# Analysis for categorical data in the preventive maintenance realm

Omar Eduardo Gutiérrez García (23168307)
Data Analytics for Artificial Intelligence – H9DAI
MSCAI_SEP23
School of Computing
National College of Ireland

## 1 Background Research

The maintenance industry is evolving thanks to the appearance of new technologies, the ability to predict and prevent asset breakdowns is the main concern and most valuable to companies (Yang et al. 2022). This research is motivated by the possibility of helping in this area. Machine Learning models can transform prediction practices and enable a better maintenance's scheduling and processing that reduces cost and labor, while making companies more productive by enhancing their assets performance. In this context, the research explores the application of data analysis different techniques around data portraying to the maintenance industry. Being specific, on data collected by Cynch![1], a leading software company providing ERP solutions. Who's customers perform day to day inspections to a very wide range of different types of assets, which we will analyse and use for training a new predictive model.

Our work relies on a ML model designed to accurately predict results of asset inspections by utilizing historical data of past predictions for similar assets and inspections performed by a variety of companies. This is part of an evolving trend of generating proactive strategies based on predicting assets condition utilizing past data (Yang et al. 2022). In addition, it proposes a novel application by utilizing a more available source of data to the majority of companies in the industry. Opening the ML realm to a wider range of organizations that are constantly looking to optimize asset performance and reduce its downtime.

[1] Explore Cynch! features and services at `http://cynch.me`

The mentioned novel application arises from the particularity of the data utilized in this research. Our partner has provided a database that is rich in categorical data around the inspections and it lacks numeric features. This sets it apart from most literature machine learning on predictive maintenance, since the common factor on all that literature is the reliance on sensors measurements. These sensors, which are attached to the assets, provide a very good idea of the asset status and can really help on detecting patterns on the asset's performance. Problem is, that the use of sensors is not a common practice in the industry, since it is costly to implement and maintain. Marking a division between the high level companies, like large manufacturing corporations, and low level asset users who can range from hospitals, smaller plants or even people that own a land mower. So, by utilizing this rough categorical data and being able to train a ML model with it, we can provide a greater range of users with a solution to predict the result of an inspection.

By prioritizing cost-effectiveness for this low level users we acknowledge a potential trade-off in accuracy. This could contribute to the ongoing debate about the worthiness of sensors investment, which by comparing applications in different industries like oil and electrical, evaluates the cost-benefit balance on the investment (Keartland & Van Zyl 2020).

# 2 Data Analytics

In this section, we dive into the data processing steps undertaken in our research project to "clean up" the data provided and prepare it for model training. Our first step then, is one of the most essential: to look at the raw data and select the most useful features we can work with (Wu et al. 2017). In our case this meant looking at the whole Cynch! database and craft a procedure that could query all the useful features in an orderly manner. We utilized SQL for this purpose, since it is a relational database. Subsequently, it was of major importance to ensure data privacy and comply with the organization's policies. To do that, authorization of the procedure was requested and approved by the Cynch! team via email.
It is important to note here that, having our use case in mind, the data extraction procedure contained already some restrictions that helped reduce the amount of data to a manageable size. By "pruning" we mean removing logs that had missing values on important features. The result was a extracted CSV file that could be use for data processing in our research but also in the future in case there is a need to rerun the experiments we have performed. This CSV is built only with public data.

Next step, after completing the extraction, is to perform data processing. As seen in the lectures, several steps were followed to this purpose. The selected tool was a Jupiter

Notebook with the CSV extracted as the input. Next, we describe each of those steps:

1. **Understanding our data**: Having the data ready, the next step is to understand which type of data we are working on. As mentioned before, in this case we are working with a big amount of categorical data and just one numeric feature. In the next table we can see each feature, a brief explanation of what it means and which type of data it is.

| **Feature name** | **Feature description** | **Data type** |
|---|---|---|
| Asset type | What kind of asset it is (monitor, mower, snowblower, scanner, etc.) | Categorical-nominal |
| Asset make | Who is the manufacturer of the asset (Echo, Jhon Deere, Sony, etc.) | Categorical-nominal |
| Asset model | The model for the asset (DE-400, TurboVision 20, etc.) | Categorical-nominal |
| Asset | Which specific asset we are using | Categorical-nominal |
| Industry | In which industry this type of asset is used (medical, power equipment, aeronautical, etc.) | Categorical-nominal |
| Store performer | Store who would perform the inspection | Categorical-nominal |
| Inspection type | Which type of inspection will be performed | Categorical-nominal |
| Inspector | Worker who would perform the inspection | Categorical-nominal |
| Asset owner | Which company/person own the asset | Categorical-nominal |
| Original checklist | To which checklist the inspection portraits to | Categorical-nominal |
| Original inspection | Which is the original inspection (a way to connect inspections from different stores which are equal) | Categorical-nominal |
| Last performed inspector | Worker who performed the last inspection for the asset | Categorical-nominal |
| Last performed inspection result | What was the result on the last inspection performed for this asset (pass, fail, n/a) | Categorical-nominal |
| Days since last performed | How many days have passed since last inspection | Numeric, ratio-scaled and discrete |

2. **Missing values**: As mentioned, the quantity of samples was already reduced in the extraction. But, data still had missing values that needed evaluating, and finding that there was just a small amount of missing values, thanks to the previous pruning, the decision was to remove them. Features found to have missing values: asset type, asset make, industry and asset owner.

As a note, there was a test performed to avoid the major pruning in the extraction phase by substituting the missing values with grouped medians or grouped modes depending on the feature. But it resulted on a very large data set that was unmanageable to our available computer processing power.

3. **Categorical data**: The majority of the features extracted for a purpose is categorical, which is the major distinction of our work. That is why dealing with it is a major step for our work. For all categorical features, we have ordinal data with no logical ranking. To solve that, we found one hot encoding to be our only solution. So it was performed for 13 features that already had a large quantity of categories. This caused a significant growth of features that had to be dealt with, this is explained on our next step.

4. **Dimensionality reduction**: After dealing with categorical data we were left we a dataset that contained around 34,000 features. This made the dimensionality reduction an absolute necessity. As suggested by our professor, we aimed for, at most, 1024 features. So, a deep neural network was designed for this purpose of autoencoding. The network consist on 2 hidden layers to encode and another 2 for decoding, utilizing an activation function of "Tanh". The data set was ran into this encoder and resulted on a 0.0051 which was deemed acceptable for our case. The output for this step was then the reduced dataset.

5. **Scaling data**: Finally, after encoding, we only had one more feature to deal with - "Days since last performed"- which was the only non-categorical. The encoding had return values between -1 and 1. So, the numeric feature was scaled to the same range utilizing a "minmax" function.

6. **Data Output**: After processing the data, we had again a CSV file as an output that is clean to be used for model training. This dataset contained 1025 features (1024 encoded categorical, 1 numeric) and 1 target column.

# 3 Machine Learning Algorithms

Literature points to some useful classification models that have been successful when training for predictive maintenance related applications - Decision trees, Random forest, Neural Networks (mentioned in literature as "Black boxes"), and Logistic Regression (Benítez et al. 1997). For that reason, those are the models used on our experiment phase to evaluate their accuracy with our data. We proceed to analyze advantages and disadvantages of each one of them:

**Decision Trees Random Forest**

This 2 models are one of the same, they are recognized for having great accuracy and to be very helpful when working with data that has complex patterns. Keartland & Van Zyl (2020) mentions that, in predictive maintenance scenarios, they show to be very precise, and it gives the example of classifying the health of machines, which is the same use as in our research. From the 2 of them, Random Forest could be considered more complex but at the same time more computationally intensive. This trade-off should be considered to select a model specially when working with large datasets (Benítez et al. 1997). In opposition to other models, the option of Decision Trees offers a structured diagram with understandable paths into the model decisions. However, in our case, that is not an advantage, due to the encoding of the data.

**Neural Networks**

A model that we have already use when encoding the data. Composed by a diverse quantity of layers and hidden layers made of neurons that create a very structured model. The Neural Networks offer a powerful option in AI to solve complex problems and capture intricate patterns in data (Benítez et al. 1997). Training the network can be very computationally demanded, but once trained, it shows great speed when performing tasks (Benítez et al. 1997), in this case, predictions. It is a great option for predictive maintenance because as the previous models, it excels when solving for complex and non-linear patterns.

The known disadvantage for this models is its denomination as ”black-box”, making it very hard to understand how it works (Benítez et al. 1997). Nevertheless, as we mentioned before, this already happens for our use case since using the encoding neural network.

**Logistic Regression**

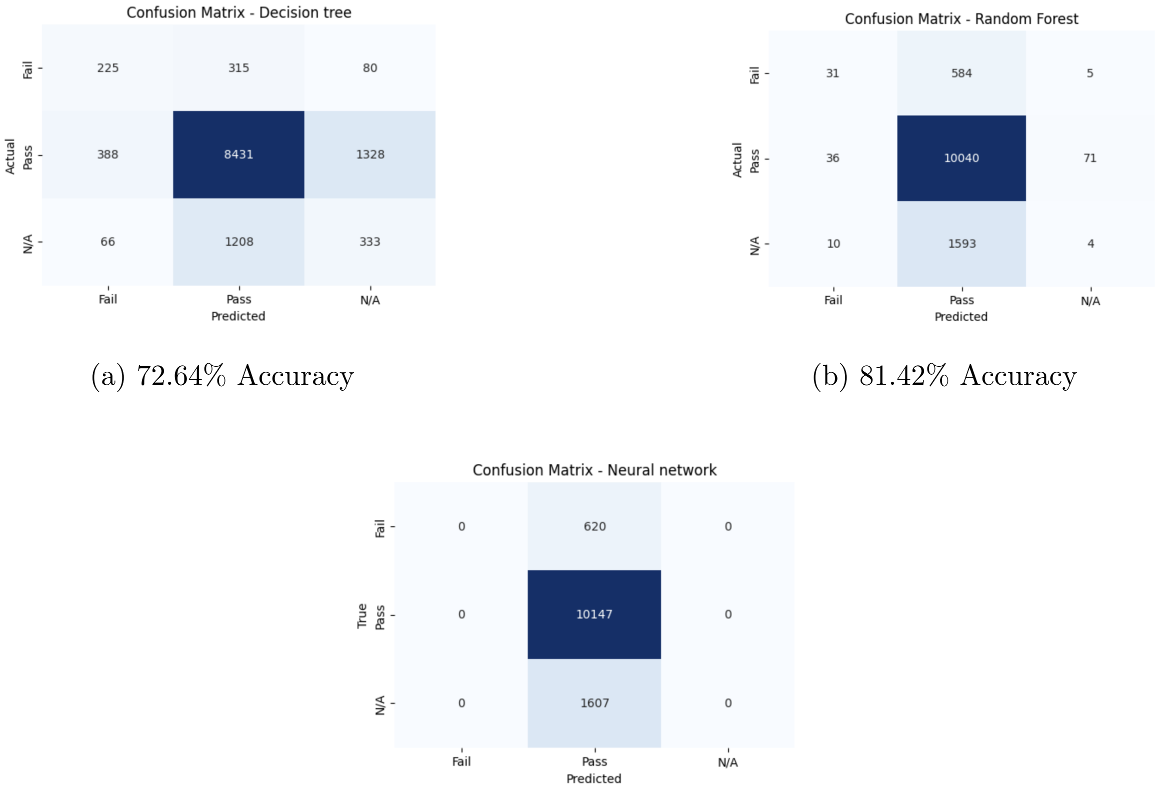
Finally we encounter a binary model, that is simple enough to normally display relationships between inputs and outputs (Keartland & Van Zyl 2020). And this simplicity also translates into computational efficiency which helps reduce the resources requirements. Its disadvantage relies in assuming linear relationships along the variables, meaning that the model generated can fail to properly represent complex patterns (Benítez et al. 1997)(Biau 2012) something that is to be expected in preventive maintenance. Another specific disadvantage for our use case is that this model only works with binary results, making us have to discard a classification in our target data to be able to work with it.

# 4 Evaluation and Discussion

In this section, evaluation is performed through a series of experiments where we tested the ML models mentioned on the past section, to find out which is the best model and also which is the best data configuration for our dataset.

## 4.1 Experiment 1

First experiment uses the data configuration mentioned on the data analytics section. Non other configurations added, this is our starting point to test the models and try to improve the data as we jump into the next experiments to make more accurate models. For this case all ML models were tested with the exception of Logistic Regression, which can only handle binary classification and this is not the starting case.
The general result from this experiment is that models performed ”good” in regards of the accuracy. Nevertheless, when looking at the confusion matrix, it was evident that this only happened because of the data distribution. In Figure 1, we can observe that the accuracy on the minority classes (Fail and N/A) was very low.



(a) 72.64% Accuracy

(b) 81.42% Accuracy

(c) 82.00% Accuracy

Figure 1: Experiment 1 results

Our network is conformed by the ”ReLu” function in 3 hidden layers and ”softmax” for the output layer. For the loss function we selected ”sparce categorical crossentropy” since it is the best suited for multicategorical models. The results are then run through a common ”argmax” function. In this experiment, we tested a diverse distribution of weights on the argmax with the purpose of having some minor classes predictions. But the results were dissapointing, in each case, when minor classes predictions appear, overall accuracy diminished.

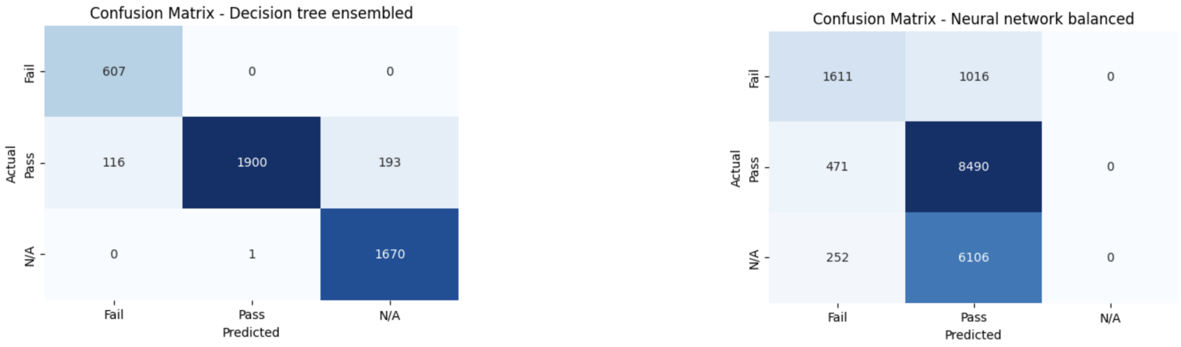
The models failed to predict for minority classes, to the point that, for neural networks, no predictions were made of the minor classes. We can conclude from this experiment that data is suffering from class imbalance, an important consideration in the data analytics realm.

## 4.2 Experiment 2

This is the class imbalance experiment, where modifying the data setting to solve for imbalance was the priority. Our distribution of classes was like this:

- Pass: 50,653 samples
- Fail: 3,169 samples
- N/A: 8,047 samples

With this in mind, we proceed to select a method for solving the issue. In our case, data set was divided into balanced subsets that contained copies of the minority class to balance the majority class (Abd Elrahman & Abraham 2013). A random subset of the majority class was used in each of the new 4 subsets with all the samples from the 2 minority classes. Having that better distribution, models were tested with each subset (ensemble learning). This update not only improved the per class accuracy, in addition, general accuracy boost to 93% as shown in figure Figure 2. Random Forest is already an ensemble learning model so it was not considered for this experiment.



(a) 93.09% Accuracy

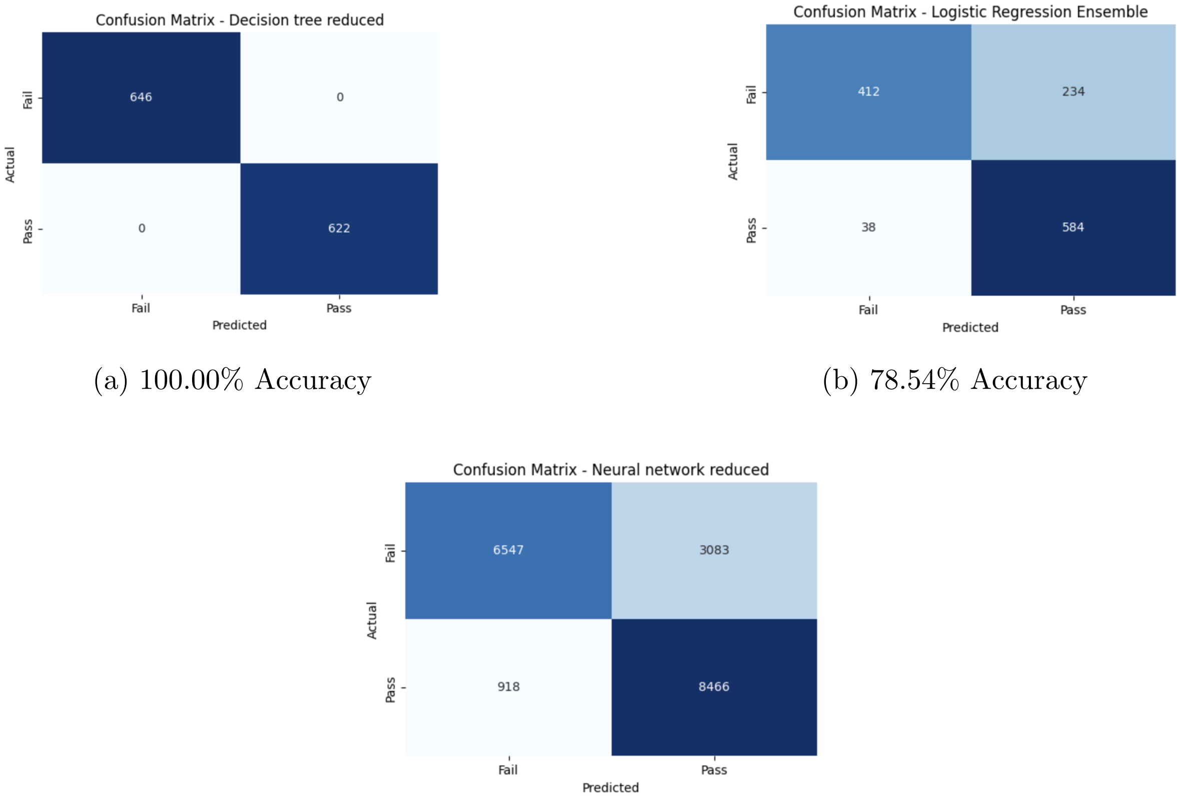
(b) 56.28% Accuracy

Figure 2: Experiment 2 results

In the results we can see a better performing Decision Tree. Neural networks also improved their per class accuracy, but it still lacked predictions for the "N/A" class, which is not ideal but can lead us to the next experiment.

## 4.3 Experiment 3

The final data setting selected is to consider only 2 classes while maintaining the setting for class imbalance and the ensemble learning. This allows us to utilize the Logistic Regression model and try to improve the modeling of the neural networks which failed to predict for "N/A", so we got rid of that class. Results for each model are showed in figure Figure 3.



(a) 100.00% Accuracy

(b) 78.54% Accuracy

(c) 78.96% Accuracy

Figure 3: Experiment 3 results

As expected, the setting helped the neural network model to improve drastically. In addition, it caused a great improvement for the Decision Tree with a 100% accuracy. Logistic regression model showed a very similar result as the Neural Network, still fell significantly short when bench marked against the results of the Decision Tree in this experiment and the last. However, removing the N/A class made the balanced subset increase in quantity from 4 on the last experiment to 15 in this one, due to the bigger imbalance between Pass and Fail classes. Making the subsets smaller in samples. This is why we were able to get a 100% accuracy, but we can consider this misleading due to the reduced quantity of test samples compared to the other experiments.

# 5 Conclusion

Our selected dataset for this project has lead us to drive through the intricacies that appear when dealing with a categorical-nominal rich features. In addition, by being an example extracted from a real company database, it required a wider range of data analytics tools to be applied. This made the research harder to execute, meaning more learning thanks to the greater challenge.
From all the methodologies applied we found that the most complex was dimensionality reduction by far, it required a lot of testing to create a working encoder. This was done with a deep neuronal network that is more computational intensive than the trained models. Another important methodology was dealing with class imbalance, as it revealed itself as a necessity after testing our first models. Without it the models would have been completely inaccurate and because of that, useless. The rest of the methodologies, even while being less complex to execute, were proved as important in our process.

On another topic, the creation of models was not the main focus of this research. Nevertheless, after testing several models with a variety of data setting, we have found a working model with Decision Trees that showed over 93% accuracy (a better result than the 3rd experiment with 100% accuracy but reduced samples). This translates into a great result for our partner Cynch!, since they can take this model as a starting point to build on a better and more robust model that can help their customers.

There is a lot that could be done as future work for this research that could help our partner build a a better dataset and with that better models. A main contribution would be the collection of more data that can be pivotal when trying to predict the result of an inspection like the manufacture date of the asset or the quantity of maintenance performed previously in a particular asset. Apart from collecting data, with more computational power available, more settings could be tested for the encoding network to reduce it loss and more samples of data could be use when training if replacing missing values instead of dropping the whole samples.

# References

Abd Elrahman, S. M. & Abraham, A. (2013), 'A review of class imbalance problem', *Journal of Network and Innovative Computing* **1**(2013), 332–340.

Benítez, J. M., Castro, J. L. & Requena, I. (1997), 'Are artificial neural networks black boxes?', *IEEE Transactions on neural networks* **8**(5), 1156–1164.

Biau, G. (2012), 'Analysis of a random forests model', *The Journal of Machine Learning Research* **13**, 1063–1095.

Keartland, S. & Van Zyl, T. L. (2020), Automating predictive maintenance using oil analysis and machine learning, *in* '2020 International SAUPEC/RobMech/PRASA Conference', IEEE, pp. 1–6.

Wu, D., Jennings, C., Terpenny, J., Gao, R. X. & Kumara, S. (2017), 'A comparative study on machine learning algorithms for smart manufacturing: tool wear prediction using random forests', *Journal of Manufacturing Science and Engineering* **139**(7), 071018.

Yang, Z., Baraldi, P. & Zio, E. (2022), 'A method for fault detection in multi-component systems based on sparse autoencoder-based deep neural networks', *Reliability Engineering & System Safety* **220**, 108278.