H-1B Visa Dataset Analysis

Spring 2017

CMPE 239

Project Report

Under guidance of

Prof. Chandrasekar Vuppalapati

Team Members

Mounika Batthina

Prasanth Ramineni

Smitha Muthuvenkataramani

Vaishnavi Reddipalli

ABSTRACT

***Abstract*—In this report, we discuss the strategies and technologies used for analyzing the H-1B visa dataset available through the 'Foreign Labor Certification Office'. In this project, we perform the analysis of the most popular work visa deriving conclusions such as, which employer sponsor a highest number of visas, what is the geographical advantage when applying for a visa, what is the average salary range for a given position and other meaningful insights from the dataset available. We intend to design and develop a knowledge extraction system which can predict meaningful conclusions. The various algorithms used to arrive at the conclusion and findings are discussed in this report.**

Keywords—Apriori, Random Forest, Linear Regression, SVM, Logistic regression, H1B data, R programming language.

Table of Contents

Chapter 1: Introduction

The H-1B is an employment-based, nonimmigrant visa category for foreign workers to work in the United States. Every year, the US immigration department receives over 200,000 petitions and selects 85,000 applications through a lottery process. With new H1B rules being enforced by the new government, non-immigrant students are not sure how things will turn out when they graduate and the competition is tremendous. It would be better for everyone if there are an analysis and insights for this process which can help students looking for work opportunities in the US to choose a better employer.

Based on the datasets available on the Foreign Labor Certification website (www.foreignlaborcert.doleta.gov), we plan to mine the datasets to derive meaningful insights and conclusions. The purpose of the system is to predict the queries and provide visual information of the data using histograms, graphs, and pie charts. The dataset on H-1B visa is structured data set and consists of multiple data items. Data classification is achieved by dividing the data sets based on case status, visa class, employer name, employer city, job title and wage rate. The system should predict and recommend the geographical location and highest wage rate gave the details of a job title. Data association is achieved by the job title, wage rate, and employer name.

Analysis of dataset is performed using various data mining algorithms such as ‘Logistic Regression algorithm’, ‘k-means algorithm’, ‘Apriori algorithm’, ‘Support vector machine algorithm’, ‘Naive Bayes’ and other classification and association algorithms for deriving accurate data insights and compare the end results using the test data to provide a visual representation of the data.

In the next section, let’s look closely at the implementation strategies for different algorithms.

Chapter 2: Algorithms

A. Logistic Regression Algorithm

Classification of the data is required to get the insights on the status of the visa based on different employers. By modeling the available history of H1B data the probability of visa being accepted or rejected can be predicted with respect to the employer, wage range, geographic location of the employer and title of the job. A regression model should be built where the dependent variable is categorized to make predictions and to dig a little deeper information.

Logistic regression is a go-to statistical technique for classification problems used in machine learning. Logistic regression can be implemented for binomial/binary, ordinal/ ordered or multinomial outcomes. A logistic function or sigmoid function is used to build algorithm which is defined as

$$g(z) = 1/ (1 + e^{-z})$$

In the above equation, z is a real number that should be transformed and e is the base of the natural logarithm. This function gives an S-shaped curve plot which transforms any input real number to a value between 1 and 0 but not including 0 and 1. Input values to the algorithm by combining them linearly using coefficients or weights for predicting the output outcome value.

Preprocess the data first by scaling the attribute range as (0,1). For example, to model the visa case status as 'CERTIFIED' or 'DENIED' from the employer attribute, the first class could be 'CERTIFIED' and the model could be written as the 'CERTIFIED' probability given employer attributes as:

$$P(X)=P (\text{Case Status=CERTIFIED}|\text{employer})$$

B. K-means Algorithm

Data clustering is required to identify the data sets which lay closer to each other so we can identify items that are identical to each other. Clustering can be applied in many ways for the H-1B dataset. For example, if we want to cluster jobs of the single category to determine the average wage, we need to implement clustering strategy. If we want to cluster employers of the same geographical region and determine the most popular job title, we need to implement clustering.

K-means is one of the most popular clustering techniques used for data analysis. A set of data points, *x1,... xn* and number of desired clusters *K* are determined as inputs for the algorithm. The clusters are assigned an initial value and the data sets are assigned to the closest cluster and a new centroid is calculated for each cluster. This process is repeated until a convergence criterion is satisfied.

C. Apriori Algorithm

Association rules play an important role in any data analysis problem. With the help of association rules, we can find interesting relations between data attributes in our H1B dataset.

It is important to analyze our data attributes in depth. This helps to catch insights better. Also, we can find data attributes that contribute to the result of an H1B application. Thus, using association rule mining we can find the factors that influence the final decision of an H1B application.

The association rules help us to generate frequent itemsets and rules. For this purpose, we will make use of three measures.

1) Support

This measure helps to gauge the popularity of an itemset. That is the frequency of an itemset in the dataset. Thus, we can find the support of high PREVAILING_WAGE with a CASE_STATUS as certified.

2) Confidence

It measures the strength of an association. It determines how frequently an item2 occurs in a transaction that contains item1. Here, we can translate it to how frequently a CASE_STATUS of certified occurs in an H1B application with high PREVAILING_WAGE.

3) Lift

It gives us a measure of the importance of a rule. The lift can help in deciding whether an association rule is valid and helpful for the problem specified.

Apriori algorithm is used to generate the frequent itemsets. This essentially means getting itemsets with support greater than a specified minimum support. This is done using the basic assumption that a subset of a frequent itemset must also be a frequent itemset. And then we can

use these frequent itemsets to derive association rules.

D. Naive Bayes Algorithm

A supervised and statistical learning methodology is required for data sets like H1B applications.There are many parameters which are independent and classifying when the dimensionality of input data set is high should be considered as a key point in this project. Once a probabilistic approach is given to a data set, coming up with many insights is a simple procedure using different available algorithms. A fast, simple and sophisticated algorithm for a small amount of training data set helps in determining useful insights for recommendation system.

Naive Bayes Algorithm is a classification with supervised learning and statistical method. It assumes a hidden probabilistic standard and acquires uncertainty on the model in an ordered manner by regulating probabilities of results. This algorithm can be used to unfold predictive and diagnostic problems.

It can take an arbitrary number of independent attributed whether categorical or continuous.This gives a precedent knowledge and practical learning algorithm when noticed data is combined. Classification from this algorithm provides a deep insight for evaluating and understanding various learning algorithms.In addition to this, it determines explicit probabilities for assumptions and is robust to input data noise.

Recommendation system uses data mining and machine learning techniques to refine unseen insights and can foresee if a user likes a given resource or a given information will be helpful for the user in future decisions. Naive Bayes algorithm states a particular hybrid recommendation method by merging collaborative filtering and bayes classification method.Results prove that this is better performance, scalable and more coverage on data set at a time and so wipe out traditional recommendation systems.

This algorithm is based on Bayes theorem, which is important for calculating conditional probabilities. This is used for determining posterior probabilities for given data set.

$$P(\mathrm{h}|\mathrm{D}) = P(\mathrm{D}|\mathrm{h})P(\mathrm{h})/P(\mathrm{D}) \quad \cap \mathrm{A}) \,/\, \mathrm{P}(\mathrm{A})$$

D is a set of tuples.Each tuple is a ‘m’ dimensional attribute vector whereas P(D) is the prior

probability of training data D,P(h) is the prior probability of hypothesis h, P(h|D) is the probability of h given D and P(D|h) is the probability of D given h.

Naive Bayes algorithm assumes each attribute is independent and determines prior probability. Let 'n' number of classes be C1,C2,C3...Cn and X: (x1,x2,x3,....xn). This algorithm predicts if X classifies into class Ci if following conditions are satisfied

- P(Cj/X) < P(Ci/X) where 1<=j <=n, j<>i maximum posterior hypothesis
- $P(C_i/X) = P(X/C_i)\ P(C_i)/P(X)$
- $P(X/C_i)\ P(C_i)$ Maximize since P(X) is constant

With class, conditional independence assumption of Naive Bayes algorithm, we can eliminate cost of computing P(X/Ci)

$$P(\mathrm{X} / \mathrm{Ci}) = \prod_{k=1}^{m} P(xi/Ci)$$

P(X/Ci) = P(x1/Ci)* P(x2/Ci)*.... *P(xn / Ci)

E. Support Vector Machine

A Support Vector Machine(SVM) is a classifier that defines the criterion to be selecting a decision surface that is maximally away from any data point, margin of the classifier.

An SVM classifier insists on a large margin around the decision boundary. A classifier with large margins does not make low certainty classification decision. Thus, minimizing the misclassification error. A SVM classifier is of the form

y = {+1 for one class and -1 for the other class}

For this model, we can choose a combination of data attributes and expect results showing some low misclassification rate.

Chapter 3: System Architecture and Data Flow

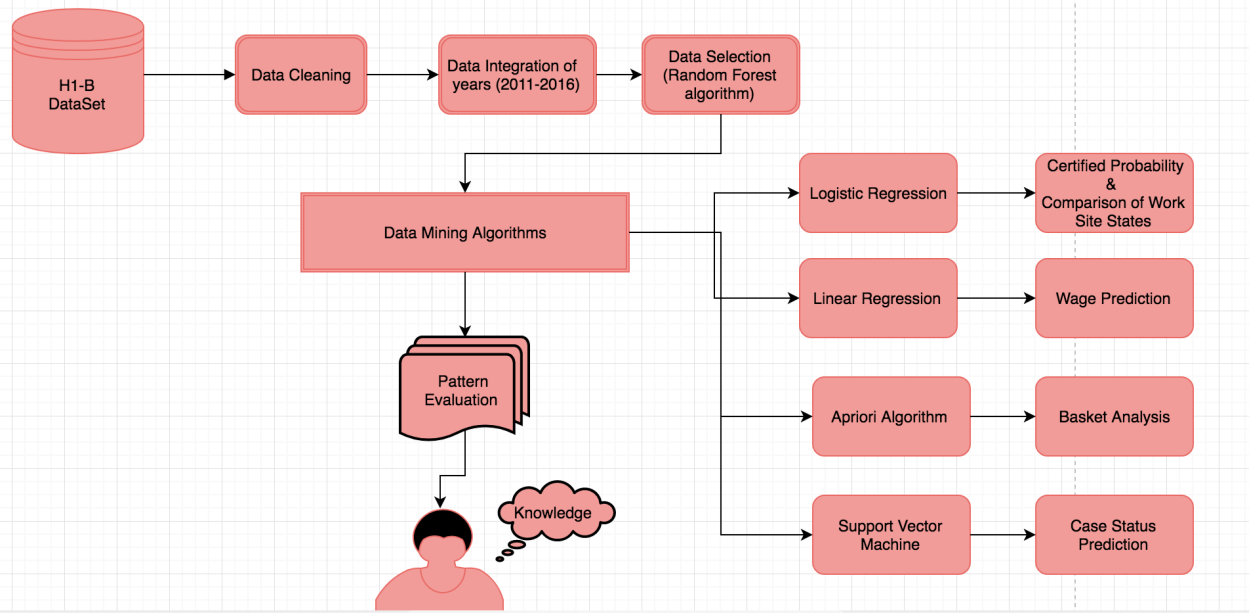


Fig 1: System Architecture

The Detailed Architecture of our project is described in the above figure.

The H1B data from 2011 - 2016 years are extracted from Foreign Labor Certification website and integrated to a single rds file considering different column labels. Relevant attributes are selected by applying random forest algorithm. The integrated and cleansed data is used for applying various data mining algorithms such as Logistic regression, Linear Regression, Apriori and SVM. Upon recognizing the patterns in the data, a UI is designed using Shiny application to present the knowledge we discovered.

Chapter 4: Data Characteristics

The dataset from the Foreign Labor Certification website consists of 40 labeled columns. The data ranges from the financial year 2011 to 2016 across multiple visa categories including L1, H-2A, H-2B and PERM programs. Datasets are divided into each financial year and which has approximately 600 thousand rows of data which results in a total row of more than 5 million data. The column labels in each dataset are differently labeled and the first task is to rename all the columns to the same name.

The below figure shows the data model diagram and the list of files providing the data:

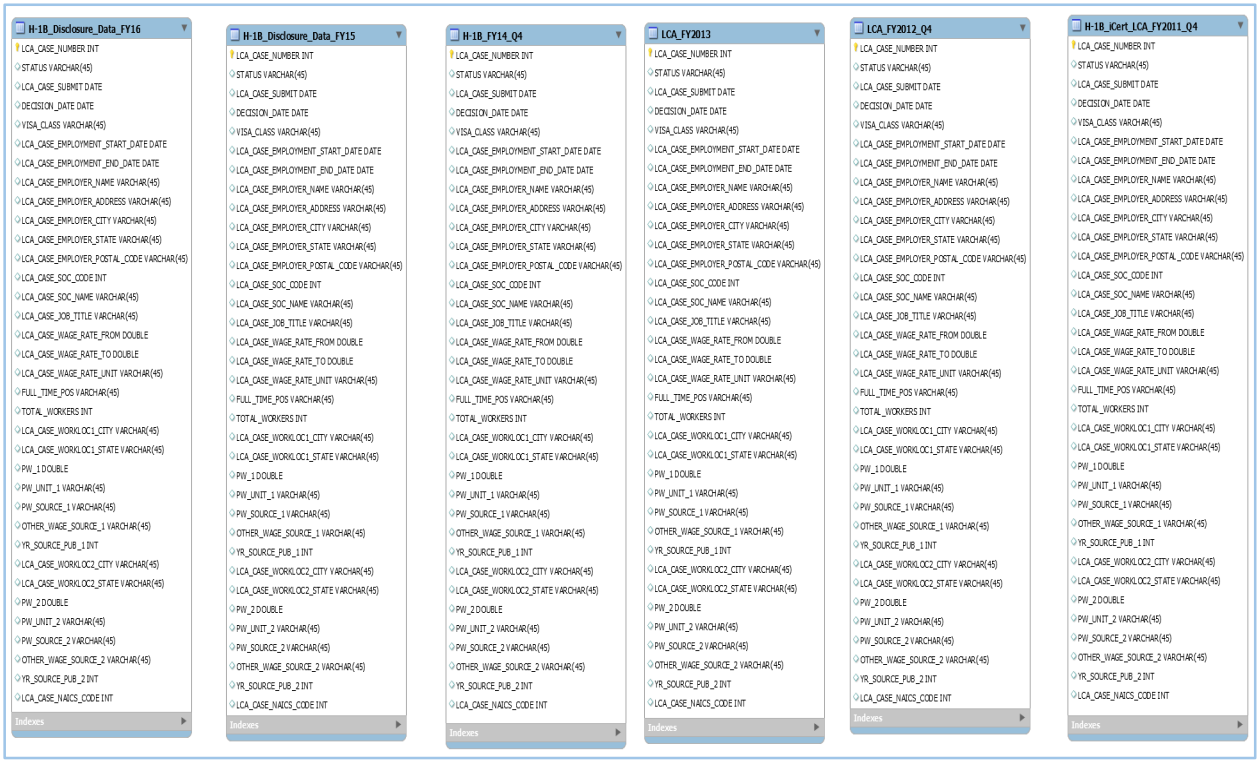


Fig 2: Data model diagram

As we can see from the above figure, there are over 40 labeled columns in total from the H1B dataset of the years 2011 to 2016. Thus, we had to do data integration and transformations before applying a variety of algorithms to derive insights. A brief description of the data wrangling and preprocessing would be explained in the next section. A snapshot of the transformed dataset on which algorithms were applied is depicted below:

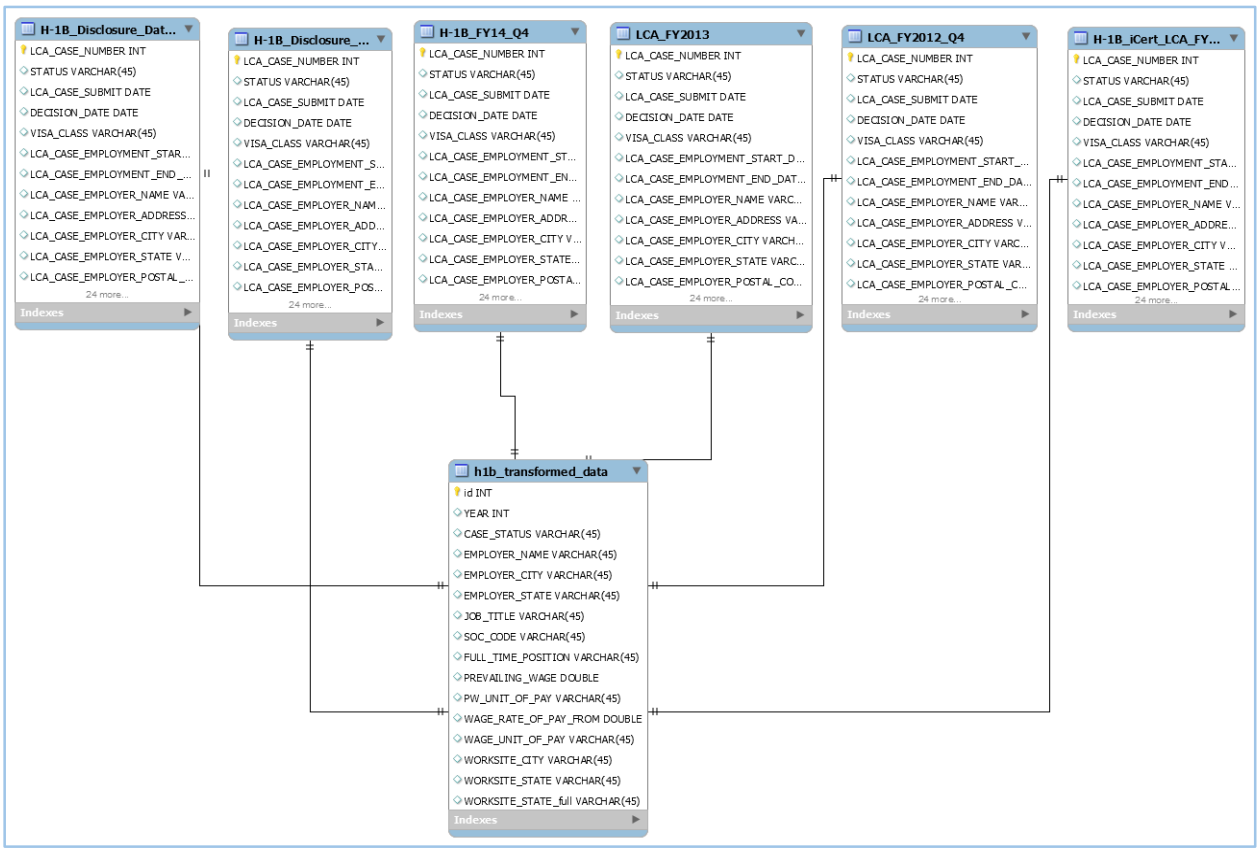


Fig 3: Integrated and transformed data

Data Integration and Wrangling

Since the foreign labor certification website provides the H1B datasets for a number of years starting from 2008, we have decided to consider the data from years 2011 to 2016.

Looking at the dataset we could identify that different years have different column names and depiction of data. Thus, we first integrate the data from 2011 to 2016 as a single dataset. This gives us around 3 million data tuples for data mining. This was obtained after omitting data attribute fields that weren't common to all datasets. After this data transformations were performed to make the data consistent and noise free. We have created two types of transformed datasets. One with NAs and the other without NAs. This is because certain algorithms like Random Forest require the dataset to be free of NAs.

Some of the transformations we performed were, converting the Wage attributes of the dataset to Yearly scale. This was done from various other scaled like weekly, hourly, bi-weekly etc. Apart from this , the FULL_TIME_POSITION attribute was converted to Y/N where it was

NAs, based on the PREVAILING_WAGE attribute. Also, we have found the geocodes of the WORKSITE_STATE attributes, and added the latitude and longitude columns for every data tuple. This becomes helpful when we needed to show a heatmap of the top states based on CASE_STATUS attribute and the Google Marker Cluster API maps. Similarly, certain noise were removed from the dataset, apart from making spelling corrections and feature extractions.

Among the 40 labeled columns of data, we identified the below fields as the most relevant attributes which can influence the `CASE_STATUS` as CERTIFIED or WITHDRAWN.

CASE_STATUS: This column has four different factors 'CERTIFIED', 'DENIED', 'CERTIFIED-WITHDRAWN' and 'WITHDRAWN'. We intend to give integer value to each of these factors from 0… 3.

VISA_CLASS: The major factor in this column is 'H-1B', but, it also includes 'E-3 Australian', 'H-1B1 Singapore', 'H-1B Chile'. For, the simplicity of the model, we consider all these visa types into a single factor.

EMPLOYER_NAME: This column gives the employer name which can be used for clustering employers from different locations applying for H-1B visa.

WORKSITE_STATE: This column gives the intended location of work of an employee post H-1B which would help in determining the geographical location from which most numbers of applications are originated.

PREVAILING_WAGE: This field helps in determining the minimum wage for a given job title and the can help in comparing the wages for a similar job title from a different employee.

YEAR: The year in which H-1B visa is filed.

Chapter 5: Analysis

To reduce the noise in the data, we have implemented spell checkers and corrected the required datasets, renamed wrongly labeled columns. Some datasets have the PREVAILING_WAGE data in hourly basis and some on a yearly basis. We need to convert the

hourly rate data into yearly basis data.

After the data sets are corrected and ready for analysis, we intend to show the below case studies.

1. Number of applications based on each employer and job title.
2. Case status for each application.
3. Wage rate comparison for two states and job title.
4. Denied percentage based on wage rate.
5. Wage prediction for future years.
6. Association rule mining for certified status.
7. Logistic regression algorithm to find probability of being certified.
8. SVM to predict case status based on prevailing wage.
9. Random forest to find influential attribute.
10. Map used to cluster applications based on job title.

Chapter 6: Findings

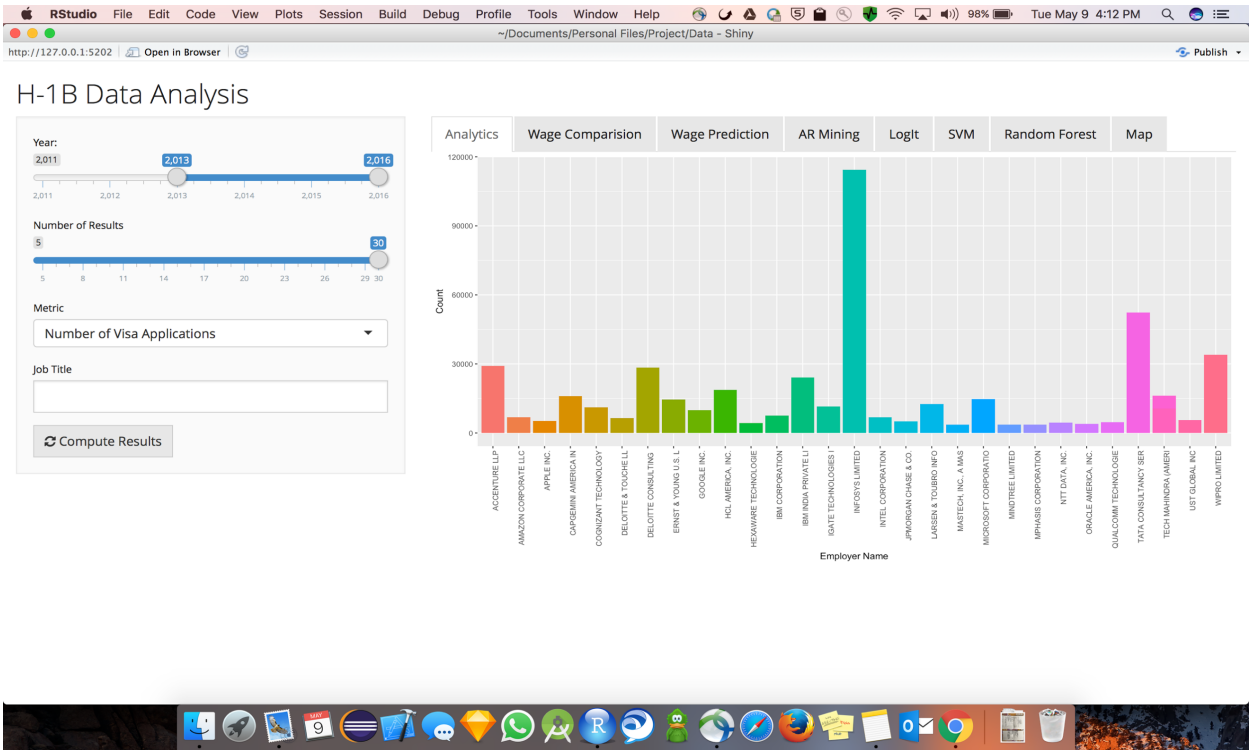
Analysis is done by implementing Apriori Algorithm, Random forest, logistic regression and SVM. Below are the few details of Shiny Application UI and Google map clustering API screen captures.

Analytics

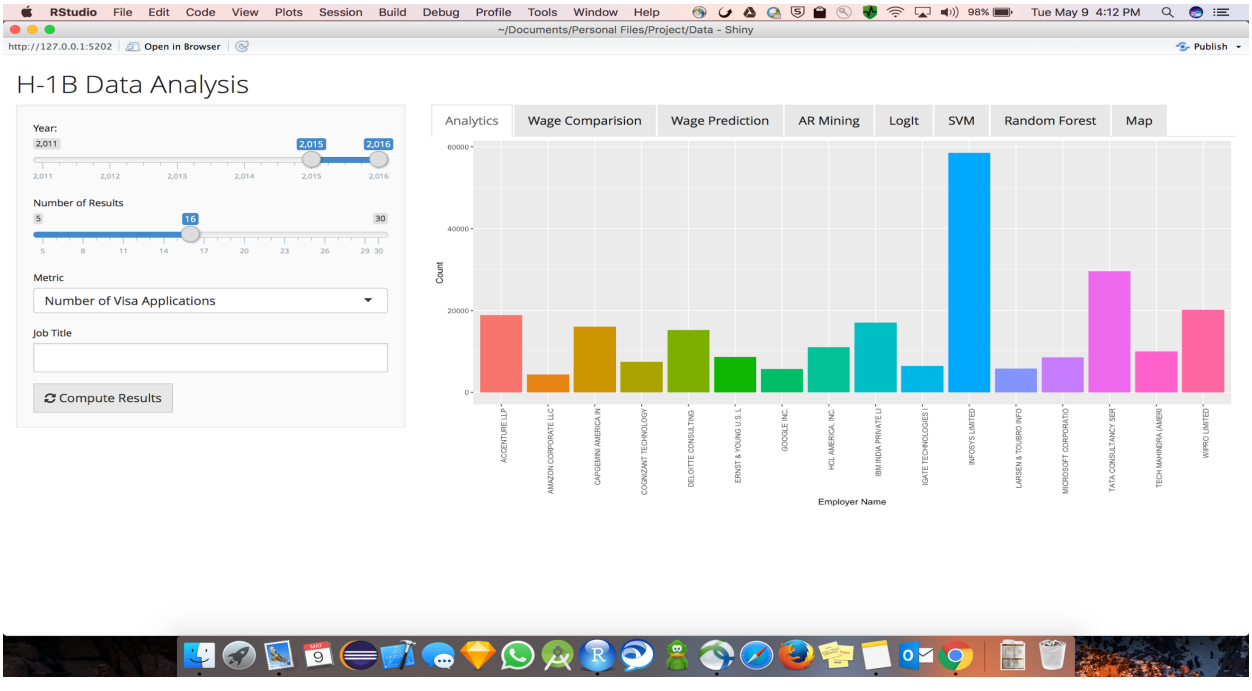
User can select the Year range and the number of results to be retrieved using the sliders provided. User should select any of the four metrics as Number of Visa Applications, Case Status, Case Denied and Denied Wage Rate.

1.Number of Visa Applications:

A Plot showing top Employer details is presented based on the User input Year and Number of top results selected. We could see the top 30 Employer names in the below plot in 2013-2016 year range who has maximum H1B applications.

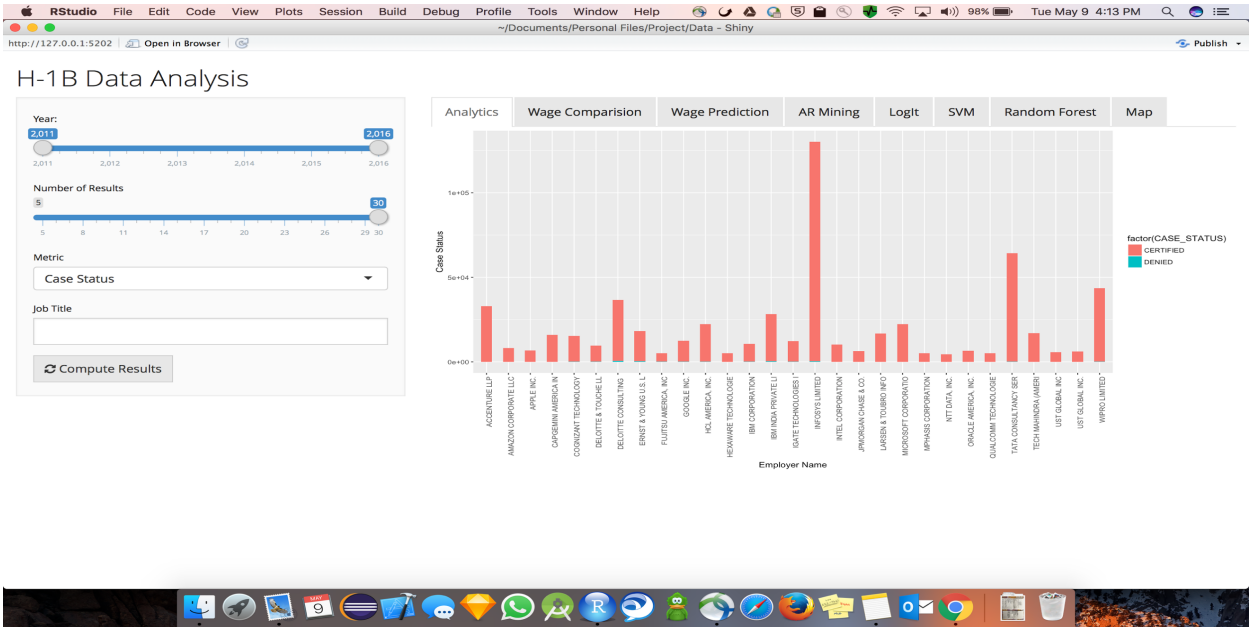


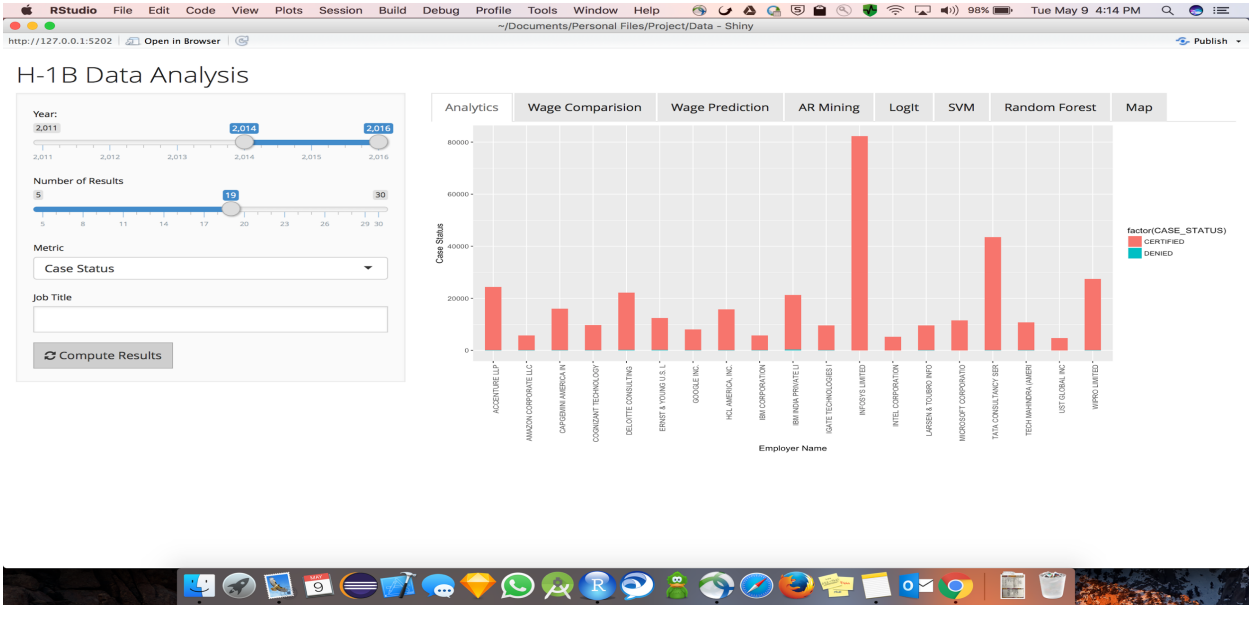
Below plot shows the top 16 Employer names in the year range of 2015 - 2016.



2. Case Status:

User can select Case Status Metric to compare the CERTIFIED and DENIED case statuses for top 30 H1B applicant employers in any year range between 2011 - 2016. This Analytics provides the user to get an idea about which employer has lesser denial rate compared to other top applicant employers.

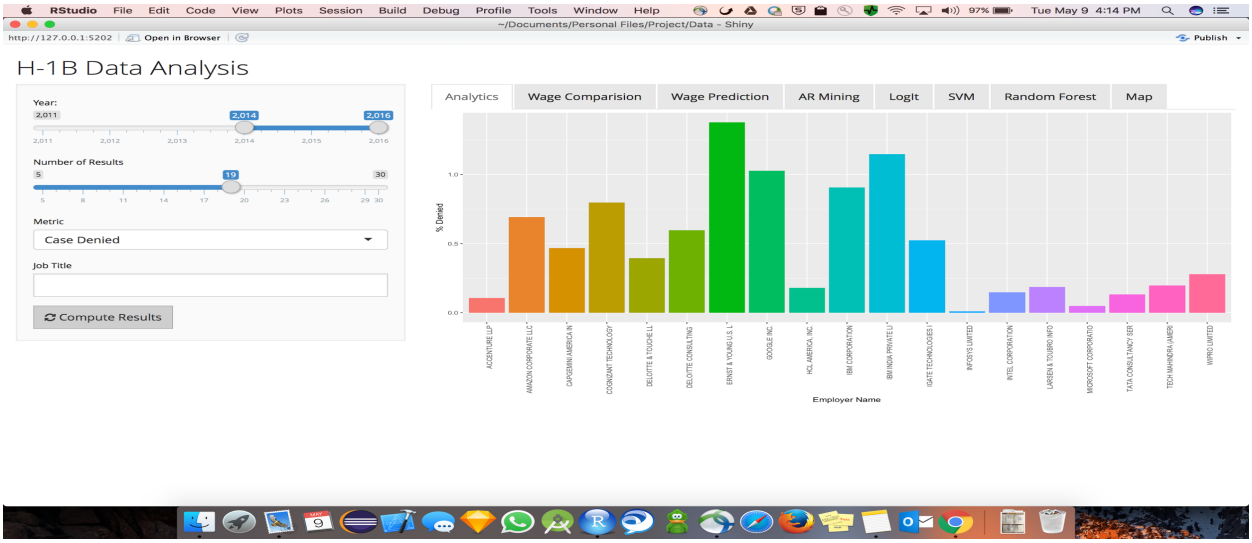




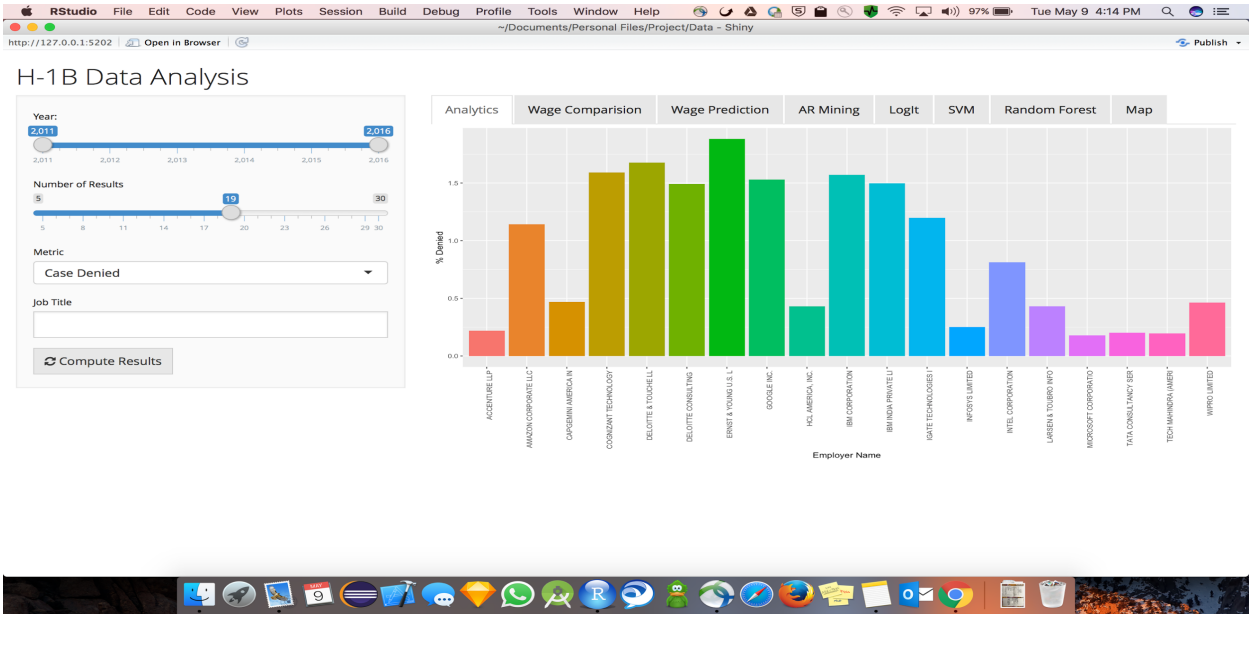
3. Case Denied:

User can select the Case Denied Metrics to compare the denied applications count for the year range 2011 - 2016. This analytical visualization helps user to check the Employer details who has more applications denied.

Below Plot shows the top 19 Employers in the year range of 2014 - 2016 who has maximum applications denied.

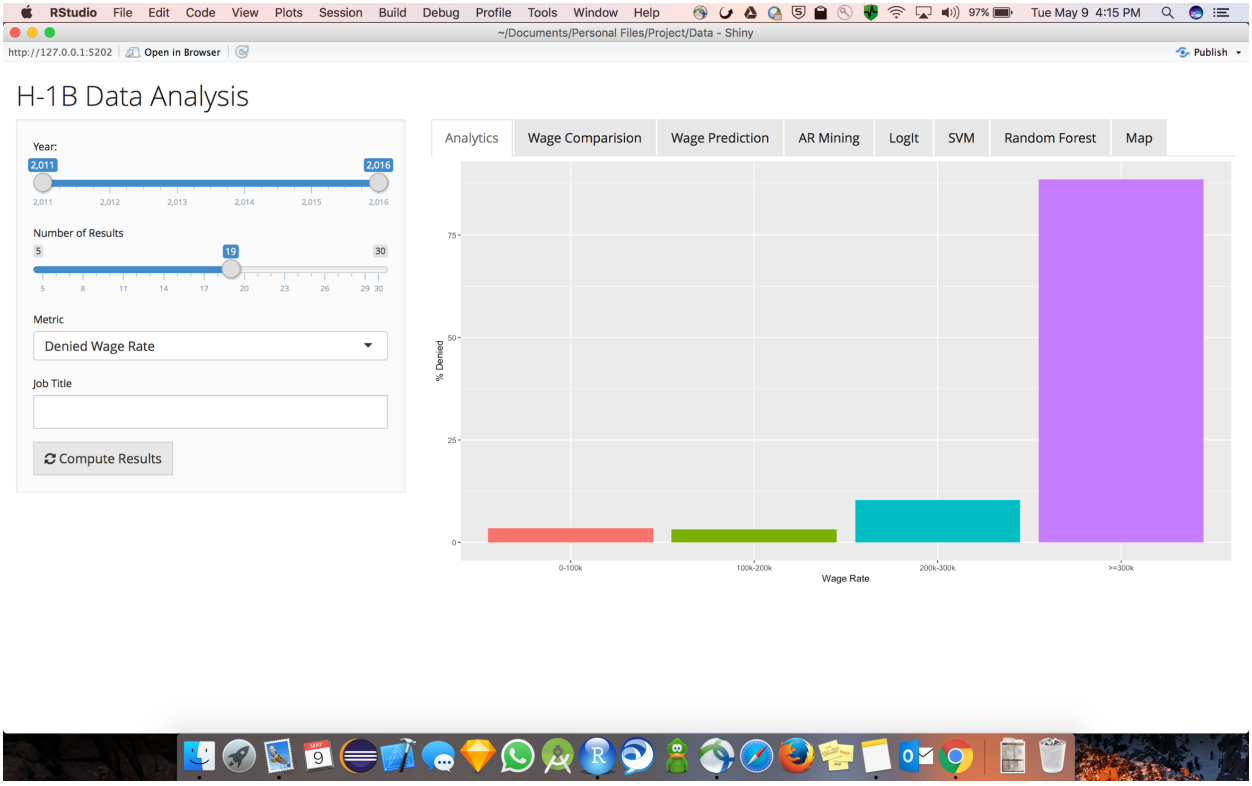


Below Plot shows the top 19 Employers in the year range of 2011 - 2016 who has maximum applications denied.



4. Denied Wage Rate:

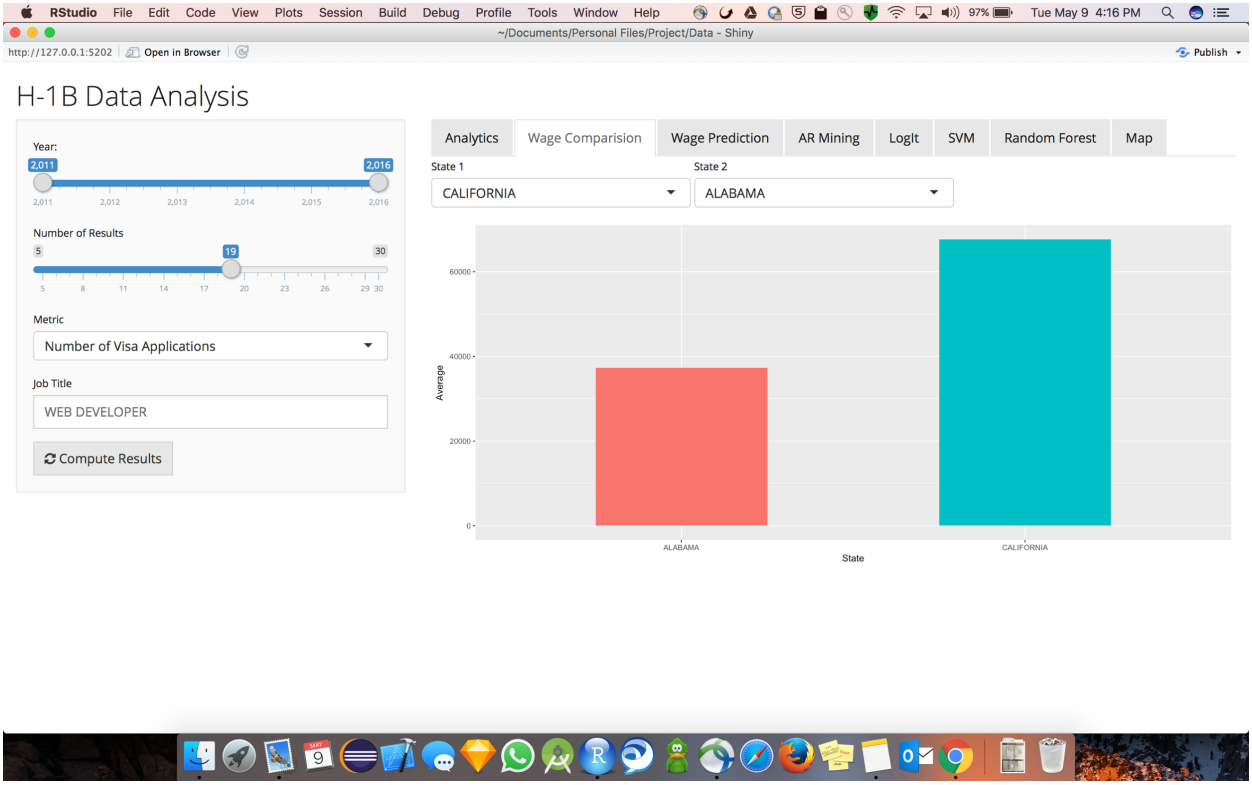
Denied Wage Rate Metric visualizes the denied percentage based on the prevailing wage rate of the employees. The Prevailing wage rate is divided into four chunks for comparison as 0-100k, 100-200k, 200-300k and above 300k. As the Prevailing wage rate increases the application denied percentage is increasing.



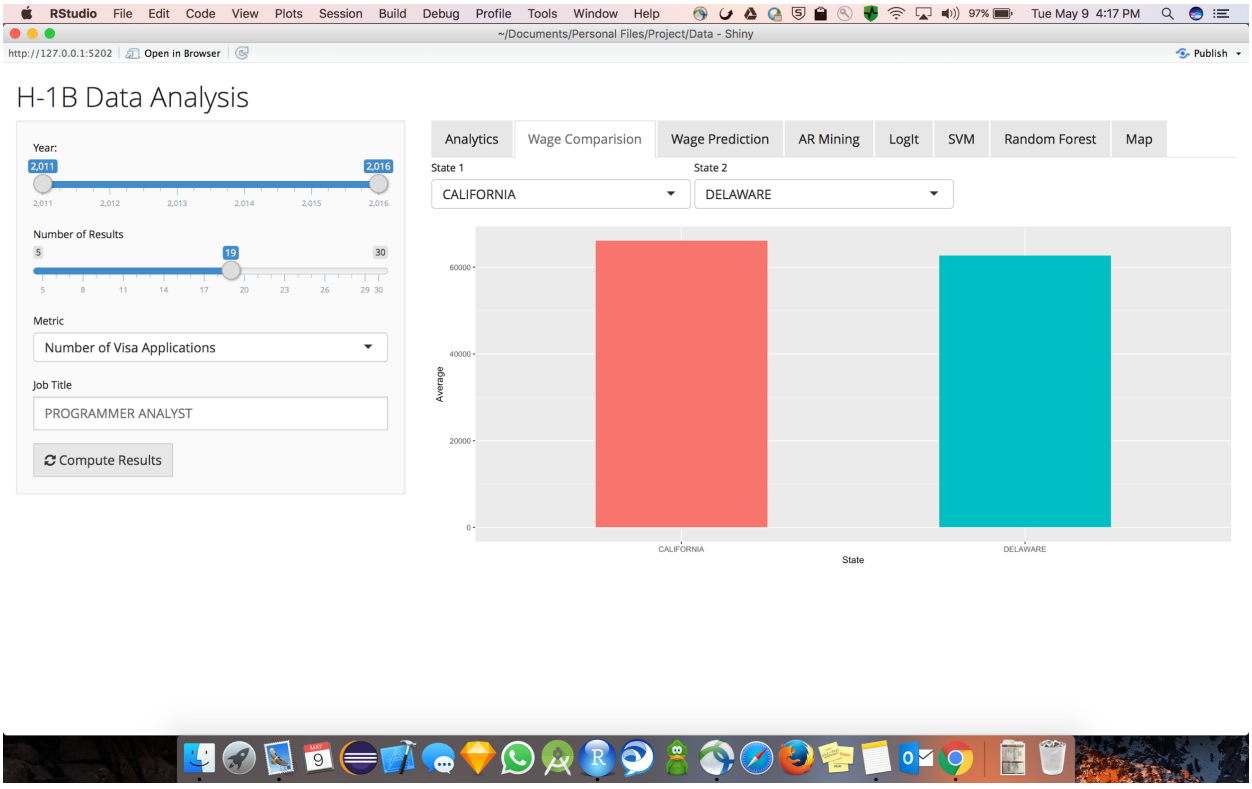
Wage Comparison

These Visual analytics helps user to compare the prevailing wage rate for a Job title in two different Work Site States.

Below Plot represents the wage rate comparison for California and Alabama States for a Web Developer position.



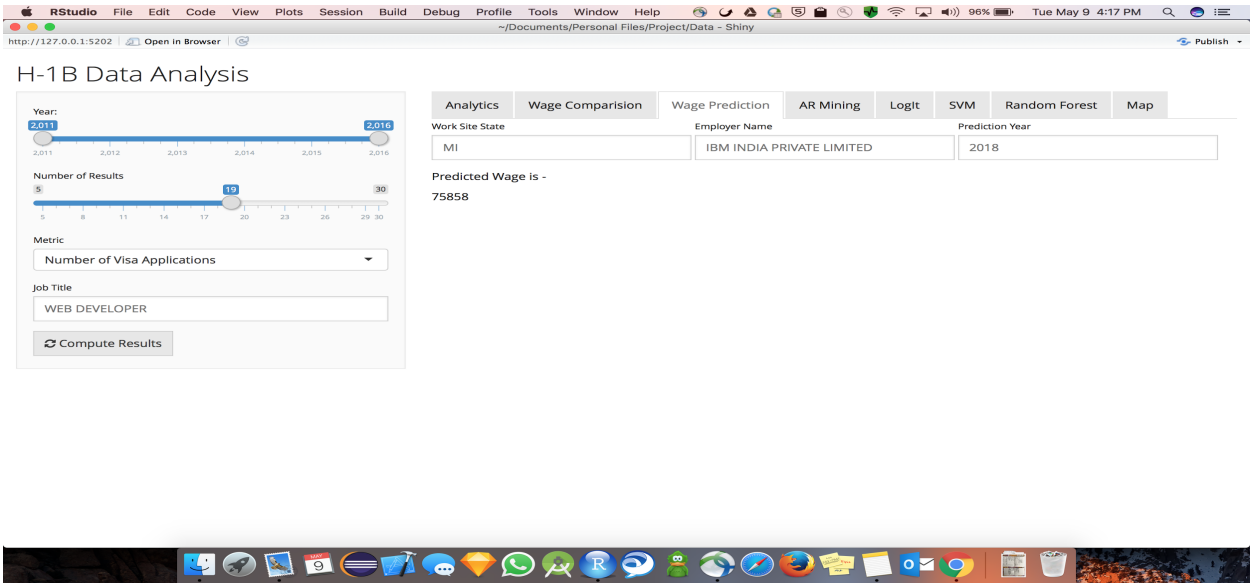
Below Plot represents the wage rate comparison for California and Delaware States for a Programmer Analyst position.



Wage Prediction

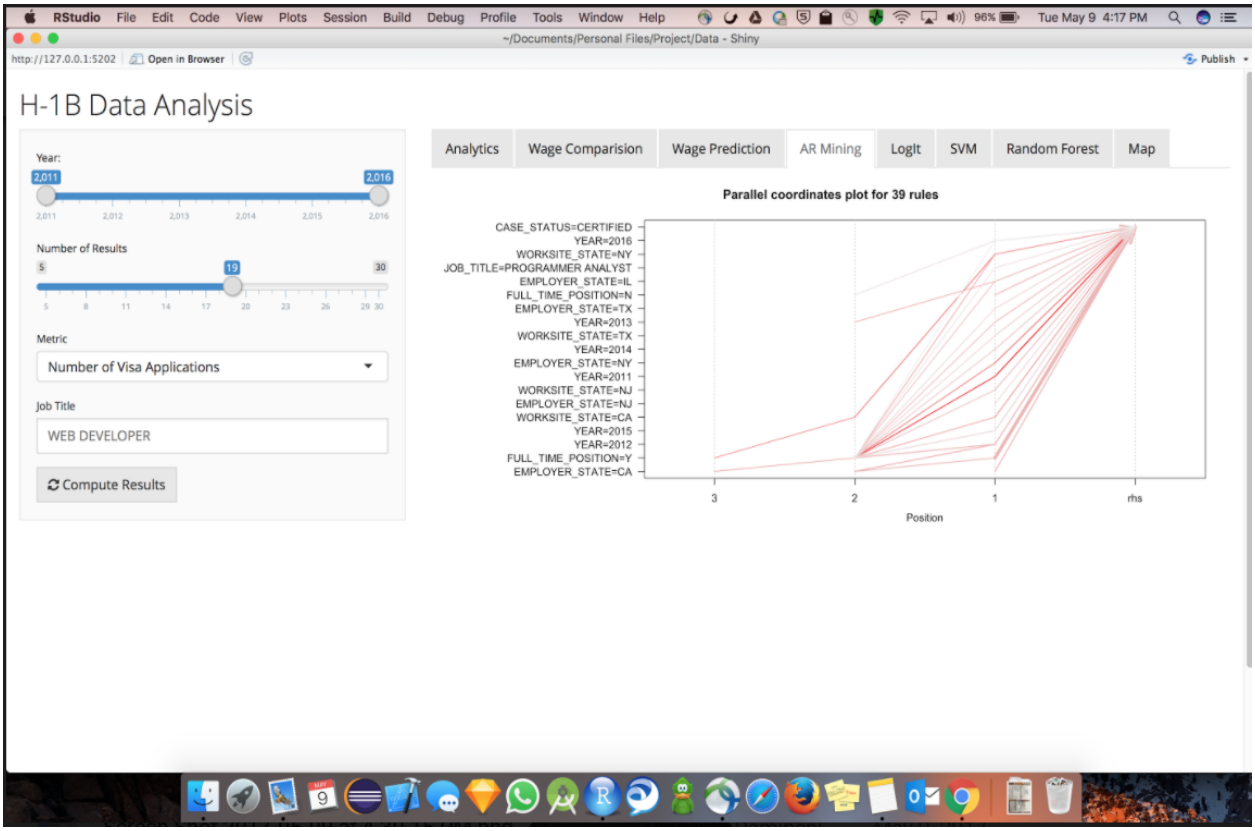
Linear regression algorithm has been used to predict the wage for the upcoming years for the respective Job title, Work Site and Employer name. User can navigate to Wage Prediction tab to input the details. For predicting the future wages, mean wage rate for each year is considered in our algorithm.

Constraint: Respective Job Title should be offered by the Employer in the given Work Site State. A minimum for previous two years of corresponding data should be available for predicting next wage rate.



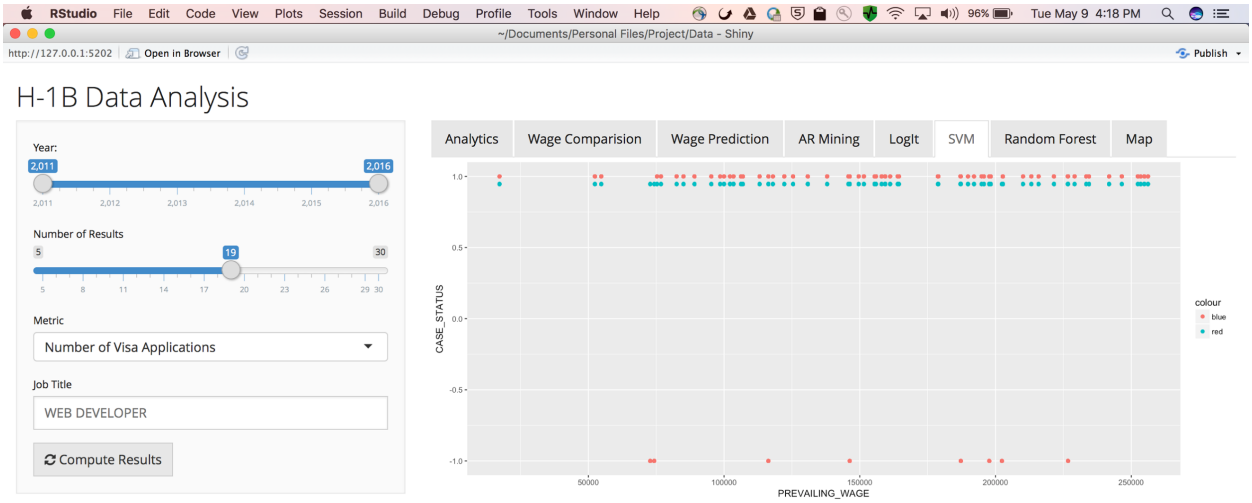
AR Mining

Below snapshot shows the Association rules between different attributes to get the application status certified.



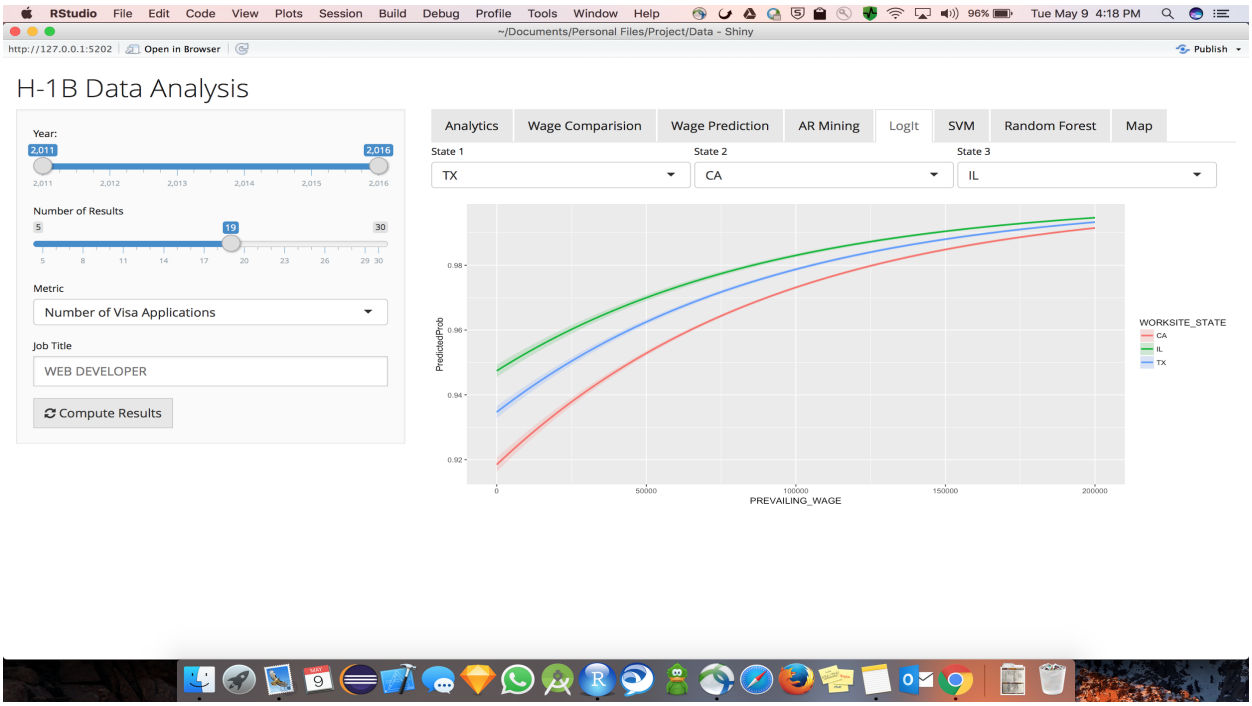
SVM

Below is the Support vector machine algorithm prediction screenshot. As there are not much data available on denied cases for training, model predicted almost every case to be certified.



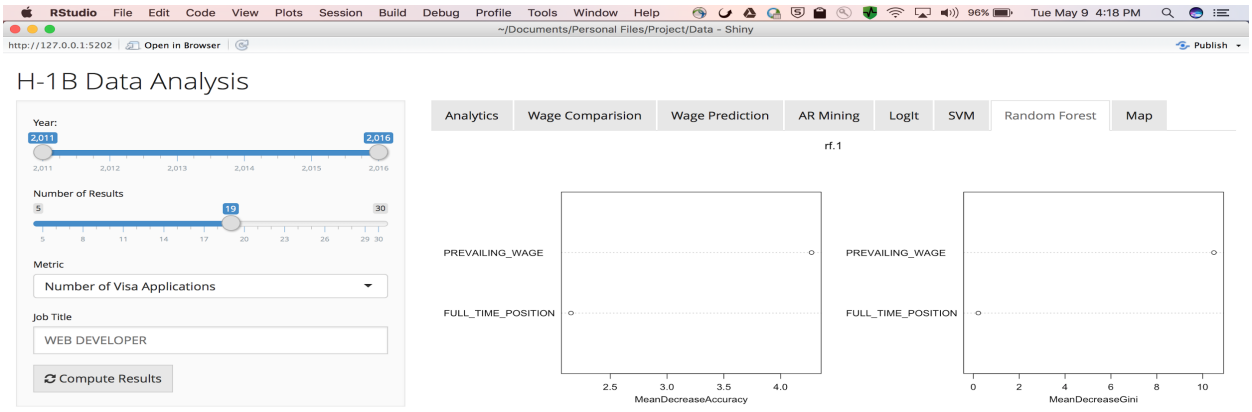
Logit (Logistic Regression)

The Probability of application being certified is identified using Logistic regression. The Entire data set from 2011 - 2016 years is used to build this model. In the Logit tab user can input three work site states for which he wants to compare the probability of application certified. Below graph shows a comparison between texas, California and Illinois for being certified.

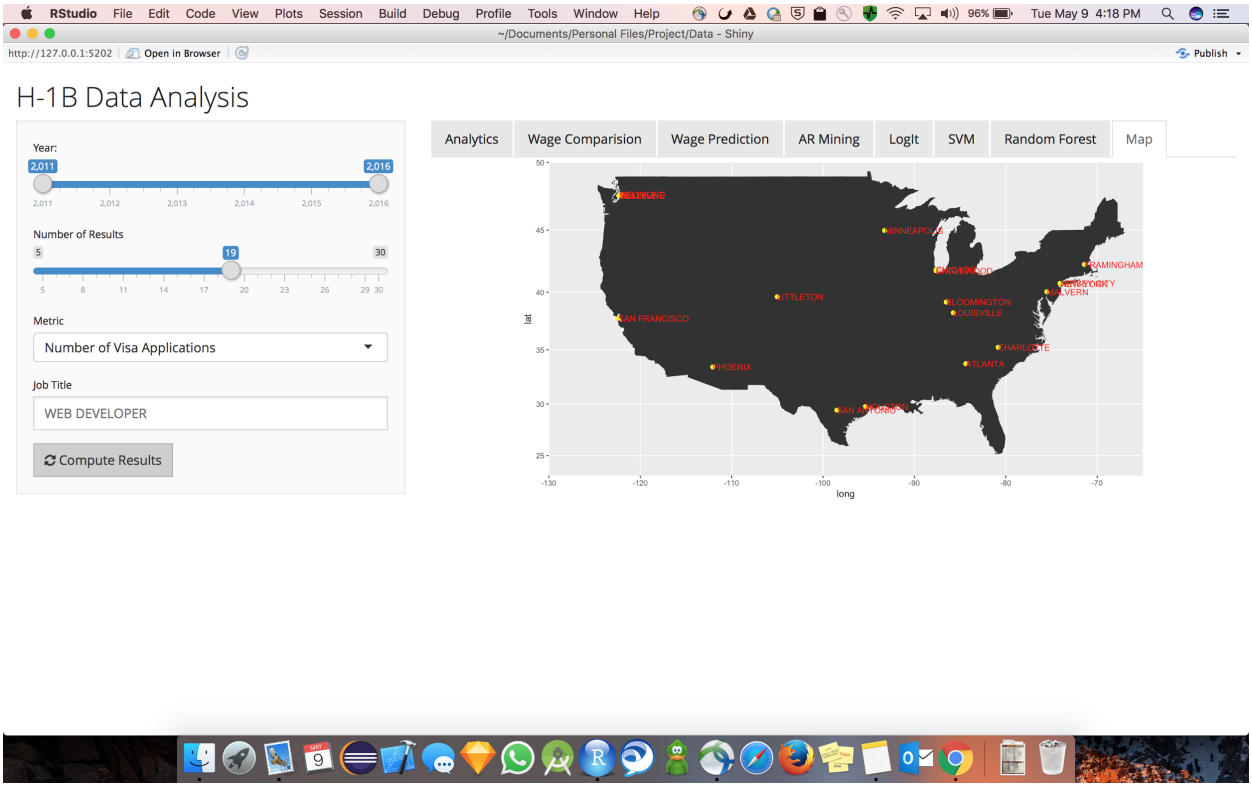


Random Forest

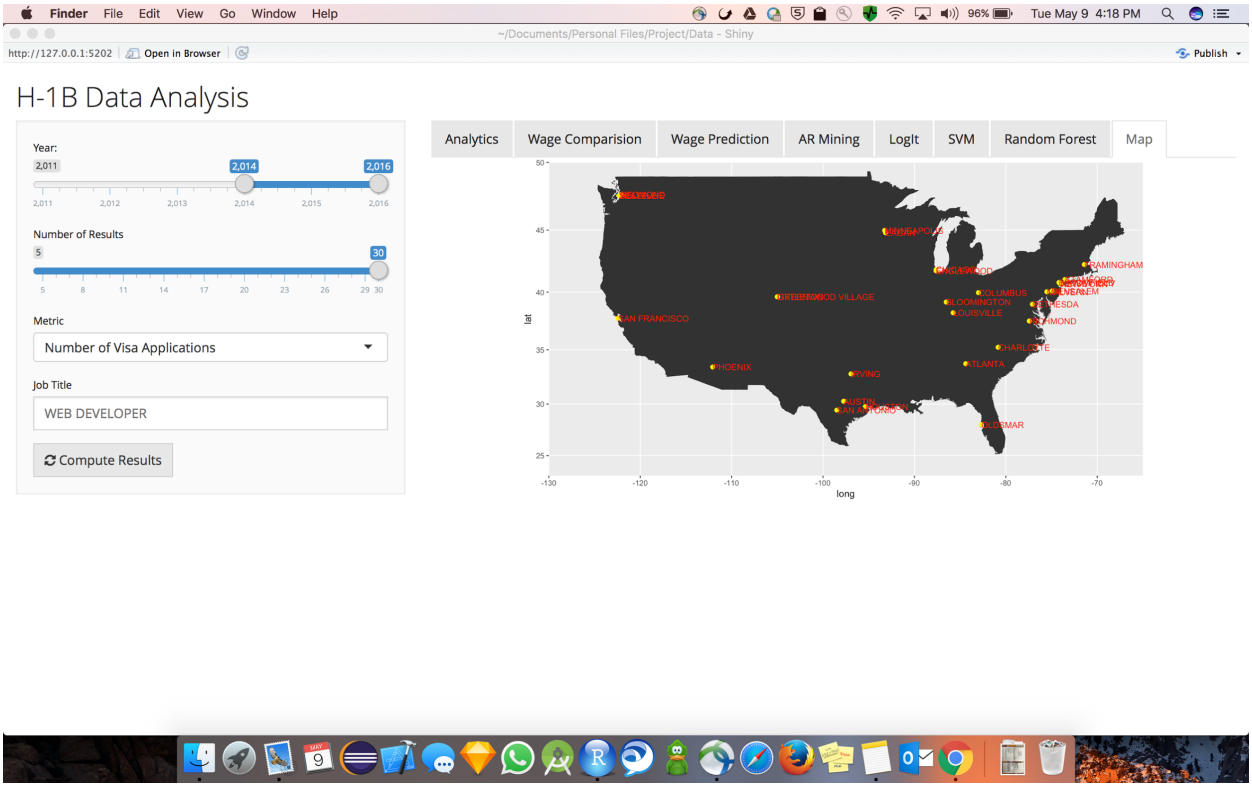
According to random forest algorithm prevailing wage and full time positions are the most influencing attributes for the case status. Below is the screenshot for this algorithm which clearly says prevailing wage is the label which can be used for any reference or pattern evaluation



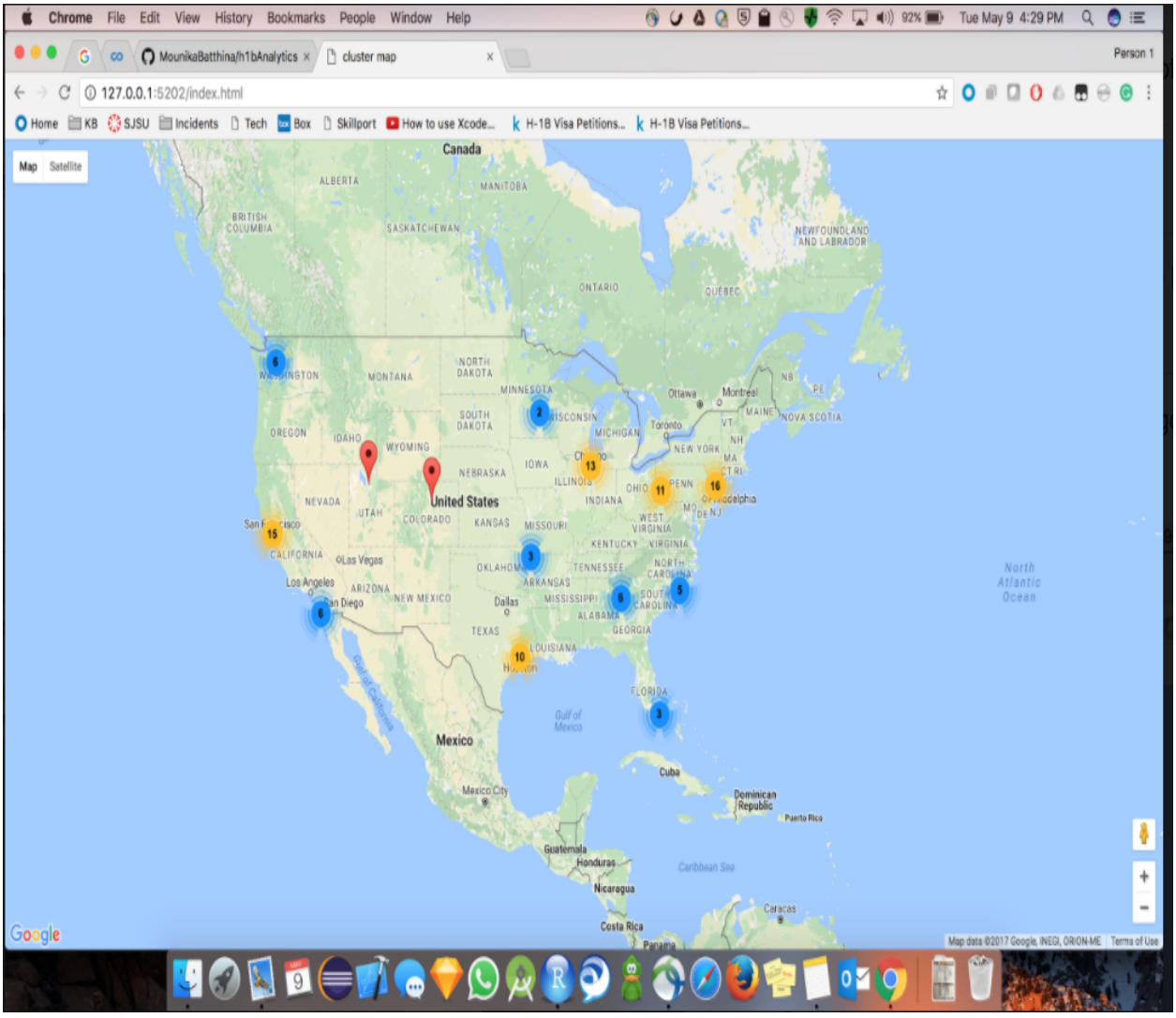
Screenshot below shows the heatmap of Top 19 cities for number of applications and the title "Web Developer" in USA map in years (2011-2016)



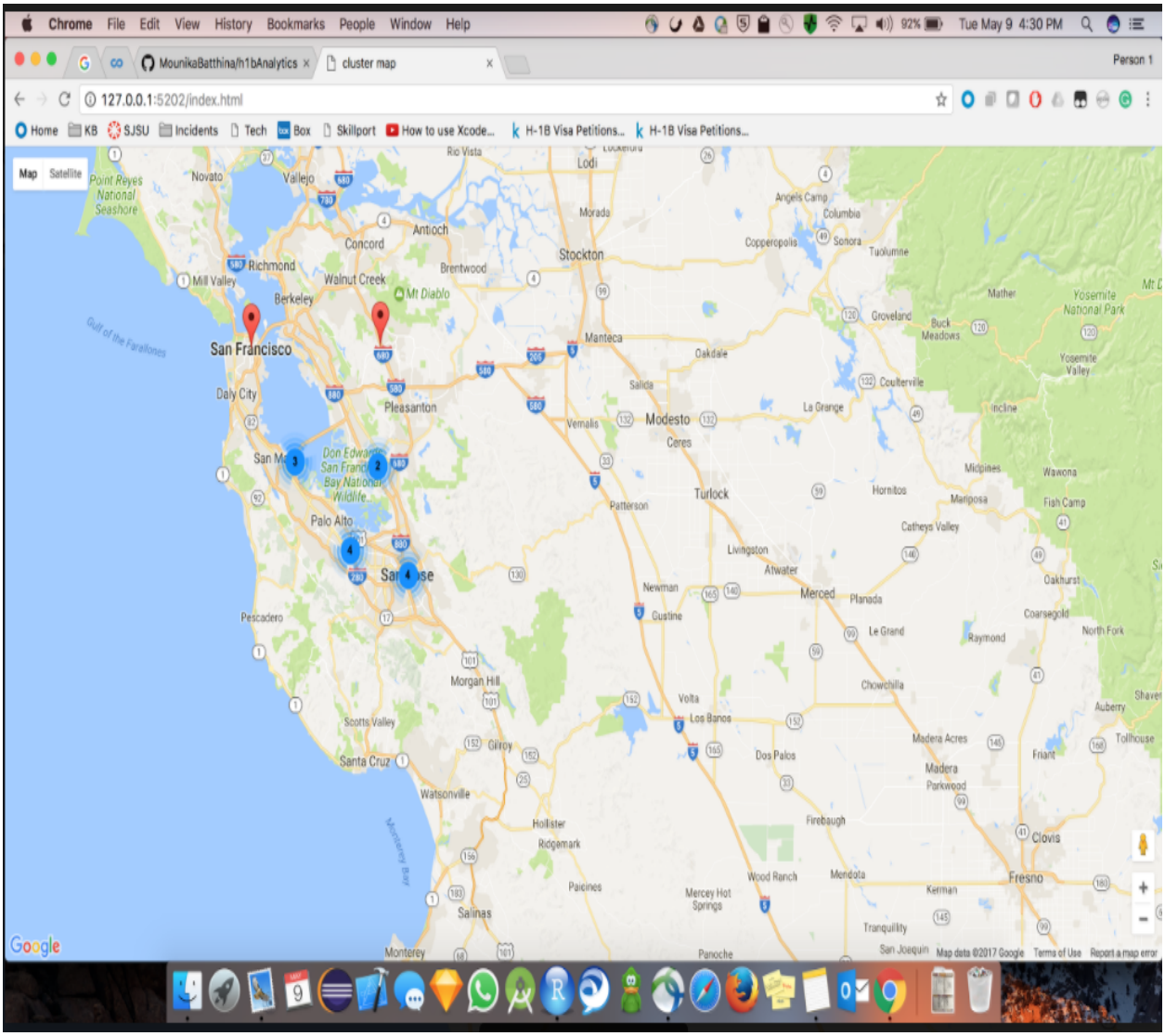
Screenshot below shows the heat map of Top 30 cities for number of applications and the title "Web Developer" in USA map in years (2014-2016)



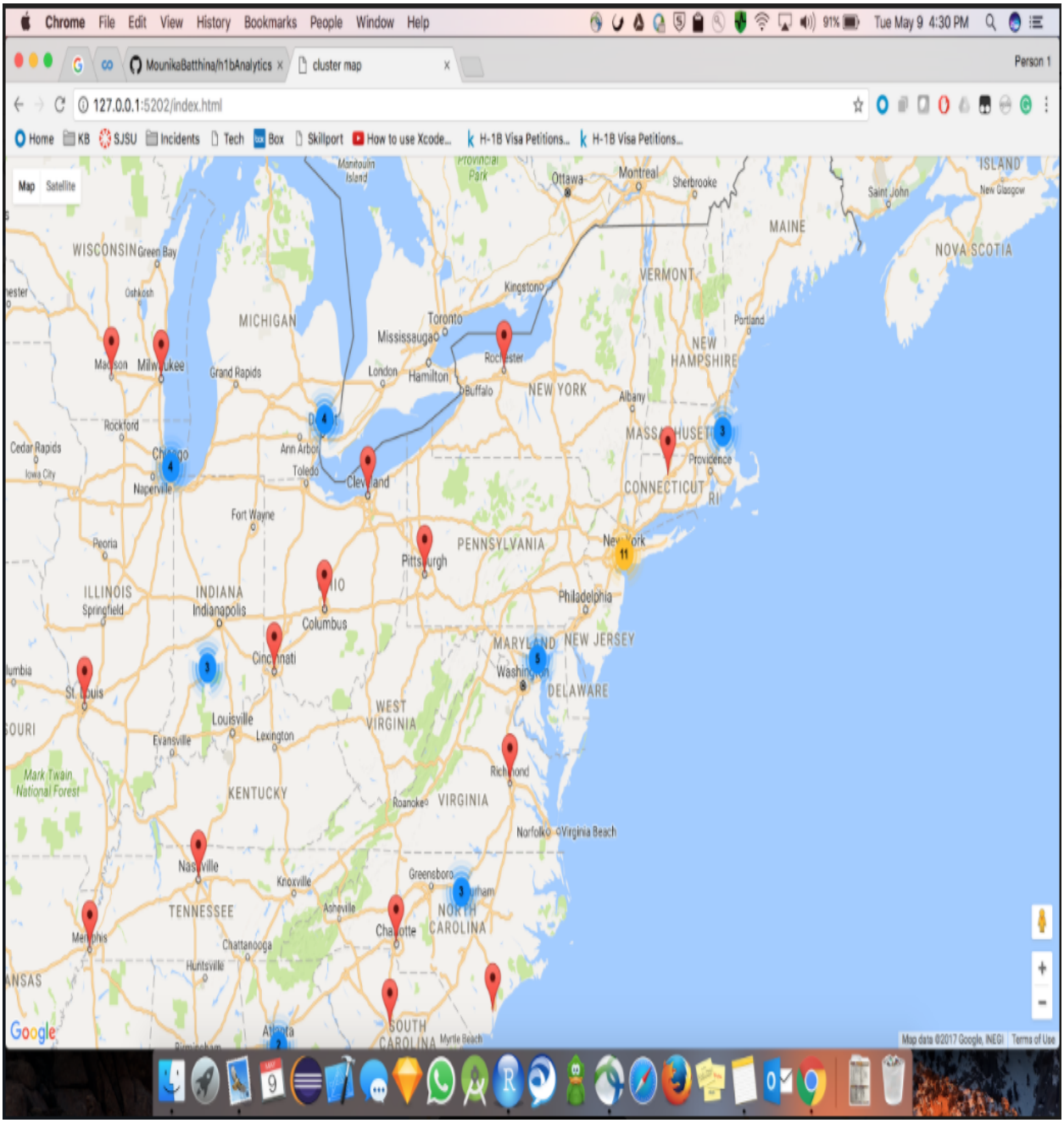
Screenshot below shows the clusters of “Web Developer” Job Title in USA map in years (2011-2016)



Screenshot below shows the clusters of “Web Developer” Job Title in years (2011-2016) when zoomed in to west coast of USA (Bay Area)



Screenshot below shows the clusters of “Web Developer” Job Title in years (2011-2016) when zoomed in to East coast of USA



Chapter 7 Code Snippet

Data Wrangling

```
#Function to calculate Annual prevailing wage
pw_unit_to_yearly <- function(prevailing_wage, pw_unit_of_pay) {
  return(ifelse(pw_unit_of_pay == "Year",
                prevailing_wage,
                ifelse(pw_unit_of_pay == "Hour",
                       2080*prevailing_wage,
                       ifelse(pw_unit_of_pay== "Week",
                              52*prevailing_wage,
                              ifelse(pw_unit_of_pay == "Month",
                                     12*prevailing_wage,
                                     26*prevailing_wage)))))
}
```

```
#-----------------------------------TRAINING MODEL-----------------------------------
# Apply linear regression to the training data set
lm<-lm(mean~YEAR,data=nmatrix)
summary(lm)

# Verify the fitted value and residuals for the train data linear model
nmatrixfitting <- data.frame(nmatrix , fitted.value= fitted (lm), residual= resid (lm))

# Verify the model fit - lwr and upr values
predict(lm,interval="confidence")

#-------------------------------------TEST DATA--------------------------------------
# Create a data from with the year for which the prevailing wage is to be predicted
newyear <- data.frame(YEAR = 2016)

# Predict the prevailing wage by applying the lm and update in the data frame
newyear$mean <- predict(lm,newyear,type = "response")
```

Logistic Regression Algorithm

```
#-----------------------------------------TEST DATA 2-----------------------------------
# Create new dataframe with the main data
# Prevailing wage range from random wages between min - max wage and all the 57 unique state values
newdata2 <- with(subset(newdata,PREVAILING_WAGE < 200000),
                 data.frame(PREVAILING_WAGE = rep(seq(from = min(PREVAILING_WAGE),
                                                      to = max(PREVAILING_WAGE),
                                                      length.out = 100), 60),
                            WORKSITE_STATE = factor(rep(unique(WORKSITE_STATE), each = 100))))

# Predict the probalilities
newdata3 <- cbind(newdata2, predict(mylogit, newdata = newdata2, type = "link",se = TRUE))

newdata3 <- within(newdata3, {
  PredictedProb <- plogis(fit)
  LL<- plogis(fit - (1.96 *se.fit))
  UL <- plogis(fit + (1.96 * se.fit))
})

head(newdata3)
saveRDS(newdata3, "glmModel.rds")
return (newdata3)
```

Linear Regression Algorithm

Apriori Algorithm

```
# Apriori Plot
output$apprioriPlot <- renderPlot({
  print("Apriori Plot...")
  inspect(Rules)
  plot(Rules, method="paracoord", control=list(reorder=TRUE))
})
```

Libraries Used

```
library(shiny)
library(ggplot2)
library(dplyr)
library(lazyeval)
library(hashmap)
library(ggrepel)
library(stats)
library(rdrop2)
library(arules)
library(arulesViz)
library(stringr)
library(e1071)
library(randomForest)
```

Random Forest Algorithm

```
#select the first 1000 case statuses
select <- train_na[1:1000,]


# Random forest training 1
rf.train.1 <- data.combined.na[1:1000 ,c("FULL_TIME_POSITION","PREVAILING_WAGE")]
rf.label <- as.factor(select$CASE_STATUS)
rf.label <- factor(rf.label)


set.seed(1234)
rf.1 <- randomForest(x= rf.train.1,y=rf.label, importance=TRUE,ntree = 1000)
```

Conclusion and Future Scope

The H1B application has become a crucial factor in the lives of thousands of foreign workers, especially in the current political scenario. Thus, it is always good to be informed about the influential features of the application, that makes it CERTIFIED or DENIED. From our analytics, we understood that as the PREVAILING_WAGE rate increases, the application denial also increases. Especially, if the PREVAILING_WAGE is over $ 300,000. Apart from this, we have a wage prediction algorithm that helps the users to know prevailing wage in companies in the specified year, based on which they can make an informed decision for their H1B application. The future work may include finding the single influential attribute that affects the CASE_STATUS decision. Also, we need to increase the accuracy of our prediction algorithm, so as to give better information to users.

For training and validation of the Logistic regression model, instead of dividing the entire data into 80 and 20 ratio, 80% of the data from each year is chosen and the validation data is 20% which taken from all the 6 years. The test data for the model is given by the user in the User Interface. The probability of application being certified is calculated and compared with different work site states.

For training and validation of Linear regression model, Mean prevailing wage is considered against each year for building the model with respect to the work site state. For training the model there should be minimum of two years previous data for the respective year.

References

[1] J. MacQueen, ”Some methods for classification and analysis of multivariate observations”, in LeCam, L.M., Neyman, J., eds.: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley and Los Angeles, CA, USA, University of California Press, vol. 1, pp. 281–297, 1967.

[2] T. Kohonen, Overture. Self-Organizing neural networks: recent advances and applications, Springer-Verlag, New York, NY, USA, 2002, pp. 1–12.

[3] I. Dhillon, Y. Guan, B. Kulis, ”Kernel k-means: spectral clustering and normalized cuts”, in Proceeding KDD ’04 Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pp.551–556, 2004.

[4] R.C. de Amorim and B. Mirkin, ”Minkowski metric, feature weighting and anomalous cluster initializing in k-means clustering”, Pattern Recognition, vol. 45, no. 3, pp. 1061–1075, 2012.

[5] D. Pelleg and A.W. Moore, ”X-means: Extending k-means with efficient estimation of the number of clusters”, in Proceedings of the 17th International Conference on Machine Learning, Morgan Kaufmann, pp. 727–734, 2000.

[6] Y. Huang and L. Li, “Naive Bayes classification algorithm based on small sample set,” 2011 IEEE International Conference on Cloud Computing and Intelligence Systems, 2011.

[7] P. Vanitha and D. M. Mayilvaganan, “Survey On Meteorological Weather Analysis Based On Naïve Bayes Classification Algorithm,” International Journal Of Engineering And Computer Science, Mar. 2016.

[8] http://cseweb.ucsd.edu/classes/wi17/cse258-a/reports/a068.pdf