**LAPORAN FINAL PROJECT DSC**

**(DATA SAINS CLUB) 2022**

****

**OLEH**

**HEPPY REIHANIA KARTIKA SARI**

**192400019**

**PROGRAM STUDI STATISTIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS PGRI ADI BUANA SURABAYA**

**2022**

**DESKRIPSI DATASET**

Klasifikasi merupakan proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui. Pada analisis ini saya menggunakan klasifikasi dengan metode Naïve Bayesian dan data yang saya pada tugas final project adalah data sekunder. Sampel yang saya pakai sebanyak 21 sampel, dengan 6 variabel. Berikut deskripsi dataset.

|  |  |  |
| --- | --- | --- |
| No. | Nama | Variabel |
| 1. | Jurusan | Y |
| 2. | Matematika | X1 |
| 3. | Bahasa Inggris | X2 |
| 4. | Bahasa Indonesia | X3 |
| 5. | Prestasi Siswa | X4 |
| 6. | Prestasi Sekolah | X5 |

**DATA PREPROCESSING**

* Langkah pertama yang harus dilakukan adalah kita mengimportt library yang dibutuhkan

**import** pandas **as** pd

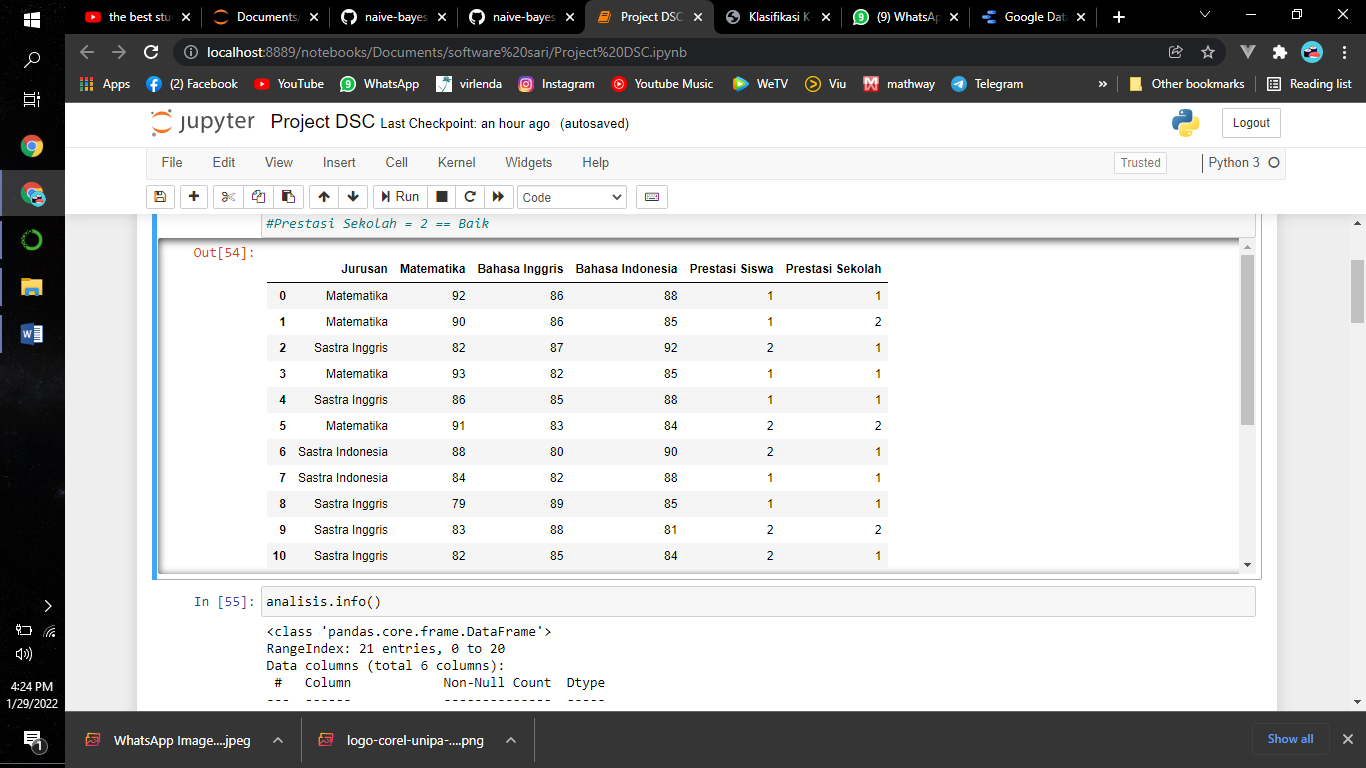
**import** numpy **as** np

**import** matplotlib.pyplot **as** plt

* Langka kedua yang harus dilakukan yaitu memanggil data yang sudah kita simpan dengan syntax seperti berikut :

analisis = pd.read\_csv("DSC.csv")

analisis



* Kemudian jika ingin melihat tipe data, dapat menggunakan syntax

analisis.info()

**[out]** <class 'pandas.core.frame.DataFrame'>

RangeIndex: 21 entries, 0 to 20

Data columns (total 6 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Jurusan 21 non-null object

1 Matematika 21 non-null int64

2 Bahasa Inggris 21 non-null int64

3 Bahasa Indonesia 21 non-null int64

4 Prestasi Siswa 21 non-null int64

5 Prestasi Sekolah 21 non-null int64

dtypes: int64(5), object(1)

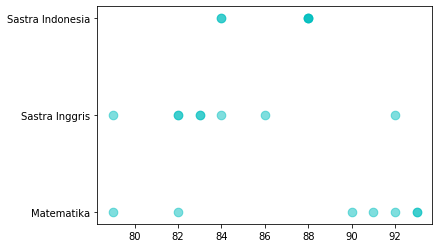
memory usage: 1.1+ KB

* Untuk memvisualisasikan persebaran data menggunakan syntax

from sklearn.cluster import KMeans

plt.scatter(analisis.Matematika, analisis.Jurusan, s = 75, c = "c", marker = "o", alpha = 0.5)

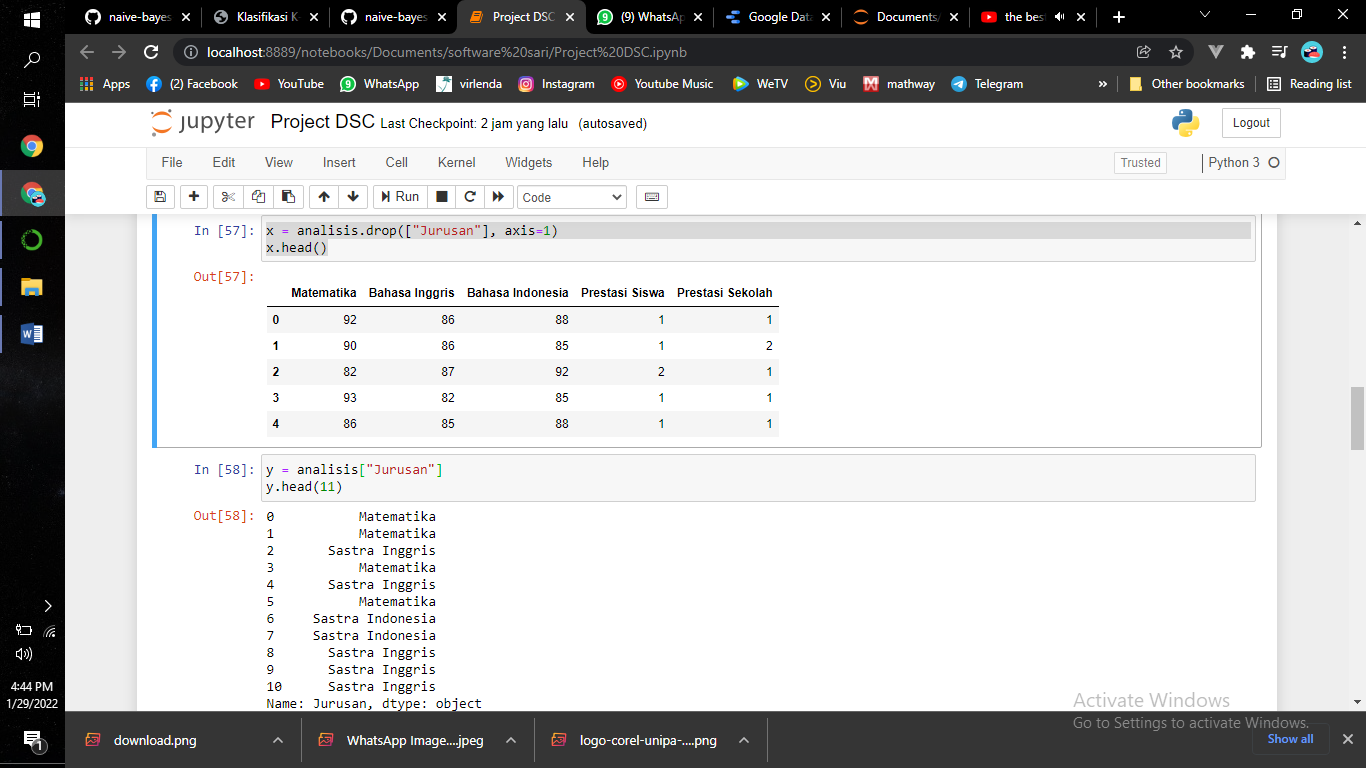
plt.show()



* Kemudian menentukan variabel independennya yang terdiri dari matematika, Bahasa Inggris, Bahasa Indonesia, prestasi siswa, prestasi sekolah. Dan varibel jurusan akan di keluarkan dengan menggunakan syntax sebagai berikut.

x = analisis.drop(["Jurusan"], axis=1)

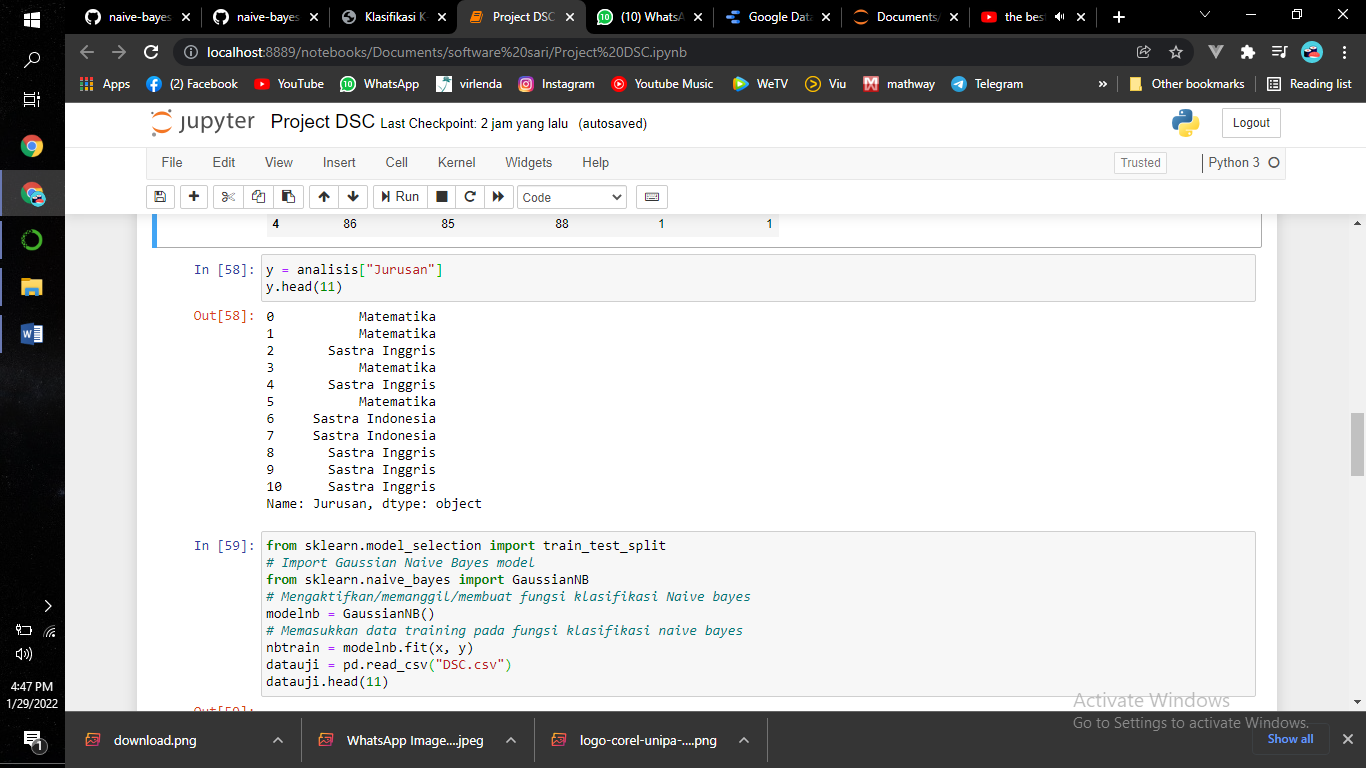
x.head()



* Kemudian menentukkan variable dependennya yaitu lulus cepat

y = analisis["Jurusan"]

y



* Mengimport package model selection dari SKlearn,

from sklearn.model\_selection import train\_test\_split

* Mengimport Import Gaussian Naive Bayes model

from sklearn.naive\_bayes import GaussianNB

* Memanggil/membuat fungsi klasifikasi Naive bayes

modelnb = GaussianNB()

* Memasukkan data training pada fungsi klasifikasi naive bayes

nbtrain = modelnb.fit(x, y)

datauji = pd.read\_csv("DSC.csv")

datauji.head()

* Kemudian menentukkan prediksi atau peramalannya dengan menggunakan sytax sebagai berikut

Y\_predict = nbtrain.predict(x\_test)

print("Prediksi Naive Bayes : ",Y\_predict)

[OUT] : Prediksi Naive Bayes : ['Sastra Indonesia' 'Matematika' 'Sastra Indonesia' 'Sastra Indonesia' 'Sastra Indonesia' 'Matematika' 'Sastra Indonesia' 'Sastra Indonesia' 'Sastra Inggris' 'Sastra Inggris' 'Sastra Indonesia' 'Sastra Inggris' 'Matematika' 'Sastra Indonesia' 'Sastra Indonesia' 'Sastra Inggris' 'Sastra Inggris' 'Sastra Indonesia' 'Sastra Indonesia' 'Sastra Indonesia' 'Sastra Indonesia']

* Kemudian menghitung nilai akurasi naïve bayes

**from** sklearn.metrics **import** accuracy\_score

accuracy**=** accuracy\_score(y\_uji, Y\_predict)

print("Akurasi Naive Bayes : ",accuracy)

[OUT] : Akurasi Naive Bayes : 0.6190476190476191

* Menghitung nilai akurasi dari klasifikasi naive bayes dengan syntax

**from** sklearn.metrics **import** classification\_report

print(classification\_report(y\_uji, Y\_predict))

[OUT] :

precision recall f1-score support

Matematika 1.00 0.43 0.60 7

Sastra Indonesia 0.46 1.00 0.63 6

Sastra Inggris 0.80 0.50 0.62 8

accuracy 0.62 21

macro avg 0.75 0.64 0.62 21

weighted avg 0.77 0.62 0.61 21

**MODELLING DAN EVALUATING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-Score | Support |
|  |  |  |  |  |
| Matematika | 1 | 0.43 | 0.6 | 7 |
| Sastra Indonesia | 0.46 | 1 | 0.63 | 6 |
| Sastra Inggris | 0.8 | 0.5 | 0.62 | 8 |
|  |  |  |  |  |
| accuracy |  |  | 0.62 | 21 |
| macro avg | 0.75 | 0.64 | 0.62 | 21 |
| weighted avg | 0.77 | 0.62 | **0.61** | 21 |

Dari table diatas dapat diketahui bahwwa nilai precision atau nilai akurasi yang didapat sebesar 0,61 yang artinya nilai keakuratan hasil prediksi yang didapat sebesar 61%

**KESIMPULAN**

Berdasarkan hasil output terakhir dari running syntax yang saya lakukan di python menghasilkan nilai nilai precision atau nilai akurasi yang didapat sebesar 0,61 yang artinya nilai keakuratan hasil prediksi yang didapat sebesar 61% dari data yang sudah saya analisis dari awal hingga terakhir.