

Chess Analysis

Mike Birlew

2023-02-11

```
ggplot(opening.wins, aes(x=winner, y= n, fill = winner)) +  
  geom_col() +  
  ylim(0,3500) +  
  labs(x = "Winner", y = "Wins", title = "Total Wins") +  
  theme(plot.title = element_text(hjust = 0.5),  
        axis.title.x = element_text(face="bold", colour="black", size = 12),  
        axis.title.y = element_text(face="bold", colour="black", size = 12))
```

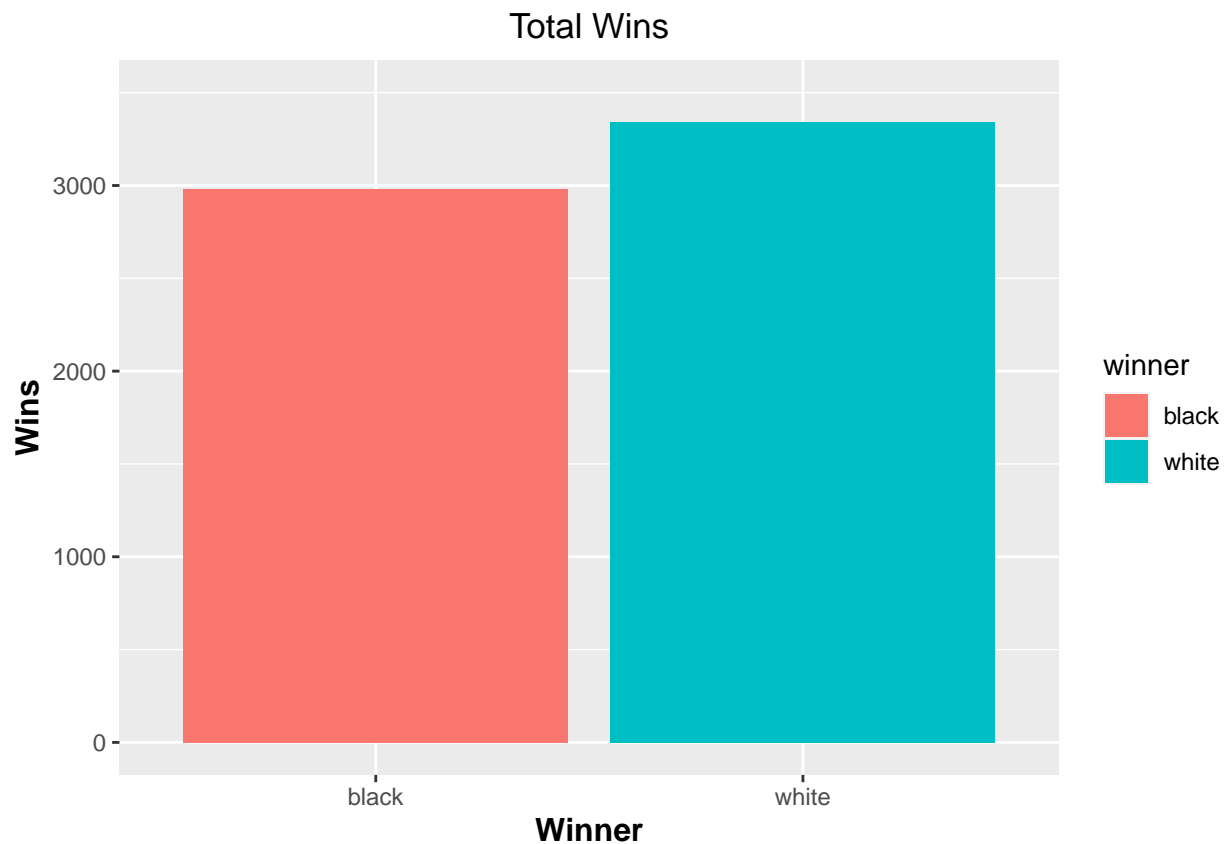


Figure 2. The box plot gives a better detail of the quantiles (the minimum here is 1) and median projection not seen in **Figure 1.** We can see more clearly that, over all, White does have a slight advantage but a majority of the games throughout are held equally. One exception is the outlier 'Black' move set #1 (A00) Polish (Sokolsky) opening.

Note: One thing that I find fascinating is the correlation to particular move sets that can give advantages to a player especially when playing as 'Black'. The dominate for black to win is move set #1 (A00) Polish

(Sokolsky) opening. If I were to try and improve my game I would study the top three opening move sets associated with the spikes of wins for 'White' and 'Black' and use them accordingly.

(Also while going through other graphs, I clearly see the constant advantage that 'White' has overall due to always moving first. But as we had discussed earlier in class, 'Black' actually does gain an overall advantage the longer the game progresses. These are very tiny gains but they show up consistently.)

```
# create number of matches win by eco per color
pivot_games <- pivot_wider(opening.wins, names_from = winner, values_from = n)
```

```
# creates data frame for black wins drops na
# adds total games and percent of moves to wins
black_win <- select(pivot_games, opening_eco, black) %>% drop_na() %>%
  mutate('total_games' = sum(black)) %>%
  mutate('percent_move_win' = (black/total_games)*100)
```

```
# creates data frame for white wins drops na
# adds total games and percent of moves to wins
white_win <- select(pivot_games, opening_eco, white) %>% drop_na() %>%
  mutate('total_games' = sum(white)) %>%
  mutate('percent_move_win' = (white/total_games)*100)
```

```
#summary_black <- c('opening_eco', 'black', 'percent_move_win')
#table_black <- summary(final_set_black[c('black', 'percent_move_win')])
#table_white <-summary(final_set_white[c('white', 'percent_move_win')])
```

```
#-----test
```

```
#summary_black <- c('opening_eco', 'black', 'percent_move_win')
table_black <- summary(black_win[c('black', 'percent_move_win')])
table_white <-summary(white_win[c('white', 'percent_move_win')])
```

```
# data objects d1 and d2 are from the previous code chunk
knitr::kables(
  list(
    # the first kable() to change column names
    kable(table_black, col.names = c('Black Wins', 'Percent to Win'),
      valign = 't'),
    # the second kable() to set the digits option
    kable(table_white, col.names = c('White Wins', 'Percent to Win'),
      valign = 't')),
  caption = 'Table 1. White/Black Win Summary (Total Games: 6325)')
```

```
# categorize for black to win
categorized_black <- cut(black_win$percent_move_win, breaks = c(0, .45,.94, 9.5),
  labels = c("small (<.45%)", "medium (.45-.9%)","large (1-9.5%)"))
```

```
# categorize for white to win
categorized_white <- cut(white_win$percent_move_win, breaks = c(0, .45,.94, 5.5),
  labels = c("small (<.45%)", "medium (.45-.9%)","large (1-5.5%)"))
```

Table 1: Table 1. White/Black Win Summary (Total Games: 6325)

Black Wins	Percent to Win	White Wins	Percent to Win
Min. : 1.00	Min. :0.03355	Min. : 1.00	Min. :0.02990
1st Qu.: 1.00	1st Qu.:0.03355	1st Qu.: 1.00	1st Qu.:0.02990
Median : 4.00	Median :0.13418	Median : 3.00	Median :0.08971
Mean : 13.87	Mean :0.46512	Mean : 13.82	Mean :0.41322
3rd Qu.: 12.00	3rd Qu.:0.40255	3rd Qu.: 11.00	3rd Qu.:0.32895
Max. :281.00	Max. :9.42637	Max. :171.00	Max. :5.11364

```
# create black category d.f.
final_set_black <- data.frame(black_win, categorized_black)

# create white category d.f.
final_set_white <- data.frame(white_win, categorized_white)

# plot black grouped wins
fig_1.1 <- ggplot(final_set_black, aes(x = categorized_black, fill = categorized_black)) +
  geom_bar() +
  labs(title = 'Chance of Black Win by Grouped Moveset', x = " ") +
  geom_text(aes(label=..count..), stat = 'count', vjust= 1) +
  scale_fill_brewer(palette="Set1")

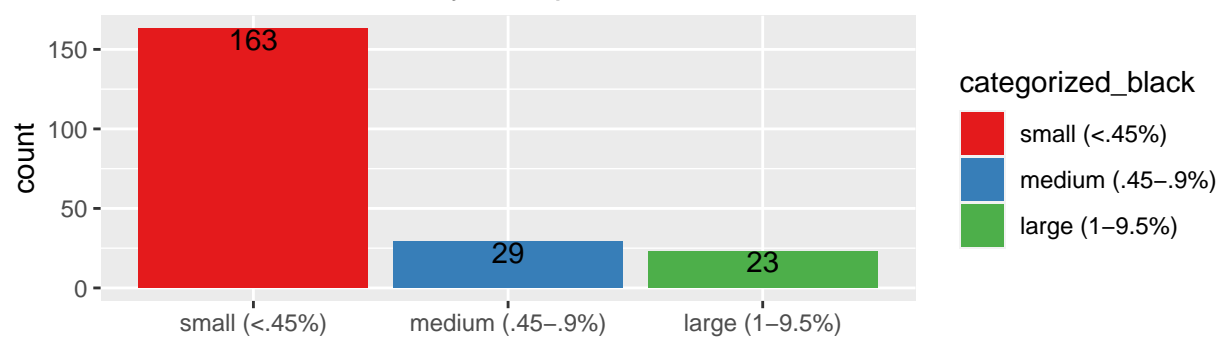
# plot white grouped wins
fig_1.2 <- ggplot(final_set_white, aes(x = categorized_white, fill = categorized_white)) +
  geom_bar() +
  labs(title = 'Chance of White Win by Grouped Moveset', x = " ") +
  geom_text(aes(label=..count..), stat = 'count', vjust=1) +
  scale_fill_brewer(palette="Set1")

#fig_1.2 + scale_fill_brewer(palette="Dark1")

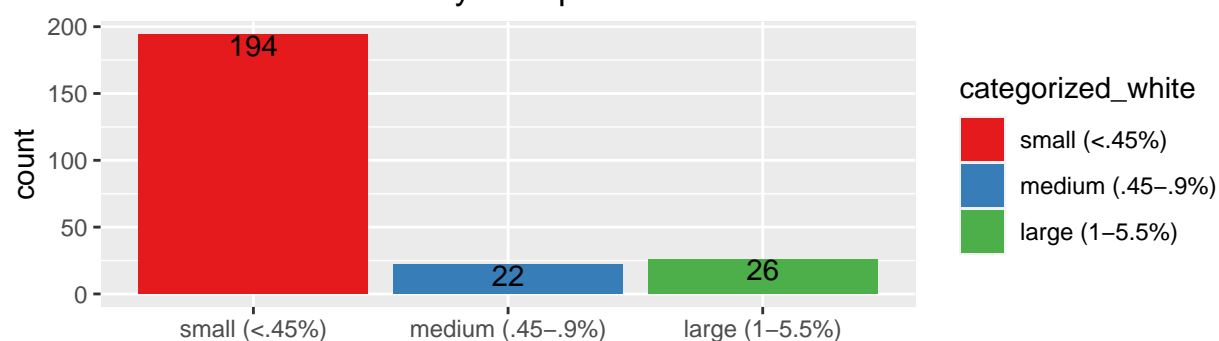
# combine white, black, plots
cowplot::plot_grid(fig_1.1, fig_1.2, nrow = 2)

## Warning: The dot-dot notation ('..count..') was deprecated in ggplot2 3.4.0.
## i Please use 'after_stat(count)' instead.
```

Chance of Black Win by Grouped Moveset



Chance of White Win by Grouped Moveset



```
most_win_white <- subset(final_set_white, percent_move_win > .45) %>%
  arrange(desc(white))
head(most_win_white)
```

```
## opening_eco white total_games percent_move_win categorized_white
## 1 54 171 3344 5.113636 large (1-5.5%)
## 2 123 160 3344 4.784689 large (1-5.5%)
## 3 155 159 3344 4.754785 large (1-5.5%)
## 4 1 135 3344 4.037081 large (1-5.5%)
## 5 138 129 3344 3.857656 large (1-5.5%)
## 6 31 122 3344 3.648325 large (1-5.5%)
```

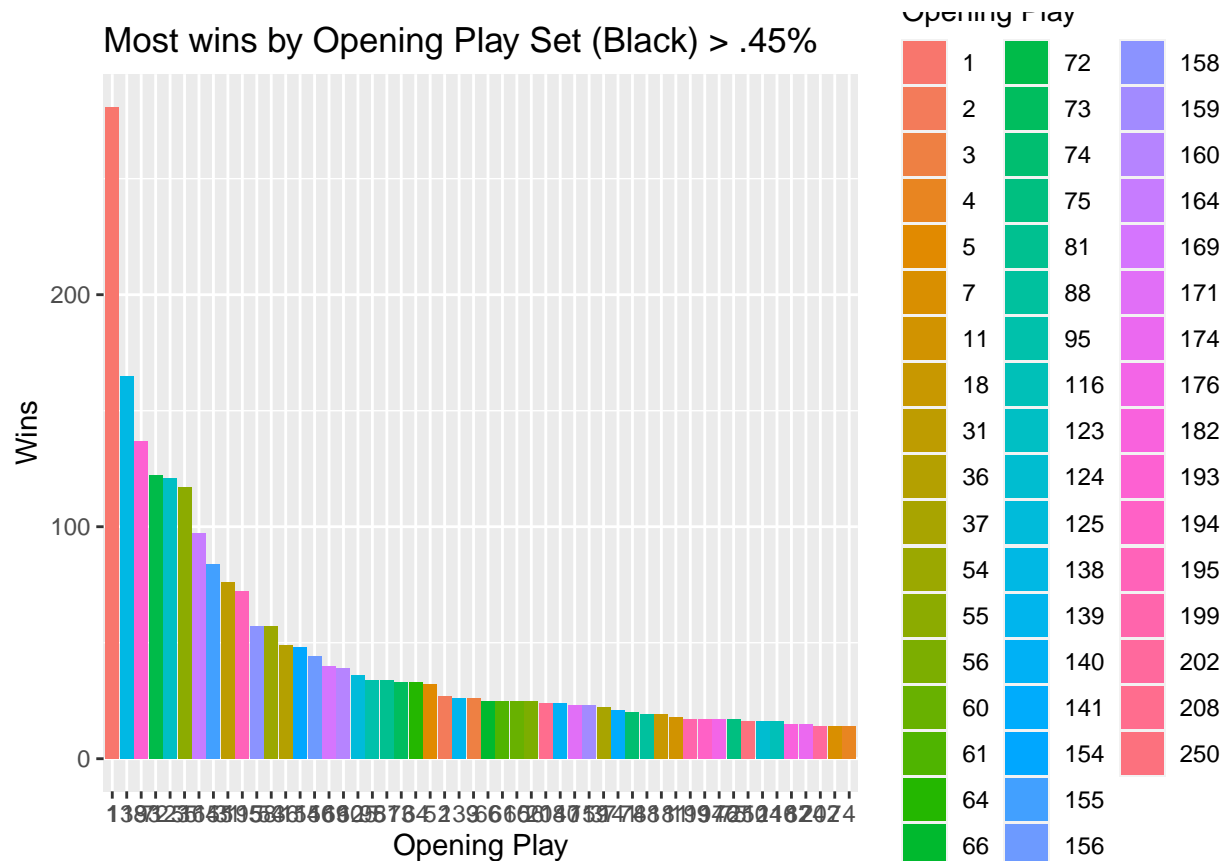
```
most_win_black <- subset(final_set_black, percent_move_win > .45)
```

```
head(most_win_black)
```

```
## opening_eco black total_games percent_move_win categorized_black
## 141 37 22 2981 0.7380074 medium (.45-.9%)
## 148 4 14 2981 0.4696411 medium (.45-.9%)
## 154 74 20 2981 0.6709158 medium (.45-.9%)
## 155 75 17 2981 0.5702784 medium (.45-.9%)
## 163 194 17 2981 0.5702784 medium (.45-.9%)
## 164 116 16 2981 0.5367326 medium (.45-.9%)
```

```
eco_percent <- ggplot(most_win_black,aes(fct_rev(fct_reorder(opening_eco, black)), black)) +
  geom_col(aes(fill = opening_eco)) +
  labs(x='Opening Play', y= 'Wins',
       title = 'Most wins by Opening Play Set (Black) > .45%', legend = 'Opening Play') +
  guides(fill=guide_legend(title = 'Opening Play'))
```

eco_percent



Appendix

```
library(tidyverse)
library(ChessPack)
library(ggplot2)
library(knitr)
library(readr)
library(cowplot)
library(kableExtra)
# turn opening plays into factors
chess.small<- chess.small %>%
  mutate(opening_eco=factor(opening_eco))
# wins vs opening moves
opening.wins <- chess.small %>% count(opening_eco, winner, sort = TRUE)%>%
  arrange(n)
ggplot(opening.wins, aes(x=winner, y= n, fill = winner)) +
  geom_col() +
  ylim(0,3500) +
```

```

labs(x = "Winner", y = "Wins", title = "Total Wins") +
  theme(plot.title = element_text(hjust = 0.5),
        axis.title.x = element_text(face="bold", colour="black", size = 12),
        axis.title.y = element_text(face="bold", colour="black", size = 12))
# Dont need!!
#chess.sort <- chess.small %>%
# mutate(sorted_eco = sort.int(opening_eco))
# create number of matches win by eco per color
pivot_games <- pivot_wider(opening_wins, names_from = winner, values_from = n)
# creates data frame for black wins drops na
# adds total games and percent of moves to wins
black_win <- select(pivot_games, opening_eco, black) %>% drop_na() %>%
  mutate('total_games' = sum(black)) %>%
  mutate('percent_move_win' = (black/total_games)*100)

# creates data frame for white wins drops na
# adds total games and percent of moves to wins
white_win <- select(pivot_games, opening_eco, white) %>% drop_na() %>%
  mutate('total_games' = sum(white)) %>%
  mutate('percent_move_win' = (white/total_games)*100)
#summary_black <- c('opening_eco', 'black', 'percent_move_win')
#table_black <- summary(final_set_black[c('black', 'percent_move_win')])
#table_white <-summary(final_set_white[c('white', 'percent_move_win')])

#-----test

#summary_black <- c('opening_eco', 'black', 'percent_move_win')
table_black <- summary(black_win[c('black', 'percent_move_win')])
table_white <-summary(white_win[c('white', 'percent_move_win')])

# data objects d1 and d2 are from the previous code chunk
knitr::kables(
  list(
    # the first kable() to change column names
    kable(table_black, col.names = c('Black Wins', 'Percent to Win'),
          valign = 't'),
    # the second kable() to set the digits option
    kable(table_white, col.names = c('White Wins', 'Percent to Win'),
          valign = 't')),
  caption = 'Table 1. White/Black Win Summary (Total Games: 6325)')
# categorize for black to win
categorized_black <- cut(black_win$percent_move_win, breaks = c(0, .45,.94, 9.5),
  labels = c("small (<.45%)", "medium (.45-.9%)","large (1-9.5%)"))
# categorize for white to win
categorized_white <- cut(white_win$percent_move_win, breaks = c(0, .45,.94, 5.5),
  labels = c("small (<.45%)", "medium (.45-.9%)","large (1-5.5%)"))
# create black category d.f.
final_set_black <- data.frame(black_win, categorized_black)
# create white category d.f.
final_set_white <- data.frame(white_win, categorized_white)
# plot black grouped wins
fig_1.1 <- ggplot(final_set_black, aes(x = categorized_black,fill = categorized_black)) +

```

```

geom_bar() +
  labs(title = 'Chance of Black Win by Grouped Moveset', x = " ") +
  geom_text(aes(label=..count..), stat = 'count', vjust= 1) +
  scale_fill_brewer(palette="Set1")
# plot white grouped wins
fig_1.2 <- ggplot(final_set_white, aes(x = categorized_white, fill = categorized_white)) +
  geom_bar() +
  labs(title = 'Chance of White Win by Grouped Moveset', x = " ") +
  geom_text(aes(label=..count..), stat = 'count', vjust=1) +
  scale_fill_brewer(palette="Set1")

#fig_1.2 + scale_fill_brewer(palette="Dark1")
# combine white, black, plots
cowplot::plot_grid(fig_1.1, fig_1.2, nrow = 2)
most_win_white <- subset(final_set_white, percent_move_win > .45) %>%
  arrange(desc(white))
head(most_win_white)
most_win_black <- subset(final_set_black, percent_move_win > .45)

head(most_win_black)
eco_percent <- ggplot(most_win_black, aes(fct_rev(fct_reorder(opening_eco, black)), black)) +
  geom_col(aes(fill = opening_eco)) +
  labs(x='Opening Play', y= 'Wins',
       title = 'Most wins by Opening Play Set (Black) > .45%', legend = 'Opening Play') +
  guides(fill=guide_legend(title = 'Opening Play'))

eco_percent

```