EURO²

Basics of Quantum Machine Learning

Jiří Tomčala

IT4Innovations,
VŠB - Technical University of Ostrava

21 Oct 2025

Part I

Introduction

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.
- It is a hybrid approach that enhances traditional machine learning models by incorporating quantum computing's strengths where applicable.

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.
- It is a hybrid approach that enhances traditional machine learning models by incorporating quantum computing's strengths where applicable.
- To encode data and adapt it to the architectures presented by quantum computers, it's necessary to perform a different data processing than usual.

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.
- It is a hybrid approach that enhances traditional machine learning models by incorporating quantum computing's strengths where applicable.
- To encode data and adapt it to the architectures presented by quantum computers, it's necessary to perform a different data processing than usual.
- This feature extraction is usually condensed into what is known as a feature map, which is nothing more than a quantum circuit that encodes data appropriately to benefit from the larger possibility space provided by qubits.

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.
- It is a hybrid approach that enhances traditional machine learning models by incorporating quantum computing's strengths where applicable.
- To encode data and adapt it to the architectures presented by quantum computers, it's necessary to perform a different data processing than usual.
- This feature extraction is usually condensed into what is known as a feature map, which is nothing more than a quantum circuit that encodes data appropriately to benefit from the larger possibility space provided by qubits.
- With this higher-dimensional space available, the data can be distributed in various ways to try to find common features among them or hidden trends not immediately visible.

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.
- It is a hybrid approach that enhances traditional machine learning models by incorporating quantum computing's strengths where applicable.
- To encode data and adapt it to the architectures presented by quantum computers, it's necessary to perform a different data processing than usual.
- This feature extraction is usually condensed into what is known as a feature map, which is nothing more than a quantum circuit that encodes data appropriately to benefit from the larger possibility space provided by qubits.
- With this higher-dimensional space available, the data can be distributed in various ways to try to find common features among them or hidden trends not immediately visible.
- The transition from a classical computational model to a quantum one entails replacing the existing framework with a generic quantum circuit.

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.
- It is a hybrid approach that enhances traditional machine learning models by incorporating quantum computing's strengths where applicable.
- To encode data and adapt it to the architectures presented by quantum computers, it's necessary to perform a different data processing than usual.
- This feature extraction is usually condensed into what is known as a feature map, which is nothing more than a quantum circuit that encodes data appropriately to benefit from the larger possibility space provided by qubits.
- With this higher-dimensional space available, the data can be distributed in various ways to try to find common features among them or hidden trends not immediately visible.
- The transition from a classical computational model to a quantum one entails replacing the existing framework with a generic quantum circuit.
- The new model incorporates quantum gates that are parameterized by angles, providing the degrees of freedom necessary for the model to explore potential solutions to a given problem.

Introduction

- **Quantum Machine Learning (QML)** is not a purely quantum process, because the stages of algorithms in which quantum computations can be incorporated and which relate to some general problem are limited.
- It is a hybrid approach that enhances traditional machine learning models by incorporating quantum computing's strengths where applicable.
- To encode data and adapt it to the architectures presented by quantum computers, it's necessary to perform a different data processing than usual.
- This feature extraction is usually condensed into what is known as a feature map, which is nothing more than a quantum circuit that encodes data appropriately to benefit from the larger possibility space provided by qubits.
- With this higher-dimensional space available, the data can be distributed in various ways to try to find common features among them or hidden trends not immediately visible.
- The transition from a classical computational model to a quantum one entails replacing the existing framework with a generic quantum circuit.
- The new model incorporates quantum gates that are parameterized by angles, providing the degrees of freedom necessary for the model to explore potential solutions to a given problem.
- In the quest for the optimal solution, the model iteratively adjusts the angles of the quantum gates within the circuit.

The active links in this presentation contain Jupyter Notebook scripts that can be run without any installation directly online in the Google Colaboratory environment.

Just click on them and open in a new window.

Part II

Data encoding

Data encoding
Basis encoding

Basis encoding encodes a classical n-bit string into a computational basis state of a n-qubit system.

This method renders each classical bit as a separate qubit and directly maps the binary representation of data onto quantum states in the computational basis.

It means that, for example, the number 237 (11101101_2) would be encoded into the quantum state using this encoding method as: $|11101101\rangle$.

This can be easily achieved by using X gates on the appropriate qubits. See an example here:

`Basis_encoding.ipynb ... https://1url.cz/GJUrb`

As can be seen, this method of encoding is very inefficient, as it essentially degrades the quantum state of qubits to ordinary classical bits.

If multiple features need to be encoded, each feature is assigned to a different group of qubits. Each feature thus has its own group of qubits.

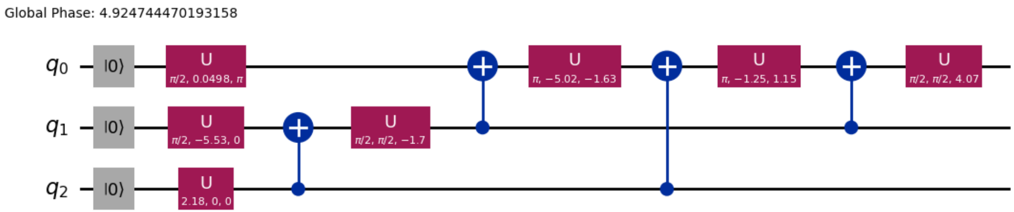
Data encoding

Amplitude encoding

Amplitude encoding encodes data into the amplitudes of a quantum state. Since the sum of the probabilities of the quantum state must be equal to 1, it is first necessary to normalize the encoded data. It means that, for example, the data vector 2, 3, 7, 0, 11, 9, 5, 1 would be encoded into the quantum state using this encoding method as:

$$\frac{1}{\sqrt{2^2+3^2+7^2+11^2+9^2+5^2+1^2}}\left(2\left|000\right\rangle+3\left|001\right\rangle+7\left|010\right\rangle+11\left|100\right\rangle+9\left|101\right\rangle+5\left|110\right\rangle+\left|111\right\rangle\right)$$

which can be realized by this quantum circuit:



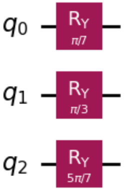
See the implementation here:

`Amplitude_encoding.ipynb ... https://1url.cz/kJUul`

Data encoding
Angle encoding

In this method, encoding is performed by rotating the angle θ of the quantum state of the qubit. Therefore, the data must be rescaled to the interval $(0, 2\pi]$.

The realization of this encoding of data values $1/7, 1/3, 5/7$ by a quantum circuit would then look like this:



As can be seen, each value requires one whole separate qubit, which is again a very inefficient way of encoding. However, for some smaller applications, this method may be suitable because it offers clearly separated input data, which can subsequently make training the model easier.

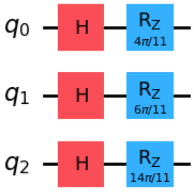
See the implementation here:

`Angle_encoding.ipynb ... https://1url.cz/kJUo0`

Data encoding
Phase encoding

Phase encoding is very similar to the previous angle encoding, but with the difference that this time the data is encoded into rotation angles ϕ of the quantum states of the qubits. Before that, however, it is necessary to put the qubits into a uniformly superposed state $|+\rangle$ using Hadamard gates. And here too, it is necessary to rescale the data to the $(0, 2\pi]$ interval in advance.

The realization of this encoding of data values $4/11, 6/11, 14/11$ by a quantum circuit would then look like this:



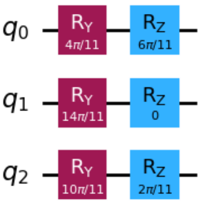
See the implementation here:

`Phase_encoding.ipynb ... https://1url.cz/bJUHb`

Data encoding
Dense angle encoding

Dense angle encoding is a combination of angle encoding and phase encoding. So it allows to encode two data values into one qubit.

The realization of this encoding of data values $4/11, 14/11, 10/11, 6/11, 0, 2/11$ by a quantum circuit would then look like this:



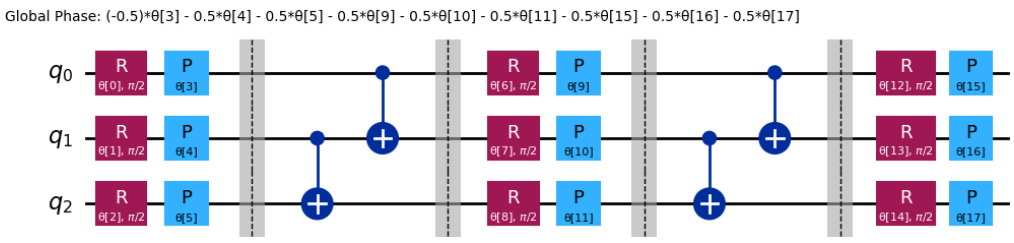
See the implementation here:

`Dense_angle_encoding.ipynb ... https://1url.cz/tJU2T`

Data encoding

Encoding with Qiskit's built-in feature maps

Built-in encodings combine the above types of encoding with one or more entanglement layers. The application of entanglement layers enables using the same qubits again for encoding additional data. This can be clearly seen from the following figure, which shows a quantum circuit for Efficient SU2 encoding of 18 input data values (denoted as θ) using only 3 qubits.



Qiskit contains many built-in encodings, or mappings, for which one can set the number of layers, the type of entanglement and the gates used. These options can be tested in the script linked below.

See and play with the implementations of encodings
Efficient SU2, Z feature map, ZZ feature map and Pauli feature map here:

`Built-in_feature_maps_encoding.ipynb ... https://1url.cz/sJU2P`

Part III

Quantum Classification and Regression

Quantum Classification and Regression

Types of Machine Learning and Algorithms

- **Supervised:** the data that is used to train the model is labeled. The goal of these algorithms is to learn the relationship between data and their corresponding labels or outputs and to generalize this to unseen data. Common tasks in this class are classification and regression.

Quantum Classification and Regression

Types of Machine Learning and Algorithms

- **Supervised:** the data that is used to train the model is labeled. The goal of these algorithms is to learn the relationship between data and their corresponding labels or outputs and to generalize this to unseen data. Common tasks in this class are classification and regression.
    - Classification: machine learning tasks where categories or labels are assigned to input data based on specific characteristics.

Quantum Classification and Regression

Types of Machine Learning and Algorithms

- **Supervised:** the data that is used to train the model is labeled. The goal of these algorithms is to learn the relationship between data and their corresponding labels or outputs and to generalize this to unseen data. Common tasks in this class are classification and regression.
  - Classification: machine learning tasks where categories or labels are assigned to input data based on specific characteristics.
  - Regression: it is a machine learning technique used to predict continuous numerical values from input data.

Quantum Classification and Regression

Types of Machine Learning and Algorithms

- **Supervised:** the data that is used to train the model is labeled. The goal of these algorithms is to learn the relationship between data and their corresponding labels or outputs and to generalize this to unseen data. Common tasks in this class are classification and regression.
  - <u>Classification:</u> machine learning tasks where categories or labels are assigned to input data based on specific characteristics.
  - <u>Regression:</u> it is a machine learning technique used to predict continuous numerical values from input data.
- **Unsupervised:** uses unlabeled data to train the machine learning model. The goal of such algorithms is to discover hidden patterns and structure in data. Some algorithms in this class are clustering and dimensionality reduction algorithms. Some generative models such as generative adversarial networks and variational autoencoders can also be considered in this category.

Quantum Classification and Regression

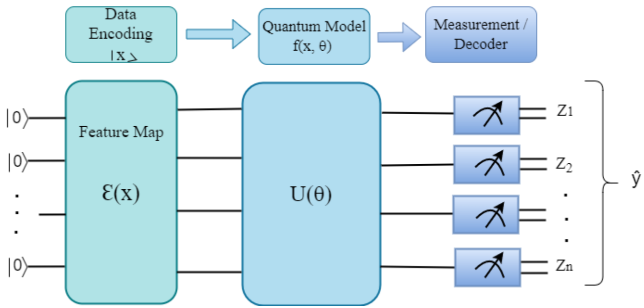
Types of Machine Learning and Algorithms

- **Supervised:** the data that is used to train the model is labeled. The goal of these algorithms is to learn the relationship between data and their corresponding labels or outputs and to generalize this to unseen data. Common tasks in this class are classification and regression.
  - Classification: machine learning tasks where categories or labels are assigned to input data based on specific characteristics.
  - Regression: it is a machine learning technique used to predict continuous numerical values from input data.
- **Unsupervised:** uses unlabeled data to train the machine learning model. The goal of such algorithms is to discover hidden patterns and structure in data. Some algorithms in this class are clustering and dimensionality reduction algorithms. Some generative models such as generative adversarial networks and variational autoencoders can also be considered in this category.
- **Reinforced:** algorithms are defined by an agent which interacts with an environment. The agent takes actions and receives feedback from its environment in the form of rewards and punishments. Eventually through this feedback mechanism, the agent learns to take the correct set of actions to perform a specific task.

Quantum Classification and Regression

Quantum Models

In the past, there was always a need to find models that could handle problems that classical computers had difficulty handling. With the advent of quantum computers, it is possible to assemble some of them. There are many candidates for quantum models. It depends on the nature of the problem that needs to be solved. Quantum models use the principles of quantum mechanics to potentially offer advantages in processing speed and handling complex, high-dimensional data. However, it is necessary to understand their structure not only for the correct selection of a model, but also for its efficient implementation.



Source: Ullah et al. 10.1109/ACCESS.2024.3353461.

Quantum Classification and Regression

Quantum Training

To train a quantum model, it is necessary, similarly to the classical model, to define a so-called **error function**. The quadratic value of the immediate error of the trained model is often used for this function. The value of such a function then increases in all directions from the correct solution, making it easier to find it through an iterative process.

The error function is then the basis for constructing the so-called **gradient function**, which can be used to determine the derivative of the error function and thus optimize the tuning of the model parameters θ. The most widely used is the gradient descent function.

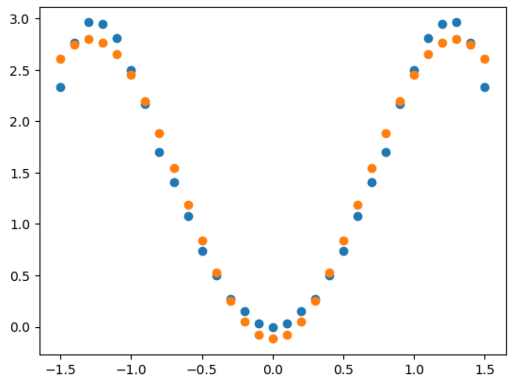
The **gradient descent function** consists of iteratively adjusting the model parameters in the opposite direction to the gradient of the error function, with the aim of minimizing the error function and improving the accuracy of the model. Of course, any other classical optimizer could also be used here instead of a gradient descent function.

A frequently used method is also the so-called **parameter-shift rule**, which is a simpler way to take the derivative of a quantum circuit. It states that the derivative of a circuit with respect to a certain parameter θ_i can be calculated with two runs of that circuit with that particular parameter shifted.

Quantum Classification and Regression
Hands-on

By adjusting the error function to a specific case, classification or regression problems can be solved in the same way. Since quantum classification will be used in the hands-on of the next chapter, here in this hands-on will be demonstrated quantum regression.

`Quantum_Regression.ipynb ... https://1url.cz/8JUEb`



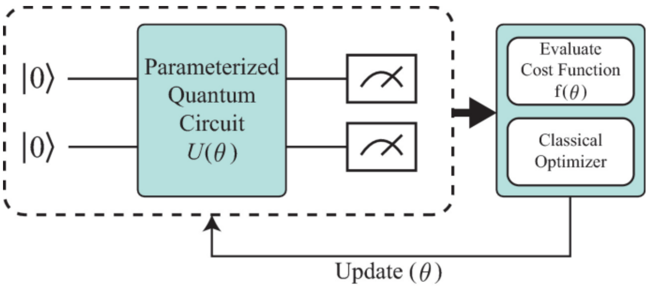
Part IV

Parameterized Quantum Circuit

Parameterized Quantum Circuit

Parameterized Quantum Circuits (PQCs) blend quantum and classical techniques, using adjustable gates for optimization. PQCs are the backbone of variational quantum algorithms. These algorithms use quantum and classical computing to solve complex problems. They are perfect for today's noisy quantum computers, offering a path to quantum advantage in the near future.

PQC acts as a fixed structure of parameterized gates, analogous to a classical neural network layer. PQC can therefore be included in **Variational Quantum Circuit (VQC)**, which uses a PQC as a component to solve an optimization problem, where the PQC's parameters are varied by a classical optimizer to minimize a cost (error) function.



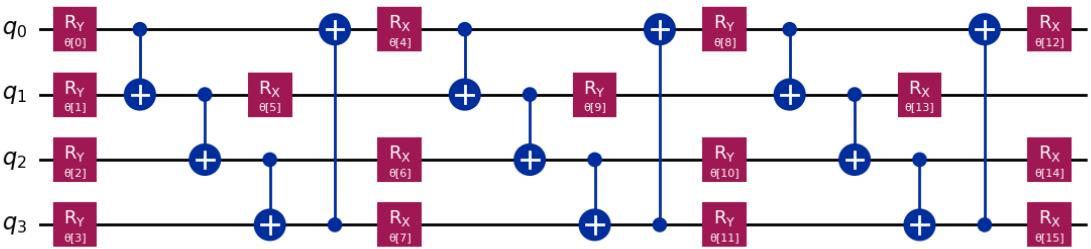
Source: Rizvi et al. 10.1007/978-3-031-47359-3_15.

Parameterized Quantum Circuit

Sometimes in the literature, PQCs (or VQCs) are referred to as **Quantum Neural Networks (QNNs)**, although it is not exactly the same thing. PQCs need not follow the general structure of a neural network. For example, not all data need to be loaded in the first (input) layer. Some data can be loaded in the first layer, apply some gates, and then can be loaded additional data (a process called *data reuploading*. Therefore, QNNs should be considered a subset of PQCs.

To make matters even more confusing, PQC is often referred to by the German term **ansatz**, which means something like a qualified or educated guess. This term comes from physics and mathematics, where it refers to guessing a solution. Here, in the field of QML, this means a PQC that implements the dependence of its individual parameters θ on the desired model output as accurately as possible, compared to the modeled reality.

Example of ansatz for image recognition from the hands-on part below:



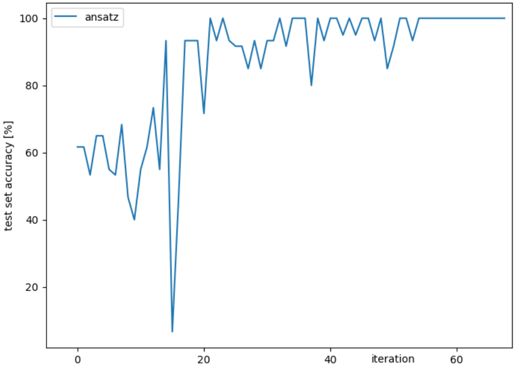
Parameterized Quantum Circuit
Hands-on

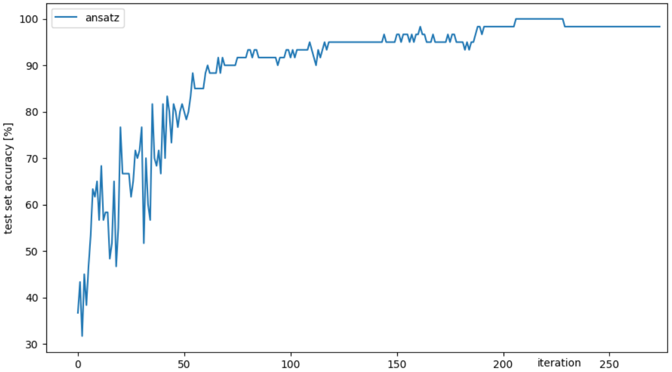
Example of using an 8-qubit QNN for quantum recognition of a 4x2 pixel image:

`QNN_4x2_8q.ipynb ... https://1url.cz/aJUR7`

The same image recognition use case, but using only a 4-qubit QNN:

`QNN_4x2_4q.ipynb ... https://1url.cz/yJUaL`





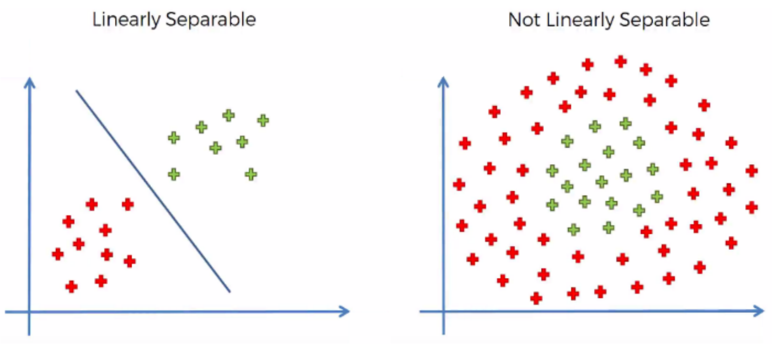
Part V

Quantum Support Vector Machine

Quantum Support Vector Machine

The **Quantum Support Vector Machine (QSVM)** proposes a hybrid method capable of exploiting the power of quantum computers, that do not necessarily contain a large number of qubits, for classification tasks. The basic idea is the same as that of a classical **Support Vector Machine (SVM)**, which is essentially a linear binary classifier that searches for a hyperplane whose distance from 2 different sets of variables is maximal.

However, the problem with classic SVM occurs when the 2 sets of variables are not linearly separable. Of course, one can always find some most optimal solution, but in this case of linear non-separability, 100% accuracy would never be achieved with such an SVM classifier.

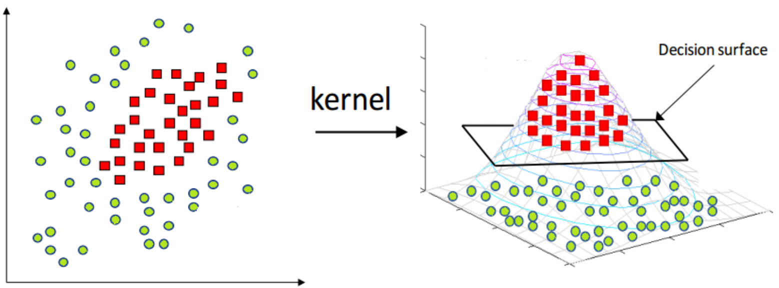


Quantum Support Vector Machine

Kernel Function

The problem of linear non-separability can be solved by introducing the **kernel function**. This function artificially increases the dimensionality of the input data so that it is linearly separable in the new space.

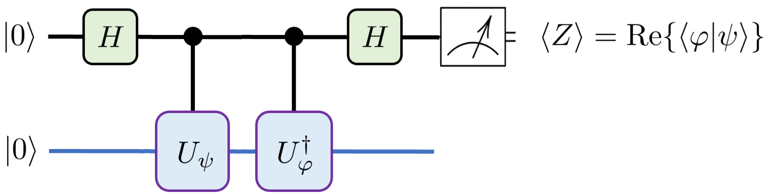
However, in the case of a more complex kernel function, this calculation can be quite demanding, and therefore **kernel matrix** is compiled in advance, which contains the values of the inner product corresponding to the given kernel function only for all pairs of input data. This approach is sometimes called the **kernel trick**.



Quantum Support Vector Machine
Quantum Kernels

The idea behind QSVM is to use quantum computers as kernel functions, which means that data will be mapped to the underlying Hilbert space on these computers. This new mapping can be done by applying a mapping gate U to the initial state, which is state $|0\rangle$ by default.

A quantum circuit can be used not only to map input data into quantum states (using feature mapping), but also to pre-calculate from these mapped states all inner products of the kernel matrix. This can be done by implementing the **Hadamard test** and measuring its output in the Z basis for all pairs of input data.



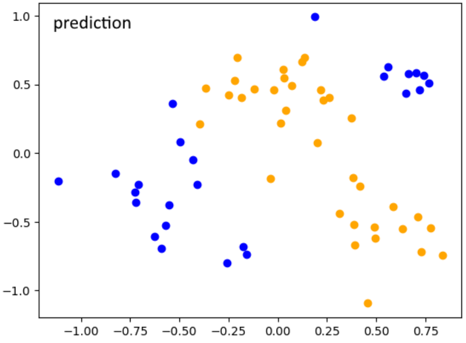
Quantum Support Vector Machine

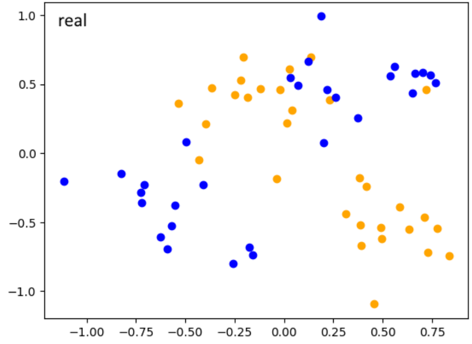
Hands-on

The use case below contains a simple concrete example of using a quantum computer in the QSVM algorithm.

It includes mapping the input data into quantum states, pre-calculating the kernel matrix using a quantum circuit with a Hadamard test, and then running a classical SVM, which finally performs the classification of the input data.

`QSVM.ipynb ... https://1url.cz/oJUkC`





Part VI

Quantum Unsupervised Learning

Quantum Unsupervised Learning

Unsupervised learning is a machine learning approach in which its algorithms look for patterns and structures in data without prior labeling or external supervision. It is most commonly used in tasks such as clustering or dimensionality reduction. These types of tasks include, in particular, customer segmentation, fraud detection, inventory optimization, and natural language processing.

Although supervised learning algorithms are usually more accurate than unsupervised learning models, they need human input and knowledge beforehand to label the data correctly. Despite the many advantages of unsupervised learning, certain problems can arise when machine learning models are allowed to operate without any human intervention.

<u>Problems with unsupervised learning:</u>

Quantum Unsupervised Learning

Unsupervised learning is a machine learning approach in which its algorithms look for patterns and structures in data without prior labeling or external supervision. It is most commonly used in tasks such as clustering or dimensionality reduction. These types of tasks include, in particular, customer segmentation, fraud detection, inventory optimization, and natural language processing.

Although supervised learning algorithms are usually more accurate than unsupervised learning models, they need human input and knowledge beforehand to label the data correctly. Despite the many advantages of unsupervised learning, certain problems can arise when machine learning models are allowed to operate without any human intervention.

Problems with unsupervised learning:

- More computationally intensive, as more training data is needed.

Quantum Unsupervised Learning

Unsupervised learning is a machine learning approach in which its algorithms look for patterns and structures in data without prior labeling or external supervision. It is most commonly used in tasks such as clustering or dimensionality reduction. These types of tasks include, in particular, customer segmentation, fraud detection, inventory optimization, and natural language processing.

Although supervised learning algorithms are usually more accurate than unsupervised learning models, they need human input and knowledge beforehand to label the data correctly. Despite the many advantages of unsupervised learning, certain problems can arise when machine learning models are allowed to operate without any human intervention.

Problems with unsupervised learning:

- More computationally intensive, as more training data is needed.
- Model training time is significantly longer.

Quantum Unsupervised Learning

Unsupervised learning is a machine learning approach in which its algorithms look for patterns and structures in data without prior labeling or external supervision. It is most commonly used in tasks such as clustering or dimensionality reduction. These types of tasks include, in particular, customer segmentation, fraud detection, inventory optimization, and natural language processing.

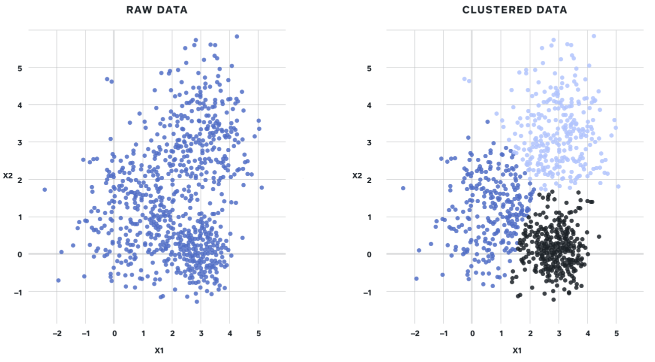
Although supervised learning algorithms are usually more accurate than unsupervised learning models, they need human input and knowledge beforehand to label the data correctly. Despite the many advantages of unsupervised learning, certain problems can arise when machine learning models are allowed to operate without any human intervention.

Problems with unsupervised learning:

- More computationally intensive, as more training data is needed.
- Model training time is significantly longer.
- The results are more probable to be inaccurate than in supervised learning.

In machine learning, data clustering is one of the most commonly used tools in **exploratory data analysis (EDA)**. It identifies data segmentation and provides insight into its structure.

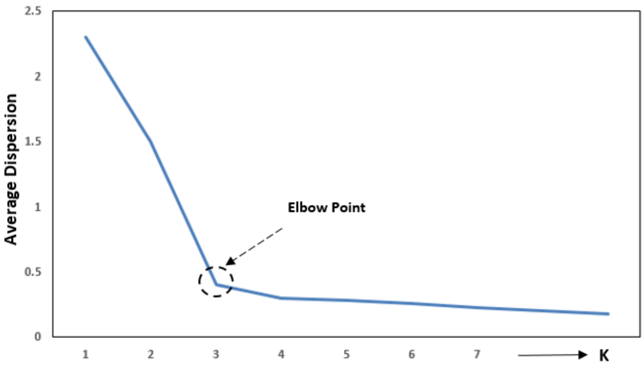
This unsupervised method is designed to find groups (clusters) in data in a way that all data within clusters share similar characteristics, while data in other clusters have characteristics that are different. The **K-means algorithm** allows these characteristics to be distinguished by taking the mean of the values within a group (cluster). The goal of K-means is simple: to group similar observations to uncover patterns that may not be apparent at first glance.



ELBOW RULE

First, it is necessary to determine in advance the value of $\boldsymbol{K}$, which is the number of clusters into which the data will later be classified. The **elbow rule** is used for this purpose. Its principle is clear from the picture below.

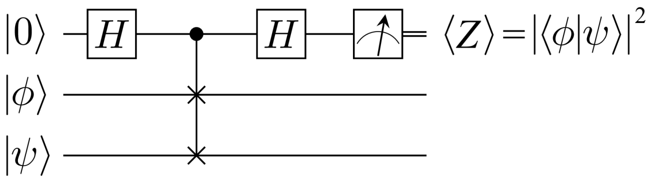
If the average dispersion within individual clusters no longer decreases with increasing number of clusters K, then the break-point indicates the optimal value of the number of clusters K.



Distance Between Quantum States and the Swap Test + Hands-on

By combining the K-means algorithm and the concept of distance (or proximity) between data points in Hilbert space, we can create a quantum version of the algorithm. All that is needed is to find a way to use the simplest possible quantum circuit to determine the similarity between two quantum states.

Although there are various techniques, the most widely used is the **swap test**. As can be seen from the figure below, if the measurement is performed in the Z basis, the result corresponds to the square of the inner product between the input quantum states ϕ and ψ. This value indicates the degree of proximity between ϕ and ψ. The maximum value is, of course, 1 (if they are the same) and the minimum value is 0 (if they are completely opposite), which represents the greatest possible distance between data points.



`Swap_test.ipynb ... https://1url.cz/aJUCF`

Thanks

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 101101903. The JU receives support from the Digital Europe Programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Türkiye, Republic of North Macedonia, Iceland, Montenegro, Serbia