AI Engineering:
Managing Risks in AI Systems

Raghu Sangwan, Professor of Software Engineering
Youakim Badr, Professor of Artificial Intelligence
Sahana Suryanarayana, Graduate Student
Manjunath Venkat, Graduate Student

School of Graduate Professional Studies
Penn State University, Malvern, PA 19355

itag | TECH360°

What is an AI System ?

- Artificial intelligence (AI) is finding application in systems that make decisions affecting the safety of people, assets, and environment
- This class of systems are referred to as AI systems
  - They try to mimic human-like intelligence in tasks such as object detection, natural language understanding, and autonomous decision making
  - They do so through the use large datasets, machine learning, computer vision, natural language processing and computational power to analyze, learn, and make decisions

Risks in AI Systems
Design and Implementation Challenges

1. *Lack of formal specification* can cause mismatch between designer objectives and what the system learns

2. *Lack of implementation transparency* make them incomprehensible for design review and inspection

3. *Lack of control of high dimensionality effects* where a miniscule change in input can change the prediction of the system

4. Data used for training and testing the system *cannot be guaranteed to be drawn from the same probability distribution*

5. *Lack of quality data* can cause the system to learn behavior that is not desired

itag | TECH360°

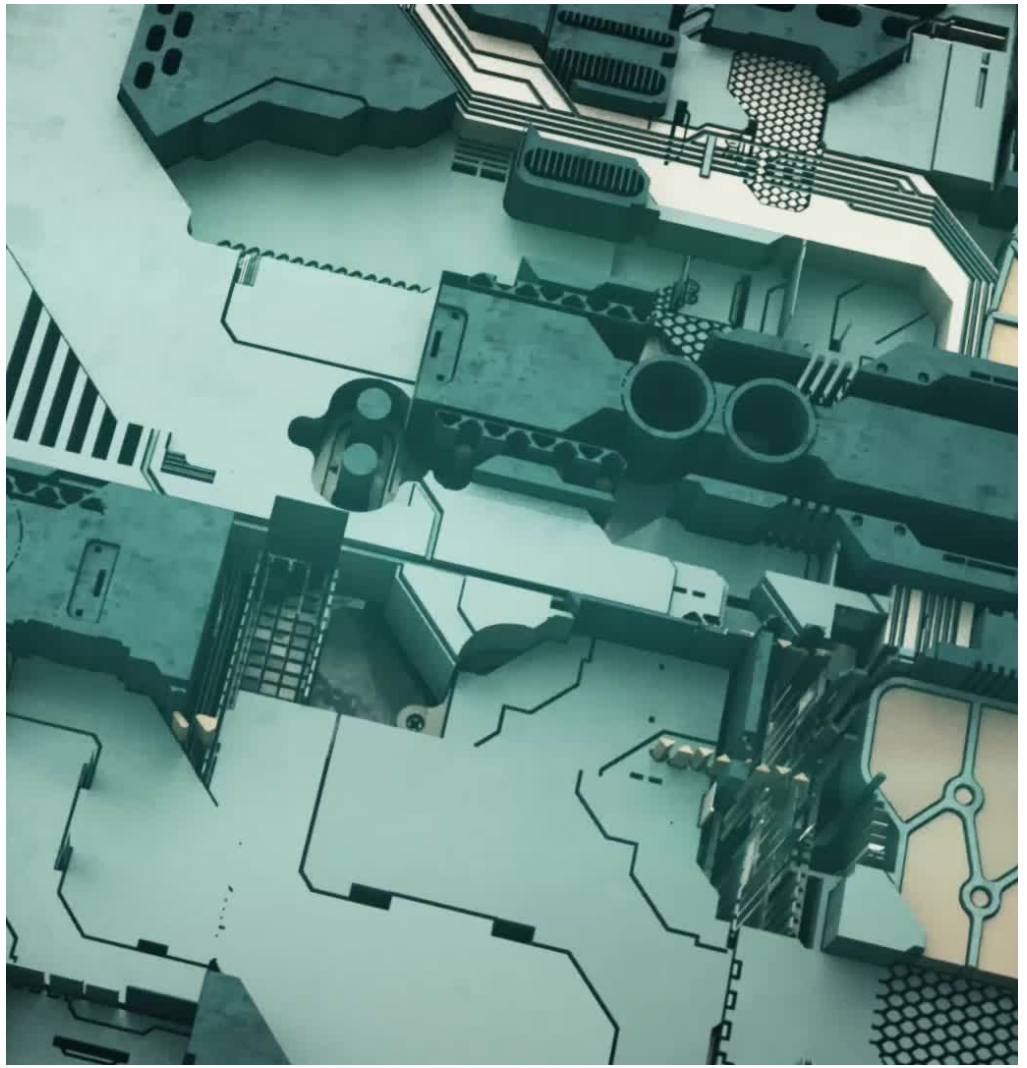
A Case Study of Predictive Maintenance of Aircraft Engines

- **Research Focus:** Improving safety of Prognostics and Health Management (PHM) systems
- **Industry Significance:** Targeted at aviation, prioritizing passenger safety and operational efficiency.
- **Key Challenges Addressed:**
  - Enhancing prediction accuracy
  - Minimizing false negatives
- **Expected Outcomes:**
  - Precise predictive maintenance
  - Ensured safety with human in the loop

itag

Dataset Overview: “Turbofan Engine Degradation Simulation Data”

- The dataset utilized in this research is the “Turbofan Engine Degradation Simulation Data”
- This dataset is generated through the use of the C-MAPSS simulator, an acronym for 'Commercial Modular Aero-Propulsion Simulation’
- This simulator replicates a large commercial turbofan engine
- Four distinct simulation datasets have been created, considering various combinations of operational conditions and fault modes



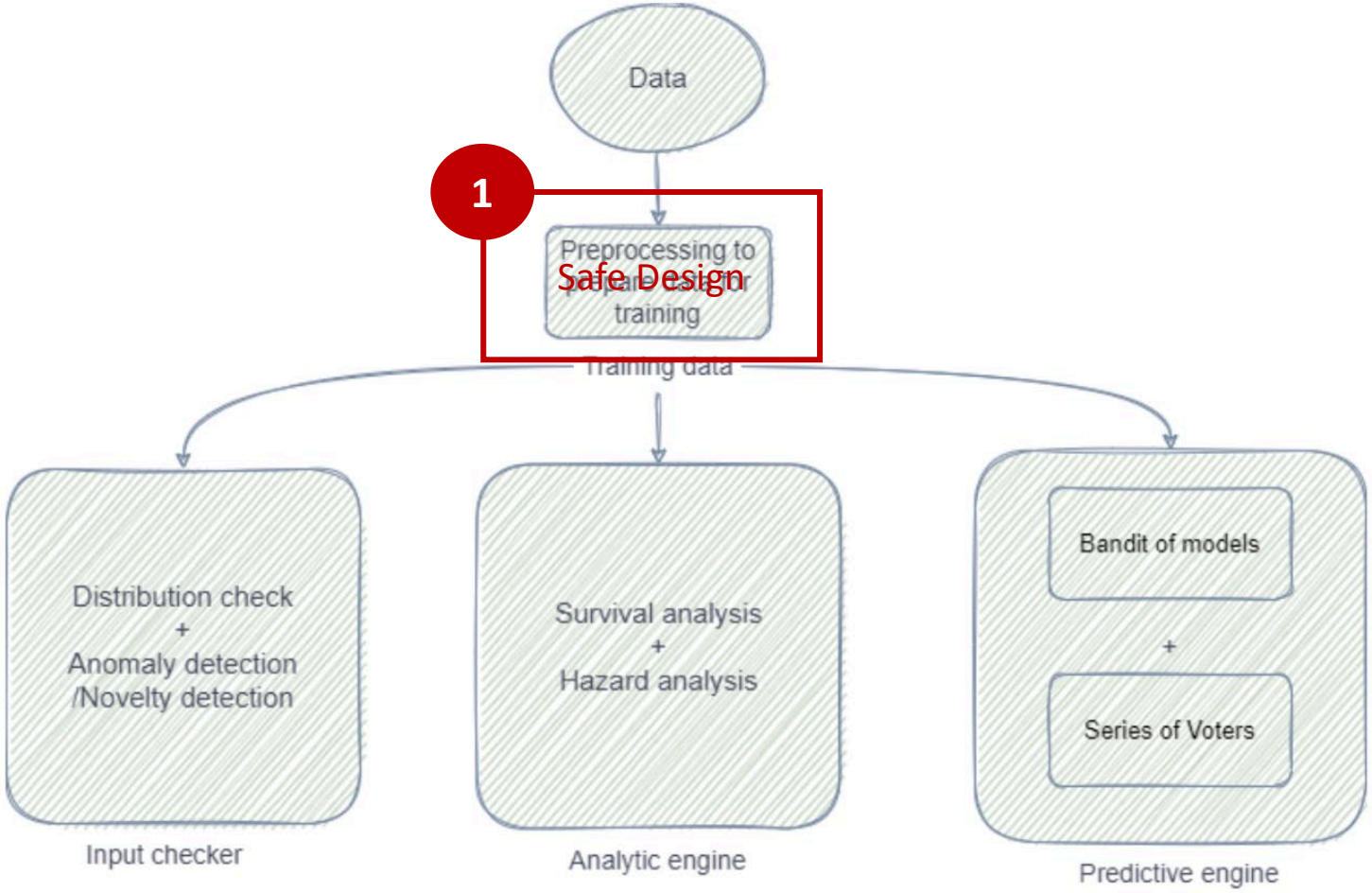
Research Agenda

Research Objective	Techniques	Rationale
Safe Design: Design the system with safety in mind	• Exploratory data analysis (EDA) and data preprocessing • Feature engineering	• Our predictive model is going to be just as good as the data, therefore, processing the data in the right manner is important to help the models arrive at the right conclusions

Safe Margin: Improve the system resilience against data shifts and perturbations	• Monitor data drift using one or more of:
	  • Statistical tests
	  • Clustering
	  • Anomaly detection
	  • Novelty detection
		• Since machine learning based predictive models learn from data, it is vital to understand and identify any change in distribution of data as change in distribution may affect the behavior of the models.

Safe Fail: Keep the system safe at the time of failure	• Hazard analysis and survival analysis • Bandit of Models • Explainable AI	• Analyze component hazard and survival ratios to track their performance trends • Introduce diversity in prediction approaches using a bandit of models and voting • Use monitoring function notifying human in the loop

Proposed Architecture



Exploratory Data Analysis and Input Checker

EDA is a crucial preliminary step in understanding and preparing the dataset

EDA helps uncover data patterns, check data quality, and assess distribution characteristics

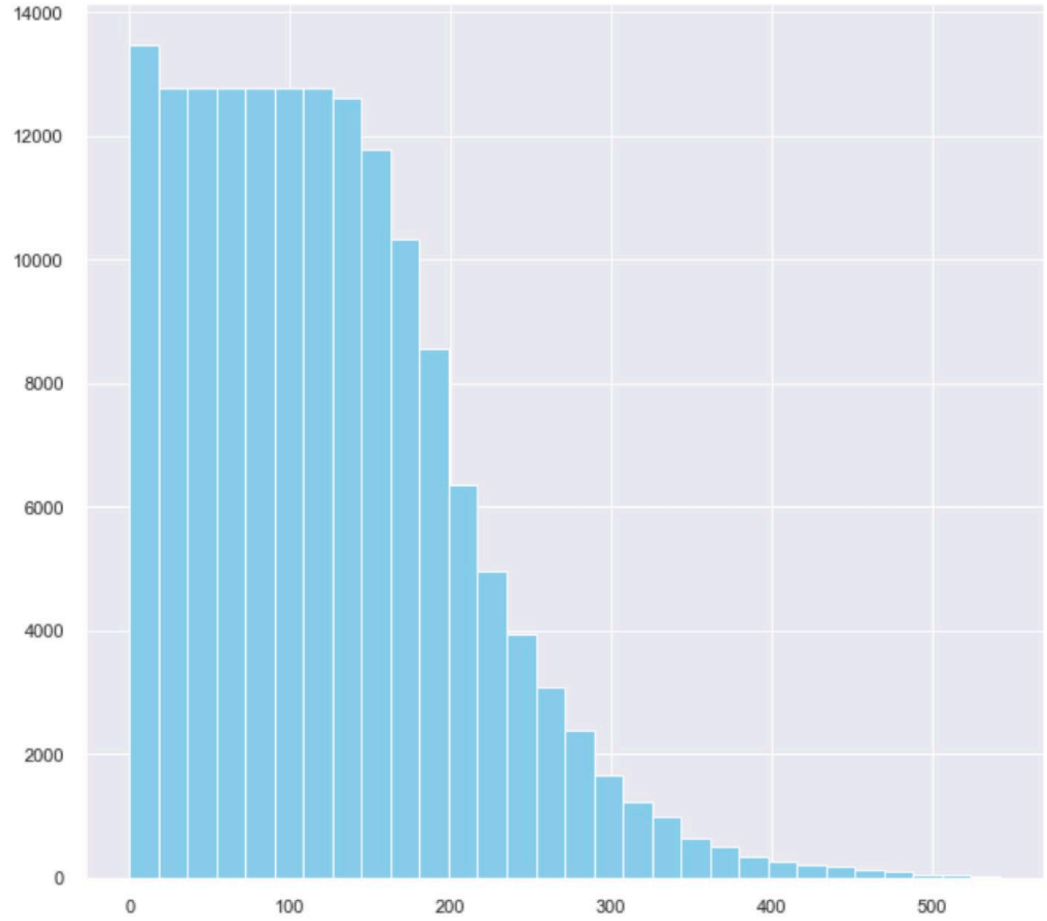
EDA includes data preprocessing tasks such as handling missing values, transforming data, and addressing non-normal distributions

Input checker performs data distribution check and, anomaly and novelty detection

Initial insights gained from EDA and input checking guide subsequent steps in the analysis.

EDA Insights and Findings

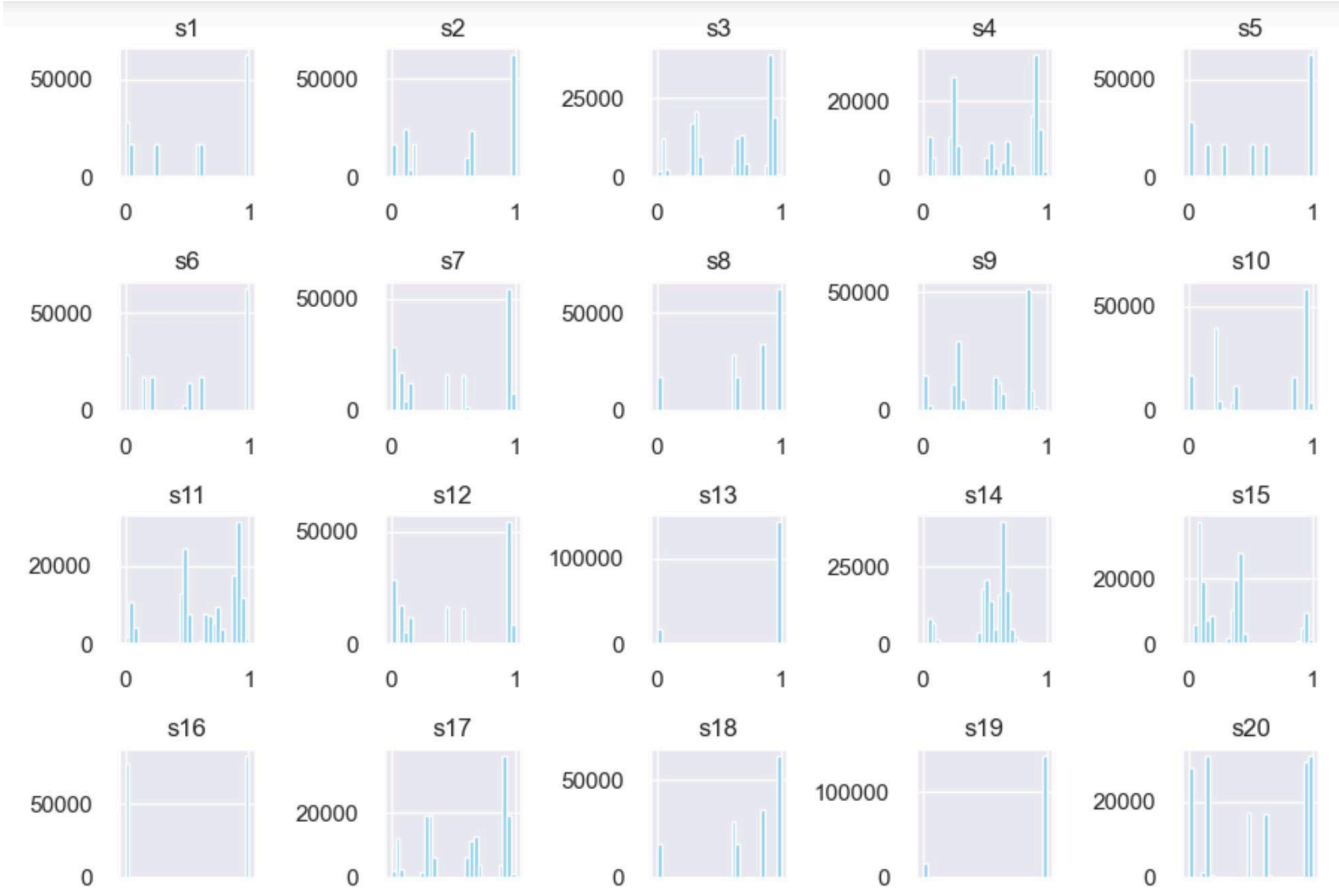
- **Null Values**: No null values were detected in any dataset features, ensuring data completeness
- **Mean Cycles**: The mean number of cycles for engines in the training set is 123
- **Histogram of remaining cycles**: Most engines start degrading after 140 cycles, with breakdowns around 200 cycles



RC Histogram

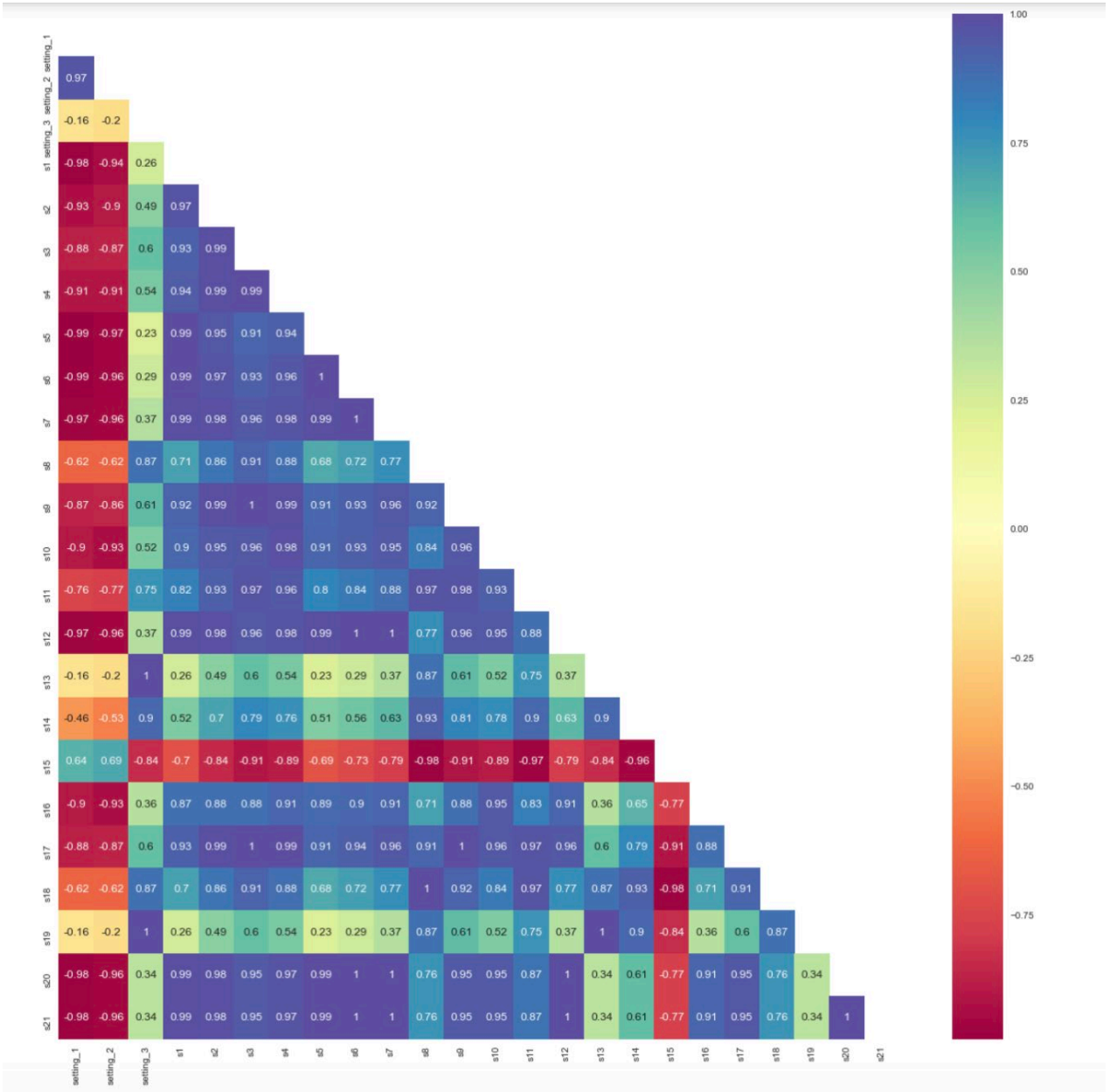
EDA Insights and Findings

Sensor Values Distribution: Sensor values in train, test, and PHM08 datasets have different scales which are normalized and scaled between 0 & 1



EDA Insights and Findings

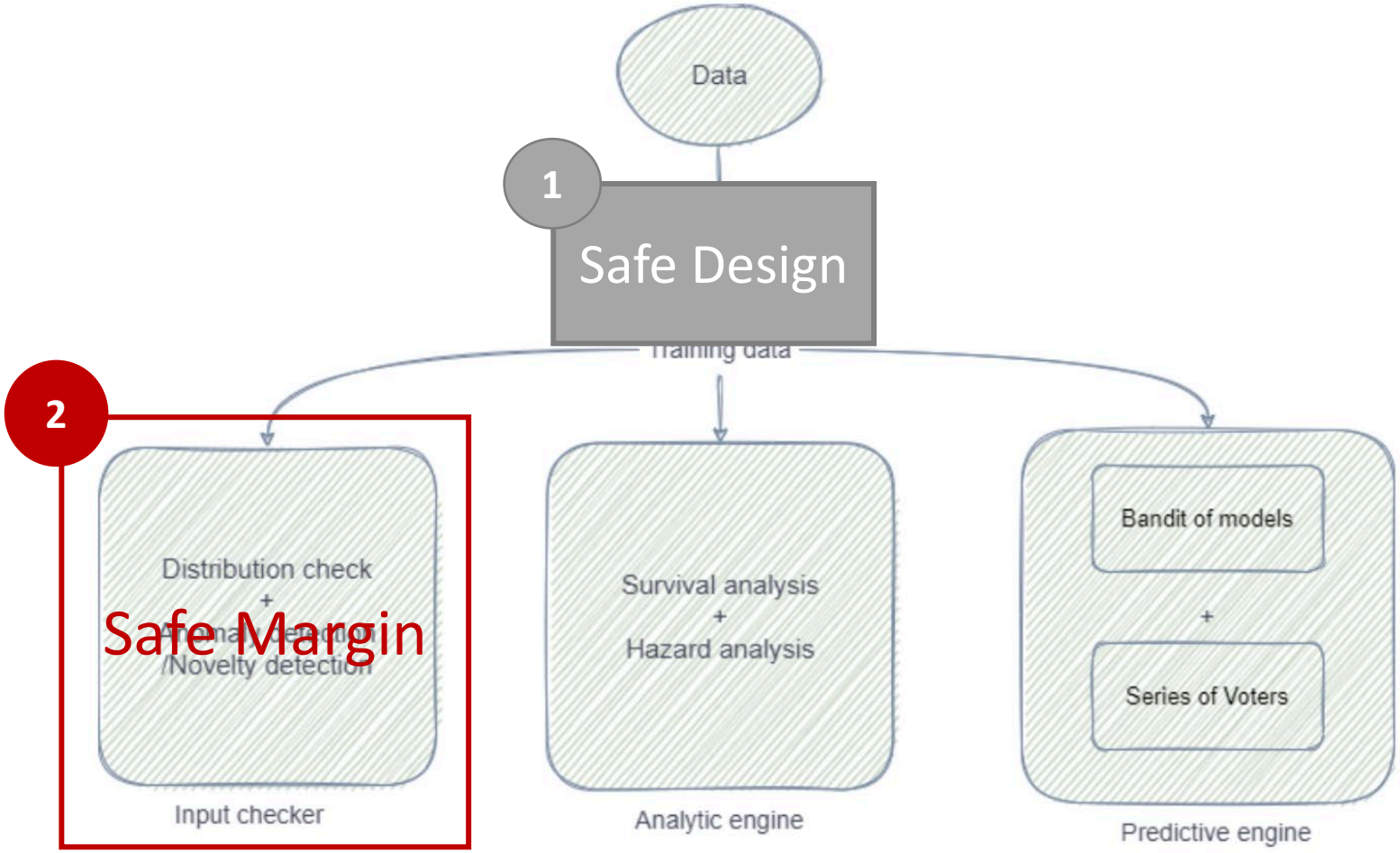
1. **Correlation Matrix**: Many features exhibit strong positive (> 90%) and negative correlations (> 90%), highlighting multicollinearity
2. **Feature Selection**: Boruta is preferred over PCA for feature selection due to skewness, non-linear distribution, and multicollinearity in the normalized features



Feature Selection with Boruta

- Boruta is a feature selection method used in our research for predictive maintenance
- Boruta adopts an "all-relevant" approach, identifying all features relevant for prediction
- In our case, Boruta recommended 18 out of 26 features for predictive maintenance
- Selected features include unit number, time in cycles, sensor readings (s1, s2, s3, etc.), and setting parameters (setting_1).
- Boruta helps address multicollinearity by reducing correlated features
- This process streamlines the feature set, enhancing predictive model efficiency and accuracy

Proposed Architecture

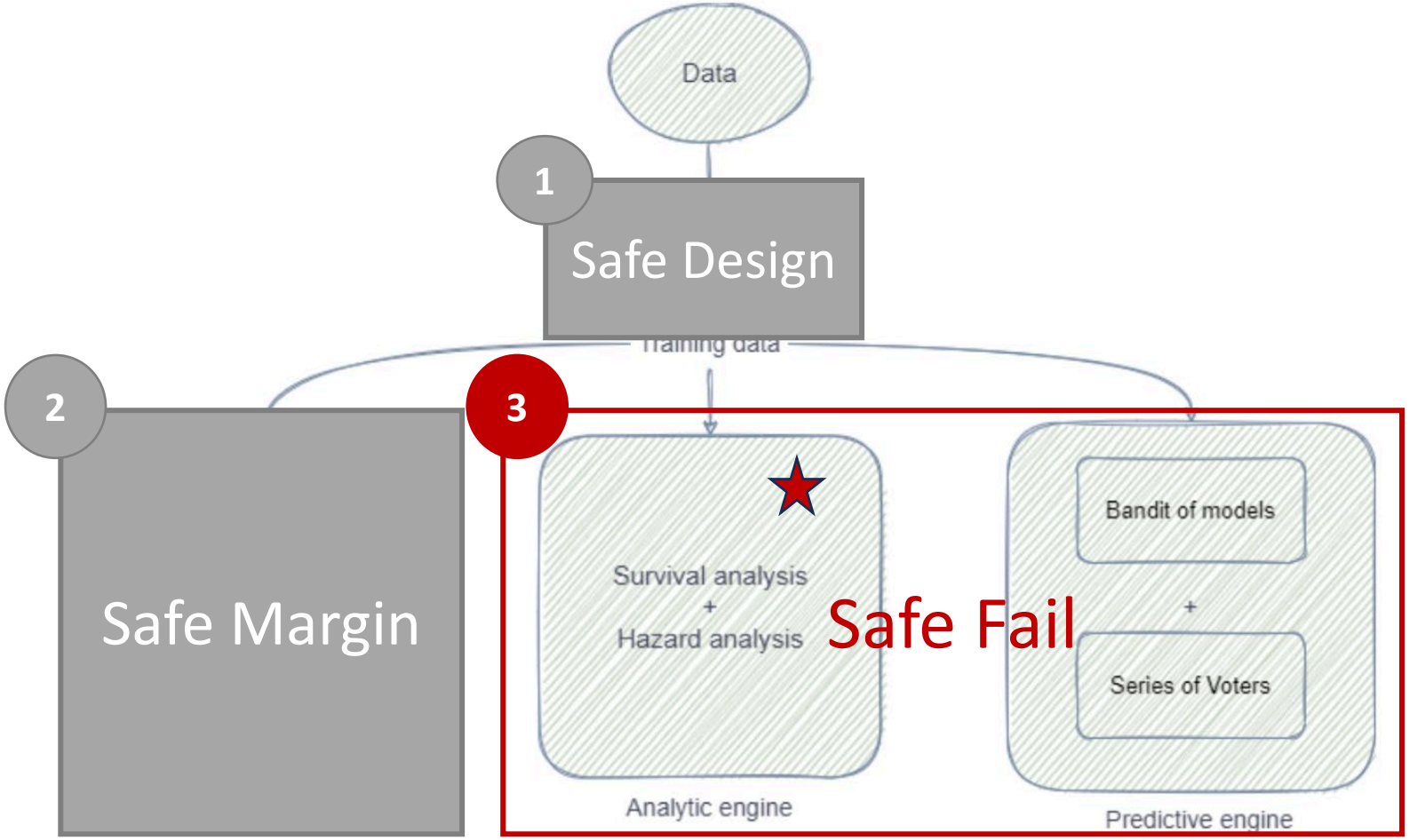


Detecting Data Drift for Model Resilience

Data drift occurs when the distribution of new data deviates from the data the model was trained on.

1. **Statistical tests:** We employ statistical tests to detect signs of data drift by comparing the distribution of train, test, and PHM08 datasets
2. **Q-Q Plots:** Quantile-quantile plots help visualize data distribution, showing consistency in sensor values across datasets despite non-normal distributions.
3. **Clustering:** K-means clustering is an unsupervised technique used for grouping data points together based on similarity
4. **Anomaly Detection:** One-class SVM identifies outliers in data distribution, with data points at extreme value ranges often classified as anomalies
5. **Novelty Detection:** An extension of one-class SVM (also called local outlier factor) is used to detect how the density of a sample differs from those of its neighbors

Proposed Architecture

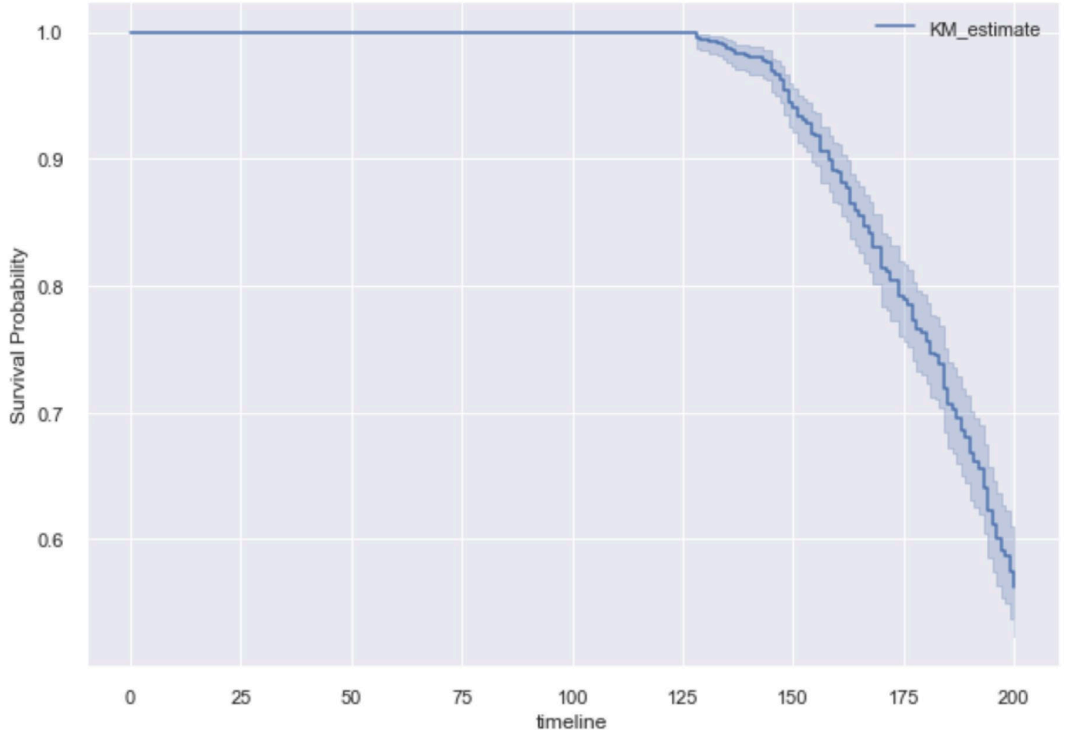


Kaplan-Meier Estimation

- Kaplan-Meier estimation is a statistical technique used for analyzing survivability or time-to-event data
- It's especially useful for predictive maintenance because it allows us to estimate and visualize survival probabilities over time
- This estimation is vital for understanding when engine breakdowns might occur and predicting maintenance needs based on time

Kaplan-Meier Survival Analysis Model Results

- **Train Set**: 100% survival probability for the first 125 cycles, dropping afterward. A 50% chance of engine survival at 200 cycles
- **Test Set**: Right-censored data with 100% survival probability up to approximately 75 cycles. A 50% probability of engine survival at 160 cycles
- **PHM08 Dataset**: 100% survival probability until 125 cycles, followed by a decline. A 50% chance of engine survival at 200 cycles



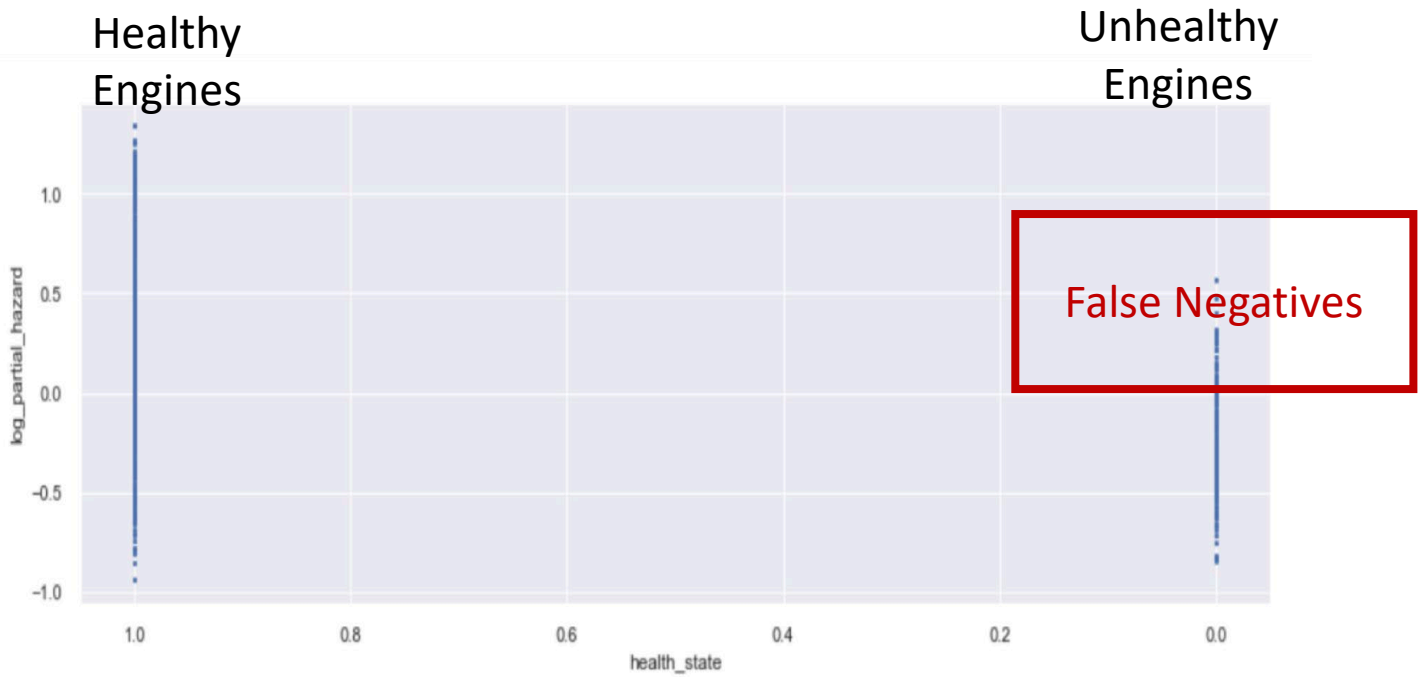
KM estimate of survival probability of the engines in train set

Cox-Proportional Hazard

- The Cox-Proportional Hazard model is employed alongside survival analysis to assess the impact of various features on survival
- It helps us understand how different features influence the risk of engine failure

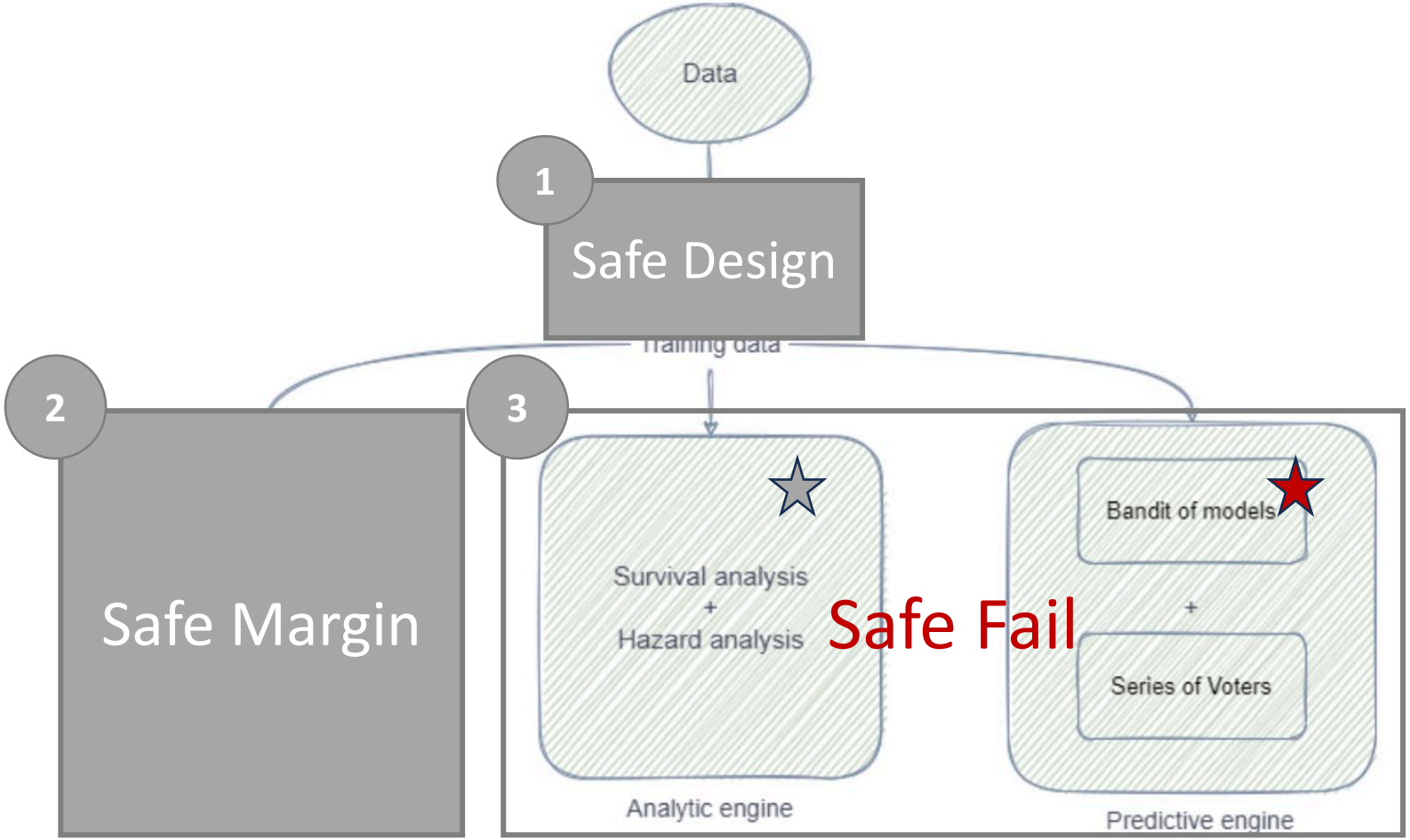
Hazard Analysis Model Results

- Figure shows a likelihood of type I error (false positives) but fewer cases of type II error (false negatives).



Scatterplot of log_partial_hazard vs health_state

Proposed Architecture



Model Diversity for Robust Predictive Maintenance

- In this research, we employ a combination of predictive models to comprehensively address the predictive maintenance of aircraft engines
  - RandomForest, XGBoost, SVM, and LSTM
- The choice of these models aims to provide versatility and diversity in addressing the complex task of predictive maintenance, considering various aspects and scenarios

Random Forest

- Random Forest (RF) is a machine learning model that utilizes an ensemble of decision trees
- RF stands out due to its versatility, interpretability, quick training times, and strong predictive performance
- It can effectively handle high feature correlation, which is common in real-world datasets
- Moreover, RF doesn't require feature scaling or normalization, making it suitable for the dataset characteristics

XGBoost

XGBoost, or Extreme Gradient Boosting, is another tree-based model that operates within a gradient boosting framework

What sets XGBoost apart is its ability to handle class imbalance and effectively address anomalies in the data

It assigns more weight to classes with low participation, improving the model's ability to predict rare events

This is crucial for predictive maintenance, where identifying rare failure events is of utmost importance

Support-Vector Machine

Support-Vector Machine (SVM) is a kernel-based model capable of performing both linear and nonlinear classification, regression, and outlier detection

We specifically employ the polynomial kernel with a degree of 3 in this research to handle nonlinear data relationships

SVMs are sensitive to the scales of features, so it's essential to standardize or normalize features before applying the model

itag | TECH360°

Long Short-Term Memory

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) that excels at capturing order dependence in sequence prediction problems

This model is particularly well-suited for handling continuous sensor readings, making it a valuable tool for analyzing aircraft engine data

LSTM considers information from previous inputs when processing current inputs, which is crucial for understanding the temporal relationships in the data

itag | TECH360°

Results

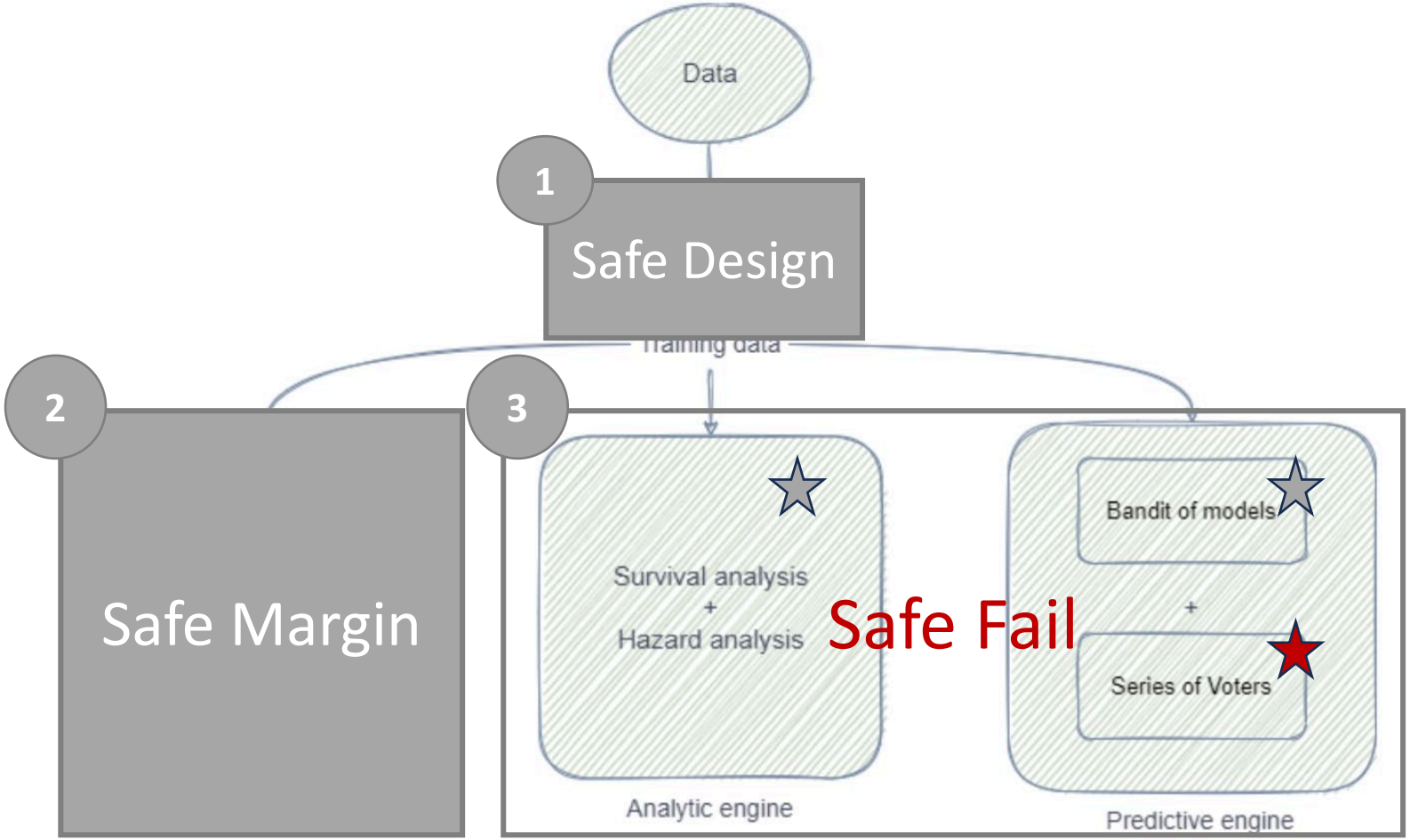
Model	Metrics			
	Accuracy	Recall	Precision	F1- score
RandomForest	85.518	86.2	87.606	86.89
XGBoost	85.423	86.06	87.558	86.805
SVM	83.63	84.5	85.97	85.17
LSTM	78.58	90.27	75.86	82.44

itag | TECH360°

Feature Importance Analysis

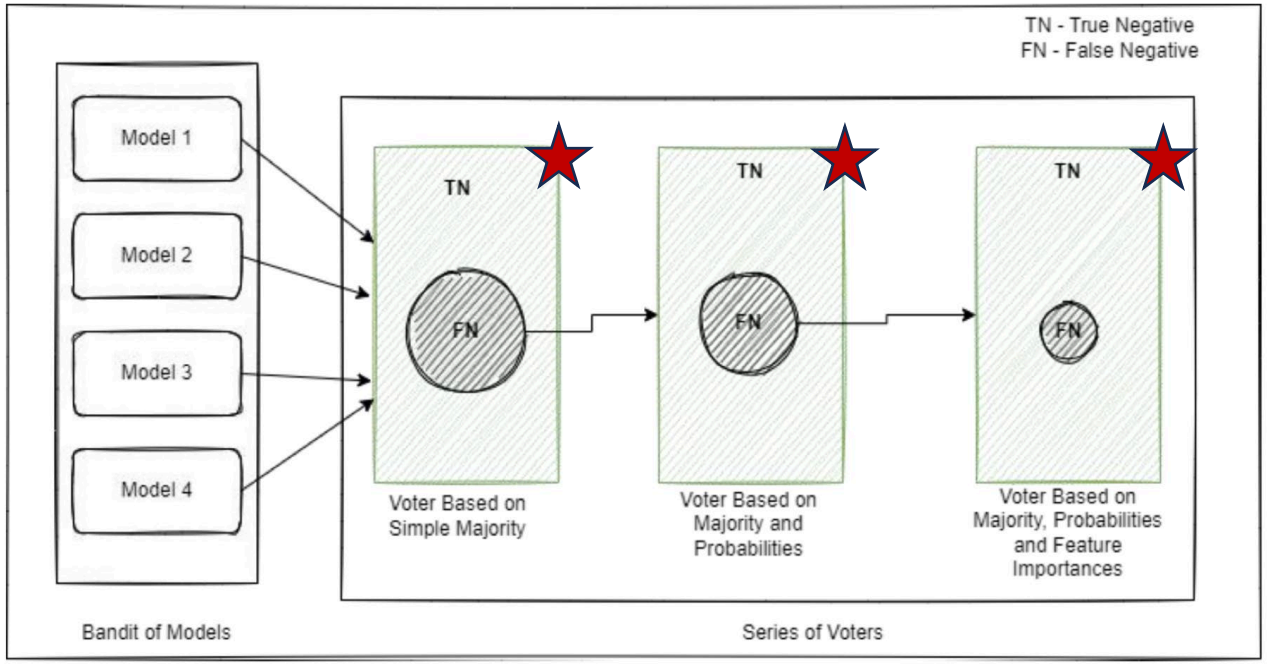
- **Feature Importance Analysis**: Analyzed feature importance using RandomForest, XGBoost, and hazard analysis.
- **Top 6 Features by RandomForest**: time_in_cycles, s13, s11, s4, s15, and s2.
- **Top 6 Features by X**GBoost: time_in_cycles, s15, s11, s13, s21, and s4.
- **Top 6 Features by hazard analysis**: s15, s21, s11, s20, s2 and s13.
- **Common Key Features**: Consistently recognized s11 and s15 as pivotal features across all models.
- These identified features play a pivotal role in the voter based predictive maintenance process, guiding decisions on when maintenance actions should be initiated to ensure aircraft engine safety and reliability.

Proposed Architecture



Enhanced Predictive Engine Architecture

- We use the concept of 'multi-armed bandits,' employing multiple models to perform the same task
- We use four different models and optimize each one for predictive maintenance
- An ensemble voter function integrates predictions from the four models, enhancing resiliency



Tables 1 and 2 provide insights into the performance of these filters, including the voter based on majority, voter based on majority and probability, and the voter based on majority, probability, and feature importance, in comparison to the four primary models, on both test data and the PHM08 dataset.

Model	False Negatives	% of total datapoints
RandomForest	91	12.87 %
XGBoost	93	13.15%
SVM	93	13.15%
LSTM	71	10.04%
Voter based on Simple majority	79	11.17%
Voter based on Majority and Probability	42	5.94%
Voter based on majority, probability, and feature importance	7	0.99%

Overview of false negatives from 707 test points

Model	False Negatives	% of total datapoints
RandomForest	3275	7.13%
XGBoost	3254	7.09%
SVM	3697	8.05%
LSTM	3208	6.99%
Voter based on Simple majority	2848	6.20%
Voter based on Majority and Probability	1186	2.58%
Voter based on majority, probability, and feature importance	59	0.13%

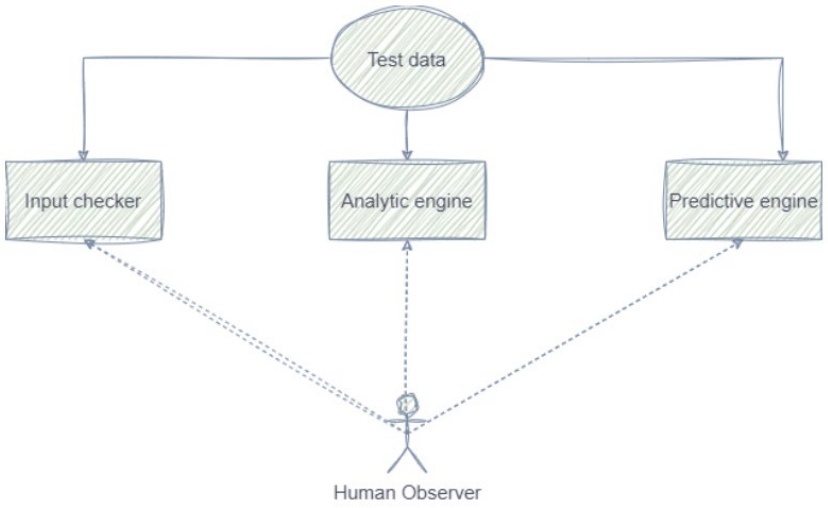
Overview of false negatives from 45918 test points

itag | TECH360°

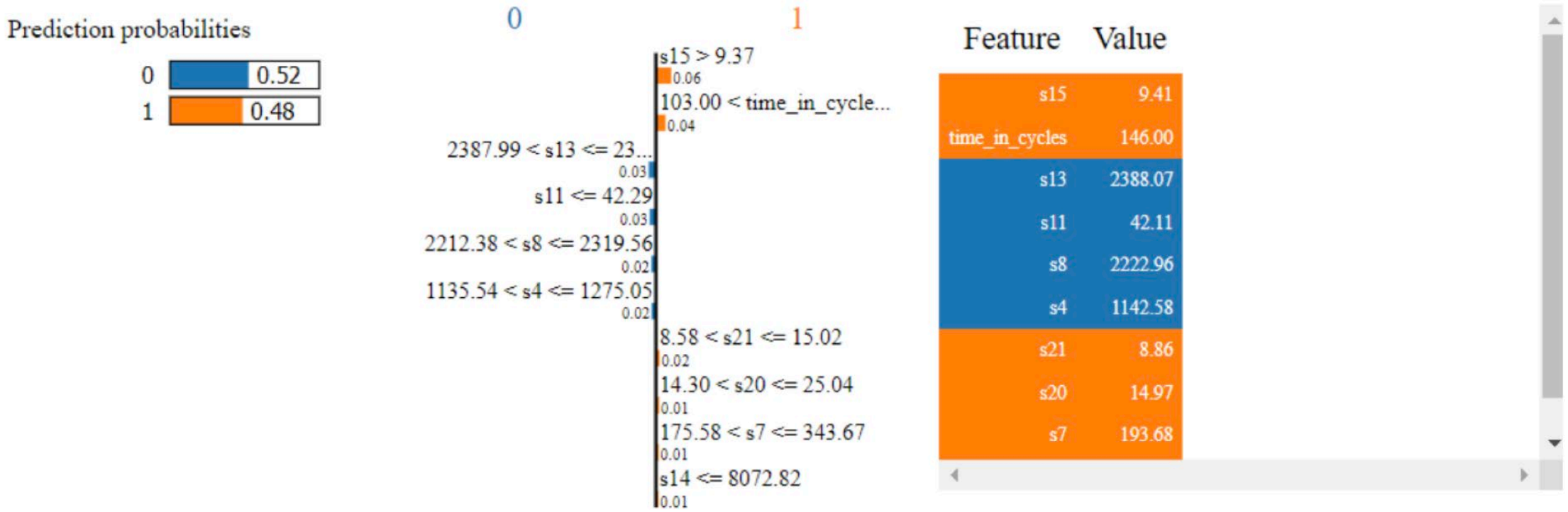
Explainable AI

Enhancing Predictive Maintenance Transparency

- Explainable AI (XAI) becomes crucial in predictive maintenance framework, where understanding model predictions and their explanations is vital
- LIME (Local Interpretable Model-Agnostic Explanations) can be employed to shed light on prediction probabilities made by each model when there is no majority consensus among them.
- Figures in following slide visually demonstrate the explanations provided by LIME for various data points, helping human operators gain insights into why certain predictions were made.
- The explanations often highlight the importance of specific features, such as s15 and time_in_cycles, which consistently emerge as critical factors in determining engine health.
- These explanations not only aid in decision-making but also offer transparency and trust in AI-driven predictive maintenance systems.



XAI using LIME



Conclusions and Future Work

- From the analysis done in these experiments, we find the proposed architecture greatly improves prognostic predictions
- To prevent the models from predicting false negatives, we have
  - Handled data imbalance and multicollinearity
  - Managed drift by monitoring the distribution of the data
  - Used multiple models as diverse ways of arriving at predictions, and
  - Used a voter architecture with LIME to produce dependable and correct predictions
    - Although we could not nullify the false negatives, voting not only reduces the number of false negatives but also provides explanations to the human in the loop
- Future work would investigate additional techniques to improve the input checker, analytic engine and predictive engine