Week 4.2 - Regression Analysis with Cricket Data

Regression Analyses with Cricket Data

In week 1, we took a brief look at the cricket match of statistics of the Indian Premier league in 2018 (IPL2018teams dataset). In this week, we will look at the player level statistics. In particular, we are interested in whether the player performance impact their salaries.

Import useful libraries

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------- tidyverse 1.3.0 --
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ---------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Import cricket data

In our data repository, there is a data set "IPL18Player.csv" which contains performance statistics as well as salary information of cricket players in the Indian Premier League in 2018.

```
IPLPlayer = read.csv("~/Google Drive/Sports Analytics Moocs/MOOC 1 - Foundations of sports analytics/We
head(IPLPlayer)
```

```
##   player_id long_scorecard_name  Salary                team matches wins
## 1      8931           AT Rayudu  343750 Chennai Super Kings      16   11
## 2    254771            D Shorey   31250 Chennai Super Kings       1    1
## 3     44613            DJ Bravo 1000000 Chennai Super Kings      16   11
## 4    214425           DJ Willey      NA Chennai Super Kings       3    2
## 5    258155           DL Chahar  125000 Chennai Super Kings      12    9
## 6     60234        F du Plessis  250000 Chennai Super Kings       6    5
##   team_runs_for team_runs_against matches_keeper byes_conceded moms innings
## 1          2809              2750              0             0    2      16
## 2           128               127              0             0    0       1
## 3          2809              2750              0             0    1      10
## 4           484               483              0             0    0       0
## 5          2117              2068              0             0    0       4
## 6          1050              1026              0             0    1       6
##   not_outs runs balls_faced fours sixes matches_bowled balls_bowled wickets
## 1        2  602         402    53    34              0           0       0
## 2        0    8           9     0     1              0           0       0
## 3        6  141          91     8    10             16         321      14
## 4        0    0           0     0     0              3          60       2
```

```
## 5        1   50          29     1     4             12          229      10
## 6        1  162         129    17     6              0            0       0
##   runs_conceded catches stumpings run_outs batting_dot_balls bowling_dot_balls
## 1             0       2         0        1               137                 0
## 2             0       0         0        0                 6                 0
## 3           533       9         0        0                29                90
## 4            95       2         0        1                 0                20
## 5           278       1         0        0                 6               118
## 6             0       1         0        0                56                 0
##   bowling_sixes no_balls balls_bowled_1_to_6 runs_conceded_1_to_6
## 1             0        0                   0                    0
## 2             0        0                   0                    0
## 3            29        0                   0                    0
## 4             3        0                  24                   38
## 5            10        2                 194                  236
## 6             0        0                   0                    0
##   balls_bowled_7_to_14 runs_conceded_7_to_14 balls_bowled_15_to_20
## 1                    0                    0                    0
## 2                    0                    0                    0
## 3                  126                  160                  195
## 4                    6                   10                   30
## 5                   37                   42                    0
## 6                    0                    0                    0
##   runs_conceded_15_to_20 event_winner
## 1                      0            1
## 2                      0            1
## 3                    373            1
## 4                     47            1
## 5                      0            1
## 6                      0            1
```

Data Exploration and Preparation

```
dim(IPLPlayer)
```

```
## [1] 149  35
```

Missing Values

```
sapply(IPLPlayer, function(x) sum(is.na(x)))
```

```
##              player_id    long_scorecard_name                 Salary
##                      0                      0                      8
##                   team                matches                   wins
##                      0                      0                      0
##          team_runs_for      team_runs_against         matches_keeper
##                      0                      0                      0
##          byes_conceded                   moms                innings
##                      0                      0                      0
##              not_outs                   runs             balls_faced
##                      0                      0                      0
##                 fours                  sixes         matches_bowled
##                      0                      0                      0
##           balls_bowled                wickets          runs_conceded
```

```
##                      0                      0                      0
##                catches              stumpings              run_outs
##                      0                      0                      0
##      batting_dot_balls      bowling_dot_balls          bowling_sixes
##                      0                      0                      0
##              no_balls    balls_bowled_1_to_6   runs_conceded_1_to_6
##                      0                      0                      0
##   balls_bowled_7_to_14  runs_conceded_7_to_14  balls_bowled_15_to_20
##                      0                      0                      0
## runs_conceded_15_to_20           event_winner
##                      0                      0
```

There are missing values in the salary variable. We will drop observations with missing values.

```
IPLPlayer = IPLPlayer %>% filter(!is.na(Salary))
dim(IPLPlayer)
```

```
## [1] 141  35
```

Create useful variables

Create dummy variables to indicate the role of the players.

- Create a variable to indicate whether a player had played as a batsman.

The variable "innings" indicates how many innings a player had batted in.

```
IPLPlayer$batsman = ifelse(IPLPlayer$innings > 0, 1, 0)
summary(IPLPlayer$batsman)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  1.0000  1.0000  0.9433  1.0000  1.0000
```

- Create a variable to indicate bowler.

```
IPLPlayer$bowler = ifelse(IPLPlayer$matches_bowled > 0, 1, 0)
summary(IPLPlayer$bowler)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  1.0000  0.6312  1.0000  1.0000
```

The last type of player that is not captured by either batsman or bowler is wicket keeper. In the dataset, the variable "matches_keeper" indicates the number of matches that a player is a wicket keeper.

Performance Measures

1. batting average = runs / the numbers of outs
2. batting strike rate = runs / 100 balls faced
3. bowling average = runs conceded / wicket taken
4. bowling strike rate = number of balls bowled / wicket taken

Notice that if a batsman has scored runs but not been dismissed, his batting average is technically infinite. Similarly, if a player did not face any ball, his batting strike would be infinite and if a player did not lose any wicket, his bowling average or bowling strike would be infinite.

We will not be able to run a regression when our variables have some infinite values.

There are two alternatives we will consider to deal with this issue. 1. Add 1 to the number of outs, balls faced, andn wickets taken in calculating the above variables. 2. Instead of creating the above measures, we can simply include total runs, total number of outs, and balls faced to measure a batsman's performance, and include runs conceded, number of balls bowled, and wickets taken to measure a bowler's performance.

```
IPLPlayer$outs = ifelse(IPLPlayer$batsman == 1,
                        IPLPlayer$innings - IPLPlayer$not_outs, 0)
summary(IPLPlayer$outs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       1       4       5       9      16
```

Create batting average, batting strke rate, bowling average, and bowling strike rate variables. Add 1 to the number of outs, balls faced, andn wickets taken in calculating these variables.

```
IPLPlayer$batting_average = IPLPlayer$runs/(IPLPlayer$outs + 1)
IPLPlayer$batting_strike = IPLPlayer$runs/(100*(IPLPlayer$balls_faced + 1))
IPLPlayer$bowling_average = IPLPlayer$runs_conceded/(IPLPlayer$wickets + 1)
IPLPlayer$bowling_strike = IPLPlayer$balls_bowled/(IPLPlayer$wickets +1)
```

```
summary(IPLPlayer$batting_average)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    4.00   12.50   15.09   23.00   65.00
```

```
summary(IPLPlayer$batting_strike)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.007391 0.011845 0.010416 0.013967 0.025000
```

```
summary(IPLPlayer$bowling_average)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00   20.05   17.49   27.47   72.00
```

```
summary(IPLPlayer$bowling_strike)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00   12.50   11.48   19.60   42.00
```

Regression Analyses

First let's run a regression of the salary on the type of player, batsman, bowler, and all-rounder.

```
reg_IPL1 = lm(data = IPLPlayer, Salary ~ batsman + bowler + batsman*bowler,
              na.action=na.omit)
summary(reg_IPL1)
```

```
##
## Call:
## lm(formula = Salary ~ batsman + bowler + batsman * bowler, data = IPLPlayer,
##     na.action = na.omit)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -715595 -432143 -153095  317857 1909405
##
## Coefficients: (1 not defined because of singularities)
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      277516     228691   1.213   0.2270
## batsman          469329     214187   2.191   0.0301 *
## bowler          -158452     102699  -1.543   0.1252
## batsman:bowler       NA         NA      NA       NA
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 577900 on 138 degrees of freedom
## Multiple R-squared:  0.05967,    Adjusted R-squared:  0.04604
## F-statistic: 4.379 on 2 and 138 DF,  p-value: 0.01433
```

Next we will first focus on performance of batsman.

We will first simply use the total number of runs, number of not outs, and number of balls faced to measure players' performance.

```
reg_IPL2 = lm(Salary ~ runs, data = IPLPlayer)
summary(reg_IPL2)
```

```
##
## Call:
## lm(formula = Salary ~ runs, data = IPLPlayer)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1196393  -346095  -153381   224843  1458940
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 387761.3    53874.1   7.198 3.49e-11 ***
## runs          1737.9      244.4   7.111 5.54e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 508500 on 139 degrees of freedom
## Multiple R-squared:  0.2667, Adjusted R-squared:  0.2615
## F-statistic: 50.57 on 1 and 139 DF,  p-value: 5.541e-11
```

```
reg_IPL3 = lm(Salary ~ runs+not_outs, data = IPLPlayer)
summary(reg_IPL3)
```

```
##
## Call:
## lm(formula = Salary ~ runs + not_outs, data = IPLPlayer)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1183927  -304309  -131782   232508  1450151
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 288032.5    60679.7   4.747 5.11e-06 ***
## runs          1491.2      248.7   5.995 1.68e-08 ***
## not_outs     89547.6    27853.8   3.215  0.00163 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 492300 on 138 degrees of freedom
## Multiple R-squared:  0.3178, Adjusted R-squared:  0.308
## F-statistic: 32.15 on 2 and 138 DF,  p-value: 3.453e-12
```

```
reg_IPL4 = lm(Salary ~ runs+not_outs+balls_faced, data = IPLPlayer)
summary(reg_IPL4)
```

```
##
## Call:
## lm(formula = Salary ~ runs + not_outs + balls_faced, data = IPLPlayer)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1156734  -307296  -109815   232490  1481786
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   301282      62890   4.791 4.26e-06 ***
## runs            2872       1712   1.678  0.09572 .
## not_outs       89450      27888   3.207  0.00167 **
## balls_faced    -2045       2508  -0.815  0.41638
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 492900 on 137 degrees of freedom
## Multiple R-squared:  0.3211, Adjusted R-squared:  0.3063
## F-statistic:  21.6 on 3 and 137 DF,  p-value: 1.621e-11
```

In the next regressions, we will use the modified batting average and batting strike variables to measure player performance.

```
reg_IPL5 = lm(Salary ~ batting_average, data = IPLPlayer)
summary(reg_IPL5)
```

```
##
## Call:
## lm(formula = Salary ~ batting_average, data = IPLPlayer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -898068 -349974 -158328  270974 1580324
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       307224      65151   4.716  5.8e-06 ***
## batting_average    20736       3195   6.491  1.4e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 520200 on 139 degrees of freedom
## Multiple R-squared:  0.2326, Adjusted R-squared:  0.2271
## F-statistic: 42.13 on 1 and 139 DF,  p-value: 1.401e-09
```

```
reg_IPL6 = lm(Salary ~ batting_average+batting_strike, data = IPLPlayer)
summary(reg_IPL6)
```

```
##
## Call:
## lm(formula = Salary ~ batting_average + batting_strike, data = IPLPlayer)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -918522 -362530 -157468  299926 1574188
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       266848      96902   2.754  0.00668 **
## batting_average    19027       4409   4.315 3.02e-05 ***
## batting_strike   6353042   11263804   0.564  0.57365
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 521500 on 138 degrees of freedom
## Multiple R-squared:  0.2343, Adjusted R-squared:  0.2233
## F-statistic: 21.12 on 2 and 138 DF,  p-value: 9.957e-09
```

We will now turn to bowlers' performance.

Again, we will first use number of runs conceded, number of balls bowled, and number of wickets taken to measure bowlers' performance.

```
reg_IPL7 = lm(Salary ~ runs_conceded, data = IPLPlayer)
summary(reg_IPL7)
```

```
##
## Call:
## lm(formula = Salary ~ runs_conceded, data = IPLPlayer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -722040 -405183 -228748  299966 2112466
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   543784.0    65344.4   8.322 7.25e-14 ***
## runs_conceded    569.3      318.3   1.789   0.0758 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 587100 on 139 degrees of freedom
## Multiple R-squared:  0.0225, Adjusted R-squared:  0.01547
## F-statistic:   3.2 on 1 and 139 DF,  p-value: 0.07583
```

```
reg_IPL8 = lm(Salary ~ runs_conceded+balls_bowled, data = IPLPlayer)
summary(reg_IPL8)
```

```
##
## Call:
## lm(formula = Salary ~ runs_conceded + balls_bowled, data = IPLPlayer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -763123 -384611 -217466  244931 2099759
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)     556491      65364   8.514 2.55e-14 ***
## runs_conceded    -2750       2005  -1.371   0.1725
## balls_bowled      4575       2729   1.676   0.0959 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 583300 on 138 degrees of freedom
## Multiple R-squared:  0.04201,    Adjusted R-squared:  0.02813
## F-statistic: 3.026 on 2 and 138 DF,  p-value: 0.05175
```

```
reg_IPL9 = lm(Salary ~ runs_conceded+balls_bowled+wickets, data = IPLPlayer)
summary(reg_IPL9)
```

```
##
## Call:
## lm(formula = Salary ~ runs_conceded + balls_bowled + wickets,
##     data = IPLPlayer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -760472 -398015 -205219  228780 2101985
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     554265      65419   8.472 3.35e-14 ***
## runs_conceded    -3049       2029  -1.503   0.1352
## balls_bowled      6343       3284   1.931   0.0555 .
## wickets         -27372      28275  -0.968   0.3347
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 583500 on 137 degrees of freedom
## Multiple R-squared:  0.04852,    Adjusted R-squared:  0.02768
## F-statistic: 2.329 on 3 and 137 DF,  p-value: 0.07723
```

In the next regression, we will use the modified bowling average and bowling strike variables to measure player performance.

```
reg_IPL10 = lm(Salary ~ bowling_average+bowling_strike, data = IPLPlayer)
summary(reg_IPL10)
```

```
##
## Call:
## lm(formula = Salary ~ bowling_average + bowling_strike, data = IPLPlayer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -770298 -421906 -160576  261420 2002737
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       653513      73278   8.918 2.55e-15 ***
## bowling_average   -34149      12463  -2.740  0.00695 **
## bowling_strike     49143      19499   2.520  0.01287 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 579800 on 138 degrees of freedom
## Multiple R-squared:  0.05365,    Adjusted R-squared:  0.03993
## F-statistic: 3.912 on 2 and 138 DF,  p-value: 0.02226
```

Lastly, we will incorporate performance measures of both batsman and bowler in the same regression.

We will first use the original variables, total number of runs, number of not outs, number of balls faced, number of runs conceded, number of balls bowled, and number of wickets in the regression.

```
reg_IPL11 = lm(Salary ~ runs+not_outs+balls_faced+runs_conceded+balls_bowled+wickets,
               data = IPLPlayer)
summary(reg_IPL11)
```

```
##
## Call:
## lm(formula = Salary ~ runs + not_outs + balls_faced + runs_conceded +
##     balls_bowled + wickets, data = IPLPlayer)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1209052  -242819   -71067   182395  1557292
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     145832      73165   1.993   0.0483 *
## runs              2090       1629   1.283   0.2017
## not_outs         59503      28199   2.110   0.0367 *
## balls_faced       -354       2412  -0.147   0.8836
## runs_conceded    -1738       1672  -1.039   0.3006
## balls_bowled      5030       2648   1.899   0.0597 .
## wickets         -22306      22565  -0.989   0.3247
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 465300 on 134 degrees of freedom
## Multiple R-squared:  0.4083, Adjusted R-squared:  0.3818
## F-statistic: 15.41 on 6 and 134 DF,  p-value: 2.199e-13
```

We will also use the modified batting average, batting strike, bowling average, and bowling strike variables to measure the player performance.

```
reg_IPL12 = lm(Salary ~ batting_average+batting_strike+bowling_average+bowling_strike,
               data = IPLPlayer)
summary(reg_IPL12)
```

```
##
## Call:
## lm(formula = Salary ~ batting_average + batting_strike + bowling_average +
##     bowling_strike, data = IPLPlayer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -935116 -303774  -79913  319134 1682837
##
```

```
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       136979     113800   1.204 0.230805
## batting_average    24184       4618   5.237 6.06e-07 ***
## batting_strike   -612223   10945429  -0.056 0.955476
## bowling_average   -31861      10798  -2.951 0.003735 **
## bowling_strike     59412      17016   3.492 0.000648 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 499300 on 136 degrees of freedom
## Multiple R-squared:  0.3083, Adjusted R-squared:  0.288
## F-statistic: 15.16 on 4 and 136 DF,  p-value: 2.853e-10
```

Self Test

- Run a regression of salary as a function of the interaction of batsman and runs and the interaction of bowler and wickets taken.
- Interpret your regression results.