Qiskit

Qiskit Advocate Mentorship Program, Spring'23

CHECKPOINT-2

Project *: Implement Date-reuploading classifier in Qiskit Machine Learning #3*

Mentor: **Atsushi Matsuo**

Mentee: **Shivani Rajput & Eraraya Ricardo Muten**

- In addition to the previously mentioned updates, we have conducted an analysis of the available options for implementing a data reuploading class. This analysis involved evaluating the pros and cons of each implementation approach. By carefully considering the different options, we aims to select the most suitable method that aligns with the project's objectives and requirements. This analysis provides valuable insights and helps in making informed decisions regarding the design and implementation of the data reuploading class.

- Furthermore, we have worked on generating a dataset using the sklearn library. To assist in this process, we are creating a tutorial specifically for the data reuploading class. This tutorial will serve as a comprehensive guide, outlining the necessary steps and techniques involved . By providing a tutorial, we aims to facilitate the understanding and adoption of the data reuploading class by other members. This will ensure that the process of generating the dataset can be replicated efficiently, allowing for wider contribution to the project's progress.

- Additionally, as part of the project's ongoing evaluation and analysis, we are actively comparing the performance of the data reuploading approach with the RealAmplitude ansatz using various types of data. This comparative testing aims to provide insights into the strengths and weaknesses of each method in different scenarios. The results of these tests will guide the team's decision-making process, helping them determine the most appropriate approach for different problem domains.

- To summarize, alongside the previously mentioned updates, the our team has conducted an analysis of the available implementation options for the data reuploading class, considering the pros and cons of each approach. They are also in the process of creating a tutorial to guide the generation of datasets using the sklearn library. Furthermore, the we will continues to compare the performance of the data reuploading approach with the RealAmplitude ansatz using different datasets. These tasks contribute to the overall progress of the project, enhancing its capabilities and refining the methods employed for optimal results.

Qiskit

Comparing Performances ⟶

Ours[Data Reuploading Class (ansatz)]

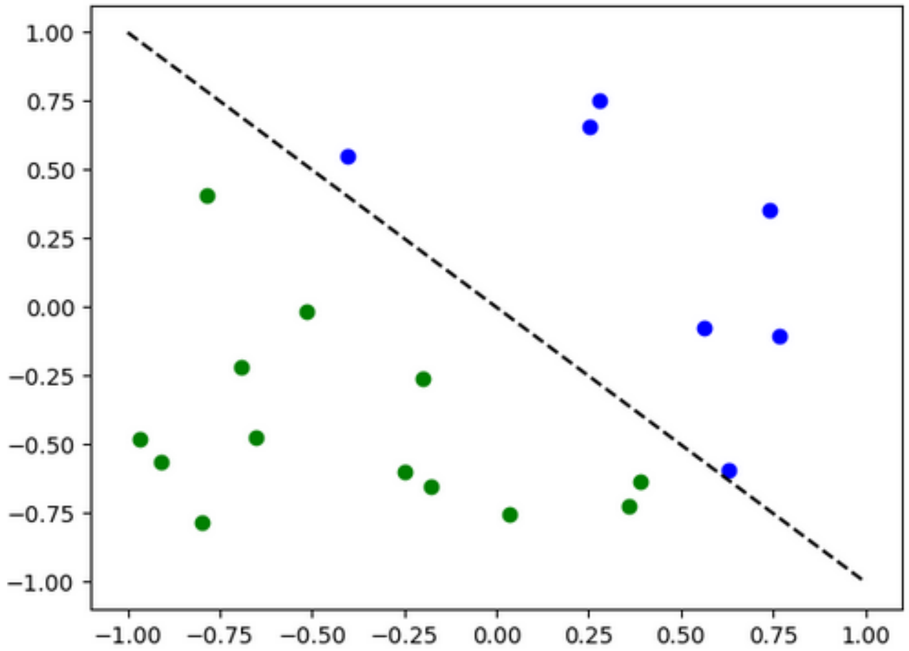
Qiskit

```
# Evaluate model and compute accuracy
y_predict = []
for x, y_target in zip(X, y):
    output = model1(Tensor(x))
    y_predict += [np.sign(output.detach().numpy())[0]]

print("Accuracy:", sum(y_predict == y) / len(y))

# Plot results
# red == wrongly classified
for x, y_target, y_p in zip(X, y, y_predict):
    if y_target == 1:
        plt.plot(x[0], x[1], "bo")
    else:
        plt.plot(x[0], x[1], "go")
    if y_target != y_p:
        plt.scatter(x[0], x[1], s=200, facecolors="none", edgecolors="r", linewidths=2)
plt.plot([-1, 1], [1, -1], "--", color="black")
plt.show()
```

```
ccuracy: 1.0
```



Qiskit documentation using the RealAmplitude ansatz

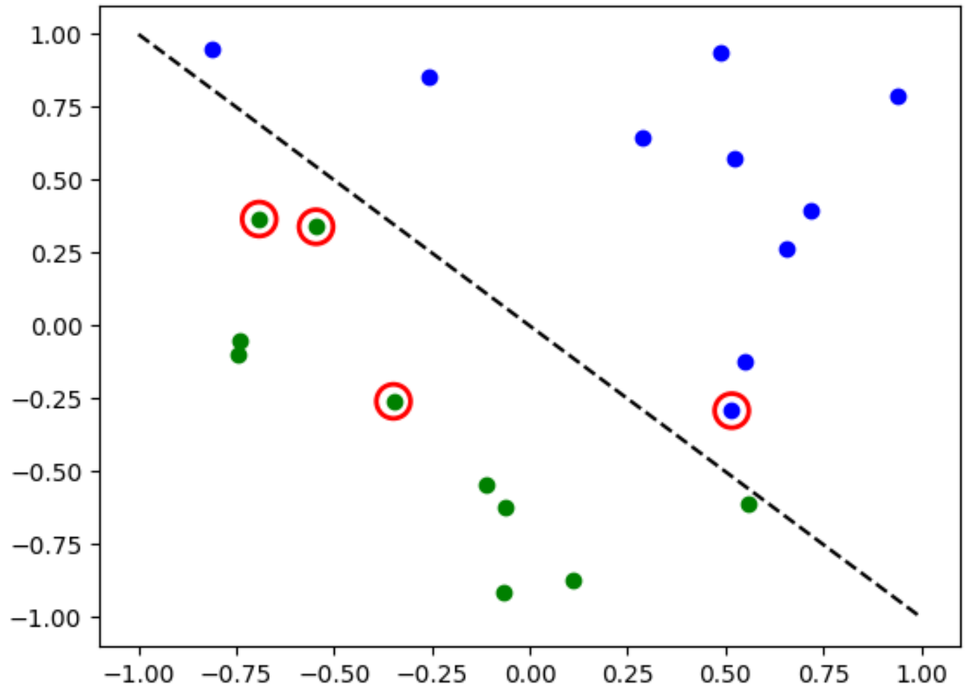
Qiskit

[7]:
```
# Evaluate model and compute accuracy
y_predict = []
for x, y_target in zip(X, y):
    output = model1(Tensor(x))
    y_predict += [np.sign(output.detach().numpy())[0]]

print("Accuracy:", sum(y_predict == y) / len(y))

# Plot results
# red == wrongly classified
for x, y_target, y_p in zip(X, y, y_predict):
    if y_target == 1:
        plt.plot(x[0], x[1], "bo")
    else:
        plt.plot(x[0], x[1], "go")
    if y_target != y_p:
        plt.scatter(x[0], x[1], s=200, facecolors="none", edgecolors="r", linewidths=2)
plt.plot([-1, 1], [1, -1], "--", color="black")
plt.show()
```

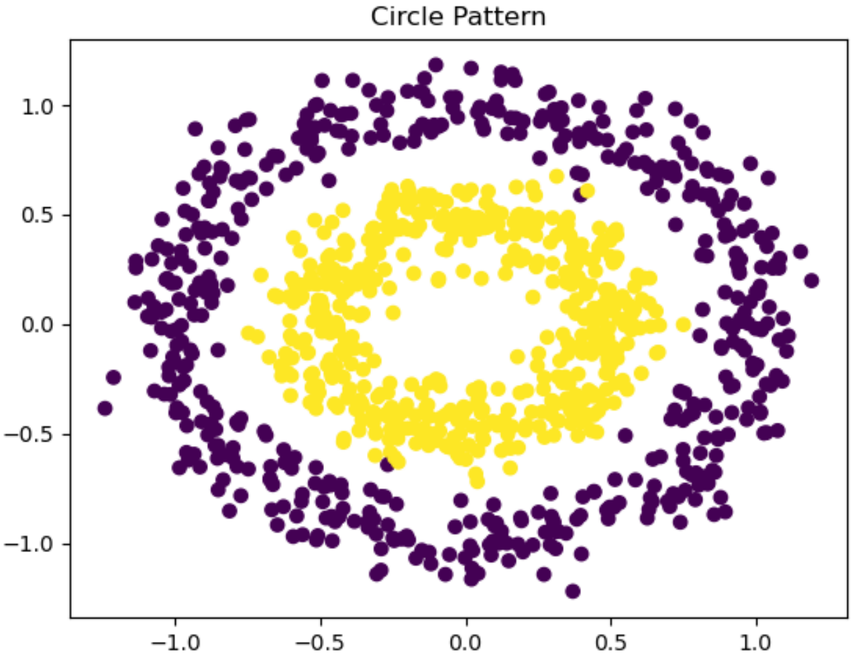
```
Accuracy: 0.8
```

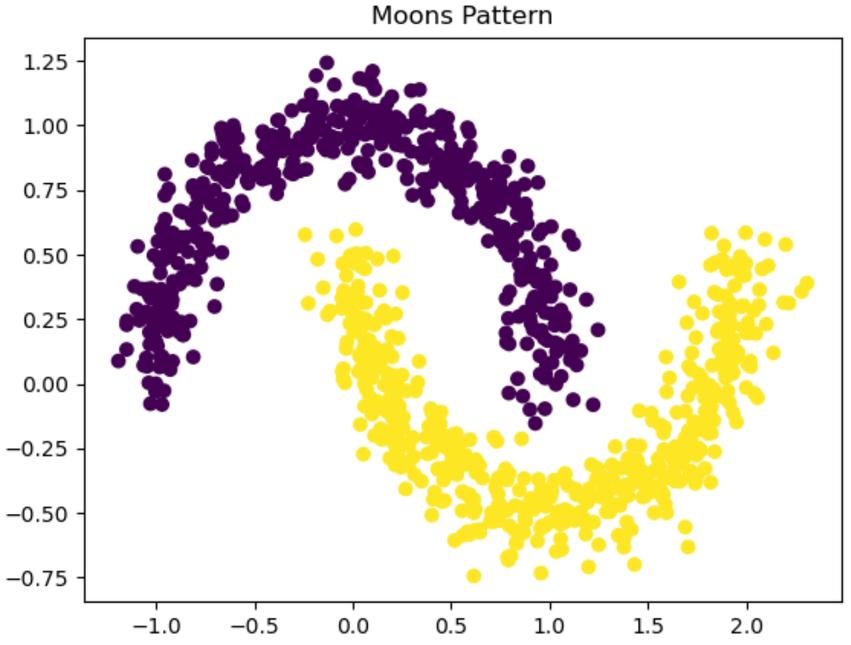


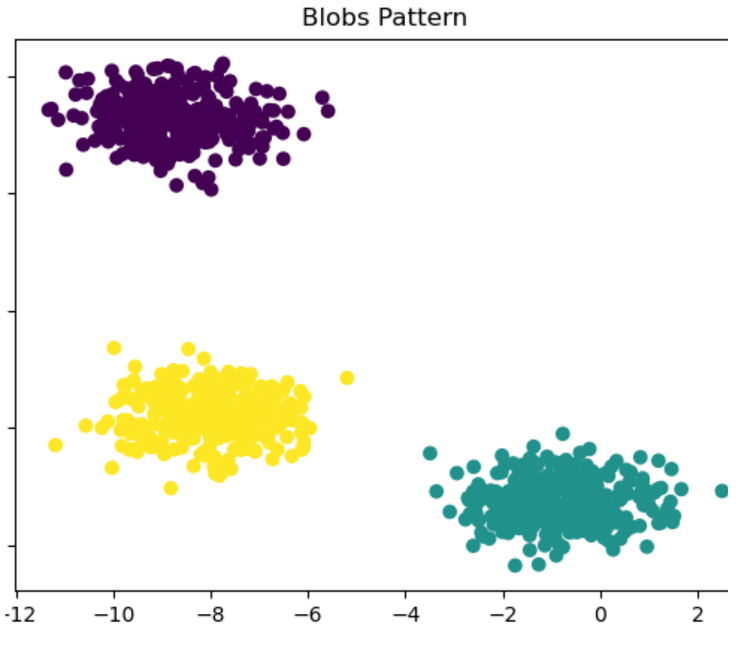
Qiskit

Datasets Generation →

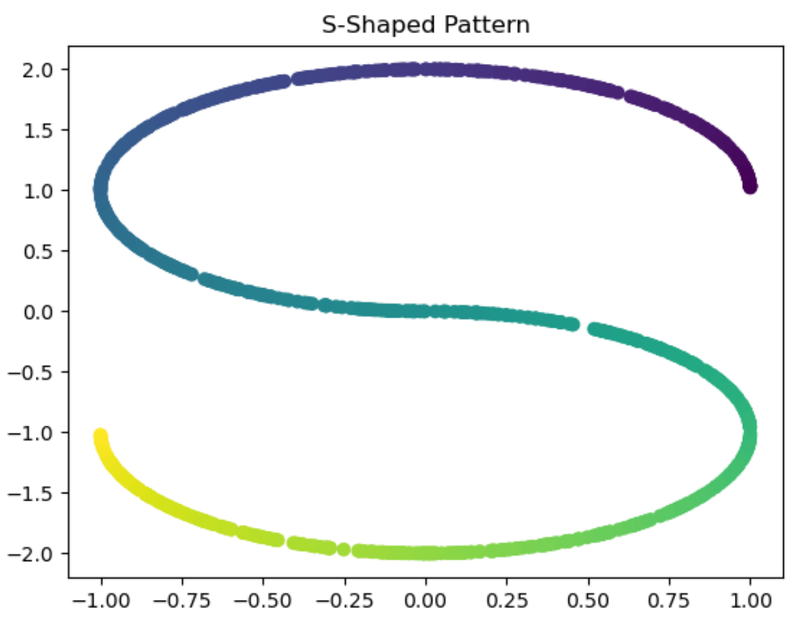
Qiskit

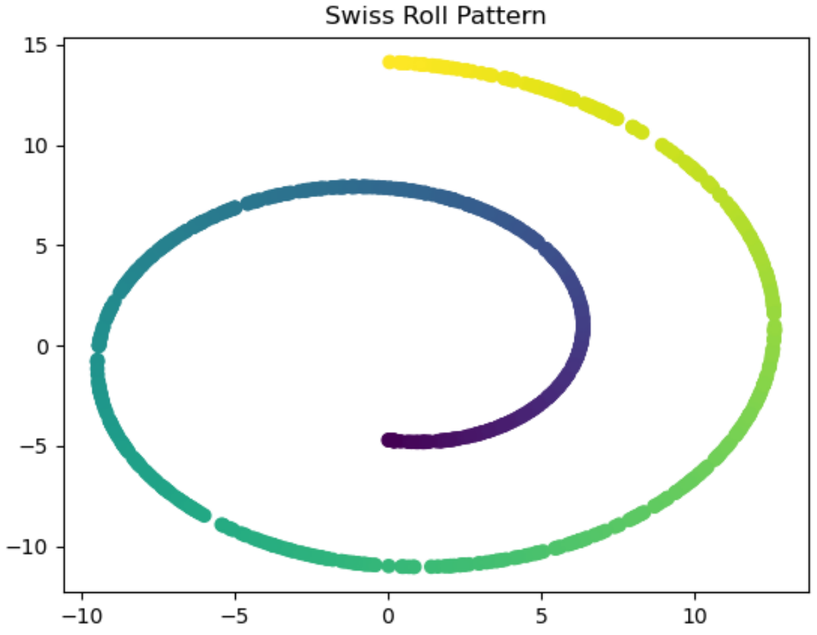

Circle Pattern
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

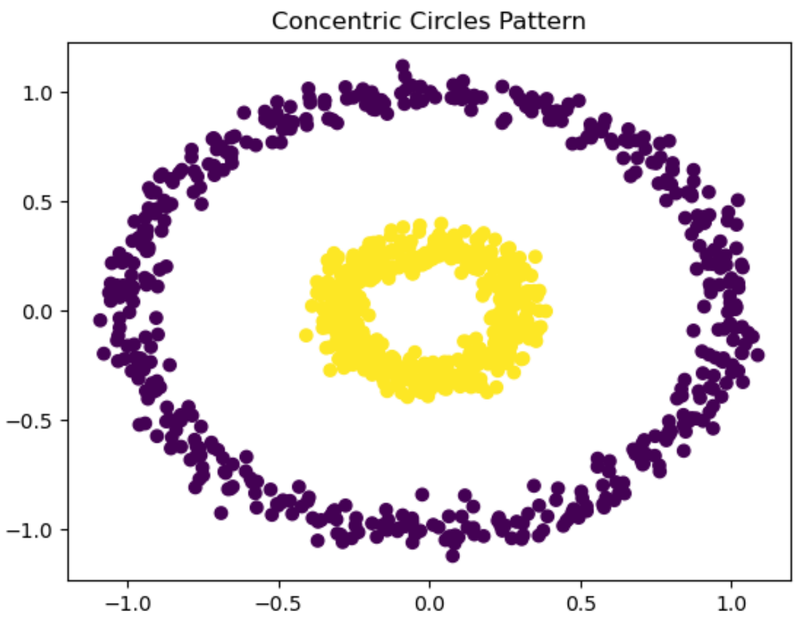

Moons Pattern
1.25
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


Blobs Pattern
-12
−10
−8
−6
−4
−2
0
2

Qiskit


S-Shaped Pattern
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


Swiss Roll Pattern
15
10
5
0
−5
−10
−10
−5
0
5
10

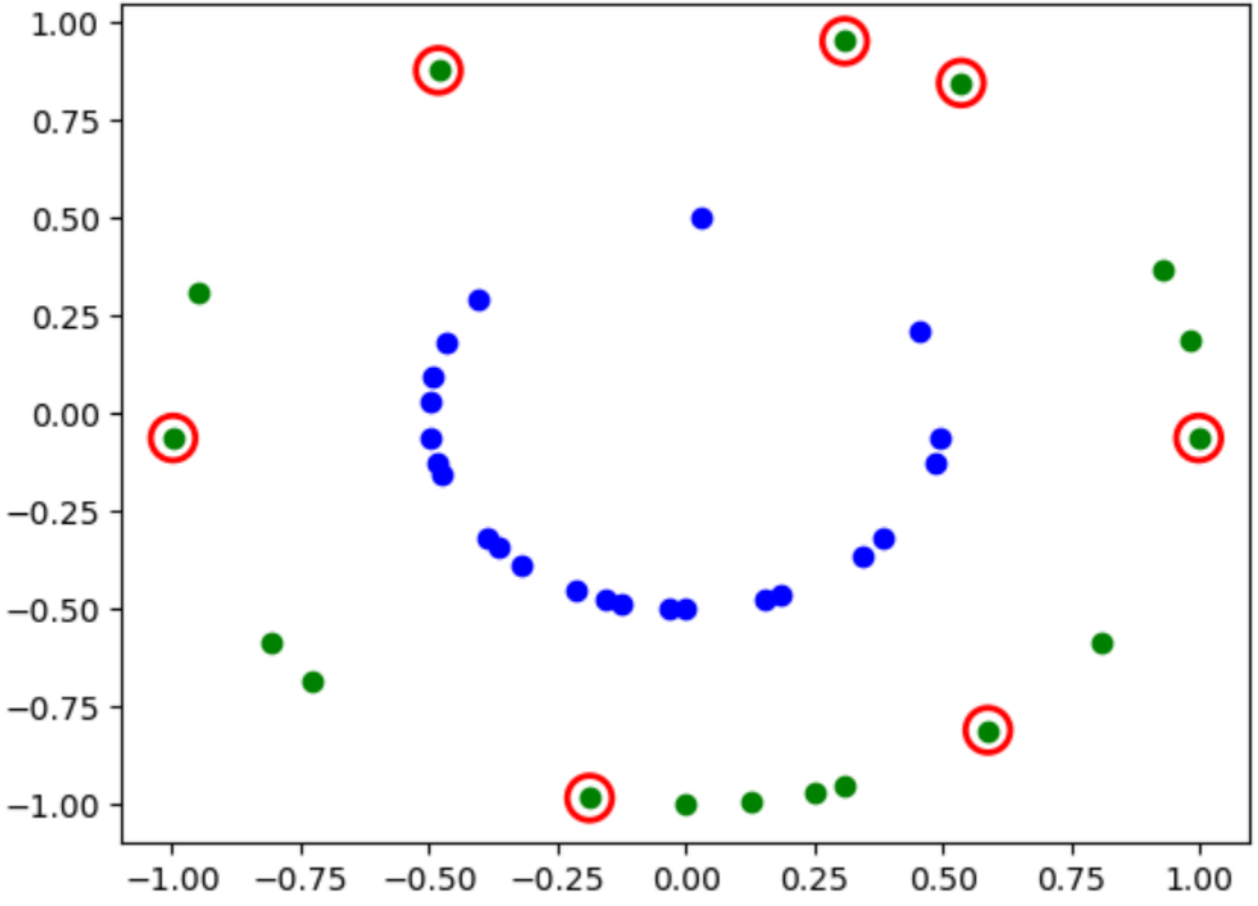

Concentric Circles Pattern
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

Qiskit

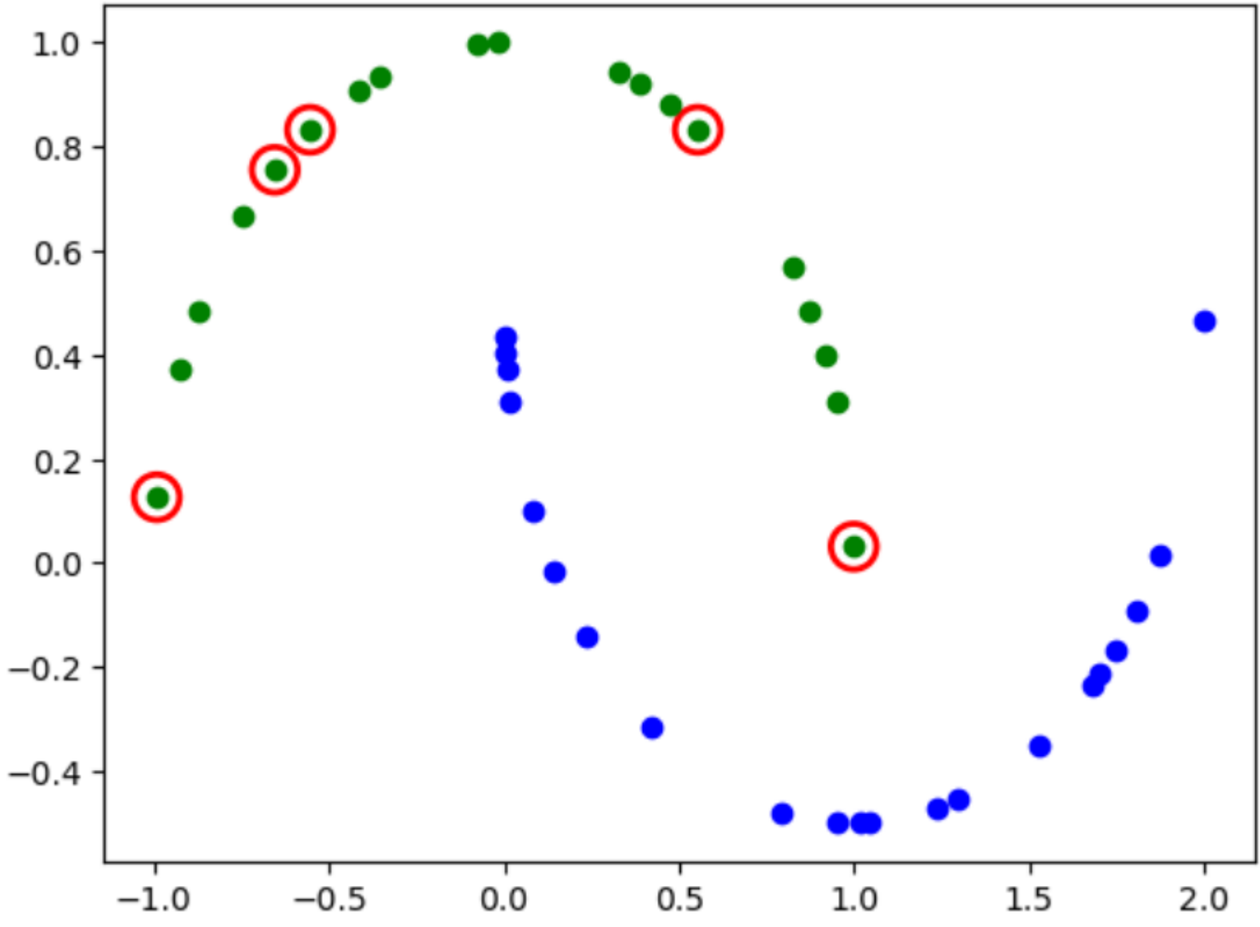
Testing model's accuracy using our generated datasets ⟶

Qiskit

```
Accuracy: 0.825
```




```
Accuracy: 0.875
```




Thank
You