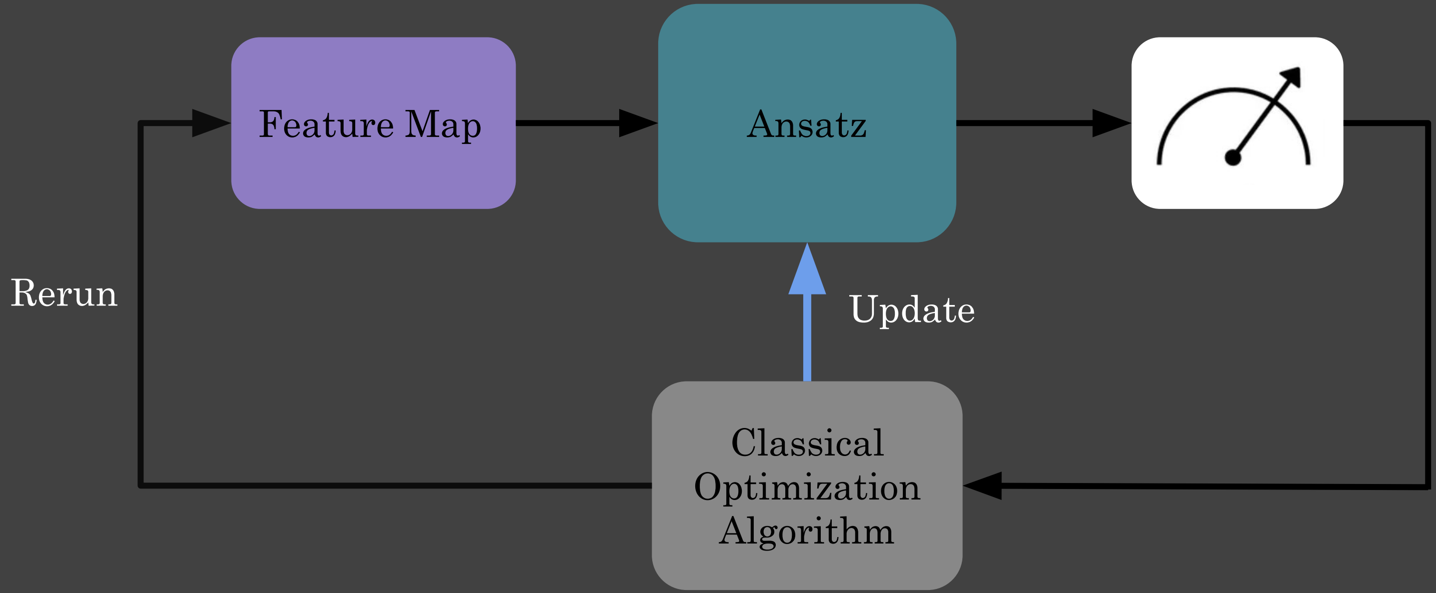
Variational Quantum Classifier

Quantum Computing

Samuel Morrás
Cristina Álvarez
Pablo Fernandez

2. Method

VQC Circuit



2.1 Encoding. Feature Map

Classical data into Quantum

- Amplitude Encoding
  - Computationally complex and generally not efficient

$$|\psi\rangle = \sum_{i=0}^{2^n-1} x_i |i\rangle$$

- Basis Encoding
  - Not scalable

$$x_i \to |x_i\rangle, \quad x \in \{0, 1, \ldots, 2^n - 1\}$$

- $\boxed{\text{Angle Encoding}}$
  - Simplicity
  - Scalability
  - Interpretation
  - Entanglement

$$U_{\Phi(\vec{x})} = \exp\left(i \sum_{S \in \mathcal{I}} \Phi(\vec{x}) \prod_{i \in S} P_i\right)$$

Pauli Feature Mapping

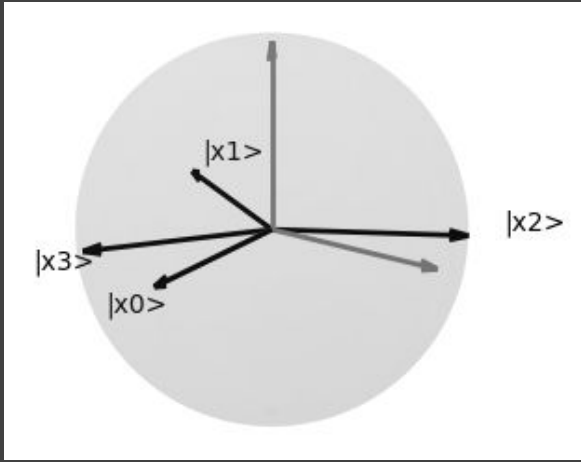
2.1 Encoding. Feature Map

Angle Encoding

ZZ Feature Map (Z Feature Map)

$$U_{\Phi(\vec{x})} = \exp\left(i\Phi(\vec{x})\prod_{i\in S} Z\right)$$

$$\Phi(\vec{x}) = \begin{cases} x_i & \text{if } S = \{i\} \\ (\pi - x_i)(\pi - x_j) & \text{if } S = \{i, j\} \end{cases}$$



Implemented using Rz and CNOT

2.2 Ansatz

Parameterized quantum circuit

· The circuit transforms the initial qubit state into a final state

· Rotation and entangling gates

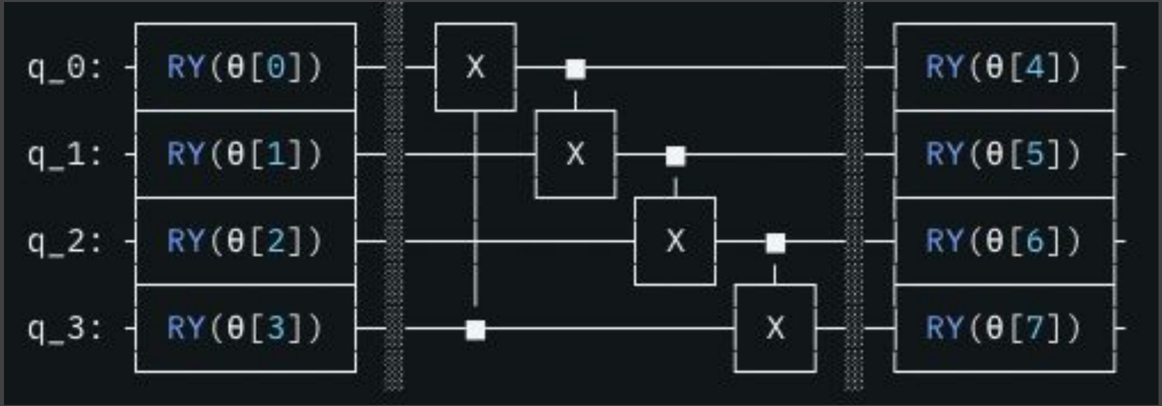
· This final state will be interpreted (measured) for the classification

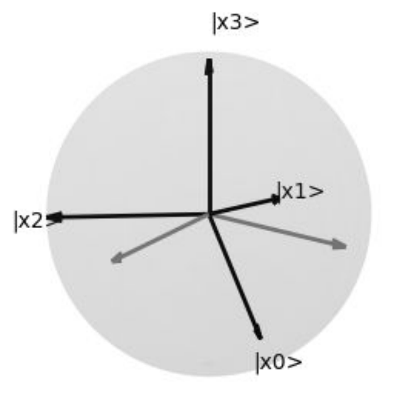
· Generic use:

- Two local: Rx, Ry, CNOT and/or CZ (Expressivity)
- Real Amplitudes: Ry and CNOT (Complexity efficiency)
- Pairwise: entanglement (Simplicity)

2.2 Ansatz

Real Amplitudes (1 layer)





· Ry angles are the parameters (optimization)

2.3 Measurement

Measurement: Interpretation of the final state

- Expected values of Pauli Matrices: X, Y, Z

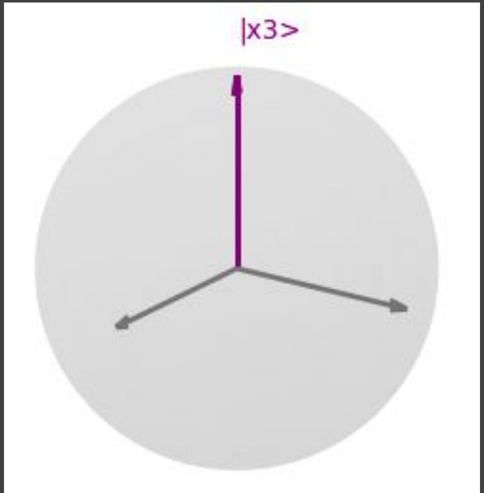
- Combinations (Hamiltonians)

- Interpretation

  - E.g. measure the third qubit:

    - $\langle Z \rangle = 1$ "Yes"

    - $\langle Z \rangle = -1$ "No"



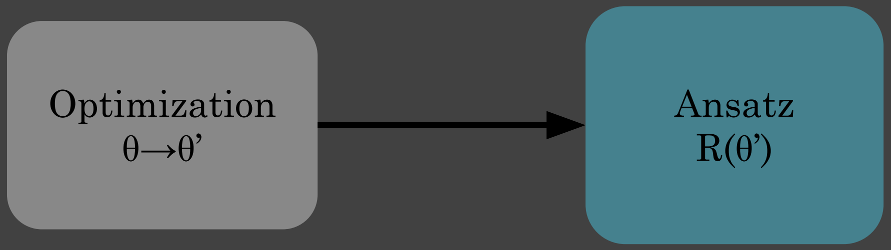
2.4 Classical Algorithm

Optimization.

· Classical algorithm tunes the angle parameters in the ansatz

· Most used ones:
  - COBYLA
  - ADAM
  - Conjugate Gradient

· How? Minimizing a loss function (related to the expected values)



3. Example: **Iris Dataset**

Versicolor

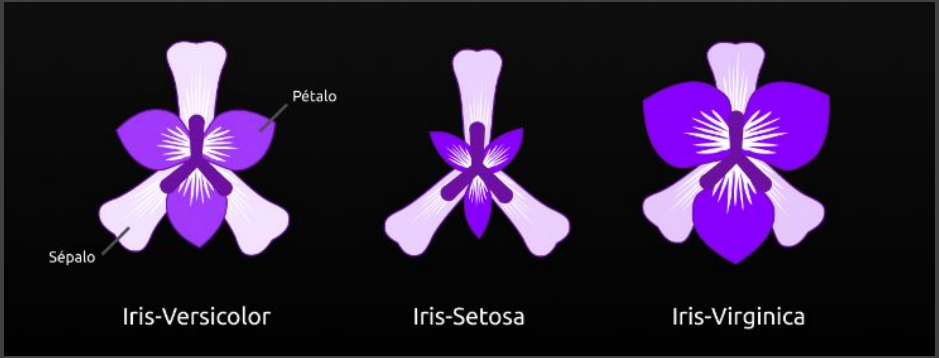
Setosa

Virginica

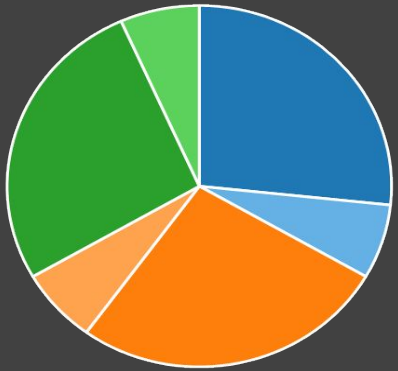
3.1 Raw data (I)

CLASSICAL: Raw Data → QUANTUM: Feature Map → QUANTUM: Variational Circuit → CLASSICAL: Optimization

3 Labels



150 Samples



4 Features

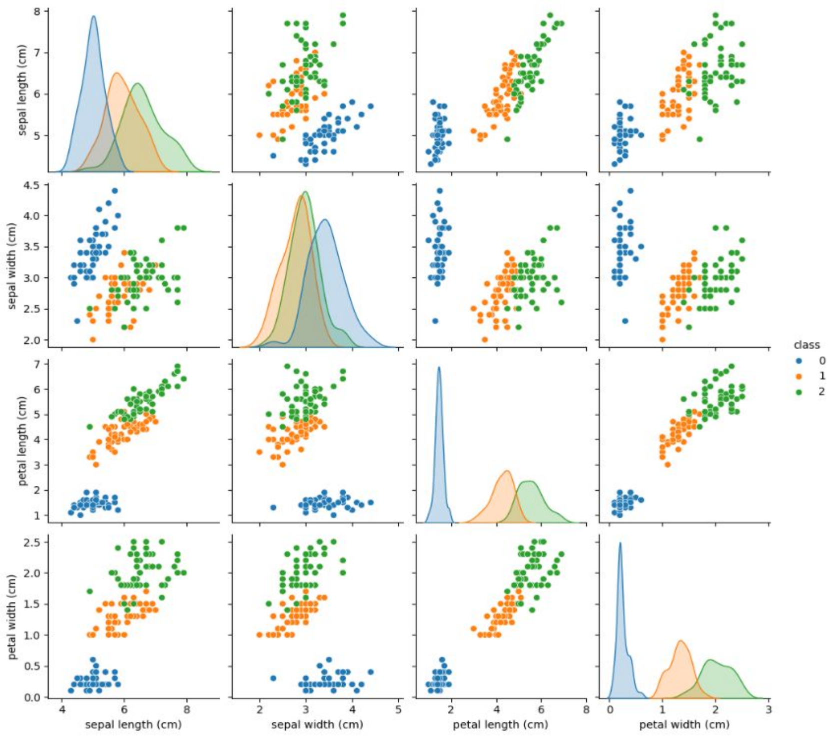
Feature	Min (cm)	Max (cm)	Mean (cm)	SD (cm)	Class Correlation
Sepal Length	4.3	7.9	5.84	0.83	**0.78**
Sepal Width	2.0	4.4	3.05	0.43	**-0.42**
Petal Length	1.0	6.9	3.76	1.76	**0.95**
Petal Width	0.1	2.5	1.20	0.76	**0.96**

3.1 Raw data (II)

CLASSICAL: Raw Data → QUANTUM: Feature Map → QUANTUM: Variational Circuit → CLASSICAL: Optimization

- Class 0 (Setosa) is linearly separable from the other two classes

- Normalization of the features: Stability and convergence.



For a Classical ML algorithm (SVC*):

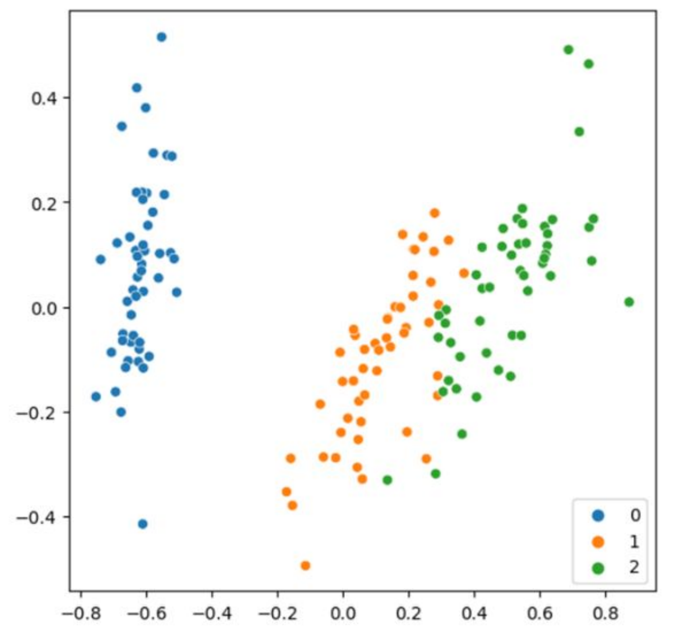
Algorithm	N° of features	Data subset	Score
SVC	4	Training	**0.99**
SVC	4	Test	**0.97**

Further information about SVC can be found in the appendix.

3.1 Raw data (III)

CLASSICAL: Raw Data | QUANTUM: Feature Map | QUANTUM: Variational Circuit | CLASSICAL: Optimization

- Reduction of features: From 4 to 2.
- PCA Transformation*



For a Classical ML algorithm (SVC):

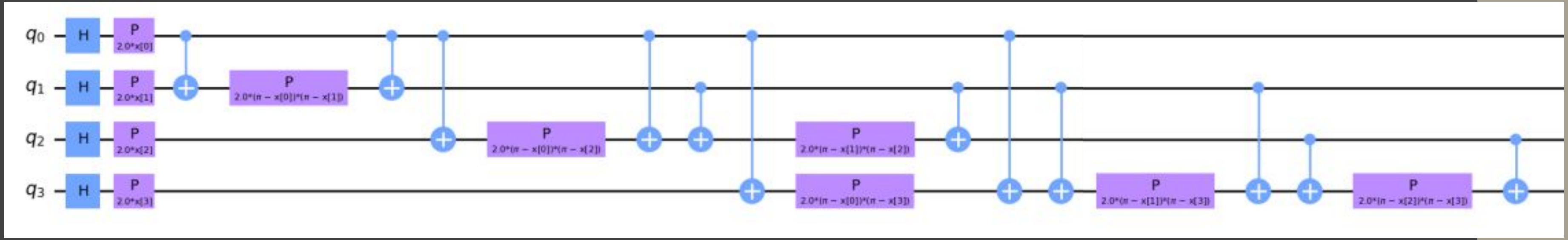
Algorithm	N° of features	Data subset	Score
SVC	2	Training	**0.97**
SVC	2	Test	**0.90**

**Further information about PCA can be found in the appendix.*

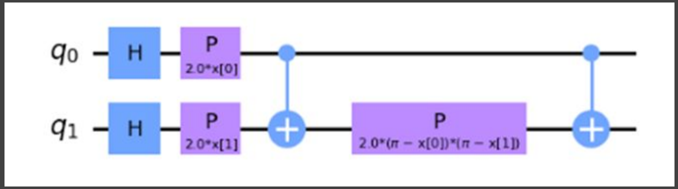
3.2 Feature Map

CLASSICAL: Raw Data → QUANTUM: Feature Map → QUANTUM: Variational Circuit → CLASSICAL: Optimization

4 Features



2 Features



ZZ Feature map:

- Encodes data mapping it into a Quantum Hilbert Space.
- Based on Rotational gates and Entangling operations.

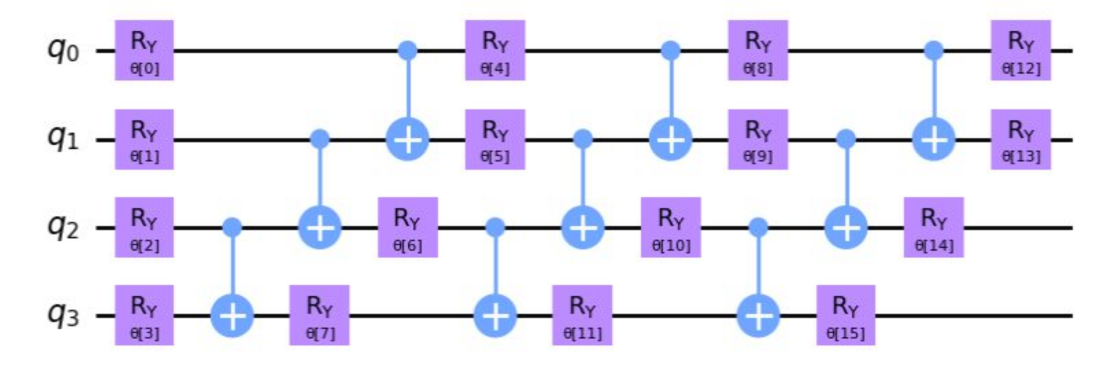
3.3 Ansatz (I)

CLASSICAL: Raw Data → QUANTUM: Feature Map → QUANTUM: Variational Circuit → CLASSICAL: Optimization

4 Features, Real Amplitude Ansatz

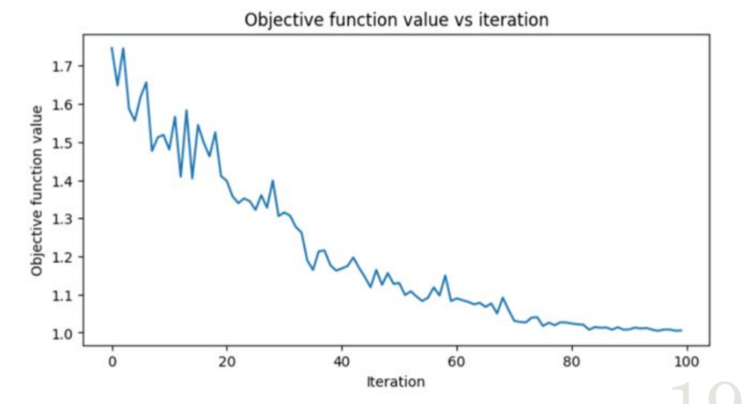
- 16 tunable parameters (weights) are optimized in order to minimize the *Objective function.

- 3 repeated layers



High number of parameters adds complexity and thus training the model takes more time.

- Training time: 655 seconds



Algorithm	N° of features	Ansatz	Data subset	Score
VQC	4	RA	Training	**0.85**
VQC	4	RA	Test	**0.87**

Further information about Objective function can be found in the appendix.

3.3 Ansatz (II)

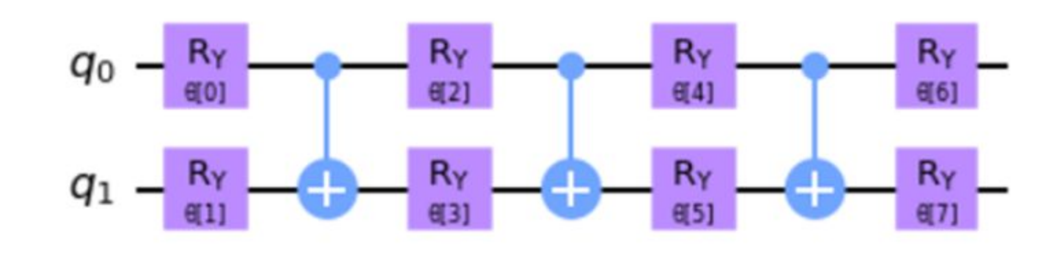
CLASSICAL: Raw Data → QUANTUM: Feature Map → QUANTUM: Variational Circuit → CLASSICAL: Optimization

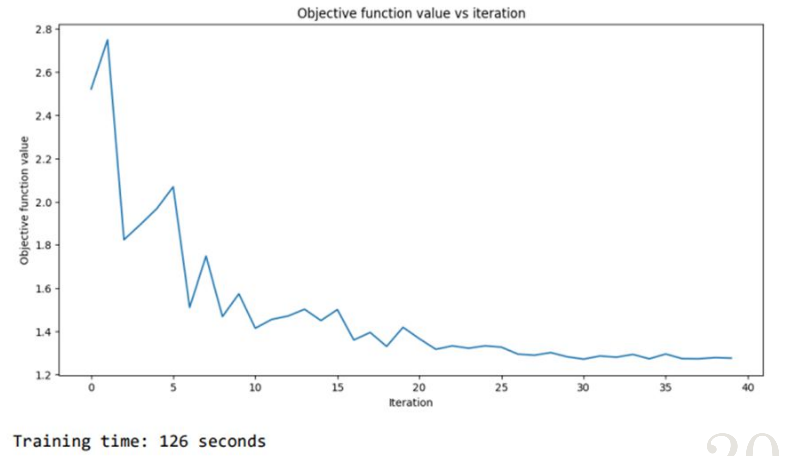
2 Features, Real Amplitude Ansatz

- 8 tunable parameters (weights) are optimized in order to minimize the *Objective function.

- 3 repeated layers

- Training time: 126 seconds





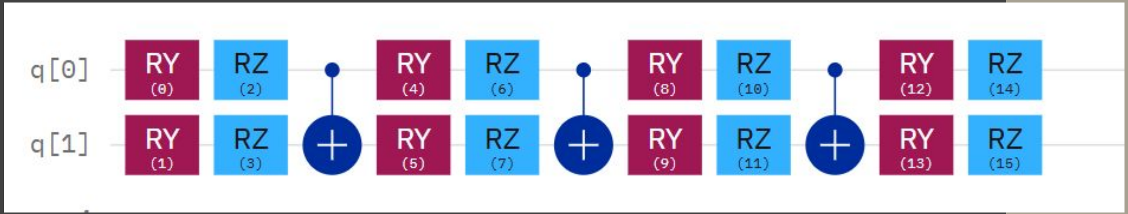
Algorithm	N° of features	Ansatz	Data subset	Score
VQC	2	RA	Training	0.58
VQC	2	RA	Test	0.63

3.3 Ansatz (III)

CLASSICAL: Raw Data → QUANTUM: Feature Map → QUANTUM: Variational Circuit → CLASSICAL: Optimization

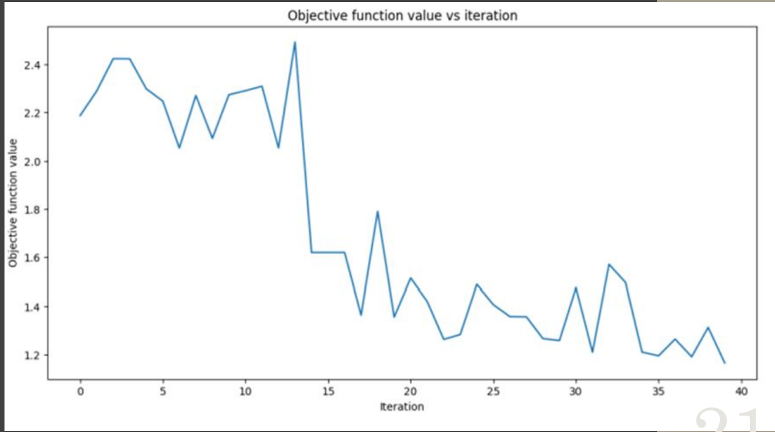
2 Features, EfficientSU2 Ansatz

- 16 tunable parameters (weights) are optimized in order to minimize the *Objective function.



- 3 repeated layers

- Training time: 161 seconds



Algorithm	N° of features	Ansatz	Data subset	Score
VQC	2	RA	Training	**0.71**
VQC	2	RA	Test	**0.67**

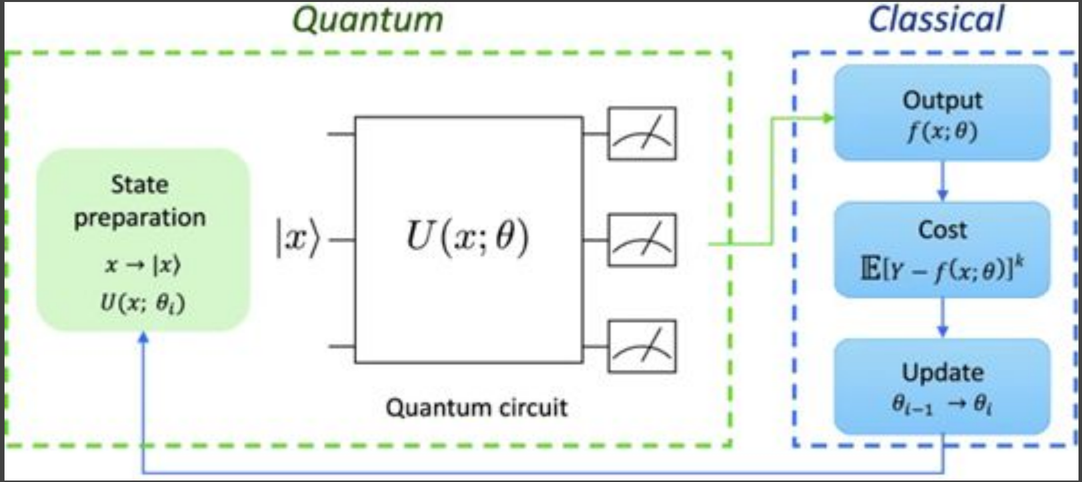
3.4 Optimization

CLASSICAL: Raw Data | QUANTUM: Feature Map | QUANTUM: Variational Circuit | CLASSICAL: Optimization

The weights of the ansatz are modified in order to to minimize the Objective function.

Unlike classical deep learning, which typically uses gradient-based methods like gradient descent or Adam, ***VQC often use Gradient-free optimizers*.***

Selection of parameters: Overfitting vs underfitting



**Further information about Gradient-free optimizer can be found in the appendix.*

3.5 Conclusion

CLASSICAL: Raw Data → QUANTUM: Feature Map → QUANTUM: Variational Circuit → CLASSICAL: Optimization

Algorithm	Data subset	Score	Data subset	Score
SVC - 4 features	Train	**0.99**	Test	**0.97**
VQC - 4 features (Real Amplitudes)	Train	**0.85**	Test	**0.87**
SVC - 2 features	Train	**0.97**	Test	**0.90**
VQC - 2 features (Real Amplitudes)	Train	**0.58**	Test	**0.63**
VQC - 2 features (EfficientSU2)	Train	**0.71**	Test	**0.67**

- **Classical vs. Quantum Models**: Classical models like SVC are mature and reliable, while quantum models need further development to achieve comparable performance.
- **Impact of Feature Reduction:** Reducing features from 4 to 2 led to slight performance drops in SVC but significant declines in VQC, particularly with the Real Amplitudes ansatz.
- **Critical Factors for VQC:** VQC performance is highly sensitive to feature count, ansatz configuration, and hyperparameters, emphasizing the importance of the selection of this parameters.

REFERENCES

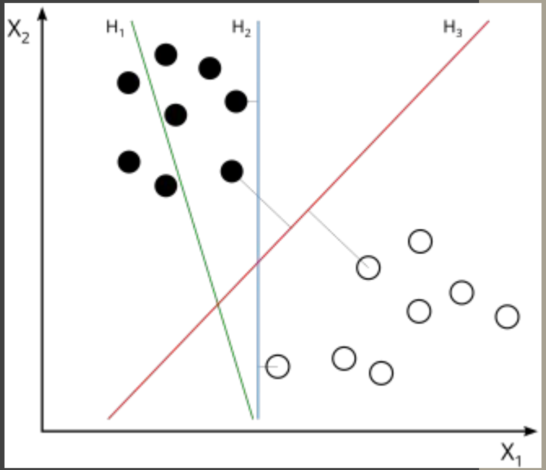
- Qikit dcumentation: https://qiskit.org/documentation
- Quantum Machine Learning Course: https://learn.qiskit.org/course/machine-learning
- Sikit-learn: https://scikit-learn.org/stable/index.html
- GitHub repository: https://github.com/qiboteam/qibo/tree/master/examples/variational_classifier
- Encoding featuremaps and ansatz: https://qiskit.org/documentation/apidoc/circuit_library.html#data-encoding-circuits

Appendix (I)

- Support Vector Machine (SVM):

Supervised max-margin models with associated learning algorithms that analyze data for classification and regression analysis.

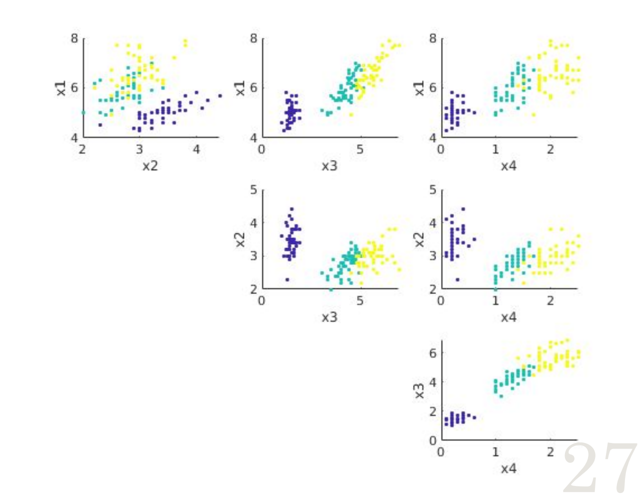
Figure 1: H1 does not separate the classes. H2 does, but only with a small margin. H3 separates them with the maximal margin.



- PCA

Technique for dimensionality reduction of the large data set. Reducing the number of features costs some accuracy but makes the large data set easier to explore and reduces the computational complexity of the model which makes ML algorithms run faster.

Figure 2: The 2D view of the iris dataset captures almost 97.76% of the variation of the original data points.



Appendix (II)

- Objective function:

The objective function is what we optimize, it's the function we aim to either minimize or maximize.

The cost function measures the model's error on the training data. While minimizing the cost is important, the model's ability to generalize is even more crucial. A model that only performs well on training data is not useful in real-world scenarios. To prevent overfitting, we add a regularization term that penalizes complexity, creating a new function that balances both error and complexity during training. In models without regularization, the objective function and the cost function are the same.

- Gradient-free

Quantum circuits can have noisy gradients or barren plateaus, where gradients are almost zero, making gradient-based optimization less effective. Gradient-free optimizers, like COBYLA or SPSA, search for the optimal parameters without directly computing gradients.