

Analiza rezervacija hotelskog smještaja

Grupa Prosječni

Mario Mrvčić, Rujana Perić, Dorjan Štrbac, Sven Winkler

2023-12-29

OPIS ZADATKA

U današnje vrijeme turistička djelatnost jako je popularna, a tržište puno konkurenčije. Zanima nas kako hoteli mogu prilagoditi svoje usluge kako bi poboljšali zadovoljnost gostiju i povećali profitabilnost.

Posjedujući skup podataka koji detaljno dokumentira rezervacije u dva različita hotela želimo istražiti, bolje razumjeti i ispitati veze između boravka u hotelu, godišnjeg doba, socio-ekonomskog statusa gosta, cijena, tipa rezervacija i sličnih parametara danih skupom podataka.

Učitavanje podataka o rezervacijama

Skup podataka sastoji se od 119390 podataka o rezervacijama u dva hotela. Svaki podatak o rezervaciji sastoji se od 31 varijable koje pružaju uvid u trendove dolaska, demografske podatke gostiju, njihove finansijske obrasce...

U nastavku je prikazan:

- podatak o broju zapisa rezervacija

```
h1_data = read.csv("Rezervacije/Rezervacije_/H1.csv")
h2_data = read.csv("Rezervacije/Rezervacije_/H2.csv")
data <- rbind(h1_data, h2_data)
dim(data)
```

```
## [1] 119390      31
```

- primjer nekoliko zapisa

```
head(data)
```

```
##   IsCanceled LeadTime ArrivalDateYear ArrivalDateMonth ArrivalDateWeekNumber
## 1           0       342            2015          July                  27
## 2           0       737            2015          July                  27
## 3           0        7            2015          July                  27
## 4           0       13            2015          July                  27
## 5           0       14            2015          July                  27
## 6           0       14            2015          July                  27
##   ArrivalDateDayOfMonth StaysInWeekendNights StaysInWeekNights Adults Children
```

```

## 1          1          0          0          2          0
## 2          1          0          0          2          0
## 3          1          0          1          1          0
## 4          1          0          1          1          0
## 5          1          0          2          2          0
## 6          1          0          2          2          0
##   Babies      Meal  Country MarketSegment DistributionChannel IsRepeatedGuest
## 1    0 BB       PRT     Direct      Direct      0
## 2    0 BB       PRT     Direct      Direct      0
## 3    0 BB       GBR     Direct      Direct      0
## 4    0 BB       GBR     Corporate   Corporate   0
## 5    0 BB       GBR     Online TA   TA/TO     0
## 6    0 BB       GBR     Online TA   TA/TO     0
##   PreviousCancellations PreviousBookingsNotCanceled ReservedRoomType
## 1                  0          0 C
## 2                  0          0 C
## 3                  0          0 A
## 4                  0          0 A
## 5                  0          0 A
## 6                  0          0 A
##   AssignedRoomType BookingChanges   DepositType      Agent      Company
## 1    C           3 No Deposit    NULL        NULL
## 2    C           4 No Deposit    NULL        NULL
## 3    C           0 No Deposit    NULL        NULL
## 4    A           0 No Deposit    304        NULL
## 5    A           0 No Deposit    240        NULL
## 6    A           0 No Deposit    240        NULL
##   DaysInWaitingList CustomerType ADR RequiredCarParkingSpaces
## 1            0   Transient     0          0
## 2            0   Transient     0          0
## 3            0   Transient    75         0
## 4            0   Transient    75         0
## 5            0   Transient   98         0
## 6            0   Transient   98         0
##   TotalOfSpecialRequests ReservationStatus ReservationStatusDate
## 1                  0     Check-Out  2015-07-01
## 2                  0     Check-Out  2015-07-01
## 3                  0     Check-Out  2015-07-02
## 4                  0     Check-Out  2015-07-02
## 5                  1     Check-Out  2015-07-03
## 6                  1     Check-Out  2015-07-03

```

- svi parametri rezervacija

```
names(data)
```

```

## [1] "IsCanceled"                   "LeadTime"
## [3] "ArrivalDateYear"              "ArrivalDateMonth"
## [5] "ArrivalDateWeekNumber"        "ArrivalDateDayOfMonth"
## [7] "StaysInWeekendNights"         "StaysInWeekNights"
## [9] "Adults"                       "Children"
## [11] "Babies"                      "Meal"
## [13] "Country"                     "MarketSegment"

```

```
## [15] "DistributionChannel"
## [17] "PreviousCancellations"
## [19] "ReservedRoomType"
## [21] "BookingChanges"
## [23] "Agent"
## [25] "DaysInWaitingList"
## [27] "ADR"
## [29] "TotalOfSpecialRequests"
## [31] "ReservationStatusDate"
                                         "IsRepeatedGuest"
                                         "PreviousBookingsNotCanceled"
                                         "AssignedRoomType"
                                         "DepositType"
                                         "Company"
                                         "CustomerType"
                                         "RequiredCarParkingSpaces"
                                         "ReservationStatus"
```

Case study: TRAJANJE BORAVKA GOSTA

Slijedi detaljna analiza trajanja boravka gosta s obrazom na različite parametre.

Trajanje prosječnog boravka tjemkom različitih godišnjih doba

Dogovorno su godišnja doba definirana na idući način:

ljeto - lipanj, srpanj, kolovoz
jesen - rujan, listopad, studeni
zima - prosinac, siječanj, veljača
 proljeće -ožujak, travanj, svibanj

Definiramo čeriti pomoćna skupa podataka za svako godišnje doba grupirajući po paramteru "ArrivalDateMonth".

```
summer = rbind(data[data$ArrivalDateMonth == "June",],  
                data[data$ArrivalDateMonth == "July",],  
                data[data$ArrivalDateMonth == "August",])  
  
autumn = rbind(data[data$ArrivalDateMonth == "September",],  
                 data[data$ArrivalDateMonth == "October",],  
                 data[data$ArrivalDateMonth == "November",])  
  
winter = rbind(data[data$ArrivalDateMonth == "December",],  
                 data[data$ArrivalDateMonth == "January",],  
                 data[data$ArrivalDateMonth == "February",])  
  
spring = rbind(data[data$ArrivalDateMonth == "March",],  
                 data[data$ArrivalDateMonth == "April",],  
                 data[data$ArrivalDateMonth == "May",])
```

Pomoćnim skupovima dodajemo i parametar "TotalStays" koji predstavlja sumu vrijednosti parametara "StaysInWeekendNights" i "StaysInWeekNights" kako bismo dobili uvid o trajanju boravka gosta neovisno o danima u tjednu.

```
library(magrittr)  
library(dplyr)  
  
##  
## Attaching package: 'dplyr'  
  
## The following objects are masked from 'package:stats':  
##  
##     filter, lag  
  
## The following objects are masked from 'package:base':  
##  
##     intersect, setdiff, setequal, union
```

```

summer_total <- summer %>%
  mutate(TotalStays = StaysInWeekendNights + StaysInWeekNights)

autumn_total <- autumn %>%
  mutate(TotalStays = StaysInWeekendNights + StaysInWeekNights)

winter_total <- winter %>%
  mutate(TotalStays = StaysInWeekendNights + StaysInWeekNights)

spring_total <- spring %>%
  mutate(TotalStays = StaysInWeekendNights + StaysInWeekNights)

```

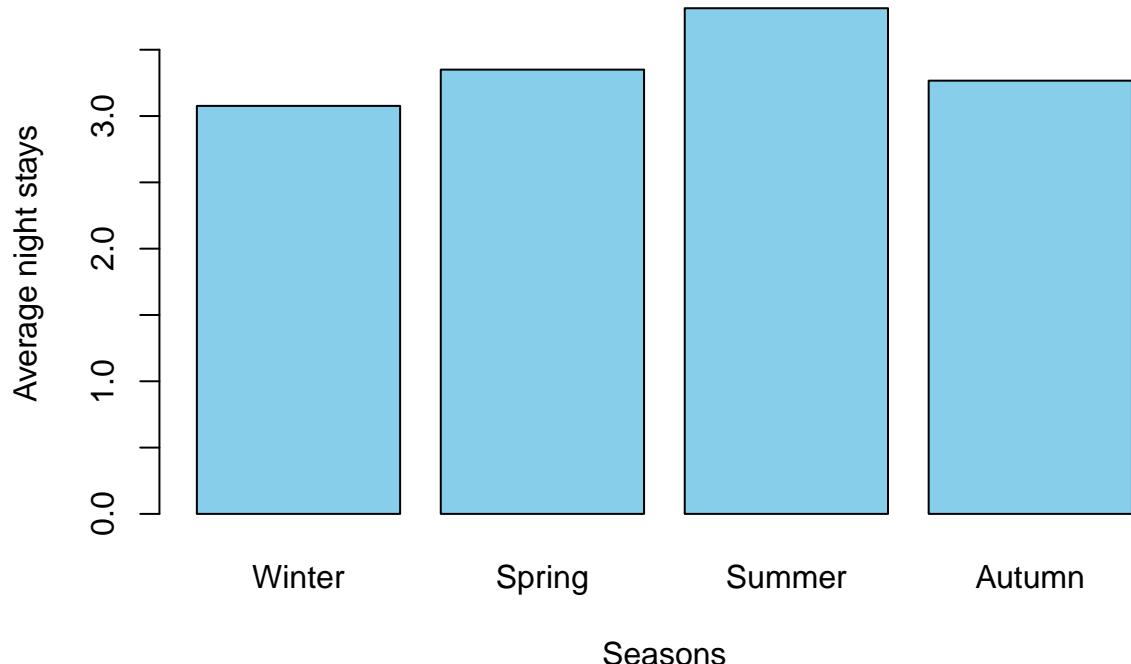
Računamo prosječnu duljinu trajanja boravka za svaki pomoći skup podataka, rezultat prikazujemo stupičastim dijagramom.

```

seasons_means_vector = data.frame(
  Season = c("Winter", "Spring", "Summer", "Autumn"),
  Means = c(mean(winter_total$TotalStays),
            mean(spring_total$TotalStays),
            mean(summer_total$TotalStays),
            mean(autumn_total$TotalStays))
)

barplot(seasons_means_vector$Means, names.arg = seasons_means_vector$Season, col = "skyblue", xlab = "Se

```



```
#binding adjusted data
data_total_stays = rbind(winter_total, spring_total, summer_total, autumn_total)
```

Iz histograma možemo zaključiti da je prosječni boravak gostiju najduži ljeti, a najkraći zimi.

Trajanje prosječnog boravka i tip iznajmljene sobe

Naše istraživačko pitanje je slijedeće:

Možemo li trajanje boravka povezati s tipom iznajmljene sobe?

Pogledajmo najprije koje sve kategorije soba za iznajmljivanje postoje.

```
#all different values of "ReservedRoomType" variable
room_types = unique(data_total_stays$AssignedRoomType)
room_types
```

```
## [1] "I"          "A"          "E"          "D"          "
## [5] "F"          "C"          "H"          "G"          "
## [9] "B"          "P"          "K"          "L"          "
```

Iz priloženog možemo vidjeti da postoji 12 različitih kategorija soba, prema tome, uzimamo 12 podskupova podataka i uspoređujemo njihove aritmetičke sredine duljine trajanja boravka (varijabla "TotayStays").

Za dati odgovor na istraživačko pitanje koristit ćemo se jednofaktorskim ANOVA modelom.

Predpostavke ANOVE su: - nezavisnost pojedinih podataka u uzorcima - normalna razdioba podataka - homogenost varijanci među populacijama

Kako nam veličine grupe nisu podjednake, radimo provjeru normalnosti Lillieforsovom inačicom Kolmogorov-Smirnovljenog testa. Razmatrat ćemo tip sobe kao varijablu koja određuje grupe i ukupnu duljinu boravka kao zavisnu varijablu.

```
require(nortest)

## Loading required package: nortest

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "I", ]$TotalStays)

##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "I", ]$TotalStays
## D = 0.26446, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "A", ]$TotalStays)

##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "A", ]$TotalStays
## D = 0.21952, p-value < 2.2e-16
```

```

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "E", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "E", ]$TotalStays
## D = 0.17111, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "D", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "D", ]$TotalStays
## D = 0.19488, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "F", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "F", ]$TotalStays
## D = 0.18765, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "C", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "C", ]$TotalStays
## D = 0.162, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "H", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "H", ]$TotalStays
## D = 0.15848, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "G", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "G", ]$TotalStays
## D = 0.16133, p-value < 2.2e-16

```

```

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "P", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "P", ]$TotalStays
## D = 0.53025, p-value = 1.337e-10

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "B", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "B", ]$TotalStays
## D = 0.18077, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$AssignedRoomType == "K", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$AssignedRoomType == "K", ]$TotalStays
## D = 0.30595, p-value < 2.2e-16

```

Iznimno malene p-vrijednosti nam govore kako podaci nisu normalno distribuirani. Usprkos tome, jednofaktorska ANOVA je robusna statistika s obzirom na predpostavku normalnosti. Dobro tolerira odstupanja od normalne distribucije.

Što se tiče homogenosti varijanci različitih populacija, potrebno je testirati:

$$H_0 : \sigma_1^2 = \sigma_2^2 = \dots = \sigma_k^2$$

$$H_1 : \text{barem dvije varijance nisu iste.}$$

Navedenu hipotezu možemo testirati Bartlettovim testom.

```

data_total_stays$AssignedRoomType <- factor(data_total_stays$AssignedRoomType)

# Check the number of observations in each group
group_counts <- table(data_total_stays$AssignedRoomType)
valid_groups <- names(group_counts[group_counts >= 2])

#If there is at least two oservations in the group, proceed
filtered_data <- subset(data_total_stays, AssignedRoomType %in% valid_groups)

#Bartlettovim test
bartlett.test(filtered_data$TotalStays ~ filtered_data$AssignedRoomType)

```

```

##
## Bartlett test of homogeneity of variances
##
## data: filtered_data$TotalStays by filtered_data$AssignedRoomType
## Bartlett's K-squared = 6714.5, df = 10, p-value < 2.2e-16

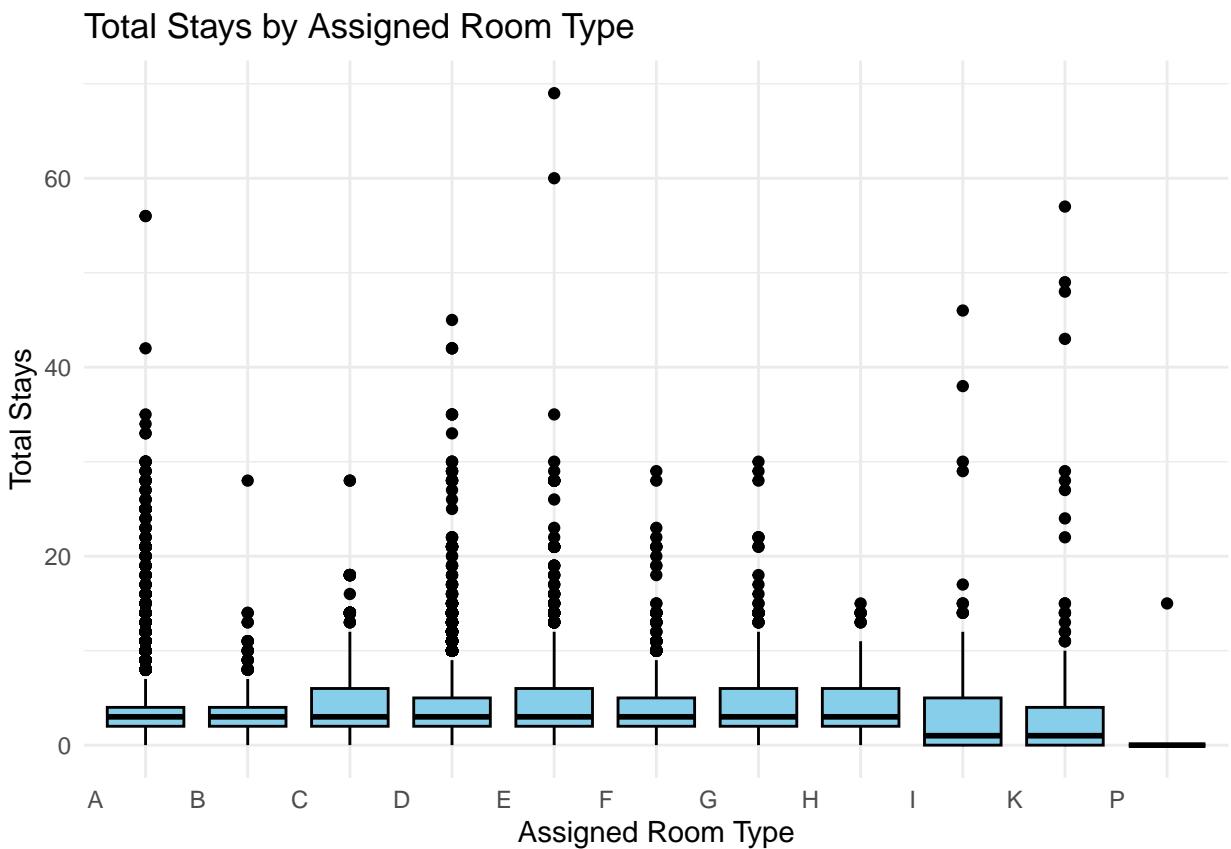
```

Varijance varijable "TotalStays" nisu podjednake, što je grafički prikazano u nastavku.

```
#box and whisker plot --> mean, rangs, median, variances....
library(ggplot2)

# Assuming filtered_data$AssignedRoomType is a factor
filtered_data$AssignedRoomType <- factor(filtered_data$AssignedRoomType)

# Create a boxplot using ggplot2
ggplot(filtered_data, aes(x = AssignedRoomType, y = TotalStays)) +
  geom_boxplot(fill = "skyblue", color = "black") +
  labs(title = "Total Stays by Assigned Room Type",
       x = "Assigned Room Type",
       y = "Total Stays") +
  theme_minimal()
```



Dani podaci o distibuciji koja nije normalna i nehomogenosti varijanci navodi nas na zaključak da prosječna duljina boravka varira ovisno o tipu iznajmljene sobe, a to ćemo sada testirati.

Naša testne hipoteze biti će:

H_0 : prosječna duljina boravka jednaka je za sve tipove iznajmljenih soba

H_1 : za barem jedan tip iznajmljene sobe prosječna duljina boravka različita je od ostalih

```
#ANOVA
a = aov(filtered_data$TotalStays ~ filtered_data$AssignedRoomType)
summary(a)
```

```

##                               Df Sum Sq Mean Sq F value Pr(>F)
## filtered_data$AssignedRoomType      10 14597 1459.7 227.4 <2e-16 ***
## Residuals                         119378 766260      6.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Zaključak:

Na temelju malene p-vrijednosti odbacujemo nullu hipotezu u korist alternativne hipoteze: **postoje razlike u duljini trajanja boravka ovisno o tipu iznajmljene sobe.**

Trajanje prosječnog boravka i kategorija gosta

Naše iduće istraživačko pitanje je slijedeće:

Možemo li trajanje boravka povezati s kategorijom gosta?

Procedura zaključivanja i odgovaranja na ovo istraživačko pitanje ista je prethodnom. Pogledajmo koje sve kategorije gostiju postoje.

```

#all different values of "CustomerType" variable
customer_types = unique(data_total_stays$CustomerType)
customer_types

```

```

## [1] "Transient"      "Contract"       "Transient-Party" "Group"

```

Iz priloženog možemo vidjeti da postoji 4 različitih kategorija gostiju, prema tome, uzimamo 4 podskupova podataka i uspoređujemo njihove aritmetičke sredine duljine trajanja boravka (varijabla "TotayStays").

Za dati odgovor na istraživačko pitanje koristit ćemo se jednofaktorskim ANOVA modelom.

Provjeravamo jesu li zadovljene predpostavke ANOVE (za provjeru predpostavke normalnosti koristimo se Lillieforsovom inačicom KS testa, a za provjeru predpostavke homogensoti varijanca koristimo se Bartlettovim testom).

```

#normality check
require(nortest)

lillie.test(data_total_stays[data_total_stays$CustomerType == "Transient", ]$TotalStays)

##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$CustomerType == "Transient", ]$TotalStays
## D = 0.19487, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$CustomerType == "Contract", ]$TotalStays)

##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$CustomerType == "Contract", ]$TotalStays
## D = 0.17507, p-value < 2.2e-16

```

```

lillie.test(data_total_stays[data_total_stays$CustomerType == "Transient-Party", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$CustomerType == "Transient-Party", ]$TotalStays
## D = 0.23459, p-value < 2.2e-16

lillie.test(data_total_stays[data_total_stays$CustomerType == "Group", ]$TotalStays)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data_total_stays[data_total_stays$CustomerType == "Group", ]$TotalStays
## D = 0.21935, p-value < 2.2e-16

```

Iznimno malene p-vrijednosti nam govore kako podaci nisu normalno distribuirani. Usprkos tome, jednofaktorska ANOVA je robusna statistika s obzirom na predpostavku normalnosti.

```

# variance homogeneity check
data_total_stays$CustomerType <- factor(data_total_stays$CustomerType)

# Check the number of observations in each group
group_counts <- table(data_total_stays$CustomerType)
valid_groups <- names(group_counts[group_counts >= 2])

#If there is at least two oservations in the group, proceed
filtered_data <- subset(data_total_stays, CustomerType %in% valid_groups)

#Bartlettovim test
bartlett.test(filtered_data$TotalStays ~ filtered_data$CustomerType)

```

```

##
## Bartlett test of homogeneity of variances
##
## data: filtered_data$TotalStays by filtered_data$CustomerType
## Bartlett's K-squared = 3615, df = 3, p-value < 2.2e-16

```

Zbog malene p-vrijednosti odbacujemo homogenost varijanci, što prikazujemo i grafički.

```

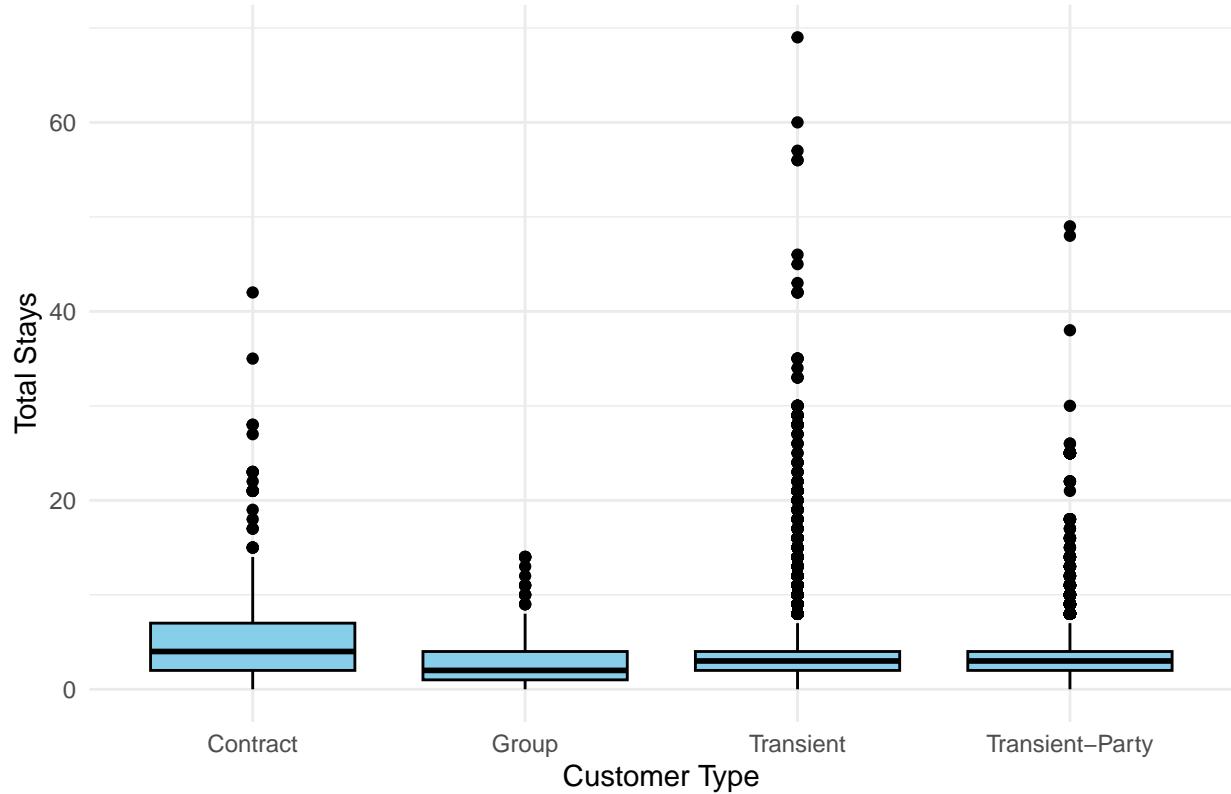
#box and whisker plot --> mean, rangs, median, variances....
library(ggplot2)

# Assuming filtered_data$CustomerType is a factor
filtered_data$CustomerType <- factor(filtered_data$CustomerType)

# Create a boxplot using ggplot2
ggplot(filtered_data, aes(x = CustomerType, y = TotalStays)) +
  geom_boxplot(fill = "skyblue", color = "black") +
  labs(title = "Total Stays by Customer Type",
       x = "Customer Type",
       y = "Total Stays") +
  theme_minimal()

```

Total Stays by Customer Type



Dani podaci o distibuciji koja nije normalna i nehomogenosti varijanci navodi nas na zaključak da prosječna duljina boravka varira ovisno o tipu gosta, a to ćemo sada i testirati jednoparametarskom ANOVOM.

Naša testne hipoteze biti će:

$$H_0 : \text{prosječna duljina boravka jednaka je za sve tipove gostiju}$$

$$H_1 : \text{za barem jedan tip gosta prosječna duljina boravka različita je no u ostalih}$$

```
#ANOVA
a = aov(filtered_data$TotalStays ~ filtered_data$CustomerType)
summary(a)
```

```
##                                     Df Sum Sq Mean Sq F value Pr(>F)
## filtered_data$CustomerType      3 18121   6040   945.5 <2e-16 ***
## Residuals                      119386 762742       6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Zaključak:

Na temelju malene p-vrijednosti odbacujemo nultu hipotezu u korist alternativne hipoteze: **postoje razlike u duljini trajanja boravka ovisno o tipu gosta.**

Diskusija:

Kao što smo i mogli naslutiti, duljina boravka gosta varira s obzirom na različite parametre. Najdulji prosječni boravak jest ljeti, a duljina boravka varira i zavisno od tipa iznajmljene sobe te kategorije gosta.

Case study: CIJENE

Case study: PANSION, POLUPANSION

“Je li vrsta uplaćenog obroka (pansion, polupansion) povezana s kategorijama kupaca?”

Opis traženih atributa

Vrstu uplaćenog obroka vidimo pod “Meal”

```
meals <- data$Meal  
levels(factor(data$Meal))  
  
## [1] "BB"      "FB"      "HB"      "SC"      "Undefined"  
  
table(meals)  
  
## meals  
## BB FB HB SC Undefined  
## 92310 798 14463 10650 1169
```

Značenja kratica:

BB -> Samo doručak FB -> Polupansion HB -> Puni pansion SC -> Bez hotelskih obroka (“Self-Catering”)
Undefined -> Nije definirano

Vrstu kategorija kupaca vidimo pod “CustomerType”

```
types <- data$CustomerType  
levels(factor(data$CustomerType))  
  
## [1] "Contract"      "Group"       "Transient"     "Transient-Party"  
  
table(types)  
  
## types  
## Contract      Group       Transient    Transient-Party  
##        4076        577        89613       25124
```

Contract -> primjeri: poslovna putovanja, rezervacije za konferencije ili događaje

Group -> primjeri: turističke grupe, konferencijski timovi, sportski timovi

Transient -> primjeri: individualni putnici, turisti koji putuju sami

Transient-Party -> primjeri: Grupe prijatelja ili obitelji koje putuju zajedno, ali nisu rezervirale kao formalna grupa

Je li vrsta uplaćenog obroka (pansion, polupansion) povezana s kategorijama kupaca?

Kopirajmo najprije podatke u novi data.frame kako ne bi promijenili prave vrijednosti

```
data_copy <- data
```

Kontingencijsku tablicu jedne kategoriske varijable moguće je dobiti pozivanjem funkcije `table()`

```
tbl = table(data_copy$Meal)
print(tbl)
```

```
##
##   BB      FB      HB      SC      Undefined
##   92310    798    14463   10650    1169
```

Pogledajmo kontingencijsku tablicu varijabli obroka i kategorija kupaca

```
tbl = table(data_copy$Meal, data_copy$CustomerType)
tbl
```

```
##
##                               Contract  Group  Transient  Transient-Party
##   BB           3260    499     70692          17859
##   FB             5       1      547            245
##   HB            613    36      8020          5794
##   SC            183    39      9968          460
##   Undefined      15      2      386            766
```

Kontingencijskoj tablici možemo dodati i sume redaka i stupaca na sljedeći način.

```
added_margins_tbl = addmargins(tbl)
print(added_margins_tbl)
```

```
##
##                               Contract  Group  Transient  Transient-Party      Sum
##   BB           3260    499     70692          17859    92310
##   FB             5       1      547            245      798
##   HB            613    36      8020          5794    14463
##   SC            183    39      9968          460    10650
##   Undefined      15      2      386            766    1169
##   Sum           4076    577     89613         25124  119390
```

Test nezavisnosti χ^2 test u programskom paketu R implementiran je u funkciji `chisq.test()` koja kao ulaz prima kontingencijsku tablicu podataka koje testiramo na nezavisnost. Ispitajmo nezavisnost pozicije igrača na terenu i njegove preferirane noge za udarce.

Pretpostavka testa je da očekivana frekvencija pojedinog razreda mora biti veća ili jednaka 5 (`chisq.test()` pretpostavlja da je ovaj uvjet zadovoljen stoga je prije provođenja testa potrebno to provjeriti).

```
for (col_names in colnames(added_margins_tbl)){
  for (row_names in rownames(added_margins_tbl)){
    if (!(row_names == 'Sum' | col_names == 'Sum') ){
      cat('Očekivane frekvencije za razred ', col_names, '-', row_names, ': ', (added_margins_tbl[row_names,
      ]))
    }
  }
}
```

```

## Očekivane frekvencije za razred Contract - BB : 3151.483
## Očekivane frekvencije za razred Contract - FB : 27.24389
## Očekivane frekvencije za razred Contract - HB : 493.7699
## Očekivane frekvencije za razred Contract - SC : 363.5933
## Očekivane frekvencije za razred Contract - Undefined : 39.90991
## Očekivane frekvencije za razred Group - BB : 446.1251
## Očekivane frekvencije za razred Group - FB : 3.856655
## Očekivane frekvencije za razred Group - HB : 69.89824
## Očekivane frekvencije za razred Group - SC : 51.47039
## Očekivane frekvencije za razred Group - Undefined : 5.649661
## Očekivane frekvencije za razred Transient - BB : 69287.01
## Očekivane frekvencije za razred Transient - FB : 598.9712
## Očekivane frekvencije za razred Transient - HB : 10855.79
## Očekivane frekvencije za razred Transient - SC : 7993.789
## Očekivane frekvencije za razred Transient - Undefined : 877.4403
## Očekivane frekvencije za razred Transient-Party - BB : 19425.38
## Očekivane frekvencije za razred Transient-Party - FB : 167.9282
## Očekivane frekvencije za razred Transient-Party - HB : 3043.541
## Očekivane frekvencije za razred Transient-Party - SC : 2241.147
## Očekivane frekvencije za razred Transient-Party - Undefined : 246.0001

```

Vidimo mogući problem: "Očekivane frekvencije za razred Group - FB : 3.856655"

χ^2 test zahtjeva frekvencije veće od 5, pa ćemo trebati izostaviti ovaj razred na neki način. FB sadrži malen broj podataka, tako da bi mogao biti izostavljen potpuno.

Također smatramo da bi bilo dobro izostaviti varijable "Undefined" jer predstavljaju nepoznanicu koju je teško interpretirati.

Odluka je maknuti Undefined i FB iz kategorije Meals.

```
cat("Kontingencijska tablica prije: \n")
```

```
## Kontingencijska tablica prije:
```

```
table(data_copy$Meal)
```

```
##
## BB FB HB SC Undefined
## 92310 798 14463 10650 1169
```

```
cat("\n-----\n")
```

```
##
## -----
```

```
data_modified <- data_copy[trimws(data_copy$Meal) != "Undefined" & trimws(data_copy$Meal) != "FB", ]
cat("Kontingencijska tablica poslije: \n")
```

```
## Kontingencijska tablica poslije:
```

```
table(data_modified$Meal)
```

```
##  
## BB           HB           SC  
##    92310      14463      10650
```

Ponovimo prethodni postupak na data_modified

```
tbl <- table(data_modified$Meal, data_modified$CustomerType)  
tbl
```

```
##  
##          Contract Group Transient Transient-Party  
##    BB        3260   499     70692         17859  
##    HB        613    36      8020          5794  
##    SC        183    39      9968          460
```

```
added_margins_tbl = addmargins(tbl)  
added_margins_tbl
```

```
##  
##          Contract Group Transient Transient-Party      Sum  
##    BB        3260   499     70692       17859  92310  
##    HB        613    36      8020        5794  14463  
##    SC        183    39      9968        460   10650  
##    Sum       4056   574     88680      24113 117423
```

```
for (col_names in colnames(added_margins_tbl)){  
  for (row_names in rownames(added_margins_tbl)){  
    if (!(row_names == 'Sum' | col_names == 'Sum')){  
      cat('Očekivane frekvencije za razred ', col_names, '-', row_names, ': ', (added_margins_tbl[row_names, ]))  
    }  
  }  
}
```

```
## Očekivane frekvencije za razred Contract - BB      : 3188.552  
## Očekivane frekvencije za razred Contract - HB      : 499.5778  
## Očekivane frekvencije za razred Contract - SC      : 367.87  
## Očekivane frekvencije za razred Group - BB        : 451.2399  
## Očekivane frekvencije za razred Group - HB        : 70.69962  
## Očekivane frekvencije za razred Group - SC        : 52.0605  
## Očekivane frekvencije za razred Transient - BB     : 69714.2  
## Očekivane frekvencije za razred Transient - HB     : 10922.72  
## Očekivane frekvencije za razred Transient - SC     : 8043.075  
## Očekivane frekvencije za razred Transient-Party - BB : 18956.01  
## Očekivane frekvencije za razred Transient-Party - HB : 2970  
## Očekivane frekvencije za razred Transient-Party - SC : 2186.994
```

Sada vidimo da su sve frekvencije zadovoljavajuće, te nemamo nepoznanica. Možemo nastaviti sa χ^2 testom.

```
chisq.test(tbl,correct=F)

##
##  Pearson's Chi-squared test
##
## data:  tbl
## X-squared = 5503.8, df = 6, p-value < 2.2e-16
```

Na temelju vrlo male P-vrijednosti(p-value < 2.2e-16), odbacujemo H₀ u korist H₁ koja kaže da su vrsta obroka i kategorija gosta zavisni podaci.

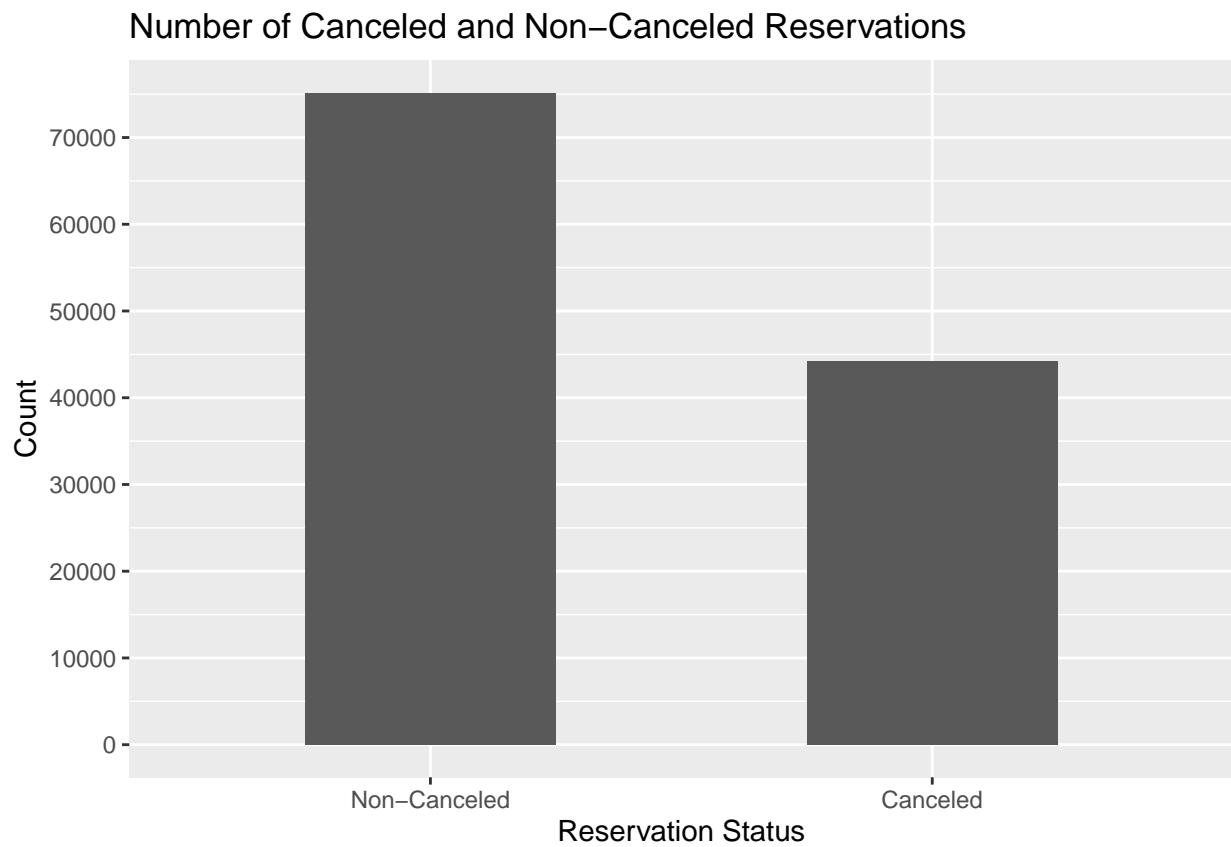
Case study: OTKAZIVANJE REZERVACIJA

Prikaz omjera otkazanih rezervacija

```
library(ggplot2)

total_count <- nrow(data)
canceled_count <- sum(data$IsCanceled == 1)
non_canceled_count <- total_count - canceled_count

ggplot(data, aes(x = factor(IsCanceled))) +
  geom_bar(width = 0.5) +
  labs(x = "Reservation Status", y = "Count", title = "Number of Canceled and Non-Canceled Reservations")
  scale_x_discrete(labels = c("Non-Canceled", "Canceled")) +
  scale_y_continuous(breaks = seq(0, max(total_count), by = 10000))
```



Kroz ovaj prikaz možemo vidjeti ukupan broj otkazanih i ne-otkazanih rezervacija za priložne podatke.

Otkazanih rezervacija je bilo: 44224 od ukupno 119390. Ovo znači da je 37.04% ukupnih rezervacija otkazano. Zanima nas ima li korelacije nekih parametara iz naših podataka sa otkazivanjem rezervacija.

Koristimo regresijske metode s obzirom da nam regresijska analiza omogućuje zaključivanje i predviđanje jedne varijable na temelju jedne ili više varijabli, te mjerjenje utjecaja jedne ili više varijabli s obzirom na ostale.

U našem slučaju ovisna varijabla će biti IsCanceled varijabla koja nam govori jesu li rezervacije otazane. Pre-gledom liste svih varijabli odabrali smo varijable za koje smatramo da će imati najveći učinak na otkazivanje

te će nam one činit neovisne varijable. Odabранe varijable su:

- LeadTime (Vrijeme od trenutka rezervacije do rezervacije)
- Previous cancelations
- Previous booking not canceled

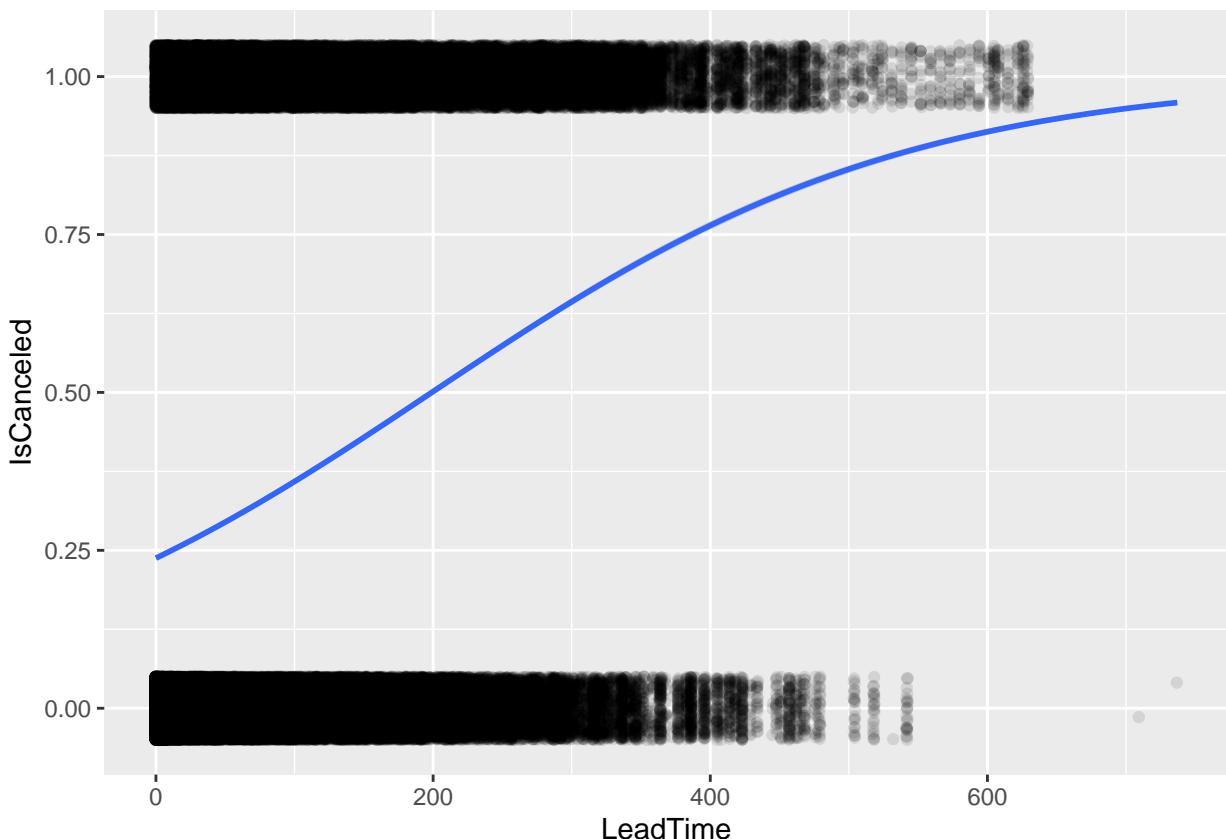
Lead Time

Prvo pitanje koje postavljamo je ima li broj LeadTime utjecaja na otkazivanje rezervacije?

```
library(ggplot2)

ggplot(data, aes(x = LeadTime, y = IsCanceled )) +
  geom_jitter(height = .05, alpha = .1) +
  geom_smooth(method = "glm", method.args = list(family = "binomial"), se = TRUE)

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



Iz prikaza modela se da naslutiti da ima. Možemo primjetiti da veće vrijednosti varijable LeadTime imaju više otkazivanja. Koristimo funkciju za logističku regresiju u programu R kako bi detaljnije analizirali:

```
logisticLT <- glm(formula = IsCanceled ~ LeadTime, family = "binomial", data = data)
summary(logisticLT)
```

```

## 
## Call:
## glm(formula = IsCanceled ~ LeadTime, family = "binomial", data = data)
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.166e+00  9.182e-03 -126.9   <2e-16 ***
## LeadTime     5.855e-03  6.137e-05    95.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
## Null deviance: 157398  on 119389  degrees of freedom
## Residual deviance: 147158  on 119388  degrees of freedom
## AIC: 147162
## 
## Number of Fisher Scoring iterations: 4

```

Objašnjenje dobivenog:

Estimate

Ovo predstavlja veličinu učinka svakog prediktora.

Standardna pogreška

Mjerena standardna devijacija procijenjenog koeficijenta koja odražava koliko precizno model procjenjuje vrijednost koeficijenta.

z-vrijednost

Izračunat kao koeficijent podijeljen sa svojom standardnom pogreškom. Koristi se za testiranje nulte hipoteze.

p-vrijednost

P-vrijednost označava vjerojatnost opažanja podataka ako je nulta hipoteza istinita. Niža p-vrijednost sugerira da je nulta hipoteza manje vjerojatna.

Ovime vidimo da model predviđa da će logaritamski izgledi osnovnog ishoda biti -1.166e+00 kada su svi prediktori 0.

P-vrijednost od <2e-16 ukazuje da LeadTime utječe na otkazivanje rezervacija na statističkoj razini od 0.05 (5%) i 0.01 (1%)

Predikcije možemo dobiti od prediktora uvrštavanjme u formulu:

$$y = \frac{1}{1 + e^{-(\alpha_1 + \alpha_2 x)}}$$

gdje je alpha_1 prediktor ovisne varijable a alpha_2 prediktor neovisne varijable. Ovo je u našem slučaju:

$$y = \frac{1}{1 + e^{-(1.116 + 0.005855x)}}$$

gdje je x broj dana prošao od rezervacije.

Ovo također možemo dobiti i kroz program R na sljedeći način:

```

# Napravimo novi podatkovni okvir s variablama predviđanja za koje želimo predvidjeti
new_data <- data.frame(LeadTime = c(0, 100, 200, 400))
# Koristimo funkciju za predviđanje
predicted_probabilities <- predict(logisticLT, newdata = new_data, type = "response")
# Prikazujemo rezultate
print(predicted_probabilities)

```

```

##          1         2         3         4
## 0.2376550 0.3589143 0.5013551 0.7643035

```

Samim time možemo zaključiti da LeadTime utječe na otkazivanje rezervacija. Odnosno rezervacije **rezervirane za daleko u budućnost imaju veću šansu za otkazivanje**.

Previous Cancellations

Nastavljamo sa promatranjem varijable Previous Cancellations. Ponavljamo postupke iz prošle analize.

```

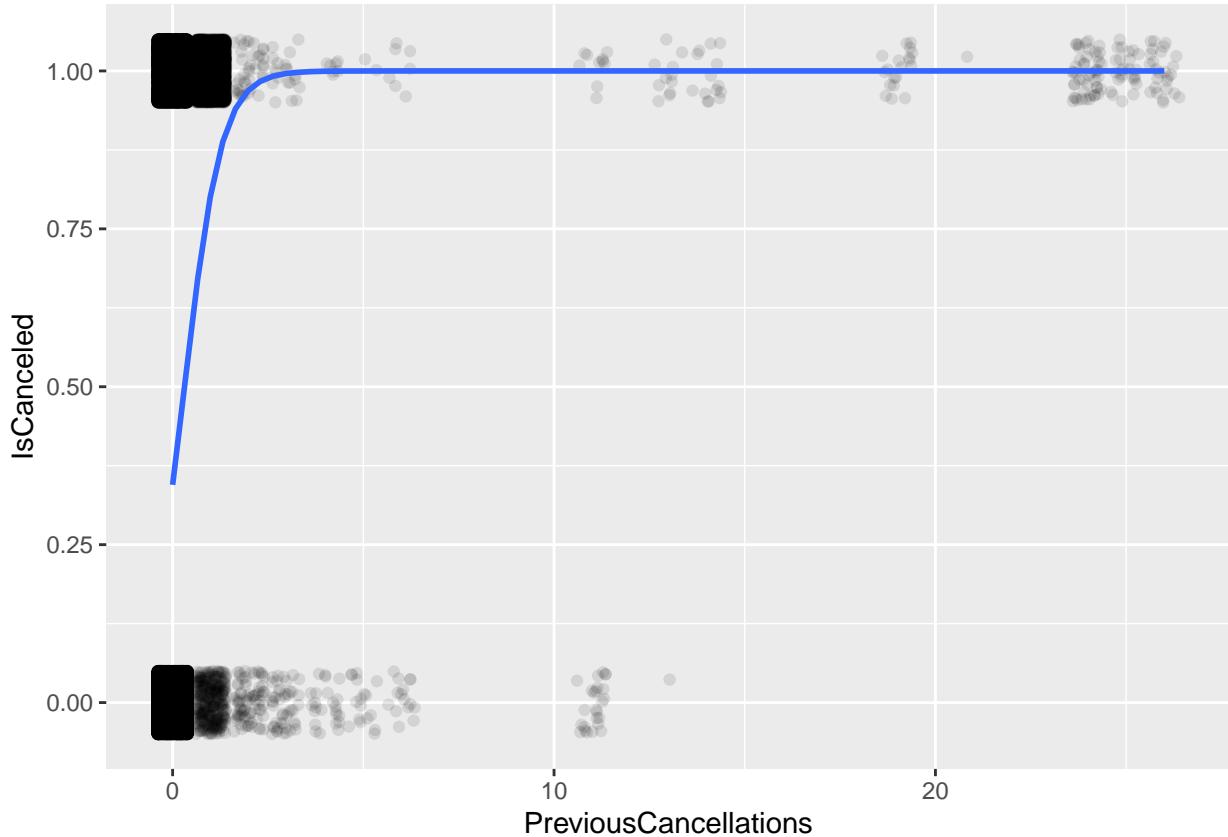
library(ggplot2)

ggplot(data, aes(x = PreviousCancellations, y = IsCanceled )) +
  geom_jitter(height = .05, alpha = .1) +
  geom_smooth(method = "glm", method.args = list(family = "binomial"), se = TRUE)

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

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

```



Na modelu se da primjetiti da prijašnja otkazivanja rezervacija imaju utjecaj na otkazivanje rezervacija.

```
logisticPC <- glm(formula = IsCancelled ~ PreviousCancellations, family = "binomial", data = data)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(logisticPC)

##
## Call:
## glm(formula = IsCancelled ~ PreviousCancellations, family = "binomial",
##      data = data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.642251  0.006261 -102.58  <2e-16 ***
## PreviousCancellations 2.060846  0.032777   62.88  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 157398  on 119389  degrees of freedom
## Residual deviance: 151456  on 119388  degrees of freedom
## AIC: 151460
```

```
##  
## Number of Fisher Scoring iterations: 6
```

Ovo nam potvrđuju i dobiveni podatci. Model predviđa da će logaritamski izgledi osnovnog ishoda biti -0.642251 kada su svi prediktori 0.

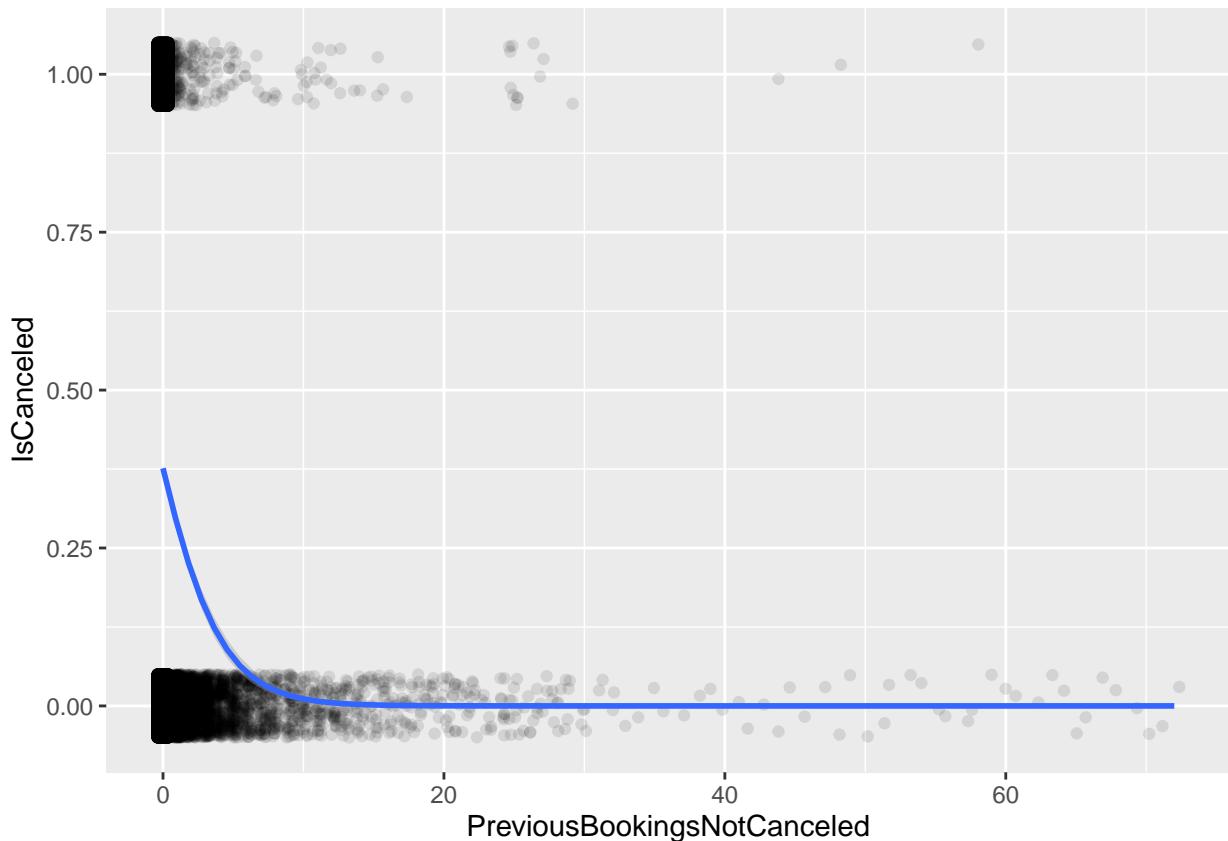
P-vrijednost od <2e-16 ukazuje da PreviousCancellations utječe na otkazivanje rezervacija na statističkoj razini od 0.05 (5%) i 0.01 (1%).

Odnosno sa brojem prijašnjih otkazanih rezervacija raste šansa za otkazivanjem budućih rezervacija.

Previous bookings not canceled

Nastavljamo sa promatranjem varijable Previous bookings not canceled. Ponavljamo postupke iz prošlih analiza.

```
library(ggplot2)  
  
ggplot(data, aes(x = PreviousBookingsNotCanceled, y = IsCanceled )) +  
  geom_jitter(height = .05, alpha = .1) +  
  geom_smooth(method = "glm", method.args = list(family = "binomial"), se = TRUE)  
  
## 'geom_smooth()' using formula = 'y ~ x'
```



Kao i kod prijašnjih modela i na ovom modelu možemo primjetiti korelaciju podataka.

```

logisticPC <- glm(formula = IsCanceled ~ PreviousBookingsNotCanceled, family = "binomial", data = data)
summary(logisticPC)

##
## Call:
## glm(formula = IsCanceled ~ PreviousBookingsNotCanceled, family = "binomial",
##      data = data)
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -0.505722  0.006046 -83.65  <2e-16 ***
## PreviousBookingsNotCanceled -0.398467  0.020170 -19.76  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 157398  on 119389  degrees of freedom
## Residual deviance: 156431  on 119388  degrees of freedom
## AIC: 156435
##
## Number of Fisher Scoring iterations: 6

```

Ovo nam opet potvrđuju dobiveni podatci. Model predviđa da će logaritamski izgledi osnovnog ishoda biti -0.505722 kada su svi prediktori 0.

P-vrijednost od <2e-16 ukazuje da PreviousBookingsNotCanceled utječe na otkazivanje rezervacija na statističkoj razini od 0.05 (5%) i 0.01 (1%).

Što daje naslutiti da je šansa za otkazivanje manja u ljudi koji su imali prijašnje ne otkazane rezervacije.

Zaključak:

Model logističke regresije snažan je alat koji nam daje uvid u ovisnost komponenti te mogućnost predikcije ukoliko postoji neka korelacija.

Ovo nam u konkretnom slučaju koristi na način da putem predikcija možemo naslutiti na moguća otkazivanja u slučaju riskantnih rezervacija te poduzimanja koraka koji bi mogli smanjiti taj broj.

Npr. u slučaju rezervacija s velikim Lead time varijablama možemo “overbookirat” te termine ili ih ne “overbokirat” u slučaju velikih “previous bookings not canceled” varijabli. Također bi se moglo uvesti naknade za otkazivanje za ljude sa velikim brojem prijašnjih otkazivanja s obzirom da je rizik za otkazivanje kod njih veći.

Case study: SKUPINE GOSTIJU IZ RAZLIČITIH ZEMALJA

ADR (Avrage daily rate)

Zanima nas možemo li povezati dnevnu potrošnju sa porjeklom zemlje gosta.

Prvo pregledavamo iz kojih sve gosti zemalja uopće dolaze.

```
allCountries = unique(data$Country)
allCountries
```

```
## [1] "PRT"  "GBR"  "USA"   "ESP"   "IRL"   "FRA"   "NULL"  "ROU"   "NOR"   "OMN"
## [11] "ARG"   "POL"  "DEU"   "BEL"   "CHE"   "CN"    "GRC"   "ITA"   "NLD"   "DNK"
## [21] "RUS"   "SWE"  "AUS"   "EST"   "CZE"   "BRA"   "FIN"   "MOZ"   "BWA"   "LUX"
## [31] "SVN"   "ALB"  "IND"   "CHN"   "MEX"   "MAR"   "UKR"   "SMR"   "LVA"   "PRI"
## [41] "SRB"   "CHL"  "AUT"   "BLR"   "LTU"   "TUR"   "ZAF"   "AGO"   "ISR"   "CYM"
## [51] "ZMB"   "CPV"  "ZWE"   "DZA"   "KOR"   "CRI"   "HUN"   "ARE"   "TUN"   "JAM"
## [61] "HRV"   "HKG"  "IRN"   "GEO"   "AND"   "GIB"   "URY"   "JEY"   "CAF"   "CYP"
## [71] "COL"   "GGY"  "KWT"   "NGA"   "MDV"   "VEN"   "SVK"   "FJI"   "KAZ"   "PAK"
## [81] "IDN"   "LBN"  "PHL"   "SEN"   "SYC"   "AZE"   "BHR"   "NZL"   "THA"   "DOM"
## [91] "MKD"   "MYS"  "ARM"   "JPN"   "LKA"   "CUB"   "CMR"   "BIH"   "MUS"   "COM"
## [101] "SUR"  "UGA"  "BGR"   "CIV"   "JOR"   "SYR"   "SGP"   "BDI"   "SAU"   "VNM"
## [111] "PLW"  "QAT"  "EGY"   "PER"   "MLT"   "MWI"   "ECU"   "MDG"   "ISL"   "UZB"
## [121] "NPL"  "BHS"  "MAC"   "TGO"   "TWN"   "DJI"   "STP"   "KNA"   "ETH"   "IRQ"
## [131] "HND"  "RWA"  "KHM"   "MCO"   "BGD"   "IMN"   "TJK"   "NIC"   "BEN"   "VGB"
## [141] "TZA"  "GAB"  "GHA"   "TMP"   "GLP"   "KEN"   "LIE"   "GNB"   "MNE"   "UMI"
## [151] "MYT"  "FRO"  "MMR"   "PAN"   "BFA"   "LBY"   "MLI"   "NAM"   "BOL"   "PRY"
## [161] "BRB"  "ABW"  "AIA"   "SLV"   "DMA"   "PYF"   "GUY"   "LCA"   "ATA"   "GTM"
## [171] "ASM"  "MRT"  "NCL"   "KIR"   "SDN"   "ATF"   "SLE"   "LAO"
```

Primjećujemo da su gosti iz 178 zemalja te uspoređujemo njihove aritmetičke sredine dnevne potrošnje (varijabla "ADR").

Kako bi vidjeli možemo li povezati dnevnu potrošnju sa porjeklom zemlje, koristit ćemo se jednofaktorskim ANOVA modelom.

Predpostavke ANOVE su u ovom projektu već navedene u analizi koja se bavila sa trajanjima boravka gostiju.

Kako nam veličine grupe nisu podjednake, radimo provjeru normalnosti Lillieforsovom inačicom Kolmogorov-Smirnovljenog testa. Ovaj test zahtjeva minimalnu veličinu uzorka od 4, stoga transformiramo podatke da dobijemo samo države koje iz kojih su došle 4 ili više osoba.

```
data$Country <- factor(data$Country)
groupCount <- table(data$Country)
validCountries <- names(groupCount[groupCount > 4])
validCountries
```

```
## [1] "AGO"  "ALB"  "AND"  "ARE"  "ARG"  "ARM"  "AUS"  "AUT"  "AZE"  "BEL"
## [11] "BGD"  "BGR"  "BHR"  "BIH"  "BLR"  "BOL"  "BRA"  "CAF"  "CHE"  "CHL"
## [21] "CHN"  "CIV"  "CMR"  "CN"   "COL"  "CPV"  "CRI"  "CUB"  "CYP"  "CZE"
## [31] "DEU"  "DNK"  "DOM"  "DZA"  "ECU"  "EGY"  "ESP"  "EST"  "FIN"  "FRA"
## [41] "FRO"  "GBR"  "GEO"  "GIB"  "GNB"  "GRC"  "HKG"  "HRV"  "HUN"  "IDN"
## [51] "IND"  "IRL"  "IRN"  "IRQ"  "ISL"  "ISR"  "ITA"  "JAM"  "JEY"  "JOR"
## [61] "JPN"  "KAZ"  "KEN"  "KOR"  "KWT"  "LBN"  "LBY"  "LKA"  "LTU"  "LUX"
```

```

## [71] "LVA"  "MAC"  "MAR"  "MDV"  "MEX"  "MKD"  "MLT"  "MNE"  "MOZ"  "MUS"
## [81] "MYS"  "NGA"  "NLD"  "NOR"  "NULL" "NZL"  "OMN"  "PAK"  "PAN"  "PER"
## [91] "PHL"  "POL"  "PRI"  "PRT"  "QAT"  "ROU"  "RUS"  "SAU"  "SEN"  "SGP"
## [101] "SRB"  "SUR"  "SVK"  "SVN"  "SWE"  "THA"  "TJK"  "TUN"  "TUR"  "TWN"
## [111] "TZA"  "UKR"  "URY"  "USA"  "VEN"  "VNM"  "ZAF"

```

Razmatrat ćemo zemlju kao varijablu koja određuje grupe i ukupnu duljinu boravka kao zavisnu varijablu.

```

library(nortest)

# Print the results
for (country in validCountries) {
  cat("Lilliefors test for", country, "\n")
  print(lillie.test(data[data$Country == country, ]$ADR))
}

## Lilliefors test for AGO
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.13276, p-value < 2.2e-16
##
## Lilliefors test for ALB
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.16255, p-value = 0.5135
##
## Lilliefors test for AND
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.21449, p-value = 0.4265
##
## Lilliefors test for ARE
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.16968, p-value = 0.0008396
##
## Lilliefors test for ARG
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.088077, p-value = 0.0003675
##
## Lilliefors test for ARM
##

```

```

## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.3169, p-value = 0.01756
##
## Lilliefors test for AUS
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.13037, p-value < 2.2e-16
##
## Lilliefors test for AUT
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.084229, p-value < 2.2e-16
##
## Lilliefors test for AZE
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.16031, p-value = 0.2915
##
## Lilliefors test for BEL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.068706, p-value < 2.2e-16
##
## Lilliefors test for BGD
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.29804, p-value = 0.004191
##
## Lilliefors test for BGR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.070699, p-value = 0.4674
##
## Lilliefors test for BHR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.23242, p-value = 0.5015
##

```

```

## Lilliefors test for BIH
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.16752, p-value = 0.4074
##
## Lilliefors test for BLR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.080857, p-value = 0.9327
##
## Lilliefors test for BOL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.44898, p-value = 2.489e-06
##
## Lilliefors test for BRA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.075925, p-value < 2.2e-16
##
## Lilliefors test for CAF
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.36386, p-value = 0.02921
##
## Lilliefors test for CHE
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.083342, p-value < 2.2e-16
##
## Lilliefors test for CHL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.10398, p-value = 0.07809
##
## Lilliefors test for CHN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR

```

```

## D = 0.062764, p-value = 6.157e-10
##
## Lilliefors test for CIV
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.25977, p-value = 0.2314
##
## Lilliefors test for CMR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.094055, p-value = 0.9987
##
## Lilliefors test for CN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.085875, p-value < 2.2e-16
##
## Lilliefors test for COL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.094405, p-value = 0.1214
##
## Lilliefors test for CPV
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.10345, p-value = 0.7278
##
## Lilliefors test for CRI
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.13102, p-value = 0.5298
##
## Lilliefors test for CUB
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.18663, p-value = 0.5665
##
## Lilliefors test for CYP
##
## Lilliefors (Kolmogorov-Smirnov) normality test

```

```

##
## data: data[data$Country == country, ]$ADR
## D = 0.2433, p-value = 4.43e-08
##
## Lilliefors test for CZE
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.11725, p-value = 5.353e-06
##
## Lilliefors test for DEU
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.074594, p-value < 2.2e-16
##
## Lilliefors test for DNK
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.092975, p-value = 1.224e-09
##
## Lilliefors test for DOM
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.18494, p-value = 0.2172
##
## Lilliefors test for DZA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.25649, p-value < 2.2e-16
##
## Lilliefors test for ECU
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.19786, p-value = 0.008122
##
## Lilliefors test for EGY
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.20372, p-value = 0.001629
##
## Lilliefors test for ESP

```

```

## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.092447, p-value < 2.2e-16
## 
## Lilliefors test for EST
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.11923, p-value = 0.005308
## 
## Lilliefors test for FIN
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.091374, p-value = 1.477e-09
## 
## Lilliefors test for FRA
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.083457, p-value < 2.2e-16
## 
## Lilliefors test for FRO
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.31753, p-value = 0.09951
## 
## Lilliefors test for GBR
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.071577, p-value < 2.2e-16
## 
## Lilliefors test for GEO
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.24251, p-value = 0.001595
## 
## Lilliefors test for GIB
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.15445, p-value = 0.3078

```

```

##
## Lilliefors test for GNB
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.3203, p-value = 0.008295
##
## Lilliefors test for GRC
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.12885, p-value = 1.977e-05
##
## Lilliefors test for HKG
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.19423, p-value = 0.006667
##
## Lilliefors test for HRV
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.10338, p-value = 0.01034
##
## Lilliefors test for HUN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.11808, p-value = 2.933e-08
##
## Lilliefors test for IDN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.2199, p-value = 0.0001714
##
## Lilliefors test for IND
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.09284, p-value = 0.002712
##
## Lilliefors test for IRL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##

```

```

## data: data[data$Country == country, ]$ADR
## D = 0.070097, p-value < 2.2e-16
##
## Lilliefors test for IRN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.091559, p-value = 0.08232
##
## Lilliefors test for IRQ
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.32563, p-value = 0.0002685
##
## Lilliefors test for ISL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.30129, p-value = 3.308e-14
##
## Lilliefors test for ISR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.14432, p-value < 2.2e-16
##
## Lilliefors test for ITA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.090051, p-value < 2.2e-16
##
## Lilliefors test for JAM
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.27916, p-value = 0.1508
##
## Lilliefors test for JEY
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.19295, p-value = 0.5122
##
## Lilliefors test for JOR
##

```

```

## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.258, p-value = 0.0007768
##
## Lilliefors test for JPN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.079822, p-value = 0.003876
##
## Lilliefors test for KAZ
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.13519, p-value = 0.4786
##
## Lilliefors test for KEN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.20117, p-value = 0.6265
##
## Lilliefors test for KOR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.062881, p-value = 0.2222
##
## Lilliefors test for KWT
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.13511, p-value = 0.6063
##
## Lilliefors test for LBN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.14174, p-value = 0.1161
##
## Lilliefors test for LBY
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.40928, p-value = 0.0002734
##

```

```

## Lilliefors test for LKA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.29064, p-value = 0.07545
##
## Lilliefors test for LTU
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.11982, p-value = 0.005843
##
## Lilliefors test for LUX
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.080451, p-value = 0.0001209
##
## Lilliefors test for LVA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.20097, p-value = 7.897e-06
##
## Lilliefors test for MAC
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.27633, p-value = 0.001946
##
## Lilliefors test for MAR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.103, p-value = 5.296e-07
##
## Lilliefors test for MDV
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.15939, p-value = 0.5454
##
## Lilliefors test for MEX
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR

```

```

## D = 0.12227, p-value = 0.00313
##
## Lilliefors test for MKD
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.43146, p-value = 8.221e-06
##
## Lilliefors test for MLT
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.10395, p-value = 0.8737
##
## Lilliefors test for MNE
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.32967, p-value = 0.08006
##
## Lilliefors test for MOZ
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.15999, p-value = 0.0002034
##
## Lilliefors test for MUS
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.33759, p-value = 0.0155
##
## Lilliefors test for MYS
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.17421, p-value = 0.02921
##
## Lilliefors test for NGA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.19305, p-value = 0.002454
##
## Lilliefors test for NLD
##
## Lilliefors (Kolmogorov-Smirnov) normality test

```

```

##
## data: data[data$Country == country, ]$ADR
## D = 0.063904, p-value < 2.2e-16
##
## Lilliefors test for NOR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.084068, p-value = 5.335e-11
##
## Lilliefors test for NULL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.21433, p-value < 2.2e-16
##
## Lilliefors test for NZL
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.14983, p-value = 0.0002916
##
## Lilliefors test for OMN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.2534, p-value = 0.003361
##
## Lilliefors test for PAK
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.26574, p-value = 0.008313
##
## Lilliefors test for PAN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.25695, p-value = 0.08868
##
## Lilliefors test for PER
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.20147, p-value = 0.003979
##
## Lilliefors test for PHL

```

```

## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.14755, p-value = 0.02827
## 
## Lilliefors test for POL
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.088419, p-value < 2.2e-16
## 
## Lilliefors test for PRI
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.24147, p-value = 0.05166
## 
## Lilliefors test for PRT
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.11358, p-value < 2.2e-16
## 
## Lilliefors test for QAT
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.1435, p-value = 0.5561
## 
## Lilliefors test for ROU
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.13098, p-value < 2.2e-16
## 
## Lilliefors test for RUS
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.072036, p-value = 2.803e-08
## 
## Lilliefors test for SAU
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: data[data$Country == country, ]$ADR
## D = 0.12362, p-value = 0.06397

```

```

##
## Lilliefors test for SEN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.27032, p-value = 0.02398
##
## Lilliefors test for SGP
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.17809, p-value = 0.003129
##
## Lilliefors test for SRB
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.32347, p-value < 2.2e-16
##
## Lilliefors test for SUR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.21495, p-value = 0.6276
##
## Lilliefors test for SVK
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.17091, p-value = 6.485e-05
##
## Lilliefors test for SVN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.1175, p-value = 0.04832
##
## Lilliefors test for SWE
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.06409, p-value = 1.181e-10
##
## Lilliefors test for THA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##

```

```

## data: data[data$Country == country, ]$ADR
## D = 0.19572, p-value = 6.435e-06
##
## Lilliefors test for TJK
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.51945, p-value = 6.431e-08
##
## Lilliefors test for TUN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.11192, p-value = 0.2509
##
## Lilliefors test for TUR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.1043, p-value = 7.101e-07
##
## Lilliefors test for TWN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.1328, p-value = 0.02507
##
## Lilliefors test for TZA
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.33007, p-value = 0.07917
##
## Lilliefors test for UKR
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.10407, p-value = 0.0651
##
## Lilliefors test for URY
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.18375, p-value = 0.007555
##
## Lilliefors test for USA
##

```

```

## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.084869, p-value < 2.2e-16
##
## Lilliefors test for VEN
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.088914, p-value = 0.861
##
## Lilliefors test for VNM
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.26562, p-value = 0.09873
##
## Lilliefors test for ZAF
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: data[data$Country == country, ]$ADR
## D = 0.15991, p-value = 2.815e-05

```

Na temelju p-vrijednosti u našim rezultatima primjećujemo:

Mnoge zemlje imaju p-vrijednosti manje od 0,05, što ukazuje da podaci o ADR-u za te zemlje ne slijede normalnu distribuciju. Na primjer, zemlje poput ARE, ARG, BEL, BRA, CHN, COL, DEU, ESP, FRA, GBR, IND, ITA, JPN, MEX, NLD, NOR, PRT i SAD sve imaju p-vrijednosti manje od 0,05.

Neke zemlje imaju p-vrijednosti veće od 0,05, što sugerira da podaci o ADR-u za te zemlje mogu razumno odgovarati normalnoj distribuciji. Na primjer, ALB, AND, BGR, BHR, CIV, CMR, CPV, CRI, CYP i drugi imaju p-vrijednosti veće od 0,05.

Važno je napomenuti da niska p-vrijednost ne znači nužno da podaci nisu korisni ili nevažeći; to jednostavno sugerira da podaci možda ne slijede normalnu distribuciju.

Usprkos tome, jednofaktorska ANOVA je robusna statistika s obzirom na predpostavku normalnosti. Dobro tolerira odstupanja od normalne distribucije.

Što se tiče homogenosti varijanci različitih populacija, potrebno je testirati:

$$H_0 : \sigma_1^2 = \sigma_2^2 = \dots = \sigma_k^2$$

$$H_1 : \text{barem dvije varijance nisu iste.}$$

Navedenu hipotezu možemo testirati Bartletovim testom.

```

filteredDataCountries <- subset(data, Country %in% validCountries)

#Bartletovim test
bartlett.test(filteredDataCountries$ADR ~ filteredDataCountries$Country)

##
## Bartlett test of homogeneity of variances
##

```

```
## data: filteredDataCountries$ADR by filteredDataCountries$Country
## Bartlett's K-squared = 4090.7, df = 116, p-value < 2.2e-16
```

Na temelju rezultata Bartlettovog testa, čini se da varijance varijable “ADR” nisu jednake među različitim zemljama. Ovo sugerira da postoje značajne razlike u varijabilnosti “ADR” vrijednosti u zemljama koje smo testirali. Ovo je grafički prikazano u nastavku.

Dani podaci o distibuciji koja nije normalna i nehomogenosti varijanci navodi nas na zaključak da ADR varira ovisno o zemlji porjekla osobe, a to ćemo sada testirati.

Naša testne hipoteze biti će:

$$H_0 : \text{ADR je jednak za osobe neovisno o porjeklu}$$

$$H_1 : \text{za barem jednu osobu sa različitim porjeklom prosječni ADR je različita od ostalih}$$

#ANOVA

```
a = aov(filteredDataCountries$ADR ~ filteredDataCountries$Country)
summary(a)
```

```
##                               Df   Sum Sq Mean Sq F value Pr(>F)
## filteredDataCountries$Country    116 14355721 123756   50.8 <2e-16 ***
## Residuals                      119159 290288518    2436
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Zaključak:

Na temelju malene p-vrijednosti odbacujemo nullu hipotezu u korist alternativne hipoteze: **postoje razlike u ADR ovisno o porjeklu osobe.**

Case study: OBITELJI S DJECOM

Zanimaju nas obrasci ponašanja obitelji s djecom poput: koji tip sobe obitelji s djecom najčešće iznajmljuju, imaju li takvi gosti više posebnih zahtjeva naspram ostalih gostiju...

Kako bismo u nastavku zadatka mogli odgovoriti na postavljena pitanja, najprije ćemo postojeći dataset filtrirati i prilagoditi.

```
library(dplyr)

families = data_total_stays %>%
  filter(Adults > 0, Children >= 0, Babies > 0 | Adults > 0, Children > 0, Babies >= 0)
```

Postoji li razlika u popularnosti traženih tipova soba s obzirom na broj djece koje obitelj ima?

Uvodimo novu varijablu koja obuhvaća i djecu i bebe i nazivamo ju "TotalNonAdults".

```
library(dplyr)

families_children_num = families %>%
  mutate(TotalNonAdults = Children + Babies)
```

Možemo naslutiti da će postojati razlika u popularnosti traženih tipova soba kod obitelji koje, primjerice, ima jedno dijete naspram obitelji koje imaju troje ili više djece.

Naše hipoteze glase:

$$H_0 : \text{u svim tipovima soba su prosječno podjednako velike obitelji}$$
$$H_1 : \text{u barem jednom tipu sobe prosječna veličina obitelji je drugačija no u ostalih}$$

Ova problematika može se testirati ANOVOM, no ANOVA ima predpostavke na nezavisnost podataka u urozkusu, normalnu distribuciju i homogenost varijanci.

Želimo izbjegći predpostavku da je uzorak iz populacije normalne distribucije pa ćemo se u testiranju hipoteze poslužiti Kruskal-Wallisovim testom.

```
#KRUSKAL-WALLIS
kruskal.test(families_children_num$TotalNonAdults ~ families_children_num$ReservedRoomType)

## 
##  Kruskal-Wallis rank sum test
##
##  data:  families_children_num$TotalNonAdults by families_children_num$ReservedRoomType
##  Kruskal-Wallis chi-squared = 4994.4, df = 7, p-value < 2.2e-16
```

Iz rezultata provedenog testa možemo vidjeti malenu p-vrijednost što nalaže odbacivanje nulte hipoteze u korist alternativne hipoteze.

Zaključak:

U barem jednom tipu sobe prosječna veličina obitelji je drugačija no u ostalih, što možemo sročiti i tako da postoji razlika u popularnosti traženih tipova soba s obzirom na broj djece koje obitelj ima, kao što smo i predpostavili.

Postoji li razlika u količini posebnih zahtjeva s obzirom na broj djece koje obitelj ima?

Postupak dolaženja do zaključka isti je kao u prethodnom primjeru, kako bismo izbjegli predpostavku normalnosti normalne distribucije populacije poslužiti ćemo se Kruskal-Wallisovim testom.

Naše hipoteze glase:

$$H_0 : \text{prosječni broj posebnih zahtjeva ne razlikuje se s obzirom na broj članova obitelji}$$

$$H_1 : \text{za barem jedan broj članova obitelji, prosječni broj posebnih zahtjeva je drugačiji}$$

#KRUSKAL-WALLIS

```
kruskal.test(families_children_num$TotalNonAdults ~ families_children_num$TotalOfSpecialRequests)
```

```
##  
## Kruskal-Wallis rank sum test  
##  
## data: families_children_num$TotalNonAdults by families_children_num$TotalOfSpecialRequests  
## Kruskal-Wallis chi-squared = 202.86, df = 5, p-value < 2.2e-16
```

Iz rezultata provedenog testa možemo vidjeti malenu p-vrijednost što nalaže odbacivanje nulte hipoteze u korist alternativne hipoteze.

Zaključak:

U ovisno o veličini obitelji, mijenja se i prosječni broj posebnih zahtjeva što možemo sročiti i tako da postoji razlika u količini posebnih zahtjeva s obzirom na broj djece koje obitelj ima.

Case study: USPOREDBA LJETNIH SEZONA

ADR (“Average daily rate index”) je glavni identifikator hotelijesko-turističke djelatnosti kojim se mjeri prosječna zarada po svakoj iznajmljenoj sobi u određenoj perodu vremena. To je omjer ukupnog prihoda hotela i broja iznajmljenih soba. Vrlo je važan kako bi hoteli znali odrediti svoju poslovnu i financijsku strategiju za budućnost. Viši ADR bi značio da hotel generira više prihoda po sobi, što može biti znak dobrog poslovanja.

Je li ljetna sezona 2017. bila profitabilnija hotelima nego ljetna sezona 2016.?

Kako bismo dobili odgovor na ovo pitanje poslužit ćemo se prethodno opisanim parametrom, ADR-om, kao indikatorom uspješnosti poslovanja hotela.

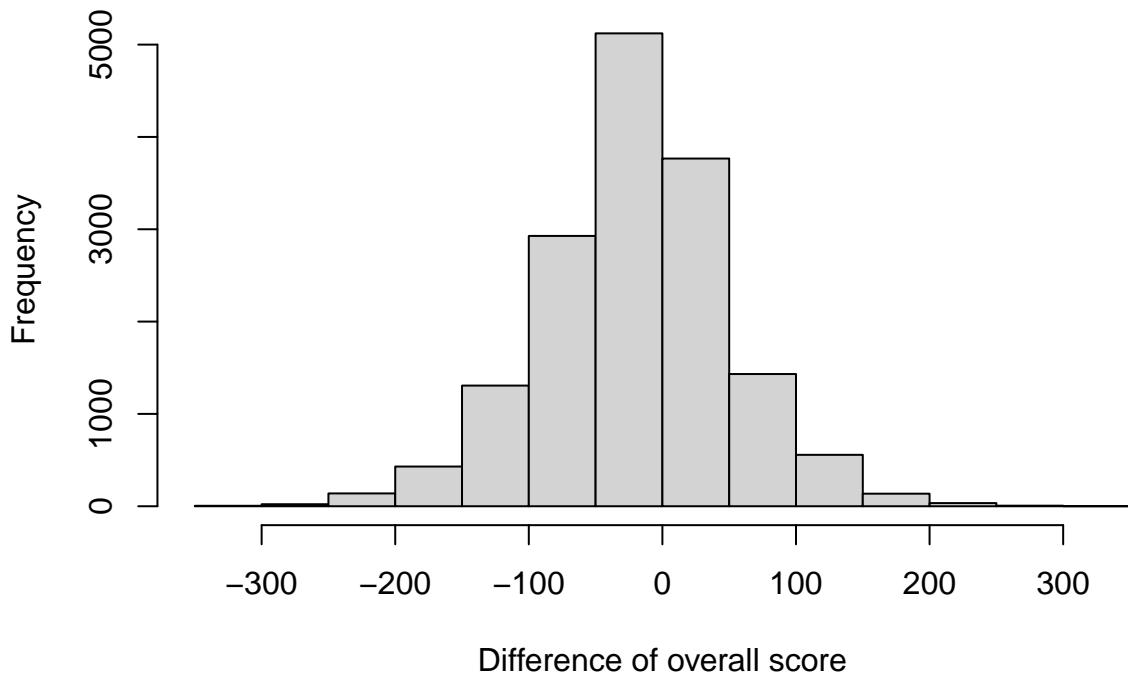
```
#creating utility datasets
#summer 2016 data
summer_2016 = rbind(data[data$ArrivalDateMonth == "June",],
                     data[data$ArrivalDateMonth == "July",],
                     data[data$ArrivalDateMonth == "August",])
summer_2016 = summer_2016[summer_2016$ArrivalDateYear == 2016, ]

#summer 2017 data
summer_2017 = rbind(data[data$ArrivalDateMonth == "June",],
                     data[data$ArrivalDateMonth == "July",],
                     data[data$ArrivalDateMonth == "August",])
summer_2017 = summer_2017[summer_2017$ArrivalDateYear == 2017, ]

#plotting data
#histogram of difference
hist(summer_2016$ADR - summer_2017$ADR,
     main=paste('Histogram of ADR in 2016 - 2017'),
     xlab='Difference of overall score')

## Warning in summer_2016$ADR - summer_2017$ADR: longer object length is not a
## multiple of shorter object length
```

Histogram of ADR in 2016 – 2017



```
#qq-plot for overall ADR
qqnorm(summer_2016$ADR - summer_2017$ADR,
      pch = 1,
      frame = FALSE,
      main= paste('QQ-plot for overall score of ADR'))
```

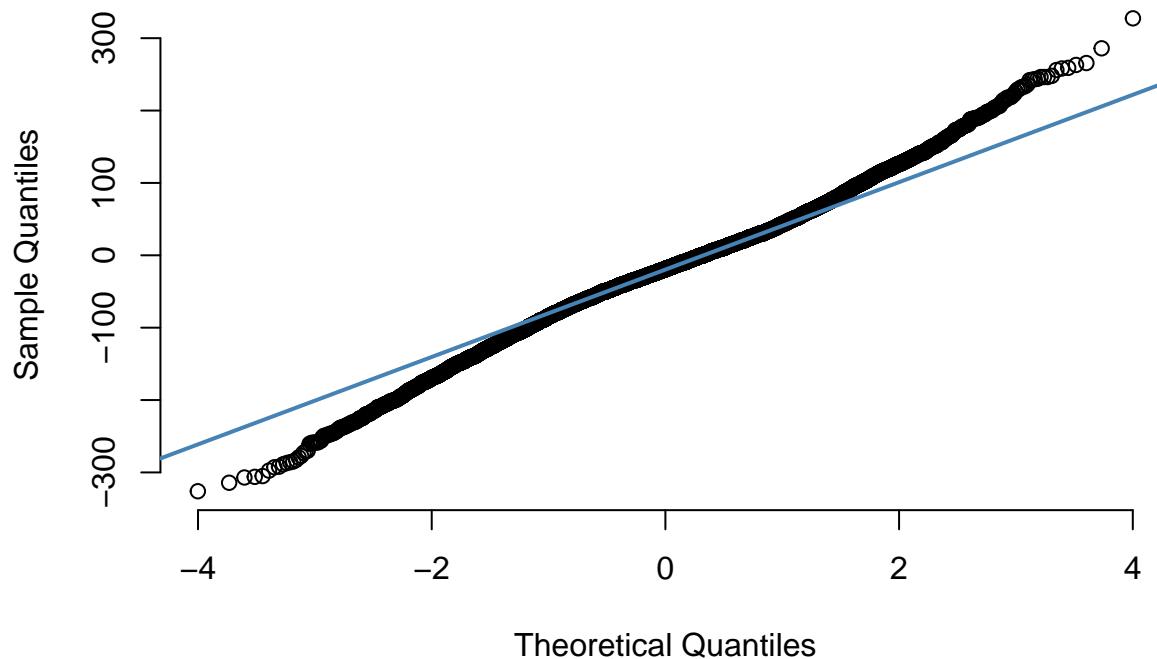


```
## Warning in summer_2016$ADR - summer_2017$ADR: longer object length is not a
## multiple of shorter object length
```

```
qqline(summer_2016$ADR - summer_2017$ADR,
       col = "steelblue", lwd = 2)
```

```
## Warning in summer_2016$ADR - summer_2017$ADR: longer object length is not a
## multiple of shorter object length
```

QQ-plot for overall score of ADR



Iz vizualizacije razlika možemo naslutiti normalnost podataka, dok iz qq-plota vidimo tek malo odstupanje lijevog repa i desnog repa, uz normalnost podataka.

Naša hipoteze su sljedeće:

$$H_0 : \text{ljetne sezone su jednako profitabilne}$$

$$H_1 : \text{sezona 2017. profitabilnija je od ljjetne sezone 2016.}$$

```
t.test(summer_2016$ADR,
       summer_2017$ADR,
       alt = "less")
```

```
##
##  Welch Two Sample t-test
##
## data: summer_2016$ADR and summer_2017$ADR
## t = -31.546, df = 30802, p-value < 2.2e-16
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -17.67074
## sample estimates:
## mean of x mean of y
## 124.8281 143.4710
```

Rezultati t-testa nalažu nam odbacivanje nulte hipoteze u korist alternativne hipoteze.

Zaključak:

2017. je hotelima bila profitabilnija godina, što smo mogli zaključiti i iz prikazanih histograma.

Kako da hoteli u budućnosti budu još profitabilniji?

Da bismo pomogli hotelima u stvaranju još bolje poslovne strategije, ispitati ćemo korelacije nekoliko parametara iz naših podataka sa ADR-om. Za utvrđivanje korelacije koristit ćemo se regresijskim metodama, gdje je kao zavisna varijabla postavljen ADR, a kao nezavisne varijable:

- "DaysInWaitingList"

Konkretno, u oba slučaja, korelaciju ćemo testirati jednostavnom linearnom regresijom.

Days in waiting list

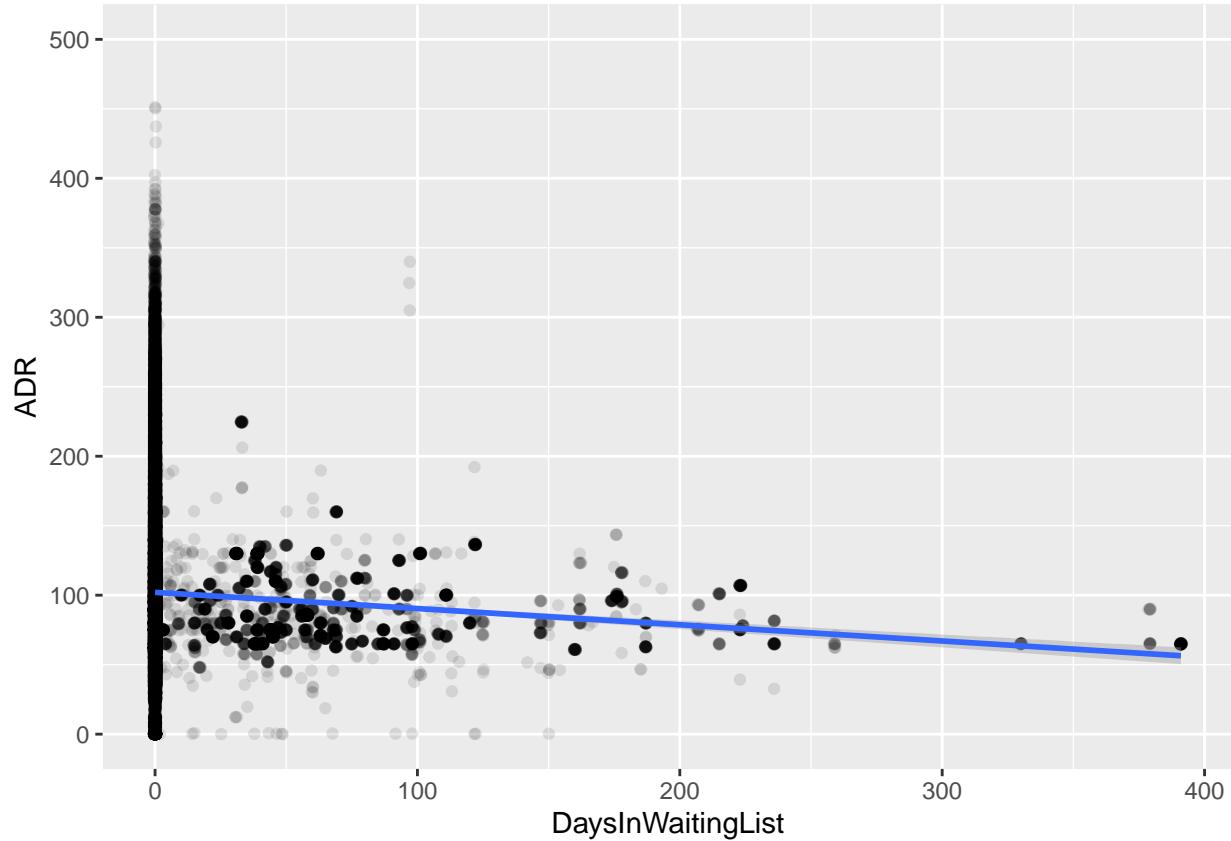
Zanima nas je li broj dana na listi čekanja koreliran sa ADR-om, tj. kada bi gost bio kraće na listi čekanja, bi li hotel bio "profitabilniji"?

```
library(ggplot2)

ggplot(data_total_stays, aes(x = DaysInWaitingList, y = ADR )) +
  geom_jitter(height = 0.5, alpha = .1) +
  geom_smooth(method = "lm", se = TRUE) +
  ylim(0, 500)

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

## Warning: Removed 4 rows containing non-finite values ('stat_smooth()').
## Warning: Removed 996 rows containing missing values ('geom_point()').
```



Iz prikaza možemo naslutiti da određena korelacija postoji, no idemo to testirati linearnom regresijom:

```
linear_model <- lm(formula = ADR ~ DaysInWaitingList, data = data)
summary(linear_model)
```

```
##
## Call:
## lm(formula = ADR ~ DaysInWaitingList, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -108.5  -32.1   -7.1   23.9 5297.9
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 102.102838  0.147402 692.68  <2e-16 ***
## DaysInWaitingList -0.117061  0.008306 -14.09  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 50.49 on 119388 degrees of freedom
## Multiple R-squared:  0.001661, Adjusted R-squared:  0.001653
## F-statistic: 198.6 on 1 and 119388 DF, p-value: < 2.2e-16
```

Zaključak:

Provedeni test nam govori da će, za povećanje varijable "DaysInWaitingList" za 1, ADR biti promjenjen

za -0.117061 (Estimate), sa standardom greškom od 0.008306 (Std. Error). Vjerojatnost da će se dogoditi rezultat jednak ili ekstremniji t-vrijednosti je vrlo blizu nuli, što znači da je parametar “DaysInWaitingList” signifikantan za predviđanje parametra ADR.

Zaključujemo da su ta dva parametra značajno korelirana, te da će **kraće vrijeme na listi čekanja pridonositi većoj profitabilnosti hotela**.