Profundiza más

Recurso de Profundización

Carga de *dataset* para regresión lineal simple:
En este ejemplo se carga un *dataset* para generar un problema básico de regresión en el que se tiene solo una variable de entrada.

```
# dataset para regresion simple|
from sklearn.datasets import make_regression

plt.title('Sample regression problem with one input variable')
X_R1, y_R1 = make_regression(n_samples = 100, n_features=1,
                             n_informative=1, bias = 150.0,
                            noise = 30, random_state=0)
```

Implementación regresión con k–vecinos más cercanos
Este es un ejemplo de implementación de k–vecinos más cercanos para regresión usando el *dataset* cargado en el ejemplo anterior:

Implementación *ridge regression*
Implementación de *rigde regression* en Python mediante *sklearn:*

```
from sklearn.linear_model import Ridge
X_train, X_test, y_train, y_test = train_test_split(X_crime, y_crime,
                                                    random_state = 0)

linridge = Ridge(alpha=20.0).fit(X_train, y_train)
```

Implementación *Lasso regression*
Implementación de *Lasso regression* en Python mediante *sklearn* y aplicando *MinMaxScaler:*

Profundiza más

```
from sklearn.linear_model import Lasso
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()

X_train, X_test, y_train, y_test = train_test_split(X_crime, y_crime,
                                                    random_state = 0)

X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

linlasso = Lasso(alpha=2.0, max_iter = 10000).fit(X_train_scaled, y_train)
```

Implementación *polinomial regression*
Implementación de regresión polinomial en Python mediante *sklearn:*

```
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures

poly = PolynomialFeatures(degree=2)
X_F1_poly = poly.fit_transform(X_F1)

X_train, X_test, y_train, y_test = train_test_split(X_F1_poly, y_F1,
                                                    random_state = 0)
linreg = LinearRegression().fit(X_train, y_train)
```