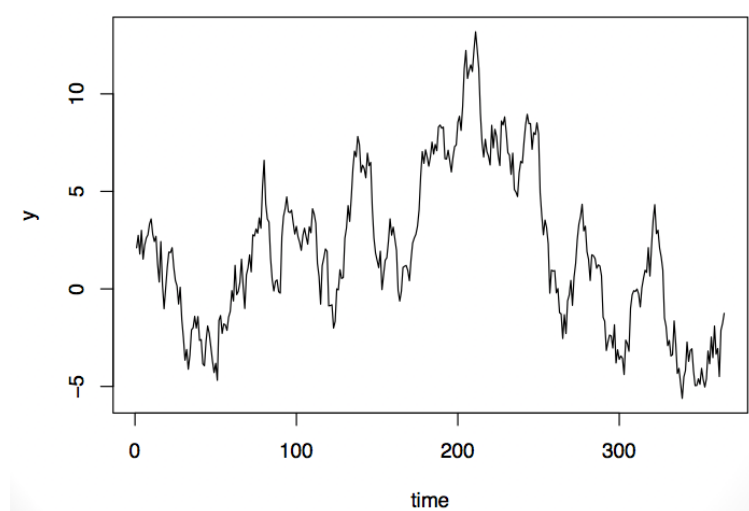


Scheda progettuale

Descrizione:

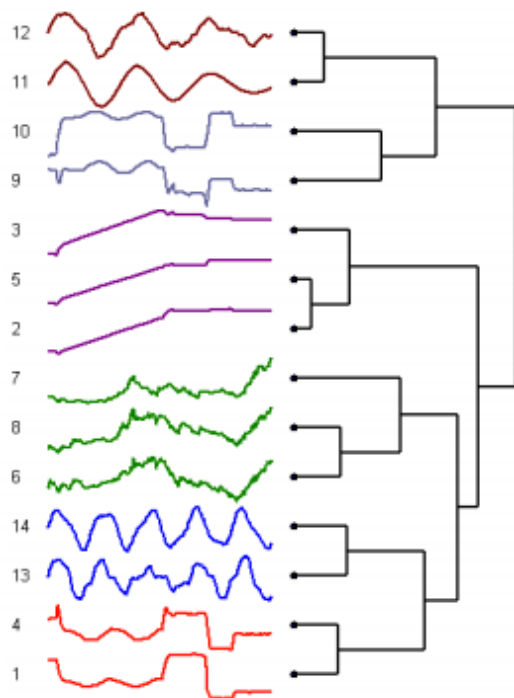
L'utilizzo e lo studio delle [Time Series](#) è diventato un aspetto cruciale negli ultimi anni. Con l'avvento dell' [NLP](#), viene spesso usato affinché si possano riconoscere e classificare queste serie temporali.

Le time series sono distribuite su due piani, dove sull'asse delle ascisse troviamo ovviamente il tempo, mentre sull'asse delle ordinate troviamo un valore che caratterizza quell'istante nel tempo. Ovviamente ad un valore nel tempo corrisponde un ed un solo valore.

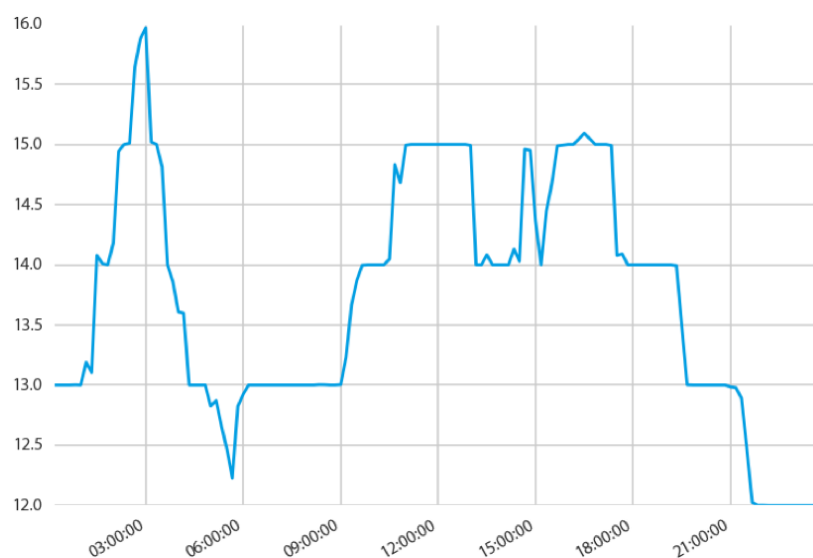


In letteratura, soprattutto nel campo del trading, ci si pone l'obiettivo di provare a prevedere valori futuri sulla base dei valori precedenti. Questo è utile in quanto, soprattutto se fatto a lungo termine, potrebbe aiutare a capire all'utente su quale titolo puntare.

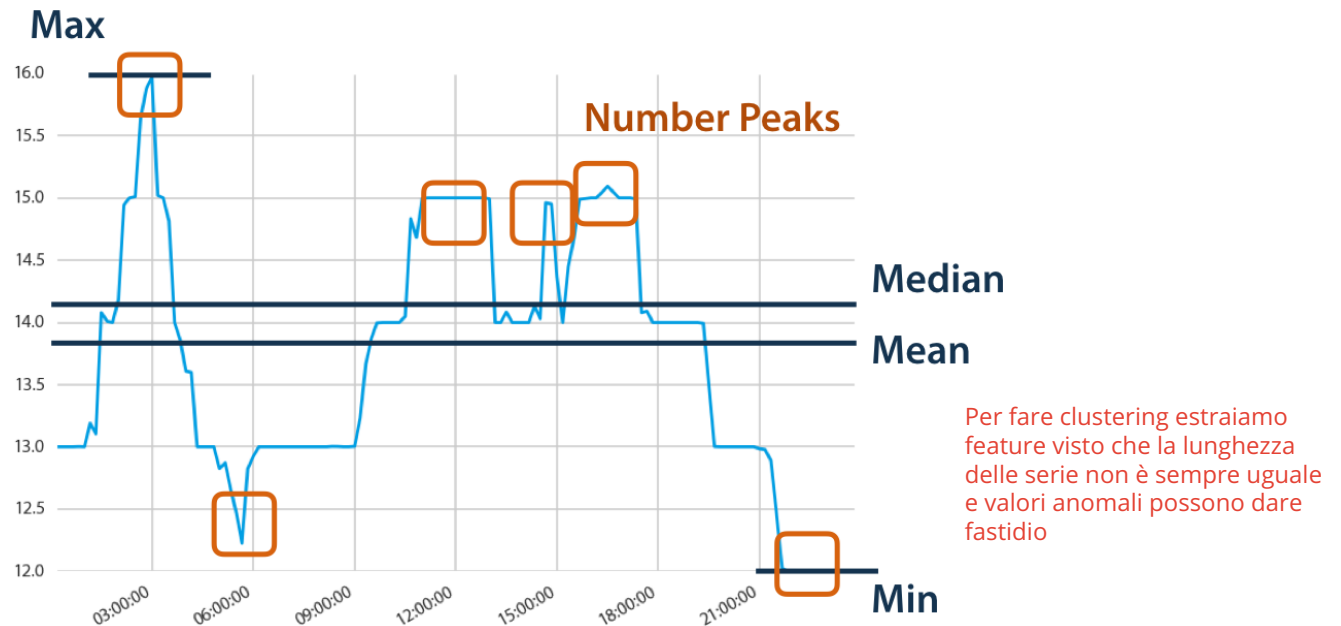
Oltre a questo obiettivo, spesso è utile cercare di associare tra loro Time Series che risultano avere un andamento simile, quindi, come nell'esempio, un'idea potrebbe essere quella di applicare tecniche di apprendimento non supervisionato per cercare di clusterizzare fra loro le time series.



Spesso, questo tipo di clusterizzazione viene fatto usando due metodi. Il primo si basa sull'applicazione del clustering sui dati grezzi, quindi applicando delle misure di similarità fra le time series. Il problema di questo tipo di approccio è che spesso le time series hanno lunghezza diversa, oppure potrebbero avere degli outlier che potrebbero rendere tale metodo inapplicabile. Un ulteriore approccio si basa sulla estrazione di feature dalla time serie, e quindi vengono poi utilizzate quest'ultime per provare a fare clustering. Per esempio, si supponga di avere la seguente time serie:



Dalla stessa è possibile estrarre numerose feature, tra cui:



Queste ultime potrebbero essere inserite in una tabella $N \times M$, dove N è il numero di time series ed M il numero di features:

	Max	Min	Median	Mean	Numb Peaks
TS ₁	10	2
TS ₂
.....
TS _n

Una volta ottenuta questa tabella, è possibile applicare dei metodi di clustering che permettano di unire fra loro le time series che risultano essere più o meno simili.

Strumenti:

Per questo progetto ci si vuole focalizzare su alcuni tools che esistono in letteratura. Uno di questi è [TSFRESH](#), utile per estrarre molte features dalle time series, ed è sviluppato in Python. Esso permette infatti di analizzare ed estrarre molte features dalle time series, da utilizzare per poter fare clustering. Ovviamente, non tutte le features sono utili ai fini del clustering, ed infatti un problema dell'utilizzo di questo applicativo sta proprio nel fatto di **capire quali features potrebbero essere discriminanti oppure no**.

Estrarre le features con TSFRESH è molto semplice. Infatti, una volta adattata la time serie che ci interessa, in modo tale che sia utilizzabile dal modulo, basterà invocare una funzione Python che ci consentirà di poterle estrarre:

```
from tsfresh import extract_features
extracted_features = extract_features(timeseries, column_id="id", column_sort="time")
```

Questa è la funzione che verrà usata per poter estrarre le features.

Quindi, una volta estratte le features e scelte quelle più discriminanti, andremo a creare la tabella, sulla quale potremo successivamente applicare un qualsiasi tipo algoritmo non supervisionato.

Alcuni dataset contenenti time series che potremmo prendere in considerazione sono recuperabili a questo [indirizzo](#). Quelle sulle quali vogliamo porre maggiori attenzioni sono gli ECG, ossia segnali di elettrocardiogrammi, i quali sono segnali di dati finiti, quindi non flussi di dati. Tali time series sono anche provviste di classe di appartenenza, quindi potrebbe essere fatto un confronto tra algoritmi supervisionati e non, in modo da poter verificare la precisione di questi ultimi nel classificare tali classi. Ovviamente, trattandosi appunto, di algoritmi non supervisionati, non possiamo dire che la distanza di due classi in termini di features risulta essere una buona base discriminante.