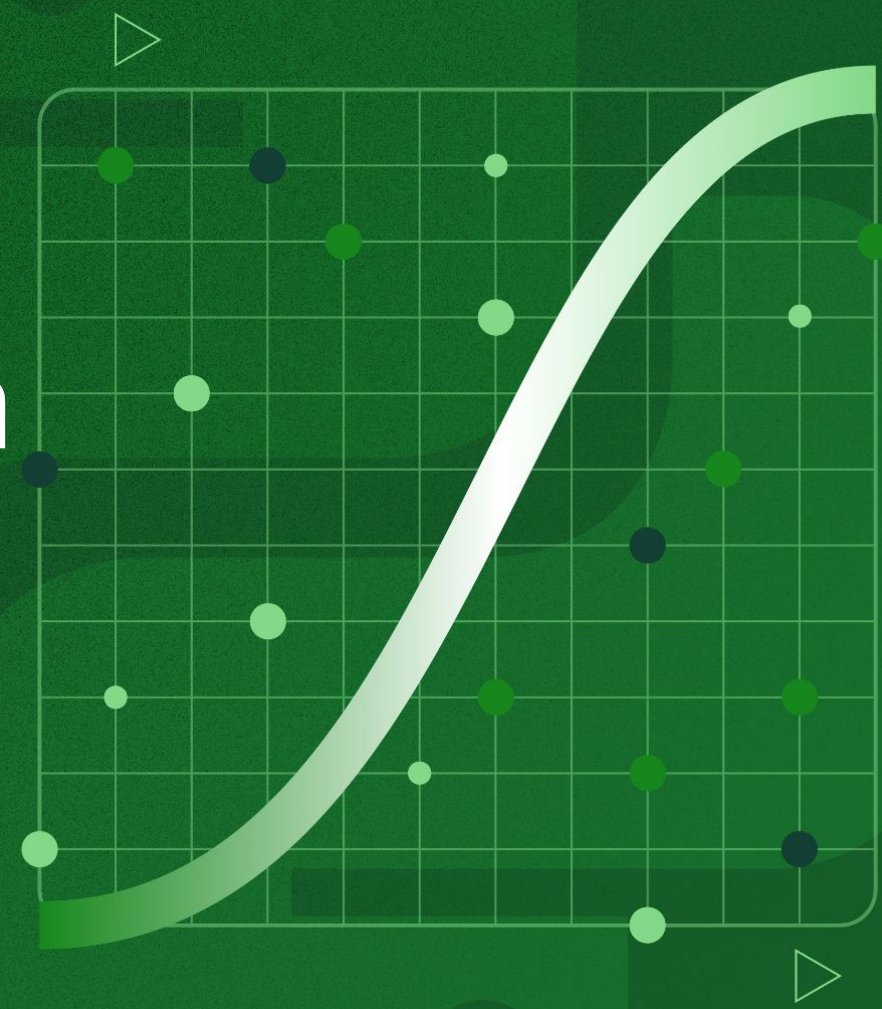
Curso de

Regresión Logística con Python y scikit-learn



Carlos Alarcón

¿Quién es Carlos Alarcón?

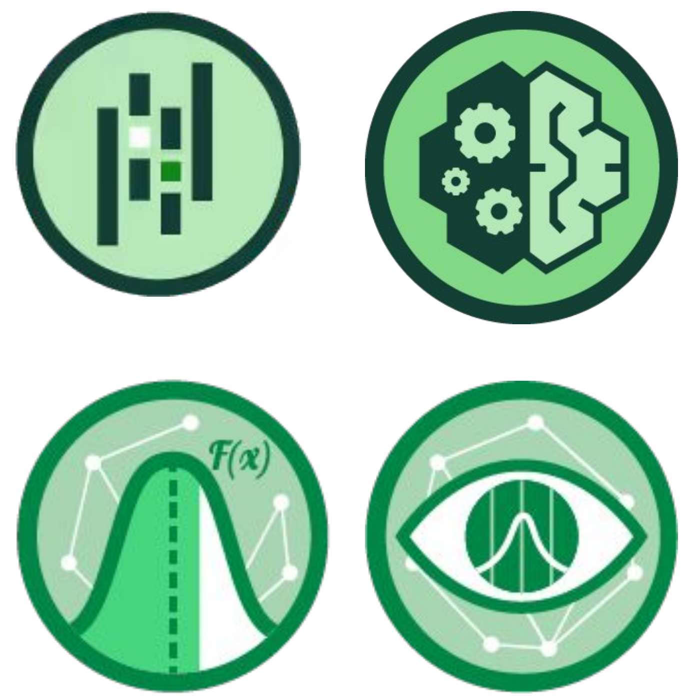
🧑‍💼 Data Architect en Platzi.

💻 Especialista en ciencia de datos, bases de datos y AI.

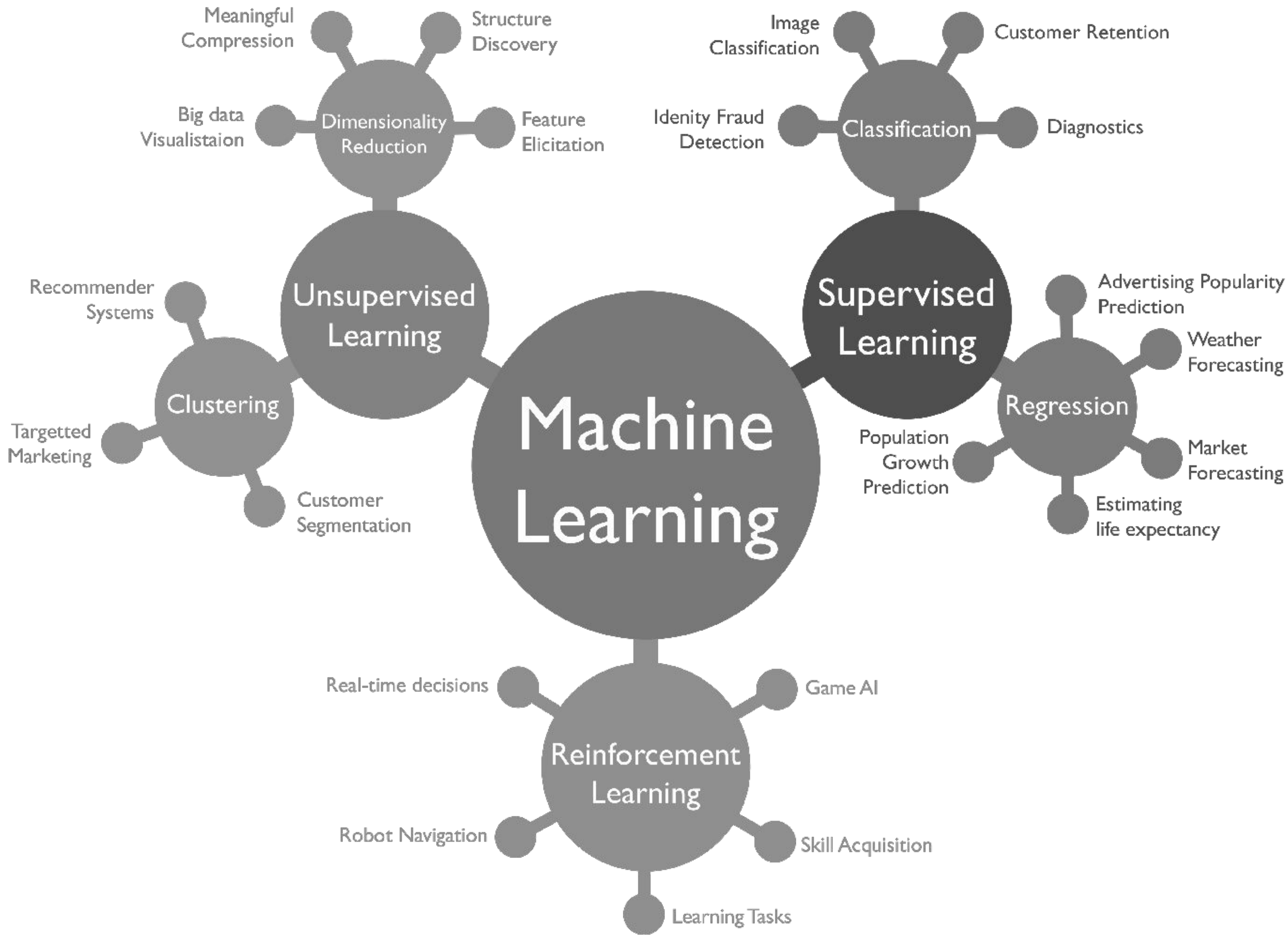
🧠 Profesor de data science y machine learning.

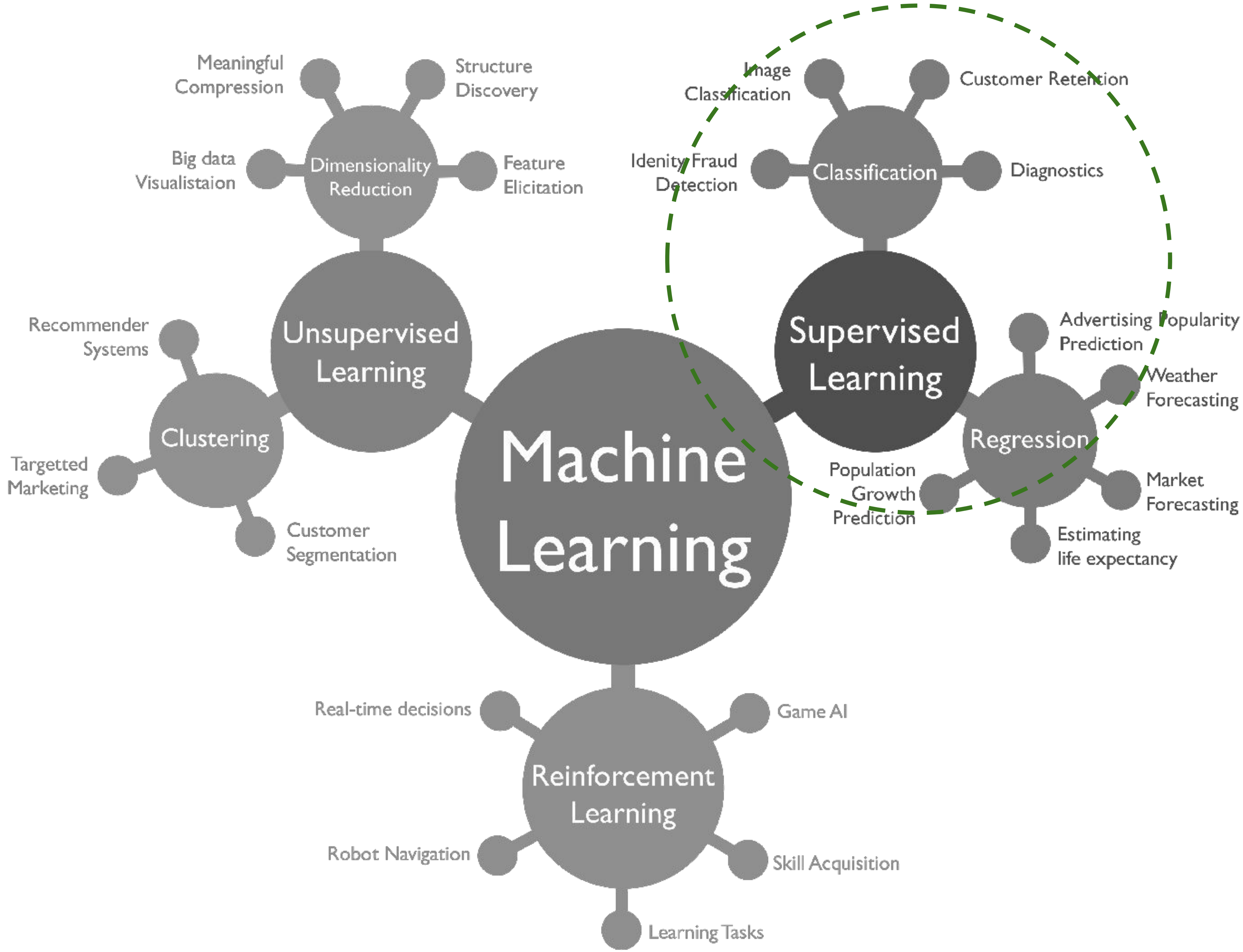
Requisitos previos

- Matemáticas para machine learning.
- Análisis exploratorio de datos con Python y Pandas.
- Visualización de datos con Matplotlib y Seaborn.
- Fundamentos de machine learning y regresión lineal.

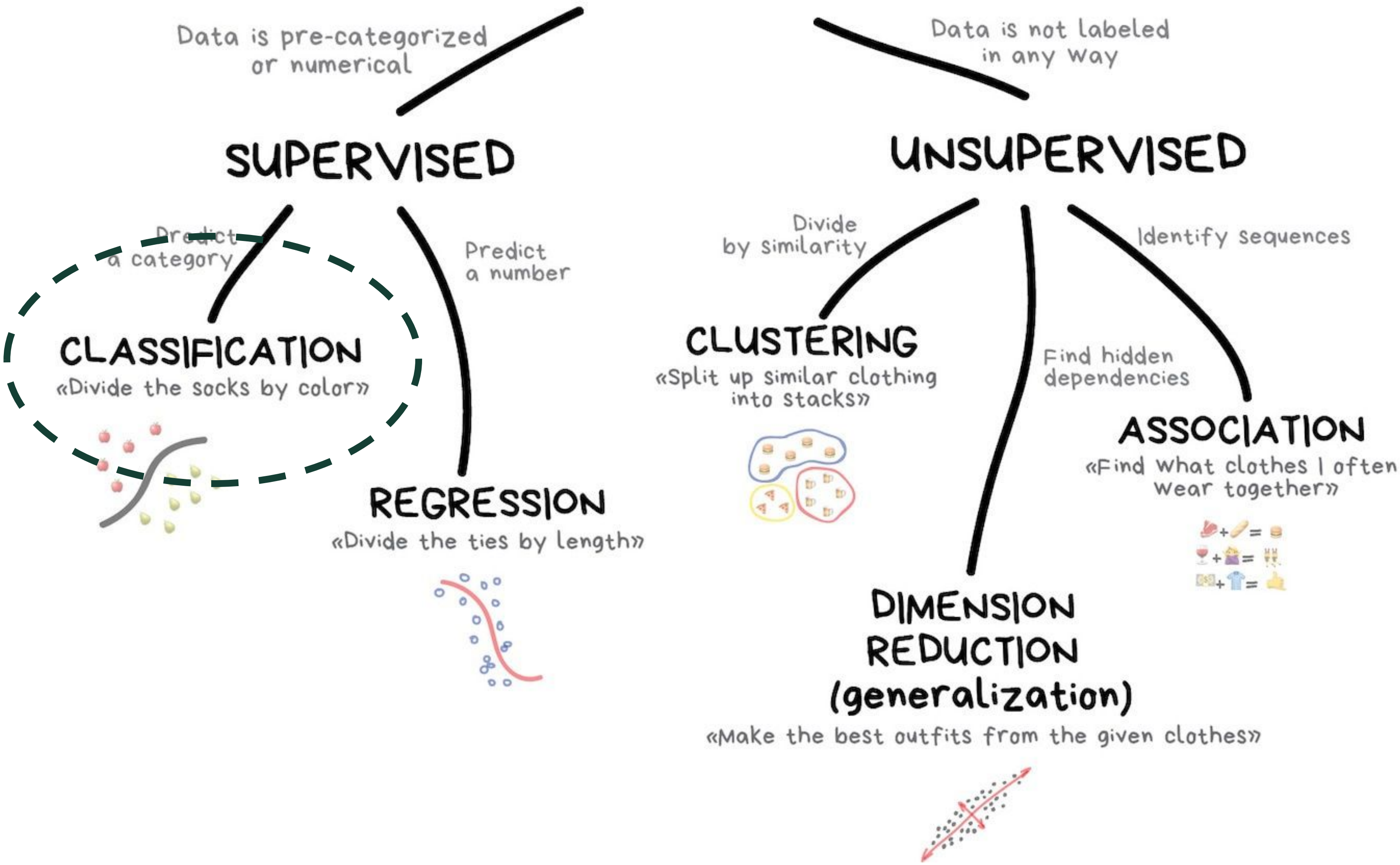


¿Qué es la regresión logística?


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

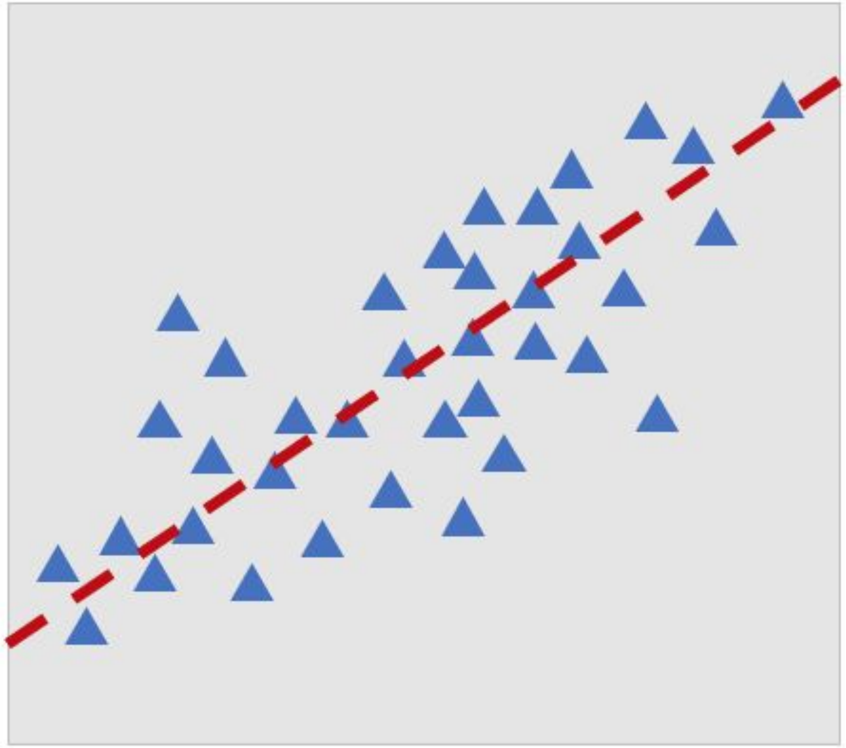

Machine Learning
Unsupervised Learning
Supervised Learning
Reinforcement Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Classical machine learning

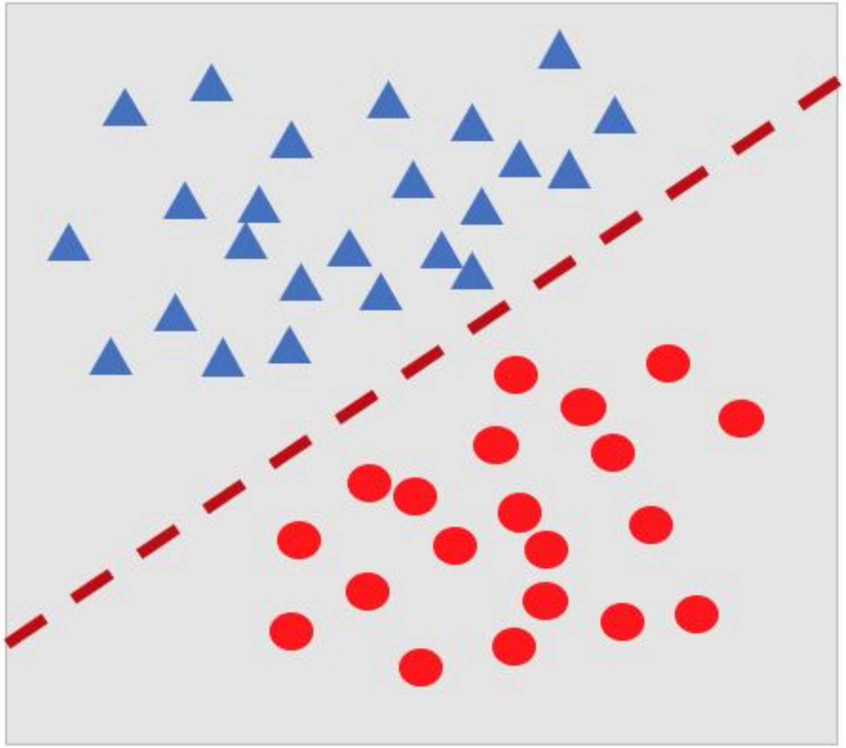


Logistic regression

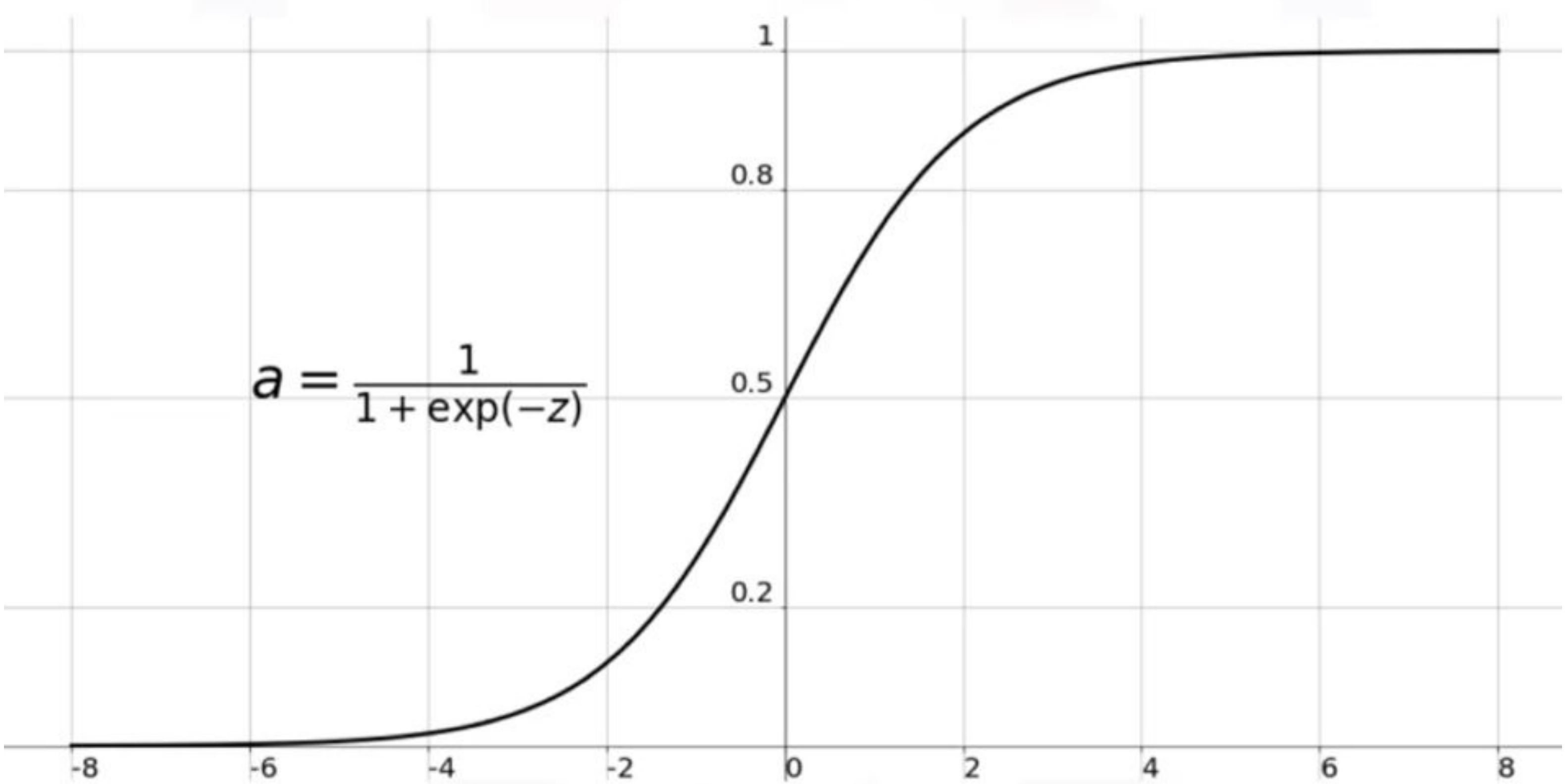
Regression



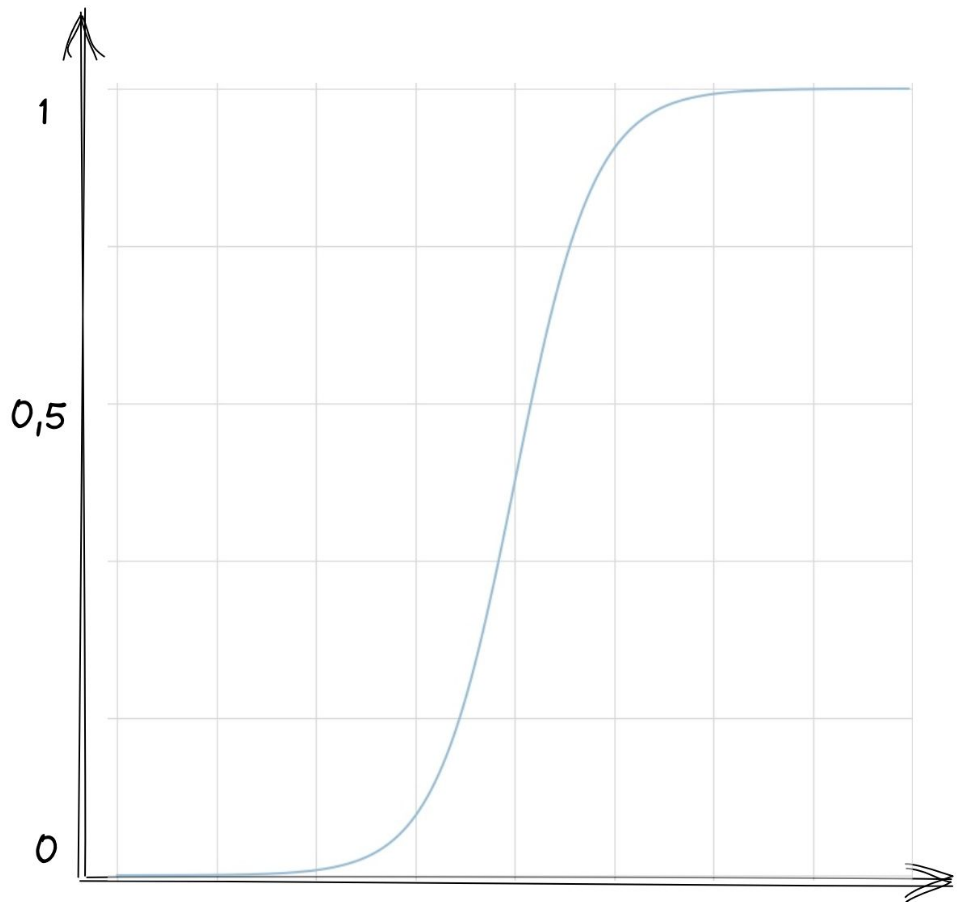
Classification



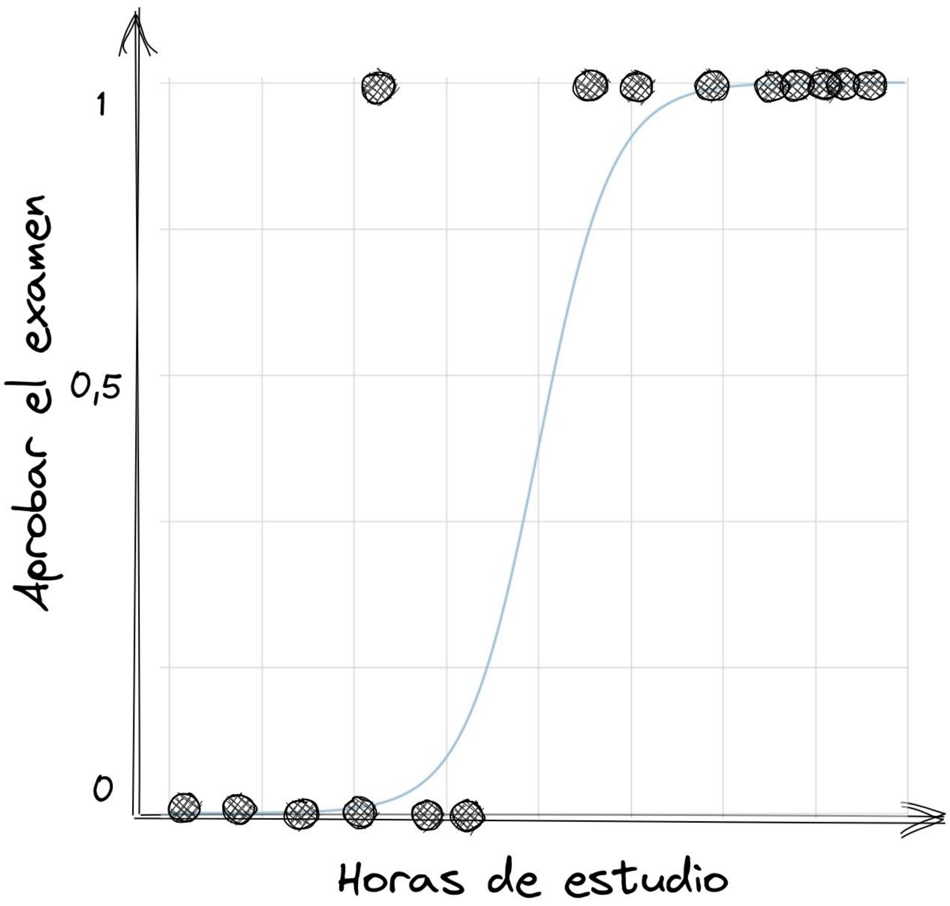
Sigmoid function



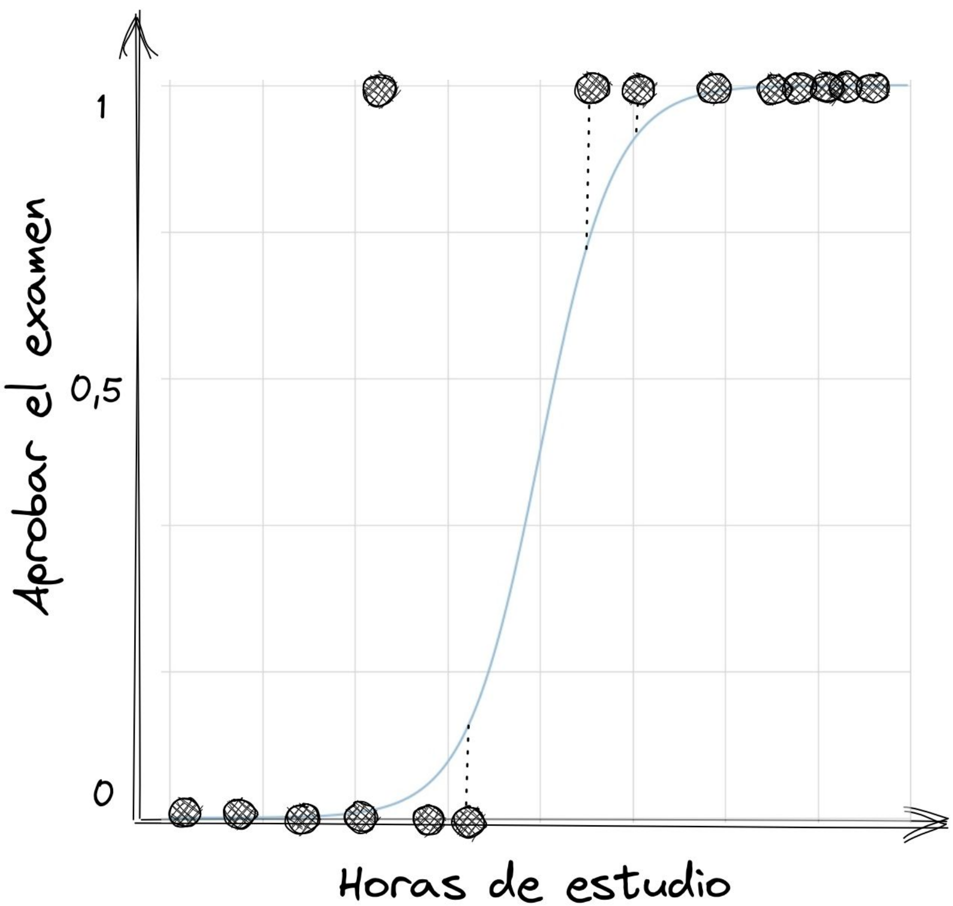
Logistic regression



Logistic regression



Logistic regression



Tu primera clasificación con regresión logística

¿Cuándo usar regresión logística?

Ventajas

- Fácil de implementar.
- Coeficientes interpretables.
- Inferencia de la importancia de cada característica.
- Clasificación en porcentajes.
- Excelentes resultados con datasets linealmente separables.
- Extendido a clasificación múltiple.

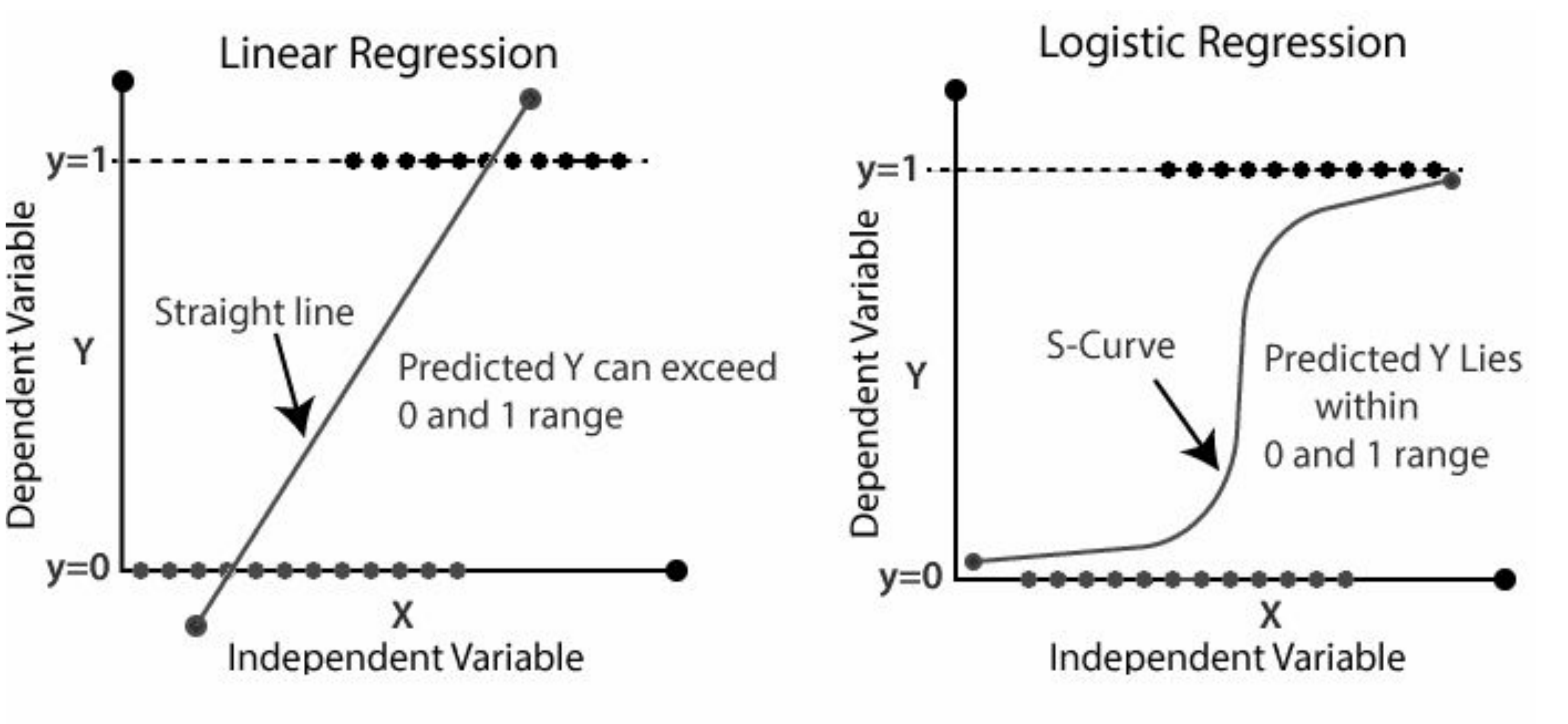
Desventajas

- Asume linealidad entre las variables dependientes.
- Overfitting sobre datasets de alta dimensionalidad.
- Le afecta la multicolinealidad de variables.
- Mejores resultados con datasets grandes.

¿Cuándo usarla?

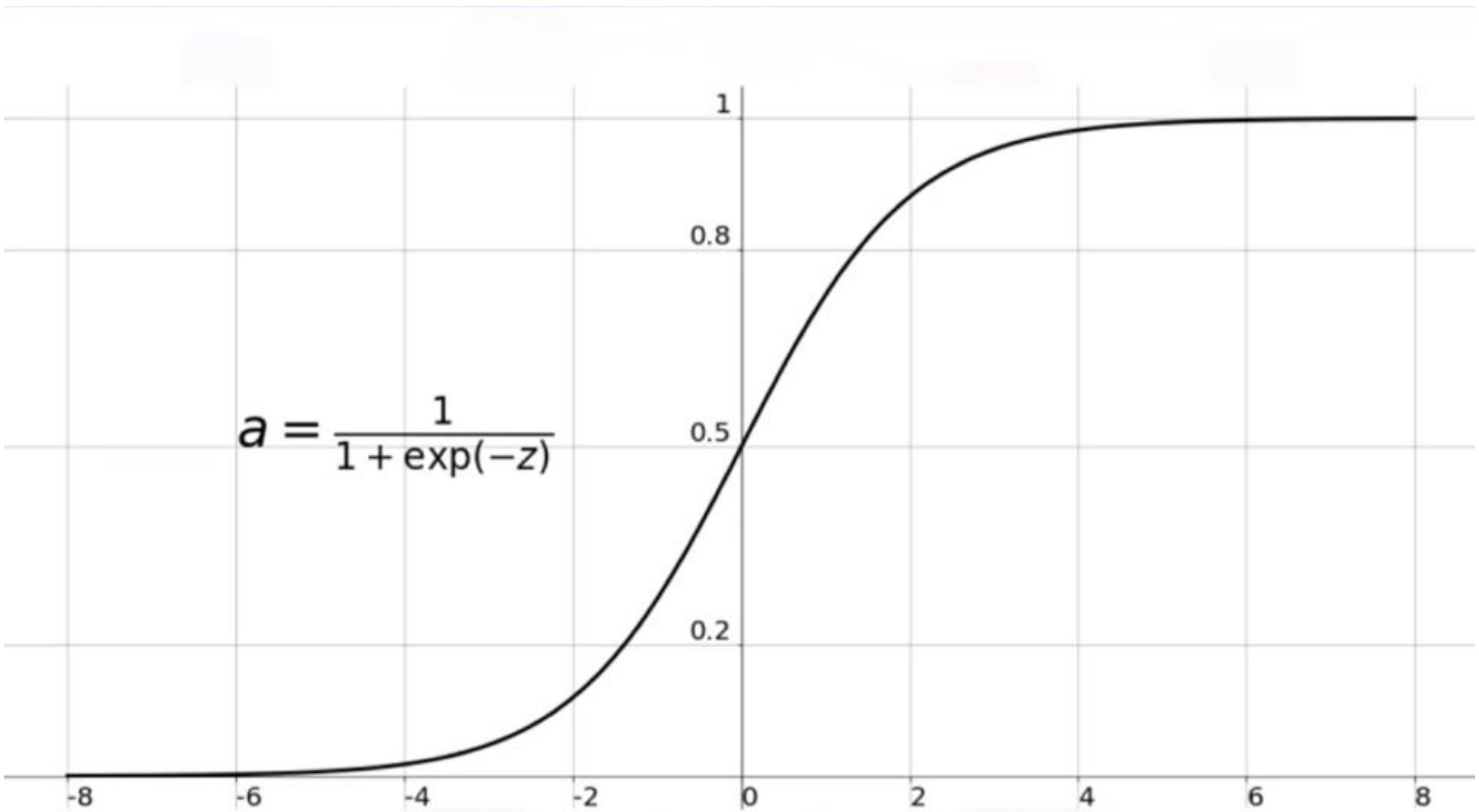
- Sencillo y rápido.
- Probabilidades de ocurrencia sobre un evento categórico.
- Dataset linealmente separable.
- Datasets grandes.
- Datasets balanceados.

Linear regression vs. logistic



Fórmula de regresión logística

Fórmula



Fórmula

$$\rho = \frac{1}{1 + e^{-(x)}}$$

Fórmula

$$\frac{1}{1+e^{-\log\left(\frac{p}{1-p}\right)}}$$

Odds

Probabilidad que el evento sea exitoso / 1 - (Probabilidad que el evento sea exitoso)

0.80 / 1 - (0.80)

0.80 / 0.20 = 4

Log odds

Odds of winning = 4/6 = 0.6666
log(Odds of winning) = log(0.6666) = -0.176
Odds of losing = 6/4 = 1.5
log(Odds of losing) = log(1.5) = 0.176

Fórmula

$$\frac{P}{1-P} = \beta_0 + \beta_1 X$$

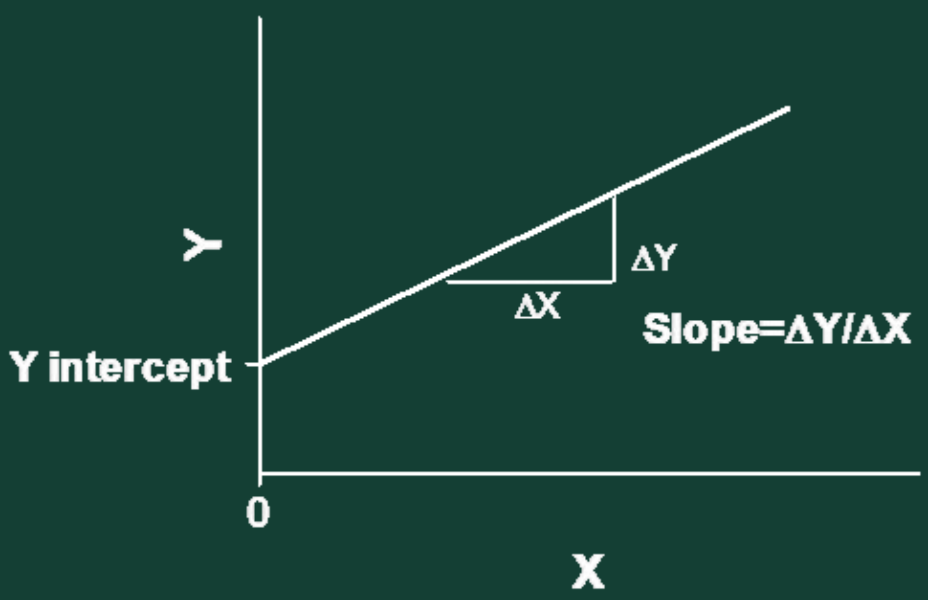
Fórmula

$$\log\left(\frac{P}{1-P}\right) = \beta_0 + B_1 X$$

Fórmula

$$Y = \beta_0 + B_1 X$$

Fórmula



Fórmula

Constant/Intercept

Independent Variable

$$Y_i = \beta_0 + \beta_1 X_i$$

Dependent Variable

Slope/Coefficient

Fórmula

$$P = \beta_0 + B_1 X$$

Fórmula

$$\frac{P}{1-P} = \beta_0 + \beta_1 X$$

Fórmula

$$\log\left(\frac{P}{1-P}\right) = \beta_0 + B_1 X$$

Fórmula

$$\exp[\log(\frac{p}{1-p})] = \exp(\beta_0 + \beta_1 x)$$

$$e^{\ln[\frac{p}{1-p}]} = e^{(\beta_0 + \beta_1 x)}$$

$$\frac{p}{1-p} = e^{\left(\beta_0 + \beta_1 x\right)}$$

$$p = e^{\left(\beta_0 + \beta_1 x\right)} - pe^{\left(\beta_0 + \beta_1 x\right)}$$

$$p = p[\frac{e^{\left(\beta_0 + \beta_1 x\right)}}{p} - e^{\left(\beta_0 + \beta_1 x\right)}]$$

$$1 = \frac{e^{\left(\beta_0 + \beta_1 x\right)}}{p} - e^{\left(\beta_0 + \beta_1 x\right)}$$

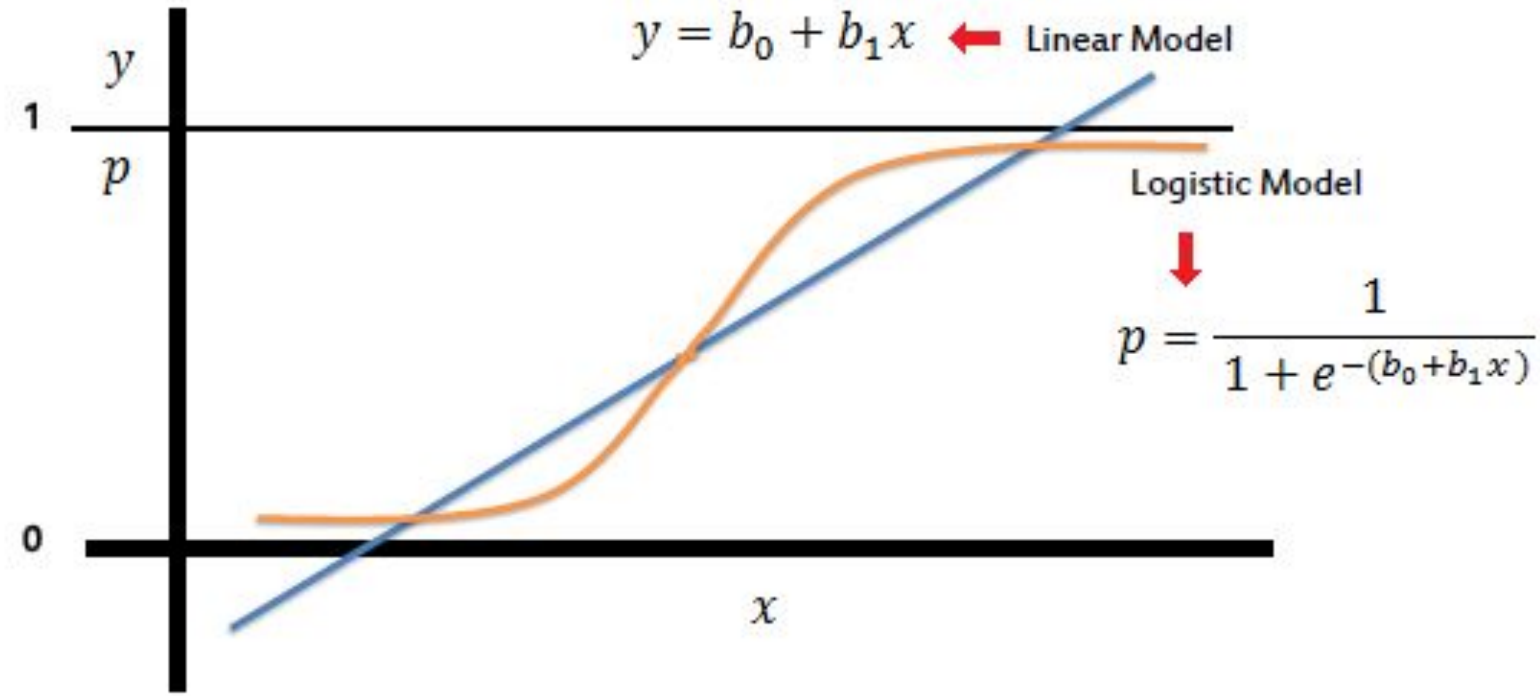
$$p[1 + e^{\left(\beta_0 + \beta_1 x\right)}] = e^{\left(\beta_0 + \beta_1 x\right)}$$

$$p = \frac{e^{\left(\beta_0 + \beta_1 x\right)}}{1 + e^{\left(\beta_0 + \beta_1 x\right)}}$$

Fórmula

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

Fórmula



Preparando los datos

Tipos de regresión logística

- Regresión binomial
- Regresión multinomial

Data pre-processing

- Eliminar duplicados.
- Evaluar valores nulos.
- Remover columnas innecesarias.
- Procesar datos categóricos.
- Remover outliers.
- Escalar data.

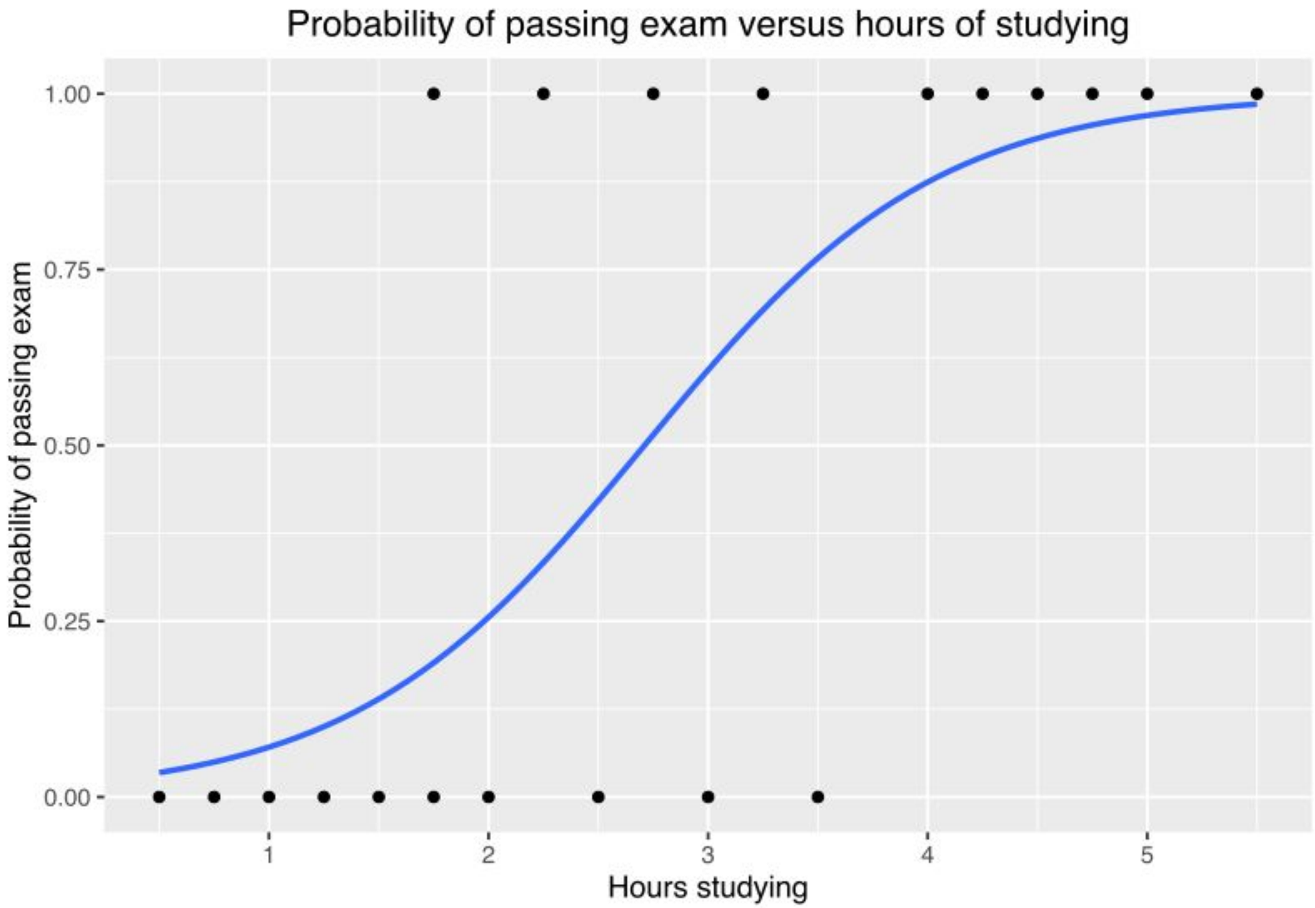
Análisis de correlación y escalabilidad de los datos

Análisis exploratorio de datos

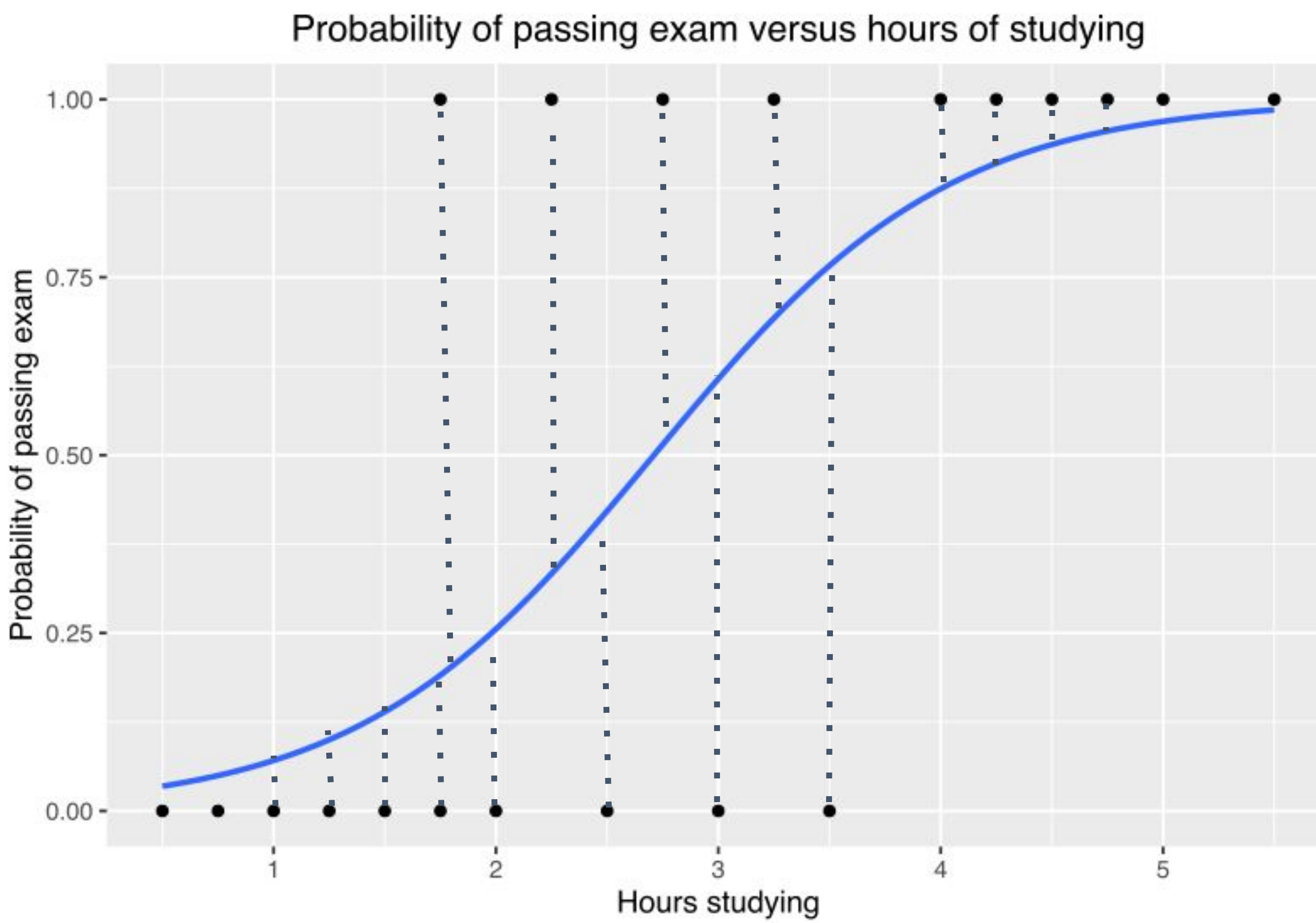
Entrenamiento con regresión logística binomial

Evaluando el modelo (MLE)

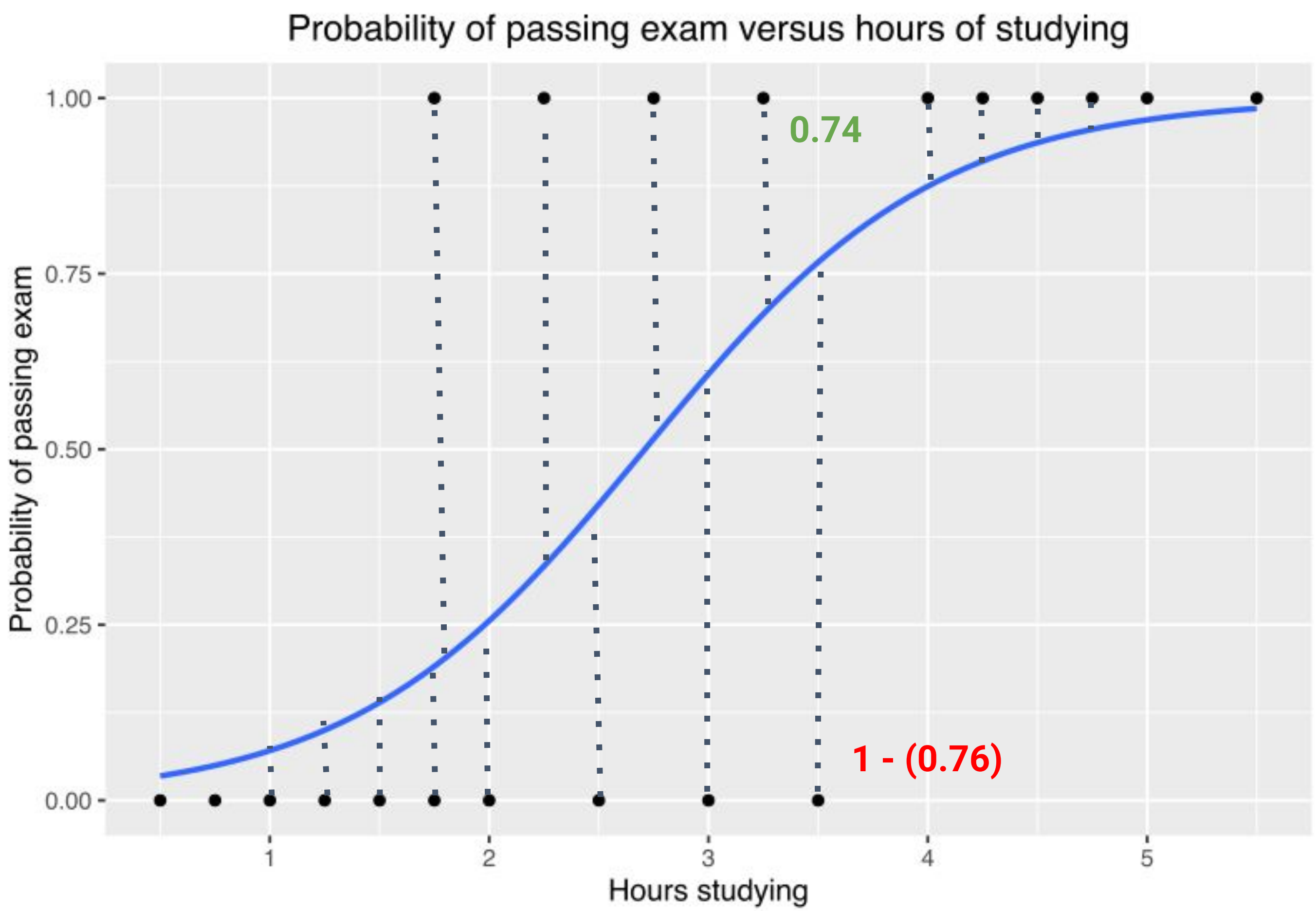
Logistic regression



Projection


Probability of passing exam versus hours of studying
Probability of passing exam
1.00
0.75
0.50
0.25
0.00
1
2
3
4
5
Hours studying

Projection

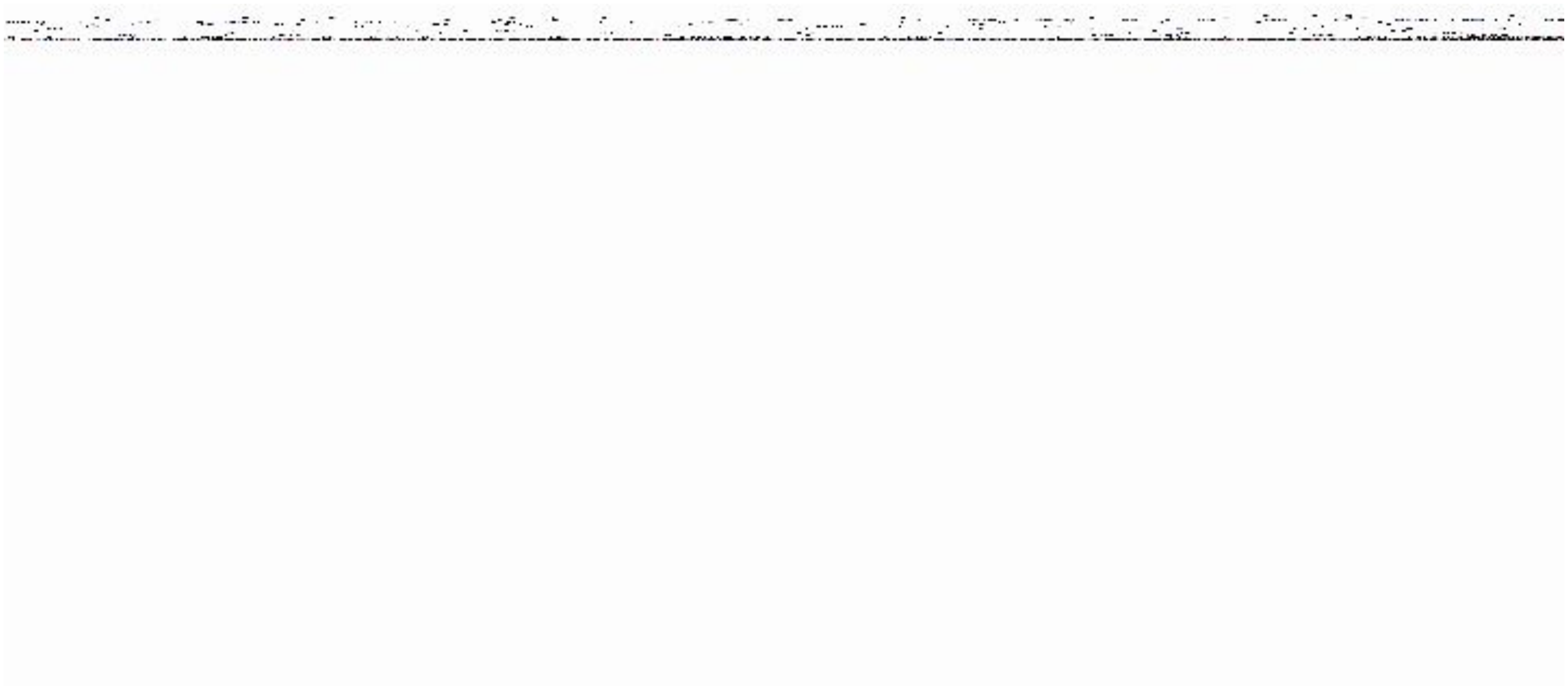

Probability of passing exam versus hours of studying
1.00
0.75
0.50
0.25
0.00
Probability of passing exam
0.74
1 - (0.76)
1
2
3
4
5
Hours studying

MLE

0.60 * 0.78 * 0.65 * 0.85 * 0.99 *
(1-0.56) * (1-0.20) * (1-0.10)*(1-0.15) * (1-0.05)
= 0.065

log(0.60) * log(0.78) * log(0.65) *log(0.85) *
log(0.99) * log(1-0.56) * log(1-0.20) * log(1-0.10)
* log(1-0.15) * log(1-0.05) = 1.039e-8

Gradient descent



Gradient descent

Cost function

ID	Actual	Predicted Probabilities
ID6	1	0.94
ID1	1	0.9
ID7	1	0.78
ID8	0	0.56
ID2	0	0.51
ID3	1	0.47
ID4	1	0.32
ID5	0	0.1

Cost function

ID	Actual	Predicted Probabilities	Corrected Probabilities
ID6	1	0.94	0.94
ID1	1	0.9	0.9
ID7	1	0.78	0.78
ID8	0	0.56	0.44
ID2	0	0.51	0.49
ID3	1	0.47	0.47
ID4	1	0.32	0.32
ID5	0	0.1	0.9

Cost function

ID	Actual	Predicted Probabilities	Corrected Probabilities	Log
ID6	1	0.94	0.94	-0,02687
ID1	1	0.9	0.9	-0.04576
ID7	1	0.78	0.78	-0.10791
ID8	0	0.56	0.44	-0.35655
ID2	0	0.51	0.49	-0.3098
ID3	1	0.47	0.47	-0.3279
ID4	1	0.32	0.32	-0.49485
ID5	0	0.1	0.9	-0.04576

Cost function

$$\text{Log loss} = \frac{1}{N} \sum_{i=1}^{N} -\left(y_i * \log(p_i) + (1-y_i) * \log(1-p_i) \right)$$

P(i) = Probabilidad de la clase 1

1- P(i) = Probabilidad de la clase 0

Cost function

Predicted probability	Actual class	$y_i \times ln(p_i)$	$(1 - y_i) \times ln(1 - p_i)$	$y_i \times ln(p_i) + (1 - y_i) \times ln(1 - p_i)$
0.8	Positive (=1)	$1 \times ln0.8 = -0.2231$	$0 \times ln0.2 = 0$	-0.2231
0.15	Positive (=1)	$1 \times ln0.15 = -1.8971$	$0 \times ln0.85 = 0$	-1.8971
0.95	Negative (=0)	$0 \times ln0.95 = 0$	$1 \times ln0.05 = -2.9957$	-2.9957

Gradient descent

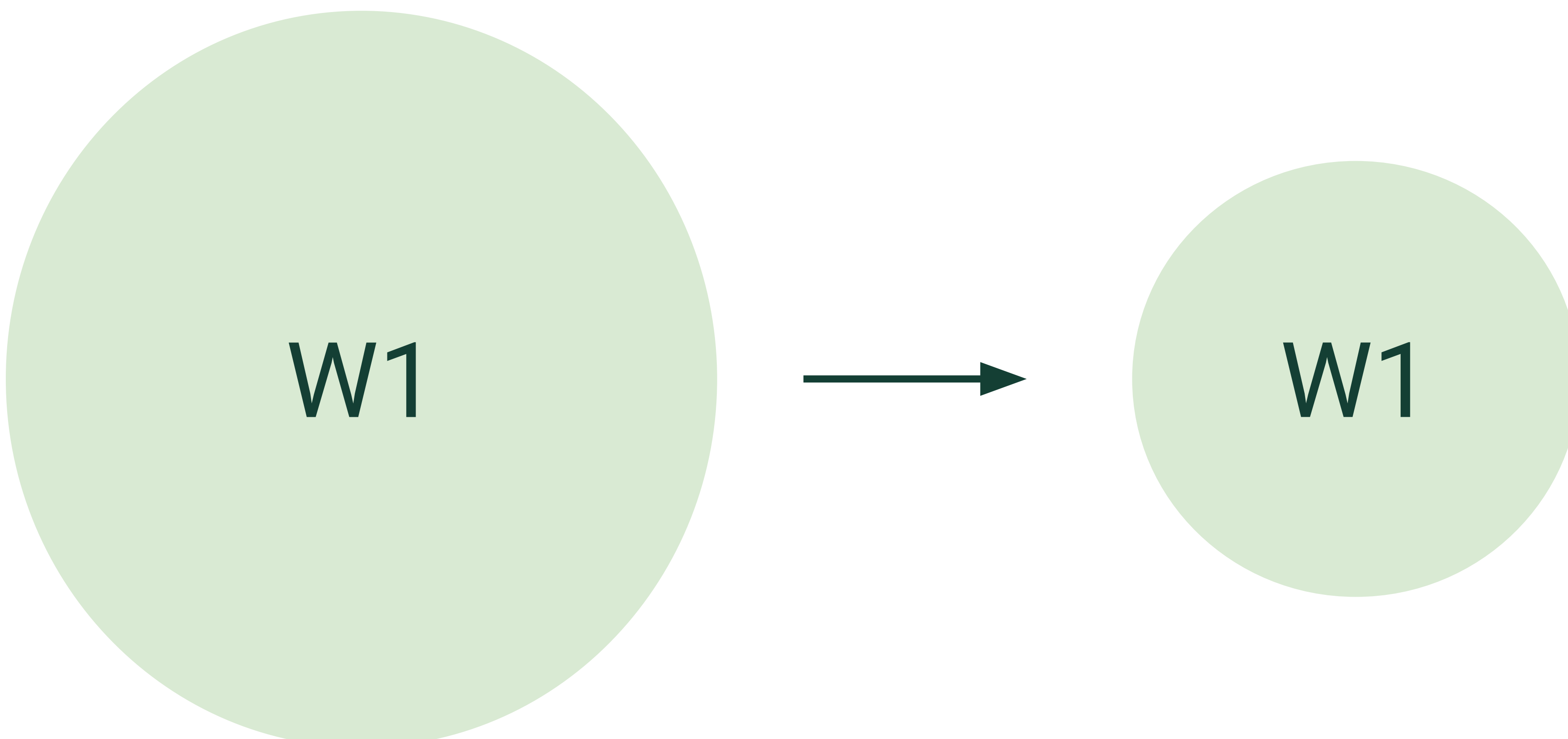
Análisis de resultados de regresión logística

Regularizers

Regularización

Reducir la complejidad en el modelo.

Regularización



Regularización

L1 Regularization

$$\text{Cost} = \sum_{i=0}^{N}\left(y_i - \sum_{j=0}^{M} x_{ij}W_j\right)^2 + \lambda \sum_{j=0}^{M} |W_j|$$

L2 Regularization

$$\text{Cost} = \underbrace{\sum_{i=0}^{N}\left(y_i - \sum_{j=0}^{M} x_{ij}W_j\right)^2}_{\text{Loss function}} + \underbrace{\lambda \sum_{j=0}^{M} W_j^2}_{\text{Regularization Term}}$$

Regularización

Parameters::

penalty : ***{'l1', 'l2', 'elasticnet', 'none'}, default='l2'***

Specify the norm of the penalty:

- `'none'`: no penalty is added;
- `'l2'`: add a L2 penalty term and it is the default choice;
- `'l1'`: add a L1 penalty term;
- `'elasticnet'`: both L1 and L2 penalty terms are added.

> **Warning:** Some penalties may not work with some solvers. See the parameter `solver` below, to know the compatibility between the penalty and solver.

New in version 0.19: l1 penalty with SAGA solver (allowing 'multinomial' + L1)

dual : ***bool, default=False***

Dual or primal formulation. Dual formulation is only implemented for l2 penalty with liblinear solver. Prefer dual=False when n_samples > n_features.

tol : ***float, default=1e-4***

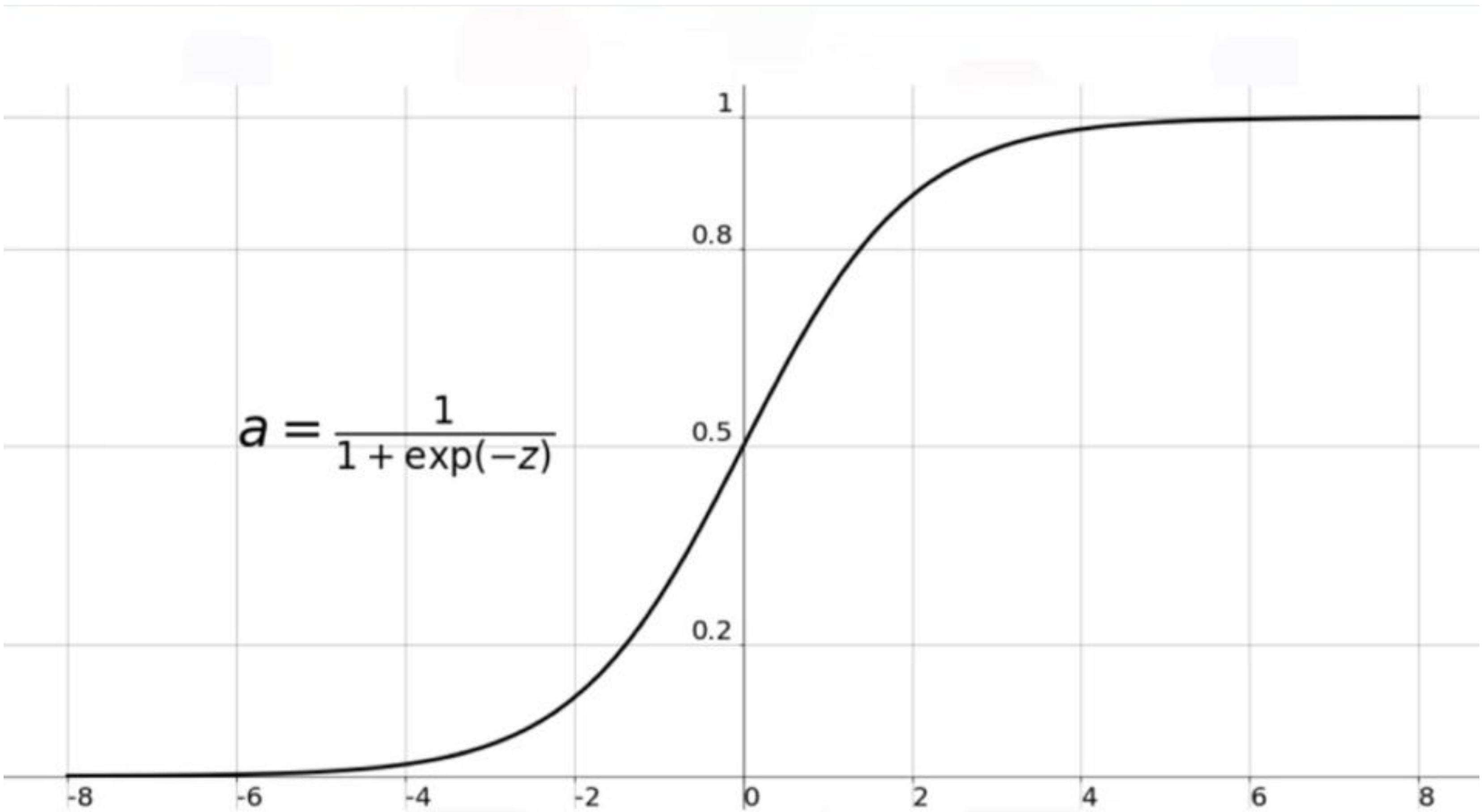
Tolerance for stopping criteria.

C : ***float, default=1.0***

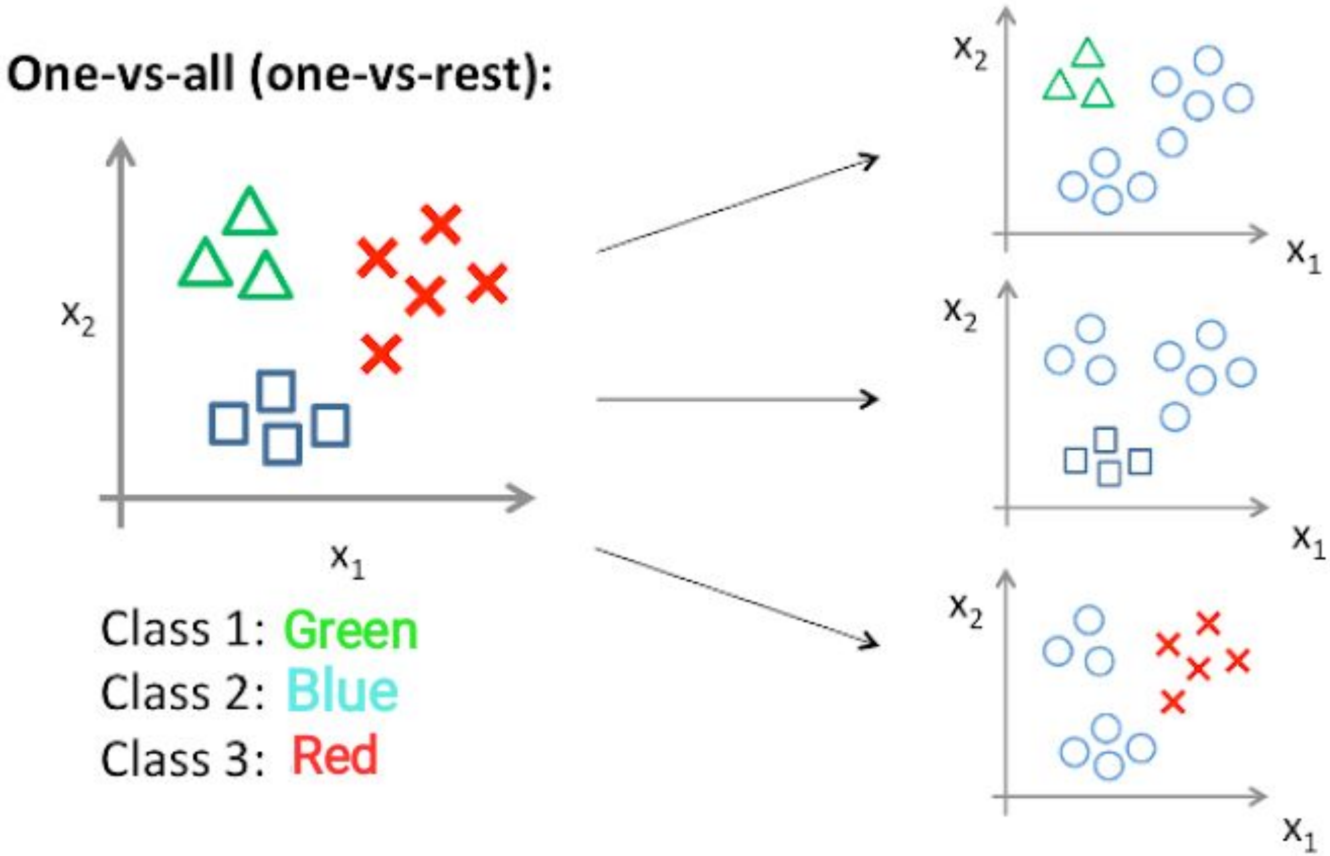
Inverse of regularization strength; must be a positive float. Like in support vector machines, smaller values specify stronger regularization.

¿Cómo funciona la regresión logística multiclase?

Fórmula

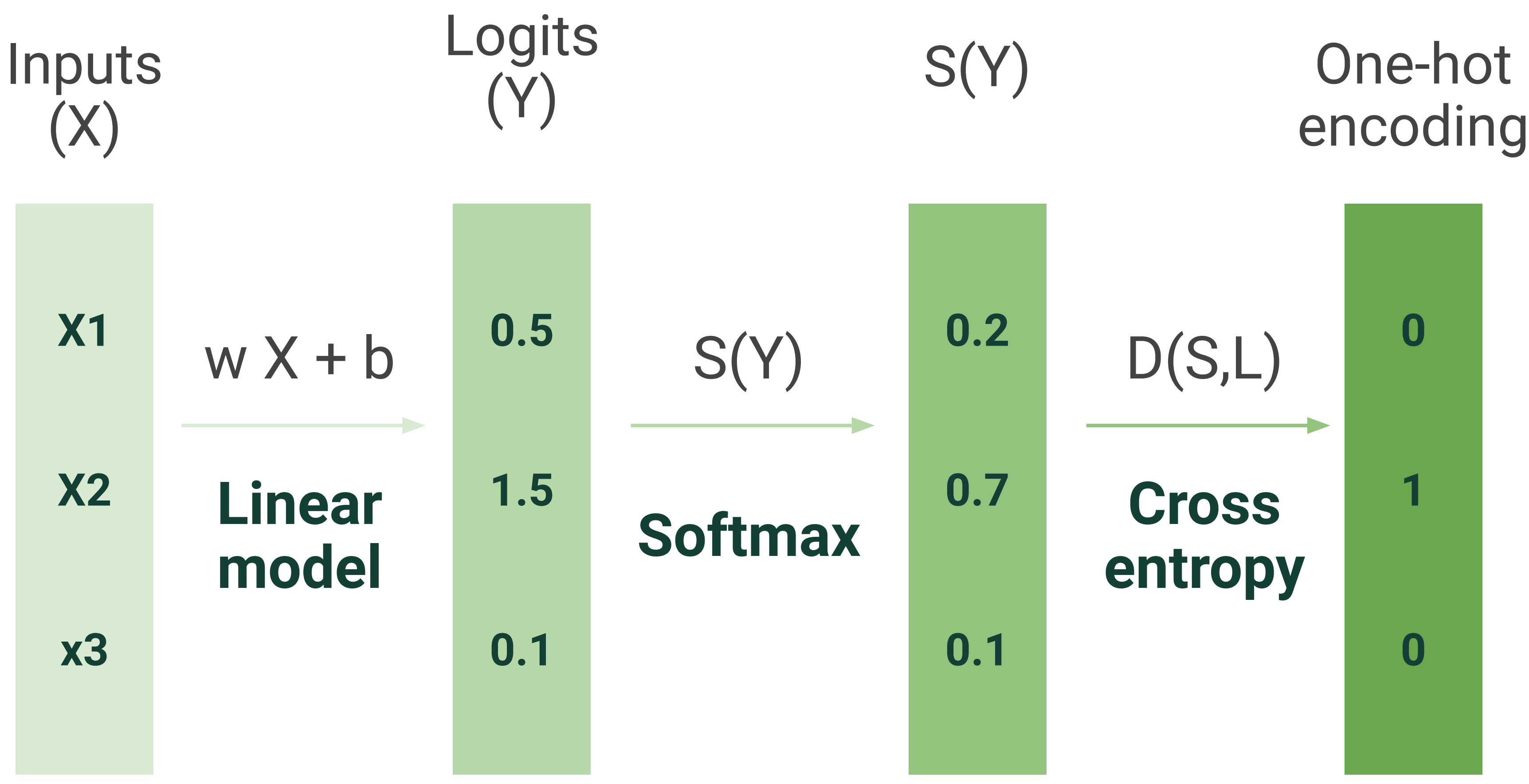


One vs. rest



One-vs-all (one-vs-rest):
x_2
x_1
Class 1: Green
Class 2: Blue
Class 3: Red
x_2
x_1
x_2
x_1
x_2
x_1

Multinominal logistic classifier



Scikit-learn solvers

	Solvers				
Penalties	**'liblinear'**	**'lbfgs'**	**'newton-cg'**	**'sag'**	**'saga'**
Multinomial + L2 penalty	no	yes	yes	yes	yes
OVR + L2 penalty	yes	yes	yes	yes	yes
Multinomial + L1 penalty	no	no	no	no	yes
OVR + L1 penalty	yes	no	no	no	yes
Elastic-Net	no	no	no	no	yes
No penalty ('none')	no	yes	yes	yes	yes
Behaviors					
Penalize the intercept (bad)	yes	no	no	no	no
Faster for large datasets	no	no	no	yes	yes
Robust to unscaled datasets	yes	yes	yes	no	no

Scikit-learn

sklearn.linear_model.LogisticRegression

class `sklearn.linear_model.`**`LogisticRegression`**(*penalty='l2'*, ***, *dual=False*, *tol=0.0001*, *C=1.0*, *fit_intercept=True*, *intercept_scaling=1*, *class_weight=None*, *random_state=None*, *solver='lbfgs'*, *max_iter=100*, *multi_class='auto'*, *verbose=0*, *warm_start=False*, *n_jobs=None*, *l1_ratio=None*) [source]

Logistic Regression (aka logit, MaxEnt) classifier.

In the multiclass case, the training algorithm uses the one-vs-rest (OvR) scheme if the 'multi_class' option is set to 'ovr', and uses the cross-entropy loss if the 'multi_class' option is set to 'multinomial'. (Currently the 'multinomial' option is supported only by the 'lbfgs', 'sag', 'saga' and 'newton-cg' solvers.)

This class implements regularized logistic regression using the 'liblinear' library, 'newton-cg', 'sag', 'saga' and 'lbfgs' solvers. **Note that regularization is applied by default**. It can handle both dense and sparse input. Use C-ordered arrays or CSR matrices containing 64-bit floats for optimal performance; any other input format will be converted (and copied).

The 'newton-cg', 'sag', and 'lbfgs' solvers support only L2 regularization with primal formulation, or no regularization. The 'liblinear' solver supports both L1 and L2 regularization, with a dual formulation only for the L2 penalty. The Elastic-Net regularization is only supported by the 'saga' solver.

Read more in the User Guide.

Carga y preprocesamiento de datos

Regresión
logística multinomial

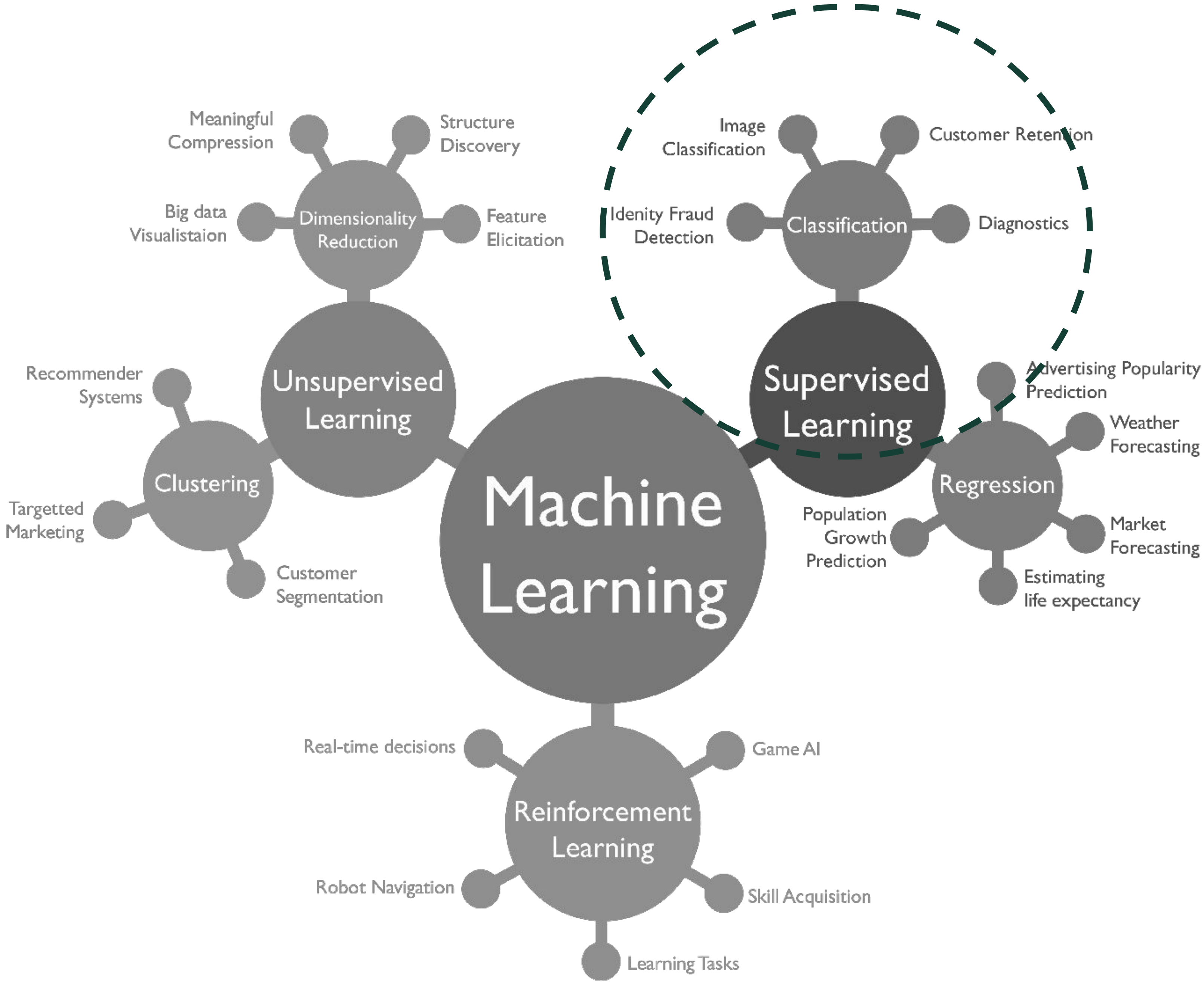
Análisis exploratorio y escalamiento de datos

Regresión logística multinomial

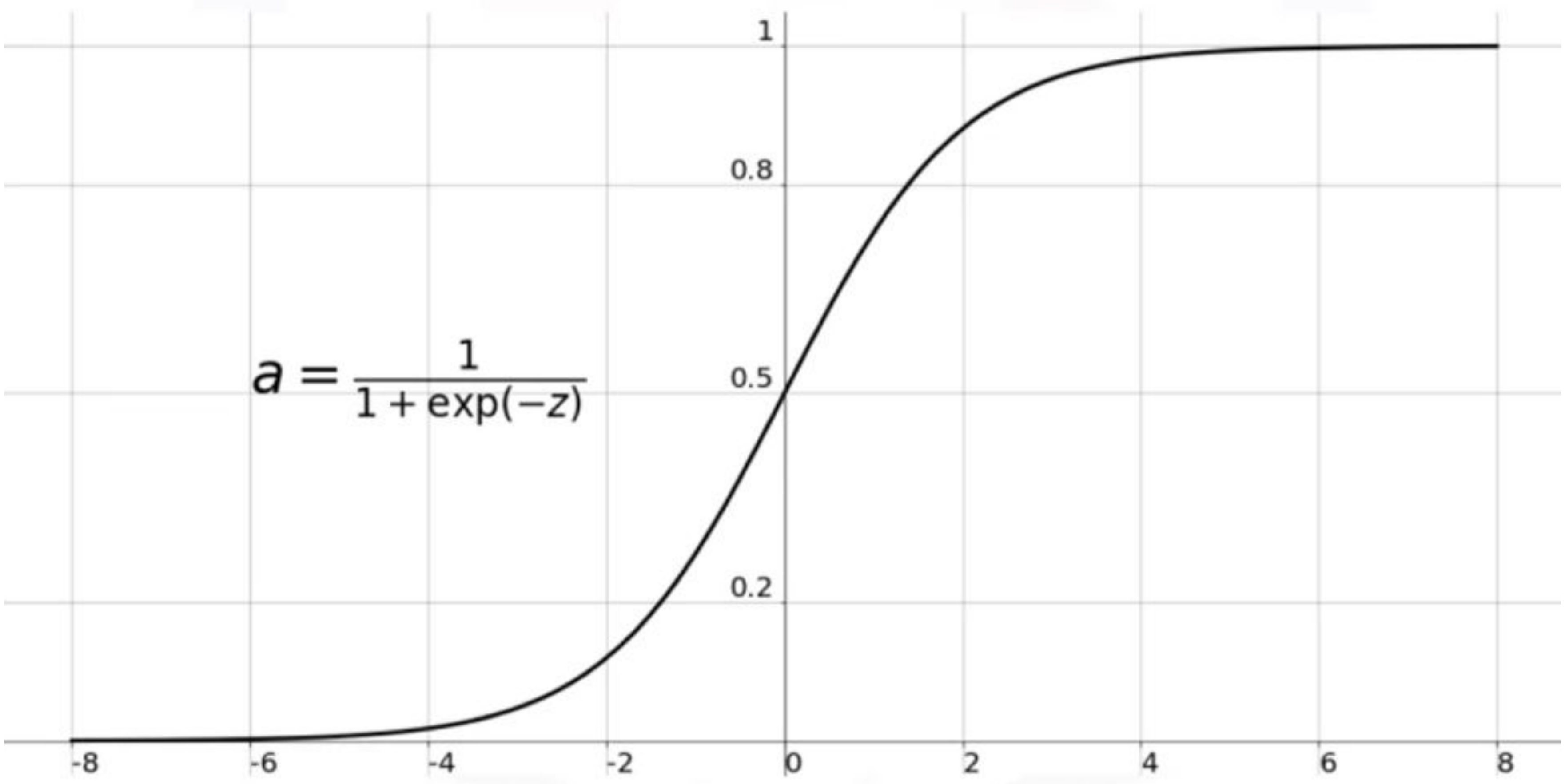
Entrenamiento y evaluación del modelo

Regresión
logística multinomial

Proyecto final y cierre


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Sigmoid function



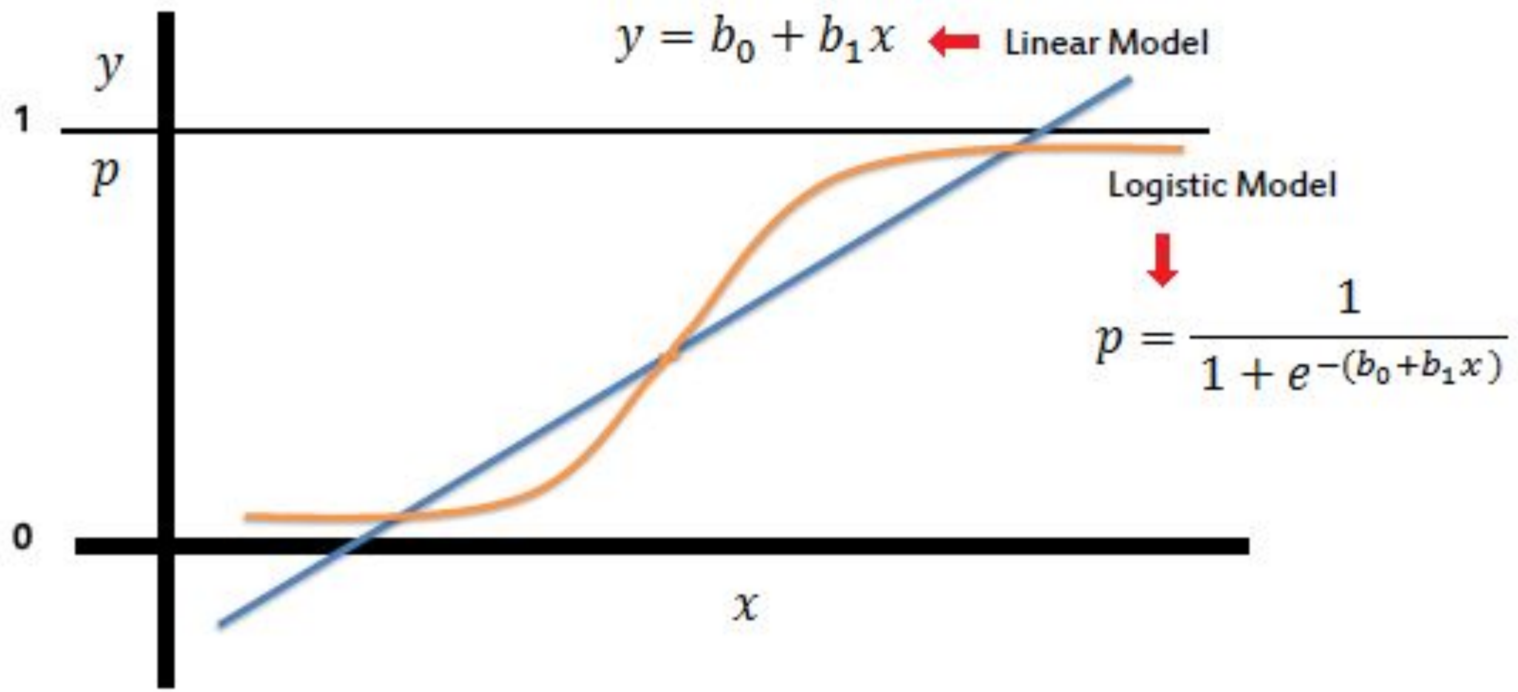
Fórmula

$$\rho = \frac{1}{1 + e^{-(x)}}$$

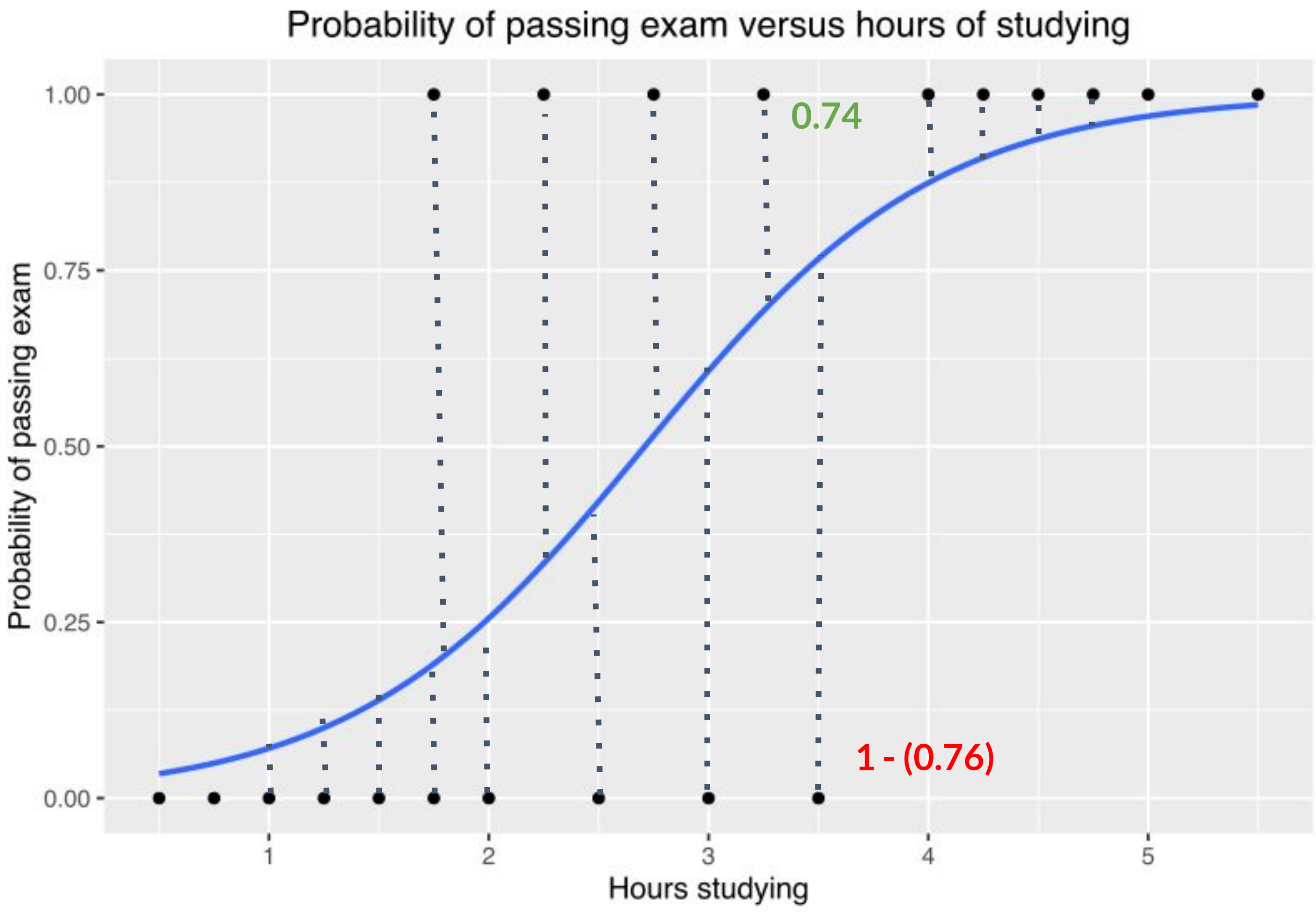
Fórmula

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

Fórmula



Projection

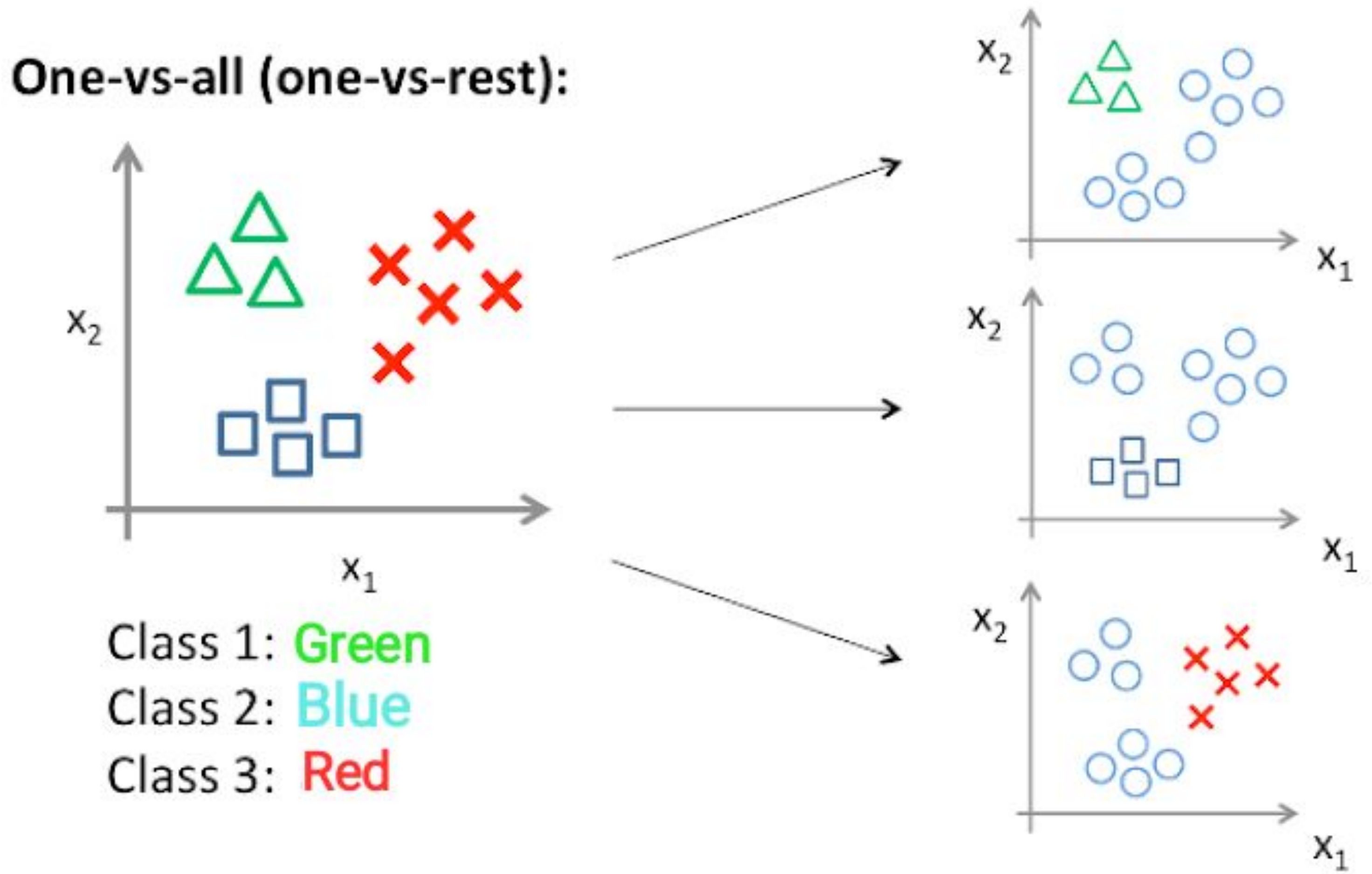


Probability of passing exam versus hours of studying
1.00
0.75
0.50
0.25
0.00
Probability of passing exam
1
2
3
4
5
Hours studying
0.74
1 - (0.76)

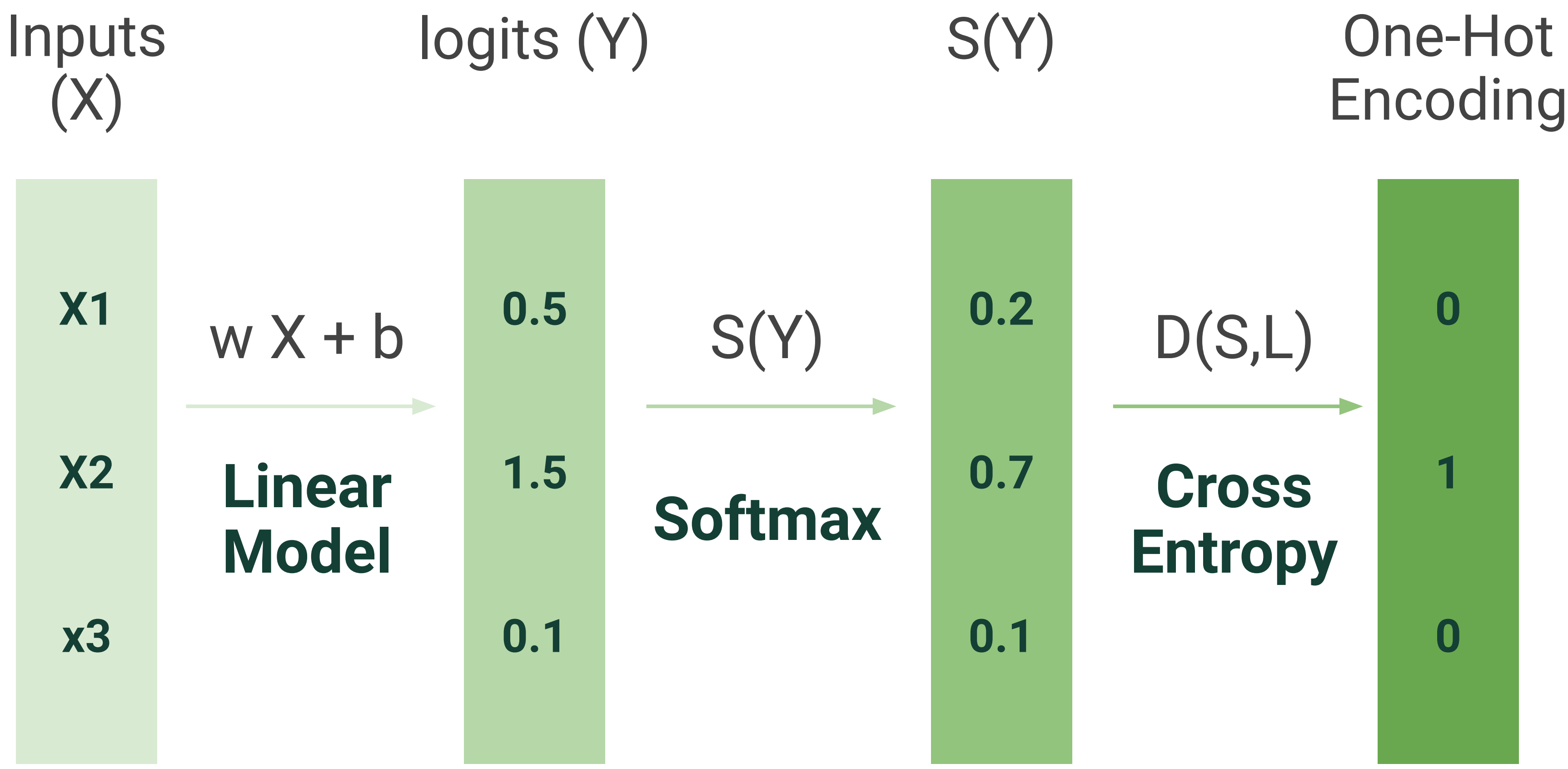
Gradient descent

Gradient descent

One vs. rest

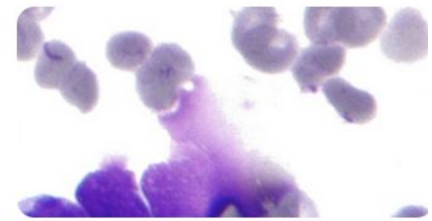

One-vs-all (one-vs-rest):
x_2
x_1
Class 1: Green
Class 2: Blue
Class 3: Red

Multinominal logistic classifier



Proyecto final

Breast Cancer Wisconsin (Diagnostic) Data Set



Predict whether the cancer is benign or malignant

Data Code (2252) Discussion (49) Metadata

About Dataset

Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image.
n the 3-dimensional space is that described in: [K. P. Bennett and O. L. Mangasarian: "Robust Linear Programming Discrimination of Two Linearly Inseparable Sets", Optimization Methods and Software 1, 1992, 23-34].

This database is also available through the UW CS ftp server:
ftp ftp.cs.wisc.edu
cd math-prog/cpo-dataset/machine-learn/WDBC/

Also can be found on UCI Machine Learning Repository: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

Attribute Information:

Usability

8.53

License

CC BY-NC-SA 4.0

Expected update frequency

Not specified

Proyecto final

Activity Overview

ACTIVITY STATS

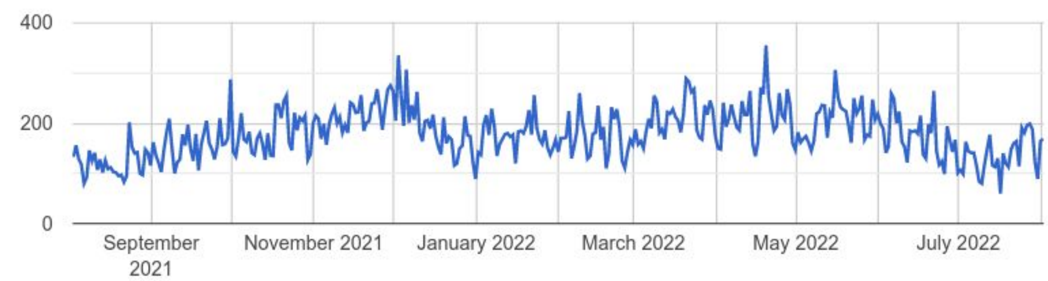
VIEWS
1315679

DOWNLOADS
225472

DOWNLOAD PER VIEW RATIO
0.17

TOTAL UNIQUE CONTRIBUTORS
1976

Downloads



NOTEBOOKS STATS

NOTEBOOKS
2252

NOTEBOOK COMMENTS
4012

UPVOTE PER NOTEBOOK RATIO
5.94

NOTEBOOK UPVOTES
13369

TOP CONTRIBUTORS

- DATAI
- Manish Kumar
- Miri Choi

DISCUSSION STATS

TOPICS
46

TOTAL COMMENTS
82

UPVOTE PER POST RATIO
0.94

DISCUSSION UPVOTES
77

Carlos Andrés Alarcón

@Alarcon7a