5. Aplicaciones para procesamiento de datos de investigación: estadísticas, correlación, interpolación, graficación.

5.1 Matriz de correlaciones.

En el siguiente ejercicio, se usarán elementos de *Numpy* y *Pandas* para crear una matriz de correlación entre dos variables y se presentará gráficamente utilizando la biblioteca externa *Seaborn*. Este ejercicio está tomado de la página *LikeGeeks* de *Mokhar Ebrahim*. Además, se usará la biblioteca *SciKit-Learn* para importar los datos de la base de datos del Cáncer de Seno. El programa está escrito en forma lineal sin utilizar procedimientos.

Una matriz de correlación es un dato tabular que representa las "correlaciones" entre pares de variables en un dato dado. La matriz de correlación es una importante métrica de análisis de datos que se calcula para resumir los datos a fin de comprender la relación entre las diversas variables y tomar decisiones en consecuencia.

También es un importante paso de preprocesamiento en los conductos de aprendizaje automático para calcular y analizar la matriz de correlación cuando se desea reducir la dimensionalidad de un dato de alta dimensión.

Un coeficiente de correlación es un número que denota la fuerza de la relación entre dos variables.

Hay varios tipos de coeficientes de correlación, pero el más común de todos ellos es el coeficiente de *Pearson* denotado por la letra griega ρ (*rho*).

Se define como la covarianza entre dos variables dividida por el producto de las desviaciones estándar de las dos variables.

$$\rho(X,Y) = \frac{COV(X,Y)}{\sigma_X \sigma_Y}$$

Donde la covarianza entre X e Y se define además como el "valor esperado del producto de las desviaciones de X e Y de sus respectivas medias".

$$COV(X,Y) = E[(X - \mu_X)(Y - \mu_Y)]$$

Por lo que la fórmula de la correlación de *Pearson* se convertirá en:

$$\rho(X,Y) = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

El valor de ρ se encuentra entre -1 y +1.

Programa:

```
import numpy as np
from sklearn.datasets import load_breast_cancer
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
print("Se importa la matriz de la base de datos del Cáncer de Seno")
breast_cancer=load_breast_cancer()
data=breast_cancer.data
features=breast_cancer.feature_names
df=pd.DataFrame(data,columns=features)
print("\nForma del DataFrame:")
print(df.shape)
print("\nResumen de la matriz de datos del Cáncer de Seno")
print(df)
print("\nPropiedades (datos columna):")
print(features)
df_small=df.iloc[:,:6]
correlation_mat=df_small.corr()
corr_pairs=correlation_mat.unstack()
print("\nCorrelación entre pares desempaquetados:")
print(corr_pairs)
sorted_pairs=corr_pairs.sort_values(kind="quicksort")
print("\nSe ordenan los datos pareados:")
print(sorted_pairs)
negative_pairs=sorted_pairs[sorted_pairs<0]
print("\nSe muestran los datos pareados con correlación menor a cero:")
print(negative_pairs)
strong_pairs=sorted_pairs[abs(sorted_pairs)>0.5]
print("\nSe muestran los datos pareados fuertemente correlacionados:")
print(strong_pairs)
cov=np.cov(df_small.T)
print("\nMatriz de covarianza calculada con Numpy:")
print(cov)
stds=np.std(df_small,axis=0)
stds_matrix=np.array([[stds[i]*stds[j] for j in range(6)] for i in range(6)])
print("\nTamaño de la matriz de desviaciones estándar: ",stds_matrix.shape)
new_corr=cov/stds_matrix
print("\nMatriz de correlaciones calculada matemáticamente:")
print(new_corr)
plt.figure(figsize=(18,4))
plt.subplot(1,2,1)
sns.heatmap(correlation_mat,annot=True)
plt.title("Matriz de correlación (Pandas) del Cáncer de Seno")
plt.xlabel("Características del núcleo celular")
plt.ylabel("Características del núcleo celular")
plt.subplot(1,2,2)
sns.heatmap(new_corr,annot=True)
plt.title("Matriz de correlación (Cov Mat) del Cáncer de Seno")
plt.xlabel("Características del núcleo celular")
plt.ylabel("Características del núcleo celular")
plt.show()
```

Corrida:

```
Se importa la matriz de la base de datos del Cáncer de Seno

Forma del DataFrame:
(569, 30)

Resumen de la matriz de datos del Cáncer de Seno
     mean radius  mean texture  ...  worst symmetry  worst fractal dimension
0          17.99         10.38  ...          0.4601                  0.11890
1          20.57         17.77  ...          0.2750                  0.08902
2          19.69         21.25  ...          0.3613                  0.08758
3          11.42         20.38  ...          0.6638                  0.17300
4          20.29         14.34  ...          0.2364                  0.07678
..           ...           ...  ...             ...                      ...
564        21.56         22.39  ...          0.2060                  0.07115
565        20.13         28.25  ...          0.2572                  0.06637
566        16.60         28.08  ...          0.2218                  0.07820
567        20.60         29.33  ...          0.4087                  0.12400
568         7.76         24.54  ...          0.2871                  0.07039

[569 rows x 30 columns]

Propiedades (datos columna):
['mean radius' 'mean texture' 'mean perimeter' 'mean area'
 'mean smoothness' 'mean compactness' 'mean concavity'
 'mean concave points' 'mean symmetry' 'mean fractal dimension'
 'radius error' 'texture error' 'perimeter error' 'area error'
 'smoothness error' 'compactness error' 'concavity error'
 'concave points error' 'symmetry error' 'fractal dimension error'
 'worst radius' 'worst texture' 'worst perimeter' 'worst area'
 'worst smoothness' 'worst compactness' 'worst concavity'
 'worst concave points' 'worst symmetry' 'worst fractal dimension']
```

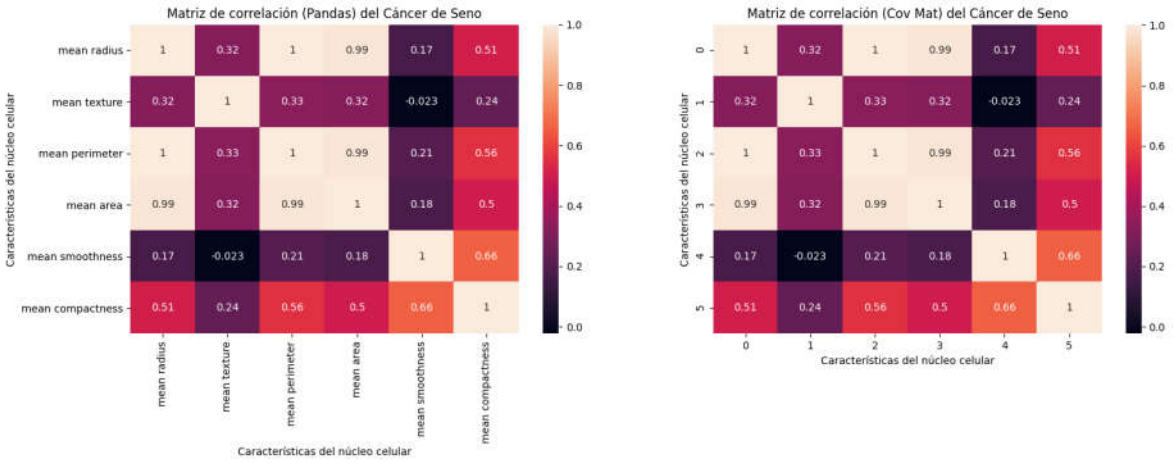
```
Correlación entre pares desempaquetados:
mean radius       mean radius          1.000000
                  mean texture         0.323782
                  mean perimeter       0.997855
                  mean area            0.987357
                  mean smoothness      0.170581
                  mean compactness     0.506124
mean texture      mean radius          0.323782
                  mean texture         1.000000
                  mean perimeter       0.329533
                  mean area            0.321086
                  mean smoothness     -0.023389
                  mean compactness     0.236702
mean perimeter    mean radius          0.997855
                  mean texture         0.329533
                  mean perimeter       1.000000
                  mean area            0.986507
                  mean smoothness      0.207278
                  mean compactness     0.556936
mean area         mean radius          0.987357
                  mean texture         0.321086
                  mean perimeter       0.986507
                  mean area            1.000000
                  mean smoothness      0.177028
                  mean compactness     0.498502
mean smoothness   mean radius          0.170581
                  mean texture        -0.023389
                  mean perimeter       0.207278
                  mean area            0.177028
                  mean smoothness      1.000000
                  mean compactness     0.659123
mean compactness  mean radius          0.506124
                  mean texture         0.236702
                  mean perimeter       0.556936
                  mean area            0.498502
                  mean smoothness      0.659123
                  mean compactness     1.000000
dtype: float64
```

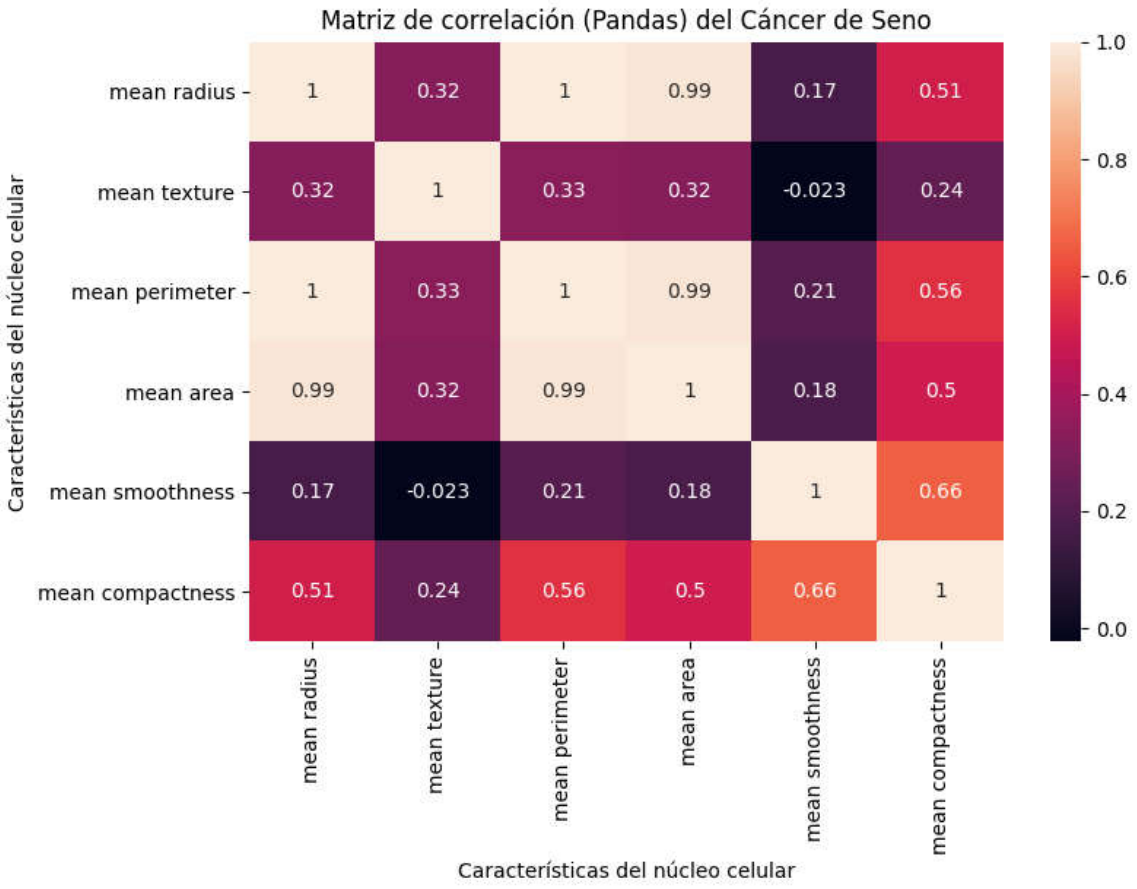
```
Se ordenan los datos pareados:
mean texture       mean smoothness    -0.023389
mean smoothness    mean texture       -0.023389
mean radius        mean smoothness     0.170581
mean smoothness    mean radius         0.170581
mean area          mean smoothness     0.177028
mean smoothness    mean area           0.177028
                   mean perimeter      0.207278
mean perimeter     mean smoothness     0.207278
mean texture       mean compactness    0.236702
mean compactness   mean texture        0.236702
mean area          mean texture        0.321086
mean texture       mean area           0.321086
                   mean radius         0.323782
mean radius        mean texture        0.323782
mean texture       mean perimeter      0.329533
mean perimeter     mean texture        0.329533
mean compactness   mean area           0.498502
mean area          mean compactness    0.498502
mean compactness   mean radius         0.506124
mean radius        mean compactness    0.506124
mean perimeter     mean compactness    0.556936
mean compactness   mean perimeter      0.556936
                   mean smoothness     0.659123
mean smoothness    mean compactness    0.659123
mean perimeter     mean area           0.986507
mean area          mean perimeter      0.986507
                   mean radius         0.987357
mean radius        mean area           0.987357
mean perimeter     mean radius         0.997855
mean radius        mean perimeter      0.997855
                   mean radius         1.000000
mean area          mean area           1.000000
mean perimeter     mean perimeter      1.000000
mean texture       mean texture        1.000000
mean smoothness    mean smoothness     1.000000
mean compactness   mean compactness    1.000000
dtype: float64
```

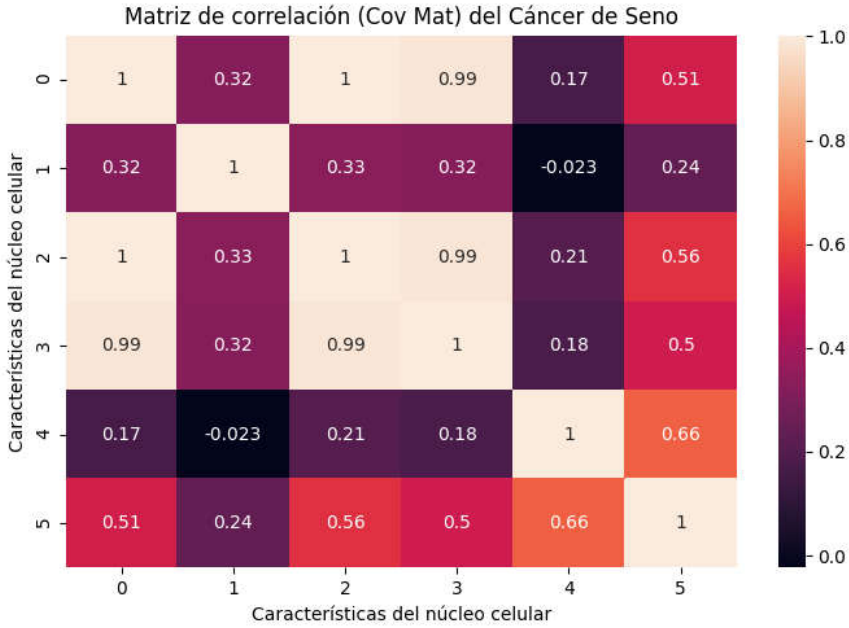
```
Se muestran los datos pareados fuertemente correlacionados:
mean compactness  mean radius         0.506124
mean radius       mean compactness    0.506124
mean perimeter    mean compactness    0.556936
mean compactness  mean perimeter      0.556936
                  mean smoothness     0.659123
mean smoothness   mean compactness    0.659123
mean perimeter    mean area           0.986507
mean area         mean perimeter      0.986507
                  mean radius         0.987357
mean radius       mean area           0.987357
mean perimeter    mean radius         0.997855
mean radius       mean perimeter      0.997855
                  mean radius         1.000000
mean area         mean area           1.000000
mean perimeter    mean perimeter      1.000000
mean texture      mean texture        1.000000
mean smoothness   mean smoothness     1.000000
mean compactness  mean compactness    1.000000
dtype: float64

Matriz de covarianza calculada con Numpy:
[[ 1.24189201e+01  4.90758156e+00  8.54471417e+01  1.22448341e+03
   8.45445983e-03  9.41970568e-02]
 [ 4.90758156e+00  1.84989087e+01  3.44397592e+01  4.85993787e+02
  -1.41477877e-03  5.37668058e-02]
 [ 8.54471417e+01  3.44397592e+01  5.90440480e+02  8.43577235e+03
   7.08360652e-02  7.14714125e-01]
 [ 1.22448341e+03  4.85993787e+02  8.43577235e+03  1.23843554e+05
   8.76178126e-01  9.26493079e+00]
 [ 8.45445983e-03 -1.41477877e-03  7.08360652e-02  8.76178126e-01
   1.97799700e-04  4.89573915e-04]
 [ 9.41970568e-02  5.37668058e-02  7.14714125e-01  9.26493079e+00
   4.89573915e-04  2.78918740e-03]]
Matriz de correlaciones calculada matemáticamente:
[[ 1.00176056  0.32435193  0.99961207  0.98909547  0.17088151  0.50701464]
 [ 0.32435193  1.00176056  0.33011322  0.32165099 -0.02342969  0.23711895]
 [ 0.99961207  0.33011322  1.00176056  0.98824361  0.20764309  0.55791673]
 [ 0.98909547  0.32165099  0.98824361  1.00176056  0.17734005  0.49937933]
 [ 0.17088151 -0.02342969  0.20764309  0.17734005  1.00176056  0.66028364]
 [ 0.50701464  0.23711895  0.55791673  0.49937933  0.66028364  1.00176056]]
```

Gráficas de las matrices de correlación:




Matriz de correlación (Pandas) del Cáncer de Seno
mean radius
mean texture
mean perimeter
mean area
mean smoothness
mean compactness
1 0.32 1 0.99 0.17 0.51
0.32 1 0.33 0.32 -0.023 0.24
1 0.33 1 0.99 0.21 0.56
0.99 0.32 0.99 1 0.18 0.5
0.17 -0.023 0.21 0.18 1 0.66
0.51 0.24 0.56 0.5 0.66 1
Características del núcleo celular
Características del núcleo celular
1.0
0.8
0.6
0.4
0.2
0.0


Matriz de correlación (Cov Mat) del Cáncer de Seno
0 1 0.32 1 0.99 0.17 0.51
1 0.32 1 0.33 0.32 -0.023 0.24
2 1 0.33 1 0.99 0.21 0.56
3 0.99 0.32 0.99 1 0.18 0.5
4 0.17 -0.023 0.21 0.18 1 0.66
5 0.51 0.24 0.56 0.5 0.66 1
0 1 2 3 4 5
Características del núcleo celular
Características del núcleo celular
1.0
0.8
0.6
0.4
0.2
0.0

5.2 Mapas de calor

Un mapa de calor es un gráfico para representar los datos en una forma bidimensional, cuyos valores son asociados a colores proporcionando un resumen visual de la información por el color.

Se usará la instrucción *heatmap()* de la biblioteca *Seaborn* y un arreglo bidimensional de datos aleatorios para generar rápidamente el mapa de calor.

Programa:

```
import numpy as np
import seaborn as sb
import matplotlib.pyplot as py
data=np.random.rand(15,15)
color="RdYlGn"
sb.set(font_scale=0.75)
heat_map=sb.heatmap(data,
                    xticklabels=False,
                    yticklabels=False,
                    cmap=color,
                    annot=True)
py.xlabel("Valores en el eje X")
py.ylabel("Valores en el eje Y")
py.show()
```

Salida: