# Player Style Clustering

Group E

12/9/2021

```
knitr::opts_chunk$set(message = FALSE, warning = FALSE)
```

I am following the cluster analysis steps from this **ATP cluster analysis**. The goal is to compare the results from this WTA cluster analysis to the results in the ATP cluster analysis to explore how playing styles might differ.

```
library(naniar)
# from https://cran.r-project.org/web/packages/naniar/vignettes/replace-with-na.html
tennis_results <- tennis_results %>%
  replace_with_na(replace = list(w_SvGms = 0, l_SvGms = 0, minutes = 0))

# selecting matches from 2011 onward
# creating the statistics that they mentioned in the table
playerstyle_WTA_cluster <- tennis_results %>%
  filter(year >= 2011, tour == "WTA", !is.na(w_1stIn), !is.na(w_svpt), !is.na(l_1stIn), !is.na(l_svpt),
  mutate(w_1stsvpct = w_1stIn/w_svpt,
         l_1stsvpct = l_1stIn/l_svpt,
         w_svpctWon = (w_1stWon+w_2ndWon)/w_svpt,
         l_svpctWon = (l_1stWon+l_2ndWon)/l_svpt,
         w_1stsvWon = w_1stWon/w_1stIn,
         l_1stsvWon = l_1stWon/l_1stIn,
         w_2ndsvWon = w_2ndWon/(w_svpt-w_1stIn),
         l_2ndsvWon = l_2ndWon/(l_svpt-l_1stIn),
         w_acepct = w_ace/w_svpt,
         l_acepct = l_ace/l_svpt,
         w_dfpct = w_df/w_svpt,
         l_dfpct = l_df/l_svpt,
         w_ptspersvgame = w_svpt/w_SvGms,
         l_ptspersvgame = l_svpt/l_SvGms,
         w_bpSavepct = w_bpSaved/w_bpFaced,
         l_bpSavepct = l_bpSaved/l_bpFaced,
         w_bppersvgame = w_bpFaced/w_SvGms,
         l_bppersvgame = l_bpFaced/l_SvGms,
         w_pctptWon = (w_1stWon+w_2ndWon+(l_svpt-l_1stWon-l_2ndWon))/(w_svpt+l_svpt),
         l_pctptWon = (l_1stWon+l_2ndWon+(w_svpt-w_1stWon-w_2ndWon))/(w_svpt+l_svpt),
         w_1stretWon = (l_1stIn-l_1stWon)/l_1stIn,
         l_1stretWon = (w_1stIn-w_1stWon)/w_1stIn,
         w_2ndretWon = (l_svpt-l_1stIn-l_2ndWon)/(l_svpt-l_1stIn),
         l_2ndretWon = (w_svpt-w_1stIn-w_2ndWon)/(w_svpt-w_1stIn),
         w_retpctWon = 1-(l_1stWon+l_2ndWon)/l_svpt,
         l_retpctWon = 1-(w_1stWon+w_2ndWon)/w_svpt,
         w_ptsperretgame = l_svpt/l_SvGms,
         l_ptsperretgame = w_svpt/w_SvGms,
         w_bpConvpct = 1-l_bpSaved/l_bpFaced,
         l_bpConvpct = 1-w_bpSaved/w_bpFaced,
         w_bpperretgame = l_bpFaced/l_SvGms,
         l_bpperretgame = w_bpFaced/w_SvGms,
         w_retace = l_ace/l_svpt,
         l_retace = w_ace/w_svpt,
         w_retdf = l_df/l_svpt,
         l_retdf = w_df/w_svpt,
         ptspermin = (w_svpt+l_svpt)/minutes) %>%
  select(1:30, 49:89)
```

```
# creating two rows for winner/loser
cluster_w <- playerstyle_WTA_cluster %>%
  select(tour, tourney_id, tourney_name, surface, tourney_date, year, month, winner_id, winner_name, win
  mutate(result = 1)

colnames(cluster_w) <- c("tour", "tourney_id", "tourney_name", "surface", "tourney_date", "year", "month

cluster_l <- playerstyle_WTA_cluster %>%
  select(tour, tourney_id, tourney_name, surface, tourney_date, year, month, loser_id, loser_name, lose
  mutate(result = 0)

colnames(cluster_l) <- c("tour", "tourney_id", "tourney_name", "surface", "tourney_date", "year", "month

final_cluster_data <- rbind(cluster_w,cluster_l)
```

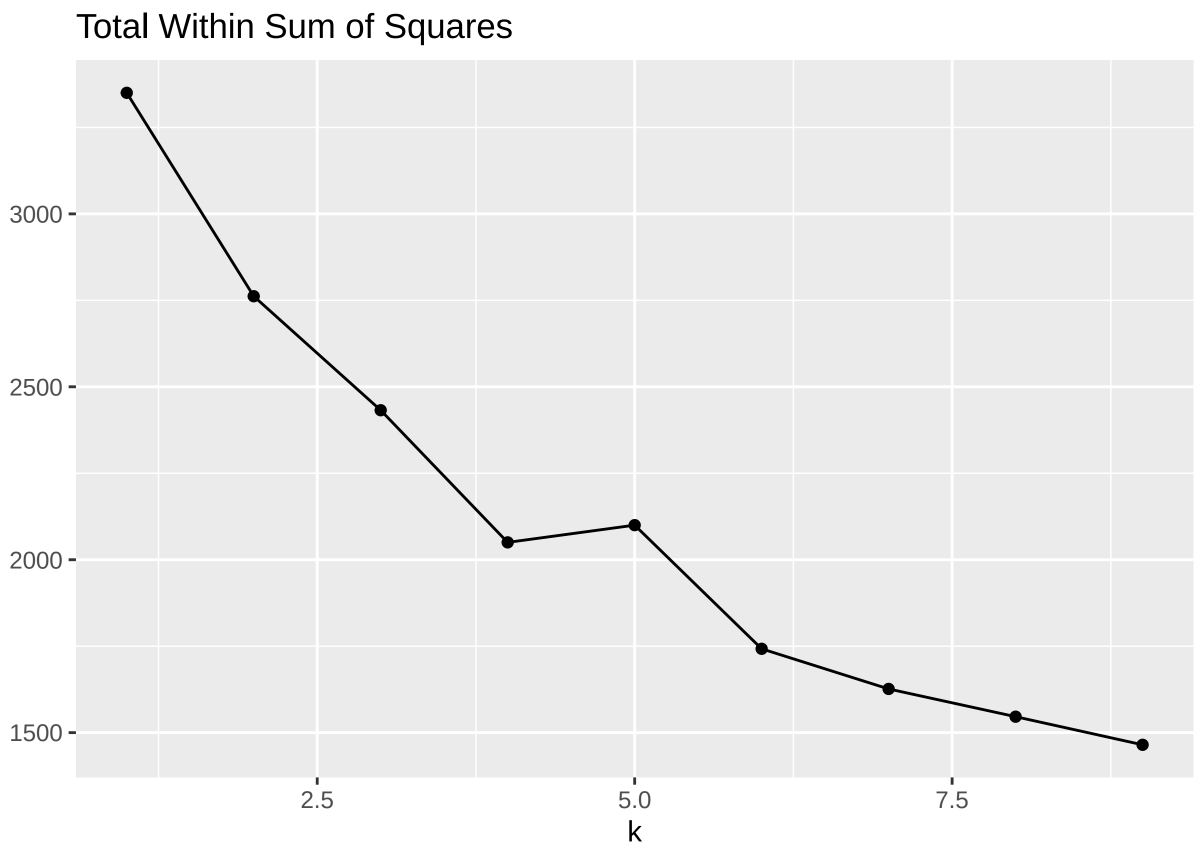
```
# calculating stats by player
WTA_player_stats <- final_cluster_data %>%
  group_by(id, name) %>%
  summarize(height = mean(height),
            age = max(age),
            win_perc = mean(result),
            perc_points_won = mean(pctptWon, na.rm = TRUE),
            "1st_serv_perc" = mean(`1stsvpct`, na.rm = TRUE),
            "1st_win" = mean(`1stsvWon`, na.rm = TRUE),
            ace_perc = mean(acepct, na.rm = TRUE),
            df_perc = mean(dfpct, na.rm = TRUE),
            "2nd_win" = mean(`2ndsvWon`, na.rm = TRUE),
            svc_perc_win = mean(svpctWon, na.rm = TRUE),
            points_per_svc_game = mean(ptspersvgame, na.rm = TRUE),
            break_point_save_perc = mean(bpSavepct, na.rm = TRUE),
            bp_per_game = mean(bppersvgame, na.rm = TRUE),
            return_1st_win = mean(`1stretWon`, na.rm = TRUE),
            return_ace_perc = mean(retace, na.rm = TRUE),
            return_df_perc = mean(retdf, na.rm = TRUE),
            return_2nd_win = mean(`2ndretWon`, na.rm = TRUE),
            return_perc_win = mean(retpctWon, na.rm = TRUE),
            points_per_return_game = mean(ptsperretgame, na.rm = TRUE),
            bp_convert_perc = mean(bpConvpct, na.rm = TRUE),
            return_bp_per_game = mean(bpperretgame, na.rm = TRUE),
            points_per_minute = mean(ptspermin, na.rm = TRUE))
```

```r
surface_stats <- final_cluster_data %>%
  group_by(id, name, surface) %>%
  summarize(count = n()) %>%
  mutate(freq = count / sum(count)) %>%
  pivot_wider(id_cols = c(id, name), names_from = surface, values_from = freq) %>%
  summarize(clay_perc = Clay, grass_perc = Grass, hard_perc = Hard) %>%
  select(-2)

final_WTA <- cbind(surface_stats, WTA_player_stats) %>%
  select(-1) %>%
  rename(id = id...5) %>%
  select(4:7, 1:3, 8:27)
```

```r
library(broom)
# cluster analysis
final_WTA_km <- final_WTA %>%
  drop_na() %>%
  select(height:points_per_minute) %>%
  mutate(across(height:points_per_minute, scale))

set.seed(13)
final_WTA_kclusts <-
  tibble(k = 1:9) %>%
  mutate(final_WTA_kclust = map(k, ~kmeans(final_WTA_km, .x)),
    glanced = map(final_WTA_kclust, glance),
    tidied = map(final_WTA_kclust, tidy),
    augmented = map(final_WTA_kclust, augment, final_WTA_km)
  )
```

```r
clusters <-
  final_WTA_kclusts %>%
  unnest(cols = c(tidied))

assignments <-
  final_WTA_kclusts %>%
  unnest(cols = c(augmented))

clusterings <-
  final_WTA_kclusts %>%
  unnest(cols = c(glanced))
```

```r
clusterings %>%
  ggplot(aes(x = k, y = tot.withinss)) +
  geom_line() +
  geom_point() + ylab("") +
  ggtitle("Total Within Sum of Squares")
```



```
set.seed(47)
WTA_clustered <- final_WTA_km %>%
  kmeans(centers = 4)

WTA_clusters <- cbind(WTA_clustered$cluster, final_WTA %>% drop_na())
```

```
WTA_clusters %>%
  rename(cluster = `WTA_clustered$cluster`) %>%
  group_by(cluster) %>%
  select(cluster, height:points_per_minute) %>%
  summarise_each(funs(mean(., na.rm = TRUE))) %>%
  mutate(cluster = as.factor(cluster)) %>%
  mutate(cluster = recode(cluster, "1" = "between tours", "2" = "top players", "3" = "strong servers (A
```

```
## # A tibble: 4 x 26
##   cluster   height   age clay_perc grass_perc hard_perc win_perc perc_points_won
##   <fct>      <dbl> <dbl>     <dbl>      <dbl>     <dbl>    <dbl>           <dbl>
## 1 between ~   170.  26.5     0.430     0.0916     0.479    0.450           0.490
## 2 top play~   177.  26.6     0.237     0.106      0.657    0.633           0.522
## 3 strong s~   172.  30.8     0.213     0.103      0.684    0.500           0.502
## 4 tier 2      177.  30.4     0.289     0.152      0.559    0.330           0.471
## # ... with 18 more variables: 1st_serv_perc <dbl>, 1st_win <dbl>,
## #   ace_perc <dbl>, df_perc <dbl>, 2nd_win <dbl>, svc_perc_win <dbl>,
## #   points_per_svc_game <dbl>, break_point_save_perc <dbl>, bp_per_game <dbl>,
## #   return_1st_win <dbl>, return_ace_perc <dbl>, return_df_perc <dbl>,
## #   return_2nd_win <dbl>, return_perc_win <dbl>, points_per_return_game <dbl>,
## #   bp_convert_perc <dbl>, return_bp_per_game <dbl>, points_per_minute <dbl>
```

```
# from https://stackoverflow.com/questions/21807987/calculate-the-mean-for-each-column-of-a-matrix-in-r
```

### ATP Replication

```
library(naniar)
# from https://cran.r-project.org/web/packages/naniar/vignettes/replace-with-na.html
tennis_results <- tennis_results %>%
  replace_with_na(replace = list(w_SvGms = 0, l_SvGms = 0, minutes = 0))

# selecting matches from 2011 onward
# creating the statistics that they mentioned in the table
playerstyle_ATP_cluster <- tennis_results %>%
  filter(year >= 2011, tour == "ATP", !is.na(w_1stIn), !is.na(w_svpt), !is.na(l_1stIn), !is.na(l_svpt),
  mutate(w_1stsvpct = w_1stIn/w_svpt,
         l_1stsvpct = l_1stIn/l_svpt,
         w_svpctWon = (w_1stWon+w_2ndWon)/w_svpt,
         l_svpctWon = (l_1stWon+l_2ndWon)/l_svpt,
         w_1stsvWon = w_1stWon/w_1stIn,
         l_1stsvWon = l_1stWon/l_1stIn,
         w_2ndsvWon = w_2ndWon/(w_svpt-w_1stIn),
         l_2ndsvWon = l_2ndWon/(l_svpt-l_1stIn),
         w_acepct = w_ace/w_svpt,
         l_acepct = l_ace/l_svpt,
         w_dfpct = w_df/w_svpt,
         l_dfpct = l_df/l_svpt,
         w_ptspersvgame = w_svpt/w_SvGms,
         l_ptspersvgame = l_svpt/l_SvGms,
         w_bpSavepct = w_bpSaved/w_bpFaced,
         l_bpSavepct = l_bpSaved/l_bpFaced,
         w_bppersvgame = w_bpFaced/w_SvGms,
         l_bppersvgame = l_bpFaced/l_SvGms,
         w_pctptWon = (w_1stWon+w_2ndWon+(l_svpt-l_1stWon-l_2ndWon))/(w_svpt+l_svpt),
         l_pctptWon = (l_1stWon+l_2ndWon+(w_svpt-w_1stWon-w_2ndWon))/(w_svpt+l_svpt),
         w_1stretWon = (l_1stIn-l_1stWon)/l_1stIn,
         l_1stretWon = (w_1stIn-w_1stWon)/w_1stIn,
         w_2ndretWon = (l_svpt-l_1stIn-l_2ndWon)/(l_svpt-l_1stIn),
         l_2ndretWon = (w_svpt-w_1stIn-w_2ndWon)/(w_svpt-w_1stIn),
         w_retpctWon = 1-(l_1stWon+l_2ndWon)/l_svpt,
         l_retpctWon = 1-(w_1stWon+w_2ndWon)/w_svpt,
         w_ptsperretgame = l_svpt/l_SvGms,
         l_ptsperretgame = w_svpt/w_SvGms,
         w_bpConvpct = 1-l_bpSaved/l_bpFaced,
         l_bpConvpct = 1-w_bpSaved/w_bpFaced,
         w_bpperretgame = l_bpFaced/l_SvGms,
         l_bpperretgame = w_bpFaced/w_SvGms,
         w_retace = l_ace/l_svpt,
         l_retace = w_ace/w_svpt,
         w_retdf = l_df/l_svpt,
         l_retdf = w_df/w_svpt,
        ptspermin = (w_svpt+l_svpt)/minutes) %>%
  select(1:30, 49:89)
```

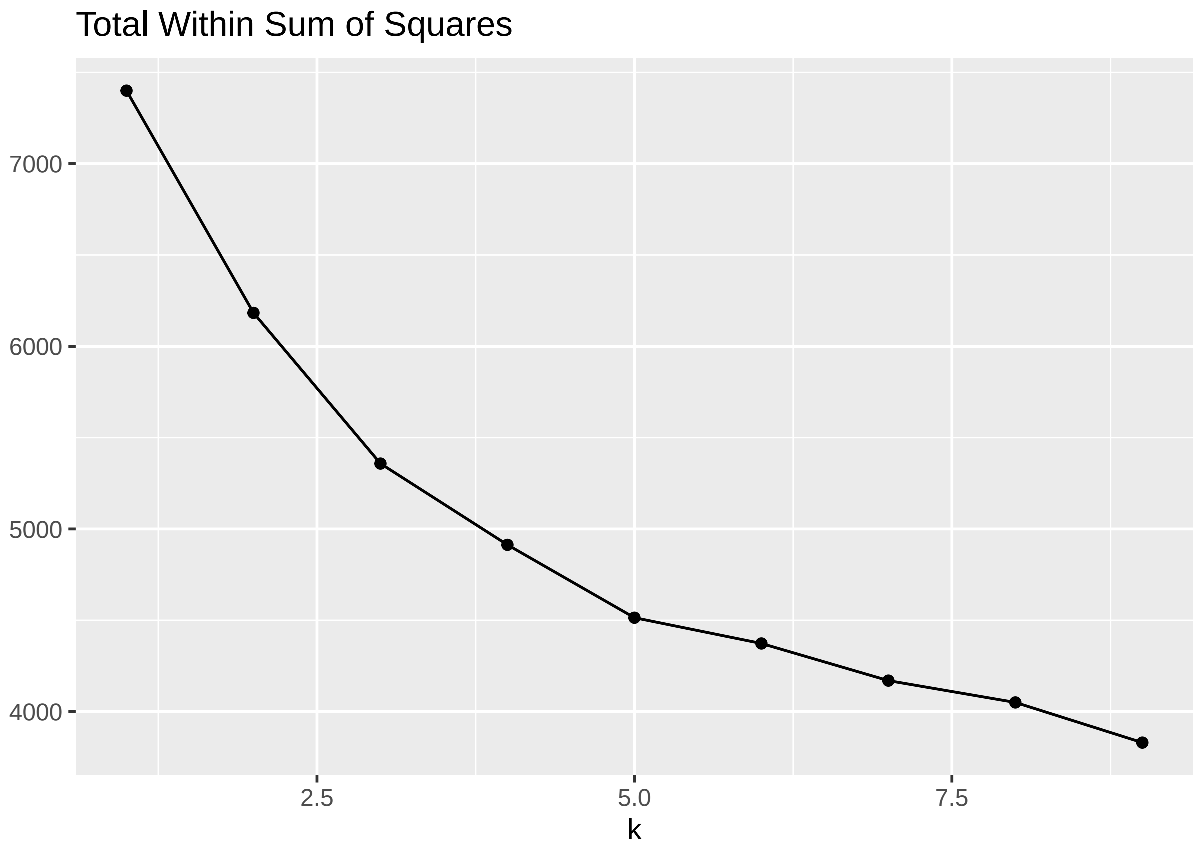
```
# creating two rows for winner/loser
cluster_w_m <- playerstyle_ATP_cluster %>%
  select(tour, tourney_id, tourney_name, surface, tourney_date, year, month, winner_id, winner_name, wi
  mutate(result = 1)

colnames(cluster_w_m) <- c("tour", "tourney_id", "tourney_name", "surface", "tourney_date", "year", "mo

cluster_l_m <- playerstyle_ATP_cluster %>%
  select(tour, tourney_id, tourney_name, surface, tourney_date, year, month, loser_id, loser_name, lose
  mutate(result = 0)

colnames(cluster_l_m) <- c("tour", "tourney_id", "tourney_name", "surface", "tourney_date", "year", "mo

final_cluster_data_m <- rbind(cluster_w_m,cluster_l_m)
```

```
# calculating stats by player
ATP_player_stats <- final_cluster_data_m %>%
  group_by(id, name) %>%
  summarize(height = mean(height), age = max(age), win_perc = mean(result), perc_points_won = mean(pctp

surface_stats_m <- final_cluster_data_m %>%
  group_by(id, name, surface) %>%
  summarize(count = n()) %>%
  mutate(freq = count / sum(count)) %>%
  pivot_wider(id_cols = c(id, name), names_from = surface, values_from = freq) %>%
  summarize(clay_perc = Clay, grass_perc = Grass, hard_perc = Hard) %>%
  select(-2)

final_ATP <- cbind(surface_stats_m, ATP_player_stats) %>%
  select(-1) %>%
  rename(id = id...5) %>%
  select(4:7, 1:3, 8:27)
```

```
# cluster analysis
final_ATP_km <- final_ATP %>%
  drop_na() %>%
  select(height:points_per_minute) %>%
  mutate(across(height:points_per_minute, scale))

set.seed(7)
final_ATP_kclusts <-
  tibble(k = 1:9) %>%
  mutate(final_ATP_kclust = map(k, ~kmeans(final_ATP_km, .x)),
    glanced = map(final_ATP_kclust, glance),
    tidied = map(final_ATP_kclust, tidy),
    augmented = map(final_ATP_kclust, augment, final_ATP_km)
  )
```

```
clusters_m <-
  final_ATP_kclusts %>%
  unnest(cols = c(tidied))

assignments_m <-
  final_ATP_kclusts %>%
  unnest(cols = c(augmented))

clusterings_m <-
  final_ATP_kclusts %>%
  unnest(cols = c(glanced))
```

```
clusterings_m %>%
  ggplot(aes(x = k, y = tot.withinss)) +
  geom_line() +
  geom_point() + ylab("") +
  ggtitle("Total Within Sum of Squares")
```



```
set.seed(47)
ATP_clustered <- final_ATP_km %>%
  kmeans(centers = 4) # use 4 to compare to WTA

ATP_clusters <- cbind(ATP_clustered$cluster, final_ATP %>% drop_na())
ATP_clusters %>%
  rename(cluster = `ATP_clustered$cluster`) %>%
  group_by(cluster) %>%
  select(cluster, height:points_per_minute) %>%
  summarise_each(funs(mean(., na.rm = TRUE))) %>%
  mutate(cluster = as.factor(cluster)) %>%
   mutate(cluster = recode(cluster, "1" = "tier 2", "2" = "top players", "3" = "between tours", "4" = "s
```

```
## # A tibble: 4 x 26
##   cluster   height   age clay_perc grass_perc hard_perc win_perc perc_points_won
##   <fct>      <dbl> <dbl>     <dbl>      <dbl>     <dbl>    <dbl>           <dbl>
## 1 tier 2      184.  31.9     0.261     0.177      0.559    0.284           0.465
## 2 top play~   186.  30.4     0.291     0.0895     0.614    0.610           0.517
## 3 between ~   184.  31.2     0.420     0.0963     0.482    0.402           0.487
## 4 strong s~   192.  30.3     0.215     0.130      0.651    0.465           0.496
## # ... with 18 more variables: 1st_serv_perc <dbl>, 1st_win <dbl>,
## #   ace_perc <dbl>, df_perc <dbl>, 2nd_win <dbl>, svc_perc_win <dbl>,
## #   points_per_svc_game <dbl>, break_point_save_perc <dbl>, bp_per_game <dbl>,
## #   return_1st_win <dbl>, return_ace_perc <dbl>, return_df_perc <dbl>,
## #   return_2nd_win <dbl>, return_perc_win <dbl>, points_per_return_game <dbl>,
## #   bp_convert_perc <dbl>, return_bp_per_game <dbl>, points_per_minute <dbl>
```

```
# from https://stackoverflow.com/questions/21807987/calculate-the-mean-for-each-column-of-a-matrix-in-r
```

```
# combining the two datasets
ATP_cluster_results <- ATP_clusters %>%
  rename(cluster = `ATP_clustered$cluster`) %>%
  group_by(cluster) %>%
  select(cluster, height:points_per_minute) %>%
  summarise_each(funs(mean(., na.rm = TRUE))) %>%
  mutate(tour = "ATP",
         cluster = as.factor(cluster)) %>%
  mutate(cluster = recode(cluster, "1" = "tier 2", "2" = "top players",
                          "3" = "between tours",
                          "4" = "strong servers (ATP)/\nstrong returners (WTA)")) %>%
  select(27, 1:26)

WTA_cluster_results <- WTA_clusters %>%
  rename(cluster = `WTA_clustered$cluster`) %>%
  group_by(cluster) %>%
  select(cluster, height:points_per_minute) %>%
  summarise_each(funs(mean(., na.rm = TRUE))) %>%
  mutate(tour = "WTA",
         cluster = as.factor(cluster)) %>%
  mutate(cluster = recode(cluster, "1" = "between tours", "2" = "top players",
                         "3" = "strong servers (ATP)/\nstrong returners (WTA)",
                         "4" = "tier 2")) %>%
  select(27, 1:26)

all_cluster_results <- rbind(ATP_cluster_results, WTA_cluster_results)
```