SYSTEM FAILURE SCORE PREDICTION

CSCI6505 - Machine Learning Term Project Scientific Paper

Burhanuddin Rangwala - B00926210 (Faculty of Computer Science)
Mufaddal Lokhandwala - B00914329 (Faculty of Industrial Engineering)

***Abstract*—This project aims to predict a system's failure score for each week of the following year. The dataset comprises weekly sensor values spread over 4 years. Analyzing the dataset, the prediction was deduced to be a regression problem. Regression Algorithms such as Linear Regression, Decision Tree Regressor, Random Forest Regressor, and SVM Regressor were used. All the models were compared using significance tests and evaluated using the mean absolute error. The evaluations pointed toward Random Forest Regressor to be a promising candidate for the problem.**

I. Introduction

FOR industries that depend on complex systems, having the ability to foresee system breakdowns can be essential. Early system failure detection and prediction helps save costly downtime, avert system failures, and maintain high levels of productivity. This project aims to predict a system's failure score for each week of the following year using machine learning algorithms.

We studied the relation between the sensors and the failure score and decided to perform feature selection as certain sensors were not influencing the failure score. Furthermore, we performed data pre-processing steps to remove any inconsistency in the data, thus, making the data more reliable for better predictions [6].

After performing feature selection, we normalized the data so that all the feature values are spread on a similar scale which improves the performance and stability of the model. After the data is ready, we feed it to training algorithms, and subsequently, predictions are made which are evaluated using mean absolute error.

Now, for deciding the superiority of the models we perform paired_ttest2x5cv [3] significance testing among the models to select the most suitable model for our prediction. The results of significance testing and mean absolute error showed that the Random Forest Regressor gives the best fit on the data with a mean absolute error of 13.74.

II. METHODOLOGY

This section provides information about data preprocessing, training algorithms, and evaluation metrics adopted to solve the problem statement as mentioned above.

A. Data Pre-Processing

It is one of the crucial steps in machine learning [6]. The performance of a model is highly influenced by the quality of the dataset and the information that can be retrieved from the given dataset.

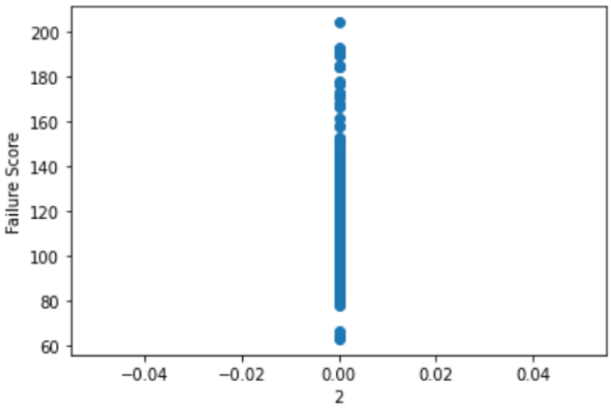

Fig. 1: Sensor 2 relationship with target variable

Initially, we normalized the sensor values to bring all the values between a similar scale, as values are distributed between -0.4 to 20, as shown in one of the distribution plots in Figure 2. It also comprises fractional as well as integer values, which can affect the performance of the model, thus, bringing all the values to the same scale increases the stability of the model.

For feature extraction we analyzed the data by plotting each sensor value with the failure score to understand the impact of sensor values on failure score, and made the following conclusions:

- Sensor values for 2 and 3 are all zeros as shown in Figure 1 irrespective of the failure score, thus, removing them won't have much impact on the prediction of the failure score.
- Sensor values for 4, 5, 11, 12, 42, and 62 have most of their readings as 0 as shown in Figure 3 (for 42, the rest are similar), thus dropping them as well.

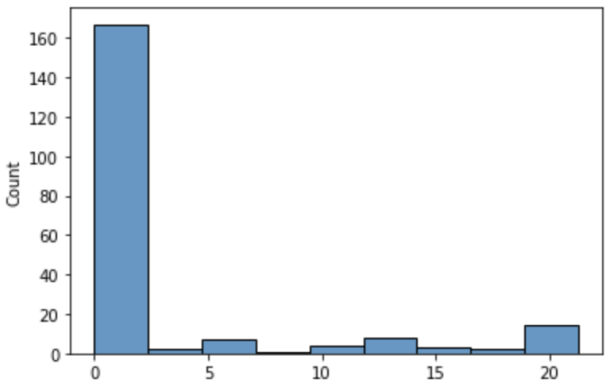

Fig. 2: A sensor value distribution between 0 to 20

B. Training Algorithms

Studying the graphs, we concluded that the problem is a regression problem therefore we decided to use the following 4 regression models and evaluated their performance using mean absolute error: -

1) **Linear Regression:** This algorithm tries to fit a linear line between the independent features and target variables. The equation resembles that of a line where the training algorithm tries to find the slope and the intercept that best fits the data. The equation can be represented as:

$$Y_i = f(X_i, \beta) + e_i \tag{1}$$

2) **Decision Tree Regressor:** A DecisionTreeRegressor is used to predict continuous values instead of discreet values. For regression instead of calculating entropy or information gain, we need a measure that tells us how much deviated our predictions[5] are from the original dataset, which calls for mean squared error as splitting criteria. The dataset is split such that each split minimizes the mean squared error and the tree is formed. This is a good fit here because it is handling non-linearity while creating splits and it also takes into account correlation between features.

3) **Random Forest Regressor:** This regressor consists of multiple decision tree regressors[7]. The final predicted value is the average of the predicted values obtained from each decision tree. This is a good fit as it has all the advantages of Decision Tree Regressor and additionally it is robust to overfitting because of using aggregated result of mutiple DTs.

4) **SVM Regressor:** SVR [8] gives us the flexibility to define how much error is acceptable in our model and will find an appropriate line (or hyperplane in higher dimensions) to fit the data. This is a good fit because it works well with linear and non-linear datasets (as shown in Figure 4), and small datasets.

C. Evaluation Criteria

Mean Absolute Error: It is a metric used in regression analysis that calculates the mean of the absolute difference between the predicted and actual values. It is used in this project for evaluating the models. It can be represented mathematically as:

$$MAE = 1/n * \sum_{(i=1)}^{n} |y_i - \hat{y_l}| \tag{2}$$

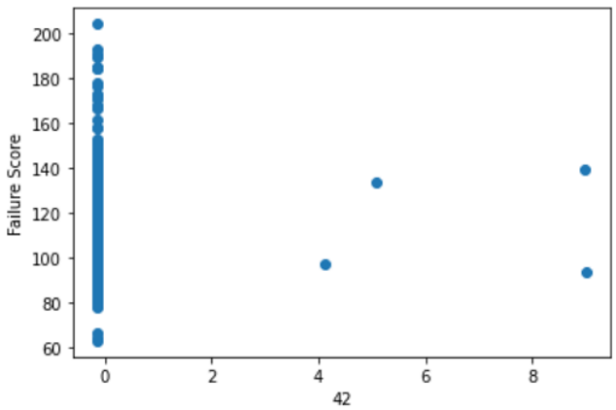

Fig. 3: Sensor 42 relationship with target variable

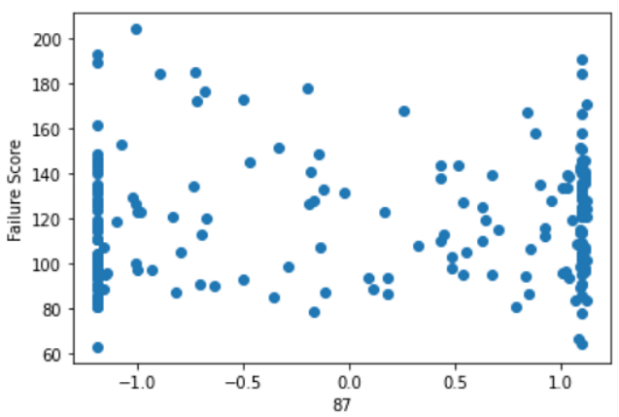

Fig. 4: Sensor 87 relationship with target variable

III. Results

The performance of the models is evaluated on the test dataset which is split from the training dataset. The split is standard 80-20 split. Mean absolute error is calculated by conducting 5 experiments as tabulated in Table 1.

Now, just comparing mean absolute errors and judging the best model from these values is not enough, as we cannot judge whether the difference between the errors is real or statistical fluke[2]. We need to perform significance testing among the models to conclude which model is better than others.

Significance Testing:
Significance tests[2], works on the principle of likelihood of samples of skill score of the models given the assumption that they were drawn from the same distribution. If this hypothesis (null hypothesis) is rejected, it means that the performance of the models are statistically different.

Here we are using the widely adopted 5x2 cross-validation paired with a modified student t-test[3], which mitigates the problem faced by other statistical measures whose methodologies violate the assumption of student t-test.

Applying this method to pairs of models, it gives a p-value and a statistical value. This p-value is compared with a reference value known as alpha. If the value is greater than alpha, we reject the null hypothesis and deduce that the performance of the models are statistically different, otherwise, the models are statistically same, thus, adopting either of them would solve the problem.

We consider the value of alpha to be 0.05 according to [3]. To overcome the dilemma of selecting a better model, after performing significance testing we consider model which has minimum mean absolute error.

MODEL NAME	MAE
Linear Regression (LR)	24.6
Decision Tree Regressor (DTR)	18.22
Random Forest Regressor (RFR)	13.74
SVM Regressor (SVR)	21.65

TABLE I: Mean Absolute Errors of each model obtained from 5 experiments

MODEL 1	MODEL 2	p	t	Null Hypothesis
LR	RFR	0.15	-1.67	Rejected
LR	DTR	0.025	-3.16	Accepted
LR	SVR	0.92	-0.10	Rejected
RFR	DTR	0.26	1.25	Rejected
RFR	SVR	0.001	-6.47	Accepted
DTR	SVR	0.58	-0.59	Rejected

TABLE II: Significance testing of all the combinations of the models using Null Hypothesis

From the results in Table 2, it is evident that Linear Regression and Decision Trees have indeed statistically similar performance as the null hypothesis holds because the p-value is less than alpha. Similarly, Random Forest and SVR are also similar. On the other hand, null hypothesis between Random Forest and Linear Regression is rejected implying that they are statistically different.

Using significance testing we have deduced some models to be similar and some to be different. For choosing the best among similar models we compare their Mean absolute error. From Table 1 we can say that Random Forest Regressor is the optimal choice for failure score prediction of the sensor system with minimum absolute error of 13.74.

IV. CONCLUSIONS

It is evident from the results that the random forest was the most efficient model among all the models for predicting the failure score. This is because random forest can handle non linearity in the data, which is bane for SVM and Linear regression models as they work best when the features and target variable have a linear relationship.

Random Forest is better than decision tree[1] because decision trees are prone to overfitting whereas random forest mitigates this issue by taking the result from multiple decision trees which introduces robustness and prevents overfitting.

V. REFERENCES

[1] Decision Tree vs Random Forest Algorithm. Analytics Vidhya web site: https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/

[2] Statistical Significance Tests For Comparing Machine Learning. Machine Learning Mastery web site: https://machinelearningmastery.com/statistical-significance-tests-for-comparing-machine-learning-algorithms/# ˜ text=Statistical %20hypothesis%20tests %20can%20aid,can%20lead%20to%20misleading%20results.

[3] paired ttest kfold cv Kfold cross-validated paired t test. web site: http://rasbt.github.io/mlxtend/user_guide/evaluate/paired_ttest_kfold_cv/

[4] Regression Trees Decision Trees For Regression. Medium web site: https://medium.com/analytics-vidhya/regression-trees-decision-tree-for-regression-machine-learning-e4d7525d8047

[5] Splitting of Dataset in Decision Tree Regressor web site: https://www.saedsayad.com/decision_tree_reg.htm

[6] Introduction to data pre-processing in machine learning. web site: https://towardsdatascience.com/introduction-to-data-preprocessing-in-machine-learning-a9fa83a5dc9d

[7] Random Forest Regression. Medium. web site: https://towardsdatascience.com/random-forest-regression-5f605132d19d

[8] Support Vector Regression. Analytics Vidhya. web site: https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/