Linearis regresszio

Kekecs Zoltan

November 19, 2021

Ennek az oranak a celja hogy megismerkedjunk a linearis regresszioval, annak logikajaval, es az ertelmezesehez szukseges alapfogalmakkal. Eloszor az ugynevezett "egyszeru" linearis regressziot (simple regression) fogjuk megismerni, ahol egy bejoslo valtozo alapjan becsuljuk meg egy kimeneti valtozo erteket. Miutan megismertuk az egyszeru regresszioval, tovabbmegyunk a tobbszoros regressziora, ahol altalanositjuk az egyszeru regressziorol nyert tudast olyan esetekre, ahol tobb prediktor (bejoslo valtozo) is szerepel a modellben.

Package-ek betoltese

Betoltjuk a kovatkazo package-eket:

```
library(psych) # for describe
library(gsheet) # to read data from google sheets
```

```
## Warning: package 'gsheet' was built under R version 4.1.1
```

```
library(car) # for scatter3d
```

```
## Warning: package 'car' was built under R version 4.1.1
```

```
library(rgl) # for scatter3d
```

```
## Warning: package 'rgl' was built under R version 4.1.1
```

```
library(psych) # for describe
library(lm.beta) # for lm.beta
```

```
## Warning: package 'lm.beta' was built under R version 4.1.1
```

```
library(gridExtra) # for grid.arrange
library(tidyverse) # for tidy code
```

```
## Warning: package 'tibble' was built under R version 4.1.1
```

Sajat funckiok betoltese

Alabb ket sajat funkciot fogunk betolteni. Az error_plotter() funkciot arra hasznaljuk majd hogy a linearis regresszioban fennmarado (rezidualis) hibat vizualizaljuk. A coef_table() funkciot pedig arra nasznaljuk majd hogy tablazatot generaljunk az eredmenyekbol. A funkciok kodjat nem fontos megerteni, de erdemes oket betolteni hogy ponotosan reprodukalhasd az orai jegyzetben latottakat.

```
error_plotter <- function(mod, col = "black", x_var = NULL){
  mod_vars = as.character(mod$call[2])
  data = as.data.frame(eval(parse(text = as.character(mod$call[3]))))
  y = substr(mod_vars, 1, as.numeric(gregexpr(pattern ='~',mod_vars))-2)
  x = substr(mod_vars, as.numeric(gregexpr(pattern ='~',mod_vars))+2, nchar(mod_vars))

  data$pred = predict(mod)

  if(x == "1" & is.null(x_var)){x = "response_ID"
  data$response_ID = 1:nrow(data)} else if(x == "1"){x = x_var}

  plot(data[,y] ~ data[,x], ylab = y, xlab = x)
  abline(mod)

  for(i in 1:nrow(data)){
    clip(min(data[,x]), max(data[,x]), min(data[i,c(y,"pred")]), max(data[i,c(y,"pred")]))
    abline(v = data[i,x], lty = 2, col = col)
  }

}


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
  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)
}
```

Egyszeru linearis regresszio

Adatmenedzsment es adat bemutatasa 1

Adatok betoltese

Mondjuk, hogy egy turistak koreben gyakran latogatott cipoboltban dolgozunk, es mivel a vilagon sokfajta cipomeretet hasznalnak es az emberek gyakran nem tudjak a sajat europai cipomeretuket, szeretnenk a magassaguk alapjan megbecsulni, mekkora az europai cipomeretuk.

Az alabbi koddal betolthetjuk az adattablat, amiben a korabbi orakon felvett kerdoivekbol szerepelnek a magassag es cipomeret adatok.

```
mydata = read.csv("https://raw.githubusercontent.com/kekecsz/PSZB17-210-Data-analysis-seminar/master/se
```

Adatok ellenorzese

Szokas szerint az adatok ellenorzesevel kezdunk, pl. View(), describe(), es summary() funkciokkal.

```
# descriptive statistics
describe(mydata)
```

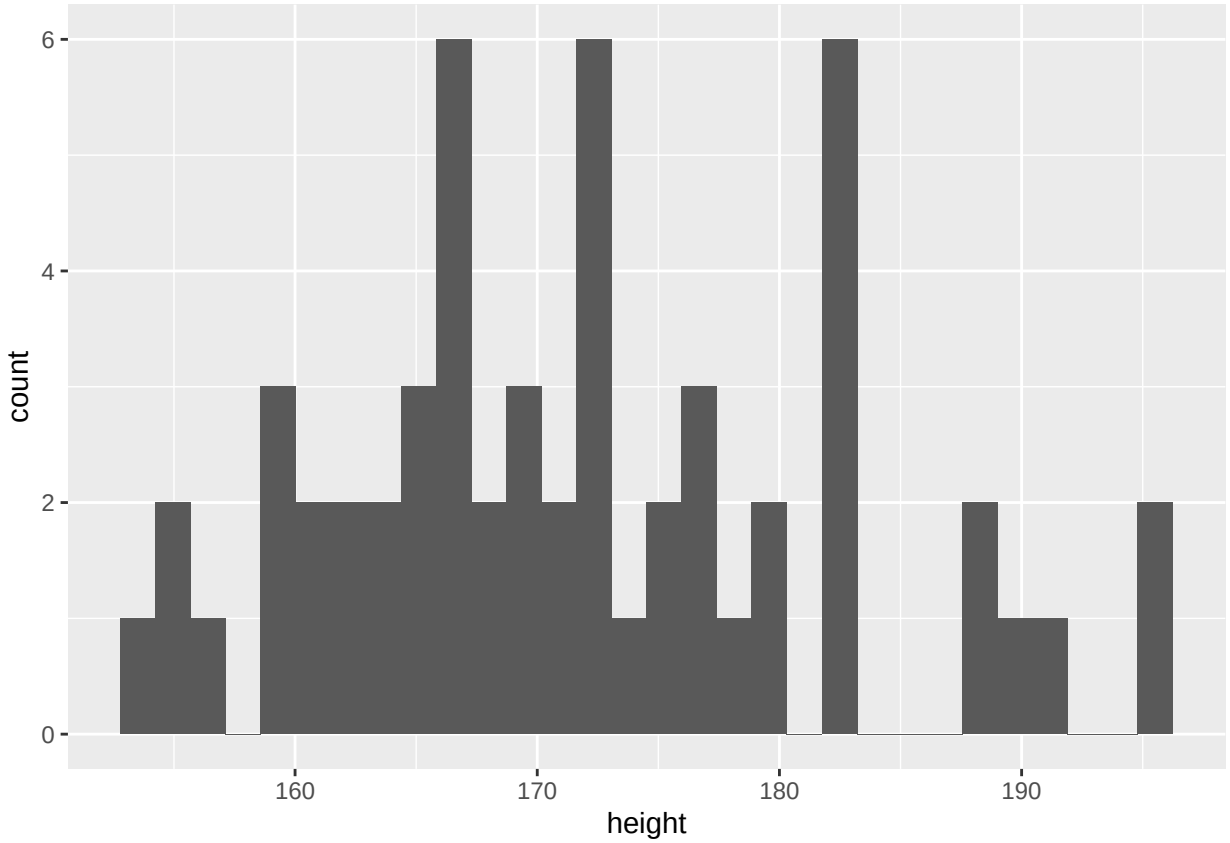
```
##                             vars  n   mean    sd median trimmed  mad min max
## gender*                        1 56   1.23  0.43      1    1.17 0.00   1   2
## height                         2 56 171.88 10.24    171  171.39 9.64 154 196
## shoe_size                      3 56  39.57  2.49     39   39.46 2.22  35  47
## hours_of_practice_per_week     4 56   7.29  3.26      7    7.35 2.97   0  14
## exam_score                     5 56  80.48  6.51     80   80.43 7.41  67  95
##                             range  skew kurtosis   se
## gender*                         1  1.24    -0.48 0.06
## height                         42  0.42    -0.46 1.37
## shoe_size                      12  0.62    -0.14 0.33
## hours_of_practice_per_week     14 -0.07    -0.70 0.44
## exam_score                     28  0.08    -0.69 0.87
```

```
mydata %>%
  summary()
```

```
##     gender              height        shoe_size     hours_of_practice_per_week
##  Length:56          Min.   :154.0   Min.   :35.00   Min.   : 0.000
##  Class :character   1st Qu.:165.0   1st Qu.:38.00   1st Qu.: 5.000
##  Mode  :character   Median :171.0   Median :39.00   Median : 7.000
##                     Mean   :171.9   Mean   :39.57   Mean   : 7.286
##                     3rd Qu.:178.5   3rd Qu.:41.00   3rd Qu.:10.000
##                     Max.   :196.0   Max.   :47.00   Max.   :14.000
##    exam_score
##  Min.   :67.00
##  1st Qu.:76.00
##  Median :80.00
##  Mean   :80.48
##  3rd Qu.:85.00
##  Max.   :95.00
```

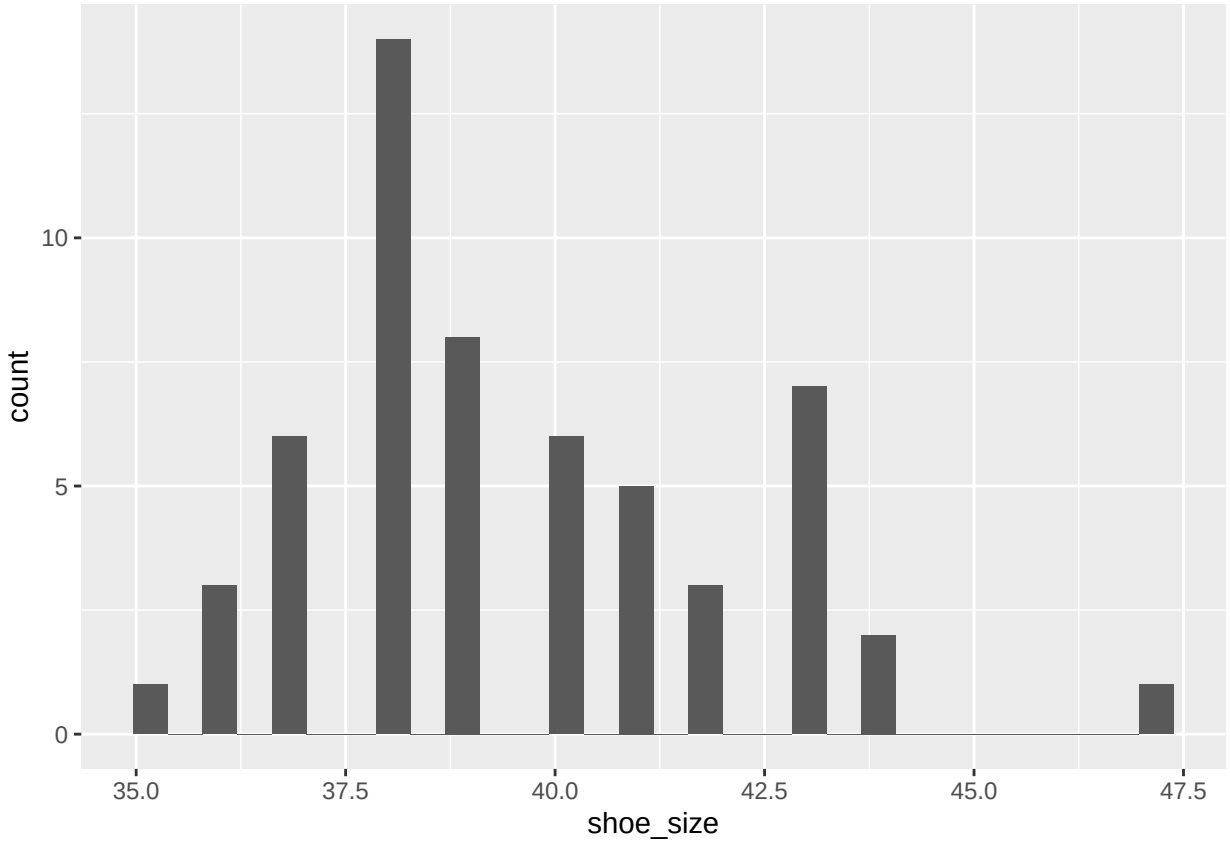
```
# histograms
mydata %>%
  ggplot() +
  aes(x = height) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

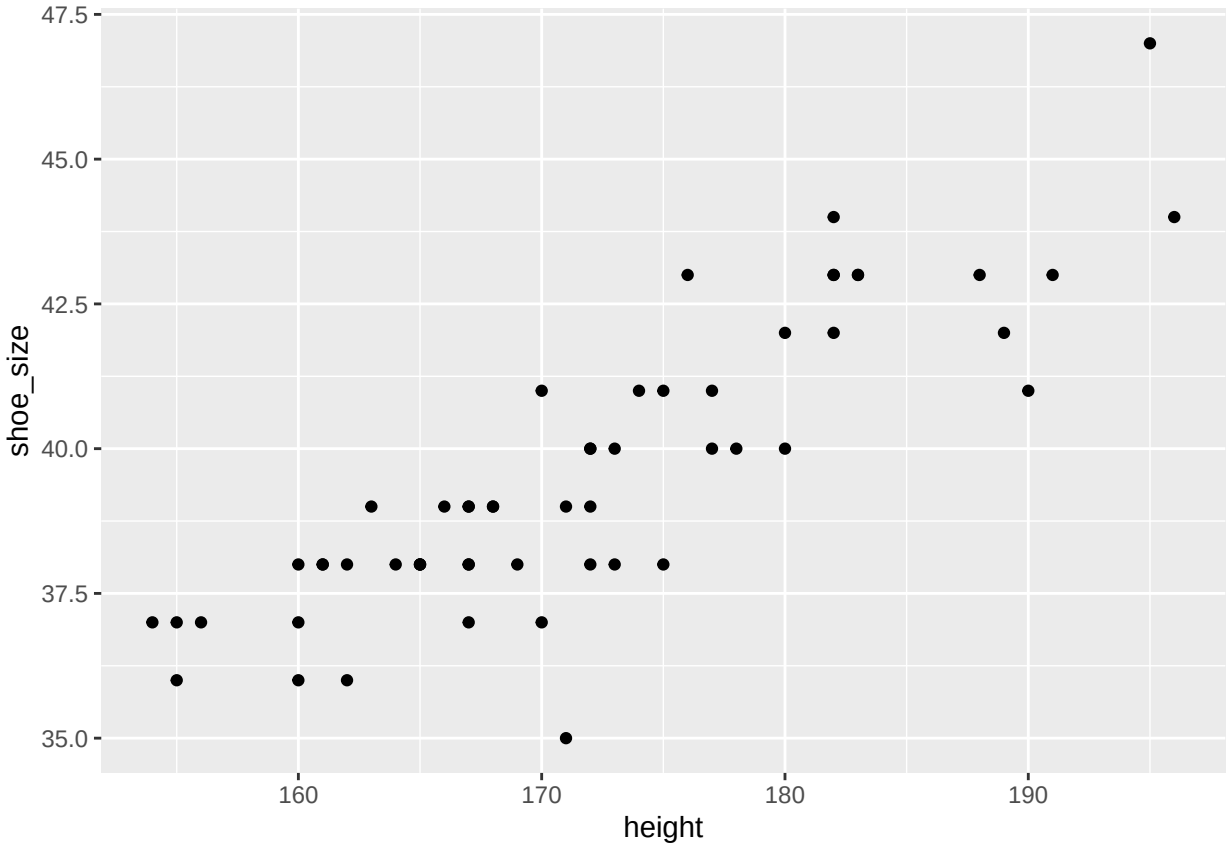



```
mydata %>%
  ggplot() +
  aes(x = shoe_size) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# scatterplot
mydata %>%
  ggplot() +
  aes(x = height, y = shoe_size) +
  geom_point()
```



Bejoslas linearis modellel

Egyszeru linearis modell felepitese

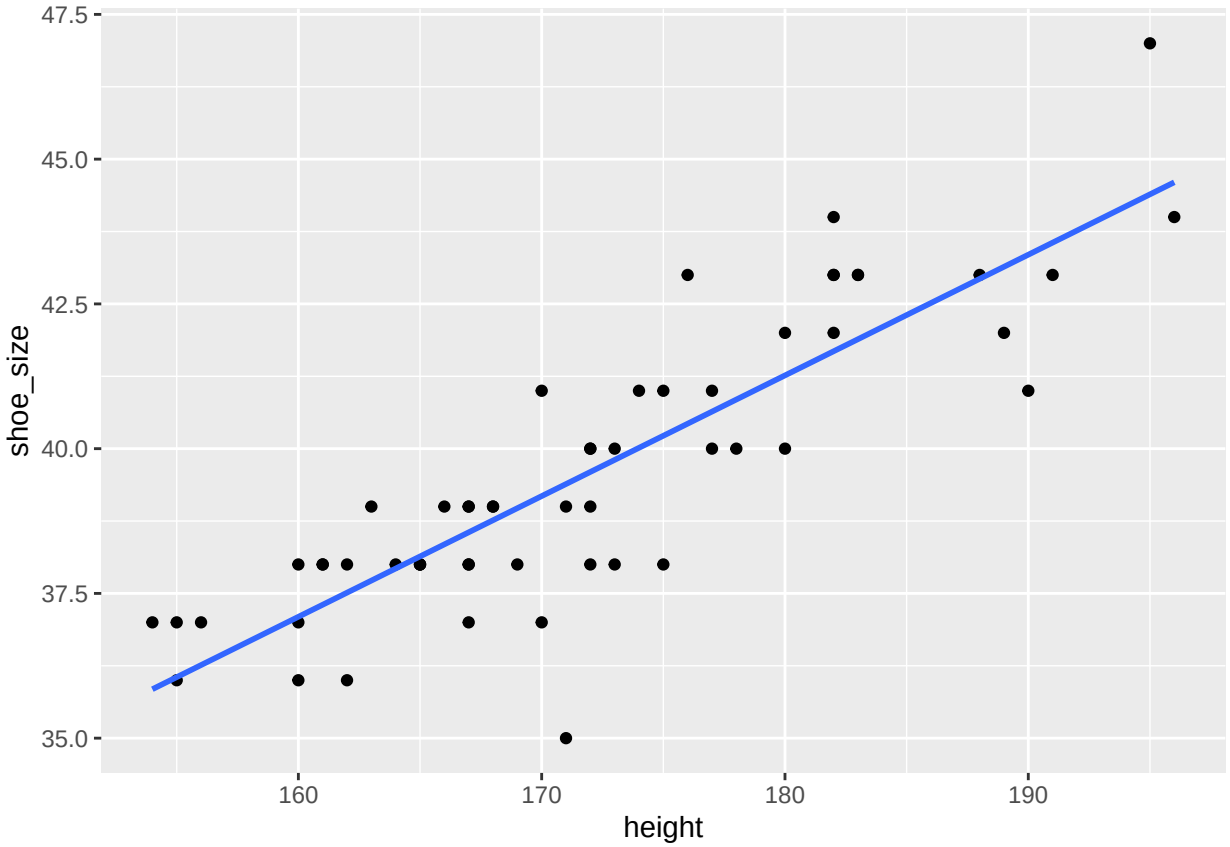
A regreszio **bejoslasra** vagy becslesre valo. Vagyis szeretnenk megtudni egy valtozo erteket (ezt altalaban bejosolt valtozonak vagy kimeneti valtozonak nevezzuk) mas bejoslo (prediktor) valtozok erteke alapjan.

Az alabbi peldaban szeretnenk megbecsulni (bejosolni/prediktalni) az egyes szemelyek EU cipomeretet (ez a bejosolt/kimeneti valtozó) a szemely magassaganak ismereteben (ez a bejoslo/prediktor valotozo). Ehhez eloszor az elozetes adataink hasznalataval felepitunk egy regresszios modellt.

Az linearis regresszioban a kimeneti valtozo es a prediktor kozotti kapcsolatot egy egyenessel modellezzuk. A modell az az egyenes lesz ami a legkozelebb esik a pont diagram pontjaihoz.

```
mydata %>%
  ggplot() +
  aes(x = height, y = shoe_size) +
  geom_point() +
  geom_smooth(method = "lm", se = F)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



A linearis regresszios modellt az **lm()** funkcioval epitjuk. Mindig ugy kell felepiteni, hogy a bejosolni kivant valtozoval kezdunk (shoe_size), majd a ~ jel utan irjuk a bejoslo valtozot (height). A kod vegen pedig azt specifikaljuk, melyik adattablaban talalhatoak ezek a valtozok a "data = ..." parameter megadasaval. A modellt elmentjuk egy objektumba (ezt most mod1-nek neveztuk el, de barminek elnevezhetnenk).

Az egyszeru linearis regresszional (simple linear regression) csak egy bejoslo valtozonk van.

A linearis regresszioban tobb bejoslo valtozot is hasznalhatunk, ilyenkor tobbszoros linearis regresszionak nevezzuk az eljarast (multiple linear regression). Errol majd kesobb lesz szo.

```
mod1 <- lm(shoe_size ~ height, data = mydata)
```

A regresszios modell megad egy matematikai egyenletet, amibe a prediktor valtozo erteket behelyettesitve megkaphatjuk a legjobb becslest a kimeneti valtozo ertekere. Ezt az egyenletet regresszios egyenletnek (regression equation) nevezzuk.

Regresszios egyenlet

A **regresszios egyenletet** igy formalizaljuk: Y = b0 + b1*X1, amelyben Y a kimeneti (bejosolt) valtozo becsult erteke, a b0 egy allando/konstans ertek ami nem fugg a bejososlo (prediktor) erteketol, amit legtobbszor intercept-nek neveznek, a b1 a regresszios egyutthato, az x1 pedig a bejososlo (prediktor) erteke az adott szemelynel.

Vagyis ugy kaphatunk egy becslest az Y bejosolt valotozo ertekere (height), ha a konstanshoz hozzaadjuk a regresszios egyutthato es a prediktor ertekenek szorzatat.

Ha kilistazzuk a modell objektumot (mod1), akkor megkaphatjuk a **regresszios egyenletet** erre a modellre amit most epitettunk.

```
mod1
```

```
##
## Call:
## lm(formula = shoe_size ~ height, data = mydata)
##
## Coefficients:
## (Intercept)       height
##      3.7426       0.2085
```

Ha a regresszios egyenlet elemei a kovetkezok:

- intercept (b0) = 3.74
- a height-hoz tartozo regresszios egyutthato (b1) = 0.21

Ezeket az adatokat a modell objektum kilistazasaval olvashatjuk le a "Coefficients:" reszbol.

Ez azt jelenti, hogy a cipomeretet bejoslo regresszios egyenlet a kovetkezo:

shoe_size = 3.74 + 0.21 * height

vagyis egy 170 cm magas ember eseten a modell altal becsult cipomeret:

3.74 + 0.21 * 170 = 39.44

Becsles a regresszios egyenlet alapjan

Ezt a szamitast nem kell kezzel vagy fejben megcsinalni, ehelyett hasznalhatod az R predict() funciojat a bejosolt ertek kiszamitasara.

A **predict()** funkcio hasznalatahoz meg kell adnunk egy adattablat (data.frame vagy tibble-t) ami a prediktor ertekeit tartalmazza, amit a kimeneti valtozo megbecslesere, bejoslasara szeretnenk hasznalni.

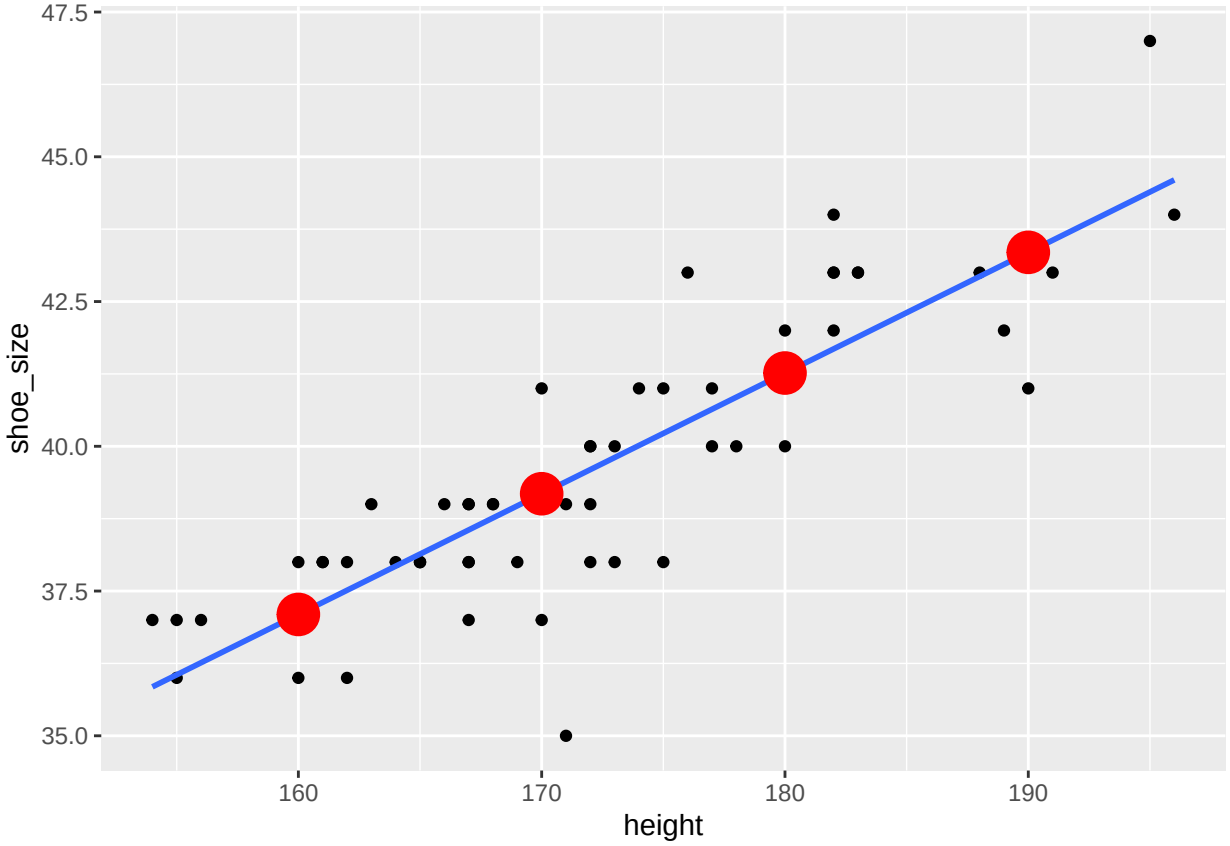
```
height_df = data.frame(height = c(160, 170, 180, 190))

predictions = predict(mod1, newdata = height_df)

height_df_with_predicted = cbind(height_df, predictions)
height_df_with_predicted
```

```
##   height predictions
## 1    160    37.09598
## 2    170    39.18057
## 3    180    41.26516
## 4    190    43.34975
```

Vagyuk eszre hogy a bejosolt ertekek **mind pontosan a regresszios egyenesre esnek**. Masszoval a regresszios egyenes minden lehetseges prediktorertekre megadja a kimeneti valtozo bejosolt erteket.

```
mydata %>%
  ggplot() +
  aes(x = height, y = shoe_size) +
  geom_point() +
  geom_smooth(method = "lm", se = F) +
  geom_point(data = height_df_with_predicted, aes(x = height, y = predictions), col = "red", size = 7)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



_____Gyakorlas_____

1. Szamold ki hogy a regresszios modell szerint a sajat magassagodhoz milyen cipomeret tartozik.
2. Epits egy egyszeru linearis regresszio modellt az lm() fugvennyel amiben az **exam_score** (ZH eredmeny) a kimeneti valtozo es az **hours_of_practice_per_week** (hetente atlagosan hany orat gyakololt) a prediktor. A modell eredmenyet mentsd el egy objektumba.
3. Ird le a regresszios fuggvenyt amivel bejosolhato a ZH eredmeny (exam_score).
4. Ertelmezd a regresszios fuggvenyt. Aki tobbet gyakorol annak magasabb vagy alacsonyabb a ZH eredmenye? (Egy abra segithet)
5. Ertelmezd a regresszios fuggvenyt. Aki egy oraval tobbet gyakorol hetente mint masok, annak mennyivel varhato hogy magasabb lesz az energiaszintje? (Opcionális: 5. Ennek a modellnek a segitsegevel becsuld meg a ZH eredmenyet olyan embereknek akik heti 2, 5, vagy 8 orat gyakorolnak.)

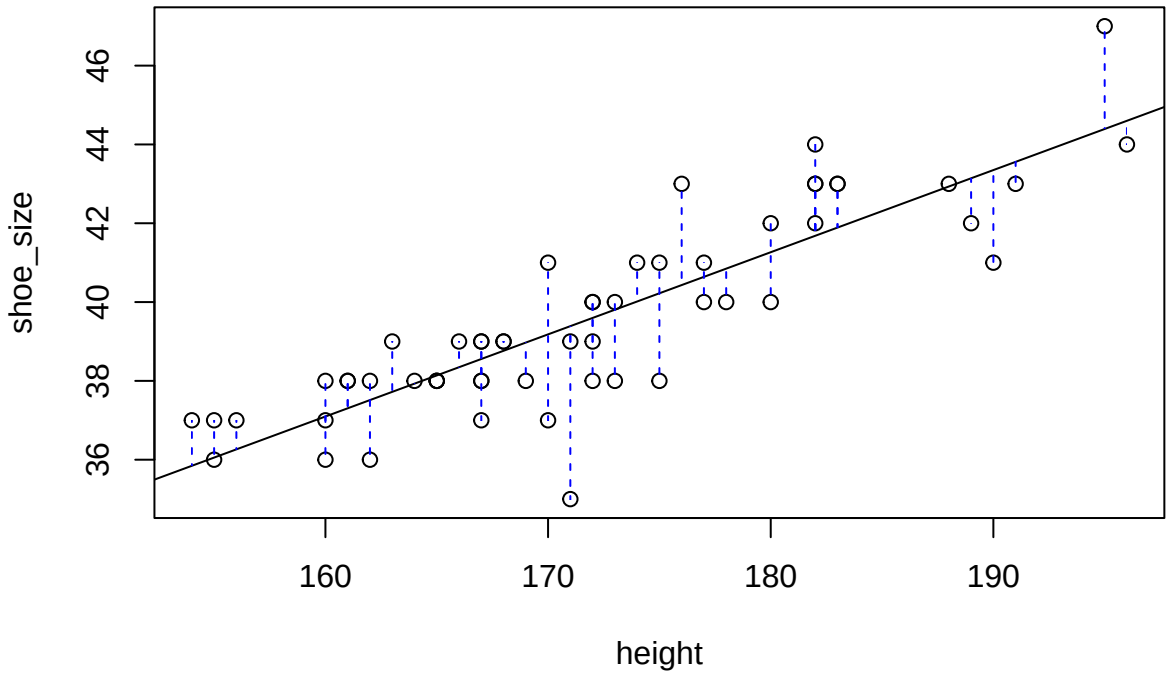
Milyen jo a modellem? (modellilleszkedes)

Hogyan merheto a becslesi/bejoslasi hatekonysag?

A modell becslesi hatekonysagat tobb fele keppen lehet merni. A legkezenfekvobb modszer, hogy meghatarozzuk, a modell becslese mennyire esett tavol a valos bejosolni kivant ertekektol. Vagyis megmerjuk a modell figyelembevetele utan fennmarado "hibat".

Ezt konnyen megtehetjuk egy olyan adatbazisban, ahol rendelkezesunkre all a bejosolni kivant valtozo valos erteke, ugy hogy kivonjuk egymasbol a valos erteket es a modell altal becsult erteket. Ez a rezidualis (fennmarado) hiba, masneven **residual error**.

```
error_plotter(mod1, col = "blue")
```



Ha vesszuk az osszes ilyen hiba ertek abszoluterteket, es osszeadjuk oket, megkapjuk a modell rezidualis abszolut hiba (residual absolute difference - RAD) erteket.

Ennel azonban joval gyakoribb hogy a rezidualis hiba negyzetosszeget hasznaljak (**residual sum of squares** - RSS) a statisztikaban. Vagyis az egyes rezidualis hiba ertekeket negyzetre emelik, majd osszeadjak oket.

Az alabbi peldaban a mod1 eredeti adattablajanak magassagertekeit hasznaljuk a cipomeret becsult ertekenek kiszamitasara (predict(mod1)), es ezt vonjuk ki az ugyan ezen adattablaban szereplo valos cimpomeret ertekekbol, igy kapjuk meg a rezidualis hibaertekeket. Majd egyenkent a negyzetuket vesszuk (RSS), es osszeadjuk oket a sum() fugvennyel.

```
RSS = sum((mydata$shoe_size - predict(mod1))^2)
RSS
```

```
## [1] 91.1467
```

Hasznos a modellunk?

Azt, hogy mennyire hasznos a modellunk (mennyit nyerunk azzal, hogy ezt a modellt hasznaljuk), meghatarozhatjuk ugy, hogy **osszehasonlitjuk** a rezidualis hibat abban az esetben **amikor a modellunket hasznaljuk** (vagyis amikor figyelembe vesszuk a prediktoraink erteket) egy olyan esettel, **amikor a**

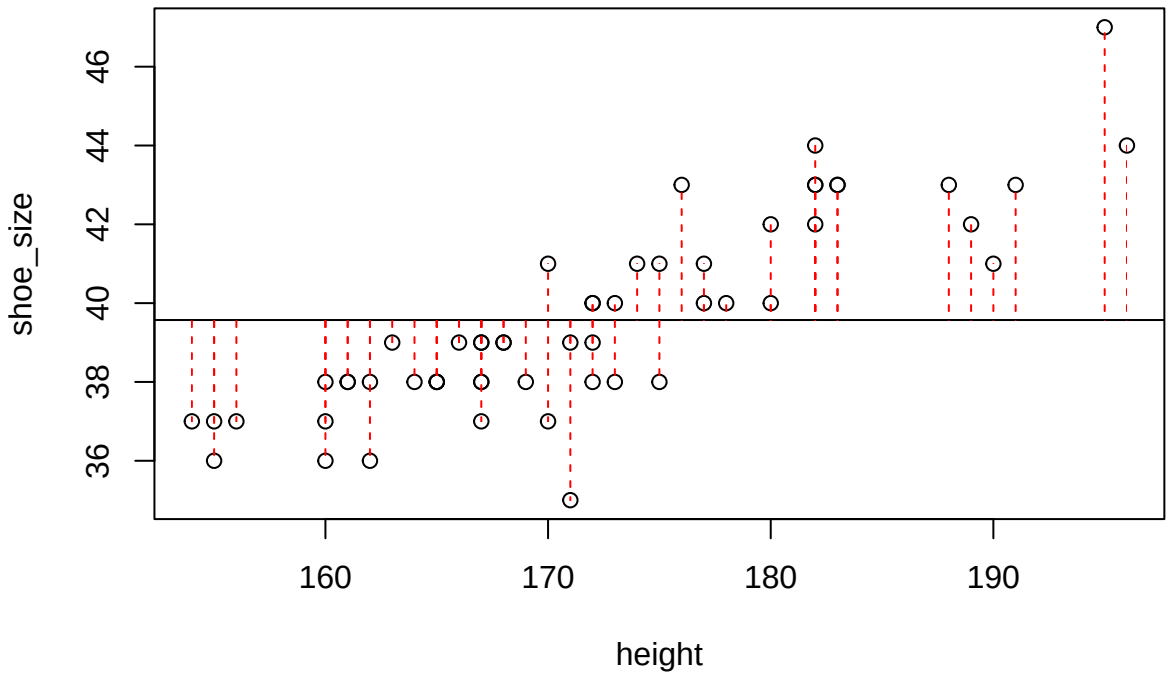
prediktorokat egyalatalan nem vesszuk figyelembe, csak a bejosolni kivant valtozo atlagat hasznaljuk a becslesre.

Az alabbi kodban epitunk egy olyan uj modellt, ahol nem veszunk figyelembe semmilyen masik valtozot, csak a cipomeret atlagat, es azt hasznaljuk fel a cipomeret becslesekent. (pl. ha tudjuk, hogy a populacioban az atlagos cipomeret 38, akkor mindenkinek ezt a cipomeretet becsuljuk majd, fuggetlenul attol, hogy milyen magas az illeto). Ezt a modellt **null modellnek** nevezzuk. Azt, hogy a bejosolt valtozo atlagat akarjuk becslesre hasznalni, ugy adhatjuk meg, hogy a ~ utan csak egy 1-est rakunk, nem irunk mas valtozonevet.

Ez persze nagy rezidualis hibahoz vezet (hiszen bar ez a populacioban az atlagos, megis a legtobb embernek nem pont 38-as a laba). A null modell altal produkalt rezidualis hibat ugyan ugy szamoljuk ki, mint a tobbi modellnel a residual sum of squared-et, viszont ennek van egy specialis neve is az irodalomban, ezt ugy hivjak, hogy **total sum of squares** (TSS), mert ez a lehetseges legegyszerubb meg ertelmes modell, ami altalaban nagy hibaval jar, igy ezt vesszuk a "teljes" hiba mennyisegnek, es ehhez viszonyitjuk a tobbi modell altal elert hibat.

Alabb kiszamoljuk a TSS-t. Lathato hogy a formula ugyan az mint az RSS eseten.

```
mod_mean <- lm(shoe_size ~ 1, data = mydata)

error_plotter(mod_mean, col = "red", x_var = "height") # visualize error
```




```
TSS = sum((mydata$shoe_size - predict(mod_mean))^2)
TSS
```

```
## [1] 341.7143
```

Mennyivel jobb a modellunk a null modellnel?

Azt, hogy mennyi informaciot nyertunk a kimeneti valtozo valtozekonysagarol (variance) a prediktorok figyelembevetelevel ahhoz kepest ha a null modellt vettuk volna figyelembe, az R^2 statisztika mutatja meg. Ennek a formulaja: 1-(RSS/TSS)

```
R2 = 1-(RSS/TSS)
R2
```

```
## [1] 0.7332663
```

Ha az R^2 ebben az esetben 0.73. Ez azt jelenti, hogy a prediktorok figyelembevetelevel (a mi esetunkben ez a magassag), a cipomeret valtozekonysaganak (atlagtol valo elteresenek) 73%-at tudjuk megmagyarazni.

R^2 = 1 azt jelenti, hogy a kimeneti valtozo variabilitasat teljesen meg tudjuk magyarazni a prediktorok ismereteben. R^2 = 0 azt jelenti, hogy a kimeneti valtozo variabilitasat egyaltalan nem magyarazzak meg a prediktorok

Hasznos a modellunk a populaciora nezve is?

Azt, hogy a modellunk hasznos-e a kimeneti valtozo bejoslasara populacio-szinten is, ugy tudjuk meghatarozni, hogy meghatarozzuk, a prediktorokat tartalmazo modell **szigifikansan jobb-e** mint a null modell a kimeneti valtozo becslesere?

Egy F tesztet hasznalhatunk a szignifikancia szint meghatarozosahoz. Ezt ugy kaphatjuk meg az R-ben hogy a ket modell altal produkalt rezidualis hibat az anova() funkcioval hasonlithatjuk ossze, melybe a null modell es a prediktorokat tartalamzo modell objektumot kell beletenni (akar tobbet mint ket modellt is lehet egyszerre). Itt az F-teszthez tartozo teszt statisztikat es p-erteket nezzuk, ha a szignifikanciara vagyunk kivancsiak.

```
anova(mod_mean, mod1)
```

```
## Analysis of Variance Table
##
## Model 1: shoe_size ~ 1
## Model 2: shoe_size ~ height
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     55 341.71
## 2     54  91.15  1    250.57 148.45 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model summary

A fentiekben lepesrol lepesre elvegeztuk a regresszio legfontosabb szamitasait sajat magunk altal irt koddal, hogy megertsetek, mi folyik egy linearis regresszio soran a "motorhazteto alatt". De ahogy sejthetitek, minderre az R-ben van egy gyorsabb es joval egyszerubb megoldas:

A modell **summary()** kikeresevel mindez a fenti informacio megkaphato, es meg tobb is.

Itt megtalalod az R^2 erteket, a modell null modellel valo osszehasonlitasanak F teszt statisztikajat es szignifikanciajat, es meg a regresszios egyenletet elemeit is.

```
summary(mod1)
```

```
##
## Call:
## lm(formula = shoe_size ~ height, data = mydata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.3890 -0.6103  0.2152  0.7477  2.6080
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.74256    2.94578    1.27    0.209
## height       0.20846    0.01711   12.18   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.299 on 54 degrees of freedom
## Multiple R-squared:  0.7333, Adjusted R-squared:  0.7283
## F-statistic: 148.4 on 1 and 54 DF,  p-value: < 2.2e-16
```

A regresszios egyutthatok (regression coefficients) konfidencia intervallumat a confint() paranccsal lehet kilistazni.

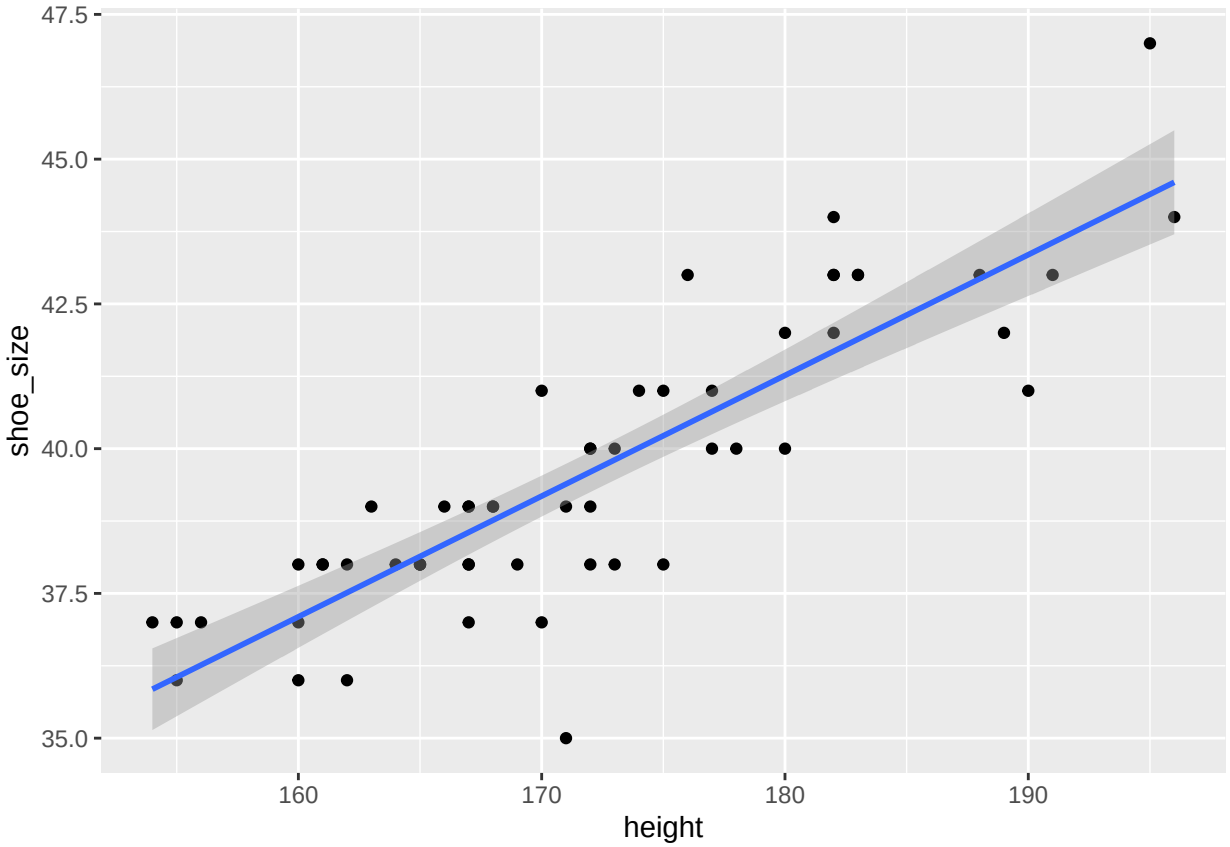
```
confint(mod1)
```

```
##                  2.5 %    97.5 %
## (Intercept) -2.1633697 9.6484844
## height       0.1741569 0.2427609
```

A regresszios becsles konfidencia intervallumat pedig a geom_smooth()-al lehet vizualizalni.

```
ggplot(mydata, aes(x = height, y = shoe_size))+
  geom_point()+
  geom_smooth(method='lm')
```

```
## `geom_smooth()` using formula 'y ~ x'
```



_____Gyakorlas_____

1. (Ezt nem kell megtenned ha ezt mar megtetted az elozo gyakorlasban, csak hasznald ugyan azt a model objektumot) Epits egy egyszeru linearis regresszio modellt az lm() fugvennyel amiben az **exam_score** (ZH eredmeny) a kimeneti valtozo es az **hours_of_practice_per_week** (hetente atlagosan hany orat gyakololt) a prediktor. A modell eredmenyet mentsd el egy objektumba.
2. Listazd ki a model summary-t a summary() fugvennyel
3. Olvasd le hogy a model ami tartalmazza az hours_of_practice_per_week prediktort szignifikansan jobb bejosloja-e az exam_score-nak mint a null modell.
4. Hatarozd meg a regresszios egyutthatok konfidencia intervallumat a confint() fuggvennyel

---

Tobbszoros linearis regresszio

Adatmenedzsment es adat bemutatasa 2

Az adatfajl betoltese: Lakasarak adattabla

Ebben a gyakorlatban lakasok es hazak arait fogjuk megbecsulni.

Egy **Kaggle**-rol szarmazo adatbazist hasznalunk, melyben olyan adatok szerepelnek, melyeket valoszinusithetoen alkalmasak **lakasok eladasi aranak bejoslasara**. Az adatbazisban az USA Kings County-bol szarmaznak az adatok (Seattle es kornyeke).

Az adatbazisnak csak egy kis reszet hasznaljuk (N = 200).

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

Adatellenoryes

Mindig ellenorizd az adatok strukturajat es integritasat.

Eloszor atvaltjuk az USA dollar-t millio forint mertekegysegre, es 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 lakasok arat a **sqm_living** (a lakas lakoreszenek alapterulete negyzetmeterben), es a **grade** (a lakas altalanos minositese a King County grading system szerint, ami a lakas minoseget, poziciojat, a haz minoseget stb. is tartalmazza) prediktorok felhasznalasaval josoljuk majd be. Kesobb a **has_basement** (tartozik-e a lakashoz pince) valtozot is hasznaljuk majd. Szoval fokuszaljunk ezekre a valtozokra az adatellenorzes soran.

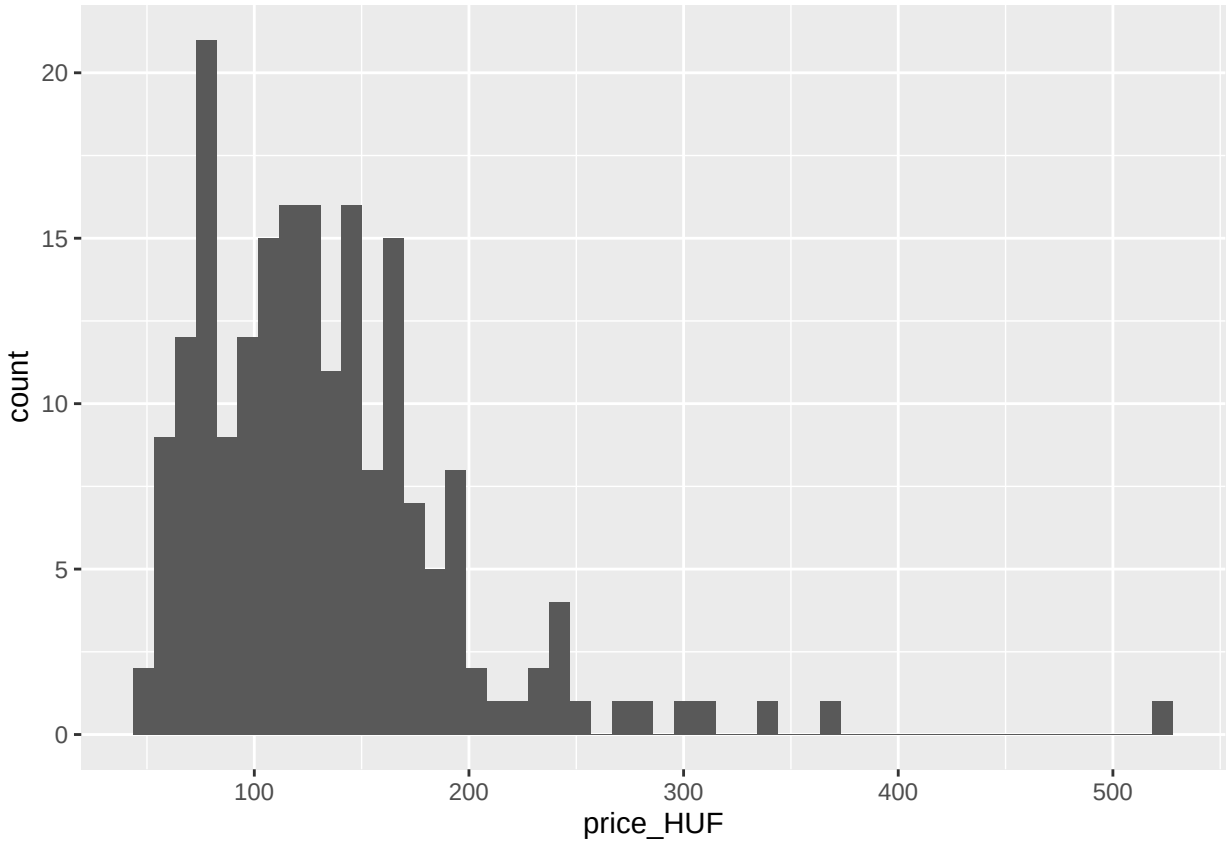
```
# leiro statiszikaka
describe(data_house)
```

```
## 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

##                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          1.68 4.700000e-01          2.00          1.72
## 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
```

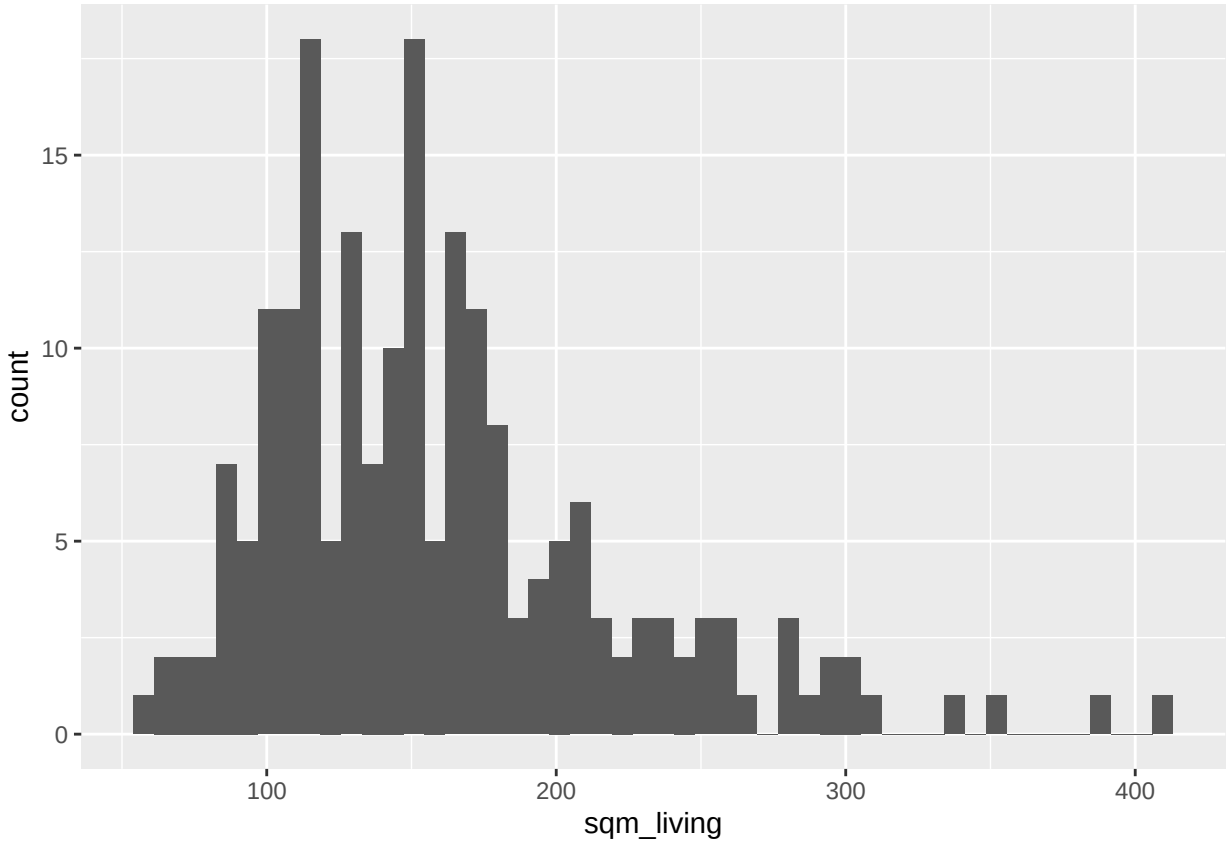
```
## sqm_living15    28 200        166.61 4.764000e+01        159.33        161.89
## sqm_lot15       29 200       1042.88 1.826810e+03        670.95        702.34
##                       mad         min           max        range  skew
## id            2.981805e+09 16000200.00 9818700320.00 9.802700e+09  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     -122.46       -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* 0.000000e+00        1.00          2.00 1.000000e+00 -0.74
## 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*    -1.46         0.03
## 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
```

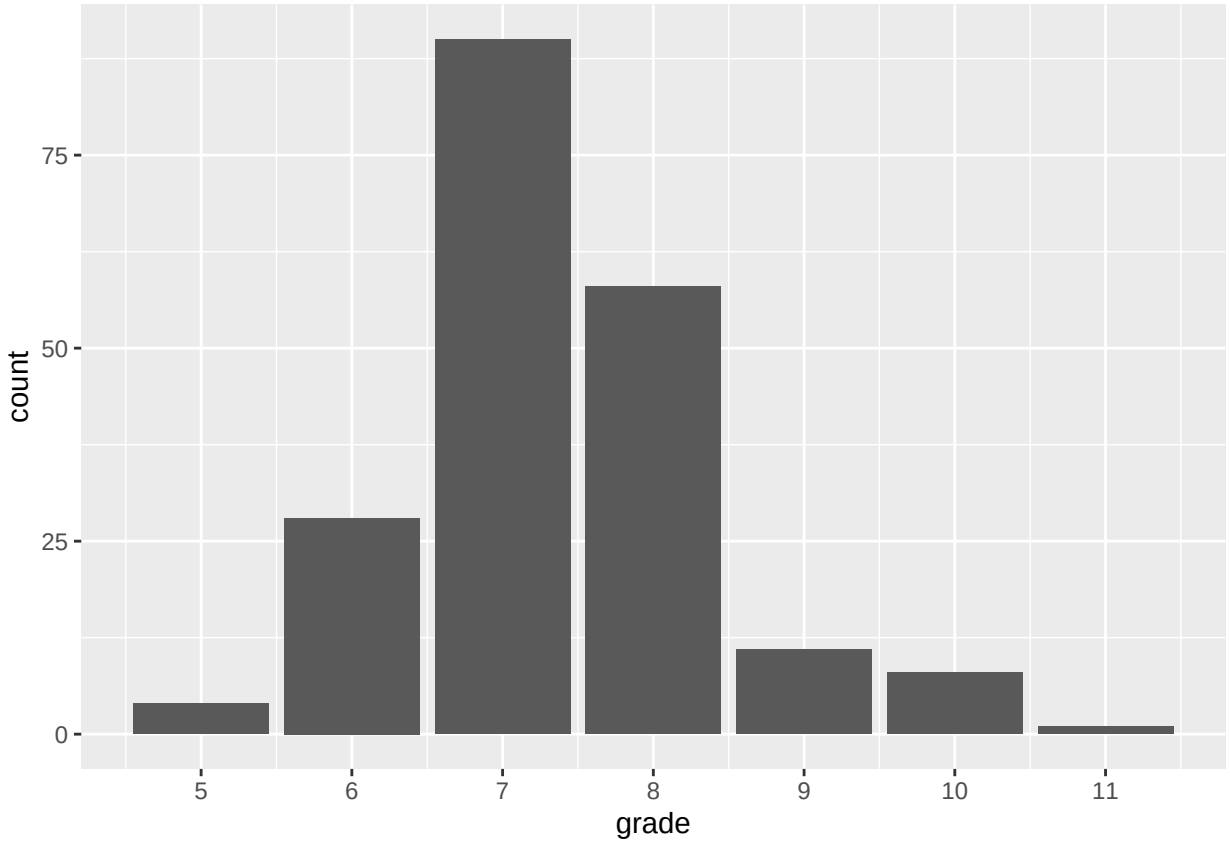
```
# hisztogramok
data_house %>%
  ggplot() +
  aes(x = price_HUF) +
  geom_histogram( bins = 50)
```



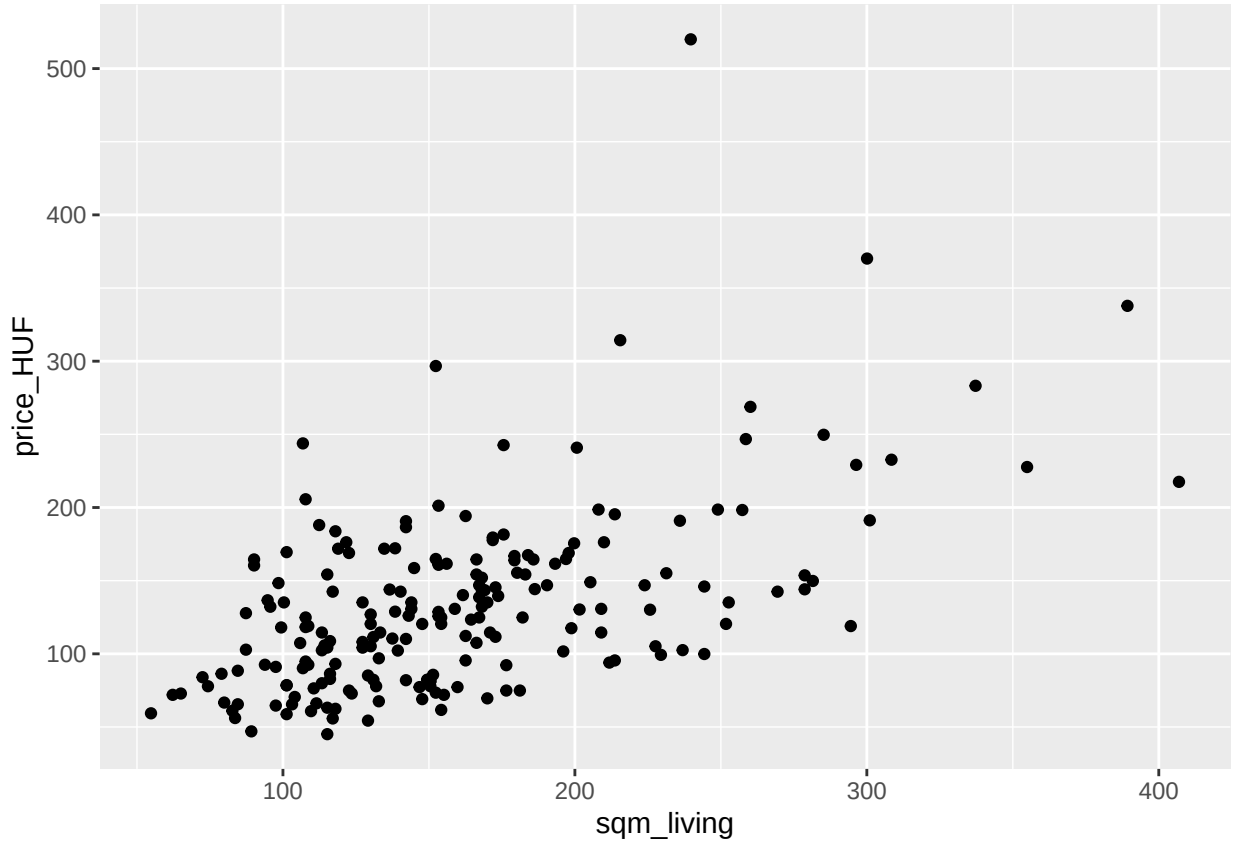

```
data_house %>%
  ggplot() +
  aes(x = sqm_living) +
  geom_histogram( bins = 50)
```



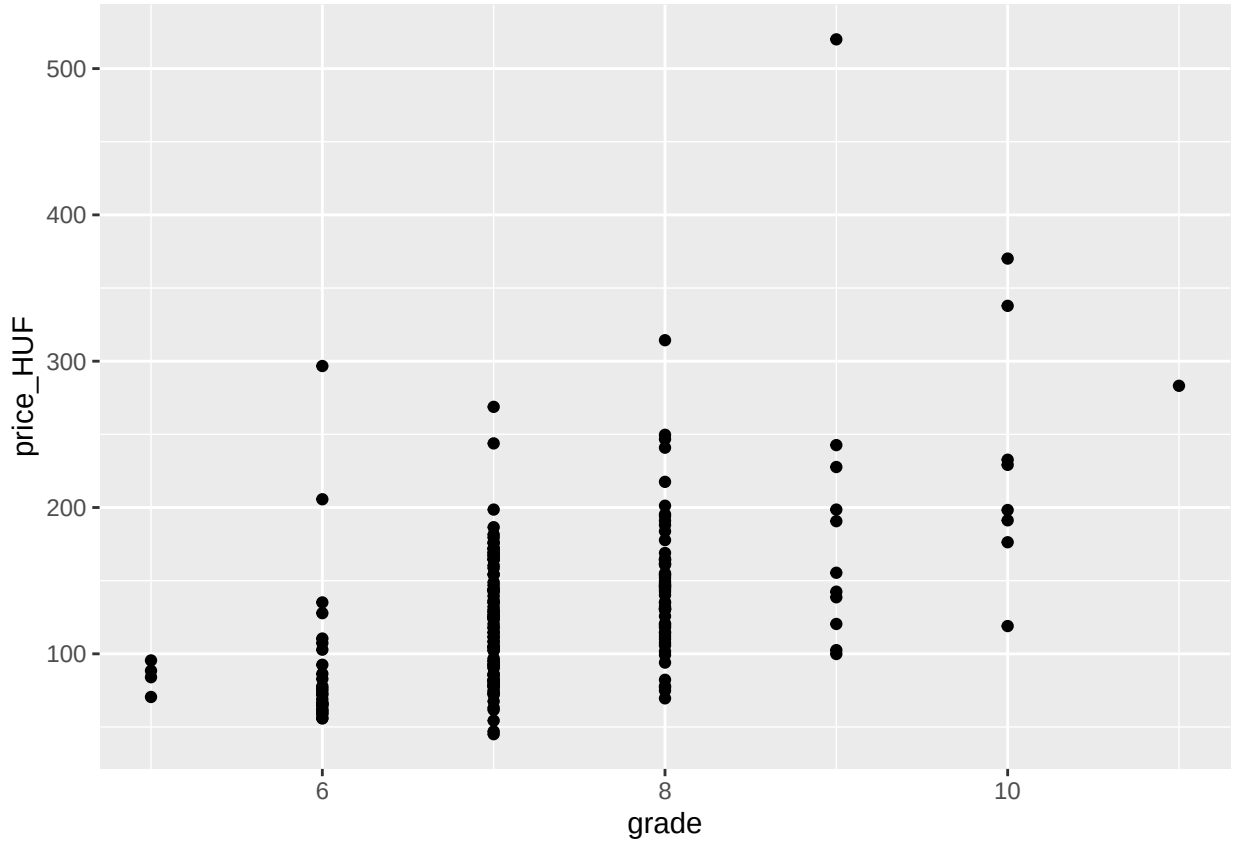

```
data_house %>%
  ggplot() +
  aes(x = grade) +
  geom_bar() +
  scale_x_continuous(breaks = 4:12)
```




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



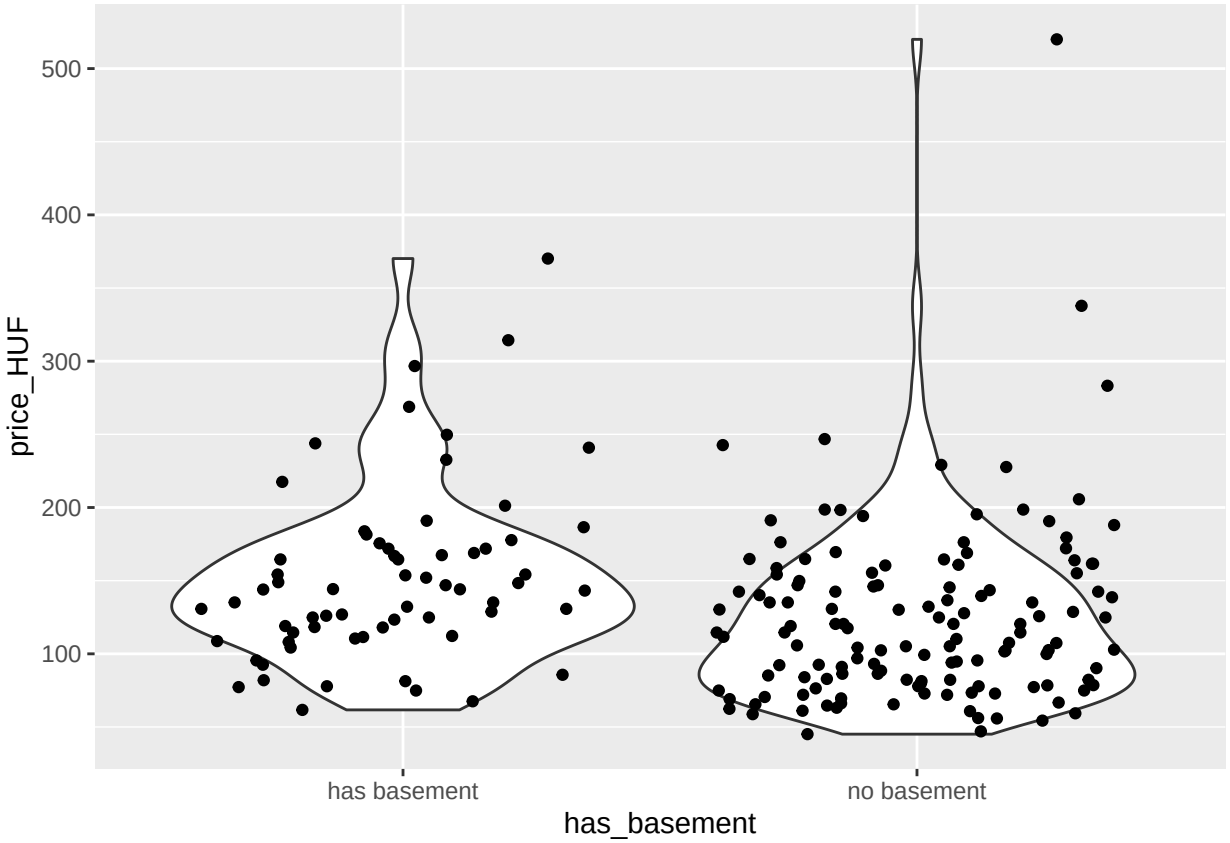

```
data_house %>%
  ggplot() +
  aes(x = grade, y = price_HUF) +
  geom_point()
```




```
# 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()
```



Tobbszoros regresszio

A regresszios modell felepitese (fitting a regression model)

A tobbszoros regresszios modellt ugyan ugy epeitjuk mint az egyszeru regresszios modellt, csak csak tobb prediktort is betehetunk a modellbe. Ezeket a prediktorvaltozokat + jellen valasztjuk el egymastol a regresszios formulaban.

Alabb **price_HUF** a bejosolt valtozo, es a **sqm_living** es a **grade** a prediktorok.

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

A regresszios egyenletet a modell objektumon keresztul erhetjuk el:

```
mod_house1
```

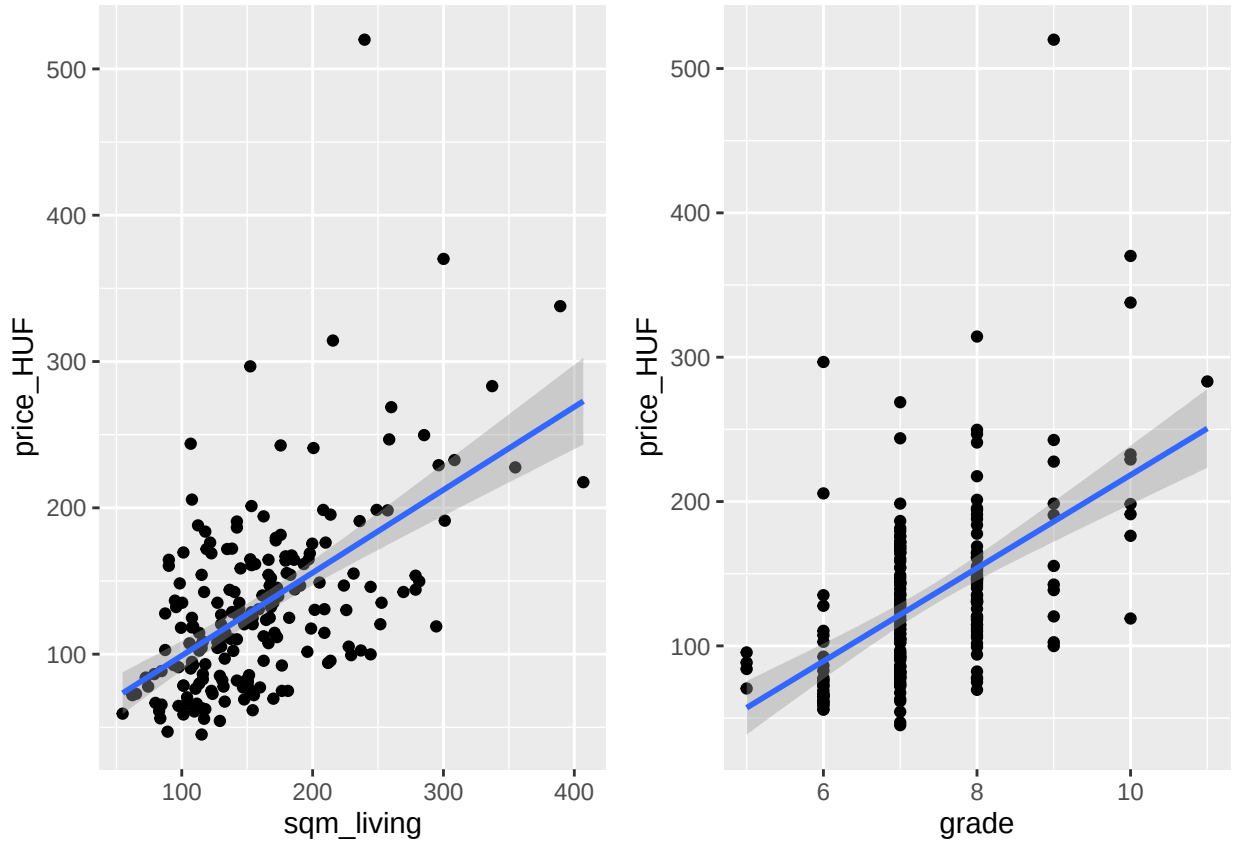
```
##
## Call:
## lm(formula = price_HUF ~ sqm_living + grade, data = data_house)
##
## Coefficients:
## (Intercept)   sqm_living        grade
##    -51.2305       0.3768      16.8485
```

A tobbszoros regresszios modellek vizualizacioja nem olyan egyertelmu mint az egyszeru regresszios modelleke.

Az egyik megoldas hogy a paronkenti osszefuggeseket vizualizaljuk egyenkent, de ez nem ragadja meg a modell tobbvaltozos jelleget.

```
# 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)
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



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 tobb prediktor eseten mar egy tobbdi-

menzios terben kepzelheto csak el a regresszios felulet, ezert a vizualizaciora altalaban megis az paronkenti scatterplot-ot szoktuk hasznalni.

```
# plot the regression plane (3D scatterplot with regression plane)

scatter3d(price_HUF ~ sqm_living + grade, data = data_house)
```

Becsles (prediction)

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

Fontos, hogy a prediktorok ertekeit egy data.frame vagy tibble formatumban kell megadnunk, es a prediktorvaltozok valtozoneveinek meg kell egyeznie a regresszios modellben hasznalt valtozonevekkel.

```
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
```

Hogyan kozoljuk az eredmenyeinket egy kutatasi jelentesben

Egy kutatsi jelentesben (pl. cikk, muhelymunka, ZH) a kovetkezo informaciokat kell leirni a regresszios modellrol:

Eloszor is le kell irni a regresszios **modell tulajdonsagait** (altalaban a "Modszerek" reszben):

"Egy linearis regresszios modellt illesztettem, melyben a lakas arat (millio HUF-ban) a lakas lakoreszenek teruletevel (m^2-ben) es a lakas King County lakas-minosites ertekevel becsultem meg."

"I built a linar 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."

Ezutan a **teljes modell bejoslasi hatekonysagat** kell jellemezni. Ezt a modellhez tartozo adjusted R^2 ertek (modositott R^2), es a modell-t a null-modellel osszehasonlito anova F-tesztjenek statiszikainak megadasaval szoktuk tenni (F-ertek, df, p-ertek). Mindezen informaciot a summary() funkcioval tudjuk lekerdezni. A modell illeszkedeset az AIC (Akaike information criterion) ertekkel is szoktuk jellemezni, amit az AIC() funcio ad meg.

Az APA publikacios kezikonyv alapjan minden szamot ket tizedesjegy pontossaggal kell megadni, kiveve a p erteket, amit harom tizedesjegy pontossaggal.

```
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" reszben igy irnank a fenti pelda eredmenyeirol:

"A tobbszoros regresszios modell mely tartalmazta a lakoterulet es a lakas minosites prediktorokat hatekonyabban tudta bejosolni a lakas arat mint a null modell. A modell a lakasar varianciajanak 0.35%-at magyarazta (F (2, 197) = 54.94, p < 0.001, Adj. R^2 = 0.35, AIC = 2137.06"

Ezen felul meg kell adnunk a **regresszios egyenletre es az egyes prediktorok becsleshez valo hozzajarulasara vontkozo adatokat**. Ezt altalaban egy osszefoglalo tablazatban szoktuk megadni, melyben a kovetkezo adatok szerepelnek prediktoronkent:

- regresszios egyutthato (regression coefficients, estimates) - summary()
- az egyutthatokhoz tartozo konfidencia intervallum (coefficient confidence intervals) - confint()
- standard beta ertekek (standardized beta values) - lm.beta() az lm.beta pakcage-ben
- a t-teszthez tartozo p-ertek (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 tablazat valahogy igy nez majd ki (ennek az elkesztesehez a fenti coef_table() sajat funkciot hasznaltam. Nem fontos ezt hasznalni, manualisan is ki lehet irogatni az eredmenyeket a kulonbozo tablazatokbol.):

```
##                   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
```

regresszios egyutthato ertelmezese

A regresszios egyutthatot ugy lehet ertelmezni, hogy a prediktor ertekenek egy ponttal valo novekedese eseten a kimeneti valtozo erteke ennyivel valtozik. Pl. ha a sqm_living-hez tartozo regresszios egyutthato 0.38, az azt jelenti hogy minden egyes ujabb negyzetmeter teruletnovekedes 0.38 millio forint arvaltozassal jar.

az intercept-hez tartozo regresszios egyutthato ertelmezese

Az intercept egyutthatoja azt mutatja meg, hogy mi lenne a bejosolt (fuggo) valtozo becsult erteke, ha minden prediktor 0 erteket vesz fel. Ez nem mindig egy realis becsles, hiszen attol fuggoen hogy milyen prediktorokat hasznalunk, lehet hogy egy adott prediktoron a 0 ertek nem ertelmes. Ettol fuggetlenul az intercept matematikai ertelmezese mindig ugyan ez marad. Az intercept egyfajta allando ertek, ami fuggetlen a prediktorok erteketol.

standard beta ertelmezese

A regresszios egyutthato elonye, hogy a kimeneti valtozo mertekegysegeben van, es nagyon egyszeru ertelemzni. Ezert ez egy "nyers" hatasmeret mutato. Viszont a hatranya hogy az erteke a hozza tartozo prediktor valtozo skalajan mozog. Ez azt jelenti, hogy az egyes egyutthato ertekek nem konnyen osszehasonlithatoak, mert a prediktorok mas skalan mozognak. Pl. az sqm_living egyutthatoja alacsonyabb mint az grade egyutthatoja, de ez onmagaban nem mond arrol semmit, hogy melyik prediktornak van nagyobb szerepe a kimeneti valtozo bejoslasaban, mert a sqm_living skalaja sokkal kiterjedtebb (50-400 m^2) mint a grade skalaja (5-11).

Ahhoz hogy ossze tudjuk hasonlitani az egyes prediktorok becsleshez hozzaadott erteket, a ket egyutthatot ugyan arra a skalara kell helyeznunk, amit standardizalassal erhetunk el (ennek egyik modja hogy a prediktor valtozokat Z-transzformaljuk, es ezeket a Z-transformalt ertekeket tesszuk a modellbe mint prediktorokat). A standard Beta egy ilyen standardizalt mutato. Ez mar direkt modon osszehasonlithato a prediktorok kozott. Ebbol mar latszik hogy a sqm_living hozzaadott erteke a price_HUF bejoslasahoz nagyobb mint a grade hozzaadott erteke.

Amikor tobb prediktor van, ez nem feltetlenul jelenti azt, hogy ha egyenkent megneznenk a prediktorok korrelaciojat a kimeneti valtozoval, akkor ugyan ilyen osszefuggest kapnank. Ez az egyutthato es a std.Beta ertek a prediktor egesz modellben betoltott szerepet jeloli, a tobbi prediktor bejoslo erejenek leszamitasaval. Vagyis elkepzelheto, hogy egy prediktor onmagaban jobban korrelal a kimeneti valtozoval mint barmelyik masik prediktor, viszont a modellben kisebb szerepet jatszik, mert a tobbi prediktor ugyan azt a reszet magyarazza a kimeneti valtozo varianciajanak, mint ez a prediktor.

_____Gyakorlas_____

1. Epits egy tobbszoros linearis regresszio modellt az lm() fugvennyel amiben az **price** a kimeneti valtozot becsuljuk meg. Hasznalhatod a **data_house** adatbazisban szereplo barmelyik valtozot felhasznalhatod a modellben, ami szerinted realisan hozzajarulhat a lakas aranak meghatarozasahoz.
2. Hatarozd meg, hogy szignifikansan jobb-e a modelled mint a null modell (a teljese modell F-teszthez tartozo p-ertek alapjan)?

3. Mekkora a teljes modell altal bejosolt varianciaarany (adj.R^2)?
4. Melyik az a prediktor, mely a legnagyobb hozzadaott ertekkel bir a becslesben?
