UNIVERSITÀ DEGLI STUDI DI TRIESTE

Statistics

Linear Regression

Luca Pennella

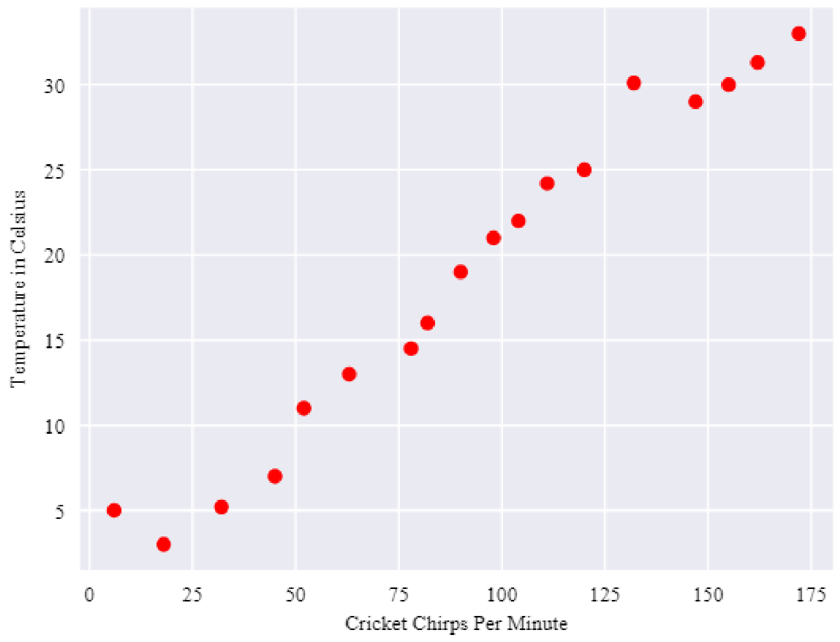
September 18th, 2024

Linear Regression

It has long been known that crickets (an insect species) chirp more frequently on hotter days than on cooler days. For decades, professional and amateur scientists have cataloged data on chirps-per-minute and temperature. As a birthday gift, your Aunt Ruth gives you her cricket database and asks you to learn a model to predict this relationship. Using this data, you want to explore this relationship.

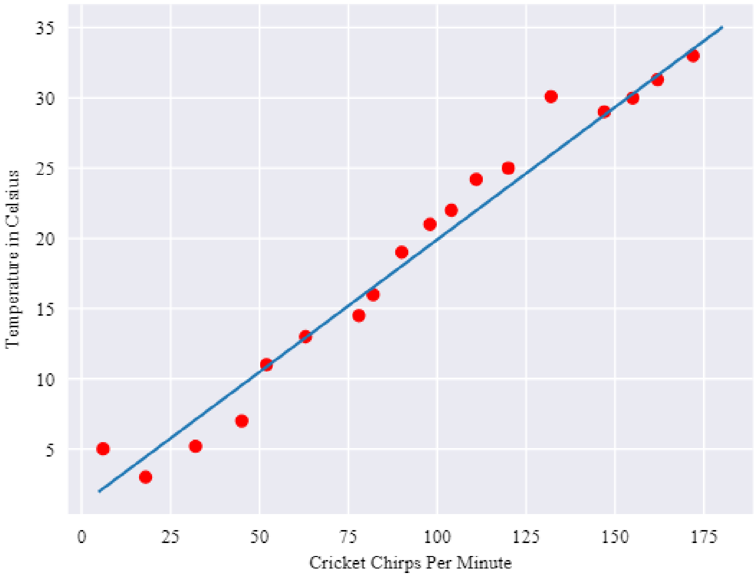
Linear Regression

First, examine your data by plotting it:



As expected, the plot shows the temperature rising with the number of chirps. Is this relationship between chirps and temperature linear? Yes, you could draw a single straight line like the following to approximate this relationship:

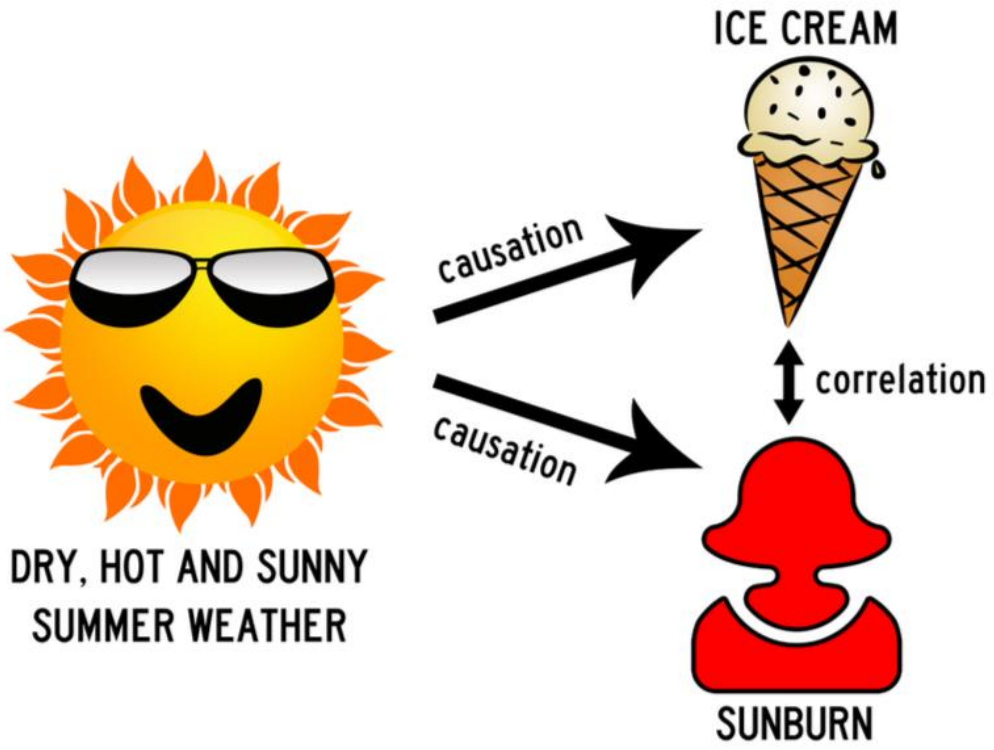
Linear Regression



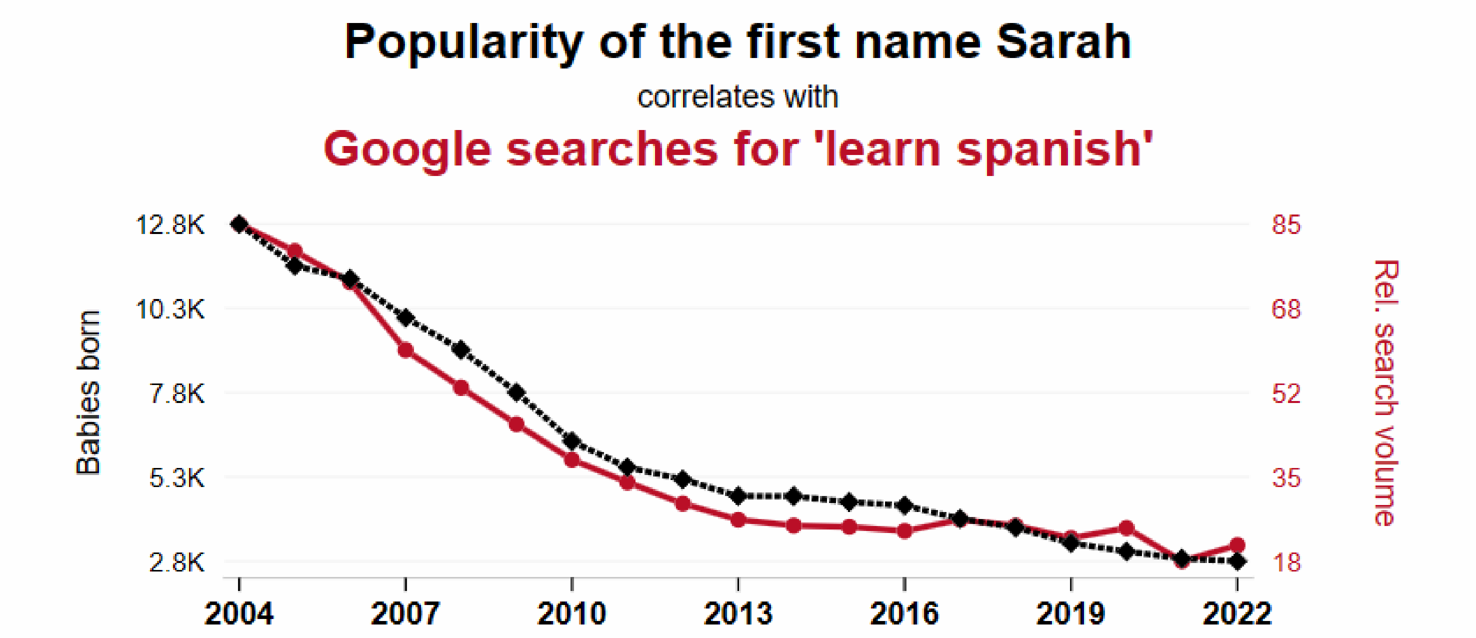
True, the line doesn't pass through every dot, but the line does clearly show the relationship between chirps and temperature. Using the equation for a line, you could write down this relationship as follows:

$$y = mx + b$$

Causality vs correlation: spurious correlations



Causality vs correlation: spurious correlations



Linear Regression

$$y = mx + b + \epsilon$$

- ***y*** is the temperature in Celsius—the value we're trying to predict.
- **m** is the slope of the line.
- **x** is the number of chirps per minute—the value of our input feature-
- **b** s the y-intercept.
- **ϵ** represents the error

By convention in machine learning, you'll write the equation for a model slightly differently:

$$y' = b + w_1 x_1 + e_1$$

where:

- ***y'*** is the predicted label (a desired output).
- **b** is the bias (the y-intercept), sometimes referred to as w_0
- $\boldsymbol{w_1}$ is the weight of feature 1.
- $\boldsymbol{x_1}$ is a feature (a known input).
- $\boldsymbol{e_1}$ is the residual.

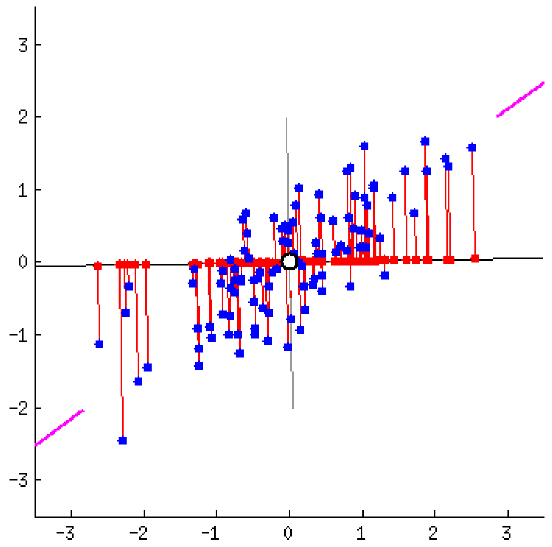
To **infer** (predict) the temperature ***y'*** for a new chirps-per-minute value $\boldsymbol{x_1}$, just substitute the $\boldsymbol{x_1}$ value into this model.

How can we tell if the line is right?

The residuals represent an estimate of the error.

$$e_i = y_i - \hat{y}_i$$

Please note, the estimated y is also reported with a 'hat' on its head.



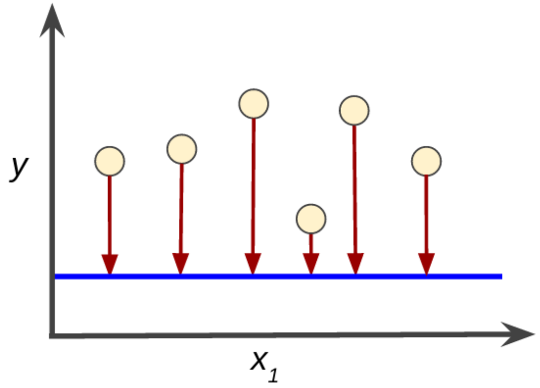
Linear Regression

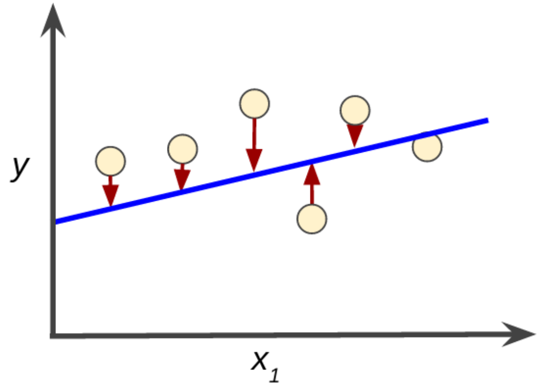
Training a model simply means learning (determining) good values for all the weights and the bias from labeled examples.
In supervised learning, a machine learning algorithm builds a model by examining many examples and attempting to find a model that minimizes loss.

Loss is the penalty for a bad prediction. That is, **loss** is a number indicating how bad the model's prediction was on a single example. If the model's prediction is perfect, the loss is zero; otherwise, the loss is greater. The goal of training a model is to find a set of weights and biases that have *low* loss, on average, across all examples.

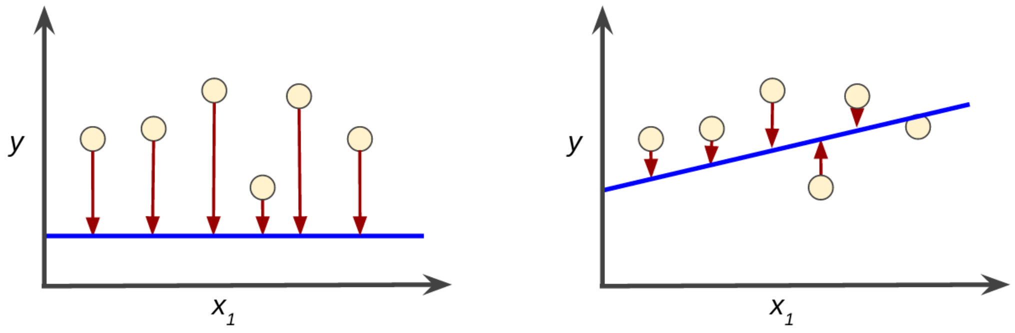
For example, Figure shows a high loss model on the left and a low loss model on the right. Note the following about the figure:

- The arrows represent loss.
- The blue lines represent predictions.





Linear Regression



Notice that the arrows in the left plot are much longer than their counterparts in the right plot. Clearly, the line in the right plot is a much better predictive model than the line in the left plot.

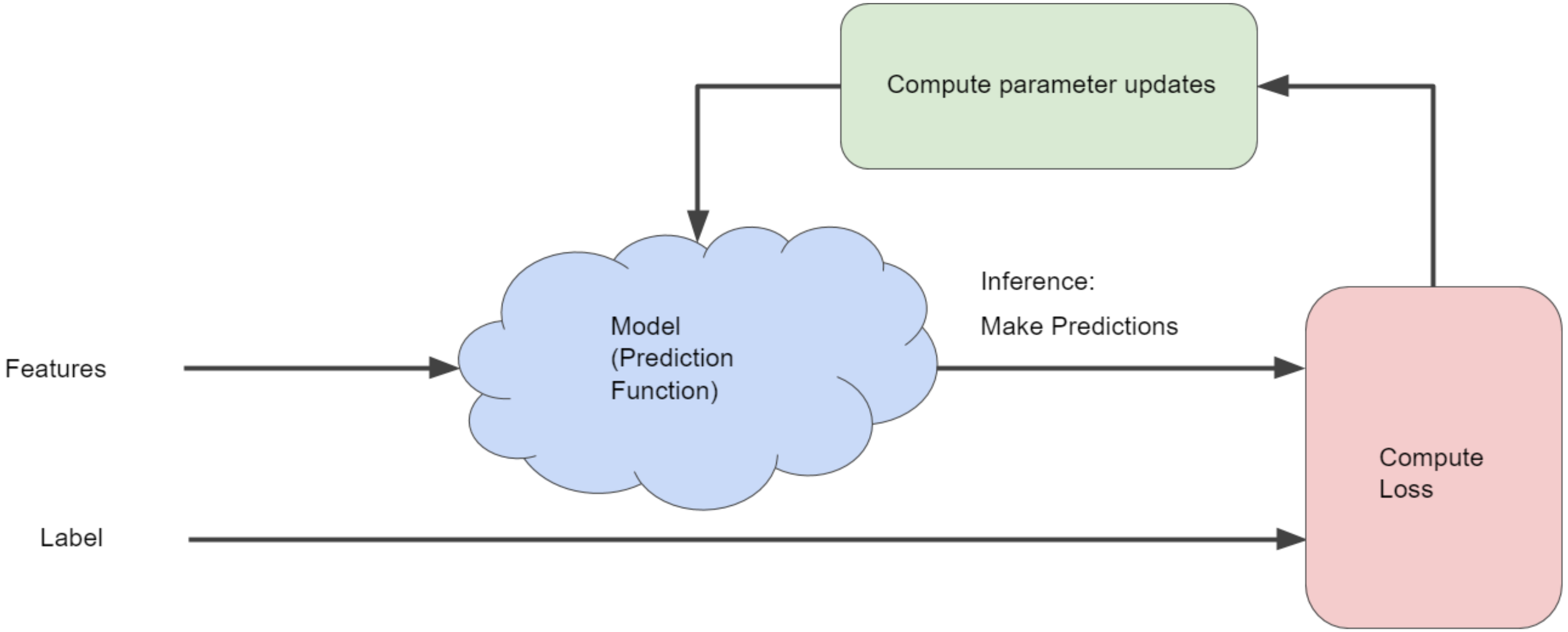
You might be wondering whether you could create a mathematical function—a loss function—that would aggregate the individual losses in a meaningful fashion.
The linear regression models we'll examine here use a loss function called **squared loss** (also known as $\mathbf{L_2}$ **loss**). The squared loss for a single example is as follows.

Mean square error (**MSE**) is the average squared loss per example over the whole dataset. To calculate MSE, sum up all the squared losses for individual examples and then divide by the number of examples:

$$MSE = \frac{1}{N} \sum_{(x,y)\in D} (y - prediction(x))^2$$

Loss function as game

In this game, the "hidden object" is the best possible model. You'll start with a wild guess ("The value of w_1 is 0.") and wait for the system to tell you what the loss is.
Then, you'll try another guess ("The value of w_1 is 0.5.") and see what the loss is. Aah, you're getting warmer. Actually, if you play this game right, you'll usually be getting warmer.
The real trick to the game is trying to find the best possible model as efficiently as possible.



Iterative strategies are prevalent in machine learning, primarily because they scale so well to large data sets.

Theoretical assumptions of the model

They are also verified by the graphs of the residuals, which reflect important properties.

- Linearity: $E(\epsilon_i) = 0$
- Homoschedasticity: $\mathrm{VAR}(\epsilon_i) = \sigma^2$
- Normality: $\epsilon_i \sim \mathcal{N}(0, \sigma^2) i.i.d.$
- Identifiability: xi not all equal $i = 1, \ldots, n$

Thus

$$y_i \sim \mathcal{N}(\alpha + \beta x_i; \sigma^2)$$

With yi independent

How do we assess the goodness of a model?

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

TSS = ESS + RSS

Total deviance = Deviance explained by the model + Deviance of the residuals

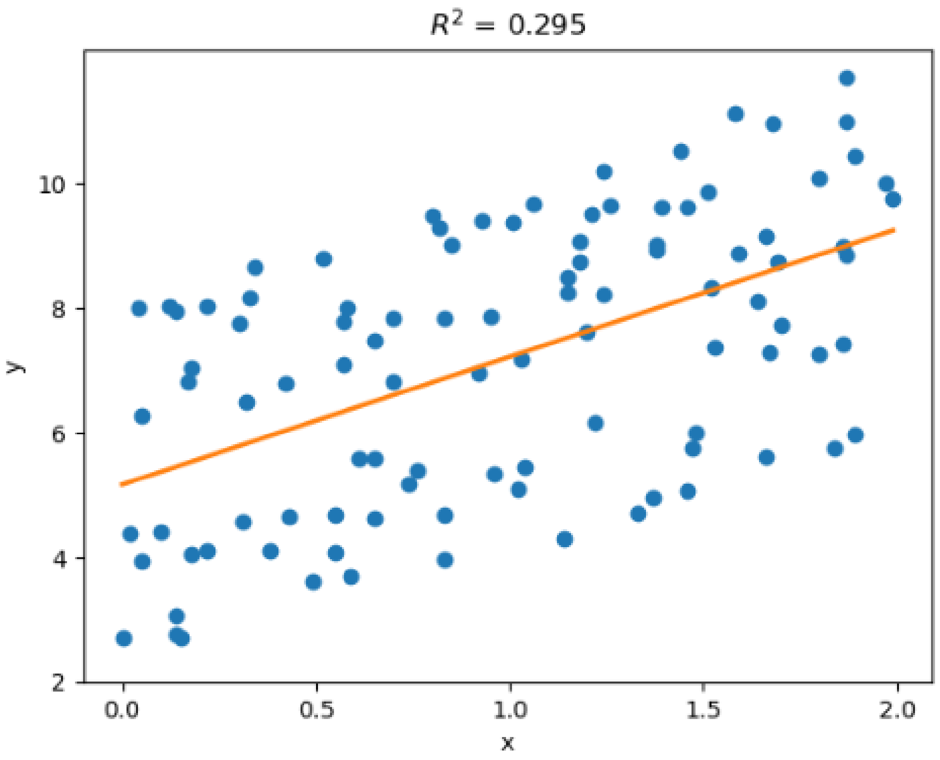
How do we assess the goodness of a model?

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

The linear determination index represents the amount of variability explained by the model.

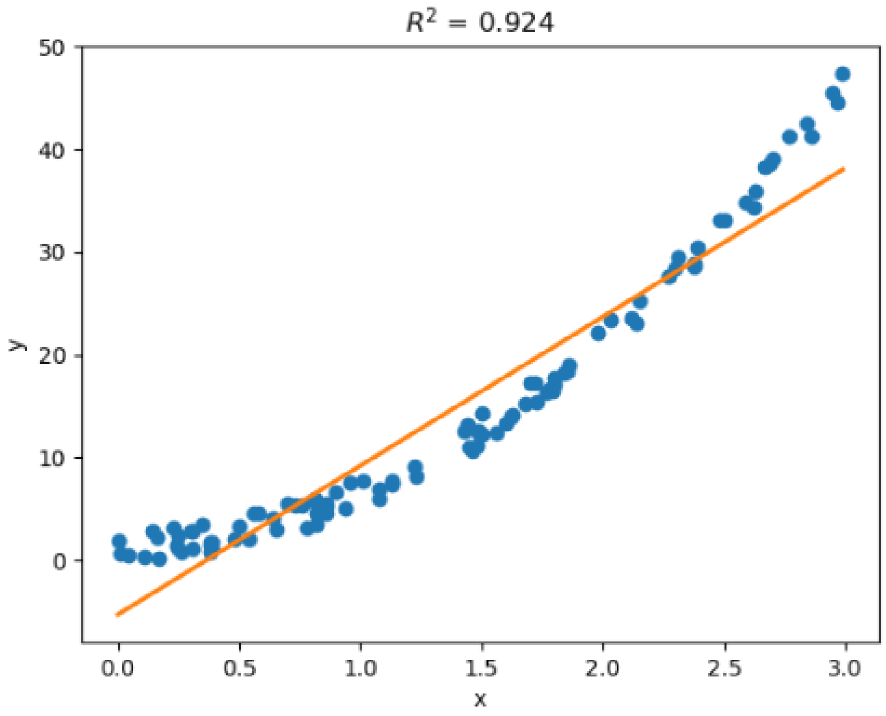
It takes values between 0 and 1: values closer to 1 are preferable.

A result from Python



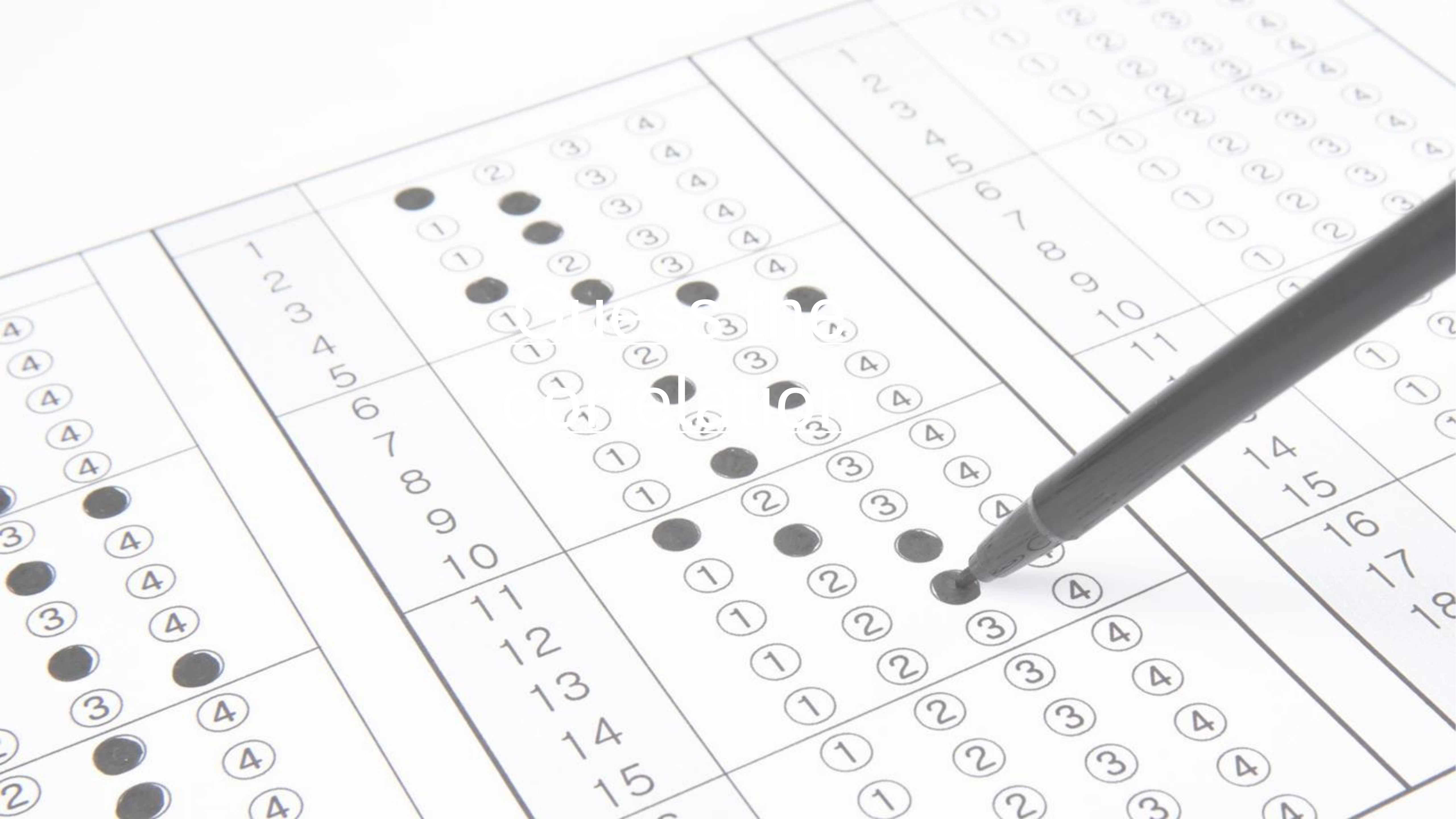

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.295
Model:                            OLS   Adj. R-squared:                  0.288
Method:                 Least Squares   F-statistic:                     41.06
Date:                Fri, 26 Jan 2024   Prob (F-statistic):           5.16e-09
Time:                        15:30:24   Log-Likelihood:                -204.11
No. Observations:                 100   AIC:                             412.2
Df Residuals:                      98   BIC:                             417.4
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          5.1683      0.365     14.143      0.000       4.443       5.893
x1             2.0530      0.320      6.408      0.000       1.417       2.689
==============================================================================
Omnibus:                       52.938   Durbin-Watson:                   1.934
Prob(Omnibus):                  0.000   Jarque-Bera (JB):                7.429
Skew:                          -0.170   Prob(JB):                       0.0244
Kurtosis:                       1.709   Cond. No.                         3.64
==============================================================================
```

A result from Python

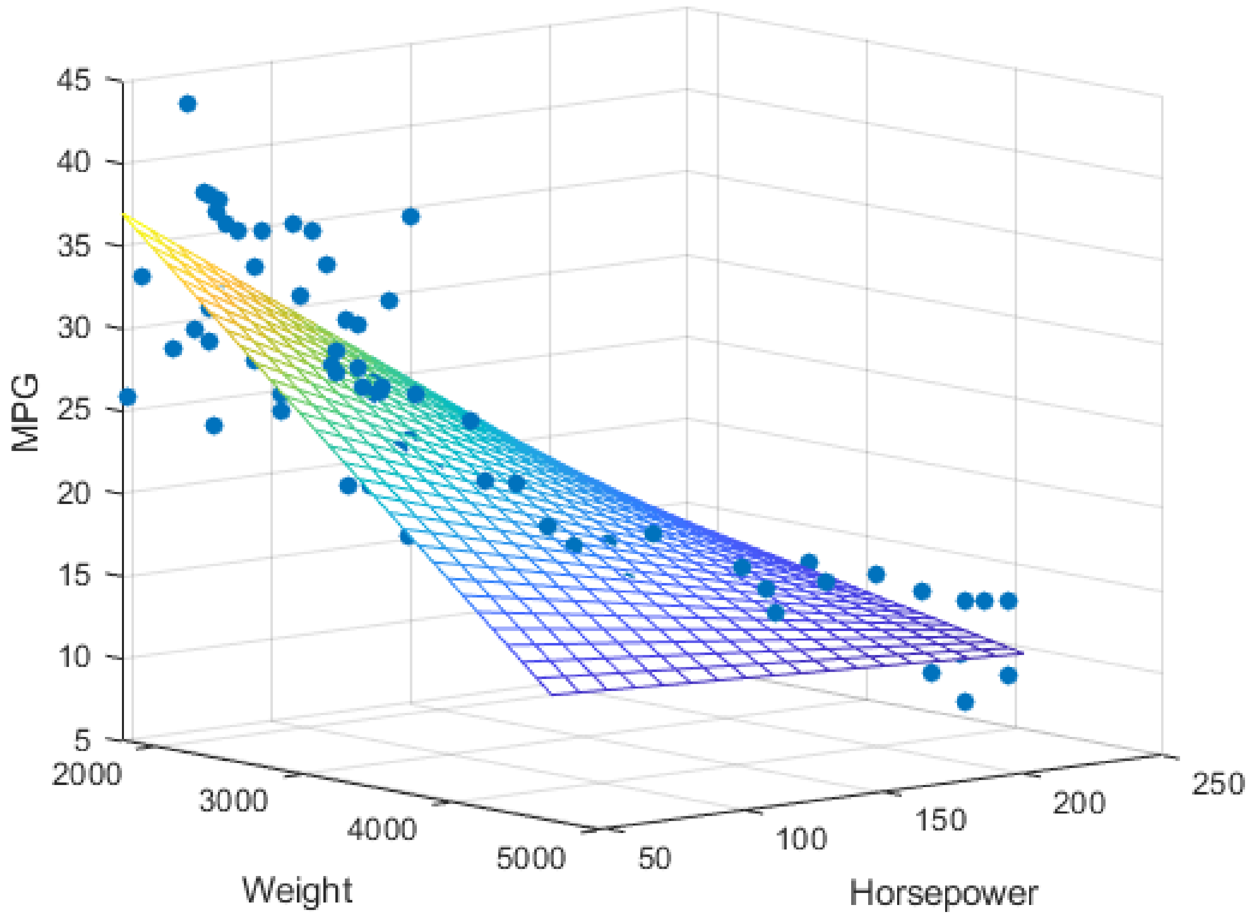



```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.924
Model:                            OLS   Adj. R-squared:                  0.923
Method:                 Least Squares   F-statistic:                     1185.
Date:                Fri, 26 Jan 2024   Prob (F-statistic):           1.57e-56
Time:                        15:31:05   Log-Likelihood:                -269.05
No. Observations:                 100   AIC:                             542.1
Df Residuals:                      98   BIC:                             547.3
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -5.3338      0.667     -7.994      0.000      -6.658      -4.010
x1            14.4993      0.421     34.417      0.000      13.663      15.335
==============================================================================
Omnibus:                        8.403   Durbin-Watson:                   1.906
Prob(Omnibus):                  0.015   Jarque-Bera (JB):                6.789
Skew:                           0.536   Prob(JB):                       0.0336
Kurtosis:                       2.307   Cond. No.                         3.84
==============================================================================
```

Guess the correlation



What if we wanted to introduce more than one explanatory variable?



UNIVERSITÀ DEGLI STUDI DI TRIESTE

Statistics

Multiple Linear Regression

Multiple Linear Regression

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \epsilon_i$$

Let us assume for the explanatory variables:

- x_{ij} are known and constant
- the variables are linearly independent of each other

In addition: $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$

How do we assess the goodness of a model?

The coefficient of linear determination R^2 grows monotonically as the number of covariates increases, even though they are not very influential.
A slightly modified index is used, which penalises models with a large number of variables.

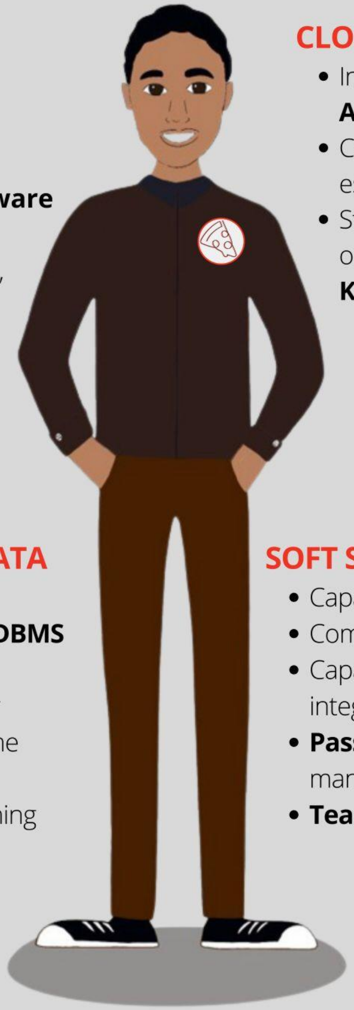
R^2 corrected (or adjusted)

$$\bar{R}^2 = 1 - \frac{n-1}{n-p-1} \cdot \frac{\mathrm{RSS}}{\mathrm{TSS}}$$

Data Analysis Jobs

Data Engineer

Il mago nel gestire i diversi flussi di dati



MATEMATICA & INFORMATICA

- Algebra lineare
- Calcolo differenziale
- Ingegneria del **software**
- Programmazione di buon livello **Python**, **Java o Scala**...
- Sistemi operativi come **Unix**
- Conoscenza **APIs**
- **Pipeline** per processare i dati

CLOUD COMPUTING

- Infrastruttura **Cloud** come **AWS**, **Azure**...
- Conoscenza Container, ad esempio **Docker**
- Strumenti di orchestrazione, come **Kubernetes**

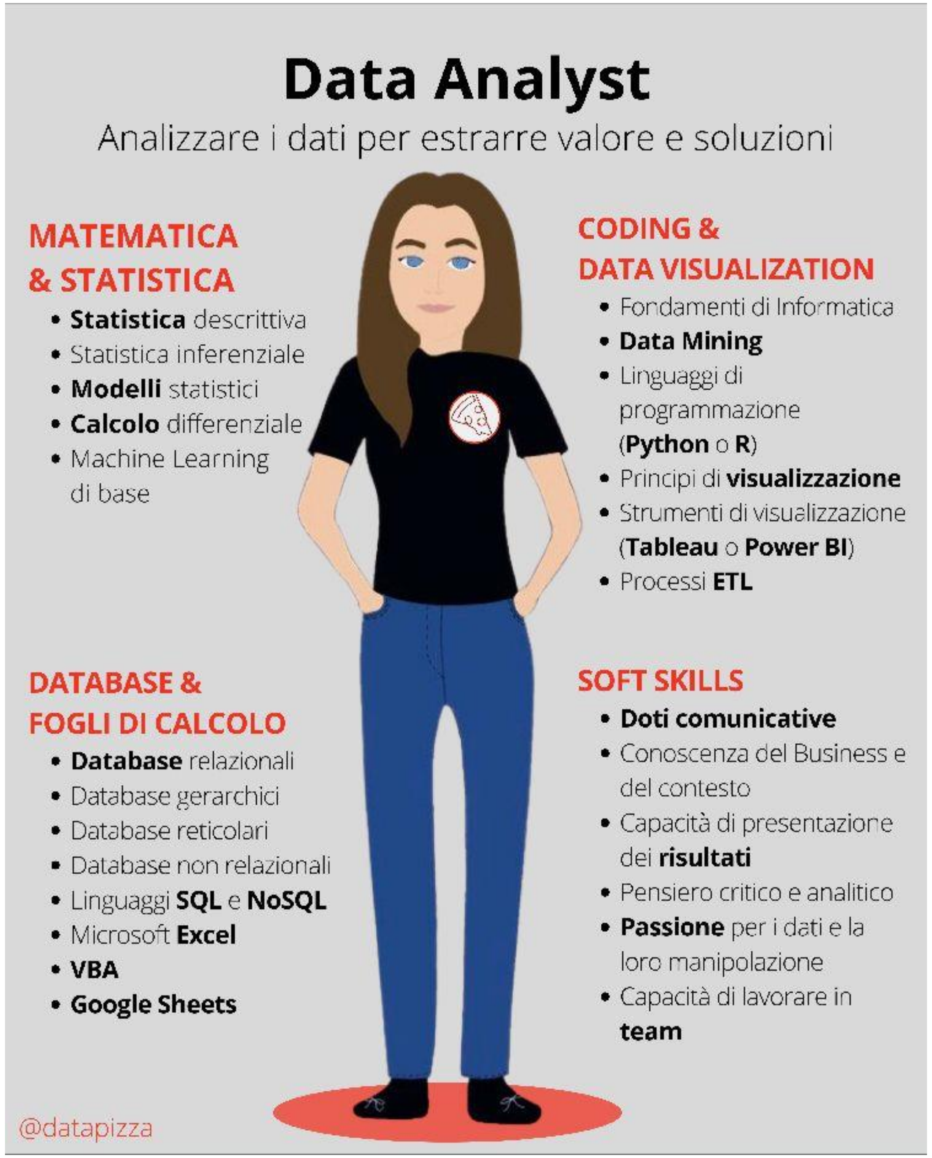
DATABASE & BIG DATA

- Processi di **ETL**
- Conoscenza sistemi **DBMS**
- **SQL** & **NoSQL**
- Software specifici per gestire **Big Data** come **Hadoop**, **Spark**...
- Gestire dati in streaming (**Apache Kafka**)

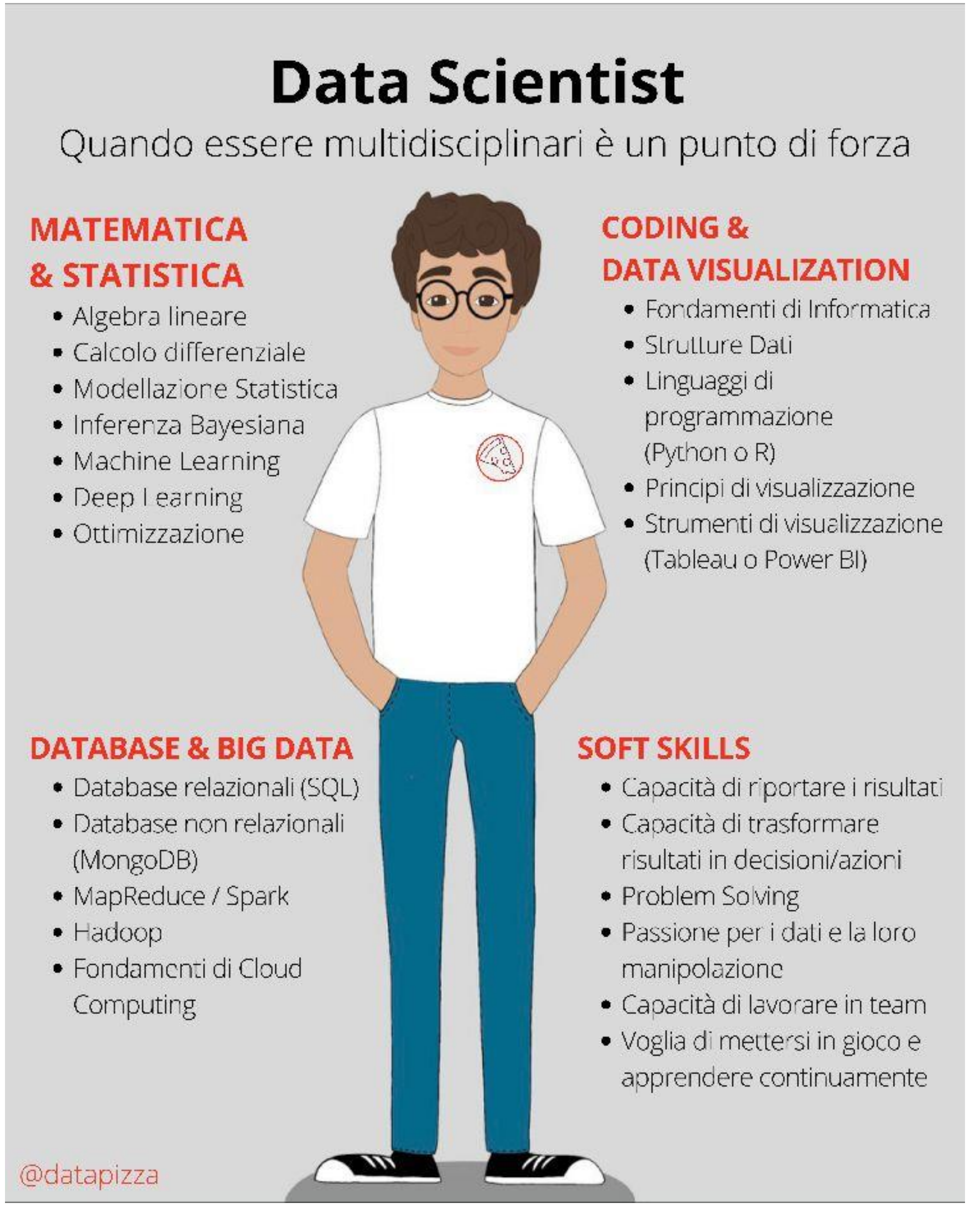
SOFT SKILLS

- Capacità **analitiche**
- Comprensione del **prodotto**
- Capacità di lavorare e integrare **tecnologie** diverse
- **Passione** per i dati e la loro manipolazione
- **Team** building

@datapizza


Data Analyst
Analizzare i dati per estrarre valore e soluzioni
MATEMATICA & STATISTICA
• Statistica descrittiva
• Statistica inferenziale
• Modelli statistici
• Calcolo differenziale
• Machine Learning di base
CODING & DATA VISUALIZATION
• Fondamenti di Informatica
• Data Mining
• Linguaggi di programmazione (Python o R)
• Principi di visualizzazione
• Strumenti di visualizzazione (Tableau o Power BI)
• Processi ETL
DATABASE & FOGLI DI CALCOLO
• Database relazionali
• Database gerarchici
• Database reticolari
• Database non relazionali
• Linguaggi SQL e NoSQL
• Microsoft Excel
• VBA
• Google Sheets
SOFT SKILLS
• Doti comunicative
• Conoscenza del Business e del contesto
• Capacità di presentazione dei risultati
• Pensiero critico e analitico
• Passione per i dati e la loro manipolazione
• Capacità di lavorare in team
@datapizza

Data Analysis Jobs



Data Journalist

- Wikipedia (some examples of enquiry and historical background).
- Deck of UniSalento slides with some historical background on Data Journalism and 'lean' and interesting examples of the application of data analysis in journalism/information