

# R'ye bařlarken



**Bulařıcı hastalık dinamiklerinin R'de modellenmesi üzerine kısa kurs**

Ankara, Trkiye, 2025

Dr Juan F Vesga

# Oturumun amaçları

- R motorunu ve sözdizimini anlamak
- R programlamanın ilk önemli kavramlarını tanıtmak
- R dilinin temel sözdizimini öğrenmek
- Veri nesnelerini anlamak
- Kullanıcı tarafından tanımlanan fonksiyonları anlamak

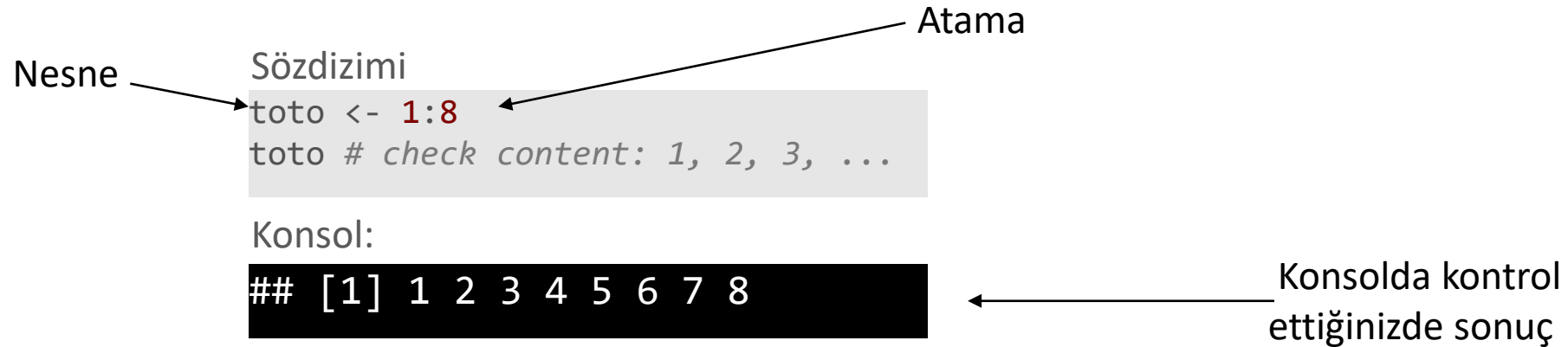
# R bilgileri nasıl kaydeder?



- *Dosya yoktur*, tümü *RAM*'dedir (yani geçici bellek)
- Verilerin, sonuçların, fonksiyonların tümü R *nesnelidir*
- *saveRDS/readRDS* kullanılarak bir *nesne* kaydedilebilir / yüklenebilir (çıktı: *.rds* dosyaları)
- *save/load* kullanılarak bir *birden fazla nesne* kaydedilebilir / yüklenebilir (çıktı: *.RData* dosyaları)
- *save.image* kullanılarak *tüm oturum* kaydedilebilir

# Nesneler nasıl oluşturulur?

- Genel sözdizimi: `object_name <- content`:



**Not:** R'de “<-” sözdizimi atamayı göstermek için kullanılır, “=” sözdizimini kullanırsanız çalışacaktır ancak R kullanıcıları arasında uzlaşma olmadığından kafa karıştırıcı olabilir

# Nesneler nasıl oluşturulur?

- Genel sözdizimi: `object_name <- content`:

Sözdizimi

```
toto <- "some text"  
toto # reassigned a different value
```

Konsol:

```
## [1] "some text"
```

**Not:** "toto" nesnesi sadece bir zarftır. İçine istediğiniz değeri koyabilir ve istediğiniz gibi değiştirebilirsiniz.

# Yuvarlak sayılar: tam sayı

```
a <- 1:10  
a
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

```
class(a)
```

```
## [1] "integer"
```

# Ondalık sayılar: sayısal

```
b <- c(-0.1, 10.123, pi)  
b
```

```
## [1] -0.100000 10.123000 3.141593
```

```
class(b)
```

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

# Metin: karakter

```
a <- c("hello world", "hello Turkey!")  
a
```

```
## [1] "hello world"      "hello Turkey!"
```

```
class(a)
```

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



# Kategorik değişkenler: faktör

```
a <- factor(c("red", "blue", "green", "red") )  
a
```

```
## [1] red blue green red  
## Levels: blue green red
```

```
class(a)
```

```
## [1] "factor"
```

```
levels(a)
```

```
## [1] "blue" "green" "red"
```

# Booleanlar: mantıksal

Mantıksal tip TRUE veya FALSE olabilir:

```
a <- c(TRUE, FALSE, TRUE, TRUE)
a
```

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

```
class(a)
```

```
## [1] "logical"
```

Booleanlar aynı zamanda 0 (FALSE) ve 1 (TRUE), olarak yorumlanabilir, bu yü

```
a + 1
```

```
## [1] 2 1 2 2
```

# Vektörler

Vektör **aynı tipte** birden fazla değeri tek boyutlu bir dizi olarak saklar:

```
a <- c(1, 2, -2, 1.123)  
a
```

```
## [1] 1.000 2.000 10.000 -1.000 1.123
```

```
length(a)
```

```
## [1] 5
```

# Matrisler

Matris **aynı tipte** birden fazla değeri bir tablo olarak saklar:

```
a <- matrix(sample(1:12),ncol = 4)
a
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  12   7   8   4
## [2,]  11  10   2   5
## [3,]   1   6   9   3
```

```
class(a)
```

```
## [1] "matrix"
```

```
dim(a)
```

```
## [1] 3 4
```

# Veri Çerçevesi

**Veri çerçevesi** değişkenlerin (sütunlar) farklı türde olabildiği (elektronik tabloya eşdeğer) bir tablodur:

```
a <- data.frame(age = c(10, 54, 3), sex = c("m", "f", "m"))  
a
```

```
##   age sex  
## 1  10  m  
## 2  54  f  
## 3   3  m
```

```
class(a)
```

```
## [1] "data.frame"
```

```
dim(a)
```

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

# Listeler

**Liste** farklı olarak saklanan her türden ve boyuttan nesnelerin koleksiyonudur. Bilgi kaydetmek için güçlü bir yapıdır.

```
age <- c(10, 54, 3)
sex <- factor(c("m", "f", "m"))
swab <- matrix(
  sample(c("+", "-"), replace = TRUE, 10), nrow = 2,
  dimnames = list(NULL, paste("t", 1:5, sep = "")))
x <- list(age = age, gender = sex, swab_results = swab)
```

# Listeler (devamı)

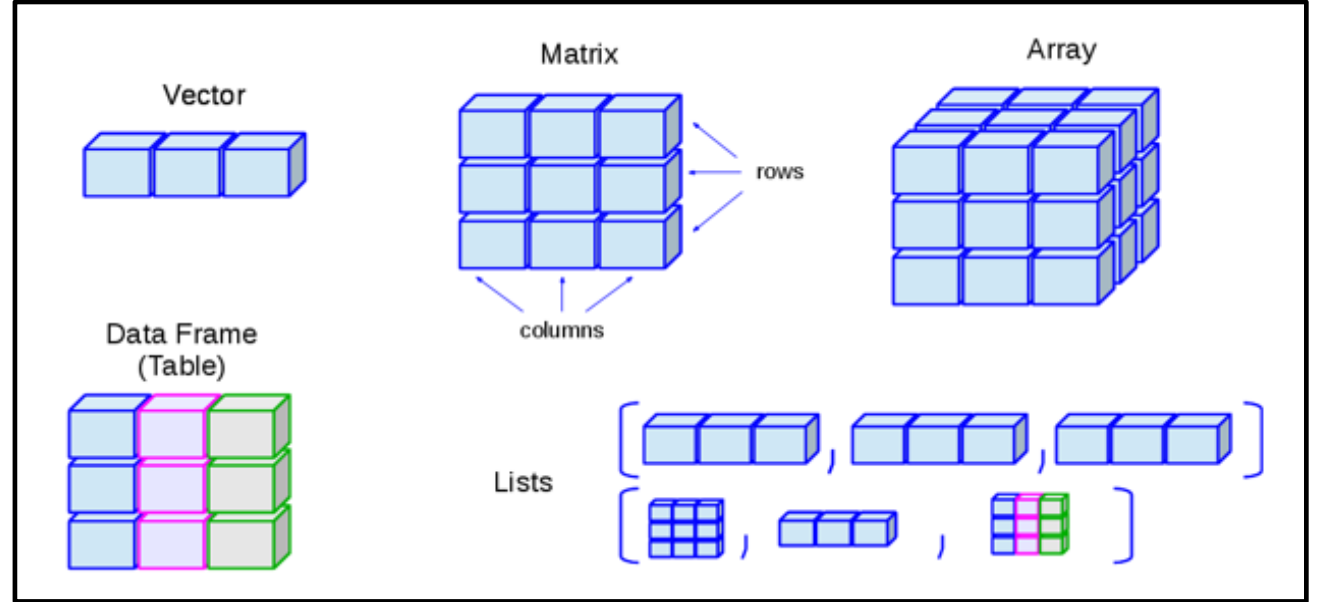
`Liste` farklı olarak saklanan her türden ve boyuttan nesnelerin koleksiyonudur. Bilgi kaydetmek için güçlü bir yapıdır.

```
x
```

```
## $age
## [1] 10 54 3
##
## $gender
## [1] m f m
## Levels: f m
##
## $swab_results
##      t1  t2  t3  t4  t5
## [1,] "+" "+" "-" "+" "+"
## [2,] "+" "+" "+" "-" "+"
```

# Veri yapıları özeti

- Vektörler
- Matrisler ve diziler
- Veri çerçeveleri
- Listeler



G. Tiwari, 2019. <https://medium.com/@tiwarigaurav2512/r-data-types-847fffb01d5b>

- R önceden yüklenmiş bazı veri setleri içermektedir



# Bir nesnedeki içeriklere erişmek

İncelediğimiz nesnelere erişilebilir veya bunlar **endeks**, **ad** veya **mantıksal duruma** göre **alt küme haline getirilebilir**. Şunlar kullanılır:

- Vektör için `object_name[]`
  - Matris / veri çerçevesi için `object_name[rows, columns]`
  - Liste için `object_name[[]]`
- ❑ Endeks, bir dizideki elemanın konumunu gösteren bir tam sayı veya tam sayı dizisi olabilir: `my_vector[2]`
  - ❑ **slot** adı kullanılarak listelere erişilebilir: `my_list[["age"]]`
  - ❑ Köşeli parantezlerin bir nesneye erişimi gösterdiğine, parantezlerin ise bir fonksiyonun argümanlarını içerdiğine dikkat edin

# Fonksiyonları kullanmak



- Fonksiyonlar karmaşık işlemleri yapmanın kestirme yollarıdır, örneğin
  - $\text{sort}(c(2, 1, 3))=c(1, 2, 3)$
  - $\text{abs}(-4)=4$
- Fonksiyonlar şunlar tarafından yazılabilir:
  - R (yani kurulmuşsa)
  - Kullanıcı (yani siz)
  - Başka biri (paketteki)

# Fonksiyon nedir

Belirli bir çıktı üzerinde yapılan işlemler dizisi

Sözdizimi: `function_name(argument1, argument2, ...)`

Örnek:

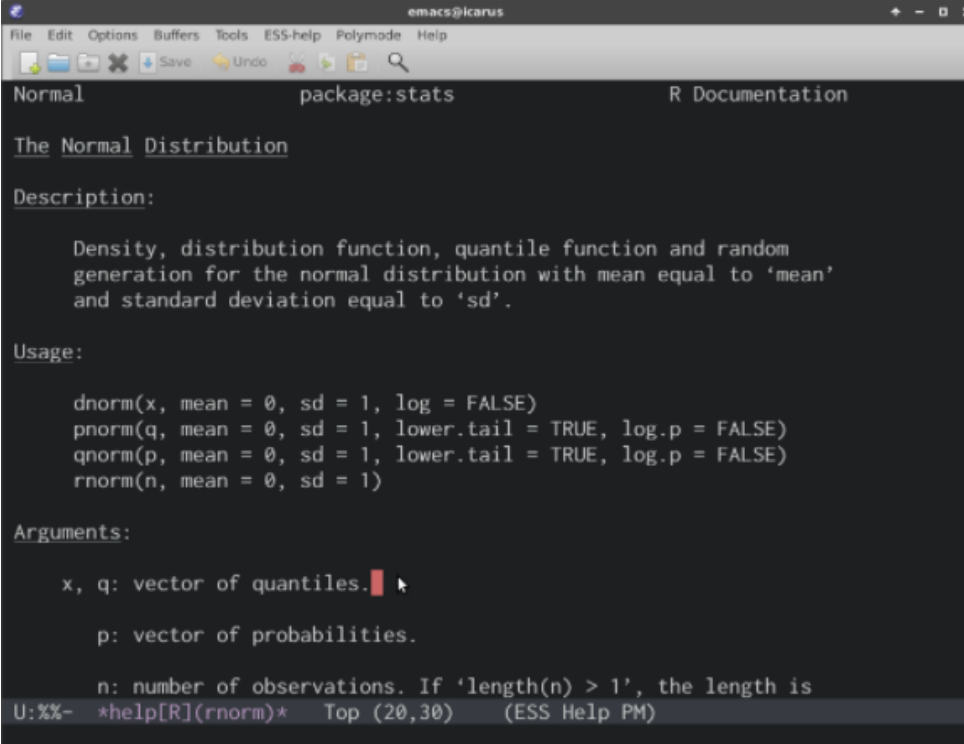
```
rnorm(8, mean=5, sd=3)
```

```
## [1] 2.713883 7.373878 9.364017 1.173302 6.436040 5.624428 7.107626  
## [8] -1,727200
```

# Fonksiyon nasıl kullanılır?

Fonksiyon R veya kullanıcı ya da paket tarafından oluşturulduğunda tanımı "?" kullanarak öğrenilebilir

?rnorm



```
emacs@icarus
File Edit Options Buffers Tools ESS-help Polymode Help
Normal package:stats R Documentation
The Normal Distribution
Description:
Density, distribution function, quantile function and random
generation for the normal distribution with mean equal to 'mean'
and standard deviation equal to 'sd'.
Usage:
dnorm(x, mean = 0, sd = 1, log = FALSE)
pnorm(q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
qnorm(p, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
rnorm(n, mean = 0, sd = 1)
Arguments:
x, q: vector of quantiles.
p: vector of probabilities.
n: number of observations. If 'length(n) > 1', the length is
U:%%- *help[R](rnorm)* Top (20,30) (ESS Help PM)
```

# Kullanıcı tarafından tanımlı fonksiyonlar

Temel R fonksiyonlardan ve paketlerdeki fonksiyonlardan ayrı olarak kendi fonksiyonlarınızı oluşturabilirsiniz.

```
Sözdizimi: function_name <- function(argument1, argument2,...)
{
  # perform an operation using your arguments
  x <- argument1 + argument2

  return(x)
}
```

- ❑ Fonksiyonlara ad vermek iyi kodlamanın önemli bir parçasıdır: anlamlı adlar kullanın! : Sum\_arguments (Good); func1 (Bad)

# Şimdiye kadar bilmemiz gerekenler

- R motorunun nasıl çalıştığı
- R nesneleri kullanarak bilgilerin nasıl saklandığı
- Bilgilerinize erişmek
- Fonksiyonların ne olduğu
- Fonksiyonların kullanıcı tarafından nasıl tanımlanabileceği

# İncelenebilecek faydalı R kaynakları

- R for Data Science - <https://r4ds.had.co.nz/>
- Advanced R <http://adv-r.had.co.nz/>
- R packages <http://r-pkgs.had.co.nz/>
- RECONLearn <https://www.reconlearn.org/>