**Academic Year:** 2023/2024

**Course:** Artificial Intelligent Systems - Intelligent Model Project

**Prof. Alfredo Milani**

**Ongoing assignmen**t: Machine Learning Basics

**Student:** Lorenzo Mariotti

**ID:** 369094

Sommario

[1. Dataset 3](#_Toc155307684)

[2. Classification 5](#_Toc155307685)

[2.1. Decision Tree 6](#_Toc155307686)

[2.2. Nearest Neighbor 7](#_Toc155307687)

[2.3. AdaBoost 8](#_Toc155307688)

# Dataset

**Link:**

<https://www.kaggle.com/datasets/fedesoriano/company-bankruptcy-prediction/data>

**Descrizione:**

Il dataset riporta i dati sulla bancarotta di aziende tratti dal *Taiwan Economic Journal* per gli anni 1999-2009. Il fallimento dell'azienda è stato definito in base al regolamento aziendale della Borsa di Taiwan.

**Attributi:**

Il dataset non presenta alcun attributo categorico quindi non è necessaria alcuna fattorizzazione.

Va tenuto però da conto della scala degli attributi, mentre la maggior parte di essi sono rappresentati da valori numerici compresi nell’intervallo [0, 1] altri come ad esempio:

* “Operating Expense Rate”,
* “Research and development expense rate”
* ”Cash flow rate Interest-bearing debt interest rate”
* …

sono rappresentati da valori compresi nell’intervallo [0, +Inf]. Tale differenza influenza in modo estremamente negativo classificatori come il K-NN pertanto si è deciso di ridurne la scala riportandola a dei valori congrui al resto degli attributi.

**Resize** **degli attributi:**

Lo script “*Preprocessing.cpp*” si occupa di scalare ogni attributo portando ogni suo valore nell’intervallo (0, 1).

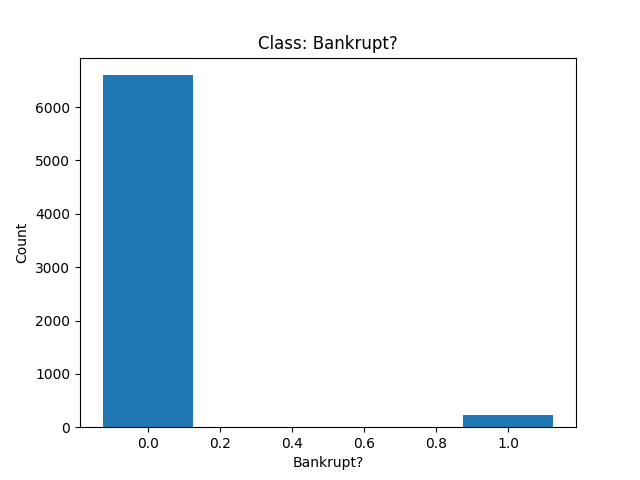
Pseudocodifica:

|  |
| --- |
| **for** each attribute **do**  m ← max(attribute)  **for** each x ⊆ attribute **do**  x = x / m |

**Classi:**

Le classi d’interesse sono rappresentate dalla colonna *“Bankrupt?”* che riporta con valori booleani se una data condizione ha portato o meno alla bancarotta dell’azienda.

**Distribuzione delle classi:**



* Numero di classi “0” = **6599**
* Numero di classi “1” = **220**

Uno sbilanciamento tale di classi rende metriche come l’accuratezza forvianti.

Per arginare questo sbilanciamento si è deciso di utilizzare due tecniche differenti:

* **Oversampling** dei dati in ingresso, tale tecnica ci permette di migliorare la rappresentazione delle classi con occorrenze minori
* **Assegnazione dei pesi**, il peso assegnato ad una classe ci indica la sua importanza delle fasi di campionamento ed è calcolato come segue:

Dove:

Wi = Peso della classe i-esima

N = Numero di campioni

Nc = Numero di classi

Ki = Occorrenze della classe i-esima nei campioni

Quindi:

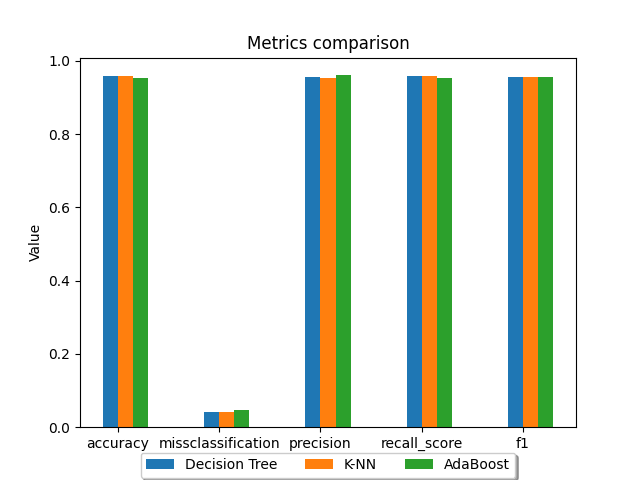
# Classification

**Dimensione del training set:** 75%

**Dimensione del testing set:** 25%

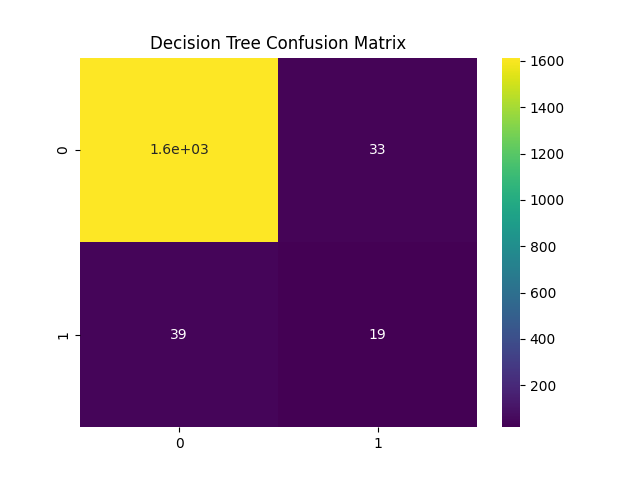
**Classificatori analizzati:**

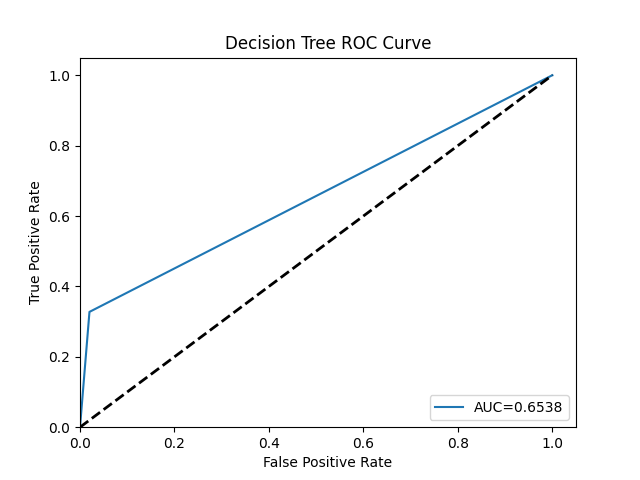
* Decision Tree
* Nearest-neighbor
* AdaBoost



## Decision Tree

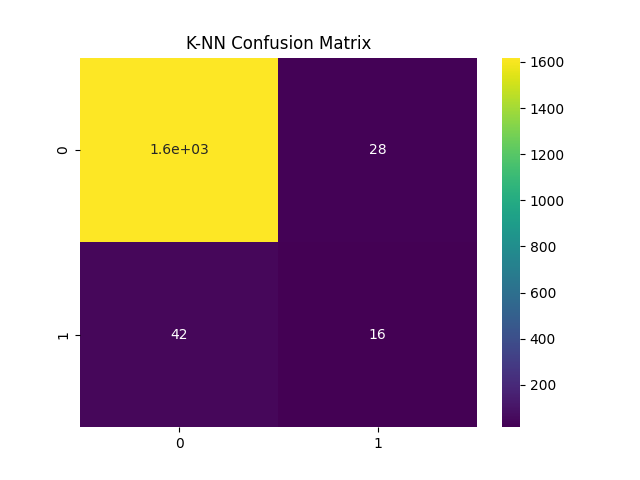
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Accuracy | Missclassification | Precision | Recall score | F1 |
| 0. 957771 | 0.042229 | 0. 955621 | 0. 957771 | 0. 956658 |

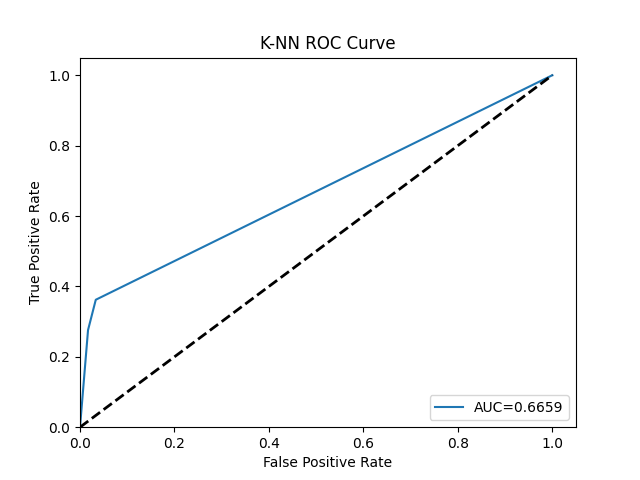




## Nearest Neighbor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Accuracy | Missclassification | Precision | Recall score | F1 |
| 0. 958944 | 0. 041056 | 0. 953927 | 0. 958944 | 0. 956214 |





## AdaBoost

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Accuracy | Missclassification | Precision | Recall score | F1 |
| 0. 953666 | 0.046334 | 0. 960552 | 0. 953666 | 0. 956792 |

