. For all calculations, ITU 2001 observed wind speed data are used. Additionally, starting from 1 to 7, totally 7 different number of inputs n were used. For each number of inputs, 3 different hidden layers were used.

According to the table Table 6.1 for each scenario, best performance results are obtained by starting the number of hidden layer according to the equation **(8.1)**.

$$h = 2 \times n + 1 \qquad \textbf{(8.1)}$$

Where:

h= number of hidden layer,

n= number of input.

On the other hand, for different scenarios, number of input varies to find best performance ANN. For example, for 1-hour later prediction, best performance result were obtained when number of input is 2 but for 24-hours later prediction the best result is obtained when number of input is 6.

Another important output coming from the Table 6.1 is that increasing the number of hours results a decrease on ANN performance. Therefore, best result was obtained for 1-hour later prediction. Figure 6.1 shows the performance changes according to the change of number of time steps.

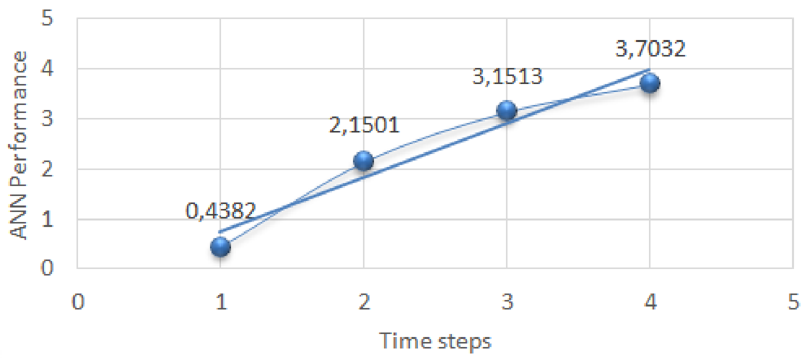

Figure 6.1: Comparison of results of different time steps.

6.2 Results for wind energy calculation

In the second step of the study, in order to make one more analysis for ANN, wind energy calculation is made by using WAsP. In the previous section detailed information about the wind energy prediction's outputs for three different scenarios are given. These three different scenarios are given below.

1stScenario:

- Observed wind speed data for ITU Maslak (in 2001) are used.
- Observed wind direction data for ITU Maslak (in 2001) are used.

2ndScenario:

- 1-hour later predicted wind speed data for ITU Maslak (in 2001) are used.
- Observed wind direction data for ITU Maslak (in 2001) are used.

3rdScenario:

- Same predicted wind speed data with scenario-2 are used.
- Observed wind direction data for Biga (Canakkale) in 2009 are used.

While the purpose of first and second scenarios are to make a comparision on wind energy prediction between observed and predicted wind speed data, the purpose of applying third scenario is to study the effect of wind direction on wind energy prediction.

Table 6.2 : Total wind energy productions through different scenarios.

Parameter		Total	Average	Minimum	Maximum
Net AEP [GWh]	scenario 1	1.368	1.368	1.368	1.368
	scenario 2	1.257	1.257	1.257	1.257
	scenario 3	1.109	1.109	1.109	1.109

According to the table Table 6.2, total wind energy production changes through different scenarios. It is clear that wind direction has an effect on wind energy calculation by WAsP. However, 1.257 GWh power is calculated for scenario-2, WAsP calculates 1.109 GWh power for scenario-3.

6.3 Results for Future Time Prediction

When making prediction for 1-hour later for the year of 2001, best performance result is obtained when number of input is 2 and number of hidden layer is 5 and which is 0.4382. If the performance value is small, then a closed result to the observed wind speed data is obtained. Therefore, much closed result is obtained for 1-hour later prediction (see Figure 5.1). On the other hand, the best performed value for feature time prediction is 2.9495 which is given in the Table 5.19. This value is very bigger than the performance value for 1-hour later prediction. Therefore, wind speed prediction data do not fit the observed wind speed data for the year of 2002 (see Figure 5.20).

That unclosed results could also seen in the Figure 5.21 which shows the comparison of the observed and predicted wind speed results for test set of the network. Results for future time prediction show a similarity with the results of 12-hour and 24-hour later prediction by means of their closeness to the observed wind speed data.

Table 6.3 : Regresion analysis results for different scenarios.

Prediction	Regression Analysis Result *R*
1-hour later prediction	0,93363
6-hour later prediction	0,56839
12-hour later prediction	0,37499
24-hour later prediction	0,4727
future prediction (2001 to 2002)	0,23854

According to Table 6.3, the best regression result is obtained for 1-hour later wind speed prediction section. On the other hand, the worst result occurs when trying to make a future time prediction with the ANN.

Prediction wind speed by using ANN is made for different time steps. There are some disadvantages of making predictions for 1-hour later, 6-hour later and so on. One of the disadvantages is getting less data from original data. For example, twelve-wind speed values are used in order to predict next value in 12-hour later prediction and these values are not predicted by ANN. This causes having less predicted wind speed data than observed wind speed data. This problem could be solved by developing some different ANN models.

Moreover, in order to have closed predicted result to observed data, number of observed data is an important parameter. Nevertheless, the accuracy of the observation is also a necessity to have acceptable outputs from prediction. To sum up, both number of observation and the accuracy of the observation are main necessities to make a good prediction.

7. CONCLUSION

In conclusion, performance analysis of ANN, wind speed prediction and wind energy prediction are made in this study. Artificial neural networks are used to make a prediction for wind speed by appliying ANN codes on Matlab. Additionally, WAsP software is used to calculate wind energy.

In order to make wind speed prediction, different scenarios are applied. Firstly, wind speed data, which are observed in ITU Maslak in 2001, are used in the ANN to make 1-hour later prediction. Furthermore, 6-hours later, 12-hours later and 24-hours later predictions are applied. The predicted results are also belong to same year of 2001. Because, different input numbers are used in these four different scenarios, predicted wind speed data are also differ from each other. The best result is obtained when making a prediction for 1-hour later.

Secondly, 2002 wind speed data of ITU Maslak are predicted by the using of 2001 wind speed data of the related area on Matlab ANN code. Predicted wind speed results are not closed to the observed result for the year of 2002.

In addition, some calculations for wind energy prediction are also made for three different scenarios. For the first scenario, observed wind speed and direction data for ITU Maslak in 2001 are used and a calculation is made to find total amount of wind energy by WAsP. The only different parameter of second scenario from first one is using 1-hour later predicted wind speed data. On the other hand, the only difference of third scenario from the second one is the direction data. The purpose of creating first scenario and second scenario is to make a comparision for wind energy prediction but making up the third one is to see the effect of direction data on wind energy calculation.

Taking everything into consideration, artificial neural networks could be a good alternative for wind speed and energy prediction. In order to have acceptable results, network architecture must be generated by taking account of number of input, number of hidden layer and type of prediction.

REFERENCES

Beck, A. (2014). Chapter 2. Optimality Conditions for Unconstrained Optimization. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB* (pp. 13-36).

Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Economics 31*, 307-327.

Box, G., M. Jenkins, G., & Reinsel, G. (2008). *Time Series Analysis: Forecasting and Control (Fourth Edition).* New Jersey: John Wiley&Sons.

Burton, T., Jenkins, N., Sharpe, D., & Bossanyi, E. (2011). *Wind Energy Handbook (Second Edition).* Chichester: Wiley.

Büyükyıldız, D. (2012). *Wind powered pumped hydro storage systems and Aslantas case study* .(Master of science thesis).Istanbul Technical University, Istanbul.

Castellani, F., Burlando, M., Taghizadeh, S., Astolfi, D., & Piccioni, E. (2013). Wind energy forecast in complex sites with a hybrid neural network and CFD based method. *Energy Procedia 45*, 188-197.

Chatfield, C. (1995). *The Analysis of Time Series: An Introduction (Fifth Edition).* Bath: CHAPMAN&HALL/CRC.

Electricity Marketting Law, 4628 (Energy Marketting Regulatory Authority March 3, 2001).

Fan, J., & Yao, Q. (2003). *Nonlinear Time Series.* United States: Springer series in statistics.

Freeman, J., & Skapura, D. (1991). *Neural Networks: Algorithm, Applications and Programing Techniques.* Houston: Addison-Wesley Publishing Company.

Jung, J., & Broadwater, R. (2014). Current Status and Future Advances for Wind Speed and Power Forecasting. *Renewable and Sustanable Energy Reviews*.

K. Sreelakshmi, P. K. (2008). Performance Evaluation of Short Term Wind Speed Prediction Techniques . *International Journal of Computer Science and Network Security* , 162-169.

Khatib, A. (2011). *Developing a wind speed prediction tool using neural networks and designing wind park in Syria using WASP software.*(Master of science thesis).Cairo University,Kassel.

Kumar, M. (2009). *Short-term load forecasting using artificial neural networks techniques*.(Bachelor of science thesis).National University of Technology, Rourkela.

Egghe, L., & Leydesdorf, L. (n.d.). The relation between Pearson's correlation coefficient r and Salton's cosine measure . *Journal of American Society for Information Science&Technology (forthcoming)*.

LadyofHats. (2007). In Wikipedia. adress: https://en.wikipedia.org/wiki/File:Complete_neuron_cell_diagram_en.

Lange, B., Wessel, A., Dobschinski, J., & Rohrig, K. (2009). Role of Wind Power Forecasting in Grid Integration. *Kasseler SysposiumEnergie-Systetechnik*.

Li, G., & Shi, J. (2009). On Comparing Three Artificial Neural Networks for Wind Speed Forecasting. *Applied Energy*.

Li, L., Wang, M., Zhu, F., & Wang, C. (2009). Wind Power Forecasting Based on Time Series and Neural Networks. *Proceedings of the Second Sysposium International Computer Science and Computationai Technology* (pp. 293-297). Huangshan: Academy Publisher.

M. Jafarin, A. R. (2010). Fuzzy Modelling Techniques and Artificial Neural Networks to Estimate Annual Energy Output of A Wind Turbine. *Renewable Energy* .

Martinez-Alvarez, F., Troncoso, A., Asencio-Cortes, G., & Riquelme, J. (2015). A Survey on Data Mining Techniques Applied to Electricity-Related Time Series Forecasting. *Energies-MPDI*, 13162-13193.

Matlab-R2015a. (2015). Levenberg-Marquardt backpropagation. *Neural Network Toolbox: Function Approximation and Nonlinear Regression*. 1994-2015 The MathWorks, Inc.

Rabunal, J., & Dorado, J. (2005). *Artificial Neural Networks in Real Life Applicaions*. United States: Idea Group Publishing.

Ramasamy, P., Chandel, S., & Yadav, A. K. (2015). Wind Speed Prediction in the Mountainous Region of India Using an Artificial Neural Network Model. *Renewable Energy*.

Sanderse, B. (2016). *Aerodynamics of wind turbine wakes.* Netherland: ECN Energy Search Centre of the Netherlands.

Sathya, R., & Abraham, A. (2013). Comparison of Supervised and Unsupervised Learning Algorithm for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligience*, Vol.2, No.2.

Schutz, A. L. (2008). Hyperbolic Fucntion (Expository Paper) . *Master of Arts in Teaching with a Specialization in the Teaching of Middle Level Mathematics*.

T. Chai, R. D. (2014). Root mean square error (RMSE) or mean absolute error (MAE)?-Arguments against avoiding RMSE in the literature. *Geoscientific Model Development* , 1247-1250.

Tong, H. (1990). *Nonlinear Time Series: A Dynamical System Approach.* Oxford: Oxford University Press.

W. Karlen, H. M. (2014). Improving the Accuracy and Efficiency of Respiratory Rate Measurements in Children Using Mobile Devices. *PLoS ONE, Volume 9* .

Yadav, N. (2015). History of Neural Networks. N. Yadav içinde, *An Introduction to the Neural Networks Methods for Differential Equations* (s. 13-15). SpringerBriefs in Computational Intelligence.

Zhang, B. T. (2001). Supplement to 2001 Bioinformatics Lecture on Neural Nets. *Artificial Neural Networks*. Seoul National University, Seoul, South Korea.

APPENDICES

APPENDIX A

APPENDICES A.1

Table A.1 : Performance results of ANN for 6-hours later prediction.

input	hidden layer	epoch	perfom1	perfom2	perfom3	perfom4	perfom5	best perform
1	3	500	2.1690	2.1501	2.1517	2.1676	2.1535	2.1501
1	4	500	2.1646	2.1746	2.1593	2.1512	2.1663	2.1512
1	5	500	2.1744	2.2011	2.1724	2.1795	2.1740	2.1724
2	5	500	2.1617	2.1731	2.1817	2.2051	2.1877	2.1617
2	6	500	2.1914	2.2648	2.1711	2.1699	2.2073	2.1699
2	7	500	2.2239	2.1817	2.2163	2.1795	2.1830	2.1795
3	7	500	2.2319	2.1893	2.1688	2.1996	2.2471	2.1688
3	8	500	2.1925	2.2060	2.2570	2.2317	2.1774	2.1774
3	9	500	2.1942	2.2606	2.2385	2.2080	2.2460	2.1942
4	9	500	2.2206	2.2389	2.1899	2.2432	2.2626	2.1899
4	10	500	2.2648	2.3068	2.2139	2.2319	2.2398	2.2139
4	11	500	2.2111	2.2910	2.3212	2.2477	2.2374	2.2111
5	11	500	2.2325	2.2618	2.2361	2.2436	2.2588	2.2325
5	12	500	2.2626	2.2297	2.2489	2.2255	2.3511	2.2255
5	13	500	2.2835	2.2984	2.2880	2.2201	2.3068	2.2201
6	13	500	2.2686	2.2357	2.2740	2.2536	2.2296	2.2296
6	14	500	2.2647	2.2504	2.2394	2.2545	2.3319	2.2394
6	15	500	2.2343	2.2424	2.2466	2.3062	2.2123	2.2123
7	15	500	2.2994	2.3130	2.2504	2.2803	2.3492	2.2504
7	16	500	2.2670	2.2331	2.2757	2.3609	2.2577	2.2331
7	17	500	2.2734	2.2934	2.3378	2.2566	2.2493	2.2493

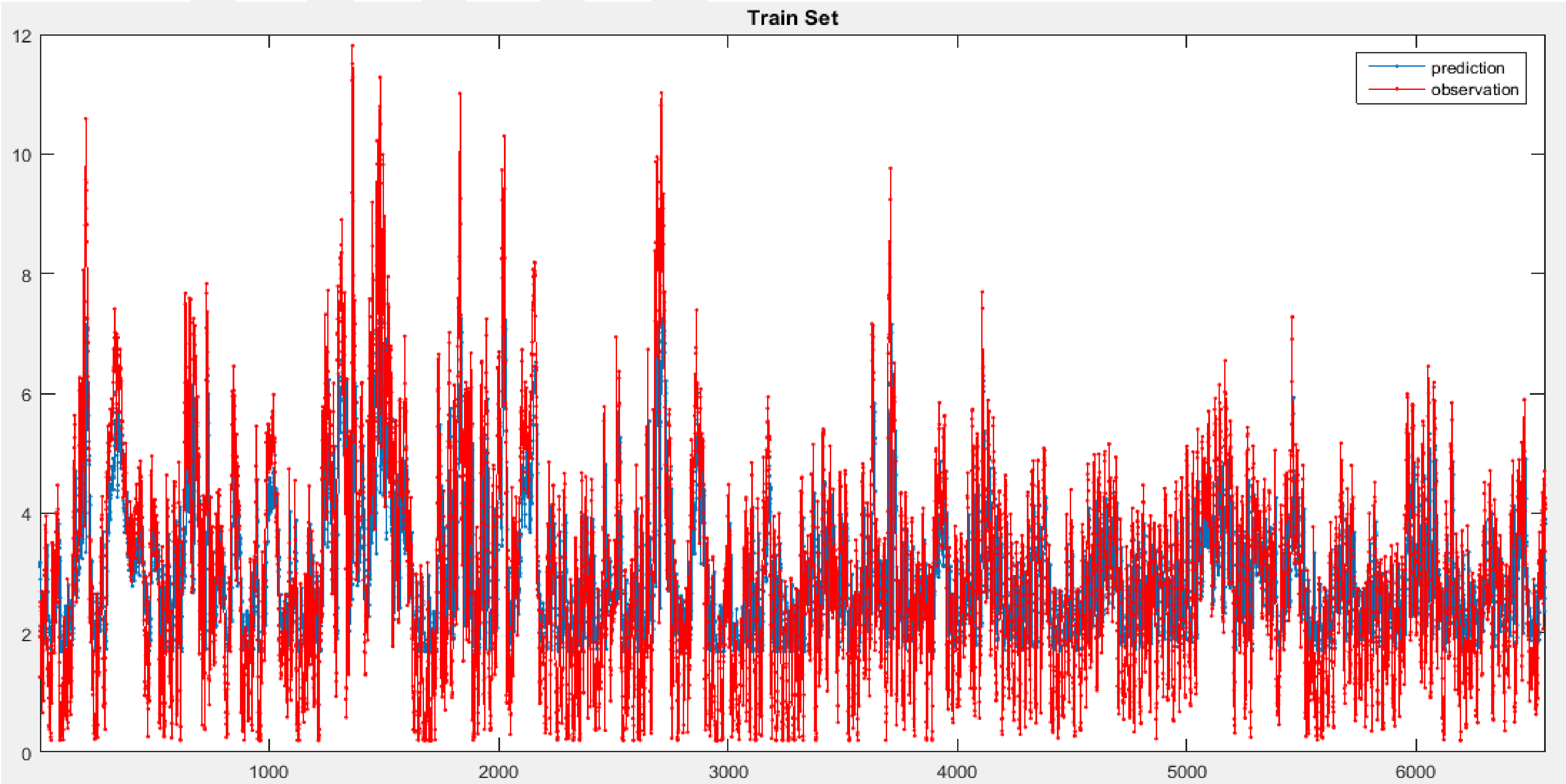

Figure A.1 : 6-hours later predictions vs observed wind speed data for train set .

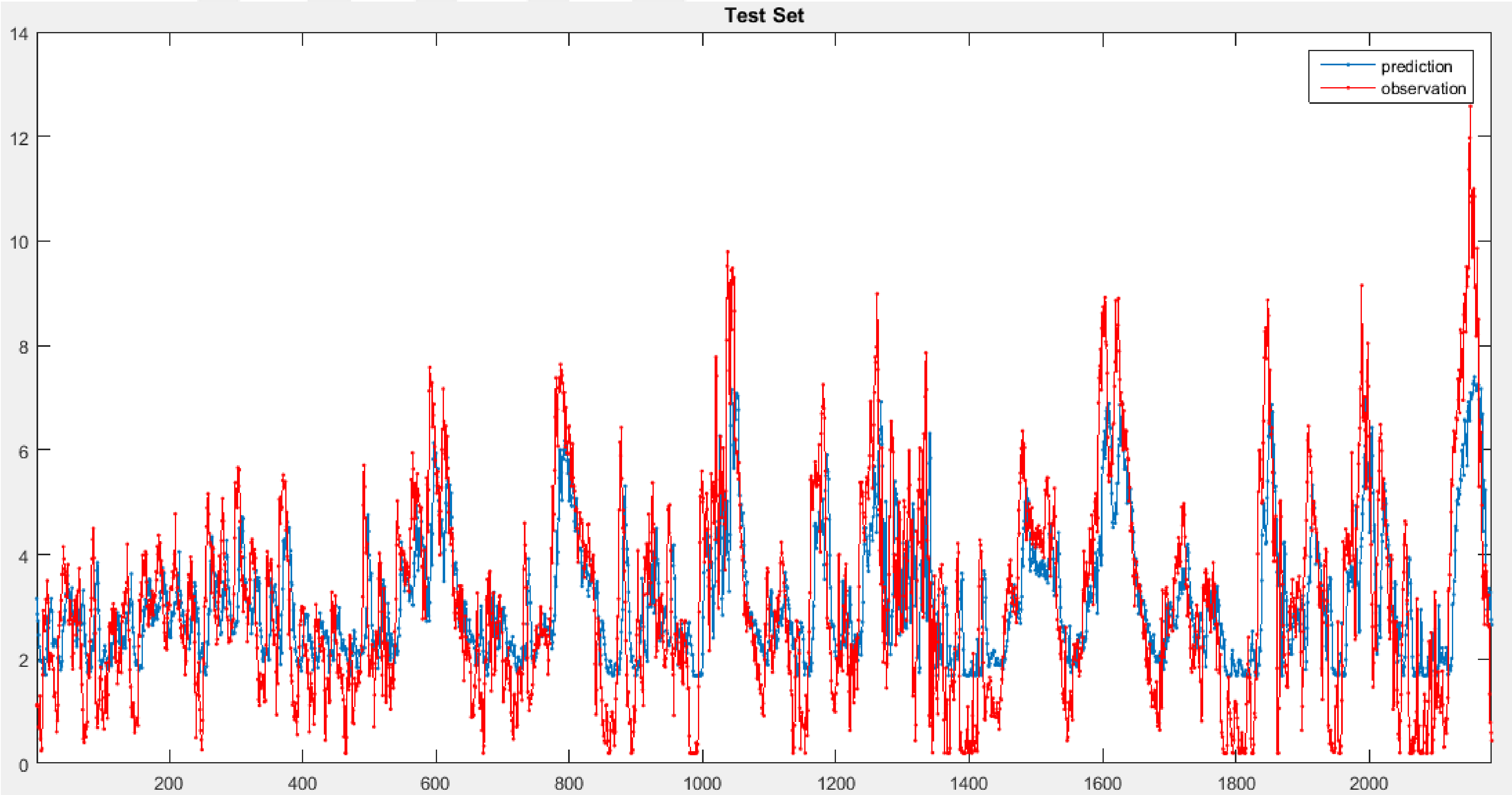

Figure A.2 : 6-hours later predictions vs observed wind speed data for test set.

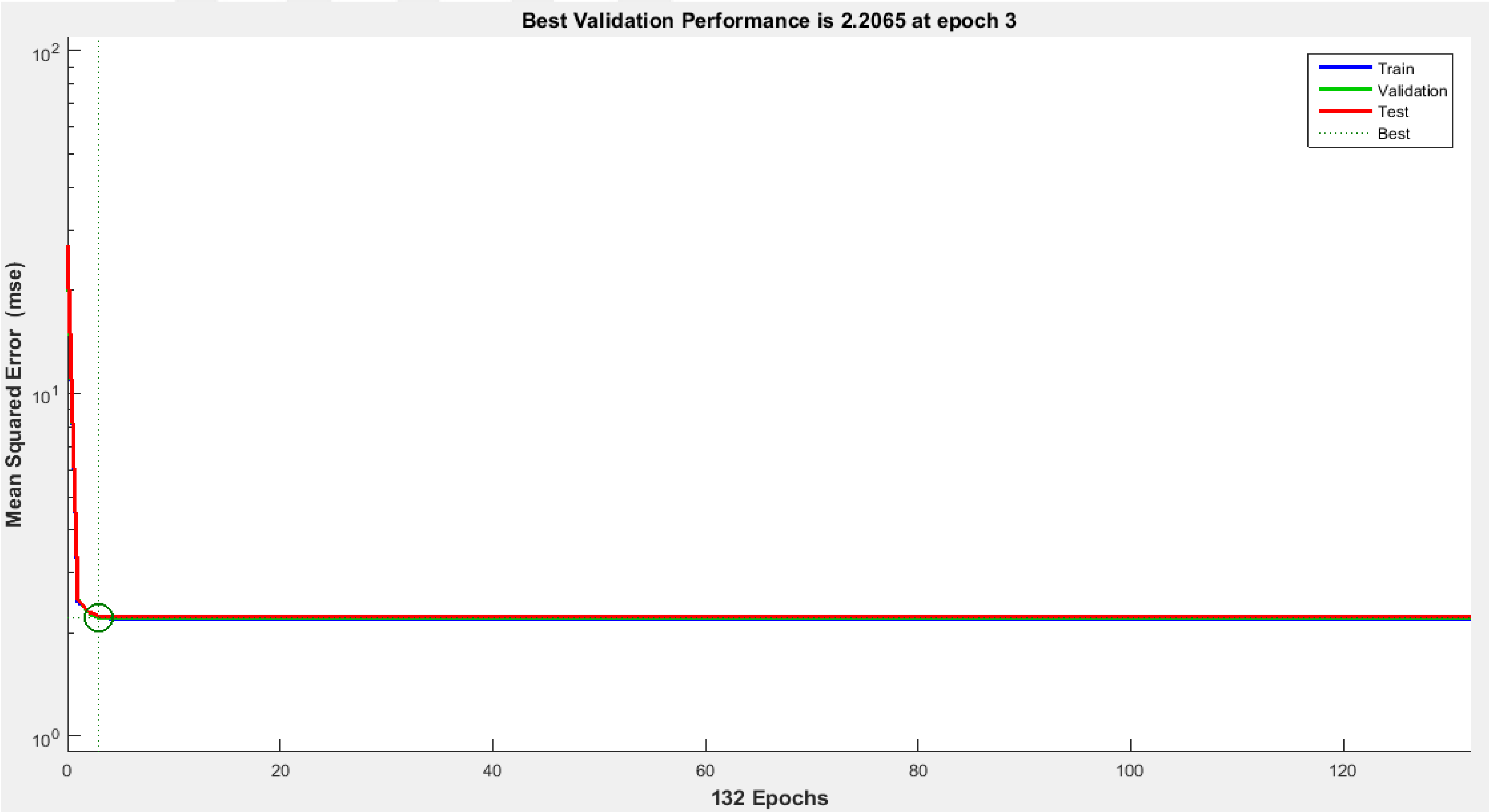

Figure A.3 : Performance analysis for 6-hours later prediction.

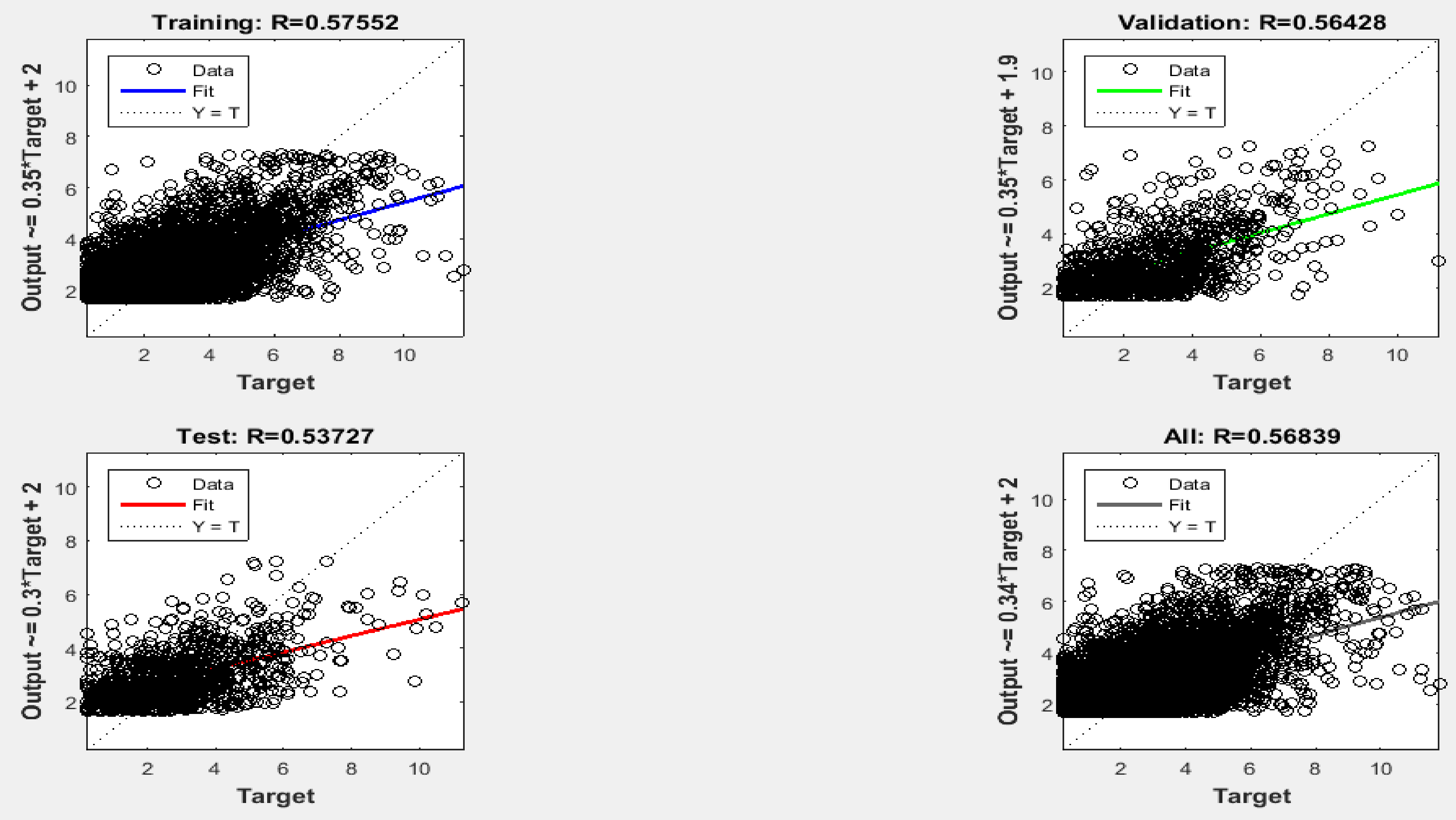

Figure A.4 : Regression analysis for 6-hours later prediction

Table A.2 : Performance results of ANN for 12-hours later prediction.

input	hidden layer	epoch	perfom1	perfom2	perfom3	perfom4	perfom5	best perform
1	3	500	3.1721	3.1927	3.1513	3.2022	3.1921	3.1513
1	4	500	3.1582	3.1784	3.1605	3.1701	3.1772	3.1582
1	5	500	3.1744	3.1570	3.1808	3.1657	3.2073	3.1570
2	5	500	3.2235	3.2021	3.2229	3.1787	3.2136	3.1787
2	6	500	3.2244	3.1878	3.1889	3.2539	3.2445	3.1878
2	7	500	3.1879	3.2329	3.1814	3.1849	3.1830	3.1814
3	7	500	3.2506	3.2293	3.2414	3.2403	3.2245	3.2245
3	8	500	3.2129	3.2902	3.1830	3.2340	3.1846	3.1830
3	9	500	3.2331	3.3153	3.2261	3.2330	3.1676	3.1676
4	9	500	3.2601	3.3609	3.3819	3.2304	3.2795	3.2304
4	10	500	3.2743	3.2485	3.2961	3.2440	3.2344	3.2344
4	11	500	3.2550	3.3358	3.2228	3.2605	3.2289	3.2228
5	11	500	3.3114	3.2690	3.2883	3.2702	3.2485	3.2485
5	12	500	3.2806	3.2363	3.2267	3.2899	3.3057	3.2267
5	13	500	3.2350	3.2817	3.2518	3.2829	3.5323	3.2350
6	13	500	3.2766	3.3247	3.2544	3.3158	3.2849	3.2544
6	14	500	3.2806	3.3358	3.3136	3.2752	3.3116	3.2752
6	15	500	3.3961	3.3426	3.2553	3.3544	3.2856	3.2553
7	15	500	3.2669	3.3321	3.2861	3.4303	3.3013	3.2669
7	16	500	3.3851	3.3559	3.2749	3.3827	3.2996	3.2749
7	17	500	3.3266	3.3055	3.2587	3.3781	3.3800	3.2587

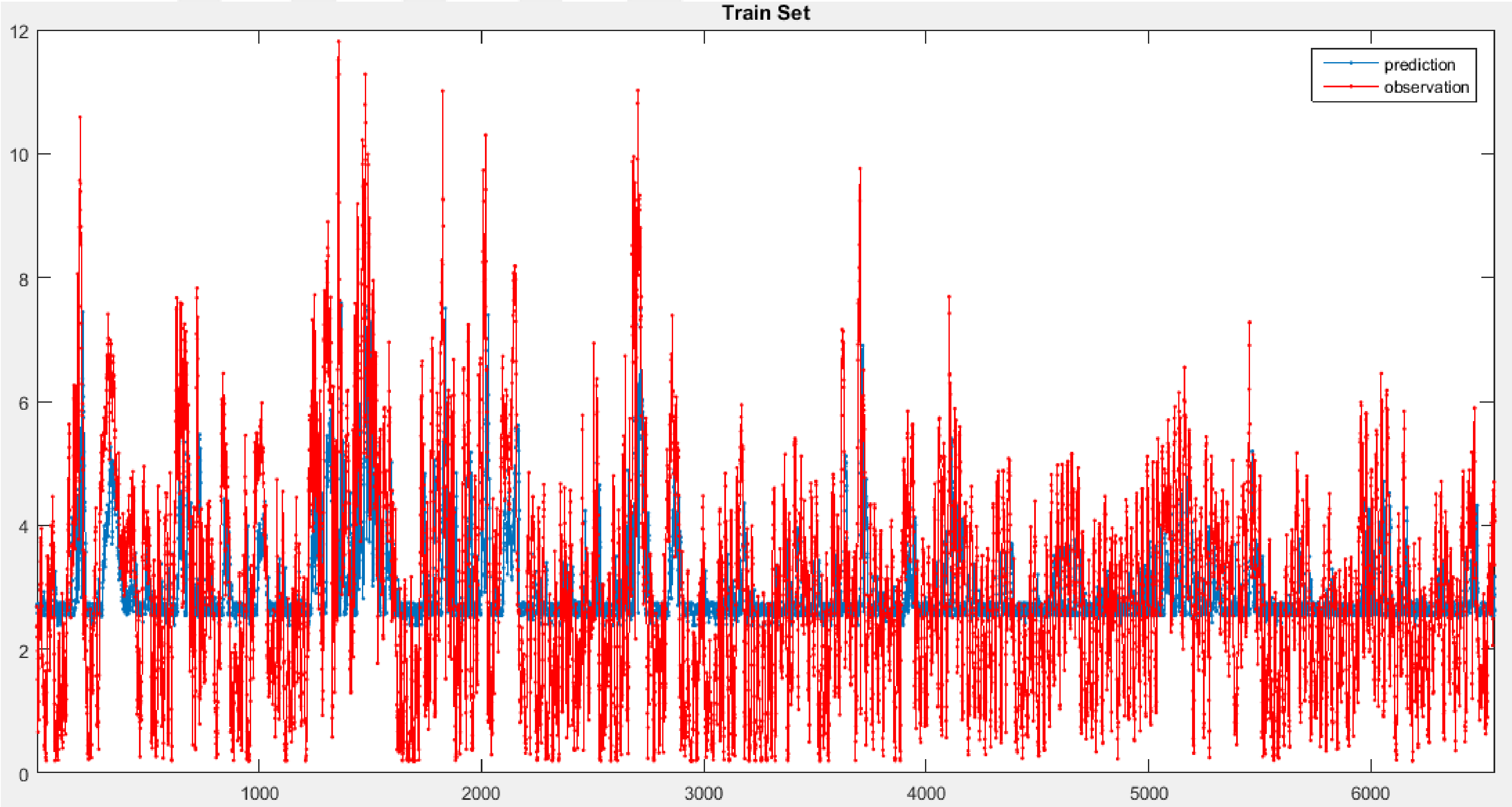

Figure A.5 : 12-hours later prediction vs observed wind speed data for train set.

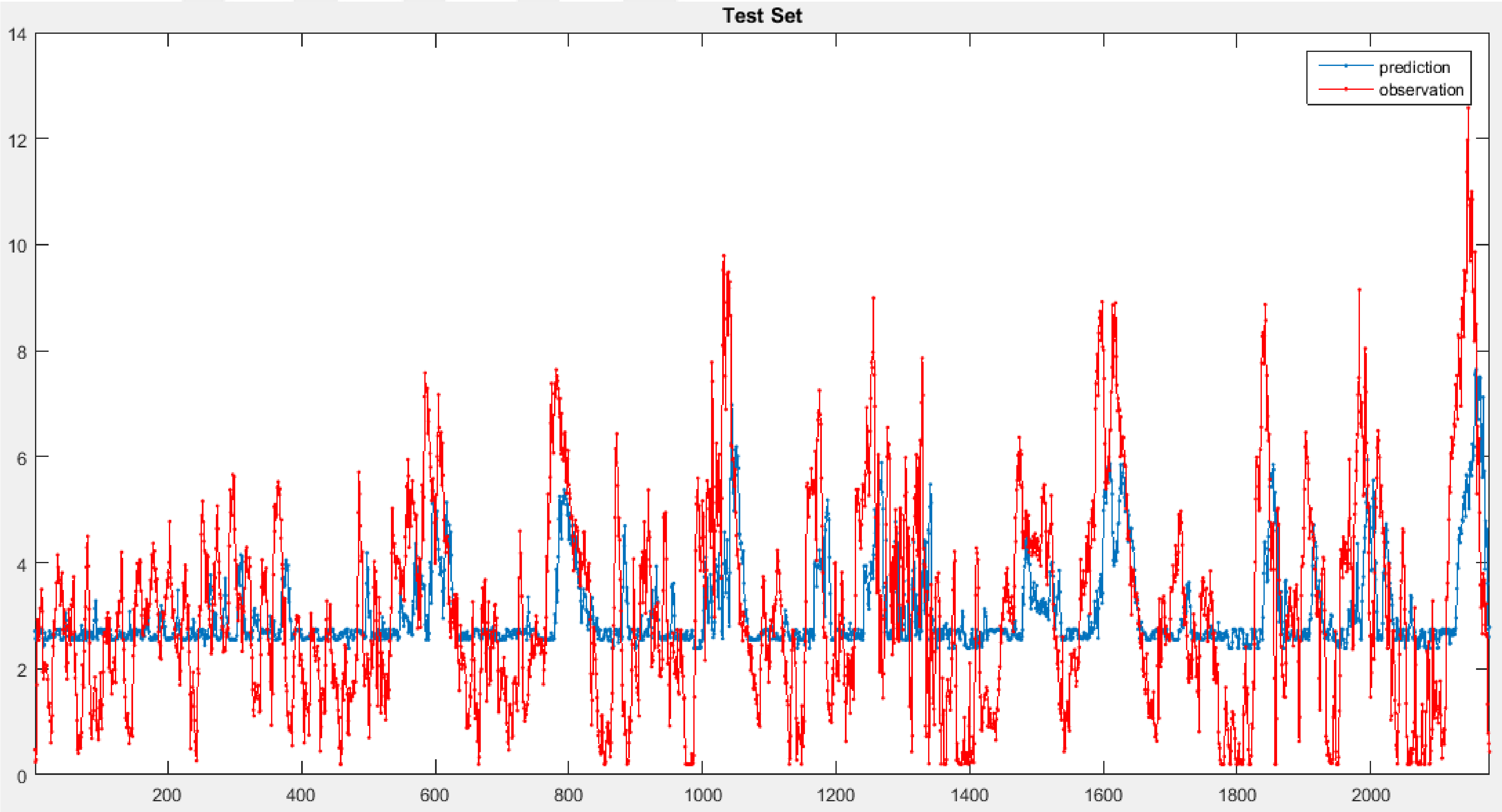

Figure A.6 : 12-hours later predictions vs observed wind speed data for test set.

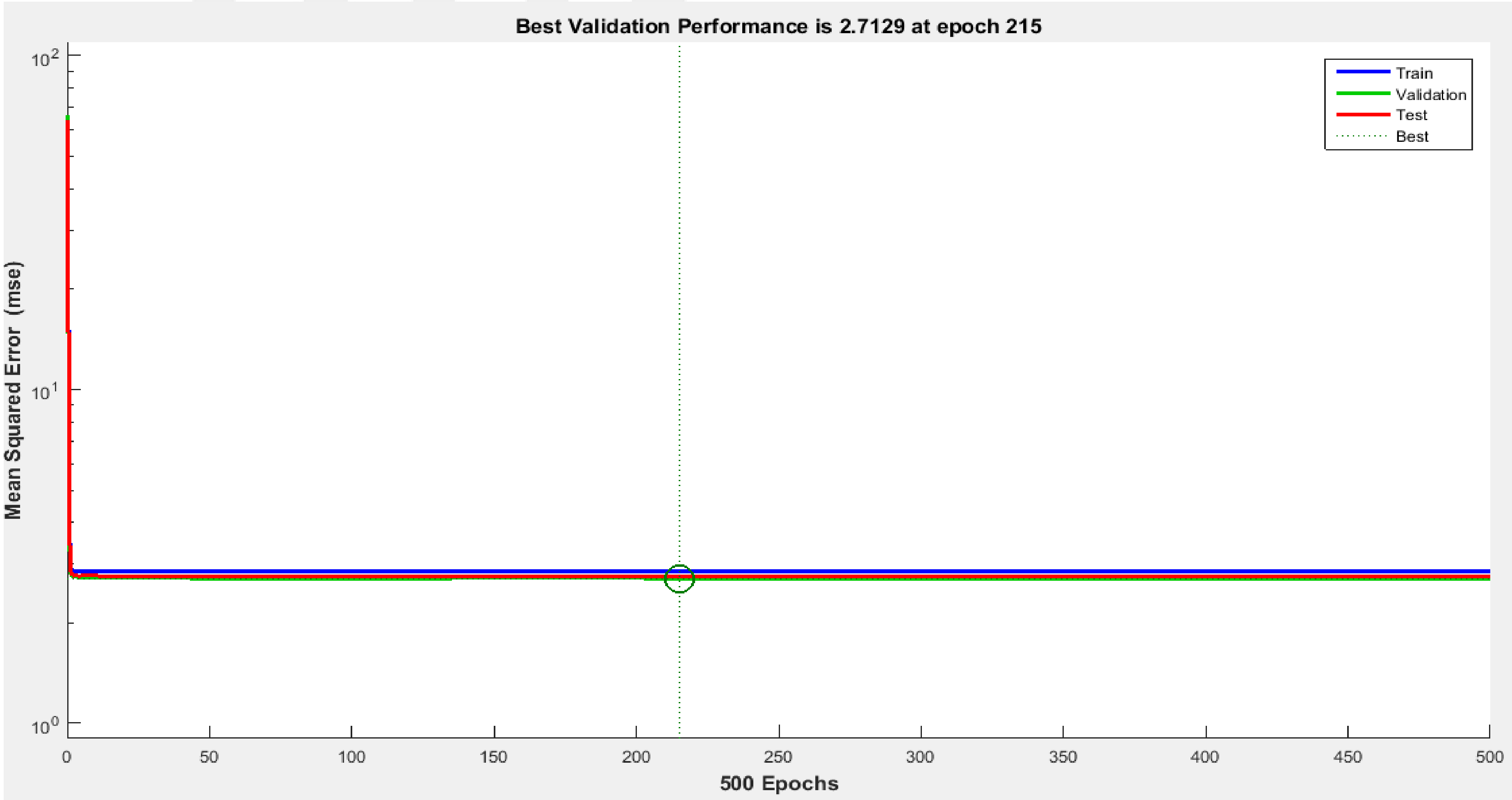

Figure A.7 : Performance analysis for 12-hours later prediction.

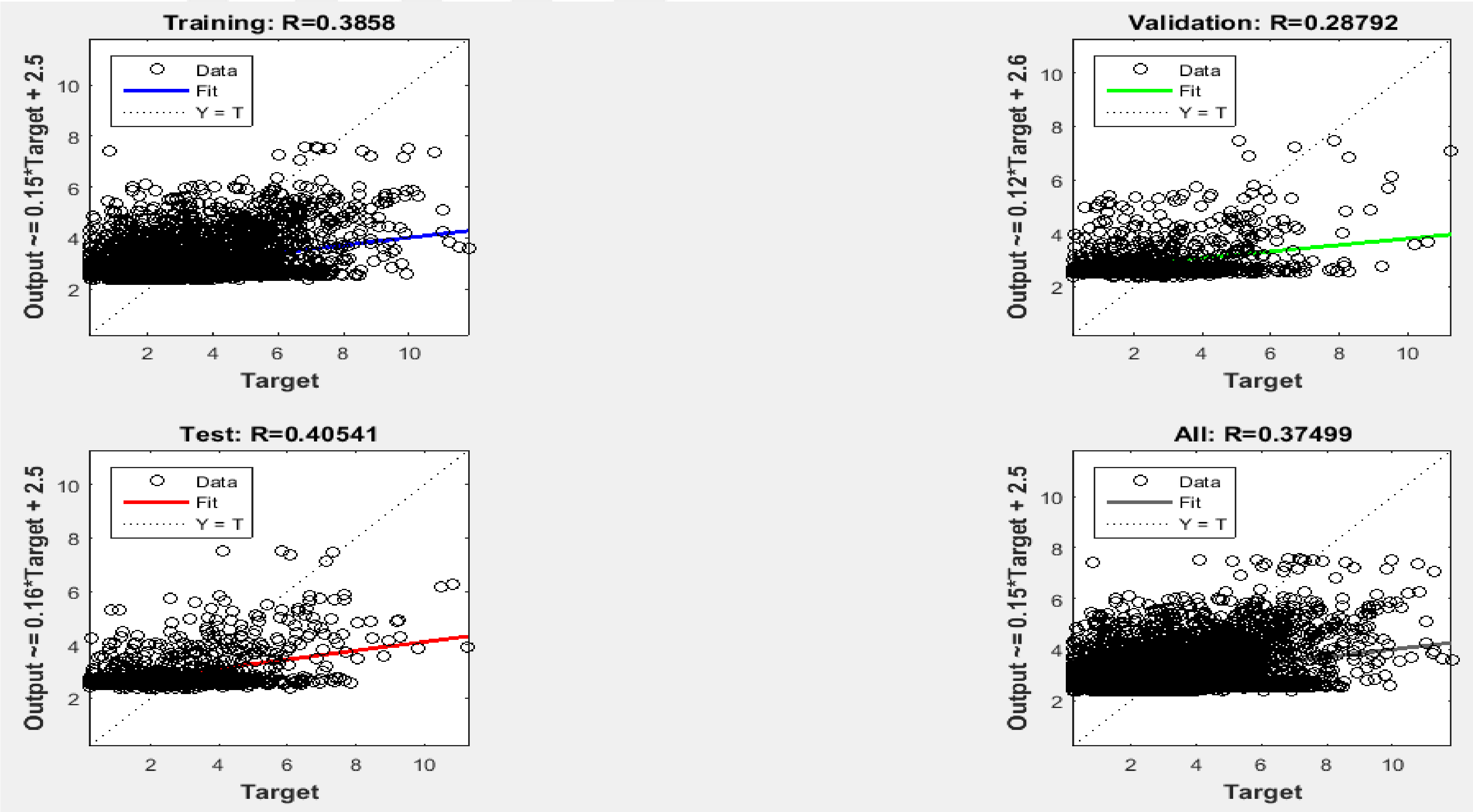

Figure A.8 : Regression analysis for 12-hours later prediction.

Table A.3 : Performance results of ANN for 24-hours later prediction.

input	hidden layer	epoch	perfom1	perfom2	perfom3	perfom4	perfom5	best perform
1	3	500	3.7784	3.8022	3.7950	3.7947	3.8116	3.7784
1	4	500	3.8064	3.8046	3.7418	3.8396	3.8212	3.7418
1	5	500	3.8332	3.8029	3.7988	3.8026	3.8176	3.7988
2	5	500	3.8328	3.8296	3.8742	3.7930	3.8449	3.7930
2	6	500	3.8848	3.8258	3.8322	3.8821	3.8187	3.8187
2	7	500	3.8729	3.8406	3.7942	3.8577	3.8318	3.7942
3	7	500	4.4235	3.8616	3.9355	3.7991	3.9548	3.7991
3	8	500	3.8429	3.8968	3.8183	3.7839	3.8759	3.7839
3	9	500	3.8735	3.8208	3.8262	3.8300	3.9078	3.8208
4	9	500	3.7938	3.8543	3.8455	3.8888	3.7693	3.7693
4	10	500	4.0512	3.8072	3.8595	3.8830	3.8990	3.8072
4	11	500	3.8607	3.8761	3.7978	3.8678	3.8061	3.7978
5	11	500	3.8301	3.8044	3.8171	3.8465	3.8059	3.8044
5	12	500	3.7105	3.8641	3.8107	3.8508	3.8255	3.7105
5	13	500	3.9288	3.9398	3.8548	3.7718	3.9533	3.7718
6	13	500	3.8601	3.8263	3.8946	3.8574	3.7032	3.7032
6	14	500	3.8285	3.9963	3.8360	3.8093	3.8701	3.8093
6	15	500	3.8996	3.8046	3.7535	3.8447	3.8332	3.7535
7	15	500	3.9084	3.7816	3.7362	3.8119	3.8199	3.7362
7	16	500	3.7891	3.9057	3.7807	3.8228	3.8050	3.7807
7	17	500	3.9008	3.8252	3.7535	3.9991	3.8187	3.7535

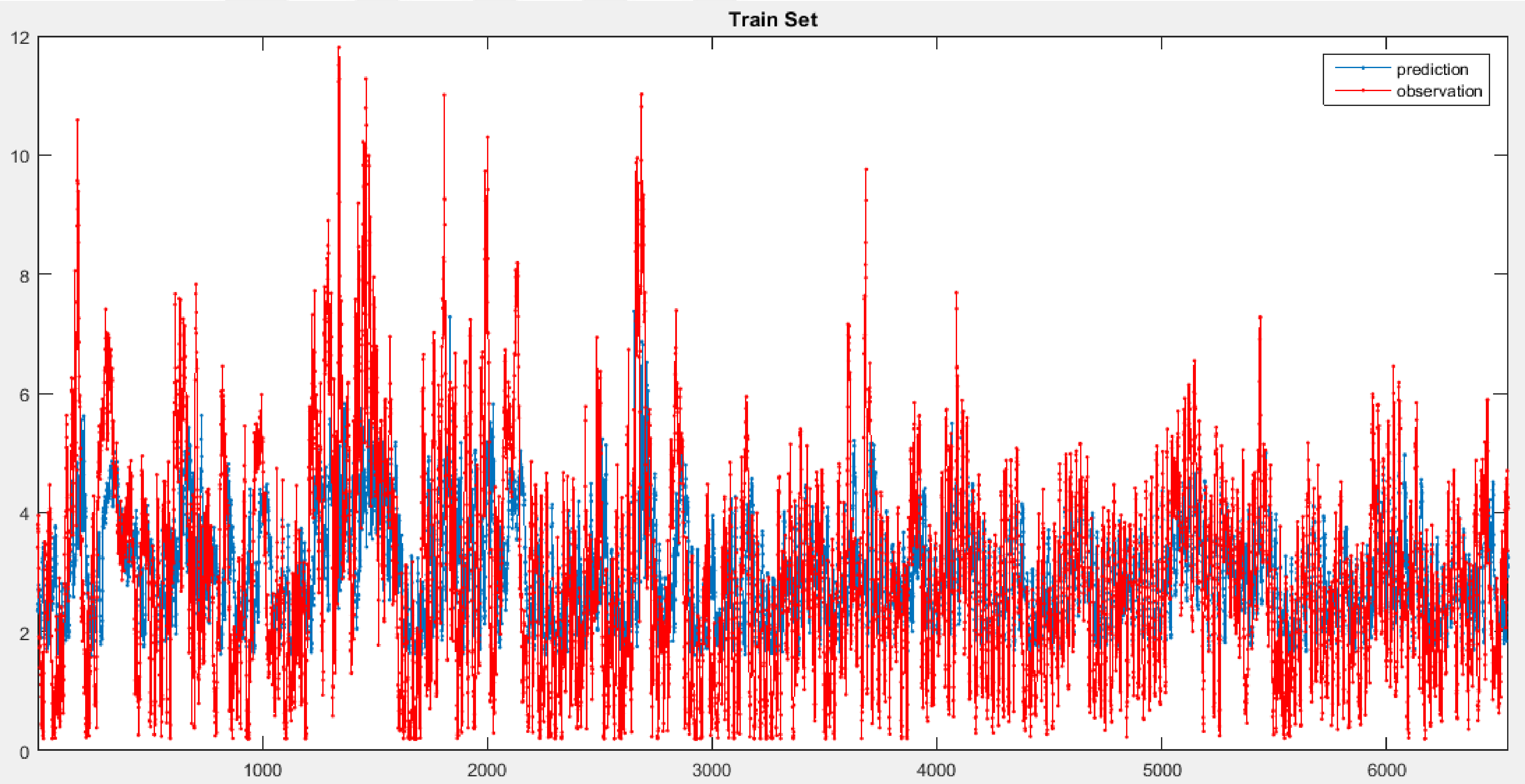

Figure A.9 : 24-hours later prediction vs observed wind speed data for train set.

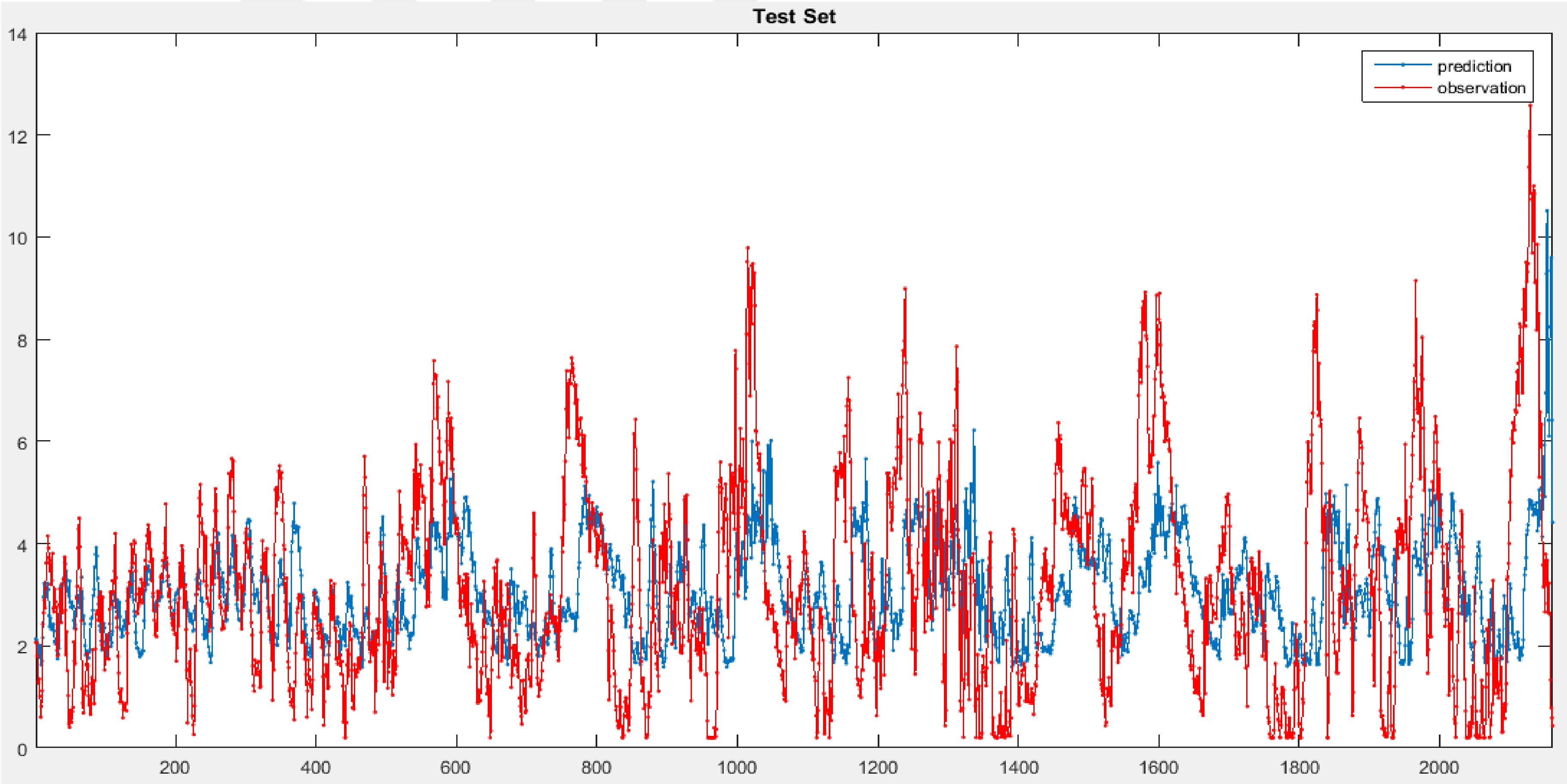

Figure A.10 : 24-hours later prediction vs observed wind speed data for test set.

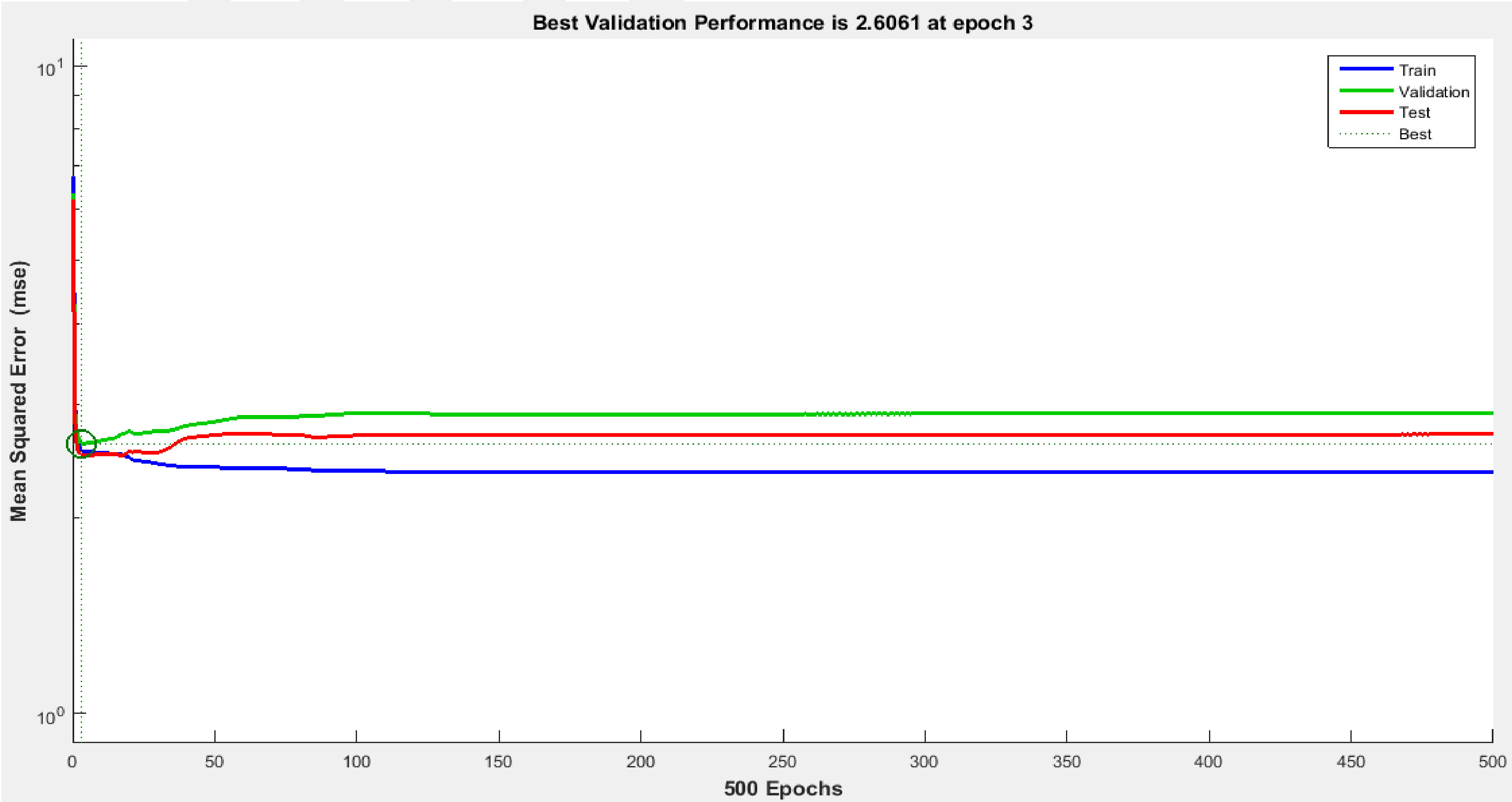

Figure A.11 : Performance analysis for 24-hours later prediction.

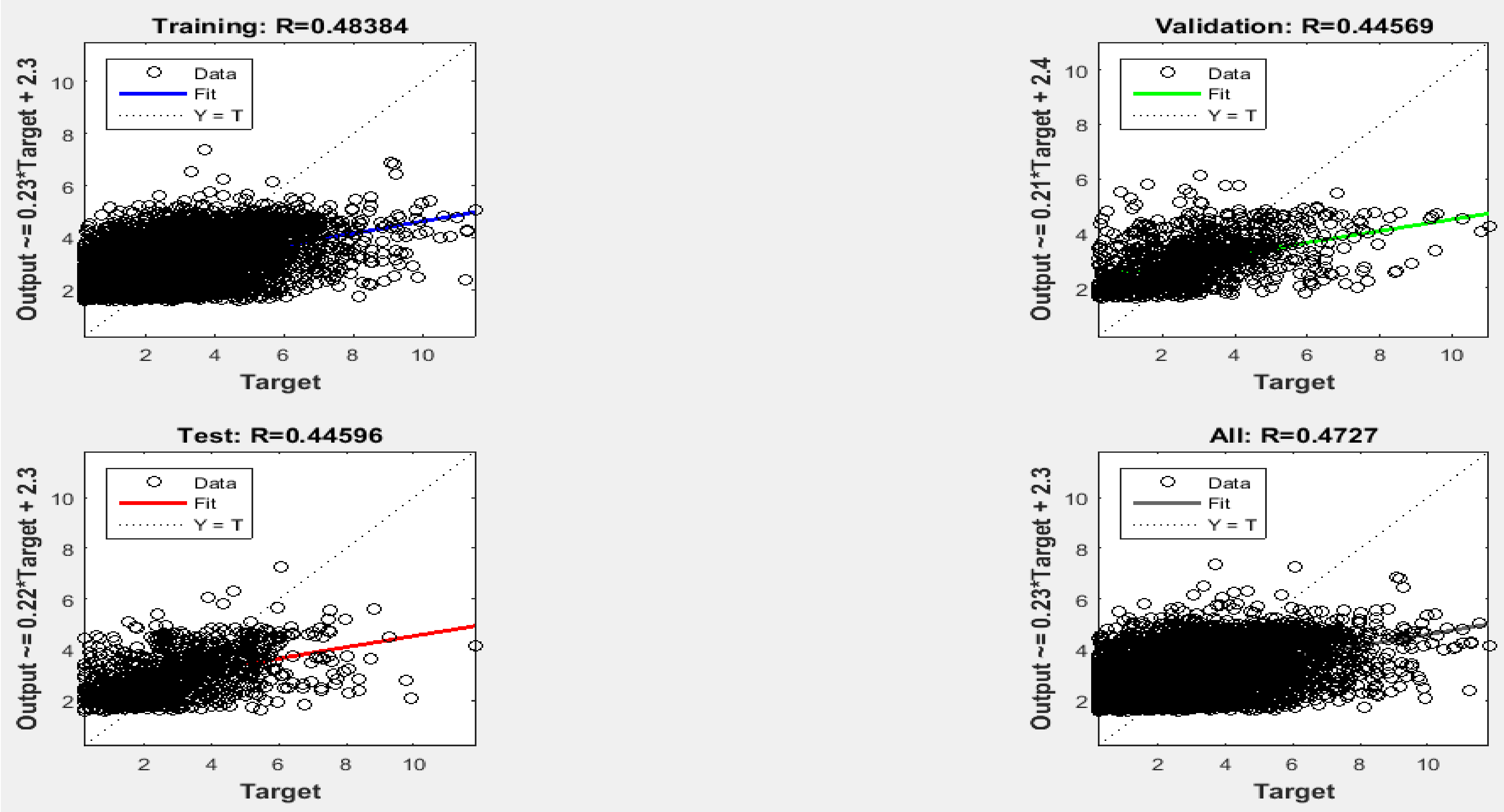

Figure A.12 : Regression analysis for 24-hours later prediction.

CURRICULUM VITAE

Name Surname: Coşkun Yıldız

Place and Date of Birth: Bayburt 23/09/1991

E-Mail: coskunyildizz@gmail.com

EDUCATION:

B.Sc.: 2013, Izmir Institute of Technology, Faculty of Engineering, Department of Mechanical Engineering