Introduction to Data Science for Non-Science background folks

Introduction

When we hear 'Data Science ' it sounds like some rocket science or some core science thing, but let me tell you its completely easy and requires common sense to understand. According to me data science is getting an outcome by using statistics and technical tools, and then using those outcomes to make business decisions.

Let me explain you with an example,

Suppose you have to open a garment shop. So before purchasing or renting a place you will first consider few factors such as whether the property is in prime location, what is the income of people in the locality, is there any competitors shop nearby. So this factor is your data and after analyzing those you will consider whether to open shop or vice versa.

Another example is suppose you want to pursue MBA. So you will make a list of colleges with their fees, faculty's information, infrastructure and placements details. Here the fees, placement, infrastructure information will be your data and after analysis of this data you will select a college.

In real world data science is used in various fields so that the organization can take a proper decision on the basis of real world data. Statistics concepts are used to analyze the data and programming language such as python, sql etc are used to extract the data from the database and applying statistics on those.

Through this blog you will get an understanding of what happens in data science analysis.

Below are the common techniques used in data science analysis,

1. Exploratory Data Analysis

Whenever a dataset is received the first thing is performing exploratory data analysis (EDA). Through EDA we can summarize the main characteristics of the dataset by using visual methods. EDA helps in making the dataset clean and ready for testing various models. Following steps are performed in EDA,

Handling missing value

The first step here is to find out whether the data has any missing value and to treat them. A dataset with blank rows can create hurdles in implementing models. Here we use programming to find whether there is a missing value. If the dataset has a missing value we handle it by using statistics. The most used technique is mean, median and mode.

Handling Outlier

Outliers are data point which differs significantly from the rest of the data. Let's see the example below,

Below table have the sample information of students name and their age of a particular school

Name	Age
Abc	9
Xyz	11
Def	50
Mno	13
Pqr	8

When we see the age column we see that student Def has age 50, which clearly indicates there was mistake while entering the age, because the usual age of school student is between 5 to 18.

The method to find outliers is:-

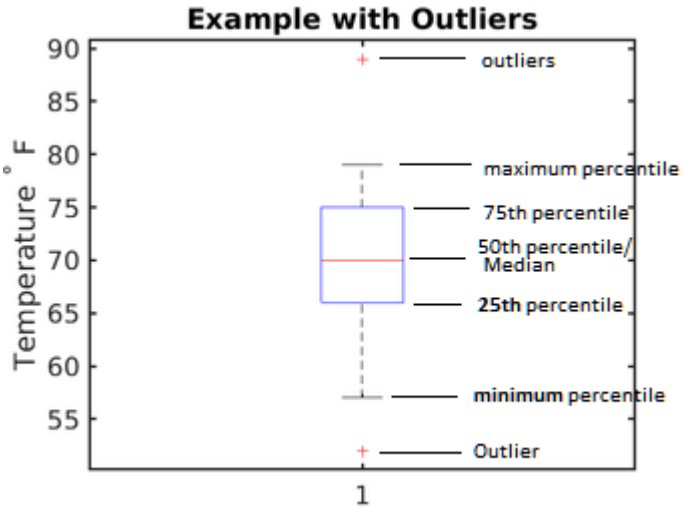
i. Percentile

In this method we select a minimum percentile and maximum percentile, then we delete the data outside those percentile.

ii. Box Plot

Box plot is a graphical representation for describing the distribution of the data.

Let's understand it through below image downloaded from Wikipedia,



In the above image we can identify the extreme data points as outliers. The maximum data will be in between 25 to 75 percentile.

How to handle the outliers?

i. Deleting the outliers point.

ii. Replacing the outlier's values with suitable values. IQR method is used here, in this method we subtract the 75^{th} percentile (Q3) and 25^{th} percentile(Q1) i.e. IQR = Q3 – Q1.

Bivariate Analysis

Analyzing 2 variables is known as Bivariate analysis. While performing EDA we compare different variable to analyze the data. For e.g. the marketing department will check the sales graph by comparing it with the advertizing spends. If sales increases through advertisement then they will spent more on advertisement, if there is no change in sales then they can make new strategies.

The visual methods used are scatter plot, bar chart, heat map, box plot etc.

Now we understood through EDA we get an idea how the dataset is and what trend it follows.

2. Natural Language Processing

Now a day's people review products on e-commerce sites and also reviews restaurants and hotels on Google. So if the manufacturer of the product or the hotel owners wants to analyze on what basis they are receiving positive and negative review, natural language processing (NLP) is used here.

NLP is a form of artificial intelligence that gives computers the ability to read, understand and interpret human language. By using NLP we can measure sentiments and determine which part of human language are important.

In NLP we divide the sentence into words, remove filler words and analyze the unique words.

Let's have a look at basic NLP operations. We will refer the example in the below image,

Creating bunch of sentences. I have intentionally corrupt the sentence.

```
In [6]: #Creating bunch of sentences. i have intentionally corrupt the sentence.
        raw_docs = ["I am writing some very basic english sentences",
        "I'm just writing it for the demo PURPOSE to make audience understand the basi
        cs .",
        "The point is to _learn HOW it works_ on #simple # data."]
```

The sentence used in the example is "I am writing some very basic english sentences, I'm just writing it for the demo PURPOSE to make audience understand the basics .,The point is to _learn HOW it works_ on #simple # data."

i. <u>The first step is to lowering the words to small case.</u>

Convert everything into lower case

```
In [8]: #import string
        raw_docs = [doc.lower() for doc in raw_docs]
        print(raw_docs)

        ['i am writing some very basic english sentences', "i'm just writing it for t
        he demo purpose to make audience understand the basics .", 'the point is to _
        learn how it works_ on #simple # data.']
```

In the above image we can see by using python code we have transformed all characters to small cap.

<u>ii. Tokenization</u>

The process of breaking down a paragraph into words and sentence is called tokenization.

Step 2 - Tokenization

The process of breaking down a text paragraphs in smaller chunks such as words or sentence is called Tokenization.

```
In [14]: #word tokenize
         from nltk.tokenize import word_tokenize
         tokenized_words = [word_tokenize(doc) for doc in raw_docs]
         print('tokenized_words:', tokenized_words)
         print()
         #Sentence tokenization
         from nltk.tokenize import sent_tokenize
         sent_tokenize = [sent_tokenize(doc) for doc in raw_docs]
         print('tokenize_sentence:', sent_tokenize)

         tokenized_words: [['i', 'am', 'writing', 'some', 'very', 'basic', 'english',
         'sentences'], ['i', "'m", 'just', 'writing', 'it', 'for', 'the', 'demo', 'pur
         pose', 'to', 'make', 'audience', 'understand', 'the', 'basics', '.'], ['the',
         'point', 'is', 'to', '_learn', 'how', 'it', 'works_', 'on', '#', 'simple',
         '#', 'data', '.']]

         tokenize_sentence: [['i am writing some very basic english sentences'], ["i'm
         just writing it for the demo purpose to make audience understand the basics
         ."], ['the point is to _learn how it works_ on #simple # data.']]
```

In the above image we can view the tokenized words and sentence below the python code.

<u>iii. Stopwords & Punctuation removal</u>

Text may contain words such as is, am, are, a etc. We would not want these words taking up space in our database or taking up processing time.

Step 3 - Punctuation Removal

In [19]:
```
#import string
import re
regex = re.compile('[%s]' % re.escape(string.punctuation))

tokenized_docs_no_punctuation = []

for review in tokenized_words:
    new_review = []
    for token in review:
        new_token = regex.sub(u'', token)
        if not new_token == u'':
            new_review.append(new_token)

    tokenized_docs_no_punctuation.append(new_review)

print(tokenized_docs_no_punctuation)
```

```
[['i', 'am', 'writing', 'some', 'very', 'basic', 'english', 'sentences'],
['i', 'm', 'just', 'writing', 'it', 'for', 'the', 'demo', 'purpose', 'to', 'm
ake', 'audience', 'understand', 'the', 'basics'], ['the', 'point', 'is', 't
o', 'learn', 'how', 'it', 'works', 'on', 'simple', 'data']]
```

In the above image all the punctuations (such as #,",_) has been removed from the doc.

Step 4 - Removing Stopwords

Stop words are considered as noise in the text. We would not want these words taking up space in our database or taking up the processing time.

In [35]:
```
nltk.download('stopwords')
from nltk.corpus import stopwords

tokenized_docs_no_stopwords = []

for doc in tokenized_docs_no_punctuation:
    new_term_vector = []
    for word in doc:
        if not word in stopwords.words('english'):
            new_term_vector.append(word)

    tokenized_docs_no_stopwords.append(new_term_vector)

print(tokenized_docs_no_stopwords)
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\admin\AppData\Roaming\nltk_data...

[['writing', 'basic', 'english', 'sentences'], ['writing', 'demo', 'purpose',
'make', 'audience', 'understand', 'basics'], ['point', 'learn', 'works', 'sim
ple', 'data']]

[nltk_data]   Package stopwords is already up-to-date!
```

All the stop words from the doc is removed and the final output is "['writing', 'basic', 'english',

'sentences'], ['writing', 'demo', 'purpose', 'make', 'audience', 'understand', 'basics'], ['point', 'learn', 'works', 'simple', 'data']"

iv. Stemming and Lemmatization

Lemmatization is a process of converting a word to its base form. In stemming last few characters of the words are removed.

Below is the example of lemmatization

```
[['writing', 'basic', 'english', 'sentence'], ['writing', 'demo', 'purpose', 'make', 'audience', 'understand', 'basic'], ['poin
t', 'learn', 'work', 'simple', 'data']]
```

Below is the example of stemming

```
[['write', 'basic', 'english', 'sentenc'], ['write', 'demo', 'purpos', 'make', 'audienc', 'understand', 'basic'], ['point', 'le
arn', 'work', 'simpl', 'data']]
```

We can clearly see the difference in both the example.

v. Bag of words (BOW) & Tf-Idf

a. Bag of words (BOW)

The bag of words collects all the unique words from the document, arranges the words in an ascending order and give us the information whether the particular word appears in any other sentence by values 0 & 1. 1 denotes that the word appears and vice-versa. Let's make this clear by below example (referred from YouTube channel 'unfold data science'),

Suppose we have two sentence text & text1

```
from sklearn.feature_extraction.text import CountVectorizer
# list of text documents
text = ["hello, my name is Neel and I am aspiring data Scientist."]
text1 = ["hello You are watching unfold data science"]
```

After applying bag of words on the text the words in the sentence are arranged in ascending order and basic stop word removal too takes place (see the image below).

```
print(vectorizer.vocabulary_)
```

```
{'hello': 4, 'my': 6, 'name': 7, 'is': 5, 'neel': 8, 'and': 1, 'am': 0, 'aspi
ring': 2, 'data': 3, 'scientist': 9}
```

```
from IPython.display import Image
Image(filename='[redacted]')
```

0	1	2	3	4	5	6	7	8	9
am	and	aspiring	data	hello	is	my	name	neel	scientist

Now let's see the common words in both sentence (text & text1)

```
# encode document
newvector = vectorizer.transform(text1)

# summarize encoded vector
print(newvector.toarray())
```

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

In the above image we see the output is [[0 0 0 1 1 0 0 0 0 0]]. In the 3rd & 4th index we have value 1; the text has word 'data' & 'hello' in 3rd & 4th index (In data science the index starts from 0). It says that the word 'data' & 'hello' is common in both the sentence.

b. Tf-IDF

Tf-Idf evaluates how relevant a word is to a document and also to find out the unique words.

For example let's consider the text in the below image

```
from sklearn.feature_extraction.text import TfidfVectorizer
# list of text documents
text = ["Neel is a aspiring data scientist in India","This is unfold data scie
nce","Data Science is a promising career"]
```

After applying bag of words and Tf-Idf to the above text we get the below result,

```
from IPython.display import Image
Image(filename=[redacted])
```

0	1	2	3	4	5	6	7	8	9	10	11
aspiring	career	data	in	india	is	neel	promising	science	scientist	this	unfold

---

```
#Focus on IDF VALUES
print(vectorizer.idf_)
```

```
[1.69314718 1.69314718 1.         1.69314718 1.69314718 1.
 1.69314718 1.69314718 1.28768207 1.69314718 1.69314718 1.69314718]
```

According to the output the words ‘data’ and ‘is’ are appearing maximum time that’s why their score is minimum. The words with maximum score are more relevant.

Conclusion

We have seen basic steps performed in data science analysis. Through this blog we got an overview of data science. The main purpose of this blog was to make the reader understand basic data science in common layman language. Once you get deep understanding of the basic it will be easy for you to understand the various models used for predicting the results.

(***credits****: A big thank you to Applied AI team for putting their efforts in making concepts clear in easy way through their lectures.*

Below sources also helped in learning and inspiring content for this blog,

1. Youtube channel – Krish Naik, Unfold data science & Code basic

2. Website – Medium, Towards data science & analytics vidya.)