Quantum Kernels

Construct and test quantum kernel estimates while preserving learning and sampling controls.

Undergraduate 22 min lesson Dr. Charith Jayasekara

By the end, you'll be able to

- Calculate overlaps for a simple feature map.
- Build a small Gram matrix and inspect its properties.
- Count training and inference kernel-estimation costs.

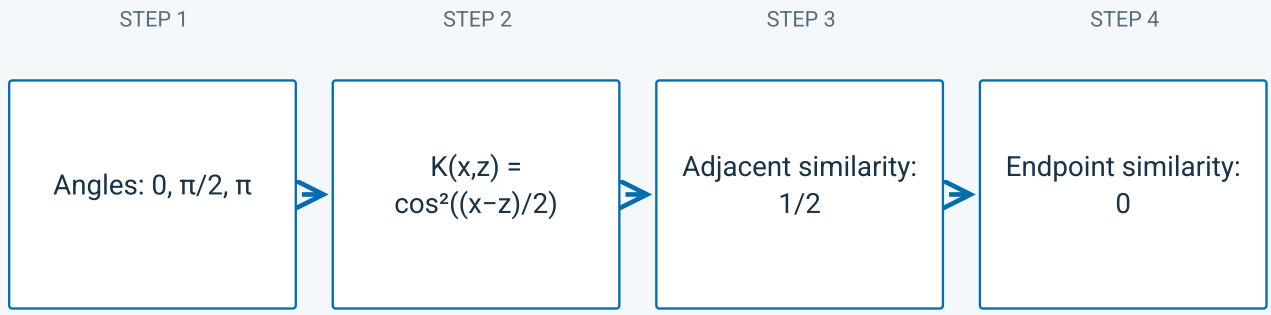
Measure similarity through state overlap

A kernel is a similarity function used by a classical learning algorithm. A quantum feature map prepares a state from an input example, and an overlap measurement estimates similarity between two such states. The quantum device supplies kernel values; a classical method can then train a classifier or regressor using that matrix.

Prerequisites: inner products, measurement statistics and the quantum-learning task description. A primary proposal and experiment studies quantum-enhanced feature spaces and kernel estimation. Havlíček and colleagues.

WORKED DIAGRAM · A30

One-qubit fidelity kernel



The ideal Gram matrix has diagonal 1, adjacent entries 1/2 and endpoint entries 0; its eigenvalues are 1 and 1 ± 1/√2.

The kernel setup

Choose a state-preparation circuit U(x) for each classical input x. A common fidelity kernel is the squared magnitude of the inner product between the two feature states. Its values lie between zero and one. Under the ideal construction, the resulting Gram matrix is positive semidefinite, a property required by many classical kernel methods.

Estimate one kernel entry by preparing U(x), applying the inverse of U(z), and measuring whether every qubit is zero. That event's probability is the overlap squared. The inverse must be implemented coherently with the correct order and angles. A SWAP-based overlap circuit is another option with different ancilla and depth requirements.

Worked example: one-qubit feature map

Use the Y-rotation feature state with angle x. Its overlap with the state at z is cosine of half the angle difference, so the kernel is

$$K(x, z) = \cos^2\left(\frac{x - z}{2}\right).$$

For inputs zero, pi over two and pi, the ideal Gram matrix has diagonal entries one, adjacent off-diagonal entries one half and the endpoint similarity zero. Thus zero and pi are orthogonal feature states, while the middle example is equally similar to both. All entries can be checked analytically before collecting shots.

The same kernel has an elementary classical formula. This example therefore demonstrates the pipeline but offers no computational advantage. A proposed quantum advantage needs a more demanding feature map and a learning task that benefits from it, together with a credible comparison against classical approximations or alternative kernels.

The three toy inputs give an explicit Gram matrix that can be checked before sampling.

$$(x_0, x_1, x_2) = (0, \pi/2, \pi), \qquad G_{ij} = K(x_i, x_j), \qquad G = \begin{pmatrix} 1 & 1/2 & 0 \\ 1/2 & 1 & 1/2 \\ 0 & 1/2 & 1 \end{pmatrix}.$$

The diagonal compares each feature state with itself. Adjacent examples have overlap-squared one half; the two endpoints are orthogonal. This exact matrix is a reference for noisy kernel estimates.

Sampling a kernel entry

If the true overlap is one half, one thousand independent measurements have a typical frequency uncertainty of about 1.6 percentage points. A kernel matrix for m training examples contains order m squared entries if all pairs are estimated independently. Measurement cost can therefore grow rapidly even before the classical learner is trained.

Finite-shot and hardware errors can produce an empirical matrix that is not exactly symmetric or positive semidefinite. Symmetrisation and matrix correction may be used, but they alter the estimator and should be reported. Reusing a noisy matrix without checking its properties can create numerical problems unrelated to the underlying learning theory.

Train and evaluate without leakage

Fit preprocessing and feature-map choices on training data, tune them using validation data and reserve a final test set. The training Gram matrix compares training examples with each other. Predicting a new example generally needs overlaps with relevant training examples, such as support vectors. Include that inference-time cost rather than counting only training.

A flexible feature-map search can overfit validation choices just as a classical hyperparameter search can. Report the selection budget and use repeated splits or independent evaluations when appropriate. The fact that kernel entries came from a quantum device does not exempt the model from ordinary statistical learning controls.

Useful features and difficult simulation are different

A kernel can be computationally difficult to estimate yet unhelpful for the labels. Conversely, a simple classically computable kernel can generalise well. The task is to establish both meaningful predictive performance and a resource advantage for obtaining that performance. Neither follows solely from the feature-space dimension.

Some quantum kernels concentrate near a common value as system size grows, making distinctions difficult to resolve. Noise can worsen that effect. A model should therefore examine kernel-value distributions and required shot precision, not merely demonstrate that an encoding entangles qubits. Problem structure and inductive bias matter.

A reproducible procedure

Define the feature map and kernel formula. Verify small analytic overlaps. Select a shot allocation and record the measurement circuit. Construct the training matrix, inspect its symmetry and spectrum, and document any corrections. Train the classical learner under a fixed tuning policy. Evaluate unseen examples and compare total cost and accuracy with well-chosen classical kernels and features.

For the one-qubit example, compare the sampled matrix with the exact cosine formula. If a large discrepancy remains after increasing shots, inspect gate inversion, angle units, wire ordering and hardware error rather than blaming statistical variation alone. This provides a useful end-to-end debugging exercise.

Verify the Gram matrix is not just a picture

For the three toy inputs, the Gram matrix has eigenvalues one, one plus one over the square root of two, and one minus one over the square root of two. All are positive, consistent with the ideal kernel’s positive-semidefinite property. The smallest is about 0.292893, so this particular tiny matrix is not singular.

Finite-shot perturbations can shift those eigenvalues. In a larger matrix with a near-zero eigenvalue, even small entry noise can produce a negative empirical value. If a correction projects the matrix back to a positive-semidefinite one, report the correction and test its effect on predictions. It is a modelling and numerical-processing choice, not an invisible quantum operation.

Also check the diagonal: ideal overlap of a state with itself is one. A noticeably lower measured diagonal may reveal hardware errors or a circuit-inversion problem. It should not be quietly interpreted as a new feature-space geometry without explaining the noisy-kernel definition.

Limits and primary context

Quantum kernels are a family of methods, not a universal replacement for classical learning. The primary feature-space work supplies a concrete research direction; its existence does not establish a deployment advantage for arbitrary datasets. Make dataset, encoding, accuracy, access assumptions and comparison methods explicit before drawing practical conclusions.

Exercises and worked answers

Exercise 1. What is the toy kernel between angles zero and pi over three?

Check the worked answer

It is cosine squared of pi over six, equal to three quarters. The corresponding all-zero event should occur with that probability in the ideal overlap circuit.

Exercise 2. Why can a training dataset with ten times as many examples require roughly one hundred times as many pairwise kernel estimates?

Check the worked answer

A full Gram matrix has a quadratic number of pairs. Symmetry reduces the constant but not the quadratic scaling. Structured approximations can change the workflow, but their accuracy and implementation must then be included in the comparison.

Check your understanding

PAUSE & THINK

What is K(0, π/3) for this feature map?

- [ ] 1/4
- [ ] 1/2
- [ ] 3/4

Answer: 3/4. The overlap squared is $\cos^2(\pi/6) = 3/4$.

Sources

- Havlíček and colleagues