Time Series Anomaly Detection with Quantum Variational Rewinding

Covalent

agnostiq

🔗 covalent.xyz

Ara Ghukasyan, PhD

Research Software Engineer

ara@agnostiq.ai

Covalent.

🔗 covalent.xyz

AgnostiqHQ / **covalent**

⭐ github.com/AgnostiqHQ/covalent

agnostiq

🔗 agnostiq.ai

PENNYLANE

(An open-source framework for QML and QC.)

```python
import pennylane as qml

observables = [
    qml.PauliZ(0) @ qml.PauliZ(1),
    qml.PauliZ(0) @ qml.PauliX(1),
    qml.PauliX(0) @ qml.PauliZ(1),
    qml.PauliX(0) @ qml.PauliX(1),
]

dev = qml.device("default.qubit", wires=2, shots=8192)

@qml.qnode(dev)
def chsh_circuit(theta):
    # Prepare Bell state.
    qml.Hadamard(wires=0)
    qml.CNOT(wires=[0, 1])

    # Apply Y-rotation by angle `theta`.
    qml.RY(theta, wires=0)

    # Multiple returns to get all 4 expectation values.
    return [qml.expval(obs) for obs in observables]
```

observables as Python objects

a *device* that defines the execution backend

circuit definitions as Python functions

Covalent

- Parallelism
- Remote task execution
- Orchestration and management

```python
def cnn(x, y):
    ...

def train(model):
    tf.train(model)
    return model

@qml.qnode(dev)
def qsvm(x, y):
    ...

def workflow(inputs):
    model1 = cnn(inputs[0], inputs[1])
    model2 = qsvm(inputs[0], inputs[1])
    return train(model1),train(model2)

print(workflow([1,2]))
```

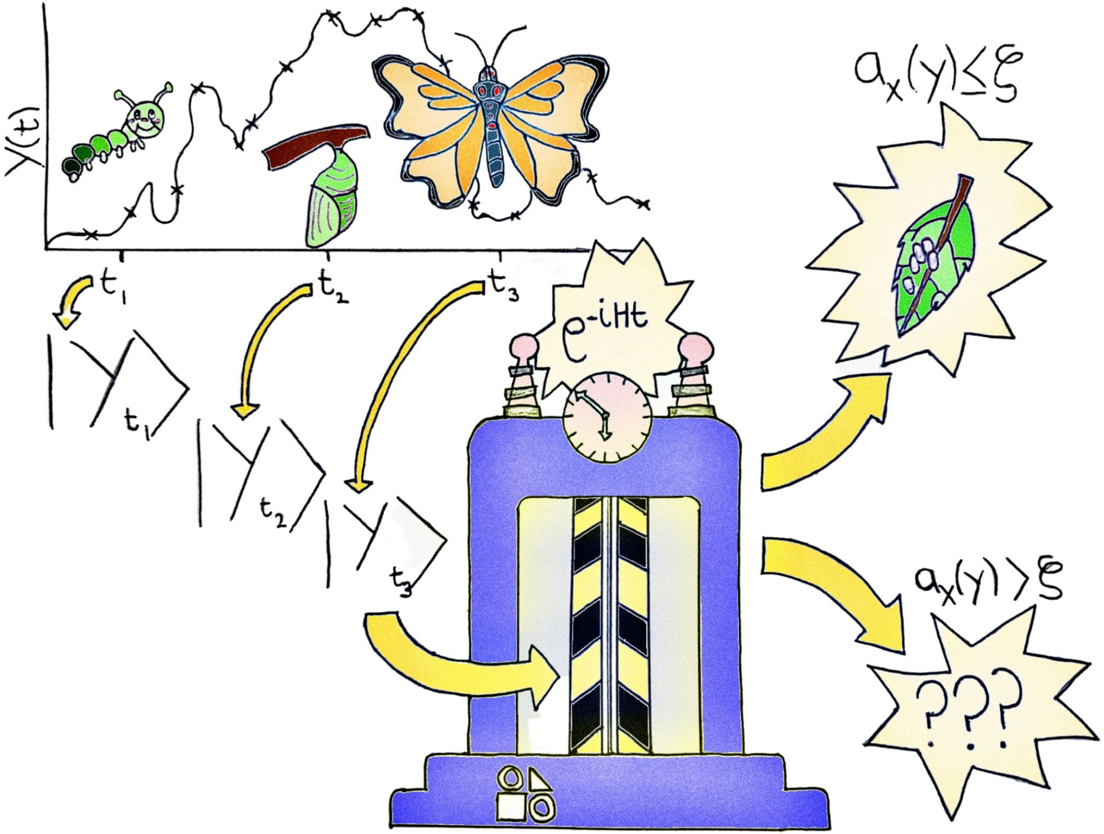
Ship your code anywhere…

```python
@ct.electron(AWSBatchExecutor(vcpu=32))
def cnn(x, y):
    ...

@ct.electron(SlurmExecutor(options))
def train(model):
    tf.train(model)
    return model

@ct.qelectron(QiskitExecutor(backend="ibmq_manila"))
@qml.qnode(dev)
def qsvm(x, y):
    ...

@ct.lattice
def workflow(inputs):
    model1 = cnn(inputs[0], inputs[1])
    model2 = qsvm(inputs[0], inputs[1])
    return train(model1),train(model2)

job_id = ct.dispatch(workflow)([1,2])
print(ct.get_result(job_id))
```

Covalent Concepts Summary

1. Define workflow tasks (functions)
    - **@ct.electron** - classical
    - **@ct.qelectron** - quantum (coming soon)

2. Define the workflow (a special function)
    - **@ct.lattice** - made up of electrons



Let's talk about
QML and Quantum Variational Rewinding

Quantum Machine Learning

What is it ?

A computer program can be considered a learner if it improves its performance on a given set of ***tasks (T)***, as measured by a ***performance metric (P)***, with increasing ***experience (E)***.

experience(E)

- Supervised data
- Unsupervised data

Data

tasks(T)

- Classification
- Regression
- Transcription
- Generation
- Anomaly Detection
- .....

Model

performance metric(P)

- Accuracy
- Error rate
- Regression Loss
- Score points (Games)
- ...

Metric

Quantum Machine Learning

Quantum Variational Algorithms

Ritz Method

(Variational method)

$$E_0 \leq \overset{\substack{\text{Efficiently computed} \\ \text{on quantum computer}}}{\boxed{\frac{\langle \Psi(\theta) | \hat{H} | \Psi(\theta) \rangle}{\langle \Psi(\theta) | \Psi(\theta) \rangle}}} := \langle \hat{O}(\theta) \rangle$$

Observables inform *quantum* performance metrics

Time Series Anomaly Detection

Time series anomaly detection is a statistical problem that aims to identify data points in a time series that deviate significantly from the expected pattern or distribution.

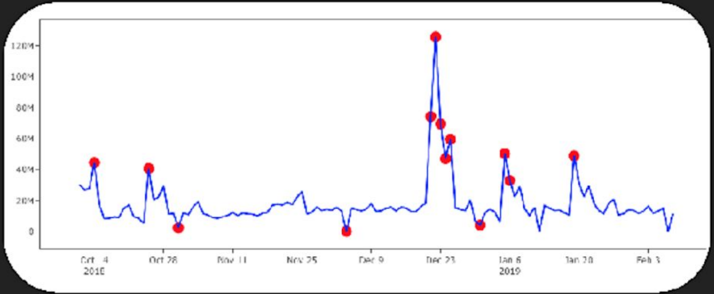
Intution -

- Credit card fraud detection while (traveling ?)
- Single collider events - high energy physics
- Financial markets
- Natural events - weather, earthquakes etc...
- Heart rate
- Brain wave

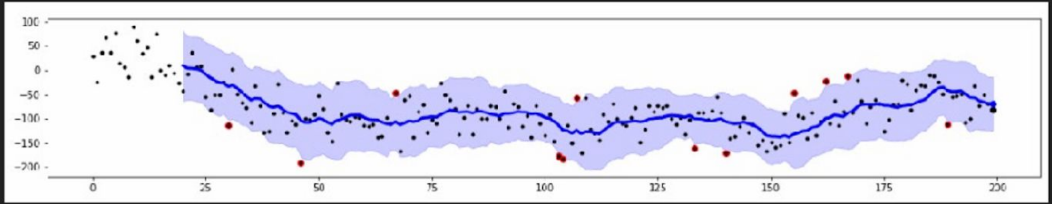
Time Series Anomaly Detection

Examples

Obvious



Not-so obvious



Time Series Anomaly Detection

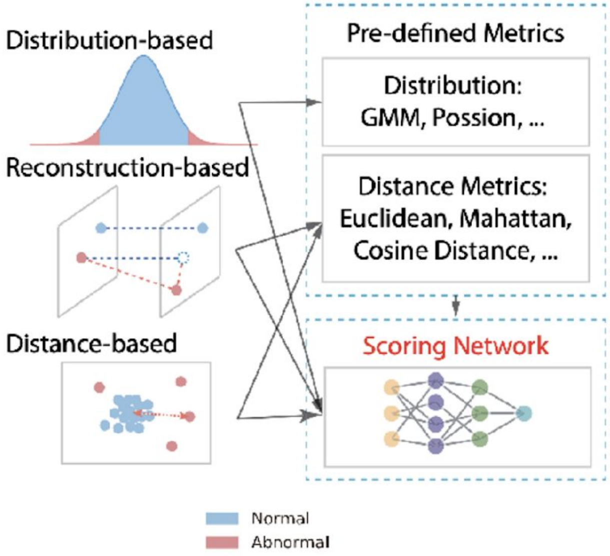
Data

$x_{t_1}, x_{t_2}, ...x_{t_n}$

- Stock data
- Experimental data
- Weather data
- Sequential data

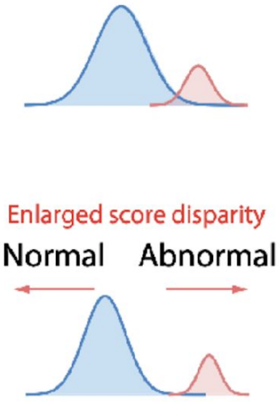
Representation Learning

$\mathcal{R}(x_{t_1}, x_{t_2}, ...x_{t_n})$



Scores (Measure)

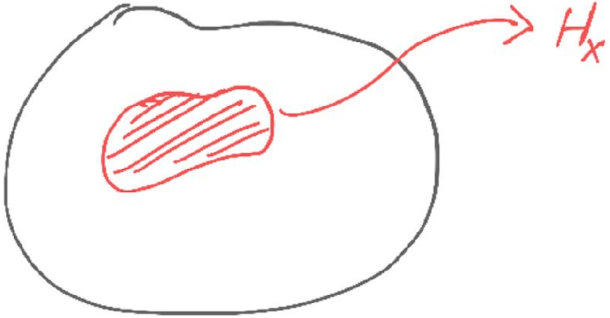
$f\big(\mathcal{R}(x_{t_1}, x_{t_2}, ...x_{t_n})\big) \to \mathbb{R}$



Quantum Variational Rewinding

Problem : Given set of *normal* time series $X : \{x^{(i)}(t)\}$, learn a model $\hat{a}_X$, which scores the *anomalous* nature of unseen time series $\bar{x}(t)$

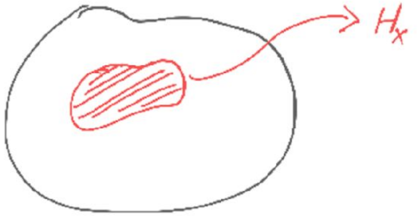
Hypothesis: For a given set X there exists a subspace of Hamiltonians H_X, that can time devolve each quantum embedded normal $|x^{(i)}(t)\rangle$ to an unique state $|o_X\rangle$



Quantum Variational Rewinding

Hypothesis: For a given set X there exists a subspace of Hamiltonians H_X, that can time devolve each quantum embedded normal $|x^{(i)}(t)\rangle$ to an unique state $|o_X\rangle$

Each $|x(t)\rangle$ is a quantum state evolved to a time t as generated by an *unknown embedding Hamiltonian* $\hat{H}_E \in H_X$



$$|x(t)\rangle = e^{-it\hat{H}_E}|o_X\rangle$$

If one is able to learn the set H_X, then for new data $|\bar{x}\rangle$, anomaly score is given by

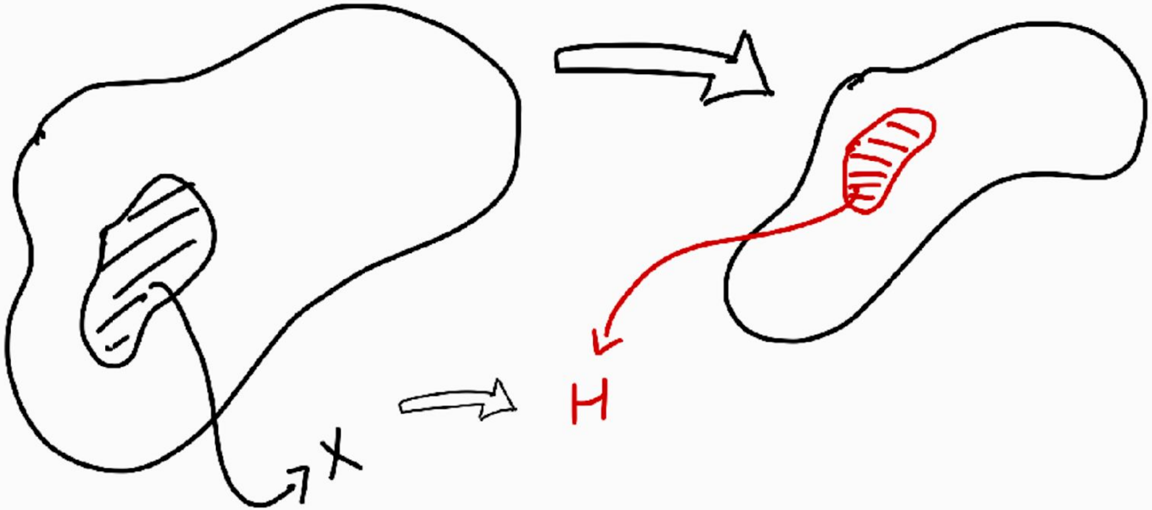
$$\hat{a}_X(\bar{x}(t)) = \mathbb{E}_{\hat{H}\sim H_X}\left[\langle\bar{x}(t)|e^{-it\hat{H}}|o_X\rangle\right]$$

Quantum Variational Rewinding

Intution

Learnt the representation of the time series in the latent space of Hamiltonian

Time series space *Hamiltonian space*



Quantum Variational Rewinding

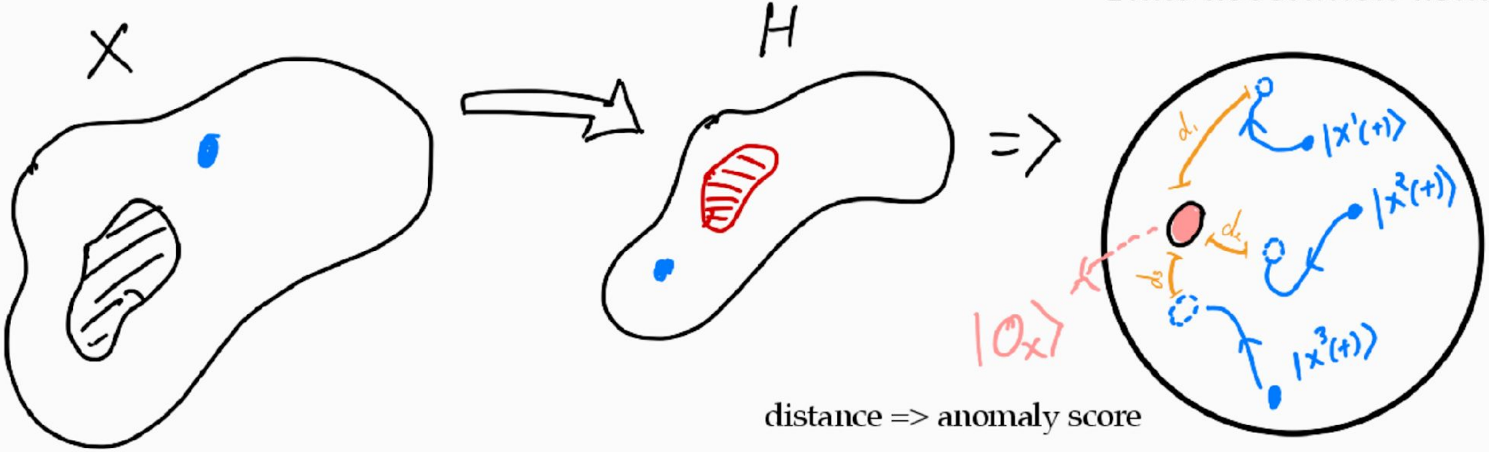
Intution

How does anomalous unseen data behave ?

Time series space

Hamiltonian space

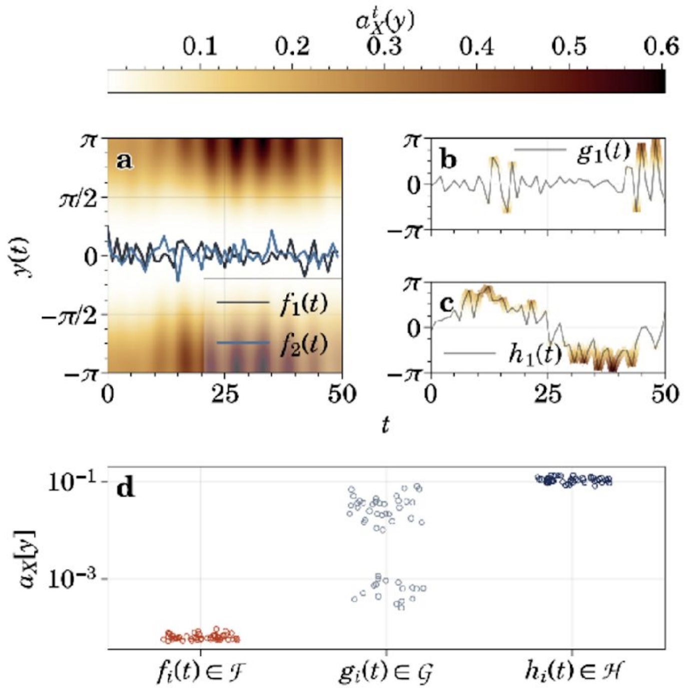
Time devolution using H



distance => anomaly score

Quantum Variational Rewinding

Results



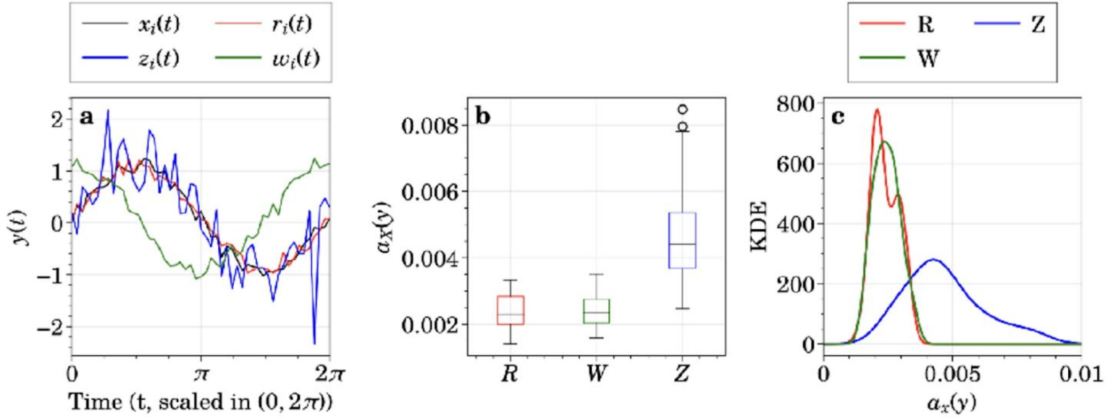
QVR applied to synthetically generated univariate training and testing data. (a) A heat-map of the time-resolved anomaly score $a_X^t(y)$. Two testing signals $f_1(t)$, $f_2(t) \in \mathcal{F}$ (generated using the same mechanism as the training data) are over-plotted and are seen in the region of low $a_X^t(y)$. (b-c) Example signals from the test sets $\mathcal{G}$ and $\mathcal{H}$ respectively, colored according to $a_X^t(y)$. $\mathcal{G}$ contains noisy signals with randomly inserted *anomaly spikes* and $\mathcal{H}$ contains noisy $\sin(t)$ signals. (d) The anomaly scores $a_X[y]$ for each time series in the sets $\mathcal{F}$, $\mathcal{G}$ and $\mathcal{H}$. The y-axis has a logarithmic scale.

🔗 https://arxiv.org/abs/2210.16438

Baker, Jack S., et al. "Quantum Variational Rewinding for Time Series Anomaly Detection." arXiv preprint arXiv:2210.16438 (2022).

Quantum Variational Rewinding

Results



Experiments with noisy univariate sinusoids. (a) an example time series from each of the sets X, R, W and Z. (b) box plots of the anomaly scores obtained from each testing set. (c) The Kernel Density Estimate (KDE) on the distribution of anomaly scores for each of the testing sets.

1. $R = \{r_i : i = 1, ..., 50\}$ where each r_i is defined as the x_i in the training set, but are not identical to the training set since ϵ_i are random.
2. $W = \{w_i : i = 1, ..., 50\}$ where each w_i has the form $w_i = (\cos(0), \cos(\frac{2\pi}{50}), \cos(\frac{4\pi}{50}), ..., \cos(2\pi) + (\zeta_i, ..., \zeta_i)$ where $\zeta_i \sim \mathcal{N}(0.1, 0.1)$.
3. $Z = \{z_i : i = 1, ..., 50\}$ where each z_i has the form $z_i = (\sin(0), \sin(\frac{2\pi}{50}), \sin(\frac{4\pi}{50}), ..., \sin(2\pi) + (\xi_i, ..., \xi_i)$ where $\xi_i \sim \mathcal{N}(0.1, 0.5)$.

🔗 https://arxiv.org/abs/2210.16438

Baker, Jack S., et al. "Quantum Variational Rewinding for Time Series Anomaly Detection." arXiv preprint arXiv:2210.16438 (2022).

Notebook: Quantum Variational Rewinding

✅ https://github.com/AgnostiqHQ/tutorials_covalent_qsite_2023

Quantum Variational Rewinding

Authors: Jack Stephen Baker, Santosh Kumar Radha

Revised by: Ara G.

✅ This notebook is based on our QVR demo for Pennylane: Quantum detection of time series anomalies

---

Background

A general time series $\boldsymbol{y}$ can be described as a sequence of p-many observations of a process/system arranged in chronolo

$$\boldsymbol{y} := (\boldsymbol{y}_t : t \in T), \quad T :=$$

❗ NOTE: We drop the bold-face for $\boldsymbol{y}$ beyond this point.

Goal

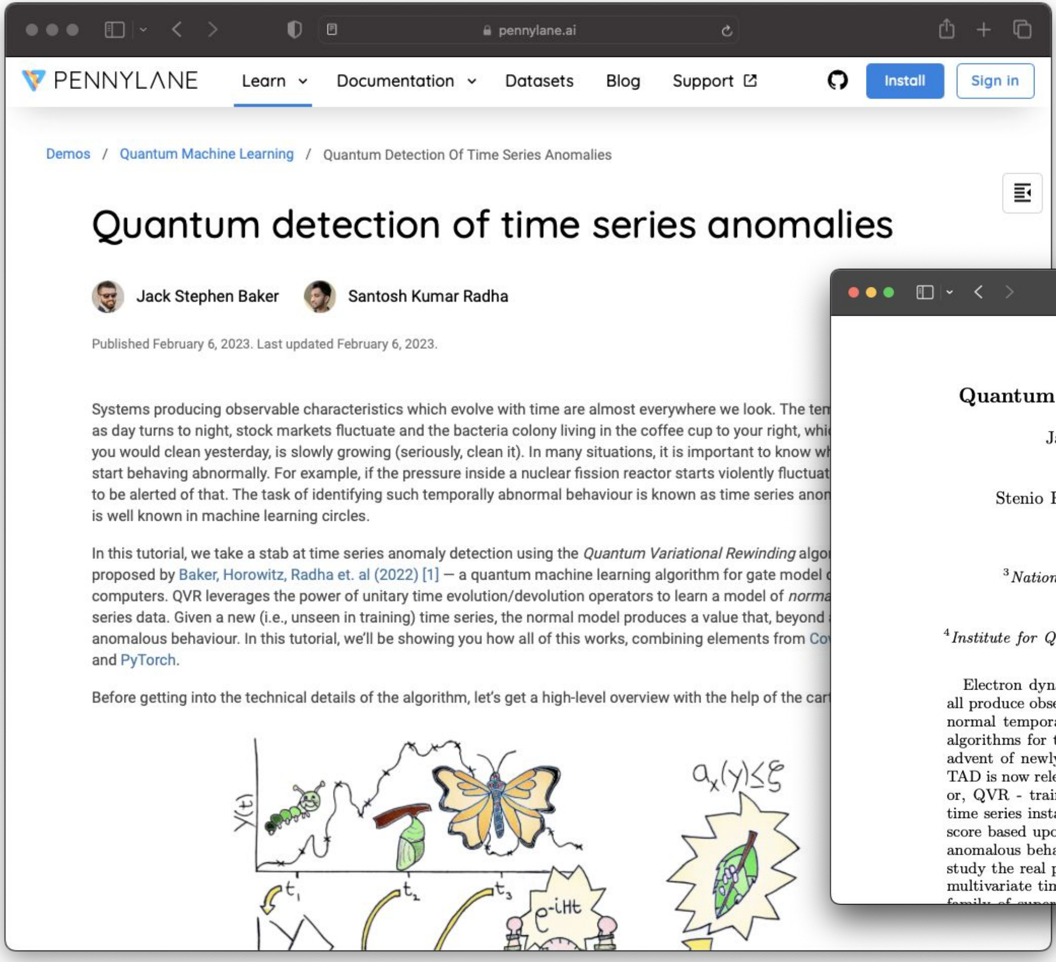
The goal of QVR and many other (classical) machine learning algorithms for time series anomaly detection is to determine a ... nces $x \in X$ (x is defined analogously to y in the above), from which the anomaly score function was learnt. When passed a general time s ... all $x \in X$. Then, for an unseen time series y and a threshold $\zeta \in \mathbb{R}$, the series is said to be anomalous should $a_X(y) > \zeta$, and norma

Embedding

The first step for doing all of this *quantumly* is to generate a sequence $\mathcal{S} := (|x_t\rangle : t \in T)$ of n-qubit quantum states corresp ... $|x_t\rangle$ is a quantum state evolved to a time t, as generated by an *unknown embedding Hamiltonian* H_E. That is, each element of $\mathcal{S}$ is defined by $|x_t\rangle =$ ... plementing a quantum feature map (see the Pennylane embedding templates for efficient quantum circuits for doing so). Next, we operate on each $|x_t\rangle$ with a parameterized $e^{-iH(\alpha,\gamma)t}$ operator to prepare the states

$$|x_t, \alpha, \gamma\rangle := e^{-iH(\alpha,\gamma)t}|x_t\rangle,$$

where we write $e^{-iH(\alpha,\gamma)t}$ as an eigendecomposition


pennylane.ai

PENNYLANE Learn Documentation Datasets Blog Support Install Sign in

Demos / Quantum Machine Learning / Quantum Detection Of Time Series Anomalies

Quantum detection of time series anomalies

Jack Stephen Baker Santosh Kumar Radha

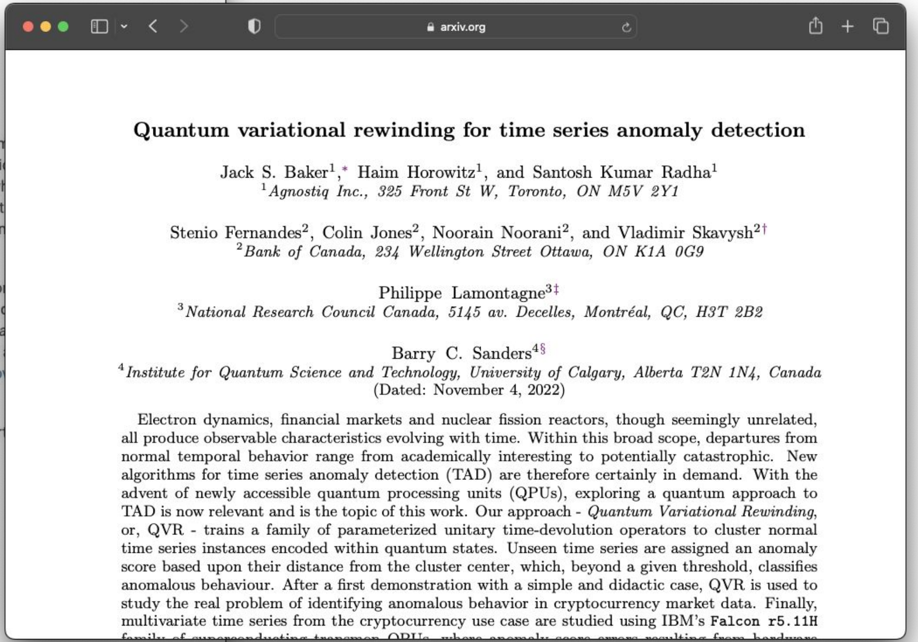
Published February 6, 2023. Last updated February 6, 2023.

Systems producing observable characteristics which evolve with time are almost everywhere we look. The tem
as day turns to night, stock markets fluctuate and the bacteria colony living in the coffee cup to your right, whi
you would clean yesterday, is slowly growing (seriously, clean it). In many situations, it is important to know wh
start behaving abnormally. For example, if the pressure inside a nuclear fission reactor starts violently fluctuat
to be alerted of that. The task of identifying such temporally abnormal behaviour is known as time series anon
is well known in machine learning circles.

In this tutorial, we take a stab at time series anomaly detection using the *Quantum Variational Rewinding* algo
proposed by Baker, Horowitz, Radha et. al (2022) [1] — a quantum machine learning algorithm for gate model
computers. QVR leverages the power of unitary time evolution/devolution operators to learn a model of *norma*
series data. Given a new (i.e., unseen in training) time series, the normal model produces a value that, beyond
anomalous behaviour. In this tutorial, we'll be showing you how all of this works, combining elements from Co
and PyTorch.

Before getting into the technical details of the algorithm, let's get a high-level overview with the help of the car

🔗 https://arxiv.org/abs/2210.16438


arxiv.org

Quantum variational rewinding for time series anomaly detection

Jack S. Baker[1,*] Haim Horowitz[1], and Santosh Kumar Radha[1]
[1] *Agnostiq Inc., 325 Front St W, Toronto, ON M5V 2Y1*

Stenio Fernandes[2], Colin Jones[2], Noorain Noorani[2], and Vladimir Skavysh[2†]
[2] *Bank of Canada, 234 Wellington Street Ottawa, ON K1A 0G9*

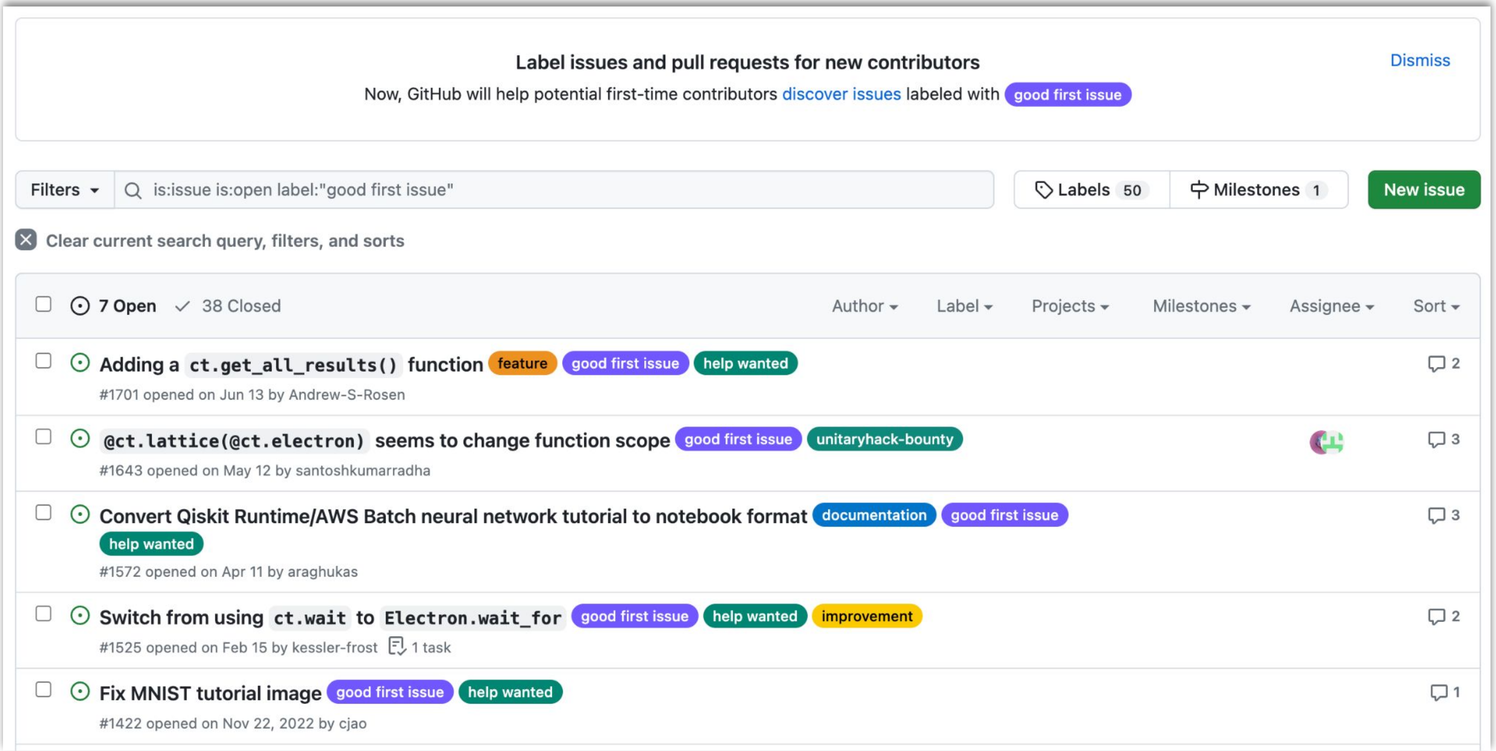
Philippe Lamontagne[3‡]
[3] *National Research Council Canada, 5145 av. Decelles, Montréal, QC, H3T 2B2*

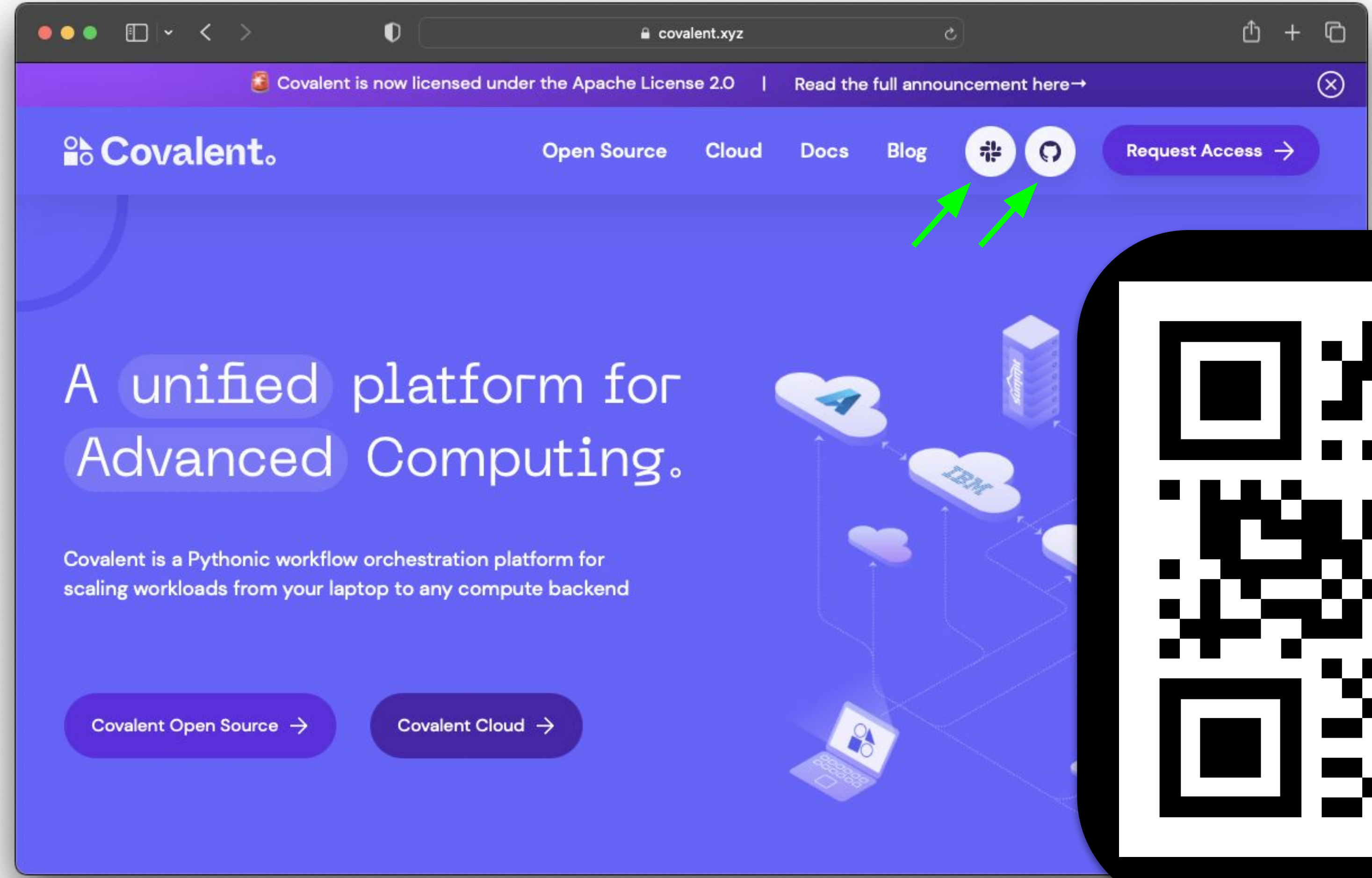
Barry C. Sanders[4§]
[4] *Institute for Quantum Science and Technology, University of Calgary, Alberta T2N 1N4, Canada*
(Dated: November 4, 2022)

Electron dynamics, financial markets and nuclear fission reactors, though seemingly unrelated, all produce observable characteristics evolving with time. Within this broad scope, departures from normal temporal behavior range from academically interesting to potentially catastrophic. New algorithms for time series anomaly detection (TAD) are therefore certainly in demand. With the advent of newly accessible quantum processing units (QPUs), exploring a quantum approach to TAD is now relevant and is the topic of this work. Our approach - *Quantum Variational Rewinding*, or, QVR - trains a family of parameterized unitary time-devolution operators to cluster normal time series instances encoded within quantum states. Unseen time series are assigned an anomaly score based upon their distance from the cluster center, which, beyond a given threshold, classifies anomalous behaviour. After a first demonstration with a simple and didactic case, QVR is used to study the real problem of identifying anomalous behavior in cryptocurrency market data. Finally, multivariate time series from the cryptocurrency use case are studied using IBM's `Falcon r5.11H`

All contributors are welcome!

🔗 **github.com/AgnostiqHQ/covalent/issues**





covalent.xyz

covalent.xyz

agnostiqHQ/covalent 🌟

@covalentxyz

Thanks for listening 😄