1. Data cleaning is completed through 6 steps.

(1) Drop all “Other” columns: there are choices “Other” in multiple choice questions. However, Other contains no information and can be deleted.

(2) Drop useless features: useless features are 'Time from Start to Finish', 'Q24', 'Q24_buckets'. Since the response time is irrelevant in the survey, thus it can be removed. Also, Q24 and Q24_buckets are already encoded and represented by Q24_Encoded, so they can be removed as well.

(3) Encode categorical feature with order: for the categorical feature that has order, I firstly replaced all the NaNs by mode values and then encoded them from low to high. Categorical features with orders are 'Q4 Level of education', 'Q6 Coding experience', 'Q13 Number of times using TPU', 'Q15 Experience of machine learning', 'Q20 Size of company', 'Q21 Data science professionals in the company', 'Q25 Expense on machine learning'.

(4) For each multiple-choice question, I combined all parts of answer into one column. For example, if “Q7” has 12 answers, “Q7” is ranging from “Q7_Part_1” to “Q7_Part_12”. Then, Q7 is represented by sum of all parts. To illustrate, I firstly replaced all NaN values by 0 and non-NaN values by 1. By summing up all column of parts, except choice “None”, into one column, I get the number of choices one selected for one multiple choice question. Higher value infers more selections, which infers more ability one has.

(5) Encode all the remaining categorical data: I encode all the remaining features by assign numbers to each category. For example, man is zero while woman is one.

(6) Remove the row with questions: Since the first row contains only question, I removed it so that the data frame contains only numbers.

2. Exploratory data analysis and feature selection

Correlation plot is shown as a heatmap. The parameter cmap is “coolwarm”, thus cooler color means stronger negative correlation while warmer color means stronger positive correlation. As the correlation plot shown, “Q1 Age” and “Q3 Country” are the most related feature to a survey respondent’s yearly compensation with the same correlation value 0.4. The feature of importance is also shown as a bar chart with ascending order. Top four important features are “Q1 Age”, “Q3 Country”, “Q23 Important activities at work”, “Q22 Machine learning”.

Feature selection is important because it prepares an input dataset that is compatible with and best fits the machine learning algorithm and it also improves the performance of machine learning models. In this assignment, feature selection is required to select the features which are highly dependent on response variable so that irrelevant features can be removed. I used Chi-Square Test of Independence to determine whether there is any feature that is independent with the target variable salary. The null hypothesis for this test is that there is no relationship between the feature and target variable salary. The alternative hypothesis is that there is a relationship between them.

A chi-square test is computed between each feature and the target variable. By setting the confidence level as 0.05, if p-value is lower than 0.05, the feature is dependent with salary; otherwise, the feature is independent with salary and should be removed. As a result, “Q8 Program language”, “Q30 Big data products”, “Q38 Primary tool” are insignificant features and removed.

3. Model implementation

Ordinal logistic regression is where the dependent variable is ordinal. Multiple binary classification with orders is required before logistic regression is conducted. Since there are 15 classes in target variable salary, the binary classification is conducted by separating the classes from low to high. Let Y be an ordinal outcome with 15 categories. Then $P(Y \leq j)$ is the cumulative probability of Y less than or equal to a specific category $j = 0, \ldots, 14$. The probabilities for each class is calculated by the new cumulative probabilities minus last cumulative probabilities. For example, I computed the probabilities for class 0, class 0 + class 1, class 0 + class 1 + class 2, etc. And then, I get probabilities for class 0, class 1, class 2, etc. After the probability for each class is computed, the class with maximum probability for each observation is selected to finish the multi-class predictions. As a result, the accuracy scores across folds are pretty similar and the average accuracy score is 80.429% with the variance 0.034%.

I tested parameter C in list [0.001,0.01,0.1,1,10,100] and plot the bias variance trade-off on that. I found that as C gets larger, the bias gets lower while variance gets larger. In order to decrease bias, complexity should be increased. I also tested parameter solver in list ['newton-cg','lbfgs','liblinear','sag'] and plot the bias variance trade-off. The bias are the same for all solvers. Thus, to minimize variance, solver “liblinear” is selected.

Scaling/normalization is necessary in feature engineering. Since all the features have different scales, scaling/normalization can remove the skewness of data so that the weightage may not be highly dependent on the magnitude of the feature.

4. Model tuning

For the logistic regression function, the parameters are as following: 'penalty', 'dual', 'tol', 'C', 'fit_intercept', 'intercept_scaling', 'class_weight', 'random_state', 'solver', 'max_iter', 'multi_class', 'verbose', 'warm_start', 'n_jobs', 'l1_ratio'.

I select C and penalty for model tunning. C is the inverse of regularization strength. For small values of C, we increase the regularization strength which will create simple models which underfit the data. For big values of C, we low the power of regularization which implies the model is allowed to increase its complexity, and therefore, overfit the data. L1 penalty leads to sparser solutions while L2 penalty leads to more constrained solutions. Moreover, the Elastic-Net penalty sparsity is between that of L1 and L2.

Grid search is the process of performing hyperparameter tuning in order to determine the optimal values for a given model. In this case, I pass predefined values for hyperparameters "C" and

"penalty" to the GridSearchCV function. GridSearchCV tries all the combinations of the values passed in the dictionary and evaluates the model for each combination using the Cross-Validation method. As a result, the optimal parameters for "C" and "penalty" are 0.01 and 'l2' respectively with the highest accuracy 80.429%.

5. Testing and Discussion

The optimal model got 80.25% accuracy and 2.181% variance on test set and 80.429% accuracy and 0.034% variance on training set. Thus, training set produce slightly higher accuracy than the test set.

To see whether the model is overfitting or underfitting, I used DecisionTreeClassifier and test different tree depths with the “max_depth” argument. A plot is created to show line plots of the model accuracy on the train and test sets with different tree depths.

The plot clearly shows that increasing the tree depth in the early stages results in a corresponding improvement in both train and test sets. This continues until a depth of around 5 levels, after which the model is shown to overfit the training dataset at the cost of worse performance on the holdout dataset.

Overfitting is the case where model performance on the training dataset is improved at the cost of worse performance on data not seen during training. Therefore, the model is overfitting. Overfitting normally occurs when there are too many parameters because training a model with so many parameters that it can fit nearly any dataset. Therefore, we can reduce the number of trainable parameters so as to reduce the complexity of model. Also, we can use smaller C parameter so as to increase the regularization strength which will create simple models which underfit the data.

According to the distribution of true target variable values and their predictions on both the training set and test set, I found that the predictions based on training set produces more results on class 0, class 10, and class 12, while the predictions based on test set produces less results on all class 0 compared to true value. It is obvious that the true value is more spread out than the predictions, because the predictions are more likely be in class 0, class 10, and class 12. In general, the predictions on salary are either very low or very high.