

R Programlama Dili

Using Of R Programming Language İn Agricultural Area

Galip ŞİMŞEK¹ Ufuk KARADAVUT²

Öz:

Bilimsel arařtırmalarda verilerin elde edilme süreçleri oldukça zaman alıcı ve zorlu süreçlerdir. Uzun uğrařlar ve zaman alması alınacak sonuçların önemini daha da artırmaktadır. Elde edilen sonuçların düzenlenmesinden sonra yapılması gereken işlem analizlerin yapılmasıdır. Verilerin istatistiksel analizlerinin yapılabilmesi için çok sayıda paket program kullanılmaktadır. Programların her biri kendine göre bir algoritma ile bunları sonuçlandırmaktadır. Programların tamına yakını ücretli ve müdahale şansı olmayan kapalı programlardır. Zaman zaman ciddi sorunlar oluşturabilmektedirler. Ancak son yıllarda R programı olarak adlandırılan “R” programı bize geniş bir alanını ücretsiz olarak kullanmamızı sağlamaktadır. Bu çalışmada R programlama dilinin istatistiksel olarak tarımsal arařtırmalarda nasıl kullanılacağı örnekler verilerek açıklanmaya çalışılmıştır.

Anahtar sözcükler: R programı, Analiz, İstatistik, Tarımsal veriler

Abstract:

The process of obtaining data in scientific research is quite time consuming and challenging. Long efforts and time-consuming increase the importance of the results. After editing the results, it should be done the statistical analysis. Numerous software programs are used for statistical analysis of the data. Each of the programs concludes with an algorithm of their own. Almost all the programs are paid and closed programs with no chance of intervention. They may cause serious problems from time to time. However, the R program, which has

¹ Ahi Evran Üniversitesi Ziraat Fakültesi, Kırşehir, Sorumlu yazar; gsimsek@hotmail.com

² Ahi Evran Üniversitesi Ziraat Fakültesi, Kırşehir, ufukkaradavut@ahievran.edu.tr

been called the "R program" in recent years, allows us to use a large area free of charge. In this study, it has been tried to explain how to use R programming language statistically in agricultural researches by giving examples.

Keywords: R program, Analysis, Statistics, Agricultural data

Giriş

Bilimsel araştırma sürecinde elde edilen verileri çözümlmek ve araştırma problemine ilişkin bulgulara ulaşmak en önemli adımı oluşturmaktadır. İstatistiksel paket programlar ise elde edilen nicel verilerin çözümlenmesinde kullanılmaktadır. Son yıllarda pek çok istatistiksel paket program piyasaya sürülmüştür. Bu istatistiksel paket programlar nicel verilerin çözümlenmesinde farklı işlem ve uygulama kolaylığı kabiliyetlerine göre önemli kolaylıklar sağlamışlardır. Bu paket programlardan biri olan R dili programlaması ise araştırma problemlerine uygulanacak analiz ve grafik oluşturma sürecinde internet üzerinden ücretsiz temin edilebilecek bir yazılımdır. R yazılımının temelini Becker ve Chambers tarafından geliştirilmiş olan "S" dili oluşturmaktadır. R yazılımı daha önceleri piyasaya sürülmüş ve ücretli olan S-PLUS'ın gelişmiş bir versiyonudur (Er ve Sönmez, 2005)

Bell Labs'da çalışan John M.Chambers ve Becker S adında açık kod kaynaklı olmayan bir istatistik yazılımı geliştirmiştir. Fakat bu programın açık kod kaynaklı olmaması zamanla bulunan yeni analizlere veya araştırmalara karşılık vermemeye başlamasına neden olmuştur (Bunn, 2008). Ross Ihaka ve Robert Gentleman ise S programının bu dezavantajını ve daha sonra S-PLUS ismiyle piyasa sürülen programın paralı olmasını protesto ederek açık kod kaynaklı R programlama dilini geliştirmişlerdir (Ihaka and Gentleman, 1996). R programlama dili S-PLUS'ın gelişmiş versiyonu olarak piyasaya ücretsiz olarak çıkarılmıştır. İsmi yine geliştiricilerinin isimlerinin baş harflerinden gelen R programlama dili. Ross Ihaka ve Robert Gentleman'ın Yeni Zelanda'da 6 yıllık bir çalışmasının sonucu olarak tamamlanmış ve insanlığın gelişmesi için bilime armağan edilmiştir.

R dili; Basic, Fortran, C gibi bir programlama dilidir. Ancak tüm bunlardan farklı olarak R programlama dili, zorlayıcı ve kullanılması zaman alıcı yazım kurallarına gerek duymaz. Araştırmacıların R dili kullanmaları için pek çok sebep vardır. Bunlardan bazıları; R ücretsizdir, kolayca indirilip çalışılan disiplinle ilişkili yüzlerce pakete ulaşılabilir. Programlandırılabilir, kullanıcılar kendi algoritmalarını fonksiyonlarını yazabileceği gibi R kütüphanesindeki paketleri ve fonksiyonları da kendi algoritmalarında kullanabilirler (Bunn, 2010). Geliştirilebilir, yeni bir yöntem için fonksiyonlar oluşturulabilir, revize edilebilir ve kütüphaneye kaydedilebilir. R' de bulunan her paket yardım paketini de içerdiğinden kavram

kargaşasını da ortadan kaldırmaktadır (Crawley, 2013). Karşılaşılan herhangi bir sorunda veya analiz için yaşanan kararsızlıklarda kolayca yapılan bir arama motoru sorgulamasıyla muhtemelen yaşanan probleme benzer bir problemi yaşamış bir araştırmacıya rastlar ve buldukları çözümler görülebilir. Program Mac, Windows, ve Linux işletim sistemlerinin hepsinde çalışmaktadır (Brusilovsky ve ark., 1997). Diğer programlama dilleri ile ilişkisi kuvvetlidir. Diğer programlarda yazılan programlar çağırılabilirler. R programlama hızlıdır, kuralları esnektir, detaylarla uğraşmak yerine fikirlere odaklanma imkanı sağlar (Gürsakal, 2012).

En önemli özelliği ücretsiz olması olan R programlama dilinin diğer MATLAB SPSS, SAS, MINITAB gibi ücretli ve açık kaynak kodlu olmayan programlardan ayıran bir diğer özelliği de başlı başına bahsedildiği gibi açık kod kaynaklı olmasıdır (Ihaka and Gentleman, 1996).. Büyük şirketlerin kalabalığı bir kaynak gibi kullanarak araştırma-geliştirmede maliyeti düşürmesi, kalabalığın aklının çok daha fazla kullanılmasına yol açtı (Davis ve Smith, 1983). Bu sayede Google, Facebook, Yahoo, Amazon gibi şirketler kalabalığın ürettiği verilerden oluştu. Son yüzyıllarda bilimsel anlayışta da bunun farkına varılmasıyla dikkati çeken büyümenin paylaşılan bilgilerin çoğalmasına bağlamalıyız (Crawley, 2013). Bilimsel kuramların, deneysel ve gözlemsel verilerin yayınlanması başkalarının da bu bilgileri ayrıntısıyla incelemesine geliştirmesine veya düzeltmesine olanak sağlayarak ileri bir anlayışın oluşmasına temel olmuştur. Bilimsel çevrelerde önemi yeni yeni anlaşılmaya başlanan veri paylaşımını ABD’de Ulusal Kanser Enstitüsü, Ulusal Sağlık Enstitüsü, Ulusal İnsan Genomu Araştırma Enstitüsü gibi kanser araştırması yapan kuruluşlar ellerindeki verileri yeni gelişmelere, bulgulara yol açmak için paylaşma kararı aldılar (Satman, 2010).

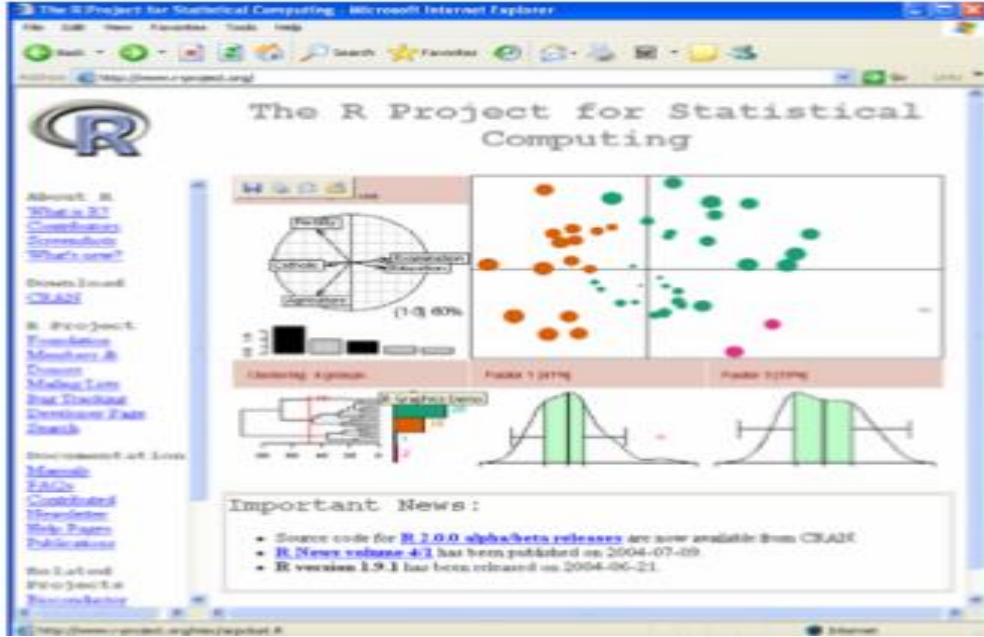
Tarihe baktığımızda başlangıçta yazılımlar ücretsizlerdi. 1960 yılında IBM ve diğer firmalar değiştirilebilecek, geliştirilebilecek başka kullanıcılarla paylaşılacak kaynak kodlu yazılımları bilgisayar ile bedava gönderiyordu. 60’lı yılların sonunda ise bu değişti. Artık bilgisayar ve programlar ayrı ayrı satılmaya başlandı. Kullanıcıların programları geliştirme, değiştirme ve başkaları ile paylaşabilmeleri için para vermeleri gerekiyor.

R programlama kalabalığının gücünü kullanarak en yeni gelişmeleri ve en güncel analiz yöntemlerini bilime sunarak bu sayede bilimin gelişmesi ve yeni fikirlerin giderek ilerlemesine vesile olmaktadır. Şimdilerde R’deki gelişmeler 12 kişilik bir ekip (R Core Team) tarafından yürütülmektedir (Ihaka and Gentleman, 1996).. R’nin her bileşeni, açık kaynak kodlu yazılım felsefesine uygun olarak uyarlanabilir, incelenebilir, refüze edilebilir. Google Merck, the Inter Continental Hotels Group,Shell,Pfizer ve Bank of America gibi şirketlerde R programlama kullanmaktadırlar (Satman, 2010).

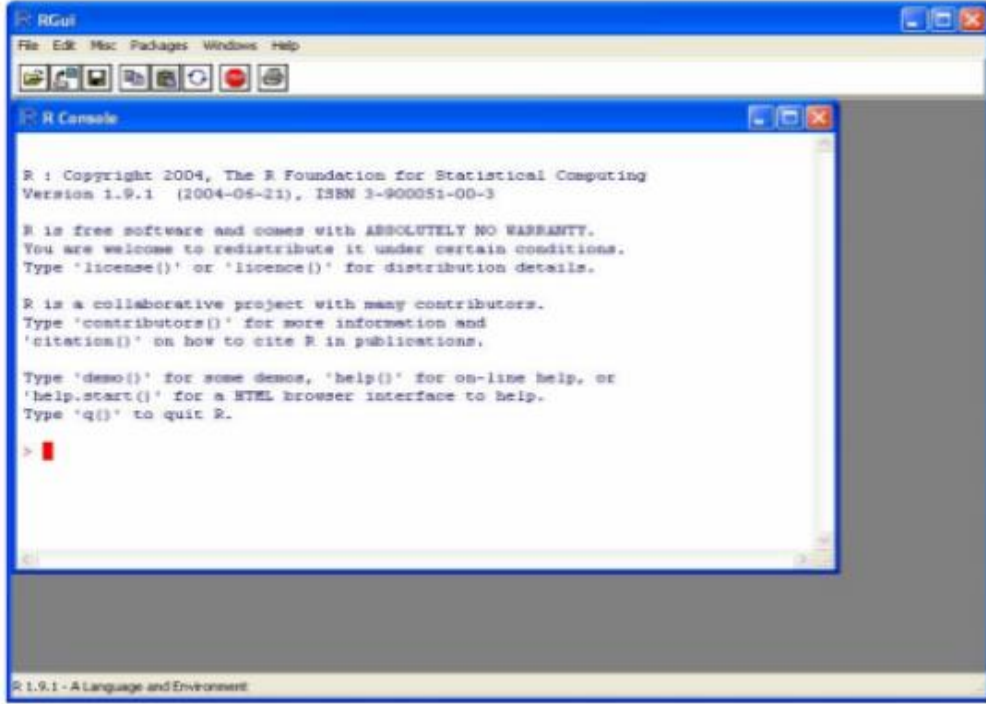
R programının açık kaynak kodlu olmasının anlamı yapılan her türlü analiz yöntemlerinin kodlarına erişebilmek ve bu kodları istediğimiz amaca göre değiştirebilmemizdir. R programının interaktif olarak kullanabileceğimiz gibi yığın işleme modunda da kullanabiliriz. Yani R ile konsoldan komut yazarak çalışabileceğimiz gibi R editöründe kayıtlı komutları da kullanabiliriz. Ve bu komutları istediğimiz zaman saklayıp değiştirip çalıştırabiliriz.

1. R programlama Dilinin Temel Operatörlerinin Manuel Uygulamalarla Tanıtılması

R Project programı <http://www.R-project.org/> internet sayfasından ücretsiz ve hızlı bir şekilde kolayca edinilebilmektedir (Ihaka and Gentleman, 1996).. İnternet sayfasının görüntüsü Şekil 1.'de verilmiştir. R programının ara yüzü ise Şekil 2.'deki gibidir.



Şekil 1. R Project internet sayfası



Şekil 2. R Project paket programı ara yüzü

2.1 Temel R Operatörlerinin Kullanımı

R programı hesap makinesi gibi kullanılabilir.

> #toplama işlemi "+" operatörü ile yapılır.

> 23+40+12+80

[1] 155

> #çıkarma işlemi "-" operatörü ile yapılır.

> 40-23

[1] 17

> #çarpma işlemi "*" operatörü ile yapılır.

> 12*45

[1] 540

> #bölme işlemi "/" operatörü ile yapılır.

> 75/5

[1] 15

> #kuvvet alma işlemi "^" veya "**" operatörü ile yapılır.

> 5^2

[1] 25

> 5**2

```
[1] 25
```

```
> #mod alma işlemi "%%" operatörü ile yapılır.
```

```
> 45%%12
```

```
[1] 9
```

```
> (9%%6)^2+(8/4)-(3*2)
```

```
[1] 5
```

2.2 R'de Değişkenlere Değer Atanması

Değişkenler pek çok veriyi daha sonra kullanmak üzere saklanmasına ve veriler üzerinde yapılacak işlemlerde kolayca ulaşmamıza olanak sağlayan sembolik ifadelerdir. Örneğin, 25 değerini X değişkenine kaydetmek için "X<-25" yapılmalıdır.

#yaş değişkenine 25 sayısını "yaş<-25" veya "yaş=25" operatörüyle atanır.

```
> yaş<-25
```

```
> yaş
```

```
[1] 25
```

```
> yaş=25
```

```
> yaş
```

```
[1] 25
```

```
> pi<-3.14
```

```
> pi
```

```
[1] 3.14
```

> #değişkene değer atandıktan sonra değişken yazılarak işleme devam edilir.

```
> x<-5
```

```
> x
```

```
[1] 5
```

```
> y<-10
```

```
> y
```

```
[1] 10
```

```
> x+y
```

```
[1] 15
```

```
> y%%x
```

```
[1] 0
```

```
> x**y
```

```
[1] 9765625
```

```

> #değişkenlere değer atayarak işlemlere değişkenlerle devam edilebilir. Örneğin;
> maliyet<-20
> kar<-gelir-maliyet
> kar
[1] 80
> #mevcut bir değişkene yeni bir değer atanırsa önceki değer değişecektir.
> boy<-1.78
> boy
[1] 1.78
> boy<-1.85
> boy
[1] 1.85

```

2.3 Değişken Tiplerini Kaydetme veya Belirleme

Her değer, değeri sınıflandıran bir veri türüne sahiptir. Örneğin; isimler karakter verileridir fakat yaşlar sayısal verilerdir.

> #"r", "data", "veri" gibi (tırnak işareti hariç) harflerden veya kelimelerden oluşan karakter verileri değişkenlere atanabilirler.

```

> s<-"r"
> s
[1] "r"
> z<-"data"
> z
[1] "data"
> d<-"veri"
> d
[1] "veri"
> #karakter içeren veriler tırnak işareti kullanılarak atama yapılır.
> #class() fonksiyonu ile verilerin tiplerini öğrenebiliriz.
> class(s)
[1] "character"
> class(z)
[1] "character"
> class(d)

```

```
[1] "character"
```

```
> x<-5
```

```
> class(x)
```

```
[1] "numeric"
```

> #eğer 5 sayısını tırnak işareti ile x değişkenine atarsak program bu veriyi karakter olarak görecektir.

```
> x<-"5"
```

```
> class(x)
```

```
[1] "character"
```

> #bu sebeple veri tiplerine göre karakter veriler tırnak işareti içinde sayısal veriler ise olduğu gibi ataması yapılmalıdır.

2.4 Vektörler Oluşturma ve Vektörler Üzerinde R Operatörleriyle İşlem Uygulamaları

Mantıksal, karakteristik ve sayısal verilerden oluşan tek boyutlu dizileri vektör olarak tanımlıyoruz (Ihaka and Gentleman, 1996).. Vektörleri R’ de şu şekilde tanımlayabiliriz:

- Sayısal vektörler
- Karakter vektörleri
- Mantıksal vektörler

Verilerin nesnesi olan vektör elemanları sayısal (numeric), karakter (character), mantıksal (logical) ve liste (list) şeklinde olabilirler.

Vektör veri kümesindeki tek bir değişken için oluşturulabilir. Vektörlerin sayısal elemanları sürekli veya kesikli sayılardır. Karakter elemanları, harflerden ve diğer karakterlerden oluşan kümedir. Mantıksal elemanlar, TRUE(doğru) veya FALSE(yanlış) şeklindedir. Listelerde ise herhangi bir türden, sınıftan nesne yer alabilir.

“c()”fonksiyonunu kullanarak bir vektöre birden fazla değeri depolayabiliriz.

```
> c(1,2,3)
```

```
[1] 1 2 3
```

> #R vektör içindeki elemanları boşluklarla ayrılmış bir şekilde yazdırır.

```
> c(1,4,5)
```

```
[1] 1 4 5
```

> #R de oluşturulan vektörler bir değişkene atanarak yazdırılabilirler. Değişkene atanan vektör elemanları yine boşluklarla ayrılmış bir şekilde görüntülenecektir.

```
> v<-c(4,5,6)
```

```
> v
```



```
[1] 4 5 6
```

```
> #vektör elemanları karakter, sayısal ve mantıksal verilerden oluşabilirler.
```

```
> a<-c("x","y","z")
```

```
> a
```

```
[1] "x" "y" "z"
```

```
> b<-c(TRUE,TRUE,FALSE)
```

```
> b
```

```
[1] TRUE TRUE FALSE
```

```
> #vektör elemanları isimlere sahip olabilirler. "names()<-c()" fonksiyonu ile elemanlara ait isimler yazdırılabilirler.
```

```
> a<-c(2,3,4,5)
```

```
> a
```

```
[1] 2 3 4 5
```

```
> names(a)<-c("a","b","c","d")
```

```
> a
```

```
a b c d
```

```
2 3 4 5
```

```
> #örneğin;2013 yılı ocak ayında yapılacak 5 farklı sınavın günlerinden oluşan vektörü şu şekilde yazabiliriz;
```

```
> günler<-c(1,2,3,4,5)
```

```
> günler
```

```
[1] 1 2 3 4 5
```

```
> names(günler)<-c("pazartesi","salı","çarşamba","perşembe","cuma")
```

```
> günler
```

```
pazartesi   salı   çarşamba   perşembe   cuma
```

```
1           2   3           4           5
```

```
> #vektörler oluşturulurken aritmetik işlemler yapılabilir ve sonuç vektörde depolanır.
```

```
> x<-c(3+3,3+5,4+1,2+2)
```

```
> x
```

```
[1] 6 8 5 4
```

```
> #eğer iki vektörü toplama "+" işlemi ile ekler isek vektör sırasına göre toplayıp sonucunu depolayacaktır.
```

```
> a<-c(1,2,3,4)
```

```
> b<-c(5,6,7,8)
```

```
> a
```

```
[1] 1 2 3 4
```

```
> b
```

```
[1] 5 6 7 8
```

```
> a+b
```

```
[1] 6 8 10 12
```

```
> c(3,1,7)+c(8,2,4)
```

```
[1] 11 3 11
```

> #iki vektörün toplamı yeni bir değişkene aktarılabilir. örneğin;

```
> a<-c(1,2,3,4)
```

```
> b<-c(4,5,6,7)
```

```
> c<-a+b
```

```
> a
```

```
[1] 1 2 3 4
```

```
> b
```

```
[1] 4 5 6 7
```

```
> c
```

```
[1] 5 7 9 11
```

> #sum() fonksiyonu ile parantez içindeki herşeyin toplamı alınabilir.

```
> x<-c(5,5,10)
```

```
> sum(x)
```

```
[1] 20
```

```
> a<-c(8,9,3,4)
```

```
> sum(a)
```

```
[1] 24
```

```
> x<-c(3,5,1,7)
```

```
> y<-sum(x)
```

```
> y
```

```
[1] 16
```

> #vektör içindeki herhangi bir elemanı indeksiyle saptanabilir. İlk veri indeks 1'de ikinci veri indeks 2'de gibidir.

```
> v<-c(5,3,1,7,8,3,4)
```

```
> v[1]
```

```
[1] 5
```

```

> v[4]
[1] 7
> v[c(1,3)]
[1] 5 1
> #R de sıralı tam sayılardan oluşan bir vektör oluşturmak için kısayol olarak "başlangıç
sayısı:bitiş sayısı" yapılabilmektedir. Örneğin;
> 1:6
[1] 1 2 3 4 5 6
> 5:30
[1] 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
> 4:8
[1] 4 5 6 7 8

> #bir vektörde bulunan bir dizi öğenin alt kümesini oluşturmak için yine ":" operatörü
kullanılabilir.
> x<-c(2:9)
> x
[1] 2 3 4 5 6 7 8 9
> x[5:7]
[1] 6 7 8
> x[4:5]
[1] 5 6
> a<-c(5,1,4)
> b<-c(2,2,8)
> c<-a[1:2]+b[2:3]
> c
[1] 7 9
> # vektör içindeki verilerin ortalamasını hesaplamak için "mean()" fonksiyonu
kullanılmaktadır. Örneğin;
> v<-c(5,2,5)
> mean(v)
[1] 4
> a<-c(4:15)
> a

```

```
[1] 4 5 6 7 8 9 10 11 12 13 14 15
> mean(a)
[1] 9.5
> # R nin karşılaştırma operatörleri vardır. Örneğin; İki değerin eşit olduğu (= =),eşit
olmadığı(!=),büyük yada küçük olduğu(<,>) kontrol edilebilir.
> 6>4
[1] TRUE
> 4==4
[1] TRUE
> "kitap"!="yemek"
[1] TRUE
> 6<4
[1] FALSE
> 4!=4
[1] FALSE
> "kitap"= "yemek"
[1] FALSE
> v<-c("kedi","köpek","fare")
> v= "kedi"
[1] TRUE FALSE FALSE
> v<-c(5,6,7,4)
> v<=6
[1] TRUE TRUE FALSE TRUE
> v<-c(2,4,3,9,11)
> x<-v<7
> x
[1] TRUE TRUE TRUE FALSE FALSE
> #indekslerin veya isimlerin vektörlerini kullanarak alt kümelere ayırma işlemi mantıksal
karşılaştırmalarla da yapılabilir.
> v<-c("küçük","orta","büyük")
> normal<-v!="küçük"
> normal
[1] FALSE TRUE TRUE
> v[normal]
```

```

[1] "orta" "büyük"
> w<-c(5,3,5,1,8)
> j<-w<4
> w[j]
[1] 3 1
> k<-w== 5
> w[k]
[1] 5 5
> # örneğin herhangi bir v vektörünün ilk üç elemanının ortalamasının 5'den büyük olup
    olmadığı mantıksal operatörlerle kontrol edilmek istenirse;
> v<-c(2,1,3,9,11,2,4,7,1)
> v
[1] 2 1 3 9 11 2 4 7 1
> mean(v[1:3])>5
[1] FALSE
> # aşağıdaki örnek gibi iki vektör toplamının elemanları arasında karşılaştırma
    operatörleriyle alt küme oluşturulabilir.
> x<-c(10,1,11,18)
> y<-c(6,10,2,0)
> toplam<-x+y
> z<-toplam>=16
> toplam[z]
[1] 16 18

```

2.5 R Programlama Dilinde Kullanılan Temel İstatistiksel Operatörler

R programlama dilinin fonksiyon kodları veriler üzerinde yapılacak olan işlemleri çok daha kolay hale getirmektedir. Örneğin bir verinin Varyans hesaplaması yapılırken gereken matematiksel işlemler C (syntax) dilinde yazılması gerekirken R programlama dilinde sadece 'var(x)' yazılması yeterli olacaktır. Aşağıdaki Çizelge 1.'de temel istatistiksel hesaplamaların R programlama dilindeki karşılıkları verilmiştir.

Çizelge 1. R'de Kullanılan Temel Fonksiyonlar

FONKSİYON	ANLAMI
-----------	--------

max(x)	x'deki en büyük değer
min(x)	x'deki en küçük değer
sum(x)	x'deki verilerin toplamı
mean(x)	x'deki verilerin aritmetik ortalaması
median(x)	x'deki verilerin medyanı
range(x)	x'deki verilerin değişim aralığı
sd(x)	x'deki verilerin standart sapması
var(x)	x'deki verilerin örneklem varyansı
mad(x)	x'deki verilerin ortalama sapması
IQR(x)	x'deki verilerin kartiller arası farkı
quantile(p,x)	x'deki verilerin p. kantili
cor(x,y)	x ve y verileri arasındaki korelasyon katsayısı
sort(x)	x verilerin sıralanması
rank(x)	x verilerinin sıraları
quantile(x)	x'in kantilleri
cumsum(x)	x'in birikimli toplamları
cumprod(x)	x'in birikimli çarpımları
cummax(x)	x'in birikimli maksimumu
cummin(x)	x'in birikimli minimum
colMeans(x)	x'in bulunduğu sütunun ortalaması
colSums(x)	x'in bulunduğu sütunun toplamı
rowMeans(x)	x'in bulunduğu sıranın ortalaması
rowSums(x)	x'in bulunduğu sıranın toplamı

2. R programlama Dilinin Diğer İstatistiksel Paket Programlarıyla Karşılaştırılması

Beaujean (2013) R yazılımının diğer paket programlara göre üç önemli özelliğini şu şekilde belirtmiştir. Birincisi R yazılımının güçlü bir programlama diline sahip olması ve bu özelliğiyle verilerin analizine olanak sağlaması ikincisi kullanıcılara kendi istatistiksel paketlerini oluşturma imkânı sağlaması ve üçüncüsü ise daha önce bahsedildiği gibi R programlama dilinin açık kod kaynaklı bir yazılım olmasıdır.

Bunların yanında dinamik bir yapıya ve sürekli kendini güncelleme özelliğine sahip R yazılımı kullanıcı dostu bir ara yüze sahip değildir. Öğrenilmesinin daha uzun zaman alması kullanıcılarda kaygı yaratmaktadır. Ancak kullanıcılar programın temel mantığını kavradıktan sonra R programının veri çözümleme sürecinde sağladığı özgürlük ve avantajlarının farkına varılabilmektedir (Bunn, 2010)..

Genel olarak R programının kullanıcıya sunduğu avantajları yazarsak; ücretsizdir, yoğun istatistiksel hesaplama işlemlerine karşı güçlüdür, açık kod kaynaklı olduğu için güncel istatistik alanında çalışan araştırmacılar algoritmalarını R ortamına kaydedebilmektedirler, özellikle Amerika ve Avrupa’da yaygın kullanım ve kullanıcı desteği vardır (Emmanuel ve ark., 2004). İstatistiksel analizlerin nasıl yapılması gerektiği hakkında düşündürür. Farklı istatistiksel paket programlardan (Minitab, Matlab, SPSS gibi) veri alabilme özelliği vardır. R dili ile C (syntax) dili benzerlik gösterir. Fonksiyonel bir programlama diline sahiptir.

Tablo 2. Minitab, SPSS ve R paket programlarının teknik karşılaştırması (Er ve Sönmez, 2005)

UYGULAMA	Minitab	SPSS	R programlama
Tek Değişkenli İst.			
Betimleyici İstatistikler	Pozitif	Pozitif	Pozitif
Frekans Dağılımları	Pozitif	Pozitif	Pozitif
Histogramlar	Pozitif	Pozitif	Pozitif
T testleri	Pozitif	Pozitif	Pozitif
Varyans Analizi			
Tek Yönlü	Pozitif	Pozitif	Pozitif
İki Yönlü	Pozitif	Pozitif	Pozitif
Genelleştirilmiş Model	Pozitif	Pozitif	Pozitif
Varyans Bileşenleri	Pozitif	Pozitif	Pozitif
Kovaryans Analizi	Pozitif	Pozitif	Pozitif
Bonferroni Test	Negatif	Pozitif	Pozitif
Kategorik Veri Analizi			
GLM	Negatif	Pozitif	Pozitif
Lojistik Regresyon	Pozitif	Pozitif	Pozitif
Probit Analizi	Negatif	Pozitif	Pozitif
Log-Linear Modeller	Negatif	Pozitif	Pozitif

Grafiksel Teknikler	Negatif	Negatif	Pozitif
Zaman Serileri			
Box-Jekins ARIMA	Pozitif	Pozitif	Pozitif
Spectral Analysis	Negatif	Pozitif	Pozitif
Regresyon Analizi			
Çoklu Regresyon	Pozitif	Pozitif	Pozitif
Ridge Regresyon	Negatif	Negatif	Pozitif
Robust Regresyon	Negatif	Pozitif	Pozitif
Doğrusal Olmayan Regresyon	Negatif	Pozitif	Pozitif
Parametrik Olmayan Testler			
Binom Testi	Negatif	Pozitif	Pozitif
Ki-Kare	Pozitif	Pozitif	Pozitif
Kolmogrov-Smirnov	Negatif	Pozitif	Pozitif
Mac-Nemar	Negatif	Pozitif	Pozitif
Diğer			
Normallik Testleri	Pozitif	Pozitif	Pozitif
Açıklayıcı Veri Analizi	Pozitif	Negatif	Pozitif
Örnekleme Teknikleri	Negatif	Negatif	Pozitif

R için kod yazmayı kolaylaştıran fonksiyonlara sahiptir. R, yaygın olarak kullanılan SPSS, SAS gibi S tabanlı istatistik paket programlarının aksine istatistiksel yazılım geliştirme imkânı sağlar (Crawley, 2013).. Veri işleme manipüle etme ve saklama özelliğine sahiptir. Matematiksel işlemlerde dizi ve özellikle matris gibi hesaplamalarda özel operatörler mevcuttur. Veri çözümlemede ve betimsel gösterimlerde kullanılabilecek gelişmiş grafiksel görüntülere ve araçlara sahiptir (Bunn, 2010)..

R programlama dilinin dezavantajları da bulunmaktadır. Bunlardan bazıları şunlardır; meşakkatli bir öğrenme süreci vardır. Karşılaşılabilecek problemlerin kullanıcı tarafından araştırma yapılarak aşılması gerekmektedir. Basit bir kullanıcı ara yüzüne sahiptir (Crawley, 2013).

MINITAB ve SPSS ile bazı istatistiksel hesaplamaları yapma kabiliyeti karşılaştırıldığında R programlamanın üstünlüğü rahatça görülmektedir.

Sonuç

R project, ücretsiz olması ve dünyanın farklı bölgelerinde bulunan araştırmacıların ortak çalışması ve açık kod kaynaklı olması sonucunda, özellikle veri işleme ve çözümlene alanlarında araştırmacıların ilgisini çekmiştir.

Yapılan bu çalışmada, son yıllarda akademik çalışmalarda yaygın olarak kullanılan R programlama dilinin tanıtılması hedeflenmiştir. Yaygın olarak kullanılan SAS, SPSS gibi istatistiksel paket programları ile R arasındaki en önemli fark R' nin bir paket program değil istatistiksel yazılım geliştirme için matematiksel ve istatistiksel bir laboratuvar görevi görmesidir.

Windows, Unix, MacOS platformlarında kullanılabilen R, kullanıcı ara yüzü dışında S plus ile pek çok açıdan birbirine benzemektedir. Öyle ki bu dillerden birinde yazılan bir kod genellikle diğerinde de çalışmaktadır. Bu iki dili birbirinden ayıran en önemli özellik R'nin ücretsiz bir dil olmasıdır. Bu çalışmada, yeni geliştirilen istatistiksel yöntemler için yazılan paketlerin kullanıcılar tarafından kolaylıkla yüklenebildiği açık kaynak kodlu olan R programlama, tarihçesi, nesne kavramı, fonksiyonları ve diğer özellikleri ile incelenmiştir.

Kaynaklar

- Beaujean, A. A. 2013. Factor Analysis Using. Practical Assessment, Research & Evaluation, 18(4):4-8
- Brusilovsky, P., Calabrese, E., Hvorecky, J., Kouchnirenko, A., & Miller, P. 1997. "Minilanguages: A Way to Learn Programming Principles". Education and Information Technologies, 2 (1):65-83.
- Bunn AG. 2008. "A dendrochronology program library in R (dplR)." Dendrochronologia, 26(2), 115–124.
- Bunn AG 2010. "Statistical and visual crossdating in R using the dplR library." Dendrochronologia, 28(4), 251–258.
- Crawley, M.J. 2013. The R Book. John Wiley and Sons Ltd., West Sussex, United Kingdom.
- Davis, R., and Smith, R. G. 1983. Negotiation as a Metaphor for Distributed Problem Solving, in Artificial Intelligence 20(1):63-109.
- Emmanuel P., Julien, C., Korbinian, S., 2004. APE: Analyses of Phylogenetics and Evolution in R language, *Bioinformatics*, Volume 20(22):289–290,
- Fikret, E. R., & Sönmez, H. 2005. Temel İstatistik Eğitiminde R For Windows Paket Programı Kullanımı. Journal of Engineering and Architectural Faculty of Eskişehir Osmangazi University.

Gürsakar, N. 2012. Betimsel istatistik: İstatistik I. Dora Basım Yayın Dağıtım.

Ihaka,R. and Gentleman,R. 1996. R: a language for data analysis and graphics. J. Comput. Graph. Statist., 5, 299–314.

Satman, M. H. 2010. İstatistik ve ekonometri uygulamaları ile R. Türkmen Kitabevi.