

# Öğrenme Araçlarını

## İçindekiler

Öğrenme Araçlarını .....	1
Hazırlama .....	1
Windows'a R Kurmak .....	1
Veri Kümelerini İndirme ve Örnek Kod .....	3
R Kullanarak Kodlama .....	4
Vektörlerle Çalışmak .....	5
Listeleri Kullanarak Verileri Düzenleme .....	5
Matrislerle Çalışmak .....	6
Temel bir matris oluşturma .....	6
Vektör düzenlemesini değiştirme .....	6
Örnekler .....	7

## Hazırlama

R, bir ortam ve bir dilin birleşimidir. Bu, John Chambers'ın istatistiklerle çalışmayı kolaylaştırmak için ilk olarak Bell Laboratuvarlarında yarattığı S programlama dilinin bir biçimidir. Rick Becker ve Allan Wilks sonunda S programlama diline de eklendi. R dilinin amacı, fikirleri hızlı ve kolay bir şekilde yazılıma dönüştürmektir. Başka bir deyişle, R, fazla programlama deneyimi olmayan birinin büyük bir öğrenme eğrisi olmadan kod oluşturmaya yardımcı olmak için tasarlanmış bir dildir. Bu kitapta S yerine R kullanılmıştır çünkü R, çoğu S kodunu değiştirmeden çalıştırabilen ücretsiz olarak indirilebilir bir üründür; aksine, S için ödeme yapmanız gerekir.

## Windows'a R Kurmak

1. Tarayıcıya R Download yazdıktan sonra “r-project.org” sitesinden son sürüm indirilir.

## Download R-4.3.1 for Windows - CRAN

**Download R-4.3.1 for Windows** (79 megabytes, 64 bit). [README](#) on the Windows binary distribution · [New features in this version](#). This build requires UCRT, ...

[R News](#) · [R-patched snapshot build](#) · [R-devel snapshot build](#) · [FAQs](#)

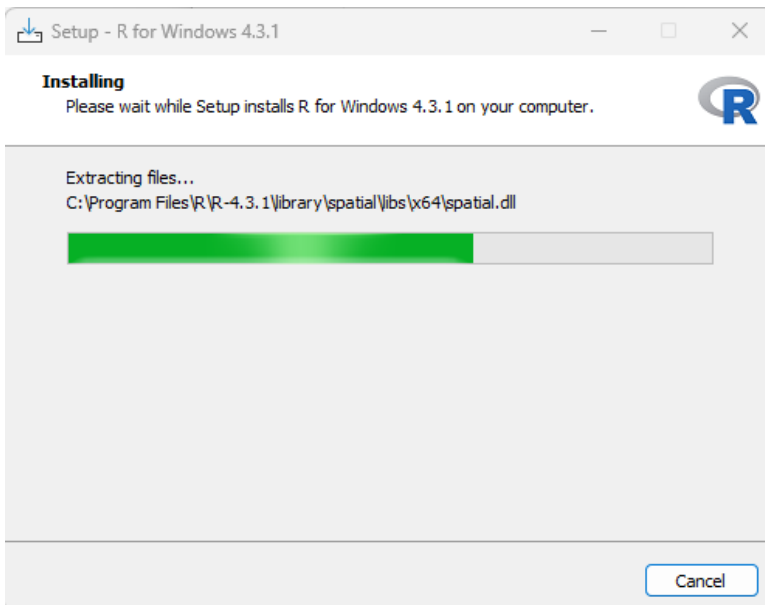
<https://cran.r-project.org> › bin 

## R for Windows - CRAN

Rtools, Tools to build **R** and **R** packages. This is what you want to build your own packages on Windows, or to build **R** itself.

Base: Binaries for base distribution. This is w... Rtools: Tools to build R and R packages. T...

## 2. İndirilen dosyanın kurulumu yapılır.




## 3. R kurulduktan sonra RStudio indirilir ve kurulur.

## RStudio Desktop - Posit

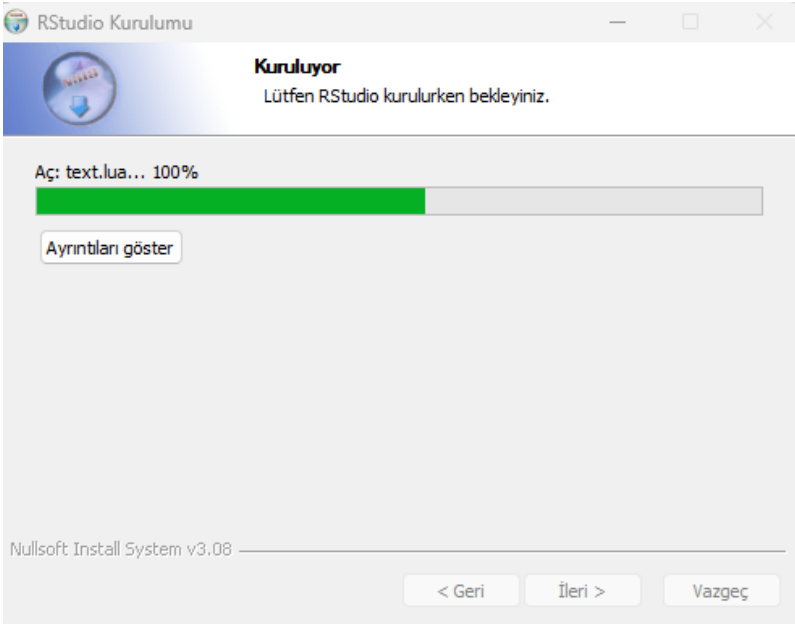
Don't want to **download** or install anything? Get started with **RStudio** on Posit Cloud for free . If you're a professional data scientist looking to **download** ...

[R Packages](#) · [Conf\(2023\)](#) · [Resources](#) · [Videos](#)

<https://posit.co> › downloads 

## Download RStudio - Posit

**Download RStudio.** **RStudio** Desktop. Open Source Edition (AGPL v3). Free. **DOWNLOAD.** Syntax highlighting, code completion, and smart indentation.



## Veri Kümelerini İndirme ve Örnek Kod

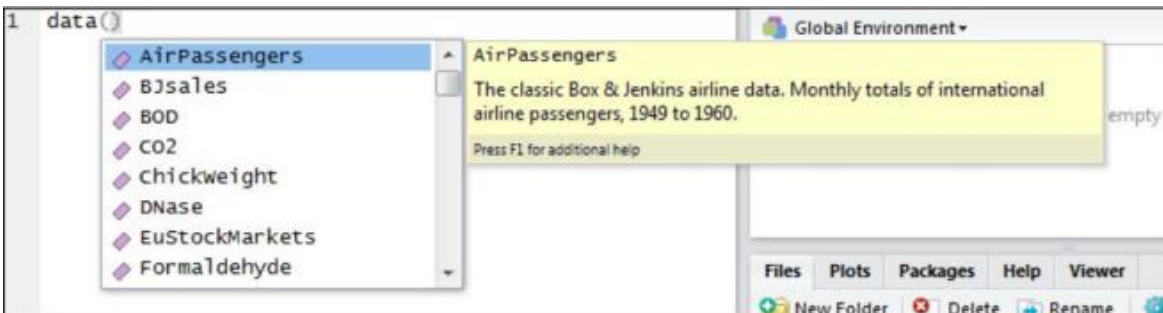
Bu kitapta kullanılan veri kümelerinin çoğu doğrudan R veri kümeleri paketlerinden gelmektedir. Bu veri kümelerinin listesini <https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/> adresinden bulabiliriz.

← ↻ <https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/>

### Index of /R-manual/R-devel/library/datasets/da

Name	Last modified	Size	Description
Parent Directory	-	-	-
<a href="#">00Index.html</a>	2023-08-28 01:47	16K	
<a href="#">AirPassengers.html</a>	2022-06-08 12:37	2.7K	
<a href="#">BJsales.html</a>	2022-06-08 12:37	1.9K	
<a href="#">BOD.html</a>	2022-06-08 12:37	2.5K	
<a href="#">ChickWeight.html</a>	2022-06-08 12:37	3.2K	
<a href="#">DNase.html</a>	2022-06-08 12:37	3.0K	
<a href="#">EuStockMarkets.html</a>	2022-06-08 12:37	1.7K	
<a href="#">Formaldehyde.html</a>	2022-06-08 12:37	2.6K	
<a href="#">HairEyeColor.html</a>	2022-06-08 12:37	4.3K	
<a href="#">Harman23.cor.html</a>	2022-06-08 12:37	1.6K	
<a href="#">Harman74.cor.html</a>	2022-06-08 12:37	1.8K	
<a href="#">Indometh.html</a>	2022-06-08 12:37	2.8K	

Bu veri kümelerini RStudio'da görmek istiyorsanız, boş bir dosyaya “data()” yazmanız yeterlidir ve IDE bunu size veri kümelerini gösterir.



Bu standart veri setlerine ek olarak, kitap tamamlayıcı veri setlerini kullanır. Çoğu durumda, bu ek veri kümeleri Python örnekleri için kullanılanlarla aynıdır ve R'nin çıktısını Python ile karşılaştırmanıza izin verir. Bu ek veri kümeleri şunlardır;

» Boston: <https://stat.ethz.ch/R-manual/R-devel/library/MASS/html/Boston.html>

» Air Quality (airquality): <https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/airquality.html>

» Titanic: <https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/Titanic.html>

» Iris: <https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/iris.html>

» SMS Spam Collection (sms\_data): <http://www.dt.fee.unicamp.br/~tiago/smsspamcollection/>

## R Kullanarak Kodlama

Çoğu programlama dilinin aksine, R, belirli bir türde bir değişken bildirmenizi gerektirmez. R, verileri R nesnelerinde depolar. Bu nesneler, depolamak için kullanılan yöntem ve veri özelliklerine bağlı olarak verilere otomatik olarak bir tür atar. Verileri depolamak için aşağıdaki R nesnelerini kullanırsınız:

1. Vektörler
2. Listler
3. Matrisler
4. Diziler
5. Faktörler
6. Veri çerçeveleri(Data Frame)

Anlaşılması ve kullanılması en kolay R-nesnesi vektördür. Bir vektör aşağıdaki temel veri türlerini içerebilir:

**Logical (Mantıksal):** Doğru veya yanlış gibi bir doğruluk değerini belirten Boolean değeri içerir.

**Numeric (Sayısal):** 12,3 veya 45 gibi herhangi bir sayıyı içerir. Sayısal bir değer bir tamsayı (ondalık olmayan bir tam sayı) veya gerçek bir sayı (ondalık içeren bir sayı) içerir.

**Integer (Tamsayı):** Bir tamsayı değeri tutar; örneğin ondalık noktası olmayan bir sayı 2L veya 34L. Sayı ondalık sayı içermemeli ve tam sayı olarak görünmesini istediğinizi belirtmek için sayının sonuna L harfini eklemelisiniz.

**Complex(Karmaşık):**  $3 + 2i$  gibi hem gerçek hem de sanal değer içeren sayısal bir değer belirtir.

**Character(Karakter):** 'a', "Merhaba", "TRUE" veya '12.3' gibi tek bir karakteri veya dizeyi tanımlar. Bir karakter veri türü sınırlayıcı olarak tek veya çift tırnak işaretlerini kullanabilir.

## Vektörlerle Çalışmak

Bir vektördeki tüm değerler aynı türde olmalıdır. Sonuç olarak, **MyVector <- c(1, "Hello", TRUE)**'yu çağırarak) gibi türleri bir vektörde karıştırdığınızda, yine de üç farklı değer elde edersiniz. Ancak **print(MyVector)** işlevi şu çıktıyı gösterir:

```
[1] "1"      "Hello"  "TRUE"
```

## Listeleri Kullanarak Verileri Düzenleme

Vektörlerin aksine, listeler birden çok veri türü içerebilir. Aslında, listeler vektörleri, diğer listeleri ve işlevleri içerebilir. Listeler, verileri tam olarak saklamayı düşündüğünüz gibi depolamada vektörlerden daha çok yönlü olabilir.

Bir liste oluşturmak için **list()** işlevini kullanmanız gerekir. Aksi takdirde R, bir vektör oluşturmak istediğinizi varsayar. Örneğin, **MyList <- list(1, "Merhaba", TRUE)**, beklediğiniz türde üç ayrı giriş içeren bir liste oluşturur. **print(MyList)** işlevi şu çıktıyı üretir;

```
[[1]]  
[1] 1  
  
[[2]]  
[1] "Merhaba"  
  
[[3]]  
[1] DOĞRU
```

Her giriş ayrıdır. Sonuç olarak, **class(MyList)** ögesini çağırdığınızda, çıktı bir listedir.

Bir liste vektörler içerebilir. Liste içinde bir vektör oluşturmak için **c()** işlevini kullanırsınız. Örneğin, **MyVectorList <- list(c(1, 2, 3))** vektör içeren bir liste oluşturur. **print(MyVectorList)** işlevi aşağıdaki çıktıyı gösterir:

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

## Matrislerle Çalışmak

### Temel bir matris oluşturma

Vektörler ve listeler tek boyutlu R nesneleridir. Ancak, verileri doğru bir şekilde ifade etmek için genellikle iki boyuta ihtiyaç duyarsınız. Bir matris, verileri ifade etmek için iki boyutlu bir yöntem sağlar. Listenin aksine, bir `matrix()` işlev çağırısı veriden daha fazlasını içerebilir. Aslında, bir `matrix()` `matrix(data, nrow, ncol, byrow, dimnames)` öğelerini içerir. Öğelerin her biri, R'nin tam olarak istediğiniz matrisi oluşturmaya yardımcı olan işlevin bir argümanıdır. İşte bu argümanların her birinin hızlı bir açıklaması;

» `data`: Matristeki veri öğelerini doldurmak için kullanılan verileri tanımlar.

» `nrow`: Oluşturulacak satır sayısını belirtir.

» `ncol`: Oluşturulacak sütun sayısını belirtir.

» `byrow`: Matris içindeki vektör öğelerinin düzenini tanımlar. Vektör öğelerini satıra göre düzenler.

» `dimnames`: Satırlara ve sütünlara atanan adları belirler.

Matrisinizin birden fazla sütun içermesini istiyorsanız, minimum satır sayısını belirtmelisiniz. Örneğin, `MyMatrix <- matrix(c(1:6), 3)` öğesini çağırmak üç satırlı bir matris oluşturur. `print(MyMatrix)` işlev çağırısı aşağıdaki çıktıyı gösterir;

```
      [,1] [,2]  
[1,]    1    4  
[2,]    2    5  
[3,]    3    6
```

### Vektör düzenlemesini değiştirme

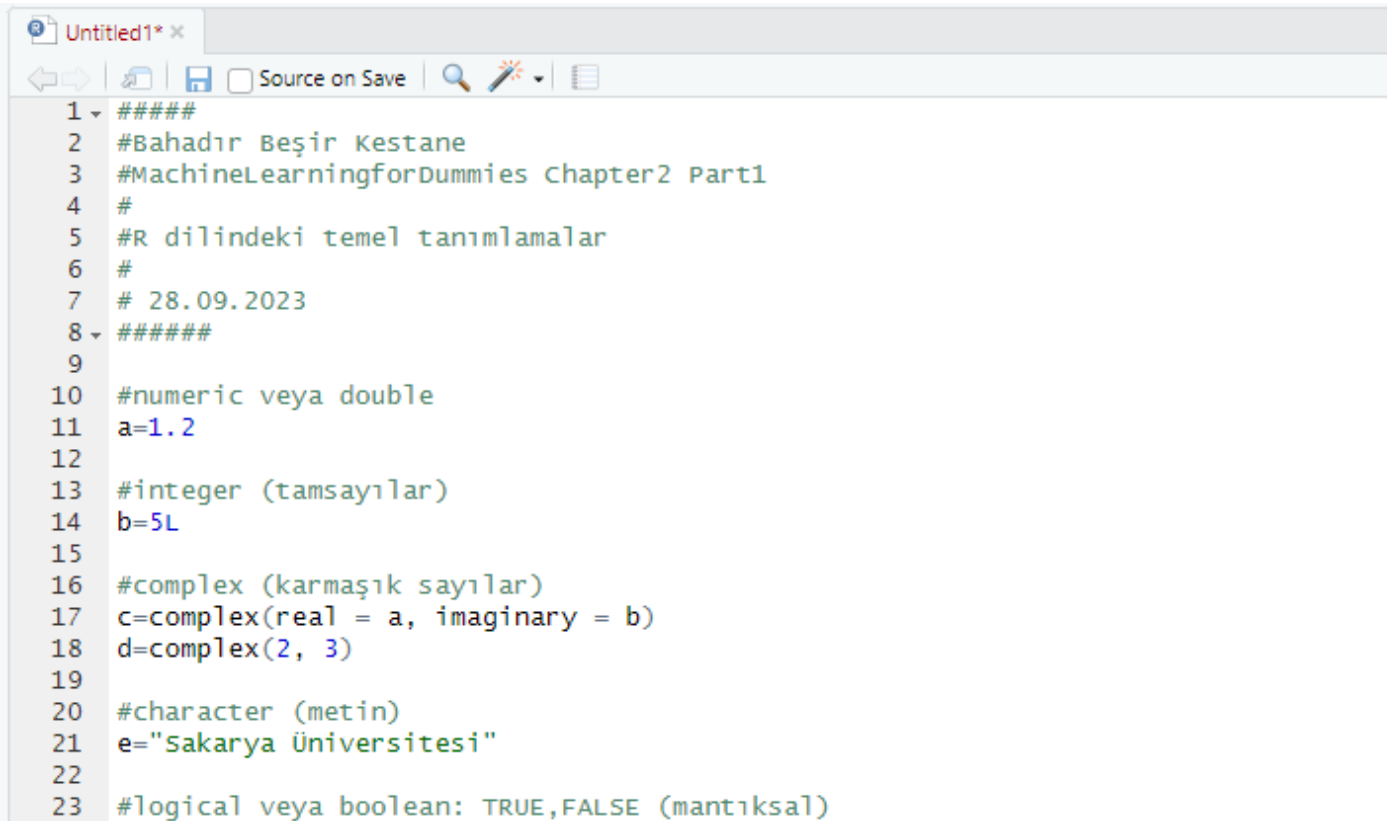
Temel matris oluşturma tekniklerini kullandığınızda, elemanları sütun sırasına göre dolduran bir matris elde edersiniz. Örneğin, `MyMatrix <- matrix(c(1:8), 2, 4)` adını verdiğinizde, şu çıktıyı alırsınız:

```
      [,1] [,2] [,3] [,4]  
[1,]    1    3    5    7  
[2,]    2    4    6    8
```

Bağımsız değişkeni DOĞRU olarak ayarladığınızda, matrisi doldurmak için kullanılan sıra bir satır sırasına dönüşür. `MyMatrix <- matrix(c(1:8), 2, 4, TRUE)` çağırılması şu çıktıyı verir:

```
      [,1] [,2] [,3] [,4]  
[1,]    1    2    3    4  
[2,]    5    6    7    8
```

## Örnekler



```
1 #####  
2 #Bahadır Beşir Kestane  
3 #MachineLearningforDummies Chapter2 Part1  
4 #  
5 #R dilindeki temel tanımlamalar  
6 #  
7 # 28.09.2023  
8 #####  
9  
10 #numeric veya double  
11 a=1.2  
12  
13 #integer (tamsayılar)  
14 b=5L  
15  
16 #complex (karmaşık sayılar)  
17 c=complex(real = a, imaginary = b)  
18 d=complex(2, 3)  
19  
20 #character (metin)  
21 e="Sakarya Üniversitesi"  
22  
23 #logical veya boolean: TRUE,FALSE (mantıksal)
```

```
24 f=TRUE
25
26
27 #vektör
28 vektor1=c(3,7,-5,2)
29 vektor2=c(1:6)
30
31 vektor7=c("abc","def","ghi","jklmno")
32
33 nchar(vektor7)
34
35 #vektör Elemanlarının Güncellenmesi
36 vektor10=c(11,12,13,14,15)
37 vektor10[1]=22 #Güncellendi
38
39
40 #MATRİSLER
41 Matris_1 = matrix(c(7,3,4,3,5,2))
42
43 #Örnek-2, nrow parametresi ile satır sayısının 3 olduğu belirtiliyor
44 Matris_2 = matrix(c(7,3,4,3,5,2),nrow=3)
45
46 #Örnek-2, ncol parametresi ile sütun sayısının 3 olduğu belirtiliyor
47 Matris_3 = matrix(c(7,3,4,3,5,2),ncol=3)
48
49 |
```