
DATA JOBS

Carlos Fernando Vásquez Guerra



DATA JOBS

MODERN DATA SCIENTIST

Data Scientist, the sexiest job of 21th century requires a mixture of multidisciplinary skills ranging from an intersection of mathematics, statistics, computer science, communication and business. Finding a data scientist is hard. Finding people who understand who a data scientist is, is equally hard. So here is a little cheat sheet on who the modern data scientist really is.

MATH & STATISTICS

- ★ Machine learning
- ★ Statistical modeling
- ★ Experiment design
- ★ Bayesian inference
- ★ Supervised learning: decision trees, random forests, logistic regression
- ★ Unsupervised learning: clustering, dimensionality reduction
- ★ Optimization: gradient descent and variants



PROGRAMMING & DATABASE

- ★ Computer science fundamentals
- ★ Scripting language e.g. Python
- ★ Statistical computing package e.g. R
- ★ Databases SQL and NoSQL
- ★ Relational algebra
- ★ Parallel databases and parallel query processing
- ★ MapReduce concepts
- ★ Hadoop and Hive/Pig
- ★ Custom reducers
- ★ Experience with xaaS like AWS

DOMAIN KNOWLEDGE & SOFT SKILLS

- ★ Passionate about the business
- ★ Curious about data
- ★ Influence without authority
- ★ Hacker mindset
- ★ Problem solver
- ★ Strategic, proactive, creative, innovative and collaborative

COMMUNICATION & VISUALIZATION

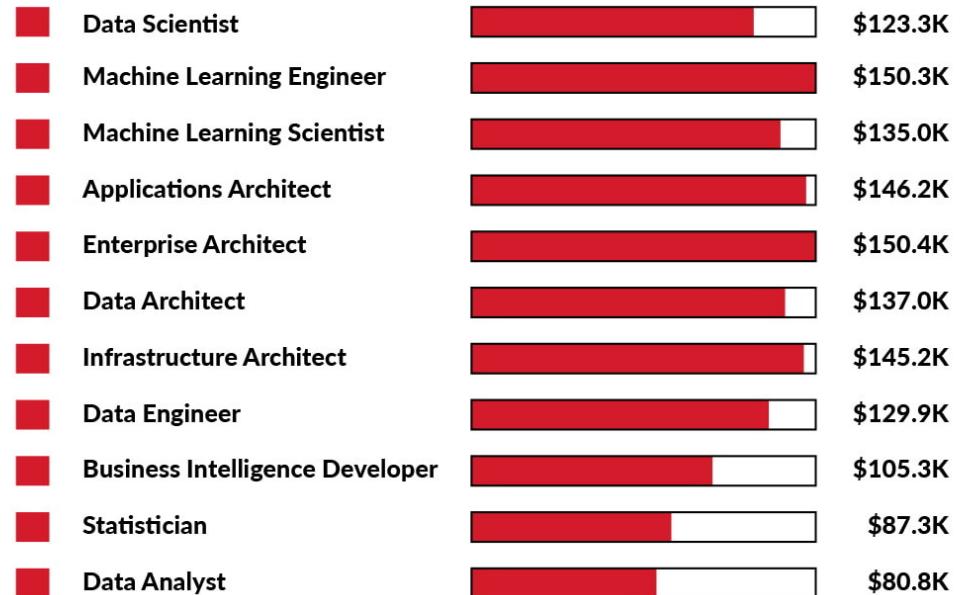
- ★ Able to engage with senior management
- ★ Story telling skills
- ★ Translate data-driven insights into decisions and actions
- ★ Visual art design
- ★ R packages like ggplot or lattice
- ★ Knowledge of any of visualization tools e.g. Flare, D3.js, Tableau

https://medium.com/@_moazzemhossain/the-10-algorithms-data-scientist-must-have-to-know-97a2c478ce94



Northeastern University

Salaries of In-Demand Data Science Jobs



Source: Lightcast™ Analyst, 2023

www.northeastern.edu

<https://graduate.northeastern.edu/knowledge-hub/data-science-careers-shaping-our-future/>

DATA JOBS

> The Anatomy of a Data Team — Different Data Roles

Most commonly used tools	Possible Job Titles	Beginner Skill Level	Intermediate Skill Level	Advanced Skill Level	Yearly Salary Range*	Learn on DataCamp	Get Hired on DataCamp Jobs*	
Data Consumers Data consumers use data to make data-driven decisions, and actively have conversations with data practitioners.	Business Intelligence tools Tableau or Power BI Spreadsheets Excel or Google Sheets	Chief Marketing Officer Human Resources Manager Manager of Sales and Business Development	Understands what data scientists, machine learning scientists, and data analysts do Knows which questions can (and can't) be answered with data Interpret the results of data projects, including calculations and visualizations.	Is able to calculate descriptive statistics Can draw common data visualizations Understands the business applications of data	Has a strong grasp of the fundamentals of business intelligence	NA	10 SKILL TRACKS	Coming soon
Business Analysts Business Analysts are responsible for using data insights to actionably result in better outcomes. They have deep knowledge of the business domain, and use a mix of strategic, analytical, and strategic non-coding tools to communicate insights derived from data.	Business Intelligence tools Tableau or Power BI Databases SQL Spreadsheets Excel or Google Sheets	Business Analyst Marketing Analyst Data Analyst Supply Chain Analyst	Is able to calculate descriptive statistics Can draw common data visualizations Understands the business applications of data	Has a deep knowledge of the business domain Is able to report and communicate using data	Can create dashboards Organizes data to solve business questions	77K	10 SKILL TRACKS	Coming soon
Data Analysts Similar to Business Analysts, Data Analysts are responsible for extracting data insights from their organization's data. They have a deep understanding of how data flows through a company's workflow and report their insights through a combination of coding and communicating skills.	Programming languages R or Python Business Intelligence tools Tableau or Power BI Databases SQL Spreadsheets Excel or Google Sheets	Business Analyst Marketing Analyst Data Analyst Supply Chain Analyst	Is able to calculate descriptive statistics Can draw common data visualizations Understands the business applications of data	Perform the data analysis workflows, including importing, manipulating, cleaning, calculating, and reporting on business data Has a strong grasp of business intelligence tools	Can create dashboards Organizes data to solve business questions	69K	10 SKILL TRACKS	Coming soon
Data Scientists Data Scientists investigate, extract, and report meaningful insights in the context of business problems to provide these insights to non-technical stakeholders. They have a deep understanding of machine learning workflows and machine learning concepts. They work exclusively with coding tools, conduct complex, one-off work with big data tools.	Programming languages R or Python Databases SQL Command line tools Git or Bash Big data tools Apache or Spark	Data Scientist Analytics Engineer Data Analyst	Is able to calculate descriptive statistics Can draw common data visualizations Understands the business applications of data	Understands fundamental statistics, including distributions, modeling, and inference Designing simple experiments such as A/B tests Understands the business applications of data	Applies analytics to business applications such as finance, marketing, and healthcare Understands supervised and unsupervised machine learning workflows Work with non-standard data types, such as time series, text, geospatial, and images.	117K	10 SKILL TRACKS	Coming soon
Machine Learning Scientists Machine Learning Scientists design and develop machine learning systems that make predictions from the organization's data. They are responsible for the customer churn and lifetime value and are responsible for developing models for the organization. They work exclusively with coding-based tools.	Programming languages R or Python Databases SQL Command line tools Git or Bash	Data Scientist Research Scientist Machine Learning Engineer	Perform the data analysis workflows, including importing, manipulating, cleaning, calculating, and reporting on business data	Performing supervised and unsupervised machine learning workflows including feature engineering, training models, testing goodness of fit, and making predictions Applies analytics to business applications such as finance, marketing, and healthcare	Perform deep learning workflows Work with non-standard data types, such as time series, text, geospatial, and images. Deploy machine learning models in production	137K	10 SKILL TRACKS	Coming soon
Statisticians Similar to Data Scientists, Statisticians are responsible for extracting data insights and designing and monitoring experiments such as A/B testing. They focus on quantifying uncertainty and presenting findings in a clear and concise manner, often in finance or healthcare.	Programming languages R Python Scala Databases SQL	Quantitative Analyst Data Scientist Clinical Data Analyst	Perform the data analysis workflows, including importing, manipulating, cleaning, calculating, and reporting on business data	Perform statistical modeling workflows, including importing, manipulating, cleaning, calculating, and inferring significance Test hypotheses and design simple experiments such as A/B tests	Design more complex experiments and understand Bayesian statistics Understand specialist models, such as survival models, generalized additive models, mixture models, structural equation models	89K	10 SKILL TRACKS	Coming soon
Programmers Programmers are highly technical individuals that work on data teams and work on submitting code to an organization's data. They bridge the gap between data science and data engineering, and data science and have a thorough understanding of designing and sharing code at scale.	Programming languages R Python Scala Databases SQL Command line tools Git or Shell	Software Engineer Data Scientist Dev-Ops Engineer	Write functions to avoid repetitive code Benchmark and optimize code to improve performance	Develop best practices for testing code Work with web APIs Develop packages for sharing code	Develop data pipelines and work with parallel programming Understand programming paradigms, such as functional programming and object-oriented programming	108K	10 SKILL TRACKS	Coming soon
Data Engineers Data Engineers are responsible for getting the right data into the hands of the right people. They use their technical skills to take terabytes of raw data coming from multiple sources and transform it into location, clean, relevant data for the organization.	Programming languages R Python Scala Databases SQL Command line tools Git or Shell Big data tools Apache or Spark Cloud platforms AWS GCP Azure	Data Engineer Software Engineer Dev-Ops Engineer	Efficiently extract, transform, and load data	Process data and automate data flow using the command line Process data in the cloud	Manage, optimize and process big datasets and large databases	112K	10 SKILL TRACKS	Coming soon

* Salary data sourced in USD from Glassdoor in the USA
At the time of publishing, DataCamp Jobs provides access to data scientist and data analyst roles, with support for more roles coming in the future



• Data Analyst vs. Data Scientist: A Comparative Guide For 2025

• What's the Difference Between a Data Analyst and a Data Scientist?

• The Top 10 Data Analytics Careers For 2025: Skills, Salaries & Career Prospects

• Data Analyst vs Data Scientist vs Data Engineer: Roles, Skills & Salary Comparison

• 12 Data Science Job Titles — Which Role Is Right for You?

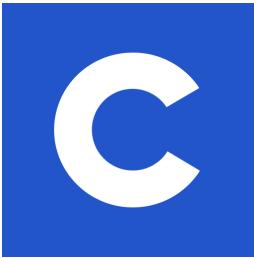
EXCEL

Carlos Fernando Vásquez Guerra



CURSOS Y MÁS

Cursos



VBA

[Curso Excel Programación en Macros VBA desde cero Especial](#)

[VBA Excel: How to Get Started and Make Your Work Easier](#)

Libros

O'REILLY

Excel Cookbook

Recipes for Mastering Microsoft Excel

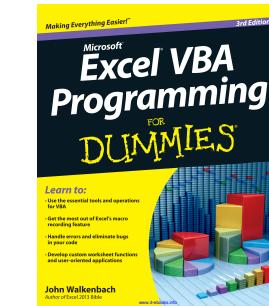
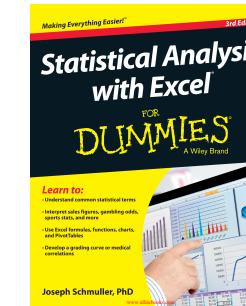
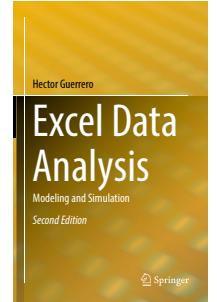


Dawn Griffiths

Excel Scientific and Engineering Cookbook



David M. Bourg



Tips

[The 15 Basic Excel Formulas Everyone Needs to Know](#)

FORECASTING

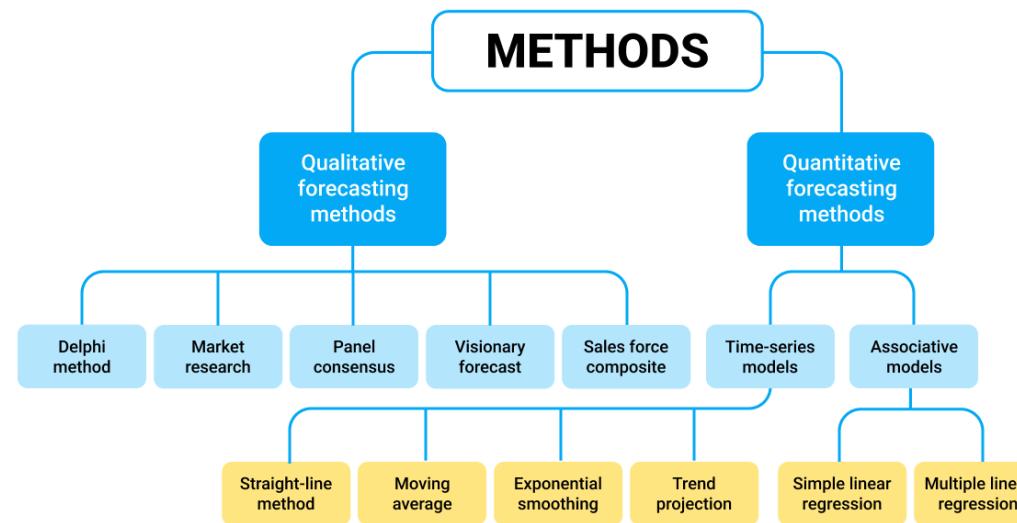
Carlos Fernando Vásquez Guerra



FORECASTING

Forecasting methods

Clockify



<https://clockify.me/forecasting-models>

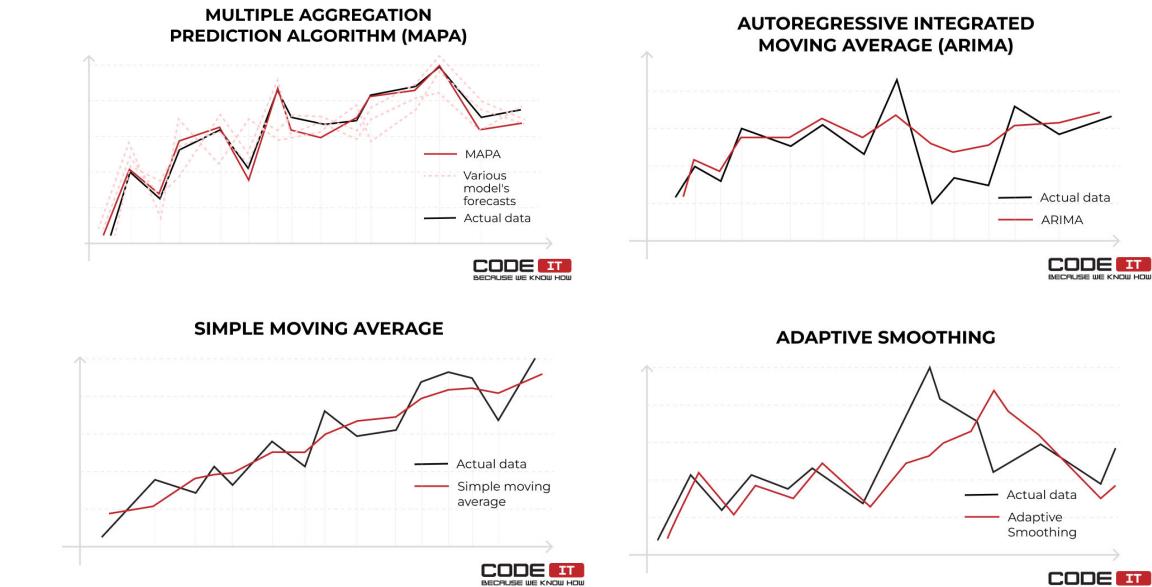
TIME SERIES

Decomposition



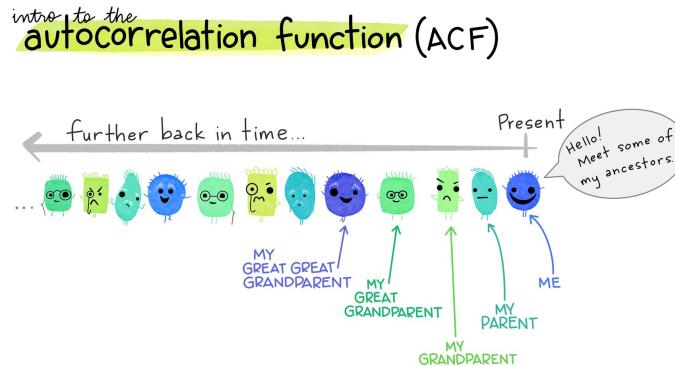
<https://www.linkedin.com/pulse/time-series-decomposition-using-r-harshal-chaudhari/>

Models



<https://codeit.us/blog/forecasting-in-supply-chain>

ACF & PACF

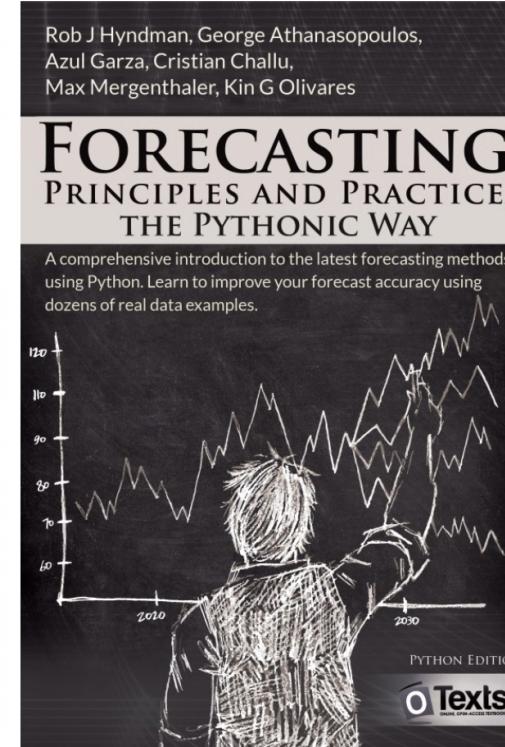
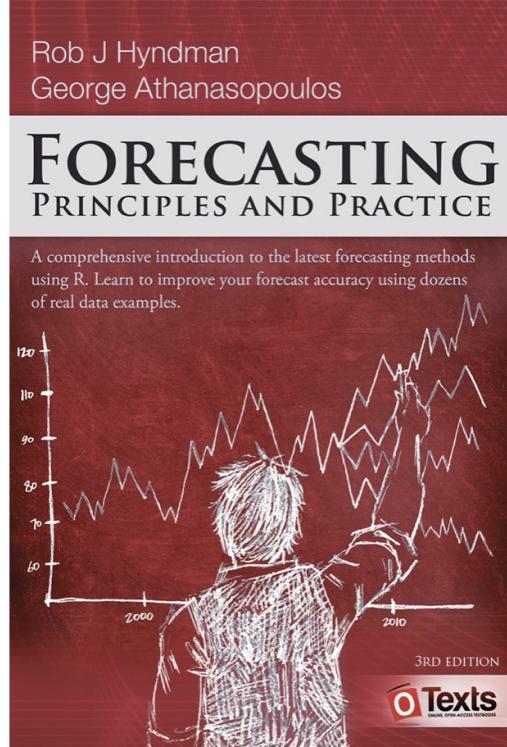
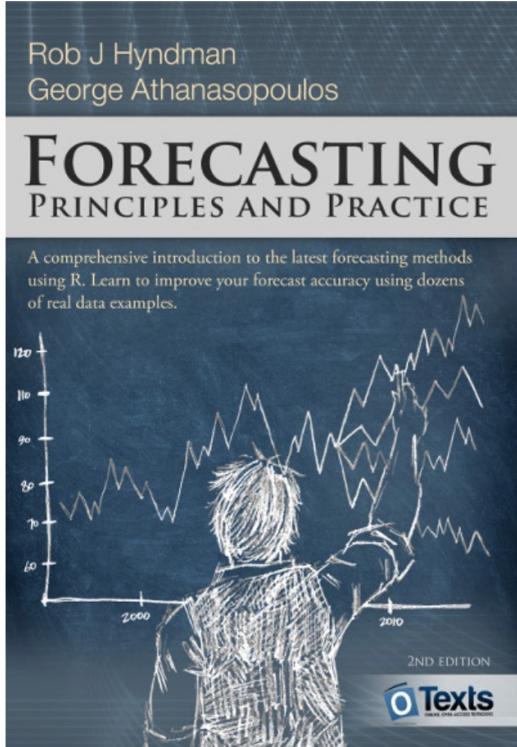


<https://allisonhorst.com/time-series-acf>

<https://domino.ai/blog/time-series-with-r>

TIME SERIES

Advanced methods and theory



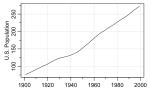
<https://otexts.com>

TIME SERIES

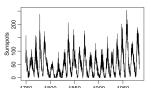
Time Series Cheat Sheet

Plot Time Series

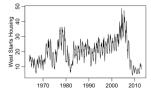
1. tsplot(x=time, y=data)



2. plot(ts(data, start=start_time, frequency=gap))



3. ts.plot(ts(data, start=start_time, frequency=gap))



Simulation

Autoregression of Order p

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + W_t$$

Moving Average of Order q

$$X_t = Z_1 + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} + \dots + \theta_q Z_{t-q}$$

ARMA (p, q)

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} + \dots + \theta_q Z_{t-q}$$

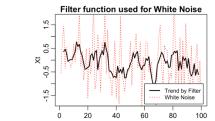
Simulation of ARMA (p, q)

```
arma.sim(model=list(ar=c(phi1, ..., phi_p),
                    ma=c(theta1, ..., theta_q)), n=n)
```

Filters

Linear Filter: filter()

```
filter(data, filter=filter_coefficients, sides=2,
       method="convolution", circular=F)
```



Differencing Filter: diff()

```
diff(data, lag=4, differences=1)
```



Parameter Estimation

Fit an ARMA time series model to the data

ar(): To estimate parameters of an AR model

```
ar(x=data, aic=T, order.max = NULL,
    c("yule-walker", "burg", "ols", "mle", "yw"))
```

```
[1] <environment: 0x0000000000404000>
ar1 = 0.997, aic = 7800, order.max = NA, method = c("yule-walker",
                                                    "burg", "ols", "mle", "yw")
```

Coefficients:

-0.396 -0.198 0.873 0.395 -0.187 0.319 -0.058 -0.301 -0.129

Order selected 9 sigma^2 estimated as 1.52

number_to_predict, fit\$pred)

lines(length(data)+1:length(data)+

number_to_predict, fit\$pred)

SampleData

0 10000 20000 30000

Time

(Complete) Auto-correlation function: acf()

```
acf(data, type='correlation', na.action=na.pass)
```

aci1:

aci1\$acf = sort1, order = c(2, 0, 5), method = c("ML"))

Coefficients:

ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8 ar9

s.e. -0.9918 0.8754 0.4916 0.3847 0.4794 0.4213 0.4012 0.3895 0.3862

signif2 estimated as 1.193: log likelihood = -230.78, aic = 461.55

number_to_predict)

h=number_to_predict)

Predicted value and Conf Interval of ARIMA

40000

30000

20000

10000

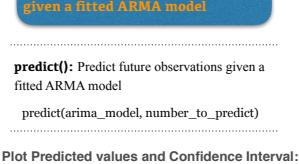
0

Time

0 50 100 150

Lag

RStudio® is a trademark of RStudio, Inc. • CC BY SA Yunjun Xia, Shuyu Huang • yx2569@columbia.edu, sh3967@columbia.edu • Updated: 2019-10



Plot Predicted values and Confidence Interval:

fit<-predict(arima_model, number_to_predict)

ts.plot(data,

xlim=c(1, length(data)+number_to_predict),

ylim=c(0, max(fit\$pred+1.96*fit\$se)))

lines(length(data)+1:length(data)+

number_to_predict, fit\$pred)

SampleData

0 10000 20000 30000

Time

0 50 100 150

Lag



RPubs

SerieTCode
Vásquez Guerra Carlos Fernando
08/2024

Datos

Es un lenguaje de programación en el que uno de sus principales el orientado a objetos y es común encontrar objetos para distintos tipos de datos, en este caso tenemos datos diferentes a series temporales. Es la forma más sencilla de manejar datos en R, siendo una librería que nos permite manejar datos de forma más eficiente y rápida. Para ver otros tipos de datos e igualmente más.

Anexo de continuar con lo relevante mencionar que para la creación de este post se tomó como referencia el libro virtual Forecasting: Principles and Practice 2nd edn. Actualmente existe una tercera edición en la cual se han combinado las partes sobre el uso de paquetes de R y se dejó al lector explorar todo contenido. El propio autor del anterior creó el paquete forecast de R y por esta razón se utilizó la segunda edición. En diversas ocasiones se colocaron hiperínculos con contenido adicional en la lectura por lo que se recomienda explorar el contenido referenciado, podrás distinguir estos enlaces cuando un conjunto de palabras tengan un formato similar a este formato.

También, se dejan aquí unos Cheat Sheets que pueden ser de utilidad cada vez que se realice un análisis de series de tiempo en R.

- Time Series Cheat Sheet.
- impulso.
- libertad.

Este último enlace hace referencia al paquete lubridate, un paquete especializado en el manejo de fechas; para ver un poco de esta librería así como el uso y manipulación de fechas con objetos básicos de R véase el post Manejo de fechas con R. Valeo lo sencillo es crear una serie de tiempo con la función base::ts().

```
set.seed(123)
datos <- caisse(nrow=100)
firstSeries <- tsdata(datos, start = 2020, frequency = 1)
```

El manejo de fechas es tan importante como aprender el manejo de factores u otros tipos de datos de uso común en un análisis de datos, por lo que aquí se dejó un pequeño resumen con ejemplos del manejo de estas en las funciones básicas de R además del uso de la librería chron, lubridate, zoo y xts.

Creación de fechas

```
class(as.Date("2020/01/23"))
[1] "Date"
```

Datos

Se tienen diferentes formas de crear fechas:

- as.Date(): Manejar fechas para tiempos.
- as.POSIXct(): Manejar fechas para tiempos.
- as.POSIXlt(): Manejar fechas para tiempos y zona horaria.

```
as.Date("1998-09-15")
[1] "1998-09-15"
```

```
as.Date("1998/09/15")
[1] "1998-09-15"
```

```
as.Date("1998-09-15")
[1] "1998-09-15"
```

• Formato por defecto: AAAA, MM, DD

<https://rpubs.com/CarlosFVG/SeriesTCode2024>



Manejo de fechas en R
Vásquez Guerra Carlos Fernando
12/01/2020

El manejo de fechas es tan importante como aprender el manejo de factores u otros tipos de datos de uso común en un análisis de datos, por lo que aquí se dejó un pequeño resumen con ejemplos del manejo de estas en las funciones básicas de R además del uso de la librería chron, lubridate, zoo y xts.

Creación de fechas

```
class(as.Date("2020/01/23"))
[1] "Date"
```

Datos

Se tienen diferentes formas de crear fechas:

- as.Date(): Manejar fechas para tiempos.
- as.POSIXct(): Manejar fechas para tiempos.
- as.POSIXlt(): Manejar fechas para tiempos y zona horaria.

```
as.Date("1998-09-15")
[1] "1998-09-15"
```

```
as.Date("1998/09/15")
[1] "1998-09-15"
```

```
as.Date("1998-09-15")
[1] "1998-09-15"
```

• Formato por defecto: AAAA, MM, DD

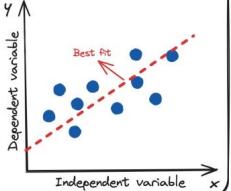
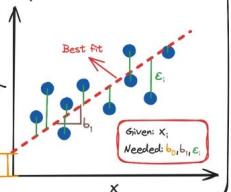
<https://rpubs.com/CarlosFVG/DatesWithRCFVG>

LINEAR REGRESSION

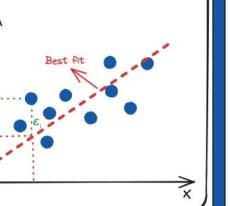
Simple Linear Regression - Clearly explained
hub.tinztwins.de

- 1. Visual Explanation**
 - Linear Regression is a simple statistical regression method, ideal for beginners.
 - You can perform Linear Regression with multiple variables or just one. In this sheet, we use a single variable known as Simple Linear Regression.

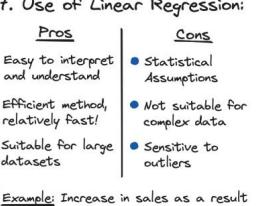
Goal: Find a line "Best fit" that represents the trend in the data.


- 2. Assumptions**
 - Linearity: Linear Relationship between independent and dependent variables
 - Independence: The residuals should be independent and uncorrelated.
 - Homoscedasticity: The variance of the residuals must be constant.
 - Normal distribution: The errors of the prediction follow a normal distribution with a mean close to zero.
- 3. Mathematical Explanation**
 - To calculate the best-fit line for the data, we use the following formula:
$$Y_i = b_0 + b_1 X_i + \epsilon_i \quad i = 1 \dots n$$
 - Y_i : Dependent Variable
 b_0 : Intercept
 b_1 : Slope
 X_i : Independent Variable
 ϵ_i : Random Errors (Residuals)
- 4. How can we calculate the Residuals?**
 - The best-fit line has the least error.
 - The Residuals are the difference between the observed values of the dependent variable and the predicted values.
$$\epsilon_i = y_i - \hat{y}_i$$

Goal: We have to minimize this error to get the best-fit line.


- 5. How can we calculate the Intercept and the Slope?**
 - We use the least squares method to calculate the intercept and the slope. This method is known as OLS regression (Ordinary Least Squares).
 - We get the best fit line when the sum of the distance squares is minimal.
$$\sum_{i=1}^n (y_i - (b_0 + b_1 X_i))^2$$


The least squares estimates are given by the following formulas:

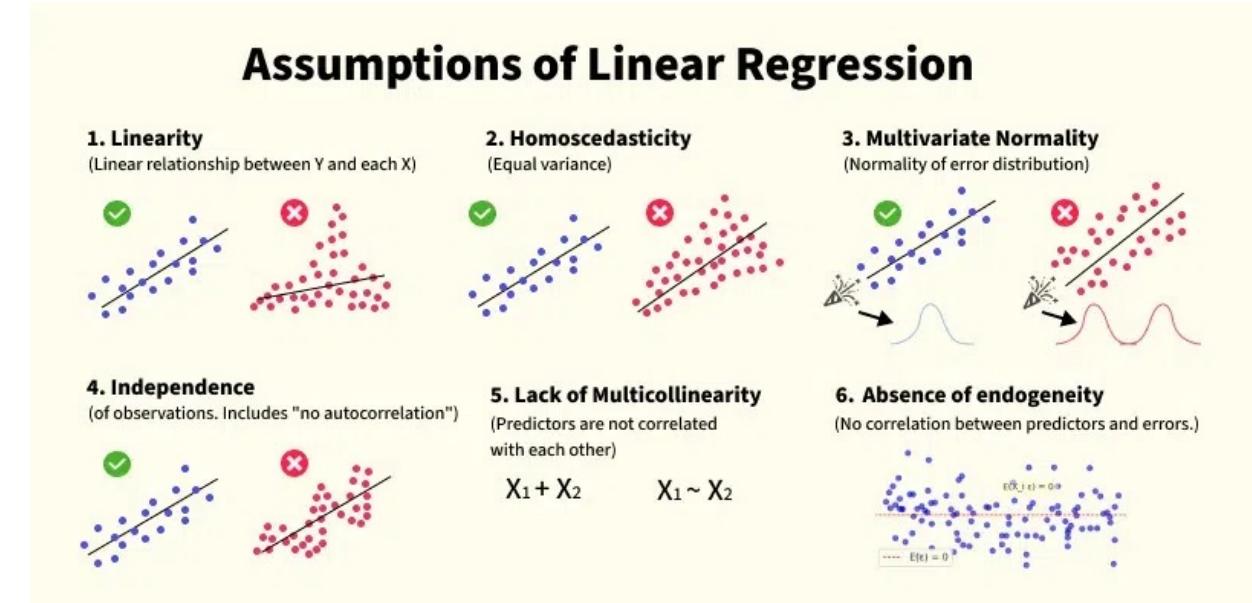
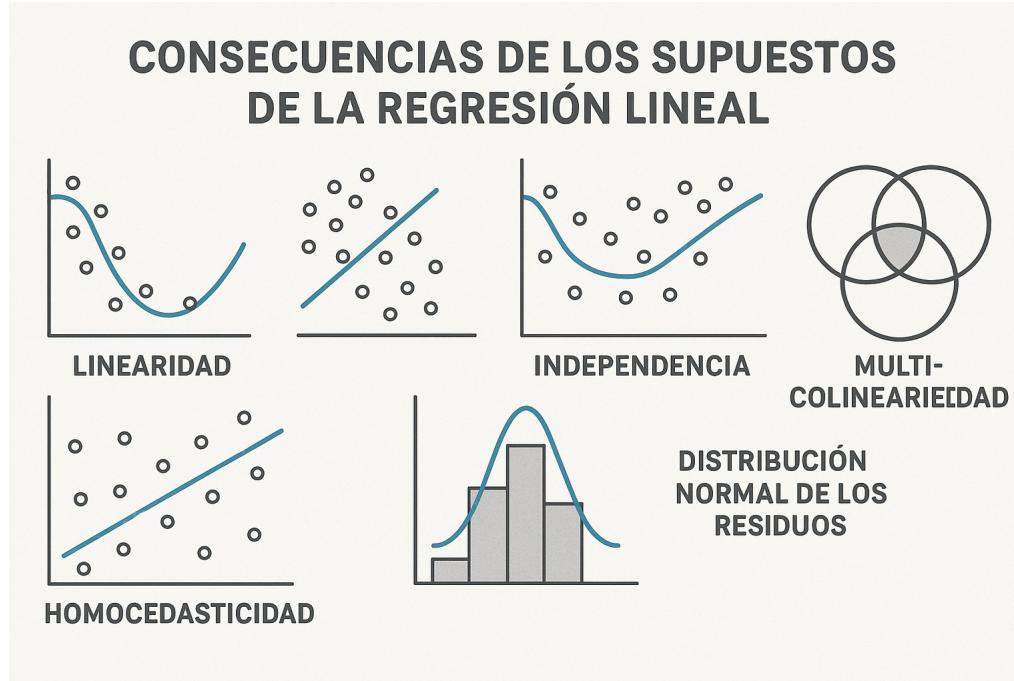
$$b_0 = \bar{y} - b_1 \bar{x}$$
$$b_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$
- 6. Evaluation**
 - Formula R-squared (R^2) or Coefficient of Determination:
$$1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$
Values between 0 and 1!
 - The coefficient of determination describes the percentage of variance of the dependent variable (y) that can be explained by the independent variable (x).
- 7. Use of Linear Regression:**

Pros	Cons
<ul style="list-style-type: none">Easy to interpret and understandEfficient method, relatively fast!Suitable for large datasetsExample: Increase in sales as a result of marketing activities	<ul style="list-style-type: none">Statistical AssumptionsNot suitable for complex dataSensitive to outliers

<https://christophm.github.io/interpretable-ml-book/limo.html>

https://faculty.nps.edu/rbassett/_book/introduction-to-linear-regression.html

LINEAR REGRESSION: WARNINGS



<https://www.geeksforgeeks.org/assumptions-of-linear-regression/>

LINEAR REGRESSION: WARNINGS

Impacto del Incumplimiento de Supuestos en Regresión lineal

Supuesto	Consecuencias del incumplimiento	Gravedad	¿Afecta predicciones?	¿Afecta inferencias?	Acciones de contingencia	Pruebas diagnósticas / gráficas sugeridas
Linealidad	El modelo no captura la relación real. Residuos grandes, baja precisión.	Alta	Sí	Sí	- Transformar variables - Probar modelos no lineales (e.g., regresión polinómica, árboles)	- Gráfico de residuos vs. valores ajustados - Componentes parciales (partial residual plots)
Independencia de los errores	Subestima errores estándar, sesga intervalos de confianza y p-valores (e.g., en series de tiempo).	Alta	Algo	Mucho	- Modelos para datos dependientes (e.g., ARIMA) - Revisar autocorrelación (Durbin-Watson)	- Prueba de Durbin-Watson - ACF/PACF de residuos
No hay observaciones influyentes/extremas	Un outlier puede dominar los resultados.	Alta	Mucho	Mucho	- Diagnóstico (Cook's D, leverage) - Modelos robustos (RLM) - Tratar o justificar outliers	- Leverage vs. residuos estudentizados-Distancia de Cook - DFBETAS
Medición precisa de las variables (sin error en X)	Sesgo en coeficientes, atenuación.	Media	Sí	Sí	- Validar mediciones - Usar modelos con error en variables (SEM, variables instrumentales)	- Comparación con medidas repetidas - Estudios de validación externa - Análisis de sensibilidad
Homoscedasticidad (varianza constante)	Menos precisión, pruebas t y F poco confiables.	Media	Algo	Sí	- Usar errores robustos (White) - Transformaciones (log, sqrt) - Regresión ponderada	- Gráfico de residuos vs. valores ajustados - Prueba de Breusch-Pagan o White
Ausencia de multicolinealidad	Inestabilidad en coeficientes, errores estándar inflados.	Media	Poco	Sí	- Eliminar variables correlacionadas- - PCA - Ridge/Lasso	- VIF (Variance Inflation Factor) - Matriz de correlación de predictores
Normalidad de errores (para inferencia)	Afecta p-valores e IC si n es pequeño. Con n grande, el CLT lo compensa.	Baja	Poco	Algo	- Aumentar n - Bootstrap para IC y pruebas - Pruebas no paramétricas	- Histograma de residuos - QQ plot - Prueba de Shapiro-Wilk o Kolmogorov-Smirnov

Fuentes y referencias

- Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). Introduction to Linear Regression Analysis. Wiley.
- Kutner, M. H., Nachtsheim, C. J., Neter, J., & Li, W. (2005). Applied Linear Statistical Models. McGraw-Hill.
- Fox, J. (2016). Applied Regression Analysis and Generalized Linear Models. Sage Publications.
- Wooldridge, J. M. (2013). Introductory Econometrics: A Modern Approach. Cengage Learning.

LINEAR REGRESSION: GOOD RESOURCES



Josep Ferrer 
@rfeers

Data Scientist & Tech Writer
@KDnuggets @DataCamp @Medium

| Outstand using data |

Join 8k data professionals on [databites.tech](#) 

626 Siguiendo 27 mil Seguidores

ML BASICS - SIMPLE LINEAR REGRESSIONS

#1 SIMPLE LINEAR REGRESSION

Linear regression is the simplest statistical regression method used for predictive analysis.

- The most common is the SIMPLE LINEAR REGRESSION

1 independent variable + 1 dependent variable

The main goal?

Find a linear relationship between the independent variable (predictor) and the dependent (output)

#2 HOW TO COMPUTE IT?

To compute the best-fit line linear regression, we use the line function.

$$Y_i = Ax + B$$

Y_i = Dependent Variable
 B = Intercept
 A = slope
 x_i = Independent variable

#3 HOW TO DEFINE THE BEST FIT?

We define the best fit line as the line that presents the least error.

The error between predicted values and the actual values should be minimum.

#4 HOW TO OBTAIN IT MATHEMATICALLY?

We use a cost function that helps us work out the optimal values for A and B .

MEAN SQUARED ERROR (MSE)

We use the average of the squared error that occurs between predicted and observed values.

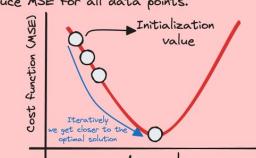
To find the optimal solution

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - (Ax_i + B))^2$$

GRADIENT DESCENT

Gradient descent is one of the optimization algorithms that optimizes the cost function.

To obtain the optimal solution we need to reduce MSE for all data points.



#5 EVALUATION

The most used metrics are,

- Coefficient of Determination or R-Squared (R²)
- Root Mean Squared Error (RMSE)

#6 ASSUMPTIONS TO APPLY IT

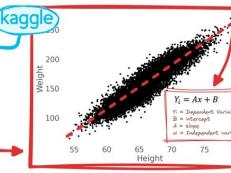
1. Linearity of residuals:
There needs to be linear dependency between the dependent and the independent variables.
2. Independence of residuals:
The error terms should not be dependent on one another
3. Normal distribution of residuals:
The mean of residuals should follow a normal distribution with a mean close to zero.
4. The equal variance of residuals:
The error terms must have constant variance.

ML BASICS - SIMPLE LINEAR REGRESSIONS EXAMPLE 1

#1 GETTING THE DATA

Today we are dealing with some **real-world data**. And the turn is for... **height and weight!**

One of the classic examples of linear dependency!



APPROACH 1 - GRADIENT DESCENT

We use the mean Square error (MSE) that occurs between predicted and observed values.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - (Ax_i + B))^2$$

This translates into defining two main functions:

```
def compute_mse(x, y, pred):  
    N = len(y)  
    MSE = np.mean((y - pred)** 2) / N  
    return MSE
```

The Function to compute MSE

```
def gradient_descent(x, y, A, B, learning_rate):  
    y_pred = A * x + B  
    J = -2 * np.sum((y - y_pred) ** 2) / N  
    A -= learning_rate * dA / N  
    B -= learning_rate * dB / N  
    return A, B
```

The Function to update A and B

GRADIENT DESCENT

gradient descent is one of the optimization algorithms that optimizes the cost function. To obtain the optimal solution we need to reduce MSE for all data points.

Cost function visualized

Initialization value

Decreasing there is a global minimum

Allocating there is a local minimum

A - value

Distribution of Height

Distribution of Weight

Both present a normal distribution.

APPROACH 2 - OLS (Ordinary Least Squares)

The goal of OLS is to find the values of A and B that minimize the sum of the squared residuals (S).

$$S = \sum_{i=1}^N (y_i - (Ax_i + B))^2$$

To minimize S , we can easily take its partial derivatives and set them to zero.

$$\frac{\partial S}{\partial A} = 0 \quad \frac{\partial S}{\partial B} = 0$$

Solving these two equations we obtain a closed mathematical solution.

$$A = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^N (x_i - \bar{x})^2}$$
$$B = \bar{y} - A\bar{x}$$

This translates into defining some lines of code to find this mathematical closed solution:

```
x_mean = np.mean(x)  
y_mean = np.mean(y)  
  
for i in range(N):  
    numerator += (x[i] - x_mean) * (y[i] - y_mean)  
    denominator += (x[i] - x_mean)** 2  
  
A = numerator / denominator  
B = y_mean - (A * x_mean)
```

APPROACH 3 - SCI-KIT LEARN

Sci-kit learn is a versatile Python library offering a wide range of machine learning tools, including algorithms for:

- Classification
- Regression
- Clustering
- And way more...

Import the library

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

Create a Linear Regression object

```
lr = LinearRegression()
```

Train it with our data

```
lr.fit([[Height]], [[Weight]])
```

Get the predicted output.

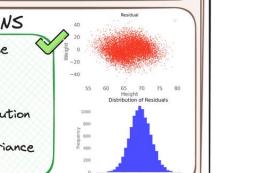
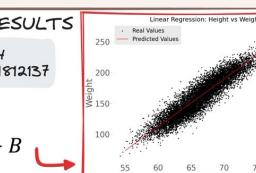
Linear Regression: Height vs Weight

Real Values Predicted Values

$Y_i = Ax + B$

ASSUMPTIONS

1. Linearity of the variables
2. Independence of residuals
3. Normal distribution of residuals
4. The equal variance of residuals.



LINEAR REGRESSION: GOOD RESOURCES

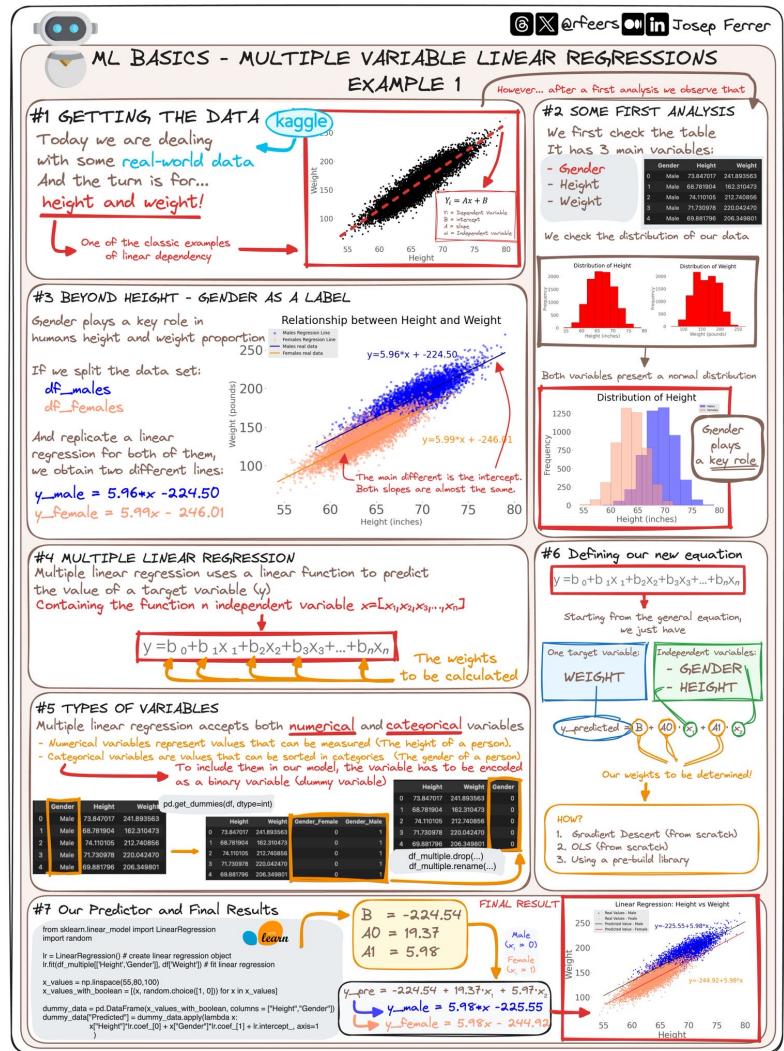


@allison_horst



<https://allisonhorst.com/linear-regression-dragons>

+ LINEAR REGRESSION

A detailed guide on multiple variable linear regression. It starts with a scatter plot of weight vs height, showing a linear relationship. It then splits the data by gender (Males vs Females) and shows two separate regression lines. The guide also covers multiple linear regression with multiple independent variables. It includes code snippets for data manipulation and regression fitting.

```

#1 GETTING THE DATA
Today we are dealing with some real-world data And the turn is for... height and weight!
One of the classic examples of linear dependency

#2 SOME FIRST ANALYSIS
We first check the table. It has 3 main variables:
- Gender
- Height
- Weight

#3 BEYOND HEIGHT - GENDER AS A LABEL
Gender plays a key role in humans height and weight proportion
If we split the data set:
df_males
df_females
And replicate a linear regression for both of them, we obtain two different lines:
y_male = 5.96x + 224.50
y_female = 5.99x + 246.01

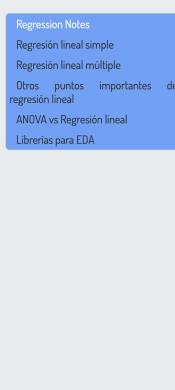
#4 MULTIPLE LINEAR REGRESSION
Multiple linear regression uses a linear function to predict the value of a target variable (y) containing the function n independent variable x=[x1, x2, x3, ..., xn]
y = b0 + b1x1 + b2x2 + b3x3 + ... + bnxn The weights to be calculated

#5 TYPES OF VARIABLES
Multiple linear regression accepts both numerical and categorical variables
- Numerical variables represent values that can be measured (The height of a person)
- Categorical variables are values that can be sorted in categories (The gender of a person)
To include them in our model, the variable has to be encoded as a binary variable (dummy variable)

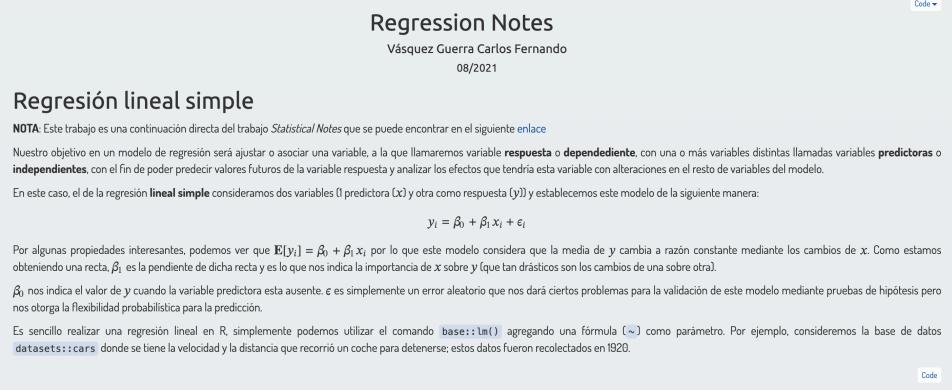
#6 Defining our new equation
y = b0 + b1x1 + b2x2 + b3x3 + ... + bnxn
Starting from the general equation, we just have
One target variable: WEIGHT
Independent variables: - GENDER | HEIGHT
y_predicted = B + A1(x1) + A2(x2)
Our weights to be determined!

#7 Our Predictor and Final Results
from sklearn.linear_model import LinearRegression
import random
lr = LinearRegression()
lr.fit(df[['Height', 'Gender']], df['Weight']) # fit linear regression
x_values = np.linspace(55, 80, 100)
x_values_with_boolean = [x, random.choice([1, 0]) for x in x_values]
dummy_data = pd.DataFrame(x_values, with_boolean, columns = ['Height', 'Gender'])
dummy_data['Predicted'] = dummy_data.apply(lambda x: x['Height'] * x['coefs'][0] + x['Gender'] * x['coefs'][1] + lr.intercept_, axis=1)
y_pre = -224.54 + 19.37*x + 5.97*x
y_male = 5.98*x + 225.55
y_female = 5.98*x - 244.92

```

A slide titled "Regression Notes" with a sub-section "Regressión lineal simple". It contains a table of data and two histograms showing the distribution of height and weight.

	Gender	Height	Weight
0	Male	73.84707	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740656
3	Male	71.739398	220.042470
4	Male	69.881798	206.546907

A slide titled "Regression Notes" with a sub-section "Regressión lineal simple". It contains a table of data and two histograms showing the distribution of height and weight.

	Gender	Height	Weight
0	Male	73.84707	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740656
3	Male	71.739398	220.042470
4	Male	69.881798	206.546907

<https://rpubs.com/CarlosFVG/RegressionNotes>

<https://statisticsbyjim.com/regression/choosing-regression-analysis/>

<https://bookdown.org/roback/bookdown-BeyondMLR/>

https://uc-r.github.io/model_selection

<https://joelcarlson.github.io/2016/05/10/Exploring-Interactions/>

https://ml-cheatsheet.readthedocs.io/en/latest/linear_regression.html

BEYOND LINEAR REGRESSION

Cheat Sheet – Regression Analysis

What is Regression Analysis?
Fitting a function $f(\cdot)$ to datapoints $y_i = f(x_i)$ under some error function. Based on the estimated function and error, we have the following types of regression

- Linear Regression:**
Fits a **line** minimizing the sum of mean-squared error for each datapoint.
- Polynomial Regression:**
Fits a **polynomial** of order k ($k+1$ unknowns) minimizing the sum of mean-squared error for each datapoint.
- Bayesian Regression:**
For each datapoint, fits a **gaussian distribution** by minimizing the mean-squared error. As the number of data points x_i increases, it converges to point $\mathcal{N}(\mu, \sigma^2) \rightarrow$ Gaussian with mean μ and variance σ^2 estimates i.e. $n \rightarrow \infty, \sigma^2 \rightarrow 0$
- Ridge Regression:**
Can fit either a **line**, or **polynomial** minimizing the sum of mean-squared error for each datapoint and the weighted L2 norm of the function parameters beta.
- LASSO Regression:**
Can fit either a **line**, or **polynomial** minimizing the the sum of mean-squared error for each datapoint and the weighted L1 norm of the function parameters beta.
- Logistic Regression:**
Can fit either a **line**, or **polynomial** with **sigmoid activation** minimizing the binary cross-entropy loss for each datapoint. The labels y are binary class labels.

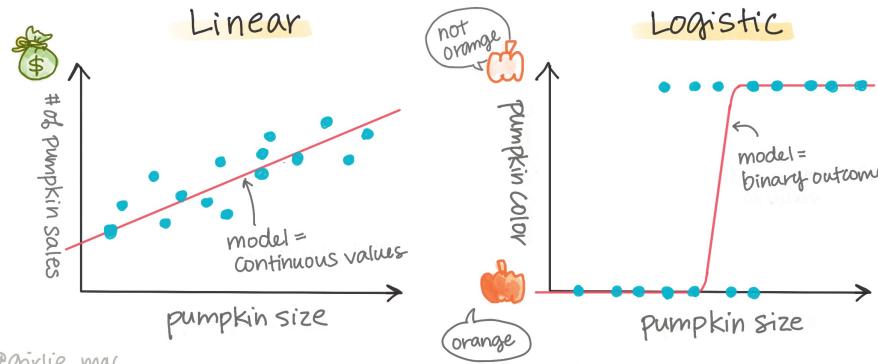
Visual Representation:

Summary:

	What does it fit?	Estimated function	Error Function
Linear	A line in n dimensions	$f_{\beta}^{linear}(x_i) = \beta_0 + \beta_1 x_i$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2$
Polynomial	A polynomial of order k	$f_{\beta}^{poly}(x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2$
Bayesian Linear	Gaussian distribution for each point	$\mathcal{N}(f_{\beta}(x_i), \sigma^2)$	$\sum_{i=0}^n \ y_i - \mathcal{N}(f_{\beta}(x_i), \sigma^2)\ ^2$
Ridge	Linear/polynomial	$f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^k \beta_j^2$
LASSO	Linear/polynomial	$f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^k \beta_j $
Logistic	Linear/polynomial with sigmoid	$\sigma(f_{\beta}(x_i))$	$\min_{\beta} \sum_i -y_i \log(\sigma(f_{\beta}(x_i))) - (1-y_i) \log(1-\sigma(f_{\beta}(x_i)))$

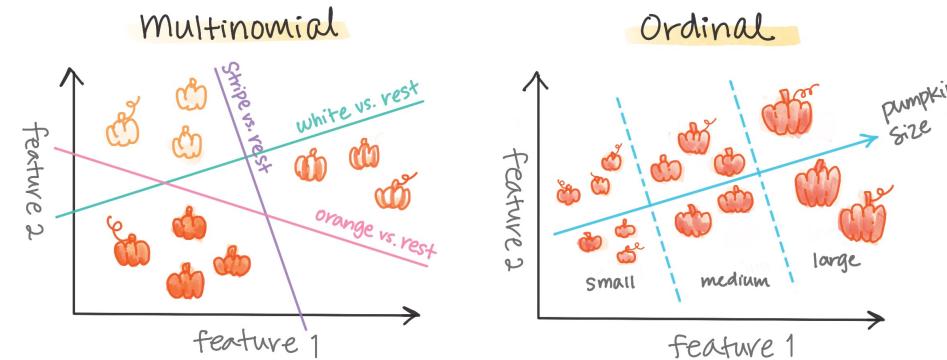
Source: <https://www.cheatsheets.aqeel-anwar.com> Tutorial: [Click here](#)

LINEAR vs. LOGISTIC REGRESSION

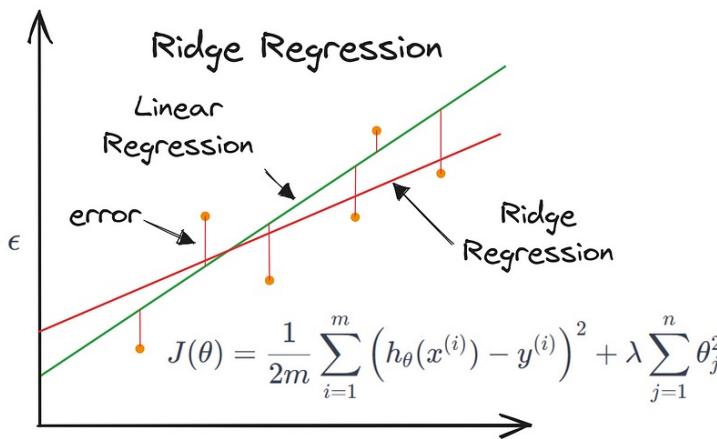
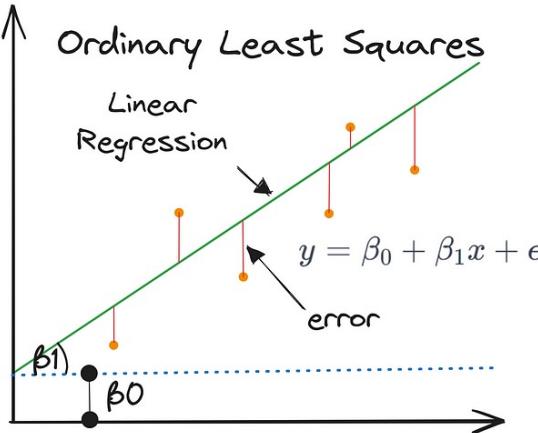
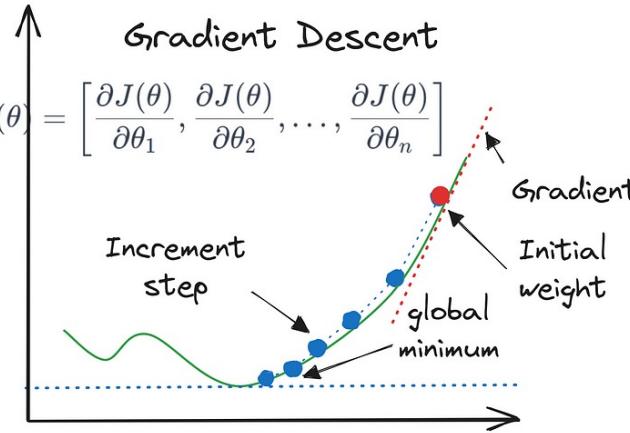
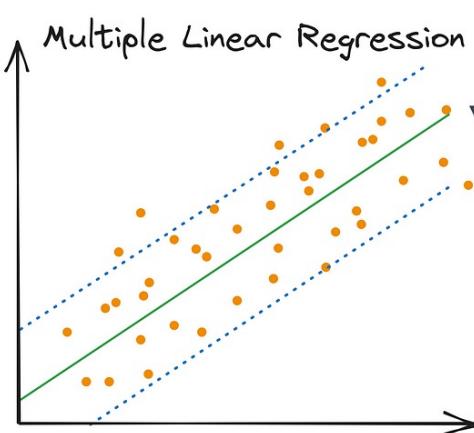


https://github.com/microsoft/ML-For-Beginners/blob/main/2-Regression/4-Logistic/solution/R/lesson_4-R.ipynb

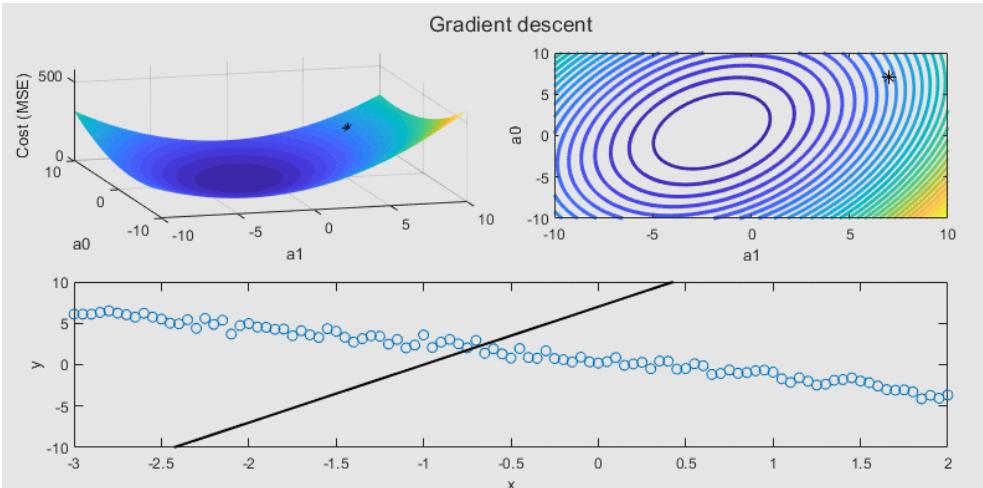
MULTINOMIAL vs. ORDINAL LOGISTIC REGRESSION



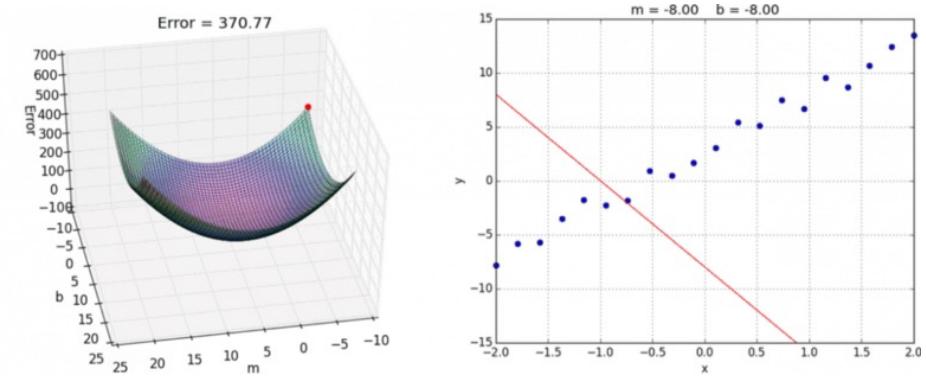
BEYOND LINEAR REGRESSION



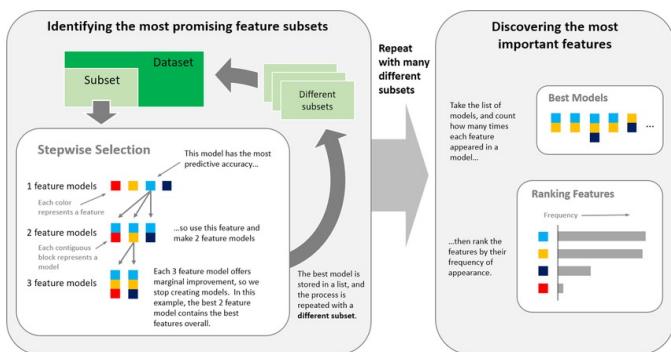
BEYOND LINEAR REGRESSION



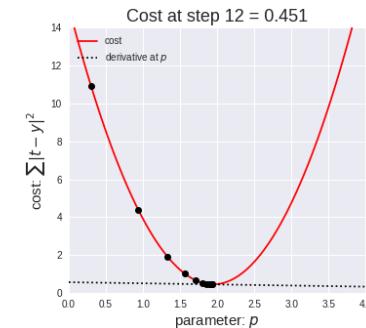
[https://www.linkedin.com/pulse/linear-regressionmostly-asked-questions-manralaitop30-manral/-](https://www.linkedin.com/pulse/linear-regressionmostly-asked-questions-manralaitop30-manral/)



<https://sar-gupta.github.io/posts/2017/10/20/gradient-descent>



https://www.researchgate.net/figure/Feature-selection-process-for-linear-regression-which-was-performed-separately-for-each_fig1_342116063



<https://medium.com/analytics-vidhya/simple-linear-regression-cost-function-gradient-descent-50c5ed085770>

BEYOND LINEAR REGRESSION

Cheat Sheet – Regression Analysis

What is Regression Analysis?
Fitting a function $f(\cdot)$ to datapoints $y_i = f(x_i)$ under some error function. Based on the estimated function and error, we have the following types of regression

- Linear Regression:**
Fits a **line** minimizing the sum of mean-squared error for each datapoint.
- Polynomial Regression:**
Fits a **polynomial** of order k ($k+1$ unknowns) minimizing the sum of mean-squared error for each datapoint.
- Bayesian Regression:**
For each datapoint, fits a **gaussian distribution** by minimizing the mean-squared error. As the number of data points x_i increases, it converges to point $\mathcal{N}(\mu, \sigma^2) \rightarrow$ Gaussian with mean μ and variance σ^2 estimates i.e. $n \rightarrow \infty, \sigma^2 \rightarrow 0$
- Ridge Regression:**
Can fit either a **line**, or **polynomial** minimizing the sum of mean-squared error for each datapoint and the weighted L2 norm of the function parameters beta.
- LASSO Regression:**
Can fit either a **line**, or **polynomial** minimizing the the sum of mean-squared error for each datapoint and the weighted L1 norm of the function parameters beta.
- Logistic Regression:**
Can fit either a **line**, or **polynomial** with **sigmoid activation** minimizing the binary cross-entropy loss for each datapoint. The labels y are binary class labels.

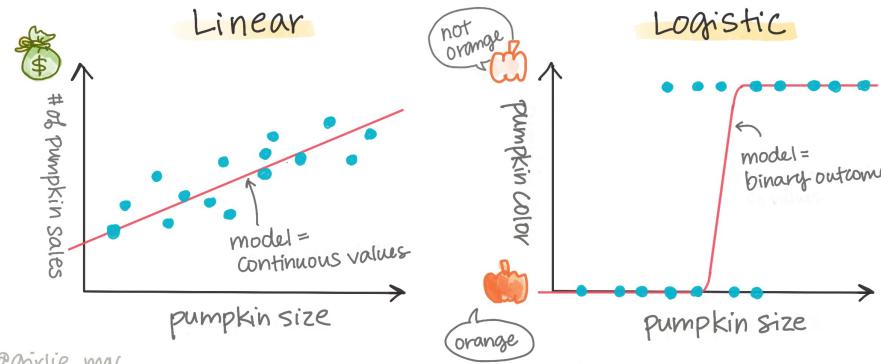
Visual Representation:

Summary:

	What does it fit?	Estimated function	Error Function
Linear	A line in n dimensions	$f_{\beta}^{linear}(x_i) = \beta_0 + \beta_1 x_i$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2$
Polynomial	A polynomial of order k	$f_{\beta}^{poly}(x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2$
Bayesian Linear	Gaussian distribution for each point	$\mathcal{N}(f_{\beta}(x_i), \sigma^2)$	$\sum_{i=0}^n \ y_i - \mathcal{N}(f_{\beta}(x_i), \sigma^2)\ ^2$
Ridge	Linear/polynomial	$f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^k \beta_j^2$
LASSO	Linear/polynomial	$f_{\beta}^{poly}(x_i) \text{ or } f_{\beta}^{linear}(x_i)$	$\sum_{i=0}^n \ y_i - f_{\beta}(x_i)\ ^2 + \sum_{j=0}^k \beta_j $
Logistic	Linear/polynomial with sigmoid	$\sigma(f_{\beta}(x_i))$	$\min_{\beta} \sum_i -y_i \log(\sigma(f_{\beta}(x_i))) - (1-y_i) \log(1-\sigma(f_{\beta}(x_i)))$

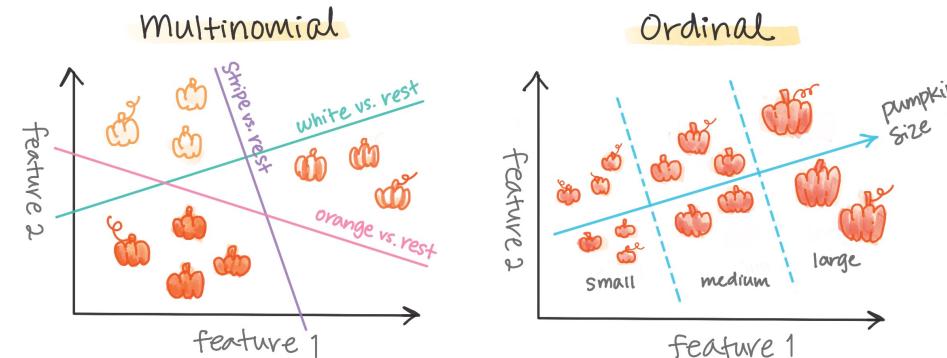
Source: <https://www.cheatsheets.aqeel-anwar.com> Tutorial: [Click here](#)

LINEAR vs. LOGISTIC REGRESSION



https://github.com/microsoft/ML-For-Beginners/blob/main/2-Regression/4-Logistic/solution/R/lesson_4-R.ipynb

MULTINOMIAL vs. ORDINAL LOGISTIC REGRESSION



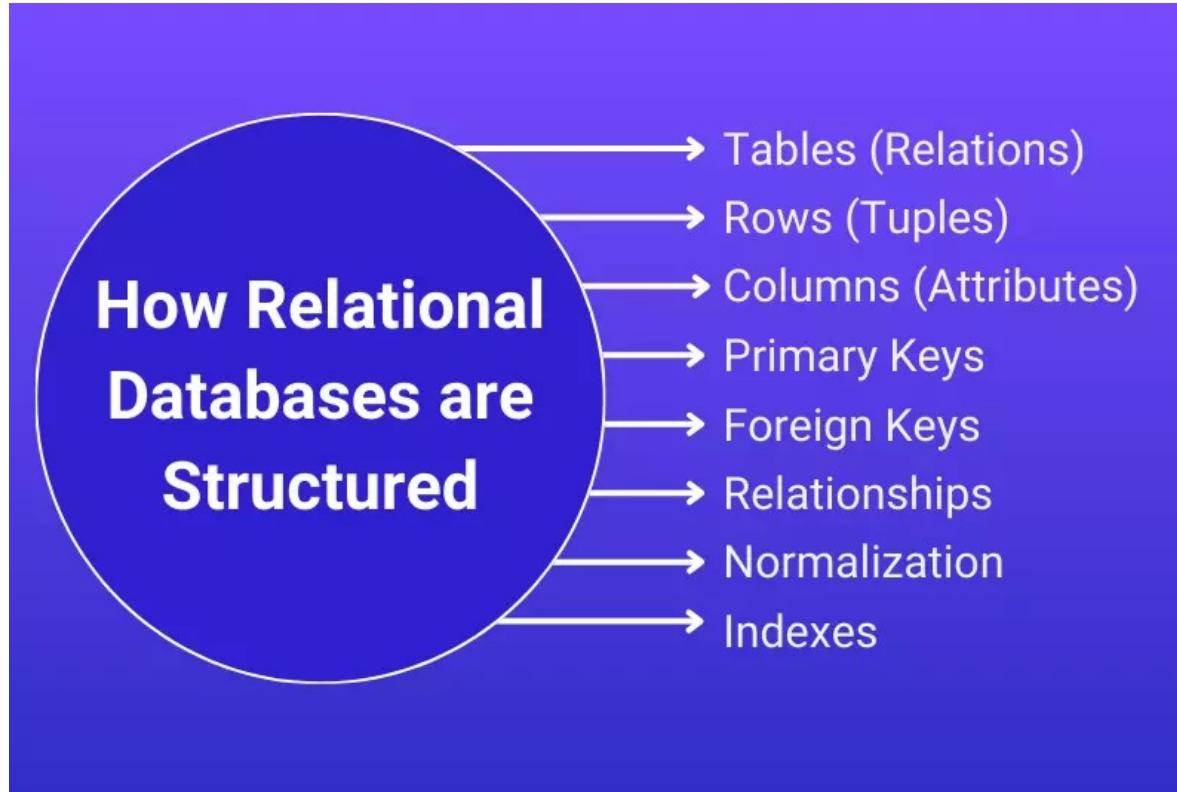
@girlie_mac

MODELADO DE DATOS

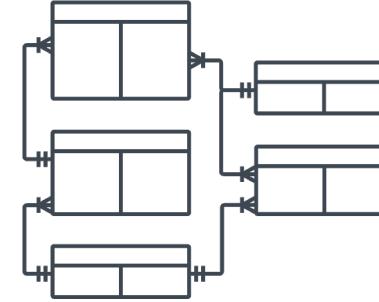
Carlos Fernando Vásquez Guerra



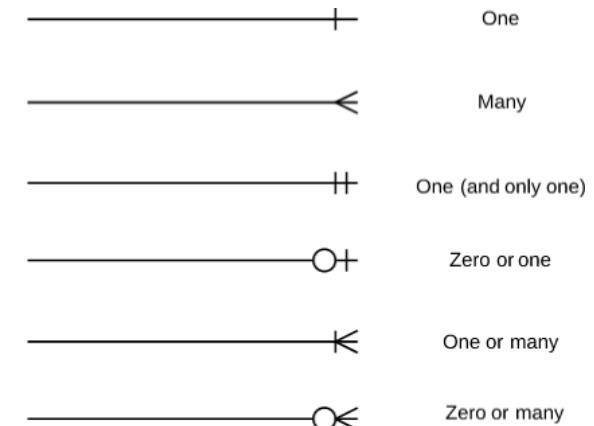
CONCEPTS



<https://www.fynd.academy/blog/relational-data-model-in-rdbms>



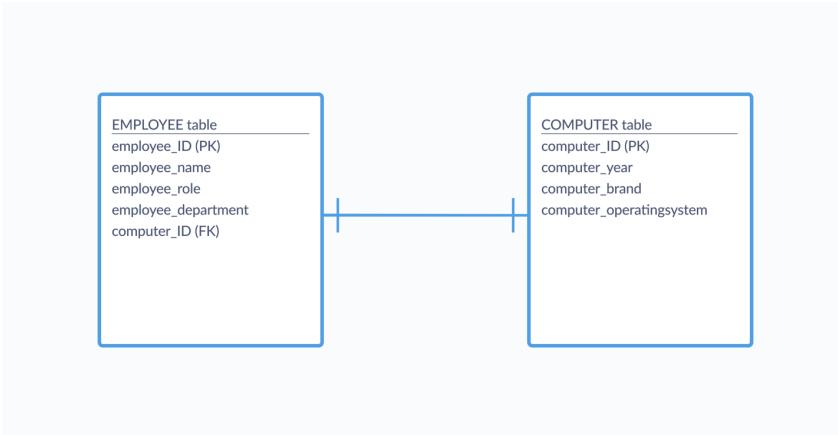
ER diagram notation



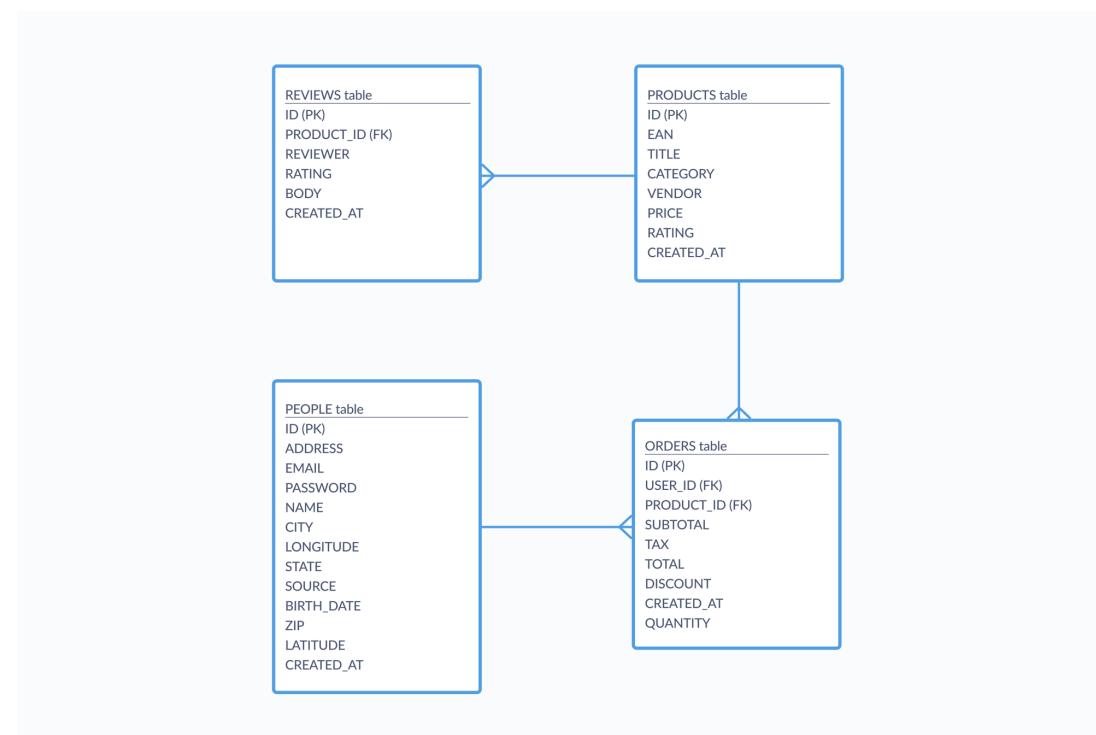
<https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning>

RELATIONS / CARDINALITY

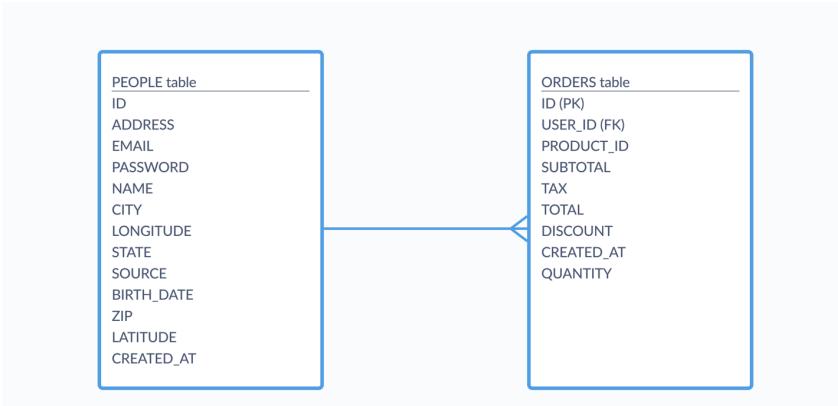
One-to-one relationship



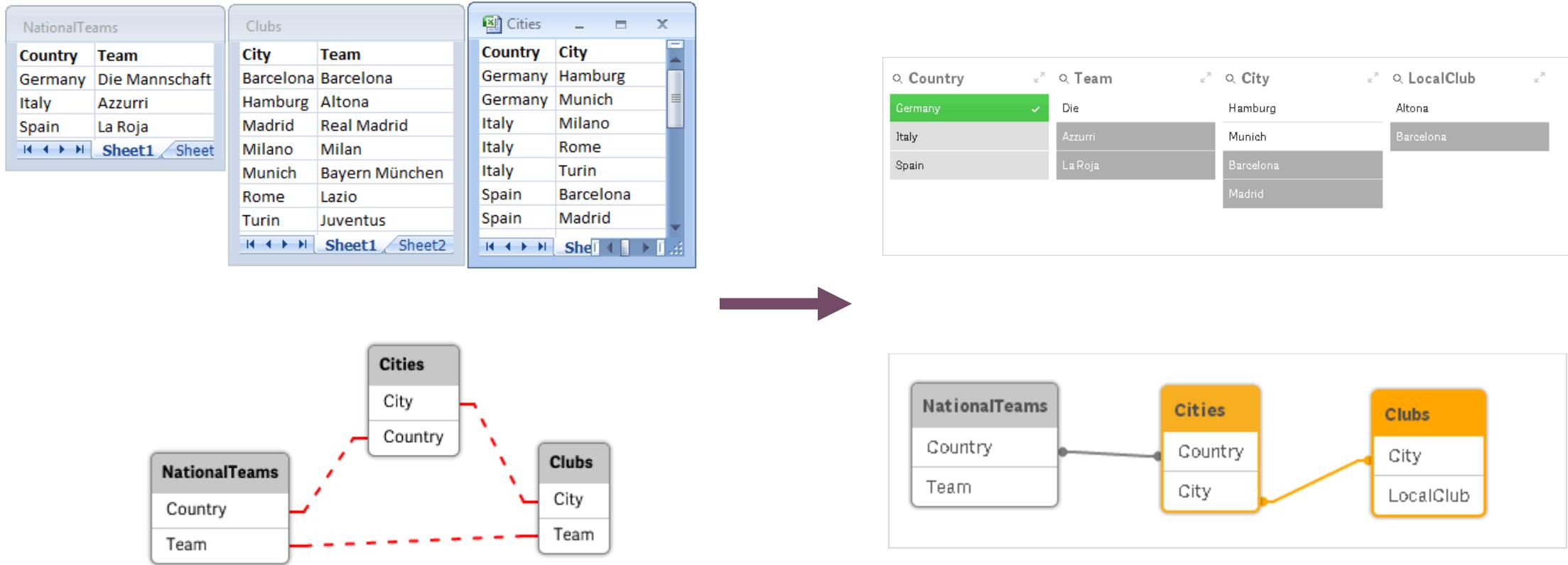
Many-to-many relationship



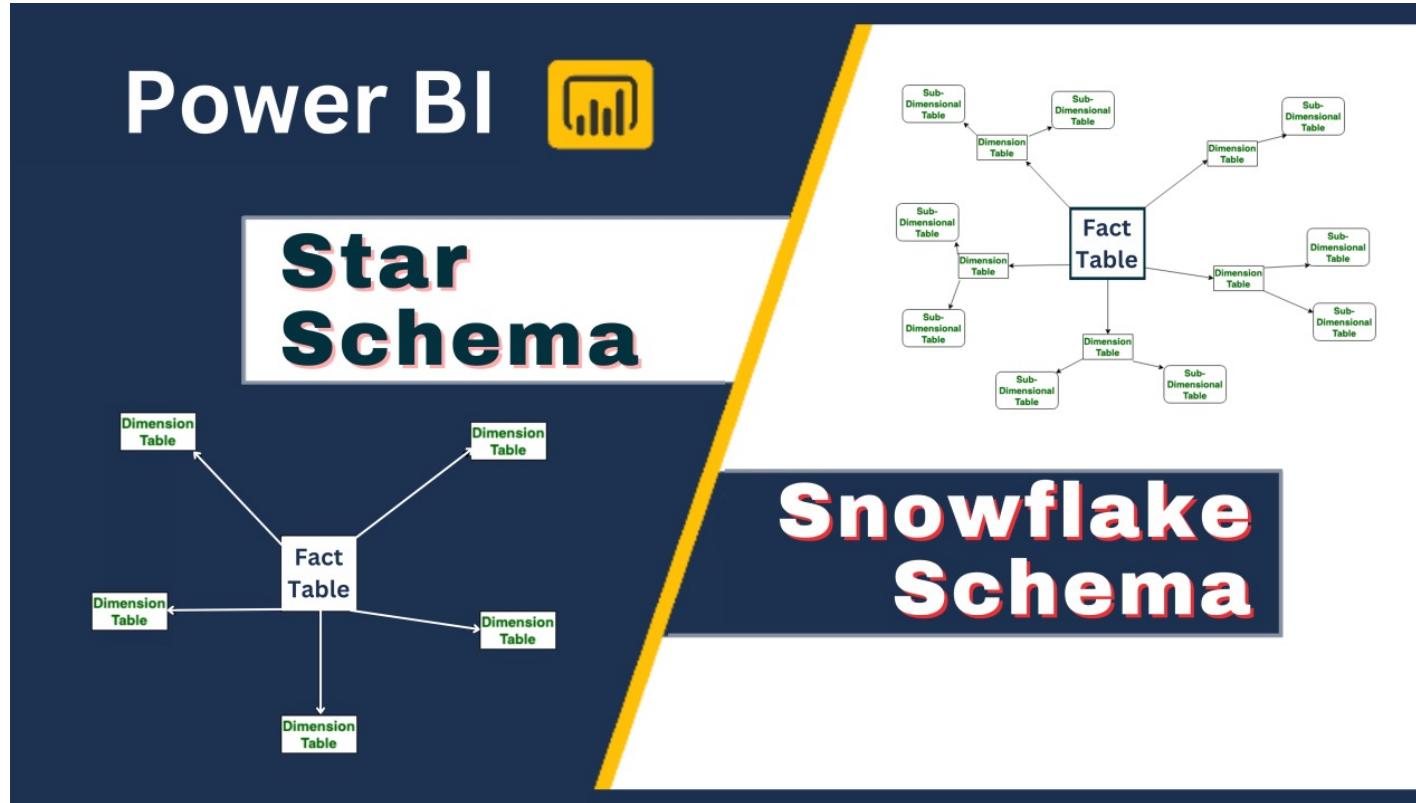
One-to-many relationship



RELATIONS / CARDINALITY



DATA MODELS SCHEMAS



KEYS

Primary Key

The diagram shows two tables: 'Student Details' and 'Student Marks'. A primary key arrow points from the 'ID' column of the 'Student Details' table to the 'ID' column of the 'Student Marks' table. A foreign key arrow points from the 'ID' column of the 'Student Marks' table back to the 'ID' column of the 'Student Details' table.

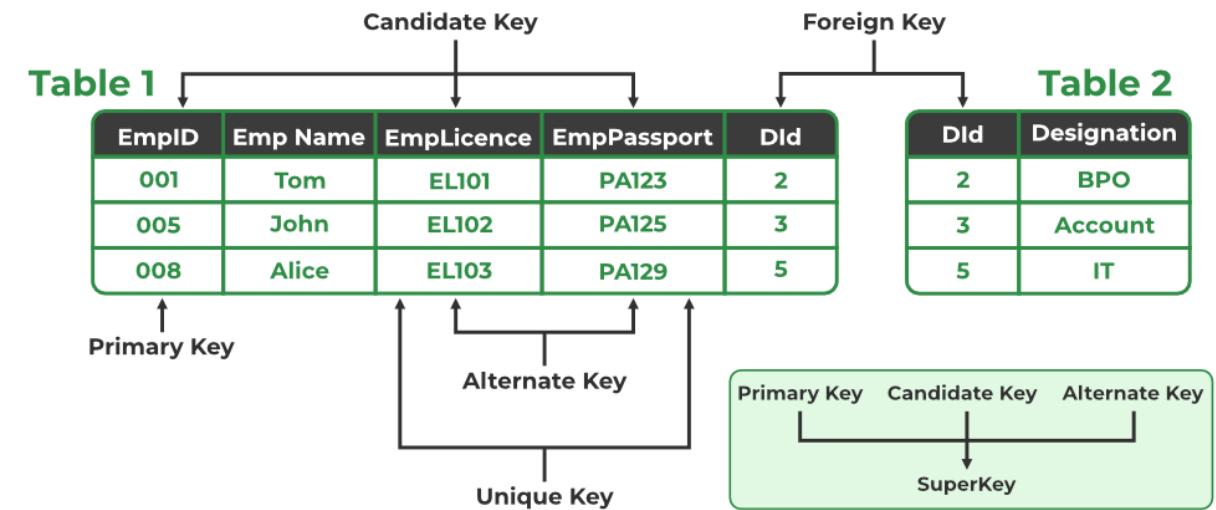
ID	Name	Course
2041	Tom	Java
2204	John	C++
2043	Alice	Python
2032	Bob	Oracle

Student Details

Foreign Key

ID	Marks
2041	65
2204	55
2043	73
2032	62

Student Marks



NORMALIZATION



Normalization in DBMS

Student Table

St.ID	Name	Subject	Grade
1	John Smith	Math, Science	A
2	Jane Doe	English, History	B

First Normal Form (1NF) is the most basic level of normalization. The rules for achieving 1NF are:

- 1 Each table should have a primary key, which uniquely identifies each record in the table.
- 2 Each column in the table should contain only atomic values, which means that a single cell should contain a single value and not a list of values.
- 3 There should be no repeating groups of data.

Student Table in 1NF

St.ID	Name	Subject	Grade
1	John Smith	Math	A
1	John Smith	Science	A
2	Jane Doe	English	B
2	Jane Doe	History	B



Levels of Normalization

The higher level is a subset of lower level.

DatabaseTown.com

TYPES OF NORMALIZATION		
1NF	First Normal Form	All data must be atomic, meaning that each cell in a table should contain only a single value and not a list of values.
2NF	Second Normal Form	In addition to meeting the rules of 1NF, a table must not contain any partial dependencies. A partial dependency exists when a non-primary key column depends on only part of a composite primary key.
3NF	Third Normal Form	In addition to meeting the rules of 2NF, a table must not contain any transitive dependencies. A transitive dependency exists when a non-primary key column depends on another non-primary key column.
BCNF	Boyce-Codd Normal Form	A relation is in BCNF if and only if for every one of its non-trivial functional dependencies $X \rightarrow Y$, X is a superkey.
4NF	Fourth Normal Form	A table is in 4NF if it is in BCNF and it has no multi-valued dependencies.
5NF	Fifth Normal Form	A relation is in 5NF if every non-trivial join dependency in R is implied by the candidate keys of R.

 **DatabaseTown.com**

<https://databasetown.com/types-of-normalization-in-dbms-with-examples/>

<https://www.metabase.com/learn/grow-your-data-skills/data-fundamentals/normalization>