Visual Data Science

Model Education Data

Markus Kiesel | 1228952

18.01.2022

Introduction

The education dataset used for this project is highly fragmented. For most indicators we have data by education level and gender which makes the features highly correlated as well. Further, many values are forward filled which has the effect that values over years are very similar. For these reasons we decided to only use the total values and only evaluate which features give us the most information. This is done by scaling and centering all features expect the target value and using logistic regression to shrink uninformative or highly correlated features to zero. The indicators we are interested in are the learning outcome, completion rate and the literacy rate.

First we load the data for all total indicators and preprocess it.

```
# load data
df <- read.csv("./data/data_total.csv")
# drop country code and name and region
df <- subset(df, select=-c(country_code, country_name, region))
# income group to factors
df <- transform(df, income_group = as.factor(income_group))
# drop rows wit null values
df <- na.omit(df)
# one-hot encode income group
dummy <- dummyVars(" ~ .", data = df)
df <- data.frame(predict(dummy, newdata = df))
# have a look at structure
str(df)
```

```
## 'data.frame':    1768 obs. of  13 variables:
##  $ year                            : num  2000 2000 2000 2000 2000 2000 2000 2000 2000 2000 ...
##  $ compulsory_education_duration   : num  10 8 8 10 9 13 11 8 8 11 ...
##  $ education_expenditure_gdp_rate  : num  4.58 3.95 3.78 4.17 3.23 ...
##  $ literacy_rate                   : num  96 86.4 94.3 96.5 92.6 ...
##  $ completion_rate                 : num  40.4 11.2 20.6 21.3 42.6 ...
##  $ income_group.High.income        : num  0 0 1 1 1 1 1 1 0 1 ...
##  $ income_group.Low.income         : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ income_group.Lower.middle.income: num  0 0 0 0 0 0 0 0 1 0 ...
##  $ income_group.Upper.middle.income: num  1 1 0 0 0 0 0 0 0 0 ...
##  $ learning_outcome                : num  422 392 423 501 477 ...
##  $ gdppc                           : num  14369 9834 15212 26995 20965 ...
##  $ population                      : num  37336 176369 15285 41072 10735 ...
##  $ education_spent                 : num  658 388 574 1125 677 ...
```

```r
# summary
summary(df)
```

```
##       year      compulsory_education_duration education_expenditure_gdp_rate
##  Min.   :2000   Min.   : 5.000                Min.   : 0.8148
##  1st Qu.:2008   1st Qu.: 9.000                1st Qu.: 3.1366
##  Median :2013   Median : 9.000                Median : 4.0724
##  Mean   :2012   Mean   : 9.582                Mean   : 4.3102
##  3rd Qu.:2017   3rd Qu.:11.000                3rd Qu.: 5.1378
##  Max.   :2020   Max.   :17.000                Max.   :44.3340
##  literacy_rate   completion_rate   income_group.High.income
##  Min.   :19.04   Min.   : 0.8579   Min.   :0.0000
##  1st Qu.:76.64   1st Qu.:26.9487   1st Qu.:0.0000
##  Median :92.55   Median :45.9482   Median :0.0000
##  Mean   :84.01   Mean   :45.5696   Mean   :0.2568
##  3rd Qu.:97.89   3rd Qu.:63.1988   3rd Qu.:1.0000
##  Max.   :99.97   Max.   :99.6616   Max.   :1.0000
##  income_group.Low.income income_group.Lower.middle.income
##  Min.   :0.0000          Min.   :0.0000
##  1st Qu.:0.0000          1st Qu.:0.0000
##  Median :0.0000          Median :0.0000
##  Mean   :0.1205          Mean   :0.2919
##  3rd Qu.:0.0000          3rd Qu.:1.0000
##  Max.   :1.0000          Max.   :1.0000
##  income_group.Upper.middle.income learning_outcome     gdppc
##  Min.   :0.0000                   Min.   :226.6    Min.   :   515.3
##  1st Qu.:0.0000                   1st Qu.:361.2    1st Qu.:  4690.4
##  Median :0.0000                   Median :420.1    Median : 10975.9
##  Mean   :0.3309                   Mean   :416.3    Mean   : 14623.5
##  3rd Qu.:1.0000                   3rd Qu.:472.8    3rd Qu.: 19278.8
##  Max.   :1.0000                   Max.   :594.2    Max.   :156299.0
##    population      education_spent
##  Min.   :     79   Min.   :   7.285
##  1st Qu.:   4166   1st Qu.: 171.356
##  Median :  11855   Median : 420.808
##  Mean   :  47618   Mean   : 616.724
##  3rd Qu.:  38730   3rd Qu.: 844.463
##  Max.   :1385439   Max.   :6280.191
```

We need to scale and center the data to be able to compare the different coefficients afterwards.

```r
# function to scale and center all non character columns
scale_numbers <- function(data, from_train=TRUE, means=c(), sds=c()) {
  num_cols <- dim(data)[2]
  if (from_train) {
    means <- rep(0, num_cols)
    sds <- rep(0, num_cols)
  }
  for (col in 1:num_cols) {
    if (class(data[, col]) != "character") {
      if (from_train) {
        means[col] <- mean(data[, col])
        sds[col] <- sd(data[, col])
      }
      data[, col] <- (data[, col] - means[col]) / sds[col]
```

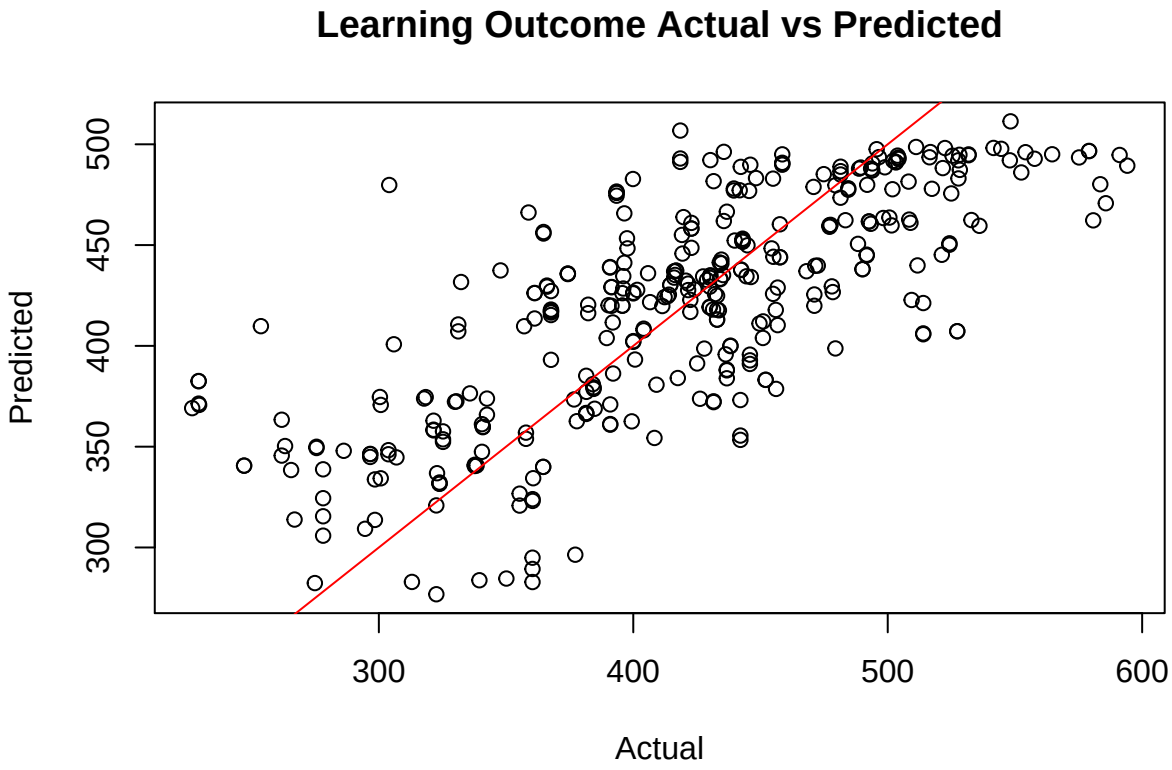
```
    }
  }
  return(list(data=data, means=means, sds=sds))
}
```

To be able to evaluate our models we split the data into training and testing sets.

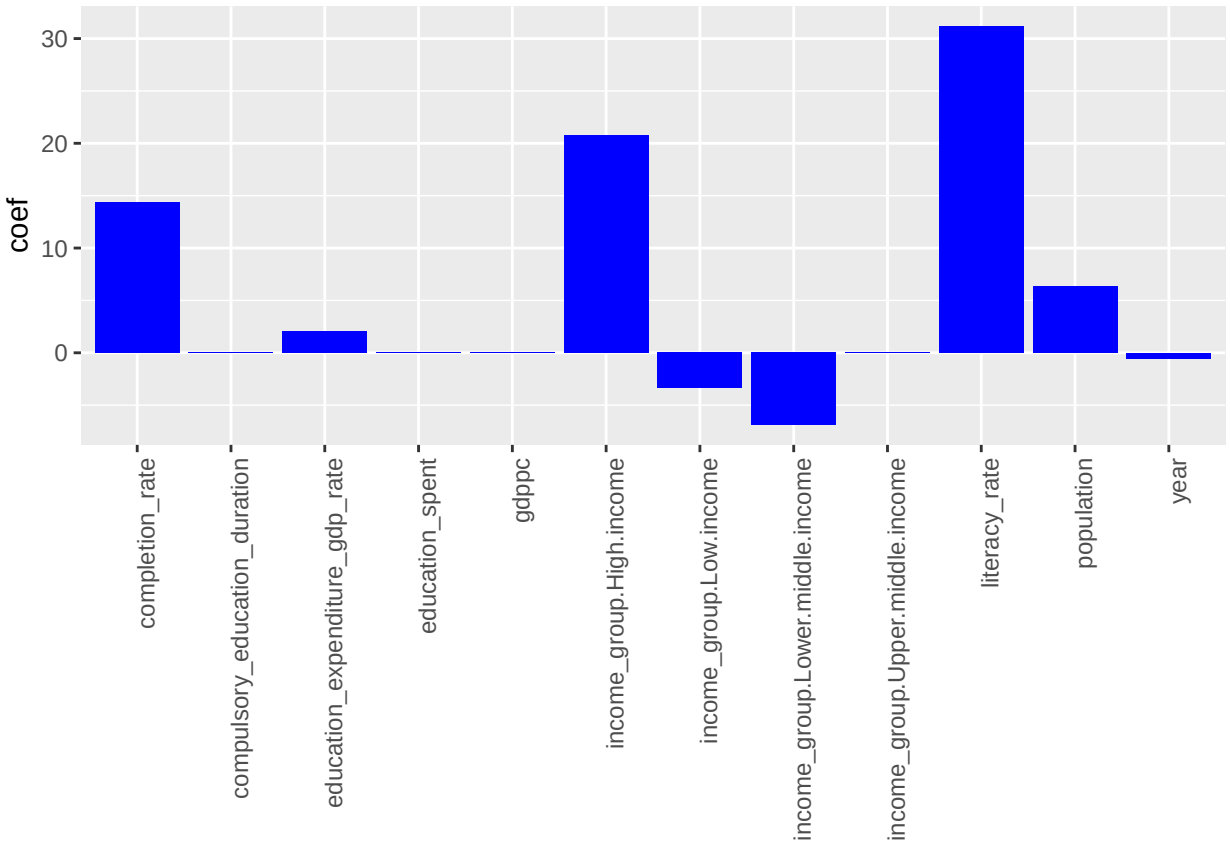
```
# Train-Test Split
set.seed(1228052)
n <- nrow(df)
train_index <- sample(1:n, n * 0.8)
test_index <- c(1:n)[-train_index]
```

Model Learning Outcome

```
# select indicator of interest
x_train <- subset(df, select=-c(learning_outcome))[train_index,]
y_train <- df[train_index, "learning_outcome"]
x_test <- subset(df, select=-c(learning_outcome))[test_index,]
y_test <- df[test_index, "learning_outcome"]
# scale and center training data
scaling <- scale_numbers(x_train)
x_train <- scaling$data
# use training data info to scale and center test data
x_test <- scale_numbers(x_test, FALSE, scaling$means, scaling$sds)$data
# model data
model_lasso <- cv.glmnet(x=as.matrix(x_train), y=y_train)
# predict
pred_lasso <- predict(model_lasso, newx=data.matrix(x_test), s="lambda.1se")
learning_outcome_rmse <- rmse(y_test, pred_lasso)

# plot actual vs predicted
plot(y_test, pred_lasso, main="Learning Outcome Actual vs Predicted", xlab="Actual", ylab="Predicted")
abline(c(0,1), col="red")
```




```
# plot coefficient impact
df_coeff <- data.frame(
  feature = dimnames(coef(model_lasso))[[1]][-1],
  coef = coef(model_lasso)[-1],
  row.names = NULL)

ggplot(df_coeff, aes(y=coef, x=feature)) +
  geom_bar(position="dodge", stat="identity", fill="blue") +
  theme(axis.title.x = element_blank(), axis.text.x = element_text(angle = 90, hjust = 1))
```



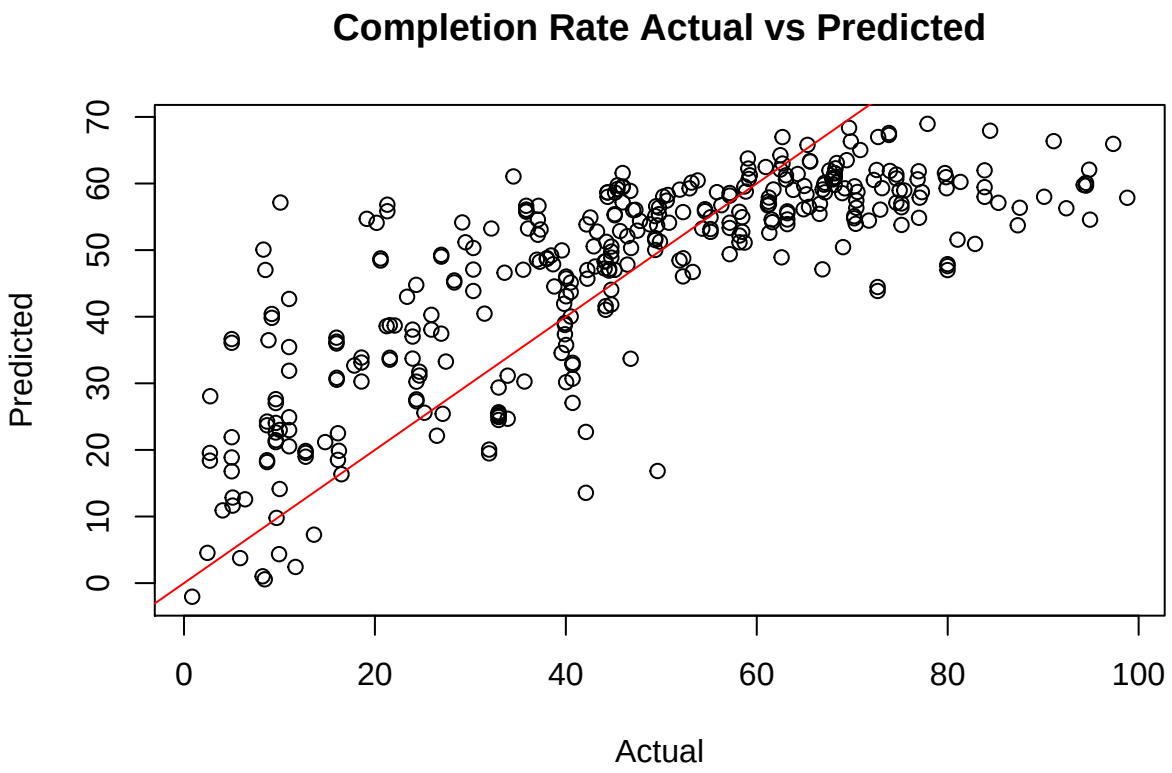
The model is not very good at predicting the learning outcome as we can see in the plot Actual vs Predicted.

We get information on which features are important to predict the learning outcome. Being in an high income country has a positive effect while being in a low or lower middle income country has a negative effect. Also the literacy rate has a high impact on the learning outcome. The gdpc and and the amount spent on education has surprysingly no impact but this my be shrunk because of the correlation with the income group. The year has almost no inpact which is in accordance that the learning outcome is rather stable over the span of 2000 to 2020.

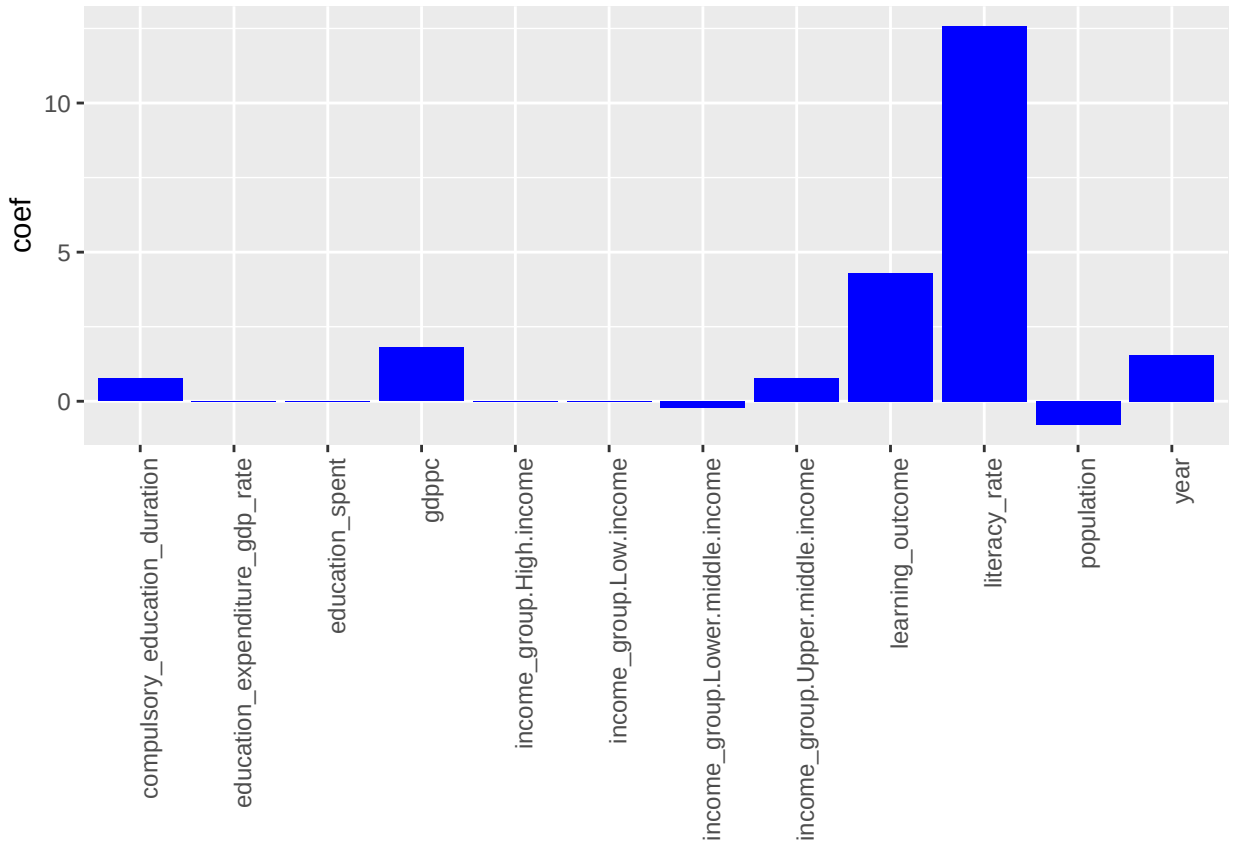
Model Completion Rate

```
# select indicator of interest
x_train <- subset(df, select=-c(completion_rate))[train_index,]
y_train <- df[train_index, "completion_rate"]
x_test <- subset(df, select=-c(completion_rate))[test_index,]
y_test <- df[test_index, "completion_rate"]
# scale and center training data
scaling <- scale_numbers(x_train)
x_train <- scaling$data
# use training data info to scale and center test data
x_test <- scale_numbers(x_test, FALSE, scaling$means, scaling$sds)$data
# model data
model_lasso <- cv.glmnet(x=as.matrix(x_train), y=y_train)
# predict
pred_lasso <- predict(model_lasso, newx=data.matrix(x_test), s="lambda.1se")
completion_rate_rmse <- rmse(y_test, pred_lasso)
```

```r
# plot actual vs predicted
plot(y_test, pred_lasso, main="Completion Rate Actual vs Predicted", xlab="Actual", ylab="Predicted")
abline(c(0,1), col="red")
```




```r
# plot coefficient impact
df_coeff <- data.frame(
  feature = dimnames(coef(model_lasso))[[1]][-1],
  coef = coef(model_lasso)[-1],
  row.names = NULL)

ggplot(df_coeff, aes(y=coef, x=feature)) +
  geom_bar(position="dodge", stat="identity", fill="blue") +
  theme(axis.title.x = element_blank(), axis.text.x = element_text(angle = 90, hjust = 1))
```



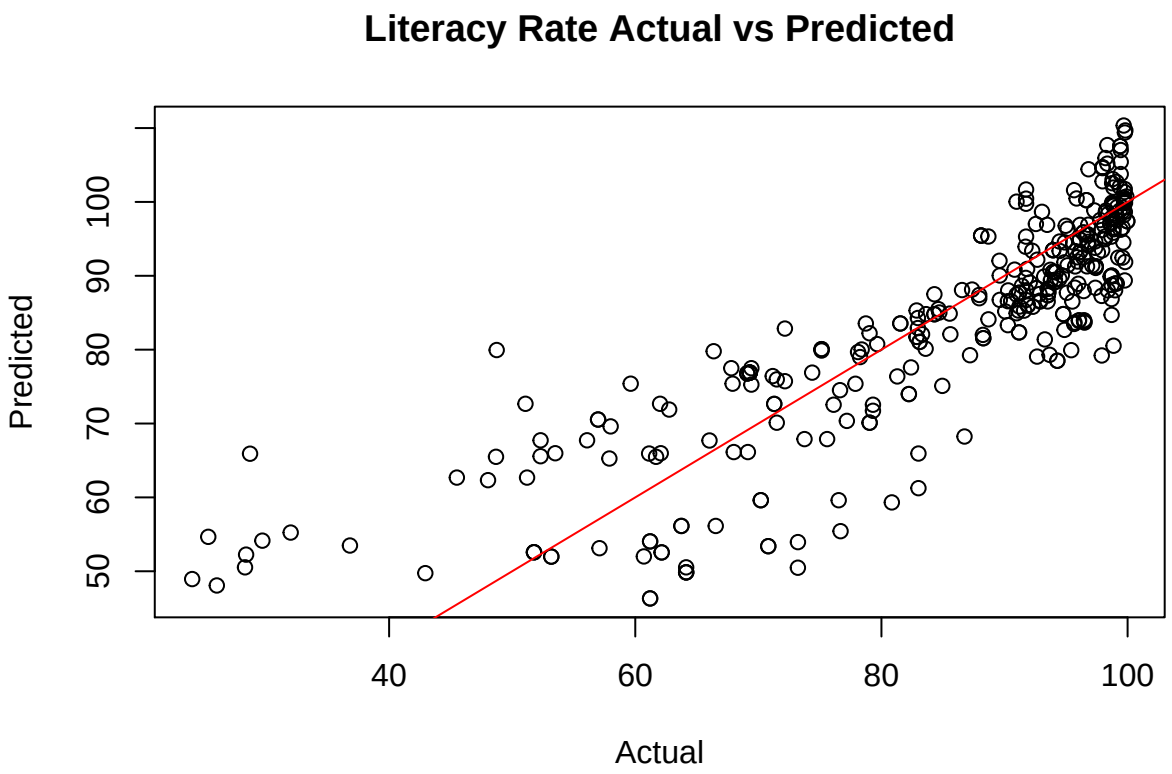
The underlying data seems to have some nonlinear relationshop as we can see from the Actual vs Predicted plot.

The highest impact for this model has the literacy rate followed by the learning outcome. Here the gdppc seems to play some role in the model while the income group has no impact. The impact of the year is probaply caused by the fact that the completion rate rises over the years for all income groups.

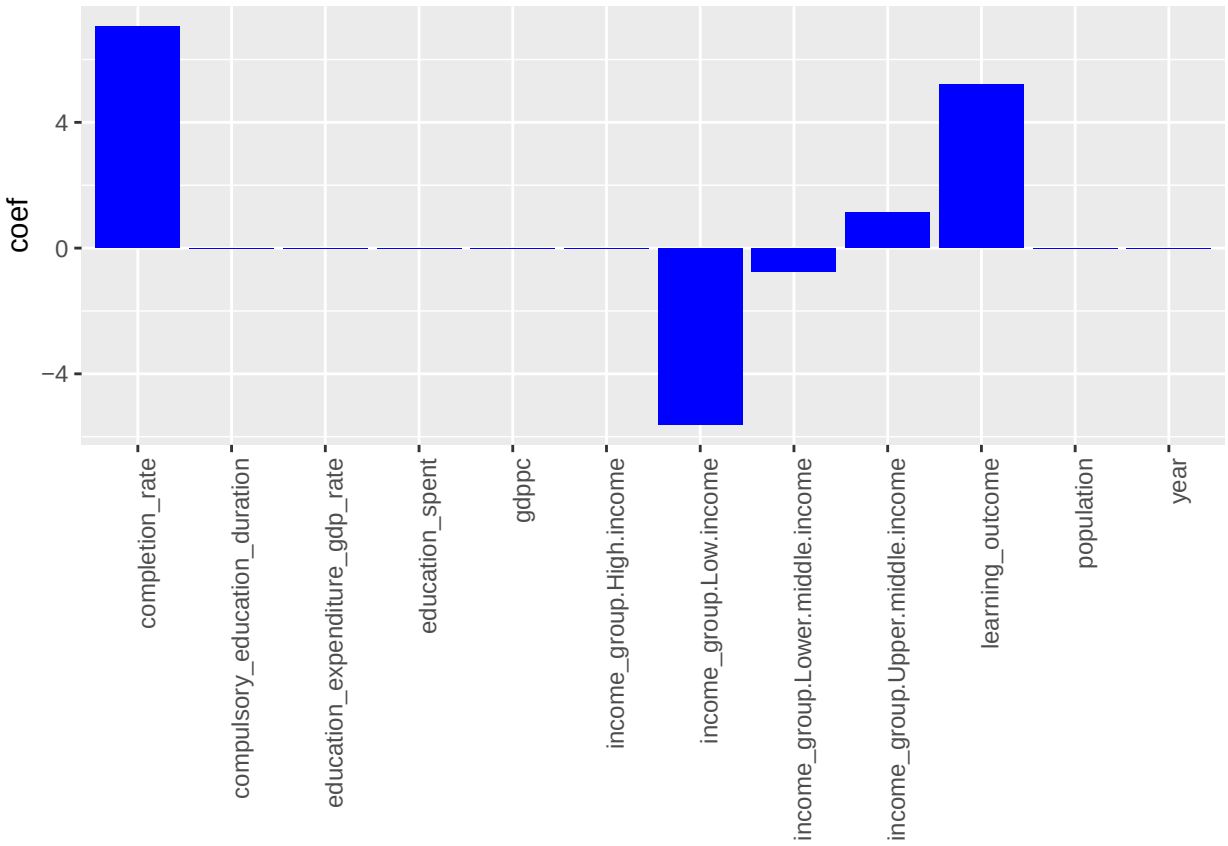
Model Literacy Rate

```
# select indicator of interest
x_train <- subset(df, select=-c(literacy_rate))[train_index,]
y_train <- df[train_index, "literacy_rate"]
x_test <- subset(df, select=-c(literacy_rate))[test_index,]
y_test <- df[test_index, "literacy_rate"]
# scale and center training data
scaling <- scale_numbers(x_train)
x_train <- scaling$data
# use training data info to scale and center test data
x_test <- scale_numbers(x_test, FALSE, scaling$means, scaling$sds)$data
# model data
model_lasso <- cv.glmnet(x=as.matrix(x_train), y=y_train)
# predict
pred_lasso <- predict(model_lasso, newx=data.matrix(x_test), s="lambda.1se")
literacy_rate_rmse <- rmse(y_test, pred_lasso)

# plot actual vs predicted
```

```r
plot(y_test, pred_lasso, main="Literacy Rate Actual vs Predicted", xlab="Actual", ylab="Predicted")
abline(c(0,1), col="red")
```




```r
# plot coefficient impact
df_coeff <- data.frame(
  feature = dimnames(coef(model_lasso))[[1]][-1],
  coef = coef(model_lasso)[-1],
  row.names = NULL)

ggplot(df_coeff, aes(y=coef, x=feature)) +
  geom_bar(position="dodge", stat="identity", fill="blue") +
  theme(axis.title.x = element_blank(), axis.text.x = element_text(angle = 90, hjust = 1))
```

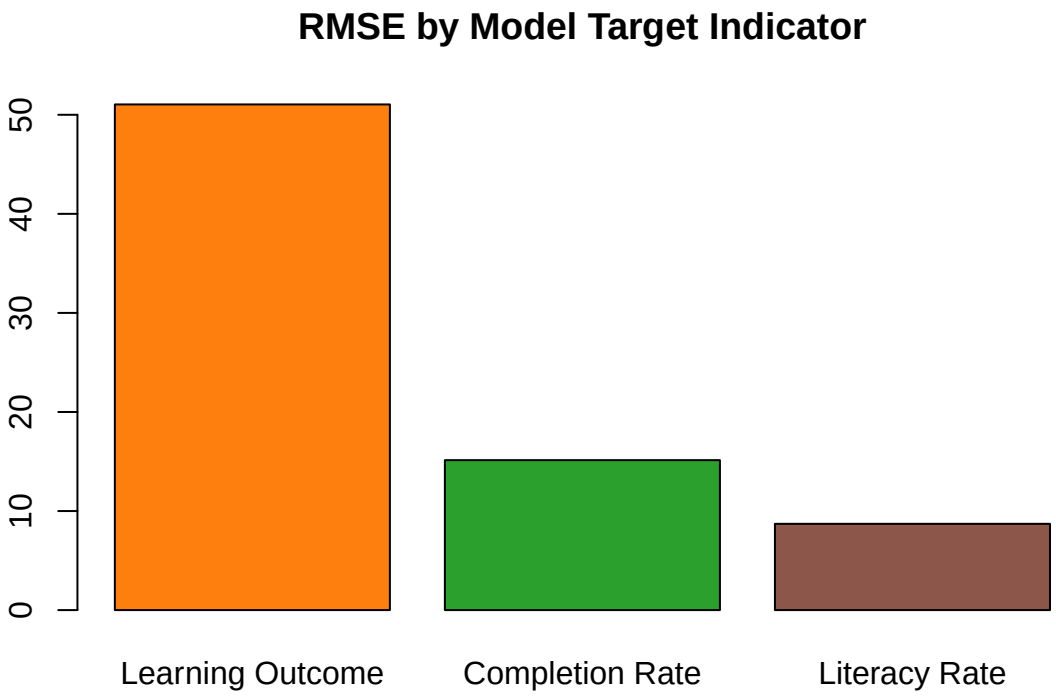


For the literacy rate we see some nonlinear relationship again but the model is the best of the three.

Here the other two target features leraning outcome and completion rate have a high impact. Also the low income group has a high negative impact while the high income group has none. Most features have no impact on the literacy rate. Because the literacy rate rose over the last 20 years we expected some importance of the year.

Conclusion

```
rmse_values <- c(learning_outcome_rmse, completion_rate_rmse, literacy_rate_rmse)
names(rmse_values) <- c("Learning Outcome", "Completion Rate", "Literacy Rate")
barplot(
  rmse_values,
  col=c("#ff7f0e", "#2ca02c", "#8c564b"),
  main="RMSE by Model Target Indicator")
```



The above plot shows that all models do not perform very well. The models for completion rate is on average wrong by 15% for literacy rate 8% and for the learning outcome by 50 points.

Nevertheless, we are able to get some insight into the relationship between the features. Our three selected features are highly correlated which is understandable. Further the gdppc and the amount spent on education do not have very much impact which was a surprise as the dashboard clearly shows a relationship.