

# Pipeline per analisi dati

## 1. Introduzione

L'analisi dati è un processo fondamentale per trasformare dati grezzi, apparentemente inutili, in informazioni preziose per decisioni ponderate.

Per garantire il successo di una corretta analisi dati, bisogna prima strutturare una pipeline la quale possa garantire efficienza e una replicabilità.

Una pipeline all'interno del mondo dei dati, rappresenta una serie di fasi sequenziali, che guidano il flusso di lavoro dalla definizione del problema fino alla sperimentazione finale.

Questo documento andrà ad esplorare tutte le fasi necessarie per garantire un pipeline ben strutturata.

## 2. Fasi della pipeline

### 2.1. Definizione del problema

La definizione del problema è forse la fase più importante della pipeline, bisogna comprendere quale problema si andrà ad affrontare attraverso modelli di machine learning. Lo scopo è quindi definire le informazioni che si vogliono estrarre dai dati e attraverso quale strategia.

La definizione del problema non finisce con il concetto di business che si vuole applicare, ma comprende anche come verrà affrontato. Bisognerà capire dunque che tipo di machine learning verrà applicato.

#### 2.1.1. Supervised learning

I modelli di Supervised learning, vengono così chiamati perché i dati sono strutturati in un certo modo. La suddivisione prevede la parte denominata "Data" e la parte chiamata "Label". Il concetto è semplice, i modelli supervisionati cercano tramite le informazioni presenti nella porzione "Data" a trovare dei pattern per predire delle "etichette". La parte supervisionata avviene durante il training del modello, in cui se la previsione della "label" dovesse essere sbagliata, il modello cercherà di correggersi. Le categorie principali per problemi supervisionati di machine learning sono:

- **Classification:** Viene utilizzata per i problemi che necessitano un'assegnazione di etichetta binaria. Un esempio pratico può essere la classificazione delle mail in spam o non spam;
- **Regression:** Utilizzata per problemi che necessitano un valore continuo. La differenza principale con la "Classification" è appunto l'output che deve essere una variabile continua e non discreta. Un esempio è la predizione del valore di una casa attraverso l'immissione di diversi parametri;
- **Raccomandation:** "Raccomandation" è un problema che prevede la raccomandazione di prodotti o servizi a degli utenti basandosi sullo storico dei dati o dei comportamenti.

#### 2.1.2. Unsupervised learning

L'approccio ai modelli di Unsupervised learning avviene quando disponiamo dei dati ma non delle "etichette". Un problema di machine learning non supervisionato è la segmentazione della clientela, si cerca quindi di raggruppare i diversi utenti per diversi parametri di somiglianza all'interno di cluster. Questo permetterà di assegnare delle etichette in base ai cluster formati.

### 2.2. Raccolta e Gestione dei dati

La raccolta e gestione dei dati è un altro passaggio chiaramente fondamentale per la pipeline, senza dati non si può analizzare niente. La raccolta dei dati si suddivide principalmente in due

categorie, stream di dati costanti che possono derivare da diverse fonti per defluire in un data warehouse o data lake, oppure in dati più statici che non hanno un flusso continuo come file CSV o xlsx.

Bisogna fare anche una distinzione dei dati raccolti e di quale categoria questi facciano parte. E' possibile ottenere dati strutturati, facilmente comprensibili e analizzabili, hanno una struttura ben definita, possono essere disposti in maniera tabellare e utilizzano tipi di dato ben definiti (interi, date, booleani, float...).

Dati invece non strutturati possono essere come immagini, audio, linguaggio testuale. Categoria di dati che è più difficile disporre all'interno di una tabella o database.

### **2.3. Pulizia e Preprocessing dei dati**

Prima di approcciarsi ad una prima esplorazione dei dati per la creazione di insight è prassi pulire o trasformare i dati.

La pulizia dei dati si occupa principalmente di eliminare eventuali dati mancanti o duplicati. Per evitare di eliminare grandi quantità di dati è possibile intraprendere una strategia di correzione dei dati, in linea con i parametri necessari oppure contrassegnati da marker specifici. La trasformazione dei dati invece prevede una loro normalizzazione convertendo le variabili in formati adeguati per il modello (Esempio: OneHotEncoder della libreria scikit-learn) o la creazione di nuove variabili.

### **2.4. Esplorazione e Analisi dei dati**

Questa fase all'interno della pipeline ha lo scopo di creare dei report dai dati presenti prima di essere analizzati dai modelli di machine learning. E' possibile comunque riscontare pattern, correlazioni o differenze molto spesso dai dati di partenza puliti.

Le analisi svolte in questa fase assieme agli insight finali daranno un quadro più dettagliato e possibilmente anche più interessante.

### **2.5. Ingegnerizzazione delle caratteristiche**

Riuscire a estrapolare e identificare le variabili più importanti, creare nuove variabili combiando o trasformando alcune già esistenti e applicarne tecniche di scaling, rappresenta la fase di Ingegnerizzazione delle caratteristiche.

Molte volte risolvere il problema prefissato senza eseguire questi passaggi è molto difficile se non impossibile. Da variabili pre esistenti come i vari pagamenti di diversi utenti si possono creare nuove variabili come la frequenza di acquisto, la spesa media ecc... Anche applicare tecniche di scaling (Esempio: MinMaxScaler scikit-learn) comporta a benefici, in maniera da garantire che le variabili siano su scale comparabili.

### **2.6. Sviluppo e Addestramento del modello**

La scelta del modello di machine learning è fondamentale per ottenere i migliori risultati per uno specifico problema.

La figura sottostante, reperita direttamente dal sito ufficiale di scikit-learn, attraverso la scelta di diversi parametri, consiglia l'algoritmo più consono al tipo di categoria e alla grandezza del dataset disponibile.

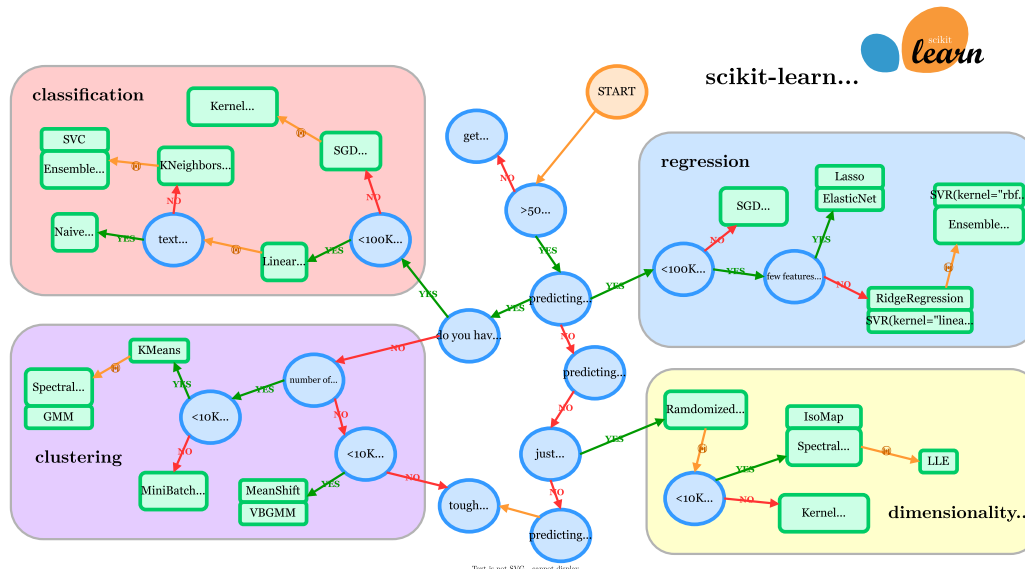


Figure 1: Scelta modello

La fase successiva è addestrare il modello, dividendo il dataset in `Training_set`, `Validation_set` e `Test_set`, la porzione di dataset `Validation_set` non è obbligatoria ma aiuta ad ottenere score del modello più alti. Le divisioni più comuni sono 80/10/10 o 70/15/15, nel caso non sia presente il `Validation_set` si divide in 80/20 o 75/25.

Se si volesse provare a migliorare l'efficienza del modello è possibile eseguire quello che viene chiamato "hyperparameter tuning", dove attraverso la documentazione si modificano parametri del modello per ottenere risultati più soddisfacenti.

## 2.7. Valutazione e Validazione del modello

La valutazione e validazione del modello ci conferisce una panoramica generale di quanto attendibili possano essere le previsioni. Ci sono diversi tipi di metriche, soprattutto tra i modelli supervisionati rispetto a quelli non supervisionati.

### 2.7.1. Supervised

Bisogna innanzitutto parlare delle differenze dei risultati che il modello ritorna in output:

- Falsi positivi: il modello predice un'etichetta in maniera positiva, ma che dovrebbe essere negativa;
- Falsi negativi: il modello predice un'etichetta in maniera negativa, ma che dovrebbe essere positiva;
- Veri positivi: il modello predice un'etichetta in maniera positiva che dovrebbe essere positiva;
- Veri negativi: il modello predice un'etichetta in maniera negativa che dovrebbe essere negativa.

La gravità degli errori dipende molto da che problema si cerca di risolvere e in quale percentuale questo avvenga.

#### Metriche:

- Recall: la capacità di un modello di identificare correttamente le istanze positive. È la proporzione di veri positivi rispetto alla somma dei veri positivi e dei falsi negativi;
- Precision: la precisione delle previsioni positive di un modello. È la proporzione di veri positivi rispetto alla somma dei veri positivi e dei falsi positivi;
- F1 score: è la media armonica della precision e della recall, ed è utilizzato per bilanciare entrambi gli aspetti quando c'è una disparità significativa tra il numero di falsi positivi e falsi negativi;

- MAE: misura l'errore medio tra le previsioni del modello e i valori reali. È la media delle differenze assolute tra le previsioni e i valori osservati;
- RMSE: misura l'errore quadratico medio tra le previsioni del modello e i valori reali. È la radice quadrata della media dei quadrati delle differenze tra le previsioni e i valori osservati.

### **2.7.2. Unsupervised**

Le metriche di modelli non supervisionati (prendiamo come prima solo problemi di clustering) si concentrano sulla qualità dei gruppi formati.

**Metriche:**

- Silhouette Score: misura quanto un oggetto è simile al proprio cluster rispetto ai cluster vicini. Varia tra  $-1$  e  $1$ , dove valori più vicini a  $1$  indicano che gli oggetti sono ben abbinati al proprio cluster e ben separati dagli altri cluster;
- Davies-Bouldin Index: misura la media delle similitudini tra ogni cluster e il cluster più simile ad esso. Valori più bassi indicano cluster ben separati e compatti.

## **2.8. Documentazione e Comunicazione**