

Aprendizaje Automático

Tema 2: Regresión y clasificación con modelos lineales y logísticos

Luis Bote Curiel
luis.bote@urjc.es

Grado en Ingeniería de Robótica Software
Curso 2023 – 2024



Universidad
Rey Juan Carlos

Escuela de Ingeniería de Fuenlabrada



Tema 2: Regresión y clasificación con modelos lineales y logísticos

Contenidos

- **Tema 2. Regresión y clasificación con modelos lineales y logísticos**

- 2.1. Regresión lineal.

- 2.1.1. Resolución mediante funciones de alto nivel de `scikit-learn`.

- 2.2. Regresión logística binaria.

- 2.2.1. Resolución mediante funciones de alto nivel de `scikit-learn`.



Tema 2: Regresión y clasificación con modelos lineales y logísticos

Contenidos

- **Tema 2. Regresión y clasificación con modelos lineales y logísticos**

- 2.1. Regresión lineal.

- 2.1.1. Resolución mediante funciones de alto nivel de `scikit-learn`.**

- 2.2. Regresión logística binaria.

- 2.2.1. Resolución mediante funciones de alto nivel de `scikit-learn`.

Tema 2: Regresión y clasificación con modelos lineales y logísticos

2.1. Regresión lineal

2.1.1. Resolución mediante `scikit-learn`

- Para obtener el modelo de regresión lineal de un conjunto de datos mediante funciones de alto nivel de `scikit-learn` se utiliza el objeto `sklearn.linear_model.LinearRegression`:

```
from sklearn.linear_model import LinearRegression
mdl = LinearRegression().fit(X,Y)
```

- Siendo \mathbf{x} la matriz que contiene los datos de entrada (cada columna es una característica diferente), e \mathbf{y} el vector de salida.

- Consulta más en:

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

Tema 2: Regresión y clasificación con modelos lineales y logísticos

2.1. Regresión lineal

2.1.1. Resolución mediante `scikit-learn`

- Para obtener predicciones basadas en el modelo de regresión lineal que se ha creado, se utiliza:

`Ye = mdl.predict(X)`

- Siendo `mdl` el modelo que se ha obtenido previamente, `x` la matriz de datos de entrada e `Ye` la estimación de los datos de salida.



Tema 2: Regresión y clasificación con modelos lineales y logísticos

Contenidos

- **Tema 2. Regresión y clasificación con modelos lineales y logísticos**

- 2.1. Regresión lineal.

- 2.1.1. Resolución mediante funciones de alto nivel de `scikit-learn`.

- 2.2. Regresión logística binaria.

- 2.2.1. Resolución mediante funciones de alto nivel de `scikit-learn`.**

Tema 2: Regresión y clasificación con modelos lineales y logísticos

2.2. Regresión logística binaria

2.2.1. Resolución mediante scikit-learn

- Para obtener el modelo de regresión logística binaria utilizando funciones de alto nivel de scikit-learn se utiliza el objeto `sklearn.linear_model.LogisticRegression`:

```
from sklearn.linear_model import LogisticRegression
mdl = LogisticRegression(penalty="none").fit(X,Y)
```

- **X** se corresponde con la matriz de los datos de entrada (características por columnas) e **Y** es el valor de salida binario (0 ó 1).
- Consulta más en:

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Tema 2: Regresión y clasificación con modelos lineales y logísticos

2.2. Regresión logística binaria

2.2.1. Resolución mediante `scikit-learn`

- En este caso, se pueden utilizar los métodos `predict` y `predict_proba` para aplicar a los datos de entrada (ya sean los de entrenamiento o nuevos) el modelo que se ha generado:

```
Ye = mdl.predict(X)
Ye_proba = mdl.predict_proba(X)
```

- Siendo `mdl` el modelo de regresión logística, `X` la matriz de datos de entrada, `Ye` el vector resultante con las clases de aplicar el modelo a los datos de entrada e `Ye_proba` el vector resultante con las probabilidades de la clase positiva de aplicar el modelo a los datos de entrada.