SRM Institute of Science and Technology

Project Abstract · LLJ 1 Team Task Review · Quantum Machine Learning

Q-DRIFT: Drift-Aware Quantum Kernel Adaptation for Spacecraft Telemetry Anomaly Detection

S. No.	Name	Register Number
1	Venkatanathan O R V	RA2411026011163
2	Mayukh Sarkar	RA2411026011218
3	Aman Salarpuria	RA2411003012435

Course and review	LLJ 1 — Team Task Review
Faculty	Dr. M. L. Sworna Kokila
Domain	Quantum Machine Learning (QML), time-series anomaly detection, concept drift
Core model	Quantum Support Vector Machine (one-class) with an adaptive quantum feature map
Dataset	NASA SMAP satellite and Mars Science Laboratory (Curiosity rover) telemetry

Abstract

Spacecraft send a continuous stream of sensor readings, called telemetry, back to Earth. Mission teams rely on anomaly-detection models to flag faults early, but these models are trained on past behaviour. When a spacecraft changes operating mode, or a sensor slowly degrades, the data distribution shifts. This is known as concept drift. A model trained on the old distribution then raises false alarms and can miss real faults.

Quantum kernel methods are a promising family of quantum machine learning models for anomaly detection. A quantum feature map encodes each reading into a quantum state, and the kernel measures similarity between states. Their performance depends strongly on this encoding circuit, yet existing work chooses the encoding once, offline, and never revisits it.

We propose **Q-DRIFT**, a drift-aware quantum kernel framework for spacecraft telemetry. A quantum-kernel one-class Support Vector Machine produces anomaly scores from windowed telemetry, reduced by PCA to 4–8 values and encoded on 4–8 simulated qubits. An online drift detector (ADWIN) monitors the stream of scores. When drift is detected, a feature-map controller evaluates a small library of candidate encodings (angle, ZZ and IQP feature maps at several bandwidths and depths) on a rolling window using kernel-target alignment. It then switches to the best candidate after a penalty for quantum circuit cost, without retraining the pipeline from scratch. Encodings that worked for a mode are cached and reused when that mode recurs.

We will evaluate Q-DRIFT on the NASA SMAP and Mars Science Laboratory telemetry benchmark (82 channels, 105 expert-labelled anomaly sequences) under natural and injected abrupt, gradual and recurring drift. It will be compared with a fixed quantum SVM, a statically optimised quantum

SVM, a classical RBF one-class SVM and an online classical model, using event-level F1, PR-AUC, false alarms, detection delay, recovery time after drift and circuit evaluations per sample. We expect Q-DRIFT to restore detection quality within a few windows after drift while spending far fewer quantum circuit evaluations than retraining from scratch. To our knowledge, this is the first drift-adaptive quantum kernel method evaluated on real spacecraft telemetry.

Keywords: quantum machine learning, quantum kernel, quantum support vector machine, concept drift, anomaly detection, spacecraft telemetry, continual learning, NISQ.

1. Introduction and Motivation

Every satellite and rover reports its health through telemetry: temperatures, battery voltages, currents, attitude and the commands it executes. Engineers at mission control watch these channels around the clock, because an undetected heater failure or power fault can end a mission. With thousands of satellites now in orbit, manual monitoring does not scale, so automated anomaly detection has become a core part of spacecraft operations.

NASA's Jet Propulsion Laboratory released a public benchmark from two real missions: the SMAP Earth-observation satellite and the Curiosity rover of the Mars Science Laboratory (MSL). Its anomalies were labelled by experts from real incident reports [1]. Models built on this data, such as LSTM forecasters, assume that "normal" behaviour stays the same after training.

In practice it does not. A spacecraft switches between operating modes, moves between sunlight and eclipse every orbit, and its sensors age. Each of these changes the statistical distribution of the data, a problem called concept drift [2]. Classical stream-learning research offers drift detectors such as ADWIN [3], but they only signal that something changed; they do not repair the model.

Quantum machine learning adds a new option. A quantum kernel embeds data into a very large quantum state space through a small parameterised circuit, the feature map, and measures similarity there [4]. Studies show that the choice of feature map and its bandwidth largely decides whether a quantum kernel works at all [5], [6]. This makes quantum kernels especially sensitive to drift, and it also suggests a remedy: when the data changes, change the encoding.

2. Problem Statement

Given a multivariate telemetry stream x_1, x_2, … whose distribution changes at unknown times, detect anomalous events with few false alarms, using a quantum kernel model whose encoding was chosen for an earlier distribution.

A fixed quantum kernel trained on one operating mode loses accuracy after drift: normal readings from the new mode fall outside the learned boundary and trigger false alarms, while real faults are hidden among them. Retraining the whole quantum model after every change is slow and costs many quantum circuit evaluations. The problem is therefore to **detect drift online and restore detection quality by re-selecting the quantum encoding, at a bounded quantum cost**, without full retraining.

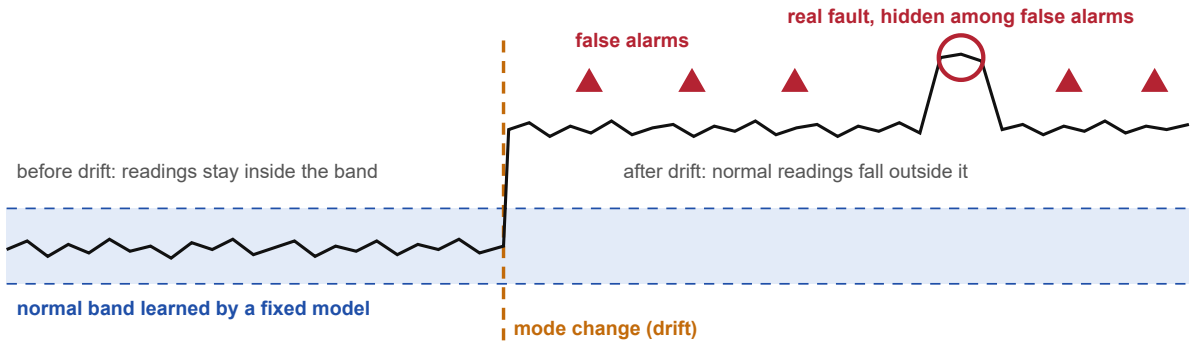


Fig. 1. Concept drift: after a mode change, a fixed model's notion of "normal" no longer fits the data.

Drift appears in three forms in spacecraft telemetry: **abrupt** (a new operating mode switches on), **gradual** (a sensor slowly ages) and **recurring** (modes that return, such as sunlight and eclipse on every orbit). Q-DRIFT is designed and evaluated for all three.

3. Literature Survey

Prior work covers anomaly detection on this data, drift detection and quantum kernels separately; none combines them (Table 1).

Ref.	Work (year)	Method	Data used	Limitation for our problem
[1]	Hundman et al., NASA JPL (2018)	LSTM forecasting + nonparametric dynamic thresholds	SMAP, MSL	Classical; adapts the threshold, not the model's representation
[12]	Schölkopf et al. (2001)	One-class SVM: learns the boundary of normal data	General	Fixed kernel; no notion of drift
[3]	Bifet & Gavaldà (2007)	ADWIN adaptive-window drift detector with false-alarm bounds	Data streams	Detects change only; does not repair the model
[2]	Gama et al. (2014)	Survey of concept-drift adaptation strategies	Survey	Classical models only
[4]	Havlíček et al. (2019)	Quantum-enhanced feature spaces; quantum kernel SVM	Synthetic	Encoding fixed; static data
[6]	Hubregtsen et al. (2022)	Trainable quantum embedding kernels via kernel-target alignment	Classification benchmarks	Encoding tuned once, offline
[5]	Shaydulin & Wild (2022)	Kernel bandwidth as a key quantum-kernel hyperparameter	Classical datasets	Bandwidth chosen once; no streams
[7]	Thanasilp et al. (2024)	Exponential concentration of quantum kernel values	Theory	Large-qubit kernels flatten; motivates small circuits

Ref.	Work (year)	Method	Data used	Limitation for our problem
[8]	Kölle et al. (2024)	Efficient quantum one-class SVM (randomized measurements, subsampling)	Anomaly benchmarks	No drift; no spacecraft data
[9]	Quantum autoencoders for time series (2024)	Reconstruction error of a quantum autoencoder	Time-series benchmarks	No drift adaptation; no spacecraft data
[10]	Wu & Keogh (2020–22)	Critique of anomaly benchmarks incl. SMAP/MSL	Benchmarks	Shows evaluation pitfalls we must avoid
[11]	ESA Anomaly Detection Benchmark (2024)	New satellite telemetry benchmark	ESA missions	Optional second dataset for validation

Table 1. Summary of related work.

4. Research Gap

To our knowledge, no published work re-selects a quantum kernel's encoding online when drift is detected, and no quantum model has been evaluated on the NASA SMAP/MSL benchmark.

- **Static encodings.** Quantum kernel studies tune the feature map or bandwidth once, before deployment [5], [6]. Nothing adapts it after the data changes.
- **No quantum work on spacecraft telemetry.** Quantum anomaly detectors are tested on network-traffic or synthetic data [8], [9]; the real spacecraft benchmark [1] has only classical results.
- **Cost is ignored.** Drift adaptation in classical ML assumes retraining is cheap. On quantum hardware every kernel entry costs circuit runs, so the cost of adapting must be measured and limited.
- **Detectors do not repair.** Drift detectors such as ADWIN [3] flag a change but leave the model as it was.

5. Objectives

1. Build a streaming pipeline for the NASA SMAP and MSL telemetry, processed strictly in time order.
2. Implement a quantum-kernel one-class SVM on 4–8 simulated qubits as the anomaly detector.
3. Design a library of candidate quantum feature maps (angle, ZZ, IQP) with bandwidth and depth settings.
4. Integrate an online drift detector (ADWIN) on the anomaly-score stream.
5. Develop a cost-aware controller that re-selects the encoding after drift using kernel-target alignment, and caches encodings for recurring modes.
6. Evaluate against fixed, statically optimised and classical baselines under abrupt, gradual and recurring drift, reporting accuracy, false alarms, recovery time and quantum cost.

6. Proposed Methodology

Scores flow left to right through the detector; when ADWIN flags drift, the controller tests the library and swaps a new encoding into the quantum model. Fig. 2 shows the seven components:

1. **Telemetry stream.** SMAP and MSL channels are read one window at a time, strictly in time order.
2. **Window + PCA.** Each channel is z-normalised on its training segment; a sliding window is compressed by PCA to 4–8 values, one per qubit.
3. **Quantum one-class SVM.** The current feature map U_c encodes each window; the quantum kernel k_c feeds a one-class SVM that learns the boundary of normal data and outputs an anomaly score s(x).
4. **Alarm.** A score above the threshold alerts the mission team.
5. **Drift detector.** ADWIN watches the stream of scores and flags drift when the mean of recent scores shifts significantly.
6. **Controller.** On drift, it scores every library candidate by kernel-target alignment on recent data, subtracts a cost penalty, and swaps the best encoding c* into the quantum model.
7. **Feature-map library and mode cache.** The library holds angle, ZZ and IQP encodings at three bandwidths and two depths; the cache stores which encoding worked for each mode so recurring modes switch instantly.

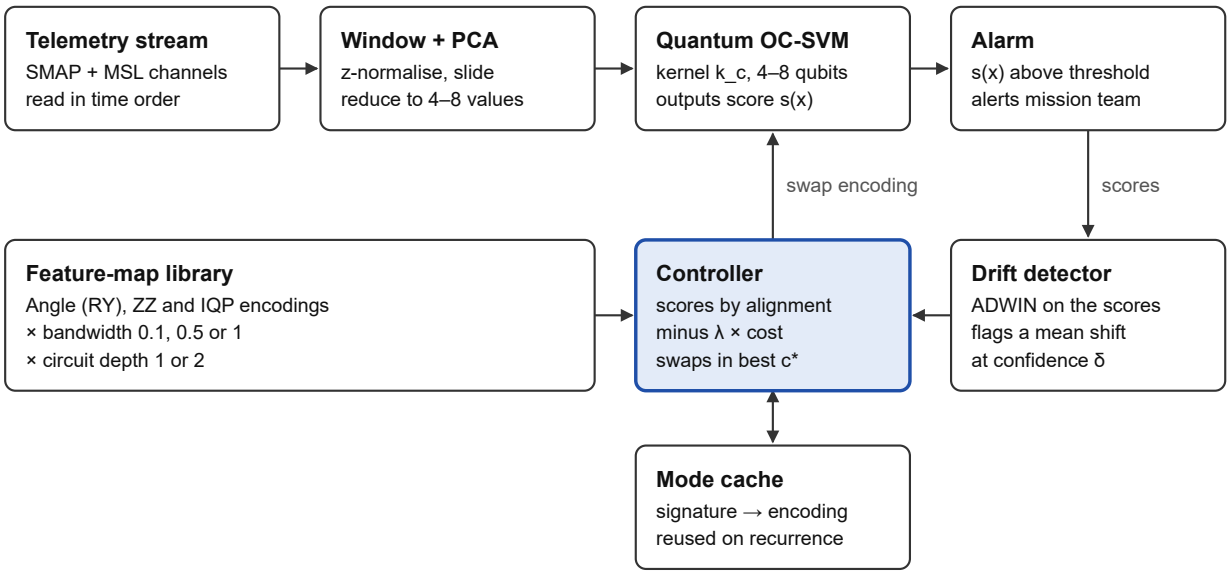


Fig. 2. Q-DRIFT system architecture (7 components).

7. Q-DRIFT Algorithm and Key Equations

Quantum kernel. Each window x is scaled by a bandwidth β and encoded by a feature-map circuit U_c from the library. The kernel is the overlap of the two quantum states [4]:

$$k_c(x, x') = |\langle 0| U_c^{\dagger}(\beta x')\, U_c(\beta x) |0\rangle|^2 \quad (1)$$

Anomaly score. The one-class SVM [12] learns support vectors x_i with weights α_i and offset ρ on normal data; a large score means anomalous:

$$s(x) = \rho - \Sigma_i\, \alpha_i\, k_c(x_i, x) \quad (2)$$

Drift test (ADWIN). The window of recent scores W is split into two parts W_0, W_1 of sizes n_0, n_1. Drift is flagged when their means differ by more than ε_{cut} [3]:

$$| \hat{\mu}_{W_0} - \hat{\mu}_{W_1} | \geq \varepsilon_{cut}, \quad \varepsilon_{cut} = \sqrt{\frac{1}{2m} \ln \frac{4|W|}{\delta}}, \quad m = \frac{1}{1/n_0 + 1/n_1} \tag{3}$$

Encoding selection. After drift, each candidate c is scored by kernel-target alignment [13] on a recent window labelled y (recent normal windows +1, synthetically perturbed copies −1), minus a cost penalty for its circuit evaluations:

$$A(K_c, y) = \frac{\langle K_c, yy^T \rangle_F}{\|K_c\|_F \, \|yy^T\|_F} \tag{4}$$

$$c^* = \arg\max_c \left[A(K_c, y) - \lambda \cdot \frac{\mathrm{Cost}(c)}{\mathrm{Cost}_{max}} \right] \tag{5}$$

Pseudo-code.

```
Algorithm Q-DRIFT
Input : telemetry stream X, library C = {angle, ZZ, IQP} x bandwidths x depths,
        drift confidence delta, cost weight lambda, cache M = {}
1  c <- best encoding on the initial normal segment (alignment - cost)
2  train one-class QSVM with kernel k_c on the initial segment
3  for each new window x_t:
4      s_t <- anomaly score s(x_t);  raise alarm if s_t > threshold
5      feed s_t to ADWIN
6      if ADWIN detects drift:
7          sig <- (mean, variance) signature of the recent window
8          if a mode in M matches sig:           # recurring drift
9              c <- M[sig]
10         else:
11             build labelled window y from recent data
12             for c' in C: score(c') <- A(K_c', y) - lambda*Cost(c')/Cost_max
13             c <- argmax score;  M[sig] <- c
14         refit one-class QSVM on the recent window with kernel k_c
15         log circuit evaluations used by this switch
```

Only the small one-class SVM is refitted on a short recent window; the pipeline, scaler and library stay fixed, which keeps each switch cheap.

8. Novelty and Contributions

Q-DRIFT proposes an adaptive learning mechanism for quantum kernels, not only an application of QML to a new dataset.

1. **Drift-triggered quantum re-encoding.** The feature map and its bandwidth are re-selected online whenever drift is detected, instead of being fixed at training time.
2. **First quantum evaluation on real spacecraft telemetry.** To our knowledge, quantum models have not been tested on the NASA SMAP/MSL benchmark; we evaluate continually, in time order, under three drift types.

3. **Resource-aware adaptation.** Every switch is charged for the quantum circuit evaluations it needs, and the controller trades detection quality against that cost through λ.
4. **Memory for recurring modes.** Encodings that worked for a mode are cached and reused instantly when the mode returns, such as sunlight and eclipse on each orbit.

Capability	Fixed quantum SVM	Statically optimised quantum SVM	Classical online model	Q-DRIFT
Quantum kernel	Yes	Yes	No	**Yes**
Encoding chosen for the data	No	Once, offline	Not applicable	**Online, after every drift**
Adapts after drift	No	No	Full retraining	**Swaps encoding only**
Reuses knowledge of past modes	No	No	No	**Yes, mode cache**
Tracks quantum cost of adapting	No	No	Not applicable	**Yes**

Table 2. How Q-DRIFT differs from the approaches it is compared against.

9. Dataset and Experimental Setup

The NASA JPL benchmark gives 82 telemetry channels with 105 expert-labelled anomaly sequences from two missions [1] (Table 3).

Mission	Telemetry channels	Anomaly sequences	Values evaluated
SMAP satellite (Earth orbit)	55	69	429,735
MSL Curiosity rover (Mars)	27	36	66,709
Total	**82**	**105**	**496,444**

Table 3. NASA SMAP/MSL telemetry benchmark [1].

Preprocessing. Each channel is z-normalised using its training segment only. A sliding window turns the stream into feature vectors, and PCA reduces each to 4–8 values, one per qubit. Data is never shuffled across time, to avoid leakage.

Drift scenarios. Natural mode changes already present in the data, plus controlled injections at known times: abrupt (a step change in level), gradual (a slow ramp) and recurring (two modes that alternate).

Baselines.

- Fixed quantum one-class SVM with one encoding chosen at the start
- Statically optimised quantum SVM (best encoding chosen offline on all training data)
- Classical one-class SVM with an RBF kernel
- Online classical model with the same ADWIN detector and retraining
- NASA's LSTM with dynamic thresholds [1], as a published reference point

Tools. Python, PennyLane [14] or Qiskit for 4–8 qubit simulation, scikit-learn (one-class SVM with precomputed kernels), the river library for ADWIN, NumPy and Matplotlib; Google Colab or a laptop CPU.

10. Evaluation Metrics and Expected Outcomes

We judge Q-DRIFT on detection quality, behaviour around drift and quantum cost, using strict event-level scoring without the "point-adjust" shortcut criticised in [10] (Table 4).

Metric	What it measures	Why it matters
Event-level F1	Whole anomaly events found vs missed and false events	Main detection quality, robust to long events
PR-AUC	Precision–recall over all thresholds	Threshold-free comparison on imbalanced data
False alarms per day	Alerts raised on normal data	Operator workload at mission control
Detection delay	Time from drift onset to ADWIN alarm	How fast the system notices change
Recovery time	Time until F1 returns to its pre-drift level	How fast the switch restores quality
Circuit evaluations per sample	Quantum kernel entries computed, incl. switches	Feasibility on real quantum hardware

Table 4. Evaluation metrics.

Expected outcomes (hypotheses to test, not results).

- After drift, the fixed quantum SVM's F1 falls and stays low; Q-DRIFT's F1 falls briefly and recovers after the switch.
- Q-DRIFT raises fewer false alarms than the fixed and statically optimised quantum SVMs in post-drift periods.
- The cache makes recovery from recurring drift faster and cheaper than from new drift.
- Q-DRIFT is competitive with classical baselines on F1; we will report honestly if a classical model is better, since our contribution is the adaptation mechanism, not a claimed quantum speed-up.

11. Applications, Limitations and Future Scope

Applications.

- Health monitoring for satellites and rovers, where operating modes change often
- Large satellite constellations, where thousands of spacecraft drift independently
- Other systems whose "normal" changes: power grids, industrial IoT sensors, patient monitors

Limitations and how we handle them.

Limitation	Our mitigation
SMAP/MSL labels have known flaws [10]	Event-level scoring; optional validation on the ESA benchmark [11]
Quantum kernel values concentrate as qubits grow [7]	Stay at 4–8 qubits; bandwidth is part of the library [5]
Kernel cost grows with the square of the samples	Sliding windows, subsampling [8], and logged circuit counts
Experiments run on simulators	Validate the final kernels on free IBM cloud quantum hardware if time allows

Table 5. Limitations and how we address them.

Future scope. Trainable feature maps that are fine-tuned after drift instead of picked from a fixed library; drift-aware quantum neural networks; noise-aware selection on real hardware; and deployment on onboard edge processors.

12. Conclusion

Q-DRIFT addresses a gap between quantum kernel methods, which depend heavily on their encoding, and real spacecraft telemetry, whose normal behaviour keeps changing. It detects drift online with ADWIN, re-selects the quantum feature map from a small library using kernel-target alignment and a cost penalty, and reuses cached encodings for recurring modes, all without full retraining. Evaluated on NASA's SMAP and MSL data against quantum and classical baselines, the project will show whether adapting the encoding restores detection quality after drift, and at what quantum cost. Beyond spacecraft, the same mechanism applies to any monitored system whose normal behaviour changes over time, such as power grids, industrial sensors and patient monitors.

References

1. K. Hundman, V. Constantinou, C. Laporte, I. Colwell and T. Soderstrom, "Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding," in *Proc. ACM SIGKDD*, 2018. arXiv:1802.04431.
2. J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys*, vol. 46, no. 4, 2014.
3. A. Bifet and R. Gavaldà, "Learning from time-changing data with adaptive windowing," in *Proc. SIAM Int. Conf. Data Mining (SDM)*, 2007, pp. 443–448.
4. V. Havlíček et al., "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, pp. 209–212, 2019.
5. R. Shaydulin and S. M. Wild, "Importance of kernel bandwidth in quantum machine learning," 2022. arXiv:2111.05451.
6. T. Hubregtsen, D. Wierichs, E. Gil-Fuster, P.-J. H. S. Derks, P. K. Faehrmann and J. J. Meyer, "Training quantum embedding kernels on near-term quantum computers," *Physical Review A*, vol. 106, no. 4, 2022. arXiv:2105.02276.
7. S. Thanasilp, S. Wang, M. Cerezo and Z. Holmes, "Exponential concentration in quantum kernel methods," *Nature Communications*, 2024. arXiv:2208.11060.

8. M. Kölle et al., "Towards efficient quantum anomaly detection: one-class SVMs using variable subsampling and randomized measurements," in *Proc. ICAART*, 2024.
9. "Applying quantum autoencoders for time series anomaly detection," 2024. arXiv:2410.04154.
10. R. Wu and E. Keogh, "Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress," 2020–22. arXiv:2009.13807.
11. "European Space Agency benchmark for anomaly detection in satellite telemetry," 2024. arXiv:2406.17826.
12. B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.
13. N. Cristianini, J. Shawe-Taylor, A. Elisseeff and J. Kandola, "On kernel-target alignment," in *Advances in Neural Information Processing Systems (NIPS)*, 2001.
14. V. Bergholm et al., "PennyLane: Automatic differentiation of hybrid quantum-classical computations," 2018. arXiv:1811.04968.