STAT 231: Problem Set 10B

Muhammad Ahsan Tahir

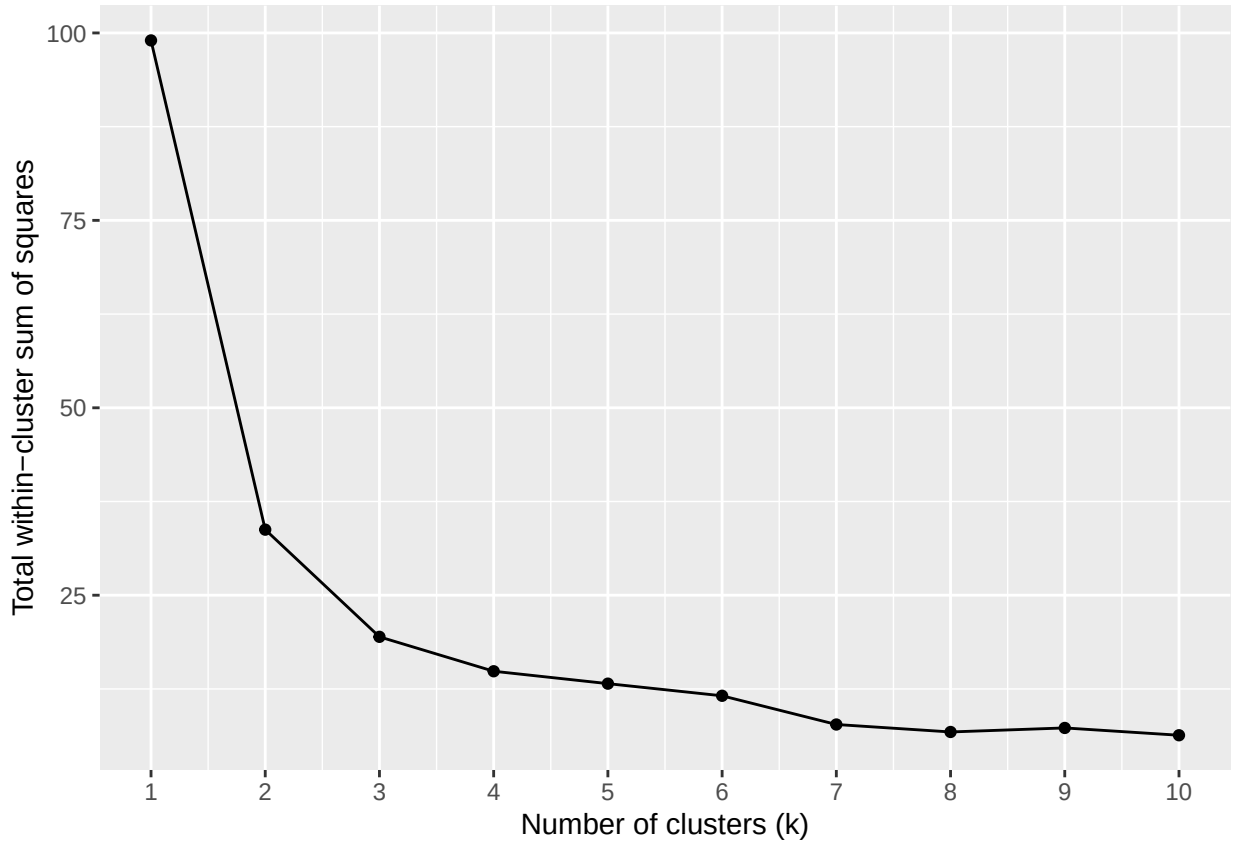
```
load("MuhammadAhsanTahir.RData")
```

```
groupedTable <- attacking_table %>%
  select(Squad, MP, W, GF, GA, Pts) %>%
  group_by(Squad) %>%
  summarise(totalMatches = sum(MP),
            totalWins = sum(W),
            totalGoals = sum(GF),
            totalPoints = sum(Pts),
            totalGoalsConceded = sum(GA)) %>%
  mutate(WinPercent = totalWins/totalMatches * 100,
         GoalsPerMatch = totalGoals/totalMatches,
         PointsPerMatch = totalPoints/totalMatches,
         GoalsConcededPerMatch = totalGoalsConceded/totalMatches
         ) %>%
  select(-totalMatches, -totalWins, -totalGoals, -totalPoints, -totalGoalsConceded)

groupedTable
```

```
## # A tibble: 34 x 5
##    Squad          WinPercent GoalsPerMatch PointsPerMatch GoalsConcededPerMatch
##    <chr>               <dbl>         <dbl>          <dbl>                 <dbl>
##  1 Arsenal              53.7         1.77           1.82                  1.13
##  2 Aston Villa          26.7         1.10           1.01                  1.62
##  3 Bournemouth          29.5         1.27           1.11                  1.74
##  4 Brentford            34.2         1.26           1.21                  1.47
##  5 Brighton             25.3         1              1.1                   1.36
##  6 Burnley              28.2         0.970          1.11                  1.40
##  7 Cardiff City         22.4         0.868          0.842                 1.88
##  8 Chelsea              57.1         1.82           1.94                  1.01
##  9 Crystal Palace       31.6         1.13           1.18                  1.41
## 10 Everton              38.4         1.36           1.42                  1.32
## # ... with 24 more rows
```

```
# set the seed for reproducibility
set.seed(1877090)

tableForClustering <- groupedTable %>%
  select(WinPercent, GoalsPerMatch, GoalsConcededPerMatch)%>%
  mutate(across(where(is.numeric),  ~scale(.)[,1], .names = "{.col}_scaled")) %>%
  select(WinPercent_scaled, GoalsPerMatch_scaled, GoalsConcededPerMatch_scaled)
```

```r
# Iterate through clustering algorithm for 10 different values of k
elbow_plot <- tibble(k = 1:10) %>%
  mutate(
    # List-column of 10 kmeans objects
    # (apply `kmeans()` to each value of `k`)
    kmeans_results = purrr::map(k, ~kmeans(tableForClustering, .x)),
    # List-column of "glanced" model summaries for each kmeans object
    # (apply `glance()` to each corresponding result after running `kmeans()`)
    glanced = purrr::map(kmeans_results, glance)) %>%
  # Turn `glanced` list-column into regular tibble columns
  unnest(cols = c(glanced))

# Construct elbow plot
ggplot(elbow_plot, aes(x = k, y = tot.withinss)) +
  geom_point() +
  geom_line() +
  scale_x_continuous(breaks = 1:10) +
  labs(x = "Number of clusters (k)",
       y = "Total within-cluster sum of squares")
```




```r
meanGoalsPerMatch <- mean(groupedTable$GoalsPerMatch)
meanGoalsConcededPerMatch <- mean(groupedTable$GoalsConcededPerMatch)
meanWinPercent <- mean(groupedTable$WinPercent)
```

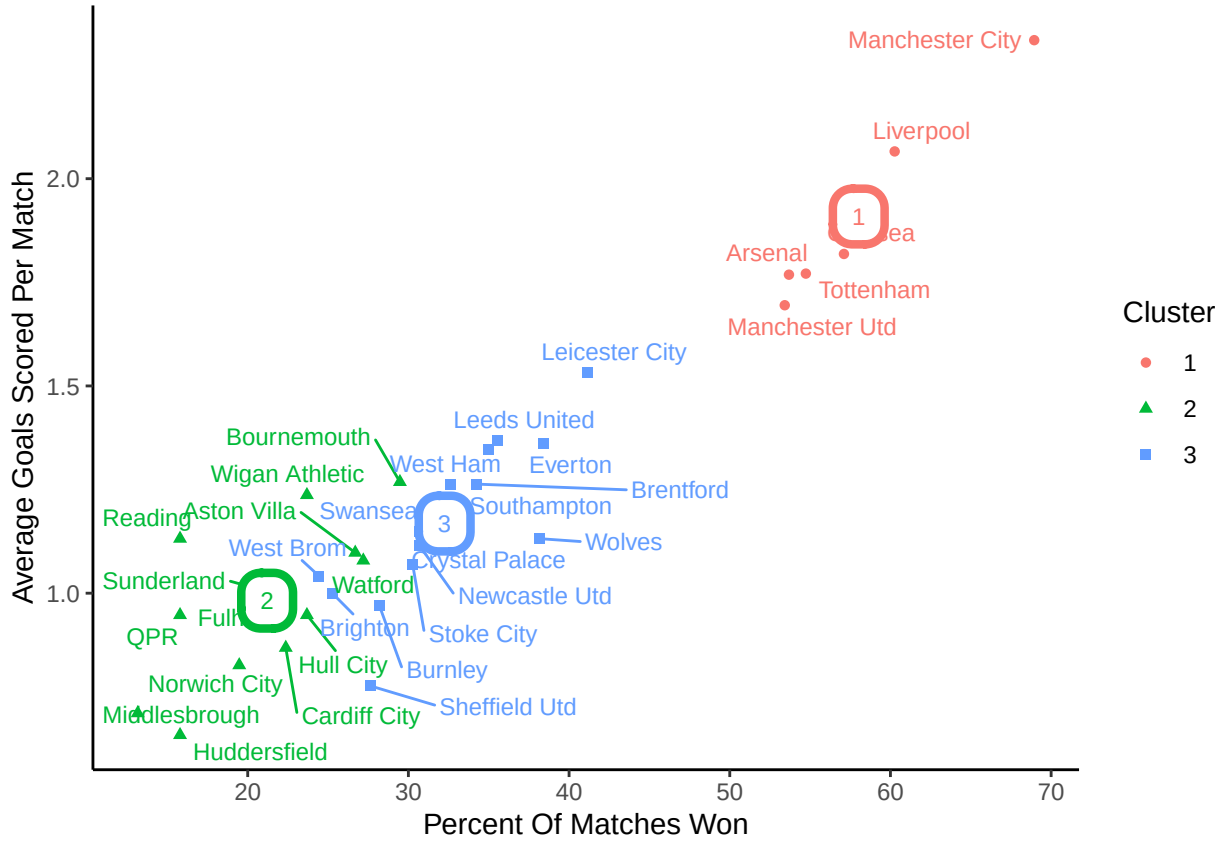
```r
sdGoalsPerMatch <- sd(groupedTable$GoalsPerMatch)
sdGoalsConcededPerMatch <- sd(groupedTable$GoalsConcededPerMatch)
sdWinPercent <- sd(groupedTable$WinPercent)
```

```r
#Thus, we should use 3 clusters from looking at the elbow plot.
# Perform k-means clustering with k = 3

# set the seed for reproducibility
set.seed(1877090)

footballKmeans <- tableForClustering %>%
  kmeans(centers = 3, nstart = 20)

footballKmeansSummaries <- tidy(footballKmeans) %>%
  mutate(GoalsPerMatch = meanGoalsPerMatch + (sdGoalsPerMatch * GoalsPerMatch_scaled),
         GoalsConcededPerMatch = meanGoalsConcededPerMatch + (sdGoalsConcededPerMatch * GoalsConcededPe
         WinPercent = meanWinPercent + (sdWinPercent * WinPercent_scaled)
         )

# Add cluster assignment as a factor to the data frame
# (argument order MUST be: kmeans object first, original data frame second)
footballWithKmeans <- augment(footballKmeans, groupedTable)

footballWithKmeans
```

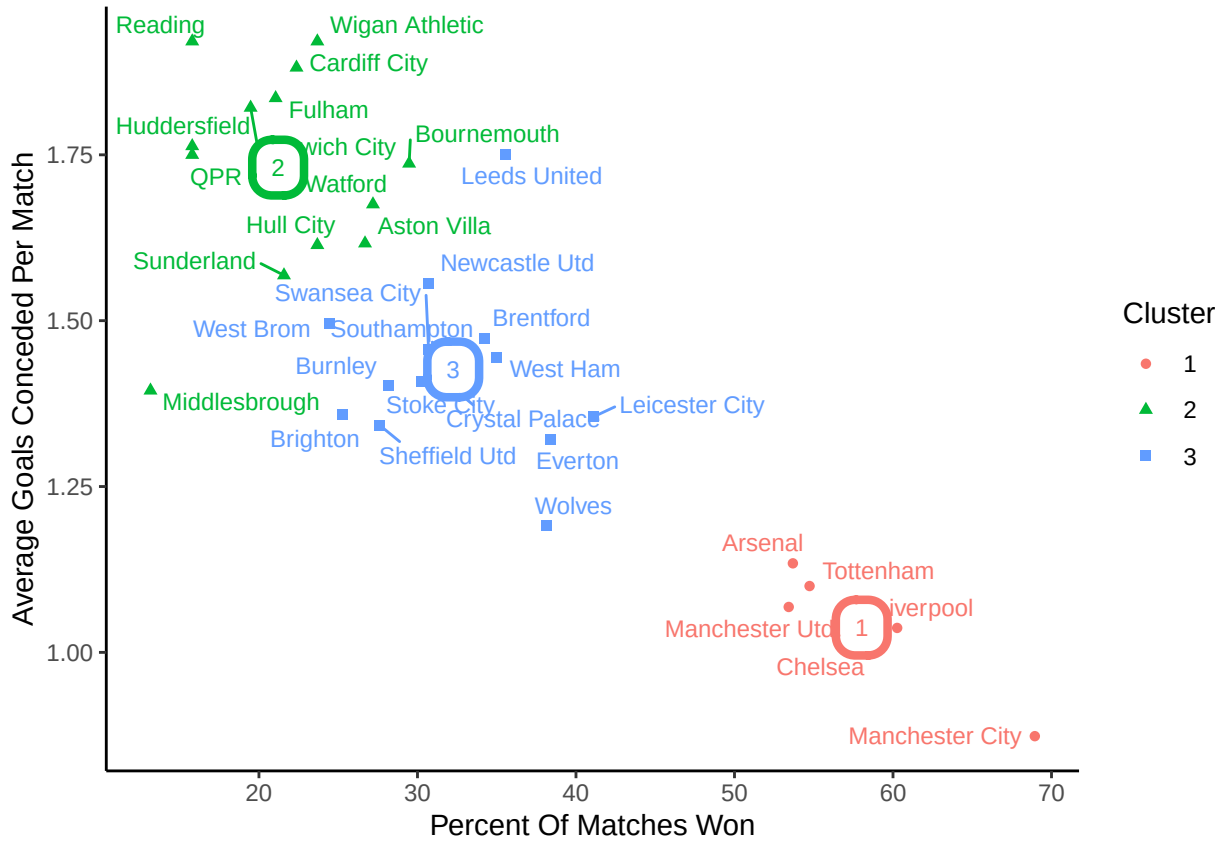
```
## # A tibble: 34 x 6
##    Squad          WinPercent GoalsPerMatch PointsPerMatch GoalsConcede~1 .clus~2
##    <chr>               <dbl>         <dbl>          <dbl>          <dbl> <fct>
##  1 Arsenal              53.7         1.77           1.82           1.13 1
##  2 Aston Villa          26.7         1.10           1.01           1.62 2
##  3 Bournemouth          29.5         1.27           1.11           1.74 2
##  4 Brentford            34.2         1.26           1.21           1.47 3
##  5 Brighton             25.3         1              1.1            1.36 3
##  6 Burnley              28.2         0.970          1.11           1.40 3
##  7 Cardiff City         22.4         0.868          0.842          1.88 2
##  8 Chelsea              57.1         1.82           1.94           1.01 1
##  9 Crystal Palace       31.6         1.13           1.18           1.41 3
## 10 Everton              38.4         1.36           1.42           1.32 3
## # ... with 24 more rows, and abbreviated variable names
## #   1: GoalsConcededPerMatch, 2: .cluster
```

```r
# Visualize the cluster assignments and centroids
ggplot(footballWithKmeans, aes(x = WinPercent, y = GoalsPerMatch)) +
  geom_point(aes(color = .cluster, shape = .cluster)) +
  geom_text_repel(aes(label = Squad, color = .cluster),
                  size = 3, max.overlaps = 15, show.legend = FALSE) +
  # Add centroid labels to plot
  geom_label(data = footballKmeansSummaries, aes(label = cluster, color = cluster),
             size = 3,
             label.r = unit(0.5, "lines"),
             label.size = 1.5,
             label.padding = unit(0.5, "lines"),
             show.legend = FALSE) +
```

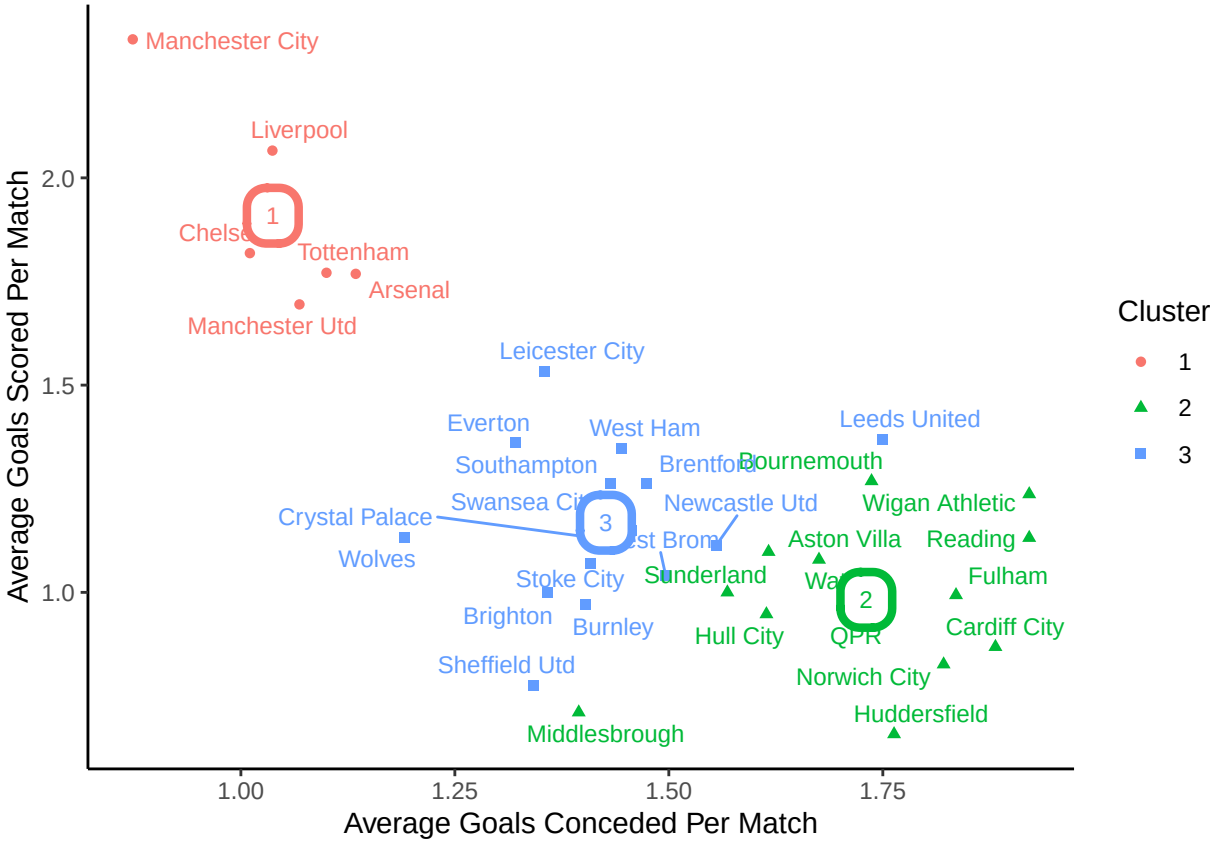
```
  labs(x = "Percent Of Matches Won",
       y = "Average Goals Scored Per Match",
       color = "Cluster",
       shape = "Cluster") +
  theme_classic()
```




```
# Visualize the cluster assignments and centroids
ggplot(footballWithKmeans, aes(x = WinPercent, y = GoalsConcededPerMatch)) +
  geom_point(aes(color = .cluster, shape = .cluster)) +
  geom_text_repel(aes(label = Squad, color = .cluster),
                  size = 3, max.overlaps = 15, show.legend = FALSE) +
  # Add centroid labels to plot
  geom_label(data = footballKmeansSummaries, aes(label = cluster, color = cluster),
             size = 3,
             label.r = unit(0.5, "lines"),
             label.size = 1.5,
             label.padding = unit(0.5, "lines"),
             show.legend = FALSE) +
  labs(x = "Percent Of Matches Won",
       y = "Average Goals Conceded Per Match",
       color = "Cluster",
       shape = "Cluster") +
  theme_classic()
```




```
# Visualize the cluster assignments and centroids
ggplot(footballWithKmeans, aes(x = GoalsConcededPerMatch, y = GoalsPerMatch)) +
  geom_point(aes(color = .cluster, shape = .cluster)) +
  geom_text_repel(aes(label = Squad, color = .cluster),
                  size = 3, max.overlaps = 15, show.legend = FALSE) +
  # Add centroid labels to plot
  geom_label(data = footballKmeansSummaries, aes(label = cluster, color = cluster),
             size = 3,
             label.r = unit(0.5, "lines"),
             label.size = 1.5,
             label.padding = unit(0.5, "lines"),
             show.legend = FALSE) +
  labs(x = "Average Goals Conceded Per Match",
       y = "Average Goals Scored Per Match",
       color = "Cluster",
       shape = "Cluster") +
  theme_classic()
```


Manchester City
Liverpool
Chelsea
Tottenham
Arsenal
Manchester Utd
Leicester City
Everton
West Ham
Southampton
Brentford
Swansea City
Crystal Palace
Wolves
Newcastle Utd
West Brom
Stoke City
Brighton
Burnley
Sunderland
Hull City
Leeds United
Bournemouth
Wigan Athletic
Aston Villa
Reading
Fulham
QPR
Cardiff City
Norwich City
Huddersfield
Sheffield Utd
Middlesbrough
1
2
3
Average Goals Scored Per Match
2.0
1.5
1.0
1.00
1.25
1.50
1.75
Average Goals Conceded Per Match
Cluster
1
2
3