Capstone Proposal **Pedro Felipe Gardeazábal Rodríguez**
Project: Predicting Salaries
Course: MLND - Udacity
Date: 05/20/2018

Predicting Salaries from H-1B Applications

Domain Background

Each year potential high skilled foreign students wanting to join the US job market graduate from US universities or employers look to import missing talent. For this purpose the US H-1B visa [1] was created. It allows employers to hire high skilled foreign workers by ensuring they will be paid the current prevailing wage or higher. This ensures that employers are fair to US nationals and are not looking to import cheaper workforce. However with the job market being competitive and the prevailing wages not well known, it is important for a job seeker or an employee to determine what a fair salary is. The prevailing wage is decided by the US Department of Labor from its wage library and it is not known in advance by the future employee or employer before the application for the H-1B visa starts.

Problem Statement

When entering the job market after college for the first time or when transitioning to a new career path it is difficult to assess what the base salary for a given position should be. This is particularly true for foreign workers when moving to a new area given that salaries do not only depend on the field of knowledge, but are also dependent on the location where the future employee is going to work. In order to help not only job applicants but also employers it will be helpful to develop a model predicting the base salary a future employee should expect based on historical data collected from US H-1B visa applications.

The data set used to solve this problem contains information on location, job position and field of knowledge. This information serves as the input to the model. The data set also has the corresponding prevailing wages and employer salaries, which are continuous target variables. Therefore, the problem at hand corresponds to a supervised regression problem that can be solved by linear regression [2] random forest regression [3].

Datasets and Inputs

For the model to be relevant in 2018 only H-1B visa applications from 2017 are considered. The data is obtained from the United States Department of Labor (USDL) and can be downloaded directly from here. From the raw file the columns of interest are:

- JOB_TITLE: Title of the job.
- SOC_NAME: Occupational name associated with the SOC_CODE.
- FULL_TIME_POSITION: 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).
- WAGE_RATE_OF_PAY_FROM: Employer's proposed wage rate.
- WAGE_UNIT_OF_PAY: Unit of pay. Valid values include Hour, Week, Bi-Weekly, Month, or Year.
- 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.

To cure the data, it was defined that the model is going to predict only annual salaries. For this reason, the entries PW_UNIT_OF_PAY, WAGE_UNIT_OF_PAY and FULL_TIME_POSITION were used to express all entries from the columns PREVAILING_WAGE and WAGE_RATE_OF_PAY_FROM as annual wages. The entries in 'WORKSITE_POSTAL_CODE' came in several formats, but in order to reduce its number of categories all entries were transformed to a five digit format.

After curing, the final cleaned data set retained the following columns where the names are self-explanatory: 'employer', 'job_title', 'occupational_name', 'prevailing_wage', 'pw_wage_period', 'employer_wage', 'employer_max_wage', 'employer_wage_period', 'city', 'county', 'state' and 'postal_code'.

Considering that salary depends on job title and job category as well as location, the columns 'job_title', 'occupational_name', 'city', 'county', 'state' and 'postal_code' are defined as the inputs.

Solution Statement

Since the purpose of this project is to predict an expected salary based on historical data, supervised learning using regression algorithms is used. The location, job title, and prevailing wage are used as predictors in trying to ascertain the target salary. According to USDL guidelines wages have to be equal or higher than the prevailing wage. It is fair to assume that a user of the developed model will not know the prevailing wage in advance, so to train the model, after dividing the data into training and validation sets, the training data is further divided in two. One set is used to predict prevailing wages as a function of location, job title and work category, and the second set is used to predict the final target salary as a function of the predicted prevailing wage and the other employment information. Each of these subsets is further divided in two in order to have training and testing sets to develop the models.

Benchmark Model

As stated above, the solution for the proposed project has two important steps: to predict prevailing wage and estimate employer wage. For the first part a standard linear regression is used as the benchmark model. The target being salaries, it is important to estimate in average how far the predictions are from the targets. This average is the basis to evaluate how accurate the developed models are compared to the benchmark model.

For the second part of the problem (predicting employer wage) the benchmark model is a Stochastic Gradient Descent (SGD) regressor [4] using the prevailing wage from the data set, while the models to be evaluated use the predicted prevailing wage from the model obtained in the first part or will not consider prevailing wage at all. Similarly in this step the metric of interest is how far on average the predictions are from the targets.

Evaluation Metrics

The purpose of this project is to develop a regression model using supervised learning. Given that the targets are continuous variables, the most appropriate metrics for evaluation correspond to those that support this kind of variable. It is important to know the difference of each prediction from the corresponding target value. In other words, the metric of interest is the absolute value of the difference between the target and the prediction. Finally, to obtain a general view of the model performance the most straightforward solution is to then calculate the average of those absolute values.

The metric of interest corresponding to the above description is the mean absolute error, in this case the mean absolute error between the targets and the predictions. The basis to judge performance is then the mean absolute error of the benchmark model.

The mathematical expression for the mean absolute error (MAE) is:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Where n is the number of data points, y_i is the target value of point i and $\hat{y}_i$ its prediction by the model.

Project Design

In the first part of predicting prevailing wage, all columns are categorical except for 'prevailing_wage' and 'employer_wage', the former being the target variable and the latter ignored in this first part. Given that the categorical variables correspond to text and some of them are sentences, the most direct approach is to concatenate per row the values of these columns into a single string by keeping spaces between words. Let's call this new column the 'merged' column. The 'merged' variable is then encoded using a TF-IDF vectorizer that allows its expression in a matrix form per data point so it can be used for a numerical linear regression.

The first step is to perform a linear regression for the benchmark model. In this step it is also determined if the most appropriate vectorizer is a simple one or one corresponding to an n-gram using a cross-validation grid-search. Since the targets spread in a large range from 20,000 to more than 300,000, their natural logarithm is taken to perform the regression. Using the natural logarithm considerably reduces the range of the targets and allows for an easy transformation to recover the predicted values.

After developing the benchmark model, a Huber regressor [5] which is less sensitive to outliers is considered by using the same input as the benchmark model. As an additional model a random forest regressor is used. For this model the input is no longer the 'merged' column but instead the original columns. Each of the columns is encoded by counting the number of categories in a column and assigning to each category an integer number from 0 to the number of categories minus one.

Given that assigning values may be interpreted by the model as giving more importance to higher values, the most common categories are assigned the highest numbers.

The mean absolute error of the Huber and random forest regressors, as well as the average combination of both models, are finally compared to the benchmark model to choose the one with the smaller error.

For the second part, the initial model is used to predict prevailing wages. These prevailing wages combined with the columns 'state' and 'postal_code' are used as inputs for the final model. The target salaries are employer wage. As in the first part, all the values are expressed in natural logarithm form. In this part, the benchmark model is an SGD regressor. For the benchmark model the prevailing wage from the data set is used. However, as mentioned before, it is fair to assume that a user of the final model will not know the prevailing wage, so for the other models the predicted prevailing wages from the initial model are used. Since the amount of data and categories are too big to handle by an SGD regressor, the training is done in batches using the OneHotEncoder method from the dummyPy package.

Similar to the first part, in addition to the SGD regressor a Huber and random forest regressors are developed and compared. The final model, as in the first part, is chosen by comparing the mean absolute error between models and choosing the one with the smallest error.

References

[1] *https://www.dol.gov/whd/immigration/h1b.htm*

[2] Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani. Chapter 3, *An Introduction to Statistical Learning*. 8^{th} Printing. Springer, 2017. 59-104. Print.

[3] Section 8.8.2. *Ibid.*, 319-323.

[4] http://scikit-learn.org/stable/modules/sgd.html#regression

[5] http://scikit-learn.org/stable/modules/linear_model.html#huber-regression.

[6] Yashu Seth. A BLOG ON DATA SCIENCE, MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE. *How to One Hot Encode Categorical Variables of a Large Dataset in Python?*. 2017.