Apunte Organicacion de datos

lcondoriz

May 2023

Índice

1. Introducción

Definición 1.1. (Machine Learning) El aprendizaje automático es una parte de la inteligencia artificial y el subcampo de la ciencia de datos. Es una tecnología en crecimiento que permite que las máquinas aprendan de datos anteriores y realicen una tarea determinada automáticamente.

Machine Leaning permite que las computadoras aprendan de las experiencias pasadas por sí mismas, utiliza métodos estadísticos para mejorar el rendimiento y predecir la salida sin ser programado explícitamente. [6]

Definición 1.2. (Data Science) Data Science un campo de estudio profundo de los datos que incluye extraer información útil de los datos y procesar esa información utilizando diferentes herramientas, modelos estadísticos y algoritmos de aprendizaje automático

Es un campo interdisciplinario que utiliza técnicas de análisis de datos, estadística y aprendizaje automático para extraer conocimientos y crear modelos predictivos a partir de grandes conjuntos de datos.

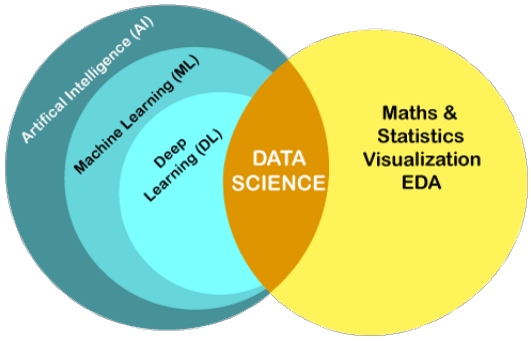

Figura 1: Data Science y Machine Learning

1.1. Variables

En el contexto del aprendizaje supervisado, se trabaja con *variables independientes* (también conocidas como características, predictores o variables de entrada) y *variables dependientes* (también llamadas variables objetivo o target o variables de salida). Las variables independientes son las que se utilizan para hacer predicciones o estimaciones, mientras que las variables dependientes son los resultados que se intentan predecir o modelar. Por ejemplo, en un problema de predicción de precios de viviendas, las características como el número de habitaciones, la ubicación y el tamaño de la propiedad serían las variables independientes, mientras que el precio de venta sería la variable dependiente.

- Variables Independientes (entradas)

- Cualitativas
  - Texto
    - ◇ Nominales (categorías, Por ejemplo: países, sexo)
    - ◇ Ordinales (poco, mucho, muchísimo, Por ejemplo: nivel de tabaquismos)
  - Números
    - ◇ Nominales
    - ◇ Ordinales
- Cuantitativas: cuando hablamos de cantidad
  - Discretas: Por ejemplo año, mes, edad, etc.
  - Continuas: Por ejemplo altura, peso, etc.

- Variables Dependientes (salidas, categorías)

Definición 1.3. (variables cualitativas) Son aquellas que describen características o cualidades y no pueden ser medidas en términos numéricos. Las variables **cuantitativas**, por otro lado, son aquellas que pueden ser medidas en términos numéricos y tienen valores numéricos.

Definición 1.4. variables nominales Son aquellas que no tienen un orden natural, como el género o el color de ojos. Las **variables ordinales** son aquellas que tienen un orden natural, como el nivel de educación (primaria, secundaria, universidad), nivel de tabaquismos: Clasificamos como leve (1), moderado (2), nivel medio (3), importante (4) y muy importante (5).

Variables y tipos de problemas (video 02a min 9:00)

1. Si la variable dependiente es **cualitativa**, el tipo de problema es de **clasificación**.
2. Si la variable dependiente es **cuantitativa**, el problema es de **regresión**. Por ejemplo si quiero predecir el precio de una propiedad.
3. Si **NO hay variable** dependientes, el problema es agrupamiento.

- **Outliers:** Valores atípicos, pueden ser errores o un dato que se sale de la norma.
- **Correlación:**
  - **Positiva:** Cuando una variable aumenta la otra también.
  - **Negativa:** Cuando una variable aumenta la otra disminuye.
  - **Sin correlación:** Cuando una variable aumenta la otra no cambia.
- **Varianza:** Es la medida de dispersión de una variable respecto a su media. Si la varianza es alta, los datos están muy dispersos, mientras que si la varianza es baja, los datos están muy agrupados.

- **Covarianza:** Es una medida de la relación lineal entre dos variables aleatorias. Indica cómo varían conjuntamente dos variables aleatorias respecto a sus medias. Si la covarianza es positiva, las variables aumentan o disminuyen conjuntamente, mientras que si la covarianza es negativa, una variable aumenta mientras la otra disminuye.

Figura 2: Metodología de Machine Learning.

1.2. Formatos de datos y herramientas

Ver Video clase.

1. **CSV:** Comma Separated Values. Es un formato de texto plano que se utiliza para almacenar datos tabulares. Cada registro se almacena en una línea y los campos se separan por comas.
2. **JSON:** JavaScript Object Notation. Es un formato de texto plano que se utiliza para almacenar datos estructurados. Se utiliza principalmente para transmitir datos entre un servidor y una aplicación web.
3. **CSR:** Compressed Sparse Row. Es un formato de matriz dispersa (con gran cantidad de ceros) que se utiliza para almacenar matrices dispersas. Se utiliza principalmente para almacenar matrices dispersas en memoria.

2. Introducción a probabilidad y estadística

2.1. Esperanza

Definición 2.1. (variable discreta) Sea X una variable discreta con función de probabilidad $P_X(x)$. La esperanza de X, denotada por $E[X]$, se define por

$$E[X] = \sum_{i=1}^{n} x_i \cdot P(X = x_i) \tag{1}$$

Definición 2.2. (variable absolutamente continua) Sea X una variable absolutamente continua con función de densidad $f_X(x)$. La esperanza de X, denotada por $E[X]$, se define por

$$E[X] = \int_{-\infty}^{\infty} x \cdot f_X(x) \cdot dx \tag{2}$$

2.2. Varianza

La varianza da una medida de cuánto varían los valores de una muestra obtenida al realizar experimentos aleatorios.

Los x_i son los valores obtenidos. Mientras que $\bar{X}$ es el promedio de los resultados del experimento. N es la cantidad de resultados obtenidos. *Video Youtube min 5:40* [2]

$$Var(X) = \frac{1}{N} \cdot \sum_{i=1}^{N} (x_i - \bar{X})^2 \tag{3}$$

Cuandio la varianza es baja, los valores de la muestra se agrupn cerca de su valor esperado. Cuando la varianza es alta, los valores de la muestra se dispersan más.

Definición 2.3. (Varianza). Sea X una variable aleatoria con esperanza finita. La varianza de X se define por

$$Var(X) = E[(X - E[X])^2] \tag{4}$$

Definición 2.4. (Desviación estándar). La desviación estándar de X se define por

$$\sigma_X = \sqrt{Var(X)} \tag{5}$$

2.3. Covarianza

Definición 2.5. (Covarianza) La covarianza es una medida de cómo varían conjuntamente dos variables aleatorias.

Sean X e Y dos variables aleatorias de varianzas finitas definidas sobre el mismo espacio de probabilidad (Ω, A, P). La covarianza de X e Y se define por

$$\begin{aligned} Cov(X,Y) &= E[(X - E[X])(Y - E[Y])] \\ &= E[XY] - E[X]E[Y] \end{aligned} \tag{6}$$

Interpretación de la covarianza:

- Si $S_{xy} > 0$ hay dependencia directa (positiva), es decir, a grandes valores de X corresponden grandes valores de Y.
- Si $S_{xy} < 0$ hay dependencia inversa (negativa), es decir, a grandes valores de X corresponden pequeños valores de Y.

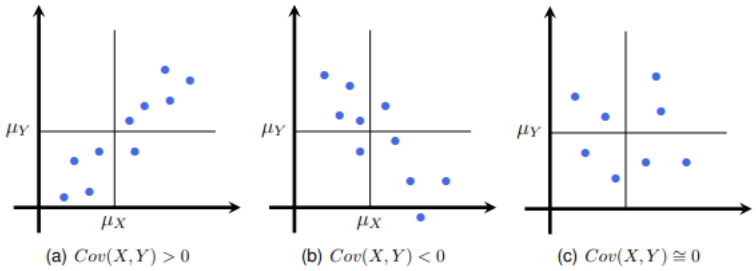

Figura 3: Covarianza

- Si $S_{xy} = 0$ no hay dependencia lineal entre X e Y.

Definición 2.6. (Covarianza muestral)

$$S_{xy} = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \tag{7}$$

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \qquad \bar{y} = \frac{1}{n} \cdot \sum_{i=1}^{n} y_i \tag{8}$$

2.4. Correlación

Es una medida de relación lineal entre dos variables cualitativas continuas. Con esta medida, se logra determinar si las variables varían conjuntamente.

Es una medida normalizada, su valor va de -1 a 1. El caso en el que la correlación es 0, indica que no existe relación lineal entre las variables.

En caso de que se 1, se trata de correlación perfecta en sentido positivo. En caso de que sea -1, se trata de correlación perfecta en sentido negativo.

El sentido positivo, indica que varían en el mismo sentido. El sentido negativo, indica que varían en sentidos opuestos.

Definición 2.7. (Coeficiente de correlación)

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sigma_X \cdot \sigma_Y} \tag{9}$$

Definición 2.8. (Coeficiente de Correlación de Pearson)

$$\begin{aligned} \rho_{xy} &= \frac{S_{xy}}{S_x \cdot S_y} \\ &= \frac{\sum_{i=1}^{n}(x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2 \cdot \sum_{i=1}^{n}(y_i - \bar{y})^2} \end{aligned} \tag{10}$$

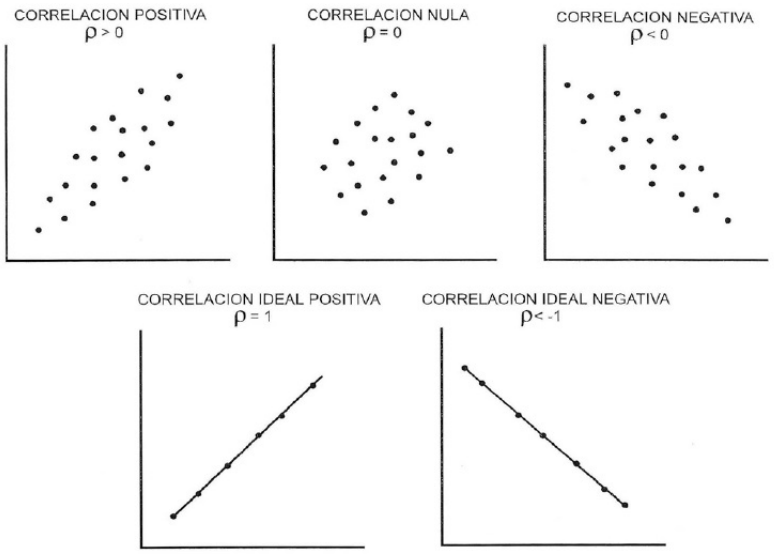

Figura 4: Coeficiente de Correlación de Pearson

Que las variables esten corelacionadas no implica que haya una relación de causalidad entre las mismas. Por ejemplo, si se toma la temperatura y la cantidad de helados vendidos, ambas variables están correlacionadas, pero no hay una relación de causalidad.

2.5. Ejemplos

Ejemplo 2.1. Ejemplo de como calcular datos variables aleatorias. Video Youtube [4]

X	Y
2	1
3	2
5	2
6	3

Cuadro 1: Datos

- **Esperanza**

$$\begin{aligned} E[X] &= \frac{1}{4} \cdot (2+3+5+6) = 4 \\ E[Y] &= \frac{1}{4} \cdot (1+2+2+3) = 2 \end{aligned} \tag{11}$$

- **Varianza**

$$\begin{aligned} Var(X) &= \frac{1}{4} \cdot [(2-4)^2 + (3-4)^2 + (5-4)^2 + (6-4)^2] = 2{,}5 \\ Var(Y) &= \frac{1}{4} \cdot ((1-2)^2 + (2-2)^2 + (2-2)^2 + (3-2)^2) = 0{,}5 \end{aligned} \tag{12}$$

- **Covarianza**

$$\begin{aligned} Cov(X,Y) &= \frac{1}{4} \cdot [(2-4)(1-2) + (3-4)(2-2) \\ &\quad + (5-4)(2-2) + (6-4)(3-2)] \\ &= \frac{1}{4} \cdot (2+0+0+2) \\ &= 1 \end{aligned} \tag{13}$$

- **Correlación**

$$\begin{aligned} \rho_{xy} &= \frac{Cov(X,Y)}{\sigma_X \cdot \sigma_Y} \\ &= \frac{1}{\sqrt{2{,}5} \cdot \sqrt{0{,}5}} \\ &= \sqrt{\frac{4}{5}} = 0{,}8944 \end{aligned} \tag{14}$$

3. NLP

El procesamiento de lenguaje natural (NLP - natural language processing) es un campo de las ciencias de la computación, de la inteligencia artificial y de la lingüística que estudia las interacciones entre las computadoras y el lenguaje humano.

3.1. Similitud coseno

La similitud coseno (Cosine similarity) es una medida de la similitud existente entre dos vectores en un espacio que posee un producto interior con el que se evalúa el valor del coseno del ángulo comprendido entre ellos. El valor de esta métrica se encuentra entre -1 y 1, es decir en el intervalo cerrado [-1,1].

Definición 3.1. Para vectores u y v, en un espacio euclídeo, distintos de cero en $\mathbb{R}^n$,

$$\cos(\theta) = \frac{u \cdot v}{\|u\|\|v\|} \tag{15}$$

- Dos vectores u y v en $\mathbb{R}^n$ son mutuamente ortogonales si y solo si $u \cdot v = 0$.
- Dos vectores u y v en $\mathbb{R}^n$ son paralelos si y solo si $u \cdot v = \|u\|\|v\|$.

Entonces un documento es similar a otro, si y solo si $\cos(\theta) \overset{tiende}{\longrightarrow} 1$.

3.2. Tokenización

La tokenización es el proceso de dividir una cadena de caracteres en unidades más pequeñas, llamadas tokens.

¿Por que tokenizar, y no usar simplente hacer un split? Porque la tokenización `nltk` es un proceso más complejo, mientrar que `split` solo separa de acuerdo a la key que se le pasa como parametro.

3.3. Bag of Words (BOW)

El modelo bolsa de palabras (del inglés, Bag of Words) es un método que se utiliza en el procesado del lenguaje para representar documentos ignorando el orden de las palabras. Con este modelo podemos tener una representación de cada documento, en función de las palabras que este contiene.

3.4. Term Frequency (Count Vectorizer)

Hace lo mismo que BOW, pero además cuenta la cantidad de veces que aparece cada palabra en cada documento.

3.5. Term Frequency x Inverse Document Frequency (TF-IDF)

Definición 3.2.

$$\text{TF-IDF}(palabra) = \log\left(\frac{N+1}{frecuencia}\right) \tag{16}$$

- N Cantidad total de documentos.
- $frecuencia$ Cantidad de documentos en los que aparece la palabra.

Entonces, si una palabra aparece en todos los documentos, su valor de TF-IDF será 0.

3.6. Normalización

3.6.1. Stemming

Es un proceso en donde se elimina la última parte de las palabras por algún proceso (hay steammers de varios tipos). El objetivo es llegar a la raíz (stem) de la palabra por medio de su prefijo.

- caballo, caballería, caballero → caball
- biblioteca, bibliotecario, bibliotecología → bibliotec
- canto, cantar, cantante → cant
- cantidad, cantar, cantante → cant

3.6.2. Lemmatization

Devuelve la base de la pala (lemma), muchas veces por medio de un **diccionario de lemas** para cada palabra.

- caballo, caballería, caballero → caballo
- biblioteca, bibliotecario → biblioteca
- canto, canta, cantamos, cantan → canto
- cantidad, cantidades → cantidad

3.6.3. Stopwords

Las stopwords o "palabras vacías"son palabras demaciado comunes o sin significado como artículos, pronombres, preposiciones, etc. Estas palabras no aportan información relevante para el análisis de texto, por lo que se suelen eliminar.

A veces (por ejemplo usando BOW o TF-IDF) queremos removerlas.

4. Big data y Almacenamiento Distribuido

4.1. Big Data

BIg Data datos que no pueden ser procesados por métodos tradicionales.
Se caracterizan por las 5 *V's*:

- Volumen: cantidad de datos.
- Velocidad: velocidad a la que se generan los datos.
- Variedad: variedad de datos.
- Veracidad: calidad de los datos.
- Valencia: valor de los datos.

4.1.1. Volumen

El impedimento de las computadoras a manejar grandes catidades de datos.
Cluster: un conjunto de computadoras que trabajan enconjunto y pueden ser vistas como un sistema único.

4.1.2. Velocidad

La velocidad con la cual se generan los mismos. Para procesar los datos se necesita que el tiempo de procesamiento sea menor al tiempo de generación de los datos, sino estos se aculan y el algoritmo puede descartar una gran volumen de datos.

4.2. Clase 6 - Spark - 1 - Spark

Definición 4.1. (Map-Reduce) Procesamiento distribuido de datos utilizando un cluster.

- Modelo de programación para procesar grandes volúmenes de datos.
- Sugue de la necesidad de procesar grandes volúmenes de datos de forma escalable.
- El usuario especifica una funcion una función **map** que procesa un par clave/valor para generar un conjunto de pares clave/valor intermedios.
- Se debe especificar una función **reduce** que combina todos los valores asociados a una misma clave intermedia.

Definición 4.2. (Map)

- Transfroma nuestros datos.
- Debe ser aplicada a cada dato de nuestro ser.
- Puede ser paralelizada y distribuirse entre las distintas máquinas de un cluster.

Algunas diferencias dependientes de la implementación:

- **Hadoop:** $Map(k, v) \rightarrow list(k2, v2)$
- **spark:** $Map(r) \rightarrow list(r')$

Definición 4.3. (Reduce)

- Combina los resultados del map.
- Es necesario procesar los datos de todas las máquinas del cluster.
- Reduce locales en paralelo y reduce entre máquinas mediente esta de shuffle & sort.

Algunas diferencias dependientes de la implementación:
Hadoop: $RecudeByKey((k, v), f) \rightarrow list(k, v)$

- El sistema agrupa todos los registros para los cuales la clave es la misma.
- Requiere que todos los registros de igual clave estén en la misma máquina que ejecute el reduce: Shuffle & Sort.

Spark:

- La función reduce toma dos valores para dar como resultado la combinación de ambos.

- El resultado de un reduce estre dos registros es un input del siguiente reduce.
- Opereaciones **conmutativas** y **asociativas** de modo de poder ejecutarse distribuidas.

Definición 4.4. (Cluster) Conjunto de computadoras que trabajan juntas y pueden ser vistas como un sistema único.

Definición 4.5. (Almacenamiento distribuido)

- FIleSystem Distribuido (DFS)
- Encargado de gastionar cómo y dónde guardar la información en una computadora, y cómo poder consultarla.
- Almacenar grandes volúmenes de datos en multiples equipos.
- Replicación de datos para tolerancia a fallos.
- Tolerancia a fallos.
- Alta disponibilidad (seguir funcionando aunque un nodo falle).
- Relativo a bajo costo.

Ejemplos de SD:

- GDS (Google File System)
- HDFS (Hadoop Distributed File System)
- CEPH (Ceph File System)
- S3 (Amazon Simple Storage Service)

4.3. Clase 6 - Spark - 2 - Spark

4.4. YouTube

MapReduce vs Spark link:

- MapReduce: procesamiento de datos distribuido en disco.
- Spark: procesamiento de datos distribuido en memoria.
- Spark es 100 más rápido que MapReduce.
- Spark es más fácil de usar que MapReduce.
- Spark es más flexible que MapReduce.
- Spark es más costoso que MapReduce.

- Spark es más complejo que MapReduce.
- Spark es más nuevo que MapReduce.

Spark trabaja con RDDs (Resilient Distributed Datasets) que son colecciones de objetos que se pueden dividir en particiones y distribuir entre los nodos del cluster. Los RDDs son inmutables y tolerantes a fallos.

Los RDDs:

- Spark utiliza RDDs para almacenar datos.
- Son como tuplas o listas.
- Datos Distribuidos.
- Tolerantes a fallos.
- Opereaciones paralelizables.
- Habilidad para datos de multiples fuentes.

Opereaciones:

- Transformaciones: crean un nuevo RDD a partir de uno existente: map, filter, flatMap, sample, union, intersection, distinct, groupByKey, reduceByKey, sortByKey, join, cogroup, cartesian.
  - RDD.filter(): Aplica una función y devuelve los elementos que son verdaderos, parecido a filter de python.
  - RDD.map(): Transforma cada elemento sin cambiar el número de elementos, parecido a pandas.apply().
    Estraer la primer letra de una lista de nombres.
  - RDD.flatMap(): Transforma cada elemento y cambia el número de elementos.
    Convertir el corpus de un text a una lista de palabras.
- Acciones: devuelven un valor al programa driver después de ejecutar un cálculo en un RDD: reduce, collect, count, first, take, takeSample, takeOrdered, saveAsTextFile, saveAsSequenceFile, saveAsObjectFile, countByKey, foreach.
  - Collect: Devuelve todos los elementos del RDD como una matriz.
  - Count: Devuelve el número de elementos en el RDD.
  - First: Devuelve el primer elemento del RDD.
  - Take: Devuelve un número especificado de elementos del RDD.
- Lazy evaluation: las transformaciones no se ejecutan hasta que no se ejecuta una acción.

5. Procesamiento de datos.

Limpieza de datos.

La limpieza de datos es un paso fundamental en el preprocesamiento de datos en el machine learning. Este proceso implica identificar y corregir o eliminar datos incorrectos, incompletos, duplicados o inconsistentes en un conjunto de datos. Aquí se explica más detalladamente la limpieza de datos:

1. **Identificar datos incorrectos o erróneos:** Durante la recopilación de datos, es posible que se hayan introducido errores o que los datos estén mal formateados. En este paso, se deben identificar y corregir los errores obvios, como valores que están fuera de rango o que no se ajustan al formato esperado. Por ejemplo, si se tiene un conjunto de datos que contiene información sobre el género de los empleados de una empresa, los datos incorrectos serían los que no son ni "masculinos"ni "femeninos".

2. **Tratamiento de datos faltantes:** Los conjuntos de datos a menudo contienen valores faltantes, ya sea porque no se recopilaron o porque se perdieron durante el proceso de almacenamiento o transferencia. Los valores faltantes pueden afectar el rendimiento de los modelos de machine learning, por lo que es importante abordarlos. Esto implica decidir si se deben eliminar las instancias con valores faltantes, imputar los valores faltantes utilizando técnicas de imputación (como el promedio, la mediana o modelos más avanzados), o considerarlos como una categoría separada.

3. **Eliminación de datos duplicados:** En algunos casos, puede haber instancias duplicadas en el conjunto de datos. Estas instancias duplicadas no aportan información adicional y pueden sesgar los resultados. Por lo tanto, es esencial identificar y eliminar las instancias duplicadas para garantizar la integridad y la calidad de los datos.

4. **Resolución de inconsistencias y valores atípicos (outliers):** Los valores atípicos son observaciones que difieren significativamente del resto de los datos. Pueden ser resultado de errores de medición o indicar situaciones inusuales. Es importante evaluar si los valores atípicos deben ser corregidos, eliminados o si contienen información relevante y deben mantenerse.

5. **Normalización y estandarización:** Los datos pueden tener diferentes escalas y rangos, lo que puede afectar el rendimiento de algunos algoritmos de machine learning. En este paso, se pueden aplicar técnicas de normalización o estandarización para asegurar que los datos tengan una distribución más uniforme y comparable

6. **Manejo de datos desbalanceados:** En algunos problemas de clasificación, puede haber una falta de equilibrio entre las clases objetivo, lo que significa que una clase puede tener muchos más ejemplos que las demás.

En estos casos, se deben utilizar técnicas de muestreo o ponderación para abordar el desequilibrio y evitar que el modelo se sesgue hacia la clase mayoritaria

5.1. Transformación y normalización de datos

La transformación y normalización de datos son técnicas utilizadas en el preprocesamiento de datos en machine learning para ajustar los datos a una escala o distribución específica. Estos procesos son importantes para garantizar que los datos sean adecuados para su uso en algoritmos de machine learning y que no se vean afectados por diferencias en las unidades o escalas de las características. A continuación, se explica con más detalle la transformación y normalización de datos:

6. Graficos Python

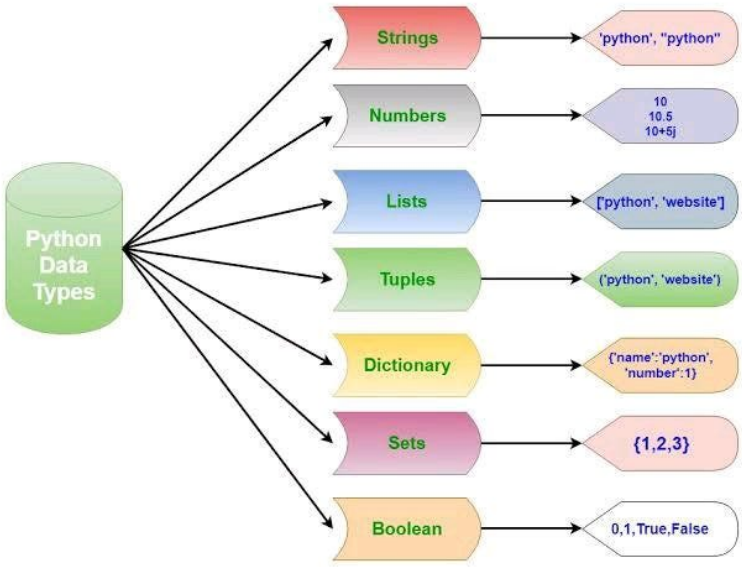

Figura 5: Tipos de variables en Python

Tipos de gráficos:

1. torta: plt.pie
2. barras: plt.bar
3. lineas: plt.plot
4. scatterplot: plt.scatter o sns.scatterplot

6.1. Groupy

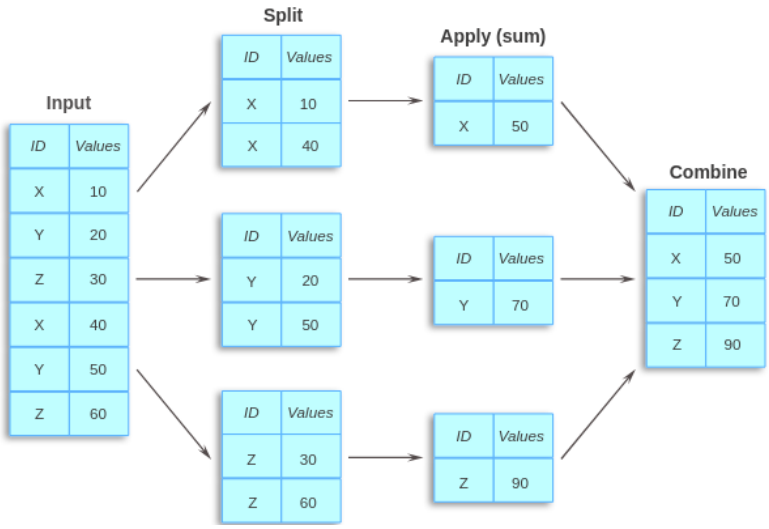

Figura 6: Groupy

Ver video [1]

7. Machine Learning.

Classical Machine Learning

1. Supercised Learning
   a) Classification:
      1) Regresión logística.
      2) K-Nearest Neighbors (k-vecinos más cercanos) (k-NN).
      3) Árboles de decisión.
      4) Bosques aleatorios.
      5) Máquinas de vectores de soporte (SVM)
      6) Naive Bayes.
   b) Regression: algoritmos y técnicas: regresión lineal, la regresión polinómica, la regresión de bosques aleatorios
2. Unsupervised Learning
   a) Clustering
   b) Association
   c) Dimensionality Reduction

Aprendizaje supervisado: Tenemos datos de entrenamiento con una salida esperada. Validación de resultados. Datos de entrada y salida etiquetados durante la fase de entrenamiento del ciclo de vida del machine learning.

Aprendizaje No Supervisado: No tenemos datos de salida sólo de entrada Cambiar representación de los datos. Facilitar entendimiento. Es el entrenamiento de modelos de datos sin procesar y sin etiquetar.

8. Supercised Learning

El Supercised Learning se puede clasificar en dos grandes grupos:

- Supercised Learning
  - Classification
    - Regresión logística
    - K-Nearest Neighbors (KNN)
    - Decision Trees
    - Random Forest
  - Regression
    - Linear Regression

Existen métodos y algoritmos que se pueden usar en ambos problemas:

Supervised Learning	**Classification**	**Regression**
Logistic Regression	✓	
k-Nearest Neighbors (KNN)	✓	✓
Decision Trees	✓	✓
Random Forest	✓	✓
Support vector machine (SVM)	✓	
Linear Regression		✓

Cuadro 2: Cuadros

Métricas Clasificación:

- Precisión.
- Recall (exhaustividad).
- Valor-F (F_1)

Ver video 03b min 40. Matriz de confucion.

8.1. Classification

En el contexto del aprendizaje automático supervisado, la variable predictora (también conocida como variable independiente o característica) es una variable

que se utiliza para predecir el valor de la variable objetivo (también conocida como variable dependiente o target). Las variables predictoras son los datos de entrada al modelo y la variable objetivo es el resultado que se desea predecir.

8.1.1. Regresión logística.

La regresión logística es un tipo de algoritmo de aprendizaje supervisado que se utiliza principalmente para resolver tareas de "clasificación" binarias, aunque contiene la palabra "regresión". Se lo utiliza para predecir el resultado de una variable categórica (una variable que puede adoptar un número limitado de categorías) en función de las **variables predictoras** o independientes. Un ejemplo de clasificación podría ser la detección de spam: un programa de aprendizaje automático puede aprender a marcar el spam después de recibir ejemplos de correos electrónicos spam marcados (**variable objetivo** "target" sería una variable binaria que indica si un correo electrónico es spam o no) por los usuarios y ejemplos de correos electrónicos regulares no spam (también llamados "ham"). Ejemplo Python [5].

Tambien existe la regresion logistica multinomial, que es una generalizacion de la regresion logistica. En este caso la variable objetivo puede tomar mas de dos valores. Ver regresion_logistica_02.ipynb

8.1.2. K-Nearest Neighbors (KNN).

Se utiliza para clasificación y regresión. k vecinos más cercanos es un método de clasificación no paramétrico. Video de youtube [7].
Se utiliza principalmente para la clasificación de datos no lineales y para resolver problemas de clasificación en los que los datos son muy complejos y desestructurados.

Resumen:

- Es sensible a conjuntos de datos no balanceados.
- Es muy sensible aoutliers.
- La normalización de los datos de entrenamiento puede mejorar drásticamente su precisión
- Si se aplica en un comjunto de datos desbalanceado, puede ser que el algoritmo siempre prediga la clase mayoritaria.

8.1.3. Decision Trees

Los árboles de decisión son modelos ampliamente utilizados para tareas de clasificación y regresión. Esencialmente, aprenden una jerarquía de preguntas if/else, que conducen a una decisión. Estas preguntas son similares a las preguntas que podría hacer en un juego de 20 preguntas. Imagina que quieres

distinguir entre los siguientes cuatro animales: osos, halcones, pingüinos y delfines. Su objetivo es llegar a la respuesta correcta haciendo la menor cantidad posible de preguntas si/si no. Puede comenzar preguntando si el animal tiene plumas, una pregunta que reduce sus posibles animales a solo dos. Si la respuesta es "sí", puedes hacer otra pregunta que podría ayudarte a distinguir entre halcones y pingüinos. Por ejemplo, podría preguntar si el animal puede volar. Si el animal no tiene plumas, sus posibles opciones de animales son delfines y osos, y deberá hacer una pregunta para distinguir entre estos dos animales, por ejemplo, preguntar si el animal tiene aletas.

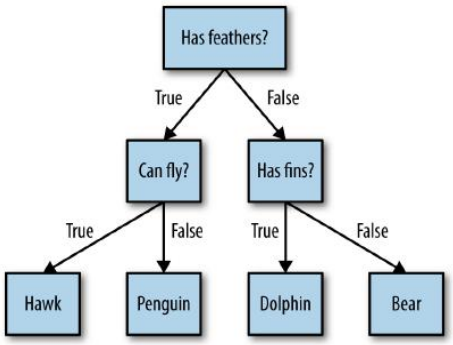

Figura 7: Ejemplo de un árbol de decisión.

Existen varios algoritmos ID3, C4.5 y CART. Pagina buena [3]:

• **ID3 - Iterative Dichotomiser 3:** Genera un árbol de decision a partir de un conjunto de ejemplos.
Este algoritmo usa las metricas de *entropía* y *ganancia de la información*. Cuando se usa la librería sklearm *sklearn.tree.DecisionTreeClassifier* en el parámetro *criterion* colocar *entropy*.

- Un nodo principal llamado raíz en la parte superior.
- Nodos terminales. como su nombre lo indica, son nodos donde termina el flujo y que ya no son raiz de ningún otro nodo. Estos nodos terminales deben contener una respuesta, o sea, la clasificación a que pertenece el objeto que ha conducido hasta él.
- Los demás nodos representan preguntas con respecto al valor de uno de los atributos.
- Las líneas nodos representan preguntas con respecto al valor de uno de sus atributos.
- Las líneas representa las posibles respuestas que los atributos pueden tomar.

Algoritmo básico

1. Calcular la entropía para todas las clases.
2. Calcular la entropía para cada valor posible de cada atributo.
3. Seleccionar el mejor atributo basado en la reducción de la entropía. usando el calculo de la ganancia de la información.
4. Iterar, para cada sub-nodo, Excluyendo el nodo raíz, que ya fue usado.

- **C4.5:** Utiliza la metrica *gini*.

1. Mitiga el sobreajuste por que emplea inherentemente el proceso de poda de un solo paso.
2. Funciona para datos discretos y continuos.
3. Es útil para datos incompletos.

Metricas:

- **Entropía:**
  La medida del desorden o la medida de la pureza. Básicamente, es la medida de la impureza o aleatoriedad de los datos.

  Para calcular la entropia de n clases se utiliza la fórmula:

$$H(S) = \sum_{i=1}^{n} -p_i \cdot \log_2(p_i) \tag{17}$$

  Dónde:

  - S: es una lista de valores posibles.
  - p_i: es la probabilidad de los valores.
  - i: cada uno de los valores.

  Importante:

  - Para una muestra homogénea la entropia es igual a cero 0. Si no existe aleatoriedad, es decir, una moneda cargada.
  - La máxima entropía viene dada por $\log_2(n)$, n son los posibles valores de salida. Si $n = 2$ (true o false) entonces, la máxima entropía es 1. O sea es la máxima incertidumbre, ejemplo moneda equilibrada.

- **Ganancia de la Información:** La ganancia de la información se aplica a cuantificar qué característica, de un conjunto de datos dados, proporciona la máxima información sobre la clasificación.

8.1.4. Random Forest - Bosques aleatorios.

Usa una técnica, o meta-algoritmo llamado Bootstrap aggregating. Se crean m tablas reducidas en atributos y para cada una de ellas entrenamos un árbol.

8.1.5. Support Vector Machines (SVM)

Es un algoritmo de aprendizaje supervisado que se puede utilizar para problemas de clasificación o regresión. El algoritmo SVM utiliza un hiperplano para separar los datos en clases. El hiperplano se selecciona de tal manera que maximiza la distancia entre los puntos de datos de las clases. El hiperplano se puede utilizar para clasificar nuevos puntos de datos.

Ventajas:

- Efectivo en espacios de alta dimensión.
- Efectivo en casos en que el número de dimensiones es mayor que el número de muestras.
- Utiliza un subconjunto de puntos de entrenamiento en la función de decisión (llamados vectores de soporte), por lo que también es eficiente en memoria.
- Versátil: se pueden especificar diferentes funciones del núcleo para la función de decisión. Se proporcionan núcleos comunes, pero también es posible especificar núcleos personalizados.

Desventajas:

- Si el número de características es mucho mayor que el número de muestras, evite el exceso de ajuste al elegir las funciones del núcleo y el término de regularización es crucial.
- Los SVM no proporcionan directamente estimaciones de probabilidad, estas se calculan utilizando una validación cruzada de cinco veces.

Parámetros:

- **C:** Parámetro de regularización. El parámetro C controla el comercio entre el ajuste de los datos de entrenamiento y la suavidad de la superficie de decisión. Un C alto significa que el clasificador intentará ajustar los datos de entrenamiento lo mejor posible, mientras que un C bajo significa que el clasificador buscará una superficie de decisión que esté lo más suave posible.
- **kernel:** Especifica el tipo de kernel que se utilizará en el algoritmo. Debe ser uno de 'lineal', 'poli', 'rbf', 'sigmoid', 'precomputed' o una llamada a un kernel personalizado.

- **degree:** Grado de la función del núcleo polinomial ('poly'). Ignorado por todos los demás núcleos.
- **gamma:** Coeficiente para 'rbf', 'poly' y 'sigmoide'. Si gamma es 'auto', entonces 1 / n_features se utilizará en su lugar.
- **coef0:** Término independiente en función del núcleo. Solo es significativo en 'poly' y 'sigmoide'.
- **probability:** Habilita la estimación de la probabilidad. Debe estar habilitado antes de llamar a fit, y se basa en una validación cruzada de cinco veces. Deshabilitarlo puede acelerar el cálculo cuando se usa svm en grandes conjuntos de datos.
- **shrinking:** Habilita o deshabilita el encogimiento heurístico de la función de decisión. Debe estar habilitado para el uso de la validación cruzada de probabilidad. Deshabilitarlo puede dar una pequeña ganancia de rendimiento.

Linealmente separable: es un conjunto de datos que se puede separar en dos grupos distintos de manera que no haya puntos de datos que se superpongan entre los dos grupos.

8.2. Regresión

Los modelos que existe para la regresión son:

- Regresión lineal y multiple: *sklearn.linear_model.Regresión Lineal*
- Regresión polinómica: *sklearn.preprocessing.PolynomialFeatures*

8.2.1. Regresión Lineal

La regresión lineal es un método estadístico que se utiliza para estudiar la relación entre una variable dependiente (también conocida como variable objetivo) y una o más variables independientes (también conocidas como variables predictoras). El objetivo de la regresión lineal es encontrar la línea (recta) que mejor se ajuste a los datos y pueda utilizarse para hacer predicciones.

$$y = a \cdot x + b \tag{18}$$

8.2.2. Regresión Lineal Múltiple

Consiste en predecir una respuesta numérica y en base a múltiples variables predictoras $x_1, x_2, ..., x_n$, suponiendo una relación lineal.

$$y = a_1 \cdot x_1 + a_2 \cdot x_2 + ... + a_n \cdot x_n + b \tag{19}$$

8.3. Métricas Regresión:

Las métricas de evaluación son medidas utilizadas para evaluar el rendimiento de un modelo de aprendizaje automático. En el caso de los problemas de regresión en el aprendizaje supervisado, estas métricas nos ayudan a determinar qué tan bien nuestro modelo está haciendo predicciones cuantitativas, como valores continuos.

Metricas para la regresión:

- Raíz del error cuadrático medio **(RMSE)**.
- Error absoluto medio **(MAE)**.
- Error cuadratico medio **(MSE)**.
- Suma Residual de los cuadrados **RSS**.

Para cada una de las métricas:

- m : número de instancias
- h : Funcion hipotesis, es el modelo entrenado. En este caso regresion lineal.
- x : Todos los valores de entrada, todas las columnas.

Definición 8.1. (Error cuadrático medio **(MSE)**)

$$MSE(X,h) = \frac{1}{m} \cdot RSS = \frac{1}{m} \cdot \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})^2 \tag{20}$$

Definición 8.2. (Error absoluto medio **(MAE)**

$$MAE(X,h) = \frac{1}{m} \cdot \sum_{i=1}^{m} |h(x^{(i)} - y^{(i)}| \tag{21}$$

Definición 8.3. (Raíz del error cuadrático medio **(RMSE)**

$$RMSE(X,h) = \sqrt{\frac{1}{m} \cdot \sum_{i=1}^{m} (h(x^{(i)} - y^{(i)}))^2} \tag{22}$$

9. Unsupervised Learning

1. Unsupervised Learning
   a) Clustering
      1) K-means.
      2) Clustering jerárquico.

3) DBSCAN.
4) Mezcla de Gaussianas.

b) Association

1) Reglas de asociación.
2) algoritmo Apriori

c) Dimensionality Reduction

- Unsupervised Learning
  - Clustering
    - K-means
    - Clustering jerárquico
    - DBSCAN
    - Mezcla de Gaussianas
  - Association
    - Reglas de asociación
    - Algoritmo Apriori
  - Dimensionality Reduction

9.1. Clustering

En este tipo de problemas se trata de agrupar los datos. Agruparlos de tal forma que queden definidos N conjuntos distinguibles, aunque no necesariamente se sepa que signifiquen esos conjuntos. El agrupamiento siempre será por características similares.

Cuantos clusters elegir:

1. Regla del codo (Elbow Method). fer
2. Método de Silhouette. fer
3. Estadistica de Hopkins. fer

Regla del codo (Elbow Method) En el grafico buscamos un 'codo', el lugar donde baja abruptamente.

- Elegimos un rango, ejemplo 1 a 10, y para cada valor:
  - Para cada centroide calculamos la distancia promedio.

9.1.1. K-means

1. El usuario decide la cantidad de grupos.
2. K-Means elige al azar K centroides.
3. Decide qué grupos están más cerca de cada centroide. Esos puntos forman un grupo.

4. K-Means recalcula los centroides al centro de cada grupo
5. K-Means vuelve a reasignar los puntos usando los nuevos centroides. Calcula nuevos grupos
6. K-means repite punto 4 y 5 hasta que los puntos no cambian de grupo.

Un ejemplo de este algoritmo seria el conjunto de datos Iris, donde se tiene 5 columnas (Largo de sépalo, Ancho de sépalo, Largo de pétalo, Ancho de pétalo, Especies). La columna Especies no la usamos por que estamos en Unsupervised Learning. Con las columnas restantes tenemos que buscar cuantos clusters hay y luego prodria compara con la columna Especies.

10. Preguntas Tipo parcial

1. El NPL sirve para...
   - detertar emociones
   - crear motor de busqueda.
   - traducir texto
   - **todas las anteriores** ✓
2. Count Vectorizer y TF-IDF siempre se guardan todas las palabras de un cuerpoz.
   - **Falso** ✓
   - Verdadero
3. Si coseno del angulo me da 0, significa que:
   - El vector es exactamente el mismo.
   - **Los vectores son perpendiculares**. ✓
   - Los vectores son paralelos pero su norma puede ser distinta.
   - Los vectores son paralelos y su norma es la misma.
4. TF-IDF siempre de mejores resultados que Count Vectorizer.
   - Verdadero
   - **Falso** ✓
5. El TF-IDF es una forma de darle importancia a la raridad"de una palabra en el cuerpo que se hace la busqueda.
   - **Verdadero**✓
   - Falso

6. Stemming necesita un vocavulario de ante mano para llevar las palabras a sus raices.
   - Verdadero
   - **Falso** ✓
7. Lemmatization puede relacionar dos palabras que a lo mejor no tienen exactamente el mismo significado.
   - **Verdadero** ✓
   - Falso

Referencias

[1] Clase 3: Pandas - Split Apply Combine. En: (). URL: https://www.youtube.com/watch?v=85CUMlOMALk.

[2] Lista YouTube Introducción a Data Science. En: (). URL: https://www.youtube.com/watch?v=ezrOvQlIMYE&list=PLeo_qKwGPZYevnuxYBfrvQ32zJJE2--Y4&index=3.

[3] Arboles de Decisión. En: (). URL: https://www.ibm.com/es-es/topics/decision-trees.

[4] varianza y desviación estandar Ejemplo de calculos Esperanza. En: (). URL: https://www.youtube.com/watch?v=6V9a4651WFw.

[5] Colab Regresion Logistica. En: (). URL: https://colab.research.google.com/drive/1JbRUFa5hnijNQdJ_HLywhMJrICkColMJ?authuser=1#scrollTo=sIl68yHmh0yS.

[6] Diferencia entre Machine Learning y Data Science. En: (). URL: https://www.javatpoint.com/data-science-vs-machine-learning#:~:text=Data%20Science%20is%20the%20study,growing%20with%20an%20immoderate%20rate..

[7] Video de YouTube KNN. En: 31 min (). URL: https://www.youtube.com/watch?v=cH-kUai4Boo.