Asignacion 3

Juan Diego Giraldo *
Universidad Industrial de Santander
Cra 27 Calle 9

28 de abril 2024

Índice

*e-mail: `Juan Giraldo: juan2181981@correo.uis.edu.co`

Resumen

Los espacios tensoriales son una herramienta matemática poderosa que se puede utilizar para analizar datos complejos de alta dimensionalidad. En este trabajo se exploró la versatilidad del uso de espacios tensoriales en el análisis de dos conjuntos de datos: un sistema de partículas con masa y el gasto en producto interno bruto en áreas como la defensa, la ciencia y tecnología, la salud y la educación. Se discutió cómo los espacios tensoriales pueden ayudar a identificar patrones y relaciones en los datos que pueden ser difíciles de detectar con otras técnicas de análisis de datos. Además, se examinaron las implicaciones prácticas de estas técnicas en estos dos contextos. En resumen, el uso de espacios tensoriales puede ser una herramienta valiosa en situaciones donde los datos son complejos o tienen muchas variables y pueden ser utilizados en una amplia variedad de campos, desde la física hasta la inteligencia artificial.

1. Introducción

En el análisis de datos, es común encontrarse con conjuntos de datos que tienen una alta dimensionalidad, lo que significa que contienen una gran cantidad de variables. En estos casos, puede ser difícil detectar patrones y relaciones entre las variables utilizando técnicas de análisis de datos convencionales. Es aquí donde los espacios tensoriales pueden ser una herramienta valiosa. Los espacios tensoriales son herramientas matemáticas que permiten representar y analizar datos de alta dimensionalidad de manera eficiente, y se utilizan en diversos campos, desde la física hasta la inteligencia artificial. En este trabajo, se explorará la versatilidad del uso de espacios tensoriales en el análisis de dos conjuntos de datos: un sistema de partículas con masa y el gasto en producto interno bruto en áreas como la defensa, la ciencia y tecnología, la salud y la educación. Se examinará cómo los espacios tensoriales pueden ayudar a identificar patrones y relaciones en los datos que pueden ser difíciles de detectar con otras técnicas de análisis de datos, y se discutirán las implicaciones prácticas de estas técnicas en estos dos contextos.
En la Sección 2 discutimos la metodología empleada, mientras que en la Sección 3 se presentan los resultados. Finalizamos el artículo con las conclusiones en la Sección 4,

2. Metodología

Tenemos dos paquetes de datos el primero sera un grupo de partículas con masa en un determinado volumen y el segundo consta de los registros del banco mundial en lo que respecta el gasto del % en PIB sobre diferentes sectores económicos de Colombia.

2.1. Sistema de partículas con masa

2.1.1. 2D

Para los datos de las masas queremos saber en primera instancia sus momentos de variable aleatoria en el eje xy únicamente, entonces se decidió a hacer un código con Python para la resolución de este trabajo.

2.1.2. Orden 0

Iniciamos con la lectura de los datos y tratamiento respectivo para su manipulación usando la librería Pandas (ver1). Inmediatamente se puede encontrar el momento de orden 0 de nuestra variable, la cual al ser masas son la suma de todas estas. Con el siguiente resultado.

Tratamiento de datos 2D

In [2]:

```
data = pd.read_csv('datosmasas.csv')
data.head(0)
df_2d = data
del(df_2d['z'])


X_i_2d = pd.DataFrame()
masas_2d = pd.DataFrame()
X_i_2d['X'] = data['x']
X_i_2d['Y'] = data['y']

masas_2d['X'] = pd.DataFrame(data['masas'])
masas_2d['Y'] = pd.DataFrame(data['masas'])



momento_0_2d = pd.DataFrame()
momento_0_2d = masas_2d['X'].sum()
print(momento_0_2d)
```

```
4627.0
```

Figura 1: Lectura de datos y su tratamiento junto con el momento de orden 0

$$M_0 = \sum \upsilon_i = 4627{,}0 \tag{1}$$

2.1.3. Orden 1

El momento de orden 1 corresponde al promedio pesado de la variable y en nuestro caso seria el centro de masas en el plano xy. Para la resolución del momento de orden 1 se uso la librería Numpy con su respectivo código esta para consultar aquí 2

Momento de orden 1 2D

In [3]:

```
n=len(data['particula'])
media_X_i_2d = pd.DataFrame(X_i_2d.apply(np.sum))
media_X_i_2d =(media_X_i_2d.transpose()/n)
print(media_X_i_2d)

for i in range(0,1533):
    media_X_i_2d.loc[i]= [821.974,775.87]

momento_1_2d = pd.DataFrame()
momento_1_2d = (((masas_2d * X_i_2d).apply(np.sum)))/momento_0_2d

print('El momento de orden 1 o el centro de masas del sistema es:')
print(momento_1_2d)
```

```
El momento de orden 1 o el centro de masas del sistema es:
X    825.815215
Y    776.918522
dtype: float64
```

Figura 2: Codigo del desarrollo del momento de orden 1

$$M_1 = \sum \upsilon_i(|x_i\rangle - |\overline{x}\rangle) = (x, y) = (821{,}973; 775{,}870) \tag{2}$$

Representados ambos valores junto con el conjunto de datos con la Figura 3

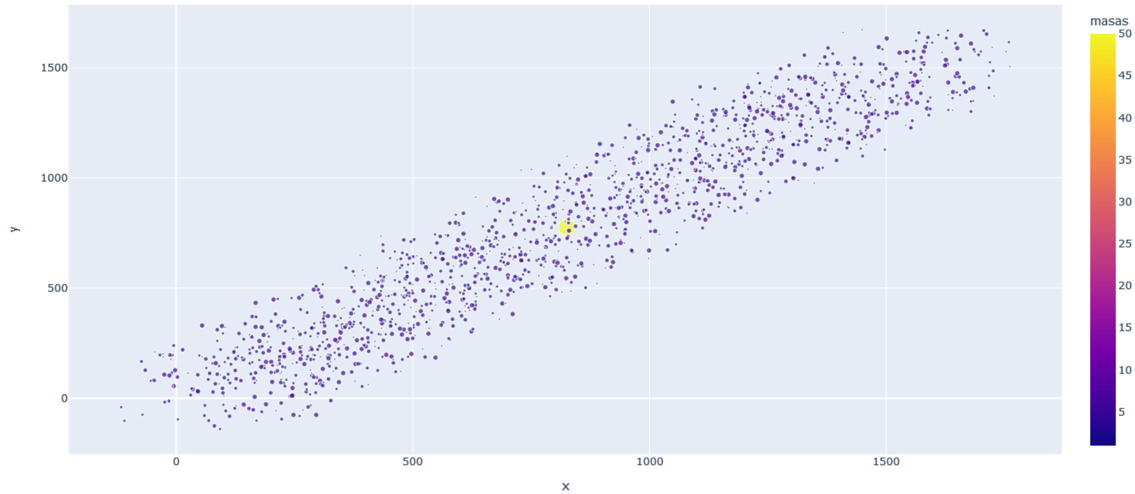


Figura 3: Gráfica de las partículas y su respectiva masa escaladas a su tamaño junto con su centro de masa

2.1.4. Orden 2

Para el momento de orden 2 corresponde al momento de inercia del sistema o la matriz de covarianza del sistema

$$M_2 = \sum v_i(|x_i\rangle - |\overline{x}\rangle)^2 = \begin{bmatrix} 8{,}925904e+08 & 53645{,}826130 \\ 9{,}448749e+08 & 14640{,}500177 \end{bmatrix} \tag{3}$$

Ahora procedemos a calcular los mismos tres momentos pero para el espacio 3D.

2.1.5. 3D

Esta vez se procesaran completamente los datos de la siguiente manera 4

Traemos y Procesamos los DataFrames de la media de los vectores y los vectores

In [142]:
```
data = pd.read_csv('datosmasas.csv')
path = 'GDP% 1960 a 2020.xlsx'
dt = pd.read_excel(path)
df = data
data.head(0)

X_i = pd.DataFrame()
masas = pd.DataFrame()
X_i['X'] = data['x']
X_i['Y'] = data['y']
X_i['Z'] = data['z']

masas['X'] = pd.DataFrame(data['masas'])
masas['Y'] = pd.DataFrame(data['masas'])
masas['Z'] = pd.DataFrame(data['masas'])
```

Figura 4: Código para la lectura y procesamiento de los datos en 3D

2.1.6. Orden 0

de manera muy similar a la forma en 2D en 3D queda de la siguiente manera

$$M_0 = \sum v_i = 4627{,}0 \tag{4}$$

2.1.7. Orden 1

Aplicamos la misma lógica para este caso y obtenemos lo siguiente.

$$M_1 = \sum v_i(|x_i\rangle - |\overline{x}\rangle) = (x, y, z) = (821{,}973; 775{,}870; 15{,}503) \tag{5}$$

Aquí su respectivo código.5

Momento de orden 1 3D

```
In [138]:
momento_1 = pd.DataFrame()
momento_1 =  (((masas) *(X_i)).apply(np.sum))/momento_0

print('El momento de orden 1 o el centro de masas del sistema es:')
print(momento_1)
```

```
El momento de orden 1 o el centro de masas del sistema es:
X    825.815215
Y    776.918522
Z     15.503350
```

Figura 5: Código del momento de orden 1 en 3D

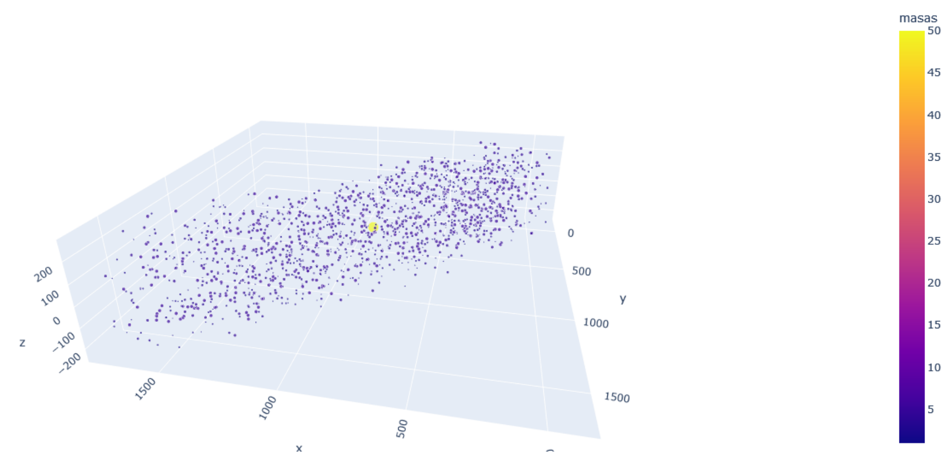


Figura 6: Gráfica de las masas con su centro de gravedad en 3D

Al darle la tercera coordenada podemos observar en 6 que resulta fácil ver la distribución de masa del sistema.

2.1.8. Orden 2

Para el momento de orden 2 corresponde al tensor de inercia del sistema o la matriz de covarianza del sistema realizado con el siguiente 7

$$M_2 = \sum \upsilon_i(|x_i\rangle - |\overline{x}\rangle)^2 = \begin{bmatrix} 9{,}018501e+08 & -4{,}522675e+06 & -4{,}522675e+06 \\ 9{,}546769e+08 & -1{,}591416e+06 & -1{,}591416e+06 \\ -1{,}591416e+06 & 1{,}045176e+08 & 1{,}045176e+08 \end{bmatrix} \tag{6}$$

In [143]:
```
X = np.vstack((X_i['X'], X_i['Y'], X_i['Z'])).T
print(X)
def cov(x,y):
    xbar, ybar = x.mean(), y.mean()
    return (np.sum(((x - xbar)*(y - ybar)))*momento_0/n)


def cov_mat(X):
    return np.array([\
                    [cov(X[0], X[1]), cov(X[0], X[2]), cov(X[0], X[2])]\
                    ,[cov(X[1], X[1]), cov(X[1], X[2]), cov(X[1], X[2])]\
                    ,[cov(X[2], X[1]), cov(X[2], X[2]), cov(X[2], X[2])]\
                    ])
tensor = pd.DataFrame(cov_mat(X.T))
print('El tensor de inecia es:')
print(tensor)
```

```
[[ -53.   79.  -91.]
 [  13. -142. -177.]
 [-109. -103. -120.]
 ...
 [1626. 1537.   96.]
 [1390. 1480.  160.]
 [1717. 1464.  -59.]]
El tensor de inecia es:
              0             1             2
0  9.018501e+08 -4.522675e+06 -4.522675e+06
1  9.546769e+08 -1.591416e+06 -1.591416e+06
2 -1.591416e+06  1.045176e+08  1.045176e+08
```

Figura 7: Código de la matriz de covarianza en 3D

2.1.9. Autovalores y Autovectores

También se no encargo el calculo de los autovalores y autovectores de estas matrices, los cuales fueron calculados usando la función .linalg.eig (ver) 9 la cual nos retorna ambos datos.
En nuestro contexto de la matriz de covarianza, los autovectores y autovalores son útiles para describir la dirección y magnitud de la variabilidad en los datos. Los autovectores son direcciones en el espacio que describen la dirección más importante de la variabilidad, y los autovalores describen la magnitud de la variabilidad en esa dirección. Osea la masa de nuestro sistema de partículas como podemos ver en el siguiente gráfico 8

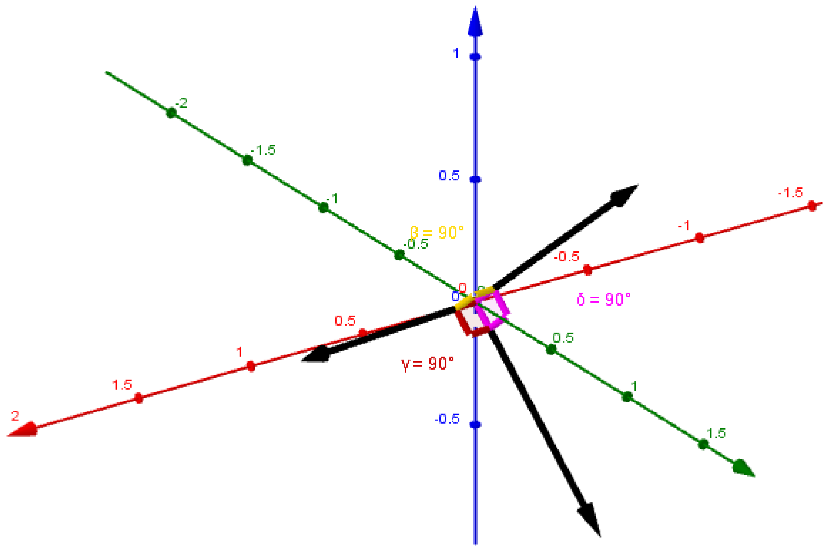


Figura 8: Autovectores de nuestro sistema de particulas

Autovalores y Autovectores 2D

```
In [6]:
Autovec_Autoval_2d = np.linalg.eig(tensor_2d)
print('El Auto vector es:',Autovec_Autoval_2d[0])
print('Los Auto valores son:\n',Autovec_Autoval_2d[1])

El Auto vector es: [ 8.92647232e+08 -4.21450071e+04]
Los Auto valores son:
 [[ 6.86725485e-01 -6.00984398e-05]
 [ 7.26916851e-01  9.99999998e-01]]
```

Figura 9: Código de python para los autovectores y autovalores

2.2. Matriz de covariancia del % en producto interno bruto

Entonces de manera similar se hace el tratamiento de datos de el registro del % de gasto del PIB Colombiano de la siguiente forma 10

```
In [148]:
data2 = pd.read_excel('GDP% 1960 a 20202.xlsx')

X = np.vstack((data2['x'], data2['Military expenditure (% of GDP)'], \
              data2['Government expenditure on education, total (% of GDP)'],\
              data2['Research and development expenditure (% of GDP)'],\
              data2['Current health expenditure (% of GDP)']   ) ).T
```

Figura 10: Código para la lectura y tratamiento de dato del PIB

Consecuentemente calculamos la matriz de covarianza dada por la ecuación 7 y con su código en Python tal que 11

$$M_2 = \sum v_i(|x_i\rangle - |\overline{x}\rangle)^2 \tag{7}$$

Calculo de la Covarianza y la Matriz de Covarianza

```
In [147]:
#Covarianza
def cov(x,y):
    ybar, xbar = y.mean(), x.mean()
    return np.sum((x - xbar)*(y - ybar))/(len(x)- 1)

# Covarianza matrix
def cov_mat(X):
    return np.array([\
                    [cov(X[1], X[1]), cov(X[1], X[2]), cov(X[1], X[3]), cov(X[1], X[4])]\
                    ,[cov(X[2], X[1]), cov(X[2], X[2]), cov(X[2], X[3]), cov(X[2], X[4])]\
                    ,[cov(X[3], X[1]), cov(X[3], X[2]), cov(X[3], X[3]), cov(X[3], X[4])]\
                    ,[cov(X[4], X[1]), cov(X[4], X[2]), cov(X[4], X[3]), cov(X[4], X[4])]\
                                   ])
```

Figura 11: Código Calculando la matriz de covariaza para los datos de

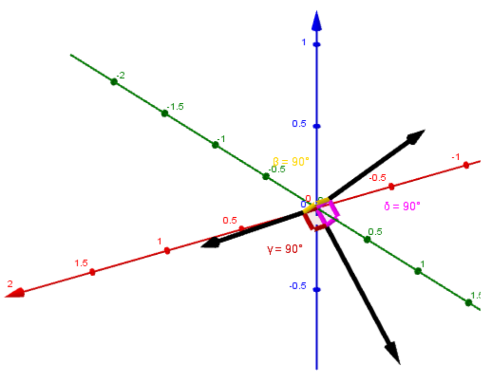
3. Resultados

Para el ejercicio de aplicación sobre el sistema de partículas se llego a los siguientes resultados para los momentos.

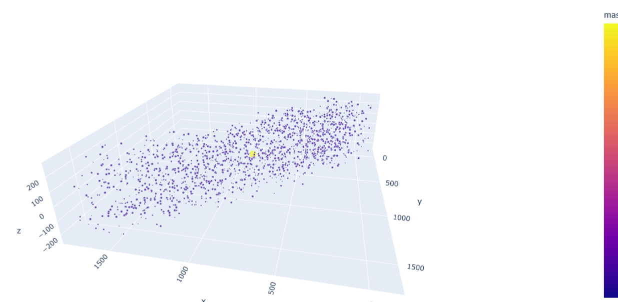
$$M_0 = \sum \upsilon_i = 4627{,}0 \tag{8}$$

$$M_1 = \sum \upsilon_i(|x_i\rangle - |\overline{x}\rangle) = (x, y, z) = (821{,}973; 775{,}870; 15{,}503) \tag{9}$$

$$M_2 = \sum \upsilon_i(|x_i\rangle - |\overline{x}\rangle)^2 = \begin{bmatrix} 9{,}018501e+08 & -4{,}522675e+06 & -4{,}522675e+06 \\ 9{,}546769e+08 & -1{,}591416e+06 & -1{,}591416e+06 \\ -1{,}591416e+06 & 1{,}045176e+08 & 1{,}045176e+08 \end{bmatrix} \tag{10}$$



(a) Gráfica de los autovectores



(b) Gráfica de sistema de partículas

En 12b podemos ver como los autovectores forman una base en donde el sistema de partículas se organiza de manera mas sencilla, dándole sentido a todos nuestros cálculos debido a las propiedades que podemos entender y manipular a partir del tensor de inercia y los autovectores para analizar mecánicamente el sistema de partículas.

El Banco Mundial `https://data.worldbank.org` mantiene una estadística de los datos económicos de casi todos los países. En particular estamos interesados en calcular la matriz de covariancia del % del producto interno bruto (GDP), nosotros vamos a hacer un análisis de estos datos para Defensa, Salud, Educación, Ciencia y Tecnología de la economía Colombiana y como se relacionan entre si mediante los resultados de la matriz de convarianza

Figura 13: Grafica de la matriz de covarianza

Entonces según la gráfica de covarianza 13 nos muestra ciertos resultados que en principio pueden ser confusos pero tienen un sentido y un porque. Nuestra matriz de covarianza es útil para entender la estructura de los datos y las relaciones entre las diferentes variables. la matriz covarianza puede mostrar si hay una relación entre estas variables y cómo fuerte es esa relación. Esta información puede ser útil para hacer inferencias estadísticas, modelar y predecir futuros valores, o reducir la dimensionalidad de los datos.

La covarianza es una medida de cómo dos variables numéricas varían juntas. Si dos variables están positivamente relacionadas, es decir, cuando una aumenta la otra también lo hace, la covarianza es positiva. Si las dos variables están inversamente relacionadas, es decir, cuando una aumenta la otra disminuye, la covarianza es negativa. Si no hay relación entre dos variables, la covarianza es cero.

Figura 14: Gráfica de la matriz de correlacion

Pero existe una limitación la matriz de covarianza se expresa en términos de las unidades originales de las variables, lo que puede hacer que sea difícil comparar las fuerzas de las relaciones entre las diferentes variables. Esto se debe a que la covarianza depende de la escala de las variables y, por lo tanto, las covarianzas más grandes pueden ser el resultado de variables que simplemente tienen valores más grandes en sus unidades originales.

En contraste, la matriz de correlación normaliza la covarianza para que la relación entre las variables esté en una escala común que oscila entre -1 y 1. Esta normalización elimina la influencia de las unidades de medida y hace que sea más fácil comparar la fuerza y dirección de las relaciones entre las variables. Además, los valores en una matriz de correlación son más fácilmente interpretables como un indicador de la relación lineal entre dos variables.

Entonces en la gráfica de la correlación (ver 14 vemos como por ejemplo la correlación que hay con la inversión en gasto militar con el resto de variables es negativa, lo que significa que cuando se aumenta el gasto en defensa disminuye el gasto en educación, salud, ciencia y tecnología.

4. Conclusiones

Los espacios tensoriales son herramientas matemáticas que permiten representar y analizar datos de alta dimensionalidad de manera eficiente. En el caso de un sistema de partículas con masa, las propiedades estadísticas como el momento angular, la energía cinética y la cantidad de movimiento se pueden describir completamente mediante la teoría de espacios tensoriales. Esto permite una comprensión más profunda del sistema, lo que nos puede ser útil en la física teórica y aplicada.

En el caso de los datos de gasto en producto interno bruto, el uso de espacios tensoriales permitió identificar patrones en las interacciones entre las diferentes áreas de gasto. Por ejemplo, se puede descubrir que el gasto en ciencia y tecnología está altamente correlacionado con el gasto en educación, lo que puede tener implicaciones para las políticas gubernamentales que buscan fomentar el crecimiento económico.

La capacidad de los espacios tensoriales para analizar datos de alta dimensionalidad los hace útiles en situaciones donde los datos son complejos o tienen muchas variables. En lugar de examinar cada variable de manera aislada, los espacios tensoriales permiten analizar las relaciones entre las variables, lo que puede revelar patrones y correlaciones más complejas.

Pero el uso de espacios tensoriales puede ser más complejo que otras técnicas de análisis de datos, lo que requiere una comprensión matemática sólida y la capacidad de aplicar técnicas de álgebra lineal. Sin embargo, la flexibilidad y el poder de estas herramientas pueden hacer que valga la pena la inversión de tiempo y recursos necesarios para comprender y aplicarlos efectivamente. En resumen, el uso de espacios tensoriales puede ser una herramienta valiosa para analizar y comprender datos complejos en una amplia variedad de campos

5. Referencias

[1] Hernándes H, Núñez L (2022) *Matemticas avanzadas: de los espacios lineales al anlisis vectorial, con aplicaciones en Maxima*