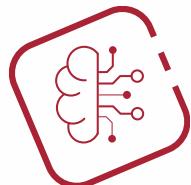




Algoritmi per il Machine Learning

Dott. Antonio Giovanni Lezzi



MACHINE LEARNING

Dott. Antonio Giovanni Lezzi





ALGORITMI DI MACHINE LEARNING

- **Ripasso sulla Regressione lineare**
- Uso di Excel
- Implementazione in Python
- Impiego di SciKit-Learn
- Coefficiente di determinazione
- Errore Quadratico Medio



REGRESSIONE

- Nei problemi di regressione si tenta di prevedere una **variabile dipendente** (solitamente indicata da Y) confrontandola con una serie di altre variabili (note come variabili indipendenti, solitamente indicate da X).
- Esistono diversi tipi di regressione, a seconda del tipo di dati che si vuole prevedere, i più diffusi abbiamo:
 - **Regressione Lineare**
 - **Regressione Logistica**
 - **Regressione di Poisson**



REGRESSIONE

- **Regressione Lineare:** quando si vuole prevedere un valore continuo (ad esempio la temperatura di oggi). Se la variabile di input è solo una allora la regressione lineare si dice semplice, altrimenti in caso contrario la regressione lineare si dice multipla.
- **Regressione Logistica:** quando si vuole prevedere in quale categoria si trova l'osservazione si sta trattando un problema di regressione logistica (si tratta di un gatto o di un cane?);
- **Regressione di Poisson:** quando si vuole prevedere un valore di conteggio che impatta il modello (ad esempio il numero di guasti informatici catastrofici in una grande azienda tecnologica in un anno di calendario).



REGRESSIONE LINEARE

- Tale termine deriva dall'applicazione nata dall'esploratore **Galton** che nel 1886 esaminò le altezze dei figli (y) in funzione delle altezze dei genitori (x) in Inghilterra.
- Galton notò con sorpresa una relazione funzionale tra le due variabili: più alti erano i genitori, più alti risultavano i figli e viceversa.
- Tuttavia, ai genitori che si collocavano agli estremi (molto bassi o molto alti) non corrispondevano figli altrettanto estremi, ovvero Galton osservò che l'altezza dei figli si spostava verso la media.
- Quindi, concluse che questo costituiva una “regressione verso la media” e la relazione funzionale fu chiamata **modello di regressione**.



COS'È LA REGRESSIONE LINEARE?

- Ipotizziamo di dover rappresentare la relazione tra i voti degli studenti in base alle ore effettive di studio, potremmo scrivere:
Voto = 5 + ore di studio * 0,25
- Si stabilisce una relazione tra le variabili voto e ore di studio ed è un esempio molto semplice di **regressione lineare**, raffigurabile tramite una linea retta su un grafico cartesiano.
- La regressione lineare è un modello che assume una **relazione lineare** tra i valori di input (x) e un unico valore di output (y).
- La x è la **variabile indipendente** (le ore di studio), mentre la y è la **variabile dipendente** (i voti) e può essere calcolata come combinazione lineare dei valori di input.



COS'È LA REGRESSIONE LINEARE?

- In particolare, la **regressione lineare** viene definita **semplice** se è presente solamente una sola variabile di input (x), come nell'esempio appena mostrato;
- Al contrario se il numero di variabili è maggiore la regressione lineare è definita **multipla**.
- La regressione lineare è un metodo statistico molto utilizzato in machine learning, nelle scienze applicate e sociali, come ad esempio ingegneria, biologia, fisica ed economia.



PREREQUISITI DELLA REGRESSIONE LINEARE

- Per costruire un modello di regressione lineare è necessario conoscere:
 - **La correlazione (r).**
 - **Il coefficiente di determinazione (R).**
 - **La varianza (σ^2).**
 - **La deviazione standard (σ).**
 - **Il residuo.**



PREREQUISITI DELLA REGRESSIONE LINEARE

- **La correlazione (r):** spiega la relazione tra due variabili e assume possibili valori nell'intervallo compreso da -1 a $+1$.
- Due variabili con correlazione positiva si dicono direttamente correlate, mentre se la correlazione è negativa le due variabili si dicono inversamente correlate.
- Se infine la correlazione assume valore pari a zero le due variabili si dicono non correlate.

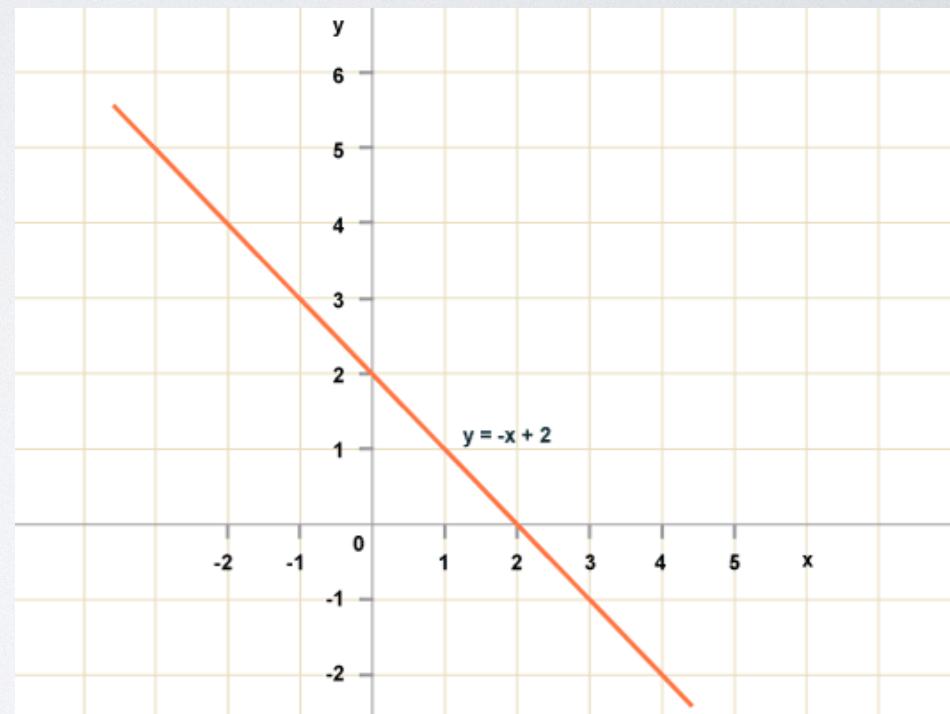


PREREQUISITI DELLA REGRESSIONE LINEARE

- **Il coefficiente di determinazione (R^2):** rappresenta una misura statistica di quanto i dati sono vicini alla linea di regressione. È dato dal quadrato della correlazione.
- **La varianza (σ^2):** rappresenta una misura dello spread nei dati, ossia la variabilità di un insieme di dati.
- **La deviazione standard (σ):** altro modo per misurare lo spread nei dati (in quanto esso rappresenta la radice quadrata della varianza).
- **Il residuo:** è una stima osservabile dell'errore statistico ed è dato dalla sottrazione tra il valore effettivo e il valore previsto.

MODELLO DI REGRESSIONE LINEARE SEMPLICE

- Un modello di regressione lineare semplice è composto dalla seguente equazione:
- **$y = mx + q$**
 - y = variabile dipendente
 - x = variabile indipendente
 - q = termine costante definito intercetta
 - m = coefficiente di relazione tra y e x
- In alcuni modelli viene aggiunto all'equazione un termine definito **errore statistico**, che si considera prossimo a zero.



ALGORITMI DI MACHINE LEARNING

- Ripasso sulla Regressione lineare
- **Uso di Excel**
- Implementazione in Python
- Impiego di SciKit-Learn
- Coefficiente di determinazione
- Errore Quadratico Medio

TROVARE LA RETTA DI REGRESSIONE LINEARE

- Trovare la retta di regressione significa semplicemente determinare il valore di q e m .
- Nella tabella la colonna x indica le **ore di studio** degli studenti (valore da 1 a 100), mentre la colonna y mostra i **voti assegnati**.

A	B	C	D
1	Studenti	x	y
2	1	95	85
3	2	85	95
4	3	80	70
5	4	70	65
6	5	60	70
7	6	55	62
8	7	65	75



A	B	C	D	E	F	G
1	Studenti	x	y	xy	X ²	Y ²
2	1	95	85	8075	9025	7225
3	2	85	95	8075	7225	9025
4	3	80	70	5600	6400	4900
5	4	70	65	4550	4900	4225
6	5	60	70	4200	3600	4900
7	6	55	62	3410	3025	3844
8	7	65	75	4875	4225	5625
9						
10	n, N	$\sum x$	$\sum y$	$\sum xy$	$\sum x^2$	$\sum y^2$
11	Totali	7	510	522	38785	38400
12						
13						
14						
15						
16	m	0,606321839				
17	q	30,39655172				
18	R ²	0,558760954				
19						



TROVARE LA RETTA DI REGRESSIONE LINEARE

- Pertanto, possiamo riassumere che **l'equazione di regressione lineare** nel nostro esempio sarà:
- $y = 0,6063x + 30,3966$

$$m = \frac{n \cdot (\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2}$$

$$q = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2}$$

$$R^2 = \frac{[n \sum (xy) - \sum x \sum y]^2}{[n(\sum x^2) - (\sum x)^2][n(\sum y^2) - (\sum y)^2]}$$

$$r = \frac{n \sum (xy) - \sum x \sum y}{\sqrt{[n(\sum x^2) - (\sum x)^2][n(\sum y^2) - (\sum y)^2]}}$$



ALGORITMI DI MACHINE LEARNING

- Ripasso sulla Regressione lineare
- Uso di Excel
- **Implementazione in Python**
- Impiego di SciKit-Learn
- Coefficiente di determinazione
- Errore Quadratico Medio

REGRESSIONE LINEARE SEMPLICE USANDO PYTHON



- I dati che vengono usati sono costituiti da due colonne, anni di esperienza e lo stipendio corrispondente.
- Si importano i pacchetti Python per l'analisi: NumPy, per aiutare con i calcoli matematici, Panda, per memorizzare e manipolare i dati e Matplotlib (opzionale), per tracciare i dati.



REGRESSIONE LINEARE SEMPLICE USANDO PYTHON

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

def linear_regression(x, y):
    N = len(x)
    x_mean = x.mean()
    y_mean = y.mean()
    B1_num = ((x - x_mean) * (y - y_mean)).sum()
    B1_den = ((x - x_mean)**2).sum()
    B1 = B1_num / B1_den
    B0 = y_mean - (B1*x_mean)
    reg_line = 'y = {} + {}β'.format(B0, round(B1, 3))
    return (B0, B1, reg_line)

def corr_coef(x, y):
    N = len(x)
    num = (N * (x*y).sum()) - (x.sum() * y.sum())
    den = np.sqrt((N * (x**2).sum() - x.sum()**2) * (N * (y**2).sum() - y.sum()**2))
    R = num / den
    return R

def predict(B0, B1, new_x):
    y = B0 + B1 * new_x
    return y
```

REGRESSIONE LINEARE SEMPLICE USANDO PYTHON

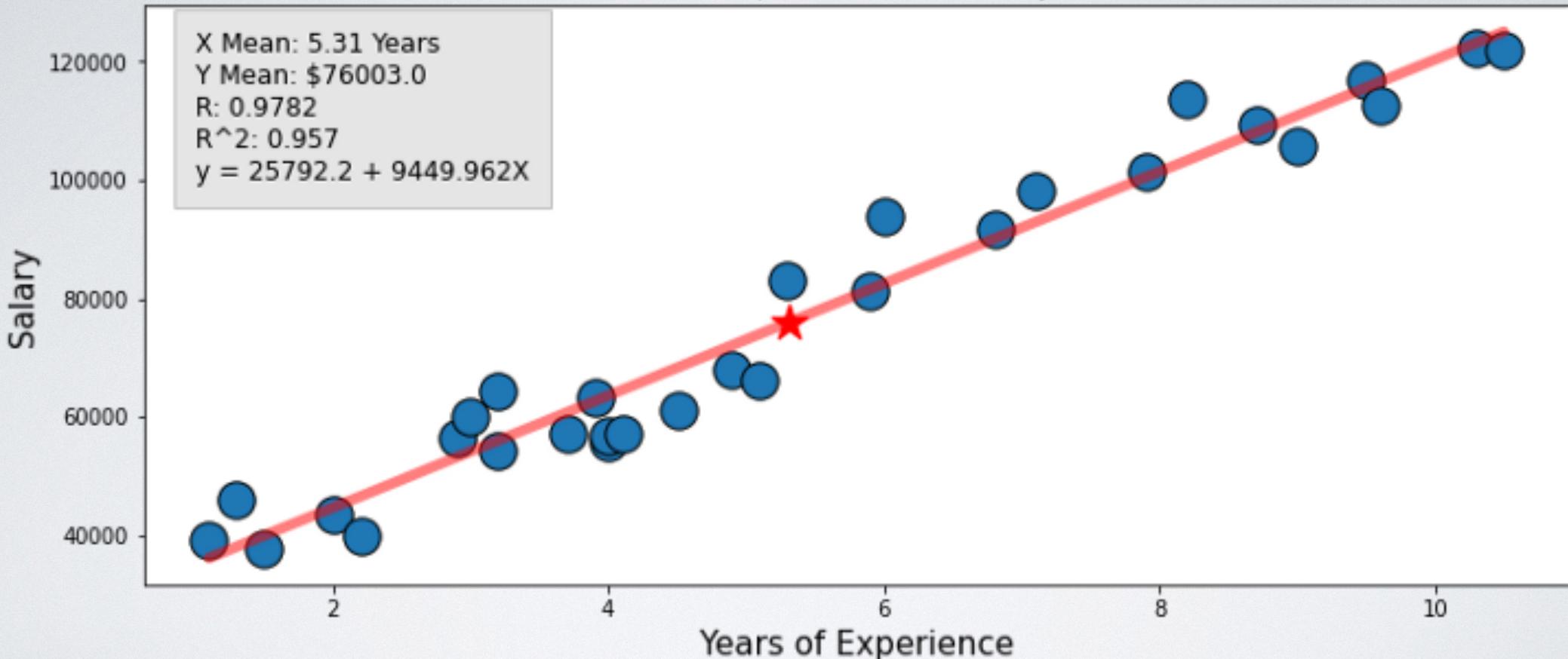
```
data = pd.read_csv('Salary_Data.csv')
x = data['YearsExperience']
y = data['Salary']

B0, B1, reg_line = linear_regression(x, y)
print('Regression Line: ', reg_line)
R = corr_coef(x, y)
print('Correlation Coef.: ', R)
print('Goodness of Fit: ', R**2)

plt.figure(figsize=(12,5))
plt.scatter(x, y, s=300, linewidths=1, edgecolor='black')
text = '''X Mean: {} Years Mean: ${}
R: {} ^2: {}
y = {} + {}x'''.format(round(x.mean(), 2), round(y.mean(), 2), round(R, 4),
                       round(R**2, 4), round(B0, 3), round(B1, 3))
plt.text(x=1, y=100000, s=text, fontsize=12, bbox={'facecolor': 'grey', 'alpha': 0.2, 'pad': 10})
plt.title('How Experience Affects Salary')
plt.xlabel('Years of Experience', fontsize=15)
plt.ylabel('Salary', fontsize=15)
plt.plot(x, B0 + B1*x, c = 'r', linewidth=5, alpha=.5, solid_capstyle='round')
plt.scatter(x=x.mean(), y=y.mean(), marker='*', s=10**2.5, c='r') # average point
plt.show()
```



How Experience Affects Salary



ALGORITMI DI MACHINE LEARNING

- Ripasso sulla Regressione lineare
- Uso di Excel
- Implementazione in Python
- **Impiego di SciKit-Learn**
- Coefficiente di determinazione
- Errore Quadratico Medio



COSTRUZIONE DEL MODELLO

- Si procede con l'importazione del **dataset di addestramento** per fare una prova da SciKit Learn.
- Come Dataset si scelgono 500 data points di dei prezzi di case a Boston, assegnando i dati alla variabile **dataset**

```
from sklearn.datasets import load_boston  
dataset = load_boston()
```
- La variabile dataset contiene una matrice con n+1 colonne, ossia n caratteristiche x e un prezzo y.
- Ogni riga del dataset di train è una casa con diverse caratteristiche x (features) e un prezzo y.



COSTRUZIONE DEL MODELLO

- In tutto ci sono tredici caratteristiche (tasso di criminalità, distanza dal fiume, % di zone residenziali, ecc.) e per vederle tutte si può effettuare una stampa.

```
print(dataset['DESCR'])
```

- Mentre per vedere i dati del dataset si usa

```
print(dataset['data'])
```

```
print(dataset['data'][0]) #per vederne uno solo (il primo)
```

- Si procede successivamente con la suddivisione della matrice in due parti. La variabile X contiene soltanto la matrice con i dati mentre la variabile y solo il vettore dei prezzi.

```
X=dataset['data']
```

```
y=dataset['target']
```



SUDDIVISIONE DEI DATI

- Si suddivide ulteriormente il dataset in un **insieme di training** e un **insieme di test** e per farlo si richiama la funzione **train_test_split()** di sklearn.

```
from sklearn.model_selection import train_test_split
```

- Prima di usarlo si fissa il generatore di numeri random a un seme per evitare la suddivisione casuale. È utile per avere sempre gli stessi risultati quando si lancia più volte l'addestramento del modello.

```
from numpy import random  
random.seed(0)
```

- Assegnamo la suddivisione alle variabili `X_train`, `y_train`, `X_test`, `y_test`, fissando la dimensione dell'insieme di test al 30% del dataset.

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3)
```



REGRESSIONE LINEARE

- Ora si importa l'algoritmo di regressione lineare da Scikit-Learn

```
from sklearn.linear_model import LinearRegression
```

- L'algoritmo LinearRegression ha due metodi principali:

- **fit()**: Addestra il modello usando i dati del dataset.
- **predict()**: Usa il modello per fare una previsione e rispondere a una query.
- In scikit-learn la classe ha lo stesso nome del modello di addestramento. Inoltre, i metodi fit() e predict() hanno lo stesso nome per ogni algoritmo. È un bel vantaggio perché permette di cambiare algoritmo di addestramento lasciando immutato il codice e i dati.



REGRESSIONE LINEARE

- Assegno la classe dell'algoritmo regressore a un'**istanza**.
`modello=LinearRegression()`
- L'istanza modello è la variabile in cui sarà registrato il modello al termine dell'addestramento.
- A questo punto **addestro il modello** con il metodo fit() passandogli in input le caratteristiche X (features) e i prezzi y dell'insieme di training.
`modello.fit(X_train,y_train)`



VERIFICA DEL MODELLO

- Per vedere se il modello funziona bene, ossia se calcola i prezzi con un margine d'errore accettabile si usa il metodo **predict()** per calcolare i prezzi delle case, passandogli soltanto la matrice delle caratteristiche X dell'insieme di test.
- Si salva il vettore con i prezzi previsti dal modello viene salvato nella variabile p.
p=modello.predict(X_test)
- Si verifica successivamente le differenze tra il vettore dei prezzi p, appena calcolato, e il vettore dei prezzi reali dell'insieme di test (y_test).



VERIFICA DEL MODELLO

- Si carico in memoria un algoritmo per calcolare l'**errore assoluto medio**.

```
from sklearn.metrics import mean_absolute_error
```

- Infine si utilizza per confrontare i due vettori.

```
m=mean_absolute_error(y_test,p)
```

- Nella variabile m viene salvato l'errore medio tra i due vettori. Quanto più basso è il valore m, tanto migliore è l'affidabilità del modello di regressione.



VERIFICA DEL MODELLO

- Nell'esecuzione del codice si potrebbe incorrere in una richiesta di rimodellamento dei dati, che sembra abbastanza inutile, tuttavia i modelli sci-kit sono costruiti per accettare solo input in un certo formato.
- La conversione di un numero in un array 2D esegue essenzialmente la seguente conversione:

np.array([32]).reshape(1, 1)



- Mettendo 32 nella posizione (0,0) di un array 2D.
- Un altro modo per convertire 32 in un array 2D è utilizzare quanto segue [[32]].



ESERCIZIO

- Stabilire sia usando Excel, sia usando il programma in Python (senza uso delle librerie) sia con SciKit-Learn, la regressione lineare del file WeatherData.csv.

Esiste una relazione tra la temperatura e l'umidità?

Indicare l'equazione della retta



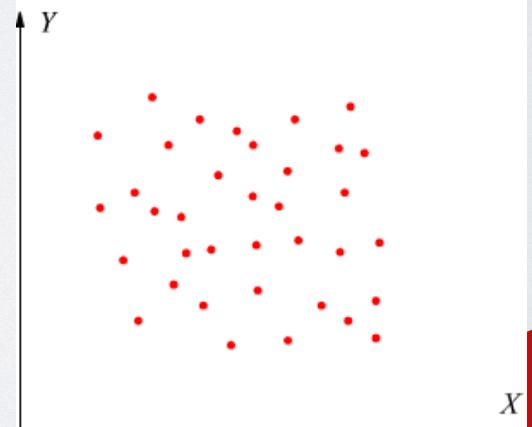
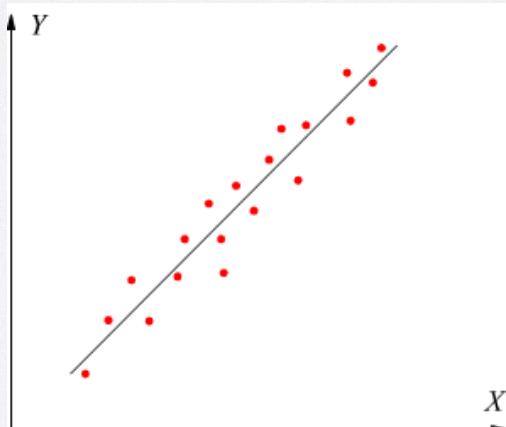
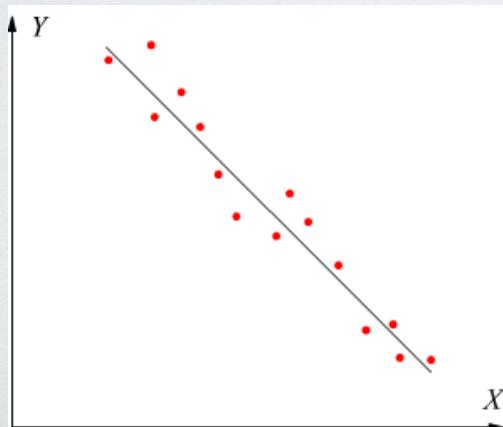
ALGORITMI DI MACHINE LEARNING

- Ripasso sulla Regressione lineare
- Uso di Excel
- Implementazione in Python
- Impiego di SciKit-Learn
- **Coefficiente di determinazione**
- Errore Quadratico Medio



COEFFICIENTE DI DETERMINAZIONE

- Se esiste una relazione lineare, i punti si distribuiscono vicino ad una retta.
- Se i punti sono molto dispersi, terzo schema, non esiste alcuna relazione





-1

correlazione negativa

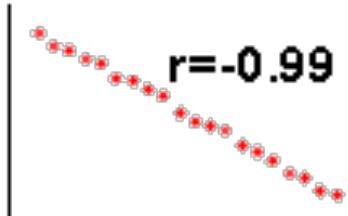
0

correlazione positiva

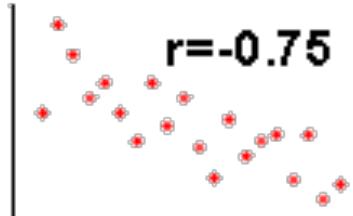
+1

assenza di
correlazione

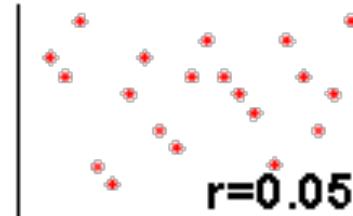
Esempi:



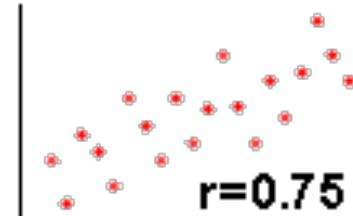
max correlazione
negativa



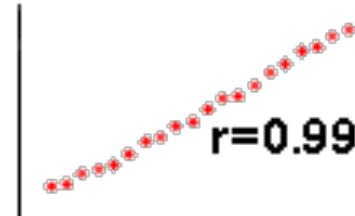
buona correlazione
negativa



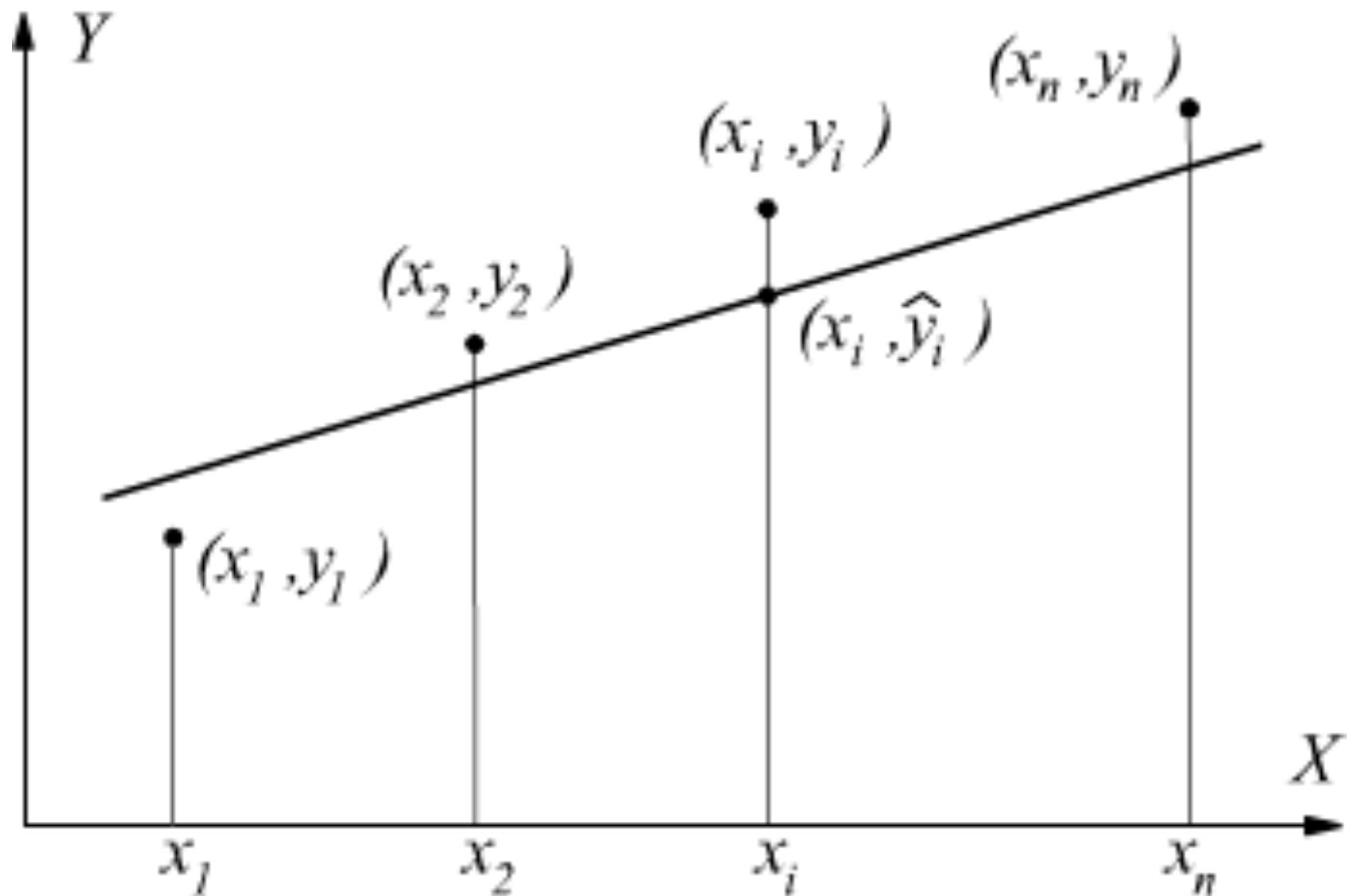
assenza di
correlazione



buona correlazione
positiva



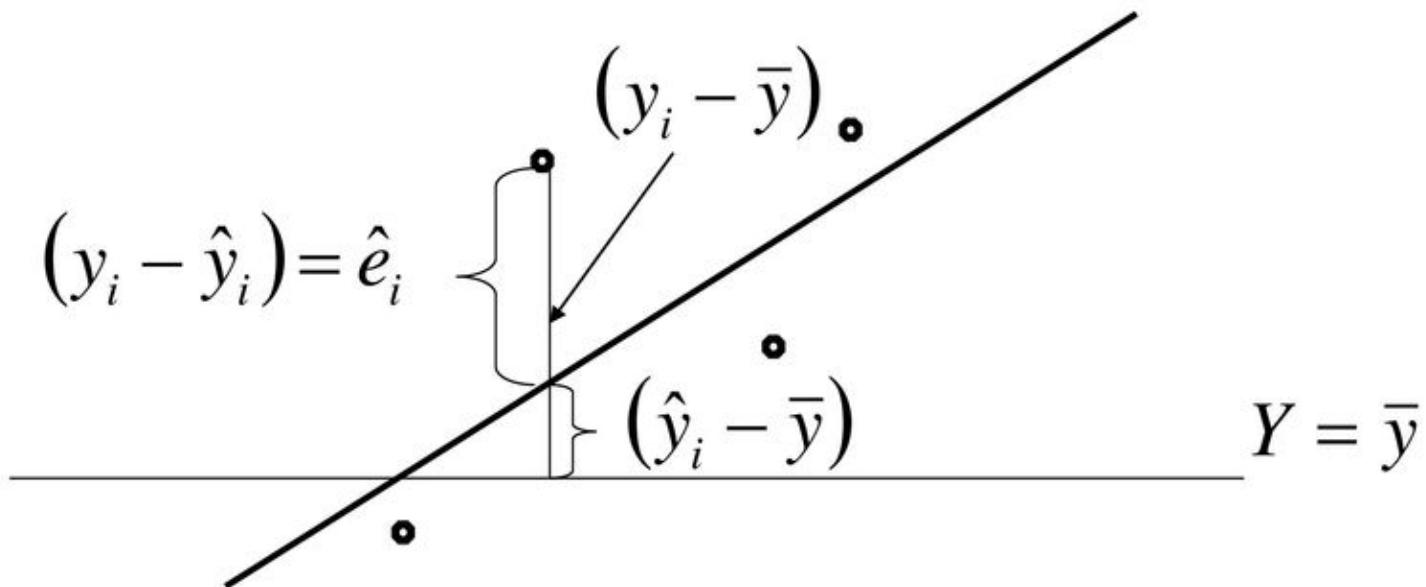
max correlazione
positiva





$$(y_i - \bar{y}) = (y_i - \hat{y}_i) + (\hat{y}_i - \bar{y})$$

$$\hat{Y} = \hat{\beta}_0 - \hat{\beta}_1 X$$





COEFFICIENTE DI DETERMINAZIONE

- Il **coefficiente di determinazione** R^2 dice quanto i dati previsti, indicati con \hat{y} , **spiega** i dati effettivi, indicato con y .
- In altre parole, rappresenta la forza dell'adattamento, tuttavia non dice nulla sul modello stesso - non dice se il modello è buono, se i dati sono distorti o se si è scelto il metodo di modellazione corretto.

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

y = valori delle variabili dipendenti

\hat{y} = valori previsti dal modello

\bar{y} = media di y



COEFFICIENTE DI DETERMINAZIONE

$$\bullet R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}$$

$ESS = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$ è la deviazione spiegata (Explained Sum of Squares)

$TSS = \sum_{i=1}^n (y_i - \bar{y})^2$ è la deviazione totale (Total Sum of Squares)

$RSS = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2$ è la deviazione residua (Residual Sum of Squares)



COEFFICIENTE DI DETERMINAZIONE

- In termini statistici esso è dato dalla frazione della varianza campionari di y predetta dai regressori x . In formule matematiche esso è dato dal rapporto tra due somme di quadrati:
 $R = ESS/TSS$ in cui ESS è la somma spiegata dei quadrati e TSS è la somma totale dei quadrati.
- La somma spiegata dei quadrati (ESS) è data dalla somma delle **differenze tra i valori predetti di y** e la media della stessa variabile dipendente.
- La somma totale dei quadrati (TSS) è invece data dalla somma delle **differenze tra i valori originari di y** e la media della stessa variabile.
- La formula può essere migliorata prendendo in considerazione la somma dei quadrati dei residui SSR, per cui si ha che:
 $R = 1 - (SSR/TSS)$ in cui SSR è la somma dei quadrati dei residui.



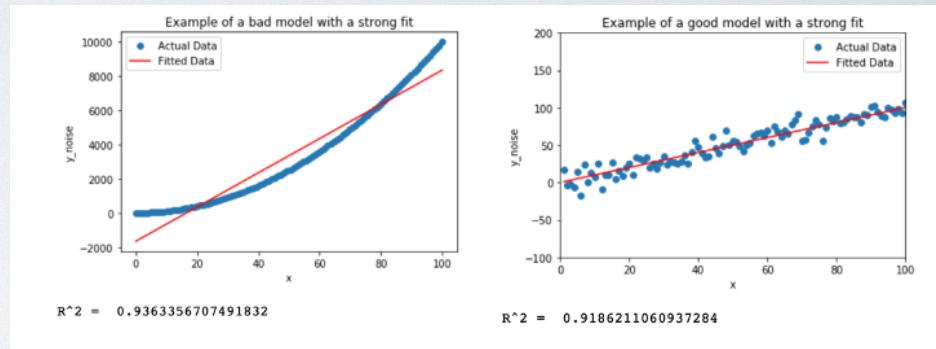
COEFFICIENTE DI DETERMINAZIONE

- Il valore R^2 varia da 0 a 1, con valori più alti che denotano un adattamento forte e valori più bassi che denotano un adattamento debole. In genere, si conviene che:
- $R^2 < 0,5 \rightarrow \text{Vestibilità debole}$
- $0,5 \leq R^2 \leq 0,8 \rightarrow \text{Adattamento moderato}$
- $R^2 > 0,8 \rightarrow \text{Forte adattamento}$



COEFFICIENTE DI DETERMINAZIONE

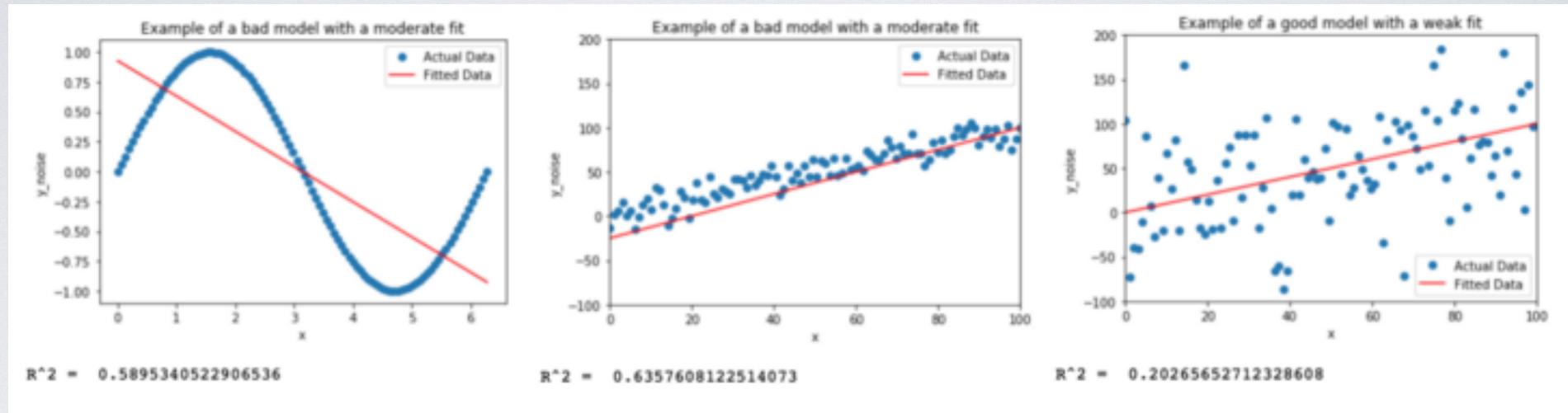
- R^2 con Forte adattamento non rappresenta quanto è buono il modello ma significa che, **in media**, i valori previsti (\hat{y}) non si discostano molto dai dati effettivi (y).
- Entrambi i grafici hanno un adattamento "forte", in quanto hanno valori R^2 elevati e catturano anche la piccola deviazione dei punti dati effettivi dalla linea adattata.
- Nonostante il modello di sinistra abbia un valore R^2 più alto, quello di destra è un modello migliore poiché non riesce a catturare la curvatura dei dati.



- **Un R^2 elevato non significa che l'adattamento sia buono o appropriato, significa semplicemente che la deviazione dei punti effettivi dai punti adattati, in media, è piccola.**



A volte, un modello può avere un valore R^2 basso, ma in realtà essere un buon modello per i dati.



A sinistra: una curva sinusoidale con una linea retta

Al centro: una linea retta con un piccolo adattamento al rumore con una linea retta volutamente inclinata

A destra: una linea retta con un grande adattamento al rumore con la linea corretta



COEFFICIENTE DI DETERMINAZIONE

- Il modello a sinistra ha una vestibilità terribile, ma con una moderata "forza di adattamento", quindi rispetto al modello a destra, si può pensare, basandosi unicamente sui valori R^2 , che il modello più a sinistra sia meglio. Questo è sbagliato, come dimostrano i grafici.
- E il modello medio? Ha un R^2 tre volte quello del modello sulla destra e, visivamente, non sembra essere completamente fuori strada.
- Quindi si potrebbe concludere che il modello medio è di gran lunga migliore del modello giusto? Sbagliato.



COEFFICIENTE DI DETERMINAZIONE

- I punti dati nel modello centrale e destro sono basati sulla stessa linea, $y = x + e$, dove e sono errori generati casualmente da una distribuzione normale.
- L'unica differenza tra loro è che le grandezze di errore sono amplificate nel grafico più a destra.
- Il modello medio è **peggiore** di quello a destra, perché è volutamente inclinato, in modo che l'equazione sia qualcosa del genere:

$$\hat{y} = 1,25 \cdot x - 25$$



COEFFICIENTE DI DETERMINAZIONE

- Il modello a destra invece è corretto: è $\hat{y} = \mathbf{x}$, esattamente la stessa della linea da cui sono stati generati i punti dati.
- Per confermarlo visivamente, si può vedere l'inclinazione sul modello centrale, mentre il modello giusto sembra essere il punto morto nei punti dati, esattamente come ti aspetteresti.
- **Pertanto, un modello con un R^2 basso potrebbe ancora prevedere correttamente la forma dei dati, ma soffre di una grande varianza nei dati.**
- Nonostante ciò, se la natura del problema è la previsione dei valori, il modello intermedio potrebbe funzionare meglio a causa della minore variazione dei punti dati, tuttavia questo non lo rende necessariamente un modello migliore.



ALGORITMI DI MACHINE LEARNING

- Ripasso sulla Regressione lineare
- Uso di Excel
- Implementazione in Python

Impiego di SciKit-Learn

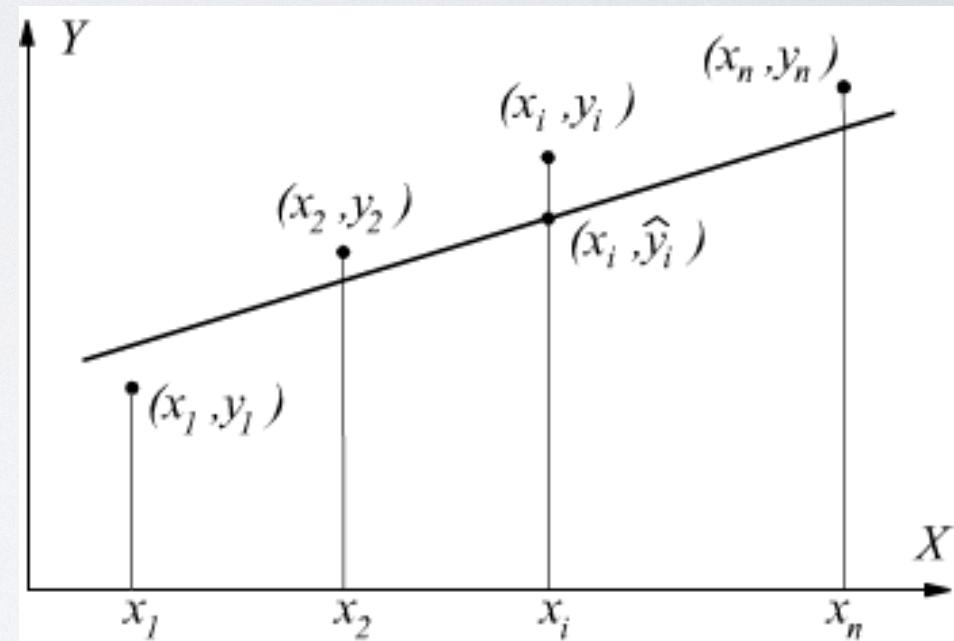
Coefficiente di determinazione

Errore Quadratico Medio



ERRORE QUADRATICO MEDIO

- Se in corrispondenza dei valori x rilevati si hanno y_i per i valori rilevati mentre per \hat{y}_i per i valori teorici
- Per il metodo dei minimi quadrati l'accostamento migliore viene ottenuto minimizzando la somma dei quadrati delle differenze tra i dati osservati e i dati teorici.
- $$\sum_{i=1}^n (y_i - \hat{y}_i)^2 = \min$$





ERRORE QUADRATICO MEDIO

- L'applicazione più comune del metodo dei minimi quadrati, consiste nel ricondurre il legame tra le due variabili x ed y ad una funzione lineare del tipo $y=a+bx$.
- Si deve dunque minimizzare la funzione:

$$\min[\phi(a, b)] = \sum_{i=1}^n (y_i - a - bx_i)^2$$



ERRORE QUADRATICO MEDIO

- per il calcolo del minimo, si dovranno valutare le derivate parziali prime rispetto ad a e b di questa funzione e dopo aver applicato gli opportuni criteri si dimostra che questa funzione ha minimo per i valori a, b:

$$b = \frac{n \cdot (\sum x_i y_i) - (\sum x_i)(\sum y_i)}{n(\sum x_i^2) - (\sum x_i)^2} = m \text{ (pendenza)}$$

$$a = \frac{(\sum y_i)(\sum x_i^2) - (\sum x_i)(\sum x_i y_i)}{n(\sum x_i^2) - (\sum x_i)^2} = q \text{ (intercetta)}$$



ERRORE QUADRATICO MEDIO

- l'equazione della retta interpolante è $y - \bar{y} = b(x - \bar{x})$
- Dove \bar{x} è il valore medio delle x , mentre \bar{y} è il valore medio della y e (\bar{x}, \bar{y}) è il baricentro della distribuzione
- è possibile dimostrare che i valori a e b possono essere espressi anche in funzione degli scarti x' ed y' :

$$a = \bar{y} - b\bar{x}$$

$$b = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} = \frac{\sum x'_i y'_i}{\sum (x'_i)^2} = m$$



ERRORE QUADRATICO MEDIO

- $b = \frac{\frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{n}}{\frac{\sum (x_i - \bar{x})^2}{n}} = \frac{\sigma_{xy}}{\sigma_x^2} = m$
- σ_x^2 = varianza di X mentre il numeratore viene definito covarianza di X ed Y
- $\sigma_{xy} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{n}$



ERRORE QUADRATICO MEDIO

- Mentre la varianza misura la variabilità dei valori di X rispetto al valor medio, la covarianza esprime la variabilità congiunta delle coppie (x,y) di valori corrispondenti rispetto al loro valor medio:

$$b = \frac{\sigma_{xy}}{\sigma_x^2}$$

- Il coefficiente b della retta interpolante esprime quindi la variabilità congiunta delle due variabili X ed Y rapportata alla variabilità della sola X.



ERRORE QUADRATICO MEDIO

- In questo caso è stata studiata la retta di regressione di Y rispetto ad X con b definito come coefficiente di regressione di Y rispetto ad X.
- In modo analogo si può studiare la retta di regressione di X rispetto ad Y, di equazione:

$$x=c+dy$$



ERRORE QUADRATICO MEDIO

- $a = \bar{y} - b\bar{x}$

$$b = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} = \frac{\sum x'_i y'_i}{\sum (x'_i)^2} = m$$

- La retta di regressione di X rispetto ad Y si può anche scrivere come
 $x - \bar{x} = d(y - \bar{y})$
- d viene detto coefficiente di regressione di X rispetto ad Y.



ERRORE QUADRATICO MEDIO

- I due coefficienti b e d hanno lo stesso segno, perché questo dipende dal numeratore che è uguale.
- Se b e d sono positivi quando una variabile cresce, cresce anche l'altra, se invece sono negativi, quando una variabile cresce, l'altra diminuisce.



ERRORE QUADRATICO MEDIO

- in analogia alla retta di regressione di Y rispetto ad X si può dimostrare che è

$$d = \frac{n \cdot (\sum x_i y_i) - (\sum x_i)(\sum y_i)}{n(\sum y_i^2) - (\sum y_i)^2}$$

$$c = \frac{(\sum x_i)(\sum y_i^2) - (\sum y_i)(\sum x_i y_i)}{n(\sum y_i^2) - (\sum y_i)^2}$$



ERRORE QUADRATICO MEDIO

- L'errore definito **residuo** che si ricava è pari alla differenza tra il valore reale e il valore previsto
- Se eleviamo al quadrato tale errore residuo ed eseguiamo per tutti gli altri campioni la stessa procedura, dividendo poi il risultato per il numero di campioni n, troviamo quello che è definito l'**errore quadratico medio o MSE (Mean squared error)**.



ERRORE QUADRATICO MEDIO

- L'MSE mostra quanto bene o male il modello descrive il set di dati analizzato.
- La formula per calcolarlo è la seguente:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$



ERRORE QUADRATICO MEDIO

- Più l'MSE è elevato e meno il modello rappresenta ciò che stiamo studiando e viceversa.
- Pertanto è bene cercare di ottenere un errore quadratico medio più piccolo possibile.
- Per raggiungere tale risultato si possono utilizzare i seguenti metodi (che tra l'altro ci permettono di determinare i migliori beta per il nostro modello).



METODO DEI MINIMI QUADRATI

- La procedura dei minimi quadrati (definita in letteratura **OLS** o **Ordinary Least Squares**, o semplicemente **Least Squares**) permette di determinare i valori dei coefficienti del vettore Beta che minimizzano l'errore quadratico medio (il quale prende il nome di **Minimum Mean Squared Error, MMSE**).
- Tale metodo dispone i dati dentro una matrice, e utilizzando le operazioni dell'algebra lineare, stima i valori ottimali di Beta.



METODO DEI MINIMI QUADRATI

- Questa tecnica funziona per **dataset non troppo grandi**. Infatti, nel caso si disponesse di un elevato numero di dati da analizzare, il tempo di esecuzione delle operazioni matriciali aumenterebbe a dismisura.
- Pertanto, non si consiglia l'utilizzo di questo metodo, qualora il dataset sia composto da più di **10 mila righe**.
- Si può valutare, per dataset più grandi, l'utilizzo della discesa del gradiente.



R² RIEPILOGO

- La metrica R² fornisce un'indicazione di quanto bene un modello si adatta ai dati, ma non è in grado di spiegare se il modello lineare è buono o meno
 - I. Fornisce un'indicazione di quanto è buona la vestibilità del modello
 - 2. L'aggiunta di predittori al modello può aumentare il valore di R² a causa del caso, rendendo i risultati fuorvianti (vedere **R² corretto**)
 - 3. L'aggiunta di predittori al modello può causare "overfitting", in cui il modello cerca di prevedere il "rumore" nei dati. Ciò riduce la sua capacità di ottenere prestazioni migliori con i "nuovi" dati che non ha visto prima.
 - 4. R² non ha significato per i modelli non lineari



ESERCIZIO

- Usando il file di regressione lineare semplice (versione senza SciKit-Learn), implementare il calcolo di MSE e confrontarlo con il valore MSE ottenuto in SciKit-Learn. Usare il file Weather.csv (Temperatura%Umidità)
- Fare uso del file diabetes_main.csv e fare una serie di indagini di regressione lineare mettendo a confronto le seguenti caratteristiche per ogni esercizio di apprendimento: (y=indice del glucosio nel sangue)
age e y,
BMI e y,
BP e y
Stampare il grafico di regressione lineare semplice, i valori di R^2 e MSE (Consegnare codice e screenshot del grafico con i valori R^2 e MSE)