Universidad Nacional
Autónoma de México

Inteligencia Artificial

Pronóstico (Aplicado en Salud)

Grupo 3

Nombre:
Barreiro Valdez Alejandro

Práctica 9

Profesor: Dr. Guillermo Gilberto Molero Castillo

14 de abril de 2022

Introducción

En la siguiente práctica se creará un modelo de regresión lineal múltiple para los datos de estudios clínicos de pacientes con cáncer de mama. Con dicho modelo se buscará predecir el área del tumor de dichos pacientes. Para ello se utilizarán modelos de prueba y de entrenamiento y se realizará una selección de variables.

Objetivos

Obtener grupos de pacientes con características similares, diagnosticadas con un tumor de mama, a través de clustering jerárquico y particional a partir de estudios clínicos a partir de imágenes digitalizadas de pacientes con cáncer de mama de Wisconsin (WDBC, Wisconsin Diagnostic Breast Cancer)

Desarrollo

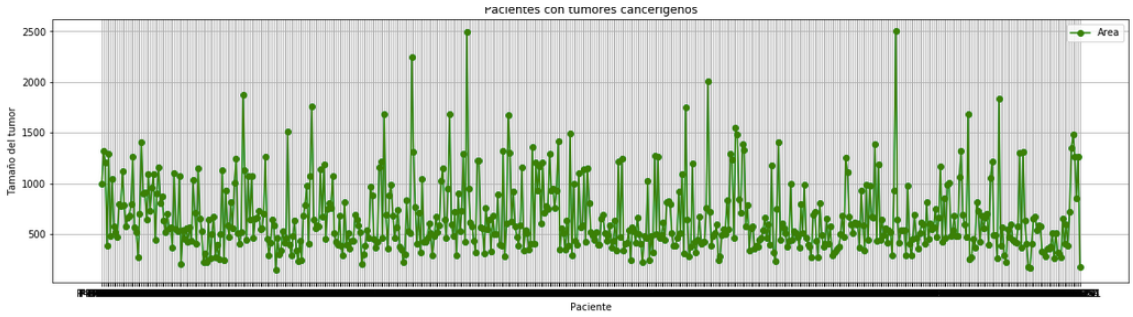
Se importaron las bibliotecas necesarias para el trabajo de la práctica y los datos sobre pacientes con cáncer de mama. Dichos datos ya fueron utilizados en prácticas pasadas.

```
import pandas as pd                 # Para la manipulación y análisis de datos
import numpy as np                  # Para crear vectores y matrices n dimensionales
import matplotlib.pyplot as plt     # Para la generación de gráficas a partir de los datos
import seaborn as sns               # Para la visualización de datos basado en matplotlib
%matplotlib inline
```

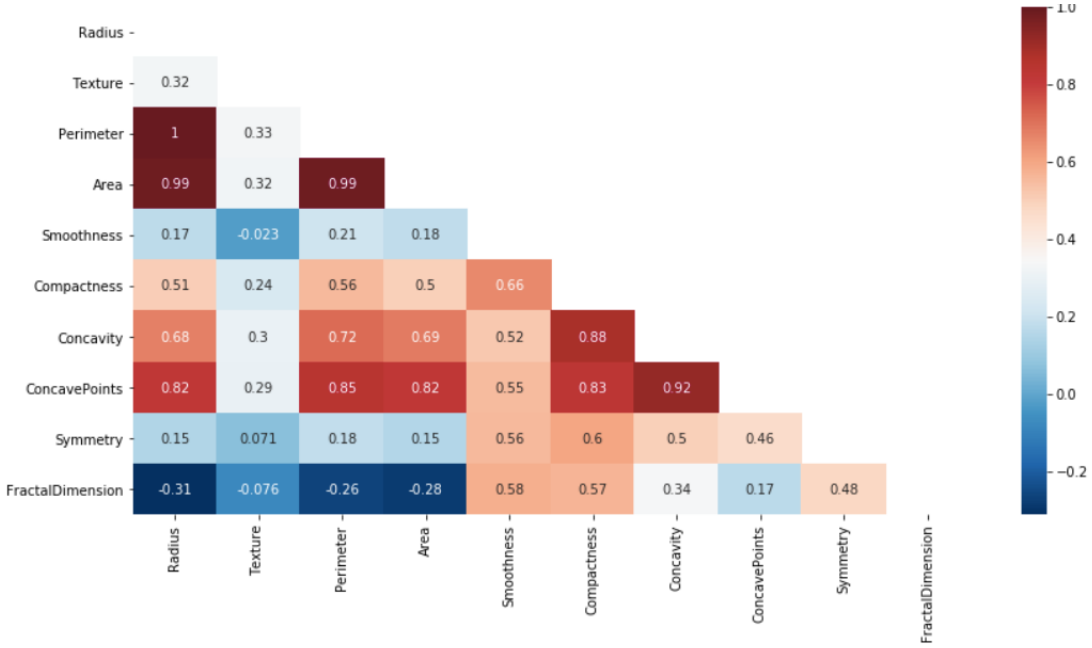
```
BCancer = pd.read_csv('WDBCOriginal.csv')
BCancer
```

	IDNumber	Diagnosis	Radius	Texture	Perimeter	Area	Smoothness	Compactness	Concavity	ConcavePoints	Symr
0	P-842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	0.
1	P-842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	0.
2	P-84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	0.
3	P-84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	0.
4	P-84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	0.
...	...	...	...	...	...	...	...	...	...	...	
564	P-926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	0.
565	P-926682	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	0.

Se generó una gráfica para visualizar el tamaño del tumor por paciente. Dicha gráfica permite observar el tamaño de tumor que tiene cada una de las pacientes en orden por cómo estos registros están acomodados en la tabla.



Se realiza la matriz de correlaciones como en prácticas pasadas para generar la selección de características. A partir de dicha matriz se seleccionan las características de textura, área, smoothness, compactness, symmetry, fractalDimension y perímetro.



Se importan las bibliotecas necesarias para la creación del modelo de regresión lineal múltiple. Este modelo que se generará se hará a partir de los mismos módulos que en la práctica pasada.

```
from sklearn import linear_model
from sklearn.metrics import mean_squared_error, max_error, r2_score
from sklearn import model_selection
```

Para las variables predictoras en este caso se hará uso de aquellas variables que fueron seleccionadas utilizando la matriz de correlaciones. La matriz a predecir será

el área del tumor. Para cada uno de estos grupos de datos se creará un arreglo con las variables pertinentes.

```
X = np.array(BCancer[['Texture',
                      'Perimeter',
                      'Smoothness',
                      'Compactness',
                      'Symmetry',
                      'FractalDimension']])
pd.DataFrame(X)

#['Radius', 'Texture', 'Perimeter', 'Smoothness', 'Compactness', 'Concavity', 'ConcavePoi
```

	0	1	2	3	4	5
0	10.38	122.80	0.11840	0.27760	0.2419	0.07871
1	17.77	132.90	0.08474	0.07864	0.1812	0.05667
2	21.25	130.00	0.10960	0.15990	0.2069	0.05999
3	20.38	77.58	0.14250	0.28390	0.2597	0.09744
4	14.34	135.10	0.10030	0.13280	0.1809	0.05883
...	...	...	...	...	...	...

```
Y = np.array(BCancer[['Area']])
pd.DataFrame(Y)
```

	0
0	1001.0
1	1326.0
2	1203.0
3	386.1
4	1297.0
...	...
564	1479.0
565	1261.0
566	858.1
567	1265.0
568	181.0

569 rows × 1 columns

Con estos arreglos se crearán cuatro grupos de datos para esta práctica. Dos serán los datos utilizados para el entrenamiento del modelo y dos serán para probar dicho modelo. Estos grupos se conocen como *train* y *test*. El tamaño de los datos de prueba será el 20% del tamaño de los datos originales. Se genera cada uno de estos grupos para las variable predicha y las variables predictoras.

```
X_train, X_test, Y_train, Y_test = model_selection.train_test_split(X, Y,
                                                                    test_size = 0.2,
                                                                    random_state = 1234,
                                                                    shuffle = True)
```

Se utiliza el mismo modelo de sklearn que en la práctica pasada. Se genera el modelo utilizando los datos de entrenamiento (*train*). Esta parte se conoce como entrenar el modelo. Posteriormente se utilizaron los datos de prueba con el modelo que se acaba de generar.

```
RLMultiple = linear_model.LinearRegression()
RLMultiple.fit(X_train, Y_train)                #Se entrena el modelo
```

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

A partir de los datos de prueba predictores se generan variables predichas utilizando el modelo recién creado. Estos datos que predice el modelo serán comparados con los valores reales de la agrupación de datos de Y *test*. Con esto se podrá ver qué tan bueno es el modelo.

```
#Se genera el pronóstico
Y_Pronostico = RLMultiple.predict(X_test)
pd.DataFrame(Y_Pronostico)
```

	0
0	405.607887
1	334.291077
2	505.762398
3	207.726058
4	604.229256
...	...
109	394.439214
110	1107.202694
111	541.131191
112	570.702628
113	2044.635054

Se utiliza una métrica de sklearn para ver qué tan parecidos son los datos que se predicen con el modelo a los reales y se tiene una similitud del 97%.

```
r2_score(Y_test, Y_Pronostico)
```

```
0.9769070115972408
```

A partir del modelo que se generó se obtuvieron los datos de los coeficientes, el intercepto, la bondad de ajuste, el residuo, el MSE y RMSE.

```
print('Coeficientes: \n', RLMultiple.coef_)
print('Intercepto: \n', RLMultiple.intercept_)
print("Residuo: %.4f" % max_error(Y_test, Y_Pronostico))
print("MSE: %.4f" % mean_squared_error(Y_test, Y_Pronostico))
print("RMSE: %.4f" % mean_squared_error(Y_test, Y_Pronostico, squared=False))   #True devuel
print('Score (Bondad de ajuste): %.4f' % r2_score(Y_test, Y_Pronostico))
```

```
Coeficientes:
 [[ 6.86261446e-01  1.63885604e+01  2.50787388e+01 -1.40602548e+03
   1.46803422e+02  6.23269303e+03]]
Intercepto:
 [-1140.33616115]
Residuo: 456.3649
MSE: 3083.2634
RMSE: 55.5271
Score (Bondad de ajuste): 0.9769
```

Se puede generar la ecuación del modelo de regresión múltiple utilizando los datos anteriores. La ecuación que define dicho modelo quedaría como:

$$\begin{aligned} Y = {} & -1140.34 + 0.69(\text{Texture}) + 16.39(\text{Perimeter}) + 25.08(\text{Smoothness}) \\ & - 1406.03(\text{Compactness}) + 146.80(\text{Symmetry}) \\ & + 6232.69(\text{FractalDimension}) + 456.36 \end{aligned}$$

La bonda de ajuste de este modelo es de 97.69% por lo que las predicciones de este modelo tiene esa efectividad. Los pronósticos se alejan en promedio 3083.2634 y 55.5271 del valor real. Estos valores corresponden al MSE y al RMSE.

Utilizando el modelo que se generó se predijo el valor del área de un tumor a partir de las variables predictoras que conforman el modelo. Los valores ingresados fueron de Texture: 18.32, Perimeter: 166.82, Smoothness: 0.08142, Compactness: 0.04462, Symmetry: 0.2372 y FractalDimension: 0.05768. El área predicha fue de 1939.80435773.

```
AreaTumorID600 = pd.DataFrame({'Texture': [18.32],
                               'Perimeter': [166.82],
                               'Smoothness': [0.08142],
                               'Compactness': [0.04462],
                               'Symmetry': [0.2372],
                               'FractalDimension': [0.05768]})
RLMultiple.predict(AreaTumorID600)
```

```
array([[1939.80435773]])
```

Conclusiones

Con lo realizado en la práctica se logró generar un modelo capaz de predecir el área de un tumor a partir de ciertas características que se seleccionaron. Se utilizaron datos de entrenamiento y de prueba para entrenar el modelo y posteriormente ponerlo a prueba para ver qué tan acorde con la realidad va. Además, se hicieron uso de todos los procesos que se llevaron a cabo en la práctica pasada para la generación de un modelo de regresión múltiple. Se generó una ecuación que contiene la fórmula para calcular el área y se interpretaron cada una de las características de este modelo. Con este segundo ejemplo se pudo observar otra aplicación de la regresión lineal múltiple y se ve su funcionalidad al tener modelos con muy buena efectividad.