Tennis Surface Report

Siddharth Gowda

7/26/2020

Introduction

It's well known that different tennis court surfaces have unique properties, and those properties affect certain aspects of the game differently. Thus, for this project we analyzed singles ATP match data from 2019, and performed analysis, comparing data from each surface type.

Background

Below is a data frame describing the general differences about each surface.

```
surface_characteristics =
  read.csv("~/Desktop/Moneyball/Data/surfaces_elements.csv")
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## incomplete final line found by readTableHeader on '~/Desktop/Moneyball/Data/
## surfaces_elements.csv'
```

```
head(surface_characteristics)
```

```
##   Surface     Speed    Spin       Strategy Risk.Factor
## 1    Clay    Slower Topspin       Tactical   Less Risk
## 2    Hard Semi-fast    Flat        Mixture Higher Risk
## 3   Grass Very Fast   Slice Serve & Volley Higher Risk
```

What we mean by speed: Speed describes the pace of the game. Generally on clay courts points last longer and baseline play is used more often than compared to other courts. This is followed by hard court and then grass, where numerous players serve and volley, leading to points ending relatively quickly.

What we mean by spin: The spin variable describes the type of spin that is generally accosted with the surface type. In reality all shots of these spin types are used across all surfaces.

What we mean by strategy: Strategy measures the common methodology players incorporate on each surface. Obviously, this isn't an exact science because players will want to play to their strengths and their opponents weakness. Nevertheless, generally players are more tactical on clay surfaces, increasing the variety in there shot selection. Hard court is a mixture of strategy types, but players do tend to brute-force and out-muscle opponents with powerful shots; grass strategy on the other hand is typically serve and volley.

What we mean by risk-factor: Risk-factor measures amount of risk (or margin of error) players allow on there strokes. For instance, when a player is trying to hit groundstroke down the line, if they aim exactly for the singles-ally line that would indicate a lower margin of error (or more risk), whereas if the player aimed

a couple feet inside of the line that would be a shot of a higher margin of error (or less risk). Typically, players take riskier shots on grass and hard courts when compared with clay.

Extra Notes: In reality every surface type uses different types of tactics, spin, risk, and speed (it mostly depends on the player); however, we are describing each surface in the simplest manner. Also, we are only going to be analyzing the major surface types (clay, grass, and hard), not specifics (i.e. green clay vs red clay).

Loading Data

For this analysis we used 2019 ATP singles match data via Jeff Sackmann on Github.

```
atp_matchies_2019_OG = read.csv("./Data/atp_matches_2019.csv")


head(atp_matchies_2019_OG)
```

```
##   tourney_id tourney_name surface draw_size tourney_level tourney_date
## 1  2019-M020     Brisbane    Hard        32             A     20181231
## 2  2019-M020     Brisbane    Hard        32             A     20181231
## 3  2019-M020     Brisbane    Hard        32             A     20181231
## 4  2019-M020     Brisbane    Hard        32             A     20181231
## 5  2019-M020     Brisbane    Hard        32             A     20181231
## 6  2019-M020     Brisbane    Hard        32             A     20181231
##   match_num winner_id winner_seed winner_entry        winner_name winner_hand
## 1       300    105453           2                   Kei Nishikori           R
## 2       299    106421           4                 Daniil Medvedev           R
## 3       298    105453           2                   Kei Nishikori           R
## 4       297    104542                       PR Jo-Wilfried Tsonga           R
## 5       296    106421           4                 Daniil Medvedev           R
## 6       295    104871                             Jeremy Chardy           R
##   winner_ht winner_ioc winner_age loser_id loser_seed loser_entry
## 1       178        JPN   29.00479   106421          4
## 2        NA        RUS   22.88569   104542                     PR
## 3       178        JPN   29.00479   104871
## 4       188        FRA   33.70568   200282          7
## 5        NA        RUS   22.88569   105683          5
## 6       188        FRA   31.88227   106034                      Q
##           loser_name loser_hand loser_ht loser_ioc loser_age          score
## 1    Daniil Medvedev          R       NA       RUS  22.88569    6-4 3-6 6-2
## 2 Jo-Wilfried Tsonga          R      188       FRA  33.70568     7-6(6) 6-2
## 3      Jeremy Chardy          R      188       FRA  31.88227        6-2 6-2
## 4     Alex De Minaur          R       NA       AUS  19.86858     6-4 7-6(2)
## 5       Milos Raonic          R      196       CAN  28.01095 6-7(2) 6-3 6-4
## 6  Yasutaka Uchiyama          R       NA       JPN  26.40383 6-4 3-6 7-6(4)
##   best_of round minutes w_ace w_df w_svpt w_1stIn w_1stWon w_2ndWon w_SvGms
## 1       3     F     124     3    3     77      44       31       17      13
## 2       3    SF      82    10    1     52      33       28       14      10
## 3       3    SF      66     2    2     47      33       26        9       8
## 4       3    QF     106    12    2     68      43       34       15      11
## 5       3    QF     129    12    3    105      68       48       25      16
## 6       3    QF     127    10    8     94      58       44       18      16
##   w_bpSaved w_bpFaced l_ace l_df l_svpt l_1stIn l_1stWon l_2ndWon l_SvGms
```

```
## 1         3         6     8    6    100      54       34       20      14
## 2         0         1    17    2     77      52       36        7      10
## 3         2         2    10    3     46      27       15        6       8
## 4         4         5     1    2     81      60       38        9      11
## 5         8         8    29    5     94      56       46       19      15
## 6         4         8    12    6     90      54       40       18      15
##   l_bpSaved l_bpFaced winner_rank winner_rank_points loser_rank
## 1        10        15           9               3590         16
## 2        10        13          16               1977        239
## 3         1         5           9               3590         40
## 4         4         6         239                200         31
## 5         2         4          16               1977         18
## 6         6         9          40               1050        185
##   loser_rank_points
## 1              1977
## 2               200
## 3              1050
## 4              1298
## 5              1855
## 6               275
```

Data Cleaning

```
atp_data_rough = clean_names(atp_matchies_2019_OG) %>%
  mutate(total_games = l_sv_gms + w_sv_gms,
         w_bp_att = l_bp_faced, l_bp_att = w_bp_faced,
         w_total_pts = w_svpt + l_svpt,
         l_total_pts = w_svpt + l_svpt) %>%
  select(tourney_name, surface, tourney_level,
         winner_name, winner_hand, winner_ht, winner_age, c(19:21, 23:46, 48),
         winner_rank_points, loser_rank_points, winner_seed, loser_seed,
         total_games, w_bp_att, l_bp_att,  w_total_pts, l_total_pts)

head(atp_data_rough)
```

```
##   tourney_name surface tourney_level        winner_name winner_hand winner_ht
## 1     Brisbane    Hard             A      Kei Nishikori           R       178
## 2     Brisbane    Hard             A    Daniil Medvedev           R        NA
## 3     Brisbane    Hard             A      Kei Nishikori           R       178
## 4     Brisbane    Hard             A Jo-Wilfried Tsonga           R       188
## 5     Brisbane    Hard             A    Daniil Medvedev           R        NA
## 6     Brisbane    Hard             A      Jeremy Chardy           R       188
##   winner_age         loser_name loser_hand loser_ht loser_age          score
## 1   29.00479    Daniil Medvedev          R       NA  22.88569    6-4 3-6 6-2
## 2   22.88569 Jo-Wilfried Tsonga          R      188  33.70568     7-6(6) 6-2
## 3   29.00479      Jeremy Chardy          R      188  31.88227        6-2 6-2
## 4   33.70568     Alex De Minaur          R       NA  19.86858     6-4 7-6(2)
## 5   22.88569       Milos Raonic          R      196  28.01095 6-7(2) 6-3 6-4
## 6   31.88227  Yasutaka Uchiyama          R       NA  26.40383 6-4 3-6 7-6(4)
##   best_of round minutes w_ace w_df w_svpt w_1st_in w_1st_won w_2nd_won w_sv_gms
## 1       3     F     124     3    3     77       44        31        17       13
```

```
## 2       3    SF      82    10    1     52       33        28        14       10
## 3       3    SF      66     2    2     47       33        26         9        8
## 4       3    QF     106    12    2     68       43        34        15       11
## 5       3    QF     129    12    3    105       68        48        25       16
## 6       3    QF     127    10    8     94       58        44        18       16
##   w_bp_saved w_bp_faced l_ace l_df l_svpt l_1st_in l_1st_won l_2nd_won l_sv_gms
## 1          3          6     8    6    100       54        34        20       14
## 2          0          1    17    2     77       52        36         7       10
## 3          2          2    10    3     46       27        15         6        8
## 4          4          5     1    2     81       60        38         9       11
## 5          8          8    29    5     94       56        46        19       15
## 6          4          8    12    6     90       54        40        18       15
##   l_bp_saved l_bp_faced winner_rank loser_rank winner_rank_points
## 1         10         15           9         16               3590
## 2         10         13          16        239               1977
## 3          1          5           9         40               3590
## 4          4          6         239         31                200
## 5          2          4          16         18               1977
## 6          6          9          40        185               1050
##   loser_rank_points winner_seed loser_seed total_games w_bp_att l_bp_att
## 1              1977           2          4          27       15        6
## 2               200           4                     20       13        1
## 3              1050           2                     16        5        2
## 4              1298                      7          22        6        5
## 5              1855           4          5          31        4        8
## 6               275                                 31        9        8
##   w_total_pts l_total_pts
## 1         177         177
## 2         129         129
## 3          93          93
## 4         149         149
## 5         199         199
## 6         184         184
```

There were numerous unnecessary variables (i.e. tourney_ID) that we decided to remove. Moreover, the data was not distributed by player match stats, instead there were variables regarding the winner and the loser. Therefore, in preparation for separating the data by player we had to rename and add column types like w_bp_att (winner break points attempted) to l_bp_faced (loser break points faced).

Creating New Data frames

```
win_atp_data = atp_data_rough[, c(1:7, 12:24, 34, 36, 38, 40, 41, 43)]

#head(win_atp_data)

loser_atp_data = atp_data_rough[, c(1:3, 8:15, 25:33, 35, 37, 39, 40, 42, 44)]

#head(loser_atp_data)
```

```
win_atp_data = win_atp_data %>% rename(c("winner_name" = "name",
                                         "winner_hand" = "hand",
```

```
                                      "winner_ht" = "height",
                                      "winner_age" = "age",
                                      "w_ace" = "ace", "w_df" = "df",
                                      "w_svpt" = "svpt",
                                      "w_1st_in" = "first_in",
                                      "w_1st_won" = "first_won",
                                      "w_2nd_won" = "second_won",
                                      "w_sv_gms" = "sv_gms",
                                      "w_bp_saved" = "bp_saved",
                                      "winner_rank" = "rank",
                                      "w_bp_faced" = "bp_faced",
                                      "winner_rank_points" = "rank_pts",
                                      "winner_seed" = "tourney_seed",
                                      "w_bp_att" = "bp_att",
                                      "w_total_pts" = "total_pts")) %>%
  mutate(winner = 1)

loser_atp_data = loser_atp_data %>% rename(c("loser_name" = "name",
                                          "loser_hand" = "hand",
                                      "loser_ht" = "height",
                                      "loser_age" = "age",
                                      "l_ace" = "ace", "l_df" = "df",
                                      "l_svpt" = "svpt",
                                      "l_1st_in" = "first_in",
                                      "l_1st_won" = "first_won",
                                      "l_2nd_won" = "second_won",
                                      "l_sv_gms" = "sv_gms",
                                      "l_bp_saved" = "bp_saved",
                                      "loser_rank" = "rank",
                                      "l_bp_faced" = "bp_faced",
                                      "loser_rank_points" = "rank_pts",
                                      "loser_seed" = "tourney_seed",
                                      "l_bp_att" = "bp_att",
                                      "l_total_pts" = "total_pts")) %>%
  mutate(winner = 0)

head(win_atp_data)
```

```
##   tourney_name surface tourney_level              name hand height      age
## 1     Brisbane    Hard             A      Kei Nishikori    R    178 29.00479
## 2     Brisbane    Hard             A    Daniil Medvedev    R     NA 22.88569
## 3     Brisbane    Hard             A      Kei Nishikori    R    178 29.00479
## 4     Brisbane    Hard             A Jo-Wilfried Tsonga    R    188 33.70568
## 5     Brisbane    Hard             A    Daniil Medvedev    R     NA 22.88569
## 6     Brisbane    Hard             A      Jeremy Chardy    R    188 31.88227
##            score best_of round minutes ace df svpt first_in first_won
## 1    6-4 3-6 6-2       3     F     124   3  3   77       44        31
## 2     7-6(6) 6-2       3    SF      82  10  1   52       33        28
## 3        6-2 6-2       3    SF      66   2  2   47       33        26
## 4     6-4 7-6(2)       3    QF     106  12  2   68       43        34
## 5 6-7(2) 6-3 6-4       3    QF     129  12  3  105       68        48
## 6 6-4 3-6 7-6(4)       3    QF     127  10  8   94       58        44
##   second_won sv_gms bp_saved bp_faced rank rank_pts tourney_seed total_games
```

```
## 1         17     13        3        6    9     3590            2          27
## 2         14     10        0        1   16     1977            4          20
## 3          9      8        2        2    9     3590            2          16
## 4         15     11        4        5  239      200                       22
## 5         25     16        8        8   16     1977            4          31
## 6         18     16        4        8   40     1050                       31
##   bp_att total_pts winner
## 1     15       177      1
## 2     13       129      1
## 3      5        93      1
## 4      6       149      1
## 5      4       199      1
## 6      9       184      1
```

```
head(loser_atp_data)
```

```
##   tourney_name surface tourney_level               name hand height      age
## 1     Brisbane    Hard             A    Daniil Medvedev    R     NA 22.88569
## 2     Brisbane    Hard             A Jo-Wilfried Tsonga    R    188 33.70568
## 3     Brisbane    Hard             A      Jeremy Chardy    R    188 31.88227
## 4     Brisbane    Hard             A     Alex De Minaur    R     NA 19.86858
## 5     Brisbane    Hard             A       Milos Raonic    R    196 28.01095
## 6     Brisbane    Hard             A  Yasutaka Uchiyama    R     NA 26.40383
##            score best_of round minutes ace df svpt first_in first_won
## 1    6-4 3-6 6-2       3     F     124   8  6  100       54        34
## 2     7-6(6) 6-2       3    SF      82  17  2   77       52        36
## 3        6-2 6-2       3    SF      66  10  3   46       27        15
## 4     6-4 7-6(2)       3    QF     106   1  2   81       60        38
## 5 6-7(2) 6-3 6-4       3    QF     129  29  5   94       56        46
## 6 6-4 3-6 7-6(4)       3    QF     127  12  6   90       54        40
##   second_won sv_gms bp_saved bp_faced rank rank_pts tourney_seed total_games
## 1         20     14       10       15   16     1977            4          27
## 2          7     10       10       13  239      200                       20
## 3          6      8        1        5   40     1050                       16
## 4          9     11        4        6   31     1298            7          22
## 5         19     15        2        4   18     1855            5          31
## 6         18     15        6        9  185      275                       31
##   bp_att total_pts winner
## 1      6       177      0
## 2      1       129      0
## 3      2        93      0
## 4      5       149      0
## 5      8       199      0
## 6      8       184      0
```

Here we had to rename all of the columns to the same name in order to bind the two data frames together.

```
atp_data = rbind.data.frame(win_atp_data, loser_atp_data)
```

Here we bound the two data frames together.

Binarization and other cleaning

```
atp_data = atp_data %>%
  mutate(hand = if_else(hand == "R", 1, 0)) %>%
  arrange(name) %>%
  filter(rank <= 100)
atp_data_no_ret = atp_data[!grepl('RET|W/O', atp_data$score),]
```

We filter the data to only include players that were at one time in the top 100. Moreover, we took out matches that ended in retirement because they were not representative a typical ATP match. This was because we wanted a sample of individuals who were similar, but relatively diverse at the same time. We binarized the data (i.e. hand dominance and win) just in case we wanted to perform tests or models that needed data in that format.

Grouping Data by indivudal

```
atp_data_player = atp_data_no_ret %>%
  group_by(name) %>%
  dplyr::summarise(mean_win_rate = mean(winner, na.rm = TRUE),
                   mean_fs_win_rate =
                     sum(first_won, na.rm = TRUE)/sum(first_in, na.rm = TRUE),
                   mean_sec_srv_rate =
                     mean((svpt - first_in)/svpt, na.rm = TRUE),
                   mean_sec_win_rate =
                     sum(second_won, na.rm = TRUE)/sum(svpt - first_in,
                                                       na.rm = TRUE),
                   mean_ace_rate = sum(ace, na.rm = TRUE)/sum(svpt,
                                                              na.rm = TRUE),
                   mean_pts_min = sum(total_pts, na.rm = TRUE)/sum(minutes,
                                                                   na.rm = TRUE),
                   mean_bp_faced_rate = sum(bp_faced, na.rm = TRUE)/sum(svpt,
                                                                        na.rm = TRUE),
                   mean_bp_att_rate =
                     sum(bp_att, na.rm = TRUE)/sum(total_pts - svpt,
                                                   na.rm = TRUE),
                   mean_bp_save_rate = sum(bp_saved, na.rm = TRUE)/sum(bp_faced,
                                                                       na.rm
                                                                       = TRUE),
                   mean_age = mean(age, na.rm = TRUE),
                   mean_df_rate = sum(df, na.rm = TRUE)/sum(svpt, na.rm = TRUE),
                   hand = mean(hand),
                   mean_rank = mean(rank, na.rm = T),
                   mean_svpt = mean(svpt, na.rm = T)) %>%
  ungroup()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
#write.csv(atp_data_player, "./Data/atp_data_player.csv")

atp_data_hard = atp_data_no_ret %>% filter(surface == "Hard") %>%
```

```
  group_by(name) %>%
  dplyr::summarise(mean_win_rate = mean(winner, na.rm = TRUE),
                   mean_fs_win_rate =
                     sum(first_won, na.rm = TRUE)/sum(first_in, na.rm = TRUE),
                   mean_sec_srv_rate =
                     mean((svpt - first_in)/svpt, na.rm = TRUE),
                   mean_sec_win_rate =
                     sum(second_won, na.rm = TRUE)/sum(svpt - first_in-df,
                                                       na.rm = TRUE),
                   mean_sec_in_rate =
                     sum(svpt - first_in-df, na.rm = TRUE)/sum(svpt,na.rm =
                                                               TRUE),
                   mean_ace_rate =
                     sum(ace, na.rm = TRUE)/sum(svpt, na.rm = TRUE),
                   mean_pts_min =
                     sum(total_pts, na.rm = TRUE)/sum(minutes, na.rm = TRUE),
                   mean_bp_faced_rate =
                     sum(bp_faced, na.rm = TRUE)/sum(svpt, na.rm = TRUE),
                   mean_bp_att_rate =
                     sum(bp_att, na.rm = TRUE)/sum(total_pts - svpt, na.rm
                                                   = TRUE),
                   mean_bp_save_rate =
                     sum(bp_saved, na.rm = TRUE)/sum(bp_faced, na.rm = TRUE),
                   mean_age = mean(age, na.rm = TRUE),
                   mean_df_rate = sum(df, na.rm = TRUE)/sum(svpt, na.rm = TRUE),
                   hand = mean(hand),
                   mean_rank = mean(rank, na.rm = T)) %>%
  ungroup()%>%
    mutate(surface = "Hard")
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
#write.csv(atp_data_hard, "./Data/atp_data_hard.csv")


  atp_data_clay = atp_data_no_ret %>% filter(surface == "Clay") %>%
    group_by(name) %>%
    dplyr::summarise(mean_win_rate = mean(winner, na.rm = TRUE),
                   mean_fs_win_rate = sum(first_won, na.rm = TRUE)/sum(first_in,
                                                                       na.rm
                                                                       = TRUE),
                   mean_sec_srv_rate = mean((svpt - first_in)/svpt,
                                            na.rm = TRUE),
                   mean_sec_win_rate =
                     sum(second_won, na.rm = TRUE)/sum(svpt - first_in-df,
                                                       na.rm = TRUE),
                   mean_sec_in_rate =
                     sum(svpt - first_in-df, na.rm = TRUE)/sum(svpt,
                                                               na.rm = TRUE),
                   mean_ace_rate = sum(ace, na.rm = TRUE)/sum(svpt,
                                                              na.rm = TRUE),
                   mean_pts_min = sum(total_pts, na.rm = TRUE)/sum(minutes,
                                                                   na.rm =
```

```
                                                                TRUE),
                  mean_bp_faced_rate = sum(bp_faced, na.rm = TRUE)/sum(svpt,
                                                                       na.rm
                                                                       = TRUE),
                  mean_bp_att_rate =
                    sum(bp_att, na.rm = TRUE)/sum(total_pts - svpt,
                                                  na.rm = TRUE),
                  mean_bp_save_rate =
                    sum(bp_saved, na.rm = TRUE)/sum(bp_faced,
                                                    na.rm = TRUE),
                  mean_age = mean(age, na.rm = TRUE),
                  mean_df_rate = sum(df, na.rm = TRUE)/sum(svpt,
                                                           na.rm = TRUE),
                  hand = mean(hand),
                  mean_rank = mean(rank, na.rm = T)) %>%
  ungroup()%>%
    mutate(surface = "Clay")
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
#write.csv(atp_data_clay, "./Data/atp_data_clay.csv")

  atp_data_grass = atp_data_no_ret %>% filter(surface == "Grass")%>%
    group_by(name) %>%
    dplyr::summarise(mean_win_rate = mean(winner, na.rm = TRUE),
                  mean_fs_win_rate =
                    sum(first_won, na.rm = TRUE)/sum(first_in, na.rm = TRUE),
                  mean_sec_srv_rate = mean((svpt - first_in)/svpt, na.rm =
                                             TRUE),
                  mean_sec_win_rate =
                    sum(second_won, na.rm = TRUE)/sum(svpt - first_in-df,
                                                      na.rm = TRUE),
                  mean_sec_in_rate =
                    sum(svpt - first_in-df, na.rm = TRUE)/sum(svpt, na.rm =
                                                                TRUE),
                  mean_ace_rate = sum(ace, na.rm = TRUE)/sum(svpt, na.rm =
                                                               TRUE),
                  mean_pts_min = sum(total_pts, na.rm = TRUE)/sum(minutes,
                                                                  na.rm =
                                                                    TRUE),
                  mean_bp_faced_rate = sum(bp_faced, na.rm = TRUE)/sum(svpt,
                                                                       na.rm =
                                                                         TRUE),
                  mean_bp_att_rate =
                    sum(bp_att, na.rm = TRUE)/sum(total_pts - svpt,
                                                  na.rm = TRUE),
                  mean_bp_save_rate = sum(bp_saved, na.rm = TRUE)/sum(bp_faced,
                                                                      na.rm =
                                                                        TRUE),
                  mean_age = mean(age, na.rm = TRUE),
                  mean_df_rate = sum(df, na.rm = TRUE)/sum(svpt, na.rm = TRUE),
                  hand = mean(hand),
                  mean_rank = mean(rank, na.rm = T)) %>%
```

```
  ungroup() %>%
    mutate(surface = "Grass")
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
#write.csv(atp_data_grass, "./Data/atp_data_grass.csv")
```

We decided to group the data by players and create rate and mean stats. This was because if we did not use means, better players, who generally go farther in tournaments and play more matches would be weighted higher in our analysis than lower ranked players. We also created rate stats to control for the fact that some players due to their play style would play longer matches than others.

Performing the Analysis

In this section we will perform both EDA analysis and create models.

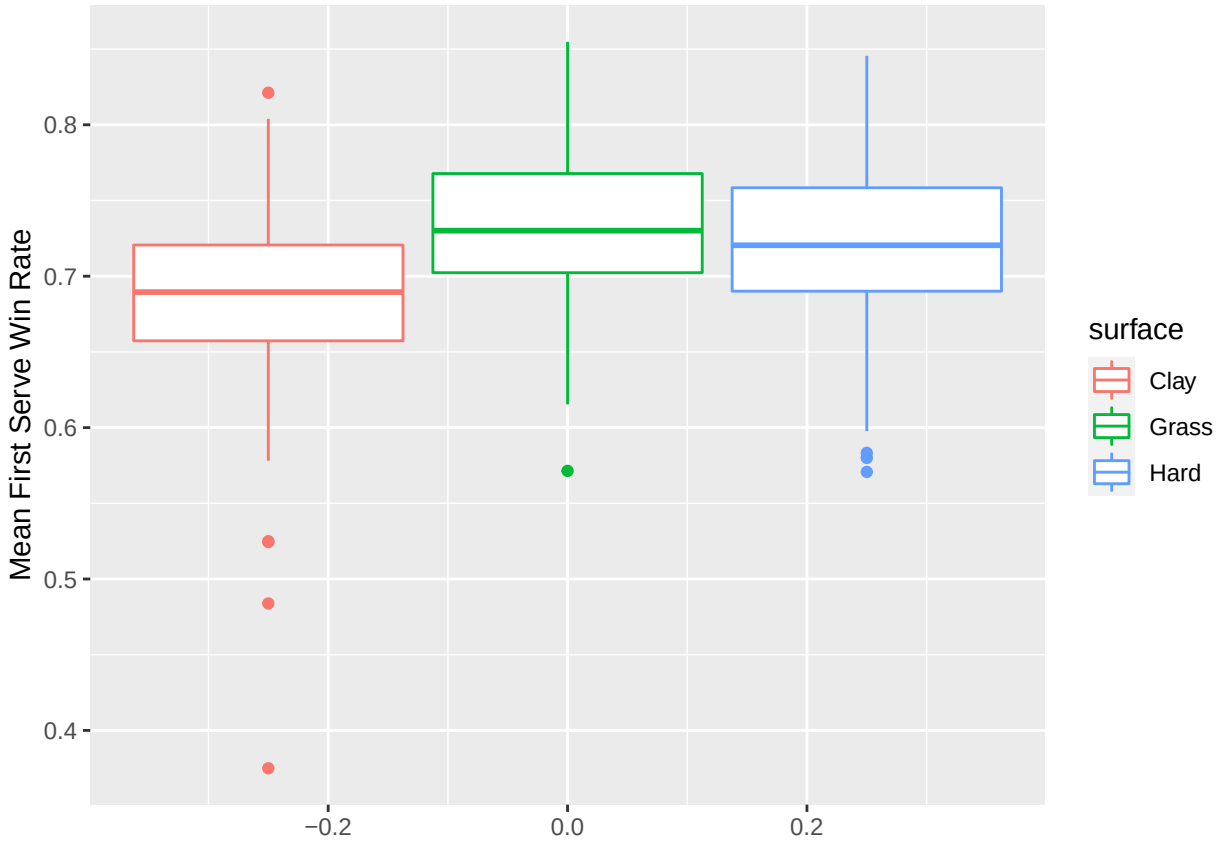
Re-loading Data

```
atp_data_hard = read.csv("~/Desktop/Moneyball/Data/atp_data_hard.csv") %>%
  mutate(surface == hard) %>%
  select(!X)
atp_data_grass = read.csv("~/Desktop/Moneyball/Data/atp_data_grass.csv") %>%
  mutate(surface == grass) %>%
  select(!X)

atp_data_clay = read.csv("~/Desktop/Moneyball/Data/atp_data_clay.csv")

atp_data_clay = atp_data_clay %>%
  mutate(surface == clay) %>%
  select(!X)

atp_data_og = read.csv("~/Desktop/Moneyball/Data/atp_data_2019.csv")

atp_data_player = read.csv("~/Desktop/Moneyball/Data/atp_data_player.csv")

atp_data_player = atp_data_player %>%
  select(-X)


atp_data_mlm = rbind.data.frame(atp_data_hard, atp_data_grass, atp_data_clay)
```

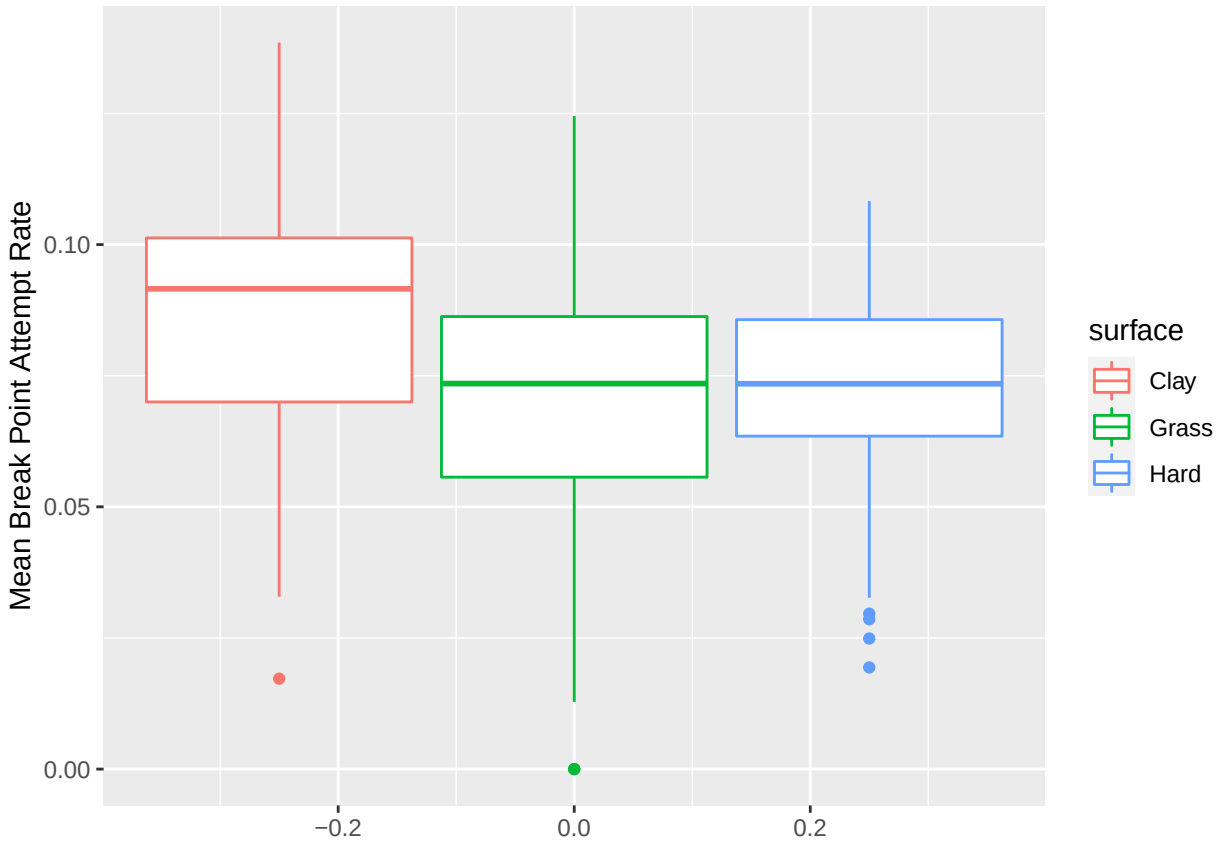
We created the atp_data_mlm data frame in order to distinguish the differences between each surface type by creating an aggregate model.

Box Plots

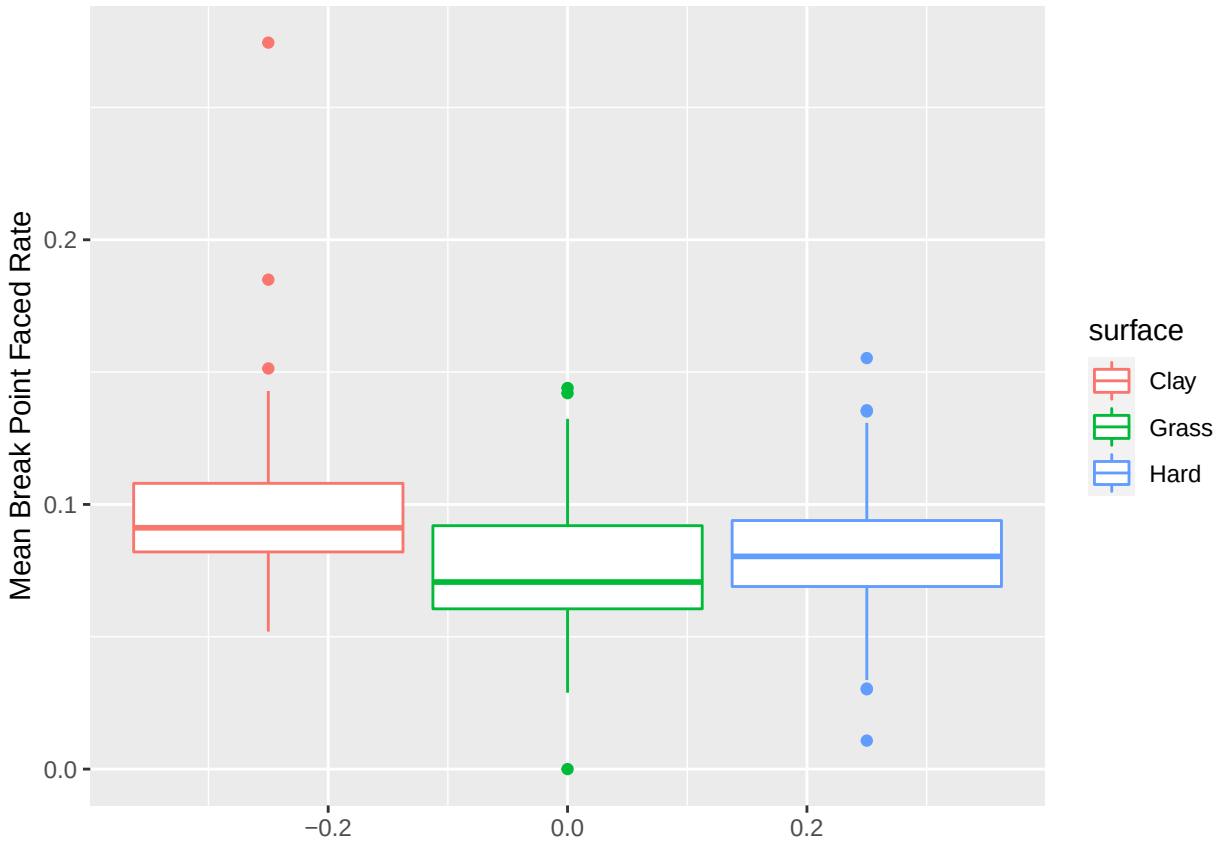
```
ggplot(atp_data_mlm) +
  geom_boxplot(aes(mean_fs_win_rate, color = surface)) +
  labs(x = "Mean First Serve Win Rate") +
  coord_flip()
```



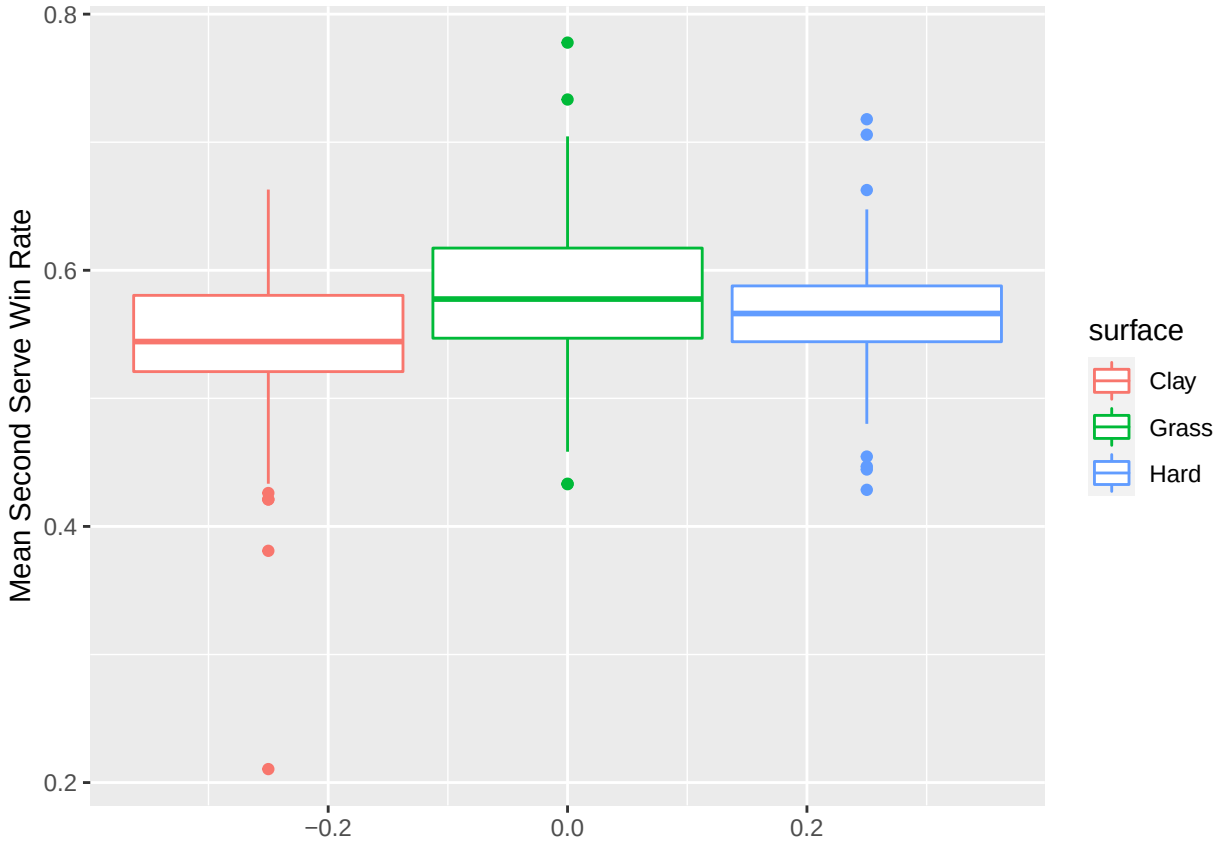

```
ggplot(atp_data_mlm) +
  geom_boxplot(aes(mean_bp_att_rate, color = surface)) +
  labs(x = "Mean Break Point Attempt Rate") +
  coord_flip()
```



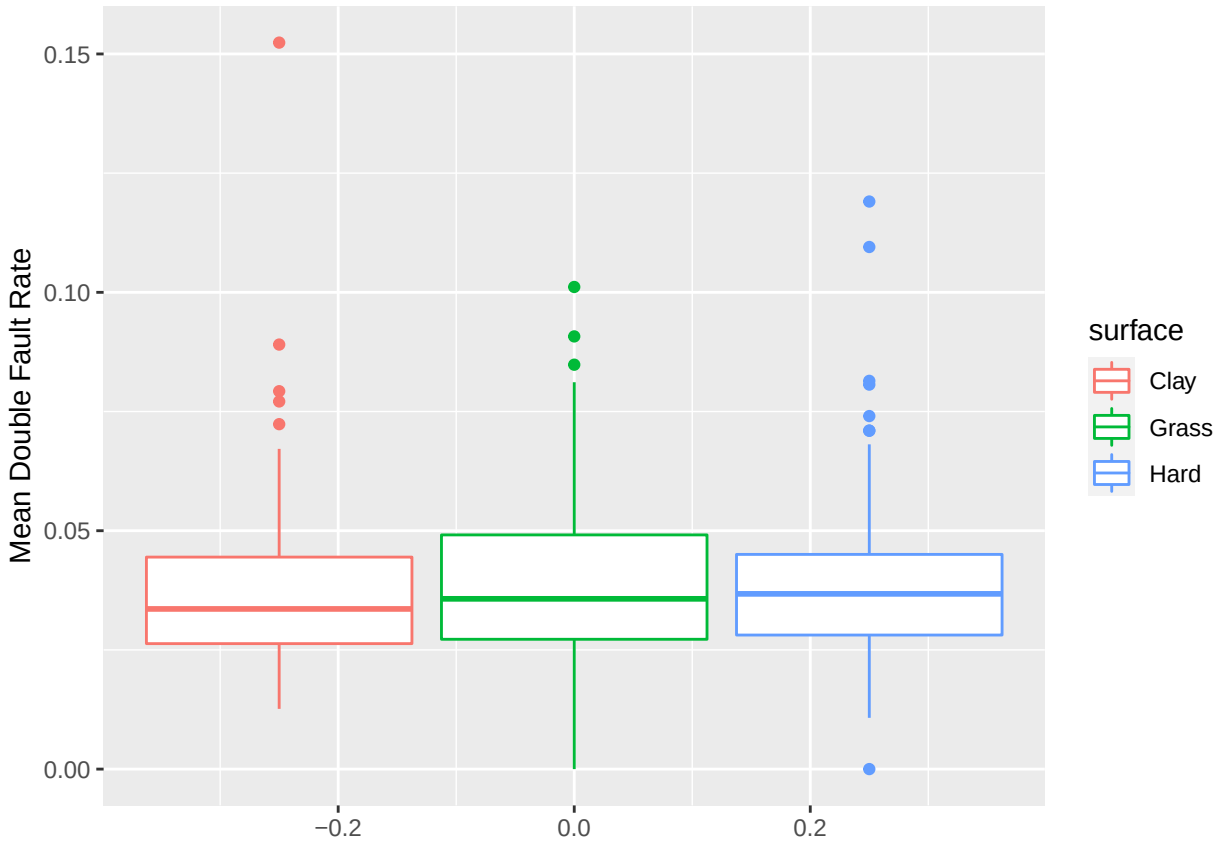

```
ggplot(atp_data_mlm) +
  geom_boxplot(aes(mean_bp_faced_rate, color = surface)) +
  labs(x = "Mean Break Point Faced Rate") +
  coord_flip()
```



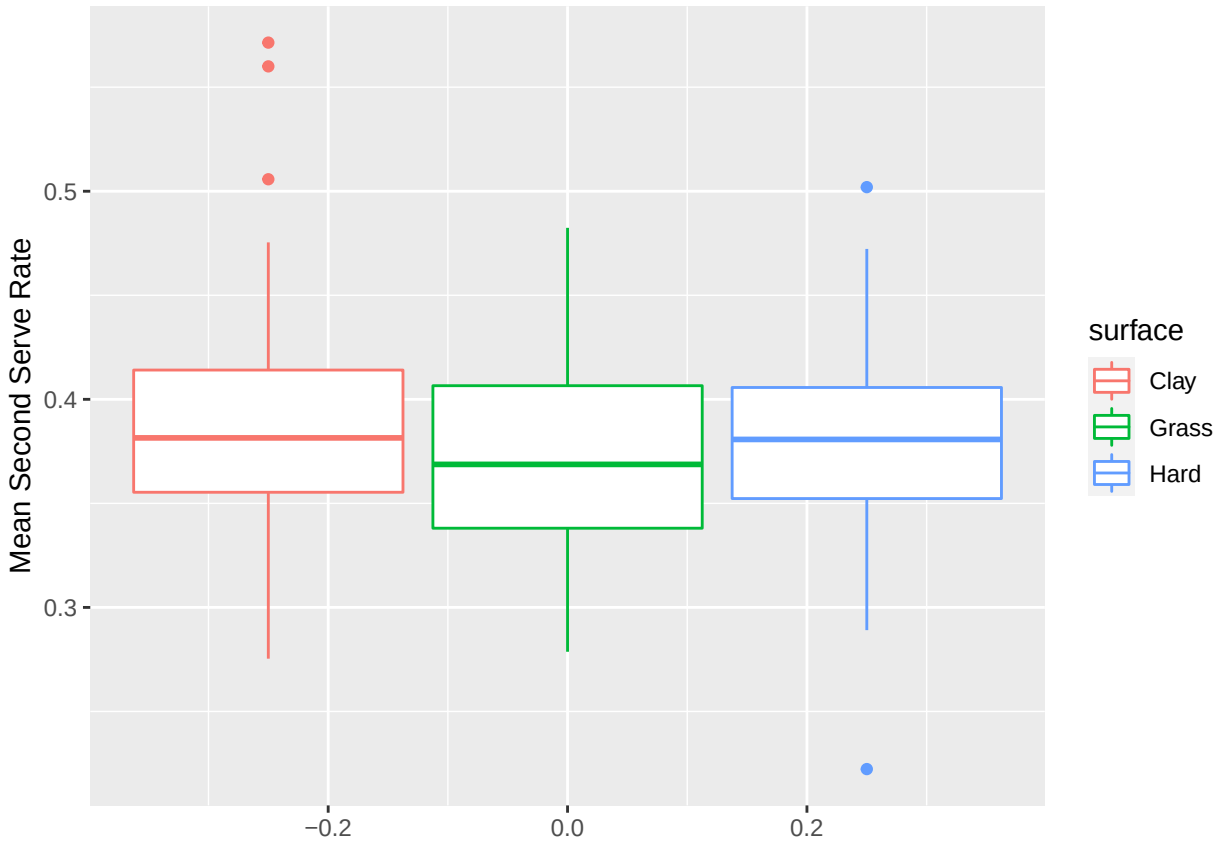

```
ggplot(atp_data_mlm) +
  geom_boxplot(aes(mean_sec_win_rate, color = surface)) +
  labs(x = "Mean Second Serve Win Rate") +
  coord_flip()
```



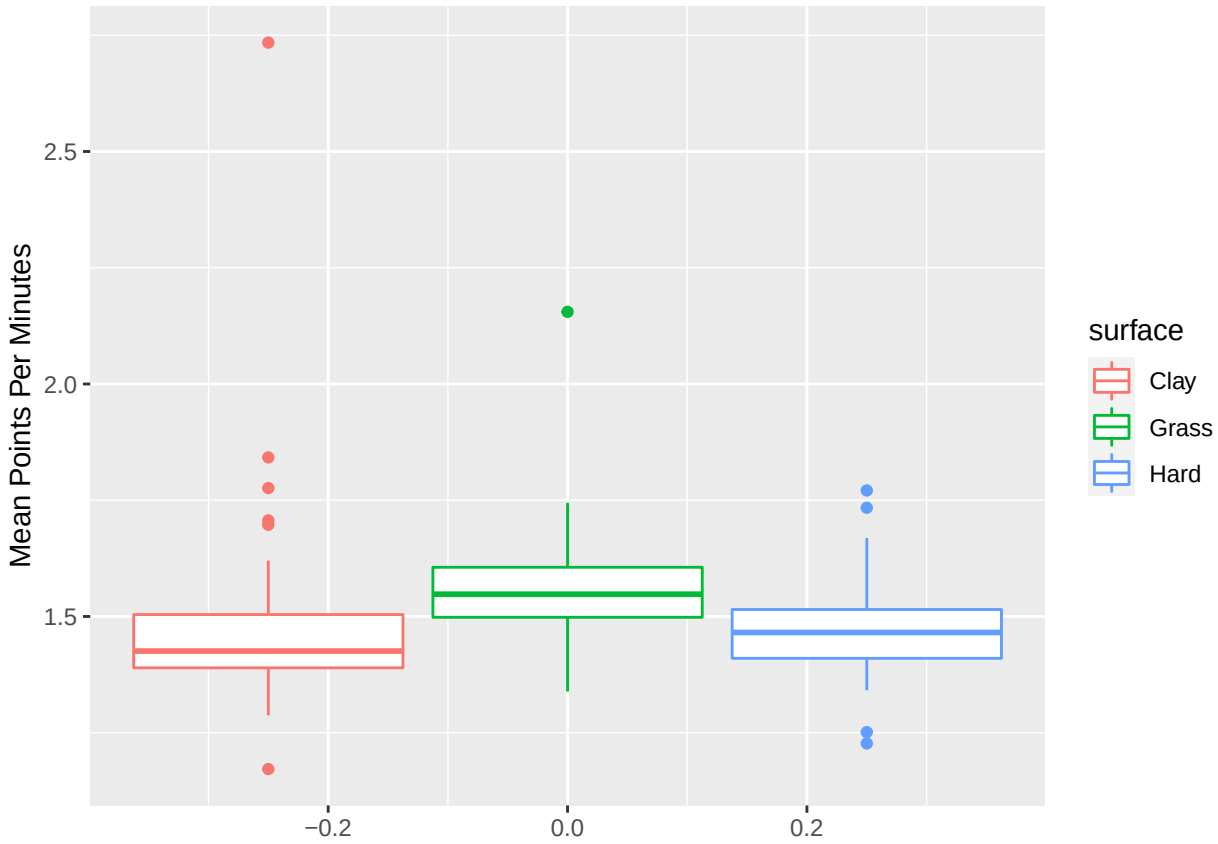

```
ggplot(atp_data_mlm) +
  geom_boxplot(aes(mean_df_rate, color = surface)) +
  labs(x = "Mean Double Fault Rate") +
  coord_flip()
```




```
ggplot(atp_data_mlm) +
  geom_boxplot(aes(mean_sec_srv_rate, color = surface)) +
  labs(x = "Mean Second Serve Rate") +
  coord_flip()
```




```
ggplot(atp_data_mlm) +
  geom_boxplot(aes(mean_pts_min, color = surface)) +
  labs(x = "Mean Points Per Minutes") +
  coord_flip()
```



Looking at these scatter plots there are defiantly some variables that differ on each surface type and others that stay the safe.

Some of our prior theories in the background section are supported by these plots. Notably, looking at points per minute statistic (ppm), generally clay has the fewest ppm followed by hard and grass, which aligns with our notion that clay is the slowest surface, then hard, and then grass.

Creating a Model & Our Ideology

We decided to create a multivariable regression because there are numerous variables that impact a person's performance on the tennis court. Then we would look at the summary statistics and other graphs in order to describe the difference in each surface type.

Variables Chosen

1. First Serve Win Rate: The proportion of points won when the first serve was in.
2. Break Point Attempt Rate: The amount of break points produced divided by the total amount of points played as the returner.
3. Break Point Faced Rate: The amount of break points faced divided by the total amount of points played as the server.
4. Second Serve Rate: The proportion of second serves attempted divided by the number of serves attempted.

What is a break point?

A break point occurs when the returner has to win one more point to win a game on the opponent's serve (also known as a service game).

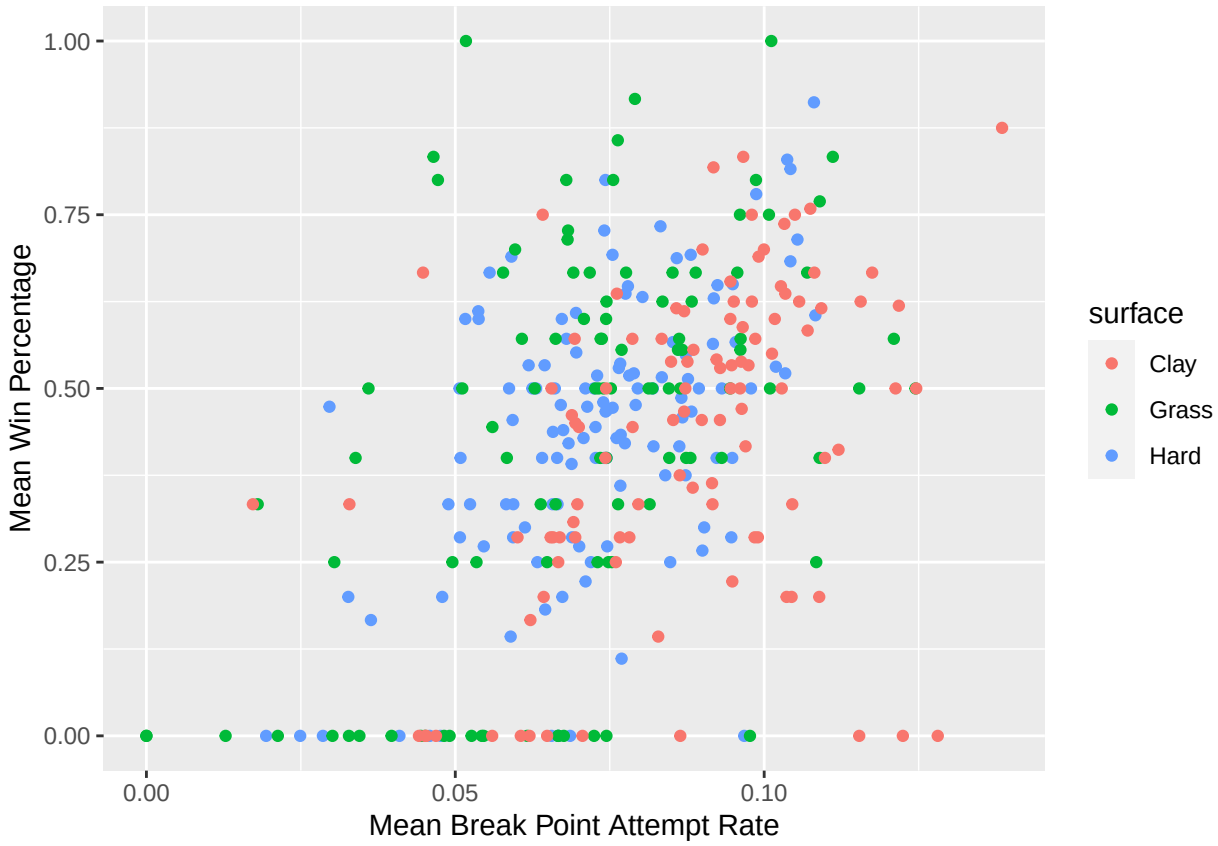
Examples: 0-40, 30-40, or ad-out

We chose these variables because it is known that the serve is the most important shot in tennis (since it is the only one that is fully controllable), and because in the ATP specifically players are expected to maintain service games. Thus, break points are an important factor in getting a leg-up on the opponent, which typically leads to more winning.

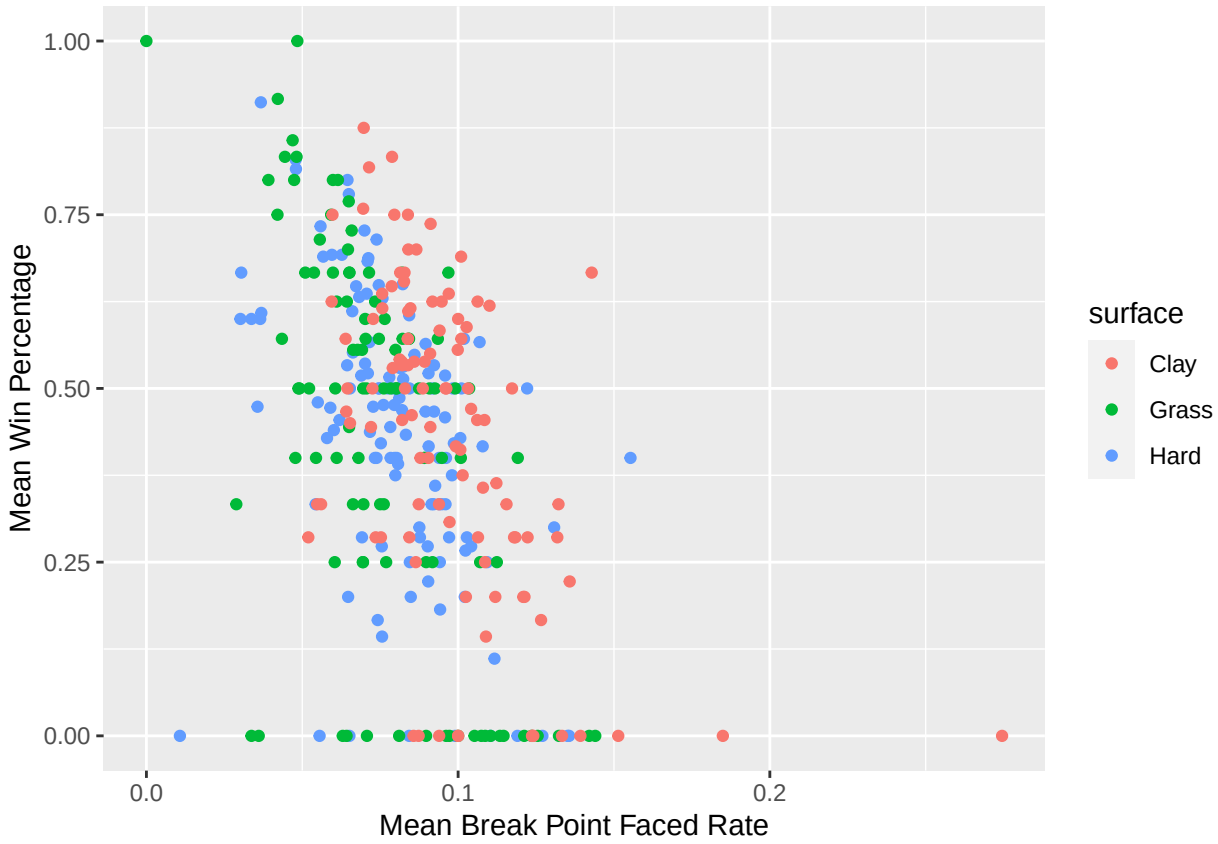
Scatter Plots of Variables Chosen

```
ggplot(atp_data_mlm) +
  geom_point(aes(mean_fs_win_rate, mean_win_rate, color = surface)) +
  labs(x = "Mean First Serve Win Rate", y = "Mean Win Percentage")
```


```
ggplot(atp_data_mlm) +
  geom_point(aes(mean_bp_att_rate, mean_win_rate, color = surface)) +
  labs(x = "Mean Break Point Attempt Rate", y = "Mean Win Percentage")
```




```
ggplot(atp_data_mlm) +
  geom_point(aes(mean_bp_faced_rate, mean_win_rate, color = surface)) +
  labs(x = "Mean Break Point Faced Rate", y = "Mean Win Percentage")
```



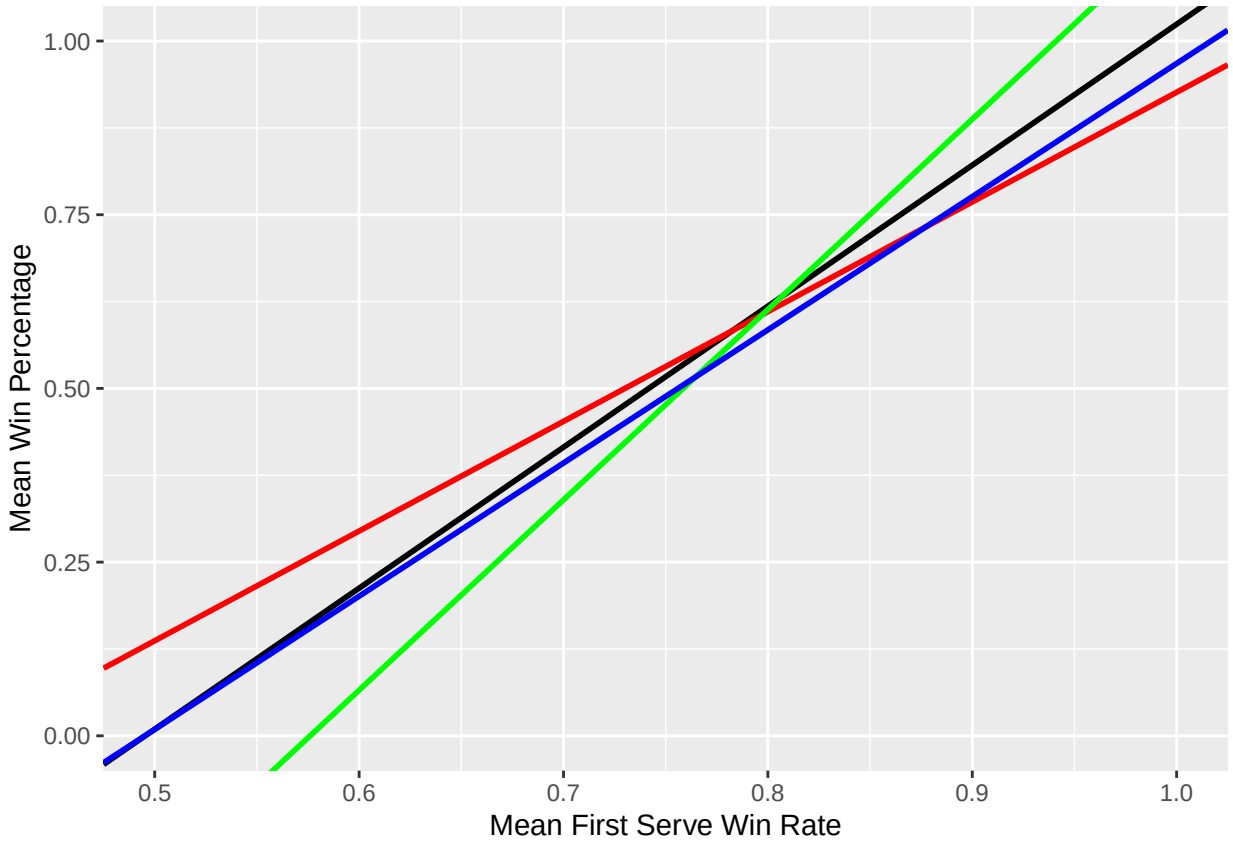

```
ggplot(atp_data_mlm) +
  geom_point(aes(mean_sec_srv_rate, mean_win_rate, color = surface)) +
  labs(x = "Mean Second Serve Rate", y = "Mean Win Percentage")
```

Above are the scatter plots of all of variable in our multivariable regression. There are certainly some differences regarding each surface type, but the scatter plot is pretty cultured so we decided to create linear models for each variable based on surface type.

Linear Models

Blacks mean not grouped by surface type (aggregate) and is meant to show the problems in a one-size fits all model.

```
#First Serve Win Rate with Winning Rate

ggplot(atp_data_hard) +
  geom_abline(intercept = -1.0051, slope = 2.0293, col = "Black", size = 1) +
  geom_abline(intercept = -0.6526, slope = 1.5788, col = "red", size = 1) +
  geom_abline(intercept = -1.5780, slope = 2.7394, col = "Green", size = 1) +
  geom_abline(intercept = -0.9492, slope = 1.9169, col = "blue", size = 1) +
  xlim(0.5, 1) +
  ylim(0,1) +
  labs(x = "Mean First Serve Win Rate", y = "Mean Win Percentage")
```

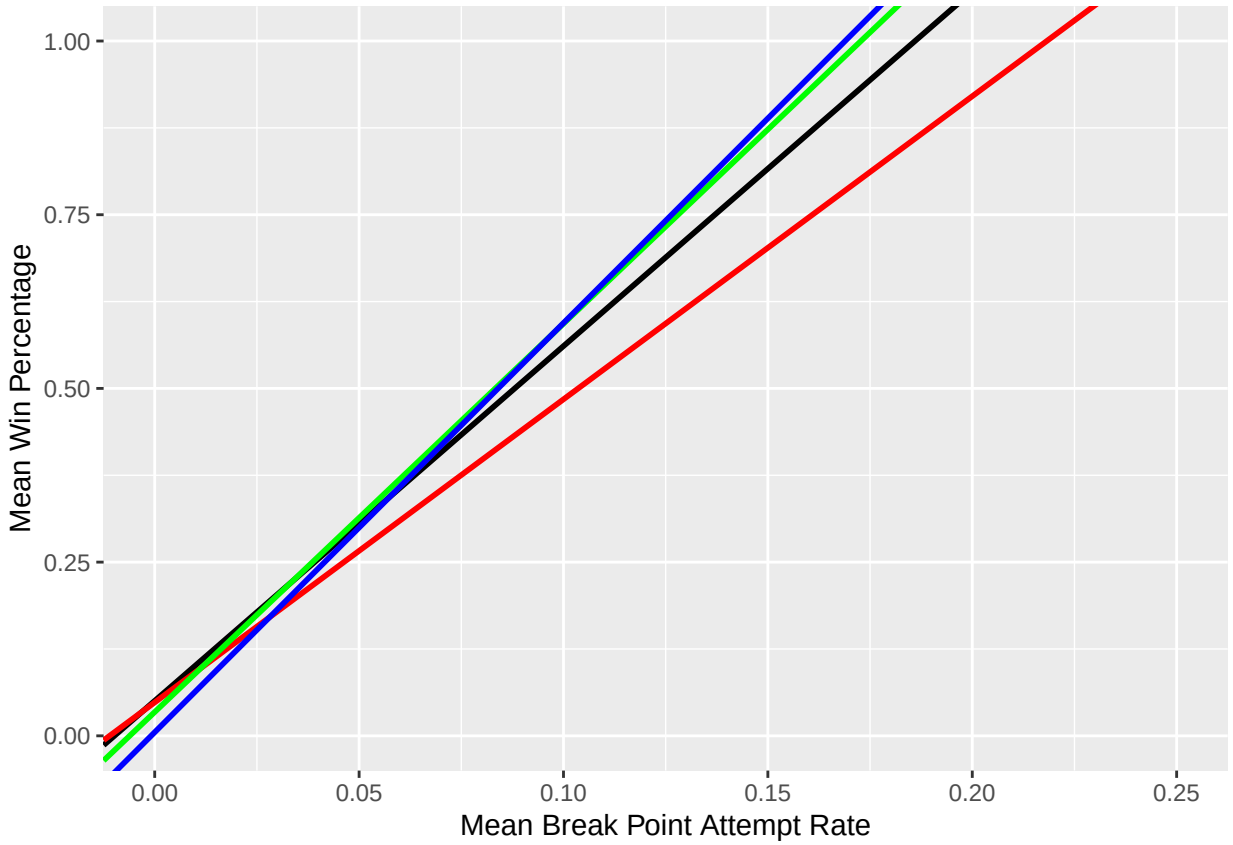



```
#summary(lm(mean_win_rate~mean_fs_win_rate, data = atp_data_player))
#summary(lm(mean_win_rate~mean_fs_win_rate, data = atp_data_clay))
#summary(lm(mean_win_rate~mean_fs_win_rate, data = atp_data_grass))
#summary(lm(mean_win_rate~mean_fs_win_rate, data = atp_data_hard))
```

As seen above, there is generally a positive association between first serve win rate and winning percentage. This makes sense because someone would expect that if a player wins more points on their first serve they will improve their chances at winning the match.

The slope for grass courts is the steepest slope (2.74) and this makes intuitive sense, because on grass courts the game is faster and players take more risk. Thus, having an effective first serve that puts the server in an advantageous position to win the next point is crucial. Clay and hard courts are a bit closer together, with slopes of 1.58 and 1.92 respectively, because the style of play is slower than that of grass. Also, the fact that clay has the shallowest slope should be expected since due to the nature of the surface, points are longer and therefore, the advantages of the serve are lessened.

```
#Break Point Attempt Rate with Winning Rate

ggplot(atp_data_mlm) +
  geom_abline(intercept = 0.05034, slope = 5.10571, col = "Black", size = 1) +
  geom_abline(intercept = 0.04816, slope = 4.36168, col = "red", size = 1) +
  geom_abline(intercept = 0.03427, slope = 5.58964, col = "Green", size = 1) +
  geom_abline(intercept = 0.00519, slope = 5.88986, col = "blue", size = 1) +
  xlim(0,0.25) +
  ylim(0,1) +
  labs(x = "Mean Break Point Attempt Rate", y = "Mean Win Percentage")
```



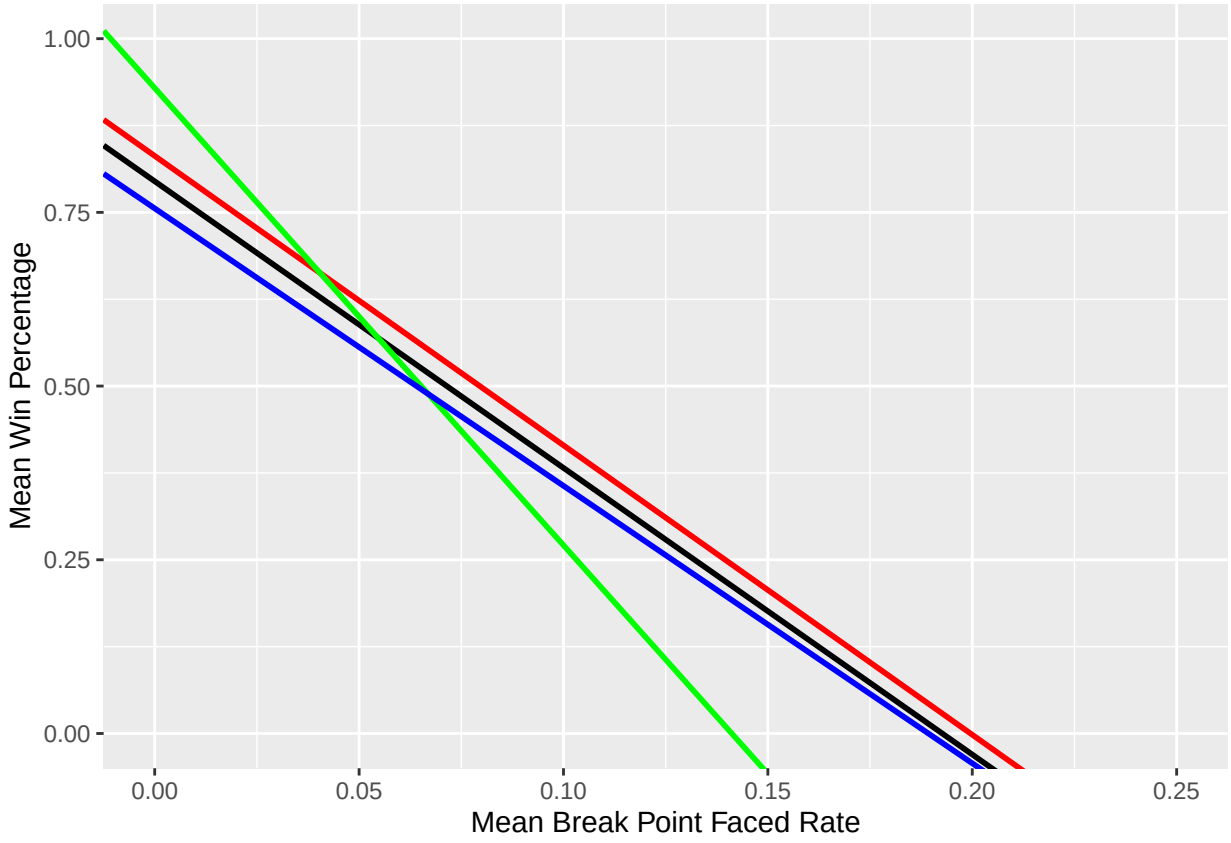

```
#summary(lm(mean_win_rate~mean_bp_att_rate, data = atp_data_player))
#summary(lm(mean_win_rate~mean_bp_att_rate, data = atp_data_clay))
#summary(lm(mean_win_rate~mean_bp_att_rate, data = atp_data_grass))
#summary(lm(mean_win_rate~mean_bp_att_rate, data = atp_data_hard))
```

The linear models show a positive correlation between break points attempted and winning percentage. This makes sense because it is difficult to break serve on the ATP tour. Thus, the more chances to break an opponents the serve the more likely it is for someone to win a match.

Along with that, the fact that clay has the shallowest slope (4.36) makes sense because due to the nature of the surface, points are longer on clay and require more shots. Thus, the ability for a player to gain free points off the serve or set themselves up to win a point in the next two or three shots is relatively weakened on the clay court. Therefore, it will be easier to generate break points, but that also works against the player when they are serving. Similarly, it makes sense that the hard court and grass courts have a steeper slope because it is generally easier to generate aces, missed returners, or win a point in the next 1 to 3 shoots (i.e. the serve and volley tactic on grass courts).

```
#Break Point Faced Rate with Winning Rate

ggplot(atp_data_mlm) +
  #geom_point(aes(mean_bp_faced_rate, mean_win_rate, color = surface)) +
  geom_abline(intercept = 0.7947, slope = -4.1248, col = "Black", size = 1) +
  geom_abline(intercept = 0.83126, slope = -4.16520 , col = "red", size = 1) +
  geom_abline(intercept = 0.92894, slope = -6.58008, col = "Green", size = 1) +
  geom_abline(intercept = 0.75574, slope = -3.99290 , col = "blue", size = 1) +
  xlim(0,0.25) +
```

```
  ylim(0,1) +
  labs(x = "Mean Break Point Faced Rate", y = "Mean Win Percentage")
```




```
#summary(lm(mean_win_rate~mean_bp_faced_rate, data = atp_data_player))
#summary(lm(mean_win_rate~mean_bp_faced_rate, data = atp_data_clay))
#summary(lm(mean_win_rate~mean_bp_faced_rate, data = atp_data_grass))
#summary(lm(mean_win_rate~mean_bp_faced_rate, data = atp_data_hard))
```

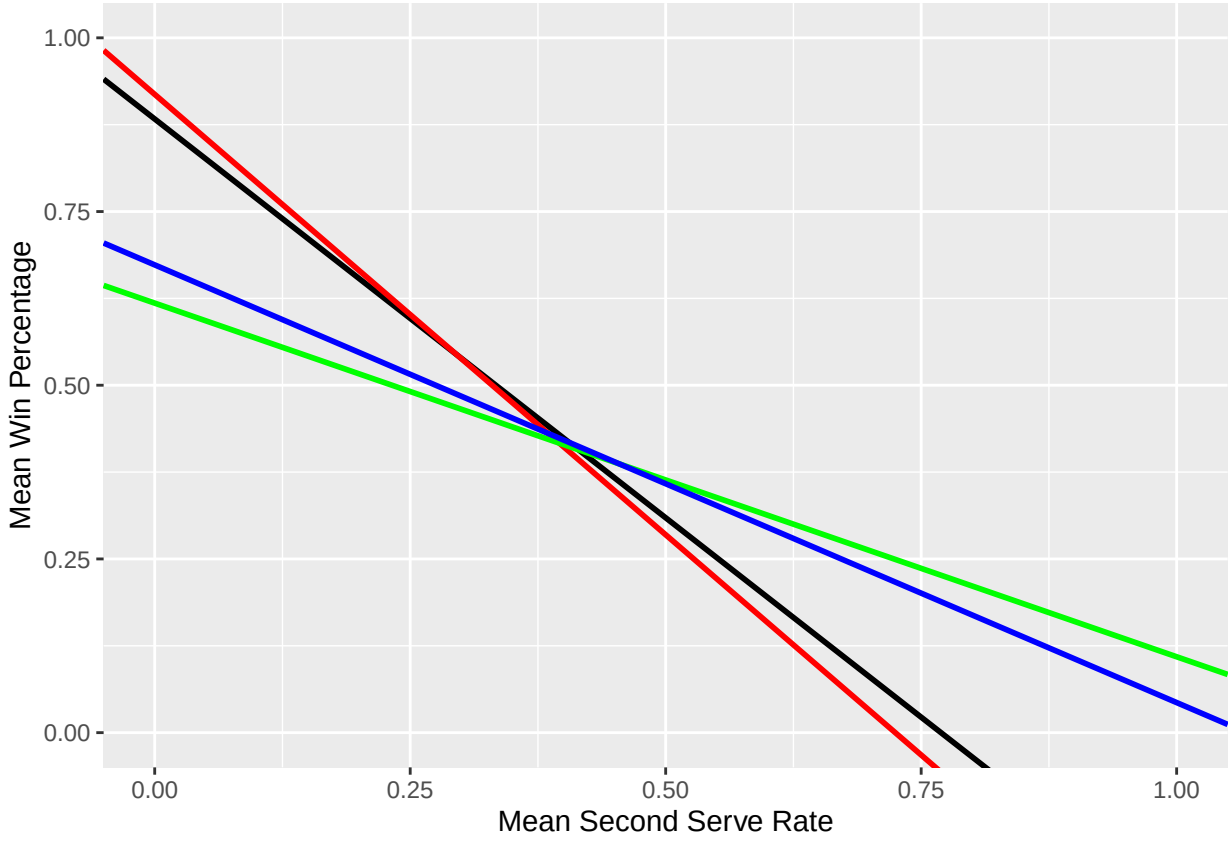
There is negative correlation between break points faced and win percentage. This should be the case because as previously mentioned ATP players are expected to win their service games. Hence, if a players faces many break points they are more likely to get broken. That notion combined with the fact that it is generally difficult to break ATP players' service games means, if a player faces many break points they are more likely to lose the match.

The fact that grass court slope is the steepest (-6.58) makes sense, because generally it is easier to maintain serve on grass, because of the way the ball skids on a grass court. Thus, if someone has a nasty slice serve or a good serve in general, they will generally win the service game. Therefore, if someone gets broken it is generally even more difficult to break back than on other surfaces.

As for clay, it makes sense that its slope is less than grass (-4.17) because as mentioned before, more break points are generated on clay. It is actually interesting that hard court has the shallowest slope (-4), and this is something that we would further study if given more time. Also, perhaps the difference between clay and hard court is not statistically significant.

```
#Second Serve Rate Rate with Winning Rate
```

```
ggplot(atp_data_mlm) +
  #geom_point(aes(mean_sec_srv_rate, mean_win_rate, color = surface)) +
  geom_abline(intercept = 0.8832, slope = -1.1479, col = "Black", size = 1) +
  geom_abline(intercept = 0.9186, slope = -1.2676 , col = "red", size = 1) +
  geom_abline(intercept = 0.6182, slope = -0.5090, col = "Green", size = 1) +
  geom_abline(intercept = 0.6732, slope = -0.6299 , col = "blue", size = 1) +
  xlim(0,1) +
  ylim(0,1) +
  labs(x = "Mean Second Serve Rate", y = "Mean Win Percentage")
```




```
#summary(lm(mean_win_rate~mean_sec_srv_rate, data = atp_data_player))
#summary(lm(mean_win_rate~mean_sec_srv_rate, data = atp_data_clay))
#summary(lm(mean_win_rate~mean_sec_srv_rate, data = atp_data_grass))
#summary(lm(mean_win_rate~mean_sec_srv_rate, data = atp_data_hard))
```

As seen above, there is a negative association between second serve rate and win percentage. This makes sense because on second serves players have to be less aggressive and more risk-adverse, resulting in serves that are easier for the opponents to attack.

This idea is exacerbated on clay because players are more risk-adverse on clay court because they know that generally they won't be winning points on the serve or set themselves up for two or three shot points. As for hard and grass, the slopes are lower because serves are not as easily attacked as they are on clay.

Results

```
multi_lm_atp_hard = lm(mean_win_rate ~ mean_fs_win_rate + mean_bp_faced_rate +
                         mean_bp_att_rate + mean_sec_srv_rate,
                       data = atp_data_hard)

summary(multi_lm_atp_hard)
```

```
##
## Call:
## lm(formula = mean_win_rate ~ mean_fs_win_rate + mean_bp_faced_rate +
##     mean_bp_att_rate + mean_sec_srv_rate, data = atp_data_hard)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39808 -0.06016  0.00321  0.07136  0.24867
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -1.2411     0.3269  -3.796 0.000231 ***
## mean_fs_win_rate     2.2583     0.3878   5.823 4.85e-08 ***
## mean_bp_faced_rate  -0.3649     0.8894  -0.410 0.682360
## mean_bp_att_rate     6.7253     0.5908  11.383  < 2e-16 ***
## mean_sec_srv_rate   -1.0914     0.2814  -3.878 0.000172 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1165 on 121 degrees of freedom
## Multiple R-squared:  0.6677, Adjusted R-squared:  0.6567
## F-statistic: 60.78 on 4 and 121 DF,  p-value: < 2.2e-16
```

```
multi_lm_atp_grass = lm(mean_win_rate ~ mean_fs_win_rate + mean_bp_faced_rate +
                          mean_bp_att_rate + mean_sec_srv_rate,
                        data = atp_data_grass)

summary(multi_lm_atp_grass)
```

```
##
## Call:
## lm(formula = mean_win_rate ~ mean_fs_win_rate + mean_bp_faced_rate +
##     mean_bp_att_rate + mean_sec_srv_rate, data = atp_data_grass)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49395 -0.10638  0.01369  0.13911  0.35986
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -0.3918     0.3941  -0.994 0.322653
## mean_fs_win_rate     1.3140     0.4521   2.907 0.004546 **
## mean_bp_faced_rate  -3.7723     1.0327  -3.653 0.000425 ***
```

```
## mean_bp_att_rate     4.5388     0.7688   5.904 5.48e-08 ***
## mean_sec_srv_rate   -0.4725     0.4361  -1.083 0.281385
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1875 on 95 degrees of freedom
## Multiple R-squared:  0.5707, Adjusted R-squared:  0.5526
## F-statistic: 31.57 on 4 and 95 DF,  p-value: < 2.2e-16
```

```
multi_lm_atp_clay = lm(mean_win_rate ~ mean_fs_win_rate + mean_bp_faced_rate +
                         mean_bp_att_rate + mean_sec_srv_rate,
                       data = atp_data_clay)

summary(multi_lm_atp_clay)
```

```
##
## Call:
## lm(formula = mean_win_rate ~ mean_fs_win_rate + mean_bp_faced_rate +
##     mean_bp_att_rate + mean_sec_srv_rate, data = atp_data_clay)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40773 -0.08472 -0.00218  0.08682  0.69366
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.5097     0.3210   1.588   0.1155
## mean_fs_win_rate     0.3635     0.3963   0.917   0.3612
## mean_bp_faced_rate  -3.8118     0.8635  -4.415 2.56e-05 ***
## mean_bp_att_rate     4.4077     0.7705   5.720 1.11e-07 ***
## mean_sec_srv_rate   -0.8924     0.3394  -2.630   0.0099 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1611 on 100 degrees of freedom
## Multiple R-squared:  0.5163, Adjusted R-squared:  0.4969
## F-statistic: 26.68 on 4 and 100 DF,  p-value: 4.555e-15
```

Evaluating the Results

All p-values were formed against the null of a coefficient of zero.

Hard Court

The most statistically significant variables for hard court were 1. Break Point Attempt rate 2. Break Point Faced Rate 3. First Serve Win Rate

Grass Court

The most statistically significant variables for grass court were 1. Break Point Attempt Rate 2. Break Point Faced Rate 3. First Serve Win Rate

Clay Court

The most statistically significant variables for grass court were 1. Break Point Attempt Rate 2. Break Point Faced Rate 3. Second Serve Rate

Exmple of Interpreting a coefficent

My example: Break points faced rate for the clay courts multivariable regression

Holding all other variables constant, someone would expect as the mean break point faced rate increases by one percent, the average winning percentage of a player would decrease by 3.81 percentage points.

Errors

Possible Reasons for High P-values in Some Variables

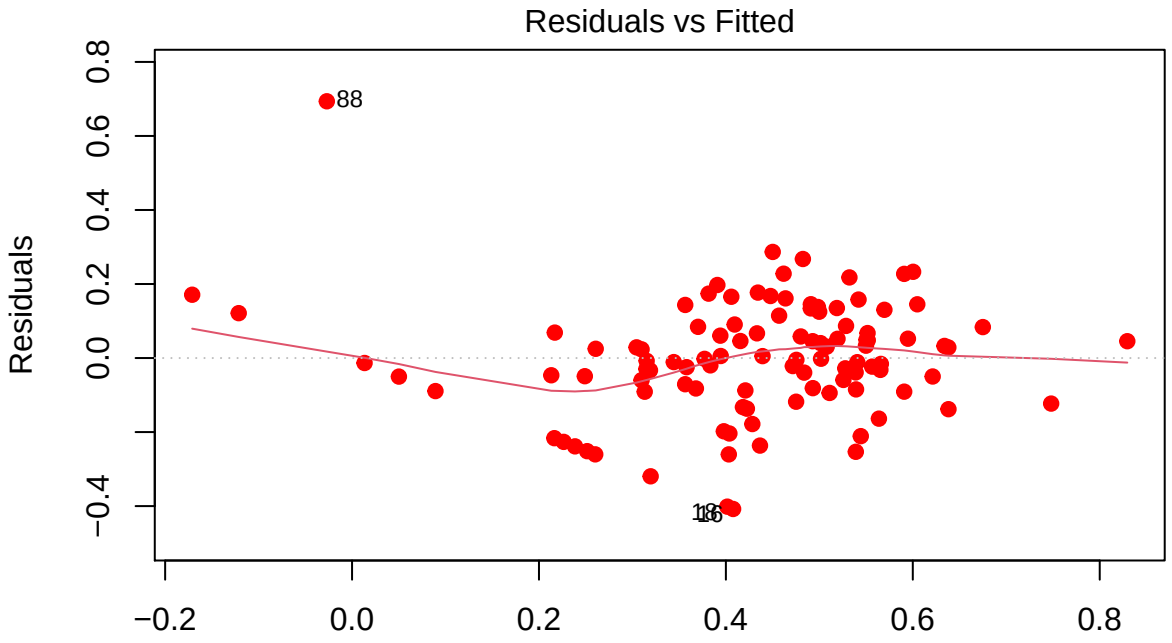
```
#rcorr(as.matrix(atp_data_player[2:13]))
#rcorr(as.matrix(atp_data_hard[2:13]))
#rcorr(as.matrix(atp_data_grass[2:13]))
#rcorr(as.matrix(atp_data_clay[2:13]))

# Used to Create Correlation Coefficent Table
```

As you can see in the correlation table, there was a high correlation between break points faced and second serve rate (-0.85 for hard, -0.69 for grass, and -0.73 for clay). This makes sense because one would expect that if someone wins a lot of their first serve points they would not face many break points and vice versa. This high correlation could cause their to be a high p-value in mean break points faced for hard courts, and first serve win rate for clay.

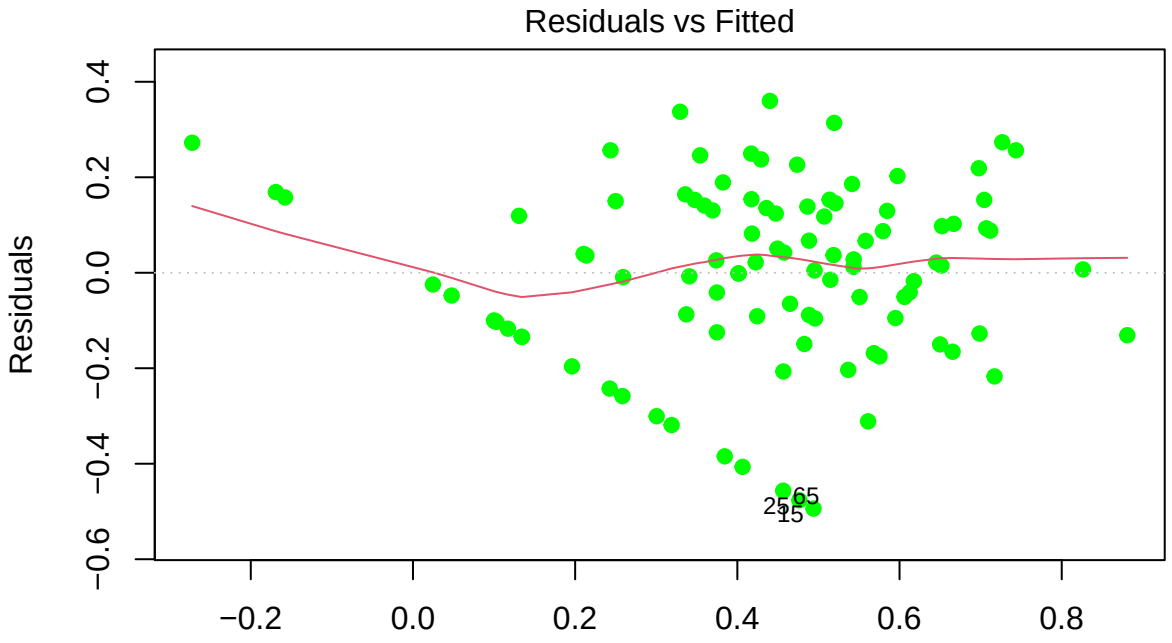
As for the 0.28 p-value for second serve rate in grass court, there was not really any other variables in the regression that it was highly associated with; hence, this obscurity is some we would further investigate if given more time.

Residual Plots

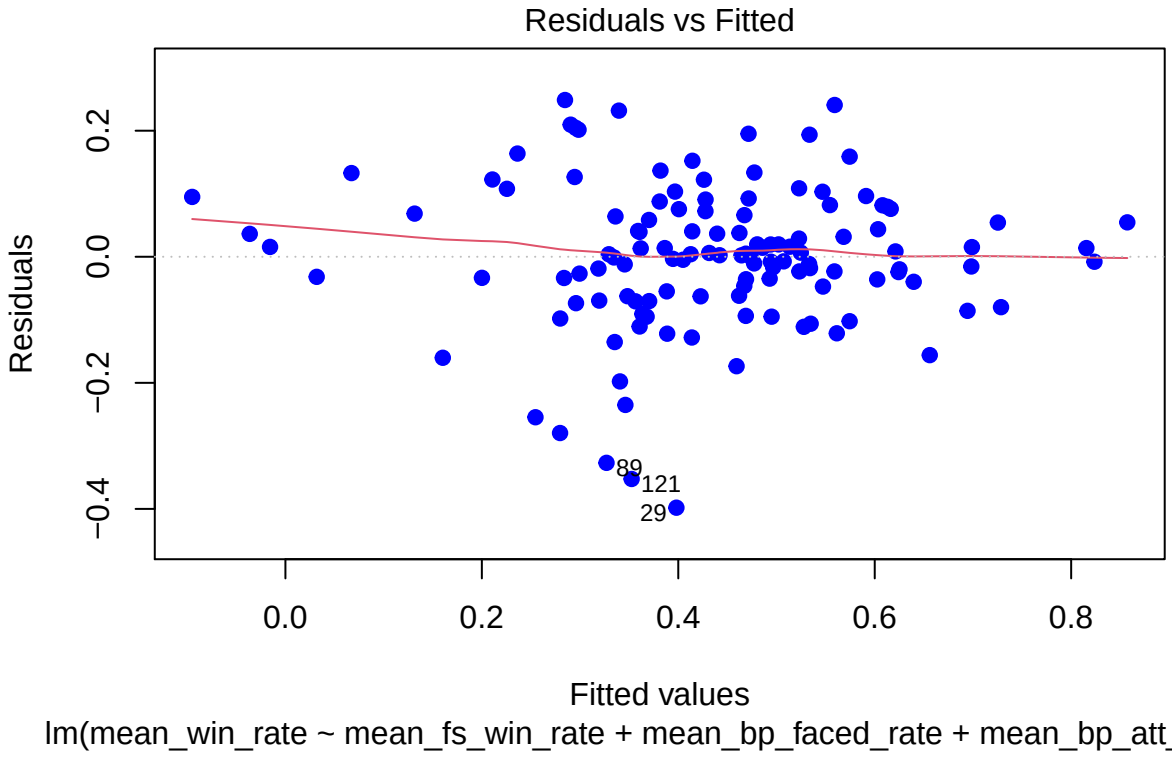
```
# Only show residual vs fitted

plot(multi_lm_atp_clay,
     pch = 19,
     col = "red",
     which = 1)
```




```
plot(multi_lm_atp_grass,
     col = "green",
     pch = 19,
     which = 1)
```



```r
plot(multi_lm_atp_hard,
     col = "blue",
     pch = 19,
     which = 1)
```



In our residual plot (Fitted Values vs Residuals) most of the values were evenly spread out, above and below zero, and the trend line is relatively straight, showing that a linear model is probably appropriate. However, in each plot there is a straight diagonal line of points. These are actually values of players who had a mean wining percentage of zero. Moreover, typically these players were lower ranked and as a result had to play the best players in world and hence, played fewer matches. We considered removing these players, but we decided not to because we had already limited our data to players in the top 100.

Problems with our Analysis and What we Would do given More Time

The largest problem with our analysis is that we had no groundstroke, net (i.e. approach shots, volley, and overhead), and other stroke data. Furthermore, we did not have tracking data, and in tennis tracking data is extremely important, because footwork, being one of, if not, the most important aspects of tennis, can only be analyzed given tracking data.

Some possible confounding variables in our study are the fact that most of data are from grandslams or ATP Masters (1000) tournaments. Another possible confounding is dominant hand. We wanted to separate the data by hand, but we realized that only 21 of the 128 players used in the analysis were left handed, thus we could not make accurate interpretations of the data.

For further explorations, we will look at the obscurities in the findings in this study, find a more complete data set, and adjust for more confounding variables.