CarVana – Don’t get kicked!: evitare le frodi in ambito di aste sui motori

Progetto Data mining & Machine Learning

Lorenzo Basile, Ada Maragno, Giangaetano Pachera, Gianluca Sperduti

Abstract

Il dataset CarVana, composto da 72983 record ognuno caratterizzato da 33 attributi, contiene informazioni su macchine usate acquistate in aste di rivendita. Alcune di queste macchine si sono rivelate dei “limoni”, cioè degli acquisti di pessima qualità. Il nostro obiettivo è stato quello di cercare di comprendere il dataset in fase di data understanding, di pulirlo e settarlo per operazioni future (data engineering), di cercare dei cluster validi con delle operazioni di clustering e, infine, di individuare il classificatore migliore che potesse discernere le macchine-frodi dai buoni acquisti. I nostri risultati sono stati relativamente soddisfacenti con un clustering K-means a 6 centroidi sulle variabili numeriche e con una classificazione con tecnica di Random Forest, che ci ha fornito risultati predittivi con 58% accuracy e gini index a 24. Inoltre, è stato fatto anche un modello di pattern mining con i valori del dataset.

1. Data description, cleaning, understanding, engineering

1.1 Data description

Il dataset Don’t get Kicked ha al suo interno 32 variabili indipendenti e una variabile dipendente. Ci è stato fornito anche un test set con 48 mila istanze con la variabile dipendente assente. La maggior parte dei dati è in forma categoriale e una minoranza in forma numerica.
In primis, nel dataset c’è una variabile RefID che indica l'id di ogni veicolo. La nostra variabile dipendente è IsBadBuy, che ci indica quali macchine o meno sono state degli acquisti di scarsa qualità. PurchDate identifica la data di acquisto del veicolo. Auction identifica la casa d'asta dalla quale quest’ultimo è stato acquistato. VehYear ne indica l'anno di produzione ed è molto correlata con la variabile VehicleAge (che ne mostra l'età). Make, Model, Trim e SubModel sono quattro variabili relative al modello della macchina, della quale descrivono brand ed equipaggiamento. La variabile Color identifica il colore dell’automobile. La variabile WheelType ne spiega il tipo di cerchi e presenta una corrispondenza esatta con la variabile WheelTypeID. La variabile VehOdo rappresenta il conteggio dell'odometro dell'automobile, che misura quanti chilometri ha percorso l'auto nella sua vita. In Kaggle è segnalato che questo valore è spesso falsato nelle auto vendute all'asta, per cui questa variabile deve essere utilizzata con cautela. La variabile VehBcost è il prezzo a cui l’automobile è stata acquistata.
Nationality ne rappresenta invece la nazionalità. Transmission riguarda il cambio dell’auto. TopThreeAmericanName spiega se la macchina fa parte o meno di una delle 3 più grandi case automobilistiche americane.
Size ne rappresenta la dimensione.

PRIMEUNIT & AUCGUART sono variabili che caratterizzano il veicolo se ambito (Primeunit) e se coperto da garanzia (Aucguart). Le variabili di tipo MMR rappresentano la valutazione del prezzo del veicolo secondo la rivista specializzata MMR.
IsOnlineSale indica se l'auto è stata acquistata online o meno. WarrantyCost è invece il prezzo della garanzia dell'auto. BYRNO è l'id del compratore d'auto.

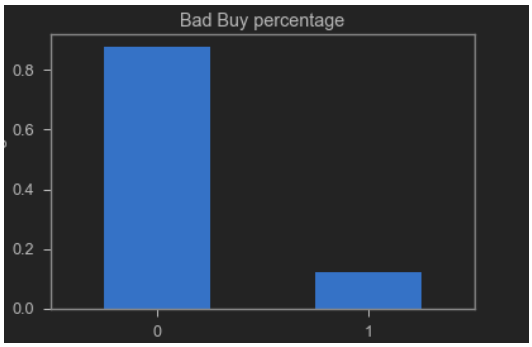
1.2 Data cleaning

Prima di addentrarci nel cleaning dei valori nulli e doppi, abbiamo optato per l'eliminazione di alcuni attributi che abbiamo ritenuto poco importanti ai fini dell'analisi: RefID e BYRNO, perché ID che non avremmo utilizzato, PurchDate che non abbiamo ritenuto informativa in nessun modo, VehYear perché strettamente correlata con VehicleAge ma di più difficile lettura, Color perché non ci dava alcuna informazione sul funzionamento del veicolo, WheelTypeID perché strettamente correlata con WheelType. PRIMEUNIT e ACGUART perché con troppi valori nulli, IsOnlineSale perché non aggiungeva informazione importante in relazione ai BadBuy. Anche TopThreeAmericanName, perché poco informativa data la presenza di Make e Model.

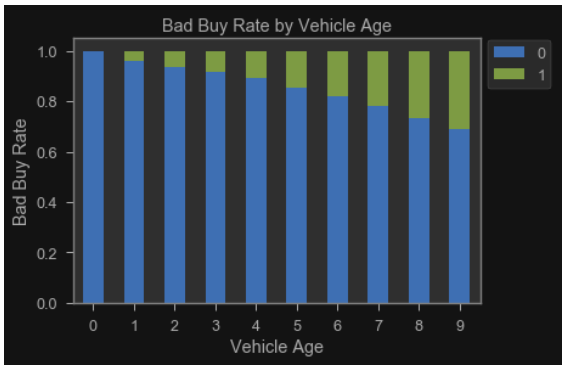
Per la pulizia dei valori, nulli, invece, abbiamo optato per strategie differenziate in base alla variabile. Per Nationality, con 5 valori nulli, abbiamo osservato che ad ogni Make corrisponde una sola nazionalità e abbiamo quindi sfruttato questa informazione. Per quanto riguarda la pulizia dei valori nulli di Trim, che sono più di 2000, la sfida è stata più grande. Abbiamo cercato di assegnare a ogni veicolo il trim del veicolo più simile a lui per Make, Model e Submodel, facendo raggruppamenti stratificati in base alle informazioni disponibili. Proprio perché il modello della macchina contiene informazioni tendenzialmente precise non solo sul trim, la stessa strategia è stata applicata ai valori nulli per Transmission, WheelType, Size e MMR.

1.3 Data understanding

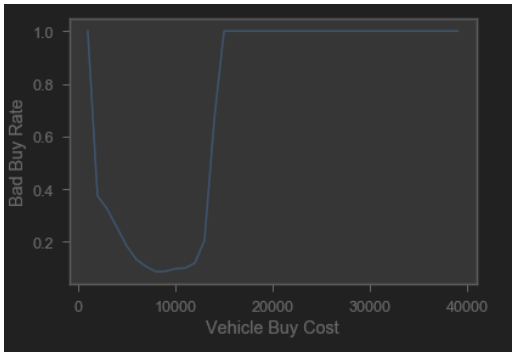
Le difficoltà della fase di data understanding, in questo caso, sono state legate alla poca conoscenza del dominio da parte dei membri del nostro gruppo. È stata quindi investita una discreta quantità di tempo e attenzione allo studio dei significati delle variabili e della loro utilità. Una prima considerazione importante da fare è che il dataset appare poco bilanciato per quanto riguarda la distribuzione della variabile dipendente. Questo tipo di situazione ci pone già di fronte a un problema di sbilanciamento del campione molto grande, soprattutto per la fase della classificazione. Il tasso di BadBuy si attesta solamente intorno al 12%.



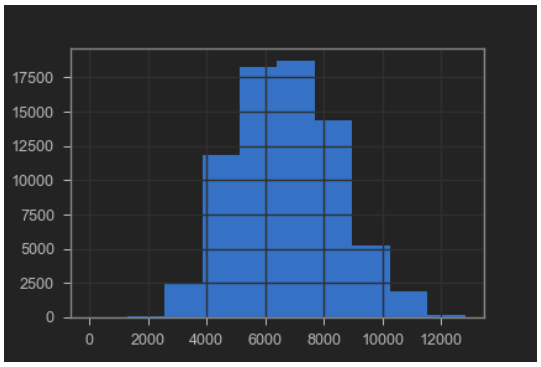
Abbiamo poi cercato di osservare quali fossero le variabili più influenti per quanto riguarda la presenza di BadBuy. Abbiamo notato che con l'avanzare dell'età la percentuale di BadBuy cresce proporzionalmente.



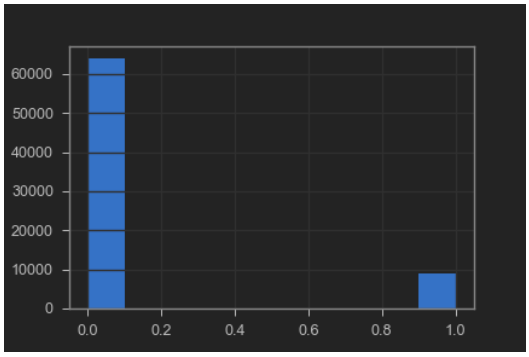
Un'altra variabile molto influente sembra essere il prezzo del veicolo, per la quale i Bad Buy si distribuiscono in forma concava.



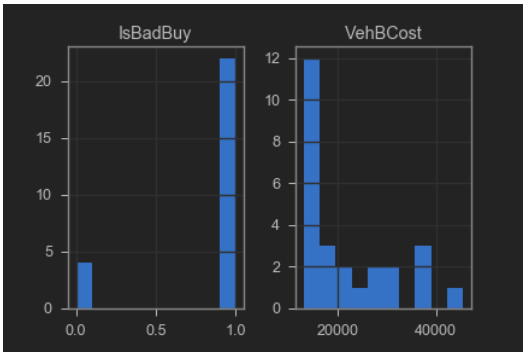
Abbiamo osservato gli outlier per una serie di variabili. La più rilevante in termini di analisi ci è sembrata quella del costo del veicolo. La distribuzione della variabile senza outlier è simile a quella di una normale:



In questo contesto i Bad buy sono distribuiti pressoché come all'interno di tutto il dataset:



Abbiamo poi osservato gli outlier in modo isolato e per quanto riguarda loro, invece, la situazione si capovolge. I BadBuy sono in grande maggioranza e la variabile del prezzo assomiglia a quella di una esponenziale inversa.



1.4 Data engineering

La fase di Data engineering è stata una delle più impegnative e intense del nostro progetto. Dopo uno studio attento, abbiamo deciso di trasformare la variabile Size in una variabile numerica crescente da 1 a 10 in base alla dimensione del veicolo. Per quanto riguarda Model, Make, Trim e SubModel abbiamo cercato di generalizzare il più possibile, in modo tale da avere valori più facilmente raggruppabili e non avere categorie numericamente esigue e statisticamente non significative.

La nostra tecnica generale per aggregare il più possibile queste variabili è stata quella di tener conto del rischio delle classi ma soprattutto del loro significato. Per quanto riguarda i Trim, ad esempio, abbiamo messo assieme tutti i GT, per la variabile SubModel tutti i Sedan ecc. Abbiamo creato, in Make, una categoria "British" dove abbiamo aggregato tutte le macchine britanniche: così anche per le Toyota e in altre occasioni simili.

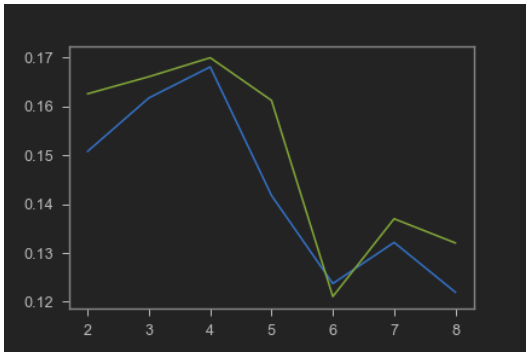
Per quanto riguarda Model, invece, abbiamo splittato per "spazio" e creato una nuova feature che abbiamo chiamato "property" e definisce per molte macchine una proprietà aggiuntiva come i cavalli o i litri. In questo modo abbiamo anche aggregato i model con lo stesso nome tra loro.

2. Data clustering

Dopo aver pulito e sistemato al meglio il dataset ci siamo trovati ad affrontare la sfida del clustering. Per l'enorme presenza di valori categorici non è stato facile ottenere clusters soddisfacenti per questo dataset. Abbiamo provato differenti approcci.

2.1 K-modes

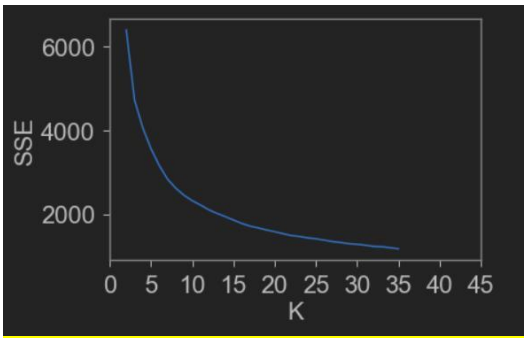
In prima fase abbiamo provato a cercare di clusterizzare il nostro dataset attraverso l'algoritmo K-modes. Per scegliere al meglio il numero di K, abbiamo definito una funzione che calcola l'entropia media per ogni clustering.



Abbiamo provato un k-modes con 6 cluster considerando solamente le variabili categoriche, ma i risultati sono stati insoddisfacenti: nonostante le mode, infatti, i gruppi trovati si sono dimostrati ampiamente eterogenei e poco esplicativi.

2.2 K-means

Dopo aver osservato la curva del gomito:

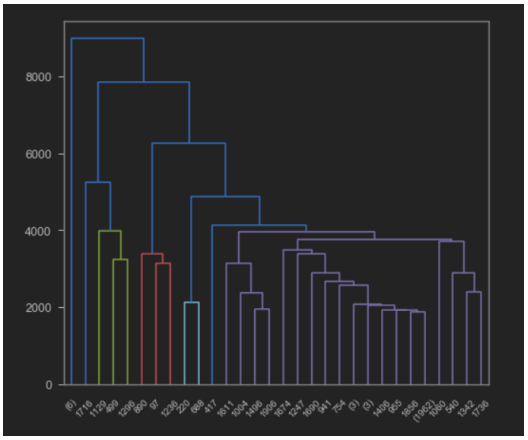


abbiamo optato per 6 cluster utilizzando tutte le nostre variabili numeriche. Il risultato del clustering è stato relativamente soddisfacente. Ad esempio, sono stati trovati due gruppi differenti di macchine di dimensione familiare con differente media di BadBuy e differente età. Nel gruppo 2 sono inscritti tutti i veicoli di dimensioni medio-grandi con rischio più elevato mentre nel gruppo 6 veicoli relativamente simili ma con rischio minore. Il cluster 3, a sua volta, non è troppo dissimile dal primo e dal secondo ma contiene macchine di valore medio con dimensione ed età leggermente più elevate. Le informazioni trovate non sono particolarmente impattanti sulla nostra analisi, ma sono facilmente leggibili e omogenee rispetto a quelle del K-Modes. Abbiamo poi tentato di descrivere i nostri dati con altre due tecniche di clustering.

	IsBadBuy	VehicleAge	VehOdo	VehBCost	WarrantyCost	OutCost	OutOdo	AverageMMR	Size_class
label									
0	0.193648	5.964688	75797.593206	5221.407867	1309.130796	0.987001	1.0	5363.123045	0.997052
1	0.073413	2.629579	51641.707920	6771.020455	825.861387	1.000928	1.0	8136.791790	0.948297
2	0.178903	6.009177	77796.093937	7083.198134	1738.229016	0.999278	1.0	7353.118117	5.463291
3	0.093262	3.240856	76893.932353	6618.674064	1290.314866	1.000000	1.0	8543.405789	1.213422
4	0.123451	5.005973	77309.138496	8563.312889	1089.745575	1.026991	1.0	9693.150717	8.577212
5	0.091668	3.364787	73812.329413	7841.322925	1450.423954	1.004220	1.0	10085.412240	4.631711

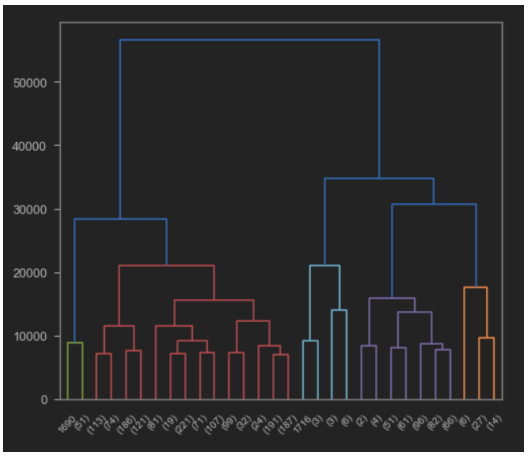
2.3 K-Means riassuntivo + clustering gerarchico

Il primo scoglio contro il quale ci siamo imbattuti è stato la costruzione della matrice delle distanze, necessaria per l'applicazione del clustering gerarchico. L'elevata dimensione del dataset non permette al calcolatore di eseguire tutti i calcoli necessari senza incappare in un Memory Error, motivo per cui la prima azione intrapresa è stata quella di "ridurre" il dataset ad un numero di righe accettabile, applicando il K-means per creare un numero sufficientemente grande di cluster. In questo modo, per ogni cluster, un certo numero di osservazioni sono rappresentate per ogni loro dimensione dal centroide del cluster dove ricadono. Dopo più tentativi, la scelta è ricaduta sulla creazione di 2000 cluster e sui centroidi di essi è stata costruita la matrice delle distanze. Nell'applicazione del clustering gerarchico, come per il seguente DB Scan, la nostra volontà è stata quella di includere per quanto possibile anche le variabili categoriali, con la accortezza di non includere le variabili del produttore e del modello dell'automobile. Abbiamo quindi dovuto definire una funzione di distanza mista, euclidea per le variabili continue, jaccard per le variabili categoriali, convertite in variabili dummy allo scopo. Successivamente, sono stati fatti più tentativi di clustering gerarchico, provando tutti i tipi di legame: single, complete e average. Sebbene il metodo single link sia risultato quello con la silhouette più elevata (0.357) esso presenta 7 cluster dimensionalmente troppo sproporzionati, con la quasi totalità delle osservazioni in un cluster solo, e gli altri sei composti da poche unità.



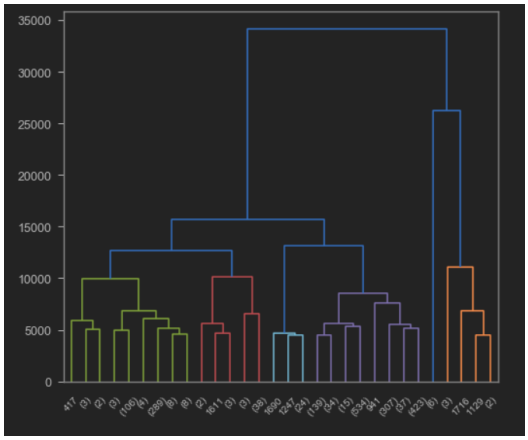

```
(array([1, 2, 3, 4, 5, 6, 7], dtype=int32),
 array([   6,    3,    1,    3,    2, 1984,    1], dtype=int64))
```

Il tipo di legame completo presenta invece il valore di silhouette più basso (0.296) ma con 5 cluster quantomeno più equamente popolati rispetto al metodo precedente, sebbene ci sia comunque un cluster dimensionalmente molto più grande rispetto agli altri.




```
(array([1, 2, 3, 4, 5], dtype=int32),
 array([  52, 1526,   13,  362,   47], dtype=int64))
```

Infine, il legame media presenta dei valori intermedi rispetto ai due precedenti, sia come silhouette (0.349) che come dimensionalità dei cluster, di numero 6. La nostra scelta è quindi ricaduta su questo tipo di legame.



```
(array([1, 2, 3, 4, 5, 6], dtype=int32),
 array([ 424,   47,   26, 1490,    6,    7], dtype=int64))
```

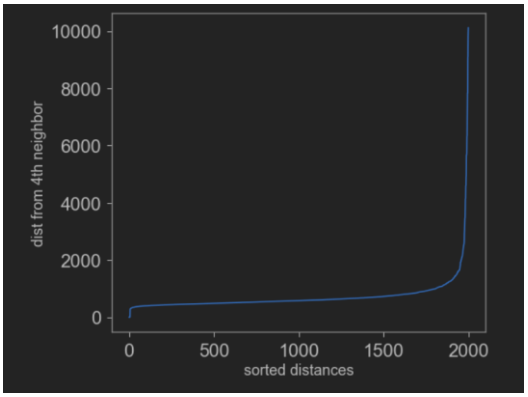
L'esito del clustering, seppur non pienamente soddisfacente, permette di distinguere 6 diversi gruppi di automobili che presentano caratteristiche diverse e rischi diversi. Nella figura sottostante, i cluster sono stati ordinati in ordine di rischio decrescente. Il primo cluster, sebbene di piccole dimensioni (7 automobili), identifica una nicchia di auto asiatiche sportive, di età relativamente giovane e valore relativamente basso, che sono tutte etichettate come bad buy.

Vale la pena ricordare che la variabile IsBadBuy non è stata utilizzata come dimensione del clustering e che quindi l'osservazione della proporzione di bad buy all'interno di ogni cluster è effettuata a posteriori.

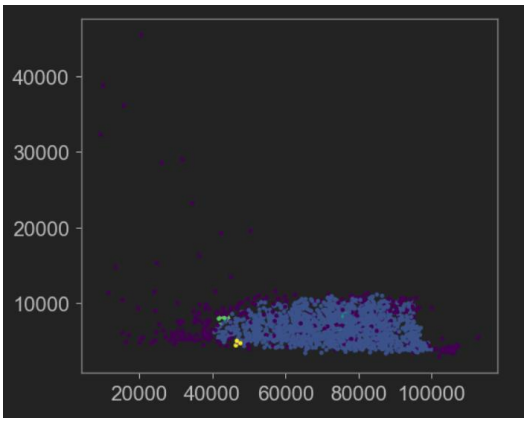
Label	IsBadBuy	VehicleAge	VehOdo	VehBCost	WarrantyCost	AverageMMR	Size_class	WheelType	Nationality	SubModel_New	Trim_New	Dim
6	1.000000	1.928571	71168.785714	33366.574286	960.714286	32669.767857	6.142857	Alloy	OTHER ASIAN	SPORT	Bas	7
3	0.105282	6.839744	104510.517675	4207.057338	2026.729838	5050.093764	2.282051	Alloy	AMERICAN	SEDAN	SE	26
2	0.213192	4.054428	26864.338796	6751.002358	878.919693	7951.579139	2.551683	Covers	AMERICAN	SEDAN	Bas	47
5	0.173312	4.500000	70652.038488	5999.036748	933.029365	6504.110149	2.669643	Alloy	AMERICAN	SEDAN	SE	6
4	0.149660	4.875662	79085.033762	6911.953411	1500.086474	7986.733215	3.785333	Alloy	AMERICAN	SEDAN	L	1490
1	0.117022	4.159563	51740.506394	7025.101574	1015.562690	7551.848401	2.865344	Covers	AMERICAN	SEDAN	Bas	424

2.4 DBSCAN

Per applicare l'agoritmo del DBScan, il punto di partenza è la matrice delle distanze trovata in precedenza. Al fine di settare i parametri dell'algoritmo in maniera ottimale, abbiamo controllato per ciascun punto a che distanza si trovasse il quarto punto più vicino, ottenendo così il classico grafico con la struttura "a gomito" da cui ricavare l'epsilon ottimale.



Il valore di epsilon scelto è di 1800. Essendo i punti del dataset molto condensati, l'output non risulta molto utile nell'interpretazione dei dati, sebbene riesca con efficacia ad individuare i punti di noise, come si può evincere dal grafico sottostante, in cui si mettono a confronto il valore dell'odometro(ascisse) con il valore del veicolo (ordinate).



I cluster individuati sono 5, di cui due molto piccoli. I punti di noise risultano essere quelli mediamente più rischiosi, mentre i punti core, come ci si può ragionevolmente aspettare, presentano un rischio nella media del dataset.

Label	IsBadBuy	VehicleAge	VehOdo	VehBCost	WarrantyCost	AverageMMR	Size_class
-1	0.179764	4.213433	64178.262213	7902.457303	1721.313129	8356.292712	4.074729
0	0.143226	4.807227	73948.023397	6796.887470	1314.161192	7894.973434	3.429857
1	0.098895	3.747222	76111.071806	7714.905252	1155.668417	0.020662	4.834199
2	0.058845	2.000000	43041.364366	7887.240975	930.670124	9611.492721	2.000000
3	0.102564	6.666667	46947.874237	4723.852259	1107.925519	3448.930556	1.666667

```
(array([-1,  0,  1,  2,  3], dtype=int64),
 array([ 362, 1619,   10,    6,    3], dtype=int64))
```

3 Data classification

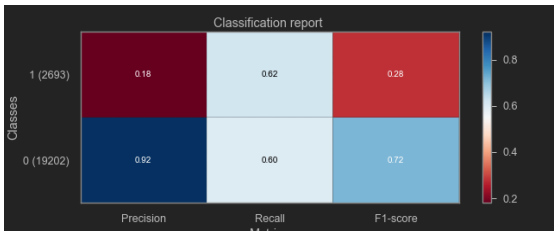
3.1 Decision Tree

Abbiamo per prima cosa tentato l'approccio con un semplice DecisionTree. Abbiamo operato una random search cross-validation sugli iperparametri (criterion, max_depth, min_samples_split, min_samples_leaf, class_weight). Abbiamo pesato differentemente le due classi della varabile target IsBadBuy perché, come anticipato nella sezione del data understanding, il dataset era sbilanciato verso la presenza di acquisti positivi. La nostra priorità, in ambito di classificazione, è stata cercare di ottenere una recall degli acquisti-frode la più alta possibile, nell'ottica per cui comprare una macchina che si rivela poi un acquisto negativo è molto più grave che perdere qualche buon affare.

Il miglior Decision Tree ottenuto è stato settato con i seguenti parametri:

```
min_samples_split= 100,
min_samples_leaf= 100,
max_depth= 8,
criterion= 'entropy',
class_weight= {0: 1, 1: 7.0}
```

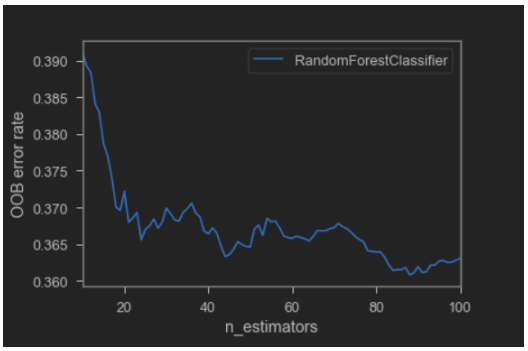
e ha fornito i seguenti risultati nel test set:



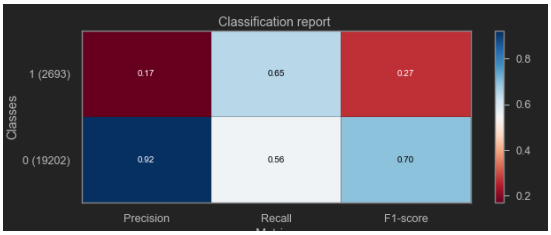
Il Gini index, utilizzato su Kaggle per giudicare la classifica dei progetti, si attesta, con questo modello, allo 0.21156. Considerando che il primo classificato ha raggiunto il Gini a 0.26719 il risultato non ci è sembrato soddisfacente. L'accuracy di questo modello si attesta al 56%. Per la classe 1, come si evince dall'immagine, abbiamo ottenuto il 18% di precision, il 62% di recall e il 28% di F1-score.

3.2 Random Forest

Il nostro secondo tentativo è stato con il Random Forest. Per scegliere il numero ottimale di alberi abbiamo eseguito l'algoritmo con un numero crescente di alberi al fine di osservare quando si raggiungesse l'errore minimo.



Abbiamo quindi optato per una foresta di 90 alberi. Anche con il random forest la nostra attenzione è stata molto voltata a dare peso ai Bad Buy.



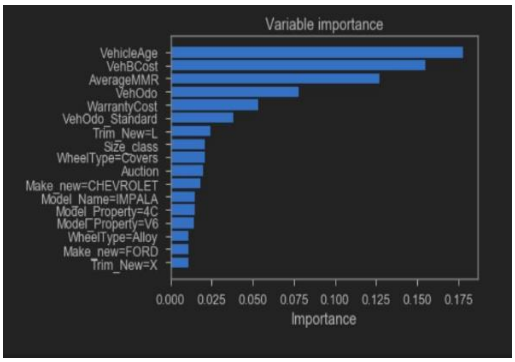
Il nostro risultato definitivo con il RandomForest è stato ottenuto con i seguenti parametri:

```
min_samples_split= 100,
min_samples_leaf= 100,
max_depth= 8,
 criterion= 'entropy',
class_weight= {0: 1, 1: 7.25},
oob_score=True,
warm_start=True,
random_state=10,
 n_estimators=90
```

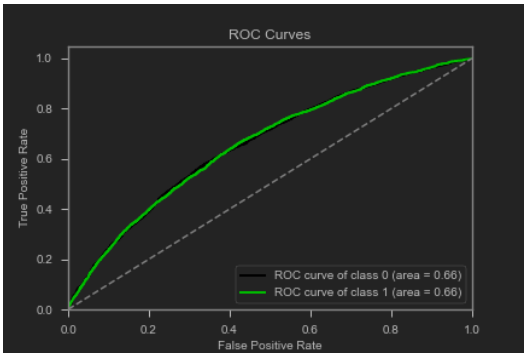
e ci ha consentito di arrivare a un esito molto più bilanciato nella recall delle auto di buona qualità e di migliorare l'F1-score generale. L'accuracy guadagna un 2% e per la classe 1, la precision guadagna un 1%, la recall guadagna un 2% e l'F1-score guadagna un 1%. Il gini index tocca lo 0.2468 e si avvicina molto di più alla parte alta della classifica di Kaggle e a un

risultato ottimale. Quindi, decidiamo di tenere questo come modello definitivo. Nell'ambito di questo classificatore, le feature più importanti sono:

1.	VehicleAge	0.178
2.	VehBCost	0.155
3.	AverageMMR	0.127
4.	VehOdo	0.078
5.	WarrantyCost	0.053
6.	VehOdo_Standard	0.038
7.	Trim_New=L	0.024
8.	Size_class	0.021
9.	WheelType=Covers	0.021
10.	Auction	0.020
11.	Make_new=CHEVROLET	0.018
12.	Model_Name=IMPALA	0.015
13.	Model_Property=4C	0.015
14.	Model_Property=V6	0.014
15.	WheelType=Alloy	0.011
16.	Make_new=FORD	0.011
17.	0Trim_New=X	0.011



Per concludere il nostro lavoro di classificazione, stampiamo quindi la curva Roc del nostro risultato.



4. Pattern Mining

Per quanto riguarda il pattern mining e le regole di associazione, abbiamo cominciato trasformando le variabili numeriche in categoriche, dividendo ciascuna in bin. Abbiamo poi proseguito cambiando le etichette di tutte le variabili in modo da renderle riconoscibili nella fase finale di studio delle regole.
Tramite la funzione apriori della libreria fim abbiamo calcolato gli itemset totali, con supporto 1% e considerando almeno 4 item per basket (n=41782), e il numero di regole per gli stessi parametri e confidenza 60%, abbiamo ottenuto 992987 regole.
A questo punto, abbiamo sperimentato un po' con i risultati filtrando in base a confidenza e lift.
Con lift > 2 e confidenza > 0.60 abbiamo ottenuto 346597 regole. Tuttavia, nessuna regola con lift > 1.5 implicava "NotBadBuy" o "IsBadBuy", per cui abbiamo deciso di abbassare la confidenza, arrivando ad abbassarla a 0.2.
Con confidenza = 0.2, nessuna regola implica "NotBadBuy", mentre quelle che implicano "IsBadBuy" sono 1387. Filtrando per lift > 2 otteniamo 84 regole.
Tra queste, abbiamo notato che molte hanno una "radice" in comune, ovvero condividono svariati item.
Ad esempio, molte Explorer con cost tra $4500 e $9094, con combinazioni di diversi modelli di grandi dimensioni, implicano un Bad Buy con un lift a 2.2, mentre la confidenza gravita sempre attorno al 27%.
Come detto in precedenza, età e prezzo sono variabili a cui fare attenzione per valutare l'acquisto di una macchina usata. Macchine vecchie, soprattutto di 8 anni ma anche 7, e poco costose, fino a $4500, indicano un pessimo acquisto con un lift tra 2.2 e 2.4, mentre macchine di 6 anni e acquistate a basso prezzo si rivelano rischiose soprattutto quando sono state stimate a un prezzo maggiore ($3800 – $7500) dalla rivista MMR (lift = 2.2).

Macchine acquistate a basso costo vanno trattate con cautela anche quando abbinate al trim TX, e in combinazione con svariate caratteristiche e modelli (lift tra 2.2 e 2.6). Infine, bisogna fare attenzione alle auto economiche ma con un alto numero alla variabile VehOdo, cioè che hanno percorso molta strada (82500 – 93500) e con cerchi in lega (lift = 2.3).