

## CHALLENGE 2 – REPORT

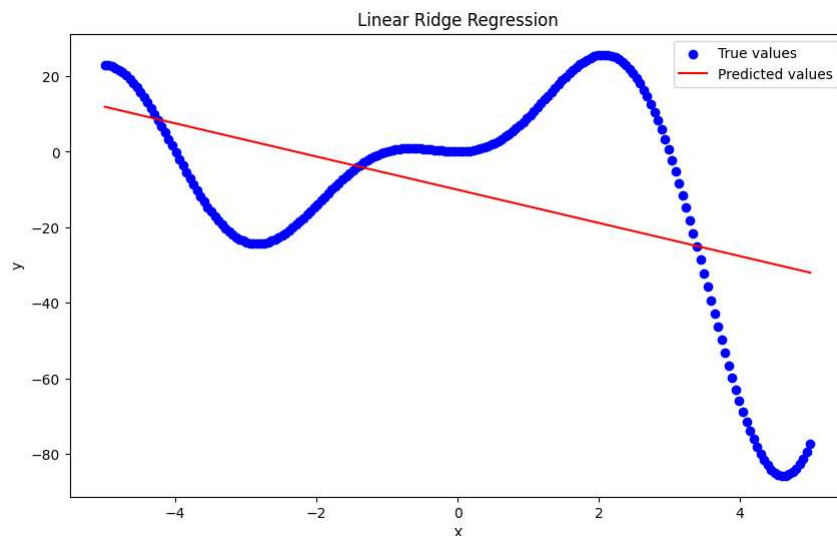
In questa challenge è stata studiata l'applicazione dei metodi di kernel su alcuni task di machine learning, nello specifico la Ridge Regression e la PCA, messe a confronto con le loro controparti kernellizzate su dei dataset creati artificialmente. L'obiettivo è stato osservare sotto quali condizioni i metodi kernellizzati performano meglio rispetto alle loro versioni base.

Per l'implementazione dei modelli si ha fatto affidamento alla libreria sklearn.

### RIDGE REGRESSION

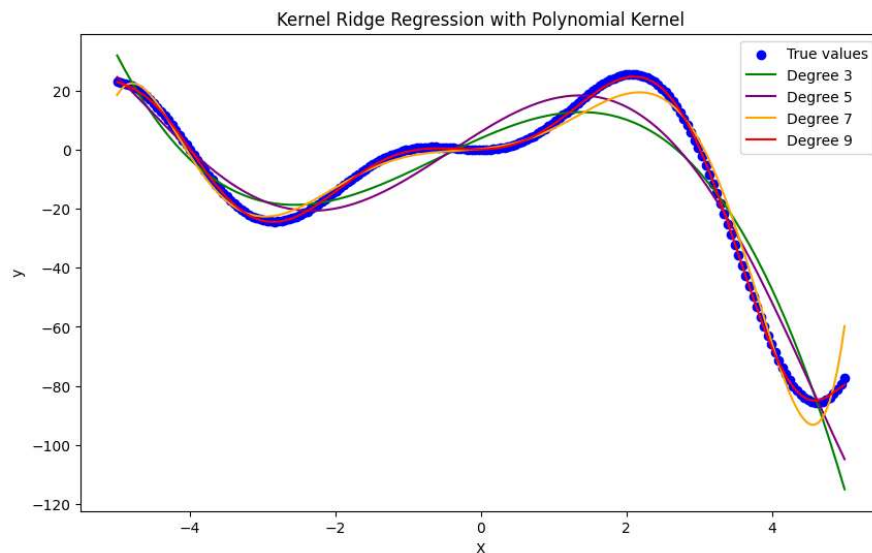
Il dataset impiegato in questa sezione include una feature continua e una variabile target continua, la cui dipendenza dalla feature segue un andamento polinomiale, con l'aggiunta di rumore gaussiano.

Di seguito è riportato il risultato della **regressione lineare**, che come si poteva prevedere risulta essere non idoneo a ricostruire relazioni non lineari tra i dati:



Kernellizzando questo metodo, prima con kernel polinomiale e poi tramite kernel gaussiano, possiamo notare come, almeno all'interno dell'intervallo dei dati a nostra disposizione, la regressione risulta fittare molto meglio i nostri dati.

Gli iperparametri sono stati selezionati tramite la classe GridSearchCV di sklearn.

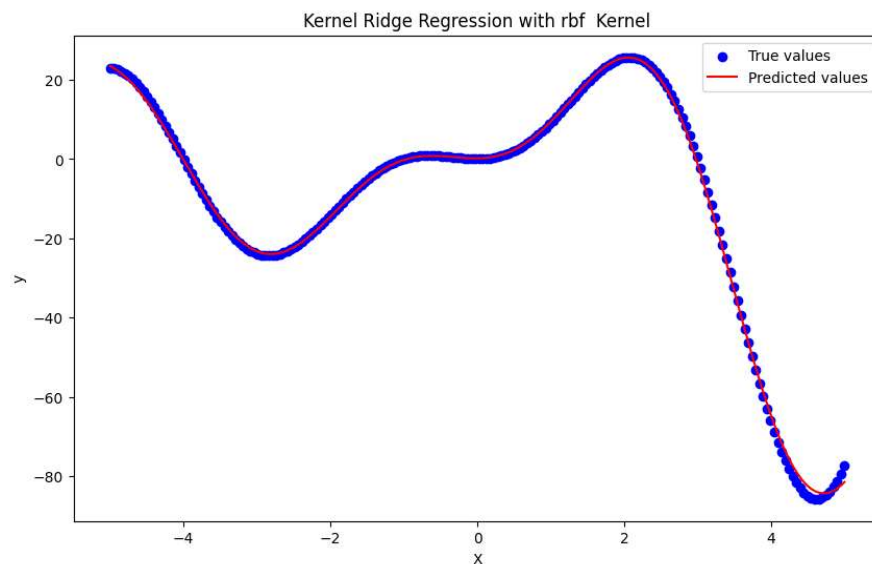


*Polynomial kernel of degree 3,  
Best alpha: 300, RMSE: 9.52*

*Polynomial kernel of degree 5,  
Best alpha: 100, RMSE: 8.53*

*Polynomial kernel of degree 7,  
Best alpha: 2100, RMSE: 3.64*

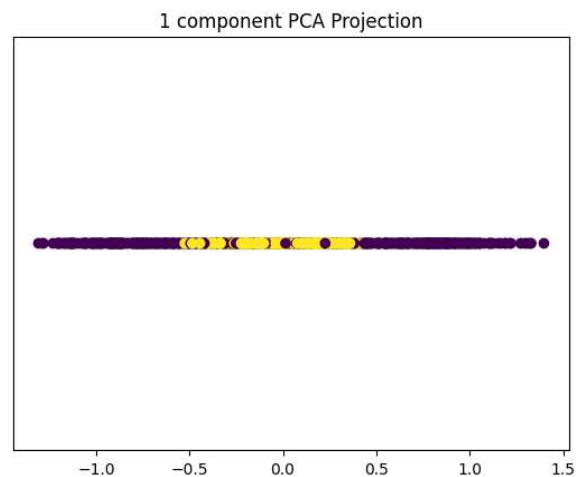
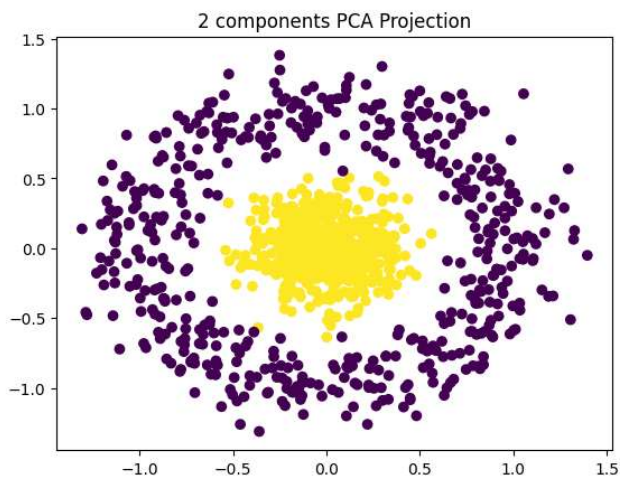
*Polynomial kernel of degree 9,  
Best alpha: 2100, RMSE: 0.54*



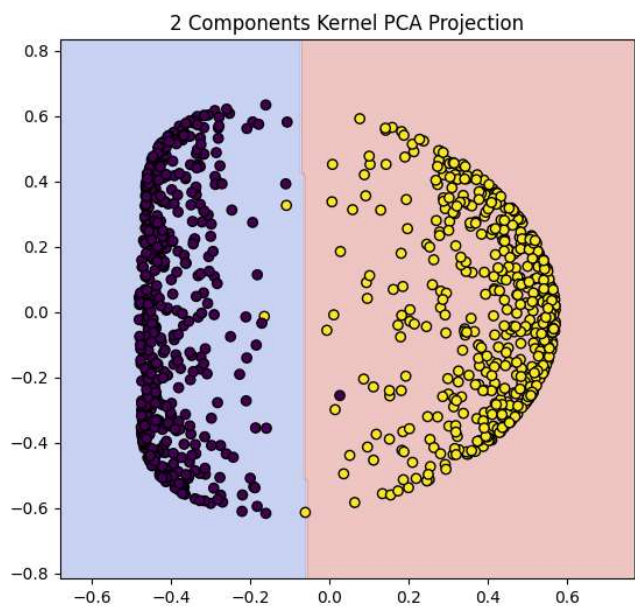
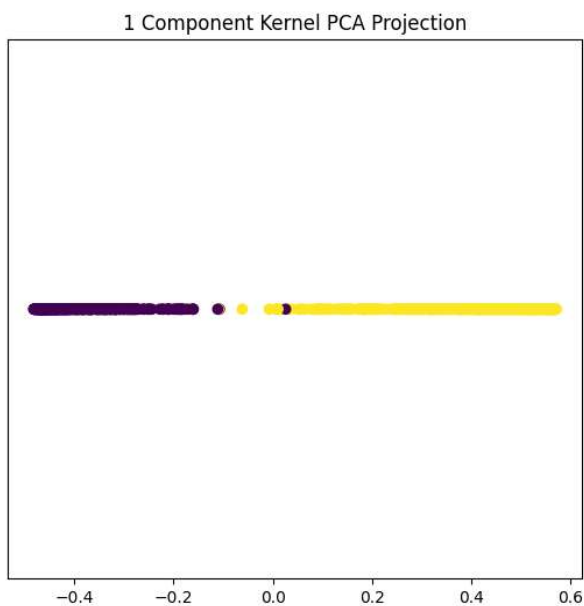
*Gaussian kernel,  
gamma: 0.28, RMSE: 0.90*

## PCA

Il dataset impiegato in questa sezione consta di due cerchi concentrici, ci troviamo ancora in uno spazio di due dimensioni. Per questo motivo PCA con due componenti principali non fa altro che ruotare il dataset, quindi ci interesseremo anche alla proiezione su una componente principale.



Risulta chiaro come anche PCA fatica a trovare relazioni non lineari, quindi si procede con la kernellizzazione del metodo. Data la natura del dataset, è stato scelto un kernel gaussiano, la cui performance è stata valutata tramite **Support Vector Classifier**, evidenziato nella proiezione in due dimensioni.

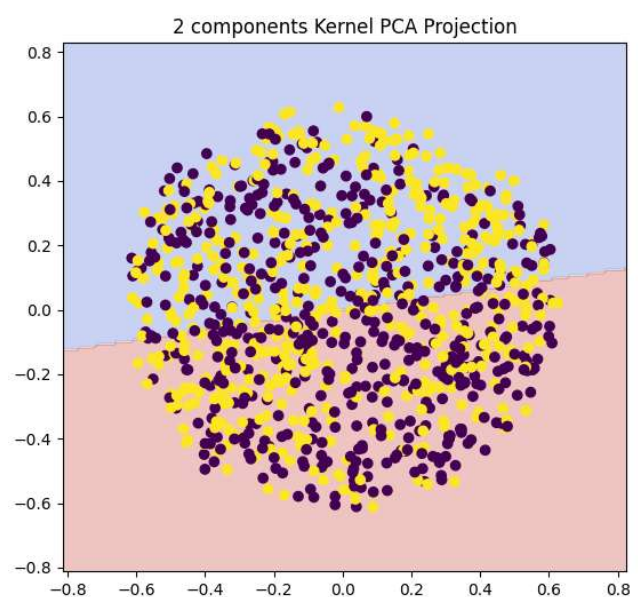
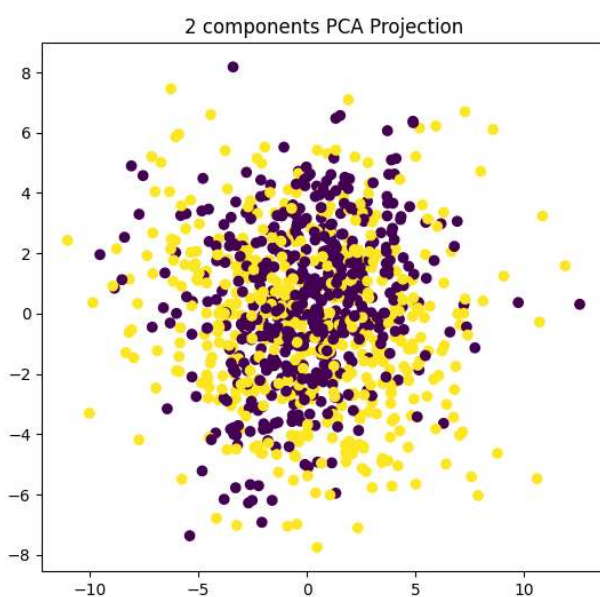


*Best gamma found: 2.35; Accuracy of SVC: 0.99*

Si nota subito come le due classi sono linearmente separabili nelle proiezioni trovate da Kernel PCA, che risulta quindi essere migliore rispetto al metodo senza kernellizzazione.

## PCA parte 2

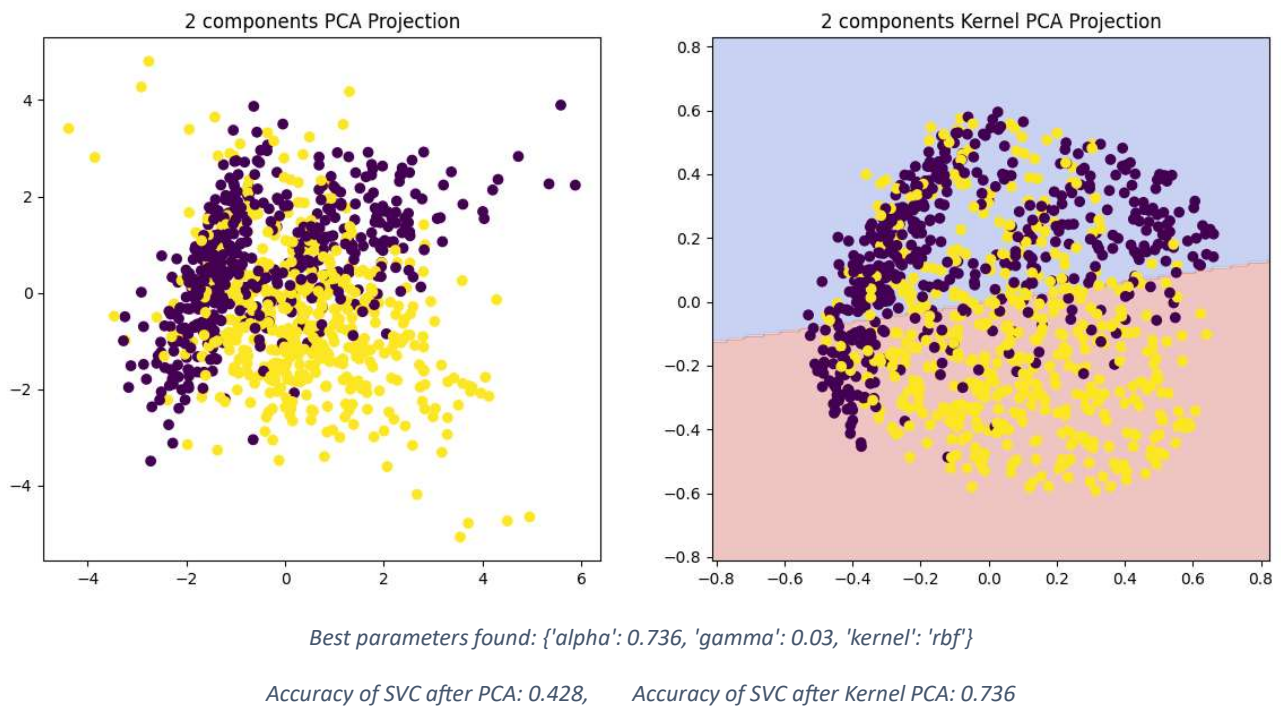
Lo scorso punto è stato ripetuto per un nuovo dataset contenente due classi, creato attraverso la funzione `make_classification`, in uno spazio a 10 dimensioni con 5 feature informative. La mancanza di informazioni riguardo alla geometria delle classi ha reso la ricerca del kernel giusto e degli iperparametri corretti molto più complessa, ottenendo anche dei risultati per nulla soddisfacenti:



*Best parameters found: {'alpha': 0.1, 'degree': 2, 'gamma': 0.05, 'kernel': 'rbf'}*

*Accuracy of SVC after PCA: 0.58, Accuracy of SVC after Kernel PCA: 0.424*

Pensando che lo scarso risultato possa essere legato ai pochi punti osservati in uno spazio alto dimensionale, è stato creato un altro dataset in uno spazio a 5 dimensioni per eseguire nuovamente questo confronto:



Il risultato della kernellizzazione in questo caso porta ad una separazione delle classi più efficace.

## CONCLUSIONI

La Kernellizzaione della Ridge Regression ha mostrato buone prestazioni sui dati non lineari, con il kernel appropriato, aumentando quindi le applicazioni possibili di questo metodo ben conosciuto.

Anche per la PCA La kernellizzaione ha portato a risultati migliori in presenza di dataset non lineari, permettend riduzioni di dimensionalità che portano a dati meglio separabili linearmente.

In generale però, la conoscenza della struttura del dataset risulta essere molto importante per la corretta scelta del kernel, che altrimenti risulta essere come la ricerca di un ago in un pagliaio. Anche la ricerca degli iperparametri sembra essere non banale, in quanto si tratta di una ricerca che prova ad essere esaustiva in tutte le possibili combinaizoni di parametri, e che dipende anch'essa da un processo trial and error per trovare la soluzione migliore possibile.