Anomaly Detection and Failure Prediction

Prof. Evgeny Burnaev, e.burnaev@skoltech.ru
Skoltech, 2021

Skoltech
Skolkovo Institute of Science and Technology

ADASE group

- 30 researchers
- DL for
  - 3D Computer Vision
  - Predictive Analytics

~ 100 **papers** in major venues, incl. NIPS, ICML, CVPR, etc.

The Best Paper Award for the research on modeling of point clouds and predicting properties of 3D shapes at the Int. Workshop on Artificial Neural Networks in Pattern Recognition (ANNPR), 2020

Geometry Processing Dataset Award for the work «ABC Dataset: A Big CAD Model Dataset For Geometric Deep Learning», Symposium on Geometry Processing, 2019

The Best Paper Award for the research in eSports at the IEEE Internet of People conference, 2019

ADASE group

- 30 researchers
- DL for
  - 3D Computer Vision
  - Predictive Analytics

The Best Paper Award for the research on modeling of point clouds and predicting properties of 3D shapes at the Int. Workshop on Artificial Neural Networks in Pattern Recognition (ANNPR), 2020

Geometry Processing Dataset Award for the work «ABC Dataset: A Big CAD Model Dataset For Geometric Deep Learning», Symposium on Geometry Processing, 2019

The Best Paper Award for the research in eSports at the IEEE Internet of People conference, 2019

~ 100 **papers** in major venues, incl. NIPS, ICML, CVPR, etc.

Moscow government prize for Scientific Achievements, 2018

“Ilya Segalovich” Yandex prize for Scientific Achievements, 2020

Industrial Expertise: since 2007

AIRBUS, LOUIS VUITTON, AIRBUS HELICOPTERS, IPSEN Innovation for patient care, ALTRAN, SAHARA Force India formula one team, PORSCHE, SBERBANK, HUAWEI, Alibaba Group 阿里巴巴集团, BOSCH

and many others...

Table of contents

Table of contents

What data are already being collected [6]

Heat & power plant - 10^3 observations/ms

Rolled metal production - 10^4 observations/ms

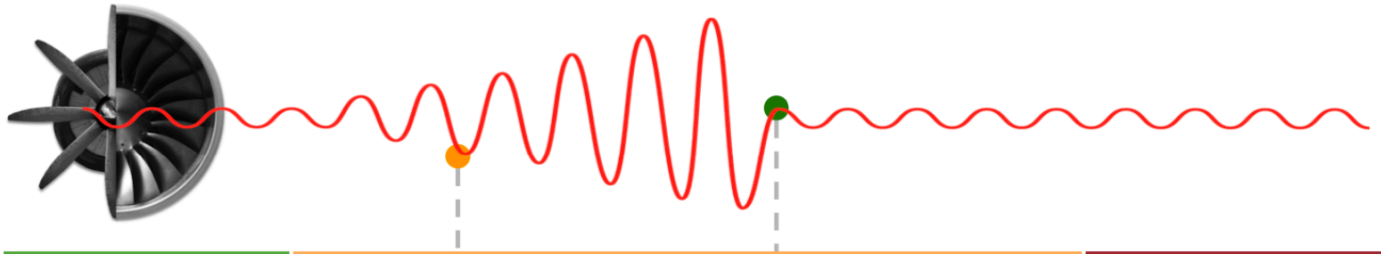
Modern aircraft – up to **0.5 Tb for one flight**

- Less than 1% of data is used, most of the data is not stored and used
- The next generation of Pratt & Whitney engines will produce up to **10 Gb/sec**

Self-driving Google car – up to **1 Gb/sec**

- Formula-1 car - **1.2 Gb/sec**

PHM concept



Exploitation	Time to Failure					Failure
Years/months	**Months**		**Weeks**	**Days**		
Optimal condition	Vibrations	Signs of wear	Decrease of main KPIs	Noise	Overheating	Failure (repair or a full replacement)
Intelligent Monitoring and Predictive Maintenance	First detection of a degradation		Corrective Maintenance			Failure is prevented
Other approaches to Maintenance	Scheduled Maintenance. Collected data is not used					Emergency Repair

PHM $\Rightarrow$ New service models

Selection of Maintenance Strategy

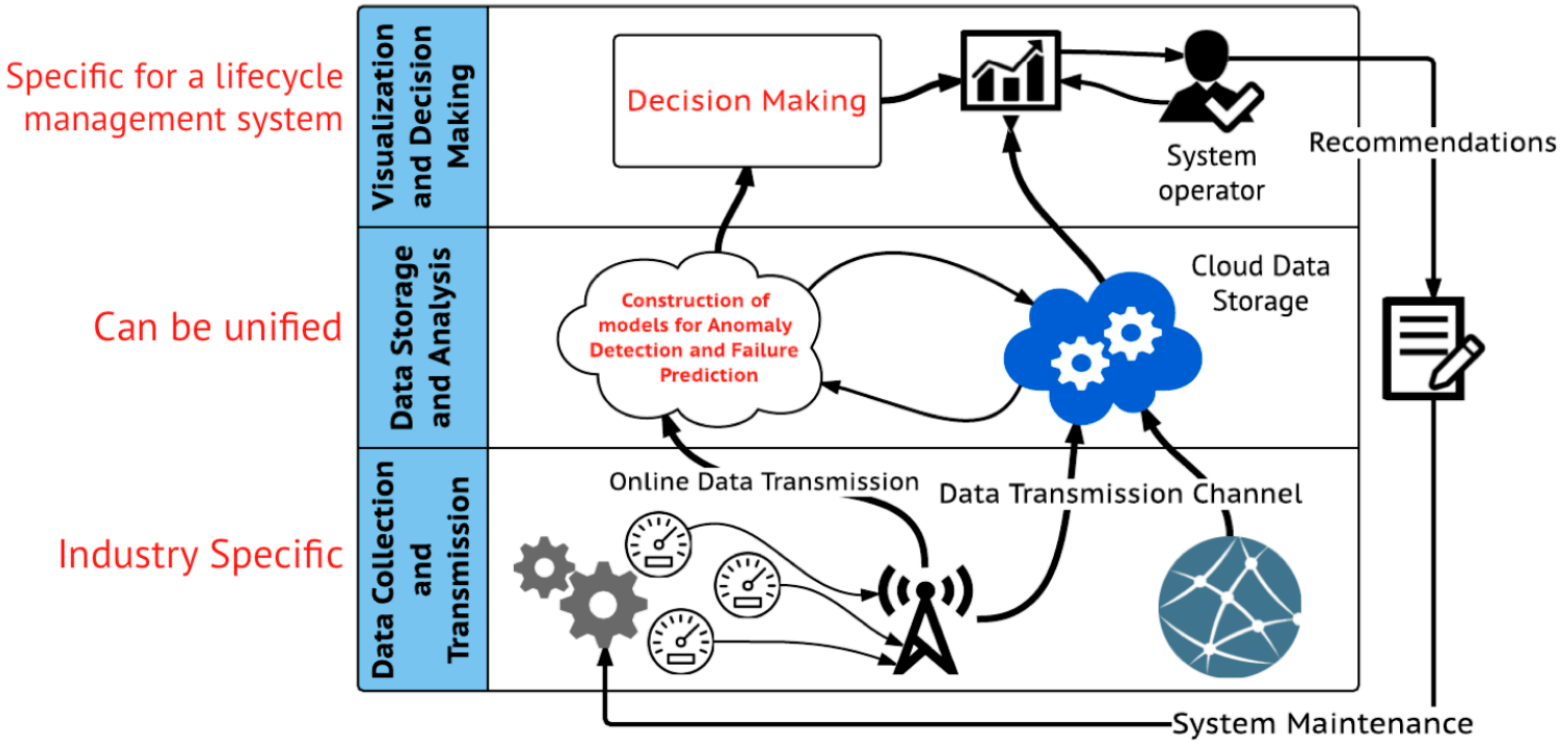
- Often when selecting a strategy it is necessary to optimize multiple targets
- Data Analytics can help to balance the “contradicting” targets

Example from Aviation

When performing technical maintenance it is necessary to

- Increase the availability of an aircraft
- Reduce maintenance costs
- Minimize a number of operational interruptions
- Increase safety

Intelligent Maintenance System



Intelligent Maintenance System

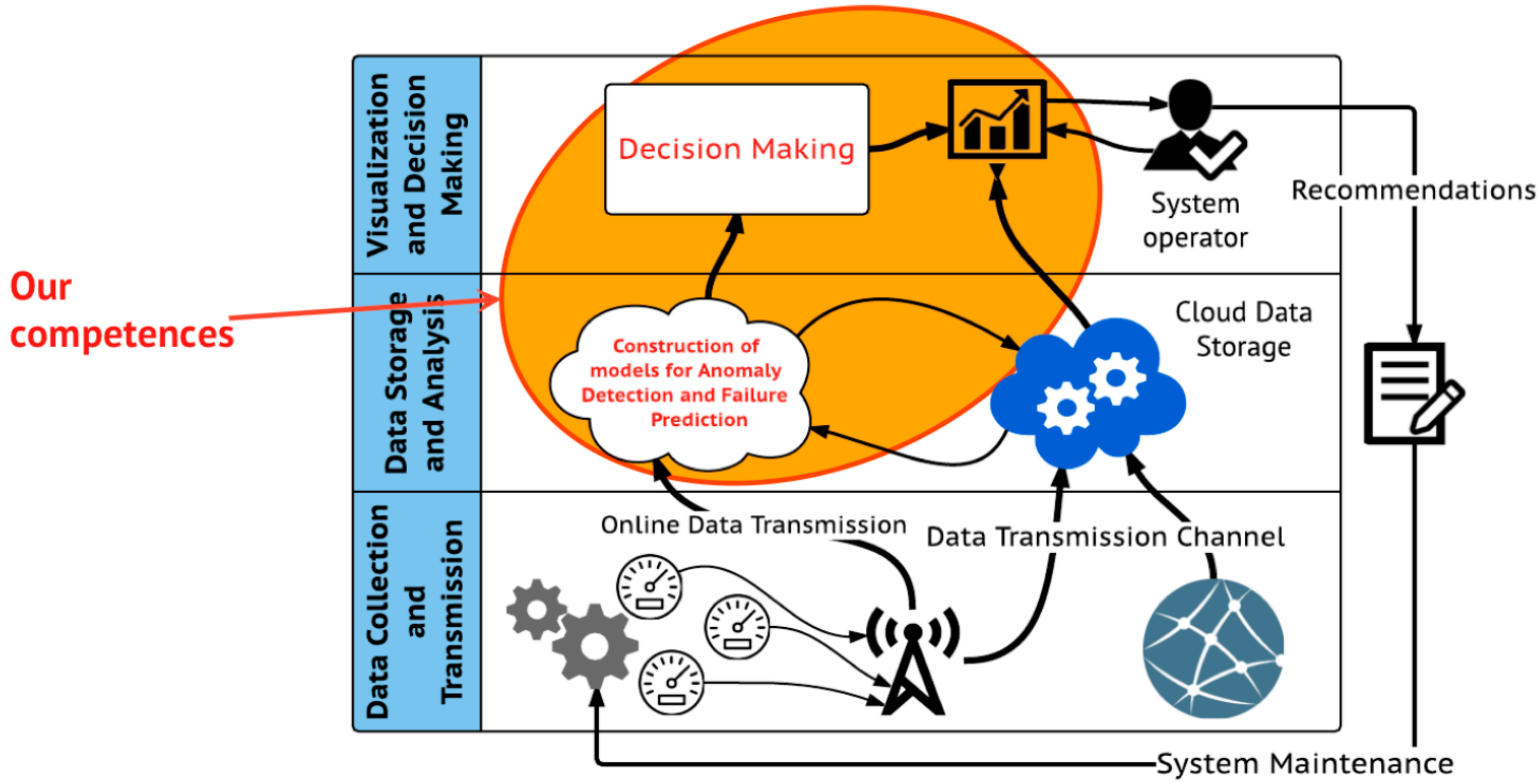
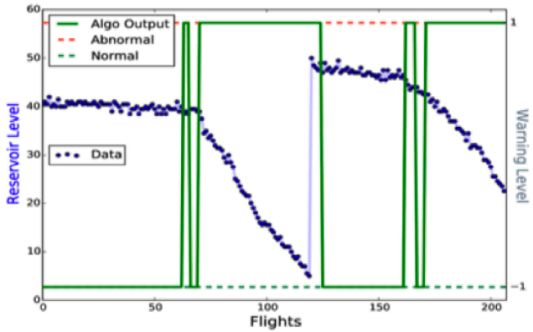
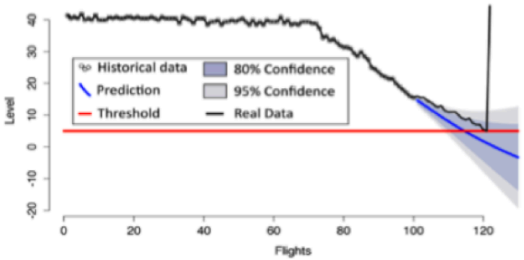


Table of contents

Aircraft Engine Cooling System

- **Objective:** optimize maintenance of the cooling system
- **Subtasks:**
  - Determine whether a current refrigerant level is critical
  - Quick leakage detection
  - Prediction of time until achieving a critical level

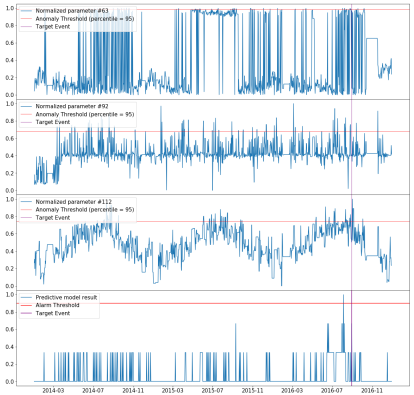




- **Data:**
  - Time-series of levels of refrigerant
  - 17 aircrafts, ~ 400 flights of each
- **Results:**
  - False positive rate is $< 1\%$
  - Rate of correct detection is $> 99\%$
  - Average error of prediction before 10 flights until achieving a critical level is < 1 flight

In collaboration with Datadvance llc.

Auxiliary Power Unit Failures [5, 6]



- **Problem:**
  - ✓ **Input:** multidimensional telemetry data (high/low pressure turbine rotor speed and vibration; burner pressure, exhaust, fuel and oil feed, etc.)
  - ✓ **Output:** events (APU failures)

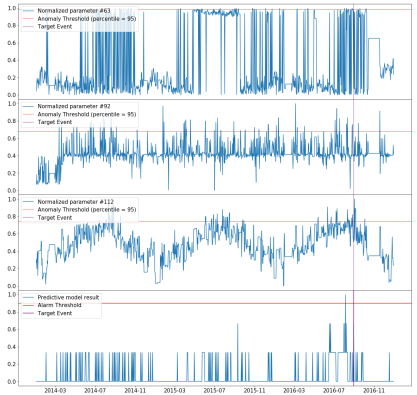
- **Data:** 3 years, ~400 flights per year of telemetry and observed failures for 30 aircrafts, 400+ parameters

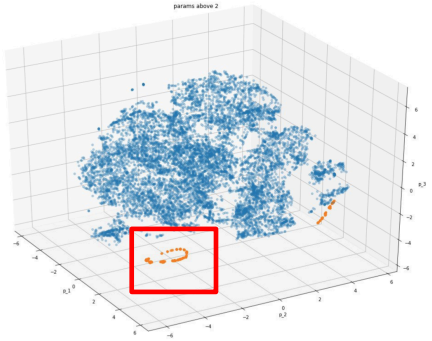
- **Objective:** predict future failures with low false alarms

- **Challenges:**
  - ✓ Heterogeneous data and noise,
  - ✓ Large volumes of high-dimensional data,
  - ✓ Imbalanced learning data sample (events are rare)

Auxiliary Power Unit Failures [5,6]

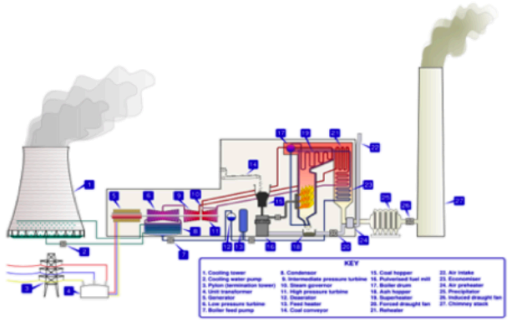
- **Solution and results:**
  - ✓ Anomaly Detection and Early warnings about some types of failures
  - ✓ High coverage rates (detected failures)
  - ✓ Low False Alarms (for ~9 accurately predicted failures we get ~1 false alarm)

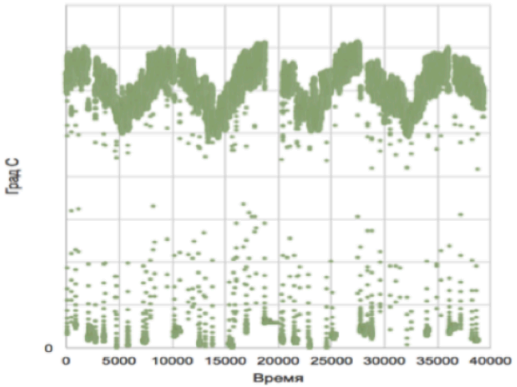




Power Losses in Thermal Power Plant

- **Objective**:
  1. Detect Power Losses in the system
  2. Localize origins of power losses
- **Subtasks**:
  - Construct a model of a system in normal regime
  - Model sensitivity analysis w.r.t. a change in a system behavior (potential failures)





- **Data:** $200+$ dimensional time-series, one-observation per 10 min.
- **Results**:
  - Detection of power losses in the system
  - Localization of the power losses origins
  - Results were confirmed by experts

In collaboration with Datadvance llc.

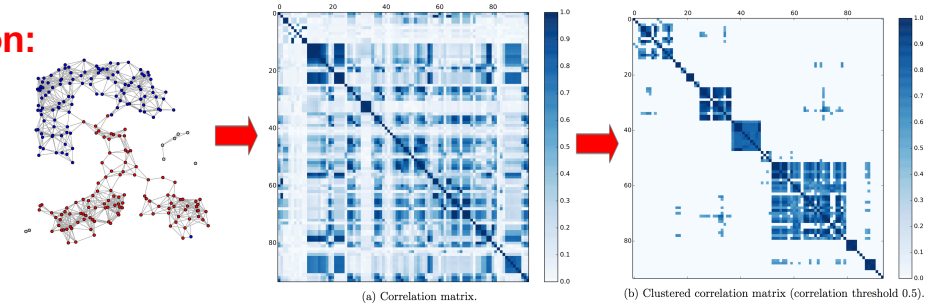
Table of contents

Challenges

- Multistream/multichannel scenarios with unstructured hypotheses/patterns
- High-dimensionality and Large data volumes
- Composite hypotheses
- Stochastic models with non-stationary and dependent observations of a very general structure
- Prior uncertainty with respect to pre- and post-change distributions
- Parametric assumptions are inefficient
- Imbalanced learning samples

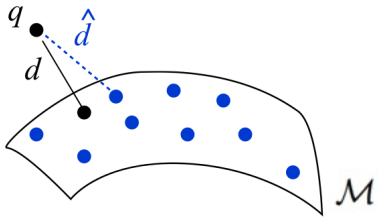
Methodology (macro-steps) [5,6]

1. Subsystems Identification:
Identification of groups of dependent parameters, corresponding to different subsystems

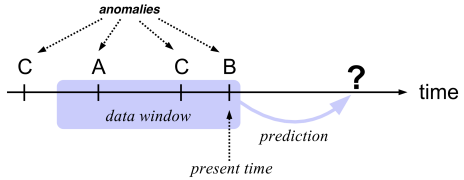

(a) Correlation matrix.

(b) Clustered correlation matrix (correlation threshold 0.5).

2. Anomaly Detection:
Detection of Anomalies based on Manifold Learning for identified subsystems



3. Events Matching:
Statistical techniques to identify best anomalies preceding warnings (and not happening anywhere else)



4. Validation:
Apply anomalies detection logics in a new airplane (left apart) to calculate the prediction

Methods. Anomaly Detection

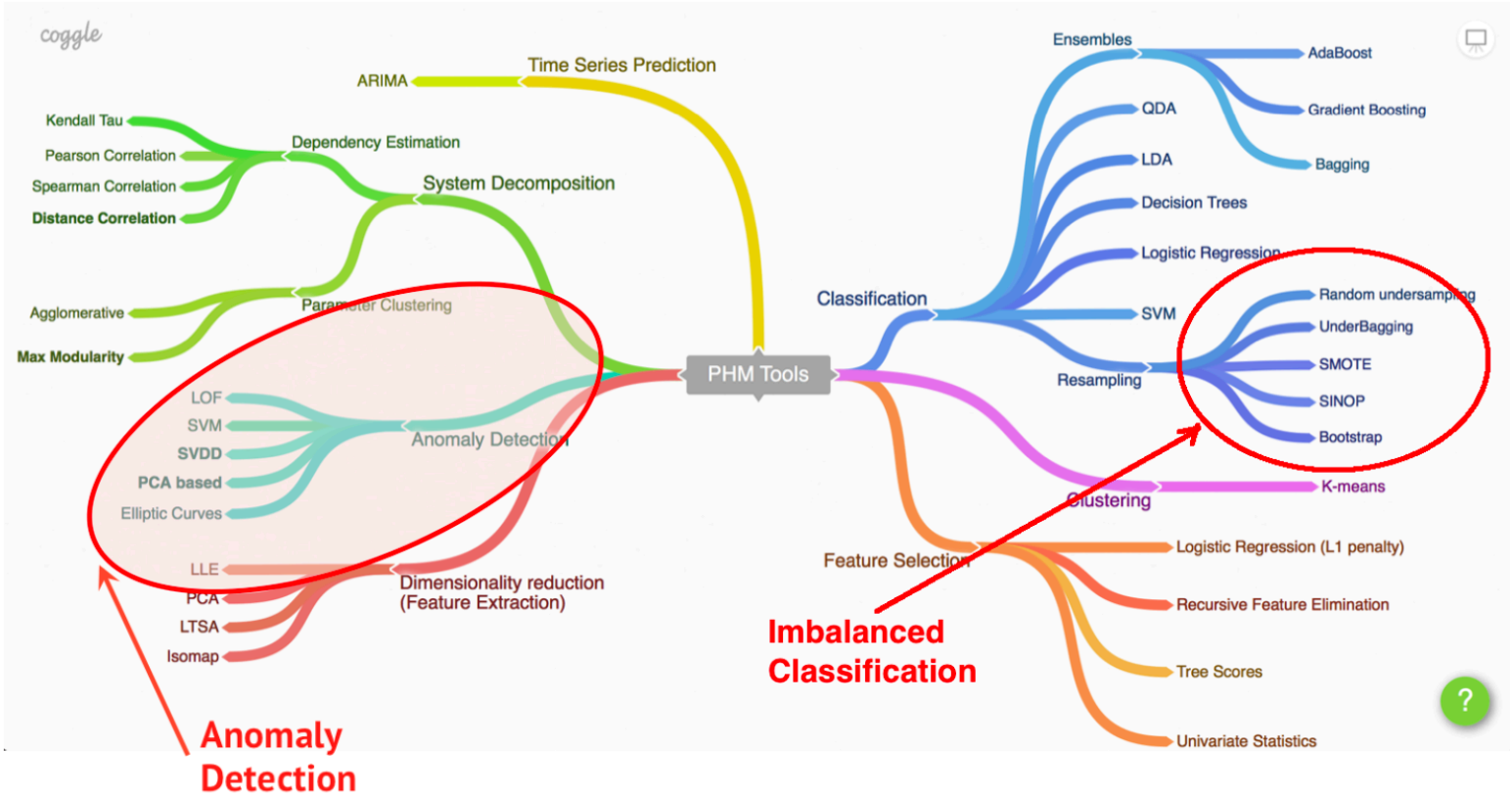
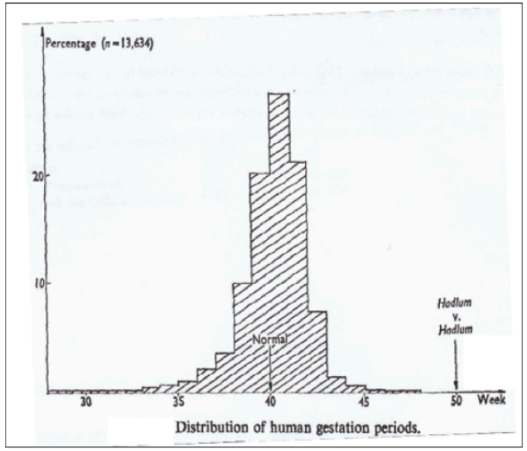
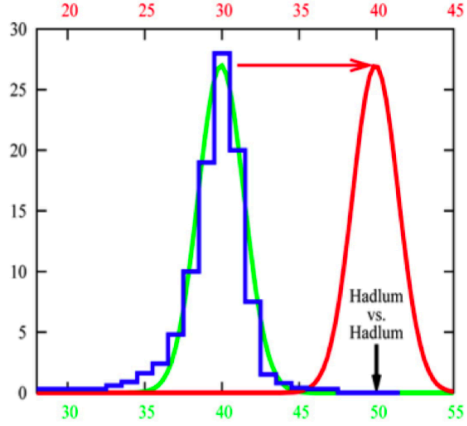


Table of contents

Hadlum vs. Hadlum (1949) [Barnett, 1978]





- The birth of a child to Mrs. Hadlum happened 349 days after Mr. Hadlum left for military service
- Average human pregnancy period is 280 days (40 weeks)
- Statistically, 349 days is an outlier

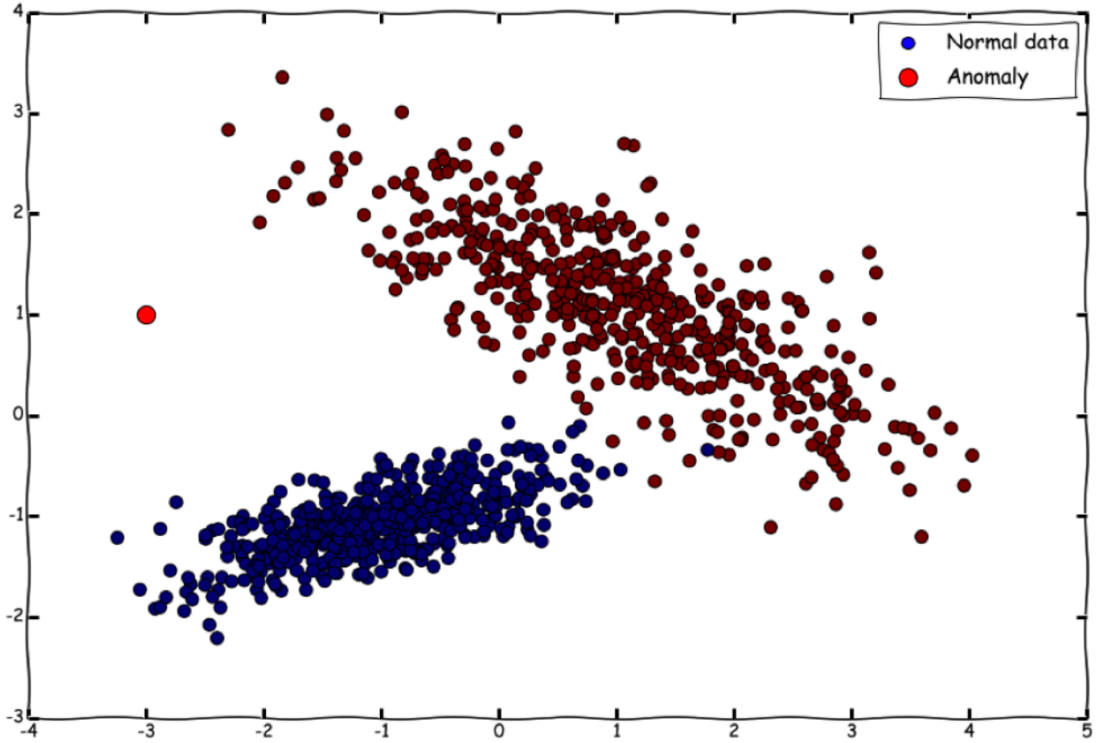
Definition of anomaly [Howkins, 1980]

"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"

In real world problems definition of anomaly depends on its context

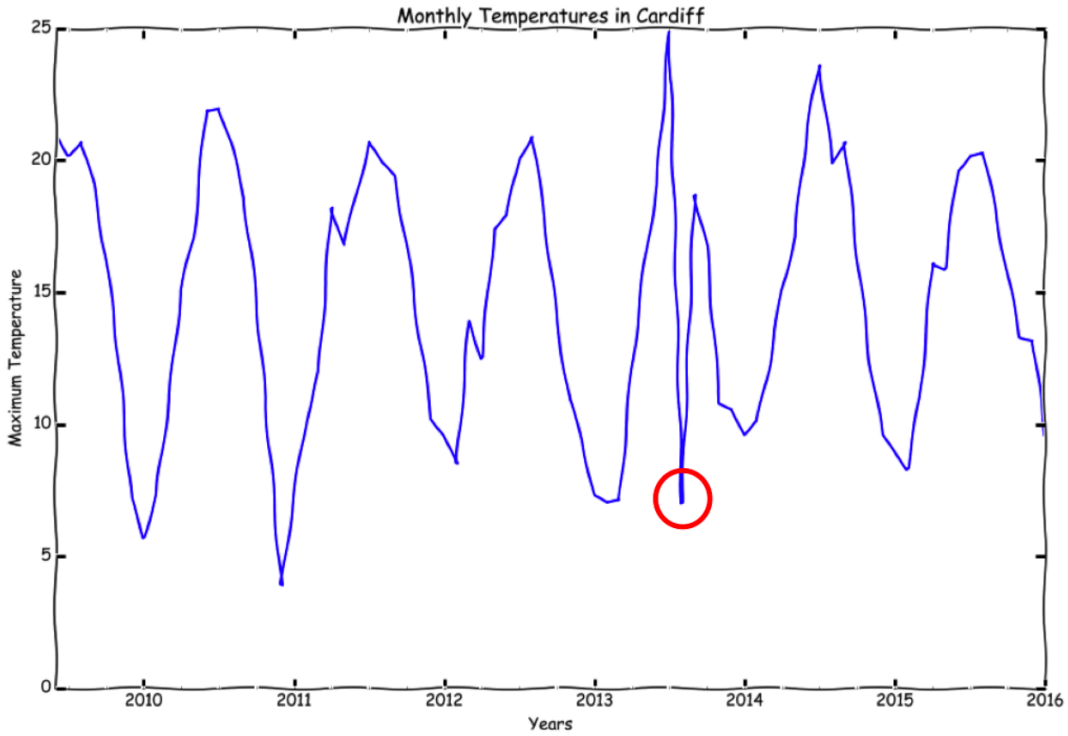
Anomaly taxonomy: Point Anomalies

- An individual data instance is anomalous w.r.t. the data



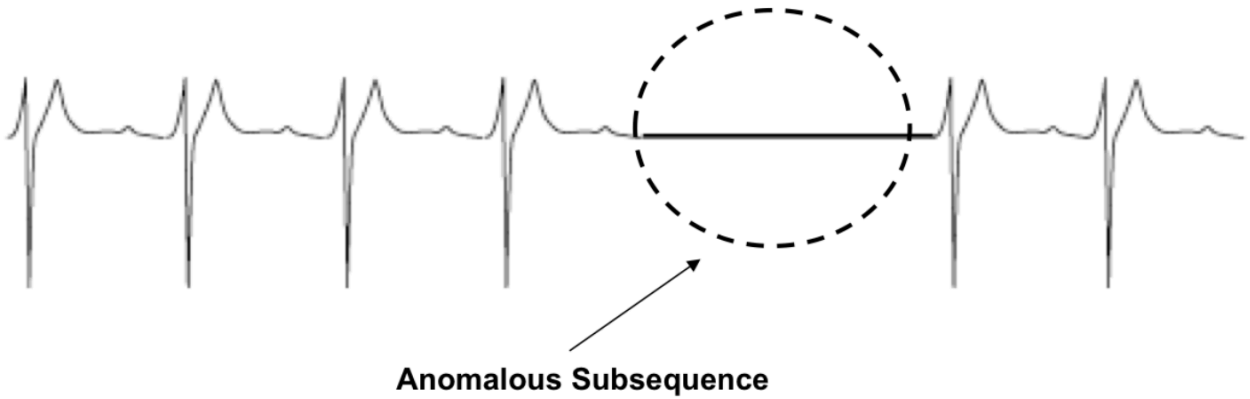
Anomaly taxonomy: Contextual Anomalies

- An individual data instance is anomalous within a context
- Requires a notion of context
- Also referred to as conditional anomalies



Anomaly taxonomy: Causal Anomalies

- A collection of related data instances is anomalous
- Requires a relationship among data instances
  - Sequential Data
  - Spatial Data
  - Graph Data
- The individual instances within a set of causal anomalies are not anomalous by themselves



Anomaly taxonomy

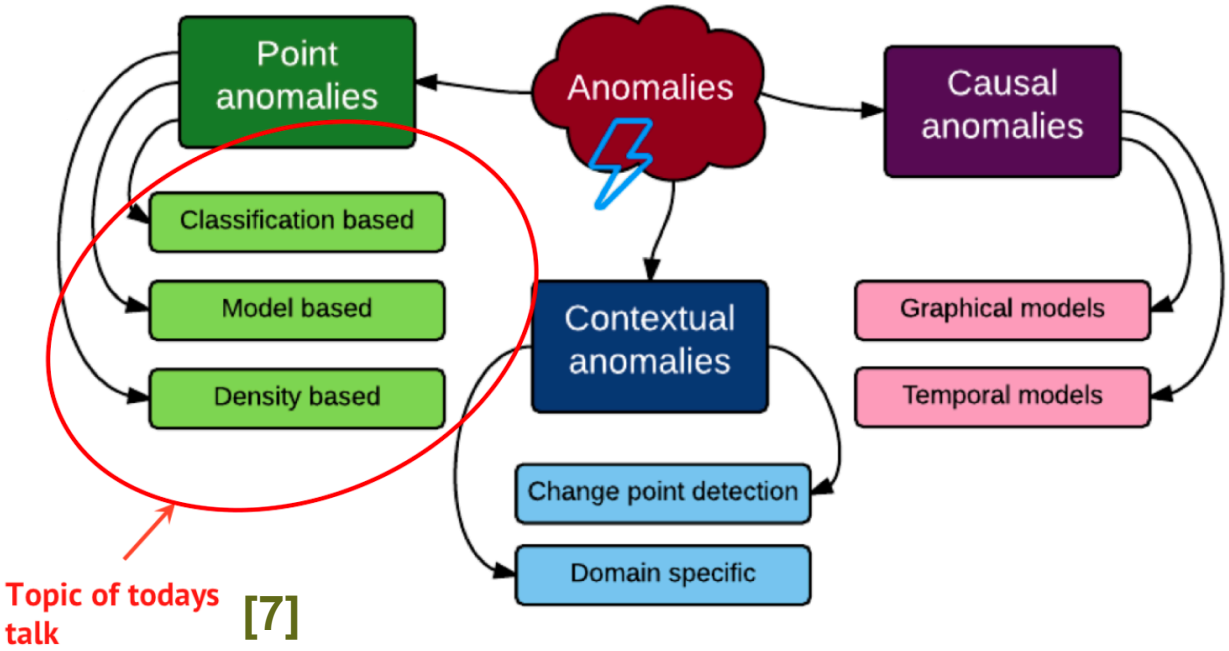
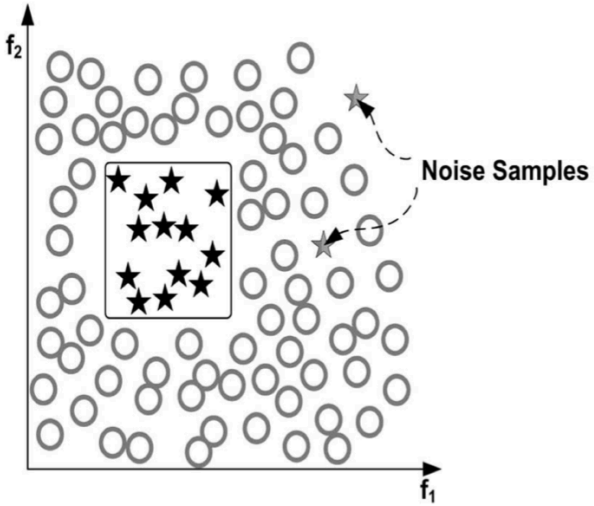
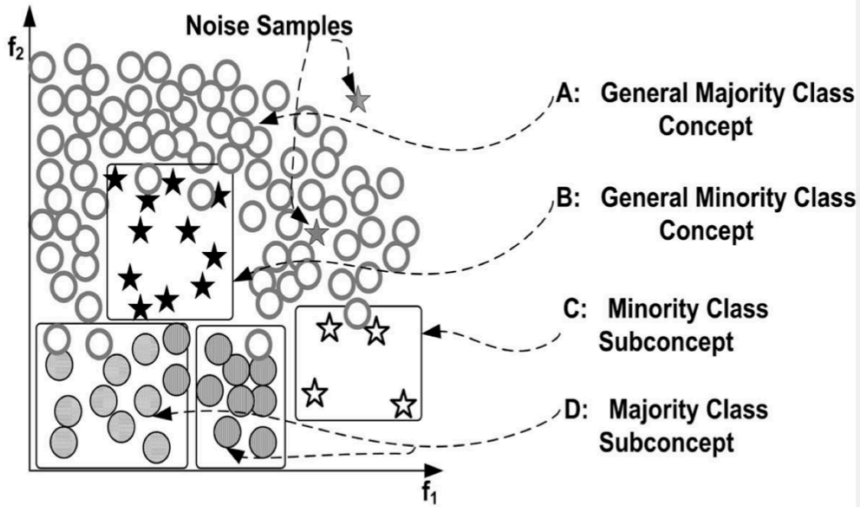


Table of contents

Introduction

- Between-class imbalance (relative imbalance)
- Relative imbalance vs. imbalance due to rare instances or "absolute rarity"
  - Within class imbalance
- Data complexity vs. imbalanced data vs. small sample size





- Binary classification: often dataset has "natural" imbalance
- Minor class (of **prime** interest) vs. major class: e.g. classification of "cancerous" vs. "healthy" mammography image
- Standard classifiers (SVM, kNN, log. reg., etc.): classes are equally important $\Rightarrow$ results are biased towards the major class
- Poor prediction of minor class while the average quality can be good:
  - target events occurs in 1% of all cases
  - classifier always gives a 'no-event' answer
  - it is wrong just 1% of all cases

- Approaches to increase importance of the minor class:
  - Adapt a probability threshold for classifiers,
  - Modify a loss function, e.g., by assigning more weight to the minor class error,
  - Resample a dataset in order to soften or remove class imbalance
- We focus on resampling: allows to use standard classifiers

Notations and Problem Statement

- Dataset $S_m = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$, where $\mathbf{x}_i \in \mathbb{R}^N$, $y_i \in \{-1, +1\}$
- $C_{+1}(S_m) = \{(\mathbf{x}_i, y_i) \in S_m \mid y_i = +1\}$ is a major class,
- $C_{-1}(S_m) = \{(\mathbf{x}_i, y_i) \in S_m \mid y_i = -1\}$ is a minor class, i.e. $|C_{+1}(S_m)| > |C_{-1}(S_m)|$
- Imbalance ratio $IR(S_m) = \frac{|C_{-1}(S_m)|}{|C_{+1}(S_m)|}$, $IR(S_m) \leq 1$

Learning a classifier

- Learn a classifier using imbalanced training sample S_m,
- The dataset S_m is resampled using a method r:
  - some observations in S_m are dropped, or
  - some new synthetic observations are added to S_m
- The result of resampling is a dataset $r(S_m)$ with $IR(r(S_m)) > IR(S_m)$,
- Standard classification model f is learned on $r(S_m)$ to construct a classifier $f_{r(S_m)} : \mathbb{R}^d \rightarrow \{-1, +1\}$

Resampling method r:

1. Takes input:
   - dataset S_m;
   - resampling multiplier $m > 1$ for resulting imbalance ratio $IR(r(S_m)) = m \cdot IR(S_m)$;
   - additional parameters, specific for the method
2. Add synthesized objects to the minor class (oversampling), or drop objects from the major class (undersampling), or both
3. Outputs resampled dataset $r(S_m)$ with imbalance ratio $IR(r(S_m)) = m \cdot IR(S_m)$

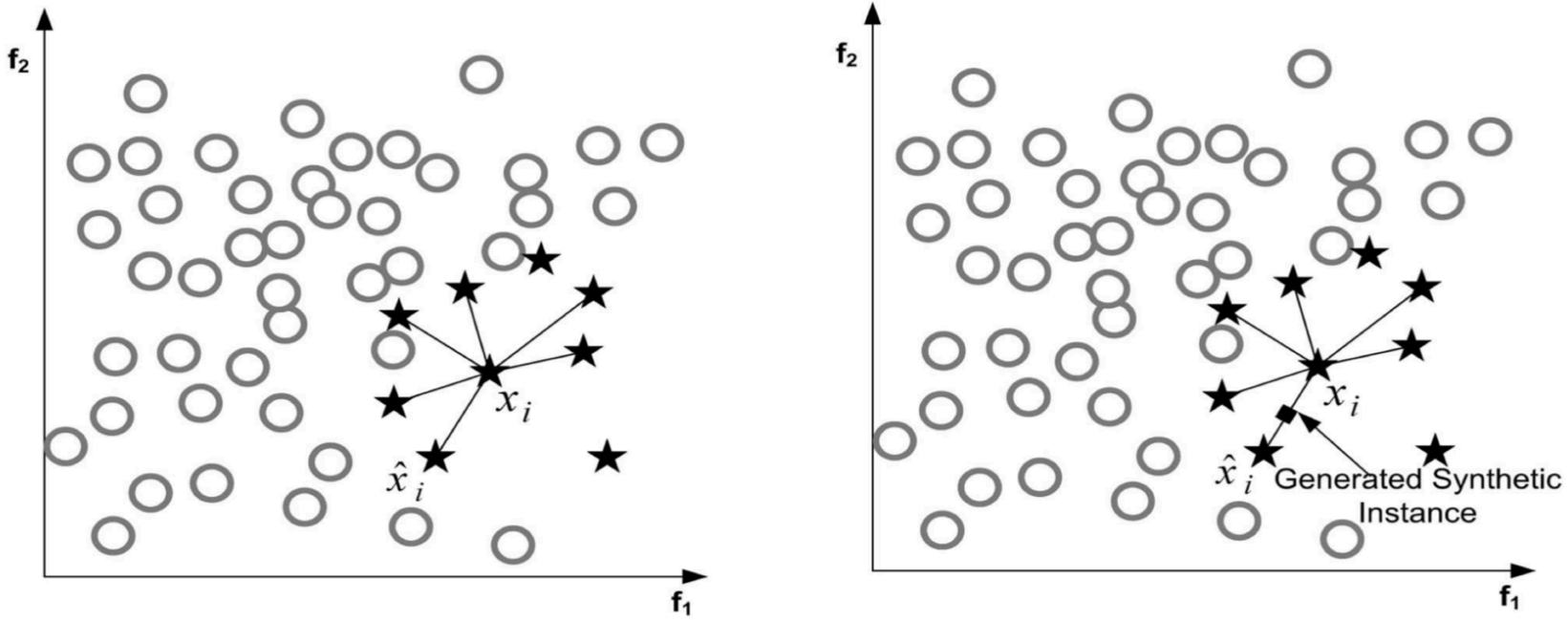
Random Oversampling (ROS)

- ROS, also known as bootstrap oversampling
- No additional input parameters
- It adds to the minor class new $(m-1)|C_{-1}(S_m)|$ objects
- Each of objects is drawn from uniform distribution on $C_{-1}(S_m)$

Random Undersampling (RUS)

- No additional input parameters
- It chooses random subset of $C_{+1}(S_m)$ with $|C_{+1}(S_m)|\frac{m-1}{m}$ elements and drops it from the dataset
- All subsets of $C_{+1}(S_m)$ have equal probabilities to be chosen

Synthetic Minority Oversampling Technique (SMOTE)



Setup of Experiments

- For each (artificial/real) dataset we varied classifier model, resampling method and multiplier
- We used Bootstrap, RUS and SMOTE with $k = 5$
- We varied resampling multiplier m from 1.25 to 10.0
- We used Decision Trees, k-Nearest Neighbors, and Logistic Regression with ℓ_1 regularization
- CV to select classifier parameters
- Resampling multiplier selection:
  - The equalizing strategy
  - CV-search

Dolan-More Curves

- $\{r_1, \ldots, r_n\}$ — the set of considered methods (e.g. resampling methods)
- $\{S_1, \ldots S_T\}$ — the set of tasks (datasets),
- q_{ti} — the quality of the method i on the dataset t,
- $p_i(\beta)$ is a fraction of tasks, on which the method i is worse than the best one not more than β times:

$$p_i(\beta) = \frac{1}{T} \left| \left\{ t : q_{ti} \geq \frac{1}{\beta} \max_i q_{ti} \right\} \right|, \ \beta \geq 1$$

Results for Decision Trees classifier

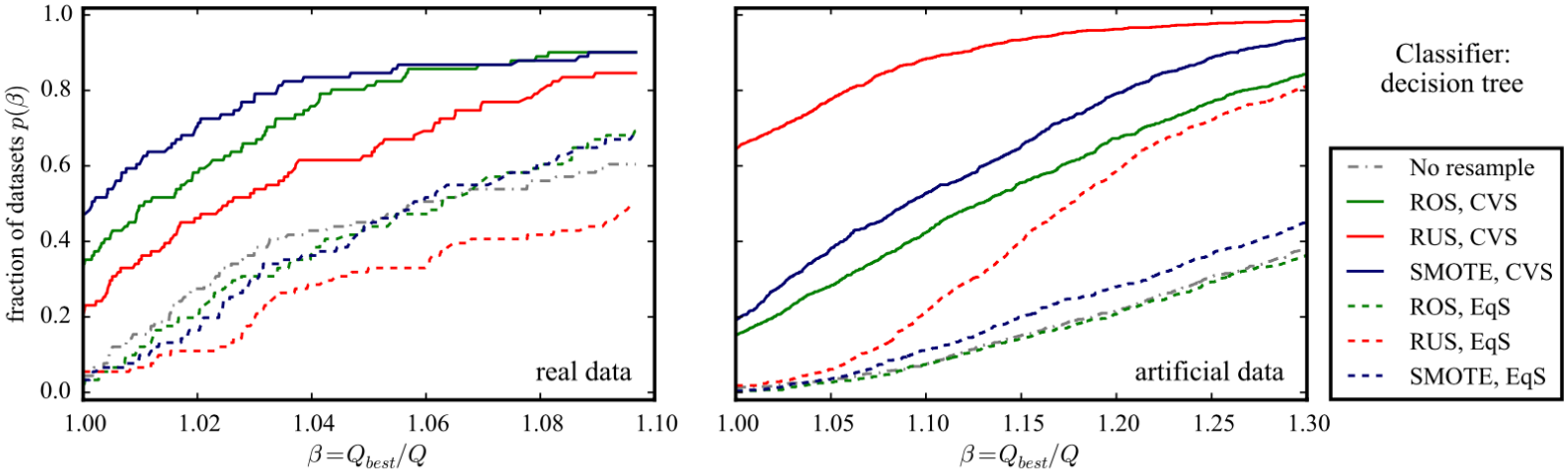


Figure – Dolan-More curves for metric Q_{PRC}^{CV}

Results for k-NN classifier

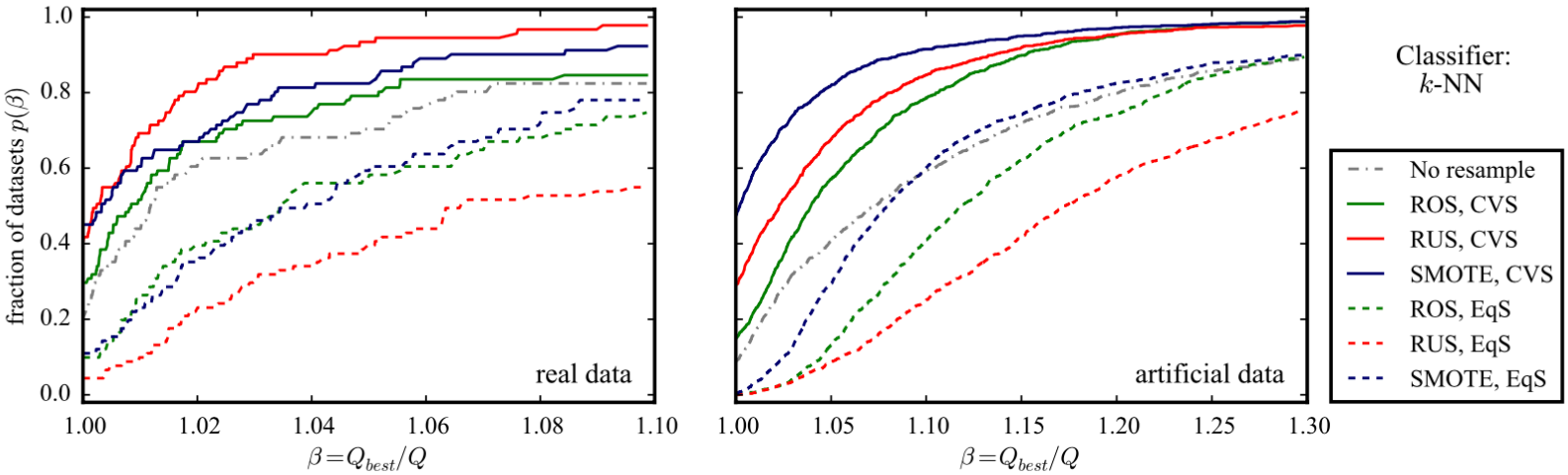


Figure – Dolan-More curves for metric Q_{PRC}^{CV}

Results for l_1-logistic regression

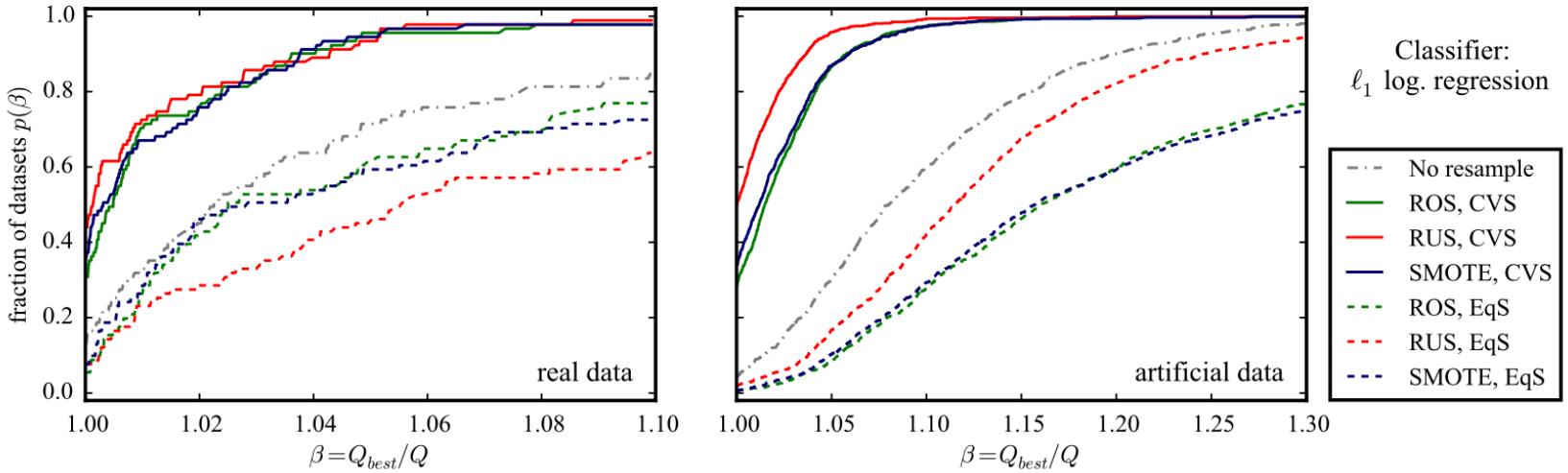


Figure – Dolan-More curves for metric Q_{PRC}^{CV}

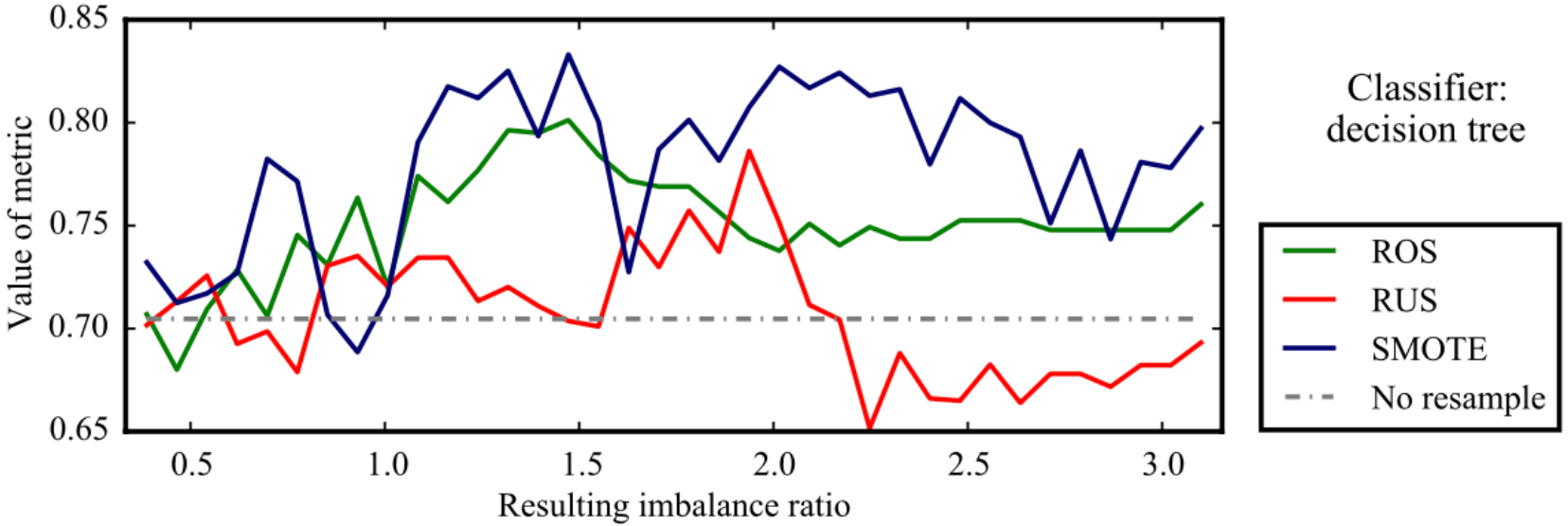


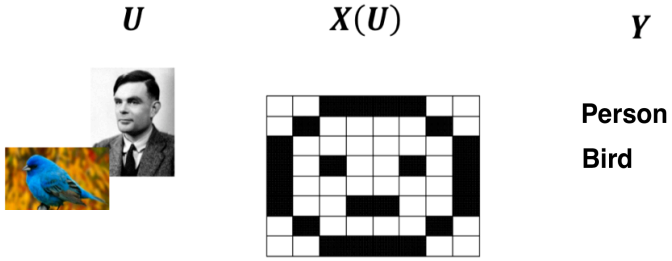
Figure – Value of Q_{PRC}^{CV} vs. resulting value of IR for dataset "Delft pump 1x3"

Comments [2,3]

- Influence of resampling on the quality strongly depends on resampling multiplier
- Resampling with CV-search of multiplier provides better results, especially for Decision trees and Logistic regression
- The equalizing strategy (EqS) shows much lower quality, especially in case of k-NN and Logistic regression
- Performance of resampling depends on classifier used
- There is no method that would always outperform the others

Table of contents

Motivation



- We consider a binary classification problem statement
- The problem is possibly imbalanced (typical for applications). E.g. we should detect cancer/no-cancer using MRI. The number of cases with cancer (minor class) is small
- The main goal of the imbalanced classification is to accurately detect a minor class
- However, standard classification approaches (logistic regression, SVM, etc.) treat all classes as equally important
- As a consequence the resulting classification model is biased towards the major class. E.g., if we predict an event occurring in just 1% of all cases and the classification model always gives a “no-event” answer, then it is wrong in just 1% of all cases

Outline

1. To deal with possible class imbalance when constructing a classifier we use a weighted error (risk) to stress the most important class (accurate detection is needed!)

2. How to select an appropriate weight value to up-weight a minor class?

3. We obtain a **generalization bound** for a **weighted binary classification** and estimate an optimal weight

4. Results of **computational experiments** demonstrate usefulness of the obtained estimate

Related works

- There exist results in classification performance with a weighted loss
- E.g. in [1] a bayesian framework for imbalanced classification with a weighted risk is proposed,
- [2] investigated the calibration of asymmetric surrogate losses,
- [3] considered the case of cost-sensitive learning with noisy labels.
- However, to the best of our knowledge, there is no studied upper bound for the excess risk with explicit dependence on the class imbalance π and the weighting scheme u that quantifies the influence on the overall classification performance

[1] G. Dupret and M. Koda, “Bootstrap re-sampling for unbalanced data in supervised learning,” European Journal of Operational Research, 2001.
[2] C. Scott, “Calibrated asymmetric surrogate losses,” Electron. J. Statist.,2012.
[3] N. Natarajan, I. S. Dhillon, and et al., “Cost-sensitive learning with noisy labels,” JMLR, 2018.

Some useful definitions

Definition: **Empirical Rademacher complexity**

G - some family of functions from Z to $[a, b]$

$S = (z_1, \ldots, z_m)$ - fixed sample

$$\widehat{\mathfrak{R}}_S(G) = \mathop{\mathrm{E}}_{\boldsymbol{\sigma}} \left[\sup_{g \in G} \frac{1}{m} \sum_{i=1}^{m} \sigma_i g(z_i) \right]$$

where $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_m)^\top$ - Rademacher variables

Some useful definitions

Definition: $\mathbf{Rademacher\ complexity}$

D - some distribution on Z

$$\mathfrak{R}_m(G) = \mathop{\mathrm{E}}_{S\sim D^m}[\widehat{\mathfrak{R}}_S(G)]$$

Theorem. Generalization bounds based on Rademacher complexity

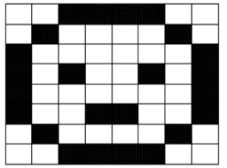
Let G be a family of functions mapping from Z to $[0,1]$. Then, for any $\delta > 0$, with probability at least $1-\delta$, each of the following holds for all $g \in G$:

$$\mathrm{E}[g(z)] \leq \frac{1}{m}\sum_{i=1}^{m} g(z_i) + 2\mathfrak{R}_m(G) + \sqrt{\frac{\log\frac{1}{\delta}}{2m}}$$

$$\textit{and} \quad \mathrm{E}[g(z)] \leq \frac{1}{m}\sum_{i=1}^{m} g(z_i) + 2\widehat{\mathfrak{R}}_S(G) + 3\sqrt{\frac{\log\frac{2}{\delta}}{2m}}$$

Problem statement

U $X(U)$ Y



Person

Bird

$x \in \mathcal{X}$ - input (feature) space

$\mathcal{Y} = \{-1, +1\}$ - output (label) space

$\mathcal{F} \subseteq \mathcal{Y}^{\mathcal{X}}$ - a class of binary classifiers. E.g.

$$\mathcal{F} = \{f_{a,b} : \ f_{a,b}(x) = 2\mathbb{I}(\langle a, x\rangle + b \geq 0) - 1\}$$

$\mathbb{P}$ - unknown distribution on $\mathcal{X} \times \mathcal{Y}$

π - prior probability of a positive class, i.e.

$$\mathbb{P} = \pi \mathbb{P}_{x|y=+1} + (1 - \pi)\mathbb{P}_{x|y=-1}$$

Problem statement

$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ - is a training sample, $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$

$\mathcal{R}_N(\mathcal{F})$ - is a Rademacher complexity of $\mathcal{F}$

$L(\hat{y}, y) = \mathbb{I}_{\hat{y} \neq y}$ - is a zero-one loss function

$u : (\mathcal{X} \times \mathcal{Y}) \to (0, +\infty)$ - some (fixed) weighting function

Problem statement

Theoretical risk

$$\mathbb{E}_{\mathbb{P}} L(f(x), y)$$

Optimal classifier

$$f^* = \arg \min_{f \in \mathcal{F}} \mathbb{E}_{\mathbb{P}} L(f(x), y)$$

Problem statement

Empirical risk:

$$\mathbb{E}_{\mathcal{D}} u(x, y) L(f(x), y) = \frac{1}{N} \sum_{i=1}^{N} u(x_i, y_i) L(f(x_i), y_i)$$

Empirical classifier

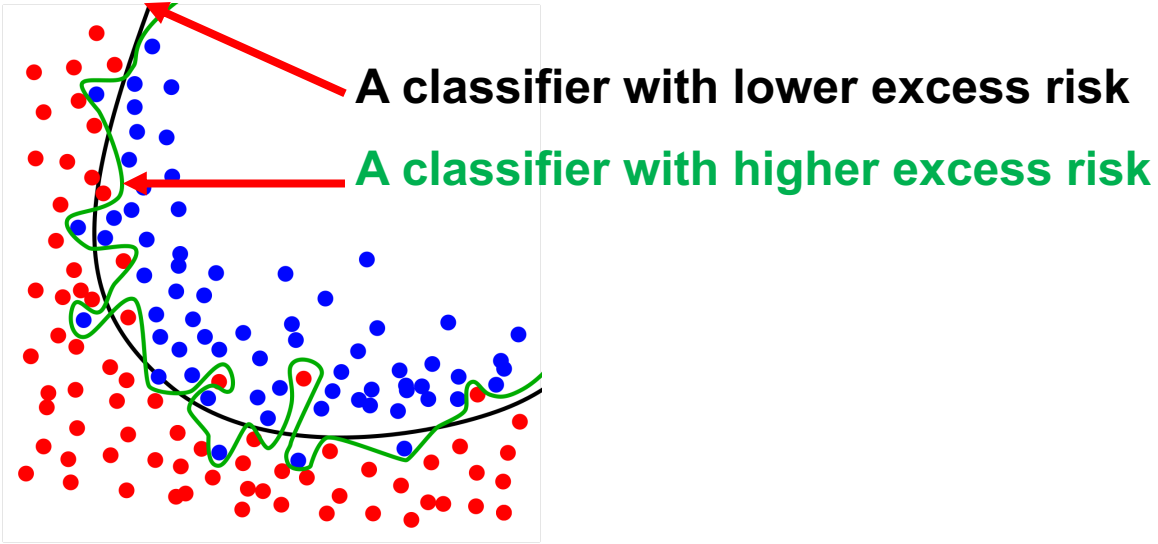
$$\hat{f} = \arg \min_{f \in \mathcal{F}} \mathbb{E}_{\mathcal{D}} u(x, y) L(f(x), y)$$

Problem statement

We would like to derive an upper bound for the excess risk:

$$\Delta(\mathcal{F}, \mathbb{P}) = \sup_{f \in \mathcal{F}} \left(\mathbb{E}_{\mathbb{P}} L(f(x), y) - \mathbb{E}_{\mathcal{D}} u(x, y) L(f(x), y) \right)$$

The excess risk characterizes a generalization ability of the classifier



Generalization bound

To derive explicit expressions we use an additional assumption

$$u(x,y) = (1 + g_+(w))\mathbb{I}_{\{y=+1\}} + (1 + g_-(w))\,\mathbb{I}_{\{y=-1\}}$$

for some positive weighting functions $g_+(w)$ and $g_-(w)$

Theorem [4]: *With probability* $1-\delta$, $\delta > 0$ *for* $\mathcal{D} \sim \mathbb{P}^N$ *the excess risk* $\Delta(\mathcal{F}, \mathbb{P})$

is upper bounded by

$$\overline{\Delta}(w) = 3\left(g_+(w)\pi + g_-(w)(1-\pi)\right) + \mathcal{R}_N(\mathcal{F}) + \left(2 + g_+(w) + g_-(w)\right)\alpha_N,$$

where $\alpha_N = \sqrt{\frac{\log \delta^{-1}}{2N}}$.

Generalization bound: optimal weight selection (I)

So, the upper bound on the excess risk is equal to

$$\overline{\Delta}(w) = 3\left(g_+(w)\pi + g_-(w)(1-\pi)\right) + \mathcal{R}_N(\mathcal{F}) + \left(2 + g_+(w) + g_-(w)\right)\alpha_N$$

By collecting the terms with w in $\overline{\Delta}(w)$ we get

$$g_+(w)\left(3\pi + \alpha_N\right) + g_-(w)\left(3(1-\pi) + \alpha_N\right)$$

We set $g_+(w) = w$ and $g_-(w) = 1/w$

The optimal weight

$$w^* = \sqrt{\frac{3(1-\pi)+\alpha_N}{3\pi+\alpha_N}} \approx \sqrt{\frac{1-\pi}{\pi}}, \text{ where } \alpha_N \approx 0 \text{ for } N \gg 1$$

Generalization bound: optimal weight selection (II)

Finally:

- We weight examples from the positive class with a weight $1 + w$
- We weight examples from the negative class with a weight $1 + 1/w$
- The optimal weight to minimize the upper bound of the excess risk is equal to

$$w^* \approx \sqrt{\frac{1-\pi}{\pi}}$$

For such optimal weight value the upper bound of the excess risk is equal to

$$\overline{\Delta}^* = 6\sqrt{\pi(1-\pi)} + \mathcal{R}_N(\mathcal{F}) + \alpha_N \left(2 + [\pi(1-\pi)]^{-\frac{1}{2}}\right)$$

Therefore, in imbalanced case ($\pi \approx 0$ or $\pi \approx 1$) for $N \gg 1$ and "standard functions classes" we get that

$$\overline{\Delta}^* \approx 0$$

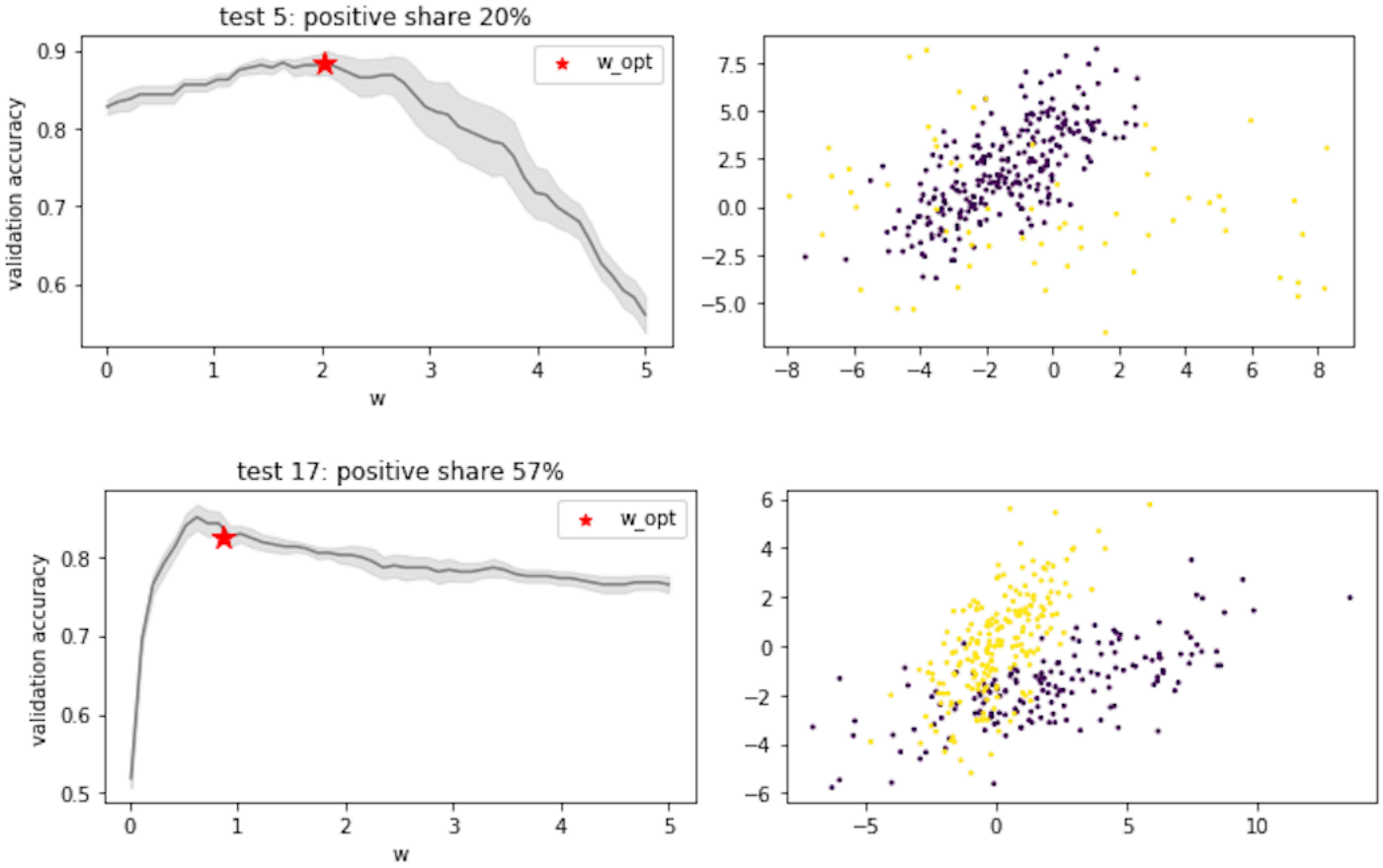
Empirical evaluation

We expect that for the optimal weight value a classifier achieve better accuracy on the test when when being trained by minimizing the weighted empirical loss

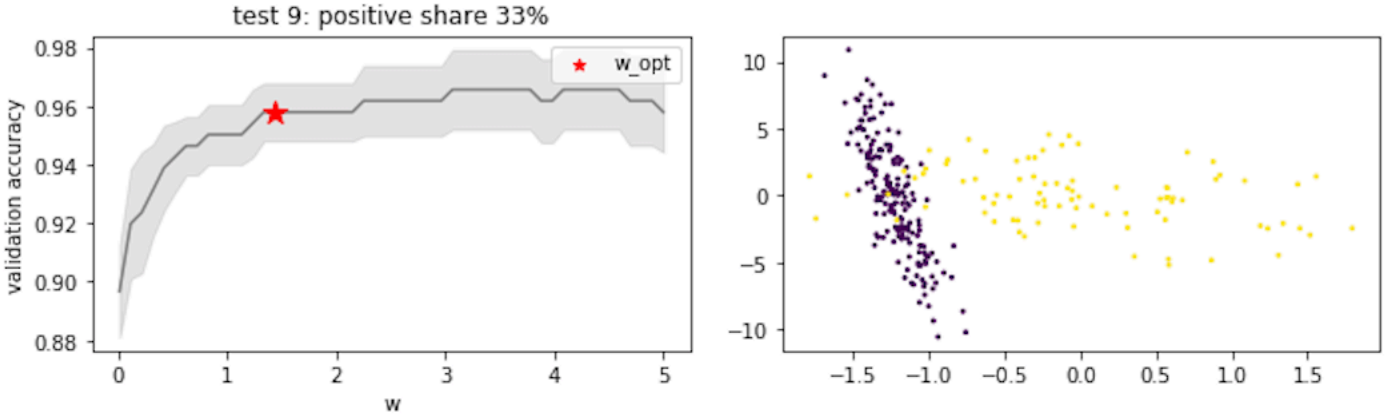
Protocol of experiments:

- Consider different values of the weight $w \in \{w_1, \ldots, w_K\}$
- Train a classifier $f_w(x)$ by minimizing a weighted empirical loss for a particular weight value $w = w_i$
- Estimate accuracy on the test set and find the weight for which accuracy is the highest
- Compare the best obtained weight with the theoretical weight calculated using the formula $w^* \approx \sqrt{\frac{1-\pi}{\pi}}$

Results: 2d toy problems



Results: 2d toy problems



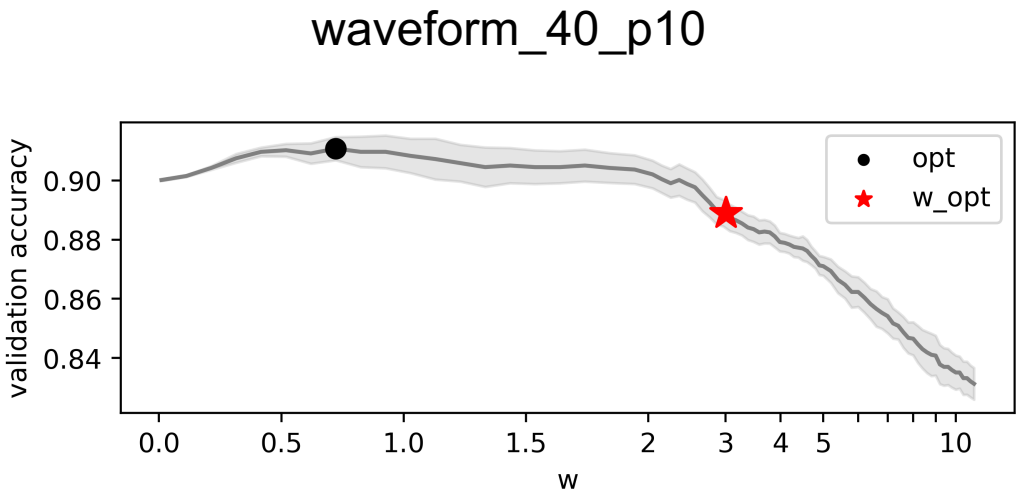
Results: real-world problems

- Datasets were taken from Penn Machine Learning Benchmarks repository: we selected diabetes, german, waveform-40, satimage, splice, spambase, hypothyroid, and mushroom, that have various types of data and features

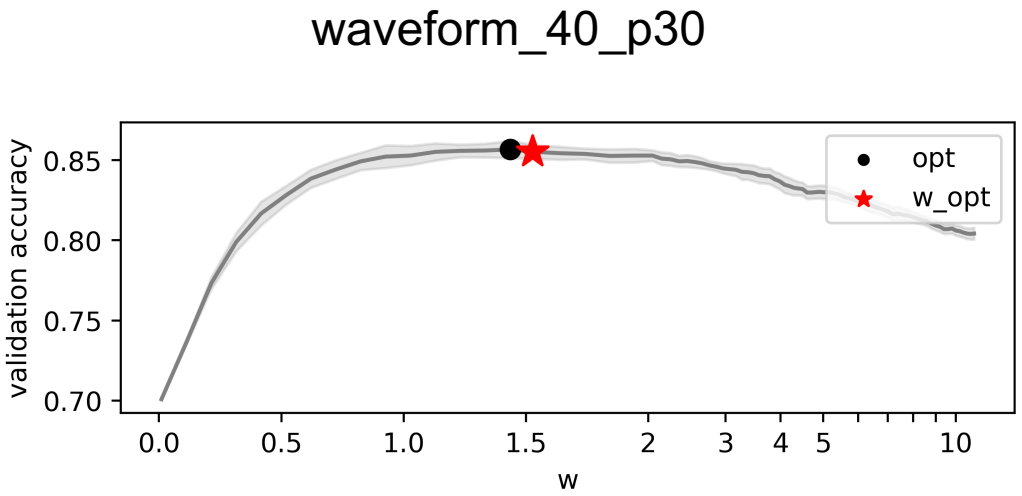
- To obtain a specific balance between classes in experiments, we used undersampling of an excess class. Using this method, we varied the positive class share among the following values: 0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 0.95, 0.99

- To measure the performance of the method, we conducted 5-fold cross-validation of a Logistic Regression classifier

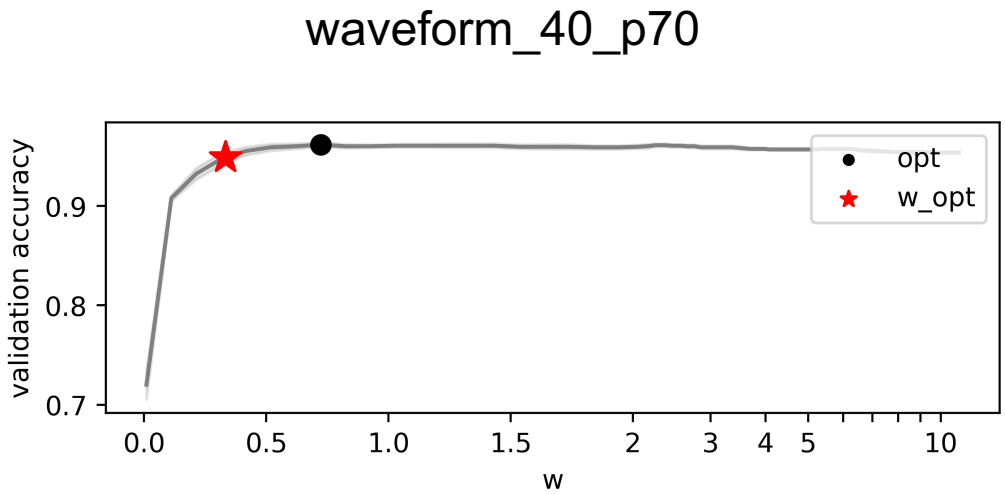
Results: real-world problems



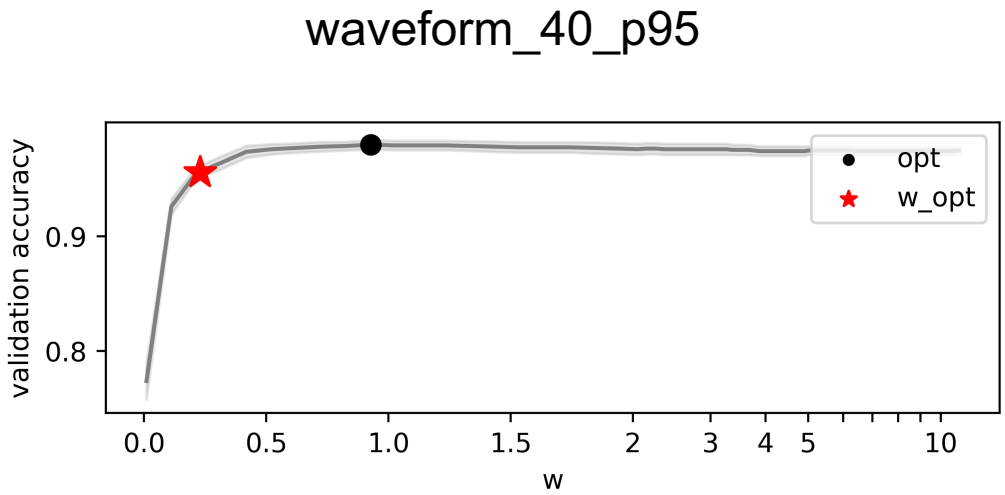
Results: real-world problems



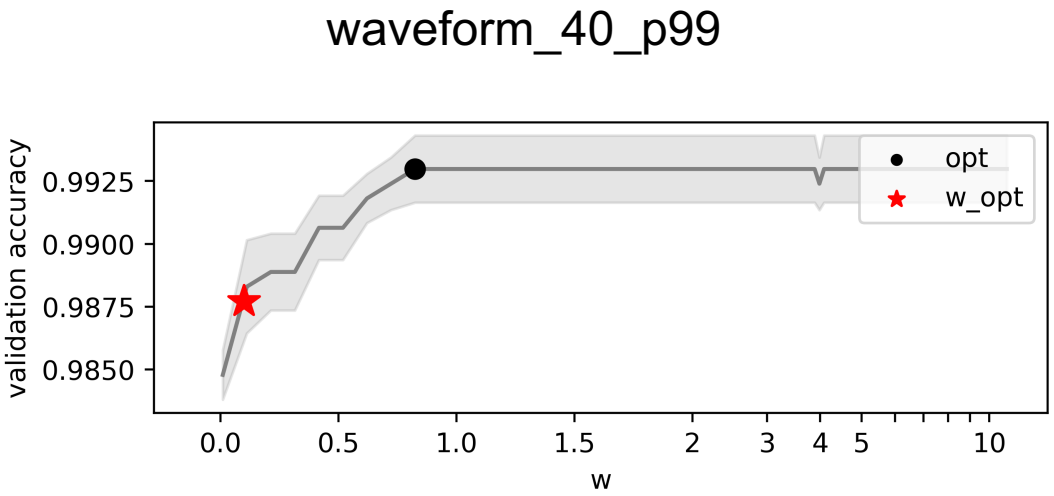
Results: real-world problems



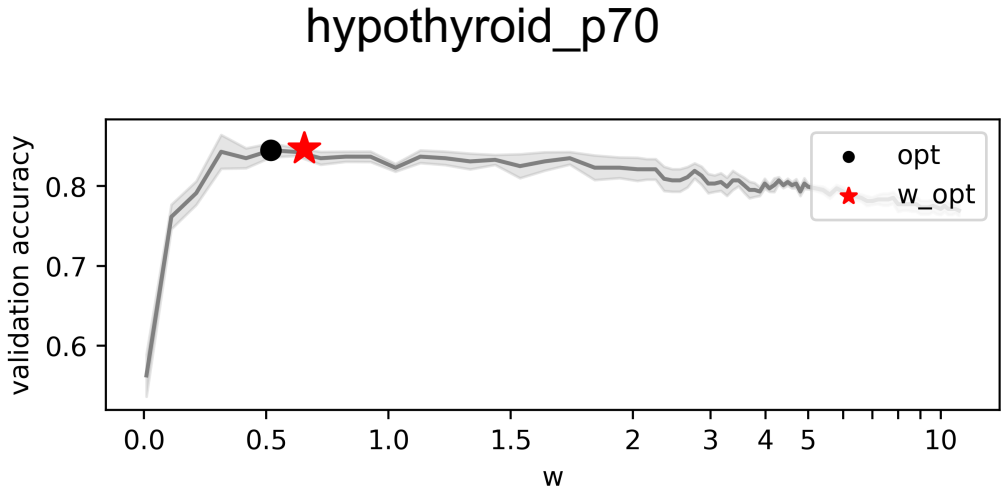
Results: real-world problems



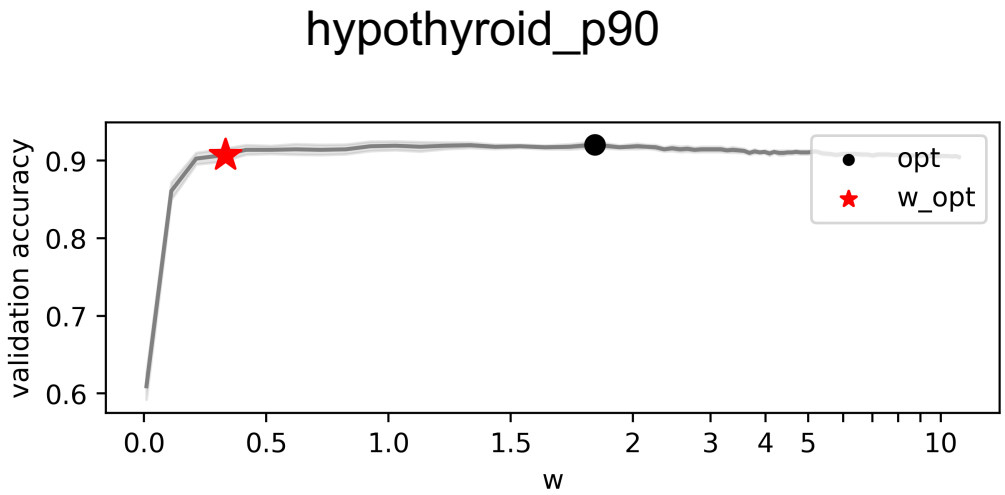
Results: real-world problems



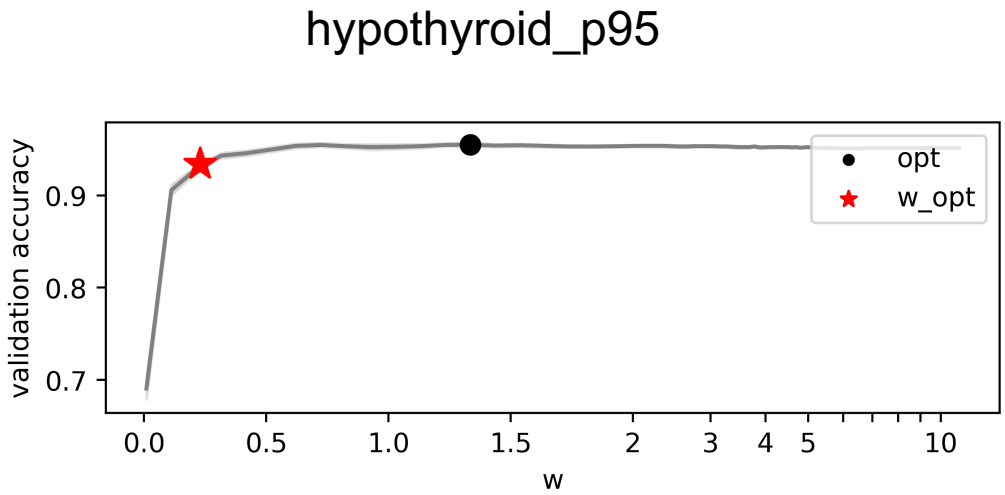
Results: real-world problems



Results: real-world problems



Results: real-world problems



Results: real-world problems

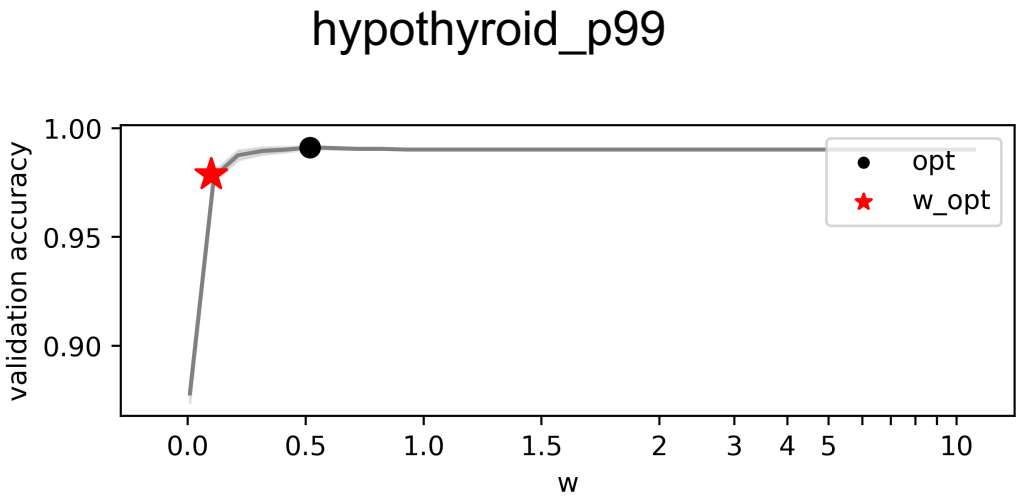


Table of contents

Problem Formulation

- Let $S = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}, \mathbf{x}_i \in \mathbb{R}^p$ be an unlabeled sample (possibly containing some anomalies)
- We want to learn $f : \mathbf{x} \to \{-1, 1\}$ using the sample

$$\mathbf{x} = \begin{cases} \text{normal, if } f(\mathbf{x}) = +1, \\ \text{anomaly, if } f(\mathbf{x}) = -1, \end{cases}$$

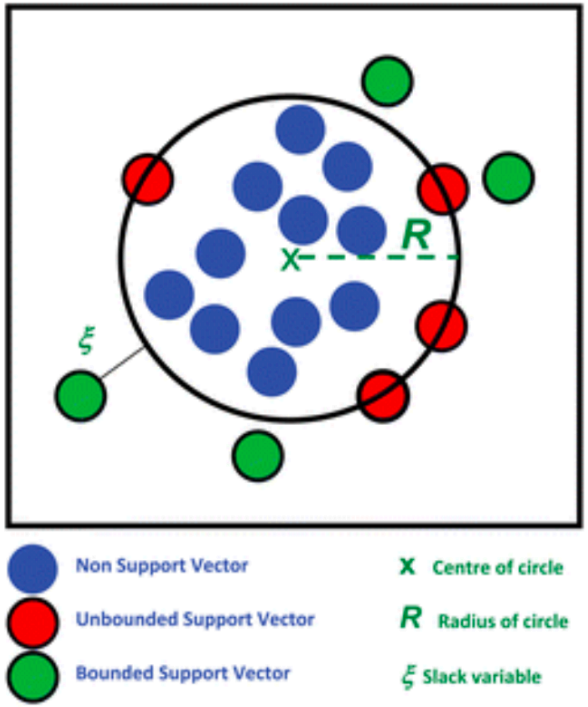
Support Vector Data Description

$$R + \frac{1}{m\nu} \sum_{i=1}^{m} \xi_i \rightarrow \min_{R,a,\xi}$$

$$s.t.\ \|\phi(\mathbf{x}_i) - a\|_2^2 \leq R + \xi_i$$

$$\xi_i \geq 0$$

$$R \geq 0$$



- ν is an upper bound on the fraction of anomalous patterns in the sample S
- $\phi(\mathbf{x}_i)$ is the mapping to a high dimensional space

Dual Problem

We consider the dual problem

$$\sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_i) - \sum_{i,j=1}^{m} \alpha_i \alpha_j \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) \to \max_{\alpha}$$

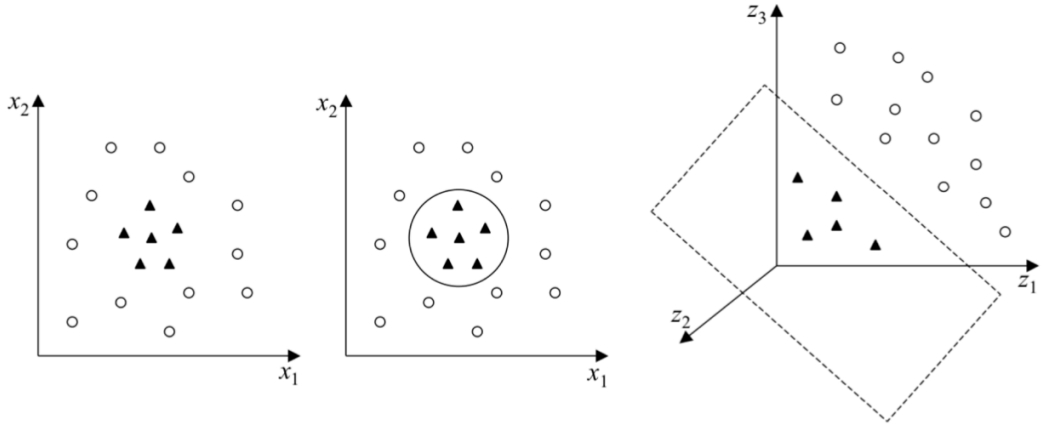
$$s.t. \quad \sum_{i=1}^{m} \alpha_i = 1$$

$$0 \leqslant \alpha_i \leqslant \frac{1}{m\nu}, \quad i = 1, \ldots, m$$

We don't need to use explicit expression for $\phi(\cdot)$, we need only a definition of a dot product. We can use a kernel trick

$$K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$$

Kernel Trick. Kernel Examples



For $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, let $\phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1x_2, x_2^2) \in \mathbb{R}^3$. Then

$$K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$$
$$= x_1^2(x_1')^2 + 2x_1x_2x_1'x_2' + x_2^2(x_2')^2 = (x_1x_1' + x_2x_2')^2 = (\mathbf{x} \cdot \mathbf{x}')^2$$

Name	Equation	hyperparameters
Linear	$x \cdot y$	—
Polynomial	$(\sigma^2 x \cdot y + d)^k$	σ^2, d, k
RBF	$\exp(-\sigma^2 \|x - y\|^2)$	σ^2
Sigmoid	$\tanh(\sigma x \cdot y + d)$	$\sigma > 0$, $d > 0$

Solution of the primal problem

- We can write out the solution of the primal problem using the solution of the dual problem

$$a = \sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i), \quad R = \|\phi(\mathbf{x}_j)\|_2^2 - 2(a \cdot \phi(\mathbf{x}_j)) + \|a\|_2^2,$$

  where we can use any $\mathbf{x}_j$, such that $\alpha_j > 0$

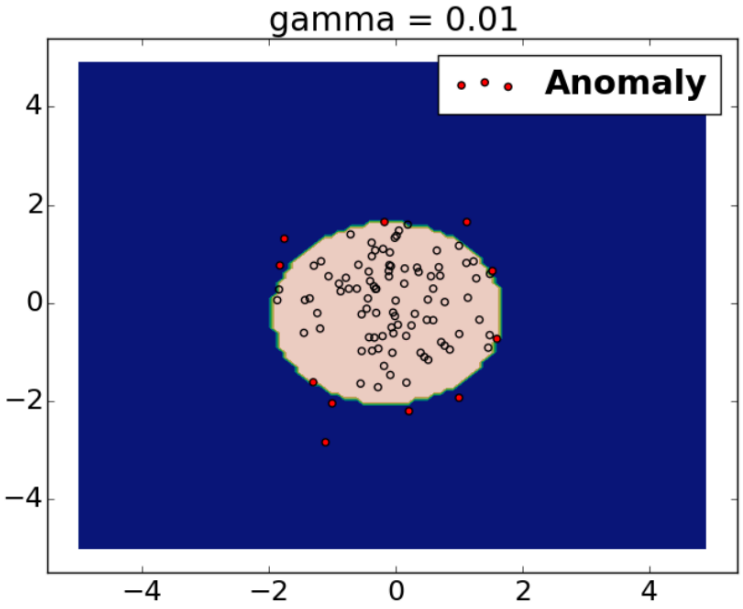
- Here $\|\phi(\mathbf{x})\|_2^2 = K(\mathbf{x}, \mathbf{x})$, $(\phi(\mathbf{x}) \cdot a) = \sum_{i=1}^{m} \alpha_i K(\mathbf{x}_i, \mathbf{x})$ and $\|a\|_2^2 = \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j)$

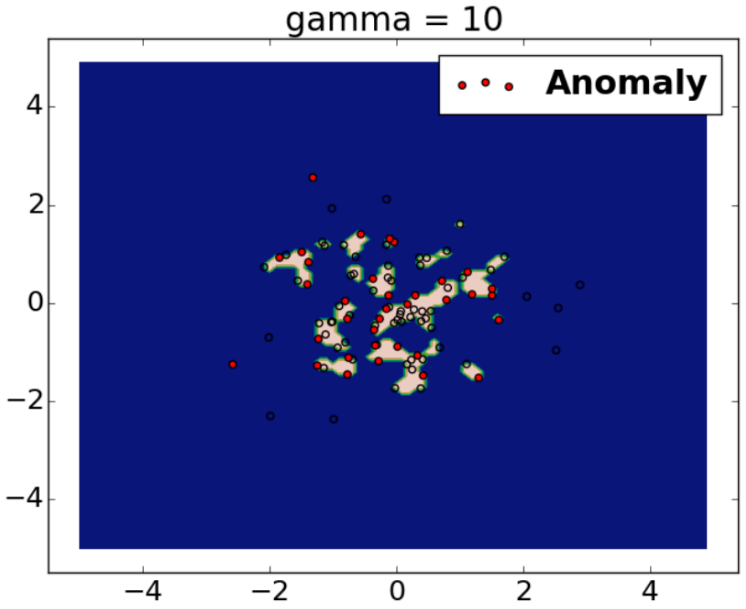
- The **decision function** has the form

$$f(\mathbf{x}) = \operatorname{sign}\left\{ R - K(\mathbf{x}, \mathbf{x}) + 2 \sum_{i=1}^{m} \alpha_i K(\mathbf{x}, \mathbf{x}_i) - \|a\|_2^2 \right\}.$$

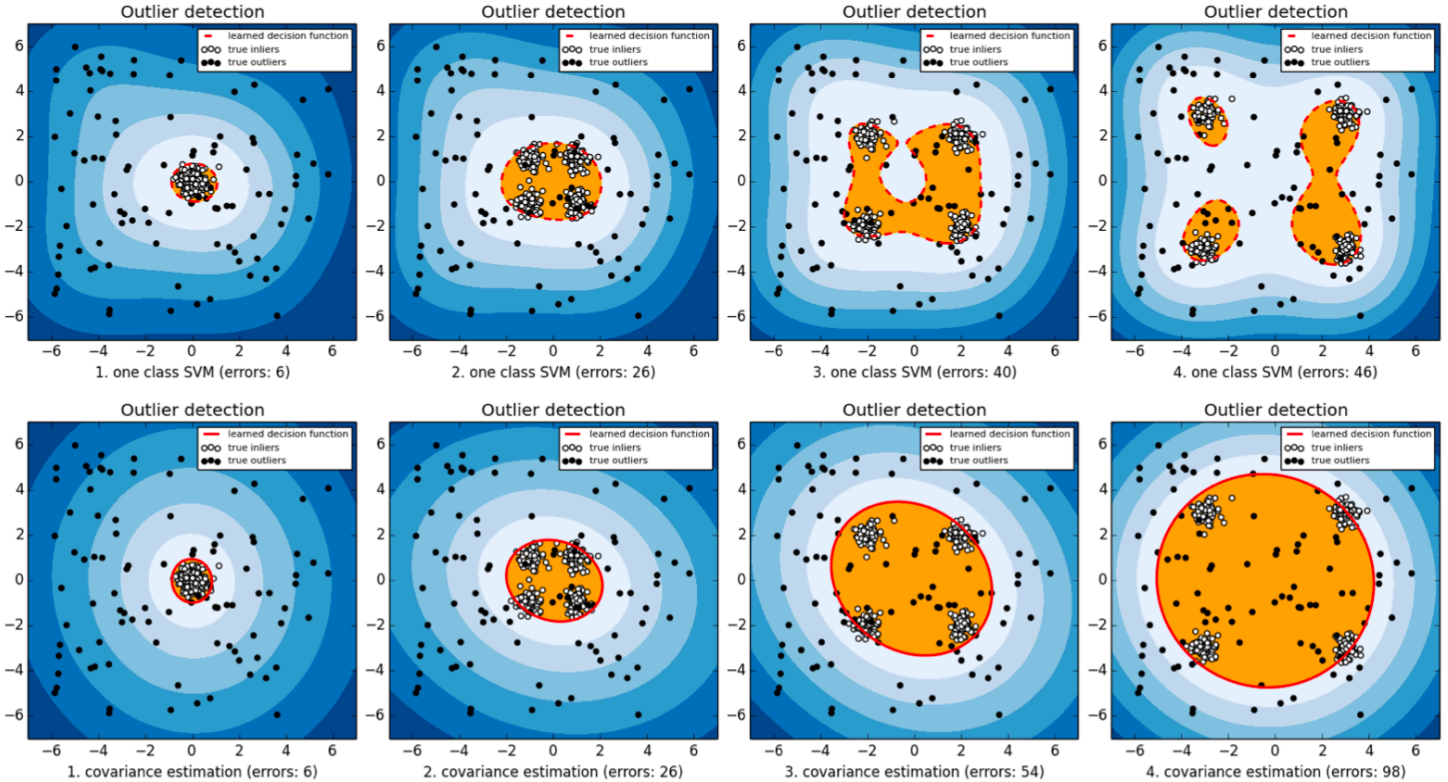
Kernel Selection [10]

Results can significantly depended on a kernel hyperparameters





Example of the decision function



1. Supervised Learning
   - Sample $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$
   - We want to learn $f : \mathbf{x} \to y$ using the sample S

Supervised Learning with Privileged Information

2. Supervised Learning with Privileged Information (Vapnik, 2009)
   - Sample $S^* = \{(\mathbf{x}_1, \mathbf{x}_1^*, y_1), \ldots, (\mathbf{x}_m, \mathbf{x}_m^*, y_m)\}$
   - We want to learn $f : \mathbf{x} \to y$ using the sample S^*
   - Privileged Information:
     - in the form of additional patterns $\mathbf{x}^*$
     - is not available at the test time
   - Example:
     - image classification problem
     - as the privileged information we can use a textual image description
     - such information is not available during the test phase

One-Class Classification with Privileged Information [8,9]

- Original patterns $(\mathbf{x}_1, \ldots, \mathbf{x}_m) \subset \mathbb{R}^p$
- Additional patterns $(\mathbf{x}_1^*, \ldots, \mathbf{x}_m^*) \subset \mathbb{R}^q$
- We train a decision rule on pairs of patterns $\{(\mathbf{x}_i, \mathbf{x}_i^*)\}_{i=1}^m \in \mathbb{R}^{p+q}$, but when making decisions we can use only test patterns $\mathbf{x} \in \mathbb{R}^p$

$$R + \frac{1}{m\nu} \sum_{i=1}^{m} \xi_i \to \min_{R,a,\xi}$$

$$s.t.\ \|\phi(\mathbf{x}_i) - a\|_2^2 \leqslant R + \xi_i$$

$$\xi_i \geqslant 0$$

$$R \geqslant 0$$

- The slack variables ξ_i characterizes the distance from the patterns $\mathbf{x}_i$ to the separating boundary $\|\phi(\mathbf{x}_i) - a\|_2$
- We assume that using the privileged patterns $(\mathbf{x}_1^*, \ldots, \mathbf{x}_m^*)$ we can refine the location of the separating boundary
- We model a slack variable ξ as

$$\xi = \xi(\mathbf{x}^*) = (\phi^*(\mathbf{x}^*) \cdot w^*) + b^*,$$

where $\phi^*(\cdot)$ is a feature map in the space of privileged patterns

Support Vector Data Description with Privileged Information

We incorporate the privileged information

$$\nu m R + \frac{\gamma}{2}\|w^*\|_2^2$$

$$+\sum_{i=1}^{m}[(w^* \cdot \phi^*(\mathbf{x}_i^*)) + b^* + \zeta_i] \to \min_{R,a,w^*,b,\zeta}$$

$$s.t.\ \|\phi(\mathbf{x}_i) - a\|_2^2 \leq R + [(w \cdot \phi^*(\mathbf{x}_i^*)) + b^*],$$

$$(w^* \cdot \phi^*(\mathbf{x}_i^*)) + b^* + \zeta_i \geq 0,\ \zeta_i \geq 0.$$

Dual Problem

Let us formulate the dual problem:

$$\sum_{i=1}^{m} \alpha_i K(\mathbf{x}_i, \mathbf{x}_i) - \frac{1}{2\nu m} \sum_{i,j} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j)$$

$$- \sum_{i,j} \frac{1}{2\gamma} (\alpha_i - \delta_i) K^*(\mathbf{x}_i^*, \mathbf{x}_j^*) (\alpha_j - \delta_j) \to \max_{\alpha, \delta}$$

$$s.t. \ \sum_{i=1}^{m} \alpha_i = \nu m, \ \sum_{i=1}^{m} \delta_i = \nu m, \ 0 \leq \delta_i \leq 1, \ \alpha_i \geq 0.$$

The **decision function** has again the same form

$$f(\mathbf{x}) = \text{sign} \left\{ R - K(\mathbf{x}, \mathbf{x}) + 2 \sum_{i=1}^{m} \alpha_i K(\mathbf{x}, \mathbf{x}_i) - \|a\|_2^2 \right\}$$

KDD-99 Challenge

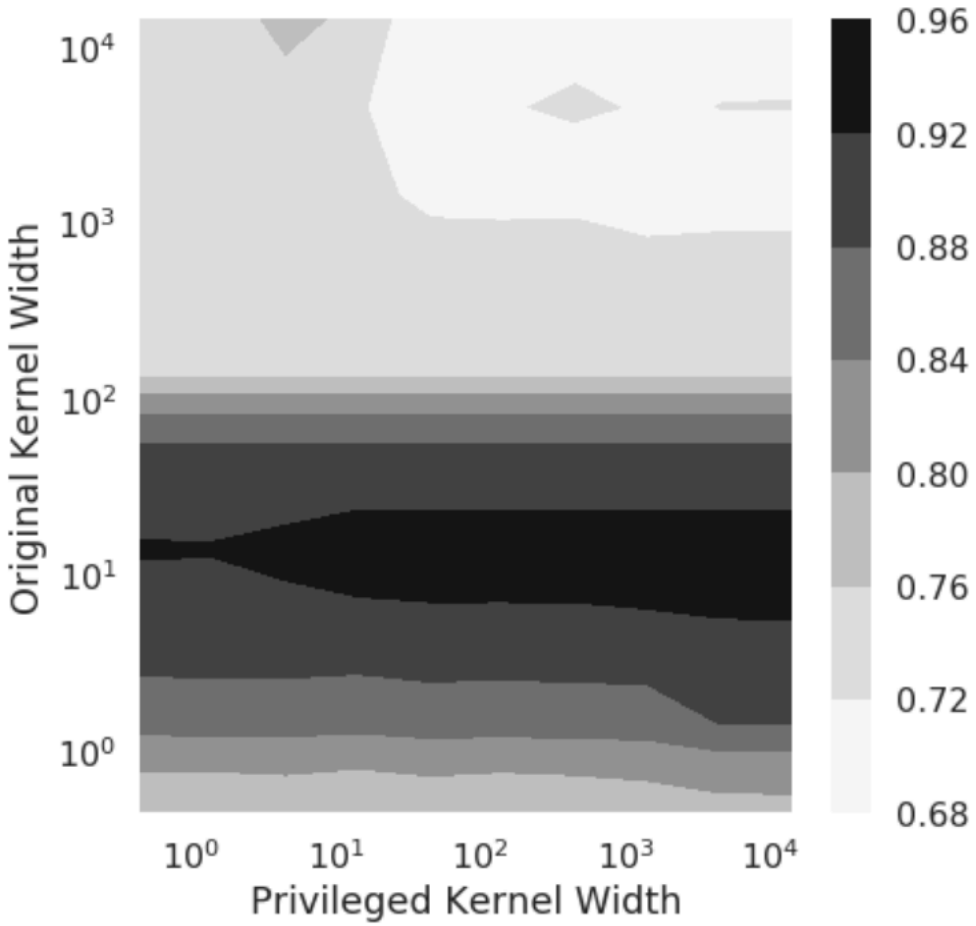
Every data sample describes TCP connection as a 41-feature vector labeled as either normal or an attack, with exactly one specific attack type

There are three types of features in this dataset:

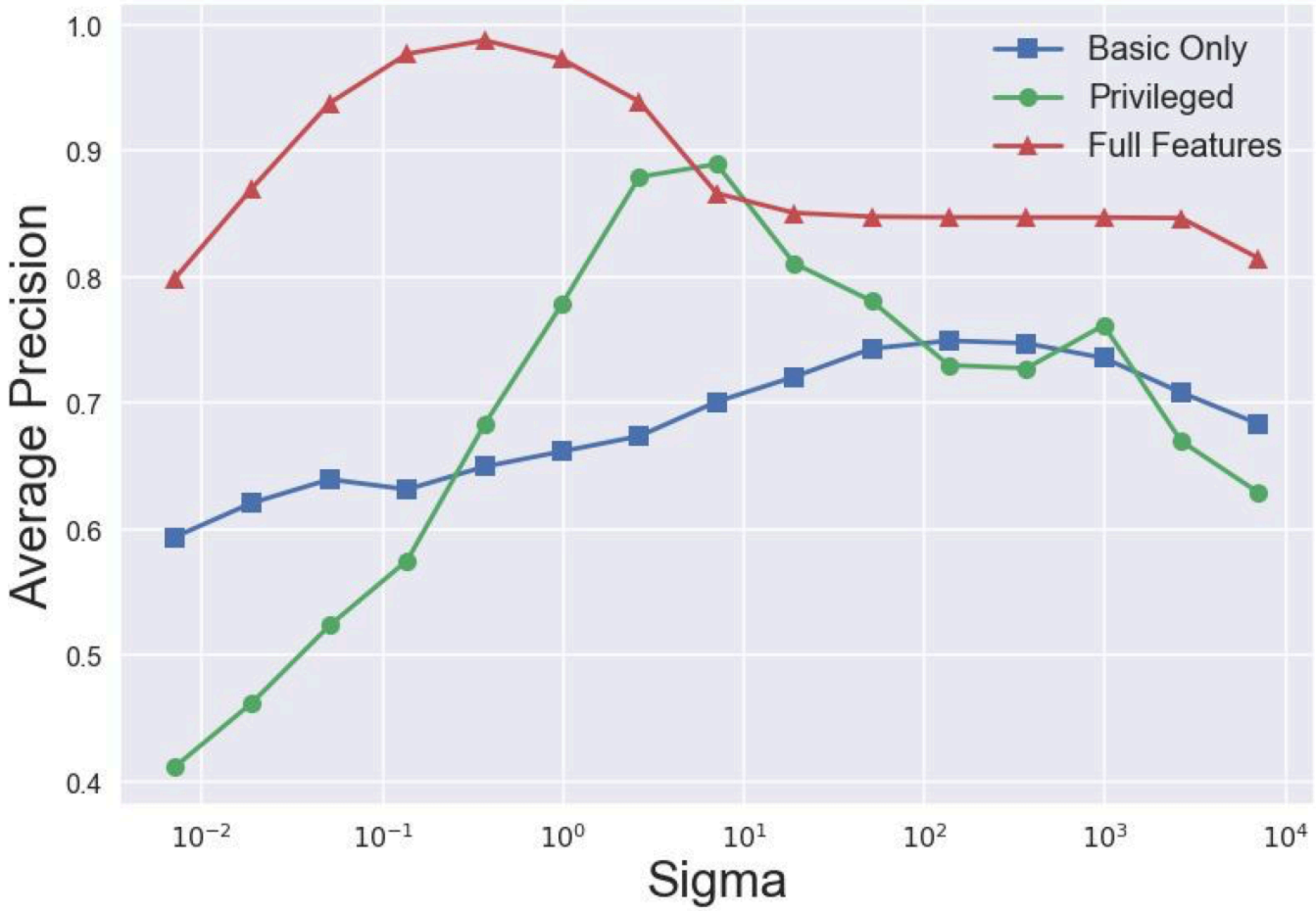
1. The first type is generated directly from TCP dump: the type of the protocol, number of fragments sent, destination network service, etc.
2. The features of the second type are proposed by domain experts.
3. The features of the third type are based on the connection history in a 2-second time window.

We test

- OC-SVM using all features,
- OC-SVM using only features of the first type, and
- OC-SVM+ with
    - features of the first type being original information and
    - the second and third types as privileged information.



(a) Performance On Test Data



Other hyperparameters are optimized using a grid search, in all experiments $\nu = 0.1$

Table of contents

Kernels [1]

Let $k(x, x')$ be a kernel that can be represented as

Kernel ridge regression has the form

$$f^*(x) = \mathbf{k}^\top(x) \left(\mathbf{K} + \lambda \mathbf{I}\right)^{-1} \mathbf{y}$$

where

- $\mathbf{y} = (y_1, \ldots, y_n)^\top$
- $\mathbf{k}(x) = (k(x, x_1), \ldots, k(x, x_n))$
- $\mathbf{K} = \{k(x_i, x_j)\}$

Complexity: $O(n^3)$

Kernels: Quadrature approximation

We assume that

$$k(x, x') = \int_{\Omega} \underbrace{\psi(\mathbf{w}, x)\psi(\mathbf{w}, x')}_{f_{xx'}(\mathbf{w})} p(\mathbf{w}) d\mathbf{w}$$

with $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|0, \sigma_p^2 \mathbf{I})$

Then we can find D-dim features, s.t.

$$k(x, x') \approx \frac{1}{D} \sum_{i=1}^{D} \psi(\mathbf{w}_i, x)\psi(\mathbf{w}_i, x'), \;\; \mathbf{w}_i \sim p(\mathbf{w})$$

$$k(x, x') \approx \hat{k}(x, x') = \langle \phi(x), \phi(x') \rangle$$

Complexity: $O(nD^2)$

Kernels: approximation accuracy

Theorem: Let

- $l = \operatorname{diam}(\mathcal{X})$
- $|\phi(\mathbf{w}^\top x)| \leq \kappa, |\phi'(\mathbf{w}^\top x)| \leq \mu \quad \forall x \in \mathcal{X}, \mathbf{w} \in \Omega$
- $(1 - f_{xx'}(\rho z))/\rho^2 \leq M \, \forall \rho \in [0, \infty)$ and $z : zz^\top = 1$

Then

$$\mathbb{P}\left(\sup_{x,x' \in \mathcal{X}} |\hat{k}(x,x') - k(x,x')| \geq \varepsilon\right) \leq \beta_d \left(\frac{\sigma_p l \kappa \mu}{\varepsilon}\right)^{\frac{2d}{d+1}} \exp\left(-\frac{D\varepsilon^2}{8M^2(d+1)}\right)$$

with $\beta_d = \left(d^{\frac{-d}{d+1}} + d^{\frac{1}{d+1}}\right) 2^{\frac{6d+1}{d+1}} \left(\frac{d}{d+1}\right)^{\frac{d}{d+1}}$

We guarantee approximation error ε with probability $1 - \delta$ if

$$D \geq \frac{8M^2(d+1)}{\varepsilon^2} \left[\frac{2}{1+\frac{1}{d}} \log \frac{\sigma_p l \kappa \mu}{\varepsilon} + \log \frac{\beta_d}{\delta}\right]$$

Kernels: approximation accuracy

Corollary: Let

- $f^*(x)$ be a KRR with regularization $\lambda = \lambda_0 n$
- $\hat{f}(x)$ be the same KRR with $\hat{k}(x, x')$
- $\sum_{i=1}^{n} y_i = 0,\ \|\mathbf{k}(x)\|_\infty \leq \kappa,\ \sigma_y^2 = \frac{1}{n}\sum_{i=1}^{n} y_i^2$

Then

$$|\hat{f}(\mathbf{x}) - f^*(\mathbf{x})| \leq \varepsilon$$

with probability $1 - \delta$ if

$$D \geq 8M^2(d+1)\sigma_y^2 \left(\frac{\lambda_0+1}{\lambda_0^2\varepsilon}\right)^2 \left[\frac{2}{1+\frac{1}{d}} \log \frac{\sigma_y \sigma_p l \kappa \mu (\lambda_0+1)}{\lambda_0^2 \varepsilon} + \log \frac{\beta_d}{\delta}\right]$$

Thanks for attention

Some References

1. M. Munkhoeva, E. Kapushev, E. Burnaev, I. Oseledets. Quadrature based features for kernel approximation. Proceedings of NIPS, Spotlight talk, 2018
2. E. Burnaev, P. Erofeev, A. Papanov. Influence of Resampling on Accuracy of Imbalanced Classification. ICMV, 2015
3. D. Smolyakov, A. Korotin, P. Erofeev, A. Papanov, E. Burnaev. Meta-learning for resampling recommendation systems, ICMV, 2019
4. Evgeny Burnaev. Generalization Bound for Imbalanced Classification. Springer Proceedings in Mathematics & Statistics, 2021
5. E. Burnaev. Rare Failure Prediction via Event Matching for Aerospace Applications. Proceedings of the 3rd International Conference on Circuits, System and Simulation (ICCSS-2019), pp. 214-220, 2019
6. E. Burnaev. On construction of early warning systems for predictive maintenance in aerospace industry. Journal of communications technology and electronics, 2019, Vol. 64, No. 12, pp. 1473-1484
7. D. Smolyakov, N. Sviridenko, V. Ishimtsev, E. Burikov, E. Burnaev. Learning Ensembles of Anomaly Detectors on Synthetic Data. ISNN 2019: Advances in Neural Networks – ISNN, Springer, 2019 pp 292-306
8. D. Smolyakov, N. Sviridenko, E. Burikov, E. Burnaev. Anomaly Pattern Recognition with Privileged Information for Sensor Fault Detection. 8th IAPR TC3 Workshop, ANNPR 2018, Siena, Italy, September 19–21, 2018, Springer LNCS Proceedings, Vol. 11081, pp. 320-332.
9. Burnaev E, Smolyakov D. One-Class SVM with Privileged Information and Its Application to Malware Detection // 16th International Conference on Data Mining Workshops (ICDMW), IEEE Conference Publications, pp. 273 - 280, 2016. DOI: 10.1109/ICDMW.2016.0046
10. E. Burnaev, P. Erofeev, D. Smolyakov. Model Selection for Anomaly Detection // Proc. SPIE 9875, Eighth International Conference on Machine Vision, 987525 (December 8, 2015); 5 P. doi:10.1117/12.2228794; http://dx.doi.org/10.1117/12.2228794