Data Mining 2, A.A. 2017/2018

Analisi di Time Series, Sequential Pattern mining, Classicazione e Outlier Detection

Francesco Cariaggi, Leonardo Cariaggi, Luciana Latorraca

Università di Pisa

Indice

Introduzione

Dataset utilizzati

IBM stock dataset: Valori delle azioni di IBM, raccolti (più o meno) quotidianamente in una Time Series in un arco di tempo di circa 50 anni. Dataset utilizzato per gli esperimenti nelle sezioni 2 e 3

UCI Abalone dataset: Insieme di misurazioni effettuate su 4177 abaloni. Dataset utilizzati per gli esperimenti nelle sezioni 4 e 5

Time Series

Obiettivi

- L'obiettivo degli esperimenti in questa sezione è individuare e studiare la similarità tra le Time Series
- Totale di 57 Time series annuali
- Primo approccio: *Autocorrelazione*

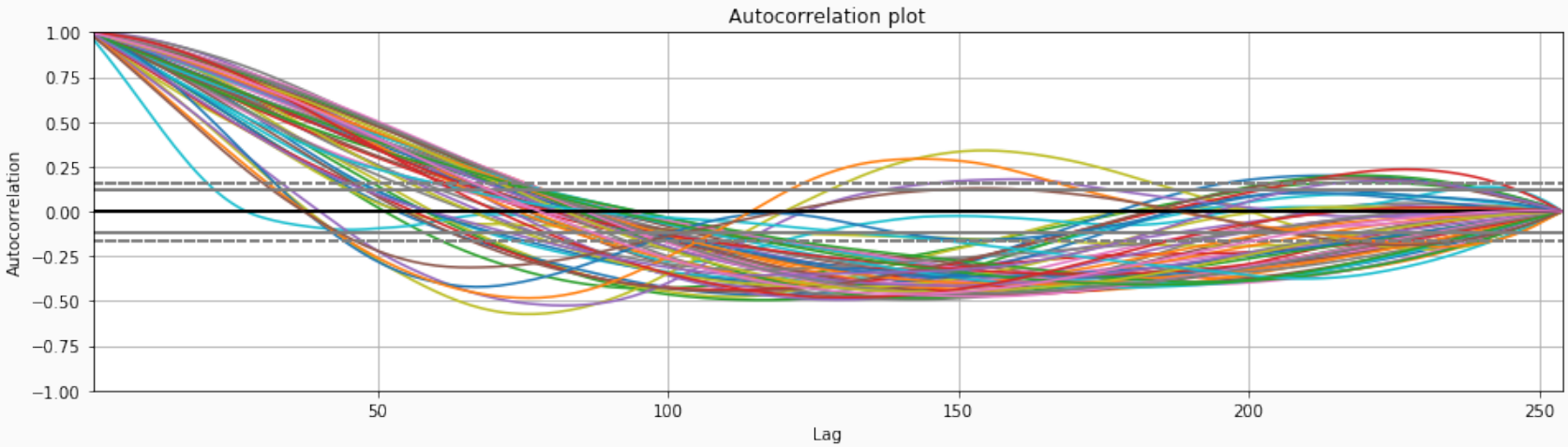
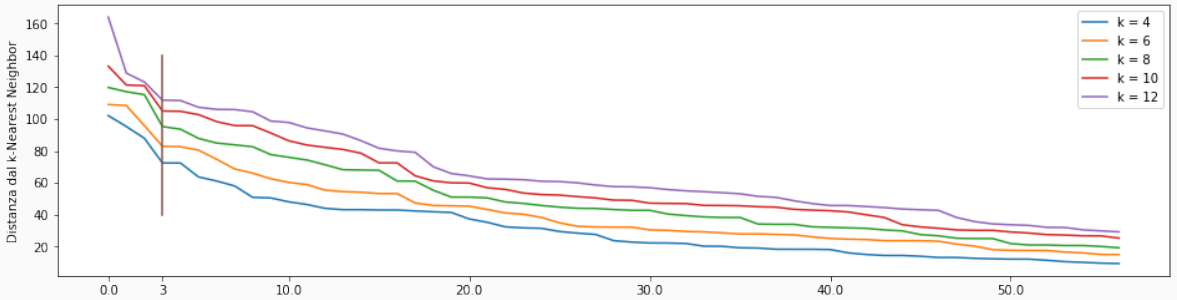


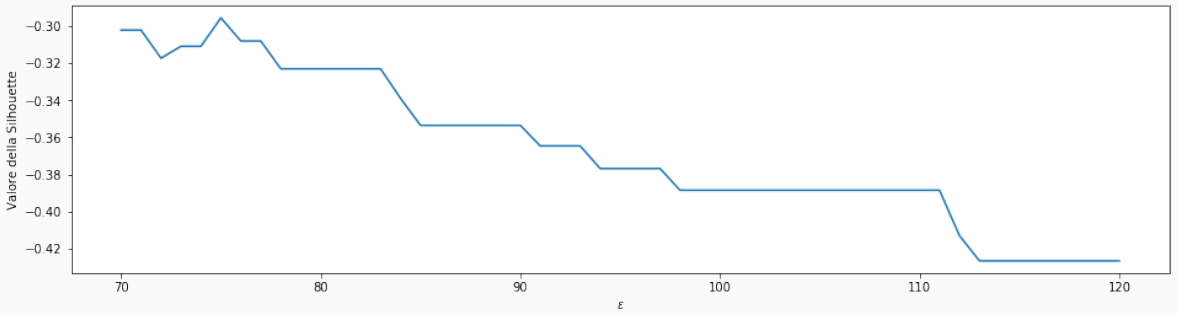
Figura 1: Grafico di autocorrelazione

Clustering con DBSCAN

DTW per le distanze tra le Time Series



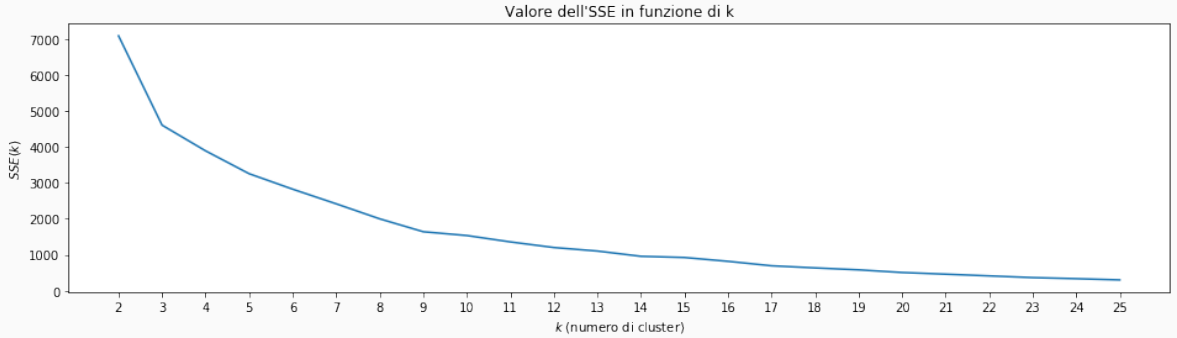
(a) Scelta del parametro *min_pts* e ϵ



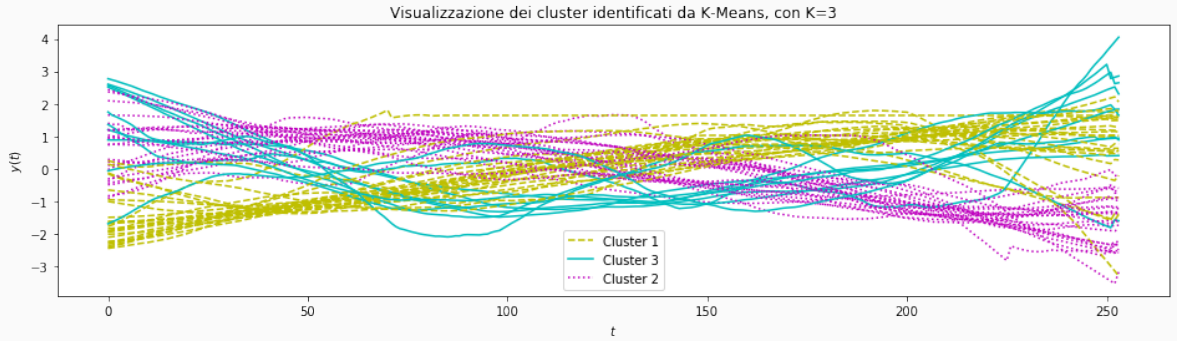
(b) Silhouette al variare di ϵ

Clustering con K-means

Risultati di K-means, con un numero di cluster $k = 3$ (scelto usando la tecnica visuale del ”punto di gomito”)



(a) SSE al variare di k



(b) Visualizzazione dei cluster individuati ($k = 3$)

Clustering (K-means) applicato a Feature Extraction

Feature extraction basata sulla trasformata di Fourier, applicata a K-means

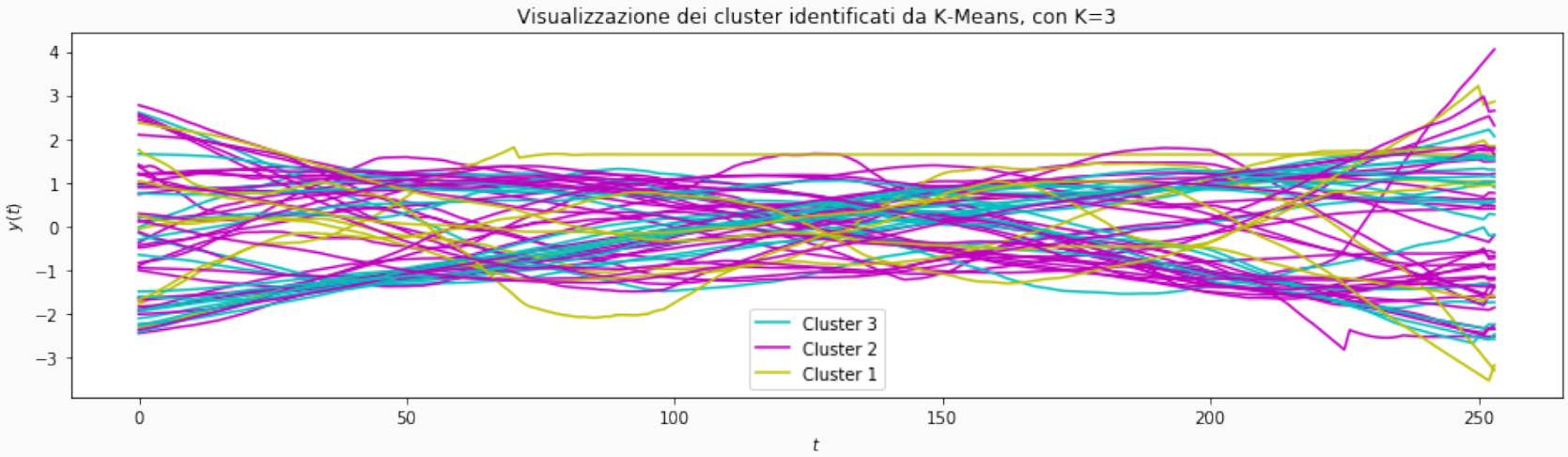


Figura 4: Visualizzazione dei cluster individuati con feature extraction basata sulla trasformata di Fourier

Sequential Patterns

Mappa degli esperimenti

- Obiettivo: trovare pattern di lunghezza $\geqslant 4$
- Discretizzazione anticipata (16 bins di egual misura)
  - con noise
  - senza noise
- Discretizzazione posticipata (8 bins di egual misura)
  - con noise
  - senza noise

Discretizzazione posticipata

1. Suddivisione della Time Series originale in 676 serie mensili;
2. Normalizzazione Z-score dei valori in ciascuna serie ottenuta;
3. Discretizzazione dei valori, utilizzando 8 bin di ampiezza fissa.

min_sup / Configurazione	**0.1**	**0.2**	**0.3**	**0.4**
Num. pattern (con noise)	11	0	0	0
Num. pattern (senza noise)	260	78	23	11

Tabella 1: Numero di *Sequential Pattern* trovati per diverse configurazioni

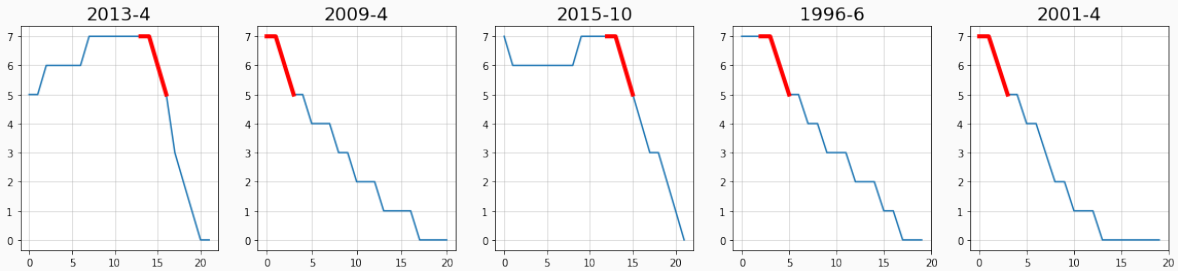


Figura 5: Visualizzazione del pattern [7,7,6,5] (Time Series con noise)

Discretizzazione anticipata

1. Discretizzazione dei valori della serie, utilizzando 16 bin di eguale ampiezza;
2. Suddivisione della serie originale in 676 serie mensili.

min_sup / Configurazione	0.1	0.2	0.3	0.4	0.5
Num. pattern (con noise)	40	21	17	12	5
Num. pattern (senza noise)	52	18	17	15	0

Tabella 2: Numero di *Sequential Pattern* trovati per diverse configurazioni

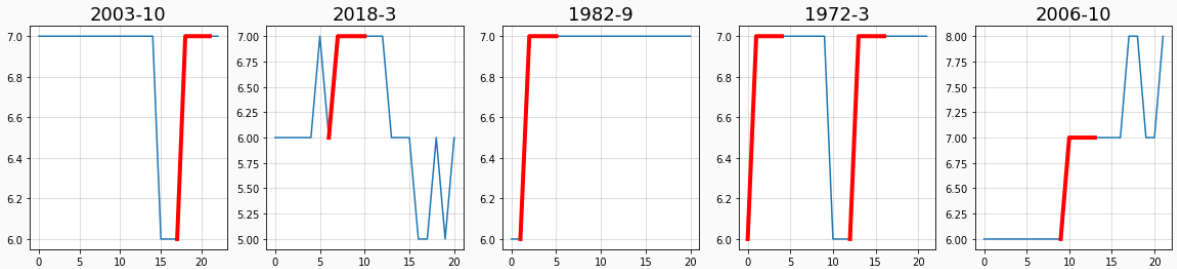


Figura 6: Visualizzazione del pattern [6,7,7,7,7] (Time Series con noise)

Classificazione

Mappa degli esperimenti

Nella parte di classificazione abbiamo sperimentato i seguenti modelli:

- Naïve-Bayes
- SVM
- Rete neurale
- Bagging (k-NN)
- Boosting (SVM)

Naïve-Bayes

Risultati ottenuti applicando un modello Naïve-Bayes. Si nota una certa variabilità tra i valori dell'accuratezza e dell'AUC nelle diverse fold

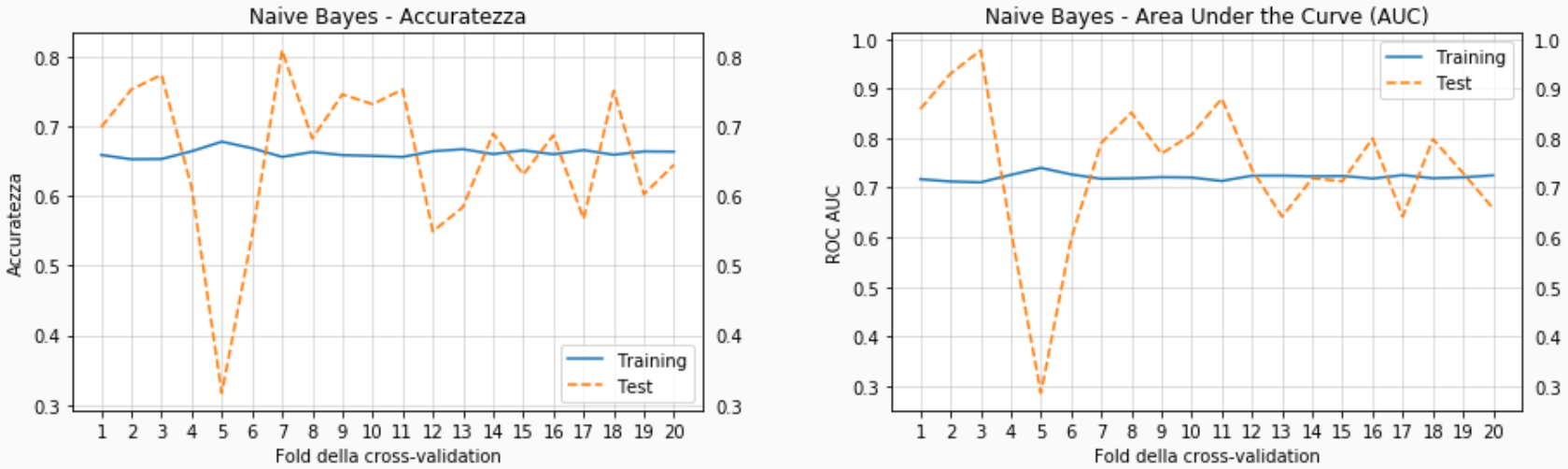


Figura 7: Risultati ottenuti con un modello Naïve Bayes

SVM e Rete neurale

Si mostrano i risultati ottenuti dopo la scelta delle migliori configurazioni di parametri. Ogni risultato è stato convalidato usando una *cross-validation*

	SVM	Rete neurale
Accuratezza media (Training)	0.768 ± 0.024	0.753 ± 0.022
Accuratezza media (Test)	0.769 ± 0.045	0.749 ± 0.026
AUC media (Training)	0.829 ± 0.022	0.810 ± 0.034
AUC media (Test)	0.820 ± 0.047	0.807 ± 0.040

Tabella 3: Risultati ottenuti con SVM e Rete neurale

Bagging (k-NN)

Risultati ottenuti applicando la tecnica di Bagging a k-NN, utilizzando 20 classificatori

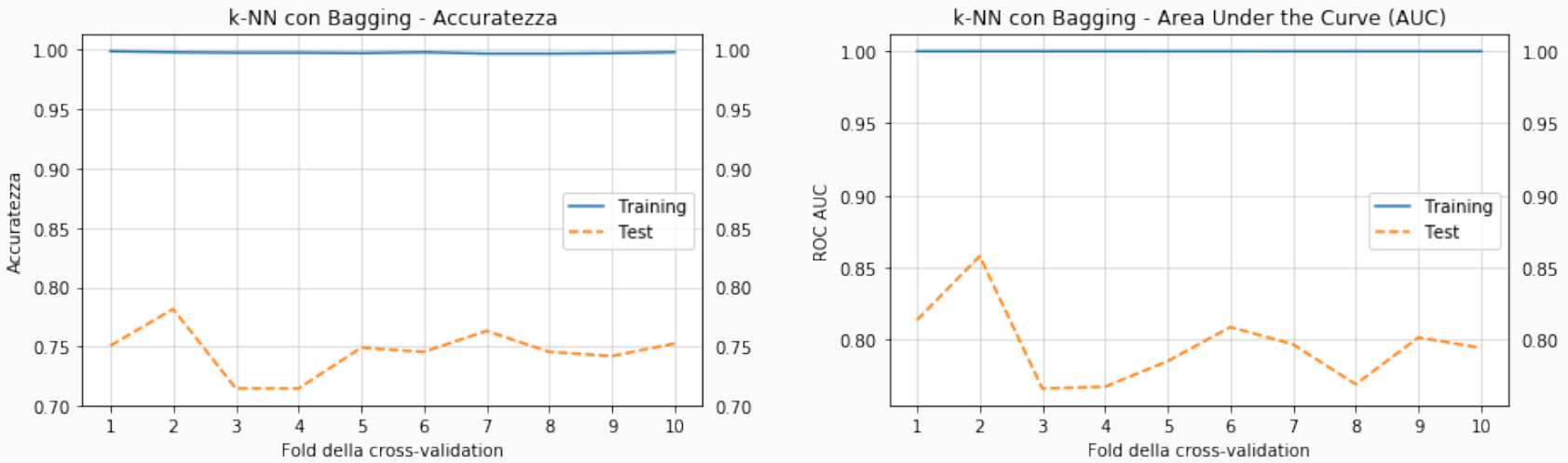


Figura 8: Risultati ottenuti con la tecnica di Bagging (k-NN)

Boosting (SVM)

Risultati ottenuti applicando la tecnica di Boosting a SVM, utilizzando 20 classificatori. Notare il fatto che i risultati sono **peggiori**

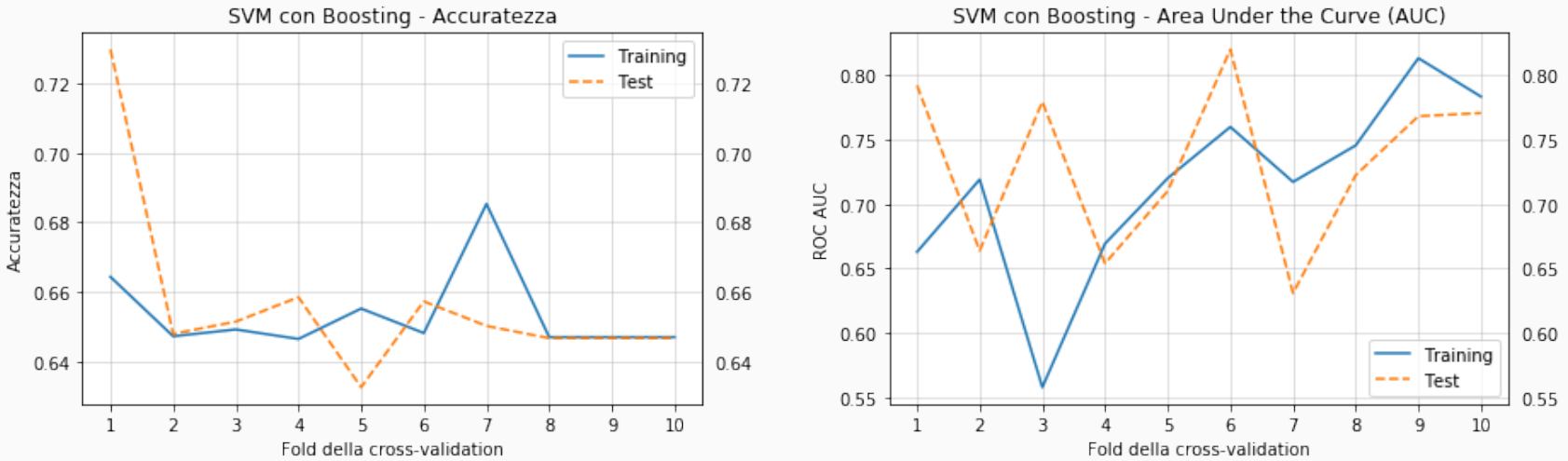


Figura 9: Risultati ottenuti con la tecnica di Boosting (SVM)

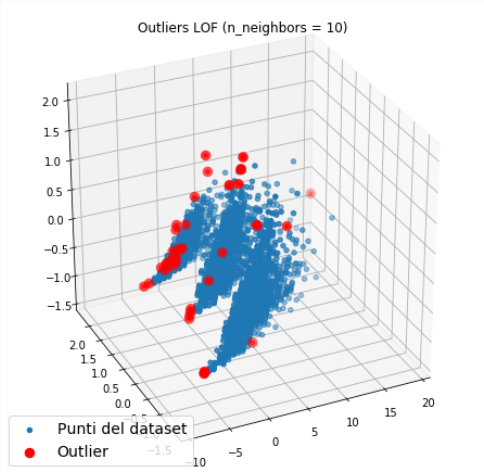
Outlier Detection

Mappa degli esperimenti

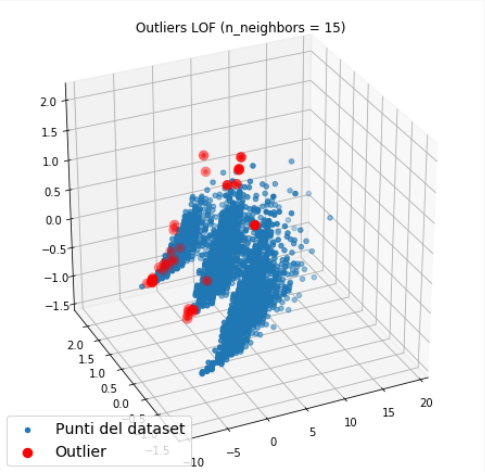
- Obiettivo: rilevare l'1% dei record nel dataset che ha la maggiore probabilità di essere un outlier
- Esperimenti condotti utilizzando tre tecniche di rilvamento degli outlier:
  - Local Outlier Factor (LOF)
  - DB(ϵ, π)
  - Approccio *depth-based*

Local Outlier Factor (LOF)

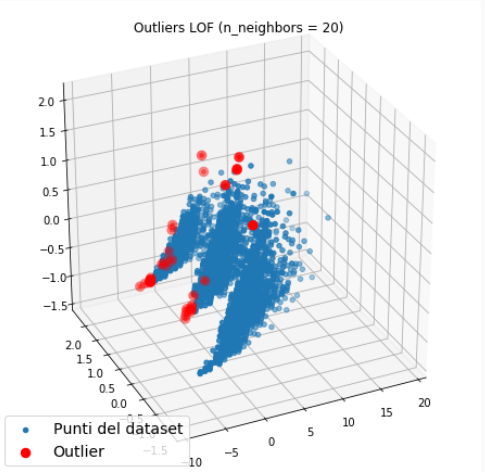
Approccio density-based, utilizzato per comparare la densità relativa di ogni punto con quella dei suoi vicini



(a) n_neighbors = 10



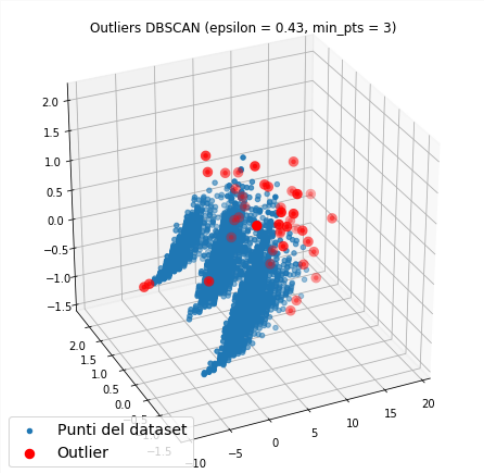
(b) n_neighbors = 15



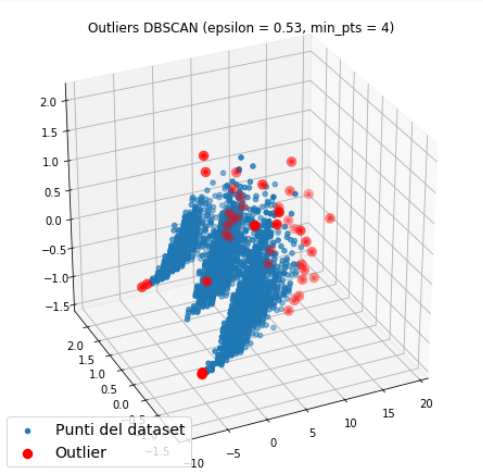
(c) n_neighbors = 20

DB(ϵ, π)

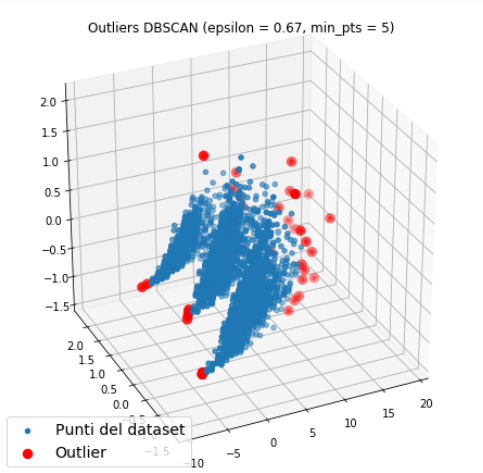
Approccio distance-based, che localizza gli outlier nelle aree a bassa densità



(a) min_pts = 3



(b) min_pts = 4



(c) min_pts = 5

Approccio depth-based

Organizzare i dati in diversi livelli di inviluppi convessi (*Convex Hull*) e localizzare gli outlier nei livelli più esterni

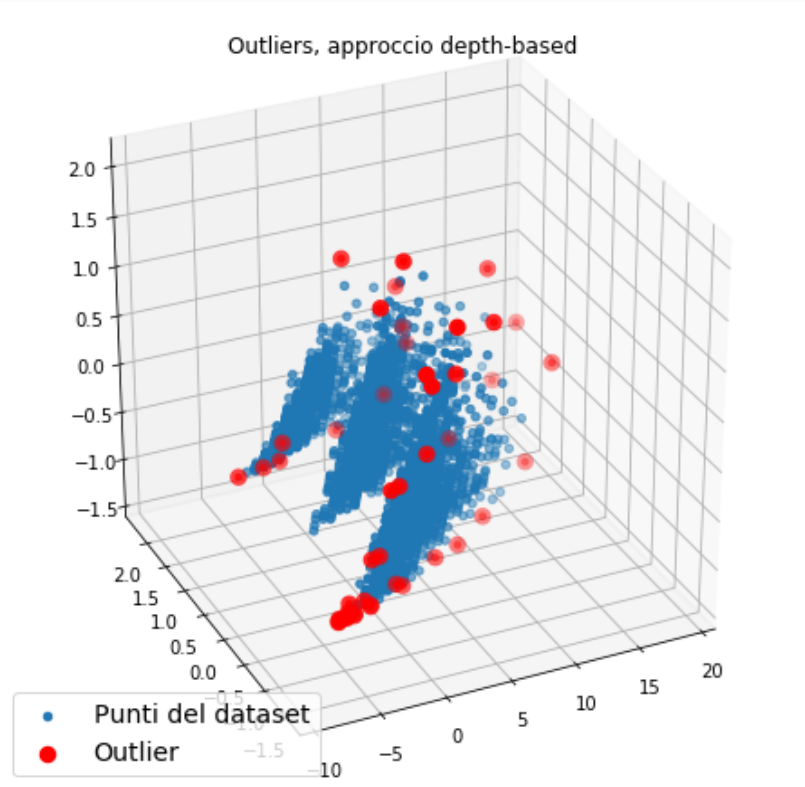


Figura 12: Outlier detection con approccio depth-based