INFO7374 - Kiva & World Development Indicators

Bowei Wang, Dongyue Li, Sarthak Agarwal, Sriram Chandramouli

7 June 2016

Contents

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 microlending 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 already disbursed by the field partner. Field partners then collect repayments from the borrowers. Some microfinance institutions may charge an interest rate but around 60% of the partners are non-profits. The amount is then repaid to the Kiva lender.

Every year 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 loans. The main focus 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 to make better decisions thereby helping in the economic development of the countries and aiding poverty alleviation. In this report we analyze the loan amounts, and their relationship with other variables such as sectors, genders and countries. We also analyze the loan amount fundings and the repayment trends for these loans.

We further analyze the loans funded across countries with their GDP and provide information about the countries that need immediate attention.

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.

2.1 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 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	Economic sector of the loan entry
`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` [to be removed]	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 dataframe,
3. load the WDI dataset into dataframe,
4. create new variables,
5. merge Kiva and WDI dataframe

3.1 Load libraries

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 dataframe

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

We create a function which generates a dataframe 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 dataframe. 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 four cores and use the function `clusterExport` which provides several ways to parallelize computations.

```
cl <- makeCluster(4)
clusterExport(cl,
              c('dfrow.from.list','isValidJSON', 'data.folder', '%>%',
                'list.select','bind_rows','fromJSON',
                'readJSONFileIntoDataFrame'))
```

We use parallel version of Lapply to apply `readJSONFileIntoDataFrame` function on JSON files, then stop 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 dataframe

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")
```

We read the WDI data for this code into the `df` dataframe.

```
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
```

We observe that the loan data is available from 2006 to 2013 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, which is character 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 dataframes

We use the `cleanedyear` and `location.country` variables in loan dataframe and join them on the basis of `country` and `year` variables in the WDI dataframe. 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 from the GDP annual growth rate.

```
final1df <- finaldf[!is.na(finaldf$NY.GDP.MKTP.KD.ZG),]
```

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

Our analysis is primarily focuses on the loan amount and its relationship with other variables. 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.

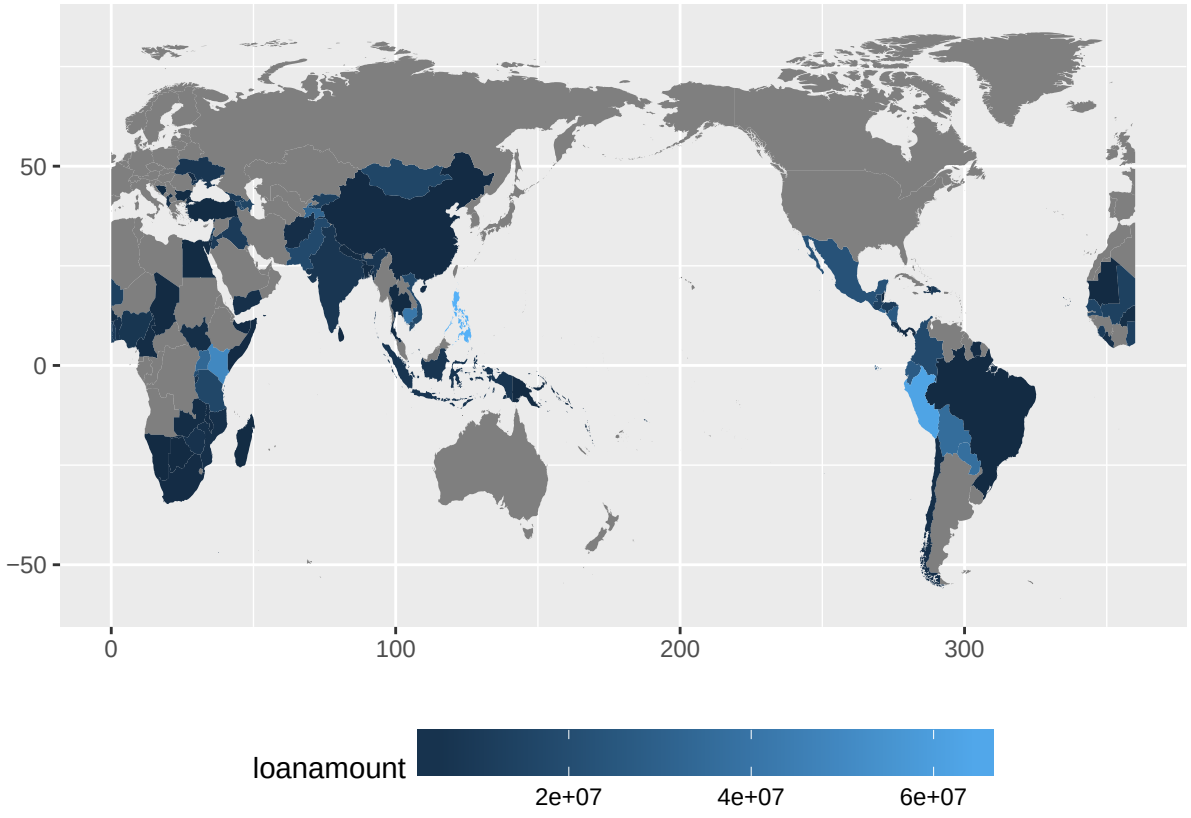
```
loan_amount_by_country <- condSum(loan.df$loan_amount_numeric,
                                  loan.df$countryF,
                                  na.rm = FALSE)
```

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 legend position to 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 loan requirement is in 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 loan requirements 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 total loan amount for every country and year.

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

We create a dataframe 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 dataframe 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 dataframe 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 Phillipines are the countries that have the largest loan requirement. Hence lenders can filter the loan postings on Kiva website for these countries.

Also, it is important to note that for Cambodia, Bolivia and Nicaragua loan requirement increases till 2010 and then decreases in 2011. This can be due to economic stagnation or policy changes within the nation.

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.

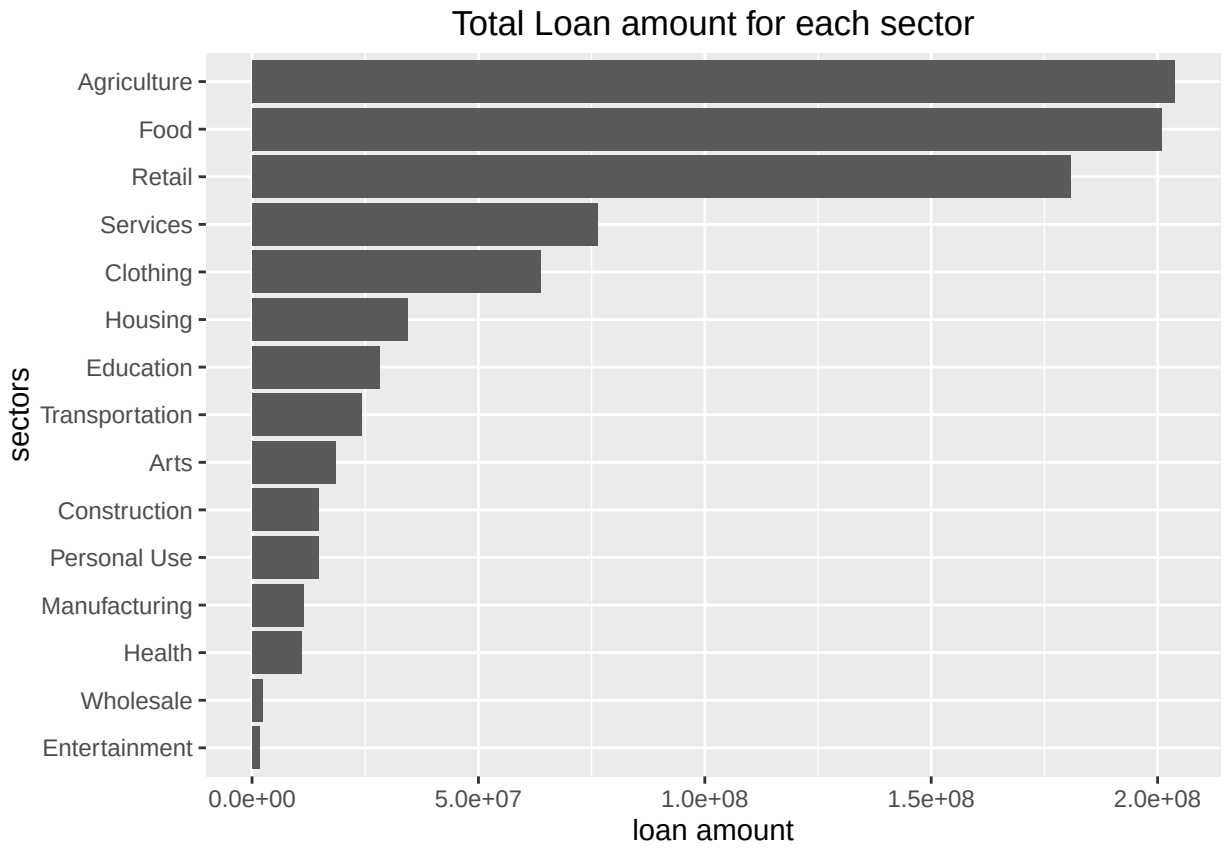
```
loan_amount_by_sector <- condSum(loan.df$loan_amount_numeric,
                                 loan.df$sector_F,na.rm = FALSE)
```

We create the `loan_amount_by_sector_df` dataframe 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.

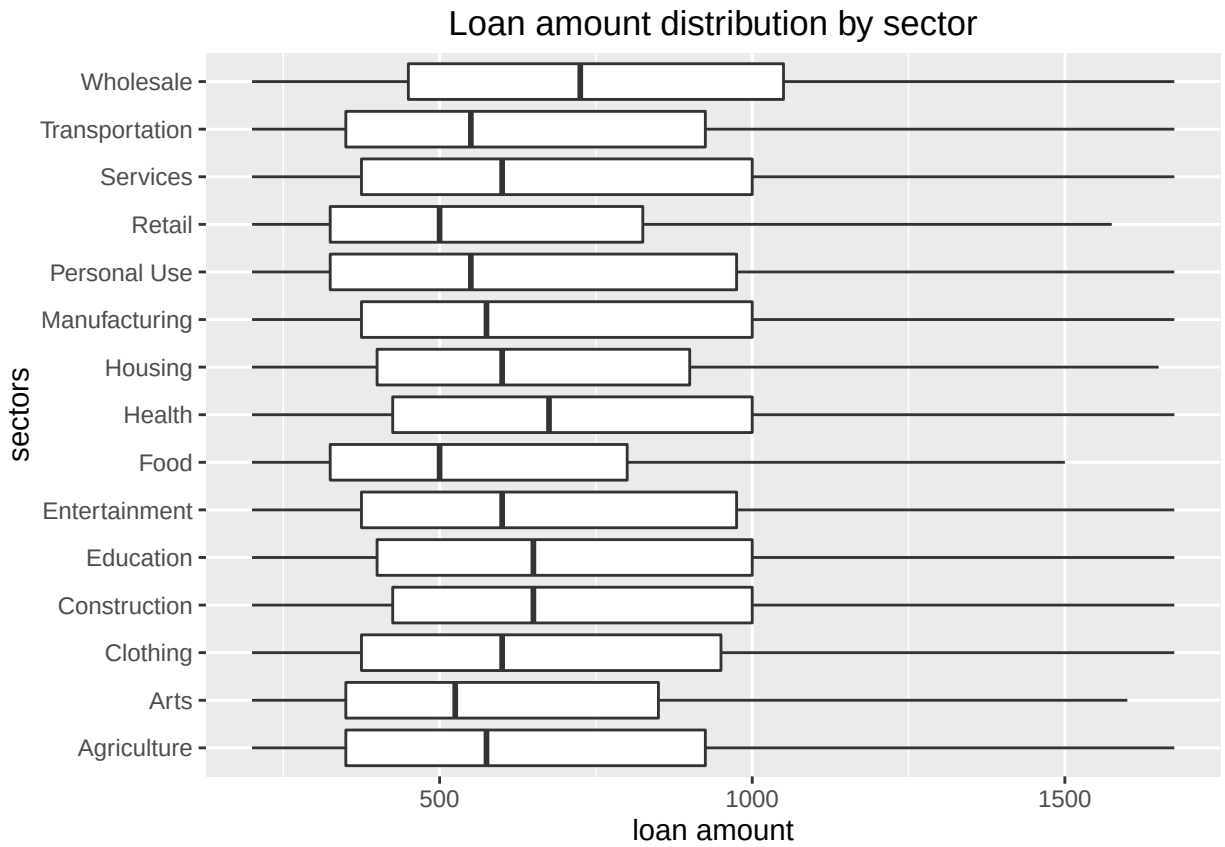
```
loan_amount_by_sector_df %>%
  ggplot (aes(x = reorder(names(loan_amount_by_sector),
                          loan_amount_by_sector), y = loan_amount_by_sector)) +
  ggtitle("Total Loan amount for each sector") +
  xlab("sectors") +
  ylab("loan amount") +
  geom_bar(stat = "identity") +
  coord_flip()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



We observe that people are asking for the highest 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 loan requirement.

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()
```



We see that wholesale has the highest average loan amount and has low variation as compared to the other sectors. Food and entertainment sectors have high variation.

4.4 `borrowers.genderF`

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

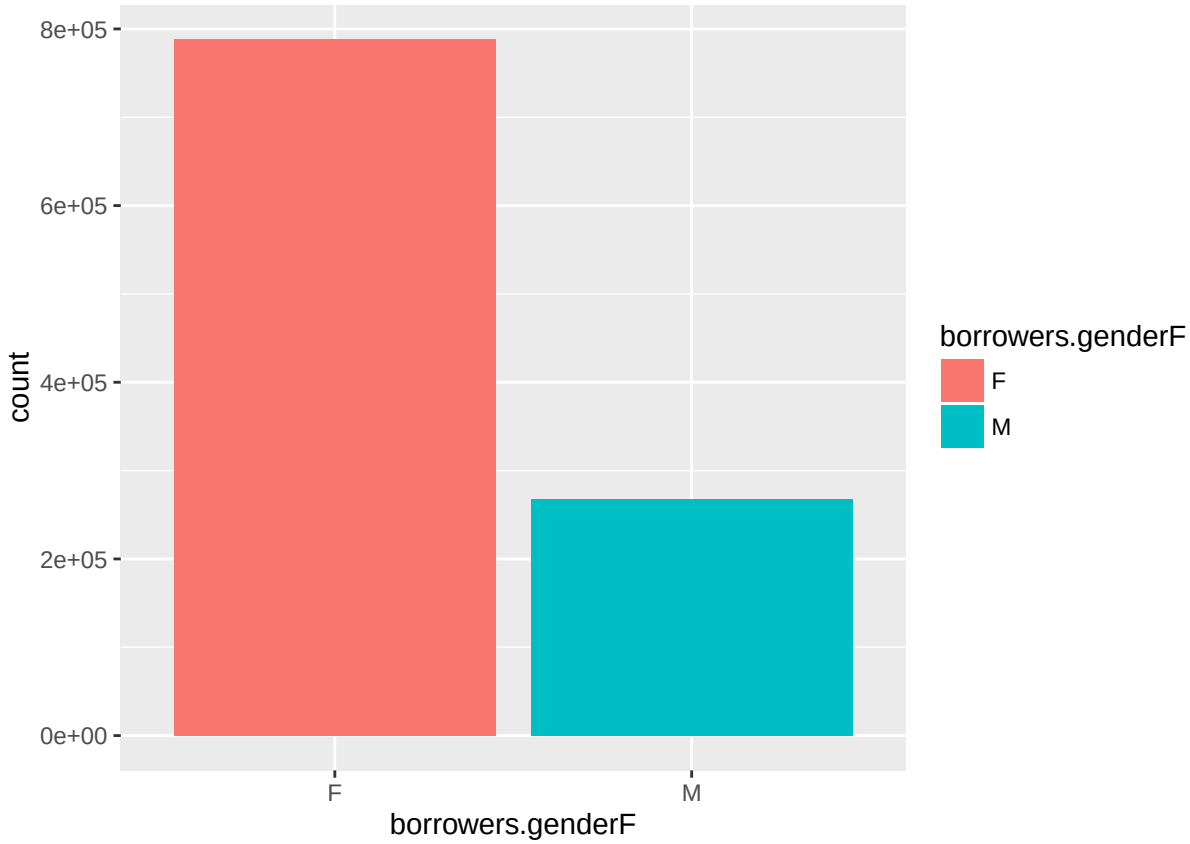
```
loan.df$borrowers.genderF <- factor(loan.df$borrowers.gender)
```

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))
```



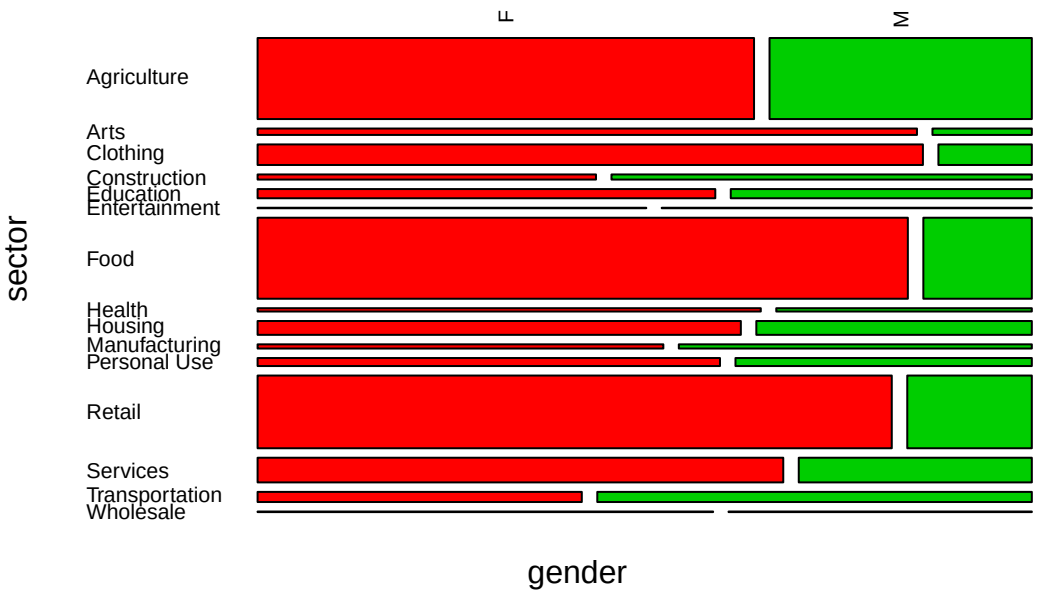
We observe that around 70 percent of borrowers on the Kiva website are females.

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. This will enable the lenders to help the society in achieving gender equality.

Below code plots a mosaic graph which represents the gender on the x-axis, sector on y-axis and the width of the bars as loan amount.

```
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 70 percent of the total loans posted are by females. However, some sectors like construction, transportation and manufacturing have more loan postings by males.

4.6 Country's GDP and their loan amount requirement

We analyze the relationship between the total loan amount for the countries with higher loan requirement 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 dataframe. We observe that in some records `posted_date` is null but `loan_amount` is not null. We remove the records with null `posted_date` before adding the `loan_amount` by year.

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

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

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

We merge the above dataframe with the WDI dataframe by year and country. We change the 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 plot different graphs based on the countries and observe the relationship between GDP and loan amount over the years.

We merge `loan_amount_by_country_df_from2006_to2011_top10_long` dataframe which has the top 10 countries with the largest amount of loan requirement for the year range 2006-2011 and `GDP_df` dataframe created above 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` for better readability.

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

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

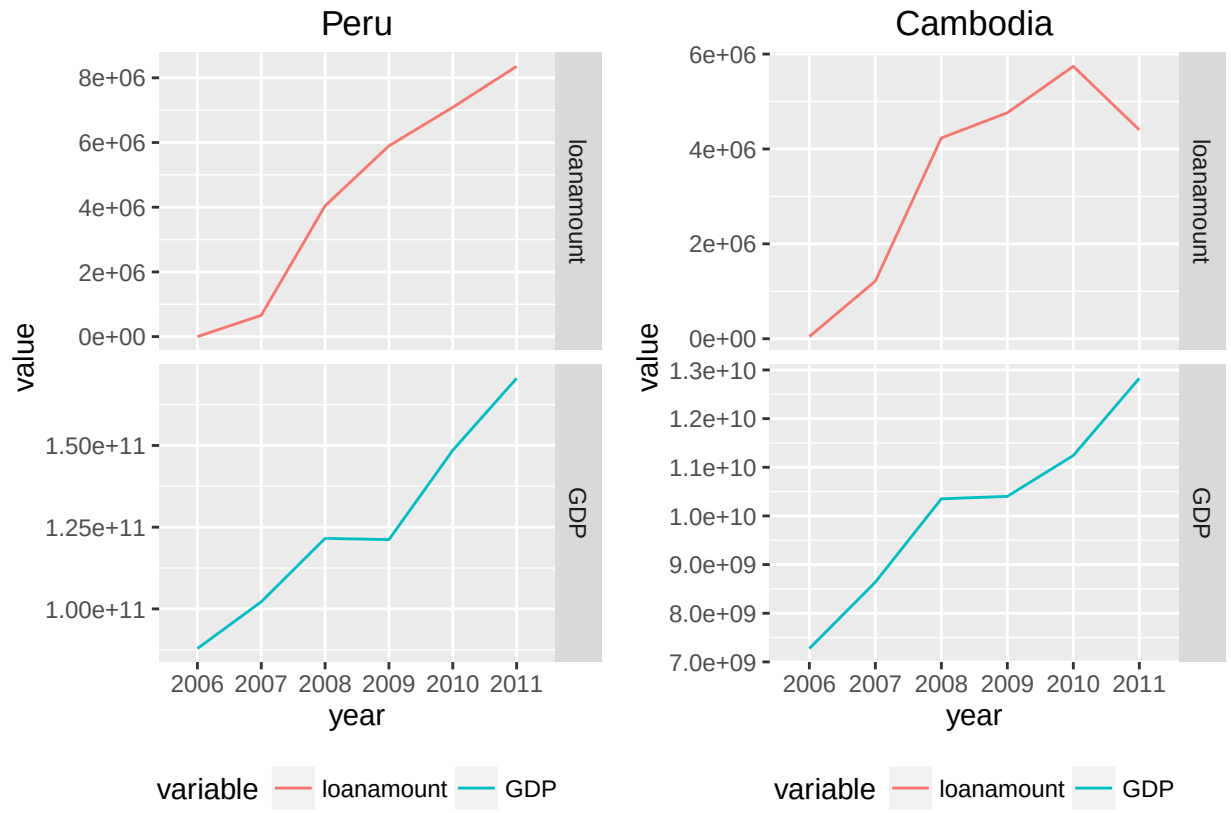
```
loan_GDP_df <- melt(loan_GDP_df[1:4], id=c("country","year"))
```

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)
```

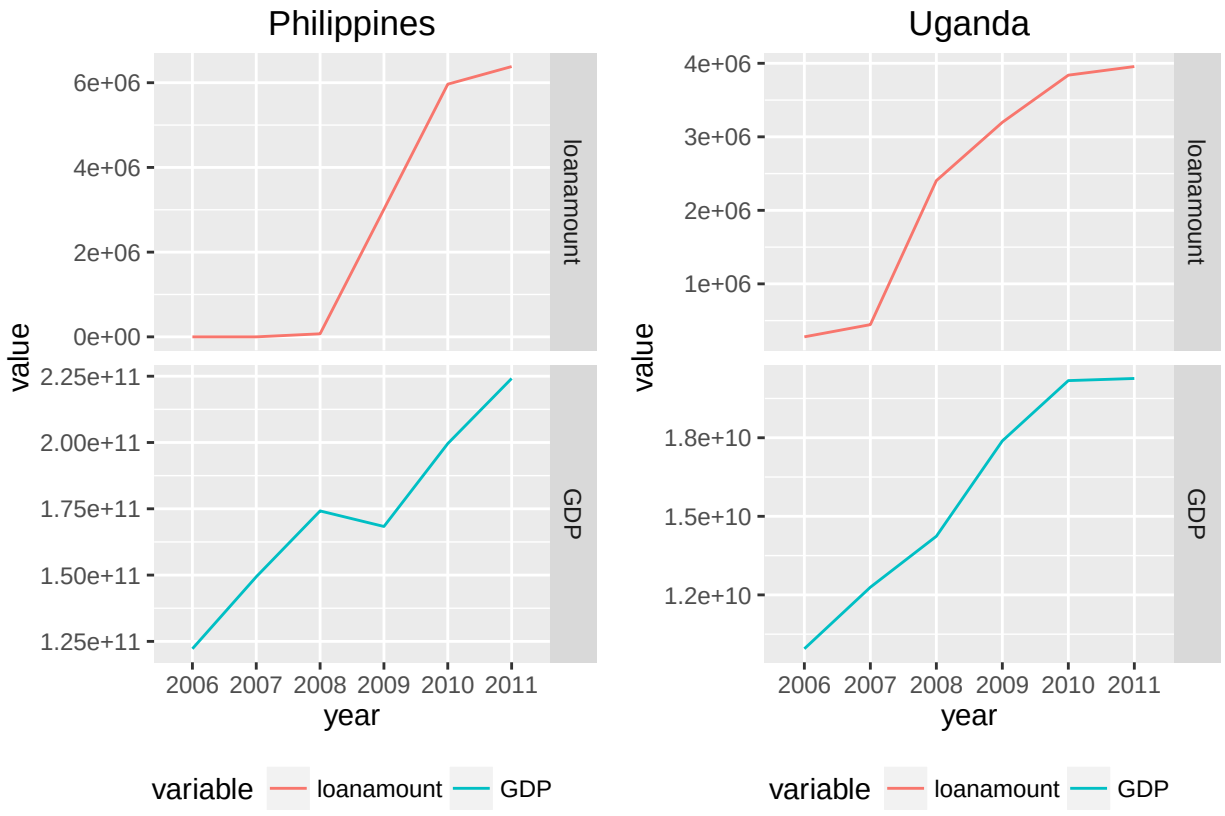
```
title <- top10_country_name[1]
loan_GDP_df_filter <- filter(loan_GDP_df, country %in% top10_country_name[1])
p1 = loan_GDP_df_filter %>%
  ggplot(aes(x=year, y=value, group=variable,
             colour=variable)) +
  geom_line()+
  ggtitle(title) +
  xlab("year") +
  ylab("value")
p1 = p1+theme(legend.position="bottom")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

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



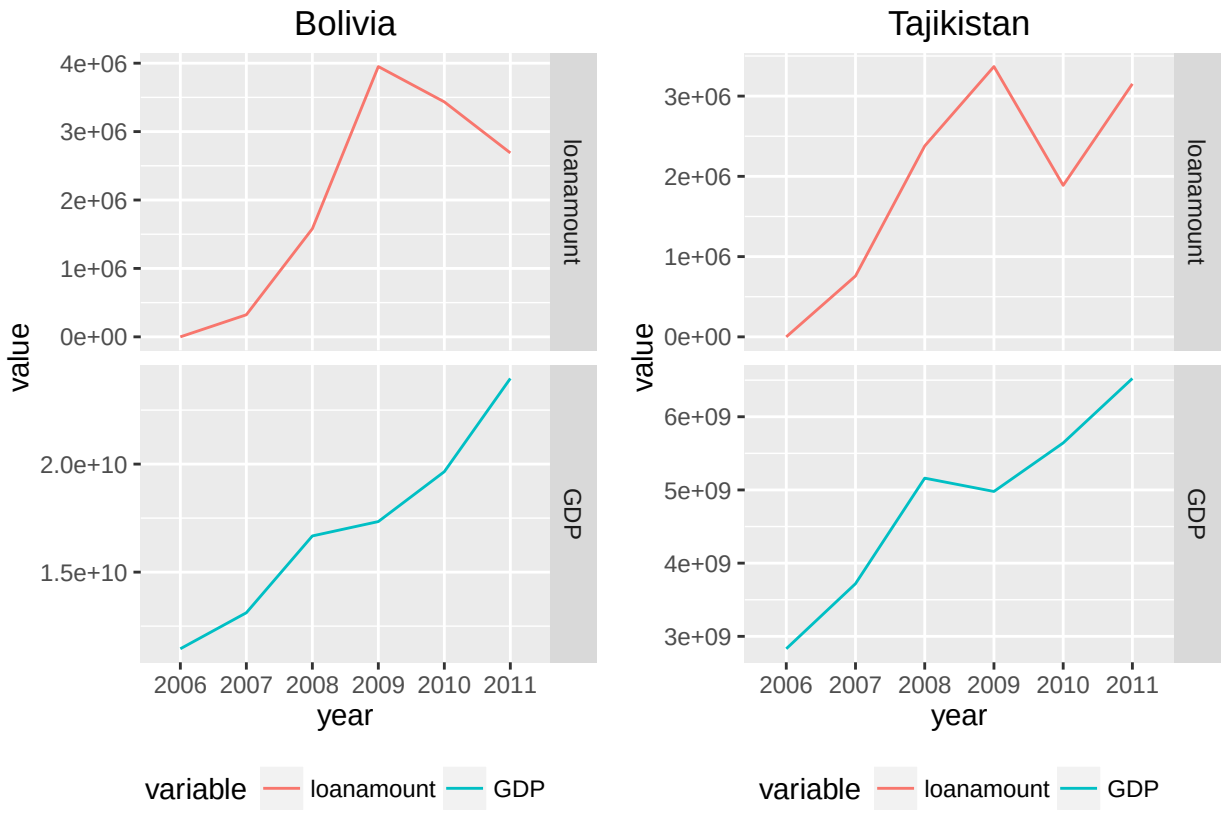
Loan requirement in Peru is increasing with its GDP, however Peru faced economic stagnation between 2008-2009. Loan requirement in Cambodia increases with its GDP till the year 2010 and decreases between 2010-2011.

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



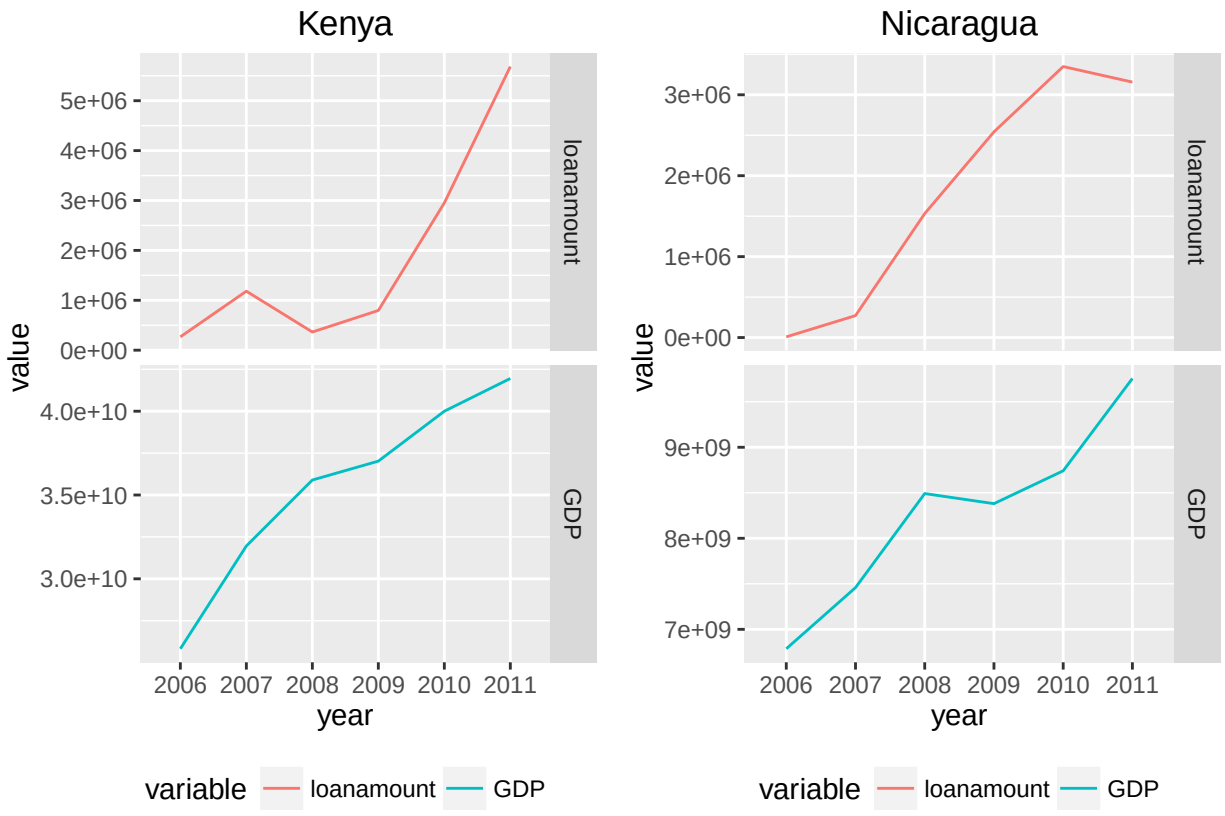
We observe that GDP of Phillipines declined between 2008 to 2009, however the requirement of loans constantly increased from 2006 to 2011. Loan requirement in Uganda is increasing with its GDP.

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



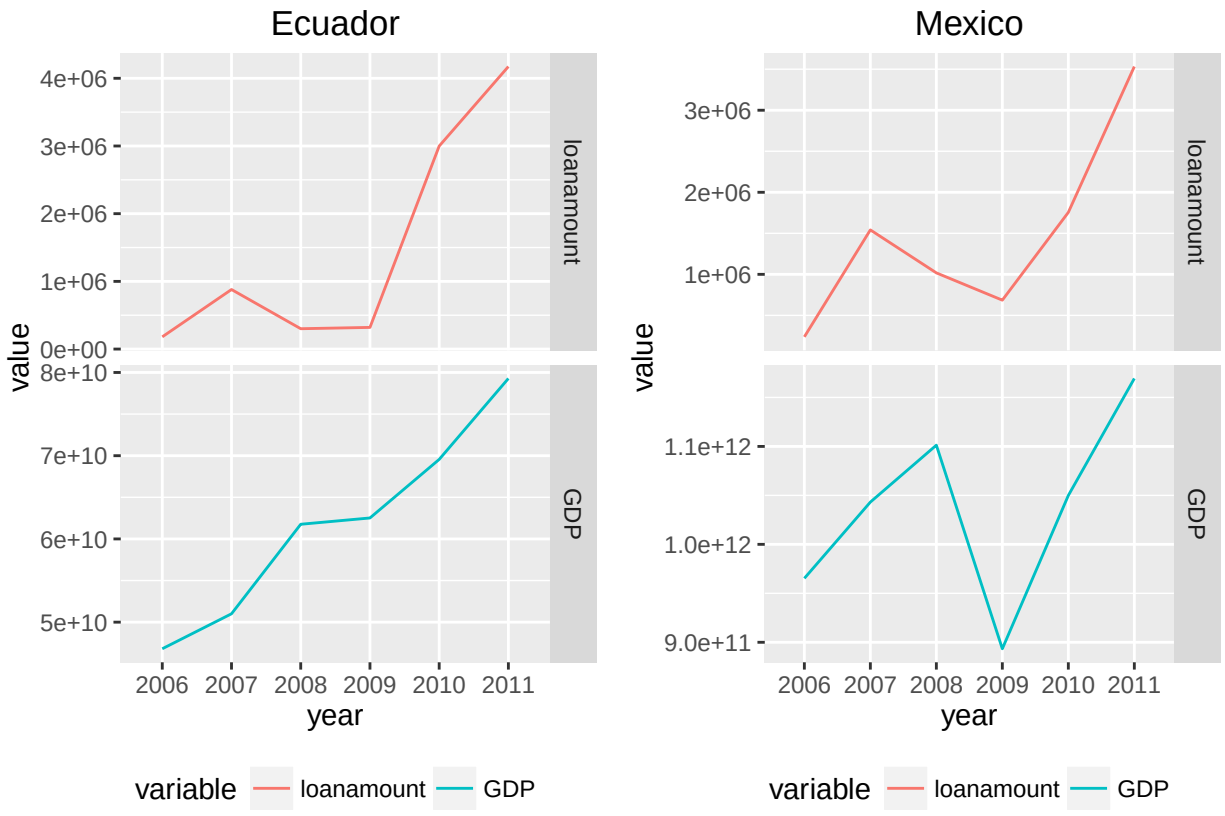
Loan amount requirement in Bolivia increases with the country's GDP but suddenly decreases between 2009-2011. We observe that Tajikistan's requirement for loans increases till 2009 then decreases till 2010 and then again increases in 2011.

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



We observe that Kiva loans got popular in Kenya only after strugling between 2006-2009. They skyrocketted between 2009-2011. Kenya's GDP increased between 2006-2011. Nicargua'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)
```



Comparing the countries which have the largest loan requirements we observe that the countries with the lowest GDP are asking for higher loan amounts.

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.

Convert `paid_amount` from character to numeric. Create `sel_finaldf` data frame from `loan.df` by selecting `loan_amount` and `sector`.

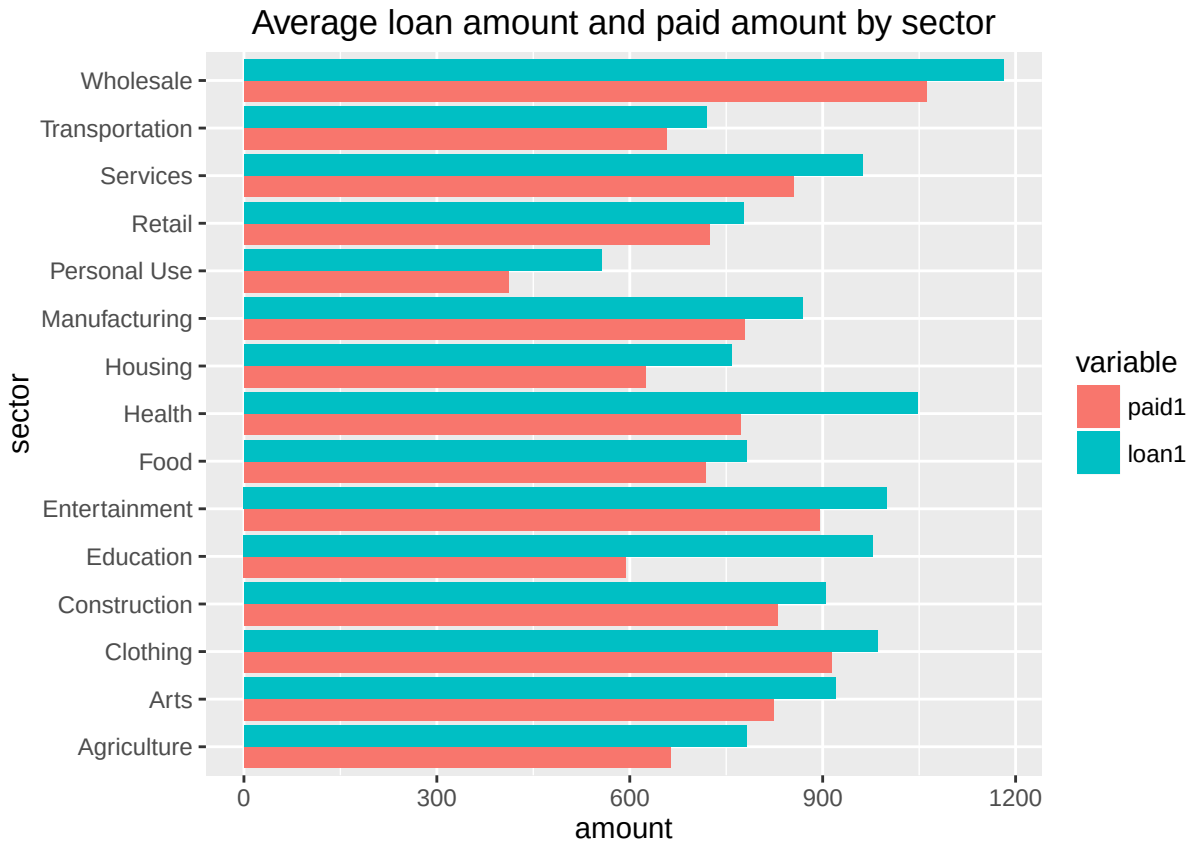
```
loan.df["paid_amount"] <- as.numeric(loan.df$paid_amount)

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()
```



Its impressive to observe that except education sector almost all sectors maintain a good loan repayment capability.

4.8 Loan amount, status and sector

To understand the working of loans, status plays an important role. 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
```

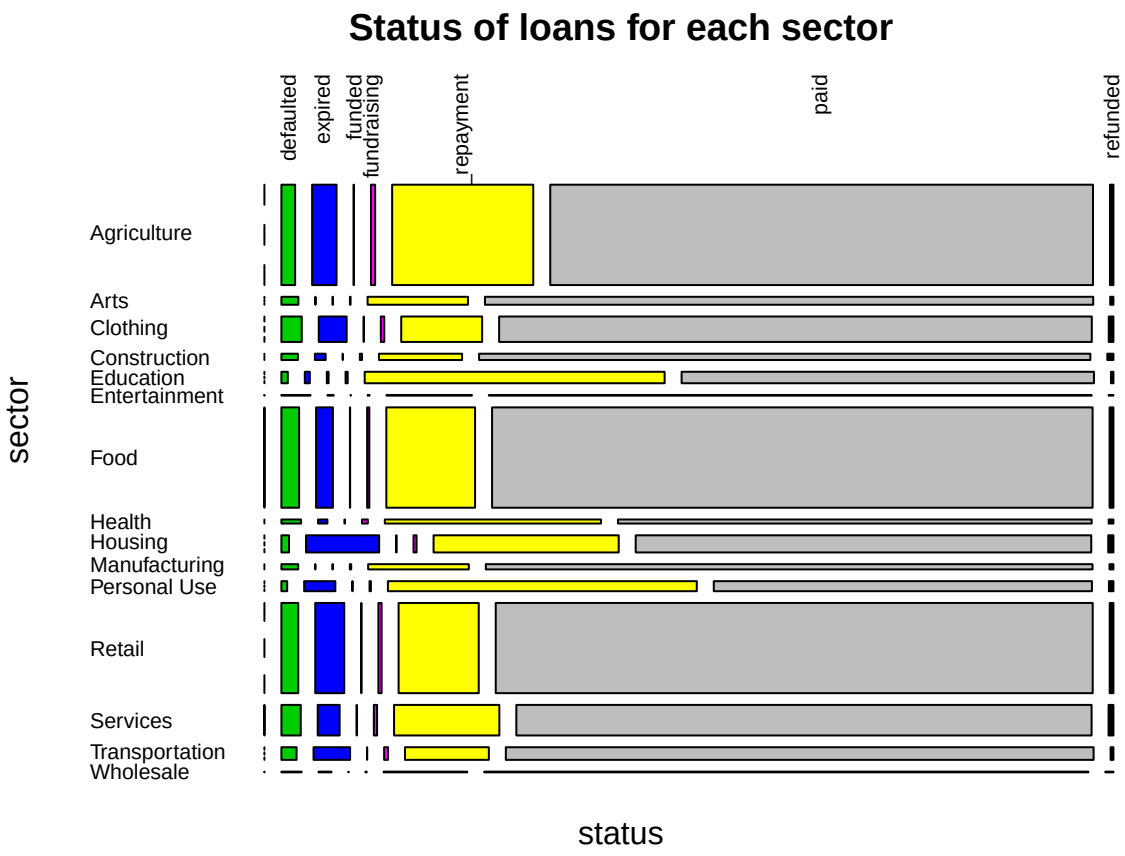
We see that there are 8 loan statuses:

- defaulted - Loans are never paid back after not being paid for 6 months. They are 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 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.

We observe that 77 percent of the loans have `paid` status and only 2 percent loans are `default`. This increases the confidence of lenders in Kiva's model.

We further visualize the status of loans based on sector.

```
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 some borrowers within the retail and food sector default in repayment.

4.9 The time from funded date to payback date by top 10 loan amount countries

Kiva does not provide any information to lenders about timline of loan repayment, but it's imperative for a lender to have knowledge of the timeline patters 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 the largest loan requirements.

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

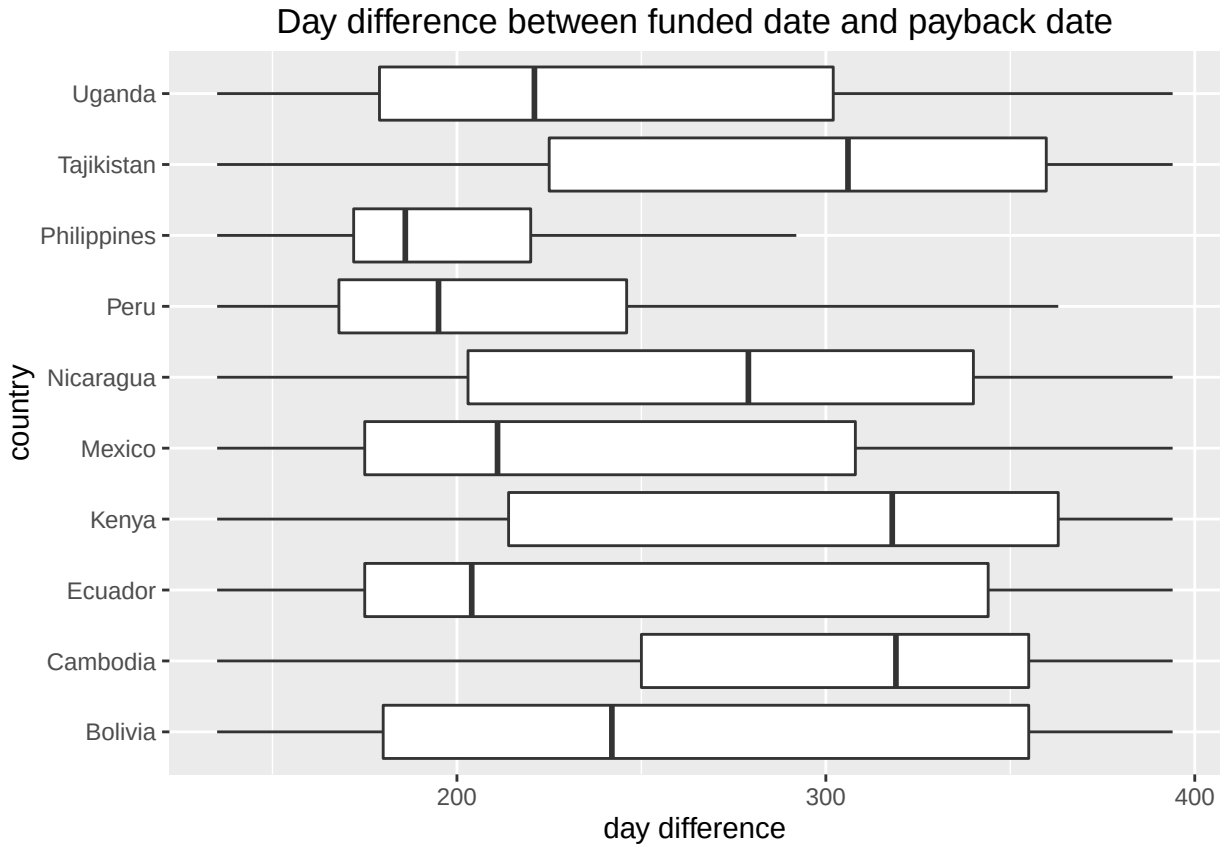
```
loan.df$day_diff <-
  as.numeric(loan.df$paid_date_cleaned - loan.df$funded_date_cleaned, "days")
```

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()
```



We observe that most of the loans are paid back within a year. We also conclude that Phillippines has the shortest average payback period whereas Kenya and Cambodia have the highest average payback period among the top 10 borrowers. Therefore, Phillipines is the fastest option to get back the money lended.

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 with other variables such as `country`, `sector`, `activity`, `status`, `lendercount`.

We convert `loan_amount` and `lender_count` into numeric.

```
loan.df["loan_amount"] <- as.numeric(loan.df$loan_amount)
loan.df["lender_count"] <- as.numeric(loan.df$lender_count)
```

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" refers to the loan amount which is greater than the average loan amount. We store this in the variable `loan_amount_check` and analyze the sector, country and activity based on this parameter.

We group the lender count into 4 quantiles - Q1, Q2, Q3, Q4. Q1 and create a variable called `lender_count_quant`. Q1 denotes lender counts which have values in between 0 and 25% of total lender count, Q2 in between 25% and 50% of total lender counts, Q3 in between 50% and 75% of total lender count and Q4 in between 75% and 100% of total lender count.

```
finaldf_loanamount <- loan.df %>%
  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'))) %>%
  select(lender_count_quant,status,activity,sector,
         location.country,loan_amount_check,borrowers.gender)

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

We convert the variables into factors to apply 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)
```

We control the number of results by setting support as 0.01, confidence as 0.5 and maxlen as 2.

To analyze loan amount, rhs value is set to `loan_amount_check` (0 and 1) and lhs is set 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 redundat 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]
inspect(rules.pruned)
```

```
##     lhs                                    rhs                     support confidence     lift
## 1   {lender_count_quant=Q4,
##      location.country=Bolivia}          => {loan_amount_check=1} 0.01137319  0.9809002 3.063337
## 2   {lender_count_quant=Q4,
##      location.country=Tajikistan}       => {loan_amount_check=1} 0.01038916  0.9752350 3.045644
## 3   {lender_count_quant=Q4,
##      activity=Retail}                   => {loan_amount_check=1} 0.01519378  0.9738937 3.041455
## 4   {lender_count_quant=Q4,
##      sector=Services}                   => {loan_amount_check=1} 0.02174598  0.9713561 3.033530
## 5   {lender_count_quant=Q4,
##      status=in_repayment}               => {loan_amount_check=1} 0.03992940  0.9704898 3.030825
```

```
## 6   {lender_count_quant=Q4,
##      borrowers.gender=M}              => {loan_amount_check=1} 0.07354058  0.9698917 3.028957
## 7   {lender_count_quant=Q4,
##      activity=Clothing Sales}         => {loan_amount_check=1} 0.01287909  0.9696970 3.028349
## 8   {lender_count_quant=Q4,
##      sector=Agriculture}              => {loan_amount_check=1} 0.05895918  0.9688464 3.025693
## 9   {lender_count_quant=Q4,
##      sector=Clothing}                 => {loan_amount_check=1} 0.01671893  0.9682708 3.023895
## 10  {lender_count_quant=Q4,
##      location.country=Cambodia}       => {loan_amount_check=1} 0.01330659  0.9681939 3.023655
## 11  {lender_count_quant=Q4}           => {loan_amount_check=1} 0.23544965  0.9669618 3.019807
## 12  {location.country=Lebanon}        => {loan_amount_check=1} 0.01103041  0.8279850 2.585785
## 13  {location.country=Paraguay,
##      borrowers.gender=F}              => {loan_amount_check=1} 0.01025821  0.7822894 2.443078
## 14  {status=expired,
##      borrowers.gender=M}              => {loan_amount_check=1} 0.01386601  0.7268827 2.270044
## 15  {location.country=Bolivia,
##      borrowers.gender=F}              => {loan_amount_check=1} 0.01004927  0.7189997 2.245425
## 16  {status=paid,
##      location.country=Paraguay}       => {loan_amount_check=1} 0.01045174  0.7140038 2.229823
## 17  {location.country=Paraguay}       => {loan_amount_check=1} 0.01214251  0.7083638 2.212209
## 18  {status=expired}                  => {loan_amount_check=1} 0.02355132  0.7031362 2.195884
## 19  {location.country=Bolivia}        => {loan_amount_check=1} 0.01376298  0.6628948 2.070210
## 20  {location.country=Ecuador,
##      borrowers.gender=F}              => {loan_amount_check=1} 0.01007719  0.5829342 1.820495
## 21  {location.country=Ecuador}        => {loan_amount_check=1} 0.01331237  0.5678962 1.773531
## 22  {location.country=Tajikistan}     => {loan_amount_check=1} 0.01463629  0.4965862 1.550831
## 23  {sector=Services,
##      borrowers.gender=M}              => {loan_amount_check=1} 0.01069630  0.4822242 1.505979
## 24  {lender_count_quant=Q1,
##      location.country=Philippines}    => {loan_amount_check=0} 0.10541382  0.9957797 1.464827
## 25  {lender_count_quant=Q1,
##      activity=Pigs}                   => {loan_amount_check=0} 0.01225034  0.9915829 1.458653
## 26  {lender_count_quant=Q2,
##      location.country=Philippines}    => {loan_amount_check=0} 0.05351233  0.9888266 1.454598
## 27  {lender_count_quant=Q1,
##      location.country=Kenya}          => {loan_amount_check=0} 0.03410416  0.9840529 1.447576
## 28  {lender_count_quant=Q1,
##      sector=Personal Use}             => {loan_amount_check=0} 0.01065683  0.9832104 1.446337
## 29  {lender_count_quant=Q1,
##      location.country=Cambodia}       => {loan_amount_check=0} 0.01217813  0.9829797 1.445997
## 30  {lender_count_quant=Q1,
##      status=in_repayment}             => {loan_amount_check=0} 0.04760621  0.9777334 1.438280
## 31  {lender_count_quant=Q1,
##      activity=Farming}                => {loan_amount_check=0} 0.02508514  0.9752564 1.434636
## 32  {activity=Food Production/Sales,
##      location.country=Philippines}    => {loan_amount_check=0} 0.01084747  0.9743989 1.433375
## 33  {lender_count_quant=Q1,
##      activity=General Store}          => {loan_amount_check=0} 0.03330499  0.9740644 1.432883
## 34  {activity=Pigs,
##      location.country=Philippines}    => {loan_amount_check=0} 0.01794560  0.9730605 1.431406
## 35  {lender_count_quant=Q2,
##      location.country=Kenya}          => {loan_amount_check=0} 0.02669406  0.9719534 1.429777
## 36  {lender_count_quant=Q1,
```

```
##      status=paid}                        => {loan_amount_check=0} 0.19674884  0.9691388 1.425637
## 37  {lender_count_quant=Q2,
##      status=paid}                        => {loan_amount_check=0} 0.17782209  0.9680975 1.424105
## 38  {sector=Food,
##      location.country=Philippines}       => {loan_amount_check=0} 0.04854114  0.9666373 1.421957
## 39  {lender_count_quant=Q2,
##      status=in_repayment}                => {loan_amount_check=0} 0.03419563  0.9661579 1.421252
## 40  {lender_count_quant=Q1,
##      sector=Agriculture}                 => {loan_amount_check=0} 0.05827555  0.9659883 1.421003
## 41  {lender_count_quant=Q1,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.21081897  0.9625789 1.415987
## 42  {lender_count_quant=Q2,
##      location.country=Cambodia}          => {loan_amount_check=0} 0.01056632  0.9607774 1.413337
## 43  {location.country=Philippines,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.16625441  0.9592456 1.411084
## 44  {lender_count_quant=Q1,
##      sector=Food}                        => {loan_amount_check=0} 0.06981146  0.9584650 1.409935
## 45  {status=paid,
##      location.country=Philippines}       => {loan_amount_check=0} 0.14962694  0.9576041 1.408669
## 46  {lender_count_quant=Q1,
##      activity=Food Production/Sales}     => {loan_amount_check=0} 0.01510616  0.9575221 1.408548
## 47  {sector=Agriculture,
##      location.country=Philippines}       => {loan_amount_check=0} 0.04355935  0.9573185 1.408249
## 48  {lender_count_quant=Q2,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.17261500  0.9557205 1.405898
## 49  {lender_count_quant=Q2,
##      activity=General Store}             => {loan_amount_check=0} 0.02251626  0.9548016 1.404546
## 50  {lender_count_quant=Q1}              => {loan_amount_check=0} 0.25738723  0.9547242 1.404433
## 51  {location.country=Philippines}       => {loan_amount_check=0} 0.18065287  0.9544941 1.404094
## 52  {lender_count_quant=Q2,
##      activity=Food Production/Sales}     => {loan_amount_check=0} 0.01149066  0.9511437 1.399166
## 53  {lender_count_quant=Q2,
##      sector=Food}                        => {loan_amount_check=0} 0.05581066  0.9507906 1.398646
## 54  {lender_count_quant=Q2,
##      location.country=Peru}              => {loan_amount_check=0} 0.02072728  0.9476998 1.394099
## 55  {lender_count_quant=Q2,
##      location.country=Nicaragua}         => {loan_amount_check=0} 0.01001942  0.9462581 1.391979
## 56  {lender_count_quant=Q2,
##      activity=Farming}                   => {loan_amount_check=0} 0.01870144  0.9449742 1.390090
## 57  {sector=Food,
##      location.country=Kenya}             => {loan_amount_check=0} 0.01923679  0.9444996 1.389392
## 58  {lender_count_quant=Q2,
##      sector=Agriculture}                 => {loan_amount_check=0} 0.05158278  0.9434192 1.387802
## 59  {lender_count_quant=Q2}              => {loan_amount_check=0} 0.22165874  0.9405274 1.383549
## 60  {sector=Food,
##      borrowers.gender=M}                 => {loan_amount_check=1} 0.01499351  0.4413457 1.378316
## 61  {activity=Fishing}                   => {loan_amount_check=0} 0.01032176  0.9000840 1.324055
## 62  {status=paid,
##      location.country=Kenya}             => {loan_amount_check=0} 0.05717502  0.8948852 1.316407
## 63  {location.country=Kenya,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.05323407  0.8938179 1.314837
## 64  {sector=Education}                   => {loan_amount_check=1} 0.01140593  0.4208320 1.314252
## 65  {sector=Retail,
##      borrowers.gender=M}                 => {loan_amount_check=1} 0.01468828  0.4201553 1.312139
```

```
## 66  {sector=Retail,
##      location.country=Kenya}            => {loan_amount_check=0} 0.01311884  0.8884324 1.306915
## 67  {location.country=Nigeria}          => {loan_amount_check=0} 0.01210495  0.8826792 1.298452
## 68  {borrowers.gender=M}                => {loan_amount_check=1} 0.10425551  0.4114828 1.285055
## 69  {location.country=Kenya}            => {loan_amount_check=0} 0.08065123  0.8718864 1.282575
## 70  {activity=Fish Selling,
##      borrowers.gender=F}                => {loan_amount_check=0} 0.01518896  0.8597667 1.264747
## 71  {status=paid,
##      activity=Fish Selling}             => {loan_amount_check=0} 0.01347509  0.8583783 1.262704
## 72  {activity=Pigs,
##      borrowers.gender=F}                => {loan_amount_check=0} 0.02102095  0.8550838 1.257858
## 73  {activity=Clothing Sales}           => {loan_amount_check=1} 0.01802167  0.4023431 1.256511
## 74  {status=paid,
##      activity=Pigs}                     => {loan_amount_check=0} 0.01865523  0.8509377 1.251759
## 75  {sector=Services}                   => {loan_amount_check=1} 0.02895195  0.4008078 1.251717
## 76  {activity=Fish Selling}             => {loan_amount_check=0} 0.01630009  0.8495509 1.249719
## 77  {activity=Pigs}                     => {loan_amount_check=0} 0.02350799  0.8416644 1.238118
## 78  {activity=General Store,
##      borrowers.gender=F}                => {loan_amount_check=0} 0.06443780  0.8385941 1.233601
## 79  {status=paid,
##      activity=General Store}            => {loan_amount_check=0} 0.05787212  0.8303172 1.221426
## 80  {status=paid,
##      location.country=Ghana}            => {loan_amount_check=0} 0.01287138  0.8280989 1.218162
## 81  {location.country=Ghana,
##      borrowers.gender=F}                => {loan_amount_check=0} 0.01331718  0.8243533 1.212653
## 82  {sector=Clothing}                   => {loan_amount_check=1} 0.02330676  0.3878234 1.211167
## 83  {lender_count_quant=Q3,
##      sector=Housing}                    => {loan_amount_check=0} 0.01056728  0.8191521 1.205001
## 84  {activity=Agriculture}              => {loan_amount_check=1} 0.01703282  0.3839476 1.199062
## 85  {activity=General Store}            => {loan_amount_check=0} 0.07064529  0.8144912 1.198145
## 86  {location.country=Ghana}            => {loan_amount_check=0} 0.01484908  0.8138688 1.197229
## 87  {lender_count_quant=Q3,
##      location.country=El Salvador}      => {loan_amount_check=0} 0.01182284  0.8114592 1.193685
## 88  {lender_count_quant=Q3,
##      location.country=Peru}             => {loan_amount_check=0} 0.01677093  0.8113471 1.193520
## 89  {location.country=Pakistan,
##      borrowers.gender=F}                => {loan_amount_check=0} 0.02070899  0.8000298 1.176872
## 90  {lender_count_quant=Q3,
##      status=paid}                       => {loan_amount_check=0} 0.15981486  0.7991728 1.175611
## 91  {activity=Retail}                   => {loan_amount_check=1} 0.02116056  0.3758937 1.173910
## 92  {activity=Grocery Store}            => {loan_amount_check=1} 0.01152147  0.3751803 1.171682
## 93  {lender_count_quant=Q3,
##      borrowers.gender=F}                => {loan_amount_check=0} 0.14115386  0.7893136 1.161108
## 94  {status=paid,
##      location.country=El Salvador}      => {loan_amount_check=0} 0.02400482  0.7877840 1.158858
## 95  {lender_count_quant=Q3,
##      sector=Food}                       => {loan_amount_check=0} 0.04368837  0.7805205 1.148173
## 96  {status=paid,
##      location.country=Nicaragua}        => {loan_amount_check=0} 0.02369478  0.7759420 1.141438
## 97  {location.country=Pakistan}         => {loan_amount_check=0} 0.02081779  0.7743634 1.139116
## 98  {lender_count_quant=Q3,
##      activity=Farming}                  => {loan_amount_check=0} 0.01725524  0.7719578 1.135577
## 99  {activity=Food Production/Sales,
##      borrowers.gender=F}                => {loan_amount_check=0} 0.03300362  0.7692152 1.131542
```

```
## 100 {status=paid,
##      activity=Food Production/Sales}     => {loan_amount_check=0} 0.02978000  0.7675833 1.129142
## 101 {lender_count_quant=Q3}              => {loan_amount_check=0} 0.19215989  0.7675775 1.129133
## 102 {sector=Personal Use,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.01104582  0.7655145 1.126099
## 103 {activity=Food Market,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.01407013  0.7619668 1.120880
## 104 {activity=Fruits & Vegetables,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.01775592  0.7611441 1.119670
## 105 {sector=Personal Use}                => {loan_amount_check=0} 0.01799952  0.7603823 1.118549
## 106 {activity=Food Production/Sales}     => {loan_amount_check=0} 0.03553110  0.7562970 1.112539
## 107 {status=in_repayment,
##      sector=Retail}                      => {loan_amount_check=0} 0.01760476  0.7522422 1.106575
## 108 {location.country=El Salvador,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.02050294  0.7512436 1.105106
## 109 {sector=Food,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.15216020  0.7488769 1.101624
## 110 {sector=Agriculture,
##      location.country=Peru}              => {loan_amount_check=0} 0.01283672  0.7467234 1.098456
## 111 {location.country=Uganda}            => {loan_amount_check=1} 0.01256809  0.3506797 1.095167
## 112 {location.country=Peru,
##      borrowers.gender=M}                 => {loan_amount_check=0} 0.01286175  0.7433914 1.093555
## 113 {status=paid,
##      activity=Fruits & Vegetables}       => {loan_amount_check=0} 0.01655910  0.7432152 1.093296
## 114 {location.country=Nicaragua}         => {loan_amount_check=0} 0.02659393  0.7418350 1.091265
## 115 {status=paid,
##      location.country=Peru}              => {loan_amount_check=0} 0.04901197  0.7393534 1.087615
## 116 {sector=Retail,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.13147144  0.7375454 1.084955
## 117 {location.country=El Salvador}       => {loan_amount_check=0} 0.03388559  0.7369027 1.084010
## 118 {activity=Fruits & Vegetables}       => {loan_amount_check=0} 0.01993004  0.7336429 1.079214
## 119 {status=paid,
##      sector=Food}                        => {loan_amount_check=0} 0.14308534  0.7335525 1.079081
## 120 {activity=Farming,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.04230379  0.7316813 1.076329
## 121 {status=paid,
##      activity=Farming}                   => {loan_amount_check=0} 0.04621971  0.7294734 1.073081
## 122 {status=in_repayment,
##      sector=Food}                        => {loan_amount_check=0} 0.02102384  0.7291945 1.072671
## 123 {status=paid,
##      sector=Retail}                      => {loan_amount_check=0} 0.12630575  0.7244937 1.065755
## 124 {location.country=Peru}              => {loan_amount_check=0} 0.05230107  0.7234334 1.064196
## 125 {status=paid,
##      borrowers.gender=F}                 => {loan_amount_check=0} 0.42636458  0.7221144 1.062256
## 126 {sector=Food}                        => {loan_amount_check=0} 0.17113895  0.7216279 1.061540
## 127 {status=paid,
##      activity=Food Market}               => {loan_amount_check=0} 0.01323727  0.7201676 1.059392
## 128 {status=in_repayment,
##      activity=Farming}                   => {loan_amount_check=0} 0.01327385  0.7173110 1.055190
## 129 {location.country=Cambodia,
##      borrowers.gender=F}                 => {loan_amount_check=1} 0.01529680  0.3369173 1.052188
## 130 {status=paid,
##      sector=Transportation}              => {loan_amount_check=0} 0.01714162  0.7131184 1.049022
## 131 {activity=Food Market}               => {loan_amount_check=0} 0.01548456  0.7121601 1.047612
```

```
## 132 {sector=Retail}                       => {loan_amount_check=0} 0.15174233  0.7116884 1.046919
## 133 {borrowers.gender=F}                  => {loan_amount_check=0} 0.53068363  0.7107675 1.045564
## 134 {activity=Farming}                    => {loan_amount_check=0} 0.06176589  0.7078276 1.041239
## 135 {activity=Food Stall}                 => {loan_amount_check=0} 0.01092161  0.7068170 1.039753
## 136 {location.country=Cambodia}           => {loan_amount_check=1} 0.01780310  0.3317425 1.036027
## 137 {sector=Agriculture}                  => {loan_amount_check=1} 0.07861962  0.3314445 1.035096
## 138 {status=paid,
##      sector=Housing}                      => {loan_amount_check=0} 0.01770874  0.7002475 1.030088
## 139 {status=paid,
##      activity=Personal Housing Expenses} => {loan_amount_check=0} 0.01638097  0.6994039 1.028848
## 140 {status=paid}                         => {loan_amount_check=0} 0.54095436  0.6974336 1.025949
## 141 {sector=Transportation}               => {loan_amount_check=0} 0.02055204  0.6883042 1.012519
## 142 {status=in_repayment}                 => {loan_amount_check=0} 0.10901103  0.6846450 1.007137
## 143 {activity=Personal Housing Expenses} => {loan_amount_check=0} 0.02602873  0.6828239 1.004458
## 144 {sector=Housing}                      => {loan_amount_check=0} 0.02760299  0.6810148 1.001797
## 145 {}                                    => {loan_amount_check=1} 0.32020647  0.3202065 1.000000
## 146 {}                                    => {loan_amount_check=0} 0.67979353  0.6797935 1.000000
```

In the below rule, we analyze the number of lenders that are associated with higher loan amount in Bolivia.

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

```
##   lhs                            rhs                       support confidence     lift
## 1 {lender_count_quant=Q4,
##    location.country=Bolivia} => {loan_amount_check=1} 0.01137319  0.9809002 3.063337
```

The support value of 0.011% indicates that the percentage of the higher loans are associated with large number of lenders in Bolivia. It has a confidence of 96% and higher lift value 3.012.

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

```
##   lhs                          rhs                       support confidence    lift
## 1 {lender_count_quant=Q4,
##    sector=Services}       => {loan_amount_check=1} 0.02174598  0.9713561 3.03353
```

We observe that large number of lenders are interested in service sector for lending higher loan amounts. This has the fourth highest lift value of 2.58 and support value is 0.01 which reflects that among all sectors, lenders are interested in services sector and 97% are associated with higher loan amounts.

We observe an interesting rule that relates gender with loan amount. Males tend to borrow larger loan amount, however they take lesser percentage of loans when compared to females. And female borrowers tend to take smaller loans, however they take larger percentage of loans when compared to male.

```
inspect(rules.pruned[68])
```

```
##    lhs                      rhs                       support   confidence
## 72 {borrowers.gender=M} => {loan_amount_check=1} 0.1042555 0.4114828
##    lift
## 72 1.285055
```

Support value of 10% states that 10% of males across various countries and sectors are associated with higher loan amount, the confidence value shows that 41% of gender who are male are associated with higher loan amount. It has a lift value of 1.28 which indicates that it is 1.28 times more likely to occur with the data.

```
inspect(rules.pruned[133])
```

```
##    lhs                       rhs                     support   confidence
## 77 {borrowers.gender=F} => {loan_amount_check=0} 0.5306836 0.7107675
##    lift
## 77 1.045564
```

This rule has very high support value of 52% which implies that females across various countries and sectors are associated with smaller loan amount. The confidence value shows that 70% of females are associated with lesser loan amount. It has a lift value of 1.04 which indicates that it is 1.04 times more likely to occur in the data.

The below rule suggests that people taking loans less than the average loan amount are more likely to pay back the loan.

```
inspect(rules.pruned[140])
```

```
##    lhs              rhs                     support   confidence lift
## 78 {status=paid} => {loan_amount_check=0} 0.5409544 0.6974336  1.025949
```

This has higher support value of 53% and a confidence of 69% stating that of all the statuses that are paid, loan amount is of lesser value. A lift of 1.02 states that this scenario is 1.02 times more likely to happen with the data.

5.2 Relationship between status and other variables

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

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

```
avgloan_df <- loan.df[loan.df$loan_amount > mean(loan.df$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)
```

We convert all variables to factors to apply association rules.

```
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 low (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 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]
```

After sorting the rules based on higher lift value we observe that Togo has the highest female count that are associated with `defaulted` loan status.

```
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
```

They have a support value of 0.01 and confidence of 97% which determines the percentage of female genders who are associated with `defaulted` loan status. They have a lift value of 7.46

Similar to Togo, Pakistan also has high female gender borrowers which are associated with `defaulted` loan status. It ranks next to Togo based on higher lift value.

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

```
##   lhs                             rhs                  support confidence     lift
## 1 {location.country=Pakistan,
##    borrowers.gender=F}        => {status=defaulted} 0.01227465   0.639444 4.908052
```

They have a support value of 0.012 and confidence of 63% which determines the percentage of female genders who are associated with defaulted loan status. They have a lift value of 4.90.

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

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

We also observed that in agriculture loans which are borrowed by males have `default` status. It has a support value of 0.013 and a confidence of 15% which states the percentage of agriculture sector in association with `default` loan status. They have a lift value of 1.162.

Below result shows the percentage of male population associated with `default` status.

```
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 has a support of 0.04%. The confidence value shows that 14% of gender who are male are associated with defaulted loan status. A lift of 1.12 states that this scenario is 1.12 times more likely to happen.

From above results, it is clear that lenders should be more cautious while funding loans to countries like Togo, Pakistan and also for agriculture sector as they are more likely to end up with `default` loan status.

Countries such as Vietnam, Nigeria and india have higher lift values with repect to `in_repayment` status.

Below rule shows the female association with respect to `in_repayment` status.

```
inspect(rules_status.pruned[65])
```

```
##    lhs                       rhs                     support   confidence
## 56 {borrowers.gender=F} => {status=in_repayment} 0.6037424 0.8767923
##    lift
## 56 1.008137
```

This rule has very high support value of 60% stating that 60% of females across various countries and sectors have their loan status as `in_repayment`. The confidence value shows that 87% of gender who are female are associated with `in_repayment` loan status. It has a lift value of 1.008 which indicates that it is 1.008 times more likely to occur.

As female population are trying to pay back the loan, there are two possibilities that can occur, either they can completely pay back the loan or they can be in defaulted loan status. As we have already stated that majority of female populations are associated with lower value of loan amounts, there are less chances that female population can be in defaulted status when it comes to larger value of loan amounts, even though they are associated with certain defaulted status.

5.3 Relationship between GDP and other variables

We have already described about loan amount distribution across various sectors, genders and countries. Let us see how the GDP have impact on these factors. Here GDP per annual growth rate is used for the comparison. This indicator from WDI is already merged with Kiva and the resultant dataframe is available as final1df.

We group GDP values into four quantiles, Q1 is the GDP growth rate in between 0% and 25% of total rates, Q2 is the GDP growth rate in between 25% and 50% of total rates, Q3 is the GDP growth rate in between 50% and 75% of total rates and Q4 is the GDP growth rate in between 75% and 100% of total growth rates. The growth rate is compared with loan amount, gender and country. The loan amount is grouped into four values - low, medium, high and very high based on the value of amount.

```
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(location.country,loan_quant,borrowers.gender,GDP_quant)
```

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

```
finaldf_GDP["location.country"] <- as.factor(finaldf_GDP$location.country)
finaldf_GDP["sector"] <- as.factor(finaldf_GDP$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 oas 0.2 and maxlen as 2.

The different quantiles of GDP growth rate are set to rhs value other variables as lhs. 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]
```

From our previous statement, it is clear that males borrow higher loans when compared to females. Below rule shows relationship between loan amount and male with respect to GDP.

```
inspect(rules_GDP.pruned[43])
```

```
##   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, GDP growth rate is high. This rule has a support value of 0.026% mentioning that 0.026% of males who takes higher loans in any country have higher growth rate with respet to GDP. It has a confidence value of 31% and lift value of 1.28 which indicates that it is 1.28 times more likely to occur in the data.

Another interesting rule with respect to female is described below. We have already stated that females borrow lesser loan amount. Below rule shows this particular relation with respect to GDP.

```
inspect(rules_GDP.pruned[50])
```

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

From the above rule, we infer that when females tend to take lesser loans, GDP growth rate of the country seems to be very low(Q1). This has a support value of 0.06 mentioning that 0.06% of females who takes lesser loans in any country have lesser growth rate with respet to GDP. It has a confidence value of 29% and a lift value of 1.16.

6. Conclusion