ONLINE HACKATHON

Quantum code challenge

Innovative Quantum Algorithms for Smart Cities

22–25 OCTOBER 2024

THE EVENT IS ENDORSED BY

PSC MIMIT - FSC 2014-2020 Programma di supporto tecnologie emergenti nell'ambito del 5G Asse I Progetto "CDL – Casa delle Tecnologie Emergenti di Cagliari" CUP G27F22000040008

OUR TEAM

QDine - Merlin

Riccardo Romanello

Daniele Lizzio Bosco

Ilenia Bariviera

THE TASK

The Problem

Our Solution

Some Results

PROBLEM STATEMENT

- We were assigned a task which involved **four different datasets**: *particulate matter*, *pollution*, *weather* and *unique attendance*. The goal was to create a model that could find and notify anomalies.

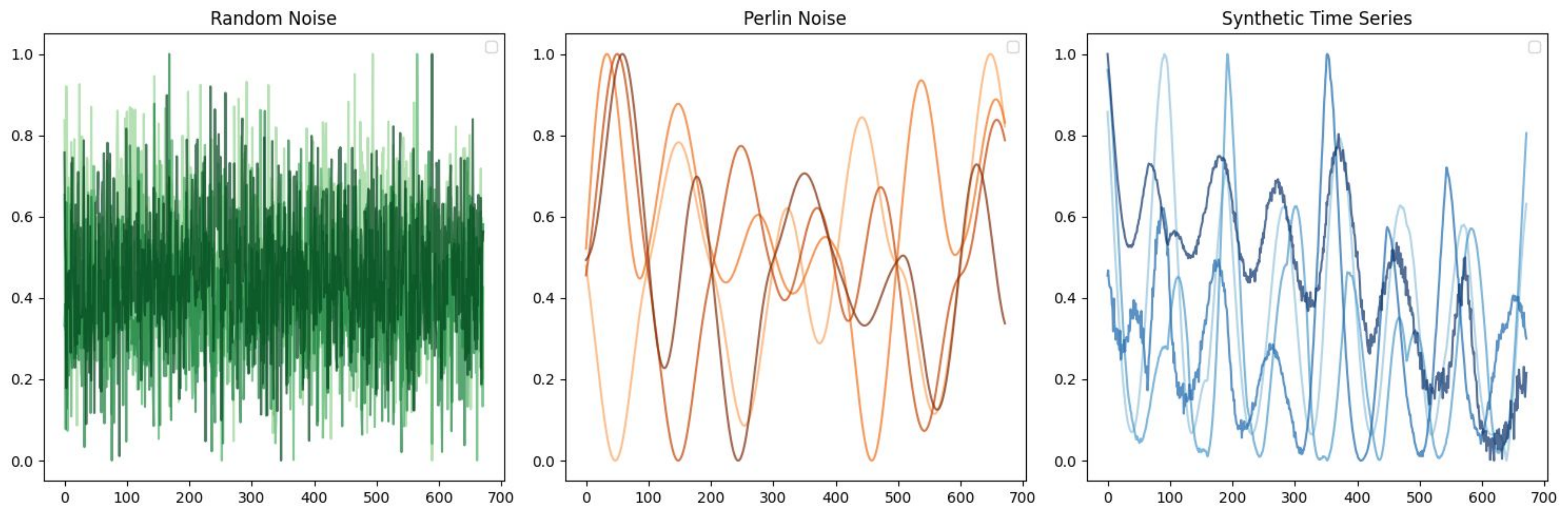
- We focused on **unique attendance**. We wanted to understand the variation in the number of people in different areas of the city and how we could identify anomalies.

- As anomalies, we hypothesized to report possible variations in gatherings of people in cases of **over tourism** or **pandemic crisis** or other purposes.

SYNTHETIC DATA GENERATION

- To evaluate our solution, we generated three different type of **synthetic data.**

- We expect a good model to recognize all the artificial data as ***anomalous***.



PROPOSED SOLUTION: Quantum Temporal Kernels

- Starting from the theoretical background from [1], we implemented a **Quantum Kernel with temporal dependency** able to compute similarity between time series.

- In particular, we used

$$C(x_t) = \mathrm{Emb}(x_t) \cdot U(t)$$

where $\mathrm{Emb}(x_t)$ is a **static mapping** and U introduces the **temporal dependency**.

- Finally, the kernel computes the **similarity** between two series $x^{(t)}$ and $y^{(t)}$ is given by

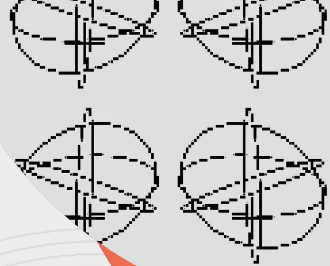
$$K(x^{(t)}, y^{(t)}) = \sum_{t \in T} \langle C(x_t) | C(y_t) \rangle$$

RESULTS

- We compared our results on a test set (*non-anomalous* data) and the synthetic ones.
- We compared our approach with four different classical kernel methods.

- Standard classical kernels:
  - Ill-suited for time series, but very fast
- Dynamic Time Warping:
  - Specialized for time series, but required hours
- Our method:
  - Tailored for time series, and quite fast

For further details see our [GitHub repo](). 🐈



Training and testing datasets of 50 samples

	Accuracy			
	Non-Anomalous	**Noise**	**Perlin**	**Synthetic**
Linear	**82**	2	6	60
Poly	66	72	**100**	**96**
RBF	78	4	10	66

Training and testing datasets of 50 samples

	Accuracy			
	Non-Anomalous	**Noise**	**Perlin**	**Synthetic**
Linear	**82**	2	6	60
Poly	66	72	**100**	**96**
RBF	78	4	10	66
DTW	**94**	**100**	4	72

Training and testing datasets of 50 samples

	Accuracy			
	Non-Anomalous	**Noise**	**Perlin**	**Synthetic**
Linear	**82**	2	6	60
Poly	66	72	**100**	**96**
RBF	78	4	10	66
DTW	**94**	**100**	4	72
Merlin	**80**	76	**80**	**92**

Thanks for your attention!