

Analiza podataka o NBA košarkašima

Marko Haralovic

2023-11-06

Učitavanje skupa podataka u radnu okolinu

```
nba_data <- read_csv("../dataset/all_seasons.csv", show_col_types = FALSE)

## New names:
## * ` ` -> `...1`

head(nba_data)

## # A tibble: 6 x 22
##   ...1 player_n~1 team_~2 age playe~3 playe~4 college country draft~5 draft~6
##   <dbl> <chr>    <chr>    <dbl>    <dbl>    <dbl> <chr>    <chr>    <chr>    <chr>
## 1     0 Randy Liv~ HOU      22    193.    94.8 Louisi~ USA     1996     2
## 2     1 Gaylon Ni~ WAS      28    190.    86.2 Northw~ USA     1994     2
## 3     2 George Ly~ VAN      26    203.    103. North ~ USA     1993     1
## 4     3 George Mc~ LAL      30    203.    102. Florid~ USA     1989     1
## 5     4 George Zi~ DEN      23    213.    120. UCLA    USA     1995     1
## 6     5 Gerald Wi~ ORL      33    198.    102. Tennes~ USA     1985     2
## # ... with 12 more variables: draft_number <chr>, gp <dbl>, pts <dbl>,
## #   reb <dbl>, ast <dbl>, net_rating <dbl>, oreb_pct <dbl>, dreb_pct <dbl>,
## #   usg_pct <dbl>, ts_pct <dbl>, ast_pct <dbl>, season <chr>, and abbreviated
## #   variable names 1: player_name, 2: team_abbreviation, 3: player_height,
## #   4: player_weight, 5: draft_year, 6: draft_round
```

Zadaci

Skup sadrži podatke igrača NBA (National Basketball Association) od sezone 1996./1997. do sezone 2022./2023. Neke od varijabli sadrže dob igrača, visinu, težinu, broj zabijenih koševa po sezoni, broj asistencija po sezoni itd.

Istraživačka pitanja:

- Razlikuje li se broj poena igrača po sezoni kroz različita desetljeća?
- Postoji li značajna statistička razlika u visini igrača koji igraju za ekipe zapadne od igrača koji igraju za ekipe istočne konferencije?
- Možemo li predvidjeti prosječni broj poena igrača u sezoni s obzirom na njegove biometrijske podatke?
- Kakva je veza između dobi igrača i prosječnog broja postignutih poena po sezoni?

Osnovni pregled skupa podataka

```
glimpse(nba_data)

## Rows: 12,844
## Columns: 22
## $ ...1          <dbl> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15~
## $ player_name   <chr> "Randy Livingston", "Gaylon Nickerson", "George Lync~
```

```
## $ team_abbreviation <chr> "HOU", "WAS", "VAN", "LAL", "DEN", "ORL", "WAS", "CH~
## $ age <dbl> 22, 28, 26, 30, 23, 33, 26, 30, 24, 24, 22, 31, 29, ~
## $ player_height <dbl> 193.04, 190.50, 203.20, 203.20, 213.36, 198.12, 231.~
## $ player_weight <dbl> 94.80073, 86.18248, 103.41898, 102.05820, 119.74829,~
## $ college <chr> "Louisiana State", "Northwestern Oklahoma", "North C~
## $ country <chr> "USA", "USA", "USA", "USA", "USA", "USA", "USA", "US~
## $ draft_year <chr> "1996", "1994", "1993", "1989", "1995", "1985", "199~
## $ draft_round <chr> "2", "2", "1", "1", "1", "2", "2", "1", "1", "1", "1~
## $ draft_number <chr> "42", "34", "12", "7", "22", "47", "30", "4", "1", "~
## $ gp <dbl> 64, 4, 41, 64, 52, 80, 73, 79, 80, 80, 82, 65, 65, 4~
## $ pts <dbl> 3.9, 3.8, 8.3, 10.2, 2.8, 10.6, 10.6, 26.8, 21.1, 21~
## $ reb <dbl> 1.5, 1.3, 6.4, 2.8, 1.7, 2.2, 6.6, 4.0, 6.3, 9.0, 5.~
## $ ast <dbl> 2.4, 0.3, 1.9, 1.7, 0.3, 2.2, 0.4, 2.0, 3.1, 7.3, 1.~
## $ net_rating <dbl> 0.3, 8.9, -8.2, -2.7, -14.1, -5.8, 6.9, 3.2, -2.9, 6~
## $ oreb_pct <dbl> 0.042, 0.030, 0.106, 0.027, 0.102, 0.031, 0.098, 0.0~
## $ dreb_pct <dbl> 0.071, 0.111, 0.185, 0.111, 0.169, 0.064, 0.217, 0.0~
## $ usg_pct <dbl> 0.169, 0.174, 0.175, 0.206, 0.195, 0.203, 0.185, 0.2~
## $ ts_pct <dbl> 0.487, 0.497, 0.512, 0.527, 0.500, 0.503, 0.618, 0.6~
## $ ast_pct <dbl> 0.248, 0.043, 0.125, 0.125, 0.064, 0.143, 0.024, 0.0~
## $ season <chr> "1996-97", "1996-97", "1996-97", "1996-97", "1996-97~
```

Razlikuje li se broj poena igrača po sezoni kroz različita desetljeća?

Što treba:

- podijeliti skup podataka na manje dijelove po desetljećima - za svako desetljeće sumirati i uprosječiti broj poena po igraču -> naivna opcija bila bi uzimati prosjeke igrača po utakmici po sezoni, metodologija je sljedeća: da ne uklanjamo igrače s malim brojem utakmica (sezona ima 82 utakmice, odrediti neki mekani prag), računanja prosjeka po desetljeću izgleda ovako: - za svaku sezonu u desetljeću, za svakog igrača množiti broj postignutih poena po utakmici s brojem utakmica, sumirati sve poene svih igrača u toj sezoni i tako za svaku sezonu, nakon toga, podijeliti ukupan broj poena s brojem utakmica i brojem igrača ne bi li se dobio prosječan broj poena po utakmici - ima li smisla u statistici uključivati i igrače s malim brojem utakmica (eventualni outlieri)

```
# u novi podatkovni okvir spremamo podatke iz originalnog okvira + stupac desetljeće
nba_data_decade <- nba_data<- nba_data %>%
  mutate(Decade = cut(as.numeric(substr(season, 1, 4)),
    breaks = seq(1990, 2030, by = 10),
    labels = c("1990s", "2000s", "2010s", "2020s"),
    right = FALSE))
head(nba_data_decade)
```

```
## # A tibble: 6 x 23
##   ...1 player_n~1 team~2 age playe~3 playe~4 college country draft~5 draft~6
##   <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr> <chr> <chr> <chr>
## 1 0 Randy Liv~ HOU 22 193. 94.8 Louisi~ USA 1996 2
## 2 1 Gaylon Ni~ WAS 28 190. 86.2 Northw~ USA 1994 2
## 3 2 George Ly~ VAN 26 203. 103. North ~ USA 1993 1
## 4 3 George Mc~ LAL 30 203. 102. Florid~ USA 1989 1
## 5 4 George Zi~ DEN 23 213. 120. UCLA USA 1995 1
## 6 5 Gerald Wi~ ORL 33 198. 102. Tennes~ USA 1985 2
## # ... with 13 more variables: draft_number <chr>, gp <dbl>, pts <dbl>,
## # reb <dbl>, ast <dbl>, net_rating <dbl>, oreb_pct <dbl>, dreb_pct <dbl>,
## # usg_pct <dbl>, ts_pct <dbl>, ast_pct <dbl>, season <chr>, Decade <fct>, and
## # abbreviated variable names 1: player_name, 2: team_abbreviation,
## # 3: player_height, 4: player_weight, 5: draft_year, 6: draft_round
```

```
#kreiranje globalnih varijabli preko kojih ćemo računati tražene podatke za zadatak
total_points_decade <- numeric(length(unique(nba_data_decade$Decade)))
print(total_points_decade)
names(total_points_decade) <- unique(nba_data_decade$Decade)
print(names(total_points_decade))
total_games_decade <- numeric(length(unique(nba_data_decade$Decade)))
print(total_games_decade)
names(total_games_decade) <- unique(nba_data_decade$Decade)
print(names(total_games_decade))
```

```
## [1] 0 0 0 0
## [1] "1990s" "2000s" "2010s" "2020s"
## [1] 0 0 0 0
## [1] "1990s" "2000s" "2010s" "2020s"
```

```
# sumiranje i uprosječivanje poena po igraču za svaku sezonu za desetljeće
# stupac pts nosi informacije o poenima po utakmici, zaokružene na dvije decimale: što dovodi do problema
```

```
nba_data_decade <- nba_data_decade %>%
  group_by(player_name, season) %>%
  mutate(TotalPointsSeason = floor(gp * pts)) %>%
  ungroup()

print(head(nba_data_decade$TotalPointsSeason))
```

```
## [1] 249 15 340 652 145 848
```

```
# za svakog igrača odredimo ukupan broj poena i kupan broj utakmica u desetljeću
```

```
points_by_decade_player <- nba_data_decade %>%
  group_by(Decade, player_name) %>%
  summarize(TotalPoints = sum(TotalPointsSeason, na.rm = TRUE),
            TotalGames = sum(gp, na.rm = TRUE), .groups = 'drop')

print(head(points_by_decade_player))
```

```
## # A tibble: 6 x 4
##   Decade player_name      TotalPoints TotalGames
##   <fct>   <chr>          <dbl>         <dbl>
## 1 1990s   A.C. Green           1850           297
## 2 1990s   A.J. Bramlett          8             8
## 3 1990s   Aaron McKie           1659           296
## 4 1990s   Aaron Williams         1278           219
## 5 1990s   Acie Earl              188            47
## 6 1990s   Adam Keefe             1171           248
```

```
total_points_decade <- sapply(levels(points_by_decade_player$Decade), function(decade) {
  sum(points_by_decade_player$TotalPoints[points_by_decade_player$Decade == decade], na.rm = TRUE)
})
```

```
print(total_points_decade)
```

```
total_games_decade <- sapply(levels(points_by_decade_player$Decade), function(decade) {
  sum(points_by_decade_player$TotalGames[points_by_decade_player$Decade == decade], na.rm = TRUE)
})
```

```
print(total_games_decade)
```

```
##    1990s    2000s    2010s    2020s
## 821680 2358920 2453891 795885
## 1990s    2000s    2010s    2020s
## 86964 244989 250084 74987
```

```
#prosječni broj poena po utakmici
```

```
average_points_per_game_decade <- total_points_decade / total_games_decade
print(average_points_per_game_decade)
```

```
##    1990s    2000s    2010s    2020s
## 9.448507 9.628677 9.812267 10.613640
```

```
decade_summary <- data.frame(
  Decade = names(total_points_decade),
  TotalPoints = total_points_decade,
  TotalGames = total_games_decade
) %>%
  mutate(AveragePointsPerGame = TotalPoints / TotalGames)
```

```
print(decade_summary)
```

```
##      Decade TotalPoints TotalGames AveragePointsPerGame
## 1990s 1990s      821680      86964          9.448507
## 2000s 2000s     2358920     244989          9.628677
## 2010s 2010s     2453891     250084          9.812267
## 2020s 2020s     795885      74987         10.613640
```

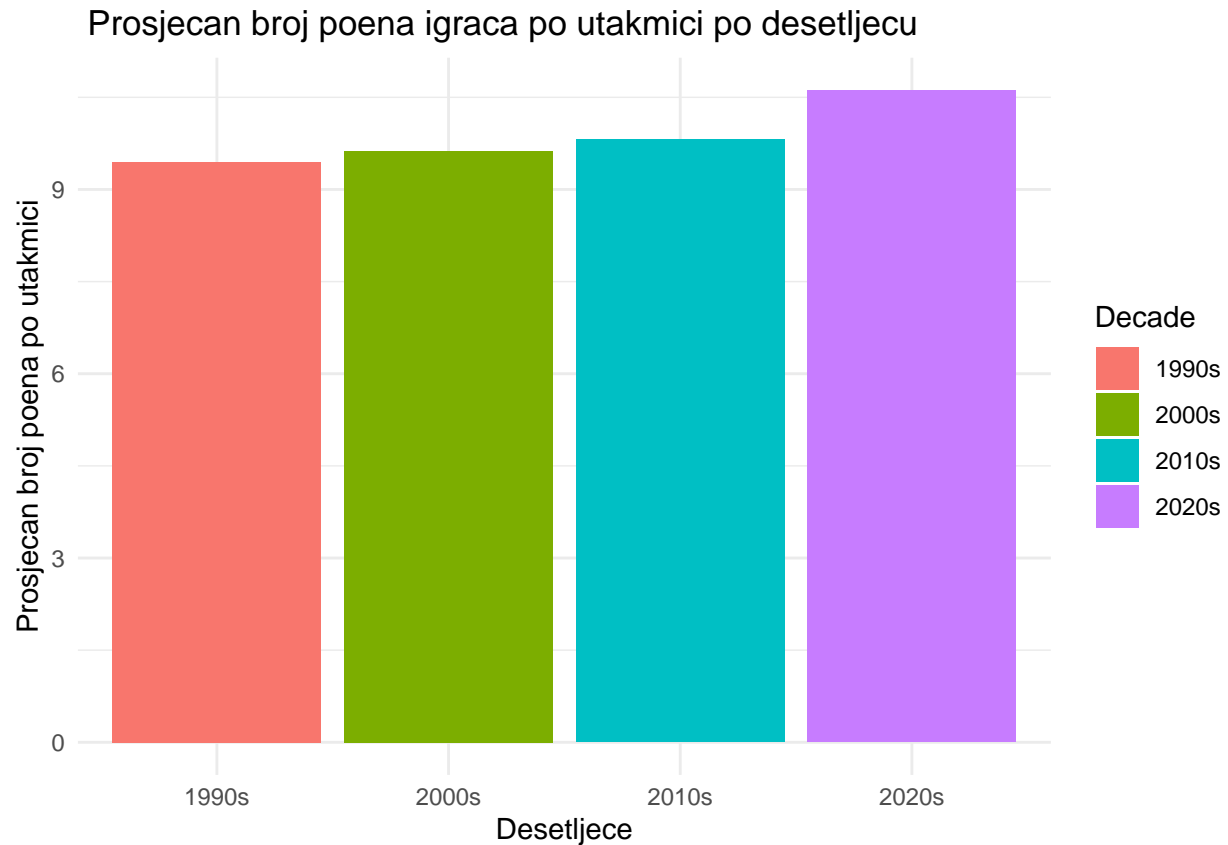
```
# ANOVA, nismo učili, ali sam našao kako raditi pa ostavljam kao opciju
```

```
# služi za usporedbu srednjih vrijednosti kontinuirane varijable, ovdje služi da vidimo razlikuju li s
```

```
anova_result <- aov(AveragePointsPerGame ~ Decade, data = decade_summary)
summary(anova_result)
```

```
##      Df Sum Sq Mean Sq
## Decade    3 0.7921  0.264
```

```
ggplot(decade_summary, aes(x = Decade, y = AveragePointsPerGame, fill = Decade)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "Prosječan broj poena igrača po utakmici po desetljeću",
       x = "Desetljeće",
       y = "Prosječan broj poena po utakmici")
```



ODGOVOR NA ZPITANJE 1:

Postoji li značajna statistička razlika u visini igrača koji igraju za ekipe zapadne od igrača koji igraju za ekipe istočne konferencije

```
#istočna i zapadna konferekncija imaju 15 timova, ovdje svakom igraču na temelju tima određujem konfere
nba_data <- nba_data %>%
  mutate(conference = case_when(
    team_abbreviation %in% c("BOS", "BKN", "NYK", "PHI", "TOR", "CHI", "CLE", "DET", "IND", "MIL", "ATL", "MIA", "ORL", "WAS", "WOL") ~ "East",
    team_abbreviation %in% c("DEN", "MIN", "OKC", "POR", "UTA", "GSW", "LAC", "LAL", "PHX", "SAC", "DAL", "MEM", "NOP", "SAS", "HOU", "IND", "CLE", "DET", "MIL", "ATL", "MIA", "ORL", "WAS", "WOL") ~ "West",
    TRUE ~ NA_character_
  ))
```

```
# izvođenje t-testa da vidimo postoji li značajna razlik
t_test_result <- t.test(player_height ~ conference, data = nba_data)

print(t_test_result)
```

```
##
## Welch Two Sample t-test
##
## data: player_height by conference
## t = 0.78218, df = 12794, p-value = 0.4341
## alternative hypothesis: true difference in means between group East and group West is not equal to 0
## 95 percent confidence interval:
## -0.1894923 0.4411384
## sample estimates:
```

```
## mean in group East mean in group West
##      200.6174      200.4916
```

ZAKLJUČAK : ###

Kakva je veza između dobi igrača i prosječnog broja postignutih poena po sezoni?

Gledati ću prosječan broj postignutih poena po utakmici kao i prosječan broj odigranih utakmica igrača po godini, grupiranih po godinama. Za najboljih 100 strijelaca svake godine izvaditi ću statistiku o prosječnoj dobi i broju sezona koje su u prosjeku provedene u NBA-u. Nije svaki igrač bio draftom izabran u ligu, tako da grupiram podatkovni okvir i izvlačim informaciju o prvoj godini igranja u ligi za svakog igrača.

```
nba_data <- nba_data %>%
  mutate(season_start_year = as.integer(sub("-", "", season)))

# podatkovni okvir s prvog godinom igranja svakog igrača
players_first_year <- nba_data %>%
  group_by(player_name) %>%
  summarise(first_year = min(season_start_year))
#head(players_first_year)

nba_data <- nba_data %>%
  left_join(players_first_year, by = "player_name")

top_scorers_average_age_exp <- nba_data %>%
  mutate(total_points = pts * gp,
         years_in_league = season_start_year - first_year + 1 # izračun broja godina u ligi
        ) %>%
  group_by(season) %>%
  top_n(100, total_points) %>%
  summarise(average_age = mean(age, na.rm = TRUE),
            average_experience = mean(years_in_league, na.rm = TRUE)) # srednja vrijednost broja godina

print(top_scorers_average_age_exp)
```

```
## # A tibble: 27 x 3
##   season average_age average_experience
##   <chr>      <dbl>          <dbl>
## 1 1996-97      27.9            1
## 2 1997-98      27.9           1.94
## 3 1998-99      28.4            2.8
## 4 1999-00      27.6           3.49
## 5 2000-01      27.3           4.24
## 6 2001-02      27.6           4.77
## 7 2002-03      27.0           5.15
## 8 2003-04      26.6           5.43
## 9 2004-05      27.1           5.91
## 10 2005-06     26.9           6.05
## # ... with 17 more rows
```

Ovdje je podatkovni okvir bez iskustva, za prvih nekoliko godina nema smisla jer se nisu skupljali raniji podaci, ups

```
top_scorers_average_age <- nba_data %>%
  mutate(total_points = pts * gp) %>% # Calculate total points for the season
```

```

group_by(season) %>%
top_n(100, total_points) %>%
summarise(average_age = mean(age, na.rm = TRUE))

print(top_scorers_average_age)

```

```

## # A tibble: 27 x 2
##   season average_age
##   <chr>      <dbl>
## 1 1996-97      27.9
## 2 1997-98      27.9
## 3 1998-99      28.4
## 4 1999-00      27.6
## 5 2000-01      27.3
## 6 2001-02      27.6
## 7 2002-03      27.0
## 8 2003-04      26.6
## 9 2004-05      27.1
## 10 2005-06      26.9
## # ... with 17 more rows

```

Vizualizacija po godinama prosječne dobi najboljih 100 strijelaca

```

top_scorers_average_age <- top_scorers_average_age %>%
  mutate(season_start_year = as.numeric(sub("-", ".", season))) # Extract start year of the season as

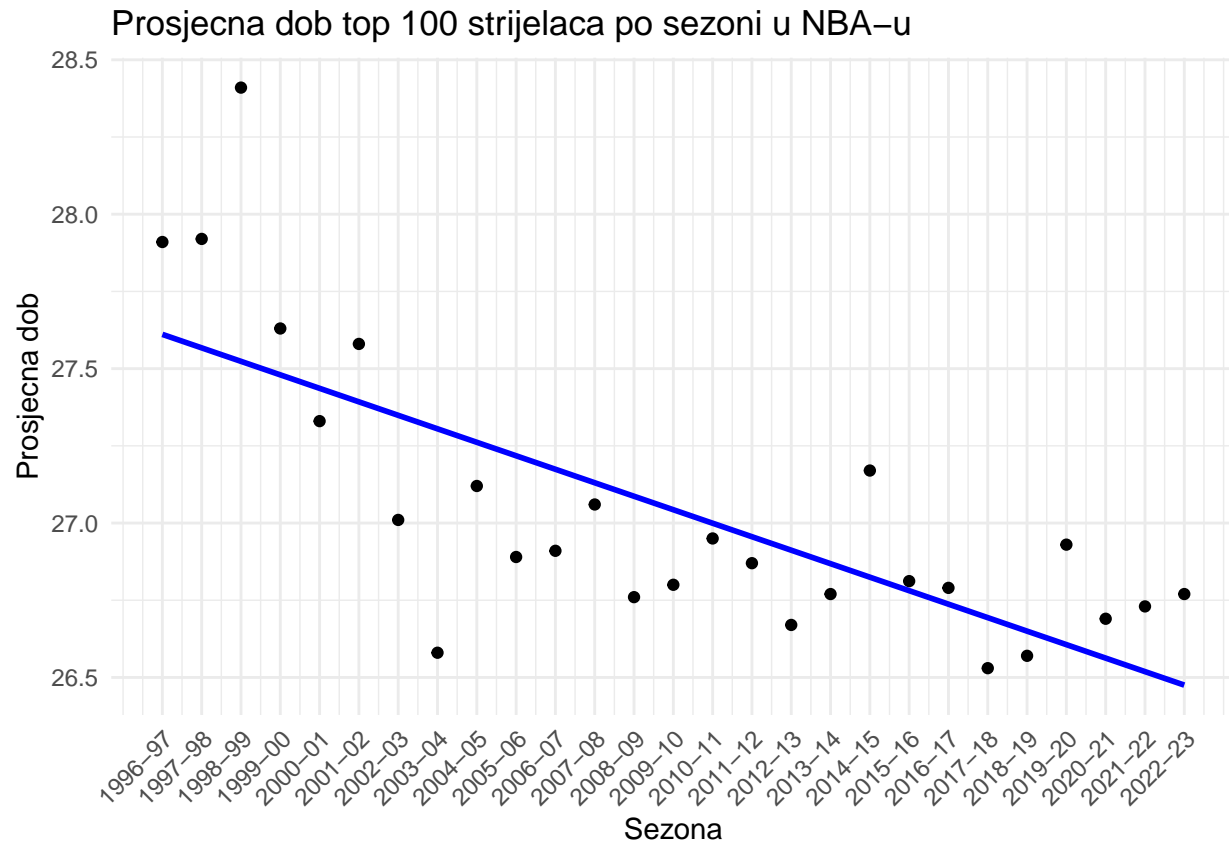
# Plot
ggplot(top_scorers_average_age, aes(x = season_start_year, y = average_age)) +
  geom_point() + #
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  theme_minimal() +
  labs(x = "Sezona", y = "Prosječna dob", title = "Prosječna dob top 100 strijelaca po sezoni u NBA-u")
  scale_x_continuous(breaks = top_scorers_average_age$season_start_year, labels = top_scorers_average_age$season_start_year)
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

```

```

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

```



Sada gledam prosječan broj postignutih poena po starosti igrača. Odrediti ću minimalnu i maksimalnu dob cijele csv datoteke, za svaku godinu dodavati poene i utakmice, izračunati prosječan broj poena igrača ovisno o njegovoj starosti.

```
points_by_age <- nba_data %>%
  group_by(age) %>%
  summarize(TotalPoints = sum(gp * pts, na.rm = TRUE),
            TotalGames = sum(gp, na.rm = TRUE), .groups = 'drop')
print(head(points_by_age))
```

```
## # A tibble: 6 x 3
##   age TotalPoints TotalGames
##   <dbl>      <dbl>      <dbl>
## 1    18         798.         162
## 2    19      29475.       3553
## 3    20     143367.      15864
## 4    21     260154.      27379
## 5    22     389562.      40309
## 6    23     511265.      55768
```

```
total_points_age <- points_by_age$TotalPoints
total_games_age <- points_by_age$TotalGames

average_points_per_game_age <- total_points_age / total_games_age

age_summary <- data.frame(
  Age = points_by_age$age,
  TotalPoints = total_points_age,
```



```

    TotalGames = total_games_age,
    AveragePointsPerGame = average_points_per_game_age
)

print(age_summary)

```

```

##      Age TotalPoints TotalGames AveragePointsPerGame
## 1    18      797.7        162      4.924074
## 2    19     29475.2       3553     8.295863
## 3    20    143367.1      15864     9.037260
## 4    21    260154.5     27379     9.501972
## 5    22    389561.8     40309     9.664388
## 6    23    511265.1     55768     9.167714
## 7    24    596214.3     62668     9.513856
## 8    25    602888.7     59642    10.108459
## 9    26    584614.5     55901    10.458033
## 10   27    574892.3     53864    10.673034
## 11   28    525864.8     48932    10.746849
## 12   29    463260.1     44430    10.426741
## 13   30    418328.2     41060    10.188217
## 14   31    339755.1     34839     9.752148
## 15   32    278378.3     29462     9.448724
## 16   33    227657.9     24412     9.325655
## 17   34    168837.1     18856     8.954025
## 18   35    116815.5     13761     8.488882
## 19   36     84199.0     10402     8.094501
## 20   37     58189.5      7234     8.043890
## 21   38     30626.4      4237     7.228322
## 22   39     17967.3      2466     7.286010
## 23   40      8394.4      1196     7.018729
## 24   41      1997.4       346     5.772832
## 25   42       798.9       157     5.088535
## 26   43       502.6       119     4.223529
## 27   44        12.0        5     2.400000

```

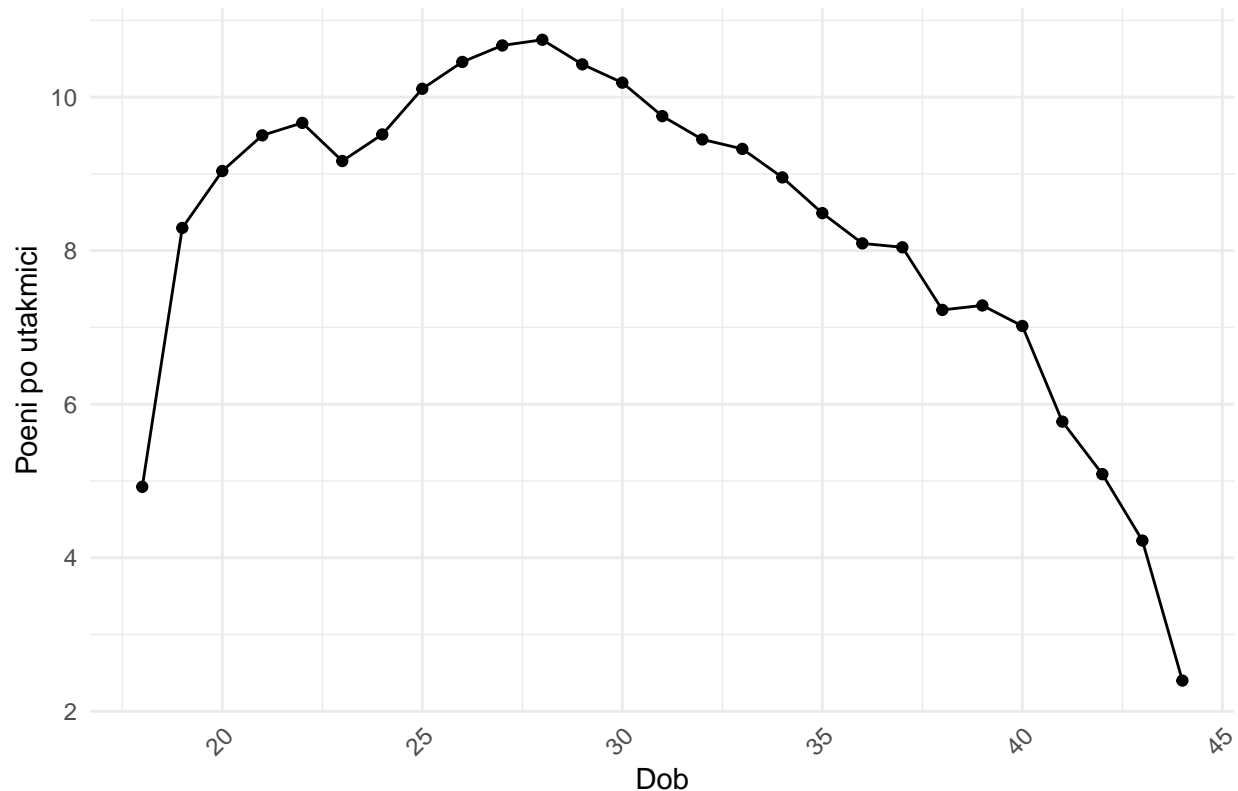
Opet ide vizualizacija, sada ćemo vizualizirati ovisnost o starosti igrača i prosjeku poena po utakmici. Na x osi nalaze se podaci o godinama igrača, a na y osi nalaze se prosjeci poena po utakmici po godinama.

```

ggplot(age_summary, aes(x = Age, y = AveragePointsPerGame)) +
  geom_point() +
  geom_line() +
  theme_minimal() +
  labs(x = "Dob", y = "Poeni po utakmici", title = "Prosjeck poena po utakmici ovisno o dobi igrača") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

```

Prosjeck poena po utakmici ovisno o dobi igrača



```
library(ggplot2)
```

Možemo li predvidjeti prosječni broj poena igrača u sezoni s obzirom na njegove biometrijske podatke

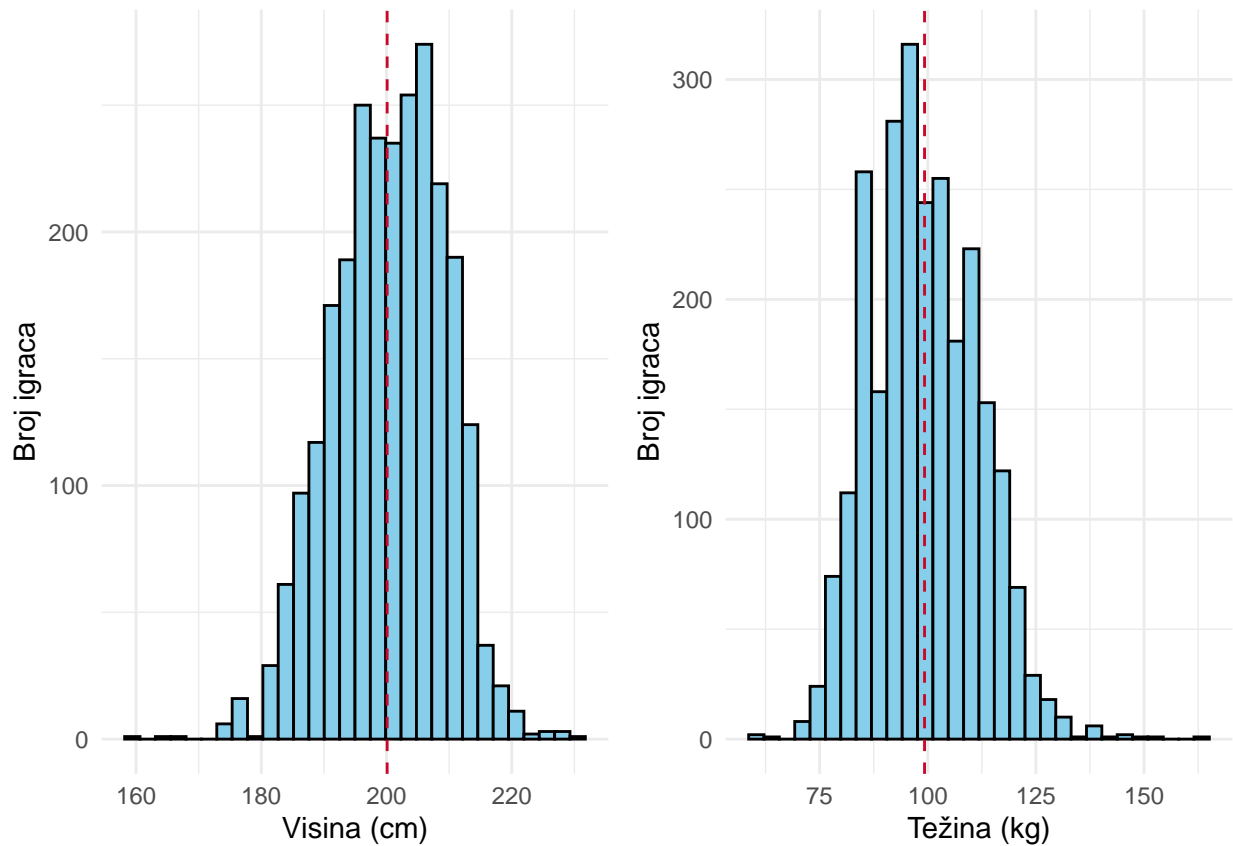
Pretpostavka je da su niži igrači u prednosti da budu najbolji u ligi, kako mogu izbjeći sve visoke, nespretno frajere koji im žele zaljepiti bananu. Bismo vidjeli je li pretpostavka opravdana. Teži frajeri bi trebali skakati bolje pa očekujem više skokova po utakmici

```
težine_visine <- nba_data %>%  
  group_by(player_name) %>%  
  summarise(player_height = mean(player_height, na.rm = TRUE),  
            player_weight = mean(player_weight, na.rm = TRUE))
```

Vizualizacija težina i visina igrača

```
p1 <- ggplot(težine_visine, aes(x = player_height)) +  
  geom_histogram(bins = 30, fill = "skyblue", color = "black") +  
  geom_vline(aes(xintercept = mean(player_height, na.rm = TRUE)), color = '#c9082a', linetype = "dashed") +  
  labs(y = "Broj igrača", x = "Visina (cm)") +  
  theme_minimal()  
  
p2 <- ggplot(težine_visine, aes(x = player_weight)) +  
  geom_histogram(bins = 30, fill = "skyblue", color = "black") +  
  geom_vline(aes(xintercept = mean(player_weight, na.rm = TRUE)), color = '#c9082a', linetype = "dashed") +  
  labs(y = "Broj igrača", x = "Težina (kg)") +  
  theme_minimal()
```

```
grid.arrange(p1, p2, ncol = 2)
```

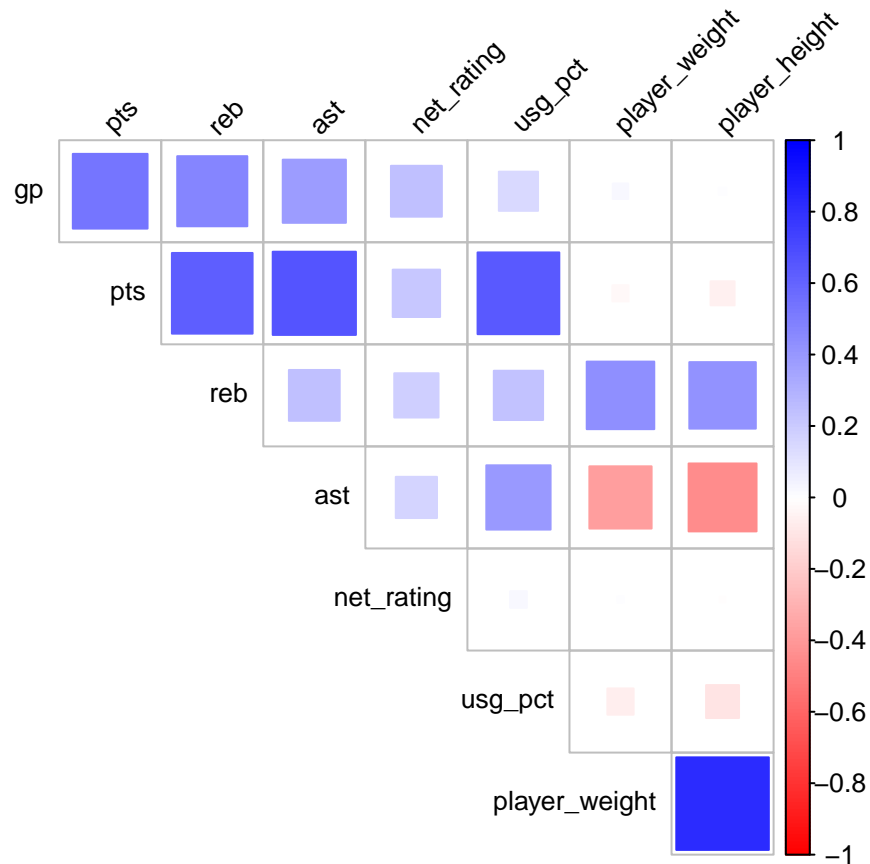


Vizualizacija korelacijske matrice između značajnih varijabli

```
df_corr <- nba_data %>%
  filter(season != '2019-01-01') %>%
  select(gp, pts, reb, ast, net_rating, usg_pct, player_weight, player_height)

corr <- cor(df_corr, use = "complete.obs")

corrplot(corr, method = "square", type = "upper", tl.cex = 0.8,
  tl.col = "black", tl.srt = 45, diag = FALSE,
  col = colorRampPalette(c("red", "white", "blue"))(200))
```



Vizualizacija ovisnosti broja poena,

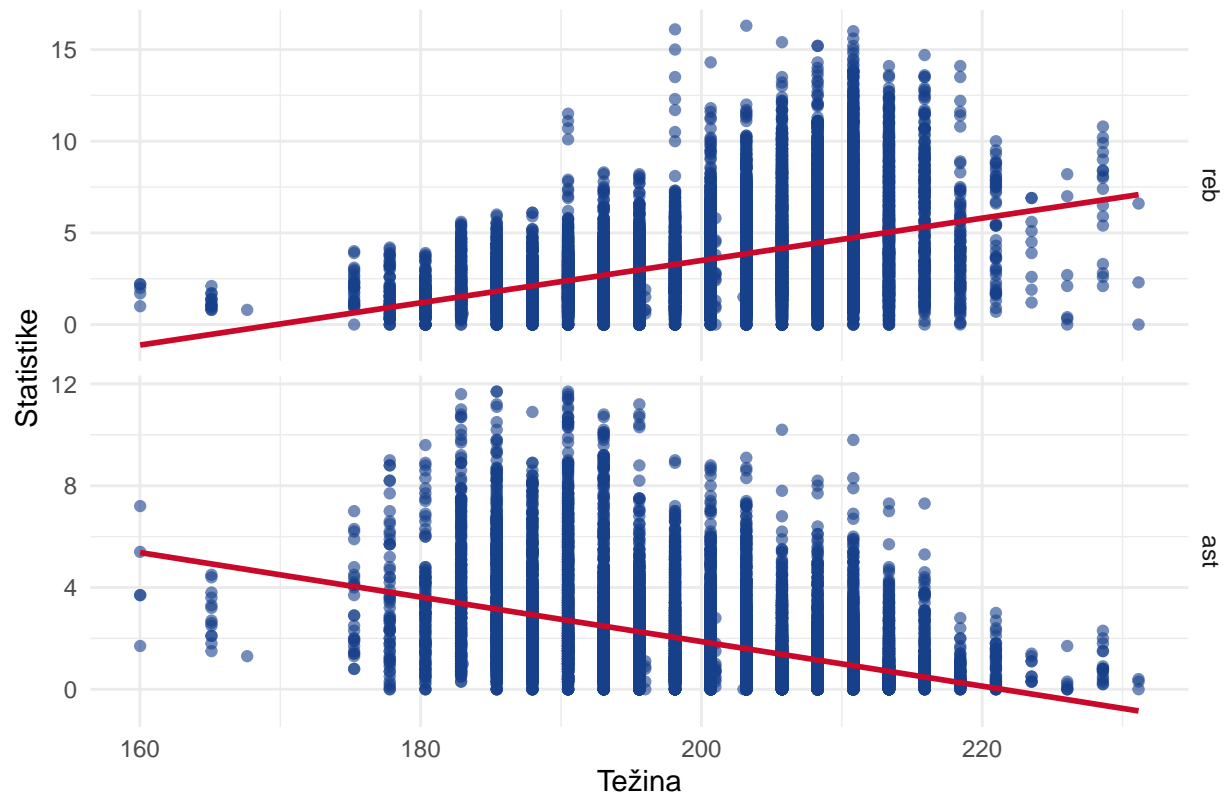
```
# Melt the dataframe
df_melted <- reshape2::melt(df_corr, id.vars = c("player_height", "player_weight"))

df_filtered <- df_melted %>%
  filter(variable %in% c("reb", "ast"))

ggplot(df_filtered, aes(x = player_height, y = value)) +
  geom_point(color = '#17408b', alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE, color = '#c9082a') +
  facet_grid(variable ~ ., scales = "free") +
  theme_minimal() +
  labs(x = "Težina", y = "Statistika") +
  theme(strip.text.x = element_text(size = 12)) +
  ggtitle("Odnos visine i skokova te asistencija")

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

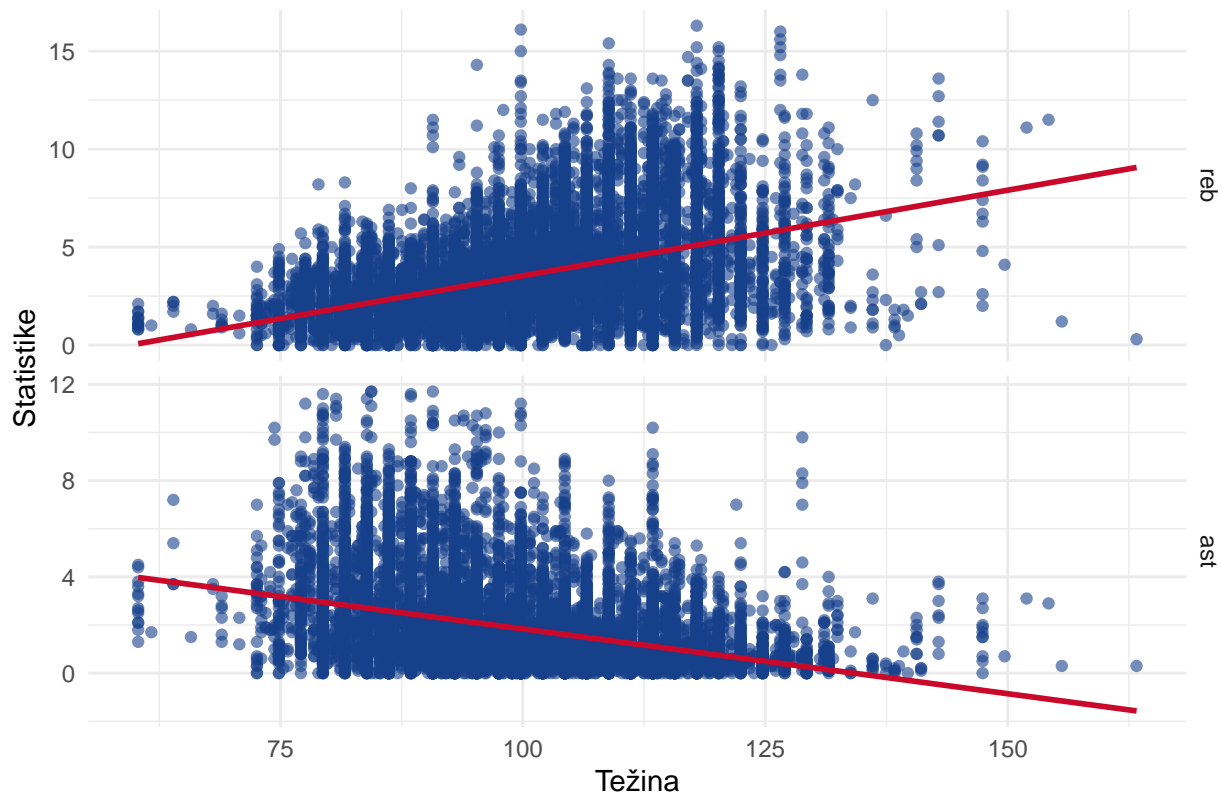
Odnos visine i skokova te asistencija



```
ggplot(df_filtered, aes(x = player_weight, y = value)) +
  geom_point(color = '#17408b', alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE, color = '#c9082a') +
  facet_grid(variable ~ ., scales = "free") +
  theme_minimal() +
  labs(x = "Težina", y = "Statistike") +
  theme(strip.text.x = element_text(size = 12)) +
  ggtitle("Odnos težine i skokova te asistencija")
```

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

Odnos težine i skokova te asistencija



Gledamo poene po tekmi ovisno o visini i težini igrača

```
#sempiranje, djelim podatke na trening i validaciju
split <- sample.split(nba_data$pts, SplitRatio = 0.7)
training_set <- subset(nba_data, split == TRUE)
testing_set <- subset(nba_data, split == FALSE)

#lagana linearna regresijica braco
model <- lm(pts ~ player_height + player_weight, data = training_set)

summary(model)

#baci predviđanja šefe
predictions <- predict(model, newdata = testing_set)

# da vidimo da model nije pijan
actual_vs_predicted <- data.frame(Actual = testing_set$pts, Predicted = predictions)
#print(actual_vs_predicted[0:5][:])

MSE <- mean((predictions - testing_set$pts)^2)
print(MSE)
#ništa ne valja :(

##
## Call:
## lm(formula = pts ~ player_height + player_weight, data = training_set)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.941 -4.601 -1.507  3.219 27.543
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  19.568010   1.780928  10.988 < 2e-16 ***
## player_height -0.072087   0.012131  -5.943 2.91e-09 ***
## player_weight  0.030941   0.008851   3.496 0.000475 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.014 on 8980 degrees of freedom
## Multiple R-squared:  0.00455,    Adjusted R-squared:  0.004329
## F-statistic: 20.52 on 2 and 8980 DF,  p-value: 1.278e-09
##
## [1] 35.76403
```

Objekt klase lm

Ovaj objekt sadržava ne samo koeficijente, već i bogati skup informacija vezanih uz stvoreni linearni model, što uključuje čak i sam podatkovni skup pomoću kojeg je model stvoren.

Nad ovim objektom možemo izvesti sljedeće funkcije:

coef - vraća koeficijente u obliku vektora

fitted.values - vraća vektor predikcijavdobiven primjenom modela na skup za treniranje

residuals - vraća vektor grešaka dobiven primjenom modela na skup za treniranje

summary - daje sažetak najvažnijih informacija o modelu