Основни функции в R

- `install.packages("package name")`: инсталира нов пакет/ библиотека;
- `library(package name)`: зарежда пакет/ библиотека;
- `c(x,y...)`: създава вектор с елементи x, y и т.н.;
- `x[]`: индексира вектора x;
- `x[x>0]`: връща стойностите на вектора x, които отговарят на условието (в случая да са положителни);
- `data.frame()`: създава таблица тип 'data.frame';
- `x[,2]`: връща колона 2 на x, ако x е таблица;
- `x[c(3,5),]`: връща третия и пети ред на x, ако x е таблица;
- `x[-1,]`: премахва ред 1 на x, ако x е таблица;
- `matrix()`: създава матрица;
- `x[1,2]`: ако x е матрица или 'data.frame', избира елемента от ред 1 и колона 2;
- `length(x)`: връща дължината на x;
- `nrow(x), ncol(x)`: връща съответно броя на колоните или редовете на x;
- `head(x), tail(x)`: връща съответно първите или последните 6 елемента на x;
- `colnames(x), rownames(x)`: връща имената съответно на колоните и редовете на x;
- `str(x)`: дава структурата на x;
- `summary(x)`: връща обобщение на x, което е различно в зависимост какъв обект е x;
- `sort(x)`: сортира елементите на x;
- `order(x)`: връща индексите на сортираните елементи на x;

- `min(x),max(x),mean(x),sd(x)`: дава съответно минималната, максималната, средната стойност или стандартното отклонение на x;
- `cbind(x,y), rbind(x,y)`: съединява две таблици, съответно по колони и по редове.;
- `round(x,n)`: закръгля числото x до n-тия знак след десетичната запетая;
- `which(x=='value')`: връща индекса на елементите на x, които отговарят на условието (в случая да са равни на value). Може да се задават различни видове условия;
- `table(x)`: връща едномерна или многомерна таблица с честотно разпределение;
- `prop.table(table(x),margin=1,2)`: връща едномерна или многомерна таблица с процентно разпределение;
- `for` цикъл:
```
for (value in range) {
  statement
}
```
- `while` цикъл:
```
while (condition) {
  statement
}
```
- оператор `if`:
```
if (test_expression) {
  statement
  } elseif (test_expression) {
    statement
    } else {
      statement
      }
```
- Дефиниране на функции в R:
```
function_name <- function(argument1, argument2, ... ){
statements
return(object)
}
```

Обработка на данни с dplyr/ tidyverse

- **select**: Избира подмножество от колони от данните
  `dat1<-dat %>% select(var1,var5)`

- **arrange**: подрежда наблюденията във възходящ или низходящ ред на зададена променлива
  `dat %>% arrange(var1)`

- **distinct**: извежда стойностите на променливата без повторения
  `dat %>% distinct(var1)`

- **slice**: избира редове по тяхната позиция
  `dat %>% slice(5000:5100)`

- **filter**: Избира подмножество от наблюденията (по редовете), въз основа на определни критерии
  `dat %>% filter(var1=='value')`

- **mutate**: Създава нови колони (с формули)
  `dat %>% mutate(var1==var2/var3-5)`

- **group_by**: променя аналитичната единица от цялата база данни на отделни групи

- **summarise**: представя данните обобщено
  `dat %>% group_by(var) %>% summarise(mean=mean(var2))`

- **gather**: преобразува таблицата в таблица само с две колони, едната съдържа имената на променливите (key), а другата - на техните стойности (value), като могат да се отделят променливи, които да не се преобразуват (в случая var1)
  `dat %>% gather("key", "value",-var1)`

- **spread**: преобразува таблица, в която са премахнати колоните, съдържащи имената на променливите и на техните стойности и са представени като отделни колони
  `dat %>% spread(key, value)`

Основни графики с ggplot2

- **Точкова**

```
ggplot(dat, aes(x=value1, y=value2)) + geom_point()
```

- **Линия**

```
ggplot(dat, aes(x=value1, y=value2)) + geom_line()
```

- **Колони**

  Една върху друга:

```
ggplot(dat, aes(x=variable, y=value,fill=variable2)) + geom_bar(stat = "identity
```

  Една до друга:

```
ggplot(dat, aes(x=variable, y=value,fill=variable2)) + geom_bar(stat = "identity
```

- **Кръгова диаграма (piechart)**

```
ggplot(dat, aes(x = "", y =value, fill=variable2)) +
geom_bar(stat = "identity")+
coord_polar("y")
```

- **Боксплот**

```
ggplot(dat, aes(x=1,y=value))+geom_boxplot()
```

- **Хистограма**

```
ggplot(dat, aes(value))+geom_histogram(bins=20,aes(y=..density..))
```

- **Хистограма и емпирична плътност**

```
ggplot(dat, aes(value))+
  geom_histogram(bins=20,aes(y=..density..))+
  geom_density(alpha=.2)
```

- **Q-Q диаграма**

```
ggplot(dat, aes(sample=value))+stat_qq()+ stat_qq_line()
```

- **Олекотена тема на графиката**: добавяме към кода `+theme_light()`

Разпределения

Разпределение	**Функция на разпределение**	**Обратна функция на разпределение**	**Плътност**	**Извадка от съотв. разпр-е**	**Други аргументи на функциите**
Дискретни разпределения					
Бернули	pbern(q,prob)	qbern(p,prob)	dbern(x,prob)	rbern(n,prob)	prob: вероятност за успех
Биномно	pbinom(q,size,prob)	qbinom(p,size,prob)	dbinom(x,size,prob)	rbinom(n,size,prob)	size: брой опити prob: вероятност за успех
Геометрично	pgeom(q,prob)	qgeom(p,prob)	dgeom(x,prob)	rgeom(n,prob)	prob: вероятност за успех
Хипергеометрично	phyper(q,m,nn,k)	qhyper(p,m,nn,k)	dhyper(x,m,nn,k)	rhyper(n,m,nn,k)	m: брой на топките от желан цвят nn: брой на топките от другия цвят k: брой изтеглени топки
Поасоново	ppois(q,lambda)	qpois(p,lambda)	dpois(x,lambda)	rpois(n,lambda)	lambda: параметър на разпр-ето
Нормално	pnorm(q,mean,sd)	qnorm(p,mean,sd)	dnorm(x,mean,sd)	rnorm(n,mean,sd)	mean: средна стойност sd: стандартно отклонение
Равномерно	punif(q,a,b)	punif(q,a,b)	punif(q,a,b)	punif(q,a,b)	a: долна граница b: горна граница
Експоненциално	pexp(q,rate)	qexp(p,rate)	dexp(x,rate)	rexp(n,rate)	rate: интензивност
Гама	pgamma(q,shape,scale)	qgamma(p,shape,scale)	dgamma(x,shape,scale)	rgamma(n,shape,scale)	shape: парам. за форма scale: парам. за мащаб
χ^2	pchiq(q,df)	qchiq(p,df)	dchiq(x,df)	rchiq(n,df)	df: степени на свобода
t	pt(q,df)	qt(p,df)	dt(x,df)	rt(n,df)	df: степени на свобода

Други полезни функции в R, свързани с разпределения:

- `sample(x,size,replace=FALSE,prob=NULL)`: съставя извадка от x с размер $size$, със или без заместване, с еднаква или предварително определна вероятност за сбъдване на всеки изход.;
- `kurtosis(X)` от пакета 'e1071': пресмята ексцеса на разпределението;
- `skewness(X)` от пакета 'e1071': пресмята асиметрията на разпределението.

Проверка на хипотези

Полезни функции в R:

- `shapiro.test(x)`: Прилага теста за нормалност на Шапиро и Уилк. H_0: разпределението на данните се различава от нормалното, т.е. ако p-стойностите са под нивото на значимост (най-често 0.05), можем да твърдим, че разпределението на данните не е статистически значимо различно от нормалното.
- `t.test(x,mu=mu0,`
  `alternative=‘‘two.sided’’, ‘‘greater’’ или ‘‘less’’)`: t-тест за една извадка от данни. H_0: истинската средна стойност е mu0.
- `prop.test(x,n,p)`: тест за вероятност за успех с нулева хипотеза, че вероятността за успех е p. С x е означен броят на успехите, а с n – големината на извадката. И тук може да се използва аргумента "alternative".
- `wilcox.test`: (на англ. Wilcoxon-Mann–Whitney или Wilcoxon rank-sum test) е непараметричен тест с нулева хипотеза, че разпределенията на две групи са един и същи. Практически този тест е сходен на t-теста, но се използва, когато разпределението на случайната величина не е нормално.