# Machine Learning

Aie_22072

```
1. def calculate_euclidean_distance(vector1, vector2):
    if len(vector1) != len(vector2):
        raise ValueError("Vectors must have the same dimensionality")

    distance_squared = sum((v1 - v2) ** 2 for v1, v2 in zip(vector1, vector2))
    return (distance_squared)**0.5


def calculate_manhattan_distance(vector1, vector2):
    if len(vector1) != len(vector2):
        raise ValueError("Vectors must have the same dimensionality")
    return sum(abs(vector1[i] - vector2[i]) for i in range(len(vector1)))

vector_a = [int(x) for x in input("Enter vector A values separated by commas: ").split(',')]
vector_b = [int(x) for x in input("Enter vector B values separated by commas: ").split(',')]

euclidean_distance_result = calculate_euclidean_distance(vector_a, vector_b)
manhattan_distance_result = calculate_manhattan_distance(vector_a, vector_b)

print("Euclidean Distance:", euclidean_distance_result)
print("Manhattan Distance:", manhattan_distance_result)


2. def label_encode_categorical(data):
    unique_values = list(set(data))
    label_mapping = {value: index for index, value in
enumerate(unique_values)}

    numeric_labels = [label_mapping[value] for value in
data]

    return numeric_labels, label_mapping

categorical_data = input("Enter categorical data
(comma-separated values): ").split(',')

numeric_labels, label_mapping =
label_encode_categorical(categorical_data)
print("Numeric Labels:", numeric_labels)
print("Label Mapping:", label_mapping)
```