# CAPSTONE PROJECT 2

**DOMAIN**: Semiconductor manufacturing process

• **CONTEXT**: A complex modern semiconductor manufacturing process is normally under constant surveillance via the monitoring of signals variables collected from sensors and or process measurement points. However, not all of these signals are equally valuable in a specific monitoring system. The measured signals contain a combination of useful information, irrelevant information as well as noise. Engineers typically have a much larger number of signals than are required. If we consider each type of signal as a feature, then feature selection may be applied to identify the most relevant signals. The Process Engineers may then use these signals to determine key factors contributing to yield excursions downstream in the process. This will enable an increase in process throughput, decreased time to learning and reduce the per unit production costs. These signals can be used as features to predict the yield type. And by analysing and trying out different combinations of features, essential signals that are impacting the yield type can be identified.

• **DATA DESCRIPTION**: sensor-data.csv : (1567, 592)

The data consists of 1567 examples each with 591 features.

The dataset presented in this case represents a selection of such features where each example represents a single production entity with associated measured features and the labels represent a simple pass/fail yield for in house line testing. Target column " –1" corresponds to a pass and "1" corresponds to a fail and the data time stamp is for that specific test point.

• **PROJECT OBJECTIVE**: We will build a classifier to predict the Pass/Fail yield of a particular process entity and analyse whether all the features are required to build the model or not.

**Steps and tasks**:

1. Import and explore the data.

2. Data cleansing:

• Missing value treatment.

• Drop attribute/s if required using relevant functional knowledge.

• Make all relevant modifications on the data using both functional/logical reasoning/assumptions.

3. Data analysis & visualisation:

• Perform detailed relevant statistical analysis on the data.

• Perform a detailed univariate, bivariate and multivariate analysis with appropriate detailed comments after each analysis.

4. Data pre-processing:

• Segregate predictors vs target attributes

• Check for target balancing and fix it if found imbalanced (read SMOTE)

• Perform train-test split and standardise the data or vice versa if required.

• Check if the train and test data have similar statistical characteristics when compared with original data.

5. Model training, testing and tuning:

• Model training:

- Pick up a supervised learning model.

- Train the model.

- Use cross validation techniques.

- Apply GridSearch hyper-parameter tuning techniques to get the best accuracy.

Suggestion: Use all possible hyper parameter combinations to extract the best accuracies.

- Use any other technique/method which can enhance the model performance.

Hint: Dimensionality reduction, attribute removal, standardisation/normalisation, target balancing etc.

- Display and explain the classification report in detail.

- Apply the above steps to atleast 3 different kind of models that you have learnt so far and models that you haven't learned till now (Randomforest, SVM, Naive bayes etc).

• Display and compare all the models designed with their train and test accuracies.

• Select the final best trained model along with your detailed comments for selecting this model.

• Save the selected model for future use.

6. Conclusion and improvisation:

• Write your conclusion on the results