ANALYSIS AND PREDCITION OF PRODUCTIVITY OF GARMENT EMPLOYEES

NAME: HANUMANTH RAM SAI JEETESH CHAMANA, UB PERSON ID: <50468947>
NAME: MAHIMITRA CHIRALA, UB PERSON ID: <50464542>
NAME: SAI KIRAN PATURI, UB PERSON ID: <50442942>

2022-12-07

"DATA DESCRIPTION AND ANALYSIS"

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ISLR2)
library(ggplot2)
library(dplyr)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(superml)
```

```
## Warning: package 'superml' was built under R version 4.2.2
```

```
## Loading required package: R6
```

```
library(Metrics)
```

```
## Warning: package 'Metrics' was built under R version 4.2.2
```

```r
library(imager)
```

```
## Warning: package 'imager' was built under R version 4.2.2

## Loading required package: magrittr
##
## Attaching package: 'magrittr'
##
## The following object is masked from 'package:purrr':
##
##     set_names
##
## The following object is masked from 'package:tidyr':
##
##     extract
##
##
## Attaching package: 'imager'
##
## The following object is masked from 'package:magrittr':
##
##     add
##
## The following object is masked from 'package:stringr':
##
##     boundary
##
## The following object is masked from 'package:tidyr':
##
##     fill
##
## The following objects are masked from 'package:stats':
##
##     convolve, spectrum
##
## The following object is masked from 'package:graphics':
##
##     frame
##
## The following object is masked from 'package:base':
##
##     save.image
```

```r
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 4.2.2
```

```r
df <- read.csv("C:\\Users\\jeete\\OneDrive\\Desktop\\R Projects\\Project\\garments_worker_productivity.c
#"The below shows a summary of the dataset"
names(df)
```

```
##  [1] "date"                  "quarter"               "department"
```

```
##  [4] "day"                   "team"                  "targeted_productivity"
##  [7] "smv"                   "wip"                   "over_time"
## [10] "incentive"             "idle_time"             "idle_men"
## [13] "no_of_style_change"    "no_of_workers"         "actual_productivity"
```

```
glimpse(df)
```

```
## Rows: 1,197
## Columns: 15
## $ date                  <chr> "1/1/2015", "1/1/2015", "1/1/2015", "1/1/2015", ~
## $ quarter               <chr> "Quarter1", "Quarter1", "Quarter1", "Quarter1", ~
## $ department            <chr> "sweing", "finishing ", "sweing", "sweing", "swe~
## $ day                   <chr> "Thursday", "Thursday", "Thursday", "Thursday", ~
## $ team                  <int> 8, 1, 11, 12, 6, 7, 2, 3, 2, 1, 9, 10, 5, 10, 8,~
## $ targeted_productivity <dbl> 0.80, 0.75, 0.80, 0.80, 0.80, 0.80, 0.75, 0.75, ~
## $ smv                   <dbl> 26.16, 3.94, 11.41, 11.41, 25.90, 25.90, 3.94, 2~
## $ wip                   <int> 1108, NA, 968, 968, 1170, 984, NA, 795, 733, 681~
## $ over_time             <int> 7080, 960, 3660, 3660, 1920, 6720, 960, 6900, 60~
## $ incentive             <int> 98, 0, 50, 50, 50, 38, 0, 45, 34, 45, 44, 45, 50~
## $ idle_time             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ idle_men              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ no_of_style_change    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ no_of_workers         <dbl> 59.0, 8.0, 30.5, 30.5, 56.0, 56.0, 8.0, 57.5, 55~
## $ actual_productivity   <dbl> 0.9407254, 0.8865000, 0.8005705, 0.8005705, 0.80~
```

```
summary(df)
```

```
##      date             quarter           department            day
##  Length:1197        Length:1197        Length:1197        Length:1197
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##       team        targeted_productivity      smv             wip
##  Min.   : 1.000   Min.   :0.0700        Min.   : 2.90   Min.   :    7.0
##  1st Qu.: 3.000   1st Qu.:0.7000        1st Qu.: 3.94   1st Qu.:  774.5
##  Median : 6.000   Median :0.7500        Median :15.26   Median : 1039.0
##  Mean   : 6.427   Mean   :0.7296        Mean   :15.06   Mean   : 1190.5
##  3rd Qu.: 9.000   3rd Qu.:0.8000        3rd Qu.:24.26   3rd Qu.: 1252.5
##  Max.   :12.000   Max.   :0.8000        Max.   :54.56   Max.   :23122.0
##                                                         NA's   :506
##    over_time       incentive         idle_time           idle_men
##  Min.   :    0   Min.   :   0.00   Min.   :  0.0000   Min.   : 0.0000
##  1st Qu.: 1440   1st Qu.:   0.00   1st Qu.:  0.0000   1st Qu.: 0.0000
##  Median : 3960   Median :   0.00   Median :  0.0000   Median : 0.0000
##  Mean   : 4567   Mean   :  38.21   Mean   :  0.7302   Mean   : 0.3693
##  3rd Qu.: 6960   3rd Qu.:  50.00   3rd Qu.:  0.0000   3rd Qu.: 0.0000
##  Max.   :25920   Max.   :3600.00   Max.   :300.0000   Max.   :45.0000
##
##  no_of_style_change no_of_workers   actual_productivity
##  Min.   :0.0000     Min.   : 2.00   Min.   :0.2337
```

```
##  1st Qu.:0.0000     1st Qu.: 9.00   1st Qu.:0.6503
##  Median :0.0000     Median :34.00   Median :0.7733
##  Mean   :0.1504     Mean   :34.61   Mean   :0.7351
##  3rd Qu.:0.0000     3rd Qu.:57.00   3rd Qu.:0.8503
##  Max.   :2.0000     Max.   :89.00   Max.   :1.1204
##
```

```r
#"Since quarter, date and time are the attributes that are related to time, and Quarter seems to be a g
df = subset(df, select = -c(date, day) )
#"Next the null values are handled "
dim(df)
```

```
## [1] 1197   13
```

```r
is.null(df)
```

```
## [1] FALSE
```

```r
lapply(df,function(x) { length(which(is.na(x)))})
```

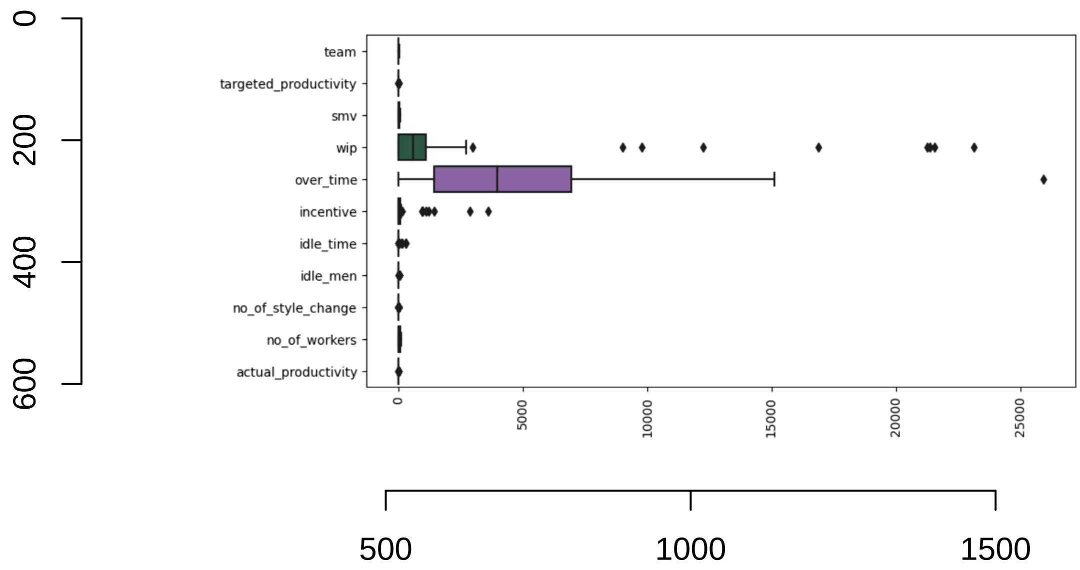
```
## $quarter
## [1] 0
##
## $department
## [1] 0
##
## $team
## [1] 0
##
## $targeted_productivity
## [1] 0
##
## $smv
## [1] 0
##
## $wip
## [1] 506
##
## $over_time
## [1] 0
##
## $incentive
## [1] 0
##
## $idle_time
## [1] 0
##
## $idle_men
## [1] 0
##
## $no_of_style_change
## [1] 0
```

```
##
## $no_of_workers
## [1] 0
##
## $actual_productivity
## [1] 0
```

```r
#By checking the wip column has 506 null values
df <- df %>% replace(is.na(.), 0)#The null values are replaced with zero
lapply(df,function(x) { length(which(is.na(x)))})#Checked again, no null values
```

```
## $quarter
## [1] 0
##
## $department
## [1] 0
##
## $team
## [1] 0
##
## $targeted_productivity
## [1] 0
##
## $smv
## [1] 0
##
## $wip
## [1] 0
##
## $over_time
## [1] 0
##
## $incentive
## [1] 0
##
## $idle_time
## [1] 0
##
## $idle_men
## [1] 0
##
## $no_of_style_change
## [1] 0
##
## $no_of_workers
## [1] 0
##
## $actual_productivity
## [1] 0
```

```r
#"Outliers are observed in the data, the below figure shows the analysis of data using box plots"
im <- load.image("C:\\Users\\jeete\\OneDrive\\Desktop\\R Projects\\Project\\Boxplot-with outlier.png")
plot(im)
```




```
i_Q1=quantile(df$incentive, probs = c(.25))
i_Q3=quantile(df$incentive, probs = c(.75))
i_IQR=i_Q3-i_Q1
i_lower = i_Q1 - 1.5*i_IQR
i_upper = i_Q3 + 1.5*i_IQR
df1<-df[df$incentive >i_lower & df$incentive <i_upper, ]
dim(df1)
```

```
## [1] 1186   13
```

```
wip_Q1=quantile(df1$wip, probs = c(.25))
wip_Q3=quantile(df1$wip, probs = c(.75))
wip_IQR=wip_Q3-wip_Q1
wip_lower = wip_Q1 - 1.5*wip_IQR
wip_upper = wip_Q3 + 1.5*wip_IQR
wip_lower
```

```
##       25%
## -1627.125
```

```
df2<-df1[df1$wip >wip_lower & df1$wip<wip_upper, ]
dim(df2)
```

```
## [1] 1177   13
```

```r
ot_Q1=quantile(df2$over_time, probs = c(.25))
ot_Q3=quantile(df2$over_time, probs = c(.75))
ot_IQR=ot_Q3-ot_Q1
ot_lower = ot_Q1 - 1.5*ot_IQR
ot_upper = ot_Q3 + 1.5*ot_IQR
ot_lower
```

```
##   25%
## -6840
```

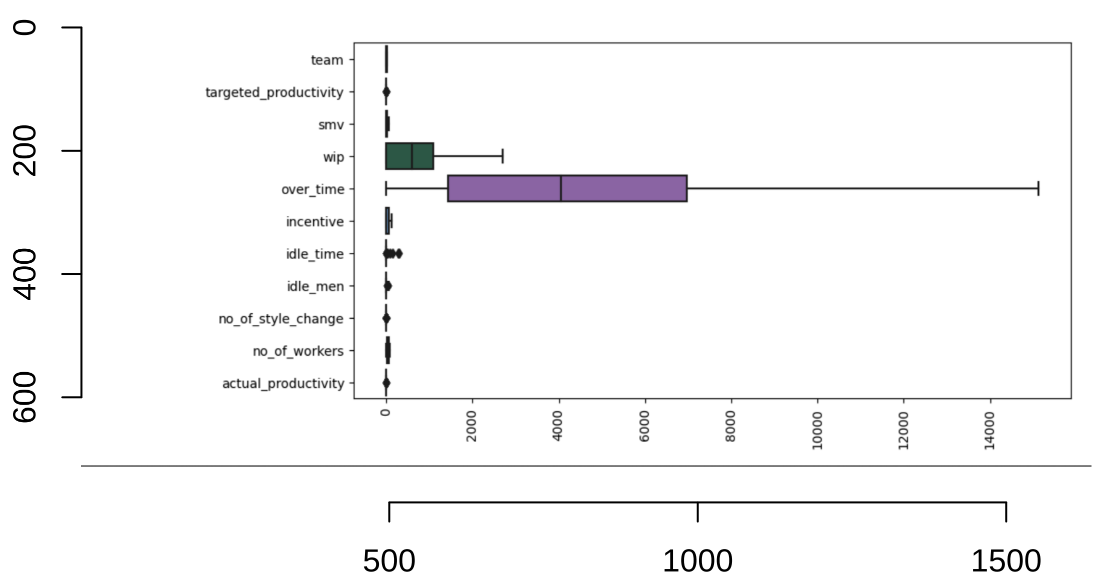
```r
f_df<-df2[df2$over_time >ot_lower & df2$over_time<ot_upper, ]
dim(f_df)
```

```
## [1] 1176   13
```

```r
dim(df)
```

```
## [1] 1197   13
```

```r
#So after updating nan's and removing outliers the dataset dimension is 1176*13
#"The below plot shows the analysis of data after the outliers are handled."
im <- load.image("C:\\Users\\jeete\\OneDrive\\Desktop\\R Projects\\Project\\Boxplot-withoutoutlier.png")
plot(im)
```



```r
#Label encoding
f_df$quarter <- as.numeric(factor(f_df$quarter))
head(f_df)
```

```
##   quarter department team targeted_productivity   smv  wip over_time incentive
## 1       1     sweing    8                  0.80 26.16 1108      7080        98
## 2       1 finishing     1                  0.75  3.94    0       960         0
## 3       1     sweing   11                  0.80 11.41  968      3660        50
## 4       1     sweing   12                  0.80 11.41  968      3660        50
## 5       1     sweing    6                  0.80 25.90 1170      1920        50
## 6       1     sweing    7                  0.80 25.90  984      6720        38
##   idle_time idle_men no_of_style_change no_of_workers actual_productivity
## 1         0        0                  0          59.0           0.9407254
## 2         0        0                  0           8.0           0.8865000
## 3         0        0                  0          30.5           0.8005705
## 4         0        0                  0          30.5           0.8005705
## 5         0        0                  0          56.0           0.8003819
## 6         0        0                  0          56.0           0.8001250
```

```r
f_df$department[f_df$department == "finishing " |
                  f_df$department == "finishing"] <- 1
f_df$department[f_df$department == "sweing" ] <- 2
head(f_df)
```

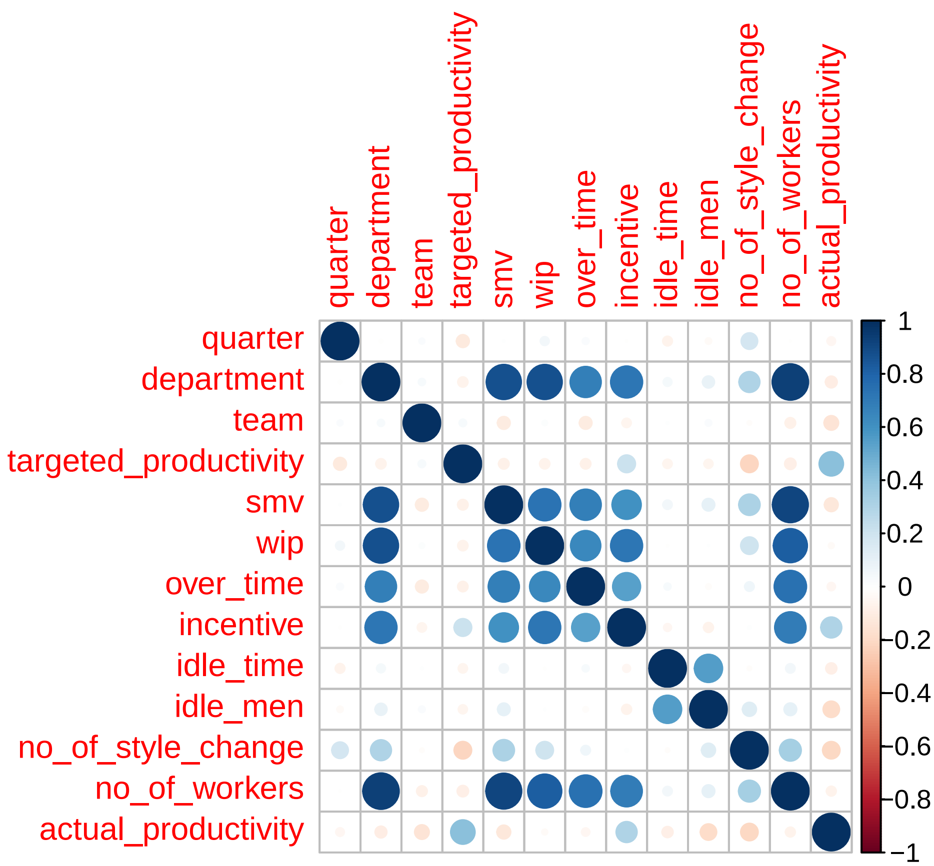
```
##   quarter department team targeted_productivity   smv  wip over_time incentive
## 1       1          2    8                  0.80 26.16 1108      7080        98
## 2       1          1    1                  0.75  3.94    0       960         0
## 3       1          2   11                  0.80 11.41  968      3660        50
## 4       1          2   12                  0.80 11.41  968      3660        50
## 5       1          2    6                  0.80 25.90 1170      1920        50
## 6       1          2    7                  0.80 25.90  984      6720        38
##   idle_time idle_men no_of_style_change no_of_workers actual_productivity
## 1         0        0                  0          59.0           0.9407254
## 2         0        0                  0           8.0           0.8865000
## 3         0        0                  0          30.5           0.8005705
## 4         0        0                  0          30.5           0.8005705
## 5         0        0                  0          56.0           0.8003819
## 6         0        0                  0          56.0           0.8001250
```

```r
tail(f_df)
```

```
##      quarter department team targeted_productivity   smv wip over_time
## 1192       2          2    7                  0.65 30.48 935      6840
## 1193       2          1   10                  0.75  2.90   0       960
## 1194       2          1    8                  0.70  3.90   0       960
## 1195       2          1    7                  0.65  3.90   0       960
## 1196       2          1    9                  0.75  2.90   0      1800
## 1197       2          1    6                  0.70  2.90   0       720
##      incentive idle_time idle_men no_of_style_change no_of_workers
## 1192        26         0        0                  1            57
## 1193         0         0        0                  0             8
## 1194         0         0        0                  0             8
```

```
## 1195         0         0        0                  0             8
## 1196         0         0        0                  0            15
## 1197         0         0        0                  0             6
##      actual_productivity
## 1192           0.6505965
## 1193           0.6283333
## 1194           0.6256250
## 1195           0.6256250
## 1196           0.5058889
## 1197           0.3947222
```

```
f_df$department <- as.numeric(factor(f_df$department))
M = cor(f_df)
#"The below figure shows the correlation between the response and the explanatory attributes"
corrplot(M)
```




```
summary(f_df)
```

```
##     quarter         department          team        targeted_productivity
##  Min.   :1.000   Min.   :1.000   Min.   : 1.000   Min.   :0.0700
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.: 3.000   1st Qu.:0.7000
##  Median :2.000   Median :2.000   Median : 6.000   Median :0.7500
##  Mean   :2.414   Mean   :1.578   Mean   : 6.427   Mean   :0.7296
##  3rd Qu.:3.000   3rd Qu.:2.000   3rd Qu.: 9.000   3rd Qu.:0.8000
##  Max.   :5.000   Max.   :2.000   Max.   :12.000   Max.   :0.8000
```

```
##       smv              wip           over_time       incentive
##  Min.   : 2.90   Min.   :   0.0   Min.   :    0   Min.   :  0.00
##  1st Qu.: 3.94   1st Qu.:   0.0   1st Qu.: 1440   1st Qu.:  0.00
##  Median :15.26   Median : 588.0   Median : 4050   Median :  0.00
##  Mean   :15.10   Mean   : 580.5   Mean   : 4581   Mean   : 25.46
##  3rd Qu.:24.26   3rd Qu.:1082.0   3rd Qu.: 6960   3rd Qu.: 50.00
##  Max.   :54.56   Max.   :2698.0   Max.   :15120   Max.   :119.00
##    idle_time          idle_men       no_of_style_change no_of_workers
##  Min.   :  0.0000   Min.   : 0.0000   Min.   :0.0000     Min.   : 2.00
##  1st Qu.:  0.0000   1st Qu.: 0.0000   1st Qu.:0.0000     1st Qu.: 9.00
##  Median :  0.0000   Median : 0.0000   Median :0.0000     Median :34.00
##  Mean   :  0.7432   Mean   : 0.3759   Mean   :0.1531     Mean   :34.65
##  3rd Qu.:  0.0000   3rd Qu.: 0.0000   3rd Qu.:0.0000     3rd Qu.:57.00
##  Max.   :300.0000   Max.   :45.0000   Max.   :2.0000     Max.   :89.00
##  actual_productivity
##  Min.   :0.2337
##  1st Qu.:0.6502
##  Median :0.7691
##  Mean   :0.7343
##  3rd Qu.:0.8502
##  Max.   :1.1204
```

```r
#"Scaling of the data ,Training and Test set preparation"
set.seed(1)

sample <- sample(c(TRUE, FALSE), nrow(f_df), replace=TRUE, prob=c(0.7,0.3))
train  <- f_df[sample, ]
test   <- f_df[!sample, ]
dim(train)
```

```
## [1] 823  13
```

```r
dim(test)
```

```
## [1] 353  13
```

```r
drop <- c("actual_productivity")
x_train = train[,!(names(train) %in% drop)]
x_test =test[,!(names(test) %in% drop)]
y_train = train[,(names(train) %in% drop)]
y_test = test[,(names(test) %in% drop)]
dim(y_train)
```

```
## NULL
```

```r
x_s_train<- scale(x_train)
x_s_test<-scale(x_test)
actual_productivity<-c(y_train)
train_xy <- cbind(x_s_train, actual_productivity)
head(train_xy)
```

```
##     quarter department        team targeted_productivity        smv        wip
## 1 -1.148396  0.8661127  0.48326139             0.7088461  1.0049466  0.9182844
## 2 -1.148396 -1.1531812 -1.51248690             0.1960293 -1.0002754 -1.0044825
## 3 -1.148396  0.8661127  1.33858208             0.7088461 -0.3261526  0.6753355
## 5 -1.148396  0.8661127 -0.08695241             0.7088461  0.9814832  1.0258760
## 8 -1.148396  0.8661127 -0.94227310             0.1960293  1.1782151  0.3751201
## 9 -1.148396  0.8661127 -1.22738000             0.1960293  0.4373118  0.2675284
##    over_time  incentive   idle_time   idle_men no_of_style_change no_of_workers
## 1  0.7734132  2.3259788 -0.04451087 -0.1019772          -0.346502     1.1107922
## 2 -1.0910540 -0.8443775 -0.04451087 -0.1019772          -0.346502    -1.1886964
## 3 -0.2684950  0.7731512 -0.04451087 -0.1019772          -0.346502    -0.1742161
## 5 -0.7985886  0.7731512 -0.04451087 -0.1019772          -0.346502     0.9755281
## 8  0.7185759  0.6113983 -0.04451087 -0.1019772          -0.346502     1.0431602
## 9  0.4443896  0.2555420 -0.04451087 -0.1019772          -0.346502     0.9304401
##   actual_productivity
## 1           0.9407254
## 2           0.8865000
## 3           0.8005705
## 5           0.8003819
## 8           0.7536835
## 9           0.7530975
```

```
dim(train_xy)
```

```
## [1] 823  13
```

```
typeof(train_xy)
```

```
## [1] "double"
```

```
train_xy <- as.data.frame(train_xy)
x_s_test <- as.data.frame(x_s_test)
```

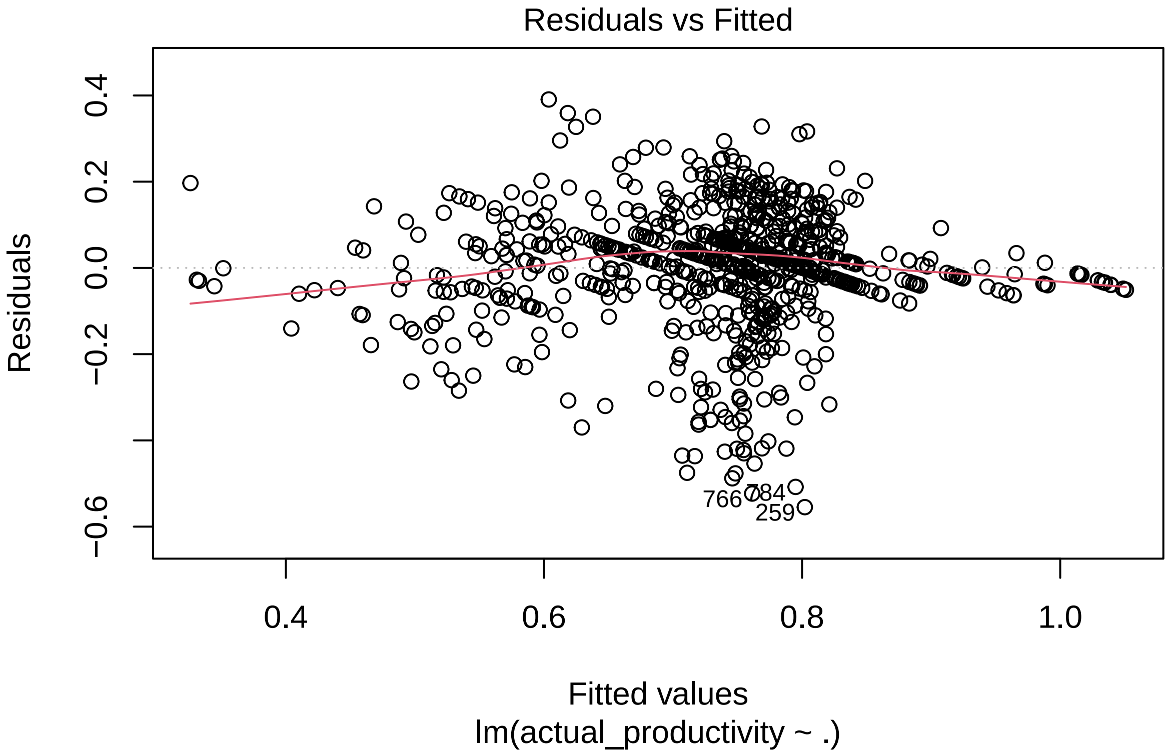
"METHODS"

```
#Linear Regression:
lr <- lm(actual_productivity ~ .,data=train_xy)
summary(lr)
```

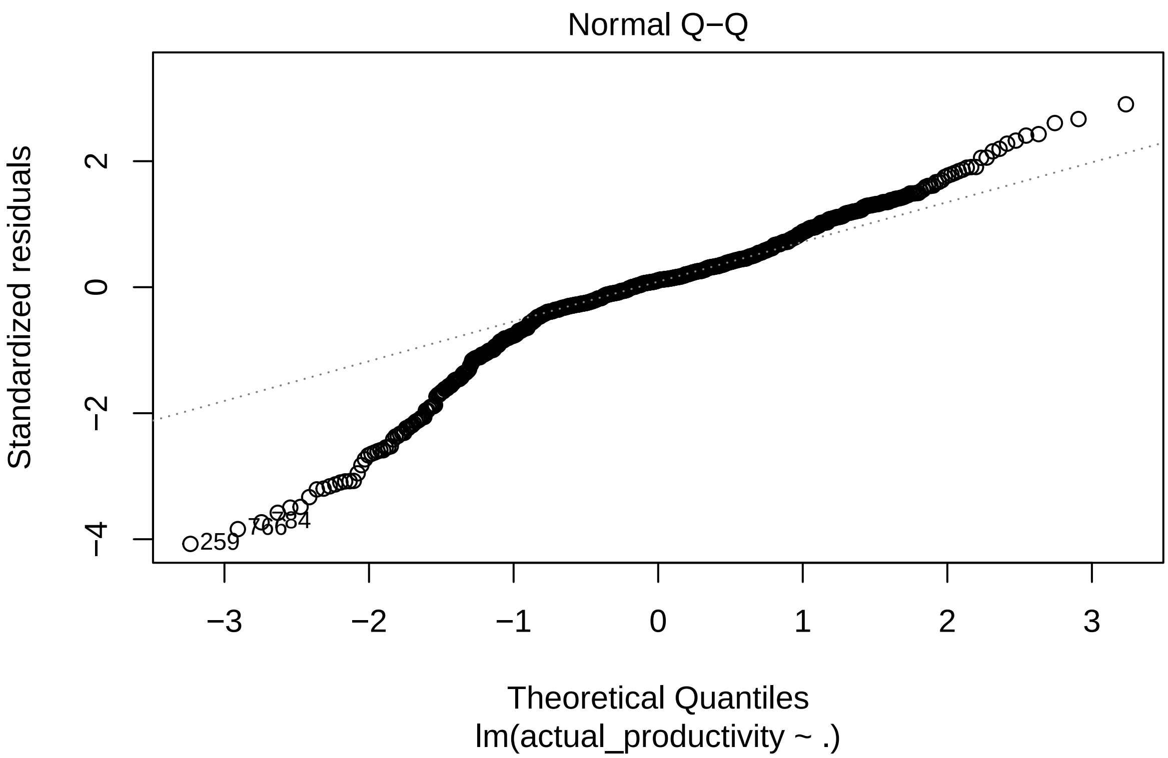
```
##
## Call:
## lm(formula = actual_productivity ~ ., data = train_xy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55479 -0.04486  0.01461  0.06996  0.39068
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
```

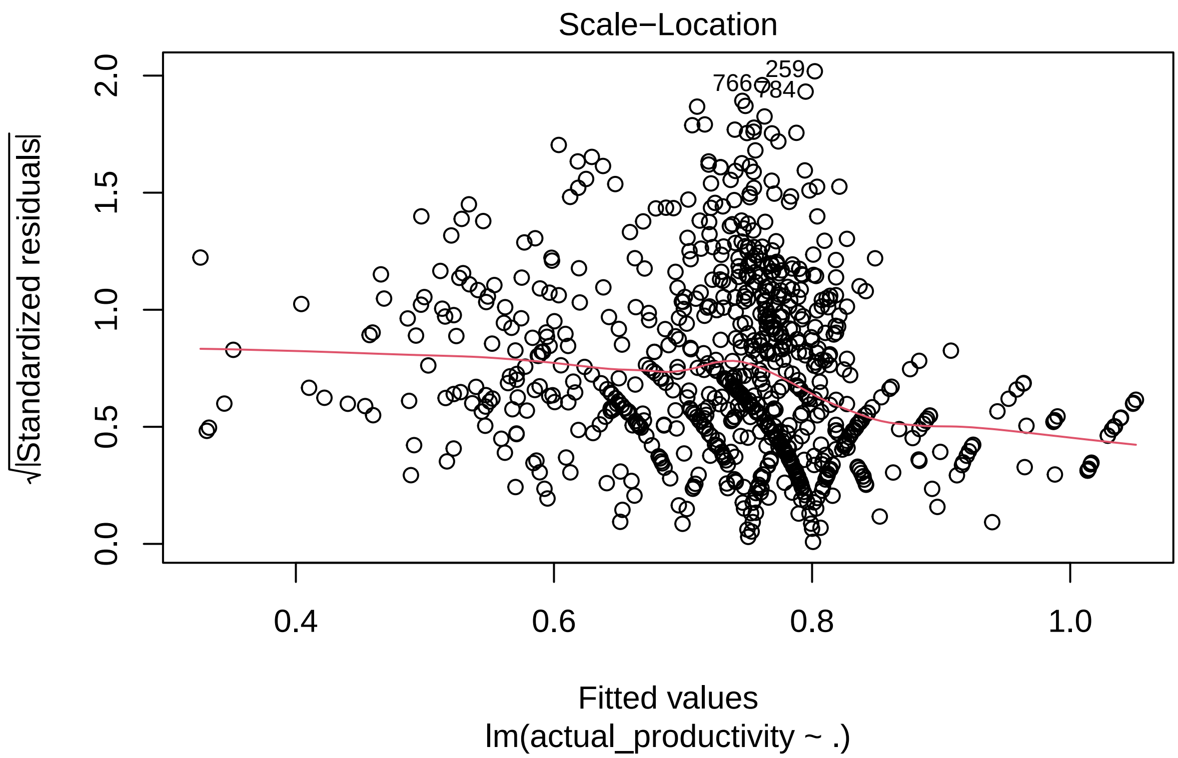
```
## (Intercept)            0.739393   0.004761 155.299  < 2e-16 ***
## quarter               -0.005987   0.004958  -1.208  0.22752    
## department            -0.088447   0.017889  -4.944 9.29e-07 ***
## team                  -0.020822   0.005141  -4.050 5.62e-05 ***
## targeted_productivity  0.037333   0.005402   6.910 9.79e-12 ***
## smv                   -0.048578   0.012001  -4.048 5.66e-05 ***
## wip                   -0.009127   0.010434  -0.875  0.38200    
## over_time             -0.008665   0.007783  -1.113  0.26593    
## incentive              0.106590   0.008735  12.202  < 2e-16 ***
## idle_time             -0.002931   0.005087  -0.576  0.56459    
## idle_men              -0.014356   0.005164  -2.780  0.00556 ** 
## no_of_style_change    -0.001902   0.005896  -0.323  0.74707    
## no_of_workers          0.060511   0.018883   3.204  0.00141 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1366 on 810 degrees of freedom
## Multiple R-squared:  0.3849, Adjusted R-squared:  0.3758 
## F-statistic: 42.24 on 12 and 810 DF,  p-value: < 2.2e-16
```

```r
lr_pred <- predict(lr,x_s_test)
#mse
lr_mse<-mean((y_test - lr_pred)^2)
#rmse
lr_rmse<-mean((y_test - lr_pred)^2)^(1/2)
#mae
lr_mae<-mae(y_test, lr_pred)
plot(lr)
```



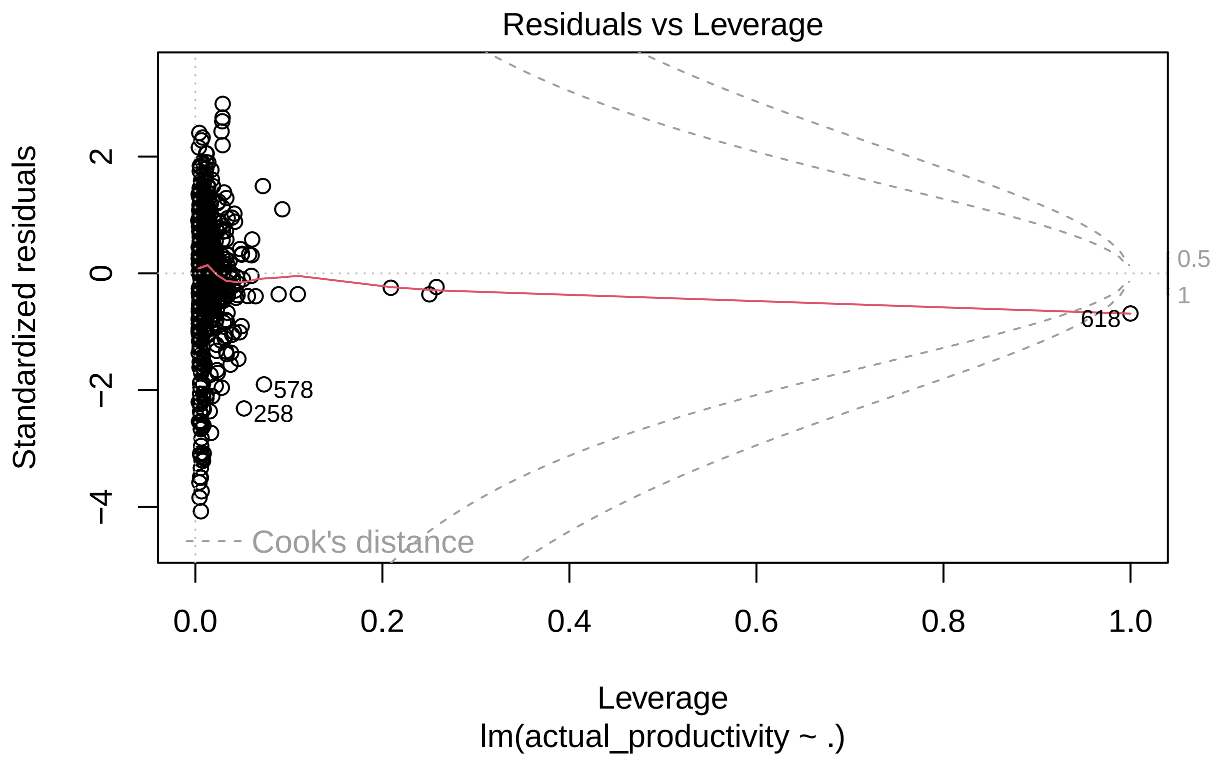
Residuals vs Fitted
Residuals
0.4
0.2
0.0
−0.2
−0.6
0.4
0.6
0.8
1.0
766
784
259
Fitted values
lm(actual_productivity ~ .)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(actual_productivity ~ .)
259
766
784




```
## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced

## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced
```




```r
lr_mae
```

```
## [1] 0.1028124
```

```r
#PCR:
library(pls)
```

```
##
## Attaching package: 'pls'

## The following object is masked from 'package:corrplot':
##
##     corrplot

## The following object is masked from 'package:stats':
##
##     loadings
```

```r
pcr_mse_tot=c()
pcr_rmse_tot=c()
pcr_mae_tot=c()
for (x in 1:11) {
  fit_pcr <- pcr(actual_productivity ~ .,ncomp = x,data=train_xy)
  summary(fit_pcr)
```

```r
  pcr_pred <- predict(fit_pcr,x_s_test)
  #mse
  pcr_mse<-mean((y_test - pcr_pred)^2)
  print(pcr_mse)
  pcr_mse_tot <- append(pcr_mse_tot,pcr_mse)
  #rmse
  pcr_rmse<-mean((y_test - pcr_pred)^2)^(1/2)
  print(pcr_rmse)
  pcr_rmse_tot <- append(pcr_rmse_tot,pcr_rmse)
  #mae
  pcr_mae<-mae(y_test, pcr_pred)
  print(pcr_mae)
  pcr_mae_tot <- append(pcr_mae_tot,pcr_mae)
}
```

```
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 1
## TRAINING: % variance explained
##                       1 comps
## X                    40.53065
## actual_productivity   0.07186
## [1] 0.0311385
## [1] 0.1764611
## [1] 0.1338422
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 2
## TRAINING: % variance explained
##                       1 comps  2 comps
## X                    40.53065    53.43
## actual_productivity   0.07186    19.51
## [1] 0.0283823
## [1] 0.1684705
## [1] 0.1246258
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 3
## TRAINING: % variance explained
##                       1 comps  2 comps  3 comps
## X                    40.53065    53.43    63.46
## actual_productivity   0.07186    19.51    21.25
## [1] 0.02737988
## [1] 0.1654687
## [1] 0.1217038
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 4
## TRAINING: % variance explained
```

```
##                      1 comps  2 comps  3 comps  4 comps
## X                   40.53065    53.43    63.46    72.07
## actual_productivity  0.07186    19.51    21.25    21.77
## [1] 0.02694372
## [1] 0.1641454
## [1] 0.1204552
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 5
## TRAINING: % variance explained
##                      1 comps  2 comps  3 comps  4 comps  5 comps
## X                   40.53065    53.43    63.46    72.07    79.50
## actual_productivity  0.07186    19.51    21.25    21.77    26.76
## [1] 0.02619705
## [1] 0.161855
## [1] 0.118313
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 6
## TRAINING: % variance explained
##                      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## X                   40.53065    53.43    63.46    72.07    79.50    86.68
## actual_productivity  0.07186    19.51    21.25    21.77    26.76    27.68
## [1] 0.02564601
## [1] 0.1601437
## [1] 0.116426
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 7
## TRAINING: % variance explained
##                      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## X                   40.53065    53.43    63.46    72.07    79.50    86.68
## actual_productivity  0.07186    19.51    21.25    21.77    26.76    27.68
##                     7 comps
## X                     91.89
## actual_productivity   28.46
## [1] 0.025164
## [1] 0.1586316
## [1] 0.1146914
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 8
## TRAINING: % variance explained
##                      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## X                   40.53065    53.43    63.46    72.07    79.50    86.68
## actual_productivity  0.07186    19.51    21.25    21.77    26.76    27.68
##                     7 comps  8 comps
## X                     91.89    95.37
## actual_productivity   28.46    31.55
## [1] 0.02469232
```

```
## [1] 0.1571379
## [1] 0.11349
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 9
## TRAINING: % variance explained
##                      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## X                   40.53065    53.43    63.46    72.07    79.50    86.68
## actual_productivity  0.07186    19.51    21.25    21.77    26.76    27.68
##                      7 comps  8 comps  9 comps
## X                      91.89    95.37    97.36
## actual_productivity    28.46    31.55    34.24
## [1] 0.02424599
## [1] 0.1557112
## [1] 0.112473
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 10
## TRAINING: % variance explained
##                      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## X                   40.53065    53.43    63.46    72.07    79.50    86.68
## actual_productivity  0.07186    19.51    21.25    21.77    26.76    27.68
##                      7 comps  8 comps  9 comps  10 comps
## X                      91.89    95.37    97.36     98.99
## actual_productivity    28.46    31.55    34.24     36.89
## [1] 0.0238949
## [1] 0.1545798
## [1] 0.111626
## Data:    X dimension: 823 12
##  Y dimension: 823 1
## Fit method: svdpc
## Number of components considered: 11
## TRAINING: % variance explained
##                      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## X                   40.53065    53.43    63.46    72.07    79.50    86.68
## actual_productivity  0.07186    19.51    21.25    21.77    26.76    27.68
##                      7 comps  8 comps  9 comps  10 comps  11 comps
## X                      91.89    95.37    97.36     98.99     99.64
## actual_productivity    28.46    31.55    34.24     36.89     36.92
## [1] 0.02361015
## [1] 0.1536559
## [1] 0.1109319
```

```
pcr_i=which.min(pcr_mae_tot)
pcr_mae_tot[pcr_i]
```

```
## [1] 0.1109319
```

```
cat('The errors for PCR is minimum at n =',pcr_i,'components.')
```

```
## The errors for PCR is minimum at n = 11 components.
```

```r
#Ridge Regression:
library(glmnet)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

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

## Loaded glmnet 4.1-4
```

```r
library(glmnetUtils)
```

```
##
## Attaching package: 'glmnetUtils'

## The following objects are masked from 'package:glmnet':
##
##     cv.glmnet, glmnet
```

```r
set.seed(42) # set seed for cross validation
#glmnet uses cross validation to select the optimal lambda values
cv_ridge <- cv.glmnet(actual_productivity ~ ., data = train_xy, alpha = 0)
lambda_select_ridge <- cv_ridge$lambda.1se  # tuned lambda
lambda_select_ridge
```

```
## [1] 0.01105035
```

```r
fit_ridge_select <- glmnet(
  actual_productivity ~ ., data = train_xy,
  alpha = 0,
  lambda = lambda_select_ridge
)
summary(fit_ridge_select)
```

```
##                  Length Class     Mode
## a0                1     -none-    numeric
## beta             12     dgCMatrix S4
## df                1     -none-    numeric
## dim               2     -none-    numeric
## lambda            1     -none-    numeric
## dev.ratio         1     -none-    numeric
## nulldev           1     -none-    numeric
## npasses           1     -none-    numeric
## jerr              1     -none-    numeric
## offset            1     -none-    logical
## call              5     -none-    call
```

```
## nobs              1      -none-    numeric
## terms             2      -none-    call
## xlev             12      -none-    list
## alpha             1      -none-    numeric
## sparse            1      -none-    logical
## use.model.frame   1      -none-    logical
## na.action         1      -none-    character
```

```r
fit_ridge_select
```

```
## Call:
## glmnetUtils:::glmnet.formula(formula = actual_productivity ~
##     ., data = train_xy, alpha = 0, lambda = lambda_select_ridge)
##
## Model fitting options:
##     Sparse model matrix: FALSE
##     Use model.frame: FALSE
##     Alpha: 0
##     Lambda summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.01105 0.01105 0.01105 0.01105 0.01105 0.01105
```

```r
ridge_pred <- predict(fit_ridge_select,x_s_test)
summary(ridge_pred)
```

```
##        s0
##  Min.   :0.4226
##  1st Qu.:0.6928
##  Median :0.7528
##  Mean   :0.7394
##  3rd Qu.:0.7935
##  Max.   :1.0351
```

```r
#mse
ridge_mse<-mean((y_test - ridge_pred)^2)
#rmse
ridge_rmse<-mean((y_test - ridge_pred)^2)^(1/2)
#mae
ridge_mae<-mae(y_test, ridge_pred)
#Lasso Regression
cv_lasso <-  cv.glmnet(actual_productivity ~ ., data = train_xy, alpha = 1)
lambda_select_lasso <- cv_lasso$lambda.1se  # tuned lambda
lambda_select_lasso
```

```
## [1] 0.01079631
```

```r
fit_lasso_select <- glmnet(
  actual_productivity ~ ., data = train_xy,
  alpha = 1,
  lambda = lambda_select_lasso
)
summary(fit_lasso_select)
```

```
##                 Length Class     Mode
## a0               1     -none-    numeric
## beta            12     dgCMatrix S4
## df               1     -none-    numeric
## dim              2     -none-    numeric
## lambda           1     -none-    numeric
## dev.ratio        1     -none-    numeric
## nulldev          1     -none-    numeric
## npasses          1     -none-    numeric
## jerr             1     -none-    numeric
## offset           1     -none-    logical
## call             5     -none-    call
## nobs             1     -none-    numeric
## terms            2     -none-    call
## xlev            12     -none-    list
## alpha            1     -none-    numeric
## sparse           1     -none-    logical
## use.model.frame  1     -none-    logical
## na.action        1     -none-    character
```

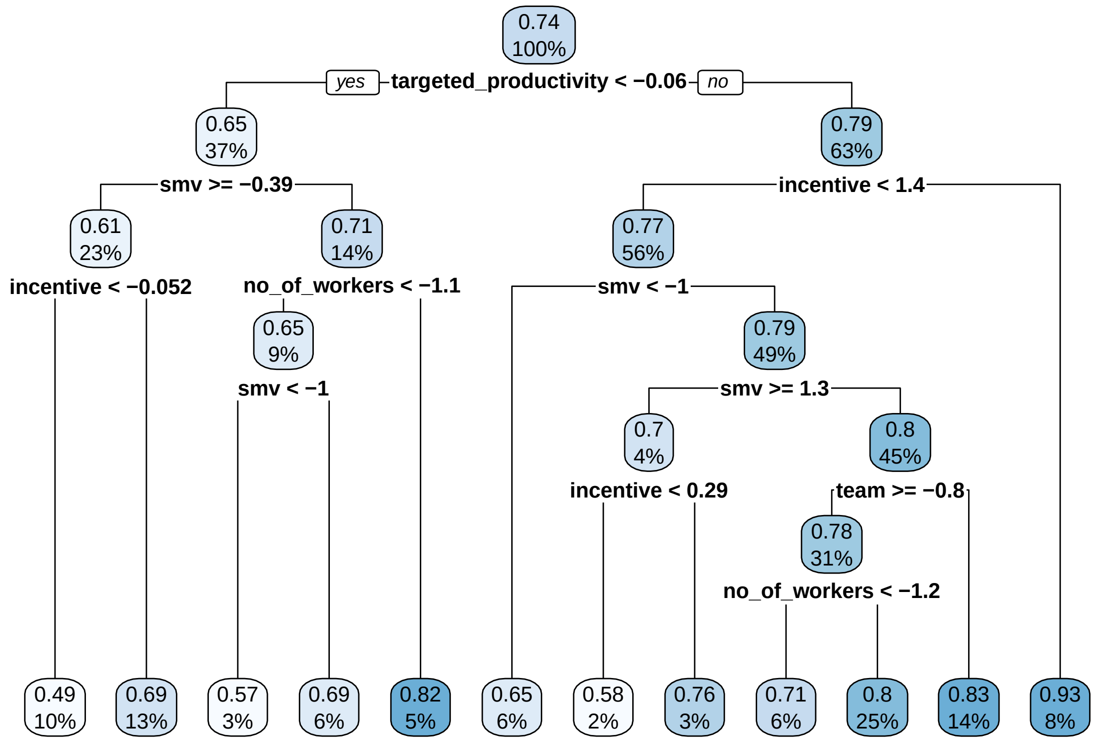
```r
lasso_pred <- predict(fit_lasso_select,x_s_test)
summary(lasso_pred)
```

```
##        s0
##  Min.   :0.4670
##  1st Qu.:0.6995
##  Median :0.7528
##  Mean   :0.7394
##  3rd Qu.:0.7850
##  Max.   :0.9796
```

```r
#mse
lasso_mse<-mean((y_test - lasso_pred)^2)
#rmse
lasso_rmse<-mean((y_test - lasso_pred)^2)^(1/2)
#mae
lasso_mae<-mae(y_test, lasso_pred)
#Regression Decision Trees
library(rpart)
library(rpart.plot)
```

```
## Warning: package 'rpart.plot' was built under R version 4.2.2
```

```r
set.seed(1)
tree <- rpart(actual_productivity ~ ., data = train_xy)
rpart.plot(tree)
```




```
summary(tree)
```

```
## Call:
## rpart(formula = actual_productivity ~ ., data = train_xy)
##   n= 823
##
##            CP nsplit rel error    xerror       xstd
## 1  0.16740085      0 1.0000000 1.0022743 0.05366311
## 2  0.05418653      1 0.8325991 0.8376889 0.04970918
## 3  0.05267721      3 0.7242261 0.7559758 0.04779840
## 4  0.03277789      4 0.6715489 0.7128306 0.04763639
## 5  0.02940076      5 0.6387710 0.6958019 0.04792505
## 6  0.01393881      6 0.6093702 0.6445767 0.04546926
## 7  0.01071585      7 0.5954314 0.6679720 0.04733166
## 8  0.01068614      8 0.5847156 0.6916517 0.05038502
## 9  0.01013002     10 0.5633433 0.7005679 0.05203827
## 10 0.01000000     11 0.5532133 0.6976985 0.05196447
##
## Variable importance
## targeted_productivity             incentive                   smv
##                    26                    20                    13
##         no_of_workers                   wip             over_time
##                    12                     9                     7
##    no_of_style_change            department                  team
##                     5                     4                     1
```

```
##            idle_time              idle_men
##                    1                     1
##
## Node number 1: 823 observations,    complexity param=0.1674009
##   mean=0.7393934, MSE=0.02985247
##   left son=2 (301 obs) right son=3 (522 obs)
##   Primary splits:
##       targeted_productivity < -0.06037903 to the left,  improve=0.16740090, (0 missing)
##       incentive             < 0.7569759   to the left,  improve=0.10855130, (0 missing)
##       no_of_style_change    < 0.8721781   to the right, improve=0.06291931, (0 missing)
##       team                  < -0.5146128  to the right, improve=0.05892859, (0 missing)
##       smv                   < 1.328923    to the right, improve=0.03493876, (0 missing)
##   Surrogate splits:
##       no_of_style_change < 0.8721781   to the right, agree=0.690, adj=0.153, (0 split)
##       wip                < 1.504832    to the right, agree=0.649, adj=0.040, (0 split)
##       smv                < 1.370886    to the right, agree=0.644, adj=0.027, (0 split)
##       idle_time          < 1.049705    to the right, agree=0.639, adj=0.013, (0 split)
##       idle_men           < 9.889357    to the right, agree=0.638, adj=0.010, (0 split)
##
## Node number 2: 301 observations,    complexity param=0.05418653
##   mean=0.6462996, MSE=0.03205896
##   left son=4 (186 obs) right son=5 (115 obs)
##   Primary splits:
##       smv                   < -0.3875185  to the right, improve=0.08424768, (0 missing)
##       department            < -0.1435343  to the right, improve=0.08333583, (0 missing)
##       wip                   < -0.9984088  to the right, improve=0.08333583, (0 missing)
##       no_of_workers         < -0.1967601  to the right, improve=0.08333583, (0 missing)
##       targeted_productivity < -1.086013   to the left,  improve=0.06396664, (0 missing)
##   Surrogate splits:
##       department    < -0.1435343  to the right, agree=0.997, adj=0.991, (0 split)
##       wip           < -0.9984088  to the right, agree=0.997, adj=0.991, (0 split)
##       no_of_workers < -0.1967601  to the right, agree=0.997, adj=0.991, (0 split)
##       over_time     < -0.3553206  to the right, agree=0.890, adj=0.713, (0 split)
##       incentive     < -0.5046965  to the right, agree=0.824, adj=0.539, (0 split)
##
## Node number 3: 522 observations,    complexity param=0.05267721
##   mean=0.7930739, MSE=0.02070121
##   left son=6 (457 obs) right son=7 (65 obs)
##   Primary splits:
##       incentive     < 1.403987    to the left,  improve=0.11976690, (0 missing)
##       smv           < -1.047202   to the left,  improve=0.09822315, (0 missing)
##       no_of_workers < -1.166152   to the left,  improve=0.08796496, (0 missing)
##       team          < -0.7997197  to the right, improve=0.08507580, (0 missing)
##       wip           < 0.881842    to the left,  improve=0.05531128, (0 missing)
##   Surrogate splits:
##       wip       < 1.39377     to the left,  agree=0.908, adj=0.262, (0 split)
##       over_time < 1.83817     to the left,  agree=0.887, adj=0.092, (0 split)
##       quarter   < 1.73059     to the left,  agree=0.877, adj=0.015, (0 split)
##
## Node number 4: 186 observations,    complexity param=0.05418653
##   mean=0.6054351, MSE=0.02085149
##   left son=8 (83 obs) right son=9 (103 obs)
##   Primary splits:
##       incentive             < -0.05178843 to the left,  improve=0.47690130, (0 missing)
```

```
##      targeted_productivity < -1.855238   to the left,  improve=0.29020950, (0 missing)
##      no_of_workers         < 1.009344    to the left,  improve=0.13805830, (0 missing)
##      team                  < -0.5146128  to the right, improve=0.05637498, (0 missing)
##      over_time             < 0.6728782   to the left,  improve=0.04042870, (0 missing)
##  Surrogate splits:
##      targeted_productivity < -1.086013   to the left,  agree=0.747, adj=0.434, (0 split)
##      no_of_workers         < 0.9642561   to the left,  agree=0.651, adj=0.217, (0 split)
##      over_time             < 0.6728782   to the left,  agree=0.634, adj=0.181, (0 split)
##      wip                   < 0.5061389   to the left,  agree=0.629, adj=0.169, (0 split)
##      no_of_style_change    < 0.8721781   to the right, agree=0.597, adj=0.096, (0 split)
##
## Node number 5: 115 observations,    complexity param=0.02940076
##   mean=0.7123934, MSE=0.04311654
##   left son=10 (73 obs) right son=11 (42 obs)
##   Primary splits:
##      no_of_workers         < -1.121064   to the left,  improve=0.14567900, (0 missing)
##      team                  < 0.9109217   to the right, improve=0.08137949, (0 missing)
##      over_time             < -1.072775   to the left,  improve=0.07389752, (0 missing)
##      smv                   < -1.00208    to the left,  improve=0.06079438, (0 missing)
##      targeted_productivity < -3.137279   to the right, improve=0.04638197, (0 missing)
##  Surrogate splits:
##      over_time             < -1.036217   to the left,  agree=0.757, adj=0.333, (0 split)
##      targeted_productivity < -3.137279   to the right, agree=0.678, adj=0.119, (0 split)
##
## Node number 6: 457 observations,    complexity param=0.03277789
##   mean=0.7742953, MSE=0.02031312
##   left son=12 (50 obs) right son=13 (407 obs)
##   Primary splits:
##      smv           < -1.047202   to the left,  improve=0.08674975, (0 missing)
##      no_of_workers < -1.166152   to the left,  improve=0.06115312, (0 missing)
##      team          < -0.7997197  to the right, improve=0.04626423, (0 missing)
##      over_time     < -1.072775   to the left,  improve=0.04508711, (0 missing)
##      incentive     < 0.7569759   to the left,  improve=0.01632616, (0 missing)
##
## Node number 7: 65 observations
##   mean=0.9251024, MSE=0.003518912
##
## Node number 8: 83 observations
##   mean=0.4943483, MSE=0.01788446
##
## Node number 9: 103 observations
##   mean=0.6949516, MSE=0.005285088
##
## Node number 10: 73 observations,    complexity param=0.01013002
##   mean=0.6522783, MSE=0.03820908
##   left son=20 (25 obs) right son=21 (48 obs)
##   Primary splits:
##      smv                   < -1.00208    to the left,  improve=0.089227930, (0 missing)
##      team                  < -0.2295059  to the right, improve=0.068379740, (0 missing)
##      targeted_productivity < -0.5731958  to the right, improve=0.021332800, (0 missing)
##      over_time             < -0.4604254  to the right, improve=0.015511070, (0 missing)
##      quarter               < 0.9080223   to the right, improve=0.000763178, (0 missing)
##
## Node number 11: 42 observations
```

```
##   mean=0.8168792, MSE=0.03444773
##
## Node number 12: 50 observations
##   mean=0.654529, MSE=0.02700604
##
## Node number 13: 407 observations,    complexity param=0.01393881
##   mean=0.7890086, MSE=0.01751226
##   left son=26 (36 obs) right son=27 (371 obs)
##   Primary splits:
##       smv           < 1.328923    to the right, improve=0.04804731, (0 missing)
##       team          < -0.7997197  to the right, improve=0.03183012, (0 missing)
##       no_of_workers < 1.009344    to the right, improve=0.02420620, (0 missing)
##       department    < -0.1435343  to the right, improve=0.02314659, (0 missing)
##       wip           < -0.9958058  to the right, improve=0.02314659, (0 missing)
##   Surrogate splits:
##       no_of_style_change < 0.8721781   to the right, agree=0.931, adj=0.222, (0 split)
##       no_of_workers      < 1.09952     to the right, agree=0.931, adj=0.222, (0 split)
##       idle_time          < 0.2885114   to the right, agree=0.916, adj=0.056, (0 split)
##       idle_men           < 2.895423    to the right, agree=0.916, adj=0.056, (0 split)
##
## Node number 20: 25 observations
##   mean=0.5713717, MSE=0.034653
##
## Node number 21: 48 observations
##   mean=0.6944172, MSE=0.0348762
##
## Node number 26: 36 observations,    complexity param=0.01071585
##   mean=0.6958889, MSE=0.01736311
##   left son=52 (13 obs) right son=53 (23 obs)
##   Primary splits:
##       incentive          < 0.2878926   to the left,  improve=0.42118860, (0 missing)
##       smv                < 2.779595    to the right, improve=0.16876830, (0 missing)
##       no_of_workers      < 1.076976    to the left,  improve=0.15153730, (0 missing)
##       wip                < 0.5868327   to the right, improve=0.11995860, (0 missing)
##       no_of_style_change < 0.8721781   to the left,  improve=0.02489631, (0 missing)
##   Surrogate splits:
##       idle_time < 0.2885114   to the right, agree=0.722, adj=0.231, (0 split)
##       idle_men  < 2.895423    to the right, agree=0.722, adj=0.231, (0 split)
##       smv       < 3.044461    to the right, agree=0.694, adj=0.154, (0 split)
##       wip       < 0.5686115   to the right, agree=0.694, adj=0.154, (0 split)
##       team      < -1.084827   to the left,  agree=0.667, adj=0.077, (0 split)
##
## Node number 27: 371 observations,    complexity param=0.01068614
##   mean=0.7980444, MSE=0.01660366
##   left son=54 (254 obs) right son=55 (117 obs)
##   Primary splits:
##       team          < -0.7997197  to the right, improve=0.035327750, (0 missing)
##       smv           < -0.9844827  to the right, improve=0.029476240, (0 missing)
##       over_time     < -1.072775   to the left,  improve=0.026767250, (0 missing)
##       no_of_workers < -1.166152   to the left,  improve=0.022544070, (0 missing)
##       department    < -0.1435343  to the right, improve=0.009908916, (0 missing)
##   Surrogate splits:
##       smv < 1.034727    to the left,  agree=0.698, adj=0.043, (0 split)
##       wip < 1.538672    to the left,  agree=0.687, adj=0.009, (0 split)
```

```
##
## Node number 52: 13 observations
##   mean=0.5821408, MSE=0.02169964
##
## Node number 53: 23 observations
##   mean=0.7601814, MSE=0.003465369
##
## Node number 54: 254 observations,    complexity param=0.01068614
##   mean=0.7816069, MSE=0.01574758
##   left son=108 (49 obs) right son=109 (205 obs)
##   Primary splits:
##       no_of_workers         < -1.166152   to the left,  improve=0.07686945, (0 missing)
##       over_time             < -1.072775   to the left,  improve=0.04394706, (0 missing)
##       incentive             < 0.6922748   to the left,  improve=0.01825998, (0 missing)
##       wip                   < 0.6449669   to the left,  improve=0.01503595, (0 missing)
##       targeted_productivity < 0.4524377   to the left,  improve=0.01148343, (0 missing)
##   Surrogate splits:
##       over_time < -1.072775   to the left,  agree=0.882, adj=0.388, (0 split)
##       smv       < -0.9939583  to the left,  agree=0.866, adj=0.306, (0 split)
##
## Node number 55: 117 observations
##   mean=0.8337293, MSE=0.0166022
##
## Node number 108: 49 observations
##   mean=0.7104426, MSE=0.03928649
##
## Node number 109: 205 observations
##   mean=0.798617, MSE=0.008621353
```

```
tree_pred <- predict(tree,x_s_test)
#mse
tree_mse<-mean((y_test - tree_pred)^2)
#rmse
tree_rmse<-mean((y_test - tree_pred)^2)^(1/2)
#mae
tree_mae<-mae(y_test, tree_pred)
#Random Forest Regression
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.2.2
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:imager':
##
##     grow
```

```
## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
set.seed(1)
rf<- randomForest(
  actual_productivity ~ ., data = train_xy,
  mtry=14, importance=TRUE, ntree = 1000
)
```

```
## Warning in randomForest.default(m, y, ...): invalid mtry: reset to within valid
## range
```

```
summary(rf)
```

```
##                 Length Class  Mode
## call               6   -none- call
## type               1   -none- character
## predicted        823   -none- numeric
## mse             1000   -none- numeric
## rsq             1000   -none- numeric
## oob.times        823   -none- numeric
## importance        24   -none- numeric
## importanceSD      12   -none- numeric
## localImportance    0   -none- NULL
## proximity          0   -none- NULL
## ntree              1   -none- numeric
## mtry               1   -none- numeric
## forest            11   -none- list
## coefs              0   -none- NULL
## y                823   -none- numeric
## test               0   -none- NULL
## inbag              0   -none- NULL
## terms              3   terms  call
```

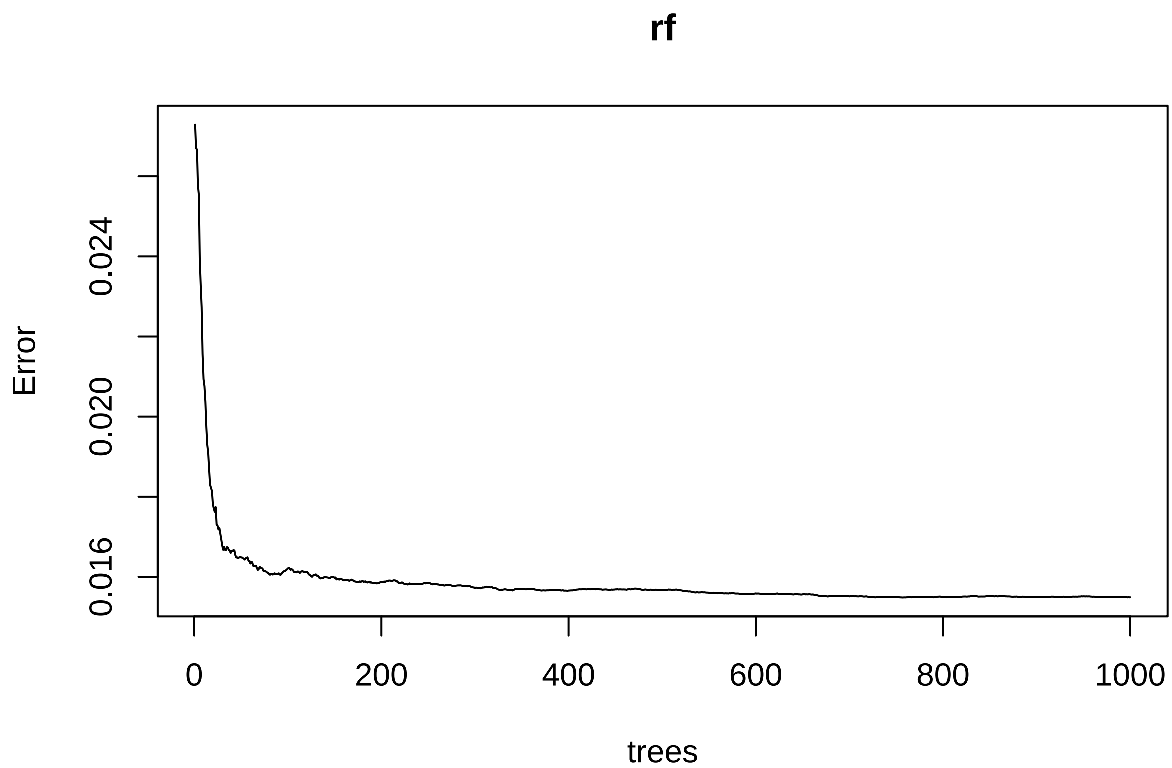
```
rf_pred <- predict(rf,x_s_test,type='response')
importance(rf)
```

```
##                         %IncMSE IncNodePurity
## quarter               33.709972     1.9332576
## department            11.537971     0.1109789
## team                  35.838784     2.3283823
## targeted_productivity 74.378932     5.5326729
## smv                   59.964077     3.3226844
## wip                   20.553286     0.5230746
## over_time             48.395981     2.7977969
## incentive             69.286847     3.2920566
```

```
## idle_time                 9.785162     0.1075025
## idle_men                  9.526176     0.1021611
## no_of_style_change        7.982383     0.0850035
## no_of_workers            55.721863     2.3927375
```

```
varImpPlot(rf)
```


```
plot(rf)
```




```
#mse
rf_mse<-mean((y_test - rf_pred)^2)
#rmse
rf_rmse<-mean((y_test - rf_pred)^2)^(1/2)
#mae
rf_mae<-mae(y_test, rf_pred)
```

"RESULT"

```
result <- data.frame (Algorithms  = c('Regression',
                                      'Principal Component Regression',
                                      'Ridge Regression',
                                      'Lasso Regression',
                                      'Regression Decision Trees',
                                      'Random Forest Regression'),
                      MAE = c(lr_mae,pcr_mae_tot[pcr_i],ridge_mae,lasso_mae,tree_mae,
                              rf_mae),
                      MSE = c(lr_mse,pcr_mse_tot[pcr_i],ridge_mse,ridge_mse,lasso_mse,
                              rf_mse),
                      RMSE= c(lr_rmse,pcr_rmse_tot[pcr_i],ridge_rmse,lasso_rmse,tree_rmse,
                              rf_rmse)
)
```

```r
#result
kable(result)
```

Algorithms	MAE	MSE	RMSE
Regression	0.1028124	0.0201304	0.1418815
Principal Component Regression	0.1109319	0.0236101	0.1536559
Ridge Regression	0.1012024	0.0202466	0.1422905
Lasso Regression	0.1015938	0.0202466	0.1444028
Regression Decision Trees	0.1139552	0.0208522	0.1601065
Random Forest Regression	0.0914535	0.0184084	0.1356774

```r
result_reshaped <- data.frame(Algorithms = result$Algorithms,
                              Errors = c(result$MAE, result$MSE, result$RMSE),
                              group = c(rep("MAE", nrow(result)),
                                        rep("MSE", nrow(result)),
                                        rep("RMSE", nrow(result))))

ggplot(result_reshaped, aes(Algorithms, Errors, col = group))+
  geom_point()+
  theme(axis.text.x = element_text(angle = 90))
```