Classification algorithms

Isai Ambrocio - A01625101

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix, accuracy_score,
ConfusionMatrixDisplay
from sklearn.naive_bayes import GaussianNB
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
```

```
df = pd.read_csv("./Iris.csv")
```

```
df.head()
```

```
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
Species
0   1            5.1           3.5            1.4           0.2  Iris-
setosa
1   2            4.9           3.0            1.4           0.2  Iris-
setosa
2   3            4.7           3.2            1.3           0.2  Iris-
setosa
3   4            4.6           3.1            1.5           0.2  Iris-
setosa
4   5            5.0           3.6            1.4           0.2  Iris-
setosa
```

```
df.drop(["Id"], axis = 1, inplace=True)
```

```
categories = dict(zip(set(df.Species), range(3)))
categories
```

```
{'Iris-versicolor': 0, 'Iris-setosa': 1, 'Iris-virginica': 2}
```

```
species_num = df.Species.map(lambda x: categories[x])
species_num.head()
```

```
0    1
1    1
2    1
3    1
4    1
Name: Species, dtype: int64
```

Regresión Logística

```
setosa = df[df.Species == "Iris-setosa"]
versicolor = df[df.Species == "Iris-versicolor"]
virginica = df[df.Species == "Iris-virginica"]

setosa
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
5	5.4	3.9	1.7	0.4	Iris-setosa
6	4.6	3.4	1.4	0.3	Iris-setosa
7	5.0	3.4	1.5	0.2	Iris-setosa
8	4.4	2.9	1.4	0.2	Iris-setosa
9	4.9	3.1	1.5	0.1	Iris-setosa
10	5.4	3.7	1.5	0.2	Iris-setosa
11	4.8	3.4	1.6	0.2	Iris-setosa
12	4.8	3.0	1.4	0.1	Iris-setosa
13	4.3	3.0	1.1	0.1	Iris-setosa
14	5.8	4.0	1.2	0.2	Iris-setosa
15	5.7	4.4	1.5	0.4	Iris-setosa
16	5.4	3.9	1.3	0.4	Iris-setosa
17	5.1	3.5	1.4	0.3	Iris-setosa
18	5.7	3.8	1.7	0.3	Iris-setosa
19	5.1	3.8	1.5	0.3	Iris-setosa

20	5.4	3.4	1.7	0.2	Iris-setosa
21	5.1	3.7	1.5	0.4	Iris-setosa
22	4.6	3.6	1.0	0.2	Iris-setosa
23	5.1	3.3	1.7	0.5	Iris-setosa
24	4.8	3.4	1.9	0.2	Iris-setosa
25	5.0	3.0	1.6	0.2	Iris-setosa
26	5.0	3.4	1.6	0.4	Iris-setosa
27	5.2	3.5	1.5	0.2	Iris-setosa
28	5.2	3.4	1.4	0.2	Iris-setosa
29	4.7	3.2	1.6	0.2	Iris-setosa
30	4.8	3.1	1.6	0.2	Iris-setosa
31	5.4	3.4	1.5	0.4	Iris-setosa
32	5.2	4.1	1.5	0.1	Iris-setosa
33	5.5	4.2	1.4	0.2	Iris-setosa
34	4.9	3.1	1.5	0.1	Iris-setosa
35	5.0	3.2	1.2	0.2	Iris-setosa
36	5.5	3.5	1.3	0.2	Iris-setosa
37	4.9	3.1	1.5	0.1	Iris-setosa
38	4.4	3.0	1.3	0.2	Iris-setosa
39	5.1	3.4	1.5	0.2	Iris-setosa
40	5.0	3.5	1.3	0.3	Iris-setosa
41	4.5	2.3	1.3	0.3	Iris-setosa
42	4.4	3.2	1.3	0.2	Iris-setosa
43	5.0	3.5	1.6	0.6	Iris-setosa
44	5.1	3.8	1.9	0.4	Iris-

```
setosa
45              4.8           3.0            1.4           0.3  Iris-
setosa
46              5.1           3.8            1.6           0.2  Iris-
setosa
47              4.6           3.2            1.4           0.2  Iris-
setosa
48              5.3           3.7            1.5           0.2  Iris-
setosa
49              5.0           3.3            1.4           0.2  Iris-
setosa

versicolor

    SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
Species
50              7.0           3.2            4.7           1.4  Iris-
versicolor
51              6.4           3.2            4.5           1.5  Iris-
versicolor
52              6.9           3.1            4.9           1.5  Iris-
versicolor
53              5.5           2.3            4.0           1.3  Iris-
versicolor
54              6.5           2.8            4.6           1.5  Iris-
versicolor
55              5.7           2.8            4.5           1.3  Iris-
versicolor
56              6.3           3.3            4.7           1.6  Iris-
versicolor
57              4.9           2.4            3.3           1.0  Iris-
versicolor
58              6.6           2.9            4.6           1.3  Iris-
versicolor
59              5.2           2.7            3.9           1.4  Iris-
versicolor
60              5.0           2.0            3.5           1.0  Iris-
versicolor
61              5.9           3.0            4.2           1.5  Iris-
versicolor
62              6.0           2.2            4.0           1.0  Iris-
versicolor
63              6.1           2.9            4.7           1.4  Iris-
versicolor
64              5.6           2.9            3.6           1.3  Iris-
versicolor
65              6.7           3.1            4.4           1.4  Iris-
versicolor
66              5.6           3.0            4.5           1.5  Iris-
versicolor
```

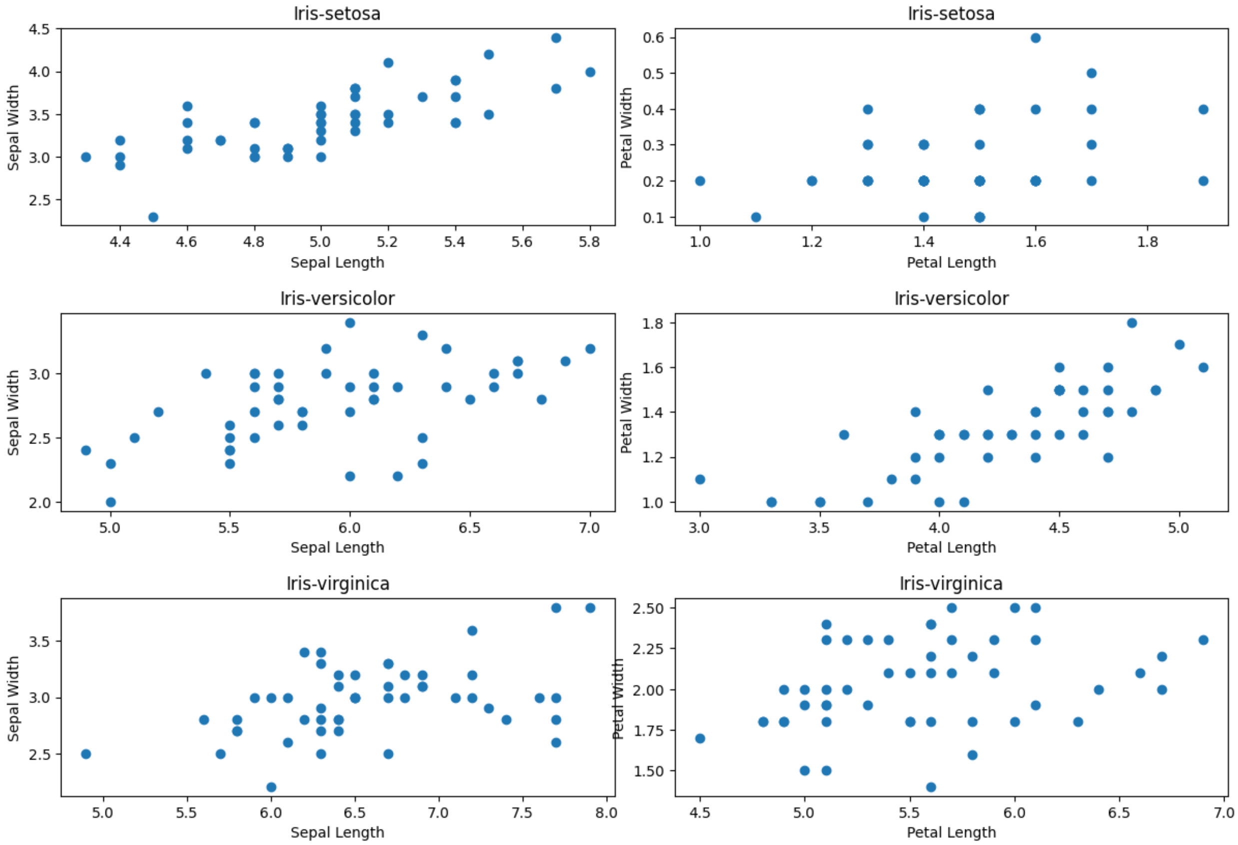
67	5.8	2.7	4.1	1.0	Iris-versicolor
68	6.2	2.2	4.5	1.5	Iris-versicolor
69	5.6	2.5	3.9	1.1	Iris-versicolor
70	5.9	3.2	4.8	1.8	Iris-versicolor
71	6.1	2.8	4.0	1.3	Iris-versicolor
72	6.3	2.5	4.9	1.5	Iris-versicolor
73	6.1	2.8	4.7	1.2	Iris-versicolor
74	6.4	2.9	4.3	1.3	Iris-versicolor
75	6.6	3.0	4.4	1.4	Iris-versicolor
76	6.8	2.8	4.8	1.4	Iris-versicolor
77	6.7	3.0	5.0	1.7	Iris-versicolor
78	6.0	2.9	4.5	1.5	Iris-versicolor
79	5.7	2.6	3.5	1.0	Iris-versicolor
80	5.5	2.4	3.8	1.1	Iris-versicolor
81	5.5	2.4	3.7	1.0	Iris-versicolor
82	5.8	2.7	3.9	1.2	Iris-versicolor
83	6.0	2.7	5.1	1.6	Iris-versicolor
84	5.4	3.0	4.5	1.5	Iris-versicolor
85	6.0	3.4	4.5	1.6	Iris-versicolor
86	6.7	3.1	4.7	1.5	Iris-versicolor
87	6.3	2.3	4.4	1.3	Iris-versicolor
88	5.6	3.0	4.1	1.3	Iris-versicolor
89	5.5	2.5	4.0	1.3	Iris-versicolor
90	5.5	2.6	4.4	1.2	Iris-versicolor
91	6.1	3.0	4.6	1.4	Iris-

```
versicolor
92           5.8           2.6            4.0           1.2  Iris-
versicolor
93           5.0           2.3            3.3           1.0  Iris-
versicolor
94           5.6           2.7            4.2           1.3  Iris-
versicolor
95           5.7           3.0            4.2           1.2  Iris-
versicolor
96           5.7           2.9            4.2           1.3  Iris-
versicolor
97           6.2           2.9            4.3           1.3  Iris-
versicolor
98           5.1           2.5            3.0           1.1  Iris-
versicolor
99           5.7           2.8            4.1           1.3  Iris-
versicolor
```

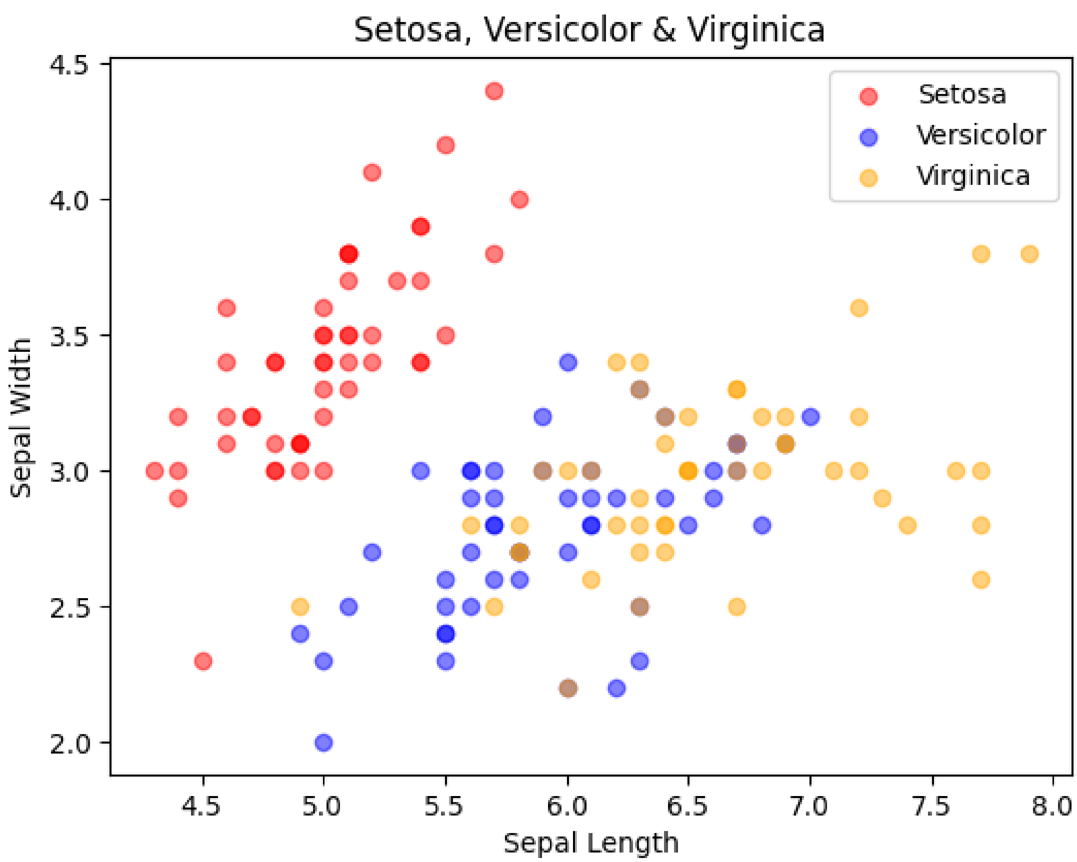
```
virginica.head()
```

```
     SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
Species
100            6.3           3.3            6.0           2.5  Iris-
virginica
101            5.8           2.7            5.1           1.9  Iris-
virginica
102            7.1           3.0            5.9           2.1  Iris-
virginica
103            6.3           2.9            5.6           1.8  Iris-
virginica
104            6.5           3.0            5.8           2.2  Iris-
virginica
```

```
fig, axes = plt.subplots(3, 2, figsize=(12, 8))
fig.tight_layout(h_pad=4)
fig.suptitle("")

for i, iris in enumerate([setosa, versicolor, virginica]):
    axes[i][0].set_title(f"{iris.iloc[0, -1].capitalize()}")
    axes[i][0].scatter(iris.SepalLengthCm, iris.SepalWidthCm)
    axes[i][0].set_ylabel("Sepal Width")
    axes[i][0].set_xlabel("Sepal Length")


    axes[i][1].set_title(f"{iris.iloc[0, -1].capitalize()}")
    axes[i][1].scatter(iris.PetalLengthCm, iris.PetalWidthCm)
    axes[i][1].set_ylabel("Petal Width")
    axes[i][1].set_xlabel("Petal Length")
```



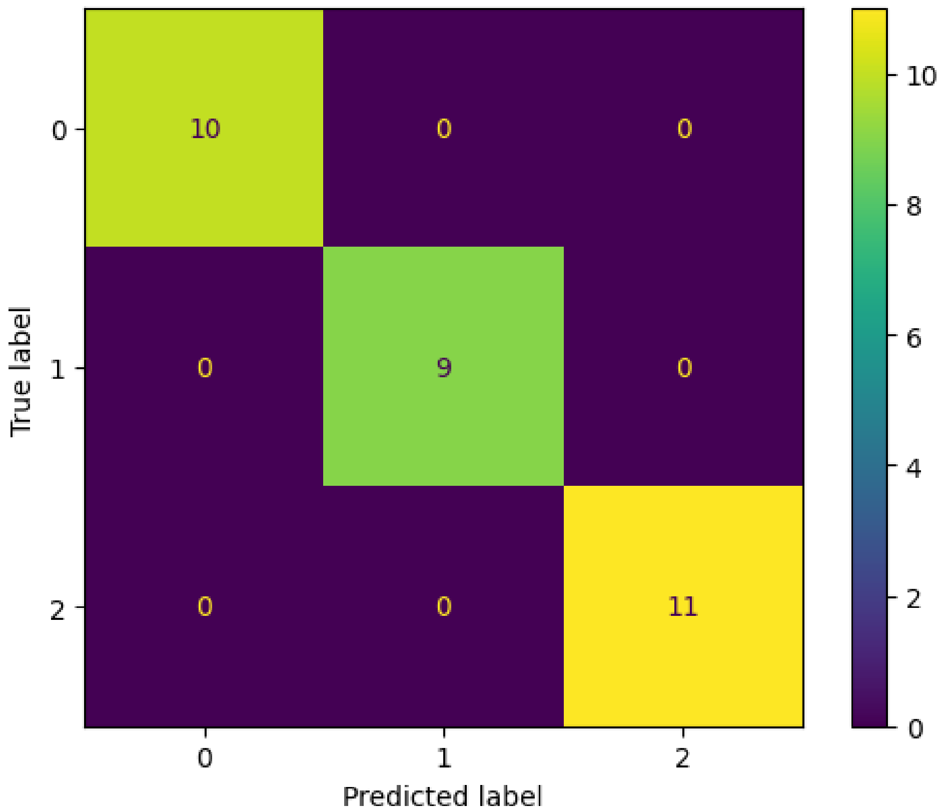

```
plt.title("Setosa, Versicolor & Virginica")
plt.xlabel("Sepal Length")
plt.ylabel("Sepal Width")
plt.scatter(setosa.SepalLengthCm,setosa.SepalWidthCm,
color='red',label = "Setosa", alpha = 0.5)
plt.scatter(versicolor.SepalLengthCm, versicolor.SepalWidthCm, color =
"blue",label = "Versicolor", alpha = 0.5)
plt.scatter(virginica.SepalLengthCm, virginica.SepalWidthCm, color =
"orange",label = "Virginica", alpha = 0.5)
plt.legend()
plt.show()
```




```
X = df.iloc[:, :-1].values
y = df.iloc[:, -1]

X_train, X_test, y_train,
y_test=train_test_split(X,y,test_size=0.2,random_state=42)

logistic_model = LogisticRegression()
logistic_model.fit(X_train, y_train)

y_hat_logistic_train = logistic_model.predict(X_train)
y_hat_logistic_test = logistic_model.predict(X_test)

logistic_train_acc = accuracy_score(y_train, y_hat_logistic_train)
logistic_test_acc = accuracy_score(y_test, y_hat_logistic_test)
print(f"Training accurancy: {logistic_train_acc:.4f}\nTest accurancy:
{logistic_test_acc:.4f}")
```

```
Training accurancy: 0.9750
Test accurancy: 1.0000
```

```
test_conf_matrix = confusion_matrix(y_test, y_hat_logistic_test)
test_cm_disp = ConfusionMatrixDisplay(test_conf_matrix)
test_cm_disp.plot()
plt.show()
```

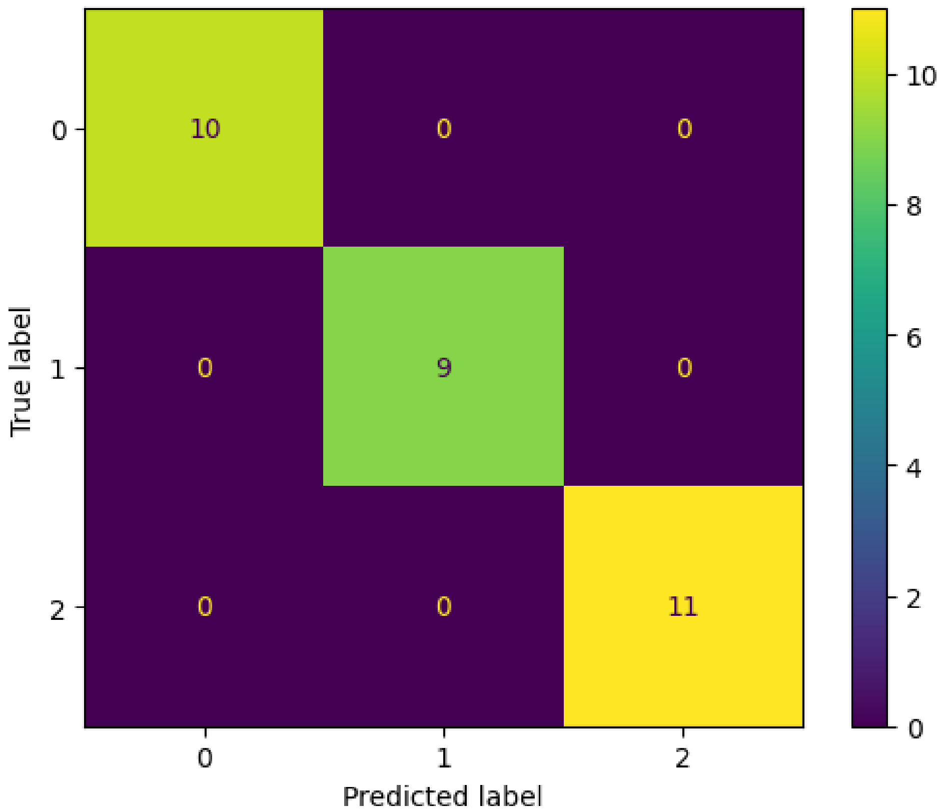


Bayes

```
nb_model = GaussianNB()
nb_model.fit(X_train, y_train)

y_hat_nb_train = nb_model.predict(X_train)
y_hat_nb_test = nb_model.predict(X_test)

nb_train_acc = accuracy_score(y_train, y_hat_nb_train)
nb_test_acc = accuracy_score(y_test, y_hat_nb_test)
print(f"Training accurancy: {nb_train_acc:.4f}\nTest accurancy:
{nb_test_acc:.4f}")
```

```
Training accurancy: 0.9500
Test accurancy: 1.0000
```

```
test_conf_matrix = confusion_matrix(y_test, y_hat_nb_test)
test_cm_disp = ConfusionMatrixDisplay(test_conf_matrix)
test_cm_disp.plot()
plt.show()
```

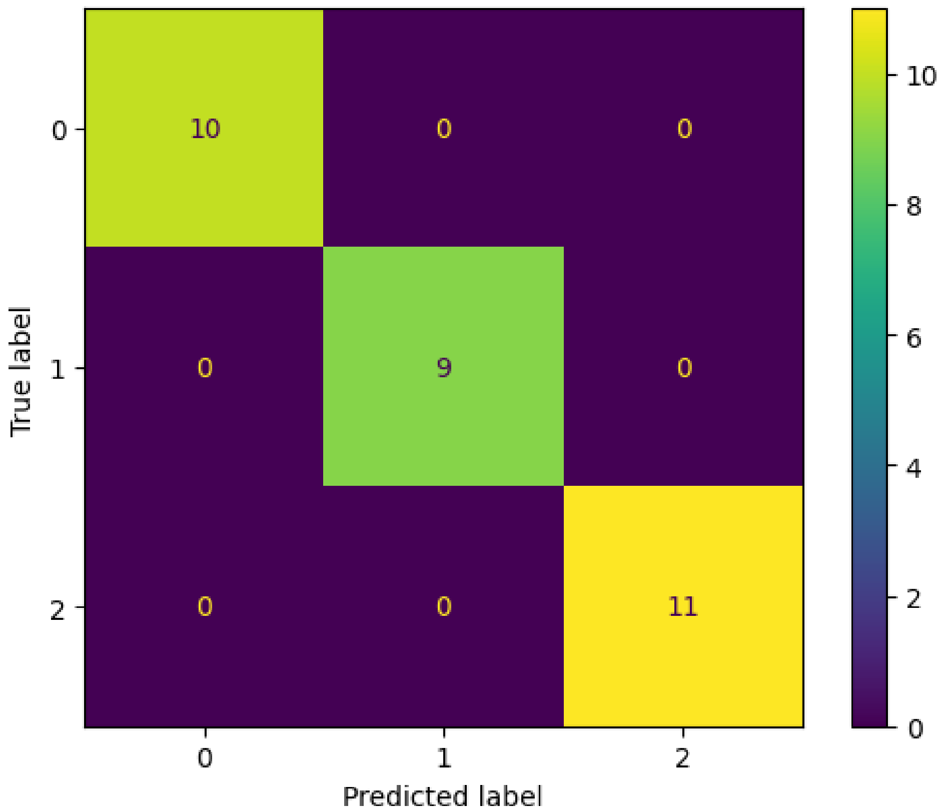


Análisis de discriminate

```
disc_model = LinearDiscriminantAnalysis()
disc_model.fit(X_train, y_train)

y_hat_disc_train = disc_model.predict(X_train)
y_hat_disc_test = disc_model.predict(X_test)

disc_train_acc = accuracy_score(y_train, y_hat_disc_train)
disc_test_acc = accuracy_score(y_test, y_hat_disc_test)
print(f"Training accurancy: {disc_train_acc:.4f}\nTest accurancy:
{disc_test_acc:.4f}")
```

```
Training accurancy: 0.9750
Test accurancy: 1.0000
```

```
test_conf_matrix = confusion_matrix(y_test, y_hat_disc_test)
test_cm_disp = ConfusionMatrixDisplay(test_conf_matrix)
test_cm_disp.plot()
plt.show()
```



¿Qué modelo fue el méjor? ¿Por qué crees que se suceda esto?

Logistica por que al tener los mismos resultados que el discriminante es más facil de interpretar los resultados.

Isai Ambrocio - A01625101