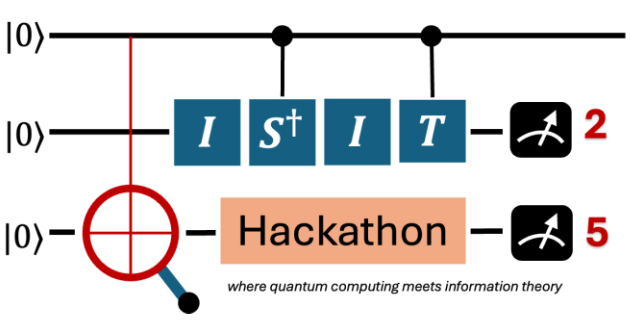
ISIT JUNE 22-27 ANN ARBOR MICHIGAN

Quantum Kernels and Feature Maps

Concepts and Examples



Outline

- Objective
- Kernel Methods for Machine Learning
- Classification & Clustering
- Kernel Principal Component Analysis
- Conclusion

Machine Learning

- ML finds and studies patterns in data
- Better with higher dimensional feature space
- ML algorithms based on it are kernel methods
- Useful for classification and clustering

ISIT JUNE 22-27 ANN ARBOR MICHIGAN

Kernel Methods

- Algorithms using kernel functions
- Best applied in Support Vector Machine (SVM)
- Supervised learning for classification tasks
- For non-linearly separable data spaces using kernels to find boundaries
- Kernel functions imply maps into high dimensional space
- ‘Kernel trick’ & ‘Spectral Clustering’

Kernel Functions

$$k(\vec{x}_i, \vec{x}_j) = \langle f(\vec{x}_i), f(\vec{x}_j) \rangle$$

where:

- k is the kernel function
- $\vec{x}_i, \vec{x}_j$ are n dimensional inputs
- f is a map from n-dimension to m-dimension space and
- $\langle a, b \rangle$ denotes the inner product

Dimensional inputs

Space definition for mapping n x m dimensions

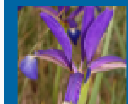
Inner product

Kernel Functions

For finite data, a kernel function can be represented as a matrix

$$K_{ij} = k(\vec{x}_i, \vec{x}_j)$$

Dataset



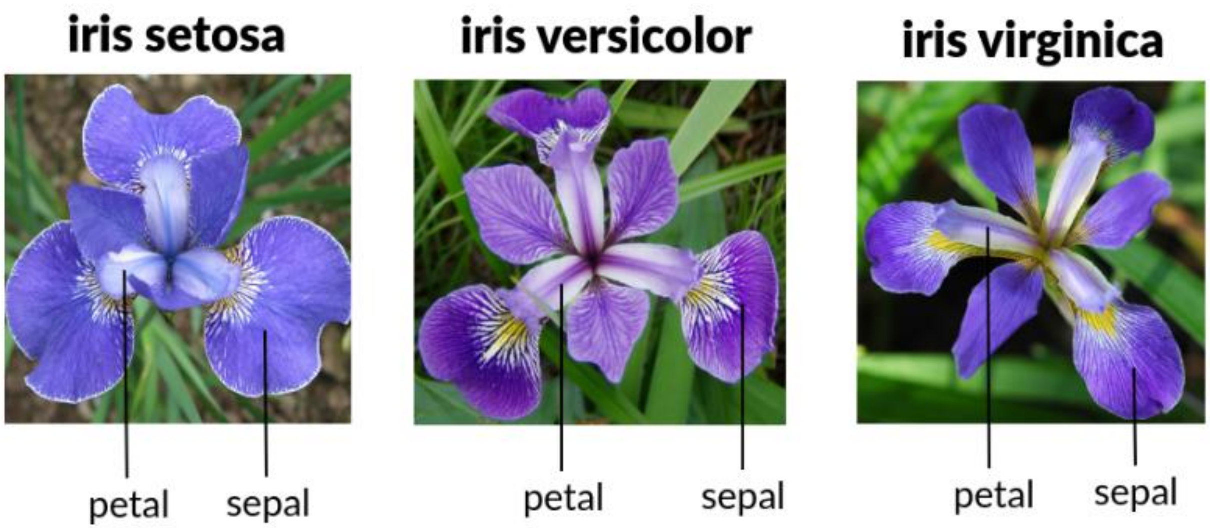
Iris

Donated on 6/30/1988

A small classic dataset from Fisher, 1936. One of the earliest known datasets used for evaluating classification methods.

Dataset Characteristics	Subject Area	Associated Tasks
Tabular	Biology	Classification

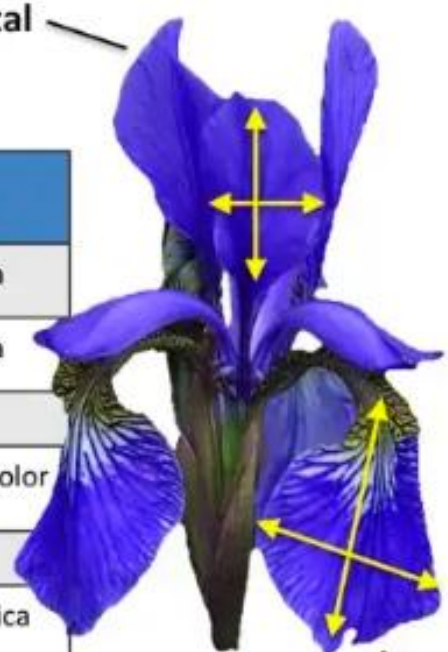
Feature Type	# Instances	# Features
Real	150	4



Iris dataset

Samples
(instances, observations)

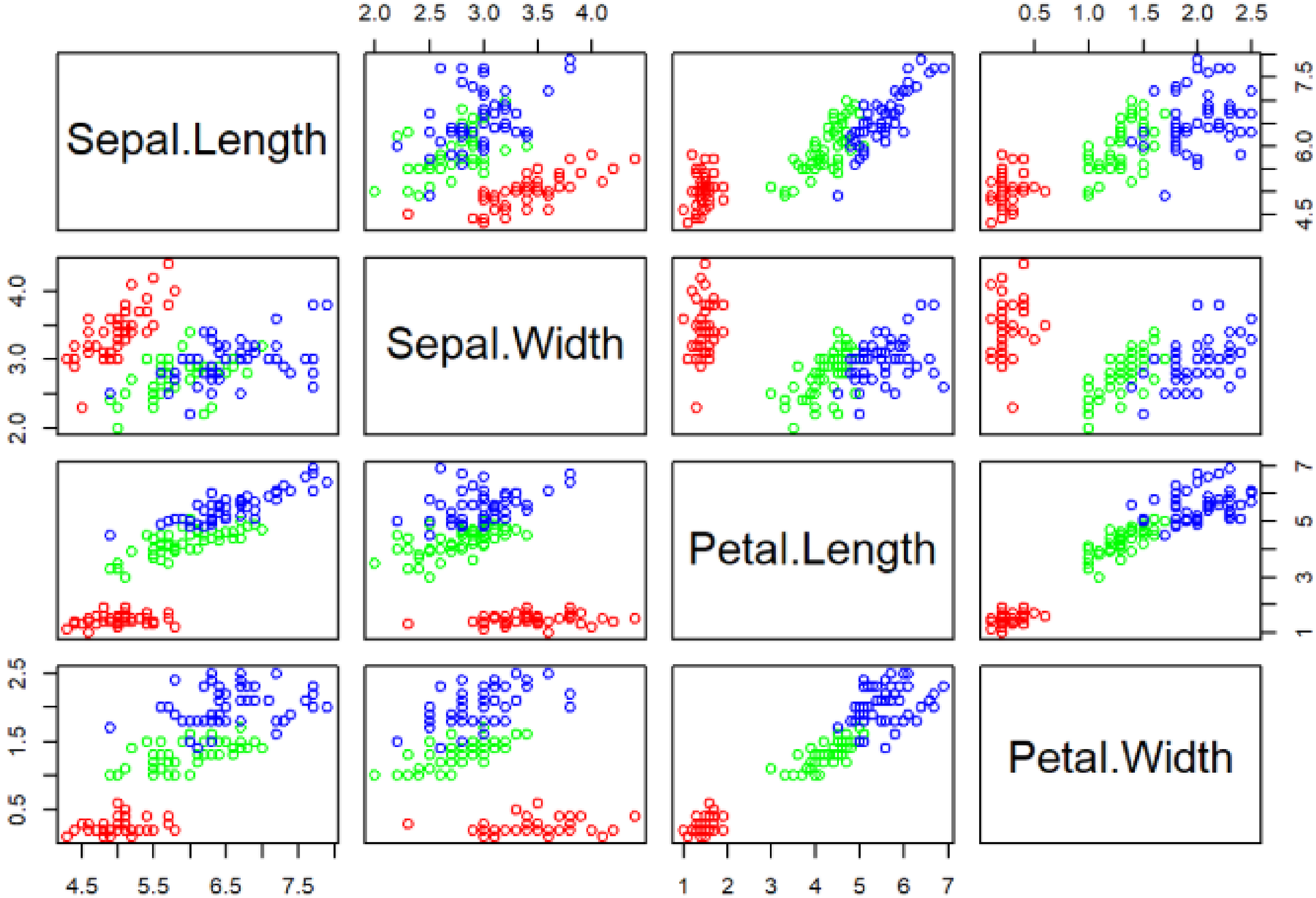
	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica



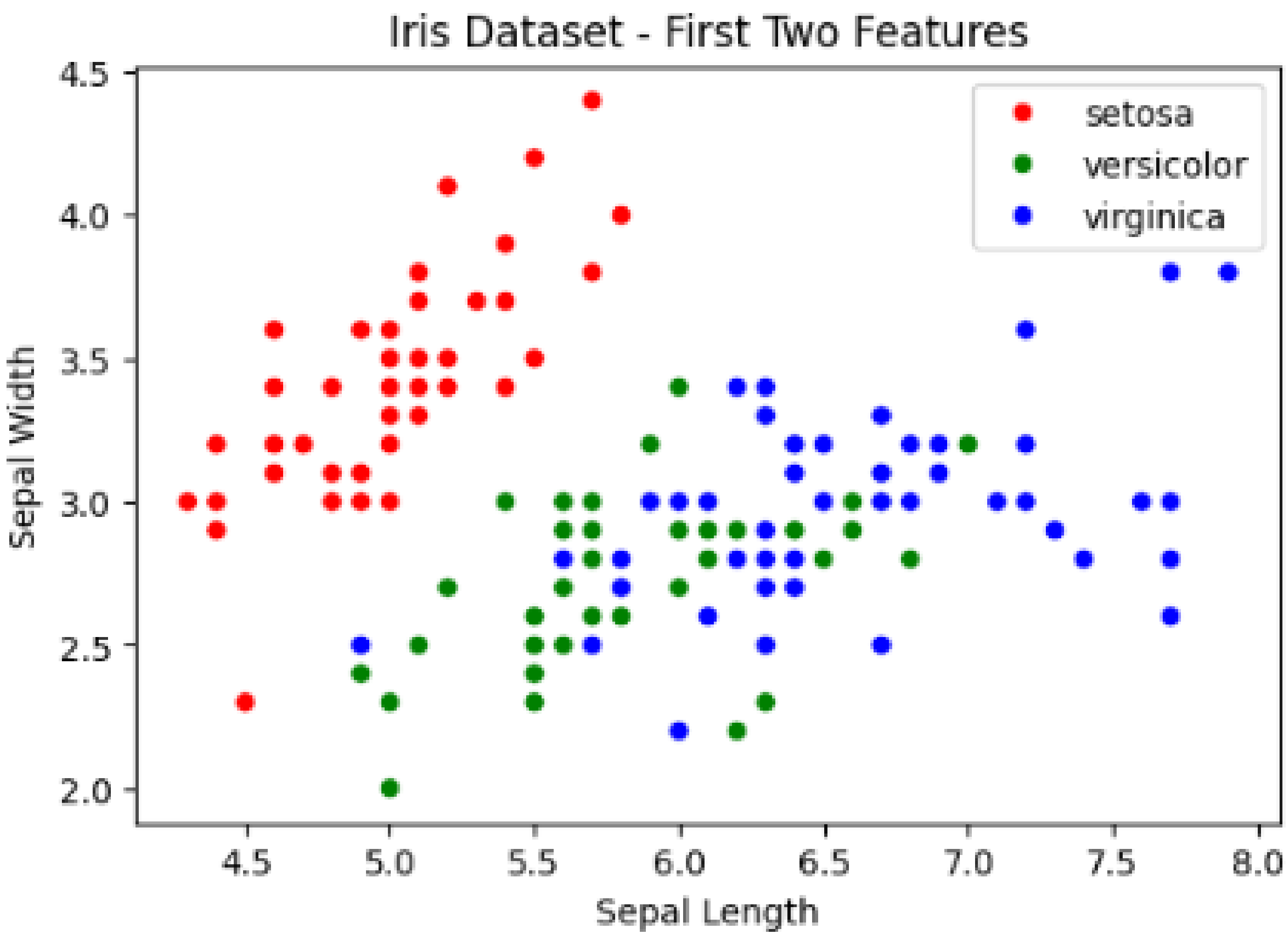
Features
(attributes, measurements, dimensions)

Class labels
(targets)

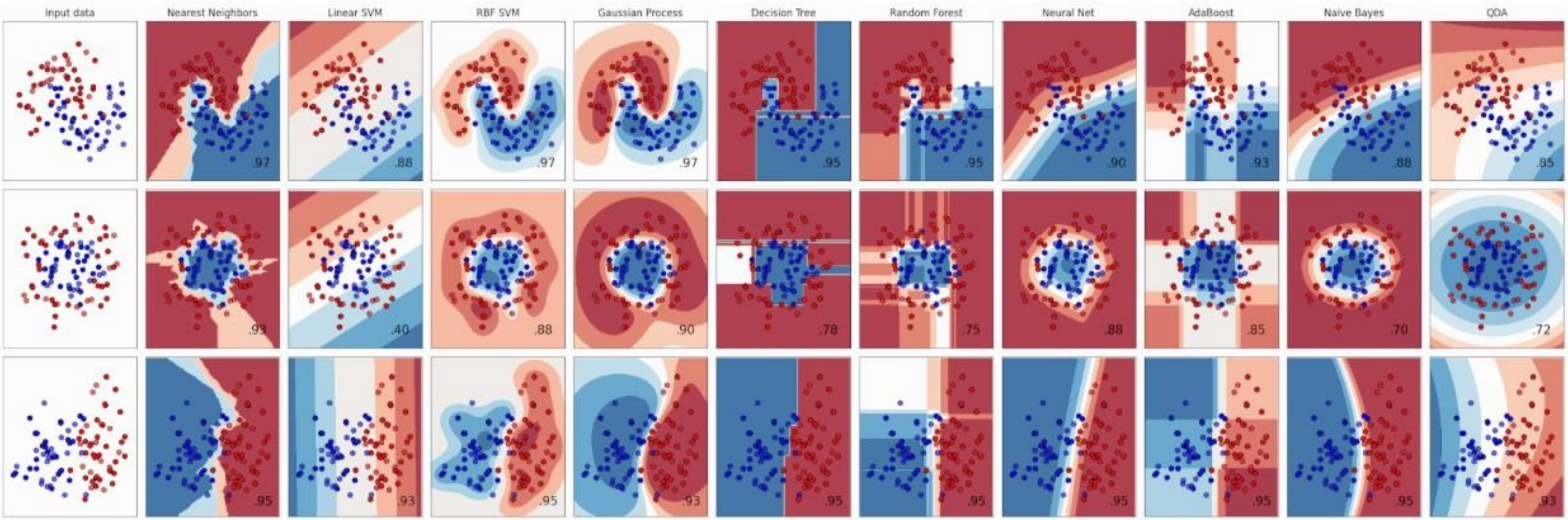
Classification



Classification



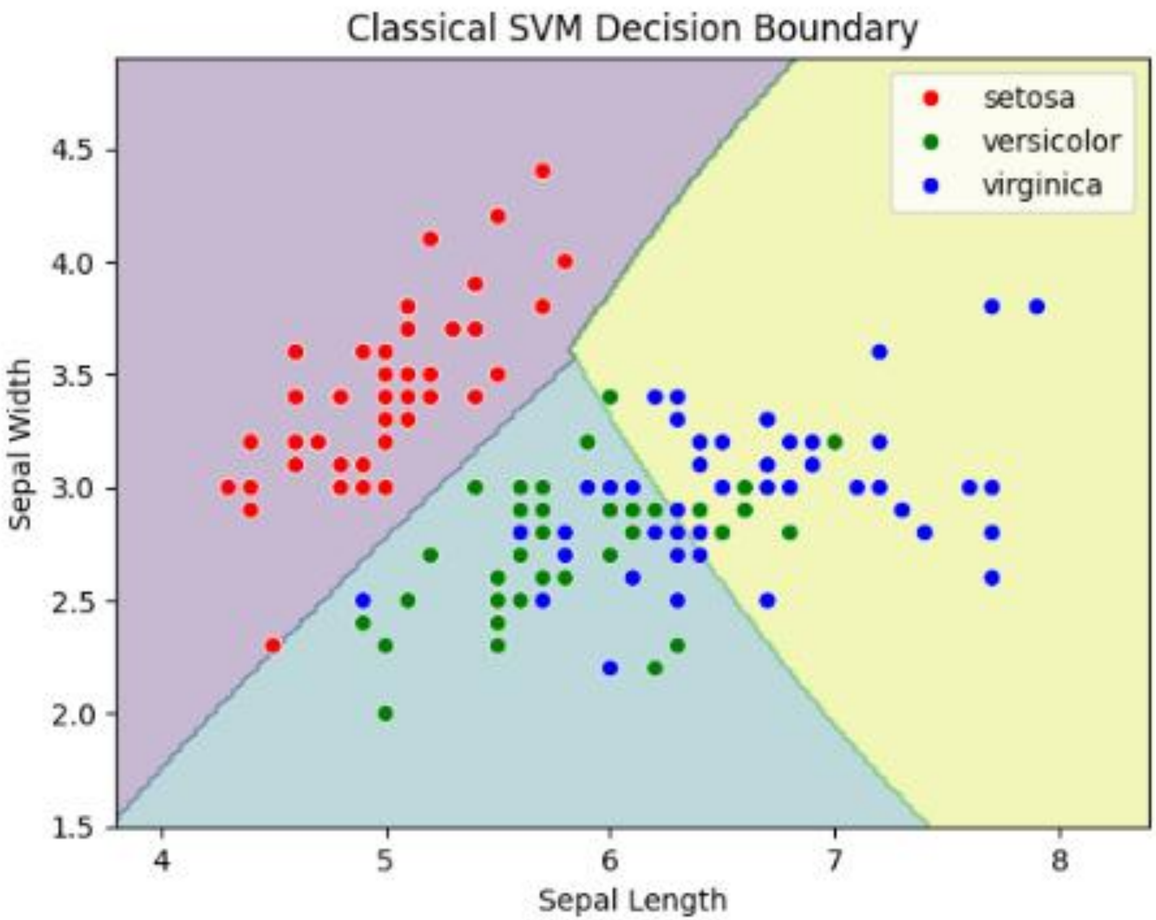
Classification



SVM Training

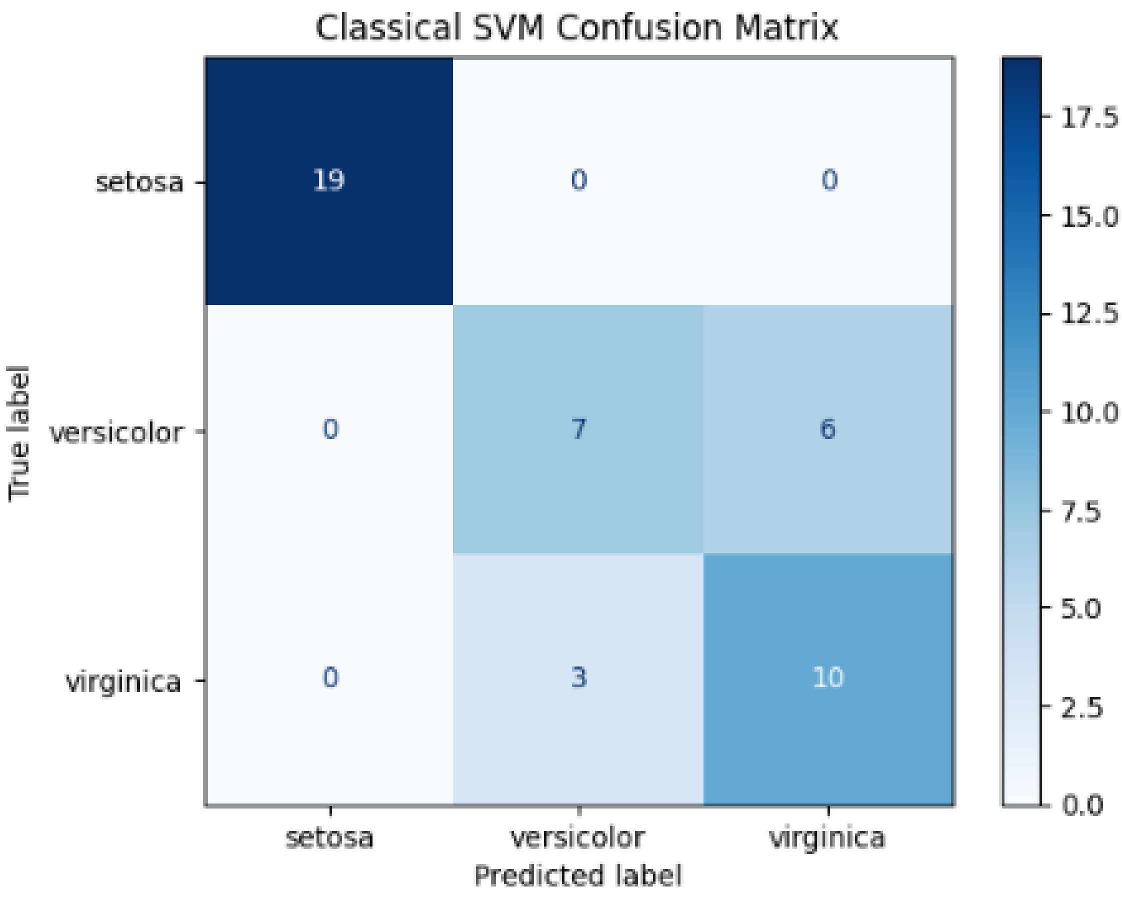
- Use kernel matrix with SVC
- Optimize 'C' with GridSearchCV

Classical SVM classification



Classical SVM Accuracy: 0.80

Classical SVM classification




```
Classical SVM Accuracy: 0.80
```

Classical SVM classification

Concept	Purpose	Notes
SVM	Classify data by separating with a hyperplane	Effective in high dimensions
RBF Kernel	Allows nonlinear separation	Common default kernel
Accuracy	Overall correctness	Simple and widely used
Confusion Matrix	Detailed error breakdown	Shows per-class performance
Decision Boundary	Visual intuition	Only possible in 2D or 3D

Qiskit Implementation

- Install Qiskit packages, prepare data
- Defining Qiskit feature map

Feature Map	What it Does	Best For
`ZFeatureMap`	Applies only Z-rotations	Simple data
`ZZFeatureMap`	Adds ZZ entanglement between qubits	Captures correlations
`PauliFeatureMap`	Uses X, Y, Z rotations	High flexibility

Quantum Kernels

- Quantum kernel machine learning
- Quantum states overlapping measure similarity
- Quantum feature mapping selection
- Enable SVMs with complex data boundaries

Quantum Kernels

$$K_{ij} = \left| \langle \phi(\vec{x}_i) | \phi(\vec{x}_j) \rangle \right|^2$$

where:

- K_{ij} is the kernel matrix
- $\vec{x}_i, \vec{x}_j$ are n dimensional inputs
- $\phi(\vec{x})$ is the quantum feature map
- $|\langle a|b \rangle|^2$ denotes the overlap of two quantum states a and b

Quantum Kernel

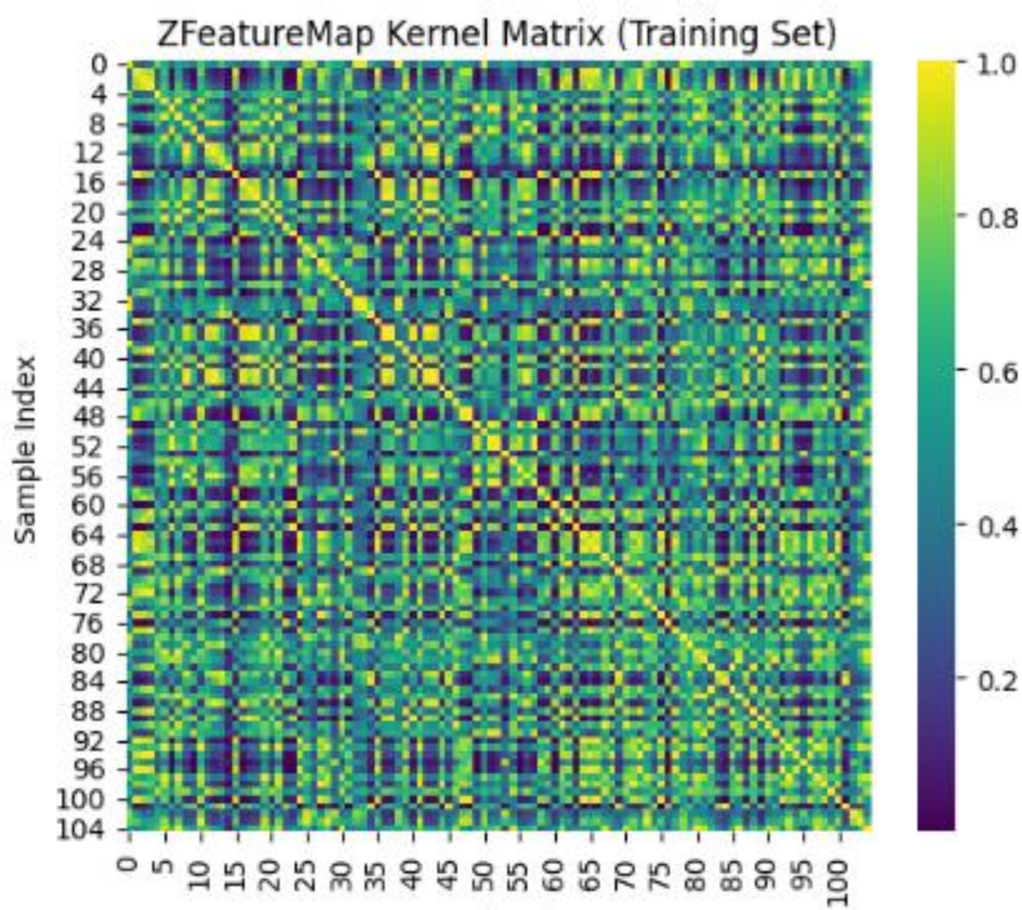
- Install Qiskit packages, prepare data
- Defining Qiskit feature map

Feature Map	What it Does	Best For
`ZFeatureMap`	Applies only Z-rotations	Simple data
`ZZFeatureMap`	Adds ZZ entanglement between qubits	Captures correlations
`PauliFeatureMap`	Uses X, Y, Z rotations	High flexibility

Feature Maps

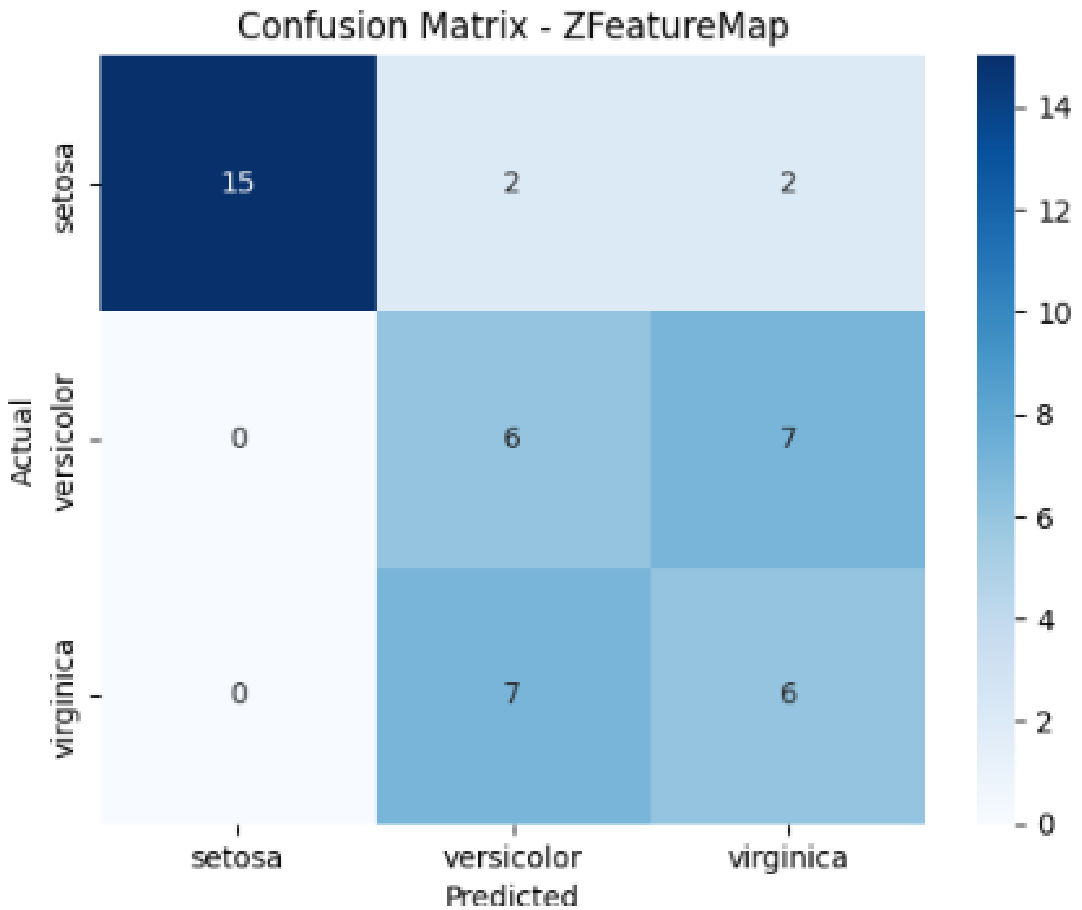
- ZFeatureMap: Z-Rotations
- ZZFeatureMap: Entaglement
- PauliFeatureMap: X-Y-Z Rotations

ZFeatureMap



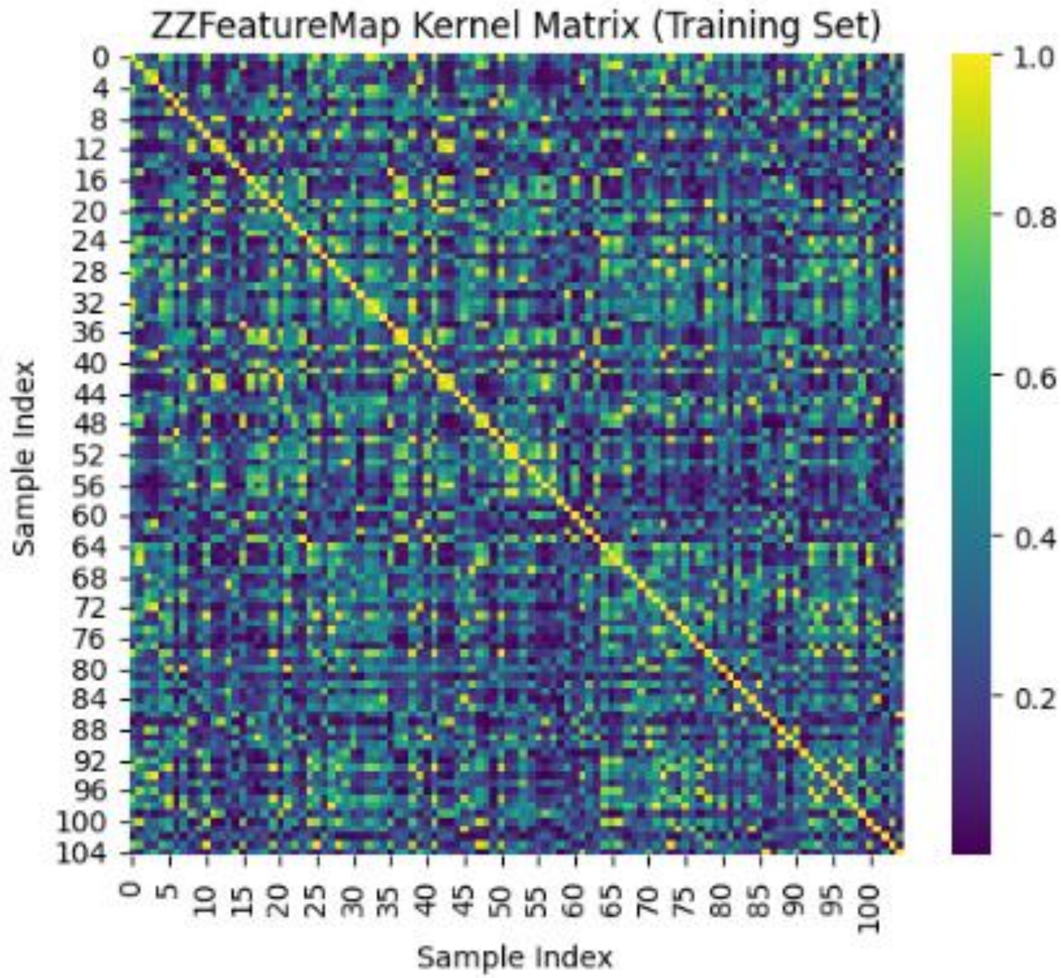
Quantum SVM Accuracy with ZFeatureMap: 0.60

ZFeatureMap



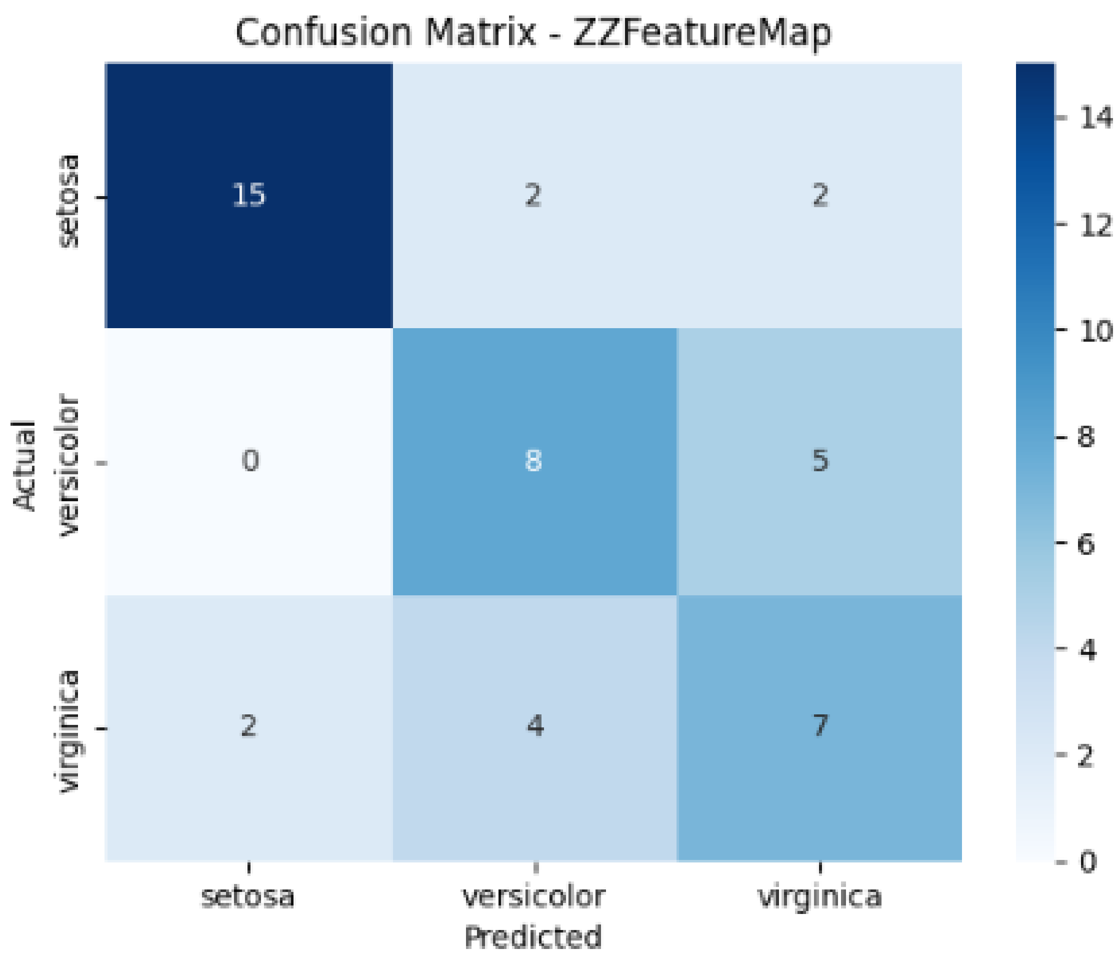

```
Quantum SVM Accuracy with ZFeatureMap: 0.60
```

ZZFeatureMap



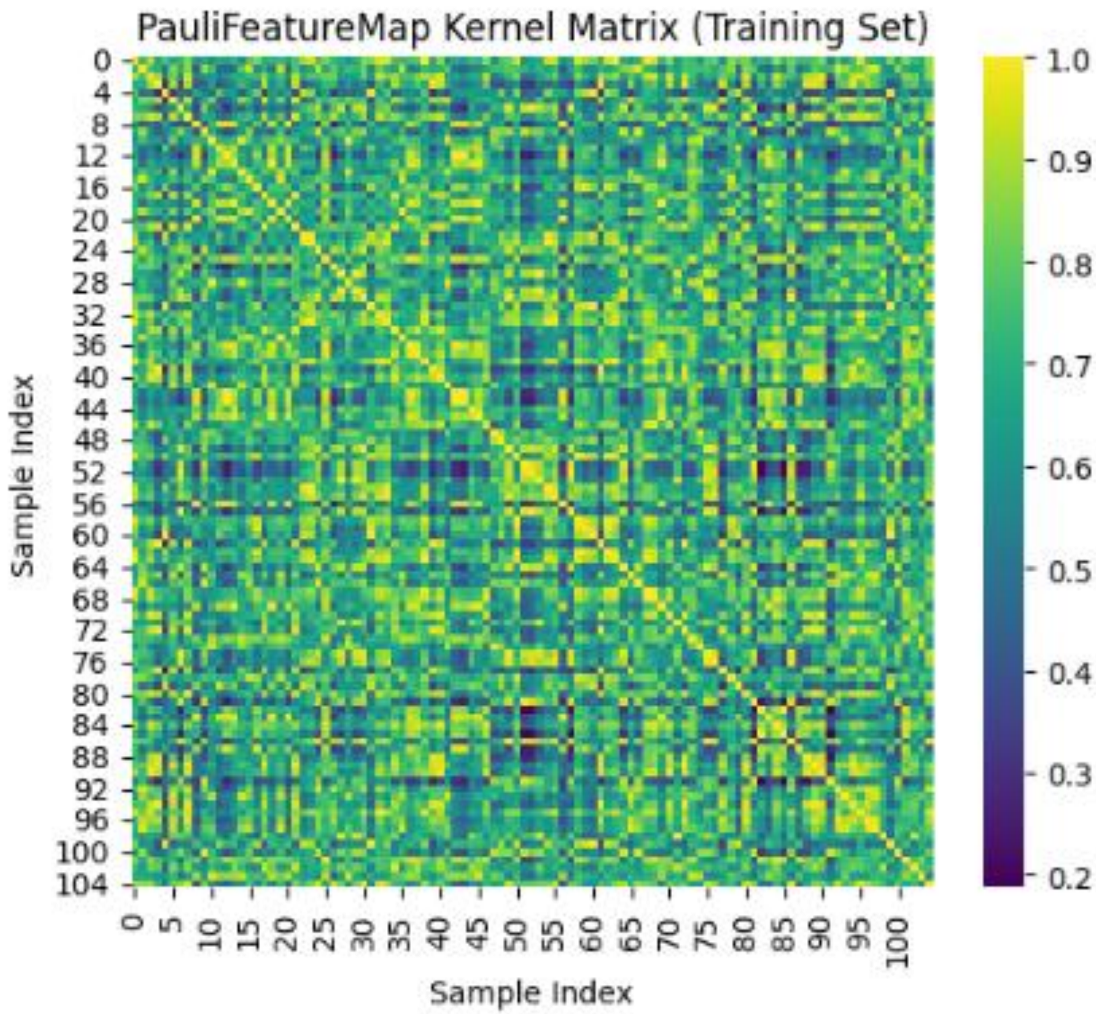
Quantum SVM Accuracy with ZZFeatureMap: 0.67

ZZFeatureMap



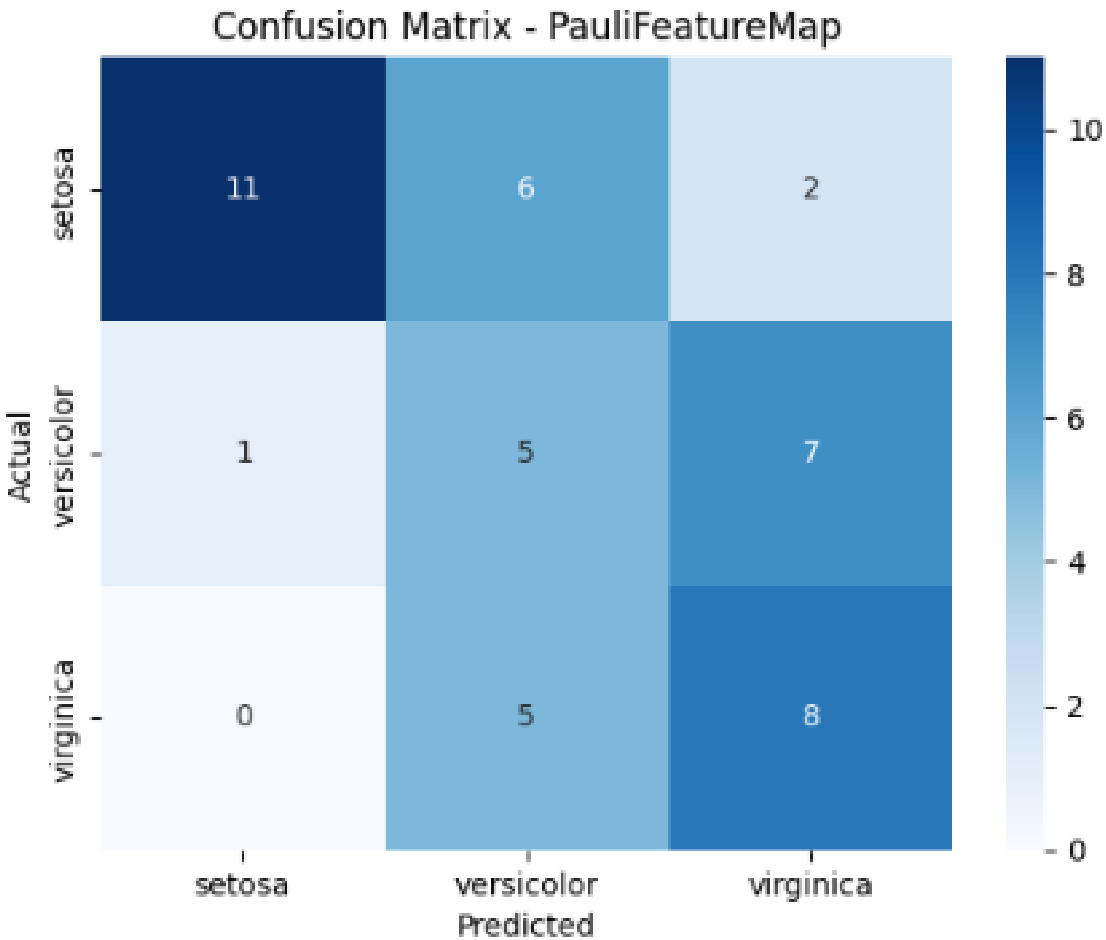

```
Quantum SVM Accuracy with ZZFeatureMap: 0.67
```

PauliFeatureMap



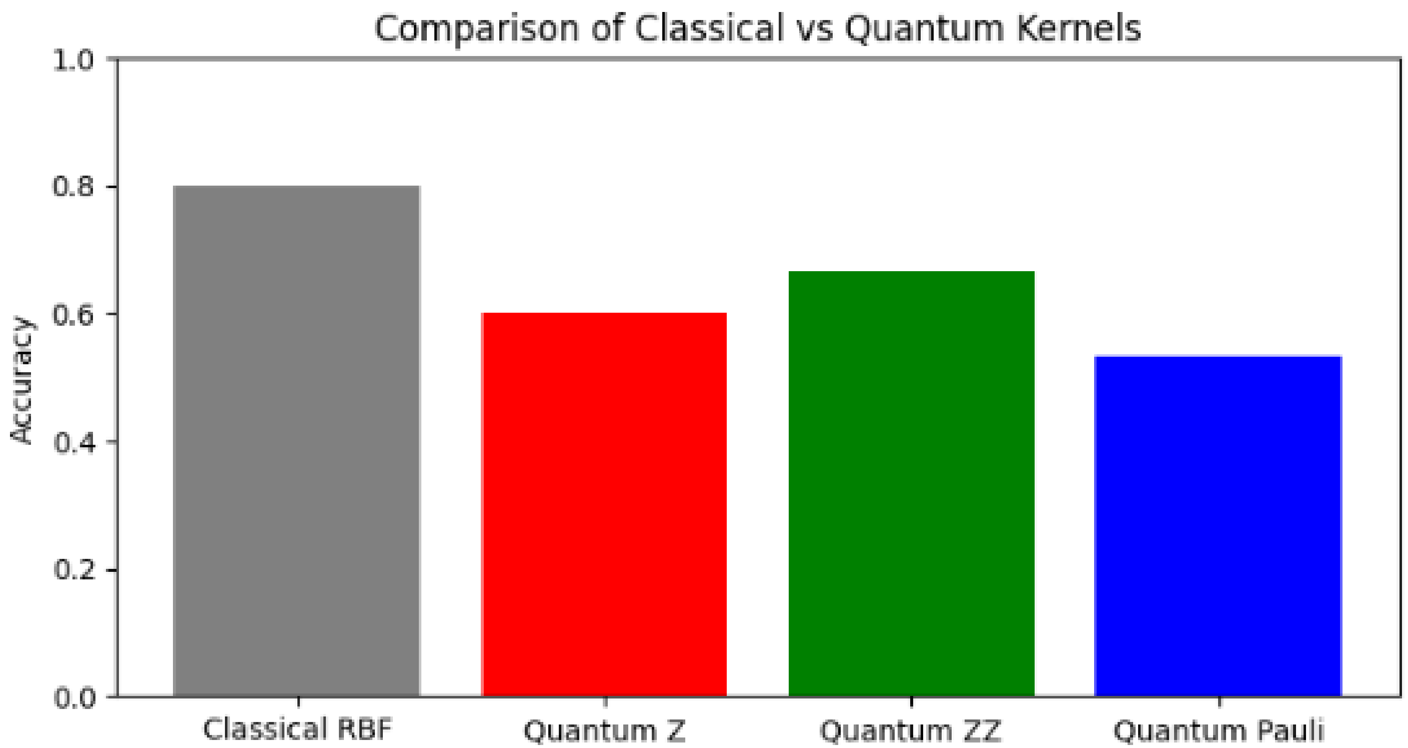

```
Quantum SVM Accuracy with PauliFeatureMap: 0.53
```

PauliFeatureMap




```
Quantum SVM Accuracy with PauliFeatureMap: 0.53
```

Results Comparison

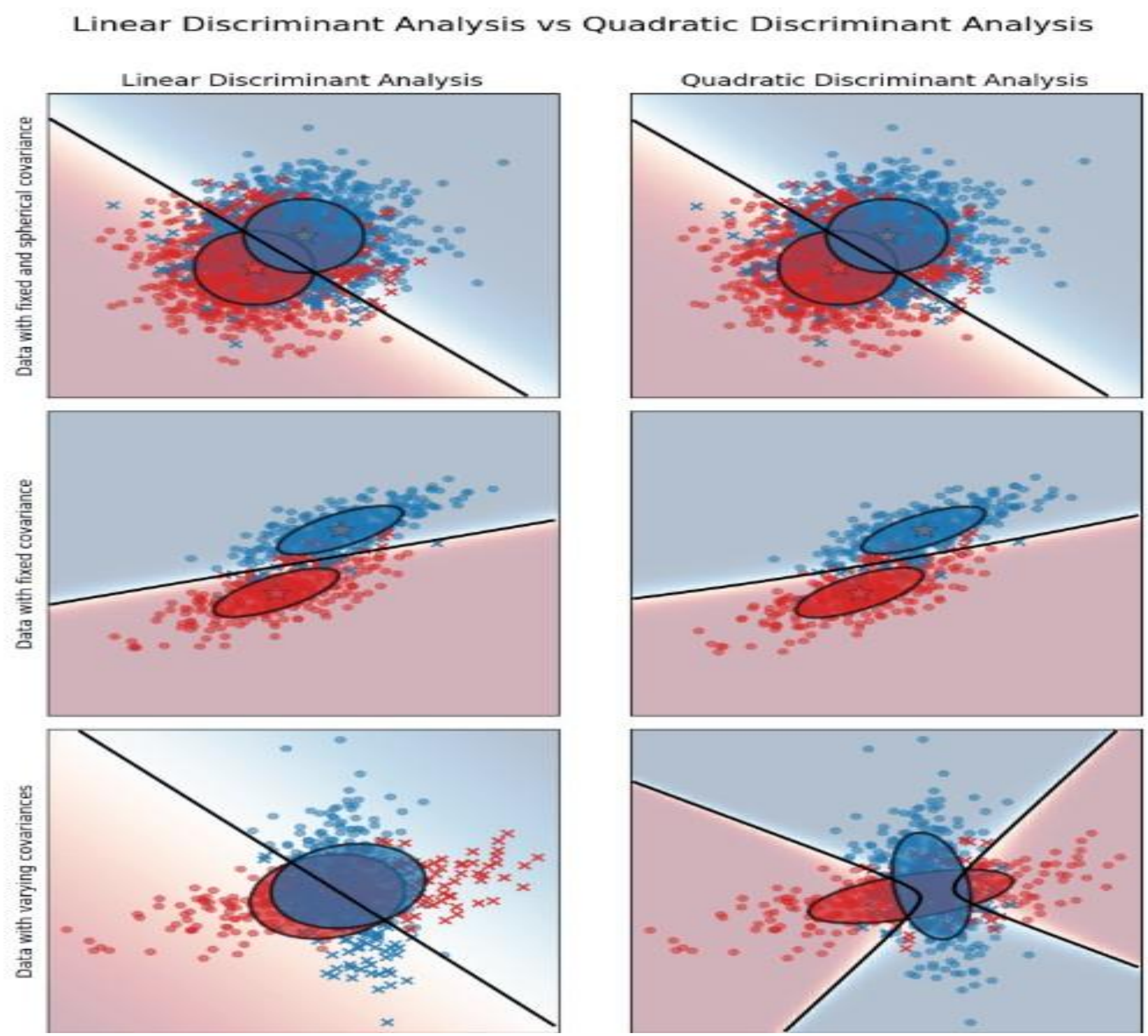



```
Classical SVM Accuracy: 0.80
Quantum SVM Accuracy with ZFeatureMap: 0.60
Quantum SVM Accuracy with ZZFeatureMap: 0.67
Quantum SVM Accuracy with PauliFeatureMap: 0.53
```

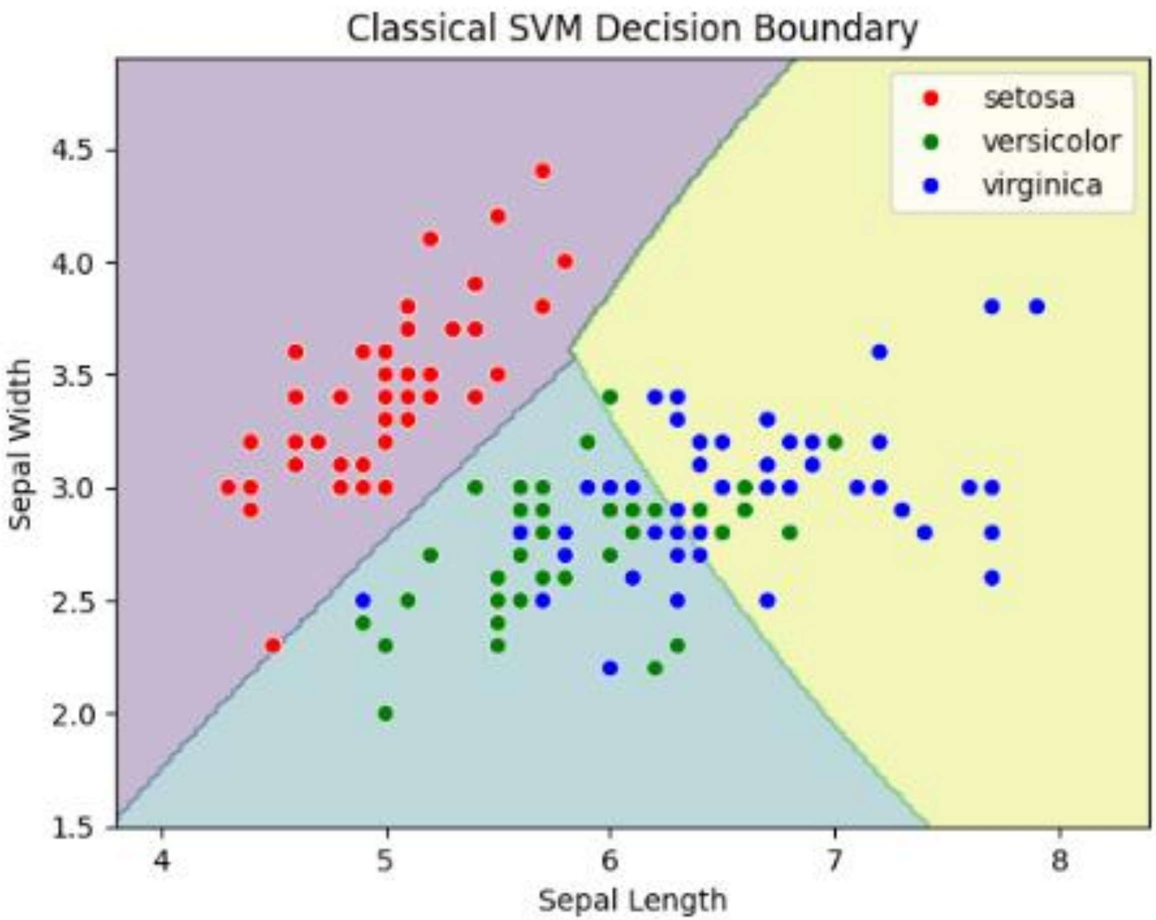
Underperformance Diagnosis

- Feature Dimensionality
- Number of Qubits and Feature Map Depth
- Quantum Kernel Evaluation Limitations
- Model Complexity and Class Structure

Analysis Comparison

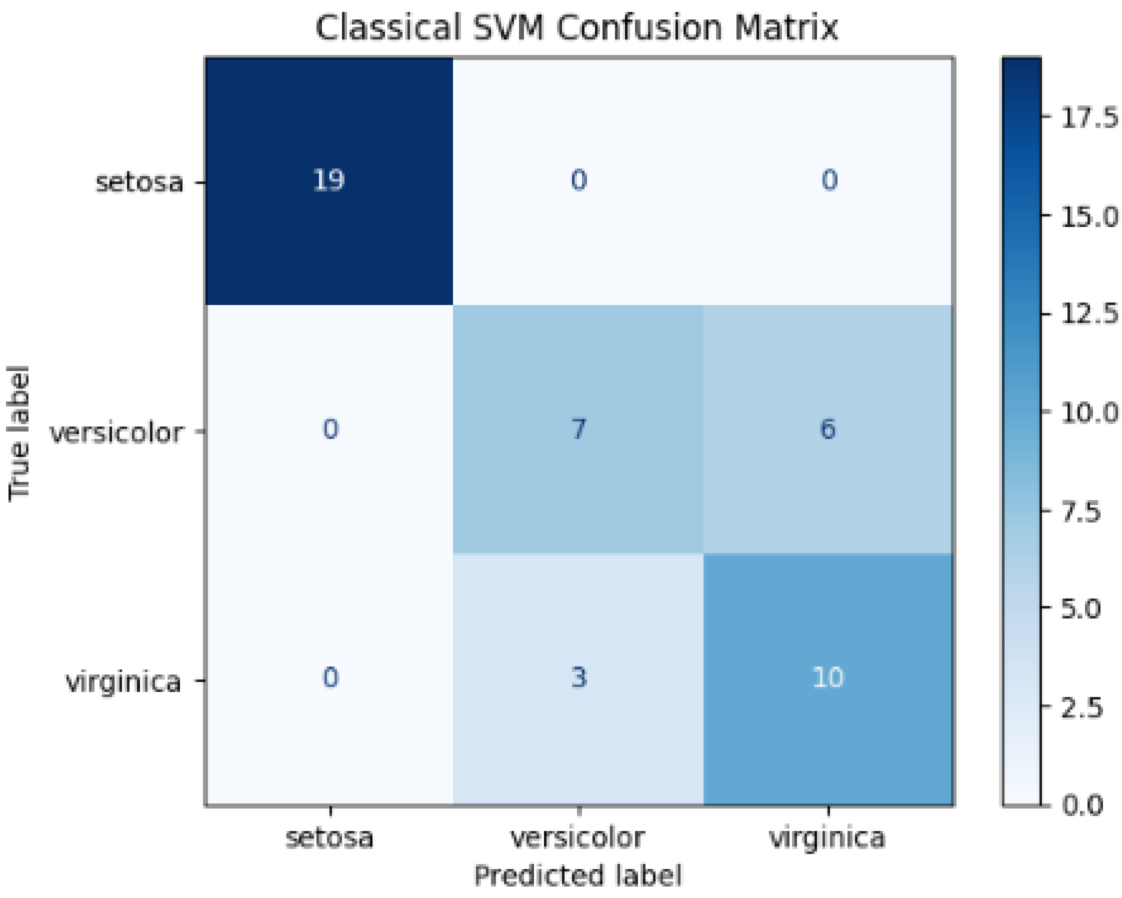


Classical SVM classification



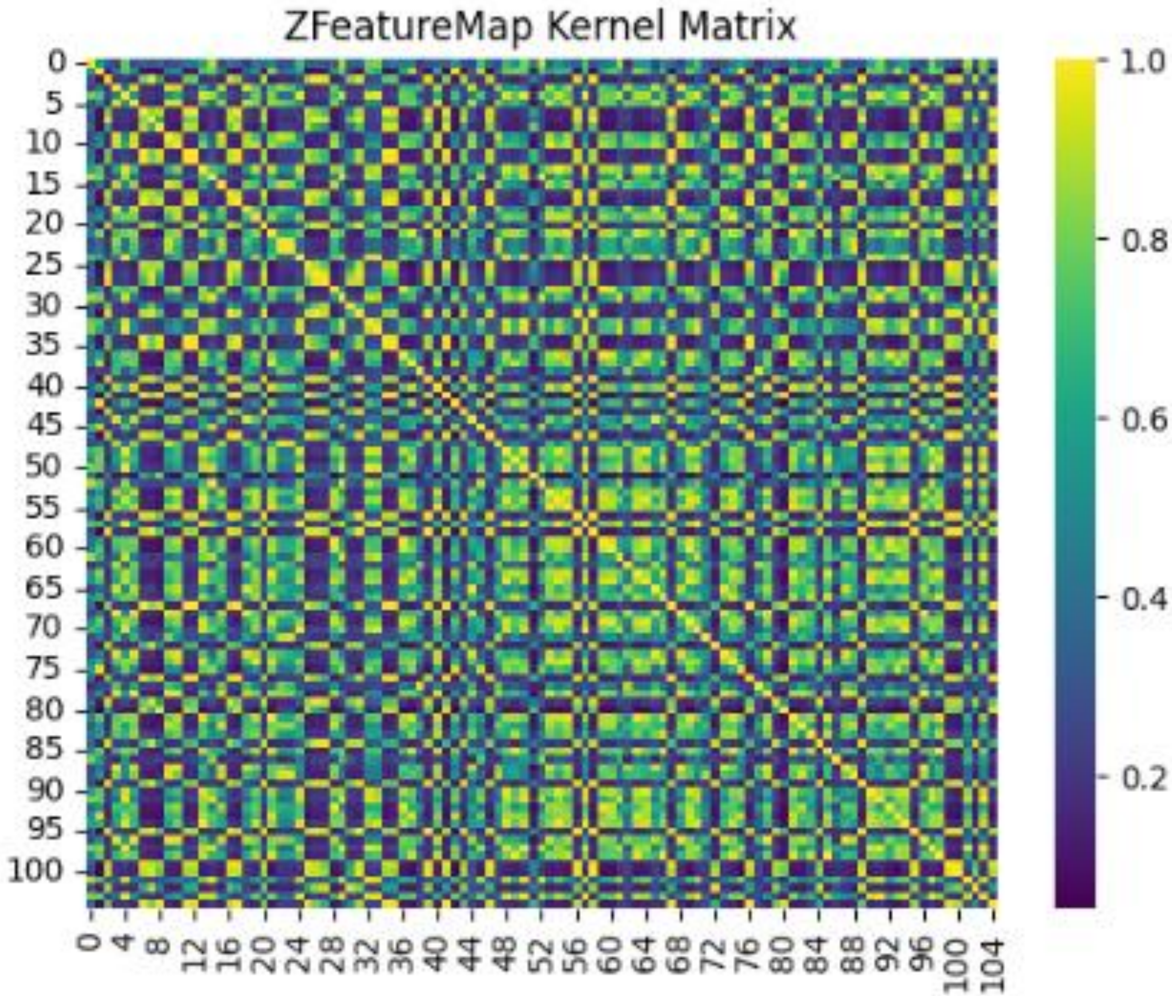

```
Classical SVM Accuracy: 0.80
```

Classical SVM classification



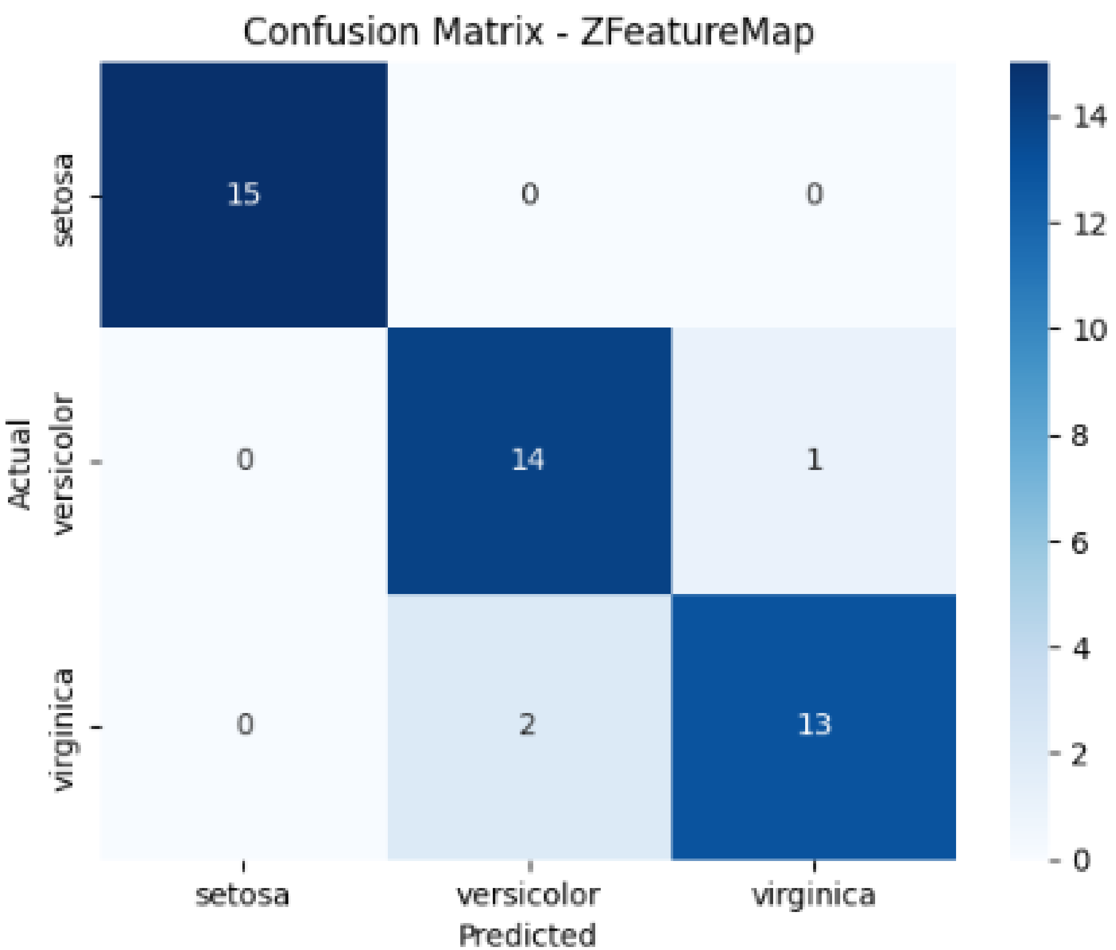

```
Classical SVM Accuracy: 0.80
```

ZFeatureMap



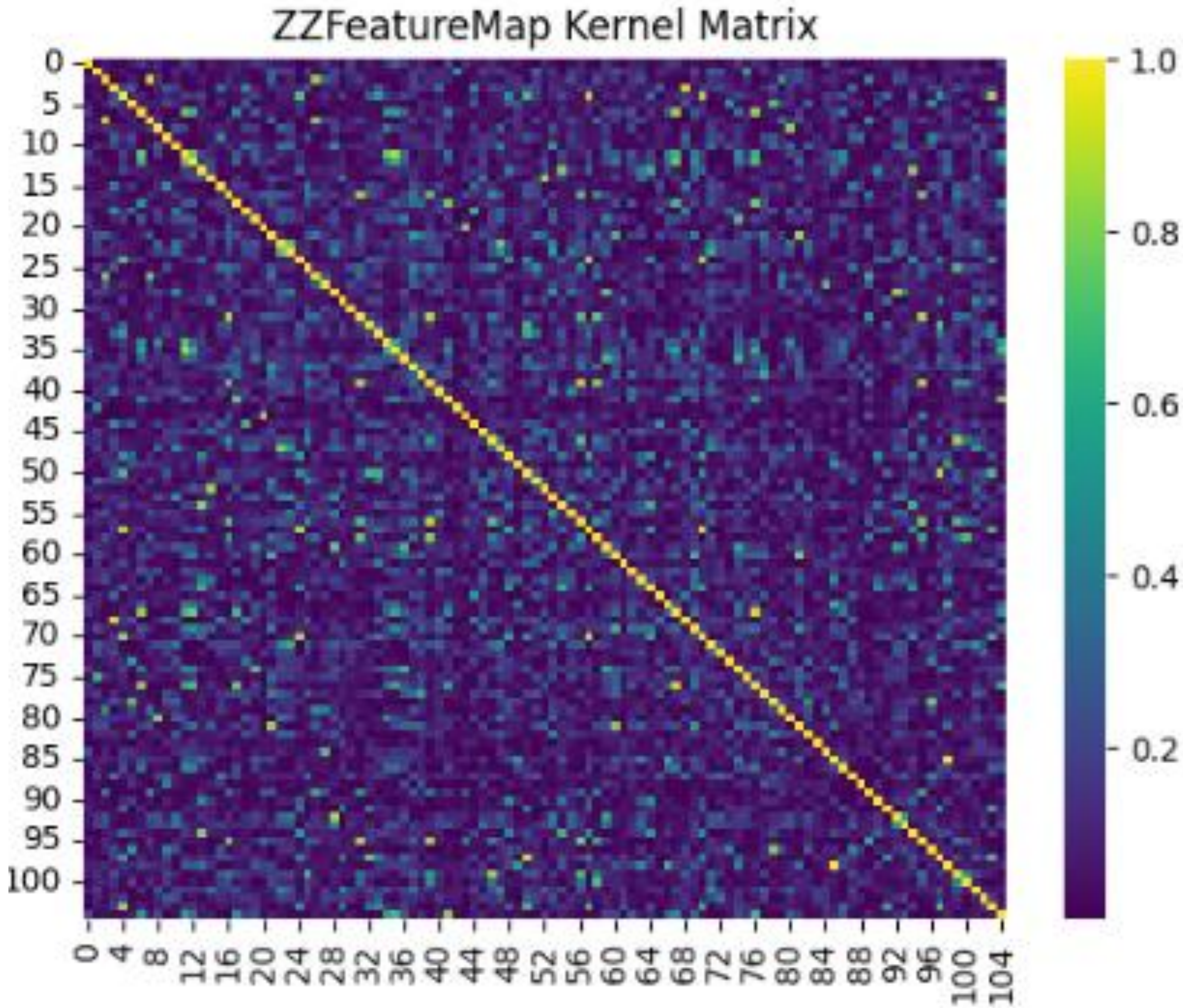
ZFeatureMap QSVM Accuracy: 0.93

ZFeatureMap



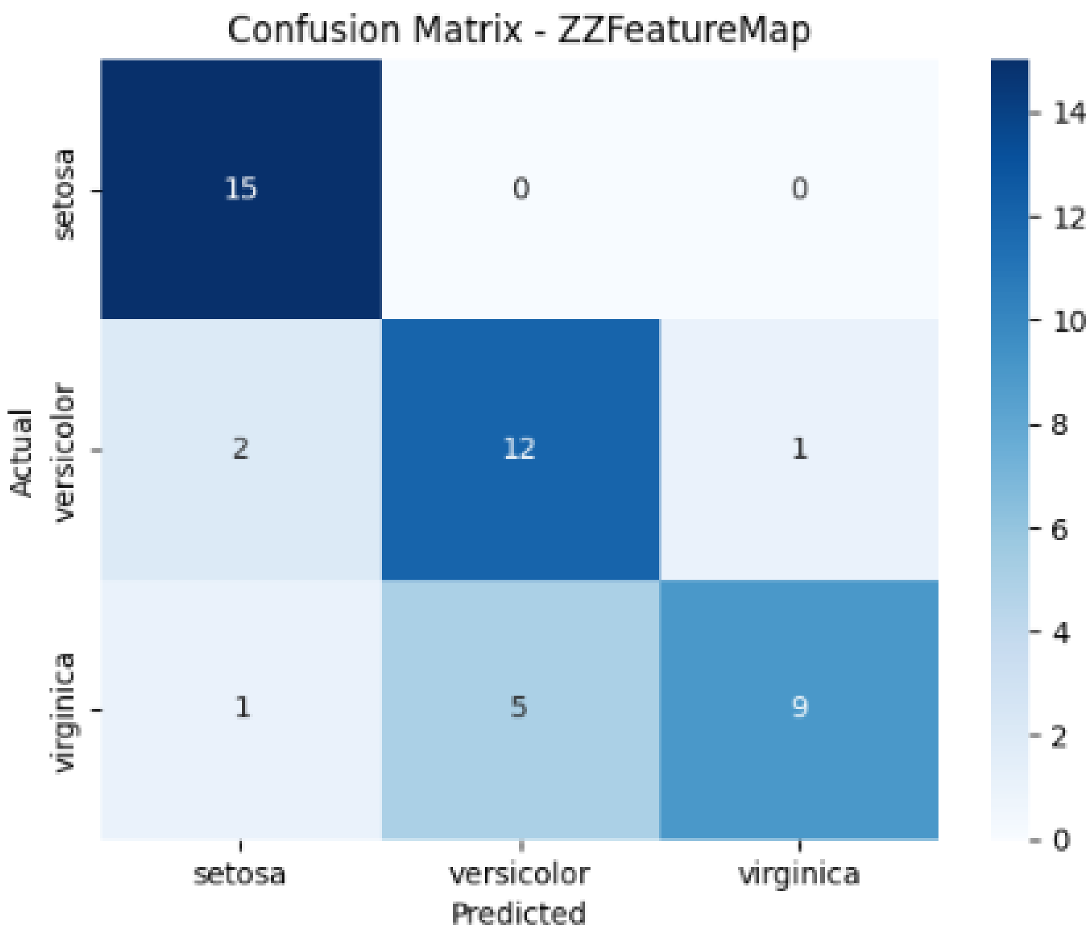
ZFeatureMap QSVM Accuracy: 0.93

ZZFeatureMap



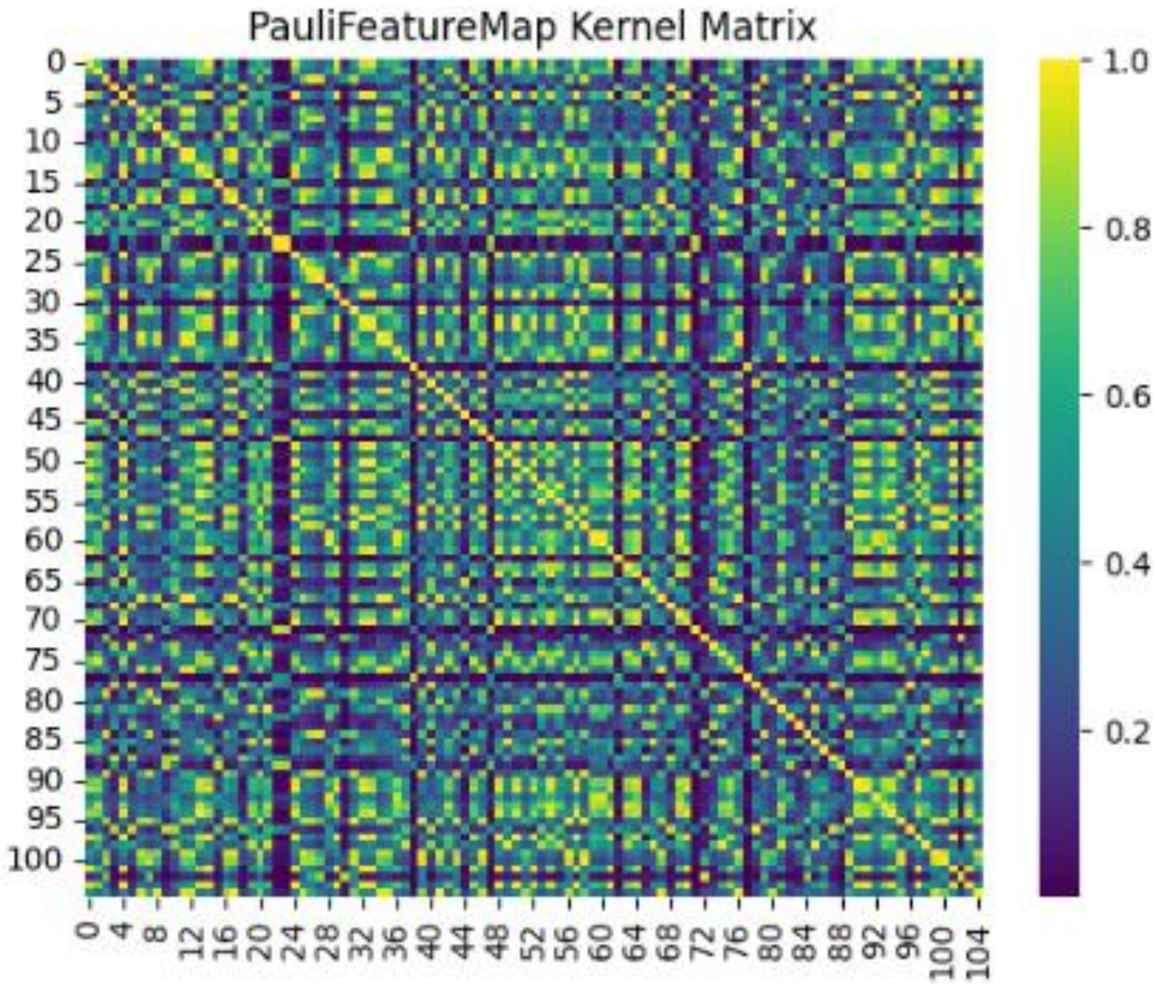
ZZFeatureMap QSVM Accuracy: 0.80

ZZFeatureMap



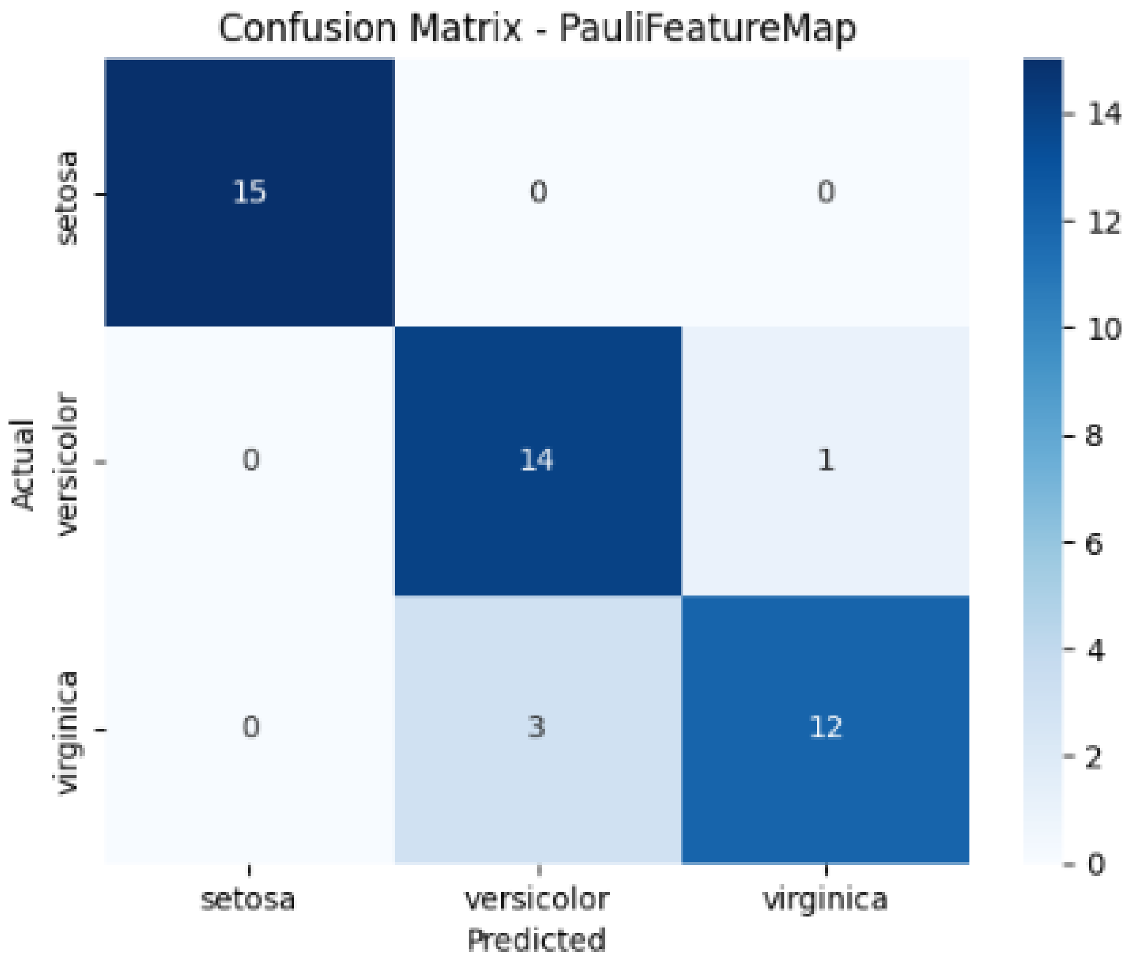
ZZFeatureMap QSVM Accuracy: 0.80

PauliFeatureMap



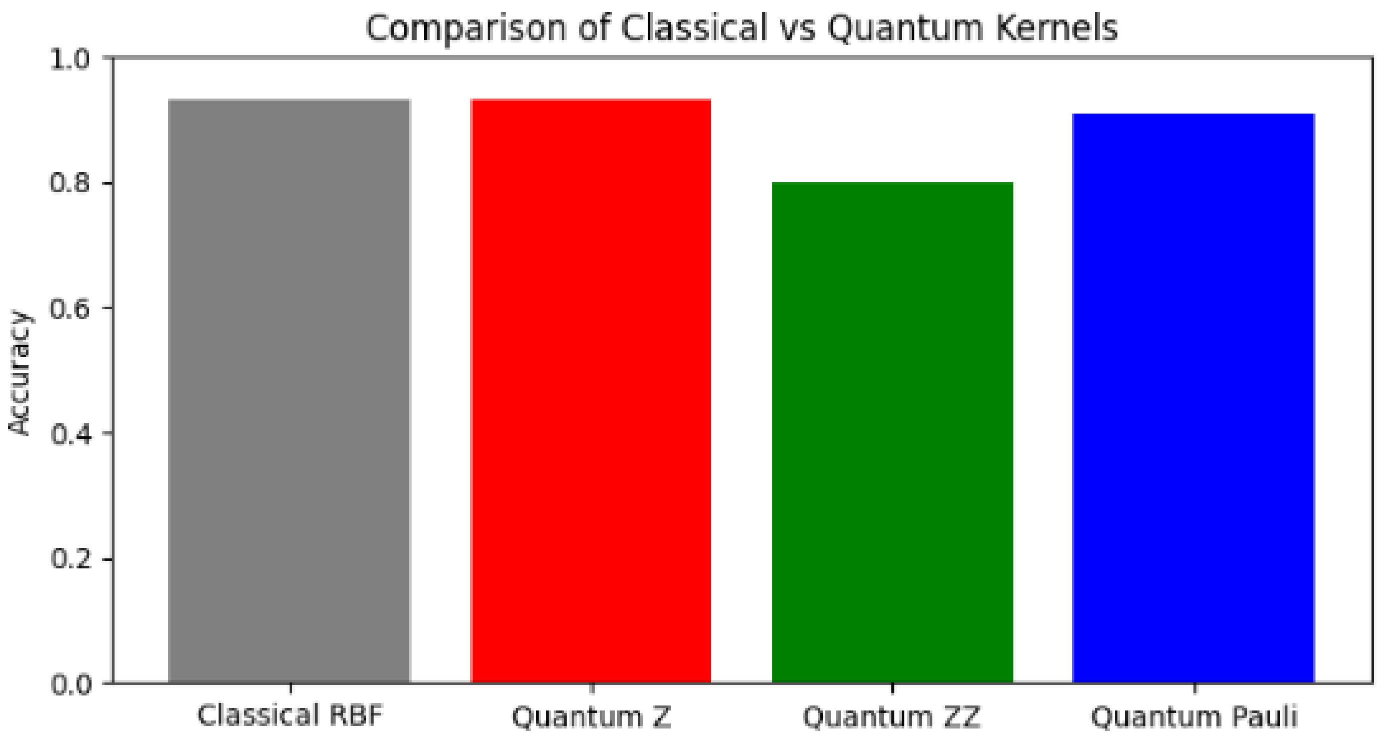
PauliFeatureMap QSVM Accuracy: 0.91

PauliFeatureMap



PauliFeatureMap QSVM Accuracy: 0.91

Results Comparison




```
ClassicalFeatureMap: 0.93
ZFeatureMap: 0.93
ZZFeatureMap: 0.80
PauliFeatureMap: 0.91
```

Summary

- Use All Four Features
- Increase Feature Map Depth
- Normalize Inputs
- Avoid Overly Complex Grid Search
- Adding Regularization to the Kernel