

Applying Machine Learning Algorithms to Support Root Cause Analysis – An Experimental Study in Automotive Engineering

Duc Tien, Vu – 15.09.2017, Munich

Advisor: Martin Kleehaus (TUM)

Dr. Valentin Solotych (IAV)

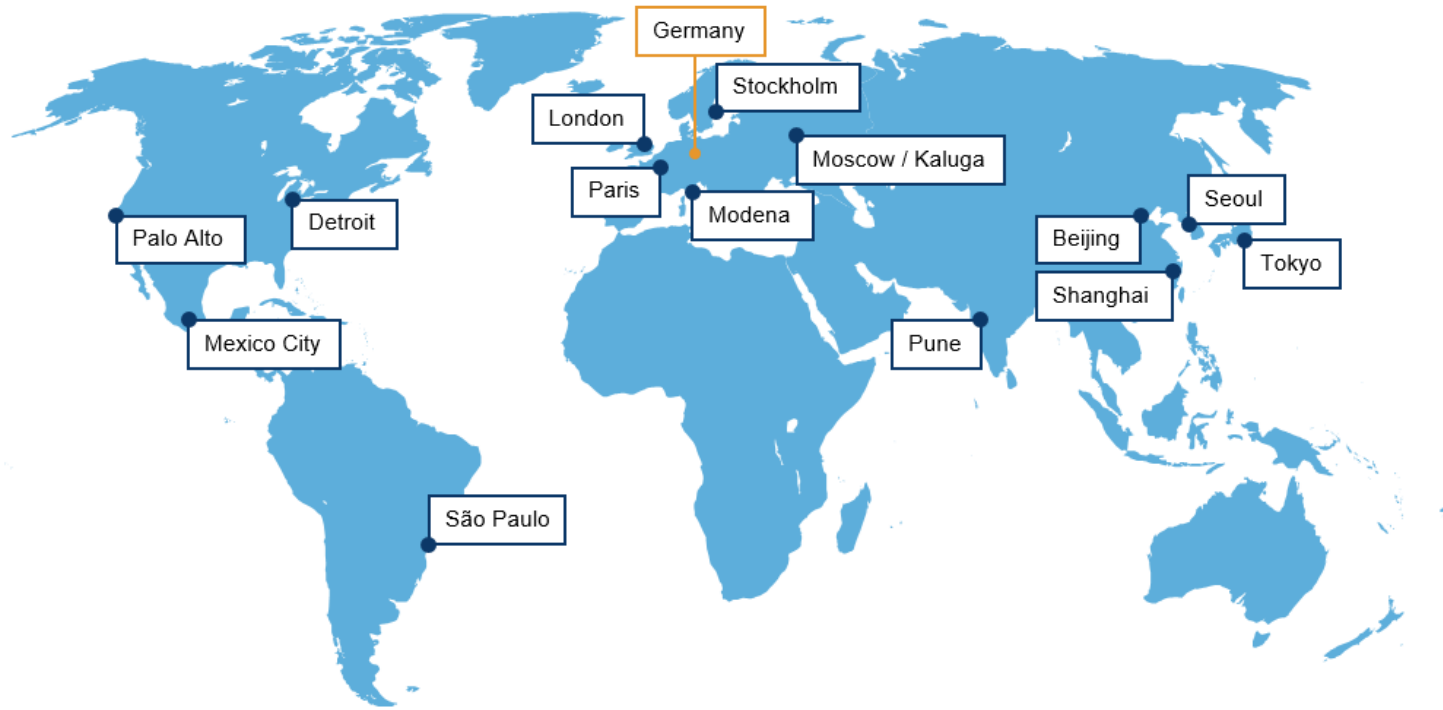
Chair of Software Engineering for Business Information Systems (sebis)

Faculty of Informatics

Technische Universität München

www.matthes.in.tum.de

IAV – Your Strong Engineering Partner



More than 30 sites worldwide



More than 30 years of experience



More than 6,500 members of staff



More than 68 % engineers



More than 750 annual turnover (€ m)

Agenda

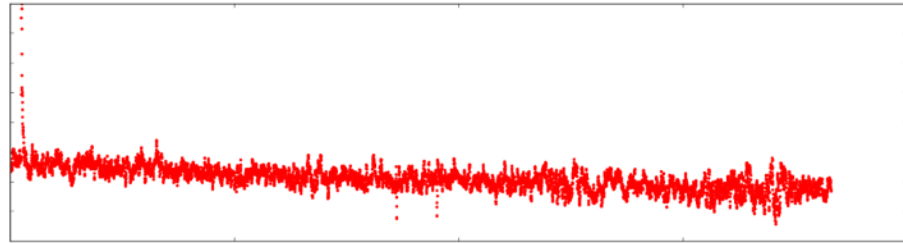


1. A little start
2. Motivation
3. Research question
4. Assessment of Anomaly Detection Techniques
5. Proposed Technique For a Recommender System
6. Implementation & Evaluation
7. Summary

A little start

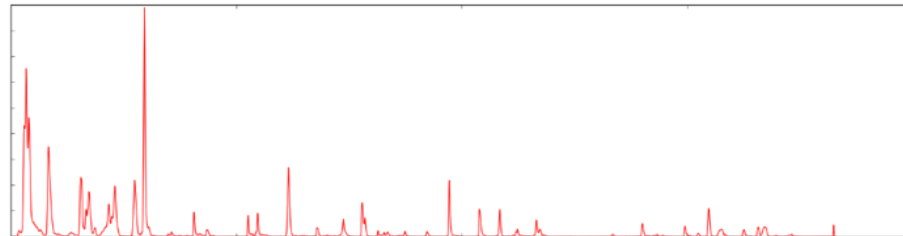
- Data from a failed emission test

Component A



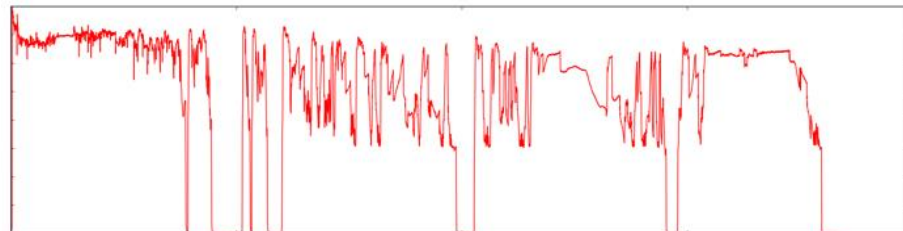
Is A the root-cause?

Emission outcome



Is B the root-cause?

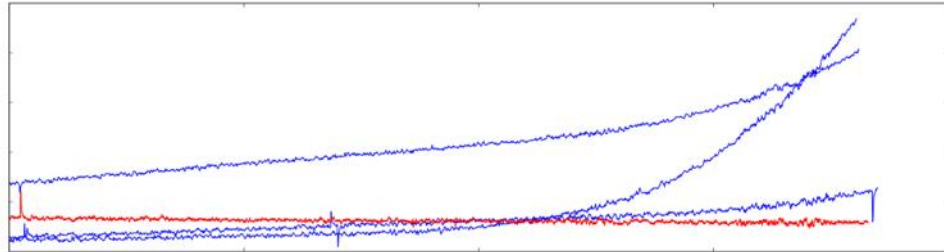
Component B



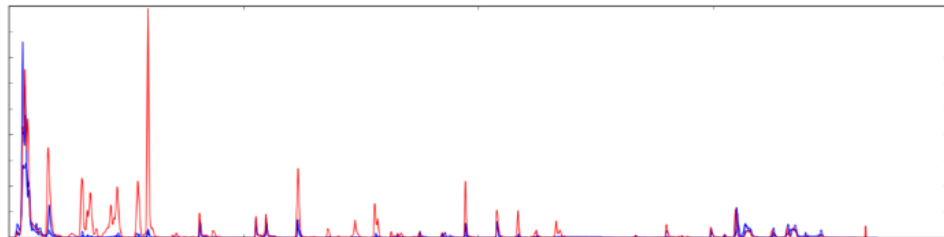
A little start

- Add more data from passed emission test
- We can tell that B influenced the emission outcome more than A

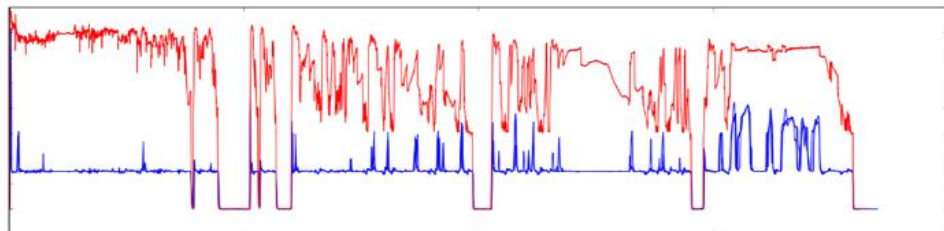
Component A



Emission outcome



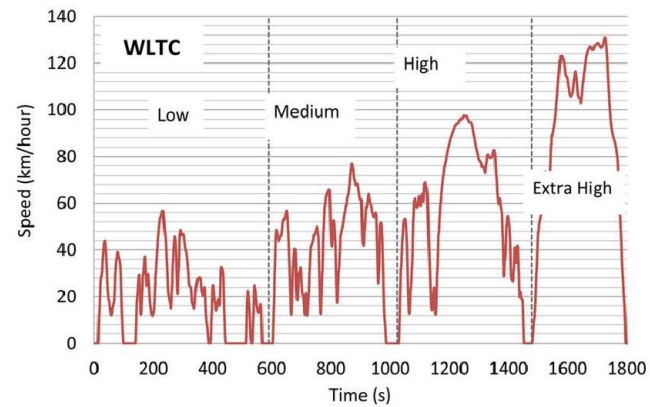
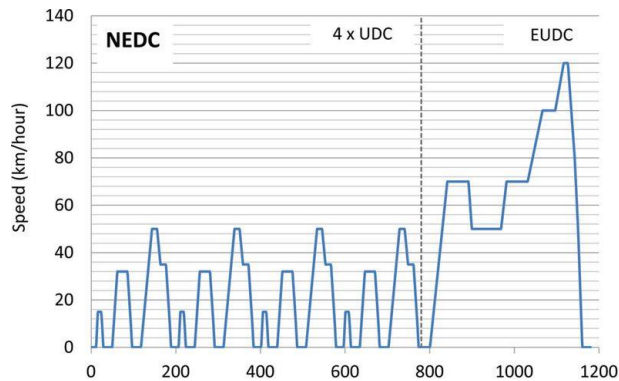
Component B



- Engineer must manually check all time series data (**800-4000 time series**) to find out which one is responsible for the anomaly
 - Requires a lot of expert knowledge and time consuming
 - Speed up the process using machine learning to detect anomaly

Motivation 2

- Currently: Fixed Driving Cycle (NEDC, WLTC, ...)



- Future: No Fixed Driving Cycle more, that means the car will be driven and measured in real life situation

→ Support from machine learning techniques is required



What are the related types of time series and anomaly?

Which algorithm can be used to detect anomalies in time series data?

How can we establish a correlation between time series

In which way can the root-cause analysis be supported?

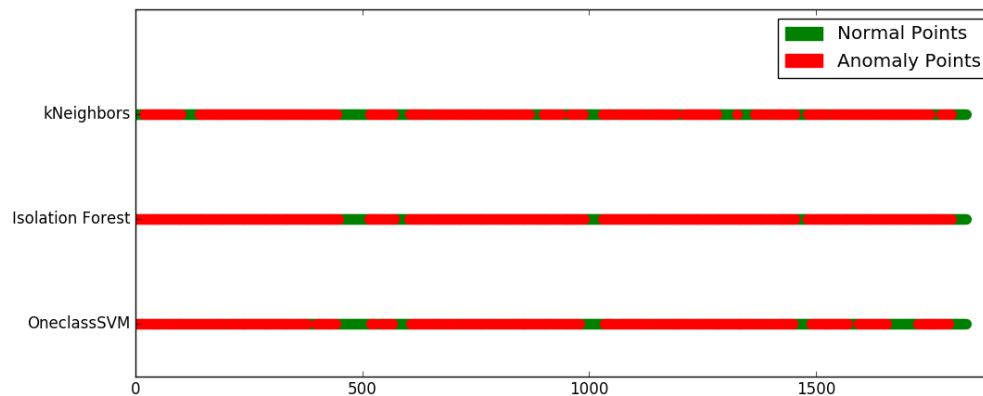
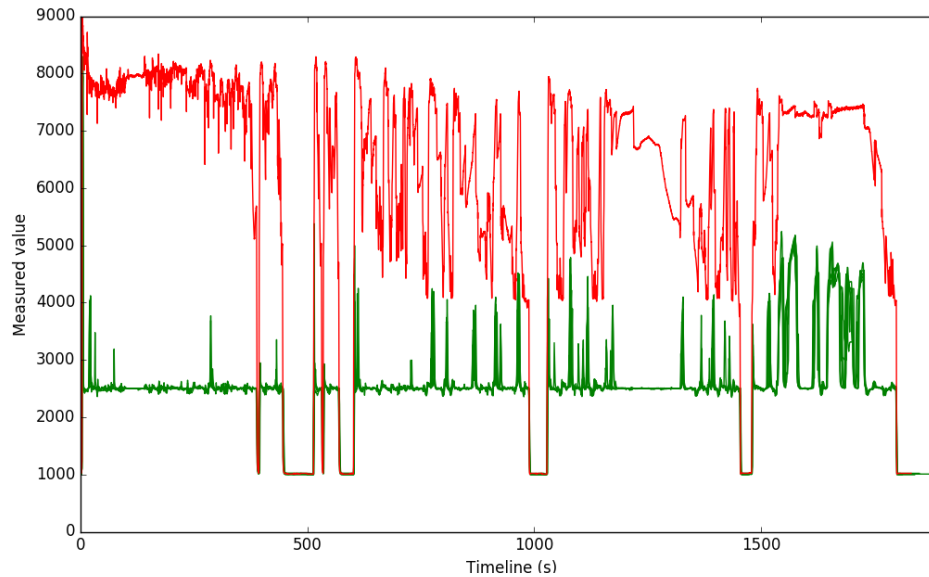
Assessment of Anomaly Detection Techniques



1. Probability-based methods
- 2. Distance-based methods**
 - kNeighbors
3. Clustering-based methods
4. Reconstruction-based methods
- 5. Domain-based methods**
 - OneClassSVM
- 6. Isolation-based methods**
 - IsolationForest
7. Information-theory – based methods

Assessment of Anomaly Detection Techniques

Accuracy test:



Accuracy

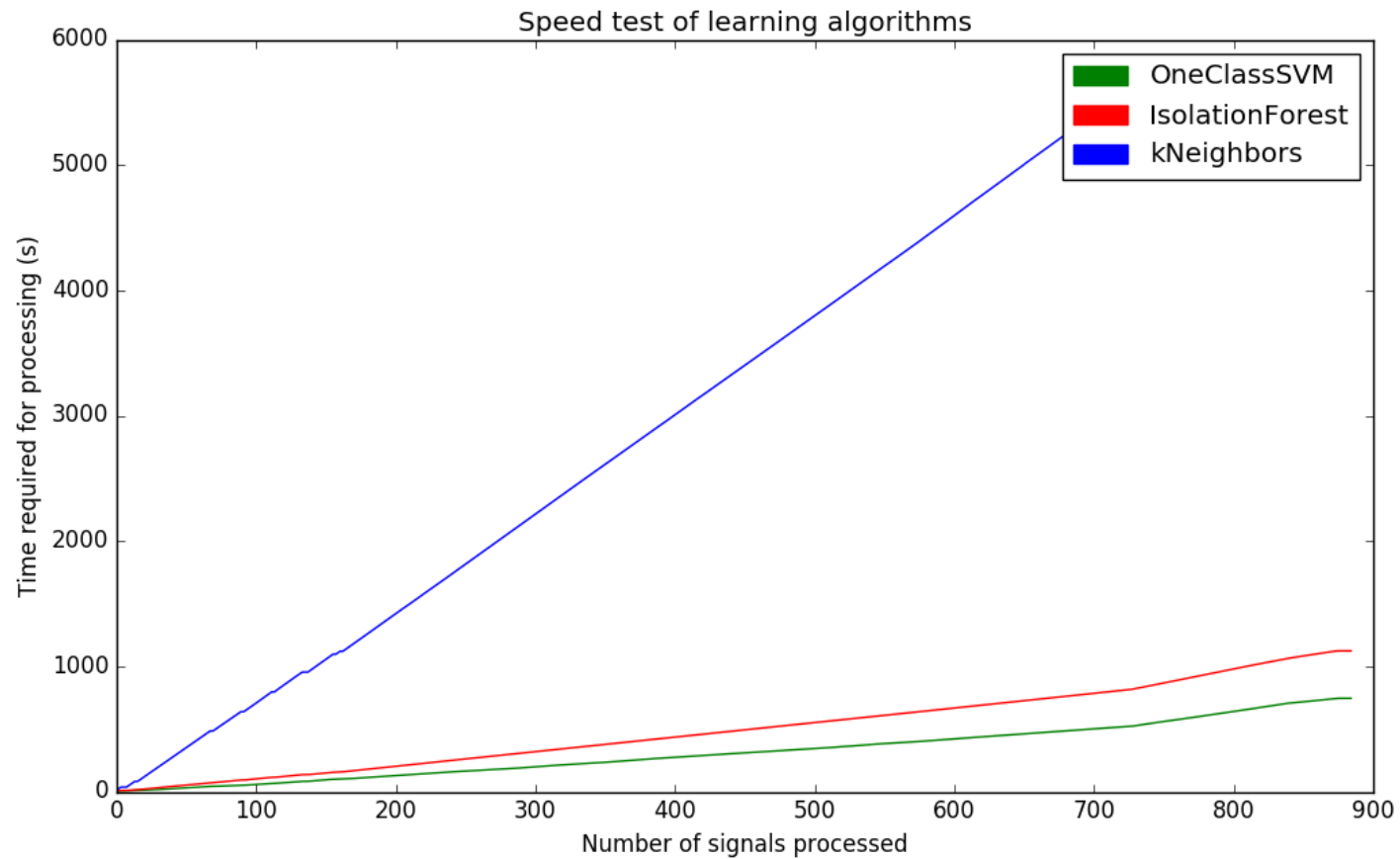
73,28%

92,67%

81,03%

Assessment of Anomaly Detection Techniques

Speed test:

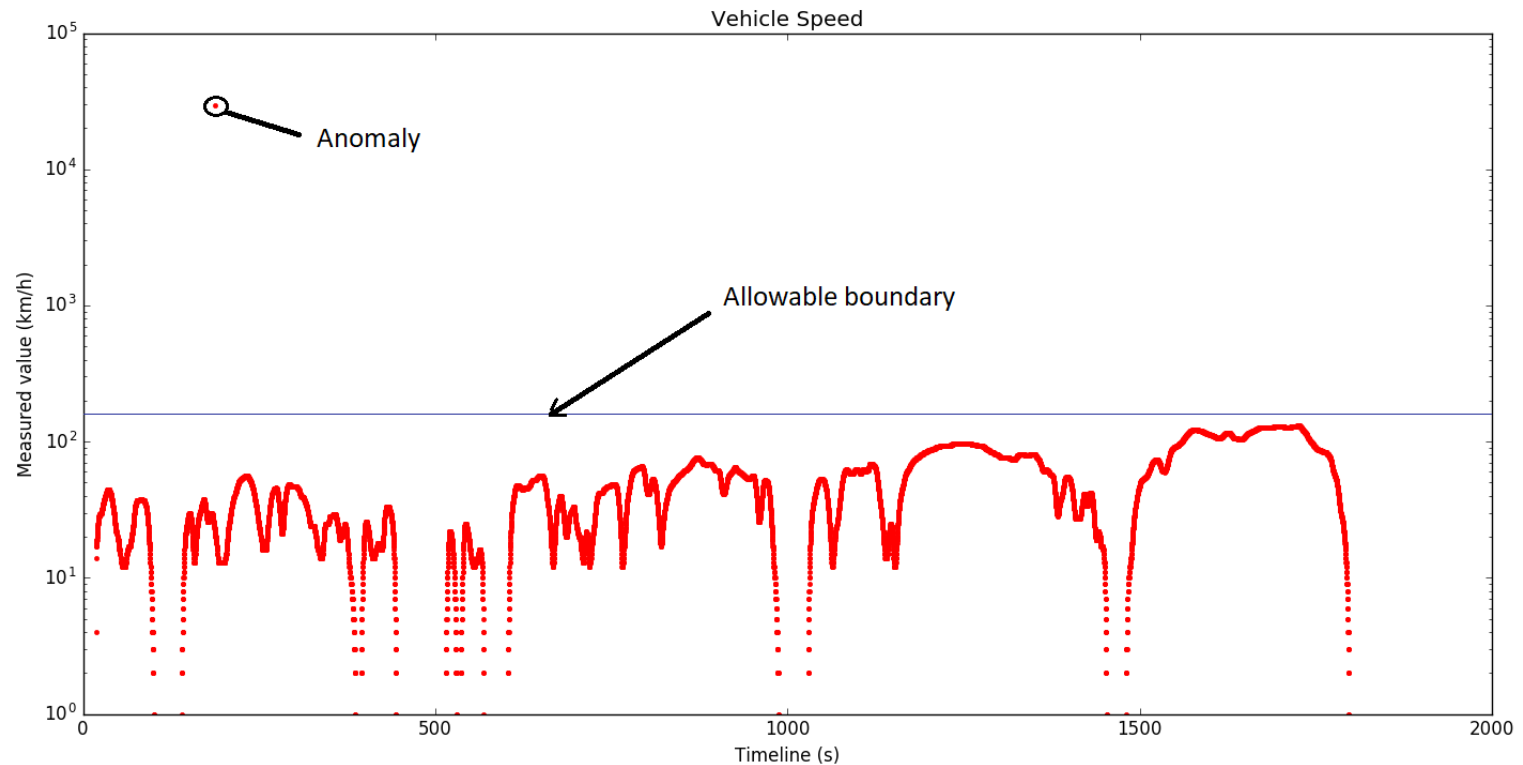


Proposed Technique for A Recommender System



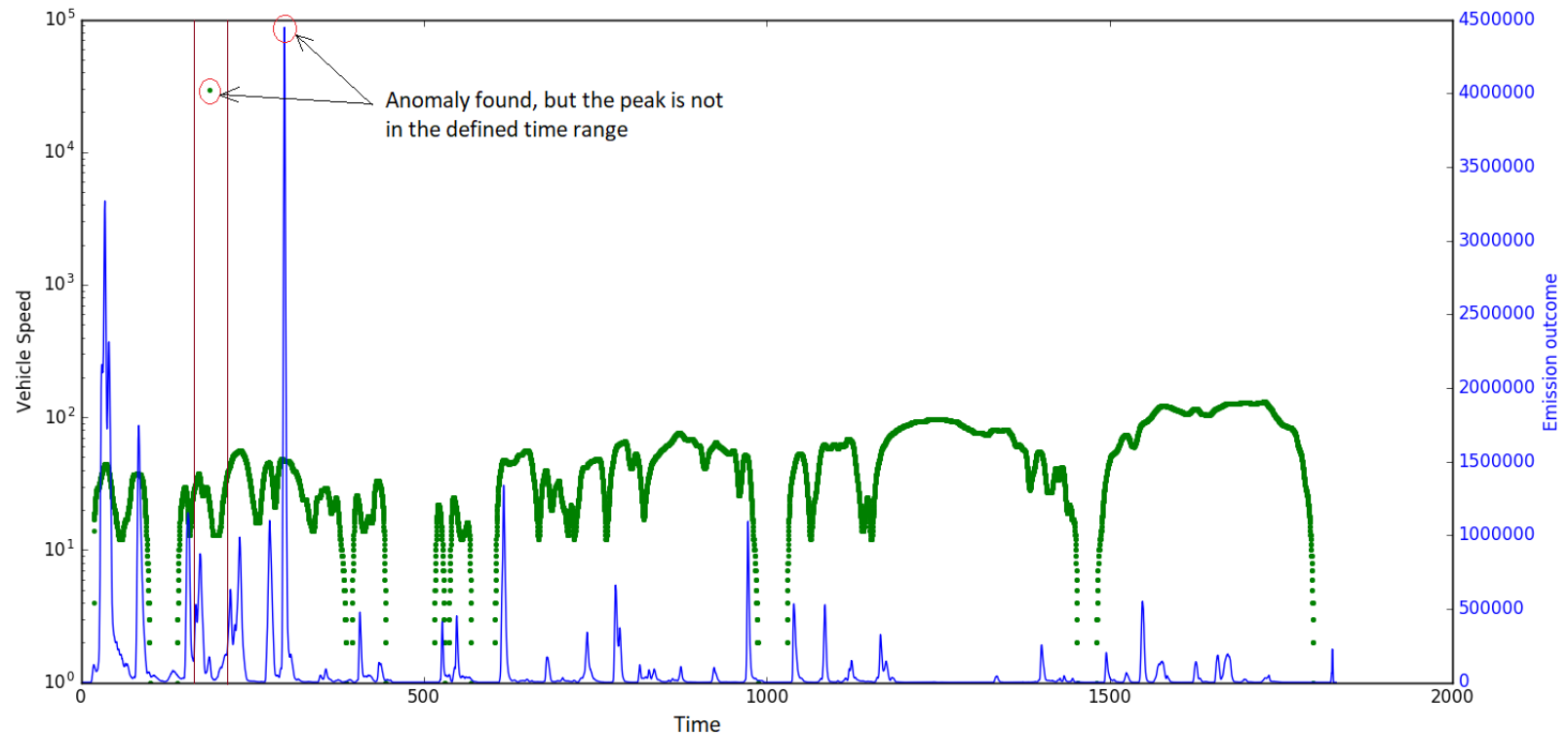
Elimination of Trivial Error (1)

- Trivial error : error that can be easily excluded from the test data
- Example: Impossible value when there is a pre-defined boundary

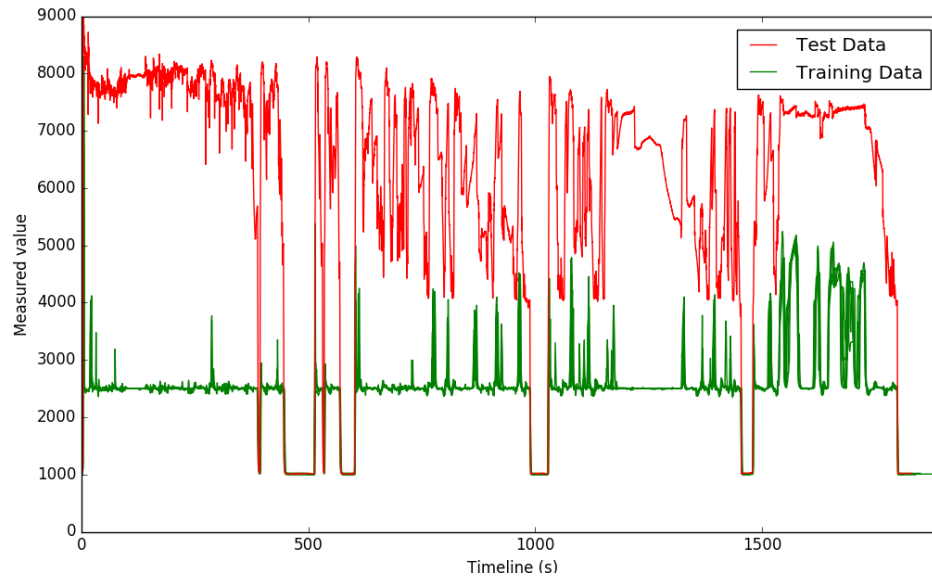


Elimination of Trivial Error (2)

- Exclude trivial error without pre-defined boundary
- Assuming that if a point anomaly is responsible for the high emission outcome, the peak of the emission outcome should be near that time point



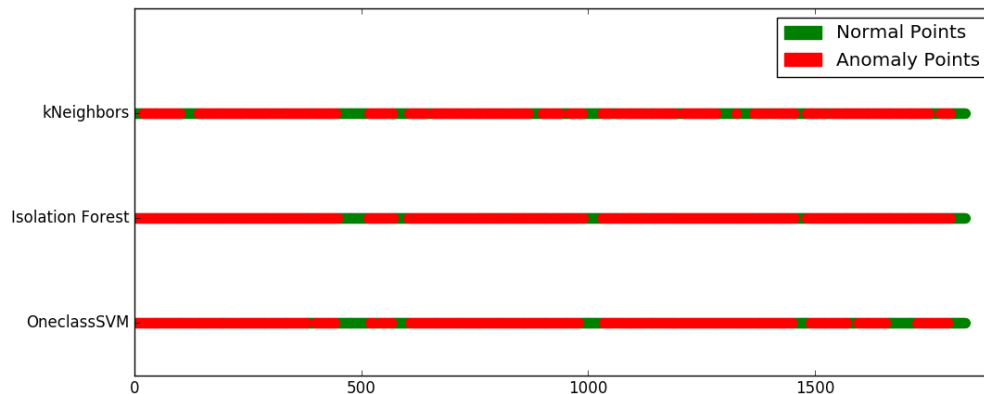
Ranking Based on Anomaly Percentage



$$\text{Anomaly percentage} = \frac{n_{adp}}{n_{adp} + n_{ndp}}$$

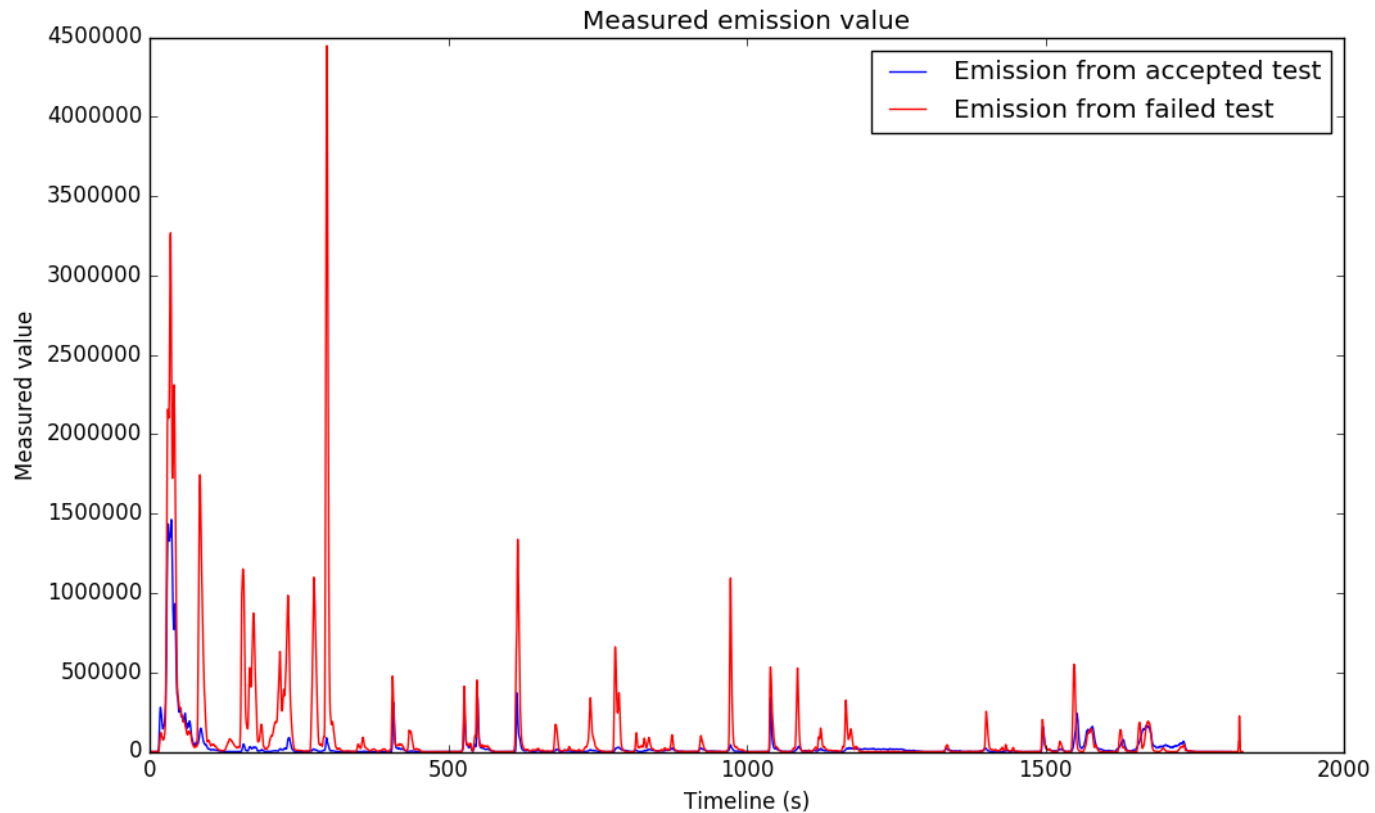
n_{adp} : Number of data points classified as **anomalous**

n_{ndp} : Number of data points classified as **normal**



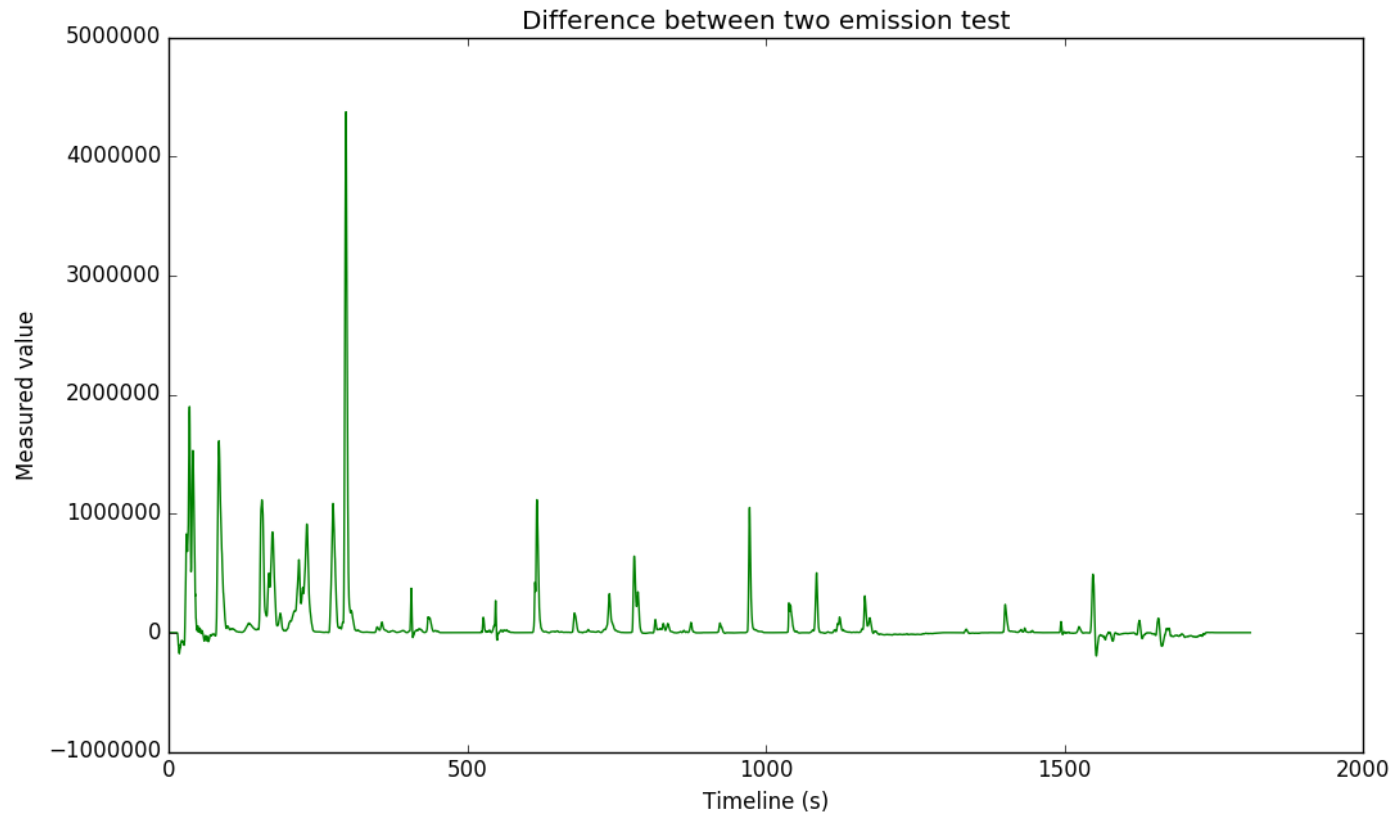
Ranking Based on Correlation (1)

- Finding the signal that mostly related to the change in emission outcome

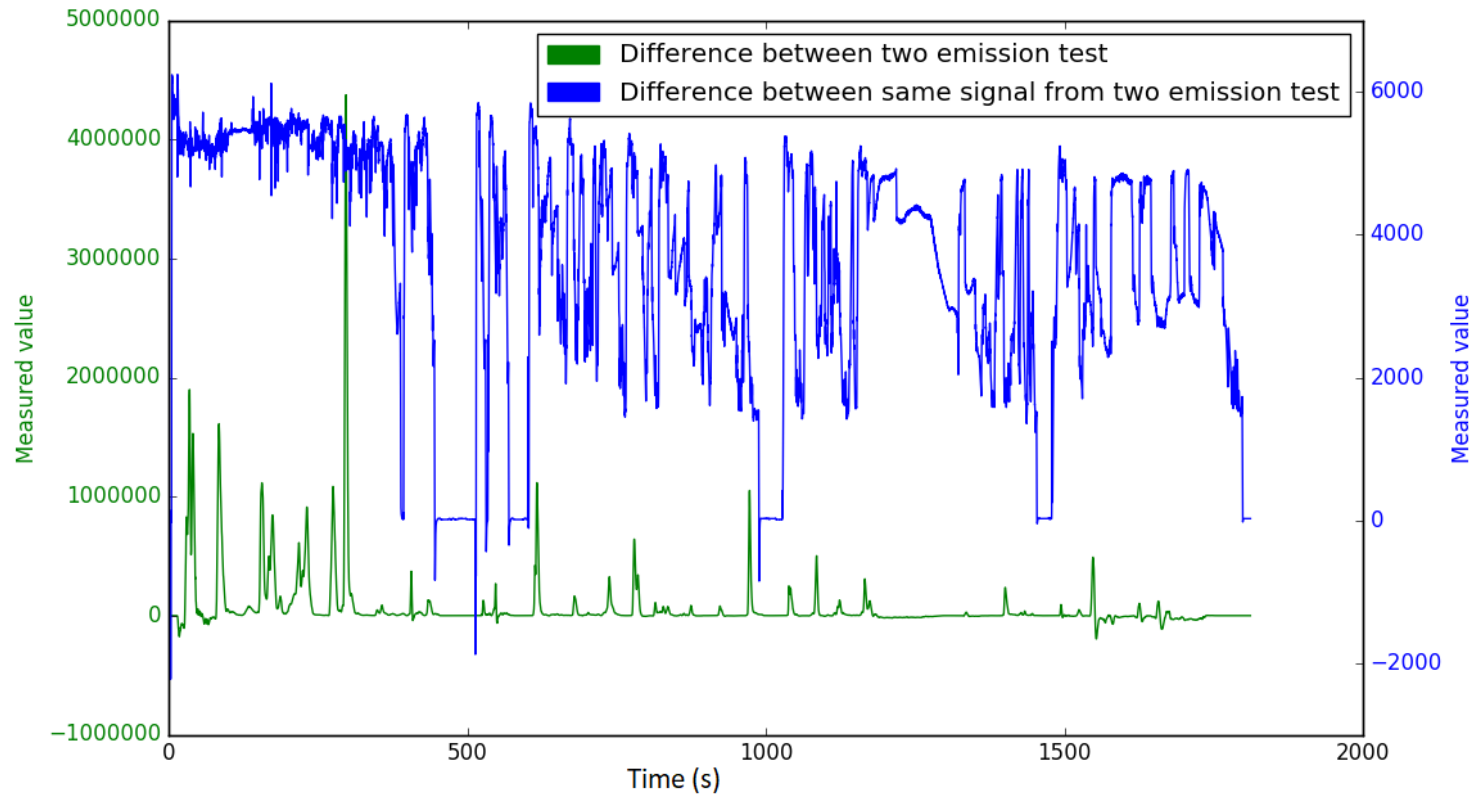


Ranking Based on Correlation (2)

- Finding the signal that mostly related to the change in emission outcome



Ranking Based on Correlation (3)



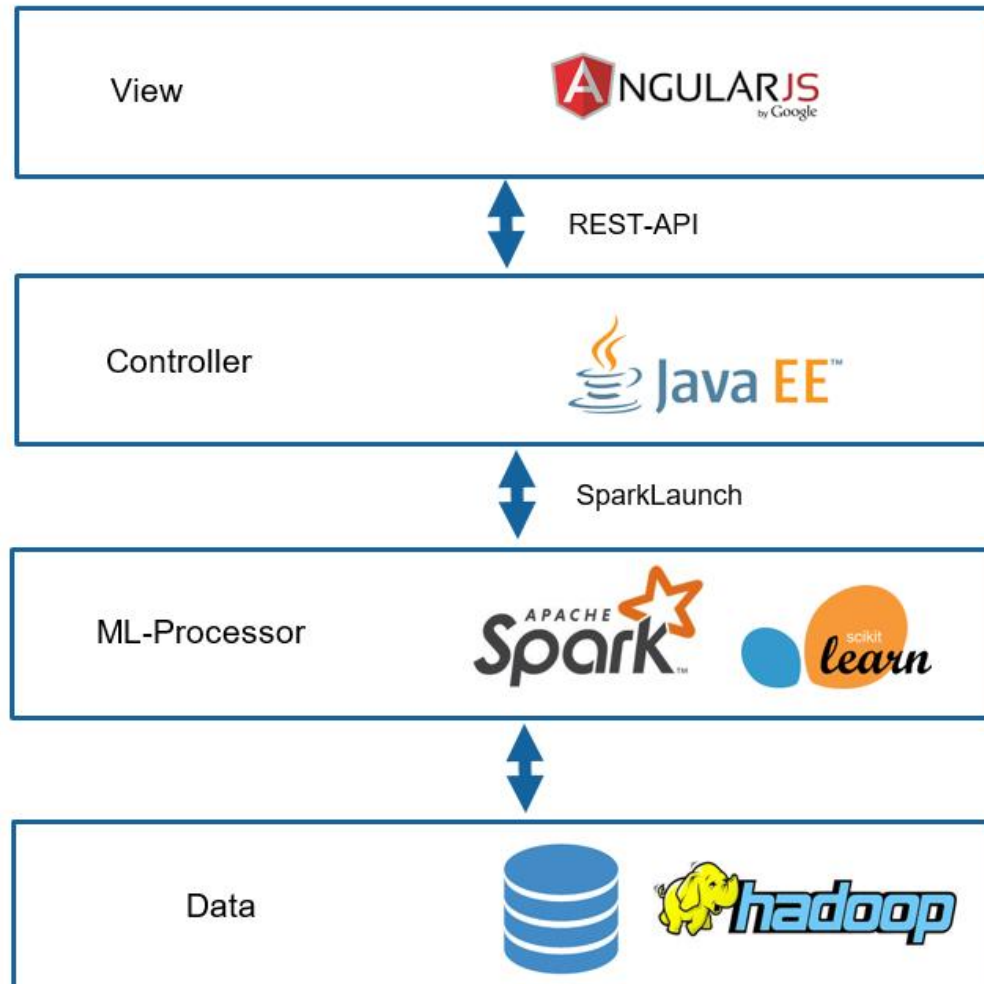
- Pearson r: Value from -1 bis 1
 - $r=-1$ → Total negative correlation
 - $r=+1$ → Total positive correlation
 - $r=0$ → No correlation

Combine of Rankings

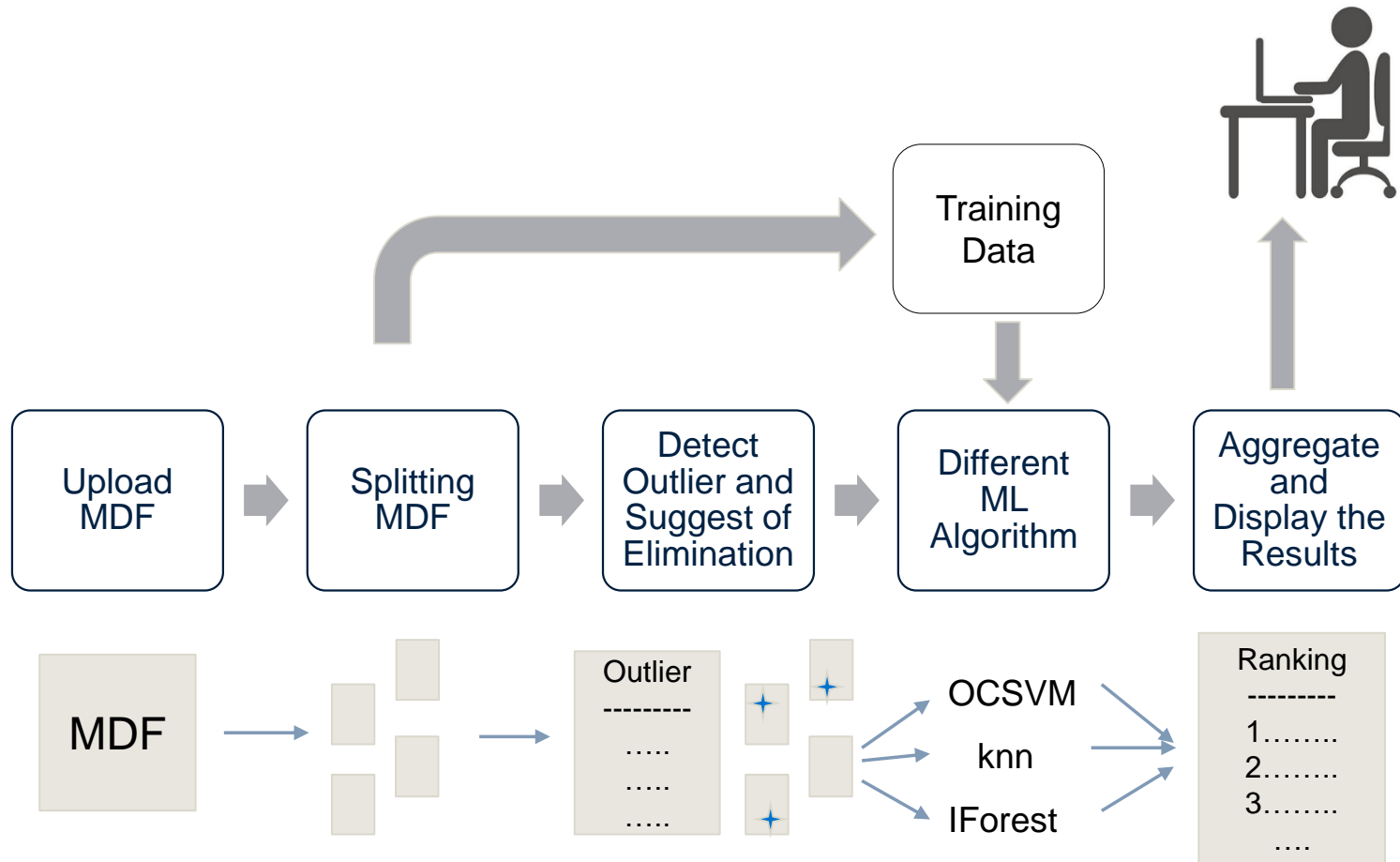
Signal	Ranking_ Anomaly	Ranking_ Correlation	Average	New ranking
A	1	7	4	3
B	4	6	5	6
C	3	2	2.5	1
D	6	3	4.5	5
E	2	4	3	2
F	5	5	5	6
G	7	1	4	3

Implementation

Technology Stack



Process of the Recommender System



TESA - Time Series Analyse App

- Dashboard
- Upload MDF-Data
- Run a python script
- Visualize result

Signal ordered and visualization

Chart

Please upload a list:

Durchsuchen... result.json

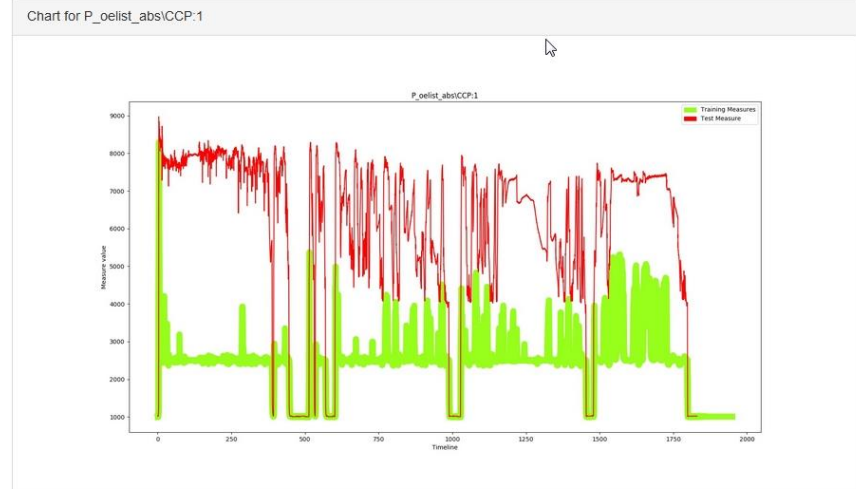
Upload Reload table

Show 10 entries Search:

Position	Signal Name	True Positive
1	BMWchas_stb_2_ub.BMWchas_b_ldl_bc\CCP:1	0.9710382514
2	P_oelist_abs\CCP:1	0.8393442623
3	Zr_auss_b_[2]\CCP:1	0.7868852459
4	Zr_auss_b_[0]\CCP:1	0.6
5	abo\CCP:1	0.2918032787
6	Lolimot_vrtein\CCP:1	0.1808743169
7	Zr_auss_b_[1]\CCP:1	0.1273224044
8	top_w\CCP:1	0.1092896175
9	Km_st\CCP:1	0.1
10	dlatmo_w\CCP:1	0.0978142077

Showing 1 to 10 of 855 entries

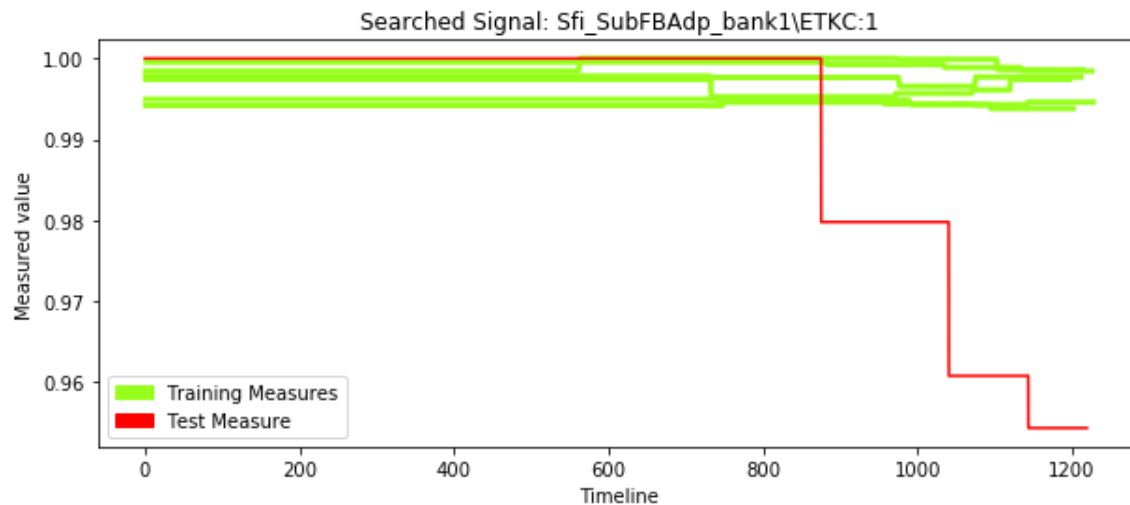
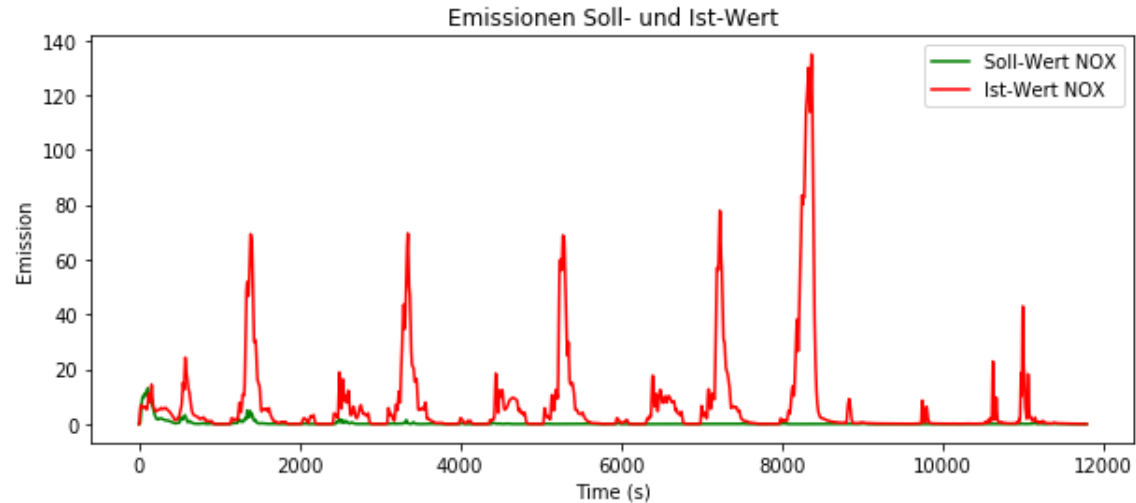
First Previous 1 2 3 4 5 ... 86 Next Last



- Experiment setting:
 - 7 problem sets
 - Each problem set is one failed emission test
 - The searched signal (the root-cause) for all problem sets is unknown
- Task:
 - Using the proposed algorithm to generate a ranking for each problem set.
 - The engineer should confirm whether the searched signal is in top 20 or not
- Result:
 - In one problem set, the searched signal is in top 20

Evaluation

- Explanation for the failure of some problem sets:
 - The searched signal doesn't have the characteristics as expected



Conclusion

- The algorithm helped pushing the root-cause to the top correctly in one case where the searched signal directly affect the emission outcome
- If the searched signal indirectly affected the emission outcome, the algorithm can not find it

Limitation

- The algorithm is highly adapted to one type of problem
- Pearson is very sensitive to time shift
- Signals related to the searched signal weren't checked

Future work

- Another algorithm using the proposed process for recommender system

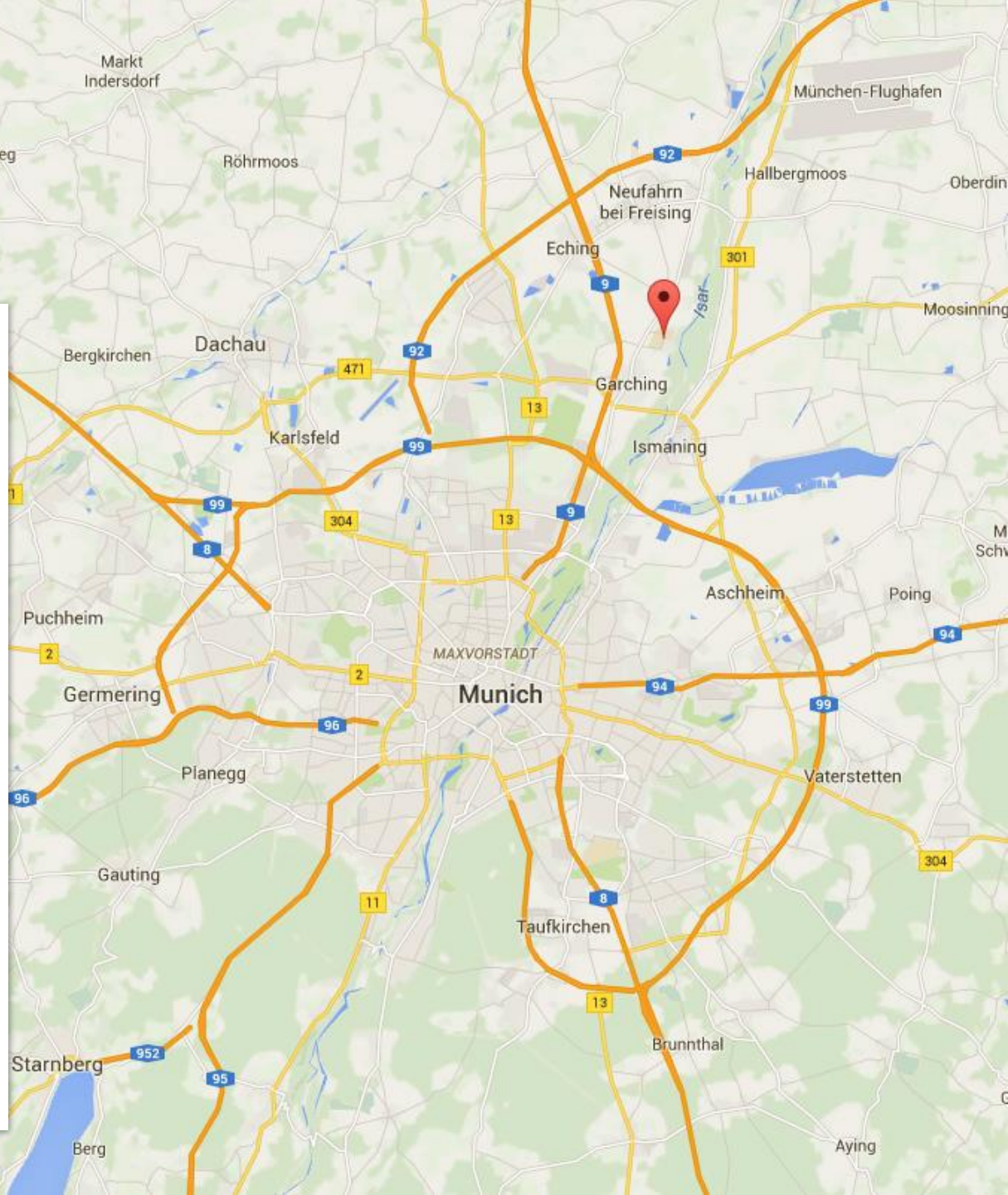


Duc Tien Vu

Technische Universität München
Faculty of Informatics
Chair of Software Engineering for
Business Information Systems

Boltzmannstraße 3
85748 Garching bei München

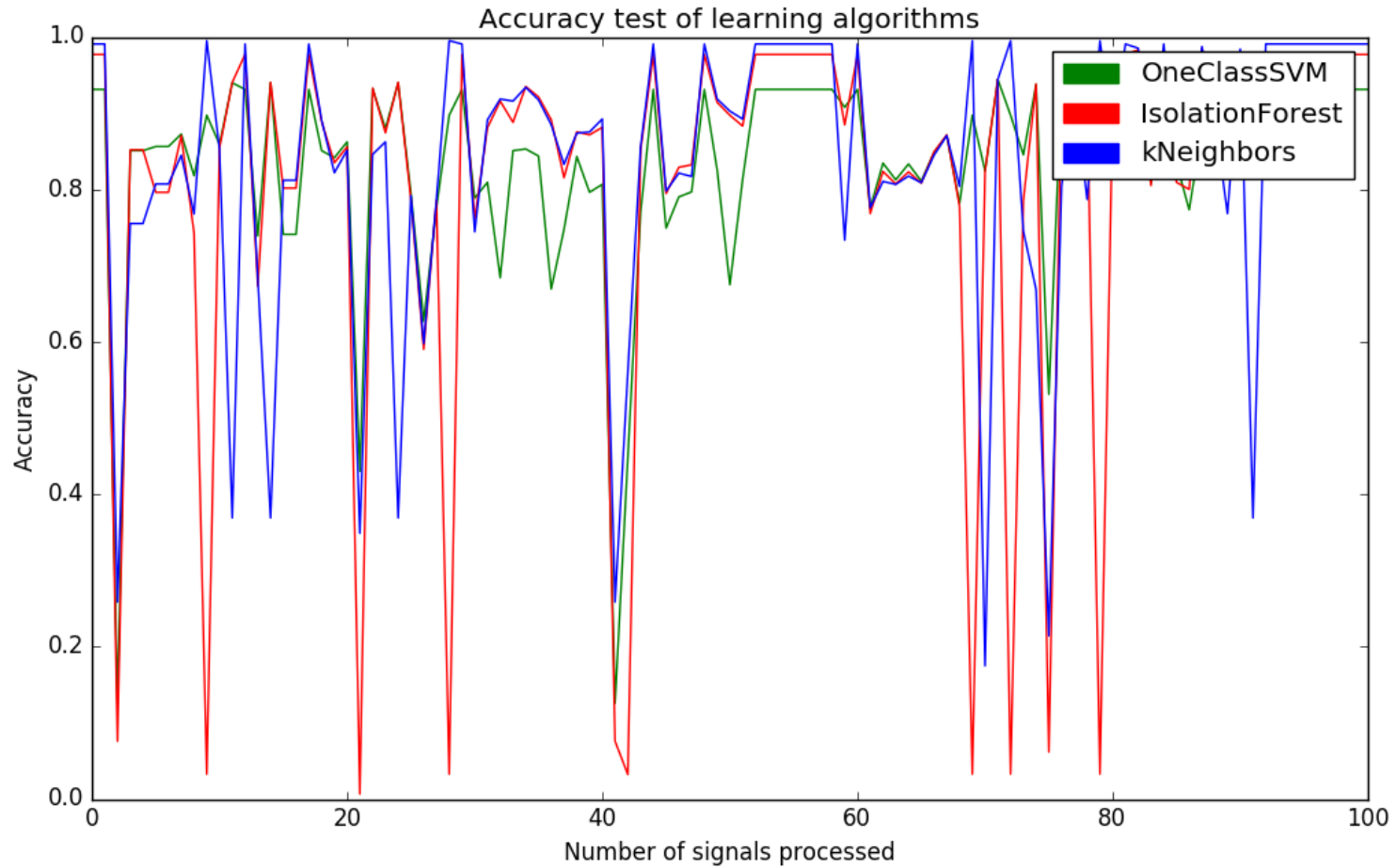
tien.vu@tum.de
www.matthes.in.tum.de



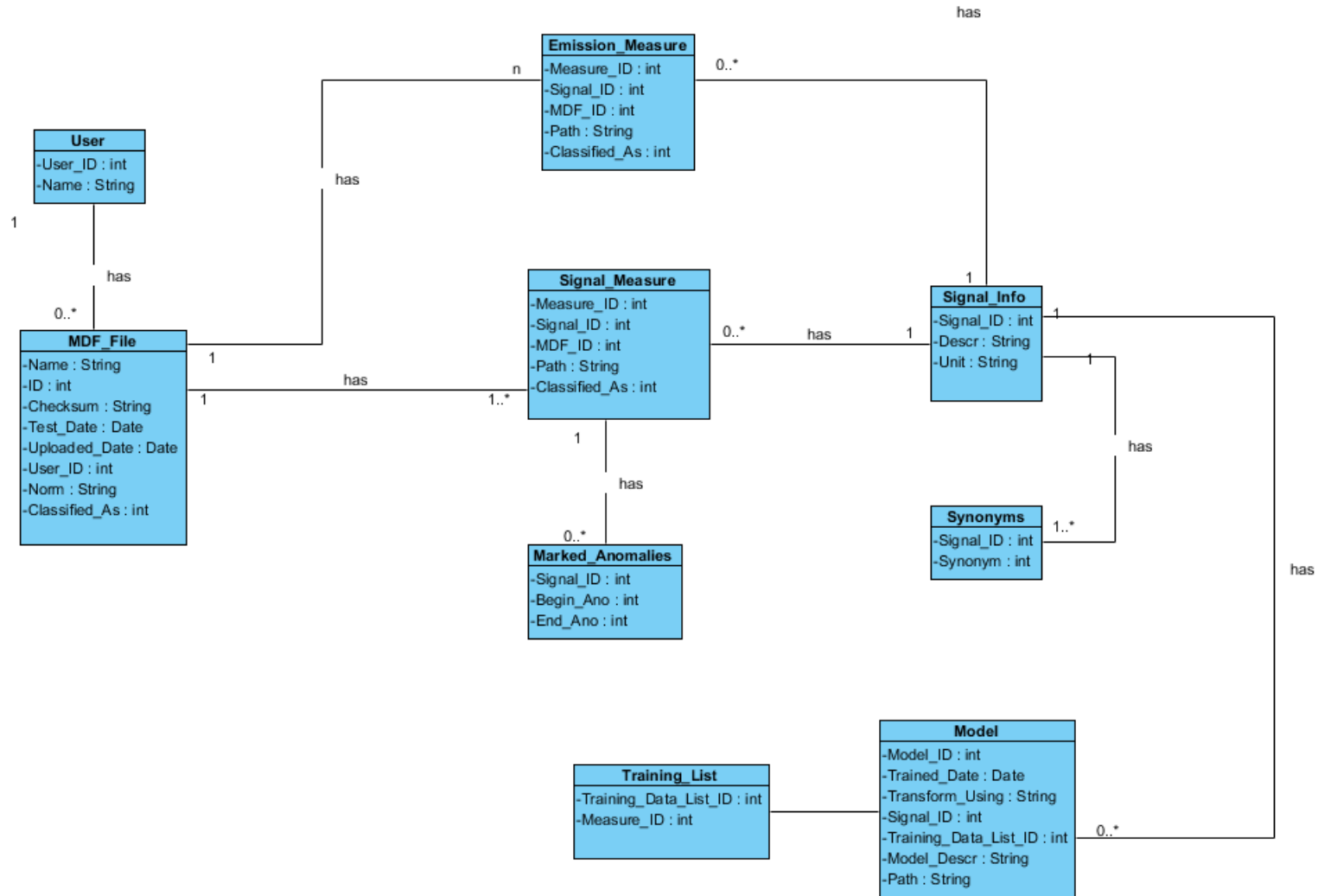
Backup slides



Assessment of Anomaly Detection Techniques

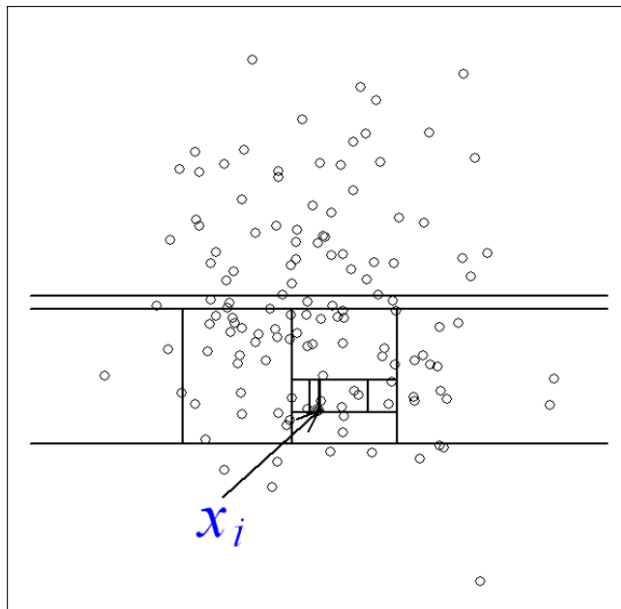


The Data Model

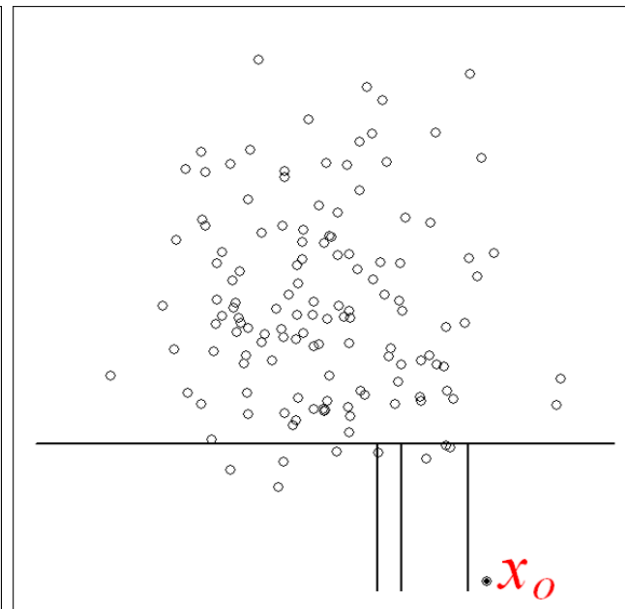


IsolationForest

- Building a forest of random split – a collection of binary tree
- The root of each tree in the forest is the original data set
- Each tree will choose randomly a feature of the data set and split it at a random point, divides the data set in two new data set
- The anomaly score of each data point is calculated by taking average of the length of each path from root to that point in all trees. Data points with lower score than normal are the anomalies.

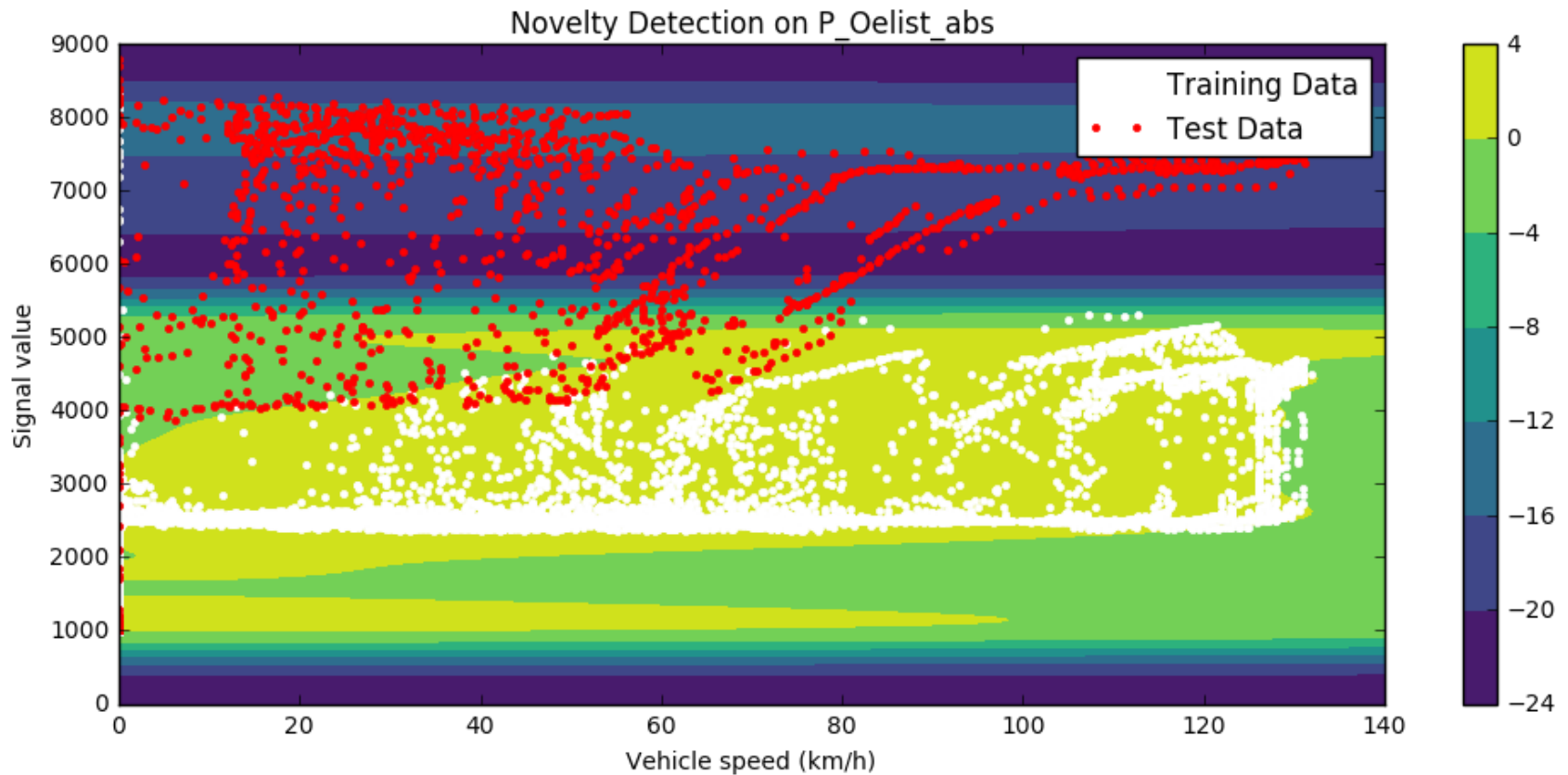


(a) Isolating x_i



(b) Isolating x_o

- Create a hyperplane that separate training data point from the rest of space
- The hyperplane form a region from the training data point where all data points outside of that region is defined as anomaly



- For each test data point:
 - Get k closest neighbors
 - Find maximum distance between those k neighbors d_{Max}
 - Find minimum distance between test data point and the k neighbors d_{test}
 - Test:
 - If $d_{test} < d_{Max}$ -> Normal data point
 - Else : Anomaly

Challenges

- Long training /evaluating time
 - Depend on algorithm, 1-9 seconds / Signal ~ 900 Signals
- Data contains noise
- The tests and signals have different resolution (1 millisecond – 1 second)
- Limited number from data sets because each emission test is very costly

Challenges

- Traditional time series analysis is only used for curve fitting / prediction problem
- Different types of time series require different algorithm
- It's hard to correctly aggregate results from different algorithm

