

# Exercise 12 - Multiple regression

Zoltan Kekecs

20 October, 2020

## Contents

<b>1</b>	<b>Tobbszoros regresszio</b>	<b>2</b>
1.1	Abstract . . . . .	2
1.2	Package-ek betoltese . . . . .	2
1.3	Az adatfajl betoltese: Lakasarak adattabla . . . . .	2
1.4	Adatellenoryes . . . . .	2
1.5	Tobbszoros regresszio . . . . .	11

# 1 Tobbszoros regresszio

## 1.1 Abstract

Ennek a gyakorlatnak az a celja hogy az egyszeru regresszirol szerzett tudast altalanositsuk olyan esetekre, ahol tobb prediktor (bejoslo valtozo) is szerepel a modellben.

Ennek a dokumentumnak a legfrissebb valtozatot megtalalod itt: <https://osf.io/e23by/>

## 1.2 Package-ek betoltese

A kovetkezo package-ek betoltesere lesz szukseg:

```
library(car) # for scatter3d
library(psych) # for describe
library(lm.beta) # for lm.beta
library(tidyverse) # for tidy format
library(gridExtra) # for grid.arrange
```

### 1.2.1 Sajat script betoltese

Ez a saját funkció arra való hogy a regressziós modell eredményeit táblázatban megjelenítsük. A funkció tartalmát nem szükséges megérteni.

```
coef_table = function(model){
  require(lm.beta)
  mod_sum = summary(model)
  mod_sum_p_values = as.character(round(mod_sum$coefficients[,4], 3))
  mod_sum_p_values[mod_sum_p_values != "0" & mod_sum_p_values != "1"] = substr(mod_sum_p_values[mod_sum_p_values != "0" & mod_sum_p_values != "1"], 1, 3)
  mod_sum_p_values[mod_sum_p_values == "0"] = "<.001"

  mod_sum_table = cbind(as.data.frame(round(cbind(coef(model), confint(model), c(0, lm.beta(model)$stan),
  names(mod_sum_table) = c("b", "95%CI lb", "95%CI ub", "Std.Beta", "p-value")
  mod_sum_table["(Intercept)", "Std.Beta"] = "0"
  return(mod_sum_table)
}
```

## 1.3 Az adatfajl betoltese: Lakasarak adattabla

Ebben a gyakorlatban lakasok és házak árát fogjuk megbecsülni.

Egy **Kaggle**-ról származó adatbázist használunk, melyben olyan adatok szerepelnek, melyeket valószínűsíthetően alkalmasak **lakasok eladási árának bejósolására**. Az adatbázisban az USA Kings County-ból származnak az adatok (Seattle és környéke).

Az adatbázisnak csak egy kis részt használjuk ( $N = 200$ ).

```
data_house = read_csv("https://raw.githubusercontent.com/kekecsz/PSZB17-210-Data-analysis-seminar/master/data_house.csv")
```

## 1.4 Adatellenőrzés

Mindig ellenőrizd az adatok strukturáját és integritását.

Eloszor átváltjuk az USA dollár-t millió forint mértékegységre, és a negyzetlab adatokat negyzetmeterre.

```
data_house %>%
  summary()
```

```
##           id           date           price
## Min.      :1.600e+07   Min.      :2014-05-06 00:00:00   Min.      : 153503
## 1st Qu.:1.885e+09   1st Qu.:2014-07-22 18:00:00   1st Qu.: 299250
## Median :3.521e+09   Median :2014-10-29 12:00:00   Median : 425000
## Mean    :4.113e+09   Mean    :2014-11-08 10:19:12   Mean    : 453611
## 3rd Qu.:6.424e+09   3rd Qu.:2015-02-28 00:00:00   3rd Qu.: 550000
## Max.    :9.819e+09   Max.    :2015-05-12 00:00:00   Max.    :1770000
## bedrooms   bathrooms   sqft_living   sqft_lot   floors
## Min.      :1.00   Min.      :0.75   Min.      : 590   Min.      : 914   Min.      :1.000
## 1st Qu.:3.00   1st Qu.:1.00   1st Qu.:1240   1st Qu.: 4709   1st Qu.:1.000
## Median :3.00   Median :1.75   Median :1620   Median : 7270   Median :1.000
## Mean    :2.76   Mean    :1.85   Mean    :1728   Mean    :12985   Mean    :1.472
## 3rd Qu.:3.00   3rd Qu.:2.50   3rd Qu.:1985   3rd Qu.:10187   3rd Qu.:2.000
## Max.    :3.00   Max.    :3.50   Max.    :4380   Max.    :217800   Max.    :3.000
## waterfront   view   condition   grade   sqft_above
## Min.      :0.000   Min.      :0.000   Min.      :3.00   Min.      : 5.00   Min.      : 590
## 1st Qu.:0.000   1st Qu.:0.000   1st Qu.:3.00   1st Qu.: 7.00   1st Qu.:1090
## Median :0.000   Median :0.000   Median :3.00   Median : 7.00   Median :1375
## Mean    :0.005   Mean    :0.145   Mean    :3.42   Mean    : 7.36   Mean    :1544
## 3rd Qu.:0.000   3rd Qu.:0.000   3rd Qu.:4.00   3rd Qu.: 8.00   3rd Qu.:1862
## Max.    :1.000   Max.    :4.000   Max.    :5.00   Max.    :11.00   Max.    :4190
## sqft_basement   yr_built   yr_renovated   zipcode
## Min.      : 0.0   Min.      :1900   Min.      : 0.00   Min.      :98001
## 1st Qu.: 0.0   1st Qu.:1946   1st Qu.: 0.00   1st Qu.:98033
## Median : 0.0   Median :1968   Median : 0.00   Median :98065
## Mean    :184.1   Mean    :1968   Mean    : 79.98   Mean    :98078
## 3rd Qu.:315.0   3rd Qu.:1993   3rd Qu.: 0.00   3rd Qu.:98117
## Max.    :1600.0   Max.    :2015   Max.    :2014.00   Max.    :98199
## lat           long           sqft_living15   sqft_lot15
## Min.      :47.18   Min.      :-122.5   Min.      : 740   Min.      : 914
## 1st Qu.:47.49   1st Qu.: -122.3   1st Qu.:1438   1st Qu.: 5000
## Median :47.58   Median : -122.2   Median :1715   Median : 7222
## Mean    :47.57   Mean    : -122.2   Mean    :1793   Mean    :11225
## 3rd Qu.:47.68   3rd Qu.: -122.1   3rd Qu.:2072   3rd Qu.:10028
## Max.    :47.78   Max.    : -121.7   Max.    :3650   Max.    :208652
## has_basement
## Length:200
## Class :character
## Mode :character
##
##
##
```

```
data_house = data_house %>%
  mutate(price_HUF = (price * 293.77)/1000000,
         sqm_living = sqft_living * 0.09290304,
         sqm_lot = sqft_lot * 0.09290304,
         sqm_above = sqft_above * 0.09290304,
         sqm_basement = sqft_basement * 0.09290304,
         sqm_living15 = sqft_living15 * 0.09290304,
         sqm_lot15 = sqft_lot15 * 0.09290304
  )
```

Egyszeru leiro statisztikak es abrak.

Kezdetben a lakások arat a **sqm\_living** (a lakás lakóreszenek alapterülete négyzetmeterben), és a **grade** (a lakás általános minősítése a King County grading system szerint, ami a lakás minőséget, pozícióját, a ház minőséget stb. is tartalmazza) prediktorok felhasználásával jósoljuk majd be. Később a **has\_basement** (tartozik-e a lakáshoz pince) változót is használjuk majd. Szóval fókuszáljunk ezekre a változókra az adatellenőrzés során.

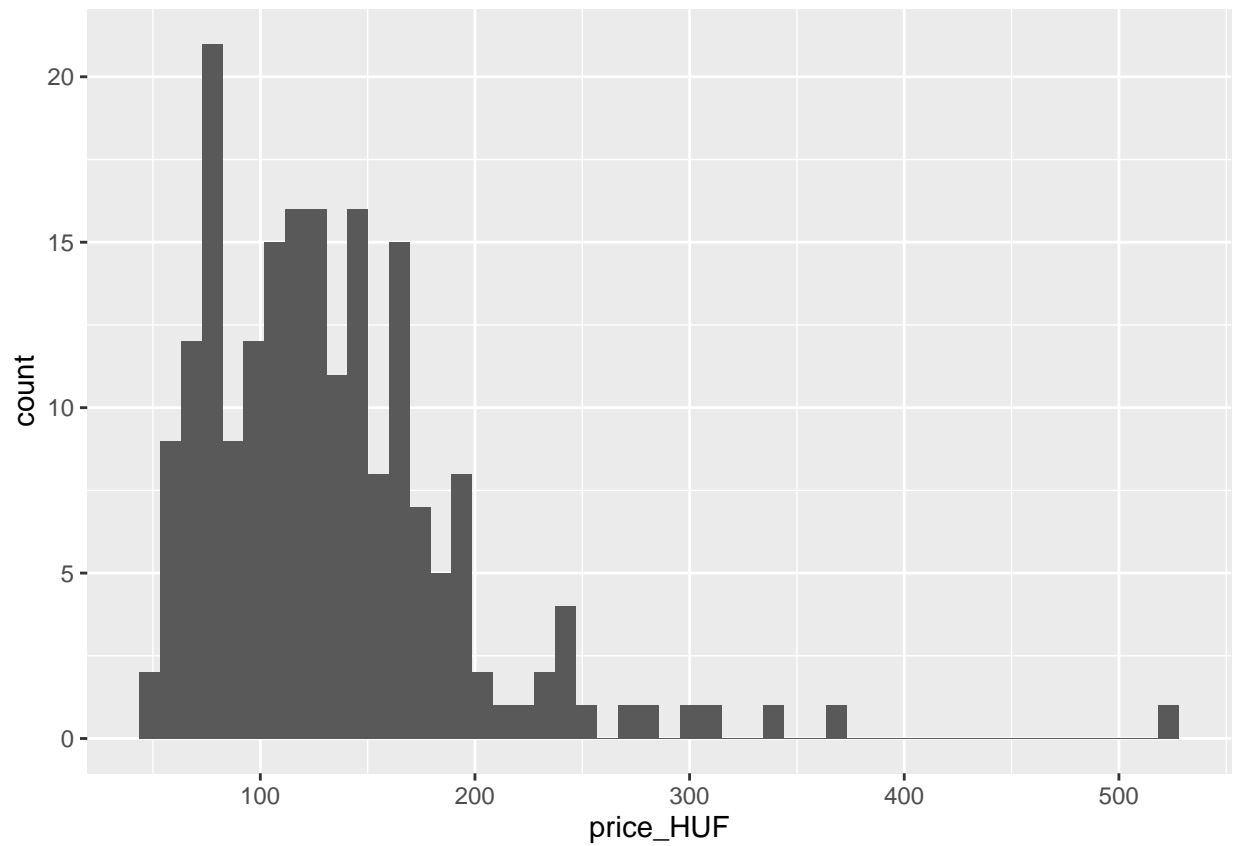
```
# leíró statisztikák
describe(data_house)
```

```
## Warning in describe(data_house): NAs introduced by coercion
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

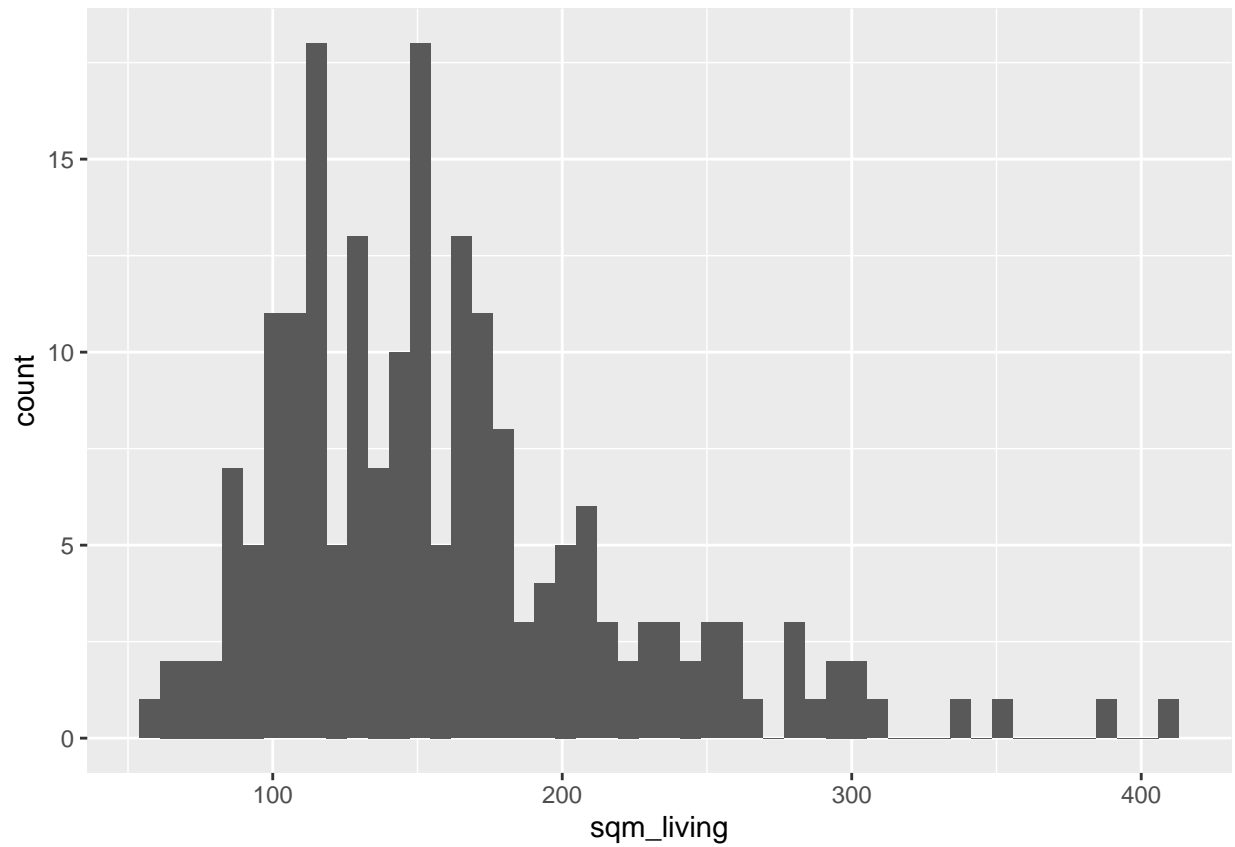
##          vars    n      mean      sd      median      trimmed
## id          1 200 4112747619.38 2.746825e+09 3520875095.00 3956631056.34
## date        2 200         NaN         NA         NA         NaN
## price       3 200   453610.89 2.111943e+05   425000.00   427743.09
## bedrooms    4 200      2.76 4.500000e-01      3.00      2.84
## bathrooms   5 200      1.85 6.600000e-01      1.75      1.83
## sqft_living  6 200    1727.61 6.629200e+02    1620.00    1650.86
## sqft_lot    7 200   12985.36 2.773609e+04    7270.00    7728.61
## floors      8 200      1.47 5.500000e-01      1.00      1.42
## waterfront  9 200      0.00 7.000000e-02      0.00      0.00
## view       10 200      0.14 6.000000e-01      0.00      0.00
## condition  11 200      3.42 6.200000e-01      3.00      3.31
## grade      12 200      7.36 1.020000e+00      7.00      7.29
## sqft_above  13 200    1543.51 6.298700e+02    1375.00    1464.11
## sqft_basement 14 200     184.10 3.250700e+02      0.00    110.75
## yr_built    15 200    1967.64 2.956000e+01    1968.50    1969.17
## yr_renovated 16 200      79.98 3.928100e+02      0.00      0.00
## zipcode     17 200    98077.98 5.407000e+01    98065.00    98074.58
## lat         18 200      47.57 1.400000e-01      47.58      47.58
## long        19 200    -122.20 1.700000e-01    -122.25    -122.22
## sqft_living15 20 200    1793.34 5.127800e+02    1715.00    1742.61
## sqft_lot15   21 200   11225.47 1.966363e+04    7222.00    7559.91
## has_basement* 22 200         NaN         NA         NA         NaN
## price_HUF    23 200     133.26 6.204000e+01    124.85    125.66
## sqm_living   24 200     160.50 6.159000e+01    150.50    153.37
## sqm_lot      25 200    1206.38 2.576770e+03     675.41    718.01
## sqm_above    26 200     143.40 5.852000e+01    127.74    136.02
## sqm_basement 27 200      17.10 3.020000e+01      0.00     10.29
## sqm_living15 28 200     166.61 4.764000e+01    159.33    161.89
## sqm_lot15    29 200    1042.88 1.826810e+03     670.95    702.34
##          mad      min      max      range      skew
## id          2.981805e+09 16000200.00 9818700320.00 9.802700e+09 0.45
## date         NA         Inf      -Inf      -Inf      NA
## price       1.853250e+05  153503.00   1770000.00 1.616497e+06 2.02
## bedrooms    0.000000e+00      1.00      3.00 2.000000e+00 -1.53
## bathrooms   1.110000e+00      0.75      3.50 2.750000e+00 0.12
## sqft_living  5.633900e+02     590.00    4380.00 3.790000e+03 1.20
```

## sqft_lot	3.977820e+03	914.00	217800.00	2.168860e+05	6.16
## floors	0.000000e+00	1.00	3.00	2.000000e+00	0.74
## waterfront	0.000000e+00	0.00	1.00	1.000000e+00	13.93
## view	0.000000e+00	0.00	4.00	4.000000e+00	4.27
## condition	0.000000e+00	3.00	5.00	2.000000e+00	1.18
## grade	1.480000e+00	5.00	11.00	6.000000e+00	0.62
## sqft_above	5.115000e+02	590.00	4190.00	3.600000e+03	1.29
## sqft_basement	0.000000e+00	0.00	1600.00	1.600000e+03	1.91
## yr_built	3.484000e+01	1900.00	2015.00	1.150000e+02	-0.32
## yr_renovated	0.000000e+00	0.00	2014.00	2.014000e+03	4.66
## zipcode	6.227000e+01	98001.00	98199.00	1.980000e+02	0.42
## lat	1.500000e-01	47.18	47.78	6.000000e-01	-0.56
## long	1.600000e-01	-122.45	-121.73	7.200000e-01	0.79
## sqft_living15	4.596100e+02	740.00	3650.00	2.910000e+03	0.94
## sqft_lot15	3.624960e+03	914.00	208652.00	2.077380e+05	6.61
## has_basement*	NA	Inf	-Inf	-Inf	NA
## price_HUF	5.444000e+01	45.09	519.97	4.748800e+02	2.02
## sqm_living	5.234000e+01	54.81	406.92	3.521000e+02	1.20
## sqm_lot	3.695500e+02	84.91	20234.28	2.014937e+04	6.16
## sqm_above	4.752000e+01	54.81	389.26	3.344500e+02	1.29
## sqm_basement	0.000000e+00	0.00	148.64	1.486400e+02	1.91
## sqm_living15	4.270000e+01	68.75	339.10	2.703500e+02	0.94
## sqm_lot15	3.367700e+02	84.91	19384.41	1.929949e+04	6.61
##	kurtosis	se			
## id	-1.04	194229829.92			
## date	NA	NA			
## price	7.84	14933.69			
## bedrooms	1.15	0.03			
## bathrooms	-0.88	0.05			
## sqft_living	1.78	46.88			
## sqft_lot	40.75	1961.24			
## floors	-0.31	0.04			
## waterfront	193.03	0.00			
## view	17.86	0.04			
## condition	0.28	0.04			
## grade	1.00	0.07			
## sqft_above	1.90	44.54			
## sqft_basement	3.43	22.99			
## yr_built	-0.96	2.09			
## yr_renovated	19.81	27.78			
## zipcode	-0.85	3.82			
## lat	-0.66	0.01			
## long	-0.22	0.01			
## sqft_living15	0.88	36.26			
## sqft_lot15	54.45	1390.43			
## has_basement*	NA	NA			
## price_HUF	7.84	4.39			
## sqm_living	1.78	4.35			
## sqm_lot	40.75	182.20			
## sqm_above	1.90	4.14			
## sqm_basement	3.43	2.14			
## sqm_living15	0.88	3.37			
## sqm_lot15	54.45	129.18			

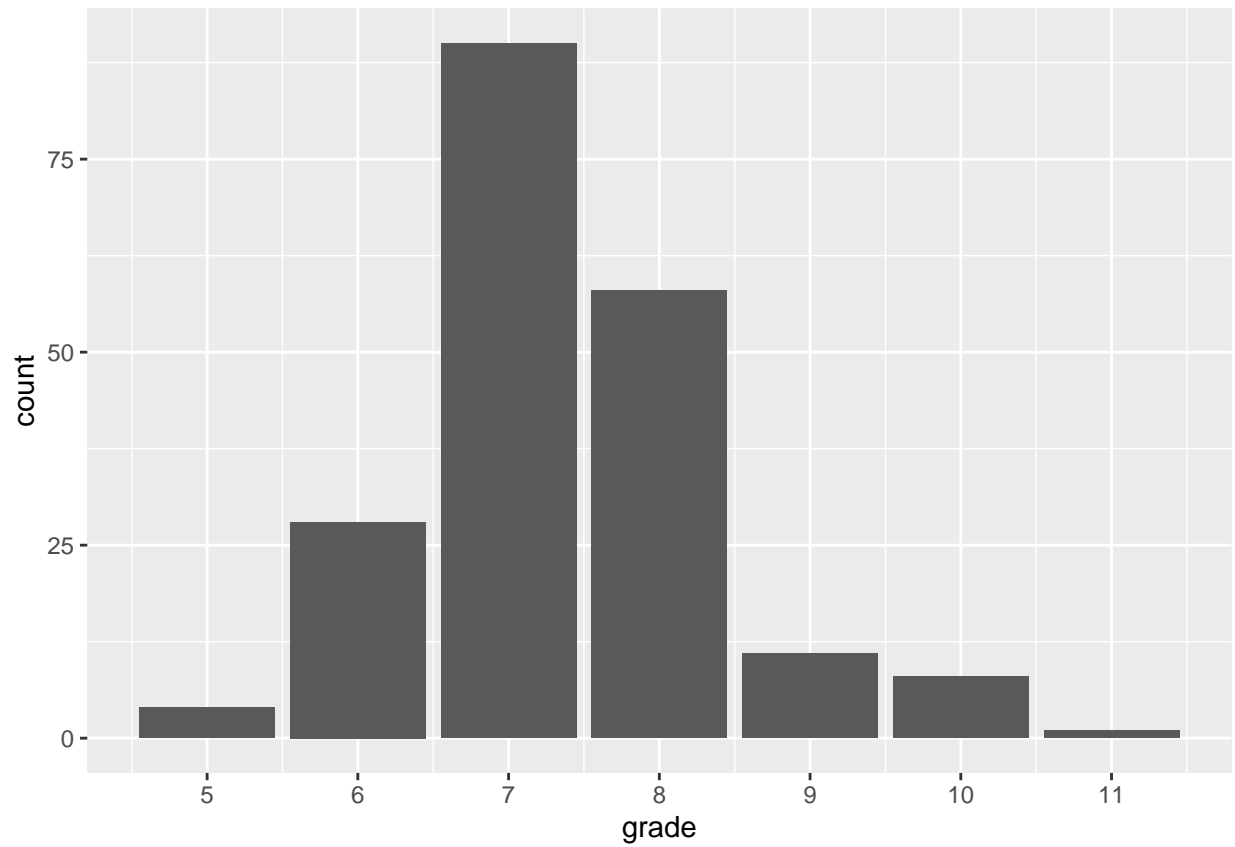
```
# hisztogramok
data_house %>%
  ggplot() +
  aes(x = price_HUF) +
  geom_histogram( bins = 50)
```



```
data_house %>%
  ggplot() +
  aes(x = sqm_living) +
  geom_histogram( bins = 50)
```

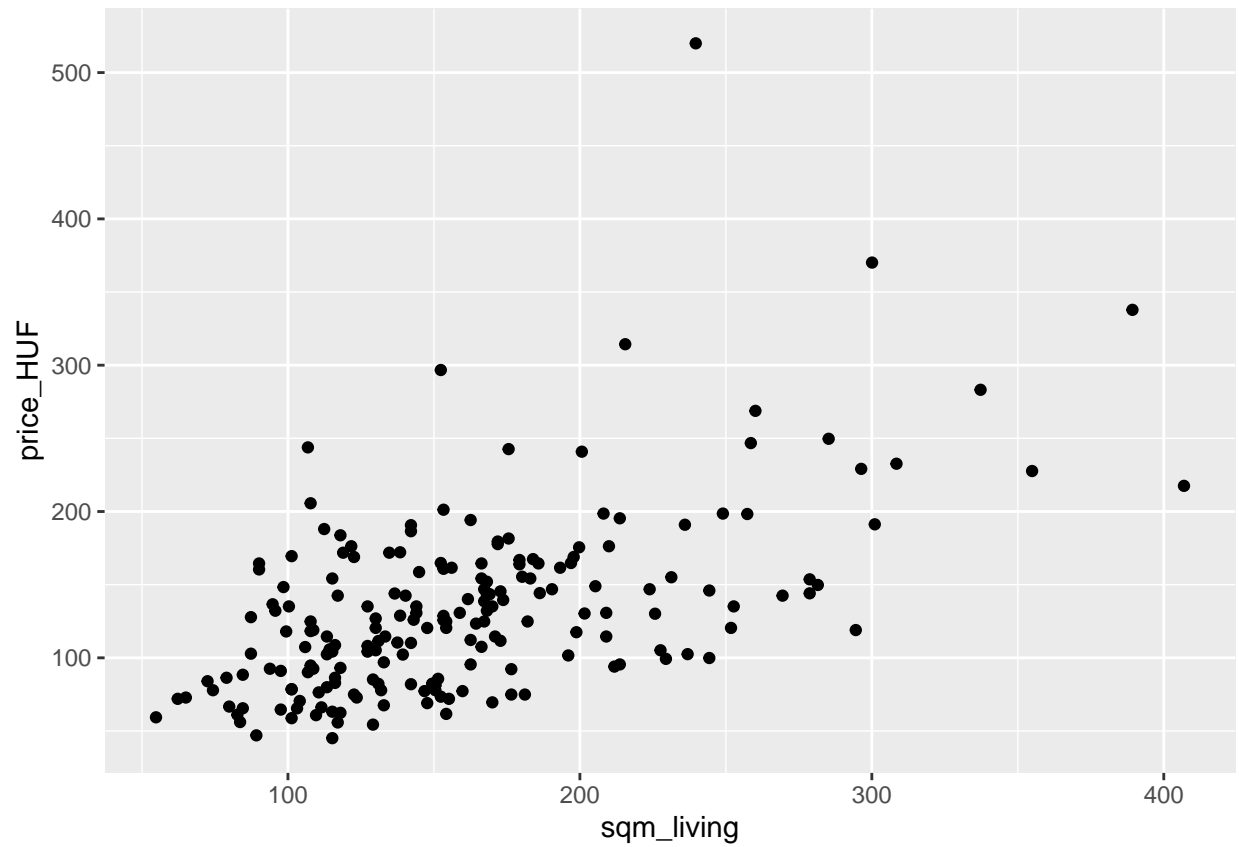


```
data_house %>%  
  ggplot() +  
  aes(x = grade) +  
  geom_bar() +  
  scale_x_continuous(breaks = 4:12)
```

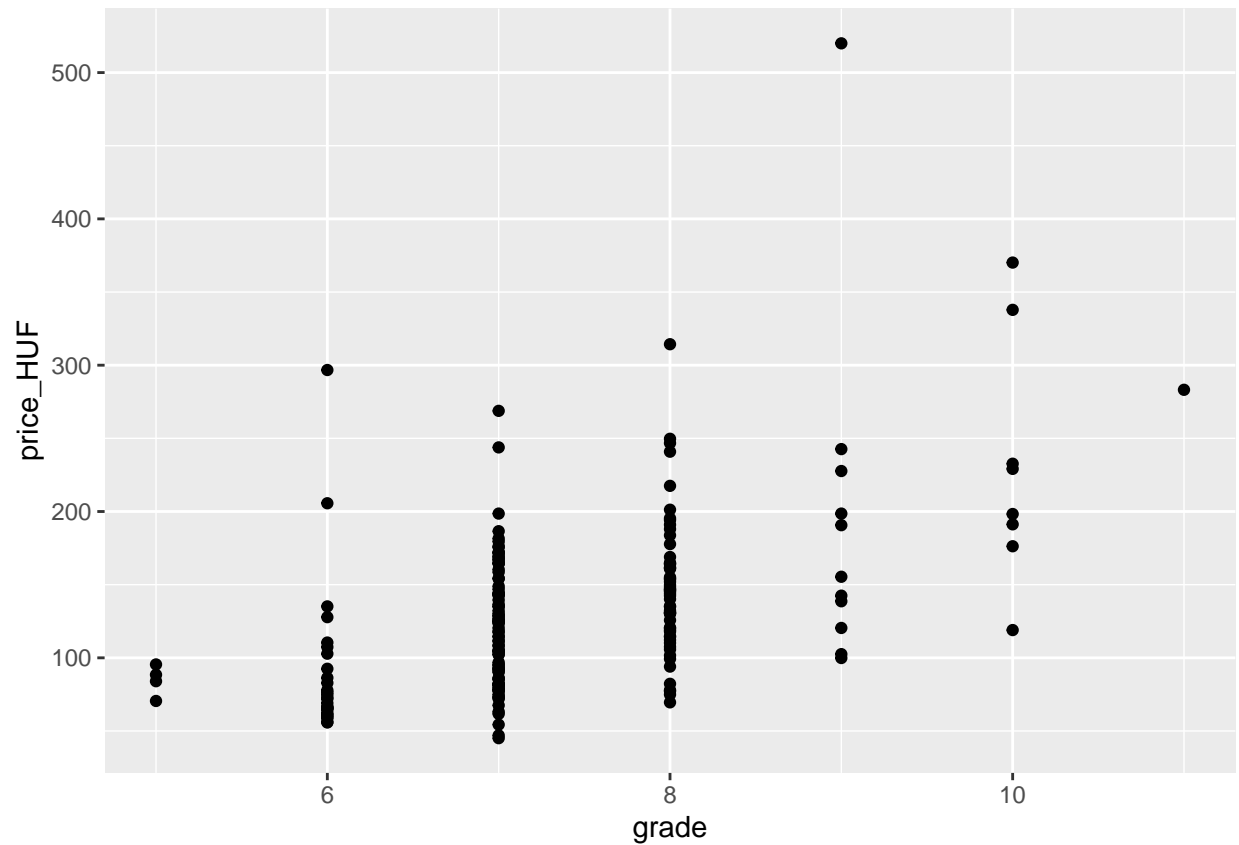


```
# scatterplot
data_house %>%
  ggplot() +
  aes(x = sqm_living, y = price_HUF) +
  geom_point()
```





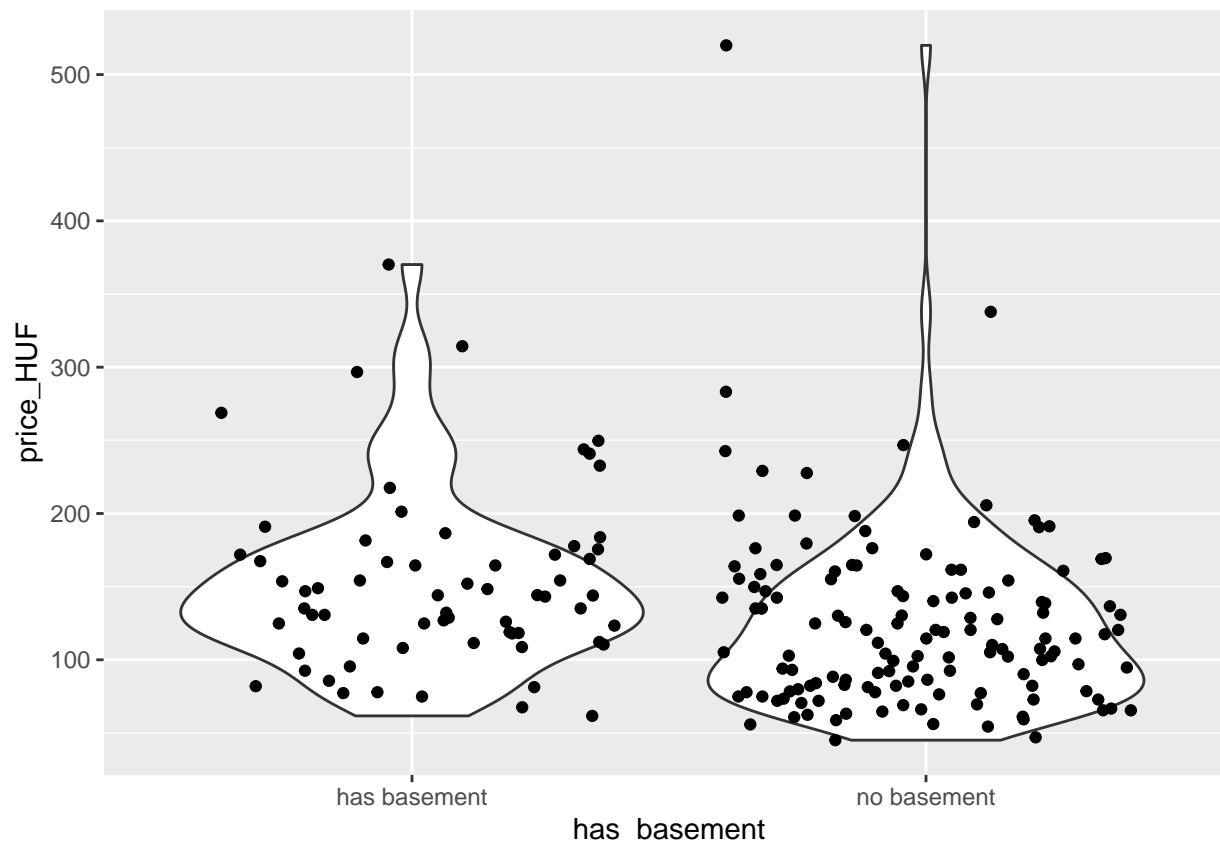
```
data_house %>%  
  ggplot() +  
  aes(x = grade, y = price_HUF) +  
  geom_point()
```



```
# leiro statisztika
table(data_house$has_basement)
```

```
##
## has basement  no basement
##           65          135
```

```
# violin plot
data_house %>%
  ggplot() +
  aes(x = has_basement, y = price_HUF) +
  geom_violin() +
  geom_jitter()
```



## 1.5 Tobbszoros regresszio

### 1.5.1 A regressziós modell felepítése (fitting a regression model)

A többszoros regressziós modellt ugyan úgy epeitjük mint az egyszeru regressziós modellt, csak csak több prediktort is betehetünk a modellbe. Ezeket a prediktorváltozókat + jellen választjuk el egymástól a regressziós formulában.

Alább **price\_HUF** a bejosolt változó, és a **sqm\_living** és a **grade** a prediktorok.

```
mod_house1 = lm(price_HUF ~ sqm_living + grade, data = data_house)
```

A regressziós egyenletet a modell objektumon keresztül érhetjük el:

```
mod_house1
```

```
##
## Call:
## lm(formula = price_HUF ~ sqm_living + grade, data = data_house)
##
## Coefficients:
## (Intercept)    sqm_living      grade
##    -51.2305      0.3768     16.8485
```

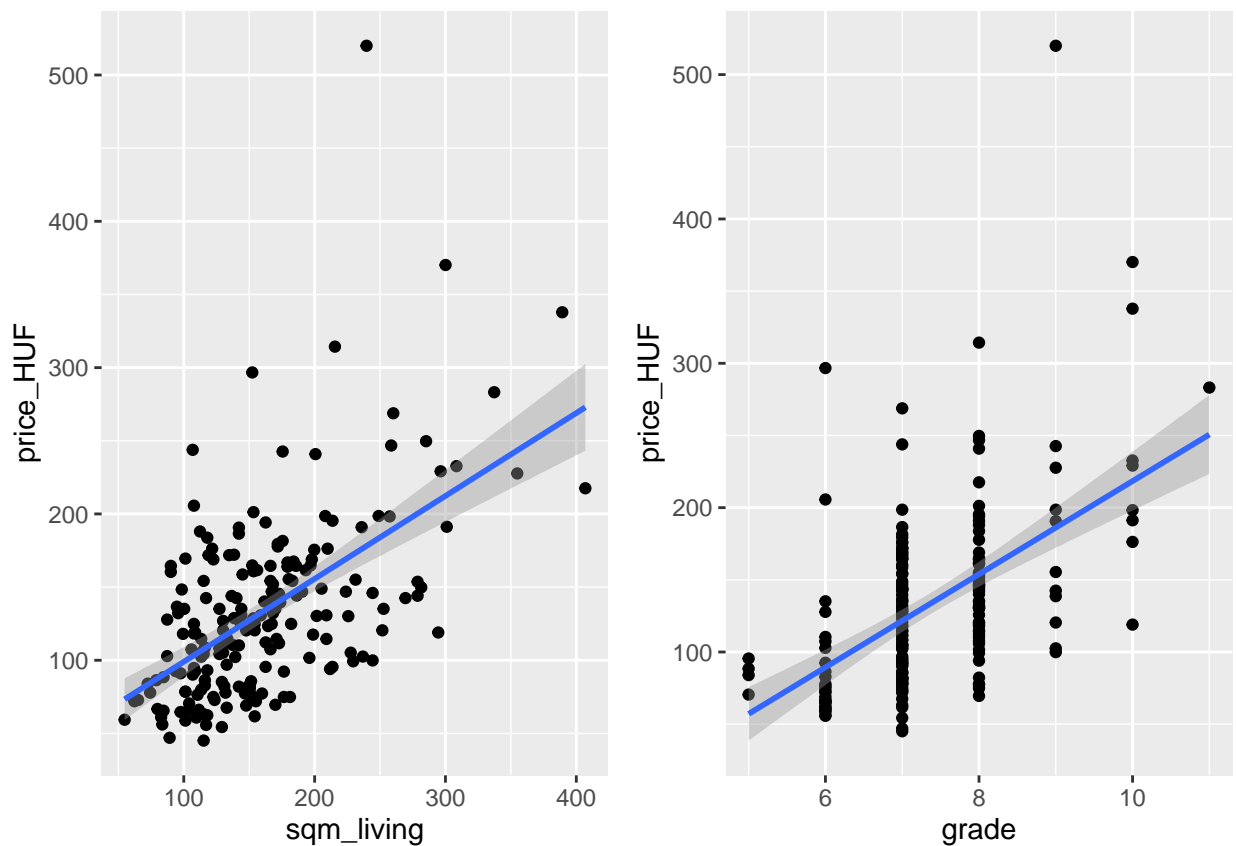
A többszoros regressziós modellek vizualizációja nem olyan egyértelmű mint az egyszeru regressziós modelleke.

Az egyik megoldás hogy a páronkénti összefüggéseket vizualizáljuk egyenként, de ez nem ragadja meg a modell többváltozós jellegét.

```
# scatterplot
plot1 = data_house %>%
  ggplot() +
  aes(x = sqm_living, y = price_HUF) +
  geom_point()+
  geom_smooth(method = "lm")

plot2 = data_house %>%
  ggplot() +
  aes(x = grade, y = price_HUF) +
  geom_point()+
  geom_smooth(method = "lm")

grid.arrange(plot1, plot2, nrow = 1)
```



Egy alternativa hogy egy haromdimenzios abran abrazoljuk a regresszios sikot. Bar ez szepen nez ki, de nem tul hasznos, es ez is csak ket prediktorvaltozoig mukodik, harom es több prediktor eseten mar egy tobbdimenzios terben kepzelhető csak el a regresszios felület, ezért a vizualizacióra általában mégis az paronkenti scatterplot-ot szoktuk használni.

```
# plot the regression plane (3D scatterplot with regression plane)
scatter3d(price_HUF ~ sqm_living + grade, data = data_house)
```

### 1.5.2 Becsles (prediction)

Ugyan úgy ahogy az egyszeru regresszional, itt is kerhetjuk a prediktorok bizonyos uj ertekekeire a kimeneti valtozo ertekeinek megbecsleset a `predict()` fuggveny segitsegevel.

Fontos, hogy a prediktorok értékeit egy data.frame vagy tibble formátumban kell megadnunk, és a prediktor-  
változók változóneveinek meg kell egyeznie a regressziós modellben használt változónevekkel.

```
sqm_living = c(60, 60, 100, 100)
grade = c(6, 9, 6, 9)
newdata_to_predict = as.data.frame(cbind(sqm_living, grade))
predicted_price_HUF = predict(mod_house1, newdata = newdata_to_predict)

cbind(newdata_to_predict, predicted_price_HUF)
```

```
##   sqm_living grade predicted_price_HUF
## 1         60     6          72.47102
## 2         60     9          123.01660
## 3        100     6           87.54459
## 4        100     9          138.09017
```

### 1.5.3 Hogyan közöljük az eredményeinket egy kutatási jelentésben

Egy kutatási jelentésben (pl. cikk, muhelymunka, ZH) a következó információkat kell leírni a regressziós  
modellrol:

Eloszor is le kell írni a regressziós **modell tulajdonságait** (általában a “Modszerek” részben):

“Egy lineáris regressziós modellt illesztettem, melyben a lakás arát (millio HUF-ban) a lakás lakoreszenek  
területével ( $m^2$ -ben) és a lakás King County lakás-minosites értékevel becsultem meg.”

“I built a linear regression model in which I predicted housing price (in million HUF) with the size of the living  
area (in  $m^2$ ) and King County housing grade as predictors.”

Ezután a **teljes modell bejoslasi hatekonysagát** kell jellemezni. Ezt a modellhez tartozó adjusted  
 $R^2$  érték (modosított  $R^2$ ), és a modell-t a null-moddellel összehasonlító anova F-tesztjenek statisztikáinak  
megadásával szoktuk tenni (F-érték, df, p-érték). Mindezen információt a summary() funkcióval tudjuk  
lekerdeezni. A modell illeszkedését az AIC (Akaike information criterion) értékkel is szoktuk jellemezni, amit  
az AIC() funkció ad meg.

Az APA publikációs kézikönyv alapján minden számot két tizedesjegy pontossággal kell megadni, kivéve a p  
értéket, amit három tizedesjegy pontossággal.

```
sm = summary(mod_house1)
sm

##
## Call:
## lm(formula = price_HUF ~ sqm_living + grade, data = data_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -109.26  -29.55   -6.79   19.65   329.24
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -51.2305    27.9831  -1.831  0.068646 .
## sqm_living     0.3768     0.0783   4.813  2.96e-06 ***
## grade        16.8485     4.7158   3.573  0.000444 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.96 on 197 degrees of freedom
```

```
## Multiple R-squared:  0.358, Adjusted R-squared:  0.3515
## F-statistic: 54.94 on 2 and 197 DF,  p-value: < 2.2e-16
```

```
AIC(mod_house1)
```

```
## [1] 2137.057
```

Vagyis az “Eredmenyek” részben így írunk a fenti pelda eredményeiről:

“A többszörös regressziós modell mely tartalmazta a lakóterület és a lakás minősítés prediktorokat hatékonyabban tudta bejósolni a lakás árát mint a null modell. A modell a lakásár varianciájának 35.15%-át magyarázta ( $F(2, 197) = 54.94$ ,  $p < .001$ ,  $\text{Adj. } R^2 = 0.35$ ,  $\text{AIC} = 2137.06$ ).”

Ezen felül meg kell adnunk a **regressziós egyenletre és az egyes prediktorok becsléshez való hozzájárulására vontkozó adatokat**. Ezt általában egy összefoglaló táblázatban szoktuk megadni, melyben a következő adatok szerepelnek prediktoronként:

- regressziós együttható (regression coefficients, estimates) - `summary()`
- az együtthatókhoz tartozó konfidencia intervallum (coefficient confidence intervals) - `confint()`
- standard beta értékek (standardized beta values) - `lm.beta()` az `lm.beta` package-ben
- a t-teszthez tartozó p-érték (p-values of the t-test) - `summary()`

```
confint(mod_house1)
```

```
##                2.5 %      97.5 %
## (Intercept) -106.415417  3.9543979
## sqm_living    0.222427  0.5312516
## grade         7.548542 26.1485134
```

```
lm.beta(mod_house1)
```

```
##
## Call:
## lm(formula = price_HUF ~ sqm_living + grade, data = data_house)
##
## Standardized Coefficients::
## (Intercept)  sqm_living      grade
##  0.0000000    0.3740724    0.2776905
```

A vegso táblázat valahogy így néz majd ki (ennek az elkészítéséhez a fenti `coef_table()` saját funkciót használtam. Nem fontos ezt használni, manuálisan is ki lehet irogatni az eredményeket a különbozo táblázatokból.):

```
##                b 95%CI lb 95%CI ub Std.Beta p-value
## (Intercept) -51.23 -106.42   3.95      0    .069
## sqm_living   0.38   0.22   0.53   0.37  <.001
## grade       16.85   7.55  26.15   0.28  <.001
```

#### 1.5.4 regressziós együttható értelmezése

A regressziós együtthatót úgy lehet értelmezni, hogy a prediktor értékenek egy ponttal való növekedése esetén a kimeneti változó értéke ennyivel változik. Pl. ha a `sqm_living`-hez tartozó regressziós együttható 0.38, az azt jelenti hogy minden egyes újabb négyzetmeter területnövekedés 0.38 millió forint árvaltozással jár.

#### 1.5.5 az intercept-hez tartozó regressziós együttható értelmezése

Az intercept együtthatója azt mutatja meg, hogy mi lenne a bejósolt (fuggo) változó becslés értéke, ha minden prediktor 0 értéket vesz fel. Ez nem mindig egy realis becslés, hiszen attól függoen hogy milyen prediktorokat használunk, lehet hogy egy adott prediktoron a 0 érték nem értelmes. Ettől függetlenül az intercept matematikai értelmezése mindig ugyan ez marad. Az intercept egyfajta allando érték, ami független a prediktorok értékeitől.

### 1.5.6 standard beta értelmezése

A regressziós együttható előnye, hogy a kimeneti változó mértekegységében van, és nagyon egyszerű értelmezni. Ezért ez egy “nyers” hatásmeret mutató. Viszont a hátránya, hogy az értéke a hozzá tartozó prediktor változó skáláján mozog. Ez azt jelenti, hogy az egyes együttható értékek nem könnyen összehasonlíthatók, mert a prediktorok más skálán mozognak. Pl. az `sqm_living` együtthatója alacsonyabb mint az `grade` együtthatója, de ez onmagában nem mond arról semmit, hogy melyik prediktornak van nagyobb szerepe a kimeneti változó bejósolásában, mert a `sqm_living` skálája sokkal kiterjedtebb (50-400 m<sup>2</sup>) mint a `grade` skálája (5-11).

Ahhoz, hogy össze tudjuk hasonlítani az egyes prediktorok becsléséhez hozzáadott értéket, a két együtthatót ugyan arra a skálára kell helyeznünk, amit standardizálással érhetünk el. A standard Beta egy ilyen standardizált mutató. Ez már direkt módon összehasonlítható a prediktorok között. Ebből már látszik, hogy a `sqm_living` hozzáadott értéke a `price_HUF` bejósoláshoz nagyobb mint a `grade` hozzáadott értéke.

Amikor több prediktor van, ez nem feltétlenül jelenti azt, hogy ha egyenként megvizsgáljuk a prediktorok korrelációját a kimeneti változóval, akkor ugyan ilyen összefüggést kapunk. Ez az együttható és a `std.Beta` érték a prediktor egész modellben betöltött szerepét jelöli, a többi prediktor bejósoló erejének lezámításával. Vagyis elképzeltethető, hogy egy prediktor onmagában jobban korrelál a kimeneti változóval mint bármelyik másik prediktor, viszont a modellben kisebb szerepet játszik, mert a többi prediktor ugyan azt a részt magyarázza a kimeneti változó varianciájának, mint ez a prediktor.

---

#### Gyakorlás

1. Építs egy többszörös lineáris regresszió modellt az `lm()` függvénnyel, amiben az **price\_HUF** a kimeneti változót becsüljük meg. Használhatod a **data\_house** adatbázisban szereplő bármelyik változót felhasználhatod a modellben, ami szerinted realisan hozzájárulhat a lakás árának meghatározásához.
  2. Határozd meg, hogy szignifikánsan jobb-e a modelled mint a null modell (a teljese modell F-tesztjeztartozó p-érték alapján)?
  3. Mekkora a teljes modell által bejósolt varianciaarány ( $\text{adj.R}^2$ )?
  4. Melyik az a prediktor, mely a legnagyobb hozzáadott értékkel bír a becslésben?
-