Task 1: What is covariance?

Covariance is nothing but a measure of correlation. IT indicates the direction of the linear relationship between variables. Covariance can vary between -∞ and +∞.Covariance is affected by the change in scale. If all the values of one variable are multiplied by a constant and all the values of another variable are multiplied, by a similar or different constant, then the covariance is changed.

$Cov(X, Y) = \Sigma [(X_i - X) * (Y_i - \bar{Y})] / (n - 1)$

Difference between Covariance and correlation is covariance measure the relationship between two variables and correlation measure the relationship and strength of two variables together.

Task2: Cross product and dot product

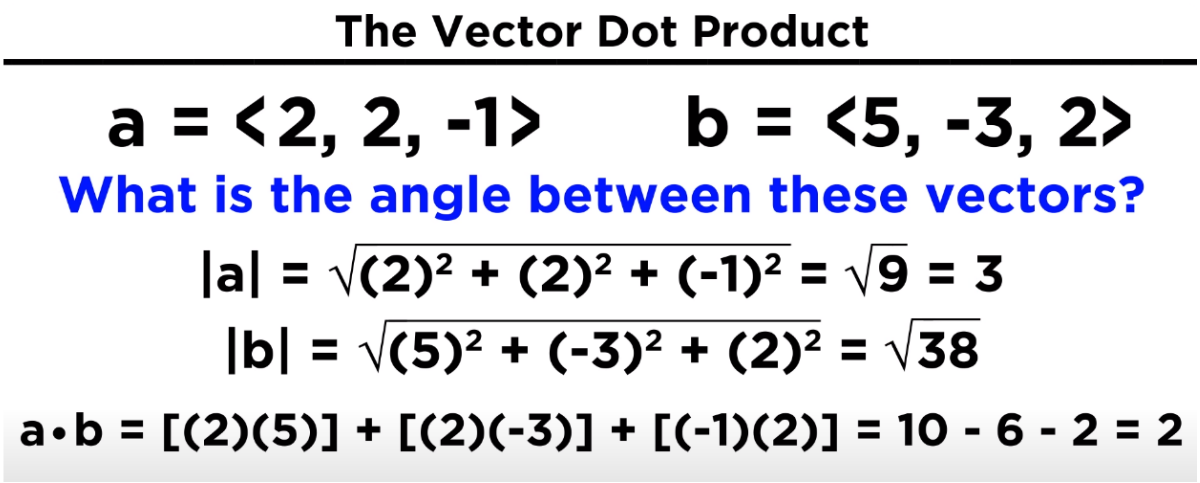
Cross product example:

$$\mathbf{x}_1 \times \mathbf{x}_2 = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 2 & -3 & 1 \\ -2 & 1 & 1 \end{vmatrix}$$

$$= \begin{vmatrix} -3 & 1 \\ 1 & 1 \end{vmatrix}\mathbf{i} - \begin{vmatrix} 2 & 1 \\ -2 & 1 \end{vmatrix}\mathbf{j} + \begin{vmatrix} 2 & -3 \\ -2 & 1 \end{vmatrix}\mathbf{k}$$

$$= [(-3)(1)-(1)(1)]\mathbf{i} - [(2)(1)-(-2)(1)]\mathbf{j} + [(2)(1)-(-2)(-3)]\mathbf{k}$$

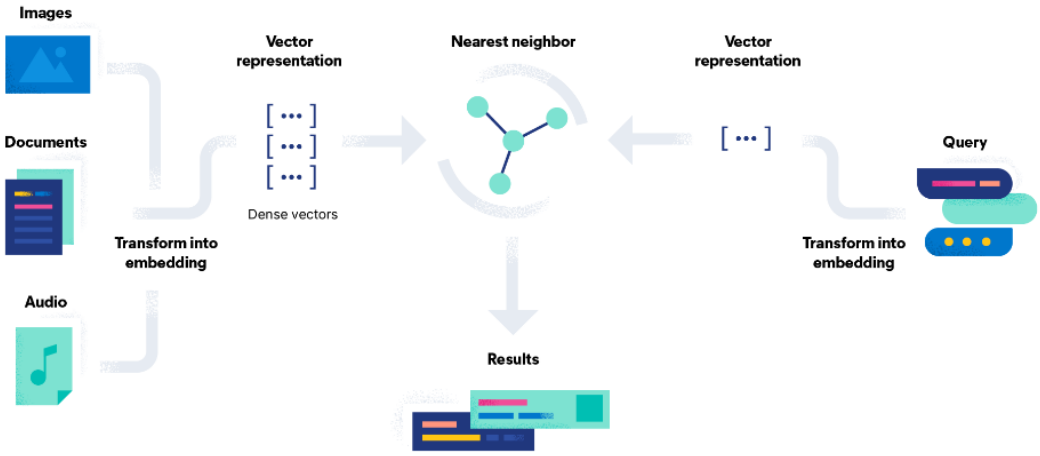
Dot Product example:



Task3: What is vector database

A vector database is a database that stores information as vectors, which are numerical representations of data objects, also known as vector embeddings. It leverages the power of these vector embeddings to index and search across a massive dataset of unstructured data and semi-structured data, such as images, text, or sensor data. Vector databases are built to manage vector embeddings, and therefore offer a complete solution for the management of unstructured and semi-structured data. A vector database is different from a vector search library or vector index: it is a data management solution that enables metadata storage and filtering, is scalable, allows for dynamic data changes, performs

backups, and offers security features. Vector embeddings are a numerical representation of a subject, word, image, or any other piece of data. Vector embeddings — also known as embeddings — are generated by large language models and other AI models.



Task4: Dropping a specific row from DF

```
#df.drop(3,index=0,inplace=True)

df.reset_index()
```

Task5: Read all sheets from excel file

```
df2 = pd.read_excel("stocks_weather.xlsx", sheet_name=None)
```

Task6: Read random rows from dataframe

```
df.sample(n=1)
```

note: n could be any number

Task7: Save 2 columns and last 3 rows in a new csv file

```
df.iloc[-2:,2:4].to_csv("karim.csv",index=False)
```

Task8: how to replace data from dataframe in specific colums with specific values

```
new_df[['windspeed','temperature']]=new_df[['windspeed','temperature']].replace([6,7,32],9000000)
```

Task9: How to detect and remove outliers using pandas

1) `# Load the dataset`
2) `# Create the dataframe`
3) `# Calculate the upper and lower limits`
   `Q1 = df_diabetes['bmi'].quantile(0.25)`
   `Q3 = df_diabetes['bmi'].quantile(0.75)`
4) `IQR = Q3 - Q1`
   `lower = Q1 - 1.5*IQR`
   `upper = Q3 + 1.5*IQR`
5) `# Create arrays of Boolean values indicating the outlier rows`
   `upper_array = np.where(df_diabetes['bmi']>=upper)[0]`

```
lower_array = np.where(df_diabetes['bmi']<=lower)[0]
```

6)
```
df_diabetes.drop(index=upper_array, inplace=True)
df_diabetes.drop(index=lower_array, inplace=True)
```

Task10:Sorting df as index and column

As column:

Df.sort_values(by=[‘col’,’col2’])

As index:

Df.sort_index(axis, ascending,inplace,kind=> quicksort,merge,heapsort,stable, na_position)