Foundations Of Neural Networks and Deep Learning

Day-2

SME: Gagan P

Contact: gaganp000999@gmail.com

recap:

Q1. Which of the following best defines Machine Learning?

A. Writing explicit rules for every possible input
B. Algorithms that learn patterns from data to make predictions
C. A database that stores large amounts of information
D. A computer that can play chess faster than humans

Q1. Which of the following best defines Machine Learning?

A. Writing explicit rules for every possible input
B. **Algorithms that learn patterns from data to make predictions**
C. A database that stores large amounts of information
D. A computer that can play chess faster than humans

Q2. Which of these is not a primary type of machine learning?

A. Supervised

B. Unsupervised

C. Reinforcement

D. Symbolic reasoning

Q2. Which of these is not a primary type of machine learning?

A. Supervised

B. Unsupervised

C. Reinforcement

D. Symbolic reasoning

Q4. In supervised learning, the training data must contain:

A. Only inputs without outputs

B. Both inputs and known labels

C. Reinforcement rewards

D. Unlabeled clusters

Q4. In supervised learning, the training data must contain:

A. Only inputs without outputs
B. Both inputs and known labels
C. Reinforcement rewards
D. Unlabeled clusters

Suppose:

```
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
```

What will np.dot(a, b) return?

A) [4, 10, 18]

B) 32

C) [5, 7, 9]

D) Error

Suppose:

```
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
```

What will np.dot(a, b) return?

A) [4, 10, 18]

B) 32

C) [5, 7, 9]

D) Error

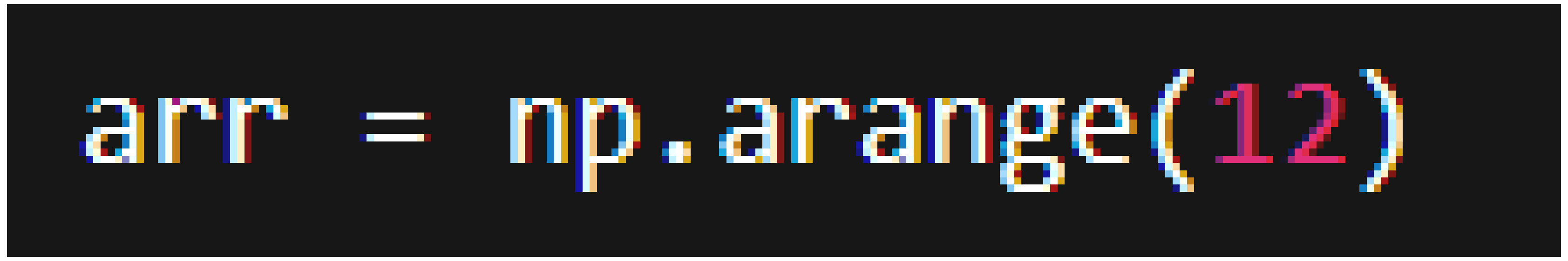
np.dot(a, b) computes the dot product:

$$1 \ast 4 + 2 \ast 5 + 3 \ast 6$$

$$=4+10+18$$

$$=32$$

You have:



```
arr = np.arange(12)
```

Which of the following will not throw an error?

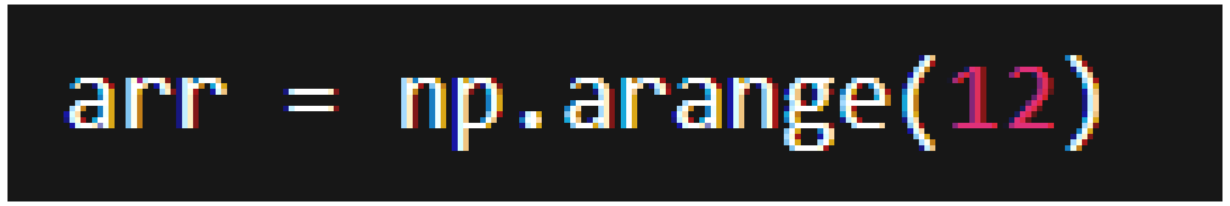
A) arr.reshape(3, 4)

B) arr.reshape(2, 6)

C) arr.reshape(4, 4)

D) arr.reshape(6, 2)

You have:



```python
arr = np.arange(12)
```

Which of the following will throw an error?

A) arr.reshape(3, 4)

B) arr.reshape(2, 6)

C) arr.reshape(4, 4)

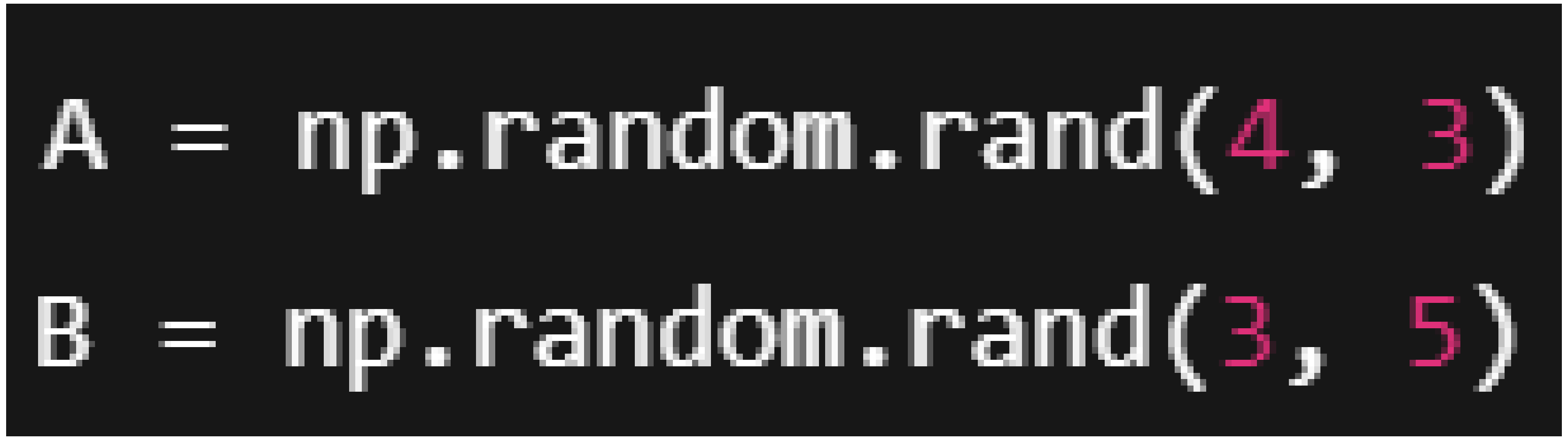
D) arr.reshape(6, 2)

arr=[0, 1, 2, ... 11]   Has 12 Elements

Now we are shaping it into a matrix which should have a total of 12 elements

4x4 matrix has a total of 16 elements, so we will get an error

You have the following NumPy arrays:



```
A = np.random.rand(4, 3)
B = np.random.rand(3, 5)
```

What will be the shape(rows, cols) of result?

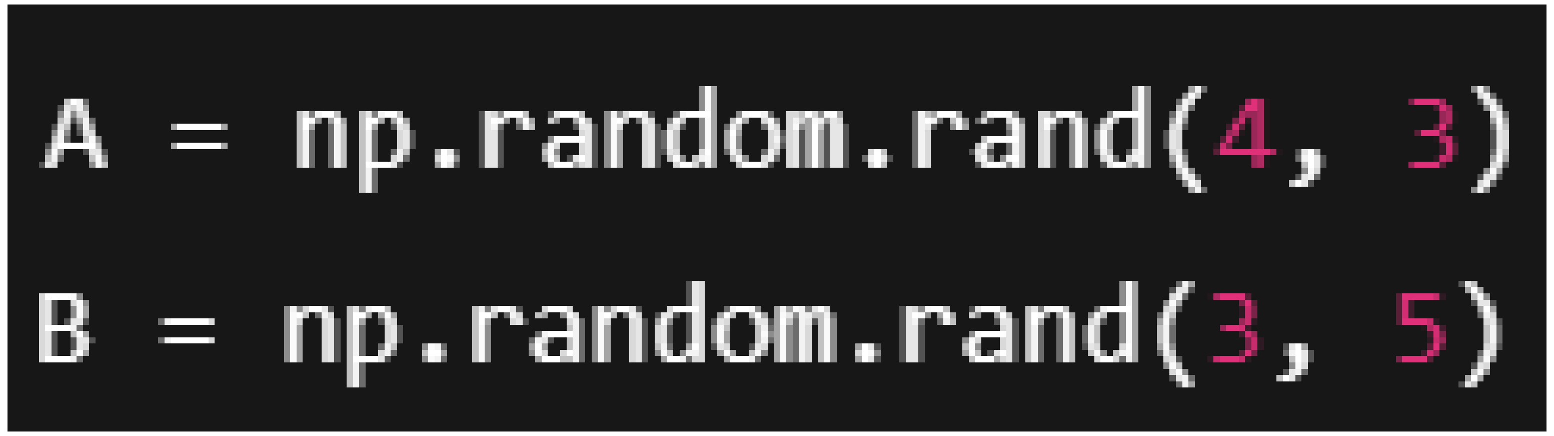
A) (4, 3)

B) (4, 5)

C) (4, 2)

D) (3, 2)

You have the following NumPy arrays:



```
A = np.random.rand(4, 3)
B = np.random.rand(3, 5)
```

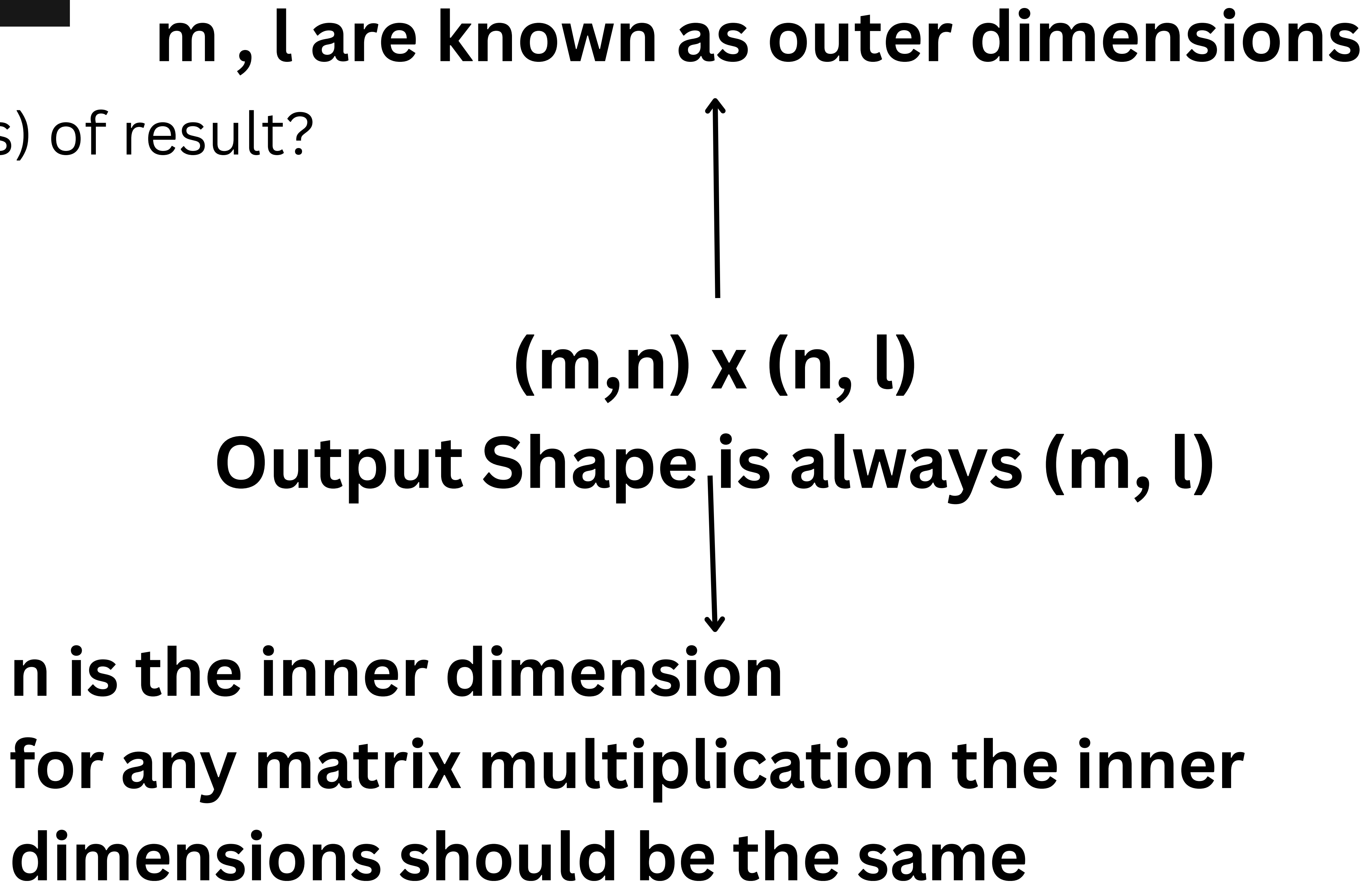
What will be the shape(rows, cols) of result?

A) (4, 3)

B) (4, 5)

C) (4, 2)

D) (3, 2)



m , l are known as outer dimensions

(m,n) x (n, l)

Output Shape is always (m, l)

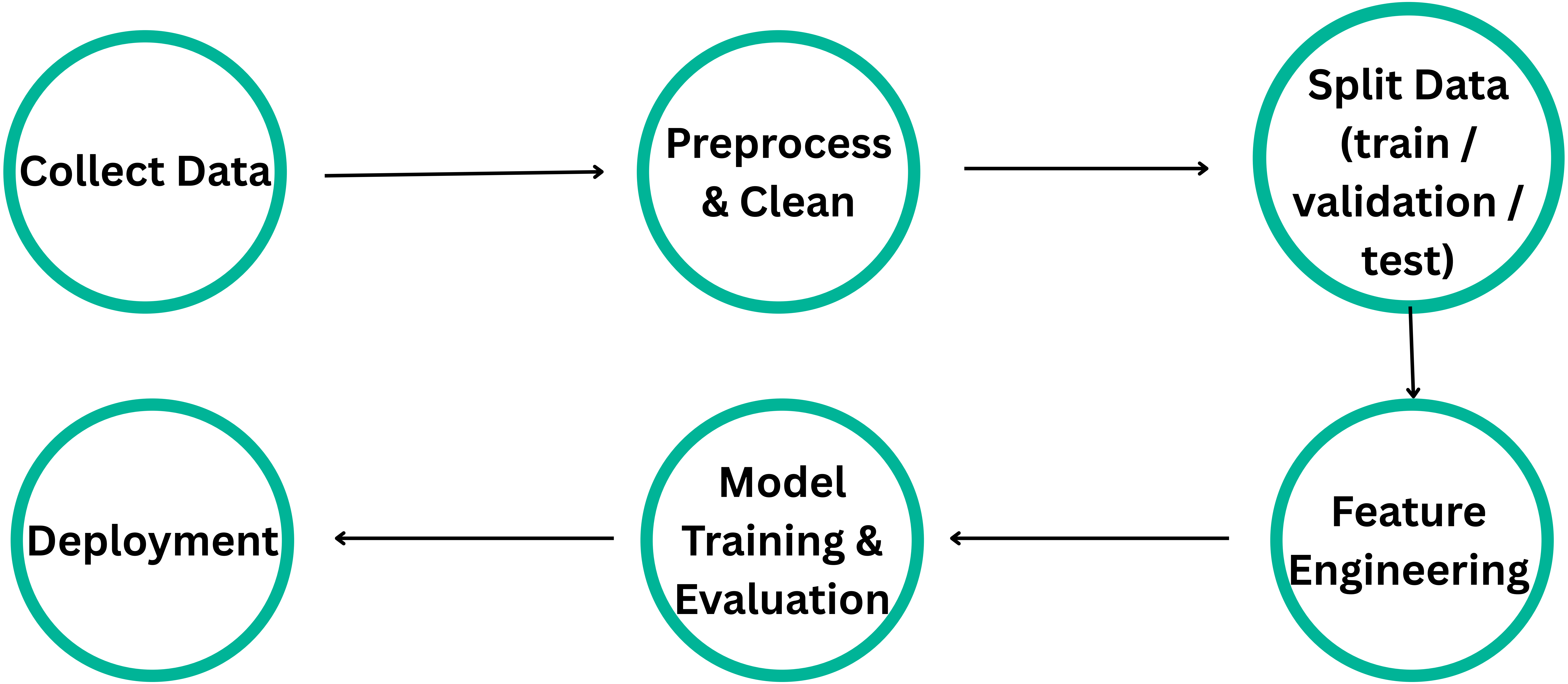
n is the inner dimension

for any matrix multiplication the inner dimensions should be the same

day 2 - data preprocessing and visualisation

How do we train an ML model??
What’s the first step?

Complete Machine Learning Pipeline


Collect Data
Preprocess & Clean
Split Data (train / validation / test)
Feature Engineering
Model Training & Evaluation
Deployment

Data Collection

this is the first step in any machine learning pipeline

Sources of data:

- Public datasets (Kaggle, UCI ML repo)
- APIs & web scraping
- Sensors, logs, user data
- Or by collecting them yourself!

Important Points To Keep In Mind

- **Quality & Accuracy** – Make sure the data is correct, consistent, and free of errors or duplicates.
- **Balanced Representation** – Avoid bias by including all relevant classes and variations (e.g., gender, age, categories).
- **Sufficient Quantity** – Gather enough samples to train and test models reliably (avoid overfitting).

Data Preprocessing

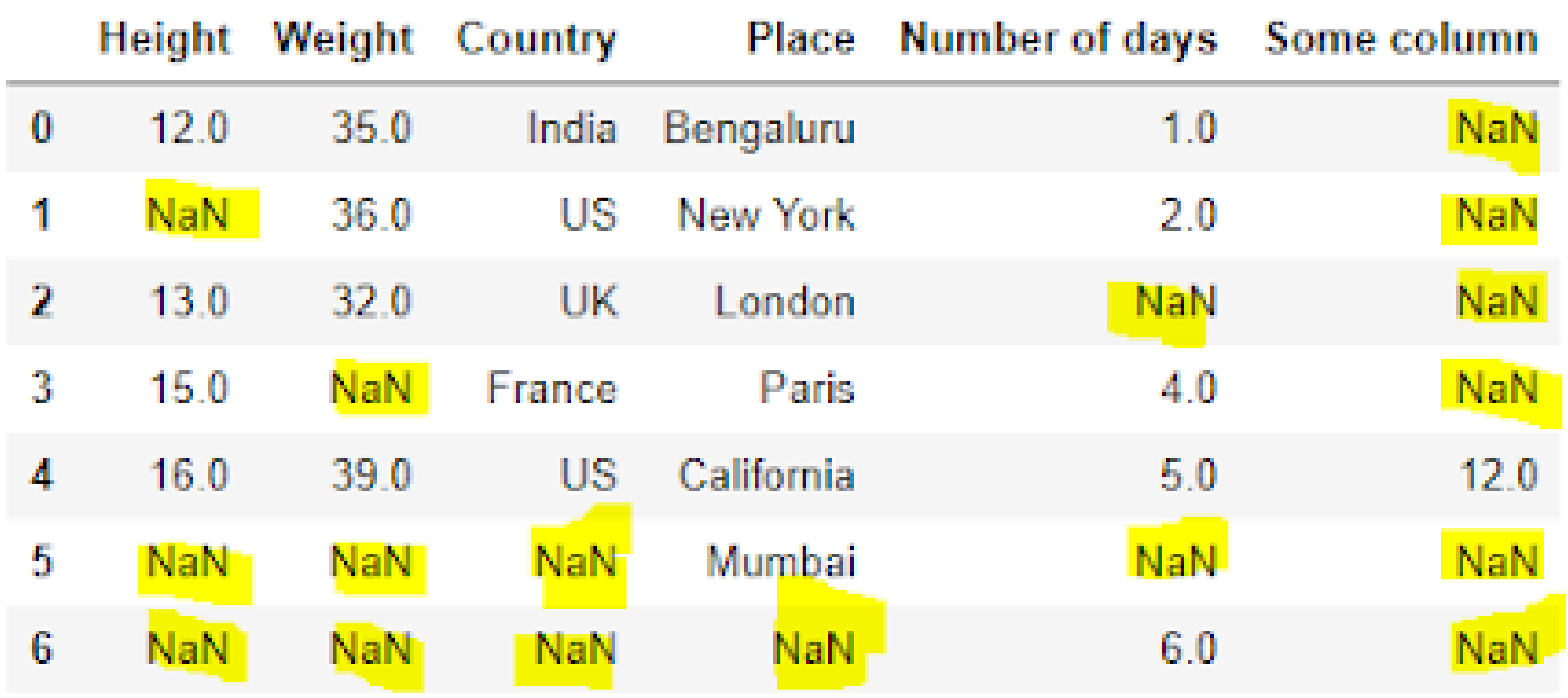
“Garbage in → garbage out” – bad data ruins any model

Typical issues:

- Missing values
- Duplicates
- Inconsistent formats
- Outliers(*if the housing prices are in the range 50 lakh to 1cr, a single value is 20 cr, this is an outlier which is very far from the median value*)

- Good preprocessing → higher accuracy & faster training.

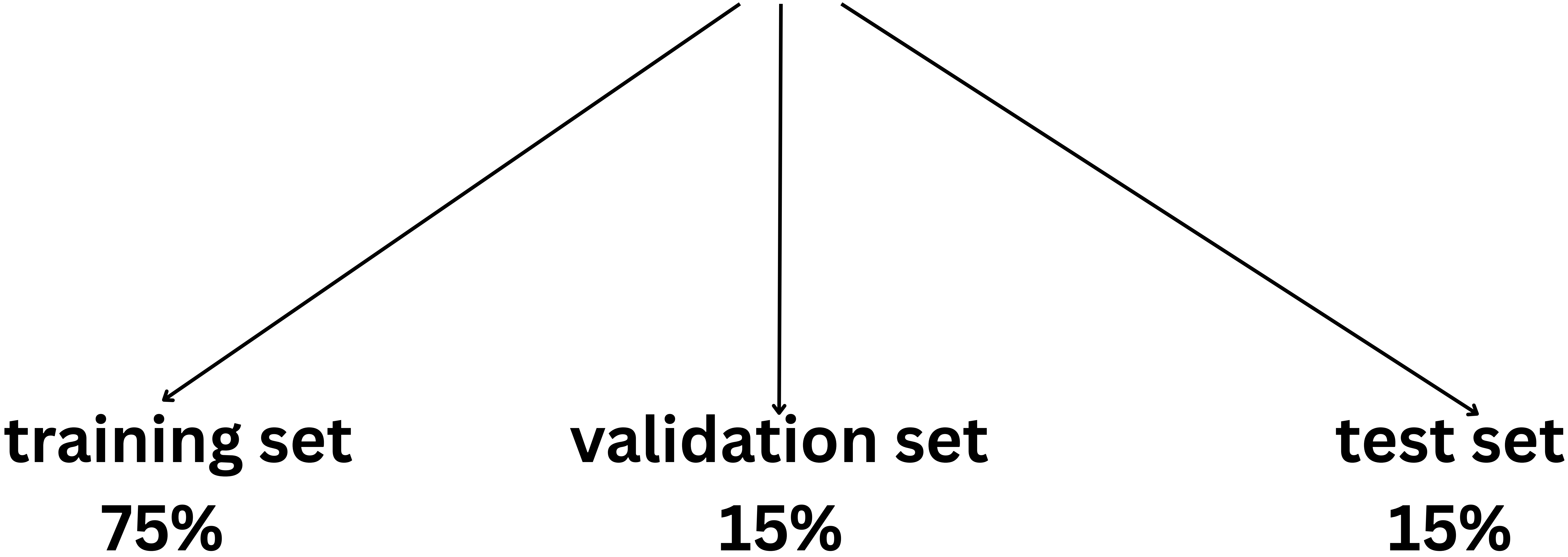
NaN: Null/Empty



	Height	Weight	Country	Place	Number of days	Some column
0	12.0	35.0	India	Bengaluru	1.0	NaN
1	NaN	36.0	US	New York	2.0	NaN
2	13.0	32.0	UK	London	NaN	NaN
3	15.0	NaN	France	Paris	4.0	NaN
4	16.0	39.0	US	California	5.0	12.0
5	NaN	NaN	NaN	Mumbai	NaN	NaN
6	NaN	NaN	NaN	NaN	6.0	NaN

ok now that we have the complete dataset, how do we train the model based on this?

we divide the dataset into



training set
75%

validation set
15%

test set
15%

(or 80:20 Train:Test)

Train set: used to learn model parameters.

Validation set: used to tune hyperparameters and pick the best model.

Test set: used once for final, unbiased evaluation.

Typical ratios → 70/15/15 or 80/20 (train/val/test).

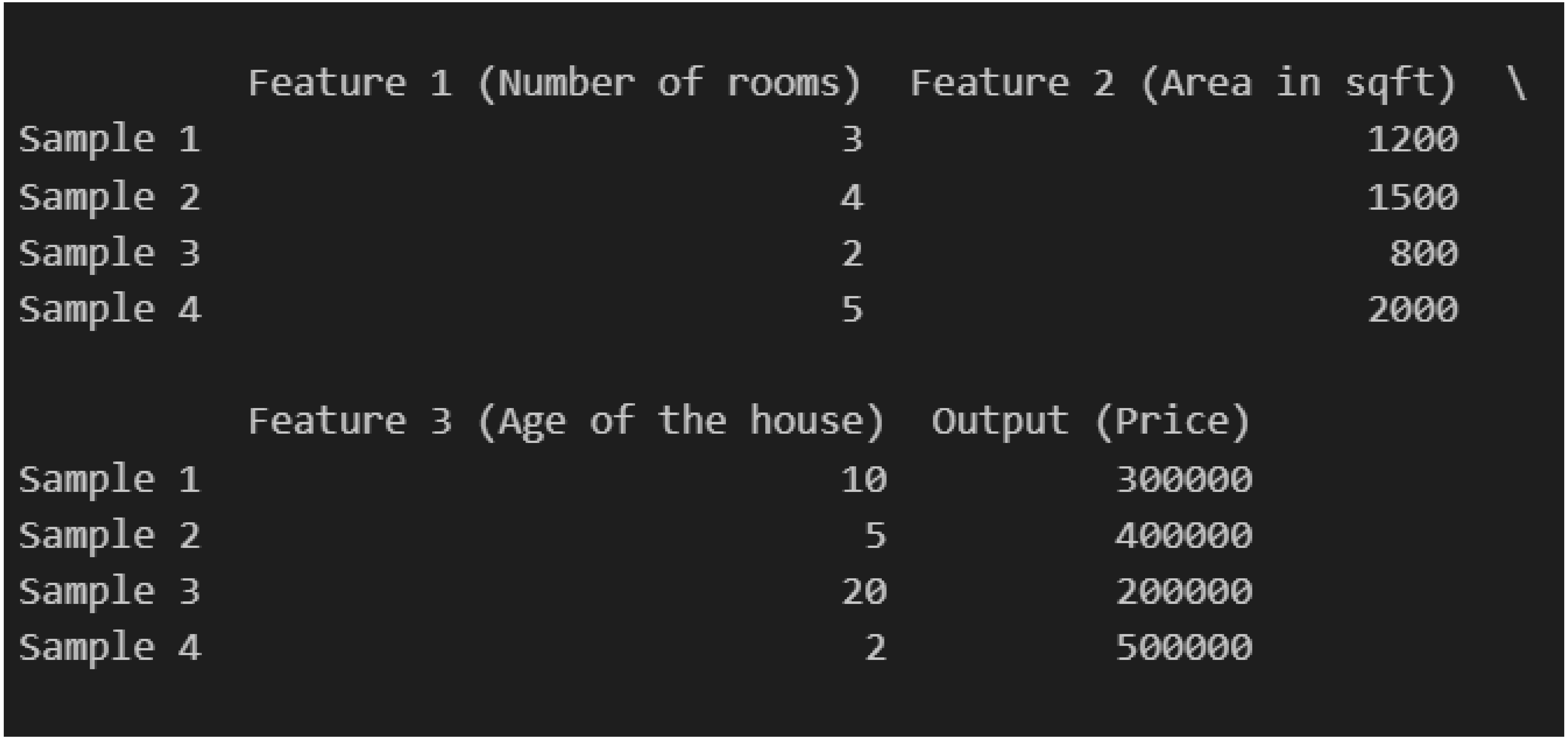
Rule: Never peek at the test set during training.

Feature Selection and Engineering

what is a feature?

A feature = measurable property (one column in the dataset).

the parameters on which the outputs depend upon



```
          Feature 1 (Number of rooms)  Feature 2 (Area in sqft)  \
Sample 1                            3                      1200   
Sample 2                            4                      1500   
Sample 3                            2                       800   
Sample 4                            5                      2000   

          Feature 3 (Age of the house)  Output (Price)  
Sample 1                            10          300000  
Sample 2                             5          400000  
Sample 3                            20          200000  
Sample 4                             2          500000  
```

how do we represent features

$x^{(i)}$ → **feature vector of the i-th sample(house)**

Example: one house → [3 rooms, 1200 sq.ft, 10 years old]

$$x^{(i)} = [3, 1200, 10]$$

$x_j^{(i)} \longrightarrow$ **value of the j-th feature of i-th house**

for example, lets say we want to see the 2nd feature(area) of the 1st house

what will be j and i?

i=1 (because, first house)
j=2 (second feature)

similarly what does this mean?

$$y^{(i)}$$

target/label (e.g., house price)

$$x^{(i)} = [3, 1200, 10] \quad y^{(i)} = 350000$$

feature selection

you have to train a model to predict housing prices, which features will you choose and why?

Look at the features:

- Latitude, Longitude
- Number of rooms, Bedrooms
- Distance to closest Cafe
- Median Income of the Buyer
- House age
- Hot water availability

feature selection

you have to train a model to predict housing prices, which features will you choose and why?

Look at the features:

- Latitude, Longitude
- **Number of rooms, Bedrooms**
- Distance to closest Cafe
- **Median Income of the Buyer**
- **House age**
- Hot water availability

feature engineering

- Features can have very different units/scales → large-scale features dominate learning.
- **Normalization**: scale values to **[0,1]**
- **Standardization:** center values to **mean = 0, std = 1**

simple example

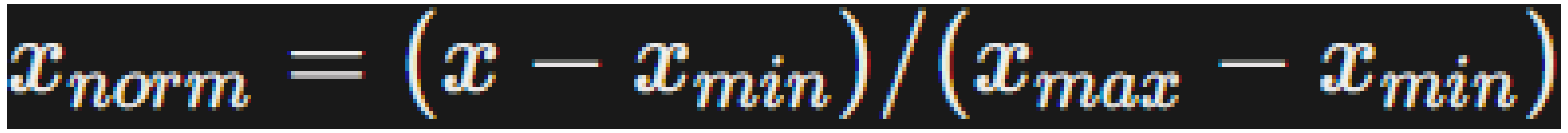
heights = [172, 173, 174, 176, 178]

subtract by 174
heights = [-2, -1, 0, 2, 4]

heights = [172, 173, 174, 176, 178]

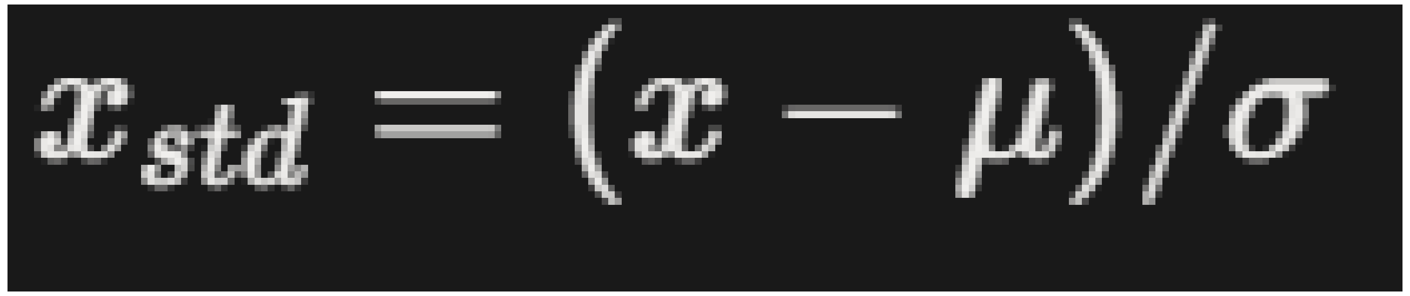
divide by max(178)
heights =[0.966, 0.972, 0.978, 0.989, 1.0]

Normalization (Min-Max Scaling)

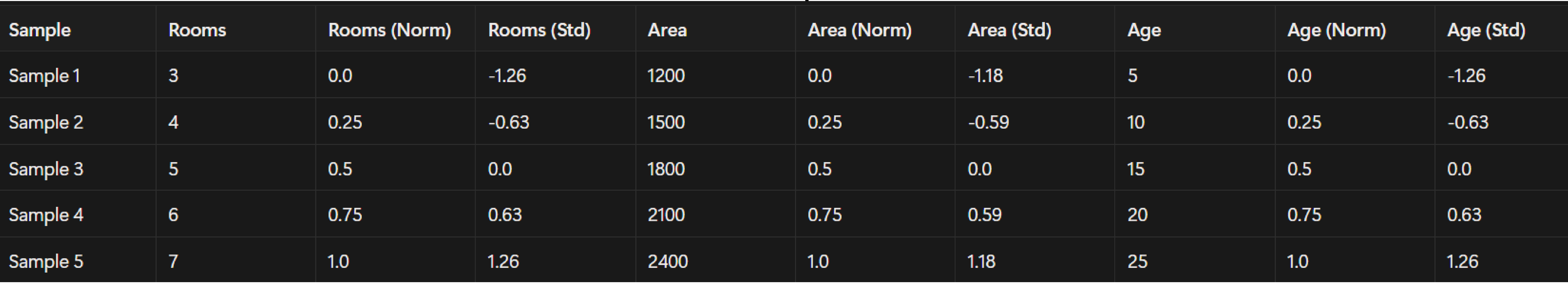


$$x_{norm} = (x - x_{min})/(x_{max} - x_{min})$$

Standardization (Z-score Scaling)



$$x_{std} = (x - \mu)/\sigma$$



Sample	Rooms	Rooms (Norm)	Rooms (Std)	Area	Area (Norm)	Area (Std)	Age	Age (Norm)	Age (Std)
Sample 1	3	0.0	-1.26	1200	0.0	-1.18	5	0.0	-1.26
Sample 2	4	0.25	-0.63	1500	0.25	-0.59	10	0.25	-0.63
Sample 3	5	0.5	0.0	1800	0.5	0.0	15	0.5	0.0
Sample 4	6	0.75	0.63	2100	0.75	0.59	20	0.75	0.63
Sample 5	7	1.0	1.26	2400	1.0	1.18	25	1.0	1.26

what we learnt

- Machine Learning Pipeline
- Different Steps Involved in training a model
- Data Collection and Preprocessing
- Training and Test Split
- What is a feature, representation in terms of x and y
- selecting the best features
- feature engineering

hands on session