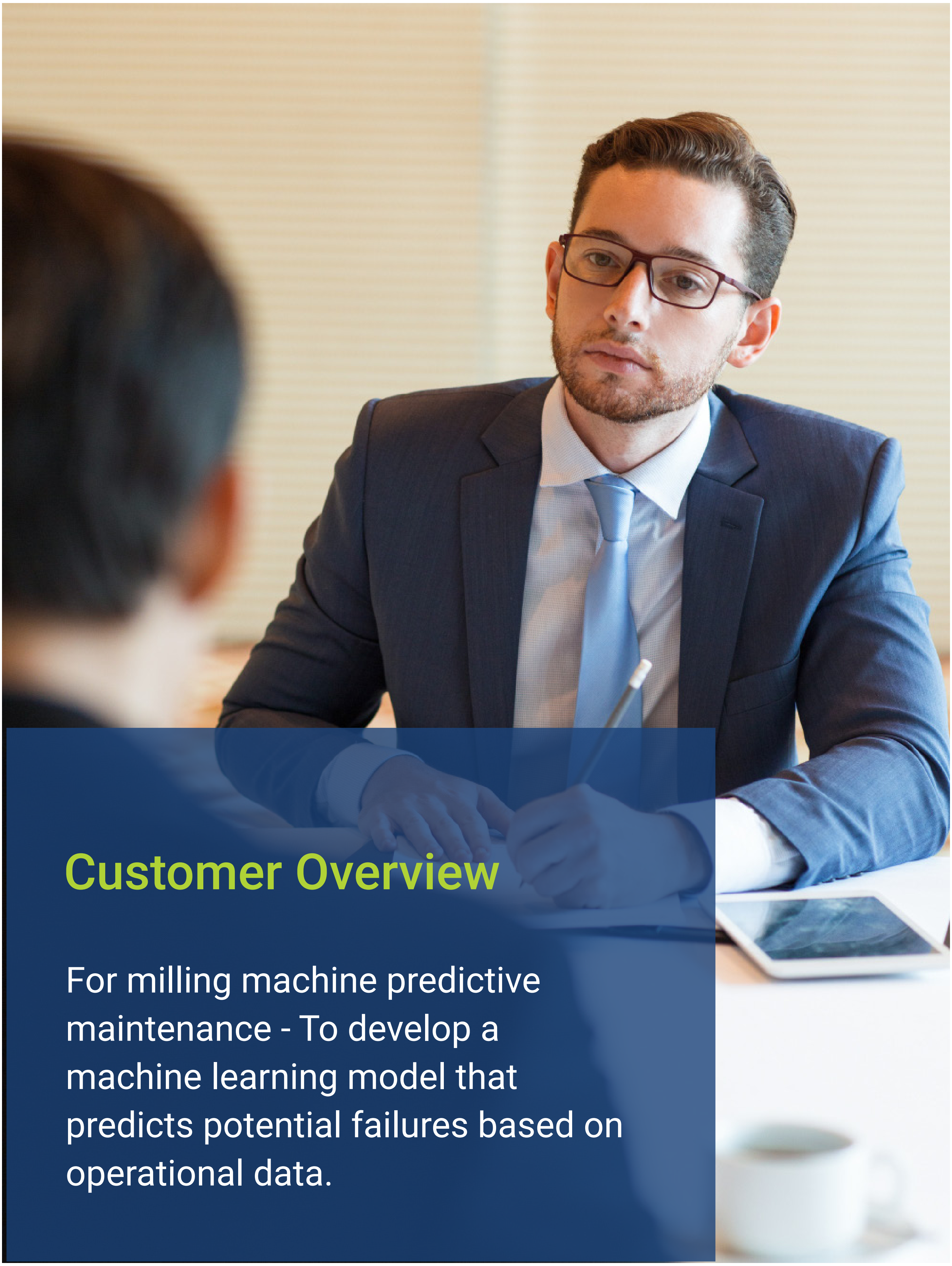
CloudAngles mlangles

mlangles Predictive AI

Milling Machine Predictive Maintenance

Usecase



Customer Overview

For milling machine predictive maintenance - To develop a machine learning model that predicts potential failures based on operational data.

Challenges

- Data Quality and Availability: Ensuring access to high-quality data including data on machine operations, environmental conditions, and historical failure records.
- Imbalanced Data: Failure data is often imbalanced, with fewer instances of failures compared to normal operations.
- Adaptation to Changing Conditions: Milling machines' operating conditions may change over time due to various factors such as aging equipment, varying workloads, or environmental changes.

About mlangles Predictive AI

mlangles is a comprehensive AI platform designed to manage the lifecycle of data and models, offering streamlined solutions for every stage of the process.

Through its MLOps component, mlangles provides a suite of tools to navigate efficiently through each phase of AI project development, encompassing data engineering, development, deployment, and monitoring. It facilitates continuous integration, continuous deployment,continuous training, and continuous monitoring (CI-CD-CT-CM), enabling enterprises to effectively manage their AI initiatives.

OBJECTIVE OF THE USE CASE

The objective is predictive maintenance for milling machines, aiming to develop a machine learning model predicting potential failures based on operational data. This enables proactive maintenance scheduling, minimizing downtime and costs, optimizing machine performance, and enhancing overall productivity and profitability.

working of the use case :

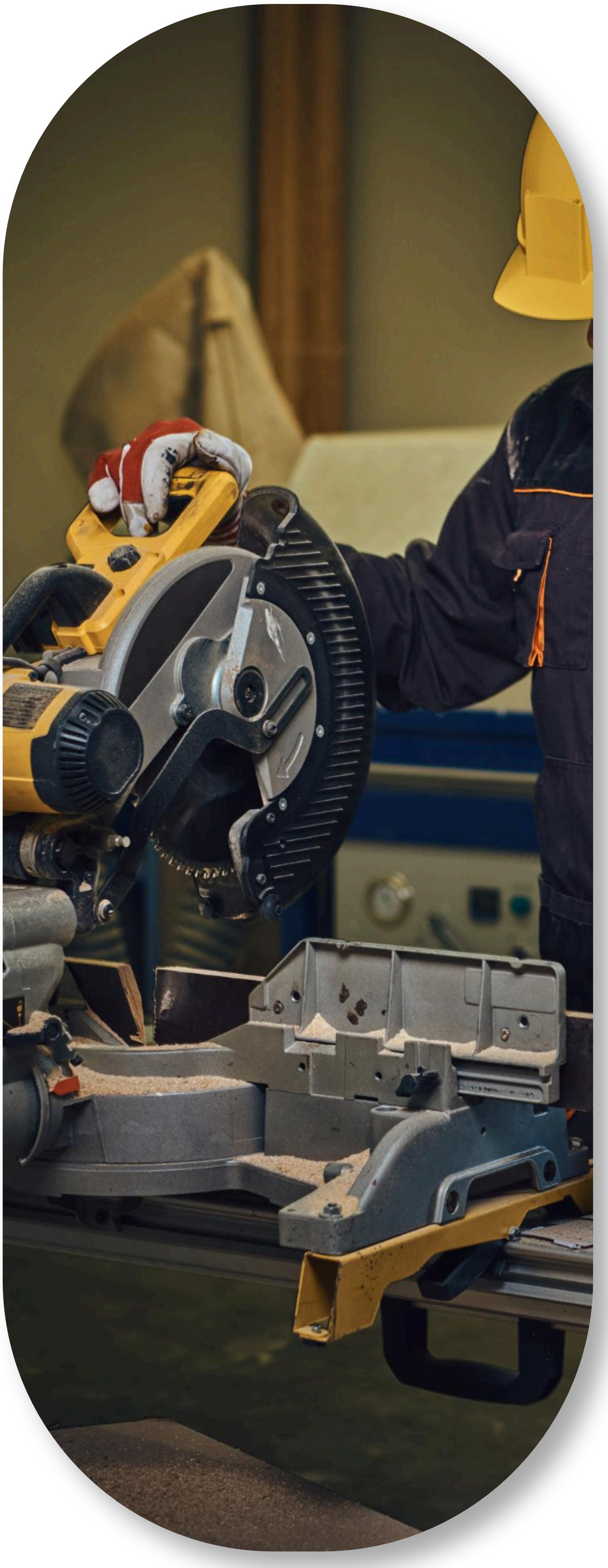
- The AI problem will be tackled through a phased approach, starting with the data engineering phase utilizing the pipeline module.
- The modelling process will follow, with the experiment tracking module aiding in the selection of suitable hyperparameters.
- Subsequently, the model will be trained and executed on the provided dataset.
- Predictions generated by the model will be presented using the serving module.
- Continuous monitoring will be implemented to maintain the accuracy and effectiveness of the model over time.

Description of the use case:

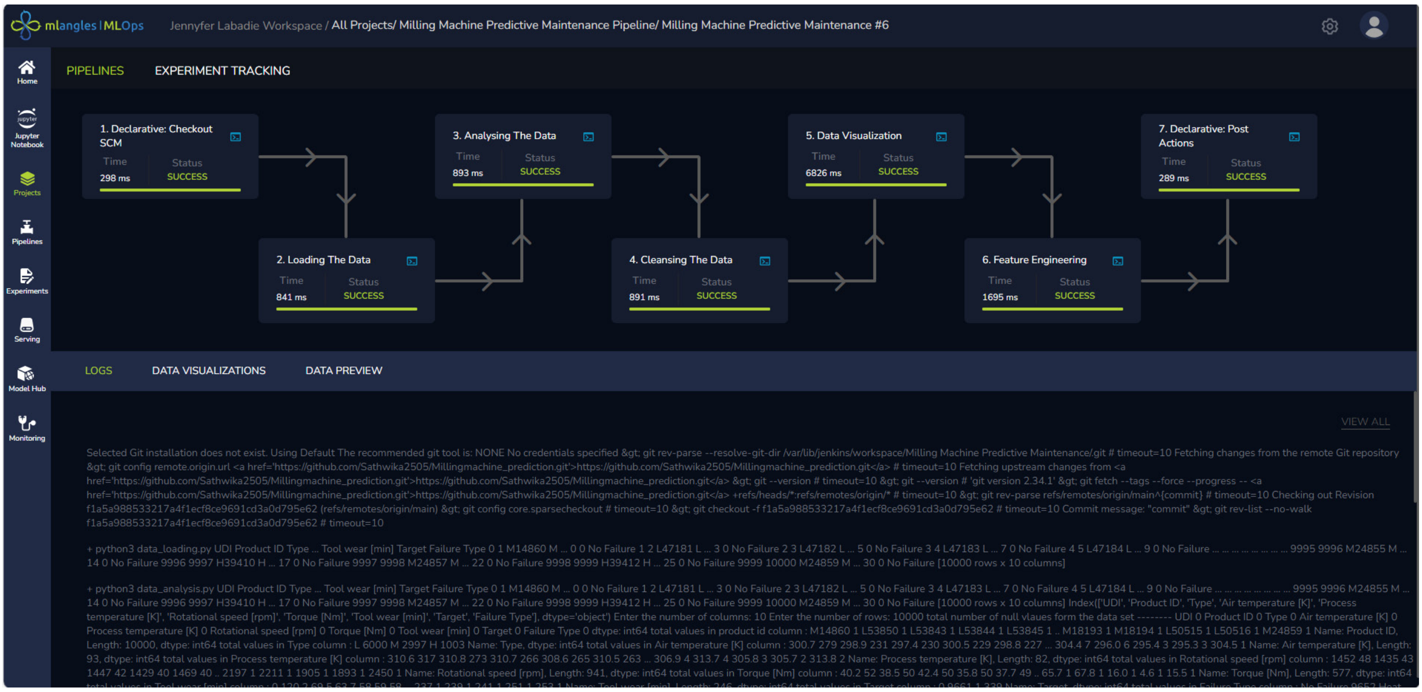
In manufacturing industries, milling machines play a critical role in shaping raw materials into finished products through precision cutting and drilling operations. However, like any mechanical system, milling machines are susceptible to wear and tear over time, leading to potential failures that can disrupt production schedules and incur significant costs. Predictive maintenance aims to address this challenge by leveraging data-driven techniques to anticipate and prevent machine failures before they occur.

The provided dataset offers a synthetic representation of real-world predictive maintenance scenarios encountered in the industry. It comprises 10,000 data points, each characterized by 14 features:

- UID (Unique Identifier): This unique identifier distinguishes each data point in the dataset.
- ProductID: Indicates the product quality variant (low, medium, or high) and a variant-specific serial number, reflecting different levels of manufacturing standards.
- Air Temperature [K]: Represents the temperature of the surrounding environment, which can influence the performance and thermal stability of the milling machine.
- Process Temperature [K]: Reflects the temperature within the milling machine during operation, influenced by both environmental conditions and internal processes.
- Rotational Speed [rpm]: Denotes the speed at which the milling machine's cutting tool rotates, impacting the efficiency and precision of material removal.
- Torque [Nm]: Indicates the amount of rotational force applied to the cutting tool, which affects the machining process's stability and effectiveness.
- Machine Failure: This binary label signifies whether the milling machine experienced a failure during a specific data point, serving as the target variable for predictive modelling.



Step 1: Data Engineering & Pipeline Creation



Data Extraction: Essential packages and libraries have been installed.

Data Analysis: Essential packages and libraries have been installed.

- Identify Missing Values: Check for crucial data gaps to maintain accuracy.
- Detect Duplicate Entries: Remove redundant data for consistency.
- Assess Feature Types: Differentiate between categorical and numerical attributes for effective analysis.
- Determine Dataset Size: Understand the dataset's scale to guide further analysis.

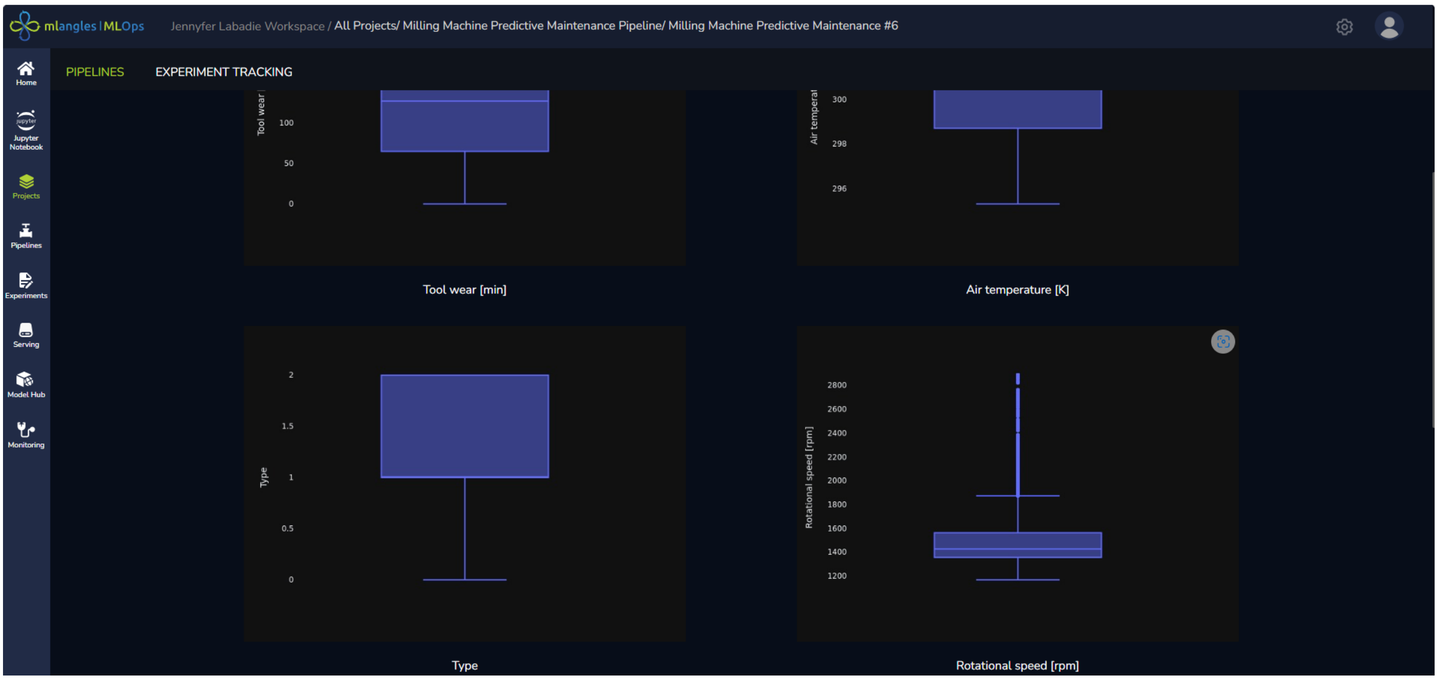
Data Visualization: This step involves representing data graphically to reveal patterns and insights. Here are brief explanations of the visualizations used in this use case.

- Box Plots: Box plots display the distribution of numerical data through quartiles, highlighting the median, interquartile range, and outliers. They help identify central tendency, spread, and skewness in the data.
- Heatmaps: Heatmaps visualize data in a matrix format using colors to represent values. They are useful for displaying correlations between variables or patterns in large datasets, making it easier to identify relationships and trends.

The box plot below illustrates the presence of outliers within the dataset. It reveals that the “Type" variable exhibits no outliers, as all data points fall within the typical range, whereas the "Rotational Speed" feature displays outliers, indicated by individual data points that significantly deviate from most values. This visualization effectively identifies potential anomalies in the data, aiding in further analysis and decision-making processes.

Data Versioning:

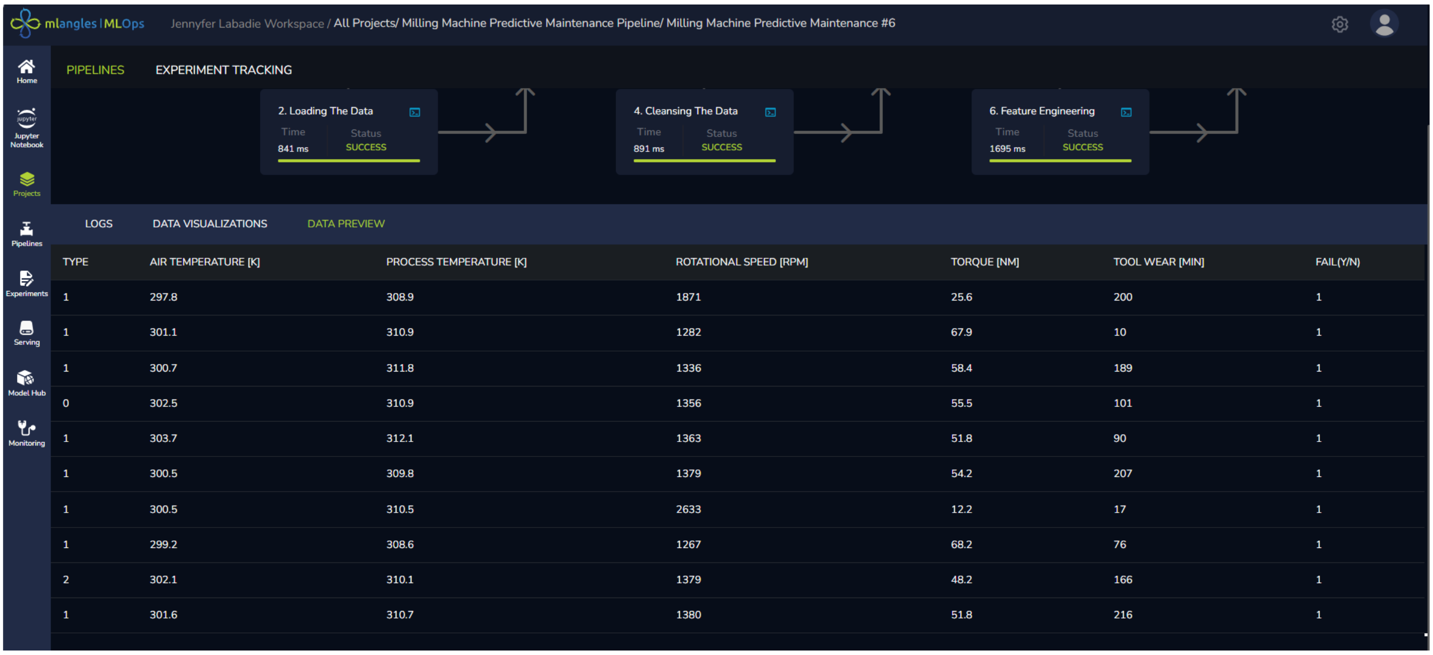
- Various processed data versions can be generated through different transformations applied to the same raw dataset, such as deleting columns or applying various transformations on specific columns.
- Throughout the data pipeline, diverse transformations can be executed at each iteration. Consequently, the resulting data at the pipeline's end is systematically versioned.
- Given that each version of the final data is distinct, models trained on these different versions will exhibit varying behaviors.



From the below, The heatmap value of air temperature and process temperature with a correlation coefficient of 0.8581 indicates a strong positive relationship between these temperatures and milling machine failure. This suggests that as air temperature and process temperature increase, the likelihood of machine failure also increases significantly. Understanding this correlation allows for proactive measures to control or mitigate temperature fluctuations, thus reducing the risk of milling machine failures and optimizing operational efficiency.

Feature Engineering: This step encompasses various tasks aimed at enhancing the quality and relevance of features used in machine learning models. This process involves balancing the dataset to address class imbalances and removing outliers from features, such as the "Type" feature where anomalies were detected during the visualization step. By balancing the dataset, we ensure that the model is not biased towards the majority class and can effectively learn from all classes. Additionally, removing outliers improves the robustness and generalization capability of the model by reducing the influence of anomalous data points that may distort the learning process. Overall, feature engineering enhances the quality of input features, leading to more accurate and reliable machine learning models.case.

Below is the image of the cleansed dataset after the data engineering steps.

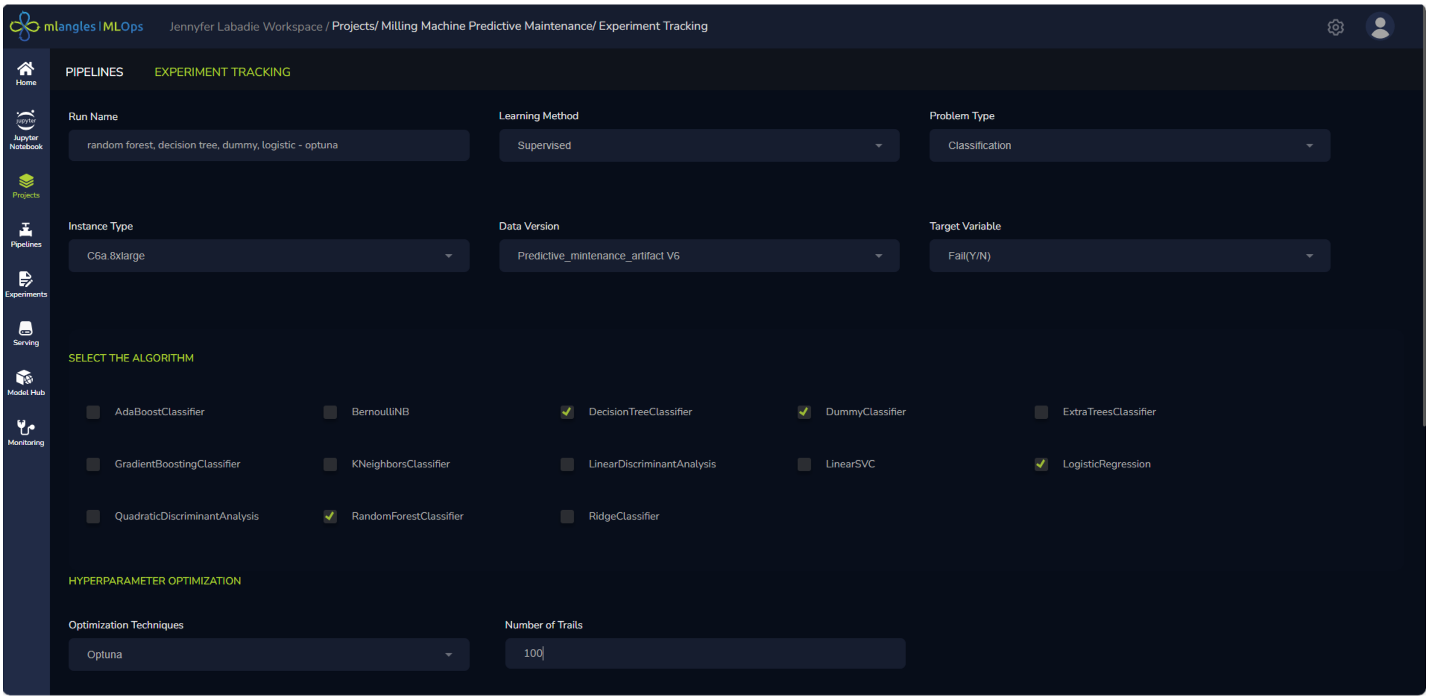


TYPE	AIR TEMPERATURE [K]	PROCESS TEMPERATURE [K]	ROTATIONAL SPEED [RPM]	TORQUE [NM]	TOOL WEAR [MIN]	FAIL(Y/N)
1	297.8	308.9	1871	25.6	200	1
1	301.1	310.9	1282	67.9	10	1
1	300.7	311.8	1336	58.4	189	1
0	302.5	310.9	1356	55.5	101	1
1	303.7	312.1	1363	51.8	90	1
1	300.5	309.8	1379	54.2	207	1
1	300.5	310.5	2633	12.2	17	1
1	299.2	308.6	1267	68.2	76	1
2	302.1	310.1	1379	48.2	166	1
1	301.6	310.7	1380	51.8	216	1

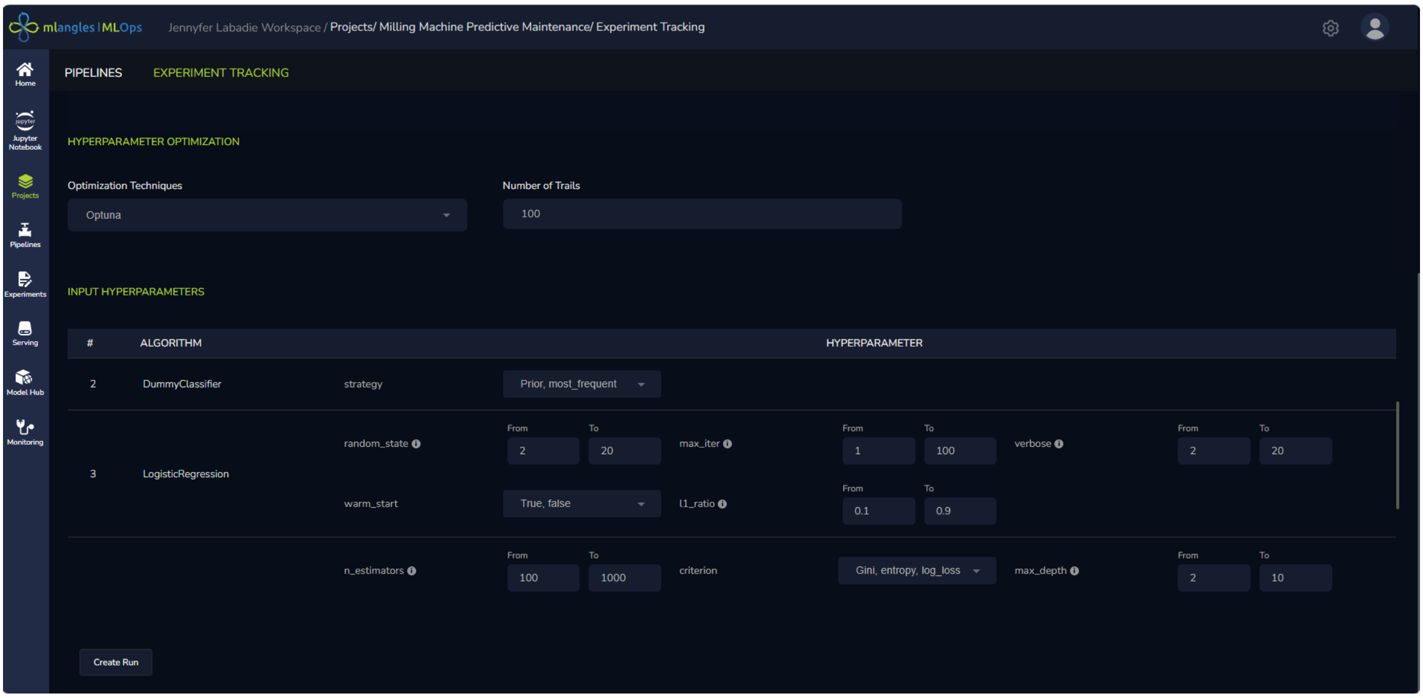
Step 2: Experiment Tracking - Modelling with Hyper-Parameter Optimizationation

After preparing the cleansed data, the next step involves training the model using this refined dataset. Given that this is a classification problem, various models are suitable for the task. Common choices include the random forest classifier, decision tree classifier, dummy classifier, and logistic regression.

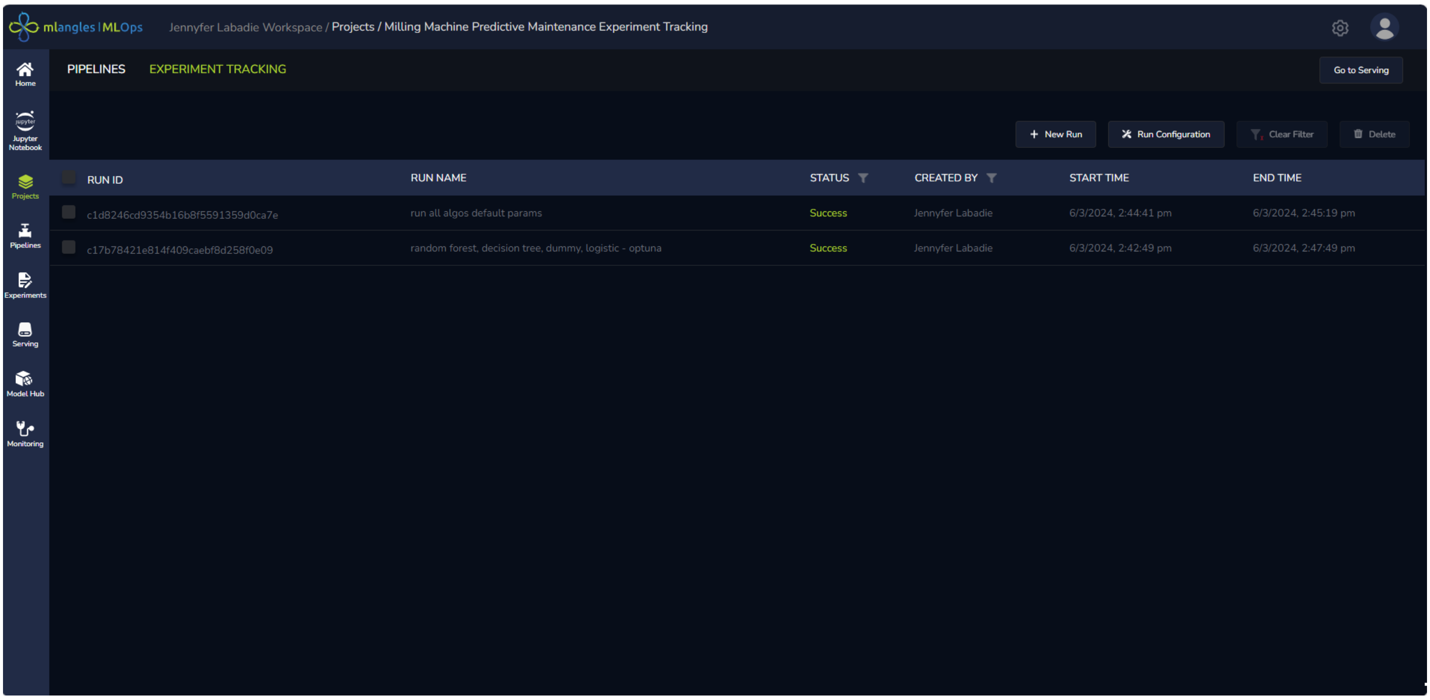
- Random Forest Classifier: This ensemble learning method constructs multiple decision trees during training and outputs the mode of the classes (classification) or the mean prediction (regression) of the individual trees.
- Decision Tree Classifier: : Decision trees split the data into subsets based on features' values and make decisions at each node. They're simple yet powerful for classification tasks..
- Dummy Classifier: This classifier serves as a baseline model by making predictions using simple rules. It's useful for comparing the performance of more sophisticated models.
- Logistic Regression: Despite its name, logistic regression is a linear model for binary classification. It calculates the probability of belonging to a class based on input features.



Additionally, to enhance model performance, a hyperparameter optimization technique called Optuna is employed. Optuna automates the process of tuning hyperparameters, such as learning rate or tree depth, to find the optimal configuration that maximizes model performance. This approach ensures that the model is fine-tuned to achieve the best possible results on the given dataset, improving its accuracy and predictive power.



Once the run is created, various details can be extracted, including hyperparameter visualizations of the best algorithm, parameters utilized during training, and metrics and artifacts. These insights provide valuable information about the model's performance and behavior, aiding in understanding its effectiveness and potential areas for improvement.



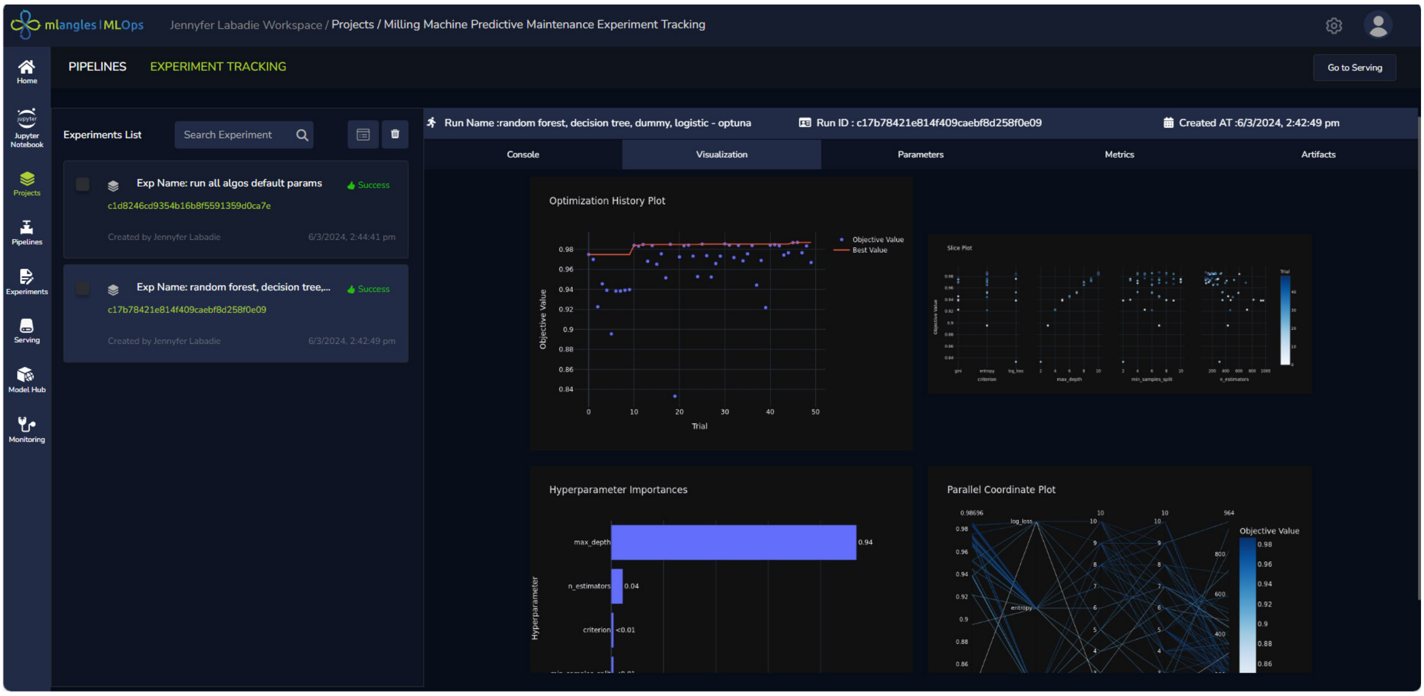
Hyperparameter visualizations offer a graphical representation of how different parameter settings impact model performance, facilitating the selection of optimal configurations.

Optimization History Plot: This plot displays the optimization history of the hyperparameter search, showing how the objective function (e.g., accuracy or loss) evolves over optimization iterations. It helps in understanding the convergence behavior and the effectiveness of the optimization algorithm.

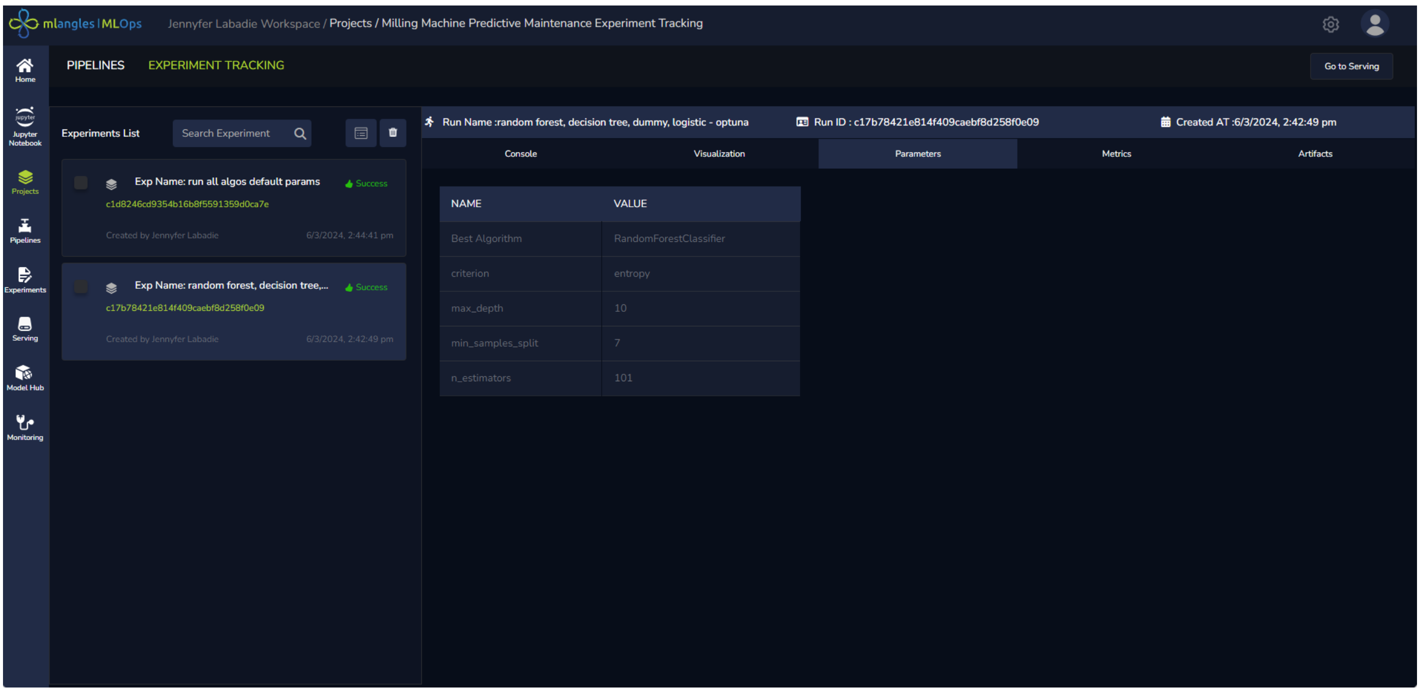
Slice Plot: A slice plot visualizes the relationship between two hyperparameters while fixing the values of other hyperparameters. It allows for the examination of interactions between hyperparameters and their effects on model performance, helping in identifying optimal parameter combinations.

Hyperparameter Importances Plot: This plot ranks the importance of hyperparameters based on their influence on model performance. It helps in identifying the most influential hyperparameters, guiding further optimization efforts or feature selection strategies.

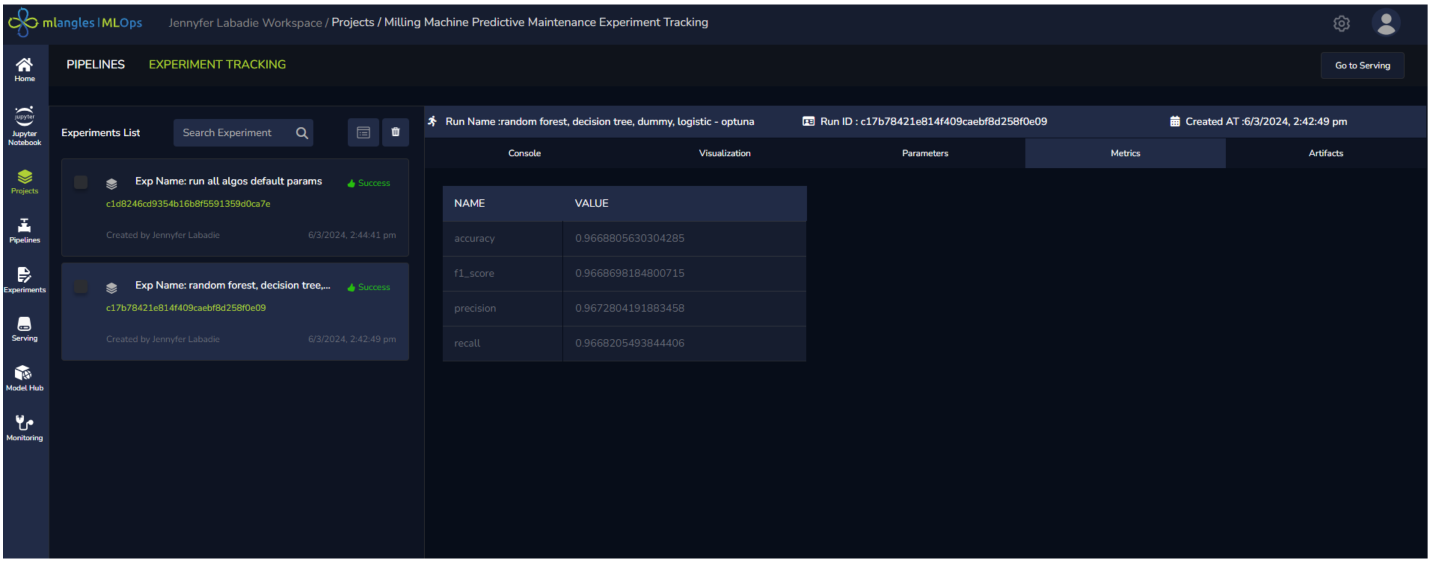
Parallel Coordinate Plot: This plot visualizes high-dimensional hyperparameter spaces by representing each hyperparameter as a vertical axis and each point in the plot as a hyperparameter configuration. Lines connecting points represent hyperparameter configurations, enabling the exploration of relationships and patterns across multiple hyperparameters simultaneously.



Additionally, extracting parameters used during training allows for reproducibility and transparency, ensuring that the model's settings are documented and accessible for future reference.



Metrics like accuracy, precision, f1score and recall provide the information about the performance of the model. Accuracy measures overall correctness, Precision focuses on the accuracy of positive predictions, Recall emphasizes capturing all positive instances, and the F1 score balances precision and recall. These metrics help in assessing the effectiveness and robustness of classification models.

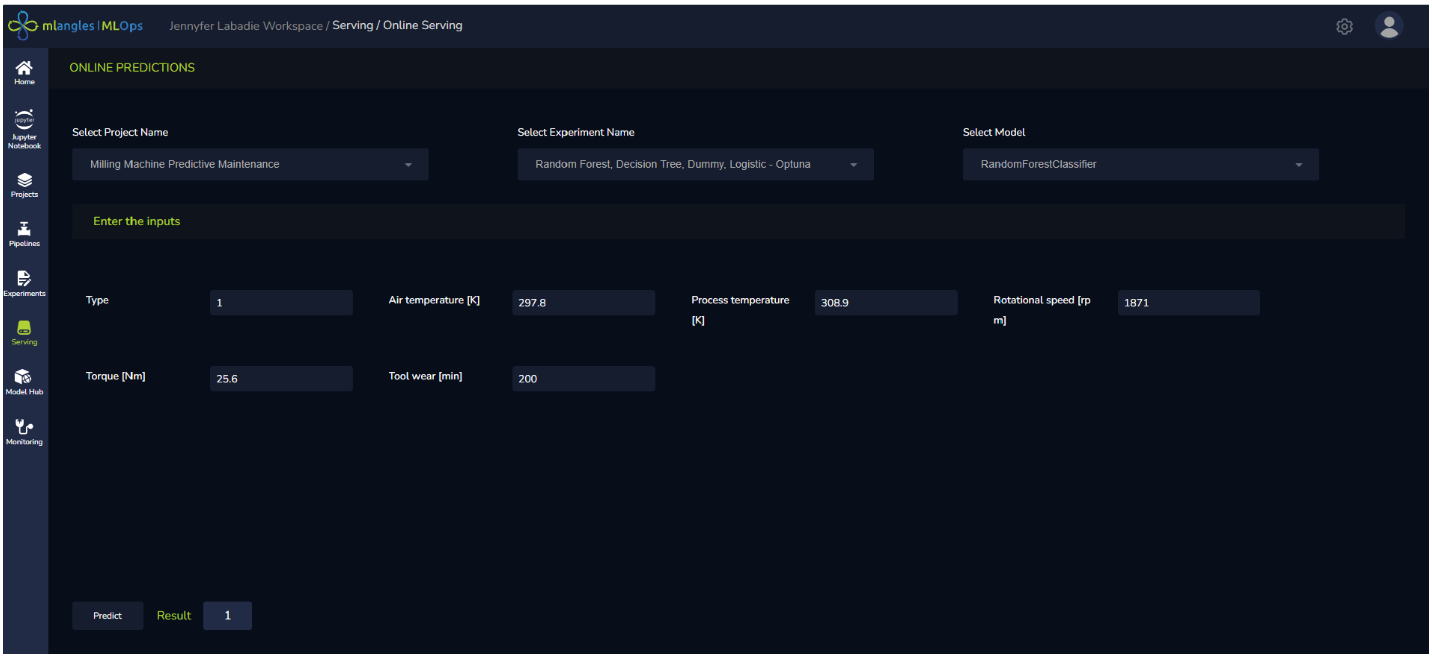


Model Versioning:

- Models are sensitive to a plethora of hyperparameters and parameters, including learning rate, loss function, and optimizers.
- Consequently, a model selected for training, with both the model and final data versions remaining constant but changes in parameters, may yield differing performance metrics.
- These diverse model versions can be uploaded to the model hub, facilitating the management of multiple iterations and variations.

Step 3: Prediction/ Serving

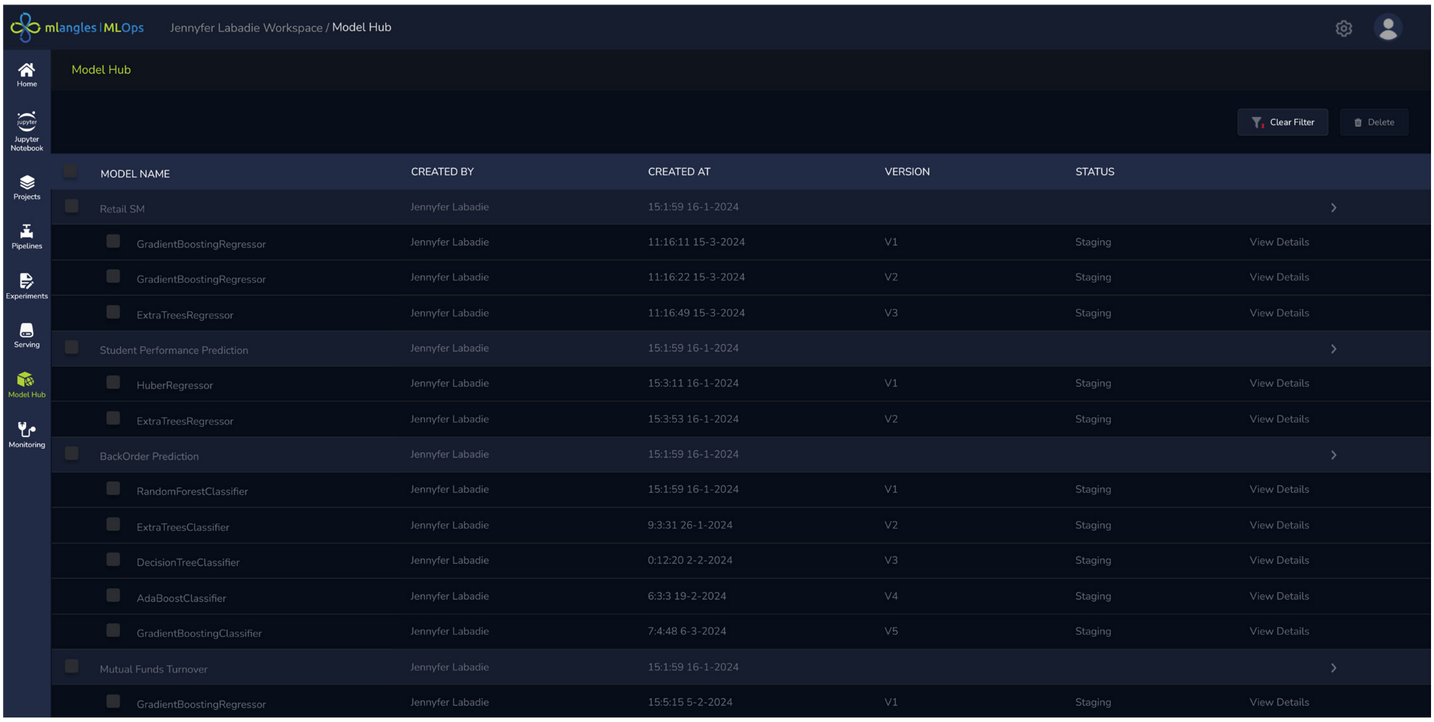
During model serving, a single data point is tested, and a result of 0 indicates that the milling machine has not failed. This real-time prediction provides immediate feedback on the machine's current operational status, enabling timely maintenance interventions and ensuring continuous production efficiency.



Model Hub:

- Trained models are uploaded to the model hub, whereupon deployment, a REST API endpoint is automatically generated.

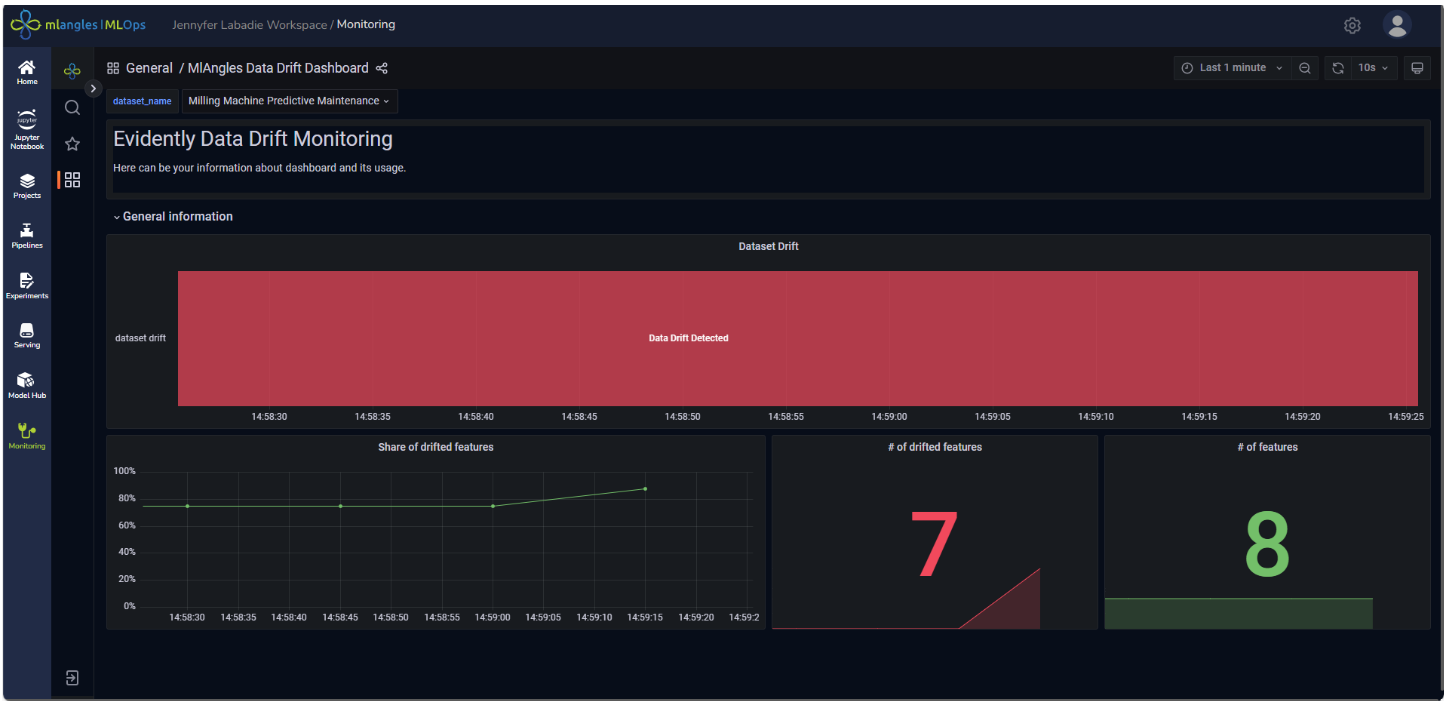
- Data is transmitted to this endpoint as a request, triggering the model to execute a prediction and return the output as the response to the request.



Step 4: Monitoring

The Below screen shows that there is drift in the data and 6 out of 8 features in the dataset have drifted. Data drift refers to the phenomenon where the statistical properties of the data change over time in a deployed machine learning model. This could be due to changes in the underlying data distribution, data collection process, or external factors influencing the data. When data drift occurs, the relationships between features and the target variable may change, impacting the model's performance and reliability.

The share of drifted features refers to the proportion of features in the dataset that have experienced a significant change or drift in their statistical properties. As these features undergo drift, their relationships with the target variable may become less relevant or even misleading, leading to decreased model accuracy and effectiveness. Therefore, monitoring and addressing data drift are essential to maintain the model's performance and ensure its continued relevance in production environments.



Conclusion

In summary, predictive maintenance for milling machines employs machine learning to prevent failures. The dataset highlights relationships between operational parameters and failures. Data analysis includes handling missing values, duplicates, and feature engineering. Training uses algorithms like random forest, decision tree, and logistic regression with Optuna for optimization. Evaluation metrics ensure model performance, while monitoring data drift maintains effectiveness. Overall, this approach optimizes maintenance schedules, reduces downtime, and enhances operational efficiency.

Business Impact of mlangles Predictive AI

- **Cost Reduction:** Predictive maintenance can reduce maintenance costs by up to 30% and decrease downtime by 50%. mlangles' Predictive AI helps avoid unexpected downtime and costly repairs by identifying potential issues before they escalate, thereby reducing maintenance costs.

- **Increased Equipment Efficiency:** Improve equipment uptime by 10-20% and increase productivity by 20-25% with the help of Predictive AI maintenance. Accurate prediction of needs helps milling machines operate at optimal levels, maximizing productivity and reducing idle time.

- **Extended Equipment Lifespan:** A study by PwC found that predictive maintenance can extend the lifespan of machinery by up to 20% compared to reactive maintenance approaches. Thus, mlangles' Predictive AI minimizes the need for premature replacements and reduces capital expenditures.

- **Improved Safety:** By detecting potential malfunctions in advance, mlangles' Predictive AI enhances workplace safety by preventing accidents associated with equipment failures.

- **Enhanced Planning and Resource Allocation:** Predictive maintenance can reduce maintenance planning time by 20% and improve resource allocation efficiency by up to 45%. With mlangles' Predictive AI, maintenance schedules enable better planning of resources and manpower, ensuring that maintenance tasks are carried out efficiently without disrupting production schedules.

- In conclusion, mlangles' Predictive AI empowers businesses to transform their approach to maintenance, resulting in cost savings, improved operational efficiency, and a safer working environment.

To setup Demo

Info.mlangles@cloudangles.com

Visit: www.mlangles.ai