Categorización de productos con Deep Learning

Keras, Tensorflow, Pandas, Numpy, etc

Agustín Sarasúa / agustin.sarasua@mercadolibre.com

Pablo Zamudio / pablo.zamudio@mercadolibre.com

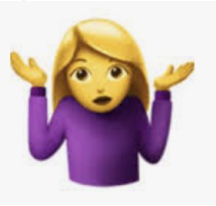
Agenda

- Intro / Problema
- Conceptos generales
  - AI vs ML vs DL
  - ¿Por que Deep Learning?
  - Tipos de algoritmos
- Redes Neuronales
  - Cost & Loss Function
  - Gradient Descent
- Intro Natural Language Processing
  - Tokenization
  - Vectorization
- Workflow de trabajo
- Intro a Python, Numpy y Pandas
- Entrenar una NN

Intro / Problema

“Desmalezadora Bordeadora Gardenplus Gp Naftera”

Accesorios para Vehículos



Hogar, Muebles y Jardín > Jardín y Exterior > Máquinas para el Jardín > **Desmalezadoras y Repuestos**

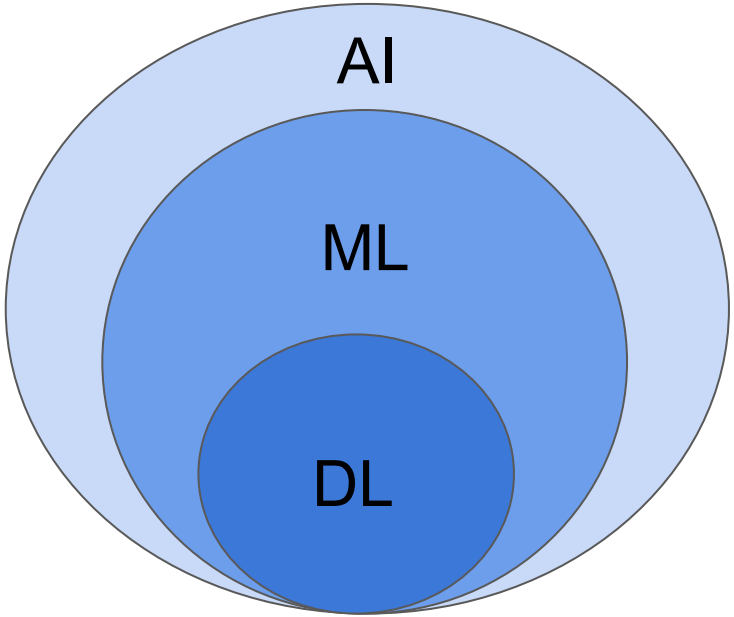
Intro / Problema

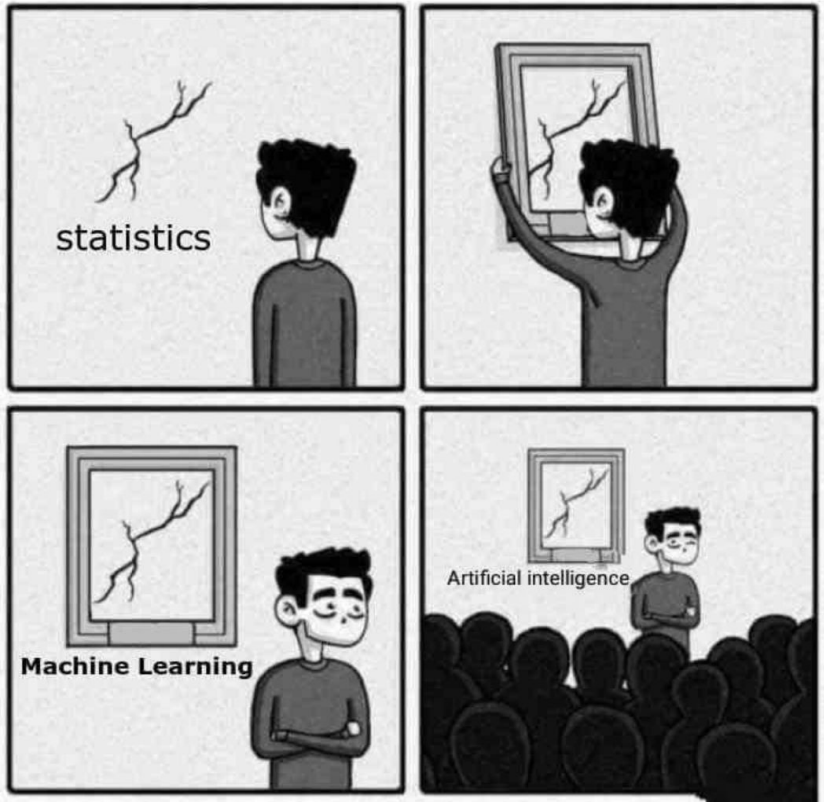
- **20** árboles de categoría (1 por país)
- **+3K** categorías
  - E.g. CELLPHONES, TABLETS, BICYCLES, etc.

Agenda

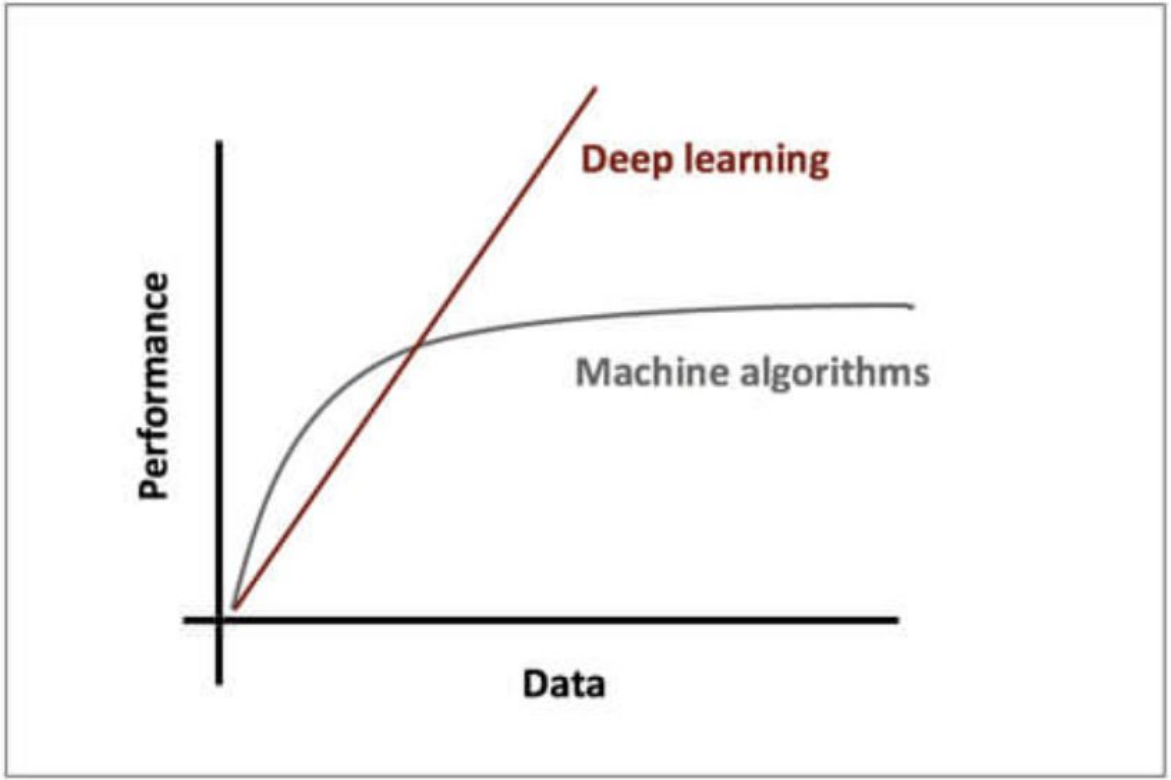
- Intro / Problema
- **Conceptos generales**
  - **AI vs ML vs DL**
  - **¿Por que Deep Learning?**
  - **Tipos de algoritmos**
- Redes Neuronales
  - Cost & Loss Function
  - Gradient Descent
- Intro Natural Language Processing
  - Tokenization
  - Vectorization
- Workflow de trabajo
- Intro a Python, Numpy y Pandas
- Entrenar una NN

AI vs ML vs DL


AI
ML
DL

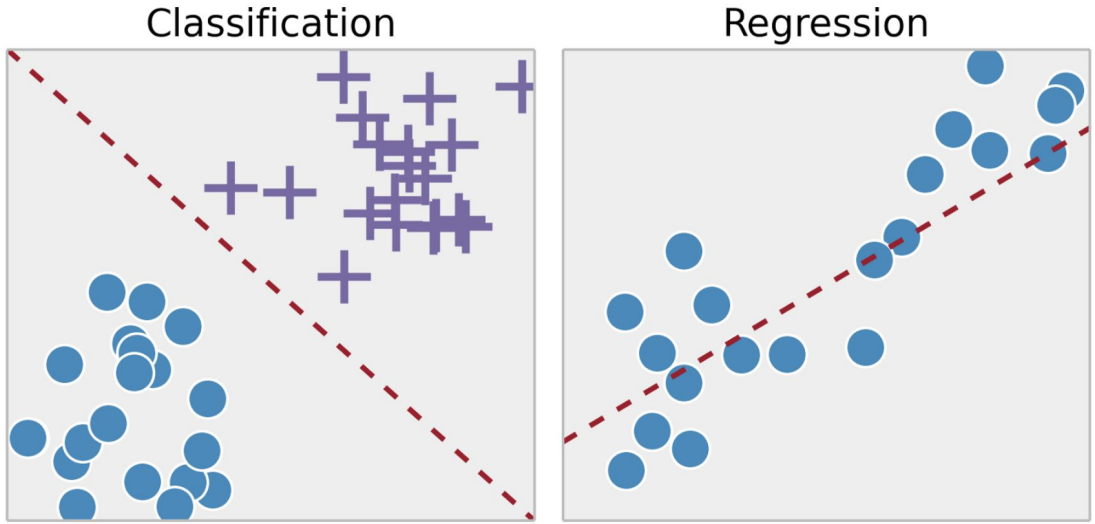

statistics
Machine Learning
Artificial intelligence

¿Por que Deep Learning?



Tipos de algoritmos

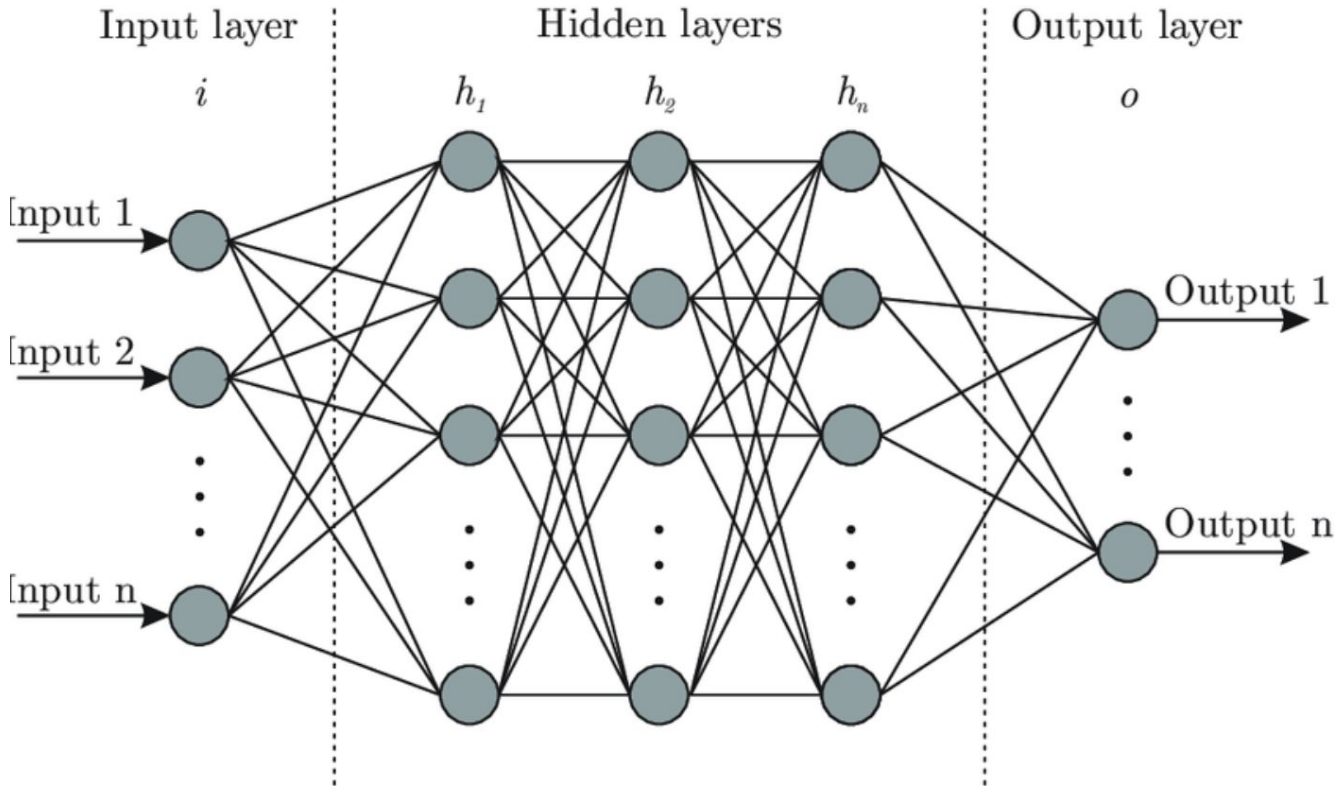
- **Supervisados**
  - **Clasificación**
  - Regresión
- No supervisados
  - Clustering
- Reinforcement Learning
- etc...



Agenda

- Intro / Problema
- Conceptos generales
  - AI vs ML vs DL
  - ¿Por que Deep Learning?
  - Tipos de algoritmos
- **Redes Neuronales**
  - **Cost & Loss Function**
  - **Gradient Descent**
- Intro Natural Language Processing
  - Tokenization
  - Vectorization
- Workflow de trabajo
- Intro a Python, Numpy y Pandas
- Entrenar una NN

Redes Neuronales - Arquitectura

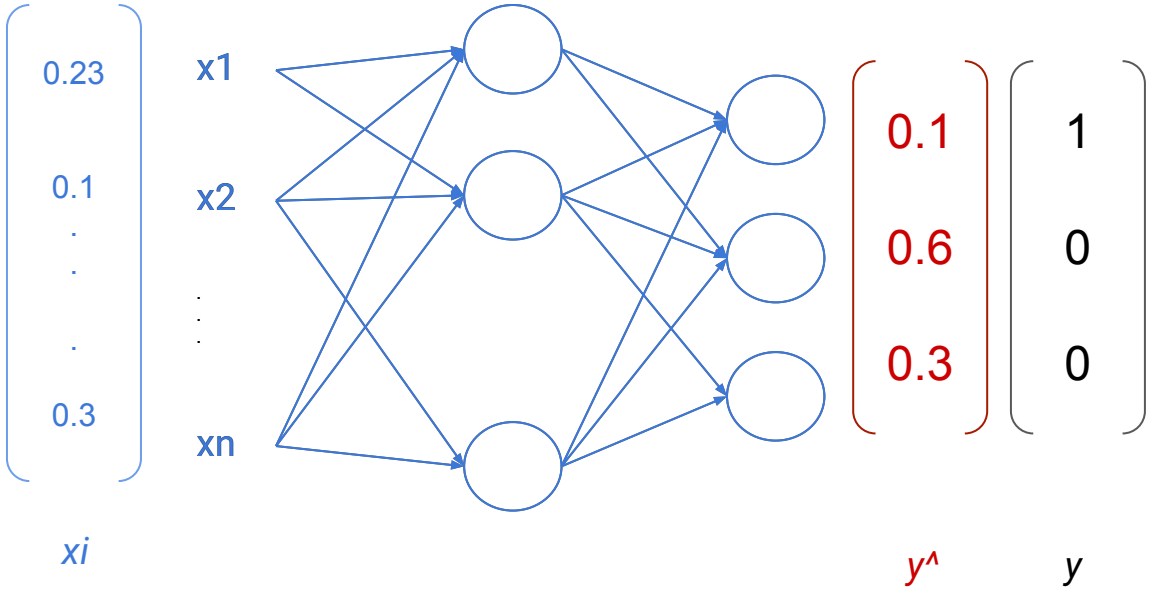


Redes Neuronales - Conceptos

- Parameters: Los aprende el modelo
- Hyperparameters:
  - Lo que podemos ajustar. Ej: #unidades #capas, learning rate
- Loss Function:
  - Aplicada a un solo ejemplo del training set
- Cost Function:
  - Aplicada al training set completo
  - Efectividad de los parámetros aprendidos sobre training set
- Features: El input que usamos para alimentar el modelo

Visualizando una predicción

"Apple Iphone X 64gb"



Loss Function: Categorical Cross Entropy

$$L(y, \hat{y}) = -\sum_{i}^{C} y \, log(\hat{y})$$

L(y,y^) = - 1*log(0.1) = 2.302

L(y,y^) = - 1*log(0.9) = 0.105

	y	$\hat{y}$		$\hat{y}$
CELLPHONE	1	0.1	N iteraciones de gradient descent	0.9
TABLET	0	0.2		0.05
CAR	0	0.7		0.05

Cost Function

$$L(y, \hat{y}) = -\sum_{i}^{C} y \log(\hat{y})$$

Para 1 dato de entrenamiento

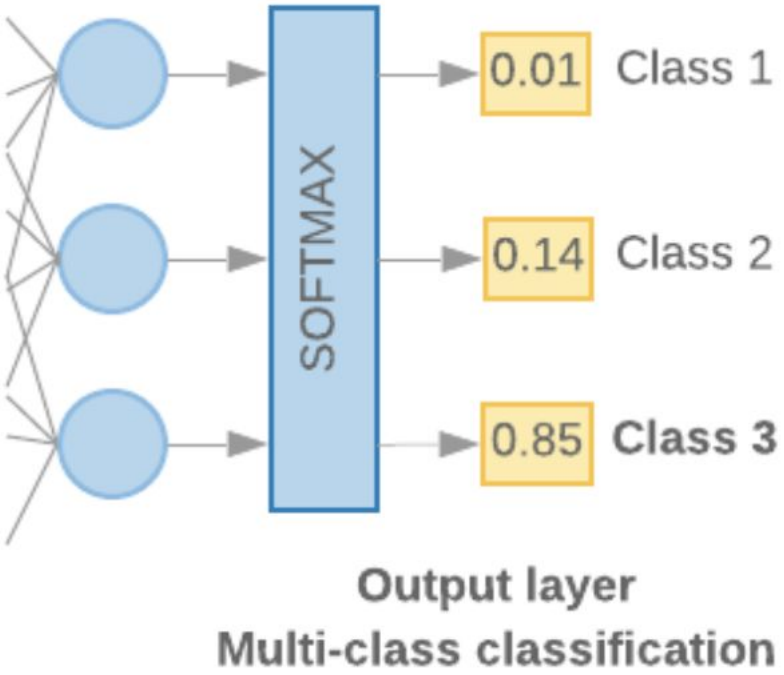
$$J(\theta) = \frac{1}{m} \sum_{i}^{m} L(y_i, \hat{y}_i)$$

Para todos los datos de entrenamiento

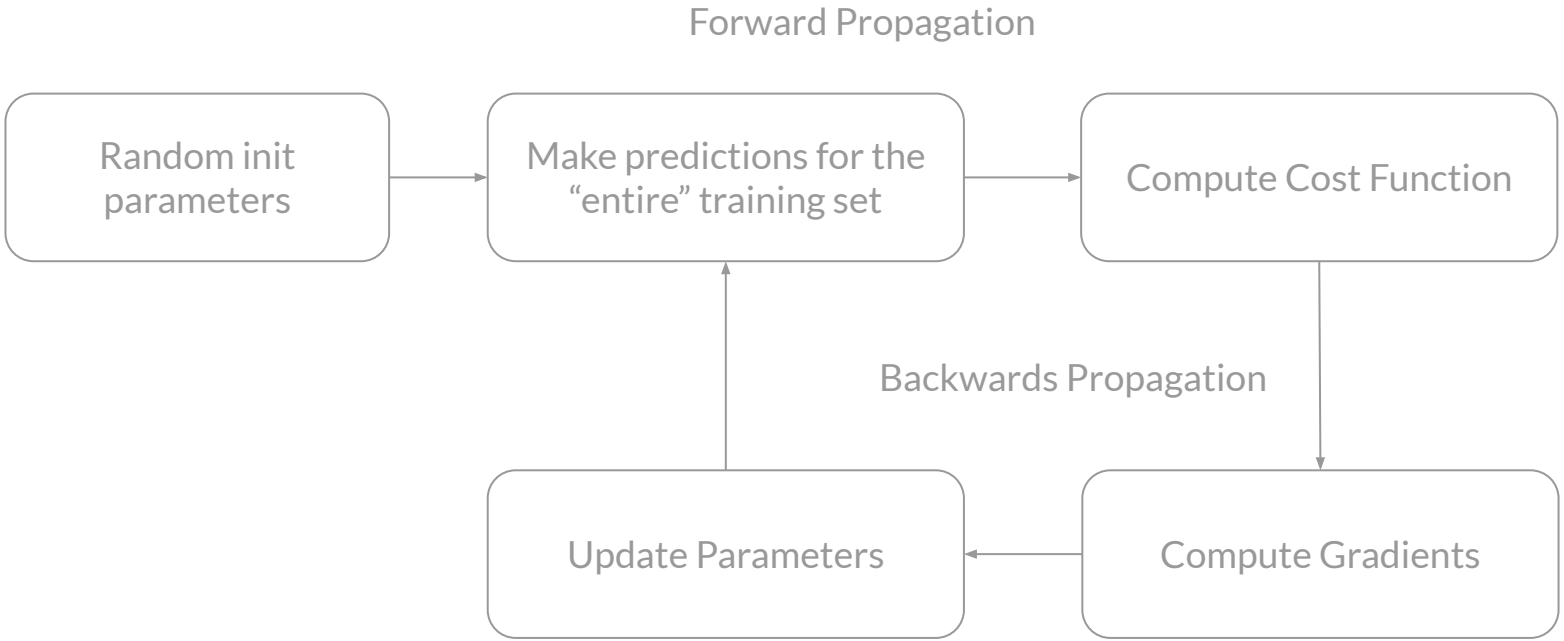
Softmax Activation

Softmax: Squashes a vector in the range of (0,1) and sum(vector) = 1

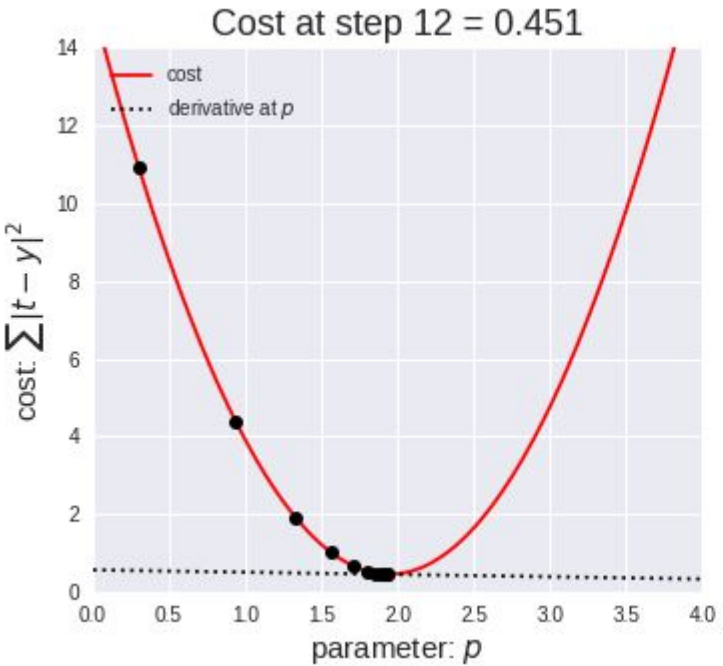
$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}}$$

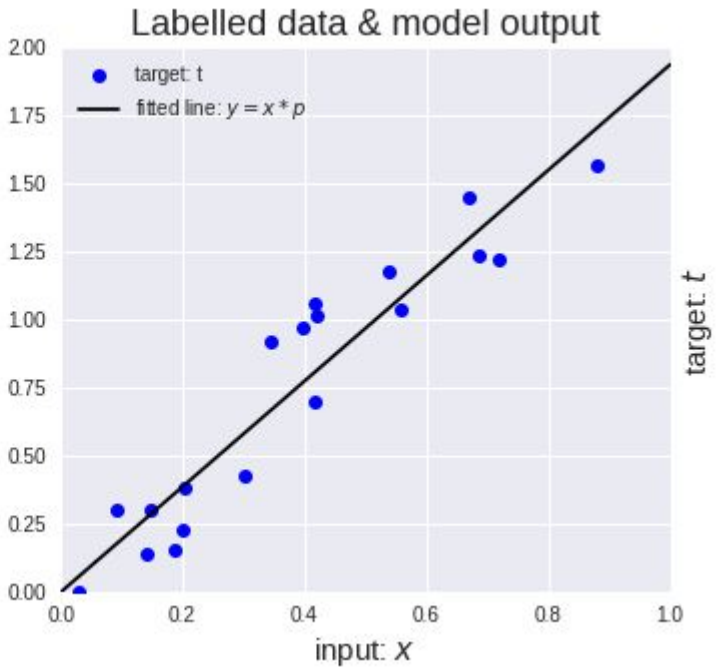


Gradient Descent

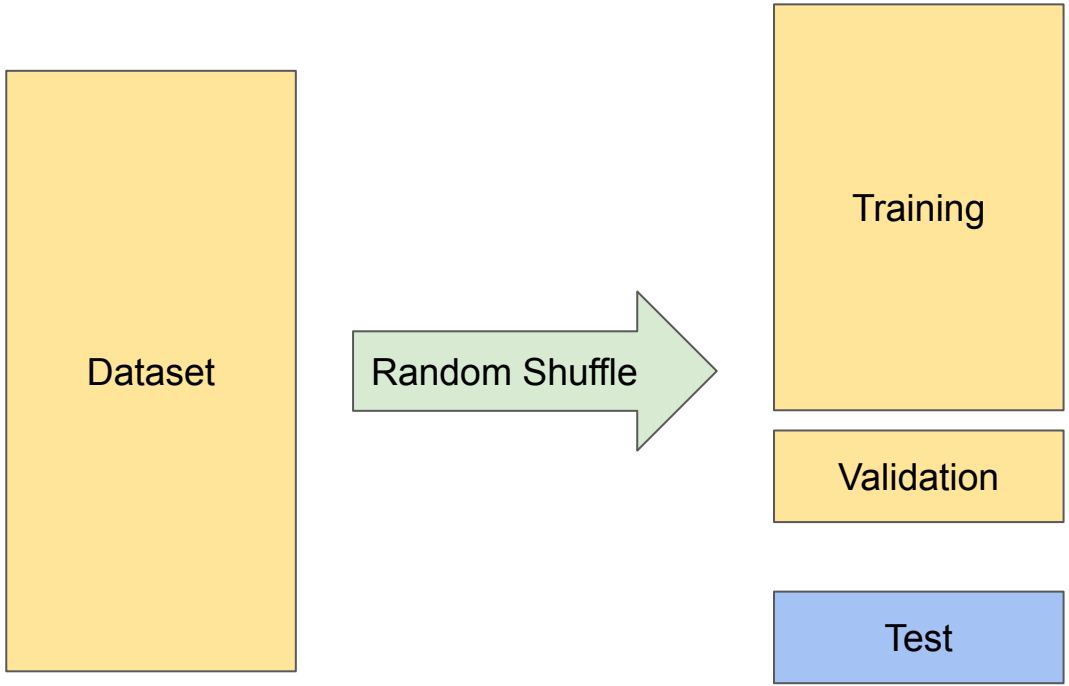


Gradient Descent





Split de Datasets

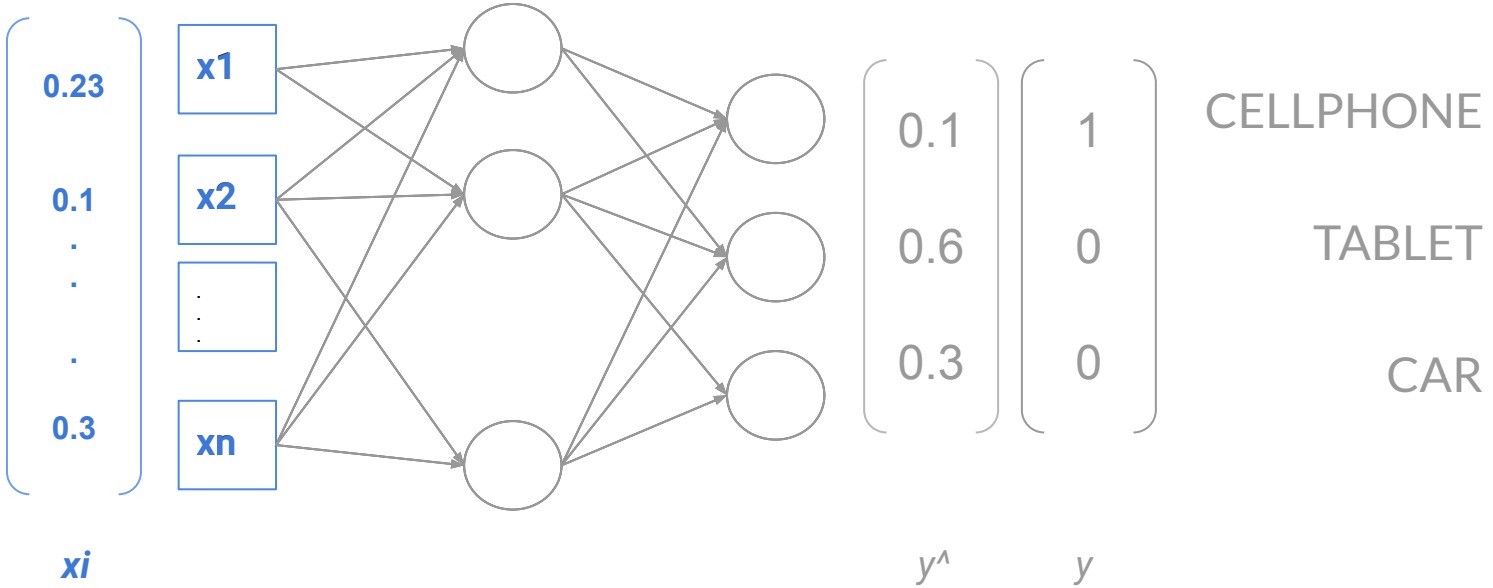


Agenda

- Intro / Problema
- Conceptos generales
  - AI vs ML vs DL
  - ¿Por que Deep Learning?
  - Tipos de algoritmos
- Redes Neuronales
  - Cost & Loss Function
  - Gradient Descent
- **Intro Natural Language Processing**
  - **Tokenization**
  - **Vectorization**
- Workflow de trabajo
- Intro a Python, Numpy y Pandas
- Entrenar una NN

Preparar los datos

"Apple Iphone X 64gb"



Tokenization

“Oferta!!! Cómoda 4 Cajones Blanco Miel Dormitorios.”

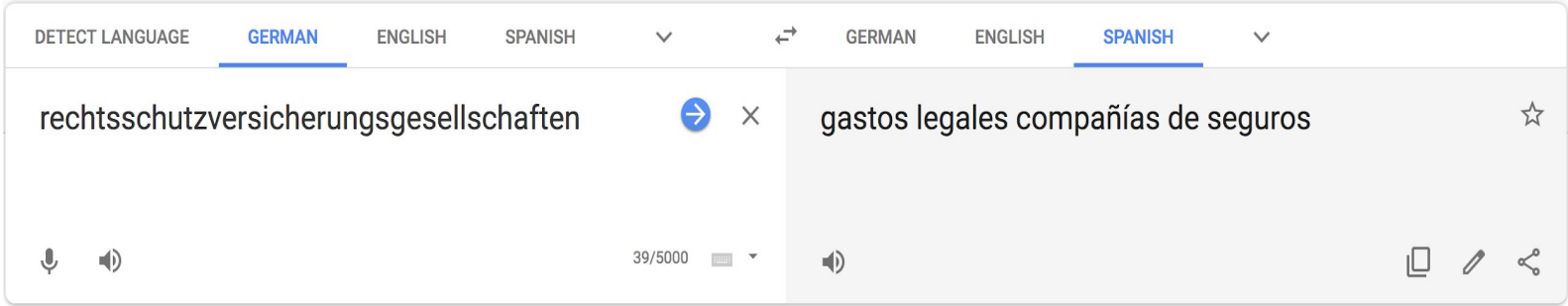
- ¿Que es el texto?
  - Secuencia de caracteres
  - **Secuencia de palabras**
  - Secuencia de frases
  - Secuencia de oraciones
  - Secuencia de párrafos

Tokenization

“Oferta!!! Cómoda 4 Cajones Blanco Miel Dormitorios.”

Tokenization

oferta comoda 4 cajones blanco miel dormitorios



Token normalization

- Stemming
  - eliminar sufijos de las palabras para retornar una nueva forma de la palabra (stem).
  - ej: perros, perras, perritos -> perr
- Lemmatization
  - usando un vocabulario retornar el significado de diccionario de la palabra (lemma).
  - ej: peces -> pez

Vectorization

Bag of Words

Vocabulario →

Oferta! Celular **apple** iphone 8 64gb + **funda** **apple** de regalo
Funda para samsung galaxy oferta
Tablet apple ipad con funda

...	apple	funda	samsung	....	celular	...
...	**2**	**1**	0	....	1	...
...	0	1	1	....	0	...
...	1	1	0	....	0	...

Vectorization

N-grams

Problema: Vocabulario muy extenso (muchas features)

celular apple iphone 8
funda samsung galaxy oferta
tablet apple ipad con funda

Celular apple	apple	Apple iphone	....	funda
1	1	1	....	0
0	0	0	....	1
0	1	0	....	1

Vectorization

- **TF-IDF**: Term frequency - Inverse Document Frequency
- **TF**: cantidad de ocurrencias del término **t** en el documento **d**
- **IDF**: importancia del término **t** en el corpus (todos los documentos)

Ejemplo: “Vendo iphone de primera de 64gb de color negro”

TF-IDF(“iphone”) = TF(“iphone”) * IDF(“iphone”) = (1/9) * (número grande)
TF-IDF(“de”) = TF(“de”) * IDF(“de”) = (3/9) * (número pequeño)

$$\operatorname{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Agenda

- Intro / Problema
- Conceptos generales
  - AI vs ML vs DL
  - ¿Por que Deep Learning?
  - Tipos de algoritmos
- Redes Neuronales
  - Cost & Loss Function
  - Gradient Descent
- Intro Natural Language Processing
  - Tokenization
  - Vectorization
- **Workflow de trabajo**
- Intro a Python, Numpy y Pandas
- Entrenar una NN

Workflow

https://developers.google.com/machine-learning/guides/text-classification/

Agenda

- Intro / Problema
- Conceptos generales
  - AI vs ML vs DL
  - ¿Por que Deep Learning?
  - Tipos de algoritmos
- Redes Neuronales
  - Cost & Loss Function
  - Gradient Descent
- Intro Natural Language Processing
  - Tokenization
  - Vectorization
- Workflow de trabajo
- **Intro a Python, Numpy y Pandas**
- **Entrenar una NN**

Practico

Repositorio en Github:

http://bit.ly/workshop-categorizacion-productos

Notebooks:

1. Intro Python, Numpy, Pandas
2. Train a NN for Products Categorization
    a. Gather & Explore Data (solución)
    b. Prepare the data for training (solución)
    c. Build, train & evaluate model (solución)

Next steps

#MeLiDataChallenge!!!

- Workshop de fast.ai con baseline (incluye grabación y notebooks)
- Inscribirse y participar para ganar entradas para Khipu!!

Toda la info en:

- https://ml-challenge.mercadolibre.com/

Muchas gracias