Practice 0: Naïve Bayes

Javier Esteban Aragoneses, Mauricio Marcos, Danyu Zhang, Daniel Alonso

February 7th, 2021

Dataset selection

In this Project, we will use a database obtained from the World Bank Databank, specifically the World Development Indicators database. It consists of economics, education, energy use and population specific information. So, in this case, our goal is to predict, using demographic variables, a development measure called "Human Development Index" (from now on, "HDI").

We have selected a country list dataset where we predict HDI category per country for this assignment. We replace *very high* and *high* HDI with only the tag **high**. We apply the same treatment to *medium* and *low* HDI with only the tag **low**. This simplifies the analysis greatly, and improves accuracy.

Our target variable here will be *hdi_cat*.

Variables:

We have ## number of features in our dataset, each one expresses the following:

- **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)

Train-Test split

We create a train-test split with 70% train and 30% test.

Creating a Naïve Bayes model

In this case, we will use the Naïve Bayes Classificator, to predict weather a country will be in the group of *very high*, *high*, *medium* or *low* HDI.

The Naïve Bayes is sometimes a good model to predict a classification through the study of features, as it takes into account prior probabilities, to estimate the prediction. So first, the algorithm will find the prior probabilities (given by the percentaje of each category in the training sample set) and then will calculate the likelihoods, using the Bayes Rule. But when doing so, it is making two big assumptions that may arise in problems later:

- The effect of the value of a predictor (or features) on a given class (high or low HDI) is independent of the values of other predictors.
- Each feature makes an equal contribution to the outcome.

This is why, after calculating every conditional probability given each feature value, it multiplies all the probabilities to find the final probability of each observation for each category.

In other words, according to our first assumption the percentage of urban population in a country is independent from the life expectancy of the country in a given class, and the total greenhouse gas emissions is also independent from both of them. Then, our second assumptions says that none of the features is meaningless to predict our categories, and that all of them gives equal contribution to the predictions. We will see that this is not the case.

Finally, the algorithm will calculate the probability of being of a specific category and then will create a classifier. For this, we find the probability of given set of inputs for all possible values of the class variable (very high, high, medium, low, very low) and pick up the output with the maximum probability.

As we can see, the model takes as prior probabilities for each cateogory, the percentaje of observations from the training set of each category. In this case, we have that prior probability of being "low HDI": 0.398 and the prior probability of a "high HDI" is 0.602. Then, it calculates the frequencies for each attribute against the target, and then the likelihoods in a table. In this case the likelihood is the probability of the predictor given a specific class. And then, it calculates the posterior probabilities.

Predicting HDI category

Performing the prediction:

Calculating a confidence interval for each prediction:

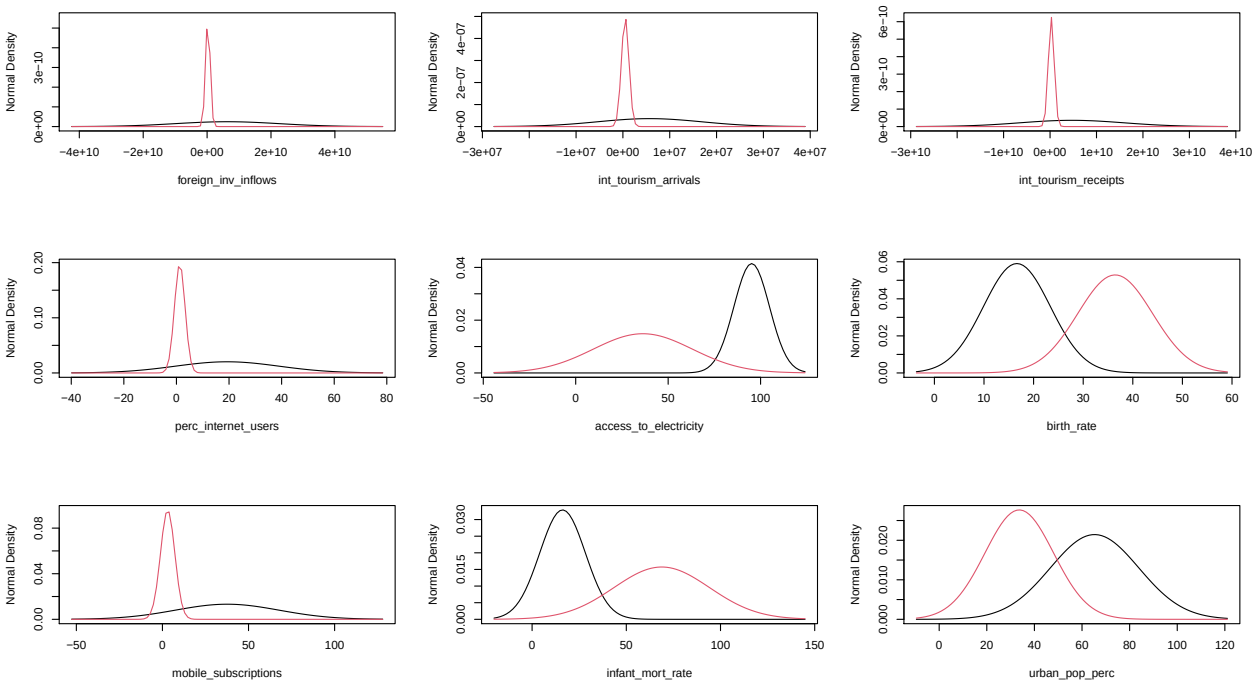
Confusion matrix with results:

```
#>            Reference
#> Prediction high low
#>       high   23   0
#>       low     7  26
```

The following result yields a model with 85.71% accuracy. The model is overall quite accurate given the small size of the training set in terms of total amount of observations. The specificity and sensitivity are both quite decent as well, hovering ~85%.

Predictive power per variable

We only plot the variables that show most strength when predicting:



Variables that classify the best

Now, we can plot each variable kernel density, distinguishing the categories, to see which variables are the most helpful at predicting each HDI category.

In other words, we can inspect every plot and see what variables behave differently for each category and thus, will be more helpful predicting the category of the observations.

The following are the variables that most help us in categorizing:

```
#> [1] "foreign_inv_inflows"   "int_tourism_arrivals"  "int_tourism_receipts"
#> [4] "perc_internet_users"   "access_to_electricity" "birth_rate"
#> [7] "mobile_subscriptions"  "infant_mort_rate"      "urban_pop_perc"
```

For example, we can clearly see that “access to electricity” is a feature that really helps us in distinguishing between the categories. In general, we could say that countries with a low percentage of the population with access to electricity are more inclined to be categorized as a low HDI country, while if an observation (a country) has a large percentage of “access to electricity” it is more inclined to be categorized as a high HDI country.

On the other hand, the variable “birth_rate” is also important, but in a different direction: if the country presents a low birth_rate it is more inclined to be categorized as high HDI, while a high “birth_rate” presented by a country is a sign of inclination towards a low HDI country.

Critics

As we can see from our description of the model, the two assumptions we take are obviously not correct in the case of our features.

First, our features are not independent one from the other, meaning that there is some kind of dependancy between them. It is pretty obvious that the demographic and economical factors of a country are in some

way dependant one from the other. For example, we can think that a country with a high infant mortality will probably have a low level of education, or in a country with a lot of inflation there will not be very high levels of savings (as the savings would lose value with time if the inflation persists).

Second, as we just showed with the plots explantions there are some variables that clearly have a much bigger impact that others in differentiationg the categories of our response variable. In other words: we can say that the birth rate of a country contributes more to predict wether this country is of low or high HDI than, for instance, the agricultural land.

Appendix: Code

```
# importing libraries
library(MASS)
library(dplyr)
library(e1071)
library(mlbench)
library(caret)

# reading the data into a dataframe
data <- read.csv('./data/data.csv')

# removing columns that aren't useful
data <- data[names(data) != "hdi" &
             names(data) != "X" &
             names(data) != "country_name" &
             names(data) != "country_code" &
             names(data) != "year_code" &
             names(data) != "year"]

# changing the very high, medium categories to
# high and low respectively
cat <- c()
for (i in 1:length(data$hdi_cat)) {
    if (data$hdi_cat[i] == "very high") {
        cat <- c(cat, "high")
    } else if (data$hdi_cat[i] == "medium") {
        cat <- c(cat, "low")
    } else {
        cat <- c(cat, data$hdi_cat[i])
    }
}

# replacing the original hdi_cat col with
# the modified version
data$hdi_cat <- cat

# train-test split
n=nrow(data)
trainset=(1:n)%in%sample(n,floor(n*0.7))
testset=!trainset

# naive bayes model
model <- naiveBayes(hdi_cat ~ ., data = data,subset = trainset)

# predict and classification of probabilities
pred = predict(model,data[testset,],type="raw")
pred_cat <- ifelse(pred[,1] > 0.9, "high", "low")

# adding 2 columns to the prediction table
# to create a confidence interval for each
# prediction
num_highs <- table(pred_cat)[1]
pred <- cbind(pred, rep(0,56))
```

```r
pred <- cbind(pred, rep(0,56))
colnames(pred) <- c("high","low","low_CI", "high_CI")

# performing the CI calculation
for (i in 1:length(pred_cat)) {
    p <- pred[i,1]
    val <- qnorm(0.975)*sqrt((p*(1-p))/num_highs)
    pred[i,3] = p-val
    pred[i,4] = p+val
}

# converting the columns to factors
pred <- as.factor(pred_cat)
data$hdi_cat <- as.factor(data$hdi_cat)

# displaying a confusion matrix to assess model quality
confusionMatrix(pred, data$hdi_cat[testset])

# assessing the quality of each variable at predicting
mms=apply(data[1:19],2,function(x) unlist(by(x,data$hdi_cat,mean)))
sds=apply(data[1:19],2,function(x) unlist(by(x,data$hdi_cat,sd)))

# plotting the kernel densities
par(mfrow=c(3,2))
for(i in 1:19){
  rrx=range(c(mms[,i]+3*sds[,i],mms[,i]-3*sds[,i]))
  rry=c(0,max(dnorm(mms[,i],mms[,i],sds[,i]+0.0001)))
  plot(0,1,type="n",xlab=colnames(mms)[i],ylab="Normal Density",xlim=rrx,ylim=rry)
  ss=seq(rrx[1],rrx[2],length.out = 100)
  for(j in 1:2) points(ss,dnorm(ss,mms[j,i],sds[j,i]),type="l",col=j)
}

# good columns, variables that are solid at predicting
# HDI category
good_cols <- c("foreign_inv_inflows", "int_tourism_arrivals", "int_tourism_receipts", "perc_internet_us
good_cols
```