WOMEN COUNT

ANNEX 2

INTEGRATING SURVEY AND GEOSPATIAL INFORMATION DATA FOR GENDER ANALYSIS

TRAINING SYLLABUS

Curriculum on Gender Statistics Training

This product was developed under the guidance of the Subgroup on Gender Statistics Training, within the Asia-Pacific Network of Statistical Training Institutes.

UN WOMEN

Introduction

This syllabus has been designed to guide trainers on how to conduct training on integrating statistical and geospatial information. The syllabus can also be used by trainees who wish to know more about this topic and people who are generally interested in analyzing gender data by integrating household survey data and geospatial data generated from Geographic Information Systems (GIS).

This syllabus is part of a wider module on this area of gender statistics. Other materials within this module might include exercises, sample data sets, Power Point presentations and example quizzes. Please refer to the additional set of materials for a comprehensive and effective learning experience.

Who is this module for?

- Statisticians and geospatial experts who wish to conduct geospatial statistical analysis with a gender angle.
- Policymakers and decision-makers who wish to utilize geospatial covariates to determine the impact of location on gender policy outcomes.
- Academics who wish to use to this module as teaching material to demonstrate the integration of statistical and geospatial information for gender analysis in classrooms.
- Anyone who is interested in using geospatial covariates, which can be linked to household survey data sets, to analyze gender data.

What do I need to know before going through this module?

This is an advanced module on gender statistics mainly targeted to applied statisticians and gender policy analysts involved in gender data analyses. No advanced knowledge on geospatial statistics is necessary. However, it would be helpful for the trainee to have basic knowledge of statistical estimation, basic geospatial concepts and statistical modelling.

Learning objectives

The expected learning outcomes for this module include:

- Learners will understand more clearly that the integration of geospatial information to household survey data provides new sources for filling data gaps for the gender related United Nations' Sustainable Development Goals (SDGs).
- This module attempts to provide basic skills in reading microdata from household surveys (Demographic and Health Survey), extracting geostatistical information from a GIS data file (e.g. raster data format) and merging the two to create a consolidated data set for further gender analysis.
- The module also presents comprehensive steps to estimate the impact of various geo-covariates on the gender-related response variable of interest (e.g. child marriage before age 15) in the household survey.

- The module attempts to introduce basic coding skills (R statistical software) related to analyzing statistical models (e.g. Random Forests, logistic regression) and extracting geostatistical information from different types of GIS data files, which are freely available on the Internet.

Note to trainer: This module assumes that trainees do not have prior experiences in using R and applying statistical models and geospatial concepts. Depending on the trainees' familiarity with R or other statistical software and knowledge in statistical modelling and geospatial concepts, it is expected that training for this module can be delivered in 2 to 3 hours. Learners should refer to the list of existing resources provided in this module for deeper understanding of theoretical assumptions on statistical and geospatial methodologies introduced in this module.

1. Introduction to integrating household survey and geospatial data

1.1 Why do we need data integration?

The monitoring of the goals and targets of the *2030 Agenda for Sustainable Development* requires reliable and timely data disaggregated by sex, geographic location and other socioeconomic characteristics. This has greatly increased the demand for data. With the advancement in GIS technologies, more governments are now encouraged to use open-source GIS software, including publicly available geospatial data, to spatially analyze data that can help them achieve the data requirements of SDGs.

The statistical community is beginning to understand the benefits and need for geographic location in conducting statistical analysis. For example, the World Bank has integrated Living Standards Measurement Survey data and geospatial information to create poverty maps in several countries. The poverty maps are useful tools to visualize and compare poverty rates across geographic areas (district and subdistrict levels)[1] disaggregated by sex. By integrating geospatial information with population data, reliable and granular data can be visualized to easily identify those areas or vulnerable groups in greatest need.

1.2 Women and environment

The integration of survey and geospatial data can provide powerful insights, particularly related to the environment. Integrating georeferenced disaster data with disaggregated socioeconomic data can provide useful inputs to evidence-based policymaking and effective disaster management[2]. Such an analytical approach is helpful to assessing the different impacts of disasters (such as drought, flood, desertification, etc.) on women and men. Besides the immediate effects of such disasters, there are often longer-term indirect impacts. For example, the stress and trauma experienced by survivors from disasters associated with natural hazards could intensify violence against women, as shown in a recent OHCHR report[3]. After two tropical cyclones hit Vanuatu in 2011, the Tanna Women's Counselling Centre reported a 300 per cent increase in new domestic violence cases. As highlighted above, integrating geospatial information and survey data can provide more insights for understanding the multidimensional linkages between gender and environment statistics.

2. Understanding child marriage using geo-covariates

2.1 Child marriage

Child marriage (marriage before the age of 18) is a fundamental violation of human rights. Girls who marry before they turn 18 are less likely to remain in school and more likely to experience domestic violence. Despite laws against child marriage, 650 million girls and women alive today were married as children[4]. Several factors such as education, wealth and other socioeconomic variables are known to influence the probability of getting married before 15 or 18. How about environment factors? Do

[1] The World Bank. https://www.worldbank.org/en/topic/measuringpoverty#3
[2] UN ESCAP. https://www.unescap.org/our-work/ict-disaster-risk-reduction/space-based-data-disaster-risk-reduction/about
[3] UN Women. https://www.unwomen.org/en/news/in-focus/end-violence-against-women/2014/environment
[4] UNICEF. https://www.unicef.org/stories/child-marriage-around-world

environment-related factors such as drought episodes, level of aridity and urbanization have any impact on child marriage?

Integrating household survey data and environment-related geo-covariates provides opportunities to use geospatial statistical models to analyze, at a detailed level, the impact of such environment related geo-covariates on child marriage.

For illustration purposes, this training module uses microdata from Demographic and Health Surveys (DHS) in Bangladesh and several environment-related geo-covariates that are identified as potentially useful predictors of child marriage. In addition, the machine-learning 'Random Forests' algorithm and logistic regression model are used to identify which predictors are significant predictors of child marriage outcomes.

2.2 Demographic and Health Surveys and geospatial covariate data sets

The DHS Bangladesh 2014 is stratified (urban and rural) and selected in two stages. It uses a sampling framework from the list of enumeration areas (EAs) of the 2011 Population and Housing Census of Bangladesh. Some 600 EA clusters are selected for this survey.

The DHS Bangladesh 2014 provides useful demographic and socioeconomic variables (e.g. age, sex, age at first marriage, marital status, education, wealth index, etc.) that can be used to analyze the impact of each of these variables on child marriage. In addition to the survey variables in the DHS Bangladesh 2014, the focus of this training module is to integrate various environment-related geo-covariates (see table below).

Table 1: Geo-covariates used in analyzing child marriage from the DHS Bangladesh 2014

Geo-covariates	Definition	Data source link
Travel_Times2015	Travelling time (in minutes) to the nearest city of more than 50,000 people	https://map. ox.ac.uk/wpcontent/uploads/ accessibility/ accessibility_to_ cities_2015_v1.0.zip
SMOD2015	Degree of urbanization	http://cidportal. jrc.ec.europa.eu/ ftp/jrc-opendata/ GHSL/GHS_SMOD_POP_ GLOBE_R2016A/
Buildup2015	Percentage of building footprint area in relation to the total cell area.	http://cidportal. jrc.ec.europa.eu/ ftp/jrc-opendata/ GHSLGHS_BUILT_LDSMT_ GLOBE_R2015B/
Aridity2015	Climate data related to evapotranspiration processes and rainfall deficit for potential vegetative growth. Higher index suggests higher humidity.	https://cgiarcsi. community/data/ global-aridity-and-pet-database/
Density2015	Number of inhabitants per cell (1km X 1km)	http://cidportal. jrc.ec.europa.eu/ ftp/jrc-opendata/ GHSL/GHS_POP_GPW4_ GLOBE_R2015A/
aIncome2013	Estimates of income in USD per grid square	https://www.worldpop.org/doi/10.5258/SOTON/WP00020
aPP2013	Mean likelihood of living in poverty per grid square	https://www.worldpop.org/doi/10.5258/SOTON/WP00020
aWealthIndex2011	Mean DHS wealth index score per grid square	https://www.worldpop.org/doi/10.5258/SOTON/WP00020

DHS surveys collect GPS location data of surveyed clusters. The coordinates of each cluster, along with other geographic information (e.g. whether the cluster is urban or rural), is stored in the GIS shapefile (see section 3.1 for the details). The shapefiles in the DHS allow us to link the values of the geo-covariates by each cluster in DHS surveys. The shapefiles are available upon request for download through the DHS programme website, following the application of geospatial displacement on the GPS cluster data to protect the confidentiality of respondents[5].

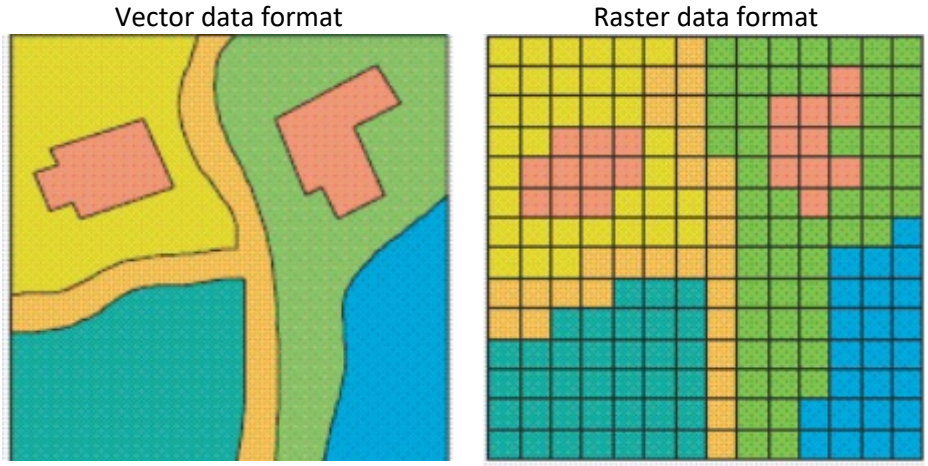
3. Integrating DHS and geospatial data

3.1 Working with GIS data

Many of the GIS data are stored in two data formats called “vector” and “raster”. The shapefile format in the DHS Bangladesh 2014 is a vector data format, which define the boundaries and shape of clusters in DHS. Vector data are represented by points (e.g. sampling locations, individual trees), lines (e.g. roads, streams), and polygons (e.g. lake, oceans).

The GIS data format of geo-covariates used in this module is a raster data format. Raster or "gridded" data are stored as a grid of values which are rendered on a map as pixels. Each pixel value represents an area on the Earth’s surface. Raster data can be continuous or categorical. Continuous raster format data files can have a range of quantitative values (e.g. population density, poverty index measuring likelihood of living in poverty per grid square). Vector and raster format data files are illustrated in Figure 1.

Figure 1: Comparison of vector and raster data format



Source: ESRI. 2019. www.esri.com

As mentioned above, the location of clusters in the DHS Bangladesh 2014 are stored in vector format data and they are represented as points, as shown in Figure 2. We can extract values from raster-format data and lay on the vector data format for visualization purposes. For example, the World Bank created a poverty map for Bangladesh by storing the estimates of mean likelihood of living in poverty per grid square in raster format data. These poverty estimates are extracted from the raster data and projected on to the cluster location map (shapefile from DHS) as shown in the background colour in Figure 3. The black dots represent the clusters, the green colour means low risk of poverty and pink

[5] Each of the clusters was displaced from the actual location by up to 2 kilometres (for urban points) and 10 kilometres (for rural points).

means high risk of poverty. It visualizes the location of clusters and risk of living in poverty on the same map.

Figure 2: Cluster locations by urban and rural from the shapefile (DHS Bangladesh 2014)

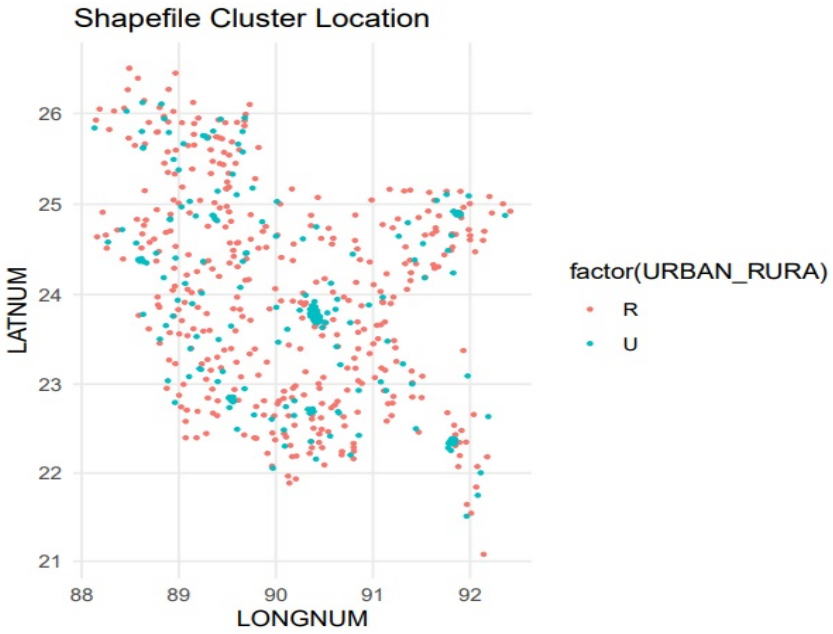
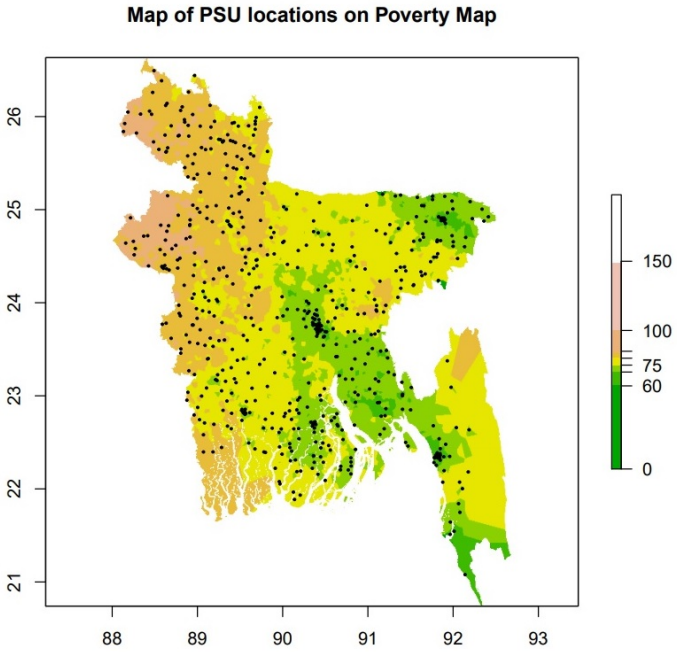


Figure 3: Map of PSU locations on poverty map from the raster file



3.2 Extracting values from a raster

As discussed in 3.1, the ultimate goal of the data integration exercise in this module is to extract statistical values stored in the raster file and link them to the respective cluster locations stored in the DHS shapefile, which is a vector format data. For both the shapefile and raster data file, a coordinate

reference system maps each point/pixel to a precise location on earth. In the shapefile, it is called "proj4string", and in the raster file, it is called "coord. ref.". The "proj4string" in the shapefile and "coord. ref." information in the raster file should be the same in order to project the statistical values onto a correct location in the DHS shapefile.

As an illustration, Table 2 shows the coordinate reference system in a shapefile and a raster file. A coordinate reference system consists of two parts: a geographic coordinate system that identifies any location on the earth's surface using two values – longitude and latitude – and a projection that converts the three-dimensional earth's surface into a two-dimensional Cartesian map represented in the data. Here for both spatial data sets, their geographic coordinate system uses "WGS 84", short for World Geodetic System, 1984 revision. And they used two different projections. The shapefile (vector format) uses latlong, which means the coordinates are the latitude and longitude of the places and no projection is needed. The raster file uses "moll"[6]. For this example, in Table 2, we need to transform the shapefile data to have an identical coordinate reference system with the raster file so that we can find the exact places for the clusters in the raster file. This transformation can be done using "spTransform" package in R (see Annex for R codes).

Table 2: Map of PSU locations on poverty map from the raster file

Type of class	Coordinate reference system
Vector (SpatialPointsDataFrame)	Proj4string: +proj = longlat + datum = W GS84 + nodefs + ellps = W GS84 + towgs84 = 0, 0, 0
Raster (RasterLayer)	Coord. Ref.:+proj = moll + lon 0 = 0 + x 0 = 0 + y 0 = 0 + ellps = W GS84 + units = m + no defs

4. Logistic regression and Random Forests

4.1 Model building

After we extract the statistical values of geo-covariates stored in the raster files, we need to merge them with DHS survey data. The DHS Bangladesh 2014 has a variable on "Age at first marriage". We can use this variable as the response variable of our interest. We can use this variable to create another column that indicates whether an individual woman or girl is married before 15 or 18. As for the independent variables, we can include the variables "Age" and "Education" from the DHS surveys and the eight geo-covariates listed in Table 1 to assess the impact of each independent variable on the response variable. Using data from the 600 clusters (DHS Bangladesh 2014), we will try to build a model that can predict the outcome of the response variable (child marriage) based on the set of independent variables mentioned above.

In this training module, we use logistic regression and Random Forests, which is a machine-learning-based algorithm to predict the outcome (marriage before 15) given a set of independent variables. Logistic regression is a traditional statistical method that is mainly focused on estimating the coefficients and the significance of the independent variables in the model. This allows us to explain

[6] This is a short name for the Mollweide projection. For further details, refer to Geocomputation with R (https://geocompr.robinlovelace.net/spatialclass.html)

and understand how the change in independent variables impacts the response variable of our interest. We can also perform hypothesis-testing based on well-established statistical theories over the centuries. Although logistics regression is also capable of predicting the outcome of the response variable[7], it is mainly an estimation- or inferencing-focused statistical method.

Unlike logistic regression models, machine-learning methods, such as Random Forests, are prediction-focused. Their primary interest is to increase the accuracy of the predicted outcome using the given input data. The process of how the model transforms inputs into outputs is, to most non-experts, a black box. Nevertheless, we can get a general idea of which variables used in the Random Forests predictive models are significant in predicting the outcome.

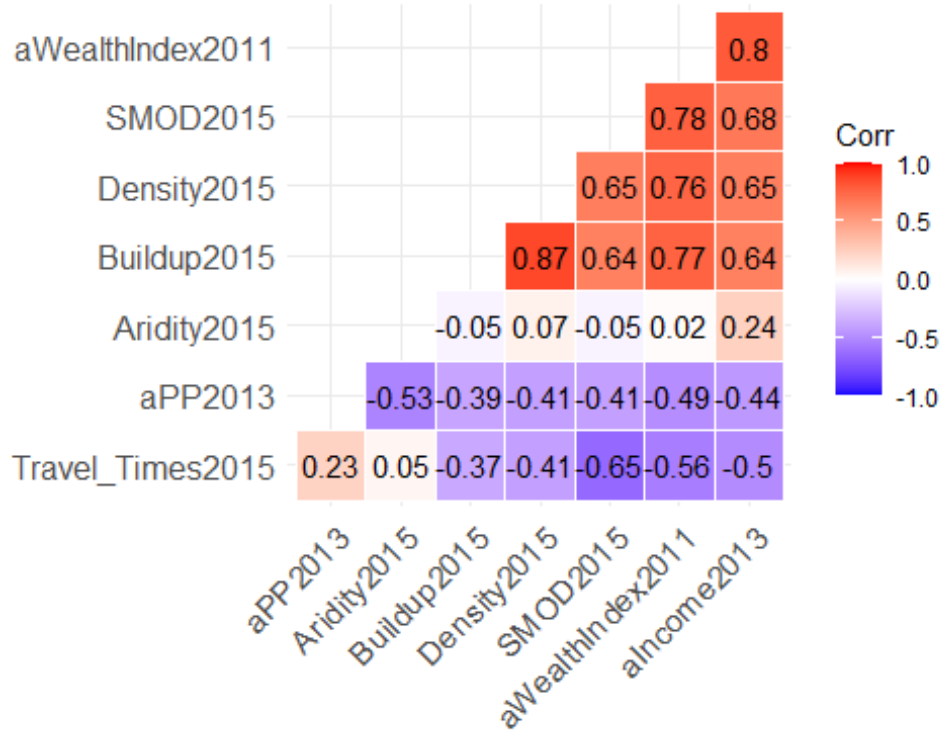
We will use both logistic regression and Random Forests to build predictive models and explain the impact each predictor variable has on the response variable. We can also compare the results of each method to strengthen and confirm our analytical results.

4.2 Correlation between the geo-covariates

Before we begin our statistical analysis, we can look at the general relationships of the independent variable by calculating the correlation matrix. Using R, we can easily create a correlation plot for the eight geo-covariates to investigate whether there is any relationship between them. Is the level of aridity ("Aridity2015") related to the mean likelihood of living in poverty ("aPP2013")? If there is a relationship between them, is it a positive or a negative one? Correlation is a useful numerical summary of the strength of a relationship between two variables ranging from -1 to 1. The correlation plot in Figure 4 shows that wealth-related variables are highly correlated to each other ("aIncome2013", "aWealthIndex2011", "aPP2013"). We can also see that urbanization-related variables ("SMOD2015", "Density2015", "Buildup2015") are correlated to each other and also correlated to the wealth-related variables. This is intuitive, since wealth is concentrated in urban areas. It is interesting to note that the level of aridity ("Aridity2015") is negatively related to the mean likelihood of living in poverty, which corresponds to the findings in the logistic regression model in 4.2.1.

[7] For this reason, some argue that logistic regression can also be considered a machine-learning method.

Figure 4: Correlation plot of geo-covariates



4.2 Logistic regression

4.2.1 Fitting a logistic regression model to the DHS

The response variable "Before15" gets coded either 0 or 1. If "Age at first marriage" is less than 15, then it gets coded 1 and otherwise it gets coded 0. Logistic regression is used because our response variable is a categorical variable which has a measurement scale consisting of a binary set of categories (e.g. yes or no, 1 or 0) instead of a continuous response variable. As noted in 4.1, we will use age and education variables from the DHS survey and geo-covariates in Table 1 as independent variables to estimate the probabilities that the variable "Before15" is equal to 0 or 1. Learners are encouraged to refer to module 6 (Analyzing data with a gender angle) of the Gender Statistics Training Curriculum for more details on logistics regression.

A logistic regression model is fitted to the DHS survey data using the "glm" (generalized linear model) function in R. The output summary of the logistic regression is shown in Figure 5.

The p-value[8] in Figure 5 suggests that variables "Age", "Education", "Aridity2015", "aIncome2013" and "aPP2013" are significant. The odds ratio (OR) shows the proportionate change in odds[9]. If the OR is greater than 1 then it indicates that, as the predictor increases, the odds of the outcome occurring increase. If it is less than 1, it indicates that as the predictor increases, the odds of the outcome occurring decrease. Therefore, the OR (0.515) of "Aridity2015"[10] indicates that as "Aridity2015" increases (more humid) the odds of getting married before 15 decreases by 0.515. That is, with more

[8] As a rule of thumb, a p-value less than 0.05 is statistically significant.
[9] Change in odds = odds after a unit change in the predictor/original odds
[10] Note that the variable Aridity2015 represents the Aridity Index. That is, the ratio between precipitation and ETO (rainfall over vegetation water). Thus, its values increase for more humid conditions, and decrease with more arid conditions.

humid conditions (high aridity index), child marriage decreases. In turn, with drought episodes and less humid conditions (low aridity index) child marriage increases. Similarly, as the level of "Education" increases, the odds of getting married before 15 decrease significantly (the baseline is "No education"). Also, as the mean likelihood of living in poverty per grid square increases, the odds of getting married before 15 increases.

Figure 5: Logistics regression output (Response variable: Before15)

	Estimate	Std. Error	z value	Pr(>|z|)	OR	
(Intercept)	-0.773	0.387	-1.998	0.046	0.462	*
Age	0.029	0.002	16.096	<0.001	1.030	***
Education: Incomplete primary	-0.131	0.049	-2.684	0.007	0.878	**
Education: Complete primary	-0.339	0.056	-6.021	<0.001	0.713	***
Education: Incomplete secondary	-0.704	0.047	-15.081	<0.001	0.495	***
Education: Complete secondary	-2.079	0.105	-19.886	<0.001	0.125	***
Education: Higher	-3.204	0.118	-27.139	<0.001	0.041	***
Aridity2015	-0.663	0.057	-11.565	<0.001	0.515	***
Buildup2015	-0.006	0.224	-0.027	0.979	0.994	
Travel_Times2015	-0.001	0.002	-0.331	0.74	0.999	
SMOD2015	0.009	0.028	0.319	0.75	1.009	
Density2015	0.000	0.000	1.952	0.051	1.000	.
aWealthIndex2011	-0.078	0.064	-1.230	0.219	0.925	
aIncome2013	-0.003	0.001	-3.763	<0.001	0.997	***
aPP2013	0.015	0.004	3.515	<0.001	1.015	***

4.2.2 Assessing the prediction power of the logistic regression model

As for assessing the prediction power of the logistics regression model, we can take a look at the confusion matrix (Figure 6). The confusion matrix shows how well the model predicted the outcome. If the model was perfect, we would only have elements on the diagonal of this matrix and 0's everywhere. In total, the model misclassified (`1426 + 4442)/21262` = 29% of the outcome, which yields an accuracy of 71 per cent (100%-29%).

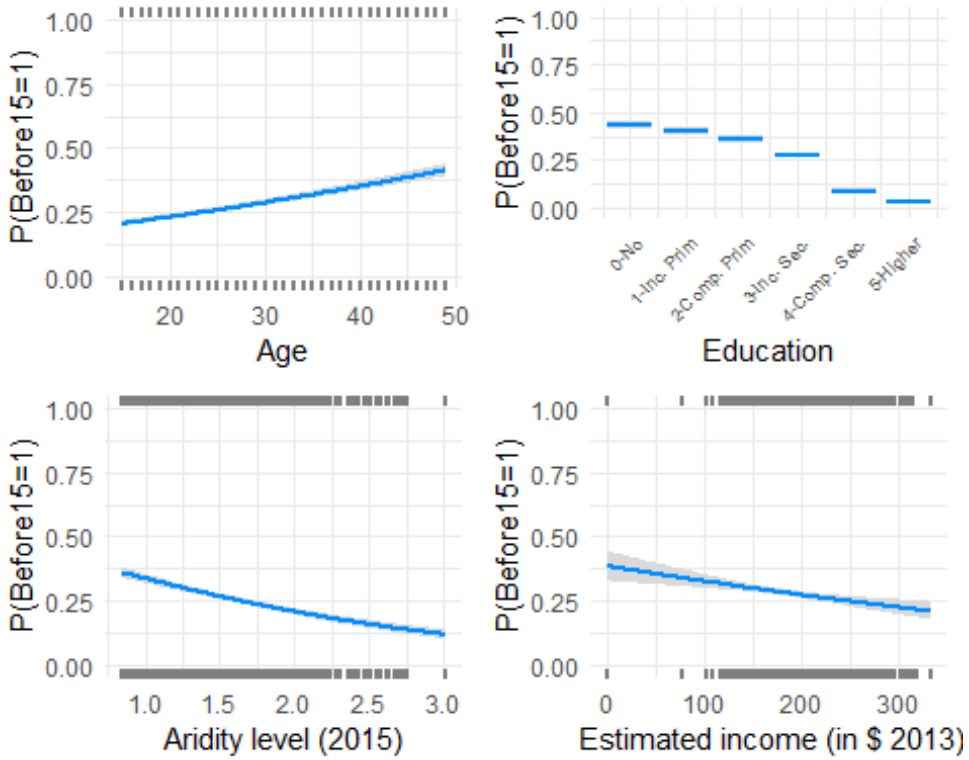
Figure 6: Confusion matrix

```
##          Predicted 0 Predicted 1 Total
## Actual 0       13499        1426 14925
## Actual 1        4442        1895  6337
## Total          17941        3321 21262
```

4.2.3 Visual representation of the logistic regression model

We can also visualize the effect of some of the most significant variables in the model. In Figure 7, we can see the marginal effects[11] of significant predictors in the model. The probability of being married before 15 increases with age. This shows the higher prevalence of the practice in the past. As the level of education increases, the probability decreases. We can notice a clear drop in the probability between the category "incomplete secondary" and "complete secondary". As the aridity (Aridity2015) level increases, the probability decreases. As the level of income (aPP2013) increases, the probability decreases.

Figure 7: Marginal effects of the logistic regression model



4.3 Random Forests

4.3.1 Brief introduction to Random Forests

For the purpose of building a predictive model in 4.1, this module uses a machine-learning technique called Random Forests. Random Forests is a type of tree-based learning algorithm that uses a collection of decision trees to perform classification or regression tasks – for example, classifying a person as 'high risk of marriage before 15' or 'low risk of marriage before 15'. The algorithm is frequently chosen to solve complex prediction tasks. As explained in 4.1, Random Forests is a prediction-focused model that aims to classify the outcome using sets of available input (independent variables) data. Unlike the logistic regression model, Random Forests does not provide significance statistics (such as coefficients, p-value, etc.) on each independent variable used in the model. However,

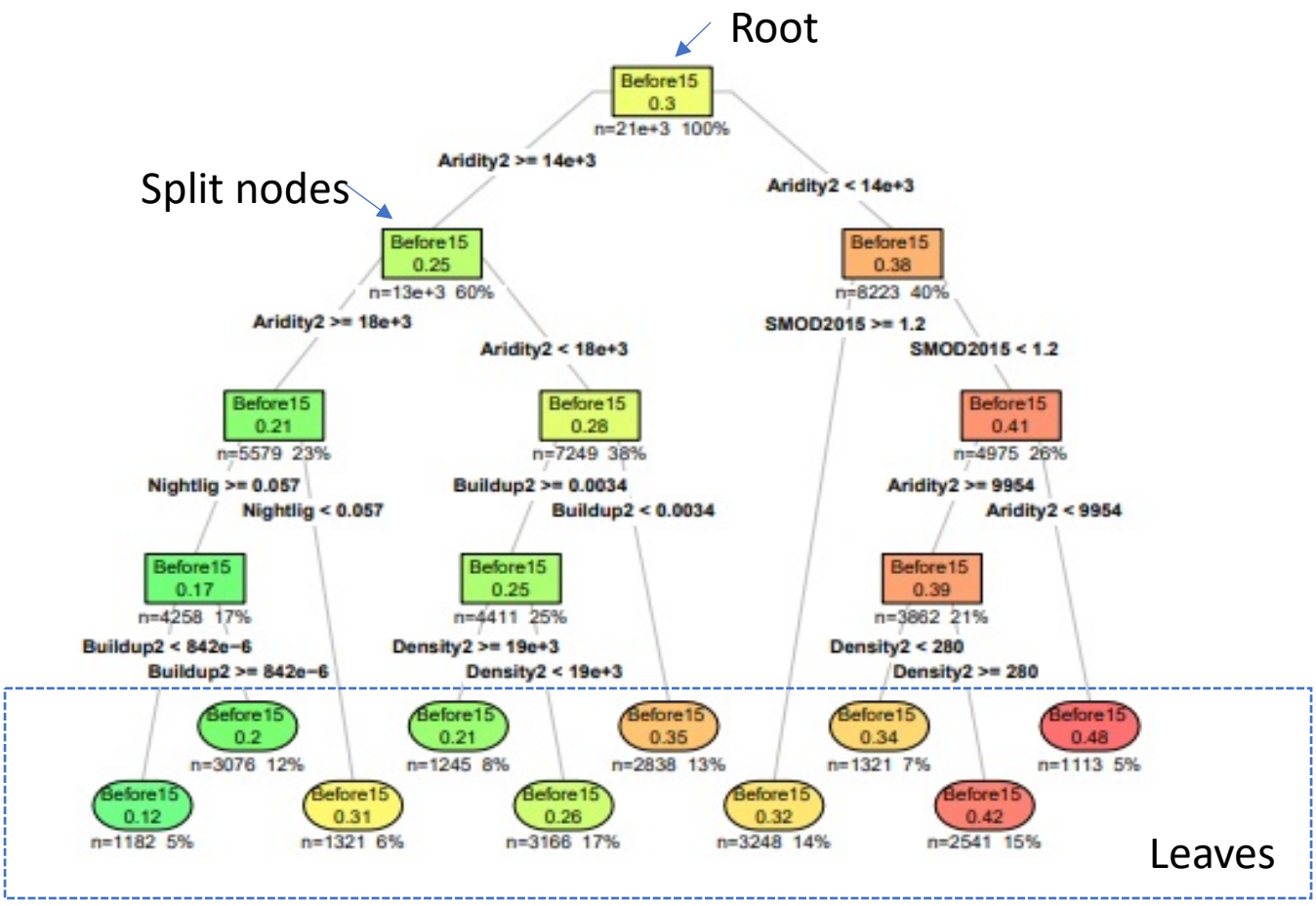
[11] The curves represent the probability of being married before 15 as a function of one variable, holding other variables in the model to their median value.

it does provide information on which variables are important and effective in classifying the outcomes into similar categories.

As mentioned before, Random Forests is basically a collection of many decision trees. The number of decision trees are determined by the improvements made in the accuracy of the model. A decision tree has a root, split nodes and terminal nodes, also known as leaves (see Figure 8). Rather than considering a single decision tree, Random Forests creates a wide variety of trees by using a bootstrapped sample from the original data and considering only a subset of the variables for split nodes until it reaches the terminal nodes. The variables for split nodes are randomly chosen from a subset of variables (e.g. Aridity2, SMOD2015) that minimize the variance in the child nodes. By repeating this process many times (n=500 by default in R), this results in a forest of different decision trees.

For example, in Figure 8, we can observe that “Aridity” is selected in a few of the split nodes, which further branches all the way down to the leaves. This suggests that “Aridity” is an important variable in splitting the data. On the leaves, we have very different levels of probabilities (different colors) for child marriage rates (before15 = 1), ranging from 0.12 to 0.48, depending on which "branches" (variables) and thresholds are applied to the leaf. The structure of the tree provides many insights on which variables influence the probability of marriage before 15 and in which direction (higher or lower).

Figure 8: A single decision tree for predicting marriage before 15



Random Forests feeds test data to all decision trees, which leads to a large number of results and by majority vote or by averaging the tree predictions at the terminal nodes, the algorithm maps input data to the predicted outcome[12].

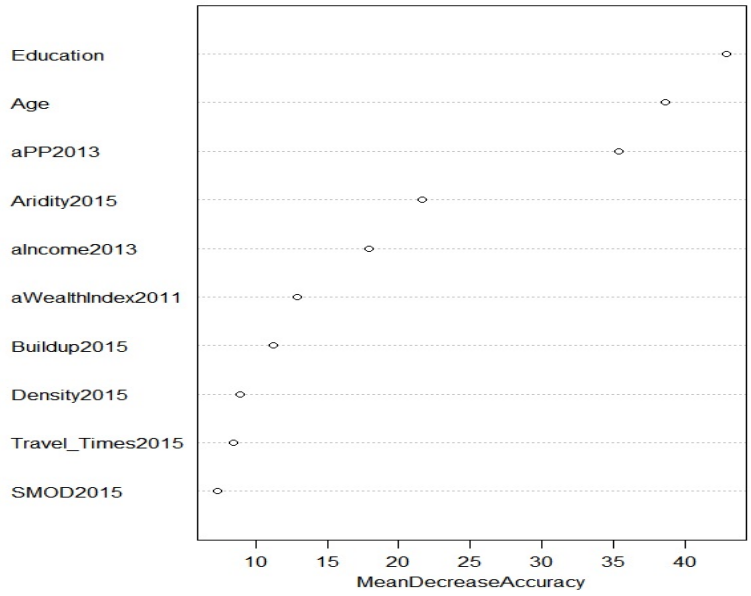
4.3.2 Applying Random Forests to understand child marriage using geo-covariates

We use "randomForest" package in R to fit a Random Forests model. Figure 9 shows the error rate with a confusion matrix. The default number of trees is set to 500 in R, but we can also set the number of trees of our choice (ntree=1000). The error rate of our model is 27.44 per cent, which means the accuracy of the model is about 73 per cent. This is slightly higher than the logistic regression model in 4.2.2 (71 per cent).

Figure 9: Accuracy rate and confusion matrix

```
##
## Call:
##  randomForest(formula = as.formula(formula_string), data = DataMerged1,
   ##  importance = T, maxnodes = 25, ntree = 1000, na.action = na.roughfi
x)
##                Type of random forest: classification
##                      Number of trees: 1000
##                No. of variables tried at each split: 3
##                OOB estimate of  error rate: 27.44%
## Confusion matrix:
##       0    1 class.error
## 0 13856  1131   0.0754654
## 1  4729  1636   0.7429694
```

Figure 10: Variable importance plot



Using the "varImPlot" function in R, we can find out which variables play an important role in the model. The variable importance plot (Figure 10) basically shows the mean decrease in accuracy when

[12] For more details on the theoretical background of Random Forests, see "A Systematic Approach for Variable Selection with Random Forests: Achieving Stable Variable Importance Values" https://ieeexplore.ieee.org/abstract/document/8038868

we remove each variable while making the decision tree. As shown in Figure 10, the first four important variables are “Education”, “Age”, “aPP2013” and “Aridity 2015”. This is also consistent with the result we get from the logistic regression in 4.2.1. where these variables were also significant.

5. Conclusion

The results from the logistic regression and Random Forests are consistent and suggest that “Education”, “Age”, “aPP2013” and “Aridity 2015” are important variables in explaining and predicting the outcome of marriage before 15 in Bangladesh. It is intuitive that “Education” empowers women and it significantly reduces the probability of being married before 15. As noted in the logistics regression, age is positively related to the probability of being married before 15. This suggests that child marriage is relatively more prevalent in the past, which might suggest relative improvements over recent years. The level of aridity, which measures humidity and rain fall, is significant in predicting child marriage before 15. Further research is needed to provide insights on these findings.

6. KEY TAKEAWAYS

- Integrating survey data and geo-covariates that are publicly available can provide new information and insights to fill data gaps for the SDGs and policy formulation.
- GIS data are stored in two data formats called “vector” and “raster”. When integrating different formats of GIS data files, the coordinates of each GIS data file must be checked before the integration and needs to be transformed so they are aligned on a same coordinate reference system.
- Logistic regression is a traditional statistical method that is mainly focused on estimating the coefficients and the significance of the independent variables in the model.
- Machine-learning methods, such as Random Forests, are prediction-focused methods. Their primary interest is to increase the accuracy of the prediction outcome using the given input data.

Annex 1: R code used in this module

1. Introduction to integrating household survey and geospatial data

We need a minimum of organization in the data and code folders, as well as some R packages.

```
# GIS packages
library(raster) ## for reading "RASTER" files
library(rgdal)  ## for reading "shapefiles"
library(sp)     ## for adjusting CRS in

# Tidy data management packages
library(dplyr)
library(data.table)

# Plotting packages
library(ggplot2)
library(RColorBrewer)

# Nice presentation of results
library(knitr)
library(papeR)

# --- Change to YOUR project folder HERE  ---- #
source_folder<-"c:/GitMain/UNWomen-GISBangladesh/"

# Specific sub-folder for data storage

shapedata_folder<-paste(source_folder, "dhsdata/BDGE71FL", sep="")
geodata_folder<-paste(source_folder, "geodata/", sep="")
data_folder<-paste(source_folder,"Data/",sep="")

# this is where all saved .Rda go and can be loaded when needed
output_folder<-paste(source_folder, "CreatedData/" , sep="")
```

2. Understanding child marriage using geo-covariates

The DHS survey

```
# Reading DHS survey data
merged1<-read.csv(file = 'bangladesh.csv')  # reading DHS Bangladesh 2014
merged1$Age<-as.numeric(merged1$Age)

#Computing the proportion of getting married before 15 by cluster
cluster_average<-aggregate(Before15~DHSCLUST,
                           data=merged1,
                           FUN=mean)
```

3. Integrating DHS and geospatial data

Reading the DHS Shapefile

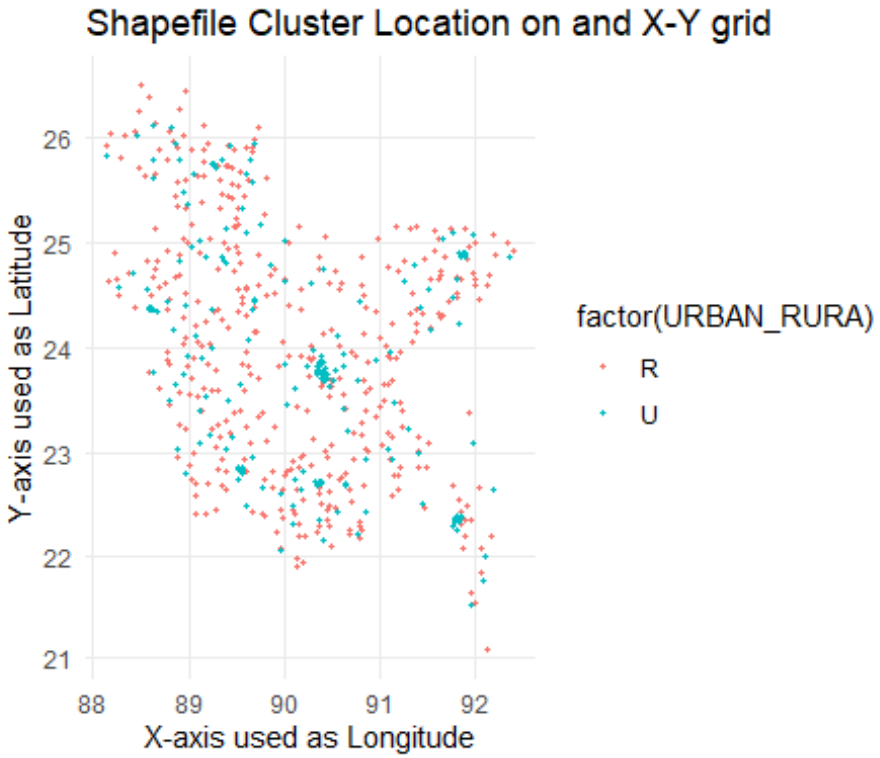
```
# Reading DHS Shapefile
dhsShapeData<-readOGR(shapedata_folder, "BDGE71FL") # Reads the shapefile in DHS
shapedata<-dhsShapeData@data                        # Reads the data part
shapefile_df <- fortify(shapedata)
shapedata<-shapedata[shapedata$LATNUM>0, ]          # Drops negative Latnum
```

Cluster locations by urban and rural from the shapefile (DHS Bangladesh 2014)

We can use the *latitude* and *longitude* of each observation to draw a "location map" of the clusters. This generates the **Figure 2** of the document.

This is not really "a map", but only points with latitude and longitude defined represented on a grid.

```
# Now the shapefile can be plotted, as points
# In the aesthetics part of the ggplot we need long, lat,
#                       and we use group for Urban or Rural.
map <- ggplot() +
  geom_point(data = shapedata,
             aes(x = LONGNUM, y = LATNUM, color = factor(URBAN_RURA)),
             size = .6) +
 ggtitle("Shapefile Cluster Location on and X-Y grid") +
         labs(x=  "X-axis used as Longitude" ,
              y = "Y-axis used as Latitude")

# Using the ggplot2 function coord_map will make things look better
# and it will also let you change the projection.
map_projected <- map +
  coord_map()+
  theme_minimal()

map_projected
```



Reading raster files

```
# Reading geographic data - access to cities
accessData<-raster(paste(geodata_folder,
                         "accessibility_to_cities_2015.tif",
                         sep=""))
accessData

#  We need to provide the same  projection reference
#  We use sp::spTransform to tell wich CRS is used.
```

```
dhsShapeData2 <- spTransform(dhsShapeData, accessData@crs)
```

Extracting values from a raster

We can now **extract** the values from the *accessData* file (a Raster object) at the locations of our household (shapefile). The result is a data frame. The first column is a sequential ID, the other columns are the extracted values, i.e. the travel time to a city for each cluster.

Extracting values takes time.

```
# Data extraction using the matching between raster and Spatial data

# CAUTION: In the following chuncks, the data.fram
#  dhs_all2000 is used as a generic data.frame (temporary)

dhs_all2000 <- raster::extract(accessData,     # raster layer
                               dhsShapeData2,  # SPDF with centroids for buffer
                                buffer = 2000, # buffer size (meters)
                                df=TRUE)       # returns a dataframe

dhs_all2000<-as.data.frame(dhs_all2000)

# Filtering to remove na values and distances equal to 0.
dhs_all2000<-dhs_all2000[!is.na(dhs_all2000$accessibility_to_cities_2015)
                         & dhs_all2000$accessibility_to_cities_2015>=0, ]

# Aggregation (mean of the travel times for each cluster)
# Name changed here to avoid erasing row data: acessData --> accessData.agg
accessData.agg<-aggregate(dhs_all2000$accessibility_to_cities_2015,
                          by=list(dhs_all2000$ID),
                          FUN=mean)
colnames(accessData.agg)<-c("DHSCLUST", "Travel_Times2015")

# Saving the file in a devoted folder
save(accessData.agg, file="CreatedData/accessData.Rda")
```

Importing other geographical information files

The exact same operations can be done with all the other geographic files we have identified (see **Table 1** of the document for a listing).

These operations take time and you want to skip these steps and directly upload the file created (see Section 4).

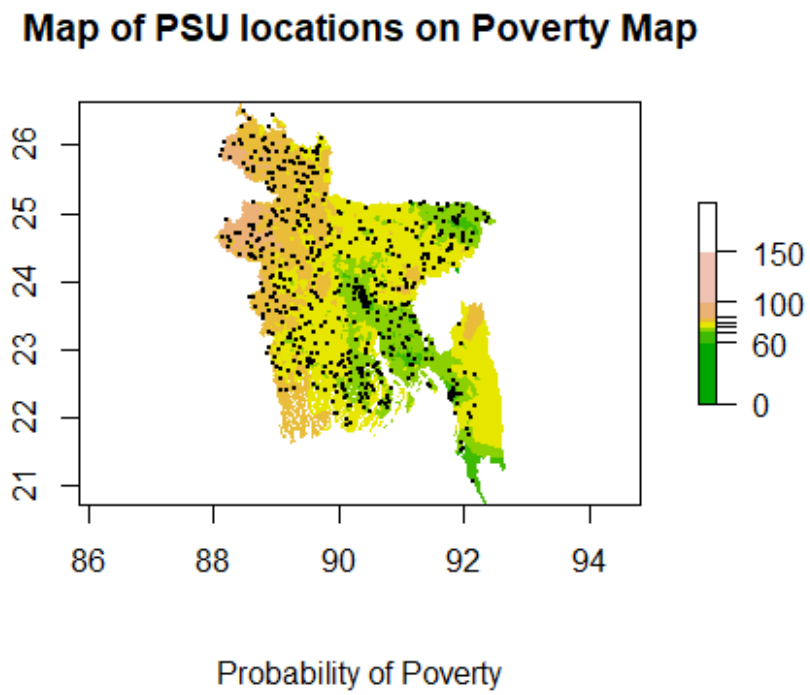
```
# Reading raster file for SMOD2015
smodData<-raster(paste(geodata_folder, "GHS_SMOD_POP2015_GLOBE_R2016A_54009_1k_v1_0.tif", s
ep=""))
dhsShapeData2 <- spTransform(dhsShapeData, smodData@crs)
dhs_all2000 <- extract(smodData,    # raster layer
                       dhsShapeData2,
                       buffer = 2000,
                       df=TRUE)
dhs_all2000<-as.data.frame(dhs_all2000)

smodData.agg<-aggregate(dhs_all2000$GHS_SMOD_POP2015_GLOBE_R2016A_54009_1k_v1_0,
                        by=list(dhs_all2000$ID),
                        FUN=mean)
colnames(smodData.agg)<-c("DHSCLUST", "SMOD2015")
save(smodData.agg, file="CreatedData/smodData.Rda")

# Reading raster file for Buildup2015
buildupData<-raster(paste(geodata_folder, "GHS_BUILT_LDS2014_GLOBE_R2016A_54009_1k_v1_0.tif
```

```
", sep=""))
dhsShapeData2 <- spTransform(dhsShapeData, buildupData@crs)
dhs_all2000 <- extract(buildupData,
                       dhsShapeData2,
                       buffer = 2000,
                       df=TRUE)
dhs_all2000<-as.data.frame(dhs_all2000)

buildupData.agg<-aggregate(dhs_all2000$GHS_BUILT_LDS2014_GLOBE_R2016A_54009_1k_v1_0,
                           by=list(dhs_all2000$ID),
                           FUN=mean)
colnames(buildupData.agg)<-c("DHSCLUST", "Buildup2015")
save(buildupData.agg, file="CreatedData/buildupData.Rda")

# Reading raster file for Density2015
densityData<-raster(paste(geodata_folder, "GHS_POP_GPW42015_GLOBE_R2015A_54009_1k_v1_0.tif"
, sep=""))
dhsShapeData2 <- spTransform(dhsShapeData, densityData@crs)
dhs_all2000 <- extract(densityData,
                       dhsShapeData2,
                       buffer = 2000,
                       df=TRUE)
dhs_all2000<-as.data.frame(dhs_all2000)

densityData.agg<-aggregate(dhs_all2000$GHS_POP_GPW42015_GLOBE_R2015A_54009_1k_v1_0,
                           by=list(dhs_all2000$ID),
                           FUN=mean)
colnames(densityData.agg)<-c("DHSCLUST", "Density2015")
save(densityData.agg, file="CreatedData/densityData.Rda")

# Reading raster file for aIncome2013
aICData<-raster(paste(geodata_folder, "bgd2013incpov.tif", sep=""))
dhs_all2000 <- extract(aICData,
                       dhsShapeData,
                       buffer = 2000,
                       df=TRUE)
dhs_all2000<-as.data.frame(dhs_all2000)
temp<-dhs_all2000[!is.na(dhs_all2000$bgd2013incpov), ]

aICData.agg<-aggregate(temp$bgd2013incpov,
                       by=list(temp$ID),
                       FUN=mean)
colnames(aICData.agg)<-c("DHSCLUST", "aIncome2013")
save(aICData.agg, file="CreatedData/aICData.Rda")

# Reading raster file for aPP2013
aPPData<-raster(paste(geodata_folder, "bgd2013ppipov.tif", sep=""))
dhs_all2000 <- extract(aPPData,
                       dhsShapeData,
                       buffer = 2000,
                       df=TRUE)
dhs_all2000<-as.data.frame(dhs_all2000)
temp<-dhs_all2000[!is.na(dhs_all2000$bgd2013ppipov), ]

aPPData.agg<-aggregate(temp$bgd2013ppipov,
                       by=list(temp$ID),
                       FUN=mean)
colnames(aPPData.agg)<-c("DHSCLUST", "aPP2013")
save(aPPData.agg, file="CreatedData/aPPData.Rda")
```

Map of PSU locations on poverty map from the raster file

This code generates **Figure 3** of the document.

```r
# Main plot using the plotting function of raster package
plot(aPPData,
     breaks=c(0, 60, 70, 75, 80, 85, 100, 150),
     col = terrain.colors(8),
     main="Map of PSU locations on Poverty Map",
     sub="Probability of Poverty")

# We can add points for each cluster location on this map
points(x=shapedata$LONGNUM,
       y=shapedata$LATNUM,
       type="p", cex=0.3, pch=21, bg=1)
```




```r
# Reading raster file for Aridity2015
memory.limit(9999999999)
aridityData <- raster(readGDAL(paste(geodata_folder, "AI_annual/ai_yr/w001001.adf", sep="")
))
dhsShapeData2 <- spTransform(dhsShapeData, aridityData@crs)
dhs_all2000 <- extract(aridityData,
                       dhsShapeData2,
                       buffer = 2000,
                       df=TRUE)
dhs_all2000<-as.data.frame(dhs_all2000)
dhs_all2000<-dhs_all2000[!is.na(dhs_all2000$band1),]

 aridityData.agg<-aggregate(dhs_all2000$band1,
                            by=list(dhs_all2000$ID),
                            FUN=mean)
colnames(aridityData.agg)<-c("DHSCLUST", "Aridity2015")
save(aridityData.agg, file="CreatedData/aridityData.Rda")

# Reading raster file for aWealthindex2011
aWIData<-raster(paste(geodata_folder, "bgd2011wipov.tif", sep=""))
dhs_all2000 <- extract(aWIData,
                       dhsShapeData,
                       buffer = 2000,
                       df=TRUE)
dhs_all2000<-as.data.frame(dhs_all2000)
```

```
temp<-dhs_all2000[!is.na(dhs_all2000$bgd2011wipov), ]

aWIData.agg<-aggregate(temp$bgd2011wipov,
                       by=list(temp$ID),
                       FUN=mean)
colnames(aWIData.agg)<-c("DHSCLUST", "aWealthIndex2011")
save(aWIData.agg, file="CreatedData/aWIData.Rda")
```

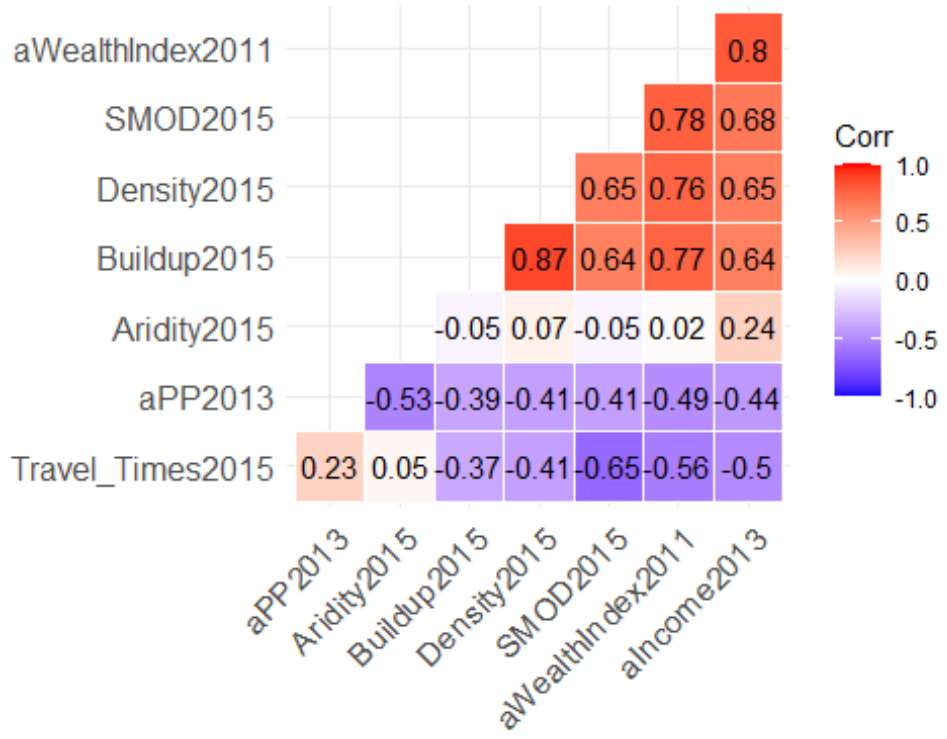
4. Logistic regression and Random Forests

Since the previous operation may take time and CPU resources, you can directly load the data sets created above and **start using the code here.**

```
### Loading  Geo-covariate for clusters ##
load("CreatedData/accessData.Rda")
load("CreatedData/smodData.Rda")
load("CreatedData/buildupData.Rda")
load("CreatedData/aridityData.Rda")
load("CreatedData/densityData.Rda")
load("CreatedData/aWIData.Rda")
load("CreatedData/aICData.Rda")
load("CreatedData/aPPData.Rda")

## Function used for merging geo-covariates to DHS data #
dhsdataMerge<-function(originalData){
  datause<-merge(originalData, accessData.agg, by=c("DHSCLUST"), all.x=T)
  datause<-merge(datause, smodData.agg, by=c("DHSCLUST"), all.x=T)
  datause<-merge(datause, buildupData.agg, by=c("DHSCLUST"), all.x=T)
  datause<-merge(datause, aridityData, by=c("DHSCLUST"), all.x=T)  ## NO .agg HERE because
you gave it to me already aggregated !!!
  datause<-merge(datause, densityData.agg, by=c("DHSCLUST"), all.x=T)
  datause<-merge(datause, aWIData.agg, by=c("DHSCLUST"), all.x=T)
  datause<-merge(datause, aICData.agg, by=c("DHSCLUST"), all.x=T)
  datause<-merge(datause, aPPData.agg, by=c("DHSCLUST"), all.x=T)
  datause<-datause[datause$DHSCLUST!=544,]
  return(datause)
}
# Using this function, we can merge the file cluster_average
# with all the Geo-covariate extracted at the cluster level
data.agg<-dhsdataMerge(cluster_average)
```

Correlation plot

```
library(ggcorrplot)

# We compute the correlation matrix of the covariates
corr_coef<-cor(data.agg[, c(3:10)],use = "p")
#And then plot it with nice options
ggcorrplot(corr_coef,
           type = "lower",         # lower triangle of the matrix only
           hc.order = TRUE,        # variable sorted from highest to lowest
           outline.col = "white",  #Color options
           lab = TRUE)
```



4.2 Logistic regression

```
# We use the dhsdataMerge function to merge the survey data (individuals)
# with all the Geo-covariate extracted at the cluster level
DataMerged1<-dhsdataMerge(merged1)

# We need to have a factor variable and not directly Before15 (that is numeric here)
DataMerged1$I_Before15 <- as.factor(DataMerged1$Before15)

# Education is a factor variable
DataMerged1$Education <- as.factor(DataMerged1$Education)
DataMerged1 <- DataMerged1 %>%                    # defining the reference category
  mutate(Education = relevel(Education, "0-No"))


# We change the unit of Aridity here
DataMerged1$Aridity2015 <- DataMerged1$Aridity2015 * 10^8

# Defining the variables of the model
Y<-"I_Before15"               # Response variable
XCovars <- c(15, 17, 57:64)   # age+education+GIS

formula_string<- paste(Y, paste(colnames(DataMerged1)[XCovars], collapse=" + "), sep="~")
print(paste(" Regression formula: ",formula_string))
```

Results as in Figure 5

```
# Logistics Regression
glm.fit <- glm(formula_string, data = DataMerged1, family = binomial)

# Nice printing of the results (using paper and knitr packages)
pretty_lm2 <- prettify(summary(glm.fit))
kable(pretty_lm2, digits = 3)
```

	Estimate	Std. Error	z value	Pr(>|z|)	OR

(Intercept)	-0.773	0.387	-1.998	0.046	0.462	*
Age	0.029	0.002	16.096	<0.001	1.030	***
Education: 1-Inc. Prim	-0.131	0.049	-2.684	0.007	0.878	**
Education: 2-Comp. Prim	-0.339	0.056	-6.021	<0.001	0.713	***
Education: 3-Inc. Sec.	-0.704	0.047	-15.081	<0.001	0.495	***
Education: 4-Comp. Sec.	-2.079	0.105	-19.886	<0.001	0.125	***
Education: 5-Higher	-3.204	0.118	-27.139	<0.001	0.041	***
Travel_Times2015	-0.001	0.002	-0.331	0.74	0.999	
SMOD2015	0.009	0.028	0.319	0.75	1.009	
Buildup2015	-0.006	0.224	-0.027	0.979	0.994	
Aridity2015	-0.663	0.057	-11.565	<0.001	0.515	***
Density2015	0.000	0.000	1.952	0.051	1.000	.
aWealthIndex2011	-0.078	0.064	-1.230	0.219	0.925	
aIncome2013	-0.003	0.001	-3.763	<0.001	0.997	***
aPP2013	0.015	0.004	3.515	<0.001	1.015	***

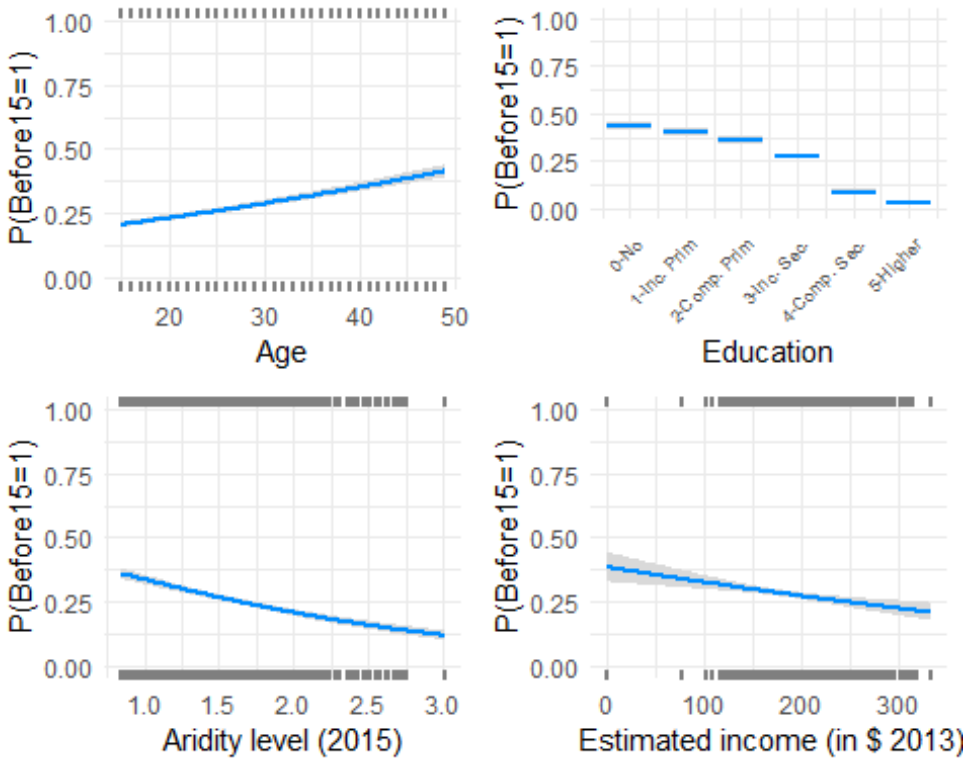
Confusion Matrix as in Figure 6

```
library("regclass")
confusion_matrix(glm.fit)

##          Predicted 0 Predicted 1 Total
## Actual 0       13499        1426 14925
## Actual 1        4442        1895  6337
## Total          17941        3321 21262
```

Visual representation of the logistic model, as in Figure 7

```
library(visreg)
library(ggpubr)

# Probabilities of married before 15 wrt
p.age <- visreg(glm.fit, "Age", scale="response", rug=2,
       xlab="Age",
       ylab="P(Before15=1)", gg=TRUE) +
  ylim(0,1) +theme_minimal()

p.education <- visreg(glm.fit, "Education", scale="response", rug=0,
       xlab="Education",
       ylab="P(Before15=1)", gg=TRUE) +
  ylim(0,1) + theme_minimal() +
 theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1, size=7))


p.aridity <- visreg(glm.fit, "Aridity2015", scale="response", rug=2,
       xlab="Aridity level (2015)",
       ylab="P(Before15=1)", gg=TRUE) +
  ylim(0,1) +theme_minimal()

p.income <- visreg(glm.fit, "aIncome2013", scale="response", rug=2,
       xlab=" Estimated income (in $ 2013)",
       ylab="P(Before15=1)", gg=TRUE) +
  ylim(0,1) +theme_minimal()

figure <- ggarrange( p.age, p.education, p.aridity, p.income,
```

```
                    #labels = c("Eduation", "Age",  "Aridity (2015)", ""),
                    ncol = 2, nrow = 2)
figure
```



4.3 Random Forests

```
set.seed(888)               # set random seed so we can reproduce the result
myRandomForest<-randomForest(as.formula(formula_string),
                             data = DataMerged1,
                             importance = TRUE,
                             maxnodes=25,
                             ntree=1000,
                             type="classification",
                             na.action = na.roughfix)
```

Accuracy rate and confusion Matrix as in Figure 9

```
myRandomForest

##
## Call:
##  randomForest(formula = as.formula(formula_string), data = DataMerged1,      importance
= TRUE, maxnodes = 25, ntree = 1000, type = "classification",      na.action = na.roughfix)

##                Type of random forest: classification
##                      Number of trees: 1000
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 27.44%
## Confusion matrix:
##       0    1 class.error
## 0 13856 1131   0.0754654
## 1  4729 1636   0.7429694
```

Variable importance plot as in Figure 10

```
varImpPlot(myRandomForest)
```