FIFA20 Regression Problems

Naga Santhosh Kartheek Karnati

3/28/2020

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(stringr)
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
library(rio)
```

```
## Warning: package 'rio' was built under R version 3.6.2
```

```
library(modelr)
library(purrr)
```

```
##
## Attaching package: 'purrr'
```

```
## The following object is masked from 'package:data.table':
##
##     transpose
```

```
f20 <- fread('D:/NEU/Spring 2020/SML/Project/Datasets/players_20.csv')

f20 <- as_tibble(f20)

f20 <- f20 %>% select(-player_url, -long_name, -dob, -player_positions, -body_type, -real_face,
                      -real_face, -player_tags, -loaned_from, -joined, -contract_valid_until,
                      -nation_position, -nation_jersey_number, -player_traits)

f20 <- f20 %>% select(-(66:91))

#View(f20)


f20_sans_gks <- f20 %>% select(-gk_diving, -gk_handling, -gk_kicking, -gk_reflexes,
                               -gk_speed, -gk_positioning, -goalkeeping_diving,
                               -goalkeeping_handling, -goalkeeping_kicking,
                               -goalkeeping_positioning, -goalkeeping_reflexes)

f20_gks <- f20%>% select(age, height_cm, weight_kg, overall, potential, value_eur, wage_eur,
                         international_reputation, weak_foot, release_clause_eur, gk_diving, gk_handlin
                         gk_positioning) %>%
  filter(!is.na(gk_diving))

dim(f20_gks)
```

```
## [1] 2036   16
```

LINEAR MODEL: FULL SUBSET SELECTION

```
library(purrr)
#Players excluding GKs:
#Full subset selection:
full_subset_model <- lm(wage_eur~. , data = subset(f20_sans_gks, select = c(- short_name, -sofifa_id, -
     -team_position, -team_jersey_number, -work_rate)))


#Summary of the model for feature selection:
summary(full_subset_model)
```

```
##
## Call:
## lm(formula = wage_eur ~ ., data = subset(f20_sans_gks, select = c(-short_name,
##     -sofifa_id, -nationality, -club, -preferred_foot, -team_position,
##     -team_jersey_number, -work_rate)))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -146004   -1879    -259    1444  233741
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -2.742e+04  5.361e+03  -5.114 3.20e-07 ***
## age                         2.683e+02  4.682e+01   5.729 1.03e-08 ***
## height_cm                   5.398e+01  2.714e+01   1.989 0.046685 *
## weight_kg                   4.949e+01  2.340e+01   2.114 0.034503 *
## overall                    -2.067e+01  5.186e+01  -0.399 0.690133
## potential                  -2.464e+01  3.688e+01  -0.668 0.504103
## value_eur                   1.070e-03  1.441e-04   7.430 1.15e-13 ***
## international_reputation    1.099e+04  3.227e+02  34.039  < 2e-16 ***
## weak_foot                   2.334e+01  1.489e+02   0.157 0.875459
## skill_moves                -3.619e+02  2.191e+02  -1.651 0.098658 .
## release_clause_eur          9.435e-04  7.287e-05  12.948  < 2e-16 ***
## pace                        2.154e+02  3.045e+02   0.707 0.479370
## shooting                    1.501e+02  3.087e+02   0.486 0.626826
## passing                     5.321e+02  3.062e+02   1.738 0.082291 .
## dribbling                  -6.434e+01  3.083e+02  -0.209 0.834715
## defending                  -1.747e+01  3.089e+02  -0.057 0.954901
## physic                      5.189e+02  3.089e+02   1.680 0.092966 .
## attacking_crossing         -7.655e+01  6.242e+01  -1.226 0.220088
## attacking_finishing        -8.168e+01  1.397e+02  -0.585 0.558866
## attacking_heading_accuracy  2.647e+01  3.379e+01   0.783 0.433530
## attacking_short_passing    -2.078e+02  1.101e+02  -1.888 0.059087 .
## attacking_volleys          -2.061e+00  2.008e+01  -0.103 0.918242
## skill_dribbling             3.765e+01  1.554e+02   0.242 0.808512
## skill_curve                -1.398e+01  1.979e+01  -0.706 0.480070
## skill_fk_accuracy          -6.462e+01  1.885e+01  -3.428 0.000611 ***
## skill_long_passing         -1.008e+02  4.866e+01  -2.071 0.038414 *
## skill_ball_control          5.921e+01  9.608e+01   0.616 0.537727
## movement_acceleration      -1.146e+02  1.376e+02  -0.832 0.405166
## movement_sprint_speed      -1.148e+02  1.688e+02  -0.680 0.496344
## movement_agility            2.743e+01  3.417e+01   0.803 0.422011
## movement_reactions         -8.340e+00  2.672e+01  -0.312 0.754974
## movement_balance            2.852e+01  2.088e+01   1.366 0.171975
## power_shot_power           -1.399e+01  6.326e+01  -0.221 0.824964
## power_jumping              -1.621e+01  1.807e+01  -0.897 0.369784
## power_stamina              -1.505e+02  7.799e+01  -1.929 0.053718 .
## power_strength             -2.948e+02  1.550e+02  -1.902 0.057163 .
## power_long_shots           -3.684e+01  6.332e+01  -0.582 0.560715
## mentality_aggression       -1.024e+02  6.261e+01  -1.636 0.101915
## mentality_interceptions     1.101e+01  6.343e+01   0.174 0.862160
## mentality_positioning       2.730e+00  2.107e+01   0.130 0.896912
## mentality_vision           -9.625e+01  6.317e+01  -1.524 0.127598
## mentality_penalties         6.046e+00  1.996e+01   0.303 0.761932
## mentality_composure        -4.049e+01  1.681e+01  -2.408 0.016054 *
## defending_marking          -2.233e+01  9.350e+01  -0.239 0.811226
## defending_standing_tackle   1.081e+01  9.499e+01   0.114 0.909405
## defending_sliding_tackle    4.495e+01  3.691e+01   1.218 0.223362
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10880 on 15031 degrees of freedom
```

```
##   (3201 observations deleted due to missingness)
## Multiple R-squared:  0.7642, Adjusted R-squared:  0.7635
## F-statistic:  1082 on 45 and 15031 DF,  p-value: < 2.2e-16
```

```
#Features that influence wage the mmost:
#age, height, weight, value, international_reputation, release clause, skill fk accuracy,
#skill long passing and mentality composure.
```

FORWARD FEATURE SELECTION:

```
#Forward selection:
library(leaps)
```

```
## Warning: package 'leaps' was built under R version 3.6.2
```

```
ffs_fit <- regsubsets(wage_eur~., data= subset(f20_sans_gks, select = c(- short_name, -sofifa_id, -nati
    -team_position, -team_jersey_number, -work_rate)), method = "forward")

ffs_fit_summary <- summary(ffs_fit)
coef(ffs_fit, 8)
```

```
##              (Intercept)                      age                value_eur
##            -1.531218e+04             2.750047e+02             1.017169e-03
## international_reputation       release_clause_eur       attacking_crossing
##             1.130867e+04             9.590736e-04             3.040206e+01
##        skill_fk_accuracy      mentality_composure defending_sliding_tackle
##            -2.954003e+01            -4.094771e+01             1.444362e+01
```

```
names(ffs_fit)
```

```
##  [1] "np"        "nrbar"     "d"         "rbar"      "thetab"    "first"
##  [7] "last"      "vorder"    "tol"       "rss"       "bound"     "nvmax"
## [13] "ress"      "ir"        "nbest"     "lopt"      "il"        "ier"
## [19] "xnames"    "method"    "force.in"  "force.out" "sserr"     "intercept"
## [25] "lindep"    "nullrss"   "nn"        "call"
```

```
#The features influencing wage the most: age, value, international reputation,
#release clause, attacking crossing, fk accuracy, mentality composure and sliding tackle.
```

RRS, Adjr2, Cp and BIC measures:

```
par(mfrow=c(2,2))
plot(ffs_fit_summary$rss,xlab="Number of Variables",ylab="RSS",type="l")

plot(ffs_fit_summary$adjr2,xlab="Number of Variables",ylab="Adjusted RSq",type="l")
which.max(ffs_fit_summary$adjr2)
```

```
## [1] 8
```

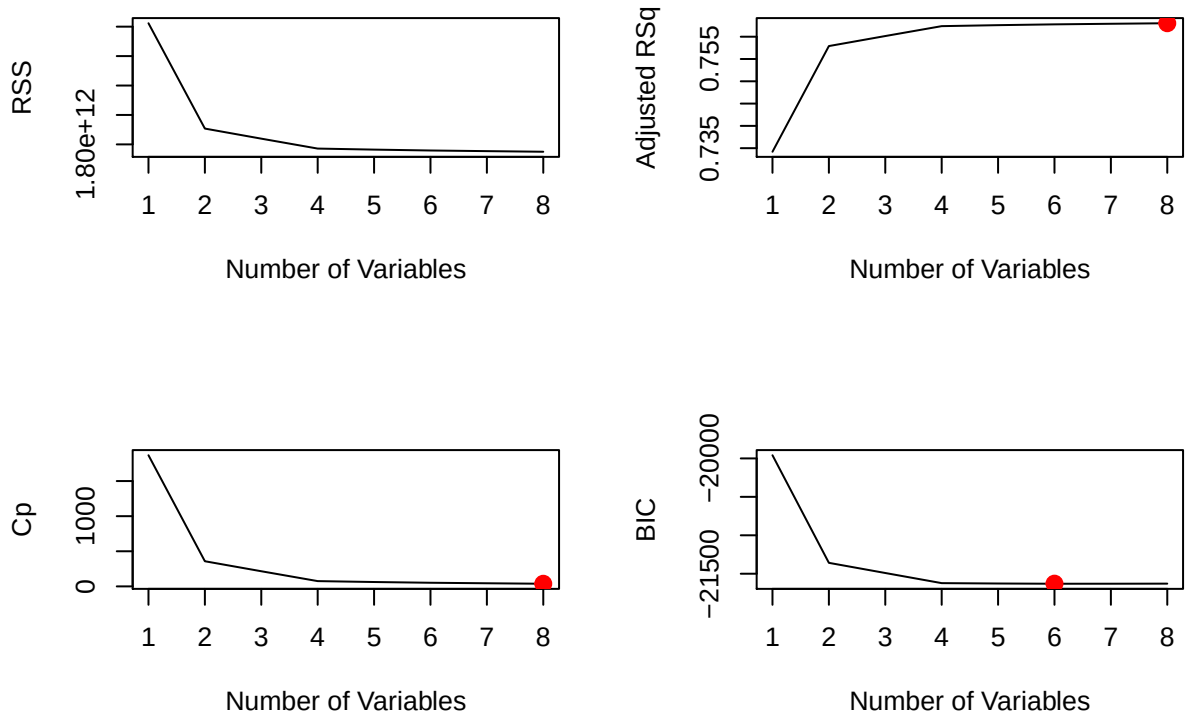
```
points(8,ffs_fit_summary$adjr2[8], col="red",cex=2,pch=20)

plot(ffs_fit_summary$cp,xlab="Number of Variables",ylab="Cp",type='l')
which.min(ffs_fit_summary$cp)
```

```
## [1] 8
```

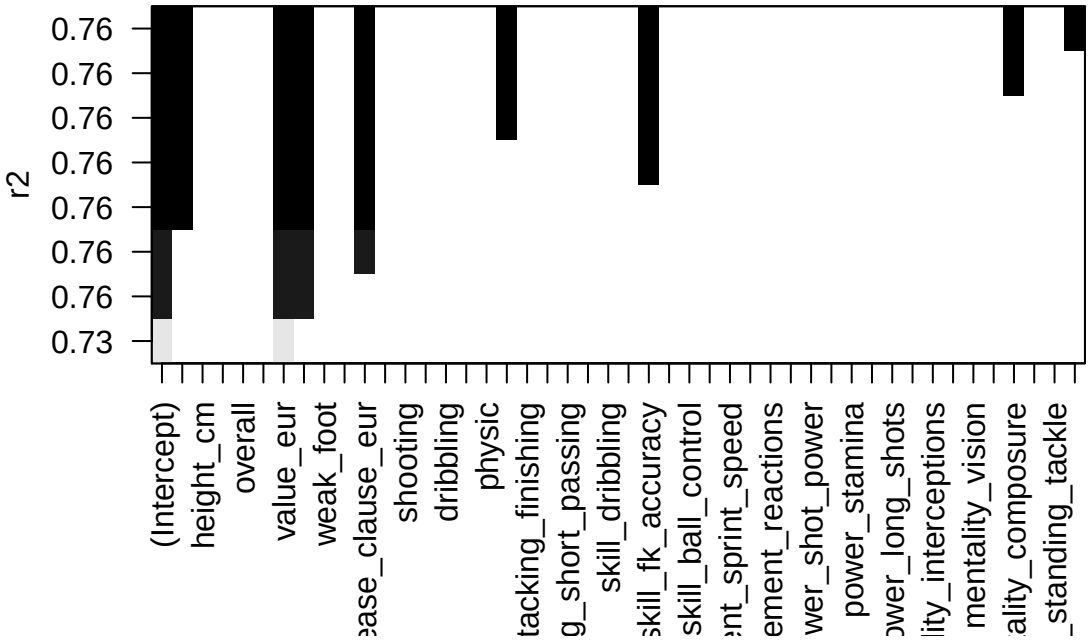
```
points(8,ffs_fit_summary$cp[8],col="red",cex=2,pch=20)

which.min(ffs_fit_summary$bic)
```

```
## [1] 6
```

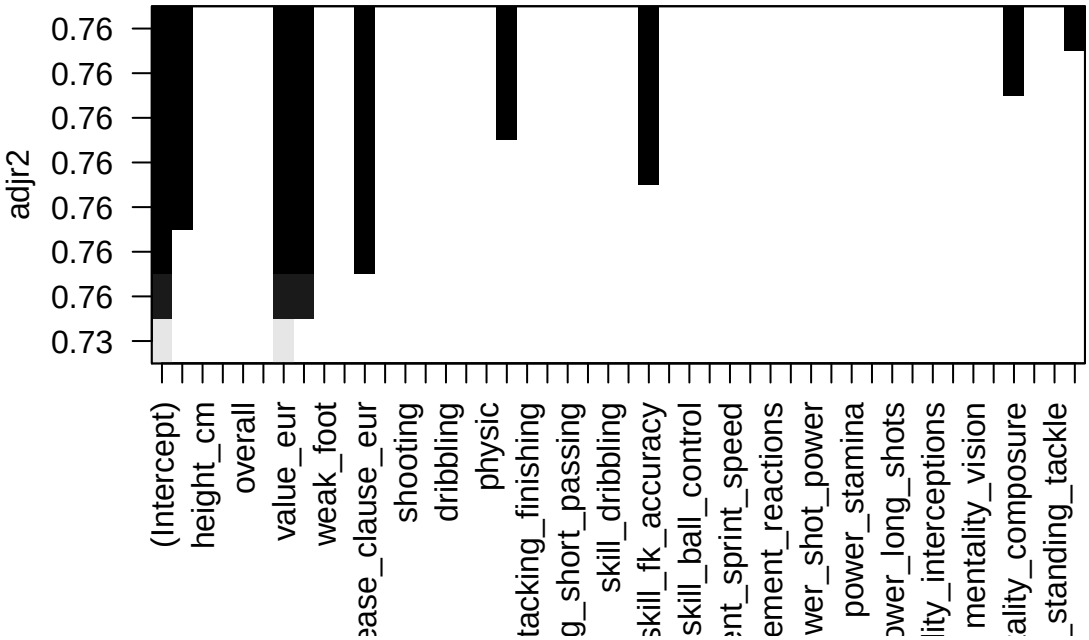
```
plot(ffs_fit_summary$bic,xlab="Number of Variables",ylab="BIC",type='l')
points(6,ffs_fit_summary$bic[8],col="red",cex=2,pch=20)
```



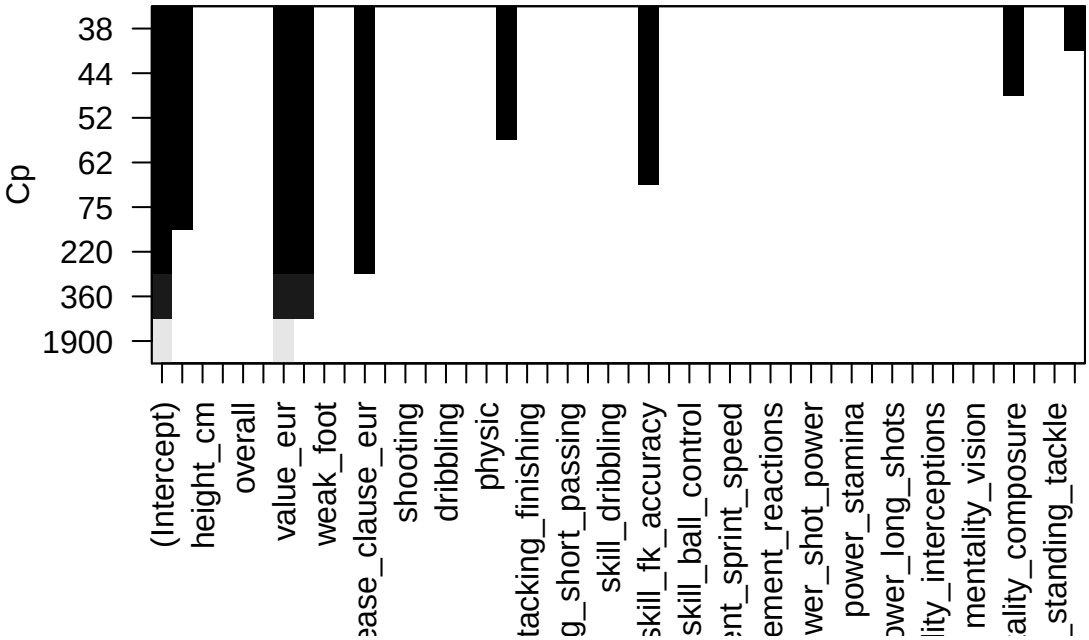

```
plot(ffs_fit,scale="r2")
```



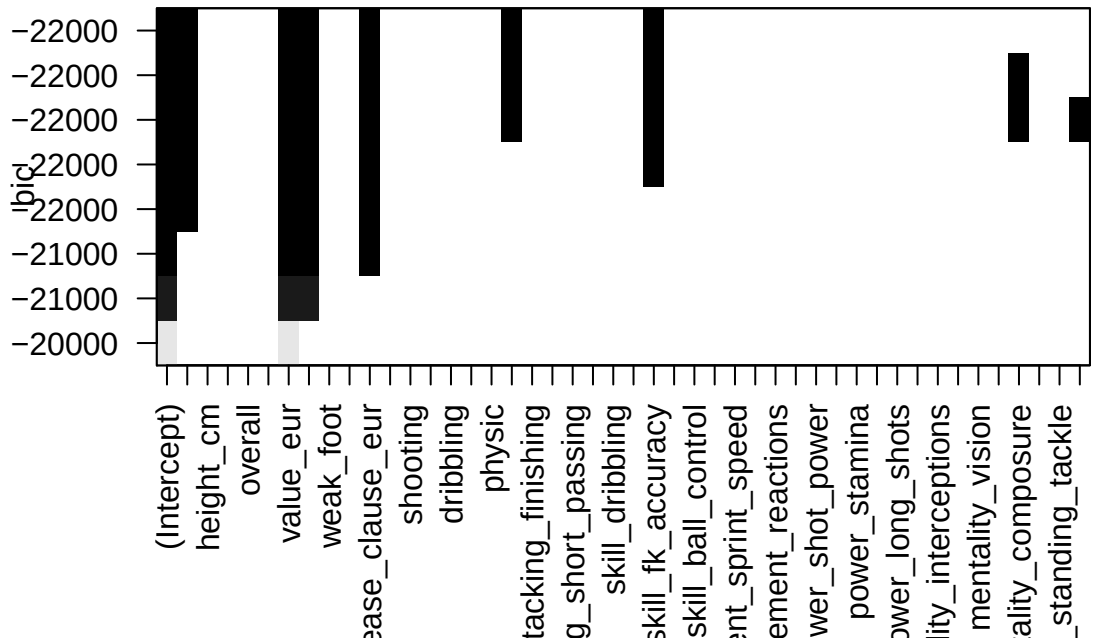

```
plot(ffs_fit,scale="adjr2")
```




```
plot(ffs_fit,scale="Cp")
```




```
plot(ffs_fit,scale="bic")
```



LINEAR MODEL FOR GKs:

```
#Full subset selection for goalkeepers:
full_subset_model_gk <- lm(wage_eur~. , data=f20_gks)
summary(full_subset_model_gk)
```

```
##
## Call:
## lm(formula = wage_eur ~ ., data = f20_gks)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -94672  -1453   -247    971  93820
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -3.783e+04  8.467e+03  -4.468 8.38e-06 ***
## age                       2.385e+02  7.442e+01   3.205 0.001374 **
## height_cm                 1.267e+02  4.860e+01   2.606 0.009235 **
## weight_kg                -2.958e+01  3.726e+01  -0.794 0.427347
## overall                  -4.649e+01  2.624e+02  -0.177 0.859380
## potential                 3.106e+01  7.232e+01   0.429 0.667635
## value_eur                 1.382e-03  3.850e-04   3.588 0.000341 ***
## international_reputation  1.123e+04  6.415e+02  17.501  < 2e-16 ***
```

```
## weak_foot                -9.520e+01  2.672e+02  -0.356 0.721697
## release_clause_eur        5.419e-04  1.909e-04   2.839 0.004577 **
## gk_diving                 1.948e+01  8.892e+01   0.219 0.826589
## gk_handling               6.669e+00  7.886e+01   0.085 0.932615
## gk_kicking                2.660e+01  4.076e+01   0.653 0.514086
## gk_reflexes               3.978e+01  8.655e+01   0.460 0.645840
## gk_speed                  1.196e+01  1.981e+01   0.604 0.546133
## gk_positioning           -7.618e+01  8.010e+01  -0.951 0.341693
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7801 on 1887 degrees of freedom
##   (133 observations deleted due to missingness)
## Multiple R-squared:  0.7969, Adjusted R-squared:  0.7953
## F-statistic: 493.6 on 15 and 1887 DF,  p-value: < 2.2e-16
```

```
#Features that influence goalkeeper's wage:
#age, height, value, international reputation and release clause.
```

Stepwise Forward selection for GKs:

```
ffs_fit_gks <- regsubsets(wage_eur ~ ., data = f20_gks, method = "forward")
ffs_fit_gks_summary <- summary(ffs_fit_gks)
coef(ffs_fit_gks, 8)
```

```
##              (Intercept)                      age                height_cm
##            -3.616198e+04             2.125745e+02             1.240526e+02
##                weight_kg                  overall                value_eur
##            -3.100028e+01             9.906753e+01             1.383660e-03
## international_reputation       release_clause_eur           gk_positioning
##             1.125060e+04             5.396116e-04            -9.921888e+01
```

```
#features selected in stepwise forward selection: age, height, weight, overall, value,
#international reputation, release clause, gk positioning.
```

RSS, Adjr2, Cp and BIC measures:

```
par(mfrow=c(2,2))
plot(ffs_fit_gks_summary$rss,xlab="Number of Variables",ylab="RSS",type="l")

plot(ffs_fit_gks_summary$adjr2,xlab="Number of Variables",ylab="Adjusted RSq",type="l")
which.max(ffs_fit_gks_summary$adjr2)
```

```
## [1] 5
```

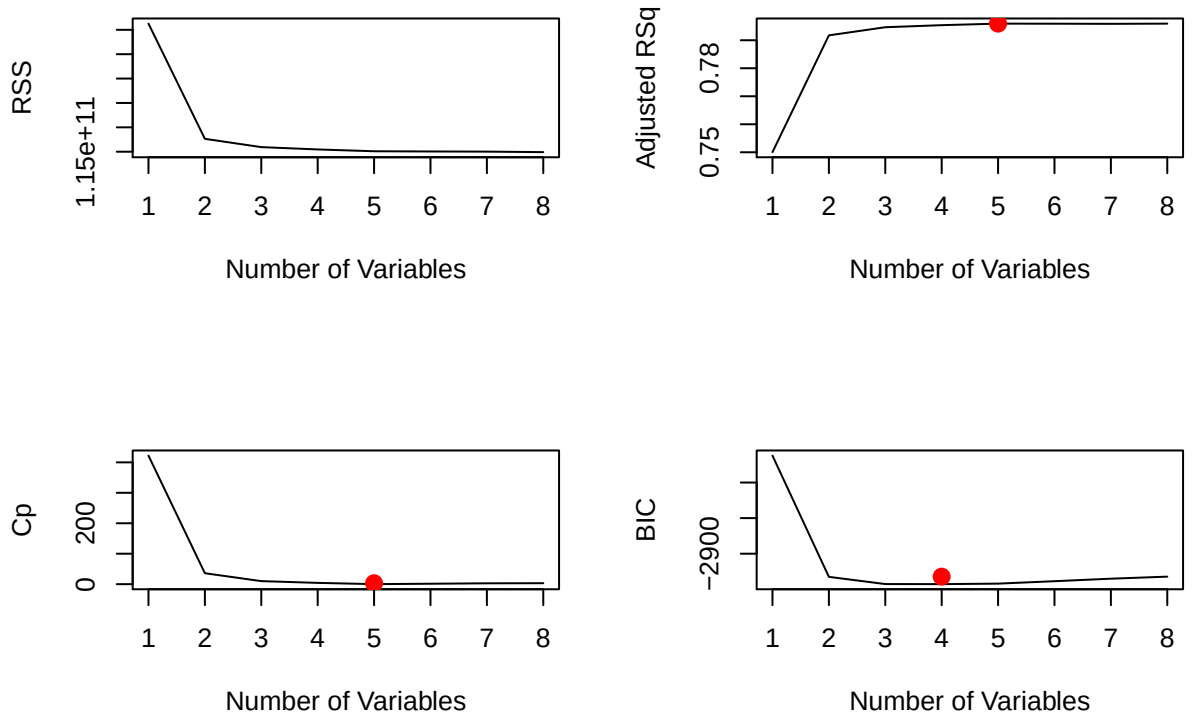
```
points(5,ffs_fit_gks_summary$adjr2[8], col="red",cex=2,pch=20)

plot(ffs_fit_gks_summary$cp,xlab="Number of Variables",ylab="Cp",type='l')
which.min(ffs_fit_gks_summary$cp)
```

```
## [1] 5
```

```
points(5,ffs_fit_gks_summary$cp[8],col="red",cex=2,pch=20)

which.min(ffs_fit_gks_summary$bic)
```

```
## [1] 4
```

```
plot(ffs_fit_gks_summary$bic,xlab="Number of Variables",ylab="BIC",type='l')
points(4,ffs_fit_gks_summary$bic[8],col="red",cex=2,pch=20)
```



Player overall: FULL SUBSET SELECTION

```
#Player overall
ovr_model <- lm(overall~. , data = subset(f20_sans_gks, select = c(- short_name, -sofifa_id, -nationali
    -team_position, -team_jersey_number, -work_rate)))


#Summary of the model for feature selection:
summary(ovr_model)
```

```
##
```

```
## Call:
## lm(formula = overall ~ ., data = subset(f20_sans_gks, select = c(-short_name,
##     -sofifa_id, -nationality, -club, -preferred_foot, -team_position,
##     -team_jersey_number, -work_rate)))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.4004 -1.0488  0.0897  1.1475  6.5464
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -9.654e+00  8.403e-01 -11.489  < 2e-16 ***
## age                         5.249e-01  6.003e-03  87.440  < 2e-16 ***
## height_cm                  -2.408e-03  4.269e-03  -0.564  0.57277
## weight_kg                   2.221e-02  3.678e-03   6.038 1.60e-09 ***
## potential                   4.359e-01  4.583e-03  95.117  < 2e-16 ***
## value_eur                   2.693e-07  2.260e-08  11.920  < 2e-16 ***
## wage_eur                   -5.115e-07  1.283e-06  -0.399  0.69013
## international_reputation   -5.828e-01  5.247e-02 -11.107  < 2e-16 ***
## weak_foot                  -4.847e-02  2.342e-02  -2.070  0.03846 *
## skill_moves                 6.198e-01  3.410e-02  18.177  < 2e-16 ***
## release_clause_eur         -6.941e-08  1.151e-08  -6.030 1.68e-09 ***
## pace                       -3.707e-02  4.789e-02  -0.774  0.43885
## shooting                   -9.474e-03  4.855e-02  -0.195  0.84529
## passing                     6.387e-02  4.817e-02   1.326  0.18489
## dribbling                   3.710e-02  4.850e-02   0.765  0.44428
## defending                   3.170e-02  4.858e-02   0.653  0.51409
## physic                      7.853e-02  4.858e-02   1.616  0.10604
## attacking_crossing         -2.320e-03  9.818e-03  -0.236  0.81321
## attacking_finishing         2.060e-02  2.198e-02   0.937  0.34865
## attacking_heading_accuracy  4.109e-02  5.305e-03   7.745 1.02e-14 ***
## attacking_short_passing     4.569e-02  1.732e-02   2.639  0.00833 **
## attacking_volleys          -5.089e-03  3.158e-03  -1.612  0.10708
## skill_dribbling            -1.655e-02  2.444e-02  -0.677  0.49820
## skill_curve                -6.744e-03  3.113e-03  -2.166  0.03029 *
## skill_fk_accuracy          -2.544e-03  2.967e-03  -0.858  0.39110
## skill_long_passing         -1.495e-02  7.654e-03  -1.953  0.05080 .
## skill_ball_control          8.123e-02  1.510e-02   5.380 7.57e-08 ***
## movement_acceleration       4.323e-02  2.164e-02   1.997  0.04581 *
## movement_sprint_speed       4.676e-02  2.654e-02   1.762  0.07814 .
## movement_agility           -4.594e-03  5.374e-03  -0.855  0.39269
## movement_reactions          1.333e-01  4.060e-03  32.832  < 2e-16 ***
## movement_balance           -1.015e-02  3.284e-03  -3.091  0.00200 **
## power_shot_power            1.767e-02  9.949e-03   1.776  0.07583 .
## power_jumping              -2.382e-03  2.843e-03  -0.838  0.40205
## power_stamina               8.426e-03  1.227e-02   0.687  0.49220
## power_strength             -1.101e-02  2.438e-02  -0.452  0.65147
## power_long_shots           -5.450e-03  9.960e-03  -0.547  0.58424
## mentality_aggression       -1.603e-02  9.849e-03  -1.628  0.10354
## mentality_interceptions    -1.202e-02  9.977e-03  -1.205  0.22829
## mentality_positioning      -3.210e-02  3.304e-03  -9.716  < 2e-16 ***
## mentality_vision           -4.422e-02  9.930e-03  -4.453 8.54e-06 ***
## mentality_penalties        -4.867e-03  3.139e-03  -1.551  0.12103
## mentality_composure         5.370e-02  2.609e-03  20.584  < 2e-16 ***
```

```
## defending_marking           7.630e-03  1.471e-02   0.519  0.60391
## defending_standing_tackle   1.191e-03  1.494e-02   0.080  0.93648
## defending_sliding_tackle   -1.385e-02  5.806e-03  -2.386  0.01705 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.712 on 15031 degrees of freedom
##   (3201 observations deleted due to missingness)
## Multiple R-squared:  0.9388, Adjusted R-squared:  0.9387
## F-statistic:  5127 on 45 and 15031 DF,  p-value: < 2.2e-16
```

```
#The significant variables to predict overall are: age, weight_kg, potential, value_eur,
#international_reputation, weak_foot, skill_moves, release_clause_eur, attacking _heading,
#attacking short passing, skill curve, skill ball control, movement reactions, movement
#balance, mentality positioning, mentality vision, mentality composure, defending sliding
#tackle.
#dataset containing only these variables:
overall_full_subset <- f20_sans_gks %>% select(overall, age, weight_kg, potential, value_eur, internati
      attacking_heading_accuracy, attacking_short_passing, skill_curve, skill_ball_control,
      movement_reactions, movement_balance, mentality_positioning, mentality_vision,
      mentality_composure, defending_sliding_tackle)
#perform linear regression on this dataset to predict player value.
#str(overall_full_subset)
```

Player overall: STEPWISE FORWARD SELECTION

```
ovr_ffs_model <- regsubsets(overall ~ ., data= subset(f20_sans_gks, select = c(- short_name, -sofifa_id
     -team_position, -team_jersey_number, -work_rate)), method = "forward")
ovr_ffs_model_summary <- summary(ovr_ffs_model)
ovr_ffs_model
```

```
## Subset selection object
## Call: regsubsets.formula(overall ~ ., data = subset(f20_sans_gks, select = c(-short_name,
##     -sofifa_id, -nationality, -club, -preferred_foot, -team_position,
##     -team_jersey_number, -work_rate)), method = "forward")
## 45 Variables  (and intercept)
##                            Forced in Forced out
## age                            FALSE      FALSE
## height_cm                      FALSE      FALSE
## weight_kg                      FALSE      FALSE
## potential                      FALSE      FALSE
## value_eur                      FALSE      FALSE
## wage_eur                       FALSE      FALSE
## international_reputation       FALSE      FALSE
## weak_foot                      FALSE      FALSE
## skill_moves                    FALSE      FALSE
## release_clause_eur             FALSE      FALSE
## pace                           FALSE      FALSE
## shooting                       FALSE      FALSE
## passing                        FALSE      FALSE
## dribbling                      FALSE      FALSE
```

```
## defending                     FALSE      FALSE
## physic                        FALSE      FALSE
## attacking_crossing            FALSE      FALSE
## attacking_finishing           FALSE      FALSE
## attacking_heading_accuracy    FALSE      FALSE
## attacking_short_passing       FALSE      FALSE
## attacking_volleys             FALSE      FALSE
## skill_dribbling               FALSE      FALSE
## skill_curve                   FALSE      FALSE
## skill_fk_accuracy             FALSE      FALSE
## skill_long_passing            FALSE      FALSE
## skill_ball_control            FALSE      FALSE
## movement_acceleration         FALSE      FALSE
## movement_sprint_speed         FALSE      FALSE
## movement_agility              FALSE      FALSE
## movement_reactions            FALSE      FALSE
## movement_balance              FALSE      FALSE
## power_shot_power              FALSE      FALSE
## power_jumping                 FALSE      FALSE
## power_stamina                 FALSE      FALSE
## power_strength                FALSE      FALSE
## power_long_shots              FALSE      FALSE
## mentality_aggression          FALSE      FALSE
## mentality_interceptions       FALSE      FALSE
## mentality_positioning         FALSE      FALSE
## mentality_vision              FALSE      FALSE
## mentality_penalties           FALSE      FALSE
## mentality_composure           FALSE      FALSE
## defending_marking             FALSE      FALSE
## defending_standing_tackle     FALSE      FALSE
## defending_sliding_tackle      FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: forward
```

```
coef(ovr_ffs_model, 8)
```

```
##         (Intercept)                 age           potential           value_eur
##       -8.923078e+00        5.460395e-01        4.982575e-01        1.193610e-07
##  skill_ball_control  movement_reactions       power_stamina      power_strength
##        1.075874e-01        1.481190e-01        4.578565e-02        5.293419e-02
## mentality_composure
##        4.926088e-02
```

```
#significant variables to predict overall using stepwise forward selection are age,
#potential, value_eur, skill ball control, movement reactions, power stamina, power
#stamina, mentality composure.
overall_step_forward <- f20_sans_gks %>% select(overall, age, potential, value_eur,
      skill_ball_control, movement_reactions, power_stamina, power_strength,
      mentality_composure)
#use this dataset to predict overall.
```

RSS, Adjr2, Cp and BIC measures:

```
par(mfrow=c(2,2))
plot(ovr_ffs_model_summary$rss,xlab="Number of Variables",ylab="RSS",type="l")

plot(ovr_ffs_model_summary$adjr2,xlab="Number of Variables",ylab="Adjusted RSq",type="l")
which.max(ovr_ffs_model_summary$adjr2)
```

```
## [1] 8
```

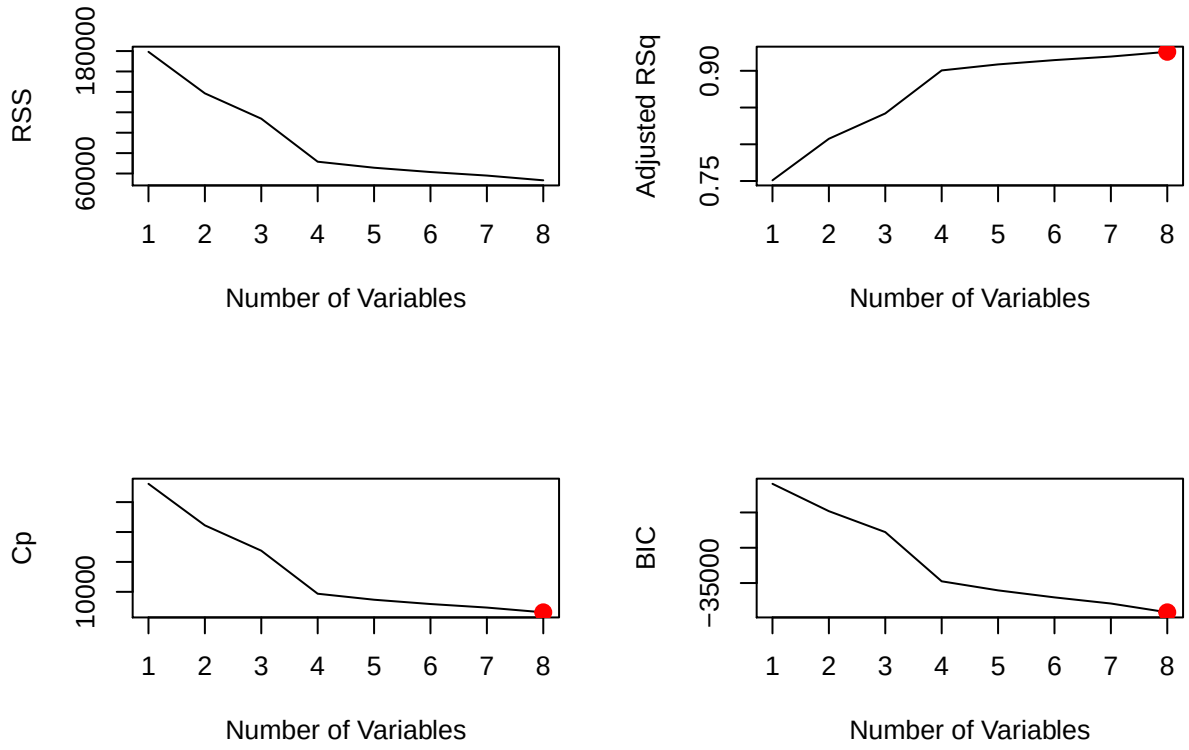
```
points(8,ovr_ffs_model_summary$adjr2[8], col="red",cex=2,pch=20)

plot(ovr_ffs_model_summary$cp,xlab="Number of Variables",ylab="Cp",type='l')
which.min(ovr_ffs_model_summary$cp)
```

```
## [1] 8
```

```
points(8,ovr_ffs_model_summary$cp[8],col="red",cex=2,pch=20)

which.min(ovr_ffs_model_summary$bic)
```

```
## [1] 8
```

```
plot(ovr_ffs_model_summary$bic,xlab="Number of Variables",ylab="BIC",type='l')
points(8,ovr_ffs_model_summary$bic[8],col="red",cex=2,pch=20)
```

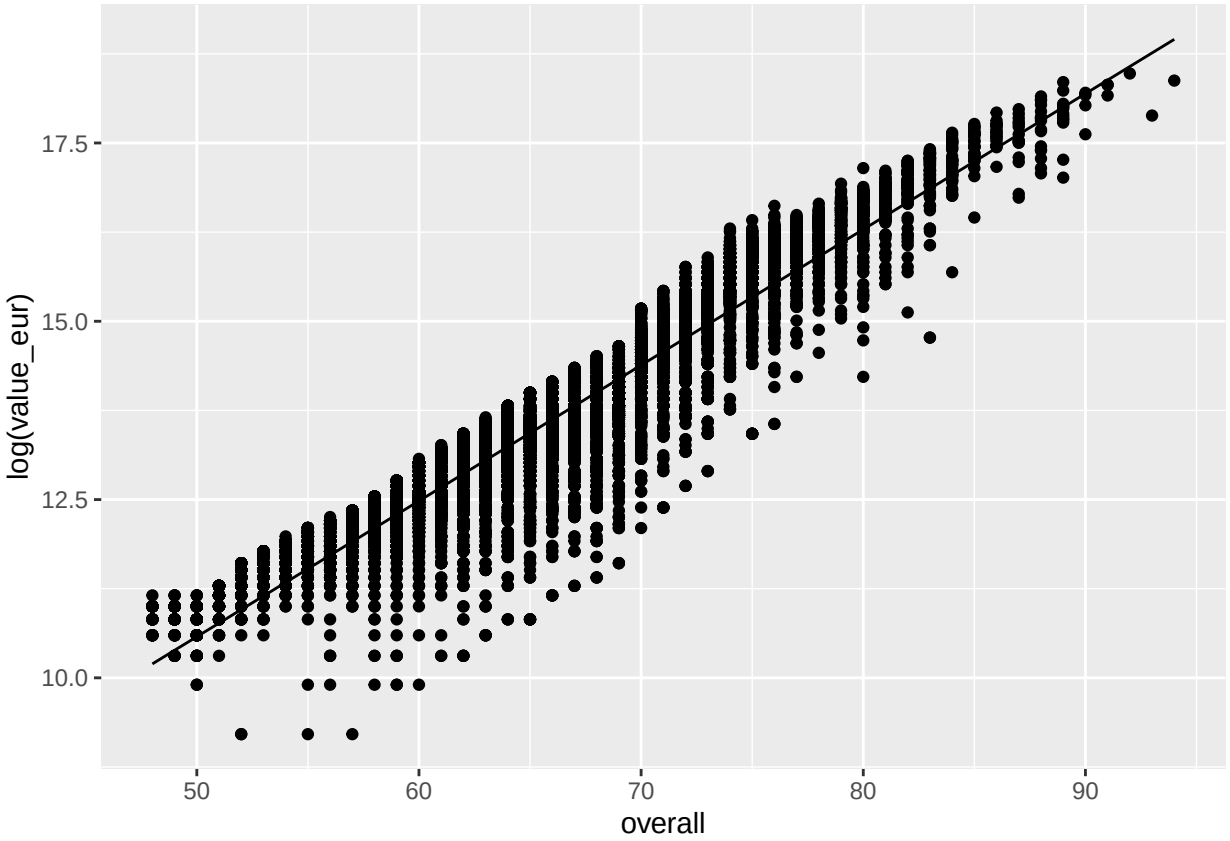


Value vs overall analysis:

```
f20_sans_gks_no0value <- f20_sans_gks%>%
  filter(value_eur != 0)

val_ovr_fit <- lm(log(value_eur) ~ overall, data= subset(f20_sans_gks_no0value, select = c(- short_name
     -team_position, -team_jersey_number, -work_rate)))

summary(val_ovr_fit)
```

```
##
## Call:
## lm(formula = log(value_eur) ~ overall, data = subset(f20_sans_gks_no0value,
##     select = c(-short_name, -sofifa_id, -nationality, -club,
##         -preferred_foot, -team_position, -team_jersey_number,
##         -work_rate)))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.69890 -0.21682  0.05642  0.30630  1.15478
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.0578231  0.0321356   32.92   <2e-16 ***
## overall     0.1903758  0.0004827  394.37   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4507 on 18026 degrees of freedom
## Multiple R-squared:  0.8961, Adjusted R-squared:  0.8961
## F-statistic: 1.555e+05 on 1 and 18026 DF,  p-value: < 2.2e-16
```

```
subset(f20_sans_gks_no0value, select = c(- short_name, -sofifa_id, -nationality, -club, -preferred_foot
     -team_position, -team_jersey_number, -work_rate))%>%
  add_predictions(val_ovr_fit)%>%
  ggplot(aes(overall))+
  geom_point(aes(y=log(value_eur))) +
geom_line(aes(y=pred))
```



#Predicting player overall using linear regression: full subset selection dataset

```
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
#using the full subset selection dataset and results to predict overall:
#View(overall_full_subset)
dim(overall_full_subset)
```

```
## [1] 18278    19
```

```
#split data into testing and training sets:
set.seed(1)
training.samples <- overall_full_subset$overall %>% createDataPartition(p = 0.8, list = FALSE)
train.data  <- overall_full_subset[training.samples, ]
test.data <- overall_full_subset[-training.samples, ]
dim(train.data)
```

```
## [1] 14625    19
```

```
dim(test.data)
```

```
## [1] 3653   19
```

```
#fit a linear model on the train set:
ovrfsm <- lm(overall~. ,data = na.omit(train.data))
#summary of the linear model:
summary(ovrfsm)
```

```
##
## Call:
## lm(formula = overall ~ ., data = na.omit(train.data))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.212  -1.164   0.270   1.400   7.741
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -1.024e+01  4.790e-01 -21.378  < 2e-16 ***
## age                          6.409e-01  6.453e-03  99.307  < 2e-16 ***
## weight_kg                    5.669e-02  3.693e-03  15.353  < 2e-16 ***
## potential                    5.635e-01  5.173e-03 108.935  < 2e-16 ***
## value_eur                    4.177e-07  2.882e-08  14.493  < 2e-16 ***
## international_reputation    -1.079e+00  6.299e-02 -17.125  < 2e-16 ***
## weak_foot                   -3.574e-02  2.934e-02  -1.218    0.223
## skill_moves                  4.378e-01  4.348e-02  10.069  < 2e-16 ***
## release_clause_eur          -1.283e-07  1.462e-08  -8.773  < 2e-16 ***
## attacking_heading_accuracy -8.972e-03  1.870e-03  -4.796 1.63e-06 ***
## attacking_short_passing     2.321e-02  3.677e-03   6.313 2.83e-10 ***
## skill_curve                  4.080e-04  2.066e-03   0.197    0.843
## skill_ball_control           3.959e-02  4.024e-03   9.840  < 2e-16 ***
## movement_reactions           2.072e-01  3.684e-03  56.245  < 2e-16 ***
## movement_balance            -8.914e-03  2.120e-03  -4.206 2.62e-05 ***
## mentality_positioning       -2.746e-02  2.402e-03 -11.431  < 2e-16 ***
## mentality_vision            -2.486e-03  2.464e-03  -1.009    0.313
## mentality_composure          3.272e-02  2.881e-03  11.357  < 2e-16 ***
## defending_sliding_tackle    -8.195e-03  1.274e-03  -6.432 1.30e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.069 on 13570 degrees of freedom
## Multiple R-squared:  0.9138, Adjusted R-squared:  0.9136
## F-statistic:  7988 on 18 and 13570 DF,  p-value: < 2.2e-16
```

```
#AIC and BIC of linear model:
AIC(ovrfsm)
```

```
## [1] 58346.9
```

```r
BIC(ovrfsm)
```

```
## [1] 58497.24
```

```r
#AIC=58346.9 BIC=58497.24
#predicted overall:
ovrpred <- predict(ovrfsm, test.data)

comparison_df <- data.frame(cbind(actual= test.data$overall, predicted = ovrpred))
#View(comparison_df)

#correlation accuracy:
cor(comparison_df, use = "complete.obs")    #95.6%
```

```
##              actual predicted
## actual    1.0000000 0.9561156
## predicted 0.9561156 1.0000000
```

```r
dim(comparison_df)
```

```
## [1] 3653    2
```

```r
dim(na.omit(comparison_df))
```

```
## [1] 3391    2
```

```r
comparison_df <- na.omit(comparison_df)

#RMSE and MAE:
# Function that returns Root Mean Squared Error
rmse <- function(error)
{
    sqrt(mean(error^2))
}

# Function that returns Mean Absolute Error
mae <- function(error)
{
    mean(abs(error))
}

#error in the model:
error <- comparison_df$predicted - comparison_df$actual

########VALIDATION SET APPROACH:
#RMSE:
rmse(error)   #2.032
```

```
## [1] 2.035841
```

```r
#MAE:
mae(error)    #1.590
```

```
## [1] 1.594226
```

```r
#######LOOCV:
# Define training control
train.control <- trainControl(method = "LOOCV")
# Train the model
ovrfs_loocv_model <- train(overall ~., data = na.omit(train.data), method = "lm",
               trControl = train.control)
# Summarize the results
print(ovrfs_loocv_model)
```

```
## Linear Regression
##
## 13589 samples
##    18 predictor
##
## No pre-processing
## Resampling: Leave-One-Out Cross-Validation
## Summary of sample sizes: 13588, 13588, 13588, 13588, 13588, 13588, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   2.071469  0.9134486  1.609075
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```r
summary(ovrfs_loocv_model)
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.212  -1.164   0.270   1.400   7.741
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -1.024e+01  4.790e-01 -21.378  < 2e-16 ***
## age                          6.409e-01  6.453e-03  99.307  < 2e-16 ***
## weight_kg                    5.669e-02  3.693e-03  15.353  < 2e-16 ***
## potential                    5.635e-01  5.173e-03 108.935  < 2e-16 ***
## value_eur                    4.177e-07  2.882e-08  14.493  < 2e-16 ***
## international_reputation    -1.079e+00  6.299e-02 -17.125  < 2e-16 ***
## weak_foot                   -3.574e-02  2.934e-02  -1.218    0.223
## skill_moves                  4.378e-01  4.348e-02  10.069  < 2e-16 ***
## release_clause_eur          -1.283e-07  1.462e-08  -8.773  < 2e-16 ***
## attacking_heading_accuracy -8.972e-03  1.870e-03  -4.796 1.63e-06 ***
## attacking_short_passing     2.321e-02  3.677e-03   6.313 2.83e-10 ***
```

```
## skill_curve                 4.080e-04  2.066e-03   0.197    0.843
## skill_ball_control          3.959e-02  4.024e-03   9.840  < 2e-16 ***
## movement_reactions          2.072e-01  3.684e-03  56.245  < 2e-16 ***
## movement_balance           -8.914e-03  2.120e-03  -4.206 2.62e-05 ***
## mentality_positioning      -2.746e-02  2.402e-03 -11.431  < 2e-16 ***
## mentality_vision           -2.486e-03  2.464e-03  -1.009    0.313
## mentality_composure         3.272e-02  2.881e-03  11.357  < 2e-16 ***
## defending_sliding_tackle   -8.195e-03  1.274e-03  -6.432 1.30e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.069 on 13570 degrees of freedom
## Multiple R-squared:  0.9138, Adjusted R-squared:  0.9136
## F-statistic:  7988 on 18 and 13570 DF,  p-value: < 2.2e-16
```

```
#predicted overall:
ovrpred_loocv <- predict(ovrfs_loocv_model, test.data)

comparison_df <- data.frame(cbind(actual= test.data$overall, predicted = ovrpred_loocv))
```

```
## Warning in cbind(actual = test.data$overall, predicted = ovrpred_loocv): number
## of rows of result is not a multiple of vector length (arg 2)
```

```
#View(comparison_df)

#correlation accuracy:
cor(comparison_df, use = "complete.obs")    #46.7%
```

```
##              actual predicted
## actual    1.0000000 0.4675116
## predicted 0.4675116 1.0000000
```

```
#error using loocv method:
error <- comparison_df$predicted - comparison_df$actual
#RMSE:
rmse(error)    #7.33
```

```
## [1] 7.332605
```

```
#MSE:
mae(error)     #3.56
```

```
## [1] 3.563983
```

```
########k Fold Cross Validation:
set.seed(2310)
train.control <- trainControl(method = "cv", number = 10)
# Train the model
ovrfs_kfcv_model <- train(overall ~., data = na.omit(train.data), method = "lm",
               trControl = train.control)
# Summarize the results
print(ovrfs_kfcv_model)
```

```
## Linear Regression
##
## 13589 samples
##    18 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 12231, 12230, 12230, 12231, 12230, 12230, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   2.071923  0.9136071  1.609739
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```
summary(ovrfs_kfcv_model)
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.212  -1.164   0.270   1.400   7.741
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -1.024e+01  4.790e-01 -21.378  < 2e-16 ***
## age                         6.409e-01  6.453e-03  99.307  < 2e-16 ***
## weight_kg                   5.669e-02  3.693e-03  15.353  < 2e-16 ***
## potential                   5.635e-01  5.173e-03 108.935  < 2e-16 ***
## value_eur                   4.177e-07  2.882e-08  14.493  < 2e-16 ***
## international_reputation   -1.079e+00  6.299e-02 -17.125  < 2e-16 ***
## weak_foot                  -3.574e-02  2.934e-02  -1.218    0.223
## skill_moves                 4.378e-01  4.348e-02  10.069  < 2e-16 ***
## release_clause_eur         -1.283e-07  1.462e-08  -8.773  < 2e-16 ***
## attacking_heading_accuracy -8.972e-03  1.870e-03  -4.796 1.63e-06 ***
## attacking_short_passing     2.321e-02  3.677e-03   6.313 2.83e-10 ***
## skill_curve                 4.080e-04  2.066e-03   0.197    0.843
## skill_ball_control          3.959e-02  4.024e-03   9.840  < 2e-16 ***
## movement_reactions          2.072e-01  3.684e-03  56.245  < 2e-16 ***
## movement_balance           -8.914e-03  2.120e-03  -4.206 2.62e-05 ***
## mentality_positioning      -2.746e-02  2.402e-03 -11.431  < 2e-16 ***
## mentality_vision           -2.486e-03  2.464e-03  -1.009    0.313
## mentality_composure         3.272e-02  2.881e-03  11.357  < 2e-16 ***
## defending_sliding_tackle   -8.195e-03  1.274e-03  -6.432 1.30e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.069 on 13570 degrees of freedom
## Multiple R-squared:  0.9138, Adjusted R-squared:  0.9136
## F-statistic:  7988 on 18 and 13570 DF,  p-value: < 2.2e-16
```

```r
#predicted overall:
ovrpred_kfcv <- predict(ovrfs_kfcv_model, test.data)

comparison_df <- data.frame(cbind(actual= test.data$overall, predicted = ovrpred_kfcv))
```

```
## Warning in cbind(actual = test.data$overall, predicted = ovrpred_kfcv): number
## of rows of result is not a multiple of vector length (arg 2)
```

```r
#View(comparison_df)

#correlation accuracy:
cor(comparison_df, use = "complete.obs")    #46.7%
```

```
##               actual predicted
## actual    1.0000000 0.4675116
## predicted 0.4675116 1.0000000
```

```r
#error using loocv method:
error <- comparison_df$predicted - comparison_df$actual
#RMSE:
rmse(error)    #7.33
```

```
## [1] 7.332605
```

```r
#MSE:
mae(error)     #3.56
```

```
## [1] 3.563983
```

#Predicting player overall using stepwise forward selection model dataset:

```r
#View(overall_step_forward)
dim(overall_step_forward)
```

```
## [1] 18278     9
```

```r
#split data into testing and training sets:
set.seed(987)
training.samples <- overall_step_forward$overall %>% createDataPartition(p = 0.8, list = FALSE)
train.data  <- overall_step_forward[training.samples, ]
test.data <- overall_step_forward[-training.samples, ]
dim(train.data)
```

```
## [1] 14625     9
```

```r
dim(test.data)
```

```
## [1] 3653    9
```

```r
#predicting overall using validation set approach, loocv and kfcv:
#####Validaiton set apparoach:
#fit a linear model on the train set:
ovrstfsm <- lm(overall~. ,data = na.omit(train.data))
#summary of the linear model:
summary(ovrstfsm)
```

```
##
## Call:
## lm(formula = overall ~ ., data = na.omit(train.data))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.0583  -1.2035   0.2403   1.4309   8.4404
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -9.772e+00  3.477e-01 -28.106  < 2e-16 ***
## age                  6.408e-01  6.180e-03 103.695  < 2e-16 ***
## potential            5.886e-01  4.932e-03 119.332  < 2e-16 ***
## value_eur            1.404e-07  4.135e-09  33.950  < 2e-16 ***
## skill_ball_control   5.058e-03  1.915e-03   2.641  0.00828 **
## movement_reactions   1.986e-01  3.574e-03  55.578  < 2e-16 ***
## power_stamina        1.882e-02  1.704e-03  11.045  < 2e-16 ***
## power_strength       2.808e-02  1.595e-03  17.608  < 2e-16 ***
## mentality_composure  3.034e-02  2.772e-03  10.943  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.106 on 14616 degrees of freedom
## Multiple R-squared:  0.9079, Adjusted R-squared:  0.9079
## F-statistic: 1.802e+04 on 8 and 14616 DF,  p-value: < 2.2e-16
```

```r
#AIC and BIC of linear model:
AIC(ovrstfsm)
```

```
## [1] 63303.5
```

```r
BIC(ovrstfsm)
```

```
## [1] 63379.4
```

```r
#AIC = 63303 and BIC = 63379.5

#predicted overall:
ovrpred <- predict(ovrstfsm, test.data)

comparison_df <- data.frame(cbind(actual= test.data$overall, predicted = ovrpred))
#View(comparison_df)

#correlation accuracy:
cor(comparison_df, use = "complete.obs")    #95.5%
```

```
##              actual predicted
## actual    1.0000000 0.9558094
## predicted 0.9558094 1.0000000
```

```
comparison_df <- na.omit(comparison_df)
#error in the model:
error <- comparison_df$predicted - comparison_df$actual
#RMSE:
rmse(error)   #2.055
```

```
## [1] 2.055502
```

```
#MAE:
mae(error)    #1.606
```

```
## [1] 1.606221
```

```
#######LOOCV:
# Define training control
train.control <- trainControl(method = "LOOCV")
# Train the model
ovrstfs_loocv_model <- train(overall ~., data = na.omit(train.data), method = "lm",
               trControl = train.control)
# Summarize the results
print(ovrstfs_loocv_model)
```

```
## Linear Regression
##
## 14625 samples
##     8 predictor
##
## No pre-processing
## Resampling: Leave-One-Out Cross-Validation
## Summary of sample sizes: 14624, 14624, 14624, 14624, 14624, 14624, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   2.107617  0.9077689  1.637907
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```
summary(ovrstfs_loocv_model)
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.0583  -1.2035   0.2403   1.4309   8.4404
##
```

```
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -9.772e+00  3.477e-01 -28.106  < 2e-16 ***
## age                  6.408e-01  6.180e-03 103.695  < 2e-16 ***
## potential            5.886e-01  4.932e-03 119.332  < 2e-16 ***
## value_eur            1.404e-07  4.135e-09  33.950  < 2e-16 ***
## skill_ball_control   5.058e-03  1.915e-03   2.641  0.00828 **
## movement_reactions   1.986e-01  3.574e-03  55.578  < 2e-16 ***
## power_stamina        1.882e-02  1.704e-03  11.045  < 2e-16 ***
## power_strength       2.808e-02  1.595e-03  17.608  < 2e-16 ***
## mentality_composure  3.034e-02  2.772e-03  10.943  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.106 on 14616 degrees of freedom
## Multiple R-squared:  0.9079, Adjusted R-squared:  0.9079
## F-statistic: 1.802e+04 on 8 and 14616 DF,  p-value: < 2.2e-16
```

```
#predicted overall:
ovrpred_loocv <- predict(ovrstfs_loocv_model, test.data)

comparison_df <- data.frame(cbind(actual= test.data$overall, predicted = ovrpred_loocv))
#View(comparison_df)

#correlation accuracy:
cor(comparison_df, use = "complete.obs")    #95.5%
```

```
##              actual predicted
## actual    1.0000000 0.9558094
## predicted 0.9558094 1.0000000
```

```
#error using loocv method:
error <- comparison_df$predicted - comparison_df$actual
#RMSE:
rmse(error)    #2.05
```

```
## [1] 2.055502
```

```
#MSE:
mae(error)     #1.60
```

```
## [1] 1.606221
```

```
########k Fold Cross Validation:
set.seed(99)
train.control <- trainControl(method = "cv", number = 10)
# Train the model
ovrfs_kfcv_model <- train(overall ~., data = na.omit(train.data), method = "lm",
               trControl = train.control)
# Summarize the results
print(ovrfs_kfcv_model)
```

```
## Linear Regression
##
## 14625 samples
##     8 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 13163, 13163, 13162, 13162, 13163, 13162, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   2.106729  0.9078012  1.637651
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```
summary(ovrfs_kfcv_model)
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.0583  -1.2035   0.2403   1.4309   8.4404
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -9.772e+00  3.477e-01 -28.106  < 2e-16 ***
## age                  6.408e-01  6.180e-03 103.695  < 2e-16 ***
## potential            5.886e-01  4.932e-03 119.332  < 2e-16 ***
## value_eur            1.404e-07  4.135e-09  33.950  < 2e-16 ***
## skill_ball_control   5.058e-03  1.915e-03   2.641  0.00828 **
## movement_reactions   1.986e-01  3.574e-03  55.578  < 2e-16 ***
## power_stamina        1.882e-02  1.704e-03  11.045  < 2e-16 ***
## power_strength       2.808e-02  1.595e-03  17.608  < 2e-16 ***
## mentality_composure  3.034e-02  2.772e-03  10.943  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.106 on 14616 degrees of freedom
## Multiple R-squared:  0.9079, Adjusted R-squared:  0.9079
## F-statistic: 1.802e+04 on 8 and 14616 DF,  p-value: < 2.2e-16
```

```
#predicted overall:
ovrpred_kfcv <- predict(ovrfs_kfcv_model, test.data)

comparison_df <- data.frame(cbind(actual= test.data$overall, predicted = ovrpred_kfcv))
#View(comparison_df)

#correlation accuracy:
cor(comparison_df, use = "complete.obs")    #95.5%
```

```
##              actual predicted
```

```
## actual    1.0000000 0.9558094
## predicted 0.9558094 1.0000000
```

```
#error using loocv method:
error <- comparison_df$predicted - comparison_df$actual
#RMSE:
rmse(error)    #2.05
```

```
## [1] 2.055502
```

```
#MSE:
mae(error)     #1.06
```

```
## [1] 1.606221
```

#Interpretation:

```
#Linear model using the forward stepwise selection dataset gave better results than
#the model using the full subset selection dataset since accuracy measures (cross validated) #RMSE and
```

#Lasso Regression for feature selection:

```
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 3.6.2
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loaded glmnet 3.0-2
```

```
data("swiss")

lasso_dataset <- f20_sans_gks %>% select(- short_name, -sofifa_id, -nationality, -club, -preferred_foot
     -team_position, -team_jersey_number, -work_rate)

#data types of columns:
#str(lasso_dataset)
#all integers

#remove overall from model matrix:
x_var <- model.matrix(overall~. , lasso_dataset)[,-4]
y_var <- lasso_dataset$overall
lambda_seq <- 10^seq(2, -2, by = -.1)
```

```r
# Splitting the data into test and train
set.seed(86)
train = sample(1:nrow(x_var), nrow(x_var)/5)
x_test = (-train)
y_test = y_var[-train]

cv_output <- cv.glmnet(x_var[train,], y_var[train],
            alpha = 1, lambda = lambda_seq)

# identifying best lamda
best_lam <- cv_output$lambda.min
best_lam
```

```
## [1] 0.01584893
```

```r
#best lambda value is 0.0158..
```

```r
#using minimum lambda value to build a lasso model again:
lasso_best <- glmnet(x_var[train,], y_var[train], alpha = 1, lambda = best_lam)
pred <- predict(lasso_best, s = best_lam, newx = x_var[-train,])

#comparing actual values and predicted values:
final <- data.frame(cbind(actual = y_var[-train], predicted = pred))
```

```
## Warning in cbind(actual = y_var[-train], predicted = pred): number of rows of
## result is not a multiple of vector length (arg 1)
```

```r
#coefficients of the best lasso model:
coef(lasso_best)
```

```
## 46 x 1 sparse Matrix of class "dgCMatrix"
##                                       s0
## (Intercept)                1.476899e+01
## (Intercept)                .
## age                        3.343268e-01
## height_cm                  .
## potential                  3.492351e-01
## value_eur                  1.923378e-07
## wage_eur                   .
## international_reputation   .
## weak_foot                  .
## skill_moves                5.329522e-01
## release_clause_eur         .
## pace                       1.200546e-02
## shooting                   .
## passing                    .
## dribbling                  .
## defending                  2.382984e-03
## physic                     3.074585e-02
## attacking_crossing         2.441529e-03
## attacking_finishing        .
```

```
## attacking_heading_accuracy  2.883591e-02
## attacking_short_passing     3.137415e-02
## attacking_volleys           .
## skill_dribbling             .
## skill_curve                 .
## skill_fk_accuracy           .
## skill_long_passing          .
## skill_ball_control          4.621178e-02
## movement_acceleration       3.938699e-03
## movement_sprint_speed       1.263983e-02
## movement_agility            .
## movement_reactions          1.092090e-01
## movement_balance           -1.259891e-02
## power_shot_power            6.285218e-03
## power_jumping               1.701330e-03
## power_stamina               2.254062e-03
## power_strength              .
## power_long_shots            .
## mentality_aggression        .
## mentality_interceptions     1.686410e-03
## mentality_positioning      -1.395848e-02
## mentality_vision           -8.168078e-03
## mentality_penalties         .
## mentality_composure         3.376486e-02
## defending_marking           3.345247e-03
## defending_standing_tackle   .
## defending_sliding_tackle    .
```

```
#coefs of variables with s0 as . have been shrunk to 0.

#error of the lasso model:
error = final$X1 - final$actual
#RMSE:
rmse(error)   #1.37
```

```
## [1] 1.37086
```

```
#MAE:
mae(error)    #1.07
```

```
## [1] 1.069118
```

```
#Of all the approaches, Lasso regression model gave the best accuracy measures on the
#test set.
length(train)
```

```
## [1] 3015
```