

Explore - LeetCode X Code, Compile & Run | X CodeChef User | CodeCh X Python - Google Drive X data\_preprocessing\_tool X Copy of data\_preprocess X

https://colab.research.google.com/drive/1-aRijAoUwpp2QIz-KI3k6hNlodWQyf87#scrollTo=AjSUXFQqo-3 90%

Open in playground Viewing

### Data Preprocessing Tools

#### Importing the libraries

```
[ ] import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

#### Importing the dataset

```
[ ] dataset = pd.read_csv('Data.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values

[ ] print(X)
```

```
[['France' 44.0 72000.0]
['Spain' 27.0 48000.0]
['Germany' 30.0 54000.0]
['Spain' 38.0 61000.0]
['Germany' 40.0 nan]
['France' 35.0 58000.0]
['Spain' nan 52000.0]
['France' 48.0 79000.0]
['Germany' 50.0 83000.0]
['France' 37.0 67000.0]]
```

```
[ ] print(y)
```

```
[ 'No' 'Yes' 'No' 'No' 'Yes' 'Yes' 'No' 'Yes' 'No' 'Yes']
```

Type here to search

14:50  
01-06-2020

Explore - LeetCode X Code, Compile & Run | X CodeChef User | CodeCh X Python - Google Drive X data\_preprocessing\_tool X Copy of data\_preproces X

https://colab.research.google.com/drive/1-aRijAoUwpp2QIz-KI3k6hNlodWQyf87#scrollTo=AjSUXFQqo-3 90%

Open in playground Viewing

### Taking care of missing data

```
[ ] from sklearn.impute import SimpleImputer
imputer = SimpleImputer(missing_values=np.nan, strategy='mean')
imputer.fit(X[:, 1:3])
X[:, 1:3] = imputer.transform(X[:, 1:3])

[ ] print(X)
```

```
[['France' 44.0 72000.0]
['Spain' 27.0 48000.0]
['Germany' 30.0 54000.0]
['Spain' 38.0 61000.0]
['Germany' 40.0 63777.77777777778]
['France' 35.0 58000.0]
['Spain' 38.77777777777778 52000.0]
['France' 48.0 79000.0]
['Germany' 50.0 83000.0]
['France' 37.0 67000.0]]
```

### Encoding categorical data

### Encoding the Independent Variable

```
[ ] from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
ct = ColumnTransformer(transformers=[('encoder', OneHotEncoder(), [0])], remainder='passthrough')
X = np.array(ct.fit_transform(X))
```

Type here to search

14:50 01-06-2020

Explore - LeetCode X Code, Compile & Run | X CodeChef User | CodeCh Python - Google Drive X data\_preprocessing\_tool X Copy of data\_preprocess X

https://colab.research.google.com/drive/1-aRijAoUwpp2QIz-KI3k6hNlodWQyf87#scrollTo=AjSUXFQqo-3 90%

Open in playground Viewing

```
[ ] print(X)
```

```
[[1.0 0.0 0.0 44.0 72000.0]
 [0.0 0.0 1.0 27.0 48000.0]
 [0.0 1.0 0.0 30.0 54000.0]
 [0.0 0.0 1.0 38.0 61000.0]
 [0.0 1.0 0.0 40.0 63777.77777777778]
 [1.0 0.0 0.0 35.0 58000.0]
 [0.0 0.0 1.0 38.77777777777778 52000.0]
 [1.0 0.0 0.0 48.0 79000.0]
 [0.0 1.0 0.0 50.0 83000.0]
 [1.0 0.0 0.0 37.0 67000.0]]
```

Encoding the Dependent Variable

```
[ ] from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
y = le.fit_transform(y)
```

```
[ ] print(y)
```

```
[0 1 0 0 1 1 0 1 0 1]
```

Splitting the dataset into the Training set and Test set

```
[ ] from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 1)
```

```
[ ] print(X_train)
```

Type here to search

14:50  
01-06-2020

Explore - LeetCode X Code, Compile & Run | X CodeChef User | CodeCh X Python - Google Drive X data\_preprocessing\_tool X Copy of data\_preprocess X

https://colab.research.google.com/drive/1-aRijAoUwpp2QIz-KI3k6hNlodWQyf87#scrollTo=AjSUXFQqo-3 90%

Open in playground Viewing

```
[ ] print(X_train)
```

```
[[0.0 0.0 1.0 38.77777777777778 52000.0]
 [0.0 1.0 0.0 40.0 63777.77777777778]
 [1.0 0.0 0.0 44.0 72000.0]
 [0.0 0.0 1.0 38.0 61000.0]
 [0.0 0.0 1.0 27.0 48000.0]
 [1.0 0.0 0.0 48.0 79000.0]
 [0.0 1.0 0.0 50.0 83000.0]
 [1.0 0.0 0.0 35.0 58000.0]]
```

```
[ ] print(X_test)
```

```
[[0.0 1.0 0.0 30.0 54000.0]
 [1.0 0.0 0.0 37.0 67000.0]]
```

```
[ ] print(y_train)
```

```
[0 1 0 0 1 1 0 1]
```

```
[ ] print(y_test)
```

```
[0 1]
```

### Feature Scaling

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train[:, 3:] = sc.fit_transform(X_train[:, 3:])
X_test[:, 3:] = sc.transform(X_test[:, 3:])
```

```
[ ] print(X_train)
```

Type here to search

14:51 01-06-2020

Explore - LeetCode X Code, Compile & Run | X CodeChef User | CodeCh X Python - Google Drive X data\_preprocessing\_tool X Copy of data\_preproces X

https://colab.research.google.com/drive/1-aRijAoUwpp2QIz-KI3k6hNlodWQyf87#scrollTo=AxjSUXFQqo-3 90%

Open in playground Viewing

[0 1]

### Feature Scaling

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train[:, 3:] = sc.fit_transform(X_train[:, 3:])
X_test[:, 3:] = sc.transform(X_test[:, 3:])

[ ] print(X_train)
```

```
[[0.0 0.0 1.0 -0.19159184384578545 -1.0781259408412425]
 [0.0 1.0 0.0 -0.014117293757057777 -0.07013167641635372]
 [1.0 0.0 0.0 0.566708506533324 0.633562432710455]
 [0.0 0.0 1.0 -0.30453019390224867 -0.30786617274297867]
 [0.0 0.0 1.0 -1.9018011447007988 -1.420463615551582]
 [1.0 0.0 0.0 1.1475343068237058 1.232653363453549]
 [0.0 1.0 0.0 1.4379472069688968 1.574991038163885]
 [1.0 0.0 0.0 -0.7401495441200351 -0.5646194287757332]]
```

```
[ ] print(X_test)
```

```
[[0.0 1.0 0.0 -1.4661817944830124 -0.9069571034860727]
 [1.0 0.0 0.0 -0.44973664397484414 0.2056403393225306]]
```

Type here to search

14:51  
01-06-2020

10 11 12 13 14 15 16 17 18 19 20 21  
22 23 24 25 26 27 28  
29 30 31

April  
Saturday

Wk 16 • 112-253

## Machine learning →

Stacking is the preprocessing step in  
tools in the data preprocessing →

- 1) Importing the libraries
- 2) Importing the datasets
- 3) Taking care of missing data
- 4) Encoding categorical data
- 5) Split the dataset into the training & test set.
- 6) Feature scaling