Predicting Human Development Index

Statistical Learning

Daniel Alonso

Master in Statistics for Data Science
Universidad Carlos III de Madrid

Academic Year 2020/2021

Statistical Learning Final Project: Predicting Country HDI

Daniel Alonso

December 19th, 2020

Contents

Dataset of choice

For this project I decided to pick a custom-built dataset obtained from the World bank Databank, specifically the World Development Indicators database. This is the "primary World Bank collection of development indicators" as stated on the database description. It has lots of economic, education, energy use, and population specific metrics.

I find demographic data fascinating, and I think this dataset will be quite good for predicting country development measures along with providing quite interesting and relevant information.

Variables

NOTE:

- **blue** = used for training/predicting
- **red** = target variable
- **green** = ID variables
- **purple** = variables excluded as either they were *components* of HDI or they were 100% correlated to another variable (like GNI/GDP, which are both 100% correlated and GNI is a component of HDI)

Variables in the original dataset as constructed using the World Bank Databank tool (variables were renamed):

- **year**: year the data was obtained in
- **year_code**: code for the year as the world bank databank sets it
- **country_name**: name of the country
- **country_code**: alpha-3 ISO 3166 code for the country
- **foreign_inv_inflows**: Foreign direct investment, net inflows (BoP, current US$)
- **exports_perc_gdp**: Exports of goods and services (as a % of GDP)
- **inflation_perc**: Inflation, consumer prices (annual %)
- **education_years**: Compulsory education, duration (years)
- **education_perc_gdp**: Government expenditure on education, total (as a % of GDP)
- **gds_perc_gdp**: Gross domestic savings (as a % of GDP)
- **gross_savings_perc_gdp**: Gross savings (as a % of GDP)
- **int_tourism_arrivals**: International tourism, number of arrivals
- **int_tourism_receipts**: International tourism, receipts (in current US$)
- **perc_internet_users**: Individuals using the Internet (as a % of population)
- **access_to_electricity**: Access to electricity (% of population)
- **agricultural_land**: Agricultural land (% of land area)
- **birth_rate**: Birth rate, crude (per 1,000 people)
- **gne**: Gross national expenditure (% of GDP)
- **mobile_subscriptions**: Mobile cellular subscriptions (per 100 people)
- **infant_mort_rate**: Mortality rate, infant (per 1,000 live births)
- **sex_ratio**: Sex ratio at birth (male births per female births)
- **greenhouse_gas_em**: Total greenhouse gas emissions (kt of CO2 equivalent)
- **urban_pop_perc**: Urban population (% of total population)
- **hdi**: human development index
- **hdi_cat**: Human development index as a category
- **life_exp**: Life expectancy at birth, total (years)
- **gdp**: GDP (current US$)
- **gni**: GNI (current US$)
- **fertility_rate**: Fertility rate, total (births per woman)

The target variable

As all these variables could perhaps tell us how developed a country is, we used a constructed categorized Human development index variable in order to classify the countries using the above variables (unless stated otherwise by their colour) as training variables.

The criteria for constructing the categorical variable *hdi_cat* was the following:

- Very high: HDI above 0.8
- High: HDI between 0.7 and 0.799
- Medium: HDI between 0.55 and 0.699
- Low: HDI under 0.55

This categorization is emulated from Wikipedia's construction and uses the same ranges as used in every Wikipedia article referencing HDI.

Data preprocessing

The data cleanup and preliminary feature engineering was done in Python and the imputation was then done in R within the same Jupyter Notebook (called *preprocessing.ipynb*).

Methodology and steps

1. Prior to importing there was a search within the .csv file with the following regex:

   ```
   \s\[[\w\S]{1,}\]
   ```

   as a find and replace and removing every instance of it. This regex matches a metadata tag that the world bank uses in their dataset. Following this step, the data was imported.

2. Metadata at the end of the dataset was removed, the dataset was filtered excluding these region codes (the codes were excluded as they were country aggregates, and we're only interested in the countries themselves).

3. The year column was converted to integer and the '..' values (which represent NAs in the World bank databank) were replaced for numpy NaNs and then the values were sorted by year and country name.

4. Data missing in later years (2020, 2019) was backfilled with data from previous years, as it still fits our modelling purposes. The data that goes the furthest back is from 18 years prior, so 2002.

5. Columns with more than 45 NA values were removed as this represents roughly 25

6. The life expectancy, GDP, GNI and fertility rate columns were removed as these are either components of HDI or (in the case of fertility rate) are 100

7. We scrape Wikipedia in order to obtain an updated metric on HDI estimates for each country. We also scrape it to obtain the alpha-3 3166 codes for each country. The data was obtained with the purpose of making it easier to join with the dataset obtained from the World Bank, as joining by country names is not reliable enough (misses 20 countries which the alpha-3 codes do not).

8. We add HDI to the main dataframe

9. Numerical columns were converted into floats and column names were simplified for easier future manipulation.

10. The categorical target variable is constructed as explained previously in this step.

11. The data is then exported as a csv, along with a json file that shows how the columns were renamed

12. The data is then imported in an R cell and a MICE imputation is performed using the 'cart' method with m = 5 in order to impute the very few missing values remaining.

Exploratory Data Analysis

Looking at variables by themselves

The plots of choice for visually analyzing the variables prior to any processing has been Boxplots and Histograms/density plots all looked at individually and categorized by *hdi_cat* which is our categorical target variable (HDI).

Given the space constraints of this report, if a plot for a specific variable is not shown here but talked about or mentioned, it will be shown in the *eda.ipynb* and *eda.html* notebook and notebook export used to perform the visual analysis. Only variables which are used in the modelling section will be visually analyzed and described.

Significantly right-skewed variables

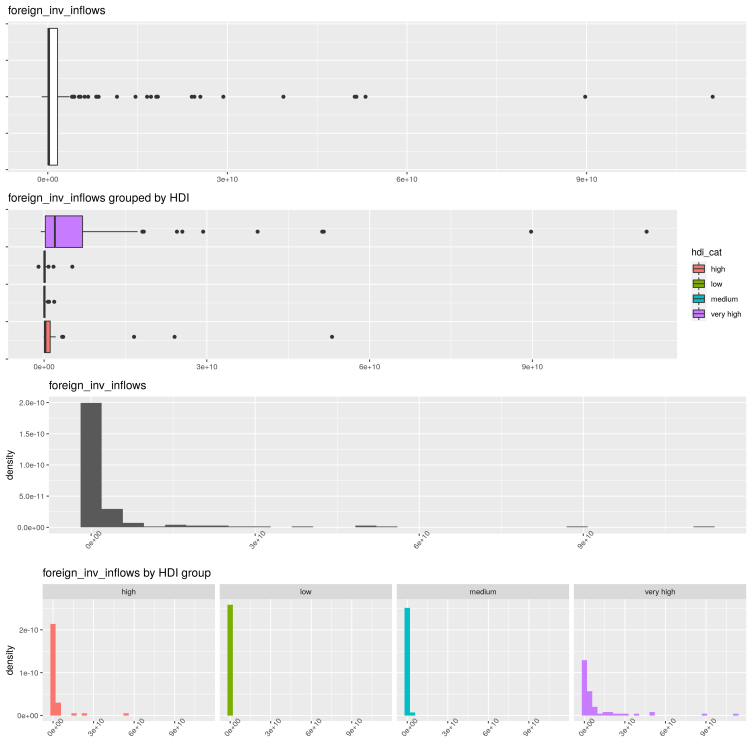

Figure 1: Foreign Investment Inflows

The following variables showed a similarly right-skewed shape with significant outliers:

- **foreign_inv_inflows**: Heavily right skewed, significant outliers (USA, UK, Germany, China. . .) etc. . . Given the histogram of *foreign_inv_inflows* when grouped by very high HDI vs low HDI, it would seem like the more developed a country is, the more foreign investment it should have. The grouped boxplot also shows the same trend.
- **int_tourism_receipts** and **int_tourism_arrivals**: Basically the same as with *foreign_inv_inflows*, we see that there's a significant tendency for higher development nations to receive significantly more tourism receipts and arrivals.
- **greenhouse_gas_em**: Just like with the previous two variables, fossil fuel use, meat production and economic activities like such produce huge amounts of greenhouse gases, and the more developed a country is, the more emissions it produces. However, this one is a bit less strongly inclined like the previous two. We can clearly see that there's many less developed countries with a high level of emissions. A notorious example of this is China, which is the 2nd country with the most emissions and

falls under the high category of HDI, by far exceeding the emission levels of most countries with very high HDI countries.

Education variables

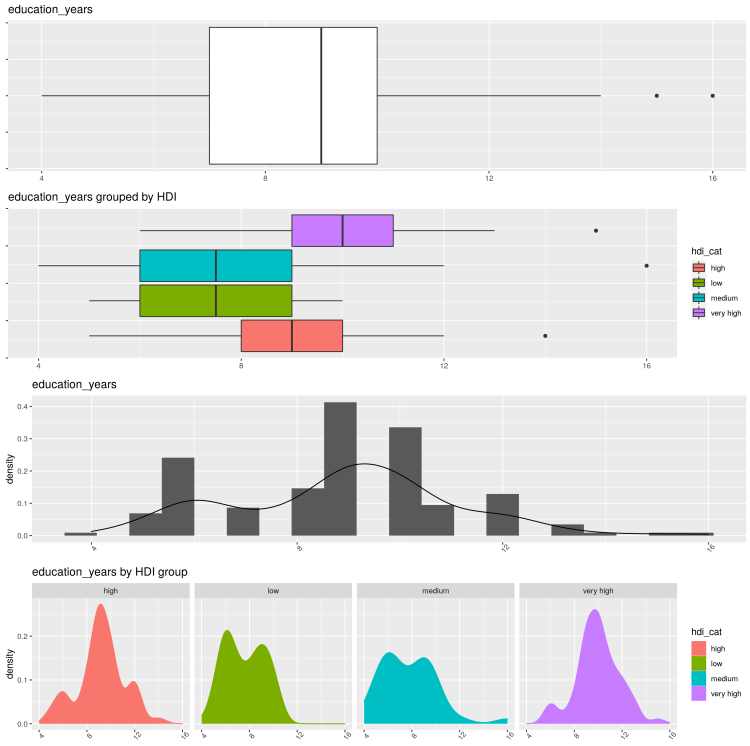

Figure 2: Years of compulsory education

- **education_years**: There's a clear tendency, both shown by our grouped boxplots and grouped density plots, for very high HDI countries to have significantly longer periods compulsory education, with few exceptions in each group, however a clear tendency that peaks between 8 and 12 years of education for both high and very high HDI countries. Low and medium HDI countries could be clumped together with similar compulsory education times, but with more variability for medium HDI countries.
- **education_perc_gdp**: How much are countries spending on education as a percentage of their GDP is a very interesting variable, as we can see that the consistency for higher and lower spendings respectively for low and very high HDI countries is quite solid, while medium HDI countries show an incredible variability versus other groups. High HDI countries show significant variability versus that of very high and low HDI countries, however, still aligning more iwth very high HDI countries than the rest.

Other economic variables

These variables might have some skewness towards more tails or longer tails than a normally distributed variable should have. Therefore they're hard to classify as particularly right or left skewed.

- **exports_perc_gdp**: This variable has a long left tail and it is right-skewed. We can see the box for medium developed countries is larger than others while very high and high HDI countries tend to have higher export amounts as percentage of GDP. Low HDI countries lag behind, as expected. It is interesting to see how medium HDI countries completely bridge the gaps between high, very high and low HDI countries in terms of exports, in the sense that there's plenty of countries with medium HDI with exports just as high as others with significantly higher HDIs.

- **inflation_perc**: For inflation we can see that independently of the HDI of a country, inflation could, perhaps, be an inevitable event of economic/political management or mismanagement and uncertainty. Either way, we can clearly see that the higher the HDI, the less uncertain such inflation rate will be. The boxes for medium and low HDI countries are significantly larger than very high and high therefore telling us that there's less consistency and high inflation events, while universal, are unpredictable, but less unpredictable and probably less common the higher the HDI of a country is.
- **gds_perc_gdp**: For gross domestic savings we can see that there's a clear difference between groups, with much more overlapping on the high end of HDI than the medium to low end of HDI. However, clearly, once again, the higher the HDI the higher the GDS. Much more variability again on those middle HDI groups (high and medium).
- **gross_savings_perc_gdp**: Gross savings as a percentage of GDP is an interesting variable, where the only defining feature of high and very high HDI countries being, again, relative consistency versus other groups, there might be seemingly no predictive capability in it due to the extremely lo variability of the values among groupsm however, it definitely differentiates very high and low HDI countries quite well.

Other interesting variables

These variables seemed like quite appropriate to me for predicting HDI or any other development measure, as they could potentially be huge differentiators between the different HDI groups.

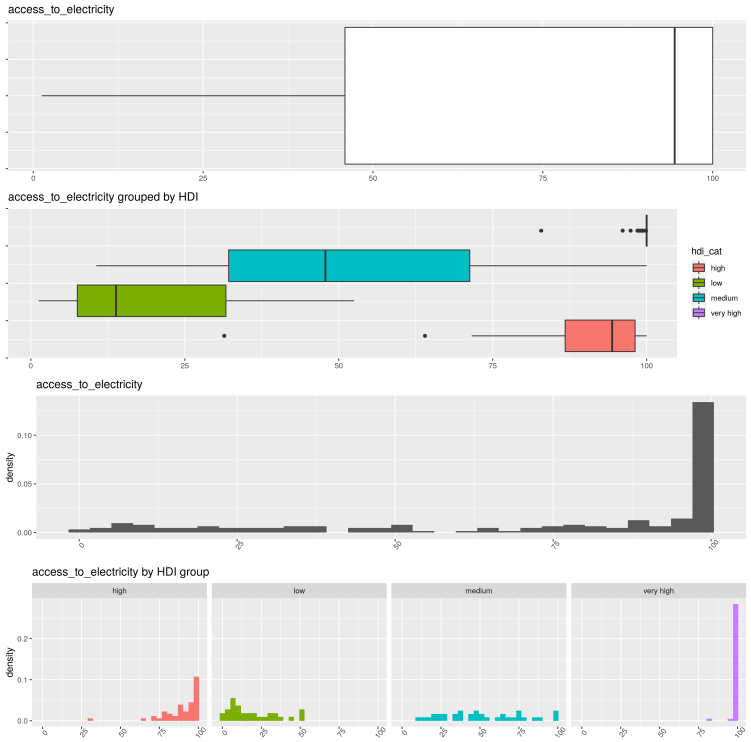

Figure 3: Percentage of population with access to electricity

- **perc_internet_users**: The percentage of internet users in each group is a surprisingly good differentiator as mentioned previously. We can see that, sure there's countries in each group with percentages that resemble other groups, but the difference in variability among groups is massive. Where the consistency of low HDI countries vs very high and high HDI countries is significantly different. We can see the standard deviations (and means) per group follow an interesting ladder: $\sigma_{low} \approx 0.714$, $\sigma_{medium} \approx 2.532$, $\sigma_{high} \approx 4.908$, $\sigma_{very\_high} \approx 20.382$.
- **access_to_electricity**: another Hugely differentiating variable with similar properties to

perc_internet_users. We can see a clear division where if the country is in the very high HDI category, it will almost certainly have near full coverage in its electricity supply to its population. Countries in the high category are also clearly differentiated with the rest while also maintaining quite high rates. Medium HDI countries show a huge variability in their coverage, lower than high HDI countries but higher than low HDI countries. Low HDI countries show a significantly lower but clearly divided coverage with medium HDI countries.

- **agricultural_land**: Agricultural land is an interesting variable that might not contribute a great deal to classifying, however, it is interesting to visualize how lower HDI countries tend to have a significantly higher percentage of agricultural land. However the rest of the variables all hover around the same values.
- **birth_rate**: Birth rate is a clearly defining variable, where we can see the lower HDI countries rise above the rest, along with the higher HDI countries jumping way down. Most very high to high HDI countries have already experienced a population growth plateau where the birth rate goes down significantly and usually either stays between 0 and 15 or very slightly above it (as it is necessary for demographic sustainability).
- **gne**: Gross national expenditure shows slightly lower values for very high HDI countries than the rest of groups, with higher variability for high HDI countries and low to medium HDI countries slightly higher. The general histogram for this value shows a long left tail with a normal-like distribution.
- **mobile_subscriptions**: An amazingly interesting variable with very similar properties to that of access to electricity and percentage of internet users. We see a very clear spread where the HDI groups are very clearly separated with very high HDI having an incredibly high variability while still being significantly higher than the rest of groups and the high HDI group lagging surprisingly behind but still a lot higher than low and medium HDI countries. The variable is quite right-skewed with most of the countries clumping around the low-medium area as there isn't an incredibly large difference between low and medium HDI countries for this one.
- **infant_mort_rate**: Quite left skewed as both high and very high HDI countries have most of their values closer to zero while low and medium HDI countries have a much larger spread with low HDI countries taking the top position given the poverty conditions. Medium HDI countries lag behind low HDI countries while still maintaining a really high infant mortality rate in general. The variable also clearly defines groups and might be quite good at classifying.
- **sex_ratio**: seemingly somewhat normally distributed with a long left tail. However quite surprisingly well defined for very high and high HDI countries. The differences between groups are interesting to see but maybe not quite as differentiating as other variables under this section.
- **urban_pop_perc**: Population of people living in urban areas is a quite spread out variable as we know the more developed a country is, the more its population tends to live in urban areas. Urban areas are usually higher income areas and tend to offer more opportunities for workers, therefore, highly developed nations, as it is clear, tend to have quite good job offerings in urban areas. We clearly see the groupings here, very high HDI countries have the largest urban populations, while low to medium HDI countries have the lowest. Interestingly so, low and medium HDI groups are not as well separated.

Correlation matrix

This correlation matrix was constructed by calculating the correlation coefficient for the variables using *Kendall*, *Spearman* and *Pearson* correlation coefficients and then imputting the values into a dataframe where only the highest correlation of the three calculated is added. This way we are able to really illustrate how correlated the variables truly are.

The idea came from observing significant differences between correlation calculated for a few pairs of variables, where the correlation coefficients nearly doubled when moving from *Pearson* to *Spearman* correlation coefficients.

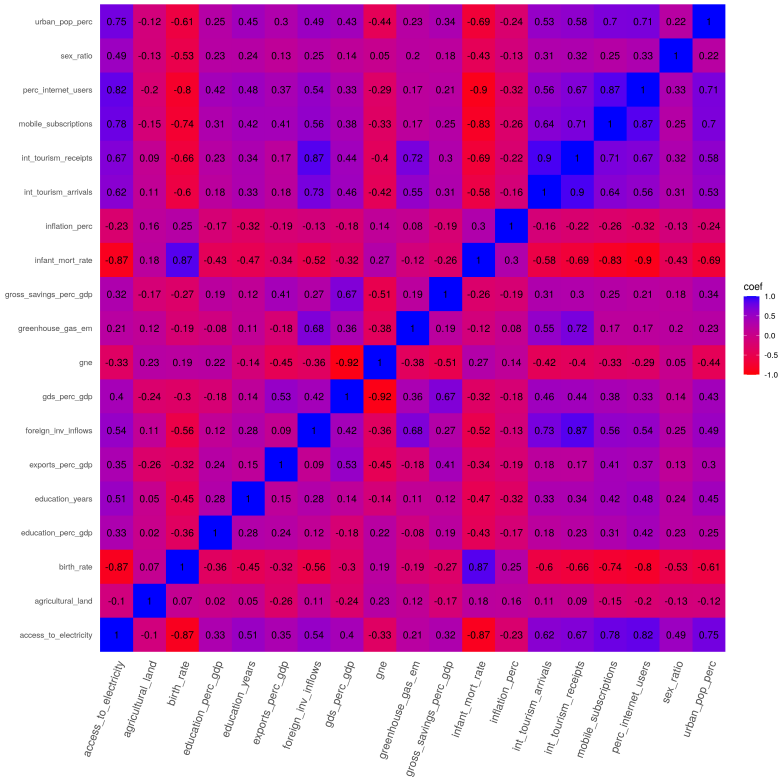

Figure 4: Correlation heatmap

Analysis

Looking at the correlations we highlight some important details:

- **int_tourism_receipts** and **int_tourism_arrivals** are extremely highly correlated at around *0.9*. These consideration will be looked at later on as they may or may not provide the model with interesting information when both are included in it.
- **access_to_electricity** and **perc_internet_users** are quite highly correlated at around *0.82* along with **perc_internet_users** and **mobile_subscriptions** at around *0.87* which also showed very similar trends in the individual variable analysis.

- **perc_internet_users** along with **access_to_electricity** and **infant_mort_rate**, both pairs with *-0.9* and *-0.87* respectively. The higher the infant mortality rate, the lower the percentage of internet users and people with access to electricity.
- **infant_mort_rate** and **birth_rate** at around *0.87* which is also quite high.

There's plenty of other interesting correlations to look at, however the general idea is that there's quite high correlation among some key variables. This makes a lot of sense, however, given how well some of these variables separate groups, this might not be a particular problem when predicting. Later on we will see whether this hurts the model in general or not.

When plotting a *pairs* plot (scatter plots) between the variables when grouped by HDI group, we see that there's a clear distinction similar to what we could appreciate in the boxplots, there's clear separation between groups in several of the plots.

Modelling: Statistical Learning

All models were prepared and ran in R using 3 repeats of 10-fold cross validation:

```
control <- trainControl(method = "repeatedcv",
                        repeats = 3,
                        number = 10 )
```

Method 1: Linear Discriminant Analysis

LDA is a generalization of fisher's linear discriminant. A method used to obtain a linear combination of deatures that classifies elements in two or more groups.

For our LDA model we used the following parametrization:

```
ldafit <- caret::train(hdi_cat ~ ., method = "lda", data = data, preProcess = c("center", "scale"),
               tuneGrid = expand.grid(alpha = seq(0, 2, 0.1), lambda = seq(0, 0.1, 0.01)),
               metric = "Accuracy", trControl = control)
```

Confusion matrix and variable importance

```
#>            Reference
#> Prediction  high low medium very high
#>   high        48   0      7         9
#>   low          0  29      5         0
#>   medium       3   3     24         0
#>   very high    1   0      0        55
```

We can see in the confusion matrix for this model that the model classifies quite decently, achieving an accuracy of ~**84.78%**, with a *95% confidence interval* of **(78.76%, 89.64%)**.

As for variable importance we see that our LDA model uses with somewhat similarly high importance the variables *infant_mort_rate*, *access_to_electricity*, *birth_rate*, *perc_internet_users*, *mobile_subscriptions* and *urban_pop_perc* for most variables, therefore confirming what we had suspected previously, that these variables would be strong group separators.

Method 2: Logistic Regression

A logistic/logit model has the purpose of modelling hte propability of a certain class or event with 2 or more categories/groups.

For our LR model we used the following parametrization:

```
lrfit <- caret::train(hdi_cat ~ ., method = "glmnet", data = data, preProcess = c("center", "scale"),
              tuneGrid = expand.grid(alpha = seq(0, 2, 0.1), lambda = seq(0, .1, 0.01)),
              metric = "Accuracy", trControl = control)
```

As shown above, we performed a grid search for both hyperparameters of the model, *lambda* and *alpha* with their respective intervals between 0 and 0.1 with steps of 0.01 for *lambda* and between 0 and 2 with steps of 0.1.

Confusion matrix and variable importance

```
#>            Reference
#> Prediction  high low medium very high
#>   high        48   0      8         8
#>   low          0  29      5         0
#>   medium       1   3     23         0
#>   very high    3   0      0        56
```

We can see in the confusion matrix for this model that the model classifies on par with the LDA model, achieving an accuracy of ~**83.15%**, with a *95% confidence interval* of **(76.95%, 88.26%)**.

The LR model seems significantly more selective with variables, essentially showing us that yes, the variables we previously suspected to be good classifying variables (*infant_mort_rate*, *access_to_electricity*, *birth_rate*, *perc_internet_users*, *mobile_subscriptions* and *urban_pop_perc*) had definitely a big role in classifying very high, high and medium HDI countries, however, this doesn't come without surprises, where a very relevant variable when classifying low HDI countries was *agricultural_land*, a variable that we didn't think would play a particularly big role in classification. However, while the model is good, it doesn't achieve the level of accuracy that we aim for (**greater than or equal to 90%**).

Method 3: Quadratic Discriminant Analysis

QDA is related to LDA where measurements per class are assumed to be normally distributed, however, it's not assumed that the covariance of each of the classes is identical.

QDA seems very appropriate for our purpose given our measurements and it shows its strengths in our model summary.

For our QDA model we used the following parametrization:

```
qdafit <- caret::train(hdi_cat ~ ., method = "qda", data = data, preProcess = c("center", "scale"),
                metric = "Accuracy", trControl = control)
```

Confusion matrix and variable importance

```
#>            Reference
#> Prediction  high low medium very high
#>   high        45   0      1         5
#>   low          0  32      1         0
#>   medium       5   0     34         0
#>   very high    2   0      0        59
```

We can see in the confusion matrix for this model that the model classifies very well, never placing medium or high HDI countries 2 categories above or below and reasonably classifying countries. The model achieves an accuracy of ~**92.39%**, with a *95% confidence interval* of **(87.56%, 95.78%)** which is significantly higher than our LR and LDA models.

For QDA, our best performing model, we notice that the consideration of variables is near-equal to that of LDA, however, the model achieves better performance.

Method 4: K-Nearest Neighbors

For our KNN model we used the following parametrization:

```
knnfit <- caret::train(hdi_cat ~ ., method = "knn", data = data, preProcess = c("center", "scale"),
                metric = "Accuracy", trControl = control)
```

Confusion matrix and variable importance

```
#>            Reference
#> Prediction  high low medium very high
#>   high        43   0     10         8
#>   low          0  30     11         0
#>   medium       3   2     15         0
#>   very high    6   0      0        56
```

We can see in the confusion matrix for this model that the model classifies worse than previous models, making several mistakes in the middle groups (high and medium), achieving an accuracy of ~**77.17%**, with a *95% confidence interval* of **(70.42%, 83.03%)**.

KNN variable importance strongly resembles that of LDA and QDA, and we could, perhaps, go as far as to say that for our conditions, it seems essentially identical. However, due to its performance versus the former 2, we do not use KNN for this analysis. At least not with the parametrization we have chosen. When testing using grid search we didn't really achieve a hugely significant difference in results, and even when higher, we didn't quite rival the excellent performance of QDA for our dataset.

Modelling: Machine Learning

For the second modelling phase we will use machine learning models. These often provide stronger predictive/classifying performance at some extra computational cost.

The focus of this part will be entirely focused on performance and increasing the accuracy and maximizing the predictive power of our chosen models, however, as overfitting could happen, I must reinforce that the main objective here is to prove how some important metrics that describe the use of technology, the environment and how other humans interact and create their offspring, all have a strong relation to how developed each respective country is.

We must not separate the predictive power of the models from the reality that how society behaves and how it uses technology and its resources is strongly related to how we, today, classify countries and choose to inhabit/migrate to them/visit them and also interact with its inhabitants.

A few key points:

- Our *trainControl* variable remains the same from the previous modelling phase, we will use 10 repeats of 3-fold cross validation for all models.
- Models will all be obtained using the *caret* package, which provides a very intuitive API for preparing and running models in R (as we could previously see in the modelling phase 1)

SVM with Polynomial Kernel

The first model we test is Support-vector machines. This is a strong supervised learning model. My chosen variant was SVM with Polynomial Kernel. This model is quite robust at classification and it's very appropriate for our problem (as a supervised learning model), we use a polynomial kernel function as our model might be fit decently using a non-linear kernel.

We tune our hyperparameters as follows:

- *degree*: we test polynomials degree 2, 3, 4 and 5
- *C*: we test costs 0.01, 0.1, 0.5 and 1
- *scale*: we use 2 for this hyperparameter

The model is trained as follows:

```
svmfit <- caret::train(hdi_cat ~ .,
                method = "svmPoly",
                data = data,
                preProcess = c("center", "scale"),
                metric = "Accuracy",
                tuneGrid = expand.grid(degree = c(2,3,4,5),
                                       C = c(0.01,0.1,0.5,1),
                                       scale = 2),
                trControl = control)
```

Our resulting table is as follows:

```
#>            Reference
#> Prediction  high low medium very high
#>   high        52   0      4         4
#>   low          0  30      1         0
#>   medium       0   2     31         0
#>   very high    0   0      0        60
```

The model already has a significantly higher accuracy than any of our previous models. There's an interesting feature in it though, it predicts the high HDI group with perfect accuracy. This might be a result of this specific run, but the model has an outstanding accuracy of **94.02%** with a *95%-CI* with a lower bound of **89.56%** and an upper bound of **96.98%**.

It is surprising how incredibly powerful and accurate the model is with the given data even after performing cross-validation.

The worst variables variables we have for this models (according to our variable importance table) are *inflation_perc*, *gne*, *greenhouse_gas_em* and *agricultural_land*. The ones with highest importance are

infant_mort_rate, *access_to_electricity*, *birth_rate*, *perc_internet_users* and *mobile_subscriptions*. We had previously had similar success with these variables.

I tried several other SVMs (linearSVM, radialSVM, etc) and their accuracy wasn't better than the one obtained by the SVM with polynomial kernel, therefore I will move on to try other models.

Neural networks

Multi-Layer Perceptron

An MLP is a type of feedforward artificial neural network. This is a very simple NN which only possesses an input layer, a hidden layer and an output layer.

It's an appropriate and very simple NN for our task and we only tune a single hyperparameter:

- *size*: we test number of neurons in the model for the values 10, 50, 100 and 150

```
mlpfit <- caret::train(hdi_cat ~ .,
                       method = "mlp",
                       data = data,
                       preProcess = c("center", "scale"),
                       metric = "Accuracy",
                       tuneGrid = expand.grid(size=c(10,50,100,150)),
                       trControl = control)
```

And these are our results:

```
#>            Reference
#> Prediction  high low medium very high
#>   high        51   0      4         3
#>   low          0  31      4         0
#>   medium       0   1     28         0
#>   very high    1   0      0        61
```

The MLP model shows even stronger results with a near-perfect accuracy of **96.2%**, with a *95%-CI* between **92.32%** and **98.46%**. The model is impressively good and quite fast to run. Only fails on very few instances but seems to aim higher, where the only mistakes it makes (with the exception of those for actual very high HDI countries) are usually looking up (high for very high, medium for high, etc).

This model tends to prefer the same top variables as the ones tested previously. This idea keeps hinting that perhaps, HDI might not be a modern enough index to measure true country development and that we might be better off either making it a more complex index, or creating a new one altogether with similar parameters and variables, but much more updated data.

Tree-based models

The tree based models I've chosen are *eXtreme Gradient Boosting* and *Boosted Logistic Regression*. Both yield outstandingly good results these are probably overfitted to the data, however, this does not prevent us from making a final conclusion about how our variables relate to HDI.

eXtreme Gradient Boosting

As for tree-based models, the first model we'll try will be *xgbTree*. Gradient boosting produces an ensemble prediction model composed of other tree-based models, where we optimize them utilizing a differentiable loss function.

This model is well known for strong performace, we will tune our model hyperparameters as follows:

- *nrounds*: we will try boosting iterations of 100 and 120
- *max_depth*: we will try a tree depth of 6 and 9
- *eta*: the learning rate will be adjusted between 0.3 and 0.5
- *gamma*: we will test the gamma parameter for regularization with 0 and 0.1
- *min_child_weight*: for the minimum leaf weight we'll try 1 and 1.2
- *subsample*: row sampling will be left as default: 1

- *colsample_bytree*: column sampling will be left as default: 1

```
xgbtreefit <- caret::train(hdi_cat ~ .,
                  method = "xgbTree",
                  data = data,
                  preProcess = c("center", "scale"),
                  metric = "Accuracy",
                  tuneGrid = expand.grid(nrounds = c(100,120),
                                         max_depth = c(6,9),
                                         eta = c(0.3, 0.5),
                                         gamma = c(0, 0.1),
                                         min_child_weight = c(1, 1.2),
                                         colsample_bytree = 1,
                                         subsample = 1),
                  trControl = control)
```

We can see our confusion matrix:

```
#>       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
#>   1.000000e+00   1.000000e+00   9.801514e-01   1.000000e+00   3.478261e-01
#> AccuracyPValue  McnemarPValue
#>   4.079771e-85            NaN
```

We get an absurd accuracy of **100%**, with a *95%-CI* between **98.02%** and **100%**. This clearly indicates our model is most likely overfit to the data. However, this does not prevent us from seeing which variables the model considers most important.

As the variable importance is concerned, *access_to_electricity* seems to be an extremely important variable, along with *infant_mort_rate*, *birth_rate* and *perc_internet_users*.

We should perhaps attempt to fit a new *xgbTree* model using only the top 3 variables and with the same previous parametrization and check its results:

```
#>       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
#>   1.000000e+00   1.000000e+00   9.801514e-01   1.000000e+00   3.478261e-01
#> AccuracyPValue  McnemarPValue
#>   4.079771e-85            NaN
```

And it still reaches **100%** accuracy, showing that most of the variables included in this specific model might not be necessary in order to determine HDI. However, even if they provide little information to the model vs the top ones, some might really contribute to determining how developed a country is, as shown in previous models.

Boosted Logistic Regression

LogitBoost is a boosting algorithm, where an AdaBoost algorithm is considered as a generalized additive model and the cost function of a logistic regression is applied.

The decision of testing this algorithm comes from the fact that it has an outrageously good accuracy with this dataset.

We tune our hyperparameters as follows:

- *nIter*: we test number of iterations between 50 and 100 in intervals of 10.

```
logitboostfit <- caret::train(hdi_cat ~ .,
                  method = "LogitBoost",
                  data = data,
                  preProcess = c("center", "scale"),
                  metric = "Accuracy",
                  tuneGrid = expand.grid(nIter = seq(50,100,10)),
                  trControl = control)
```

Our resulting table is as follows:

```
#>       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
#>   1.000000e+00   1.000000e+00   9.801514e-01   1.000000e+00   3.478261e-01
#> AccuracyPValue  McnemarPValue
#>   4.079771e-85            NaN
```

We can see that this model achieves a **100%** accuracy with a *95%-CI* with a lower bound of **98.02%** and an upper bound of **100%**.

Doesn't get any better than that, does it?

The worst variables variables we have for this models (according to our variable importance table) are *inflation_perc*, *gne*, *greenhouse_gas_em* and *agricultural_land*. The ones with highest importance are *infant_mort_rate*, *access_to_electricity*, *birth_rate*, *perc_internet_users* and *mobile_subscriptions*. This

model is quite more selective with variables, where some variables have importance 0 for some categories and 100 for others. We can see here how strong some of these variables are at classifying HDI.

Conclusion and final thoughts

As far as interests go, I am definitely profoundly interested in this topic of measuring the level of development a country has or is undergoing.

Initially, I thought that, perhaps, predicting HDI was going to be the final goal of this project, however, I have realized that there could be a far more interesting task at hand (for either future me, or future someone else), and that might be to construct a new index or improve HDI.

HDI is a somewhat dated measure. It was originated in the annual Human Development reports produced by the Human Development Report Office of the UNDP. The reports were launched in 1990 and had a purpose that, while noble, at the time was created with what nowadays we would consider lacklusted data.

Today we measure thousands of different metrics of each country, and maybe a more modern measure could maybe improve on or even replace HDI. As a final master project idea, I'm considering taking on this task (if possible), as a new measure to classify and determine the level of development a country has, could be proven to be very useful.

What HDI pretty much does nowadays is to classify countries by how developed they are, but as far as I've noticed, it doesn't seem to have much more use other than appear in the UNDP Human Development Report. The index is simply a weighted index of other measures, basically, a summary of 3-4 variables (life expectancy, mean/expected years of schooling and gross national income). These, while solid at classifying countries, might miss a few bits and pieces when it comes to accurately painting a picture of a country.

And what is development anyway? From which point of view do we see it? The economy? Education? Healthcare?, perhaps we need to modernize this notion and use more modern data to do so. A more cultural/technological approach to it could be applied.

A good example of a country which might see its "theoretical new HDI" slightly changed could be, for instance, Singapore, where the GNI, life expectancy and Education indexes are extraordinarily high, however, the country does not have the same freedom of speech or strict data protection regulations that countries in the EU have. This shows lack of development in legislation, as Singapore is a significantly younger nation than European states.

Many of these metrics (like freedom of speech) will be a challenge to accurately construct a full picture of (in numeric data), but the task seems highly interesting to me!

Anyway, the main conclusion is, the top metrics selected are incredibly powerful at predicting a country's HDI and therefore, how developed they theoretically are.