Loan Default

Big Data & Business Intelligence

Studente: Rocco Persiani
Professore: Gianfranco Lombardo

Anno accademico 2021/2022

UNIVERSITÀ
DI PARMA

Indice

1 Data Exploration

2 Data Preprocessing

3 Feature Selection e Feature Scaling

4 Model Comparison

5 Gradient Boosting Classifier

6 Ricerca degli Iperparametri

7 Metriche di Valutazione e Conclusioni

Data Exploration

Rappresenta il primo passo verso l'analisi dei dati, serve a specificare all'interno del Dataset:

- **Il tipo di variabili.**

```
Data columns (total 34 columns):
 #   Column                     Non-Null Count   Dtype
---  ------                     --------------   -----
 0   ID                         148670 non-null  int64
 1   year                       148670 non-null  int64
 2   loan_limit                 145326 non-null  object
 3   Gender                     148670 non-null  object
 4   approv_in_adv              147762 non-null  object
 5   loan_type                  148670 non-null  object
 6   loan_purpose               148536 non-null  object
 7   Credit_Worthiness          148670 non-null  object
 8   open_credit                148670 non-null  object
 9   business_or_commercial     148670 non-null  object
 10  loan_amount                148670 non-null  int64
 11  rate_of_interest           112231 non-null  float64
 12  Interest_rate_spread       112031 non-null  float64
 13  Upfront_charges            109028 non-null  float64
 14  term                       148629 non-null  float64
 15  Neg_ammortization          148549 non-null  object
 16  interest_only              148670 non-null  object
 17  lump_sum_payment           148670 non-null  object
 18  property_value             133572 non-null  float64
 19  construction_type          148670 non-null  object
 20  occupancy_type             148670 non-null  object
 21  Secured_by                 148670 non-null  object
 22  total_units                148670 non-null  object
 23  income                     139520 non-null  float64
 24  credit_type                148670 non-null  object
 25  Credit_Score               148670 non-null  int64
 26  co-applicant_credit_type   148670 non-null  object
 27  age                        148470 non-null  object
 28  submission_of_application  148470 non-null  object
 29  LTV                        133572 non-null  float64
 30  Region                     148670 non-null  object
 31  Security_Type              148670 non-null  object
 32  Status                     148670 non-null  int64
 33  dtir1                      124549 non-null  float64
dtypes: float64(8), int64(5), object(21)
memory usage: 38.6+ MB
```

- **I valori assumibili.**

	ID	year	loan_limit	Gender	approv_in_adv	...	LTV	Region	Security_Type	Status	dtir1
0	24890	2019	cf	Sex Not Available	nopre	...	98.728814	south	direct	1	45.0
1	24891	2019	cf	Male	nopre	...	NaN	North	direct	1	NaN
2	24892	2019	cf	Male	pre	...	80.019685	south	direct	0	46.0
3	24893	2019	cf	Male	nopre	...	69.376900	North	direct	0	42.0
4	24894	2019	cf	Joint	pre	...	91.886544	North	direct	0	39.0

Come possiamo notare il nostro target è Status e ci troviamo di fronte ad un task di classificazione

- **I possibili Missing Values**

```
ID                           0
year                         0
loan_limit                3344
Gender                       0
approv_in_adv              908
loan_type                    0
loan_purpose               134
Credit_Worthiness            0
open_credit                  0
business_or_commercial       0
loan_amount                  0
rate_of_interest         36439
Interest_rate_spread     36639
Upfront_charges          39642
term                        41
Neg_ammortization          121
interest_only                0
lump_sum_payment             0
property_value           15098
construction_type            0
occupancy_type               0
Secured_by                   0
total_units                  0
income                    9150
credit_type                  0
Credit_Score                 0
co-applicant_credit_type     0
age                        200
submission_of_application  200
LTV                      15098
Region                       0
Security_Type                0
Status                       0
dtir1                    24121
```

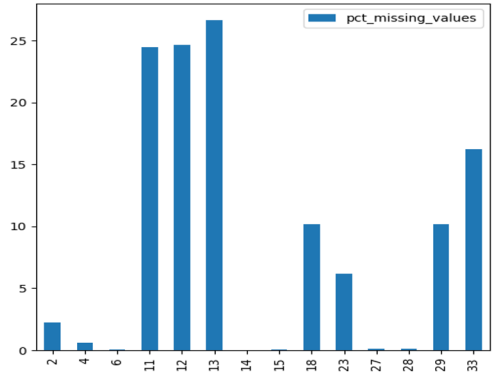
```
**********VERIFICA**********
ID                          0
year                        0
loan_limit                  0
Gender                      0
approv_in_adv               0
loan_type                   0
loan_purpose                0
Credit_Worthiness           0
open_credit                 0
business_or_commercial      0
loan_amount                 0
rate_of_interest            0
Interest_rate_spread        0
Upfront_charges             0
term                        0
Neg_ammortization           0
interest_only               0
lump_sum_payment            0
property_value              0
construction_type           0
occupancy_type              0
Secured_by                  0
total_units                 0
income                      0
credit_type                 0
Credit_Score                0
co-applicant_credit_type    0
age                         0
submission_of_application   0
LTV                         0
Region                      0
Security_Type               0
Status                      0
dtir1                       0
dtype: int64
```

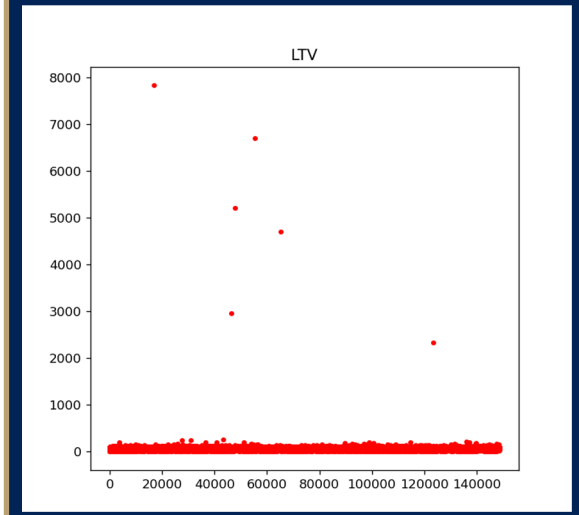

Missing Values

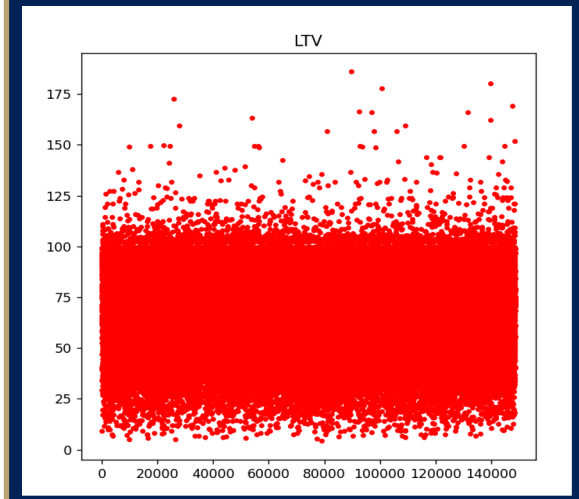
Per trattare i dati mancanti ho deciso di andare a eliminare completamente i valori con una percentuale bassa , mentre per i rimanenti , nel caso di :

Valori Quantitativi : I valori mancanti sono sostituiti dal valore della mediana della colonna di appartenenza

Valori Qualitativi : I valori mancanti sono sostituiti con il valore della moda della colonna di appartenenza







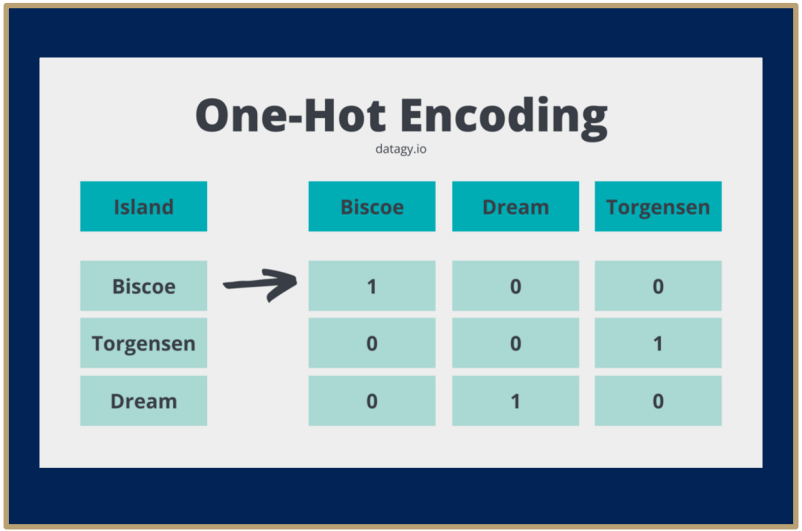
Outliers

I cosidetti outliers sono **valori anomali** che potrebbero andare ad influire in maniera negativa sull'addestramento del modello.

Ho considerato outlier i valori che si trovano a più di 3 deviazioni standard dalla media della colonna di appartenenza, questi ultimi in seguito verranno rimossi.

La **deviazione standard** è un valore che rappresenta la differenza di ogni osservazione dalla media della variabile.

Osservando un grafico di una specifica features ci accorgiamo di come la distribuzione dei vari valori cambi una volta rimossi gli outlier.



One Hot Encoding

Per poter lavorare sulle variabili qualitative, il modello necessita che quest'ultime siano codificate in modo da poter essere comprese.

Il **One Hot Encoding** è una tecnica che consente di codificare le variabili categoriche andando a sostituirle con 0 o 1 a seconda del valore numerico che assumono.

In particolare è utilizzato quando la scelta è binaria.


```
Score Model 1 : 0.999942321557318

Score Model 2 : 0.999971160778659

Score Model 3 : 0.8227253064167267
```

Mutual Information

E’ una tecnica di feature selection che va a misurare la **dipendenza fra variabili aleatorie** che va a definire quanto la nostra features è **descrittiva** della nostra target.

Avendo fatto dei test utilizzando tre dataset:

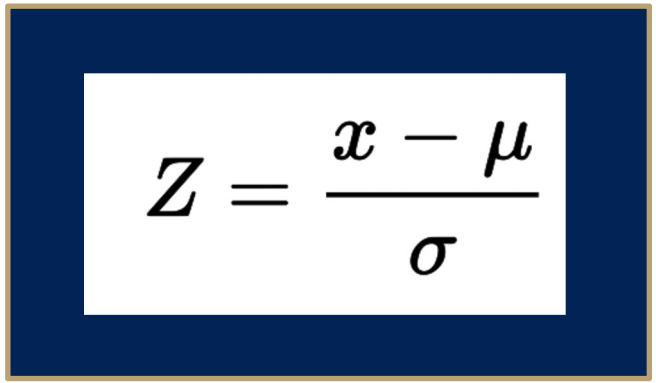
- Il primo ho ritenuto tutto il dataset iniziale valido
- Il secondo dataset ho considerato solo le features con score maggiore di 0.2
- Il terzo dataset ho tenuto in conto delle features con score minore di 0.2.

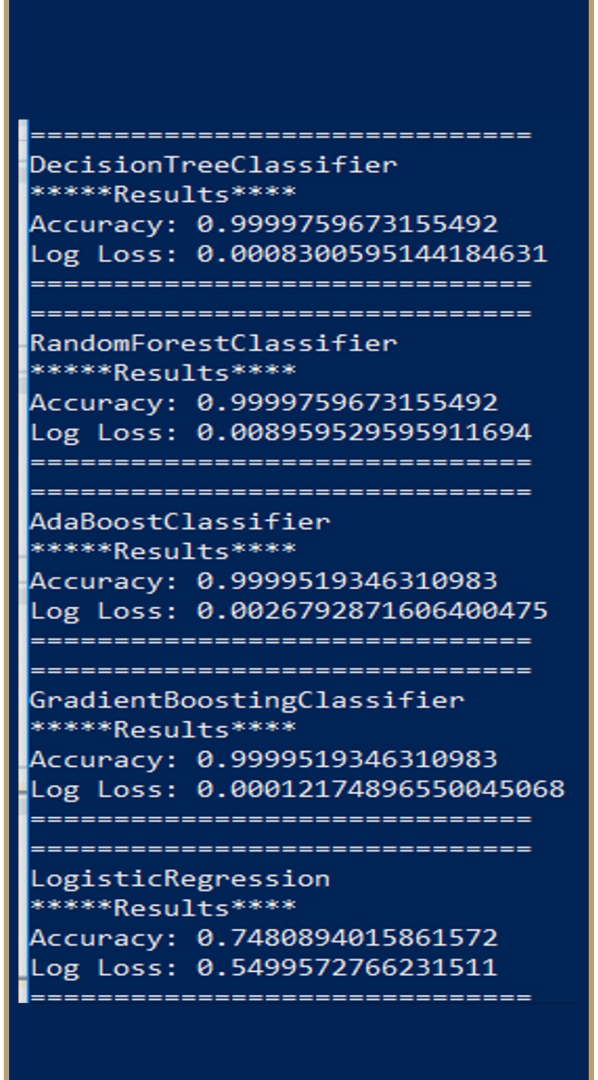
Come possiamo notare dai risultati abbiamo lo stesso score per i primi due dataset mentre uno score inferiore per il terzo.

Siccome lo score sul dataset completo e quello con feature con score maggiore di 0.2 sono simili ho optato per usare il dataset completo.

StandardScaler - Standardization

Questa tecnica di feature scaling va a modificare le feature X andando a sottrarne il valore medio e dividendo per la deviazione standard , in modo da avere tutte le feature intorno all'origine.





Model Comparison

Confrontiamo vari modelli e scegliamo i più performanti in base ai risultati di log loss e accuracy sul training di quest'ultimi:

Accuracy : indice che riassume la capacità del modello di rispondere in maniera corretta

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

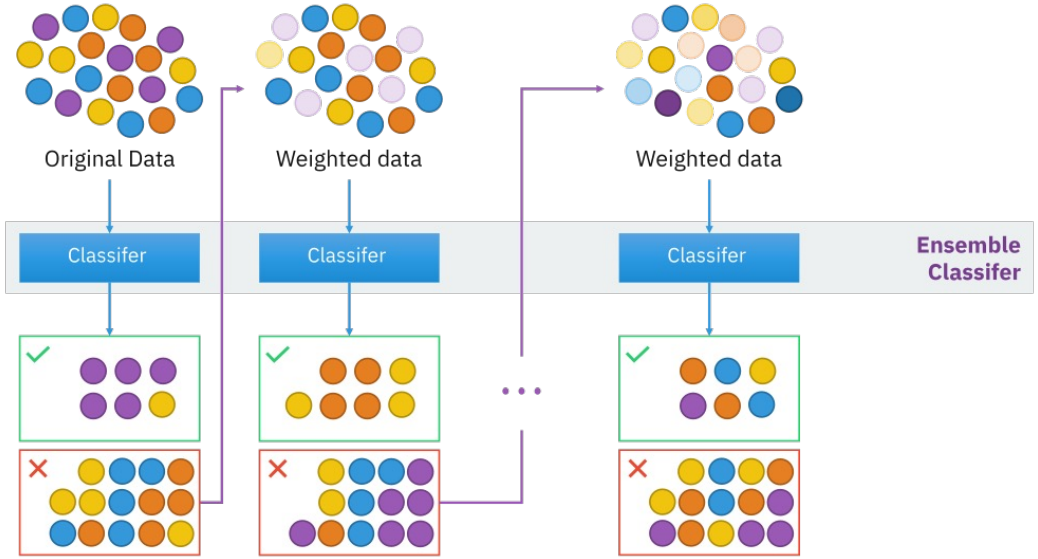
Log Loss : un valore basso di log loss indica previsioni migliori da parte del modello

Il modello più performante in termini di accuratezza e log loss è il **GradientBoostingClassifier**

Gradient Boosting Classifier

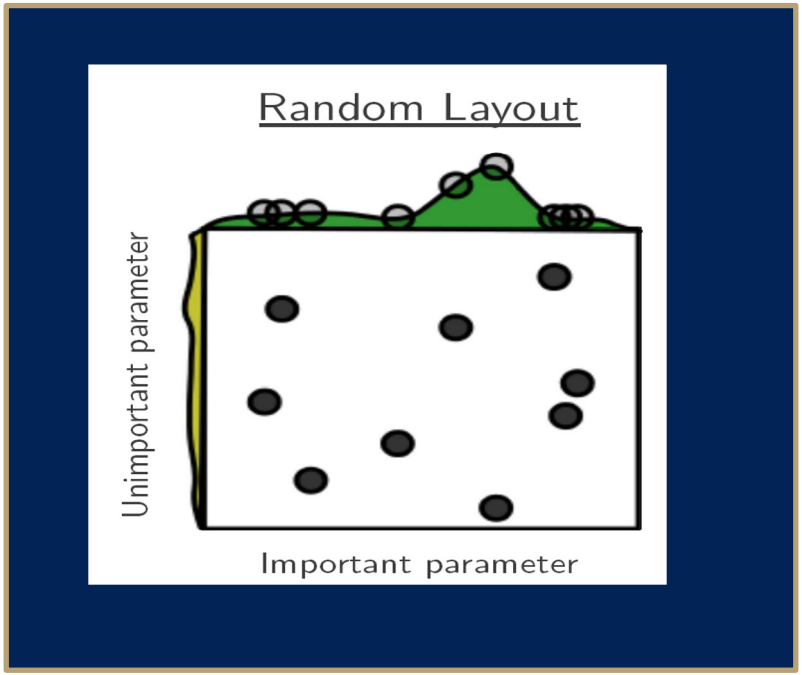
E’ un modello che si basa sulla composizione di **diversi classificatori addestrati in sequenza**.

Ogni predittore cercherà di aumentare l’accuratezza di quello precedente, andando a **minimizzare la funzione di Loss** utilizzando la discesa del gradiente.



E’ un modello di ensemble learning appartenente alla categoria detta “boosting”.

Ricerca degli Iperparametri



Random Search

E’ una tecnica in cui vengono utilizzate **combinazioni casuali degli iperparametri** per trovare i migliori per il modello costruito.

E’ più efficiente della grid search.

Metriche di valutazione

Precision

E' definita come il rapporto tra i true positive e la somma dei true positive e dei false positive, indica l'accuratezza con cui il modello prevede le classi positive

$$Precision = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Positive(FP)}$$

Recall

indica il rapporto di istanze positive correttamente individuate dal modello

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

F1 Score

E' definita come la media armonica tra precision e recall

$$\text{F1 score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Conclusioni

Gradient Boosting Final Evaluation

Durante la fase di progettazione sicuramente la parte più delicata e allo stesso tempo più importante a cui prestare attenzione è quella della **pulizia dei dati**, con l'ausilio di tecniche di **data cleaning**, poichè dati puliti fanno arrivare a risultati corretti mentre dati sporchi producono risultati errati.

Una volta ottenuto un dataset appropriato sono passato alla parte di **feature selection** e **feature scaling** che sono tecniche che si occupano di non considerare feature che non hanno contenuto informativo e normalizzazione dei dati di cui abbiamo a disposizione in modo da rendere più efficiente la loro analisi.

Dopo aver confrontato vari modelli e trovato il migliore ho cercato gli **iperparametri** che rendevano il modello più efficiente possibile.

Valutazione finale:

```
==============================
Accuray del modello finale: 0.9999759673155492
Precision del modello finale: 1.0
Recall del modello finale: 0.9999048887198022
F1 Score del modello finale: 0.9999524420982547
==============================
```