DS5110 - Kiva Crowdfunding

Deep Bhalodia

April 21, 2019

Importing Required Packages

```
package_install_load <- function(x)
{
  if (!require(x, character.only = TRUE))
  {
    install.packages(x, dep = TRUE)
    if (!require(x, character.only = TRUE))
    {
      stop('Package not found')
    }
  }
}

## Block - Load Required Packages
packages <- c("rlang","tidyverse","ggplot2","gridExtra","dplyr","sqldf","readxl","readr","tidytext","ti
              ,"maps","lubridate","treemap","stringi","stringr","plyr","leaflet")
invisible(lapply(packages, package_install_load))
```

Function - Set Working Directory

```
set_workspace <- function(dir_path)
{

  #dir.create(dir_path) # Create Directory
  setwd(dir_path) #Set Working Directory
  print(paste("Working Directory Set to : ",dir_path)) # Print Message
}

## Setting Workspace
set_workspace("/Users/deep/DMDP/KIVA");
```

```
## [1] "Working Directory Set to :  /Users/deep/DMDP/KIVA"
```

(I) Data Acquisition:

(a) Acquisition of Data:

```
loan <- read_csv("kiva_loans.csv")


reg_loc <- read_csv("kiva_mpi_region_locations.csv")


theme <- read_csv("loan_theme_ids.csv")
```

```
theme_reg <- read_csv("loan_themes_by_region.csv")


flood_dataset <- read_excel("Flood_Risk.xlsx")


data(world.cities)
world_cities <- world.cities

remove(world.cities)
```

(b) Glimpse of Data:

```
head(loan)
```

```
## # A tibble: 6 x 20
##       id funded_amount loan_amount activity sector use   country_code
##    <dbl>         <dbl>       <dbl> <chr>    <chr>  <chr> <chr>
## 1 6.53e5           300         300 Fruits ~ Food   To b~ PK
## 2 6.53e5           575         575 Rickshaw Trans~ to r~ PK
## 3 6.53e5           150         150 Transpo~ Trans~ To r~ IN
## 4 6.53e5           200         200 Embroid~ Arts   to p~ PK
## 5 6.53e5           400         400 Milk Sa~ Food   to p~ PK
## 6 1.08e6           250         250 Services Servi~ purc~ KE
## # ... with 13 more variables: country <chr>, region <chr>, currency <chr>,
## #   partner_id <dbl>, posted_time <dttm>, disbursed_time <dttm>,
## #   funded_time <dttm>, term_in_months <dbl>, lender_count <dbl>,
## #   tags <chr>, borrower_genders <chr>, repayment_interval <chr>,
## #   date <date>
```

```
glimpse(loan)
```

```
## Observations: 671,205
## Variables: 20
## $ id                 <dbl> 653051, 653053, 653068, 653063, 653084, 108...
## $ funded_amount      <dbl> 300, 575, 150, 200, 400, 250, 200, 400, 475...
## $ loan_amount        <dbl> 300, 575, 150, 200, 400, 250, 200, 400, 475...
## $ activity           <chr> "Fruits & Vegetables", "Rickshaw", "Transpo...
## $ sector             <chr> "Food", "Transportation", "Transportation",...
## $ use                <chr> "To buy seasonal, fresh fruits to sell.", "...
## $ country_code       <chr> "PK", "PK", "IN", "PK", "PK", "KE", "IN", "...
## $ country            <chr> "Pakistan", "Pakistan", "India", "Pakistan"...
## $ region             <chr> "Lahore", "Lahore", "Maynaguri", "Lahore", ...
## $ currency           <chr> "PKR", "PKR", "INR", "PKR", "PKR", "KES", "...
## $ partner_id         <dbl> 247, 247, 334, 247, 245, NA, 334, 245, 245,...
## $ posted_time        <dttm> 2014-01-01 06:12:39, 2014-01-01 06:51:08, ...
## $ disbursed_time     <dttm> 2013-12-17 08:00:00, 2013-12-17 08:00:00, ...
## $ funded_time        <dttm> 2014-01-02 10:06:32, 2014-01-02 09:17:23, ...
## $ term_in_months     <dbl> 12, 11, 43, 11, 14, 4, 43, 14, 14, 11, 11, ...
## $ lender_count       <dbl> 12, 14, 6, 8, 16, 6, 8, 8, 19, 24, 3, 16, 1...
## $ tags               <chr> NA, NA, "user_favorite, user_favorite", NA,...
## $ borrower_genders   <chr> "female", "female, female", "female", "fema...
## $ repayment_interval <chr> "irregular", "irregular", "bullet", "irregu...
```

```
## $ date                <date> 2014-01-01, 2014-01-01, 2014-01-01, 2014-0...
```

```
head(reg_loc)
```

```
## # A tibble: 6 x 9
##   LocationName  ISO   country  region world_region   MPI geo      lat   lon
##   <chr>         <chr> <chr>    <chr>  <chr>        <dbl> <chr>  <dbl> <dbl>
## 1 Badakhshan, ~ AFG   Afghani~ Badak~ South Asia   0.387 (36.7~  36.7  70.8
## 2 Badghis, Afg~ AFG   Afghani~ Badgh~ South Asia   0.466 (35.1~  35.2  63.8
## 3 Baghlan, Afg~ AFG   Afghani~ Baghl~ South Asia   0.3   (35.8~  35.8  69.3
## 4 Balkh, Afgha~ AFG   Afghani~ Balkh  South Asia   0.301 (36.7~  36.8  66.9
## 5 Bamyan, Afgh~ AFG   Afghani~ Bamyan South Asia   0.325 (34.8~  34.8  67.8
## 6 Daykundi, Af~ AFG   Afghani~ Dayku~ South Asia   0.313 (33.6~  33.7  66.0
```

```
glimpse(reg_loc)
```

```
## Observations: 2,772
## Variables: 9
## $ LocationName <chr> "Badakhshan, Afghanistan", "Badghis, Afghanistan"...
## $ ISO          <chr> "AFG", "AFG", "AFG", "AFG", "AFG", "AFG", "AFG", ...
## $ country      <chr> "Afghanistan", "Afghanistan", "Afghanistan", "Afg...
## $ region       <chr> "Badakhshan", "Badghis", "Baghlan", "Balkh", "Bam...
## $ world_region <chr> "South Asia", "South Asia", "South Asia", "South ...
## $ MPI          <dbl> 0.387, 0.466, 0.300, 0.301, 0.325, 0.313, 0.319, ...
## $ geo          <chr> "(36.7347725, 70.81199529999999)", "(35.1671339, ...
## $ lat          <dbl> 36.73477, 35.16713, 35.80429, 36.75506, 34.81001,...
## $ lon          <dbl> 70.81200, 63.76954, 69.28775, 66.89754, 67.82121,...
```

Summary Statistics Data

```
#Lets identify the total funded amount by Kiva to the field agents
total_funded_amnt <- sum(loan$funded_amount)
total_funded_amnt
```

```
## [1] 527563815
```

```
#Lets identify the average and median amount of total funded amount by Kiva to borroweres
summary(loan$funded_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     250     450     786     900  100000
```

Data Preparation

Data Cleaning & Shaping

(I) Loan Time Formats

```
#separate funded time into years only, group by years and indentify the analysis
loan <- loan %>% mutate(funding_year = year(funded_time))
```

(II) Removal and Alteration of Columns

```r
#Unique observation where kiva has active loans
reg_loc <- unique(reg_loc[, !(colnames(reg_loc) %in% c("geo"))])
loan <- unique(loan[, !(colnames(loan) %in% c("date"))])
theme <- unique(theme[, !(colnames(theme) %in% c("id"))])
```

(iii) Modifying "borrower_genders" variable to replace every instance with single gender i.e male or female

```r
loan <- loan %>% mutate(gender = ifelse(str_detect(borrower_genders, "female"), "female", "male"))
```

Data Exploration:

Exploratory Data Plots: General

(i) Loan

```r
#levels of sector
table(loan$sector)
```

```
##
##    Agriculture           Arts       Clothing   Construction      Education
##         180302          12060          32742           6268          31013
##  Entertainment           Food         Health        Housing  Manufacturing
##            830         136657           9223          33731           6208
##   Personal Use         Retail       Services Transportation      Wholesale
##          36385         124494          45140          15518            634
```

```r
#levels of repayment interval
table(loan$repayment_interval)
```

```
##
##    bullet irregular   monthly    weekly
##     70728    257158    342717       602
```

```r
#levels of gender
table(loan$gender)
```

```
##
## female   male
## 528461 138523
```

```r
#summary statsistics of funded amount by kiva
summary(loan$funded_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     250     450     786     900  100000
```

(ii) MPI_REGION_LOC

```r
#identifying which columns with observations across the table is not having missing values
colSums(!is.na(reg_loc))
```

```
## LocationName          ISO      country       region world_region
##          984         1008         1008          984         1008
##          MPI          lat          lon
##          984          892          892
```

```
#identify which columns with observations across the table has missing values
colSums(is.na(reg_loc))
```

```
## LocationName          ISO      country       region world_region
##           25            1            1           25            1
##          MPI          lat          lon
##           25          117          117
```

```
#Lets remove the records where MPI is missing, since there will be no meaning of the other variables
reg_loc <- reg_loc %>%
  filter(!is.na(MPI))


theme_reg$country <- ifelse(theme_reg$country == 'Viet Nam','Vietnam',theme_reg$country)
reg_loc$country <- ifelse(reg_loc$country == 'Viet Nam','Vietnam',reg_loc$country)


theme_reg$country <- ifelse(theme_reg$country == 'Myanmar (Burma)','Myanmar',theme_reg$country)
reg_loc$country <- ifelse(reg_loc$country == 'Myanmar (Burma)','Myanmar',reg_loc$country)
```

Join of loan and reg_loc

####Benefit, we can map active loans data with world region and measurement poverty index. Alsom will be able to map latitude and longitude of the location.

```
#joining laon and reg_loc dataset by country
loan_reg_loc <- loan %>%
  left_join(reg_loc,by = c("country"="country"))
```

Join of theme and theme_reg

####Benefit, we can map active loans data with world region and measurement poverty index. Alsom will be able to map latitude and longitude of the location.

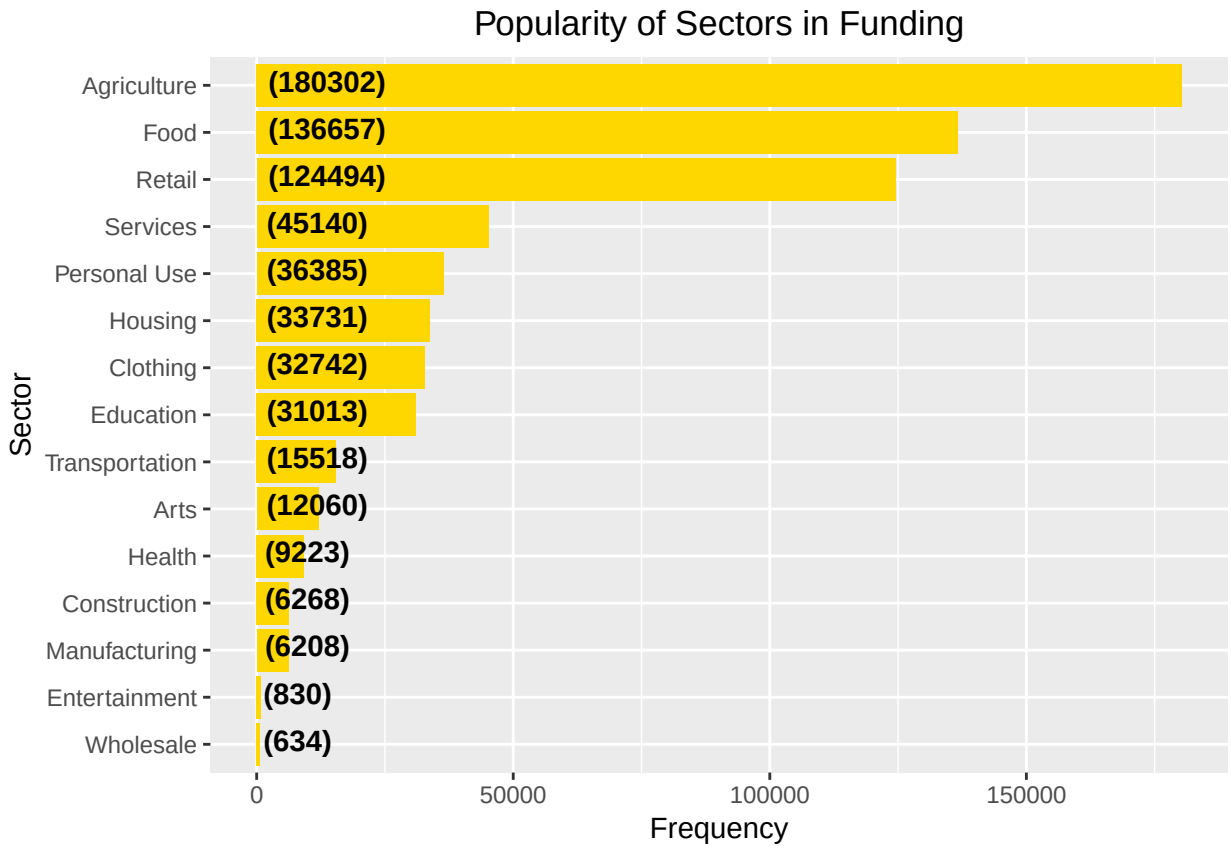
```
#natural join by country and region
theme_reg_join <- theme_reg %>%
  left_join(theme,by = c("Partner ID" = "Partner ID"))
```

ggplots - General

(i) Popularity of sector where large number of customers got funded

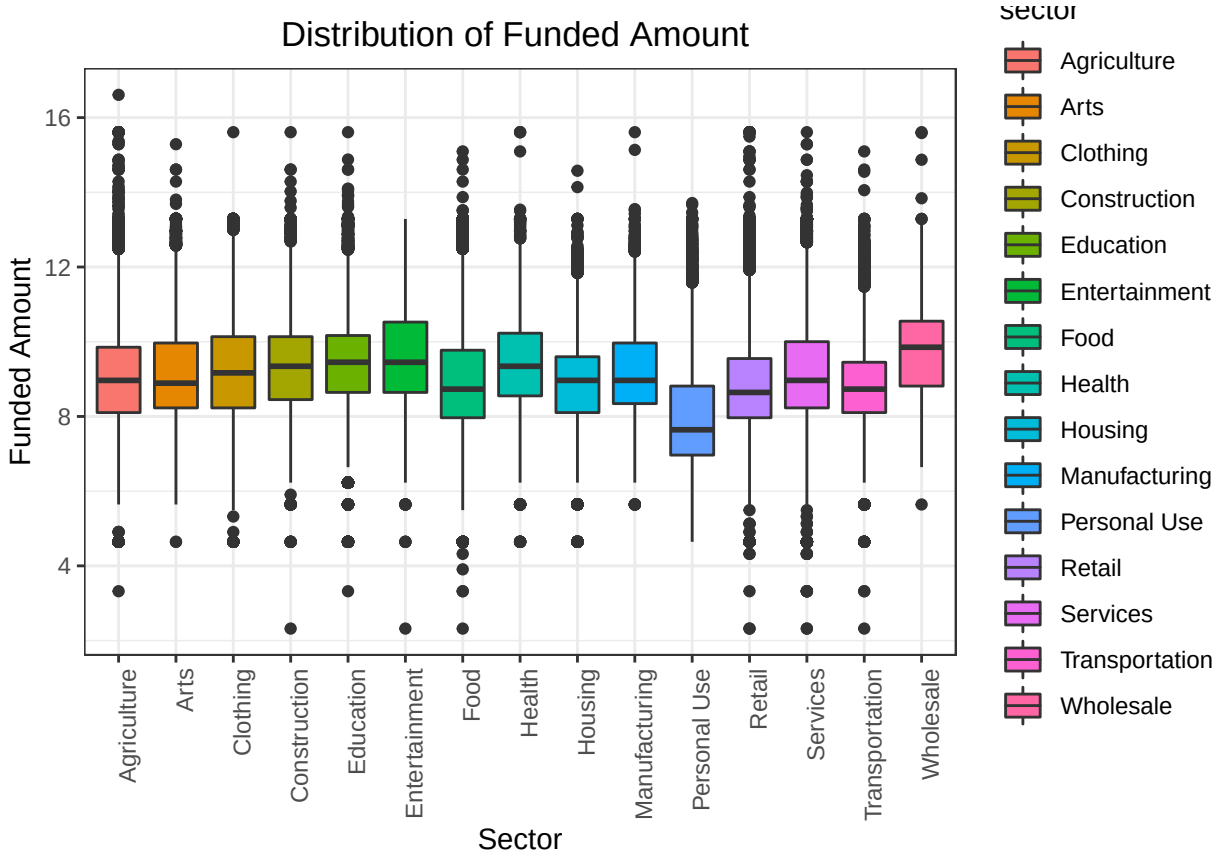
```
loan %>%
  group_by(sector) %>%
  dplyr::summarise(Count = n()) %>%
  arrange(desc(Count)) %>%
  mutate(sector = reorder(sector,Count)) %>%
  ggplot(aes(x = sector,y = Count)) +
  geom_bar(stat = "identity", fill = "Gold") +
  geom_text(aes(x = sector, y = 2, label = paste0("(",Count,")",sep=""))
            , hjust = -.1, vjust = .3, fontface = "bold") +
  xlab("Sector") +
  ylab("Frequency") +
```

```
  ggtitle("Popularity of Sectors in Funding") +
  coord_flip()+
  theme(plot.title = element_text(hjust = 0.5))
```



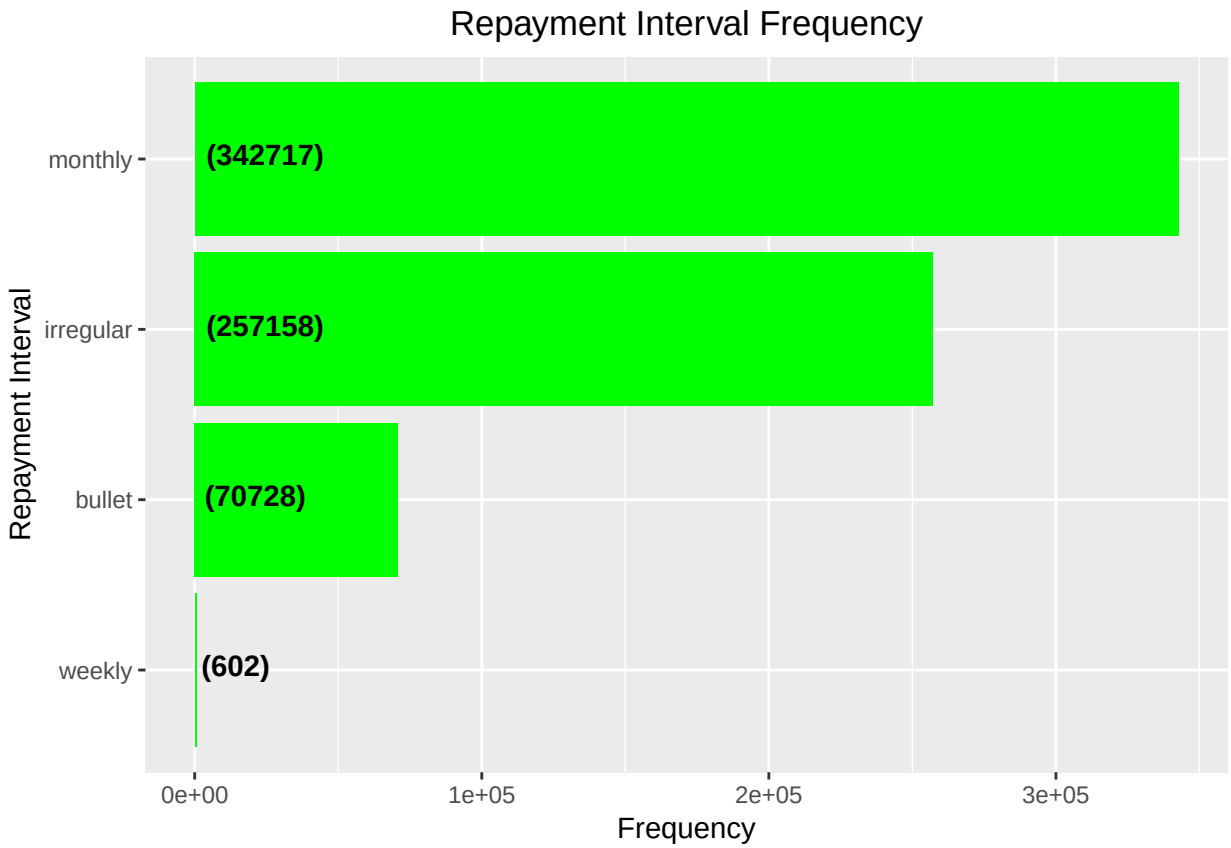
(ii) Popularity of sector where large number of funds been given

```
#Box-Plot, since using categorical against continuous

loan %>%
      mutate(fill = as.factor(sector))%>%
      ggplot(aes(x = sector, y= log2(funded_amount), fill = sector)) +
      geom_boxplot() +
      labs(x= 'Sector',y = 'Funded Amount',
           title = paste0("Distribution of", ' Funded Amount ')) +
       theme_bw() + theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  theme(plot.title = element_text(hjust = 0.5))
```



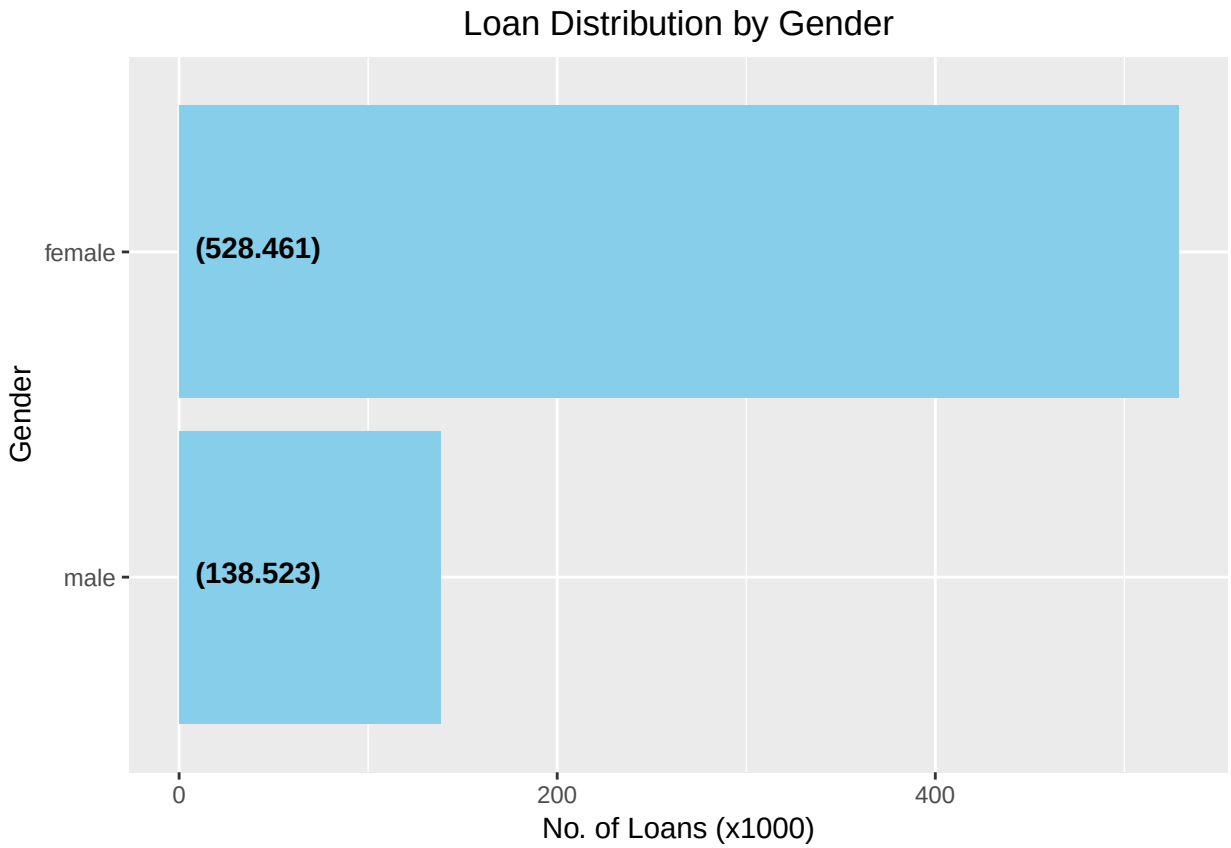
(iii) Identifying the most number of repayment interval

```
loan %>%
  group_by(repayment_interval) %>%
  dplyr::summarise(count = n()) %>%
  arrange(desc(count)) %>%
  mutate(repayment_interval = reorder(repayment_interval,count)) %>%
  ggplot(aes(x = repayment_interval,y = count)) +
  geom_bar(position = position_dodge(), stat = "identity", fill = "green") +
  geom_text(aes(x = repayment_interval, y = 2, label = paste0("(",count,")",sep="")), hjust = -.1, vjus
  xlab("Repayment Interval") +
  ylab("Frequency") +
  ggtitle("Repayment Interval Frequency") +
  coord_flip()+
  theme(plot.title = element_text(hjust = 0.5))
```



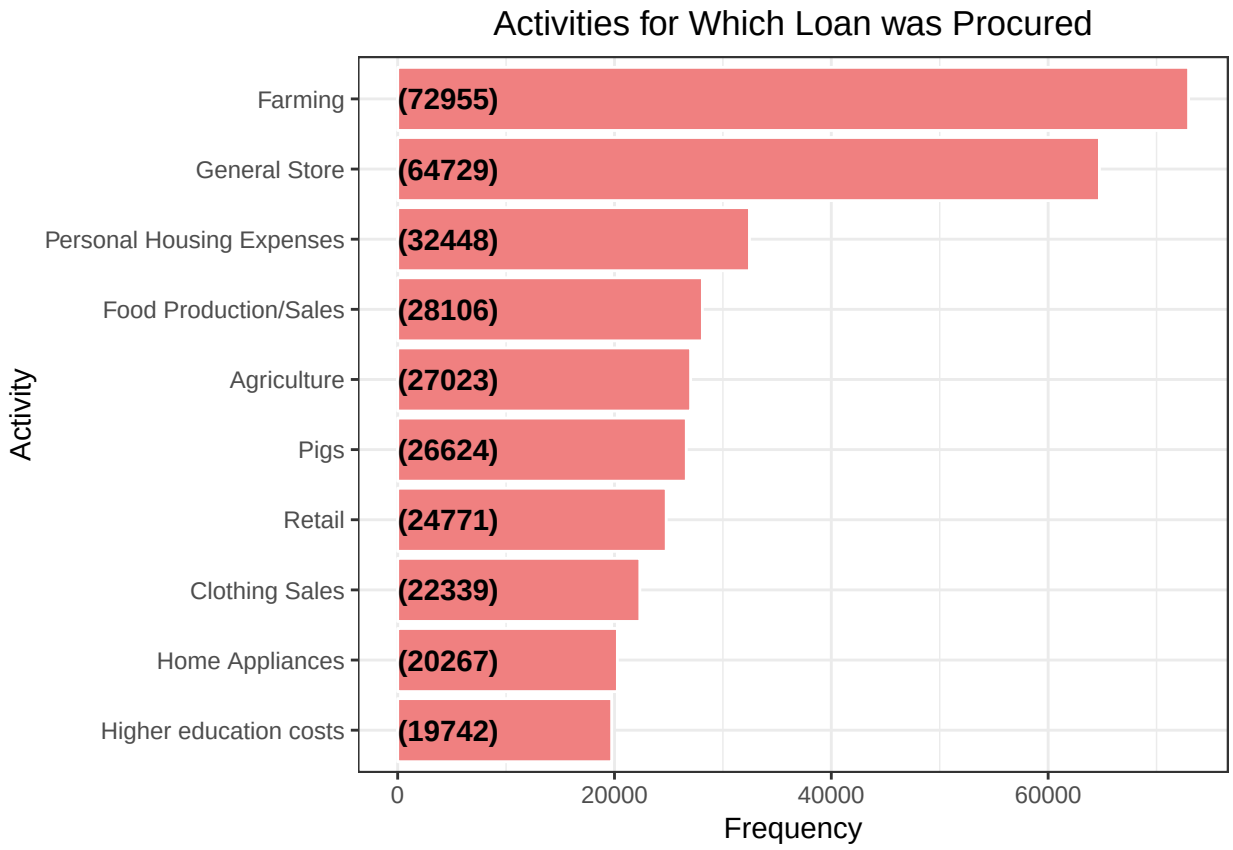
(iv) Identifying the gender to whom the loans been given

```
loan %>%
  filter(!is.na(gender)) %>%
  group_by(gender) %>%
  dplyr::summarise(count = n()) %>%
  arrange(desc(count)) %>%
  mutate(gender = reorder(gender,count)) %>%
  ggplot(aes(x = gender,y = count/1000)) +
  geom_bar(position = position_dodge(), stat = "identity", fill = "skyblue") +
  geom_text(aes(x = gender, y = 2, label = paste0("(",count/1000,")",sep="")), hjust = -.1, vjust = .3,
  xlab("Gender") +
  ylab("No. of Loans (x1000)") +
  ggtitle("Loan Distribution by Gender") +
  coord_flip()+
  theme(plot.title = element_text(hjust = 0.5))
```



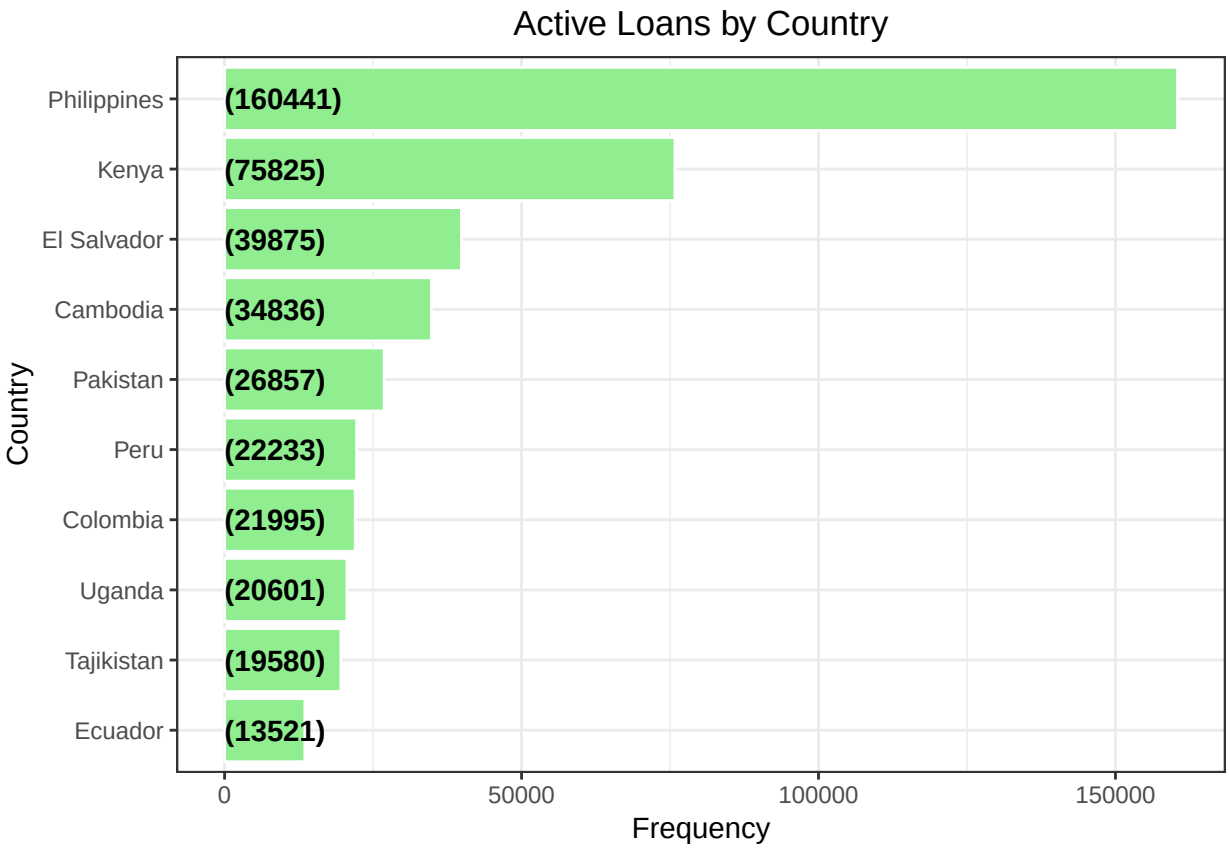
(v) Popularity of the activity where kiva has active loans

```
loan %>%
  group_by(activity) %>%
  dplyr::summarise(Count = n()) %>%
  arrange(desc(Count)) %>%
  ungroup() %>%
  mutate(activity = reorder(activity,Count)) %>%
  top_n(10) %>%

  ggplot(aes(x = activity,y = Count)) +
  geom_bar(stat='identity',colour="white", fill = "light coral") +
  geom_text(aes(x = activity, y = 1, label = paste0("(",Count,")",sep="")),
            hjust=0, vjust=.5, size = 4, colour = 'black',
            fontface = 'bold') +
  labs(x = 'Activity',
       y = 'Frequency',
       title = 'Activities for Which Loan was Procured') +
  coord_flip() +
   theme_bw()+
  theme(plot.title = element_text(hjust = 0.5))
```



(vi) Popularity by Country where KIVA has active loans

```
loan %>%
  group_by(country) %>%
  dplyr::summarise(Count = n()) %>%
  arrange(desc(Count)) %>%
  ungroup() %>%
  mutate(country = reorder(country,Count)) %>%
  head(10) %>%

  ggplot(aes(x = country,y = Count)) +
  geom_bar(stat='identity',colour="white", fill = "light green") +
  geom_text(aes(x = country, y = 1, label = paste0("(",Count,")",sep="")),
            hjust=0, vjust=.5, size = 4, colour = 'black',
            fontface = 'bold') +
  labs(x = 'Country',
       y = 'Frequency',
       title = 'Active Loans by Country') +
  coord_flip() +
   theme_bw()+
  theme(plot.title = element_text(hjust = 0.5))
```



The following plot shows the most popular themes in a bar chart. We have removed rows where the theme was not mentioned.

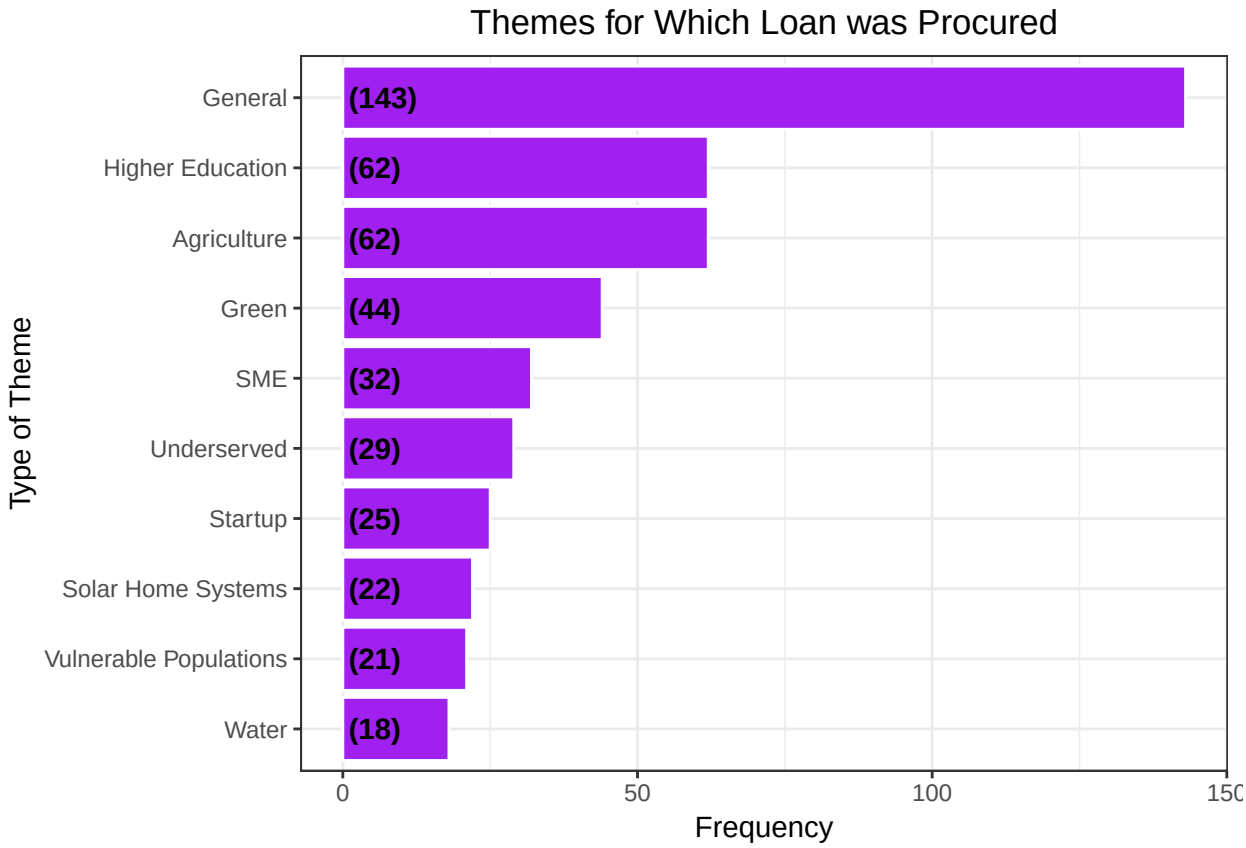
General is the most popular theme which does not give us a lot of information.

Underserved is the next popular theme, followed by Agriculture, Rural Inclusion, Water and Higher Education

(vii) Popularity of Loan Themes

```
theme %>%
  dplyr::rename (themeType = `Loan Theme Type`) %>%
  filter(!is.na(themeType)) %>%
  group_by(themeType) %>%
  dplyr::summarise(Count = n()) %>%
  arrange(desc(Count)) %>%
  ungroup() %>%
  mutate(themeType = reorder(themeType,Count)) %>%
  head(10) %>%

  ggplot(aes(x = themeType,y = Count)) +
  geom_bar(stat='identity',colour="white", fill = "purple") +
  geom_text(aes(x = themeType, y = 1, label = paste0("(",Count,")",sep="")),
            hjust=0, vjust=.5, size = 4, colour = 'black',
```

```
          fontface = 'bold') +
  labs(x = 'Type of Theme',
       y = 'Frequency',
       title = 'Themes for Which Loan was Procured') +
  coord_flip() +
   theme_bw()+
  theme(plot.title = element_text(hjust = 0.5))
```



Centralizing the daatframe as appropriate for the Project

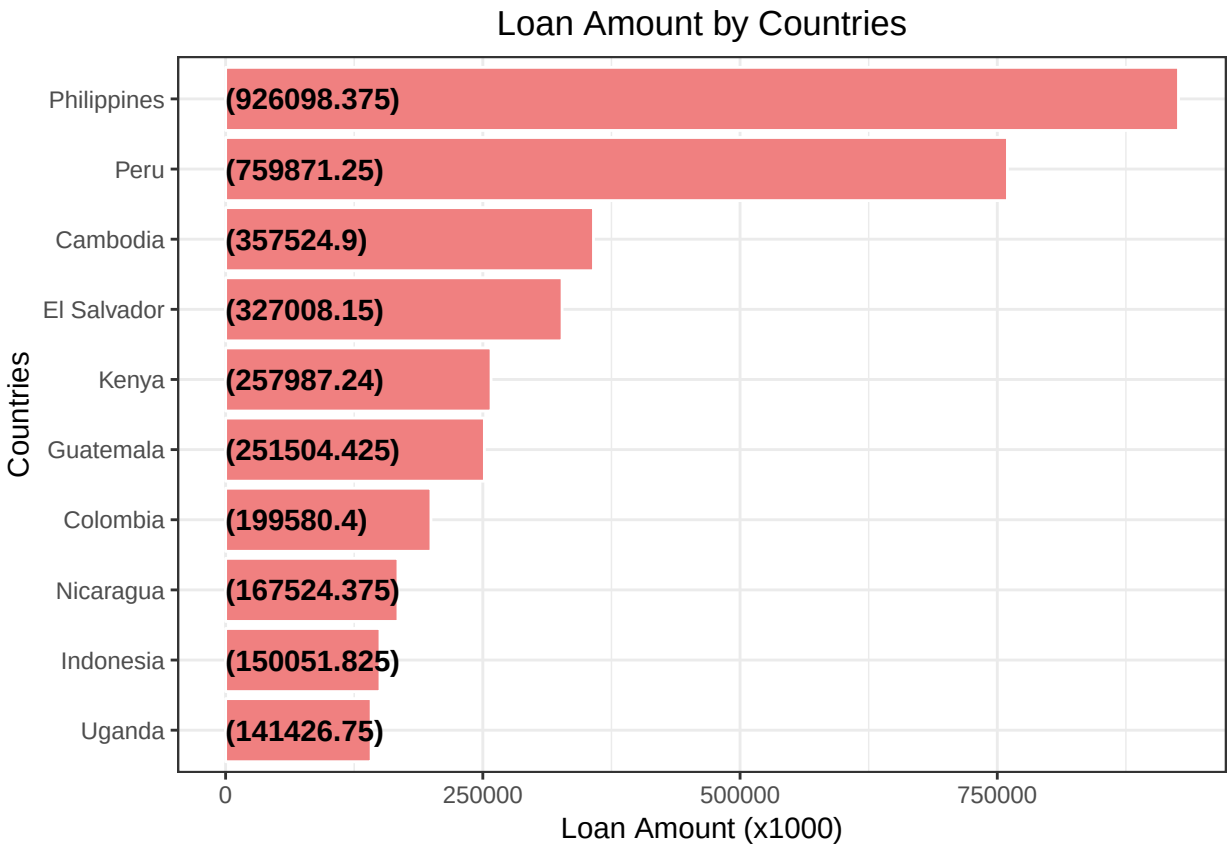
```
colnames(world_cities) <- gsub("\\.", "_", colnames(world_cities))

world_cities_ctry <- sqldf("select country_etc as country,median(lat) as lat, median(long) as lon
                           from world_cities
                           group by country_etc")


country_loans <- sqldf("select wc.country,sum(funded_amount) amount,wc.lat,wc.lon,avg(MPI) MPI
                       from loan cl
                       inner join world_cities_ctry wc on wc.country = cl.country
                       left join reg_loc rl on rl.country = wc.country
                        group by wc.country,wc.lat,wc.lon")
```

The below function generates a map of the region assigned and plots the loans disbursed

```
plotmap_by_country <- function(country_loans)
{

  center_lon = median(country_loans$lon,na.rm = TRUE)
  center_lat = median(country_loans$lat,na.rm = TRUE)


  leaflet(country_loans) %>% addTiles() %>%
    addCircles(lng = ~lon, lat = ~lat,radius = ~(amount/1000) ,
               color ="Blue")  %>%
    # controls
    setView(lng=center_lon, lat=center_lat,zoom = 2)
}
```

The function below creates plots for indicating countries top 10 countries in a region where maximum loan has been disbursed

```
  country_loans %>%
    group_by(country) %>%
    dplyr::summarise(tot_amt = sum(amount)) %>%
    arrange(desc(tot_amt)) %>%
    ungroup() %>%
    mutate(country = reorder(country,tot_amt)) %>%
    head(10) %>%

    ggplot(aes(x = country,y = tot_amt/1000)) +
    geom_bar(stat='identity',colour="white", fill = "Light coral") +
    geom_text(aes(x = country, y = 1, label = paste0("(",tot_amt/1000,")",sep="")),
              hjust=0, vjust=.5, size = 4, colour = 'black',
              fontface = 'bold') +
    labs(x = 'Countries',
         y = 'Loan Amount (x1000)',
         title = 'Loan Amount by Countries') +
    coord_flip() +
    theme_bw()+
    theme(plot.title = element_text(hjust = 0.5))
```

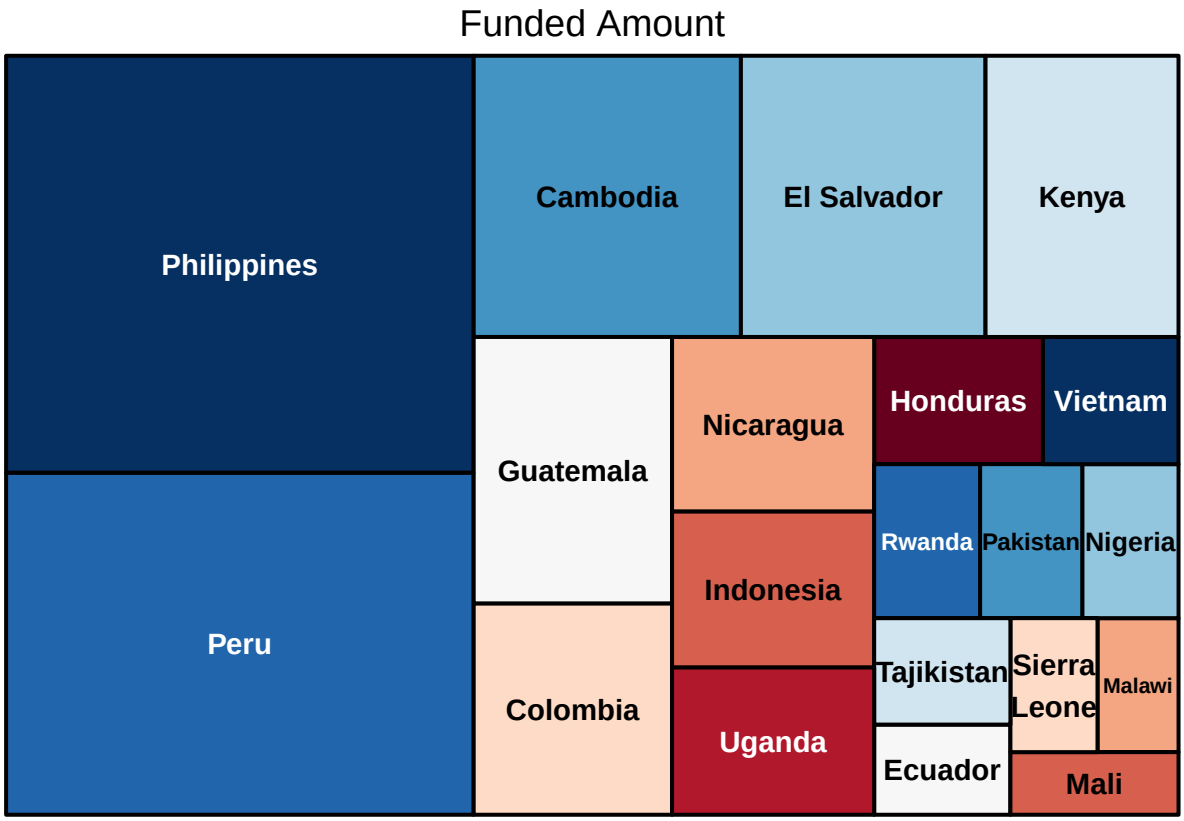


The Map below shows the areas of Africa were KIVA loans have been disbursed

```
#plotmap_by_country(country_loans)
```

Tree Map of the funded loan amount by Country

```
loans_funded_amount = country_loans %>%
  group_by(country) %>%
  dplyr::summarise(tot_amt = sum(amount)) %>%
  arrange(desc(tot_amt)) %>%
  ungroup() %>%
  mutate(country = reorder(country,tot_amt)) %>%
  head(20)


treemap(loans_funded_amount,
        index="country",
        vSize = "tot_amt",
        title="Funded Amount",
        palette = "RdBu",
        fontsize.title = 14
)
```



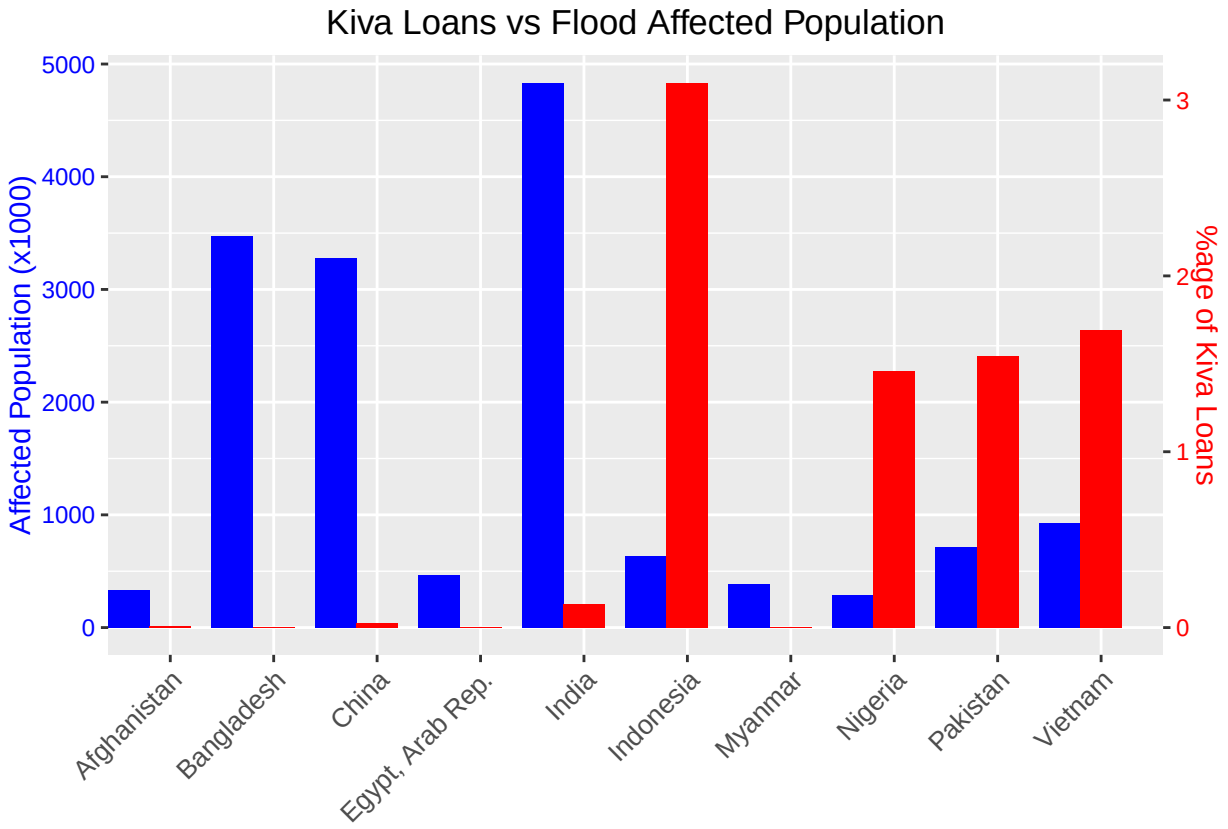
Flood Calamities Analysis

```
flood_dataset <- read_excel("Flood_Risk.xlsx")
```

```
## New names:
## * `` -> `..2`
## * `` -> `..3`
```

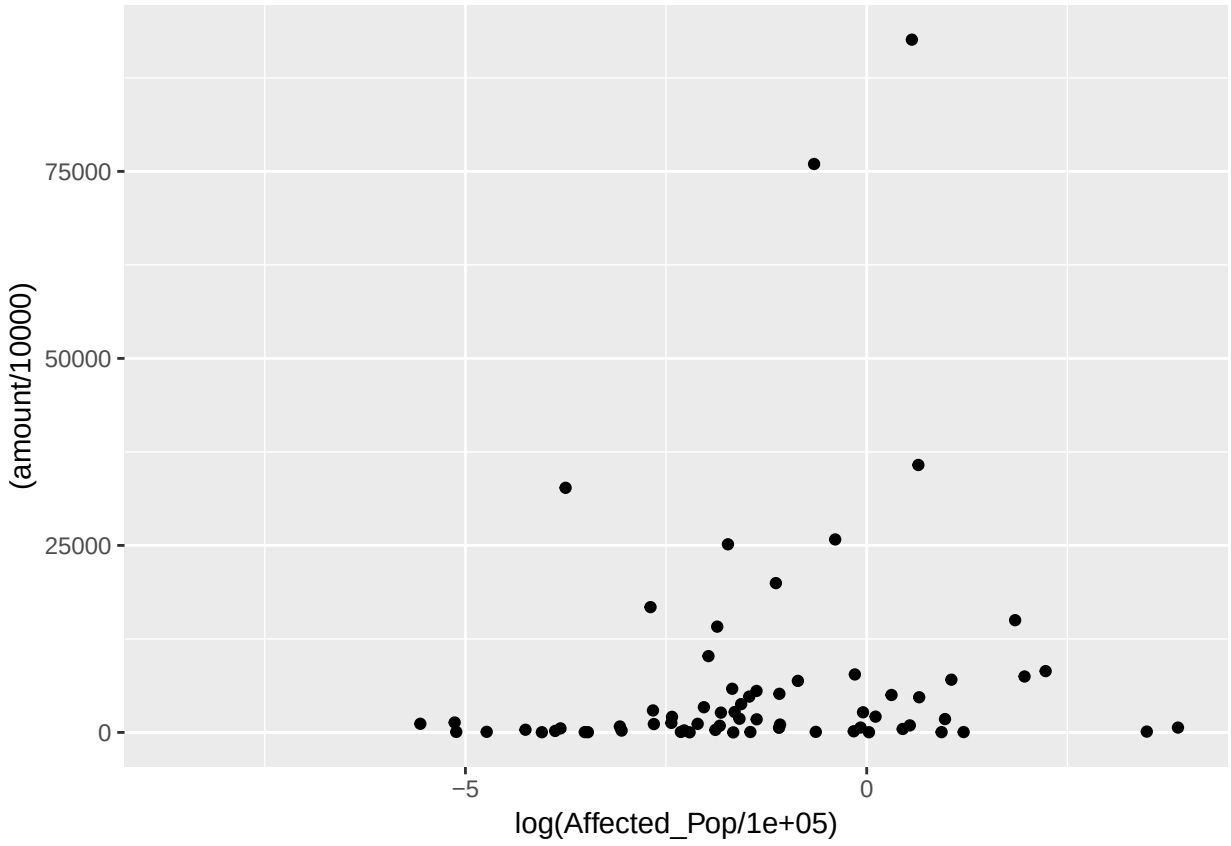
```
flood_dataset <- flood_dataset[6:nrow(flood_dataset),]
colnames(flood_dataset) <- flood_dataset[1,]
flood_dataset <- flood_dataset[-1,]
colnames(flood_dataset)[3] <- "Affected_Pop"


flood_dataset$Affected_Pop <- as.integer(flood_dataset$Affected_Pop)


flood_risks <- flood_dataset %>%
  left_join(country_loans, by=c("Country"="country")) %>%
  mutate(tot_amt = sum(country_loans$amount)) %>%
  mutate(Percentage_Prop = (amount/tot_amt)*100) %>%
  select("Country","Affected_Pop","amount","MPI","Percentage_Prop")


top_10_flood_risks_funding <- flood_risks %>%
  arrange(desc(Affected_Pop)) %>%
  head(10)
```

```r
top_10_flood_risks_funding$Percentage_Prop <- ifelse(top_10_flood_risks_funding$Percentage_Prop > 0 ,to
top_10_flood_risks_funding$Percentage_Prop[is.na(top_10_flood_risks_funding$Percentage_Prop)] <- 0



scaleFactor <- max(top_10_flood_risks_funding$Affected_Pop/1000) / max(top_10_flood_risks_funding$Percer


ggplot(top_10_flood_risks_funding, aes(x=Country,  width=.4)) +
  geom_col(aes(y=Affected_Pop/1000), fill="blue", position = position_nudge(x = -.4)) +
  geom_col(aes(y=Percentage_Prop * scaleFactor), fill="red") +
  scale_y_continuous(name="Affected Population (x1000)", sec.axis=sec_axis(~./scaleFactor, name="%age o
  theme(
    axis.title.y.left=element_text(color="blue"),
    axis.text.y.left=element_text(color="blue"),
    axis.title.y.right=element_text(color="red"),
    axis.text.y.right=element_text(color="red")
  ) +
  labs(title = "Kiva Loans vs Flood Affected Population", x = element_blank())+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(axis.title.y = element_text(vjust = 0.6)) +
  theme(axis.text.x =
                  element_text(size  = 10,
                               angle = 45,
                               hjust = 1,
                               vjust = 1))
```



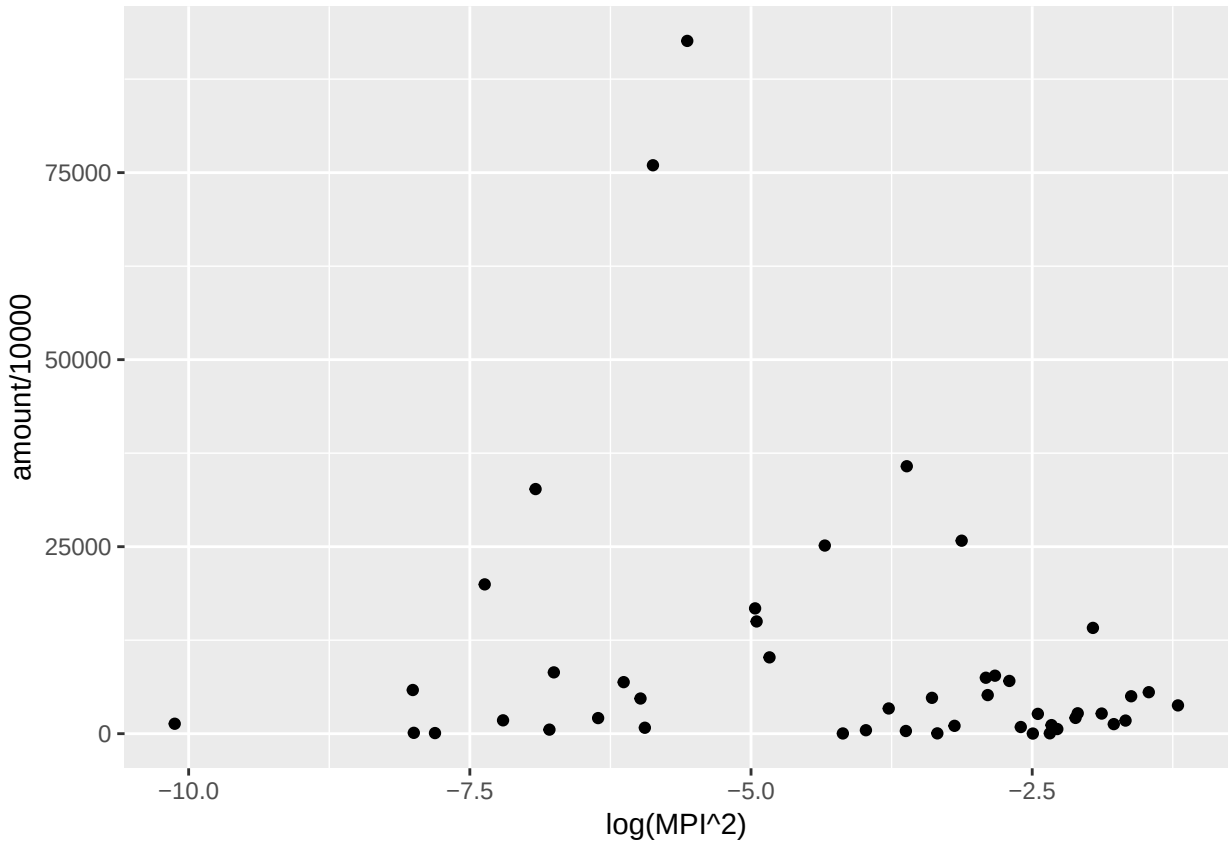
```
flood_risks %>%
  ggplot(aes(x=log(Affected_Pop/100000))) +
  geom_point(aes(y=(amount/10000)))
```

```
## Warning: Removed 96 rows containing missing values (geom_point).
```




```
flood_risks %>%
  ggplot(aes(x=log(MPI^2))) +
  geom_point(aes(y=amount/10000))
```

```
## Warning: Removed 116 rows containing missing values (geom_point).
```


amount/10000
75000
50000
25000
0
−10.0
−7.5
−5.0
−2.5
log(MPI^2)