Lesson 2 Data Science Process

Data Science involves the following major steps with data:

- gathering data
- data preparation
- data wrangling
- analyse Data
- train the model
- test the model
- deployment

We need to identify the data sources where data can be collected. It would be data from files, database, internet, or mobile devices. The quantity and quality of the collected data will determine the efficiency of the output. The more will be the data, the more accurate will be the prediction.

After collecting the data, we need to prepare it for further steps. Data preparation is a step where we put our data into a suitable place and prepare it to use in our training. Data exploration is used to understand the nature of data that we have to work with. We need to understand the characteristics, format, and quality of data. We find correlations, general trends, and outliers. Now the next step is preprocessing of data for its analysis.

Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.

It is not necessary that data we have collected is always of our use as some of the data may not be useful. In real-world applications, collected data may have various issues, including:

- missing values
- duplicate data
- invalid data
- noise

So, we use various filtering techniques to clean the data.

It is mandatory to detect and remove the above issues because it can negatively affect the quality of the outcome.

Now the cleaned and prepared data is passed on to the analysis step. This step involves:

- selection of analytical techniques
- building models
- review the result

The aim of this step is to build a model to analyze the data using various analytical techniques and review the outcome. It starts with the determination of the type of the problems, where we select the for example some classical techniques such as Classification, Regression, Cluster analysis, Association, etc. then build the model using prepared data, and evaluate the model. Hence, in this step, we take the data and use some algorithms to build the model.

Now the next step is to train the model, in this step we train our model to improve its performance for better outcome of the problem.

We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various patterns, rules, and, features.

Once our model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it.

Testing the model determines the percentage accuracy of the model as per the requirement of project or problem. The last step of Data Science life cycle is deployment, where we deploy the model in the real-world system.

If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project, we will check whether it is improving its performance using available data or not. The deployment phase is similar to making the final report for a project.

Getting datasets

A dataset is a collection of data in which data is arranged in some order. A dataset can contain any data from a series of an array to a database table. The most supported file type for a tabular dataset is "Comma Separated File," or CSV. But to store a "tree-like data," we can use the JSON file more efficiently.

Types of data in datasets

- Numerical data (house price, temperature, etc.)
- Categorical data (Yes/No, True/False, Blue/green, etc.)
- Ordinal data (These data are similar to categorical data but can be measured on the basis of comparison)

Types of datasets

Data analysis incorporates different domains, each requiring explicit sorts of datasets. A few sorts of datasets include:

- image datasets (ImageNet, CIFAR-10,MNIST)
- text datasets (Gutenberg task dataset, IMDb film reviews dataset)
- time series datasets (climate information, sensor readings)
- tabular datasets (Tabular datasets are organized information coordinated in tables or calculation sheets)

Popular sources for datasets

- Kaggle
- Google's Dataset Search
- Microsoft Datasets
- Government Datasets
- Scikit-learn dataset

Data Preprocessing

Data preprocessing is a process of preparing the raw data and making it suitable for a Data analysis.

When we are working with some project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So for this, we use data preprocessing task.

Data Preprocessing involves the following steps:

- getting the dataset
- importing libraries
- finding missing data
- encoding data
- splitting dataset into training and test set
- feature scaling

Get the Dataset

To create a model, the first thing we required is a dataset. The collected data for a particular problem in a proper format is known as the **dataset**.

Dataset may be of different formats for different purposes, such as, if we want to create a machine learning model for business purpose, then dataset will be different with the dataset required for a liver patient. So each dataset is different from another dataset. To use the dataset in our code, we usually put it into a CSV file. However, sometimes, we may also need to use an HTML or xlsx file.

CSV file

CSV stands for "Comma-Separated Values". This is a file format which allows us to save the tabular data, such as spreadsheets.

Here we will use a demo dataset for data preprocessing, and for practice, it can be downloaded from the internet. For real-world problems, we can download datasets online from various sources such as https://www.kaggle.com/uciml/datasets (https://www.kaggle.com/uciml/datasets), https://archive.ics.uci.edu/ml/index.php (https://archive.ics.uci.edu/ml/index.php) etc.

We can also create our dataset by gathering data using various API with Python and put that data into a .csv file.

Importing Libraries

In order to perform data preprocessing using Python, we need to import some predefined Python libraries. These libraries are used to perform some specific jobs. There are three specific libraries that we will use for data preprocessing, which are:

- Numpy. Numpy is used for including any type of mathematical operation in the code. It is the fundamental package for scientific calculation in Python. It also supports to add large, multidimensional arrays and matrices.
- Matplotlib. The second library is matplotlib, which is a Python 2D plotting library, and with this library, we need to import a sub-library pyplot. This library is used to plot any type of charts in Python.
- Pandas. The last library is the Pandas library, which is one of the most famous Python libraries and used for importing and managing the datasets. It is an open-source data manipulation and analysis library.

All these libraries will be imported as below:

In [ ]:
```
```

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

Importing the Datasets

Now we need to import the datasets which we have collected for our project. But before importing a dataset, we need to save .csv file in working directory.

read_csv() function

Now to import the dataset, we will use read_csv() function of pandas library, which is used to read a csv file and performs various operations on it. Using this function, we can read a csv file locally as well as through an URL.

In [2]:
```
df= pd.read_csv('train.csv')
df
```

Out[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

Extracting variables

To extract a variable, we will use iloc method of Pandas. It is used to extract the required rows and columns from the dataset.

In [3]:
```
x= df.iloc[:,:-1].values
x
```

Out[3]:
```
array([[1, 0, 3, ..., 'A/5 21171', 7.25, nan],
       [2, 1, 1, ..., 'PC 17599', 71.2833, 'C85'],
       [3, 1, 3, ..., 'STON/O2. 3101282', 7.925, nan],
       ...,
       [889, 0, 3, ..., 'W./C. 6607', 23.45, nan],
       [890, 1, 1, ..., '111369', 30.0, 'C148'],
       [891, 0, 3, ..., '370376', 7.75, nan]], dtype=object)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

In this code, the first colon(:) is used to take all the rows, and the second colon(:) is for all the columns. Here we've used :-1, because we don't want to take the last column.

Finding missing data

The next step of data preprocessing is to handle missing data in the datasets. If our dataset contains some missing data, then it may create a huge problem for our model. Hence it is necessary to handle missing values present in the dataset. There are two ways to handle missing data, which are:

- deleting the particular row
- calculating the mean

In [5]:
```
# Replacing missing data with the some value
df.fillna({'Embarked':"Not specified", "Age":35})
#  If all values are NA, drop that row or column
df.dropna(how='all')
# take more info from https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.dropna.html
```

Out[5]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns