Glossary

Term	Definition
Categorical data	Categorical data is a type of data that is used to group information with similar characteristics
Numerical data	Numerical data is a type of data that expresses information in the form of numbers.
Ordinal data	Ordinal data is a kind of qualitative data that groups variables into ordered categories. The categories have a natural order or rank based on some hierarchal scale, like from high to low.
One-hot encoding	One-hot encoding in machine learning is the conversion of categorical information into a format that may be fed into machine learning algorithms to improve prediction accuracy.
Cross validation	Cross-validation, sometimes called rotation estimation or out-of-sample testing, is any of various similar model validation techniques for assessing how the results of a statistical analysis will generalize to an independent data set.
Ridge Regression	Ridge regression is a method of estimating the coefficients of multiple-regression models in scenarios where the independent variables are highly correlated.
R-squared value	R-squared is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable.
Mean Square Error (MSE):	Mean square error is calculated by taking the average, specifically the mean, of errors squared from data as it relates to a function. A larger MSE indicates that the data points are dispersed widely around its central moment (mean), whereas a smaller MSE suggests the opposite.
Root Mean Square Error (RMSE):	Root mean square error or root mean square deviation is one of the most commonly used measures for evaluating the quality of predictions. It shows how far predictions fall from measured true values using Euclidean distance.
Mean Absolute Error (MAE):	Mean Absolute Error (MAE) is a regressive loss measure looking at the absolute value difference between a model's predictions and ground truth, averaged out across the dataset.
Mean Absolute Percentage Error (MAPE):	Percentage equivalent of mean absolute error (MAE). Mean absolute percentage error measures the average magnitude of error produced by a model, or how far off predictions are on average.
Decision Tree Leaf Size:	The size of a leaf is the number of Examples in its subset. The tree is generated in such a way that every leaf has at least the minimal leaf size number of Examples.
kFoldLoss:	kfoldLoss uses the mean of the observed response values in the training-fold data as the predicted response value for observations with missing predictor values.
Decision Tree Pruning:	In machine learning and data mining, pruning is a technique associated with decision trees. Pruning reduces the size of decision trees by removing parts of the tree that do not provide power to classify instances.

Overfitting:	Overfitting is an undesirable machine learning behavior that occurs when the machine learning model gives accurate predictions for training data but not for new data.
Bias:	A high bias model typically includes more assumptions about the target function or end result. A low bias model incorporates fewer assumptions about the target function. A linear algorithm often has high bias, which makes them learn fast.
Variance:	A low variance indicates that the data points tend to be very close to the mean, and to each other. A high variance indicates that the data points are very spread out from the mean, and from one another.
Correlation Matrix:	A correlation matrix is a statistical technique used to evaluate the relationship between two variables in a data set. The matrix is a table in which every cell contains a correlation coefficient, where 1 is considered a strong relationship between variables, 0 a neutral relationship and -1 a not strong relationship.
Gradient boosting:	The gradient Boosting algorithm is based on ensemble learning where an ML model makes predictions based on n number of distinct models. From these models, this learning approach finds less biased and varied data points.

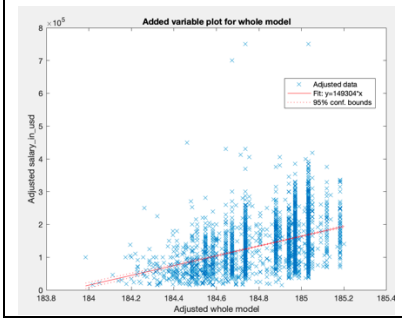
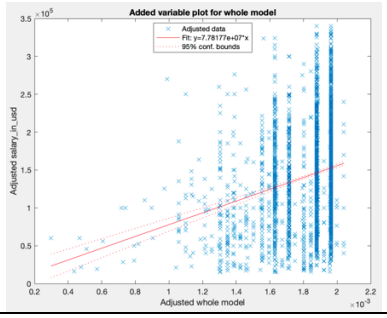
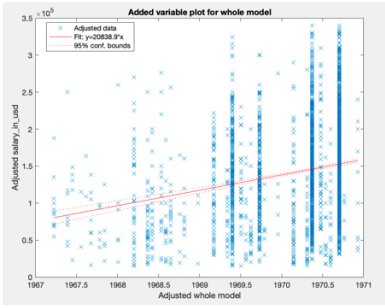
Intermediate Results

Originally, the dataset contained 10 features and 1 target variable. When dependencies were assessed through Exploratory Data Analysis (EDA), Salary (in USD) resulted in no dependency over variables like Job_Title, Salary (in home currency) and hence, these variables were dropped from any further analysis.

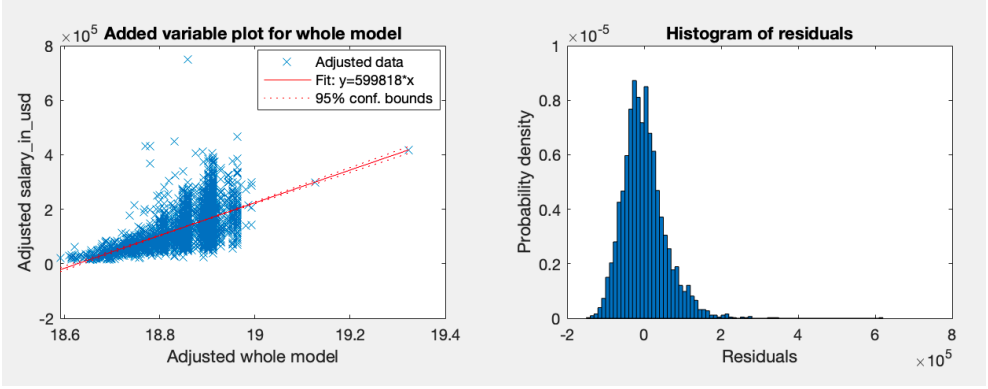
Since there were as many as 6 categorical features (and only 2 numerical) which were taken into account, there was a challenge of handling the format specially for Linear Regression model which doesn't accept any categorical values to train. Hence, for the sake of Linear Regression, one-hot encoding needed to be implemented which resulted in 194 columns in total. This made the training heavier and time taken was reflective of the extra processing.
The encoded data was tried to be fed to Decision Tree as well but the model didn't respond well to the same and then, I had to use the categorical data as it is for training Decision tree model. Hence, now we were dealing with two sets of data. One with the binary encoded values and one without. For this reason, it became difficult for me to run the code in one single file. Separating the code files and comparing model performances explicitly proved to be a better option.

There were many techniques I would've liked to implement like correlation matrix in the beginning instead of box plotting to assess the relationships, but the datatypes were a limitation there.

I tried fitting my LM model with fewer features and the results are as below:
With following features respectively:

Employment type, Experience level, Company size, Work year, Remote ratio	Employment type, Company size, Work year, Remote ratio	Company size, Work year, Remote ratio
		

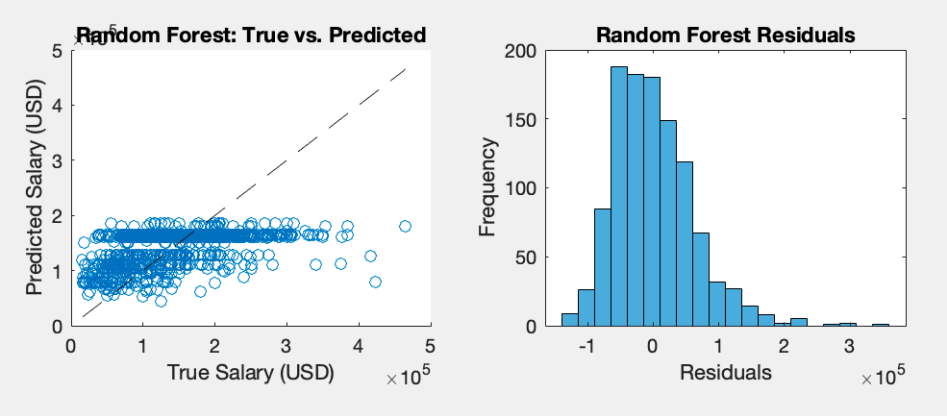
Plotting base LM model with all of the selected features looked like below:



Implementation Details & Choices

At one point, when I realised that the data is not linear enough to be predicted well through my chosen models, I tried running the same through Random Forest model to see and compare the results. The results obtained through Random Forest are as below (eve after optimising and hyper-tuning a bit):

Mean Absolute Error (MAE)	Mean Squared Error (MSE)	Root Mean Squared Error (RMSE)	Time taken
47301	3.7497e+09	61235	1.7729e+05



These results were clearly not impressive enough to be included in my analysis. Even implementing bagging or changing number of trees didn't make much difference to the results and hence, the approach was dropped. It made better sense to compare two very different kind of models like Linear Regression and Decision Tree where both models have their own different rules to implement to get to the result.