

# Logisztikus regresszió

Kekecs Zoltan

November 10, 2021

## Logisztikus regresszió

### Absztrakt

Ebben a gyakorlatban megtanuljuk, hogyan készítsünk előrejelző modelleket binomiális kimeneti változókra. Elsősorban a logisztikus regresszió használatát fogjuk tárgyalni.

### Loading packages

```
library(pscl) # for pR2
library(lmtest) # for lrtest
# library(dominanceanalysis) # for dominanceAnalysis()
library(tidyverse) # for dplyr and ggplot2
library(domir)
```

## Adatkezelés és leíró statisztika

### Adatbázis

A Heart Disease adatbázist fogjuk használni, amely egy jól ismert adatbázis, amelyet kategorizációs problémák bemutatására használnak. Az adatkészlet különböző adatokat tartalmaz olyan betegekről, akiket szívbetegség gyanújával vizsgáltak.

Az alábbi kódban a “dec =”, ” azt jelzi, hogy a tizedesjel ebben az online adatkészletben “,” (az R-ben alapértelmezett “.” helyett).

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

Az adatkészlet a következő változókat tartalmazza:

- age - életkor években kifejezve
- sex - (1 = férfi; 0 = nő)
- cp - mellkasi fájdalom típusa (0 = tünetmentes, 1 = tipikus angina, 2 = atipikus angina, 3 = nem anginás fájdalom)
- trestbps - nyugalmi szisztolés vérnyomás (mm Hg-ban a kórházba való felvételnél)
- chol - szérumszénvesztés mg/dl-ben
- fbs - (éhségszénvesztés > 120 mg/dl) (1 = igaz; 0 = hamis)

- restecg - nyugalmi elektrokardiográfiás eredmények: (0 = normális; 1 = ST-T-hullám eltérés; 2 = valószínű vagy biztos bal kamrai hipertrófia)
- thalach - a terheléses vizsgálat során elért maximális pulzusszám.
- exang - terhelés okozta angina pectoris (1 = igen; 0 = nem)
- oldpeak - a terhelés által kiváltott ST-depresszió a nyugalomhoz képest
- meredekség - a terhelés ST-csúcsának meredeksége
- ca - a flouroszópiával színezett fő erek száma (0-3)
- thal - 3 = normális; 6 = rögzített defektus; 7 = visszafordítható defektus
- disease\_status - van-e szívbetegség vagy nincs (heart\_disease vs. no\_heart\_disease)

Az adatkészletről további információkat itt találhatsz:

Translated with [www.DeepL.com/Translator](http://www.DeepL.com/Translator) (free version)

<https://www.kaggle.com/ronitf/heart-disease-uci>; <https://archive.ics.uci.edu/ml/datasets/Heart+Disease>

## Adatmenedzsment

Azzal kezdjük, hogy rendbe tesszük az adatállományunkat, a kategorikus változókat faktorokként definiáljuk, a faktorszinteket átkódoljuk, hogy informatívabbak legyenek, és olyan új változóneveket adunk, amelyek informatívabbak.

```
heart_data = heart_data %>%
  mutate(sex = factor(recode(sex,
                             "1" = "male",
                             "0" = "female")),
         cp = factor(recode(cp,
                             "0" = "asymptomatic",
                             "1" = "typical_angina",
                             "2" = "atypical_angina",
                             "3" = "non_anginal_pain")),
         fbs = factor(recode(fbs,
                             "1" = "true",
                             "0" = "false")),
         disease_status = factor(disease_status)
  ) %>%
  rename(chest_pain = cp,
         sys_bloodpressure = trestbps,
         blood_sugar_over120 = fbs,
         max_HR = thalach,
         cholesterol = chol)

names(heart_data)[1] = "age"
```

## Adatellenőrzés

Mindig ellenőrizd az adatokat kódolási hibák vagy értelmetlen adatok szempontjából, és vizsgálja meg az adatokat, hogy megérted, milyen típusú adatokkal van dolgod.

```
heart_data %>%
  summary()
```

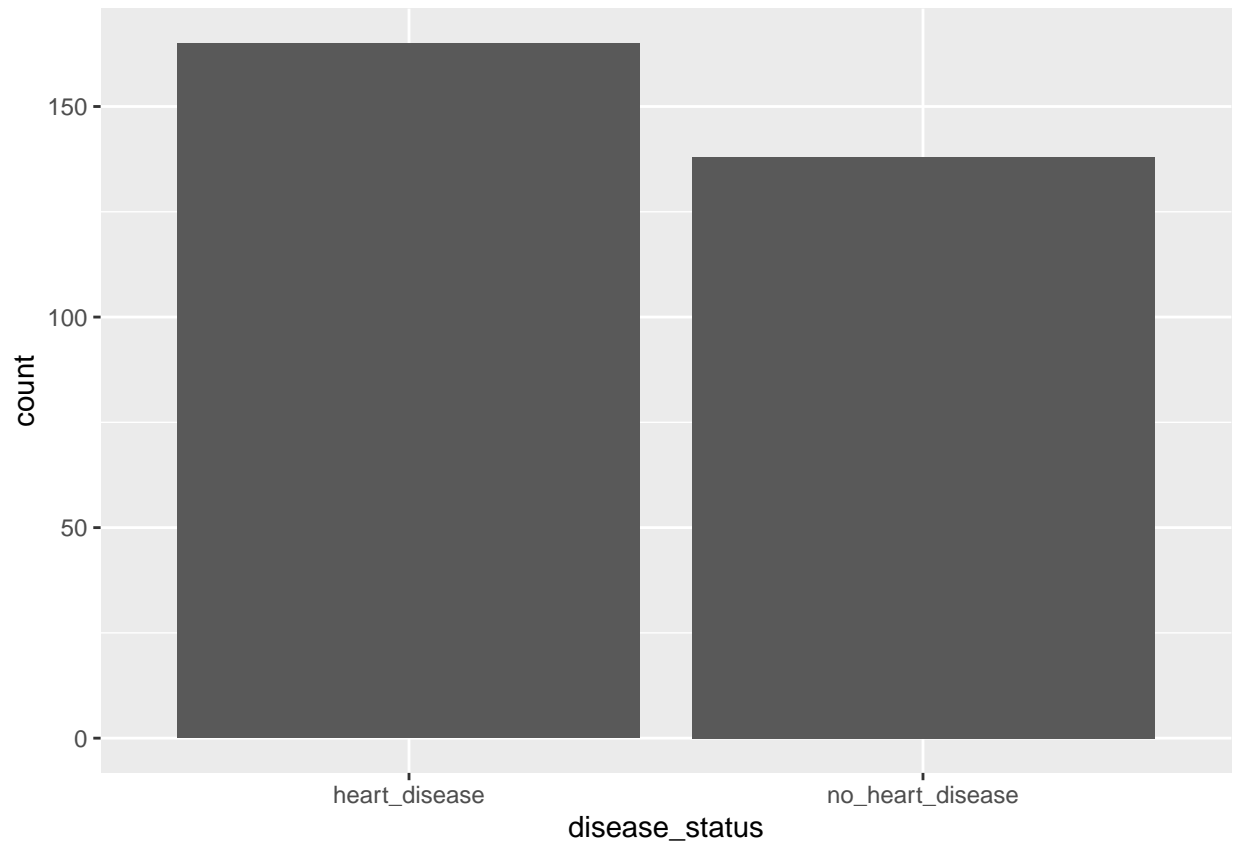
```
##      age          sex          chest_pain  sys_bloodpressure
## Min.   :29.00    female: 96    asymptomatic   :143    Min.    : 94.0
## 1st Qu.:47.50    male  :207    atypical angina : 87    1st Qu.:120.0
## Median :55.00                    non_anginal_pain: 23    Median  :130.0
## Mean   :54.37                    typical_angina  : 50    Mean    :131.6
## 3rd Qu.:61.00                                     3rd Qu.:140.0
## Max.    :77.00                                     Max.    :200.0
## cholesterol  blood_sugar_over120  restecg          max_HR
## Min.    :126.0  false:258          Min.    :0.0000    Min.    : 71.0
## 1st Qu.:211.0  true : 45          1st Qu.:0.0000    1st Qu.:133.5
## Median :240.0                    Median :1.0000    Median  :153.0
## Mean    :246.3                    Mean   :0.5281    Mean    :149.6
## 3rd Qu.:274.5                    3rd Qu.:1.0000    3rd Qu.:166.0
## Max.    :564.0                    Max.    :2.0000    Max.    :202.0
##      exang      oldpeak      slope      ca
## Min.    :0.0000    Min.    :0.00    Min.    :0.000    Min.    :0.0000
## 1st Qu.:0.0000    1st Qu.:0.00    1st Qu.:1.000    1st Qu.:0.0000
## Median :0.0000    Median :0.80    Median :1.000    Median :0.0000
## Mean    :0.3267    Mean    :1.04    Mean    :1.399    Mean    :0.7294
## 3rd Qu.:1.0000    3rd Qu.:1.60    3rd Qu.:2.000    3rd Qu.:1.0000
## Max.    :1.0000    Max.    :6.20    Max.    :2.000    Max.    :4.0000
##      thal      disease_status
## Min.    :0.000    heart_disease   :165
## 1st Qu.:2.000    no_heart_disease:138
## Median :2.000
## Mean    :2.314
## 3rd Qu.:3.000
## Max.    :3.000
```

## Kutatási kérdés és feltáró adatelemzés

A fő célunk az lesz, hogy olyan modellt hozzunk létre, amely hatékonyan képes megjósolni a `disease_status`-t, vagyis azt, hogy az adott személynek van-e szívbetegsége vagy nincs. Ehhez három fő prediktort fogunk használni: a terheléses vizsgálat során elért maximális pulzusszámot (`max_HR`), a nyugalmi szisztolés vérnyomást (`sys_bloodpressure`) és a mellkasi fájdalom jelenlétét (`chest_pain`).

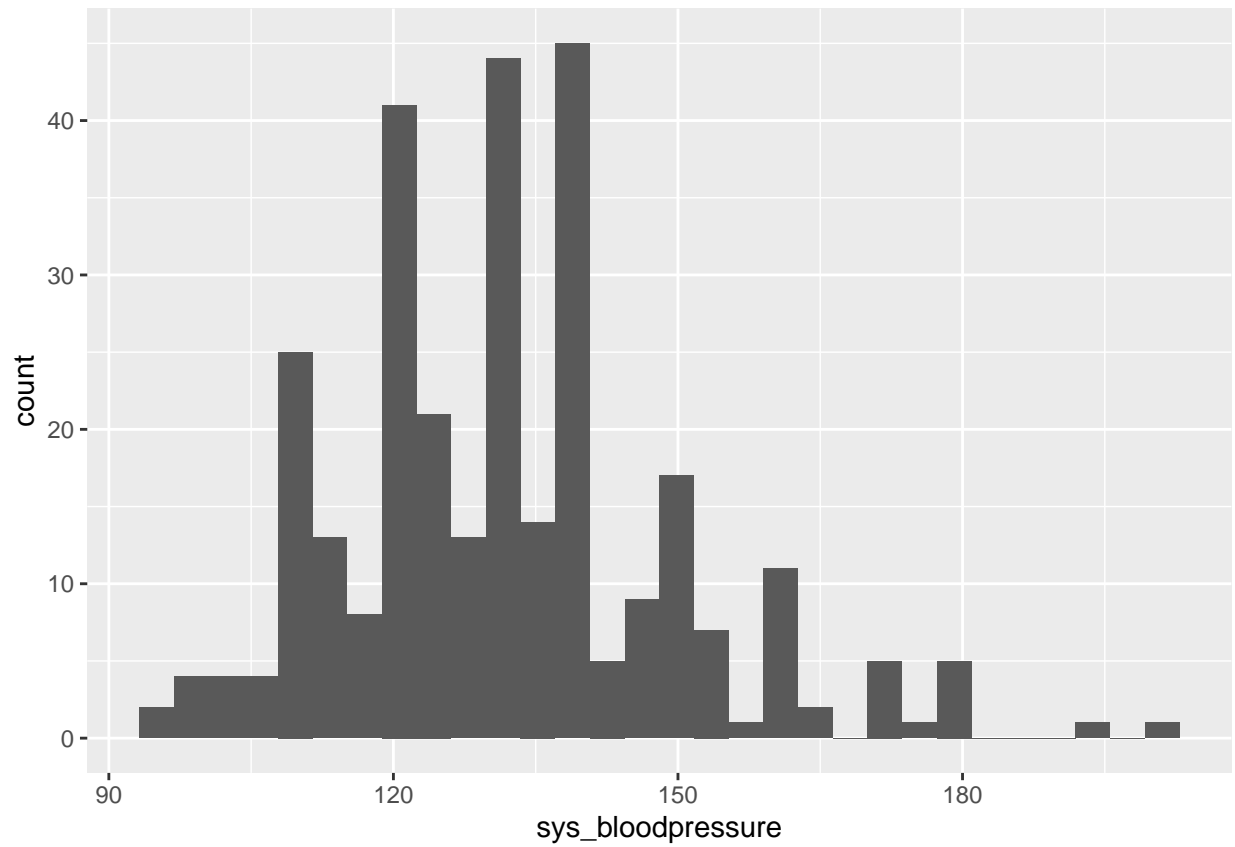
Vizsgáljuk meg tehát ezeket a változókat ábrákkal.

```
heart_data %>%
  ggplot() +
    aes(x = disease_status) +
    geom_bar()
```



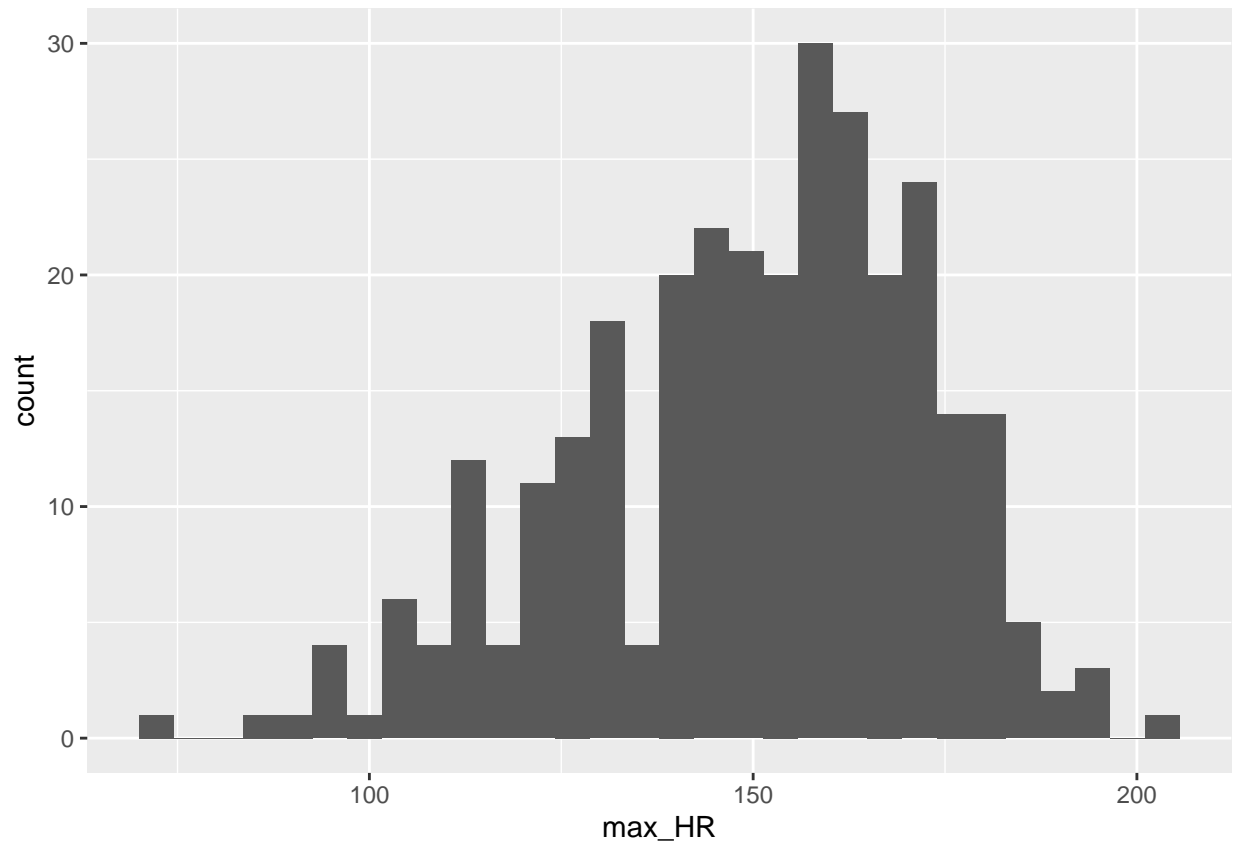
```
heart_data %>%  
  ggplot() +  
    aes(x = sys_bloodpressure) +  
    geom_histogram()
```

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

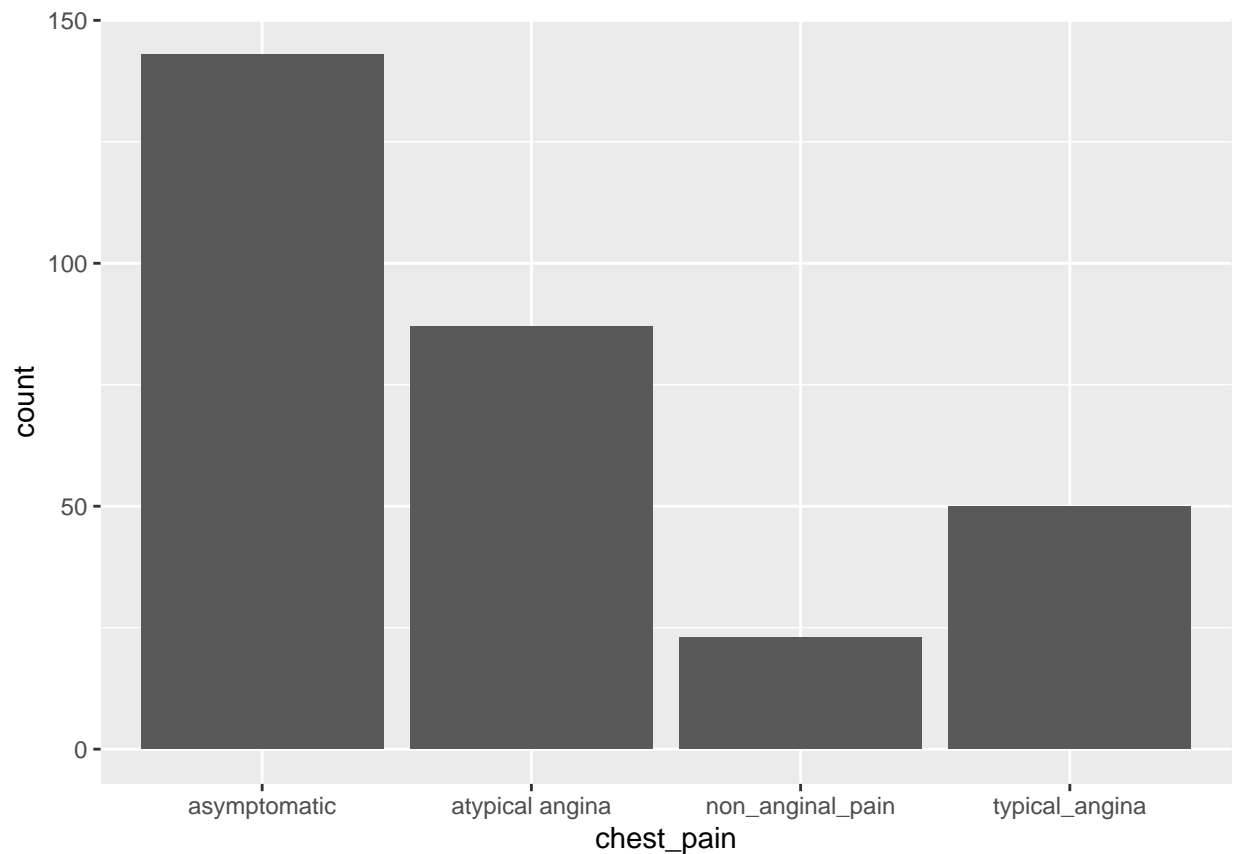


```
heart_data %>%  
  ggplot() +  
    aes(x = max_HR) +  
    geom_histogram()
```

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



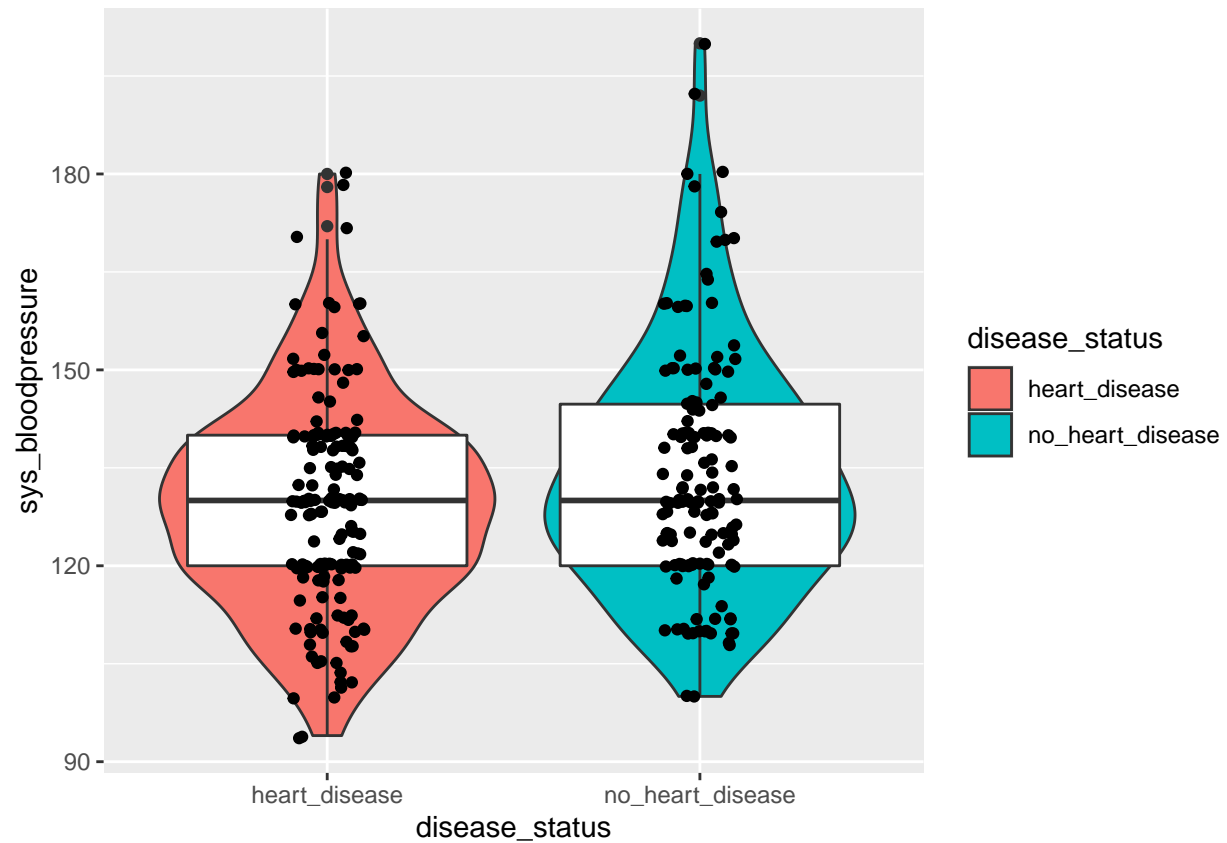
```
heart_data %>%  
  ggplot() +  
    aes(x = chest_pain) +  
    geom_bar()
```



```
heart_data %>%
  group_by(disease_status) %>%
  summarize(mean = mean(sys_bloodpressure),
            sd = sd(sys_bloodpressure))
```

```
## # A tibble: 2 x 3
##   disease_status    mean    sd
##   <fct>          <dbl> <dbl>
## 1 heart_disease    129.  16.2
## 2 no_heart_disease 134.  18.7
```

```
heart_data %>%
  ggplot() +
  aes(y = sys_bloodpressure, x = disease_status) +
  geom_violin(aes(fill = disease_status)) +
  geom_boxplot() +
  geom_jitter(width = 0.1)
```

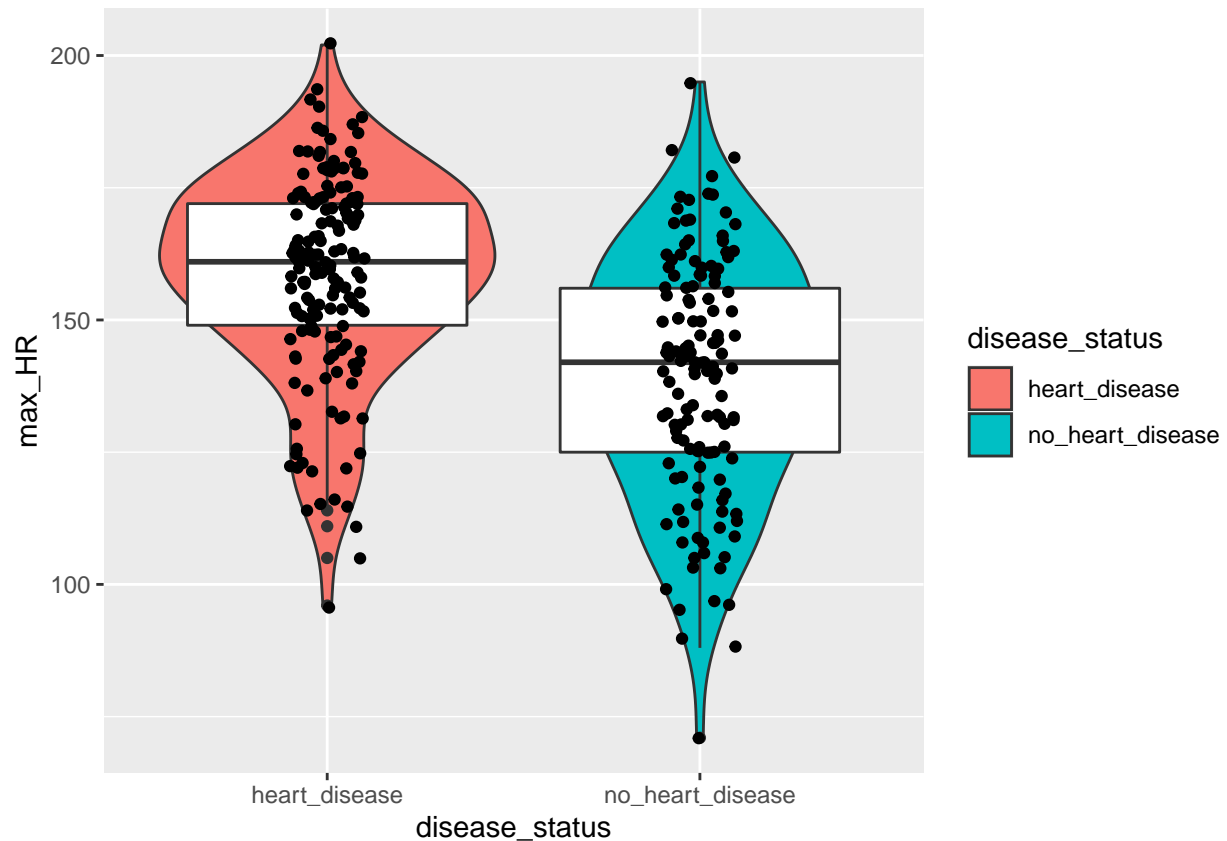


```
heart_data %>%
  group_by(disease_status) %>%
  summarize(mean = mean(max_HR),
            sd = sd(max_HR))
```

```
## # A tibble: 2 x 3
##   disease_status    mean    sd
##   <fct>          <dbl> <dbl>
## 1 heart_disease    158.   19.2
## 2 no_heart_disease 139.   22.6
```

```
heart_data %>%
  ggplot() +
  aes(y = max_HR, x = disease_status) +
  geom_violin(aes(fill = disease_status)) +
  geom_boxplot() +
  geom_jitter(width = 0.1)
```



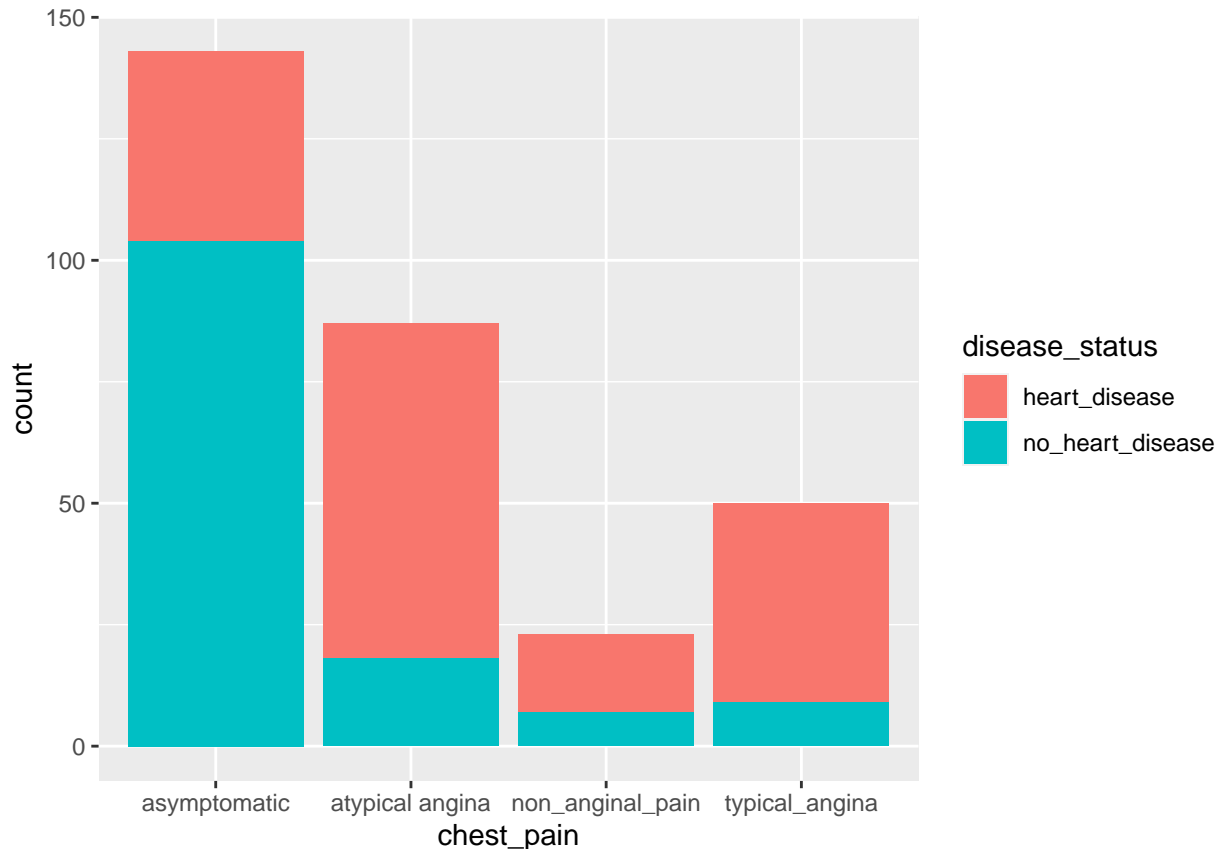


```
heart_data %>%
  group_by(disease_status, chest_pain) %>%
  summarize(n = n()) %>%
  spread(disease_status, n)
```

## 'summarise()' has grouped output by 'disease\_status'. You can override using  
## the '.groups' argument.

```
## # A tibble: 4 x 3
##   chest_pain      heart_disease no_heart_disease
##   <fct>          <int>          <int>
## 1 asymptomatic        39            104
## 2 atypical angina      69             18
## 3 non_anginal_pain    16              7
## 4 typical_angina      41              9
```

```
heart_data %>%
  ggplot() +
  aes(x = chest_pain, fill = disease_status) +
  geom_bar()
```



A vérnyomás, a szívfrekvencia és a mellkasi fájdalom betegségstátussal való kapcsolatának elemzése alapján úgy tűnik, hogy a szisztolés vérnyomás csak csekély mértékben, míg a terhelés által kiváltott szívfrekvencia jelentősebb mértékben függ össze a szívbetegséggel.

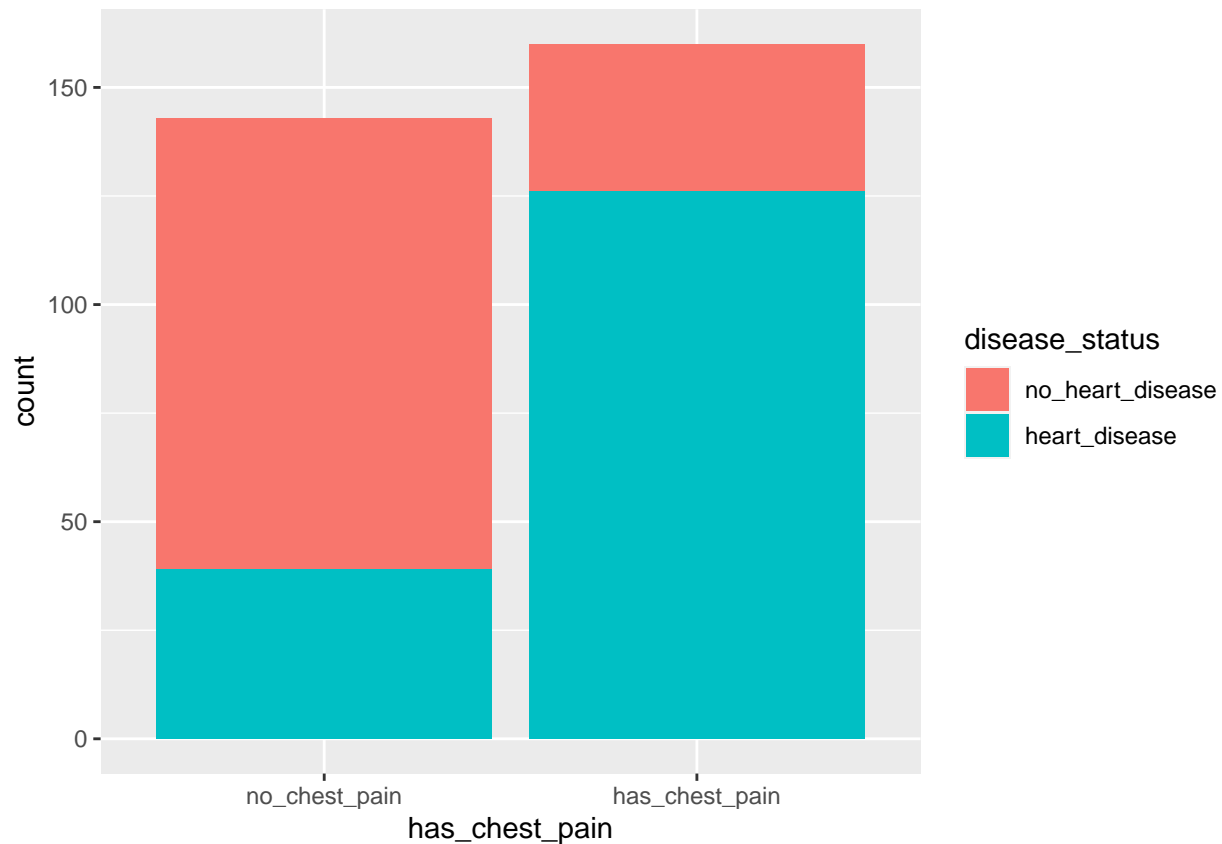
A mellkasi fájdalom is szívbetegségre utaló jelnek tűnik, de nem tűnik úgy, hogy a különböző mellkasi fájdalomkategóriák között lényeges különbség lenne, ezért egyesítjük a három mellkasi fájdalomtípust egyetlen csoportba a tünetmentes csoporttal szemben. Létrehozunk tehát egy új változót `has_chest_pain` néven, amelynek lehetséges szintjei `no_chest_pain` és `has_chest_pain`.

Az explorátoros elemzésből az is kiderül, hogy a `disease_status` faktorban a referenciaszint a `heart_disease`. Egy regressziós keretrendszerben, amikor egy eseményt akarunk megjósolni, jobb, ha a `no_heart_disease` a referenciaszint, az `heart_disease` pedig nem a referenciaszint, így a pozitív regressziós együtthatók az bejósolni kívánt esemény nagyobb esélyének felelnek meg. Ezért megadjuk, hogy a `no_heart_disease` legyen a `disease_status` referenciaszintje. Hasonló okokból adjuk meg, hogy a `no_chest_pain` legyen a `has_chest_pain` változó referenciaszintje.

```
heart_data = heart_data %>%
  mutate(has_chest_pain = factor(recode(chest_pain,
    "asymptomatic" = "no_chest_pain",
    "typical_angina" = "has_chest_pain",
    "atypical_angina" = "has_chest_pain",
    "non_anginal_pain" = "has_chest_pain"), levels = c("no_chest_pain", "has_chest_pain"),
    disease_status = factor(disease_status, levels = c("no_heart_disease", "heart_disease"))
  )

heart_data %>%
  ggplot() +
```

```
aes(x = has_chest_pain, fill = disease_status) +  
geom_bar()
```



## Logosztikus regressziós elemzés

### A lineáris regresszió alkalmatlansága kategorikus kimenetek előrejelzésére

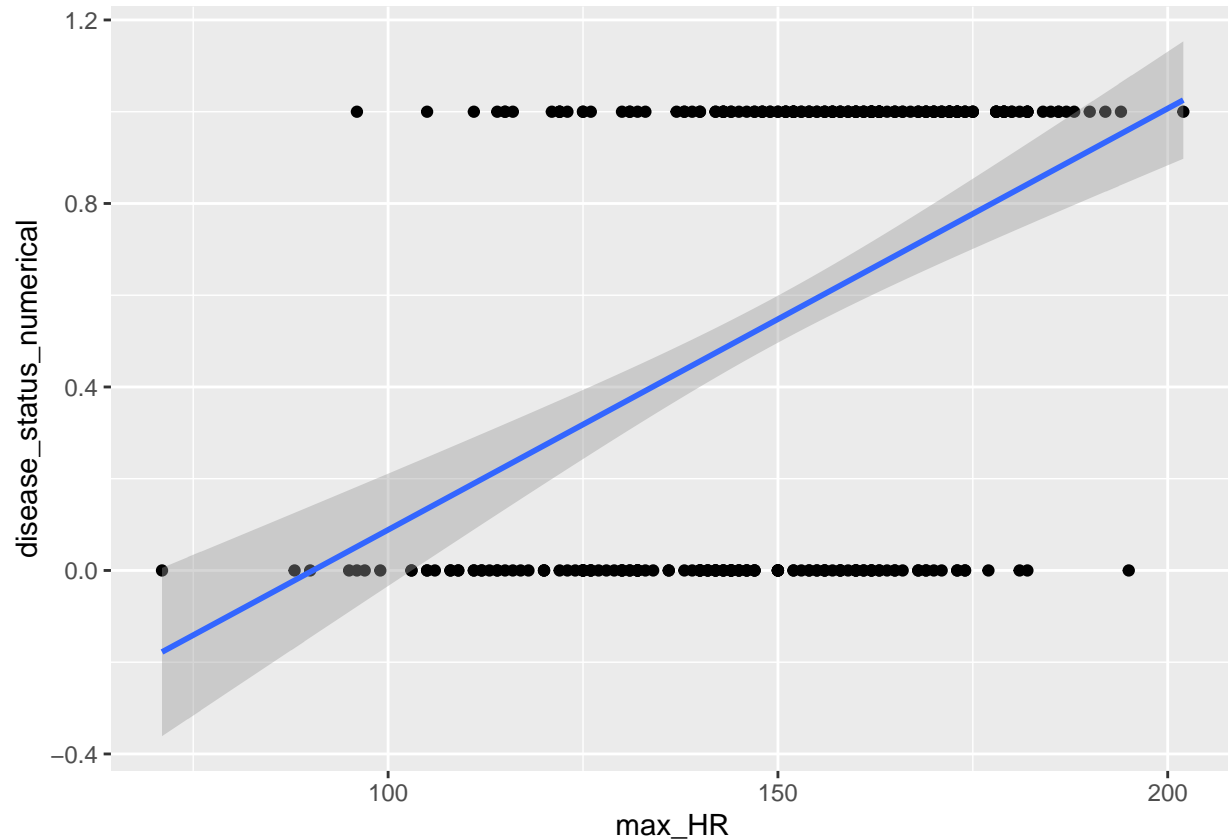
Az előző feladatok alapján azt gondolhatod, hogy egy lineáris regressziós modell illesztése az adatokra jó megoldás lenne itt, ezért kezdjük azzal, hogy megnézzük, mi lenne az eredménye egy szabályos lineáris regressziónak ezekkel az adatokkal.

Nézzük meg egy lehetséges prediktor és a megjósolt eredmény kapcsolatát. Itt a betegség állapotának és a testmozgás során elért maximális pulzusszámnak az összefüggéseit ábrázoljuk egy pontdiagram segítségével, hogy lássuk azt a regressziós egyenest, amelyet egy egyszerű lineáris regresszióval kapnánk. (Ehhez újra kell kodolnunk a disease\_status változót, hogy numerikus változó legyen, hiszen a lineáris regresszió kimeneti változója csak numerikus lehet. Ezért a "no\_heart\_disease" 0 kódot kap, a "heart\_disease" pedig 1-es kódot.)

```
heart_data = heart_data %>%  
  mutate(disease_status_numerical = recode(disease_status,  
                                            "no_heart_disease" = 0,  
                                            "heart_disease" = 1))  
  
heart_data %>%
```

```
ggplot() +
  aes(y = disease_status_numerical, x = max_HR) +
  geom_point() +
  geom_smooth(method = "lm")
```

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



```
lm(disease_status_numerical ~ max_HR, data = heart_data)
```

```
##
## Call:
## lm(formula = disease_status_numerical ~ max_HR, data = heart_data)
##
## Coefficients:
## (Intercept)      max_HR
##   -0.829920     0.009185
```

A grafikonból és a regressziós egyenletből több probléma is kitűnik:

1. A regressziós egyenes nagyon rosszul illeszkedik az adatokhoz.
2. Ha szabályos regressziós modellt illesztünk az imént létrehozott disease\_status\_numerical változóval mint kimenettel és a max\_HR prediktorral, akkor a max\_HR-re 0,009-es regressziós együtthatót kapunk. Ez azt jelenti, hogy minden 1 pontos pulzusszám-növekedés esetén 0,009 pontos növekedés

következik be a kimeneti változóban. Ha most kiszámítjuk a várható kimeneti értéket egy olyan személy esetében, akinek 120 a legmagasabb elért pulzusszám. Az eredmény  $-0,83 + 0,009 \cdot 120 = 0,25$ . Ennek az előrejelzésnek nem igazán van értelme, ha a kimenetel csak 0 vagy 1 lehet. Ezt az előre jelzett értéket úgy tekinthetjük, mint annak a valószínűségét, hogy a kimeneti változó értéke 0 helyett 1 lesz. De azt is láthatjuk, hogy a modell könnyen adhat negatív számokat is előrejelzésként, miközben a valószínűségek csak 0 vagy 1 értéket vehetnek fel. Tegyük fel hogy egy nagyon sportos személynél 90-es maximális pulzusszámot mértünk a terheléses teszten. A jóslat “valószínűsége” annak, hogy ennek a személynek szívbetegsége van,  $-0,02$  lenne, de negatív valószínűség nem létezik. Tehát egy másik megközelítésre van szükségünk, amely reális predikciókat ad.

## Logisztikus regresszió alapötlete

A megoldás az, hogy a hagyományos lineáris modell helyett generalizált lineáris modelleket (GLM) használunk az előrejelzéshez. A GLM-eket olyan kimeneti változók modellezésére tervezték, amelyek nem normális eloszlásúak. A GLM-ek családjának egyik tagja a logisztikus regresszió, amelyet kifejezetten bináris kimenetek (kategorikus kimeneti változók aminek csak két szintje lehet) modellezésére terveztek.

A GLM-ek alapgondolata az, hogy egy olyan kapcsolási függvényt (link függvényt) használnak, amely a megjósolt kimeneti változót realisztikus skálájúvá transzformálja. A logisztikus regresszió által használt linkfüggvény a logit függvény (amely a bejósolni kívánt esemény esélyének természetes alapú logaritmusát). Tehát a logisztikus regresszióban ahelyett, hogy a tényleges kimeneti értéket (“heart disease”, “no heart disease”) jósolnánk meg, a bejósolni kívánt esemény odds-jának természetes alapú logaritmusát jósoljuk meg. Ez a szám pedig a szokásos lineáris regressziós keretben kezelhető, mivel az  $\log(\text{odds})$  a negatív végtelen és a végtelen között bármilyen értéket felvehet. Miután tehát kiszámítottuk a bejósolni kívánt esemény  $\log(\text{odds})$ -ját a regressziós egyenlet segítségével bármely esetre, egy egyszerű számítással megkaphatjuk ebből az esemény odds-ját vagy az esemény valószínűségét is.

## A logisztikus regressziós modell R-ben

Most próbáljuk ki ezt a gyakorlatban. Készítünk egy logisztikus regressziós modellt, ahol a szívbetegség meglétét a `max_HR` prediktorral jelezzük előre.

A regressziós modellt hasonló formulával építhetjük fel, mint a lineáris regressziós modellt, azzal a különbséggel, hogy a `glm()` függvényt használjuk az `lm()` helyett, és meg kell adnunk a `family = binomial()` értéket, hogy logisztikus regressziót kapjunk.

```
mod1 = glm(disease_status ~ max_HR, family = binomial(), data = heart_data)

summary(mod1)

##
## Call:
## glm(formula = disease_status ~ max_HR, family = binomial(), data = heart_data)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1383  -1.0780   0.6043   0.9200   2.1354
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.391452   0.987133  -6.475 9.50e-11 ***
## max_HR       0.043951   0.006531   6.729 1.71e-11 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 359.26  on 301  degrees of freedom
## AIC: 363.26
##
## Number of Fisher Scoring iterations: 4
```

## Odds, esélyhányados és valószínűség

Most, hogy felépítettünk egy modellt, megnézhetjük, mit jósol a modell. A regressziós egyenlet segítségével ugyanúgy kiszámíthatjuk az egyes megfigyelésekre vonatkozó előrejelzett értéket, vagy akár új esetekre vonatkozó értéket, mint korábban, de nem szabad elfelejtenünk, hogy az általunk előrejelzett érték a bejósolni kívánt **esemény bekövetkezésének odds-jának a természetes alapú logaritmusa**, vagyis röviden az esemény  $\log(\text{odds})$ -ja. Az esemény  $\log(\text{odds})$  értékét nyers formában nehéz értelmezni, ezért általában **átváltjuk Odds-ra**. Ahhoz, hogy ennek értelme legyen, meg kell értenünk az Odds és a valószínűségek jelentését, és azt, hogy hogyan alakíthatjuk át egyiket a másikra.

### Esély (Odds)

Valamely esemény **esélye** (odds-ja) azt tükrözi, hogy az esemény bekövetkezésének mi az esélye. Az odds-ot általában számpárként ábrázolják. Ha az esemény/eredmény egy repülőgép-szerencsétlenség túlélése, és a túlélés esélye 1 a 4-hez, ez azt jelenti, hogy átlagosan minden 1 túlélő személyre 4 olyan személy jut, aki nem éli túl. Ez úgy is felírható, hogy az odds 0,25, mivel  $1/4 = 0,25$ . Ha a túlélés esélye 2 az egyhez, ez azt jelenti, hogy átlagosan minden 2 túlélőre 1 ember jut, aki meghal. Az esélyt itt is fel lehet írni úgy, hogy 2, mivel  $2/1 = 2$ .

Az odds-t az exponenciális, **exp()**, függvény segítségével kiszámíthatjuk az  $\log(\text{Odds})$  értékéből:  $\text{Odds} = \exp(\log(\text{Odds}))$ . Fontos felismerni, hogy az odds értéke függ a nézőpontunktól, attól, hogy mi az az esemény, ami minket érdekel. Ebben a példában a túlélés az érdekes eseményünk, és ha az érdekes eseményt a repülőgép-szerencsétlenségben való nem túlélésre (halálra) (a másik lehetséges kimenetelre) cserélnénk, akkor az odds a túlélés esélyének inverze lenne. Tehát ha a túlélés odds-ja 0,25, akkor a halál odds-ja  $1/0,25 = 4$ . Ha a túlélés odds-ja 2, akkor a halálozás odds-ja  $1/2 = 0,5$ .

### Esélyhányados (Odds ratio)

Egy másik fontos, az esélyekkel kapcsolatos fogalom az **esélyhányados** (odds ratio). Ez egy hatásméret-mutató, amely annak értékelésére használható, hogy egy bizonyos csoporthoz való tartozás vs. a csoporthoz való nem tartozás milyen hatással van a vizsgált esemény esélyeire. Az esélyhányados az érdekes esemény odds-jainak aránya két csoport között. Ezt úgy számítják ki, hogy a csoportban a vizsgált esemény esélyét elosztják a csoporton kívüli esemény esélyével. Alapvetően azt mutatja meg, hogy mennyivel nagyobb vagy kisebb az esemény odds-ja (kockázata) az egyik csoportban, mint a másik csoportban. Egy egyszerű példa, ha egy esemény odds-ját hasonlítjuk össze két csoportban, tehát számítsuk ki a nemhez tartozó odds ratio-t egy hajószerecsétlenségben való halálozásra nézve. Képzeli el, hogy minden 1 férfira, aki túlél egy repülőgép-szerencsétlenséget, átlagosan 6 olyan férfi jut, aki meghal a hajószerecsétlenségben, így a férfiak esetében a hajószerecsétlenségben való halálozás esélye 6 lesz. Tegyük fel, hogy valamilyen oknál fogva a nőknél kisebb a kockázata annak, hogy hajószerecsétlenségben meghalnak: minden 1 nőre, aki túléli a balesetet, 2 nő jut, aki meghal, így a repülőgép-balesetben való halálozás esélye a nők esetében 2. Most kiszámíthatjuk az esélyhányadost a két odds egymással való elosztásával: Tehát a hajóbalesetben való halálra a férfiak esélyhányadosa a nőkhez képest  $6/2 = 3/1 = 3$ , mivel a férfiak odds-ja a hajóbalesetben való halálra háromszorosa a nőkének.

Az oddshoz hasonlóan az esélyhányados is függ a nézőpontunktól: attól, hogy mi az a csoport, amelyik érdekel. Tehát ha a férfiak (a nőkhez képest) repülőgép-szerencsétlenségben való halálának esélyhányadosa 3, akkor a nők (a férfiakhoz képest) repülőgép-szerencsétlenségben való halálának esélyhányadosa  $2/6 = 1/3 = 0,3333$ . A 0,33-as esélyhányados azt jelenti, hogy a repülőgép-balesetben való halál esélye 0,33-szor akkora a nőknél, mint a férfiaknál. Más szóval, a halálozás esélye 33% a nők körében a férfiakhoz képest.

## Valószínűség

A valószínűség (probability) azt mutatja, hogy a sok kísérlet során várhatóan milyen arányban fordul elő az érdekes esemény. Ha a valószínűség  $1/4$ , akkor átlagosan 4-ből 1 alkalommal számíthatunk arra, hogy az adott eseményt látjuk. Tehát ha a repülőgép-szerencsétlenség túlélésének valószínűsége  $1/4 = 0,25 = 25\%$ , akkor 4 emberből 1 túléli, 3 pedig meghal. Ha viszont a valószínűség  $3/4 = 0,75 = 75\%$ , akkor azt várjuk, hogy 4 emberből átlagosan 3 túléli és 1 meghal.

A valószínűséget a  $\log(\text{Odds})$ -ból a következő képlettel tudjuk kiszámítani:  $p = \text{Odds} / (1 + \text{Odds})$  vagyis  $p = \exp(\log(\text{Odds})) / (1 + \exp(\log(\text{Odds})))$ .

## A logisztikus regresszió eredményeinek értelmezése

Most, hogy tudjuk, mi az esély (odds), az esélyhányados és a valószínűség, és hogyan kapcsolódnak egymáshoz, készen állunk a modell eredményeinek értelmezésére. Használjuk a regressziós egyenletet, hogy megkapjuk a 182-es maximális pulzusszámú egyénre vonatkozó bejósolt értéket. A regressziós egyenlet a következő eredményt adja:  $-6,39 + 0,044 * 182 = 1,628$ . Ez a  $\log(\text{odds})$  érték. Ha az érték negatív, az azt jelenti, hogy annak a valószínűsége, hogy az adott esemény bekövetkezik, kisebb, mint annak a valószínűsége, hogy nem következik be. Tehát a mi esetünkben, mivel a  $\log(\text{odds})$  pozitív szám, azt mondhatjuk, hogy nagyobb a valószínűsége annak, hogy az illetőnek szívbetegsége van (ez számunkra az érdekes esemény), mint annak, hogy az illetőnek nincs szívbetegsége. Ezen kívül azonban nehéz tovább értelmezni a  $\log(\text{odds})$ -ot anélkül, hogy odds-á alakítanánk.

A  $\log(\text{odds})$  értéket az  $\exp()$  függvény segítségével alakíthatjuk át odds-á.  $\exp(1,628) = 5,09$ . Tehát annak az esélye, hogy a személy szívbetegségben szenved azzal szemben hogy nem szenved abban 5,09, ami azt jelenti, hogy körülbelül ötször nagyobb az esélye annak, hogy ez a személy szívbeteg, mint annak, hogy nem szenved. Ezt a fenti képlet segítségével valószínűségekre is átváltoztathatjuk:  $p = \text{Odds} / (1 + \text{Odds})$ . Esetünkben  $5,09 / (1 + 5,09) = 0,84$ , ami azt jelenti, hogy a modellünk szerint 84% a valószínűsége annak, hogy ez a személy szívbetegségben szenved.

Mint korábban is láttuk, ezt nem kell kézzel kiszámítanunk minden egyes személyre, az adathalmazunk minden egyes az eredeti adatbázisunkban szereplő megfigyelésére megkaphatjuk a bejósolt  $\log(\text{odds})$  értékeket a `predict()` függvény használatával a modell objektumon.

```
predict(mod1)
```

| ## | 1          | 2           | 3          | 4           | 5          | 6           |
|----|------------|-------------|------------|-------------|------------|-------------|
| ## | 0.20124085 | 1.82743841  | 1.16816913 | 1.43187684  | 0.77260756 | 0.11333828  |
| ## | 7          | 8           | 9          | 10          | 11         | 12          |
| ## | 0.33309471 | 1.21212042  | 0.72865628 | 1.25607170  | 0.64075371 | -0.28222328 |
| ## | 13         | 14          | 15         | 16          | 17         | 18          |
| ## | 1.12421785 | -0.06246686 | 0.72865628 | 0.55285114  | 1.16816913 | -1.38100541 |
| ## | 19         | 20          | 21         | 22          | 23         | 24          |
| ## | 1.12421785 | 0.24519214  | 0.68470499 | 1.47582813  | 1.43187684 | -0.37012585 |
| ## | 25         | 26          | 27         | 28          | 29         | 30          |
| ## | 1.43187684 | 0.72865628  | 0.50889985 | -0.98544385 | 0.50889985 | 0.28914343  |
| ## | 31         | 32          | 33         | 34          | 35         | 36          |

|    |             |             |             |             |             |             |
|----|-------------|-------------|-------------|-------------|-------------|-------------|
| ## | 0.99236399  | -0.23827200 | 1.87138969  | 0.28914343  | -0.89754128 | 0.64075371  |
| ## | 37          | 38          | 39          | 40          | 41          | 42          |
| ## | 1.08026656  | 0.86051013  | 0.11333828  | 0.24519214  | -0.15036943 | 1.51977941  |
| ## | 43          | 44          | 45          | 46          | 47          | 48          |
| ## | 0.11333828  | -0.10641814 | 1.60768198  | 1.16816913  | 1.51977941  | 0.46494857  |
| ## | 49          | 50          | 51          | 52          | 53          | 54          |
| ## | -1.33705413 | 0.64075371  | 0.15728957  | 0.24519214  | 0.02543571  | 1.30002299  |
| ## | 55          | 56          | 57          | 58          | 59          | 60          |
| ## | 1.16816913  | 0.55285114  | 1.78348712  | 1.73953584  | 1.25607170  | 0.59680242  |
| ## | 61          | 62          | 63          | 64          | 65          | 66          |
| ## | -0.67778485 | 0.46494857  | 1.95929227  | -0.58988228 | 0.86051013  | 1.60768198  |
| ## | 67          | 68          | 69          | 70          | 71          | 72          |
| ## | -0.10641814 | 1.30002299  | 1.08026656  | 0.77260756  | 0.06938700  | 0.37704600  |
| ## | 73          | 74          | 75          | 76          | 77          | 78          |
| ## | 2.48670769  | 1.78348712  | 0.86051013  | 0.68470499  | 0.90446142  | 0.81655885  |
| ## | 79          | 80          | 81          | 82          | 83          | 84          |
| ## | 1.69558455  | 0.37704600  | 1.47582813  | 1.08026656  | 0.64075371  | 1.43187684  |
| ## | 85          | 86          | 87          | 88          | 89          | 90          |
| ## | -1.02939513 | 0.64075371  | 0.24519214  | 0.46494857  | 0.55285114  | -1.02939513 |
| ## | 91          | 92          | 93          | 94          | 95          | 96          |
| ## | 1.30002299  | 0.99236399  | 1.03631527  | 0.59680242  | -0.32617457 | -1.51285927 |
| ## | 97          | 98          | 99          | 100         | 101         | 102         |
| ## | 0.50889985  | 0.06938700  | 0.72865628  | 1.21212042  | 1.43187684  | -0.01851557 |
| ## | 103         | 104         | 105         | 106         | 107         | 108         |
| ## | 1.47582813  | 2.13509741  | 0.77260756  | -1.33705413 | -0.63383357 | 0.28914343  |
| ## | 109         | 110         | 111         | 112         | 113         | 114         |
| ## | 0.72865628  | 0.59680242  | 0.37704600  | 1.21212042  | -0.54593099 | 0.68470499  |
| ## | 115         | 116         | 117         | 118         | 119         | 120         |
| ## | 0.42099728  | 1.08026656  | 0.99236399  | 0.72865628  | 1.16816913  | 0.28914343  |
| ## | 121         | 122         | 123         | 124         | 125         | 126         |
| ## | -1.02939513 | 1.60768198  | 1.16816913  | 0.94841270  | 1.47582813  | 2.04719484  |
| ## | 127         | 128         | 129         | 130         | 131         | 132         |
| ## | -0.10641814 | 1.16816913  | 1.03631527  | -1.07334642 | 0.77260756  | 0.72865628  |
| ## | 133         | 134         | 135         | 136         | 137         | 138         |
| ## | 0.72865628  | 0.33309471  | 0.77260756  | 0.77260756  | -2.17212855 | -0.23827200 |
| ## | 139         | 140         | 141         | 142         | 143         | 144         |
| ## | -0.85358999 | -1.77656698 | 0.50889985  | 1.56373070  | 1.21212042  | -0.15036943 |
| ## | 145         | 146         | 147         | 148         | 149         | 150         |
| ## | -1.29310284 | -0.10641814 | 0.15728957  | 1.12421785  | 1.03631527  | 0.20124085  |
| ## | 151         | 152         | 153         | 154         | 155         | 156         |
| ## | -0.32617457 | -0.89754128 | 0.42099728  | 0.28914343  | 0.28914343  | -0.63383357 |
| ## | 157         | 158         | 159         | 160         | 161         | 162         |
| ## | 1.47582813  | 1.25607170  | -0.06246686 | 0.77260756  | 1.03631527  | 0.90446142  |
| ## | 163         | 164         | 165         | 166         | 167         | 168         |
| ## | 1.60768198  | 1.21212042  | 1.21212042  | -1.64471313 | -0.72173614 | 0.64075371  |
| ## | 169         | 170         | 171         | 172         | 173         | 174         |
| ## | 0.06938700  | 0.42099728  | -0.15036943 | 0.99236399  | 0.64075371  | 1.21212042  |
| ## | 175         | 176         | 177         | 178         | 179         | 180         |
| ## | -0.58988228 | -1.38100541 | 0.64075371  | 0.55285114  | -1.11729770 | -1.46890799 |
| ## | 181         | 182         | 183         | 184         | 185         | 186         |
| ## | -0.58988228 | -1.38100541 | 1.03631527  | 0.86051013  | -0.76568742 | 0.33309471  |
| ## | 187         | 188         | 189         | 190         | 191         | 192         |
| ## | -0.06246686 | -1.60076184 | 0.77260756  | 0.55285114  | -0.15036943 | -0.63383357 |
| ## | 193         | 194         | 195         | 196         | 197         | 198         |



```
## -1.42495670 -0.15036943 0.42099728 -0.23827200 0.06938700 0.77260756
##      199      200      201      202      203      204
## -2.04027469 0.55285114 1.38792556 -0.19432071 -1.51285927 0.20124085
##      205      206      207      208      209      210
## -0.01851557 0.68470499 -0.15036943 0.50889985 -0.28222328 0.72865628
##      211      212      213      214      215      216
## 0.20124085 -0.23827200 -0.23827200 0.02543571 -0.06246686 -0.41407714
##      217      218      219      220      221      222
## -2.12817726 -0.58988228 -0.80963871 0.20124085 0.37704600 -1.51285927
##      223      224      225      226      227      228
## 1.25607170 -0.54593099 -0.85358999 -0.89754128 -1.86446955 -0.67778485
##      229      230      231      232      233      234
## 0.59680242 -0.63383357 0.28914343 -0.94149256 -0.01851557 -2.17212855
##      235      236      237      238      239      240
## -1.60076184 1.21212042 1.12421785 1.08026656 0.72865628 0.46494857
##      241      242      243      244      245      246
## -1.46890799 -0.10641814 -0.58988228 -2.52373883 -1.77656698 0.90446142
##      247      248      249      250      251      252
## 0.20124085 -1.11729770 2.17904869 0.02543571 -1.02939513 -0.10641814
##      253      254      255      256      257      258
## -1.73261570 -0.89754128 -0.89754128 0.06938700 -0.67778485 -0.85358999
##      259      260      261      262      263      264
## 0.37704600 1.60768198 0.86051013 0.64075371 -2.21607983 1.03631527
##      265      266      267      268      269      270
## -1.64471313 -0.58988228 -1.24915156 -0.85358999 -1.29310284 -1.86446955
##      271      272      273      274      275      276
## -0.06246686 -0.01851557 -3.27091068 0.46494857 -1.20520027 0.99236399
##      277      278      279      280      281      282
## -1.77656698 -0.19432071 0.28914343 -0.89754128 -0.89754128 0.46494857
##      283      284      285      286      287      288
## -0.50197971 1.56373070 -0.32617457 -1.11729770 0.72865628 0.81655885
##      289      290      291      292      293      294
## -0.10641814 -0.67778485 0.68470499 -0.23827200 0.02543571 0.20124085
##      295      296      297      298      299      300
## -0.06246686 -0.06246686 -0.41407714 -2.43583626 -0.98544385 -0.58988228
##      301      302      303
## -0.19432071 -1.33705413 1.25607170
```

## Modell teljesítmény

### Pszedó R négyzet

A modell előrejelző képességének ismert mutatója az  $R^2$  index. A logisztikus regresszióhoz azonban nem létezik pontos  $R^2$  mutató. Ehelyett a megmagyarázott variancia arányát különböző statisztikai eljárásokkal becsüljük, amelyeket pseudo R négyzet módszereknek nevezünk. Több pseudo R négyzet index létezik, mint például a Cox és Snell  $R^2$ , Nagelkerke  $R^2$  és a McFadden  $R^2$ . Ezek egyike sem általánosan elfogadott jó  $R^2$  becslés, de a Cox és Snell  $R^2$ , valamint a Nagelkerke  $R^2$  komoly hátrányokkal küszködik, ezért ha egyáltalán használunk  $R^2$ -t logisztikus regresszióhoz, akkor az a McFadden  $R^2$ . A Cox és Snell  $R^2$ -tel az a probléma, hogy van egy felső határa, amely alacsonyabb, mint 1. Azok számára, akik a jó öreg  $R^2$  indexhez szoktak hozzá, amely 0 és 1 között bármilyen értéket felvehet, és így a megmagyarázott variancia arányát mutatja, ez megnehezítheti a Cox és Snell  $R^2$  értelmezését. A Nagelkerke  $R^2$ -t ennek ellensúlyozására dolgozták ki azáltal, hogy a Cox és Snell  $R^2$  skáláját kiterjesztették, hogy az 1-ig terjedjen. Az ehhez használt korrekciót azonban gyakran túlkompenzációnak tekintik, és ez irreálisan magas  $R^2$ -értékeket adhat

vissza.

Így marad a **McFadden  $R^2$** . Ezt a mutatót a `pscl` csomag `pR2()` függvényének futtatásával kaphatjuk meg.

A `pR2()` függvény outputja megmutatja számunkra a modell log likelihoodját is az “llh” oszlopban. Ezt a számot -2-vel megszorozva kiszámíthatjuk a -2 Log Likelihoodot (-2LL), amelyet a szakirodalomban “devianciának” is neveznek. Ennek ugyanaz az általános jelentése, mint a rezidual sum of squares-nek (RSS) a hagyományos regresszióban. Összehasonlítva a bejósolni kívánt változóra kiszámított becült értéket a tényleges értékkel, és ezeket a különbségeket összegzi, hogy a modell teljes hibájának mértékét adja meg. Az RSS-hez és az AIC-hoz hasonlóan a -2LL-t is nehéz értelmezni az adott modell kontextusán kívül, mivel értéke függ a minta méretétől, és a modellben szereplő paraméterek számától, de ugyan azon az adaton ugyan annak a kimeneti változónak a bejósolására szolgáló két modell összehasonlítására alkalmas. Lényegében megmutatja a hiba összegét, amely a modellünkben szereplő prediktorokkal magyarázott összes variancia figyelembevétele után marad, és minél alacsonyabb ez a szám, annál jobb a modell illeszkedése. (Ez persze azt is jelenti, hogy minél magasabb a log likelihood, annál jobb a modell illeszkedése).

```
pR2(mod1)
```

```
## fitting null model for pseudo-r2
```

```
##          llh      llhNull      G2      McFadden      r2ML      r2CU
## -179.6284602 -208.8190283  58.3811363  0.1397888  0.1752517  0.2342923
```

```
# -2LL, deviance
```

```
pR2(mod1)["llh"] * -2
```

```
## fitting null model for pseudo-r2
```

```
##          llh
## 359.2569
```

## Előrejelzési pontosság

Bizonyos helyzetekben nem elég, ha az érdekes esemény odds-ját kapjuk meg az egyes személyekre, hanem szeretnénk egy konkrét választ kapni arra, hogy az adott megfigyelés melyik kategóriába sorolható, abba, amiben az esemény bekövetkezik, vagy abba amiben nem következik be. (Vagyis például hogy egy személynek van-e vagy nincs szívbetegsége.) Ahhoz hogy konkrét csoportokba soroljuk a megfigyeléseket, általában egy határértéket használunk, és ez alapján a  $\log(\text{odds})$ , odds, vagy valószínűségi határérték alapján soroljuk csoportba a megfigyeléseket. A legegyszerűbb megoldás, ha a megfigyeléseket a “nincs esemény” kategóriába soroljuk, ha az adott megfigyelés esetében az esemény valószínűsége 50% vagy annál kisebb, és “van esemény” kategóriába soroljuk, ha az érdekes esemény valószínűsége 50%-nál nagyobb az adott megfigyelés esetében. A fenti átváltási formulák alapján könnyen belátható hogy az 50%-os valószínűség 1-es odds-nak és 0  $\log(\text{odds})$ -nak felel meg, így valójában közvetlenül használhatjuk a modellünk előrejelzését (amely  $\log(\text{odds})$ -ban van megadva) a kategorizáláshoz: a 0 vagy annál kisebb  $\log(\text{odds})$ -ot “no\_heart\_disease”, míg a 0-nál nagyobb  $\log(\text{odds})$ -ot “heart\_disease”-ként kódoljuk, mivel a szívbetegség az érdekes esemény a vizsgálatunkban. Az alábbi kódban ezt úgy tesszük, hogy a kimeneti változó (disease status) bejósolt értékét egy új változóba mentjük, amelynek neve “pred\_mod1”.

```
heart_data = heart_data %>%
  mutate(pred_mod1 = predict(mod1)) %>%
  mutate(pred_mod1 = case_when(pred_mod1 <= 0 ~ "no_heart_disease",
                                pred_mod1 > 0 ~ "heart_disease"))
```

Most már összehasonlíthatjuk a modell előrejelzéseit a tényleges `disease_status` értékekkel, hogy lássuk, hányszor kategorizálta helyesen a modellünk a megfigyeléseket. Az eredmények azt mutatják, hogy a 303 esetből 213 esetben (70%) a modell helyesen kategorizálta a megfigyelést, vagyis a modell 70%-ban helyesen jósolta be a `max_HR` alapján hogy a személynek van-e szívbetegsége vagy sem.

```
# coding correct guesses

heart_data = heart_data %>%
  mutate(correct_prediction = case_when(pred_mod1 == disease_status ~ "correct",
                                         pred_mod1 != disease_status ~ "incorrect"))

# correct categorization rate overall

heart_data %>%
  group_by(correct_prediction) %>%
  summarise(count = n()) %>%
  mutate(freq = count / sum(count))

## # A tibble: 2 x 3
##   correct_prediction count  freq
##   <chr>                <int> <dbl>
## 1 correct                213  0.703
## 2 incorrect              90  0.297
```

### Előrejelzési pontosság eredménykategóriánként

Ez a 70%-os előrejelzési arány jó eredménynek tűnhet, de a modell teljesítményét ahhoz képest kell értékelni, hogy milyen pontos lenne egy olyan modell, amely nem használ semmilyen prediktort.

A hagyományos lineáris modellekhez hasonlóan a logisztikus regresszióban is építhetünk null modellt. A lineáris regresszióban a null modell a kimeneti változó átlagát használta előrejelzésként. Ugyanezt tesszük a logisztikus regresszióban is a nullmodellben. Vagyis, az egyes megfigyelésekre vonatkozó előrejelzett  $\log(\text{odds})$ -ot az érdekes esemény előfordulási aránya alapján számítjuk ki. Például a szívbetegség adatbázisban amellyel most dolgozunk, a null modell az alapján számítja ki a becsült  $\log(\text{odds})$ -ot, hogy milyen a szívbetegség előfordulási aránya a teljes mintában. Mivel a vizsgálatban 165 (54%) résztvevőnek volt szívbetegsége, 138-nak (46%) pedig nem volt, annak ez odds-ja hogy valaki szívbetegségben szenved 1.2 (mivel az adathalmazban 1.2-szer annyi szívbeteg van, mint ahányan nem szenvednek szívbetegségben).  $\log(1.2) = 0.18$ . Ez az a szám, amit a null modell előrejelzett kimenetként fog használni minden egyes személyre. Szóval a null modellben minden emberre ugyan azt az esélyt jósoljuk, mint ami a teljese mintában az esemény előfordulási esélye. Ez nem egy túl szofisztikált becslés, de hát ilyen ez a null modell.

Ha ugyanazt az egyszerű módszert használjuk a megfigyelések csoportba sorolására, amit fent is említettünk, hogy a 0 vagy alacsonyabb  $\log(\text{odds})$  esetén “nem esemény” a tippünk, míg 0 feletti  $\log(\text{odds})$ -nál “esemény” a bejósolt csoport, akkor mivel a null modell által jósolt  $\log(\text{odds})$  mindenre 0.18, ez azt eredményezi hogy a null modell mindenkit szívbeteg (“heart\_disease”) csoportba sorol. Ez az “előrejelzés” az esetek 54%-ára helyes, mivel az adathalmazban lévő emberek 54%-ának valóban van szívbetegsége.

```
mod_null = glm(disease_status ~ 1, family = binomial(), data = heart_data)

summary(mod_null)
```

```
##
```

```
## Call:
## glm(formula = disease_status ~ 1, family = binomial(), data = heart_data)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.254  -1.254   1.103   1.103   1.103
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.1787     0.1154   1.549   0.121
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 417.64  on 302  degrees of freedom
## AIC: 419.64
##
## Number of Fisher Scoring iterations: 3
```

```
head(predict(mod_null))
```

```
##           1           2           3           4           5           6
## 0.1786918 0.1786918 0.1786918 0.1786918 0.1786918 0.1786918
```

Más szóval, a null modell előrejelzése a két lehetséges kategória közül mindig a gyakoribb kategória lesz az adathalmaz minden megfigyelése esetén. Ez azt jelenti, hogy minél gyakoribb az egyik kategória/esemény a megfigyelt kimeneti változóban, annál pontosabb lesz összességében a null modell előrejelzése.

Tegyük fel például, hogy logisztikus regresszió segítségével szeretnénk megjósolni, hogy egy személynek van-e COVID-ja néhány egészségügyi mérőszám alapján. Ha azonban az emberek 95%-a nem COVID-os a vizsgált személyek közül, akkor még a nullmodell is 95%-os pontossággal fogja megjósolni a kimenetet mindenféle egészséggel kapcsolatos prediktor használata nélkül, mivel mindenki számára azt fogja jóslni, hogy “nincs COVID”, és mivel az emberek 95%-a nem COVID-os, ez az előrejelzés az egyének 95%-ánál helyes lesz. Valójában a null modell minden egyes esetet helyesen fog a gyakoribb kategóriába sorolni, de tévedni fog minden olyan esetben, amely a másik kategóriába tartozik.

Ez azt jelenti, hogy nem elegendő a modell predikciós hatékonyságát megnézni az összes adaton. Hanem a predikciós hatékonyságot kategóriánként külön is meg kell vizsgálni. Amint azt már megjegyeztük, a példánkban a null modell előrejelzése az esetek 54%-ában összességében helyes, és a ténylegesen szívbetegekben szenvedő esetek 100%-ában helyes, de a ténylegesen nem szívbetegek esetében 0%-ban helyes. Ehhez képest a prediktorral rendelkező modell 165 szívbeteaggel rendelkező esetből 130 esetben helyesen jóslta meg, hogy a személynek szívbetege van (79%), és 138 szívbetege nélküli esetből 83 esetben helyesen jóslta meg, hogy a személynek nincs szívbetege (60%). Tehát a prediktorokkal ellátott modellünk előrejelzési pontossága lényegesen nagyobb mindkét lehetséges kimenetel előrejelzésében a nullmodellhez képest.

```
# percentage of heart disease

heart_data %>%
  group_by(disease_status) %>%
  summarise(count = n()) %>%
  mutate(freq = count / sum(count))
```

```
## # A tibble: 2 x 3
##   disease_status count freq
```

```
##   <fct>           <int> <dbl>
## 1 no_heart_disease   138 0.455
## 2 heart_disease     165 0.545
```

*# crosstab of disease\_status and predicted values*

```
heart_data %>%
  group_by(disease_status, pred_mod1) %>%
  summarize(n = n()) %>%
  spread(disease_status, n)
```

## 'summarise()' has grouped output by 'disease\_status'. You can override using the '.groups' argument.

```
## # A tibble: 2 x 3
##   pred_mod1      no_heart_disease heart_disease
##   <chr>           <int>         <int>
## 1 heart_disease      55           130
## 2 no_heart_disease  83           35
```

*# correctly categorized as having heart disease*

```
heart_data %>%
  filter(disease_status == "heart_disease") %>%
  group_by(correct_prediction) %>%
  summarise(count = n()) %>%
  mutate(freq = count / sum(count))
```

```
## # A tibble: 2 x 3
##   correct_prediction count  freq
##   <chr>           <int> <dbl>
## 1 correct           130 0.788
## 2 incorrect         35 0.212
```

*# correctly categorized as not having heart disease*

```
heart_data %>%
  filter(disease_status == "no_heart_disease") %>%
  group_by(correct_prediction) %>%
  summarise(count = n()) %>%
  mutate(freq = count / sum(count))
```

```
## # A tibble: 2 x 3
##   correct_prediction count  freq
##   <chr>           <int> <dbl>
## 1 correct           83 0.601
## 2 incorrect        55 0.399
```

## A modell érzékenységének finomhangolása

Lehetőség van a modell érzékenységének finomhangolására, ha azt szeretnénk, hogy a modell érzékenyebben érzékelje az esemény jelenlétét vagy hiányát, azáltal, hogy megváltoztatjuk a küszöbértéket, amely alapján

egy esetet egy bizonyos kategóriába sorolunk. Amint fentebb említettük, a küszöbérték általában  $\log(\text{odds}) > 0$  (ami megegyezik az  $\text{Odds} > 0$  és a valószínűség  $> 50\%$ -val) ahhoz, hogy egy esetet a “van esemény” kategóriába soroljuk, de ha azt szeretnénk, hogy modellünk érzékenyebb legyen az esemény kimutatására, akkor csökkenthetjük ezt a küszöbértéket. Vagy ha azt akarjuk, hogy a modell érzékenyebb legyen azoknak az eseteknek a kategorizálására, ahol nincs esemény, akkor növelhetjük ezt a küszöbértéket. Ez különösen akkor fontos, ha nagy a különbség az esemény és a “nem esemény” előfordulási aránya között a mintában.

A fenti példánkban a modellünk pontosabban kategorizálta a szívbetegséggel rendelkező eseteket (79%), mint a szívbetegséggel nem rendelkező eseteket (60%). Ha úgy akarjuk hangolni a modellt, hogy jobban tudja helyesen kategorizálni a szívbetegséggel nem rendelkezőket, akkor növelhetjük a küszöbértéket, amely alapján egy esetet szívbetegnek minősíthetünk. Az alábbi kódban a küszöbértéket  $\log(\text{odds}) > 0$ -ról  $\log(\text{odds}) > 0,4$ -re növeljük.

Az eredmények azt mutatják, hogy ezzel az új küszöbértékkel az esetek 72%-ában helyesen tudjuk kategorizálni a szívbetegséggel nem rendelkező eseteket, míg a szívbetegséggel rendelkezők helyes kategorizálása még mindig elfogadható: 64%. Ez előnyös lehet, ha a prioritásunk az, hogy minimalizáljuk azokat az eseteket, amikor tévesen diagnosztizáljuk a szívbetegeket. A szívbetegség helyes felismerésének pontosságával és az általános felismerési pontossággal “fizetünk ezért”, de bizonyos esetekben ez még így is kívánatos lehet.

```
heart_data = heart_data %>%
  mutate(pred_mod1_tuned = predict(mod1)) %>%
  mutate(pred_mod1_tuned = case_when(pred_mod1_tuned <= 0.4 ~ "no_heart_disease",
                                     pred_mod1_tuned > 0.4 ~ "heart_disease"))

# coding correct guesses

heart_data = heart_data %>%
  mutate(correct_prediction_tuned = case_when(pred_mod1_tuned == disease_status ~ "correct",
                                              pred_mod1_tuned != disease_status ~ "incorrect"))

# correct categorization rate overall

heart_data %>%
  group_by(correct_prediction_tuned) %>%
  summarise(count = n()) %>%
  mutate(freq = count / sum(count))
```

```
## # A tibble: 2 x 3
##   correct_prediction_tuned count  freq
##   <chr>                  <int> <dbl>
## 1 correct                  204 0.673
## 2 incorrect                99 0.327
```

```
# crosstab of disease_status and predicted values
```

```
heart_data %>%
  group_by(disease_status, pred_mod1_tuned) %>%
  summarize(n = n()) %>%
  spread(disease_status, n)
```

```
## 'summarise()' has grouped output by 'disease_status'. You can override using
## the '.groups' argument.
```

```
## # A tibble: 2 x 3
##   pred_mod1_tuned no_heart_disease heart_disease
##   <chr>           <int>         <int>
## 1 heart_disease      39          105
## 2 no_heart_disease   99          60
```

```
# correctly categorized as having heart disease
```

```
heart_data %>%
  filter(disease_status == "heart_disease") %>%
  group_by(correct_prediction_tuned) %>%
  summarise(count = n()) %>%
  mutate(freq = count / sum(count))
```

```
## # A tibble: 2 x 3
##   correct_prediction_tuned count  freq
##   <chr>                   <int> <dbl>
## 1 correct                 105 0.636
## 2 incorrect                60 0.364
```

```
# correctly categorized as not having heart disease
```

```
heart_data %>%
  filter(disease_status == "no_heart_disease") %>%
  group_by(correct_prediction_tuned) %>%
  summarise(count = n()) %>%
  mutate(freq = count / sum(count))
```

```
## # A tibble: 2 x 3
##   correct_prediction_tuned count  freq
##   <chr>                   <int> <dbl>
## 1 correct                 99 0.717
## 2 incorrect                39 0.283
```

## A modell szignifikánsan jobb, mint a null modell?

Összehasonlíthatjuk a modellt amiben a prediktorok is szerepelnek null modellel egy likelihood ratio teszttel, hogy megnézzük, hogy a mi modellünk szignifikánsan jobb-e a kimenetel előrejelzésében, mint a null modell. Ezt úgy tehetjük meg, hogy a null modell objektumát és a prediktorokat tartalmazó modell objektumát is az `lrtest()` függvénybe helyezzük. A likelihood ratio test a két modell loglikelihoodját állítja szembe egymással, és egy Chi-négyzet tesztstatisztikát és egy p-értéket ad. Ha ez a p-érték kisebb mint 0.05, akkor a modellek szignifikánsan különböznek egymástól az előrejelzési pontosság tekintetében. A nagyobb logLikelihooddal rendelkező modellnek jobb az illeszkedése. Az alábbi példában a teszt szignifikáns eredményt ad, ami azt jelzi, hogy a `max_HR` prediktort tartalmazó modell szignifikánsan jobb, mint a null modell.

A lineáris regressziós modellekhez hasonlóan az Akaike információs kritérium (AIC) segítségével is összehasonlíthatjuk két modell illeszkedését. Mint korábban, az a modell, amelynek AIC-értéke legalább 2 ponttal alacsonyabb, szignifikánsan jobb, mint a másik modell. Ha az AIC differencia kisebb, mint 2 pont, akkor nincs elég bizonyítékunk ahhoz, hogy elvessük a null hipotézist, miszerint a két modell előrejelzési pontossága megegyezik.

```
lrtest(mod_null, mod1)
```

```
## Likelihood ratio test
##
## Model 1: disease_status ~ 1
## Model 2: disease_status ~ max_HR
##   #Df LogLik Df  Chisq Pr(>Chisq)
## 1    1 -208.82
## 2    2 -179.63  1 58.381  2.16e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(mod_null, mod1)
```

```
##           df      AIC
## mod_null  1 419.6381
## mod1      2 363.2569
```

### A regressziós együtthatók értelmezése a modellben

A lineáris regresszióhoz hasonlóan a modell együtthatókat (estimate) a modell summary-ban találja, és a `confint()` függvény segítségével megkaphatjuk az ezekhez a regressziós együtthatókhoz tartozó konfidencia intervallumokat.

```
summary(mod1)
```

```
##
## Call:
## glm(formula = disease_status ~ max_HR, family = binomial(), data = heart_data)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1383  -1.0780   0.6043   0.9200   2.1354
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.391452   0.987133  -6.475 9.50e-11 ***
## max_HR       0.043951   0.006531   6.729 1.71e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 359.26  on 301  degrees of freedom
## AIC: 363.26
##
## Number of Fisher Scoring iterations: 4
```



```
confint(mod1)
```

```
## Waiting for profiling to be done...
```

```
##                2.5 %      97.5 %  
## (Intercept) -8.41257396 -4.53271457  
## max_HR      0.03164292  0.05731192
```

A becsléseket ugyanúgy értelmezhetjük, mint a lineáris regresszió esetében, azzal a fontos különbséggel, hogy a logisztikus regresszióban a modell által megjósolt eredmény nem a kimeneti változó eredeti skáláján van, hanem a modell az esemény log(odds)-ját jósolja meg.

Példánkban a max\_HR-nek megfelelő regressziós együttható 0.032. Ez azt jelenti, hogy a max\_HR skálán minden egyes lépcsőfoknál az esemény (szívbetegség) log(odds) értéke 0.044-gyel nő. Ezt gyakran átkonvertáljuk esélyhányadosra (odds ratio), ezt az  $\exp()$  függvénnyel tehetjük meg. A max\_HR-nek mint prediktorhoz tartozó esélyhányados  $\exp(0.044) = 1.045$ . Ez egy könnyebben értelmezhető hatásméret mint a log(odds): azt mutatja, hogy egy 1 ponttal magasabb max\_HR értékkel rendelkező személynek 1.045-ször nagyobb az esélye a szívbetegségre.

Fontos, hogy amikor a regressziós egyenlet alapján kiszámítjuk az előre jelzett kimenetelt, a becsléseket mindig az eredeti log(odds) skálán kell használnunk, és csak a végén kell az eredményt odds-ra (és ha szükséges, valószínűségekre) konvertálni. Az odds-t és a valószínűséget nem lehet közvetlenül a regressziós egyenletbe behelyettesíteni.

Az interceptet a lineáris regresszióhoz hasonlóan kell értelmezni: ez a bejósolni kívánt esemény becsült log(odds) értéke, ha az összes prediktor értéke nulla.

A lineáris regresszióhoz hasonlóan a modell summary-ban a prediktorokhoz tartozó p-értékeket vagy a confint() függvény által mutatott konfidencia intervallumokat használhatjuk arra, hogy megítéljük, egy adott prediktornak van-e hozzáadott előrejelző értéke a modellben. Más szóval, hogy az adott prediktor regressziós együtthatója szignifikánsan különbözik-e nullától.

## Mit kell leírni egy cikkben az eredményekről?

Most már minden adatunk megvan az eredmények leírásához.

A statisztikai elemzés részbe ezt íránk:

“Binomiális logisztikus regressziós elemzést végeztünk, ahol a szívbetegség jelenléte (disease\_status) volt a bejósolni kívánt esemény (”nincs szívbetegség” volt a referenciaszint), és a modell egy prediktort tartalmazott: a testmozgás során elért maximális pulzusszámot (max\_HR).”

Az eredmények részben először a modellünk egészének modellilleszkedéséről és előrejelzési pontosságáról számolunk be:

“A max\_HR prediktort tartalmazó logisztikus regressziós modell szignifikánsan jobb modellilleszkedést mutatott, mint a null modell ( $\chi^2 = 58.38$ ,  $df = 1$ ,  $p < 0.001$ , a modell AIC-ja = 363.26, a modell -2LL-je = 359.26, a null modell AIC-ja = 419.64, a nullmodell -2LL-je = 417.64). A modell a variancia 14%-át magyarázta meg (McFadden  $R^2 = 0.14$ ). Szívbetegség a mintánkban az esetek 54.5%-ában fordult elő (303 személyből 165). A végleges modell a szívbetegség jelenlétét az esetek 79%-ában, a szívbetegség hiányát pedig a mintánkban szereplő esetek 60%-ában helyesen jósolta meg, az összesített helyes előrejelzési arány 70% volt.”

Ez után általában az egyes prediktorok regressziós együtthatóira vonatkozó információkat, valamint a modellhez hozzáadott prediktív értékekre vonatkozó információkat közölné.

Ezeket táblázatos formában szoktuk bemutatni. A táblázatnak tartalmaznia kell a regressziós együtthatót és az ahhoz tartozó konfidenciaintervallumot, az esélyhányadost, a Z teszt-statisztikát, és a p-értéket minden egyes prediktorra és az interceptre külön-külön.

---

### *Gyakorlas*

---

Jó okunk van feltételezni, hogy a terheléses vizsgálat során elért maximális pulzusszám (`max_HR`), a nyugalmi szisztolés vérnyomás (`sys_bloodpressure`) és a mellkasi fájdalom (`has_chest_pain`) segíthet a szívbetegség (`disease_status`) előrejelzésében.

1. Készíts logisztikus regressziós modellt, amelyben a bejósolt változó a szívbetegség jelenléte (`disease_status`), a prediktorok pedig a terheléses vizsgálat során elért maximális pulzusszám (`max_HR`), a nyugalmi szisztolés vérnyomás (`sys_bloodpressure`) és a mellkasi fájdalom jelenléte (`has_chest_pain`).
  2. Mekkora ennek a modellnek az előrejelzési pontossága? Határozd meg, hogy összességében mennyire pontosak az előrejelzések.
  3. Határozd meg a modell illeszkedését az AIC és a -2LL segítségével, valamint a modell pszeudo  $R^2$  értékét.
-