

# ATP Data Analysis

19.1.2024.

Instalacija potrebnih paketa.

```
# install.packages("dplyr")
# install.packages("lubridate")
# install.packages("ggplot2")
# install.packages("caret")
# install.packages("nortest")
# install.packages("fastDummies")
```

Učitavanje biblioteka.

```
library(dplyr)
library(lubridate)
library(ggplot2)
library(caret)
library(nortest)
library(fastDummies)
```

Učitavanje i opis podataka

```
all_matches <- data.frame()
for (year in 1991:2023) {
  file_name <- paste0("dataset/atp_matches_", year, ".csv")
  matches_year <- read.csv(file_name, stringsAsFactors = FALSE)
  all_matches <- rbind(all_matches, matches_year)
}

dim(all_matches)

## [1] 104682      49
```

Skup podataka sadrži informacije o 104682 teniska meča održana od 1991. do 2023. godine uključivo. Svaki meč opisan je s 49 ispod navedenih značajki:

```
names(all_matches)

##  [1] "tourney_id"          "tourney_name"        "surface"
##  [4] "draw_size"           "tourney_level"       "tourney_date"
##  [7] "match_num"           "winner_id"           "winner_seed"
## [10] "winner_entry"         "winner_name"         "winner_hand"
## [13] "winner_ht"           "winner_ioc"          "winner_age"
## [16] "loser_id"            "loser_seed"          "loser_entry"
## [19] "loser_name"          "loser_hand"          "loser_ht"
## [22] "loser_ioc"           "loser_age"           "score"
## [25] "best_of"              "round"               "minutes"
## [28] "w_ace"                "w_df"                "w_svpt"
## [31] "w_1stIn"              "w_1stWon"            "w_2ndWon"
## [34] "w_SvGms"              "w_bpSaved"           "w_bpFaced"
```

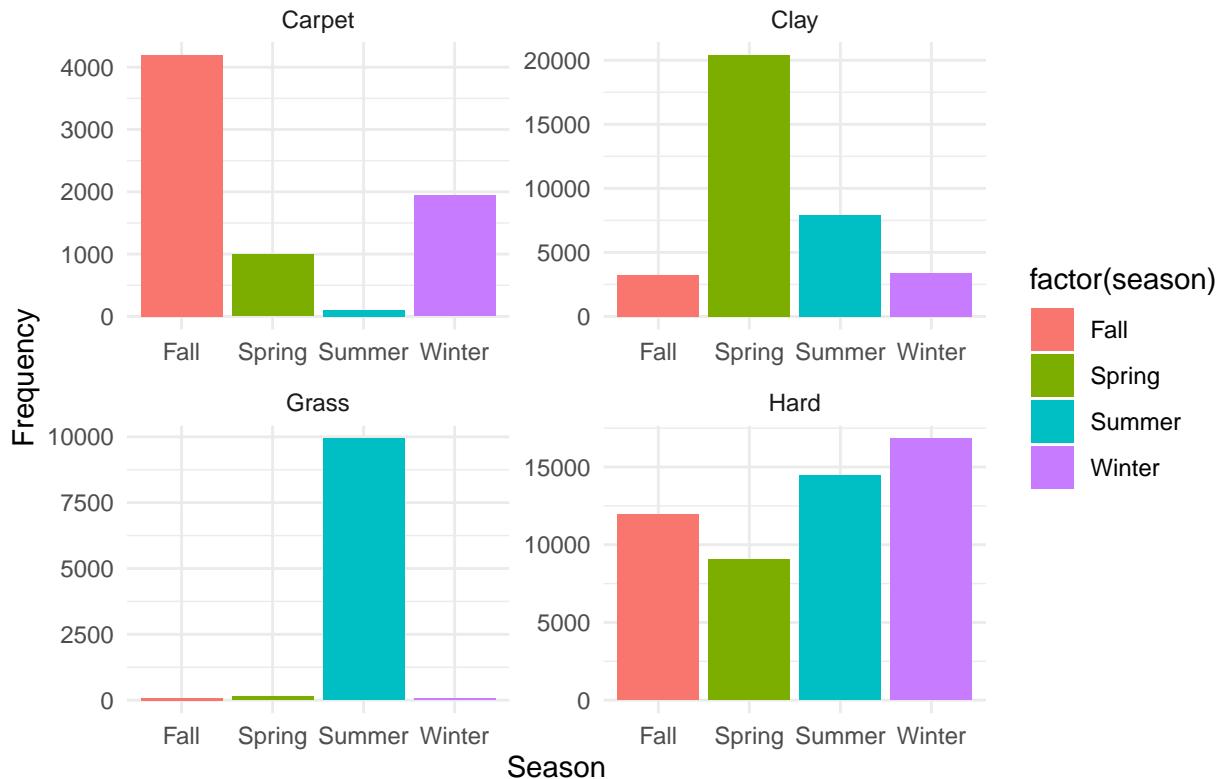
```

## [37] "l_ace"
## [40] "l_1stIn"
## [43] "l_SvGms"
## [46] "winner_rank"
## [49] "loser_rank_points"
## [40] "l_df"
## [41] "l_1stWon"
## [42] "l_bpSaved"
## [43] "winner_rank_points"
## [44] "l_svpt"
## [45] "l_2ndWon"
## [46] "l_bpFaced"
## [47] "loser_rank"

```

**Zadatak 1.** Kakva je distribucija mečeva na specifičnim podlogama u različitim godišnjim dobima?

Season Frequencies for Each Surface



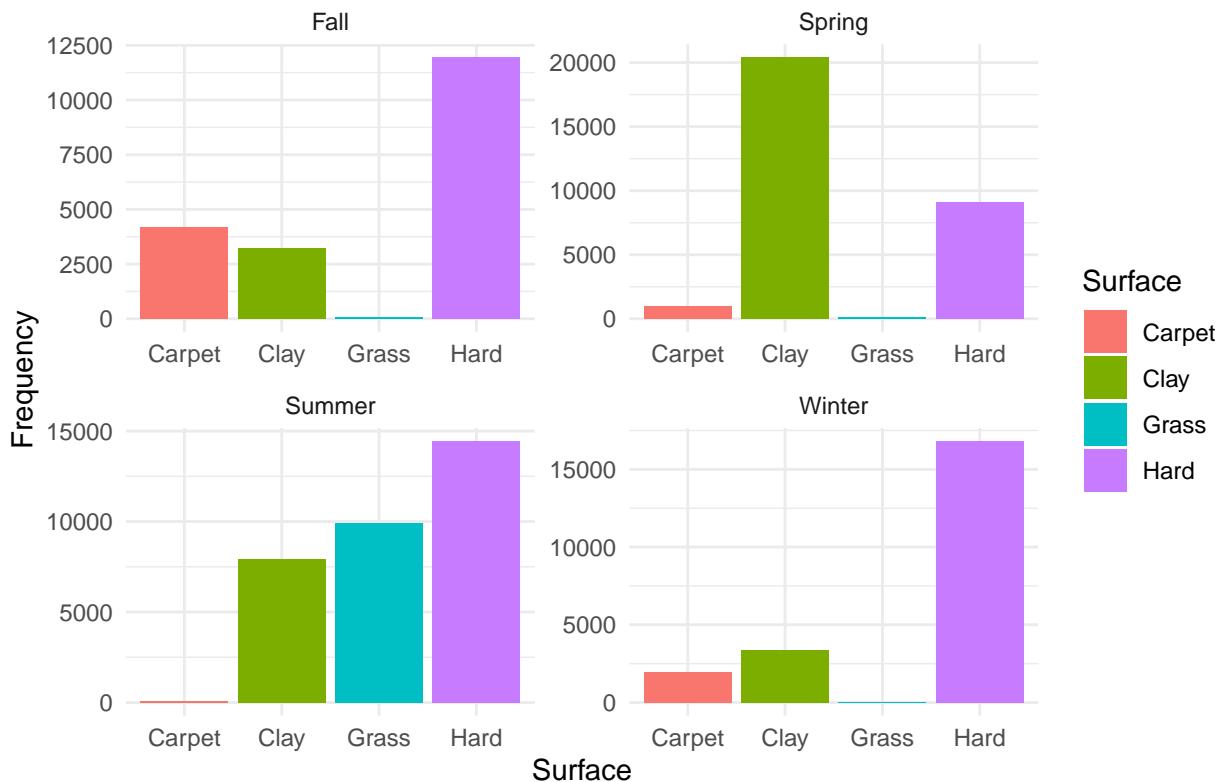
U prvom histogramu prikazana je raspodjela teniskih mečeva prema godišnjim dobima na podlozi od tepiha. Podloga od tepiha najmanje je korištena podloga za igranje mečeva. Najčešće se podloga od tepiha koristila u jesen, dosta rjeđe zimi, zatim na proljeće, a najmanje se mečeva na podlozi od tepiha igra na ljeto.

Sljedeći histogram predstavlja raspodjelu mečeva prema godišnjim dobima na zemljanoj podlozi. Mečevi na zemlji najčešće se igraju u proljetnom dijelu sezone. Dosta manje mečeva igra se na ljeto zatim otprilike podjednako na jesen i zimi.

Treći histogram opisuje distribuciju teniskih mečeva prema godišnjim dobima na travi. Teniski mečevi na travi igraju se uglavnom ljeti, a svega nekoliko mečeva igra se u preostalim godišnjim dobima.

U posljednjem histogramu promatrana je raspodjela mečeva prema godišnjim dobima na tvrdoj podlozi. Sveukupno najviše mečeva igra se na tvrdoj podlozi te je raspodjela prema godišnjim dobima manje izražena nego kod drugih podloga. Najviše mečeva na tvrdoj podlozi održava se zimi, zatim u ljeto pa na jesen te najmanje u proljetnom dijelu sezone.

## Surface Frequencies for Each Season



Prvi histogram prikazuje raspodjelu mečeva prema podlogama u jesen. Uvjerljivo najviše mečeva u jesen održava se na tvrdoj podlozi. Dosta manje mečeva igra se na podlozi od tepiha, a nešto malo manje na zemlji. Najmanje mečeva u jesenskom dijelu sezone igra se na travi.

Idući histogram prikazuje raspodjelu mečeva prema podlogama u proljeće. U proljetnom dijelu sezone uvjerljivo najviše teniskih mečeva igra se na podlozi od zemlje. Više od dvostruko manje mečeva održava se na tvrdoj podlozi. Jako malo mečeva održava se na podlozi od tepiha, a još manje na travi.

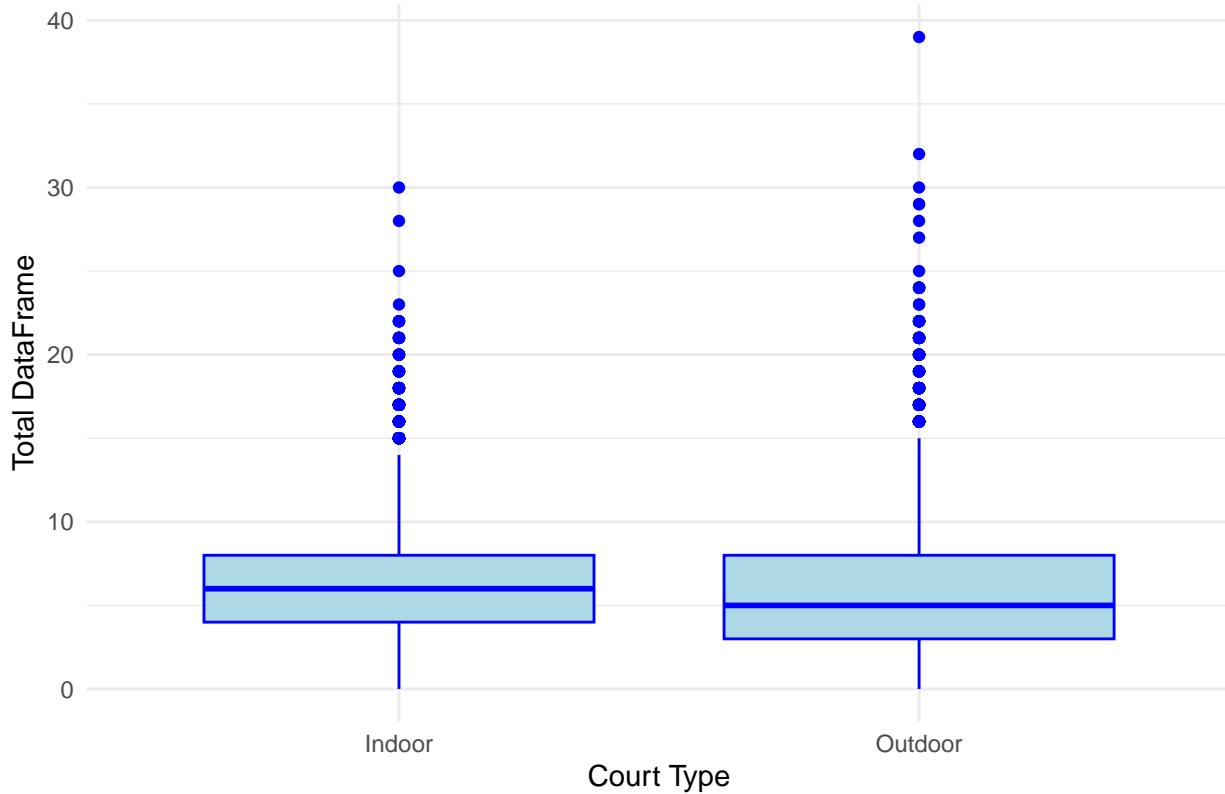
U trećem histogramu promatramo raspodjelu mečeva prema podlogama tijekom ljeta. Najviše mečeva održava se na tvrdoj podlozi, zatim na travi pa na podlozi od zemlje. Svega nekoliko mečeva igra se na podlozi od tepiha.

Zadnji histogram opisuje raspodjelu mečeva prema podlogama zimi. Tijekom zime prednjače mečevi na tvrdoj podlozi. Dosta manje mečeva igra se na zemlji, zatim na podlozi od tepiha te najmanje na travi.

**Zadatak 2.** Postoji li značajna razlika u prosječnom broju dvostrukih pogrešaka između mečeva odigranih na otvorenom u odnosu na mečeve odigrane na zatvorenom terenu?

Prvo provjeravamo ukazuje li boxplot za moguću značajnu razliku.

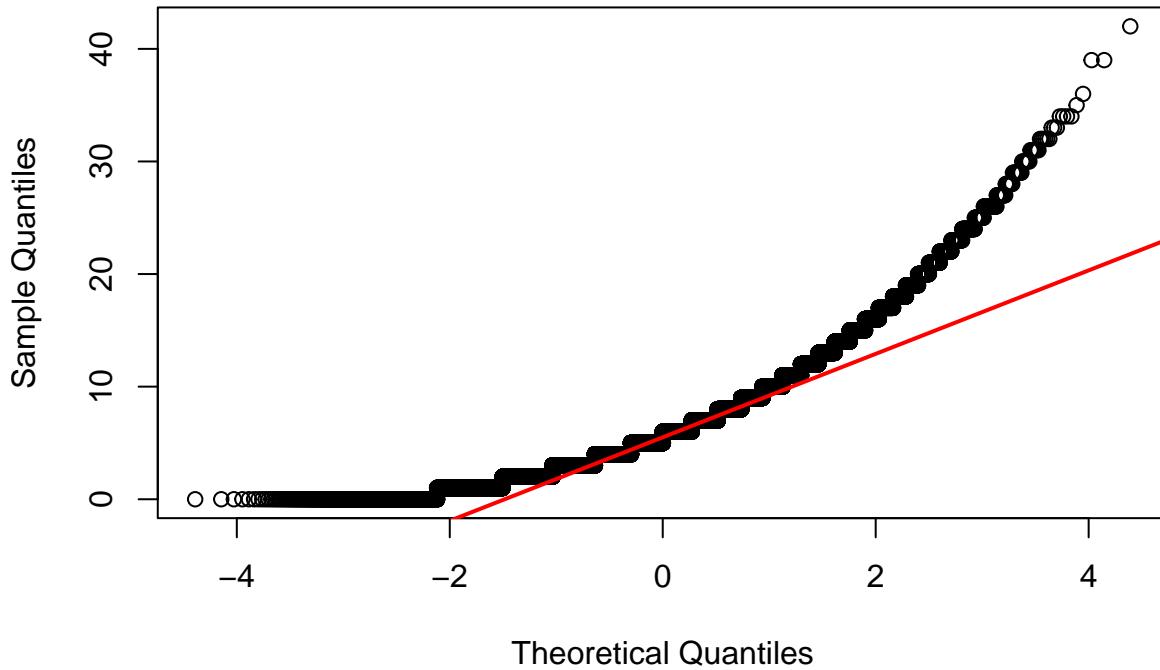
## Comparison of Outdoor and Indoor Court Types



Grafički prikaz ukazuje na moguću razliku između prosječnog broja dvostrukih pogrešaka između mečeva odigranih na otvorenom i zatvorenom. Kako bismo provjerili možemo li prihvati nultu hipotezu koja pretpostavlja da nema razlike, provest ćemo t-test. Najprije moramo provjeriti pretpostavke o normalnoj distribuciji i homogenosti varijanci. Normalnu distribuciju prvo provjeravamo pomoću qq-plota, a zatim i Lilliefors testom.

```
qqnorm(open_surface_data, main = "Normal Q-Q Plot for outdoor courts")
qqline(open_surface_data, col = "red", lwd = 2)
```

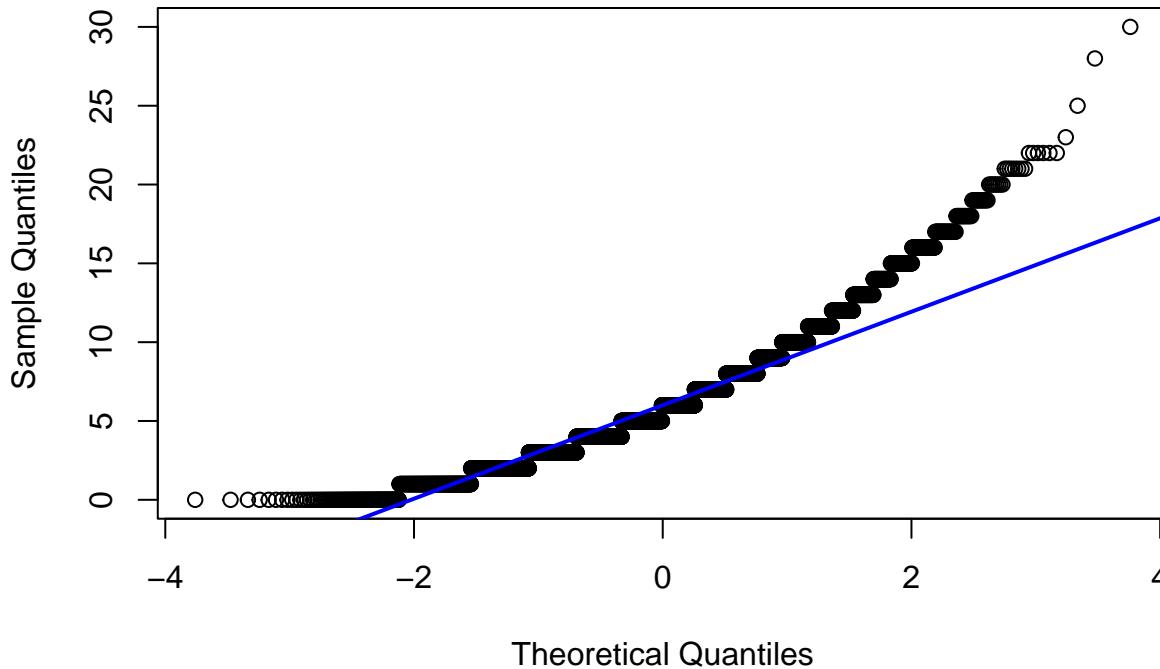
### Normal Q-Q Plot for outdoor courts



Theoretical Quantiles

```
qqnorm(closed_surface_data, main = "Normal Q-Q Plot for indoor courts")
qqline(closed_surface_data, col = "blue", lwd = 2)
```

### Normal Q-Q Plot for indoor courts



Theoretical Quantiles

Iz qq-plota vidljivo je da distribucije nisu normalne niti za mečeve na otvorenom niti na zatvorenom jer postoji značajno odstupanje repa. Zatim provodimo Lilliefors test:

```

lillie_test_outdoor <- lillie.test(open_surface_data)
lillie_test_indoor <- lillie.test(closed_surface_data)

##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: open_surface_data
## D = 0.12974, p-value < 2.2e-16

##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: closed_surface_data
## D = 0.12216, p-value < 2.2e-16

```

Za oba skupa podataka (otvoreni teren i zatvoren teren), rezultati testova normalnosti (Lilliefors test) pokazuju da podaci nisu normalno distribuirani (p-vrijednosti su manje od 0.05) što se moglo zaključiti i iz grafova.

Provjerimo F-test za provjeru homogenosti varijanci:

```
var_test_result <- var.test(open_surface_data, closed_surface_data)
```

```
print(var_test_result)
```

```

##
##  F test to compare two variances
##
## data: open_surface_data and closed_surface_data
## F = 1.1441, num df = 88596, denom df = 5877, p-value = 4.316e-12
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.101871 1.187308
## sample estimates:
## ratio of variances
## 1.144146

```

F-test za usporedbu varijanci pokazuje da postoji značajna razlika u varijancama između otvorenog terena i zatvorenog terena (p-vrijednost < 0.05). Kako pretpostavke za t-test nisu zadovoljene koristimo Wilcoxonov test:

```
print(wilcox.test(open_surface_data, closed_surface_data, alternative = "two.sided"))
```

```

##
##  Wilcoxon rank sum test with continuity correction
##
## data: open_surface_data and closed_surface_data
## W = 258377269, p-value = 0.3191
## alternative hypothesis: true location shift is not equal to 0

```

Wilcoxon rank-sum test ne pokazuje značajnu razliku u srednjim vrijednostima (medijanama) između otvorenog i zatvorenog terena (p-vrijednost = 0.3191, dakle ne možemo odbaciti nullu hipotezu).

Na temelju ovih rezultata, možemo zaključiti da nema značajne razlike u prosječnom broju dvostrukih pogrešaka između mečeva odigranih na otvorenom terenu i mečeva odigranih na zatvorenom terenu.

### Zadatak 3. Ima li razlike u broju serviranih asova na različitim podlogama?

Provjerimo za početak postoje li lako uočljive razlike u broju serviranih asova na različitim podlogama pomoću grafičkog prikaza.

```
t3 <- all_matches %>%
  filter(!is.na(w_ace) & !is.na(l_ace) & !is.na(surface) & w_ace != "" & l_ace != "" & surface != "")
```

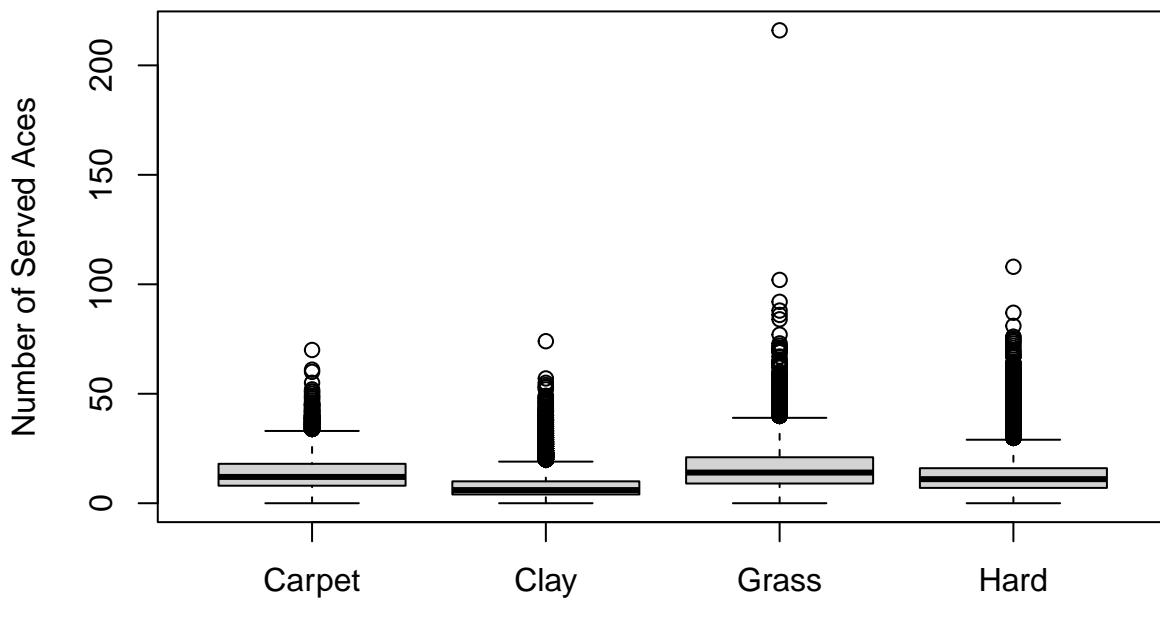
```
t3 <- select(t3, surface, w_ace, l_ace)
```

```
t3 <- t3 %>%
  mutate(aces = w_ace + l_ace)
```

```
boxplot(aces ~ surface, data = t3,
        xlab = "Surface", ylab = "Number of Served Aces",
        main = "Boxplot of Served Aces by Surface")
```

**Boxplot of Served Aces by Surface**



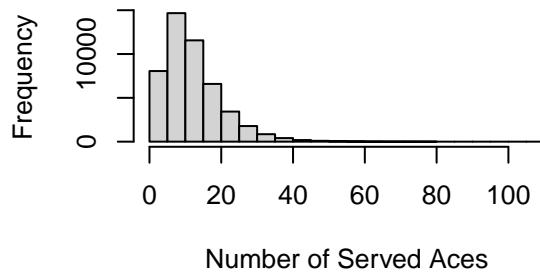
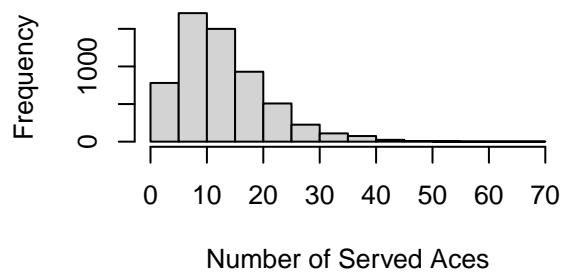
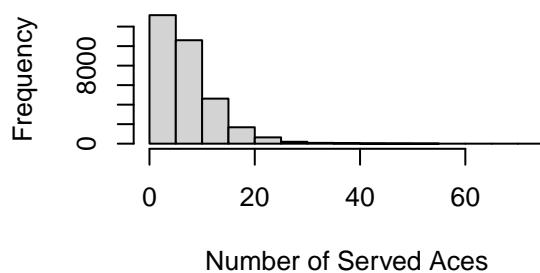
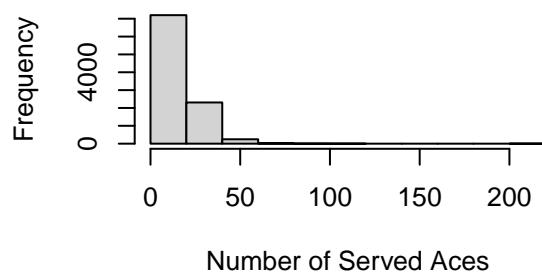
plot ukazuje na to da postoje razlike u broju asova s obzirom na podlogu.

Nulta hipoteza jest da nema razlike u broju serviranih asova na različitim podlogama. Može li se odbaciti ista pokušat ćemo provjeriti ANOVA testom. ANOVA analizira razliku srednje vrijednosti između više od dvije grupe. Kako bi taj test mogao biti korišten najprije moramo provjeriti jesu li zadovoljene prepostavke:

1. provjera normalne distribucije

Prvo ćemo isertati histograme serviranih asova za svaku od površina:

```
par(mfrow = c(2, 2))
for (surface in unique(t3$surface)){
  hist(t3$aces[t3$surface==surface],
       main = paste("Histogram of served aces on" , surface),
       xlab = "Number of Served Aces",
       ylab = "Frequency")
}
```

**Histogram of served aces on Hard****Histogram of served aces on Carpet****Histogram of served aces on Clay****Histogram of served aces on Grass**

Histogrami ukazuju na to da distribucija serviranih asova nije normalna niti na jednoj od podloga. Normalna distribucija još se provjerava i Lilliefors testom:

```
require(nortest)
print(lillie.test(t3$aces[t3$surface=='Hard']))  
  
##  
## Lilliefors (Kolmogorov-Smirnov) normality test  
##  
## data: t3$aces[t3$surface == "Hard"]  
## D = 0.11436, p-value < 2.2e-16  
  
print(lillie.test(t3$aces[t3$surface=='Carpet']))  
  
##  
## Lilliefors (Kolmogorov-Smirnov) normality test  
##  
## data: t3$aces[t3$surface == "Carpet"]  
## D = 0.10864, p-value < 2.2e-16  
  
print(lillie.test(t3$aces[t3$surface=='Clay']))  
  
##  
## Lilliefors (Kolmogorov-Smirnov) normality test  
##  
## data: t3$aces[t3$surface == "Clay"]  
## D = 0.13505, p-value < 2.2e-16  
  
print(lillie.test(t3$aces[t3$surface=='Grass']))  
  
##  
## Lilliefors (Kolmogorov-Smirnov) normality test
```

```

## 
## data: t3$aces[t3$surface == "Grass"]
## D = 0.10802, p-value < 2.2e-16

```

Za svaku od 4 podloge p-vrijednost je manja od 0.05 zbog čega odbacujemo pretpostavku da je distribucija normalna.

## 2. provjera homogenosti varijanci

Homogenost varijanci provjerava se Bartlettovim testom:

```
bartlett.test(t3$aces ~ t3$surface)
```

```

## 
##  Bartlett test of homogeneity of variances
## 
## data: t3$aces by t3$surface
## Bartlett's K-squared = 7049.2, df = 3, p-value < 2.2e-16

```

P-vrijednost u Bartlettovom testu manja je od kritične vrijednosti od 0.05 čime se zaključuje da homogenost varijanci nije zadovoljena.

Isto se vidi i u ispisu varijance za svaku od podloga.

```
var((t3$aces[t3$surface=='Hard']))
```

```
## [1] 65.28138
```

```
var((t3$aces[t3$surface=='Carpet']))
```

```
## [1] 63.26659
```

```
var((t3$aces[t3$surface=='Clay']))
```

```
## [1] 31.9019
```

```
var((t3$aces[t3$surface=='Grass']))
```

```
## [1] 104.5289
```

Kako niti jedna od pretpostavki nije zadovoljena koristit ćemo Kruskal-Wallis test, neparametarsku alternativu ANOVA testu.

```
kruskal.test(aces~surface, data=t3)
```

```

## 
##  Kruskal-Wallis rank sum test
## 
## data: aces by surface
## Kruskal-Wallis chi-squared = 13657, df = 3, p-value < 2.2e-16

```

Nakon provedenog testa dobivamo p-vrijednost manju od 0.05 što znači da možemo odbaciti nultu hipotezu, odnosno da postoji razlika u broj serviranih asova u odnosu na podlogu. Intuitivno ovaj rezultat ima smisla jer loptica ne odskače jednako od svih podloga, npr. na zemljanoj podlozi loptica se sporije odbija i obično zadržava niže dok se na travi odbija brzo.

## Zadatak 4. Kakva je veza između vrste terena i vjerojatnosti da će mečevi otići u peti set?

Postavljamo nultu hipotezu kako ne postoji statistički značajne veze između vrste terena i vjerojatnosti da će mečevi otići u peti set, a alternativna hipoteza sugerira prisutnost takve veze. Kako bismo testirali ovu hipotezu, koristit ćemo  $\chi^2$  test.

Najprije, provjeravamo pretpostavke kako bismo osigurali ispravnu primjenu testa.

Omogućena je nezavisnost podataka jer rezultat jednog teniskog meča ne utječe na rezultat drugog meča.

Također, osiguravamo da su nam podaci kategorički, klasifikacijom vrsta terena i ishoda mečeva u diskretne kategorije. Stvaramo kontingencijsku tablicu:

```
t4 <- all_matches[all_matches$best_of == 5, ]
t4$sets_played <- sapply(strsplit(as.character(t4$score), ""), function(x) sum(x == "-"))

contingency_table <- table(t4$surface, t4$sets_played == 5)
print(contingency_table)

##
```

	FALSE	TRUE
Carpet	700	179
Clay	5550	1240
Grass	3471	819
Hard	9090	2054

Kontingencijskoj tablici dodajemo sume redaka i stupaca:

```
##
```

	FALSE	TRUE	Sum
Carpet	700	179	879
Clay	5550	1240	6790
Grass	3471	819	4290
Hard	9090	2054	11144
Sum	18811	4292	23103

Još jedna pretpostavka testa je da očekivana frekvencija pojedinog razreda mora biti veća ili jednaka 5, stoga i to provjeravamo:

```
## Očekivane frekvencije za razred FALSE - Carpet : 715.7022
## Očekivane frekvencije za razred FALSE - Clay : 5528.576
## Očekivane frekvencije za razred FALSE - Grass : 3493.018
## Očekivane frekvencije za razred FALSE - Hard : 9073.704
## Očekivane frekvencije za razred TRUE - Carpet : 163.2978
## Očekivane frekvencije za razred TRUE - Clay : 1261.424
## Očekivane frekvencije za razred TRUE - Grass : 796.9822
## Očekivane frekvencije za razred TRUE - Hard : 2070.296
```

Sve očekivane pretpostavke su zadovoljene, nastavljamo sa  $\chi^2$  testom.

```
chi_square_result <- chisq.test(contingency_table)
print(chi_square_result)
```

```
##
## Pearson's Chi-squared test
##
## data: contingency_table
## X-squared = 3.2059, df = 3, p-value = 0.361
```

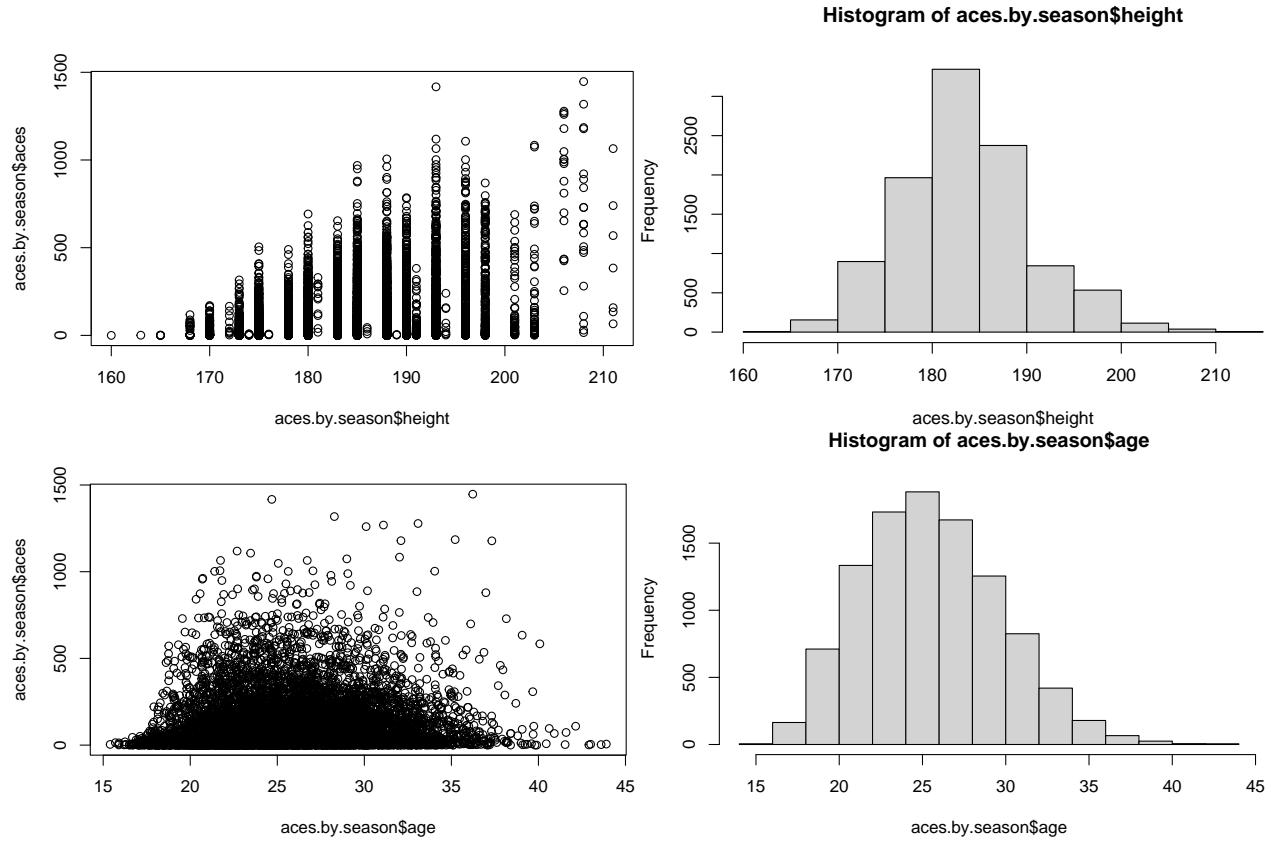
Rezultati  $\chi^2$  testa ukazuju na to da ne postoji statistički značajna veza između vrste terena na kojem se održavaju teniski mečevi i vjerojatnosti da će mečevi otici u peti set (p-vrijednost = 0.361). S obzirom na p-vrijednost veću od 0.05, ne odbacujemo nultu hipotezu.

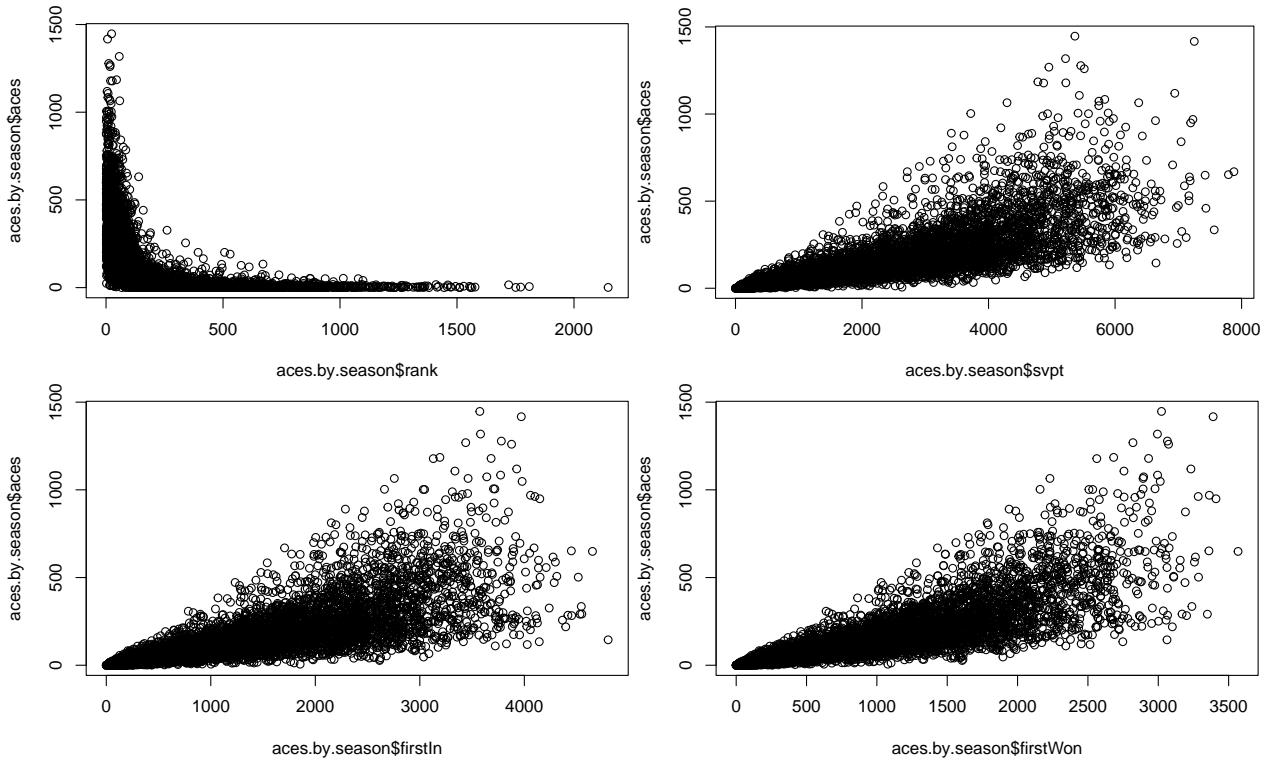
## Zadatak 5. Možemo li procijeniti broj asova koje će igrač odservirati u tekućoj godini (zadnjoj dostupnoj sezoni) na temelju njegovih rezultata iz prethodnih sezona?

Kako bismo predvidjeli broj asova za nekog igrača prvo moramo sagledati koje bi sve značajke mogle utjecati na broj odserviranih asova:

- Visina igrača
- Starost igrača
- Rank igrača
- Broj servi u sezoni
- Broj uspješnih prvih servi
- Broj osvojenih prvih servi

S obzirom da se podatci odnose pojedinačne mečeve, potrebno ih je agregirati za svakog igrača na razini zone. Prilikom agregacije uzet ćemo srednju vrijednost za starost i rank igrača dok ćemo ostale statistike o servi sumirati. Kako bi dobili dojam o odnosu ovih varijabli i broja asova prikazat ćemo ih pojedinačno grafički pomoću scatter plota.





Starost i visina igrača naizgled imaju nelinearni odnos prema broju aseva. Međutim, ako ispišemo histograme starosti i visina vidimo da je taj odnos proizašao iz frekvencije pojavljivanja tih značajki. Ovo ne vrijedi u potpunosti za visine igrača, ali pokazalo se da visina nije znatno utjecala na točnost modela. Zbog toga te dvije značajke nećemo dalje uzimati u obzir.

Rank igrača ima snažan obrnuto proporcionalan odnos s brojem asova što intuitivno ima smisla. Bolji igrači će imati veći broj asova. Važno je uočiti da ovaj odnos nije linearan i te ćemo morati uzeti inverz ranga prilikom regresije.

Ukupan broj servi, prvih uspješnih i prvih osvojenih servi ima linearan odnos prema broju aseva. Također uočavamo da su ove tri značajke naizgled jako korelirane. To ćemo morati ispitati i po potrebi uzeti samo jedno od ovih značajki za našu regresiju.

Valja spomenuti da su u obzir još bili uzeti podatci o servama, ali u postotcima. Oni su imali sličan problem kao i starost i visina. Scatter plot u odnosu na broj asova je imao oblik normalne distribucije. Dominantna ruka također nije uzeta u obzir jer prednost koju ona donosi ovisi o dominantnoj ruci suparnika, a pošto su podatci agregirani prednosti i hendikepi dominantne ruke će se poništiti.

Provjerit ćemo naše pretpostavke koristeći model jednostavne regresije, po jedan za svaku značajku. Broj asova će biti zavisna varijabla

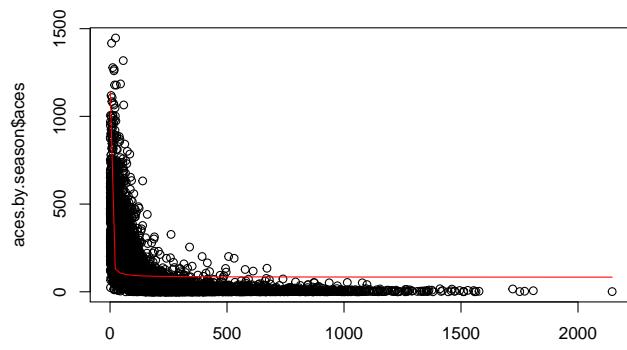
```
fit.invRank = lm(aces ~ invRank, data=aces.by.season)

fit.svpt = lm(aces ~ svpt, data=aces.by.season)

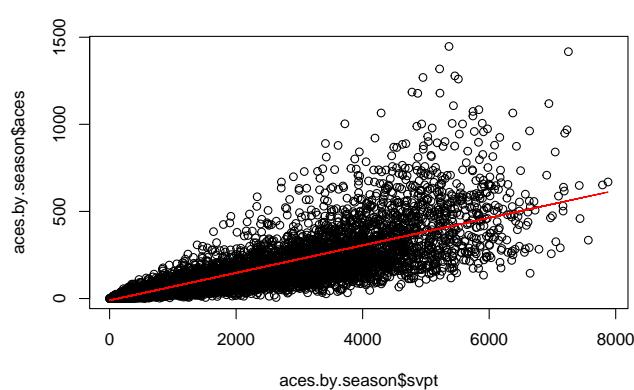
fit.firstIn = lm(aces ~ firstIn, data=aces.by.season)

fit.firstWon = lm(aces ~ firstWon, data=aces.by.season)

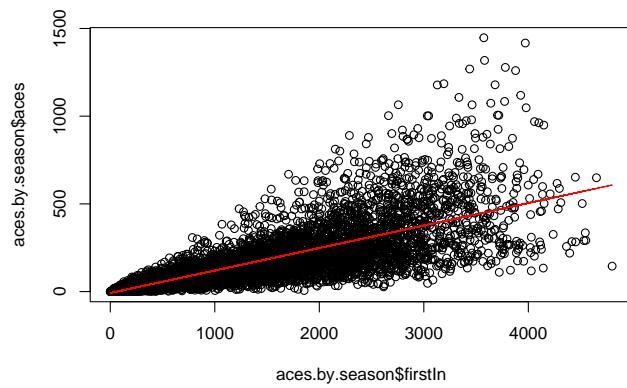
f = function(x, coeffs)
  return(coeffs[[1]] + + coeffs[[2]] * (1 / x))
plot(aces.by.season$rank, aces.by.season$aces)
curve(f(x, fit.invRank$coefficients), add = TRUE, col = "red")
```



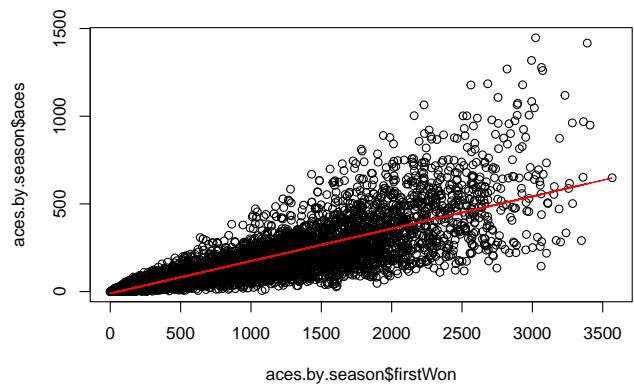
```
plot(aces.by.season$svpt, aces.by.season$aces)
lines(aces.by.season$svpt, fit.svpt$fitted.values, col='red')
```



```
plot(aces.by.season$firstIn, aces.by.season$aces)
lines(aces.by.season$firstIn, fit.firstIn$fitted.values, col='red')
```

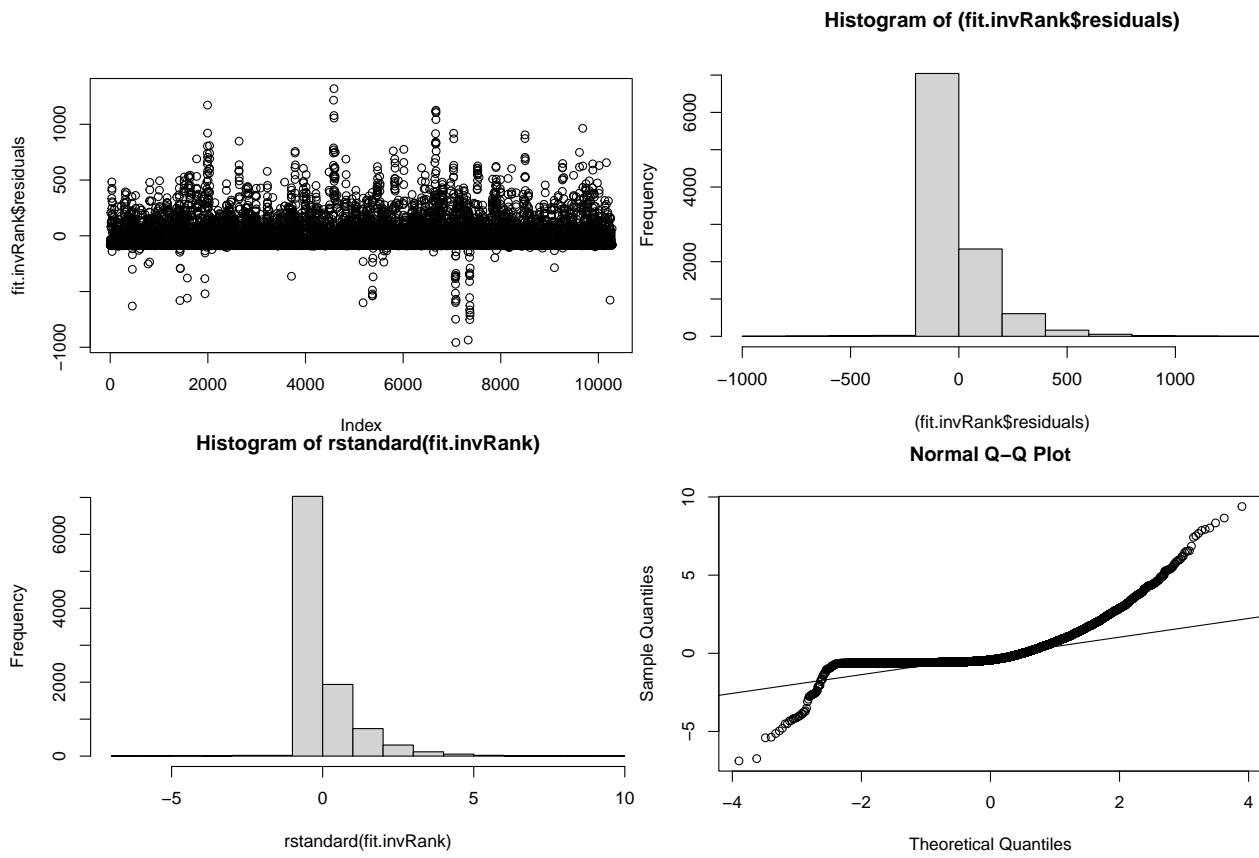


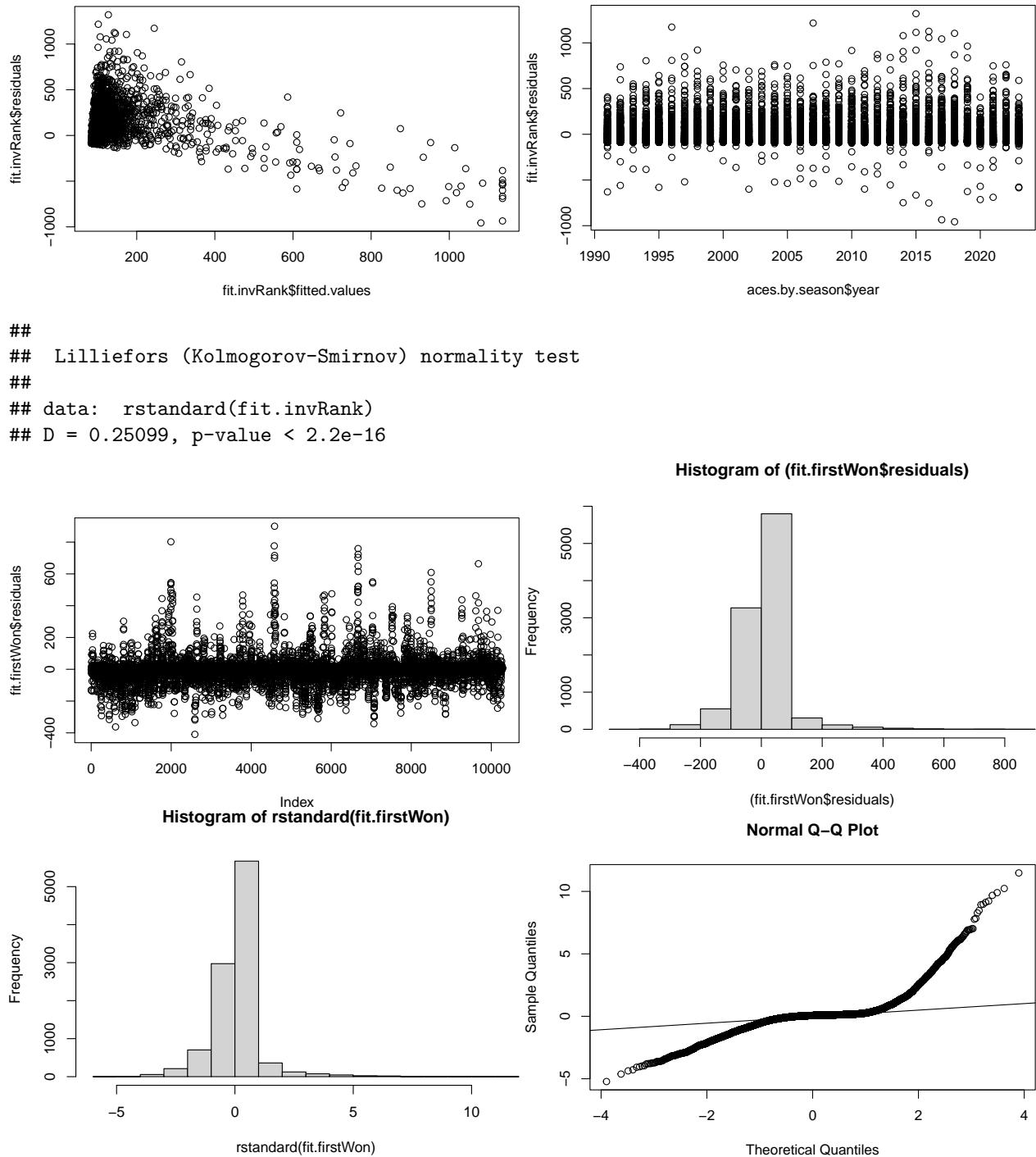
```
plot(aces.by.season$firstWon, aces.by.season$aces)
lines(aces.by.season$firstWon, fit.firstWon$fitted.values, col='red')
```

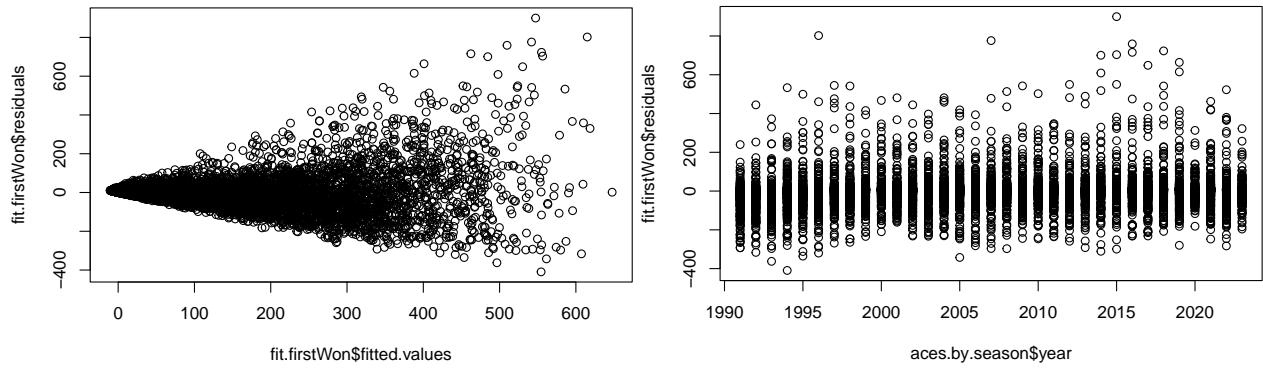


### Normalnost reziduala i homogenost varijance

Normalnost reziduala provjerit ćemo grafički pomoću histograma i qq plota te statistički pomoću Lillieforsovog testa.







```
## 
## Lilliefors (Kolmogorov-Smirnov) normality test
## 
## data: rstandard(fit.firstWon)
## D = 0.23869, p-value < 2.2e-16
```

Iz histograma i qq plota možemo zaključiti da reziduali ukupan broj servi, uspješnih prvih servi i osvojenih prvih servi značajno odstupaju od normalne distribucije dok za inverz ranka reziduali više nalikuju eksponencijalnoj distribuciji nego normalnoj.

Također treba prijetiti da za ukupan broj servi, uspješnih prvih servi i osvojenih prvih servi u ovisnosti o predviđanjima reziduali pokazuju heterogenost varijance, šire se povećanjem  $\hat{y}$ . Međutim u ovisnosti o godinama reziduali pokazuju homogenu varijancu.

### Korelacija značajki

Unatoč neobjećavajućim rezultatima ispitivanja normalnosti i homogenosti varijance reziduala pokušat ćemo napraviti regresiju nad skupom relevantnih značajki. Kako bi to napravili prvo moramo odrediti koje su značajke korelirane.

```
cor(cbind(aces.by.season$aces, aces.by.season$invRank, aces.by.season$svpt, aces.by.season$firstIn, ace
```

```
##      [,1]     [,2]     [,3]     [,4]     [,5]
## [1,] 1.0000000 0.4164105 0.8363657 0.8230129 0.8617121
## [2,] 0.4164105 1.0000000 0.4393532 0.4467202 0.4660817
## [3,] 0.8363657 0.4393532 1.0000000 0.9948097 0.9942303
## [4,] 0.8230129 0.4467202 0.9948097 1.0000000 0.9965806
## [5,] 0.8617121 0.4660817 0.9942303 0.9965806 1.0000000
```

Kao što smo i prepostavili ranije, ukupan broj servi, uspješnih prvih servi i osvojenih prvih servi pokazuju snažnu pozitivnu korelaciju. Stoga ćemo od te tri značajke uzeti samo ukupan broj osvojenih prvih srebi jer ima najveću korelaciju s brojem asova od te tri.

Inverzni rank također pokazuje značajnu korelaciju s ostatkom značajki.

```
fit.multi = lm(aces ~ invRank + firstWon, data=aces.by.season)
summary(fit.multi)
```

```
## 
## Call:
## lm(formula = aces ~ invRank + firstWon, data = aces.by.season)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -406.35  -15.93    5.03   11.35  903.20 
## 
```

```

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10.05495   1.01610 -9.896 < 2e-16 ***
## invRank     47.76983  14.30098  3.340 0.00084 ***
## firstWon    0.18255   0.00121 150.863 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 78.41 on 10279 degrees of freedom
## Multiple R-squared:  0.7428, Adjusted R-squared:  0.7428
## F-statistic: 1.485e+04 on 2 and 10279 DF, p-value: < 2.2e-16

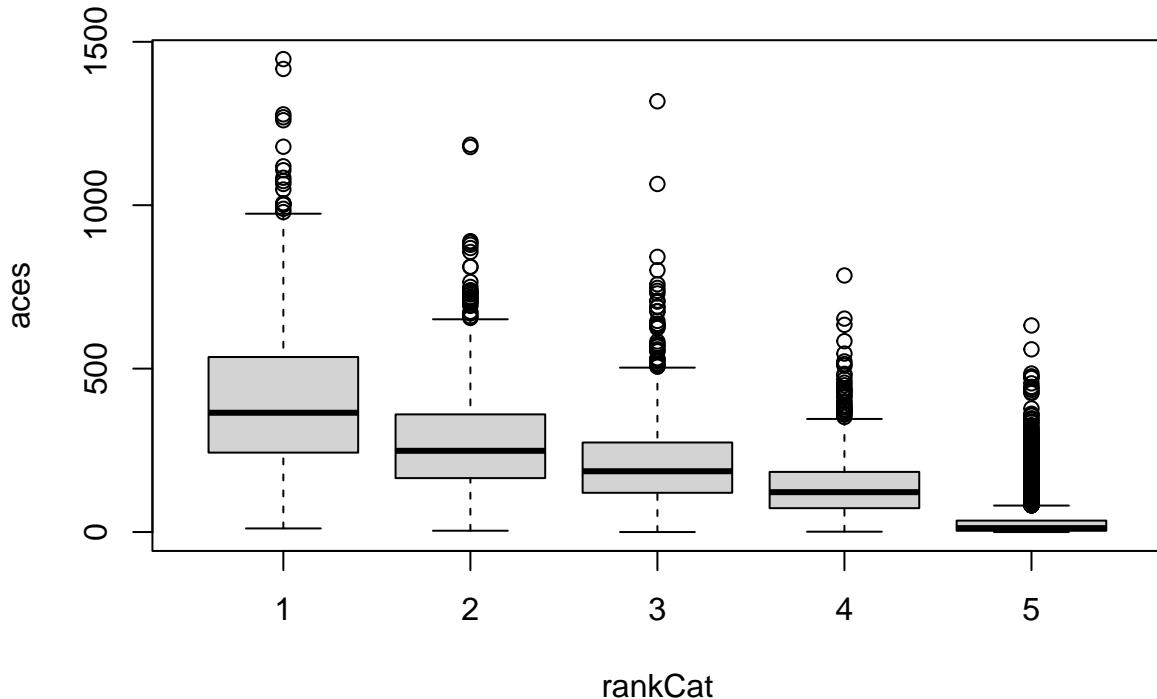
```

Iz modela možemo vidjeti da rank igrača jako slabo utječe na objašnjavanje varijance podataka. Stoga ćemo pokušati pretvoriti rank igrača u kategoričku varijablu na način da svrtamo igrače u sljedeće kategorije: - Rank 1 - 20 - Rank 21 - 50 - Rank 50 - 100 - Rank 100+

```

aces.by.season <- aces.by.season %>%
  mutate(rankCat = case_when(
    rank < 25 ~ 1,
    rank < 50 ~ 2,
    rank < 75 ~ 3,
    rank < 100 ~ 4,
    TRUE ~ 5
  ))
  boxplot(aces~rankCat, data=aces.by.season)

```



```

aces.by.season.d = dummy_cols(aces.by.season, select_columns = 'rankCat')

fit.multi.d = lm(aces ~ firstWon + rankCat_1 + rankCat_2 + rankCat_3 + rankCat_4, data = aces.by.season)
summary(fit.multi.d)

```

```
##
```

```

## Call:
## lm(formula = aces ~ firstWon + rankCat_1 + rankCat_2 + rankCat_3 +
##      rankCat_4, data = aces.by.season.d)
##
## Residuals:
##    Min     1Q Median     3Q    Max 
## -446.70 -15.48   2.97   9.77 862.75 
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -7.875695  1.035250 -7.608 3.04e-14 ***
## firstWon     0.191089  0.002302  83.018 < 2e-16 ***
## rankCat_1    14.459791  5.356223  2.700  0.00695 **  
## rankCat_2   -26.230447  4.436337 -5.913 3.47e-09 ***
## rankCat_3   -31.537385  3.784755 -8.333 < 2e-16 *** 
## rankCat_4   -29.510775  3.162099 -9.333 < 2e-16 *** 
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 77.37 on 10276 degrees of freedom
## Multiple R-squared:  0.7497, Adjusted R-squared:  0.7495 
## F-statistic:  6154 on 5 and 10276 DF,  p-value: < 2.2e-16

```

Ova metoda nažalost također nije dala nikakvo značajno poboljšanje. Međutim postoji još jedan način da poboljšamo rezultate a to je uvođenje nove zakašnjene variabile. Naime pretpostavljamo da će broj aseva za nekog igrača biti takođe povezan s brojem aseva koje je ostavio u prošloj sezoni. Zato ćemo uvesti novu značajku prevAces u naš model.

```

aces.by.season.d$prevAces = c(NA, aces.by.season.d$aces[1:length(aces.by.season.d$aces)-1])

fit.multi.timelag = lm(aces ~ firstWon + rankCat_1 + rankCat_2 + rankCat_3 + rankCat_4 + prevAces, data
summary(fit.multi.timelag)

##
## Call:
## lm(formula = aces ~ firstWon + rankCat_1 + rankCat_2 + rankCat_3 +
##      rankCat_4 + prevAces, data = aces.by.season.d)
##
## Residuals:
##    Min     1Q Median     3Q    Max 
## -378.24 -23.90    7.86   18.12 749.22 
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -2.077e+01  8.288e-01 -25.06  <2e-16 ***
## firstWon     1.674e-01  1.832e-03  91.41  <2e-16 *** 
## rankCat_1   -1.097e+02  4.484e+00 -24.47  <2e-16 *** 
## rankCat_2   -9.625e+01  3.592e+00 -26.80  <2e-16 *** 
## rankCat_3   -6.649e+01  3.004e+00 -22.13  <2e-16 *** 
## rankCat_4   -4.641e+01  2.492e+00 -18.62  <2e-16 *** 
## prevAces     4.498e-01  5.626e-03  79.94  <2e-16 *** 
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.76 on 10274 degrees of freedom

```

```

##      (1 observation deleted due to missingness)
## Multiple R-squared:  0.8457, Adjusted R-squared:  0.8456
## F-statistic:  9382 on 6 and 10274 DF,  p-value: < 2.2e-16

```

Ovime smo dobili osjetno bolji model. Za kraj ćemo još primjeniti model za nekoliko igrača da vidimo kako radi.

```

player.samples <- aces.by.season.d[sample(nrow(aces.by.season.d), 10),]

name <- player.samples$name
year <- player.samples$year
target_aces <- player.samples$aces
prediction_aces <- predict(fit.multi.timelag, newdata = player.samples)

print(data.frame(name, year, target_aces, prediction_aces))

##           name year target_aces prediction_aces
## 1    Pablo Cuevas 2020        142     161.62334
## 2   Shuzo Matsuoka 1997        34     110.58950
## 3 Clinton Marsh 1993         0     -10.05214
## 4   Nicolas Mahut 2014       275     203.39232
## 5     Bjorn Borg 1992        18      11.62724
## 6 Mohamed Safwat 2020        17      2.83920
## 7       Zhe Li 2019        16     14.23366
## 8     Mardy Fish 2011       486     488.51914
## 9    Sandon Stolle 1991        76     78.16530
## 10 Bjorn Fratangelo 2016        72     60.72393

```

Model na žalost ne radi sjajno, ali to je i za očekivati jer su narušeni uvjeti normalnosti i homogenosti varijanci reziduala. Za ovaj problem potreban je složeniji model.