Tugas 9: Judul tugas Naïve Bayes

Siti aisah -0110222129 [1*]

[1] Teknik Informatika, STT Terpadu Nurul Fikri, Depok
[2] Sistem Informasi, STT Terpadu Nurul Fikri, Depok

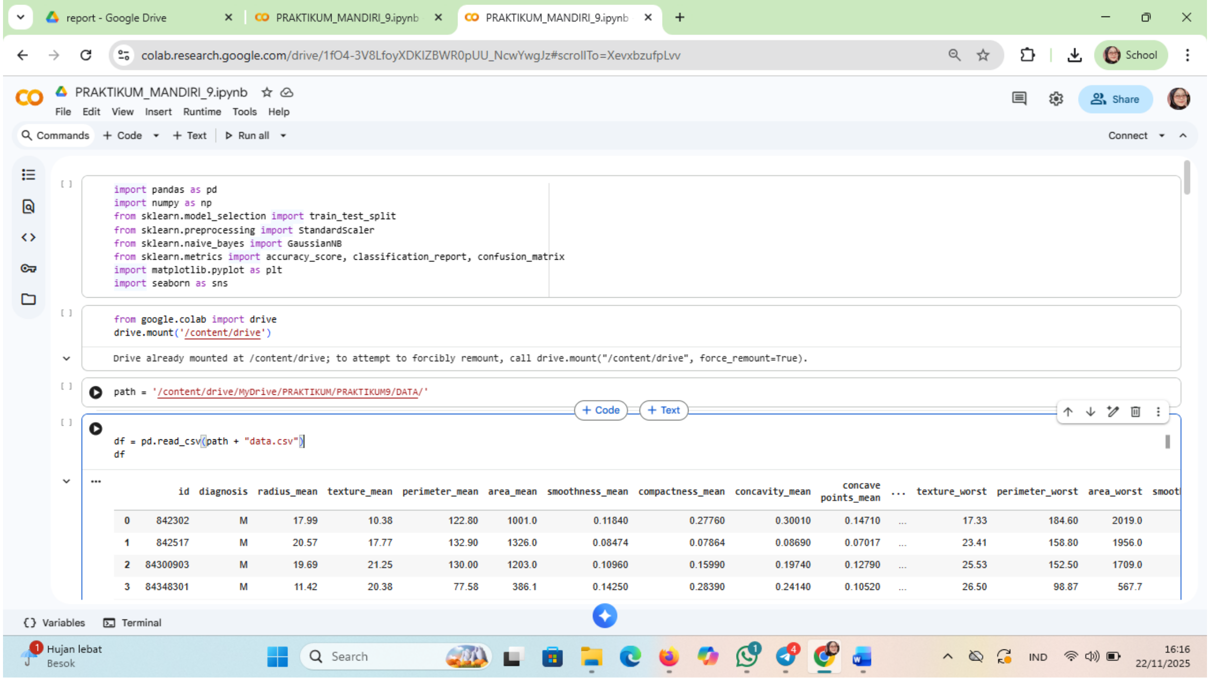
*E-mail: siti22129ti@student.nurulfikri.ac.id

Abstract.

Machine Learning (Pembelajaran Mesin) adalah cabang dari Kecerdasan Buatan (AI) yang memungkinkan sistem untuk belajar secara mandiri dari data, mengenali pola, dan membuat keputusan atau prediksi tanpa diprogram secara **eksplisit** untuk setiap

Tugas mandiri 1

1.Menyiapkan semua *tools* (library) yang dibutuhkan dan Mengakses data dari Google Drive.



report - Google Drive
PRAKTIKUM_MANDIRI_9.ipynb
PRAKTIKUM_MANDIRI_9.ipynb
colab.research.google.com/drive/1fO4-3V8LfoyXDKIZBWR0pUU_NcwYwgJz#scrollTo=XevxbzufpLvv
School
PRAKTIKUM_MANDIRI_9.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
Connect
+ Code + Text
Variables Terminal
Hujan lebat
Besok
Search
IND
16:16
22/11/2025

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

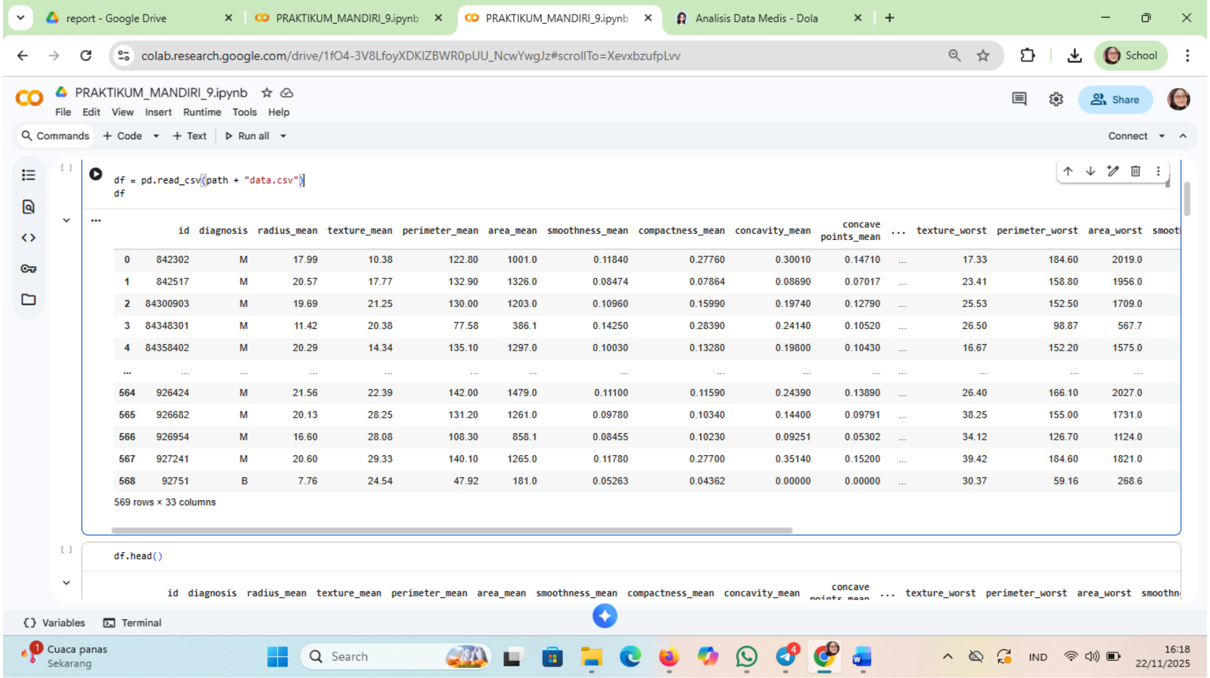
```
path = '/content/drive/MyDrive/PRAKTIKUM/PRAKTIKUM9/DATA/'
```

```
df = pd.read_csv(path + "data.csv")
df
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texture_worst	perimeter_worst	area_worst	smoot
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	...	17.33	184.60	2019.0	
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	...	23.41	158.80	1956.0	
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	...	25.53	152.50	1709.0	
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	...	26.50	98.87	567.7	

1.1 hasil dari menyiapkan semua tools dan menghubungkan ke drive

2. melihat data



```
df = pd.read_csv(path + "data.csv")
df
```

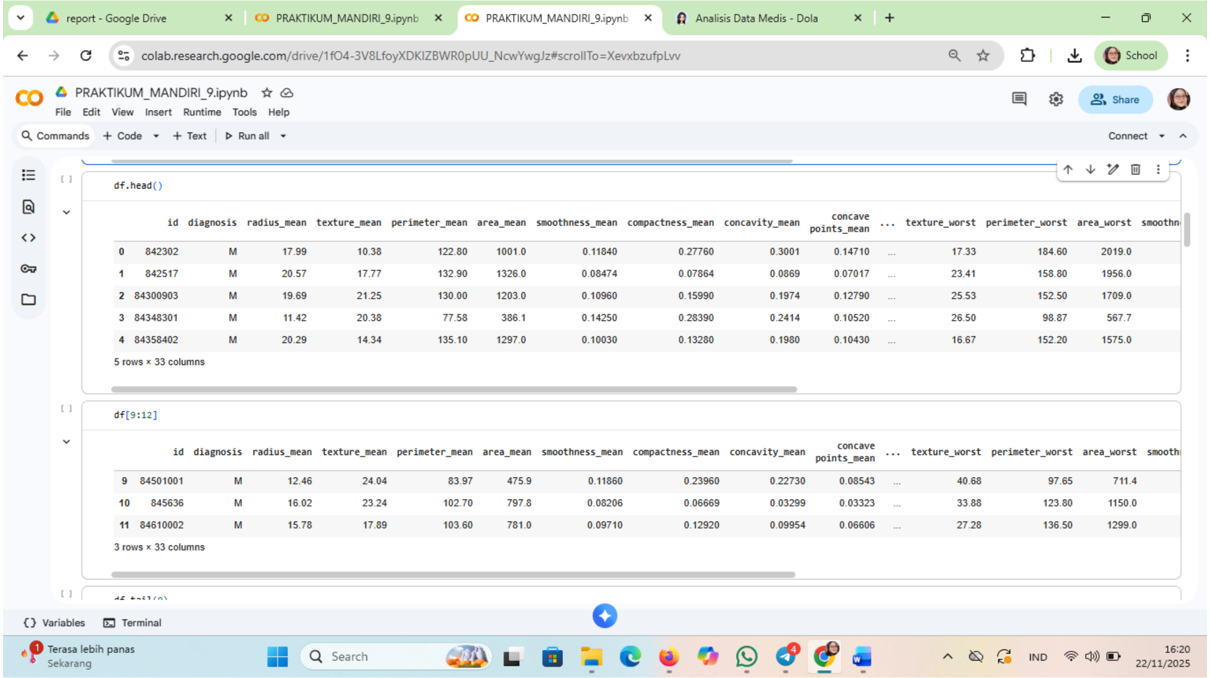
	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texture_worst	perimeter_worst	area_worst	smoot
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	...	17.33	184.60	2019.0	
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	...	23.41	158.80	1956.0	
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	...	25.53	152.50	1709.0	
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	...	26.50	98.87	567.7	
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	...	16.67	152.20	1575.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
564	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	...	26.40	166.10	2027.0	
565	926682	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	...	38.25	155.00	1731.0	
566	926954	M	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	...	34.12	126.70	1124.0	
567	927241	M	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	...	39.42	184.60	1821.0	
568	92751	B	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	...	30.37	59.16	268.6	

569 rows × 33 columns

```
df.head()
```

1.2 hasil dari melihat data

3. menampilkan sejumlah baris pertama



```
df.head()
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texture_worst	perimeter_worst	area_worst	smoothn
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	...	17.33	184.60	2019.0	
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	...	23.41	158.80	1956.0	
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	...	25.53	152.50	1709.0	
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	...	26.50	98.87	567.7	
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	...	16.67	152.20	1575.0	

5 rows × 33 columns

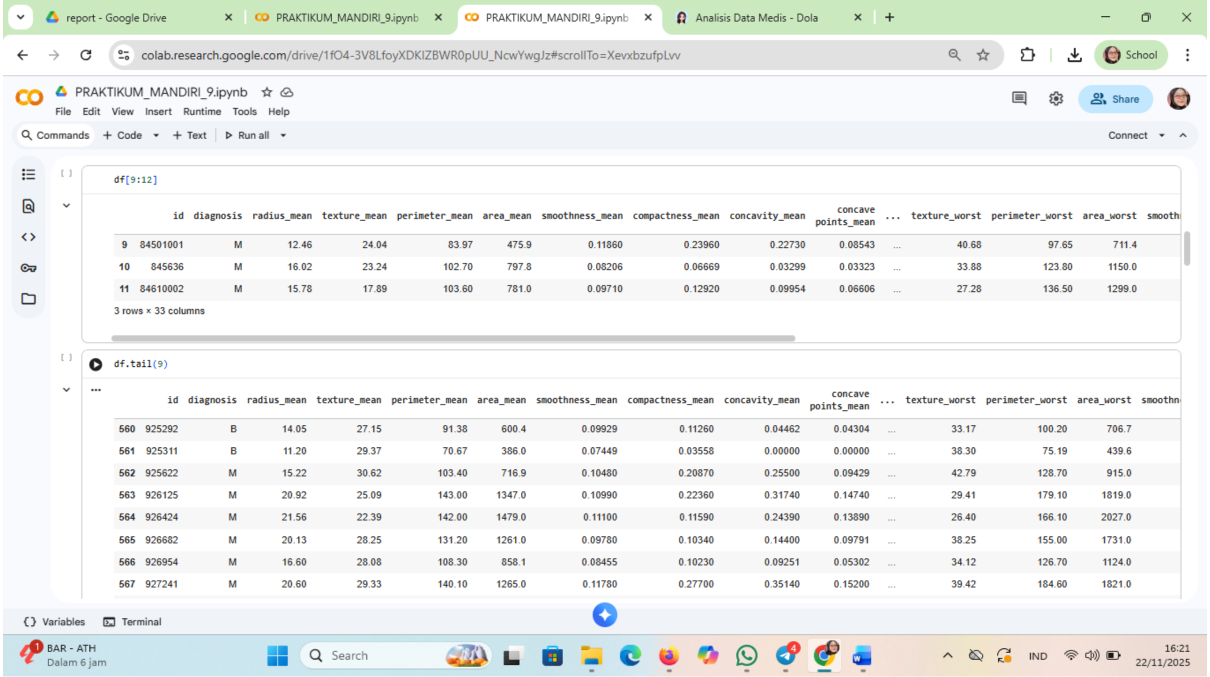
```
df[9:12]
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texture_worst	perimeter_worst	area_worst	smooth
9	84501001	M	12.46	24.04	83.97	475.9	0.11860	0.23960	0.22730	0.08543	...	40.68	97.65	711.4	
10	845636	M	16.02	23.24	102.70	797.8	0.08206	0.06669	0.03299	0.03323	...	33.88	123.80	1150.0	
11	84610002	M	15.78	17.89	103.60	781.0	0.09710	0.12920	0.09954	0.06606	...	27.28	136.50	1299.0	

3 rows × 33 columns

***1.3.** menampilkan baris petama*

4. menampilkan sejumlah baris terakhir



```
df[9:12]
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texture_worst	perimeter_worst	area_worst	smooth
9	84501001	M	12.46	24.04	83.97	475.9	0.11860	0.23960	0.22730	0.08543	...	40.68	97.65	711.4	
10	845636	M	16.02	23.24	102.70	797.8	0.08206	0.06669	0.03299	0.03323	...	33.88	123.80	1150.0	
11	84610002	M	15.78	17.89	103.60	781.0	0.09710	0.12920	0.09954	0.06606	...	27.28	136.50	1299.0	

3 rows × 33 columns

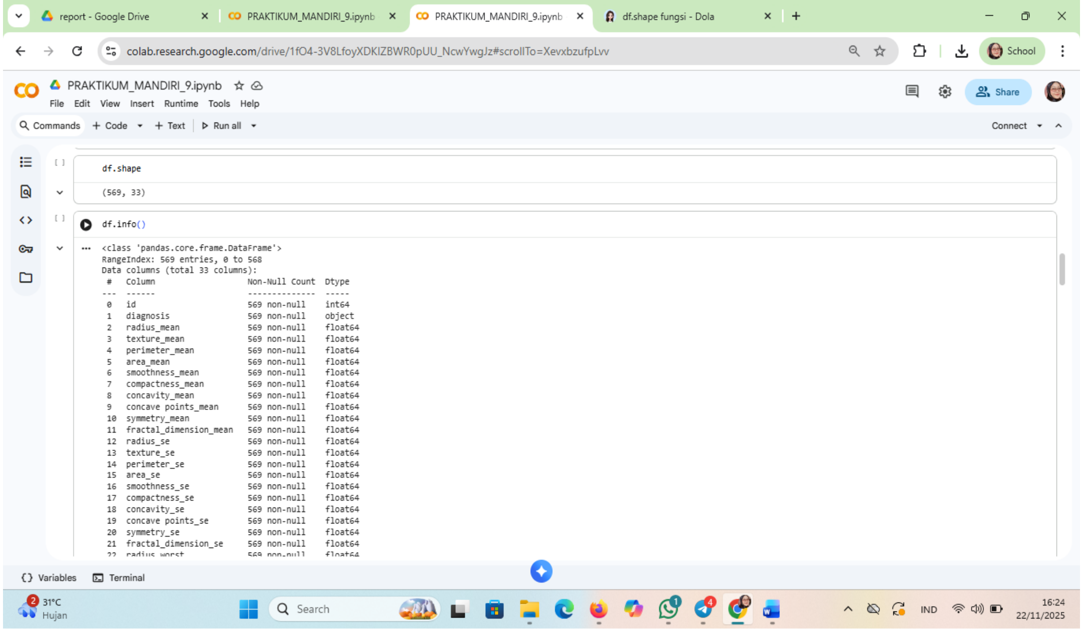
```
df.tail(9)
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texture_worst	perimeter_worst	area_worst	smoothn
560	925292	B	14.05	27.15	91.38	600.4	0.09929	0.11260	0.04462	0.04304	...	33.17	100.20	706.7	
561	925311	B	11.20	29.37	70.67	386.0	0.07449	0.03558	0.00000	0.00000	...	38.30	75.19	439.6	
562	925622	M	15.22	30.62	103.40	716.9	0.10480	0.20870	0.25500	0.09429	...	42.79	128.70	915.0	
563	926125	M	20.92	25.09	143.00	1347.0	0.10990	0.22360	0.31740	0.14740	...	29.41	179.10	1819.0	
564	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	...	26.40	166.10	2027.0	
565	926682	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	...	38.25	155.00	1731.0	
566	926954	M	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	...	34.12	126.70	1124.0	
567	927241	M	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	...	39.42	184.60	1821.0	

report - Google Drive
PRAKTIKUM_MANDIRI_9.ipynb
PRAKTIKUM_MANDIRI_9.ipynb
Analisis Data Medis - Dola
colab.research.google.com/drive/1fO4-3V8LfoyXDKIZBWR0pUU_NcwYwgJz#scrollTo=XevxbzufpLvv
School
PRAKTIKUM_MANDIRI_9.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text ▷ Run all
Connect
Variables
Terminal
BAR - ATH
Dalam 6 jam
Search
IND
16:21
22/11/2025

*1.4 **.** menampilkan sejumlah baris terakhir*

5. mengetahui dimensi dari sebuah DataFrame pada library Pandas


report - Google Drive
PRAKTIKUM_MANDIRI_9.ipynb
PRAKTIKUM_MANDIRI_9.ipynb
df.shape fungsi - Dola
colab.research.google.com/drive/1fO4-3V8LfoyXDKIZBWR0pUU_NcwYwgJz#scrollTo=XevxbzufpLvv
School
PRAKTIKUM_MANDIRI_9.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Share
Connect
Variables Terminal
31°C
Hujan
Search
IND
16:24
22/11/2025

```
df.shape
```

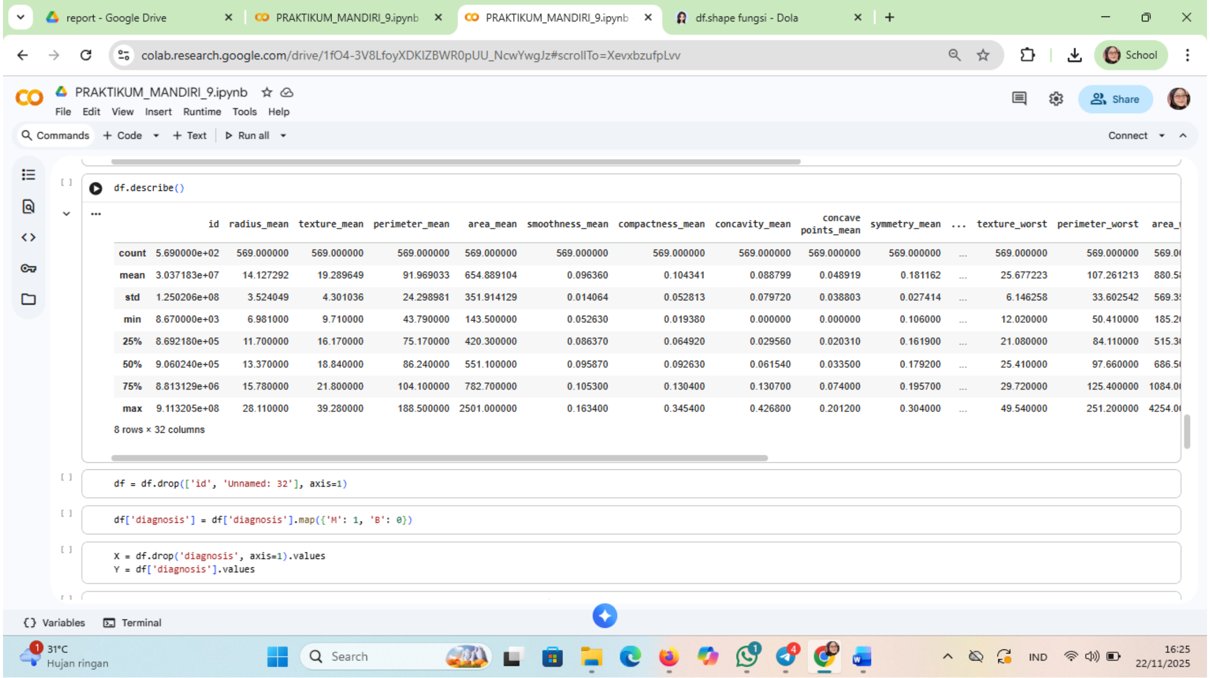
```
(569, 33)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   id                       569 non-null    int64
 1   diagnosis                569 non-null    object
 2   radius_mean              569 non-null    float64
 3   texture_mean             569 non-null    float64
 4   perimeter_mean           569 non-null    float64
 5   area_mean                569 non-null    float64
 6   smoothness_mean          569 non-null    float64
 7   compactness_mean         569 non-null    float64
 8   concavity_mean           569 non-null    float64
 9   concave points_mean      569 non-null    float64
 10  symmetry_mean            569 non-null    float64
 11  fractal_dimension_mean   569 non-null    float64
 12  radius_se                569 non-null    float64
 13  texture_se               569 non-null    float64
 14  perimeter_se             569 non-null    float64
 15  area_se                  569 non-null    float64
 16  smoothness_se            569 non-null    float64
 17  compactness_se           569 non-null    float64
 18  concavity_se             569 non-null    float64
 19  concave points_se        569 non-null    float64
 20  symmetry_se              569 non-null    float64
 21  fractal_dimension_se     569 non-null    float64
 22  radius_worst             569 non-null    float64
```

1.5 mengetahui dimensi dari sebuah DataFrame pada library Pandas

6. menampilkan statistik deskriptif dari sebuah DataFrame pada library Pandas



```
df.describe()
```

	id	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	...	texture_worst	perimeter_worst	area_
count	5.690000e+02	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	...	569.000000	569.000000	569.0
mean	3.037183e+07	14.127292	19.289649	91.969033	654.889104	0.096360	0.104341	0.088799	0.048919	0.181162	...	25.677223	107.261213	880.5
std	1.250206e+08	3.524049	4.301036	24.298981	351.914129	0.014064	0.052813	0.079720	0.038803	0.027414	...	6.146258	33.602542	569.3
min	8.670000e+03	6.981000	9.710000	43.790000	143.500000	0.052630	0.019380	0.000000	0.000000	0.106000	...	12.020000	50.410000	185.2
25%	8.692180e+05	11.700000	16.170000	75.170000	420.300000	0.086370	0.064920	0.029560	0.020310	0.161900	...	21.080000	84.110000	515.3
50%	9.060240e+05	13.370000	18.840000	86.240000	551.100000	0.095870	0.092630	0.061540	0.033500	0.179200	...	25.410000	97.660000	686.5
75%	8.813129e+06	15.780000	21.800000	104.100000	782.700000	0.105300	0.130400	0.130700	0.074000	0.195700	...	29.720000	125.400000	1084.0
max	9.113205e+08	28.110000	39.280000	188.500000	2501.000000	0.163400	0.345400	0.426800	0.201200	0.304000	...	49.540000	251.200000	4254.0

8 rows × 32 columns

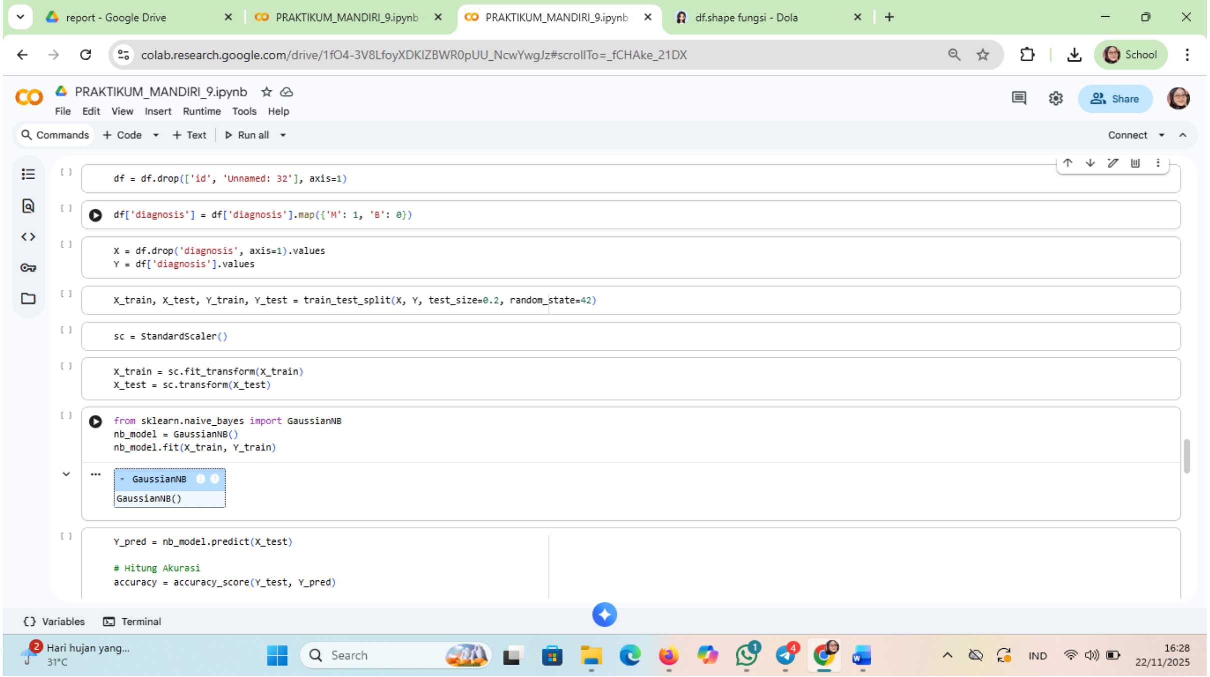
```
df = df.drop(['id', 'Unnamed: 32'], axis=1)
```

```
df['diagnosis'] = df['diagnosis'].map({'M': 1, 'B': 0})
```

```
X = df.drop('diagnosis', axis=1).values
Y = df['diagnosis'].values
```

1.6 menampilkan statistik deskriptif dari sebuah DataFrame pada library Pandas

7. membuat dan melatih model *machine learning* menggunakan algoritma *Gaussian Naive Bayes*



```
df = df.drop(['id', 'Unnamed: 32'], axis=1)
```

```
df['diagnosis'] = df['diagnosis'].map({'M': 1, 'B': 0})
```

```
X = df.drop('diagnosis', axis=1).values
Y = df['diagnosis'].values
```

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=42)
```

```
sc = StandardScaler()
```

```
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

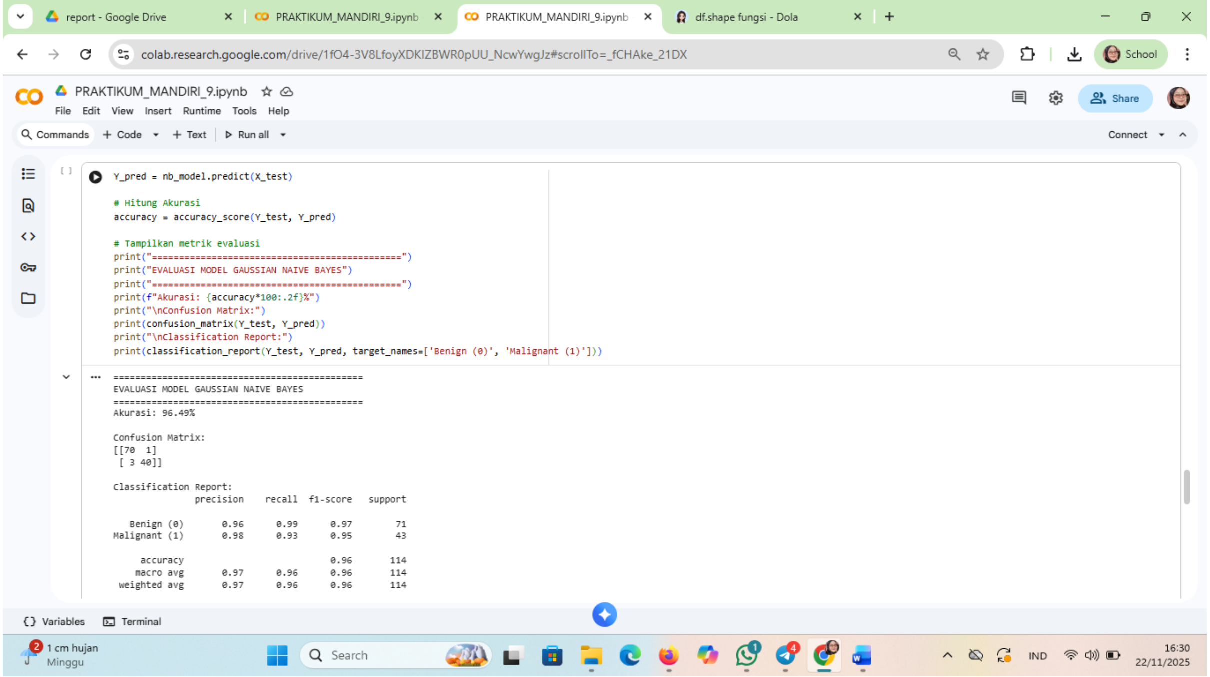
```
from sklearn.naive_bayes import GaussianNB
nb_model = GaussianNB()
nb_model.fit(X_train, Y_train)
```

```
GaussianNB
GaussianNB()
```

```
Y_pred = nb_model.predict(X_test)

# Hitung Akurasi
accuracy = accuracy_score(Y_test, Y_pred)
```

1.7 membuat dan melatih model machine learning menggunakan algoritma Gaussian Naive Bayes

8. mengevaluasi kinerja model *Gaussian Naive Bayes*



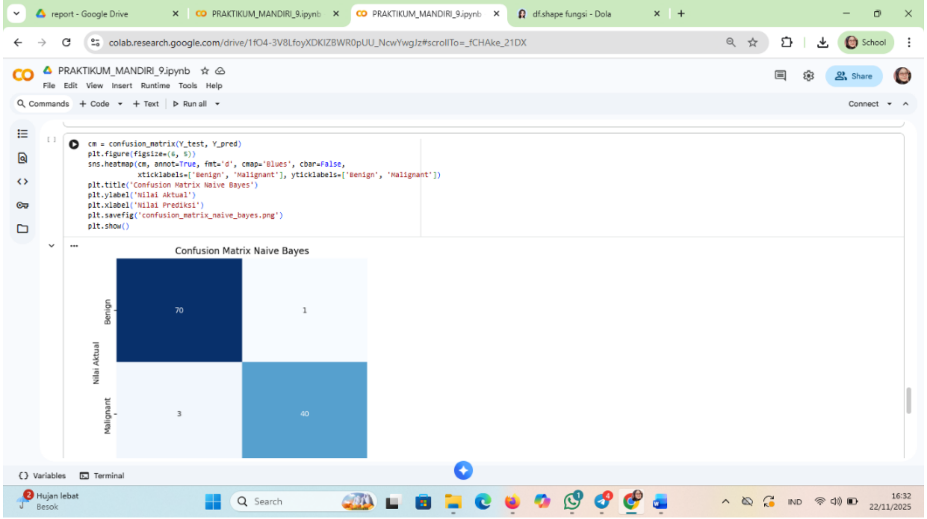
report - Google Drive
PRAKTIKUM_MANDIRI_9.ipynb
PRAKTIKUM_MANDIRI_9.ipynb
df.shape fungsi - Dola
colab.research.google.com/drive/1fO4-3V8LfoyXDKIZBWR0pUU_NcwYwgJz#scrollTo=_fCHAke_21DX
School
PRAKTIKUM_MANDIRI_9.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Share
Connect
Variables Terminal
1 cm hujan
Minggu
Search
IND
16:30
22/11/2025

```
Y_pred = nb_model.predict(X_test)

# Hitung Akurasi
accuracy = accuracy_score(Y_test, Y_pred)

# Tampilkan metrik evaluasi
print("=============================================")
print("EVALUASI MODEL GAUSSIAN NAIVE BAYES")
print("=============================================")
print(f"Akurasi: {accuracy*100:.2f}%")
print("\nConfusion Matrix:")
print(confusion_matrix(Y_test, Y_pred))
print("\nClassification Report:")
print(classification_report(Y_test, Y_pred, target_names=['Benign (0)', 'Malignant (1)']))
```

```
=============================================
EVALUASI MODEL GAUSSIAN NAIVE BAYES
=============================================
Akurasi: 96.49%

Confusion Matrix:
[[70  1]
 [ 3 40]]

Classification Report:
               precision    recall  f1-score   support

   Benign (0)       0.96      0.99      0.97        71
Malignant (1)       0.98      0.93      0.95        43

     accuracy                           0.96       114
    macro avg       0.97      0.96      0.96       114
 weighted avg       0.97      0.96      0.96       114
```

1.8 mengevaluasi kinerja model Gaussian Naive Bayes

9. untuk membuat dan menampilkan visualisasi *confusion matrix* dari hasil prediksi model *Gaussian Naive Bayes*.



report - Google Drive
PRAKTIKUM_MANDIRI_9.ipynb
PRAKTIKUM_MANDIRI_9.ipynb
df.shape fungsi - Dola
colab.research.google.com/drive/1fO4-3V8LfoyXDKIZBWR0pUU_NcwYwgJz#scrollTo=_fCHAke_21DX
School
PRAKTIKUM_MANDIRI_9.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Share
Connect
Confusion Matrix Naive Bayes
Benign
Malignant
Nilai Aktual
70
1
3
40
Variables Terminal
Hujan lebat
Besok
Search
IND
16:32
22/11/2025

```
cm = confusion_matrix(Y_test, Y_pred)
plt.figure(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False,
            xticklabels=['Benign', 'Malignant'], yticklabels=['Benign', 'Malignant'])
plt.title('Confusion Matrix Naive Bayes')
plt.ylabel('Nilai Aktual')
plt.xlabel('Nilai Prediksi')
plt.savefig('confusion_matrix_naive_bayes.png')
plt.show()
```

1.9 untuk membuat dan menampilkan visualisasi confusion matrix dari hasil prediksi model Gaussian Naive Bayes.

Referensi:

Munir, S., Seminar, K. B., Sudradjat, Sukoco, H., & Buono, A. (2022). The Use of Random Forest Regression for Estimating Leaf Nitrogen Content of Oil Palm Based on Sentinel 1-A Imagery. *Information*, *14*(1), 10. https://doi.org/10.3390/info14010010

Seminar, K. B., Imantho, H., Sudradjat, Yahya, S., Munir, S., Kaliana, I., Mei Haryadi, F., Noor Baroroh, A., Supriyanto, Handoyo, G. C., Kurnia Wijayanto, A., Ijang Wahyudin, C., Liyantono, Budiman, R., Bakir Pasaman, A., Rusiawan, D., & Sulastri. (2024). PreciPalm: An Intelligent System for Calculating Macronutrient Status and Fertilizer Recommendations for Oil Palm on Mineral Soils Based on a Precision Agriculture Approach. *Scientific World Journal*, *2024*(1). https://doi.org/10.1155/2024/1788726

LINK GITHUB: https://github.com/Sitiaisah1604/machine-learning