**LENDING CLUB DATA ANALYSIS**

**(Assignment-2 Report)**

**Advance in Data Sciences and Architecture**

**INFO 7390 - SPRING 2017**

PROFESSOR:

**SRIKANTH KRISHNAMURTHY**

TEAM MEMBERS (Team 9):

**AASHRI TANDON**

**PRAGATI SHAW**

**SARTHAK AGARWAL**

Table of Contents

[Data Download 3](#_Toc479364345)

[Data Preprocessing 5](#_Toc479364346)

[Loan Data Files 5](#_Toc479364347)

[Declined Loan Data Files 7](#_Toc479364348)

[Luigi 8](#_Toc479364349)

[Dockerizing and Scheduling 9](#_Toc479364350)

[Exploratory data Analysis 11](#_Toc479364351)

[Loan data 11](#_Toc479364352)

[Declined Loan Data 15](#_Toc479364353)

[Jupyter Notebook 18](#_Toc479364354)

[Contributions 19](#_Toc479364355)

[References 19](#_Toc479364357)

# **Data Download**

We need to scrape the data from the following website: <https://www.lendingclub.com/info/download-data.action>

We must download loan status data and declined loan data for all the values in the dropdown. The download URL changes through AJAX on every update of dropdown value.

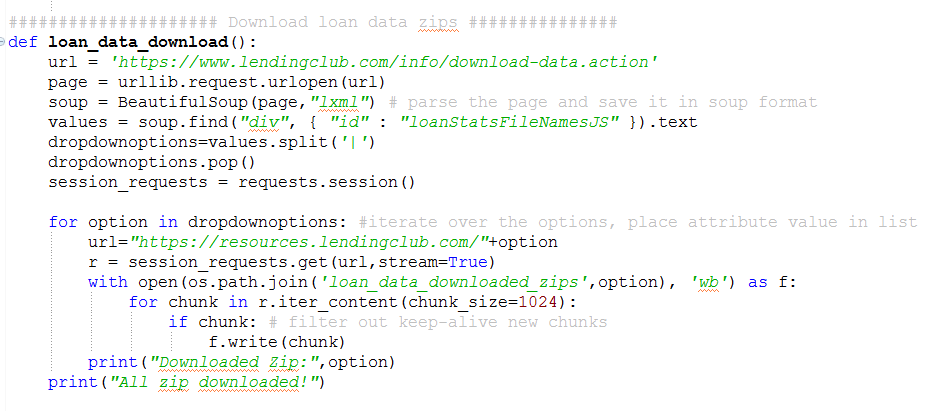
**Our approach to download**:

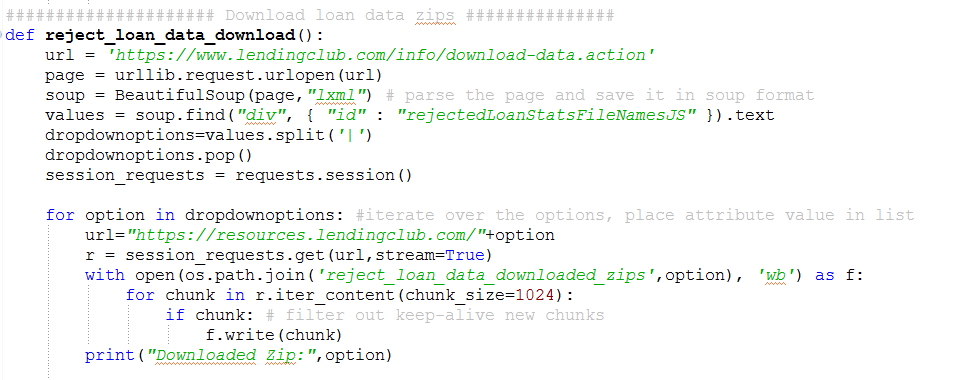
**Approach 1**: Using Selenium Webdriver

We had used selenium package in python, that opens the url, finds the drop down, changes the drop down and clicks on Download button. It works well on local machine. But installing ChromeDriver.exe or FirefoxDriver.exe on docker image caused problems.

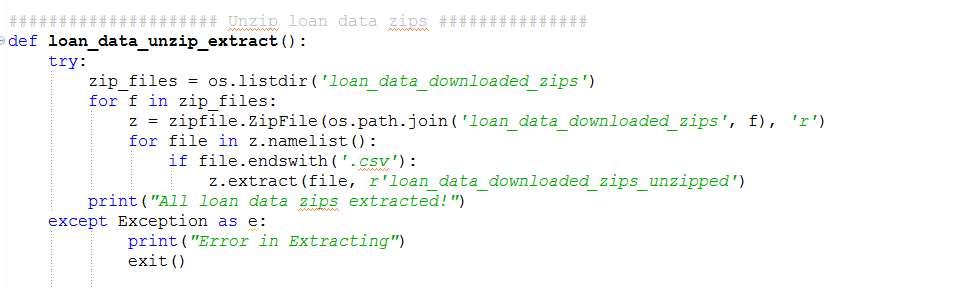
**Approach 2**: Using hidden element (Used)

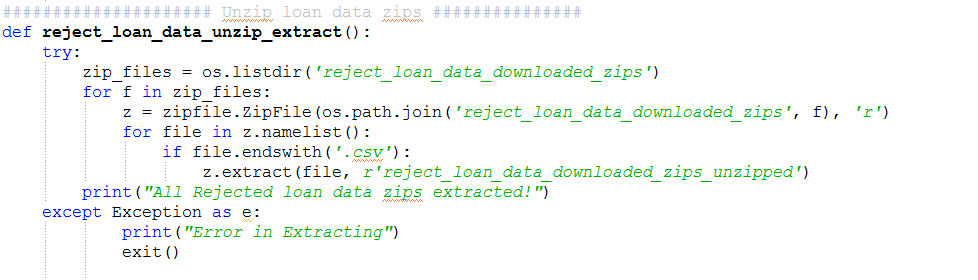
On inspecting the lending club web page, we discovered out that there are two hidden elements “loanStatsFileNamesJS” and “rejectedLoanStatsFileNamesJS”, that consists of all the values of filenames to be downloaded. Using session.request.get we have download the zip file.





Once the zip files are downloaded, we have unzipped the files in their respective folders.

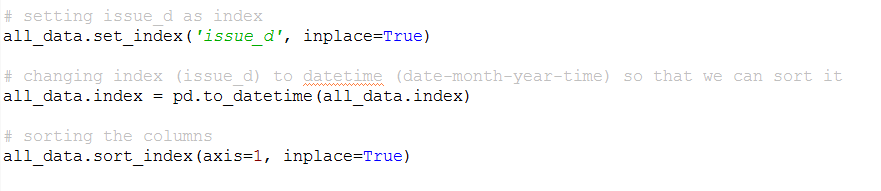




# **Data Preprocessing**

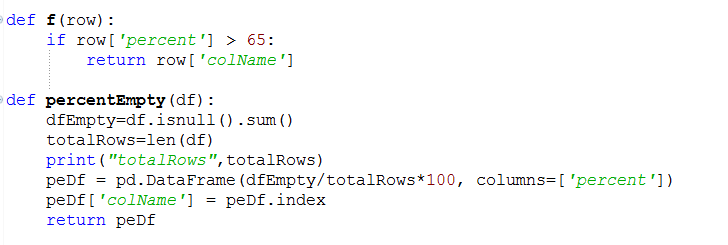
## **Loan Data Files**

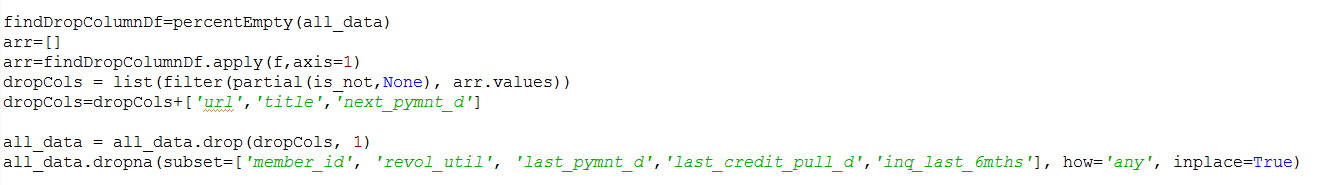
1. We have combined the csv files for all the years in one data frame and timestamped the data based on issue\_d column

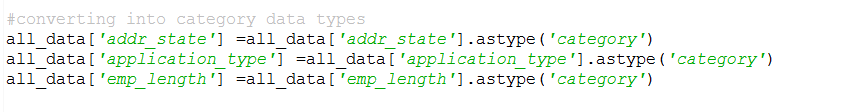


1. **Feature Engineering**:

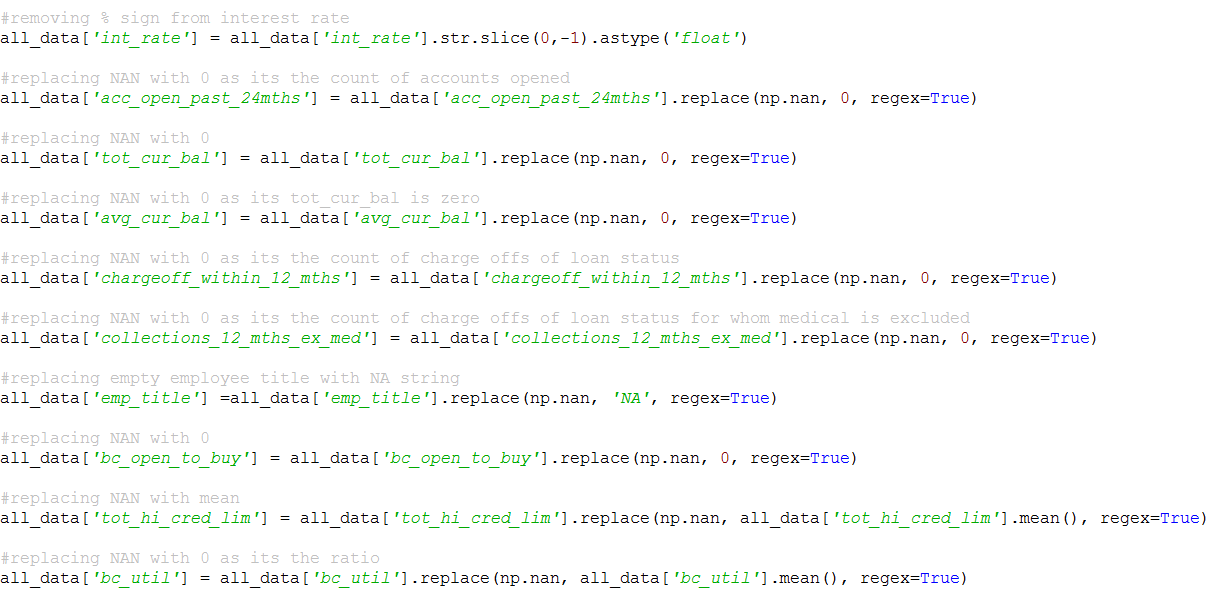
It involves removing the columns that would not be required for modeling. Any column that has more than 65% data missing would be automatically dropped. Because filling more than half the data is going to hamper the data.

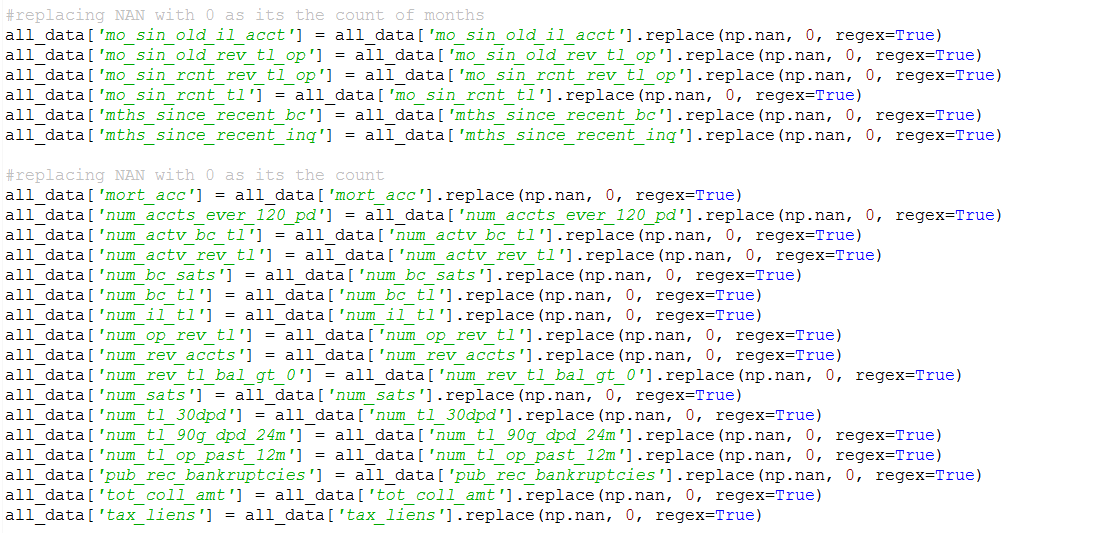


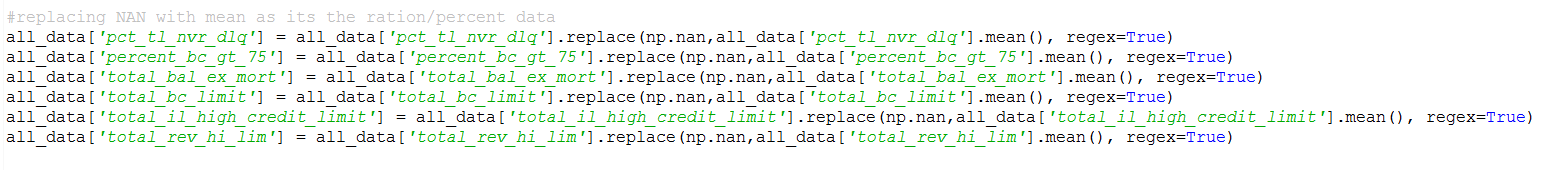


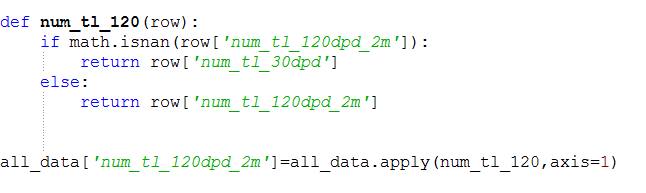


1. Filling missing values. Most of the columns were the count, hence they were replaced by 0 if empty.

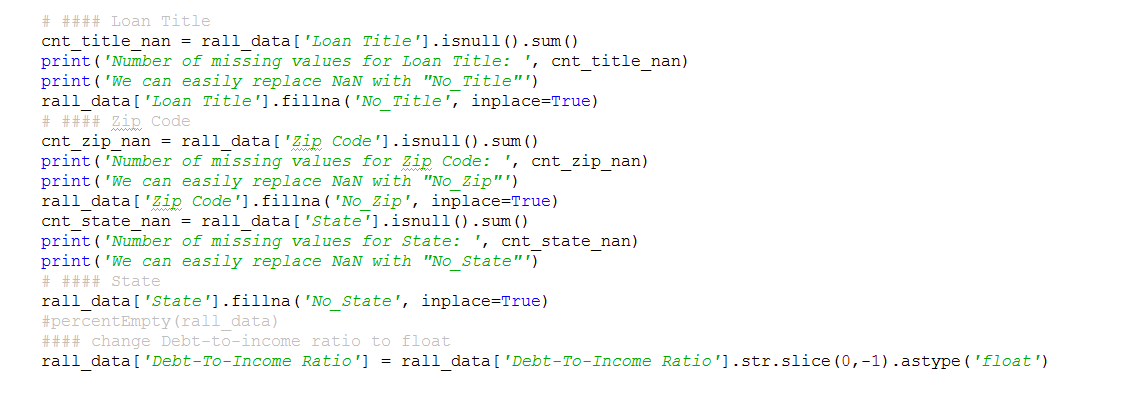




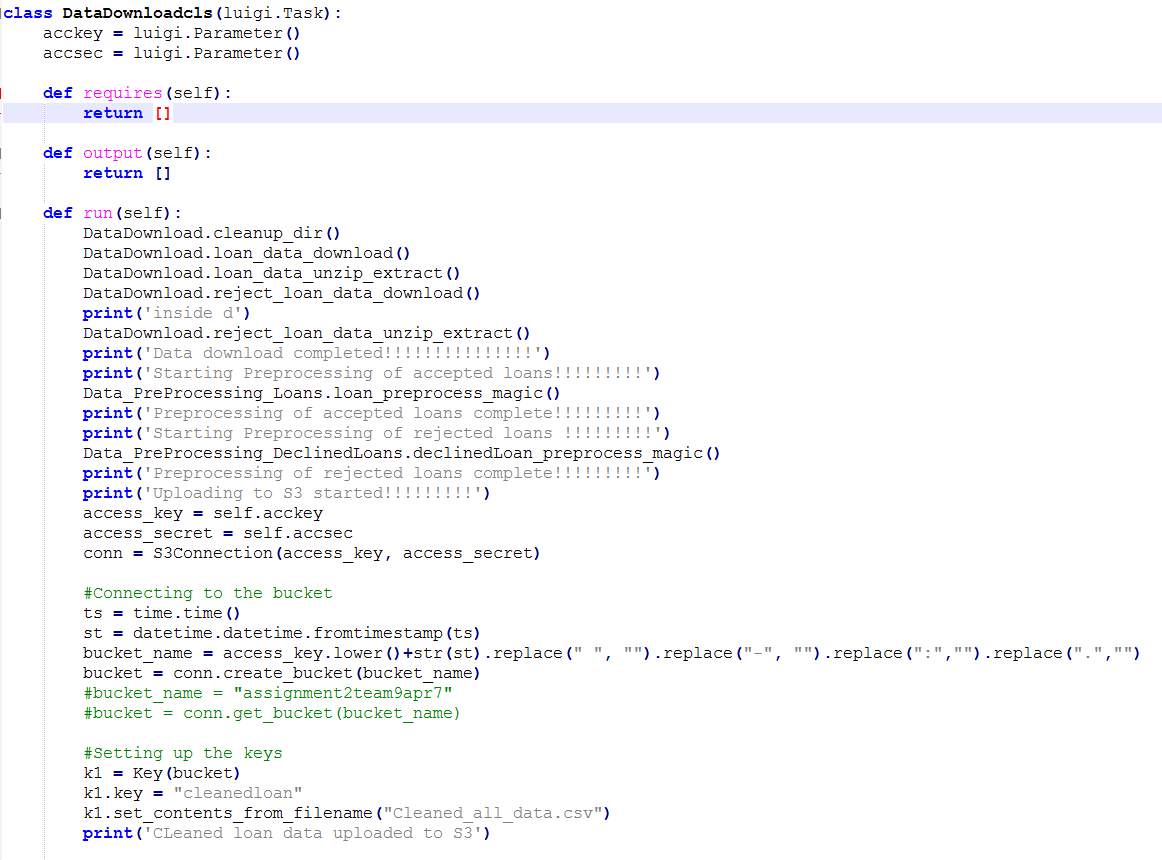


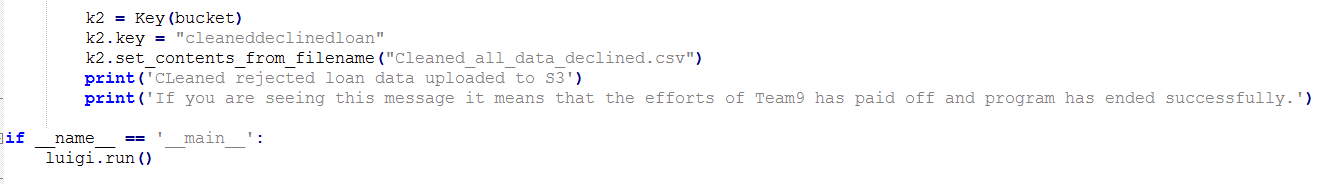


## **Declined Loan Data Files**



## **Luigi**





# **Dockerizing and Scheduling**

1. Created a docker image file-

*FROM continuumio/anaconda3*

*RUN pip install luigi*

*ADD DataDownload.py /*

*ADD Data\_PreProcessing\_Loans.py /*

*ADD Data\_PreProcessing\_DeclinedLoans.py /*

*ADD Pipeline\_Luigi.py /*

1. Running the docker image:

The program takes four arguments. Year, Amazon accessKey , Amazon secretAccessKey and location in the following format, but the order can be changed:

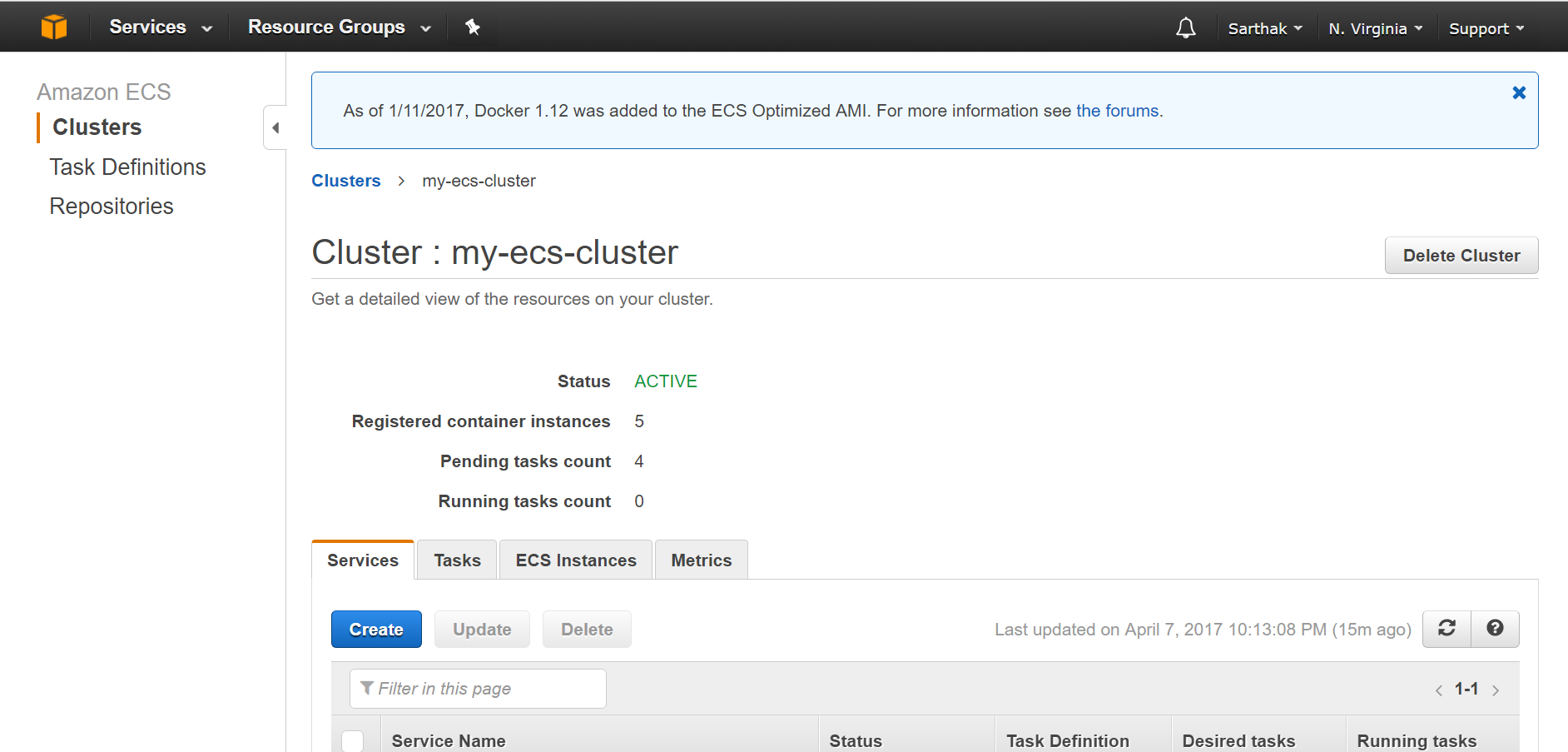
*docker run sarthakag/assignment2docker python Pipeline\_Luigi.py DataDownloadcls --local-scheduler --acckey "<access key>" --accsec "<secret access key>"*

# **Scheduling**

For scheduling purposes, we researched on deploying our Docker image on Elastic Container Service (ECS) on AWS.

The idea behind ECS is that you create an ECS Cluster—which is a group of EC2 Instances managed by ECS—define what Docker containers you want to run, and ECS will take care of deploying those containers across the Cluster, rolling out new versions, and integrating with other AWS infrastructure. The steps are:

1. Create an ECS Cluster
2. Create an ELB
3. Create IAM Roles
4. Create an Auto Scaling Group
5. Run Docker containers in your ECS Cluster



# 

We followed the official documentation and some blogs but were not able to run the Docker image on ECS.

# **Exploratory data Analysis**

## **Loan data**

These files contain complete loan data for all loans issued through the period stated, including the current loan status (Current, Late, Fully Paid, etc.) and latest payment information.

The information available for each loan consists of all the details of the loans at the time of their issuance as well as the status of loan such as how much principal has been paid so far, how much interest, if the loan was fully paid or defaulted, or if the borrower is late on payments etc

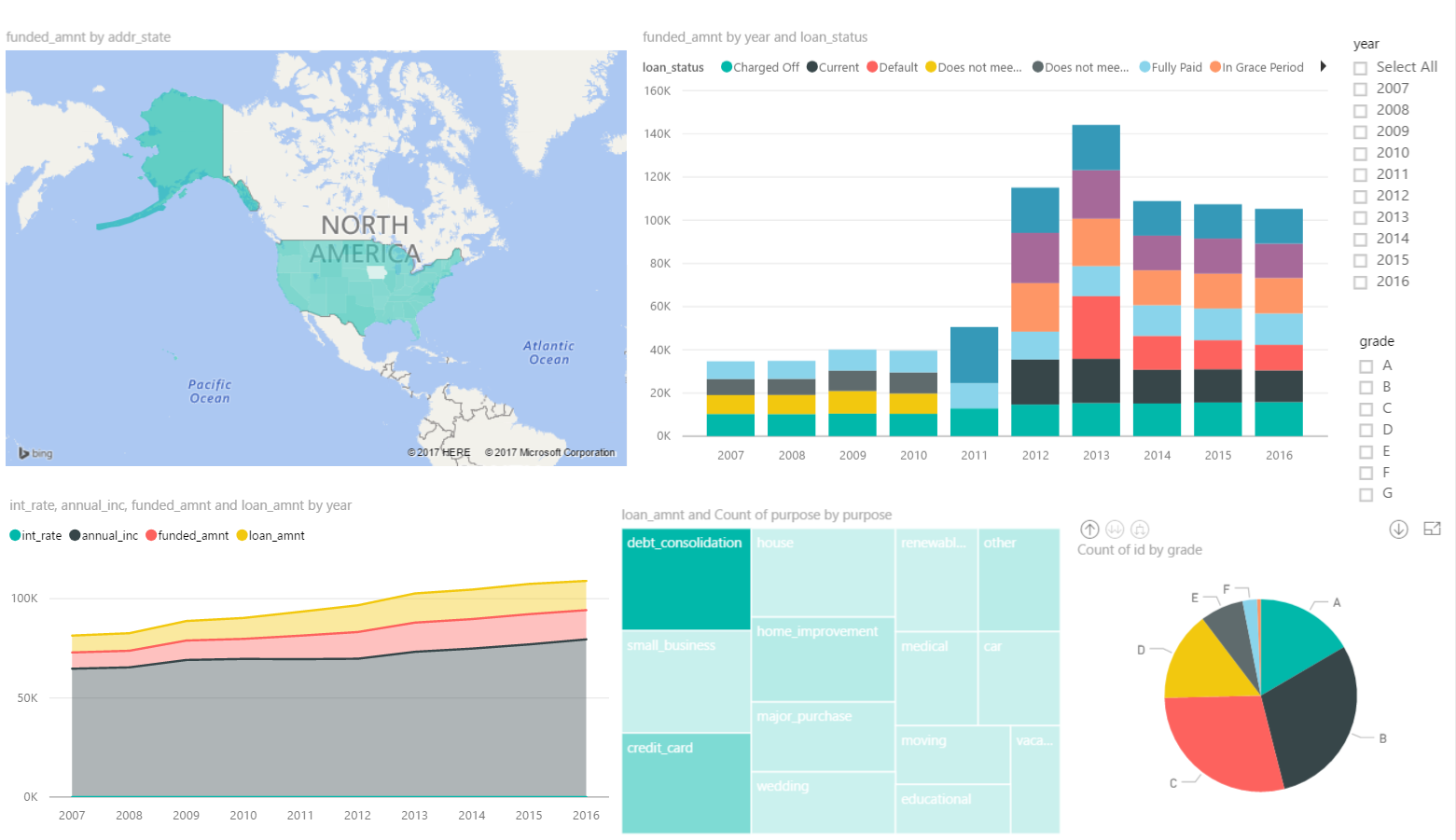
**PowerBi link**:

[https://app.powerbi.com/view?r=eyJrIjoiYWM2MTZhYWItMDUyZi00NDgxLWI2OGUtZjQzN2I3NGNiY2IxIiwidCI6IjZhYmZjNzNmLWRhNjQtNDEzNy05ZjlmLTE1ZmFhZTU2ZjY4NSIsImMiOjN9](https://app.powerbi.com/view?r=eyJrIjoiYWM2MTZhYWItMDUyZi00NDgxLWI2OGUtZjQzN2I3NGNiY2IxIiwidCI6IjZhYmZjNzNmLWRhNjQtNDEzNy05ZjlmLTE1ZmFhZTU2ZjY4NSIsImMiOjN9%20)

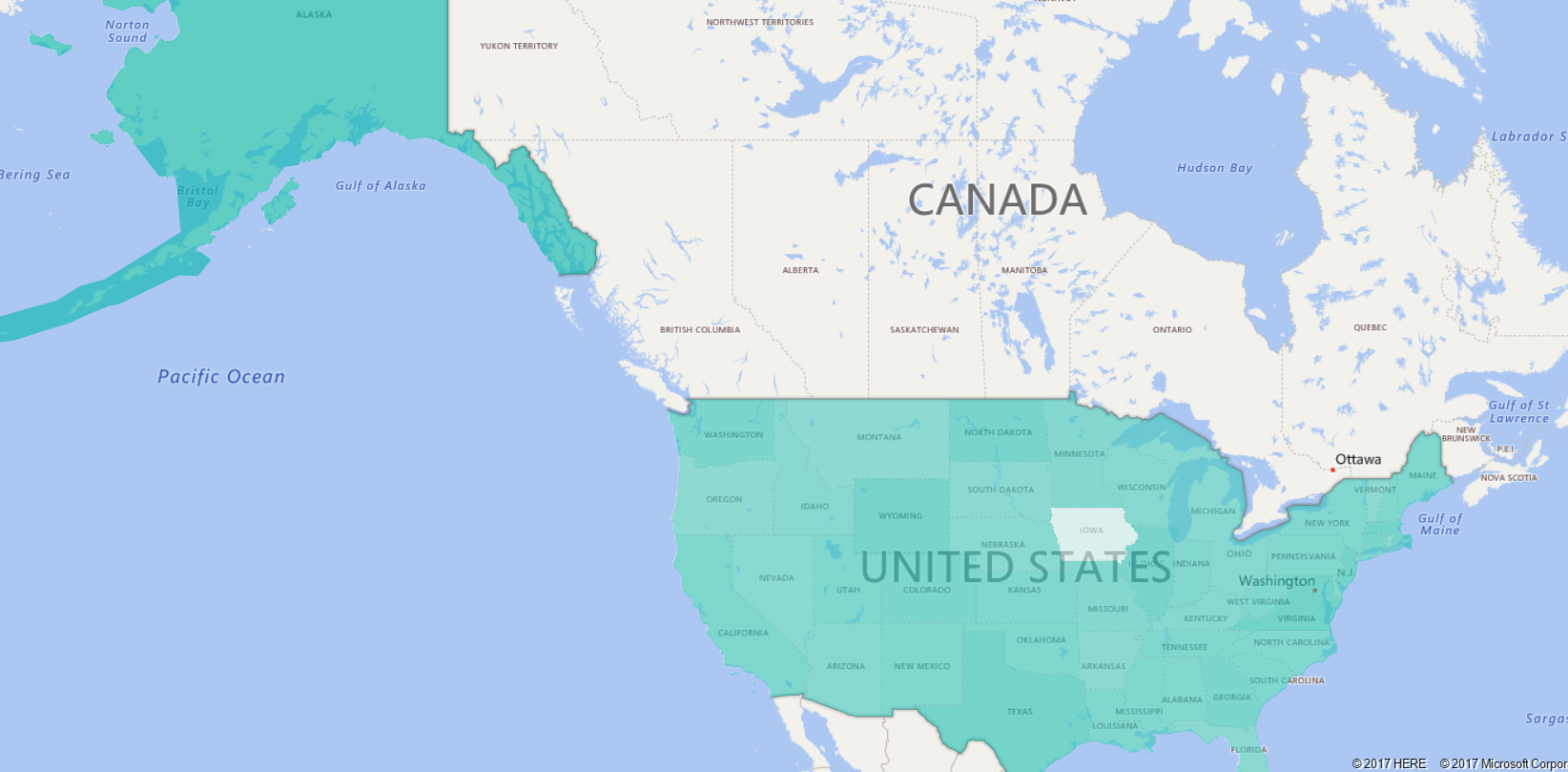
**Procedure**:

The data sets have been preprocessed. The cleaned data set have been referred to do the exploratory analysis on power BI.

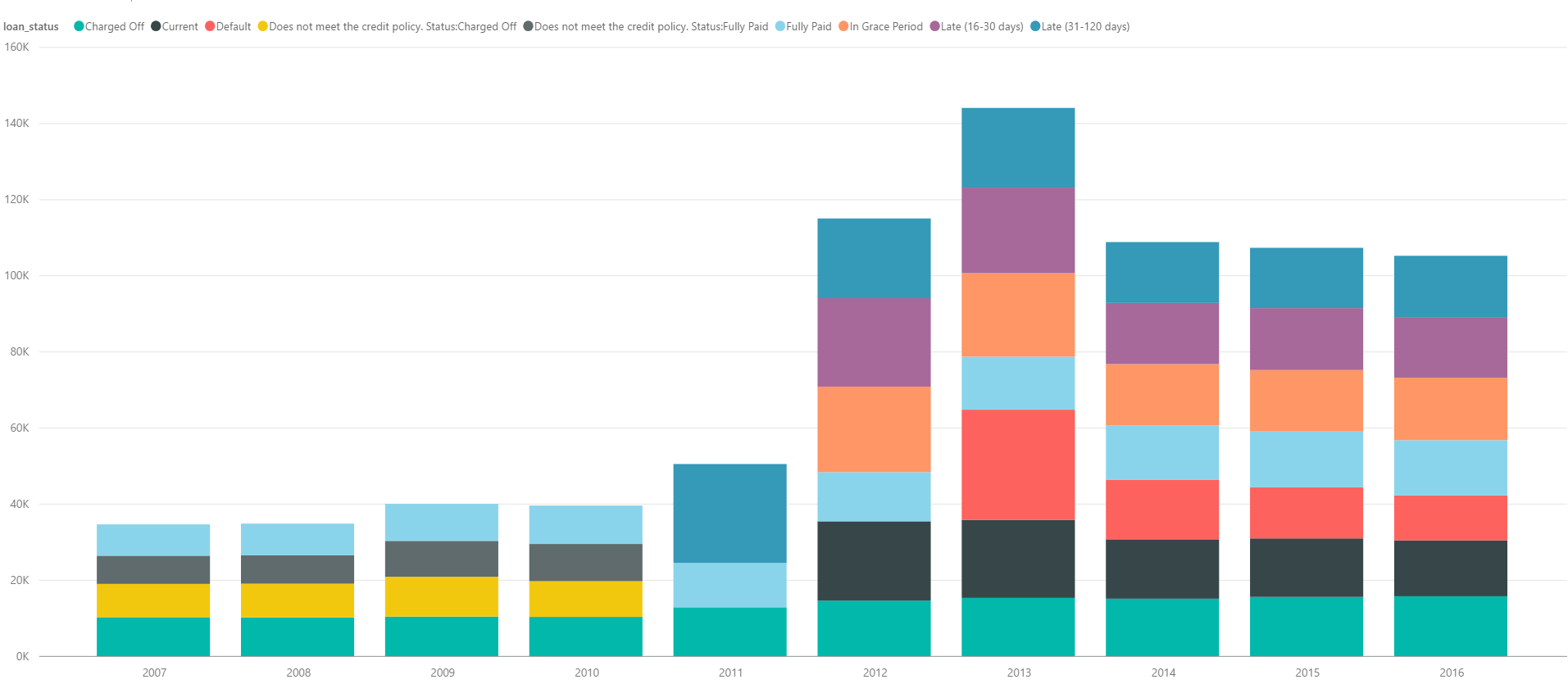
1. New columns have been generated for year and month
2. Hierarchy for year and month have been created



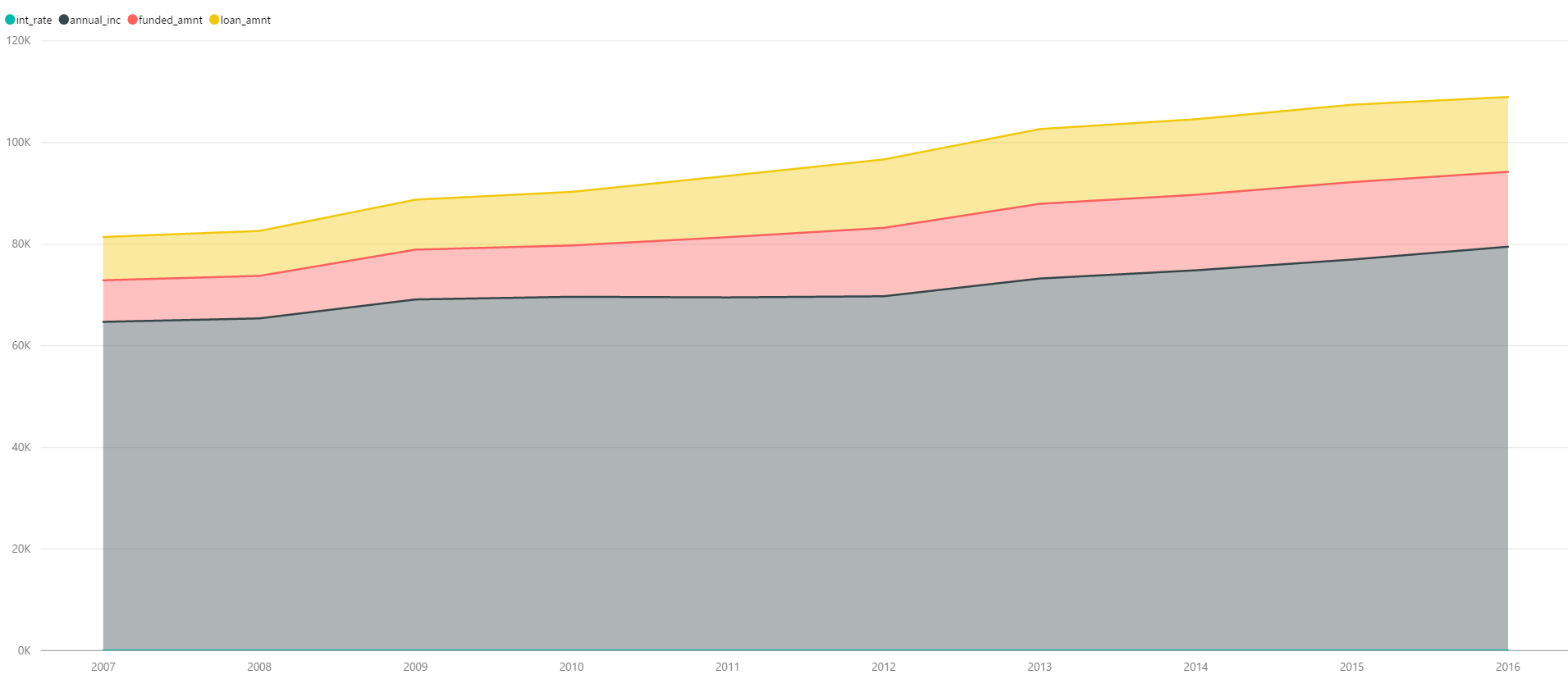
Loan counts over the years, as per region (state)



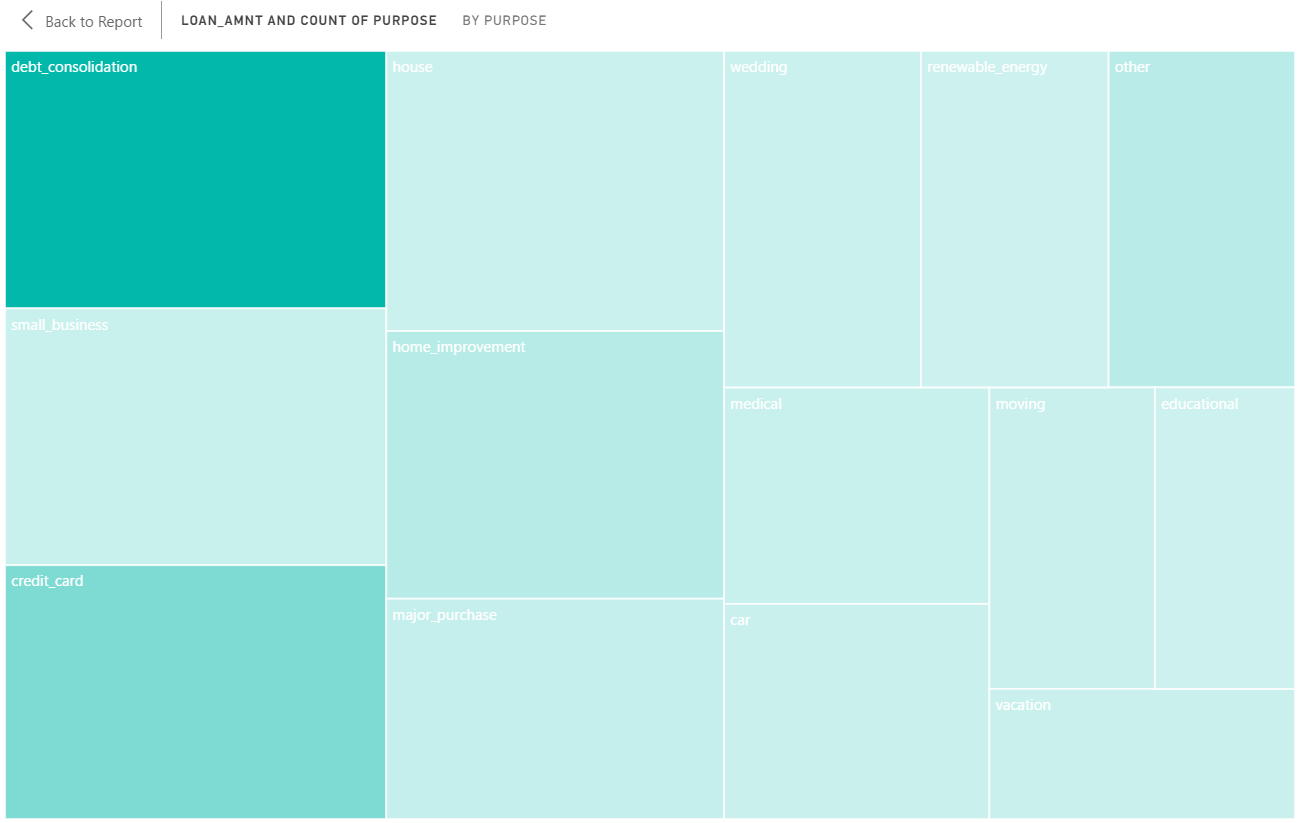
Average funded amount over the year with loan status distribution



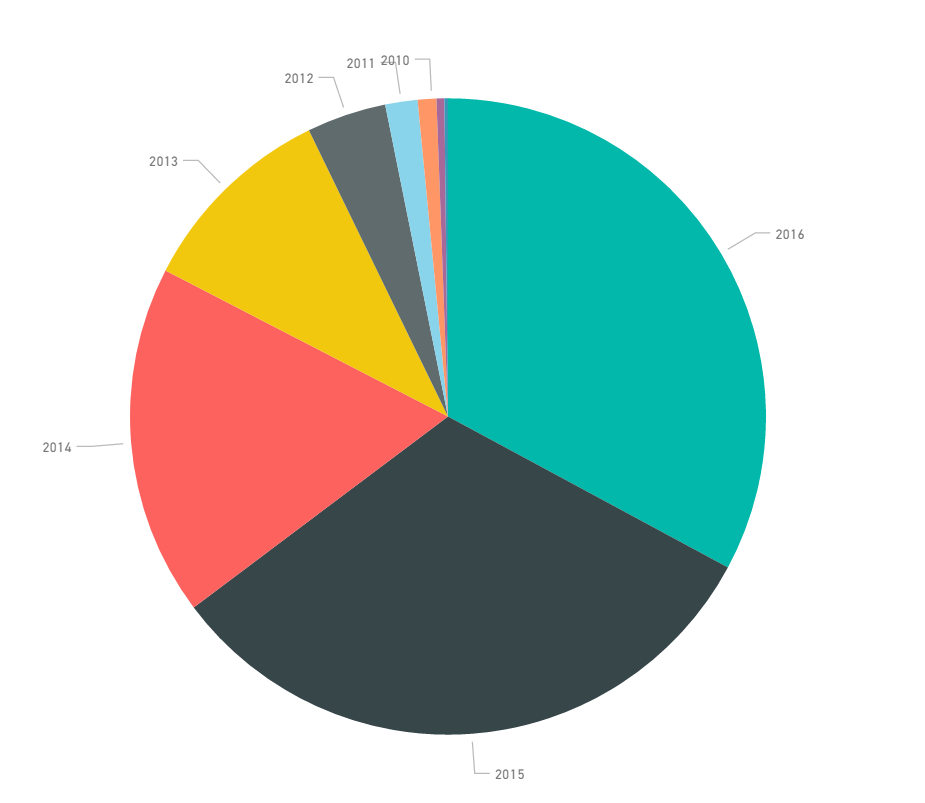
Average interest rate, average annual income, average funded amount and average loan amount by year.



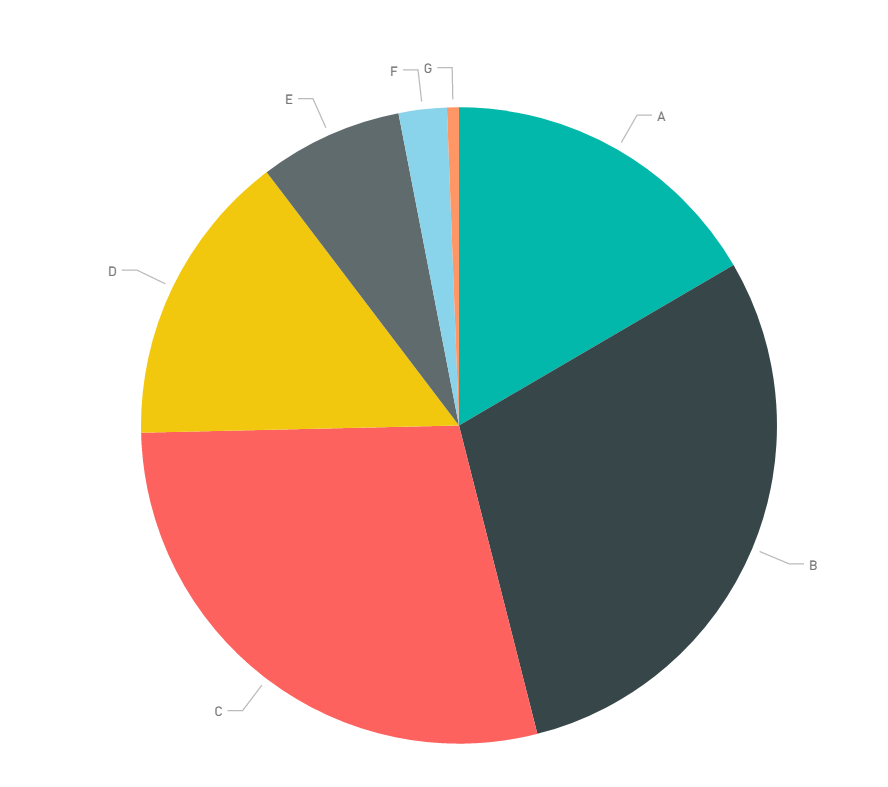
Average loan amount as per different loan purposes



Loan count by year and month



Loan count by different grades



**Observations**

* Based on each loan application and credit report, every loan is assigned a grade ranging from A1 to G5 with a corresponding interest rate. As we see that number of loan issued for grade C is the maximum with an interest rate which ranges from 12.74% - 15.99%
* Defaulting loans have higher average DTI ratio than loans with other different loan statuses.
* Defaulting loans have lower annual income than current/ fully paid loans
* Defaulting loans have higher interest rates than non-defaulting loans. We observe there are more defaults for grades other than A
* Defaulting loans are maximum for ‘Debt consolidation’ purpose
* Defaulting loans counts are maximum in Texas and New York region over the years.

## **Declined Loan Data**

These files contain the list and details of all loan applications that did not meet Lending Club's credit underwriting policy.

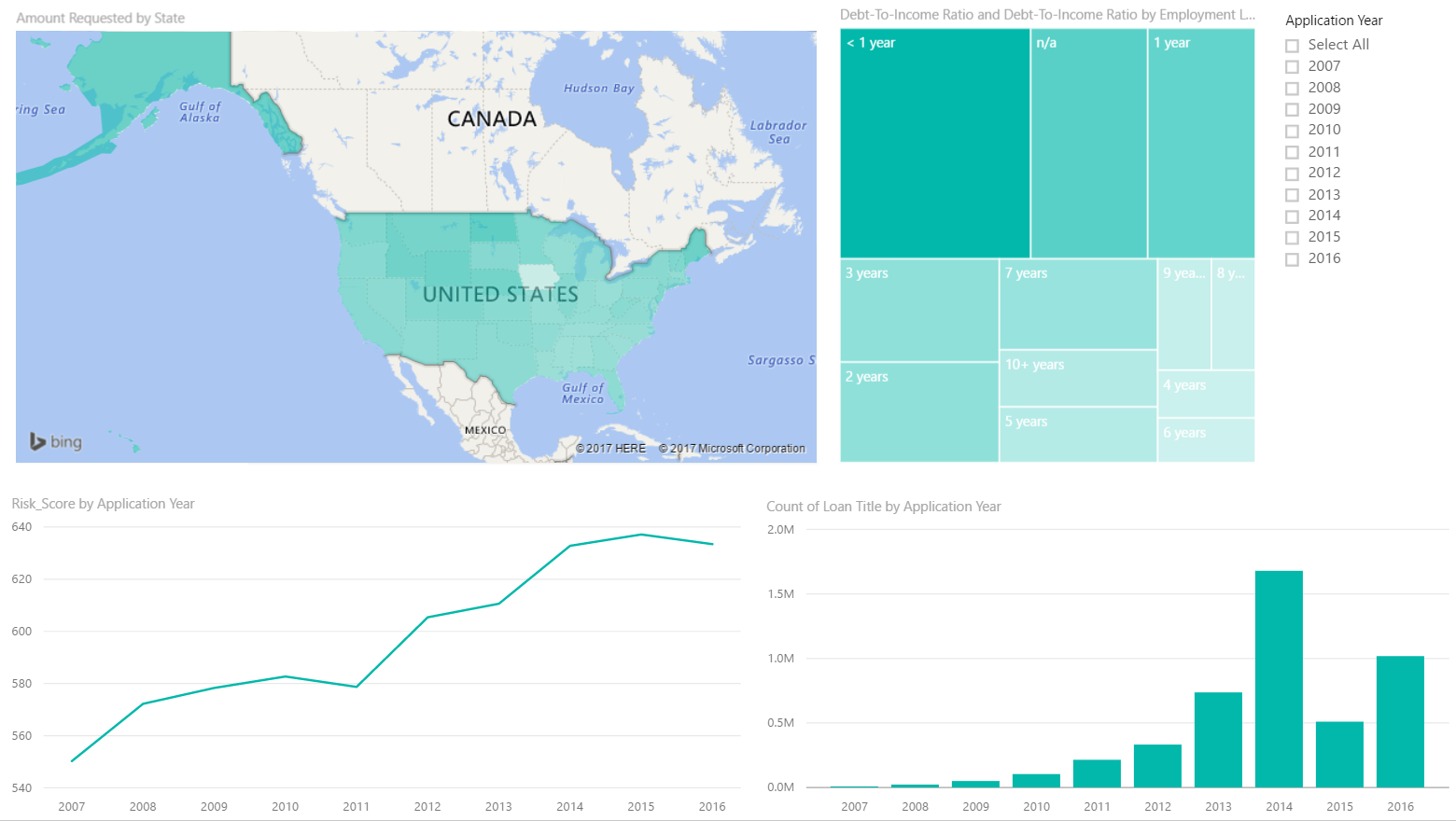
PowerBi link:

[https://app.powerbi.com/view?r=eyJrIjoiMTZhMzUwMzgtYWVlZS00NWJkLWJlODktMDdjMmNmNzJhMzZhIiwidCI6IjZhYmZjNzNmLWRhNjQtNDEzNy05ZjlmLTE1ZmFhZTU2ZjY4NSIsImMiOjN9](https://app.powerbi.com/view?r=eyJrIjoiMTZhMzUwMzgtYWVlZS00NWJkLWJlODktMDdjMmNmNzJhMzZhIiwidCI6IjZhYmZjNzNmLWRhNjQtNDEzNy05ZjlmLTE1ZmFhZTU2ZjY4NSIsImMiOjN9%20)

**Procedure**:

The data sets have been preprocessed. The cleaned data set have been referred to do the exploratory analysis on power BI.

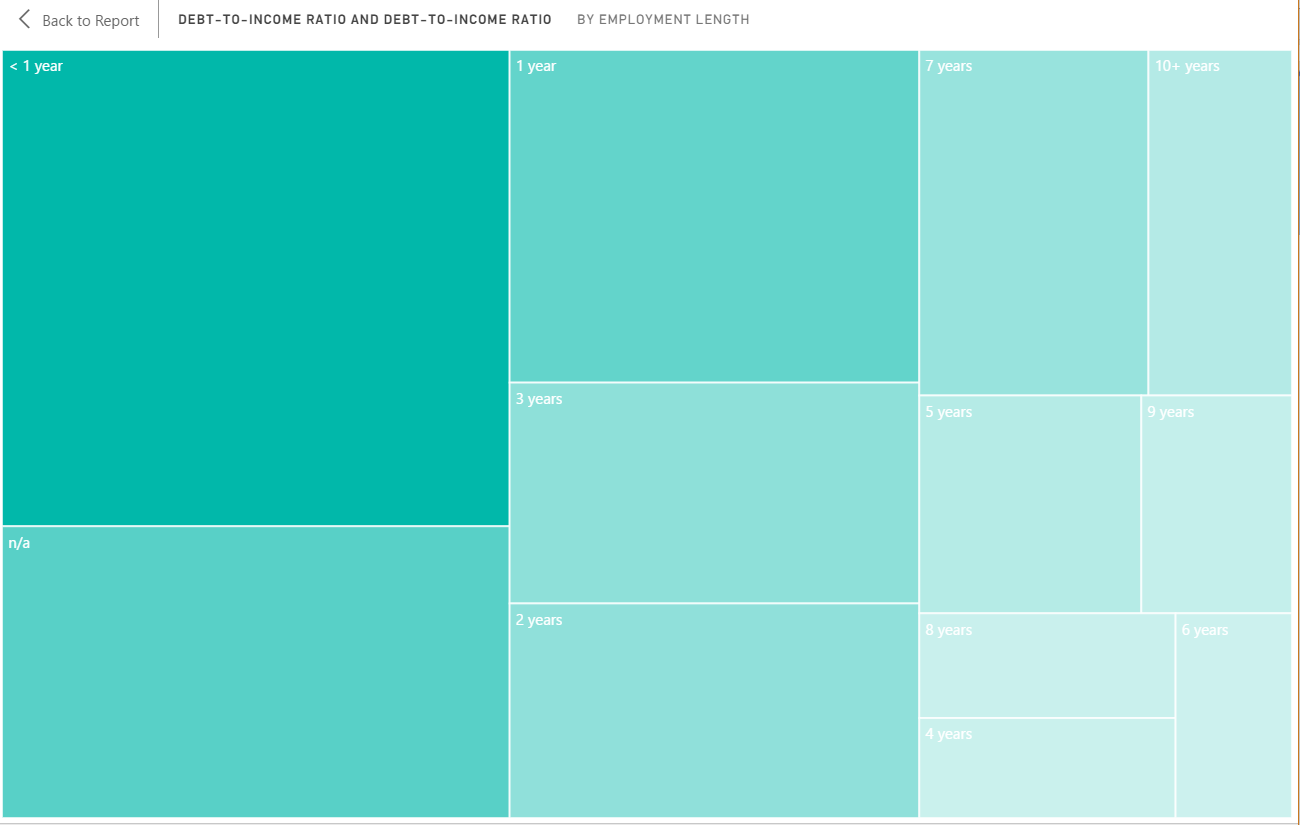
1. For declined data set - the ‘Application Date’ column has been split to create three different columns, then a date hierarchy has been created taking year, month and day



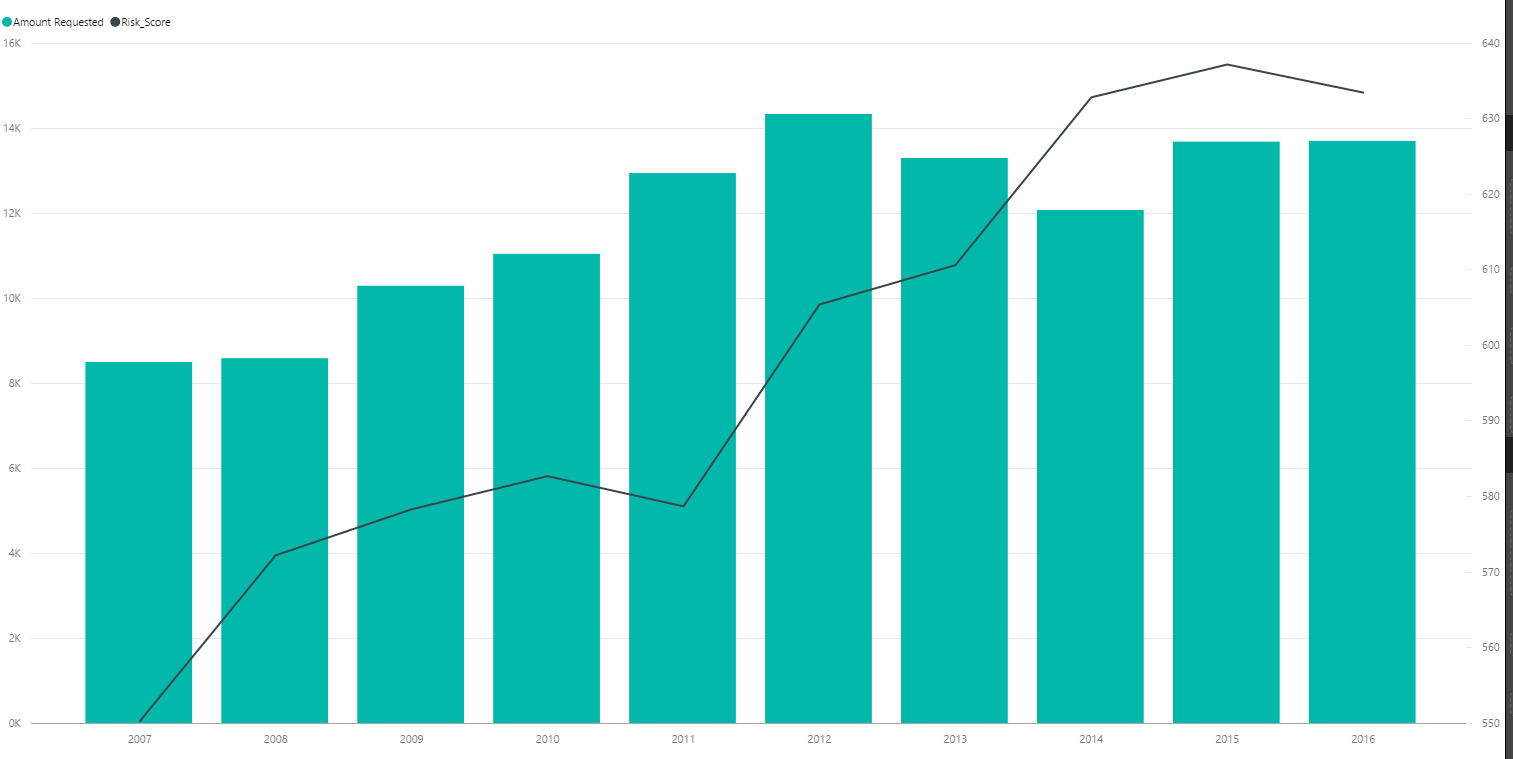
Loan counts over the years by region (state)



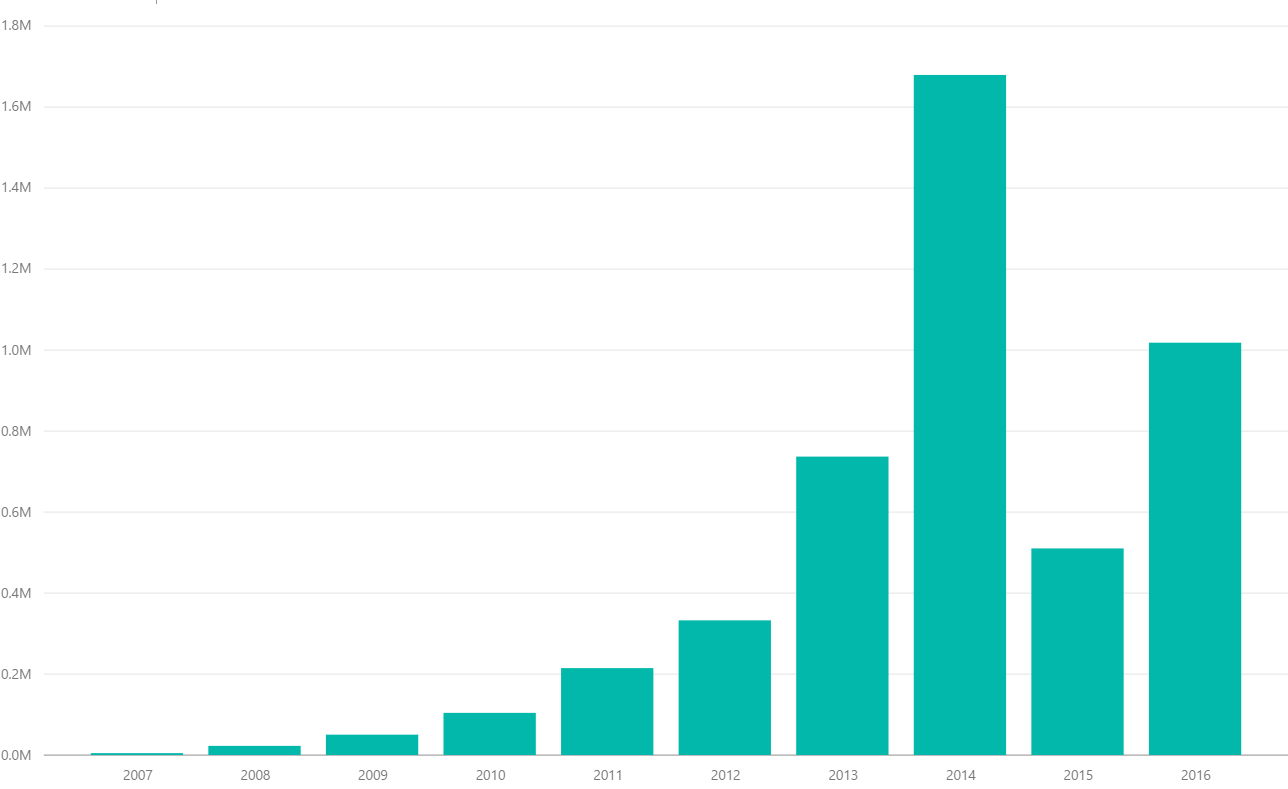
Average Debt-To-Income ratio, as per length of employment



Risk score and amount requested over the years



Count of loans over the years



**Observations**

* For applications prior to November 5, 2013 the risk score is the borrower's FICO score. For applications after November 5, 2013 the risk score is the borrower's Vantage score. We observer that risk score and amount requested is greater for borrower’s whose length of employment is greater.
* The average DTI ratio is higher for borrower’s whose length of employment is less than 1 year
* Maximum loans are from CA region

## **Jupyter Notebook**

The exploratory data analysis has been discussed in the uploaded Jupyter Notebook itself.

# **Contributions**

# C:\Users\pragati\AppData\Local\Microsoft\Windows\INetCacheContent.Word\Untitled.png

# **References**

<https://www.lendingclub.com/info/download-data.action>