Gisual Senior Machine Learning Engineer – Code challenge Writeup (Tejaswi Rao)

Approach:

For this analysis, I was tasked with creating a model to predict the probability of events based on weather conditions, locations, and timestamps. The dataset provided contained information such as temperature, dew point, wind speed, wind direction, precipitation, location, time, and event occurrence. The goal was to build a predictive model and evaluate its performance using appropriate metrics.

Data Preprocessing:

I started by performing exploratory data analysis to gain insight into the features and their distributions. I tested the missing values and since there were none, I didn't have to take any action in this regard. Another important observation that was made was the occurrence of duplicates. So, I removed the duplicates and kept only those that were a unique row. This was necessary because, When I tried to take all the rows into account, all the metrics shows maximum result, mainly because the same data will be present in training and testing data eventually making it as two training datasets which is dangerous. To avoid that, I had to trim down the duplicates. Other notable change is that of time, since I had the time-of-day column, there was no need for the time, so I deleted those and just kept the date-month-year. Categorical variables were encoded using one-hot encoding or label encoding. I also addressed the issue of class imbalance by applying the SMOTE (Synthetic Minority Over-sampling Technique) algorithm to balance the dataset. I split the dataset into training and testing sets to ensure unbiased evaluation of the model's performance.

Model Selection and Training:

I experimented with various machine learning algorithms such as Random Forest, Logistic Regression, and Gradient Boosting. After evaluating their performance using appropriate metrics, I selected the Random Forest model as it showed promising results. I trained the Random Forest model on the training set and fine-tuned its hyperparameters using techniques like grid search or randomized search to optimize its performance. I also tried other methods like Boosting classifier, Neural Networks, regression and Support vector machines.

HyperParameter Tuning:

To optimize the model's hyperparameters, I used RandomizedSearchCV, which performs a randomized search over a hyperparameter space, considering different combinations of hyperparameters. This technique helps find the best hyperparameters for the model, improving its performance.

Model Evaluation:

I evaluated the performance of the trained Random Forest model using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. These metrics provided insights into how well the model predicted event occurrences based on the weather conditions. Additionally, I visualized the model's predictions and compared them with the actual event occurrences to gain further insights into the

model's performance. Since I have used SMOTE to make it an balanced dataset, Accuracy can also be a great metric to consider. As far as the results are concerned,

```
Accuracy: 0.8461538461538461
Precision: 1.0
Recall: 0.6666666666666666
F1-score: 0.8
ROC-AUC: 0.8333333333333333
```

Insights:

Through the analysis, I gained the following insights:

Temperature, wind speed and precipitation are important features in predicting event occurrences. Higher values of precipitation and wind speed are associated with a higher probability of events.

Location and time of the day does not play a huge role in event occurrences.

The Random Forest model demonstrated strong predictive performance, achieving high accuracy, precision, recall, F1-score, and ROC-AUC values. This indicates that the model effectively captures the relationship between weather conditions and event occurrences.

The calibration curve and precision-recall curve provided insights into the model's calibration, reliability, and trade-off between precision and recall. These curves helped us understand how the model's predicted probabilities align with the actual probabilities and how well it performs at different decision thresholds.

Overall, the Random Forest model, along with the selected features, proved to be effective in predicting event occurrences based on weather conditions. The insights gained from the analysis can be used to understand the factors contributing to event probabilities and make informed decisions or take appropriate actions based on the weather conditions.

Improvements and Future Work:

In future iterations, I can consider additional feature engineering techniques to further enhance the model's performance. This may include feature interactions, or creating new features based on domain knowledge.

Additionally, collecting more data, especially for event occurrences, can help improve the model's performance and reduce any class imbalance issues. Conducting further analysis on the misclassified samples or exploring the model's interpretability can provide deeper insights into the relationships between weather conditions and event occurrences.

Bonus- In the code, I have mentioned about the events taking place indoor and outdoor mainly because of the weather. To strengthen my argument, from the latitude and longitude, I created a new column named location using Geocoding API through google cloud console which pinpoints to the exact street address from which you can get the name of the building or name of the location where the event is taking place. To conclude, by the limited data I have seen, most of the locations were indoors and as the temperature, rain and wind speed increases, the event takes place indoors.