CS6140 Final Project Proposal

CS6140 - 2023 Fall
Hyun Seung Lim

Goal of Project:

This project is designed to perform a regression analysis to accurately predict the annual salaries of individuals employed in data science-related jobs. The primary objective is to compare the effectiveness of two distinct preprocessing pipelines—a 'basic' pipeline and a 'complex' pipeline—in achieving accurate salary predictions.

Project Motivation: The underlying motivation for this project is to test the principle of "Occam's Razor" in the context of a regression task. Specifically, the project aims to determine whether a simpler preprocessing approach is more effective or if a more elaborate preprocessing strategy, characterized by the addition of synthetic features and the use of ordinal scales for certain categorical variables, yields better results.

Dataset: https://www.kaggle.com/datasets/arnabchaki/data-science-salaries-2023/data

- Target variable:
    - salary_in_usd: A person's salary in USD.
- Feature variables:
    - work_year: The year the salary was paid.
    - experience_level: The experience level in the job during the year
    - employment_type: The type of employment for the role
    - job_title: The role worked in during the year.
    - salary: The total gross salary amount paid.
    - salary_currency: The currency of the salary paid as an ISO 4217 currency code.
    - employee_residence: Employee's primary country of residence in during the work year as an ISO 3166 country code.
    - remote_ratio: The overall amount of work done remotely
    - company_location: The country of the employer's main office or contracting branch
    - company_size: The median number of people that worked for the company during the year

Evaluation methodology:

- For each of the two pre-processing pipeline ('basic' or 'complex'), I will evaluate the pre-processed training data using K-fold cross validation on the following four models: random forest regressor, linear regressor, linear ridge regressor, and linear lasso regressor, using sklearn's provided interface for these models.

- The evaluation metrics to use are:
    - R2: Captures how well the model captures total variance in the data
    - MSE: Captures the average squared difference
    - MAE: Captures the raw difference between y_true and y_pred.

- Then first I will select the pre-processing pipeline which has better mean R2, MSE, and MAE scores.
- Then I will select the model between linear, ridge, and lasso, which performed best in the selected preprocessing pipeline. Random forest regressor is a baseline model to compare the performance of the other three.

Model Selection:

- Given the selected model from the evaluation, I will implement the model from scratch.
- Optimization algorithm will be gradient descent algorithm, and loss function will be MSE.

4) Final evaluation and hyper-parameter tuning

- I will then apply K-fold cross validation of the selected model across different values of hyper-parameters.

- The chosen hyper parameters will then define my model, and the test set will be evaluated on this model to give the final performance metric.

Preprocessing:

'Basic' Pipeline:

1) Drop missing values
2) Standardize numerical features to increase convergence of gradient descent, and to increase comparability between features regardless of the choice of units.
3) Data Imputation: If missing values in features, use decision rules such as using mean or median for continuous, or most frequent for categorical. Determine this from EDA.

4) One hot encoding: Use one hot encoding to convert categorical features into numeric features.

'Complex' Pipeline: Apply the steps in 'basic' pipeline and then further add the following:

1) Convert categories into ordinal scale: If categorical variables have ordinal property, then use integers to represent categories.
2) Add synthetic features
3) Apply rare encoding: For each category in a categorical column, encode an uncommon category with the string "rare".

Workload distribution:
As this project is conducted by 1 person, myself, there is no distribution of work.

Related work:

- https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3526707
- http://ijasret.com/VolumeArticles/FullTextPDF/842_47._SALARY_PREDICTION_USING_MACHINE_LEARNING.pdf