

After Sales Services of Financial Companies and Institutions

Project Report

Collect/ Store/Retrieve Data

DA 5020 – Spring 2017 – Prof Kathleen Durant

Aditya Kiran E R – 001250409

Himadri Sardana – 001698954



Northeastern University

Table of Contents

<i>Project Proposal</i>	<i>3</i>
<i>Project Goal</i>	<i>3</i>
<i>Technical Details</i>	<i>3</i>
<i>What is this Data?</i>	<i>4</i>
<i>Scraping</i>	<i>5</i>
<i>Loading</i>	<i>5</i>
<i>Mongo Installation</i>	<i>5</i>
<i>Sample Data</i>	<i>6</i>
<i>Code Execution</i>	<i>6</i>
<i>Top 10 companies with complaints Visualization</i>	<i>7</i>
<i>Total complaint volume Visualization</i>	<i>8</i>
<i>Top 5 Companies with product complaint volume</i>	<i>8</i>
<i>Comparison of product complaint volume for MA and NJ</i>	<i>9</i>
<i>Mortgage Complaint Volume</i>	<i>9</i>
<i>Product volume trend change</i>	<i>10</i>
<i>Conclusion</i>	<i>11</i>
<i>References</i>	<i>11</i>

Project Proposal

Finances are probably one of the largest part of our lives where planning is of extreme importance. Everybody must plan and deal with finances on a day to day basis. When there are millions of people dealing with finances, there are bound to be problems that organizations which provide financial products. Some problems more common than the others.

Project Goal

We decided to analyze data available on the Consumer Financial Protection Bureau and compare trends, and changes in complaint volumes over the years for different products to help the companies improve their after sales service.

Technical Details

We planned to determine what the most frequently occurring complaint for a company is, and then come up with efficient ways to solve the issue and satisfy the consumer.

We used grepsr to extract our data and Stored the data for easy extraction into a Mongo database and extracted the required data against queries that were used to analyze the extracted data and provide companies with statistics on what their most frequently occurring complaints are and how they can reduce them and consumers with statistics to choose the best company and product.

All our statistical graphs were drawn using plotly

What is this data?

Each week thousands of consumers' complaints are sent about financial products and services to companies for response. Complaints are listed in the database after the company responds or after they've had the complaint for 15 calendar days, whichever comes first. The data has consumer's description of what happened if the consumer opts to share it and after taking steps to remove personal information.

The Consumer Complaint Database contains complaints we've received about consumer financial products and services, including:

- Bank accounts or services
- Consumer loans
- Credit cards
- Credit reporting
- Debt collection
- Money transfers
- Mortgages
- Prepaid cards
- Payday loans
- Student loans (private student loans and federal student loan servicing)
- Virtual currency
- Other consumer loan (such as pawn and title loans)
- Other financial service (such as credit repair and debt settlement)

[illegible]

```
df<-read.csv("Consumer_Complaints.csv", header=TRUE, sep = ",")
##---cleaning the data before loading into db-----
#removing periods from column names and using single space instead
names(df)<-gsub(x = names(df),
                pattern = "(\\.\\.\\.)+",
                replacement = " " )
#removing unnecessary strings
df$`Sub product`<-gsub("I do not know"," ",df$`Sub product`)
#putting all dates in one format
df$`Date received`<-gsub("-", "/",df$`Date received`)
df$`Date sent to company`<-gsub("-", "/",df$`Date sent to company`)
#storing them as DATE type
df[[1]]<- as.Date(df[[1]], "%m/%d/%Y")
df[[11]]<- as.Date(df[[11]], "%m/%d/%Y")
```

```

2017-04-23T11:17:05.738-0400 I NETWORK [thread1] waiting for connections on port 27017
2017-04-23T11:17:05.768-0400 I NETWORK [thread1] connection accepted from 127.0.0.1:52864 #1 (1 connection now open)
2017-04-23T11:17:05.772-0400 I NETWORK [conn1] received client metadata from 127.0.0.1:52864 conn1: { application: { name: "MongoDB Shell" }, driver: { name: "MongoDB Internal Client", version: "3.4.3" }, os: { type: "Windows", name: "Microsoft Windows 8", architecture: "x86_64", version: "6.2 (build 9200)" } }
2017-04-23T11:17:39.797-0400 I NETWORK [thread1] connection accepted from 127.0.0.1:52865 #2 (2 connections now open)
2017-04-23T11:17:39.798-0400 I NETWORK [conn2] received client metadata from 127.0.0.1:52865 conn2: { driver: { name: "MongoDB", version: "1.6.2" }, os: { type: "Windows", name: "Windows", version: "6.2 (9200)", architecture: "x86_64" }, platform: { cflags: "gcc386e driver=199901 CC=GCC 4.9.3 CFLAGS="" LD_FLAGS="" } }
2017-04-23T11:18:11.462-0400 I COMMAND [conn2] command test.db.getMore { getMore: 21856229377, collection: "df" } originatingCommand { find: "df", filter: {}, projection: {}, sort: { _id: 1 }, skip: 0, limit: 0, noCursorTimeout: false } planSummary: IXSCAN { _id: 1 } cursorid: 21856229377 keysExamined: 34668 docExamined: 34668 numYields: 272 nreturned: 34668 reslen: 16776886 locks: { Global: { acquireCount: { r: 546 } }, Database: { acquireCount: { r: 273 } }, Collection: { acquireCount: { r: 273 } } } protocol: op_query 103ms
2017-04-23T11:18:23.772-0400 I COMMAND [conn2] command test.db.getMore { getMore: 21856229377, collection: "df" } originatingCommand { find: "df", filter: {}, projection: {}, sort: { _id: 1 }, skip: 0, limit: 0, noCursorTimeout: false } planSummary: IXSCAN { _id: 1 } cursorid: 21856229377 keysExamined: 34864 docExamined: 34864 numYields: 278 nreturned: 34864 reslen: 16776942 locks: { Global: { acquireCount: { r: 558 } }, Database: { acquireCount: { r: 279 } }, Collection: { acquireCount: { r: 279 } } } protocol: op_query 148ms

```

Sample Data from CSV

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Date received	Product	Sub-product/Issue	Sub-issue	Company	State		ZIP code	Consumer	Submitted	Date sent to company	Company response to consu	Timely res	Consumer	Complaint ID	
2	7/29/2013	Consumer	Vehicle loan	Managing the loan or	Wells Fargo	VA		24540	N/A	Phone	7/30/2013	Closed with explanation	Yes	No	468882	
3	7/29/2013	Bank accou	Checking a	Using a debit or ATM c	Wells Fargo	CA		95992	N/A	Web	7/31/2013	Closed with explanation	Yes	No	468889	
4	7/29/2013	Bank accou	Checking a	Account opening, closi	Santander	NY		10065	N/A	Fax	7/31/2013	Closed	Yes	No	468879	
5	7/29/2013	Bank accou	Checking a	Deposits and withdraw	Wells Fargo	GA		30084	N/A	Web	7/30/2013	Closed with explanation	Yes	No	468949	
6	7/29/2013	Mortgage	Conventior	Loan servicing, payme	Franklin Cr	CT		6106	N/A	Web	7/30/2013	Closed with explanation	Yes	No	475823	
7	7/29/2013	Bank accou	Checking a	Deposits and withdraw	Bank of An	TX		75025	N/A	Web	7/30/2013	Closed with explanation	Yes	No	468981	
8	7/29/2013	Debt collec	Other (i.e.)	Cont'd atte	Debt is not	NRA Group	VA	20147	N/A	Web	8/7/2013	Closed with non-monetary r	Yes	No	467801	
9	7/29/2013	Debt collec	I do not kn	Cont'd atte	Debt was p	SunTrust B	FL	32818	N/A	Referral	8/1/2013	Closed with explanation	Yes	Yes	475728	
10	7/29/2013	Credit card		Billing statement	Citibank	OH		45247	N/A	Referral	7/30/2013	Closed with explanation	Yes	Yes	469026	
11	7/29/2013	Mortgage	Other mor	Loan servicing, payme	Wells Fargo	NV		89511	N/A	Referral	7/30/2013	Closed with explanation	Yes	Yes	469035	
12	7/29/2013	Mortgage	Other mor	Loan modification, coll	Bank of Am	NC		27949	N/A	Referral	7/30/2013	Closed with non-monetary r	Yes	No	469037	
13	7/29/2013	Mortgage	Other mor	Loan servicing, payme	JPMorgan	CA		90703	N/A	Referral	7/30/2013	Closed with explanation	Yes	No	469284	
14	7/29/2013	Mortgage	Other mor	Loan modification, coll	Citibank	CA		95821	N/A	Referral	7/31/2013	Closed with explanation	Yes	Yes	480488	
15	7/29/2013	Debt collec	Credit card	Communic	Frequent o	Asset Man	PA	19145	N/A	Phone	8/1/2013	Closed with explanation	No		468904	
16	7/29/2013	Debt collec	Credit card	Communic	Frequent o	Wells Fargo	PA	19145	N/A	Phone	8/1/2013	Closed with non-monetary r	Yes	No	468905	
17	7/29/2013	Mortgage	Conventior	Loan modification, coll	JPMorgan	NY		14092	N/A	Phone	7/31/2013	Closed with explanation	Yes	No	469057	
18	7/29/2013	Mortgage	Conventior	Loan modification, coll	Ocwen	FL		33426	N/A	Web	7/31/2013	Closed with explanation	Yes	Yes	469060	
19	7/29/2013	Mortgage	Conventior	Loan modification, coll	JPMorgan	NY		10019	N/A	Web	7/30/2013	Closed with explanation	Yes	Yes	469070	
20	7/29/2013	Mortgage	Conventior	Loan servicing, payme	Citibank	NJ		7604	N/A	Web	7/30/2013	Closed with explanation	Yes	No	472236	
21	7/29/2013	Credit card		APR or interest rate	Synchrony	WA		98548	N/A	Web	7/29/2013	Closed with monetary relief	Yes	No	469131	

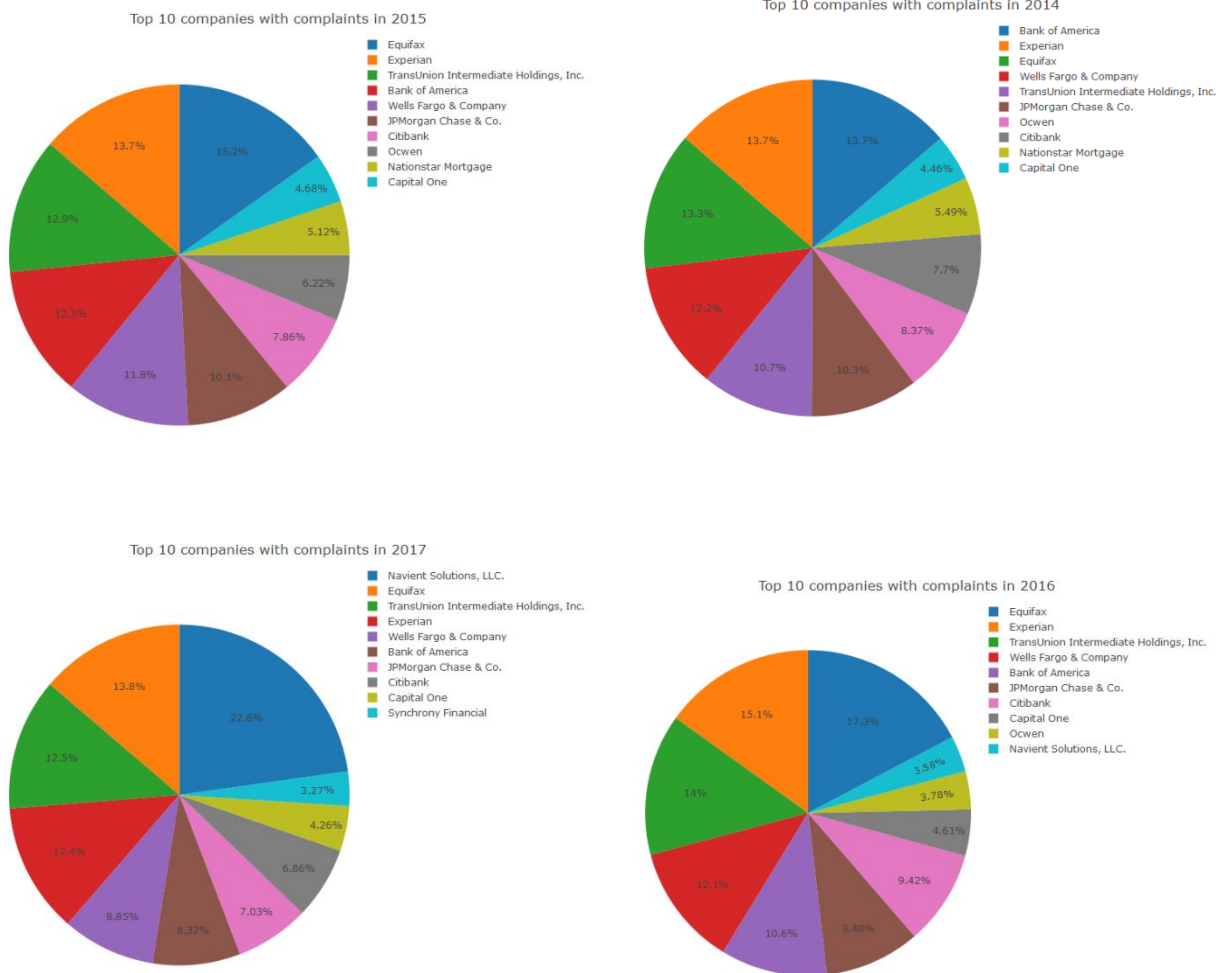
Successful execution of code

```

> library(mongolite)
> library(plotly)
>
> #loading csv as usual
> df<-read.csv("Consumer_Complaints.csv", header=TRUE, sep = ",")
> ##----cleaning the data before loading into db-----
> #removing periods from column names and using single space instead
> names(df)<-gsub(x = names(df),
+               pattern = "(\\.|_)+",
+               replacement = " ")
> #removing unnecessary strings
> df$`sub product`<-gsub("I do not know", " ",df$`sub product`)
> #putting all dates in one format
> df$`Date received`<-gsub("-", "/",df$`Date received`)
> df$`Date sent to company`<-gsub("-", "/",df$`Date sent to company`)
> #storing them as DATE type
> df[[1]]<- as.Date(df[[1]], "%m/%d/%Y")
> df[[11]]<- as.Date(df[[11]], "%m/%d/%Y")
>
>
> ##-----inserting data into mongoDB-----
> mongoDf<- mongo("df")
> mongoDf$insert(df)
List of 5
 $ nInserted : num 751524
 $ nMatched   : num 0
 $ nRemoved   : num 0
 $ nUpserted  : num 0
 $ writeErrors: list()
> #exporting data to display the inserted data; see screenshots
> mongoDf$export(file("text.txt"))
> CompanyComplaintVolume <- as.data.frame(table(mongoDf$find(fields = '{"company" : 1, "field2":1, "_id":0}')))
>
> ## Arranging in descending order of volume content
> CompanyComplaintVolume <- CompanyComplaintVolume[rev(order(CompanyComplaintVolume$Freq)),]
>
>
> ## Analysing top 100 rows
> CompanyComplaintVolume2 <- CompanyComplaintVolume[1:5,]
>
> ## Companies by volume
>
> p <- plot_ly(
+   x = CompanyComplaintVolume2[[1]],
+   y = CompanyComplaintVolume2[[2]],
+   name = "Company Comparison",
+   type = "bar"
+ )
> p

```

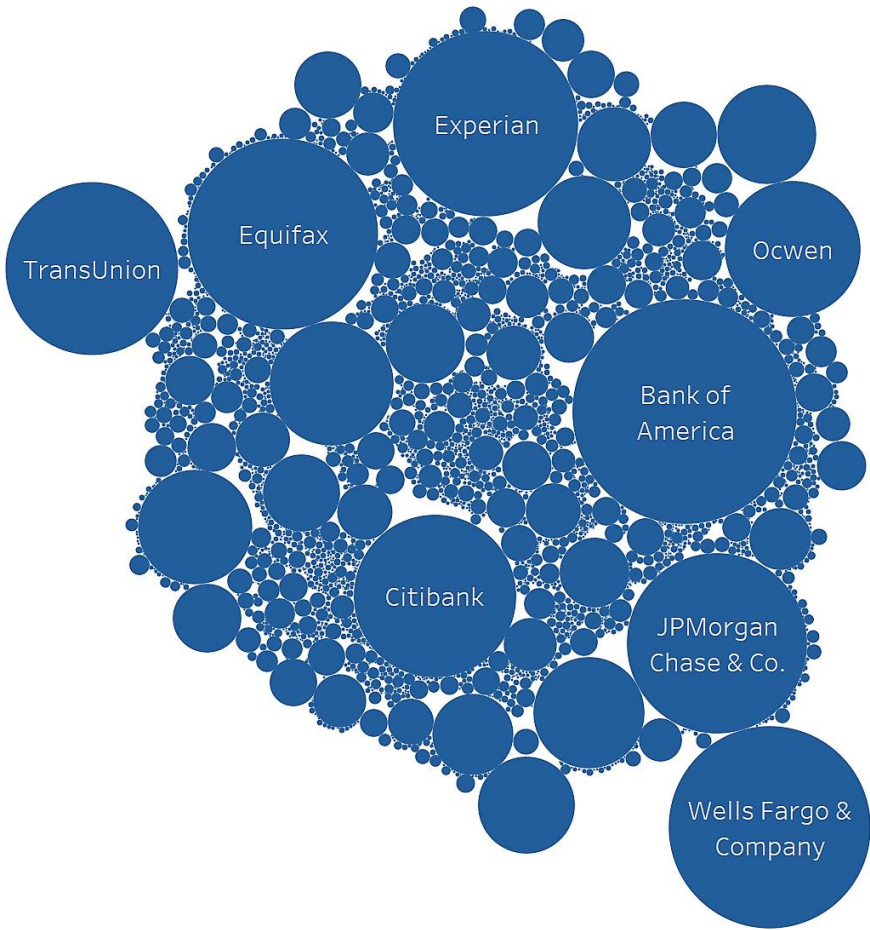
Analysing Data to get the top 10 companies with complaints in Years 2017-2014



From the above charts, we can see that the complaints for mortgages has seen a decrease in number over the years while the complaint volume for credit reporting has increased. This analysis is helpful in communicating with financial organizations the need for better product service for credit reporting.

Further, an analysis for the top 5 companies with maximum complaint volume has been performed , which helps the Consumer Finance bureau communicate the necessary improvement areas for those organizations.

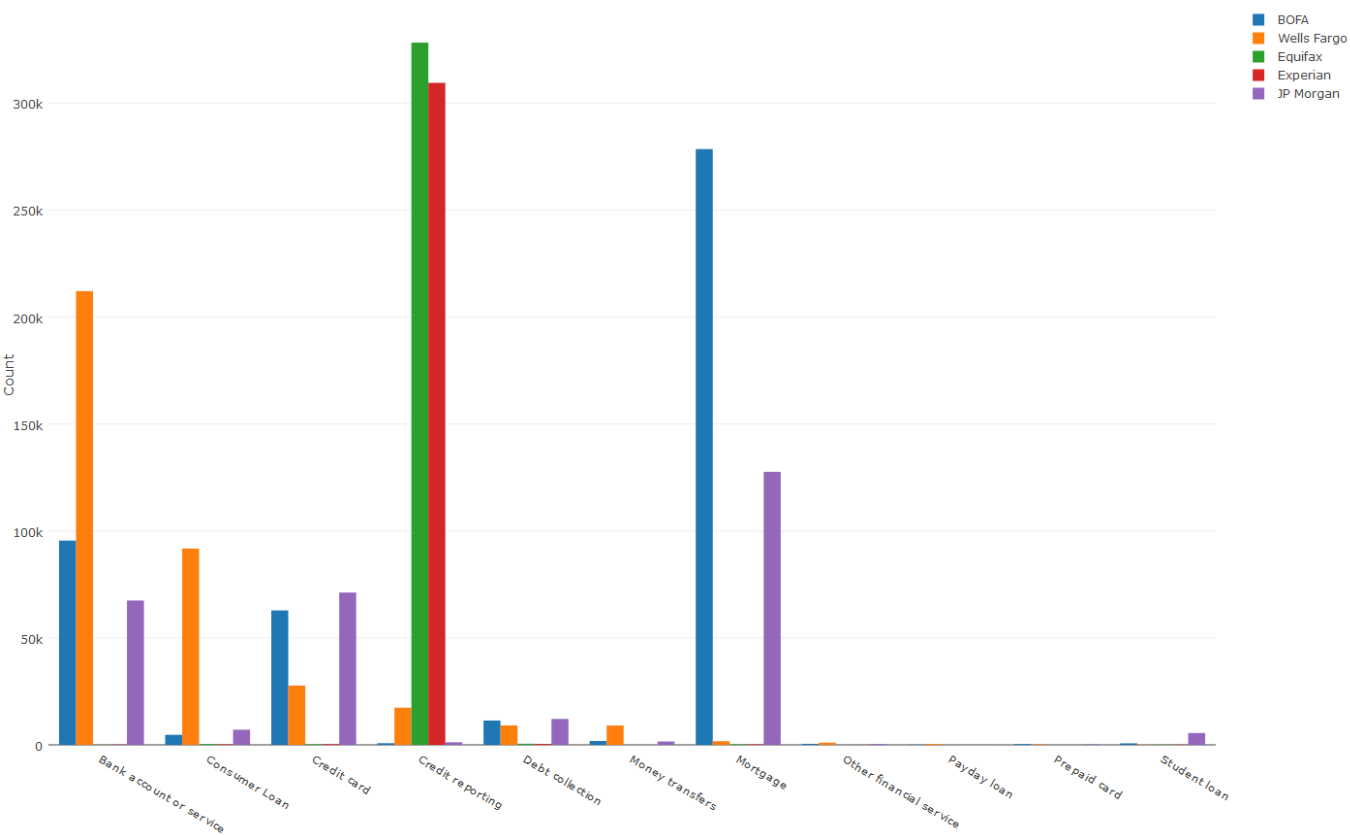
Total Complaint Volume with respect to companies



Top 5 Companies with complaints

Name of company	Freq
Bank of America	457135
wells Fargo & Company	370643
Equifax	330064
Experian	311157
JPMorgan Chase & Co.	294490

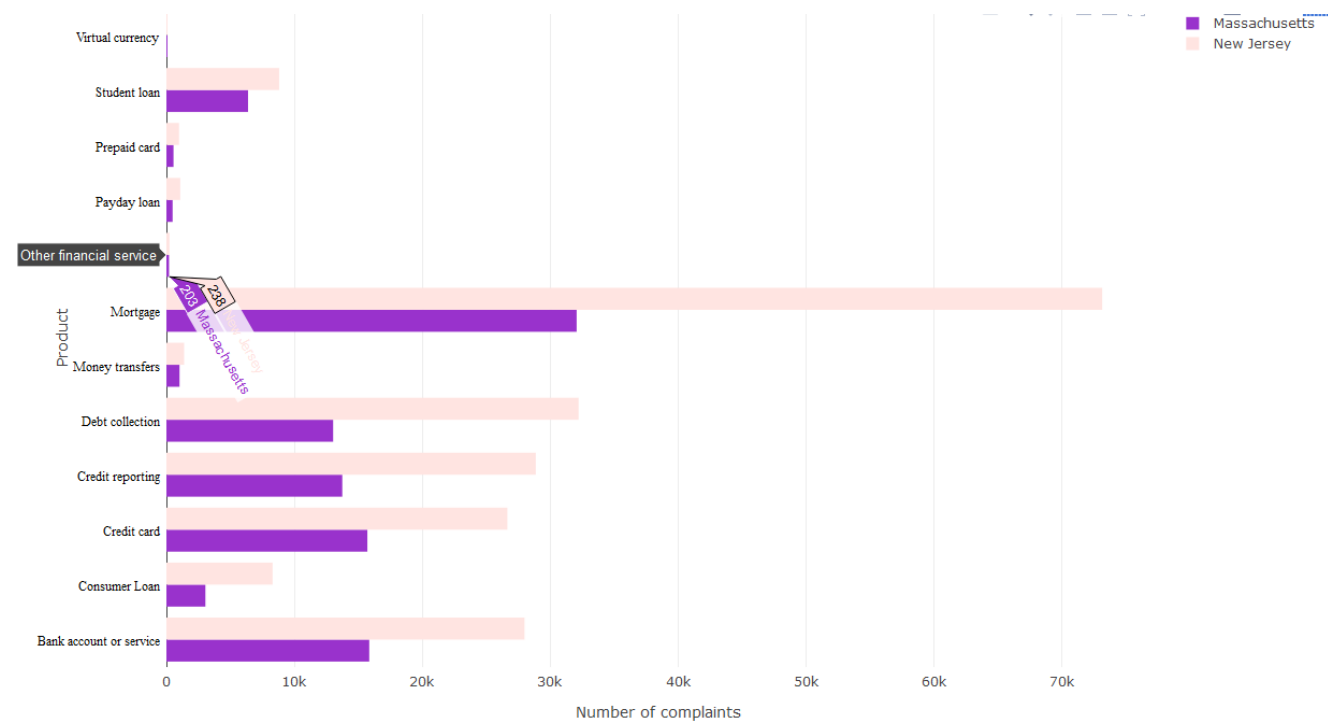
Visualization for Top 5 Companies with complaints with respect to the Product



From the above chart, we can conclude that Bank of America needs to work on their Mortgage product, Equifax on their credit reporting product, Wells Fargo on their bank account services etc.

The above chart helps different companies focus on the after sales support for products with the highest complaint volume.

Chart showing product comparison for two different states

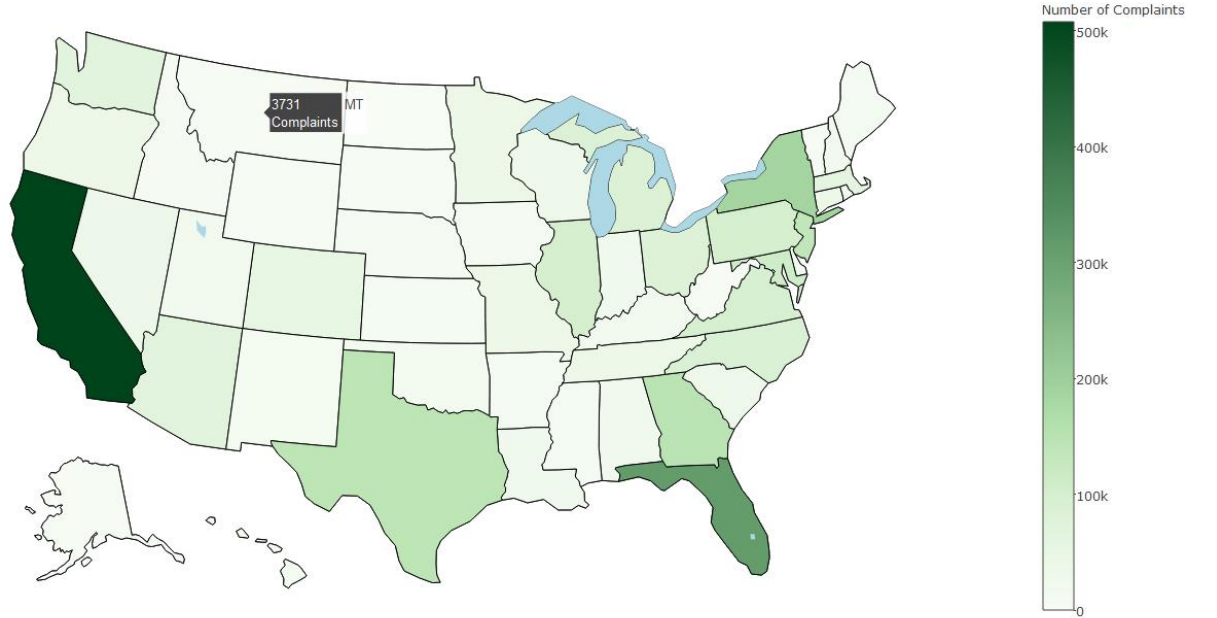


The above chart helps a customer choose where they buy a product. It also helps a company narrow down their improvement process for the product.

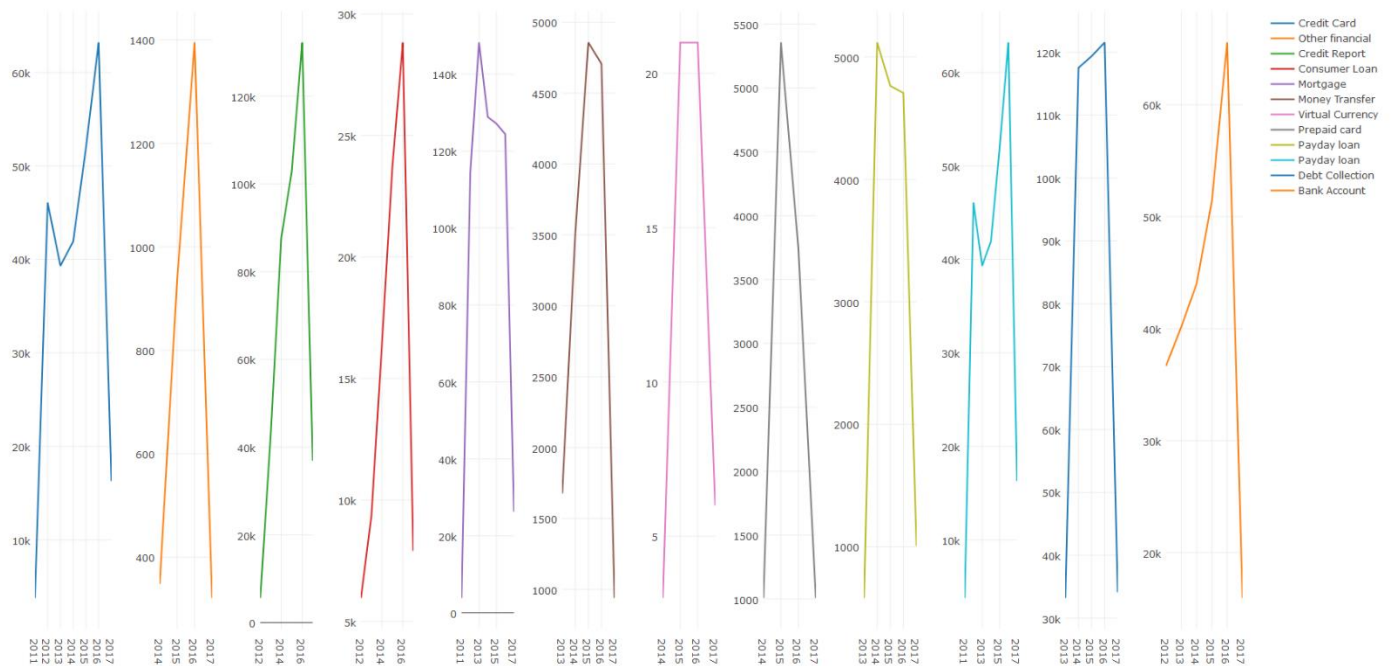
Chart showing Mortgage complaint volume for different states

The chart displayed below helps understand the demographics of the complaint volume for the Mortgage product for different companies from the data collected. As displayed from the chart, it can be concluded that mortgage complaints peak in California and it's not wise to invest in mortgage related products in California.

2013-2017 US Mortgage Complaints by State



Graphical representation of change in Volume of complaints for all products over time



Conclusion

The above project helped us gain wonderful knowledge about the aftersales services and complaint handling statistics for different companies. It also helped us analyze data of humongous proportions and work on coming up with statistics that would help a financial company improve its products and help a customer choose his/her company and product based on statistics and numbers than on impulse.

References

<https://plot.ly/>

<https://docs.mongodb.com/manual/>

<http://stackoverflow.com/>

<https://www.consumerfinance.gov/>