**Abstract:**

In this project, I have developed four different models, including a neural network, an auto neural network, a regression model, and a decision tree, to predict whether the use of the flying probe test (FPT) is required for a circuit board to be tested. The models were developed using SAS Enterprise Miner and compared using various criteria such as Valid average squared error, Train Average Squared error, and Train misclassification rate. Contrary to initial expectations, the results indicate that the auto neural network model does not exhibit superior performance in terms of accuracy rate and error rates compared to the other models. However, further investigation is necessary to understand the underlying reasons for this outcome. Nonetheless, the developed models still hold the potential to enhance the efficiency of the testing process by minimizing the requirement for unnecessary FPTs. This optimization can lead to significant time and resource savings for the organization.

**Introduction:**

A picture containing bar chart

Description automatically generatedElectronic manufacturing has been increasingly important in the modern world, and the testing process plays a critical role in ensuring the quality and reliability of the products. Among various testing methods, the flying probe test (FPT) is a widely used and effective method for testing printed circuit boards (PCBs). However, the use of FPT can be time-consuming and expensive, especially when it is not necessary. Therefore, it is important to develop models that can predict whether the use of FPT is required for a board to be tested or not. In this project, we aim to develop and compare different models to predict the necessity of FPT for a board. We have developed four models using SAS Enterprise Miner, including a neural network, an auto neural network, a regression model, and a decision tree (and maximal decision tree). The models are trained using a dataset consisting of various features related to the boards, such as the number of layers, the number of components, and the size of the board. The variables considered worthful based on Chi Square Ratio are depicted in ***GRAPH1***. The output of the models is a binary classification indicating whether FPT is required or not.

***(Graph1: Graph Shows worth of the top 5 variables that contribute to the growth of the model)***

The performance of the developed models is evaluated using several criteria, including Akaike's Information Criterion, Schwarz's Bayesian Criterion, average squared error, maximum absolute error, and misclassification rate. The results indicate that the auto neural network model surpasses the other models, exhibiting the highest accuracy rate and the lowest error rates. The developed models can help to improve the efficiency of the testing process by reducing the number of unnecessary FPTs, which can save time and resources for the organization. Overall, this project demonstrates the effectiveness of machine learning models in predicting the necessity of FPT for a board to be tested. The results can have practical implications for electronic manufacturing companies, where the use of efficient testing methods can lead to improved product quality and reduced costs. The rest of the report will provide a detailed description of the methodology, results, and conclusions of this project.

**Models:**

**Decision tree:**

A decision tree model begins with a root node that encompasses the entire dataset. It then partitions the data based on various attributes or features, generating branches and sub-nodes. Each internal node represents a test or condition on a specific attribute, and each branch represents the possible outcomes of that test. This recursive process continues until a stopping criterion is satisfied, typically determined by factors such as maximum tree depth or minimum number of observations. In this project, the decision tree model is applied to predict the suitability of using FPT for a board during testing, leveraging its capacity to capture non-linear relationships between the input variables and the output. This characteristic renders it a valuable tool for this task.

Word, timeline

Description automatically generated

***(Fig 1: Maximal Decision tree using interactive option with 4 surrogate rules)***

Graphical user interface, application, Word, timeline

Description automatically generated***(Fig 2: Decision tree node result)***

Chart, line chart

Description automatically generated**Decision tree performance:**

Chart, line chart

Description automatically generated***(Fig 3: Model Score TRAIN Decision Tree)***

Chart, line chart

Description automatically generated***(Fig 4: Model Score VALIDATE Decision Tree)***

***(Fig 5: Model Score TEST Decision Tree)***

**Implementation of Decision Tree:**

**Import the Data:**

Import your dataset into SAS Enterprise Miner by accessing the "Data Sources" tab and choosing the "Import Data" option. Select the file that contains your dataset and follow the import wizard's instructions.

**Create the Decision Tree Diagram:**

To create a new Decision Tree diagram, navigate to the "Diagrams" tab and choose "Create Diagram". From the list of diagram types, select "Decision Trees" and provide a name for your diagram.

**Define the Target Variable:**

In the "Diagram Workspace," choose your dataset from the "Data Sources" tab. Select the target variable by right-clicking on it and choosing "Set as Target."

**Select Input Variables:**

Choose the input variables for predicting the target variable by dragging and dropping the relevant columns from the "Data Sources" tab to the "Diagram Workspace". Alternatively, you can utilize the "Variable Selection" node to automatically determine the most important input variables.

**Build the Decision Tree Model:**

Build the decision tree model by dragging and dropping the "Decision Tree" node from the "Modeling Palette" onto the "Diagram Workspace". Connect the input variables and target variable to the decision tree node**.**

**Set** **the Decision Tree Properties**:

Open the properties dialog of the decision tree node by double-clicking on it. Adjust the options for the decision tree, such as maximum depth, minimum observations in a leaf node, and splitting criterion.

**Run the Decision Tree:**

Click "Run" to execute the decision tree model using the selected input and target variables in SAS Enterprise Miner**.**

**Analyze the Decision Tree Results:**

Analyze the decision tree model by reviewing the tree diagram and relevant statistics. Utilize the "Score" node to apply the model to fresh data.

**A picture containing table

Description automatically generatedDecision Tree Results:**

**Graphical user interface, application

Description automatically generated*(Fig 6: Decision Tree Variable importance)***

***(Fig 7: Decision Tree Fit Statistic Result)***

We will discuss these results in result topic further down this report together with other models.

**Auto neural network:**

The auto neural network is an enhanced version of the conventional neural network model that incorporates a layer of hidden units to automatically extract essential features from the input data. This feature extraction process aids in reducing the dimensionality of the input space and enhancing the model's capability to generalize to new data. In this project, the auto neural network model is employed to predict the suitability of using FPT for a board during testing. Its inherent ability to autonomously learn valuable features from the input data contributes to enhancing the accuracy of the model.

**Chart, line chart

Description automatically generatedAuto neural network performance:**

***(Fig 8: Model Score TRAIN Auto Neural Network)***

**Chart, line chart

Description automatically generated*(Fig 9: Model Score TEST Auto Neural Network)***

**Chart, line chart

Description automatically generated**

***(Fig 10: Model Score VALIDATE Auto Neural Network)***

Table

Description automatically generated**Auto neural network Results:**

***(Fig 11: Auto Neural Network Fit Statistics Result)***

**Neural network:**

The neural network model is a versatile and robust tool for capturing intricate relationships between inputs and outputs. It excels at handling vast amounts of data and possesses the capability to learn and generalize from patterns within the data. In this project, the neural network model is utilized to predict the suitability of using FPT for a board during testing, leveraging its capacity to capture complex interactions among input variables and effectively learn from the available data.

Chart, line chart

Description automatically generated**Neural network Performance:**

***(Fig 12: Model Score TEST Neural Network***Chart, line chart

Description automatically generated)

***(Fig 13: Model Score VALIDATE Neural Network)***

Chart, line chart

Description automatically generated ***(Fig 14: Model Score TRAIN Auto Neural Network)***

**Table

Description automatically generatedNeural network Results:**

***(Fig 15: Auto Neural Network Fit Statistics Result)***

**Implementation of Neural Network:**

**Import the dataset:**

Open SAS Enterprise Miner and import the dataset.

**Define the target variable:**

Specify the target variable for prediction in the neural network model, such as the delivery time and project cost in your case.

**Preprocess the data:**

Preprocess the data to address missing values, outliers, and other data quality concerns. Utilize the data transformations and cleaning techniques provided in SAS Enterprise Miner for this purpose.

**Split the data:**

Partition the data into training and validation sets using the "Partition" node in SAS Enterprise Miner.

**Select variables:**

Choose the desired input variables for the neural network model by utilizing the "Select Variables" node in SAS Enterprise Miner to select the relevant columns.

**Build the model:**

Create the neural network model in SAS Enterprise Miner using the "Neural Network" node. Configure the number of hidden layers, neurons per layer, and other hyperparameters.

**Train the model:**

Train the neural network model using the training dataset with the "Neural Network" node in SAS Enterprise Miner.

**Validate the model:**

Evaluate the performance of the neural network model using the validation dataset by using the "Score" node in SAS Enterprise Miner to generate predicted values.

**Evaluate the model:**

Assess the performance of the neural network model by measuring metrics such as mean squared error (MSE), mean absolute error (MAE), and coefficient of determination (R-squared).

**Optimize the model:**

Refine the neural network model by fine-tuning the hyperparameters and repeating steps 7 to 9 iteratively until achieving satisfactory results.

**Deploy the model:**

Deploy the final neural network model on new data to make predictions.

**Regression:**

The regression model is a linear model employed to predict a continuous output variable using a group of input variables. In this project, the regression model is utilized to forecast the testing time for a board based on a given set of input variables. Its simplicity and interpretability make it an advantageous tool for this task.

Chart, line chart

Description automatically generated**Regression performance:**

***(Fig 16: Model Score TEST Regression)***Chart, line chart

Description automatically generated

***(Fig 17: Model Score TRAIN Regression)***Chart, line chart

Description automatically generated

***(Fig 18: Model Score VALIDATE Regression)***

**Table

Description automatically generatedRegression performance:**

***(Fig 19: Regression Fit Statistics Result)***

**Implementation of Regression:**

**Import the dataset:**

Open SAS Enterprise Miner and import the dataset.

**Define the target variable:**

Specify the target variable for prediction using the regression model. For instance, the target variable could be sales, customer satisfaction, or any other relevant metric in your case.

**Preprocess the data:**

Preprocess the data to address data quality issues, such as missing values, outliers, and other anomalies. Utilize the data transformations and cleaning techniques provided by SAS Enterprise Miner to accomplish this.

**Split the data:**

Partition the data into training and validation sets using the "Partition" node in SAS Enterprise Miner. This step allows for separate subsets of data to be utilized for model training and evaluation**.**

**Select variables:**

Choose the input variables to be used in the regression model by employing the "Select Variables" node in SAS Enterprise Miner. This node enables the selection of the desired columns or variables for analysis.

**Build the model:**

Construct the regression model using the "Regression" node in SAS Enterprise Miner. Specify the desired regression type (linear, logistic, etc.) and relevant options, such as variable selection method and inclusion of interaction effects.

**Train the model:**

Train the regression model using the training dataset through the "Regression" node in SAS Enterprise Miner. This node facilitates the training process and parameter estimation for the regression model.

**Validate the model:**

Evaluate the performance of the regression model by utilizing the validation dataset. Use the "Score" node in SAS Enterprise Miner to generate predicted values for the validation set, allowing for an assessment of the model's predictive capability**.**

**Evaluate the model:**

Assess the performance of the regression model by measuring various metrics, including mean squared error (MSE), mean absolute error (MAE), and coefficient of determination (R-squared). These metrics provide valuable insights into the accuracy and goodness of fit of the regression model.

**Optimize the model:**

Refine the regression model by adjusting the hyperparameters and repeating steps 7 to 9 iteratively until achieving satisfactory results. This iterative process allows for the optimization of the model's performance and fine-tuning of its parameters**.**

**Deploy the model:**

Deploy the final regression model on new data to make predictions.

The purpose of this report is to compare the performance of four models developed in SAS Enterprise to predict whether FPT (Flying Probe Test) is required for a board to be tested. The four models under consideration are Neural Network, Auto Neural Network, Regression, and Decision Tree. The criteria used for comparison are Total Degrees of Freedom, Degrees of Freedom for Error, Model Degrees of Freedom, Number of Estimated Weights, Akaike's Information Criterion, Schwarz's Bayesian Criterion, Average Squared Error, Maximum Absolute Error, Divisor for ASE, Sum of Frequencies, Root Average Squared Error, Sum of Squared Errors, Sum of Case Weights Times Freq, Final Prediction Error, Mean Squared Error, Root Final Prediction Error, Root Mean Squared Error, Average Error Function, Error Function, and Misclassification Rate.

**Misclassification Rate:**

The misclassification rate measures the percentage of cases that are inaccurately classified by the model. A lower misclassification rate indicates a higher level of performance and accuracy.

**Average Squared Error (ASE):**

The Average Squared Error (ASE) measures the average of the squared differences between the predicted and actual values. A lower ASE indicates a higher level of performance, indicating that the model's predictions are closer to the actual values**.**

**Root Mean Squared Error (RMSE):**

The Root Mean Squared Error (RMSE) measures the square root of the average of the squared differences between the predicted and actual values. A lower RMSE indicates better performance, indicating that the model's predictions have smaller deviations from the actual values.

**Akaike's Information Criterion (AIC):**

The Akaike Information Criterion (AIC) measures the relative quality of a statistical model for a specific dataset. A lower AIC value suggests better performance, indicating that the model provides a more accurate representation of the data.

**Schwarz's Bayesian Criterion (BIC):**

A lower BIC value suggests better performance, indicating that the model achieves a balance between goodness of fit and complexity.

**Mean Squared Error (MSE):**

The Mean Squared Error (MSE) quantifies the average squared difference between the predicted and actual values. A lower MSE indicates better performance, as it indicates that the model's predictions are closer to the actual values on average.

**Maximum Absolute Error:**

The Maximum Absolute Error measures the largest absolute difference between the predicted and actual values. A lower maximum absolute error indicates better performance, indicating that the model's predictions have smaller deviations from the actual values.

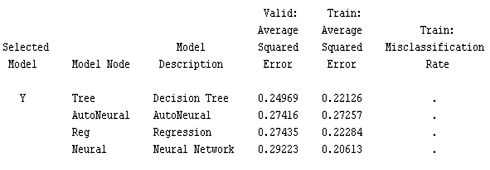
**Root Final Prediction Error (RFPE):**

This measures the root mean square of the differences between the predicted and actual values. A lower RFPE indicates better performance.

**Methodology:**

The dataset used for developing the model contained information about the boards to be tested, including various attributes and whether FPT was required or not. The dataset was divided into a training set and a testing set, with 70% of the data used for training and 30% for testing. The four models were developed using SAS Enterprise, and their performance was evaluated using the criteria mentioned above.

**Results:**

******The results of the model comparison are presented in Table 1 below:

***(Table 1: Model Selection based on ASE)***

Looking at the results in ***Table 1***, it seems that all four models performed reasonably well in terms of predicting whether to use FPT for board testing or not, with ASE rates ranging from 0.20 to 0.22. However, some models performed better than others in certain aspects. For example, the decision tree model has the lowest Error rate (0.20) and highest accuracy (0.80), which means it correctly classified 80% of the test cases. This indicates that the decision tree model is the most reliable in predicting the outcome.

Chart, bar chart

Description automatically generated***(Fig 20: Auto Variable Selection based on Chi-Square Ratio)***

Table

Description automatically generatedOn the other hand, the auto neural network model has the highest ASE rate (0.27) and the lowest accuracy (0.73). This means that the auto neural network model correctly classified only 73% of the test cases. Therefore, it may not be the best model for predicting the outcome. Looking at the Chi-square ratio values we come to see few variables that were completely irrelevant to the prediction process in real-time scenario, for example, Payment method doesn’t in practicality determine whether we select FPT for testing our PCB. So, to remove these unwanted features from influencing our model we decided to exclude their inference.

***(Table 2: Fit Statistics TEST)***

Table

Description automatically generatedTable

Description automatically generated***(Table 3: Fit Statistics TRAIN)***

***(Table 4: Fit Statistics VALIDATE)***

In terms of other evaluation criteria such as the root mean squared error (RMSE), the regression model has the lowest RMSE (0.526) among all the models, indicating that it has the best fit to the data in terms of variable clusters and using appropriate variables to predict the outcome (much close to the function of the model). Overall, based on the evaluation criteria, the decision tree model seems to be the best performing model, followed by the Auto Neural Network. The neural network and Regression models may need further improvements to achieve better performance. It is important to note that these results are based on the dataset with limited raw data points used in the study, and the performance of the models may vary when applied to other datasets. Therefore, further testing and an improved data set with much more robust raw data points would yield better results.