



COMP SCI 7015 - Software Engineering & Project

Sprint Retrospective 1

a1882259

Nilangi Maheesha Sithumini Edirisinghe

Group IF_PG1

1. Snapshots (Group)

1.1 Product Backlog and Task Board

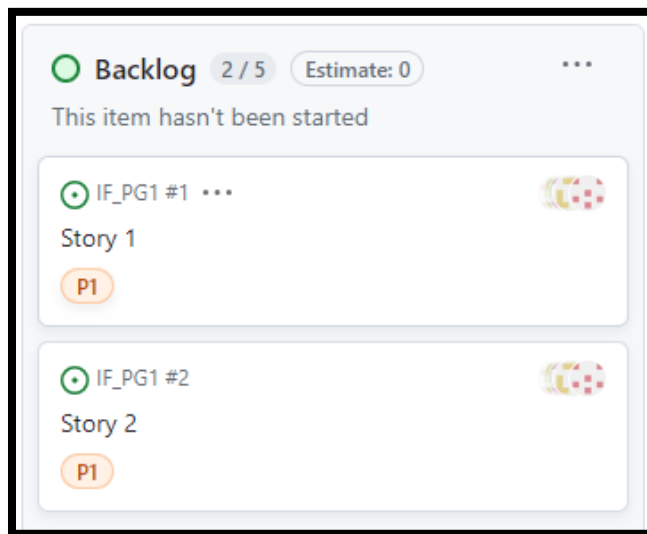


Figure 1. The Screenshot of the Product Backlog

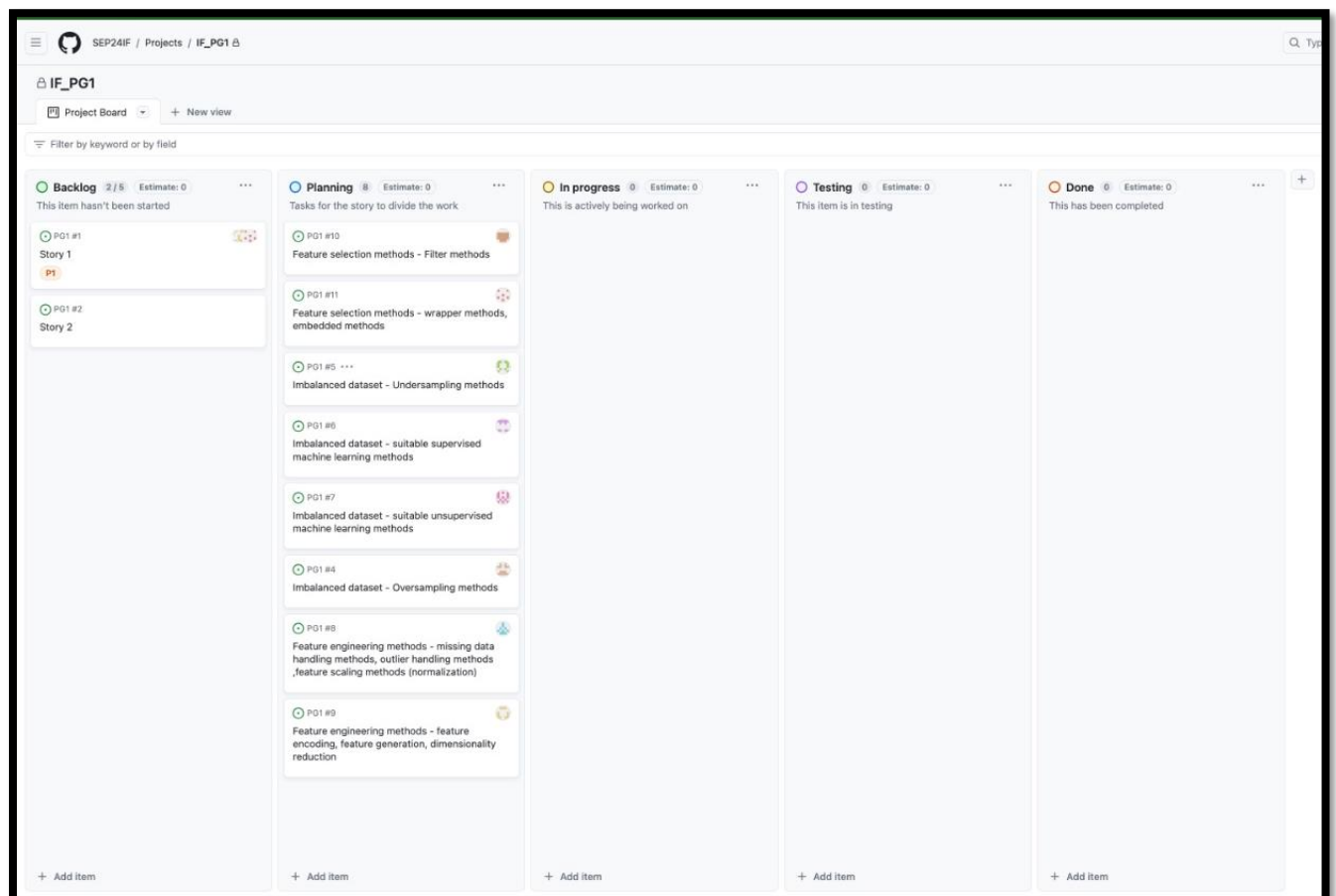


Figure 2. The Screenshot of the Task Board

1.2 Sprint Backlog and User Stories

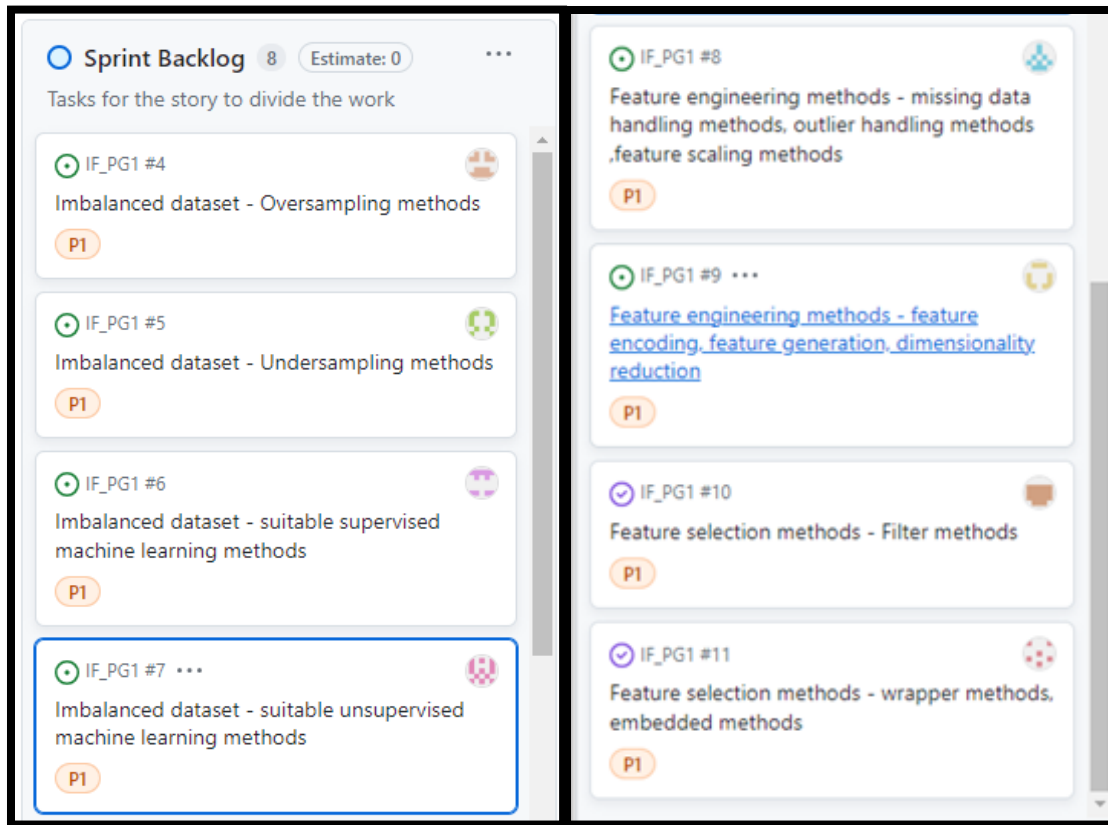


Figure 3. The Screenshot of the Sprint Backlog

User Story 1

As a software engineer, I want to research :

- 1) feature engineering methods
- 2) feature selection methods
- 3) Machine Learning (ML) techniques to approach a problem having an imbalanced dataset

Acceptance criteria :

- 1) Identify and explore at least 5 techniques each for:
 - Feature engineering methods
 - Feature selection methods
 - ML techniques to approach a problem with an imbalanced dataset.
- 2) Report your findings.

1.3 Definition of Done (DOD)

Due to the nature of the first sprint, which does not require committing code, the following DOD applies once we gain access to the platform. While conducting research and awaiting access to the InsightFactory platform, we have formulated our DOD to ensure we can work efficiently once we gain access and begin developing our machine-learning models. The DOD will be adjusted later if necessary.

Code Standards

- 1) Readability and Maintainability:
 - Code must be written with clarity and follow established style guides (e.g., PEP 8 for Python).
 - All functions and classes should have clear, concise docstrings explaining their purpose, parameters, and return values.
- 2) Testing:
 - Unit tests must cover critical functions, especially for data preprocessing, feature engineering, and model evaluation.
 - The code should pass all tests before being merged into the main branch.

Documentation

- 1) Reproducibility:
 - All scripts and notebooks must be structured to allow easy reruns by other team members.
 - Include a README file that details how to set up the environment, run the analysis, and where to find each component of the project.
- 2) Data Pipeline Documentation:
 - The entire data pipeline from raw data ingestion to final model output should be documented.
 - Include details on data cleaning steps, feature selection/engineering process, and the reasoning behind selecting specific machine learning models and hyperparameters.
- 3) Balanced Data Handling:
 - Clearly, document the rebalancing techniques used (e.g., SMOTE, undersampling), along with the rationale for choosing them.
 - Include performance metrics before and after rebalancing to illustrate the impact of the techniques.

Machine Learning Model

1) Model Selection and Validation:

- All selected models must be validated using appropriate techniques, including cross-validation and/or out-of-sample testing.
- Performance metrics must be calculated for each model and compared against baseline models (e.g., logistic regression).

2) Hyperparameter Tuning:

- Document the hyperparameter tuning process, including which parameters were tuned, the ranges tested, and the final selected values.

3) Model Interpretability:

- If possible, include model interpretability steps (e.g., feature importance, SHAP values) to provide insights into how the model makes predictions.

Review and Sign-Off

1) Peer Review:

- All code and documentation must undergo peer review before being marked as complete.
- Reviewers must verify that all DoD criteria have been met.

2) Sign-Off:

- The team must collectively sign off on each completed user story, ensuring all acceptance criteria are met and all necessary documentation is provided.

1.4 Summary of Changes

As this is the first snapshot of the sprint, this section is not applicable to this report.

2. What went well in the sprint? (Individual)

In the first sprint, our primary focus was on researching machine learning approaches for imbalanced datasets, feature engineering methods, and feature selection techniques. Our group leader effectively broke down the user story into manageable tasks and delegated them to each group member. My specific task was to investigate filtering methods for feature selection, including Mutual Information, Correlation Coefficient, Chi-Square Test, and Variance Threshold. I meticulously conducted this research, committed the results to GitHub, and presented my findings during the sprint meeting with the product-owner.

The group collectively presented our findings, which received positive feedback from the product owner. The research I conducted identified key feature selection techniques that are suitable for our project, which will serve as a strong foundation moving forward. Overall, the sprint was successful, meeting all user story requirements.

3. What could be improved? (Individual)

During the presentation, one group member presented unsupervised machine learning methods, which the product owner pointed out were irrelevant since the data was labeled, making only supervised methods applicable. This misunderstanding led to challenges in explaining the relevance of unsupervised methods during the presentation. The group leader later acknowledged the error in assigning this task. This incident highlighted the need for better communication within the group to ensure that all tasks are aligned with the project's objectives. Moving forward, we must improve our internal communication to avoid such mistakes in future sprints.

4. What Will the Group Commit to Improve in the Next Sprint? (Individual)

In the next sprint, the group is committed to transitioning from theoretical research to practical implementation. Based on the feedback received, our focus will be on actively trying out the techniques we have researched, particularly in feature engineering, feature selection, and machine learning approaches for imbalanced datasets. We understand that simply identifying suitable methods is not sufficient; it is essential to validate these methods through hands-on experimentation.

Thus, in the upcoming sprint, we will implement and evaluate at least three techniques in each category. By doing so, we aim to gain empirical insights into what works best for our specific problem context. Our goal is to ensure that any conclusions we draw are based on real-world data and performance outcomes, ultimately allowing us to produce a robust model for submission to the InsightFactory leaderboard.

5. Comment on Your Progress This Sprint (Individual)

During this sprint, I focused on researching filtering methods for feature selection, specifically investigating Mutual Information, Correlation Coefficient, Chi-Square Test, and Variance Threshold. My research was documented and shared via GitHub and presented to the product owner and the group. These techniques are vital for our project, as they will allow us to select the most relevant features for our machine learning models. The slides I prepared were well-received, and my contribution was integral in laying the groundwork for our feature selection process. The slides I prepared are as follows:

Feature Selection – Filter Methods

Mutual Information :

- Measures the dependency between two variables, specifically the input feature and the target variable.
- It provides a quantitative measure of the amount of information obtained about one random variable through the other.
- Unlike traditional correlation metrics, MI can capture non-linear relationships.
- In feature selection, it ranks features by their MI scores with the target variable. Features with higher MI scores are considered more informative.

Intuition Behind Mutual Information

- High MI Value:** Feature provides significant information about the target variable, suggesting a strong dependency between them.
- Low MI Value:** Feature does not provide much information about the target variable, indicating weak or no dependency.

Advantages of Mutual Information

- Captures Non-linear Relationships:** MI is not limited to linear dependencies and can capture more complex relationships between features and the target.
- Flexible:** It can be applied to both discrete and continuous variables (although handling continuous variables might require discretization).
- Interpretability:** MI provides a clear measure of how much a feature contributes to reducing uncertainty about the target variable.

$$\text{Mutual Information} = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \left[\frac{p(x, y)}{p(x)p(y)} \right]$$

Feature Selection – Filter Methods


What is a ... ?

Correlation Coefficient

A correlation coefficient is a number that is used to describe the strength of a relationship between two variables.

These numbers range from -1 to +1, with zero describing no correlation at all. If two variables have a correlation coefficient of 1, they have a perfectly linear correlation.

That means when one goes up, the other will go up in the same proportion, and when one goes down, so will the other.



- Good for Continuous Data:** Since our project likely involves continuous sensor data, the correlation coefficient is appropriate and effective.
- Not Ideal for Categorical Data:** If we have categorical data (e.g., wagon types), correlation coefficients won't be as useful. In such cases, other methods like Chi-Square tests or Mutual Information might be more appropriate.

Considerations:

- Only Linear Relationships:** Correlation coefficients primarily measure linear relationships. If your data has non-linear relationships, it might not capture all relevant patterns.
- Continuous Data Focus:** It's particularly well-suited for continuous features, which aligns well with your sensor data inputs.

Figure 4: My Contribution - Slide 1 and Slide 2

Feature Selection – Filter Methods

Chi Square Test for Categorical Data :

- The Chi-Square test is a statistical test used to determine if there is a significant association between two categorical variables. If our data includes categorical variables (e.g., discrete events captured by sensors), the Chi-Square test is effective for selecting features that are statistically significant in relation to the target variable. It works by assessing whether the observed frequency of rail breaks differs significantly across different levels of the categorical features.

To detect an association using the Chi-Square test:

- Calculate the Chi-Square statistic:** Compare the observed frequencies with the expected frequencies (assuming no association).
 - Compare to the critical value or p-value:**
 - Critical Value:** If the Chi-Square statistic is greater than the critical value (from a Chi-Square distribution table based on degrees of freedom), there's likely an association.
 - P-value:** If the p-value is less than your significance level (e.g., 0.05), reject the null hypothesis, indicating a significant association.
- In brief: If the Chi-Square statistic is high or the p-value is low, it suggests the variables are likely associated.

The Formula for Chi Square Is

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

where:

c = degrees of freedom
 O = observed value(s)
 E = expected value(s)

Advantages of Chi-Square Test for Rail Break Prediction:

- Detects Relationships:** Identifies significant associations between categorical factors (e.g., wagon type, rail conditions) and rail break occurrences.
- Simple Analysis:** Provides a straightforward way to assess which factors might be linked to rail breaks.

How to Use:

- For Discrete Data:** If we are given categorical inputs like wagon type or maintenance schedules, use the Chi-Square test directly to see if they are associated with rail breaks.
- For Continuous Data (e.g., acceleration, vibration):** Convert these into categories (e.g., low, medium, high) through binning, then apply the Chi-Square test to evaluate their impact on rail break predictions.

Figure 5: My Contribution - Slide 3 and Slide 4

Feature Selection – Filter Methods

Variance Threshold

- Variance Thresholding is a simple feature selection method that removes features with low variance, assuming that such features provide little information for predictive modeling.
- Feature selector that removes all low-variance features.
- This feature selection algorithm looks only at the features (X), not the desired outputs (y), and can thus be used for unsupervised learning.

- Good for Continuous Data:** Variance thresholding works best with continuous data, making it appropriate for our sensor data inputs.
- Less Useful for Discrete Data:** It can be used with discrete data, but its utility is more limited compared to continuous data.
- Suitable for Our Project:** Variance thresholding is suitable for our inputs (sensor data) and can help improve model performance by removing uninformative features, which are unlikely to contribute to accurately predicting rail breaks.

Advantages for Rail Break Prediction:

- Simplification:** Reduces the dataset size by eliminating features with little variability, making the model simpler and faster.
- Noise Reduction:** Helps in removing noise by filtering out features that don't change much and are unlikely to contribute to rail break predictions.

How to Use for This Project:

- Identify Low-Variance Features:** Apply variance thresholding to sensor data (e.g., acceleration, vibration) to detect features that show minimal variation across samples.
- Set a Threshold:** Choose a variance threshold value; features with variance below this threshold are discarded.
 - Example: If a sensor's readings don't vary significantly across time or different rails, it might not be informative for predicting rail breaks.
- Implementation:** In Python, we can use `VarianceThreshold` from `scikit-learn` to automate this process.

Figure 6: My Contribution - Slide 5 and Slide 6

Feature selection methods - Filter methods #10



A1898921 opened this issue 3 weeks ago · 1 comment



A1898921 (Yong Kheng Beh) commented 3 weeks ago · edited by a1873818



1. Correlation Coefficient
2. Fisher Ratio
3. ReliefF
4. MI



A1898921 changed the title ~~Feature selection methods - filter methods~~ Feature selection methods - Filter methods 3 weeks ago



A1898921 assigned Nilangi-Edirisinghe 3 weeks ago



Nilangi-Edirisinghe commented 2 weeks ago



1. Mutual Information-Based Feature Selection
 - Rationale: Mutual Information (MI) is well-suited for capturing both linear and non-linear relationships between the input features (like sensor data) and the target variable (rail break occurrence). Given the complexity of the sensor data, which may involve non-linear dependencies, MI can effectively measure the relevance of each feature.
 - Reference: Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence.
 - Summary: Use MI to filter out features with low dependency on the target variable, retaining those that contribute most significantly to predicting rail breaks.
 2. Chi-Square Test for Categorical Data
 - Rationale: If our data includes categorical variables (e.g., discrete events captured by sensors, frequency domain data), the Chi-Square test is effective for selecting features that are statistically significant in relation to the target variable. It works by assessing whether the observed frequency of rail breaks differs significantly across different levels of the categorical features.
 - Reference: Hall, M. A. (1999). Correlation-based feature selection for machine learning. Waikato University.
 - Summary: Use the Chi-Square test to filter categorical features that show a significant association with rail break occurrences, ensuring that only relevant variables are considered in the model.
 3. Variance Threshold
 - Rationale: The Variance Threshold method is straightforward but effective for filtering out features with low variability across samples, which are unlikely to contribute to the predictive power of the model. For sensor data that might include noise or features with minimal change over time, this method helps in simplifying the dataset.
 - Reference: Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. Journal of Machine Learning Research.
 - Summary: Apply a variance threshold to remove features with low variance, focusing on those that provide meaningful information about rail break risks.
 4. Correlation-Based Feature Selection (CFS)
 - Rationale: Correlation-Based Feature Selection (CFS) evaluates the correlation of features with the target and also considers the redundancy among features. This is particularly useful when dealing with sensor data where multiple features may be highly correlated with each other, as it helps in selecting a subset that is both highly relevant and non-redundant.
 - Reference: Hall, M. A. (1999). Correlation-based feature selection for machine learning. Waikato University.
 - Summary: Use CFS to identify and select features that are most correlated with rail breaks while avoiding redundant features, thereby optimizing the predictive power of the model.
- Summary
- These methods—Mutual Information, Chi-Square Test, Variance Threshold, and Correlation-Based Feature Selection—are well-suited for filtering features in our rail break prediction project. They effectively handle the types of data we will be given (e.g., sensor readings, tonnage data) and will help in narrowing down to the most informative features, improving our model's performance and interpretability.



Nilangi-Edirisinghe closed this as completed 2 weeks ago

Figure 7: Task assigned and completed on GitHub

6. Requirements Changes (Individual)

In this sprint, there were no changes to the requirements. However, based on feedback from the product owner, future requirement changes may be necessary, especially as we progress to the implementation phase. In particular, we may need to adjust our plans to include additional feature selection techniques or handle more complex data relationships. Nevertheless, sometimes we might need to consider all the features and then implement various kinds of machine learning techniques to get a better prediction if the relationship between features and target is very subtle and non-linear. I will keep a close watch on these potential needs, and adapt our strategies as required to meet any new demands.