PREDICTIVE MAINTENANCE

INTRODUCTION

Product reliability is a key contributor to the success of companies that manufacture equipment. In order to avoid the impact of unexpected breakdowns, the simplest process for performing scheduled maintenance is to assign technicians to check the equipment on a regular basis. This is referred to as **Preventative Maintenance**. However, this approach is not cost-effective, since too frequent visits result in wasted labor and travel costs, while too large a gap between visits can result in problems occurring without warning.

Predictive Maintenance *(PdM)* is a promising alternative that makes predictions about equipment failures to allow for advance scheduling of corrective maintenance. It thus aims to :

- Prevent unexpected equipment breakdowns and improve asset reliability for customers
- Reduces the additional operational costs caused by over-maintenance.

OBJECTIVE

In this study, our focus is on formulating the PdM model for ATMs under the Bank of Baroda. In simple terms, the goal is to predict whether or not an ATM will fail in the near future. This is thus analogous to a **binary classification problem**.

SOURCE OF DATA

We have two main sources from which instances is to be generated.

1. **System logs**, that include **error event data**. This data is provided by the equipment owner.
    - **TermId** : ATM Id. There are **3681** unique machines.
    - **FaultStartTime :** Start time of system error, in the format, **YYYY-MM-DD HH:MM:SS**.
    - **FaultEndTime :** End time of system error, in the format, **YYYY-MM-DD HH:MM:SS**.
    - **FaultDesc :** Type of error or system warning. There are **103** different errors.
    - **Age :** Difference in start time and end time, in nearest minutes.

TermId	FaultStartTime	FaultEndTime	FaultDesc	age
1FNDEO04	2020-09-23 15:48:35.933	2020-09-23 16:29:27.947	type 3 currency cassette low (NCR)	41
1FNDAU04	2020-09-23 15:48:30.100	2020-09-23 16:29:22.147	supervisor mode alarm is on (NCR)	41
1FNDAU04	2020-09-23 15:48:30.100	2020-09-23 16:29:22.147	close (NCR)	41
1FNBOM76	2020-09-23 15:47:54.227	2020-09-23 16:28:52.513	supervisor mode alarm is on (NCR)	41
1FNBOM76	2020-09-23 15:47:54.227	2020-09-23 16:28:52.513	close (NCR)	41
1FNBND15	2020-09-23 15:47:40.983	2020-09-23 16:28:34.803	down - communication failure (NCR)	41
1FNBND13	2020-09-23 15:47:40.977	2020-09-23 16:28:34.630	magnetic card read/write suspended (NCR)	41
1FNBIK09	2020-09-23 15:47:22.947	2020-09-23 16:28:16.950	down - communication failure (NCR)	41
1FNBHR02	2020-09-23 15:47:16.870	2020-09-23 16:28:11.190	magnetic card read write warning (NCR)	41
1FNANA22	2020-09-23 15:46:27.297	2020-09-23 16:27:23.673	type 1 currency cassette low (NCR)	41

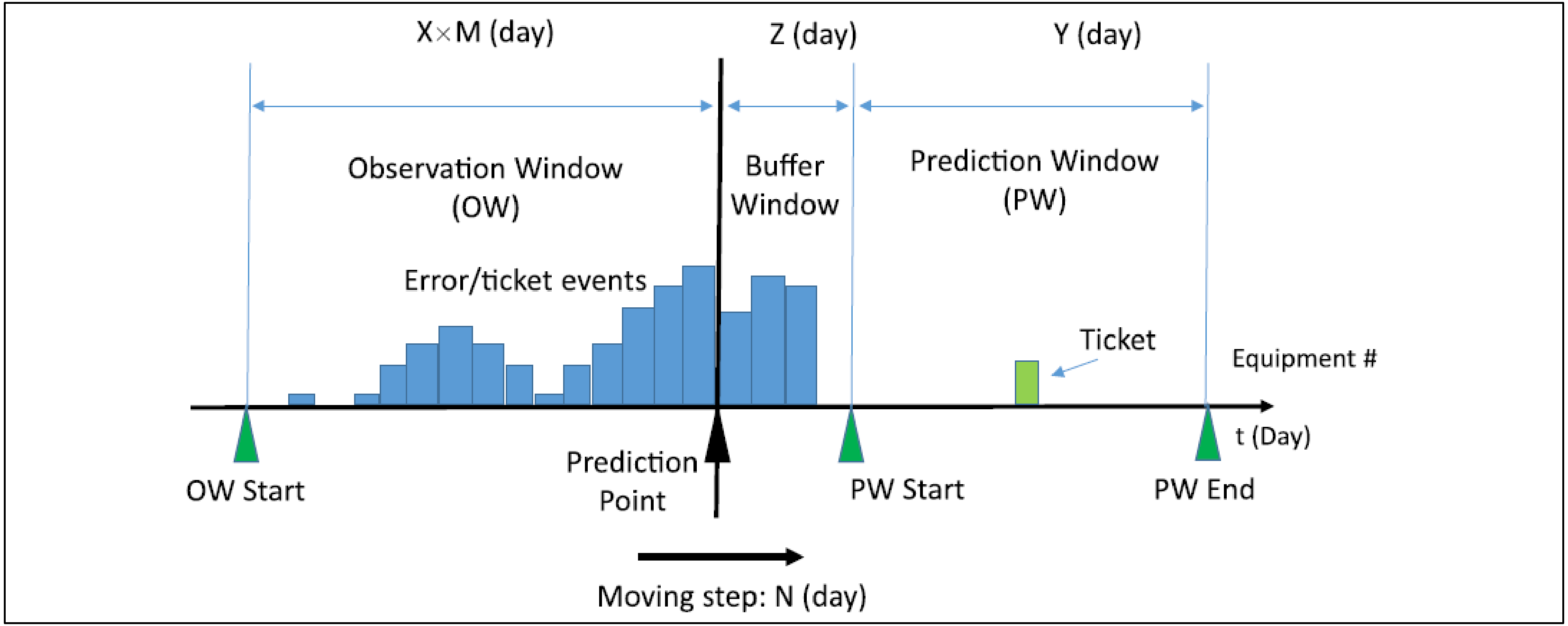
2. **Ticket creation data**, that gives information regarding the tickets created in response to past failures. The data is given by the **Maintenance Service Providers (MSP).**
    - **Ticket ID** : Identification number for ticket raised.
    - **ATM ID** : Machine Id. There are **3702** unique machines for which tickets are created.
    - **Ticket Start :** Time when ticket is created, in the format, **DD-MM-YYYY HH:MM**.
    - **Ticket Stop :** Time of closing ticket, in the format, **DD-MM-YYYY HH:MM**.
    - **Fault :** Description of the error for which the ticket is created.
    - **Activity comment :** Comment on the maintenance process.

Ticket ID	ATM ID	Ticket start	Ticket stop	Fault	Activity comment
2183494	1FNDAU14	05-08-2020 13:25	07-08-2020 12:56	ATM DOWN DUE TO LINK PROBLEM	FLM Re-Dispatched
2204298	1FNJAI44	10-08-2020 13:51	11-08-2020 14:53	ATM IS MARK DOWN	Spare Pending
2183494	1FNDAU14	07-08-2020 13:12	10-08-2020 11:37	ATM DOWN DUE TO LINK PROBLEM	Bank
2196013	1FNBNW20	14-08-2020 14:04	16-08-2020 13:21	CASH HANDLER FATAL ERROR,CASSETTE FAULTED	FLM Re-Dispatched
2204402	1FNNEA01	03-08-2020 08:19	03-08-2020 09:12	ALL CASSETTES FAULTED	FLM Re-Dispatched
2196013	1FNBNW20	16-08-2020 14:08	16-08-2020 19:48	CASH HANDLER FATAL ERROR,CASSETTE FAULTED	SLM
2196013	1FNBNW20	16-08-2020 19:48	19-08-2020 14:46	CASH HANDLER FATAL ERROR,CASSETTE FAULTED	Bank
2196013	1FNBNW20	19-08-2020 14:46	20-08-2020 15:39	CASH HANDLER FATAL ERROR,CASSETTE FAULTED	SLM
2196013	1FNBNW20	20-08-2020 15:39	21-08-2020 19:41	CASH HANDLER FATAL ERROR,CASSETTE FAULTED	FLM Re-Dispatched
2203133	1FNBNG03	06-08-2020 12:03	07-08-2020 13:49	CARD READER FAULTED	SLM

NOTE : Only data for the month of August has been considered in the analysis.

DATA GENERATION

Prediction Point	A time point when the model makes a prediction as to whether or not the device will fail in the near future. For each different prediction point, we get a separate set of values of features and the target.
Observation Window (OW)	It consists of all occurrences of errors and systems warnings, prior to the prediction point. The OW is further divided into **X periods** each of length **M days**. Each sub-window is called a **measurement unit**. Features are created based on data observed during these measurement units. Length of entire OW is (X x M).
Prediction Window (PW)	Having a **length Y**, the prediction window comes after the prediction point. The PW gives us the value of the target variable for a particular instance. We define the target as **1**, if there is any ticket created during this period, **0**, otherwise.
Buffer Window	Between the prediction point and the prediction window, there might be an optional buffer window of **Z days**, which controls the minimum time duration between prediction point and impending ticket creation. For training data, the buffer window will have no tickets.



Source - Predictive Maintenance based on event-log analysis : A case study
By J. Wang, C. Li, S. Han, S. Sarkar, X. Zhou.

Steps taken in data generation :

- Take a particular length of OW, with optimal values of **X** and **M** (discussed later), select length of **Y**. Set Z = 0.
- Extract features from the observation window. The different features created are discussed later.
- A particular time period consisting of OW, prediction point and PW will give us one row or data instance in our dataset.
- Shift the prediction point by 2 days (can be varied), to generate more rows. This means shifting the entire period of observation window-prediction point-prediction window by 2 days, along a time axis of length **N.**
- This entire window from OW start to PW end is shifted with respect to time in such a way that PW end does not go beyond 31st August, 2020. This implies that in this problem, **N** = 31 days.
- For one particular machine, the above 5 steps will give a set of rows.
- If these steps are repeated for all 3681 machines, we construct the full dataset for the month of August.

FEATURE EXTRACTION

We define five types of features as created from the data in each observation window.

1. Basic statistics-based features

- This set of features give the count of errors of each type in each measurement unit of an observation window.
- These features can be denoted by a vector **B**, such that **B = (c_{ij} , i ϵ [1, T], j ϵ [1, X])**, where T denotes the number of error types,
  X denotes the number of measurement units,
  c_{ij} denotes the number of errors of type i in j-th measurement unit.

Thus there are a total of T x X basic statistics-based features.

For example, by taking **X = 2**, we get the following table.

We get a total of **206 features**.

e1.m1	......	e103.m1	e1.m2	......	e103.m2
8		0	9		0
8		0	8		0
8		0	6		0
8		0	7		0
7		0	8		0
8		0	7		0
0		0	1		0
0	......	0	1	......	0

2. Advanced statistics-based features

- For a given error event of a particular type, we define its ***distance*** from the prediction point as $\mathbf{d = t_p - t_e}$, where t_p denotes timestamp of prediction point, and t_e denotes timestamp of the error instance.
- The time interval between two continuous occurrences of an error of given type, known as the ***error interval*** is denoted by **v.**
- Then the advanced statistics-based features are given by a vector **A**, such that,
  A = $\mathbf{(min(D_i), max(D_i), mean(D_i), mean(V_i), stdDev(V_i), i \in [1, T])}$,
  where T denotes the number of different types of errors.
  D_i denotes the set of distances of errors of type i in the observation window,
  V_i denotes the set of error intervals for error type i in observation window.

There are a total of 5 x T such features. Since we have 103 error types, we get **515 features**.

min(D39)	max(D39)	mean(D39)	mean(V39)	std(V39)	min(D40)	max(D40)	mean(D40)	mean(V40)	std(V40)
319.5167	1002.717	618.5802	45.54667	35.48192	0	0	0	0	0
319.5167	1002.717	618.5802	45.54667	35.48192	0	0	0	0	0
319.5167	895.5833	606.5393	44.31282	33.26314	0	0	0	0	0
0	0	0	0	0	363.5167	911.6333	671.3214	91.35278	61.2717
0	0	0	0	0	363.5167	911.6333	703.2042	182.7056	115.616
0	0	0	0	0	363.5167	829.5167	633.7278	233	111.6333
307.1833	502.5333	404.8583	195.35	0	796.7167	810.95	803.8333	14.23333	0
307.1833	502.5333	404.8583	195.35	0	796.7167	796.7167	796.7167	0	0

We give a snapshot of features generated for **error types 39 and 40.**

STEPS OF ALGORITHM

Step 1 : Experiments on model parameters. Tune X, M, Y and Z.

Step 2 : Experiments on features and feature selection. Predictive effectiveness of features groups and their combination

Step 3 : Classification model building

Step 4 : Model validation and Selection.

In each step we evaluate the results using the metrics, Precision, Recall, F1-Score, and AUC score.

Confusion Matrix :

Actual \ Predicted	0	1
0	True negative(**TN**)	False positive(**FP**)
1	False negative(**FN**)	True positive(**TP**)

Precision : Proportion of positive identifications that are actually true. Mathematically, ***precision*** = $\frac{TP}{TP+FP}$

Recall : Proportion of actual positives that are identified correctly. Mathematically, ***recall*** = $\frac{TP}{TP+FN}$

F1-Score : Harmonic mean of *precision* and *recall*. Mathematically, ***F1-score*** = $2.\frac{Precision\,.\,Recall}{Precison+Recall}$

AUC score : Area under the ROC curve which plots TP rate against FP rate, for different probability thresholds.

1. EXPERIMENTS ON MODEL PARAMETERS

Features **:** Basic and Advanced statistics-based

Classifier **:** Random Forest (maximum depth = 5, number of trees = 50)

Category 1 : Tune X keeping Y, Z, M fixed.

		Precision	Recall	F1-score	AUC
Y = 5 Z = 0 M = 5	X = 1	0.6336	0.5899	0.6109	0.6414
	X = 2	0.6222	0.5487	0.5831	0.6287
	X = 3	0.6038	0.6290	0.6162	0.6211
	X = 4	0.5950	0.6819	0.6355	0.6332
	X = 5	0.6020	0.5733	0.5873	0.6077

Category 2 : Tune Y given fixed X, M, and Z. We observe that bigger the Y value, better is the performance.

		Precision	Recall	F1-score	AUC
X = 4 M = 5 OW = 20 Z = 0	Y = 2	0.6792	0.0195	0.0379	0.6256
	Y = 3	0.7113	0.1013	0.1773	0.6307
	Y = 4	0.6767	0.3110	0.4262	0.6399
	Y = 5	0.5950	0.6819	0.6355	0.6332
	Y = 6	0.5885	0.9837	0.7360	0.6498

Category 3 : Tune M on OW, Y, and Z fixed.

		Precision	Recall	F1-score	AUC
OW = 20 Y = 6 Z = 0	X x M = 1 x 20	0.5889	0.9788	0.7354	0.6507
	X x M = 2 x 10	0.5885	0.9870	0.7374	0.6519
	X x M = 4 x 5	0.5885	0.9837	0.7360	0.6563
	X x M = 5 x 4	0.5872	0.9846	0.7357	0.6494

2. EXPERIMENTS ON FEATURES

We have two sets of features, basic statistics based features (***B***) and advanced statistics based features (***A***). We want to evaluate the predictive effectiveness of these groups individually and of their combination.

Classifier **:** Random Forest (maximum depth = 5, number of trees = 50)

Parameters **:** X x M = **2 x 10** (OW = 20), Y = **6**, Z = **0**

Data **:** There are 2 x 103 = **206** basic statistics based features, 5 x 103 = **515** advanced statistics based features (total, 721)

We do a **60%-40%** split of the dataset consisting of 8964 instances, into training and test data.

Size of training set is **5378** samples, with 3117 positive samples and 2261 negative samples

Size of test set is **3586** samples, with 2078 positive samples and 1508 negative samples

Feature Groups	Precision	Recall	F1-score	AUC
B	0.5818	0.9808	0.7298	0.6115
A	0.5925	0.9774	0.7377	0.6491
B, A	0.5885	0.9870	0.7374	0.6519

Both sets of features, B and A, individually or together are significant in predicting the target.

FUTURE TASKS

- Using Random Forest or some other technique, the most important error types are to be identified.
- A few of the error events of similar type could be clubbed together to denote a single type, in order to reduce the number of unique errors.
- Maintenance of machines or ticket creation comprise 2 levels, First Level Maintenance (FLM) and Second Level Maintenance (SLM). From hereon, we focus on only the tickets corresponding to SLM.
- Generate more data from other months.
- The pattern-based (***P***), failure similarity based (*F*) and profile based features (***R***) need to be extracted.
- Different models, apart from Random Forest are to be fitted.

OTHER FEATURES TO BE EXTRACTED

(Contd.)

3. Pattern-based features

- We define a pattern in the occurrence of event errors, as a combination of error types that repeats in different observation windows.
- **Confidence of a pattern** is the ratio of count of instances where the pattern leads to failure, to the total count of all instances containing that pattern.
- We select a pattern as a feature only if its confidence exceeds a predefined threshold.
- In any instance, if a pattern is present in the observation window, the corresponding feature is set to 1, otherwise 0.
- Pattern based features are denoted by a vector, $\mathbf{P = (p_r , r \in [1, Q])}$, where Q is the number of selected patterns. Q depends on the confidence threshold.

4. Feature Similarity based features

- Failure similarity features deal with the repeated failures of the device in the past.
- Failures of a given type often repeat and the occurrence of each such failure is preceded by similar types of errors. This helps to predict future failures by looking at past failures and the errors leading to them.
- For a particular prediction point or a particular instance, the different types of errors appearing in the observation window is denoted by a set **G**.
- A ticket created most recently with respect to the current instance, also has an observation window prior to it, which further consists of a collection of error types denoted by the set **H**.
- The failure similarity feature **F** for the given instance is calculated as the **Jaccard distance between G and H**.

5. Profile-based features

Profile based features denoted by the vector **R**, include equipment related information such as,

- Device model name
- Device ID/code
- Date of installation
- Device location