Módulo 3: Aprendizaje automático y visualización

IFCD66 – Data Science

IFCD66 – Data Science
Principios matemáticos

Principios matemáticos

Conceptos

Estadística descriptiva: Las estadísticas descriptivas proporcionan resúmenes simples sobre la muestra y sobre las observaciones que se han realizado. Dichos resúmenes pueden ser cuantitativos, es decir, estadísticas resumidas, o visuales, es decir, gráficos fáciles de entender. Estos resúmenes pueden formar la base de la descripción inicial de los datos como parte de un análisis estadístico más extenso, o pueden ser suficientes por sí mismos para una investigación en particular.

Por ejemplo, el porcentaje de tiro en baloncesto es una estadística descriptiva que resume el rendimiento de un jugador o un equipo. Este número es el número de disparos exitosos dividido entre el número total de disparos realizados. Por ejemplo, un jugador que tira al 33% está acertando aproximadamente un tiro de cada tres. El porcentaje resume o describe múltiples eventos discretos. Considere también el promedio de calificaciones. Este número único describe el rendimiento general de un estudiante en toda la gama de experiencias de su curso.6

El uso de estadísticas descriptivas y resumidas tiene una larga historia y, de hecho, la simple tabulación de poblaciones y de datos económicos fue la primera forma en que apareció el tema de la estadística. Más recientemente, se ha formulado una colección de técnicas de resumen bajo el título de análisis exploratorio de datos.

En el mundo de los negocios, las estadísticas descriptivas proporcionan un resumen útil de muchos tipos de datos. Por ejemplo, los inversores y los corredores pueden utilizar una cuenta histórica del comportamiento de la rentabilidad mediante la realización de análisis empíricos y analíticos de sus inversiones para tomar mejores decisiones de inversión en el futuro.

Principios matemáticos

Conceptos

Análisis exploratorio de datos: El análisis exploratorio de datos es una forma de analizar datos definido por John W. Tukey (E.D.A.: Exploratory data analysis) es el tratamiento estadístico al que se someten las muestras recogidas durante un proceso de investigación en cualquier campo científico. Para mayor rapidez y precisión, todo el proceso suele realizarse por medios informáticos, con aplicaciones específicas para el tratamiento estadístico. Los E.D.A., no necesariamente, se llevan a cabo con una base de datos al uso, ni con una hoja de cálculo convencional; no obstante el programa SPSS y los lenguajes de programación R y Python son las aplicaciones más utilizadas, aunque no las únicas.

Los pasos seguidos en el E. D. A. son básicamente dos:

- Medición y descripción de los datos tecnológicos —tipológicos— y dimensiones, por medio de la Estadística descriptiva. Aquí tenemos, por un lado, las medidas de tendencia central (promedios que, en una sola cifra, resumen todos los valores de una muestra: media, mediana y moda son las más habituales) y, por otro, las medidas de dispersión (que calculan hasta qué punto la muestra se agrupa o no en torno a esos promedios). Dentro de este apartado, se ha de procurar, además, calibrar la confianza de las muestras a través de tres estadímetros básicos: la desviación estándar de la muestra, la curtosis y la asimetría.
- Comparación de los caracteres de una muestra, o de varias muestras diferentes por medio de la Estadística inferencial. Las pruebas más frecuentemente utilizadas comienzan por las más sencillas comparaciones visuales —a través de gráficas como la campana de Gauss, nubes de dispersión o diagramas de caja y arbotantes—, pasando por las socorridas tablas de contingencia (incluido la prueba del $\chi 2$), y por los típicos Análisis de Varianza (que no es más que una confrontación muy precisa de los promedios de varias muestras), hasta llegar a los más complejos análisis multivariantes de conglomerados.

Conceptos

Ejemplo de medición y descripción:

Estadísticos descriptivos	Longitud	Anchura	Grosor	Peso
Número de Mediciones	383	383	383	383
Error estándar	0,01	0,007	0,007	0,086
Medición máxima	142 mm	127 mm	94 mm	1025 g
Medición mínima	29 mm	27 mm	12 mm	16 g
Recorrido	115 mm	98 mm	82 mm	1009 g
Moda	82 mm	60 mm	38 mm	236 g
Mediana	75 mm	61 mm	39 mm	219 g
Rango intercuartil	25 mm	18 mm	17 mm	207 g
Media aritmética	77 mm	62 mm	39 mm	247 g
Desviación estándar	19 mm	14 mm	13 mm	167 g
Coeficiente de variación	25 %	23 %	33 %	68 %
Varianza	376,84	198,67	170,96	27 838,44
Simetría	0,53	0,53	0,48	1,32
Curtosis	0,47	0,83	0,43	2,44

Vamos a ver como podemos categorizar las variables estadísticas y algunos de estos métodos estadísticos…

Conceptos

Variables estadísticas

Una variable estadística es el conjunto de valores que puede tomar cierta característica de la población sobre la cual se hace el estudio estadístico y sobre la cual es posible medirlo.

Estas variables pueden ser: edad, peso, notas de un examen, ingresos mensuales, horas de sueño de una semana, precio mediano del alquiler de un barrio concreto, etc…

Las variables estadísticas se pueden clasificar por diferentes criterios. Según su medición hay dos tipos de variables:

- Cualitativa (o categórica): son las variables que pueden tomar como valores calidades o categorías.
  - Sexo (hombre, mujer). Nominal
  - Salud (buena, regular, mala). Ordinal
- Cuantitativas (o numérica): variables que toman valores numéricos.
  - Número de casas (1, 2,…). Discreta
  - Temperatura (12.5; 24.3; 35.2,…). Continúa

Conceptos

Métodos estadísticos descriptivos

Los métodos estadísticos que hemos visto antes los vamos a clasificar en:

Posición central: las medidas de posición central (o centralización) son medidas que tienden a localizar en qué punto se encuentra la parte central de un conjunto ordenado de datos de una variable cuantitativa. Son, por ejemplo, la **media**, la **mediana** y la **moda**.

Posición no central: las medidas de posición no central (o medidas de tendencia no central) permiten conocer puntos característicos de una serie de valores, que no necesariamente tienen que ser centrales. La intención de estas medidas es dividir el conjunto de observaciones en grupos con el mismo número de valores en cada uno. Son, por ejemplo, los **cuartiles** y los **percentiles**.

Dispersión: las medidas de dispersión o medidas de variabilidad muestran la variabilidad de un conjunto de datos, indicando la concentración o más grande de datos respecto a las medidas de centralización. Son, por ejemplo, el **rango**, el **rango intercuartílico**, la **varianza**, la **desviación típica**, la **desviación media** y el **coeficiente de variación.**

Distribución de variables: informan sobre la forma de la distribución de una variable. Estas medidas permiten saber las características de su asimetría y homogeneidad sin necesidad de graficar los datos. Son, por ejemplo, la a**simetría** y la **curtosis**.

Frecuencias: La frecuencia es una medida que sirve para comparar la aparición de un elemento Xi en un conjunto de elementos (X1, X2,..., XN). Mediante tablas de distribuciones de frecuencia se puede presentar organizadamente el recuento de datos. Las frecuencias de cada elemento se pueden expresar tanto como **frecuencias absolutas** (número total de apariciones) como **frecuencias relativas** (proporción de apariciones).

Conceptos

Métodos estadísticos de posición central

Media: también llamada media aritmética, es un promedio estándar que a menudo se denomina promedio. La función de Python sería **np.mean()**

Mediana: a diferencia de la media, la mediana representa el valor de la variable de posición central en un conjunto de datos ordenados. Es decir, una vez ordenadas los datos, la mediana es el valor del conjunto tal que el 50% de los elementos son menores o iguales y el otro 50%, más grandes o iguales. La función de Python sería **np.median()**

Moda: es el valor que aparece con mayor frecuencia en un conjunto de datos. La función de Python sería **np.mode()**

Métodos estadísticos de posición no central

Cuartiles : Los cuartiles son los tres valores que dividen una serie de datos ordenada en cuatro partes iguales. El primer cuartil (Q1) deja a la izquierda el 25% de los datos. El segundo (Q2) deja a izquierda y derecha el 50% y coincide con la media. El tercero (Q3) deja a la derecha el 25% de valores.

Percentiles: Los percentiles son los 99 puntos que dividen una serie de datos ordenados en 100 partes iguales, es decir, que contienen el mismo número de elementos cada una. El percentil 50 es la media. La función de Python sería **np.percentile()**

Conceptos

Métodos estadísticos de dispersión:

Rango: El rango o el recorrido estadístico es la diferencia entre el valor máximo y el mínimo de un conjunto de elementos.

```
np.max(pesos) - np.min(pesos)
```

Rango intercuartílico: El rango intercuartílico IQR (o rango interquartil) es una estimación estadística de la dispersión de una distribución de datos. Consiste en la diferencia entre el tercero y el primer cuartil. Mediante esta medida se eliminan los valores extremadamente alejados. El rango intercuartílico es altamente recomendable cuando la medida de tendencia central utilizada es la media (puesto que este estadístico es insensible a posibles irregularidades a los extremos).

```
np.percentile()
```

Varianza: La varianza mide la dispersión de los datos de una muestra respecto de la media, calculando la media de los cuadrados de las distancias de todos los datos.

```
np.var()
```

Conceptos

Métodos estadísticos de dispersión:

Desviación típica : La desviación típica es la medida de dispersión asociada a la media. Mide la media de las desviaciones de los datos respecto a la media a las mismas unidades de los datos.

```
np.std()
```

Desviación media: La desviación media es la media de los valores absolutos de la diferencia de cada valor de la distribución con la media aritmética.

```
np.mean() y np.abs()
np.mean(np.absolute(data – np.mean(data)))
```

Coeficiente de variación: El coeficiente de variación mesura la variación de los datos respecto a la media, sin tener en cuenta las unidades en que están. El coeficiente de variación toma valores entre 0 y 1. Si el coeficiente es próximo al 0, significa que hay poca variabilidad en los datos y es una muestra muy compacta. En cambio, si tienden a 1, es una muestra muy dispersa y la media pierde confianza. De hecho, cuando el coeficiente de variación supera el 30% (0,3), se llama que la media es poco representativa. Se calcula como la desviación típica dividida por la media aritmética y se expresa en porcentaje.

```
np.mean() y np.std()
```

Conceptos

Métodos estadísticos de distribución de variables:

Asimetría: La asimetría es la medida que indica la simetría de la distribución de una variable respecto de la media aritmética, sin necesidad de hacer la representación gráfica. Los coeficientes de asimetría indican si hay el mismo número de elementos a izquierda y derecha de la media.

En data science, la asimetría se puede usar para evaluar si un conjunto de datos sigue o no una distribución normal. Un conjunto de datos que sigue una distribución normal es simétrico en torno a su media, mientras que un conjunto de datos que no sigue una distribución normal puede ser asimétrico. La asimetría también se puede usar para identificar patrones o tendencias en un conjunto de datos. Por ejemplo, si un conjunto de datos tiene una asimetría positiva, eso puede indicar que hay una mayor cantidad de valores por encima de la media que por debajo de ella. Por otro lado, si un conjunto de datos tiene una asimetría negativa, eso puede indicar que hay una mayor cantidad de valores por debajo de la media que por encima de ella.

Hay tres tipos de curva de distribución según su asimetría:

- Asimetría negativa: la cola de la distribución se alarga para valores inferiores a la media. En este caso, el coeficiente de asimetría es negativo
- Simétrica: hay el mismo número de elementos a izquierda y derecha de la media. En este caso, coinciden la media, la media y la moda. La distribución se adapta a la forma de la campana de Gauss, o distribución normal que corresponde a un coeficiente de asimetría igual a cero
- Asimetría positiva: la cola de la distribución se alarga para valores superiores a la media que proporciona un coeficiente de asimetría positivo.

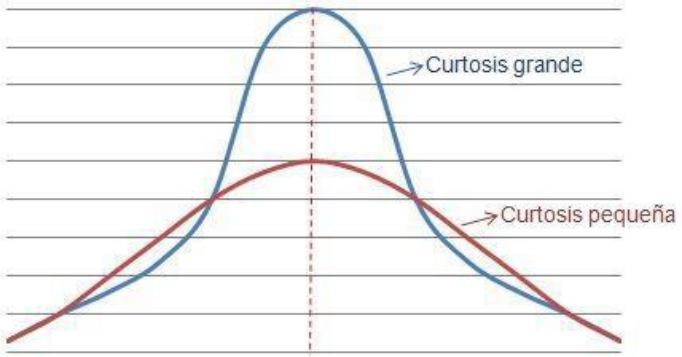
Conceptos

Métodos estadísticos de distribución de variables:

Curtosis: La curtosis es una medida de forma que mide como escarpada o apuntada está una curva o distribución. Este coeficiente indica la cantidad de datos que hay próximas a la media, de forma que cuanto más grado de curtosis, más escarpada (o apuntada) será la forma de la curva.

En data science, la curtosis se puede usar para evaluar si un conjunto de datos sigue o no una distribución normal. Un conjunto de datos que sigue una distribución normal tiene una curtosis cercana a 3, mientras que un conjunto de datos que no sigue una distribución normal puede tener una curtosis distinta de 3.

La curtosis también se puede usar para identificar patrones o tendencias en un conjunto de datos. Por ejemplo, si un conjunto de datos tiene una curtosis mayor que 3, eso puede indicar que hay una mayor cantidad de valores extremos (es decir, muy alejados de la media) en el conjunto de datos. Por otro lado, si un conjunto de datos tiene una curtosis menor que 3, eso puede indicar que hay una menor cantidad de valores extremos en el conjunto de datos.



Conceptos

EJERICIO M3-1:

Calcula en PYTHON las medidas descriptivas central, no central, de dispersión y de distribución de variables de los diferentes datos del archivo “iris.arff” que está en “Materiales de clase”.”MODUL 3” :

- sepal length
- sepal width
- petal length
- petal width

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import skew
from scipy.stats import kurtosis
```

Conceptos

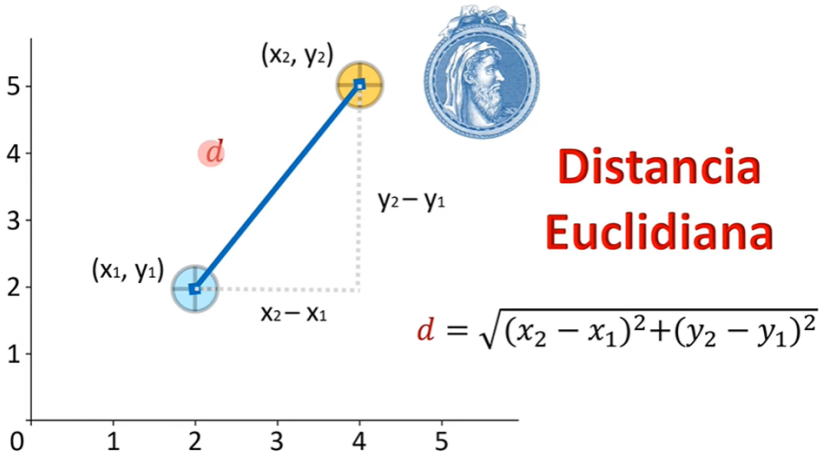
Distancia euclídea (clusters): es la distancia entre dos puntos. Se calcula con el teorema de Pitágoras.

```
distance = np.linalg.norm(point_1 - point_2)
```

En data science, la distancia Euclídea se puede usar para comparar vectores de características o para clasificar o agrupar datos.

Por ejemplo, supongamos que tienes un conjunto de datos con dos características: el peso y la altura de una persona. La distancia Euclídea te permite calcular la distancia entre dos personas en este espacio de dos dimensiones. Así, puedes comparar dos personas y determinar si tienen características similares o diferentes.

D=3,6



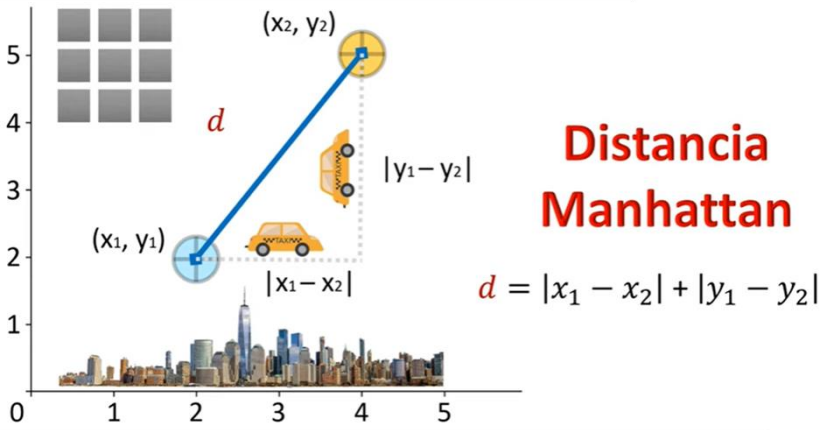
Conceptos

Distancia Mahattan (clusters): es la suma de las distancias de los ejes. Toma del supuesto movimiento de un taxi en el barrio de Manhattan, en New York. Se calcula como la suma de las diferencias absolutas de las coordenadas de los puntos en cada dimensión. distance = np.sum(np.abs(point_1 - point_2))

En data science, la distancia de Manhattan se puede usar para comparar vectores de características o para clasificar o agrupar datos.

Por ejemplo, supongamos que tienes un conjunto de datos con dos características: el peso y la altura de una persona. La distancia de Manhattan te permite calcular la distancia entre dos personas en este espacio de dos dimensiones. Así, puedes comparar dos personas y determinar si tienen características similares o diferentes.

D=5



Distancia Manhattan

$$d = |x_1 - x_2| + |y_1 - y_2|$$

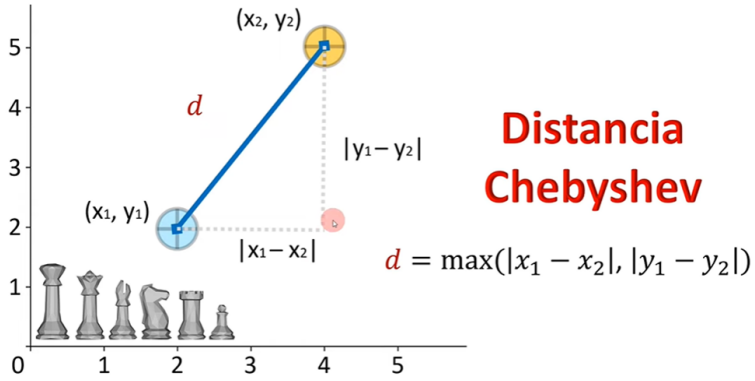
Conceptos

Distancia Chebyshev (clusters): también llamada distancia del tablero de ajedrez. Es el máximo de las distancias respecto sus ejes. Serían los movimientos del rey en un tablero de ajedrez. distance = np.max(np.abs(point_1 - point_2))

Se calcula como el máximo de las diferencias absolutas de las coordenadas de los puntos en cada dimensión. En data science, la distancia de Chebyshev se puede usar para comparar vectores de características o para clasificar o agrupar datos.

Por ejemplo, supongamos que tienes un conjunto de datos con dos características: el peso y la altura de una persona. La distancia de Chebyshev te permite calcular la distancia entre dos personas en este espacio de dos dimensiones. Así, puedes comparar dos personas y determinar si tienen características similares o diferentes.

D=3



Distancia Chebyshev

$$d = \max(|x_1 - x_2|, |y_1 - y_2|)$$

Conceptos

Matriz de confusión: se usa para valorar la precisión de una tarea de clasificación.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
Kappa statistic                          0
Mean absolute error                      0.4444
Root mean squared error                  0.4714
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    1,000    0,333      1,000    0,500      ?        0,500     0,333     Iris-setosa
                 0,000    0,000    ?          0,000    ?          ?        0,500     0,333     Iris-versicolor
                 0,000    0,000    ?          0,000    ?          ?        0,500     0,333     Iris-virginica
Weighted Avg.    0,333    0,333    ?          0,333    ?          ?        0,500     0,333

=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = Iris-setosa
 50  0  0 |  b = Iris-versicolor
 50  0  0 |  c = Iris-virginica
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         138               92      %
Incorrectly Classified Instances        12                8      %
Kappa statistic                          0.88
Mean absolute error                      0.0533
Root mean squared error                  0.2309
Relative absolute error                 12      %
Root relative squared error             48.9898 %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     Iris-setosa
                 0,880    0,060    0,880      0,880    0,880      0,820    0,910     0,814     Iris-versicolor
                 0,880    0,060    0,880      0,880    0,880      0,820    0,910     0,814     Iris-virginica
Weighted Avg.    0,920    0,040    0,920      0,920    0,920      0,880    0,940     0,876

=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = Iris-setosa
  0 44  6 |  b = Iris-versicolor
  0  6 44 |  c = Iris-virginica
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0,980    0,000    1,000      0,980    0,990      0,985    0,990     0,987     Iris-setosa
                 0,940    0,030    0,940      0,940    0,940      0,910    0,952     0,880     Iris-versicolor
                 0,960    0,030    0,941      0,960    0,950      0,925    0,961     0,905     Iris-virginica
Weighted Avg.    0,960    0,020    0,960      0,960    0,960      0,940    0,968     0,924

=== Confusion Matrix ===

  a  b  c   <-- classified as
 49  1  0 |  a = Iris-setosa
  0 47  3 |  b = Iris-versicolor
  0  2 48 |  c = Iris-virginica
```

Conceptos

Tipos de medidas de cálculo de error (clasificación)

- Error de clasificación: es la suma de los errores dividido por el número de predichos, donde (a,b)=0 si a=b y 1 si a<>b.

$$error_S(h) = \frac{1}{n} \sum_{x \in S} \partial(f(x), h(x))$$

Valor predicho (h(x))	Valor observado (real f(x))	Error
Vender	Vender	0
Vender	Vender	0
No Vender	Vender	1
No Vender	No Vender	0
No Vender	Vender	1
No Vender	No Vender	0
Vender	Vender	0
Vender	No Vender	1
Vender	Vender	0
No Vender	Vender	1
		4

Número de medidas	10

ERRORES	
Errores	0,4

Conceptos

Tipos de medidas de cálculo de error (regresión):

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
Kappa statistic                          0
Mean absolute error                      0.4444
Root mean squared error                  0.4714
Relative absolute error                100      %
Root relative squared error            100      %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         138               92      %
Incorrectly Classified Instances        12                8      %
Kappa statistic                          0.88
Mean absolute error                      0.0533
Root mean squared error                  0.2309
Relative absolute error                 12      %
Root relative squared error             48.9898 %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

```
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

- Error absoluto relativo: En estadística, el error relativo se calcula dividiendo el error absoluto entre el valor real. Con esto lo que hacemos es calcular el error que cometemos por unidad de medida. Es decir, cuanto nos equivocamos por kilo, por metro o por persona… de modo que ahora ya podemos comparar unas medidas con otras.

Principios matemáticos

Conceptos

El error absoluto y el error relativo son medidas comunes de la precisión de un modelo de machine learning. El error absoluto se refiere a la diferencia entre el valor real y el valor predicho, mientras que el error relativo se refiere a la diferencia entre el valor real y el valor predicho expresado como un porcentaje del valor real.

Un ejemplo de cómo se utilizan estas medidas podría ser en el caso de un modelo de regresión lineal que se utiliza para predecir el precio de una casa. El error absoluto sería la diferencia entre el precio real de la casa y el precio predicho por el modelo, mientras que el error relativo sería esa misma diferencia expresada como un porcentaje del precio real.

Por ejemplo, si el precio real de una casa es de $200,000 y el modelo lo predice en $180,000, el error absoluto sería de $20,000 y el error relativo sería del 10% (20,000/200,000).

Error absoluto es útil para determinar cuanto se aleja el valor predicho con respecto al valor real mientras que el Error relativo es mas usado para comparar las diferencias entre varios modelos o para determinar que tan preciso es un modelo respecto a un valor específico, es decir es útil para determinar cuan específico es el modelo en ciertos puntos o rango de valores específicos.

En general, se espera que un buen modelo tenga tanto un error absoluto bajo como un error relativo bajo. A menudo, se utilizan ambas medidas juntas para tener una comprensión más completa de la precisión del modelo.vamos a ver un ejemplo de cálculo.

Conceptos

Tipos de medidas de cálculo de error (regresión):

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
Kappa statistic                          0
Mean absolute error                      0.4444
Root mean squared error                  0.4714
Relative absolute error                100      %
Root relative squared error            100      %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         138               92      %
Incorrectly Classified Instances        12                8      %
Kappa statistic                          0.88
Mean absolute error                      0.0533
Root mean squared error                  0.2309
Relative absolute error                 12      %
Root relative squared error             48.9898 %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

```
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

- Error medio absoluto: En estadística, el error absoluto medio es una medida de la diferencia entre dos variables continuas. Considerando dos series de datos (unos calculados y otros observados) relativos a un mismo fenómeno, el error absoluto medio sirve para cuantificar la precisión de una técnica de predicción comparando por ejemplo los valores predichos frente a los observados, el tiempo real frente al tiempo previsto, o una técnica de medición frente a otra técnica alternativa de medición.

$$MAE_S(h) = \frac{1}{n}\sum_{x \in S} \left| f(x) - h(x) \right|$$

Principios matemáticos

Conceptos

El error medio absoluto (MAE, por sus siglas en inglés) es una medida de precisión utilizada en el aprendizaje automático y el análisis estadístico para evaluar el rendimiento de un modelo predictivo. Se calcula como la media de los valores absolutos de la diferencia entre los valores predichos y los valores reales.

Por ejemplo, si un modelo predice los precios de venta de casas en un área determinada y los precios reales son $200,000, $225,000, $210,000 y $230,000, y el modelo predice $210,000, $212,500, $205,000 y $225,000, el MAE sería:

$MAE = (|200,000 - 210,000| + |225,000 - 212,500| + |210,000 - 205,000| + |230,000 - 225,000|) / 4 = 15,000 / 4 = $3,750

En este ejemplo, el modelo tiene un MAE de $3,750, lo que indica que en promedio, el modelo se desvía en $3,750 de los precios reales de venta de las casas.

El MAE es una medida útil para evaluar el rendimiento de un modelo predictivo ya que tiene en cuenta tanto los valores subestimados como los valores sobreestimados, lo que lo hace más equitativo que otras medidas de precisión como el error cuadrático medio (MSE). Sin embargo, debido a que el MAE utiliza valores absolutos, no tiene en cuenta la magnitud de las diferencias entre los valores predichos y los valores reales. Por lo tanto, se recomienda utilizar MAE junto con otras medidas de precisión para obtener una mejor comprensión del rendimiento del modelo.

Conceptos

Tipos de medidas de cálculo de error (regresión):

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
Kappa statistic                          0
Mean absolute error                      0.4444
Root mean squared error                  0.4714
Relative absolute error                100      %
Root relative squared error            100      %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         138               92      %
Incorrectly Classified Instances        12                8      %
Kappa statistic                          0.88
Mean absolute error                      0.0533
Root mean squared error                  0.2309
Relative absolute error                 12      %
Root relative squared error             48.9898 %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

```
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

- Error medio absoluto relativo: En estadística, el error medio absoluto relativo se calcula dividiendo el error absoluto medio entre el valor real.

$$RAE_S(h) = \frac{\sum_{x \in S} |f(x) - h(x)|}{\sum_{x \in S} |\bar{f} - f(x)|}$$

Principios matemáticos

Conceptos

El error medio absoluto relativo (EMAR) es una medida utilizada para evaluar la precisión de un modelo predictivo. Se calcula como la diferencia entre el valor predicho y el valor real, expresado en términos de porcentaje del valor real. Es decir, se divide el error medio absoluto (EMA) por el valor real y se multiplica por 100.

Por ejemplo, si un modelo predictivo tiene un EMA de 20 y el valor real es 100, el EMAR sería de 20%. Esto significa que el modelo se equivoca en un 20% del valor real en promedio. Un EMAR más bajo indica un modelo más preciso.

En data science, el EMAR es una herramienta útil para comparar la precisión de diferentes modelos predictivos y seleccionar el mejor para una aplicación específica. También se utiliza para evaluar la precisión de un modelo en comparación con una línea base, como un modelo de regresión lineal simple.

Conceptos

Tipos de medidas de cálculo de error (regresión):

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
Kappa statistic                          0
Mean absolute error                      0.4444
Root mean squared error                  0.4714
Relative absolute error                100      %
Root relative squared error            100      %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         138               92      %
Incorrectly Classified Instances        12                8      %
Kappa statistic                          0.88
Mean absolute error                      0.0533
Root mean squared error                  0.2309
Relative absolute error                 12      %
Root relative squared error             48.9898 %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

```
Mean absolute error                       0.035
Root mean squared error                   0.1586
Relative absolute error                   7.8705 %
Root relative squared error              33.6353 %
```

- Error cuadrático medio: En estadística, el error cuadrático medio (ECM) de un estimador mide el promedio de los errores al cuadrado, es decir, la diferencia entre el estimador y lo que se estima. El ECM es una función de riesgo, correspondiente al valor esperado de la pérdida del error al cuadrado o pérdida cuadrática. La diferencia se produce debido a la aleatoriedad o porque el estimador no tiene en cuenta la información que podría producir una estimación más precisa.

$$MSE_S(h) = \frac{1}{n}\sum_{x \in S}(f(x) - h(x))^2$$

Principios matemáticos

Conceptos

El error cuadrático medio (MSE, por sus siglas en inglés) es una métrica comúnmente utilizada en data science para evaluar la precisión de un modelo de predicción. Sirve para medir la diferencia entre los valores predichos por el modelo y los valores reales.

Por ejemplo, si estamos tratando de predecir el precio de una casa en función de su tamaño y su ubicación, el MSE nos ayudaría a medir cuán lejos están nuestras predicciones del precio real de las casas. Si el MSE es bajo, significa que nuestras predicciones son bastante precisas, mientras que si es alto, significa que nuestras predicciones son menos precisas.

Un ejemplo concreto podría ser el siguiente: si tenemos un conjunto de datos con 10 casas y sus respectivos precios reales, y nuestro modelo predice los siguientes precios: 200.000, 250.000, 225.000, 275.000, 225.000, 250.000, 225.000, 300.000, 250.000 y 275.000. Si calculamos el MSE, podríamos ver que es de alrededor de 25.000 (lo que significa que nuestras predicciones están, en promedio, 25.000 dólares lejos del precio real).

Conceptos

Tipos de medidas de cálculo de error (regresión):

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
Kappa statistic                          0
Mean absolute error                      0.4444
Root mean squared error                  0.4714
Relative absolute error                100      %
Root relative squared error            100      %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         138               92      %
Incorrectly Classified Instances        12                8      %
Kappa statistic                          0.88
Mean absolute error                      0.0533
Root mean squared error                  0.2309
Relative absolute error                 12      %
Root relative squared error             48.9898 %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

```
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

- Error cuadrático medio relativo: En estadística, el error cuadrático medio relativo se calcula dividiendo el error cuadrático medio entre el valor real.

$$RSE_S(h) = \frac{\sum_{x \in S} \left(f(x) - h(x)\right)^2}{\sum_{x \in S} \left(\bar{f} - f(x)\right)^2}$$

Principios matemáticos

Conceptos

El Error Cuadrático Medio Relativo (ECMR) es una medida de precisión utilizada para evaluar la bondad de ajuste de un modelo predictivo en relación a los datos reales. Se utiliza para comparar el rendimiento de diferentes modelos y para evaluar si un modelo es adecuado para un conjunto de datos específico.

Por ejemplo, si se tiene un modelo de regresión lineal que se utiliza para predecir el precio de una propiedad en función de su tamaño, el ECMR se calcula dividiendo el error cuadrático medio entre los valores predichos y los valores reales, y luego se divide por el precio medio de las propiedades en el conjunto de datos. Si el ECMR es muy alto, significa que el modelo no se ajusta bien a los datos reales, lo que indica que es necesario encontrar un modelo más preciso.

Conceptos

Tipos de medidas de cálculo de error (regresión):

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
Kappa statistic                          0
Mean absolute error                      0.4444
Root mean squared error                  0.4714
Relative absolute error                100      %
Root relative squared error            100      %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         138               92      %
Incorrectly Classified Instances        12                8      %
Kappa statistic                          0.88
Mean absolute error                      0.0533
Root mean squared error                  0.2309
Relative absolute error                 12      %
Root relative squared error             48.9898 %
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
```

```
Mean absolute error                       0.035
Root mean squared error                   0.1586
Relative absolute error                   7.8705 %
Root relative squared error              33.6353 %
```

- Raíz cuadrada error cuadrático medio: En estadística, es una estimación que mide la raíz cuadrada de la diferencia media al cuadrado entre los valores estimados y los valores reales de un conjunto de datos. En el análisis de regresión, el RECM calcula la raíz cuadrada de las diferencias medias al cuadrado entre los puntos y la línea de regresión. Es decir, la raíz cuadrada de la media de los cuadrados de los residuos. .

$$RMSE_S(h) = \sqrt{\frac{1}{n}\sum_{x \in S}(f(x) - h(x))^2}$$

Principios matemáticos

Conceptos

La Raíz Cuadrada del Error Cuadrático Medio (RMSE, por sus siglas en inglés) es una medida comúnmente utilizada para evaluar el rendimiento de un modelo de predicción en data science. Se calcula como la raíz cuadrada de la media de los errores cuadrados entre los valores predichos por el modelo y los valores reales.

Por ejemplo, si un modelo de predicción de precios de viviendas predice valores para 10 viviendas y los errores cuadrados entre los valores predichos y los valores reales son (1, 4, 9, 4, 1, 4, 9, 16, 25, 36), el RMSE sería la raíz cuadrada de la media de esos errores cuadrados, que es √((1+4+9+4+1+4+9+16+25+36)/10) = √(90/10) = √9 = 3.

La ventaja de utilizar el RMSE es que al tener las unidades de la variable objetivo, permite comparar la magnitud del error en las predicciones del modelo con los valores reales. Por lo general, un RMSE bajo indica un mejor rendimiento del modelo. Sin embargo, es importante tener en cuenta que este no es el único indicador para evaluar el rendimiento de un modelo (de hecho, ninguno lo es...).

Conceptos

Tipos de medidas de cálculo de error: vamos a ver un ejemplo de cálculo.

Valor predicho (h(x))	Valor observado (real f(x))	Error	Error al cuadrado
100	102	2	4
102	110	8	64
105	95	10	100
95	75	20	400
101	103	2	4
105	110	5	25
105	98	7	49
40	32	8	64
220	215	5	25
100	103	3	9
		70	744

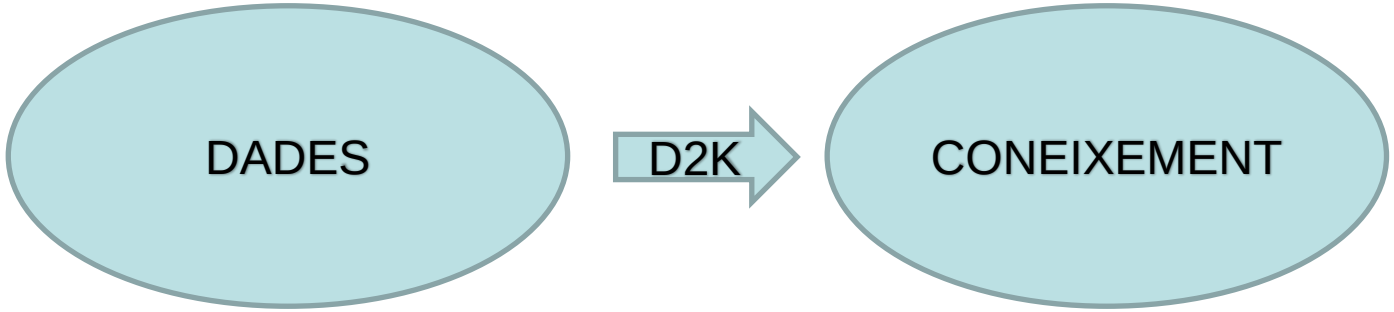
Número de medidas	10

ERRORES	
Error Medio Absoluto	7
Error Cuadràtico medio	74,4
Raíz Cuadrada del Error Cuadrático Medio	8,625543

IFCD66 – Data Science
Minería de datos y Data Science

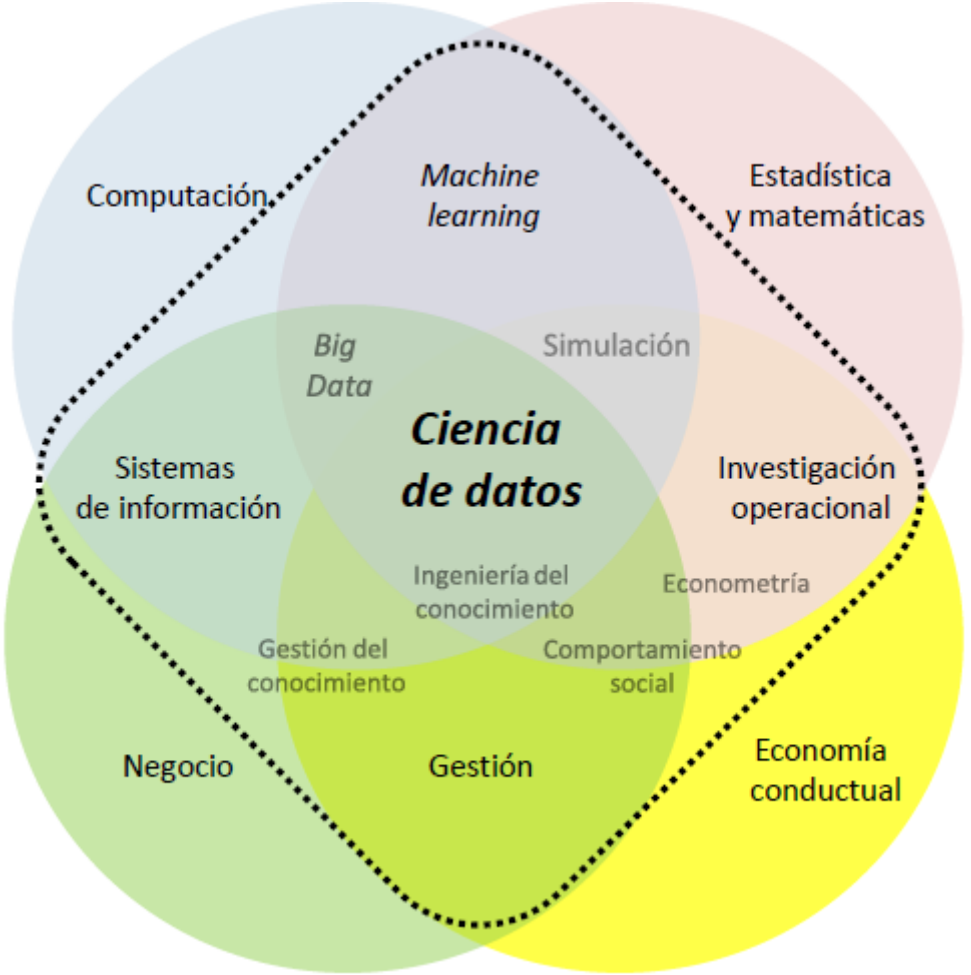
Mineria de dades i ciència de dades

Hi ha diverses paraules al voltant de tot el tractament de dades, però la idea principal és aquesta:



- Mineria de dades: La mineria de dades està associada a eines i intel·ligència empresarial.
- Ciència de dades: La ciència de dades està associada amb una professió inquisitiva.
- Anàlisi de dades (Intel·ligent): Semblant a la mineria de dades, usat principalment per a estadística.
- Anàlisi predictiva (de dades): Un nom més elegant per a l'anàlisi de dades.
- Big Data:
  - No tots els projectes Big Data requereixen anàlisis.
  - No tots els projectes de ciència de dades requereixen una infraestructura per a Big Data.
- Extracció de coneixement (des de bases de dades), KDD: El descobriment de coneixement en bases de dades (KDD, de l'anglès Knowledge Discovery in Databases) és bàsicament un procés automàtic en el qual es combinen descobriment i anàlisi. El procés consisteix a extreure patrons en forma de regles o funcions, a partir de les dades, perquè l'usuari els analitzi.

La Ciència de dades inclou tot un conjunt de tècniques i àrees diverses



Mineria de dades i ciència de dades

Els actors principals de la ciència de dades són:

- ✓ Director de sistemes de la informació (Chief information officer , CIO): Terme tradicional per als alts executius de Sistemes de la informació i Tecnologies de la informació
- ✓ Data Manager: Terme tradicional per al responsable de la gestió de bases de dades
- ✓ Director de dades (Chief Data Officer, CDO): responsable de la direcció de l'empresa i la utilització de la informació com un actiu, a través del processament de dades, l'anàlisi, l'extracció de dades, l'intercanvi d'informació i altres mitjans
- ✓ Científic de dades: més que una persona, és un conjunt integrat d'habilitats que abasta matemàtiques, aprenentatge automàtic, intel·ligència artificial, estadístiques, bases de dades i optimització, juntament amb un profund coneixement de l'elaboració de problemes per a dissenyar solucions efectives

Mineria de dades i ciència de dades

La ciència de dades bàsicament, i des d'un punt de vista de les dades, es pot dividir en:

- Les meves dades són valuoses per a mi (in->in): companyies d'assegurances i pòlisses
  - ✓ Dades internes útils per a l'organització.
  - ✓ Intel·ligència empresarial clàssica.

- Aquestes dades són valuoses per a mi (out->in): lladres no molt llestos i xarxes socials... Detroit
  - ✓ Dades externes útils per a l'organització.
  - ✓ Mitjans socials, Internet, dades obertes, ...

- Les meves dades són valuoses per a uns altres (in->out): dades de telecos útils per altres usos.. Smart Steps
  - ✓ Dades internes útils per a altres organitzacions.... O perquè no IA's ???
  - ✓ Les meves dades tenen utilitat per a uns altres,

- Aquestes dades són valuoses per a uns altres (out->out): fundacions que analitzen i donen informació
  - ✓ Dades externes útils per a altres organitzacions: Healthcaredataanalysis.org
  - ✓ Aquestes dades tenen utilitat per a uns altres, ... (Científic de dades freelance)

- Creant dades (0->out): Waze
  - ✓ Col·leccionar dades que poden tenir valor. (Emprenedor de dades)

Exemples de tipus de dades:

- Dades de telecomunicacions: Valuós per a comerciants, trànsit, ajuntaments, policia...
- Altres dades de geolocalització (*Flickr ,Instagram, *Wikiloc ,...): Valuós per a agències de viatge...
- Dades en consum d'energia: medi ambient
- Dades del transport públic (bus, metro,tren, taxi, trànsit, ...): per turisme, consum, comerç...
- Dades de xarxes socials i cerques web : Valuós per a gairebé tot...
- Dades d'ús de targetes de crèdit: Valuós per a comerços, ajuntaments, ...
- Dades de policia: asseguradores, agents immobiliaris, ...
- Dades comercials (Amazon , Ebay , segundamano.es ,...) : demografia, sociologia...
- Dades climatològiques: Valuós per a comerços.

IFCD66 – Data Science
Tareas y técnicas

Tareas, tècnicas

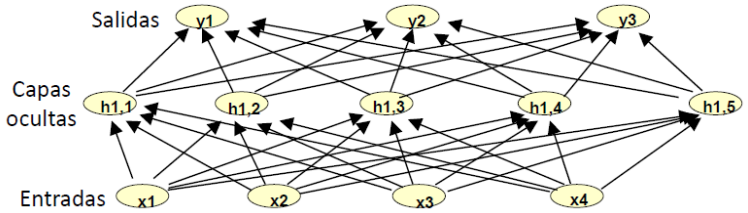
Hi ha dos tipus de tasques quan parlem de ciència de dades:

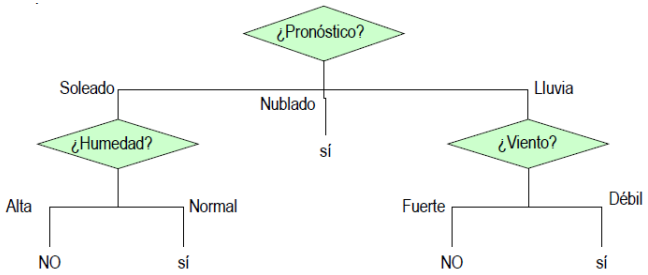
- Predictives: són les que tenen una variable de sortida, i prediuen el valor d'un atribut:
  - ¿Cuáles serán las ventas el año próximo?
  - ¿Es esta transacción fraudulenta?
  - ¿Qué tipo de seguro es más probable que contrate el cliente X?
  - Classificació/Categorització: la variable de sortida és nominal
  - Regressió: la variable de sortida és numèrica.
- Descriptives: No hi ha variable de sortida, proporcionan información sobre las relaciones entre los datos y sus características:
  - Genera información del tipo:
    - Los clientes que compran pañales suelen comprar cerveza.
    - El tabaco y el alcohol son los factores más importantes en la enfermedad Y.
    - Los clientes sin televisión y con bicicleta tienen características muy diferenciadas del resto.
  - Clustering: l'objectiu és descobrir grups de dades
  - Anàlisi exploratori:
    - Regles d'associació, dependències funcionals: les variables són nominals
    - Anàlisi factorial/de correlació, anàlisis de dispersió, anàlisi multivariable: les variables són numèriques

Tareas, tècnicas

Relación entre tareas y tècnicas

TÉCNICA	PREDICITVA / SUPERVISADA		DESCRIPTIVA / NO SUPERVISADA		
	Clasificación	Regresión	*Clustering*	Reglas de asociación	Otros (factorial, correlación…)
Redes neuronales	✓	✓	✓ *		
Árboles de decisión	✓ (c4.5)	✓ (CART)	✓		
Kohonen			✓		
Regresión linear (local, global), exp..		✓			
Regresión logística	✓				
K-means	✓ *		✓		
A Priori (associaciones)				✓	
Análisis factorial, análisis multivariable					✓
CN2	✓				
K-NN (vecinos más próximos)	✓		✓		
FBR	✓				
Clasificadores básicos	✓	✓			





Tipos de aprendizaje

- Aprendizaje supervisado: tiene un propósito específico (target) a deducir/predecir a partir de los datos. Los datos de entrenamiento etiquetados con el valor real de la clase o función.
  - D={(x,y)}, x=x1,…,xn son los atributos de entrada; y es la salida (target)
  - Clasificación/Categorización: la salida es categórica.
  - Regresión: la salida es numérica.

- Aprendizaje no supervisado: no tiene un propósito específico. Los datos de entrenamiento no etiquetados
  - D={(x1,…,xn)}
  - Clustering: descubrir grupos entre los datos.
  - Análisis exploratorio: encontrar relaciones entre las variables.

Predictivas

- Clasificación: Una clasificación puede verse como la dependencia de un atributo el cual puede tomar un valor de entre varias etiquetas de clase.
  - Ejemplo: de entre todos los clientes Movistar, ¿cuáles responderán positivamente a una oferta dada? Este ejemplo tiene dos clases (binario): responde y no responde
  - La idea es determinar las reglas exactas que clasifican un caso como positivo onegativo a partir de los otros atributos.

- Regresión: El objetivo es predecir el valor de una variable continua a partir de otras variables continuas o categóricas.
  - Ejemplo: necesitamos conocer el número de futuros clientes o de pacientes, ingresos, llamadas, ganancias, costes, etc. a partir de resultados previos (días, semanas, meses o años anteriores).

Descriptivas

- Clustering: El objetivo es encontrar grupos de individuos porque son "similares".
  - Se diferencia de la clasificación en que no conocemos los grupos (ni su número)
  - Ejemplo: determinar qué tipos de clientes tengo atendiendo a sus patrones de compra.

- Asociaciones y dependencias funcionales:
  - Una **asociación** entre dos atributos ocurre cuando la frecuencia de que dos valores concretos ocurran juntos es relativamente alta.
    - Ejemplo: analizar pares de libros que se compran frecuentemente juntos en una tienda de libros.

  - Una **dependencia funcional** es un patrón en el que se establece que uno o más atributos determinan el valor de otro.
    - Ejemplo: Supongamos una base de datos médicos que contiene dos columnas, "Enfermedad" y "Síntoma". Si se cumple que todos los pacientes que padecen neumonía tienen como síntoma fiebre podemos decir que fiebre está asociado a neumonía. Si esta asociación sucede para cada par de valores de las columnas "Enfermedad" y "Síntoma", entonces existe una dependencia funcional entre "Enfermedad" y "Síntoma.

Descriptivas

- Correlaciones: Permiten determinar el grado de similitud entre variables numéricas en términos de su relación en magnitud (Pearson) o por su orden (Spearman).
  - Ejemplo: En una cadena de supermercados se analizan los datos por tienda y se observa que el número de clientes y las ventas totales están positivamente correlacionadas.

Ejemplos de problemas a solucionar:

- Agente bancario: Debo ofrecerle una hipoteca a este cliente ?
  - Tarea: Clasificación
  - Técnica: Árbol de decisión ?
- Gerente de supermercado: Cuando mis clientes compran huevos, cogen también aceite ?
  - Tarea: Reglas de asociación
  - Técnica: A priori(%confianza y %apoyo) ?
- Gerente de personal: Qué tipo de empleados tengo ?
  - Tarea: Clustering
  - Técnica: K-means ?
- Supervisor de fábrica: Cuántos fallos para X módulo esperamos cada mes
  - Tarea: Regresión
  - Técnica: Regresión lineal, red neuronal… ?
- Predicción de tiempo: Saber si podemos jugar un partido dependiendo del tiempo:
  - Tarea : Clasificación
  - Técnica: Árbol de decisión ?
- Fabricación vidrio: que % de distintos compuestos poner para conseguir cierta calidad de vidrio ?
  - Tarea: Clasificación
  - Técnica:

IFCD66 – Data Science Herramientas

Herramientas

Weka

Weka es una colección de algoritmos de aprendizaje automático para tareas de minería de datos. Contiene herramientas para la preparación, clasificación, regresión, agrupación, minería de reglas de asociación y visualización de datos. Encontrado solo en las islas de Nueva Zelanda, el Weka es un ave no voladora.

Weka es un software de código abierto emitido bajo la Licencia Pública General GNU.

Explorer

Experimenter

Manuales WEKA

- http://matema.ujaen.es/jnavas/web_recursos/archivos/weka%20master%20recursos%20naturales/TutorialWeka.pdf
  - Para ver ejemplo de uso de Explorer i Experimenter
- https://knowledgesociety.usal.es/sites/default/files/MANUAL%20WEKA.pdf
  - Para ver la parte de ficheros arff y filtros.
- https://oa.upm.es/62854/1/TFG_ALEJANDRO_DOCASAL_GARC%C3%8DA.pdf
  - Para ver una descripción detallada de los clasificadores en Weka. Es un PFG de Alejandro Docasal Garcia, de la Universidad de Politécnica de Madrid (pág. 21-41)

Herramientas

Weka

	Algoritmo	Función
Bayes	*AODE*	Promediado, estimadores de una dependencia.
	BayesNet	Aprender redes Bayesianas.
	NaiveBayes	Clasificador probabilístico estándar *NaiveBayes*.
	NaiveBayesSimple	Implementación de *NaiveBayes* simple
	NaiveBayesUpdateable	Clasificador *NaiveBayes* incremental el cual aprende una instancia a la vez.
Árboles	*ADTree*	Construye árboles de decisión alternativos.
	DecisionStump	Construye árboles de decisión de un nivel.
	Id3	Algoritmo árbol de decisión basado en "divide y vencerás".
	J48	Aprende árbol de decisión C4.5 (C4.5 implementados, revisión 8).
	LMT	Construye árboles de modelo logístico
	NBTree	Construye un árbol de decisión con clasificadores *NaiveBayes* en las hojas.
	RandomForest	Construcción de árboles aleatorios.
	RandomTree	Construir un árbol que considera un número aleatorio de características dadas en cada nodo.
	REPTree	Aprendizaje de árbol rápido que usa la poda en la reducción de errores
	UserClassifier	Deja a los usuarios construir ellos mismos el árbol de decisión.
Reglas	*ConjuntiveRule*	Aprende regla conjuntiva simple.
	DecisionTable	Construye una tabla de decisión simple del clasificador mayoritario.
	JRip	Algoritmo *RIPPER* (poda incremental reducida para producir reducción de error) para rapidez, regla de inducción eficaz
	Nnge	Método vecino más cercano de generación de reglas usando ejemplos generalizados no anidados.
	OneR	Clasificador 1R.
	Part	Obtiene reglas a partir de árboles de decisión construidos usando J4.8.
	Prism	Algoritmo de cobertura simple para reglas.
	Ridor	Regla de aprendizaje ondular hacia abajo.
	ZeroR	Predice la clase mayoritaria (si es nominal) o el valor promedio (si es numérico).
Funciones	*Logistic*	Construye modelos de regresión logística lineal.
	MultilayerPerceptron	Red neuronal de propagación hacia atrás
	RBFNetwork	Implementa una red de función radial básica.
	SimpleLogistic	Construye modelos de regresión logística lineal con selección de atributo incorporado.
	SMO	Algoritmo de optimización mínimo secuencial para soporte de clasificación de vectores.
	VotedPerceptron	Algoritmo *perceptron* votado.
	Winnow	Perceptron motivado a error con actualizaciones múltiples.
Perezosos	*IB1*	Aprendizaje basado en instancia un vecino más cercano básico.
	IBk	Clasificador k vecino más cercano.
	KStar	Vecino más cercano con función de distancia generalizado.
	LBR	Clasificador de Reglas Bayesianas Perezosas.
	LWL	Algoritmo general para aprendizaje localmente pesado.
Meta	*AdaBoostM1*	Aumentar usando el método AdaBoostM1
	Bagging	Un clasificador bolsa (bag), trabaja por regresión también.
	MultiBoostAB	CombinaKboostin y bagging usando el método *MultiBoosting*
	MultiClassClassifier	Usa un clasificador de dos clases para conjuntos de datos multiclases.
	Stacking	Combina varios clasificadores usando el método apilado (*stacking*).
	StackingC	Versión más eficiente de *stacking*.
	Vote	Combina clasificadores usando promedio de estimados de probabilidad o predicciones numéricas.

Tabal resumen algoritmos de clasificación WEKA. Fuente: Universidad de las Ciencias informáticas. La Habana, 2015

Herramientas

Python

Aquí podemos ver algunos de los paquetes y funciones que usaremos para crear y predecir modelos con Python

from **sklearn.model_selection** import **train_test_Split**

from **sklearn.linear_model** import **LogisticRegression**
https://aprendeia.com/algoritmo-regresion-logistica-machine-learning-teoria/

from **sklearn.svm** import **SVC**
https://aprendeia.com/maquinas-vectores-de-soporte-clasificacion-teoria/

from **sklearn.neighbors** import **KNeighborsClassifier**
https://aprendeia.com/algoritmo-k-vecinos-mas-cercanos-teoria-machine-learning/

from **sklearn.tree** import **DecisionTreeClassifier**
https://aprendeia.com/arboles-de-decision-clasificacion-teoria-machine-learning/

from **sklearn** import **metrics**
from **sklearn.tree** import **plot_tree**

Herramientas

Python - tensorflow

Para crear redes neuronales, usaremos el paquete tensorflow:

import tensorflow as tf

Como recurso, hay un buen ejemplo y sencillo de la creación de una red neuronal con tensorflow en el siguiente enlace:
https://www.youtube.com/watch?v=iX_on3VxZzk

Funciones de activación de neuronas : La función de activación se encarga de devolver una salida a partir de un valor de entrada, normalmente el conjunto de valores de salida en un rango determinado como (0,1) o (-1,1). Se buscan funciones que las derivadas sean simples, para minimizar con ello el coste computacional.

Sigmoid – Sigmoide: La función sigmoide transforma los valores introducidos a una escala (0,1), donde los valores altos tienen de manera asintótica a 1 y los valores muy bajos tienden de manera asintótica a 0.

- Características de la función sigmoide:
  - Satura y mata el gradiente.
  - Lenta convergencia.
  - No esta centrada en el cero.
  - Esta acotada entre 0 y 1.
  - Buen rendimiento en la última capa.

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

Python - tensorflow

Funciones de activación de neuronas

Tanh – Tangent Hyperbolic – Tangente hiperbólica: La función tangente hiperbólica transforma los valores introducidos a una escala (-1,1), donde los valores altos tienen de manera asintótica a 1 y los valores muy bajos tienden de manera asintótica a -1.

- Características de la función tangente hiperbólica:
  - Muy similar a la signoide
  - Satura y mata el gradiente.
  - Lenta convergencia.
  - Centrada en 0.
  - Esta acotada entre -1 y 1.
  - Se utiliza para decidir entre uno opción y la contraria.
  - Buen desempeño en redes recurrentes.

$$f(x) = \frac{2}{1 + e^{-2x}} - 1$$

ReLU – Rectified Lineal Unit: La función ReLU transforma los valores introducidos anulando los valores negativos y dejando los positivos tal y como entran.

- Características de la función ReLU:
  - Activación Sparse – solo se activa si son positivos.
  - No está acotada.
  - Se pueden morir demasiadas neuronas.
  - Se comporta bien con imágenes.
  - Buen desempeño en redes convolucionales.

$$f(x) = \max(0, x) = \begin{cases} 0 \; for \;\; x < 0 \\ x \; for \;\; x \geq 0 \end{cases}$$

Python - tensorflow

Funciones de activación de neuronas

Leaky ReLU – Rectified Lineal Unit: La función Leaky ReLU transforma los valores introducidos multiplicando los negativos por un coeficiente rectificativo y dejando los positivos según entran.

- Características de la función Leaky ReLU:
  - Similar a la función ReLU.
  - Penaliza los negativos mediante un coeficiente rectificador.
  - No está acotada.
  - Se comporta bien con imágenes.
  - Buen desempeño en redes convolucionales.

$$f(x) = \begin{cases} 0 & for\ x < 0 \\ a \cdot x & for\ x \geq 0 \end{cases}$$

Softmax – Rectified Lineal Unit: La función Softmax transforma las salidas a una representación en forma de probabilidades, de tal manera que el sumatorio de todas las probabilidades de las salidas de 1.

- Características de la función Softmax:
  - Se utiliza cuando queremos tener una representación en forma de probabilidades.
  - Esta acotada entre 0 y 1.
  - Muy diferenciable.
  - Se utiliza para para normalizar tipo multiclase.
  - Buen rendimiento en las últimas capas.

$$f(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

Python - tensorflow

Funciones pérdida: La función de pérdida, también denominada función objetivo, en esencia, nos dice, durante el proceso de entrenamiento de la red, lo lejos que está en un momento dado, lo que la red nos ofrece como salida y el resultado que nosotros consideramos que es el correcto o deseado. El valor de la función de pérdida será luego un dato de entrada en el algoritmo de aprendizaje.

Dado que los algoritmos de aprendizaje como el descenso de gradiente, calculan la derivada de la función de pérdida, ésta debe ser una función continua y derivable.

- Error Cuadrático Medio ('Mean Square Error', MSE): Una función muy conocida que calcula la distancia 'geométrica' al valor objetivo. Hablar de distancia geométrica es una forma de visualizarlo que nos orienta cuando pensamos en dos o tres dimensiones. Más allá de esas dimensiones, es sólo una forma de entenderlo. Además, decimos distancia, pero en realidad es la distancia elevada al cuadrado. Una variante de ésta seria la que, en lugar del cuadrado de la distancia, elige el valor absoluto ('Absolute Error'). MSE se puede usar, por ejemplo, en problemas de regresión a valores arbitrarios y con una última capa sin función de activación.

- Entropía cruzada Categórica ('Categorical Cross Entropy'): Sin entrar en detalles, la entropía cruzada, en general, es una medida de la distancia entre distribuciones de probabilidad. La entropía cruzada suele ser adecuada en modelos de redes cuya salida representa una probabilidad, como cuando hacemos una clasificación categórica con función de activación 'softmax'. Se puede utilizar, por ejemplo, en problemas de clasificación categórica con una sola etiqueta de salida y precedida de una función de activación 'Softmax'.

- Entropía cruzada binaria ('Binary Cross Entropy'): Una variante de la anterior pero en que tratamos con clasificación binaria y , por tanto, la función de activación sería una sigmoide.

- Entropía Cruzada Categórica Dispersa ('Sparse Categorical Cross Entropy'): Una variante que se suele usar en el caso de trabajar con números enteros.

Python - tensorflow

Datos categóricos en aprendizaje profundo

Los modelos de machine learning y deep learning (aprendizaje profundo), como los de Keras, requieren que todas las variables de entrada y salida sean numéricas. Esto significa que si sus datos contienen datos categóricos, debes codificarlo a números antes de que puedas ajustar y evaluar un modelo.

Una variable categórica es una variable cuyos valores toman el valor de las etiquetas. Por ejemplo, la variable puede ser "tipo_planta " y puede tomar los valores "setosa", "versicolor" y "virgínica". A veces, los datos categóricos pueden tener una relación ordenada entre las categorías, como " primero ", " segundo " y " tercero ". Este tipo de datos categóricos se conoce como ordinal y la información de pedido adicional puede ser útil.

Las tres técnicas más populares son una:

- Codificación de enteros: cada etiqueta única se asigna a un entero
- Codificación en caliente: cada etiqueta se asigna a un vector binario
- Inclusión aprendida: se aprende una representación distribuida de las categorías

Presentación de conclusiones

Es importante que una vez comprobados varios algoritmos, sepamos explicar el porque de la bondad de uno o de otro.

Tendremos que explicar:

- Análisis exploratorio de los datos
- Evaluación de los datos y los distintos modelos
- Resumen ejecutivo del trabajo: resultados y conclusiones.

Aquí os dejo un buen ejemplo de presentación de la evaluación de datos, resultados y conclusiones. Es un PFG de Alejandro Docasal Garcia, de la Universidad Politécnica de Madrid (pág. : 41-50 y capítulo 4)

https://oa.upm.es/62854/1/TFG_ALEJANDRO_DOCASAL_GARC%C3%8DA.pdf

IFCD66 – Data Science
Ejercicios

Ejercicios – Evaluación de resultados

Evaluación

Como interpretar los resultados en WEKA de una regresión

- Coeficiente de correlación: este indica la fuerza y la dirección de la relación lineal entre las variables independientes y dependiente.
- Error absoluto medio: Este indica la diferencia promedio entre los valores predecidos y los valores reales. Un valor bajo indica un buen ajuste del modelo.
- Error cuadrático medio: Este indica la diferencia promedio entre los valores predecidos y los valores reales, pero tiene en cuenta las desviaciones extremas. Un valor bajo indica un buen ajuste del modelo.
- Error absoluto relativo: Este indica el porcentaje de error absoluto promedio entre los valores predecidos y los valores reales. Un valor bajo indica un buen ajuste del modelo.
- Error cuadrático relativo: Este indica el porcentaje de error cuadrático promedio entre los valores predecidos y los valores reales. Un valor bajo indica un buen ajuste del modelo.

Además es importante considerar la cantidad de datos y si los datos están balanceados, para asegurar que el modelo no esté sobreajustado.

Ejercicios – Evaluación de resultados

Evaluación

Como interpretar los resultados en WEKA de una regresión

Por ejemplo, en el ejercicio M-04, con un algoritmo "SimpleLinearRegression":

Correlation coefficient	0.5542
Mean absolute error	1.9059
Root mean squared error	2.3486
Relative absolute error	83.9797 %
Root relative squared error	83.0345 %

Los resultados indican un ajuste moderado del modelo:

- El coeficiente de correlación es moderado (0.5542) lo que indica una relación lineal moderada entre las variables independientes y la variable dependiente.
- Los errores absoluto medio y cuadrático medio son relativamente altos (1.9059 y 2.3486 respectivamente), lo que indica que hay una gran diferencia entre los valores predecidos y los valores reales.
- Los errores absolutos y cuadráticos relativos también son altos (83.9797% y 83.0345% respectivamente), lo que indica que el modelo no es muy preciso.

Ejercicios – Evaluación de resultados

Evaluación

Como interpretar los resultados en WEKA de una regresión

Si en el ejercicio M-04, usamos un algoritmo "LinearRegression":

Correlation coefficient	0.9279
Mean absolute error	0.8443
Root mean squared error	1.0523
Relative absolute error	37.2022 %
Root relative squared error	37.2033 %

Los nuevos resultados un mejor ajuste del modelo en comparación con los resultados previos:

- El coeficiente de correlación es alto (0.9279) lo que indica una relación lineal fuerte entre las variables independientes y la variable dependiente.
- Los errores absoluto medio y cuadrático medio son bajos (0.8443 y 1.0523 respectivamente), lo que indica que hay poca diferencia entre los valores predecidos y los valores reales.
- Los errores absolutos y cuadráticos relativos también son bajos (37.2022% y 37.2033% respectivamente), lo que indica que el modelo es preciso.

Selección de atributos

La selección de atributos es un proceso importante en el campo de la ciencia de datos que tiene como objetivo seleccionar los atributos más relevantes para una tarea específica. La idea detrás de esto es que al utilizar solo los atributos relevantes, se pueden lograr mejores resultados en tareas como la clasificación o la regresión, y al mismo tiempo se reduce el sobreajuste y el tiempo de entrenamiento.

Existen varios métodos para seleccionar atributos, cada uno con sus propias ventajas y desventajas. Algunos ejemplos incluyen:

- Selección de atributos basada en la importancia: este método consiste en calcular una medida de importancia para cada atributo y luego seleccionar los atributos con las medidas más altas. Este método es fácil de implementar y entender, pero puede ser sensible a ruido y redundancia en los datos.
- Selección de atributos basada en la eliminación recursiva: este método consiste en eliminar recursivamente los atributos menos relevantes y evaluar el rendimiento en una tarea específica en cada paso. Este método es más robusto que la selección basada en la importancia, pero es más computacionalmente costoso.
- Selección de atributos basada en la regularización: este método consiste en agregar un término de regularización a un modelo de aprendizaje automático que penaliza los atributos con coeficientes grandes. Este método es eficiente en términos computacionales y puede manejar la selección de atributos y la regularización al mismo tiempo.

En general, la selección de atributos es una técnica esencial en el proceso de análisis de datos y puede ayudar a mejorar significativamente el rendimiento de un modelo. Sin embargo, es importante tener en cuenta que la selección de atributos debe ser utilizada en conjunción con otras técnicas, como la validación cruzada, para garantizar que los resultados sean confiables.

Ejercicios – Selección de atributos

Selección de atributos

Algunos ejemplos de cada tipo de selección de atributos en WEKA:

- Selección de atributos basada en la importancia: el filtro "InfoGainAttributeEval" en WEKA es un ejemplo de una medida de importancia basada en la ganancia de información. También se puede utilizar el filtro "GainRatioAttributeEval" que calcula la razón de ganancia de información.
- Selección de atributos basada en la eliminación recursiva: el filtro "RecursiveFeatureElimination" en WEKA es un ejemplo de un método de eliminación recursiva. Este filtro permite especificar un algoritmo de evaluación y un algoritmo de clasificación para evaluar los atributos.
- Selección de atributos basada en la regularización: el filtro "AttributeSelectedClassifier" en WEKA es un ejemplo de un método de regularización. Este filtro permite especificar un algoritmo de evaluación, un algoritmo de selección de atributos y un algoritmo de clasificación para entrenar un modelo regularizado.

Selección de atributos

Como interpretar los resultados en WEKA de selección de atributos

- Evaluación de atributos. Por ejemplo:
  - Método de búsqueda = Ranker
  - Método de evaluación = InfoGainAttributeEval
- Evaluación de conjuntos de atributos Filter. Por ejemplo:
  - Método de búsqueda = Greedy Stepwise
  - Método de evaluación = CfsSubsetEval
- Método Wrapper. Por ejemplo:
  - Método de búsqueda = Greedy Stepwise
  - Método de evaluación = ClassifierSubsetEval
- Medidas
  - average merit (y su desviación típica): Es la media del valor del atributo en el conjunto de datos. Es una medida de centralidad que indica el valor promedio del atributo en el conjunto de datos. Se refiere a la media de la correlaciones (medidas con InfoGain) en los cinco ciclos de validación cruzada (si usamos 5).
  - average rank (y su desviación típica): Es la media de la clasificación del atributo, donde una clasificación más baja indica que el atributo es más relevante para la tarea de clasificación. Es una medida de importancia relativa del atributo en el conjunto de datos. Se refiere al orden medio en el que quedó cada atributo en cada uno de los cinco ciclos (si usamos 5).
  - attribute: Es el número de atributo en el conjunto de datos. Puede ser utilizado para identificar el atributo correspondiente.

Selección de atributos

Vamos a ver un ejemplo del ejercicio M3-02:

=== Attribute selection 5 fold cross-validation (stratified), seed: 1 ===

Average merit		average rank	attribute
1.386	+- 0.022	1.2 +- 0.4	4 petalwidth
1.368	+- 0.017	1.8 +- 0.4	3 petallength
0.711	+- 0.053	3 +- 0	1 sepallength
0.323	+- 0.059	4 +- 0	2 sepalwidth

Para cada atributo, se proporciona average merit y average rank del valor del atributo en el conjunto de datos, así como el número de atributo (attribute).

- El atributo con el valor de merit **medio más alto** es petalwidth (1.386), seguido de petallength (1.368), sepallength (0.711) y sepalwidth (0.323).
- El atributo con el valor de rank **medio más bajo** es petalwidth (1.2) y petallength (1.8), seguido de sepallength (3) y sepalwidth (4)

Ranker nos ordenaba los atributos, y ha considerado que petalwidth y petallength son los mejores, cosa que ya sabíamos desde el principio, cuando visualizamos en la pestaña de Preprocess, el desglose de los valores de cada atributo por clase.

Por ejemplo, como para sepallength y sepalwidth, la desviación típica es de cero, eso quiere decir que sepallength quedó como tercer atributo en los cinco ciclos de validación cruzada, y que sepalwidth quedó siempre el cuarto. Petalwidth y petallength debieron de quedar ambos a veces primero y a veces segundo, por eso el orden medio es de 1.2. En resumen, petalwidth y petallength son los mejores atributos, a bastante diferencia de los 2 siguientes. Sepalwidth se ve que es particularmente malo.

Ejercicios

Conceptos

EJERCICIO M3-2:

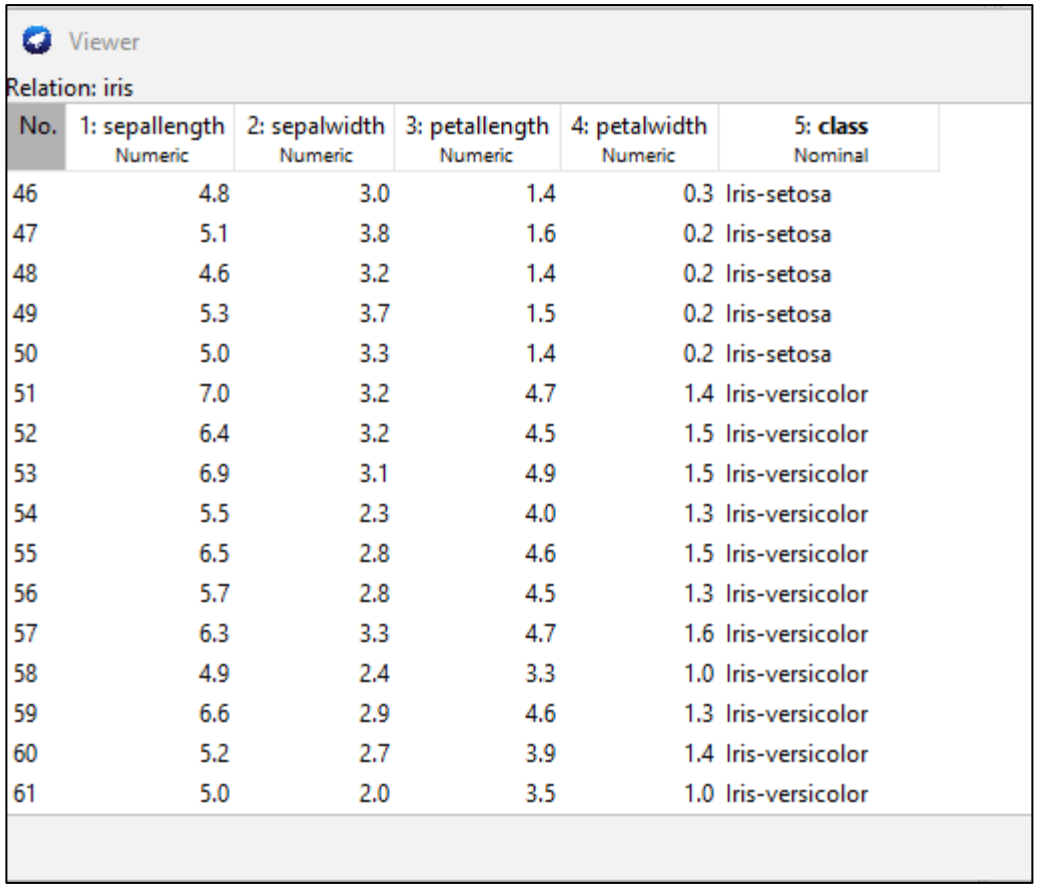
Vamos a usar un ejemplo con datos que nos clasifican el tipo de iris de plantas según la longitud y anchura de sus pétalos y sépalos.

Usaremos el archivo : "iris.arff"

Tarea: clasificación

Técnicas: vamos a probar varias, a ver cual nos da un mejor resultado…

- Reglas:
  - OneR
  - ZeroR
- Árbol:
  - J48
- Red neuronal
- Redactar el documento explicativo del análisis.



Viewer

Relation: iris

No.	1: sepallength Numeric	2: sepalwidth Numeric	3: petallength Numeric	4: petalwidth Numeric	5: class Nominal
46	4.8	3.0	1.4	0.3	Iris-setosa
47	5.1	3.8	1.6	0.2	Iris-setosa
48	4.6	3.2	1.4	0.2	Iris-setosa
49	5.3	3.7	1.5	0.2	Iris-setosa
50	5.0	3.3	1.4	0.2	Iris-setosa
51	7.0	3.2	4.7	1.4	Iris-versicolor
52	6.4	3.2	4.5	1.5	Iris-versicolor
53	6.9	3.1	4.9	1.5	Iris-versicolor
54	5.5	2.3	4.0	1.3	Iris-versicolor
55	6.5	2.8	4.6	1.5	Iris-versicolor
56	5.7	2.8	4.5	1.3	Iris-versicolor
57	6.3	3.3	4.7	1.6	Iris-versicolor
58	4.9	2.4	3.3	1.0	Iris-versicolor
59	6.6	2.9	4.6	1.3	Iris-versicolor
60	5.2	2.7	3.9	1.4	Iris-versicolor
61	5.0	2.0	3.5	1.0	Iris-versicolor

Ejercicios

Conceptos

EJERCICIO M3-3:

Vamos a usar un ejemplo de predicción de partidos dependiendo de las previsiones de tiempo, teniendo en cuenta los hechos pasados.

Viewer

Relation: weather

No.	1: outlook Nominal	2: temperature Numeric	3: humidity Numeric	4: windy Nominal	5: play Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no

Viewer

Relation: weather

No.	1: outlook Nominal	2: temperature Numeric	3: humidity Numeric	4: windy Nominal	5: play Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no

Viewer

Relation: weather.symbolic

No.	1: outlook Nominal	2: temperature Nominal	3: humidity Nominal	4: windy Nominal	5: play Nominal
1	sunny	hot	high	FALSE	no
2	sunny	hot	high	TRUE	no
3	overcast	hot	high	FALSE	yes
4	rainy	mild	high	FALSE	yes
5	rainy	cool	normal	FALSE	yes
6	rainy	cool	normal	TRUE	no
7	overcast	cool	normal	TRUE	yes
8	sunny	mild	high	FALSE	no
9	sunny	cool	normal	FALSE	yes
10	rainy	mild	normal	FALSE	yes
11	sunny	mild	normal	TRUE	yes
12	overcast	mild	high	TRUE	yes
13	overcast	hot	normal	FALSE	yes
14	rainy	mild	high	TRUE	no

Usaremos varios archivos :

- "weather.arff"
- "weather.numeric.arff"
- "weather.nominal.arff"

Tarea: clasificación

Técnica: vamos a probar varias, a ver cual nos da un mejor resultado…

- Reglas
  - OneR
  - ZeroR
- Árbol
  - J48
- Red neuronal
- Redactar el documento explicativo del análisis.

Ejercicios

Conceptos

EJERCICIO M3-4:

Vamos a usar un ejemplo de clasificación de tiendas. El objetivo principal es predecir si una tienda hará ventas altas o no (Sales >8 ->ventas altas) de Sillitas y encontrar características importantes que influyan en las ventas. Habrá que tocar columnas o generar de nuevas ?

Usaremos el archivo : "datos_sillitas_sales.csv".

Tarea: clasificación ?

Técnica: vamos a probar varias, a ver cual nos da un mejor resultado…

- Generar archivo arff (Pentaho,…)
- Reglas
  - OneR
  - ZeroR
- Árbol
  - J48
- Red neuronal
- Redactar el documento explicativo del análisis.

- Sales: Ventas unitarias (en miles) en cada ubicación
- CompPrice: precio que cobra el competidor en cada ubicación
- Income : nivel de ingresos de la comunidad (en miles de dólares)
- Advertising: presupuesto de publicidad local para la empresa en cada ubicación (en miles de dólares)
- Population: tamaño de la población en la región (en miles)
- Price: precio que la empresa cobra por los asientos de seguridad en cada sitio
- ShelveLoc : un factor con niveles Malo, Bueno y Medio que indica la calidad de la ubicación de las estanterías para los asientos de automóvil en cada sitio
- Age: Edad media de la población local
- Education: nivel de educación en cada ubicación
- Urban: un factor con niveles No y Yes para indicar si la tienda se encuentra en una ubicación urbana o rural.
- US: un factor con niveles No y Yes para indicar si la tienda está en EE. UU. O no.

Ejercicios

Conceptos

EJERCICIO M3-5:

Vamos a usar un ejemplo de clasificación de pacientes. El objetivo principal es predecir si un paciente operado de melanoma sobrevivirá, morirá por melanoma o morirá por otras causas.

Usaremos el archivo : "melanoma.csv".

Tarea: clasificación ?

Técnica: vamos a probar varias, a ver cual nos da un mejor resultado…

- Generar archivo arff (Pentaho,…)
- Reglas
  - OneR
  - ZeroR
- Árbol
  - J48
- Red neuronal
- Redactar el documento explicativo del análisis.

time

Survival time in days since the operation, possibly censored.

status

The patients status at the end of the study. 1 indicates that they had died from melanoma, 2 indicates that they were still alive and 3 indicates that they had died from causes unrelated to their melanoma.

sex

The patients sex; 1=male, 0=female.

age

Age in years at the time of the operation.

year

Year of operation.

thickness

Tumour thickness in mm.

ulcer

Indicator of ulceration; 1=present, 0=absent.

Ejercicios

Conceptos

EJERCICIO M3-6: Prognosis de recurrencia de cáncer mamario

Este dominio de cáncer de mama se obtuvo del Centro Médico Universitario, Instituto de Oncología, Ljubljana, Yugoslavia. Gracias a M. Zwitter y M. Soklic por proporcionar los datos

El dataset está compuesto por 286 registros de pacientes que presentaron o no recurrencia de cáncer de mama después de cinco años de una cirugía.

Usaremos el archivo : "breast-cancer.arff". Los nombres originales de las variables, una breve explicación y los tipos de los datos:

- age (rango de edad): 10-19, 20-29, 30-39, 40-49,...
- menopause (momento de la menopausia): lt40, ge40, premeno.
- tumor-size (tamaño del tumor extirpado en mm): 0-4, 5-9, 10-14, ...
- inv-nodes (una métrica de presencia de células cancerosas en los nodos linfáticos): 0-2, 3-5, 6-8, 9-11,...
- node-caps (evidencia de que células cancerosas atravesaron la cápsula de los nódulos linfáticos): yes, no
- deg-malig (grado histológico del tumor: bajo, intermedio, alto): 1, 2, 3.
- breast (mama afectada): left, right.
- breast-quad (cuadrante de la mama): left-up, left-low, right-up, right-low, central.
- irradiat (radioterapia): yes, no.
- Class (clase) Indica recurrencia, es la variable a predecir (no-recurrencia: 201 casos, recurrencia: 85 casos)
- Los datos faltantes están indicados por "?" o por un código "nan"

Ejercicios

Conceptos

EJERCICIO M3-7: Predicción del tipo de vidrio que saldrá

Usaremos el archivo : "glass.arff". Los nombres originales de las variables, una breve explicación y los tipos de los datos:

- Id number: 1 to 214
- RI: refractive index
- Na: Sodium (unit measurement: weight percent in corresponding oxide, as are attributes 4-10)
- Mg: Magnesium
- Al: Aluminum
- Si: Silicon
- K: Potassium
- Ca: Calcium
- Ba: Barium
- Fe: Iron
- Type of glass: (class attribute)
  - -- 1 building_windows_float_processed
  - -- 2 building_windows_non_float_processed
  - -- 3 vehicle_windows_float_processed
  - -- 4 vehicle_windows_non_float_processed (none in this database)
  - -- 5 containers
  - -- 6 tableware
  - -- 7 headlamps

Ejercicios

Conceptos

EJERCICIO M3-8: Predicción de si un paciente tienes diabetes

Usaremos el archivo : "diabetes.arff". Los nombres originales de las variables, una breve explicación y los tipos de los datos:

Todas las variables son numéricas

- Number of times pregnant
- Plasma glucose concentration a 2 hours in an oral glucose tolerance test
- Diastolic blood pressure (mm Hg)
- Triceps skin fold thickness (mm)
- 2-Hour serum insulin (mu U/ml)
- Body mass index (weight in kg/(height in m)^2)
- Diabetes pedigree function
- Age (years)

- Variable de clase binaria (nominal)

Ejercicios

Conceptos

EJERCICIO M3-9: Predicción de si un paciente morirá o vivirá de hepatitis con todos los indicadores presentes.

Usaremos el archivo : "hepatitis.arff" (vigilad los datos por favor…):

- AGE integer
- SEX { male, female}
- STEROID { no, yes}
- ANTIVIRALS { no, yes}
- FATIGUE { no, yes}
- MALAISE { no, yes}
- ANOREXIA { no, yes}
- LIVER_BIG { no, yes}
- LIVER_FIRM { no, yes}
- SPLEEN_PALPABLE { no, yes}
- SPIDERS { no, yes}
- ASCITES { no, yes}
- VARICES { no, yes}
- BILIRUBIN real
- ALK_PHOSPHATE integer
- SGOT integer
- ALBUMIN real
- PROTIME integer
- HISTOLOGY { no, yes}

- Variable de clase binaria (nominal)

Ejercicios

Conceptos

EJERCICIO M3-10: Predicción del colesterol

Usaremos el archivo : "cholesterol.arff". Los nombres originales de las variables, una breve explicación y los tipos de los datos: Podemos predecirlo ? Que variables son las que influyen más ?

- 'age' real
- 'sex' : (1 = male; 0 = female)
- 'cp' : 1: typical angina, 2: atypical angina, 3: non-anginal pain, 4: asymptomatic
- 'trestbps' : resting blood pressure (in mm Hg on admission to the % hospital)
- 'fbs' : (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)
- 'restecg' : resting electrocardiographic (0: normal, 1: having ST-T wave abnormality, 2: showing probable or definite left ventricular hypertrophy
- 'thalach' : maximum heart rate achieved
- 'exang' : exercise induced angina (1 = yes; 0 = no)
- 'oldpeak' : ST depression induced by exercise relative to rest
- 'slope' : the slope of the peak exercise ST segment (1: upsloping, 2: flat, 3: downsloping)
- 'ca' : number of major vessels (0-3) colored by flourosopy
- 'thal' : 3 = normal; 6 = fixed defect; 7 = reversable defect

- Variable de clase numérica (colesterol))

THE CORNER
STUDY CENTRE

Carrer Vicenç Llivina, 08940 Cornellà de Llobregat

www.thecorner.cat

info@thecorner.cat