

## 1. Introducción

### 1.1. Motivación del proyecto

### 1.2. Objetivos del proyecto

### 1.3. Cronograma

### 1.4. Estructura del documento

## 2. Materiales y métodos

### 2.1. Fundamentos de la computación cuántica: superposición, entrelazamiento e interferencia

### 2.2 Computación basada en circuitos y adiabática

### 2.3 Quantum Machine Learning: (Enfoques cuánticos puros)

### 2.4 Enfoques híbridos : VQC ...

### 2.5 Herramientas utilizadas: Qiskit, TensorFlow y Keras

## 3. Clasificador cuántico implementado

### 3.1. Técnicas de codificación de datos en sistemas cuánticos

### 3.2. Selección de puertas cuánticas

### 3.3. Definición de la función de pérdida para el entrenamiento

### 3.2. Descripción de las bases de datos empleadas

### 3.3. Proceso de entrenamiento y validación del modelo

## 4. Experimentación y Análisis de Resultados

### 4.1. Entrenamiento del modelo cuántico

### 4.2. Métricas de evaluación: precisión, eficiencia, recall

### 4.3. Comparativa con algoritmos clásicos equivalentes

### 4.4. Discusión de los resultados obtenidos

## 5. Conclusiones y trabajos futuros

QKE (Quantum Kernel Estimator) emplea un feature map fijo para codificar datos, calcula un kernel cuántico (inner product) y luego usa un SVM clásico para entrenar el modelo

sí tiene **parámetros entrenables** en el circuito y requiere un bucle variacional: el circuito se ejecuta, se mide una función de coste, un optimizador clásico actualiza los parámetros, y así iterativamente hasta converger.