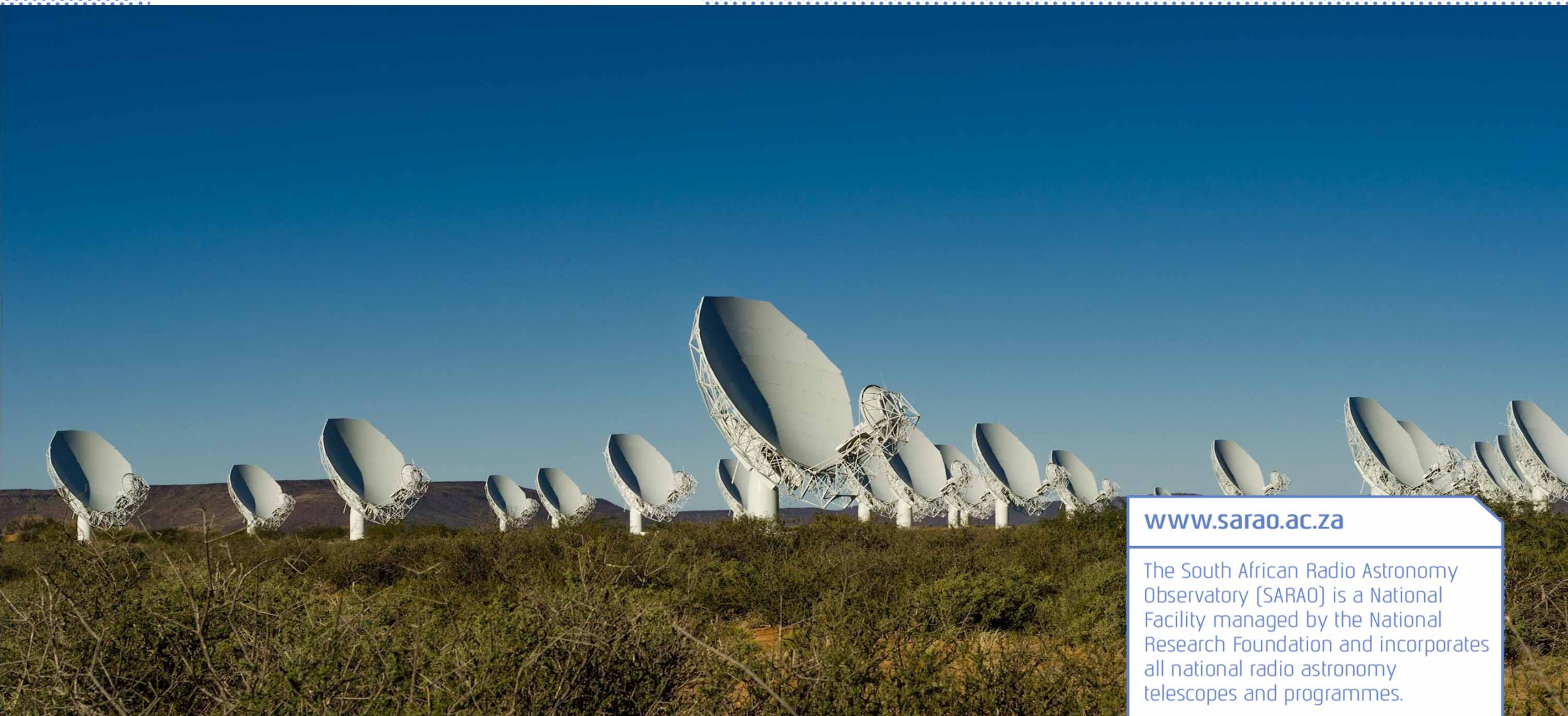
science & innovation
Department:
Science and Innovation
REPUBLIC OF SOUTH AFRICA

NRF National Research Foundation

SARAO South African Radio Astronomy Observatory



www.sarao.ac.za

The South African Radio Astronomy Observatory (SARAO) is a National Facility managed by the National Research Foundation and incorporates all national radio astronomy telescopes and programmes.

science & innovation
Department:
Science and Innovation
REPUBLIC OF SOUTH AFRICA

NRF National Research Foundation | SARAO South African Radio Astronomy Observatory

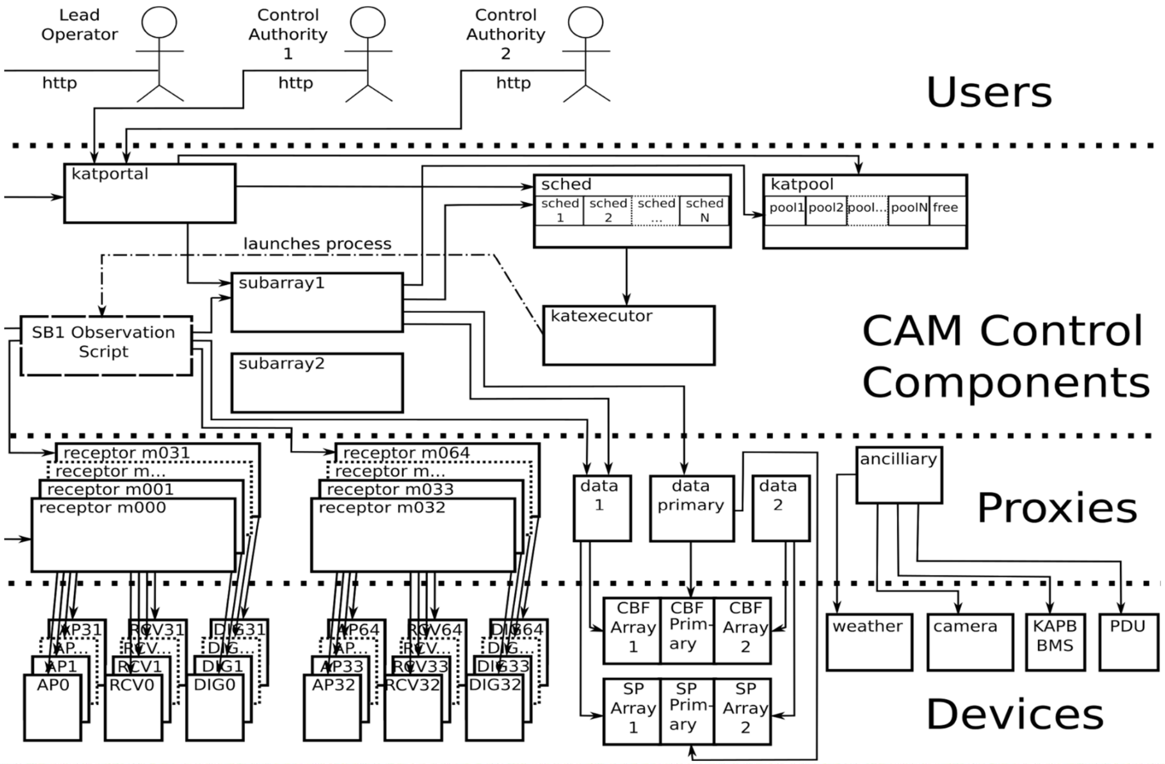
Contact information

Sipho Mnisi
Software Engineer
Email: smnisi@sarao.ac.za

AGENDA

SYSTEMS MODELLING, AI/ML ALGORITHMS APPLIED TO CONTROL SYSTEMS

MEERKAT ARCHITECTURE



MEERKAT CAM

MEERKAT CAM(CONTROL AND MONITORING)

Is a predominantly a software system to which a large number of devices and components that make up the 64 receptor radio telescope in the Karoo, South Africa, are connected via the Karoo Array Telescope Communication Protocol(KATCP).

KATCP is used extensively for internal communications between CAM components and other subsystems.

CAM MAIN FUNCTIONS

- Hardware Control
- Monitoring and Archiving
- Observation Control
- System Control

MeerKAT CAM has many software components, some components connect to hardware devices and others connect to software components.

Components can call requests on connected components for control purposes.

A KATCP request is analogous to method or command calls of other platforms. For monitoring purposes, KATCP provides the concept of sensors.

WHAT DOES CAM DO?

A sensor is a fundamental concept in KATCP and a collection of sensor types are available.

The following types are currently supported: *integer, float, boolean, timestamp, discrete, address* and string.

Sensors always have a status and the following statuses are supported: *unknown, nominal, warn, error, failure, unreachable* and *inactive*.

Example of a sensor:


```
{
  "name": "m000_rsc_rxl_cryostat_pressure",
  "time": 1505982067.202219,
  "value": 1013.25,
  "status": "nominal",
  "value_ts": 1505977839.44
}
```

What we would want to do

- **Predict which device is going to fail**
- **Predict the remaining life of a device**

Predictive Maintenance and Monitoring

Predictive maintenance (PdM): is a type of condition-based maintenance that monitors the condition of hardware assets through sensor devices to predict when the asset will require maintenance and thus, prevent equipment failure.

Predictive Maintenance?

- provides continuous insights on the ***actual*** condition of the equipment rather than relying on the ***expected*** condition of the equipment based on a historical baseline.
- corrective maintenance is only carried out when there is a need to do so, and so avoids incurring unnecessary maintenance costs and machine downtime
- uses time series historical and failure data to predict the future potential health of equipment and so anticipate problems in advance

Predictive maintenance also differs from preventive maintenance in the diversity and breadth of real-time data used in monitoring the equipment.

Various condition monitoring techniques such as sound (ultrasonic acoustics), temperature (thermal), lubrication (oil, fluids) and vibration analysis can identify anomalies and provide advance warnings of potential problems.

A rising temperature in a component, for example, could indicate airflow blockages or wear and tear; unusual vibrations could indicate misalignment of moving parts; changes in sound can provide early warnings of defects that can't be picked up by the human ear.

What to predict?

The end result is a model that:

- Predict which device is going to fail; *classification problem*
- Predict the remaining life of a device; *regression problem*

Classification: refers to a predictive modeling problem where a class label is predicted for a given example of input data. For example, given an email, classify if it is spam or not.

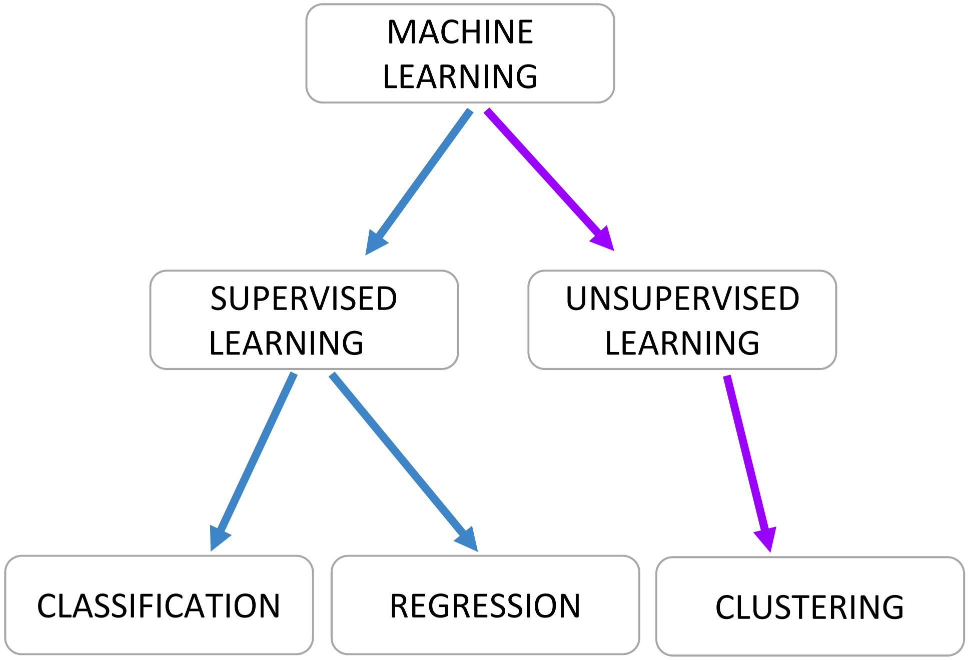
Regression: a predictive modeling technique that estimates the relationship between two or more variables. It focuses on the relationship between a dependent (target) variable and an independent variable(s) (predictors). An example would be a house being predicted to sell for a value in a certain range given a number of factors(predictors)

How do we get there?

Machine Learning

Machine learning (ML) is a subfield of Artificial Intelligence (AI) focused on building computer systems that learn from data.

Depending on the nature of the data and the desired outcome, one of four learning models can be used: *supervised, unsupervised, semi-supervised, or reinforcement.*



Actions

Convert business use case into a machine learning use case.

The following questions will inform a decision for a use case which ultimately will get converted into a hypothesis.

- How the output of the PdM model will be used?
- What is the definition of device failure or breakdown?
- What signals have patterns of breakdowns or degraded performance?
- The frequency with which signals are collected?
- How much normal and failure data is available?

Creating features:

- Features - representation of data in a form that can be used to map input data to output predictions.

Build model and get actionable insights

- Once you have created features, then you can select an algorithm and build a model.

Actions

Forming a hypothesis

A hypothesis in machine learning is a model that approximates a target function and performs mappings of inputs(data) to outputs(predictions).

Given the classification and regression scenario of estimating when a device will fail or the device's remaining life, the following hypothesis could be drawn.

- Predict if equipment **will fail in the next 'K'** period
- Predict if equipment **will fail in the next 'K1', 'K2'...Kn** period
- Predict if equipment **will fail in the next 'K'** period due to fault in part 'N'
- Predict time to failure or remaining life of equipment
- Find anomalies

Once the use case has been converted into a hypothesis, then perform the data exploration exercise to determine whether the data set and the use case are in alignment for this use case or not.

Steps to build Model

1. **Define ML use cases**
   Define ML use cases with hypothesis
2. **Data Exploration**
   Perform exploratory data analysis to understand the data
3. **Select algorithm**
   Select the right ML algorithm for the task
4. **Data pipeline & feature engineering**
   Create the right features from raw data for the ML task
5. **Build ML model**
   Develop the first iteration of the ML model
6. **Iterate ML model**
   Refine the ML model to improve performance and efficacy
7. **Present Results**
   Present results of the model in a way that demonstrates its value
8. **Plan for deployment**
   Prepare for deployment in production
9. **Operationalize model**
   Deploy and operationalize ML model in production
10. **Monitor model**
    Monitor deployed ML model and retrain or rebuild when performance degrades

Start a new ML project

What to look for in PdM

When building a Predictive Maintenance model, for a device that has sensor data transmitted from it, where the aim is to determine when this device will fail - the objective is to look for patterns in the data exploration such as:

- the speed, efficiency, pressure etc being good/optimal when the device is new degrading as the device gets older.
- heat, noise, vibration could start low when the device is new and increasing as it ages.

These are some patterns that when spotted in the data exploration phase, could inform a decision in building a machine learning model.

Creating labels

Strategies for creating labels

A label in ML is a description that explains a piece of data to a model so that it can learn from that description. For a PdM model, the following actions are used to create labels amongst others:

- Convert failure signals into a label
- Convert degrade signals into a label

In Predictive Maintenance, one of the issues is that data labeling is a requirement in most cases. Most of the time these need to be created.

Select a label for classification

Selecting a label for classification: Will this device fail

A label needs to be created for classification to determine when a device will fail.

When a device fails, that is the final label. To catch the failure earlier, data needs to

be tagged before failure as label data. As such, that becomes the failure data, which

is a positive label. The remaining data becomes a negative label. Depending on the

use case, a determination is made of how far before the failure, the failure is to be

predicted and then tag that much period before failure as the label.

Select a label for classification

Signal value on Day 1	Normal	
Signal value on Day 2	Normal	
Signal value on Day 3	Normal	Create negative label
Signal value on Day 4	Normal	
Signal value on Day 5	Normal	
Signal value on Day 6	About to Fail	
Signal value on Day 7	About to Fail	Create positive label
Signal value on Day 8	About to Fail	
Signal value on Day 9	About to Fail	
Signal value on Day 10	Fail	

predicted and then tag that much period before failure as the label.

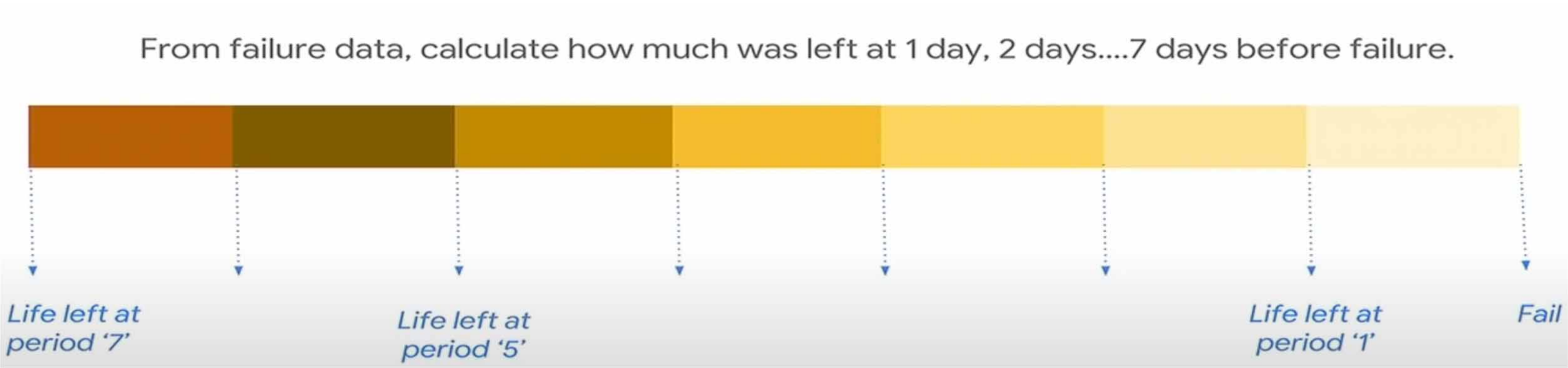
Create a label for regression

Selecting a label for regression: What is the remaining life of a device

When predicting the remaining life of a device, a deep learning model could easily

be built where there is a final failure and the sensor data.

The data can then be tagged at various places, for example, how the signal was at 15% life, at 30% life, at 60% life to enrich the model.

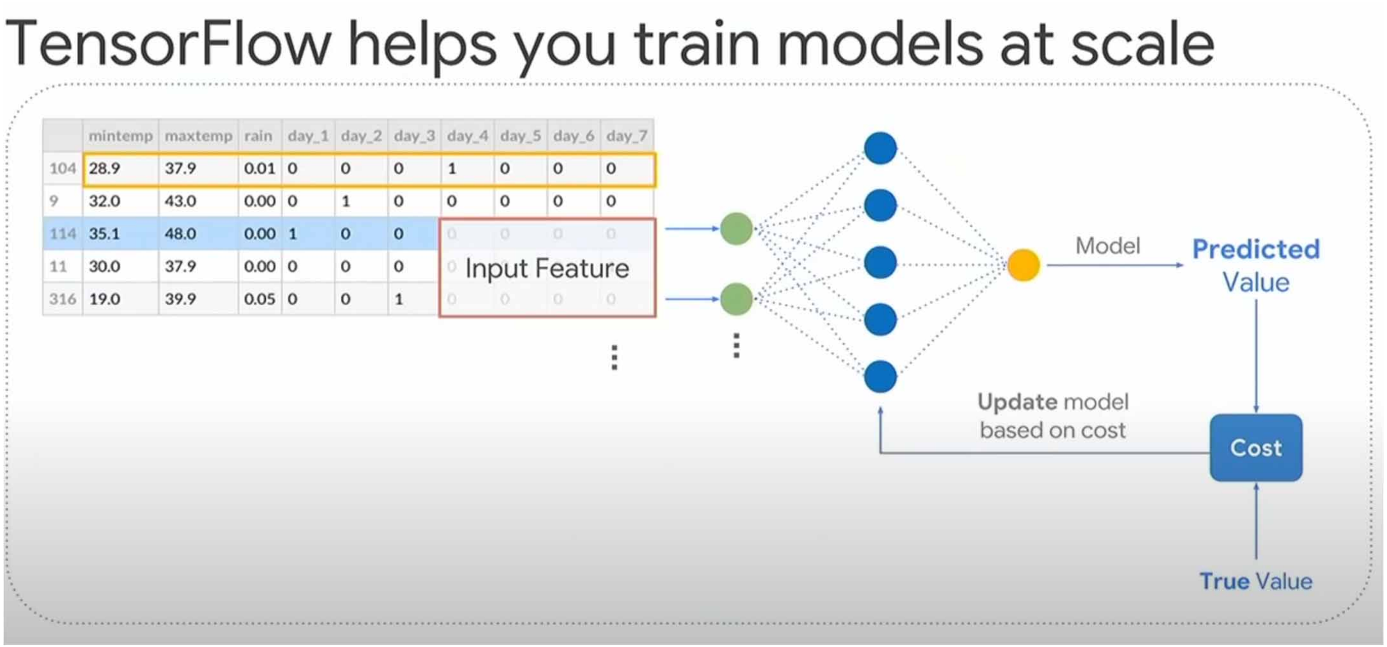


Algorithms for PdM

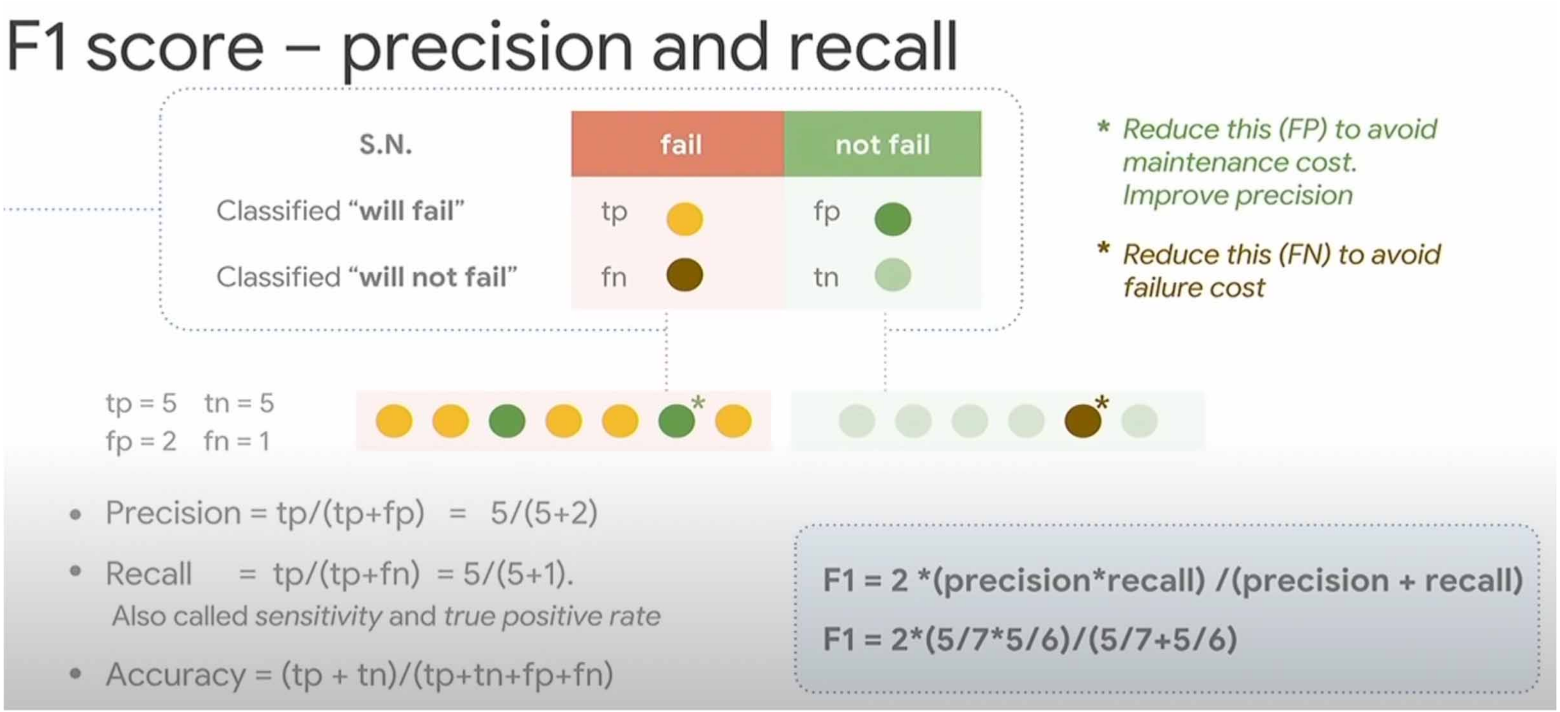
Will it fail?	Will it fail for reason Y?	After how long will it fail?	Is the behaviour anomalous?
Classification	**Multiclass classification**	**Regression**	**Anomaly detection**
Recurrent Neural Network (RNN), Long Short Term Memory (LSTM)	RNN, LSTM	RNN, LSTM	Conditional AutoEncoder
Deep Neural Network (DNN) Classification	DNN Classification	DNN Regression	AutoEncoder
Traditional ML e.g. Random Forest SVM, Decision Trees	Random Forest, Decision Trees, Hidden Markov chain	RF Regression	Anomaly MASF

DNN - Deep Neural Network **RNN** - Recurrent Neural Network **LSTM** - Long Short Term Memory

TensorFlow Example



Metrics


F1 score – precision and recall
S.N.
fail
not fail
Classified "will fail"
tp
fp
Classified "will not fail"
fn
tn
* Reduce this (FP) to avoid maintenance cost. Improve precision
* Reduce this (FN) to avoid failure cost
tp = 5 tn = 5
fp = 2 fn = 1
*
*
• Precision = tp/(tp+fp) = 5/(5+2)
• Recall = tp/(tp+fn) = 5/(5+1).
Also called sensitivity and true positive rate
• Accuracy = (tp + tn)/(tp+tn+fp+fn)
F1 = 2 *(precision*recall) /(precision + recall)
F1 = 2*(5/7*5/6)/(5/7+5/6)

Conclusion

PdM is a technique that helps to forecast when problems will arise

Challenge/Problem: difficult to make accurate predictions for complex equipment.

THANK YOU!