Lezione 9 17/04/2024

Data understanding & exploration

Anche se il dataset è già preparato, bisogna guardarlo, analizzare i dati anche tramite visualizzazioni. Anche mentre il sistema è in produzione, bisogna controllare man mano che i dati utilizzati siano gli stessi rispetto a quelli usati per il training.

Appena ho ottenuto il modello si devono fare dei **sanity checks**, ovvero dei controlli di qualità.

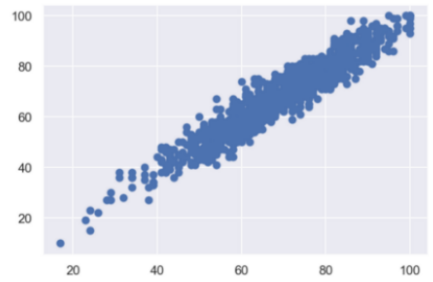
Si possono utilizzare le **tabelle**, andando a capire quali valori sono continui, discreti, categorici, quali sono operazioni di count... i dati discreti, categorici e di count sono tutti discreti, sono numeri interi, ma hanno un semantica diversa. Quindi non basta fare un controllo di tipo.

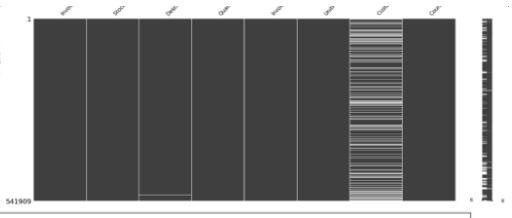
I sanity check che posso applicare dipendono dal dominio. Alcuni sono banali, per esempio latitudine e longitudine bisogna controllare se rientrano nei range corretti. Posso vedere le distribuzioni di variabili categoriche o continue per vedere se è corretto. Posso vedere se una feature è presente in un numero sufficiente di esempi (o sufficienti esempi diversi). Poi c'è un aspetto di consistenza dei dati, per esempio una persona non può avere più di un'età. Tutti questi controlli quindi controllano quali siano i valori ammissibili per ciascuna feature, in base alla sua **semantica**.

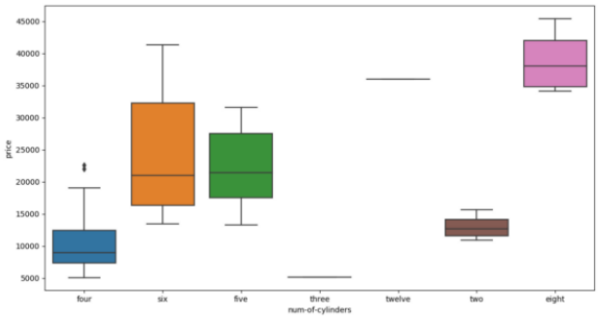
Per fare questi controlli si possono fare visualizzazioni grafiche, degli script o anche query SQL.

Se viene trovato un **errore**, si può eliminare il record (se non sono tanti gli errori), altrimenti provo a correggere, per esempio inserendo il valore medio. A volte invece non si possono correggere, e quindi bisogna capire perché sono errati alla fonte.

```
plt.scatter(Performance_df['Reading_Score'], Performance_df['Writing_Score'])
```





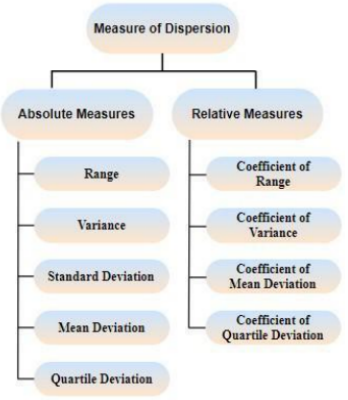


Nel grafico a sinistra possiamo vedere una correlazione tra reading_score e writing_score. Nel secondo grafico possiamo vedere con linee bianche quali righe abbiano una feature vuota. Nel terzo invece possiamo vedere la distribuzione dei numeri di cilindri delle auto.

La **data exploration** è il processo di andare a vedere quali sono le variabili e qual è il loro ruolo. Si possono fare **analisi univariate**, capendo le caratteristiche della variabile, oppure quella **bivariata**, dove vado a capire il comportamento rispetto a due variabili.

Misure di dispersione

Vado a calcolare la varianza, la mediana, misure assolute, relative... i quartili...

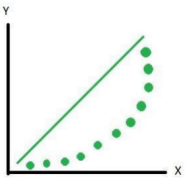


Box plot

Il box plot disegna un box dal primo al terzo quartile. La linea orizzontale mostra la media. Quelle verticali vanno da ciascun quartile al minimo o al massimo.

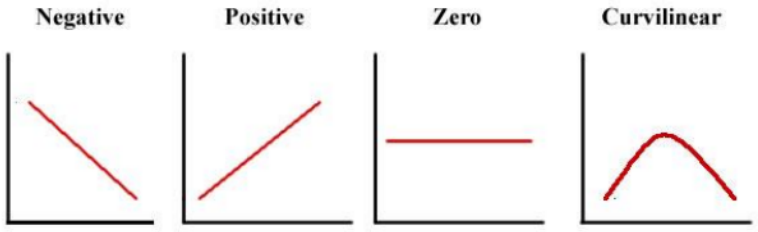
QQ plot

Sono usati per capire se la distribuzione dei dati deriva da una distribuzione teorica tipo la normale o esponenziale.



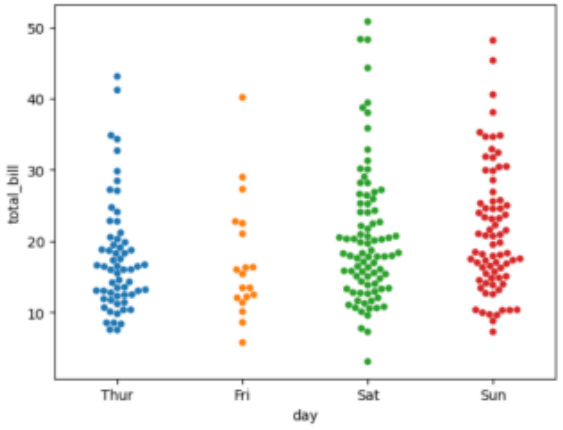
Scatter plot

Ci da informazioni sulla relazione tra due variabili, permette di vedere se ci sono dei pattern, è utilizzata per valutare se una feature può essere un buon modo di predire una variabile target.

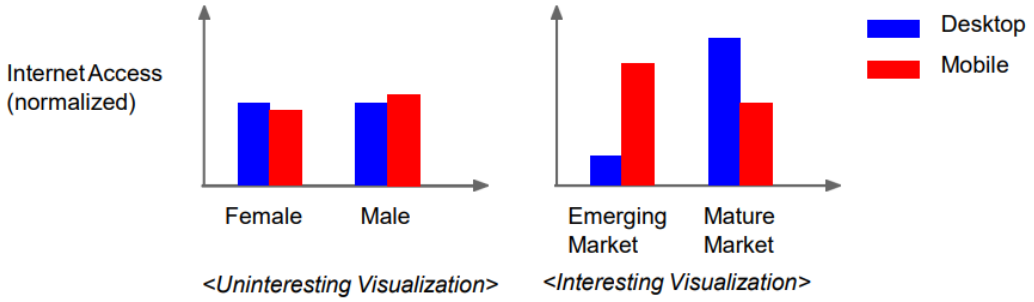


Swarmplot

Partono dall'idea di uno scatterplot, ma per la variabili categoriche. L'estensione di questi violini, è il numero di item.



Siccome i modelli di visualizzazioni e di variabili sono tanti, c'è un problema di capire qual è la migliore visualizzazione. Per esempio se sto studiando un dataset che contiene l'accesso ad internet, tramite computer o tramite telefonino, e se la persona è uomo o donna, e se vengono da un mercato emergente o da un mercato maturo.



Per esempio non è interessante la visualizzazione del sesso, perché essendo 50/50 non ci da informazione. Mentre invece la provenienza da paese in via di sviluppo o sviluppato è più interessante.

Tra i tipi di analisi ML c'è quella di identificare quali features contribuiscono di più alla qualità del modello.

Guardando i risultati, devo capire questi aspetti, anche se c'è una sorgente dati che produce errori. In alcuni casi si possono eliminare questi esempi, o capirne la provenienza.

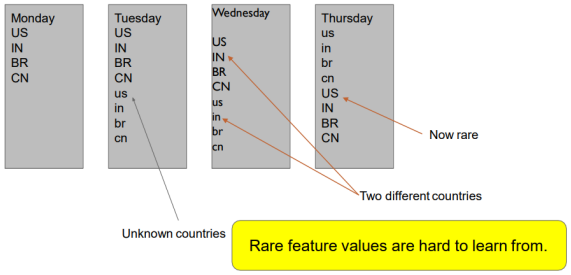
Infine bisogna capire se il modello è "fair", ovvero che il modello non produce bias, che è corretto.

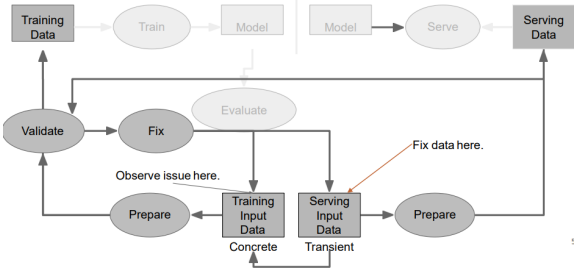
Poi bisogna anche poter identificare nuovi tipi di "spam", ovvero se ci sono degli utenti che stanno abusando il sistema (esempio l'algoritmo identifica lo

spam, e qualcuno trova dei punti deboli per non fare identificare il proprio spam).

Data validation

Il mondo può cambiare dopo la creazione del modello, per esempio potremmo avere l'origine di un paese scritto con le due lettere maiuscole, ma improvvisamente quelli che producono quel dato decidono di passarlo minuscolo senza comunicarlo.





I problemi vanno individuati e poi vanno sistemati.

Come faccio ad accorgermi che qualcosa non va? Buona prassi è che quando creo un sistema, metto anche dei meccanismi di controllo con **alert**. Per esempio per accorgersi quando compaiono dei nuovi valori nel campo "country".