Final project for 15.062 Data Mining

Certification/Denial predictions of labor condition application (LCA) for H-1B visa petitions

Sohae Kim

Executive Summary

The labor condition application (LCA) is an application filed by prospective employers on behalf of employees applying for an employment-based, non-immigrant visa (H-1B, or variants of H-1B) in the United States and is necessary to get certified for H-1B visa petitions. In this project, I aim to predict the certification and denial decisions of the LCA (decision from the step 1 by using (1) k-nearest neighbors classifier, (2) classification trees, (3) logistic regression and (4) neural nets. I have processed, cleaned and explored the LCA raw data obtained from the DOLETA's OFLC webpage. Then, I have built four prediction models on CERTIFIED/DENIED decisions with the balanced and unbalanced data sets using two different approaches in treating the categorical variables, and evaluated the performance of the classification models compared to two baseline models. From the data processing, cleaning and exploitation, we have learned that the units of wages need to be consistent with the amount of wage and that the proposed wage rate of pay needs to be at least the prevailing wage. From the prediction models, we have obtained higher accuracy using the four classification models with balanced data set than using the baseline models. The classification tree performed the best among the four models, probably because the U.S. DOL makes the CERTIFIED/DENIED decisions based on some rules and logics and the classification tree model can mimic the decision making process better than other models. From the classification tree model, I find the chance of CERTIFIED decisions increases, as H1B_DEPENDENT exists, AGENT_REPRESENTING_EMPLOYER exists, PW_SOURCE is from OES and the EMPLOYER_NAME files more applications. On top of the current four classification models that already improved the prediction accuracy form the baseline models, I believe that the prediction models can be further improved by working on several possible approaches such as implementing PCA, checking invalid and inaccurate information and analyzing text variables better.

Introduction

The labor condition application (LCA) is an application filed by prospective employers on behalf of employees applying for an employment-based, non-immigrant visa (H-1B, or variants of H-1B) in the United States. The H-1B visa is one of the most common visa statuses that international students/scholars apply for or hold with a secured job position. Understanding and predicting the certification/denial of the visa can benefit the visa applicants, sponsor companies and lawyers in ways that they can strategize how to prepare for the visa petition. For example, the lawyers can screen the applicants' conditions through the predictor and determine whether to take the case or not, and which attributes of the petition need to be strengthened to get certified. Although lawyers with abundant experience in the H-1B visa petition might have built the instinct and/or reasoning to predict the decision based on the experience, novice lawyers, visa applicants and sponsor companies do not have any tool to gauge the probability of certification by themselves. Therefore, building the prediction model for the H-1B visa decision benefits those for the first screening before legal consultations or case study in detail.

The H-1B application steps include followings. Step 1: The prospective employer files the LCA and gets certified by the United States Department of Labor Employment and Training Administration (DOLETA)'s Office of Foreign Labor Certifications (OFLC). This involves that the attestations from the employers such as the prospective wages, the confirmed prevailing wages in the location of work and working conditions. Step 2: Once the LCA is certified by the DOLETA, the H1-B visa petitions should be filed at the proper United States Citizenship and Immigration Services (USCIS) office. Then, a decision letter will be sent out after the review.

In this project, I aim to predict the certification and denial decisions of the LCA (decision from the step 1 by using (1) k-nearest neighbors classifier, (2) classification trees, (3) logistic regression and (4) neural nets. Although the final decisions from the step 2 are not publicly available, the decisions from the step 1 is publicly available in the DOLETA's OFLC webpage: https://www.foreignlaborcert.doleta.gov/performancedata.cfm#dis. Since certification from the step 1 is necessary for the approval of the H1-B visa, we build the prediction model based on the available data of the step 1 decision to gauge the final decisions. From the DOLETA's OFLC webpage, I primarily used the most recent quarterly data (Oct. 1 – Dec. 31, 2017) for this report and have also performed preliminary examination of the second most recent but larger dataset (Oct. 1, 2016 – Sep. 30, 2017) – 2018 Q1 data since it was released in the first quarter of 2018.

Raw data processing and cleaning

The 2018 Q1 data consists of 52 variables with 94,622 records. The variables include each applicant's information such as case status, visa class, decision date and job title; employer's information such as name, address, total number of workers and proposed wage rate by the employer; and attorney’s information such as name, city and state. The full list of variable names and descriptions is tabulated in Appendix A. To avoid the curse of dimensionality, I manually omitted some variables in three groups. First, the variables that include the same information with another variable were omitted. For example, EMPLOYER_ADDRESS, EMPLOYER_POSTAL_CODE, EMPLOYER_PHONE and EMPLOYER_PHONE_EXT were omitted, since they explain who the employer is, as described in EMPLOYER_NAME. In a similar manner, I omitted AGENT_ATTORNEY_CITY, AGENT_ATTORNEY_STATE, WORKSITE_COUNTY and WORKSITE_POSTAL_CODE as well. Second, the variables with multiple entries of NA, “N/A” or “NA” were omitted such as EMPLOYER_BUSINESS_DBA, WAGE_RATE_OF_PAY_TO, LABOR_CON_AGREE and PUBLIC_DISCLOSURE_LOCATION. Third, date variables were omitted as variables for models such as CASE_SUBMITTED, DECISION_DATE, EMPLOYMENT_START_DATE, EMPLOYMENT_END_DATE and ORIGINAL_CERT_DATE.

The variables I believe important intuitively without looking at data in detail are as below:

- CASE_STATUS: Output variable. Certified, denied, certified but withdrawn, and withdrawn.
- EMPLOYER_NAME: Names of employers. We can gauge the wage levels, job title, number of employees, and credibility on visa support.
- AGENT_ATTORNEY_NAME: Names of agents/attorneys filing the petition on behalf of applicants/employers. We can expect applications to be of better quality when filed by attorneys compared to applications filed by applicants/employers. Furthermore, some attorneys may have more insight and experience resulting in higher certification rates.
- JOB_TITLE: Title of job. We can gauge the wage levels.
- PREVAILING_WAGE: Prevailing Wage for job requested. Basic rule of thumb for the LCA certification is to keep the right of work of American citizens by prohibiting worker visa holders from working at lower wage than the wage levels that is ought to be paid.
- WAGE_RATE_OF_PAY_FROM: Wage rate of pay starting from the value, proposed by the employers.

Regarding the categorical variables such as EMPLOYER_NAME, SOC_CODE, PW_WAGE_LEVELS and PW_SOURCE_YEAR, I approached in two ways. The first approach involves grouping levels of categories into fewer levels of categories if there are too many levels of factors, and then converting the categorical variables into dummy 0-1 variables for each level of categories. For example, JOB_TITLE contains 21,515 levels, and some of levels like ASIC DESIGN ENGINEER and ASIC DESIGN ENG barely have any difference each other. Therefore, I eliminated JOB_TITLE and used SOC_CODE to group them by different occupation groups, instead of converting JOB_TITLE into dummy 0-1 variables. SOC_CODE is occupational code classified by the Standard Occupational Classification (SOC) system, and the first two digits represent different major occupation groups. For example, SOC_CODE starting with 11- represents management occupations. Similarly, NAICS_CODE is industry code associated with the employer, classified by the North American Industrial Classification System (NAICS), and I used NAICS_CODE to group EMPLOYER_NAME by different industry group and eliminated JOB_TITLE. All the continuous numerical variables, with exception to the dummy 0-1 variables, were normalized.

The second approach to treat the categorical variables uses the frequency of each level of categories. In this approach, I kept the EMPLOYER_NAME and AGENT_ATTORNEY_NAME variables because I hypothesized that certain employers or attorneys may have a higher chance of certification. If certain employers and attorneys have abundant experience on the H1-B petition filing, they may have a better strategy on successful LCA certification, thereby increasing the certification chance compared to the applications with the same job title and/or the same industry. This abundant experience on the filing can be implicitly inferred from the frequency of EMPLOYER_NAME and/or AGENT_ATTORNEY_NAME among applications. Therefore, I transformed all categorical variables including EMPLOYER_NAME, AGENT_ATTORNEY_NAME, SOC_CODE, NAICS_CODE, PW_WAGE_LEVEL, PW_SOURCE and H1B_DEPENDENT, etc. into numerical variables of the frequency of categories, and then, normalized the values.

Among the 94,622 records, 80,960 cases (85.6%) are certified, 8,900 cases (9.4%) are certified but withdrawn, 1,559 cases (1.6%) are denied, and 3,202 cases (3.4%) are withdrawn before the decision. Since the data is unbalanced with respect to the certification/denial decisions, I used both balanced and unbalanced data sampling to compare the prediction accuracy of analytical models.

Iterations of data exploitation, processing and cleaning

After initial data processing and cleaning, I mainly examined the proposed wage rate of pay and prevailing wage. One of the most important factors of H-1B approval is that the work visa holder does not get paid a lower wage than American citizens for the same work to ensure the work visa holder does not take over American citizens' right of labor. Due to this reason, I assumed that the proposed wage rate should be at least the prevailing wage to get certified and considered them important.

Figure 1 shows the raw data distribution of the wage rate proposed by employers and prevailing wage for the jobs. I find that these wages are grouped in four: (1) both prevailing wage and the proposed wage rate of pay are less than 10K; (2) prevailing wage is less than 10K but the proposed wage rate of pay is bigger than 10K; (3) prevailing wage is bigger than 10K but the proposed wage rate of pay is less than 10K; and (4) both prevailing wage and the proposed wage rate of pay is bigger than 10K. Here, we can ask why the wage distribution is discontinuous around 10K. The answer for this question is that the application asks the amount of wages and units of the wage separately so hourly, weekly, bi-weekly, monthly and yearly wages are segregated in different groups. Even in a single record, mismatch between the units of prevailing wage & the proposed wage rate of pay exists for 147 records in 2018 Q1 data. Therefore, I changed the hourly, weekly, bi-weekly and monthly wages into hourly wages by assuming 52 working weeks per year, 5 working days per week, 40 working hours per week.

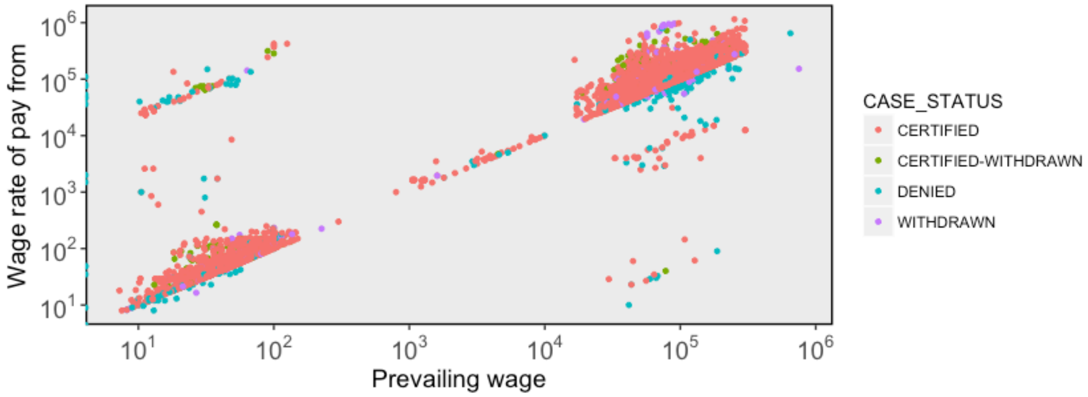


Figure 1. Raw data distribution of prevailing wage & wage rate of pay proposed by employers

Figure 2 shows the distribution of prevailing wage and the proposed wage rate of pay after unifying the units of wages as yearly-based. Here, I notice that two groups of outliers exist in this scatter plot, and these two groups are all denied: (1) the prevailing wage and/or the proposed wage rate of pay is very low, less than 10K; and (2) the prevailing wage is very high, larger than ~1M.

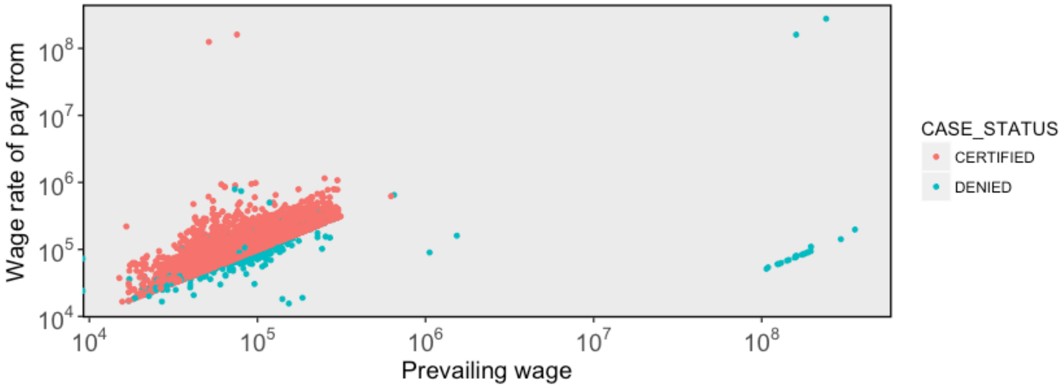


Figure 2. Distribution of prevailing wage & wage rate of pay proposed by employers after unifying the units of wages to USD per year

The second group of very high prevailing wage consists of 22 records and shares two similarities between the records. First, the top 22 cases in the prevailing wage denote the unit of prevailing wages as hourly-based, thereby making the converted yearly-based wage outrageously high. Second, for 20 cases among those top 22 cases, the proposed wage rage of pay is in the range between 50K and 200K yearly. A couple of cases reported the proposed wage rate of pay as ~77K and ~133K hourly. These two similarities imply that the applicants made mistakes on the unit of prevailing wage or misinterpreted the unit as how to get paid for the yearly wage.

So far, we have examined the prevailing wage and the proposed wage rate of pay and found that those outlier groups are denied without exceptions. I believe that these outliers do not need the prediction model since if-else conditional test can determine the decisions. Therefore, I eliminate those outliers from the data used for the prediction model development.

Figure 3 shows the wage distribution after eliminating the outlier groups for CERTIFIED and DENIED cases, separately. The CERTIFIED group has a linear cut-off for the wage rate of pay below the prevailing wage and has a dense population around the cut-off. The DENIED group also has a dense population around the cut-off, and the wage rate of pay higher than the prevailing wage does not guarantee the certification. Due to this strong relationship between the wage rate of pay, prevailing wage and the decision, I added an extra variable, WAGE_DIFF, which is the wage difference between the proposed wage rate of pay and the prevailing wage. Further exploitation did not provide any more insights on the difference between the CERTIFIED and DENIED groups. Therefore, here is where we need a good prediction model.

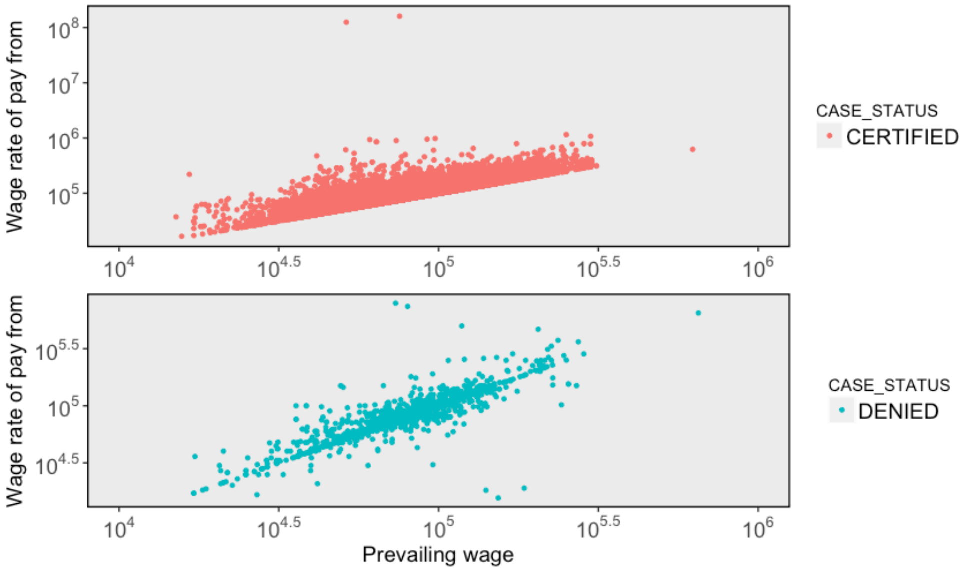


Figure 3. Distribution of the prevailing wage & proposed wage rate of pay after unifying the units of wages and eliminating the outlier groups

Prediction models

I have used four different classification methods for the prediction models of the decisions: (1) classification tree, (2) logistic regression, (3) k-nearest-neighbors classifier and (4) a neural network.

To evaluate the predictive performance of each model, I have examined accuracy of the prediction. Data is split into three groups, which are training, validation and test sets that consist of 50%, 25% and 25%, respectively. Models were built based on the training set; the probability cutoff value to determine 0 or 1 for the results was optimized as values around 0.5 using the validation set; and the accuracy of each model was calculated by using the test set.

Two baseline models were also built as benchmarks, the first one based on all-CERTIFIED prediction (zeroR) and the second one based on the relationship between the prevailing wage, the proposed wage rate of pay and CASE_STATUS. The all-CERTIFIED prediction predicts CERTIFIED for every record, since it is the majority of the data. The wage-based model predicts 100% DENIED if the proposed wage rate of pay is smaller than the prevailing wage and 100% CERTIFIED otherwise, as learned from the data exploitation.

Results

The prediction accuracy using the four different classification methods and the two base models are tabulated in Table 1. For all of the mothers, I tested both balanced and unbalanced data set and approached in two ways to deal with the categorical variables, as tabulated along each column in Table 1. These accuracies are the averaged accuracies over 20 random data samples of ~3,000 records per sample from 82,500 records.

Using the balanced data set, the classification tree model and the neural network model perform the best and better than the two base models. Since the CERTIFIED/DENIED decisions are made based on rules and logics in the U.S. Department of Labor (DOL), the classification tree model with some criteria on variables of a single record seems to mimic the decision making process better than the other methods. Although we cannot directly guess how the neural network method yields a higher accuracy than the other two methods, it is very interesting to note that it performs pretty well.

Table 1. Accuracy of four classification methods (Classification tree, Logistic regression, K-nearest neighbors and Neural network) models and two base models (all-CERTIFIED model and wage-based model) using two approaches in treating the categorical variables (using 0-1 dummy variables and frequency variables) for balanced and unbalanced data set

		Four classification models				Baseline models	
		Classification tree	Logistic regression	K-nearest neighbors	Neural network	All-CERTIFIED	Wage-based
Balanced	Dummy	0.756	0.676	0.700	0.739	0.500	0.603
	Frequency	0.777	0.720	0.697	0.746	0.500	0.603
Unbalanced	Dummy	0.981	0.951	0.980	0.974	0.982	0.985
	Frequency	0.980	0.979	0.981	0.973	0.982	0.985

Using the unbalanced data set, the baseline models perform better than the other four classification models. I think that this might be because the data is too biased toward CERTIFIED so that building classification models have a high noise-to-signal (DENIED) ratio to be accurate. In other words, too few data of DENIED (1.8% on average) compared to CERTIFIED (98.2% on average) cannot be enough to build a model and to discriminate them from CERTIFIED data in the model.

Interpretation of the classification trees

Since the classification tree model predicts CASE_STATUS the best using the balanced data set, we can infer what factors contribute to the CERTIFIED/DENIED decisions. Appendix B includes six exemplary classification trees with the balanced data set using the two different approaches treating the categorical variables, three trees for each. As understood from the data exploitation, WAGE_DIFF is one of the most important variables and screen out the first group of DENIAL. Other important variables include H1B_DEPENDENT, AGENT_REPRESENTING_EMPLOYER and PW_SOURCE, as well as EMPLOYER_NAME if frequency is used for categorical variables. The chance of certification increases, as H1B_DEPENDENT exists, AGENT_REPRESENTING_EMPLOYER exists, PW_SOURCE is from OES (Occupational Employment Statistics by U.S. DOL Bureau of Labor Statistics), and the EMPLOYER files more applications. However, we may need a bit more careful examination on these trends because the model might be inclined toward the CERTIFIED if the record just belongs to the majority due to the unbalanced probability of CERTIFIED and DENIED decisions. The existence of AGENT_REPRESENTING_EMPLOYER, PW_SOURCE from OES and frequent data from certain EMPLOYER_NAME belong to majority of data, while the existence of H1B_DEPENDENT does not.

Possible improvements in the future

- Implementation of principal component analysis (PCA)
  To avoid the curse of dimensionality, I omitted some variables but think that some correlations between the variables exist, as examined in the data exploitation. To further reduce the data dimension, we could perform PCA. I believe that the PCA and changing weights between the correlated variables will allow us to reduce the dimensions further and to reduce the computational cost with a bigger dataset.

- Another prediction model by checking invalid and inaccurate information
  According to the Code of Federal Regulations Title 20, 20 CFR 655.740 - What actions are taken on labor condition applications?, LCA will be denied, when either of both of the following two conditions exist: (i) When the application is not properly completed; and (ii)When the Form ETA 9035 or ETA 9035E contains obvious inaccuracies. Therefore, we could build the prediction models based on these conditions.

SOC_CODE and NAICS_CODE can be example values to check the validity. SOC_CODE and NAICS_CODE are occupational and industry classification codes that are classified by U.S. DOL Bureau of Labor Statistics and U.S. Department of Commerce, Census Bureau. Therefore, a set of codes is already assigned and the codes cannot be any other values than the assigned set of codes. Information about the prevailing wage such as PREVAILING_WAGE, PW_WAGE_LEVEL and PW_SOURCE can be another set of values to be check the validity.

- Text analysis on text variables
  Text analysis on some variables such as EMPLOYER_NAME and JOB_TITLE could be beneficial to identify the text information better and to increase the predicting accuracies. For example, five different levels exist for AMAZON FULFILLMENT SERVICES, INC. in the variable EMPLOYER_NAME, which are “AMAZON FULFILLMENT SERVICES INC”, “AMAZON FULFILLMENT SERVICES INC.”, “AMAZON FULFILLMENT SERVICES, INC”, “AMAZON FULFILLMENT SERVICES, INC,”, “AMAZON FULFILLMENT SERVICES, INC.”. The vast majority of applications chose the last level, thus few applications (3.5%, 0.4%, 1.6% and 0.1%) were included for the rest of the levels, thereby diluting the EMPLOYER_NAME data levels to small frequencies. If we treat them as just one level by using text analysis, the grouping and/or counting frequency will be more accurate, thereby enabling the prediction models to perform with a better accuracy.

Conclusions

I have processed, cleaned and explored the LCA raw data obtained from the DOLETA’s OFLC webpage. Then, I have built four prediction models on CERTIFIED/DENIED decisions with the balanced and unbalanced data sets using two different approaches in treating the categorical variables, and evaluated the performance of the classification models compared to two baseline models. From the data processing, cleaning and exploitation, we have learned that the units of wages need to be consistent with the amount of wage and that the proposed wage rate of pay needs to be at least the prevailing wage. From the prediction models, we have obtained higher accuracy using the four classification models with balanced data set than using the baseline models. The classification tree performed the best among the four models, probably because the U.S. DOL makes the CERTIFIED/DENIED decisions based on some rules and logics and the classification tree model can mimic the decision making process better than other models. From the classification tree model, I find the chance of CERTIFIED decisions increases, as H1B_DEPENDENT exists, AGENT_REPRESENTING_EMPLOYER exists, PW_SOURCE is from OES and the EMPLOYER_NAME files more applications. On top of the current four classification models that already improved the prediction accuracy form the baseline models, I believe that the prediction models can be further improved by working on several possible approaches such as implementing PCA, checking invalid and inaccurate information and analyzing text variables better.

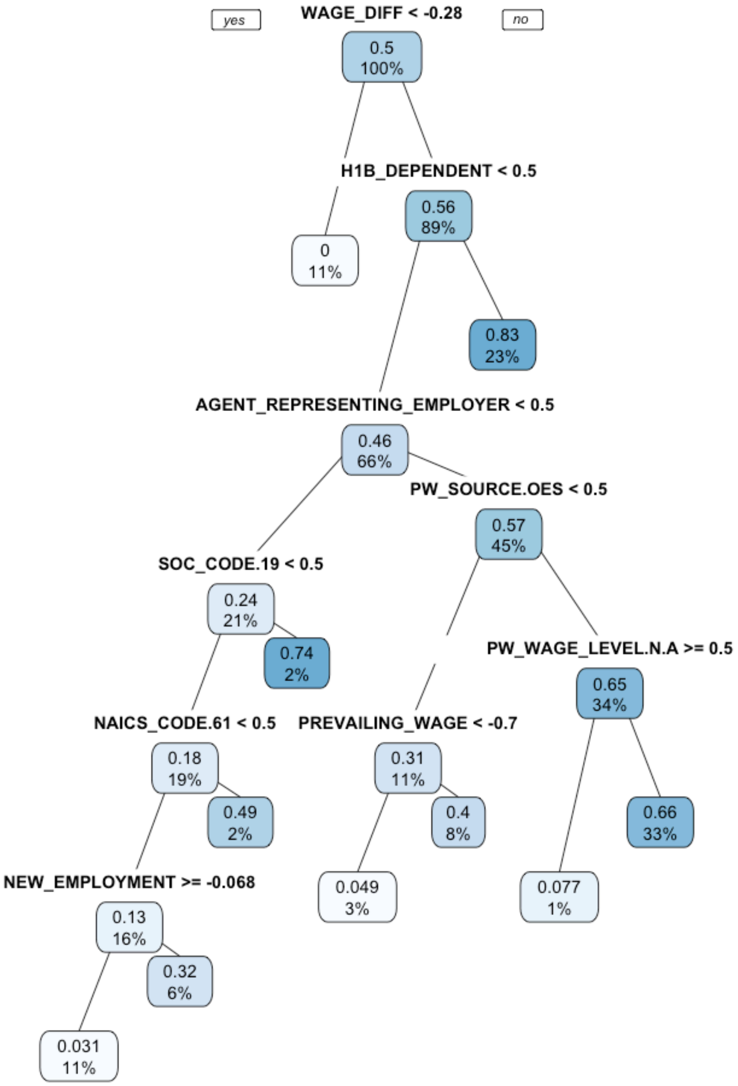
Appendix A. List of names and descriptions of 52 variables in data

Name	Description
CASE_NUMBER	Unique identifier assigned to each application submitted for processing to the Chicago National Processing Center.
CASE_STATUS	Status associated with the last significant event or decision. Valid values include “Certified,” “Certified-Withdrawn,” Denied,” and “Withdrawn”.
CASE_SUBMITTED	Date and time the application was submitted.
DECISION_DATE	Date on which the last significant event or decision was recorded by the Chicago National Processing Center.
VISA_CLASS	Indicates the type of temporary application submitted for processing. Values include H-1B, E-3 Australian, H-1B1 Chile, and H-1B1 Singapore.
EMPLOYMENT_START_DATE	Beginning date of employment.
EMPLOYMENT_END_DATE	Ending date of employment.
EMPLOYER_NAME	Name of employer submitting labor condition application.
EMPLOYER_BUSINESS_DBA	Trade Name or dba name of employer submitting labor condition application, if applicable.
EMPLOYER_ADDRESS	Contact information of the Employer requesting temporary labor certification.
EMPLOYER_CITY	
EMPLOYER_STATE	
EMPLOYER_POSTAL_CODE	
EMPLOYER_COUNTRY	
EMPLOYER_PROVINCE	
EMPLOYER_PHONE	
EMPLOYER_PHONE_EXT	
AGENT_REPRESENTING_EMPLOYER	Y = Employer is represented by an Agent or Attorney; N = Employer is not represented by an Agent or Attorney.
AGENT_ATTORNEY_NAME	Name of Agent or Attorney filing an H-1B application on behalf of the employer.
AGENT_ATTORNEY_CITY	City information for the Agent or Attorney filing an H-1B application on behalf of the employer.
AGENT_ATTORNEY_STATE JOB_TITLE	State information for the Agent or Attorney filing an H-1B application on behalf of the employer.
SOC_CODE SOC_NAME	Title of the job.
NAICS_CODE	Occupational code associated with the job being requested for temporary labor condition, as classified by the Standard Occupational Classification (SOC) System.
TOTAL_WORKERS NEW_EMPLOYMENT	Occupational name associated with the SOC_CODE.
CONTINUED_EMPLOYMENT	Industry code associated with the employer requesting permanent labor condition, as classified by the North American Industrial Classification System (NAICS).
CHANGE_PREVIOUS_EMPLOYMENT	Total number of foreign workers requested by the Employer(s).
NEW_CONCURRENT_EMPLO	Indicates requested worker(s) will begin employment for new

YMENT	employer, as defined by USCIS I-29.
CHANGE_EMPLOYER	Indicates requested worker(s) will be continuing employment with same employer, as defined by USCIS I-29.
AMENDED_PETITION	Indicates requested worker(s) will be continuing employment with same employer without material change to job duties, as defined by USCIS I-29. Indicates requested worker(s) will begin employment with additional employer, as defined by USCIS I-29.
FULL_TIME_POSITION	Indicates requested worker(s) will begin employment for new employer, using the same classification currently held, as defined by USCIS I-29. Indicates requested worker(s) will be continuing employment with same employer with material change to job duties, as defined by USCIS I-29. Y = Full Time Position; N = Part Time Position.
PREVAILING_WAGE	Prevailing Wage for the job being requested for temporary labor condition.
PW_UNIT_OF_PAY	Unit of Pay. Valid values include “Daily (DAI),” “Hourly (HR),” “Bi-weekly (BI),” “Weekly (WK),” “Monthly (MTH),” and “Yearly (YR)”.
PW_WAGE_LEVEL	Variables include "I", "II", "III", "IV" or "N/A."
PW_SOURCE	Variables include "OES", "CBA", "DBA", "SCA" or "Other".
PW_SOURCE_YEAR	Year the Prevailing Wage Source was Issued.
PW_SOURCE_OTHER	If "Other Wage Source", provide the source of wage.
WAGE_RATE_OF_PAY_FROM	Employer’s proposed wage rate.
WAGE_RATE_OF_PAY_TO	Maximum proposed wage rate.
WAGE_UNIT_OF_PAY H-1B_DEPENDENT	Unit of pay. Valid values include “Hour", "Week", "Bi-Weekly", "Month", or "Year".
WILLFUL_VIOLATOR	Y = Employer is H-1B Dependent; N = Employer is not H-1B Dependent.
SUPPORT_H1B	Y = Employer has been previously found to be a Willful Violator; N = Employer has not been considered a Willful Violator.
LABOR_CON_AGREE	Y = Employer will use the temporary labor condition application only to support H-1B petitions or extensions of status of exempt H-1B worker(s); N = Employer will not use the temporary labor condition application to support H-1B petitions or extensions of status for exempt H-1B worker(s);
PUBLIC_DISCLOSURE_LOCATION	Variables include "Place of Business" or "Place of Employment."
WORKSITE_CITY	City information of the foreign worker's intended area of employment.
WORKSITE_COUNTY	County information of the foreign worker's intended area of employment.
WORKSITE_STATE	State information of the foreign worker's intended area of employment.
WORKSITE_POSTAL_CODE	Zip Code information of the foreign worker's intended area of employment.
ORIGINAL_CERT_DATE	Original Certification Date for a Certified_Withdrawn application.

Appendix B. Classification trees obtained from the balanced data set using two approaches in treating the categorical variables

(1) Using 0-1 dummy variables for categorical variables

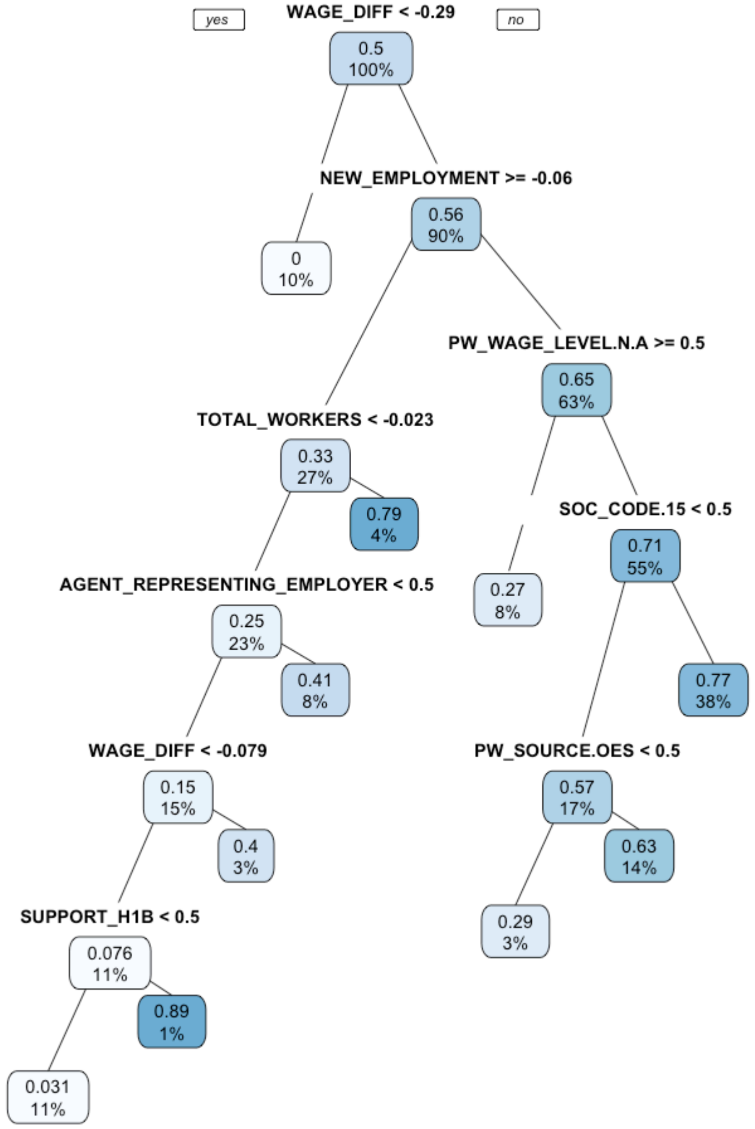


```
> tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
[1] 0.7751322751 0.5000000000
Call:
rpart(formula = CASE_STATUS ~ ., data = trdf)
  n= 1512

             CP nsplit    rel error        xerror            xstd
1 0.11751662971      0 1.0000000000 1.0019039799 0.0005129764936
2 0.09424011553      1 0.8824833703 0.8853609429 0.0096536203510
3 0.06265397247      2 0.7882432548 0.7907412855 0.0152519086046
4 0.03741146910      3 0.7255892823 0.7282904858 0.0186329988865
5 0.02505766060      4 0.6881778132 0.7059872871 0.0205875228654
6 0.01158164935      5 0.6631201526 0.6712938752 0.0211931731774
7 0.01153095213      7 0.6399568539 0.6825411233 0.0218155213959
8 0.01007514378      8 0.6284259018 0.6796094004 0.0218017972063
9 0.01000000000      9 0.6183507580 0.6725979566 0.0219033233631

Variable importance
                  WAGE_DIFF                             H1B_DEPENDENT
SUPPORT_H1B AGENT_REPRESENTING_EMPLOYER                 PW_SOURCE.OES
PW_SOURCE.Other
                         20                                        15
15                                  10                              6
6
          PW_WAGE_LEVEL.N.A                               SOC_CODE.19
NEW_EMPLOYMENT                                        PREVAILING_WAGE
NAICS_CODE.61       WAGE_RATE_OF_PAY_FROM
                          5                                         5
4                                    3                              3
2
           AMENDED_PETITION                      PW_SOURCE_YEAR.2017
CHANGE_EMPLOYER                                 CONTINUED_EMPLOYMENT
SOC_CODE.25         PW_SOURCE_YEAR.2016
                          1                                         1
1                                    1                              1
1

Node number 1: 1512 observations,    complexity param=0.1175166297
  mean=0.5, MSE=0.25
  left son=2 (159 obs) right son=3 (1353 obs)
  Primary splits:
      WAGE_DIFF                   <  -0.2849143498    to   the   left,
improve=0.11751662970, (0 missing)
      H1B_DEPENDENT       <  0.5                       to  the  left,
improve=0.08779194662, (0 missing)
      SUPPORT_H1B          <  0.5                      to  the  left,
improve=0.08704420541, (0 missing)
```

```
      NEW_EMPLOYMENT  < -0.06827305677 to the right, improve=0.05985907180, (0 missing)
      VISA_CLASS.H.1B < 0.5            to the left,  improve=0.05358914399, (0 missing)
  Surrogate splits:
      CHANGE_EMPLOYER < 10.84468214    to the right, agree=0.896, adj=0.013, (0 split)
      SOC_CODE.37     < 0.5            to the right, agree=0.896, adj=0.013, (0 split)
      PREVAILING_WAGE < 5.01118414     to the right, agree=0.896, adj=0.006, (0 split)
      SOC_CODE.47     < 0.5            to the right, agree=0.896, adj=0.006, (0 split)

Node number 2: 159 observations
  mean=0, MSE=0 

Node number 3: 1353 observations,    complexity param=0.09424011553
  mean=0.5587583149, MSE=0.2465474604 
  left son=6 (1001 obs) right son=7 (352 obs)
  Primary splits:
      H1B_DEPENDENT     < 0.5            to the left,  improve=0.10678967870, (0 missing)
      SUPPORT_H1B       < 0.5            to the left,  improve=0.10458372890, (0 missing)
      NEW_EMPLOYMENT    < -0.06827305677 to the right, improve=0.07880448930, (0 missing)
      PW_WAGE_LEVEL.N.A < 0.5            to the right, improve=0.07115664323, (0 missing)
      VISA_CLASS.H.1B   < 0.5            to the left,  improve=0.05882655325, (0 missing)
  Surrogate splits:
      SUPPORT_H1B      < 0.5            to the left,  agree=0.991, adj=0.966, (0 split)
      AMENDED_PETITION < 0.03783364027  to the left,  agree=0.761, adj=0.082, (0 split)
      CHANGE_EMPLOYER  < 1.105905772    to the left,  agree=0.741, adj=0.003, (0 split)

Node number 6: 1001 observations,    complexity param=0.06265397247
  mean=0.4625374625, MSE=0.2485965583 
  left son=12 (317 obs) right son=13 (684 obs)
  Primary splits:
      AGENT_REPRESENTING_EMPLOYER < 0.5            to the left,  improve=0.09517244411, (0 missing)
      NEW_EMPLOYMENT              < -0.06827305677 to the right, improve=0.07032208163, (0 missing)
      PW_WAGE_LEVEL.N.A           < 0.5            to the right, improve=0.05620172723, (0 missing)
      WAGE_RATE_OF_PAY_FROM       < -0.1172501965  to the left,  improve=0.04650677875, (0 missing)
      VISA_CLASS.H.1B             < 0.5            to the left,  improve=0.04266317884, (0 missing)
  Surrogate splits:
      NEW_EMPLOYMENT      < -0.06827305677 to the right, agree=0.744, adj=0.192, (0 split)
      NAICS_CODE.61       < 0.5            to the right, agree=0.709, adj=0.082, (0 split)
      PREVAILING_WAGE     < -1.058994038   to the left,  agree=0.701, adj=0.057, (0 split)
      SOC_CODE.19         < 0.5            to the right, agree=0.700, adj=0.054, (0 split)
      PW_SOURCE_YEAR.2017 < 0.5            to the left,  agree=0.697, adj=0.044, (0 split)

Node number 7: 352 observations
  mean=0.8323863636, MSE=0.1395193053 

Node number 12: 317 observations,    complexity param=0.0250576606
  mean=0.2365930599, MSE=0.1806167839 
  left son=24 (283 obs) right son=25 (34 obs)
  Primary splits:
      SOC_CODE.19      < 0.5            to the left,  improve=0.16543026110, (0 missing)
      NAICS_CODE.61    < 0.5            to the left,  improve=0.13187124770, (0 missing)
      WAGE_DIFF        < -0.2807026681  to the left,  improve=0.12875492530, (0 missing)
      NEW_EMPLOYMENT   < -0.06827305677 to the right, improve=0.09080559254, (0 missing)
      AMENDED_PETITION < 0.03783364027  to the left,  improve=0.05285954735, (0 missing)

Node number 13: 684 observations,    complexity param=0.0374114691
  mean=0.567251462, MSE=0.2454772409
  left son=26 (166 obs) right son=27 (518 obs)
  Primary splits:
      PW_SOURCE.OES         < 0.5            to the left,  improve=0.08422271315, (0 missing)
      PW_SOURCE.Other       < 0.5            to the right, improve=0.08263026846, (0 missing)
      PW_WAGE_LEVEL.N.A     < 0.5            to the right, improve=0.06863413308, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.7069275797  to the left,  improve=0.05349398691, (0 missing)
      PREVAILING_WAGE       < -0.7806546902  to the left,  improve=0.04284547656, (0 missing)
  Surrogate splits:
      PW_SOURCE.Other       < 0.5            to the right, agree=0.993, adj=0.970, (0 split)
      PW_WAGE_LEVEL.N.A     < 0.5            to the right, agree=0.874, adj=0.482, (0 split)
      PW_SOURCE_YEAR.2017   < 0.5            to the left,  agree=0.781, adj=0.096, (0 split)
      PW_SOURCE_YEAR.2016   < 0.5            to the right, agree=0.778, adj=0.084, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.507259605   to the left,  agree=0.768, adj=0.042, (0 split)

Node number 24: 283 observations,    complexity param=0.01158164935
  mean=0.1766784452, MSE=0.1454631722
  left son=48 (246 obs) right son=49 (37 obs)
  Primary splits:
      NAICS_CODE.61    < 0.5            to the left,  improve=0.09924278303, (0 missing)
      WAGE_DIFF        < 0.1896289892   to the left,  improve=0.09672109165, (0 missing)
      NEW_EMPLOYMENT   < -0.06827305677 to the right, improve=0.09507283491, (0 missing)
      SOC_CODE.25      < 0.5            to the left,  improve=0.08006258086, (0 missing)
      AMENDED_PETITION < 0.03783364027  to the left,  improve=0.06895157261, (0 missing)
  Surrogate splits:
      SOC_CODE.25 < 0.5            to the left,  agree=0.922, adj=0.405, (0 split)
      WAGE_DIFF   < 1.302159012    to the left,  agree=0.873, adj=0.027, (0 split)
      SOC_CODE.21 < 0.5            to the left,  agree=0.873, adj=0.027, (0 split)

Node number 25: 34 observations
  mean=0.7352941176, MSE=0.1946366782

Node number 26: 166 observations,    complexity param=0.01007514378
  mean=0.313253012, MSE=0.2151255625
  left son=52 (41 obs) right son=53 (125 obs)
  Primary splits:
      PREVAILING_WAGE       < -0.7006264828  to the left,  improve=0.10664560090, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.7365002132  to the left,  improve=0.08957493413, (0 missing)
      SOC_CODE.15           < 0.5            to the left,  improve=0.07977609245, (0 missing)
      VISA_CLASS.H.1B       < 0.5            to the left,  improve=0.04865497076, (0 missing)
      NAICS_CODE.42         < 0.5            to the left,  improve=0.03291307611, (0 missing)
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < -0.7365002132  to the left,  agree=0.976, adj=0.902, (0 split)
      SOC_CODE.19           < 0.5            to the right, agree=0.777, adj=0.098, (0 split)
      SOC_CODE.13           < 0.5            to the right, agree=0.771, adj=0.073, (0 split)
      SOC_CODE.27           < 0.5            to the right, agree=0.771, adj=0.073, (0 split)
      NAICS_CODE.44         < 0.5            to the right, agree=0.771, adj=0.073, (0 split)

Node number 27: 518 observations,    complexity param=0.01153095213
```

```
  mean=0.6486486486, MSE=0.2279035793 
  left son=54 (13 obs) right son=55 (505 obs)
  Primary splits:
      PW_WAGE_LEVEL.N.A       < 0.5           to the right, improve=0.03692122171, (0 missing)
      WAGE_RATE_OF_PAY_FROM   < -0.7069275797 to the left,  improve=0.02533107890, (0 missing)
      PW_WAGE_LEVEL.Level.III < 0.5           to the left,  improve=0.02243929732, (0 missing)
      PREVAILING_WAGE         < -1.388071967  to the left,  improve=0.02222520176, (0 missing)
      SOC_CODE.15             < 0.5           to the left,  improve=0.01831010596, (0 missing)
  Surrogate splits:
      SOC_CODE.53   < 0.5           to the right, agree=0.979, adj=0.154, (0 split)
      NAICS_CODE.48 < 0.5           to the right, agree=0.977, adj=0.077, (0 split)

Node number 48: 246 observations,    complexity param=0.01158164935
  mean=0.1300813008, MSE=0.113160156 
  left son=96 (162 obs) right son=97 (84 obs)
  Primary splits:
      NEW_EMPLOYMENT        < -0.06827305677 to the right, improve=0.16777037590, (0 missing)
      AMENDED_PETITION      < 0.03783364027  to the left,  improve=0.14752255100, (0 missing)
      NAICS_CODE.51         < 0.5            to the left,  improve=0.10387413100, (0 missing)
      WAGE_DIFF             < -0.2788685728  to the left,  improve=0.09304573265, (0 missing)
      WAGE_RATE_OF_PAY_FROM < 0.09654090939  to the left,  improve=0.05942394960, (0 missing)
  Surrogate splits:
      CONTINUED_EMPLOYMENT  < -0.0279406749  to the left,  agree=0.793, adj=0.393, (0 split)
      CHANGE_EMPLOYER       < 0.003402786991 to the left,  agree=0.785, adj=0.369, (0 split)
      WAGE_DIFF             < -0.03347410691 to the left,  agree=0.744, adj=0.250, (0 split)
      PREVAILING_WAGE       < 0.7412602643   to the left,  agree=0.715, adj=0.167, (0 split)
      WAGE_RATE_OF_PAY_FROM < 0.04947260455  to the left,  agree=0.715, adj=0.167, (0 split)

Node number 49: 37 observations
  mean=0.4864864865, MSE=0.249817385 

Node number 52: 41 observations
  mean=0.0487804878, MSE=0.04640095181 

Node number 53: 125 observations
  mean=0.4, MSE=0.24 

Node number 54: 13 observations
  mean=0.07692307692, MSE=0.07100591716 

Node number 55: 505 observations
  mean=0.6633663366, MSE=0.2233114401 

Node number 96: 162 observations
  mean=0.03086419753, MSE=0.02991159884 

Node number 97: 84 observations
  mean=0.3214285714, MSE=0.2181122449 

                       [,1]         [,2]
Accuracy       0.7751322751 0.7645502646
Sensitivity    0.8809523810 0.8677248677
```

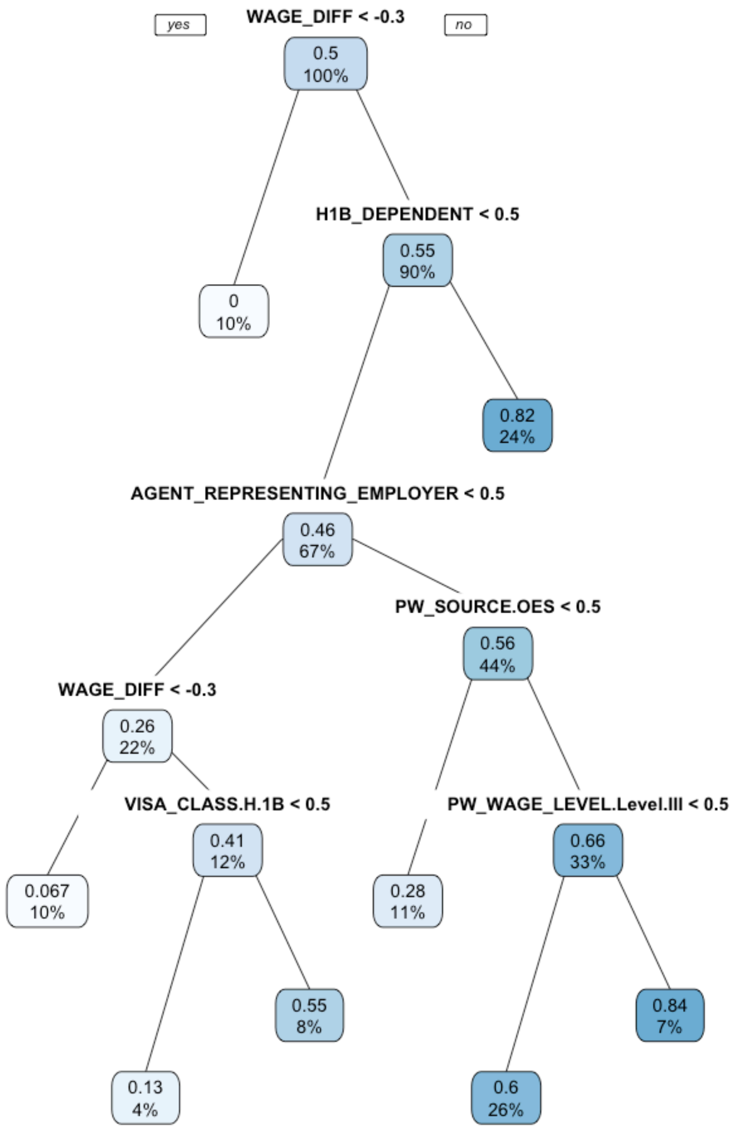
```
Specificity    0.6693121693 0.6613756614
Pos Pred Value 0.7270742358 0.7192982456
Neg Pred Value 0.8489932886 0.8333333333
```



```
> tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
[1] 0.7566137566 0.5000000000
Call:
rpart(formula = CASE_STATUS ~ ., data = trdf)
  n= 1512

             CP nsplit    rel error       xerror            xstd
1 0.11094783248      0 1.0000000000 1.0015963700 0.0004701283817
2 0.08122099389      1 0.8890521675 0.8917937375 0.0093461669749
3 0.05316611855      2 0.8078311736 0.8122764366 0.0164031290604
4 0.03920469848      3 0.7546650551 0.7724249790 0.0190691107590
5 0.01820957666      4 0.7154603566 0.7210425530 0.0206880568728
6 0.01432447382      5 0.6972507799 0.7231943235 0.0216554618749
7 0.01394173629      6 0.6829263061 0.7172716142 0.0219923236268
8 0.01012415326      8 0.6550428335 0.6934419383 0.0215030545967
9 0.01000000000      9 0.6449186803 0.6814475675 0.0213344477764

Variable importance
                  WAGE_DIFF                        NEW_EMPLOYMENT
PW_WAGE_LEVEL.N.A                                    TOTAL_WORKERS
CONTINUED_EMPLOYMENT            AMENDED_PETITION
                         22                                    20
9                               8                               4
4
            CHANGE_EMPLOYER           CHANGE_PREVIOUS_EMPLOYMENT
SOC_CODE.15                                          SUPPORT_H1B
PREVAILING_WAGE       WAGE_RATE_OF_PAY_FROM
                          4                                    3
3                               3                               3
3
AGENT_REPRESENTING_EMPLOYER                         H1B_DEPENDENT
PW_SOURCE.OES                                      VISA_CLASS.H.1B
PW_SOURCE.Other                 SOC_CODE.13
                          3                                    2
2                               2                               2
1
                SOC_CODE.17
                          1

Node   number   1:   1512   observations,          complexity
param=0.1109478325
  mean=0.5, MSE=0.25
  left son=2 (151 obs) right son=3 (1361 obs)
  Primary splits:
      WAGE_DIFF                <  -0.2881880086    to  the  left,
improve=0.11094783250, (0 missing)
      NEW_EMPLOYMENT         <   -0.06018574666   to   the   right,
improve=0.07376857572, (0 missing)
      SUPPORT_H1B           <  0.5                    to  the  left,
improve=0.07216806690, (0 missing)
      H1B_DEPENDENT        <  0.5                     to  the  left,
improve=0.06873704908, (0 missing)
```

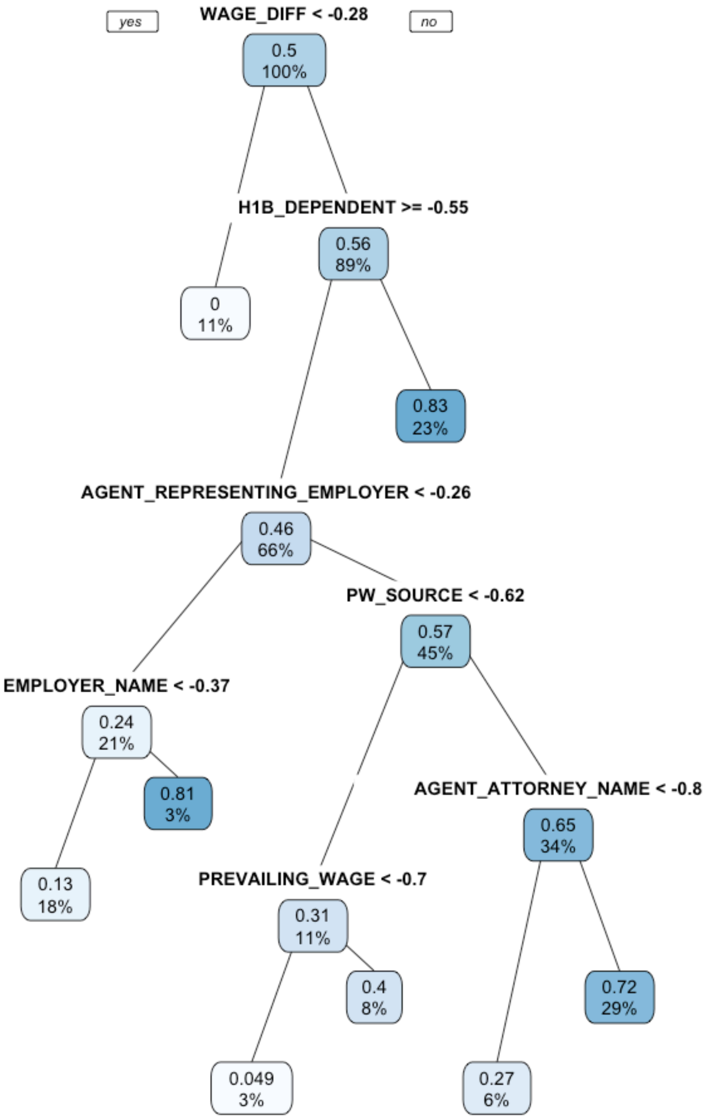
```
      PW_WAGE_LEVEL.N.A < 0.5            to the right, improve=0.05106639952, (0 missing)
  Surrogate splits:
      SOC_CODE.47           < 0.5            to the right, agree=0.902, adj=0.020, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.772465109   to the left,  agree=0.901, adj=0.013, (0 split)
      SOC_CODE.43           < 0.5            to the right, agree=0.901, adj=0.007, (0 split)

Node number 2: 151 observations
  mean=0, MSE=0 

Node number 3: 1361 observations,    complexity param=0.08122099389
  mean=0.5554739162, MSE=0.2469226446 
  left son=6 (406 obs) right son=7 (955 obs)
  Primary splits:
      NEW_EMPLOYMENT      < -0.06018574666 to the right, improve=0.09135683693, (0 missing)
      SUPPORT_H1B         < 0.5            to the left,  improve=0.07804347629, (0 missing)
      H1B_DEPENDENT       < 0.5            to the left,  improve=0.07576642975, (0 missing)
      PW_WAGE_LEVEL.N.A   < 0.5            to the right, improve=0.06556791018, (0 missing)
      PW_SOURCE_YEAR.2017 < 0.5            to the left,  improve=0.05339649734, (0 missing)
  Surrogate splits:
      PREVAILING_WAGE       < -1.124248216   to the left,  agree=0.732, adj=0.101, (0 split)
      VISA_CLASS.H.1B       < 0.5            to the left,  agree=0.730, adj=0.094, (0 split)
      WAGE_RATE_OF_PAY_FROM < -0.9416475393  to the left,  agree=0.729, adj=0.091, (0 split)
      TOTAL_WORKERS         < 0.1488328109   to the right, agree=0.719, adj=0.057, (0 split)
      CONTINUED_EMPLOYMENT  < 0.2424696058   to the right, agree=0.710, adj=0.027, (0 split)

Node number 6: 406 observations,    complexity param=0.03920469848
  mean=0.3251231527, MSE=0.2194180883 
  left son=12 (348 obs) right son=13 (58 obs)
  Primary splits:
      TOTAL_WORKERS               < -0.02321245675 to the left,  improve=0.1663533142, (0 missing)
      AMENDED_PETITION            < 0.02264325911  to the left,  improve=0.1188556974, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < 0.5            to the left,  improve=0.1187260377, (0 missing)
      CHANGE_EMPLOYER             < 0.003899026122 to the left,  improve=0.1008077137, (0 missing)
      NEW_EMPLOYMENT              < 0.09510580444  to the left,  improve=0.1002063556, (0 missing)
  Surrogate splits:
      NEW_EMPLOYMENT             < 0.09510580444  to the left,  agree=0.966, adj=0.759, (0 split)
      AMENDED_PETITION           < 0.02264325911  to the left,  agree=0.943, adj=0.603, (0 split)
      CHANGE_EMPLOYER            < 0.003899026122 to the left,  agree=0.941, adj=0.586, (0 split)
      CONTINUED_EMPLOYMENT       < -0.03505805807 to the left,  agree=0.938, adj=0.569, (0 split)
      CHANGE_PREVIOUS_EMPLOYMENT < 0.1024402574   to the left,  agree=0.924, adj=0.466, (0 split)

Node number 7: 955 observations,    complexity param=0.05316611855
  mean=0.6534031414, MSE=0.2264674762 
  left son=14 (119 obs) right son=15 (836 obs)
  Primary splits:
      PW_WAGE_LEVEL.N.A   < 0.5            to the right, improve=0.09292178488, (0 missing)
      PW_SOURCE_YEAR.2017 < 0.5            to the left,  improve=0.06941074774, (0 missing)
      PW_SOURCE_YEAR.2016 < 0.5            to the right, improve=0.06630019428, (0 missing)
      SUPPORT_H1B         < 0.5            to the left,  improve=0.04742934655, (0 missing)
      H1B_DEPENDENT       < 0.5            to the left,  improve=0.04580658588, (0 missing)
  Surrogate splits:
      PW_SOURCE_YEAR.2016   < 0.5            to the right, agree=0.882, adj=0.050, (0 split)
      PREVAILING_WAGE       < -1.990981482   to the left,  agree=0.880, adj=0.034, (0 split)
      PW_SOURCE_YEAR.2017   < 0.5            to the left,  agree=0.880, adj=0.034, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.655457891   to the left,  agree=0.879, adj=0.025, (0 split)
      NAICS_CODE.92         < 0.5            to the right, agree=0.877, adj=0.017, (0 split)

Node number 12: 348 observations,    complexity param=0.01432447382
  mean=0.2471264368, MSE=0.186054961 
  left son=24 (222 obs) right son=25 (126 obs)
  Primary splits:
      AGENT_REPRESENTING_EMPLOYER < 0.5            to the left,  improve=0.08362766662, (0 missing)
      WAGE_DIFF                   < -0.07862971755 to the left,  improve=0.08061671936, (0 missing)
      SUPPORT_H1B                 < 0.5            to the left,  improve=0.03884038324, (0 missing)
      PW_WAGE_LEVEL.Level.III     < 0.5            to the left,  improve=0.03142254662, (0 missing)
      SOC_CODE.19                 < 0.5            to the left,  improve=0.03097555922, (0 missing)
  Surrogate splits:
      PREVAILING_WAGE        < 0.3238590621   to the left,  agree=0.710, adj=0.198, (0 split)
      WAGE_RATE_OF_PAY_FROM  < 0.05038418641  to the left,  agree=0.704, adj=0.183, (0 split)
      PW_WAGE_LEVEL.Level.IV < 0.5            to the left,  agree=0.678, adj=0.111, (0 split)
      WAGE_DIFF              < 0.6349385089   to the left,  agree=0.664, adj=0.071, (0 split)
      SOC_CODE.17            < 0.5            to the left,  agree=0.658, adj=0.056, (0 split)

Node number 13: 58 observations
  mean=0.7931034483, MSE=0.1640903686 

Node number 14: 119 observations
  mean=0.268907563, MSE=0.1965962856 

Node number 15: 836 observations,    complexity param=0.01820957666
  mean=0.7081339713, MSE=0.20668025 
  left son=30 (260 obs) right son=31 (576 obs)
  Primary splits:
      SOC_CODE.15         < 0.5            to the left,  improve=0.03983697872, (0 missing)
      PW_SOURCE_YEAR.2017 < 0.5            to the left,  improve=0.03906430542, (0 missing)
      PW_SOURCE_YEAR.2016 < 0.5            to the right, improve=0.03694177618, (0 missing)
      VISA_CLASS.H.1B     < 0.5            to the left,  improve=0.03229427301, (0 missing)
      SUPPORT_H1B         < 0.5            to the left,  improve=0.02844498405, (0 missing)
  Surrogate splits:
      SOC_CODE.13     < 0.5            to the right, agree=0.774, adj=0.273, (0 split)
      SOC_CODE.17     < 0.5            to the right, agree=0.755, adj=0.212, (0 split)
      PREVAILING_WAGE < -0.9819611606  to the left,  agree=0.732, adj=0.138, (0 split)
      SOC_CODE.29     < 0.5            to the right, agree=0.731, adj=0.135, (0 split)
      VISA_CLASS.H.1B < 0.5            to the left,  agree=0.725, adj=0.115, (0 split)

Node number 24: 222 observations,    complexity param=0.01394173629
  mean=0.1531531532, MSE=0.1296972648 
  left son=48 (170 obs) right son=49 (52 obs)
  Primary splits:
      WAGE_DIFF     < -0.07862971755 to the left,  improve=0.14822041410, (0 missing)
      SUPPORT_H1B   < 0.5            to the left,  improve=0.13249031130, (0 missing)
      H1B_DEPENDENT < 0.5            to the left,  improve=0.10246729100, (0 missing)
      SOC_CODE.19   < 0.5            to the left,  improve=0.09183818883, (0 missing)
      NAICS_CODE.61 < 0.5            to the left,  improve=0.08441332225, (0 missing)
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < 0.5961020628   to the left,  agree=0.793, adj=0.115, (0 split)
      SOC_CODE.19           < 0.5            to the left,  agree=0.784, adj=0.077, (0 split)
      NAICS_CODE.61         < 0.5            to the left,  agree=0.784, adj=0.077, (0 split)
      NAICS_CODE.51         < 0.5            to the left,  agree=0.779, adj=0.058, (0 split)
      FULL_TIME_POSITION    < 0.5            to the right, agree=0.775, adj=0.038, (0 split)

Node number 25: 126 observations
  mean=0.4126984127, MSE=0.2423784329

Node number 30: 260 observations,    complexity param=0.01012415326
  mean=0.5730769231, MSE=0.2446597633
  left son=60 (41 obs) right son=61 (219 obs)
  Primary splits:
      PW_SOURCE.OES               < 0.5            to the left,  improve=0.06016093971, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < 0.5            to the left,  improve=0.05225306146, (0 missing)
      PW_SOURCE.Other             < 0.5            to the right, improve=0.05079971404, (0 missing)
      PW_SOURCE_YEAR.2016         < 0.5            to the right, improve=0.04743561487, (0 missing)
      PW_SOURCE_YEAR.2017         < 0.5            to the left,  improve=0.03780856095, (0 missing)
  Surrogate splits:
      PW_SOURCE.Other           < 0.5            to the right, agree=0.992, adj=0.951, (0 split)
      NEW_CONCURRENT_EMPLOYMENT < 0.5            to the right, agree=0.854, adj=0.073, (0 split)
      WAGE_RATE_OF_PAY_FROM     < 4.330906151    to the right, agree=0.854, adj=0.073, (0 split)
      PREVAILING_WAGE           < 3.87100485     to the right, agree=0.846, adj=0.024, (0 split)
      WAGE_DIFF                 < 3.559195063    to the right, agree=0.846, adj=0.024, (0 split)

Node number 31: 576 observations
  mean=0.7690972222, MSE=0.177586685

Node number 48: 170 observations,    complexity param=0.01394173629
  mean=0.07647058824, MSE=0.07062283737
  left son=96 (161 obs) right son=97 (9 obs)
  Primary splits:
      SUPPORT_H1B            < 0.5            to the left,  improve=0.52243331920, (0 missing)
      H1B_DEPENDENT          < 0.5            to the left,  improve=0.41490468830, (0 missing)
      VISA_CLASS.H.1B        < 0.5            to the left,  improve=0.01632726294, (0 missing)
      PW_WAGE_LEVEL.Level.II < 0.5            to the right, improve=0.01161399610, (0 missing)
      WAGE_RATE_OF_PAY_FROM  < -0.4668398269  to the right, improve=0.01161399610, (0 missing)
  Surrogate splits:
      H1B_DEPENDENT < 0.5            to the left,  agree=0.988, adj=0.778, (0 split)

Node number 49: 52 observations
  mean=0.4038461538, MSE=0.2407544379

Node number 60: 41 observations
  mean=0.2926829268, MSE=0.2070196312

Node number 61: 219 observations
  mean=0.6255707763, MSE=0.2342319802

Node number 96: 161 observations
  mean=0.03105590062, MSE=0.03009143166
```

```
Node number 97: 9 observations
  mean=0.8888888889, MSE=0.0987654321

                       [,1]         [,2]
Accuracy       0.7566137566 0.7447089947
Sensitivity    0.8359788360 0.8650793651
Specificity    0.6772486772 0.6243386243
Pos Pred Value 0.7214611872 0.6972281450
Neg Pred Value 0.8050314465 0.8222996516
```



```
> tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
[1] 0.753968254 0.500000000
Call:
rpart(formula = CASE_STATUS ~ ., data = trdf)
  n= 1512

             CP nsplit    rel error       xerror            xstd
1 0.10769230769      0 1.0000000000 1.0014404835 0.0004465542337
2 0.08866790922      1 0.8923076923 0.8959017095 0.0113552975997
3 0.05569916224      2 0.8036397831 0.8072429217 0.0151077290727
4 0.04667693611      3 0.7479406209 0.7815427849 0.0186613631637
5 0.02525644633      4 0.7012636847 0.7100620900 0.0207644048150
6 0.01962564831      5 0.6760072384 0.7055492565 0.0210213642693
7 0.01286784661      6 0.6563815901 0.6882542562 0.0215509952233
8 0.01000000000      7 0.6435137435 0.6871994930 0.0222666294761

Variable importance
                  WAGE_DIFF                    H1B_DEPENDENT
SUPPORT_H1B                           AGENT_REPRESENTING_EMPLOYER
PW_SOURCE.OES              PW_SOURCE.Other
                         22                               14
14                              9                          8
7
          PW_WAGE_LEVEL.N.A                   VISA_CLASS.H.1B
NEW_EMPLOYMENT                                PREVAILING_WAGE
PW_WAGE_LEVEL.Level.III                SOC_CODE.15
                          4                                3
3                              2                           2
2
           AMENDED_PETITION             PW_WAGE_LEVEL.Level.II
PW_SOURCE_YEAR.2017                             NAICS_CODE.54
WAGE_RATE_OF_PAY_FROM         PW_SOURCE_YEAR.2016
                          2                                2
1                              1                           1
1
                SOC_CODE.19                      SOC_CODE.13
SOC_CODE.27
                          1                                1
1

Node   number   1:   1512   observations,          complexity
param=0.1076923077
  mean=0.5, MSE=0.25
  left son=2 (147 obs) right son=3 (1365 obs)
  Primary splits:
      WAGE_DIFF                 <  -0.3046590705     to  the  left,
improve=0.10769230770, (0 missing)
      H1B_DEPENDENT       <  0.5                    to  the  left,
improve=0.08675346957, (0 missing)
      SUPPORT_H1B         <  0.5                    to  the  left,
improve=0.08245383201, (0 missing)
```

```
      NEW_EMPLOYMENT    < -0.04780941156  to the right, improve=0.06142357501, (0 missing)
      PW_WAGE_LEVEL.N.A < 0.5             to the right, improve=0.05095766723, (0 missing)
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < -1.905902614    to the left,  agree=0.904, adj=0.014, (0 split)
      SOC_CODE.37           < 0.5             to the right, agree=0.904, adj=0.014, (0 split)
      PREVAILING_WAGE       < 4.386725697     to the right, agree=0.903, adj=0.007, (0 split)

Node number 2: 147 observations
  mean=0, MSE=0 

Node number 3: 1365 observations,    complexity param=0.08866790922
  mean=0.5538461538, MSE=0.2471005917 
  left son=6 (1006 obs) right son=7 (359 obs)
  Primary splits:
      H1B_DEPENDENT     < 0.5             to the left,  improve=0.09936920861, (0 missing)
      SUPPORT_H1B       < 0.5             to the left,  improve=0.09184731168, (0 missing)
      NEW_EMPLOYMENT    < -0.04780941156  to the right, improve=0.07490112370, (0 missing)
      PW_WAGE_LEVEL.N.A < 0.5             to the right, improve=0.06730572004, (0 missing)
      VISA_CLASS.H.1B   < 0.5             to the left,  improve=0.05668935992, (0 missing)
  Surrogate splits:
      SUPPORT_H1B      < 0.5             to the left,  agree=0.985, adj=0.944, (0 split)
      AMENDED_PETITION < -0.003659157996 to the left,  agree=0.766, adj=0.109, (0 split)

Node number 6: 1006 observations,    complexity param=0.05569916224
  mean=0.4602385686, MSE=0.2484190286 
  left son=12 (336 obs) right son=13 (670 obs)
  Primary splits:
      AGENT_REPRESENTING_EMPLOYER < 0.5             to the left,  improve=0.08424761654, (0 missing)
      NEW_EMPLOYMENT              < -0.04780941156  to the right, improve=0.06905628083, (0 missing)
      PW_SOURCE.OES               < 0.5             to the left,  improve=0.05141534721, (0 missing)
      PW_WAGE_LEVEL.Level.III     < 0.5             to the left,  improve=0.05132002690, (0 missing)
      PW_SOURCE.Other             < 0.5             to the right, improve=0.05096607489, (0 missing)
  Surrogate splits:
      NEW_EMPLOYMENT        < -0.04780941156  to the right, agree=0.734, adj=0.202, (0 split)
      PREVAILING_WAGE       < -1.044592675    to the left,  agree=0.693, adj=0.080, (0 split)
      SOC_CODE.19           < 0.5             to the right, agree=0.688, adj=0.065, (0 split)
      PW_SOURCE_YEAR.2017   < 0.5             to the left,  agree=0.683, adj=0.051, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.016098878    to the left,  agree=0.681, adj=0.045, (0 split)

Node number 7: 359 observations
  mean=0.8161559889, MSE=0.1500453907 

Node number 12: 336 observations,    complexity param=0.02525644633
  mean=0.255952381, MSE=0.1904407596 
  left son=24 (149 obs) right son=25 (187 obs)
  Primary splits:
      WAGE_DIFF        < -0.3020302372   to the left,  improve=0.14919863890, (0 missing)
      NEW_EMPLOYMENT   < -0.04780941156  to the right, improve=0.09641130563, (0 missing)
      NAICS_CODE.61    < 0.5             to the left,  improve=0.08635215947, (0 missing)
      AMENDED_PETITION < -0.003659157996 to the left,  improve=0.08385564083, (0 missing)
      SOC_CODE.19      < 0.5             to the left,  improve=0.07948155039, (0 missing)
  Surrogate splits:
      SOC_CODE.15            < 0.5             to the right, agree=0.735, adj=0.403, (0 split)
      PW_WAGE_LEVEL.Level.II < 0.5             to the right, agree=0.723, adj=0.376, (0 split)
      PREVAILING_WAGE        < -0.3749158944   to the right, agree=0.705, adj=0.336, (0 split)
      NAICS_CODE.54          < 0.5             to the right, agree=0.690, adj=0.302, (0 split)
      NEW_EMPLOYMENT         < -0.04780941156  to the right, agree=0.655, adj=0.221, (0 split)

Node number 13: 670 observations,    complexity param=0.04667693611
  mean=0.5626865672, MSE=0.2460703943
  left son=26 (165 obs) right son=27 (505 obs)
  Primary splits:
      PW_SOURCE.OES         < 0.5             to the left,  improve=0.10701877440, (0 missing)
      PW_SOURCE.Other       < 0.5             to the right, improve=0.10551874310, (0 missing)
      PW_WAGE_LEVEL.N.A     < 0.5             to the right, improve=0.08023645394, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.7417247649   to the left,  improve=0.04919732725, (0 missing)
      PW_SOURCE_YEAR.2016   < 0.5             to the right, improve=0.04574011848, (0 missing)
  Surrogate splits:
      PW_SOURCE.Other       < 0.5             to the right, agree=0.993, adj=0.970, (0 split)
      PW_WAGE_LEVEL.N.A     < 0.5             to the right, agree=0.881, adj=0.515, (0 split)
      PW_SOURCE_YEAR.2016   < 0.5             to the right, agree=0.782, adj=0.115, (0 split)
      PW_SOURCE_YEAR.2017   < 0.5             to the left,  agree=0.782, adj=0.115, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.62844559     to the left,  agree=0.766, adj=0.048, (0 split)

Node number 24: 149 observations
  mean=0.06711409396, MSE=0.06260979235

Node number 25: 187 observations,    complexity param=0.01962564831
  mean=0.4064171123, MSE=0.2412422431
  left son=50 (63 obs) right son=51 (124 obs)
  Primary splits:
      VISA_CLASS.H.1B     < 0.5             to the left,  improve=0.16444506600, (0 missing)
      PW_SOURCE_YEAR.2016 < 0.5             to the right, improve=0.07726158176, (0 missing)
      SOC_CODE.15         < 0.5             to the left,  improve=0.06901269498, (0 missing)
      NAICS_CODE.61       < 0.5             to the left,  improve=0.06693483306, (0 missing)
      PW_SOURCE.Other     < 0.5             to the right, improve=0.06433964329, (0 missing)
  Surrogate splits:
      SOC_CODE.13                < 0.5             to the right, agree=0.722, adj=0.175, (0 split)
      SOC_CODE.27                < 0.5             to the right, agree=0.722, adj=0.175, (0 split)
      VISA_CLASS.H.1B1.Singapore < 0.5             to the right, agree=0.690, adj=0.079, (0 split)
      WAGE_RATE_OF_PAY_FROM      < -1.30148446     to the left,  agree=0.684, adj=0.063, (0 split)
      NAICS_CODE.31              < 0.5             to the right, agree=0.684, adj=0.063, (0 split)

Node number 26: 165 observations
  mean=0.2787878788, MSE=0.2010651974

Node number 27: 505 observations,    complexity param=0.01286784661
  mean=0.6554455446, MSE=0.2258366827
  left son=54 (393 obs) right son=55 (112 obs)
  Primary splits:
      PW_WAGE_LEVEL.Level.III < 0.5             to the left,  improve=0.04264929053, (0 missing)
      WAGE_RATE_OF_PAY_FROM   < -1.131164232    to the left,  improve=0.04076601403, (0 missing)
      PREVAILING_WAGE         < -1.222613316    to the left,  improve=0.03697594220, (0 missing)
      PW_WAGE_LEVEL.N.A       < 0.5             to the right, improve=0.03451751947, (0 missing)
      VISA_CLASS.H.1B         < 0.5             to the left,  improve=0.02953674263, (0 missing)
  Surrogate splits:
      CHANGE_PREVIOUS_EMPLOYMENT < 8.047647424     to the left,  agree=0.782, adj=0.018, (0 split)
      NAICS_CODE.71              < 0.5             to the left,  agree=0.782, adj=0.018, (0 split)

Node number 50: 63 observations
  mean=0.126984127, MSE=0.1108591585

Node number 51: 124 observations
  mean=0.5483870968, MSE=0.2476586889

Node number 54: 393 observations
  mean=0.6030534351, MSE=0.2393799895

Node number 55: 112 observations
  mean=0.8392857143, MSE=0.1348852041

                       [,1]         [,2]
Accuracy       0.7539682540 0.7513227513
Sensitivity    0.8862433862 0.9126984127
Specificity    0.6216931217 0.5899470899
Pos Pred Value 0.7008368201 0.6900000000
Neg Pred Value 0.8453237410 0.8710937500
```

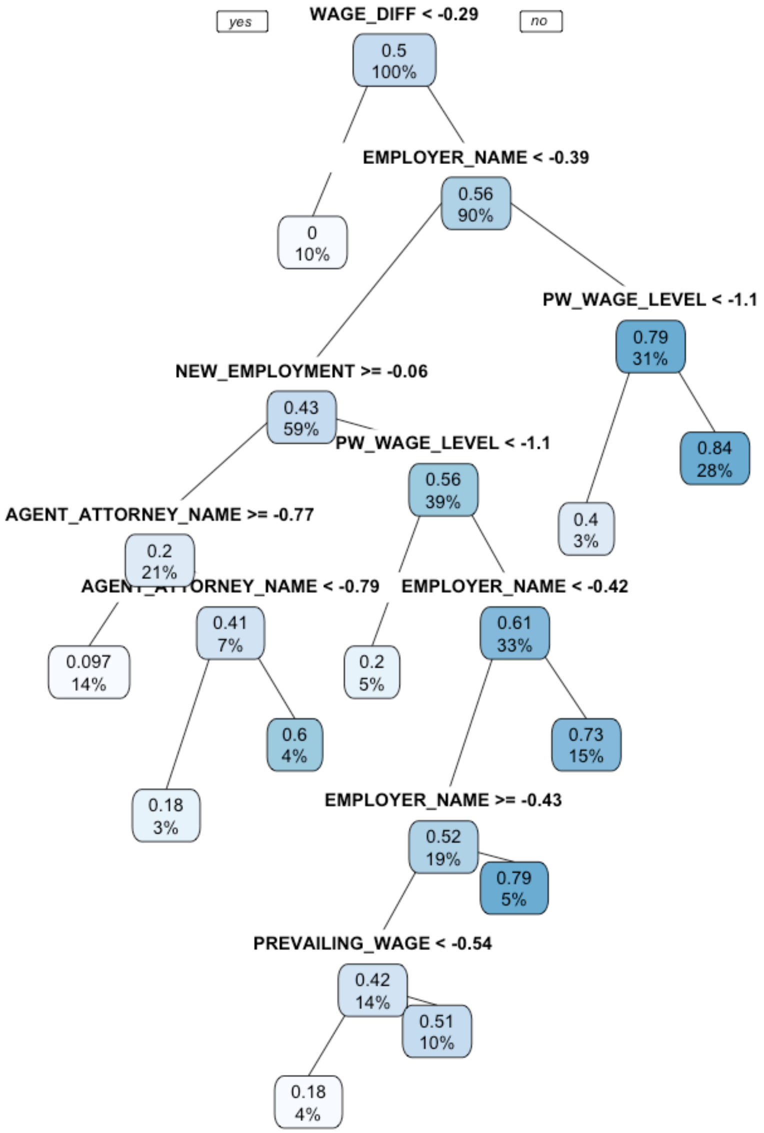


(2) Using frequency values for categorical variables

```
> tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
[1] 0.7764550265 0.5000000000
Call:
rpart(formula = CASE_STATUS ~ ., data = trdf)
  n= 1512

             CP nsplit    rel error        xerror            xstd
1 0.11751662971      0 1.0000000000 1.0020403497 0.0005314641173
2 0.09424011553      1 0.8824833703 0.8858496954 0.0096666600901
3 0.06265397247      2 0.7882432548 0.8076105517 0.0155994142098
4 0.04963191633      3 0.7255892823 0.7656022359 0.0187988913901
5 0.03852757243      4 0.6759573660 0.7011180686 0.0203190257391
6 0.01007514378      6 0.5989022211 0.6266152979 0.0232724143050
7 0.01000000000      7 0.5888270773 0.6352104137 0.0235853584330

Variable importance
                   WAGE_DIFF                  AGENT_ATTORNEY_NAME
H1B_DEPENDENT                                         SUPPORT_H1B
AGENT_REPRESENTING_EMPLOYER                 EMPLOYER_NAME
                          18                                   16
15                              14                             10
8
                   PW_SOURCE                         PW_WAGE_LEVEL
PREVAILING_WAGE                              WAGE_RATE_OF_PAY_FROM
NEW_EMPLOYMENT             AMENDED_PETITION
                           6                                    3
3                               2                               2
2
              PW_SOURCE_YEAR                    SOC_CODE
                           1                           1

Node   number   1:   1512   observations,           complexity
param=0.1175166297
  mean=0.5, MSE=0.25
  left son=2 (159 obs) right son=3 (1353 obs)
  Primary splits:
      WAGE_DIFF              <   -0.2849143498    to   the   left,
improve=0.11751662970, (0 missing)
      H1B_DEPENDENT      <   -0.553106408        to   the   right,
improve=0.08779194662, (0 missing)
      SUPPORT_H1B          <   -0.5550827383    to   the   right,
improve=0.08704420541, (0 missing)
      EMPLOYER_NAME      <   -0.37312401         to   the   left,
improve=0.07531538166, (0 missing)
      NEW_EMPLOYMENT    <    -0.06827305677    to    the    right,
improve=0.05985907180, (0 missing)
  Surrogate splits:
      SOC_CODE       < -1.365253184   to the left,  agree=0.896,
adj=0.013, (0 split)
```

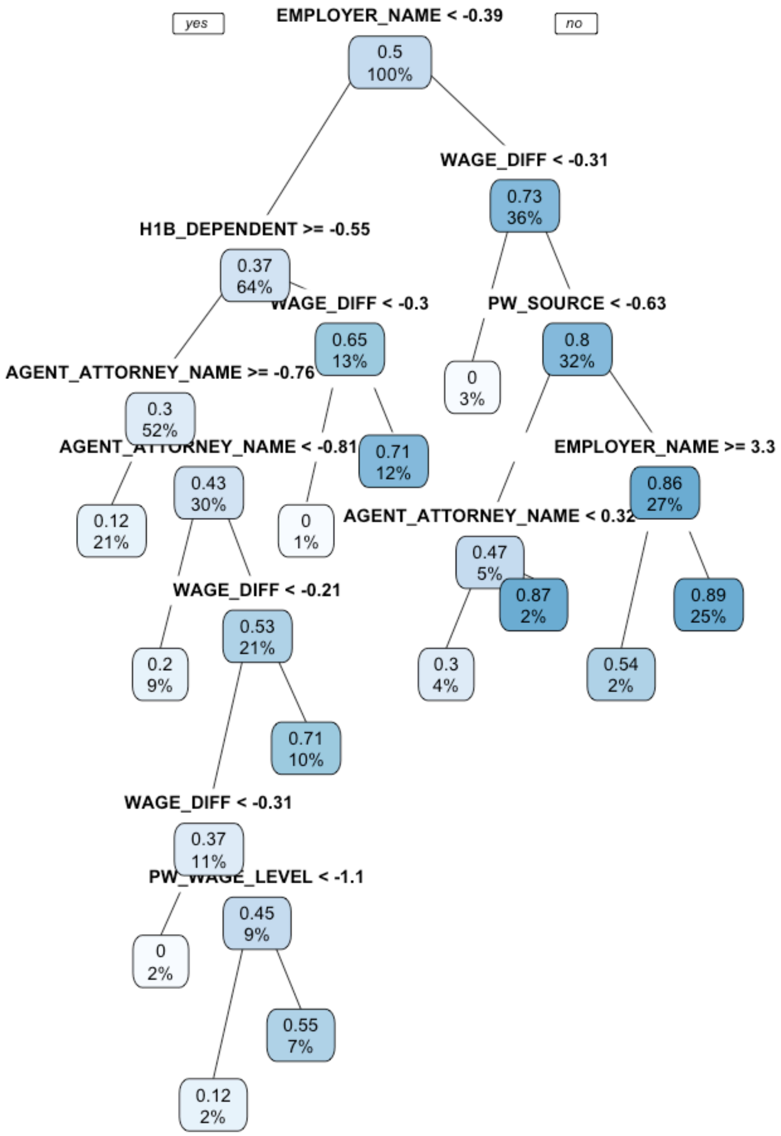
```
      CHANGE_EMPLOYER < 10.84468214    to the right, agree=0.896, adj=0.013, (0 split)
      PREVAILING_WAGE < 5.01118414     to the right, agree=0.896, adj=0.006, (0 split)

Node number 2: 159 observations
  mean=0, MSE=0 

Node number 3: 1353 observations,    complexity param=0.09424011553
  mean=0.5587583149, MSE=0.2465474604 
  left son=6 (1001 obs) right son=7 (352 obs)
  Primary splits:
      H1B_DEPENDENT  < -0.553106408   to the right, improve=0.10678967870, (0 missing)
      SUPPORT_H1B    < -0.5550827383  to the right, improve=0.10458372890, (0 missing)
      EMPLOYER_NAME  < -0.3746827698  to the left,  improve=0.09394038273, (0 missing)
      NEW_EMPLOYMENT < -0.06827305677 to the right, improve=0.07880448930, (0 missing)
      PW_WAGE_LEVEL  < -1.142071348   to the left,  improve=0.07115664323, (0 missing)
  Surrogate splits:
      SUPPORT_H1B      < -0.5550827383  to the right, agree=0.991, adj=0.966, (0 split)
      AMENDED_PETITION < 0.03783364027  to the left,  agree=0.761, adj=0.082, (0 split)
      PW_SOURCE_YEAR   < -4.353678328   to the right, agree=0.741, adj=0.006, (0 split)
      CHANGE_EMPLOYER  < 1.105905772    to the left,  agree=0.741, adj=0.003, (0 split)

Node number 6: 1001 observations,    complexity param=0.06265397247
  mean=0.4625374625, MSE=0.2485965583 
  left son=12 (317 obs) right son=13 (684 obs)
  Primary splits:
      AGENT_REPRESENTING_EMPLOYER < -0.2552031106  to the left,  improve=0.09517244411, (0 missing)
      AGENT_ATTORNEY_NAME         < 0.3339908615   to the right, improve=0.09517244411, (0 missing)
      EMPLOYER_NAME               < -0.3746827698  to the left,  improve=0.08339986742, (0 missing)
      NEW_EMPLOYMENT              < -0.06827305677 to the right, improve=0.07032208163, (0 missing)
      PW_WAGE_LEVEL               < -1.142071348   to the left,  improve=0.05620172723, (0 missing)
  Surrogate splits:
      AGENT_ATTORNEY_NAME < 0.3339908615   to the right, agree=1.000, adj=1.000, (0 split)
      NEW_EMPLOYMENT      < -0.06827305677 to the right, agree=0.744, adj=0.192, (0 split)
      PREVAILING_WAGE     < -1.058994038   to the left,  agree=0.701, adj=0.057, (0 split)
      PW_SOURCE_YEAR      < -2.039008986   to the left,  agree=0.697, adj=0.044, (0 split)
      SOC_CODE            < -1.364860696   to the left,  agree=0.694, adj=0.035, (0 split)

Node number 7: 352 observations
  mean=0.8323863636, MSE=0.1395193053 

Node number 12: 317 observations,    complexity param=0.04963191633
  mean=0.2365930599, MSE=0.1806167839 
  left son=24 (269 obs) right son=25 (48 obs)
  Primary splits:
      EMPLOYER_NAME    < -0.3746827698  to the left,  improve=0.32766909120, (0 missing)
      WAGE_DIFF        < -0.2807026681  to the left,  improve=0.12875492530, (0 missing)
      NEW_EMPLOYMENT   < -0.06827305677 to the right, improve=0.09080559254, (0 missing)
      SOC_CODE         < -1.240628821   to the right, improve=0.06205084986, (0 missing)
      AMENDED_PETITION < 0.03783364027  to the left,  improve=0.05285954735, (0 missing)
  Surrogate splits:
      AMENDED_PETITION < 0.03783364027  to the left,  agree=0.858, adj=0.062, (0 split)
      TOTAL_WORKERS    < 0.02172660431  to the left,  agree=0.852, adj=0.021, (0 split)
```

```
Node number 13: 684 observations,    complexity param=0.03852757243
  mean=0.567251462, MSE=0.2454772409 
  left son=26 (166 obs) right son=27 (518 obs)
  Primary splits:
      PW_SOURCE             < -0.6217253388  to the left,  improve=0.08422271315, (0 missing)
      AGENT_ATTORNEY_NAME   < -0.798170725   to the left,  improve=0.07802293550, (0 missing)
      PW_WAGE_LEVEL         < -1.142071348   to the left,  improve=0.06863413308, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.7069275797  to the left,  improve=0.05349398691, (0 missing)
      PREVAILING_WAGE       < -0.7806546902  to the left,  improve=0.04284547656, (0 missing)
  Surrogate splits:
      PW_WAGE_LEVEL         < -1.142071348   to the left,  agree=0.874, adj=0.482, (0 split)
      PW_SOURCE_YEAR        < -2.039008986   to the left,  agree=0.781, adj=0.096, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.507259605   to the left,  agree=0.768, adj=0.042, (0 split)
      PREVAILING_WAGE       < -1.778911487   to the left,  agree=0.766, adj=0.036, (0 split)
      VISA_CLASS            < -2.928316284   to the left,  agree=0.759, adj=0.006, (0 split)

Node number 24: 269 observations
  mean=0.1338289963, MSE=0.115918796 

Node number 25: 48 observations
  mean=0.8125, MSE=0.15234375 

Node number 26: 166 observations,    complexity param=0.01007514378
  mean=0.313253012, MSE=0.2151255625 
  left son=52 (41 obs) right son=53 (125 obs)
  Primary splits:
      PREVAILING_WAGE       < -0.7006264828  to the left,  improve=0.10664560090, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.7365002132  to the left,  improve=0.08957493413, (0 missing)
      SOC_CODE              < -0.1536980364  to the left,  improve=0.07977609245, (0 missing)
      EMPLOYER_NAME         < -0.4152105253  to the right, improve=0.06295336041, (0 missing)
      VISA_CLASS            < -1.230523379   to the left,  improve=0.04865497076, (0 missing)
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < -0.7365002132  to the left,  agree=0.976, adj=0.902, (0 split)
      SOC_CODE              < -1.282830652   to the left,  agree=0.789, adj=0.146, (0 split)
      VISA_CLASS            < -1.230523379   to the left,  agree=0.765, adj=0.049, (0 split)
      NAICS_CODE            < -1.30640459    to the left,  agree=0.765, adj=0.049, (0 split)
      FULL_TIME_POSITION    < -2.733183463   to the left,  agree=0.765, adj=0.049, (0 split)

Node number 27: 518 observations,    complexity param=0.03852757243
  mean=0.6486486486, MSE=0.2279035793 
  left son=54 (86 obs) right son=55 (432 obs)
  Primary splits:
      AGENT_ATTORNEY_NAME   < -0.7980058731  to the left,  improve=0.12693600030, (0 missing)
      EMPLOYER_NAME         < -0.4058579664  to the left,  improve=0.05595392415, (0 missing)
      PW_WAGE_LEVEL         < -0.7242915889  to the left,  improve=0.04092636389, (0 missing)
      NAICS_CODE            < -1.297741549   to the left,  improve=0.03984661172, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.7069275797  to the left,  improve=0.02533107890, (0 missing)
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < -1.085847257   to the left,  agree=0.847, adj=0.081, (0 split)
      NAICS_CODE            < -1.302910769   to the left,  agree=0.844, adj=0.058, (0 split)
      SOC_CODE              < -1.359085512   to the left,  agree=0.842, adj=0.047, (0 split)
      PW_SOURCE_YEAR        < -2.039008986   to the left,  agree=0.842, adj=0.047, (0 split)
      PREVAILING_WAGE       < -1.596712066   to the left,  agree=0.840, adj=0.035, (0 split)

Node number 52: 41 observations
  mean=0.0487804878, MSE=0.04640095181 

Node number 53: 125 observations
  mean=0.4, MSE=0.24 

Node number 54: 86 observations
  mean=0.2674418605, MSE=0.1959167117 

Node number 55: 432 observations
  mean=0.724537037, MSE=0.199583119 

                       [,1]         [,2]
Accuracy       0.7764550265 0.7910052910
Sensitivity    0.8306878307 0.8412698413
Specificity    0.7222222222 0.7407407407
Pos Pred Value 0.7494033413 0.7644230769
Neg Pred Value 0.8100890208 0.8235294118
```



```
> tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
[1] 0.7777777778 0.5500000000
Call:
rpart(formula = CASE_STATUS ~ ., data = trdf)
  n= 1512

              CP nsplit    rel error       xerror          xstd
1  0.11094783248      0 1.0000000000 1.0034394415 0.0006876593666
2  0.10517019724      1 0.8890521675 0.9066176526 0.0119336114301
3  0.06731789190      2 0.7838819703 0.7937842045 0.0165792730029
4  0.03188192287      3 0.7165640784 0.7239316152 0.0195401705483
5  0.02091934157      4 0.6846821555 0.7013880516 0.0211372657188
6  0.01762960305      5 0.6637628140 0.6948331319 0.0216511700038
7  0.01755609679      6 0.6461332109 0.6745801097 0.0223707130502
8  0.01247575460      8 0.6110210173 0.6506875500 0.0225483895372
9  0.01187927712      9 0.5985452627 0.6417225945 0.0232383383701
10 0.01000000000     10 0.5866659856 0.6338946384 0.0234816352251

Variable importance
              EMPLOYER_NAME                              WAGE_DIFF
NEW_EMPLOYMENT                                      PW_WAGE_LEVEL
AGENT_ATTORNEY_NAME AGENT_REPRESENTING_EMPLOYER
                         26                                    19
12                             9                                8
5
            PREVAILING_WAGE                         TOTAL_WORKERS
WAGE_RATE_OF_PAY_FROM                                    SOC_CODE
CONTINUED_EMPLOYMENT                VISA_CLASS
                          5                                     4
3                           2                                   1
1
           AMENDED_PETITION                         H1B_DEPENDENT
SUPPORT_H1B                                       CHANGE_EMPLOYER
NAICS_CODE
                          1                                     1
1                          1                           1

Node   number   1:   1512   observations,          complexity
param=0.1109478325
  mean=0.5, MSE=0.25
  left son=2 (151 obs) right son=3 (1361 obs)
  Primary splits:
      WAGE_DIFF              <   -0.2881880086     to   the   left,
improve=0.11094783250, (0 missing)
      EMPLOYER_NAME      <    -0.3785190158      to    the    left,
improve=0.10050181180, (0 missing)
      NEW_EMPLOYMENT    <    -0.06018574666    to    the    right,
improve=0.07376857572, (0 missing)
      SUPPORT_H1B          <   -0.6276629288     to   the   right,
improve=0.07216806690, (0 missing)
      H1B_DEPENDENT      <    -0.6125790915      to    the    right,
improve=0.06873704908, (0 missing)
```

```
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < -1.772465109   to the left,  agree=0.901, adj=0.013, (0 split)
      SOC_CODE              < -1.374152137   to the left,  agree=0.901, adj=0.007, (0 split)
      NAICS_CODE            < -1.311193169   to the left,  agree=0.901, adj=0.007, (0 split)

Node number 2: 151 observations
  mean=0, MSE=0 

Node number 3: 1361 observations,    complexity param=0.1051701972
  mean=0.5554739162, MSE=0.2469226446 
  left son=6 (897 obs) right son=7 (464 obs)
  Primary splits:
      EMPLOYER_NAME  < -0.3862979456  to the left,  improve=0.11829474250, (0 missing)
      NEW_EMPLOYMENT < -0.06018574666 to the right, improve=0.09135683693, (0 missing)
      SUPPORT_H1B    < -0.6276629288  to the right, improve=0.07804347629, (0 missing)
      H1B_DEPENDENT  < -0.6125790915  to the right, improve=0.07576642975, (0 missing)
      PW_WAGE_LEVEL  < -1.128559757   to the left,  improve=0.06556791018, (0 missing)
  Surrogate splits:
      TOTAL_WORKERS        < -0.02321245675 to the left,  agree=0.733, adj=0.218, (0 split)
      CONTINUED_EMPLOYMENT < 0.2424696058   to the left,  agree=0.685, adj=0.075, (0 split)
      NEW_EMPLOYMENT       < 0.09510580444  to the left,  agree=0.681, adj=0.065, (0 split)
      AMENDED_PETITION     < 0.02264325911  to the left,  agree=0.678, adj=0.056, (0 split)
      CHANGE_EMPLOYER      < 0.3723569947   to the left,  agree=0.676, adj=0.050, (0 split)

Node number 6: 897 observations,    complexity param=0.0673178919
  mean=0.4325529543, MSE=0.245450896 
  left son=12 (312 obs) right son=13 (585 obs)
  Primary splits:
      NEW_EMPLOYMENT              < -0.06018574666 to the right, improve=0.11557535660, (0 missing)
      AGENT_ATTORNEY_NAME         < 0.284253762    to the right, improve=0.08990949707, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.2683489623  to the left,  improve=0.08990949707, (0 missing)
      SUPPORT_H1B                 < -0.6276629288  to the right, improve=0.08308498243, (0 missing)
      H1B_DEPENDENT               < -0.6125790915  to the right, improve=0.08256065263, (0 missing)
  Surrogate splits:
      EMPLOYER_NAME               < -0.3971884472  to the right, agree=0.720, adj=0.196, (0 split)
      AGENT_REPRESENTING_EMPLOYER < -0.2683489623  to the left,  agree=0.709, adj=0.163, (0 split)
      AGENT_ATTORNEY_NAME         < 0.284253762    to the right, agree=0.709, adj=0.163, (0 split)
      VISA_CLASS                  < -1.16321874    to the left,  agree=0.690, adj=0.109, (0 split)
      PREVAILING_WAGE             < -1.184303608   to the left,  agree=0.687, adj=0.099, (0 split)

Node number 7: 464 observations,    complexity param=0.02091934157
  mean=0.7931034483, MSE=0.1640903686 
  left son=14 (47 obs) right son=15 (417 obs)
  Primary splits:
      PW_WAGE_LEVEL       < -1.128559757   to the left,  improve=0.10385770940, (0 missing)
      PW_SOURCE           < -0.5992121319  to the left,  improve=0.06838423797, (0 missing)
      AGENT_ATTORNEY_NAME < -0.7629612096  to the left,  improve=0.03480777648, (0 missing)
      SUPPORT_H1B         < -0.6276629288  to the right, improve=0.03066752724, (0 missing)
      H1B_DEPENDENT       < -0.6125790915  to the right, improve=0.02750723602, (0 missing)
  Surrogate splits:
      PW_SOURCE           < -0.5992121319  to the left,  agree=0.909, adj=0.106, (0 split)
      PW_SOURCE_YEAR      < -1.937709229   to the left,  agree=0.905, adj=0.064, (0 split)
      AGENT_ATTORNEY_NAME < -0.7874284629  to the left,  agree=0.903, adj=0.043, (0 split)

Node number 12: 312 observations,    complexity param=0.01762960305
  mean=0.2019230769, MSE=0.1611501479 
  left son=24 (206 obs) right son=25 (106 obs)
  Primary splits:
      AGENT_ATTORNEY_NAME         < -0.7680034476  to the right, improve=0.13254063020, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.2683489623  to the left,  improve=0.12948665290, (0 missing)
      EMPLOYER_NAME               < -0.3971884472  to the right, improve=0.07343660356, (0 missing)
      WAGE_DIFF                   < -0.07862971755 to the left,  improve=0.06599748333, (0 missing)
      TOTAL_WORKERS               < -0.02321245675 to the left,  improve=0.05361127048, (0 missing)
  Surrogate splits:
      AGENT_REPRESENTING_EMPLOYER < -0.2683489623  to the left,  agree=0.997, adj=0.991, (0 split)
      PREVAILING_WAGE             < 0.3238590621   to the left,  agree=0.737, adj=0.226, (0 split)
      WAGE_RATE_OF_PAY_FROM       < 0.3836889588   to the left,  agree=0.737, adj=0.226, (0 split)
      WAGE_DIFF                   < -0.06387737084 to the left,  agree=0.686, adj=0.075, (0 split)
      TOTAL_WORKERS               < -0.02321245675 to the left,  agree=0.679, adj=0.057, (0 split)

Node number 13: 585 observations,    complexity param=0.03188192287
  mean=0.5555555556, MSE=0.2469135802 
  left son=26 (81 obs) right son=27 (504 obs)
  Primary splits:
      PW_WAGE_LEVEL         < -1.128559757   to the left,  improve=0.08343253968, (0 missing)
      PW_SOURCE_YEAR        < -1.937709229   to the left,  improve=0.06710893855, (0 missing)
      H1B_DEPENDENT         < -0.6125790915  to the right, improve=0.05457215131, (0 missing)
      SUPPORT_H1B           < -0.6276629288  to the right, improve=0.05400073529, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.8388327755  to the left,  improve=0.05111414993, (0 missing)
  Surrogate splits:
      PREVAILING_WAGE       < -1.990981482   to the left,  agree=0.868, adj=0.049, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.655457891   to the left,  agree=0.867, adj=0.037, (0 split)
      NAICS_CODE            < -1.309172057   to the left,  agree=0.865, adj=0.025, (0 split)
      PW_SOURCE_YEAR        < -1.937709229   to the left,  agree=0.865, adj=0.025, (0 split)
      SOC_CODE              < -1.371230089   to the left,  agree=0.863, adj=0.012, (0 split)

Node number 14: 47 observations
  mean=0.4042553191, MSE=0.2408329561 

Node number 15: 417 observations
  mean=0.8369304556, MSE=0.1364778681 

Node number 24: 206 observations
  mean=0.09708737864, MSE=0.08766141955 

Node number 25: 106 observations,    complexity param=0.01187927712
  mean=0.4056603774, MSE=0.2411000356 
  left son=50 (49 obs) right son=51 (57 obs)
  Primary splits:
      AGENT_ATTORNEY_NAME   < -0.7873044734  to the left,  improve=0.17570279650, (0 missing)
      PW_WAGE_LEVEL         < -0.9422280658  to the left,  improve=0.06171414156, (0 missing)
      TOTAL_WORKERS         < -0.08056087931 to the left,  improve=0.05328849531, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.9884693752  to the left,  improve=0.04161521764, (0 missing)
      EMPLOYER_NAME         < -0.4267483801  to the right, improve=0.03581029938, (0 missing)
  Surrogate splits:
      EMPLOYER_NAME         < -0.4251925941  to the left,  agree=0.698, adj=0.347, (0 split)
      WAGE_RATE_OF_PAY_FROM < -0.6197545235  to the left,  agree=0.670, adj=0.286, (0 split)
      NAICS_CODE            < -1.241658745   to the left,  agree=0.642, adj=0.224, (0 split)
      PREVAILING_WAGE       < -1.145584865   to the left,  agree=0.642, adj=0.224, (0 split)
      SOC_CODE              < -1.325338951   to the left,  agree=0.585, adj=0.102, (0 split)

Node number 26: 81 observations
  mean=0.1975308642, MSE=0.1585124219

Node number 27: 504 observations,    complexity param=0.01755609679
  mean=0.6130952381, MSE=0.2372094671
  left son=54 (280 obs) right son=55 (224 obs)
  Primary splits:
      EMPLOYER_NAME         < -0.4220810223  to the left,  improve=0.04779686333, (0 missing)
      SOC_CODE              < -1.29192271    to the left,  improve=0.04726729581, (0 missing)
      PREVAILING_WAGE       < -0.8009777946  to the left,  improve=0.04195186578, (0 missing)
      PW_SOURCE_YEAR        < -1.937709229   to the left,  improve=0.03888124751, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.7567524487  to the left,  improve=0.03797630690, (0 missing)
  Surrogate splits:
      H1B_DEPENDENT       < -0.6125790915  to the right, agree=0.734, adj=0.402, (0 split)
      SUPPORT_H1B         < -0.6276629288  to the right, agree=0.720, adj=0.371, (0 split)
      AGENT_ATTORNEY_NAME < -0.7840807475  to the left,  agree=0.653, adj=0.219, (0 split)
      SOC_CODE            < -0.1603034302  to the left,  agree=0.653, adj=0.219, (0 split)
      AMENDED_PETITION    < 0.02264325911  to the left,  agree=0.605, adj=0.112, (0 split)

Node number 50: 49 observations
  mean=0.1836734694, MSE=0.149937526

Node number 51: 57 observations
  mean=0.5964912281, MSE=0.2406894429

Node number 54: 280 observations,    complexity param=0.01755609679
  mean=0.5178571429, MSE=0.2496811224
  left son=108 (207 obs) right son=109 (73 obs)
  Primary splits:
      EMPLOYER_NAME               < -0.4267483801  to the right, improve=0.10811108910, (0 missing)
      AGENT_ATTORNEY_NAME         < -0.748206464   to the right, improve=0.09715518092, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.2683489623  to the left,  improve=0.09275487598, (0 missing)
      PREVAILING_WAGE             < -0.5659798268  to the left,  improve=0.07212985302, (0 missing)
      WAGE_RATE_OF_PAY_FROM       < -0.7567524487  to the left,  improve=0.06451354800, (0 missing)
  Surrogate splits:
      AGENT_ATTORNEY_NAME < -0.7878004312  to the right, agree=0.757, adj=0.068, (0 split)
      NAICS_CODE          < -1.308028802   to the right, agree=0.743, adj=0.014, (0 split)

Node number 55: 224 observations
  mean=0.7321428571, MSE=0.1961096939

Node number 108: 207 observations,    complexity param=0.0124757546
  mean=0.4202898551, MSE=0.2436462928
  left son=216 (57 obs) right son=217 (150 obs)
  Primary splits:
      PREVAILING_WAGE             < -0.536224527   to the left,  improve=0.09350362976, (0 missing)
      WAGE_RATE_OF_PAY_FROM       < -0.7449096129  to the left,  improve=0.07284482759, (0 missing)
      AGENT_ATTORNEY_NAME         < -0.748206464   to the right, improve=0.05617100952, (0 missing)
      SOC_CODE                    < -1.269951157   to the left,  improve=0.05524820008, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.2683489623  to the left,  improve=0.05243586646, (0 missing)
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < -0.5289673438  to the left,  agree=0.937, adj=0.772, (0 split)
      SOC_CODE              < -1.29192271    to the left,  agree=0.802, adj=0.281, (0 split)
      PW_SOURCE_YEAR        < -1.937709229   to the left,  agree=0.739, adj=0.053, (0 split)
      VISA_CLASS            < -1.16321874    to the left,  agree=0.734, adj=0.035, (0 split)
      NAICS_CODE            < -1.305640215   to the left,  agree=0.734, adj=0.035, (0 split)

Node number 109: 73 observations
  mean=0.7945205479, MSE=0.1632576468

Node number 216: 57 observations
  mean=0.1754385965, MSE=0.1446598954

Node number 217: 150 observations
  mean=0.5133333333, MSE=0.2498222222

                       [,1]         [,2]
Accuracy       0.7777777778 0.7857142857
Sensitivity    0.8227513228 0.8148148148
Specificity    0.7328042328 0.7566137566
Pos Pred Value 0.7548543689 0.7700000000
Neg Pred Value 0.8052325581 0.8033707865
```



```
> tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
[1] 0.7645502646 0.5000000000
Call:
rpart(formula = CASE_STATUS ~ ., data = trdf)
  n= 1512

               CP nsplit    rel error        xerror               xstd
1  0.11927207774      0 1.0000000000 1.0010960872 0.0003901439582
2  0.07477622311      1 0.8807279223 0.8919106441 0.0164923554691
3  0.04859809766      2 0.8059516992 0.8209221322 0.0183619653882
4  0.04755908693      3 0.7573536015 0.8163494990 0.0208662328257
5  0.02778619854      4 0.7097945146 0.7772086997 0.0216917508948
6  0.02597710990      5 0.6820083160 0.7454007911 0.0223804580989
7  0.02395833732      6 0.6560312061 0.7276428965 0.0224343265142
8  0.02190352021      7 0.6320728688 0.7010193478 0.0235625792209
9  0.01367423664      8 0.6101693486 0.6599536426 0.0236410499068
10 0.01251688149      9 0.5964951120 0.6509482269 0.0236254566878
11 0.01047894807     11 0.5714613490 0.6474111859 0.0241068545961
12 0.01000000000     12 0.5609824009 0.6205958735 0.0237703397569

Variable importance
              EMPLOYER_NAME                                   WAGE_DIFF
AGENT_ATTORNEY_NAME                    AGENT_REPRESENTING_EMPLOYER
H1B_DEPENDENT                       SUPPORT_H1B
                         20                                          19
13                                 9                                  7
6
                  PW_SOURCE                               TOTAL_WORKERS
PW_WAGE_LEVEL                                            NEW_EMPLOYMENT
AMENDED_PETITION                PREVAILING_WAGE
                          4                                           4
3                                 3                                   2
2
      WAGE_RATE_OF_PAY_FROM                       CONTINUED_EMPLOYMENT
SOC_CODE                 CHANGE_EMPLOYER                      VISA_CLASS
PW_SOURCE_YEAR
                          2                                           1
1                                 1                                   1
1

Node    number    1:    1512    observations,          complexity
param=0.1192720777
  mean=0.5, MSE=0.25
  left son=2 (974 obs) right son=3 (538 obs)
  Primary splits:
      EMPLOYER_NAME     <   -0.392590771          to   the   left,
improve=0.11927207770, (0 missing)
      WAGE_DIFF             <   -0.3046590705      to   the   left,
improve=0.10769230770, (0 missing)
      H1B_DEPENDENT     <   -0.5511350306         to   the   right,
improve=0.08675346957, (0 missing)
```

```
      SUPPORT_H1B    < -0.5771593149   to the right, improve=0.08245383201, (0 missing)
      NEW_EMPLOYMENT < -0.04780941156  to the right, improve=0.06142357501, (0 missing)
  Surrogate splits:
      TOTAL_WORKERS        < -0.01220746605  to the left,  agree=0.729, adj=0.240, (0 split)
      AMENDED_PETITION     < -0.003659157996 to the left,  agree=0.687, adj=0.121, (0 split)
      CONTINUED_EMPLOYMENT < 0.3325550385    to the left,  agree=0.671, adj=0.074, (0 split)
      CHANGE_EMPLOYER      < 0.2681569126    to the left,  agree=0.662, adj=0.050, (0 split)
      NEW_EMPLOYMENT       < 0.10093098      to the left,  agree=0.660, adj=0.045, (0 split)

Node number 2: 974 observations,    complexity param=0.04859809766
  mean=0.3716632444, MSE=0.2335296772 
  left son=4 (779 obs) right son=5 (195 obs)
  Primary splits:
      H1B_DEPENDENT               < -0.5511350306   to the right, improve=0.08076255197, (0 missing)
      NEW_EMPLOYMENT              < -0.04780941156  to the right, improve=0.07638014476, (0 missing)
      WAGE_DIFF                   < -0.3028767536   to the left,  improve=0.07575640328, (0 missing)
      AGENT_ATTORNEY_NAME         < 0.2591668455    to the right, improve=0.07498269182, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.243621524    to the left,  improve=0.07498269182, (0 missing)
  Surrogate splits:
      SUPPORT_H1B      < -0.5771593149   to the right, agree=0.980, adj=0.903, (0 split)
      AMENDED_PETITION < -0.003659157996 to the left,  agree=0.813, adj=0.067, (0 split)
      SOC_CODE         < -1.396764036    to the right, agree=0.802, adj=0.010, (0 split)

Node number 3: 538 observations,    complexity param=0.07477622311
  mean=0.7323420074, MSE=0.1960171916 
  left son=6 (48 obs) right son=7 (490 obs)
  Primary splits:
      WAGE_DIFF           < -0.3050793806   to the left,  improve=0.26802721090, (0 missing)
      PW_SOURCE           < -0.6316668749   to the left,  improve=0.06719986815, (0 missing)
      PW_WAGE_LEVEL       < -1.125259665    to the left,  improve=0.06081310016, (0 missing)
      EMPLOYER_NAME       < 3.335813705     to the right, improve=0.04862867119, (0 missing)
      AGENT_ATTORNEY_NAME < 0.3223728988    to the left,  improve=0.04682342152, (0 missing)

Node number 4: 779 observations,    complexity param=0.04755908693
  mean=0.3029525032, MSE=0.211172284 
  left son=8 (321 obs) right son=9 (458 obs)
  Primary splits:
      AGENT_ATTORNEY_NAME         < -0.758951988    to the right, improve=0.10928257840, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.243621524    to the left,  improve=0.10825036530, (0 missing)
      WAGE_DIFF                   < -0.3024134859   to the left,  improve=0.09361140087, (0 missing)
      NEW_EMPLOYMENT              < -0.04780941156  to the right, improve=0.06061702108, (0 missing)
      EMPLOYER_NAME               < -0.4324421652   to the right, improve=0.06021503497, (0 missing)
  Surrogate splits:
      AGENT_REPRESENTING_EMPLOYER < -0.243621524    to the left,  agree=0.999, adj=0.997, (0 split)
      NEW_EMPLOYMENT              < -0.04780941156  to the right, agree=0.733, adj=0.352, (0 split)
      EMPLOYER_NAME               < -0.4048527385   to the right, agree=0.710, adj=0.296, (0 split)
      VISA_CLASS                  < -1.173640163    to the left,  agree=0.634, adj=0.112, (0 split)
      PREVAILING_WAGE             < -1.044592675    to the left,  agree=0.632, adj=0.106, (0 split)

Node number 5: 195 observations,    complexity param=0.02190352021
  mean=0.6461538462, MSE=0.2286390533 
  left son=10 (18 obs) right son=11 (177 obs)
  Primary splits:
      WAGE_DIFF      < -0.3048056316   to the left,  improve=0.18570375830, (0 missing)
      PW_WAGE_LEVEL  < -1.125259665    to the left,  improve=0.07812136712, (0 missing)
      PW_SOURCE_YEAR < -2.039856404    to the left,  improve=0.06058374209, (0 missing)
      NAICS_CODE     < -0.1057265349   to the left,  improve=0.04500984285, (0 missing)
      SOC_CODE       < -1.23823868     to the left,  improve=0.04125132939, (0 missing)
  Surrogate splits:
      SOC_CODE              < -1.389481555    to the left,  agree=0.918, adj=0.111, (0 split)
      PREVAILING_WAGE       < -1.420168199    to the left,  agree=0.918, adj=0.111, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.401655327    to the left,  agree=0.918, adj=0.111, (0 split)

Node number 6: 48 observations
  mean=0, MSE=0 

Node number 7: 490 observations,    complexity param=0.0259771099
  mean=0.8040816327, MSE=0.1575343607 
  left son=14 (76 obs) right son=15 (414 obs)
  Primary splits:
      PW_SOURCE                   < -0.6316668749   to the left,  improve=0.12720707210, (0 missing)
      PW_WAGE_LEVEL               < -1.125259665    to the left,  improve=0.11131723620, (0 missing)
      H1B_DEPENDENT               < -0.5511350306   to the right, improve=0.04832180034, (0 missing)
      AGENT_ATTORNEY_NAME         < 0.3223728988    to the left,  improve=0.04654534595, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.243621524    to the right, improve=0.04654534595, (0 missing)
  Surrogate splits:
      PW_WAGE_LEVEL         < -1.125259665    to the left,  agree=0.916, adj=0.461, (0 split)
      PW_SOURCE_YEAR        < -2.039856404    to the left,  agree=0.853, adj=0.053, (0 split)
      AGENT_ATTORNEY_NAME   < -0.8056784396   to the left,  agree=0.851, adj=0.039, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.127688768    to the left,  agree=0.847, adj=0.013, (0 split)

Node number 8: 321 observations
  mean=0.1214953271, MSE=0.1067342126 

Node number 9: 458 observations,    complexity param=0.02778619854
  mean=0.4301310044, MSE=0.2451183234 
  left son=18 (142 obs) right son=19 (316 obs)
  Primary splits:
      AGENT_ATTORNEY_NAME   < -0.8063359799   to the left,  improve=0.09355773063, (0 missing)
      WAGE_DIFF             < -0.3050793806   to the left,  improve=0.08224096184, (0 missing)
      EMPLOYER_NAME         < -0.4324421652   to the right, improve=0.06332741706, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.6252342203   to the left,  improve=0.05854053306, (0 missing)
      PW_WAGE_LEVEL         < -1.125259665    to the left,  improve=0.03903958576, (0 missing)
  Surrogate splits:
      PREVAILING_WAGE       < -1.203943615    to the left,  agree=0.714, adj=0.077, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.130039082    to the left,  agree=0.712, adj=0.070, (0 split)
      FULL_TIME_POSITION    < -2.426412239    to the left,  agree=0.703, adj=0.042, (0 split)
      WAGE_DIFF             < -3.935413089    to the left,  agree=0.694, adj=0.014, (0 split)
      EMPLOYER_NAME         < -0.4309094193   to the left,  agree=0.692, adj=0.007, (0 split)

Node number 10: 18 observations
  mean=0, MSE=0 

Node number 11: 177 observations
```

```
  mean=0.7118644068, MSE=0.2051134731

Node number 14: 76 observations,    complexity param=0.01367423664
  mean=0.4736842105, MSE=0.2493074792
  left son=28 (53 obs) right son=29 (23 obs)
  Primary splits:
      AGENT_ATTORNEY_NAME          < 0.3223728988    to the left,  improve=0.2728010209, (0 missing)
      AGENT_REPRESENTING_EMPLOYER < -0.243621524    to the right, improve=0.2728010209, (0 missing)
      H1B_DEPENDENT                < -0.5511350306   to the right, improve=0.2043839758, (0 missing)
      SUPPORT_H1B                  < -0.5771593149   to the right, improve=0.1611851852, (0 missing)
      PREVAILING_WAGE              < 0.8652790838    to the right, improve=0.0850024050, (0 missing)
  Surrogate splits:
      AGENT_REPRESENTING_EMPLOYER < -0.243621524    to the right, agree=1.000, adj=1.000, (0 split)
      H1B_DEPENDENT                < -0.5511350306   to the right, agree=0.763, adj=0.217, (0 split)
      SUPPORT_H1B                  < -0.5771593149   to the right, agree=0.737, adj=0.130, (0 split)
      EMPLOYER_NAME                < -0.3833942955   to the right, agree=0.724, adj=0.087, (0 split)
      SOC_CODE                     < -1.271610464    to the right, agree=0.724, adj=0.087, (0 split)

Node number 15: 414 observations,    complexity param=0.01047894807
  mean=0.8647342995, MSE=0.1169688908
  left son=30 (35 obs) right son=31 (379 obs)
  Primary splits:
      EMPLOYER_NAME        < 3.335813705     to the right, improve=0.08179726365, (0 missing)
      CONTINUED_EMPLOYMENT < 0.3325550385    to the right, improve=0.04449677697, (0 missing)
      H1B_DEPENDENT        < -0.5511350306   to the right, improve=0.03049365543, (0 missing)
      SUPPORT_H1B          < -0.5771593149   to the right, improve=0.02988631979, (0 missing)
      PW_WAGE_LEVEL        < -0.7043798753   to the left,  improve=0.02486628865, (0 missing)
  Surrogate splits:
      PREVAILING_WAGE < -1.427905667    to the left,  agree=0.918, adj=0.029, (0 split)

Node number 18: 142 observations
  mean=0.2042253521, MSE=0.1625173577

Node number 19: 316 observations,    complexity param=0.02395833732
  mean=0.5316455696, MSE=0.2489985579
  left son=38 (165 obs) right son=39 (151 obs)
  Primary splits:
      WAGE_DIFF             < -0.213246257    to the left,  improve=0.11509714750, (0 missing)
      PW_WAGE_LEVEL         < -1.125259665    to the left,  improve=0.08376384197, (0 missing)
      EMPLOYER_NAME         < -0.4324421652   to the right, improve=0.07813113320, (0 missing)
      PW_SOURCE             < -0.6316668749   to the left,  improve=0.05639213205, (0 missing)
      WAGE_RATE_OF_PAY_FROM < -0.6254967553   to the left,  improve=0.04142435346, (0 missing)
  Surrogate splits:
      WAGE_RATE_OF_PAY_FROM < -0.2403453602   to the left,  agree=0.652, adj=0.272, (0 split)
      SOC_CODE              < -0.1800640999   to the left,  agree=0.595, adj=0.152, (0 split)
      PREVAILING_WAGE       < 0.004133279084  to the left,  agree=0.554, adj=0.066, (0 split)
      NAICS_CODE            < -1.213336253    to the left,  agree=0.544, adj=0.046, (0 split)
      EMPLOYER_NAME         < -0.4324421652   to the right, agree=0.535, adj=0.026, (0 split)

Node number 28: 53 observations
  mean=0.3018867925, MSE=0.210751157
```

```
Node number 29: 23 observations
  mean=0.8695652174, MSE=0.1134215501 

Node number 30: 35 observations
  mean=0.5428571429, MSE=0.2481632653 

Node number 31: 379 observations
  mean=0.8944591029, MSE=0.09440201614 

Node number 38: 165 observations,    complexity param=0.01251688149
  mean=0.3696969697, MSE=0.2330211203 
  left son=76 (28 obs) right son=77 (137 obs)
  Primary splits:
      WAGE_DIFF           < -0.3052899568   to the left,  improve=0.11987647390, (0 missing)
      EMPLOYER_NAME       < -0.4324421652   to the right, improve=0.09627066069, (0 missing)
      PW_WAGE_LEVEL       < -1.125259665    to the left,  improve=0.07537190111, (0 missing)
      AGENT_ATTORNEY_NAME < -0.7991030375   to the left,  improve=0.06544507007, (0 missing)
      PW_SOURCE           < -0.6316668749   to the left,  improve=0.05844433060, (0 missing)

Node number 39: 151 observations
  mean=0.7086092715, MSE=0.2064821718 

Node number 76: 28 observations
  mean=0, MSE=0 

Node number 77: 137 observations,    complexity param=0.01251688149
  mean=0.4452554745, MSE=0.2470030369 
  left son=154 (34 obs) right son=155 (103 obs)
  Primary splits:
      PW_WAGE_LEVEL       < -1.125259665    to the left,  improve=0.14343313740, (0 missing)
      PW_SOURCE           < -0.6316668749   to the left,  improve=0.12750135320, (0 missing)
      EMPLOYER_NAME       < -0.4324421652   to the right, improve=0.08709099356, (0 missing)
      AGENT_ATTORNEY_NAME < -0.7991030375   to the left,  improve=0.05787960922, (0 missing)
      FULL_TIME_POSITION  < -2.426412239    to the left,  improve=0.04977560180, (0 missing)
  Surrogate splits:
      PW_SOURCE             < -0.6316668749   to the left,  agree=0.869, adj=0.471, (0 split)
      PREVAILING_WAGE       < -1.67798671     to the left,  agree=0.796, adj=0.176, (0 split)
      PW_SOURCE_YEAR        < -2.039856404    to the left,  agree=0.796, adj=0.176, (0 split)
      WAGE_RATE_OF_PAY_FROM < -1.534833098    to the left,  agree=0.796, adj=0.176, (0 split)
      EMPLOYER_NAME         < -0.4017872466   to the right, agree=0.759, adj=0.029, (0 split)

Node number 154: 34 observations
  mean=0.1176470588, MSE=0.1038062284 

Node number 155: 103 observations
  mean=0.5533980583, MSE=0.2471486474 

                       [,1]         [,2]
Accuracy       0.7645502646 0.7698412698
Sensitivity    0.8253968254 0.8492063492
Specificity    0.7037037037 0.6904761905
Pos Pred Value 0.7358490566 0.7328767123
```

```
Neg Pred Value 0.8012048193 0.8207547170
```

Appendix C. R code written for the project

```
# Final project for 15.062 Data Mining
# Certification/Denial predictions of labor condition application (LCA) for H-1B visa
petitions
# Written by Sohae Kim

setwd("/Users/s_kim/Library/Mobile
Documents/3L68KQB4HG~com~readdle~CommonDocuments/Documents/15.062/FinalProject")
set.seed(1)

#install.packages("readxl")
#install.packages("RColorBrewer")
#install.packages("wesanderson")
#library(RColorBrewer)
#library(wesanderson)
library(readxl)
library(ggplot2)
library(stringr)

df <- read_excel("H-1B_FY2018.xlsx")
#df <- read_excel("H-1B_Disclosure_Data_FY17.xlsx") # Consider uncomment only when
your computer has enough free ram ~3G.
df.backup <- df
df <- df.backup
colnames(df)[which(names(df) == "H-1B_DEPENDENT")] <- "H1B_DEPENDENT"
colSums(is.na(df)*1)
df[,c("EMPLOYER_BUSINESS_DBA","EMPLOYER_ADDRESS","EMPLOYER_POSTAL_CODE","EMPLOYER_COUN
TRY","EMPLOYER_PROVINCE","EMPLOYER_PHONE","EMPLOYER_PHONE_EXT",
#"AGENT_ATTORNEY_NAME",

"AGENT_ATTORNEY_CITY","AGENT_ATTORNEY_STATE","JOB_TITLE","SOC_NAME","WAGE_RATE_OF_PAY_
TO","LABOR_CON_AGREE","PUBLIC_DISCLOSURE_LOCATION","WORKSITE_COUNTY",
      "WORKSITE_POSTAL_CODE","ORIGINAL_CERT_DATE")] <- NULL
summary(df)

factor.list <-
c("CASE_STATUS","VISA_CLASS","EMPLOYER_NAME","EMPLOYER_CITY","EMPLOYER_STATE","AGENT_R
EPRESENTING_EMPLOYER","SOC_CODE","NAICS_CODE","FULL_TIME_POSITION",

"PW_UNIT_OF_PAY","PW_WAGE_LEVEL","PW_SOURCE","PW_SOURCE_YEAR","PW_SOURCE_OTHER","WAGE_
UNIT_OF_PAY","H1B_DEPENDENT","WILLFUL_VIOLATOR","SUPPORT_H1B",
                 "WORKSITE_CITY","WORKSITE_STATE")
for (i in 1:length(factor.list)) {
  df[which(df[,factor.list[i]] == "N/A" | df[,factor.list[i]] == "NA", arr.ind =
TRUE),factor.list[i]] <- NA
}
df[,factor.list] <- lapply(df[,factor.list], factor)
colSums(is.na(df)*1)
levels(df$PW_WAGE_LEVEL) <- c(levels(df$PW_WAGE_LEVEL), "N/A")
#levels(df$H1B_DEPENDENT) <- c(levels(df$H1B_DEPENDENT), "N/A")
#levels(df$SUPPORT_H1B) <- c(levels(df$SUPPORT_H1B), "N/A")
df[,c("WILLFUL_VIOLATOR","PW_SOURCE_OTHER")] <- NULL
df[is.na(df$PW_WAGE_LEVEL),]$PW_WAGE_LEVEL<- "N/A"
df[is.na(df$AGENT_REPRESENTING_EMPLOYER),]$AGENT_REPRESENTING_EMPLOYER<- "N" #"N/A"
df[is.na(df$H1B_DEPENDENT),]$H1B_DEPENDENT<- "N" #"N/A"
df[is.na(df$SUPPORT_H1B),]$SUPPORT_H1B<- "N" #"N/A"
colSums(is.na(df)*1)

df$H1B_DEPENDENT <- 1*(df$H1B_DEPENDENT == "Y")
df$SUPPORT_H1B <- 1*(df$SUPPORT_H1B == "Y")
df$AGENT_REPRESENTING_EMPLOYER <- 1*(df$AGENT_REPRESENTING_EMPLOYER == "Y")
df$FULL_TIME_POSITION <- 1*(df$FULL_TIME_POSITION=="Y")
#df$WILLFUL_VIOLATOR <- 1*(df$WILLFUL_VIOLATOR == "Y")
df$SOC_CODE <- str_extract(df$SOC_CODE, "\\d{2}")
df$NAICS_CODE <- str_extract(df$NAICS_CODE, "\\d{2}")
df[,c("SOC_CODE","NAICS_CODE")] <- lapply(df[,c("SOC_CODE","NAICS_CODE")], factor)

summary(df)

tmp <- which(df$WAGE_RATE_OF_PAY_FROM > 9.99E+8, arr.ind=TRUE) #For 2018 data, this
```

```
data point seems like a fake application with FROM YOUR SECRET SANTA on the
WORKSITE_CITY: I-200-17347-732455DENIED 12/13/17      12/15/17      H-1B   12/25/17
       12/25/20      DELOITTE CONSULTING LLP          1700 MARKET STREET  PHILADELPHIA
       PA     19103  UNITED STATES OF AMERICA         2152462300          Y
       MICHAELS, REBECCA   TORONTO             U.S. LEGAL ASSISTANT       23-1011
       LAWYERS      54161  1      1      0      0      0      0      0      Y
       1,000,000,000.00    Year   N/A    OES    2017   OFLC ONLINE DATA CENTER
       1,000,000,000.00    0.00   Year   N      N      NA                  FROM YOUR
       SECRET SANTA
if(length(tmp)!=0) df <- df[-tmp,]

df_cert_den <- df[which(df$CASE_STATUS == "DENIED" | df$CASE_STATUS == "CERTIFIED"),]
factor.list <-
c("CASE_STATUS","VISA_CLASS","EMPLOYER_NAME","EMPLOYER_CITY","EMPLOYER_STATE","SOC_COD
E","NAICS_CODE","PW_UNIT_OF_PAY", #"H1B_DEPENDENT","SUPPORT_H1B",

"PW_WAGE_LEVEL","PW_SOURCE","PW_SOURCE_YEAR","WAGE_UNIT_OF_PAY","WORKSITE_CITY","WORKS
ITE_STATE")
df_cert_den[factor.list] <- lapply(df_cert_den[factor.list], factor)

######################################
# Exploratory visualizations with df #
######################################
boxplot(df$WAGE_RATE_OF_PAY_FROM)
boxplot(df$WAGE_RATE_OF_PAY_FROM ~ df$CASE_STATUS)
ggplot(data = df, aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color =
CASE_STATUS)) + geom_point(size = 1.0, show.legend = TRUE) +
  scale_y_log10(name="Wage rate of pay from", #limits=c(1e4, 1e9),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
  scale_x_log10(name="Prevailing wage", #limits=c(1e4, 1e6),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
  theme( axis.text.x = element_text(face="bold", size=14),
         axis.text.y = element_text(face="bold", size=14),
         axis.title.x  = element_text(size=14),
         axis.title.y  = element_text(size=14),
         panel.grid.major = element_blank(),
         panel.grid.minor = element_blank(),
         panel.background = element_rect(colour = "black", size=1))
ggplot(data = df, aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color =
CASE_STATUS)) + geom_point(size = 1.0, show.legend = TRUE)
ggplot(data = df[-c(exc,which(df$CASE_STATUS=="WITHDRAWN",arr.ind=TRUE)),], aes(x =
PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color = CASE_STATUS)) +
  geom_point(size = 1.0, show.legend = TRUE)+ scale_y_log10() + scale_x_log10()
ggplot(data = df[c(which(df$CASE_STATUS=="DENIED",arr.ind=TRUE)),], aes(x =
PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color = CASE_STATUS)) +
  geom_point(size = 1.0, show.legend = TRUE)+ scale_y_log10() + scale_x_log10()
plot(df$CASE_SUBMITTED, df$DECISION_DATE, type="p")
hist(df$DECISION_DATE,"days")
hist(df$CASE_SUBMITTED,"months")
boxplot(df$EMPLOYMENT_START_DATE ~ df$CASE_STATUS)
#Withdrawn tends to be old datas
ggplot(df, aes(x = CASE_SUBMITTED)) + geom_histogram() + scale_y_log10()


#######################################################################
# Exploratory visualizations with df_cert_den, only CERTIFIED & DENIED #
#######################################################################
boxplot(df_cert_den$WAGE_RATE_OF_PAY_FROM)
boxplot(df_cert_den$WAGE_RATE_OF_PAY_FROM ~ df_cert_den$CASE_STATUS)

#Need to convert the wages in the same unit
#Wages are grouped in four: (1) Prevailing wage & Wage rate of pay from < ~10K; (2)
Prevailing wage < ~10K & Prevailing wage & Wage rate of pay from > ~10K;
#(3) Prevailing wage > ~10K & Wage rate of pay from < ~10K; (4) Prevailing wage & Wage
rate of pay from > ~10K;
ggplot(data = df_cert_den, aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color =
CASE_STATUS)) + geom_point(size = 1.0, show.legend = TRUE) +
  scale_y_log10(name="Wage rate of pay from", #limits=c(1e4, 1e9),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
  scale_x_log10(name="Prevailing wage", #limits=c(1e4, 1e6),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
  theme( axis.text.x = element_text(face="bold", size=14),
         axis.text.y = element_text(face="bold", size=14),
         axis.title.x  = element_text(size=14),
         axis.title.y  = element_text(size=14),
         panel.grid.major = element_blank(),
         panel.grid.minor = element_blank(),
         panel.background = element_rect(colour = "black", size=1)) #+
theme_linedraw()
ggplot(data = df_cert_den[c(which(df_cert_den$CASE_STATUS=="DENIED",arr.ind=TRUE)),],
aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color = CASE_STATUS)) +
geom_point(size = 1.0, show.legend = TRUE) + scale_y_log10() + scale_x_log10()

############################################
# Conversion of wages in a same unit, Year #
############################################
#There are 147 records that have mismatch between the units of prevailing wage & wage
rate of pay from 2018 Q1 data
#There are 844 records that have mismatch between the units of prevailing wage & wage
rate of pay from 2017 data
cbind(df_cert_den[which(df_cert_den$PW_UNIT_OF_PAY!=df_cert_den$WAGE_UNIT_OF_PAY,arr.i
nd=TRUE),]$PW_UNIT_OF_PAY,

df_cert_den[which(df_cert_den$PW_UNIT_OF_PAY!=df_cert_den$WAGE_UNIT_OF_PAY,arr.ind=TRU
E),]$WAGE_UNIT_OF_PAY,

df_cert_den[which(df_cert_den$PW_UNIT_OF_PAY!=df_cert_den$WAGE_UNIT_OF_PAY,arr.ind=TRU
E),]$PREVAILING_WAGE,

df_cert_den[which(df_cert_den$PW_UNIT_OF_PAY!=df_cert_den$WAGE_UNIT_OF_PAY,arr.ind=TRU
E),]$WAGE_RATE_OF_PAY_FROM,

#df_cert_den[which(df_cert_den$PW_UNIT_OF_PAY!=df_cert_den$WAGE_UNIT_OF_PAY,arr.ind=TR
UE),]$WAGE_RATE_OF_PAY_TO,

df_cert_den[which(df_cert_den$PW_UNIT_OF_PAY!=df_cert_den$WAGE_UNIT_OF_PAY,arr.ind=TRU
E),]$CASE_STATUS)

# Conversion for prevailing wage
df_cert_den$PREVAILING_WAGE <-
df_cert_den$PREVAILING_WAGE*26*(df_cert_den$PW_UNIT_OF_PAY == "Bi-Weekly") +
df_cert_den$PREVAILING_WAGE*52*40*(df_cert_den$PW_UNIT_OF_PAY == "Hour") +
  df_cert_den$PREVAILING_WAGE*12*(df_cert_den$PW_UNIT_OF_PAY == "Month") +
df_cert_den$PREVAILING_WAGE*52*(df_cert_den$PW_UNIT_OF_PAY == "Week") +
df_cert_den$PREVAILING_WAGE*1*(df_cert_den$PW_UNIT_OF_PAY == "Year")
# Conversion for wage rate of pay from & to
df_cert_den$WAGE_RATE_OF_PAY_FROM <-
df_cert_den$WAGE_RATE_OF_PAY_FROM*26*(df_cert_den$WAGE_UNIT_OF_PAY == "Bi-Weekly") +
  df_cert_den$WAGE_RATE_OF_PAY_FROM*52*40*(df_cert_den$WAGE_UNIT_OF_PAY == "Hour") +
  df_cert_den$WAGE_RATE_OF_PAY_FROM*12*(df_cert_den$WAGE_UNIT_OF_PAY == "Month") +
  df_cert_den$WAGE_RATE_OF_PAY_FROM*52*(df_cert_den$WAGE_UNIT_OF_PAY == "Week") +
  df_cert_den$WAGE_RATE_OF_PAY_FROM*1*(df_cert_den$WAGE_UNIT_OF_PAY == "Year")
#df_cert_den$WAGE_RATE_OF_PAY_TO <-
df_cert_den$WAGE_RATE_OF_PAY_TO*26*(df_cert_den$WAGE_UNIT_OF_PAY == "Bi-Weekly") +
#  df_cert_den$WAGE_RATE_OF_PAY_TO*52*40*(df_cert_den$WAGE_UNIT_OF_PAY == "Hour") +
#  df_cert_den$WAGE_RATE_OF_PAY_TO*12*(df_cert_den$WAGE_UNIT_OF_PAY == "Month") +
#  df_cert_den$WAGE_RATE_OF_PAY_TO*52*(df_cert_den$WAGE_UNIT_OF_PAY == "Week") +
#  df_cert_den$WAGE_RATE_OF_PAY_TO*1*(df_cert_den$WAGE_UNIT_OF_PAY == "Year")

#ggplot(data = df_cert_den, aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color
= CASE_STATUS)) + geom_point(size = 0.4, show.legend = TRUE) +
#  theme( axis.text.x = element_text(face="bold", size=14), axis.text.y =
element_text(face="bold", size=14)) +
#  scale_x_continuous(name="Prevailing wage") + #, limits=c(0, 3e5)) + #,
breaks=seq(10,1e8,8))
#  scale_y_continuous(name="Wage rate of pay from") #, limits=c(0, 1e6))
ggplot(data = df_cert_den, aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color =
CASE_STATUS)) + geom_point(size = 1.0, show.legend = TRUE) +
#scale_color_manual(values = wes_palette(n=2, name="GrandBudapest")) +
#scale_color_brewer(palette="Dark2") +
  scale_x_log10(name="Prevailing wage", #limits=c(1e4, 1e6),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
  scale_y_log10(name="Wage rate of pay from", #limits=c(1e4, 1e9),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
  theme( axis.text.x = element_text(face="bold", size=14), axis.text.y =
element_text(face="bold", size=14),
         axis.title.x  = element_text(size=14), axis.title.y  = element_text(size=14),
         panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
         panel.background = element_rect(colour = "black", size=1))
summary(df_cert_den)

###########################################
# Removing outliers that are 100% DENIED. #
###########################################
#Outliers in two groups: (1) Prevailing wage and/or is very low, less than 10K; (2)
Prevailing wage is very high, larger than ~1M.
#Outlier group (1)
cbind(df_cert_den[which(df_cert_den$PREVAILING_WAGE <
10000.0,arr.ind=TRUE),]$CASE_STATUS,
      df_cert_den[which(df_cert_den$PREVAILING_WAGE <
10000.0,arr.ind=TRUE),]$PREVAILING_WAGE,
      df_cert_den[which(df_cert_den$PREVAILING_WAGE <
10000.0,arr.ind=TRUE),]$WAGE_RATE_OF_PAY_FROM #,
df_cert_den[which(df_cert_den$PREVAILING_WAGE <
10000.0,arr.ind=TRUE),]$WAGE_RATE_OF_PAY_TO
      )
cbind(df_cert_den[which(df_cert_den$WAGE_RATE_OF_PAY_FROM <
10000.0,arr.ind=TRUE),]$CASE_STATUS,
      df_cert_den[which(df_cert_den$WAGE_RATE_OF_PAY_FROM <
10000.0,arr.ind=TRUE),]$PREVAILING_WAGE,
      df_cert_den[which(df_cert_den$WAGE_RATE_OF_PAY_FROM <
10000.0,arr.ind=TRUE),]$WAGE_RATE_OF_PAY_FROM #,
df_cert_den[which(df_cert_den$WAGE_RATE_OF_PAY_FROM <
10000.0,arr.ind=TRUE),]$WAGE_RATE_OF_PAY_TO
      )
# All DENIED

#Outlier group (2)
head(cbind(df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$CASE_STATUS,
           df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$EMPLOYER_NAME,
           df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$VISA_CLASS,
           df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$WAGE_RATE_OF_PAY_FROM,
           #df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$WAGE_RATE_OF_PAY_TO,
           df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$WAGE_UNIT_OF_PAY,
           df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$PREVAILING_WAGE,
           df_cert_den[order(df_cert_den$WAGE_RATE_OF_PAY_FROM,decreasing =
TRUE),]$PW_UNIT_OF_PAY), 50)
#Not always DENIED
head(cbind(df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"CASE_STATUS"],
           df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"EMPLOYER_NAME"],
           df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"VISA_CLASS"],
           df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"WAGE_RATE_OF_PAY_FROM"],
           #df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"WAGE_RATE_OF_PAY_TO"],
           df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"WAGE_UNIT_OF_PAY"],
           df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"PREVAILING_WAGE"],
           df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"PW_UNIT_OF_PAY"],
           df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE),"AGENT_ATTORNEY_NAME"]),
    #125) # 2017 data
    30) # 2018 Q1 data
#View(df_cert_den[order(df_cert_den$PREVAILING_WAGE,decreasing =
TRUE)[1:150],c("CASE_STATUS","JOB_TITLE","VISA_CLASS","WAGE_RATE_OF_PAY_FROM","WAGE_RA
TE_OF_PAY_TO","WAGE_UNIT_OF_PAY","PREVAILING_WAGE","PW_UNIT_OF_PAY")])
#Top 22 cases have denoted that the unit of prevailing wages are hourly-based, and the
converted yearly-based wage is outrageously high. The wage rage of pay from is around
50K to 200K in yearly-based.
#A couple of cases report ~77K and ~133K hourly based, and another couple report ~60K
and ~68K hourly based.
#Overall, the prevailing wage in yearly-based does not make common sense and mismatch
a lot to the wage rate pay from as well. This would be due to the misinterpretation of
the prevailing wage & unit.
#Always DENIED
which(df_cert_den$CASE_STATUS=="DENIED" & df_cert_den$AGENT_ATTORNEY_NAME ==
"BRADSHAW, MELANIE",arr.ind=TRUE) # For 2018 data
#df_cert_den[which(df_cert_den$CASE_STATUS=="DENIED",
arr.ind=TRUE),]$AGENT_ATTORNEY_NAME[order(freq, decreasing = T),]

#df_cert_den$PW_UNIT_OF_PAY <- "Year"
#df_cert_den$WAGE_UNIT_OF_PAY <- "Year"

summary(df_cert_den)

# delete outlier, since they do not need a model to predict the decision. In the
outlier range, all records are DENIED.
df_cdwo_outlier <- df_cert_den[-which(df_cert_den$PREVAILING_WAGE < 1e4 |
df_cert_den$WAGE_RATE_OF_PAY_FROM < 1e4 | df_cert_den$PREVAILING_WAGE >
1e6,arr.ind=TRUE),]

#############################################################################
# Visualization of the data without outler, which needs a prediction model #
#############################################################################
#ggplot(data = df_cdwo_outlier, aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM,
color = CASE_STATUS)) + geom_point(size = 0.6, show.legend = TRUE) +
#  scale_y_log10(name="Wage rate of pay from", #limits=c(1e4, 1e9),
#                breaks = scales::trans_breaks("log10", function(x) 10^x),
#                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
#theme_linedraw() +
#  scale_x_log10(name="Prevailing wage", limits=c(1e4, 1e6),
#                breaks = scales::trans_breaks("log10", function(x) 10^x),
#                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
#theme_linedraw() +
#  theme( axis.text.x = element_text(face="bold", size=14), axis.text.y =
element_text(face="bold", size=14),
#         axis.title.x  = element_text(size=14), axis.title.y  =
element_text(size=14),
#         panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
#         panel.background = element_rect(colour = "black", size=1)) # may want to run
twice when an error occurs. Warning msg with removed missing values are fine.
#library(ggpubr)
library(cowplot)
theme_set(theme_grey())
dn.plot <- ggplot(data =
df_cdwo_outlier[c(which(df_cdwo_outlier$CASE_STATUS=="DENIED",arr.ind=TRUE)),], aes(x
= PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color = CASE_STATUS)) +
  geom_point(size = 1.0, show.legend = TRUE) + scale_color_manual(values="#00BFC4") +
  scale_y_log10(name="Wage rate of pay from", #limits=c(1e4, 1e9),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
#theme_linedraw() +
  scale_x_log10(name="Prevailing wage", limits=c(1e4, 1e6),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
border(color = "black", size = 1.0, linetype = 1) + #theme_linedraw() +
  theme( axis.text.x = element_text(face="bold", size=14), axis.text.y =
element_text(face="bold", size=14),
         axis.title.x  = element_text(size=14), axis.title.y  = element_text(size=14),
         panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
         panel.background = element_rect(colour = "black", size=1),
legend.text=element_text(size=14))
ct.plot <- ggplot(data =
df_cdwo_outlier[c(which(df_cdwo_outlier$CASE_STATUS=="CERTIFIED",arr.ind=TRUE)),],
aes(x = PREVAILING_WAGE, y = WAGE_RATE_OF_PAY_FROM, color = CASE_STATUS)) +
  geom_point(size = 1.0, show.legend = TRUE) +
  scale_y_log10(name="Wage rate of pay from", #limits=c(1e4, 1e9),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
#theme_linedraw() +
  scale_x_log10(name="Prevailing wage", limits=c(1e4, 1e6),
                breaks = scales::trans_breaks("log10", function(x) 10^x),
                labels = scales::trans_format("log10", scales::math_format(10^.x))) +
border(color = "black", size = 1.0, linetype = 1) +#theme_linedraw() +
  theme( axis.text.x = element_text(face="bold", size=14), axis.text.y =
element_text(face="bold", size=14),
         axis.title.x  = element_blank(), axis.title.y  = element_text(size=14),
         panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
         panel.background = element_rect(colour = "black", size=1),
legend.text=element_text(size=14))
plot_grid(ct.plot, dn.plot, align = "v", nrow = 2) #+ panel_border(colour = "black",
size = 1, linetype = 1,  remove = FALSE)  # May need to run several times due to Error
in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y,  : polygon edge not
found
# We can see a hard rule on the decision: If the wage rate to pay from is lower than
the prevailing wage, they are 100% DENIED. Otherwise, they can be either CERTIFIED or
DENIED.
# Is there any difference between the CERTIFIED and the DENIED but with the paid wage
rate higher than the prevailing wage? I cannot see that much difference.
# Therefore, here is where we need a good prediction model.
summary(df_cdwo_outlier[c(which(df_cdwo_outlier$CASE_STATUS=="DENIED",arr.ind=TRUE)),]
)
summary(df_cdwo_outlier[c(which(df_cdwo_outlier$CASE_STATUS=="DENIED"&df_cdwo_outlier$
WAGE_RATE_OF_PAY_FROM > df_cdwo_outlier$PREVAILING_WAGE,arr.ind=TRUE)),])
summary(df_cdwo_outlier[c(which(df_cdwo_outlier$CASE_STATUS=="CERTIFIED",arr.ind=TRUE)
),])

#########################
# Cleaning data further #
#########################
# Data still has some NAs, but I will just delete the rows with NAs since it is not
that many.
colSums(is.na(df_cdwo_outlier)*1)
cmp.df_cdwo_outlier <- df_cdwo_outlier[complete.cases(df_cdwo_outlier),]
denied.ind <- which(cmp.df_cdwo_outlier$CASE_STATUS=="DENIED",arr.ind=TRUE)
denied_though.ind <-
which(cmp.df_cdwo_outlier$CASE_STATUS=="DENIED"&cmp.df_cdwo_outlier$WAGE_RATE_OF_PAY_F
ROM > cmp.df_cdwo_outlier$PREVAILING_WAGE,arr.ind=TRUE)
certified.ind <- which(cmp.df_cdwo_outlier$CASE_STATUS=="CERTIFIED",arr.ind=TRUE)
summary(cmp.df_cdwo_outlier[denied.ind,])
summary(cmp.df_cdwo_outlier[c(which(cmp.df_cdwo_outlier$CASE_STATUS=="DENIED"&cmp.df_c
dwo_outlier$WAGE_RATE_OF_PAY_FROM >
cmp.df_cdwo_outlier$PREVAILING_WAGE,arr.ind=TRUE)),])
summary(cmp.df_cdwo_outlier[certified.ind,])


###################################
# Functions for all methods I use #
###################################
library(rpart)
library(caret)
library(rpart.plot)
library(neuralnet)
library(forecast)
library(adabag)
library(mlr) #For createDummyFeatures
library(plyr)
```

```
#Tree
tree <- function(trdf,vadf,tedf){
  tr <- rpart(CASE_STATUS ~ ., data = trdf)
  validation.prediction <- predict(tr, vadf)
  accr <- c(0,0.5)
  for (cr in c(seq(0.5,0.7, by=0.05),seq(0.5,0.3, by=-0.05))) {
    if(confusionMatrix(ifelse(validation.prediction>cr, 1, 0), vadf$CASE_STATUS,
positive = "1")$overall[[1]] > accr[1]) {
      accr[1] <- confusionMatrix(ifelse(validation.prediction>cr, 1, 0),
vadf$CASE_STATUS, positive = "1")$overall[[1]]
      accr[2] <- cr
    }
  }
  print(accr)
  #rpart.plot(tr,cex=0.9)
  rpart.plot(tr, type=1, extra = 100, digits=-2, fallen.leaves=FALSE,cex=0.9)
  confusion.val <- confusionMatrix(ifelse(validation.prediction>accr[2], 1, 0),
vadf$CASE_STATUS, positive = "1")
  confusion.test <- confusionMatrix(ifelse(predict(tr, tedf)>accr[2], 1, 0),
tedf$CASE_STATUS, positive = "1")
  summary(tr)
  confusion.test
  print(cbind(c(confusion.val$overall[1], confusion.val$byClass[1:4]),
c(confusion.test$overall[1], confusion.test$byClass[1:4])))
  return(cbind(c(confusion.val$overall[1], confusion.val$byClass[1:4]),
c(confusion.test$overall[1], confusion.test$byClass[1:4])))
}
#Logistic regression
logreg <- function(trdf,vadf,tedf){
  lr <- glm(CASE_STATUS ~ ., data = trdf, family = "binomial")
  validation.prediction <- predict(lr, vadf, type = "response")
  accr <- c(0,0.5)
  for (cr in c(seq(0.5,0.7, by=0.05),seq(0.5,0.3, by=-0.05))) {
    if(confusionMatrix(ifelse(validation.prediction>cr, 1, 0), vadf$CASE_STATUS,
positive = "1")$overall[[1]] > accr[1]) {
      accr[1] <- confusionMatrix(ifelse(predict(lr, vadf, type = "response")>cr, 1,
0), vadf$CASE_STATUS, positive = "1")$overall[[1]]
      accr[2] <- cr
    }
  }
  print(accr)
  confusion.val <- confusionMatrix(ifelse(validation.prediction>accr[2], 1, 0),
vadf$CASE_STATUS, positive = "1")
  confusion.test <- confusionMatrix(ifelse(predict(lr, tedf, type =
"response")>accr[2], 1, 0), tedf$CASE_STATUS, positive = "1")
  summary(lr)
  confusion.test
  return(cbind(c(confusion.val$overall[1], confusion.val$byClass[1:4]),
c(confusion.test$overall[1], confusion.test$byClass[1:4])))
}
#knn
knearn <- function(trdf,vadf,tedf){
  accr <- c(0,1)
  for (cr in 1:15) {
    kn <- class::knn(train = trdf[, -1], test = vadf[,-1], cl = trdf[, 1], k = cr,
prob=TRUE)
    if(confusionMatrix(kn, vadf[,1], positive = "1")$overall[[1]] > accr[1]) {
      accr[1] <- confusionMatrix(kn, vadf[,1], positive = "1")$overall[[1]]
      accr[2] <- cr
    }
    #print(c(cr,accr,confusionMatrix(class::knn(train = trdf[, -1], test = tedf[,-1],
cl = trdf[, 1], k = cr, prob=TRUE), tedf[,1], positive = "1")$byClass[[11]]))
  }
  confusion.val <- confusionMatrix(class::knn(train = trdf[, -1], test = vadf[,-1], cl
= trdf[, 1], k = accr[2], prob=TRUE), vadf[,1], positive = "1")
  confusion.test <- confusionMatrix(class::knn(train = trdf[, -1], test = tedf[,-1],
cl = trdf[, 1], k = accr[2], prob=TRUE), tedf[,1], positive = "1")
  summary(kn)
  confusion.test
  return(cbind(c(confusion.val$overall[1], confusion.val$byClass[1:4]),
c(confusion.test$overall[1], confusion.test$byClass[1:4])))
}
#Neural net
nnet_opt <- function(i,trdf,vadf,tedf){
  form_nn <- as.formula(paste("CASE_STATUS ~ ",paste(names(trdf[,-1]), collapse=" +
"),sep = ""))
  nn <- neuralnet(formula = form_nn , data = trdf, linear.output = T, hidden = i)
  validation.prediction <- as.vector(neuralnet::compute(nn, vadf[, -1])$net.result)
  #print(c(max(ifelse(validation.prediction>0.5, 1,
0)),ifelse(validation.prediction>0.5, 1, 0)))
  #print(vadf$CASE_STATUS)
  accr <- c(0,0.5)
  for (cr in c(seq(0.5,0.65, by=0.05),seq(0.5,0.3, by=-0.05))){
    #print(table(ifelse(validation.prediction>cr, 1, 0)))
    #print(cr)
    #print(table(vadf$CASE_STATUS))
    #print(confusionMatrix(ifelse(validation.prediction>cr, 1, 0), vadf$CASE_STATUS,
positive = "1")$byClass[[11]])
    if(confusionMatrix(ifelse(validation.prediction>cr, 1, 0), vadf$CASE_STATUS,
positive = "1")$overall[[1]] > accr[1]) {
      accr[1] <- confusionMatrix(ifelse(validation.prediction>cr, 1, 0),
vadf$CASE_STATUS, positive = "1")$overall[[1]]
      accr[2] <- cr
    }
  }
  print(accr)
  confusion.val <- confusionMatrix(ifelse(validation.prediction>accr[2], 1, 0),
vadf$CASE_STATUS, positive = "1")
  confusion.test <- confusionMatrix(ifelse(as.vector(neuralnet::compute(nn, tedf[, -
1])$net.result)>accr[2], 1, 0), tedf$CASE_STATUS, positive = "1")
  summary(nn)
  plot(nn)
  confusion.test
  return(cbind(c(confusion.val$overall[1], confusion.val$byClass[1:4]),
c(confusion.test$overall[1], confusion.test$byClass[1:4])))
}


#################################
# (1) Category into dummy variables #
#################################
#####2018
train.ind <- c(sample(denied.ind, length(denied.ind)*0.5), sample(certified.ind,
length(denied.ind)*0.5))
valid.ind <- c(sample(setdiff(denied.ind, train.ind),length(denied.ind)*0.25),
sample(setdiff(certified.ind, train.ind),length(denied.ind)*0.25))
test.ind <- c(sample(setdiff(denied.ind,
c(train.ind,valid.ind)),length(denied.ind)*0.25), sample(setdiff(certified.ind,
c(train.ind,valid.ind)),length(denied.ind)*0.25))

#train.ind <- c(sample(denied_though.ind, length(denied_though.ind)*0.5),
sample(certified.ind, length(denied_though.ind)*0.5))
#valid.ind <- c(sample(setdiff(denied_though.ind,
train.ind),length(denied_though.ind)*0.25), sample(setdiff(certified.ind,
train.ind),length(denied_though.ind)*0.25))
#test.ind <- c(sample(setdiff(denied_though.ind,
c(train.ind,valid.ind)),length(denied_though.ind)*0.25), sample(setdiff(certified.ind,
c(train.ind,valid.ind)),length(denied_though.ind)*0.25))
#####2017
#train.ind <- c(sample(denied.ind, length(denied.ind)*0.12), sample(certified.ind,
length(denied.ind)*0.12))
#valid.ind <- c(sample(setdiff(denied.ind, train.ind),length(denied.ind)*0.06),
sample(setdiff(certified.ind, train.ind),length(denied.ind)*0.06))
#test.ind <- c(sample(setdiff(denied.ind,
c(train.ind,valid.ind)),length(denied.ind)*0.06), sample(setdiff(certified.ind,
c(train.ind,valid.ind)),length(denied.ind)*0.06))

# Reset data with the complete cases without outliers & initialize
cmp.df_cdwo_outlier <- df_cdwo_outlier[complete.cases(df_cdwo_outlier),]
cmp.df_cdwo_outlier$CASE_STATUS <- 1*(cmp.df_cdwo_outlier$CASE_STATUS == "CERTIFIED")
cmp.df_cdwo_outlier[,c("CASE_NUMBER","CASE_SUBMITTED","DECISION_DATE","EMPLOYMENT_STAR
T_DATE","EMPLOYMENT_END_DATE","EMPLOYER_NAME","EMPLOYER_CITY","EMPLOYER_STATE","AGENT_
ATTORNEY_NAME",

"PW_UNIT_OF_PAY","PW_SOURCE_OTHER","WAGE_UNIT_OF_PAY","WORKSITE_CITY","WORKSITE_STATE"
)] <- NULL
cmp.df_cdwo_outlier$WAGE_DIFF <- cmp.df_cdwo_outlier$WAGE_RATE_OF_PAY_FROM -
cmp.df_cdwo_outlier$PREVAILING_WAGE
cmp.df_cdwo_outlier <- createDummyFeatures(cmp.df_cdwo_outlier, cols =
c("VISA_CLASS","SOC_CODE","NAICS_CODE","PW_WAGE_LEVEL","PW_SOURCE","PW_SOURCE_YEAR"),
method = "reference")
# Assgin to train, validation and test data sets
train.df <- cmp.df_cdwo_outlier[train.ind,]
valid.df <- cmp.df_cdwo_outlier[valid.ind,]
test.df <- cmp.df_cdwo_outlier[test.ind,]
# Normalize continuous numerical variables of the train, validation and test data sets
conum.ind <- which(names(train.df) %in%
c("TOTAL_WORKERS","NEW_EMPLOYMENT","CONTINUED_EMPLOYMENT","CHANGE_PREVIOUS_EMPLOYMENT"
,"NEW_CURRENT_EMPLOYMENT","CHANGE_EMPLOYER",

"AMENDED_PETITION","PREVAILING_WAGE","WAGE_RATE_OF_PAY_FROM","WAGE_DIFF"))
normalization.model <- preProcess(train.df[,conum.ind], method = c("center","scale"))
train.df.norm <- train.df
valid.df.norm <- valid.df
test.df.norm <- test.df
train.df.norm[,conum.ind] <- predict(normalization.model, train.df[,conum.ind])
valid.df.norm[,conum.ind] <- predict(normalization.model, valid.df[,conum.ind])
test.df.norm[,conum.ind] <- predict(normalization.model, test.df[,conum.ind])
summary(train.df.norm)

cat.val.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5), LR
= rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
cat.tst.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5), LR
= rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
#Base prediction
confusion.val <- confusionMatrix((valid.df$WAGE_RATE_OF_PAY_FROM >=
valid.df$PREVAILING_WAGE)*1, valid.df$CASE_STATUS, positive = "1")
confusion.test <- confusionMatrix((test.df$WAGE_RATE_OF_PAY_FROM >=
test.df$PREVAILING_WAGE)*1, test.df$CASE_STATUS, positive = "1")
cat.val.Accuracy$Model <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
cat.tst.Accuracy$Model <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

confusion.val <- confusionMatrix(rep(1,length(valid.ind)), valid.df$CASE_STATUS,
positive = "1")
confusion.test <- confusionMatrix(rep(1,length(test.ind)), test.df$CASE_STATUS,
positive = "1")
cat.val.Accuracy$Rand <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
cat.tst.Accuracy$Rand <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

#Tree
tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
cat.val.Accuracy$TR <- tmp[,1]
cat.tst.Accuracy$TR <- tmp[,2]

#Logistic regression
tmp <- logreg(train.df.norm, valid.df.norm, test.df.norm)
#tmp <- logreg(train.df.norm[,-c(72,83,88)], valid.df.norm[,-c(72,83,88)],
test.df.norm[,-c(72,83,88)])
#tmp <- logreg(train.df.norm[,-43], valid.df.norm[,-43], test.df.norm[,-43])
#tmp <- logreg(train.df.norm[,c(1:73,75:82,84:87,89:92)],
valid.df.norm[,c(1:73,79:82,84:87,89:92)], test.df.norm[,c(1:73,79:82,84:87,89:92)])
#Warnings ignored
#tmp <- logreg(train.df.norm[,c(1:42,44:77,79:82,84:87,89:107)],
valid.df.norm[,c(1:42,44:77,79:82,84:87,89:107)],
test.df.norm[,c(1:42,44:77,79:82,84:87,89:107)]) #Warnings ignored
#tmp <- logreg(train.df.norm[,c(1:49,52:75,77:85,87:96)],
valid.df.norm[,c(1:49,52:75,77:85,87:96)], test.df.norm[,c(1:49,52:75,77:85,87:96)])
#Warnings ignored
cat.val.Accuracy$LR <- tmp[,1]
cat.tst.Accuracy$LR <- tmp[,2]

#knn
```

```
tmp <- knearn(train.df.norm, valid.df.norm, test.df.norm)
cat.val.Accuracy$KNN <- tmp[,1]
cat.tst.Accuracy$KNN <- tmp[,2]

#Neural net
tmp <- nnet_opt(1,train.df.norm, valid.df.norm, test.df.norm) #Due to a warning or
error, may need to run one or a couple more.
#tmp <- nnet_opt(1,train.df.norm, valid.df.norm, test.df.norm)
cat.val.Accuracy$NN1 <- tmp[,1]
cat.tst.Accuracy$NN1 <- tmp[,2]

tmp <- nnet_opt(2,train.df.norm, valid.df.norm, test.df.norm)
cat.val.Accuracy$NN2 <- tmp[,1]
cat.tst.Accuracy$NN2 <- tmp[,2]

tmp <- nnet_opt(3,train.df.norm, valid.df.norm, test.df.norm)
cat.val.Accuracy$NN3 <- tmp[,1]
cat.tst.Accuracy$NN3 <- tmp[,2]


###############################################
# (2) Category into numerical frequency variables #
###############################################
cmp.df_cdwo_outlier <- df_cdwo_outlier[complete.cases(df_cdwo_outlier),]
cmp.df_cdwo_outlier$CASE_STATUS <- 1*(cmp.df_cdwo_outlier$CASE_STATUS == "CERTIFIED")
cmp.df_cdwo_outlier[,c("CASE_NUMBER","CASE_SUBMITTED","DECISION_DATE","EMPLOYMENT_STAR
T_DATE","EMPLOYMENT_END_DATE","EMPLOYER_CITY","EMPLOYER_STATE",
#"AGENT_ATTORNEY_NAME","EMPLOYER_NAME",

"PW_UNIT_OF_PAY","PW_SOURCE_OTHER","WAGE_UNIT_OF_PAY","WORKSITE_CITY","WORKSITE_STATE"
)] <- NULL
cmp.df_cdwo_outlier$WAGE_DIFF <- cmp.df_cdwo_outlier$WAGE_RATE_OF_PAY_FROM -
cmp.df_cdwo_outlier$PREVAILING_WAGE
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, VISA_CLASS =
ave(seq(nrow(cmp.df_cdwo_outlier)), VISA_CLASS, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, SOC_CODE =
ave(seq(nrow(cmp.df_cdwo_outlier)), SOC_CODE, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, NAICS_CODE =
ave(seq(nrow(cmp.df_cdwo_outlier)), NAICS_CODE, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, PW_WAGE_LEVEL =
ave(seq(nrow(cmp.df_cdwo_outlier)), PW_WAGE_LEVEL, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, PW_SOURCE =
ave(seq(nrow(cmp.df_cdwo_outlier)), PW_SOURCE, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, PW_SOURCE_YEAR =
ave(seq(nrow(cmp.df_cdwo_outlier)), PW_SOURCE_YEAR, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, H1B_DEPENDENT =
ave(seq(nrow(cmp.df_cdwo_outlier)), H1B_DEPENDENT, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, SUPPORT_H1B =
ave(seq(nrow(cmp.df_cdwo_outlier)), SUPPORT_H1B, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, EMPLOYER_NAME =
ave(seq(nrow(cmp.df_cdwo_outlier)), EMPLOYER_NAME, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, AGENT_ATTORNEY_NAME =
ave(seq(nrow(cmp.df_cdwo_outlier)), AGENT_ATTORNEY_NAME, FUN=length))
train.df <- cmp.df_cdwo_outlier[train.ind,]
valid.df <- cmp.df_cdwo_outlier[valid.ind,]
test.df <- cmp.df_cdwo_outlier[test.ind,]

normalization.model <- preProcess(train.df[,-1], method = c("scale","center"))
train.df.norm <- train.df
valid.df.norm <- valid.df
test.df.norm <- test.df
train.df.norm[,-1] <- predict(normalization.model, train.df[,-1])
valid.df.norm[,-1] <- predict(normalization.model, valid.df[,-1])
test.df.norm[,-1] <- predict(normalization.model, test.df[,-1])
summary(train.df.norm)

num.val.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5), LR
= rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
num.tst.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5), LR
= rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
#Base prediction
```

```
confusion.val <- confusionMatrix((valid.df$WAGE_RATE_OF_PAY_FROM >=
valid.df$PREVAILING_WAGE)*1, valid.df$CASE_STATUS, positive = "1")
confusion.test <- confusionMatrix((test.df$WAGE_RATE_OF_PAY_FROM >=
test.df$PREVAILING_WAGE)*1, test.df$CASE_STATUS, positive = "1")
num.val.Accuracy$Model <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
num.tst.Accuracy$Model <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

confusion.val <- confusionMatrix(rep(1,length(valid.ind)), valid.df$CASE_STATUS,
positive = "1")
confusion.test <- confusionMatrix(rep(1,length(test.ind)), test.df$CASE_STATUS,
positive = "1")
num.val.Accuracy$Rand <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
num.tst.Accuracy$Rand <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

#Tree
tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
num.val.Accuracy$TR <- tmp[,1]
num.tst.Accuracy$TR <- tmp[,2]

#Logistic regression
tmp <- logreg(train.df.norm, valid.df.norm, test.df.norm) #need to check 1:17 and
513:516
num.val.Accuracy$LR <- tmp[,1]
num.tst.Accuracy$LR <- tmp[,2]

#knn
tmp <- knearn(train.df.norm, valid.df.norm, test.df.norm)
num.val.Accuracy$KNN <- tmp[,1]
num.tst.Accuracy$KNN <- tmp[,2]

#Neural net
tmp <- nnet_opt(1,train.df.norm, valid.df.norm, test.df.norm)
num.val.Accuracy$NN1 <- tmp[,1]
num.tst.Accuracy$NN1 <- tmp[,2]

tmp <- nnet_opt(2,train.df.norm, valid.df.norm, test.df.norm)
num.val.Accuracy$NN2 <- tmp[,1]
num.tst.Accuracy$NN2 <- tmp[,2]

tmp <- nnet_opt(3,train.df.norm, valid.df.norm, test.df.norm)
num.val.Accuracy$NN3 <- tmp[,1]
num.tst.Accuracy$NN3 <- tmp[,2]







##################################################
#         (3) Unbalanced raw sampling            #
# Treating category variables by dummy variables #
##################################################
cmp.df_cdwo_outlier <- df_cdwo_outlier[complete.cases(df_cdwo_outlier),]
length.cmp <- dim(cmp.df_cdwo_outlier)[1]
######### With 2018 Q1 data
train.ind <- sample(1:length.cmp, length(denied.ind)*0.5)
valid.ind <- sample(setdiff(1:length.cmp, train.ind),length(denied.ind)*0.25)
test.ind <- sample(setdiff(1:length.cmp,
c(train.ind,valid.ind)),length(denied.ind)*0.25)

#train.ind <- sample(1:length.cmp, length(denied_though.ind))
#valid.ind <- sample(setdiff(1:length.cmp, train.ind),length(denied_though.ind)*0.5)
#test.ind <- sample(setdiff(1:length.cmp,
c(train.ind,valid.ind)),length(denied_though.ind)*0.5)
######## With 2017 data
#train.ind <- sample(1:length.cmp, length(denied.ind)*0.24)
#valid.ind <- sample(setdiff(1:length.cmp, train.ind),length(denied.ind)*0.12)
#test.ind <- sample(setdiff(1:length.cmp,
c(train.ind,valid.ind)),length(denied.ind)*0.12)
```

```
# Reset data with the complete cases without outliers & initialize
cmp.df_cdwo_outlier <- df_cdwo_outlier[complete.cases(df_cdwo_outlier),]
cmp.df_cdwo_outlier$CASE_STATUS <- 1*(cmp.df_cdwo_outlier$CASE_STATUS == "CERTIFIED")
cmp.df_cdwo_outlier[,c("CASE_NUMBER","CASE_SUBMITTED","DECISION_DATE","EMPLOYMENT_STAR
T_DATE","EMPLOYMENT_END_DATE","EMPLOYER_NAME","EMPLOYER_CITY","EMPLOYER_STATE","AGENT_
ATTORNEY_NAME",

"PW_UNIT_OF_PAY","PW_SOURCE_OTHER","WAGE_UNIT_OF_PAY","WORKSITE_CITY","WORKSITE_STATE"
)] <- NULL
cmp.df_cdwo_outlier$WAGE_DIFF <- cmp.df_cdwo_outlier$WAGE_RATE_OF_PAY_FROM -
cmp.df_cdwo_outlier$PREVAILING_WAGE
cmp.df_cdwo_outlier <- createDummyFeatures(cmp.df_cdwo_outlier, cols =
c("VISA_CLASS","SOC_CODE","NAICS_CODE","PW_WAGE_LEVEL","PW_SOURCE","PW_SOURCE_YEAR"),
method = "reference")
# Assgin to train, validation and test data sets
train.df <- cmp.df_cdwo_outlier[train.ind,]
valid.df <- cmp.df_cdwo_outlier[valid.ind,]
test.df <- cmp.df_cdwo_outlier[test.ind,]
# Normalize continuous numerical variables of the train, validation and test data sets
conum.ind <- which(names(train.df) %in%
c("TOTAL_WORKERS","NEW_EMPLOYMENT","CONTINUED_EMPLOYMENT","CHANGE_PREVIOUS_EMPLOYMENT"
,"NEW_CURRENT_EMPLOYMENT","CHANGE_EMPLOYER",

"AMENDED_PETITION","PREVAILING_WAGE","WAGE_RATE_OF_PAY_FROM","WAGE_DIFF"))
normalization.model <- preProcess(train.df[,conum.ind], method = c("center","scale"))
train.df.norm <- train.df
valid.df.norm <- valid.df
test.df.norm <- test.df
train.df.norm[,conum.ind] <- predict(normalization.model, train.df[,conum.ind])
valid.df.norm[,conum.ind] <- predict(normalization.model, valid.df[,conum.ind])
test.df.norm[,conum.ind] <- predict(normalization.model, test.df[,conum.ind])
summary(train.df.norm)

###################
# Prediction calculation #
###################
ub.cat.val.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5),
LR = rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
ub.cat.tst.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5),
LR = rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
#Base prediction
confusion.val <- confusionMatrix((valid.df$WAGE_RATE_OF_PAY_FROM >=
valid.df$PREVAILING_WAGE)*1, valid.df$CASE_STATUS, positive = "1")
confusion.test <- confusionMatrix((test.df$WAGE_RATE_OF_PAY_FROM >=
test.df$PREVAILING_WAGE)*1, test.df$CASE_STATUS, positive = "1")
ub.cat.val.Accuracy$Model <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
ub.cat.tst.Accuracy$Model <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

confusion.val <- confusionMatrix(rep(1,length(valid.ind)), valid.df$CASE_STATUS,
positive = "1")
confusion.test <- confusionMatrix(rep(1,length(test.ind)), test.df$CASE_STATUS,
positive = "1")
ub.cat.val.Accuracy$Rand <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
ub.cat.tst.Accuracy$Rand <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

#Tree
tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
ub.cat.val.Accuracy$TR <- tmp[,1]
ub.cat.tst.Accuracy$TR <- tmp[,2]

#Logistic regression
tmp <- logreg(train.df.norm, valid.df.norm, test.df.norm) #Warnings ignored
#tmp <- logreg(train.df.norm[,-c(50:51,72,83,88)], valid.df.norm[,-c(50:51,72,83,88)],
test.df.norm[,-c(50:51,72,83,88)])
#tmp <- logreg(train.df.norm[,c(1:75,78,81:88,90:92)],
valid.df.norm[,c(1:75,78,81:88,90:92)], test.df.norm[,c(1:75,78,81:88,90:92)])
#Warnings ignored
#tmp <- logreg(train.df.norm[,c(1:49,52:75,77:85,87:92)],
valid.df.norm[,c(1:49,52:75,77:85,87:92)], test.df.norm[,c(1:49,52:75,77:85,87:92)])
#Warnings ignored
```

```
#tmp <- logreg(train.df.norm[,c(1:49,52:75,77:85,87:96)],
valid.df.norm[,c(1:49,52:75,77:85,87:96)], test.df.norm[,c(1:49,52:75,77:85,87:96)])
#Warnings ignored
ub.cat.val.Accuracy$LR <- tmp[,1]
ub.cat.tst.Accuracy$LR <- tmp[,2]

#knn
tmp <- knearn(train.df.norm, valid.df.norm, test.df.norm)
ub.cat.val.Accuracy$KNN <- tmp[,1]
ub.cat.tst.Accuracy$KNN <- tmp[,2]

#Neural net
tmp <- nnet_opt(1,train.df.norm, valid.df.norm, test.df.norm) #Due to a warning or
error, may need to run one or a couple more.
#tmp <- nnet_opt(1,train.df.norm, valid.df.norm, test.df.norm)
ub.cat.val.Accuracy$NN1 <- tmp[,1]
ub.cat.tst.Accuracy$NN1 <- tmp[,2]

tmp <- nnet_opt(2,train.df.norm, valid.df.norm, test.df.norm)
ub.cat.val.Accuracy$NN2 <- tmp[,1]
ub.cat.tst.Accuracy$NN2 <- tmp[,2]

tmp <- nnet_opt(3,train.df.norm, valid.df.norm, test.df.norm)
ub.cat.val.Accuracy$NN3 <- tmp[,1]
ub.cat.tst.Accuracy$NN3 <- tmp[,2]



##################################################
#         (4) Unbalanced raw sampling            #
# Treating category variables by using frequency #
##################################################
cmp.df_cdwo_outlier <- df_cdwo_outlier[complete.cases(df_cdwo_outlier),]
cmp.df_cdwo_outlier$CASE_STATUS <- 1*(cmp.df_cdwo_outlier$CASE_STATUS == "CERTIFIED")
cmp.df_cdwo_outlier[,c("CASE_NUMBER","CASE_SUBMITTED","DECISION_DATE","EMPLOYMENT_STAR
T_DATE","EMPLOYMENT_END_DATE","EMPLOYER_CITY","EMPLOYER_STATE",
#"AGENT_ATTORNEY_NAME","EMPLOYER_NAME",

"PW_UNIT_OF_PAY","PW_SOURCE_OTHER","WAGE_UNIT_OF_PAY","WORKSITE_CITY","WORKSITE_STATE"
)] <- NULL
cmp.df_cdwo_outlier$WAGE_DIFF <- cmp.df_cdwo_outlier$WAGE_RATE_OF_PAY_FROM -
cmp.df_cdwo_outlier$PREVAILING_WAGE
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, VISA_CLASS =
ave(seq(nrow(cmp.df_cdwo_outlier)), VISA_CLASS, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, SOC_CODE =
ave(seq(nrow(cmp.df_cdwo_outlier)), SOC_CODE, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, NAICS_CODE =
ave(seq(nrow(cmp.df_cdwo_outlier)), NAICS_CODE, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, PW_WAGE_LEVEL =
ave(seq(nrow(cmp.df_cdwo_outlier)), PW_WAGE_LEVEL, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, PW_SOURCE =
ave(seq(nrow(cmp.df_cdwo_outlier)), PW_SOURCE, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, PW_SOURCE_YEAR =
ave(seq(nrow(cmp.df_cdwo_outlier)), PW_SOURCE_YEAR, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, H1B_DEPENDENT =
ave(seq(nrow(cmp.df_cdwo_outlier)), H1B_DEPENDENT, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, SUPPORT_H1B =
ave(seq(nrow(cmp.df_cdwo_outlier)), SUPPORT_H1B, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, EMPLOYER_NAME =
ave(seq(nrow(cmp.df_cdwo_outlier)), EMPLOYER_NAME, FUN=length))
cmp.df_cdwo_outlier <- transform(cmp.df_cdwo_outlier, AGENT_ATTORNEY_NAME =
ave(seq(nrow(cmp.df_cdwo_outlier)), AGENT_ATTORNEY_NAME, FUN=length))
train.df <- cmp.df_cdwo_outlier[train.ind,]
valid.df <- cmp.df_cdwo_outlier[valid.ind,]
test.df <- cmp.df_cdwo_outlier[test.ind,]

normalization.model <- preProcess(train.df[,-1], method = c("scale","center"))
train.df.norm <- train.df
valid.df.norm <- valid.df
test.df.norm <- test.df
train.df.norm[,-1] <- predict(normalization.model, train.df[,-1])
```

```
valid.df.norm[,-1] <- predict(normalization.model, valid.df[,-1])
test.df.norm[,-1] <- predict(normalization.model, test.df[,-1])
summary(train.df.norm)

ub.num.val.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5),
LR = rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
ub.num.tst.Accuracy <- data.frame(Model = rep(NA,5), Rand = rep(NA,5), TR = rep(NA,5),
LR = rep(NA,5), KNN = rep(NA,5), NN1 = rep(NA,5), NN2 = rep(NA,5), NN3 = rep(NA,5))
#Base prediction
confusion.val <- confusionMatrix((valid.df$WAGE_RATE_OF_PAY_FROM >=
valid.df$PREVAILING_WAGE)*1, valid.df$CASE_STATUS, positive = "1")
confusion.test <- confusionMatrix((test.df$WAGE_RATE_OF_PAY_FROM >=
test.df$PREVAILING_WAGE)*1, test.df$CASE_STATUS, positive = "1")
ub.num.val.Accuracy$Model <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
ub.num.tst.Accuracy$Model <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

confusion.val <- confusionMatrix(rep(1,length(valid.ind)), valid.df$CASE_STATUS,
positive = "1")
confusion.test <- confusionMatrix(rep(1,length(test.ind)), test.df$CASE_STATUS,
positive = "1")
ub.num.val.Accuracy$Rand <- c(confusion.val$overall[1], confusion.val$byClass[1:4])
ub.num.tst.Accuracy$Rand <- c(confusion.test$overall[1], confusion.test$byClass[1:4])

#Tree
tmp <- tree(train.df.norm, valid.df.norm, test.df.norm)
ub.num.val.Accuracy$TR <- tmp[,1]
ub.num.tst.Accuracy$TR <- tmp[,2]

#Logistic regression
tmp <- logreg(train.df.norm, valid.df.norm, test.df.norm) #need to check 1:17 and
513:516
ub.num.val.Accuracy$LR <- tmp[,1]
ub.num.tst.Accuracy$LR <- tmp[,2]

#knn
tmp <- knearn(train.df.norm, valid.df.norm, test.df.norm)
ub.num.val.Accuracy$KNN <- tmp[,1]
ub.num.tst.Accuracy$KNN <- tmp[,2]

#Neural net
tmp <- nnet_opt(1,train.df.norm, valid.df.norm, test.df.norm)
ub.num.val.Accuracy$NN1 <- tmp[,1]
ub.num.tst.Accuracy$NN1 <- tmp[,2]

tmp <- nnet_opt(2,train.df.norm, valid.df.norm, test.df.norm)
ub.num.val.Accuracy$NN2 <- tmp[,1]
ub.num.tst.Accuracy$NN2 <- tmp[,2]

tmp <- nnet_opt(3,train.df.norm, valid.df.norm, test.df.norm)
ub.num.val.Accuracy$NN3 <- tmp[,1]
ub.num.tst.Accuracy$NN3 <- tmp[,2]
```