Outlier Detection – Anomaly Detection

(Data Driven Models in Engineering Problems)

Eleni I. Vlahogianni,

Associate Professor NTUA
Traffic Engineering Laboratory
Department of Transportation Planning and Engineering
School of Civil Engineering

ε.δε.μ² ΕΠΙΣΤΗΜΗ ΔΕΔΟΜΕΝΩΝ & ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Definition of an Outlier

"An outlier is an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism" [Hawkins 1980]

Intuition (Statistics-based)

- normal data objects follow a "generating mechanism", e.g. some given statistical process, abnormal objects deviate from this generating mechanism

- Why outliers (and their detection) are important? *One person's noise could be another person's signal."*

Applications

- **Fraud detection**
  - Purchasing behavior of a credit card owner usually changes when the card is stolen card is stolen
  - Abnormal buying patterns can characterize credit card abuse
- **Medicine**
  - Unusual symptoms or test results may indicate potential health problems of a patient
- **Security and Surveillance**
  - abnormal motion detection in a video scene
- **Traffic Operations and Safety**
  - Abnormal speed distribution, or time streams of vehicle traffic/crowd flow data may indicate and accident or a non recurrent congestion, or crowd panic conditions
- **Detecting measurement errors**
  - Data derived from sensors (e.g. in a given scientific experiment) may contain measurement errors
  - Abnormal values could provide an indication of a measurement error
  - Removing such errors can be important in other data mining and data analysis tasks analysis tasks

Anomaly detection: an example

Users' mobility features:

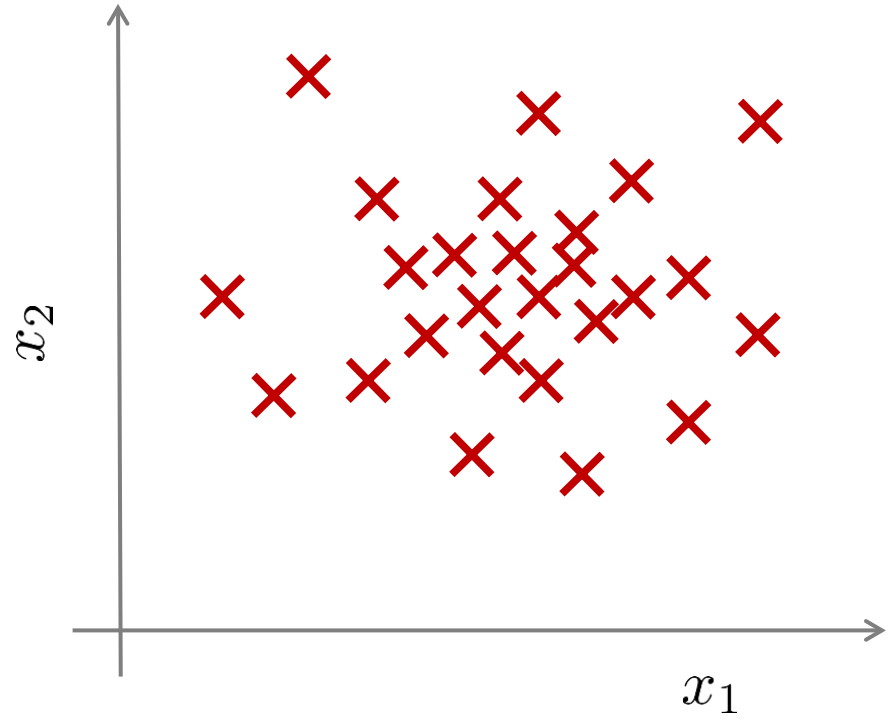
x_1 = trips/day

x_2 = duration between trips

…

Dataset: $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$

New user: x_{test}



Anomaly Detection:

Do we have access to labeled data (including anomalous cases)?

- Enough anomalous cases? supervised learning approach (dataset division into three parts!)
- Very few of no knowledge of anomaly: Unsupervised learning approach (very limited labeled data could be critical!) – How to proceed?

  Training (normal cases e.g. 60% of cases) – CV (normal cases e.g. 20% of cases , 50% of cases with anomaly) – Test (normal cases e.g. 20% of cases , 50% of cases with anomaly)

Some remarks

– Evaluation based on classification matrix…
  Precision/Recall and F_1–score

What features to choose?

Accuracy is not a good measure for imbalanced datasets!!!

Oversampling, Undersampling, Cost Sensitive Learning

Types of Anomaly

Point anomalies (a point that significantly deviates form others)

Contextual anomalies (anomaly within a context)

Collective anomalies (a collection of points are rare)

- Requires structure (temporal, spatial etc)

Novelty vs Outlier Detection

"**novelty detection**”: your data set contains only good data, **and** you're trying to determine whether new observations fit within the existing data set

"**outlier detection**", your data set may contain **outliers**, which you want to identify.

Unsupervised Anomaly Detection

No labels assumed

Assumption: anomalies are very rare compared to normal data

General Approach

- Build a profile of "normal" behavior (univariate or multivariate thinking)
- Use the "normal" profile to detect anomalies (observations whose characteristics differ significantly from the normal profile)

Methods

- Statistical (model based)
- Distance based
- Density based
- Clustering
- other

Statistical Based Approaches

Outliers are objects that are fit poorly by a statistical model.

- Estimate a parametric model describing the distribution of the data
- Apply a statistical test that depends on

  Properties of test instance ,

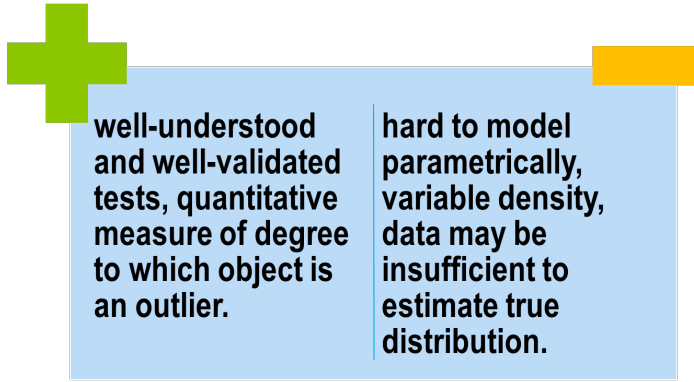
  Parameters of model (e.g., mean, variance)

  Confidence limit (related to number of expected outliers)

Multivariate Gaussian distribution – Outlier defined by Mahalanobis distance > threshold

Grubbs' test (for univariate data, H 0: There is no outlier in data)

- Assume data comes from normal distribution
- Detects one outlier at a time, remove the outlier, and repeat



Distance-based outlier detection

Outliers are objects far away from other objects.

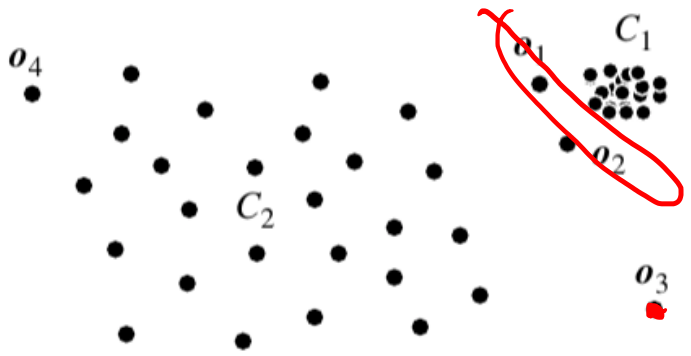
- the greater is the distance of the object to its neighbors, the more likely it is an outlier

Approach:

- Outlier score is distance to kth nearest neighbor.
- Score sensitive to choice of k.

Examples:

- KNN (Outlier Detection based on the distance of an object to its k nearest neighbor)



Easier to define a proximity measure for a dataset than determine its statistical distribution, quantitative measure of degree to which object is an outlier, deals naturally with multivariate data.	$O(n^2)$ complexity, score sensitive to choice of k, does not work well if data has widely variable density.

Density-based outlier detection

Outliers are objects in regions of low density.

Examples:

- LOF (Local Outlier Factor) – local density deviation of a given data point with respect to the data points near it
- locality is given by k nearest neighbors
- the reachability distance $reachdist_k(o, o') = max[dist_k(o), dist(o, o')]$
- Local Reachability Density (the inverse of the sum of all of the reachability distances of all the k–nearest neighboring points
- Score LOR: the ratio of the average of the lrds of k number of neighbors of a point and the lrd of that point

  LOF(k) ~ 1 means Similar density as neighbors,
  LOF(k) < 1 means Higher density than neighbors (Inlier),
  LOF(k) > 1 means Lower density than neighbors (Outlier)

Quantitative measure of degree to which object is an outlier, can work well even if data has variable density	$O(n^2)$ complexity, must choose parameters k for nearest neighbor d for distance threshold

Cluster-based outlier detection

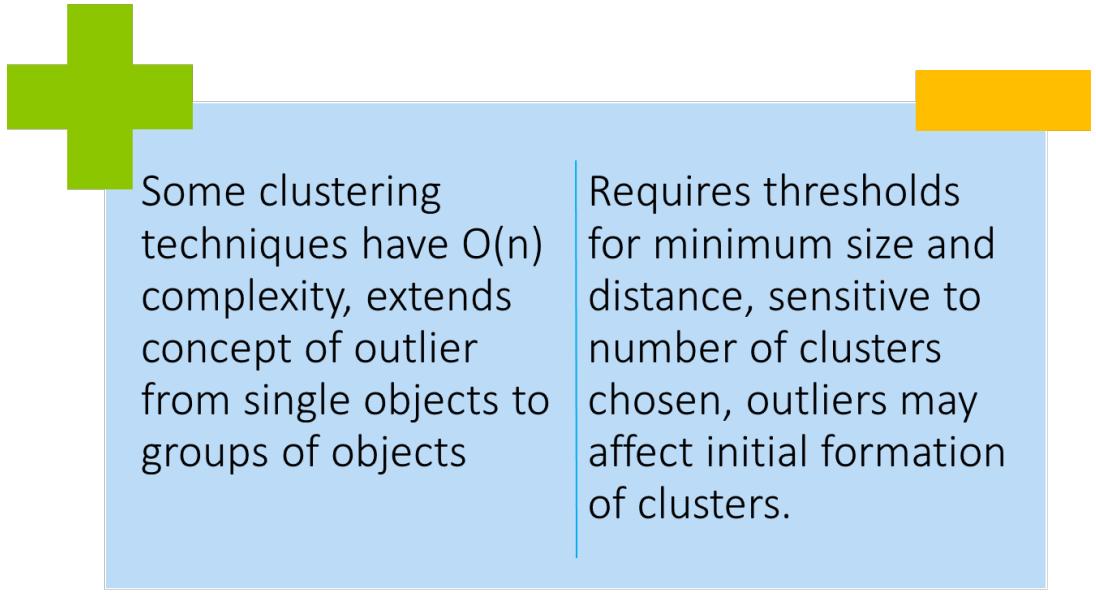
Outliers are objects that do not belong strongly to any cluster.

Approach:

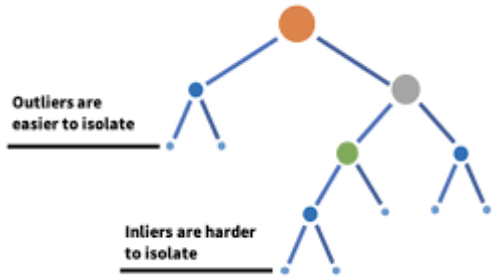
- Assess degree to which object belongs to any cluster
- Eliminate object(s) to improve objective function
- Discard small clusters far from other clusters.

Examples:

- EMOutlier
- KMeansOutlierDetection



Other approaches

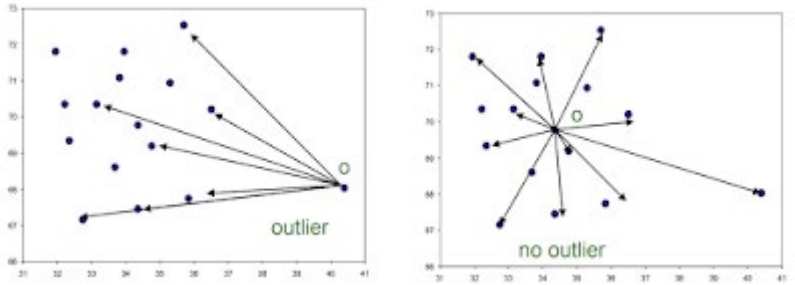


Tree-based outlier detection

- Isolation forest (identifies anomalies by isolating outliers in the data) – Solitude package

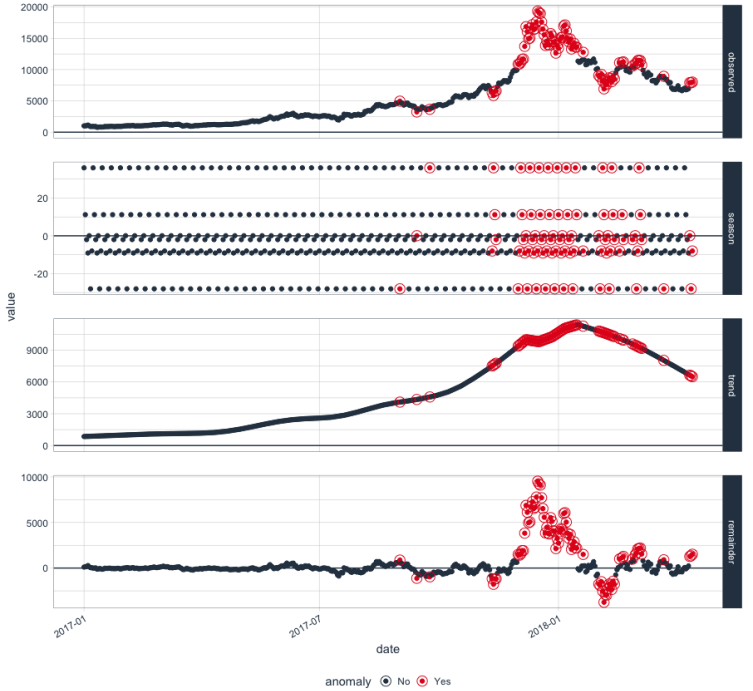
Angle-based outlier detection

- especially useful for high-dimensional data, as angle is a more robust measure than distance in high-dimensional space – abodOutlier package



Time series anomaly detection

- Time series decomposition – anomalize package
- Error reconstruction via time series modeling (including deep learning/recurrent nn)



Graph & Network Outlier Detection

- Outliers (in large graphs) can be portions of the network, which might be nodes, edges, or even subgraphs

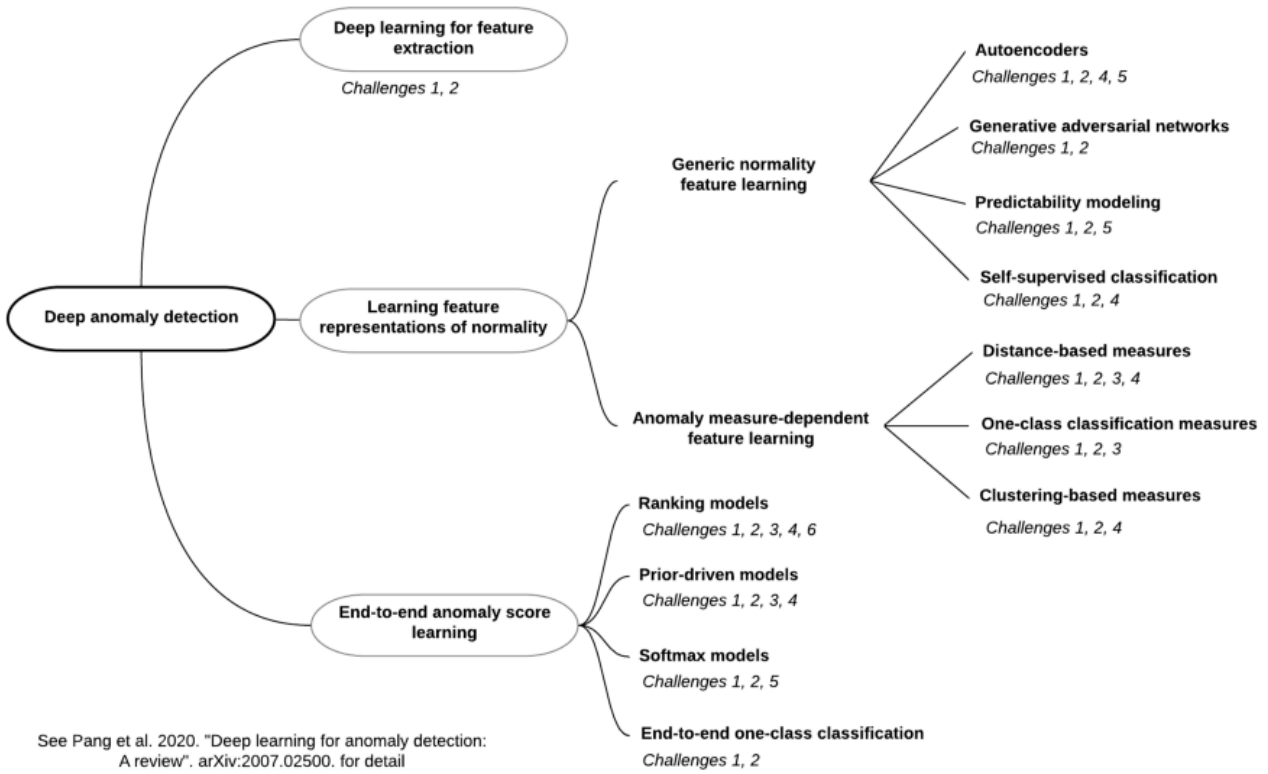
https://github.com/yzhao062/anomaly-detection-resources

https://github.com/pridiltal/ctv-AnomalyDetection

Other approaches

Deep learning for anomaly detection

- difficulty to achieve high anomaly detection recall rate (Challenge #1)
- Anomaly detection in high-dimensior and/or not-independent data (Challenge #2)
- data-efficient learning of normality/abnormality (Challenge #3)
- noise-resilient anomaly detection (Challenge #4)
- detection of those complex anomalie (Challenge #5)
- anomaly explanation (Challenge #6)



Guansong Pang, Chunhua Shen, Longbing Cao, Anton van den Hengel. “Deep Learning for Anomaly Detection: A Review”. 2020. arXiv preprint: 2007.02500.

Smartphone based driving and mobility analytics (your example!)

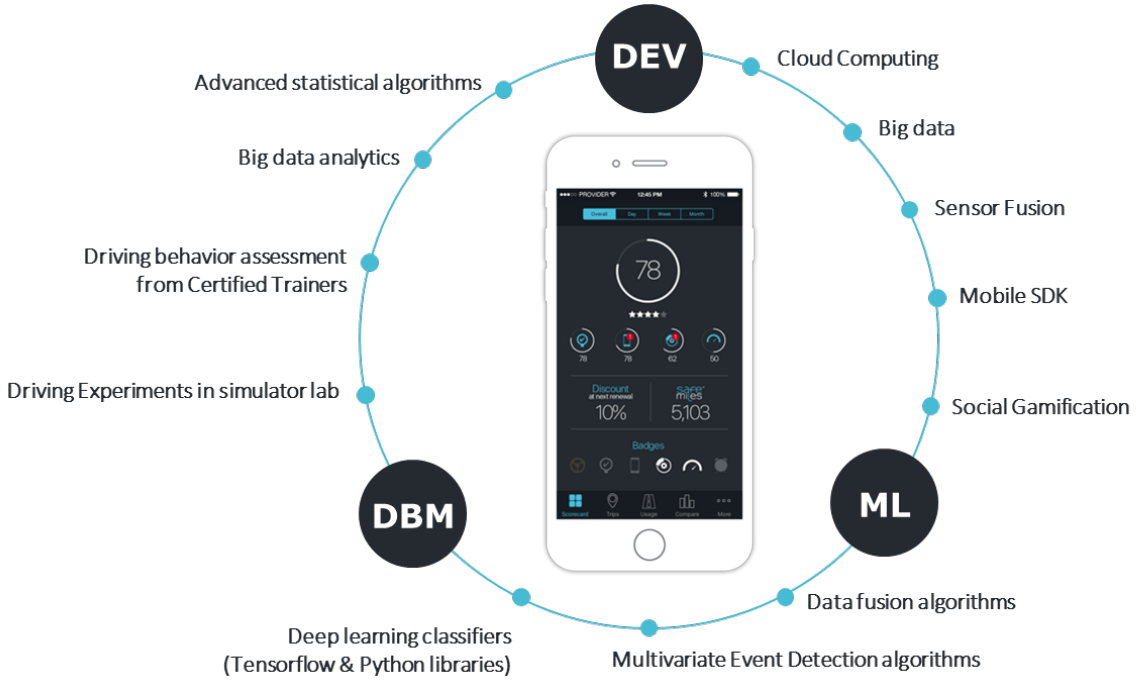
Usefulness to

- Customized pricing policies (Pay-As-You-Drive, Pay-How-You-Drive)
- Raising the awareness of the "efficient" mobility profile
- Identifying the actual human behavior

The constraint: Deliver a product that is sustainable

- Battery consumption
- Accuracy and scalability
- Address industry questions
- Useful and interesting to users

Smartphone based driving and mobility analytics



"See your trip details where you were wrong, improve your driving behavior and be rewarded"

Harsh events detection

Braking Accelerating Cornering

Driver distraction

Clear "Noise" / in car activity

Driver ID recognition

Driver / Passenger / Mass Transit recognition

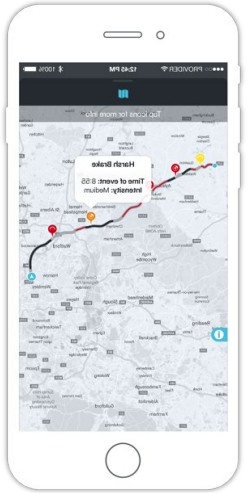
Detection, Modeling and Policy Questions

Detection problems

- Is it a car trip?
- Who drives the car?
- Is it a harsh event?
- Are you distracted?

Modeling and Policy

- How long should I monitor you to know your overall driving behavior?
- How can I develop customized driving policies?
- Do specific user profiles exist?
- Can I use driving analytics for large scale network management?



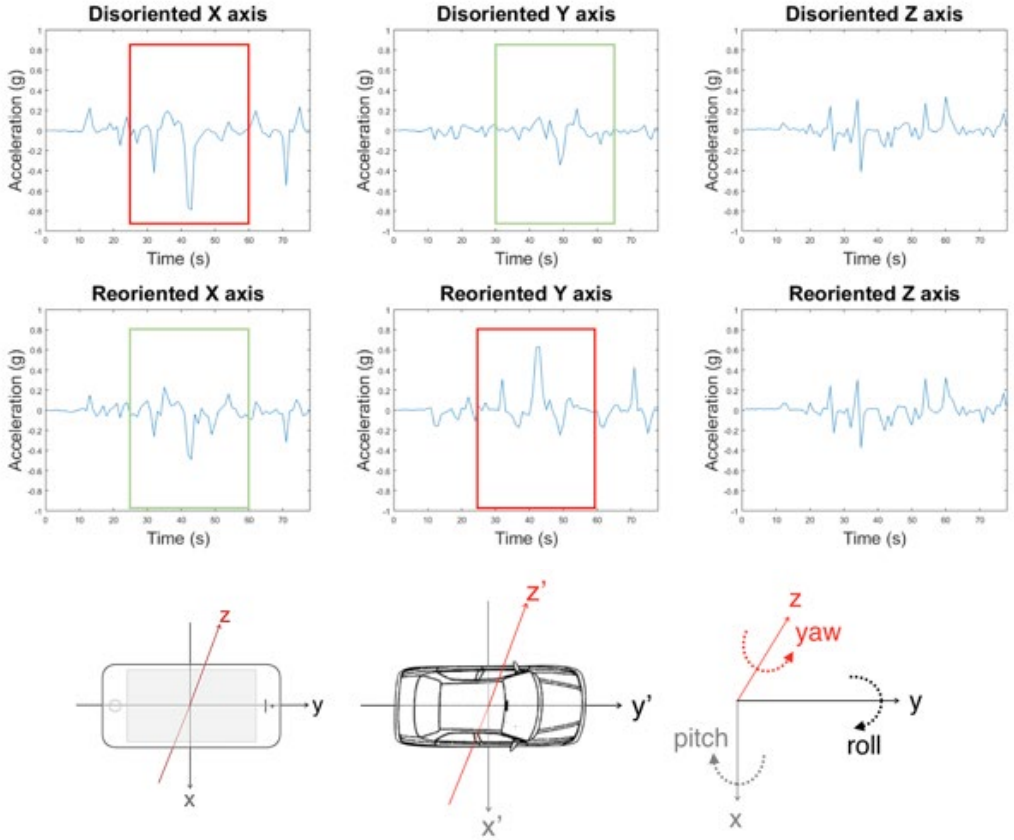
Data Specs and Preparation

Data resolution

- strong dependence on the type of application

Data preparation

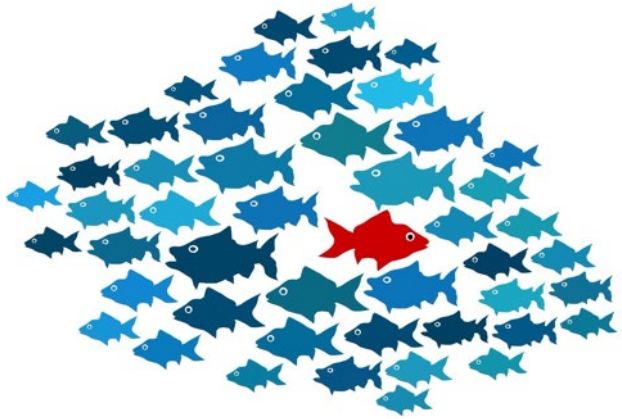
- Device Orientation
- Activity (Walking, Standing etc.)
- Erroneous Values
- etc



Detecting Harsh Driving Events

Uncertainties

- The smartphone's position is highly uncertain (on the deck, on the pocket, inside bags etc)
- Different types of smartphones (iOS, android, old, new sensors etc)
- Noisy signals

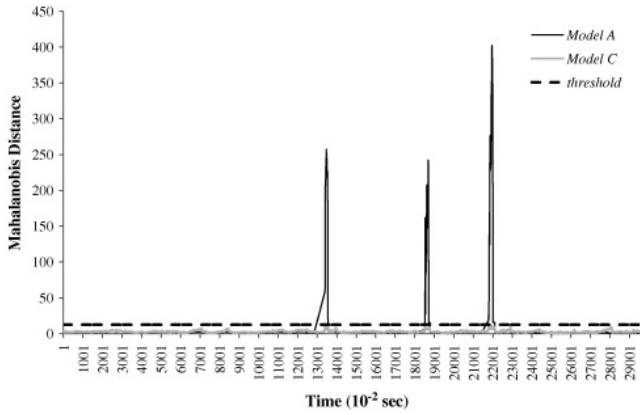


Per Trip Solution → A (rather) Simple Outlier Detection Strategy

- We can define the basic driving pattern and quantify the divergence from it
- We fuse data from other sensors to evaluate the detection (e.g. use GPS data to evaluate acceleration and deceleration patterns)
- The process results to a slightly varying set to thresholds for every trip

Harsh Accelerations Harsh Brakes

Harsh Cornerings



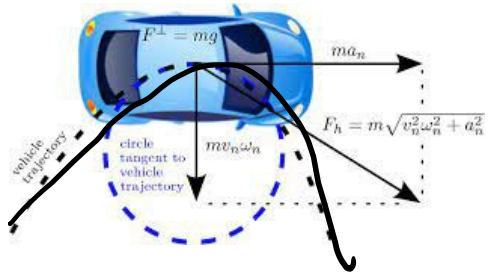
Other approaches?

Example I : harsh event detection (unsupervised approach)

The dataset: driving data from n drivers collected using the accelerometer, gyroscope and GPS of the smartphone

NewAccelX	NewAccelY	NewAccelZ	NewRotRateX	NewRotRateY	NewRotRateZ	locationSpeed
0.001	-0.002	-0.007	0.002	-0.001	0	0
0.007	0	0.011	-0.001	0.001	0	0
0.008	0.004	0.015	0	0	-0.001	0
0.008	0.002	0.004	0.002	-0.001	-0.001	0
0.009	0.001	0.026	-0.002	-0.001	0.002	0
0.012	-0.001	0.003	0	0	-0.001	0
0.007	0	0.004	-0.001	0	0	0
-0.008	-0.006	0.002	0.001	-0.001	0	0
-0.004	0.005	0.022	0	0.001	0.001	0
0.007	0.007	0.016	0	-0.002	0.003	0
0.013	0.011	0.019	-0.001	0	0.001	0
0.013	0.002	0.011	0.001	0.002	-0.002	0
-0.008	-0.003	-0.002	0.002	-0.003	-0.001	0
-0.043	-0.005	0.02	0.004	-0.018	0	0
-0.008	-0.005	-0.002	0.001	-0.002	-0.002	0
0.009	-0.038	0.019	0.002	-0.002	0.001	0
0.028	0.009	0.021	0.004	-0.01	0.014	0.42
-0.013	0.004	-0.07	0.004	0.011	0.006	0
0.017	-0.013	0.009	-0.002	-0.005	0.033	0
0.042	-0.017	-0.018	-0.01	-0.01	0.01	0
-0.039	0.026	-0.008	0.004	-0.004	0.007	0.45
0.028	0.017	0.021	0.002	0.029	0.016	0.45

Detect harsh cornerings



The data are confidential!

Submit:

- a script
- a technical report (less than 10 pages)
- csv with the detected cornerings

Next course

Forecasting