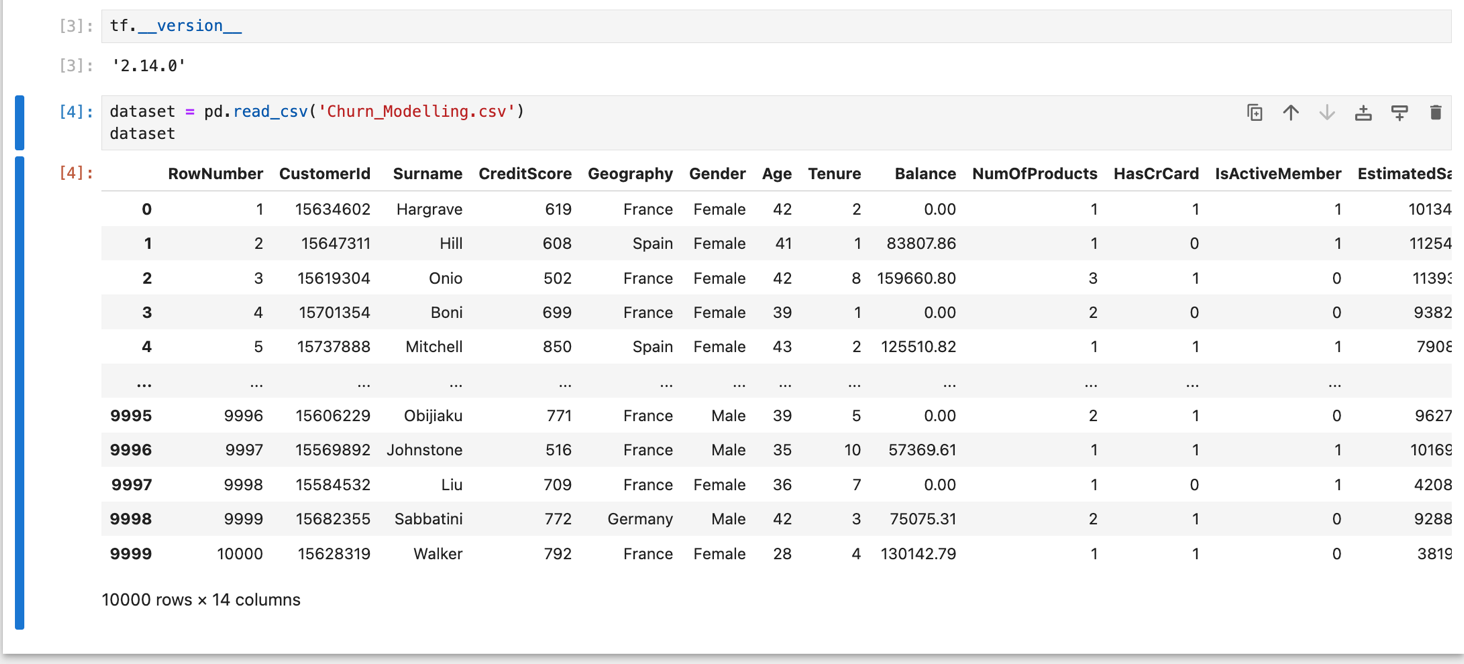
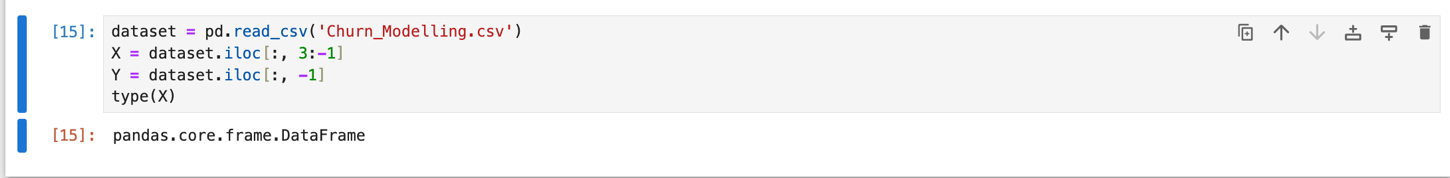
**The Dataset:**

****

**Please pay attention:**

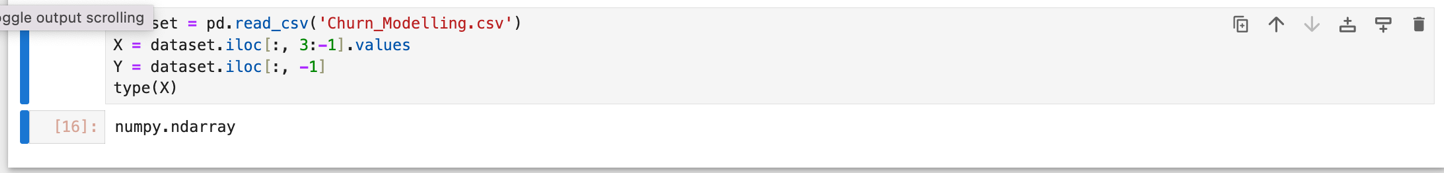
X = dataset.iloc[:, 3:-1]

X is a dataframe (& with column index)

****

X = dataset.iloc[:, 3:-1].values

X is a NumPy array (without column index)

****

**Encoding the categorical data:**

**Male/Female => 1/0**

# Encoding categorical data

# Label Encoding the "Gender" column

from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

X[:, 2] = le.fit\_transform(X[:, 2])

print(X)

**Geography: France, Germany, Spain**

**(1, 0, 0)**

**(0, 1, 0)**

**(0, 0, 1)**

# One Hot Encoding the "Geography" column

from sklearn.compose import ColumnTransformer

from sklearn.preprocessing import OneHotEncoder

ct = ColumnTransformer(transformers=[('encoder', OneHotEncoder(), [1])], remainder='passthrough')

X = np.array(ct.fit\_transform(X))

print(X)

The code is used for performing one-hot encoding on a specific column ("Geography") of the dataset. One-hot encoding is a technique commonly used in machine learning to convert categorical data into a numerical format that can be used by machine learning algorithms. Here's an explanation of each part of the code:

**1- Importing the Necessary Libraries:**

from sklearn.compose import ColumnTransformer

from sklearn.preprocessing import OneHotEncoder

This code imports the required classes from the scikit-learn library.

* ColumnTransformer is used to apply transformations to specific columns, and
* OneHotEncoder is used for one-hot encoding.

**2 - Creating a ColumnTransformer Object:**

ct = ColumnTransformer(transformers=[('encoder', OneHotEncoder(), [1])], remainder='passthrough')

**ColumnTransformer** is instantiated with several parameters:

* transformers: This is a list of transformation tuples. In this case, there's one tuple containing three elements:

'encoder': This is a string identifier for the transformer.

OneHotEncoder(): This is an instance of the OneHotEncoder class. It will be used to one-hot encode the specified column.

[1]: This is a list that specifies the index of the column to be one-hot encoded. In this code, it's column 1.

remainder='passthrough': This parameter specifies what to do with the remaining columns in the dataset that are not explicitly transformed. 'passthrough' means that the remaining columns will not be modified and will be included as is in the final output.

**3 - Transforming the Data:**

X = np.array(ct.fit\_transform(X))

ct.fit\_transform(X) applies the transformation defined by the ColumnTransformer to the input data X.

* The result is an array where the "Geography" column (column 1) has been one-hot encoded, and the remaining columns (if any) have been included as is.
* The transformed data is stored in the variable X.

After running this code, we will have a modified X that contains one-hot encoded values for the "Geography" column, making it suitable for use in machine learning models that require numerical input. The other columns, if any, remain unchanged and are also included in the X array.

**Splitting the dataset into a Test Set & Training Set:**

# 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 = 0)