Robust Feature Selection and Robust PCA for Internet Traffic Anomaly Detection

C. Pascoal[1], M.R. Oliveira[1], R. Valadas[2],
P. Filzmoser[3], P. Salvador[4] and A. Pacheco[1]

[1]CEMAT - IST - UTL, [2]IT - IST - UTL, [3]TU Wien, [4]IT - UA

March 29, 2012

Outline

CEMAT
Dept. of Mathematics,
Tech. Univ. of Lisbon,
Portugal

Introduction

In our work, **“Robust”** means **“Robust”** in the **Statistical** sense

Introduction

In our work, **“Robust”** means **“Robust”** in the **Statistical** sense

- A **branch of Statistics** created by Huber and Hampel in the 80′s

Introduction

In our work, **“Robust”** means **“Robust”** in the **Statistical** sense

- A **branch of Statistics** created by Huber and Hampel in the 80′s
  - is an **alternative** approach to **Classical Statistics**

Introduction

In our work, **"Robust"** means **"Robust"** in the **Statistical** sense

- A **branch of Statistics** created by Huber and Hampel in the 80's
  - is an **alternative** approach to **Classical Statistics**
  - provides estimators that are **not unduly affected** by **outliers** or **small departures** from model assumptions

Introduction

In our work, **“Robust”** means **“Robust”** in the **Statistical** sense

- A **branch of Statistics** created by Huber and Hampel in the 80′s
  - is an **alternative** approach to **Classical Statistics**
  - provides estimators that are **not unduly affected** by **outliers** or **small departures** from model assumptions

Outlier detection

Introduction

In our work, **“Robust”** means **“Robust”** in the **statistical** sense

- A **branch of Statistics** created by Huber and Hampel in the 80′s
  - is an **alternative** approach to **Classical Statistics**
  - provides estimators that are **not unduly affected** by **outliers** or **small departures** from model assumptions

Feature selection **Outlier detection**

Introduction

In our work, **“Robust”** means **“Robust”** in the **statistical** sense

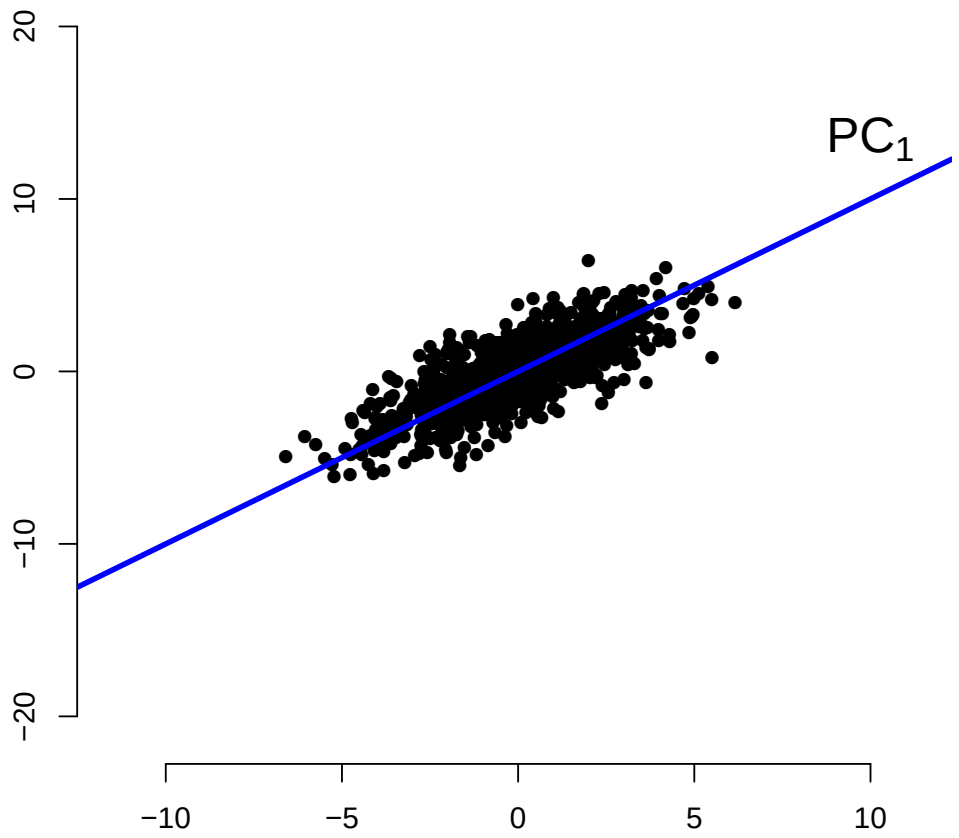
- A **branch of Statistics** created by Huber and Hampel in the 80′s
  - is an **alternative** approach to **Classical Statistics**
  - provides estimators that are **not unduly affected** by **outliers** or **small departures** from model assumptions

Robust **Robust**

Feature selection **Outlier detection**

Outlier Detection
PCA

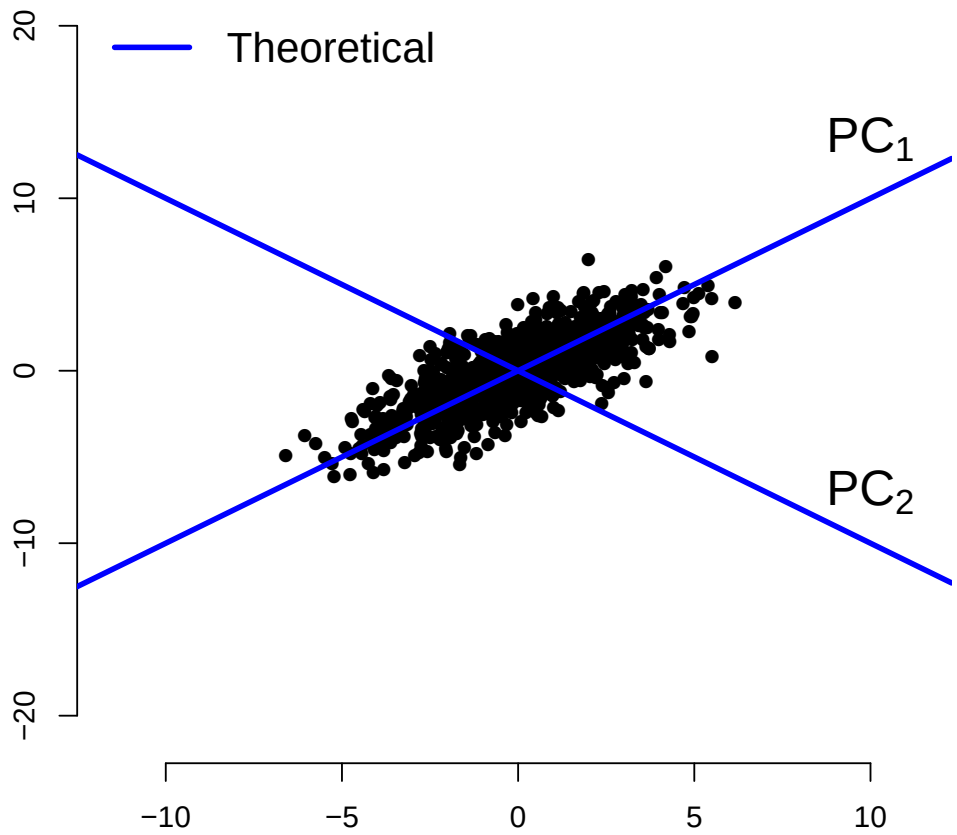
PC_1 - Linear combination of the features, with maximum variance



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Outlier Detection
PCA

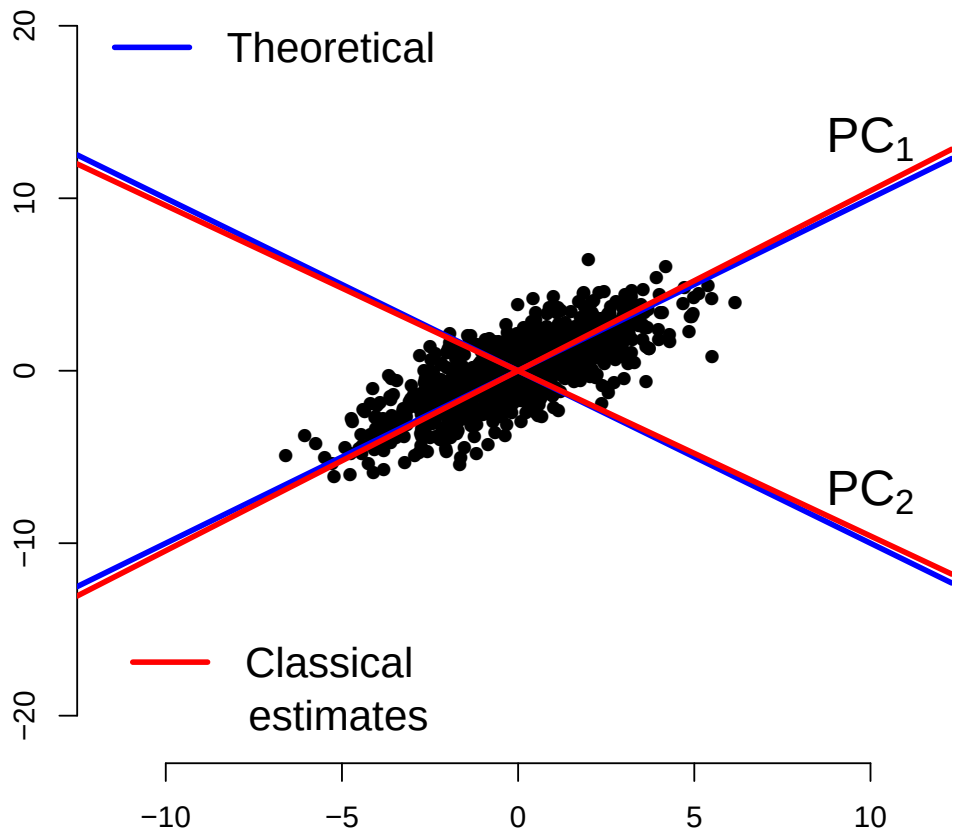
PC_2 - Orthogonal to PC_1, with maximum variance



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

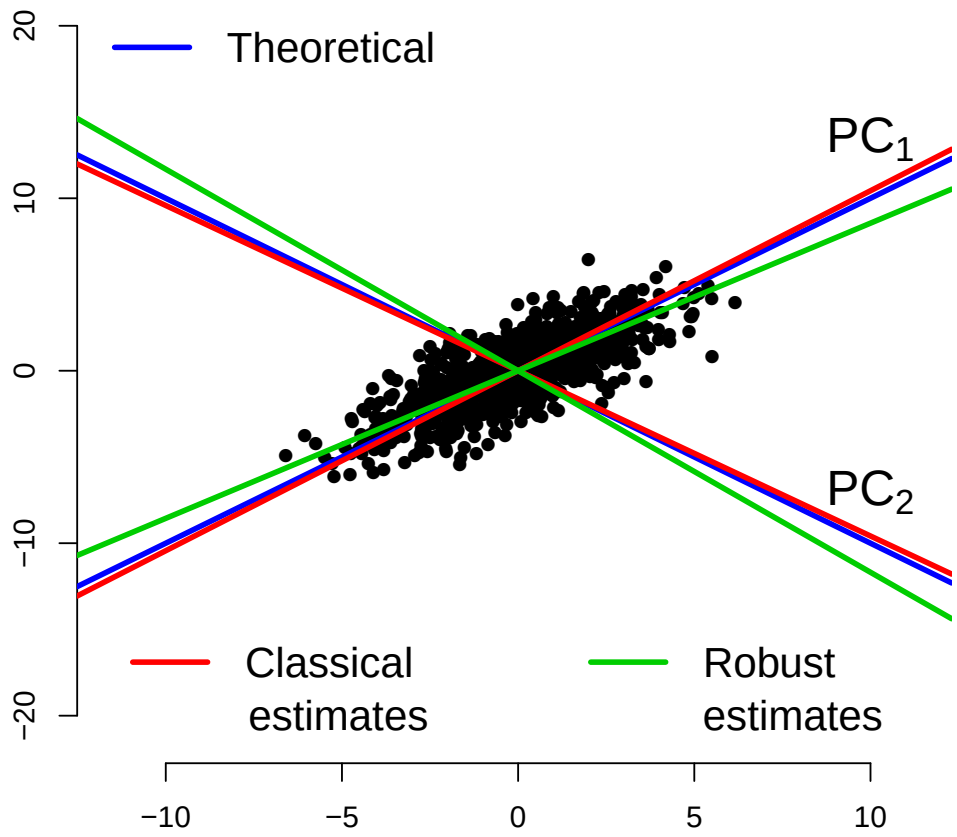
Outlier Detection
PCA

Classical estimator: **very sensitive to outliers**



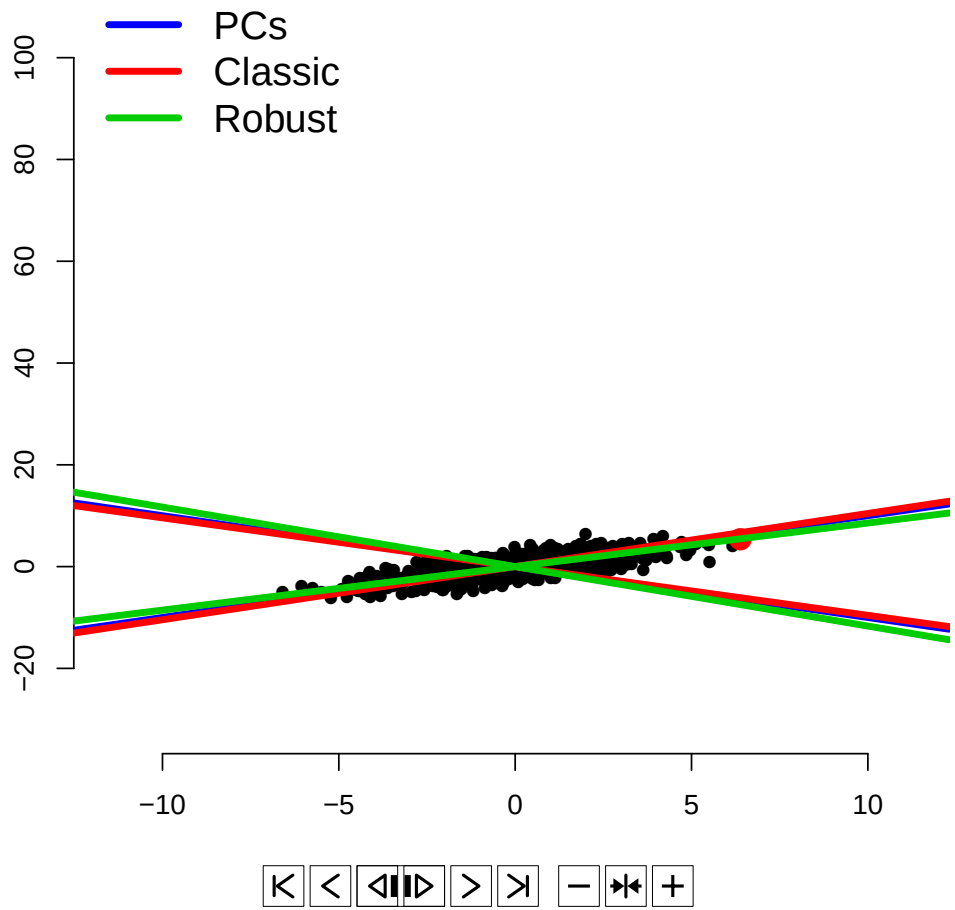
Outlier Detection
PCA

Robust estimator: **insensitive to a small amount of outliers**



Outlier Detection
PCA

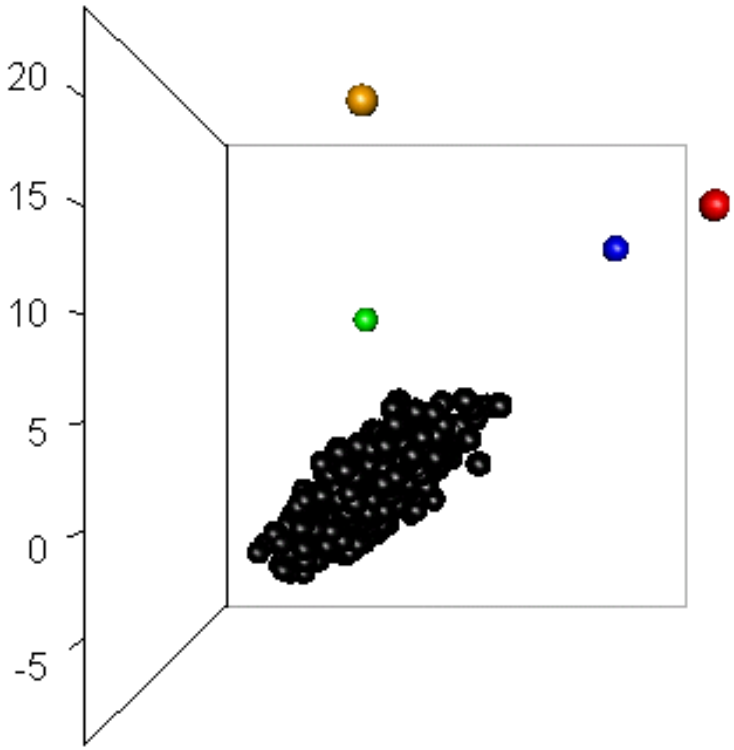
Effect of a single outlier:



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Outlier Detection
PCA

Toy example: How does PCA detect outliers?



Outlier Detection
PCA

Main idea: An **outlier** is a point **very far** from the majority of the data, **projected** into a proper space

Outlier Detection
PCA

Main idea: An **outlier** is a point **very far** from the majority of the data, **projected** into a proper space

Algorithm

Outlier Detection
PCA

Main idea: An **outlier** is a point **very far** from the majority of the data, **projected** into a proper space

Algorithm

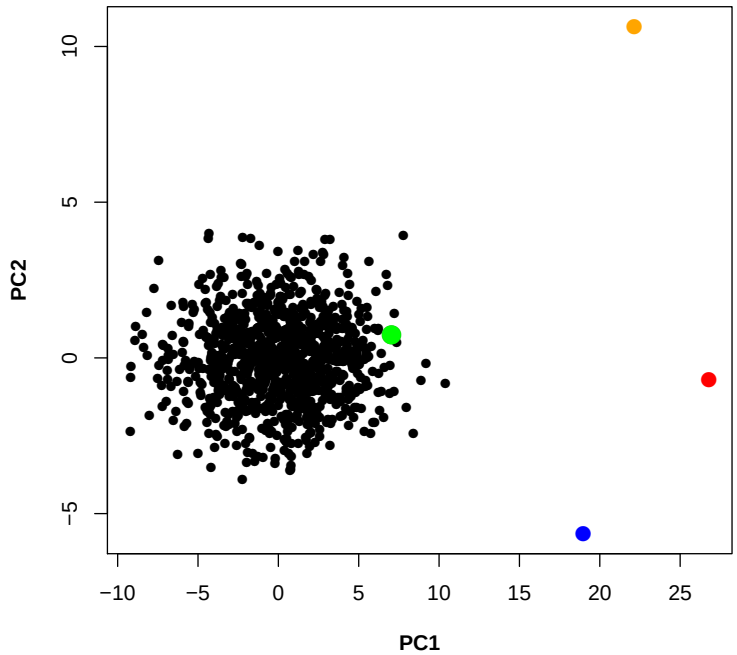
1. **Estimate** the first PCs and **project** the data into the main directions

Outlier Detection
PCA

Main idea: An **outlier** is a point **very far** from the majority of the data, **projected** into a proper space

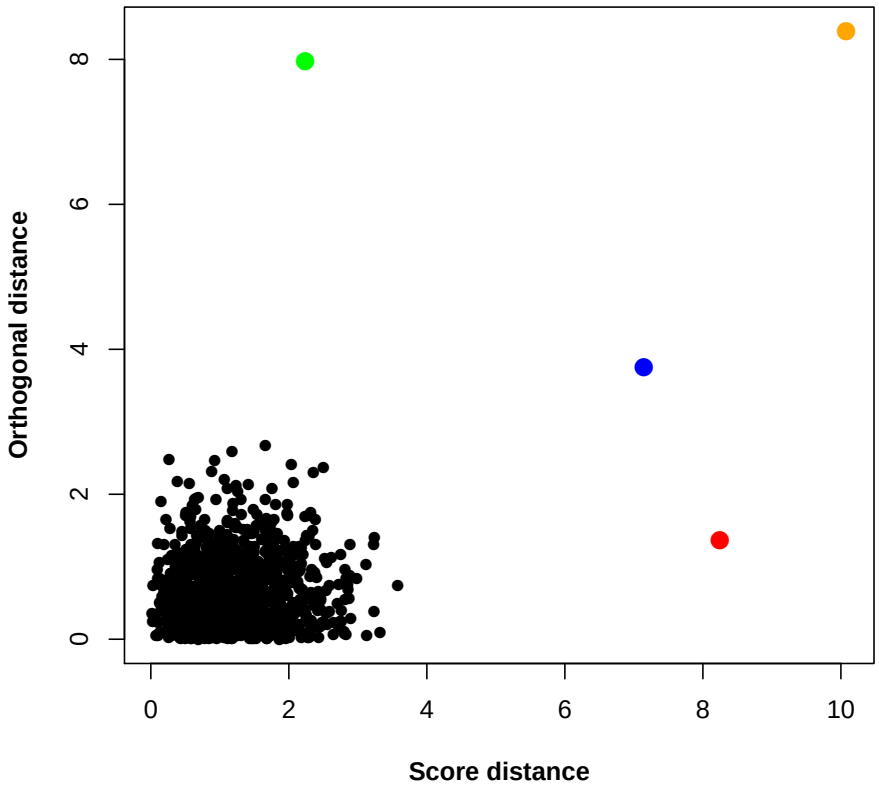
Algorithm

1. **Estimate** the first PCs and **project** the data into the main directions



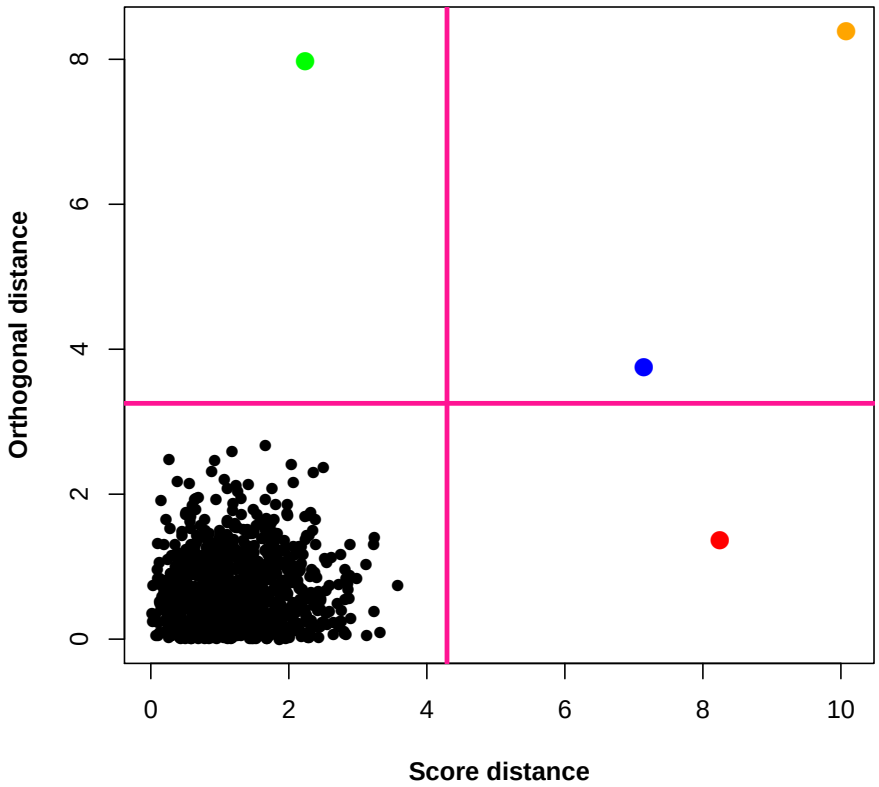
Outlier Detection
PCA

2. Calculate **Score distance** and **Orthogonal distance**



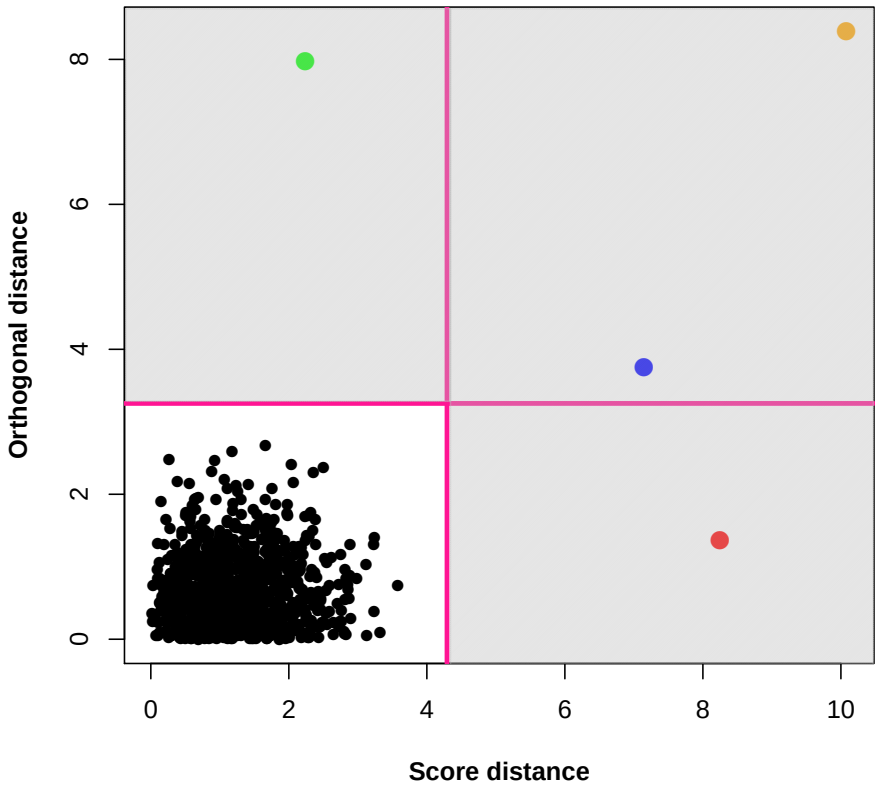
Outlier Detection
PCA

3. Calculate the **thresholds**



Outlier Detection
PCA

4. **Classify** an observation as **anomaly** or **regular**



Filter method

Feature Selection

Filter method

- Selects the **most informative features** to detect anomalies

Feature Selection

Filter method

- Selects the **most informative features** to detect anomalies
- Based on a measure of **association between features and classes** (**anomalous** or **regular**)

Feature Selection

Filter method

- Selects the **most informative features** to detect anomalies
- Based on a measure of **association between features and classes** (**anomalous** or **regular**)
- We use **Mutual Information**

Mutual Information (MI)

Mutual Information (MI)

- Index that measures **linear** and **non-linear** **dependencies** between features

Mutual Information (MI)

- Index that measures **linear** and **non-linear dependencies** between features
- The **MI** between X and Y is

Feature Selection

Mutual Information (MI)

- Index that measures **linear** and **non-linear** **dependencies** between features
- The **MI** between X and Y is

$$\mathrm{MI}(X, Y) = \iint f_{XY}(x, y) \ln\left(\frac{f_{XY}(x, y)}{f_X(x) f_Y(y)}\right) \mathrm{d}x \, \mathrm{d}y$$

Feature Selection

Mutual Information (MI)

- Index that measures **linear** and **non-linear** **dependencies** between features
- The **MI** between X and Y is

$$\mathrm{MI}(X, Y) = \iint f_{XY}(x, y) \ln\left(\frac{f_{XY}(x, y)}{f_X(x) f_Y(y)}\right) \mathrm{d}x \, \mathrm{d}y$$

Estimation

- When the joint probability density function $f_{XY}(x, y)$ is **unknown**...

Feature Selection

Mutual Information (MI)

- Index that measures **linear** and **non-linear** **dependencies** between features
- The **MI** between X and Y is

$$\mathrm{MI}(X, Y) = \iint f_{XY}(x, y) \ln\left(\frac{f_{XY}(x, y)}{f_X(x) f_Y(y)}\right) \mathrm{d}x \, \mathrm{d}y$$

Estimation

- When the joint probability density function $f_{XY}(x, y)$ is **unknown**...several **estimators** were **proposed**!

Feature Selection

Mutual Information (MI)

- Index that measures **linear** and **non-linear** **dependencies** between features
- The **MI** between X and Y is

$$\mathrm{MI}(X, Y) = \int\int f_{XY}(x, y) \ln\left(\frac{f_{XY}(x, y)}{f_X(x) f_Y(y)}\right) \mathrm{d}x \, \mathrm{d}y$$

Estimation

- When the joint probability density function $f_{XY}(x, y)$ is **unknown**...several **estimators** were **proposed**!
- There is **no single best estimator**!

Feature Selection

Mutual Information (MI)

- Index that measures **linear** and **non-linear** **dependencies** between features
- The **MI** between X and Y is

$$\mathrm{MI}(X, Y) = \iint f_{XY}(x, y) \ln\left(\frac{f_{XY}(x, y)}{f_X(x) f_Y(y)}\right) \mathrm{d}x \, \mathrm{d}y$$

Estimation

- When the joint probability density function $f_{XY}(x, y)$ is **unknown**...several **estimators** were **proposed**!
- There is **no single best estimator**!
- **Sensitive**

Feature Selection

Mutual Information (MI)

- Index that measures **linear** and **non-linear** **dependencies** between features
- The **MI** between X and Y is

$$\mathrm{MI}(X, Y) = \int\int f_{XY}(x, y) \ln\left(\frac{f_{XY}(x, y)}{f_X(x) f_Y(y)}\right) \mathrm{d}x \, \mathrm{d}y$$

Estimation

- When the joint probability density function $f_{XY}(x, y)$ is **unknown**...several **estimators** were **proposed**!
- There is **no single best estimator**!
- **Sensitive**
  - to **atypical observations** in each **class**
  - to **mislabeling errors**

Robustification of the MI estimator

Robustification of the MI estimator

Main idea: **Exclude** outliers from the data used to estimate **MI**

Robustification of the MI estimator

Main idea: **Exclude** outliers from the data used to estimate **MI**

Algorithm

Robustification of the MI estimator

Main idea: **Exclude** outliers from the data used to estimate **MI**

Algorithm

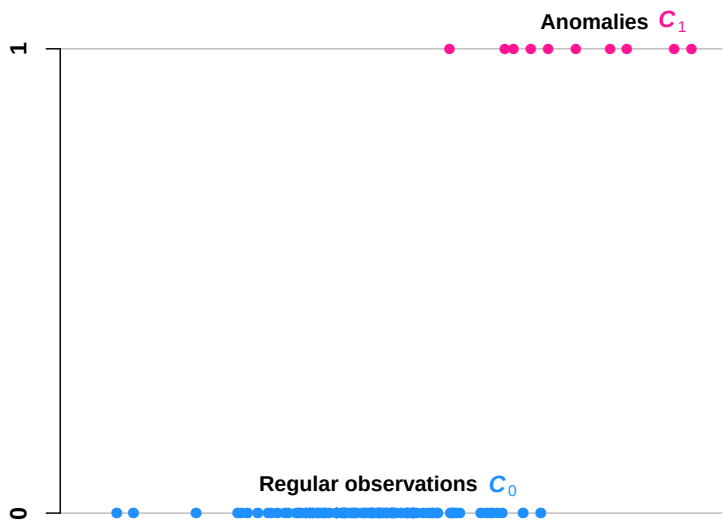
1. Given a **labeled data set**, **separate** in 2 subsets:

Robustification of the MI estimator

Main idea: **Exclude** outliers from the data used to estimate **MI**

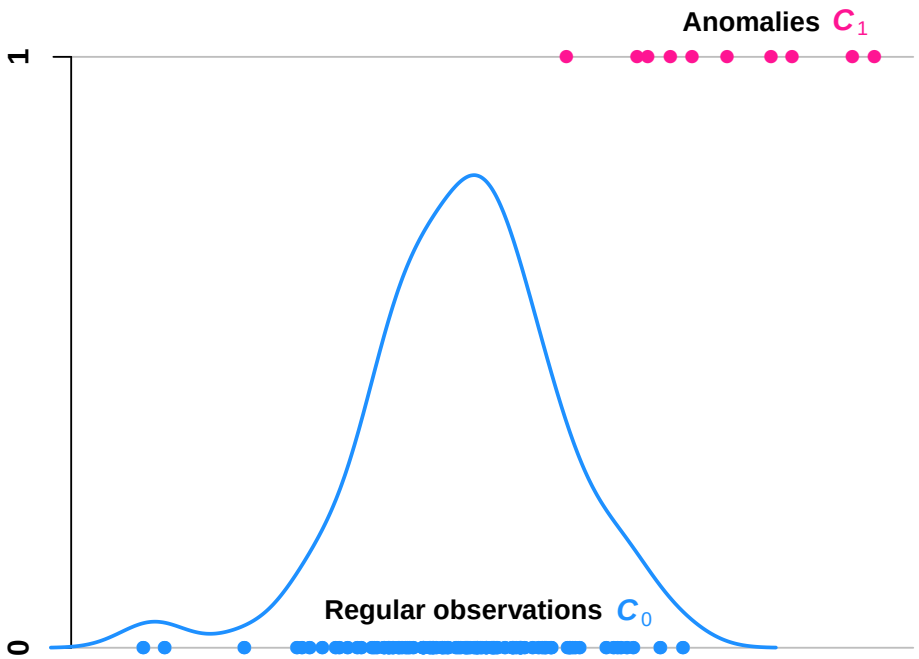
Algorithm

1. Given a **labeled data set**, **separate** in 2 subsets:
   - $\mathcal{C}_0$ - regular observations (**label** 0)
   - $\mathcal{C}_1$ - anomalies (**label** 1)



Feature Selection
Robustification of the MI estimator

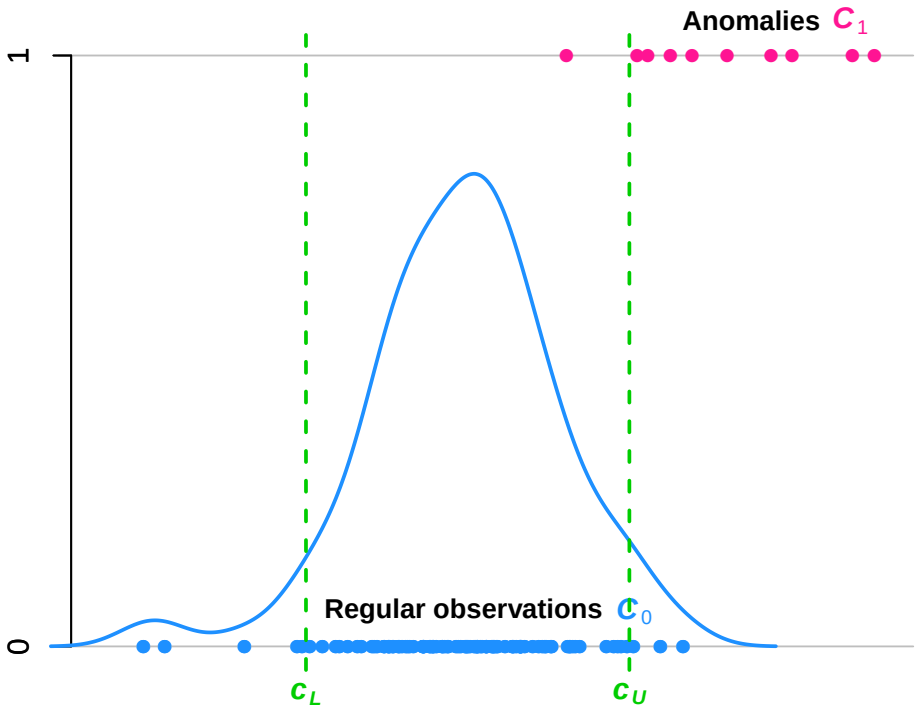
2. Assume that **regular observations** have **approximately** normal distribution



Feature Selection
Robustification of the MI estimator

Main idea: Identify **outliers** among **regular observations**

3. Calculate **robust CI**, $[c_L, c_U] = \mathbf{med}\,(\mathcal{C}_0) \pm 1.96\mathbf{mad}\,(\mathcal{C}_0)$

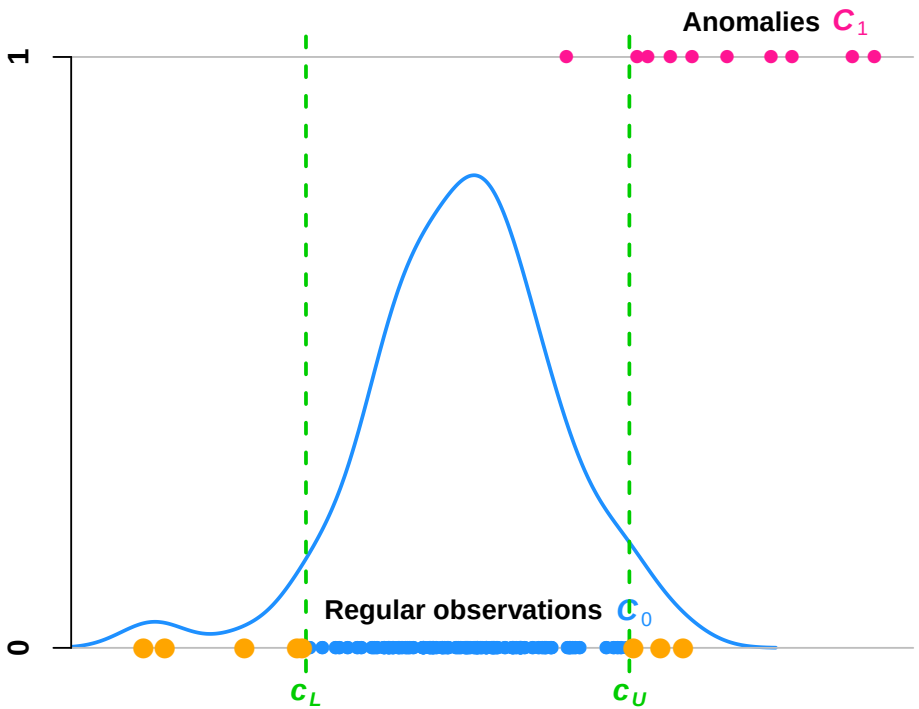


CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Feature Selection
Robustification of the MI estimator

Main idea: Identify **outliers** among **regular observations**

4. **Exclude** from $\mathcal{C}_0$ the observations **outside CI**

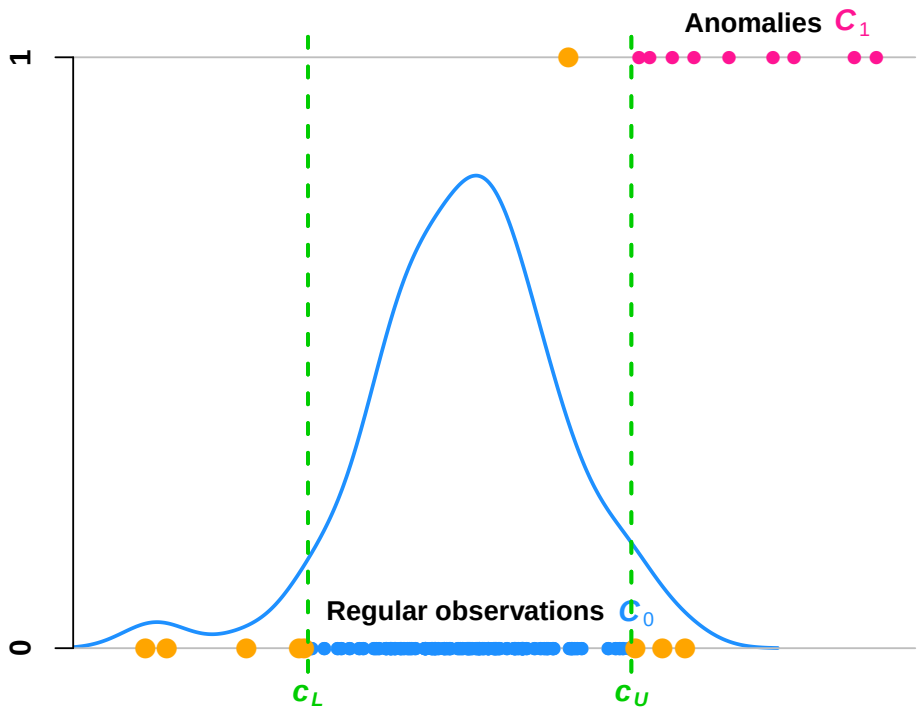


CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Feature Selection
Robustification of the MI estimator

Main idea: Identify potentially **mislabeled anomalies**

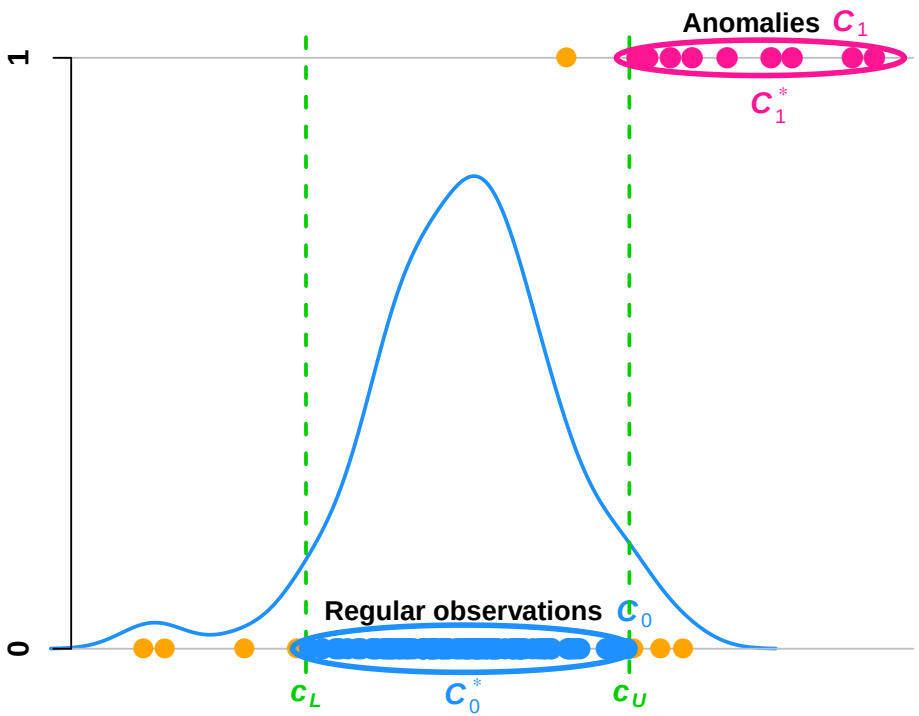
5. **Exclude** from $\mathcal{C}_1$ the observations **within CI**



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Feature Selection
Robustification of the MI estimator

6. **Estimate MI** based on $\mathcal{C}_0^* \cup \mathcal{C}_1^*$



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Proposed methods

Proposed methods

- Number of features **fixed in advance**

Proposed methods

- Number of features **fixed in advance**
- **User-defined threshold** (fixed in advance) for **MI**

Proposed methods

- Number of features **fixed in advance**
- **User-defined threshold** (fixed in advance) for **MI**
- **PCA**, ...

Proposed methods

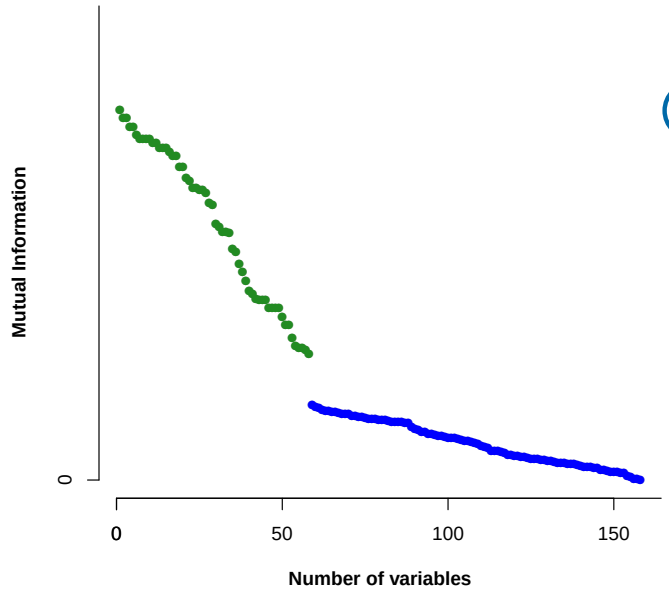
- Number of features **fixed in advance**
- **User-defined threshold** (fixed in advance) for **MI**
- **PCA**, ...

Automatic method to select relevant features

Automatic method to select features

Automatic method to select features

- **Partition** the ordered features in 2 **subsets**:
  - **Relevant features**: highest MI
  - **Non-relevant features**: others
- Search for the partition with **highest separation** between subsets (**sharp drop** on the orderer MI)

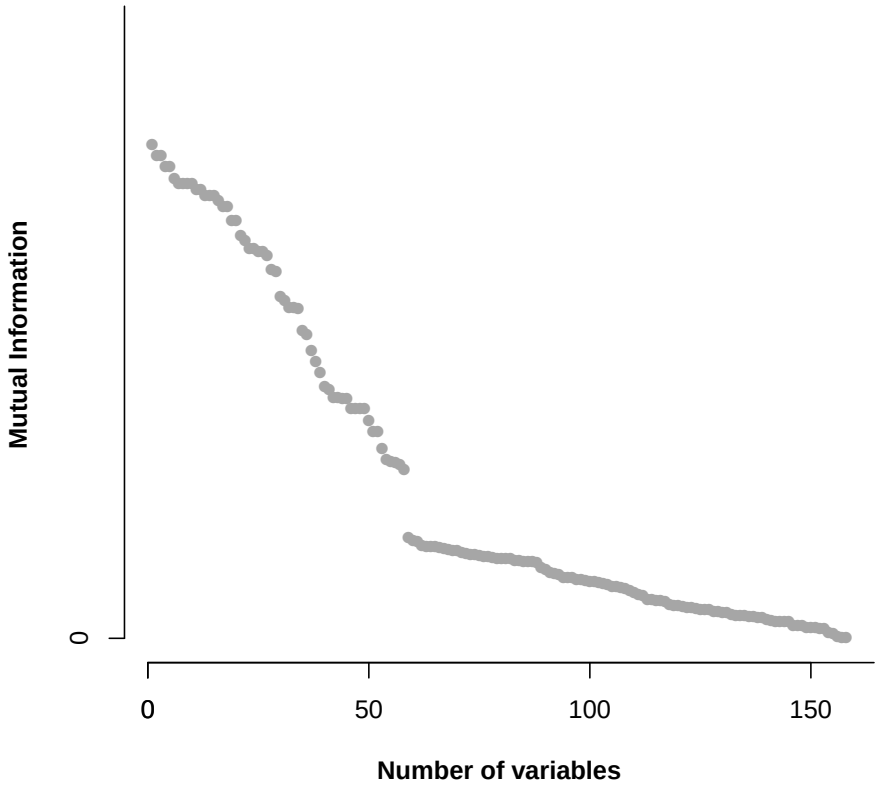


Algorithm

1. **Order the MI estimates**, $\boldsymbol{mi} = \left(mi_{(1)}, \ldots, mi_{(p)}\right)$

Algorithm

1. **Order the MI estimates**, $\boldsymbol{mi} = \left(mi_{(1)}, \ldots, mi_{(p)}\right)$



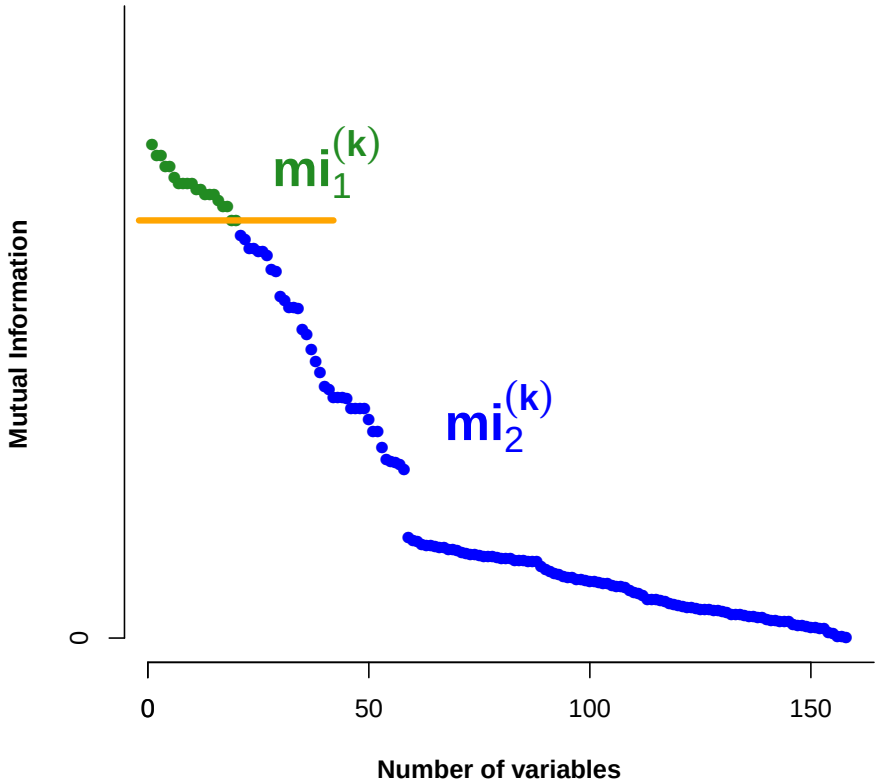
CEMAT
Dept. of Mathematics,
Tech. Univ. of Lisbon,
Portugal

Feature Selection
Automatic method to select features

2. **Separate *mi* in 2 groups**: $\boldsymbol{mi}_1^{(k)} = \left(mi_{(1)}, \ldots, mi_{(k)}\right)$ and $\boldsymbol{mi}_2^{(k)} = \left(mi_{(k+1)}, \ldots, mi_{(p)}\right)$

Feature Selection
Automatic method to select features

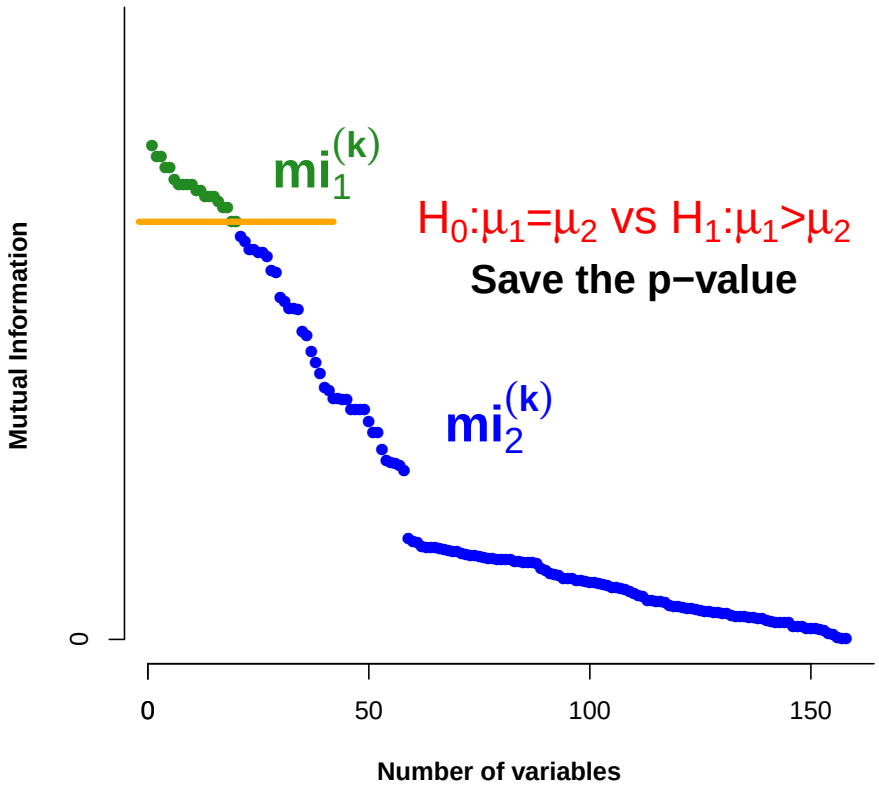
2. **Separate *mi* in 2 groups**: $\boldsymbol{mi}_1^{(k)} = \left(mi_{(1)}, \ldots, mi_{(k)}\right)$ and $\boldsymbol{mi}_2^{(k)} = \left(mi_{(k+1)}, \ldots, mi_{(p)}\right)$



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

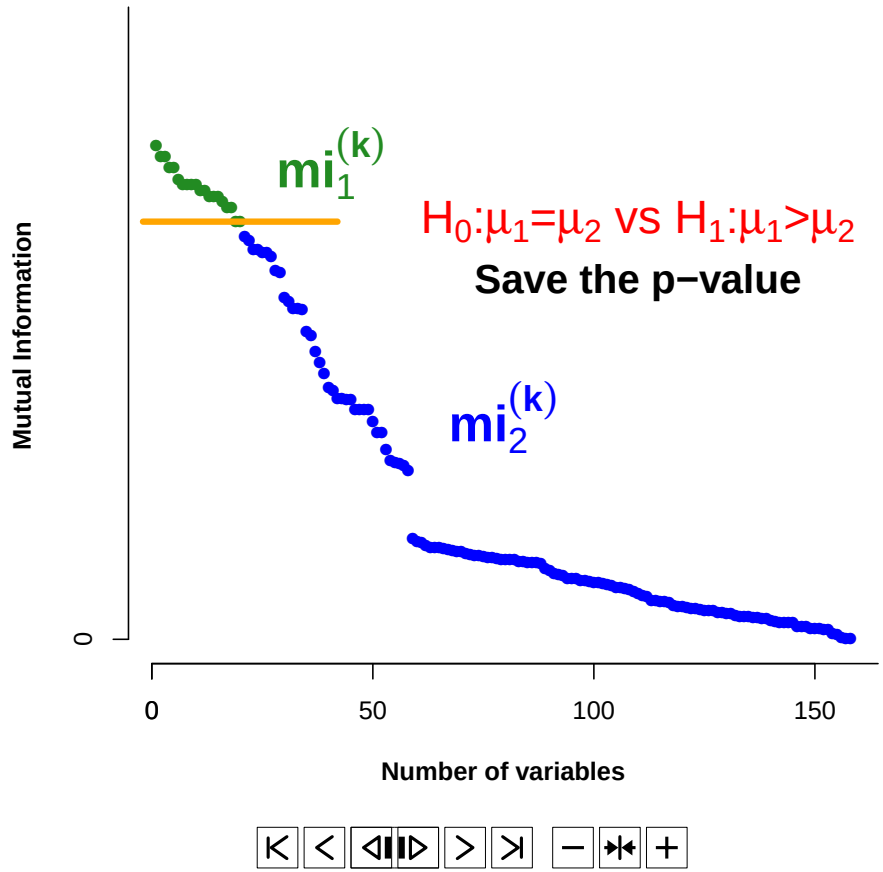
Feature Selection
Automatic method to select features

3. **Compute p-value of the one-sided *t-test*** (difference in means)



Feature Selection
Automatic method to select features

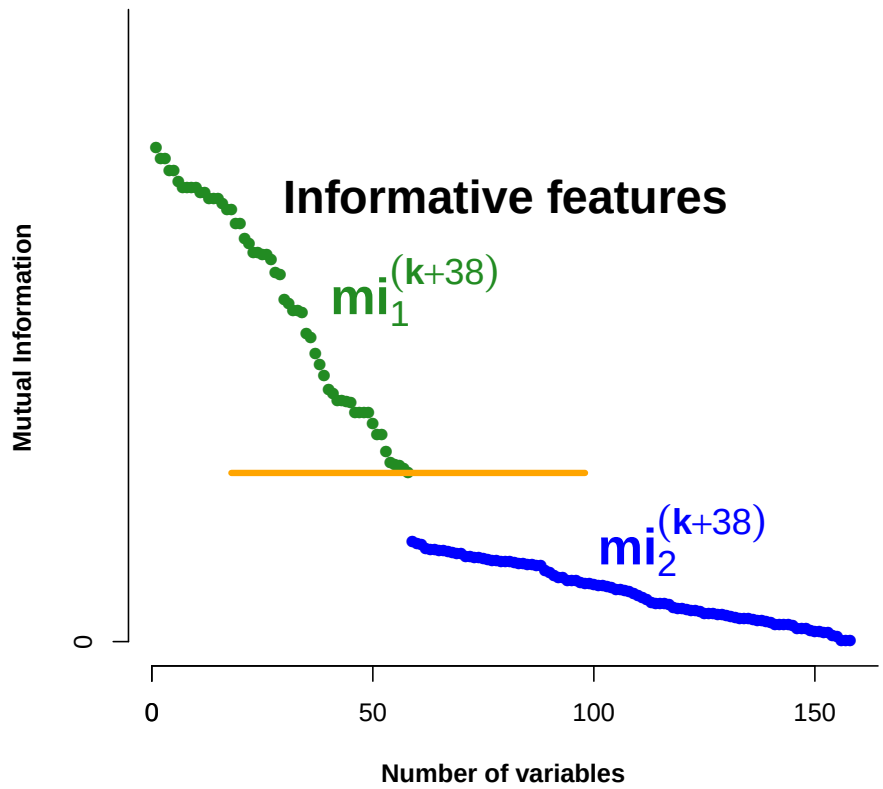
4. **Update** $mi_1^{(k+1)}$ and $mi_2^{(k+1)}$ and **repeat** step 2., 3. and 4.



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Feature Selection
Automatic method to select features

5. **Choose k corresponding to the lowest p-value**, highest separation



CEMAT
Dept. of Mathematics, Tech. Univ. of Lisbon, Portugal

Results
Objects and features

- To **evaluate** our method we need a data set with **all traffic objects correctly labeled** - **perfect ground-truth**

Results
Objects and features

- To **evaluate** our method we need a data set with **all traffic objects correctly labeled** - **perfect ground-truth**
- **Impossible** to obtain with real data

Results
Objects and features

- To **evaluate** our method we need a data set with **all traffic objects correctly labeled** - **perfect ground-truth**
- **Impossible** to obtain with real data

we arranged a **small private laboratory network** with **highly-protected** Internet access (CISCO ASA 5510)

Traffic types:

Traffic types:

Licit Traffic

Traffic types:

Licit Traffic

- **10 users** invited to **generate traffic** from inside lab network
- **Mix** of predominant Internet **applications**:
  - File sharing (**BitTorrent**)
  - Video **Streaming**
  - Web browsing (**HTTP**)

Traffic types:

Licit Traffic

- **10 users** invited to **generate traffic** from inside lab network
- **Mix** of predominant Internet **applications**:
  - File sharing (**BitTorrent**)
  - Video **Streaming**
  - Web browsing (**HTTP**)

Attacks

Traffic types:

Licit Traffic

- **10 users** invited to **generate traffic** from inside lab network
- **Mix** of predominant Internet **applications**:
  - File sharing (**BitTorrent**)
  - Video **Streaming**
  - Web browsing (**HTTP**)

Attacks

- **Produced** only within the lab network
- **2** broad classes of **attacks**:
  - **Port-scan** (NMAP)
  - **Snapshots** (emulated using FTP)

Results
Dataset, traffic object definition and features

Dataset: captured **March 2**, **2011**, for **8 hours**

Dataset: captured **March 2**, **2011**, for **8 hours**

Traffic objects

Results
Dataset, traffic object definition and features

Dataset: captured **March 2**, **2011**, for **8 hours**

Traffic objects

- **Aggregates** all traffic of one application that **enters or leaves** one machine in 5 minutes (***datastream***)
  - The **same IP** source address
  - **One** of the **TCP port** numbers **equal**

Dataset: captured **March 2**, **2011**, for **8 hours**

Traffic objects

- **Aggregates** all traffic of one application that **enters or leaves** one machine in 5 minutes (***datastream***)
  - The **same IP** source address
  - **One** of the **TCP port** numbers **equal**

Features

Dataset: captured **March 2**, **2011**, for **8 hours**

Traffic objects

- **Aggregates** all traffic of one application that **enters or leaves** one machine in 5 minutes (***datastream***)
  - The **same IP** source address
  - **One** of the **TCP port** numbers **equal**

Features

- **5 traffic characteristics** (computed in 0.1 seconds interval):
  - Nr **packets**: upstream (PUp) and downstream (PDw)
  - Nr **bytes**: upstream (BUp) downstream (BDw)
  - Nr **active TCP sessions** (Ses)
- **8 summary statistics**: min, Q_1, med, mean, Q_3, max, sd, MAD

Results
Performance evaluation

Scenario	Detector	Nr Ftrs	Performance measures		
			Recall	FPR	Precision
B1	**∅-NR**	-	1	0.063	0.500
	∅-R	-	1	0.063	0.500
	NR-NR	17	**1**	**0**	**1**
	NR-R	17	**1**	**0**	**1**
	R-NR	5	**1**	**0**	**1**
	R-R	5	**1**	**0**	**1**
B2	**∅-NR**	-	0.167	0	1
	∅-R	-	1	0.679	0.240
	NR-NR	13	0.167	0	1
	NR-R	13	1	0	1
	R-NR	7	0.167	0	1
	R-R	7	**1**	**0**	**1**
B3	**∅-NR**	-	0.273	0	1
	∅-R	-	**0.818**	**0.783**	**0.333**
	NR-NR	18	0.273	0	1
	NR-R	18	0.454	0.522	0.294
	R-NR	7	0.273	0.044	0.750
	R-R	7	**1**	**0**	**1**

Results
Performance evaluation

Scenario	Detector	Nr Ftrs	Performance measures		
			Recall	FPR	Precision
B1	**∅-NR**	-	1	0.063	0.500
	∅-R	-	1	0.063	0.500
	NR-NR	17	1	0	1
	NR-R	17	1	0	1
	R-NR	5	1	0	1
	R-R	5	**1**	**0**	**1**
B2	**∅-NR**	-	0.167	0	1
	∅-R	-	1	0.679	0.240
	NR-NR	13	0.167	0	1
	NR-R	13	1	0	1
	R-NR	7	0.167	0	1
	R-R	7	**1**	**0**	**1**
B3	**∅-NR**	-	0.273	0	1
	∅-R	-	0.818	0.783	0.333
	NR-NR	18	0.273	0	1
	NR-R	18	0.454	0.522	0.294
	R-NR	7	**0.273**	**0.044**	**0.750**
	R-R	7	**1**	**0**	**1**

Results
Performance evaluation

Scenario	Detector	Nr Ftrs	Performance measures		
			Recall	FPR	Precision
R1	**∅-NR**	-	1	0.033	0.600
	∅-R	-	1	0.100	0.333
	NR-NR	15	1	0.067	0.429
	NR-R	15	1	0.100	0.333
	R-NR	10	**1**	**0**	**1**
	R-R	10	1	0.017	0.750
R2	**∅-NR**	-	0.417	0	1
	∅-R	-	1	0.600	0.286
	NR-NR	5	1	0.020	0.923
	NR-R	5	1	0.040	0.857
	R-NR	10	**1**	**0**	**1**
	R-R	10	1	0.040	0.857
R3	**∅-NR**	-	0.125	0	1
	∅-R	-	0.875	0.786	0.298
	NR-NR	6	0.125	0	1
	NR-R	6	0.688	0.476	0.355
	R-NR	7	0.063	0	1
	R-R	7	**1**	**0**	**1**

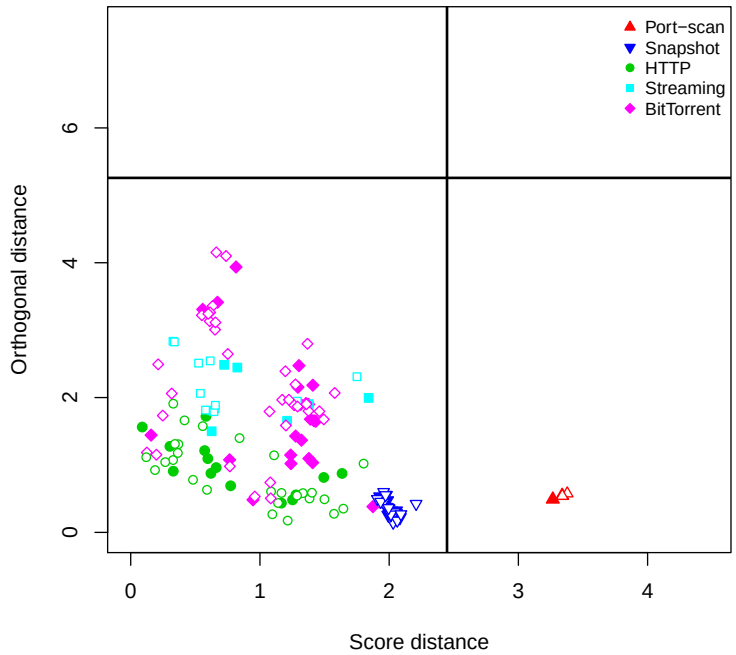
R3

Detector: R-NR

Features: 7

- **Ses**: Q_3, max, sd, MAD
- **BUp**: max, sd
- **BDw**: mean

Recall	False Positive	Precision
0.125 (0.125)	0 (0)	1 (1)



Results
Performance evaluation

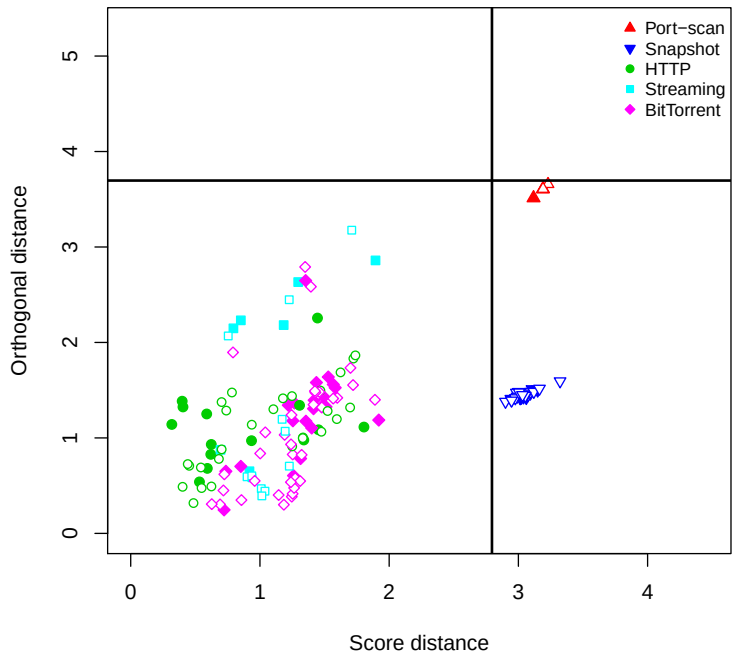
R3

Detector: R-R

Features: 7

- **Ses**: Q_3, max, sd, MAD
- **BUp**: max, sd
- **BDw**: mean

Recall	False Positive	Precision
1 (0.875)	**0** (0.786)	**1** (0.298)



Conclusions

- **Feature selection** is found to be an **important preprocessing step**, based on **robust MI**, which leads to **better results**

Conclusions

- **Feature selection** is found to be an **important preprocessing step**, based on **robust MI**, which leads to **better results**
- The combination of **robust feature selection** and **robust PCA outlier detection**:

Conclusions

- **Feature selection** is found to be an **important preprocessing step**, based on **robust MI**, which leads to **better results**
- The combination of **robust feature selection** and **robust PCA outlier detection**:
  - achieves **very high performance**
  - **adaptive** to different traffic conditions

Conclusions

- **Feature selection** is found to be an **important preprocessing step**, based on **robust MI**, which leads to **better results**
- The combination of **robust feature selection** and **robust PCA outlier detection**:
  - achieves **very high performance**
  - **adaptive** to different traffic conditions
- **Robust statistics** brings effective **tools** for anomaly detection
  - can be **helpful** in addressing other related network problems

Thank you
Obrigada