# Regression

Francisco Trejo & Diego Ochoa

## What is Linear regression

Linear regression is simple why to graph out a relationship between our predictor and target as simple linear equation.This method is very simple and can be used in combination with other methods but assumes the relationship between our X and Y is a linear expression.

## Link to CSV file

https://www.kaggle.com/datasets/cashncarry/fifa-22-complete-player-dataset (https://www.kaggle.com/datasets/cashncarry/fifa-22-complete-player-dataset)

```
fifaData = read.csv("C:/Users/Diego/Downloads/players_fifa22.csv")
fifaData = fifaData[, c("Age", "Height", "Weight", "Overall", "Growth", "Potential", "WageEUR",
"ValueEUR")]
fifaData = fifaData[fifaData$WageEUR != 0, ]
```

## Dividing the data set into an 80/20 train/test set

```
set.seed(1301)
i <- sample(1:nrow(fifaData), nrow(fifaData)*0.8, replace=FALSE)
train <- fifaData[i,]
test <- fifaData[-i,]

lmName <- lm(formula = WageEUR ~ ValueEUR, data = train)
summary(lmName)
```

```
## 
## Call:
## lm(formula = WageEUR ~ ValueEUR, data = train)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -178311   -3334   -2249     625  210022 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 3.056e+03  9.604e+01   31.82   <2e-16 ***
## ValueEUR    2.075e-03  1.142e-05  181.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 11170 on 15345 degrees of freedom
## Multiple R-squared:  0.6826, Adjusted R-squared:  0.6826 
## F-statistic: 3.3e+04 on 1 and 15345 DF,  p-value: < 2.2e-16
```

# Data Analysis

Notice that the correlation between a player’s value and wage is high, meaning players are getting paid what they are worth, and so is a player’s height and weight, meaning most players have a similar body type. However a player’s age has low correlation to their value and wage.

```
sprintf("corralation between a player's value vs thier wage:    %#.4f", cor(train$ValueEUR, trai
n$WageEUR))
```

```
## [1] "corralation between a player's value vs thier wage:    0.8262"
```

```
sprintf("corralation between a player's value vs thier age:     %#.4f", cor(train$ValueEUR, trai
n$Age))
```

```
## [1] "corralation between a player's value vs thier age:     0.0352"
```

```
sprintf("corralation between a player's wage vs thier age:      %#.4f", cor(train$ValueEUR, trai
n$Age))
```

```
## [1] "corralation between a player's wage vs thier age:      0.0352"
```

```
sprintf("corralation between a player's Height vs thier Weight: %#.4f", cor(train$Height, train
$Weight))
```

```
## [1] "corralation between a player's Height vs thier Weight: 0.7640"
```

```
sprintf("The avgerage player's height is %#.2fcm and the standard diviation is %#.2f", mean(train$Height), sd(train$Height))
```

```
## [1] "The avgerage player's height is 181.31cm and the standard diviation is 6.85"
```

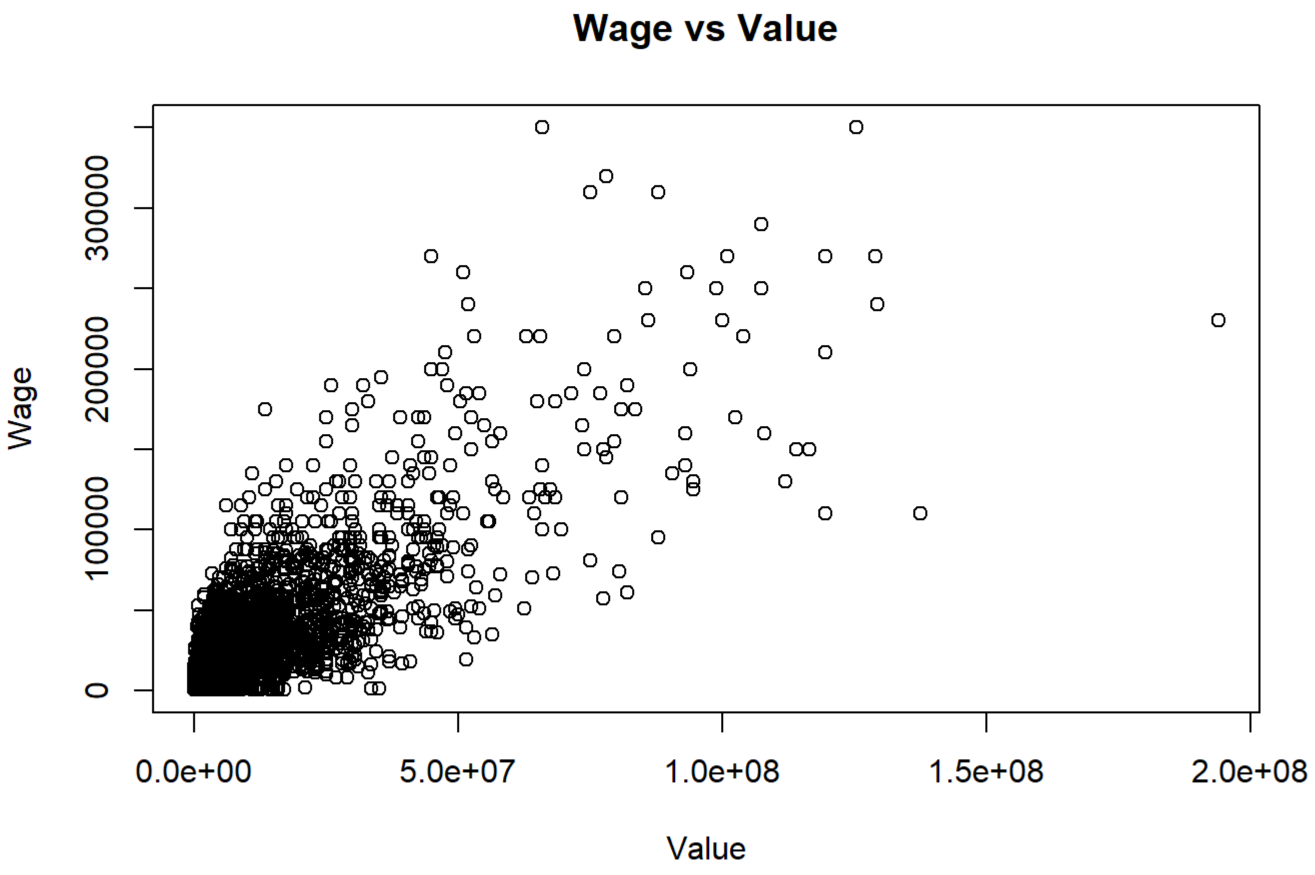
```
sprintf("The avgerage player's height is  %#.2fkg and the standard diviation is %#.2f", mean(train$Weight), sd(train$Weight))
```

```
## [1] "The avgerage player's height is  74.95kg and the standard diviation is 7.05"
```

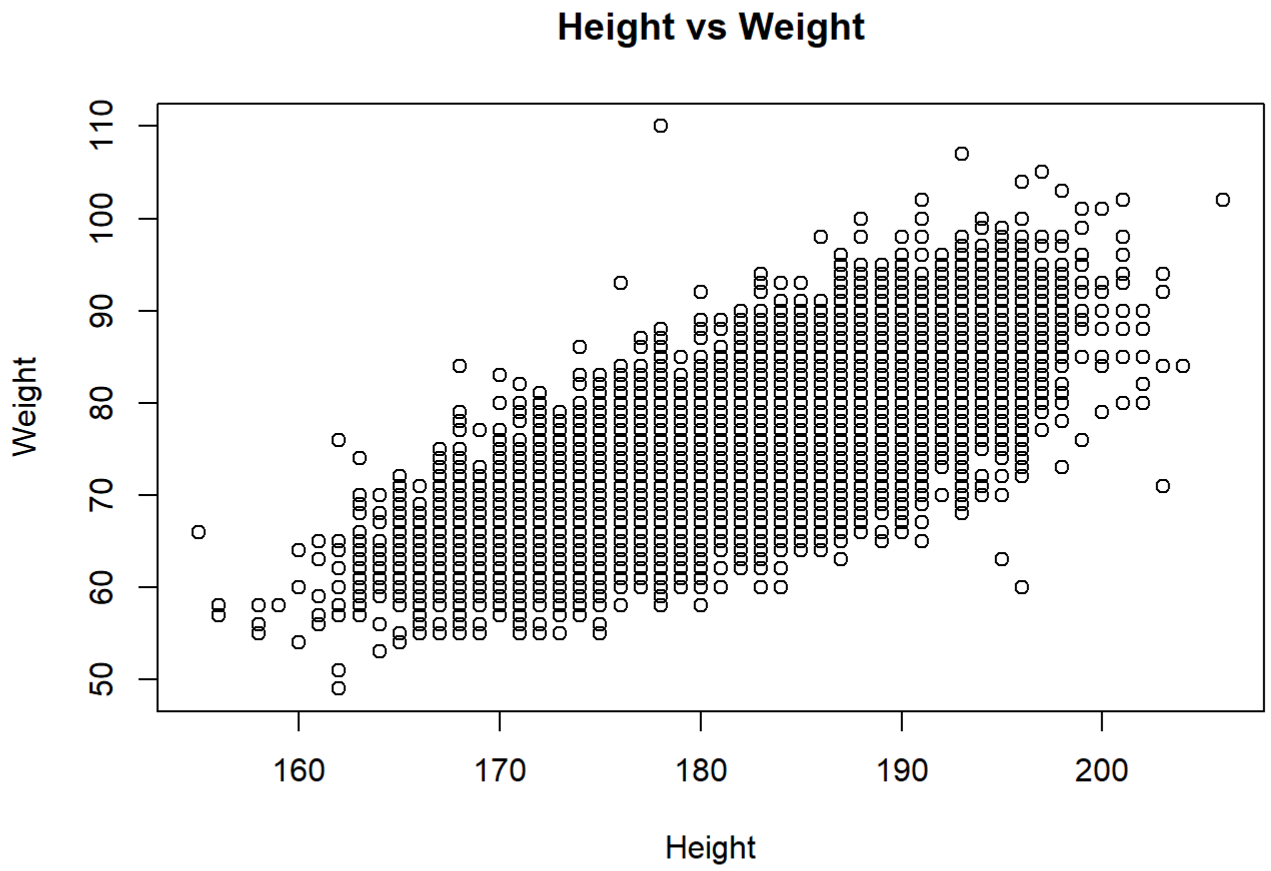
## Graphs

In these graphs notice that there is a bell curve relation with height, weight, and age vs value.

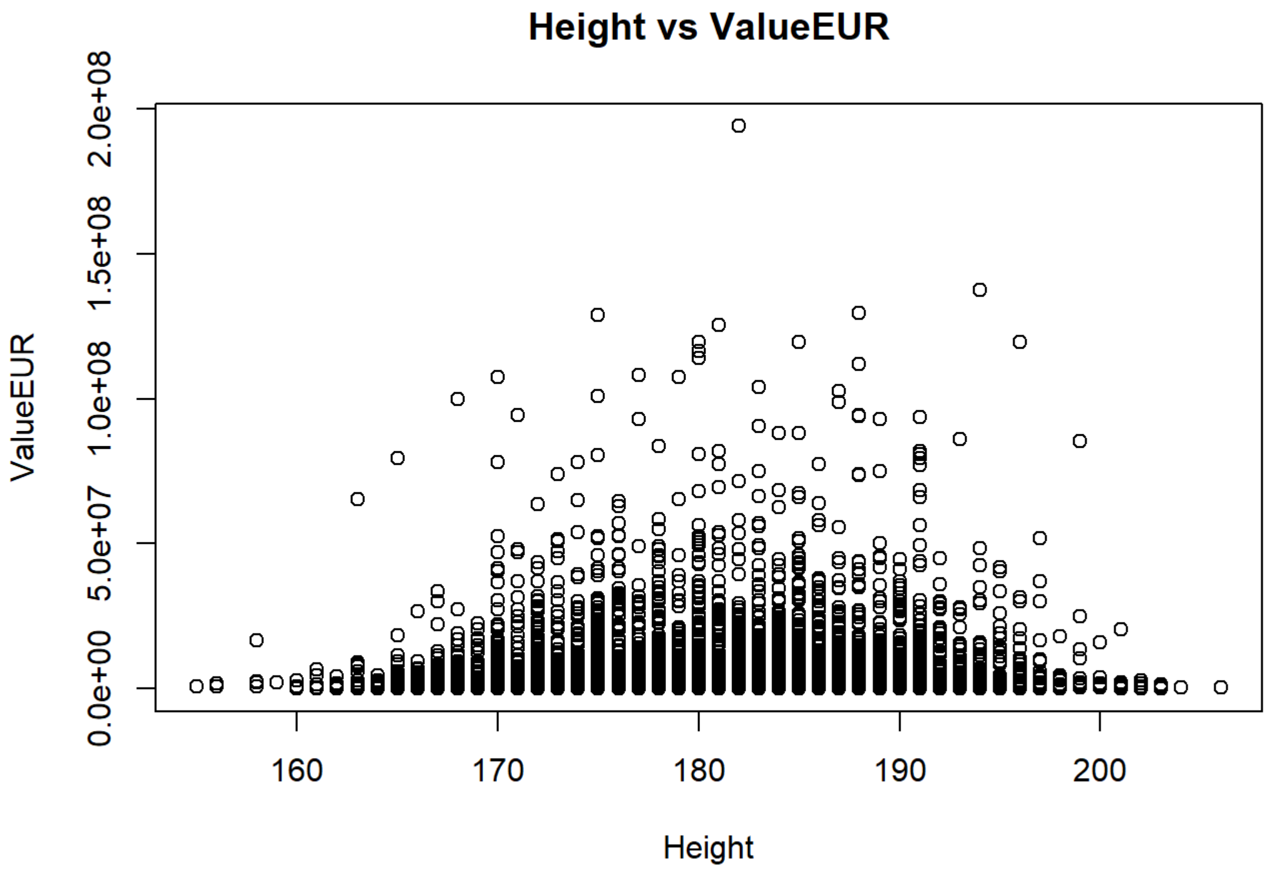
```
plot(fifaData$ValueEUR,fifaData$WageEUR, main = "Wage vs Value", xlab="Value", ylab="Wage")
```



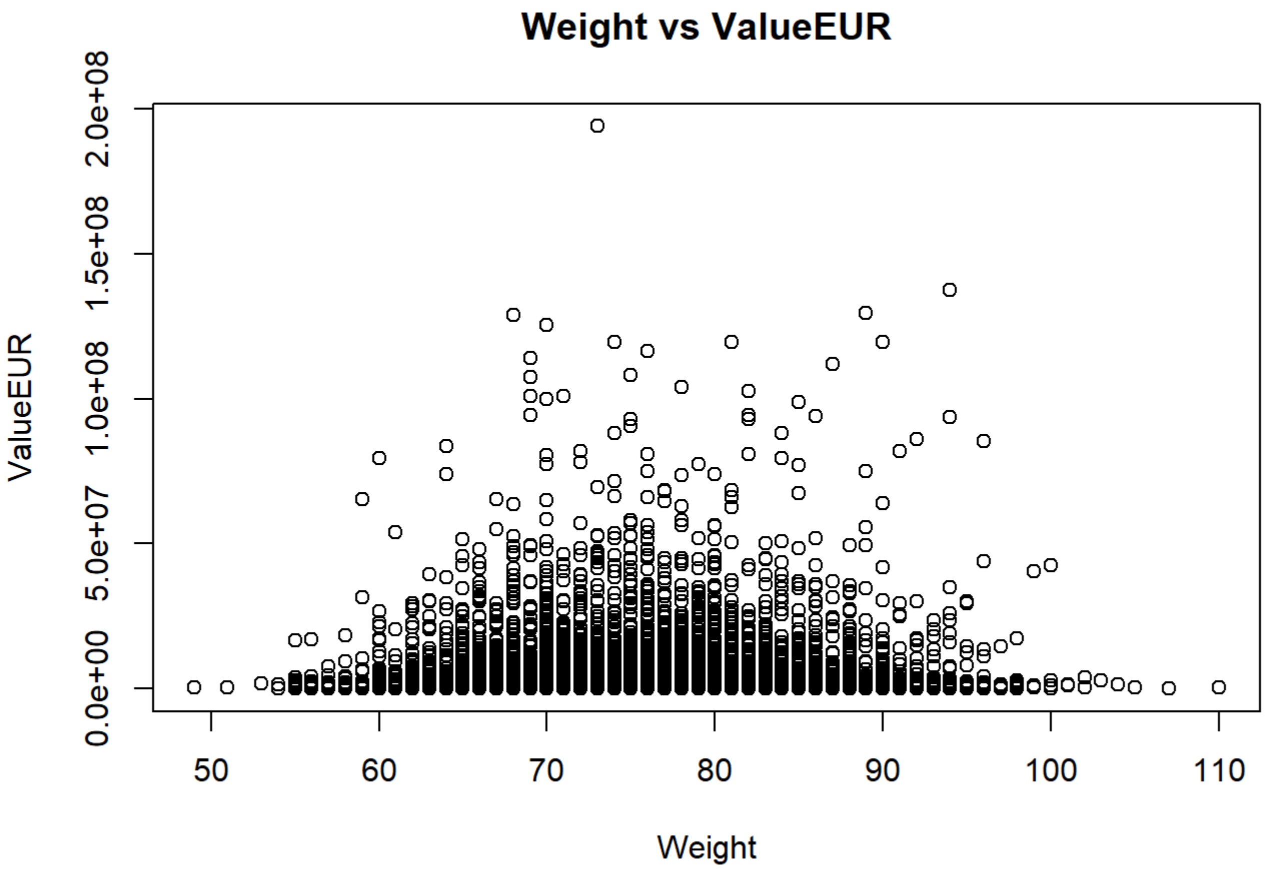
```
plot(fifaData$Height,fifaData$Weight, main = "Height vs Weight", xlab="Height", ylab="Weight")
```



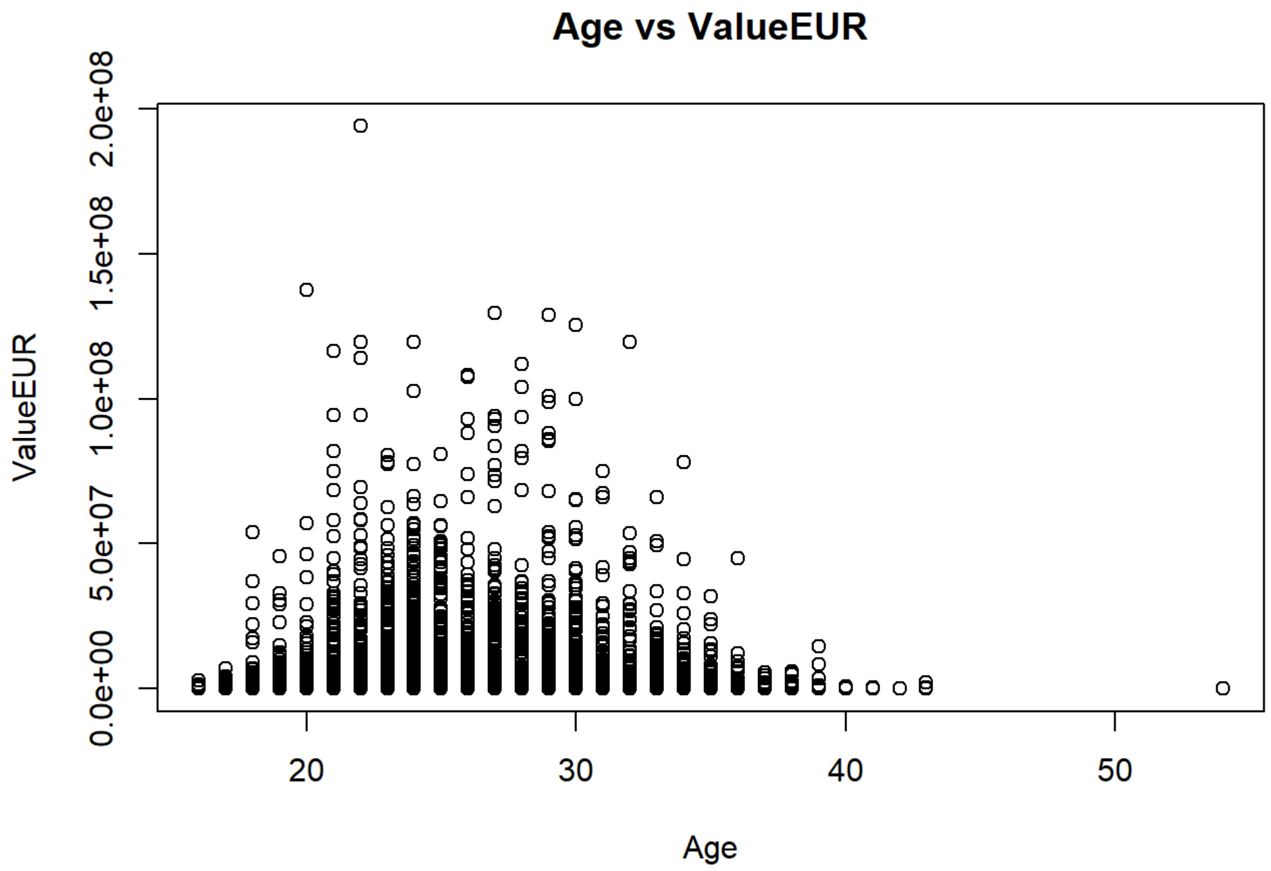
```
plot(fifaData$Height,fifaData$ValueEUR, main = "Height vs ValueEUR", xlab="Height", ylab="ValueE
UR")
```



```
plot(fifaData$Weight,fifaData$ValueEUR, main = "Weight vs ValueEUR", xlab="Weight", ylab="ValueE
UR")
```



```
plot(fifaData$Age,fifaData$ValueEUR, main = "Age vs ValueEUR", xlab="Age", ylab="ValueEUR")
```



## What our summary tells us:

Our Residuals have a wide range however, our median is close to the 1st and 3rd quarter which tells us there are a lot of outliers. Our P value is also low telling us that our predictor (player value) does influence our target(player wage). The player’s value has a small stander error and high t-value which also suggests a relationship but our intercept is notably less accurate. Lastly our R-squared isn’t too high which means the player’s value can only give us a rough idea of the player’s wage.

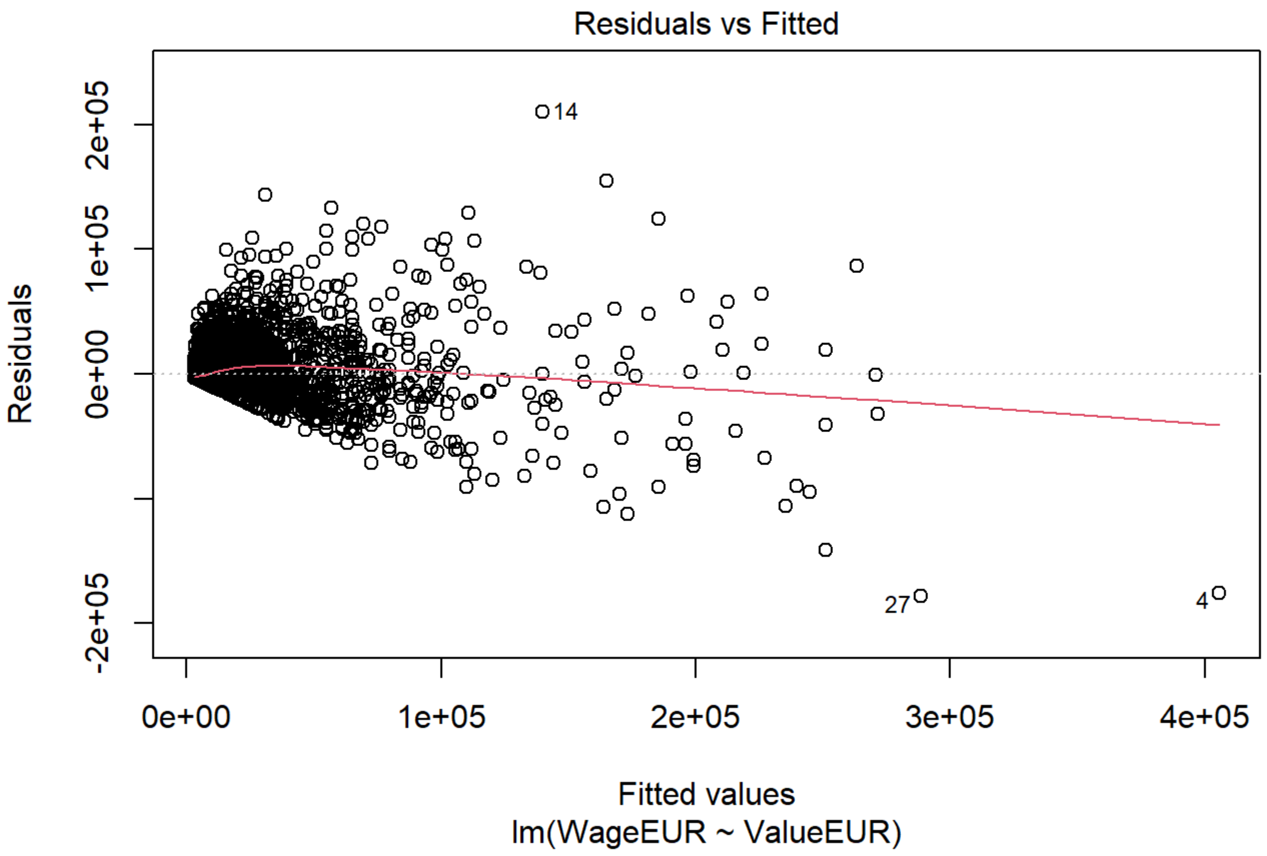
## What our regression plots shows us:

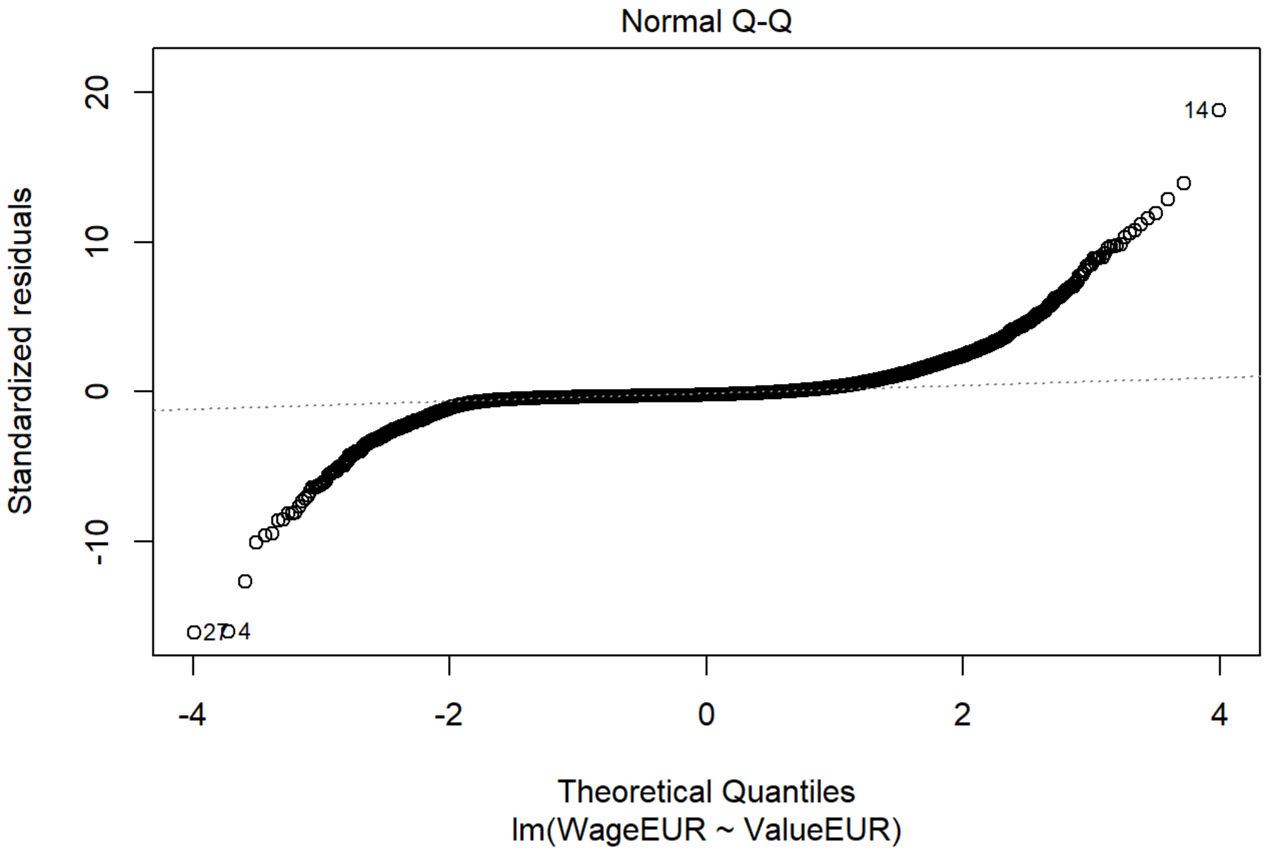
Residual vs Fitted: this plot shows us the pattern of our residuals, ours is mostly horizontal which means our target and predictor have a mostly linear relationship. Normal Q-Q: The plot bends at the ends of graph, which means that residuals are not properly distributed along smaller and larger values. Scale-Location: we can see our line isn’t horizontal at all which means that our player’s wage is not spread evenly along the range of the player’s value Residuals vs Leverage: we have a few outliers outside the Cook’s distance and some others that are near it

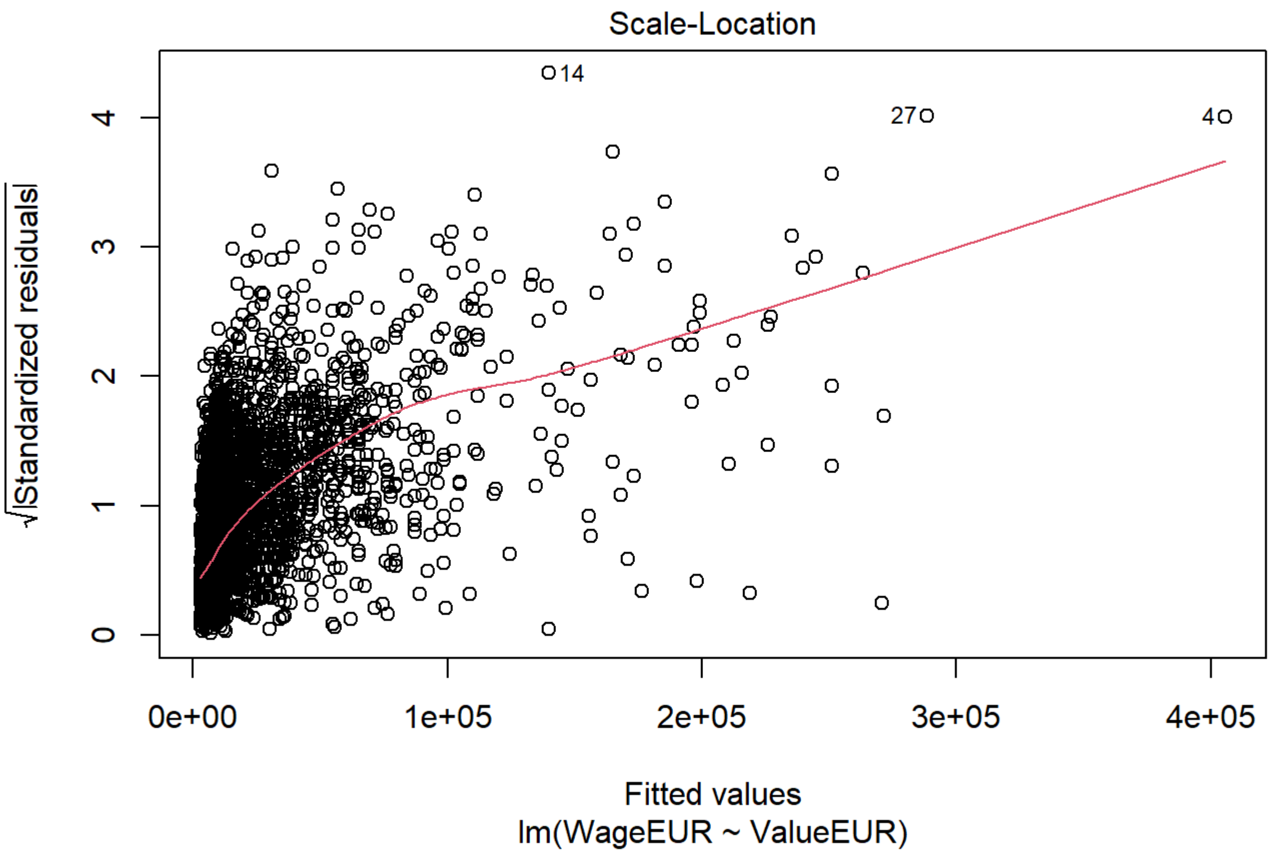
```
lmValue <- lm(formula = WageEUR ~ ValueEUR, data = train)
summary(lmValue)
```

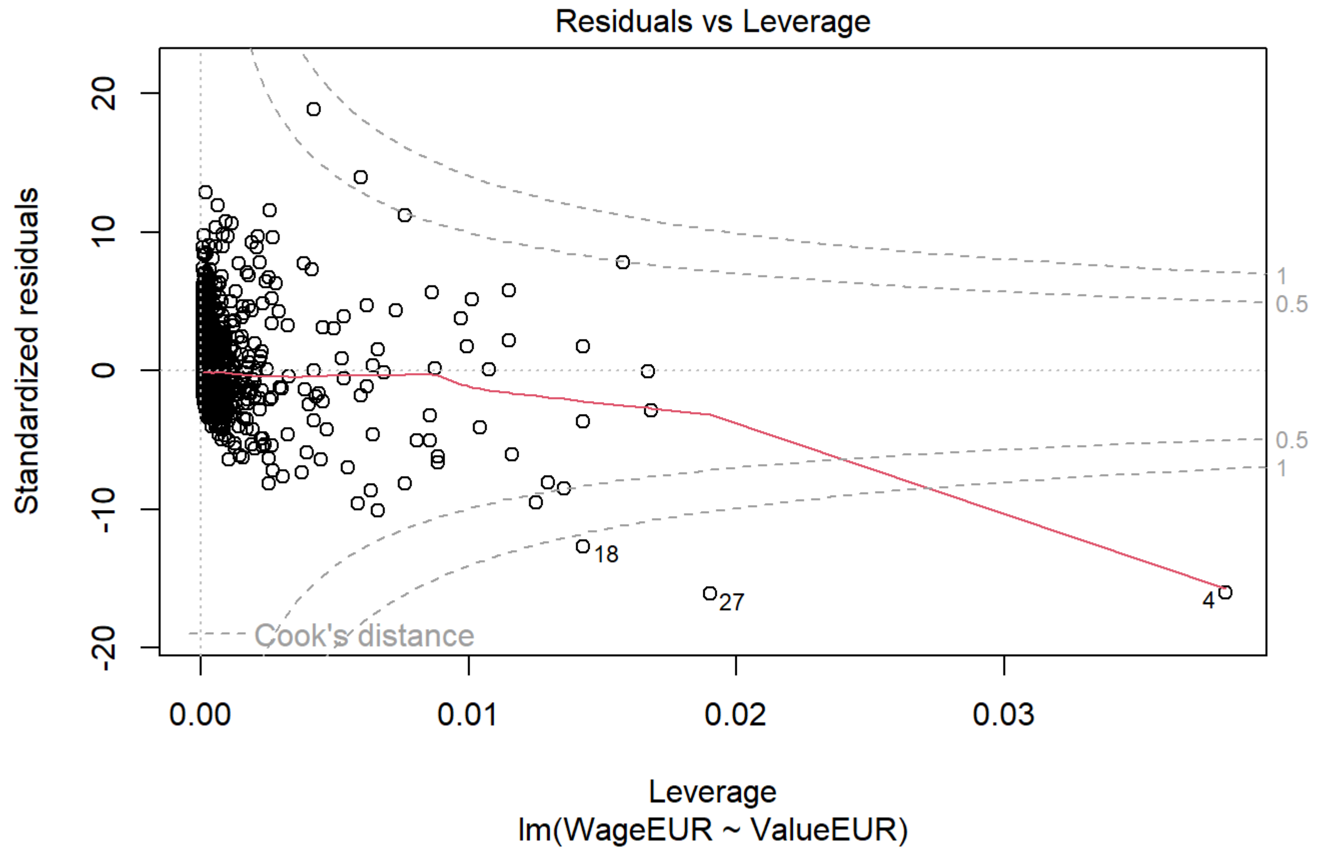
```
##
## Call:
## lm(formula = WageEUR ~ ValueEUR, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -178311   -3334   -2249     625  210022
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.056e+03  9.604e+01   31.82   <2e-16 ***
## ValueEUR    2.075e-03  1.142e-05  181.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11170 on 15345 degrees of freedom
## Multiple R-squared:  0.6826, Adjusted R-squared:  0.6826
## F-statistic: 3.3e+04 on 1 and 15345 DF,  p-value: < 2.2e-16
```

```
plot(lmValue)
```


Residuals vs Fitted
Residuals
2e+05
1e+05
0e+00
-2e+05
14
27
4
0e+00
1e+05
2e+05
3e+05
4e+05
Fitted values
lm(WageEUR ~ ValueEUR)


Normal Q-Q
Standardized residuals
20
10
0
-10
14
27
4
-4
-2
0
2
4
Theoretical Quantiles
lm(WageEUR ~ ValueEUR)





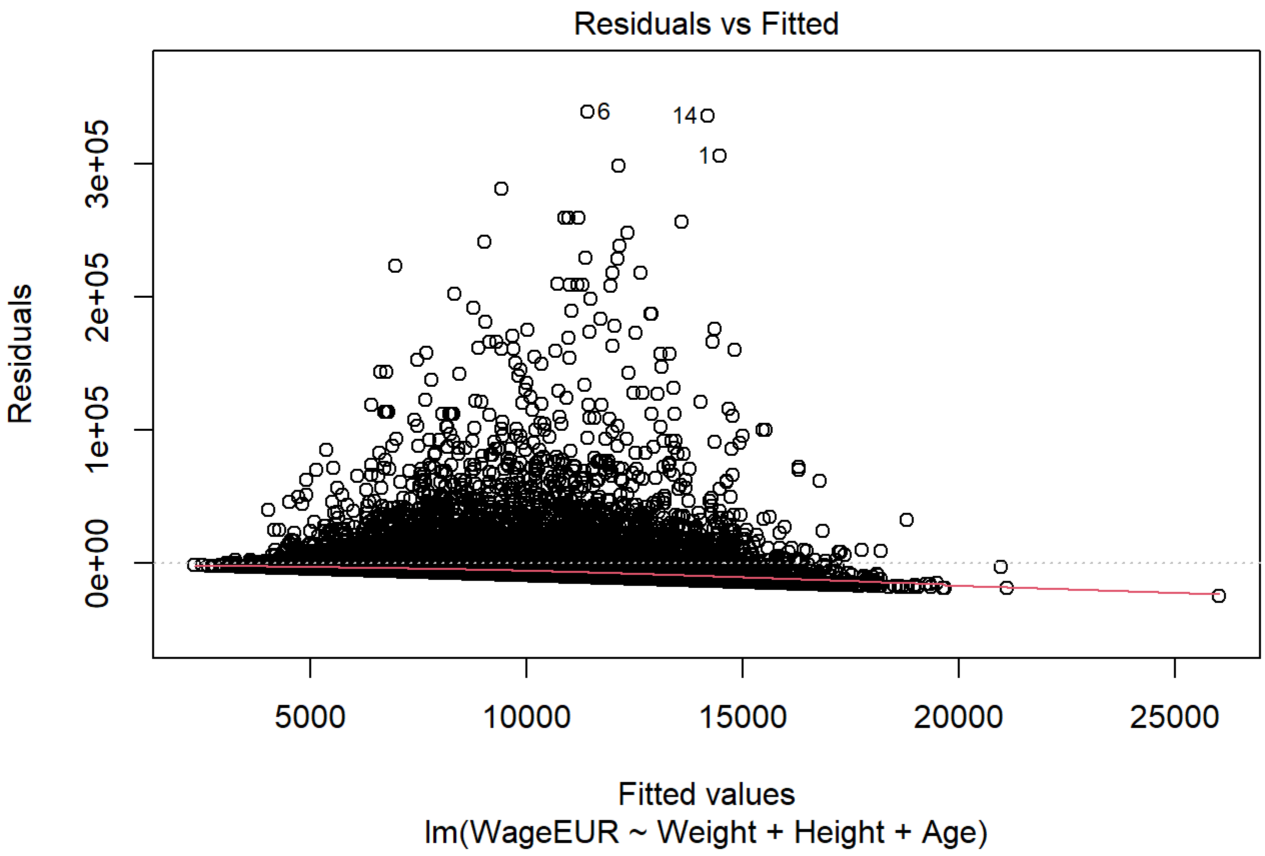
###

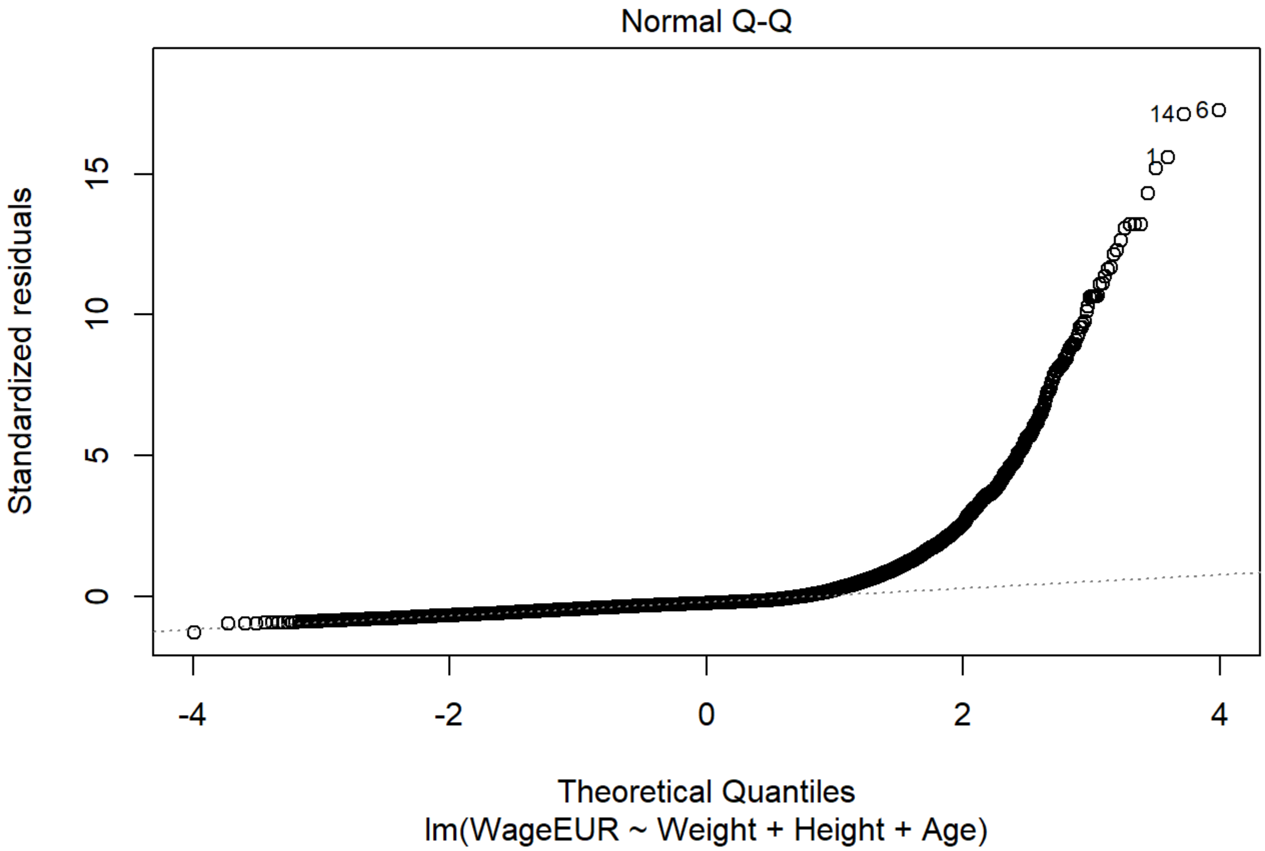
Multiple linear regression model Wage prediction using only physical characteristics

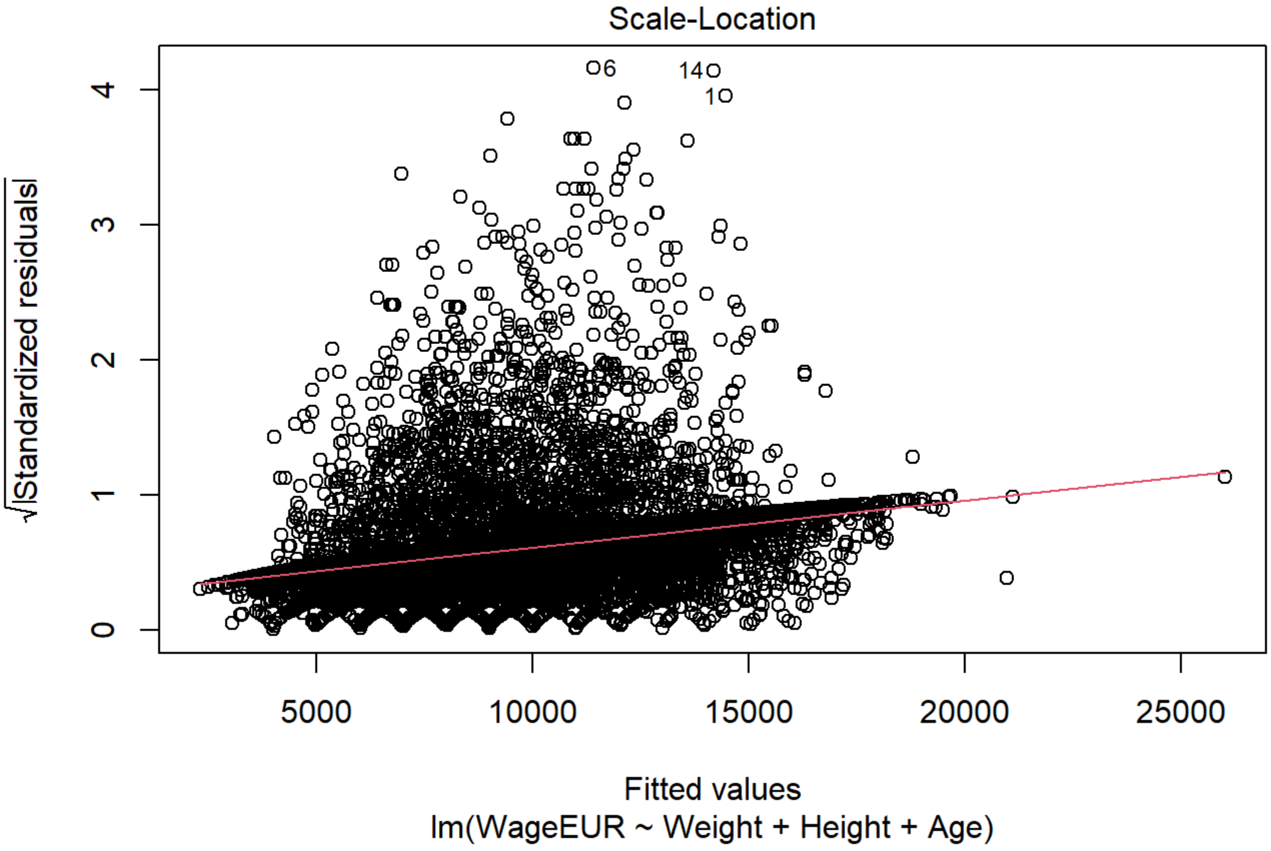
```
lmPhy <- lm(formula = WageEUR ~ Weight + Height + Age, data = train)
summary(lmPhy)
```

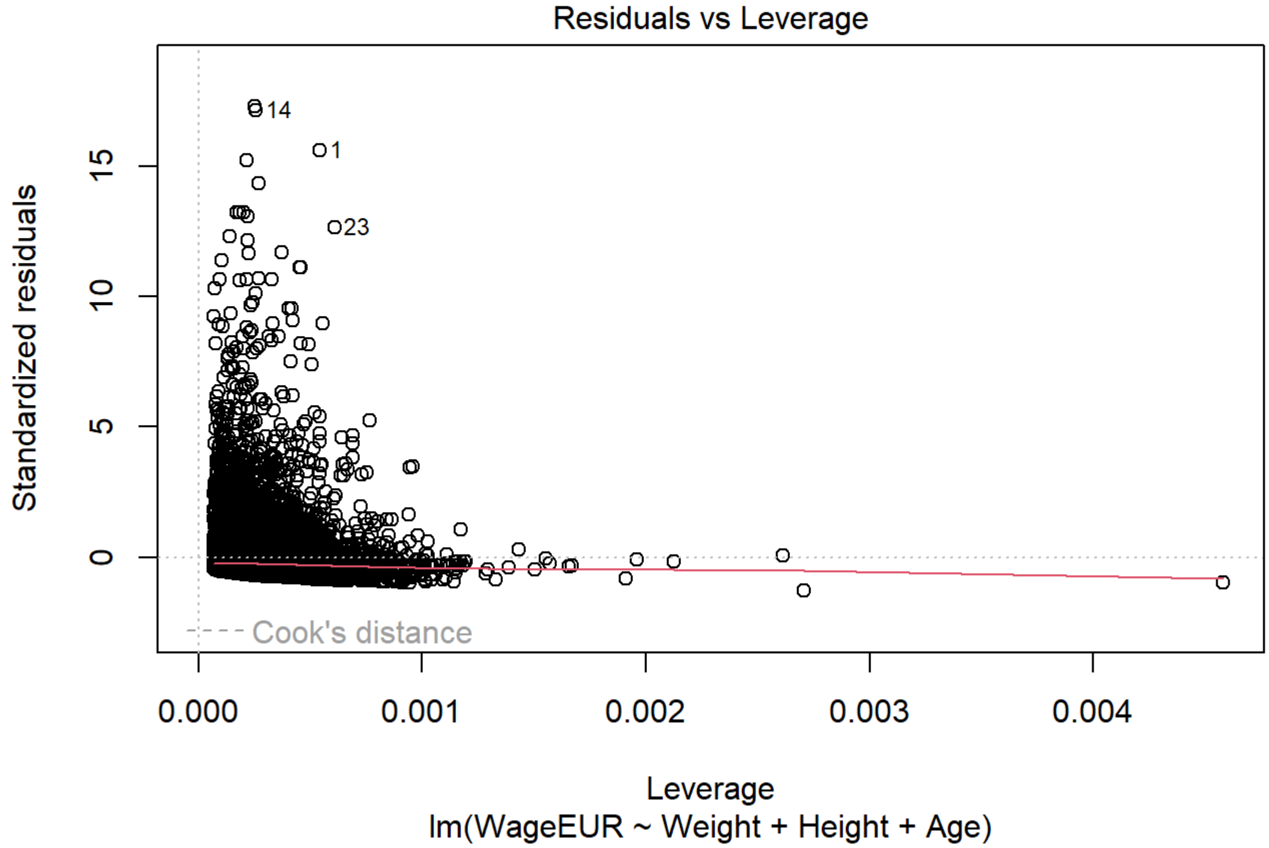
```
##
## Call:
## lm(formula = WageEUR ~ Weight + Height + Age, data = train)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -25308  -7301  -4712   -806 338583
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5953.21    4945.48  -1.204  0.22870
## Weight        105.81      36.04   2.936  0.00333 **
## Height        -43.07      36.22  -1.189  0.23442
## Age           591.97      34.62  17.098  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19590 on 15343 degrees of freedom
## Multiple R-squared:  0.02305,    Adjusted R-squared:  0.02286
## F-statistic: 120.6 on 3 and 15343 DF,  p-value: < 2.2e-16
```

```
plot(lmPhy)
```


Residuals vs Fitted
Residuals
0e+00
1e+05
2e+05
3e+05
6
14
1
5000
10000
15000
20000
25000
Fitted values
lm(WageEUR ~ Weight + Height + Age)


Normal Q-Q
Standardized residuals
0
5
10
15
14
6
1
-4
-2
0
2
4
Theoretical Quantiles
lm(WageEUR ~ Weight + Height + Age)





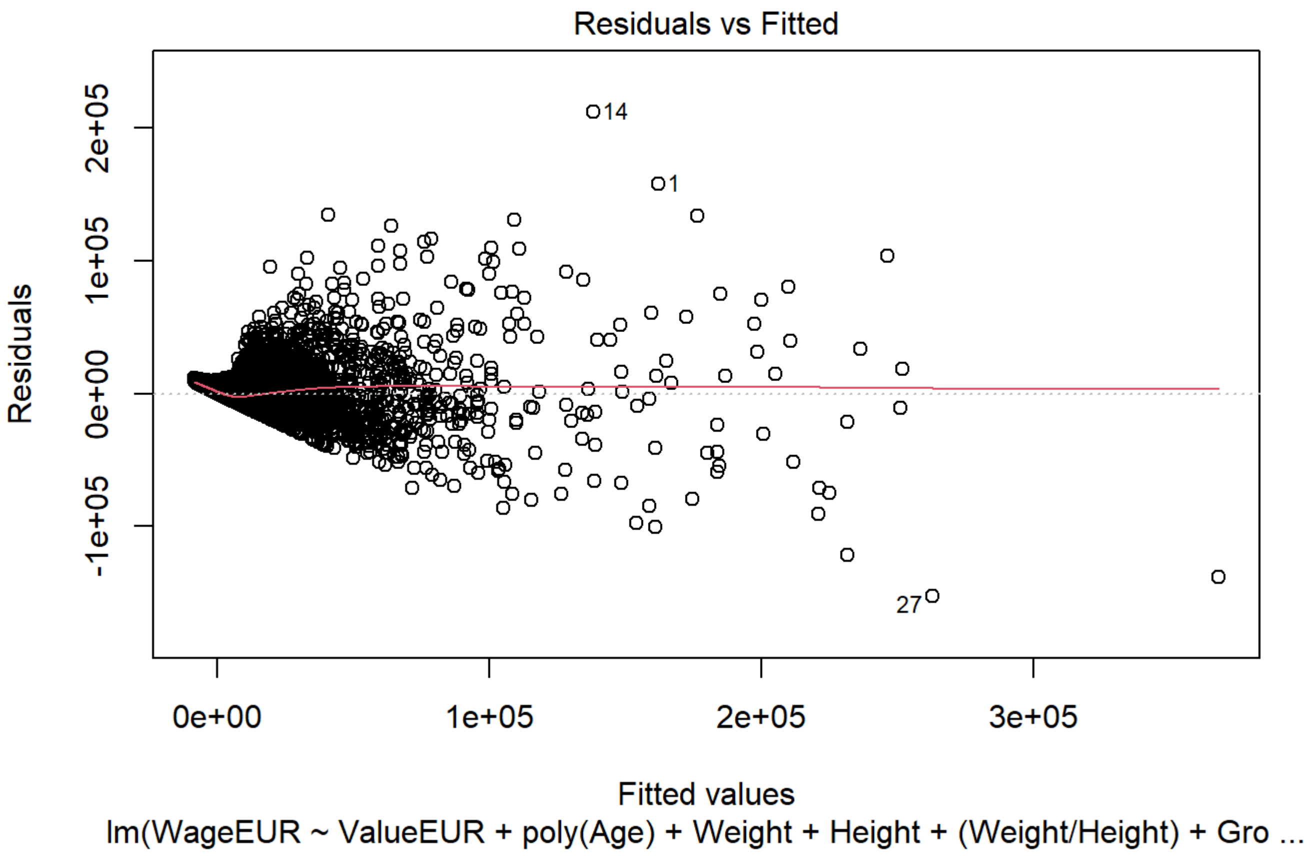
###

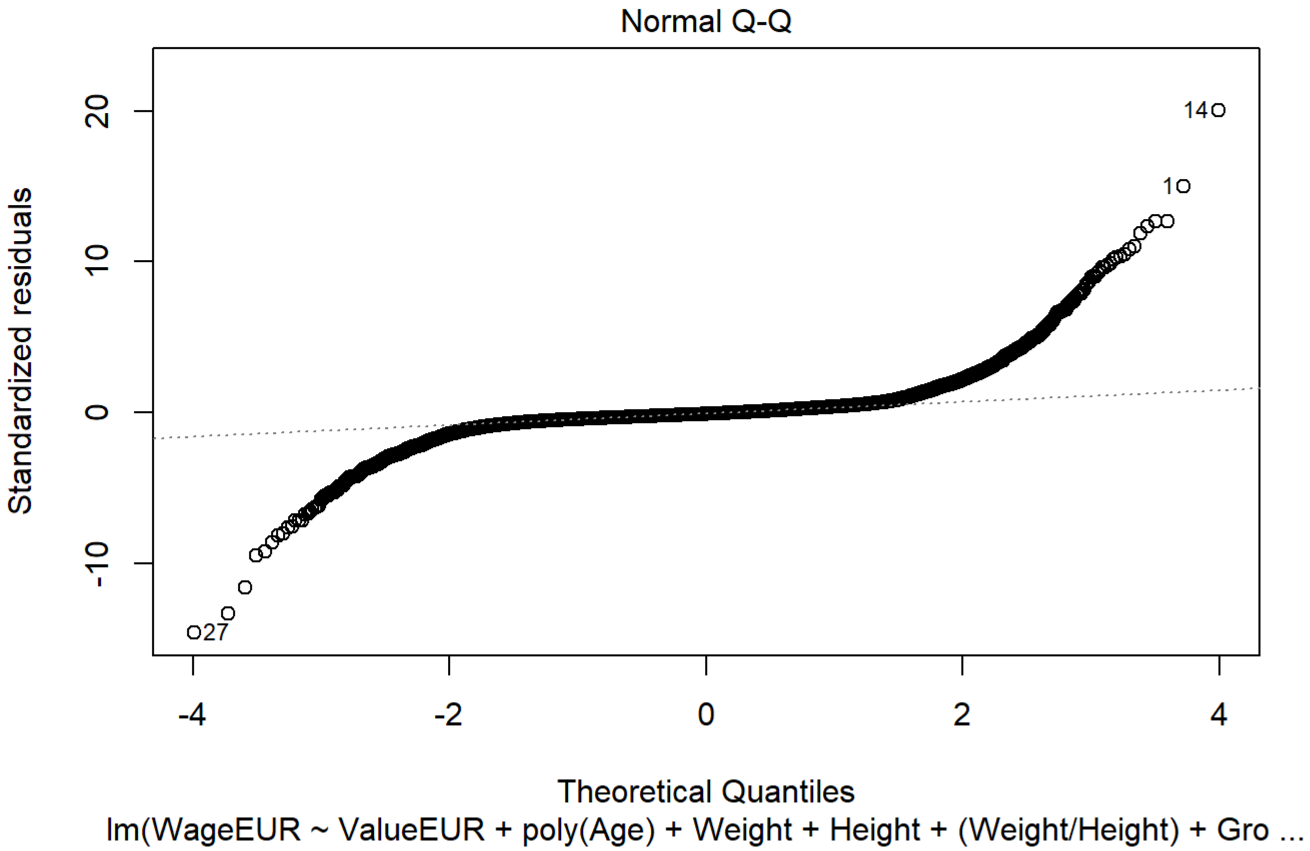
3rd Regression model Wage prediction using player’s value and physical characteristics in a weird combination

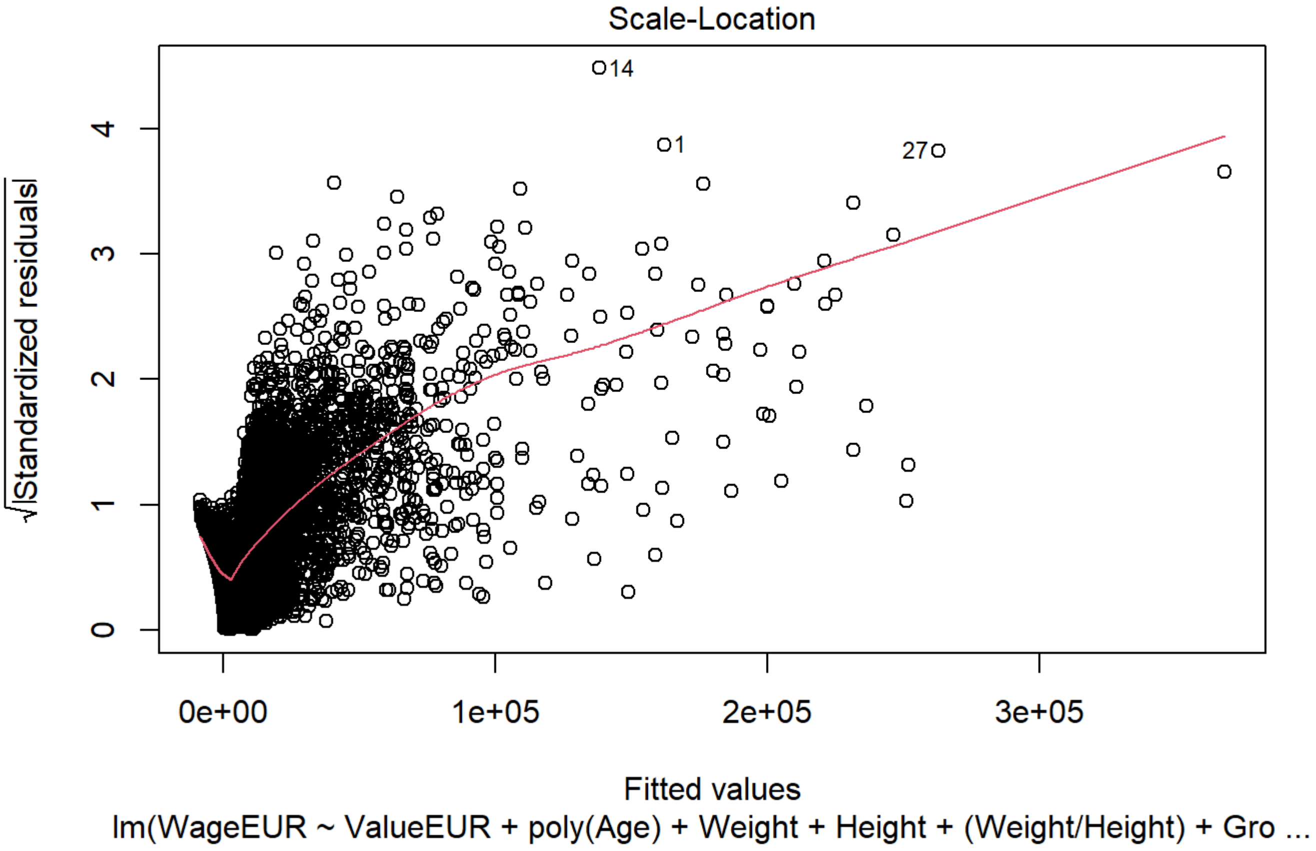
```
lmAll <- lm(formula = WageEUR ~ ValueEUR + poly(Age) + Weight + Height + (Weight / Height) + Gro
wth + Overall, data = train)
summary(lmAll)
```

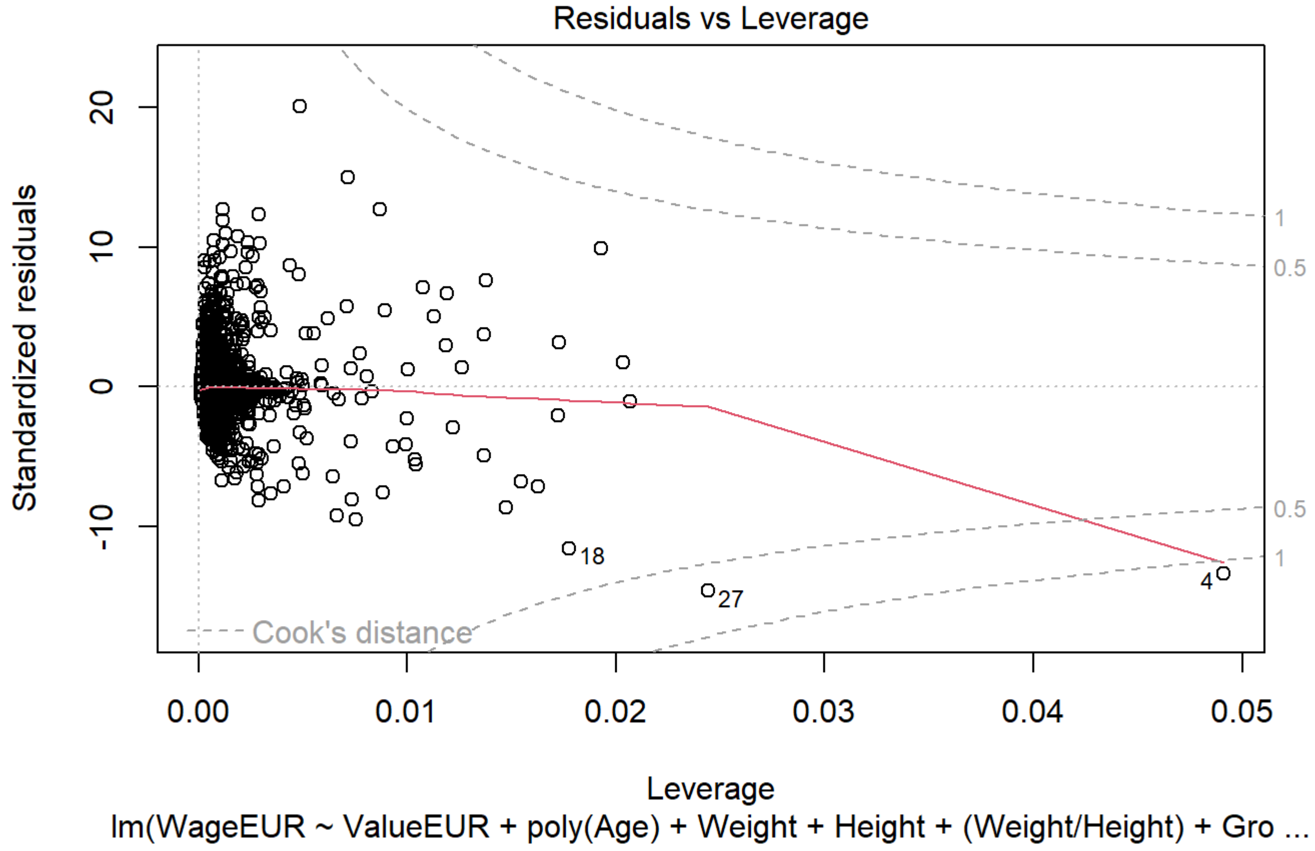
```
##
## Call:
## lm(formula = WageEUR ~ ValueEUR + poly(Age) + Weight + Height +
##     (Weight/Height) + Growth + Overall, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -152731   -3432   -1012    2136  211779
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -7.757e+04  2.076e+04  -3.737 0.000187 ***
## ValueEUR       1.809e-03  1.355e-05 133.460  < 2e-16 ***
## poly(Age)      2.125e+05  2.132e+04   9.967  < 2e-16 ***
## Weight         4.544e+02  2.807e+02   1.619 0.105434
## Height         2.500e+02  1.143e+02   2.188 0.028702 *
## Growth         2.028e+02  3.277e+01   6.189 6.21e-10 ***
## Overall        5.709e+02  1.822e+01  31.337  < 2e-16 ***
## Weight:Height -2.689e+00  1.530e+00  -1.758 0.078818 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10570 on 15339 degrees of freedom
## Multiple R-squared:  0.7156, Adjusted R-squared:  0.7155
## F-statistic:  5514 on 7 and 15339 DF,  p-value: < 2.2e-16
```

```
plot(lmAll)
```


Residuals vs Fitted
Residuals
2e+05
1e+05
0e+00
-1e+05
14
1
27
0e+00
1e+05
2e+05
3e+05
Fitted values
lm(WageEUR ~ ValueEUR + poly(Age) + Weight + Height + (Weight/Height) + Gro ...


Normal Q-Q
Standardized residuals
20
10
0
-10
14
1
27
-4
-2
0
2
4
Theoretical Quantiles
lm(WageEUR ~ ValueEUR + poly(Age) + Weight + Height + (Weight/Height) + Gro ...


Scale-Location
√|Standardized residuals|
0
1
2
3
4
0e+00
1e+05
2e+05
3e+05
Fitted values
lm(WageEUR ~ ValueEUR + poly(Age) + Weight + Height + (Weight/Height) + Gro ...
14
1
27


Residuals vs Leverage
Standardized residuals
-10
0
10
20
0.00
0.01
0.02
0.03
0.04
0.05
Leverage
lm(WageEUR ~ ValueEUR + poly(Age) + Weight + Height + (Weight/Height) + Gro ...
Cook's distance
1
0.5
0.5
1
18
27
4

# Model Comparison

Based on the three linear regression models we see that a player’s value is the best indicator of their wage and that their physical characteristics are not however, in the 3rd model that used all the variables it’s R-squared value was 0.03 higher than the 1st model making it slightly more accurate. So even though the physical characteristics aren’t a good predictor, they can still improve an already good one. Also I don’t know why but age is the only characteristic that is polynomial, none of the others are for some reason.

```
#Original model
print("Original model")
```

```
## [1] "Original model"
```

```
pred <- predict(lmValue, test)
correlation <- cor(pred, test)
print(paste("correlation: ", correlation))
```

```
## [1] "correlation:  0.0521831449139791" "correlation:  0.0213938506066272"
## [3] "correlation:  0.0408802922955291" "correlation:  0.59949369986399"
## [5] "correlation:  -0.117824880444884" "correlation:  0.567838174716748"
## [7] "correlation:  0.821487523079122"  "correlation:  1"
```

```
mse <- mean((pred - test$WageEUR)^2)
print(paste("mse: ", mse))
```

```
## [1] "mse:  105714898.376408"
```

```
rmse <- sqrt(mse)
print(paste("rmse: ", rmse))
```

```
## [1] "rmse:  10281.7750596095"
```

```
#Physical characteristics model
print("Physical characteristics model")
```

```
## [1] "Physical characteristics model"
```

```
pred <- predict(lmPhy, test)
correlation <- cor(pred, test)
print(paste("correlation: ", correlation))
```

```
## [1] "correlation:  0.984204100499768"  "correlation:  0.17409465641443"
## [3] "correlation:  0.414746948686233"  "correlation:  0.463306160042564"
## [5] "correlation:  -0.854778925708158" "correlation:  -0.259425339885656"
## [7] "correlation:  0.17335595833715"   "correlation:  0.0565618976917593"
```

```
mse <- mean((pred - test$WageEUR)^2)
print(paste("mse: ", mse))
```

```
## [1] "mse:  309499198.563844"
```

```
rmse <- sqrt(mse)
print(paste("rmse: ", rmse))
```

```
## [1] "rmse:  17592.5893081105"
```

```
#Best model
print("Best model")
```

```
## [1] "Best model"
```

```
pred <- predict(lmAll, test)
correlation <- cor(pred, test)
print(paste("correlation: ", correlation))
```

```
## [1] "correlation:  0.214496297261678"  "correlation:  0.0480750170341174"
## [3] "correlation:  0.0940517300415538" "correlation:  0.768238438853873"
## [5] "correlation:  -0.262928679254263" "correlation:  0.625327447472819"
## [7] "correlation:  0.835059949565218"  "correlation:  0.970417049189244"
```

```
mse <- mean((pred - test$WageEUR)^2)
print(paste("mse: ", mse))
```

```
## [1] "mse:  96982906.6534792"
```

```
rmse <- sqrt(mse)
print(paste("rmse: ", rmse))
```

```
## [1] "rmse:  9847.98998037057"
```

# Evaluations

As you can see the mean square errors and their square roots reaffirm what we said previously. Using only physical characteristics is worse than using the player’s value but using both is the most accurate.