

Fraud detection with Quantum Machine Learning

Gianmarco Sarnelli

Giovanni Varutti

Edoardo Zappia

Stime di mercato

Dimensione del mercato globale del
rilevamento e prevenzione delle frodi 2022

Grand View Research: 25.67
miliardi di dollari

previsione di crescita annuale composta: 17.6%
dal 2023 al 2030 [1]

Global Market Insights: 30
miliardi di dollari

previsione di crescita annuale composta: 17.6%
dal 2023 al 2030 [2]

[1] Grand View Research GVG:

<https://www.grandviewresearch.com/industry-analysis/fraud-detection-prevention-market#:~:text=The%20global%20fraud%20detection%20%26%20prevention.USD%2028.98%20billion%20in%202023.>

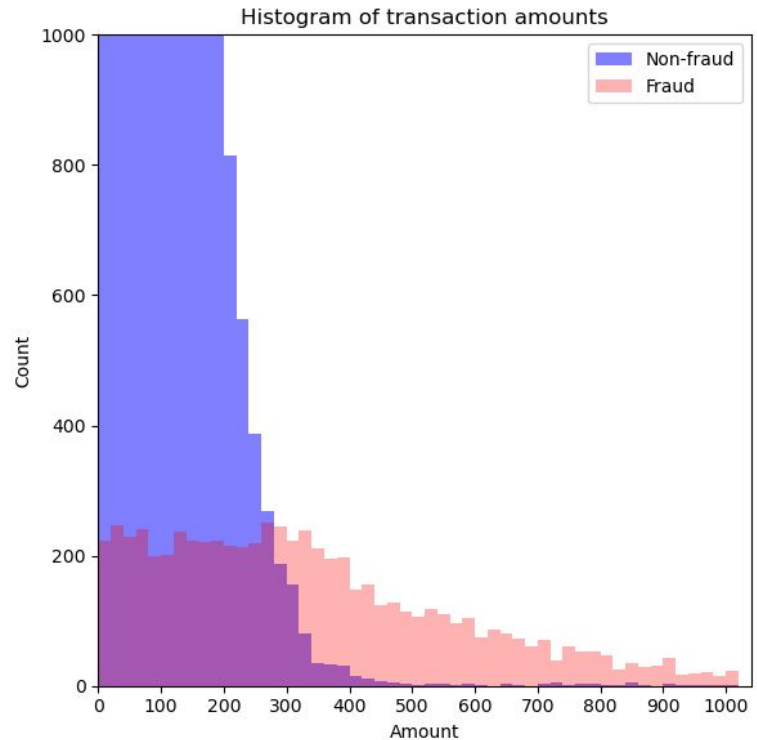
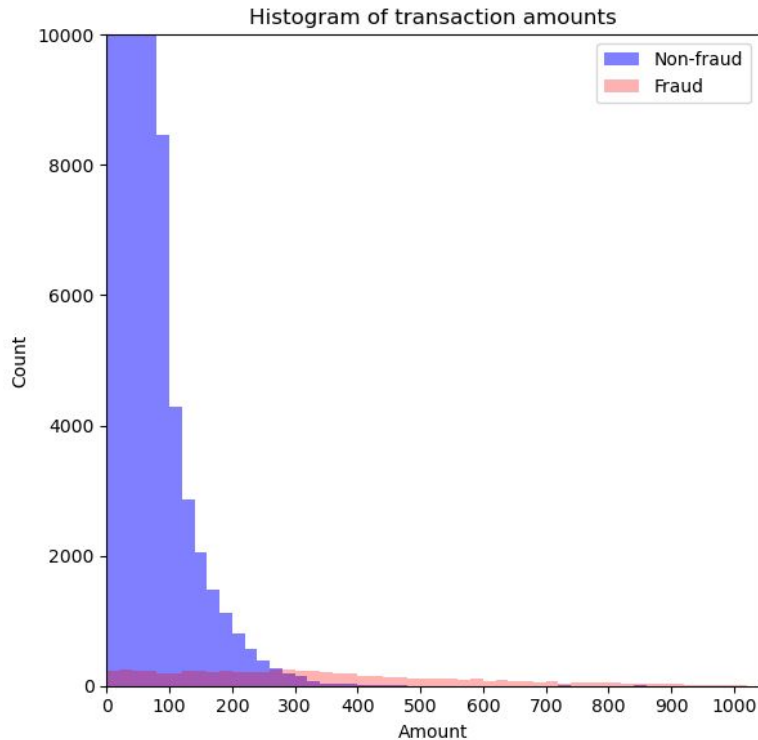
[2] Global Market Insights:

<https://www.gminsights.com/industry-analysis/fraud-detection-and-prevention-market>

Obiettivo del progetto

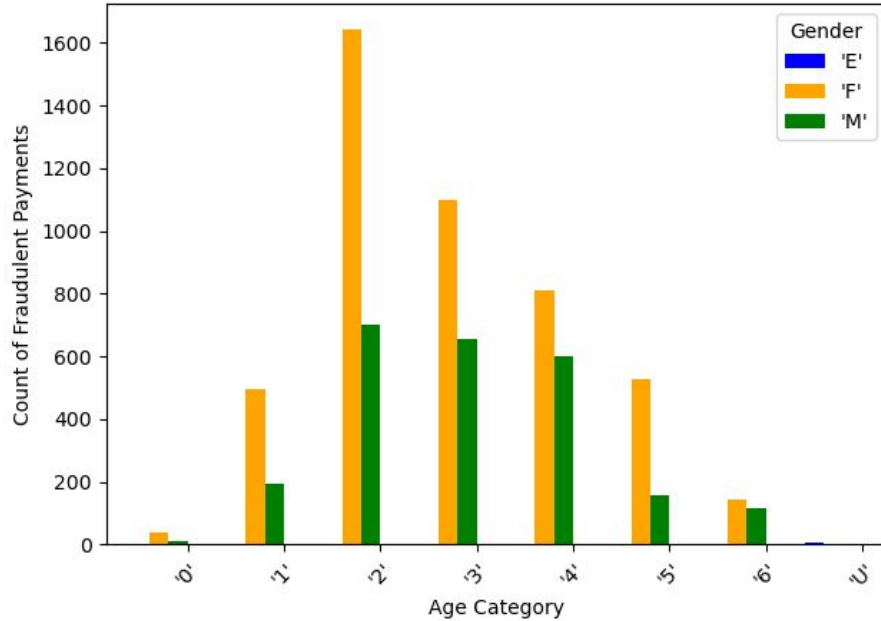
Confrontare le performance di diverse feature map con diversi modelli di quantum machine learning

Data exploration

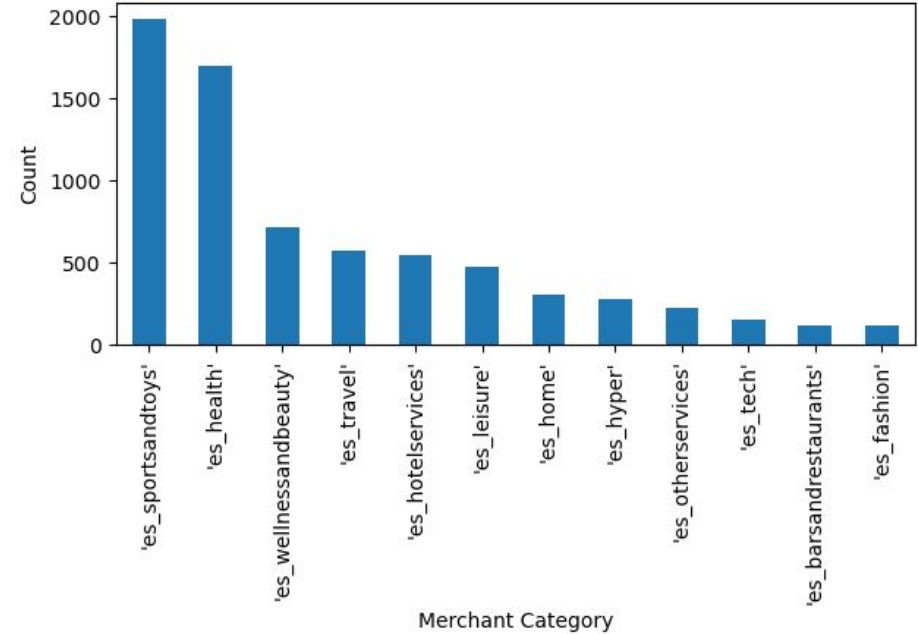


Data exploration

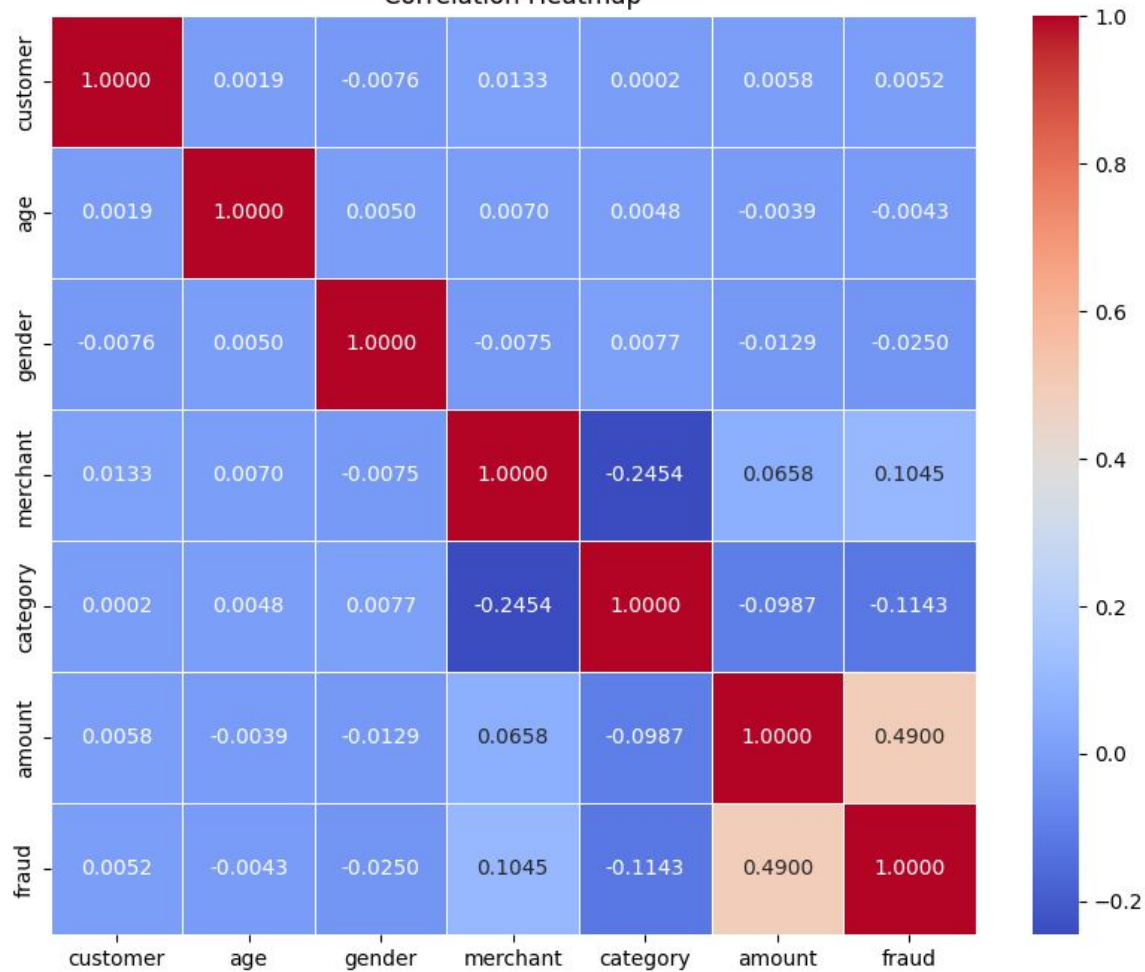
Count of Fraudulent Payments by Age Category and Gender



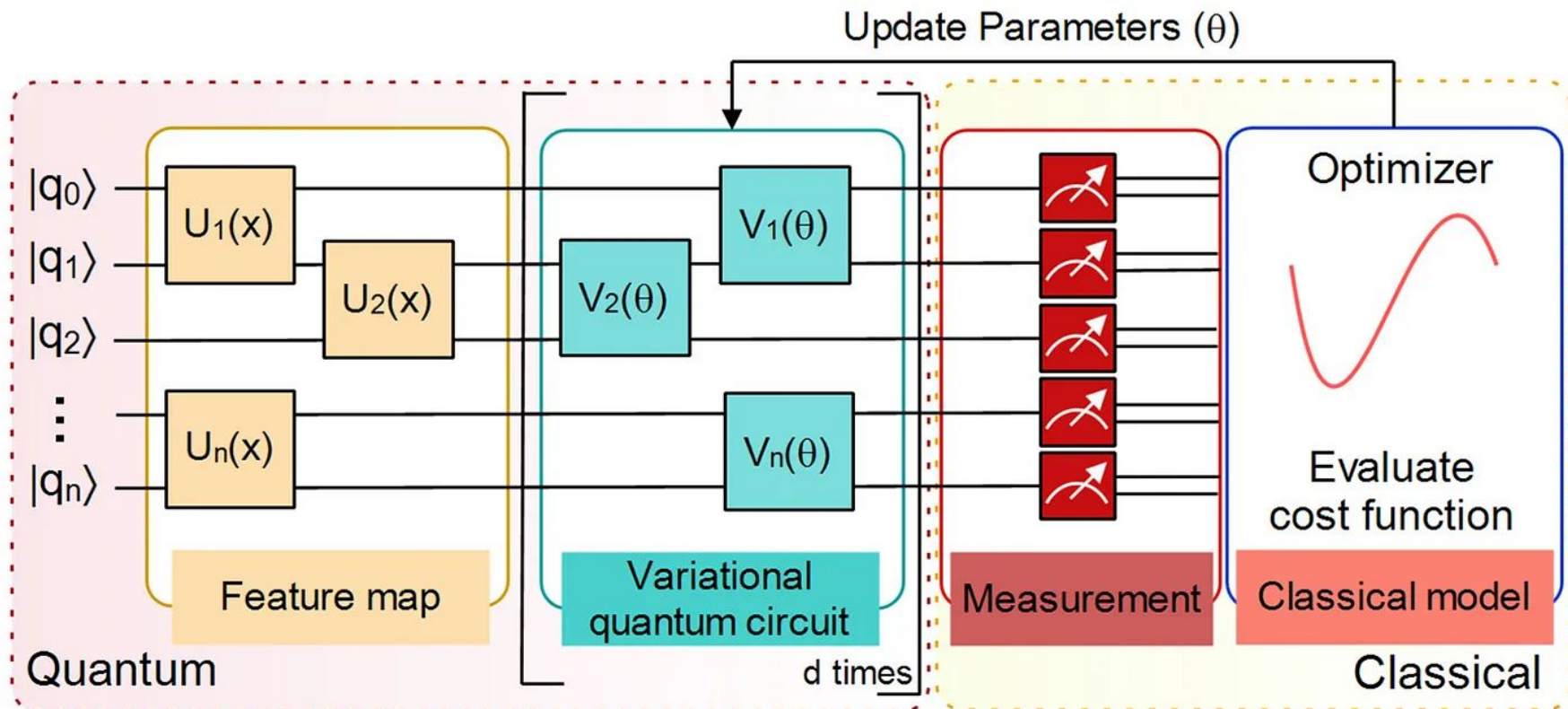
Barplot of Fraudulent Payments by Merchant Category



Correlation Heatmap

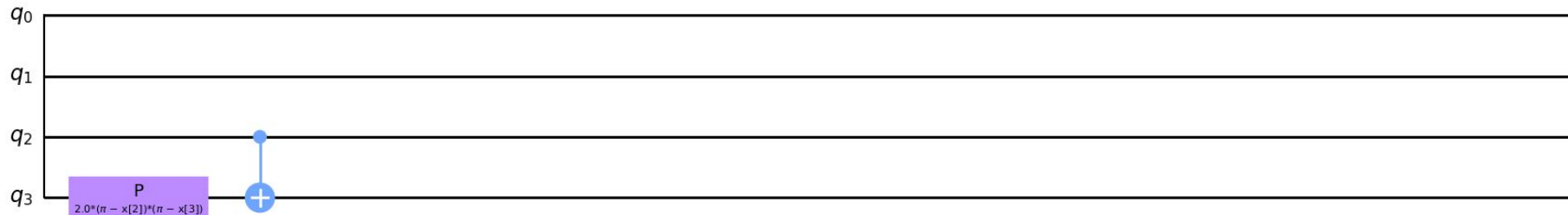
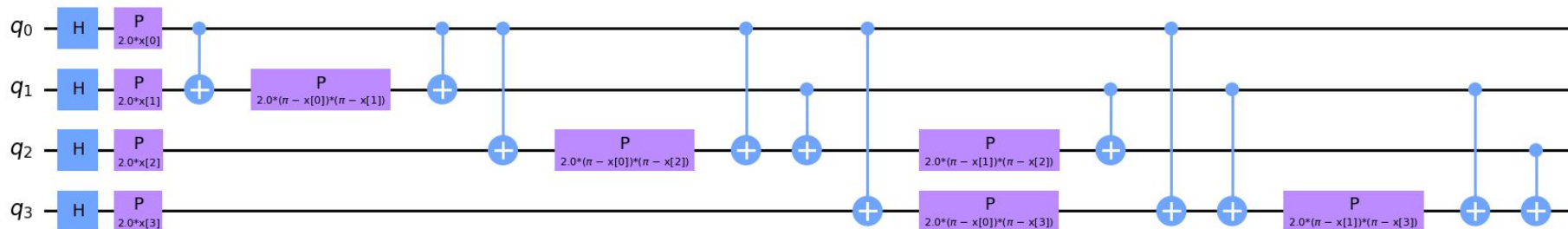


Variational Quantum Classifier



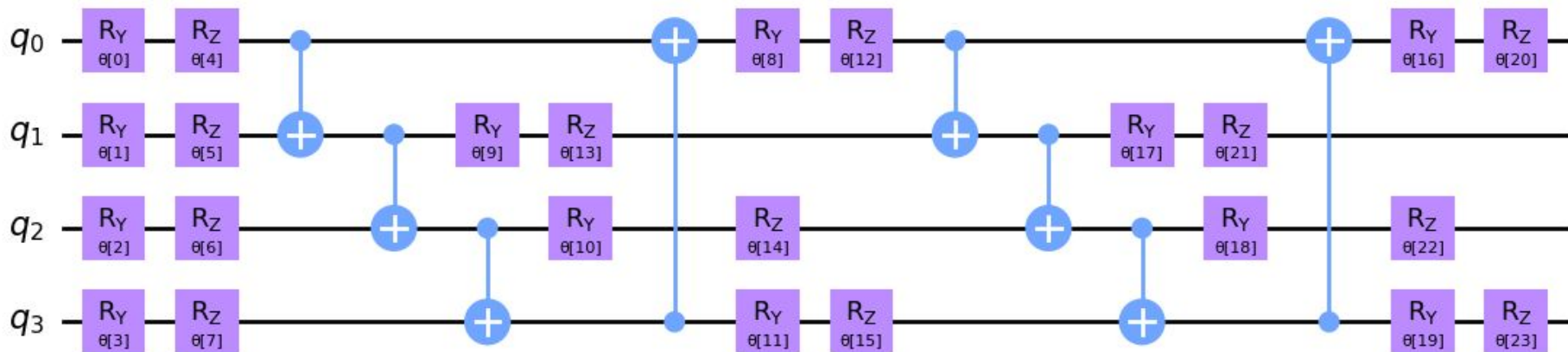
Feature map utilizzate

- ZZFeatureMap
- ZFeatureMap
- PauliFeatureMap (Z, X, ZY)
- ZZFeatureMap (profondità 2)
- ZFeatureMap (profondità 2)

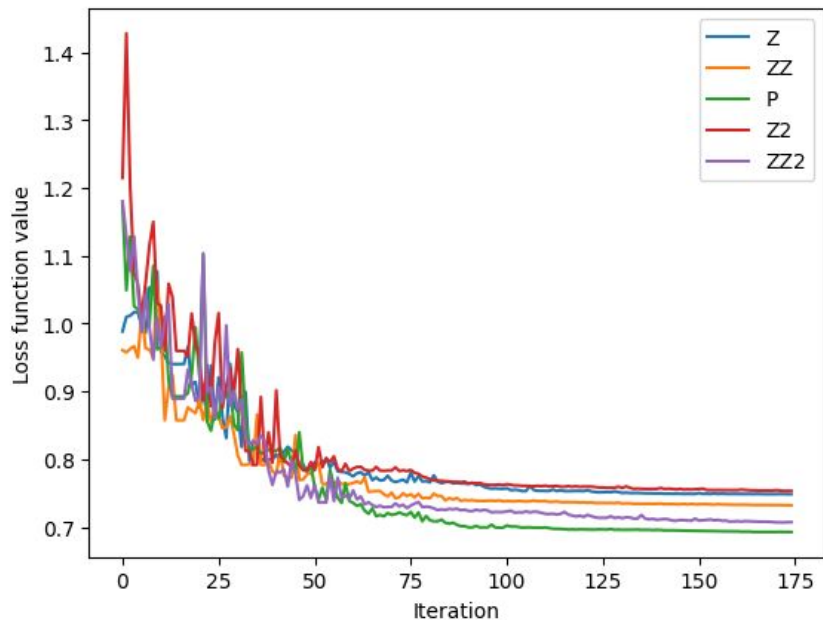


Ansatz utilizzate

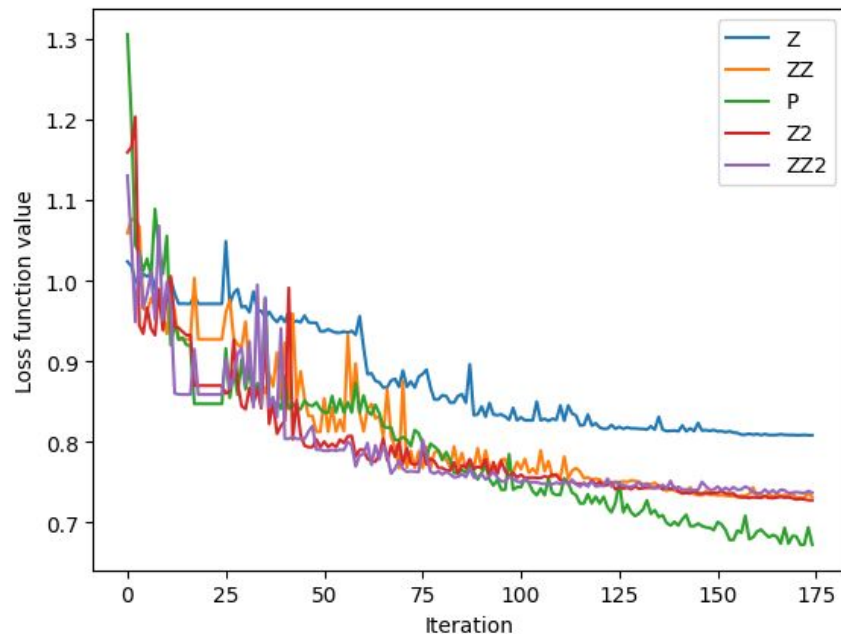
- RealAmplitudes (profondità 3)
- NLocal personalizzata (profondità 3)



Simulatore noiseless: loss functions

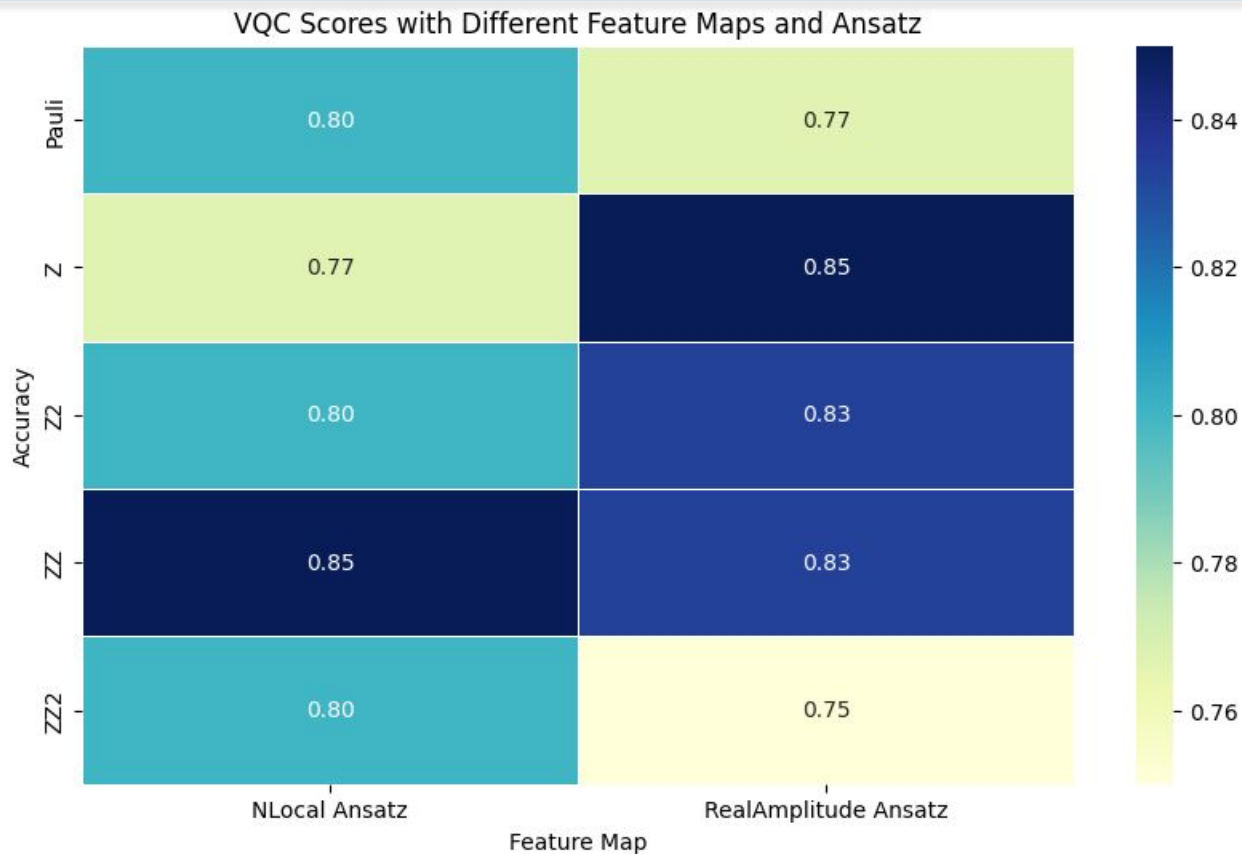


- Ansatz: RealAmplitudes

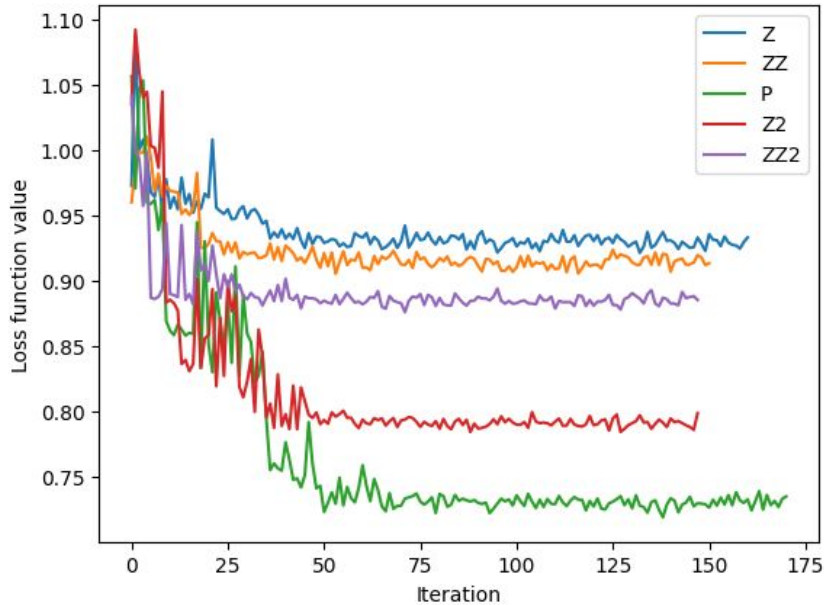


- Ansatz: NLocal personalizzata

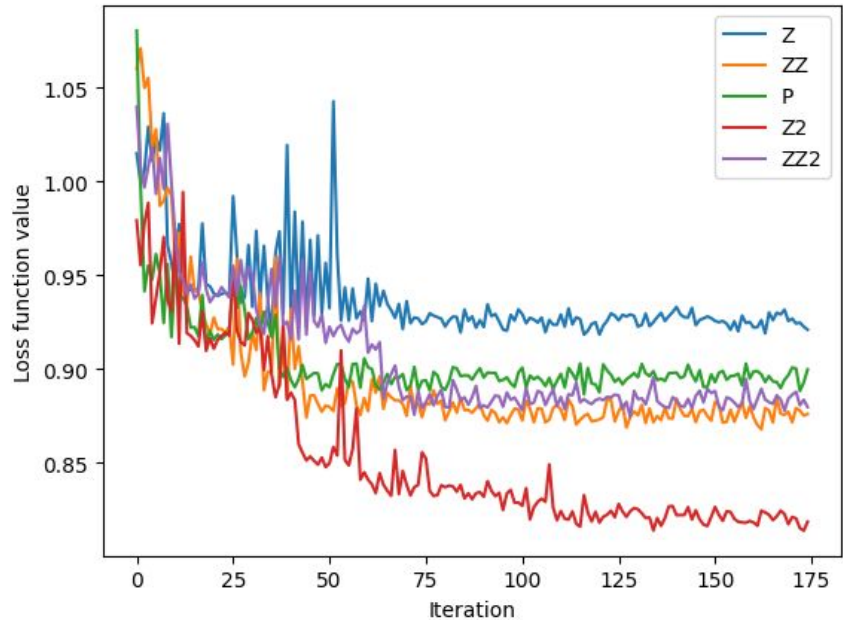
Simulatore noiseless: accuracy



Simulatore noisy: loss functions

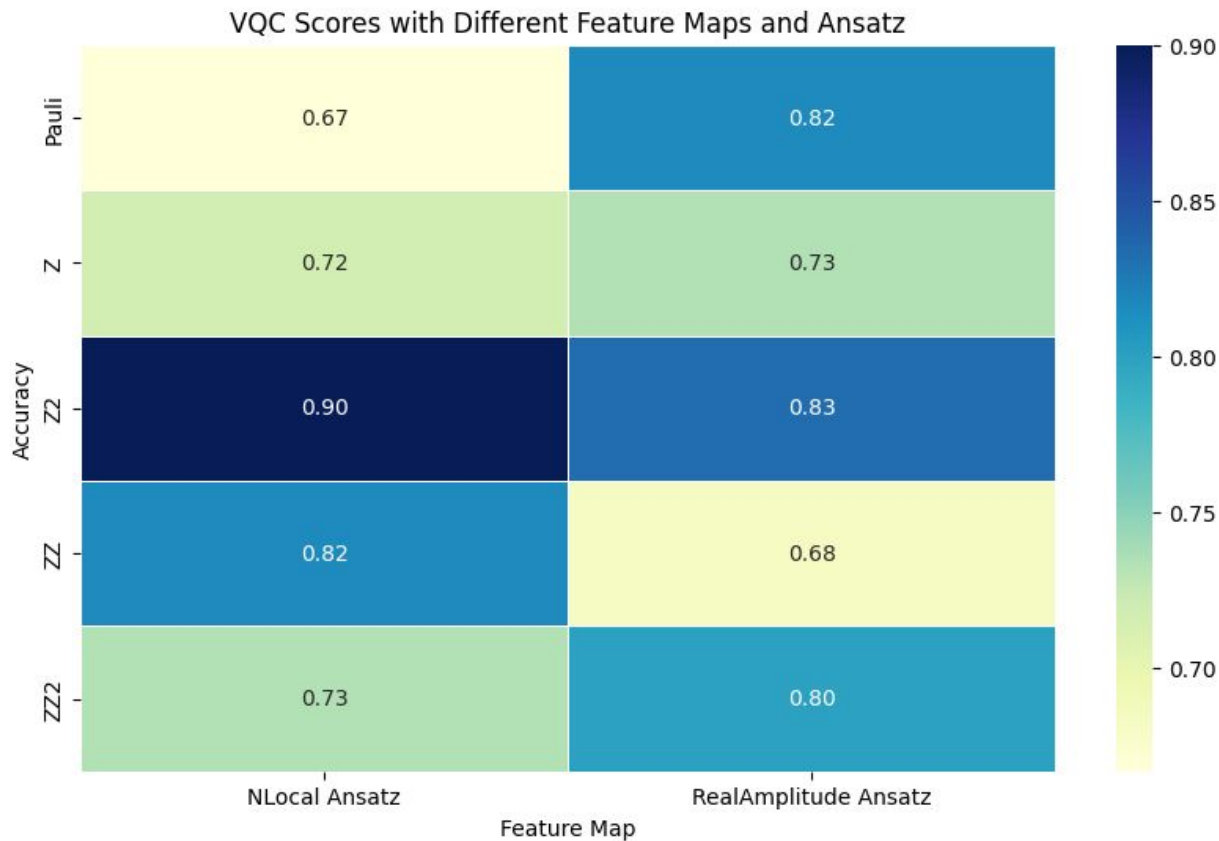


- Ansatz: RealAmplitudes



- Ansatz: NLocal personalizzata

Simulatore noisy: accuracy



Quantum Support Vector Classifiers

kernel classici



kernel quantistici

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

Quantum Kernels

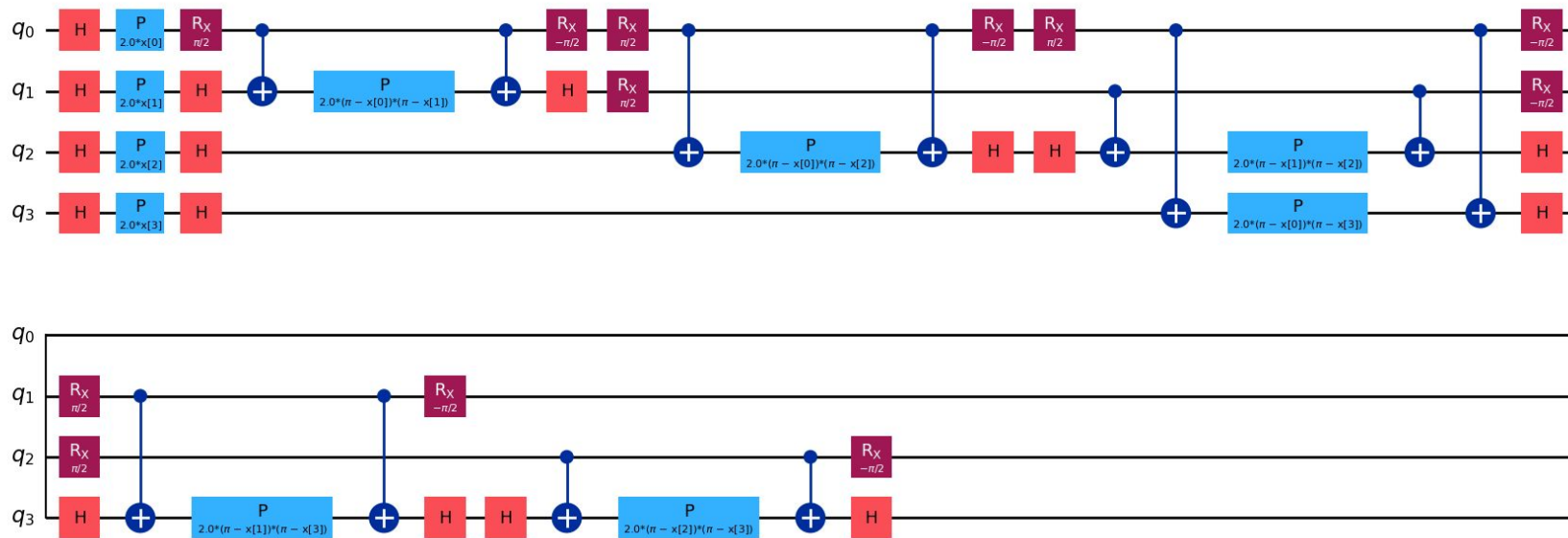
- Funzione ad hoc con SVC
- Fidelity Quantum Kernel con SVC
- Fidelity Quantum Kernel con QSVC

Trainable Quantum Kernels

- diversi ansatz + feature maps
- simulazioni noiseless
- simulazione con noise

Feature maps utilizzate

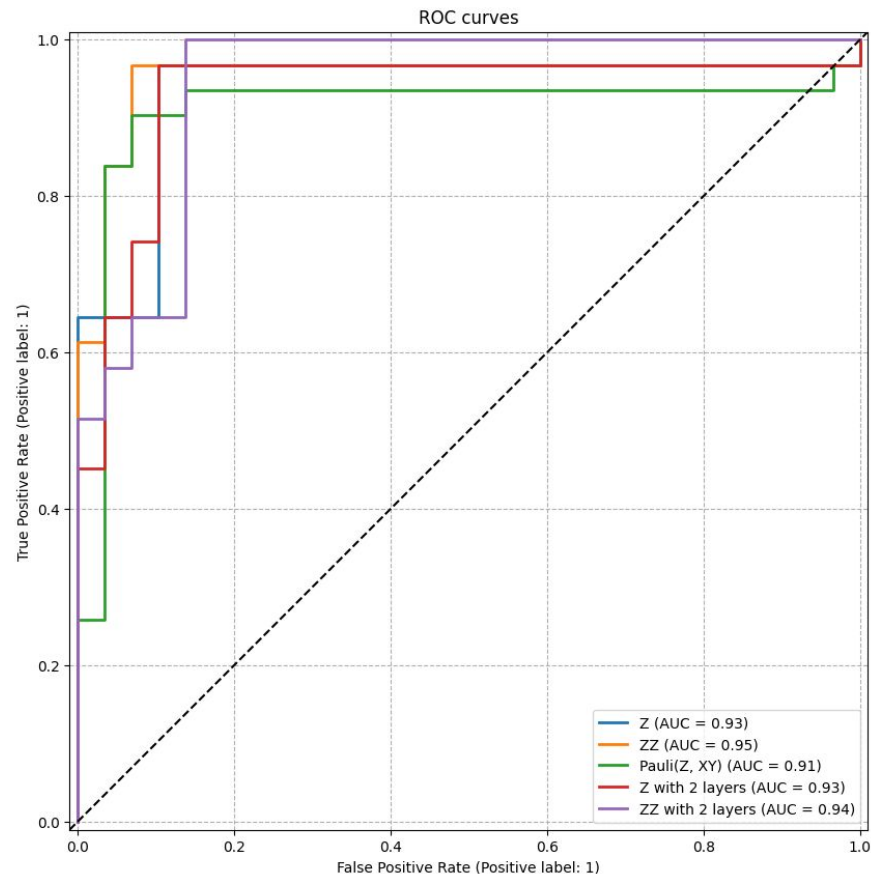
- ZZFeatureMap
- ZFeatureMap
- PauliFeatureMap (Z, XY)
- ZZFeatureMap (profondità 2)
- ZFeatureMap (profondità 2)



Quantum Kernels

classification results

Feature Map	ROC AUC score	Weighed Av. Accuracy
Z Feature Map (profondita' 1)	0.9344	0.88
ZZ Feature Map (profondita' 1)	0.9511	0.92
Pauli Feature Map	0.9077	0.88
Z Feature Map (profondita' 2)	0.9310	0.88
ZZ Feature Map (profondita' 2)	0.9444	0.92



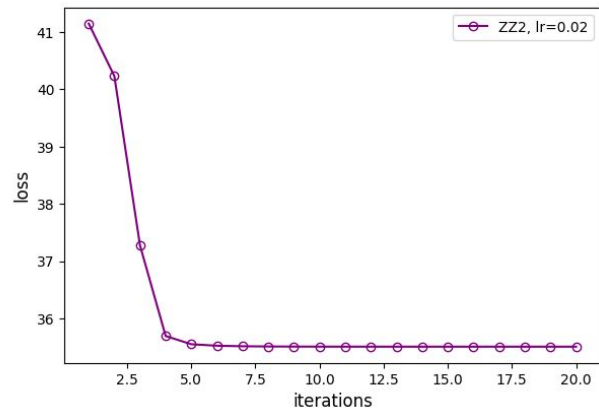
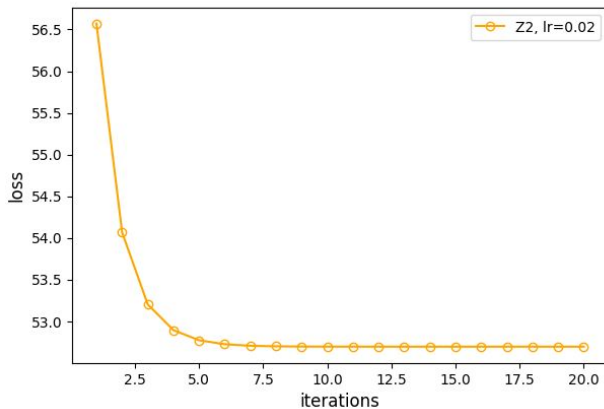
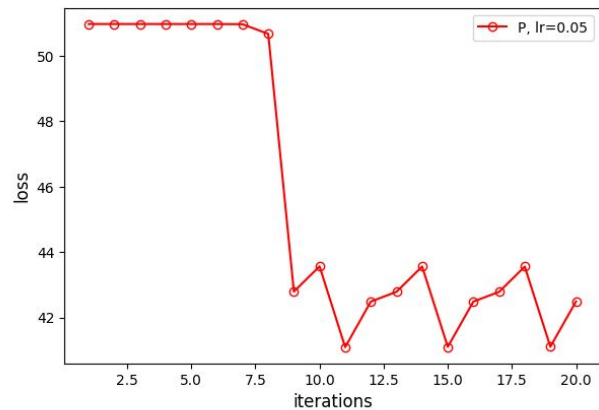
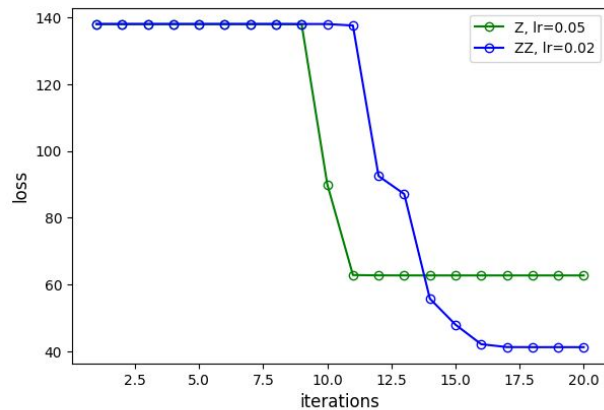
Trainable Quantum Kernels

1. Le feature maps vengono combinate con dei circuiti quantistici parametrici
2. Ottimizzazione dei parametri (SPSA, learning rate e perturbation variabili)
3. Il kernel ottimizzato viene usato nel modello QSVC

Ansatz testati:

- Singolo layer rotazionale con singolo parametro
- Singolo layer rotazionale con parametri multipli
- RealAmplitudes

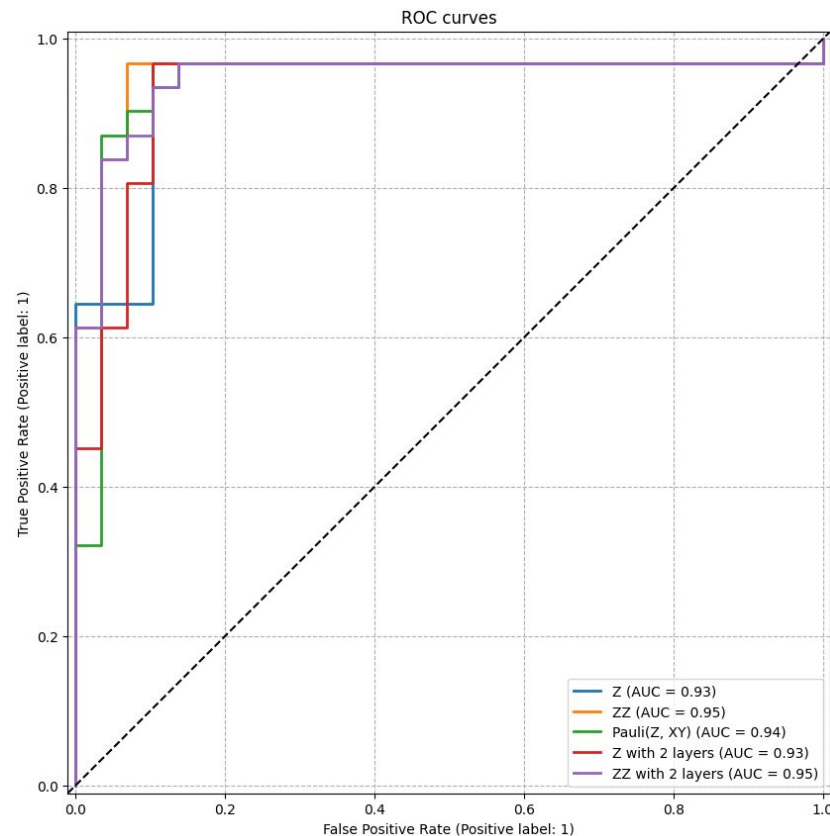
Layer rotazionale con singolo parametro: loss functions



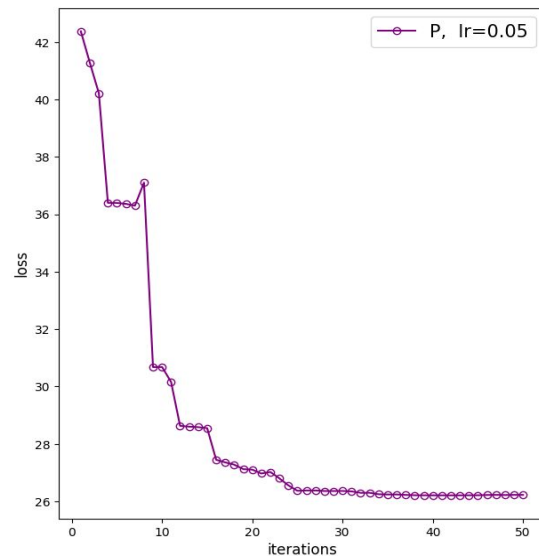
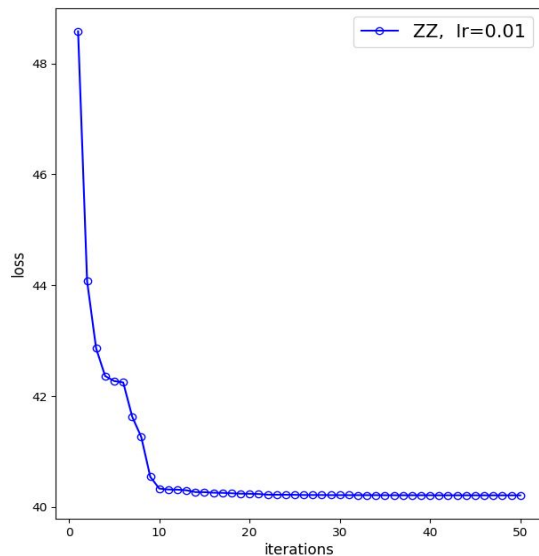
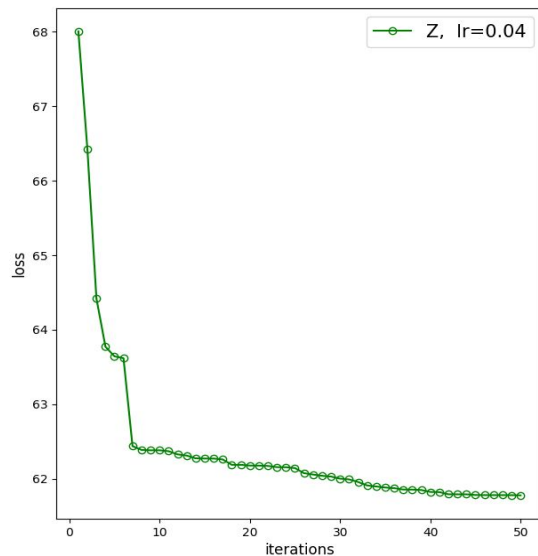
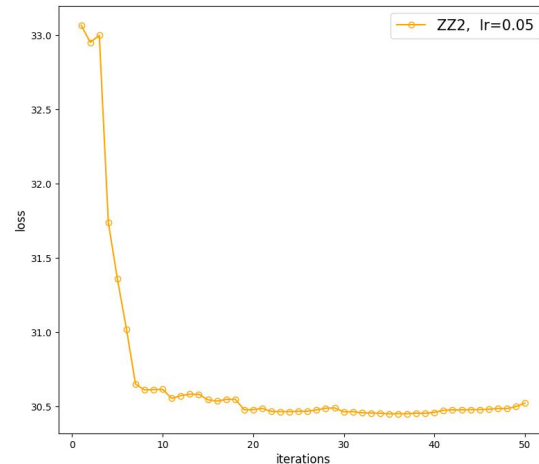
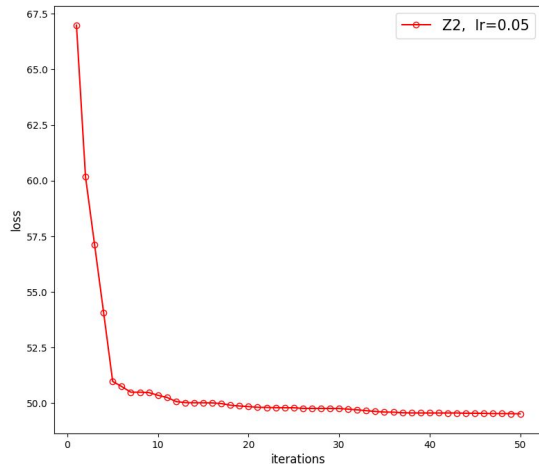
Layer rotazionale con singolo parametro

Risultati

Feature Map	ROC AUC score	Weighed Av. Accuracy
Z Feature Map (profondita' 1)	0.9344	0.88
ZZ Feature Map (profondita' 1)	0.9511	0.92
Pauli Feature Map	0.9388	0.92
Z Feature Map (profondita' 2)	0.9321	0.88
ZZ Feature Map (profondita' 2)	0.9466	0.92



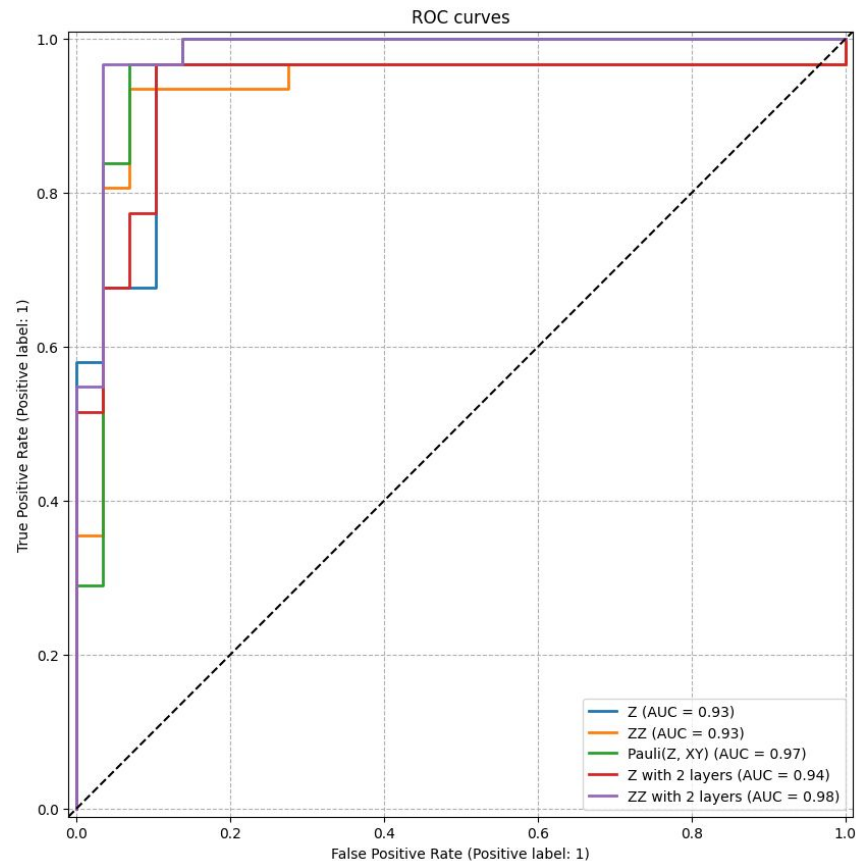
Layer rotazionale con parametri multipli: loss functions



Layer rotazionale con parametri multipli

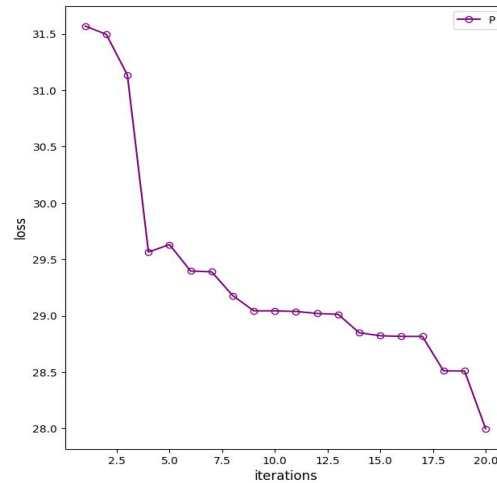
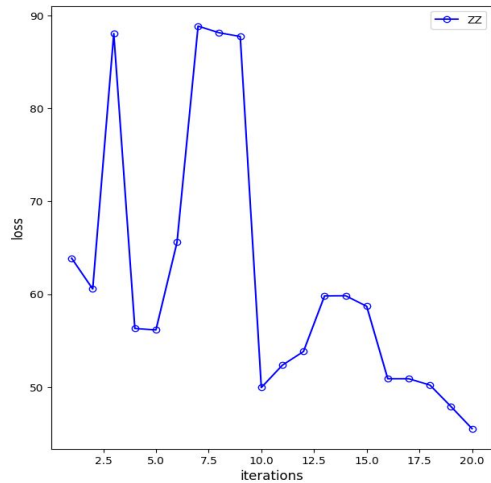
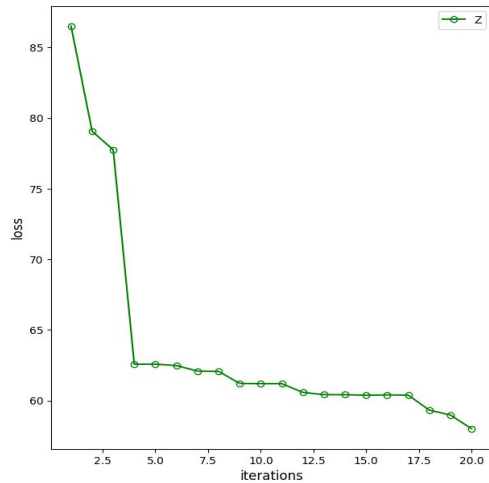
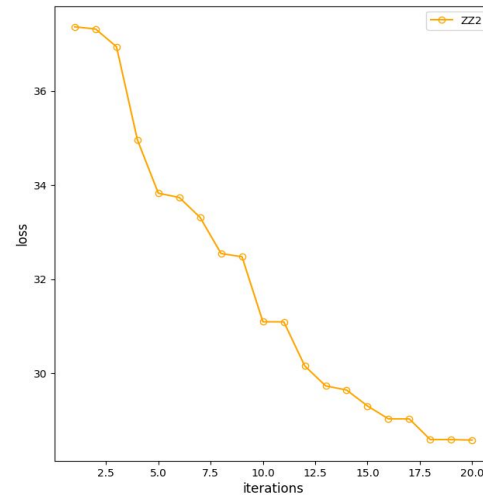
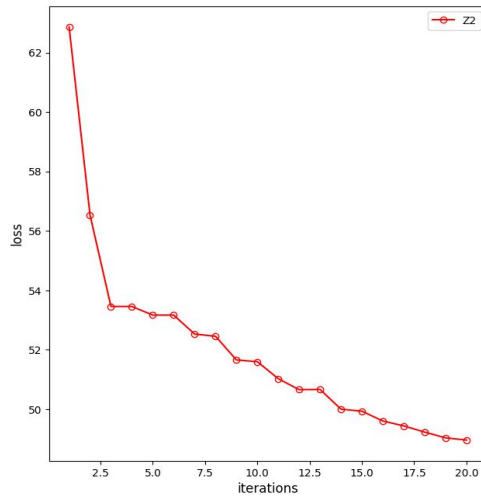
Risultati

Feature Map	ROC AUC score	Weighed Av. Accuracy
Z Feature Map (profondita' 1)	0.9344	0.88
ZZ Feature Map (profondita' 1)	0.9344	0.93
Pauli Feature Map	0.9677	0.95
Z Feature Map (profondita' 2)	0.9355	0.90
ZZ Feature Map (profondita' 2)	0.9811	0.95



RealAmplitudes

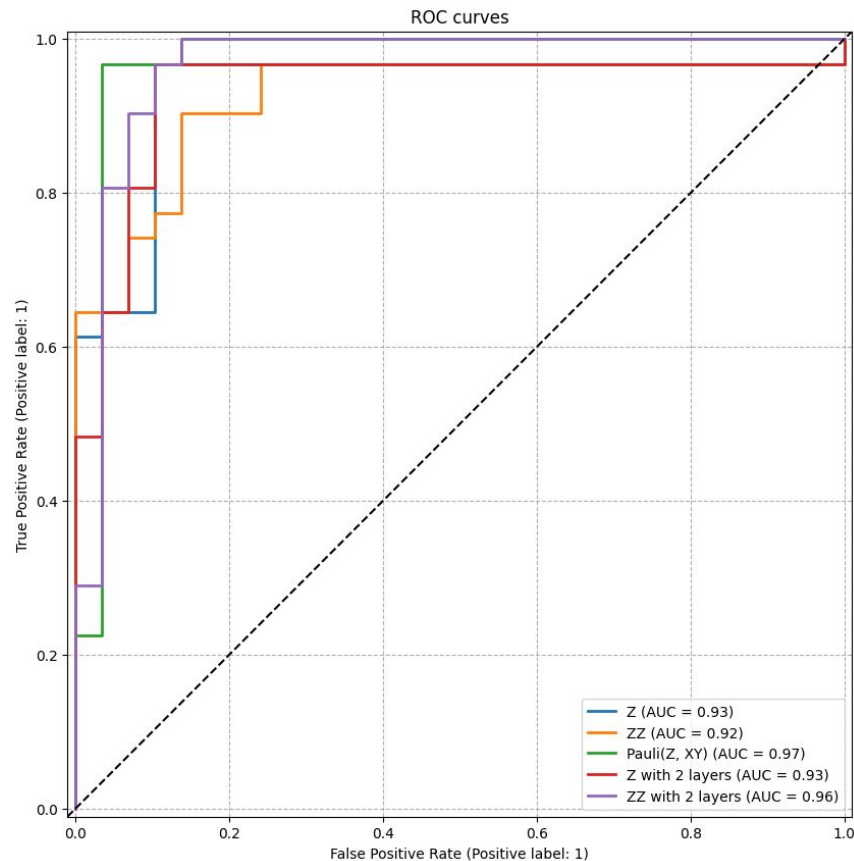
Loss Functions



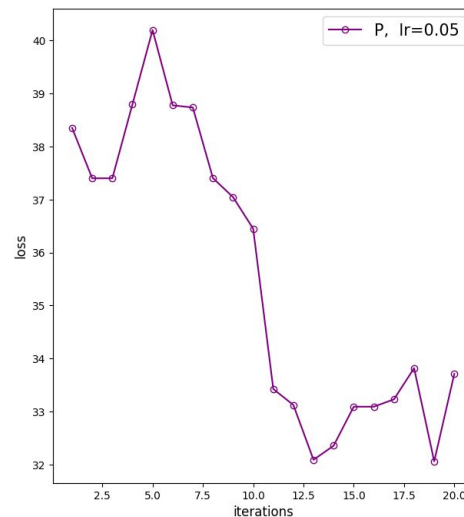
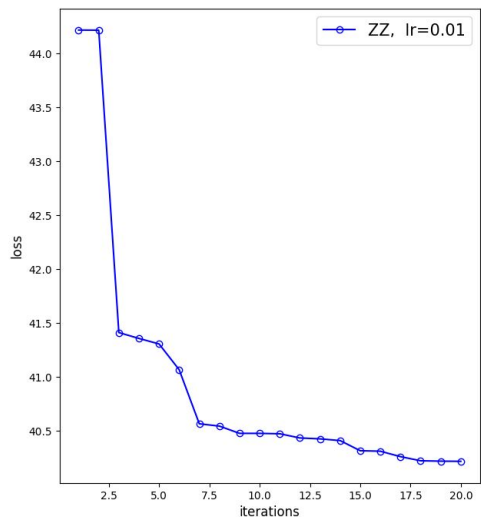
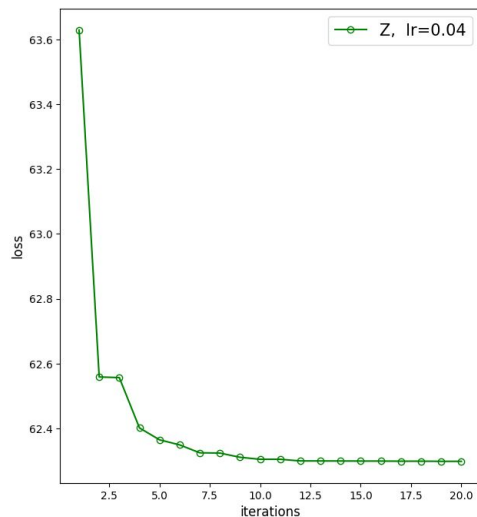
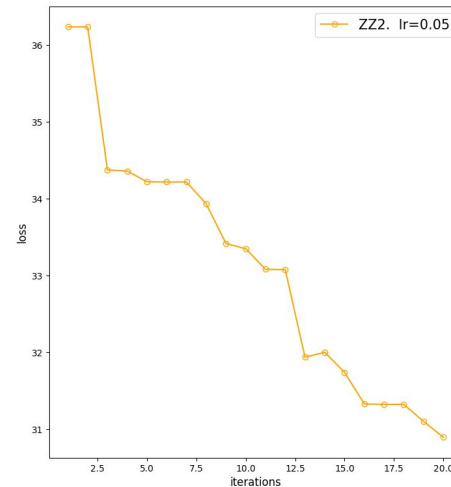
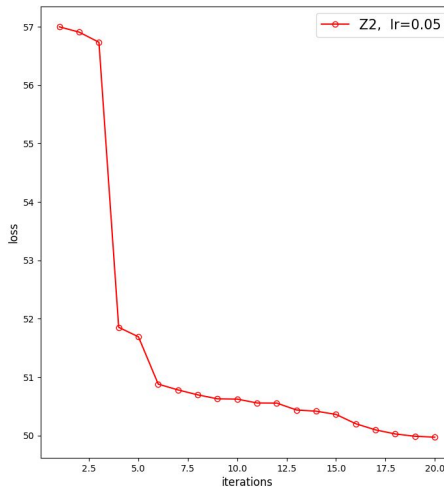
RealAmplitudes

Risultati

Feature Map	ROC AUC score	Weighed Av. Accuracy
Z Feature Map (profondita' 1)	0.9333	0.88
ZZ Feature Map (profondita' 1)	0.9244	0.83
Pauli Feature Map	0.9700	0.97
Z Feature Map (profondita' 2)	0.9344	0.90
ZZ Feature Map (profondita' 2)	0.9644	0.90



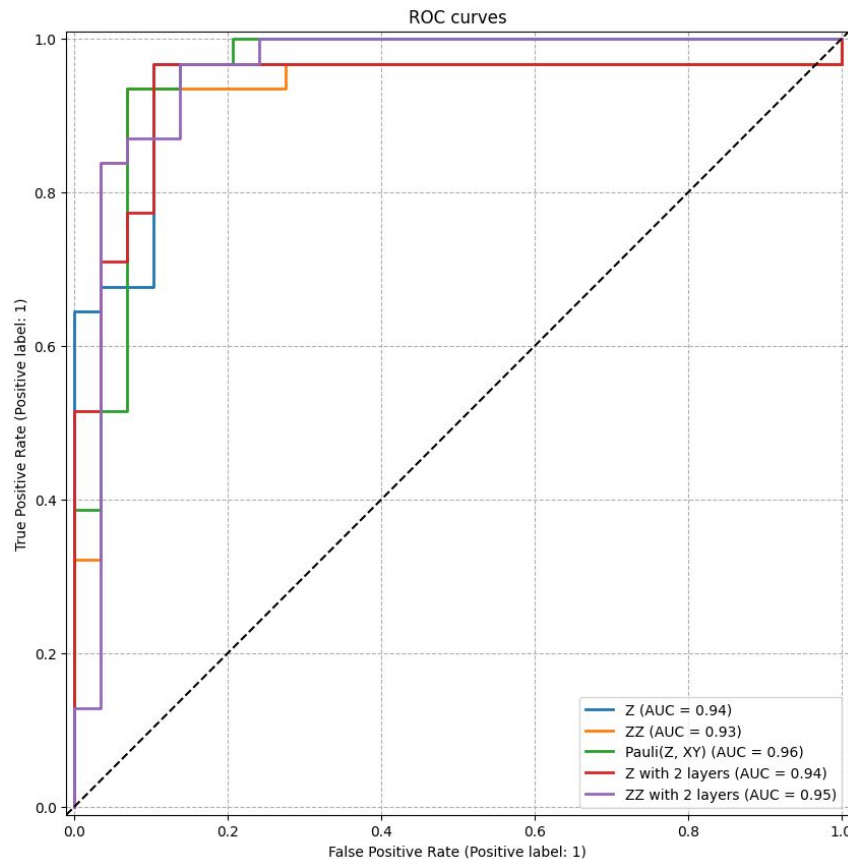
Layer rotazionale con parametri multipli: simulazione con noise



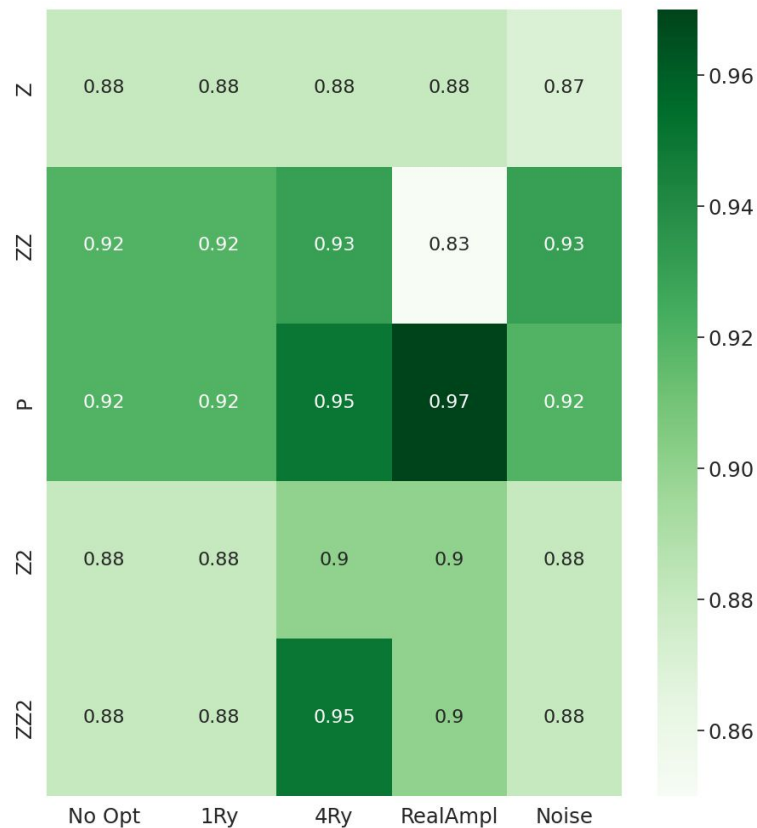
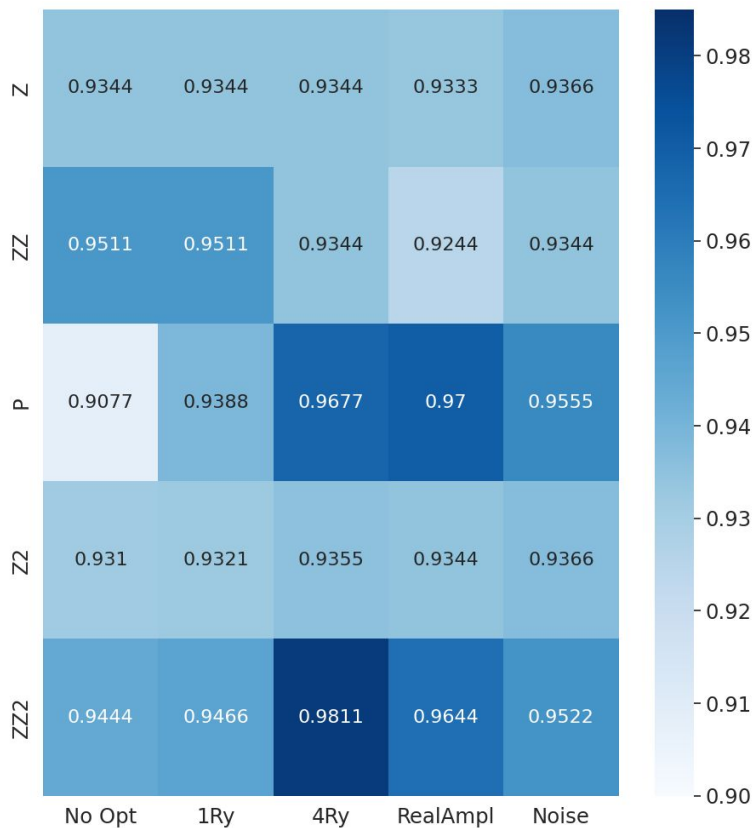
Layer rotazionale con parametri multipli: simulazione con noise

Risultati

Feature Map	ROC AUC score	Weighed Av. Accuracy
Z Feature Map (profondita' 1)	0.9366	0.87
ZZ Feature Map (profondita' 1)	0.9344	0.88
Pauli Feature Map	0.9555	0.92
Z Feature Map (profondita' 2)	0.9366	0.88
ZZ Feature Map (profondita' 2)	0.9522	0.88



Risultati Finali



Risultati Finali

Per fare un confronto con modelli puramente classici:

- SVC con kernel polinomiale: ROC AUC = 0.9577, Accuracy = 0.91
- SVC con kernel gaussiano: ROC AUC = 0.9792, Accuracy = 0.94

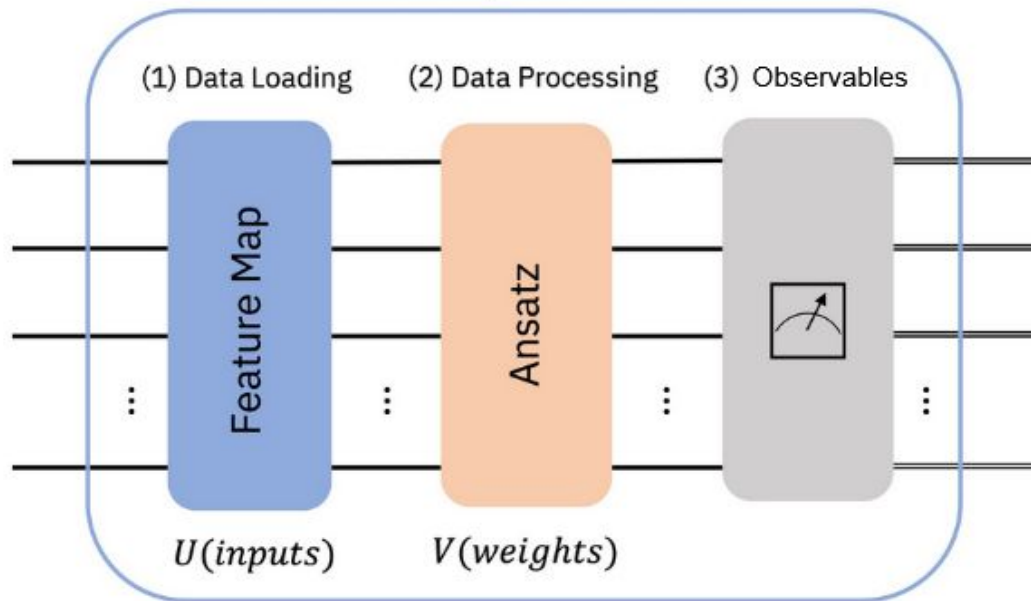
Questi risultati sono stati ottenuti con un dataset “large” contenente 14000 osservazioni anziche' 200

Quantum Neural Networks

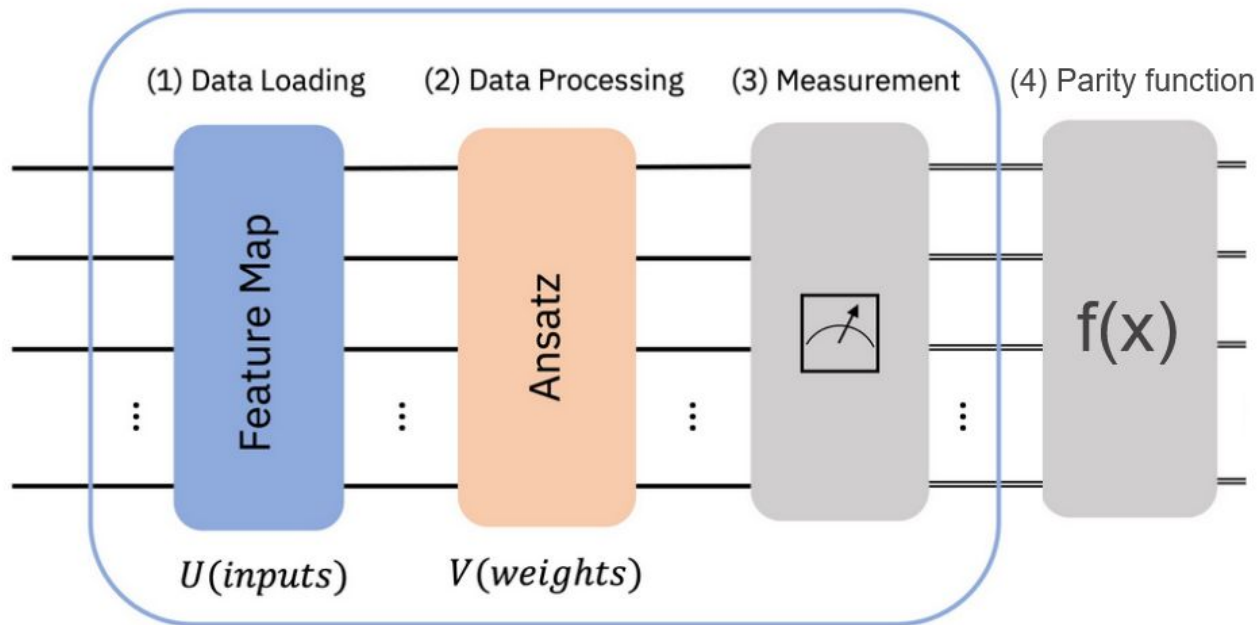
Abbiamo implementato tre differenti modelli di Neural Networks quantistiche:

- Estimator QNN
- Sampler QNN
- QNN ibrida

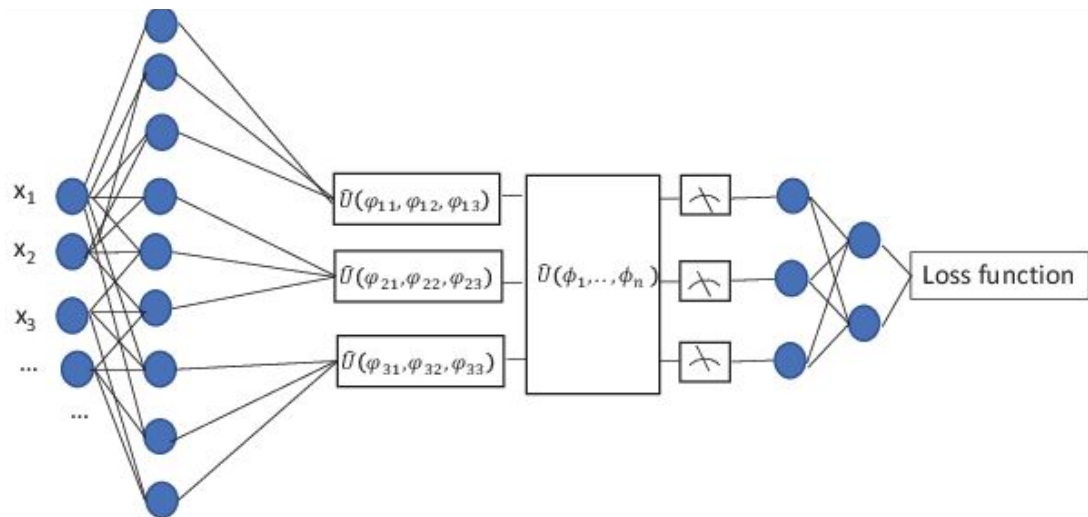
Estimator QNN



Sampler QNN



QNN ibrida



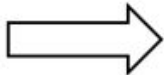
Dataset usato: alcune considerazioni

Andando ad analizzare il dataset si può notare che contiene sia feature numeriche (l'importo della transazione) sia feature categoriche, come in particolare la feature "categoria" che rappresenta la categoria dell'acquisto.

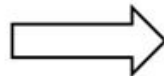
Per la feature "categoria" abbiamo provato sia un encoding numerico (dataset "thin") sia un encoding binario (dataset "large") ed abbiamo notato che per alcuni modelli quest'ultimo porta a risultati migliori

Binary Encoding

Places
New York
Boston
Chicago
California



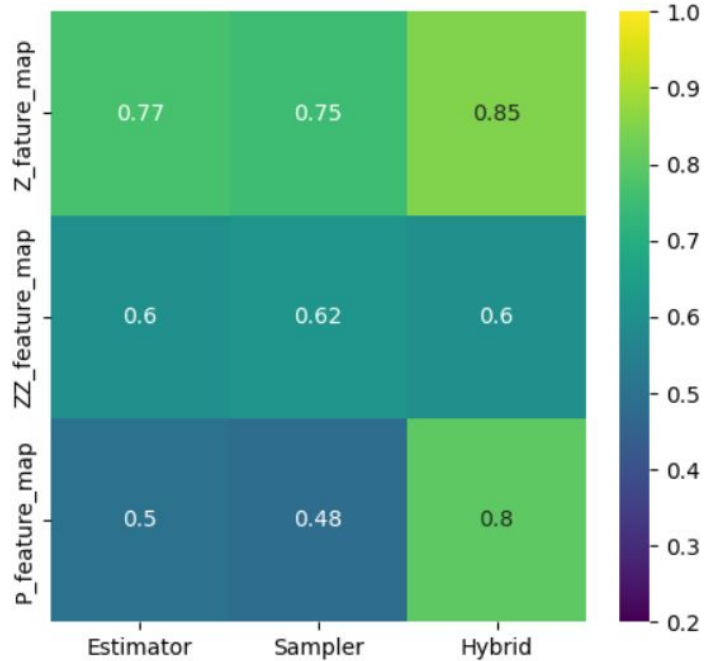
Feature	Map
New York	0
Boston	1
Chicago	2
California	3



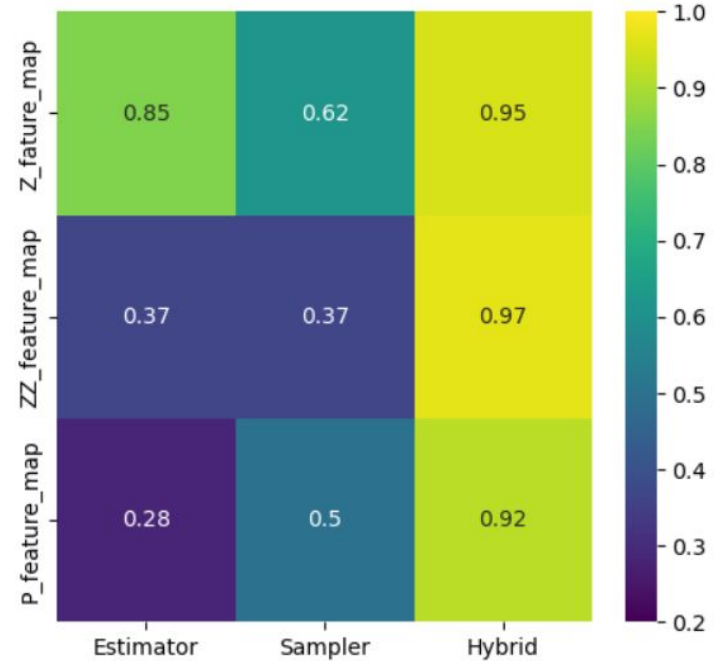
Feature	Encoded
New York	0
Boston	1
Chicago	10
California	11

Risultati QNN

Dataset Thin

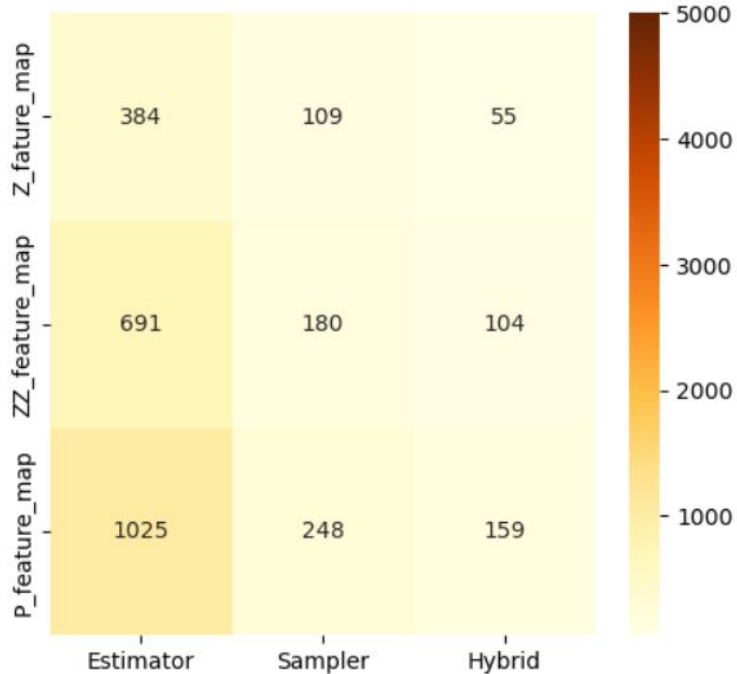


Dataset Large

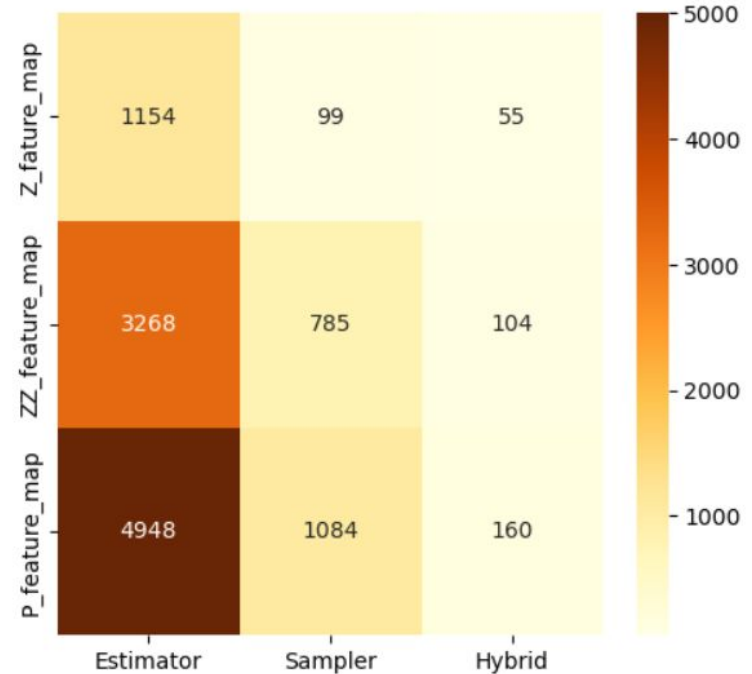


Risultati QNN

Dataset Thin



Dataset Large



Ulteriori test

Avendo notato che il modello con risultati migliori è la QNN ibrida, abbiamo fatto ulteriori test per esaminare le potenzialità di questo modello.

Abbiamo continuato ad usare l'encoding "large" ed abbiamo aumentato il numero di epoche dell'allenamento e la grandezza del dataset. Inoltre abbiamo aumentato la complessità del modello classico e di quello quantistico.

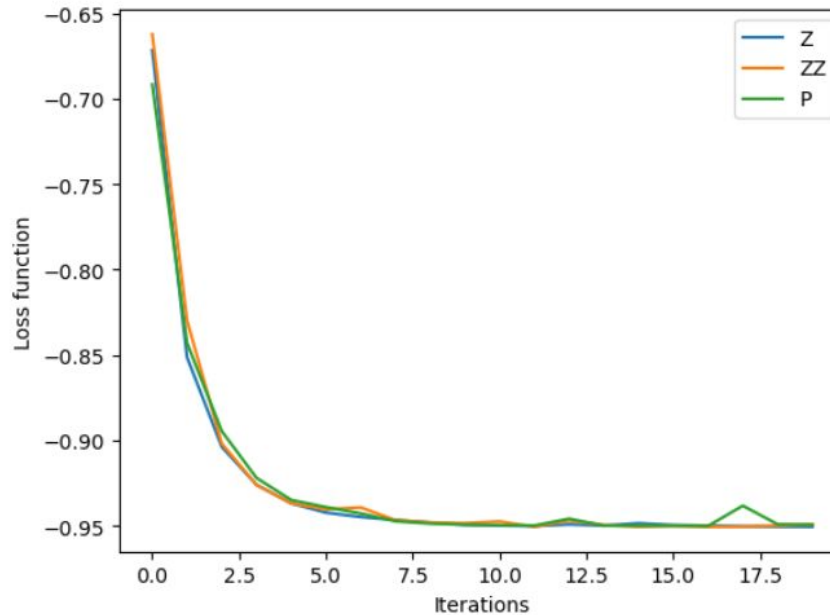
Successivamente abbiamo confrontato tale modello con un modello noisy ed un modello puramente classico

Modello ibrido

```
-- -- -- --  
Time spent for Hybrid QNN with feature map Z and dataset Large : 2040.5503914356232  
The accuracy is : 0.9633333333333334
```

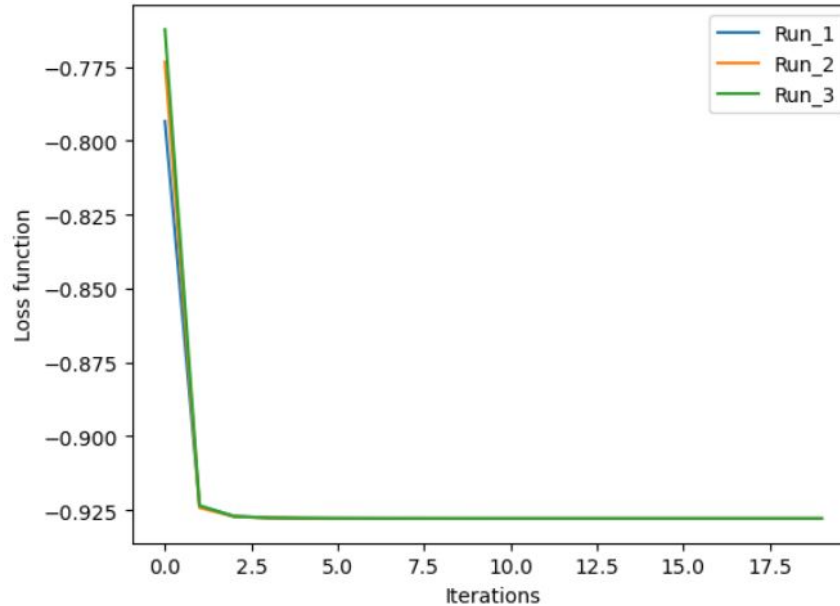
```
-- -- -- --  
Time spent for Hybrid QNN with feature map ZZ and dataset Large : 4192.692080497742  
The accuracy is : 0.9633333333333334
```

```
-- -- -- --  
Time spent for Hybrid QNN with feature map P and dataset Large : 6661.815709114075  
The accuracy is : 0.9633333333333334
```



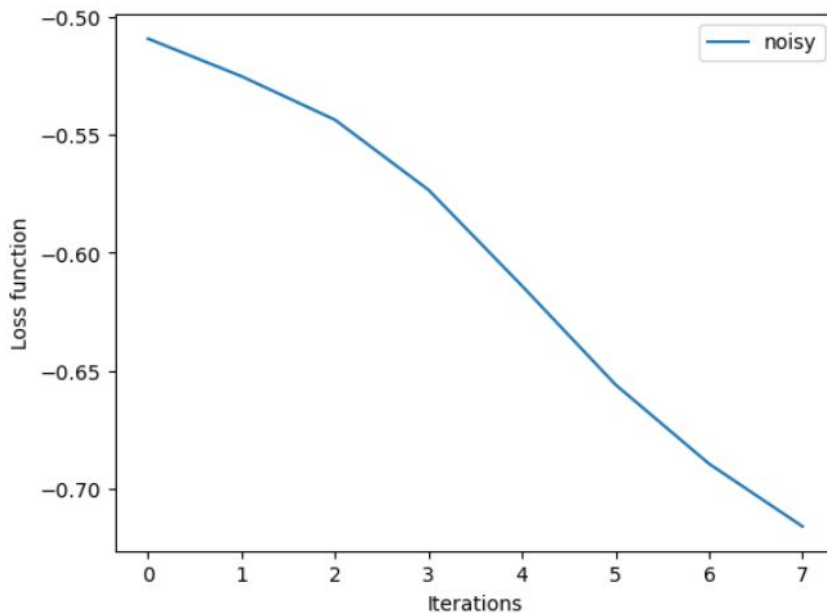
Modello classico

```
-- -- --  
Time spent for Classical QNN number 0 with dataset Large : 19.899761199951172  
The accuracy is : 0.935  
-- -- --  
Time spent for Classical QNN number 1 with dataset Large : 19.916569471359253  
The accuracy is : 0.935  
-- -- --  
Time spent for Classical QNN number 2 with dataset Large : 20.651018619537354  
The accuracy is : 0.935  
-- -- --
```



Modello noisy

Time spent for Noisy QNN with dataset Large : 1102.8102567195892
The accuracy is : 0.9333333333333333



Conclusioni

Se consideriamo l'encoding numerico tutti i modelli hanno risultati comparabili, mentre se usiamo l'encoding binario ed aumentiamo il numero di qubit il modello ibrido risulta il migliore.

I risultati del modello ibrido sono influenzati in parte dal layer classico, anche se il modello puramente classico non ottiene gli stessi risultati di accuracy.

In conclusione il modello ibrido può portare ad ottimi risultati e l'utilizzo di un layer classico prima di uno quantistico può risolvere i problemi legati all'uso di troppi qubit.

Considerazioni finali

Dai nostri esperimenti si è potuto notare che le performance degli algoritmi di quantum machine learning sono ancora paragonabili se non inferiori a quelle dei corrispettivi algoritmi classici.

Il problema è che attualmente non è possibile mettere in pratica quegli algoritmi quantistici che potrebbero portare uno speedup esponenziale rispetto alle prestazioni classiche.

Un esempio tra tutti è l'algoritmo per la QSVM che richiede una complessità computazionale di $O(\log(N * M))$.

Tale algoritmo è basato su altri algoritmi quantistici come l'HHL che al momento non sono ancora stati implementati con successo.

Grazie della visione