PCA / Anomaly Detection

Anastasios Giovanidis

Sorbonne-LIP6

Mai 15, 2020

Bibliography

A. Giovanidis 2020

B.1 Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. “An introduction to statistical learning: with applications in R”. Springer Texts in Statistics.

☞ Chapter 10
ISBN 978-1-4614-7137-0 (DOI 10.1007/978-1-4614-7138-7)

B.2 Qi Liao, and Slawomir Stanczak. “Network State Awareness and Proactive Anomaly Detection in Self-Organizing Networks”. IEEE Globecom 2015 Workshops,
DOI: 10.1109/GLOCOMW.2015.7414141

B.3 Anukool Lakhina, Mark Crovella, and Christophe Diot. “Diagnosing Network-Wide Traffic Anomalies”. SIGCOMM ’04. pp. 219-230
https://doi.org/10.1145/1015467.1015492

B.4 Code Practicals: ”Data mining of network measurements”,
https://github.com/mcrovella/mining-low-dim-network-data

PCA

A. Giovanidis 2020

The idea behind PCA is to find a low-dimensional set of axes that summarise data (unsupervised - no labels). Why?

- Many features in the data set can be highly correlated: No need to keep both.
- Other features may have very low variance and do not sufficiently differentiate between possible classes.
- Remove the redundancy, describe the data-set with less properties.

☞ PCA only looks at feature variance: features that present high variance are more likely to have a good split between classes.

How does it work? (I)

A. Giovanidis 2020

PCA **is not** feature selection.

☞ It rather constructs a new set of properties with reduced dimension based on a **linear combination** of the total number of features.

- PCA performs a linear transformation moving the original set of features to a new (reduced) space.
- The new space is composed by some principal components.
- These new features do not have real meaning, only algebraic.

☞ Principal components are **orthogonal to each other**, thus uncorrelated.

How does it work? (II)

A. Giovanidis 2020

Perform **Singular Value Decomposition** (SVD), providing a set of **singular vectors** and its respective **singular values** as a result.

- The (left- or right-) singular vectors represent the new set of axes of the principal component space and the singular values carry the quantity of variance that each singular vector has.
- To reduce the dimension: keep those singular vectors with more variance and discard those with less variance.

PCA Method (I)

A. Giovanidis 2020

- Data-set $\{a_j\}$, $j = 1, \ldots, n$. Each a_j has m features.
- Let $\mathbf{A} := [a_1, \ldots, a_n]$ be the $m \times n$ matrix.

Method:

- Center each row (feature) of the $\mathbf{A}$ matrix (row or column centering depends on the application).
- Perform SVD of $\mathbf{A}$,

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T. \tag{1}$$

$\mathbf{\Sigma}$ is a diagonal matrix with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r \geq 0$. With $r = rank(\mathbf{A})$.
$\mathbf{U}$ is $m \times r$. It is unitary with orthogonal columns, it holds $\mathbf{U}^T\mathbf{U} = \mathbf{I}_{r \times r}$.
$\mathbf{V}$ is $n \times r$. It is unitary with orthogonal columns, it holds $\mathbf{V}^T\mathbf{V} = \mathbf{I}_{r \times r}$.

PCA Method (II)

A. Giovanidis 2020

Using SVD we have written each data-object a_j (i.e. here $m \times 1$ column of the matrix $\mathbf{A}$) as a linear combination of the r column m-vectors $\mathbf{u}$. Note that $r = min\{m, n\}$.

$$a_j \quad = \quad v_{j,1}\sigma_1\mathbf{u}_1 + \ldots v_{j,2}\sigma_2\mathbf{u}_2 + \ldots + v_{j,r}\sigma_r\mathbf{u}_r \tag{2}$$

The idea of PCA is to remove the least-significant components with low weight (singular values) σ and just keep the $k < r$ first principal components, so that the data-object can be approximated by

$$a_j^{(k)} \quad = \quad v_{j,1}\sigma_1\mathbf{u}_1 + \ldots v_{j,2}\sigma_2\mathbf{u}_2 + \ldots + v_{j,k}\sigma_k\mathbf{u}_k. \tag{3}$$

PCA Method (III)

A. Giovanidis 2020

We can approximate the original data-matrix, by keeping only the $k < r$ first (largest) singular values.

- Let $k < r$ be the desired approximation. Then,

$$\mathbf{A}^{(k)} = \mathbf{U}_k \mathbf{\Sigma}_k (\mathbf{V}_k)^T. \tag{4}$$

$\mathbf{\Sigma}_k$ is the $k \times k$ upper-left submatrix of $\mathbf{\Sigma}$ with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k$.
$\mathbf{U}_k$ is $m \times k$. The left k columns of $\mathbf{U}$.
$\mathbf{V}_k$ is $n \times k$. The left k columns of $\mathbf{V}$.

Note that $\mathbf{A}^{(k)}$ is again $m \times n$, it has the same dimensions as $\mathbf{A}$.

Approximation Error

A. Giovanidis 2020

How well does $\mathbf{A}^{(k)}$ approximate $\mathbf{A}$? Use the Frobenius norm of the error:

$$\begin{aligned}
\left\|\mathbf{A}-\mathbf{A}^{(k)}\right\|_F &= \left\|\mathbf{U\Sigma V}^T-\mathbf{U}_k\mathbf{\Sigma}_k(\mathbf{V}_k)^T\right\|_F \\
&= \left\|[\mathbf{U}_k\mathbf{U}_{r-k}]\begin{bmatrix}\mathbf{\Sigma}_k & 0\\ 0 & \mathbf{\Sigma}_{r-k}\end{bmatrix}\begin{bmatrix}\mathbf{V}_k^T\\ \mathbf{V}_{r-k}^T\end{bmatrix}-\mathbf{U}_k\mathbf{\Sigma}_k(\mathbf{V}_k)^T\right\|_F \\
&= \left\|\mathbf{U}_{r-k}\mathbf{\Sigma}_{r-k}\mathbf{V}_{r-k}^T\right\|_F \\
&= \left\|\mathbf{\Sigma}_{r-k}\right\|_F \\
&= \sqrt{\sigma_{r-k}^2+\sigma_{r-k+1}^2+\ldots+\sigma_r^2}.
\end{aligned} \tag{5}$$

- Normalised error: $\frac{\|\mathbf{\Sigma}_{r-k}\|_F}{\|\mathbf{A}\|_F}$.

A. Giovanidis 2020

New data dimension k

If $\mathbf{A}^{(k)}$ is again $m \times n$, the same dimensions as $\mathbf{A}$, what did we change?

We can obtain a new data-set with reduced dimension k, from the original one. To find this, let us project the initial data-set $\mathbf{A}$ on the new space of $k < r$ principal components:

$$\begin{aligned}
\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^T \quad \Rightarrow \quad & (\mathbf{U}_k)^T\mathbf{A} = (\mathbf{U}_k)^T\mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^T \Rightarrow \\
(\mathbf{U}_k)^T\mathbf{A} \quad &= \quad \left[\mathbf{I}_{k\times k},\ 0_{k\times(r-k)}\right]\boldsymbol{\Sigma}(\mathbf{V})^T \Rightarrow \\
(\mathbf{U}_k)^T\mathbf{A} \quad &= \quad \left[\boldsymbol{\Sigma}_{k\times k},\ 0_{k\times(r-k)}\right]\begin{bmatrix} \mathbf{V}_k^T \\ \mathbf{V}_{r-k}^T \end{bmatrix} \Rightarrow \\
(\mathbf{U}_k)^T\mathbf{A} \quad &= \quad \boldsymbol{\Sigma}_{k\times k}(\mathbf{V}_k)^T \qquad (6)
\end{aligned}$$

Take data-sample $a_j = (a_{1,j}, \ldots, a_{m,j})^T$, for some $1 \leq j \leq n$.
By projecting the sample a_j on the matrix $(\mathbf{U}_k)^T$ we result in a description with dimension k, that is $\tilde{a}_j = (\tilde{a}_{1,j}, \ldots, \tilde{a}_{k,j})^T$.

Definitions

A. Giovanidis 2020

We have projected the a_j data sample on the space described by the first k (orthogonal) columns of $\mathbf{U}$. The projected data-sample has $k < m$ new features.

☞ The principal component of the set of original features $a_{1,j}, \ldots, a_{m,j}$ is the normalized linear combination of the features

$$\mathbf{u}_1^T a_j \quad = \quad u_{1,1} a_{1,j} + \ldots + u_{m,1} a_{m,j},$$

with the largest variance. Here $\sum_{j=1}^{m} u_{j,1}^2 = 1$.

☞ Also, $\mathbf{u}_2^T = (u_{1,2}, u_{2,2}, \ldots, u_{m,2})$ is the second component, or the direction of second maximum data variance, and so on...

The reduced dimensionality data-set with $k \times n$ shape is given by

$$\tilde{\mathbf{A}}_{k\times n} \quad = \quad \mathbf{\Sigma}_k (\mathbf{V}_k)^T. \tag{7}$$

A. Giovanidis 2020

Alternative

Another way to obtain the new data-set with reduced dimension k, is by projecting the initial data-set $\mathbf{A}$ on $\mathbf{V}_k$:

$$\begin{aligned}\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^T \quad \Rightarrow \quad \mathbf{A}\mathbf{V}_k &= \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^T\mathbf{V}_k \Rightarrow \\ \mathbf{A}\mathbf{V}_k \quad &= \quad \mathbf{U}_k\boldsymbol{\Sigma}_{k\times k} \end{aligned} \tag{8}$$

Take the feature vector $a_i = (a_{i,1}, \dots, a_{i,n})^T$, with $1 \leq i \leq m$.
By projecting the sample a_i on the matrix $\mathbf{V}_k$ we result again in a description with dimension k, that is

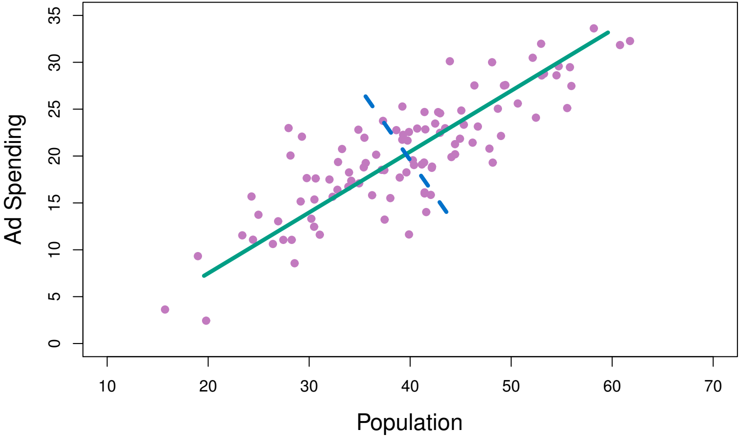
$$\tilde{a}_i \quad = \quad (\tilde{a}_{i,1}, \dots, \tilde{a}_{i,k})^T = (\sigma_1 u_{i,1}, \dots, \sigma_k u_{i,k})^T .$$

The reduced dimensionality data-set with $m \times k$ dimension is given by

$$\tilde{\mathbf{A}}_{m\times k} \quad = \quad \mathbf{U}_k\boldsymbol{\Sigma}_k. \tag{9}$$

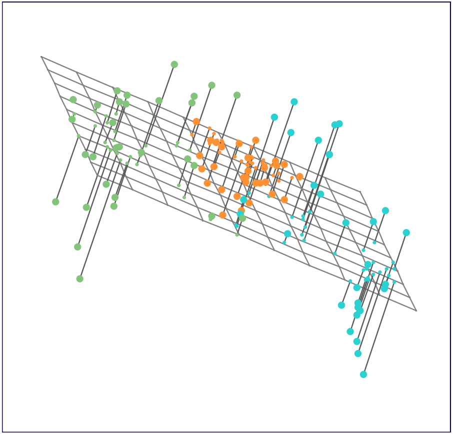
Geometry of 2 components: 2 features

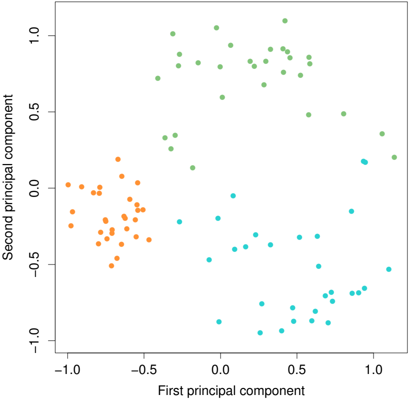
A. Giovanidis 2020



Geometry of 2 components: 3 features

A. Giovanidis 2020





Proportion of variance explained

A. Giovanidis 2020

☞ **Question:** How much of the information in a given data set is lost by projecting the observations onto the first few principal components?

- The total variance present in the (centered) data

$$Total\ \ Var = \sum_{i=1}^{m} Var(a_i) \quad = \quad \sum_{i=1}^{m} \frac{1}{n} \sum_{j=1}^{n} (a_{i,j})^2$$

- The variance explained by the ℓ-th principal component

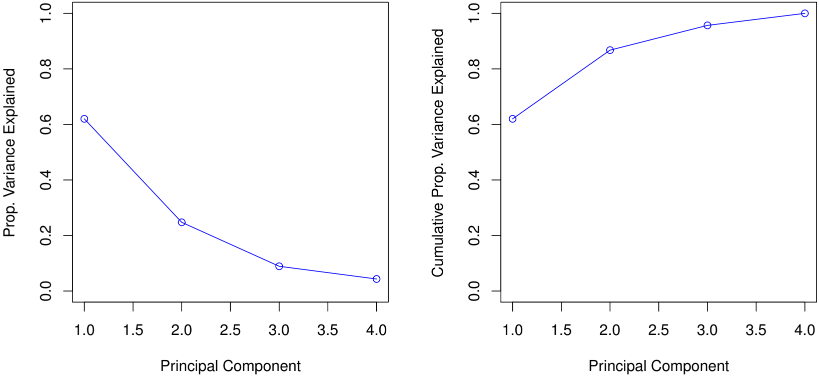
$$Var_\ell \quad = \quad \frac{1}{n} \sum_{j=1}^{n} \left(\sum_{i=1}^{m} u_{i,\ell} a_{i,j} \right)^2$$

⇴ Hence, the cumulative prop. of variance explained (PVE) by k PCs is

$$PVE = \frac{\sum_{\ell=1}^{k} Var_\ell}{Total\ \ Var}$$

PVA curves

A. Giovanidis 2020



How many PCs to keep?

A. Giovanidis 2020

☞ If we use all available r PCs then we get a PVE equal to 1. But then we do not get any dimensionality reduction!

Actually, we want to use the smallest number of principal components required to get a good understanding of the data.

- We can choose the PCs by observing the PVA plot for an elbow.
- In unsupervised learning it depends on the application.
- In supervised learning the best number of PCs can be selected by cross-validation.

A. Giovanidis 2020

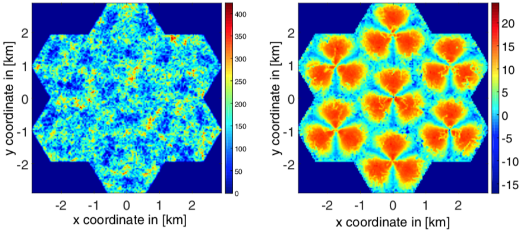
PCA Network Applications:
PCA for 5G

Application no.1: PCA for 5G

A. Giovanidis 2020

☞ We refer to the paper in reference [B.2]

Aim: In 5G cellular networks, to infer the network state and detect anomalous network behaviour.



(a) Number of UEs

(b) Average SINR

A. Giovanidis 2020

How: (2 step)

Step.1 Dimensionality reduction through PCA (visualization)

Step.2 Clustering and classification in low dimensions

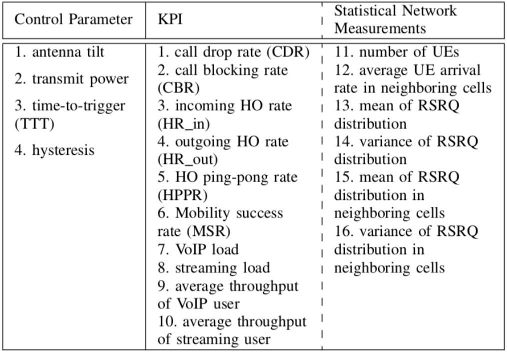
☞ Dimensionality reduction can very much improve clustering and classification results.

Input:16 data features

A. Giovanidis 2020

- Control Parameters
- Key Performance Indicators (KPI)
- Statistical Network Measurements

TABLE I: Selected 4 control parameters and 16 network metrics

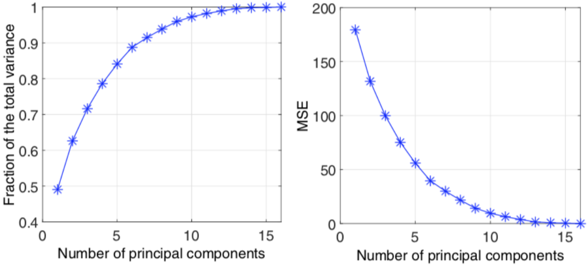


Control Parameter	KPI	Statistical Network Measurements
1. antenna tilt 2. transmit power 3. time-to-trigger (TTT) 4. hysteresis	1. call drop rate (CDR) 2. call blocking rate (CBR) 3. incoming HO rate (HR_in) 4. outgoing HO rate (HR_out) 5. HO ping-pong rate (HPPR) 6. Mobility success rate (MSR) 7. VoIP load 8. streaming load 9. average throughput of VoIP user 10. average throughput of streaming user	11. number of UEs 12. average UE arrival rate in neighboring cells 13. mean of RSRQ distribution 14. variance of RSRQ distribution 15. mean of RSRQ distribution in neighboring cells 16. variance of RSRQ distribution in neighboring cells

How many principal components

A. Giovanidis 2020

How much can the 3 principal components explain from the total variance?

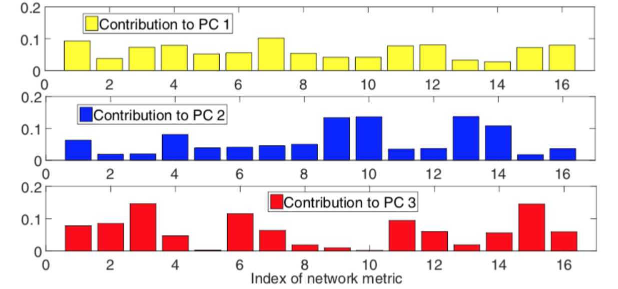


☞ **Answer:** Over 70% !

Explained variance per feature per PC

A. Giovanidis 2020

Interestingly, different components can explain better different features.



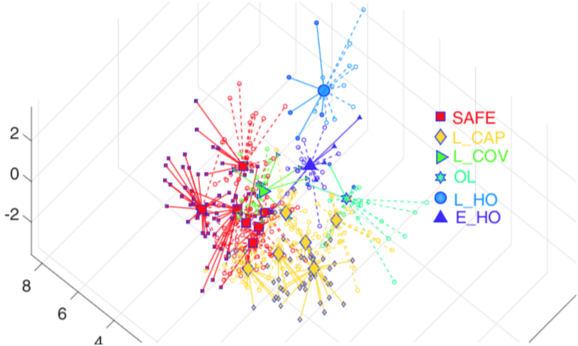
(d) Contribution of 16 network metrics to the top 3 PCs

Some features are more pronounced so the 1st PC explains most, but others are more subtle.

Visualisation in 3D

A. Giovanidis 2020

In the 3D-plane the data points can be visualised



Clustering and Anomaly detection

A. Giovanidis 2020

The network states can be clustered in low dimension.
These classes can be used for anomaly detection:

- One class for SAFE state
- Various classes for various types of anomalous states

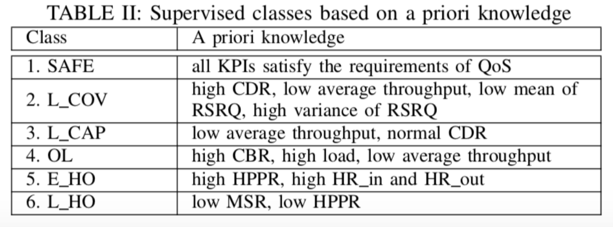


TABLE II: Supervised classes based on a priori knowledge

Class	A priori knowledge
1. SAFE	all KPIs satisfy the requirements of QoS
2. L_COV	high CDR, low average throughput, low mean of RSRQ, high variance of RSRQ
3. L_CAP	low average throughput, normal CDR
4. OL	high CBR, high load, low average throughput
5. E_HO	high HPPR, high HR_in and HR_out
6. L_HO	low MSR, low HPPR

Note: RSRQ is Reference Signal Received Quality. For CDR, CBR, HPPR, HR-in, HR-out, MSR see feature table above.

A. Giovanidis 2020

Clustering method

One can use K-means, with $K = 6$ equal to the number of various states-modes. As an interesting hint, one can use an even larger K and after this, group back together as in hierarchical clustering.

☞ Very often some data are labeled empirically, so we can characterise which cluster refers to which class-mode.

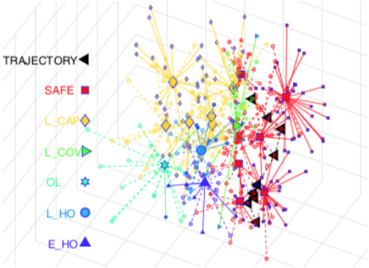
After having clustered, we can use a classifier to predict the class of future network states, and track the status of the network.

This approach is very similar to Exercise 1 in the $\mathrm{TP11 : clustering}$.

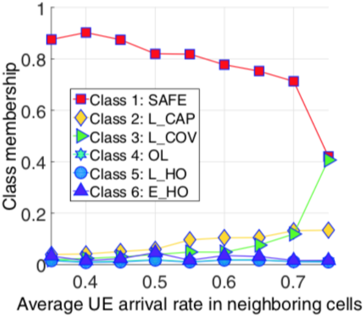
Anomaly Visualisation in 3D

A. Giovanidis 2020

In the 3D-plane we can see the evolution towards an anomalous state



(a) Trajectory of network state.



(b) Class memberships.

A. Giovanidis 2020

PCA Network Applications:

Network-traffic anomaly detection

Network-traffic anomaly

A. Giovanidis 2020

Source material from [B.3]

In network applications it is very important to detect anomalies.

An anomaly is characterised by unusual and significant changes in a network's traffic levels, which can often span multiple links.

Anomalies need to be diagnosed. One must extract and interpret anomalous patterns from large amounts of high-dimensional noisy data.

Anomaly causes may be due to:

- Denial of Service (DoS) attacks,
- Router misconfigurations.

Anomaly detection

A. Giovanidis 2020

☞ Anomalies can create congestion in the network and stress resource utilisation in a router. This is bad from an operational standpoint.

☞ Some anomalies may have dramatic impact on the end user customer, without being dangerous for the networks.

Anomaly detection designates those points in time at which the network is experiencing an anomaly. The algorithm should have a high detection probability and a low false alarm probability.

Network and traffic volume

A. Giovanidis 2020

The studied network has the following structure:

- Nodes are the PoP (Point of Presence)
- Origin-Destination (OD) flows
- The path followed by each OD flow is determined by routing tables.

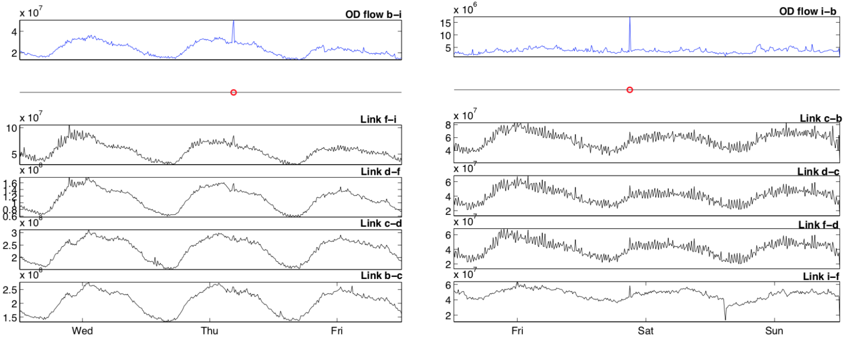
The traffic observed on each link is the superposition of these OD flows.

> Volume anomaly is a sudden (with respect to time-step) positive or negative change in an OD flow's traffic. Such an anomaly originates outside the network, so it propagates from the origin PoP to the destination PoP.

☞ Networks: Sprint (13 PoPs, 49 Links), Abilene (11 PoPs, 41 Links).

Traffic Anomaly by an OD flow

A. Giovanidis 2020



(a) Example 1

(b) Example 2

Figure 1: Examples of anomalies at the OD flow level (top row) that we want to diagnose from link traffic.

Method (I)

A. Giovanidis 2020

☞ The anomaly detection method uses PCA to separate **normal** and **residual** network-wide traffic conditions.

- The measurement matrix **A** is of dimension $T \times n$, where T is the number of successive time intervals of interest, and n the number of links in the network.
- Each column j denotes the sampled time-series of the j-th link, each row i represents an instance of all the links at time i.
- Here $T = 1008$ of 10-min bins in a week long time-series, and n the number of links (41 or 49 depending on the studied network).

PCA is a coordinate transformation method that maps a given set of data points onto new axes, the principal components.

Method (II)

A. Giovanidis 2020

- Adjust **A** so that its columns have zero mean (centered).
- Choose a number of principal components that capture the maximum variance from the original data, but still have a reduced dimension.

Variance captured per PC

A. Giovanidis 2020

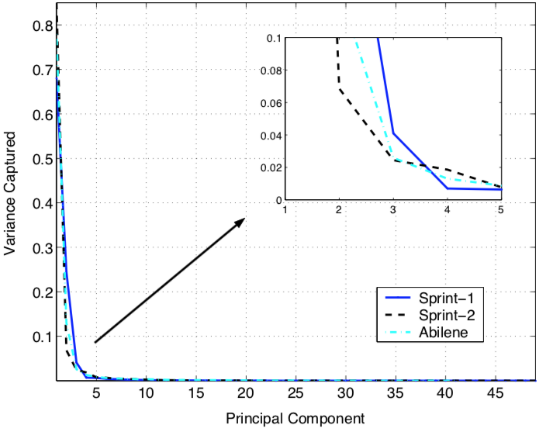


Figure 2: Fraction of total link traffic variance captured by each principal component.

PC definition

A. Giovanidis 2020

We can find n principal components (directions) $\mathbf{v}_j$, $j = 1, \ldots, n$

- The first PC points in the direction of maximum variance

$$\mathbf{v}_1 \quad = \quad \arg\max_{\|\mathbf{v}\|=1} \|\mathbf{A}\mathbf{v}\|$$

- Once the $k-1$ PCs have been determined, the k-th PC corresponds to the maximum variance of the residual

$$\mathbf{v}_k \quad = \quad \arg\max_{\|\mathbf{v}\|=1} \left\| \left(\mathbf{A} - \sum_{i=1}^{k-1} \mathbf{A}\mathbf{v}_i \mathbf{v}_i^T \right) \mathbf{v} \right\|$$

Normal and Anomalous Set

A. Giovanidis 2020

As the above Figure shows, the first 4 PCs capture most of the variance and hence the most significant temporal patterns common to the ensemble of all link traffic time-series.

The current method works by separating the principal axes into two sets, corresponding to

- normal part (y^*) and
- residual variation ($\tilde{y}$) in traffic.

$$y \quad = \quad y^* + \tilde{y}$$

Anomaly detection

A. Giovanidis 2020

Suppose we keep $k < n$ dimensions that sufficiently capture the variance. These components are characterised by the vectors $(\mathbf{v}_1, \ldots, \mathbf{v}_k)$.

We can form the matrix $\mathbf{V}_k = (\mathbf{v}_1, \ldots, \mathbf{v}_k)$ of size $n \times k$. Then

$$\begin{array}{ll} (\textit{normal}) & y^* = \mathbf{V}_k \mathbf{V}_k^T y = \mathbf{C} y \\ (\textit{residue}) & \tilde{y} = \left(\mathbf{I}_{n\times n} - \mathbf{V}_k \mathbf{V}_k^T\right) y = \tilde{\mathbf{C}} y \end{array}$$

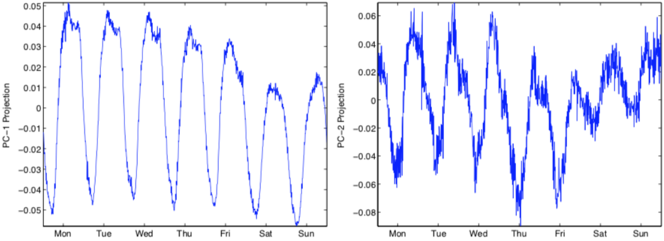
The information about the anomaly is found in the residue. If

$$\|\tilde{y}\|^2 = \left\|\tilde{C} y\right\|^2 > \delta$$

then an anomaly is declared at that specific moment ! (threshold rule)

Example: normal part

A. Giovanidis 2020



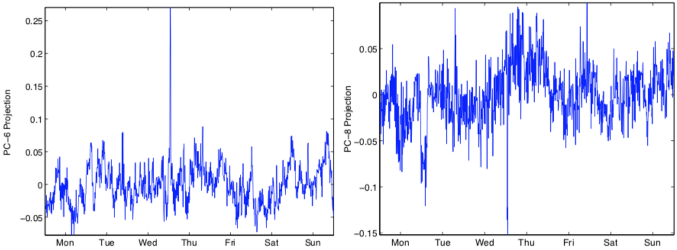
$\mathbf{u}_1$ $\mathbf{u}_2$

(a) Normal Behavior

This part of the data variance is described by the k principal components. We do not expect to find anomalies in this part. ($\mathbf{u}_1 = \sigma_1^{-1}\mathbf{A}\mathbf{v}_1$, $\mathbf{u}_2 = \sigma_2^{-1}\mathbf{A}\mathbf{v}_2$)

A. Giovanidis 2020

Example: residue part



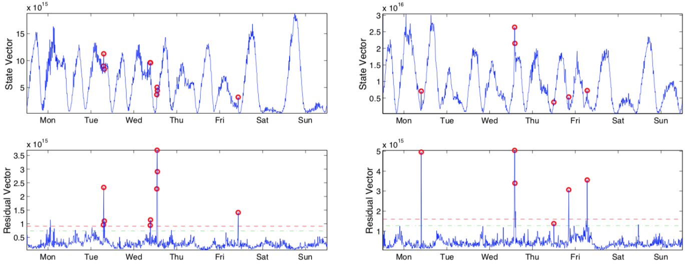
$\mathbf{u}_6$ $\mathbf{u}_8$

(b) Anomalous Behavior

This part of the data variance is the residue, described by the $n - k$ remaining components, or by $\mathbf{A}$ after subtracting the first k components. It is where anomalies live.

Example: anomaly detection

A. Giovanidis 2020



(a) Sprint-1

(b) Sprint-2

Figure 4: Timeseries plots of state vector squared magnitude ($\|\mathrm{y}\|^2$, upper) and residual vector squared magnitude ($\|\tilde{\mathrm{y}}\|^2$, lower) for two weeks of Sprint data.

An example of anomaly detection on time-series, by observing the residue and using a threshold rule. All time instants when the threshold is exceeded are declared anomalous.

A. Giovanidis 2020

END