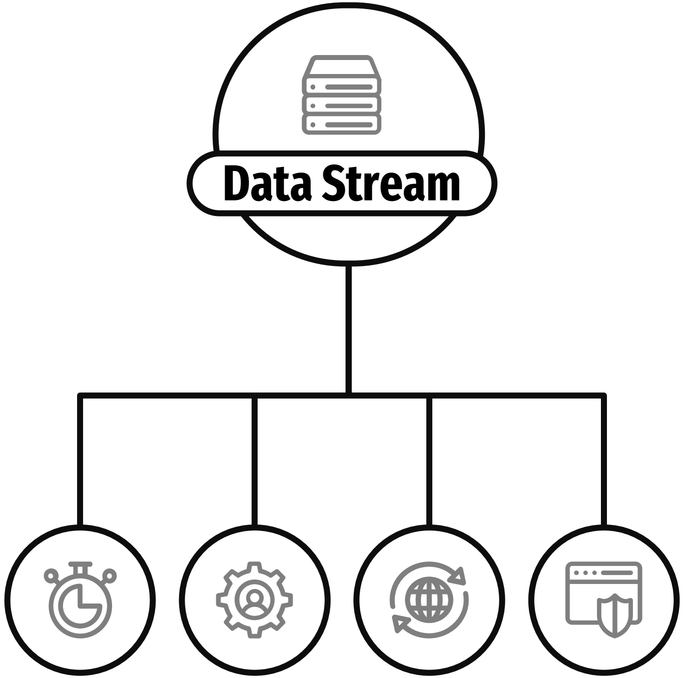
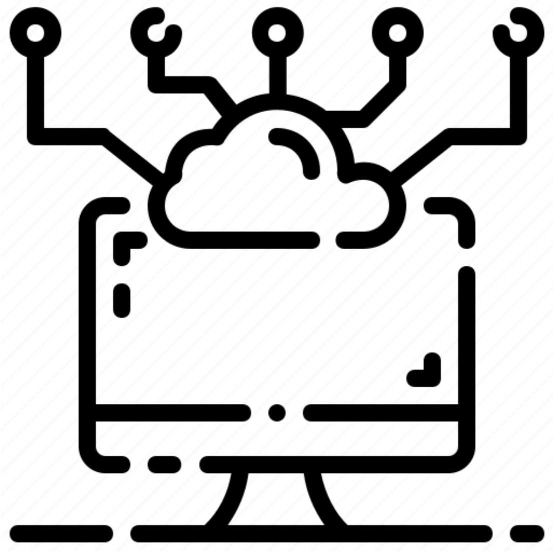
TEDA Algorithm

Pedro Henrique Meira de Andrade
Git: pedrohmeiraa
Supervisor: Prof. Dr. Ivanovitch Medeiros Dantas da Silva
Graduate Program in Electrical and Computer Engineering (PPgEEC/UFRN)



TEDA

- **TEDA (Typicality and Eccentricity Data Analytics) Algorithm** was proposed by Angelov in 2014 to detect anomalies in data streams.
- **Typicality** is the similarity of a sample to the rest of the set (based on distances between samples), while **eccentricity** is its opposite when indicating how different a sample is from the rest of the collected data.
- Datastreams.
- TEDA uses the sum of the geometric distances between the analyzed sample and the other samples in the set. The higher this value, the greater the eccentricity of the sample in relation to the others and, consequently, the lower the value of typicality.



Advantages x Disadvantages:

Advantages: MCUs are cheap, widespread and have low energy comsumption.

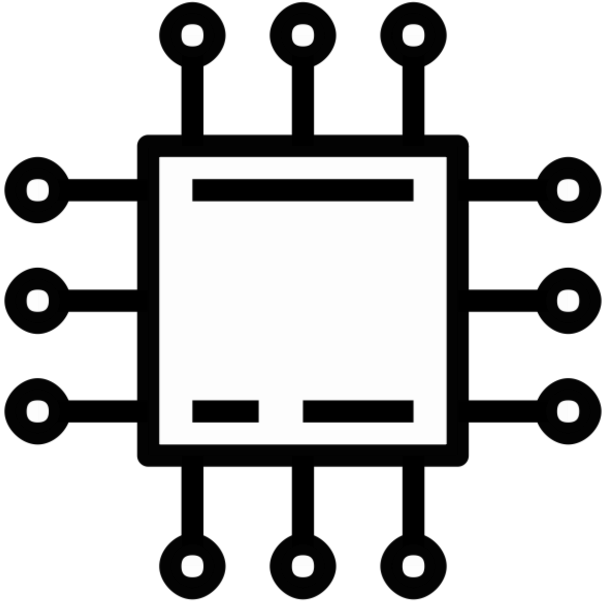
Disadvantages: Low processing power and memory constraints.

Applications: Vehicles, audio processing, industry, home appliances, etc.

TEDA

TEDA has advantages over traditional statistical methods for detecting anomalies:

- **no need for prior knowledge of the data**, therefore, it is widely used for data streams and time series.
- it is **not necessary to know the mathematical model** or the data distribution, being an important advantage for real-world problems.
- It measures the typicality and eccentricity of each sample through geometric distances (Euclidean or cosine or Mahalonobis).
- Computational effort is **low** and **fast**.



TEDA

Therefore, the eccentricity of the data sample ***x***, at the instant of time ***k***:

$$\xi_k(x) = \frac{2\pi_k(x)}{\sum_{i=1}^{k} \pi_k(x_i)}, k > 2 \text{ ,denominador} > 0$$

TEDA

The modeling of the considered data stream can be given by an ordered vector:

$$\boldsymbol{x} = \{x_1, x_2, \dots, x_k, \dots\} \ \therefore \ x_i \in \ \mathbb{R}^n, i \in \mathbb{N}$$

$$d(x_i, x_j) = euclidean\ OR\ cosine\ OR\ mahalanobis$$

The **sum of the distances** from a particular sample (***x***) to each of the other ***k*** elements:

$$\pi_k(x) = \sum_{i=1}^{k} d(x, x_i)\,, k > 2$$

Cat Jump

TEDA

This equation was rewritten so that the **eccentricity** could be calculated recursively:

$$\xi_k(x) = \frac{1}{k} + \frac{(\overset{\text{mean}}{\boxed{\mu_k}} - x_k)^T(\overset{\text{mean}}{\boxed{\mu_k}} - x_k)}{k\underset{\text{variance}}{\boxed{\sigma_k^2}}}$$

TEDA

For that, μ_k and σ_k^2 values for each iteration are calculated recursively using:

$$\mu_k(x) = \frac{k-1}{k}\mu_{k-1} + \frac{1}{k}\mu_k, \;\; \mu_1 = x_1$$

$$\sigma_k^2(x) = \frac{k-1}{k}\sigma^2{}_{k-1} + \frac{1}{k-1}\|x_k - \mu_k\|^2, \sigma_1^2 = 0$$

TEDA

Following the same reasoning, the typicality ($\tau_k(x)$) of the data sample ***x***, at time ***k***, is given by:

$$\tau_k(x) = 1 - \xi_k(x) = \frac{k-1}{k} - \frac{(\mu_k - x_k)^T(\mu_k - x_k)}{k\sigma_k^2}$$

TEDA

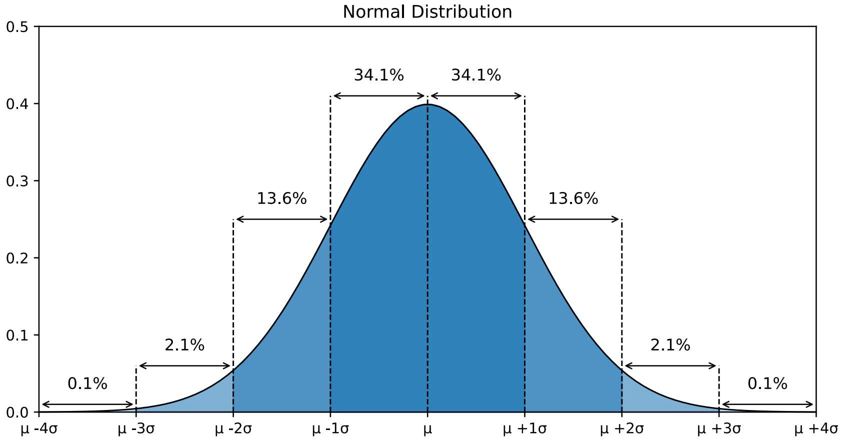
Finally, the normalized eccentricity ($\zeta_k(x)$) and normalized typicality ($t_k(x)$) are given:

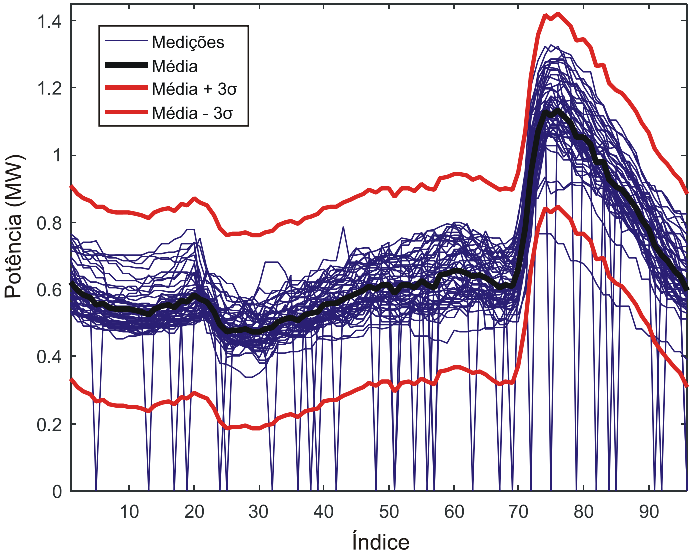
$$\zeta_k(x) = \frac{\xi_k(x)}{2}, \sum_{i=1}^{k} \zeta_i(x) = 1, k \geq 2$$

$$t_k(x) = \frac{\tau_k(x)}{2}, \sum_{i=1}^{k} t_i(x) = 1, k \geq 2$$

TEDA

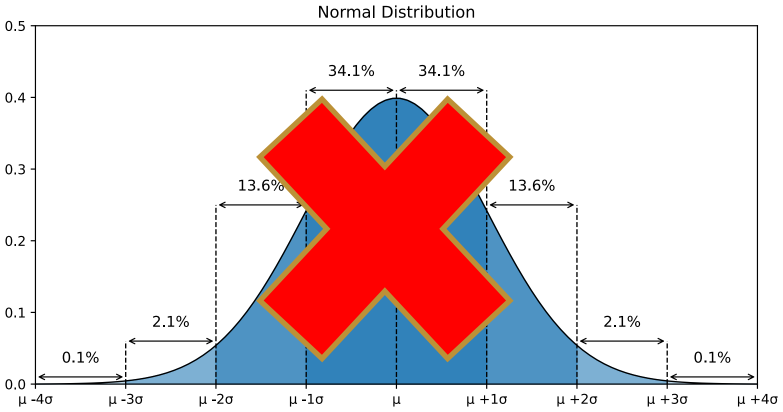
- Now, the next step: Outlier Detection!
- One of the simplest and most well-known methods in the literature is to use “$m\sigma$” as a threshold for classification -> **Gaussian distribution.**





TEDA

- But if my dataset do not have a **Gaussian distribution?**



TEDA

- In a dataset with a **significant number of samples** and for **any data distribution**, it is possible to use **Chebyshev Inequality**.
- This inequality states that the probability of the samples being away from the mean is lower or equal to $\frac{1}{m^2}$.

$$\Pr(|X-\mu| \geq m\sigma) \leq \frac{1}{m^2}$$

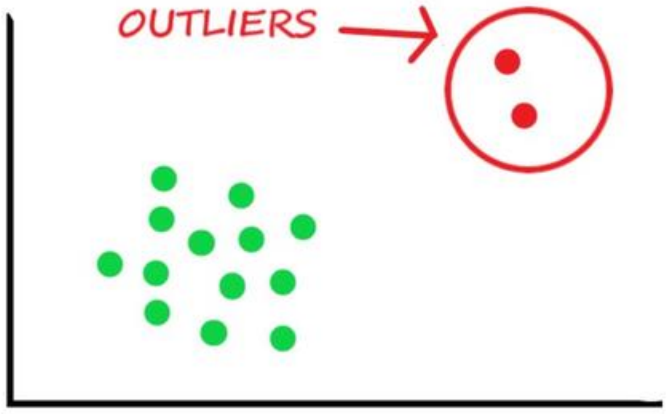
TEDA

- In 1996, Bernieri’s work have adapted it to use normalized eccentricity, as expressed in Equation below:

$$\zeta_k(x) \geq \frac{m^2 + 1}{2k}$$

TEDA

- Therefore, the value of ***m*** represents the ***threshold of sensitivity of the method***. The larger the value of ***m***, the less sensitive the algorithm will be. If the normalized typicality ($\zeta_k(x)$) is greater than the second term, being a true proposition, then x_k will be an outlier.



$$\boxed{\zeta_k(x) \geq \frac{m^2+1}{2k}}$$

“All models are wrong, but some are useful”.

—George E. P. Box (British statistician)

Let’s go to jupyter