CAPSTONE PROJECT

MACHINE LEARNING PROJECT

(PREDICTIVE MAINTENANCE FOR INDUSTRIAL MACHINES)

Presented By:

Parth Bhagyesh Dave Swarrnim Startup and Innovation University – Computer Engineering

edunet foundation

OUTLINE

- **Problem Statement**
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

In modern industries, **unexpected machinery failures** can lead to costly downtime and delays. To address this, the aim is to **develop a predictive maintenance model** that can **anticipate failures before they happen**. By **analysing sensor data** from machines, the model will learn to **detect patterns that typically occur before a failure**. The focus is on creating a **multiclass classification system** capable of predicting **specific failure types** such as:

1. **Tool wear failure**
2. **Heat dissipation failure**
3. **Power failure**
4. **Random failures**
5. **Overstrain failures**

This solution empowers industries to **perform maintenance proactively**, reducing **unexpected breakdowns**, **downtime**, and **overall operational costs**.

PROPOSED SOLUTION

- In response to unplanned equipment failures, our solution involves developing a predictive maintenance model powered by sensor data and machine learning.
- **Data Collection:**
  - Acquired historical sensor data from industrial machines.
  - Dataset includes operational metrics and labeled failure types.
  - Covers variables like temperature, torque, speed, and tool wear.
- **Data Preprocessing:**
  - Manual Step: Dropped non-informative column UDI
  - Automated Steps via IBM AutoAI:
    - Handled missing values (if any)
    - Encoded categorical variable (Type)
    - Scaled and normalized numerical features
    - Split data into training and testing sets
    - Selected suitable transformations for model compatibility

PROPOSED SOLUTION

- **Machine Learning Algorithm:**
  - IBM AutoAI selected 3 algorithms as per data and select multiple models and create pipelines.
  - Top Models Selected:
    1. Snap Random Forest Classifier
    2. Random Forest Classifier
    3. Snap Decision Tree Classifier
  - Models were compared based on:
    1. Prediction Accuracy
    2. Training and Inference Time
    3. Deployment:
    4. Best model deployed using IBM Cloud AutoAI platform.
    5. Deployment considerations:
    6. High Accuracy

PROPOSED SOLUTION

- **Deployment:**
  - Deployed as a REST API endpoint.
  - Model accessibility:
    a) Python
    b) Javascript
    c) Java
    d) Scala
    e) cURL
- **Evaluation:**
  - Model performance was assessed using metrics like accuracy and confusion matrix.
  - Evaluation considered:
    1. Model precision
    2. Class-wise prediction performance
    3. Deployment response time
  - Snap Random Forest Classifier achieved 99.5% accuracy.

SYSTEM APPROACH

System Requirements:

- Platform: IBM Watson Studio (Cloud-based AutoAI)
- RAM: 8 GB or higher (recommended for local processing)
- Processor: Intel i5/i7 or equivalent
- Storage: more than 1 GB for dataset and model artifacts
- Stable internet connection

Libraries & Tools Used

- requests – For interacting with deployed model API
- Cloud Object Storage: IBM Cloud Object Storage is a highly scalable cloud storage service, designed for high durability, resiliency and security. Store, manage and access your data via our self-service portal and RESTful APIs. Connect applications directly to Cloud Object Storage use other IBM Cloud Services with your data.
- IBM Watsonx.ai Studio: provides the environment and tools for you to collaboratively work on data to solve your business problems. You can choose the tools you need to analyze and visualize data, to cleanse and shape data, to ingest streaming data, or to create and train machine learning models.
- IBM Watsonx.ai Runtime: IBM watsonx.ai Runtime provides a full range of tools and services so that you can build, train, and deploy Machine Learning models. Choose the tool with the level of automation or autonomy that matches your needs.
- Jupyter notebook: to work with data and testing model.

ALGORITHM & DEPLOYMENT

- **Algorithm Selection:**
- IBM AutoAI automatically explored various machine learning pipelines. It shortlisted three high-performing algorithms:
  - **Snap Random Forest Classifier** (Best Performer)
  - Random Forest Classifier
  - Snap Decision Tree Classifier
    Snap Random Forest was chosen due to its high accuracy (99.5%) and ability to handle complex classification tasks effectively.
- **Data Input:**
- The model utilized features such as:

a) Air temperature [K]

b) Process temperature [K]

c) Rotational speed [rpm]

d) Torque [Nm]

e) Tool wear [min]

f) Machine type and product ID (encoded)

g) The target variable was the **Failure Type** (e.g., Tool Wear, Heat Dissipation, etc.).

ALGORITHM & DEPLOYMENT

- **Training Process:**
  - Model training was handled by IBM AutoAI
  - AutoAI performed automatic feature engineering and hyperparameter optimization (HPO-1, HPO-2)
  - Dataset split for training and validation internally by AutoAI pipelines
- **Prediction Process:**
  - The trained model predicts the **failure type** based on sensor input
  - Model supports batch and real-time prediction through deployed API
  - Prediction confidence scores (probabilities) are also provided

RESULT

UDI	Product ID	Type	Air temperature [K]	Process temperature [K]	Rotational speed [rpm]	Torque [Nm]	Tool wea
1	M14860	M	298.1	308.6	1551	42.8	
2	L47181	L	298.2	308.7	1408	46.3	
3	L47182	L	298.1	308.5	1498	49.4	
4	L47183	L	298.2	308.6	1433	39.5	
5	L47184	L	298.2	308.7	1408	40	
6	M14865	M	298.1	308.6	1425	41.9	
7	L47186	L	298.1	308.6	1558	42.4	
8	L47187	L	298.1	308.6	1527	40.2	
9	M14868	M	298.3	308.7	1667	28.6	
10	M14869	M	298.5	309	1741	28	
11	H29424	H	298.4	308.9	1782	23.9	
12	H29425	H	298.6	309.1	1423	44.3	
13	M14872	M	298.6	309.1	1339	51.1	
14	M14873	M	298.6	309.2	1742	30	
15	L47194	L	298.6	309.2	2035	19.6	
16	L47195	L	298.6	309.2	1542	48.4	
17	M14876	M	298.6	309.2	1311	46.6	
18	M14877	M	298.7	309.2	1410	45.6	
19	H29432	H	298.8	309.2	1306	54.5	
20	M14879	M	298.9	309.3	1632	32.5	
21	H29434	H	298.9	309.3	1375	42.7	
22	L47201	L	298.8	309.3	1450	44.8	
23	M14882	M	298.9	309.3	1581	30.7	
24	L47203	L	299	309.4	1758	25.7	
25	M14884	M	299	309.4	1561	37.3	
26	L47205	L	299	309.5	1861	23.3	
27	L47206	L	299.1	309.5	1512	39	

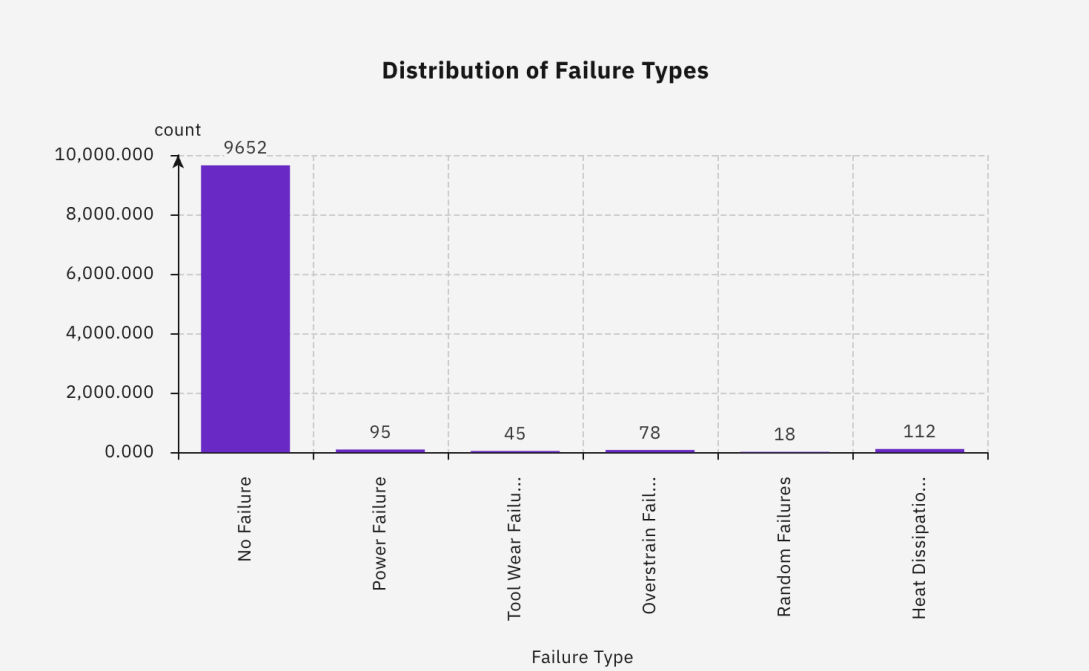
Product ID	Type	Air temperature [K]	Process temperature [K]	Rotational speed [rpm]	Torque [Nm]	Tool wear [
M14860	M	298.1	308.6	1551	42.8	
L47181	L	298.2	308.7	1408	46.3	
L47182	L	298.1	308.5	1498	49.4	
L47183	L	298.2	308.6	1433	39.5	
L47184	L	298.2	308.7	1408	40	
M14865	M	298.1	308.6	1425	41.9	
L47186	L	298.1	308.6	1558	42.4	
L47187	L	298.1	308.6	1527	40.2	
M14868	M	298.3	308.7	1667	28.6	
M14869	M	298.5	309	1741	28	
H29424	H	298.4	308.9	1782	23.9	
H29425	H	298.6	309.1	1423	44.3	
M14872	M	298.6	309.1	1339	51.1	
M14873	M	298.6	309.2	1742	30	
L47194	L	298.6	309.2	2035	19.6	
L47195	L	298.6	309.2	1542	48.4	
M14876	M	298.6	309.2	1311	46.6	
M14877	M	298.7	309.2	1410	45.6	
H29432	H	298.8	309.2	1306	54.5	
M14879	M	298.9	309.3	1632	32.5	
H29434	H	298.9	309.3	1375	42.7	
L47201	L	298.8	309.3	1450	44.8	
M14882	M	298.9	309.3	1581	30.7	
L47203	L	299	309.4	1758	25.7	
M14884	M	299	309.4	1561	37.3	
L47205	L	299	309.5	1861	23.3	
L47206	L	299.1	309.5	1512	39	

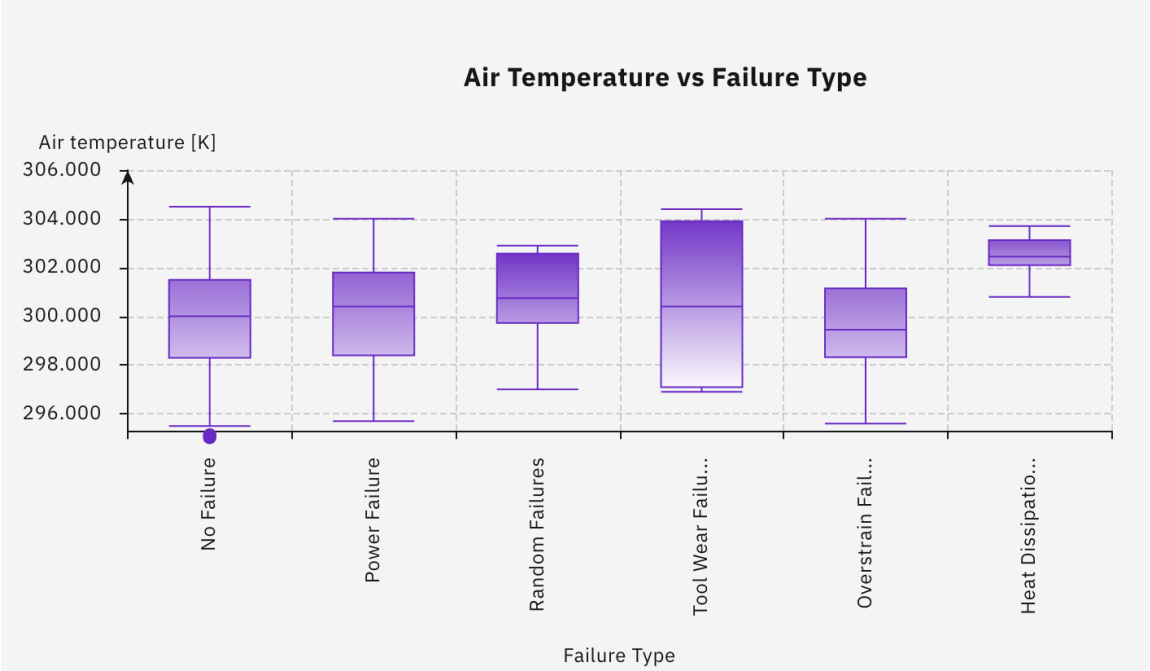
Remove unwanted column from the data such as **UDI**

edunet foundation

RESULT

Data Exploration Graphs (Before Modelling)





Bar Plot :

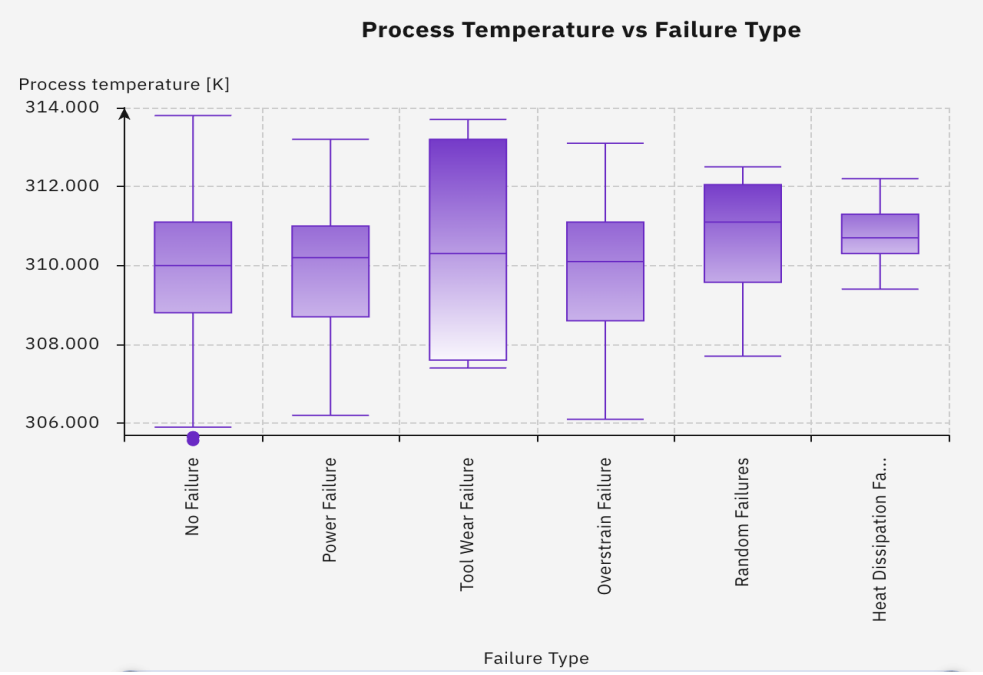
- Shows distribution of Failure Type (e.g., “No - Failure” vs. “Tool Wear”)
- Detect class imbalance

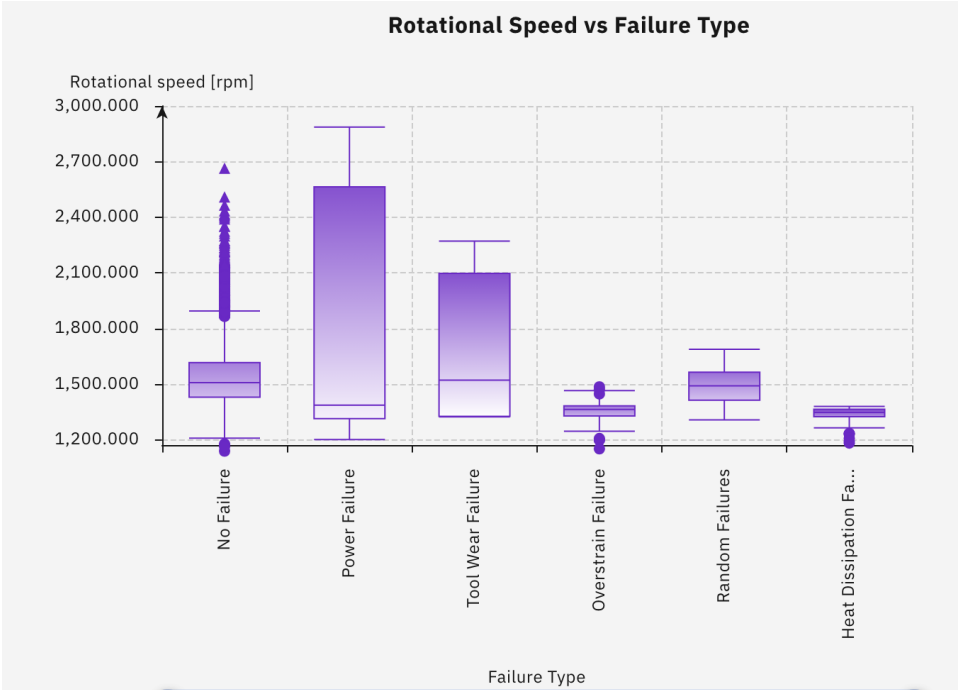
Box Plots:

- Compare sensor readings
- Shows spread, median, and outliers

edunet foundation

RESULT





Box Plots:

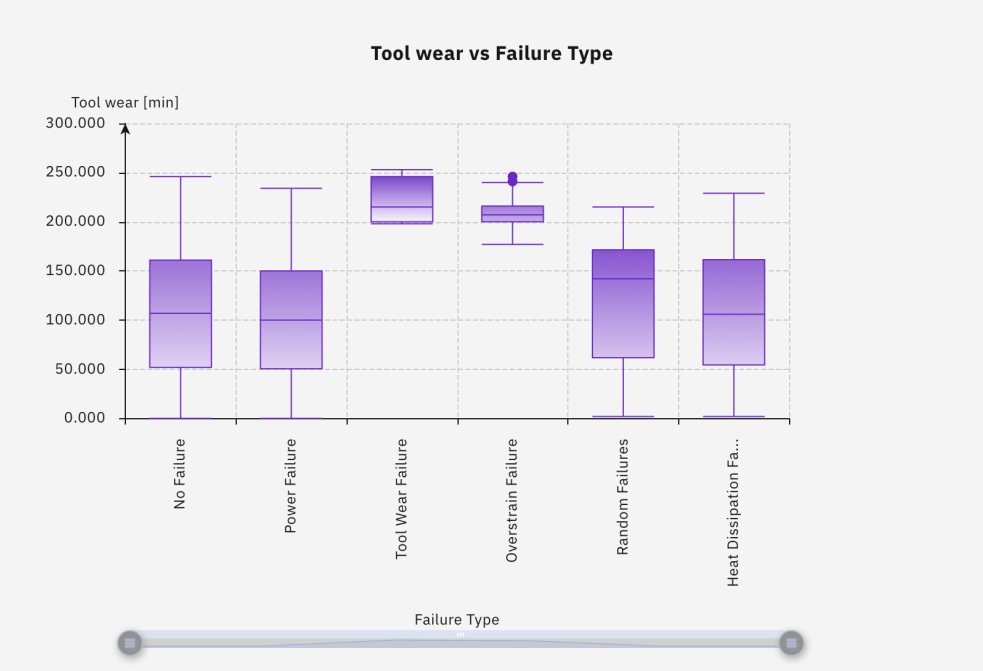
- Compare sensor readings
- Shows spread, median, and outliers

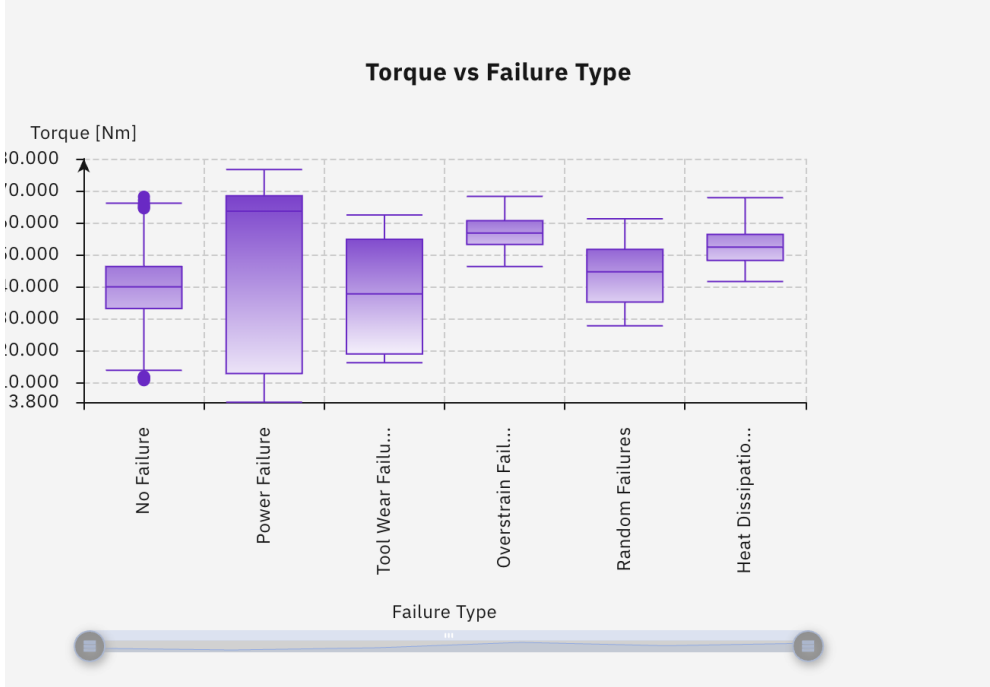
Box Plots:

- Compare sensor readings
- Shows spread, median, and outliers

edunet foundation

RESULT





Box Plots:

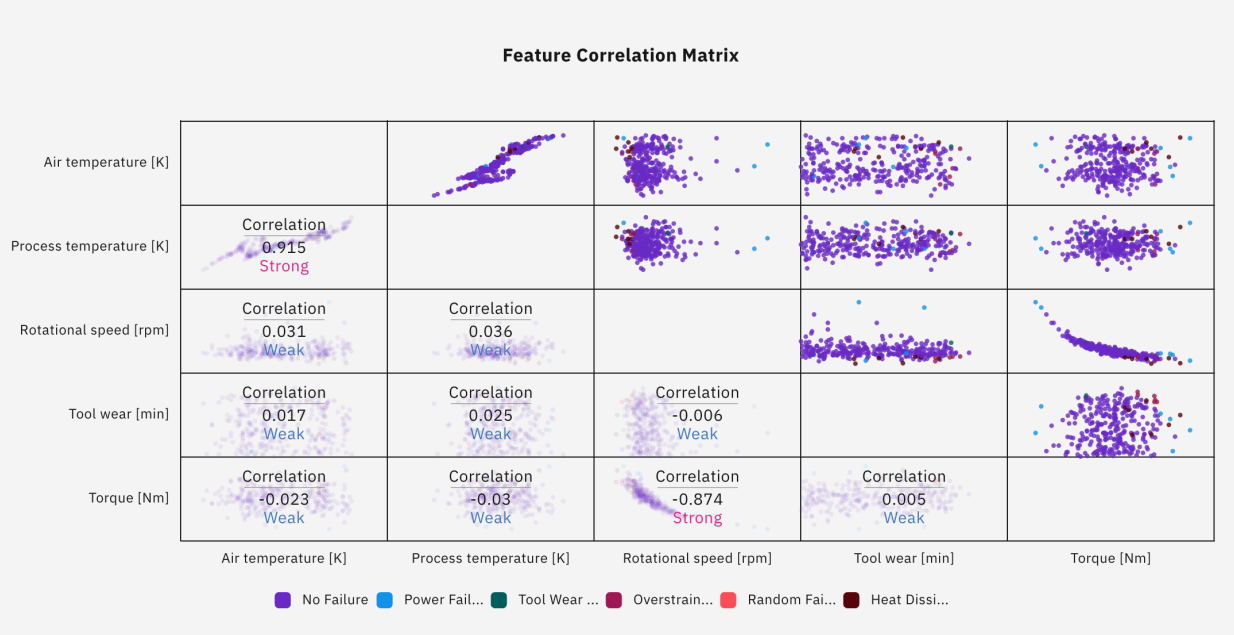
- Compare sensor readings
- Shows spread, median, and outliers

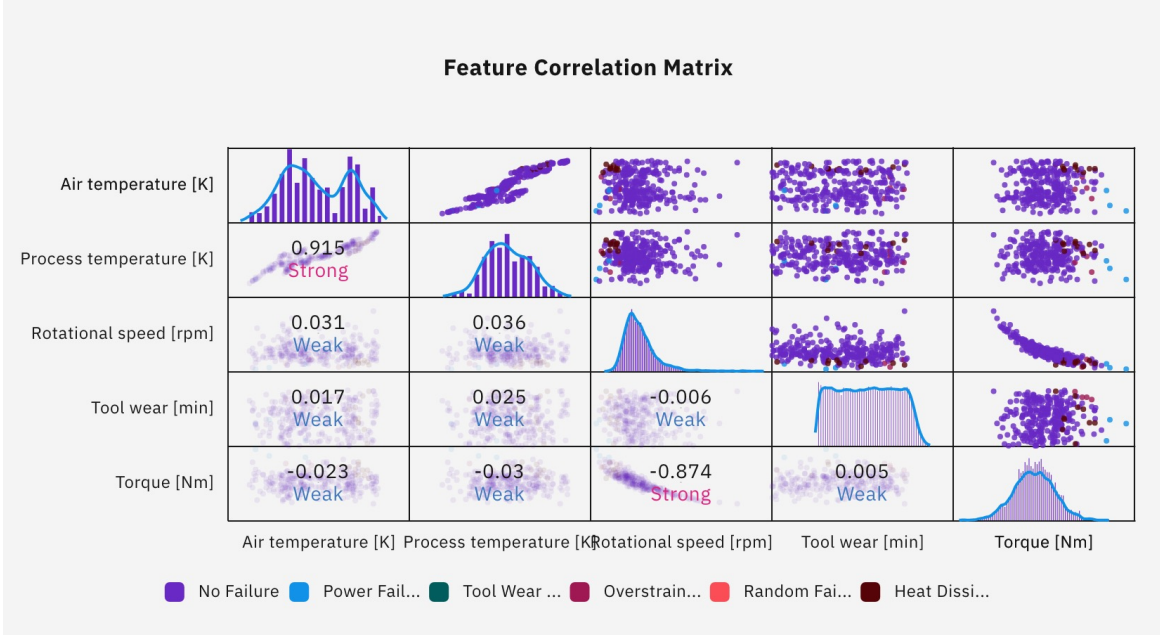
Box Plots:

- Compare sensor readings
- Shows spread, median, and outliers

edunet foundation

RESULT





Pair Plot

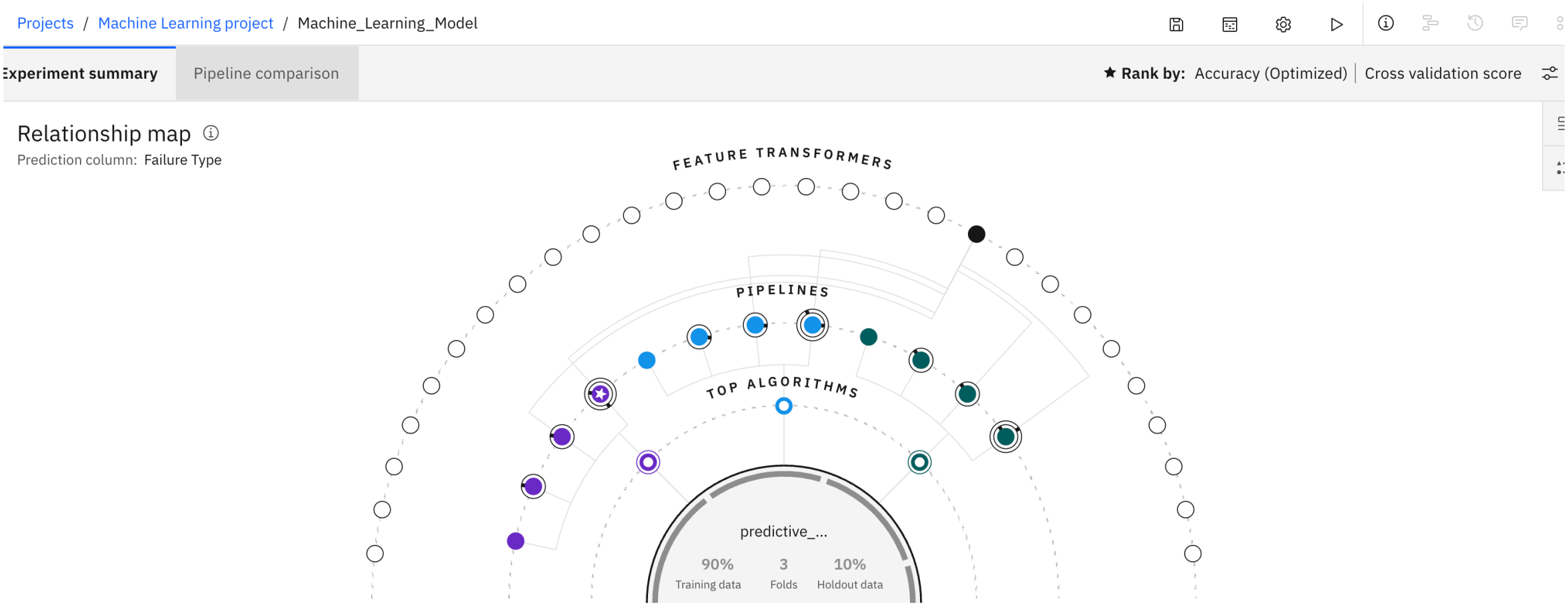
- Pairwise scatter plots of numerical features, coloured by failure type
- Helps detect clusters or relationships

Correlation Heatmap

- Identify relationships between features
- Useful for feature selection or understanding redundancy

edunet foundation

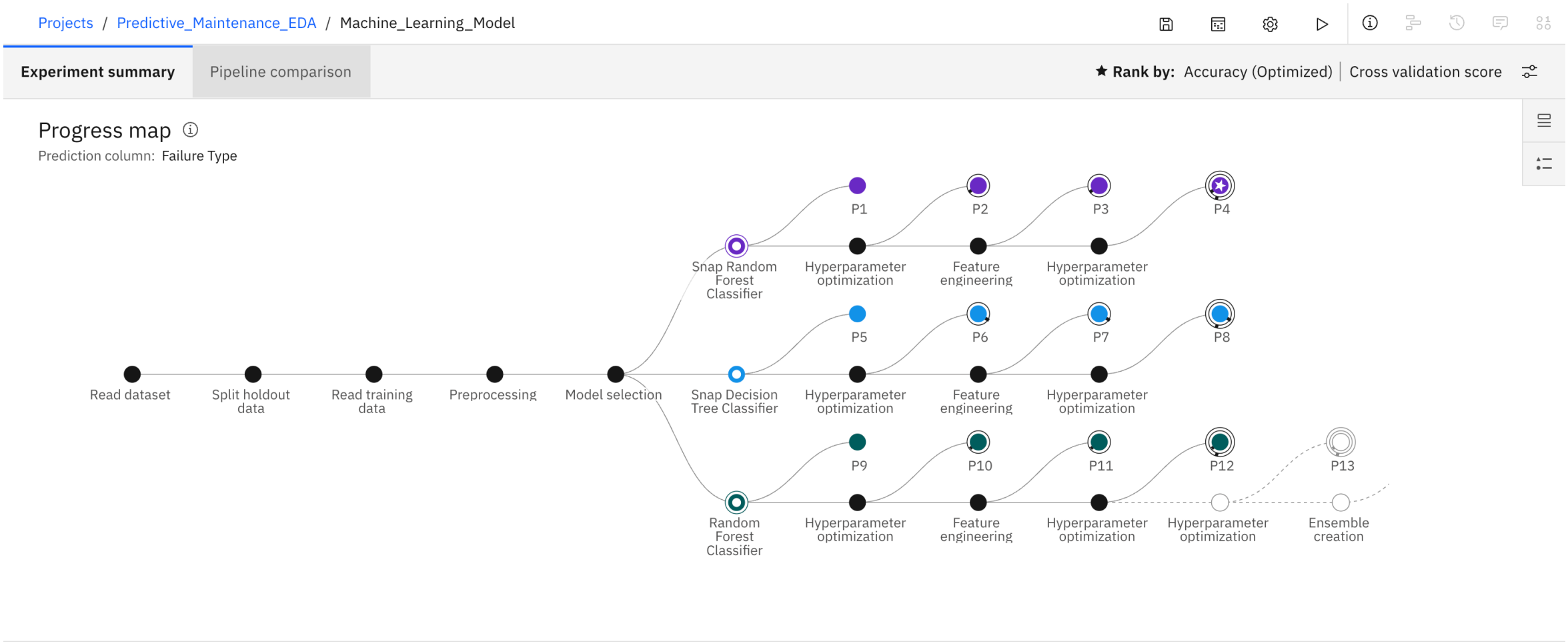
RESULT



IBM WatsonX AutoAI performed EDA and built models using multiclass classification algorithm

edunet foundation

RESULT



IBM WatsonX AutoAI performed EDA and built models using multiclass classification algorithm

edunet foundation

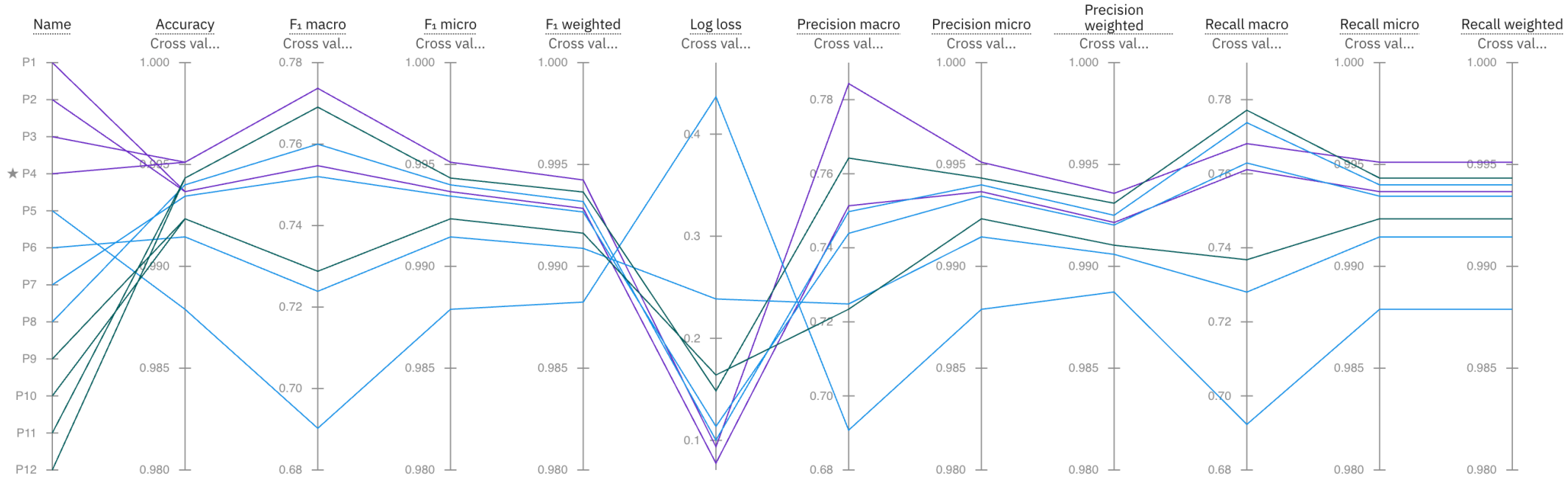
RESULT

Projects / Predictive_Maintenance_EDA / Machine_Learning_Model

Experiment summary | **Pipeline comparison** | ★ **Rank by:** Accuracy (Optimized) | Cross validation score

Metric chart

Prediction column: Failure Type



Pipeline comparison based on accuracy

edunet foundation

RESULT

Pipeline leaderboard

	Rank	Name	Algorithm	Specialization	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	**Pipeline 4**	Snap Random Forest Classifier		0.995	HPO-1 FE HPO-2	00:00:37
	2	**Pipeline 3**	Snap Random Forest Classifier		0.995	HPO-1 FE	00:00:29
	3	**Pipeline 12**	Random Forest Classifier		0.994	HPO-1 FE HPO-2	00:00:54
	4	**Pipeline 11**	Random Forest Classifier		0.994	HPO-1 FE	00:00:38
	5	**Pipeline 8**	Snap Decision Tree Classifier		0.994	HPO-1 FE HPO-2	00:00:25
	6	**Pipeline 2**	Snap Random Forest Classifier		0.994	HPO-1	00:00:09
	7	**Pipeline 1**	Snap Random Forest Classifier		0.994	*None*	00:00:04
	8	**Pipeline 7**	Snap Decision Tree Classifier		0.993	HPO-1 FE	00:00:22
	9	**Pipeline 10**	Random Forest Classifier		0.992	HPO-1	00:00:11
	10	**Pipeline 9**	Random Forest Classifier		0.992	*None*	00:00:02
	11	**Pipeline 6**	Snap Decision Tree Classifier		0.991	HPO-1	00:00:04
	12	**Pipeline 5**	Snap Decision Tree Classifier		0.988	*None*	00:00:01

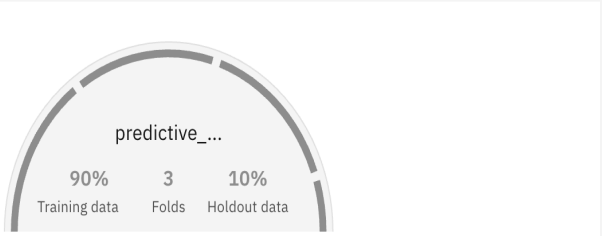
Pipeline comparison based on accuracy

edunet foundation

RESULT

Training data split

The percentage of the training data used to train and optimize the pipelines vs the percentage used as holdout data, to test the pipelines. You can split the training data or upload a separate file for testing.



Training and testing data

Access the rows split into the holdout and training data sets for this experiment programmatically.

Reconfigure your experiment and select **Training data only** in the **Experiment Settings > Data source settings > Final training data set** setting to enable this feature.

Training sources

These are the sources you selected for training this experiment.

predictive_maintenance_csv_shaped.csv
Size: 0.472 MB

Preprocessing summary

1 **Classify features**

Feature types View

Categorical	Numerical	Auto-classified
0	0	9

2 **Initial feature engineering**

Column Exclusions View

User excluded columns
0

Text feature engineering View

User specified text columns
0

3 **Missing value imputation**

Missing value imputation information is not supported for this prediction type.

Summary of EDA and Model creation by IBM AutoAI

RESULT

File Edit View Run Kernel Help Trusted Memory:276 / 8192 MB

Code Python 3.11

Model Testing

[3]:

```
import requests

# Replace with your actual IBM Cloud API key
API_KEY = "add your API Key"

# Get access token
token_response = requests.post(
    'https://iam.cloud.ibm.com/identity/token',
    data={"apikey": API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'}
)
mltoken = token_response.json()["access_token"]
header = {'Content-Type': 'application/json', 'Authorization': f'Bearer {mltoken}'}

# Payload: test data including known Target labels
payload_scoring = {
    "input_data": [
        {
            "fields": ["Product ID", "Type", "Air temperature [K]", "Process temperature [K]",
                       "Rotational speed [rpm]", "Torque [Nm]", "Tool wear [min]", "Target"],
            "values": [
                ["M18096","M",300.8,309.4,1342,62.4,113,1], # Heat Dissipation Failure
                ["L47184","L",298.2,308.7,1408,40,9,0], # No Failure
                ["L47428","L",298,308.3,1362,56.8,216,1], # Overstrain Failure
                ["L47622","L",297.4,308.5,1399,61.5,61,1], # Power Failure
                ["H32866","H",301.6,310.5,1602,32.3,2,0], # Random Failures
                ["L50791","L",301.7,310.9,1405,46.4,207,1] # Tool Wear Failure

            ]
        }
    ]
}
```

Model testing and outcome

RESULT

Model testing and outcome

CONCLUSION

- Successfully developed a **predictive maintenance model** using real-time sensor data to anticipate industrial machine failures.
- Leveraged **IBM AutoAI** for automated data preprocessing, algorithm selection, and model optimization.
- **Snap Random Forest Classifier** achieved an impressive **99.5% accuracy**, making it the ideal choice for deployment.
- The system enables **proactive maintenance**, minimizing downtime, reducing operational costs, and improving machine reliability.
- Deployed model is accessible through multiple platforms (Python, Java, JavaScript, cURL), ensuring **easy integration** into existing systems.

FUTURE SCOPE

- **Integration of More Sensor Types**: Extend the current dataset by integrating additional machine sensors (e.g., vibration, acoustic, or thermal imaging) for richer insights and improved failure prediction.
- **Model Generalization**: Adapt the trained model to work across different types of industrial machines and environments to ensure scalability and robustness.
- **Real-Time Edge Deployment**: Implement the model on edge devices directly connected to machines for real-time monitoring and faster response to anomalies.
- **Automated Retraining Pipelines**: Set up an automated feedback loop where new operational data helps in continuous retraining and refinement of the model.
- **Cost-Aware Maintenance Scheduling**: Enhance the system to not only predict failures but also recommend cost-optimized maintenance schedules based on operational hours and failure severity.
- **Wider Platform Integration**: Expand model usability by integrating with industrial platforms like PLCs, SCADA systems, or cloud-based monitoring dashboards for seamless operations.

REFERENCES

- Dataset: https://www.kaggle.com/datasets/shivamb/machine-predictive-maintenance-classification
- IBM Cloud Dashboard: https://cloud.ibm.com/login
- IBM watsonx.ai Studio: https://cloud.ibm.com/services/data-science-experience/crn%3Av1%3Abluemix%3Apublic%3Adata-science-experience%3Aau-syd%3Aa%2Faab9312cccf3498ca3d3f656bf4e348d%3A04329999-f229-40fd-9edb-b53031652aa5%3A%3A?paneId=manage
- Watson Studio learning path: https://dataplatform.cloud.ibm.com/docs/content/wsj/getting-started/quickstart-tutorials.html?context=wx
- Documentation for Cloud Pak for Data as a Service: https://au-syd.dai.cloud.ibm.com/docs/content/wsj/getting-started/welcome-main.html?context=cpdaas&audience=wdp
- Documentation for IBM watsonx as a Service: https://au-syd.dai.cloud.ibm.com/docs/content/wsj/getting-started/welcome-main.html?context=wx&audience=wdp
- GITHUB: https://github.com/parthdaveprojects/machine_learning_project#

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence
IBM SkillsBuild

Parth Dave

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 16, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/6391bb5a-49a9-4cb2-bb04-7fdd2097d0d4

IBM

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution
IBM SkillsBuild

Parth Dave

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 17, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/7c5abf52-f3aa-4902-b84e-161704c75bba

IBM

edunet foundation

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Parth Dave

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 17 Jul 2025 (GMT)

Learning hours: 20 mins

edunet foundation

THANK YOU

edunet foundation