INFO7374 - Kiva & World Development Indicators

Bowei Wang, Dongyue Li, Sarthak Agarwal, Sriram Chandramouli

7 June 2016

Contents

1. Introduction **2**

Analysis Goals . . . 2

2. Data Profile **2**

2.1 Dataset description . . . 3

2.2 Description of rows and observations . . . 3

3. Dataset preparation **4**

3.1 Load libraries . . . 4

3.2 Loading the Kiva loans dataset into a data frame . . . 4

3.3 Loading the WDI dataset into data frame . . . 5

3.4 Create new variables . . . 6

`cleanedyear` . . . 6

`countryF` . . . 6

`loan_amount_numeric` . . . 7

`yearF` . . . 7

`funded_date_cleaned` . . . 7

`paid_date_cleaned` . . . 7

`sector_F` . . . 7

3.5 Merging Kiva and WDI data frames . . . 7

4. Variable summaries and visualizations **8**

4.1 Total loan amount by country . . . 8

4.2 Loan requirement by country . . . 9

4.3 Loan amount and sector . . . 10

4.4 Gender . . . 12

4.5 Loan amount, gender and sector . . . 13

4.6 Country's GDP and their loans . . . 14

4.7 Average loan amount, paid amount and sector . . . 20

4.8 Loan amount, status and sector . . . 21

4.9 The time taken from funded date to payback date . . . 23

5. Relationships between variables **24**

5.1 Relationship between loan amount and other variables 24

5.2 Relationship between status and other variables 27

5.3 Relationship between GDP and other variables 29

5.4 Relationship between country's loan amount and its labor force 30

6. Conclusion **32**

1. Introduction

A microfinance institution is an organization that is a source of financial services for small businesses lacking access to traditional banking systems. Kiva (www.kiva.org) is a micro lending website that works with such microfinance institutions to raise funds for low income entrepreneurs and provide loans for their business. Through this platform, anyone with an internet connection can make an interest free loan.

Kiva has field partners which are mainly microfinance institutions but also schools, NGOs and social enterprises. These field partners work at the local level and disburse loans to the borrowers. Loans are pre-disbursed which means that the loan is given out before being funded on the Kiva website. The field partners then collect stories, photos, and videos from the borrower and post them on Kiva which are published on the website after review. Anyone who can access the Kiva website can browse through the borrower's profile to make a loan for at least $25. Kiva aggregates all the money and backfills the loan that is already disbursed by the field partner. Field partners then collect repayments from the borrowers. The amount is then repaid to the Kiva lender.

Every year the World Bank releases a collection of World Development Indicators (WDI). The data represents the economic, demographic, social, environmental, educational and cultural indicators. It contains over 800 indicators covering more than 150 countries representing the most current and accurate global development data available which includes national, regional and global estimates.

Analysis Goals

Kiva's complete business model is based around lending and borrowing. The main focus of Kiva is to be a medium between the loan borrowers and lenders. Hence lenders are very important for Kiva and eventually to loan borrowers. The key objective of this analysis is to help Kiva and loan lenders make better decisions thereby helping in the economic development of the countries and aiding poverty alleviation. Specifically, this report seek to answer three questions:

1. What is the distribution of loan amount across countries and what is its relationship with country's GDP, working population and labor force?
2. What are the factors that can help lenders make better decisions?
3. What is the gender wise participation in Kiva's loans?

To perform this analysis, we merge the loans data with indicator codes from WDI dataset. This report provides graphical visualizations followed by association rules.

2. Data Profile

In the following two sub sections we understand the data and its structure. We further identify the relevant variables within the Kiva dataset that are important for our analysis.

2.1 Dataset description

The Kiva dataset consists of information on lenders and loans. We work primarily with the loans dataset and reference the lenders dataset where necessary. The data is available at www.build.kiva.org website and has three sub directories - `lenders`, `loans` and `loans_lenders`.

To analyze the structure, we load the 2129 files from the `loans` sub directory.

2.2 Description of rows and observations

Each loan entry in Kiva has the following 30 variables:

```
##  [1] "id"                              "name"
##  [3] "description"                     "status"
##  [5] "funded_amount"                   "basket_amount"
##  [7] "paid_amount"                     "image"
##  [9] "video"                           "activity"
## [11] "sector"                          "themes"
## [13] "use"                             "delinquent"
## [15] "location"                        "partner_id"
## [17] "posted_date"                     "planned_expiration_date"
## [19] "loan_amount"                     "lender_count"
## [21] "currency_exchange_loss_amount" "bonus_credit_eligibility"
## [23] "tags"                            "borrowers"
## [25] "terms"                           "payments"
## [27] "funded_date"                     "paid_date"
## [29] "journal_totals"                  "translator"
```

For our analysis, we choose the following variables and create a few additional variables:

Variable	Type	Description
`location.country`	Character	The country of the loan borrower
`countryF`	Character	Country variable converted to factor variable
`borrowers.gender`	Character	Gender of the borrower
`borrowers.GenderF`	Character	Gender variable converted to factor variable
`loan_amount`	Number	The amount of loan taken by the borrower
`loan_amount_numeric`	Number	Country variable converted to factor variable
`status`	Character	Loan payment status
`statusF`	Character	Status variable converted to factor variable
`sector`	Character	Sector of the loan
`sector_F`	Character	Sector variable converted to factor variable
`paid_amount`	Number	The amount of money paid back by the borrower
`posted_date`	Character	Date on which the loan is posted on Kiva
`cleanedyear`	Numeric	Year extracted from the `posted_date`
`yearF`	Character	`cleanedyear` variable converted to factor variable
`funded_date`	Character	Date on which loan is funded
`funded_date_cleaned`	Date	`funded_date` converted to YYYY-MM-DD format
`paid_date`	Character	Date on which loan is paid off
`paid_date_cleaned`	Date	`paid_date` converted to YYYY-MM-DD format
`day_diff`	Numeric	Difference between funded date and paid date
`activity`	Character	Reason for which loan is taken
`lender_count`	Number	Number of lenders

Units of these variables are:

Variable	Unit
loan_amount	Dollar
loan_amount_numeric	Dollar
paid_amount	Dollar
paid_amount_numeric	Dollar

3. Dataset preparation

To prepare the data for analysis, we:

1. Load the necessary libraries
2. Load the Kiva loans dataset into a data frame
3. Load the WDI dataset into a data frame
4. Create new variables
5. Merge Kiva and WDI data frame

3.1 Load libraries

The following libraries are used in this report:

- dplyr
- magrittr
- RJSONIO
- rlist
- WDI
- ggplot2
- parallel
- ffbase
- reshape2
- gridExtra
- gapminder
- grid
- maps
- arules

3.2 Loading the Kiva loans dataset into a data frame

In the next few steps we load the JSON files under the Kiva `loans` sub directory into R. Since there are more than 1 million loan entries, we use parallel computing to load the complete data.

We create a function which generates a data frame from a list.

```
dfrow.from.list = function(aList) {
  data.frame(rbind(unlist(aList)),
             stringsAsFactors=FALSE)
}
```

We use the above function and create another function to read a JSON file into a data frame. We use `isValidJSON` function to skip invalid JSON files.

```
readJSONFileIntoDataFrame <-
  function (filename) {
    if(isValidJSON(paste(data.folder,
          filename,
          sep=""))){
    paste(data.folder,
          filename,
          sep="") %>%
      fromJSON() %>%
      { .$loans } %>%
      list.select(posted_date, location.country = location$country, sector, loan_amount,
                  funded_date, paid_date, status, lender_count, activity,
                  borrowers = borrowers[1], paid_amount ) %>%
      lapply(dfrow.from.list) %>%
      bind_rows()
    }
  }
```

We create a socket cluster which creates a set of copies of R running in parallel. Based on the system configuration we use two cores and use the function `clusterExport` which provides several ways to parallelize computations.

```
cl <- makeCluster(2)
clusterExport(cl,
              c('dfrow.from.list','isValidJSON', 'data.folder', '%>%',
                'list.select','bind_rows','fromJSON',
                'readJSONFileIntoDataFrame'))
```

We use the parallel version of `Lapply` to apply `readJSONFileIntoDataFrame` function on JSON files, then stop the cluster when finished.

```
create.loan.df.cl = function(cl, loan.file.in) {
  loan.file.in %>%
    { parLapply(cl, ., readJSONFileIntoDataFrame) } %>%
    bind_rows()
}
loan.df = create.loan.df.cl(cl, loan.file)
stopCluster(cl)
```

3.3 Loading the WDI dataset into data frame

We choose the following Indicator codes from the WDI dataset. We add them to the `indicator.codes` variable.

```
indicator.codes = c("NY.GDP.MKTP.CD", "NY.GDP.MKTP.KD.ZG","SP.POP.DPND.OL",
                    "SP.POP.DPND.YG", "SL.TLF.TOTL.IN", "EN.POP.SLUM.UR.ZS")
```

The meaning of the chosen indicator codes are:

- NY.GDP.MKTP.CD - GDP at market prices (current US$)
- NY.GDP.MKTP.KD.ZG - GDP Annual growth rate
- SP.POP.DPND.OL - Age dependency ratio (% of working-age population)
- SP.POP.DPND.YG - Age dependency ratio, young (% of working-age population)
- SL.TLF.TOTL.IN - Labor Force
- EN.POP.SLUM.UR.ZS - Poverty, population living in slums (% of urban population)

We read the WDI data for these code into the df data frame.

```
df <- WDI(indicator = indicator.codes, extra = TRUE)
```

3.4 Create new variables

Following new variables are created for our analysis.

- cleanedyear
- countryF
- loan_amount_numeric
- yearF
- funded_date_cleaned
- paid_date_cleaned
- sector_F

cleanedyear

We create a numeric variable called cleanedyear by extracting year from posted_date. This variable is used for merging the Kiva and WDI dataset.

```
cleanedyear <- substr(loan.df[['posted_date']],1,4)
loan.df[["cleanedyear"]] <- as.numeric(cleanedyear)
summary(loan.df[["cleanedyear"]])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    2005    2011    2013    2012    2014    2016   17030
```

From the summary shown above we observe that the loan data is available from 2005 to 2016 and most of the loans are in the later years.

countryF

We create this variable by converting location.country into a factor variable.

```
loan.df$countryF <- factor(loan.df$location.country)
```

loan_amount_numeric

We create this variable by converting the `loan_amount` variable to numeric.

```
loan.df$loan_amount_numeric <- as.numeric(loan.df$loan_amount)
```

yearF

We create this variable by converting the `cleanedyear` variable into a factor variable.

```
loan.df$yearF <- factor(loan.df$cleanedyear)
```

funded_date_cleaned

We create this variable from the `funded_date` variable by removing unnecessary time information and extracting the date in the format YYYY-MM-DD.

```
loan.df$funded_date_cleaned <-
  as.Date(substr(loan.df$funded_date, 1, 10), "%Y-%m-%d")
```

paid_date_cleaned

We create this variable from the `paid_date` variable by removing the unnecessary time information and extracting the date in the format YYYY-MM-DD.

```
loan.df$paid_date_cleaned <-
  as.Date(substr(loan.df$paid_date, 1, 10), "%Y-%m-%d")
```

sector_F

We create this variable by converting `sector` variable into a factor variable.

```
loan.df$sector_F <- factor(loan.df$sector)
```

3.5 Merging Kiva and WDI data frames

We use the `cleanedyear` and `location.country` variables in loans data frame and join them on the basis of `country` and `year` variables in the WDI data frame. We use left join to gather all the data from Kiva dataset after merging.

```
finaldf <-
  merge(loan.df, df, by.x = c("location.country", "cleanedyear"),
        by.y = c("country", "year"), all.x = TRUE
  )
```

We remove the `null` records.

```
final1df <- finaldf[!is.na(finaldf$NY.GDP.MKTP.KD.ZG),]
final1df <- finaldf[!is.na(finaldf$SP.POP.DPND),]
final1df <- finaldf[!is.na(finaldf$SP.POP.DPND.OL),]
final1df <- finaldf[!is.na(finaldf$SP.POP.DPND.YG),]
final2df <- finaldf[!is.na(finaldf$SL.TLF.TOTL.IN),]
```

4. Variable summaries and visualizations

In the next few sections, we analyze the significant single and multiple variables for our analysis.

4.1 Total loan amount by country

We visualize the loans and their distribution across various countries by plotting the loan amounts on a world map.

We load the map data from `world2` and exclude Antarctica from the map.

```
world <- map_data("world2")
world <- subset(world,region!="Antarctica")
```

We calculate the total loan amount for each country.

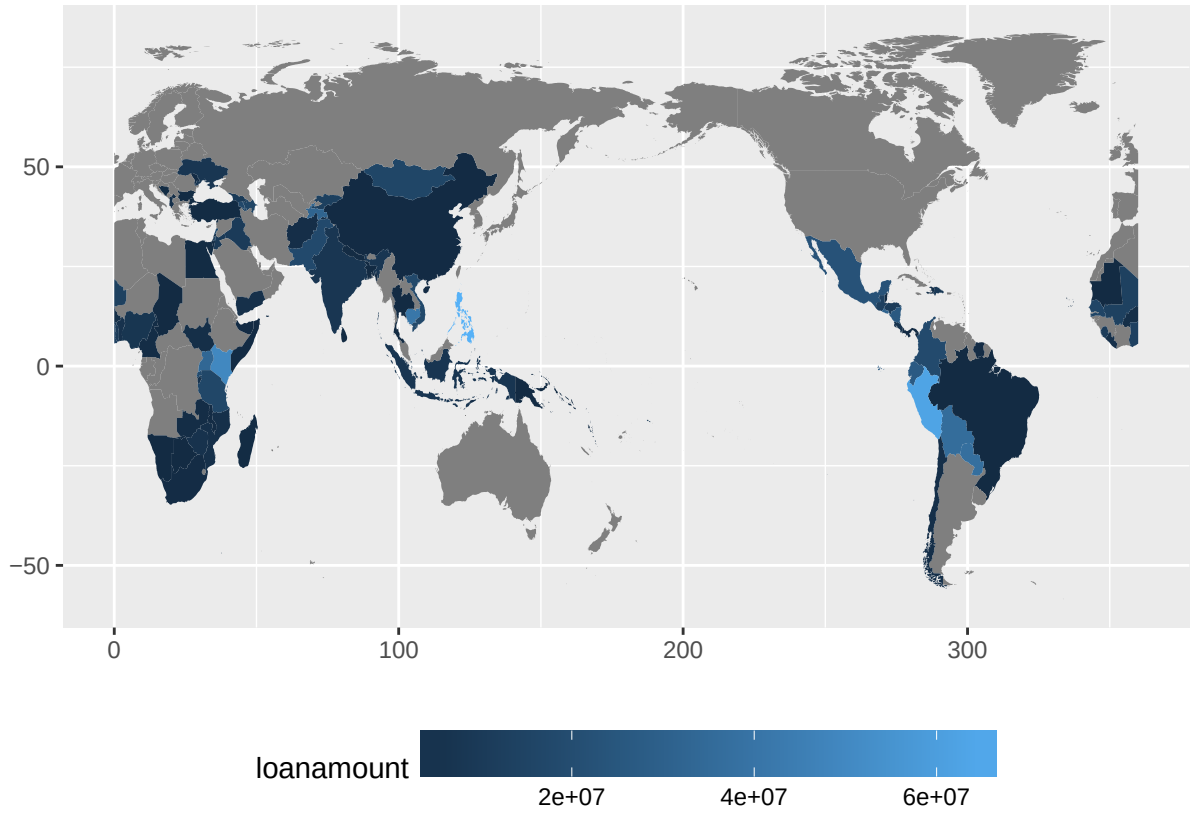
```
loan_amount_by_country <- condSum(loan.df$loan_amount_numeric,
                                  loan.df$countryF,
                                  na.rm = FALSE)
```

We match the country name with `loan_amount_by_country` variable.

```
world$loanamount <- loan_amount_by_country[
  match(world$region,
        names(loan_amount_by_country),
        nomatch=NA)]
```

We fill the data with the total loan amount by country and move the legend to the bottom of the graph.

```
map <- qplot(long, lat, data = world,
             group = group, fill = loanamount,
             geom ="polygon",ylab="",xlab="")
map + theme(legend.position="bottom",
            legend.key.width = unit(3, "line"))
```



We observe that amongst 250 countries for which loans are posted on the Kiva website, maximum loans are borrowed by the developing countries of South America, Africa and Asia. Our report focuses on these countries and aims at providing insights that will help in redirecting more funds to these countries.

4.2 Loan requirement by country

We identify the top 10 countries that have the largest loans for the year range 2006-2011. This will help lenders to focus on funding loans for these specific countries thereby catering to their economic development.

We calculate the total loan amount for every country by year.

```
loan_amount_by_country <-
  condSum(loan.df$loan_amount_numeric,
          list(loan.df$countryF,loan.df$yearF),
          na.rm = FALSE)
```

We create a data frame of the above matrix.

```
loan_amount_by_country_df <-
  data.frame(row.names(loan_amount_by_country),
             loan_amount_by_country)
```

We subset the above data frame for the year range 2006-2011 and rename columns for readability.

```
loan_amount_by_country_df_from2006_to2011 <-
  loan_amount_by_country_df[,c("row.names.loan_amount_by_country.",
                               "X2006","X2007","X2008","X2009",
                               "X2010","X2011")]
colnames(loan_amount_by_country_df_from2006_to2011) <-
  c("country", "2006", "2007","2008",
    "2009", "2010","2011")
```

We calculate total loan amount for each country from 2006-2011.

```
loan_amount_by_country_df_from2006_to2011$total <-
  rowSums(loan_amount_by_country_df_from2006_to2011[2:7])
```

We sort this data frame on the basis of total loan amount and get the records for top 10 countries which have the largest total loan amounts.

```
loan_amount_by_country_df_from2006_to2011_top10 <- head(arrange(
  loan_amount_by_country_df_from2006_to2011,
              desc(total)), n = 10)
loan_amount_by_country_df_from2006_to2011_top10
```

```
##        country   2006    2007    2008    2009    2010    2011    total
## 1         Peru      0  657625 4043050 5895425 7088725 8355600 26040425
## 2     Cambodia  45175 1215850 4231575 4763000 5741750 4401750 20399100
## 3  Philippines      0       0   71800 3011800 5965450 6381850 15430900
## 4       Uganda 279500  445475 2404175 3196850 3838700 3955875 14120575
## 5      Bolivia      0  322550 1580025 3950900 3434250 2688725 11976450
## 6   Tajikistan      0  758300 2378850 3368350 1887850 3152600 11545950
## 7        Kenya 267375 1182325  364025  795750 2951175 5684550 11245200
## 8    Nicaragua   8750  271125 1532100 2541800 3349050 3156800 10859625
## 9      Ecuador 180600  879625  300500  320050 2998475 4173475  8852725
## 10      Mexico 238425 1541575 1017850  685350 1755425 3532300  8770925
```

We observe that Peru, Cambodia and Philippines are the countries that have the largest loan amount. Hence lenders can filter the loan postings on the Kiva website for these countries.

Also, it is important to note that for Cambodia, Bolivia and Nicaragua loan amount increases till 2010 and then decreases in 2011.

4.3 Loan amount and sector

This analysis provides an understanding of the total loan amount for each sector. There are 15 sectors for which loans are posted on the Kiva website.

We calculate the total loan amount for each sector.

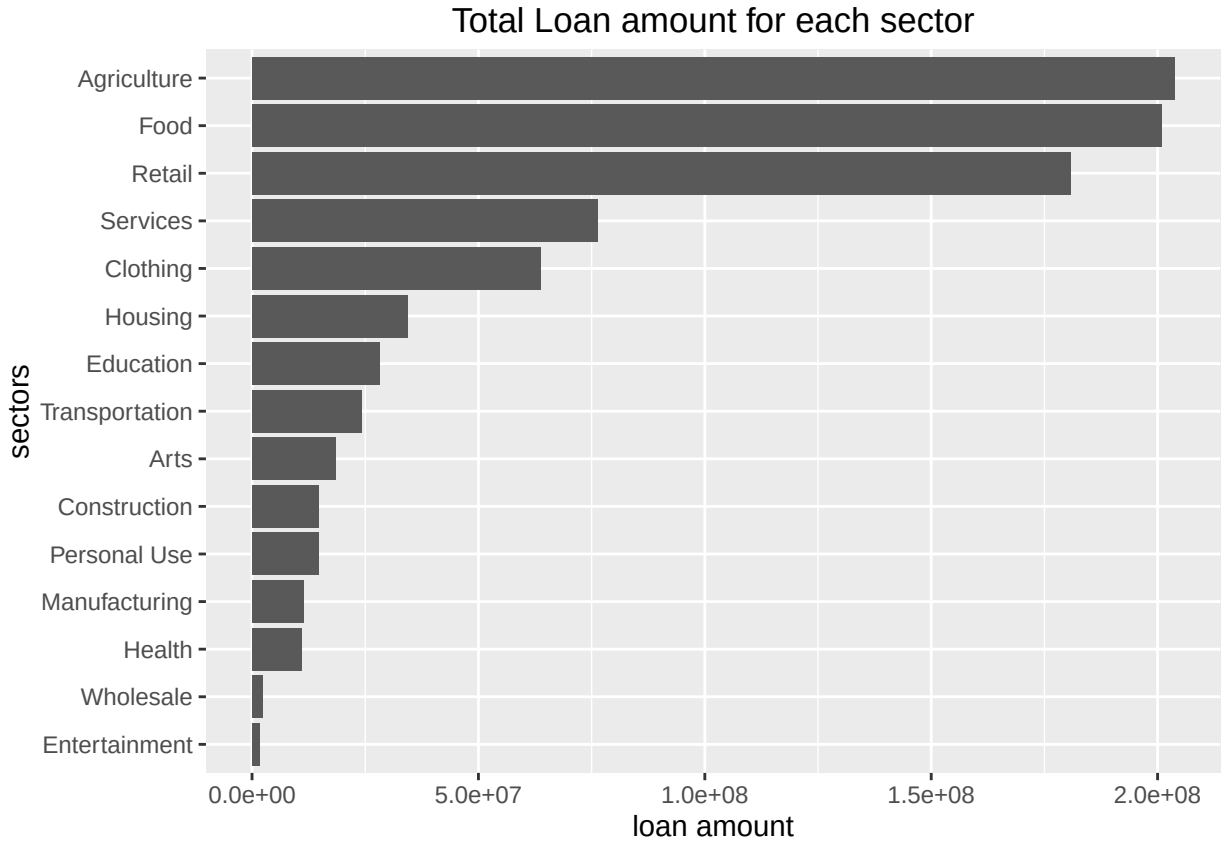
```
loan_amount_by_sector <- condSum(loan.df$loan_amount_numeric,
                                 loan.df$sector_F,na.rm = FALSE)
```

We create the `loan_amount_by_sector_df` data frame from the `loan_amount_by_sector` variable created above.

```
loan_amount_by_sector_df <-data.frame(names(loan_amount_by_sector),
                                      loan_amount_by_sector)
```

The code chunk below plots the graph for total loan amount for each sector and sorts it on the basis of loan amount.

```
loan_amount_by_sector_df %>%
  ggplot (aes(x = reorder(names(loan_amount_by_sector),
                          loan_amount_by_sector), y = loan_amount_by_sector)) +
  ggtitle("Total Loan amount for each sector") +
  xlab("sectors") +
  ylab("loan amount") +
  geom_bar(stat = "identity") +
  coord_flip()
```

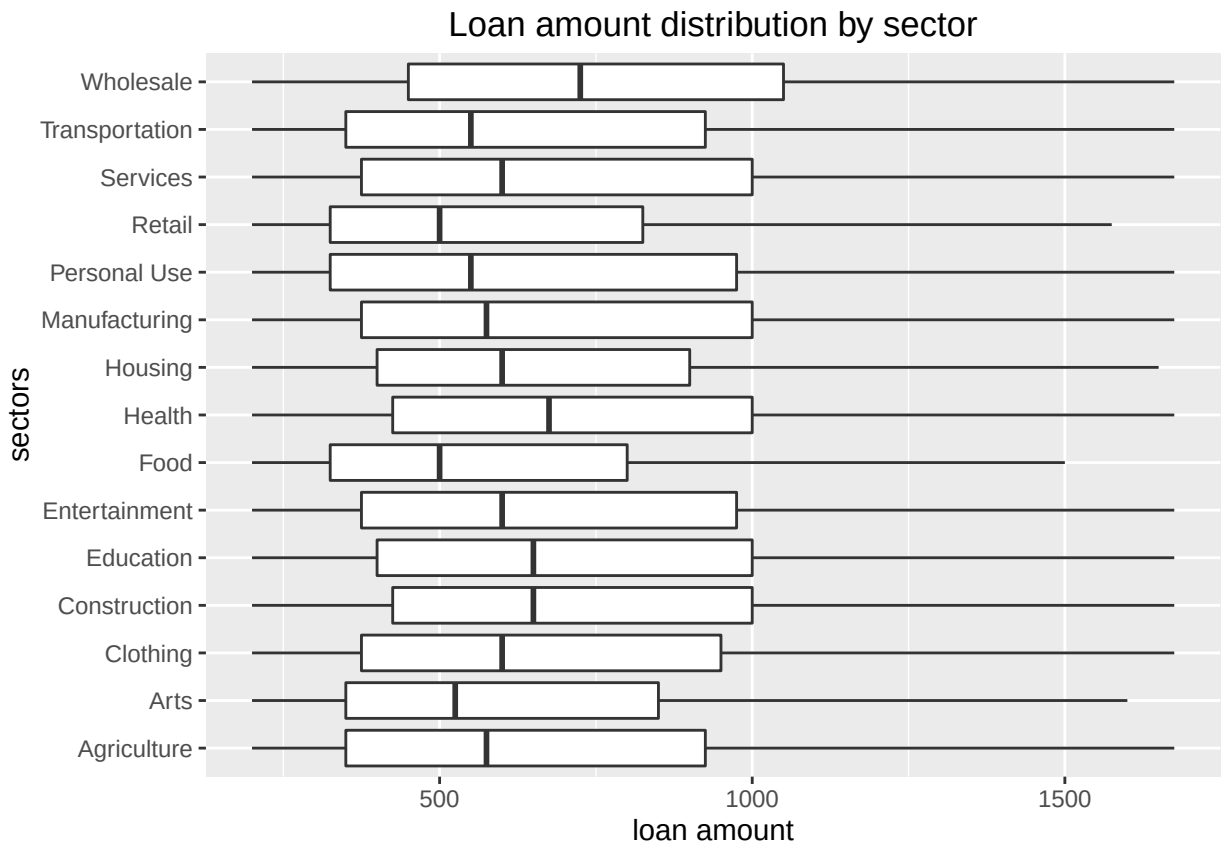


We observe that people are asking for the higher amount of loans in the agriculture, food and retail sectors. Lenders should emphasize on funding loans in these specific sectors. Entertainment and wholesale sectors have the least loans.

To further analyze the total loan amount distribution by sector we use a box plot. We modify the wide scale to 10 ~ 90 percentile to eliminate extreme variables.

```
loan.df %>%
ggplot(aes(sector_F,loan_amount_numeric)) +
ggtitle("Loan amount distribution by sector") +
xlab("sectors") +
```

```
ylab("loan amount") +
geom_boxplot(outlier.shape = NA) +
scale_y_continuous(limits = quantile(loan.df$loan_amount_numeric, c(0.1, 0.9))) +
coord_flip()
```



In this graph, we can observe that food and retail sectors have the smallest loan amount distribution. On the other side, wholesale and service sectors have the widest loan amount distribution.

4.4 Gender

The `borrowers.genderF` variable represents the loan borrower's gender. We create this variable by converting `borrowers.gender` into a factor variable.

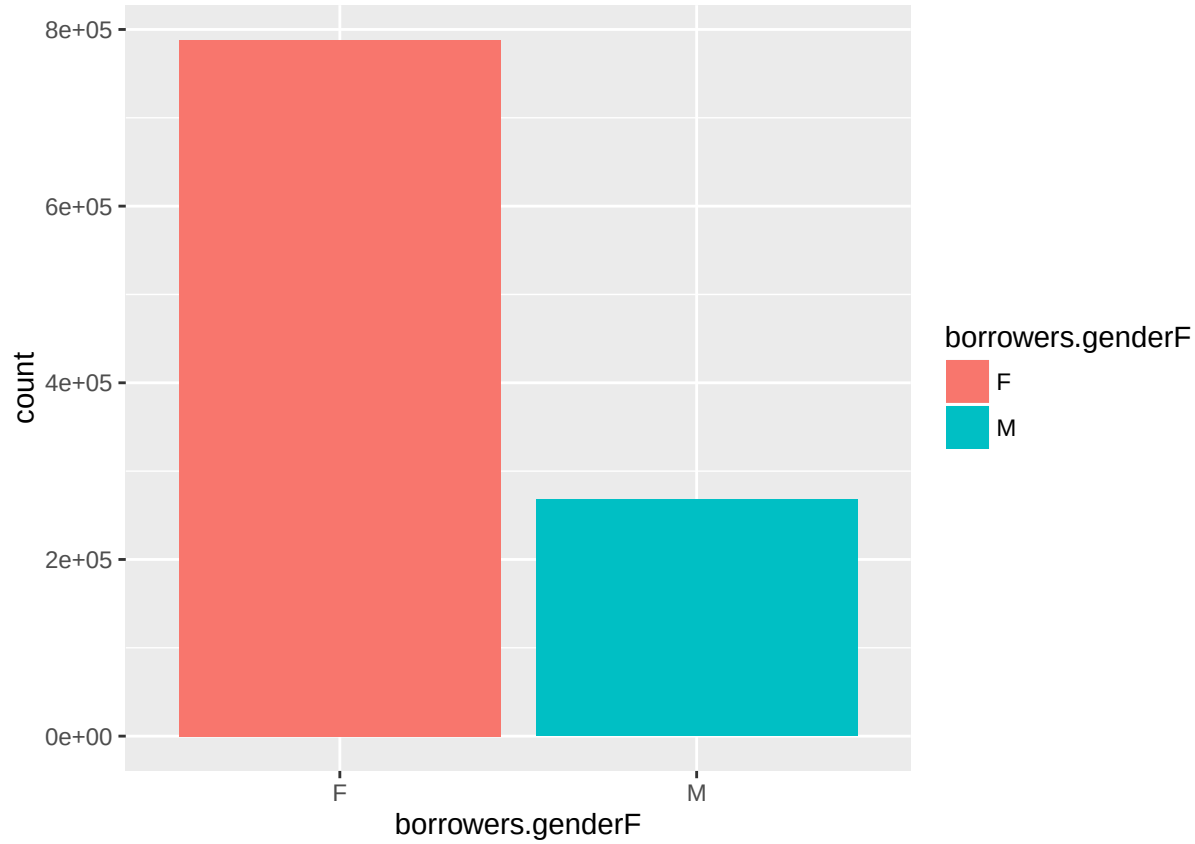
```
loan.df$borrowers.genderF <- factor(loan.df$borrowers.gender)
```

The code below prints the summary statistics and a bar graph for visual representation.

```
summary(loan.df$borrowers.genderF)
```

```
##      F      M
## 788113 267500
```

```
loan.df %>%
  ggplot(aes(x=borrowers.genderF)) +
  geom_bar(aes(fill=borrowers.genderF))
```



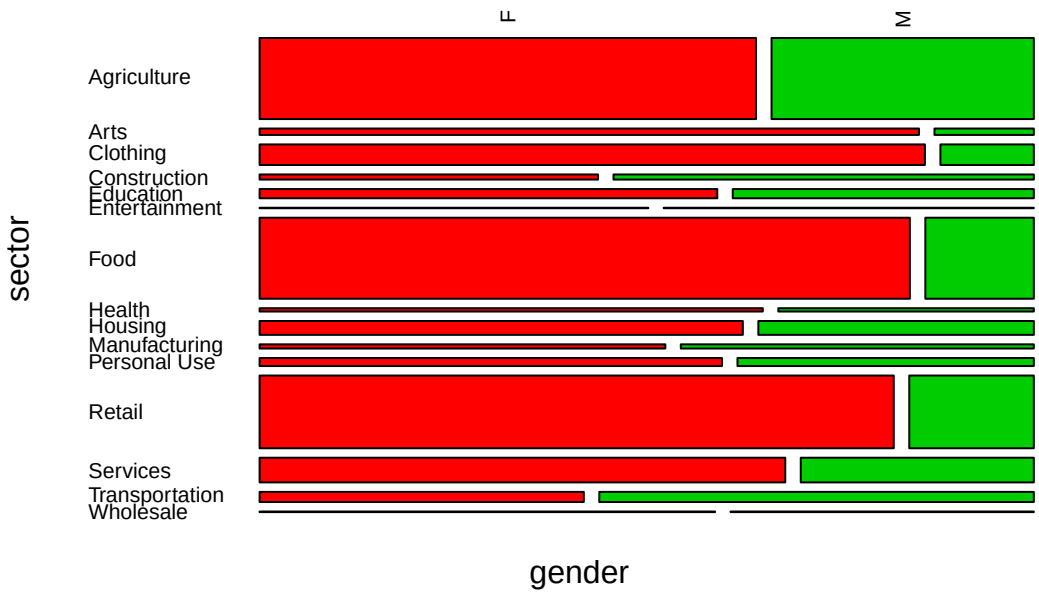
We observe that around 70% of borrowers on the Kiva website are females.

4.5 Loan amount, gender and sector

We analyze the relationship between the total loan amount, gender and sector. Our goal is to visualize the gender based distribution of loan amount in each sector.

Below code plots a mosaic graph which represents the gender on the x-axis and sector on the y-axis.

```
mosaicplot(sector_F ~ borrowers.genderF, data = loan.df,
           main = "Gender of borrower for each sector",
           xlab = "gender", ylab = "sector",
           dir = c('h','v'), las = 2,
           col = c(2:9))
```



We observe that in most sectors, women made more requests than male. However, some sectors like construction, transportation and manufacturing have more requests taken by males.

4.6 Country's GDP and their loans

We analyze the relationship between the total loan amount for the countries and the GDP of those countries. As Kiva aims to alleviate poverty in the developing countries, this analysis helps Kiva and the lenders to contribute towards the country's development by making informed decisions on funding loans.

We conduct this analysis for a year range of 2006 to 2011.

We drop all null `posted_date` records from the loans data frame.

```
loan.df <- loan.df[!is.na(loan.df$cleanedyear),]
```

We create a data frame called `GDP_df` with GDP, year and top 10 countries which have the largest loan amount.

```
GDP_df <- data.frame(df$year, df$NY.GDP.MKTP.CD, df$country)
```

We have a data frame called `loan_amount_by_country_df_from2006_to2011_top10`. It has the data related to top 10 countries which have the largest loan amount from 2006 to 2011. We change its data structure by converting all year columns into rows and converting the data to `long` format using `melt()` package.

```
loan_amount_by_country_df_from2006_to2011_top10_long <-
  melt(loan_amount_by_country_df_from2006_to2011_top10[1:7], id="country")
```

We merge `loan_amount_by_country_df_from2006_to2011_top10_long` data frame and `GDP_df` data frame by `country` and `year`.

```
loan_GDP_df <- merge(loan_amount_by_country_df_from2006_to2011_top10_long,
                     GDP_df, by.x = c("country", "variable"),
                     by.y = c("df.country", "df.year"))
```

We rename the columns of the `loan_GDP_df` data frame for better readability.

```
colnames(loan_GDP_df) <- c("country", "year", "loanamount","GDP")
```

We convert `loanamount` and `GDP` columns into rows using melt function.

```
loan_GDP_df <- melt(loan_GDP_df[1:4], id=c("country","year"))
```

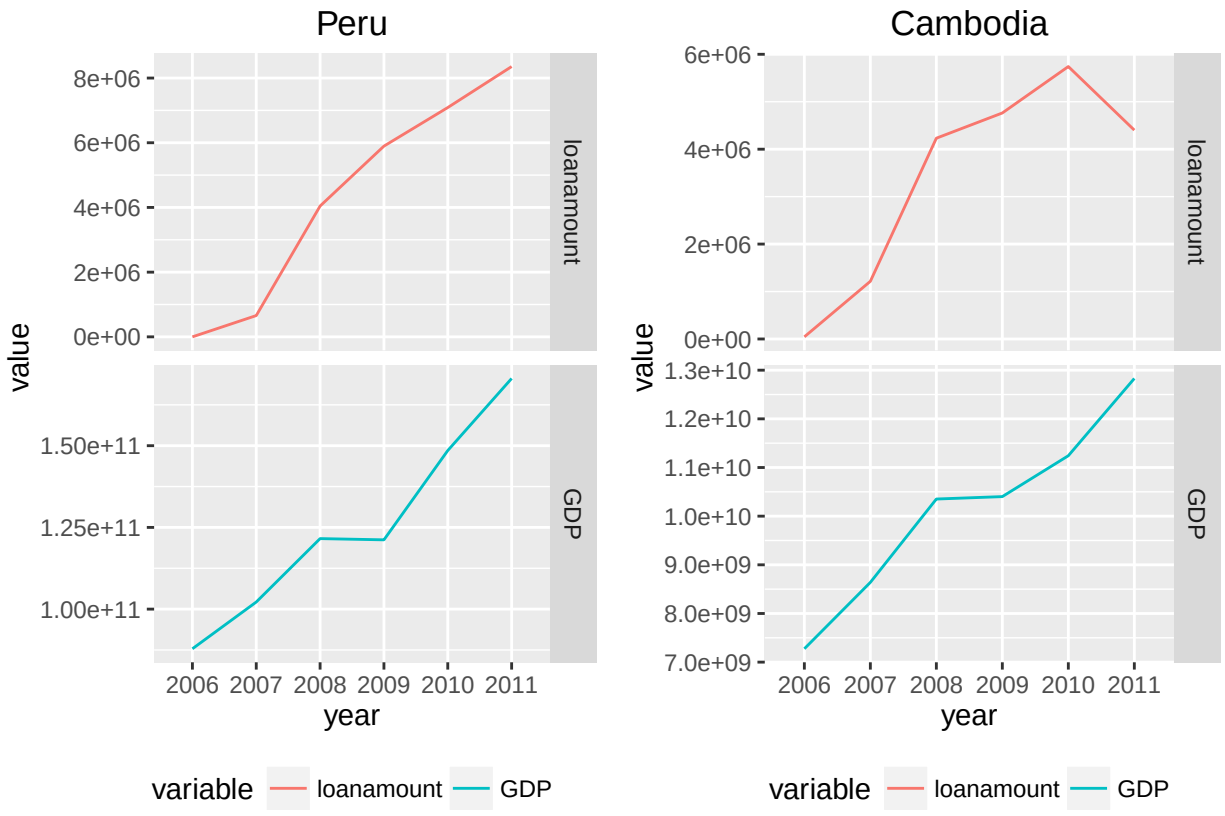
We get the country names and store it in a variable `top10_country_name`.

```
top10_country_name <-
  loan_amount_by_country_df_from2006_to2011_top10$country %>%
  sapply(as.character)
```

We plot different graphs based on the countries and observe the relationship between GDP and loan amount over the year range of 2006 to 2011.

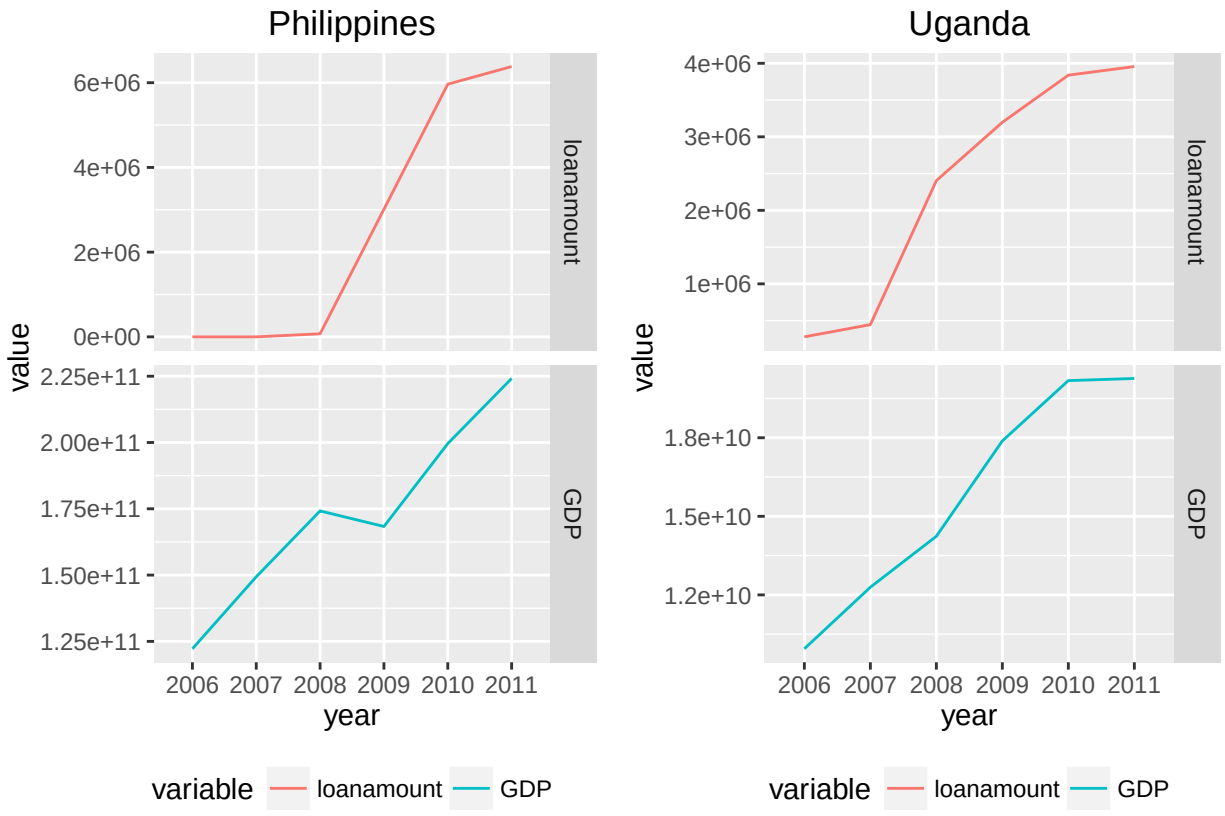
```
title <- top10_country_name[1]
loan_GDP_df_filter <- filter(loan_GDP_df, country %in% top10_country_name[1])
p1 = loan_GDP_df_filter %>%
  ggplot(aes(x=year, y=value, group=variable,
             colour=variable)) +
  geom_line()+
  ggtitle(title) +
  xlab("year") +
  ylab("value")
p1 = p1+theme(legend.position="bottom") + facet_grid(variable ~ ., scales = "free_y")
```

```
grid.arrange(p1, p2, ncol=2)
```



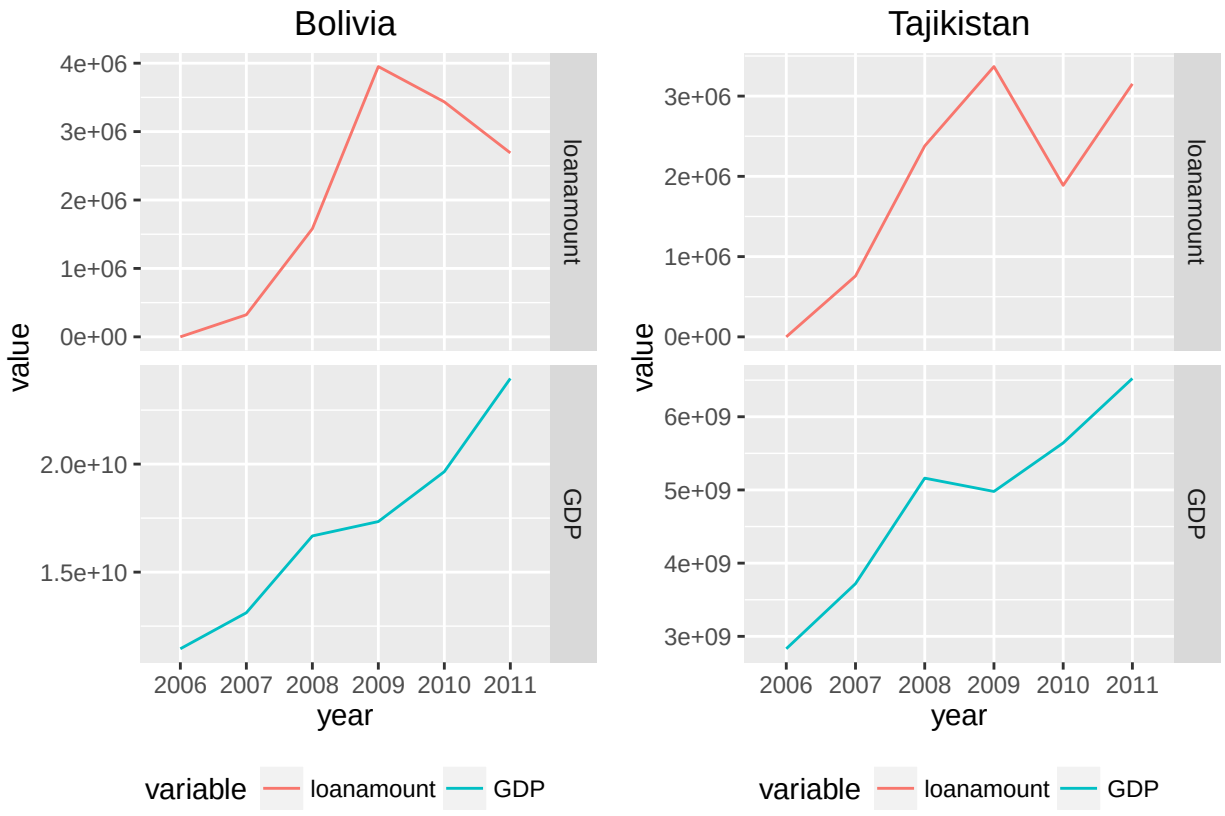
Loan requirement in Peru is increasing with its GDP, however Peru faced economic stagnation from 2008-2009. Loan requirement in Cambodia increases with its GDP till the year 2010 and decreases from 2010-2011.

```
grid.arrange(p3, p4, ncol=2)
```



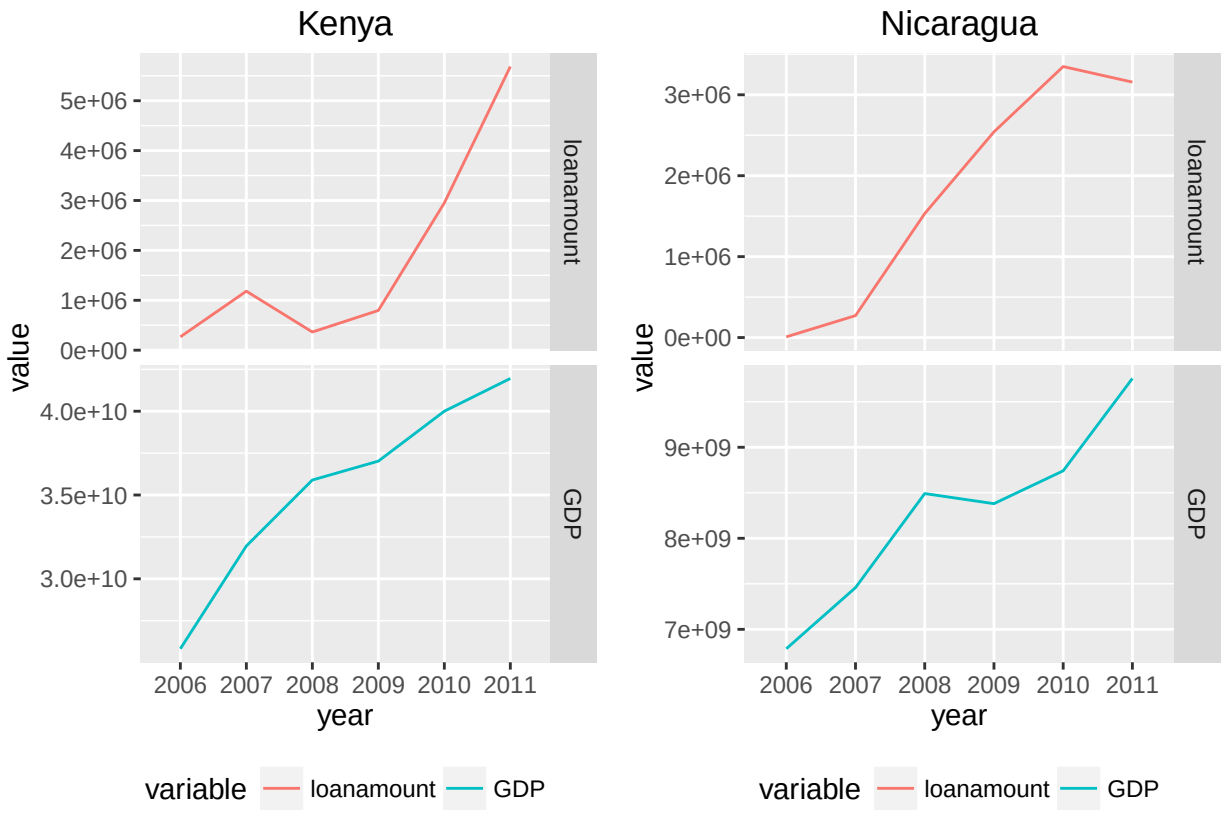
We observe that GDP of Philippines declined from 2008 to 2009, however the loan amount constantly increased from 2006 to 2011. Loan amount in Uganda is increasing with its GDP.

```
grid.arrange(p5, p6, ncol=2)
```



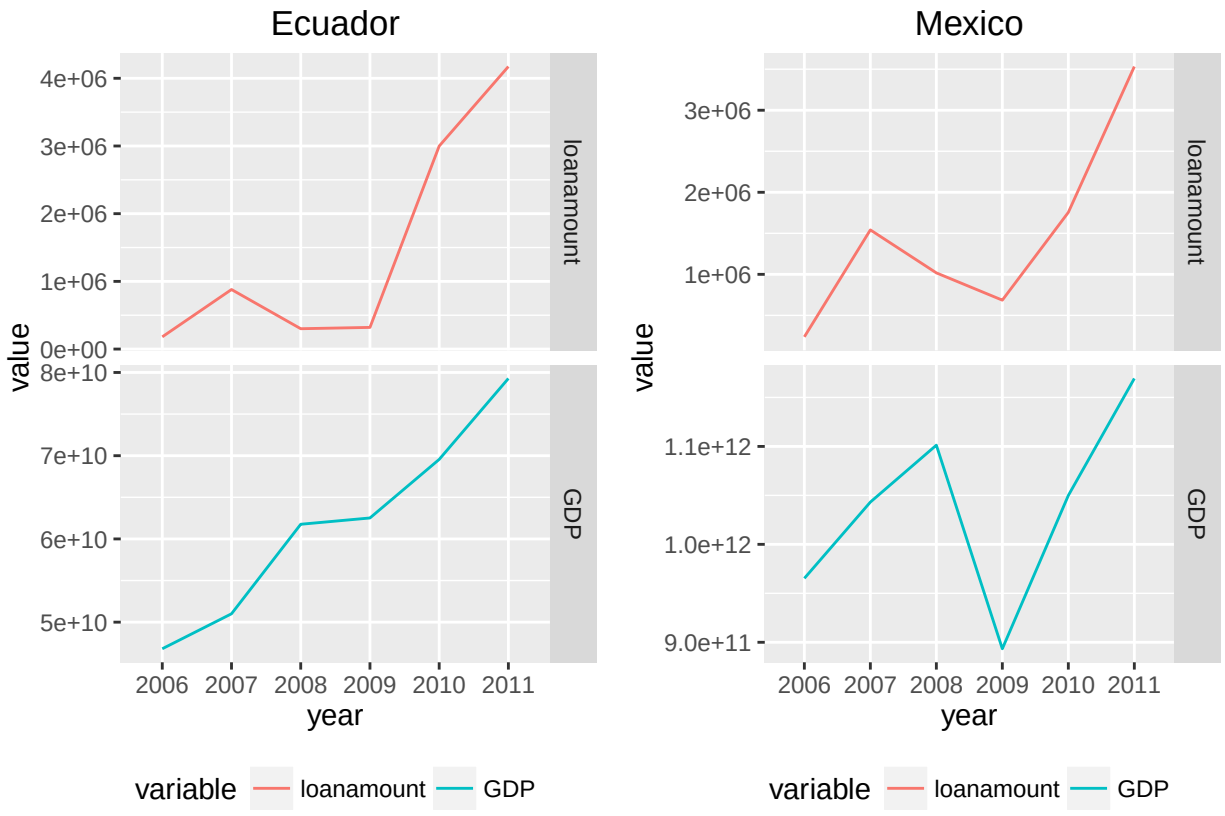
Loan amount in Bolivia increases with the country's GDP but suddenly decreases from 2009-2011. We observe that Tajikistan's loans amount increases till 2009 then decreases till 2010 and then again increases in 2011.

```
grid.arrange(p7, p8, ncol=2)
```



We observe that Kiva loans got popular in Kenya only after struggling from 2006-2009. They skyrocketed from 2009-2011, however Kenya's GDP increased between 2006-2011. Nicaragua's GDP increased till 2008 and then decreased in 2009 and then increased in 2010 and 2011. However, the loans requirement increased till 2010 and then decreased in 2011.

```
grid.arrange(p9, p10, ncol=2)
```



From the above graphs we observe that in Ecuador GDP increases from 2006-2011. However, loan amount increases and then decreases till 2009 and then increases till 2011. In Mexico, GDP increases till 2008, decreases in 2009 and then increases till 2011. However, loan amount increases in 2007, decreases till 2009 and then increases till 2011.

4.7 Average loan amount, paid amount and sector

Kiva doesn't guarantee that loans will be paid back but it's important for the lender to have that information before lending money. We analyze the average loan amount against the average paid back amount for every sector. `Loan amount`, `paid amount` and `sector` are required variables for this analysis. The average value of loan and paid amount for each sector is calculated and represented in a bar graph.

We convert `paid_amount` variable from character to numeric and then create `sel_finaldf` data frame from `loan.df` by selecting `loan_amount` and `sector`.

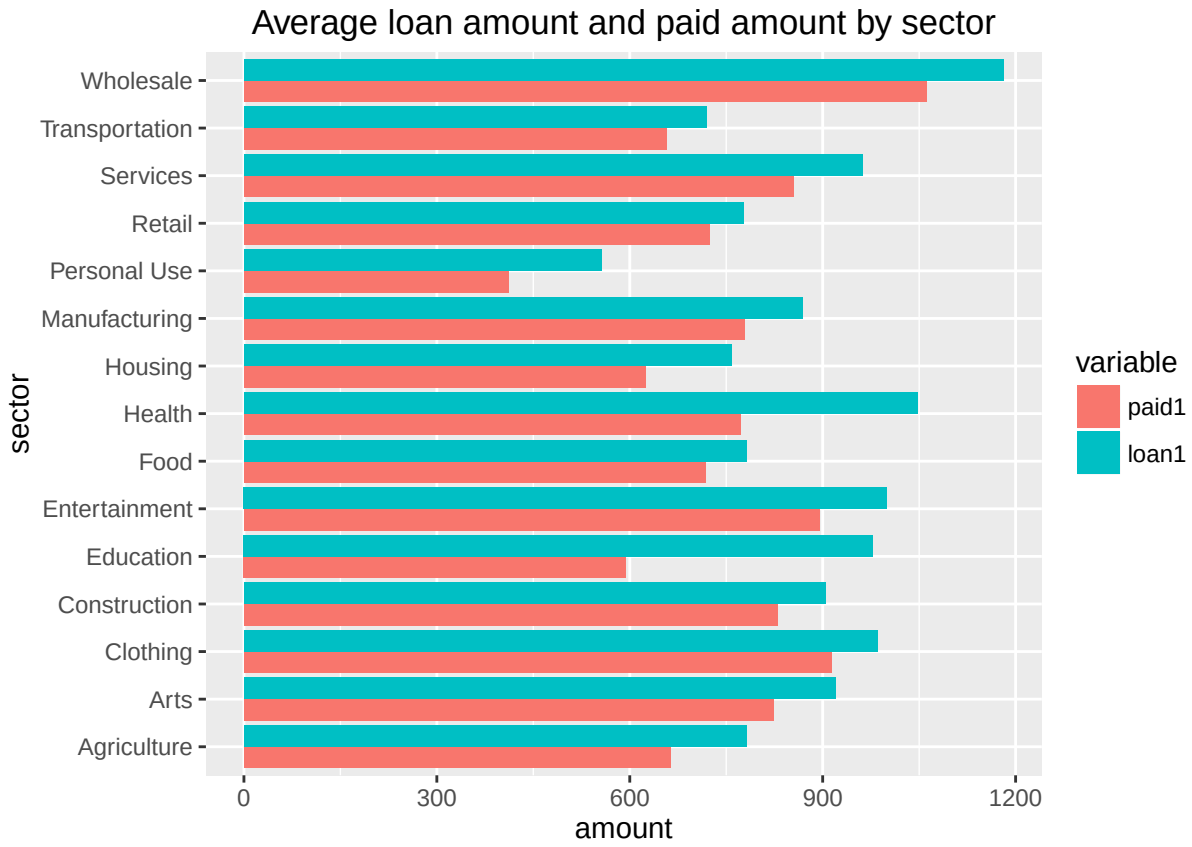
```
loan.df["paid_amount"] <- as.numeric(loan.df$paid_amount)
sel_finaldf <- select(loan.df,paid_amount,loan_amount,sector)
sel_finaldf <- na.omit(sel_finaldf)
```

We calculate the mean value of paid amount and loan amount for each sector.

```
sel_bysec <- ddply(sel_finaldf, .(sector), summarize,
                   paid1 = mean(paid_amount),
                   loan1 = mean(as.numeric(loan_amount)))
```

We calculate average loan amount and paid amount by sector.

```
dfm <- melt(sel_bysec, id.vars = c('sector'))
ggplot(dfm, aes(x = factor(sector), y = value, fill = variable) ) +
  geom_bar(stat="identity", position = 'dodge') +
  ggtitle("Average loan amount and paid amount by sector") +
  xlab("sector") +
  ylab("amount") +
  coord_flip()
```



We compare the average loan amount and paid amount for each sector in the graph above and observe that the education and health sector have the largest difference between paid loan amount and total loan amount. In other words, the loan amount that is not paid is higher in education and health sector than other sectors.

4.8 Loan amount, status and sector

To understand the working of loans, status plays an important role. We see that there are 8 loan statuses in Kiva data:

- `defaulted` - Loans which are never paid back after not being paid for 6 months. They are a financial loss to the lender of that loan.
- `expired` - If the loan doesn't get fully funded within 30 days then it "expires".
- `funded` - Loans that are completely funded are in "funded" state.
- `fundraising` - These are the loans which are not yet funded. Lenders can only lend money to these loans.

- `in_repayment` - When the loan is in repayment, it means that the loan has been disbursed to the borrower and they are in the process of using the funds.
- `paid` - These are the loans which are fully paid back by the borrower.
- `refunded` - Few loans or a portion of it needs to be refunded, those loans maintain "refunded" status.

To analyze the status variable we create a factor variable called `statusF` from the `status` character variable.

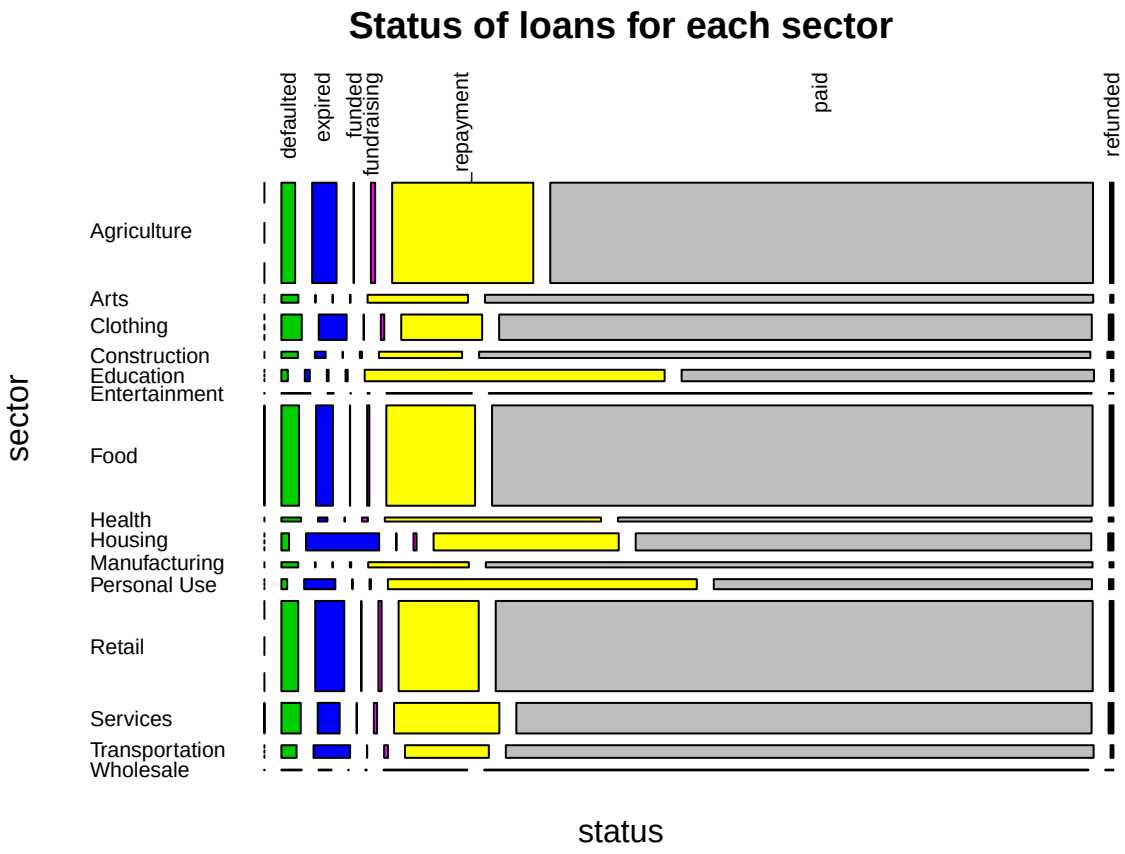
```
loan.df$statusF <- factor(loan.df$status)
table(loan.df$statusF)
```

```
##
##                 defaulted      expired       funded  fundraising
##             4         22443        34787          469         4411
## in_repayment          paid     refunded
##       165366       805562         5541
```

We observe that majority of the loans have `paid` status and only a minority of loans are `default`. This increases the confidence of lenders in Kiva's model.

We further visualize the status of loans based on sector and represent it in a mosaic plot where loan status is on the x axis and sector is on y axis.

```
op <- par(mar = rep(2, 4))
mosaicplot(sector_F ~ statusF, data = loan.df,
           main = "Status of loans for each sector",
           xlab = "status", ylab = "sector",
           dir = c('h','v'), las = 2,
           col = c(2:9), cex.axis = 0.7)
```




```
par(op)
```

We conclude that majority of the loans are paid back, however large number of loans are also in `in_repayment` status. "Education" and "personal use" sectors have the largest number of loans in `in_repayment` status. "Housing" sector has the largest number of loans in the `expired` status.

4.9 The time taken from funded date to payback date

Kiva does not provide any information to lenders about timeline of loan repayment, but it's imperative for a lender to have a knowledge of the timeline matters for prior loans before lending money. We analyze the time period when a loan is funded and is paid back for the countries which have a larger loan amount.

We calculate the difference in days by subtracting funded date from paid date and convert the days into numeric.

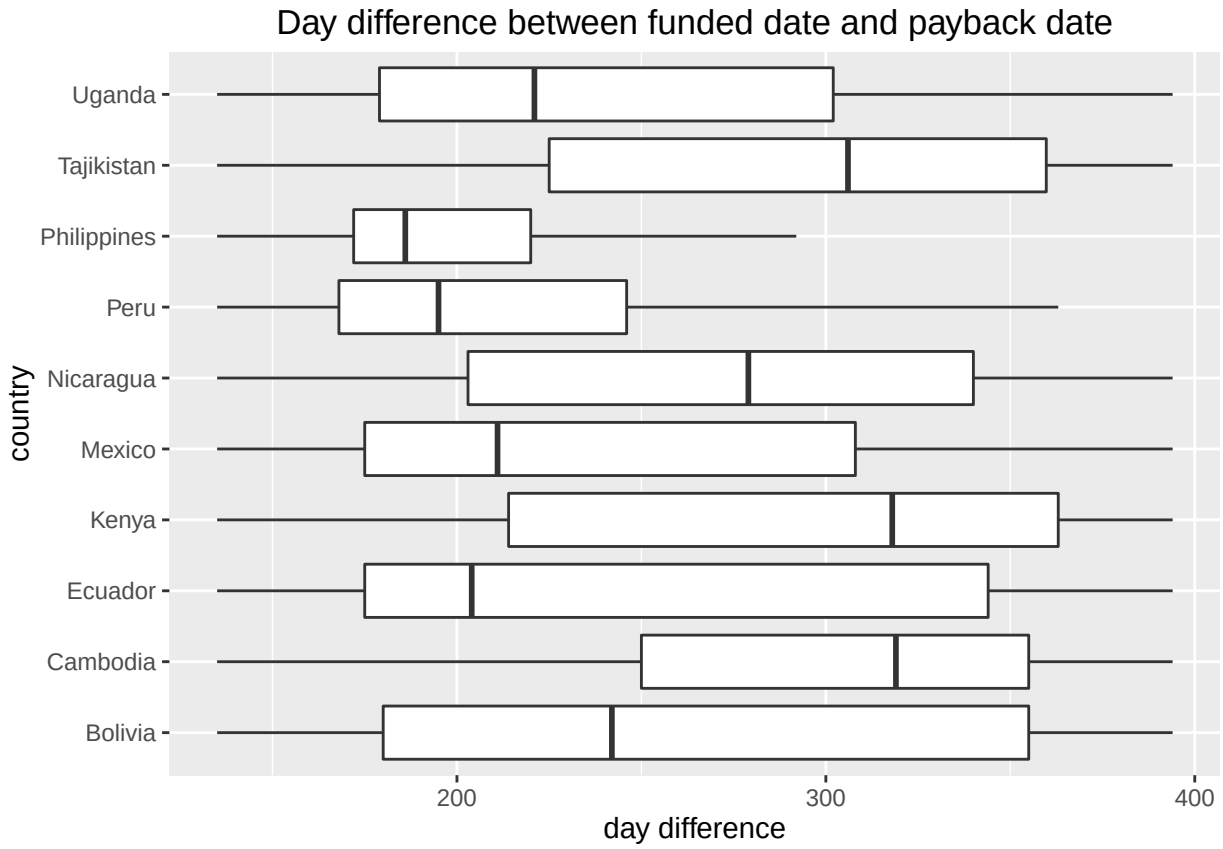
```
loan.df$day_diff <-
  as.numeric(loan.df$paid_date_cleaned - loan.df$funded_date_cleaned, "days")
```

We select top 10 countries by loan amount using filter.

```
loan.df_date <-
  filter(loan.df, loan.df$countryF %in% c(top10_country_name))
```

We plot a boxplot and change the scale of the graph to eliminate outliers.

```
loan.df_date %>%
  ggplot(aes(countryF, day_diff)) +
  ggtitle("Day difference between funded date and payback date") +
  xlab("country") +
  ylab("day difference") +
  geom_boxplot(outlier.shape = NA) +
  scale_y_continuous(limits = quantile(loan.df_date$day_diff, c(0.1, 0.9), na.rm = TRUE)) +
  coord_flip()
```



The above graph helps us visualize the day difference between funded date and paid date of the loans for high loan borrower countries. We observe that most of the loans are paid back within a year. We also conclude that Philippines has the shortest payback period whereas Kenya and Cambodia have the longest payback period among the top 10 borrowers. Therefore, Philippines is the fastest option to get back the money.

5. Relationships between variables

In the next few sections we analyze the relationship between several variables using association rules.

5.1 Relationship between loan amount and other variables

We compare the loan amount and its relationship with `country`, `sector`, `activity`, `status`, `lendercount` and working age population from the WDI dataset.

We divide the loan amount into two categories: "0" and "1" where "0" refers to the loan amount which is less than the average loan amount and "1" which refers to the loan amount that is greater than the average loan amount. We store this in the variable `loan_amount_check`.

We group the lender count into 4 quantiles - Q1, Q2, Q3, Q4 and store it in `lender_count_quant`. Q1 denotes the lender count between 0 and 25% of the total lender count, Q2 denotes the lender count between 25% and 50% of the total lender counts, Q3 denotes the lender count between 50% and 75% of the total lender count and Q4 denotes the lender count between 75% and 100% of the total lender count.

Similarly we group the percentage of both young (`SP.POP.DPND.YG`) and old (`SP.POP.DPND.OL`) working age population into 4 quantiles.

```
finaldf_loanamount <- final1df %>%
  mutate(loan_amount_check = ifelse(loan_amount >=mean(loan_amount), 1, 0)) %>%
  mutate(lender_count_quant = cut(lender_count,
                                  quantile (lender_count, c (0, .25, .5, .75, 1)),
                                  labels=c('Q1','Q2','Q3','Q4'))) %>%
  mutate(working_pop_young = cut(SP.POP.DPND.YG,
                                 quantile (SP.POP.DPND.YG, c (0, .25, .5, .75, 1)),
                                 labels=c('Y1','Y2','Y3','Y4'))) %>%
  mutate(working_pop_old = cut(SP.POP.DPND.OL,
                               quantile (SP.POP.DPND.OL, c (0, .25, .5, .75, 1)),
                               labels=c('O1','O2','O3','O4'))) %>%
  select(lender_count_quant,status,activity,sector,
         location.country,loan_amount_check,borrowers.gender,working_pop_young,
         working_pop_old)

finaldf_loanamount <- data.frame(finaldf_loanamount)
```

We convert the variables into factors to apply the association rules.

```
finaldf_loanamount["lender_count_quant"] <-
  as.factor(finaldf_loanamount$lender_count_quant)
finaldf_loanamount["status"] <-
  as.factor(finaldf_loanamount$status)
finaldf_loanamount["activity"] <-
  as.factor(finaldf_loanamount$activity)
finaldf_loanamount["sector"] <-
  as.factor(finaldf_loanamount$sector)
finaldf_loanamount["location.country"] <-
  as.factor(finaldf_loanamount$location.country)
finaldf_loanamount["loan_amount_check"] <-
  as.factor(finaldf_loanamount$loan_amount_check)
finaldf_loanamount["borrowers.gender"]  <-
  as.factor(finaldf_loanamount$borrowers.gender)
finaldf_loanamount["working_pop_young"]  <-
  as.factor(finaldf_loanamount$working_pop_young)
finaldf_loanamount["working_pop_old"]  <-
  as.factor(finaldf_loanamount$working_pop_old)
```

We control the number of results by setting the support value as 0.01, confidence as 0.5 and set the maximum variables to three.

We set the rhs value to `loan_amount_check` (0 and 1) and lhs to default. The result is sorted by lift value.

```
apriori.appearance = list(rhs=c("loan_amount_check=0",
                                "loan_amount_check=1"), default="lhs")
apriori.parameter = list(support=0.01, confidence=0.2, minlen=1, maxlen=3)
rules = apriori(finaldf_loanamount, parameter =
                  apriori.parameter,appearance =   apriori.appearance)
rules.sorted <- sort(rules, by = "lift")
```

The redundant rules are identified and removed using the below code.

```
subset.matrix <- is.subset(rules.sorted, rules.sorted)
subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA
redundant <- colSums(subset.matrix, na.rm=T) >= 1
rules.pruned <- rules.sorted[!redundant]
```

The below rule has the third highest lift value.

```
inspect(rules.pruned[3])
```

```
##   lhs                              rhs                       support confidence     lift
## 1 {lender_count_quant=Q4,
##    location.country=Cambodia} => {loan_amount_check=1} 0.02044636  0.9994448 2.829576
```

The rule states that Cambodia has high number of lenders which are associated with higher loan amount. The support value indicates that in 2% of the total loan amount across all countries, Cambodia has high lenders and high loan amount. The confidence score indicates that 99 percent of high loan amount in Cambodia have high lender count. The lift value of 2.8 means that this scenario is 2.8 times more likely to occur in data.

```
inspect(rules.pruned[29])
```

```
##   lhs                         rhs                       support confidence    lift
## 1 {working_pop_young=Y2,
##    working_pop_old=O4}   => {loan_amount_check=1} 0.01697331  0.5853491 1.65721
```

The above rule states that higher loan amount is associated with lesser number of young working population and more number of old working population.

The support value indicates that 1% of all the countries that have high old working population and low young working population also have high loan amount. The confidence score indicates that 58% of the countries that have high old working population and low young working population have high loan amount. The lift value means that this scenario is 1.6 times more likely to occur in the data.

```
inspect(rules.pruned[31])
```

```
##   lhs                         rhs                       support confidence     lift
## 1 {borrowers.gender=M,
##    working_pop_young=Y1} => {loan_amount_check=1} 0.04266757  0.5757587 1.630058
```

This rule suggests that the countries with low young male working population have high loans. The support value indicates that out of all the countries 4% of the countries borrow high loan amount and have low young male working population. The confidence score indicates that the probability of transactions that low young male working population also borrow high loans is 57%. The lift value of 1.6 means that this scenario is 1.6 times more likely to occur in the data.

```
inspect(rules.pruned[101])
```

```
##    lhs                       rhs                     support   confidence
## 77 {borrowers.gender=M} => {loan_amount_check=1} 0.1113929 0.4484048
##    lift
## 77 1.2695
```

This rule states that the high loan amount is associated with male borrowers. This is interesting because we have analyzed earlier that the total percentage of loans borrowed by males is low as compared to females but in this association rule we observe that the loan amounts borrowed by males is higher. The support value indicates that out of all loan amounts 11% of each transaction, the loan amounts are high and are taken by males. The confidence score indicates that 44% of the males borrow high loan amount. The lift value of 1.2 indicates that this scenario is 1.2 times more likely to occur in the data.

5.2 Relationship between status and other variables

We select `defaulted` and `in_repayment` statuses which represent the loans that are not paid back completely. We analyze the risk that is associated with higher loan amounts, therefore we filter loans that are above the average value.

We create a new data frame after filtering `defaulted` and `in_repayment` status and select the required variable for comparison.

```
avgloan_df <- loan.df[as.numeric(loan.df$loan_amount) > mean(as.numeric(loan.df$loan_amount)),]
finaldf_status <- avgloan_df %>%
  select(status,sector,
            location.country,borrowers.gender) %>%
  filter(status == "defaulted" | status == "in_repayment")

finaldf_status <- data.frame(finaldf_status)
```

For applying asociation rules, we convert all the variables to factors.

```
finaldf_status["status"] <- as.factor(finaldf_status$status)
finaldf_status["sector"] <- as.factor(finaldf_status$sector)
finaldf_status["location.country"] <- as.factor(finaldf_status$location.country)
finaldf_status["borrowers.gender"] <- as.factor(finaldf_status$borrowers.gender)
```

We control the number of results by setting confidence and support values to 0.01 as only one status is predicted for higher values. We set `defaulted` and `in_repayment` status as rhs and other variables as lhs. The result is sorted by the lift value.

```
apriori.appearance = list(rhs=c("status=defaulted","status=in_repayment"),
                          default="lhs")
apriori.parameter = list(support=0.01,confidence=0.01,minlen=1,maxlen=3)

rules_status = apriori(finaldf_status, parameter =
                         apriori.parameter,appearance = apriori.appearance)

rules_status.sorted <- sort(rules_status, by = "lift")
```

The redundant rules are identified and removed using the below code.

```
subset.matrix <- is.subset(rules_status.sorted, rules_status.sorted)
subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA
redundant <- colSums(subset.matrix, na.rm=T) >= 1
rules_status.pruned <- rules_status.sorted[!redundant]
```

```
inspect(rules_status.pruned[1])
```

```
##   lhs                          rhs                  support confidence     lift
## 1 {location.country=Togo,
##    borrowers.gender=F}    => {status=defaulted} 0.0114908  0.9731638 7.469518
```

The above rule states that Togo has the higher number of female borrowers that are associated with `defaulted` loan status. The support value indicates that for 1% of all the loans, female borrowers in Togo have `defaulted` loan status. The confidence score indicates that 97% of the female borrowers in Togo have `defaulted` loan status. The lift value of 7.46 indicates that it is 7.4 times more likely to occur in the data.

Similar to Togo, Pakistan also has higher number of female borrowers who are associated with `defaulted` loan status.

```
inspect(rules_status.pruned[4])
```

```
##    lhs                              rhs                 support    confidence
## 19 {location.country=Pakistan} => {status=defaulted} 0.01227465 0.5768025
##    lift
## 19 4.427247
```

The support value indicates that for 1% of all the loans borrowed in Pakistan have `defaulted` loan status. The confidence score indicates that 63% of the loan status in Pakistan is `defaulted`. The lift value of 4.4 indicates that this scenario 4.4 times more likely to occur in the data.

```
inspect(rules_status.pruned[10])
```

```
##   lhs                        rhs                  support confidence     lift
## 1 {sector=Agriculture,
##    borrowers.gender=M} => {status=defaulted} 0.01369223   0.151476 1.162654
```

The above rule identifies the relationship between the loans borrowed by males in agriculture sector and `defaulted` loan status. The support value indicates that in 1% of the total loans, males take loans for agriculture sector and have `defaulted` loan status. The confidence score indicates that 15% of the loans in agriculture sector are borrowed by males and have `defaulted` loan status. The lift value of 1.1 indicates that this scenario is 1.1 times more likely to occur in the data.

```
inspect(rules_status.pruned[37])
```

```
##    lhs                       rhs                 support    confidence
## 51 {borrowers.gender=M} => {status=defaulted} 0.04544621 0.1459326
##    lift
## 51 1.120106
```

This rule identifies the relationship between the male borrowers and `defaulted` loan status. The support value indicates that 4% of the total loans are taken by males and are `defaulted`. The confidence score indicates that 14% of the loans that are taken by males are in `defaulted` loan status. The lift value of 1.1 indicates that this scenario is 1.1 times more likely to occur in the data.

5.3 Relationship between GDP and other variables

We observe the relationship between GDP and the other variables. We use GDP per annual growth rate for this analysis. This indicator from the WDI dataset is already merged with the Kiva dataset and the resultant data frame is `final1df`.

We group the GDP values into four quantiles - Q1 is the GDP growth rate between 0% and 25% of the total growth rate, Q2 is the GDP growth rate between 25% and 50% of the total growth rate, Q3 is the GDP growth rate between 50% and 75% of the total growth rate and Q4 is the GDP growth rate between 75% and 100% of the total growth rate. The GDP growth rate is compared with loan amount, gender and country.

Similarly, the loan amount is grouped into four values - "low", "medium", "high" and "very high" based on the value of the loan amount.

```
final1df <- finaldf[!is.na(finaldf$NY.GDP.MKTP.KD.ZG),]
final1df["loan_amount"] <- as.numeric(final1df$loan_amount)
finaldf_GDP <- final1df %>%
  mutate(GDP_quant = cut(NY.GDP.MKTP.KD.ZG,
                         quantile (NY.GDP.MKTP.KD.ZG, c (0, .25, .5, .75, 1)),
                         labels=c('Q1','Q2','Q3','Q4'))) %>%
  mutate(loan_quant = cut(loan_amount,
                          quantile (loan_amount, c (0, .25, .5, .75, 1)),
                          labels=c('low','medium','high','veryhigh'))) %>%
  select(sector,loan_quant,borrowers.gender,GDP_quant)
```

To apply the rules of association, the variables are converted to factors.

```
finaldf_GDP["sector"] <- as.factor(finaldf_GDP$sector)
finaldf_GDP["borrowers.gender"] <- as.factor(finaldf_GDP$borrowers.gender)
finaldf_GDP["GDP_quant"] <- as.factor(finaldf_GDP$GDP_quant)
finaldf_GDP["loan_quant"] <- as.factor(finaldf_GDP$loan_quant)
```

We control the number of results by setting support as 0.01, confidence as 0.2 and setting the maximum variables to three.

We set the rhs value to different quantiles of GDP growth rate (Q1,Q2,Q3,Q4) and lhs to default. The result is sorted by lift value.

```
apriori.appearance = list(rhs=c("GDP_quant=Q1",
                                "GDP_quant=Q2",
                                "GDP_quant=Q3",
                                "GDP_quant=Q4"),
                          default="lhs")
apriori.parameter = list(support=0.01,confidence=0.2,minlen=1,maxlen=3)
rules_GDP = apriori(finaldf_GDP, parameter =
                      apriori.parameter,appearance =   apriori.appearance)
rules_GDP.sorted <- sort(rules_GDP, by = "lift")
```

The redundant rules are found and removed using the below code.

```
subset.matrix <- is.subset(rules_GDP.sorted, rules_GDP.sorted)
subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA
redundant <- colSums(subset.matrix, na.rm=T) >= 1
rules_GDP.pruned <- rules_GDP.sorted[!redundant]
```

The following rule identifies the relationship between the loan amount and males with respect to GDP.

```
inspect(rules_GDP.pruned[7])
```

```
##   lhs                       rhs               support confidence     lift
## 1 {loan_quant=veryhigh,
##    borrowers.gender=M}  => {GDP_quant=Q4} 0.02608095  0.3196045 1.280852
```

The rule states that in countries where males are associated with higher loans, the GDP growth rate is high. The support value indicates that 2% the countries have high loan amount borrowed by males and high GDP growth rate. The confidence value indicates that countries where 31% of the males borrowed larger loan amount have high GDP growth rate. The lift value indicates that this scenario is 1.28 times more likely to occur in the data.

We have stated earlier that males borrow low percentage of loan amount when compared to females and they tend to take larger value of loan amount. Here, we have arrived at an interesting finding that the GDP growth rate of the countries where males borrow higher value of loan amount is high.

```
inspect(rules_GDP.pruned[10])
```

```
##    lhs                                rhs            support    confidence
## 90 {sector=Retail,loan_quant=low} => {GDP_quant=Q4} 0.02088465 0.308412
##    lift
## 90 1.235997
```

The rule suggests that countries where lesser loan amount is associated with retail sector have higher GDP growth rate. The support value indicates that 2% of all the countries that have low retail sector loans have high GDP growth rate. The confidence value states that about 29% of countries that have retail sector requiring lesser loan amount have high GDP growth rate. The lift value indicates that this scenario is 1.23 times more likely to occur in the data.

The following rule identifies the relationship between loan amount and female with respect to GDP.

```
inspect(rules_GDP.pruned[16])
```

```
##   lhs                      rhs               support confidence     lift
## 1 {loan_quant=low,
##    borrowers.gender=F} => {GDP_quant=Q1} 0.06361991  0.2931788 1.160637
```

The rule states that in countries where females are associated with lesser loans, the GDP growth rate is low. The support value indicates that 6% of all the countries have lesser loan amount borrowed by females and lower GDP growth rate. The confidence value indicates that countries where 29% of the females borrows lesser loan amounts have low GDP growth rate. The lift value indicates that this scenario is 1.16 times more likely to occur in the data. Even though females take less value of loan amount when compared to male, the above rules states that GDP growth rate of the countries where females borrow lesser value of loan amount is low.

5.4 Relationship between country's loan amount and its labor force

We observe the relationship between the country's labor force and the other variables like loan amount and gender.

This indicator from the WDI is already merged with the Kiva dataset and the resultant data frame is `final2df`.

We group the labor force values into four quantiles, Q1 is the labor force between 0% and 25% of the total labor force, Q2 is the labor force between 25% and 50% of the total labor force, Q3 is the labor force between 50% and 75% of the total labor force and Q4 is the labor force between 75% and 100% of the total labor force.

The loan amount is grouped into four values - "low", "medium", "high" and "very high" based on the value of amount.

```
final2df["loan_amount"] <- as.numeric(final2df$loan_amount)
finaldf_LF <- final2df %>%
  mutate(LF_quant = cut(SL.TLF.TOTL.IN,
      quantile (SL.TLF.TOTL.IN, c (0, .25, .5, .75, 1)),
          labels = c('Q1','Q2','Q3','Q4'))) %>%
  mutate(loan_quant = cut(loan_amount,
      quantile (loan_amount, c (0, .25, .5, .75, 1)),
          labels=c('low','medium','high','veryhigh'))) %>%
  select(loan_quant,borrowers.gender,LF_quant)
```

To apply the rules of association, the variables are converted to factors.

```
finaldf_LF["location.country"] <- as.factor(finaldf_LF$location.country)
finaldf_LF["sector"] <- as.factor(finaldf_LF$sector)
finaldf_LF["borrowers.gender"] <- as.factor(finaldf_LF$borrowers.gender)
finaldf_LF["LF_quant"] <- as.factor(finaldf_LF$LF_quant)
finaldf_LF["loan_quant"] <- as.factor(finaldf_LF$loan_quant)
```

We control the number of results by setting support as 0.01, confidence as 0.2 and maxlen as 2.

The different quantiles of labor force are set to rhs and the other variables are set to lhs. The result is sorted by the lift value.

```
apriori.appearance = list(rhs=c("LF_quant=Q1",
      "LF_quant=Q2","LF_quant=Q3", "LF_quant=Q4"),default="lhs")
apriori.parameter = list(support=0.01,confidence=0.2,minlen=1,maxlen=3)
rules_LF = apriori(finaldf_LF, parameter =
                  apriori.parameter, appearance =  apriori.appearance)
rules_LF.sorted <- sort(rules_LF, by = "lift")
```

The redundant rules are found and removed using the below code.

```
subset.matrix <- is.subset(rules_LF.sorted, rules_LF.sorted)
subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA
redundant <- colSums(subset.matrix, na.rm=T) >= 1

rules_LF.pruned <- rules_LF.sorted[!redundant]
```

```
inspect(rules_LF.pruned[2])
```

```
##    lhs                                    rhs           support
## 41 {loan_quant=low,borrowers.gender=F} => {LF_quant=Q4} 0.1000844
##    confidence lift
## 41 0.4701914  1.881684
```

The above rule states that the countries with high labor force have low female borrowers. The support value indicates that 10% of the countries have low female borrowers and high labor force. The confidence score shows that 47% of the countries with low female borrowers have high labor force. The lift value of 1.8 indicates that this scenario is 1.8 times more likely to occur in the data.

As expected, the opposite scenario is also true as observed in the next association rule.

```
inspect(rules_LF.pruned[5])
```

```
##   lhs                      rhs              support confidence     lift
## 1 {loan_quant=high,
##    borrowers.gender=M} => {LF_quant=Q1} 0.02595728  0.3848512 1.526804
```

The rule states that the countries with low labor force have high male borrowers. The support value indicates that 2% of the countries have high male borrowers and low labor force. The confidence score shows that 38% of the countries with high male borrowers have low labor force. The lift value of 1.8 indicates that this scenario is 1.8 times more likely to occur in the data.

6. Conclusion

Our analysis provides insights to the lenders that could influence their lending decisions and describes the relationship of the loan amount with other variables from the WDI dataset. This report provides better insights to the lenders to make intelligent business decisions by checking the credit worthiness of the borrowers and history of defaults. It also aids the lender in choosing potential countries (by GDP and labor forces at play) that warrant investments and guarantee returns. The analysis further drills down to various industrial sectors and their past performance that would enable lenders to choose sectors wisely.

The analysis successfully answers the subsequent research questions.

1. What is the distribution of loan amount across countries and what is its relationship with country's GDP, working population and labor force?

In our analysis we concluded that the developing countries have the largest loan amounts. Specifically we observed that Peru, Cambodia and Philippines are the largest borrowers in the world. We analyzed the relationship between the loan amounts of these countries(top 10 countries with largest loan amounts) with their GDP over five years and observed that though the GDP of these countries are not consistent but there is a general trend that as the loan amount increases, the GDP also increases. From the association rules, we found that a country with lesser number of young working population and a higher number of old working population is more likely to have higher loan amounts.

2. What are the factors that can help lenders make better decisions?

While analyzing the Kiva dataset we observed that the largest number of borrowers are in the agriculture, food and retail sectors. We also observed that the education and health sectors have the largest deficit of loans that are not paid.

We further analyzed the difference between the funded date and paid back date of the top loan borrower countries and concluded that Philippines has the lowest turnaround time and Kenya & Cambodia have the highest. Hence according to our analysis, out of the fifteen sectors, lenders should lend money in the agriculture, food and retail sectors. Lending money to Kenya and Cambodia may not entail timely repayment as they have the highest turnaround time. While applying association rules we were unable to find any significant association that could help the lenders make better decision while lending money.

3. What is the gender wise participation in Kiva's loans?

While performing this research, we observed that 70% of the total loans are taken by females but they tend to take out smaller loans than males in all sectors, except in construction and transportation sector where males take 59% and 64% respectively. From the association rules we observed that Togo has the highest number of female loan defaulters. We also observed that the countries with strong/high labor force have lesser number of female borrowers and the countries with a weaker labor force have higher number of male borrowers.