Estadistica Multivariada
Tarea 1

Carlos Cuauhtemoc Gutierrez Salazar

Fecha de entrega: Martes 31 de Enero de 2023

Ejercicio 1. (30 pts) Demuestre que la matriz de centrado $\mathbf{P}=\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}'$ cumple las siguientes propiedades:

- Tiene rango $(n-1)$, es decir, tiene $n-1$ columnas o renglones linealmente independientes
- Sus valores propios son 1 o 0.

Ejercicio 2. (10 pts) Suponga que $\mathbf{X}$ tiene media cero y matriz de covarianza $\boldsymbol{\Sigma}=\begin{pmatrix}1 & 0\\ 0 & 2\end{pmatrix}$. Sea $\mathbf{Y}=\mathbf{X_1}+\mathbf{X_2}$. Escriba $\mathbf{Y}$ como una transformacion lineal, es decir, encuentre la matriz de transformacion A y calcule $\operatorname{Var}(\mathbf{Y})$

Ejercicio 3. (30 pts) Considere las siguientes funciones:

$$\begin{aligned}
&f_1(x_1,x_2)=4x_1x_2\exp\left(-x_1^2\right) \quad x_1,x_2>0\\
&f_2(x_1,x_2)=2 \quad 0<x_1,x_2<1, \text{ y } x_1+x_2<1\\
&f_3(x_1,x_2)=\frac{1}{2}\exp(-x_1) \quad x_1>|x_2|
\end{aligned}$$

y diga si son pdf's, si es así calcule $E(X),\operatorname{Var}(X),E(X_1\mid X_2)$ y $E(X_2\mid X_1)$

Ejercicio 4. (20 pts) Descargue de moodle los datos denotados como datos_tarea1.dat, los cuales consisten de 42 mediciones de variables relacionadas con la calidad de aire de la ciudad de Los Angeles. Estos dados fueron registrados a las 12:00 de mediodia en diferetes dias. (a) Grafique los diagramas de dispersión marginal para todas las variables. (b) Construya los arreglos $\hat{\mathbf{x}},\mathbf{S}_n$ y $\mathbf{R}$ (interprete las entradas de $\mathbf{R}$).

Ejercicio 5. (10 pts) Agrege y discuta las actividades 1 y 2 vistas en la sesión de taller.

Ejercicio 1.

- Tiene rango $(n-1)$, es decir, tiene $n-1$ columnas o renglones linealmente independientes

Para una matrix idepotente, su rango es igual a la suma de sus elementos en su diagonal principal, esto es

$$\begin{aligned}
&\rightarrow \\
&rank(P) = (1 - \frac{1}{n}) \cdot n \\
&rank(P) = n - 1
\end{aligned}$$

- Sus valores propios son 1 o 0.

$$\begin{aligned}
&det(P - \lambda I) = 0, \\
&det(I - \frac{1}{n} J_n - \lambda I) = 0 \\
&si\ \lambda = 1 \\
&\rightarrow det(-\frac{1}{n} J_n) = 0,
\end{aligned}$$

Por propiedad de determinantes, podemos sacar el valor escalar del calculo de la determinante de la siguiente manera

$$B = kA,\ det(B) = k^n det(A) \tag{1}$$

por lo que, sabemos que

$$\begin{aligned}
&det(-\frac{1}{n} J_n) = 0, \\
&\frac{-1}{n} det(J_n) = 0
\end{aligned}$$

donde podemos observar que se cumple que 0 es un valor propio de la matriz de centrado

Para $\lambda = 0$

$$P - \lambda I = \begin{pmatrix} 1 - \frac{1}{n} & -\frac{1}{n} & \cdots & -\frac{1}{n} \\ -\frac{1}{n} & 1 - \frac{1}{n} & & \vdots \\ \vdots & & \ddots & \vdots \\ -\frac{1}{n} & \cdots & \cdots & 1 - \frac{1}{n} \end{pmatrix}$$

Donde podemos observar que si a una fila especifica sumamos todas las demas filas ($n-1$ filas), obtenemos una fila de ceros. Por ello un valor propio de la matriz de centrado es el 0.

Ejercicio 2.

Buscamos una matriz A que cumpla $AX = Y$. Basta con tomar los valores de cada parametro o X_i, de manera que si

$$\begin{aligned} &Y = X_1 + X_2, \\ &\to AX = \begin{pmatrix} 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = X_1 + X_2 = Y \end{aligned}$$

Ahora, si sabemos que

$$Var(aX + bY) = a^2 Var(X) + b^2 Var(Y) + 2ab \cdot Cov(X, Y) \tag{2}$$

Entonces

$$\begin{aligned} &Var(Y) = Var(X_1 + X_2) = Var(X_1) + Var(X_2) + 2Cov(X_1, X_2) \\ &\to Var(Y) = 1 + 2 + 0 = 3 \end{aligned}$$

Ejercicio 3.

$$f_1(x_1,x_2) = 4x_1x_2\exp\left(-x_1^2\right) \quad x_1,x_2>0$$

- Verificar que sea funcion de probabilidad

$$\int_{x_1}\int_{x_2} f(x_1,x_2)d_{x_1}d_{x_2} = 1$$
$$\rightarrow 1 = \int_{x_1}\int_{x_2} 4x_1x_2e^{-x_1^2}d_{x_2}d_{x_1} = 2\int_0^{\infty}\left[x_1e^{-x_1^2}\right]\left.\tfrac{2x_2^2}{2}\right|_0^{\infty} d_{x_1}$$

Se puede observar que la integral no converge, por lo que no es funcion de probabilidad conjunta

$$f_2(x_1,x_2) = 2 \quad 0<x_1,x_2<1, \text{ y } x_1+x_2<1$$

Vale notar que el comportamiento para ambas x_i es el mismo, por lo que tienen el mismo valor esperado, varianza y funciones marginales.

- Verificar que sea funcion de probabilidad

$$\int_{x_1}\int_{x_2} f(x_1,x_2)d_{x_1}d_{x_2} = 1$$
$$\rightarrow 1 = \int_0^1\int_0^{1-x_1} 2d_{x_2}d_{x_1} = 2\int_0^1(1-x_1)d_{x_1}$$
$$= 2\int_0^1 d_{x_1} - 2\int_0^1 x_1d_{x_1} = 2-1 = 1$$

- Obtener los siguientes valores: $E[X], Var(X), E(X_1|X_2), E(X_2|X_1)$

$$E(X) = \begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix} = \begin{pmatrix}E[X_1]\\ E[X_2]\end{pmatrix}$$
$$E[X_1] = \int_0^1\int_0^{1-x_1} 2x_1d_{x_2}d_{x_1} = 2\int_0^1 x_1d_{x_1} - 2\int_0^1 x_1^2d_{x_1} = 1-\frac{2}{3} = \frac{1}{3}$$
$$\rightarrow E(X) = \begin{pmatrix}1/3\\ 1/3\end{pmatrix}$$

$$Var(X) = \begin{pmatrix} E[(X_1-\mu_1)^2] & E[(X_1-\mu_1)(X_2-\mu_2)] \\ E[(X_2-\mu_2)(X_1-\mu_1)] & E[(X_2-\mu_2)^2]\end{pmatrix}$$
$$\rightarrow E[(X_1-\mu_1)^2] = \int_0^1\int_0^{1-x_1} 2(x_1-1/3)^2d_{x_2}d_{x_1}$$
$$(x_1-1/3)^2\cdot(1-x_1) = -x_1^3+\frac{5}{3}x_1^2-\frac{7}{9}x_1+\frac{1}{9}$$
$$E[(X_1-\mu_1)^2] = 2\int_0^1\left[-x_1^3+\tfrac{5}{3}x_1^2-\tfrac{7}{9}x_1+\tfrac{1}{9}\right]d_{x_1} = 2\left[-\tfrac{1}{4}+\tfrac{5}{9}-\tfrac{7}{18}+\tfrac{1}{9}\right] = \frac{1}{18}$$
$$E[(X_2-\mu_2)(X_1-\mu_1)] = 2\int_0^1\int_0^{1-x_1}(x_1-\frac{1}{3})(x_2-\frac{1}{3})d_{x_2}d_{x_1}$$
$$u = x_2 - 1/3$$
$$d_u = d_{x_2}$$
$$E[(X_2-\mu_2)(X_1-\mu_1)] = \int_0^1 (x_1-\frac{1}{3})\left.(x_2-\tfrac{1}{3})\right|_0^{1-x_1} d_{x_1} = \int_0^1\left[(x-\tfrac{1}{3})(x-\tfrac{2}{3})^2\right]$$
$$= \int_0^1\left[x_1^3-\tfrac{5}{3}x_1^2+\tfrac{7}{9}x_1-\tfrac{1}{9}\right]d_{x_1} = \frac{1}{4}-\frac{5}{9}+\frac{7}{18}-\frac{1}{9} = \frac{1}{36}$$
$$\rightarrow Var(X) = \begin{pmatrix}1/18 & -1/36\\ -1/36 & 1/18\end{pmatrix}$$

$$E(X_1|X_2) = \int_{x_1} f(x_1|x_2)d_{x_1}$$
$$f(x_1|x_2) = \frac{f(x_1, x_2}{f(x_2)}$$
$$f(x_2) = \int_{x_1} f(x_1, x_2)d_{x_2}$$
$$\to f(x_2) = \int_0^(1 - x_2)2d_{x_1} = 2(1 - x_2)$$
$$x_1 + x_2 < 1, 0 < x_1, x_2 < 1$$
$$\to f(x_1|x_2) = \frac{2}{2(1 - x_2)} = \frac{1}{1 - x_2}$$
$$x_1 + x_2 < 1, 0 < x_1, x_2 < 1$$
$$E(X_1|X_2) = \int_0^1 \frac{x_1}{1 - x_2} d_{x_1}$$
$$\to E(X_1|X_2) = \frac{1}{2(1 - x_2)}$$
$$x_1 + x_2 < 1, 0 < x_1, x_2 < 1$$
$$\to E(X_2|X_1) = \frac{1}{2(1 - x_1)}$$
$$x_1 + x_2 < 1, 0 < x_1, x_2 < 1$$

Reitero que el comportamiento para ambas x_i es el mismo, por lo que tienen el mismo valor esperado, varianza y funciones marginales.

$$f_3(x_1, x_2) = \frac{1}{2}\exp(-x_1) \quad x_1 > |x_2|$$

- Verificar que sea funcion de probabilidad

$$\int_{x_1}\int_{x_2} f(x_1, x_2)d_{x_1}d_{x_2} = 1$$
$$\to 1 = \int_0^\infty \int_{-x_1}^{x_1} \frac{1}{2}exp(-x_1)d_{x_2}d_{x_1}$$
$$= \int_0^\infty x_1 exp(-x_1)d_{x_1}(\sim \Gamma(\alpha = 2, \beta = 1))$$
$$= 1$$

- Obtener los siguientes valores: $E[X], Var(X), E(X_1|X_2), E(X_2|X_1)$

$$E(X) = \begin{bmatrix}\mu_1 \\ \mu_2\end{bmatrix} = \begin{bmatrix}E[X_1] \\ E[X_2]\end{bmatrix}$$
$$E[X_1] =$$

Ejercicio 4.

Con las funciones de la libreria de *numpy*, se pueden manejar las matrices sin dificultad, como al usar la funcion para invertir entre otras.

```
import pandas as pd
import seaborn as sns
import numpy as np
from itertools import combinations

df = pd.read_csv(
    'C:\\MCE 2nd\\Estadistica Multivariada\\Tareas\\Datos_tarea1.txt', sep='  '
    , header = None)
df.columns = ["a", "b", "c", "d", "e", "f", "g"]


sns.pairplot(df[:])


x = np.array(df)
x = x.astype(np.float32)

n = x.shape[0]
p = x.shape[1]

# 42 observaciones de un vector aleatorio X = (X_1, X_2, X_3, ...)

x_miu = sum(x)/n

x1 = x.copy()

print("X")
print(x)
for i in range(n):
    x1[i] -= x_miu

print("centralized X")
print(x1)

# Sx = E[(x2-mu_x2)(x3-mu_x3)]
print("to get covariance matrix S")

S = np.zeros((p, p))

for i in range(p):
    S[i, i] = sum(x1[:, i]**2)/n

for i in combinations(range(p), 2):
    # centralized x1
    a = sum(list(map(lambda x: x[0]*x[1], x1[:, [i[0], i[1]]])))/n
    # print(i)
    # print(a)
    S[i[0], i[1]] = a
    S[i[1], i[0]] = a

print(S)

print("to get matriz diagonal D")

D = np.zeros((p, p))
for i in range(p):
    D[i, i] = S[i, i]
print(D)

print("to get correlation matrix R")

D1 = np.linalg.inv(D**0.5)

```

```
R = np.matmul(np.matmul(D1, S), D1)
print(R)

with open("T1log.txt", "w") as f:
    f.write("X\n")
    f.write(str(x))
    f.write("\ncentralized X\n")
    f.write(str(x1))
    f.write("\nto get covariance matrix S\n")
    f.write(str(S))
    f.write("\nto get matriz diagonal D\n")
    f.write(str(D))
    f.write("\nto get correlation matrix R\n")
    f.write(str(R))
```

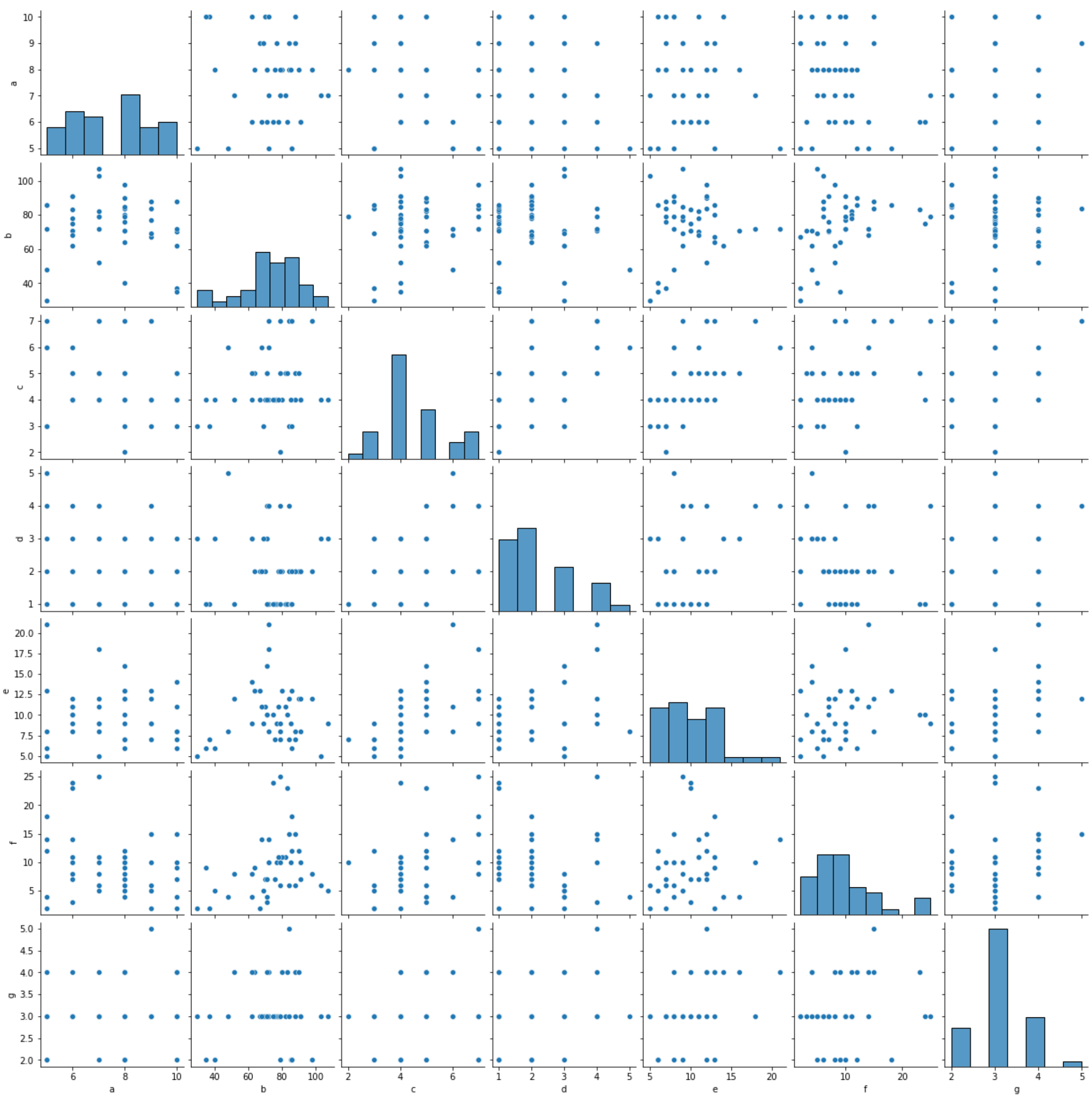

Figura 1: Correlacion muestral en pares de parametros.

Los graficos de correlacion no parecen tener una gran inclinacion de ningun tipo, esto sugiere que puede que no halla una relacion en los parametros de los datos, quiza a excepcion para la segunda variable con la 4ta y la 5ta. Esto se puede visualizar mas facilmente con la matriz de correlaciones.

Vale la pena notar que el maximo valor en la matrix R es de 0,55 $\sim$ entre la variable c y e que asumo corresponden a las variables de CO y NO_3.

https://rpubs.com/bridgetmcgowan/471921

```python
import pandas as pd
import seaborn as sns
df = pd.read_csv(
    'C:\\MCE 2nd\\Estadistica Multivariada\\Tareas\\Datos_tarea1.txt', sep='  '
    , header = None)
df.columns = ["a", "b", "c", "d", "e", "f", "g"]
sns.heatmap(df.corr(), cmap='PuOr')
```

Resultado en consola del codigo: https://github.com/gutierrez310/MCE-EstadisticaMultivariada/blob/main/T1-log.txt

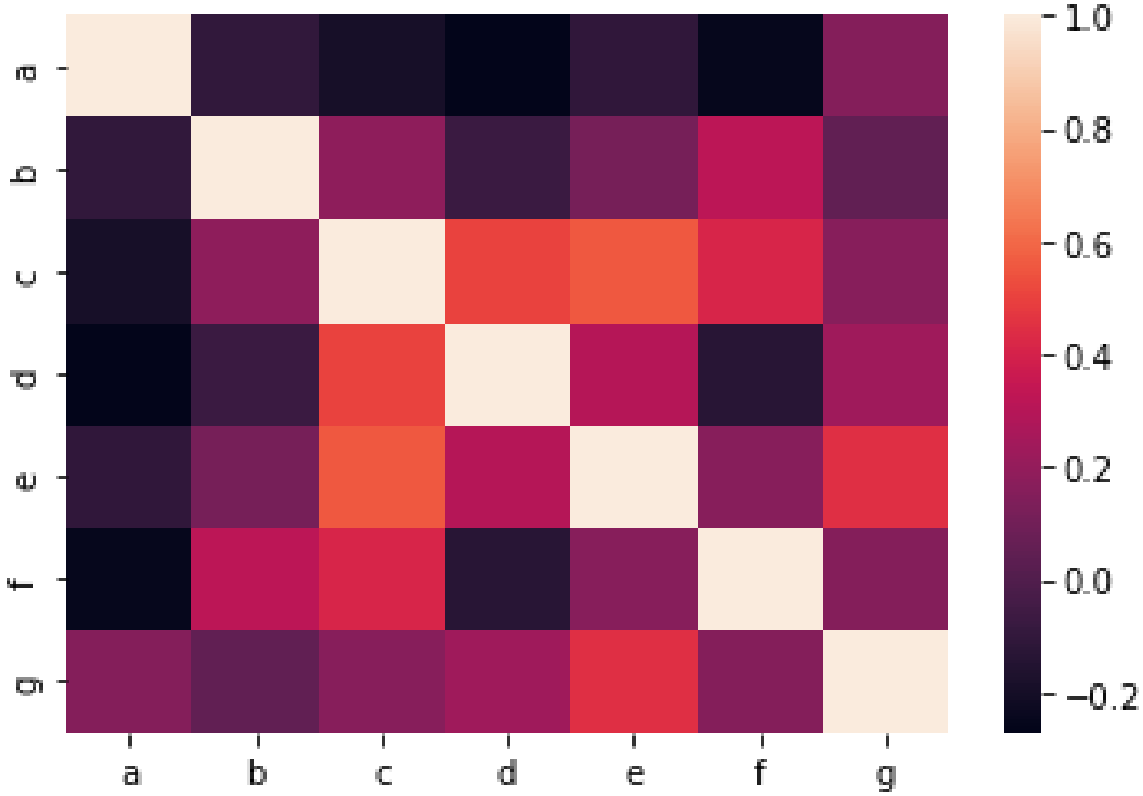

Figura 2: Mapa de calor para la correlacion de los datos

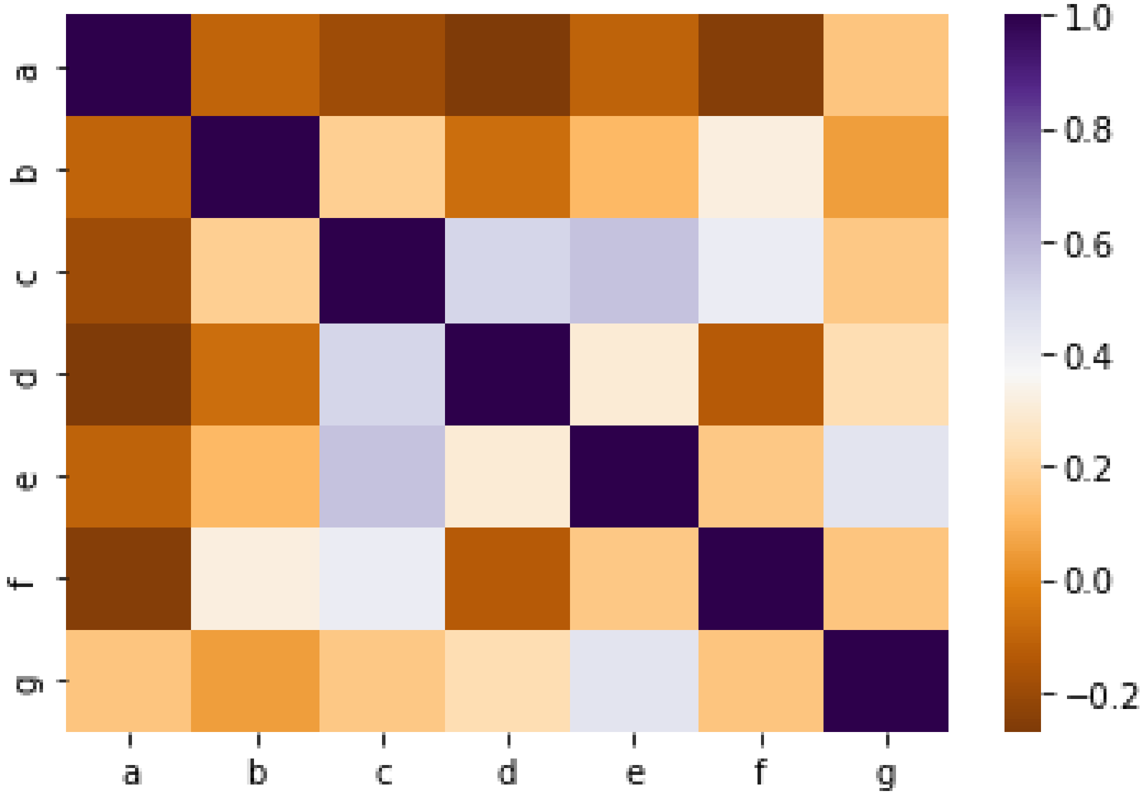

Figura 3: El mismo mapa de calor pero con diferentes colores para facilitar la visualizacion

Ejercicio 5.

- Ejercicio 1

  - (a) Obtenga el vector de medias $\bar{x}$ y la matriz de covarianzas muestrales S_x
  - (b) Obtenga la matriz de covarianzas muestrales de los datos estandarizados a media cero y varianza unidad.
  - (c) Sea el vector aleatorio $\mathbf{Y}= (Y_1, Y_2)$, donde $Y_1 = X_1 + 2X_2X_3$ y $Y_2 = X_1 + X_2$. Calcule la matriz de observaciones de $\mathbf{Y}$ mediante una operacion matricial en la que aparezca la matriz de datos $\mathbf{X}$.
  - (d) Calcule la matriz de covarianzas del vector aleatorio $\mathbf{Z} = (Z_1, Z_2)$, donde $Z_1 = Y_1/\sqrt{6}$ y $Z_2 = Y_2/\sqrt{2}$
  - (e) Calcule las matrices de correlacion de $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$ y de la matriz de datos obtenida en el apartado (b)

$$X = \begin{pmatrix} -2 & 1 & 4 \\ 3 & 0 & -1 \\ 5 & 1 & 2 \\ -1 & 3 & 6 \\ 2 & -7 & 4 \\ -1 & 0 & -1 \end{pmatrix}$$

A decir verdad, el codigo del *Ejercicio 4* primero fue hecho para este ejercicio, por lo que se reusan ciertas lineas de codigo (el maestro menciono que estaba bien usar codigo para estos ejercicios)

```python
import numpy as np
from itertools import combinations

x = np.array([[-2., 1., 4.],
              [3., 0., -1.],
              [5., 1., 2],
              [-1., 3., 6.],
              [2., -7., 4.],
              [-1., 0., -1]])

n = x.shape[0]
p = x.shape[1]

# 6 observaciones de un vector aleatorio X = (X_1, X_2, X_3)

x_miu = sum(x)/n

x1 = x.copy()

print("X")
print(x)
for i in range(n):
    x1[i] -= x_miu

print("centralized X")
print(x1)

print("standarized X")
vars_x = np.var(x, axis=0)*n/(n-1)
for i in range(x.shape[0]):
    x1[i] /= vars_x
print(x1)

# Sx = E[(x2-mu_x2)(x3-mu_x3)]
print("to get covariance matrix S")

x1 = x.copy()
for i in range(n):
```

```
    x1[i] -= x_miu

S = np.zeros((p, p))

for i in range(p):
    S[i, i] = sum(x1[:, i]**2)/n

for i in combinations(range(p), 2):
    # centralized x1
    a = sum(list(map(lambda x: x[0]*x[1], x1[:, [i[0], i[1]]])))/n
    # print(i)
    # print(a)
    S[i[0], i[1]] = a
    S[i[1], i[0]] = a

print(S)

print("to get matriz diagonal D")

D = np.zeros((p, p))
for i in range(p):
    D[i, i] = S[i, i]
print(D)

print("to get correlation matrix R")

D1 = np.linalg.inv(D**0.5)

R = np.matmul(np.matmul(D1, S), D1)
print(R)

# =============================================================================
# R = np.ones((p, p))
#
# for i in combinations(range(p), 2):
#     R[i[0], i[1]] = S[i[0], i[1]] / D[i[0], i[0]]**0.5 / D[i[1], i[1]]**0.5
#     R[i[1], i[0]] = S[i[0], i[1]] / D[i[0], i[0]]**0.5 / D[i[1], i[1]]**0.5
# =============================================================================
print("Y")
A = np.array([[-1, 2, -1],  # relaciones con X
              [1, 1, 0]])

Y = np.matmul(x, np.transpose(A))

print(Y)

times = np.matmul

print("covariance matrix of Y")
S_y = times(times(A, S), np.transpose(A))
print(S_y)

print("to get matriz diagonal D_y")
p = len(A)
D_y = np.zeros((p, p))
for i in range(p):
    D_y[i, i] = S_y[i, i]
print(D_y)

print("to get correlation matrix R_y")

D1 = np.linalg.inv(D_y**0.5)

R_y = np.matmul(np.matmul(D1, S_y), D1)
print(R_y)

print("Z")

B = np.array([[1/(6**0.5), 0],  # relaciones con Y
              [0, 1/(2**0.5)]])

```

```python
Z = np.matmul(Y, np.transpose(B))

print(Z)

print("covariance matrix of Z")
S_z = times(times(B, S_y), np.transpose(B))
print(S_z)

print("to get matriz diagonal D_y")
p = len(B)
D_z = np.zeros((p, p))
for i in range(p):
    D_z[i, i] = S_z[i, i]
print(D_z)

print("to get correlation matrix R_z")

D1 = np.linalg.inv(D_z**0.5)

R_z = np.matmul(np.matmul(D1, S_z), D1)
print(R_z)


print("std X")

x1 = x.copy()
p = 3
for i in range(n):
    x1[i] -= x_miu

for i in range(x.shape[0]):
    x1[i] /= vars_x**0.5
print(x1)

S_x1 = np.zeros((p, p))

for i in range(p):
    S_x1[i, i] = sum(x1[:, i]**2)/n

for i in combinations(range(p), 2):
    a = sum(list(map(lambda x: x[0]*x[1], x1[:, [i[0], i[1]]])))/n
    S_x1[i[0], i[1]] = a
    S_x1[i[1], i[0]] = a

print("covariance matrix of X")
print(S_x1)

D_x1 = np.zeros((p, p))
for i in range(p):
    D_x1[i, i] = S_x1[i, i]

print("to get matriz diagonal D_x")
print(D_x1)

print("to get correlation matrix R_x")

D1 = np.linalg.inv(D_x1**0.5)

R_x1 = np.matmul(np.matmul(D1, S_x1), D1)
print(R_x1)


with open("T1.1-log.txt", "w") as f:
    f.write("X\n")
    f.write(str(x))
    f.write("\ncentralized X\n")
    f.write(str(x1))
    f.write("\nto get covariance matrix S\n")
    f.write(str(S))
    f.write("\nto get matriz diagonal D\n")
    f.write(str(D))
```

```
    f.write("\nto get correlation matrix R\n")
    f.write(str(R))
    f.write("\nY")
    f.write(str(Y))
    f.write("\ncovariance matrix of Y\n")
    f.write(str(S_y))
    f.write("\nto get matriz diagonal D_y\n")
    f.write(str(D_y))
    f.write("\nto get correlation matrix R_y\n")
    f.write(str(R_y))
    f.write("\nZ")
    f.write(str(Z))
    f.write("\ncovariance matrix of Z\n")
    f.write(str(S_z))
    f.write("\nto get matriz diagonal D_y\n")
    f.write(str(D_z))
    f.write("\nto get correlation matrix R_z\n")
    f.write(str(R_z))
    f.write("std X")
    f.write(str(x1))
    f.write("covariance matrix of X")
    f.write(str(S_x1))
    f.write("to get matriz diagonal D_x")
    f.write(str(D_x1))
    f.write("to get correlation matrix R_x")
    f.write(str(R_x1))
```

Vale la pena notar que para el calculo de los valores de **Y** en (c), se hace uso de los siguiente:

$$X \cdot A^T = Y^T$$

Resultado en consola del codigo: https://github.com/gutierrez310/MCE-EstadisticaMultivariada/blob/main/T1.1-log.txt

- Ejercicio 2

  - El diagrama de dispersion de las variables $(x_1, x_2), (x_1, x_3), (x_2, x_3)$
  - $\bar{x}, s_{11}, s_{22}, s_{12}, r_{12}$
  - Varianza total, varianza generalizada, desviacion tipica generalizada

```
import pandas as pd
import seaborn as sns
import numpy as np
from itertools import combinations

df = pd.read_csv(’C:\\MCE 2nd\\datos_ejercicio_semana_1.csv’, sep=’,’
    , header = None)
df.columns = ["a", "b", "c"]

sns.pairplot(df[:])

x = np.array(df)
x = x.astype(np.float32)

n = x.shape[0]
p = x.shape[1]

print("X")
print(x)

print("mean of X")
x_miu = sum(x)/n
print(x_miu)

print("to get covariance matrix S")
```

```python

x1 = x.copy()
for i in range(n):
    x1[i] -= x_miu

S = np.zeros((p, p))

for i in range(p):
    S[i, i] = sum(x1[:, i]**2)/n

for i in combinations(range(p), 2):
    # centralized x1
    a = sum(list(map(lambda x: x[0]*x[1], x1[:, [i[0], i[1]]])))/n
    # print(i)
    # print(a)
    S[i[0], i[1]] = a
    S[i[1], i[0]] = a

print(S)

print("to get matriz diagonal D")

D = np.zeros((p, p))
for i in range(p):
    D[i, i] = S[i, i]
print(D)

print("to get correlation matrix R")

D1 = np.linalg.inv(D**0.5)

R = np.matmul(np.matmul(D1, S), D1)
print(R)

valores_propios, vectores_propios = np.linalg.eig(S)

print("varianza total")
# =============================================================================
# res=0
# for i in range(p):
#     res+=S[i,i]
# print(res)
# =============================================================================
print(total_var := sum(valores_propios))

print("varianza generalizada")
# =============================================================================
# res=1
# for i in valores_propios:
#     res *= i
# print(res)
# =============================================================================
print(gen_var := np.linalg.det(S))

with open("T1.2-log.txt", "w") as f:
    f.write("X\n")
    f.write(str(x))
    f.write("\nmean of X\n")
    f.write(str(x_miu))
    f.write("\nto get covariance matrix S\n")
    f.write(str(S))
    f.write("\nto get matriz diagonal D\n")
    f.write(str(D))
    f.write("\nto get correlation matrix R\n")
    f.write(str(R))
    f.write("\nvarianza total\n")
    f.write(str(total_var))
    f.write("\nvarianza generalizada\n")
    f.write(str(gen_var))
```

Ya que el determinante de la matriz de covarianzas, o su varianza generalizada no es 0, sabemos que ninguna de sus variables tiene dependencia o colinearidad.

La varianza que mayor peso tiene en la varianza total es la de la primera variable o componente, por lo que es el *"primer componente"*

Del grafico de dispersiones junto con la matriz de correlacion podemos observar como entre la variable 1 y variable 3, apodadas (a y , c) en mi codigo, existe una correlacion muy cercana a -1, esto se puede observar en las dispersiones ya que hay una inclinacion en los datos que va del lado superior-izquierdo, al inferior-derecho.

Resultado en consola del codigo: https://github.com/gutierrez310/MCE-EstadisticaMultivariada/blob/main/T1.2-log.txt

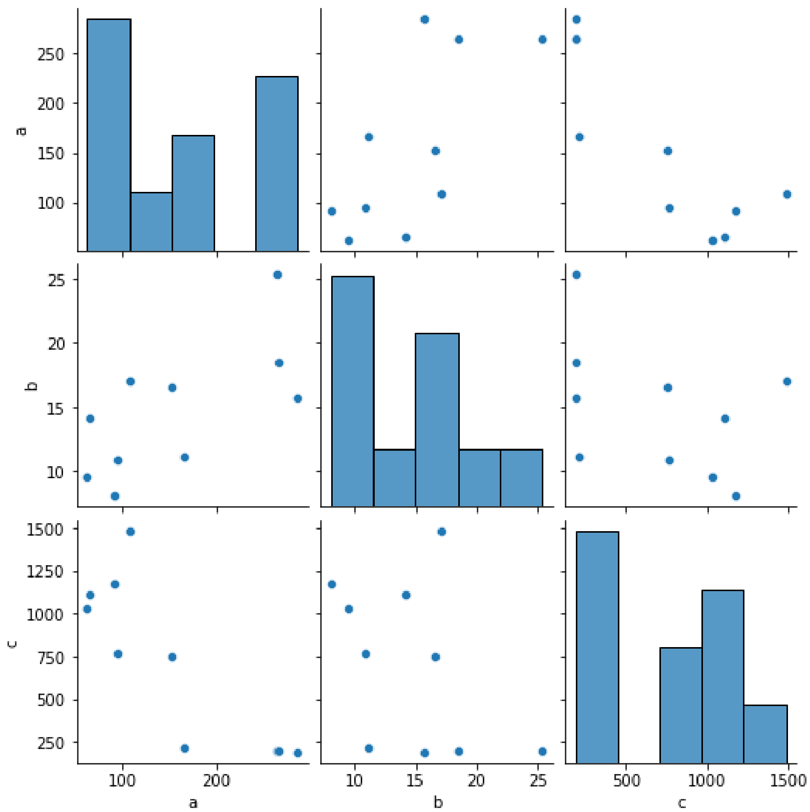

Figura 4: Correlacion muestral en pares de parametros.