

```
import tensorflow as tf
```

📄 The default version of TensorFlow in Colab will soon switch to TensorFlow 2.x.
We recommend you [upgrade](#) now or ensure your notebook will continue to use TensorFlow 1.x via the %tensc

```
import pandas as pd
```

Double-click (or enter) to edit

```
data = pd.read_csv('Churn.csv')
data.head(10)
```

📄

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balanc
0	1	15634602	Hargrave	619	France	Female	42	2	0.0
1	2	15647311	Hill	608	Spain	Female	41	1	83807.8
2	3	15619304	Onio	502	France	Female	42	8	159660.8
3	4	15701354	Boni	699	France	Female	39	1	0.0
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.8
5	6	15574012	Chu	645	Spain	Male	44	8	113755.7
6	7	15592531	Bartlett	822	France	Male	50	7	0.0
7	8	15656148	Obinna	376	Germany	Female	29	4	115046.7
8	9	15792365	He	501	France	Male	44	4	142051.0
9	10	15592389	H?	684	France	Male	27	2	134603.8

```
data.info()
```

📄

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
RowNumber      10000 non-null int64
CustomerId     10000 non-null int64
Surname        10000 non-null object
CreditScore    10000 non-null int64
Geography      10000 non-null object
Gender         10000 non-null object
Age            10000 non-null int64
Tenure         10000 non-null int64
Balance        10000 non-null float64
NumOfProducts 10000 non-null int64
HasCrCard      10000 non-null int64
IsActiveMember 10000 non-null int64
EstimatedSalary 10000 non-null float64
Exited         10000 non-null int64
dtypes: float64(2), int64(9), object(3)
memory usage: 1.1+ MB
```

```
data.isnull().any()
```

```
↳ RowNumber      False
   CustomerId     False
   Surname        False
   CreditScore    False
   Geography      False
   Gender         False
   Age            False
   Tenure         False
   Balance        False
   NumOfProducts  False
   HasCrCard      False
   IsActiveMember False
   EstimatedSalary False
   Exited         False
   dtype: bool
```

```
data_new = data.drop(['RowNumber','CustomerId','Surname'],axis=1)
```

```
data_new.head()
```

```
# Dropped the columns which are unique for all users
```

```
↳
```

	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	Exited
0	619	France	Female	42	2	0.00	1	1	0	0
1	608	Spain	Female	41	1	83807.86	1	0	0	0
2	502	France	Female	42	8	159660.80	3	1	0	0
3	699	France	Female	39	1	0.00	2	0	0	0
4	850	Spain	Female	43	2	125510.82	1	1	0	0

```
import numpy as np
```

```
import numpy as np
```

```
data_new.Geography = data_new.Geography.astype('category')
data_new.Gender = data_new.Gender.astype('category')
```

```
data_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 11 columns):
CreditScore      10000 non-null int64
Geography        10000 non-null category
Gender           10000 non-null category
Age              10000 non-null int64
Tenure           10000 non-null int64
Balance          10000 non-null float64
NumOfProducts   10000 non-null int64
HasCrCard        10000 non-null int64
IsActiveMember   10000 non-null int64
EstimatedSalary  10000 non-null float64
Exited           10000 non-null int64
dtypes: category(2), float64(2), int64(7)
memory usage: 723.0 KB
```

```
from sklearn.preprocessing import LabelEncoder
labelencoder_x = LabelEncoder()
data_new.iloc[:, 1] = labelencoder_x.fit_transform(data_new.iloc[:, 1]) #applying on Geography
data_new.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 11 columns):
CreditScore      10000 non-null int64
Geography        10000 non-null category
Gender           10000 non-null category
Age              10000 non-null int64
Tenure           10000 non-null int64
Balance          10000 non-null float64
NumOfProducts   10000 non-null int64
HasCrCard        10000 non-null int64
IsActiveMember   10000 non-null int64
EstimatedSalary  10000 non-null float64
Exited           10000 non-null int64
dtypes: category(2), float64(2), int64(7)
memory usage: 723.0 KB
```

	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	619	0	Female	42	2	0.00	1	1	1	151960.00	0
1	608	2	Female	41	1	83807.86	1	0	1	151600.00	0
2	502	0	Female	42	8	159660.80	3	1	1	151600.00	0
3	699	0	Female	39	1	0.00	2	0	1	151600.00	0
4	850	2	Female	43	2	125510.82	1	1	1	151600.00	0

```
#apply encoder on Gender as well
labelencoder_x_2 = LabelEncoder()
data_new.iloc[:, 2] = labelencoder_x_2.fit_transform(data_new.iloc[:, 2]) #applying on Gender
data_new.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 11 columns):
CreditScore      10000 non-null int64
Geography        10000 non-null category
Gender           10000 non-null category
Age              10000 non-null int64
Tenure           10000 non-null int64
Balance          10000 non-null float64
NumOfProducts   10000 non-null int64
HasCrCard        10000 non-null int64
IsActiveMember   10000 non-null int64
EstimatedSalary  10000 non-null float64
Exited           10000 non-null int64
dtypes: category(2), float64(2), int64(7)
memory usage: 723.0 KB
```

	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActive
0	619	0	0	42	2	0.00	1	1	
1	608	2	0	41	1	83807.86	1	0	
2	502	0	0	42	8	159660.80	3	1	
3	699	0	0	39	1	0.00	2	0	
4	850	2	0	43	2	125510.82	1	1	

#One hot encoding.

```
from keras.utils import to_categorical
encoded = pd.DataFrame(to_categorical(data_new.iloc[:, 1]))
#no need to encode Gender, as there are only two categories
```

```
x = pd.concat([encoded, data_new], axis = 1)
x.head()
```

	0	1	2	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActive
0	1.0	0.0	0.0	619	0	0	42	2	0.00	1	1	
1	0.0	0.0	1.0	608	2	0	41	1	83807.86	1	0	
2	1.0	0.0	0.0	502	0	0	42	8	159660.80	3	1	
3	1.0	0.0	0.0	699	0	0	39	1	0.00	2	0	
4	0.0	0.0	1.0	850	2	0	43	2	125510.82	1	1	

#Dropping the existing "geography" category, and one of the onehotcoded columns.

```
x = x.drop(['Geography', 0], axis = 1)
x.head()
```

	1	2	CreditScore	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActive
0	0.0	0.0	619	0	42	2	0.00	1	1	
1	0.0	1.0	608	0	41	1	83807.86	1	0	
2	0.0	0.0	502	0	42	8	159660.80	3	1	
3	0.0	0.0	699	0	39	1	0.00	2	0	
4	0.0	1.0	850	0	43	2	125510.82	1	1	

#train and test set split, and feature scaling

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

```
X_train, X_test, Y_train, Y_test = train_test_split(x, y, test_size = 0.2, random_state = 0)
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

```
import keras
from keras.models import Sequential
from keras.layers import Dense #to add layers
```

```
classifier = Sequential()
classifier.add(Dense(output_dim = 6, init = 'uniform', activation = 'relu', input_dim =12))
classifier.add(Dense(output_dim = 6, init = 'uniform', activation = 'relu'))
```

```
classifier.add(Dense(output_dim = 1, init = 'uniform', activation = 'sigmoid'))
```

```
classifier.compile(optimizer = 'adam', loss = "binary_crossentropy", metrics = ['accuracy'])
```

```
↳ /usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:7: UserWarning: Update your
    import sys
    /usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:8: UserWarning: Update your
    /usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:11: UserWarning: Update you
    # This is added back by InteractiveShellApp.init_path()
```

```
classifier.fit(X_train, Y_train, batch_size = 10, nb_epoch = 100)
```

```
↳
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: UserWarning: The `nb_ipo
"""Entry point for launching an IPython kernel.
```

```
Epoch 1/100
8000/8000 [=====] - 1s 184us/step - loss: 0.2321 - acc: 0.9386
Epoch 2/100
8000/8000 [=====] - 1s 143us/step - loss: 0.0055 - acc: 1.0000
Epoch 3/100
8000/8000 [=====] - 1s 136us/step - loss: 0.0011 - acc: 1.0000
Epoch 4/100
8000/8000 [=====] - 1s 133us/step - loss: 4.2760e-04 - acc: 1.0
Epoch 5/100
8000/8000 [=====] - 1s 137us/step - loss: 2.0391e-04 - acc: 1.0
Epoch 6/100
8000/8000 [=====] - 1s 151us/step - loss: 1.0813e-04 - acc: 1.0
Epoch 7/100
8000/8000 [=====] - 1s 156us/step - loss: 6.1000e-05 - acc: 1.0
Epoch 8/100
8000/8000 [=====] - 1s 158us/step - loss: 3.5682e-05 - acc: 1.0
Epoch 9/100
8000/8000 [=====] - 1s 158us/step - loss: 2.1396e-05 - acc: 1.0
Epoch 10/100
8000/8000 [=====] - 1s 168us/step - loss: 1.3056e-05 - acc: 1.0
Epoch 11/100
8000/8000 [=====] - 1s 156us/step - loss: 8.0650e-06 - acc: 1.0
Epoch 12/100
8000/8000 [=====] - 1s 162us/step - loss: 5.0246e-06 - acc: 1.0
Epoch 13/100
8000/8000 [=====] - 1s 162us/step - loss: 3.1524e-06 - acc: 1.0
Epoch 14/100
8000/8000 [=====] - 1s 141us/step - loss: 1.9869e-06 - acc: 1.0
Epoch 15/100
8000/8000 [=====] - 1s 137us/step - loss: 1.2549e-06 - acc: 1.0
Epoch 16/100
8000/8000 [=====] - 1s 139us/step - loss: 7.9611e-07 - acc: 1.0
Epoch 17/100
8000/8000 [=====] - 1s 144us/step - loss: 5.0662e-07 - acc: 1.0
Epoch 18/100
8000/8000 [=====] - 1s 143us/step - loss: 3.3043e-07 - acc: 1.0
Epoch 19/100
8000/8000 [=====] - 1s 140us/step - loss: 2.2825e-07 - acc: 1.0
Epoch 20/100
8000/8000 [=====] - 1s 138us/step - loss: 1.7198e-07 - acc: 1.0
Epoch 21/100
8000/8000 [=====] - 1s 143us/step - loss: 1.3986e-07 - acc: 1.0
Epoch 22/100
8000/8000 [=====] - 1s 140us/step - loss: 1.2148e-07 - acc: 1.0
Epoch 23/100
8000/8000 [=====] - 1s 139us/step - loss: 1.1154e-07 - acc: 1.0
Epoch 24/100
8000/8000 [=====] - 1s 145us/step - loss: 1.0709e-07 - acc: 1.0
Epoch 25/100
8000/8000 [=====] - 1s 148us/step - loss: 1.0517e-07 - acc: 1.0
Epoch 26/100
8000/8000 [=====] - 1s 145us/step - loss: 1.0447e-07 - acc: 1.0
Epoch 27/100
8000/8000 [=====] - 1s 144us/step - loss: 1.0415e-07 - acc: 1.0
Epoch 28/100
```

```
8000/8000 [=====] - 1s 148us/step - loss: 1.0403e-07 - acc: 1.0
Epoch 29/100
8000/8000 [=====] - 1s 144us/step - loss: 1.0397e-07 - acc: 1.0
Epoch 30/100
8000/8000 [=====] - 1s 148us/step - loss: 1.0395e-07 - acc: 1.0
Epoch 31/100
8000/8000 [=====] - 1s 148us/step - loss: 1.0395e-07 - acc: 1.0
Epoch 32/100
8000/8000 [=====] - 1s 146us/step - loss: 1.0394e-07 - acc: 1.0
Epoch 33/100
8000/8000 [=====] - 1s 150us/step - loss: 1.0393e-07 - acc: 1.0
Epoch 34/100
8000/8000 [=====] - 1s 185us/step - loss: 1.0393e-07 - acc: 1.0
Epoch 35/100
8000/8000 [=====] - 1s 185us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 36/100
8000/8000 [=====] - 1s 173us/step - loss: 1.0393e-07 - acc: 1.0
Epoch 37/100
8000/8000 [=====] - 1s 169us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 38/100
8000/8000 [=====] - 1s 142us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 39/100
8000/8000 [=====] - 1s 140us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 40/100
8000/8000 [=====] - 1s 142us/step - loss: 1.0393e-07 - acc: 1.0
Epoch 41/100
8000/8000 [=====] - 1s 148us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 42/100
8000/8000 [=====] - 1s 144us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 43/100
8000/8000 [=====] - 1s 145us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 44/100
8000/8000 [=====] - 1s 141us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 45/100
8000/8000 [=====] - 1s 150us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 46/100
8000/8000 [=====] - 1s 145us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 47/100
8000/8000 [=====] - 1s 139us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 48/100
8000/8000 [=====] - 1s 142us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 49/100
8000/8000 [=====] - 1s 139us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 50/100
8000/8000 [=====] - 1s 142us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 51/100
8000/8000 [=====] - 1s 139us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 52/100
8000/8000 [=====] - 1s 141us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 53/100
8000/8000 [=====] - 1s 139us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 54/100
8000/8000 [=====] - 1s 142us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 55/100
8000/8000 [=====] - 1s 138us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 56/100
8000/8000 [=====] - 1s 135us/step - loss: 1.0392e-07 - acc: 1.0
Epoch 57/100
```