BIA 610 Project-Final Report

Pooja Kalra, Janit Modi, Jaebin Park, Sandeep Kumar Moparthy

1. Business Understanding:

Deciding a place for one's new start up or establishing a new unit of an existing business can be a difficult task. A significant number of factors have to be considered in making such an important decision. This project gives an ability, for decision makers, to arrive at an informed decision on which industry yields the best in which state, in which county and in which city. This data analytical model solution based on CBP help decision makers in a company to

- Find out potential business locations
- Identify investment opportunities

In the final outcome, we intend to fully automate the process, where the client only has to provide a few factors, such an intended average payroll, the establishment capacity, or the industry type that they're involved in, and our model will run through the entire data set and provide us with all the necessary information.

This data analytic solution gives a high, medium and low potential location for a business based on 2 key metrics namely: underlying industry and the annual payroll specific to employee count in the establishment. This gives the company the opportunity to improve their strategies by analyzing the location and employee size.

Our analysis on County level average payroll, state average payroll and US average payroll and it allowed us to segregate counties to low, medium and high payroll for investments. Locations corresponding to lower average payroll can benefit small startups. On the contrary, locations with higher average payroll can be investment heavens for larger firm who want to expand their business.

For the purpose of simplicity and demonstration, we have used data from the state of Texas (Fipstate Code 48) for Health care industry (Naics- 62---) specifically.

We have performed data preprocessing on 3 datasets. We have combined the State and County files by NAICS and the FIPS State Code, with which we used the NAICS to merge the US file to create a master set that was used for further modeling and analysis.

Once we had our master dataset, we created our response variable “Countybins” which categorizes the establishments into low, medium and large. We split the data into training and test sets . we fit the model using the training set and predicted using the test set.

Based on our analysis, our final model is 92% accurate.

2. Data Understanding

County Business Patterns (CBP), ZIP Code Business Patterns (ZBP) are an annual series that provide sub-national economic data by industry. These programs cover most of the U.S. economy and feature industry and geographic statistics which supplement those published in the Economic Census. Data are published at the U.S. level and by State, County, Metropolitan area, ZIP code, and Congressional District. All data are classified by an industry code (NAICS) and can be viewed with employment-size class breakouts by establishment, and by legal form of organization at some geographic levels. CBP covers most of the country's economic activity. The series excludes data on self-employed individuals, employees of private households, railroad employees, agricultural production employees, and most government employees.

The CBP annual series provides information that is critical for understanding the Nation's changing economic structure and performance. The series is used to study the economic activity of small areas, analyze economic changes over time; and as a benchmark for statistical series, surveys, and databases between economic censuses. Businesses use the data for analyzing market potential, measuring the effectiveness of sales and advertising programs, setting sales quotas, and developing budgets. Government agencies use the data for administration and planning. Statistics from these surveys are widely used by policy officials, economic analysts, business decision-makers, and the news media.

Since 1998, County Business Patterns has been tabulated based on the North American Industry Classification System (NAICS). Data were tabulated according to the Standard Industrial Classification (SIC) System for prior periods.

- 2012 to 2016 data use NAICS 2012
- 2008 to 2011 data use NAICS 2007
- 2003 to 2007 data use NAICS 2002
- 1998 to 2002 data use NAICS 1997
- 1988 to 1997 data use 1987 SIC
- 1974 to 1987 data use 1972 SIC

Prior to 2012, County Business Patterns lagged by one year in the adoption of the classification system employed in the Economic Census. Starting in 2012, the classification system was changed in the same year.

There are eight different datasets that considered for this project. These datasets, when integrated and work with intellectually can yield in interesting insights.

Datasets

Complete Congressional District File [<1.0 MB]

Complete County File [15.6 MB]

Complete Metropolitan Area File [7.5 MB]

Complete State File [10.9 MB]

Complete U.S. File [<1.0 MB]

Complete ZIP Code Industry Detail File [28.2 MB]

Complete ZIP Code Totals File [<1.0 MB]

CBP and NES Combined Report [14.3 MB]

The CBP and NES Combined Report is CSV file with 822289 rows and 19 variables.

The Complete County File is a CSV file with 2124893 rows and 26 variables.

The complete metropolitan area file is a CSV file with 936105 rows and 23 variables

The complete state file is a csv file with 448310 rows and 84 variables

The complete US file is a csv file with 13002 rows and 83 variables

The complete ZIP code industry Detail file is a csv with 8418283 rows and 12 variables

The Complete Zipcode totals File is a csv with 38722 rows and 13 variable

3. Data Preparation:

As mentioned in the project proposal, the County Business Partners included eight different datasets. For the purposes of our analysis, we decided to focus on three: the Complete County, State, and US files. Each dataset included around 20 columns, many of which were either irrelevant or unusable. For example, many features, such as Annual Payroll for the Industry, had supporting variables, like Annual Payroll Noise Flag, that was formatted in way that made it difficult to include in the analysis. Thus, we moved forward by removing these columns.

The next set of variables that needed managing were the Number of Establishments in each Industry by Employee Size Class. As it was dealing with size, there were at least a dozen columns, each representing a different size class, from 1-5 to 100-499 to 5000+. Because one of our objectives for the analysis is to provide companies with insight on not only where some industries thrive, but also what size companies are typically present in those areas, we decided to create three classes: small, medium, and large. Establishments with 1-99 employees were included in the small class, those with 100-999 employees in the medium class, and the rest in the large class. With this data, it could be possible to receive the output that companies want through classification.

Finally, in order to create a central feature that we can use to combine our multiple datasets, we needed to clean the NAICS industry codes. The issue was that these codes were given as factors with either multiple hyphens (“-“) or back slashes (“/”). To make this variable usable, we converted each index into a string, replaced the characters that were not numbers into an empty space, and converted it back into an integer. We ran into a problem when performing this function. Our dataset was too large (the County dataset has over two million rows) and it was

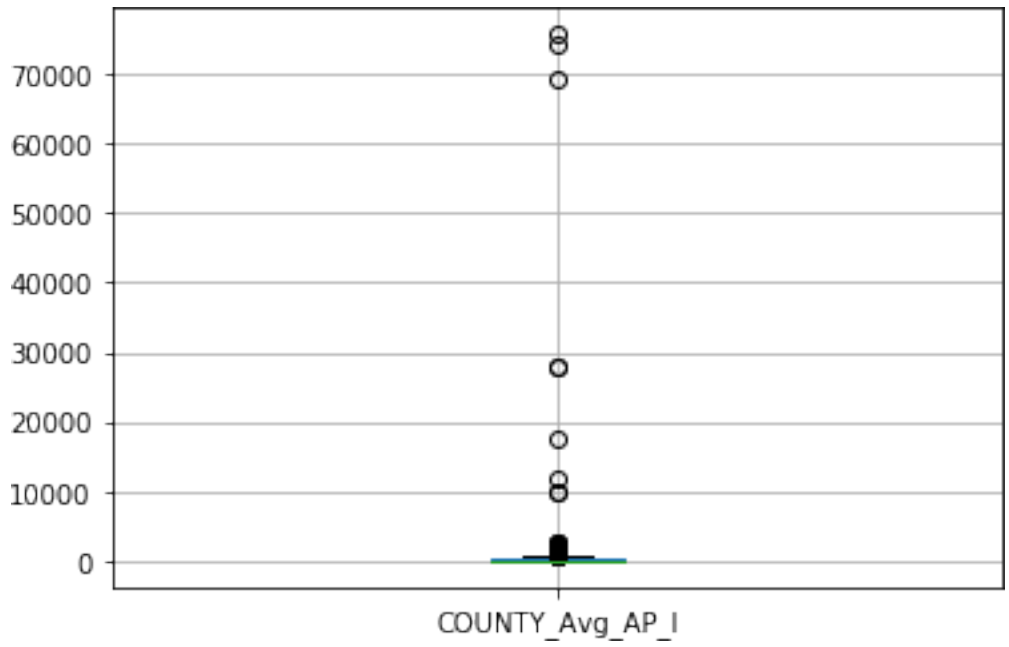
computationally taking too long. We decided to randomly sample our dataset to *1/sqrt(n)* to make it much more manageable.

We performed these steps for the three datasets. Because some datasets have features that others don't, some data specific preprocessing needed to be done. After which, we combined the State and County files by NAICS and the FIPS State Code, with which we used the NAICS to merge the US file to create a master set that will be using for further modeling and analysis. Once we had our master dataset, some variables were no longer necessary, leading us to drop the features NACIS, State_lfo, and US_lfo. We concluded processing the data by creating separate list that contained the Average Annual Payroll for the County, State, and US. The average was calculated by the dividing the annual pay roll by the number of establishments in the industry

After the initial data processing and exploring exercise we narrowed down our focus to the state of Texas as it had the greatest number of counties. In order to execute these activities, we used the python library of pandas and functools to merge the datasets.

The key purpose of our analysis is to identify counties that best support a specific field of business. For this particular report we have focused on the health care industry and tried to identify counties that are most profitable for this industry. In order to achieve this, we had to identify the 'naics' code that uniquely identifies health care businesses. After closely observing the dataset we realized that the 'naics' codes that begin with the number '62' relate to the healthcare industry. Based on this observation we filtered the 'naics' column by first converting it into a string and later using the startswith() function to get all the rows that begin with '62'.

Using Recursive Feature Elimination, we eliminated features that did not strongly contribute to the analysis. Allowing us to narrow down our analysis to 22 integral features and build a model that accurately gives the required insights.



- ***Recursive feature elimination (RFE)***

RFE is a feature selection method that fits a model and removes the weakest feature (or features) until the specified number of features is reached. Features are ranked by the model's coef_ or feature_importances_ attributes, and by recursively eliminating a small number of features per loop, RFE attempts to eliminate dependencies and collinearity that may exist in the model.

RFE requires a specified number of features to keep, however it is often not known in advance how many features are valid. To find the optimal number of features cross-validation is used with RFE to score different feature subsets and select the best scoring collection of features. The RFECV visualizer plots the number of features in the model along with their cross-validated test score and variability and visualizes the selected number of features.

Our final dataset includes the following features:

State_lfo_- : State- legal form of organization

state_lfo_Z : State legal form of organization S-Corporations

US_lfo_- : US legal form of organization

US_lfo_G : US legal form of organization Government

US_lfo_O : US legal form of organization Other

US_lfo_Z : US legal form of organization S-Corporations

COUNTY_emp : County total mid-march employees

COUNTY_qp1 : County first quarter payroll

COUNTY_ap : County annual payroll

COUNTY_est : County total number of establishments

COUNTY_sz_small : County establishment class size, small

COUNTY_sz_med : County establishment class size, medium

state_qp1: State first quarter payroll

state_ap: State annual payroll

state_est: State total number of establishments

state_sz_small: State establishment class size, small

state_sz_med: State establishment class size, medium

state_sz_lrg: State establishment class size, large

US_sz_small: US establishment class size, small

US_sz_med: UA establishment class size, medium

US_sz_lrg: US establishment class size, large

4. Modeling:

We decided to create an initial model with just the variables regarding Counties from the state Texas. We fitted various models on the Average Payroll for the Counties namely:

- **Logistic Regression:** It is used to predict the probability of a categorical dependent variable and is a Machine Learning classification algorithm. The dependent variable in logistic regression is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.). In other words, the model predicts P(Y=1) as a function of X. We have used logistic regression to train our dataset and predict response variable "countybins".

- **Decision tree:** Decision tree builds a classification model based on the structure of a tree. It granulates a dataset into smaller and smaller subsets while simultaneously an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. We have used Decision Tree Classifier and fitted the model on data and target.

- **Random Forest**: A Random Forest is an ensemble technique that uses multiple decision trees and a technique called Bootstrap Aggregation which is capable of performing both regression and classification tasks simultaneously. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees.

- **Neural networks**: They are a collection of algorithms, modeled loosely after the human brain, that are modeled to recognize patterns. They interpret sensory data through a kind of machine perception, labeling or clustering raw input.

  Our goal is to find the optimized value. there are different types of optimizers such as optimizer = ['SGD', 'RMSprop', 'Adagrad', 'Adadelta', 'Adam', 'Adamax', 'Nadam']

- **RMSProp is Root Mean Square Propagation**

  RMSProp tries to resolve Adagrad's radically diminishing learning rates by using a moving average of the squared gradient. It utilizes the magnitude of the recent gradient descents to normalize the gradient

- **Adagrad — Adaptive Gradient Algorithm**

  We perform larger updates for infrequent parameters and smaller updates for frequent parameters.

- **Adadelta**

  Adadelta is an extension of Adagrad and it also tries to reduce Adagrad's aggressive, monotonically reducing the learning rate. It does this by restricting the window of the past accumulated gradient to some fixed size of w. Running average at time t then depends on the previous average and the current gradient

- **Adam — Adaptive Moment Estimation**
  Calculates the individual adaptive learning rate for each parameter from estimates of first and second moments of the gradients.

- **Nadam- Nesterov-accelerated Adaptive Moment Estimation**
  Nadam is employed for noisy gradients or for gradients with high curvatures. We have used Neural Networks with different optimizers and grid search patterns to fit our model.

- **GridSearch CV**

  It is the process of performing hyper parameter tuning in order to determine the optimal values for a given model. This is significant as the performance of the entire model is based on the hyper parameter values specified.To stipulate values for hyper parameters, we have used the module such as GridSearchCV of the sklearn library. The estimator parameter of GridSearchCV requires the model we are using for the hyper parameter tuning process. The param_grid parameter requires a list of parameters and the range of values for each parameter of the specified *estimator*. Using grid search CV we further filtered the parameters and based our analysis on the best features. We have used 'sigmoid' activation for this analysis.

- Naive Bayes classifiers are a collection of classification algorithms based on **Bayes' Theory:** It is a collection of algorithms which share a common principle, i.e. every pair of features being classified is independent of each other. We have also used Naive Bayes to train our dataset.
- K Nearest Neighbor: KNN makes prediction on the training dataset and does not require learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.

- **Support Vector Machine** (**SVM**): It is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples. We have used 3 kernel implementations of support vector machine namely,
    - **Linear:** A linear kernel can be used as normal dot product any two given observations. The product between two vectors is the sum of the multiplication of each pair of input values.
    - **Polynomial Kernel:** A polynomial kernel is a more generalized form of the linear kernel. The polynomial kernel can distinguish curved or nonlinear input space
    - **Radial Basis Function Kernel:** The kernel function is a measure of similarity between two sets of features. RBF can map an input space in infinite dimensional space.

5. Evaluation

Below are the evaluation metrics we have used to compare our model efficiencies and pick the best mode

Accuracy, Precision, and Recall:

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Accuracy

Accuracy is the quintessential classification metric. It is pretty easy to understand and easily suited for binary as well as a multiclass classification problem.

Accuracy = (TP+TN)/(TP+FP+FN+TN)

Accuracy is the proportion of true results among the total number of cases examined.

Accuracy is a valid choice of evaluation for classification problems which are well balanced and not skewed or No class imbalance.

B. Precision

Let's start with *precision*, which answers the following question: what proportion of **predicted Positives** is truly Positive?

Precision = (TP)/(TP+FP)

In the asteroid prediction problem, we never predicted a true positive.

And thus precision=0

Precision is a valid choice of evaluation metric when we want to be very sure of our prediction. For example: If we are building a system to predict if we should decrease the credit limit on a particular account, we want to be very sure about our prediction or it may result in customer dissatisfaction.

C. Recall

Another very useful measure is *recall*, which answers a different question: what proportion of actual Positives is correctly classified?

Recall = (TP)/(TP+FN)

In the asteroid prediction problem, we never predicted a true positive.

And thus recall is also equal to 0.

Recall is a valid choice of evaluation metric when we want to capture as many positives as possible. For example: If we are building a system to predict if a person has cancer or not, we want to capture the disease even if we are not very sure.

6. F1 Score:

This is a popular *evaluation metric* and is often used extensively in classification projects.

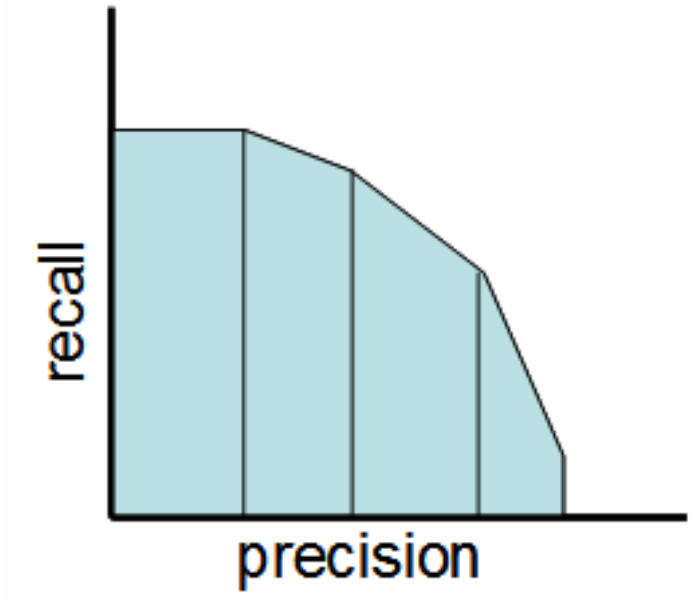
$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

For the sake of explanation let us start with a binary prediction problem. ***We are predicting if an asteroid will hit the earth or not.***

So if we say "No" for the whole training set. Our precision here is 0. What is the recall of our positive class? It is zero. What is the accuracy? It is more than 99%.

And hence the F1 score is also 0. And thus we get to know that the classifier that has an accuracy of 99% is basically worthless for our case. And hence it solves our problem.

We want to have a model with both good precision and recall.



Precision-Recall Tradeoff

Simply stated the ***F1 score sort of maintains a balance between the precision and recall for your classifier***. If your precision is low, the F1 is low and if the recall is low again your F1 score is low.

4. Categorical Crossentropy

The log loss is a generic solution for multiclass problem. The classifier in a multiclass setting must assign a probability to each class. If there are N samples belonging to M classes, then the *Categorical Crossentropy* is the summation of -ylogp values:

$$LogarithmicLoss = \frac{-1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} * \log(p_{ij})$$

y_ij is 1 if the sample i belongs to class j else 0

p_ij is the probability our classifier predicts of sample i belonging to class j.

When the output of a classifier is multiclass prediction probabilities. We generally use Categorical Crossentropy in case of Neural Nets. In general, minimizing Categorical cross- entropy gives greater accuracy for the classifier.

Considering accuracy and recall scores as our measures to identify the best classifier for the data we have realized our model is sound. In the table below it can be observed that our logistic regression performs with an accuracy of 92 percent and a recall rate of 100 percent. We have used recall to justify any class imbalance and it also can be observed that there are other measures of performance that our final model satisfies.

Model	Accuracy	precision (0)	precision(1)	precision(2)	precision(3)	recall(0)	recall(1)	recall(2)	recall(3)	F1(0)	F1(1)	F1(2)	F1(3)
Logreg	0.92	0.96	0.89	0.68	0.93	1	0.89	0.59	0.88	0.98	0.89	0.63	0.90
RFC	0.89	1	0.88	0.43	0.79	1	0.89	0.45	0.69	1	0.89	0.44	0.73
NB	0.80	0.97	0.974	0.41	0.50	0.89	0.79	0.43	0.67	0.93	0.76	0.42	0.57
KNN	0.42	0.47	0.31	0.00	0.00	0.73	0.20	0.00	0.00	0.57	0.24	0.00	0.00
SVM-lin	0.64	0.73	0.59	0.33	1.00	0.77	0.59	0.41	0.31	0.75	0.59	0.37	0.48
SVM-rbf	0.30	0.46	0.38	0.08	0.07	0.35	0.30	0.18	0.12	0.40	0.34	0.11	0.09
Keras	0.51												

DT	0.92	1	0.90	0.63	0.82	1	0.92	0.55	0.88	1	0.91	0.59	0.85

Deployment

After successfully executing the above models we can classify client data into 4 categories namely,

0 - Extremely low

1- low

2 - medium

3 - high

This enables the client to identify locations to establish their business or a potential investment opportunity (based on client requirement, extending from low average payroll to high average payroll).

Based on the client requirement, we can filter the data we have and suggest which county is ideal for the client.

Discuss any issues the rm should be aware of regarding deployment

Client should be aware of the fact that recommendations should be very carefully chosen based on the client's capacity for the facility that is under consideration since the predictions are only suggestive in nature and do not guarantee success.

Ethical Considerations:

α. Quality: The client expects the product to be of good quality.

β. Data Privacy: Clients data will not be exposed to use for any other businesses or any third-party organizations.

χ. Accuracy: Our suggestions are 92% accurate with 95% confidence interval.

Risk

Below are the risks associated with our model.

1. Client data quality is poor

Mitigation: Conduct regular meetings with the client and communicate reports of corrupt data

2. External Factors (climate, governance):

Mitigation: Increase risk tolerance of the model.

3. Lack of adequate details in requirements

Mitigation: Thorough requirements gathering should be conducted initially and client should be kept in the loop at every step of development. Also, requirements should be formalized through proper documentation

Future scope:

The model, now, classifies the average payroll of a firm into one of the 4 levels based on the data for health care industry, provided in Texas state alone. The model can be extended to any particular industry and in state, metropolitan or country level for analysis that lead to informed decisions.

With time, the model can be fine-tuned to become an interface based interactive product that is attractive aesthetically and in performance.

The model not only is capable helping out an entrepreneur, but can also classify the states, counties of the United States into zones for census reporting and federal decisions. The model is an inspiration for many analytical models to be performed on such census data.

Contributions

The project had a fair share of contribution from each member of the team and it is difficult if not impossible to map the individual contribution of each member as we worked so cohesively. However, as the requirement of this report we have tried to segregate the project in work products and highlighted the contribution of each member as follows,

A) Data Cleaning and Preparation: This task was spearheaded by Jaebin Park and was assisted by Sandeep Moparthy and Janit Modi. Pooja Kalra assisted in identifying the important variables that could be considered for our analysis.

B) Modeling: Sandeep Moparthy initiated this activity and was assisted by Jaebin Park to identifying the models that could be executed on our data. Janit Modi and Pooja Kalra executed these models and tested the accuracy it achieved.

C) Evaluation: Pooja Kalra further evaluated the models on various parameters and was assisted by Sandeep Moparthy in executing these activities.

D) Report: The report was edited and compiled by Janit Modi while being assisted by Pooja Kalra and Sandeep Moparthy. Janit Modi formatted the report and added the necessary graphs for appropriate representation of the models.

E) Presentation: The final task was a truly collective in nature as every member worked on different topic of the presentation which was finally compiled by Jaebin Park.

Biblography:

https://www.scikit-yb.org/en/latest/api/model_selection/rfecv.html

https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python#kernels

https://www.geeksforgeeks.org

https://towardsdatascience.com/the-5-classification-evaluation-metrics-you-must-know-aa97784ff226

https://medium.com/datadriveninvestor/an-introduction-to-grid-search-ff57adcc0998

https://machinelearningmastery.com/k-nearest-neighbors-for-machine-learning/