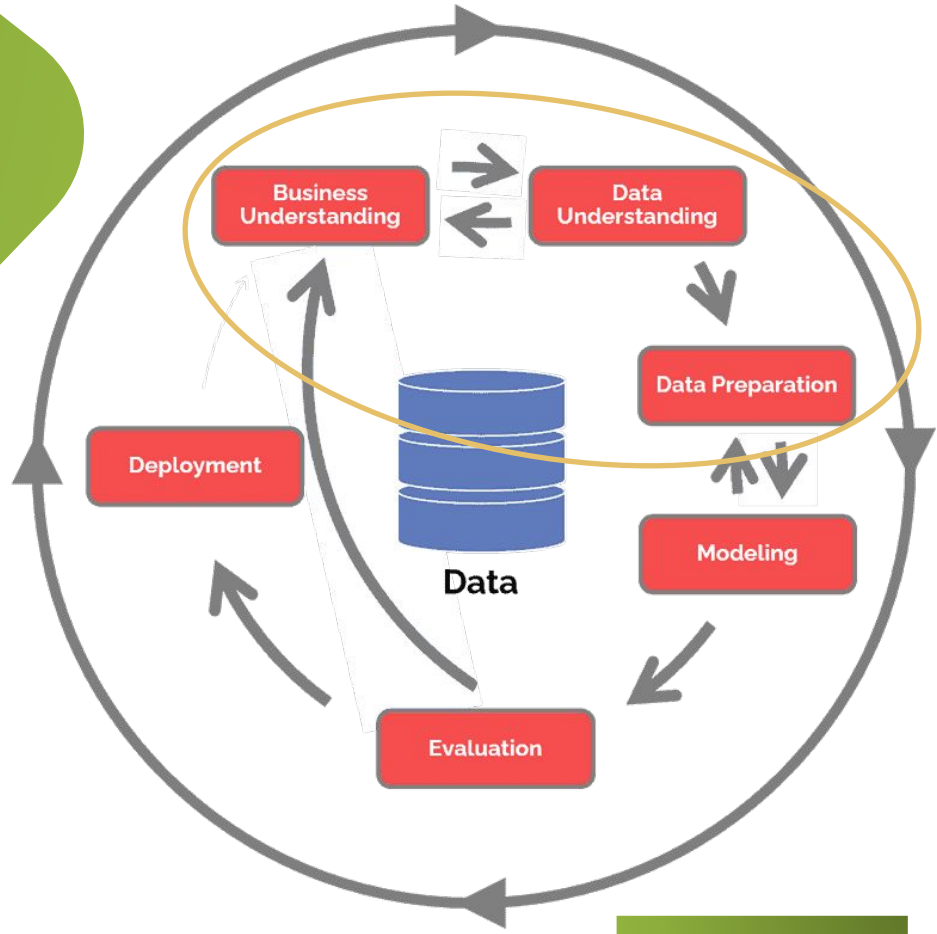
Revelis

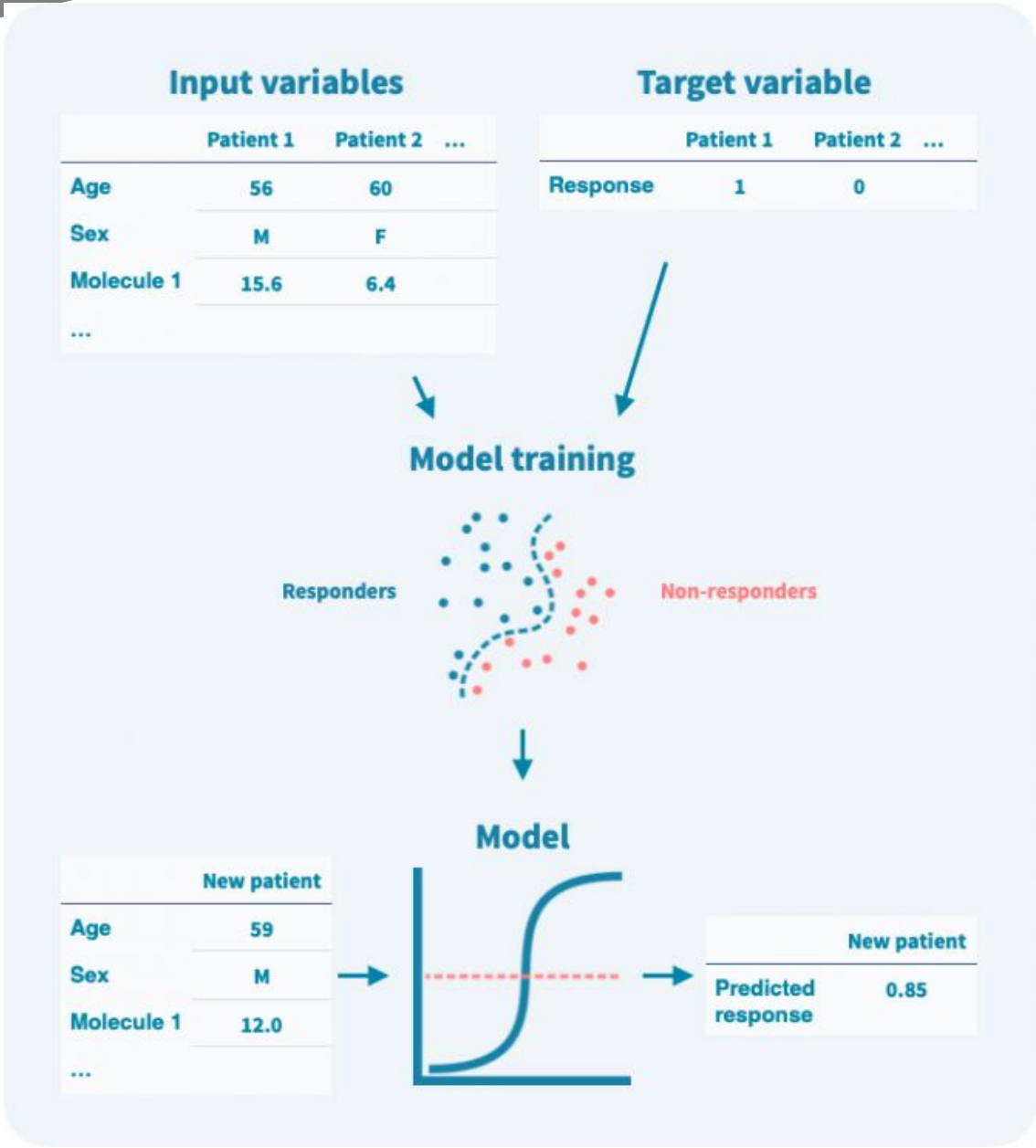


Machine Learning – EDA e Preprocessing dei dati

Davide Iacopino

Individuazione variabile target

- Individuare qual è la **variabile target**, ossia la variabile che vogliamo che il modello usi come risultato del suo processo di inferenza

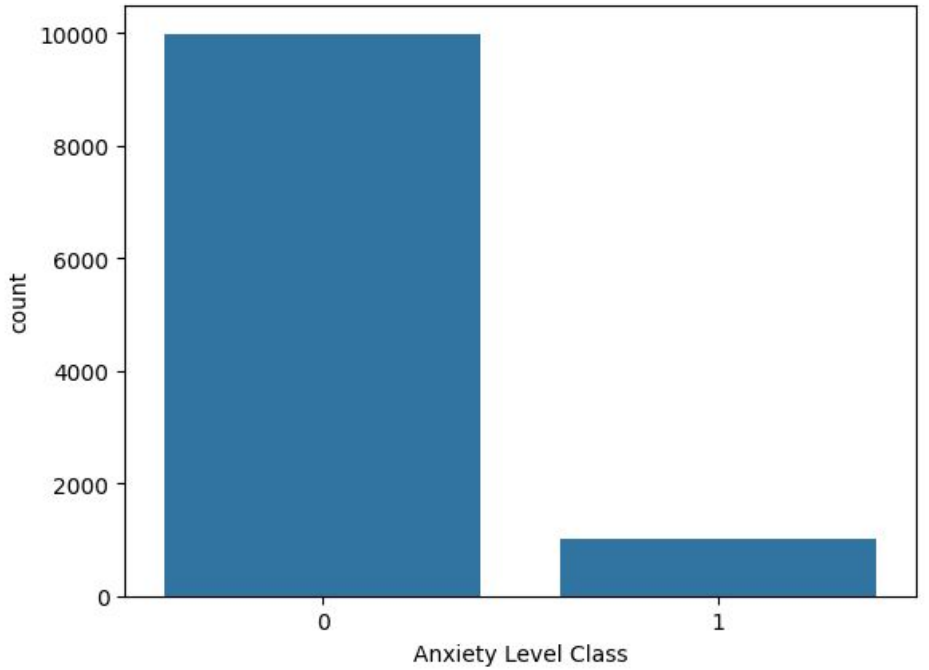


- Analisi **distribuzione dei valori della variabile target/bilanciamento delle classi**

Revelis

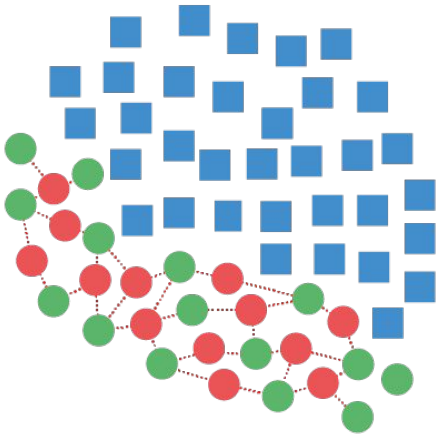
Gestione classi sbilanciate

- Classi sbilanciate se 1 o più classi hanno molti più esempi rispetto alle altre classi



Soluzioni:

- **Oversampling della classe meno frequente (e.g. SMOTE, duplicazione, data augmentation)**
- **Undersampling della classe più frequente**
- **Assegnare un peso maggiore agli errori sulla classe meno frequente**

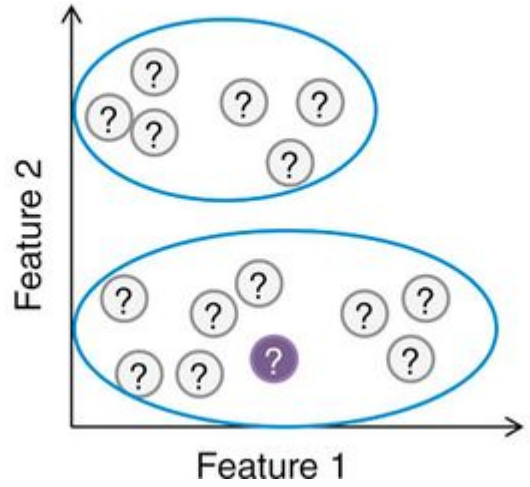


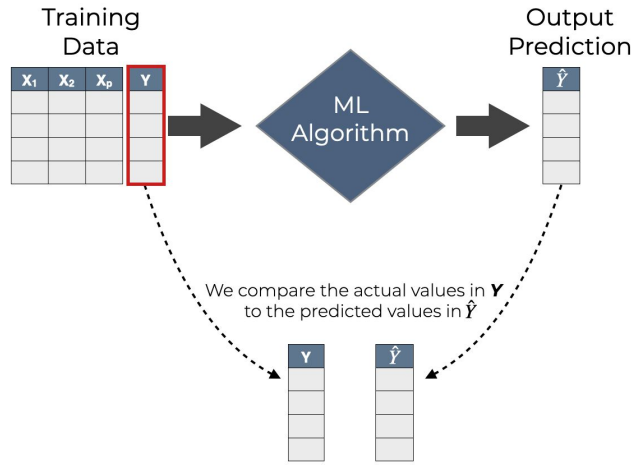
Revelis

ML Supervisionato vs non-supervisionat

Le tecniche di ML si dividono in 3 categorie:

- ML supervisionato: nel dataset è presente la variabile target, ossia i dati sono etichettati con l'output atteso

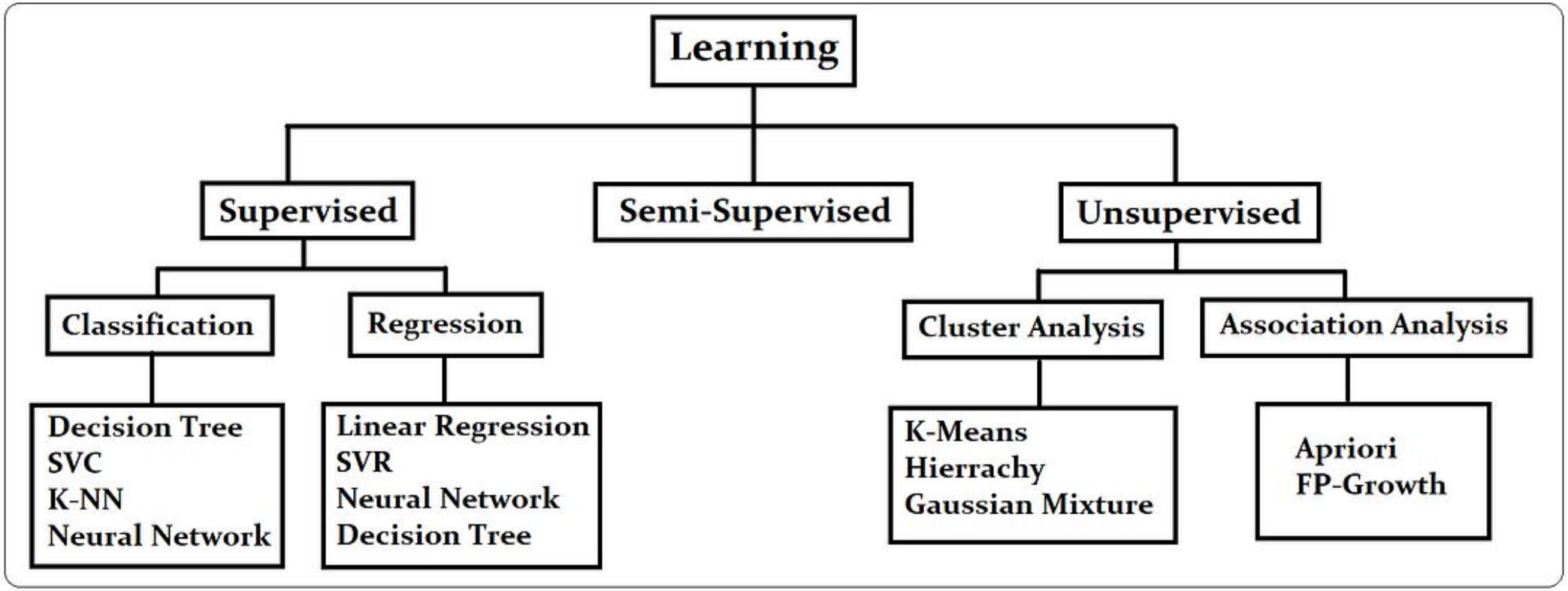




- ML non supervisionato: dataset non etichettato
- ML semi-supervisionato: dataset etichettato solo per alcuni esempi

Revelis

ML Supervisionato
vs
non-supervisionat



Revelis

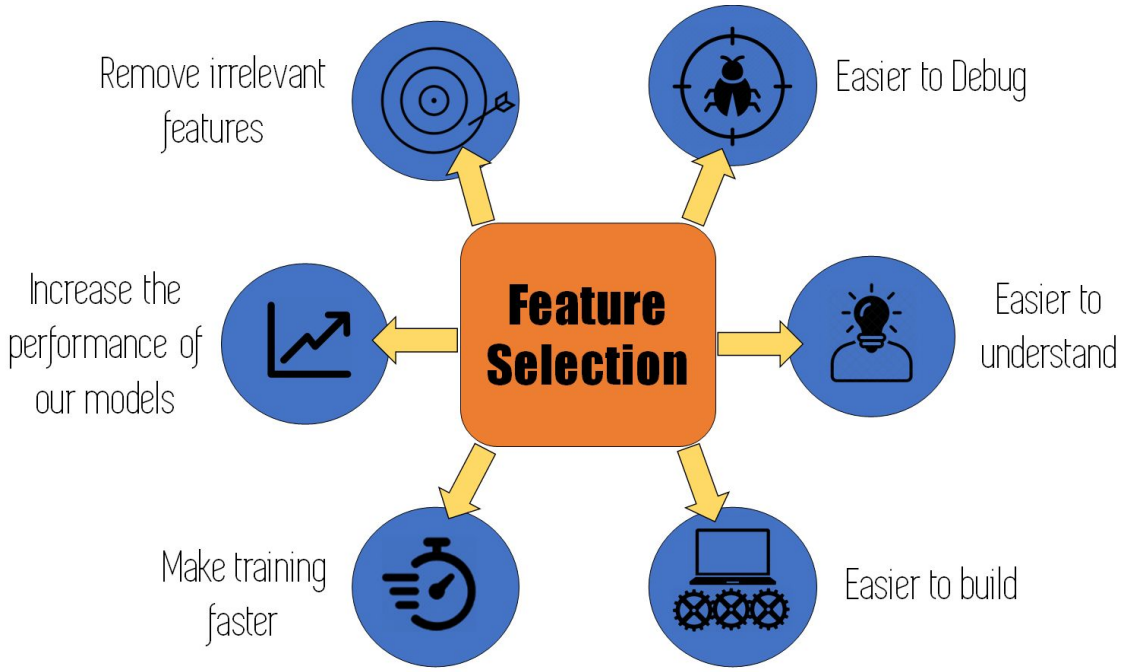
Feature selection

Individuare e eliminare eventuali **colonne fortemente correlate** ad altre presenti nel dataset o **ridondanti** (che non danno informazioni aggiuntive) o **non utili**

index	id	title	type	description	release_year	age_certification	runtime	imdb_id
0	tm84618	Taxi Driver	MOVIE	A mentally unstable Vietnam War veteran works ...	1976	R	113	tt0075314
1	tm127384	Monty Python and the Holy Grail	MOVIE	King Arthur, accompanied by his squire, recrui...	1975	PG	91	tt0071853

Revelis

Feature selection

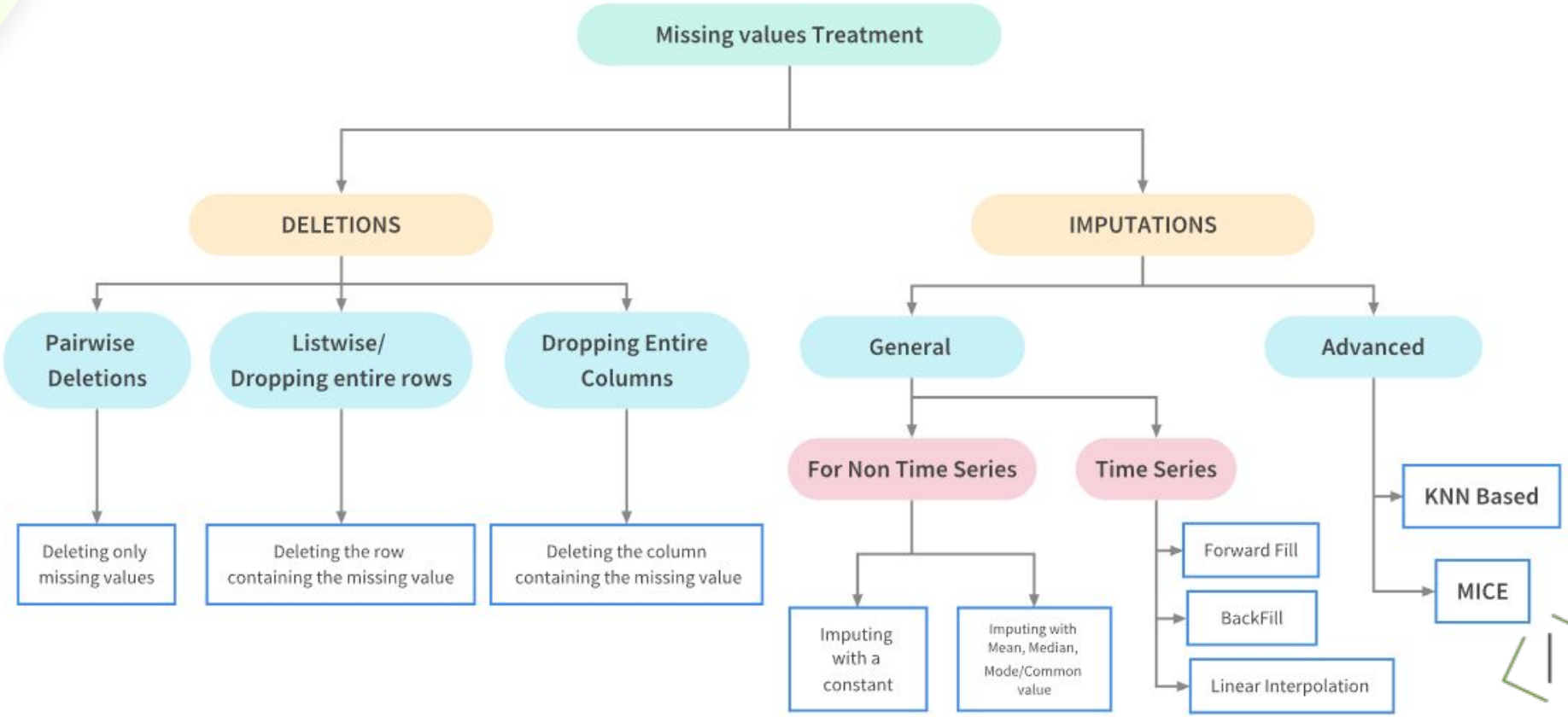


- Individuare e selezionare le colonne utili al modello per l'apprendimento e la risoluzione del task (e per la presentazione dei risultati finali)
- Utile perché alcuni modelli/tecniche richiedono di non elaborare troppe features (problema dell'elevata dimensionalità)

Revelis

Gestione valori mancanti e righe duplicate

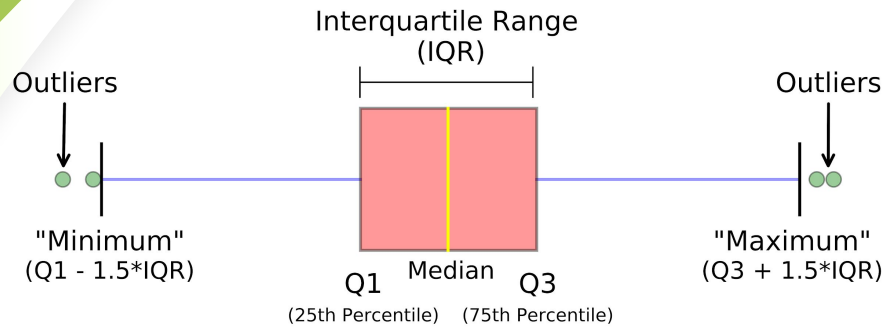
- Le righe duplicate vanno eliminate
- Per alcuni esempi, i valori di alcuni campi possono mancare
- In molti casi questo è un problema per le operazioni successive, quindi va gestito



Revelis

Gestione dati anomali (outliers)

1. Identificare outliers
   - Metodi grafici: box-plot
   - metodi statistici: Intervallo Interquartile (IQR), z-score
   - Metodi basati su tecniche di ML: DBSCAN, KNN, One-class SVM



2. Gestione (se sono tanti e creano problemi)
   - Rimozione o imputazione (se ipotizziamo siano dati errati)
   - Trasformazione (smorzando la loro distanza dai dati «normali»)

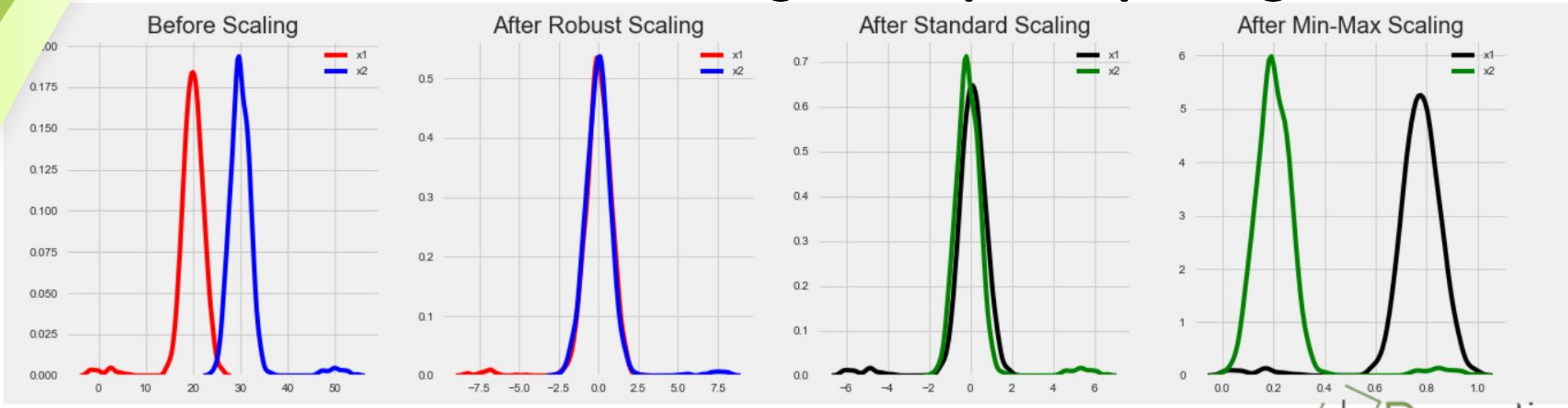
3. Utilizzo tecniche più robuste (meno sensibili agli outliers)

Revelis

Gestione scale di valori diverse

	gender	hsc_p	ssc_p	age	height	salary	suffer_from_disease
0	M	81.4	82.2	44	6.1	120000	no
1	M	75.2	86.2	40	5.9	80000	no
2	F	80.0	83.2	34	5.4	210000	yes
3	F	85.4	72.2	46	5.6	50000	yes
4	M	68.4	87.2	28	5.11	70000	no

- In molti casi, il fatto che diverse feature abbiano scale di valori molto diverse è un problema
- Si gestisce **scalando/normalizzando** i dati
- Attenzione: Da eseguire **dopo lo splitting dei dati**



Revelis

Feature engineering

	Datetime	Count	year	month	day	dayofweek_num	dayofweek_name
0	2012-08-25 00:00:00	8	2012	8	25	5	Saturday
1	2012-08-25 01:00:00	2	2012	8	25	5	Saturday
2	2012-08-25 02:00:00	6	2012	8	25	5	Saturday
3	2012-08-25 03:00:00	2	2012	8	25	5	Saturday
4	2012-08-25 04:00:00	2	2012	8	25	5	Saturday

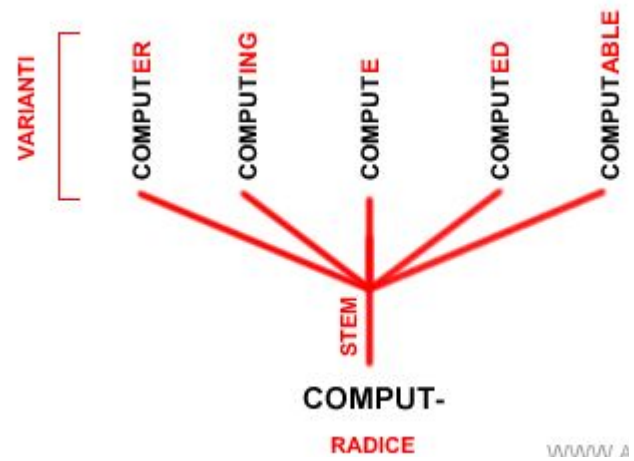
Costruire nuove feature a partire da quelle già presenti:

- **Estrarre informazioni** da campi con valori complessi
- La maggior parte dei modelli non lavorano coi dati categorici (testo), quindi **trasformare dati categorici in numerici**

Temperature	Order	Binary	Temperature_0	Temperature_1	Temperature_2
Hot	1	001	0	0	1
Cold	2	010	0	1	0
Very Hot	3	011	0	1	1
Warm	4	100	1	0	0
Hot	1	001	0	0	1
Warm	4	100	1	0	0
Warm	4	100	1	0	0
Hot	1	001	0	0	1
Hot	1	001	0	0	1
Cold	2	010	0	1	0

Revelis

Preprocessing



Abbiamo visto le principali operazioni di preprocessing per dati tabellari. Il preprocessing, però, **dipende dal tipo di dato**:

- Con le **immagini**, bisogna effettuare per esempio un **resize** per far sì che abbiano tutte la stessa dimensione
- Coi **testi**, spesso bisogna suddividere il testo in token, rimuovere le «stopwords», effettuare lemmatizzazione o stemming, riduzione a lowercase ecc
- Con le **serie temporali**, spesso bisogna raggruppare (cambiare granularità)

Revelis

Splitting del dataset

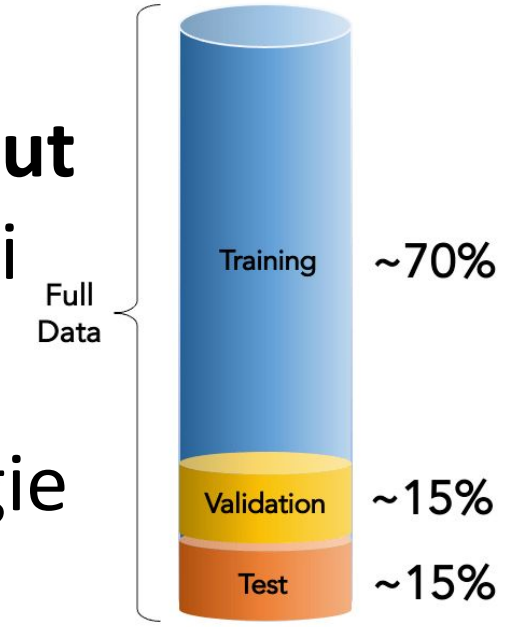
Non tutto il dataset deve essere usato per addestrare il modello

- Abbiamo bisogno di 3 insiemi di dati:

1. **Training set**: per **addestrare il modello**, ossia per fargli apprendere a risolvere il task
2. **Validation set**: per **valutare le performance del modello durante l'addestramento e l'hyperparameter tuning** su dati che non fanno parte del training set (dati su cui il modello non ha appreso)
3. **Test set**: per **valutare le performance del modello finale** su dati su cui il modello non ha appreso e su cui non sono state prese delle decisioni durante lo sviluppo

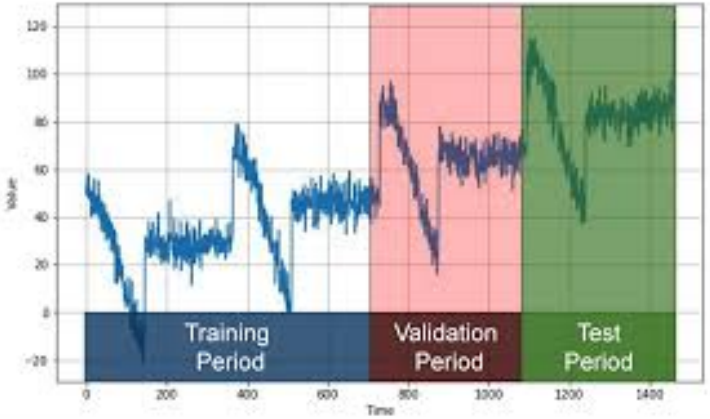
Splitting del dataset

- Tecnica più semplice: **holdout** (split statico con percentuali tipo 70%-15%-15%)



- Per avere varietà e far sì che tutte le tipologie di esempio siano presenti in tutti i set, si fa uno **shuffle** dei dati prima di suddividerli

- **Questo non ha senso per le serie temporali**, per cui i set devono essere continui e contigui tra loro



Revelis

Splitting del dataset

Tecniche più evolute:

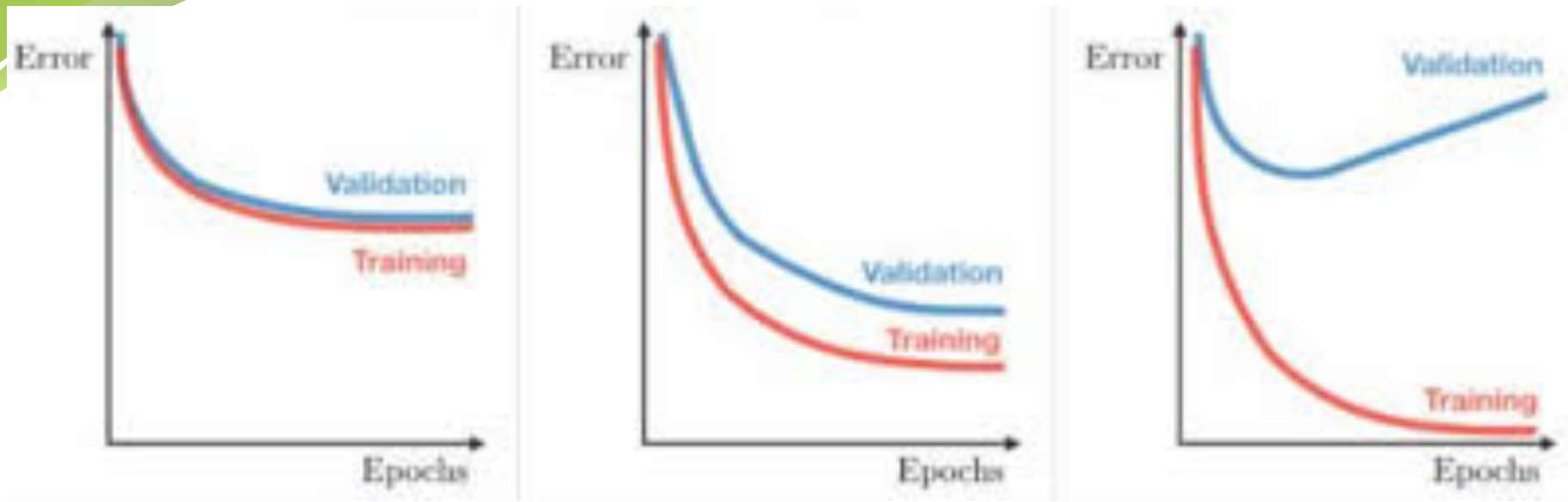
- **k-fold CV**: Il dataset viene diviso in **k blocchi** (fold), a turno uno come validation e gli altri come training (**riduce problemi di poca varietà/eterogeneità** dei dati nel training set)
- **stratified split/stratified k-fold CV**: **mantiene la distribuzione delle classi** (adatto per dataset sbilanciati)
- **Leave-One-Out Cross-Validation (LOOCV): una sola osservazione come test**, il resto come training (adatto per dataset piccoli)

Generalizzazione

- Separare training, validation e test set serve per valutare la **capacità di generalizzazione del modello**
  - Capacità di generalizzazione è la capacità di apprendere dei pattern dal training set e **applicarli correttamente su dati che non ha visto** durante l'addestramento, quindi generalizzare le informazioni apprese dal training set su tutti i dati
- Per questo è necessario che il training set copra quanto più possibile la varietà delle tipologie di dato su cui il modello dovrà essere applicato

Revelis

Underfitting e overfitting



- **Overfitting**: il modello ha appreso in modo troppo specifico le informazioni del training set, quindi fatica a generalizzarle su dati nuovi
- **Underfitting**: il modello non ha appreso dal training set le informazioni necessarie per effettuare l'inferenza, quindi fatica sia sui dati del training set che sui dati nuovi

Revelis

Underfitting e overfitting

La capacità di apprendere e di generalizzare del modello dipende da tanti fattori:

- Varietà/eterogeneità dei dati: nel training set devono essere presenti tutte le tipologie di dato, la CV può aiutare
- Quantità e complessità dei dati: pochi dati portano ad avere meno informazioni da apprendere e a non coprire tutte le possibili tipologie di casistiche, data augmentation può aiutare
- Complessità del modello: più è complesso, più informazioni apprenderà dai dati del training set, maggiore sarà il rischio di overfitting

Revelis

Revelis

linkedin.com/company/revelis-srl/
facebook.com/revelissrl

Salvatore Iiritano
CEO
salvatore.iiritano@revelis.eu

Davide Iacopino
Data Analyst
davide.iacopino@revelis.eu

Rende
V.le della Resistenza, 19/C
87036 Rende (CS)

Parma
Largo L. Mercantini, 13
43125 Parma (PR)

Telefono
(+39) 335.1099492

Fax
(+39) 0984.494269

info@revelis.eu

www.revelis.eu