Tree assignment

Ahmad, Beka, and Rafael

12/11/2017

# Set global figure size  
knitr::opts\_chunk$set(fig.width=6, fig.height=4.0)

## Introduction

This report attempts to analyze what attributes are most predictive of player’s wages (Euros), by generating simple and multiple linear regression models on FIFA 18 data obtained from Kaggle.com. Multiple variables are available to evaluation to find the best model to achieve our goal stated above. Based on that, we chose to examine Age, Nationality, Club, Overall rating, Potential Overall, Special, Value (Euros), Reactions and Preferred Positions. These are the most significant variables in predicting player’s wages in FIFA18. There are around 1000 observations in the sample data set.

library(readr)  
fifa18 <- read\_csv("/Users/Beka/Downloads/fifa18.csv")

## Parsed with column specification:  
## cols(  
## ID = col\_integer(),  
## Name = col\_character(),  
## Age = col\_integer(),  
## Overall = col\_integer(),  
## Potential = col\_integer(),  
## Value = col\_integer(),  
## Wage = col\_integer(),  
## Special = col\_integer(),  
## Reactions = col\_integer(),  
## `Preferred Positions` = col\_character(),  
## Striker = col\_double(),  
## Winger = col\_integer(),  
## AttMid = col\_double(),  
## DefMid = col\_integer(),  
## CenterBack = col\_integer(),  
## Wingback = col\_double()  
## )

fifacom = fifa18 [complete.cases(fifa18),]  
attach (fifacom)

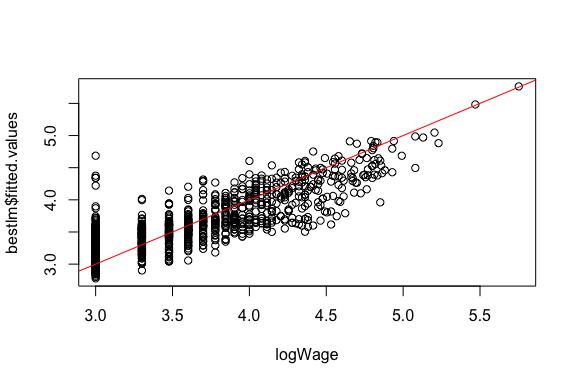
## Final Linear Regression Model

This was our final linear regression model:

logWage = log10(Wage)  
Age.c = Age - mean (Age, na.rm=T)  
Potential.c = Potential - mean (Potential)  
logValue.c = log10(Value) - mean (log10(Value))  
Age.Potential = Age.c \* Potential.c  
Potential.logValue = Potential.c \* logValue.c  
  
  
bestlm = lm (logWage ~ Age.c + Potential.c + logValue.c + Age.Potential + Potential.logValue)  
  
summary(bestlm)

##   
## Call:  
## lm(formula = logWage ~ Age.c + Potential.c + logValue.c + Age.Potential +   
## Potential.logValue)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.68646 -0.17963 -0.01017 0.18630 0.91477   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.6357505 0.0118795 306.052 < 2e-16 \*\*\*  
## Age.c 0.0237419 0.0028193 8.421 < 2e-16 \*\*\*  
## Potential.c 0.0155257 0.0037094 4.186 3.1e-05 \*\*\*  
## logValue.c 0.5516767 0.0358725 15.379 < 2e-16 \*\*\*  
## Age.Potential 0.0007755 0.0003924 1.976 0.0484 \*   
## Potential.logValue 0.0090034 0.0021967 4.099 4.5e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3076 on 969 degrees of freedom  
## Multiple R-squared: 0.6773, Adjusted R-squared: 0.6756   
## F-statistic: 406.7 on 5 and 969 DF, p-value: < 2.2e-16

plot (logWage, bestlm$fitted.values)  
abline(0, 1, col = 'red')



## Analysis Method - Recursive Partitioning

Start with modeling Wage vs all predictors.

#This code demonstrates recursive partition analysis  
library (rpart)  
logValue = log10(Value)  
attack = rowMeans(cbind(Striker, Winger))  
midfld = rowMeans(cbind(AttMid, DefMid))  
back = rowMeans(cbind(CenterBack, Wingback))  
tree1 = rpart (logWage~Age+Overall+Potential+logValue+Special+Reactions+attack+midfld+back, maxdepth=5)  
print (tree1$cptable)

## CP nsplit rel error xerror xstd  
## 1 0.51723832 0 1.0000000 1.0046400 0.04123358  
## 2 0.09903283 1 0.4827617 0.4850281 0.02642430  
## 3 0.03298382 2 0.3837289 0.4016798 0.02417591  
## 4 0.02618287 3 0.3507450 0.3759569 0.02401013  
## 5 0.01244981 4 0.3245622 0.3566507 0.02403632  
## 6 0.01000000 5 0.3121123 0.3447878 0.02433936

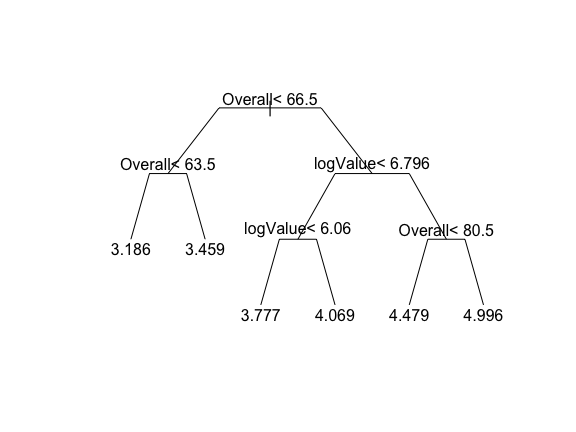
Find the tree with the smallest xerror:

opt1 = which.min (tree1$cptable [,"xerror"])  
opt1

## 6   
## 6

Plot the tree:

par (mfrow=c(1,1))  
plot(tree1, uniform = TRUE, margin = 0.1, branch = 0.5,   
 compress = TRUE)  
text(tree1)



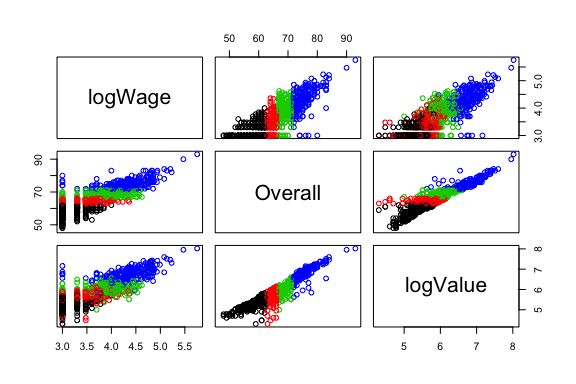
This tree says that Overall is the single most important predictor. Players have the highest Wage when Overall is more than 80.5 and logValue is more than 6.796. That groups of players has an average Wage of round (10^4.996, 1) (Per week)

The tree also says that if Overall is less than 63.5 predicted wage is round (10^3.186, 1) (Per week)

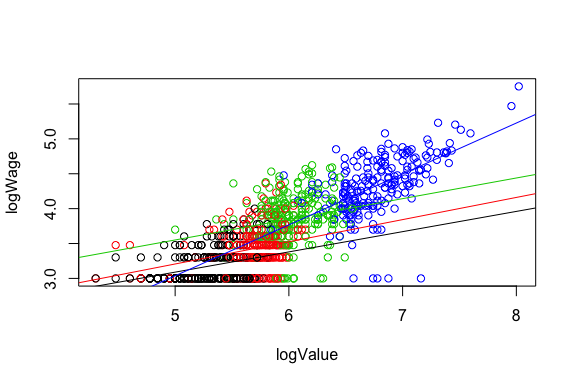
logValue matters if Overall is less than 66.5 and logValue is also less than 6.06

Interaction between Overall and logValue

Overall.levels = cut(Overall, breaks=quantile(Overall), 1:4)  
pairs (cbind.data.frame (logWage, Overall, logValue), col=Overall.levels)



plot (logValue, logWage, col=Overall.levels)  
abline (lm (logWage[Overall.levels==1] ~ logValue[Overall.levels==1]), col=1)  
abline (lm (logWage[Overall.levels==2] ~ logValue[Overall.levels==2]), col=2)  
abline (lm (logWage[Overall.levels==3] ~ logValue[Overall.levels==3]), col=3)  
abline (lm (logWage[Overall.levels==4] ~ logValue[Overall.levels==4]), col=4)



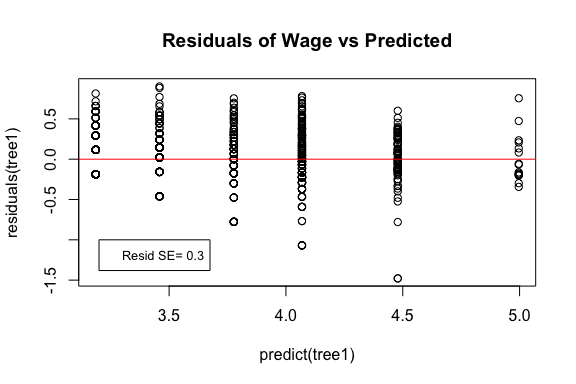
Plot logWage vs predicted.

plot (predict(tree1), logWage, main="Actual Wage vs Predicted")  
abline (0, 1, col='red')  
rsq1 = cor (predict(tree1), Wage)^2  
legend (3.2, 5.5, c(paste("Rsq=", round (rsq1, 2))), cex=0.8)



Residual plot

plot (predict(tree1), residuals(tree1), main="Residuals of Wage vs Predicted")  
abline (0, 0, col='red')  
resid.se = sd (residuals (tree1))  
legend (3.2, -1, c(paste ("Resid SE=", round (resid.se, 2))), cex=0.8)



The Large Positive Outlier is S.Bytyqi

max.res.players = which.max(residuals(tree1))  
fifacom [max.res.players, ]

## # A tibble: 1 x 16  
## ID Name Age Overall Potential Value Wage Special Reactions  
## <int> <chr> <int> <int> <int> <int> <int> <int> <int>  
## 1 10897 S. Bytyqi 22 64 71 625000 23000 1662 60  
## # ... with 7 more variables: `Preferred Positions` <chr>, Striker <dbl>,  
## # Winger <int>, AttMid <dbl>, DefMid <int>, CenterBack <int>,  
## # Wingback <dbl>

predict (tree1) [max.res.players]

## 722   
## 3.458937

10^predict (tree1) [max.res.players]

## 722   
## 2876.981

# Summary

The for the Log Wage tree is 0.3, which is lower than the for the final linear regression model, 0.67.

The residual SE for the tree is 0.3, which is similar (not same) than the residual SE for the final linear regression model, 0.3076.