


```
import numpy as np
import pandas as pd
```



```
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import OrdinalEncoder
```

```
df = pd.read_csv('covid_demyDataet_pakistan.csv')
```

```
df.head()
```



|   | age | gender | fever | cough | city      | has_covid |
|---|-----|--------|-------|-------|-----------|-----------|
| 0 | 60  | Male   | 103.0 | Mild  | Karachi   | No        |
| 1 | 27  | Male   | 100.0 | Mild  | Lahore    | Yes       |
| 2 | 42  | Male   | 101.0 | Mild  | Lahore    | No        |
| 3 | 31  | Female | 98.0  | Mild  | Karachi   | No        |
| 4 | 65  | Female | 101.0 | Mild  | Islamabad | No        |



Next steps:

[Generate code with df](#)[View recommended plots](#)[New interactive sheet](#)

```
df.isnull().sum()
```




|           |    |
|-----------|----|
|           | 0  |
| age       | 0  |
| gender    | 0  |
| fever     | 10 |
| cough     | 0  |
| city      | 0  |
| has_covid | 0  |




dtype: int64

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(df.drop(columns=['has_covid']),df['has_covid'],
                                                test_size=0.2)
```

```
X_train
```



|     | age | gender | fever | cough  | city      |
|-----|-----|--------|-------|--------|-----------|
| 0   | 60  | Male   | 103.0 | Mild   | Karachi   |
| 94  | 79  | Male   | NaN   | Strong | Karachi   |
| 73  | 34  | Male   | 98.0  | Strong | Karachi   |
| 98  | 5   | Female | 98.0  | Strong | Islamabad |
| 72  | 83  | Female | 101.0 | Mild   | Karachi   |
| ... | ... | ...    | ...   | ...    | ...       |
| 39  | 50  | Female | 103.0 | Mild   | Karachi   |
| 25  | 23  | Male   | NaN   | Mild   | Islamabad |
| 38  | 49  | Female | 101.0 | Mild   | Lahore    |
| 78  | 11  | Male   | 100.0 | Mild   | Peshawar  |
| 47  | 18  | Female | 104.0 | Mild   | Peshawar  |



80 rows × 5 columns

Next steps:

[Generate code with X\\_train](#)[View recommended plots](#)[New interactive sheet](#)

## 1. normal way without using coloum transformation

```
# adding simple imputer to fever col
si = SimpleImputer()
X_train_fever = si.fit_transform(X_train[['fever']])
```

```
# also the test data
X_test_fever = si.fit_transform(X_test[['fever']])
```

```
X_train_fever.shape
```

```
(80, 1)
```

```
# Ordinalencoding -> cough
oe = OrdinalEncoder(categories=[['Mild','Strong']])
X_train_cough = oe.fit_transform(X_train[['cough']])
```

```
# also the test data
X_test_cough = oe.fit_transform(X_test[['cough']])
```

```
X_train_cough.shape
```

```
(80, 1)
```

```
# OneHotEncoding -> gender,city
ohe = OneHotEncoder(drop='first',sparse=False)
X_train_gender_city = ohe.fit_transform(X_train[['gender','city']])
```

```
# also the test data
X_test_gender_city = ohe.fit_transform(X_test[['gender','city']])
```

```
X_train_gender_city.shape
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/preprocessing/_encoders.py:975: FutureWarning: `sparse` was renamed to `sparse_output`
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/preprocessing/_encoders.py:975: FutureWarning: `sparse` was renamed to `sparse_output`
warnings.warn(
(80, 4)
```

```
# Extracting Age
X_train_age = X_train.drop(columns=['gender','fever','cough','city']).values
```

```
# also the test data
X_test_age = X_test.drop(columns=['gender','fever','cough','city']).values
```

```
X_train_age.shape
```

```
(80, 1)
```

```
X_train_transformed = np.concatenate((X_train_age,X_train_fever,X_train_gender_city,X_train_cough),axis=1)
```

```
# also the test data
X_test_transformed = np.concatenate((X_test_age,X_test_fever,X_test_gender_city,X_test_cough),axis=1)
```

```
X_train_transformed.shape
```

```
(80, 7)
```

## with coloum transformation

```
from sklearn.compose import ColumnTransformer
```

```
transformer = ColumnTransformer(transformers=[
    ('tnf1',SimpleImputer(),['fever']),
    ('tnf2',OrdinalEncoder(categories=[['Mild','Strong']]),['cough']),
    ('tnf3',OneHotEncoder(sparse=False,drop='first'),['gender','city'])
],remainder='passthrough')
```

```
transformer.fit_transform(X_train).shape
```

```
transformer.transform(X_test).shape
```

```
⚡ /usr/local/lib/python3.10/dist-packages/sklearn/preprocessing/_encoders.py:975: FutureWarning: `sparse` was renamed to `sparse_output`  
warnings.warn(  
(80, 7)
```

```
transformer.transform(X_test).shape
```

```
⚡ (20, 7)
```

Start coding or [generate](#) with AI.