

**Московский авиационный институт
(Национальный исследовательский университет)**

Факультет: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Искусственный интеллект»

Лабораторная работа № 0

Студент: Дюсекеев А.Е.

Группа: М80-304Б

Москва, 2020

Постановка задачи

Требуется сформировать/получить два набора данных соответствующие следующим критериям:

- 1) Один из датасетов должен представлять собой корпус документов. Язык, источник и тематика произвольна
- 2) Второй датасет должен содержать категориальные, количественные признаки. Для данного датасета определить предсказываемые признаки (для задачи регрессии и классификации). Если такого признака нет, спроектировать

Данные датасеты будут в дальнейшем использованы в оставшихся лабораторных работах.

По каждому датасету построить распределения признаков (в случае корпуса документов – построить распределение слов) и объяснить имеющуюся картину. Вычислить статистические характеристики признаков. Обнаружить и решить возможные проблемы с данными. Если решить данную проблему невозможно, объяснить почему.

Решение задач

Программы написаны на языке программирования **python 3.7.2**.

Были использованы следующие **библиотеки**:

- **matplotlib** - библиотека на языке программирования Python для визуализации данных двумерной (2D) графикой (3D графика также поддерживается).
- **numpy** - это open-source модуль для python, который предоставляет общие математические и числовые операции в виде пре-скомпилированных, быстрых функций.
- **pandas** - мощный инструмент для анализа данных.
- **statistics** - статистические функции.
- **re** - предназначен для работы с регулярными выражениями

Использованные **датасеты**:

- <https://finance.yahoo.com/quote/MSFT/history?period1=511045200&period2=1560891600&interval=1d&filter=history&frequency=1d>
- <https://www.kaggle.com/aadityanaik/shakespeareworks/downloads/shakespeareworks.zip>

Одна из важных статей:

- <https://tproger.ru/translations/basic-statistics-in-python-descriptive-statistics/>
- <https://habr.com/ru/post/349860/>

```
import matplotlib.pyplot as plt, numpy as np
from pandas import read_csv
from statistics import mode
file = read_csv('MSFT.csv')
title = ['Open','High','Low','Adj Close','Volume']

x = file[title].values
y = file['Close'].values
n = len(title)
```

```
for i in range(0, n):
    plt.xlabel(title[i])
    plt.ylabel('Close')
    plt.plot(file[title[i]].values, y, 'ro')
    plt.show()
```

#Mean value is a characteristic that describes the average value in a data set. In the case of the average value of the “middle” of the dataset,

the arithmetic average of its values will be. The average value reflects a typical indicator in the data set.

If we randomly select one of the indicators, then we will most likely get a value close to the average.

```
sum = sum(file['Close'])
print(sum)
num = len(file['Close'])
print(num)
avg = sum/num
print('Средняя цена',avg)
```

#The median is a measure of central tendency.

#It is needed to determine typical values in a data set, but it does not require calculations.

```
middle = len(file['Close'])/2+0.5
list_sorted=sorted(file['Close'])
mediane = list_sorted[int(middle)]
print('Median',mediane)
```

#Min&Max of a function are the largest and smallest value of the function

```
max = max(file['Close'])
print('Max',max)
min = min(file['Close'])
print('Min',min)
```

#Mode is defined as the value that is most commonly found in a data set.

```
mode = mode(file['Close'])
print('Mode',mode)
```

#Range of a set of data is the difference between the largest and smallest values.

It is the first characteristic that answers the question “How much does my data vary?”.

```
r = max - min
print('Range', r)
```

#Standard deviation is also a measure of data variation. shows how much the data differ from the arithmetic mean.

```
sd = np.std(file['Close'])
print('Standard deviation',sd)
```

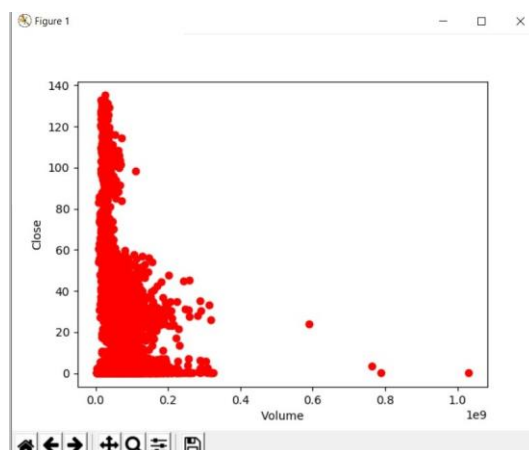
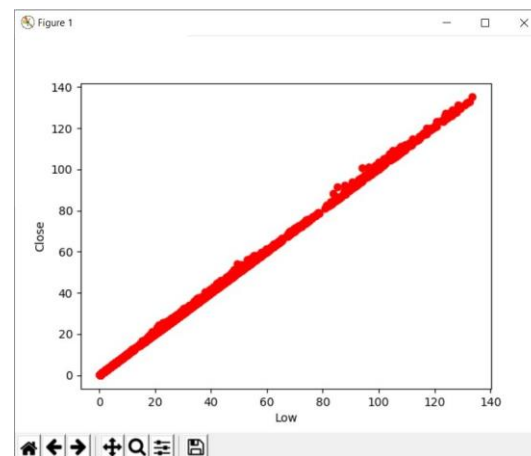
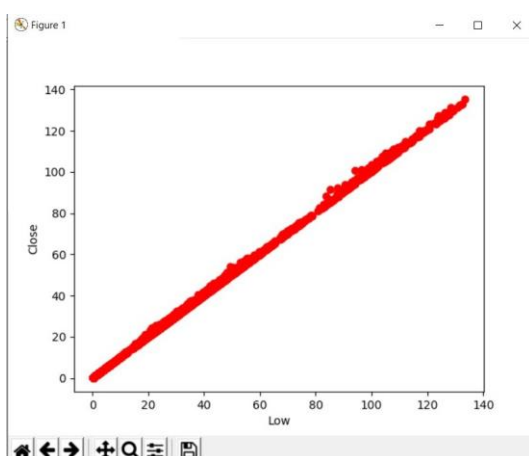
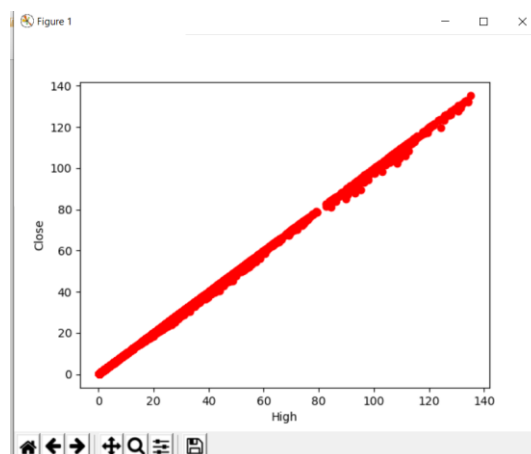
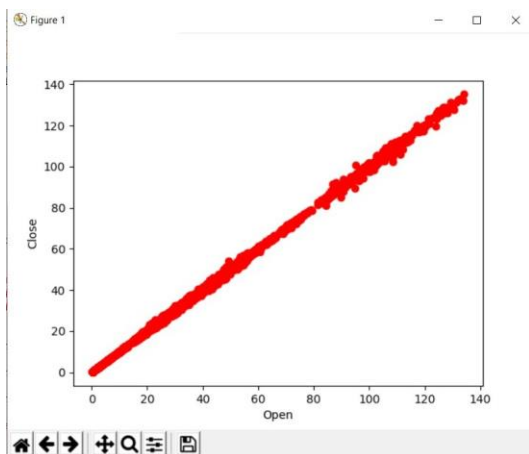
#Dispersion is simply the square of the standard deviation. It reflects the measure of dispersion.

```
d = sd * sd
print('Dispersion',d)
```

Результат работы

```
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Alex\Desktop\solution.py =====
220692.18524999983
8385
Средняя цена 26.319878980321985
Median 26.030001000000002
Max 135.16000400000001
Min 0.090278
Mode 0.361111
Range 135.069726
Standard deviation 24.723854515739685
Dispersion 611.2689821154617
>>> |
```

На рисунках, приведенных ниже, изображены различные распределения.



Я исследовал историю акций **Microsoft Corporation (MSFT)**.

Можно сделать следующие выводы:

- Достаточно предсказуемыми являются цена акции на момент закрытия биржи, а так же максимальная и минимальная цена акции в торговый день.
- Отличий между ценой на момент закрытия и скорректированной ценой практически нет. Скорректированная цена может отличаться денежными дивидендами, дивидендами по акциям. Стоит отметить, что скорректированная цена закрытия акций также отражает предложения о правах, которые могут возникнуть. Предоставление прав - это вопрос прав, предоставляемых существующим акционерам, что дает

- акционерам право подписки на выпуск прав пропорционально их акциям.
- Цена закрытия далеко не всегда имеет значительное отличие от максимальной цены за день. Выбросы — исключения из правил, например, падение рынка, локальные неудачи; аналогично с минимальной ценой, там выбросы вверх. Они могут быть обусловлены ростом рынка, либо локальной удачей.
 - Цена открытия и закрытия рынка относится к числу важнейших моментов, по которым трейдеры вполне могут судить о настоящей рыночной ситуации. Именно эти оба показателя, зачастую позволяют составить правильный, краткосрочный трендовый прогноз и предугадать направление предстоящих сделок, а также определить исходные точки входа и выхода в торговых сессиях. На рисунке мы имеем некоторую усредненную версию двух предыдущих графиков.

Прелюдия к исходному коду для 2 датасета

Был взят датасет с работами Квентина Тарантино.

django_unchained_script
inglorious_basterds_script
pulp_fiction_script
reservoir_dogs_screenplay
tarantino_scripts

Все эти файлы были объединены в один, названный впоследствии t.txt. Нам интересно посмотреть на гистограмму, которая показывает распределение слов.

Исходный код для 2 датасета

```
import re #for regular expressions
import string
import matplotlib.pyplot as plt
frequency = {} # объявляем частотную переменную
document_text = open('t.txt', 'r') #подключаем файл с всем содержимым
text_string = document_text.read().lower() #преобразовываем файл в строку в нижнем регистре
match_pattern = re.findall(r'\b[a-z]{3,15}\b', text_string)

for word in match_pattern: #идем по полученному массиву слов
    count = frequency.get(word,0) #ищем вхождения слова
    frequency[word] = count + 1

frequency_list = frequency.keys()
list = []
for words in frequency_list:
    list.append([words, frequency[words]]) #создаем массив объектов вида {слово, частота}
    #print (words, frequency[words])

list.sort(key=lambda x: x[1])
```

```

#print(list[-5:])

s=[]
n=[]
for i in reversed(list[-5:]): #тут повернули список
    s.append(i[1])
    n.append(i[0])

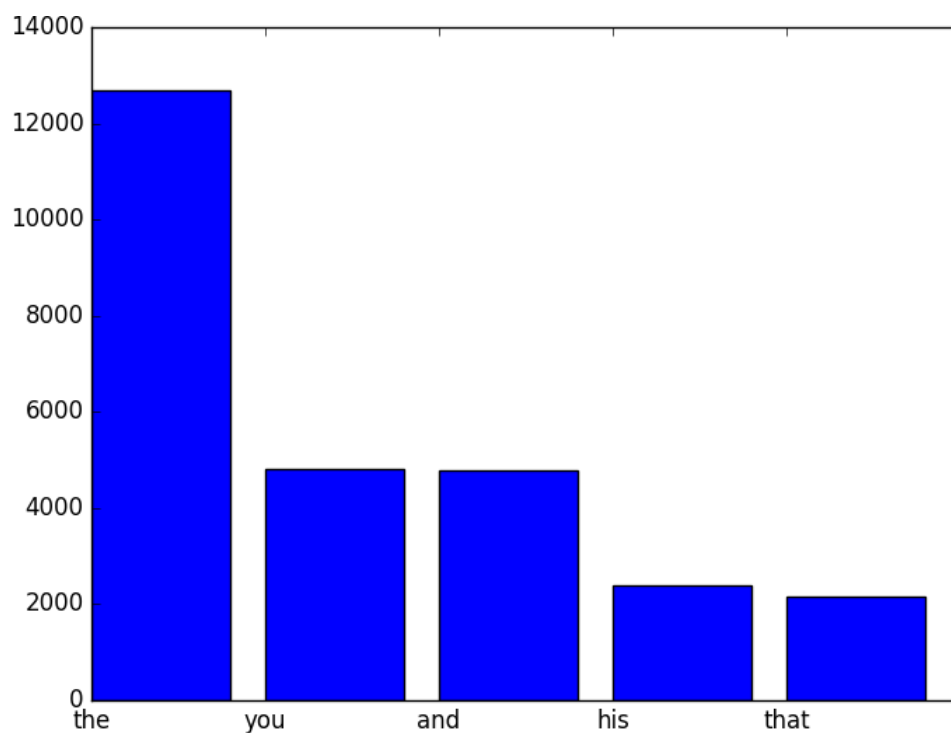
    ## s и n - это слово и его частота.
#print(s)
#print(n)

x=range(len(s))

ax = plt.gca()
ax.bar(x, s, align='edge')
ax.set_xticks(x)
ax.set_xticklabels(n)
plt.show()

```

Результат работы



Больше всего встречается определенный артикль ***the***, а так же местоимения ***you***, ***his*** и служебные части речи, такие как ***and*** и ***that***, что, вообще говоря, не удивительно для английского языка.