Università degli Studi di Milano - Bicocca
Scuola di Scienze
Dipartimento di Informatica, Sistemistica e Comunicazione
Corso di Laurea Magistrale in Informatica

Progetto Machine Learning

Riconoscimento della qualità del vino rosso

Alessandro Gherardi - 817084
Simone Giuseppe Locatelli - 816781

Anno Accademico 2020 - 2021

Indice

Introduzione

Per lo sviluppo dell'elaborato è stato scelto un dataset, disponibile al pubblico solo a scopo di ricerca, rappresentante un campione di vini rossi del nord-Portogallo, proposto da *Cortez et al. 2009*. Il dataset è stato recuperato utilizzando la piattaforma online Kaggle al seguente link. Il contesto in cui si applica questo particolare set di dati è quello di offrire una previsione sulle preferenze di gusto del vino che si basa su test analitici facilmente disponibili nella fase di certificazione.

L'obiettivo principale del progetto, perciò, è modellare la qualità del vino (buono/cattivo), sulla base di test fisico-chimici, implementando 2 diversi modelli di machine learning supervisionati: una SVM (Support Vector Machine) ed una Rete Neurale, confrontando le loro prestazioni utilizzando una 10-fold cross validation come metodo di valutazione, in modo da ottenere una stima della bontà di previsione dei modelli col minimo bias possibile.

Le metriche utilizzate per discriminare i modelli ottenuti saranno pertanto:

- Matrice di confusione complessiva.
- Accuracy, Precision, Recall, F-Measure.
- ROC e relativa AUC.

Analisi del Dataset

Il dataset è composto da 1599 istanze suddivise in 12 features o attributi, di cui l'ultima, la variabile target, è l'indicatore di qualità espresso da 0-10. Da una veloce analisi è possibile denotare come siano presenti solamente dati numerici, i quali appunto descrivono proprietà fisico-chimiche dei vini, mentre la variabile di target è rappresentata da valori interi. Nonostante la piccola dimensione, non sono presenti valori nulli o in altro formato tali da dover richiedere ulteriori pre-processing dei dati. Inoltre, come si evince dalla figura sottostante, il dataset originario pone un problema di classificazione multi-classe, le quali assumono valori da 3 a 8.

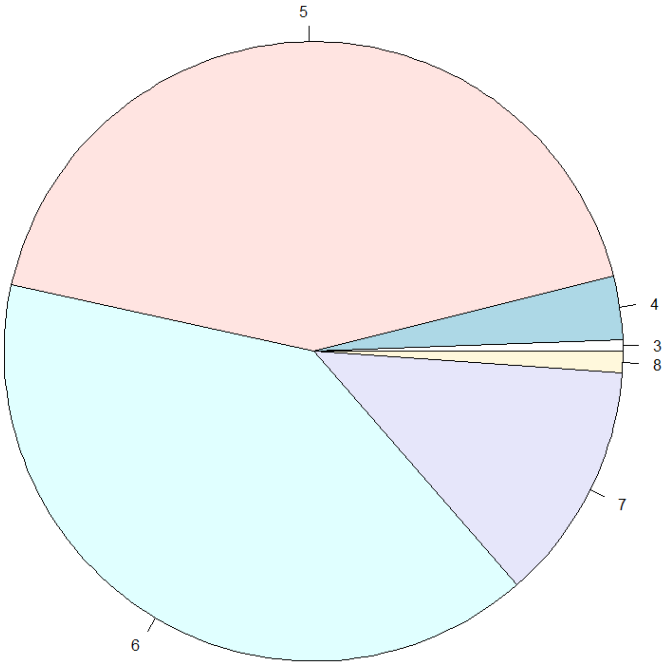


Fig. 1: Distribuzione della variabile di output.

La Fig.1 rende chiaro come il dataset sia fortemente sbilanciato sulle classi 5 e 6, e non presenti le classi: 0, 1, 2, 9 e 10. Dato questo fatto, unito all'obiettivo principale del progetto, ovvero stabilire se un vino è buono oppure no date le sue proprietà, è stato deciso di trasformare il problema di classificazione multi-classe ad un problema di classificazione binario.

Bilanciamento dataset

Data la scarsità di osservazioni per le classi 3,4,7 e 8, è stato deciso di passare da un problema multiclasse a uno binario. Per cui sono state ottenute due classi per la variabile target, assumendo che un vino buono abbia una qualità maggiore o uguale a 6. In questo modo il dataset sarà popolato da 744 osservazioni negative e 855 positive, ottenendo una distribuzione molto bilanciata visibile in Fig.2.

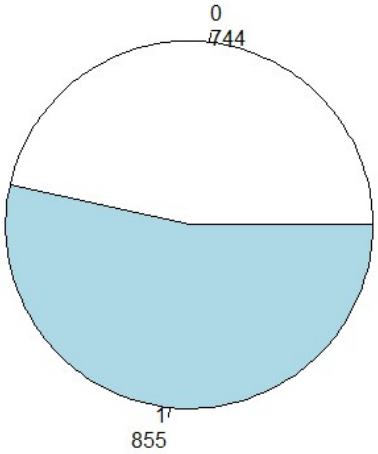


Fig. 2: Nuova distribuzione della variabile di output.

Analisi delle Covariate

Poichè si è trattato un dataset dalle dimensioni ridotte è stata scartata l'idea di eseguire una PCA per ridurre la dimensione dello spazio delle features su cui lavorare. L'esplorazione del dataset, quindi, è continuata analizzando le singole covariate (11), per decidere quali di esse potessero essere ridondanti o non utili alla scelta binaria. Questa decisione viene presa attraverso:

- Distribuzione delle covariate rispetto alla variabile di output (Fig.3)
- Matrice di correlazione (Fig.4)

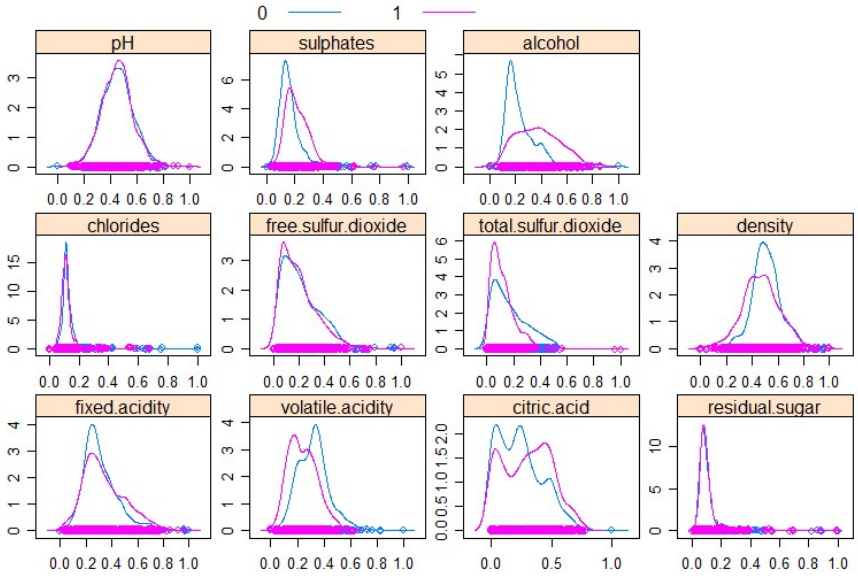


Fig. 3: Distribuzione delle covariate rispetto alla variabile di output.

Dal grafico utilizzato per effettuare la prima analisi sulle covariate possono essere già fatte alcune riflessioni. Ogni box in Fig.3 rappresenta le distribuzioni (2) dei vari attributi rispetto alla variabile di target, in blu quella relativa il target negativo e in rosa quella con target positivo. Si possono già estrapolare alcune covariate la cui distribuzione è praticamente identica per entrambi i target, cioè non sono estrememante utili per il task di classificazione. Esse sono:

- pH
- Chlorides
- Residual sugar
- Free sulphur dioxide

La seconda parte dell'analisi, invece, ha previsto l'utilizzo di una matrice di correlazione, visibile in Fig.4, in modo tale da fornire un ulteriore sogliatura sulle covariate da mantenere.

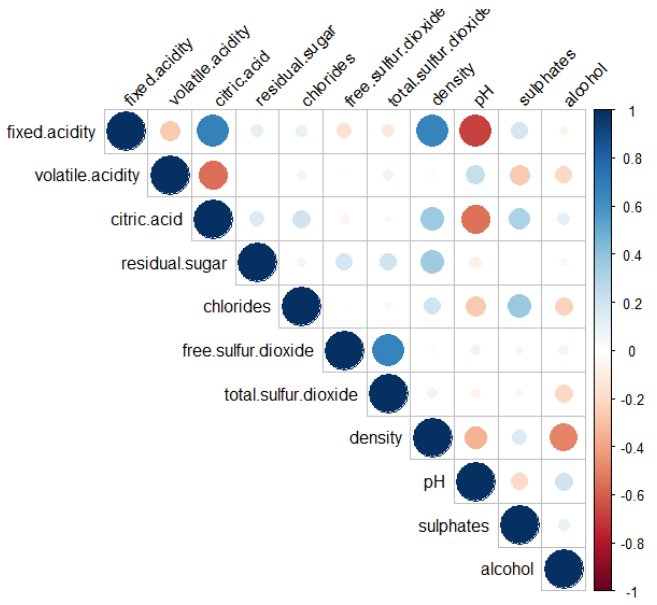


Fig. 4: Matrice di correlazione.

Nel grafico di cui sopra, ogni cerchio rappresenta la correlazione tra due covariate con valori da -1 (completamente scorrelate) a 1 (completamente correlate). La dimensione, invece, indica il valore di correlazione, più grande è e più il valore si avvicina a -1 o 1, mentre il colore indica il tipo di correlazione: rosso scorrelate e blu correlate.
Dall'analisi si individuano 3 correlazioni principali:

- Fixed acidity - Citric acid a valore **0.672**
- Fixed acidity - Density a valore **0.668**
- Free sulphur dioxide - Total sulphur dioxide a valore **0.667**

Analisi delle covariate a rischio eliminazione

Partendo dalle covariate singole, è stato deciso di analizzare le singole distribuzioni. Di seguito vengono espresse media, varianza e differenza interquartile delle singole covariate rispetto al target 1 o 0, per capire se nel task di classificazione, una certa feature possa risultare rilevante nella discriminazione delle due classi.

Covariata	Media	Varianza	Diff Interquartile
pH 1	0.449	0.0148	0.149
pH 0	0.450	0.0147	0.157
chlorides 1	0.118	0.00387	0.0342
chlorides 0	0.135	0.00867	0.0334
residual sugar 1	0.112	0.00952	0.0479
residual sugar 0	0.112	0.00911	0.0479
free sulfur dioxide 1	0.201	0.0199	0.190
free sulfur dioxide 0	0.219	0.0235	0.211

Tabella 1: Indici di dispersione delle distribuzioni delle covariate rispetto al target

Si può notare, come già visto in precedenza, che anche a livello matematico statistico le varie distribuzioni delle covariate rispetto al target 0 o 1 non differiscono di molto. In particolare le distribuzioni di *pH* e *residual sugar* sono praticamente identiche, a differenza di *chlorides*, la cui varianza differisce, o di *free sulfur dioxide*, che differisce sia per varianza che per differenza interquartile. Da ciò si evince che le covariate *pH* e *residual sugar* non portano vantaggio alla classificazione, poichè non differiscono nella rappresentazione del singolo target e quindi possono essere eliminate dal dataset.
Per l'analisi delle covariate correlate, invece, è stato deciso di analizzare le intere distribuzioni a coppie (visibili in Fig.5,6,7). In questo modo si può vedere se una covariate presenta dei valori che possono contenere la seconda, in modo da poterla sostituire.

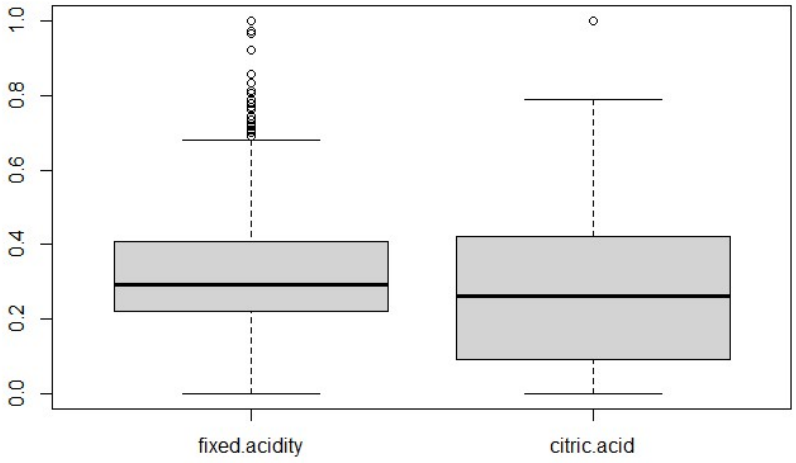


Fig. 5: Distribuzioni di fixed acidity e citric acid.

In Fig.5 si può notare come la distibuzione di *citric acid* contiene e esprime in maniera più ampia la distribuzione di *fixed acidity*. Infatti, la mediana e il primo quartile delle due distribuzioni sono praticamente identici. Da ciò è stato deciso di eliminare *fixed acidity* dal dataset, poichè ben rappresentato da *citric acid*. La prossima figura avrebbe dovuto rappresentare la coppia *fixed acidity* - *density*, ma per la decisione appena presa è stato sostituito con la coppia *citric acid* - *density*, anch'esse covariate correlate.

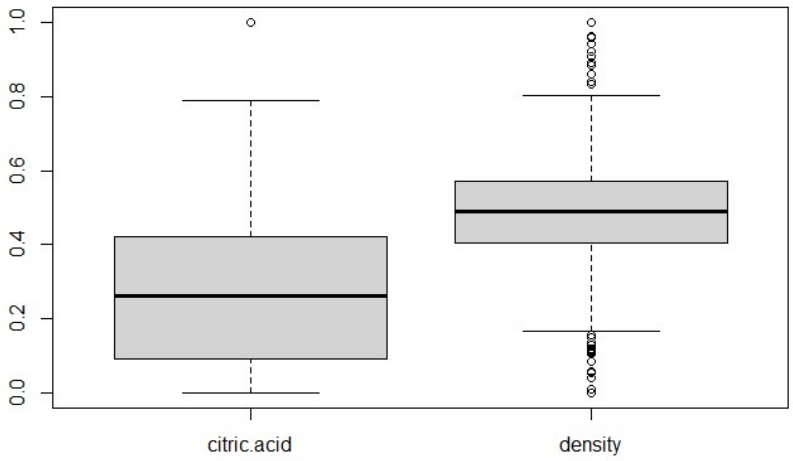


Fig. 6: Distribuzioni di citric acid e density.

Le distribuzioni in Fig.6, purchè correlate, hanno diverse mediane, diversi quartili e numero diversi outliers, perciò entrambe le covariate verranno mantenute nel dataset.

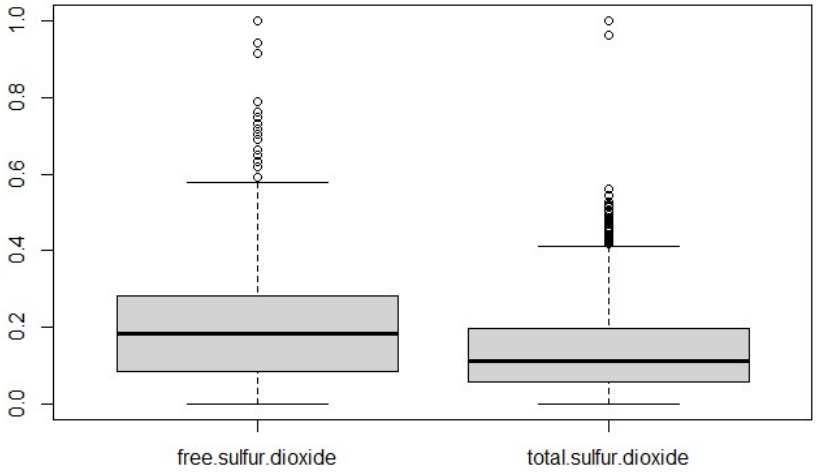


Fig. 7: Distribuzioni di free sulfur dioxide e total sulfur dioxide.

In Fig.7 si può notare come le due distribuzioni siano abbastanza similari, tanto da poter decidere di eliminare una delle due covariate. La scelta migliore sembrerebbe essere quella di man-

tenere *free sulfur dioxide*, ma la covariata in questione era già risultata poco rilevante all'analisi delle singole covariate, quindi è stato deciso di eliminarla in favore di *total sulfur dioxide*. Ricapitolando, le covariate eliminate sono:

- **pH**, poichè non rilevante per la classificazione.
- **residual sugar**, poichè non rilevante per la classificazione.
- **fixed acidity**, per correlazione con **citric acid**.
- **free sulfur dioxide**, poichè non rilevante alla classificazione e per correlazione con **total sulfur dioxide**.

Normalizzazione dataset

Infine, il dataset è stato normalizzato su valori nel range [0-1], in quanto le varie covariate, rispecchiando misure fisico-chimiche differenti, sono rappresentate su scale differenti. In questo modo è possibile anche ridurre l'impatto di eventuali outliers.

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{1}$$

SVM

Il primo modello di machine learning che è stato deciso di utilizzare per identificare un vino rosso è la Support Vector Machine. Una SVM può essere utilizzata sia per problemi di classificazione che di regressione. L'algoritmo si preoccupa di trovare gli iperpiani che separano due (o più) classi. Dopo averli trovati, cerca il miglior iperpiano attraverso i Support Vector, ovvero i punti più vicini all'iperpiano del set di dati, che crano un margine con l'iperpiano separatore.
Il motivo della scelta di una SVM risiede nella natura della variabile target, infatti l'algoritmo ottiene la massima efficacia nei problemi di classificazione binaria. Un altro motivo è la dimensione stessa del dataset, infatti la SVM riesce ad ottenere un buon grado di accuratezza anche senza un numero enorme di istanze di addestramento. Infine, l'utilizzo dei metodi kernel risulta utile per elaborare problemi con covariate non linearmente separabili, come nel caso in esame, mappandole in uno spazio a dimensione superiore, al fine di trovare l'iperpiano che le separa. Per tale scopo sono state utilizzate varie funzioni di trasformazione, al fine di capire quale kernel si adattasse meglio ai dati in esame. La funzione scelta è stata, infine, la Radial Basis Function:

$$K(x, x') = exp\left(- \frac{\|x - x'\|^2}{2\sigma^2} \right) \tag{2}$$

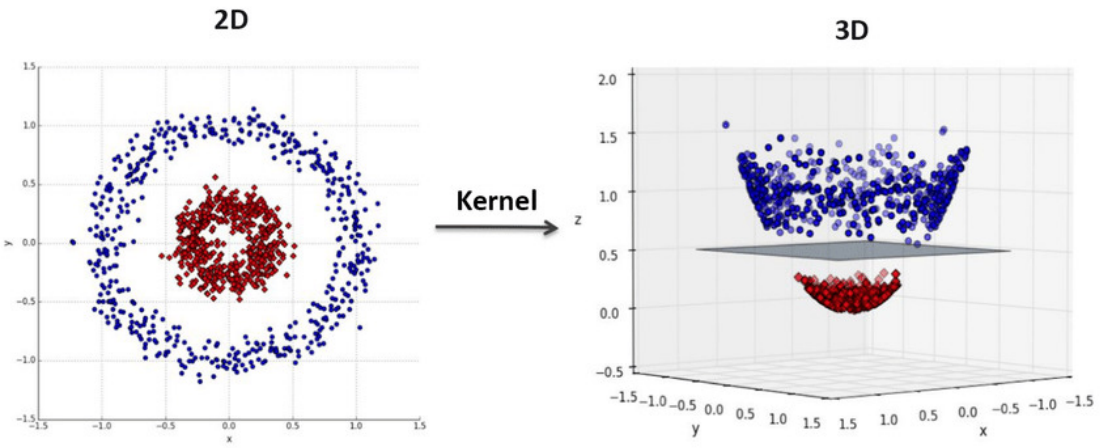


Fig. 8: Trasformazione mediante un kernel non lineare.

In Fig.8 si può vedere un esempio di come funziona un kernel non lineare. Viene preso uno spazio con variabili non linearmente separabili in R^2 e vengono mappate in uno spazio di dimensione maggiore. In questo modo le variabili in R^3 riescono ad essere separate da un iperpiano parallelo agli assi X e Y.

Rete Neurale

Il secondo modello di machine learning che è stato impiegato per identificare la qualità di un vino rosso è una Rete Neurale. Il motivo di questa scelta risiede nella tipologia del dataset, a valori numerici, e con ogni vettore, rappresentante una particolare istanza, etichettato. Perciò è possibile indurre modelli supervisionati, quali appunto la NN. Una NN si distingue per la sua capacità di generalizzare, ovvero può inferire relazioni tra dati mai visti dopo aver appreso pattern determinanti. Oltre a questo, una rete neurale si presta bene a problemi di classificazione (in questo caso binario), dove normalmente si hanno molte istanze su cui allenarsi, perciò questo può anche essere visto come un caso limite dove la particolare architettura viene stressata per ottenere un buon grado di accuratezza, nonostante la limitatezza dei dati a disposizione.

Dato il pre-processing effettuato sul dataset e la selezione delle covariate più esplicative, il nostro modello di rete neurale prevede 7 neuroni di input, un unico strato nascosto con 3 neuroni, ed infine 2 neuroni di output.

La struttura è visibile in Fig.9.

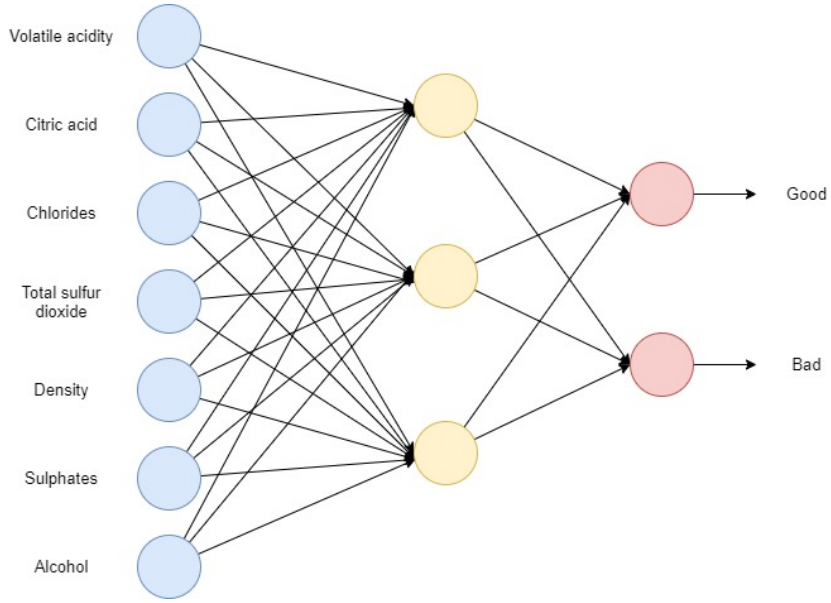


Fig. 9: Struttura della rete neurale.

La funzione di attivazione scelta per questo particolare tipo di problema è la funzione logistica:

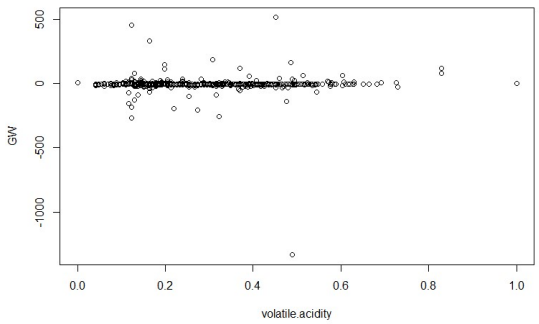
$$f(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

Essa viene molto utile in problemi di classificazione binari ed inoltre comprime il nostro range di valori nell'intervallo [0-1]. Mentre come funzione per calcolare l'errore su una particolare istanza viene adottato l'errore quadratico medio:

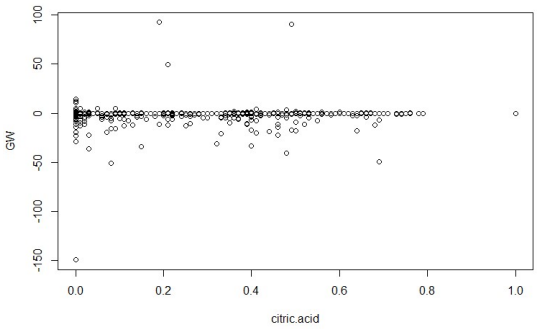
$$SSE = \sum_{i=1}^{n} (y_i - \overline{y_i})^2 \tag{4}$$

L'esempio prende in considerazione il modello migliore (accuratezza ottenuta) allenato durante la 10-fold cross validation. La rete converge ad una soluzione ottima dopo 13007 iterazioni.

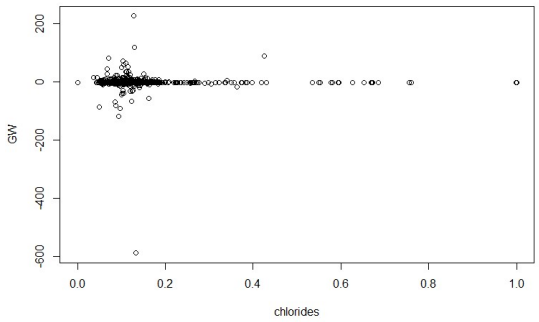
Un'ulteriore analisi può essere fatta sui pesi generalizzati della rete: essi ci danno informazioni utili circa l'effetto di ciascuna covariata rispetto la predittività del modello. In Fig.10 possiamo analizzare i pesi generalizzati per ogni covariata utilizzata dal modello: possiamo notare che generalmente i predittori utilizzati tendono ad avere un effetto lineare, soprattutto nel caso di *Volatile acidity* e *Alcohol* i quali risulteranno avere poco impatto durante l'apprendimento del modello.



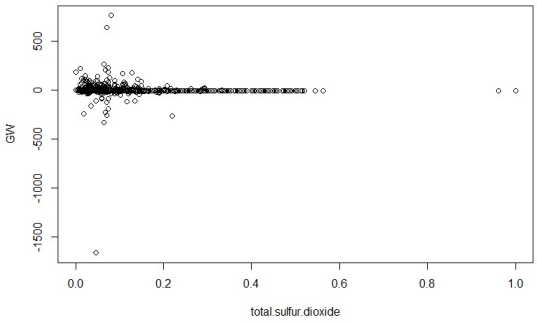
Volatile acidity



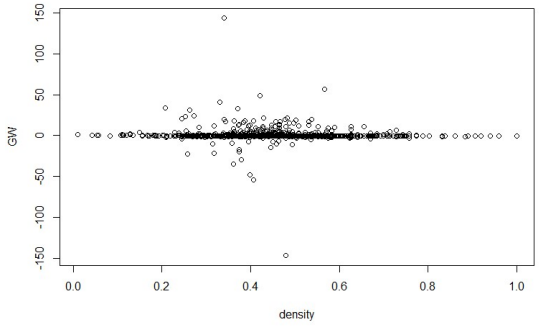
Citric Acid



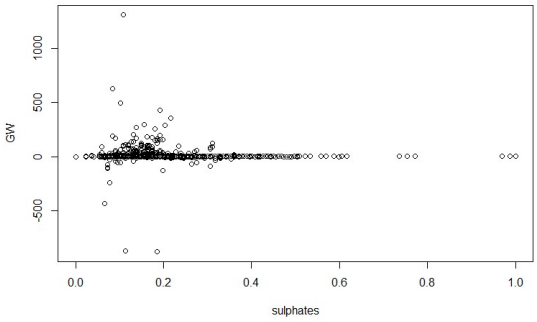
Chlorides



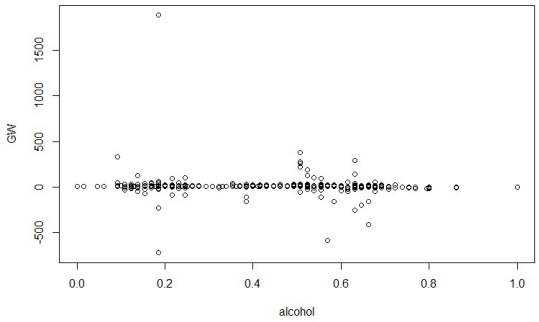
Total sulfur dioxide



Density



Sulphates



Alcohol

Fig. 10: Pesi generalizzati per ogni attributo usato dalla rete neurale.

Analisi dei risultati

Gli esperimenti condotti sulla SVM e sulla NN hanno previsto l'utilizzo di una 10-fold cross validation come metodo di valutazione dei modelli. Questo dovuto anche alla dimensione ridotta del dataset. Per ottenere risultati coerenti e comparabili, sono state mantenute le medesime suddivisioni del dataset su cui indurre i modelli.

Poichè la 10-fold cross validation produrrà 10 modelli diversi, i risultati ottenuti ad ogni iterazione sono infine stati mediati: ovvero è stata estrapolata una matrice di confusione totale, e tramite essa sono state calcolate le misure di: Accuracy, Precision, Recall e F-Measure.

Di seguito vengono riportate le matrici di confusione ottenute:

	Bad	Good
Bad	557	187
Good	210	645

Tabella 2: Matrice di confusione complessiva SVM.

	Bad	Good
Bad	538	206
Good	198	657

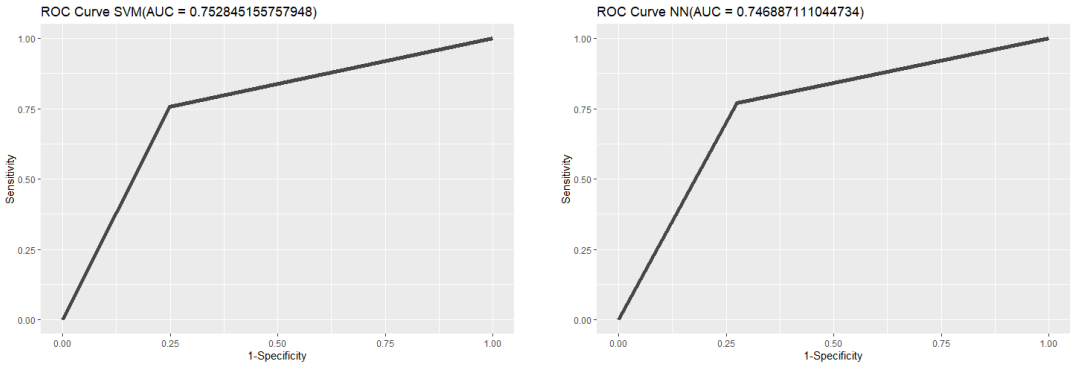
Tabella 3: Matrice di confusione complessiva NN.

Di seguito, invece, sono riportate le misure estratte dalle matrici di confusione:

	Accuracy	Precision	Recall	F-Measure
SVM	**75.17**	**77.52**	75.43	76.46
NN	74.73	76.12	**76.84**	**76.48**

Tabella 4: Misure di performance per modello.

È possibile notare che, tendenzialmente, non abbiamo un modello che sovrasta l'altro, sia le matrici di confusione complessive sia le misure riportate sono molto simili, per cui non è possibile decretare quale modello sia effettivamente migliore. A conferma di ciò, di seguito, vengono riportate le curve ROC ottenute dai 2 modelli (avendo effettuato una 10-fold CV la sensitività e la specificità sono state mediate).



SVM NN

Fig. 11: Curve ROC.

Anche in questo caso è possibile notare come i risultati siano praticamente indistinguibili, abbiamo una differenza di 1 punto sulle AUC dei modelli a favore della SVM. Di seguito, invece, sono riportati i boxplot raffiguranti la sensitività e la specificità dei modelli: notiamo come mediana e quartili siano praticamente identici anche in questo esperimento.

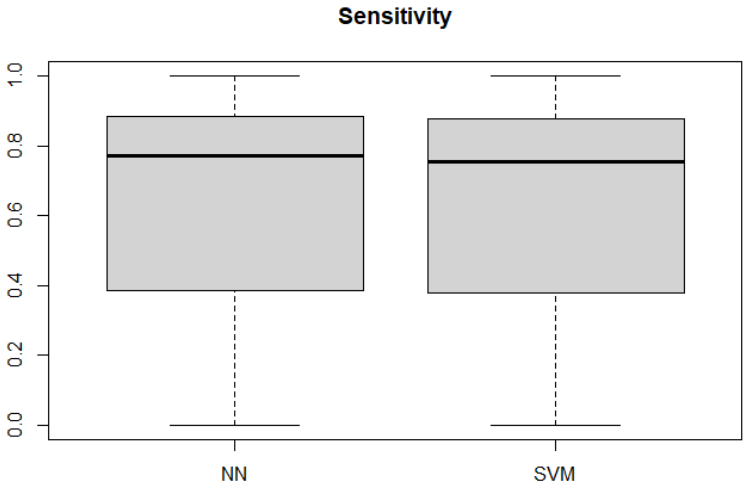


Fig. 12: Analisi sensitività.

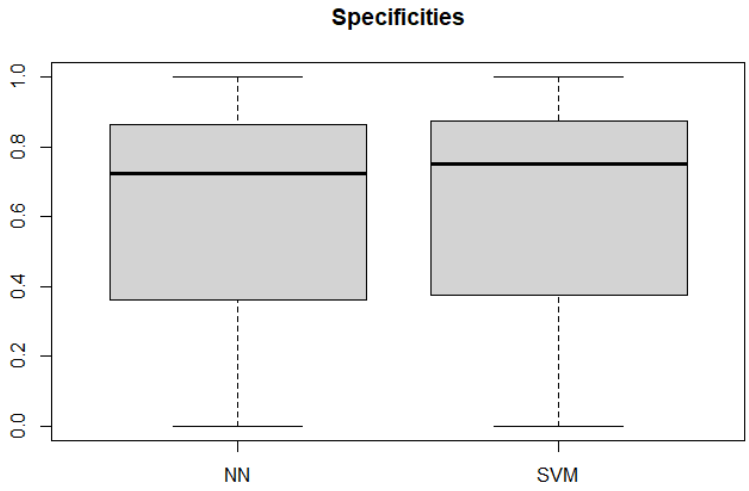


Fig. 13: Analisi specificità.

Infine, una comparazione può essere fatta anche a livello di tempi di allenamento dei singoli modelli. Otteniamo, mediamente, che la SVM risulta molto più veloce da allenare rispetto alla rete neurale, rispettivamente abbiamo circa 0.2 secondi contro un range che varia dai 2.4 secondi ai 16 secondi (questa differenza è dovuta al fatto che in determinate configurazioni la rete converge ad una soluzione ottima in meno step).

Questo, considerati i risultati ottenuti precedentemente, può essere un fattore discriminante nella scelta di quale modello confa meglio alla situazione esaminata.
Infine, si è voluto testare se effettivamente le covariate rimosse fossero ridondanti, perciò sono stati effettuati gli stessi esperimenti utilizzando il dataset intero (a 11 covariate). I risultati ottenuti indicano generalmente una differenza dell'1%, appurando quindi come gli attributi scartati non portino informazioni aggiuntive utili al computo, ed anzi, nel caso della Rete Neurale utilizzare meno covariate si rispecchia in un training generalmente più veloce. I risultati sono riportati qui di seguito.

	Accuracy	Precision	Recall	F-Measure
SVM	76.04	78.36	76.25	77.29
NN	75.1	76.47	77.19	76.83

Tabella 5: Misure di performance per modello utilizzando il dataset intero.

Conclusioni

È stato affrontato il problema della classificazione della qualità dei vini rossi sulla base di proprietà fisico-chimiche. Per fare questo è stato utilizzato il dataset proposto da Cortez et al. 2009. Esso ci ha permesso di indurre due diversi modelli di machine learning supervisionati, al fine di valutarne le prestazioni e di individuare quello che meglio si confa alla particolare situazione. In seguito ad un'analisi preliminare sul dataset, è stato effettuato un leggero pre-processing, dove sono state eliminate alcune covariate ridondanti ed è stato trasformato il problema di classificazione da multi-classe a binario.

Gli esperimenti, viste le dimensioni ridotte del dataset, sono stati condotti utilizzando una 10-fold cross validation come metodo di valutazione dei modelli utilizzati: una SVM ed una Rete Neurale. Per la Support Vector Machine, vista la non linearità del nostro spazio delle features, è stato utilizzato un kernel radiale, mentre la Rete Neurale è stata progettata con un solo layer nascosto costituito da 3 neuroni e 2 neuroni di output, visto il problema di classificazione binario.

I risultati ottenuti dalle matrici di confusione complessive, estratti dalla 10-fold cross validation, confermano come i due modelli qui proposti si equivalgono in termini delle misure di performance utilizzate. Un fattore determinante che può essere preso in considerazione è il tempo di training degli stessi: si vede perciò come la SVM sia superiore in questi termini, ottenendo tempi migliori e più costanti rispetto alla Rete Neurale di un'ordine di grandezza.