ECON F215: Assignment

Soumitra Shewale	(2021B3A70788H)
Suryashashank Venkata Gudipudi	(2021B3AA0866H)
C. Varun	(2021B3AA3031H)
Swarup Bhuyan	(2021A7PS2821H)
Jyotirmoy Singh	(2021B3A72513H)

1. Introduction

We were tasked with replicating the ML methods in the paper Machine learning approach for predicting under-five mortality determinants in Ethiopia: evidence from the 2016 Ethiopian Demographic and Health Survey.

During this assignment, we:

1. Downloaded and cleaned the data
2. Understood the research question and replicated the variables used in the paper.
3. Provided descriptive statistics for the variables we used in our paper. (See Section 2).
4. Replicated the ML exercises in the research article (See section 3).
5. Interpreted the results (See Section 4).

This document is probably inside a .zip file and many other files relevant to this assignment. We have included the following:

- Two R scripts, named “preprocess.R” and “analysis.R”, namely preprocessing our dataset and analyzing the dataset. Our scripts are commented, so anyone can understand them, even with little knowledge of R.
- The original dataset we used, downloaded from DHS, called “ETIR71FL.dta”
- A folder containing our plot images, called “plots”
- The preprocessed dataset, called “original_dt.rda”
- The preprocessed dataset split into training and testing data, called “downsampled_train.rda” and “test_dt.rda”
- “vars.txt” - A text file documenting all the variables in our dataset.

We used the individual dataset to create a new data frame that includes all children with variables that correspond to the household. We did this because the children’s recode only included those under 5.

Using this new dataset of all children, first, we calculated whatever new variables we would need for our analysis. In doing this, we naturally had to filter out observations that didn’t provide any data, for example, NaN or NA values.

After that, we split the dataset in a ratio of 80:20 for our training and testing data, respectively. Then, we downsampled our training data to equal the proportion of under-five mortalities and children alive above 5. By doing this, we could finally run our analysis on our data. We performed all our training using k-folds cross-validation with k = 10, just like the paper's authors did.

During this entire process, we used little tricks in R that we knew and learned about to speed up the entire process or reduce code repetition.

2. Overview of the variables used

Our variable list for the child dataframe (dt) is:

- caseid: caseid, the ID of the observation
- v212: age of mother at first birth
- v115: time taken to get to water source
- v113: source of water
- v116: type of toilet facility
- v208: births in last five years
- v025: type of place of residence (urban/rural)
- v106: highest education level of mother
- v190: wealth index
- v364: contraceptive use
- v024: region
- v437: mother's weight in kgs
- v438: mother's height in cm
- v367: whether wanted last child
- v136: number of household members
- v426: when child was first breastfed
- b4: sex of child
- b5: whether child is alive
- b11: interval preceding birth
- midx: index of child (whether child is 1st, 2nd or 3rd, etc.)
- m14: number of antenatal visits during pregnancy
- m15: place of delivery
- m70: whether baby underwent postnatal check within 2 months

Note that these are the variables before we transform them into the format the paper needs them in. We will perform rudimentary analysis only on the variables we use in the ML models we reproduce.

On transforming these variables into the form the paper requires, we have:

- b5: whether child is alive (0 for no, 1 for yes)
- m_age_at_birth: mother's age at first birth (0 if < 20, 1 if > 20)
- sex: sex of child (0 if female, 1 if male)
- midx: index of child (whether child is 1st, 2nd or 3rd, etc.)
- birth_interval: interval preceding birth (0 for < 2 years, 1 for 2-4 years, 2 for > 4 years)
- v115: time taken to get to water source
- water_source: whether water source is "improved" according to WHO categories (1 for yes, 0 for no)
- toilet_facility: whether toilet facility is "improved" according to WHO categories (1 for yes, 0 for no)

- v208: births in last five years
- residence: 0 for rural, 1 for urban
- mother_edu: mother’s education: 0 for none, 1 for primary, and 2 for secondary and higher
- wealth_index: 0 for poorest and poor, 1 for middle, and 2 for rich and richest
- contraceptive_use: 0 for using, 1 for not using
- mothers_bmi: BMI of the mother
- place_of_delivery: place of delivery, whether place was a public (1), private (2), or NGO (3) hospital, or a home (0)
- antenatal_visits: 0 if no antenatal visits, 1 if 1-4 visits, and 2 if 5+ visits
- postnatal_care: 1 if postnatal care received, 0 if not
- wanted_child: 0 if child was wanted then, 1 if child was wanted later, 2 if child was not wanted
- breastfed: 0 if child was breastfed more than an hour of birth, 1 if within an hour
- v136: household size

State dummy variables:

- tigray
- afar
- amhara
- oromia
- somali
- benishangul
- snnpr
- gambela
- harari
- addis_adaba
- dire_dawa

These are all variables that can be found in Table 3 of the paper. The table also defines which categories are base categories.

We will now perform some rudimentary analysis on these variables:

```
> summary(dt$m_age_at_birth)
   0    1 
6462 3974 
> summary(dt$sex)
   0    1 
5050 5386 

> summary(dt$birth_interval)
   0    1    2 NA's 
```

```
2347 4107 1838 2144
> summary(dt$water_source)
   0    1
3939 6497
> summary(dt$toilet_facility)
   0    1
6452 3984
> summary(dt$residence)
   0    1
8535 1901
> summary(dt$mother_edu)
   0    1    2
6699 2649 1088
> summary(dt$place_of_delivery)
   0    1    2    3
7033 2955  250  198
> summary(dt$wealth_index)
   0    1    2
3926 4492 2018
> summary(dt$contraceptive_use)
   0    1
2704 7732
> summary(dt$mothers_bmi)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.4185 18.4704 20.0746 20.5402 22.0754 83.8548
> summary(dt$antenatal_visits)
   0    1    2 NA's
2434 3176 1452 3374
> summary(dt$postnatal_care)
   0    1 NA's
6395  667 3374
> summary(dt$wanted_child)
   0    1    2
8311 1469  656
> summary(dt$v115)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.0    20.0    40.0   228.3   120.0   998.0
> summary(dt$v208)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.000   2.000   1.755   2.000   5.000
> summary(dt$b5)
   0    1
 623 9813
> summary(dt$breastfed)
   0    1 NA's
2756 7287  393

> summary(dt$midx)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.000   1.000   1.377   2.000   5.000

> chisq.test(dt$b5, dt$residence)

      Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$residence
X-squared = 27.494, df = 1, p-value = 1.576e-07

> chisq.test(dt$b5, dt$mother_edu)

      Pearson's Chi-squared test

data:  dt$b5 and dt$mother_edu
X-squared = 14.842, df = 2, p-value = 0.0005985

> chisq.test(dt$b5, dt$place_of_delivery)

      Pearson's Chi-squared test

data:  dt$b5 and dt$place_of_delivery
X-squared = 23.643, df = 3, p-value = 2.966e-05

> chisq.test(dt$b5, dt$birth_interval)

      Pearson's Chi-squared test

data:  dt$b5 and dt$birth_interval
X-squared = 68.924, df = 2, p-value = 1.08e-15

> chisq.test(dt$b5, dt$water_source)

      Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$water_source
X-squared = 8.5731, df = 1, p-value = 0.003412

> chisq.test(dt$b5, dt$toilet_facility)

      Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$toilet_facility
X-squared = 0.05142, df = 1, p-value = 0.8206


> chisq.test(dt$b5, dt$m_age_at_birth)
```

```
      Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$m_age_at_birth
X-squared = 0.99726, df = 1, p-value = 0.318

> chisq.test(dt$b5, dt$sex)

      Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$sex
X-squared = 13.216, df = 1, p-value = 0.0002776

> chisq.test(dt$b5, dt$antenatal_visits)

      Pearson's Chi-squared test

data:  dt$b5 and dt$antenatal_visits
X-squared = 44.437, df = 2, p-value = 2.242e-10

> chisq.test(dt$b5, dt$postnatal_care)

      Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$postnatal_care
X-squared = 5.6731, df = 1, p-value = 0.01723

> chisq.test(dt$b5, dt$wanted_child)

      Pearson's Chi-squared test

data:  dt$b5 and dt$wanted_child
X-squared = 18.4, df = 2, p-value = 0.000101

> chisq.test(dt$b5, dt$wealth_index)

      Pearson's Chi-squared test

data:  dt$b5 and dt$wealth_index
X-squared = 29.835, df = 2, p-value = 3.322e-07






> chisq.test(dt$b5, dt$contraceptive_use)
```

```
	Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$contraceptive_use
X-squared = 15.627, df = 1, p-value = 7.714e-05

> chisq.test(dt$b5, dt$breastfed)

	Pearson's Chi-squared test with Yates' continuity correction

data:  dt$b5 and dt$breastfed
X-squared = 1.317, df = 1, p-value = 0.2511
```

We will now summarize these variables in one place and discuss the above results.

- Mother's age at birth (m_age_at_birth): The chi-squared test for this variable's correlation with child mortality shows that the correlation is pretty low. Most of the mothers in our sample seem to have first given birth below the age of 20.
- Sex of the child (sex): The chi-squared test shows that this is pretty well correlated with child mortality. The sample includes an even proportion of both sexes.
- Birth interval (birth_interval): Most of the children in the sample seem to be born 2-4 years after the child before them. A lot of these observations are also NA, since they might be the first child. The chi-squared test seems to show that this is also pretty well correlated with child mortality.
- Water source (water_source): The sample mostly represents people with "improved" water sources, according to WHO. The chi-squared test shows that this too, is pretty well correlated with child mortality.
- Toilet facility (toilet_facility): The sample contains largely the people with "improved" toilet facility, according to WHO. The chi-squared test shows that this doesn't have that much correlation with child mortality.
- Residence (residence): The sample consists mostly of rural respondents. The chi-square test shows high correlation with child mortality.
- Mother's education (mother_edu): The chi-square test shows good correlation with this variable. In our sample, most of the mothers are uneducated, or have minimal education.
- Place of Delivery(place_of_delivery): Most of the children (~⅓) were born at home. This is followed by a public hospital. The number of children born in an NGO hospital or a private hospital is relatively less. The chi-square test shows a very high correlation between place of delivery and child mortality.
- Wealth Index (wealth_index): Most of the children come from families that are poor or middle class. The chi-square test shows a very high correlation between wealth index and child mortality.
- Contraceptive Use (contraceptive_use): A lot of people do not use contraceptives. The chi-square test shows a high correlation between contraceptive use and child mortality.
- Mother's BMI (mothers_bmi): The average BMI of the mother is 20.37. The median is 20.

- Time to get to water source (v115): The time to get to the nearest water source in minutes. The data seems heavily skewed toward people that don’t have quick access to water sources.
- Antenatal Visits (antenatal_visits): A lot of mothers had 0 antenatal visits. There are a lot of NA values as there is not much data available. The chi-square test shows a very high correlation between antenatal visits and child mortality.
- Postnatal Care (postnatal_care): Most of the mothers do not receive postnatal care. There are NA values as the data was not available. The chi-square test shows some correlation between postnatal visits and child mortality.
- Wanted Child (wanted_child): Most of the mothers wanted the child with very few not wanting them even later. The chi-square test shows moderate correlation between whether the child was wanted and child mortality.
- Births in last few years (v208): Most of the households survey have around two births in the last 5 years.
- Child mortality (b5): Our dependent variable throughout the paper. There are 635 under 5 mortalities in our 10,641 observations.
- Birth order of child (midx): Most children in our dataset are first children.
- Breastfed (breastfed): A lot of children were breastfed within an hour of birth. The chi-square test shows little correlation between breastfeeding and child mortality

3. Replicating the ML models

First, we'll run through the pre-processing part of the script. First, we will load the libraries we will use for the rest of the time.

```r
# Install and Load libraries that we will use
# install.packages("haven", "dplyr", "tidyr", "class", "caret",
"randomForest")
library(haven)
library(dplyr)
library(tidyr)
library(class)
library(caret)
library(randomForest)
```

Then, we load the dataset. We are using the dataset's Individual recode to gather information about the family and the children.

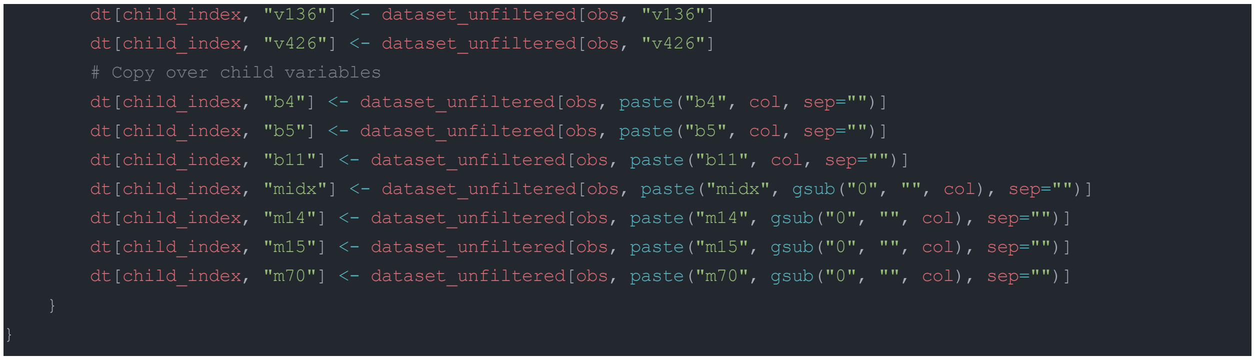
```r
# Load dataset that we will filter
# Source:
https://dhsprogram.com/data/dataset/India_Standard-DHS_2006.cfm
dataset_unfiltered <- read_dta("./ETIR71FL.dta")
```

Now, we need to take the individual's data and transform it into the children's data. We create the children's dataframe's structure here:

```r
dt = data.frame(
    caseid=c(), # caseid, the ID of the observation
    v212=c(), # age of mother at first birth
    v115=c(), # time taken to get to water source
    v113=c(), # source of water
    v116=c(), # type of toilet facility
    v208=c(), # births in last five years
    v025=c(), # type of place of residence (urban/rural)
    v106=c(), # highest education level of mother
    v190=c(), # wealth index
    v364=c(), # contraceptive use
    v024=c(), # region
    v437=c(), # mother's weight in kgs
    v438=c(), # mother's height in cm
    v367=c(), # whether wanted last child
    v136=c(), # number of household members
    v426=c(), # when child was first breastfed
    b4=c(), # sex of child
    b5=c(), # whether child is alive
    b11=c(), # interval preceding birth
    midx=c(), # index of child (whether child is 1st, 2nd or 3rd,
etc.)
    m14=c(), # number of antenatal visits during pregnancy
    m15=c(), # place of delivery
    m70=c() # whether baby underwent postnatal check within 2 months
)
```

Now, we copy the data into the dataframe:

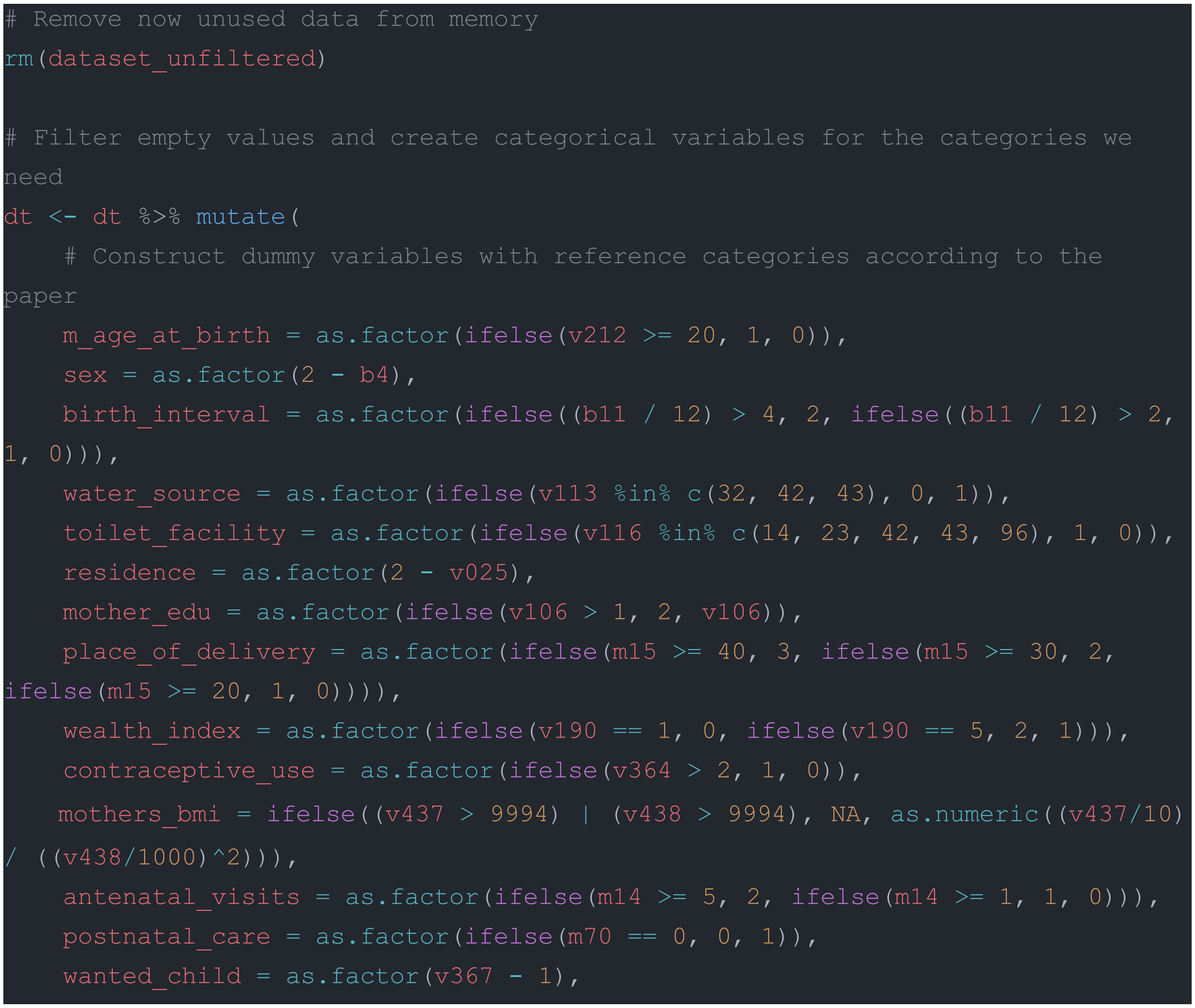
```
for (obs in 1:nrow(dataset_unfiltered)) {
    for (col in c("_01", "_02", "_03", "_04", "_05", "_06")) {
        # If observation is not present, there are no more kids in the household
        if (is.na(dataset_unfiltered[obs, paste("b4", col, sep="")])) {
            break
        }

        # If the observation has a missing household index, it will be filtered later, so ignore it
        if (is.na(dataset_unfiltered[obs, paste("midx", gsub("0", "", col), sep="")])) {
            next
        }

        # Count the row for the current child
        child_index <- nrow(dt) + 1

        # Insert new data into the row
        # Add child line number (in-household identifier) to caseid to identify child
        dt[child_index, "caseid"] <- paste(dataset_unfiltered[obs, "caseid"], dataset_unfiltered[obs, paste("b16",
col, sep="")])
        dt[child_index, "v212"] <- dataset_unfiltered[obs, "v212"]
        # Copy over the household variables
        dt[child_index, "v115"] <- dataset_unfiltered[obs, "v115"]
        dt[child_index, "v113"] <- dataset_unfiltered[obs, "v113"]
        dt[child_index, "v116"] <- dataset_unfiltered[obs, "v116"]
        dt[child_index, "v208"] <- dataset_unfiltered[obs, "v208"]
        dt[child_index, "v025"] <- dataset_unfiltered[obs, "v025"]
        dt[child_index, "v106"] <- dataset_unfiltered[obs, "v106"]
        dt[child_index, "v190"] <- dataset_unfiltered[obs, "v190"]
        dt[child_index, "v364"] <- dataset_unfiltered[obs, "v364"]
        dt[child_index, "v024"] <- dataset_unfiltered[obs, "v024"]
        dt[child_index, "v437"] <- dataset_unfiltered[obs, "v437"]
        dt[child_index, "v438"] <- dataset_unfiltered[obs, "v438"]
        dt[child_index, "v367"] <- dataset_unfiltered[obs, "v367"]
```



```
        dt[child_index, "v136"] <- dataset_unfiltered[obs, "v136"]
        dt[child_index, "v426"] <- dataset_unfiltered[obs, "v426"]
        # Copy over child variables
        dt[child_index, "b4"] <- dataset_unfiltered[obs, paste("b4", col, sep="")]
        dt[child_index, "b5"] <- dataset_unfiltered[obs, paste("b5", col, sep="")]
        dt[child_index, "b11"] <- dataset_unfiltered[obs, paste("b11", col, sep="")]
        dt[child_index, "midx"] <- dataset_unfiltered[obs, paste("midx", gsub("0", "", col), sep="")]
        dt[child_index, "m14"] <- dataset_unfiltered[obs, paste("m14", gsub("0", "", col), sep="")]
        dt[child_index, "m15"] <- dataset_unfiltered[obs, paste("m15", gsub("0", "", col), sep="")]
        dt[child_index, "m70"] <- dataset_unfiltered[obs, paste("m70", gsub("0", "", col), sep="")]
    }
}
```

We loop first through every observation in the individual dataset, and then loop through their children. We then skip the observation every time no more children are left.

Then, we quickly clear up some memory space and create all the dummy variables we need for our analysis.



```
# Remove now unused data from memory
rm(dataset_unfiltered)

# Filter empty values and create categorical variables for the categories we need
dt <- dt %>% mutate(
    # Construct dummy variables with reference categories according to the paper
    m_age_at_birth = as.factor(ifelse(v212 >= 20, 1, 0)),
    sex = as.factor(2 - b4),
    birth_interval = as.factor(ifelse((b11 / 12) > 4, 2, ifelse((b11 / 12) > 2, 1, 0))),
    water_source = as.factor(ifelse(v113 %in% c(32, 42, 43), 0, 1)),
    toilet_facility = as.factor(ifelse(v116 %in% c(14, 23, 42, 43, 96), 1, 0)),
    residence = as.factor(2 - v025),
    mother_edu = as.factor(ifelse(v106 > 1, 2, v106)),
    place_of_delivery = as.factor(ifelse(m15 >= 40, 3, ifelse(m15 >= 30, 2, ifelse(m15 >= 20, 1, 0)))),
    wealth_index = as.factor(ifelse(v190 == 1, 0, ifelse(v190 == 5, 2, 1))),
    contraceptive_use = as.factor(ifelse(v364 > 2, 1, 0)),
    mothers_bmi = ifelse((v437 > 9994) | (v438 > 9994), NA, as.numeric((v437/10) / ((v438/1000)^2))),
    antenatal_visits = as.factor(ifelse(m14 >= 5, 2, ifelse(m14 >= 1, 1, 0))),
    postnatal_care = as.factor(ifelse(m70 == 0, 0, 1)),
    wanted_child = as.factor(v367 - 1),
    breastfed = as.factor(ifelse(v426 <= 100, 1, 0)),

    # Create dummy variables for regions
    tigray = as.factor(ifelse(v024 == 1, 1, 0)),
    afar = as.factor(ifelse(v024 == 2, 1, 0)),
    amhara = as.factor(ifelse(v024 == 3, 1, 0)),
    oromia = as.factor(ifelse(v024 == 4, 1, 0)),
    somali = as.factor(ifelse(v024 == 5, 1, 0)),
    benishangul = as.factor(ifelse(v024 == 6, 1, 0)),
    snnpr = as.factor(ifelse(v024 == 7, 1, 0)),
    gambela = as.factor(ifelse(v024 == 8, 1, 0)),
    harari = as.factor(ifelse(v024 == 9, 1, 0)),
    addis_adaba = as.factor(ifelse(v024 == 10, 1, 0)),
    dire_dawa = as.factor(ifelse(v024 == 11, 1, 0)),
)
```

We make sure that the types of the columns are correct, and we save the processed dataset for descriptive statistics we ran before:

```
# Correct the types of columns
dt$v208 <- as.numeric(dt$v208)
dt$midx <- as.numeric(dt$midx)
dt$v136 <- as.numeric(dt$v136)
dt$v115 <- as.numeric(dt$v115)
dt$b5 <- as.factor(dt$b5)

# Remove NA values for BMI so they don't affect us later
dt <- dt %>% filter(!is.na(mothers_bmi))

# Save the dt so that we can load it for other analysis on the
variables
save(dt, file="original_dt.rda")
```

Then, we split the testing and the training data:

```
# Separate training and testing data
test_sample <- sample.int(length(dt[["b5"]]),
round(length(dt[["b5"]]) * 0.2))
```

```
train_dt <- dt[-test_sample,]
test_dt <- dt[test_sample,]
```

Now, we downsample the training data to make sure that there is no bias towards one or the other outcome in the training data, just like the paper discussed. After that, we save the datasets so that we can load these up later quickly.

```
# Separate dead and alive in training data
alive <- train_dt %>% filter(b5 == 1)
dead <- train_dt %>% filter(b5 == 0)

# Downsample training data
downsample <- sample.int(length(alive[["b5"]]),
round(length(dead[["b5"]])))
alive_downed <- alive[downsample,]
downsampled = rbind(dead, alive_downed)

# Save the dataset so far
save(downsampled, file="downsampled_train.rda")
save(test_dt, file="test_dt.rda")
```

Now, in the regression script, we load up the libraries and dataset we will use.

```
# Install and Load libraries that we will use
# install.packages("haven", "dplyr", "tidyr", "class", "caret",
"randomForest")
library(haven)
library(dplyr)
library(tidyr)
library(class)
library(caret)
library(randomForest)

# Load datasets
load("downsampled_train.rda")
load("test_dt.rda")
```

We can start off with logistic regression:

```
# Run the regression for the logistic regression model to predict whether the
child is alive or not
# Extra findings:
mdl <- glm(b5 ~ m_age_at_birth + sex + midx + birth_interval + v115 +
water_source + toilet_facility + v208 + residence + mother_edu + wealth_index +
contraceptive_use + tigray + afar + amhara + oromia + somali + benishangul +
snnpr + gambela + harari + addis_adaba + dire_dawa + mothers_bmi +
place_of_delivery + antenatal_visits + postnatal_care + wanted_child +
breastfed + v136,
               family = "binomial",
               data = downsampled)

# See coefficients of the model
summary(mdl)

# See odds ratios with confidence intervals for the model
exp(cbind(coef(mdl), confint(mdl)))

# Look at the performance of the model
glm_predicted <- predict(mdl, newdata = test_dt, type = "response")
actual_values <- test_dt$b5
pred_values <- as.factor(ifelse(glm_predicted > 0.4, 1, 0))
confusionMatrix(actual_values, pred_values)
```

Here, we run the logistic regression, look at the coefficients, the confusion matrix, and some other metrics like accuracy, specificity, sensitivity, etc.
We will look at the model's performance later in Section 4.

Now, we can continue with k-Nearest Neighbors:

```
# Select variables we will use for our kNN model
knn_train <- downsampled %>% select(
    m_age_at_birth, # mother's age at birth
    sex, # sex of child
    midx, # birth order of child
    # birth_interval, # birth interval
    v115, # time to water source
    water_source, # water source
    toilet_facility, # toilet facility
    v208, # births in last five years
    residence, # type of place of residence (urban/rural)
    mother_edu, # mother's education
    wealth_index, # wealth index
    contraceptive_use, # contraceptive use

    # regions
    tigray,
    afar,
    amhara,
    oromia,
    somali,
    benishangul,
    snnpr,
    gambela,
    harari,
    addis_adaba,
    dire_dawa,

    mothers_bmi, # mother's bmi
    place_of_delivery, # place of delivery
    # antenatal_visits, # number of antenatal visits
    # postnatal_care, # whether child received postnatal care
    wanted_child, # whether parents wanted child
    # breastfed, # whether child was breastfed within 1 hour or less after birth
    v136 # number of people in household
)
```

Here, we select only the variables we will need for k-Nearest Neighbours. Some variables, like antenatal_visits, postnatal_care, breastfed, and birth_interval, are commented out because they have significant NA values and seem to throw the kNN off. This results in errors.

We run a similar selecting process on the testing data:

```
# Select variables we will use for our kNN model
knn_test <- test_dt %>% select(
    m_age_at_birth, # mother's age at birth
    sex, # sex of child
    midx, # birth order of child
    # birth_interval, # birth interval
    v115, # time to water source
    water_source, # water source
    toilet_facility, # toilet facility
    v208, # births in last five years
    residence, # type of place of residence (urban/rural)
    mother_edu, # mother's education
    wealth_index, # wealth index
    contraceptive_use, # contraceptive use

    # regions
    tigray,
    afar,
    amhara,
    oromia,
    somali,
    benishangul,
    snnpr,
    gambela,
    harari,
    addis_adaba,
    dire_dawa,

    mothers_bmi, # mother's bmi
    place_of_delivery, # place of delivery
    # antenatal_visits, # number of antenatal visits
    # postnatal_care, # whether child received postnatal care
    wanted_child, # whether parents wanted child
    # breastfed, # whether child was breastfed within 1 hour or less after birth
    v136 # number of people in household
)
```

We select the exact same variables this time as well.

We also normalize all the numeric variables we use in kNN:

```
# Normalize numeric variables
knn_train$v115 <- scale(knn_train$v115)
knn_test$v115 <- scale(knn_test$v115)

knn_train$v208 <- scale(knn_train$v208)
knn_test$v208 <- scale(knn_test$v208)

knn_train$v136 <- scale(knn_train$v136)
knn_test$v136 <- scale(knn_test$v136)
```

```
knn_train$midx <- scale(knn_train$midx)
knn_test$midx <- scale(knn_test$midx)

knn_train$mothers_bmi <- scale(knn_train$mothers_bmi)
knn_test$mothers_bmi <- scale(knn_test$mothers_bmi)
```

Then, it's just a matter of running the kNN and seeing its results:

```
# Run the kNN model to predict whether the child is alive or not,
with k = 5
classifier_knn <- knn(
    train = knn_train,
    test = knn_test,
    cl = downsampled$b5,
    k = 1,
    prob = TRUE
)

# Print info about the kNN model
actual_values <- test_dt$b5
print("Confusion matrix:")
confusionMatrix(actual_values, classifier_knn)
```

We will look at the results for this too, in Section 4.

Random Forests

We also replicated the random forests classification algorithm, which was used in the paper to classify the under 5 child mortality rate.

To select our dataset, we again used our downsampled dataset as given in the research paper, and we split our dataset into 80% training and 20% testing.

```
# Run the random forest model
rf <- randomForest(b5 ~ m_age_at_birth + sex + midx + v115 +
water_source + toilet_facility + v208 + residence + mother_edu +
wealth_index + contraceptive_use + tigray + afar + amhara +
oromia + somali + benishangul + snnpr + gambela + harari +
addis_adaba + dire_dawa + mothers_bmi + place_of_delivery +
wanted_child + v136, data = downsampled, ntree = 500)

# Run it on our test data
y_pred <- predict(rf, newdata = test_dt)

# Look at the confusion matrix, and look at the plot
print("Confusion matrix:")
table(test_dt$b5,y_pred)
plot(rf)
varImpPlot(rf)
```

We used variables that did not have NA values to predict whether the child was alive or dead.

4. Interpretation of the Results

First, for the logistic regression model, these were our coefficients:

```
                                  2.5 %    97.5 %
(Intercept)        0.7775634 0.04222856 15.216346
m_age_at_birth1    1.8550244 1.05309296  3.348116
sex1               0.7664920 0.45577773  1.278926
midx                      NA         NA        NA
birth_interval1    2.0471624 1.10920713  3.802963
birth_interval2    2.2564402 0.99286137  5.205384
v115               1.0003439 0.99915634  1.001576
water_source1      0.7748421 0.42948041  1.385642
toilet_facility1   0.8310457 0.43885412  1.568134
v208               0.7574204 0.48758081  1.175602
residence1         2.0414158 0.57971971  7.602948
mother_edu1        1.0002414 0.52013279  1.953384
mother_edu2        0.7431437 0.22139184  2.653320
wealth_index1      1.3228756 0.64956121  2.705982
wealth_index2      0.6912978 0.19151332  2.605188
contraceptive_use1 0.7721297 0.39405013  1.483159
tigray1            1.3584796 0.27651098  6.579500
afar1              1.6024081 0.32423742  7.778532
amhara1            1.5580018 0.31216272  7.604164
oromia1            1.5256331 0.32890503  6.807164
somali1            3.0493481 0.66681675 13.896556
benishangul1       1.4144908 0.27743839  7.000505
snnpr1             1.3043569 0.27423610  5.951024
gambela1           1.0611012 0.20244798  5.425719
harari1            0.9159006 0.17661014  4.664563
addis_adaba1       1.3737752 0.12187425 34.329704
dire_dawa1                NA         NA        NA
mothers_bmi        0.9835821 0.89812875  1.075517
place_of_delivery1 1.7140998 0.74662051  4.109693
place_of_delivery2 2.2940114 0.22464420 56.134068
place_of_delivery3 0.5939917 0.10872961  3.852650
antenatal_visits1  1.9896013 1.07711266  3.733250
antenatal_visits2  1.8573230 0.76909142  4.650709
postnatal_care1    1.5155992 0.45756708  6.315580
wanted_child1      1.9027268 0.81709416  4.813311
wanted_child2      0.5027801 0.20790062  1.228231
breastfed1         0.6291110 0.33140302  1.163161
v136               1.1979524 1.05464252  1.370213
```

The coefficients that the paper got were:

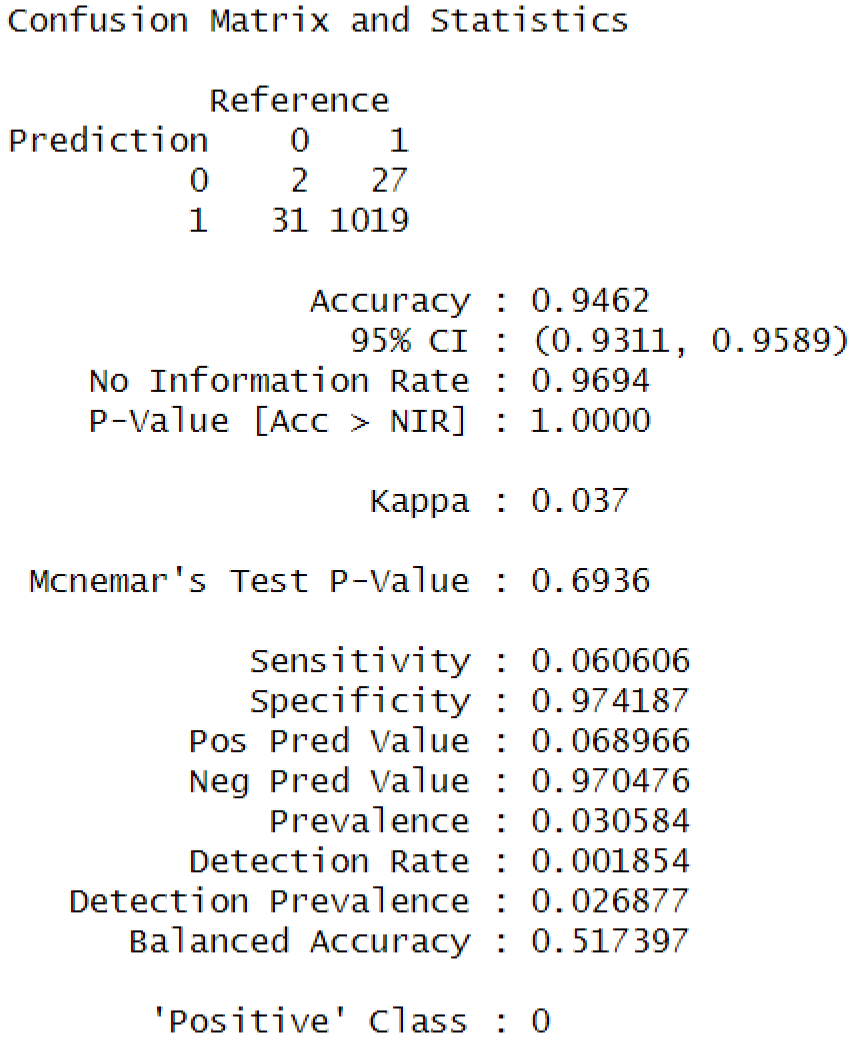
Table 3 Logistic regression analysis of under-five mortality in Ethiopia

Variables	Odds ratio	Lower 95 % CI	Upper 95% CI	*p* value
(Intercept)	0.033	0.006	0.193	**0.0001**
Mothers age first birth (Ref: < 20)				
> 20	0.600	0.353	1.018	0.059
Sex (Ref: female)				
Male	2.018	1.398	2.913	**0.0001**
Birth order (Ref: 1st/2nd)				
3rd or higher	2.129	1.131	4.008	**0.020**
Birth interval (Ref: < 2)				
2–4 years	0.527	0.309	0.898	**0.019**
> 4 years	0.385	0.190	0.779	**0.008**
Time to water source	1.000	0.999	1.000	0.244
Water source (Ref: unimproved)				
Improved	0.585	0.348	0.985	**0.044**
Toilet facility (Ref: improved)				
Unimproved	1.713	0.744	3.943	0.206
Births in last 5 years	1.163	0.744	1.816	0.508
Residence (Ref: rural)				
Urban	0.527	0.181	1.541	0.243
Mother's education (Ref: no education)				
Primary	0.928	0.513	1.680	0.805
Secondary/higher	1.856	0.480	7.178	0.370
Wealth index (Ref: low)				
Middle	1.342	0.698	2.581	0.378
High	1.694	0.937	3.064	0.082
Contraceptive use (Ref: using)				
Not using	1.174	0.735	1.876	0.502
Region				
Addis Ababa	1.124	0.485	2.605	0.786
Afar	0.573	0.228	1.435	0.235
Amhara	0.885	0.354	2.211	0.794
Ben-Gumuz	1.494	0.587	3.803	0.400
Dire Dawa	1.021	0.408	2.554	0.965
Gambella	0.623	0.243	1.597	0.325
Harari	1.175	0.495	2.790	0.715
SNNP	1.221	0.376	3.960	0.740
Somali	1.504	0.287	7.881	0.629
Tigray	1.733	0.519	5.787	0.372
Mother's BMI (Ref: normal)				
Overweight	0.527	0.170	1.640	0.269
Underweight	1.402	0.868	2.264	0.168
Place of delivery (Ref: fac with CS delivery)				
Facility without CS delivery	2.850	1.182	6.869	**0.020**
Home	1.185	0.617	2.275	0.610
Antenatal visits (Ref: no visit)				
1–4 visits	0.616	0.381	0.995	**0.048**
5+ visits	0.437	0.208	0.917	**0.029**
Postnatal care (Ref: no)				
Yes	0.264	0.080	0.872	**0.029**
Child wanted (Ref: wanted then)				
Wanted later	0.768	0.369	1.599	0.482
Not at all	1.407	0.749	2.642	0.289
Breastfeeding (Ref: > an hour of birth)				
Within 1 h of birth	0.242	0.147	0.398	**0.0001**
vHousehold size	0.498	0.345	0.719	**0.0001**

Our coefficients are roughly in the same order of magnitude as the coefficients in the paper. We chalk the small differences in these coefficients up to model misspecification. Some of the variables in the paper are not well described, making it much harder to replicate them, and get exact values.

The wealth index variable for example; There is no indication on DHS websites, resources, or otherwise about which categories are considered “low”, “middle” or “high” on the scale, but the paper uses this. To account for this, we have used what we can only call our best guess as to how the variable was created. In this case, we considered the categories “poorest” and “poor” in the “low” bin, “richest” and “rich” in the “high” bin, and “middle” in the “middle” bin.

This aside, we should now take a look at the confusion matrix:



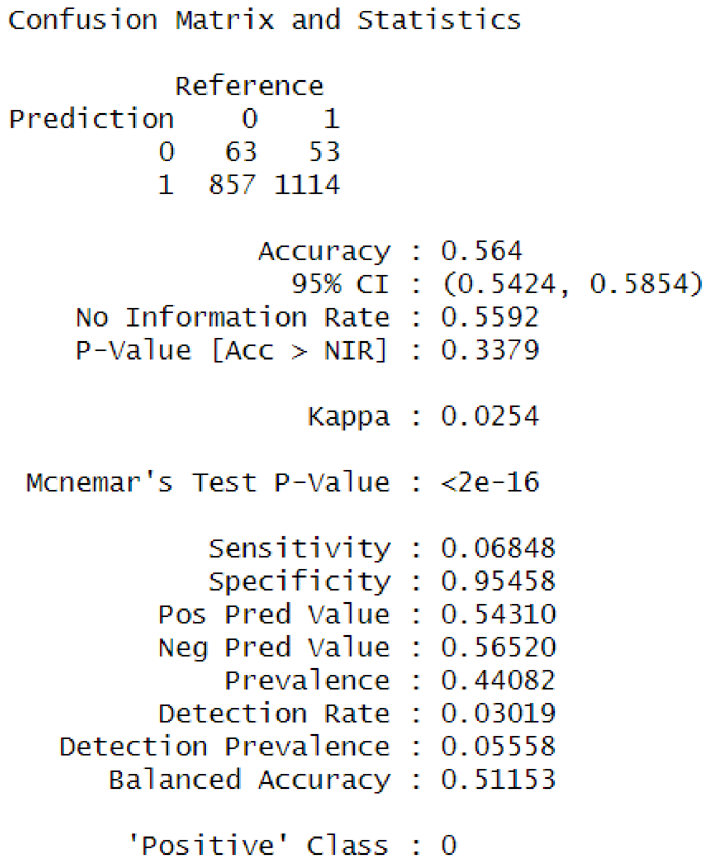
```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0    2   27
         1   31 1019
                                          
               Accuracy : 0.9462          
                 95% CI : (0.9311, 0.9589)
    No Information Rate : 0.9694          
    P-Value [Acc > NIR] : 1.0000          
                                          
                  Kappa : 0.037           
                                          
 Mcnemar's Test P-Value : 0.6936          
                                          
            Sensitivity : 0.060606        
            Specificity : 0.974187        
         Pos Pred Value : 0.068966        
         Neg Pred Value : 0.970476        
             Prevalence : 0.030584        
         Detection Rate : 0.001854        
   Detection Prevalence : 0.026877        
      Balanced Accuracy : 0.517397        
                                          
       'Positive' Class : 0               
```

It seems that our model has extremely high accuracy, while the specificity, and positive and negative predictive values come out to be very low.

We believe that the reason for the specificity being that high is that the number of children alive is much higher, causing a much larger value specificity, since specificity is the ratio of the true negatives (in this case the children that are alive, and are predicted correctly by the model to be alive) to the sum of the true negatives and false positives (in this case the children that are alive, irrespective of the model's prediction.)

Accounting for all this, the balanced accuracy comes out to be around 50% which is pretty close to what was achieved in the paper.

We will now look at the k-Nearest Neighbours method and the result it gives us.



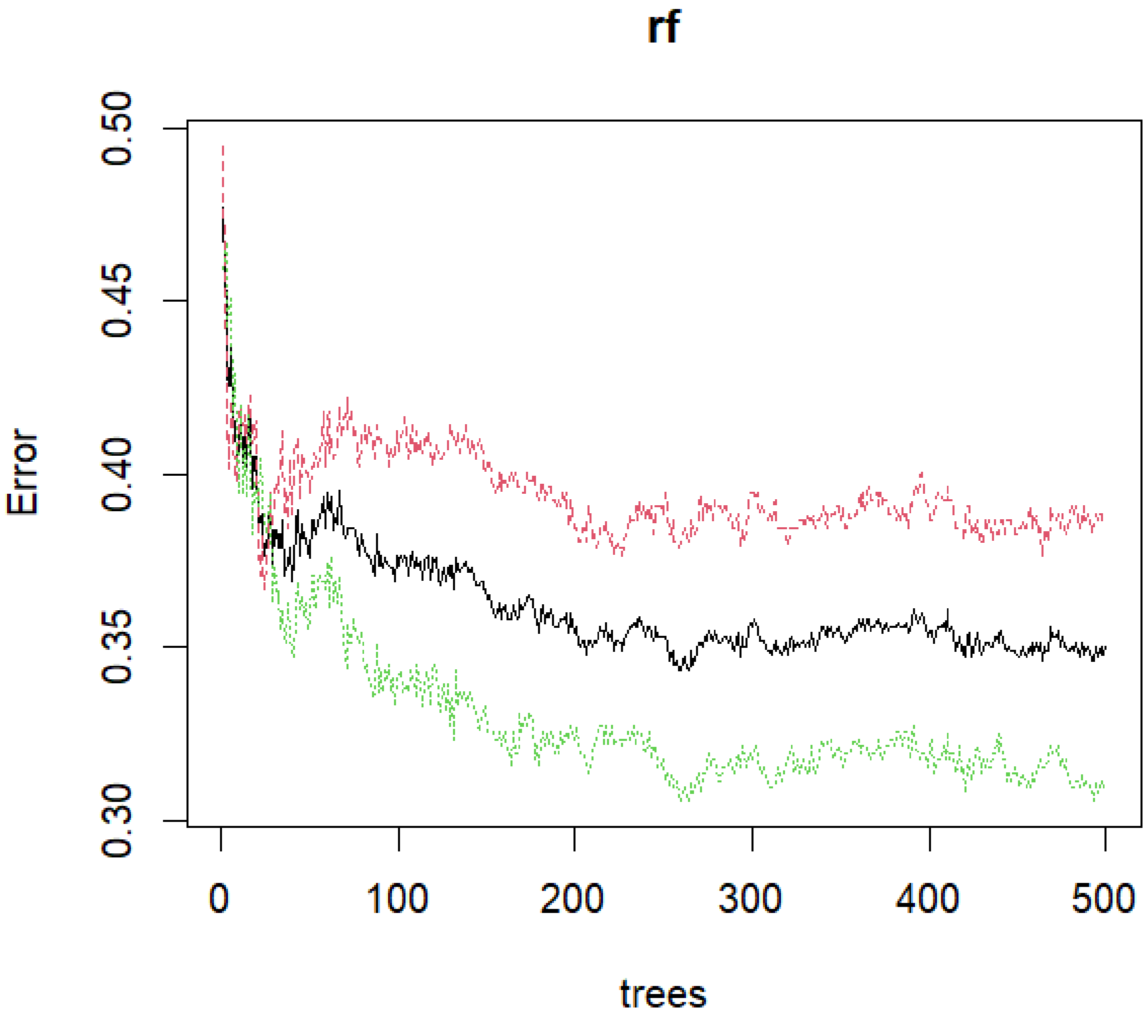
```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0   63   53
         1  857 1114

               Accuracy : 0.564
                 95% CI : (0.5424, 0.5854)
    No Information Rate : 0.5592
    P-Value [Acc > NIR] : 0.3379

                  Kappa : 0.0254

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.06848
            Specificity : 0.95458
         Pos Pred Value : 0.54310
         Neg Pred Value : 0.56520
             Prevalence : 0.44082
         Detection Rate : 0.03019
   Detection Prevalence : 0.05558
      Balanced Accuracy : 0.51153

       'Positive' Class : 0
```

The confusion matrix is once again pretty similar to the paper. The model has a low accuracy of around 55-50% usually, based on the value of k (In the screenshot, we have used k = 1). The sensitivity of the model is again pretty low, while the specificity remains high.

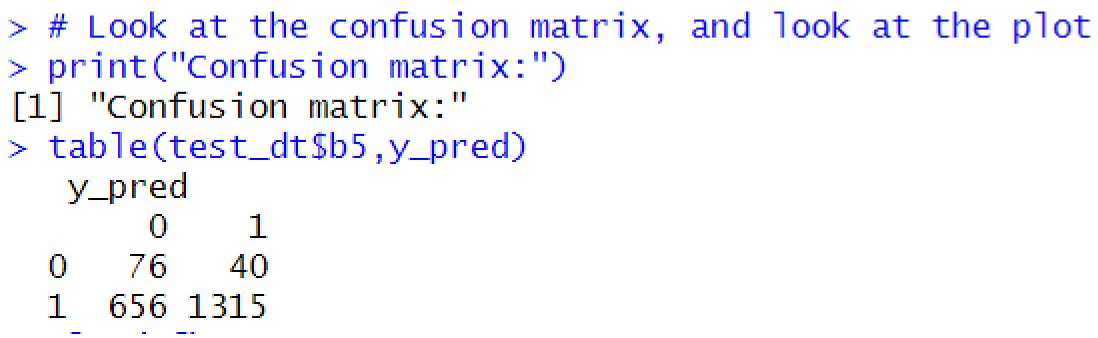
The reason for why the specificity is so high once again is the same as the reason for it in the logistic regression: the test data represent children that are alive much more since it has not been downsampled like the training data.

Overall, the kNN model also shows little promise. Our results seem to be in line with the paper once again.

Now, we will look at the results we got from the random forest method. This is the graph for the error and the variables in the random forest trees model.

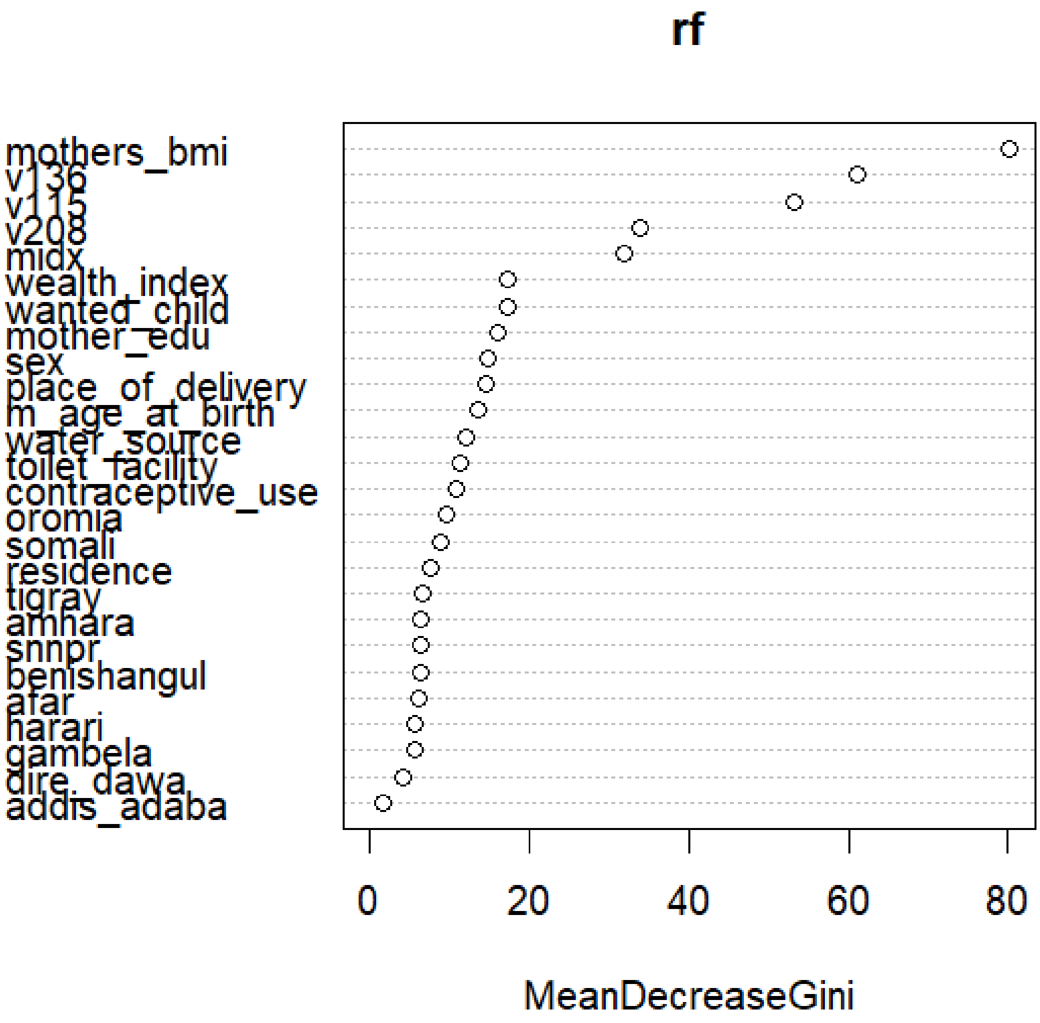


The Test dataset gave the following confusion matrix after predicting.



```
> # Look at the confusion matrix, and look at the plot
> print("Confusion matrix:")
[1] "Confusion matrix:"
> table(test_dt$b5,y_pred)
   y_pred
       0    1
  0   76   40
  1  656 1315
```

We now look at the variables which show importance in the random forest model



We see that the mothers_bmi has the highest mean decrease in gini value, which shows that it is the most significant variable, whereas the most insignificant variable according to the model is whether the person belongs to the place Addis Ababa.

The model's accuracy ***was predicted to be 66.49% which is almost similar to the paper, which predicted an accuracy of 67.2%.***

5. Conclusion

Overall, our models have more or less the same accuracy, specificity, and other indicators as the paper. However, the differences from the paper mainly arise in the variables. The paper is slightly vague in its methods of acquiring the variables they use. Due to this, we can't replicate the exact variables that the paper uses. For example, the variable for whether a health facility has the ability to perform C-section deliveries. There is no indication in the DHS dataset, on DHS websites, or on WHO websites whether an institute has this facility. We also faced issues with NaN values as we had to drop a few independent variables as we could not train the model using KNN or Random Forests without dropping them. This might have introduced a bias.

For these reasons, while our results don't match the research paper exactly, we have succeeded at the objective of this assignment: getting hands-on training while being exposed to various research questions, data sets, and machine learning techniques.