
GAZİ ÜNİVERSİTESİ
Gazi
1926

YÜKSEK BOYUTLU GENOM VERİLERİNDE SIRALI ÖRÜNTÜLERE SAHİP BAĞIMLI ÖZELLİKLERİN SEÇİMİ İÇİN S TESTİ UYARLAMASI

Deniz CEBELİ

YÜKSEK LİSANS TEZİ
İSTATİSTİK ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AĞUSTOS 2022

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Deniz CEBELİ

08/08/2022

YÜKSEK BOYUTLU GENOM VERİLERİNDE SIRALI ÖRÜNTÜLERİ SAHİP BAĞIMLI ÖZELLİKLERİN SEÇİMİ İÇİN S TESTİ UYARLAMASI

(Yüksek Lisans Tezi)

Deniz CEBELİ

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Ağustos 2022

ÖZET

Yüksek boyutlu verilerde özellik seçimi makine öğrenmesindeki kritik adımlardan biridir. Yüksek boyutlu veriler çok sayıda niteliğe karşın az sayıda gözlem içeren veri yapılarıdır. Özellikle gen verilerine ilişkin çalışmalarda bu tarz verilerle çok sık karşılaşılmaktadır. Son yıllarda makine öğrenmesi tekniklerinin yaygınlaşmasıyla genom çapında ilişkilendirme çalışmaları (GWAS) artış göstermiştir. Bu tarz çalışmalarda tek nükleotid polimorfizm (SNP) düzeyindeki artış ile marker değerlerindeki artış veya azalış örüntüleri tespit edilmeye çalışılır. İstatistikte bu tarz örüntüler Jonckheere-Terpstra (JT), Terpstra-Magel (TM), Ferdhiana-Terpstra-Magel (FTM), KTP, Modified JT ve S testi gibi sıralı alternatif testleriyle incelenir. Ancak, yüksek boyutlu veriler için bu testlerin kullanımı hesaplama zamanı bakımından ekonomik değildir. Bu nedenle, bu testlerin yüksek boyutlu veriler için uyarlanması önem arz etmektedir. Bu çalışmada, aşırı çarpık dağılımlarda ve/veya konveks/konkav alternatif hipotez durumlarında JT testine göre daha iyi sonuçlar veren S istatistiğinin yüksek boyutlu veriler için uyarlanmış algoritması önerilmiştir. Elde edilen sonuçlar S istatistiğinin yüksek boyutlu veriler için daha kullanışlı olduğunu göstermektedir.

Bilim Kodu	:	20515
Anahtar Kelimeler	:	Yüksek boyutlu veri, Özellik seçimi, Sıralı alternatif, Genom, Simülasyon
Sayfa Adedi	:	43
Danışman	:	Prof. Dr. Bülent ALTUNKAYNAK

ADAPTATION OF S TEST FOR SELECTION OF DEPENDENT ATTRIBUTES WITH ORDERED PATTERNS IN HIGH-DIMENSIONAL GENOME DATA

(M. Sc. Thesis)

Deniz CEBELİ

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

August 2022

ABSTRACT

Feature selection in high-dimensional data is one of the critical steps in machine learning. High-dimensional data are data structures that contain many attributes but few observations. Especially in studies on gene data, such data are frequently encountered. With the widespread use of machine learning techniques in recent years, the number of genome-wide association studies (GWAS) has increased. In such studies, the relationship between the increase in the single-nucleotide polymorphism (SNP) level and the patterns of increase or decrease in the marker values are tried to be determined. In statistics, such patterns are examined with ordered alternative tests such as Jonckheere-Terpstra (JT), Terpstra-Magel (TM), Ferdhiana Terpstra-Magel (FTM), KTP, Modified JT and S test. However, the use of these tests for high-dimensional data is not economical in terms of computation time. Therefore, these tests need to be adapted for high-dimensional data. Lin et al. (2019) proposed the fastJT algorithm for high-dimensional data. On the other hand, power test statistics than JT statistics are available in the literature, especially in extremely skewed distributions and/or convex/concave alternative hypothesis situations (Shan et al., 2014, Altunkaynak & Gamgam, 2020). In this study, an adapted algorithm of S statistics for high-dimensional data is proposed, which gives better results than JT test in extreme skewed distributions and/or convex/concave alternative hypothesis situations. The results show that the S statistic is more useful for high-dimensional data.

Science Code	:	20515
Key Words	:	High dimensional data, feature selection, ordered alternatives, genom, simulation
Page Number	:	43
Supervisor	:	Prof. Dr. Bülent ALTUNKAYNAK

TEŞEKKÜR

Bu çalışmanın gerçekleştirilmesinde, destek ve emeklerini esirgemeyip bana her daim yol gösteren, değerli bilgi ve tecrübeleriyle benim ufkumu açan, kullandığı her kelimenin hayatıma kattığı önemini asla unutmayacağım saygıdeğer danışman hocam; Prof. Dr. Bülent ALTUNKAYNAK' a sonsuz teşekkür ve saygılarımı sunarım. Çalışmalarım süresince tüm zorlukları benimle göğüsleyen ve hayatımın her evresinde bana sonsuz destek olan kıymetli anneme, babama ve kardeşlerime teşekkürü borç bilirim. Bu süreçte her anımda yanımda olan sevgili arkadaşlarıma da çok teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET iv

ABSTRACT v

TEŞEKKÜR vi

İÇİNDEKİLER vii

ÇİZELGELERİN LİSTESİ viii

ŞEKİLLERİN LİSTESİ ix

SİMGELER VE KISALTMALAR x

1. GİRİŞ 1

2. PROBLEM TANIMI VE LİTERATÜR ÇALIŞMALARI 3

3. SIRALI ALTERNATİF TESTLERİ 7

3.1. JT Testi 8

3.2. S Testi 13

3.3. Hızlandırılmış S Testi 14

3.3.1. Motivasyon 14

3.3.2. Algoritma 16

4. SİMÜLASYON ÇALIŞMASI 19

4.1. Simülasyon Senaryoları 19

4.2. Simülasyon Sonuçları 22

5. SONUÇ VE ÖNERİLER 30

KAYNAKLAR 33

EKLER 37

ÖZGEÇMİŞ 43

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 2.1. Genom çapı çalışmaları için veri yapısı..... 3

Çizelge 3.1. Jonckheere verisi 9

Çizelge 3.2. U_{ij} değerlerinin hesaplanması için hızlandırılmış JT algoritması 10

Çizelge 3.3. U_{12} istatistiğinin hesaplanması 12

Çizelge 3.4. Sıra sayılarıyla Jonckheere verisi 14

Çizelge 3.5. D_{ij} değerlerinin hesaplanması için hızlandırılmış S algoritması 16

Çizelge 3.6. D_{12} değerinin hesaplanması 16

Çizelge 4.1. Simülasyon çalışmasında kullanılan senaryolar 21

Çizelge 4.2. Normal(0,1)+a dağılımından seçilen verilerle güç değerleri 22

Çizelge 4.3. t_3+a dağılımından seçilen verilerle güç değerleri 24

Çizelge 4.4. Ki-kare(1)+a dağılımından seçilen verilerle güç değerleri 26

Çizelge 4.5. Üstel(1)+a dağılımından seçilen verilerle güç değerleri..... 28

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 2.1. VEGF-A, VEGF-C ve MCP1 proteinlerinin plazma düzeyleri 4

Şekil 2.2. Fenitoin plazma konsantrasyonu 4

Şekil 3.1. Doğrusal, konveks ve konkav alternatif hipotez yapıları 15

Şekil 4.1. Dağılımlara ilişkin histogram grafikleri 20

Şekil 4.2. Normal dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri. 23

Şekil 4.3. t dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri. 25

Şekil 4.4. Ki-kare dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri. 27

Şekil 4.5. Üstel dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri. 29

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklamalar**
pg/ml	Pikogram/Mililitre

Kısaltmalar	**Açıklamalar**
ABCB1	ATP Bağlama Kaset Alt Ailesi B Üyesi 1
CALGB 80303	Kanser ve Lösemi Grubu B'nin faz III çalışması
CYP2C19	Sitokrom P450 Ailesi 2 Alt Ailesi C Üyesi 19
CYP2C9	Sitokrom P450 Ailesi 2 Alt Ailesi C Üyesi 9
FASTJT	Hızlandırılmış Jonckheere-Terpstra testi
FASTS	Hızlandırılmış Shan’s S testi
FTM	Ferdhiana Terpstra-Magel testi
GWAS	Genom Çapında İlişkilendirme Çalışmaları
JT	Jonckheere-Terpstra testi
KTP	Terpstra-Chang-Magel testi
MARK	Marker Değerleri
MCP1	Monosit Kemotaktik Protein-1
MJT	Modifiye Edilmiş Jonckheere-Terpstra testi
MRG	Manyetik Rezonans Görüntüleme
S	Shan’s S testi
SNP	Tek Nükleotid Polimorfizm
TM	Terpstra-Magel testi
VEGF-A	Vasküler Endotelyal Büyüme Faktörü-A
VEGF-C	Vasküler Endotelyal Büyüme Faktörü-C

1. GİRİŞ

Nicel bir özelliğin sıralı bir özellikle bağlantılı olup olmadığını değerlendirmek için sıralı alternatif testleri kullanılır. Bu tarz çalışmalar özellikle sağlık alanında yaygın olarak bulunmaktadır. Amonyak seviyeleri ile hepatik ensefalopatinin şiddeti arasındaki ilişki (Ong ve diğerleri., 2003), anormal MRG bulgularının kemik iliği hastalığıyla ilişki (Bredella ve diğerleri, 2006), insan genlerindeki tek nükleotid polimorfizmleri (SNP, Single-nucleotide polymorphism) ile kantitatif fenotipler arasındaki ilişki (Hoffmeyer ve diğerleri, 2000; Cheng ve diğerleri, 2005; Kawaguchi ve diğerleri, 2012; Uchiyama ve diğerleri, 2012; Tan ve diğerleri, 2014; Yorifuji ve diğerleri, 2018) bu tarz çalışmalara örnek olarak verilebilir.

Sıralı alternatiflerin incelenmesi için hem parametrik hem de parametrik olmayan yaklaşımlar söz konusudur. Ancak parametrik yöntemlerin, aykırı değerlere karşı sağlam olmaması ve bunun yanında normallik varsayımı ve varyans homojenliği varsayımı gibi istatistiksel varsayımları gerektirmesi parametrik olmayan yöntemlerin daha yaygın olarak kullanılmasını sağlamıştır. Bu alanda kullanılan parametrik olmayan ilk test Jonckheere-Terpstra (JT) testidir (Jonckheere, 1954). Ancak farklı dağılım yapıları altında ve farklı alternatif hipotezler durumunda JT istatistiğine göre daha güçlü olan testler literatürde yer almaktadır (Altunkaynak ve Gamgam, 2020). Özellikle Shan (S) test istatistiği aşırı çarpık dağılımlarda ve konveks/konkav alternatif hipotez yapılarında diğer testlere göre daha iyi sonuçlar vermektedir (Shan ve diğerleri, 2014).

Diğer yandan, yüksek boyutlu verilerde birçok özellik içerisinden sıralı alternatifleri oluşturan özelliklerin seçilmesi hesaplama zamanı açısından oldukça maliyetlidir. Bu tarz yüksek boyutlu veriler için genom çapında ilişkilendirme çalışmaları (GWAS) örnek olarak verilebilir. Özellikle SNP çalışmalarında SNP düzeyindeki artış ile marker değerlerindeki artış veya azalış örüntüleri tespit edilmeye ve binlerce SNP arasından seçim yapılmaya çalışılır (Allabi ve diğerleri, 2005; Komatsu ve diğerleri, 2021). Bu nedenle JT veya S testi gibi testlerin doğrudan bu tarz çalışmalara uygulanması hesaplama zamanı açısından uygun değildir. Çünkü bu testler sıralı özelliğin farklı düzeylerde yer alan sayısal değerlerin birbirleriyle karşılaştırılmasına dayalıdır. Bu nedenle, Lin ve diğerleri (2019) fastJT algoritmasını önermişlerdir. Bu algoritma sayesinde gereksiz sorgulamalar ortadan kaldırılmış ve test istatistiğinin hesaplanması daha kısa bir sürede gerçekleştirilmiştir. Ancak daha önce de belirtildiği gibi birçok değişken barındıran GWAS verileri çarpık dağılımlar

ve farklı alternatif hipotez yapıları içerebilmektedir. Bu tarz durumlar için küçük veri yapılarında S testinin JT testine göre daha iyi sonuçlar vermiş olması büyük boyutlu veriler için S testinin JT testine benzer şekilde uyarlanması uygun olacaktır. Bu nedenle bu çalışmada S testinin yüksek boyutlu veriler için hızlandırılmış bir versiyonunu önermekteyiz.

Tez çalışmasının ikinci bölümünde problem tanımı verilmiş ve literatür çalışmalarıyla desteklenmiştir. Üçüncü bölümde JT ve S sıralı alternatif testleri verilmiş ve hızlandırılmış algoritmaların nasıl uygulanacağı detaylandırılmıştır. Dördüncü bölüm simülasyon çalışması ile JT ve S testlerinin karşılaştırılmasına ilişkindir. Son bölümde sonuçlar verilmiştir.

2. PROBLEM TANIMI VE LİTERATÜR ÇALIŞMALARI

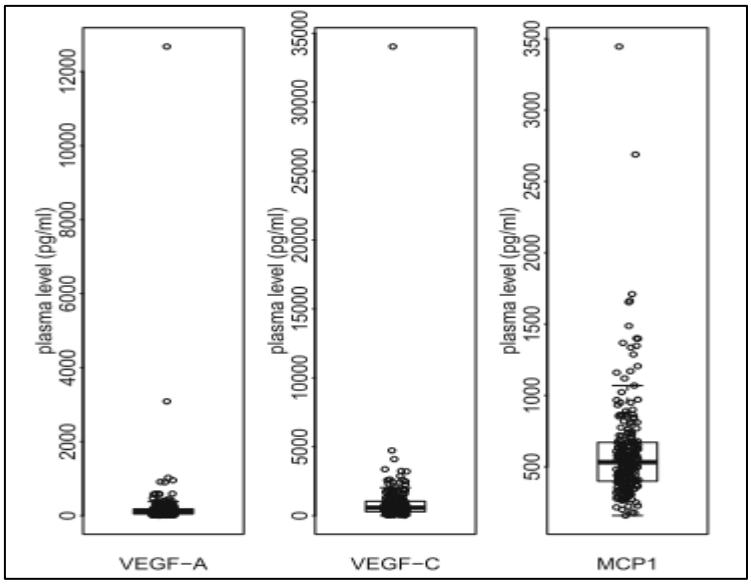
İnsan genlerindeki tek nükleotid polimorfizmleri (SNP'ler) ile kantitatif fenotipler arasındaki ilişkilerin incelendiği genom çalışmaları literatürde önemli bir yer tutmaktadır. Bu tarz çalışmalar için *p* tane SNP ve *q* tane marker içeren bir veri yapısı aşağıdaki gibi verilebilir.

Çizelge 2.1. Genom çapı çalışmaları için veri yapısı

Geno	SNP:1	SNP:2	…	SNP:p	MARK:1	MARK:2	…	MARK:q
1	0	2	…	2	-0,784	0,415	…	1,451
2	1	1	…	1	2,157	-2,011	…	-2,137
3	1	0	…	1	1,135	-1,072	…	1,099
…	…	…	…	…	…	…	…	…
100	2	0	…	1	0,212	0,473	…	-0,541

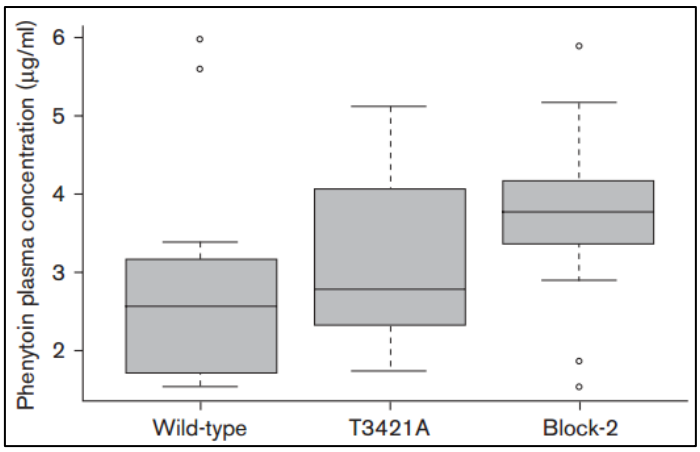
Burada SNP'ler farklı düzeylere (0, 1, 2,…) sahip faktörlerdir. Markerlardan elde edilen ölçümler ise bağımlı değişken görevi görmektedir ve her bir marker için birden fazla SNP aynı anda (bağımlı SNP'ler) örüntü yapısına sahip olabilmektedir. Örneğin, SNP düzeylerinin farklı proteinleri (VEGF-A, VEGF-C, MCP1, vb.) temsil ettiği marker değerlerinin ise plazma düzeylerini (pg/ml) ifade ettiği çalışmalar önemli genom çalışmaları arasında yer almaktadır (Innocenti ve diğerleri, 2018; Hoffmeyer ve diğerleri, 2000; Rakvag ve diğerleri, 2005; Cheng ve diğerleri, 2005; Takahisa ve diğerleri, 2012; Tan ve diğerleri, 2014; Uchiyama ve diğerleri, 2012; Yorifuji ve diğerleri, 2018; Lin ve diğerleri, 2019).

Bu tarz çalışmalarda SNP'lerin düzeylerine göre marker değerlerinin konum parametresinde sıralı bir artış veya azalış örüntüsü araştırılır ve bu örüntüyü oluşturan SNP alt kümesi seçilir. Örneğin, CALGB 80303 protein biomarker çalışmasından elde edilen bir grafik aşağıdaki gibi verilmiştir (Lin ve diğerleri, 2019).



Şekil 2.1. VEGF-A, VEGF-C ve MCP1 proteinlerinin plazma düzeyleri (Lin ve diğerleri, 2019)

SNP düzeyleri (0:VEGF-A, 1:VEGF-C, 2:MCP1) arttıkça plazma düzeylerinde bir artış gözlemlenmektedir (Şekil 2.1).



Şekil 2.2. Fenitoin plazma konsantrasyonu (Allabi ve diğerleri, 2005)

Şekil 2.2'de verilen örüntü yapısına göre ise siyah bir popülasyonda CYP2C9, CYP2C19 ve ABCB1 genetik polimorfizmlerinin fenitoin metabolizması üzerindeki etkisini belirlemek için yapılan çalışmada da yer almaktadır (Allabi ve diğerleri, 2005).

Bu tarz sıralı alternatiflerin incelenmesi için literatürde parametrik veya parametrik olmayan testler mevcuttur. Ancak, Şekil 2.2 incelendiğinde bu tarz verilerde dikkat çekici bir başka

durum ise aykırı değerlerin varlığıdır. Aykırı değer durumunda daha güçlü olan parametrik olmayan testler tercih edilir. Bu amaç için geliştirilmiş testlerden en önemlileri Jonckheere-Terpstra (JT), Terpstra-Magel (TM), Ferdhiana-Terpstra-Magel (FTM), Terpstra-Chang-Magel (KTP), Modified JT ve Shan (S) testi olarak verilebilir.

Diğer yandan, bu tarz GWAS verilerinde araştırma konusu belli bir marker ölçümü üzerinde sıralı örüntüler oluşturan birden fazla SNP'nin eşanlı seçimidir. Çok sayıda SNP (özellik) içerisinden önemli SNP'lerin seçilmesi aynı zamanda makine öğrenmesinde önemli problemlerden birisidir. Ve özellik seçimi olarak adlandırılmaktadır. Tek bir SNP ve tek bir marker ölçüm değeri olduğunda JT, TM, FTM, KTP, Modified JT ve S testlerini uygulamak kolaydır. Ancak yüzlerce SNP'nin ve marker ölçümünün bulunduğu veri yapılarında bu testlerin kullanımı işlem zamanı açısından ekonomik değildir. Bu nedenle klasik testlerin büyük boyutlu veriler için uyarlanması ihtiyacı ortaya çıkmıştır. Bu konuda bilinen tek çalışma Lin ve diğerleri (2019) tarafından yapılmıştır ve JT testi çok boyutlu veriler için uyarlanmasıdır. Bu çalışmada fastJT adlı uyarlanmış hesaplama algoritması ile yüksek boyutlu veriler için işlem zamanı $O(n^2)$ den $O(n\log n)$ düzeyini indirgenmiştir.

Ancak, literatürde farklı veri yapılarında JT testine göre daha güçlü test istatistikleri vardır. Örneğin, Neuhäuser, Liu ve Hothorn (1998) yaptıkları çalışmada MJT testinin JT testine göre daha güçlü olduğunu göstermişlerdir. Shan (2014), normal, t, üstel ve karma dağılım durumlarını içeren simülasyon çalışmasıyla aşırı çarpık dağılımlarda ve konveks/konkav alternatif hipotez yapılarında S testinin diğer testlere göre daha iyi sonuçlar verdiğini göstermiştir. Altunkaynak ve Gamgam (2020) farklı dağılım yapıları altında ve farklı alternatif hipotezler durumunda JT istatistiğine göre daha güçlü olan testleri göstermişlerdir.

Bu çalışmada JT testine göre daha iyi sonuçlar veren S testinin yüksek boyutlu veriler için uyarlaması yapılmıştır. Geliştirilen algoritma fastJT algoritmasıyla benzer mantığa dayalıdır ve klasik S testinin sahip olduğu işlem zamanını $O(n^2)$ den $O(n\log n)$ düzeyine indirgemektedir.

3. SIRALI ALTERNATİF TESTLERİ

$X_{i1}, X_{i2}, ..., X_{in}$, $i = 1, ..., k$, k popülasyondan n_i boyutunda rastgele bağımsız örnekler olsun. Sürekli kümülatif dağılım fonksiyonu $F_i(x) = F((x - \theta_i)/\sigma_i)$, burada $-\infty < \theta_i < +\infty$ ve $\sigma_i > 0$ sırasıyla konum ve ölçek parametreleridir. Popülasyonların ortak sürekli kümülatif dağılım fonksiyonuna sahip olup olmadığını belirleyen H_0 hipotezi,

$$H_0 : F_1(x) = F_2(x) = ... = F_k(x) \qquad \forall x \tag{3.1}$$

biçiminde ifade edilir.

Belirli varsayımlar altında ve farklı H_1 biçimleri Eş. 3.1'deki H_0 hipotezi test etmek için bir dizi test istatistiği önerilmiştir. Sıralı alternatif, dağılımların stokastik olarak sıralandığını belirtir, Buna göre,

$$H_1 : F_1(x) \geq F_2(x) \geq ... \geq F_k(x) \quad \exists x : F_1(x) > F_k(x) \tag{3.2}$$

dir.

H_1'in doğruluğu, X_i, X_{i+1}, $i = 1, 2, ..., k-1$'den daha küçük olma eğilimindedir, çünkü $F_i(x) \leq F_{i+1}(x)$, $P(X_i \geq X_{i+1}) \geq 1/2$ anlamına gelir. Konum modelinin özel durumu için Eş. 3.2, (Terpstra ve diğerleri, 2011)

$$H_1 : \theta_1 \leq \theta_2 \leq ... \leq \theta_k \qquad (\theta_1 < \theta_k) \tag{3.3}$$

biçiminde ifade edilir.

Benzer şekilde, sıralı alternatif hipotezi

$$H_1 : F_1(x) \leq F_2(x) \leq ... \leq F_k(x) \qquad \exists x : F_1(x) < F_k(x) \tag{3.4}$$

X_i, X_{i+1}, $i=1,2,...,k-1$'den daha büyük olma eğilimindedir. $F_i(x) \le F_{i+1}(x)$, Eş. 3.4'te verilen H_1'in doğruluğu altında $P(X_i \ge X_{i+1}) \ge 1/2$ anlamına gelir. Konum modeli için Eş. 3.4,

$$H_1 : \theta_1 \le \theta_2 \le ... \le \theta_k \quad (\theta_1 > \theta_k) \tag{3.5}$$

ifadesi ile eşdeğerdir.

3.1. JT Testi

Jonckheere-Terpstra testi (JT), k-grup durumunda sıralı alternatifleri test etmek için Terpstra (1952) ve Jonckheere (1954) tarafından önerilmiştir. Parametrik olmayan bu test istatistiği $k(k-1)/2$ adet Mann-Whitney istatistiğinin toplamı olarak

$$JT = \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} U_{ij} \tag{3.6}$$

biçiminde tanımlanır.

n_i ve n_j sırasıyla i. ve j. gruplar için örneklem büyüklükleri olsun. Bu durumda i. ve j. gruplar için Mann-Whitney istatistiği (U_{ij}) aşağıdaki gibi tanımlanır:

$$U_{ij} = \sum_{l=1}^{n_i} \sum_{m=1}^{n_j} I(X_{il} < X_{jm}) \tag{3.7}$$

Burada $X_{il} < X_{jm}$ doğruysa, $I(X_{il} < X_{jm})$ bir indikatör fonksiyon olmak üzere, $I(X_{il} < X_{jm}) = 1$ aksi halde $I(X_{il} > X_{jm}) = 0$ olur.

JT istatistiği H_0'ın doğruluğu altında normal dağılır. Bu istatistiğin ortalaması ve varyansı sırasıyla aşağıdaki biçimde hesaplanır.

$$E(JT) = \frac{N^2 - \sum_{i=1}^{k} n_i^2}{4} \tag{3.8}$$

ve

$$V(JT) = \frac{N^2(2N+3) - \sum_{i=1}^{k} n_i^2(2n_i+3)}{72}. \tag{3.9}$$

Jonckheere (1954)'den alınan yapay veri seti Çizelge 3.1'de görülmektedir.

Çizelge 3.1. Jonckheere verisi (Jonckheere, 1954)

I	II	III	IV
19	21	40	49
20	61	99	110
60	80	100	151
130	129	149	160

i. grubun konum parametresi θ_i olmak üzere $H_0 : \theta_1 = \theta_2 = \theta_3 = \theta_4$ hipotezi $H_1 : \theta_1 \le \theta_2 \le \theta_3 \le \theta_4$ (en az bir eşitsizlik doğru) alternatif hipotezine karşı test edilmek istensin. Bu durumda

$$JT = \sum_{i=1}^{3} \sum_{j=i+1}^{4} U_{ij} = U_{12} + U_{13} + U_{14} + U_{23} + U_{24} + U_{34}$$

olarak yazılır. Burada U_{12} aşağıdaki gibi hesaplanır.

$$\begin{aligned} U_{12} &= \sum_{l=1}^{5} \sum_{m=1}^{5} I(X_{1l} < X_{2m}) \\ &= I(19<21) + I(19<61) + I(19<80) + I(19<129) \\ &\quad + I(20<21) + I(20<61) + I(20<80) + I(20<129) \\ &\quad + I(60<21) + I(60<61) + I(60<80) + I(60<129) \\ &\quad + I(130<21) + I(130<61) + I(130<80) + I(130<129) \end{aligned}$$

$X_{il} < X_{jm}$ doğruysa $I(X_{il} < X_{jm}) = 1$ aksi halde 0 olduğundan,

$$U_{12} = 1+1+1+1+1+1+1+1+0+1+1+1+0+0+0+0 = 11$$

biçiminde elde edilir. Benzer şekilde $U_{13} = 12$, $U_{14} = 13$, $U_{23} = 11$, $U_{24} = 12$ ve $U_{34} = 12$ olarak hesaplanır. Buradan JT test istatistiğinin hesaplanan değeri

$$JT = 11+12+13+11+12+12 = 71$$

olarak bulunur.

Daha önce de ifade edildiği gibi JT istatistiği tüm mümkün *i* ve *j* ($i < j$) grupları üzerinden elde edilen U istatistiklerinin toplamıdır. Görüldüğü gibi her bir U_{ij} istatistiğinin hesaplanmasında $n_i \times n_j$ tane sorgulama ve JT istatistiğinin hesaplanmasında ise $\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} n_i \times n_j$ tane sorgulama yapmak gerekmektedir. Buna göre Çizelge 3.1'deki veri seti için $(4 \times (4-1)/2) \times 4 \times 4 = 96$ tane sorgulama yapılmaktadır. Bu sayı grup sayısı ve örneklem büyüklükleri arttıkça hızlı bir şekilde artmaktadır. Özellikle gen açıklama verileri gibi yüksek boyutlu veriler söz konusu olduğunda hesaplama zamanı oldukça maliyetli bir hale gelmektedir. Bu durumun üstesinden gelmek için Lin ve diğerleri (2019) JT istatistiğinin hesaplanması için hızlı JT algoritmasını önermişlerdir. Algoritma uygulanmadan önce her bir grup içinde gözlem değerlerinin küçükten büyüğe doğru sıralanması gerekmektedir. Algoritmaya göre her bir U_{ij} değerinin hesaplanmasında Çizelge 3.2'de verilen adımlar uygulanmaktadır.

Çizelge 3.2. U_{ij} değerlerinin hesaplanması için hızlandırılmış JT algoritması

Algoritma: U_{ij} istatistiğinin hesaplanması
1: $l \leftarrow$ *i.* grup için başlangıç indeksi
2: $m \leftarrow$ *j.* grup için başlangıç indeksi
3: while $l<n_i$ and $m<n_j$
4: if ($X_{il} < X_{jm}$) then
5: $U \leftarrow U + n_j - m + 1$
6: $l \leftarrow l + 1$

Çizelge 3.2. (devam) U_{ij} değerlerinin hesaplanması için hızlandırılmış JT algoritması

```
7:    else if ( X_il = X_jm ) then
8:        a'yı 1'den başlatarak X_i(l+a)!=X_il şartı sağlanana kadar arttır.
9:        b'yi 1'den başlatarak X_j(m+b)!=X_jm şartı sağlanana kadar arttır.
10:       U ← U + 0.5(a + 1)(b + 1)
11:       l ← l + a
12:       m ← m + b
13:   else if ( X_il > X_jm ) then
14:       m ← m + 1
15:   end if
16: end while
17: return U
```

Algoritmanın mantığı, sıralı iki vektör alındığında ilk vektörün *i*. gözlemi ikinci vektörün *j*. gözleminden küçük ise (*j*+1)., (*j*+2).,… gözlemlerinden de küçük olma durumuna dayalıdır. Bu durumda, ilk vektörün i. gözlemi ile (*j*)., (*j*+1)., (*j*+2).,… gözlemlerin her birini kıyaslamak yerine hepsinin sıralı örüntüye uyduğunu kabul ederek tek bir toplama işlemi yapmak işlem sayısını önemli ölçüde azaltmaktadır.

Çizelge 3.1'den görüldüğü gibi her bir grupta veriler küçükten büyüğe doğru sıralıdır. Daha önce de söz edildiği gibi JT istatistiği tüm mümkün *i* ve *j* ($i < j$) grupları üzerinden elde edilen U istatistiklerinin toplamıdır. Klasik yaklaşımla U istatistiğinin elde edilmesinde örneğin I ve II. gruplar için toplam 4×4=16 tane kıyaslama yapmak gerekmektedir.

U_{12} değerinin hesaplanması,

$$\begin{aligned} U_{12} = {} & I(19<21)+I(19<61)+I(19<80)+I(19<129) \\ & +I(20<21)+I(20<61)+I(20<80)+I(20<129) \\ & +I(60<21)+I(60<61)+I(60<80)+I(60<129) \\ & +I(130<21)+I(130<61)+I(130<80)+I(130<129) \end{aligned}$$

biçiminde ifade edilir. Burada sorgulamaların sonuçları toplamı,

$$U_{12} = 1+1+1+1+1+1+1+1+0+1+1+1+0+0+0+0 = 11$$

olarak ifade edilir.

Önerilen algoritmada ise sadece 8 karşılaştırma ile U_{12} istatistiği elde edilebilmektedir. Örneğin, $19 < 21$ eşitsizliği sağlandığı için algoritma $19 < 61$, $19 < 80$ ve $19 < 129$ karşılaştırmalarını yapmamaktadır. Çünkü 21 ikinci grubun zaten en küçük elemanıdır. Eğer $19 < 21$ oluyorsa $19 < 61$, $19 < 80$ ve $19 < 129$ eşitsizlikleri de sağlanacaktır. Bu durumda algoritma, 21 ve sonrasında gelen gözlemlerin sayısını U istatistiğine eklemektedir. Bu mantığı birinci gruptaki her bir gözlem için sırayla tekrarlayarak test istatistiği daha az sorgulama ile elde edilebilmektedir. Önerilen algoritma uygulandığında yapılan sorgulamalar ve U_{12} istatistiğinin değeri Çizelge 3.3'de verilmiştir.

Çizelge 3.3. U_{12} istatistiğinin hesaplanması

Sorgulama	Sonuç	U_{12}
19 < 21	Doğru	4
20 < 21	Doğru	4 + 4 = 8
60 < 21	Yanlış	8
60 < 61	Doğru	8 + 3 = 11
130 < 21	Yanlış	11
130 < 61	Yanlış	11
130 < 80	Yanlış	11
130 < 129	Yanlış	11

Çizelge 3.3'de görüldüğü gibi sorgulama doğru olduğunda ikinci grup için eşitsizliğin sağındaki sayıya eşit veya büyük olan gözlemlerin sayısı bulunur ve U istatistiğine eklenir. Aksi halde eşitsizliğin sağındaki sayı bir sonraki gözlem değeri ile değiştirilmekte ve U değeri değişmemektedir. Bu işlemler birinci grubun (eşitsizliğin sol tarafı) her bir değeri için tekrarlanır. Görüldüğü üzere algoritmanın uygulanması ile sorgulama sayısı ve bu sayede işlem yükü önemli ölçüde azalmaktadır. Özellikle gen çalışmalarındaki verilerin büyüklüğü dikkate alındığında önerilen algoritmanın etkisi daha da artmaktadır. Bu algoritma, Lin ve diğerleri (2019) tarafından `fastJT` adında bir R paketi olarak kodlanmıştır. Bu paket, aynı zamanda çapraz doğrulama (cross validation) için *n*-kat (*n*-fold) fonksiyonunu da içermektedir.

3.2. S Testi

Shan ve diğerleri (2014) gözlemlerin büyüklüklerini de dikkate alacak şekilde S sıralı alternatif testini önermişlerdir. S testinde $I(X_{il} < X_{jm})$ ifadesi R_{jm} ve R_{il} sıra sayıları arasındaki fark ile ağırlıklandırılmaktadır. Bu sayede önerilen test istatistiği gözlem değerlerinin büyüklüklerine karşı da duyarlı olmaktadır. Ayrıca gözlem değerlerinin yerine sıra sayılarını dikkate aldığı için aykırı değerlere ve dağılımdaki çarpıklıklara karşı sağlam (robust) olma özelliğini korumaktadır. S istatistiğinin hesaplanmasında önce birleştirilmiş örnekleme (grup farkı olmaksızın) sıra sayıları atanır ve daha sonra aşağıdaki formül uygulanmaktadır.

$$S = \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} D_{ij},$$

$$D_{ij} = \sum_{l=1}^{n_i} \sum_{m=1}^{n_j} Z_{ijlm} \tag{3.10}$$

$$Z_{ijlm} = (R_{jm} - R_{il}) I(X_{jm} > X_{il}). \tag{3.11}$$

Eş. 3.11'deki $R_{il}(R_{jm})$ birleştirilmiş verilerdeki $X_{il}(X_{jm})$ gözlemine karşılık gelen sıra sayısıdır.

H_0 hipotezinin doğruluğu altında S istatistiği, Eş. 3.12'deki ortalama ve Eş. 3.13'deki varyans ile normal dağılıma sahiptir.

$$E(S) = \frac{N+1}{6} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} n_i n_j \tag{3.12}$$

$$V(S) = \frac{N^2 + N}{12} - \frac{(N+1)^2}{36} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} n_i n_j$$

$$+2\left[\sum_{i=1}^{k-1} n_i \left(\sum_{j=i+1}^{k} n_j \right)_2 + \sum_{i=2}^{k} n_i \left(\sum_{j=2}^{i-1} n_j \right)_2 \right] CovA,$$

$$+2\left(\sum_{i=1}^{k-2} \sum_{j=i+1}^{k-1} \sum_{l=j+1}^{k} n_i n_j n_l \right) CovB. \tag{3.13}$$

Eş. 3.13'de verilen, $CovA = \frac{2N^2 + N - 1}{90}$ ve $CovB = \frac{-7N^2 - 11N - 4}{360}$ biçiminde tanımlıdır.

Çizelge 3.4'deki veri kümesi üzerinde S test istatistiğinin hesaplanması aşağıda verilmiştir.

Çizelge 3.4. Sıra sayılarıyla Jonckheere verisi (Jonckheere, 1954)

I	II	III	IV
19 (1)	21 (3)	40 (4)	49 (5)
20 (2)	61 (7)	99 (9)	110 (11)
60 (6)	80 (8)	100 (10)	151 (15)
130 (13)	129 (12)	149 (14)	160 (16)

Çizelge 3.4'de parantez içinde verilen sayıları birleştirilmiş örneğe atanan sıra sayılarını göstermektedir. Bu veri seti için

$$D_{12} = \sum_{l=1}^{n_i=4} \sum_{m=1}^{n_j=4} Z_{12lm} = (R_{2m} - R_{1l}) I(X_{2m} > X_{1l})$$

biçiminde yazılır.

Benzer olarak, $D_{13} = 78$, $D_{14} = 111$, $D_{23} = 48$, $D_{24} = 81$ ve $D_{34} = 61$ olarak elde edilir. Bu değerler aşağıdaki formülde yerine yazıldığında,

$$\begin{aligned} S &= \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} D_{ij} \\ &= D_{12} + D_{13} + D_{14} + D_{23} + D_{24} + D_{34} \\ &= 57 + 78 + 111 + 48 + 81 + 61 = 436 \end{aligned}$$

olarak bulunur.

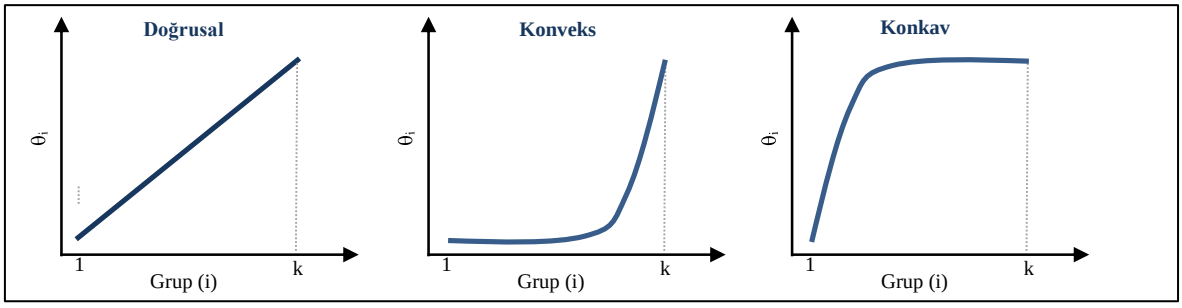
3.3. Hızlandırılmış S Testi

3.3.1. Motivasyon

Sıralı alternatif testlerinde konum parametreleri için karşıt hipotez Eş. 3.14'teki gibi verilir.

$$H_1 : \theta_1 \le \theta_2 \le ... \le \theta_k \quad (\theta_1 < \theta_k). \tag{3.14}$$

Yokluk hipotezinin reddedilmesi durumunda karşıt hipotez Şekil 3.1'de verildiği gibi doğrusal, konveks veya konkav olmak üzere farklı örüntü yapısına sahip olabilir.



Şekil 3.1. Doğrusal, konveks ve konkav alternatif hipotez yapıları

Örneğin $k = 3$ şöyle ki 3 grup (düzey) olduğunda konum parametreleri $(\theta_1; \theta_2; \theta_3)$ doğrusal, konveks ve konkav alternatif hipotezler için sırasıyla (0; 0,5; 1), (0; 0; 1) ve (0; 1; 1) biçiminde verilebilir. Dolayısıyla sıralı alternatif testleri değerlendirilirken farklı alternatif test durumlarının da dikkate alınması gerekir. Literatür çalışmaları dikkate alındığında alternatif hipotezin farklı örüntü yapılarına göre S istatistiğinin JT istatistiğine göre daha güçlü olduğu görülmüştür. Örneğin, Shan ve diğerleri (2014) farklı alternatif hipotez örüntülerini, farklı dağılımları ve farklı örneklem büyüklüklerini dikkate alarak yaptıkları simülasyon çalışmasında S testinin genel olarak JT testinden daha güçlü olduğunu göstermişlerdir. Benzer sonuçlar, çok sayıda sıralı alternatif testinin karşılaştırılması için Altunkaynak ve Gamgam (2020) tarafından yapılan çalışmada da yer almaktadır.

Yüksek boyutlu verilerde çok fazla sayıda değişken dikkate alınacağından farklı alternatif hipotez yapılarının ve farklı dağılımların ortaya çıkma olasılığı daha fazla olacaktır. Bu nedenle yüksek boyutlu verilerde sıralı örüntüleri tespit etmek için S istatistiğinin kullanımının daha uygun olacağı söylenebilir. Ancak, JT testinde olduğu gibi S testinde de hesaplama zamanı bir problem olarak durmaktadır. Çünkü S testinde de JT testinde olduğu kadar sorgulama yapmak gerekmektedir. Bu nedenle bu tez çalışmasında hızlandırılmış S testi önerilmektedir.

3.3.2. Algoritma

Test istatistiğinin elde edilmesinden önce her bir grupta yer alan değerlerin küçükten büyüğe doğru sıralanması gerekmektedir. Test istatistiği için *i*. ve *j*. gruplarına ait D_{ij} değerinin elde edilmesine ilişkin algoritmanın adımları aşağıda verilmektedir.

Çizelge 3.5. D_{ij} değerlerinin hesaplanması için hızlandırılmış S algoritması

Algoritma: D_{ij} istatistiğinin hesaplanması
$l \leftarrow i$. grup için başlangıç indeksi
$m \leftarrow j$. grup için başlangıç indeksi
while $l<n_i$ and $m<n_j$
if ($X_{il} < X_{jm}$) then
$k \leftarrow m$
$R \leftarrow 0$
while $k<n_j$
$R \leftarrow R + R_{jk}$
$k \leftarrow k + 1$
end while
$D \leftarrow D + R - (n_j - m + 1)R_{il}$
$l \leftarrow l + 1$
else if ($X_{il} > X_{jm}$) then
$m \leftarrow m + 1$
end if
end while
return D

Önerilen algoritmanın mantığı JT algoritmasıyla benzerdir. Çizelge 3.1'de verilen veri için algoritmanın çalışma prensibi aşağıdaki gibi verilebilir.

Çizelge 3.6. D_{12} değerinin hesaplanması

Sorgulama	Sonuç	D_{12}
19 < 21	Doğru	30 – 4(1) = 26
20 < 21	Doğru	26 + 30 – 4(2) = 48
60 < 21	Yanlış	48
60 < 61	Doğru	48 + 27 – 3(6) = 57
130 < 21	Yanlış	57
130 < 61	Yanlış	57
130 < 80	Yanlış	57
130 < 129	Yanlış	57

Görüldüğü üzere algoritmanın uygulanması ile sorgulama sayısı ve bu sayede işlem yükü önemli ölçüde azalmaktadır. Çizelge 3.6'daki 19 < 21 sorgulaması için hesaplamanın nasıl yapıldığı şu şekilde açıklanabilir. 19 birinci grubun ilk gözlem değeri 21 ise ikinci grubun gözlem değeridir. 19 < 21 eşitsizliği sağlandığından 19 ikinci gruptaki bütün gözlem değerlerinden küçük olacaktır. Bu nedenle ikinci gruba atanan sıra sayılarından 19 değerine atanan sıra sayılarının farkı alınarak toplanmalıdır. Bu toplam $(3-1)+(7-1)+(8-1)+(12-1)$ şeklinde yazılır. Parantezler düzenlendiğinde bu toplam $(3+7+8+12)-4(1)$ yani $30-4(1)=26$ şeklinde ifade edilebilir.

Genel olarak ifade etmek gerekirse $X_{il} < X_{jm}$ sağlandığında $\sum_{k=m}^{n_j} R_{jk} - (n_j + m + 1)R_{il}$ değeri D_{ij}'ye eklenmektedir aksi halde D_{ij} değeri değişmemektedir.

Algoritmanın uygulanabilmesi için `fastS`, `FastSG` ve `generateData` adlarında R fonksiyonları hazırlanmıştır.

4. SİMÜLASYON ÇALIŞMASI

Bu bölümde fastJT ve fastS testlerinin karşılaştırılması için ele alınan simülasyon senaryoları hakkında bilgi verilmiş ve veri üretme sürecine ilişkin detaylar açıklanmıştır.

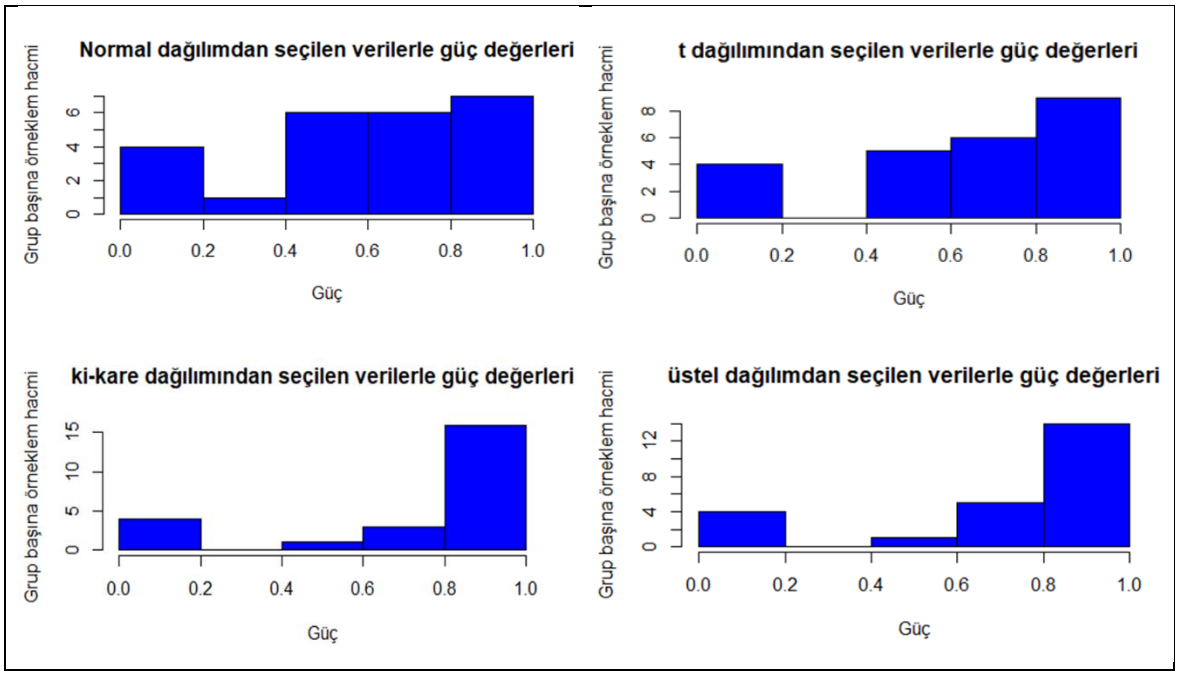
4.1. Simülasyon Senaryoları

Markerlar üzerinde birden fazla SNP'nin örüntü oluşturabilmesi için *k* tane düzeye sahip ilişkili SNP'ler oluşturulmuştur. Bu çalışmada 100 tane SNP içine marker değerlerinde sıralı örüntü oluşturan 3 tane SNP yerleştirilmiştir. Bunun için önce çok değişkenli normal dağılımdan aşağıdaki parametrelere sahip 3 rastgele değişken seçilmiştir.

$$(X_1, X_2, X_3) \sim N_3\left(\begin{bmatrix}10\\5\\2\end{bmatrix}, \begin{bmatrix}1 & 0{,}8 & 0{,}9\\0{,}8 & 1 & 0{,}7\\0{,}9 & 0{,}7 & 1\end{bmatrix}\right) \tag{4.1}$$

Eş. 4.1'deki X_1, X_2, X_3 değişkenleri kendi aralarında ilişkilidir ancak süreklidir. Daha sonra X_1, X_2, X_3 değişkenleri düzey sayısı dikkate alınarak kategorik değişkenlere (SNP1, SNP2, SNP3) dönüştürülmüştür. Bu nedenle yukarıda yer alan ortalama vektörünün bir önemi yoktur isteğe bağlı değerler alınabilir. Her bir SNP'nin her bir düzeyi için marker verileri aşağıda anlatılan şekliyle farklı dağılımlardan rastgele üretilmiştir. Geriye kalan 97 SNP ise her bir düzey eşit olasılıklarla olacak şekilde oluşturularak SNP veri matrisi tamamlanmıştır. SNP veri matrisinde SNP'ler karıştırılmış ve testlerin ilişkili SNP'leri tespit etme durumları incelemiştir.

Marker verilerinin üretilmesinde iki simetrik (Normal ve t) iki de çarpık dağılım (ki-kare ve üstel) dikkate alınmıştır. Şekil 4.1'de 3 düzeyli bir SNP için simülasyon ile elde edilen verilerin dağılımlara ilişkin histogram grafikleri yer almaktadır.



Şekil 4.1. Dağılımlara ilişkin histogram grafikleri

Her bir dağılım için farklı konum parametreleri dikkate alınarak farklı senaryolar üretilmiştir.

Konum parametreleri farklı sıralı alternatif örüntülerini içerecek biçimde doğrusal, konveks ve konkav yapıda oluşturulmuştur. Örnek olarak, 3 düzeyli bir SNP için karşılık gelen marker değişkeninin konum parametreleri doğrusal bir alternatif örüntüsü için (0; 0,2; 0,4), konveks bir alternatif örüntüsü için (0, 0, 1) ve konkav yapı için (0, 1, 1) biçiminde verilebilir.

Her bir düzey için varyansları eşit tutabilmek için belli bir dağılımdan üretilen verilere sabitler eklenmiştir. Bu sayede istenilen örüntülere sahip konum parametreleri için dağılımlar oluşturulmuştur. Örneğin, üç düzeyli bir özellik için Normal(0,1)+a şeklinde verilen bir dağılım için önce Normal(0,1) dağılımından rastgele sayılar seçilmiş daha sonra her bir düzey için sırasıyla 0; 0,2 ve 0,4 sabitleri eklenerek konum parametreleri θ=(0; 0,2; 0,4) olan senaryo oluşturulmuştur. Çalışmada yer alan senaryolar Çizelge 4.1'de verilmiştir.

Çizelge 4.1. Simülasyon çalışmasında kullanılan senaryolar

Grup sayısı	Dağılım	Sabit (a)	Konum Parametreleri	Örneklem Büyüklüğü
3	Normal(0,1)+a	(0, 0, 0)	θ=(0, 0, 0)	(5, 5, 5)
	t_3+a	(0; 0,2; 0,4)	θ=(0; 0,2; 0,4)	(10, 10, 10)
		(0; 0,5; 0,5)	θ=(0; 0,5; 0,5)	(15, 15, 15)
		(0, 1, 1)	θ=(0, 1, 1)	(20, 20, 20)
		(0, 0, 1)	θ=(0, 0, 1)	
		(0, 0, 2)	θ=(0, 0, 2)	
	Ki-kare(1)+a	(0, 0, 0)	θ=(1, 1, 1)	
	Üstel(1)+a	(0; 0,2; 0,4)	θ=(1; 1,2; 1,4)	
		(0; 0,5; 0,5)	θ=(1; 1,5; 1,5)	
		(0, 1, 1)	θ=(1, 2, 2)	
		(0, 0, 1)	θ=(1, 1, 2)	
		(0, 0, 2)	θ=(1, 1, 3)	
4	Normal(0,1)+a	(0, 0, 0, 0)	θ=(0, 0, 0, 0)	
	t_3+a	(0; 0,2; 0,4; 0.8)	θ=(0; 0,2; 0,4; 0,8)	
		(0; 0,5; 0,5; 0,5)	θ=(0; 0,5; 0,5; 0,5)	
		(0; 0,5; 1; 1)	θ=(0; 0,5; 1; 1)	
		(0, 0, 0, 1)	θ=(0, 0, 0, 1)	
		(0, 0, 1, 1)	θ=(0, 0, 1, 1)	
	Ki-kare(1)+a	(0, 0, 0, 0)	θ=(1, 1, 1, 1)	
	Üstel(1)+a	(0; 0,2; 0,4; 0,8)	θ=(1; 1,2; 1,4; 1,8)	
		(0; 0,5; 0,5; 0,5)	θ=(1; 1,5; 1,5; 1,5)	
		(0; 0,5; 1; 1)	θ=(1; 1,5; 2; 2)	
		(0, 0, 0, 1)	θ=(1, 1, 1, 2)	
		(0, 0, 1, 1)	θ=(1, 1, 2, 2)	

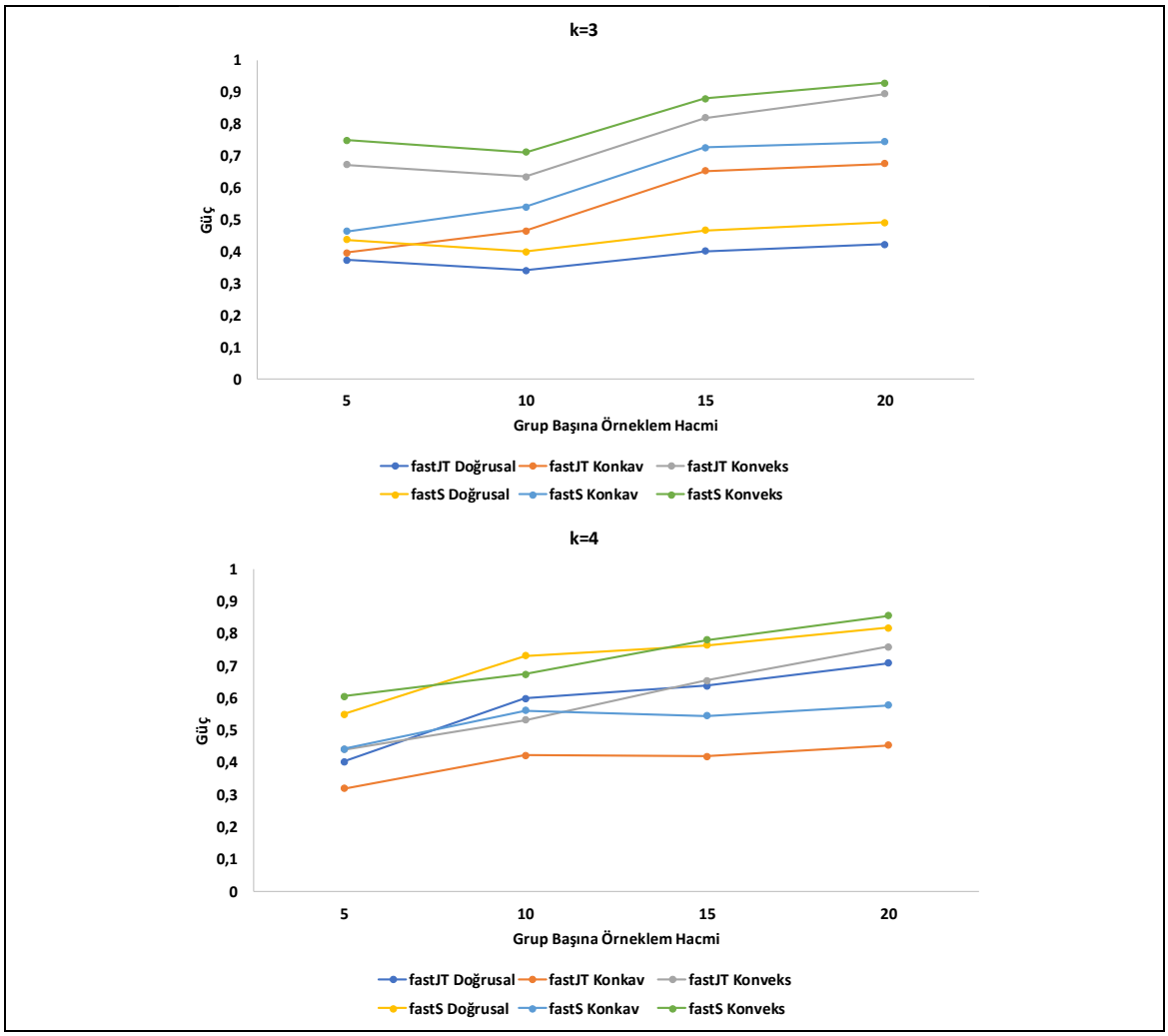
Bu senaryolar altında fastJT ve fastS testlerinin SNP1, SNP2 ve SNP3 özelliklerini tespit etme sayıları dikkate alınarak %5 anlamlı düzeyinde güç değerleri hesaplanmıştır. H_0 doğruyken reddetme olasılıkları yani deneysel hatalar her iki test için de %5'e yakın olduğu sonuçlarda verilmemiştir. Simülasyon çalışmasına ait R kodları EK-1, EK-2 VE EK-3'de verilmiştir.

4.2. Simülasyon Sonuçları

Çizelge 4.2. Normal(0,1)+a dağılımından seçilen verilerle güç değerleri

Konum Parametreleri (a)	Örneklem büyüklüğü	fastJT	fastS
(0, 0, 0)	(5, 5, 5)	0,0455	0,0507
	(10, 10 10)	0,0469	0,0506
	(15, 15, 15)	0,0453	0,0496
	(20, 20, 20)	0,0469	0,0510
(0; 0,2; 0,4)	(5, 5, 5)	0,3741	0,4370
	(10, 10 10)	0,3402	0,3992
	(15, 15, 15)	0,4019	0,4671
	(20, 20, 20)	0,4231	0,4915
(0; 0,5; 0,5)	(5, 5, 5)	0,3968	0,4631
	(10, 10, 10)	0,4646	0,5407
	(15, 15, 15)	0,6528	0,7273
	(20, 20, 20)	0,6760	0,7444
(0, 1, 1)	(5, 5, 5)	0,6981	0,7761
	(10, 10, 10)	0,5158	0,5995
	(15, 15, 15)	0,7190	0,7933
	(20, 20, 20)	0,7892	0,8457
(0, 0, 1)	(5, 5, 5)	0,6735	0,7500
	(10, 10, 10)	0,6361	0,7116
	(15, 15, 15)	0,8211	0,8802
	(20, 20, 20)	0,8959	0,9304
(0, 0, 2)	(5, 5, 5)	0,9795	0,9938
	(10, 10, 10)	0,8227	0,8868
	(15, 15, 15)	0,8371	0,8912
	(20, 20, 20)	0,8450	0,8919
(0, 0, 0, 0)	(5, 5, 5, 5)	0,0468	0,0572
	(10, 10, 10, 10)	0,0443	0,0534
	(15, 15, 15, 15)	0,0454	0,0547
	(20, 20, 20, 20)	0,0455	0,0542
(0; 0,2; 0,4; 0,8)	(5, 5, 5, 5)	0,4027	0,5503
	(10, 10, 10, 10)	0,5988	0,7328
	(15, 15, 15, 15)	0,6384	0,7640
	(20, 20, 20, 20)	0,7087	0,8183
(0; 0,5; 0,5; 0,5)	(5, 5, 5, 5)	0,3197	0,4426
	(10, 10, 10, 10)	0,4222	0,5614
	(15, 15, 15, 15)	0,4188	0,5459
	(20, 20, 20, 20)	0,4537	0,5781
(0; 0,5; 1, 1)	(5, 5, 5, 5)	0,5151	0,6754
	(10, 10, 10, 10)	0,6747	0,7952
	(15, 15, 15, 15)	0,7434	0,8446
	(20, 20, 20, 20)	0,8807	0,9379
(0, 0, 0, 1)	(5, 5, 5, 5)	0,4410	0,6055
	(10, 10, 10, 10)	0,5332	0,6741
	(15, 15, 15, 15)	0,6556	0,7814
	(20, 20, 20, 20)	0,7596	0,8560
(0, 0, 1, 1)	(5, 5, 5, 5)	0,4866	0,6467
	(10, 10, 10, 10)	0,6952	0,8162
	(15, 15, 15, 15)	0,8569	0,9250
	(20, 20, 20, 20)	0,7684	0,8534

Normal dağılımından seçilen verilerle güç değerleri incelendiğinde tüm senaryolar için fastS istatistiğine ait güç değerlerinin fastJT istatistiğinden daha yüksek olduğu görülmektedir. Bu durum Şekil 4.1'de daha açık bir şekilde görülmektedir. Özellikle küçük örneklem durumunda fastS testi daha belirgin şekilde iyi sonuçlar vermektedir. Örneğin, (0; 0,5; 0,5; 0,5) şeklindeki konkav sıralı alternatif için örneklem hacmi her bir grup için 5 iken fastS testinin güç değeri fastJT'ye göre %38,4 daha fazladır.

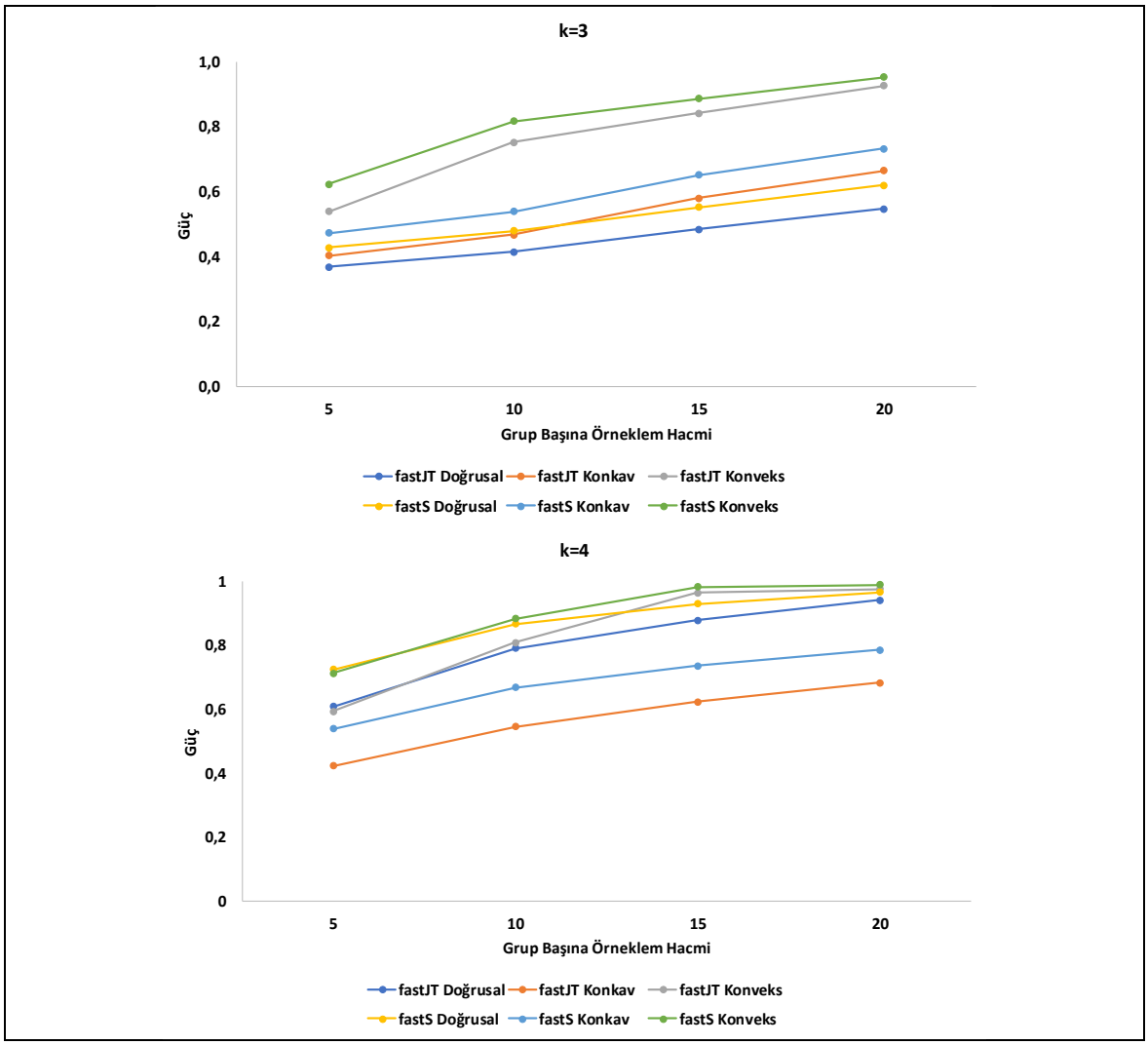


Şekil 4.2. Normal dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri.

Çizelge 4.3. t_3+a dağılımından seçilen verilerle güç değerleri

Konum Parametreleri	Örneklem büyüklüğü	fastJT	fastS
(0, 0, 0)	(5, 5, 5)	0,0444	0,0494
	(10, 10, 10)	0,0454	0,0499
	(15, 15, 15)	0,0459	0,0502
	(20, 20, 20)	0,0453	0,0495
(0; 0,2; 0,4)	(5, 5, 5)	0,3690	0,4277
	(10, 10, 10)	0,4159	0,4796
	(15, 15, 15)	0,4843	0,5532
	(20, 20, 20)	0,5484	0,6203
(0; 0,5; 0,5)	(5, 5, 5)	0,4033	0,4735
	(10, 10, 10)	0,4686	0,5390
	(15, 15, 15)	0,5803	0,6525
	(20, 20, 20)	0,6651	0,7335
(0, 1, 1)	(5, 5, 5)	0,6484	0,7279
	(10, 10, 10)	0,6723	0,7317
	(15, 15, 15)	0,8414	0,8904
	(20, 20, 20)	0,8914	0,9271
(0, 0, 1)	(5, 5, 5)	0,5402	0,6232
	(10, 10, 10)	0,7536	0,8176
	(15, 15, 15)	0,8433	0,8876
	(20, 20, 20)	0,9267	0,9546
(0, 0, 2)	(5, 5, 5)	0,8333	0,8917
	(10, 10, 10)	0,9119	0,9473
	(15, 15, 15)	0,9762	0,9853
	(20, 20, 20)	0,9999	1
(0, 0, 0, 0)	(5, 5, 5, 5)	0,0466	0,0567
	(10, 10, 10, 10)	0,0453	0,0546
	(15, 15, 15, 15)	0,0452	0,0549
	(20, 20, 20, 20)	0,0453	0,0549
(0; 0,2; 0,4; 0,8)	(5, 5, 5, 5)	0,6098	0,7251
	(10, 10, 10, 10)	0,7915	0,8685
	(15, 15, 15, 15)	0,8799	0,9313
	(20, 20, 20, 20)	0,9426	0,9677
(0; 0,5; 0,5; 0,5)	(5, 5, 5, 5)	0,4231	0,5390
	(10, 10, 10, 10)	0,5460	0,6693
	(15, 15, 15, 15)	0,6233	0,7360
	(20, 20, 20, 20)	0,6839	0,7857
(0; 0,5; 1, 1)	(5, 5, 5, 5)	0,6348	0,7266
	(10, 10, 10, 10)	0,8958	0,9387
	(15, 15, 15, 15)	0,9483	0,9714
	(20, 20, 20, 20)	0,9799	0,9900
(0, 0, 0, 1)	(5, 5, 5, 5)	0,5936	0,7130
	(10, 10, 10, 10)	0,8103	0,8838
	(15, 15, 15, 15)	0,9650	0,9830
	(20, 20, 20, 20)	0,9773	0,9912
(0, 0, 1, 1)	(5, 5, 5, 5)	0,8348	0,9099
	(10, 10, 10, 10)	0,9246	0,9527
	(15, 15, 15, 15)	0,9864	0,9936
	(20, 20, 20, 20)	0,9772	0,9888

t dağılımından seçilen verilerle güç değerleri incelendiğinde tüm senaryolar için fastS istatistiğine ait güç değerlerinin fastJT istatistiğinden daha yüksek olduğu Şekil 4.2'den de açıkça görülmektedir. Özellikle küçük örneklem durumunda fastS testi daha belirgin şekilde iyi sonuçlar vermektedir. Örneğin, (0; 0,5; 0,5; 0,5) şeklindeki konveks sıralı alternatif için örneklem hacmi her bir grup için 5 iken fastS testinin güç değeri fastJT'ye göre %27,4 daha fazladır.

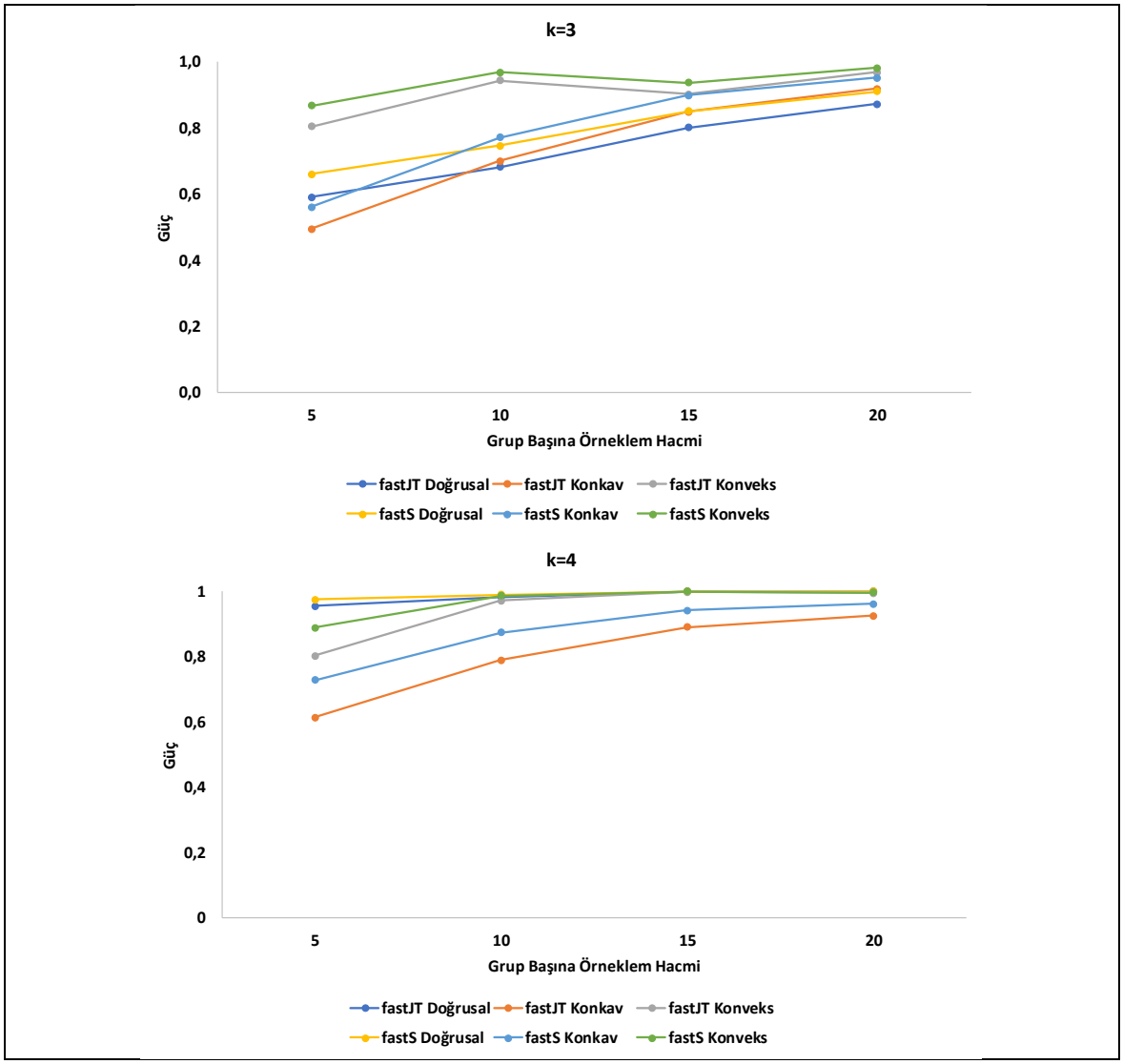


Şekil 4.3. t dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri.

Çizelge 4.4. Ki-kare(1)+a dağılımından seçilen verilerle güç değerleri

Konum Parametreleri	Örneklem büyüklüğü	fastJT	fastS
(1, 1, 1)	(5, 5, 5)	0,0451	0,0500
	(10, 10, 10)	0,0455	0,0500
	(15, 15, 15)	0,0456	0,0499
	(20, 20, 20)	0,0457	0,0497
(1; 1,2; 1,4)	(5, 5, 5)	0,5911	0,6607
	(10, 10, 10)	0,6821	0,7463
	(15, 15, 15)	0,8008	0,8511
	(20, 20, 20)	0,8724	0,9115
(1; 1,5; 1,5)	(5, 5, 5)	0,4946	0,5610
	(10, 10, 10)	0,6999	0,7708
	(15, 15, 15)	0,8503	0,8992
	(20, 20, 20)	0,9196	0,9517
(1, 2, 2)	(5, 5, 5)	0,7890	0,8573
	(10, 10, 10)	0,9422	0,9664
	(15, 15, 15)	0,9865	0,9941
	(20, 20, 20)	0,9876	0,9942
(1, 1, 2)	(5, 5, 5)	0,8051	0,8685
	(10, 10, 10)	0,9448	0,9692
	(15, 15, 15)	0,9024	0,9372
	(20, 20, 20)	0,9682	0,9824
(1, 1, 3)	(5, 5, 5)	0,8173	0,9031
	(10, 10, 10)	0,9533	0,9700
	(15, 15, 15)	0,9988	0,9993
	(20, 20, 20)	0,9999	1
(1, 1, 1, 1)	(5, 5, 5, 5)	0,0450	0,0550
	(10, 10, 10, 10)	0,0465	0,0563
	(15, 15, 15, 15)	0,0449	0,0541
	(20, 20, 20, 20)	0,0459	0,0546
(1; 1,2; 1,5; 2)	(5, 5, 5, 5)	0,9549	0,9761
	(10, 10, 10, 10)	0,9832	0,9906
	(15, 15, 15, 15)	0,9998	0,9998
	(20, 20, 20, 20)	0,9996	0,9996
(1; 1,5; 1,5; 1,5)	(5, 5, 5, 5)	0,6133	0,7288
	(10, 10, 10, 10)	0,7903	0,8745
	(15, 15, 15, 15)	0,8903	0,9422
	(20, 20, 20, 20)	0,9256	0,9621
(1; 1,5; 2; 2)	(5, 5, 5, 5)	0,9202	0,9607
	(10, 10, 10, 10)	0,9945	0,9973
	(15, 15, 15, 15)	0,9981	0,9992
	(20, 20, 20, 20)	0,9998	0,9999
(1, 1, 1, 2)	(5, 5, 5, 5)	0,8037	0,8891
	(10, 10, 10, 10)	0,9716	0,9871
	(15, 15, 15, 15)	0,9988	0,9996
	(20, 20, 20, 20)	0,9948	0,9965
(1, 1, 2, 2)	(5, 5, 5, 5)	0,8547	0,9151
	(10, 10, 10, 10)	0,9837	0,9897
	(15, 15, 15, 15)	0,9983	0,9992
	(20, 20, 20, 20)	0,9997	0,9998

Ki-kare dağılımından seçilen verilerle güç değerleri incelendiğinde tüm senaryolar için fastS istatistiğine ait güç değerlerinin fastJT istatistiğinden daha yüksek olduğu Şekil 4.3'den de açıkça görülmektedir. Ancak, normal ve t dağılımlarından elde edilen sonuçlarla kıyaslandığında farkların daha düşük olduğu söylenebilir. Bununla birlikte yine küçük örneklem durumunda fastS testi daha belirgin şekilde iyi sonuçlar vermektedir.

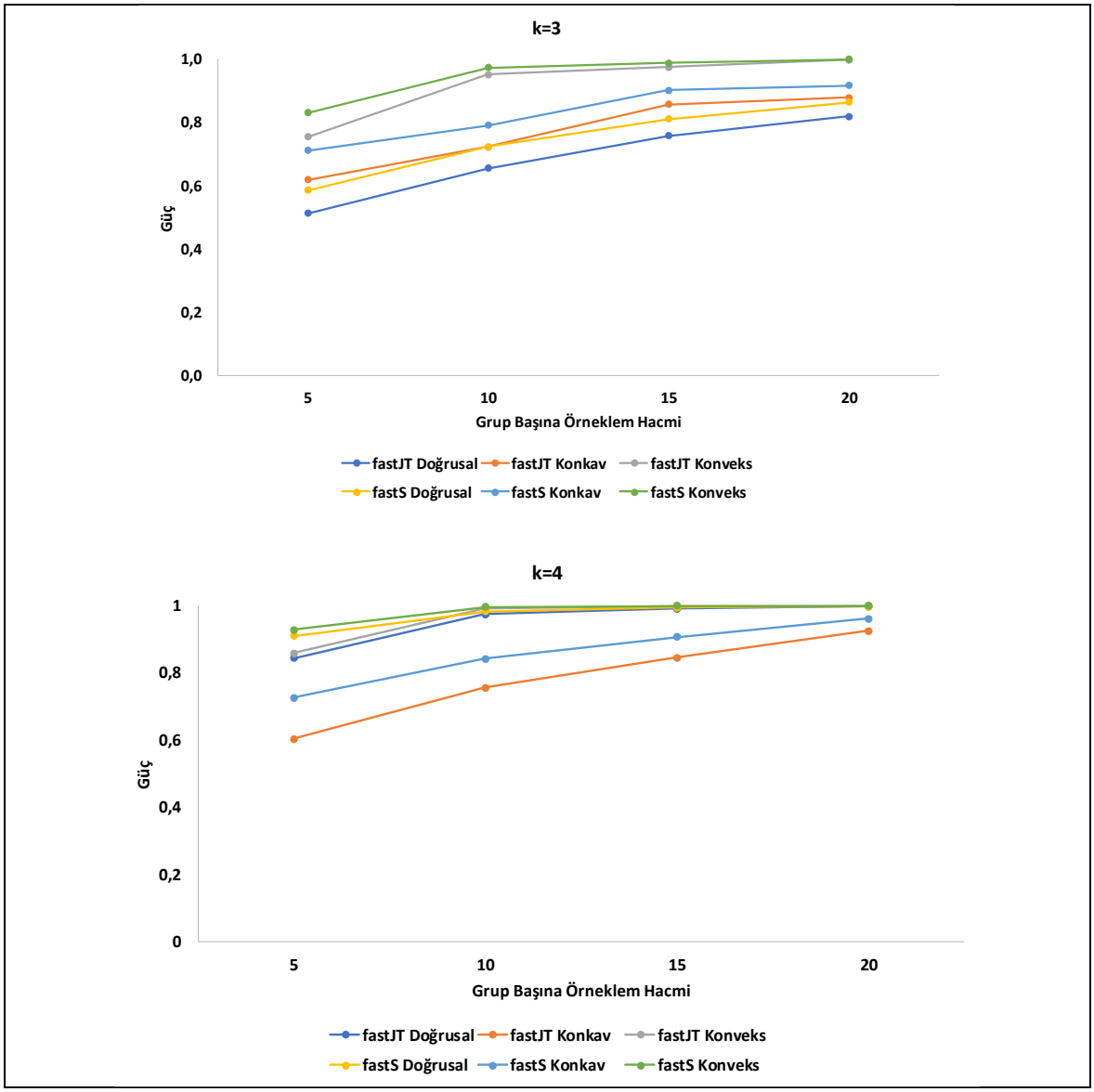


Şekil 4.4. Ki-kare dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri.

Çizelge 4.5. Üstel(1)+a dağılımından seçilen verilerle güç değerleri

Konum Parametreleri	Örneklem büyüklüğü	fastJT	fastS
(1, 1, 1)	(5, 5, 5)	0,0457	0,0514
	(10, 10, 10)	0,0452	0,0504
	(15, 15, 15)	0,0468	0,0513
	(20, 20, 20)	0,0457	0,0500
(1; 1,2; 1,4)	(5, 5, 5)	0,5131	0,5855
	(10, 10, 10)	0,6553	0,7237
	(15, 15, 15)	0,7582	0,8108
	(20, 20, 20)	0,8193	0,8645
(1; 1,5; 1,5)	(5, 5, 5)	0,6190	0,7111
	(10, 10, 10)	0,7233	0,7910
	(15, 15, 15)	0,8573	0,9024
	(20, 20, 20)	0,8794	0,9181
(1, 2, 2)	(5, 5, 5)	0,6774	0,7450
	(10, 10, 10)	0,9150	0,9543
	(15, 15, 15)	0,9885	0,9953
	(20, 20, 20)	0,9762	0,9877
(1, 1, 2)	(5, 5, 5)	0,7543	0,8305
	(10, 10, 10)	0,9527	0,9742
	(15, 15, 15)	0,9752	0,9882
	(20, 20, 20)	0,9988	0,9997
(1, 1, 3)	(5, 5, 5)	0,7674	0,7874
	(10, 10, 10)	0,9981	0,9995
	(15, 15, 15)	0,9998	1
	(20, 20, 20)	1	1
(1, 1, 1, 1)	(5, 5, 5, 5)	0,0457	0,0550
	(10, 10, 10, 10)	0,0451	0,0541
	(15, 15, 15, 15)	0,0435	0,0531
	(20, 20, 20, 20)	0,0459	0,0552
(1; 1,2; 1,4; 1,8)	(5, 5, 5, 5)	0,8440	0,9116
	(10, 10, 10, 10)	0,9757	0,9841
	(15, 15, 15, 15)	0,9926	0,9961
	(20, 20, 20, 20)	0,9988	0,9992
(1; 1,5; 1,5; 1,5)	(5, 5, 5, 5)	0,6040	0,7261
	(10, 10, 10, 10)	0,7567	0,8437
	(15, 15, 15, 15)	0,8465	0,9083
	(20, 20, 20, 20)	0,9262	0,9619
(1; 1,5; 2, 2)	(5, 5, 5, 5)	0,8350	0,8888
	(10, 10, 10, 10)	0,9978	0,9992
	(15, 15, 15, 15)	0,9991	0,9996
	(20, 20, 20, 20)	0,9998	0,9999
(1, 1, 1, 2)	(5, 5, 5, 5)	0,8602	0,9292
	(10, 10, 10, 10)	0,9933	0,9978
	(15, 15, 15, 15)	0,9993	0,9997
	(20, 20, 20, 20)	0,9996	0,9998
(1, 1, 2, 2)	(5, 5, 5, 5)	0,9780	0,9906
	(10, 10, 10, 10)	0,9976	0,9992
	(15, 15, 15, 15)	0,9997	0,9998
	(20, 20, 20, 20)	0,9999	0,9999

Üstel dağılımından seçilen verilerle güç değerleri incelendiğinde tüm senaryolar için fastS istatistiğine ait güç değerlerinin fastJT istatistiğinden daha yüksek olduğu Şekil 4.4'de de görülmektedir. Özellikle küçük örneklem durumunda fastS testi daha iyi sonuçlar vermektedir. Örneğin, (1; 1,5; 1,5; 1,5) şeklindeki konkav sıralı alternatif için örneklem hacmi her bir grup için 5 iken fastS testinin güç değeri fastJT'ye göre %20,2 daha fazladır.



Şekil 4.5. Üstel dağılımdan gelen verilerle farklı alternatif hipotez durumları için fastJT ve fastS testlerinin güç değerleri.

5. SONUÇ VE ÖNERİLER

Bu çalışmada, S test istatistiği yüksek boyutlu veri yapısına sahip GWAS verilerinde sıralı alternatiflerin tespiti için uyarlanmıştır. Önerilen fastS istatistiği ile mevcut fastJT istatistiğinin karşılaştırılması için bir simülasyon çalışması yapılmıştır. Simülasyon çalışmasında gerçek veri yapısına uygun şekilde SNP düzeyindeki artış ile marker değerlerindeki artış örüntüsü farklı senaryolar için üretilmiştir. Tüm senaryolarda simülasyon çalışması yapılarak fastS istatistiğinin fastJT istatistiğine göre daha iyi sonuçlar verdiği gözlemlenmiştir. Ayrıca, fastS istatistiği için R fonksiyonları oluşturulmuştur. Dolayısıyla son yıllarda önemli bir çalışma alanı haline gelen GWAS verilerinde sıralı alternatiflerin tespiti (özellik seçimi) için fastS istatistiği kullanılabilir.

KAYNAKLAR

Allabi, A. C., Gala, J. L. and Horsmans, Y. (2005). CYP2C9, CYP2C19, ABCB1 (MDR1) genetic polymorphisms and phenytoin metabolism in a Black Beninese population. *Pharmacogenetics and Genomics,* 15(11), 779-786.

Altunkaynak, B. and Gamgam, H. (2020). npordtests: An R Package of Nonparametric Tests for Equality of Location Against Ordered Alternatives. *The R Journal*, 12(1),147-171.

Bredalla, M. A., Steinbach, L. S., Morgan, S., Ward, M. and Davis, J. C. (2006). Mri of the sacroiliac joints in patients with moderate to severe ankylosing spondylitis. *American Journal of Roentgenology*, 187(6),1420-1426.

Buning, H., and Kossler, W. (1996). Robustness and efficiency of some tests for ordered alternatives in the c-sample location problem. *Journal of Statistical Computation and Simulation*, 55(4), 337–352.

Cheng, Q., Yang, W., Raimondi, S. C., Pui, C. H., Relling, M. V. and Evans, W. E. (2005). Karyotypic abnormalities create discordance of germline genotype and cancer cell phenotypes. *Nature Genetics*, 37(8),878–882.

Hoffmeyer, S., Burk, O., Von Richter, O., Arnold, H. P., Brockmöller, J., Johne, A., Cascorbi, I., Gerloff, T., Roots, I., Eichelbaum, M. and Brinkmann, U. (2000). Functional polymorphisms of the human multidrug resistance gene: Multiple sequence variations and correlation of one allele with p glycoprotein expression and activity in vivo. *Proceedings of the National Academy of Sciences of the United States of America*, 97(7),3473–3478.

Innocenti, F., Jiang, C., Sibley, A. B., Etheridge, A. S., Hatch, A. J., Denning, S., Niedzwiecki, D., Shterev, I. D., Lin, J., Furukawa, Y., Kubo, M., Kindler, H. L., Auman, J. T., Venook, A. P., Hurwitz, H. I., McLeod, H. L., Ratain, M. J., Gordan, R., Nixon, A. B. & Owzar, K. (2018). Genetic variation determines VEGF-A plasma levels in cancer patients. *Scientific reports*, 8(1), 1-9.

Jonckheere, A. R. (1954). A distribution-free k-sample test against ordered alternatives. *Biometrika,* 41(1), 133–145.

Kawaguchi, T., Sumida, Y., Umemura, A., Matsuo, K., Takahashi, M., Takamura, T., Yasui, K., Saibara, T., Hashimoto, E., Kawanaka, M., Watanabe, S., Kawata, S., Imai, Y., Kokubo, M., Shima, T., Park, H., Tanaka, H., Tajima, K., Yamada, R., Matsuda, F., Okanoue, T. and Japan, D. (2012). Study Group of Nonalcoholic Fatty Liver. Genetic polymorphisms of the human PNPLA3 gene are strongly associated with severity of non-alcoholic fatty liver disease in Japanese. *PloS one*, 7(6),e38322–e38322.

Komatsu, H., Takeuchi, H., Ono, C., Yu, Z., Kikuchi, Y., Kakuto, Y. and Tomita, H. (2021). Association between OLIG2 Gene SNP rs1059004 and negative self-schema constructing trait factors underlying susceptibility. *Frontiers in Psychiatry*, 12, 631475.

Lin, J., Sibley, A., Shterev, I., Andrew, N., Innocenti, F., Chan, C. and Owzar, K. (2019). fastJT: An R package for robust and efficient feature selection for machine learning and genome-wide association studies. *BMC Bioinformatics*, 20, 333.

Lin, J., Sibley, A., Shterev, I. and Owzar, K. (2017). fastJT: Efficient Jonckheere-Terpstra Test statistics for robust machine learning and genome-wide association studies. *R package version 1.0.4*.

Neuhäuser, M., Liu, P. Y. and Hothorn, L. A. (1998). Nonparametric tests for trend: Jonckheere's Test, a modification and a maximum test. *Biometrical Journal*, 40(8), 899–909.

Ong, J. P., Aggarwal, A., Krieger, D., Easley, K. A., Karafa, M. T., Van Lente, F., Arroliga, A. C. and Mullen, D. (2003). Correlation between ammonia levels and the severity of hepatic encephalopathy. *The American Journal of Medicine*, 114(3),188-193.

Rakvag, T. T., Klepstad, P., Baar, C., Kvam, T. M., Dale, O., Kaasa, S., Krokan, H. E. and Skorpen, F. (2005). The val158met polymorphism of the human catechol-o-methyltransferase (comt) gene may influence morphine requirements in cancer pain patients. *Pain*, 116,73–8.

Shan, G. G., Young, D. and Kang, L. (2014). A new powerful nonparametric rank test for ordered alternative problem. *Plos One*, 9(11),1–10.

Takahisa, K., Yoshio, S., Atsushi, U., Keitaro, M., Meiko, T., Toshinari, T., Kohichiroh, Y., Toshiji, S., Etsuko, H., Miwa, K., Sumio, W., Sumio, K., Yasuharu, I., Miki, K., Toshihide, S., Hyohun, P., Hideo, T., Kazuo, T., Ryo, Y., Fumihiko, M. and Takeshi, O. (2012). Genetic polymorphisms of the human pnpla3 gene are strongly associated with severity of non-alcoholic fatty liver disease in japanese. *PLoS ONE*, 7,38322.

Tan, H. L., Zain, S. M., Mohamed, R., Rampal, S., Chin, K. F., Basu, R. C., Cheah, P. L., Mahadeva, S. and Mohamed, Z. (2014). Association of glucokinase regulatory gene polymorphisms with risk and severity of non-alcoholic fatty liver disease: An interaction study with adiponutrin gene. *Journal of Gastroenterology*, 49(6),1056–1064.

Terpstra, J. T. (1952). The asymptotic normality and consistency of Kendall's test against trend, when ties are present in one ranking. *Indagationes Mathematicae*, 14(3),327–333.

Terpstra, J. T., Chang, C. H. and Magerl, R. C. (2011). On the use of Spearman's Correlation Coefficient for testing ordered alternatives. *Journal of Statistical Computation and Simulation*, 81(11),1381–1392.

Terpstra, J. T. and Magel, R. C. (2003). A new nonparametric test for the ordered alternative problem. *Journal of Nonparametric Statistics*, 15(3),289–301.

Uchiyama, T., Kanno, H., Ishitani, K., Fujii, H., Ohta, H., Matsui, H., Kamatani, N. and Saito, K. (2012). An snp in CYP39A1 is associated with severe neutropenia induced by docetaxel. *Cancer Chemotherapy and Pharmacology*, 69(6),1617–1624.

Yorifuji, K., Uemura, Y., Horibata, S., Tsuji, G., Suzuki, Y., Miyagawa, K., Nakayama, K., Hirata, K.-i., Kumagai, S. and Emoto, N. (2018). CHST3 and CHST13 polymorphisms as predictors of bosentan-induced liver toxicity in Japanese patients with pulmonary arterial hypertension. *Pharmacological Research*, 135,259–264.

EKLER

EK-1. fastS.R fonksiyonu: X ve Y değişkenleri için S istatistiğinin hesaplanması

```
fastS<-function (formula, data, alpha = 0.05, na.rm = TRUE, verbose = TRUE)
{
  #Compute Modified S test (Fast S) Statistics
  dp = as.character(formula)
  DNAME <- paste(dp[[2L]], "and", dp[[3L]])
  METHOD <- "Fast S test"
  TEST <- "fastS"
  if (na.rm) {
    completeObs <- complete.cases(data)
    data <- data[completeObs, ]
  }
  if (any(colnames(data) == dp[[3L]]) == FALSE)
    stop("The name of group variable does not match the variable names in the data. The
group variable must be one factor.")
  if (any(colnames(data) == dp[[2L]]) == FALSE)
    stop("The name of response variable does not match the variable names in the data.")

  y = data[, dp[[2L]]]
  r = rank(data[, dp[[2L]]])
  group = data[, dp[[3L]]]
  if (!is.factor(group))
    stop("The group variable must be a factor.")
  if (!is.numeric(y))
    stop("The response must be a numeric variable.")
  n <- length(y)
  x.levels <- levels(factor(group))
  p <- NROW(x.levels)
  y.n <- r.n <- NULL
  Eq1 = Eq2 = Eq3 = S = nfak = sy.n = S=0
for (i in x.levels) {
  y.n[i] <- length(y[group == i])
}

sy.n = cumsum(y.n)
r.n = matrix(c(r, group, y), ncol = 3, nrow = n)

for (i in 1:(p - 1)) {
  for (j in (i + 1):p) {
    nfak = nfak + y.n[i] * y.n[j]
    Eq1 = Eq1 + y.n[i] * choose(y.n[j], 2)
    for (k in (sy.n[i] - y.n[i] + 1):sy.n[i]) {
      for (m in (sy.n[j - 1] + 1):sy.n[j]) {
        if (r.n[k, 3] < r.n[m, 3])
        {
          S = S + sum(r.n[m:sy.n[j], 1])-(sy.n[j]-m+1)*r.n[k, 1]
          break
        }
      }
    }
  }
}
CovA = (2 * n * n + n - 1)/90
CovB = (-7 * n * n - 11 * n - 4)/360
for (i in 2:p) {
  for (j in 1:(i - 1)) {
    Eq2 = Eq2 + y.n[i] * choose(y.n[j], 2)
  }
}
for (i in 1:(p - 2)) {
  for (j in (i + 1):(p - 1)) {
    for (l in (j + 1):p) {
      Eq3 = Eq3 + y.n[i] * y.n[j] * y.n[l]
    }
  }
```

EK-1. (devam) fastS.R fonksiyonu: X ve Y değişkenleri için S istatistiğinin hesaplanması

```
}
ES = (n + 1)/6 * nfak
VS = ((n * n + n)/12 - (n + 1)^2/36) * nfak + 2 * (Eq1 + Eq2) * CovA + 2 * Eq3 *
CovB
Z = (S - ES)/sqrt(VS)
p.value = 1 - pnorm(Z, mean = 0, sd = 1, lower.tail = TRUE,log.p = FALSE)
if (verbose) {
  cat("------------------------------------------------------------",
      "\n", sep = " ")
  cat("  Test :", METHOD, "\n", sep = " ")
  cat("  data :", DNAME, "\n\n", sep = " ")
  cat("  Statistic =", S, "\n", sep = " ")
  cat("  Mean =", ES, "\n", sep = " ")
  cat("  Variance =", VS, "\n", sep = " ")
  cat("  Z =", Z, "\n", sep = " ")
  cat("  Asymp. p-value =", p.value, "\n\n",
      sep = " ")
  cat(if (p.value > alpha) {
    "  Result : Null hypothesis is not rejected."
  }
  else {
    "  Result : Null hypothesis is rejected."
  }, "\n")
  cat("------------------------------------------------------------",
      "\n\n", sep = " ")
}
result <- list()
result$statistic <- S
result$mean <- ES
result$variance <- VS
result$Z <- Z
result$p.value <- p.value
result$alpha <- alpha
result$method <- METHOD
result$data <- data
result$formula <- formula
attr(result, "class") <- "owt"
invisible(result)
}
```

EK-2. fastSG.R fonksiyonu: Genom verisi için S istatistiğinin hesaplanması

```
FastSG<-function (Y, X, outTopN = 15L, numThreads = 1L)
{
  #Modified S test (Fast S) for Genome-Wide Association
  Mark<-Y
  Geno<-X
  zMark=0;k=1;
  num_sample <- dim(Mark)[1]
  nMark<-dim(Mark)[2]
  nGeno<-dim(Geno)[2]
  markerNames <- colnames(Mark)
  SNPNames <- colnames(Geno)
  for (i in 1:nMark)
    for (j in 1:nGeno)
    {
      veri <- data.frame(Geno[,j],Mark[,i])
      colnames(veri)<-c("A","B")
      newdata <- veri[order(veri$A,veri$B),]
      newdata$A<-as.factor(newdata$A)
      zMark[k]<-fastS(B~A,newdata,verbose = FALSE)$Z
      k<-k+1
    }
  MarkZ<-NULL
  for (i in 1:nMark)
  {
    L<-(i-1)*nGeno+1
    U<-i*nGeno
    #MarkZ<-cbind(MarkZ,as.matrix(zMark[L:U],1,1000))
    MarkZ<-cbind(MarkZ,as.matrix(zMark[L:U]))
  }
  rownames(MarkZ)<-SNPNames
  BestJMark<-NULL
  BestSNP<-NULL
  for (i in 1:nMark)
  {
    BestSNP <- cbind(BestSNP, names(MarkZ[order(-abs(MarkZ[,i])),][1:outTopN,i]))
    BestJMark <- cbind(BestJMark, MarkZ[order(-abs(MarkZ[,i])),][1:outTopN,i])
  }
  rownames(BestJMark)<-NULL
  class(res) <- "JTGenome"
  res<-list()
  res$Mark<-BestJMark
  res$SNP<-BestSNP
  attr(res, "class") <- "owt"
  invisible(res)
}
```

EK-3. generateData.R fonksiyonu: Simülasyon tasarımı

```
# Gerekli paketler
#library(fastJT)
#library(MASS)
#library(tidyverse)
#library(GGally)


# Girdi parametreleri
SampleSize=20 # <-- Burayı çizelgeye göre değiştir.
nSNP=100
nMArk=2
REALSNP<-c("SNP:98","SNP:99","SNP:100")
FastJPower<-0
FastSPower<-0
itNum<-10    # 5000 olacak

############################################################
# Yapay verinin üretilmesi 3 düzeyli korelasyonlu 3 SNP için
############################################################

# create the variance covariance matrix and the mean vector
sigma<-rbind(c(1,0.8,0.7), c(0.8,1, 0.9), c(0.7,0.9,1))
mu<-c(10, 5, 2)

# generate the multivariate normal distribution
SNPDataCont<-as.data.frame(mvrnorm(SampleSize, mu=mu, Sigma=sigma))

#################### SIMULATION ################################################
for (it in 1:itNum)
{
# create categorical dependent vectors
SNPData<-SNPDataCont%>%transmute(SNP1= case_when(V1<quantile(V1,0.33)~1,
                                    V1<quantile(V1,0.67)~2,
                                    TRUE~3),
                                SNP2= case_when(V2<quantile(V2,0.33)~1,
                                    V2<quantile(V2,0.67)~2,
                                    TRUE~3),
                                SNP3= case_when(V3<quantile(V3,0.33)~1,
                                    V3<quantile(V3,0.67)~2,
                                    TRUE~3))

# Create random marker data for each SNP
# MARK1
SNPData<-SNPData%>%mutate(Mrk1=case_when(SNP1==1~rnorm(SampleSize,0,1),    #  Buralarda
değişiklik olacak
                                         SNP1==2~rnorm(SampleSize,0.2,1),
                                         SNP1==3~rnorm(SampleSize,0.4,1)))
# MARK2
SNPData<-SNPData%>%mutate(Mrk2=case_when(SNP1==1~rnorm(SampleSize,0,1),
                                         SNP1==2~rnorm(SampleSize,0.2,1),
                                         SNP1==3~rnorm(SampleSize,0.4,1)))


# For chi square distribution --> rchisq(SampleSize, 1, ncp = 0) + sabit
# For exponential distribution --> rexp(SampleSize, rate = 1) + sabit
# For t distribution --> rt(SampleSize, 3, ncp) + sabit
# For normal distribution --> rnorm(SampleSize,0,1) + sabit

#boxplot(SNPData$Mrk1~SNPData$SNP1)

# Create 100-3 SNP

y<-rep(sample( 1:3, SampleSize, replace=TRUE, prob=c(1/3, 1/3, 1/3) ),times=nSNP-3)

X<-matrix(y,nrow=SampleSize)
```

EK-3. (devam) generateData.R fonksiyonu: Simülasyon tasarımı

```
MrkData<-cbind(SNPData$Mrk1,SNPData$Mrk2)
MrkD<-as.matrix(MrkData)
SNPData<-cbind(X,SNPData$SNP1,SNPData$SNP2,SNPData$SNP3)
SNPD<-as.matrix(SNPData)
colnames(MrkD) <- paste0("Mrk:",1:nMArk)
colnames(SNPD) <- paste0("SNP:",1:nSNP)

# Running fastJT and fastSG tests

res1 <- fastJT(Y=MrkD, X=SNPD, outTopN=3)
res2 <- FastSG(Y=MrkD, X=SNPD, outTopN=3)

# Detection of correct SNPs for Mark1
Marker1FastJ<-tibble(Z=cbind(res1$J[,1],res1$XIDs[,1]))
FastJSNP1<-filter(Marker1FastJ,Z[,1]>1.64)

Marker1FastS<-tibble(Z=cbind(res2$Mark[,1],res2$SNP[,1]))
FastSSNP1<-filter(Marker1FastS,Z[,1]>1.64)

FastJPower<-FastJPower+length(intersect(REALSNP, FastJSNP1$Z[,2]))
FastSPower<-FastSPower+length(intersect(REALSNP, FastSSNP1$Z[,2]))

# Detection of correct SNPs for Mark2
Marker2FastJ<-tibble(Z=cbind(res1$J[,2],res1$XIDs[,2]))
FastJSNP2<-filter(Marker2FastJ,Z[,2]>1.64)

Marker2FastS<-tibble(Z=cbind(res2$Mark[,2],res2$SNP[,2]))
FastSSNP2<-filter(Marker2FastS,Z[,2]>1.64)

FastJPower<-FastJPower+length(intersect(REALSNP, FastJSNP2$Z[,2]))
FastSPower<-FastSPower+length(intersect(REALSNP, FastSSNP2$Z[,2]))

}
FastJPower/(dim(res1$J)[1]*dim(res1$J)[2]*itNum)
FastSPower/(dim(res1$J)[1]*dim(res1$J)[2]*itNum)
```



GAZİ GELECEKTİR...