Challenge Activities

1. What steps would you take to solve this problem? Please describe as completely and clearly as possible all the steps that you see as essential for solving the problem.

Data Cleaning:

- Load the datasets and inspect for any discrepancies or inconsistencies
- Replace all “na” strings with NaN values
- Handle missing values by replacing them with each column median
- Convert every column into a numerical type

Exploratory Data Analysis (EDA):

- Investigate data distribution
- Analyze the class distribution to understand any class imbalance issues

Feature Engineering:

- Handle data distribution problems by applying either log or cubic transformation to each column, selecting the best method for the column
- Create a correlation matrix and check which features have a moderate to high correlation with the target
- Apply PCA to find the inflection point of which the number of components retain 99% of data variance
- Select features based on the number of components found by the PCA using ANOVA f-classif

Model Training:

- Split the previous_years data into training and testing sets to evaluate the models performance
- Train multiple predictive models, such as Logistic Regression, Random Forest, and Gradient Boosting, to find the best-performing model
- Tune the hyperparameters of the models using techniques like Random Search to optimize performance
- Evaluate model performance using metrics such as recall, precision, F1-score, and ROC-AUC score, with a focus on recall to minimize false negatives
- Choose best model based on the recall metric, changing the threshold of the model if needed

Production Deployment:

- Select the best-performing model based on recall and other relevant metrics
- Validate the model on the air_system_present_year.csv dataset to ensure it generalizes well to new data
- Deploy the model into production, ensuring that it is integrated into the existing maintenance workflow
- Monitor the model's performance over time, tracking key metrics and retraining the model as needed to maintain accuracy with the help of MLflow

2. Which technical data science metric would you use to solve this challenge? Ex: absolute error, rmse, etc.

Recall: Focused on minimizing false negatives to avoid costly corrective maintenance.

Other than that, confusion matrix to check the actual number of false positives and false negatives of the models, along with precision, f1-score and ROC-AUC score, to see if the model can discriminate between the target classes.

3. Which business metric would you use to solve the challenge?

Cost reduction, since the primary goal of the business is to reduce the costs with maintenance. Another one would be ROI, how much of the investment is returned when the model is in production.

4. How do technical metrics relate to the business metrics?

Recall and Cost Reduction: Higher recall means fewer defective trucks are missed, leading to fewer expensive corrective maintenances and thus reducing overall costs.

Precision and Operational Efficiency: Higher precision ensures that non-defective trucks are not unnecessarily sent for maintenance, improving operational efficiency.

F1 Score and Overall Performance: A high F1 score ensures a good balance between precision and recall, indicating a reliable model that helps in cost reduction and efficient maintenance planning.

5. What types of analyzes would you like to perform on the customer database?

Descriptive analysis (mean, median, main characteristics of the data) to help identify the data distribution and if needs to be fixed.

Correlation analysis to potentially help reduce the dimension of the dataset, or to help select the best features.

6. What techniques would you use to reduce the dimensionality of the problem?

PCA to find the number of components needed to retain 99% of data variance and correlation analysis to identify features with high correlation, removing redundancy.

7. What techniques would you use to select variables for your predictive model?

ANOVA with f-clasif to select the features with the highest relation to the target. The number of features selected is based on the number of components found when applying the PCA

8. What predictive models would you use or test for this problem? Please indicate at least 3.

Logistic Regression – It's the simplest, and yet effective, of classification models. It can handle high-dimensional data when regularized and can indicate the probability for the result to be in each class

Random Forest - An ensemble method that is robust to overfitting and can handle high-dimensional data well. It provides feature importance scores, which help in interpretability and feature selection

Gradient Boosting - Another ensemble method that builds trees sequentially to correct the errors of the previous trees. It often provides high accuracy and it doesn't need a class balance parameter

9. How would you rate which of the trained models is the best?

Recall - Using primarily recall, as false negatives have an impact of $500 in the cost of maintenance

Precision - Used to ensure that the model is not sending too many non-defective trucks for maintenance, thereby avoiding unnecessary costs.

ROC-AUC Score - Used to assess the model's overall ability to discriminate between positive and negative classes.

10. How would you explain the result of your model? Is it possible to know which variables are most important?

The model can generalize data well, and only a tweak on its threshold was needed to improve its performance drastically. To know the most important variables, it's possible to use the function from scikit-learn permutation_importance(). It permutates the values of each column of the training data used, and checks the impact on the desired metric, that in this case is recall. So, if a feature appears with a permutation score of 0.12, it indicates that by permutating its values, the recall saw a decay of 0.12, making the model worse. So, this would be an important feature, since the model relies on the correct values of this feature to make correct predictions.

11. How would you assess the financial impact of the proposed model?

We need to use a confusion matrix to see the exact amount of false negatives and false positives to be able to calculate the cost of mistakes by the model. To calculate the overall cost of the model, just the cost of mistake to the cost of maintenance to trucks that were sent correctly to maintenance.

12. What techniques would you use to perform the hyperparameter optimization of the chosen model?

Random search, as it's faster than grid search and more times than often, produce the exact same results. So, by using random search, we gain time and computational resources.

13. What risks or precautions would you present to the customer before putting this model into production?

The model is trained based on historical data of the previous_years dataset. If 2 years from now, the main reasons as to why the truck has a failure starts to change a lot, the model won't know the difference and it will need to be trained again. So, continuous monitoring of model results is a must to not get stuck using an outdated model that gives the wrong answer the majority of times.

14. If your predictive model is approved, how would you put it into production?

Using fast-api seems to be the fastest way to put a model into production. Also, using MLflow, it would be able to change the status of the model, and make it way easier to monitor it.

15. If the model is in production, how would you monitor it?

Using MLflow, since it allows the user to see in a dashboard all informations about the model, such as its metrics, the parameters, set a stage for the model, so there’s no confusion as to what model is in testing and which one is in production.

16. If the model is in production, how would you know when to retrain it?

The first thing to look is at the metrics. If they are dropping significantly, it indicates that the model is not performing as well as it could. Another way is by detecting data drifts, when the statistical properties of the data that’s coming in are changing over time.

To handle these issues, it’s possible to set-up a periodic review schedule for the model, to review its metrics and performance overall, avoiding getting stuck with a very bad model.