RASTSAL KODLARIN DESTEK AĞIRLIKLARININ İSTATİSTİKLERİ

2012
YÜKSEK LİSANS TEZİ
MATEMATİK

Eda TEKİN

RASTSAL KODLARIN DESTEK AĞIRLIKLARININ İSTATİSTİKLERİ

Eda TEKİN

Karabük Üniversitesi
Fen Bilimleri Enstitüsü
Matematik Anabilim Dalında
Yüksek Lisans Tezi
Olarak Hazırlanmıştır

KARABÜK
Ocak 2012

Eda TEKİN tarafından hazırlanan "RASTSAL KODLARIN DESTEK AĞIRLIKLARININ İSTATİSTİKLERİ" başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Doç. Dr. Ayşe NALLI .....
Tez Danışmanı, Matematik Anabilim Dalı

Yrd. Doç. Dr. Can Murat DİKMEN .....
Tez Danışmanı, Zonguldak Karaelmas Üniversitesi

Bu çalışma, jürimiz tarafından oy birliği ile Matematik Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 09/ 01/ 2012

Ünvanı, Adı SOYADI (Kurumu)	İmzası
Başkan : Doç. Dr. Ayşe NALLI (KBÜ)	.....
Üye : Yrd. Doç. Dr. Nil ORHAN ERTAŞ (KBÜ)	.....
Üye : Yrd. Doç. Dr. Hakan BOSTANCI (KBÜ)	.....
Üye : Yrd. Doç. Dr. Can Murat DİKMEN (ZKÜ)	.....
Üye : Yrd. Doç. Dr. İbrahim ÖZEN (ÇAKÜ)	.....

..../..../2012

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıştır.

Doç. Dr. Nizamettin KAHRAMAN .....
Fen Bilimleri Enstitüsü Müdürü

"Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim."

Eda TEKİN

ÖZET

Yüksek Lisans Tezi

RASTSAL KODLARIN DESTEK AĞIRLIKLARININ İSTATİSTİKLERİ

Eda TEKİN

Karabük Üniversitesi
Fen Bilimleri Enstitüsü
Matematik Anabilim Dalı

Tez Danışmanı:
Doç. Dr. Ayşe NALLI
Yrd. Doç. Dr. Can Murat DİKMEN
Ocak 2012, 59 sayfa

Bu tezde öncelikle kod tanımı, kodlama teorisinde çok önemli yeri olan lineer kodlar, parametreleri ve özellikleri incelenmiş, daha sonra genelleştirilmiş lineer kod tanımı verilip parametreleri incelenmiştir. Çalışmamızın 5. bölümünde, genelleştirilmiş kodlar için, *kxn* lik bir matrisin rankının *r* olma olasılığı formülünden yararlanılarak bu matrisin beklenti formülü, beklenti formülünden yararlanılarak farklı I ve J alt matrislerine karşılık gelen rank fonksiyonlarının kovaryansları, destek ağırlık polinomu $W_C^r(z)$ nin beklentisi, ve son olarak bir *C* [*n*,*k*] kodunun destek ağırlıkları arasındaki kovaryans formülleri elde edildi. Son bölümde ise lineer olmayan bir kod: Fibonacci koduna değinildi.

Anahtar Sözcükler : Genelleştirilmiş ağırlıklar, rastsal matrislerin rankları, destek ağırlıkların momentleri.

Bilim Kodu **:** 204.1.025

ABSTRACT

M.Sc. Thesis

STATISTICS OF THE SUPPORT WEIGHTS OF RANDOM CODES

Eda TEKİN

Karabük University
Graduate School of Natural and Applied Sciences
Department of Mathematics

Thesis Advisor:
Assoc. Prof. Dr. Ayşe NALLI
Assist. Prof. Dr. Can Murat DİKMEN
January 2012, 59 pages

In this thesis, firstly code definition is given. Linear codes, its parameters and their characteristics that play an important role in codin theory are investigated. Generalized linear codes are also defined and their parameters are investigated. In the fifth chapter, we obtain *kxn* matrices expectation formula by using the formula of probability of rank of a *kxn* matrix to be *r*; we obtain the rank functions coveriances that correspondes to *I* and *J* submatrices, formula of $W_C^r(z)$ support weight enumerator and finally we obtain the coveriance formulas within a *C* [*n,k*] codes support weight enumerators by using expectation formula. . In last chapter, Fibonacci code that is not a lineer code is studied.

Key Word : Generalized weights, ranks of random matrices, moments of the support weights.

Science Code : 204.1.025

TEŞEKKÜR

Bu tez çalışmasının planlanmasında, araştırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteğini esirgemeyen, iki dönem tez danışmanlığımı yürüten, engin bilgi ve tecrübelerinden tüm yüksek lisans dönemimde yararlandığım, yönlendirme ve bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren sayın hocam Yrd. Doç. Dr. İbrahim ÖZEN'e sonsuz teşekkürlerimi sunarım.

Tez danışmanı değişikliğinden sonra çalışmamın devamını sağlayan, her an beni destekleyen değerli hocam Doç. Dr. Ayşe NALLI'ya tüm kalbimle teşekkür ederim.

Ayrıca ZKÜ öğretim üyesi sayın Can Murat DİKMEN'e, KBÜ öğretim üyelerine ve bu süreçteki tüm özverisinden dolayı eşim Kubilay TEKİN'e teşekkürü bir borç bilirim.

İÇİNDEKİLER

Sayfa

SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

C : kod

q : alfabedeki simge sayısı

n : uzunluk

k : eleman sayısı

F_q^n : uzunluğu n olan q elemanlı sonlu cisim

u : Hamming uzaklığı

d : minimum mesafe

R : kod oranı

δ : göreceli minimum mesafe

w : ağırlık

A_i : ağırlık kümesi

$W_C(t)$: ağırlık polinomu (hesaplayıcısı)

G : üreten matris

$C^{\perp}$: C kodunun duali

H : (parite) kontrol matrisi

H_k (Ham(k): Hamming kodu

$\|D\|$: D nin destek ağırlığı

d_r : r. Destek ağırlığı

A_i^r : destek ağırlık kümesi

$W_C^r(t)$: destek ağırlık polinomu

E : beklenti

cov : kovayans

P : olasılık

BÖLÜM 1

GİRİŞ

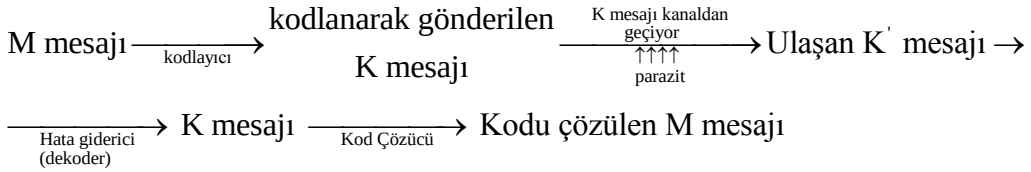
Kodlama teorisi ilk olarak 1940'lı yılların sonlarına doğru C. E. Shannon, R. W. Hamming ve J. C. Golay'ın yazdıkları makalelerde bazı mühendislik problemleri ile bağlantılı olarak ortaya çıkmıştır. Bu konu cebirdeki matematik kavramları kullanılarak geliştirilmiş ve "Cebirsel Kodlama Teorisi" adını almıştır.

Modern iletişimin gereksinimleri sonucu bugüne kadar çok sayıda kod inşa edildi ve bunun sonucu olarak bu kodların taşınması sırasında oluşabilecek hataların düzeltilmesi gerekliliğinden hata düzelten kodlar teorisi de oldukça gelişti. Bu teori sayesinde modern telekomünikasyon ve uzay iletişiminde verilerin hızlı ve doğru bir şekilde iletilmesi sağlandı [1]. Hata düzeltici kodlar teorisi bilgi transferi ya da depolanması esnasında orijinal bilgiye yapılan ekleri optimize etme ve iletilen bilgide meydana gelebilecek hataları düzeltme gibi konularla ilgilenir. Örneğin; bir mesajı bir kanal boyunca hızlı ve güvenilir bir şekilde iletmek isteyelim. Kanal bir telefon hattı, yüksek frekanslı bir radyo bağlantısı olabilir. Ekipman eksikliği, insan hatası ya da yıldırım sebebiyle bilginin iletimi sırasında hatalar oluşabilir. Bu hatalardan mesajı korumak için fazladan veri eklenir.

Hata düzeltici kodlamanın ardındaki düşünce, kod sözcüklerini birbirlerinden yeterince farklı seçip, gönderilen sözcüğün birkaç terimi parazit nedeniyle yanlış iletilse bile, iletilen sözcüğün hala tanınabilir, yani asıl sözcüğe diğer bütün kod sözcüklerine olduğundan daha yakın olmasını sağlamaktır.

Bilgi akışının gerçekleştiği ortama kanal denir; telefon hatları ya da atmosfer kanal örnekleridir. Parazit ya da gürültü adını verdiğimiz istenmeyen dış etkenler alıcıya ulaşan mesajın gönderilenden farklı olmasına neden olabilir. Aşağıdaki şemada bilgi akışı gösterilmektedir [2].

Yollamak istediğimiz mesaja M diyelim. Örneğin M, “evet” mesajı olabilir. Bu mesaj kodlanarak K kod sözcüğü haline getirilir. K’nın, 000 olduğunu kabul edelim. K, gideceği yere doğru kanalda yol alırken parazite maruz kalır ve ulaşacağı yere K yerine K’ olarak ulaşır. K’, 010 olabilir. Bu aşamada bir hata giderici devreye girer ve yanlış olan K’ mesajını K kod sözcüğü olarak düzeltir. Hata giderici 010 mesajının 000’a 111’den daha yakın olduğunu anlar.



İletişimde amaç, kaynaktan gönderilen mesajı doğruluğu yüksek bir olasılıkla iletmektir. Mesajı göndermek için alfabe olarak adlandırılan sonlu kümeler kullanılır. Bu küme genellikle sonlu bir halka veya cisim olarak alınır. İletilecek mesaj, oluşabilecek hatalardan korunması için şifrelenir. Şifrelenen mesaj, kodun elemanları olan kod sözcükleridir. Kod sözcükleri kanala gönderilir. Bazı terimleri değişmiş yani hata olmuş olabilir. Dekoder hata olup olmadığını kontrol eder, hata varsa düzeltir ve orjinal mesaj elde edilip alıcıya gönderilir.

Bir kodun minimum uzaklığı ne kadar büyük olursa o kod o kadar hata düzelteceğinden, minimum uzaklıkları büyük olan kodların elde edilmesi önemlidir. Araştırmacıların kodlar üzerine yapmış oldukları çalışmaların bir kısmı $\mathbb{F}_q$ sonlu cismi üzerinde yeni kodlar elde edilmesi ve bunlara karşılık gelen cebirsel kodların oluşturulması ile ilgilidir. Belirli halkalar üzerinde tanımlı kodlar kullanılarak da cisimler üzerinde kodlar elde edilebilir.

Halkalar üzerinde tanımlı lineer kodlarla ilgili çalışmalar 1970’lerde başlamıştır. Belirli halkalar ve bu halkaları belirlerken kullanılan Galois cisimleri arasında uygun dönüşümlerin tanımlanması yoluyla, belirli halkalardaki kodlarla bu cisimlerdeki kodlar arasındaki ilişkiler belirlenmiştir.

Bu tezin içeriği kısaca özetlenirse, tezin 2. bölümünde genel olarak kod tanımı, kodların yapısını oluşturan parametreler ve bu parametrelerin özelliklerine değinilmiştir ve birkaç kod örneği verip bunların parametreleri ile verimlilikleri karşılaştırılmıştır. Tezin 3. bölümünde bizim üzerinde çalıştığımız lineer kodlar verilip bu kodların temel tanımları, parametrelerini verilmiştir. Lineer kodların üreteç ve kontrol matrisleri, bu matrislerle ilgili teoremler ve örnekler verildi. Bölümün sonunda temel lineer kodlara birkaç örnek verilmiştir. 4. bölümde yeni ancak hızla gelişen alan olan genelleştirilmiş ağırlık tanımlarını verildi. Genelleştirilmiş kodlarla ilgili destek ağırlık tanımı, ağırlık kümesi, ağırlık polinomu (ağırlık hesaplayıcı) gibi önemli tanımlara yer verildi. 5. bölümde bu tanımlar ile ilgili ispatları yaparken yararlandığımız temel formüllere de 4. bölümde değinildi. Teorik ispatların yapıldığı 5. bölümde genelleştirilmiş kodlar için, *kxn* bir matrisin rankının *r* olma olasılığı formülünden yararlanılarak bu matrisin beklenti formülü, beklenti formülünden yararlanılarak farklı I ve J alt matrislerine karşılık gelen rank fonksiyonlarının kovaryansların, destek ağırlık polinomu $W_C^r(z)$ nin beklentisi, ve son olarak bir *C* [*n,k*] kodunun destek ağırlıkları arasındaki kovaryans formülleri elde edilmiştir. Son olarak 6. Bölümde, lineer kodların destek ağırlıklarının istatistiklerinin Maple programı yardımıyla da elde edilebildiğini gösteren örnekler verilmiştir. 7. Bölümde ise lineer olmayan bir kod çeşidi fibonacci kodlarına değinilmiştir ve fibonacci kodları için kodlama ve dekodlama metodu verilmiştir.

BÖLÜM 2

KOD TANIMI VE LİNEER KODLARIN BAZI ÖZELLİKLERİ

2.1. KOD TANIMI VE ÖZELLİKLERİ

Tanım 2.1: Haber ve bilgiler gönderilirken iletişimde kolaylık sağlamak amacıyla bazen, mesajda kullanılan harf, sözcük ya da sözcük grupları belirli bir kurala göre başka simgelerle değiştirilir. Bu amaçla kullanılan simgeler sistemine kod denir.

Tanım 2.2: İletişimde ve bilgi aktarımında kodlama, bir kaynaktan alınan bilginin sembollere dönüştürülme sürecidir.

Tanım 2.3: Kod sembollerinin alıcı tarafından anlaşılabileceği bir şekilde bilgiye dönüştürülmesine dekodlama denir.

Tanım 2.4: Bir lineer kod C, $\mathbb{F}_q$ q elemanlı sonlu bir cisim olmak üzere, $\mathbb{F}_q^n$ in bir lineer alt uzayıdır. Burada q ya alfabedeki simge sayısı, n ye kodun uzunluğu, C nin boyutu k ya da kodun bilgi sembollerinin sayısı denir

Tanım 2.5: Her $a=(a_1,a_2,...,a_n)$, $b=(b_1,b_2,...,b_n)\in\mathbb{F}_q^n$ için a ve b arasındaki Hamming uzaklığı $d(a,b)=\left|\{i|a_i\neq b_i\}\right|$ şeklinde tanımlanır ve

$$d:\mathbb{F}_q^n\times\mathbb{F}_q^n\to\mathbb{N}\cup\{0\}$$
$$(a,b)\to d(a,b)$$

biçiminde tanımlanan dönüşüm;

i) $\forall a,b \in \mathbb{F}_q^n$ için $d(a,b) \geq 0, d(a,b) = 0 \Leftrightarrow a = b$

ii) $\forall a,b \in \mathbb{F}_q^n$ için $d(a,b) = d(b,a)$

iii) $\forall a,b,c \in \mathbb{F}_q^n$ için $d(a,b) \leq d(a,c) + d(c,b)$

özelliklerini sağlar ve bir metriktir.

Tanım 2.6: Bir C kodunun minimum uzaklığı $d(C) = \min\{d(x,y) | x \neq y,\ x,y \in C\}$ biçiminde tanımlanır. Uzunluğu *n*, eleman sayısı *k*, minimum uzaklığı *d* olan bir *C* koduna $[n,k,d]$ kodu denir. Burada *n,k,d C* kodunun parametreleridir.

C , $[n,k,d]$ kod olsun. Bu parametreler arasında aşağıdaki ilişkiler vardır;

i) *k* yeteri kadar büyükse kod fazla sayıda mesajı şifreler.
ii) *d* büyükse kod daha fazla hata düzeltir.
iii) *k* büyüdükçe d küçülür.
iv) *n* küçüldükçe mesaj daha hızlı iletilir.

Kodu belirleyen üç parametreden ikisi belli iken diğerinin alabileceği en iyi değeri belirlemek önemlidir.

Tanım 2.7: *C,* alfabesi *q* elemandan oluşan bir $[n,k,d]$ kod olsun. Verilen *n* ve *d* değerleri için *k* nin alabileceği en büyük değer $A_q(n,d)$ ile gösterilir. Buna *C* kodunun sınırı denir.

Örnek 2.1: C = {(0,0,0,0,0),(0,1,1,0,1),(1,0,1,1,0),(1,1,0,1,1)} bir $[5,4,3]$ koddur.

Teorem 2.1: Her n $\geq$1 için

i) $A_q(n,d) = q^n$

ii) $A_q(n,n) = q$ dir [1].

Teorem 2.2: q=2 olan bir kod için d tek olmak üzere bir $[n,k,d]$ kodunun var olması için gerek ve yeter koşul bir $[n+1,k,d+1]$ kodunun var olmasıdır [3].

Sonuç 2.1: d tek sayı ise $A_2(n+1,d+1)=A_2(n,d)$

d çift sayı ise $A_2(n-1,d-1)=A_2(n,d)$ dir [3].

Örnek 2.2: $A_2(5,3)=4$ olduğuna göre yukarıdaki sonuçtan $A_2(6,4)=4$ tür. $[6,4,4]$ kodu aşağıdaki gibi $[5,4,3]$ kodundan elde edilir. C $[5,4,3]$ kodu

C = {(0,0,0,0,0),(0,1,1,0,1),(1,0,1,1,0),(1,1,0,1,1)} idi. Bu durumda D $[6,4,4]$ kodu

D = {(0,0,0,0,0,0),(0,1,1,0,1,1),(1,0,1,1,0,1),(1,1,0,1,1,0)} olur.

Teorem 2.3: *C*, d minumum uzaklığına sahip bir kod olsun.

i) d ≥k+1 ise *C* kodu herhangi bir kod sözcüğündeki *k* tane hatayı tespit eder.
ii) d ≥2t+1 ise *C* kodu herhangi bir kod sözcüğündeki *t* tane hatayı düzeltir [3].

Sonuç 2.2: *d* minimum uzaklığına sahip olan bir *C* kodu herhangi bir kod sözcüğünde *d*-1 tane hatayı tespit etmekte ya da $\frac{d-1}{2}$ tane hatayı düzeltmekte kullanılır [4].

Tanım 2.8**:** *C* ve *D* iki $[n,k,d]_q$ kod olsun.

i) *D* kodu *C* nin kod sözcüklerinde aynı yerdeki elemanlara aynı permütasyon uygulanarak elde edilsin.
ii) *D* kodu *C* nin kod sözcüklerinde iki bileşenin yer değiştirmesi ile elde edilsin.

Yukarıdaki koşullardan en az biri sağlanıyorsa *D* kodu, *C* koduna denktir denir.

Örnek 2.3: C = {(0,0,1,1),(0,1,2,1),(1,2,2,0),(1,0,2,1)} kodu için $\sigma = \begin{pmatrix} 0 & 1 & 2 \\ 2 & 0 & 1 \end{pmatrix}$ permütasyonu C kodunun 3. konumundaki sembollere uygulanırsa, yeni elde edilen kod C_1 = {(0,0,0,1),(0,1,1,1),(1,2,1,0),(1,0,1,1)} olur. Daha sonra 1. konumdakilerle 4. konumdakiler yer değiştirildiğinde C_2 = {(1,0,0,0),(1,1,1,0),(0,2,1,1),(1,0,1,1)} kodu elde edilir. Burada C kodu C_2 koduna denktir.

Teorem 2.4: Bir $[n,k,2t+1]_q$ kod ve $A_q(n,d)=m$ olmak üzere

$m.\left\{\binom{n}{0}+\binom{n}{1}.(q-1)+...+\binom{n}{t}.(q-1)^t\right\} \le q^n$ eşitsizliğini sağlar [5].

Tanım 2.9: Bir $[n,k,2t+1]_q$ kodu için $A_q(n,d)=m$ olmak üzere

$m.\left\{\binom{n}{0}+\binom{n}{1}.(q-1)+...+\binom{n}{t}.(q-1)^t\right\} = q^n$

şartı sağlanıyorsa bu koda mükemmel kod denir.

Örnek 2.4: Bir $[n,2,n]_q$ kodu mükemmel bir koddur.

2.2. BAZI ÖZEL KODLAR

2.2.1. Kod A (8,7) Parite Kontrol Kodu

Birçok bilgisayarda bilgi birimi olarak bir byte, 8 bitlik bir dizi, kullanılır. Örneğin mikro bilgisayarlar için kullanımı evrensel olan ASCII kodu ‘a’, ‘B’, ‘3’ gibi karakterleri byte yardımıyla sunar. Bir byte 0 ile 255 arasında her değeri temsil edebilir. Örneğin 1 i temsil eden ASCII kodu $1 \longleftrightarrow 1000110$ dir. Biz bu kodu 10001101 olarak kodlarız. Benzer şekilde $A \longleftrightarrow 1000001$ olan ASCII kodu 10000010 olarak kodlanır. Böylece bir byte transfer edildiğinde ve bir bit yanlış iletildiğinde bu bytetaki 1 lerin sayısı tek olacaktır. Böylece alıcı tekrar mesajın yollanmasını isteyebilir. Burada alıcının hangi bitin yanlış iletildiğini söylemesi

mümkün değildir. Ayrıca bu bytetaki herhangi 2 bitin yanlış iletilmesi durumunda da alıcı hatayı fark etmeyerek mesajın geçmesine izin verecektir. Kısaca bu kodu değerlendirecek olursak;

i) (8,7) parite kontrol kodu çok ekonomiktir (encode edilen mesaj orijinal mesajın 1/7 i kadar daha uzundur).
ii) Hatalar düzeltilemez bu yüzden sadece alıcının, mesajın tekrar gönderimini isteyebileceği şartlarda uygundur (hatalar tespit edilirken hatanın nerede olduğu bulunamaz).
iii) Transfer sürecindeki hata olasılığı çok düşük olmalıdır (kod bir bytetaki iki hatayla baş edemez) [6].

2.2.2. Kod B Üçlü Tekrar Kodu

Mesajın doğru iletilmesinden çok emin olmak isteyen ekstra tedbirli bir telgraf operatörünü hayal edelim. Operatör her biti 3 defa tekrarlamaya karar veriyor. $0 \to 000,\ 1 \to 111$. Alıcıya 101 bloğunun ulaştığını kabul edelim. İki 1 in yanlış iletilme olasılığına göre bir 0 ın yanlış iletilme olasılığının daha büyük olduğunu düşünerek bu mesajı 111 şeklinde düzeltecektir. Bu düzeltme tekrar bilginin istenmesinden daha hızlı olacaktır ancak, iki 1 in yanlış gitme olasılığı da her zaman var olduğundan yine de bir risk vardır. Kısaca;

i) Kod ekonomik değildir (encode edilen mesaj orjinalden üç kat daha uzundur).

ii) Üçlü blok koddaki tek hataları düzeltebilir veya alternatif olarak tekrar mesajın istenmesinin mümkün olduğu durumlarda tek veya çift hataları tespit edebilir.

iii) Hatayı düzeltmede olabilecek hata olasılığı orta, hatayı tespit etmede hata olasılığı oldukça düşüktür [6].

2.2.3. Kod C Üçlü Kontrol Kodu

abc, a,b ve *c* ler 0 ve 1 olmak üzere bu koda 'xyz' üçlü kontrol bitlerini ekleyerek *abc* mesajını üçlü bloklara aşağıdaki gibi bölelim:

i) *abx* deki 1 lerin sayısı çift olsun,
ii) *acy* deki 1 lerin sayısı çift olsun,
iii) *bcz* deki 1 lerin sayısı çift olsun.

Örneğin *abc*=110 için *x*=0, *y*=1, *z*=1 dir. Böylece kod kelimemiz 110011 şeklinde kodlanır. Üçlü kontrol kodu sadece hatayı tespit etmekle kalmaz, hatayı düzeltir.

a yanlış iletildiyse (1) ve (2) şartları sağlanmaz.
b yanlış iletildiyse (1) ve (3) şartları sağlanmaz.
c yanlış iletildiyse (2) ve (3) şartları sağlanmaz.
x yanlış iletildiyse (1) şartı sağlanmaz.
y yanlış iletildiyse (2) şartı sağlanmaz.
z yanlış iletildiyse (3) şartı sağlanmaz.

Ancak bu hatalardan başka bir hatanın da bu şartları sağlamamaya yol açması mümkündür. Örneğin *a* ve *x* yanlış iletildiyse, sadece (2) şartı sağlanmaz ve eğer alıcı yukarıdaki düzeltme stratejisini uygularsa aslında doğru iletilen *y* yi düzeltir [6].

BÖLÜM 3

LİNEER KODLAR VE PARAMETRELERİ

Bu bölümde lineer kodların temel kavramlarına değinilmiştir ve ispatları yaparken kullandığımız notasyonlar tanımlanmıştır. Konumuza çok temel bir kavram olan lineer kodun tanımıyla başlayalım.

Tanım 3.1: Bir lineer kod C, $\mathbb{F}_q$ q elemanlı sonlu bir cisim olmak üzere, $\mathbb{F}_q^n$ in bir lineer alt uzayıdır. Burada q ya alfabedeki simge sayısı, n ye kodun uzunluğu, C nin boyutu k ya da kodun bilgi sembollerinin sayısı denir.

Tanım 3.2: Bir m kod vektöründeki sıfırdan farklı koordinatların sayısına m nin ağırlığı denir, $|m|$ ile gösterilir.

Hata düzeltici kodlamanın ardındaki düşünce, kod sözcüklerini birbirlerinden yeterince farklı seçip, gönderilen sözcüğün birkaç terimi parazit nedeniyle yanlış iletilse bile, iletilen sözcüğün hala tanınabilir, yani asıl sözcüğe diğer bütün kod sözcüklerine olduğundan daha yakın olmasını sağlamaktır. Bu yakınlığı ölçmek için bir uzaklık fonksiyonu kullanılır.

Tanım 3.3: $u(a,b)$ mesafesi, a ve b nin birbirinden farklı koordinatlarının sayısıdır, kısaca;

$$u\left((a_1,...,a_n),(b_1,...,b_n)\right)=\left|\{i\,|\,a_i\neq b_i, i=1,2,...,n\}\right|=:\left|\vec{a}-\vec{b}\right|$$

şeklinde tanımlanır. Bu u fonksiyonuna Hamming uzaklığı denir.

Örnek 3.1:

u(000, 111) = 3,

u(0100101, 1101101) = 2,

u(01101010, 01101010) = 0.

Tanım 3.4: C kodunun minimum mesafesi (uzaklığı), negatif olmayan

$d : CxC \to \mathbb{Z}$

$d = \min\left\{u(a,b) \middle| a,b \in C, a \neq b\right\}$ tam sayısıdır.

Örnek 3.2: C = {1101000, 0110100, 0011010, 0001101, 1000110, 0100011, 1010001} olsun. $a \neq b$ olan her kod sözcüğü için, u(a, b) = 4 olduğuna göre d(C) = 4 tür.

Tanım 3.5: C kodunun tek bir simge hatasını düzeltmeye olanak verebilmesi için, herhangi iki kod sözcüğünün arasındaki uzaklık en az 3 olmalıdır, yani d(C)≥ 3 olmalıdır. C kodunun iki simge hatasını düzeltmeye olanak verebilmesi için de, d(C) ≥ 5 olmalıdır. Genel olarak, C kodunun e simge hatasını düzeltmeye olanak verebilmesi için, d(C) ≥ 2e + 1 olmalıdır [1]. Bu durumda, C ye e-hata düzelten kod denir, yani C en fazla e tane hata düzeltme özelliğine sahiptir. Ayrıca t minimum uzaklığına sahip olan bir C kodu herhangi bir kod sözcüğünde t-1 tane hatayı tespit edebilmektedir.

Örnek 3.3: Bir önceki örnekteki C = {1101000, 0110100, 0011010, 0001101, 1000110, 0100011, 1010001} kodu için C nin hata düzeltme kapasitesi d(C)=4≥2e+1 olduğundan dolayı e=1 dir. Bu C kodunun herhangi bir kod sözcüğündeki 3 hata tespit edilebilir.

3.1. LİNEER KOD PARAMETRELERİ

Kodları tanımlarken tasarımlarda yaptığımız gibi bazı parametreler kullanmak işimizi kolaylaştırır. Kısaca bu parametreleri vermek istersek, alfabedeki simge sayısı

q, uzunluğu *n*, boyutu (bilgi sembollerinin sayısı) *k* olan bir lineer C kodu kısaca $[n,k,d]_q$ olarak adlandırılır.

Tanım 3.6: C nin elemanlarına kod vektörleri veya kod kelimeleri; bileşenlerine de koordinatları veya konumu denir. Bunlara ilave olarak, kod oranı (oran) $R=k/n$ ve göreceli minimum mesafe $\delta=d/n$ parametreleri de kullanılır.

Tanım 3.7: $\mathbb{F}_q^n$ vektör uzayının herhangi bir $x=(x_1,x_2,...,x_n)$ elemanının ağırlığı $w(x)=\left|\left(i\middle| x_i\neq 0, i=1,2,...,n,\ \ x_i\in\mathbb{F}_q\right)\right|$ şeklinde tanımlanır.

Tanım 3.8: C $\mathbb{F}_q$ üzerinde bir lineer kod ise $w(C)=\min\left\{w(x)\middle| x\neq 0, x\in C\right\}$ ye C kodunun ağırlığı denir.

Önerme 3.1: Her $x,y\in\mathbb{F}_q^n$ için $d(x,y)=w(x-y)$ dir [5].

Teorem 3.1: C $\mathbb{F}_q$ üzerinde n uzunluğunda bir kod ise $d(C)=w(C)$ dir [5].

İspat: $x=(x_1,x_2,...,x_n)$ ve $y=(y_1,y_2,...,y_n)$ olmak üzere $d(C)=\min\left\{d(x,y)\middle| x\neq y,\ x,y\in C\right\}$ olduğundan

$\Rightarrow \exists x,y\in C$ için $d(C)=d(x,y)=w(x-y)$

$\geq \min\left\{w(x)\middle| x_i\neq 0, x\in C\right\}=w(C)$ bulunur.

$d(C)\geq w(C)$ dir.

$w(C)=\min\left\{w(x)\middle| x_i\neq 0, x\in C\right\}$ olduğundan

bir $x\in C$ için $w(C)=w(x)$ bulunur.

$\exists x \in C$ için

$$w(C) = w(x) = w(x-0) = d(x,0) \geq \min\left\{ d(x,y) \middle| x \neq y,\ x, y \in C \right\} = d(C)$$

olduğu görülür.

$w(C) \geq$ d(C) dir.

$w(C) =$ d(C) olur.

Yukarıdaki teoremden m elemanlı lineer bir C kodunun minimum uzaklığını belirlemek için $\binom{m}{2} = \frac{1}{2}.m.(m-1)$ tane karşılaştırma yapmak yerine m-1 tane kod sözcüğünün ağırlığına bakmak yeterli olacaktır sonucu elde edilir.

3.2. ÜRETEN MATRİS VE PARİTE KONTROL MATRİSİ

Bir C kodunun bazı önemlidir çünkü bir koddaki her kelime baz vektörlerinin bir lineer kombinasyonudur ve bazın hiçbir kombinasyonu gereksiz değildir.

Tanım 3.9: C bir $[n,k]_q$ kodu olsun ve C nin bir $\{b_i\}$ bazı belirlensin. G, satırları C nin $\{b_i\}$ baz vektörleri olan $k \times n$ tipinde bir matris olsun. Her $e \in C$ kod vektörü bu satır vektörlerinin bir lineer kombinasyonudur. Bu G matrisine C nin üretici matrisi denir [6].

Örnek 3.4: $\mathbb{F}_2$ üzerindeki $C = \{(0,0,0),(0,1,1),(1,0,1),(1,1,0)\}$ kodunun bir tabanı $S = \{(0,1,1),(1,0,1)\}$ olduğu için $G = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$ matrisi C nin üretici matrisidir. C kodu $\mathbb{F}_2$ üzerinde $[3,2,2]$ kodudur.

Örnek 3.5: $C = \{(0,0,...,0),(1,1,...,1),...,(q-1,q-1,...,q-1)\} \subseteq \mathbb{F}_q^n$, $\mathbb{F}_q$ üzerinde n uzunluğunda bir kod olsun. Bu kodun bir tabanı $S = \{(1,1,1)\}$ olduğundan C, üretici matrisi $G = [1,1,...,1]_{1xn}$ olan bir $[n,1,n]$ koddur.

Tanım 3.10: $u=(u_1,u_2,...,u_n)$ ve $v=(v_1,v_2,...,v_n)\in\mathbb{F}_q^n$ olmak üzere

$\bullet : \mathbb{F}_q^n \times \mathbb{F}_q^n \longrightarrow \mathbb{F}_q$

$(u,v) \to u.v = u_1v_1 + u_2v_2 + ... + u_nv_n$

biçiminde tanımlanan dönüşüme bir iç çarpım denir. $u.v$=0 ise u ve v birbirine diktir denir.

Tanım 3.11: C bir $[n,k,d]_q$ kod olsun. $C^{\perp}=\left\{v\in\mathbb{F}_q^n \middle| u.v=0, \forall u\in C\right\}$ kümesine C nin duali denir.

Teorem 3.2: C bir $[n,k]$ kod ve $G=\begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ & \cdot & & \\ & \cdot & & \\ & \cdot & & \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix}_{kxn}$, C kodunun üretici matrisi olsun. bir $v=(v_1,v_2,...,v_n)\in\mathbb{F}_q^n$ olsun. $v\in C^{\perp}$ olması için gerekli ve yeterli koşul $[v_1v_2...v_n]_{1xn}\,G^T_{nxk}=[00...0]_{1xk}$ olmasıdır.

İspat: $\Rightarrow$: $\forall v=(v_1,v_2,...,v_n)\in\mathbb{F}_q^n$ için $v\in C^{\perp}$ olsun. Bu durumda her $u\in C$ için $v.u$=0

dır ve $G=\begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ & \cdot & & \\ & \cdot & & \\ & \cdot & & \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix}_{kxn}$ ve $G^T=\begin{bmatrix} g_{11} & g_{21} & \cdots & g_{k1} \\ g_{12} & g_{22} & \cdots & g_{k2} \\ & \cdot & & \\ & \cdot & & \\ & \cdot & & \\ g_{1n} & g_{2n} & \cdots & g_{kn} \end{bmatrix}_{nxk}$ dır.

G, C nin üretici matrisi olduğundan

$u_1=(g_{11},g_{12},...,g_{1n}), u_2=(g_{21},g_{22},...,g_{2n}),...,u_k=(g_{k1},g_{k2},...,g_{kn})\in C$ dir.

$v.u_1=0, v.u_2=0,...,v.u_k=0$ olduğundan

$$[v_1 v_2 ... v_n]_{1xn} G^T_{nxk} = [v_1 v_2 ... v_n]_{1xn} \begin{bmatrix} g_{11} & g_{21} & \cdots & g_{k1} \\ g_{12} & g_{22} & \cdots & g_{k2} \\ & \cdot & & \\ & \cdot & & \\ & \cdot & & \\ g_{1n} & g_{2n} & \cdots & g_{kn} \end{bmatrix}_{nxk}$$

$$= [v.u_1 \ v.u_2 \ ... \ v.u_k]_{1xk}$$

$$= [0 \ 0 \ ... \ 0]_{1xk} \text{ olur.}$$

$\Leftarrow : v = (v_1, v_2, ..., v_n) \in \mathbb{F}_q^n$ olmak üzere

$[v_1 v_2 ... v_n]_{1xn} G^T_{nxk} = [0 \ 0 \ ... \ 0]_{1xk}$ iken $v \in C^{\perp}$ olduğunu gösterelim.

$[v_1 v_2 ... v_n]_{1xn} G^T_{nxk} = [0 \ 0 \ ... \ 0]_{1xk}$ olduğundan

$$[v_1 v_2 ... v_n]_{1xn} \begin{bmatrix} g_{11} & g_{21} & \cdots & g_{k1} \\ g_{12} & g_{22} & \cdots & g_{k2} \\ & \cdot & & \\ & \cdot & & \\ & \cdot & & \\ g_{1n} & g_{2n} & \cdots & g_{kn} \end{bmatrix}_{nxk} = [0 \ 0 \ ... \ 0]_{1xk} \text{ dır.}$$

$$\Rightarrow \begin{cases} v_1.g_{11} + v_2.g_{12} + ... + v_n.g_{1n} = 0, \\ v_1.g_{21} + v_2.g_{22} + ... + v_n.g_{2n} = 0, \\ \qquad\qquad \cdot \\ \qquad\qquad \cdot \\ \qquad\qquad \cdot \\ v_1.g_{k1} + v_2.g_{k2} + ... + v_n.g_{kn} = 0, \end{cases} \tag{3.1}$$

dir. G, C kodunun üretici matrisi olduğundan her $u \in C$ için $u = \lambda_1.(g_{11}, g_{12}, ..., g_{1n}) + \lambda_2.(g_{21}, g_{22}, ..., g_{2n}) + ...\lambda_k.(g_{k1}, g_{k2}, ..., g_{kn})$ olacak biçimde $(\lambda_1, \lambda_2, ..., \lambda_k) \in \mathbb{F}_q$ vardır. Eşitliğin her iki tarafı v ile çarpıldığında;

$$u.v = \left(\lambda_1.\left(g_{11}, g_{12}, ..., g_{1n}\right) + \lambda_2.\left(g_{21}, g_{22}, ..., g_{2n}\right) + ...\lambda_k.\left(g_{k1}, g_{k2}, ..., g_{kn}\right)\right).v$$

$$= \left(\left(\lambda_1.g_{11}, \lambda_1.g_{12}, ..., \lambda_1.g_{1n}\right) + \left(\lambda_2.g_{21}, \lambda_2.g_{22}, ..., \lambda_2.g_{2n}\right) + ... + \left(\lambda_k.g_{k1}, \lambda_k.g_{k2}, ..., \lambda_k.g_{kn}\right)\right)\left(v_1, v_2, ..., v_n\right)$$

$$= (\lambda_1.g_{11} + \lambda_2.g_{22} + ... + \lambda_k.g_{k1}, \lambda_1.g_{12} + \lambda_2.g_{22} + ... + \lambda_k.g_{k2}, ..., \lambda_1.g_{1n} + \lambda_2.g_{2n} + ... + \lambda_k.g_{kn})\left(v_1, v_2, ..., v_n\right)$$

$$= (\lambda_1.g_{11} + \lambda_2.g_{22} + ... + \lambda_k.g_{k1})v_1 + (\lambda_1.g_{12} + \lambda_2.g_{22} + ... + \lambda_k.g_{k2})v_2 + ... + (\lambda_1.g_{1n} + \lambda_2.g_{2n} + ... + \lambda_k.g_{kn})v_n$$

$$= \lambda_1.\left(v_1.g_{11}, v_2.g_{12}, ..., v_n.g_{1n}\right) + \lambda_2\left(v_1.g_{21}, v_2.g_{22}, ..., v_n.g_{2n}\right) + ... + \lambda_k\left(v_1.g_{k1}, v_2.g_{k2}, ..., v_n.g_{kn}\right)$$

(3.1) kullanılarak;

$= \lambda_1.0 + \lambda_2.0 + ... + \lambda_k.0 = 0$ bulunur.

$v \in C^{\perp}$ dir.

Önerme 3.2: C, $\mathbb{F}_q$ üzerinde bir lineer $[n,k]$ kod ise $C^{\perp}$ de $\mathbb{F}_q$ üzerinde bir lineer $[n, n-k]$ koddur [3].

Not 3.1: C bir $[n,k]$ kod olmak üzere $\left(C^{\perp}\right)^{\perp} = C$ dir.

Tanım 3.12: C bir $[n,k]$ kod ise $C^{\perp}$ in üretici matrisine parite kontrol matrisi denir ve H ile gösterilir. H $(n-k)xn$ tipinde $G.H^T = 0$ koşulunu sağlayan bir matristir. Bir C lineer $[n,k]$ kodunun parite kontrol matrisi H ise C kodu,

$C = \left\{ x = (x_1, x_2, ..., x_n) \in \mathbb{F}_q^n \middle| \begin{bmatrix} x_1 & x_2 & ... & x_n \end{bmatrix}_{1xn} . H^T_{nx(n-k)} = [0]_{1x(n-k)} \right\}$ biçiminde ifade edilir [6].

Örnek 3.7: $\mathbb{F}_2$ üzerindeki bir *C* kodunun parite kontrol matrisi $H = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}_{2x4}$ şeklinde ise lineer *C* kodu

$C = \left\{ x = (x_1, x_2, x_3, x_4) \in \mathbb{F}_2^4 \middle| \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}_{1x4} . H^T_{4x2} = \begin{bmatrix} 0 & 0 \end{bmatrix}_{1x2} \right\}$ biçiminde belirlenir.

$$\begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix} . H^T = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix} . \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

$\Rightarrow x_1 + x_2 = 0, \quad x_3 + x_4 = 0$ dır. Buradan

$C = \{(0,0,0,0), (1,1,1,1), (0,0,1,1), (1,1,0,0)\} \subseteq \mathbb{F}_2^4$ olduğu görülür. Bu örnekte $C = C^{\perp}$ olduğu kolayca görülür. Dolayısıyla H parite kontrol matrisi *C* kodu için aynı zamanda bir üreten matristir.

Örnek 3.8: $\mathbb{F}_2$ üzerindeki bir *C* kodunun parite kontrol matrisi $H = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}_{1x3}$ biçiminde ise lineer *C* kodu $C = \left\{ x = (x_1, x_2, x_3,) \in \mathbb{F}_2^3 \middle| \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}_{1x3} . H^T_{3x1} = [0]_{1x1} \right\}$ biçiminde belirlenir.

$$\begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}.H^T = \begin{bmatrix} 0 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}.\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix}$$

$\Rightarrow x_1 + x_2 + x_3 = 0$ dır. Buradan $C = \{(0,0,0),(1,0,1),(0,1,1),(1,1,0)\} \subseteq \mathbb{F}_2^3$ olur.

Örnek 3.9: a) (3,1) üçlü tekrar kodu için

Üreten matris

$$\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

Parite Kontrol Matris

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

b) (8,7) parite kontrol kodu için:

Üreten matris

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Parite Kontrol Matris

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Tanım 3.13: C uzunluğu n olan bir lineer kod ve A_i ağırlığı i olan kod kelimelerin sayısı olsun. $A_i = \left|\{e \in C : |e| = i\}\right|$ dir. A_i kümesi C kodunun ağırlık kümesini oluşturur. Yukarıdaki tanımdan farklı olarak A_i ye bağlı $W_C(t) = \sum_i A_i t^{n-i}$ polinomuna C nin ağırlık hesaplayıcısı (polinomu) denir [4].

Bu formül bir kodun üreten matrisinin rank fonksiyonuna bağlı olarak da hesaplanabilir. Bir $[n,k]$ kodunun üreten matrisi G olsun. $I \subset \{1,2,...,n\}$ indeks kümesi, r_I I indeks kümesindeki sütun kümesi ile eşleşen sütun alt matrisinin rankı olmak üzere $W_C(1+t) = \sum_{I \subset \{1,2,...,n\}} q^{k-r_I} t^{|I|}$ denklemine de eşittir [7].

3.3. HAMMING KODLAR

Tanım 3.14: H_k: Kolonları sıfırdan farklı, uzunluğu k olan tüm ikili dizilerden oluşan matris olsun. Hamming kodu, parite kontrol matrisi H_k olan koddur, Ham(k) ile gösterilebilir [7].

Örnek 3.10: Ham(3) kodunun parite kontrol matrisi aşağıdaki gibidir.

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Önerme 3.3: *C*, parite kontrol matrisi H olan bir kod olsun. Öyleyse H deki her *d*-1 sütun lineer bağımsızdır ve lineer bağımlı olan bir *d* sütun kümesi vardır.

3.3.1. Hamming Kod Parametreleri

Ham(*k*), $n=2^k-1$ blok uzunluğuna ve $m= 2^k-k-1$ rankına sahiptir, Ham(*k*) nın minimumum mesafesi $d=3$ tür. Bu yüzden Hamming kodları ancak bir tek hatayı düzeltebilir.

Örnek 3.11: Hamming [7,4,3] kodu 4 uzunluğunda $16=2^4$ tane aşikar sözcükle başlar: 0001, 0010, 0011, 0100, 0101, 0110,… vb. her abcd kelimesi (her bir a,b,c,d 0 veya 1 den oluşmakta) için Hamming [7,4] koduna aşağıdaki şekilde Hamming parite biti ekleyebiliriz.

e=b+c+d

f=a+c+d

g=a+b+d

olacak şekilde abcd, abcdefg haline gelir. Burada, a,b,c,d, $\mathbb{F}_2$ sonlu cismine aittir. [7,4] notasyonu, uzunluğu 4 olan sözcükleri 7 uzunluğundaki kod kelimelerine dönüştürür.

Teorem 3.3: Hamming kodunun ağırlık dağılım polinomu aşağıdaki gibidir.

$$W_{Ham}(z) = 2^{-k}\left[(z+1)^n + n(z+1)^{2^{k-1}-1}.(z-1)^{2^{k-1}}\right]$$

Yardımcı Bilgi: Bir H matrisini ele alalım bu H matrisini üreten matris olarak alan kod R kodu olsun, H matrisini parite kontrol matris olarak ele alan kod Hamming kodudur. Bu yüzden bu iki kod birbirine dual olup dual kodlar arasındaki MacWilliams özdeşliği,

$W_{Ham}(1+z) = q^{-k}.z^n.W_R(1+\frac{q}{z})$ dir.

İspat:

$W_R(z) = 1.z^n + (2^k - 1)z^{n-(2^{k-1})}$ olup,

$$W_{Ham}(1+z) = 2^{-k}.z^n.W_R(1+\frac{2}{z})$$

$$= 2^{-k}.z^n\left[\left(1+\frac{2}{z}\right)^n + \left(2^k-1\right)\left(1+\frac{2}{z}\right)^{n-(2^{k-1})}\right]$$

$$= 2^{-k}\left[(z+2)^n + \left(2^k-1\right)(z+2)^{n-2^{k-1}}.z^{2^{k-1}}\right]$$

$$W_{Ham}(z) = 2^{-k}\left[(z+1)^n + n(z+1)^{2^{k-1}-1}.(z-1)^{2^{k-1}}\right]$$

elde edilir.

BÖLÜM 4

GENELLEŞTİRİLMİŞ KODLAR

Wei ilk olarak bir kodun *r.* genelleştirilmiş Hamming ağırlığını, herhangi *r* boyutlu alt kodlarının minimum destek ağırlığı olarak tanımlamıştır. Ayrıca bir kodun bir kanaldaki performansının, kodun ağırlık hiyerarşisi ile tam olarak karakterize edilebildiğini de göstermiştir. Diğer bir taraftan Kasami genelleştirilmiş Hamming ağırlıklarının çözümlenmeleri için kullanım kolaylığı sağladığını göstermiştir çünkü kodun çapraz karmaşıklığı ve ağırlığı arasında güçlü bir ilişki vardır [8]. Genelleştirilmiş Hamming ağırlıkları günümüze kadar birçok kişi tarafından çalışılmıştır. Hatta bu çalışmalar üst ve alt sınırlara kadar uzanır. Hamming ağırlıkları kodlama teorisinin uygulamalarından da öteye bir kodun yapısıyla ilgili bilgi sağlarlar ve kodları sınıflandırmada kullanılan parametreleri oluştururlar. Genelleştirilmiş ağırlık kavramı destek ağırlık polinomuna kadar uzanır. Destek ağırlık polinomlarını ilk defa Tsfasman ve Vladut ortaya çıkarmıştır [7]. Boyutu ve destek ağırlığı belirli bir kodun alt kodlarının sayısıyla ilgili bilgiyi bu çalışmasında vermiştir. Destek ağırlık polinomları kodun yapısıyla ilgili bilgi elde etmeye, kodların tüm denklik sınıflarının ağırlık dağılımlarını hesaplamak için verilen parametrelerin varlığına karar vermeye yararlar. Ancak çok kısa kodlar için bile destek ağırlık polinomunu bulmak oldukça güçtür. Bugüne kadar bu ağırlık polinomları ile ilgili çok az sonuç elde edilmiştir.

Bu bölümde yeni ancak hızla gelişen alan olan genelleştirilmiş ağırlık tanımları verilmiştir [7].

Bir lineer kodun *r* boyutlu bir *D* alt kodunun destek ağırlığını *D* deki en az bir kod kelimesinde sıfırdan farklı konumların sayısı olarak tanımlayalım. *r.* genelleştirilmiş ağırlık d_r yi, *r* boyutlu alt kodlar üzerindeki destek ağırlıklarının minimumu olarak tanımlarız.

Tanım 4.1: $D \subset \mathbb{F}_q^n$ r boyutlu bir lineer alt uzay olsun. $\mathbb{F}_q^n$ deki vektörler üzerinde tanımlı Hamming ağırlığı (D nin destek ağırlığı)

$$\|D\| = \left|Supp(D)\right|, Supp(D) = \{i : \exists v \in D, v_i \neq 0\}$$

ile tanımlıdır [14]. Hamming ağırlığının alt uzaylar kümesine genelleştirilmiş halidir.

Tanım 4.2: Bir C lineer kodunun r. destek ağırlığı r=1,2,..,k olmak üzere $d_r = d_r(C) = \min\left\{\|D\| \, D \subseteq C, \dim D = r\right\}$ olarak tanımlanır [9]. Açık olarak, $d_1 = d$ kodun minimum mesafesidir.

Bir alt uzayın $\|D\|$ağırlığı ile bu alt uzayın vektörlerinin ağırlıkları arasında basit bir ilişki vardır. Bu ilişki aşağıda belirtilmiştir.

Önerme 4.1: $\dim D = r$ olduğunda,

$$\|D\| = \frac{1}{q^r - q^{r-1}} \sum_{v \in D} \|v\|$$

olur.

Tanım 4.3: C bir lineer $[n,k,d]_q$ kodu olsun. $A_r = A_r(C)$, C'de destek ağırlığı r olan uzayların sayısı olsun destek ağırlık kümesi;

$$A_i^r = \left|\left\{D \subseteq C \middle| \dim D = r, \|D\| = i\right\}\right|$$

dir.

Tanım. 4.4: A_i^r C nin destek ağırlığı i olan r boyutlu alt kodların sayısı olsun. r. destek ağırlık polinomu (hesaplayıcısı)

$$W_C^r(t) = \sum_i A_i^r t^{n-i}$$

homojen polinomudur.

Teorem 4.1: A_i^r C nin destek ağırlığı i olan r boyutlu alt kodların sayısı olsun. r. destek ağırlık polinomu (hesaplayıcısı)

$$W_C^r(t) = \sum_I \begin{bmatrix} k-r_I \\ r \end{bmatrix} (t-1)^{|I|}$$

dir [10].

İspat: A_i^r C nin destek ağırlığı i olan r boyutlu alt uzayların sayısı ve r_I, I üreten matrisinin sütunları ile indekslenmiş alt matrisin rankı olsun. α I daki tüm terimleri sıfır olan kodların tümü olacak şekilde tanımlanırsa, α I daki tüm sütunlara dik olmalıdır. Bu şekilde tanımlanan α lar k-r_I boyutludur.

$\begin{bmatrix} n \\ k \end{bmatrix}$, n boyutlu uzayda k boyutlu alt uzayların sayısı olup, I da sıfırı olan r boyutlu alt uzayların sayısı

$$\begin{bmatrix} k-r_I \\ r \end{bmatrix}$$

dir.

Sadece I da sıfırlanan alt uzayların sayısı, I^C tümleyenin içinde sıfırı olanların çıkarılmasıyla elde edilir ki bu da;

$$\begin{bmatrix} k-r_I \\ r \end{bmatrix} - \sum_{\substack{|I_J|=1 \\ I_J \subseteq I^C}} \begin{bmatrix} k-r_{I\cup I_J} \\ r \end{bmatrix} + \sum_{\substack{|I_J|=2 \\ I_J \subseteq I^C}} \begin{bmatrix} k-r_{I\cup I_J} \\ r \end{bmatrix} - \sum_{\substack{|I_J|=3 \\ I_J \subseteq I^C}} \begin{bmatrix} k-r_{I\cup I_J} \\ r \end{bmatrix} \ldots$$

denklemine eşittir. Bu denklem düzenlenerek;

$$\begin{aligned} W_C^r(t) &= \sum_I \left(\sum_{I_J \subseteq I^c} (-1)^{|I_J|} \begin{bmatrix} k-r_{I\cup I_J} \\ r \end{bmatrix} \right) t^{|I|} \\ &= \sum_I \sum_{J\supset I} (-1)^{|J|-|I|} \begin{bmatrix} k-r_J \\ r \end{bmatrix} t^{|I|} \\ &= \sum_{J\supset I} \sum_I (-1)^{|J|-|I|} \begin{bmatrix} k-r_J \\ r \end{bmatrix} t^{|I|} \\ &= \sum_{J\supset I} \sum_{\substack{|I|=i \\ 0\le i\le j}} (-1)^{|J|-i} \begin{bmatrix} k-r_J \\ r \end{bmatrix} t^i \begin{bmatrix} |J| \\ i \end{bmatrix} \\ &= \sum_J \begin{bmatrix} k-r_J \\ r \end{bmatrix} \cdot \left(\sum_i \begin{bmatrix} |J| \\ i \end{bmatrix} (-1)^{|J|-i} t^i \right) \\ &= \sum_J \begin{bmatrix} k-r_J \\ r \end{bmatrix} (t-1)^{|J|} \end{aligned}$$

elde edilir. Keyfi *I* için destek ağırlık polinomu

$$W_C^r(t) = \sum_I \begin{bmatrix} k-r_I \\ r \end{bmatrix} (t-1)^{|I|} \tag{4.1}$$

dir.

Bir kodun ağırlık polinomunun, üreten matrisin rank fonksiyonuna bağlı olarak elde edilebildiğini önceki bölümde belirtmiştik. G, boyutu *k* olan bir *C* lineer kodunun bir üreten matrisi olsun. *I* indeks kümesi ile sabitlenmiş alt matrisin rankı r_I iken ağırlık polinomu $W_C^r(t) = \sum_I \begin{bmatrix} k-r_I \\ r \end{bmatrix} (t-1)^{|I|}$ dır [11].

Ağırlık dağılımının momentlerinin q^{-r_I} rank fonksiyonlarının bağıntısından elde edilebilir [12]. Bu polinom ve genelleştirilmiş formüller daha önce çalışılmıştır [13].

Bu çalışmada rastsal bir üreten matrisin $\begin{bmatrix} k-r_I \\ r \end{bmatrix}$ rank fonksiyonlarının momentleriyle ilgilenilmiştir. Belirli bir *r* rankı için *kxn* rastsal matrisinin rankının *r* olma olasılığının formülü

$$P(n,k,r) = q^{-kn+\binom{r}{2}} \frac{[n]!}{[n-r]!} \frac{[k]!}{[k-r]!} \frac{1}{[r]!} \tag{4.2}$$

olarak verilmiştir [4].

Çalışmamızda genellikle iki sütun matrisinin verilen ranklara sahip olma olasılığıyla ilgilenilmiştir.. Bu olasılığı elde etmek için problem aşağıdaki forma indirgendi. *I* ve *J* alt matrisler ile eşleşen indeks kümeleri olsun. Sütun alt matrisi rankı m olan $I \cap J$ matrisi ile eşleşsin. *I* sütun kümesi ile eşleşen matrisin rankının olasılık formülü

$$P(\left|I \setminus (I \cap J)\right|, k-m, r_I - m) \tag{4.3}$$

ile verilir [14].

Çalışmamızda elde ettiğimiz formüller için Gauss bağıntıları kullanılmıştır. Bu bağıntılar:

$$\begin{bmatrix} k \\ r \end{bmatrix}_q = \frac{[k]_q!}{[k-r]_q![r]_q!}$$

dır. Burada *q* faktoriyel

$$[k]_q! = \prod_{i=1}^{k} (q^i - 1)$$

çarpımıdır. Bu katsayılar yaygın olarak q hesaplı sonlu cisimler üzerindeki lineer cebirdeki kombinasyonlar da kullanılır. Örneğin $\begin{bmatrix} k \\ r \end{bmatrix}_q$ katsayısı $\mathbb{F}_q$ üzerinde k boyutlu bir lineer uzayın r boyutlu alt uzaylarının sayısını verir. Aşağıdaki notasyonlarda q yu kullanmayacağız. 5. Bölümde sıklıkla kullanacağımız özdeşlikler aşağıda verilmiştir.

$$\frac{[k]!}{[k-d]!} = q^{kd-\binom{d}{2}} \prod_{i=0}^{d-1} \left(1-q^{-k+i}\right) \tag{4.4}$$

$$\begin{bmatrix} m+n \\ k \end{bmatrix} = \sum_{j=0}^{k} q^{(k-j)(m-j)} \begin{bmatrix} m \\ j \end{bmatrix} \begin{bmatrix} n \\ k-j \end{bmatrix} \tag{4.5}$$

$$x^n = \sum_{i=0}^{n} \begin{bmatrix} n \\ i \end{bmatrix} \prod_{j=0}^{i-1} (x-q^j) \tag{4.6}$$

Bu bağıntılar V. Kac'ın çalışmasında yer almaktadır [13].

BÖLÜM 5

ÇALIŞMALAR VE SONUÇLAR

5.1. YÜKSEK RANK FONKSİYONLARININ MOMENTLERİ

Bu bölümde yüksek rank fonksiyonları arasındaki beklenti ve kovaryanslar elde edilmiştir.

Teorem 5.1: $r_{n,k}$, $r \le k \le n$ olmak üzere rastsal bir *kxn* lik matrisin rankı olsun. Bu rank fonksiyonunun beklentisi $\mathbb{E}$

$$\mathbb{E}\left(\begin{bmatrix} k-r_{n,k} \\ r \end{bmatrix}\right) = q^{-rn}\begin{bmatrix} k \\ r \end{bmatrix} \tag{5.1}$$

dir [15].

İspat: Beklenti formülü aşağıdaki gibidir.

$$\mathbb{E}\left(\begin{bmatrix} k-r_{n,k} \\ r \end{bmatrix}\right) = \sum_{s} P(n,k,s)\begin{bmatrix} k-s \\ r \end{bmatrix}$$

(4.2) eşitliğindeki $P(n,k,s)$ yi formülde yerine koyarsak

$$\mathbb{E}\left(\begin{bmatrix} k-r_{n,k} \\ r \end{bmatrix}\right) = q^{-kn}\begin{bmatrix} k \\ r \end{bmatrix}\sum_{s} q^{\binom{s}{2}}\begin{bmatrix} k-r \\ s \end{bmatrix}\frac{[n]!}{[n-s]!}$$

elde edilir. Burada (4.4) özelliğini kullanarak

$$\mathbb{E}\left(\begin{bmatrix} k-r_{n,k} \\ r \end{bmatrix}\right) = q^{-kn}\begin{bmatrix} k \\ r \end{bmatrix}\sum_{s}\begin{bmatrix} k-r \\ s \end{bmatrix} q^{sn}\prod_{i=0}^{s-1}(1-q^{-n+i})$$

elde edilir. (4.6) formülünden *s* üzerinden toplam $q^{n(k-r)}$ ye eşittir teorem ispatlanır. Yukarıdaki formülde r yerine 1 koyarsak [14] deki sonuç elde edilir.

$$\mathbb{E}\left(q^{-r_{n,k}}\right) = q^{-k} + q^{-n} - q^{-n-k}$$

Aynı çalışmada bu beklentiden kod ağırlıklarının birinci momentlerinin elde edilebileceği gösterilmiştir.

Aşağıda yüksek rank fonksiyonları arasındaki bağıntılar elde edilmiştir. Kodlama teorisi açısından sütun alt matrislerinin ranklarını hesaplamak yeterlidir. *I* indeks kümesindeki sütunlar ile eşleşen alt matrisin rankı kolaylık sağlaması açısından r_I ile gösterilecektir.

Teorem 5.2: *I* ve *J* rastsal bir matrisin sütunlarının iki altkümesi olsunlar. *I* ve *J* ye karşılık gelen rank fonksiyonlarının kovaryansları aşağıdaki gibidir [15].

$$\operatorname{cov}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}, \begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) = q^{-r(|I|+|J|)}\begin{bmatrix} k \\ r \end{bmatrix}\sum_{i=0}^{r} q^{i^2}\begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix}(q^{|I\cap J|(r-i)}-1) \qquad (5.2)$$

İspat: Bu teoremin ispatında *I* ve *J* kümelerinin kardinalitelerini belirtmek için aynı sembolleri kullanacağız. Notasyonu kolaylaştırmak için $I \cap J$ yerine *IJ*, $I \setminus (I \cap J)$ yerine $\bar{I}$, $J \setminus (I \cap J)$ yerine de $\bar{J}$ sembollerini kullanacağız. Kovaryans aşağıdaki gibidir:

$$\operatorname{cov}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}, \begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) = \mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) - \mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\right)\mathbb{E}\left(\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) \qquad (5.3)$$

Bağlı beklenti aşağıdaki toplamla ifade edilmiştir:

$$\sum_{s,t,m} P(IJ,\ k,\ m)P(\bar{I},\ k-m,\ s-m)P(\bar{J},\ k-m,\ t-m)\begin{bmatrix} k-s \\ r \end{bmatrix}\begin{bmatrix} k-t \\ r \end{bmatrix}$$

Burada r_I ve r_J rankları s ve t olarak sabitlenmiştir. $r_I=s$ ve $r_J=t$ olasılığını elde etmek için $I \cap J$ nin satırları ile oluşmuş matrisin rankı üzerinden bir toplam oluşturacağız. $P(IJ,\ k,\ m)$ olasılığını $r_{IJ}=m$ şeklinde sabitleyelim. Öyleyse $r_I=s$ ve $r_J=t$ olma olasılığı, bu $P(IJ,\ k,\ m)$ olasılığının $P(\bar{I},\ k-m,\ s-m)P(\bar{J},\ k-m,\ t-m)$ çarpımı ile çarpılmasıyla elde edilir.

s üzerinden toplam alarak bağlı beklentiyi hesaplayacağız. Bu toplam aşağıdadır.

$$\sum_{s} P(\bar{I},\ k-m,\ s-m)\begin{bmatrix} k-s \\ r \end{bmatrix} = \sum_{s} q^{-\bar{I}(k-m)+\binom{s-m}{2}}\begin{bmatrix} \bar{I} \\ s-m \end{bmatrix}\frac{[k-m]!}{[k-s]!}\begin{bmatrix} k-s \\ r \end{bmatrix}$$

Bir değişken değişimiyle $u=s-m$ olarak alalım ve Gauss bağıntısını tekrar düzenleyelim:

$$\sum_{s} P(\bar{I},\ k-m,\ s-m)\begin{bmatrix} k-s \\ r \end{bmatrix} = \sum_{u} q^{-\bar{I}(k-m)+\binom{u}{2}}\begin{bmatrix} \bar{I} \\ u \end{bmatrix}\begin{bmatrix} k-m \\ r \end{bmatrix}\frac{[k-m-r]!}{[k-m-r-u]!}$$

Sağdaki kesir bir çarpım haline getirilebilir ve toplamın yeni şekli (4.4) formülünden:

$$\sum_{u} q^{-\bar{I}(k-m)+\binom{u}{2}}\begin{bmatrix} \bar{I} \\ u \end{bmatrix}\begin{bmatrix} k-m \\ r \end{bmatrix} q^{-\binom{u}{2}}\prod_{j=0}^{u-1}(q^{k-m-r}-q^{j})$$

olur. (4.6) formülünden

$$\sum_{s} P(\bar{I},\ k-m,\ s-m)\begin{bmatrix} k-s \\ r \end{bmatrix} = q^{-r\bar{I}}\begin{bmatrix} k-m \\ r \end{bmatrix}$$

elde edilir. *t* parametreleri üzerinden toplam da benzer şekilde elde edilir. Böylece bağıl beklenti

$$\mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) = \sum_m P(IJ,\ k,\ m) q^{-r(\bar{I}+\bar{J})} \begin{bmatrix} k-m \\ r \end{bmatrix}^2$$

formülüne indirgenir. Tüm terimleri açıkça yazacak olursak, bağıl beklenti şu toplamla verilir:

$$\begin{aligned}
\mathbb{E} &= q^{-kIJ-r(\bar{I}+\bar{J})} \sum_m q^{\binom{m}{2}} \frac{[IJ]![k]!}{[IJ-m]![k-m]![m]!} \left(\frac{[k-m]!}{[k-m-r]![r]!}\right)^2 \\
&= q^{-kIJ-r(\bar{I}+\bar{J})} \sum_m \left(\prod_{j=0}^{m-1}(q^{IJ}-q^j)\right) \begin{bmatrix} k \\ r \end{bmatrix}\begin{bmatrix} k-r \\ m \end{bmatrix} \frac{[k-m]!}{[k-r-m]![r]!} \\
&\underset{m\to k-r-m}{=} q^{-kIJ-r(\bar{I}+\bar{J})} \sum_m \left(\prod_{j=0}^{k-r-m-1}(q^{IJ}-q^j)\right) \begin{bmatrix} k \\ r \end{bmatrix}\begin{bmatrix} k-r \\ m \end{bmatrix}\begin{bmatrix} r+m \\ r \end{bmatrix}
\end{aligned}$$

(4.5) den elde edilebilen aşağıdaki özdeşliği yerine koyarsak,

$$\begin{bmatrix} r+m \\ r \end{bmatrix} = \sum_{i=0}^{r} q^{i^2} \begin{bmatrix} r \\ i \end{bmatrix}\begin{bmatrix} m \\ i \end{bmatrix} \tag{5.4}$$

$$\begin{aligned}
\mathbb{E} &= q^{-kIJ-r(\bar{I}+\bar{J})} \sum_{m,i} \left(\prod_{j=0}^{k-r-m-1}(q^{IJ}-q^j)\right) \begin{bmatrix} k \\ r \end{bmatrix}\begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix} q^{i^2} \frac{[k-r-i]!}{[k-r-m]![m-i]!} \\
&\underset{v:=m-i}{=} q^{-kIJ-r(\bar{I}+\bar{J})} \begin{bmatrix} k \\ r \end{bmatrix} \sum_{v,i} \left(\prod_{j=0}^{k-r-v-i-1}(q^{IJ}-q^j)\right) \begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix} q^{i^2} \begin{bmatrix} k-r-i \\ v \end{bmatrix} \\
&\underset{v\to k-r-i-v}{=} q^{-kIJ-r(\bar{I}+\bar{J})} \begin{bmatrix} k \\ r \end{bmatrix} \sum_{v,i} \left(\prod_{j=0}^{v-1}(q^{IJ}-q^j)\right) \begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix} q^{i^2} \begin{bmatrix} k-r-i \\ v \end{bmatrix} \\
&= q^{-kIJ-r(\bar{I}+\bar{J})} \begin{bmatrix} k \\ r \end{bmatrix} \sum_i q^{IJ(k-r-i)} \begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix} q^{i^2}
\end{aligned}$$

$$\mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) = q^{-r(I+J)}\begin{bmatrix} k \\ r \end{bmatrix}\sum_{i=0}^{r} q^{i^2}\begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix} q^{IJ(r-i)} \tag{5.5}$$

Tekrar (4.5) özdeşliğini kullanılarak

$$\begin{bmatrix} k \\ r \end{bmatrix} = \begin{bmatrix} (k-r)+(r) \\ r \end{bmatrix} = \sum_{i=0}^{r} q^{i^2}\begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix}$$

elde edilir. Bir rank fonksiyonunun beklentisinden

$$\mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\right)\mathbb{E}\left(\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) = q^{-r(I+J)}\begin{bmatrix} k \\ r \end{bmatrix}^2$$

$$= q^{-r(I+J)}\begin{bmatrix} k \\ r \end{bmatrix}\sum_{i=0}^{r} q^{i^2}\begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix} \tag{5.6}$$

elde edilir. (5.5) ve (5.6) eşitlikleri bize sonucu verir.

5.2. RASTSAL KODLARIN DESTEK AĞIRLIKLARININ MOMENTLERİ

Teorem 5.3: Destek ağırlık polinomu $W_C^r(z)$ nin beklentisi

$$\mathbb{E}W_C^r(z) = \begin{bmatrix} k \\ r \end{bmatrix}\left(1+q^{-r}(z-1)\right)^n \tag{5.7}$$

dir [15].

İspat: Destek ağırlık polinomu aşağıdaki gibidir,

$$W_C^r(1+z) = \sum_{I}\begin{bmatrix} k-r_I \\ r \end{bmatrix} z^{|I|}$$

z yerine *z-1* yazıp her iki tarafın beklentisini alırsak basit bir toplam işlemiyle aşağıdaki sonuç elde edilir.

$$\mathbb{E}W_C^r(z) = \sum_I \mathbb{E}\begin{bmatrix} k-r_I \\ r \end{bmatrix}(z-1)^{|I|}$$

$$= \sum_{i=0}^{n}\binom{n}{i} q^{-ri}\begin{bmatrix} k \\ r \end{bmatrix}(z-1)^i$$

$$= \begin{bmatrix} k \\ r \end{bmatrix}\left(1+q^{-r}(z-1)\right)^n$$

Benzer şekilde ağırlık polinomunun beklentisi aşağıdaki şekilde hesaplanmıştır [12].

$$\mathbb{E}W_C(z) = z^n + \frac{\left(q^k-1\right)}{q^n}(z+q-1)^n$$

Teorem 5.4: Bir C $[n,k]$ kodunun destek ağırlıkları arasındaki kovaryans bağıntısı

$$\operatorname{cov}\left(W_C^r(x),\ W_C^r(y)\right) =$$

$$\begin{bmatrix} k \\ r \end{bmatrix} q^{-rn}\sum_{i=0}^{r} q^{i(i-n)}\begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix}\left(xy+\left(q^i-1\right)\left(x+y\right)+q^{i+r}-2q^i+1\right)^n$$

$$-\begin{bmatrix} k \\ r \end{bmatrix}^2 q^{-2nr}\left(x+q^r-1\right)^n\left(y+q^r-1\right)^n \quad \text{dir [15].} \tag{5.8}$$

İspat: Aşağıdaki eşitliği açarak kovaryansları elde edeceğiz.

$$\operatorname{cov}\left(\sum_I \begin{bmatrix} k-r_I \\ r \end{bmatrix} x^{|I|},\ \sum_J \begin{bmatrix} k-r_J \\ r \end{bmatrix} y^{|J|}\right). \tag{5.9}$$

S_i , $W_C^r(1+x)$ deki x_i lerin katsayısı olsun yani,

$$\mathrm{S}_i = \sum_{|I|=i}\begin{bmatrix} k-r_I \\ r \end{bmatrix} \quad \text{dir.}$$

Bu iki eşitlikten yararlanarak,

$$\operatorname{cov}\left(W_C^r(1+x),\ W_C^r(1+y)\right)=\sum_{i,j}\operatorname{cov}\left(S_i,\ S_j\right)\mathrm{x}^i\mathrm{y}^i. \tag{5.10}$$

elde edilir. Bu kovaryansı elde etmek için $I\cap J$ kesişiminin kardinalitesini sabitleyelim ve $I\cup J$ kümesini parçalayalım. Son olarak *m* parametresi üzerinde toplam alırsak,

$$\operatorname{cov}\left(S_i,\ S_j\right)=$$

$$=\operatorname{cov}\left(\sum_{|I|=i}\begin{bmatrix}k-r_I\\ r\end{bmatrix},\ \sum_{|J|=j}\begin{bmatrix}k-r_J\\ r\end{bmatrix}\right)$$

$$=\sum_{|I|=i,\ |J|=j}\operatorname{cov}\left(\begin{bmatrix}k-r_I\\ r\end{bmatrix},\ \begin{bmatrix}k-r_J\\ r\end{bmatrix}\right)$$

$$=q^{-r(i+j)}\begin{bmatrix}k\\ r\end{bmatrix}\sum_{|I|=i,\ |J|=j}\sum_{m=0}^{r}q^{m^2}\begin{bmatrix}k-r\\ m\end{bmatrix}\begin{bmatrix}r\\ m\end{bmatrix}\left(q^{IJ(r-m)}-1\right)$$

$$=q^{-r(i+j)}\begin{bmatrix}k\\ r\end{bmatrix}\sum_{m=0}^{r}q^{m^2}\begin{bmatrix}k-r\\ m\end{bmatrix}\begin{bmatrix}r\\ m\end{bmatrix}$$

$$\times\left(\sum_{t}\binom{n}{ti-tj-tn-i-j+t}\left(q^{t(r-m)}-1\right)\right) \tag{5.11}$$

Son eşitlikteki iç çarpım x^iy^i nin katsayısıdır ve

$$\left(\left(q^{r-m}xy+x+y+1\right)^n-\left(x+1\right)^n\left(y+1\right)^n\right)$$

denklemine eşittir.

i ve *j* üzerinden toplam alınıp (4.5) özdeşliğinden yararlanılarak

$$\operatorname{cov}\left(W_C^r(1+x),\ W_C^r(1+y)\right)=\begin{bmatrix}k\\ r\end{bmatrix}q^{-rn}\sum_{m=0}^{r}q^{m^2}\begin{bmatrix}k-r\\ m\end{bmatrix}\begin{bmatrix}r\\ m\end{bmatrix}\left(q^{-m}xy+x+y+q^r\right)^n$$

$$-\begin{bmatrix}k\\ r\end{bmatrix}^2 q^{-2rn}\left(x+q^r\right)^n\left(y+q^r\right)^n$$

elde edilir. Sırasıyla *x* yerine *x-1* ve *y* yerine *y-1* konulursa

$$\operatorname{cov}\left(W_C^r(x),\ W_C^r(y)\right)=\begin{bmatrix}k\\ r\end{bmatrix}q^{-rn}\sum_{i=0}^{r}q^{i(i-n)}\begin{bmatrix}k-r\\ i\end{bmatrix}\begin{bmatrix}r\\ i\end{bmatrix}\left(xy+\left(q^i-1\right)\left(x+y\right)+q^{i+r}-2q^i+1\right)^n$$

$$-\begin{bmatrix}k\\ r\end{bmatrix}^2 q^{-2nr}\left(x+q^r-1\right)^n\left(y+q^r-1\right)^n \quad \text{elde edilir.}$$

BÖLÜM 6

MAPLE DENEMELERİ

Bu bölümde destek ağırlıklarının istatistiklerinin, Maple programı yardımıyla da elde edilebildiğini gösteren örnekler verilmiştir.

kxn matrisin rankının *r* olma olasılığı;

```
> #unassign 'r',r;
> P:=(n,k,r)->
q^(-(n-r)*(k-r))*mul((1-q^(i-k))*(1-q^(i-n))/(1-q^(i-r)),i=0..r-1);
```

$$P := (n, k, r) \to q^{-(n-r)\,(k-r)}\, mul\left(\frac{\left(1-q^{i-k}\right)\left(1-q^{i-n}\right)}{1-q^{i-r}}, i = 0..r-1\right)$$

```
>add(q^(-(n-r[I])*(k-r[I]))*mul((1-q^(i-k))*(1-q^(i-n))/(1-q^(i-r[I])),i=0..r-
1)*mul((q^i-1),i=1..k-r[I])/(((q^i-1),i=1..k-r[I]-r)*((q^i-1),i=1..r)));
```

```
n:=
k:=
r:=
```

> n:=9:

k:=7:

r:=5:

> simplify(add(P(n,k,i),i=0..min(k,n)));

$$1$$

Aşağıdaki E formülü $\begin{bmatrix} k-r_I \\ r \end{bmatrix}_q$ q binomun beklentisidir.

E:=(n,k,r)->add(P(n,k,rI)*mul((q^i-1),i=1..k-rI)/(mul((q^i-1),i=1..k-rI-r)*mul((q^i-1),i=1..r)),rI=0..min(k,n,k-r));

$$E := (n, k, r) \rightarrow add\left(\frac{P(n,k,rI)\ mul\left(q^i - 1, i = 1..k - rI\right)}{mul\left(q^i - 1, i = 1..k - rI - r\right)\ mul\left(q^i - 1, i = 1..r\right)}, rI = 0..\min(k, n, k - r) \right)$$

> for i from 0 to 15

do print('n',n,'k',k,'r',r+i):

sort(factor(E(n,k,r+i)));

end do;

$$n, n, k, k, r, r$$

```
> n:=9: k:=7: r:=5:
for i from 0 to 15
do print('n',n,'k',k,'r',r+i):
sort(factor(E(n,k,r+i)));
end do;
```

$$n, 9, k, 7, r, 5$$

$$\frac{\left(q^2+q+1\right)\left(q^2-q+1\right)\left(q^6+q^5+q^4+q^3+q^2+q+1\right)}{q^{45}}$$

$$n, 9, k, 7, r, 6$$

$$\frac{q^6+q^5+q^4+q^3+q^2+q+1}{q^{54}}$$

$$n, 9, k, 7, r, 7$$

$$\frac{1}{q^{63}}$$

$$n, 9, k, 7, r, 8$$

$$0$$

$$n, 9, k, 7, r, 9$$

$$0$$

$$n, 9, k, 7, r, 10$$

$$0$$

$$n, 9, k, 7, r, 11$$

$$0$$

$$n, 9, k, 7, r, 12$$

$$0$$

$n, 9, k, 7, r, 13$
0

$n, 9, k, 7, r, 14$
0

$n, 9, k, 7, r, 15$
0

$n, 9, k, 7, r, 16$
0

$n, 9, k, 7, r, 17$
0

$n, 9, k, 7, r, 18$
0

$n, 9, k, 7, r, 19$
0

$n, 9, k, 7, r, 20$
0

```
> n:=9: k:=7: r:=3:
for i from 0 to 15
do print('n',n,'k',k,'r',r+i):
sort(factor(E(n,k,r+i)));
end do;
```

$$n, 9, k, 7, r, 3$$

$$\frac{1}{q^{27}}\left(\left(q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^2-q+1\right)\left(q^4+q^3+q^2+q+1\right)\right)$$

$$n, 9, k, 7, r, 4$$

$$\frac{1}{q^{36}}\left(\left(q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^2-q+1\right)\left(q^4+q^3+q^2+q+1\right)\right)$$

$$n, 9, k, 7, r, 5$$

$$\frac{\left(q^2+q+1\right)\left(q^2-q+1\right)\left(q^6+q^5+q^4+q^3+q^2+q+1\right)}{q^{45}}$$

$$n, 9, k, 7, r, 6$$

$$\frac{q^6+q^5+q^4+q^3+q^2+q+1}{q^{54}}$$

$$n, 9, k, 7, r, 7$$

$$\frac{1}{q^{63}}$$

$$n, 9, k, 7, r, 8$$

$$0$$

$n, 9, k, 7, r, 9$
0

$n, 9, k, 7, r, 10$
0

$n, 9, k, 7, r, 11$
0

$n, 9, k, 7, r, 12$
0

$n, 9, k, 7, r, 13$
0

$n, 9, k, 7, r, 14$
0

$n, 9, k, 7, r, 15$
0

$n, 9, k, 7, r, 16$
0

$n, 9, k, 7, r, 17$
0

$n, 9, k, 7, r, 18$
0

```
> n:=23: k:=19: r:=9:
> for i from 0 to 15
do print('n',n,'k',k,'r',r+i):
sort(factor(E(n,k,r+i)));
end do;
```

$$n, 23, k, 19, r, 9$$

$$\frac{1}{q^{207}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{6}-q^{3}+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{8}+1\right)\left(q^{8}-q^{7}+q^{5}-q^{4}+q^{3}-q+1\right)\left(q^{6}-q^{5}+q^{4}-q^{3}+q^{2}-q+1\right)\left(q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{2}-q+1\right)\left(q^{4}-q^{2}+1\right)\left(q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\right)$$

$$n, 23, k, 19, r, 10$$

$$\frac{1}{q^{230}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{6}-q^{3}+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{8}+1\right)\left(q^{8}-q^{7}+q^{5}-q^{4}+q^{3}-q+1\right)\left(q^{6}-q^{5}+q^{4}-q^{3}+q^{2}-q+1\right)\left(q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{2}-q+1\right)\left(q^{4}-q^{2}+1\right)\left(q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\right)$$

$$n, 23, k, 19, r, 11$$

$$\frac{1}{q^{253}}\left(\left(q^2-q+1\right)\left(q^4-q^2+1\right)\left(q^{12}+q^{11}+q^{10}+q^9+q^8\right.\right.$$
$$+q^7+q^6+q^5+q^4+q^3+q^2+q+1\Big)\left(q^6-q^5+q^4-q^{3}\right.$$
$$+q^2-q+1\Big)\left(q^8-q^7+q^5-q^4+q^3-q+1\right)\left(q^8\right.$$
$$+1\Big)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8\right.$$
$$+q^7+q^6+q^5+q^4+q^3+q^2+q+1\Big)\left(q^2+q+1\right)\left(q^6\right.$$
$$+q^3+1\Big)\left(q^6-q^3+1\right)\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}\right.$$
$$+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2$$
$$\left.\left.+q+1\right)\right)$$

$n, 23, k, 19, r, 12$

$$\frac{1}{q^{276}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.\right.$$
$$+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\Big)\left(q^6+q^{3}\right.$$
$$+1\Big)\left(q^6-q^3+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}\right.$$
$$+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\Big)$$
$$\left(q^4+1\right)\left(q^8+1\right)\left(q^8-q^7+q^5-q^4+q^3-q+1\right)\left(q^6\right.$$
$$-q^5+q^4-q^3+q^2-q+1\Big)\left(q^{12}+q^{11}+q^{10}+q^9+q^8\right.$$
$$\left.\left.+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\right)$$

$n, 23, k, 19, r, 13$

$$\frac{1}{q^{299}}\left(\left(q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^6-q^5+q^4-q^3+q^2-q+1\right)\left(q^8-q^7+q^5-q^4+q^3-q+1\right)\left(q^4+1\right)\left(q^8+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^6+q^3+1\right)\left(q^6-q^3+1\right)\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\right)$$

$n, 23, k, 19, r, 14$

$$\frac{1}{q^{322}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^2+q+1\right)\left(q^2-q+1\right)\left(q^6+q^3+1\right)\left(q^6-q^3+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^4+1\right)\left(q^8+1\right)\left(q^8-q^7+q^5-q^4+q^3-q+1\right)\right)$$

$n, 23, k, 19, r, 15$

$$\frac{1}{q^{345}}\left(\left(q^4+1\right)\left(q^8+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^2-q+1\right)\left(q^6+q^3+1\right)\left(q^6-q^3+1\right)\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\right)$$

$n, 23, k, 19, r, 16$

$$\frac{1}{q^{368}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.\right.$$
$$+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\Big)\left(q^{2}-q\right.$$
$$+1\Big)\left(q^{6}+q^{3}+1\right)\left(q^{6}-q^{3}+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}\right.$$
$$+q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}$$
$$\left.\left.+q+1\right)\right)$$

$n, 23, k, 19, r, 17$

$$\frac{1}{q^{391}}\left(\left(q^{2}+q+1\right)\left(q^{2}-q+1\right)\left(q^{6}+q^{3}+1\right)\left(q^{6}-q^{3}\right.\right.$$
$$+1\Big)\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.$$
$$\left.\left.+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\right)$$

$n, 23, k, 19, r, 18$

$$\frac{1}{q^{414}}\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^{9}\right.$$
$$\left.+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)$$

$n, 23, k, 19, r, 19$

$$\frac{1}{q^{437}}$$

$n, 23, k, 19, r, 20$

0

$n, 23, k, 19, r, 21$

0

$n, 23, k, 19, r, 22$

0

$n, 23, k, 19, r, 23$

0

$n, 23, k, 19, r, 24$

0

```
> n:=23: k:=19: r:=11:
for i from 0 to 15
do print('n',n,'k',k,'r',r+i):
sort(factor(E(n,k,r+i)));
end do;
```

$n, 23, k, 19, r, 11$

$$\frac{1}{q^{253}}\left(\left(q^2-q+1\right)\left(q^4-q^2+1\right)\left(q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^6-q^5+q^4-q^3+q^2-q+1\right)\left(q^8-q^7+q^5-q^4+q^3-q+1\right)\left(q^8+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\left(q^2+q+1\right)\left(q^6+q^3+1\right)\left(q^6-q^3+1\right)\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\right)$$

$n, 23, k, 19, r, 12$

$$\frac{1}{q^{276}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.\right.$$
$$+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\big)\left(q^{6}+q^{3}\right.$$
$$+1\big)\left(q^{6}-q^{3}+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}\right.$$
$$+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\big)$$
$$\left(q^{4}+1\right)\left(q^{8}+1\right)\left(q^{8}-q^{7}+q^{5}-q^{4}+q^{3}-q+1\right)\left(q^{6}\right.$$
$$\left.-q^{5}+q^{4}-q^{3}+q^{2}-q+1\right)\left(q^{12}+q^{11}+q^{10}+q^{9}+q^{8}\right.$$
$$\left.\left.+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\right)$$

$n, 23, k, 19, r, 13$

$$\frac{1}{q^{299}}\left(\left(q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{6}-q^{5}+q^{4}-q^{3}\right.\right.$$
$$+q^{2}-q+1\big)\left(q^{8}-q^{7}+q^{5}-q^{4}+q^{3}-q+1\right)\left(q^{4}\right.$$
$$+1\big)\left(q^{8}+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.$$
$$+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\big)\left(q^{6}+q^{3}\right.$$
$$+1\big)\left(q^{6}-q^{3}+1\right)\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}\right.$$
$$\left.\left.+q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}\right.\right.$$
$$\left.\left.+q+1\right)\right)$$

$n, 23, k, 19, r, 14$

$$\frac{1}{q^{322}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.\right.$$
$$+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}+q^{4}+q^{3}+q^{2}+q+1\big)\left(q^{2}+q\right.$$
$$+1\big)\left(q^{2}-q+1\right)\left(q^{6}+q^{3}+1\right)\left(q^{6}-q^{3}+1\right)\left(q^{16}+q^{15}\right.$$
$$+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^{9}+q^{8}+q^{7}+q^{6}+q^{5}$$
$$\left.+q^{4}+q^{3}+q^{2}+q+1\right)\left(q^{4}+1\right)\left(q^{8}+1\right)\left(q^{8}-q^{7}+q^{5}\right.$$
$$\left.\left.-q^{4}+q^{3}-q+1\right)\right)$$

$n, 23, k, 19, r, 15$

$$\frac{1}{q^{345}}\left(\left(q^4+1\right)\left(q^8+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}\right.\right.$$
$$+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\Big)$$
$$\left(q^2-q+1\right)\left(q^6+q^3+1\right)\left(q^6-q^3+1\right)\left(q^{18}+q^{17}+q^{16}\right.$$
$$+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6$$
$$\left.\left.+q^5+q^4+q^3+q^2+q+1\right)\right)$$

$n, 23, k, 19, r, 16$

$$\frac{1}{q^{368}}\left(\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.\right.$$
$$+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\Big)\left(q^2-q\right.$$
$$\left.+1\right)\left(q^6+q^3+1\right)\left(q^6-q^3+1\right)\left(q^{16}+q^{15}+q^{14}+q^{13}\right.$$
$$+q^{12}+q^{11}+q^{10}+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2$$
$$\left.\left.+q+1\right)\right)$$

$n, 23, k, 19, r, 17$

$$\frac{1}{q^{391}}\left(\left(q^2+q+1\right)\left(q^2-q+1\right)\left(q^6+q^3+1\right)\left(q^6-q^3\right.\right.$$
$$\left.+1\right)\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}\right.$$
$$\left.\left.+q^9+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)\right)$$

$n, 23, k, 19, r, 18$

$$\frac{1}{q^{414}}\left(q^{18}+q^{17}+q^{16}+q^{15}+q^{14}+q^{13}+q^{12}+q^{11}+q^{10}+q^9\right.$$
$$\left.+q^8+q^7+q^6+q^5+q^4+q^3+q^2+q+1\right)$$

$n, 23, k, 19, r, 19$

$$\frac{1}{q^{437}}$$

$n, 23, k, 19, r, 20$

$$0$$

$n, 23, k, 19, r, 21$

$$0$$

$n, 23, k, 19, r, 22$

$$0$$

$n, 23, k, 19, r, 23$

$$0$$

$n, 23, k, 19, r, 24$

$$0$$

$n, 23, k, 19, r, 25$

$$0$$

$n, 23, k, 19, r, 26$

$$0$$

gibi sonuçların maple kullanılarak karşılaştırılmasıyla da; $r_{n,k}$, $r \le k \le n$ olmak üzere rastsal bir *kxn* lik matrisin rankı r olan bu rank fonksiyonunun beklentisinin $\mathbb{E}$

$$\mathbb{E}\left(\begin{bmatrix} k-r_{n,k} \\ r \end{bmatrix}\right) = q^{-rn}\begin{bmatrix} k \\ r \end{bmatrix}$$

olduğu görülmüştür. Benzer şekilde; $W_C^r(1+z) = \sum_I \begin{bmatrix} k-r_I \\ r \end{bmatrix} z^{|I|}$ eşitliğinden yararlanarak her iki tarafın beklentisini alıp z yerine z-1 koyarak ve yukarıda verilen

bir *kxn* matrisin rankının r olma beklentisinin formülü kullanılarak bölüm 4 te de ispatlandığı gibi $\mathbb{E}W_C^r(z) = \begin{bmatrix} k \\ r \end{bmatrix}\left(1+q^{-r}(z-1)\right)^n$ formülü elde edilmiştir.

$$\operatorname{cov}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}, \begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) = \mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) - \mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\right)\mathbb{E}\left(\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right)$$

Eşitliğini hesaplamak için $\mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\right)$ bilindiğinden $\mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right)$ yi hesaplamak yeterli olacaktır.

k: satırların sayısı
II: *I* matrisi
JJ: *J* matrisi
IJ: *I* ve *J* nin arakesiti
II_IJ: *I* fark *I* arakesit *J*
JJ_IJ: *J* fark *I* arakesit *J*
s: *I* nın rankı
t: *J* nin rankı
m: *I* arakesit *J* nin rankı

olmak üzere EE:= $\operatorname{cov}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}, \begin{bmatrix} k-r_J \\ r \end{bmatrix}\right)$ formülü

```
>EE:=(II,JJ,IJ,k,r)->add(add(add(P(IJ,k,m)*P(II-IJ,k-m,s-m)*P(JJ-IJ,k-m,t-
m)*mul(q^i-1,i=1..k-s)*mul(q^i-1,i=1..k-t)/(mul(q^i-1,i=1..k-s-r)*mul(q^i-
1,i=1..r)^2*mul(q^i-1,i=1..k-t-r)),m=0..min(k,IJ,s,t)),t=0..JJ),s=0..II);#
```

$$EE := (II, JJ, IJ, k, r) \to add\Big(add\Big(add\Big(\big(P(IJ, k, m)\ P(II - IJ, k - m, s - m)\ P(JJ - IJ, k - m, t - m)\ mul\big(q^i - 1, i = 1..k - s\big)\ mul\big(q^i - 1, i = 1..k - t\big)\big) \Big/ \big(mul\big(q^i - 1, i = 1..k - s - r\big)\ mul\big(q^i - 1, i = 1..r\big)^2\ mul\big(q^i - 1, i = 1..k - t - r\big)\big), m = 0..\min(k, IJ, s, t)\Big), t = 0..JJ\Big), s = 0..II\Big)$$

> II:=8;

JJ:=0;

IJ:=0;

k:=4;

r:=3;

s;

t;

m;

$II := 8$

$JJ := 0$

$IJ := 0$

$k := 4$

$r := 3$

s

t

m

> factor(EE(II,JJ,IJ,k,r));

$$\frac{1}{q^{31}}\left(\left(q^2+1\right)^2\left(q^{22}+q^{20}+q^{19}+3\,q^{18}+3\,q^{16}+q^{15}+2\,q^{14}+q^{13}+q^{12}-q^{11}-q^9-2\,q^8-2\,q^6-2\,q^5-2\,q^4-2\,q^2-1\right)\left(q+1\right)^2\right)$$

> II:=8;

JJ:=0;

IJ:=0;

k:=4;

r:=2;

s;

t;

m;

$II := 8$

$JJ := 0$

$IJ := 0$

$k := 4$

$r := 2$

s

t

m

> factor(EE(II,JJ,IJ,k,r));

$$\frac{1}{q^{29}}\left(\left(q^2+q+1\right)^2\left(q^2+1\right)^2\left(q^{22}+q^{19}+2\,q^{18}-q^{17}+q^{15}-q^{14}-q^{12}-2\,q^{11}-q^{10}-2\,q^8+q^6+2\,q^3+1\right)\right)$$

Yukarıdakilere benzer birçok deneme sonucu dikkatle incelendiğinde $\mathbb{E}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}\begin{bmatrix} k-r_J \\ r \end{bmatrix}\right) = q^{-r(I+J)}\begin{bmatrix} k \\ r \end{bmatrix}\sum_{i=0}^{r} q^{i^2}\begin{bmatrix} k-r \\ i \end{bmatrix}\begin{bmatrix} r \\ i \end{bmatrix} q^{IJ(r-i)}$ eşitliğinin sağlandığı görülmüştür. Bu sonuçlardan yararlanarak da $\operatorname{cov}\left(\begin{bmatrix} k-r_I \\ r \end{bmatrix}, \begin{bmatrix} k-r_J \\ r \end{bmatrix}\right)$ hesaplanmıştır. Son olarak da tüm bu elimizdeki verilerden $\operatorname{cov}\left(W_C^r(x), W_C^r(y)\right)$ polinomu elde edilmiştir.

BÖLÜM 7

LİNEER OLMAYAN BİR KOD: FIBONACCI KODU VE MATRİSİ

Fibonacci sayılar teorisi, Fibonacci Q matris olarak bilinen teori ile bütünleşmiştir. [16]. Mertebesi 2 olan Fibonacci Q matrisi 2*x*2 $Q=\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}$ matrisi ile temsil edilir ve $\det Q^n=-1$ dir [17]. Burada $Q=Q_1$ olup $F(n)$ ve F$(n-1)$ terimleri belli iken $f(n+1)$ matrisinin elde edilmesini $Q_1\begin{bmatrix}F(n)\\ F(n-1)\end{bmatrix}=\begin{bmatrix}F(n+1)\\ F(n)\end{bmatrix}$ eşitliğiyle sağlar. Q_1 matrisinin *n.* kuvveti $Q_1^n=\begin{bmatrix}F(n+1) & F(n)\\ F(n) & F(n-1)\end{bmatrix}$ ve $\det Q_1^n=F(n+1)F(n-1)-F^2(n)=(-1)^n$ dir [17]. Burada $F(n-1),\ F(n)$ ve $F(n+1)$ $n=0,\pm1,\pm2,\pm3,...$ olmak üzere;

$$F(n+1)=F(n)+F(n-1) \tag{7.1}$$

$$F(1)=F(2)=1$$

şartlarını sağlayan Fibonacci sayılarıdır. (7.1) denklemi Cassini formülü olarak adlandırılır [16]. Stokhov, 1997 de Fibonacci matris teorisinin, p-Fibonacci sayılarının notasyonlarını kullanarak Fibonacci p sayılarını göstermiştir [18]. $p=0,1,2,3,...$ olmak üzere Q_p matrisi,aşağıdaki şartlar sağlanmak üzere,

$F_p(1)=F_p(2)=...=F_p(p)=F_p(p+1)$
$n>p+1$ için $F_p(n)=F_p(n-1)+F_p(n-p+1)$

$$Q_p = \begin{bmatrix} 1 & 1 & 0 & 0 & . & 0 & 0 \\ 0 & 0 & 1 & 0 & . & 0 & 0 \\ 0 & 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & . & 1 & 0 \\ 0 & 0 & 0 & 0 & . & 0 & 1 \\ 1 & 0 & 0 & 0 & . & 0 & 0 \end{bmatrix}_{(p+1)x(p+1)}$$

ile verilir [19]. Örnek verecek olursak p=0,1,2,3 için Q_p matrisleri aşağıdaki gibidir.

$$Q_0 = [1],\ Q_1 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix},\ Q_2 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \text{ ve } Q_3 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$\det(Q_p) = (-1)^p$ dir. Q_p matrisinin genişlemesi aşağıda verilmiştir [17].

$$Q_p^n = \begin{bmatrix} F_p(n+1) & F_p(n) & ... & F_p(n-p+2) & F_p(n-p+1) \\ F_p(n-p+1) & F_p(n-p) & ... & F_p(n-2p+2) & F_p(n-2p+1) \\ ... & ... & ... & ... & ... \\ F_p(n-1) & F_p(n-2) & ... & F_p(n-p) & F_p(n-p-1) \\ F_p(n) & F_p(n-1) & ... & F_p(n-p+1) & F_p(n-p) \end{bmatrix}_{(p+1)x(p+1)}$$

Yine aynı çalışmada

$Q_p^n = Q_p^{n-1} + Q_p^{n-p-1}$ (toplama özelliği)

$Q_p^n x Q_p^m = Q_p^m x Q_p^n = Q_p^{n+m}$ (çarpma özelliği)

eşitlikleri ve Q_p^n matrisinin determinantının $p = 2k$ için 1 e p=2k+1 için $(-1)^n$ e eşit olduğu ispatlanmıştır.

Ayrıca özel olarak Q_1^n matrisinin tersi, n=2k için $Q_1^{-2k}=\begin{bmatrix} F_{2k-1} & -F_{2k} \\ -F_{2k} & F_{2k+1} \end{bmatrix}$ ve n=2k+1 için $Q_1^{-(2k+1)}=\begin{bmatrix} -F_{2k} & F_{2k+1} \\ F_{2k+1} & -F_{2k+2} \end{bmatrix}$ dir [16].

7.1. FIBONACCI KODLAMA VE DEKODLAMA METODU

Elimizdeki bir mesajı p=0,1,2,3,… olmak üzere (p+1) mertebeli bir M matrisi temsil etsin. Bu mesajın kodlama matrisi olarak (p+1) mertebeli Q_p^n matrisini, dekodlama matrisi olarak da Q_p^{-n} matrisini alalım. $Mx\,Q_p^n=E$ dönüşümüne Fibonacci kodlaması, $Ex\,Q_p^{-n}=M$ dönüşümüne ise Fibonacci dekodlaması denir [16]. Burada E kod matrisidir. Fibonacci kodlama\dekodlama metoduna bir örnek verelim.

$$M=\begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}_{2x2} \tag{7.2}$$

matrisi mesajımızı temsil etsin. Matrisin tüm elemanlarının pozitif tam sayılar, yani $m_1, m_2, m_3, m_4 > 0$ olduğunu kabul edelim. p=1 için n-Fibonacci matrisinin herhangi bir değerini kodlama matrisi olarak seçelim. n=2 için

$$Q_1^2=\begin{bmatrix} F_1(3) & F_1(2) \\ F_1(2) & F_1(1) \end{bmatrix}=\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \tag{7.3}$$

dir. Q_1^2 nin ters matrisi

$$Q_1^{-2}=\begin{bmatrix} F_1(1) & -F_1(2) \\ -F_1(2) & F_1(3) \end{bmatrix}=\begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix}$$

dir. Fibonacci kodlama mesajı M mesajının Q_1^2 matrisiyle çarpımından oluşur.

$$MxQ_1^2 = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 2m_1+m_2 & m_1+m_2 \\ 2m_3+m_4 & m_3+m_4 \end{bmatrix} = \begin{bmatrix} e_1 & e_2 \\ e_3 & e_4 \end{bmatrix} = E \qquad (7.4)$$

$e_1=2m_1+m_2$, $e_2=m_1+m_2$, $e_3=2m_3+m_4$, $e_4=m_3+m_4$ olur [20]. Daha sonra E kod mesajı kanala yollanır. (7.3) ile verilen mesajın dekodlaması aşağıdaki gibidir:

$$\begin{bmatrix} e_1 & e_2 \\ e_3 & e_4 \end{bmatrix}\begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} e_1-e_2 & -e_1+2e_2 \\ e_3-e_4 & -e_3+2e_4 \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} = M$$

Örneğin, mesajımız 12456212 ondalık sayıların bir dizisi olsun. Mesajımızı $M = \begin{bmatrix} 124 & 56 \\ 21 & 2 \end{bmatrix}$ matrisiyle temsil edelim. Q_1^2 yi kodlama matrisi olarak seçersek,

$$E = MxQ_1^2 = \begin{bmatrix} 124 & 56 \\ 21 & 2 \end{bmatrix}\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 304 & 180 \\ 44 & 23 \end{bmatrix} \text{ ve}$$

$$M = ExQ_1^{-2} = \begin{bmatrix} 304 & 180 \\ 44 & 23 \end{bmatrix}\begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} 124 & 56 \\ 21 & 2 \end{bmatrix}$$

elde edilir. Buradan M nin elemanlarının verilerdeki çift veya tek sayılar olarak seçilebildiği söylenir [16].

KAYNAKLAR

1. Helleseth T., Kløve T., Levenshtein V.I. and Ytrehus O., "Bounds on the minimum support weights" ***IEEE Transaction on Information Theory,*** 41(4): 432–440 (1995).

2. Jones G. A. and Jones J. M., "Information and coding theory", 2nd ed., ***Springer***, Berlin, 54-63 (2002).

3. Hill R., "A first course in coding theory", 1st ed., ***Oxford University Press***, Oxford, 25-29 (1986).

4. Van Lint J. H. and Wilson R. M., "A course in combinatorics", 2nd ed., ***Cambridge Univ. Press***, Cambridge, 33-44 (2001).

5. Roman S., "Gradute texts in mathematics", 2nd ed**.**, ***Springer Verlay***, 34-38 (1992).

6. Pretzel O., "Error correcting codes and finite fields", 1st ed., ***Imperial College***, London, 56-64 (1992).

7. Tsfasman M. A. and Vladut S. G., "Algebraic geometric codes", 1st ed., ***Kluwer Academic Publishers***, Dordrecht, 17-29 (1991).

8. Kasami T., Takata T., Fujiwara T. and Lin S., "On the optimum bit orders with respect to the state complexity of trellis diagrams for binary linear codes", ***IEEE Transaction on Information Theory,*** 39(1): 242–245 (1993).

9. Wei V., "Generalized Hamming weights for linear codes", ***IEEE Transaction on Information Theory***, 37 (5): 1412–1418 (1991).

10. Britz T., "Higher support matroids", ***Discrete Mathematics***, 307(1): 2300-2308 (2007).

11. Helleseth T., Kløve T. and Mykkeltveit J., "The weight distribution of irreducible cyclic codes with block length n_1 $((q^1 -1)/N)$"**,** ***Discrete Mathematics***, 18(2): 179-211 (1977).

12. Helleseth T., Kløve T. and Ytrehus O., "Generalized Hamming weights of linear codes", ***IEEE Transaction on Information Theory,*** 38(3): 1133–1140 (1992).

13. Kac V. and Cheung P., "Quantum Calculus", 1st ed., ***Universitext Springer***, Berlin, 14-27 (2001).

14. Klyachko A. A. and Özen İ., "Correlations between the ranks of submatrices and weights of random codes" ***Finite Fields and Their Applications,*** 15(4): 497–516 (2009).

15. Özen İ. ve Tekin E., "Moments of the support weight distribution of linear codes", ***Designs, Codes and Cryptography***, doi: 10.1007/s10623-011-9597-7 (2011).

16. Basu M. and Prasad B., "The generalized relations among the code elements for Fibonacci coding theory**"**, ***Chaos, Solutions and Fractals,*** 41(5): 2517-2525 (2009).

17. Stakhov A. P., "Fibonacci matrices a generalization of the Cassini formula, and a new coding theory", ***Chaos, Solutions and Fractals***, 30(1):56-66 (2006).

18. Blinovsky V., Erez U. and Litsyn S. "Weight distribution moments of random linear/coset codes", ***Designs Codes and Cryptography,*** 57(2): 127–138 (2010).

19. Gallager R. G., "Low density Parity-Check Codes", 1st ed., ***MIT Press***, Cambridge (1963).

ÖZGEÇMİŞ

Eda TEKİN 1987 yılında Bursa'da doğdu; ilk ve orta öğrenimini aynı şehirde tamamladı. Bursa Kız Lisesi'nden mezun oldu. 2005 yılında Dumlupınar Üniversitesi Matematik Bölümü'nde öğrenime başlayıp 2009 yılında iyi derece ile mezun oldu. 2009 yılında Karabük Üniversitesi Matematik Bölümü'nde Araştırma görevlisi olarak göreve başladı ve halen aynı yerde çalışmaya devam etmektedir.

ADRES BİLGİLERİ

Adres : Karabük Üniversitesi
Fen Fakültesi Matematik Bölümü
Balıklarkayası Mevkii / KARABÜK

Tel : (541) 278 7075
E-posta : edavatansever@karabuk.edu.tr