

Data Analysis ARPAID

October 7, 2020

```
[190]: import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

```
[93]: D01 = pd.read_excel("01 DatosFusionados-Cuestionario Previo al uso de la App de
↳ Conjuntos.xlsx", na_values=['-'])
D02 = pd.read_excel("02 DatosFusionados-Cuestionario Posterior al uso de la App
↳ de Conjuntos.xlsx", na_values=['-'])
D03 = pd.read_excel("03 DatosFusionados-Encuesta de Satisfacción sobre el uso
↳ de la App ARPaid.xlsx")
D04 = pd.read_excel("04 Accesos a la App por Alumno.xlsx")
D05 = pd.read_excel("05 Datos Fusionados - Calificaciones Ejercicios Pre y Post
↳ App.xlsx", skiprows=1, na_values=['-'])
```

```
[75]: D01.Nota_20
```

```
[75]: 0      14.25
1      10.00
2       9.33
3       8.83
4      10.08
...
148    16.50
149    10.08
150     9.08
151     1.83
152    12.50
Name: Nota_20, Length: 153, dtype: float64
```

```
[76]: variables_de_interes = ['P01', 'P02', 'P03', 'P04', 'P05', 'P06', 'P07', 'P08',
↳ 'P09', 'P10',
                              'P11', 'P12', 'P13', 'P14', 'P15', 'P16', 'P17', 'P18',
↳ 'P19', 'P20',
                              'Tiempo', 'Nota_20', 'Class']
```

```
[77]: D01 = D01.loc[:, variables_de_interes]
D02 = D02.loc[:, variables_de_interes]
```

```
[78]: D01_pivote = pd.pivot_table(D01, index='Class' , values=['Nota_20'],
    ↪aggfunc=['mean', 'std'] )
D01_pivote.round(1)
```

```
[78]:          mean      std
      Nota_20 Nota_20
Class
0          8.7      3.0
1         12.0      3.8
```

```
[79]: #D01_pivote.drop(columns=['Nota_20']).T.plot(kind='line')
```

```
[80]: D02_pivote = pd.pivot_table(D02, index='Class' , values=['Nota_20'],
    ↪aggfunc=['mean', 'std'])
D02_pivote.round(1)
```

```
[80]:          mean      std
      Nota_20 Nota_20
Class
0         11.9      4.0
1         13.5      3.7
```

```
[81]: #D02_pivote.drop(columns=['Nota_20']).T.plot(kind='line')
```

1 D05

```
[94]: D05.head()
```

```
[94]:  Nombre      Apellido(s)  Número de ID  Dirección de correo \
0  Miguel      Alonso Diaz    78641104X  malond05@estudiantes.unileon.es
1   Mario  Álvarez San Martín  71561891J  malvas12@estudiantes.unileon.es
2  Miguel      Antolin Pajares  71945526P  mantop00@estudiantes.unileon.es
3   Mario      Bresme Díez    71043320T  mbresd00@estudiantes.unileon.es
4  Álvaro      Cuervo Santos  02772971E  acuers00@estudiantes.unileon.es
```

```
      Class Título  Separador  Cáncamo  Portaherramienta  Examen Conjunto
0         1    ea         4.0      8.00             6.5         9.00
1         1    ea         6.0      9.00             8.0         9.00
2         1    ea         NaN      NaN             NaN         6.50
3         1    ea         5.5      5.50             6.0         7.25
4         1    ea         4.0      6.75             8.0         5.00
```

```
[110]: ejercicios = [ 'Cáncamo',          'Portaherramienta',      'Examen Conjunto']
D05.loc[:, ejercicios]
```

```
[110]:
```

| | Cáncamo | Portaherramienta | Examen Conjunto |
|-----|---------|------------------|-----------------|
| 0 | 8.00 | 6.5 | 9.00 |
| 1 | 9.00 | 8.0 | 9.00 |
| 3 | 5.50 | 6.0 | 7.25 |
| 4 | 6.75 | 8.0 | 5.00 |
| 5 | 6.00 | 7.5 | 7.00 |
| .. | ... | ... | ... |
| 140 | 5.00 | 9.0 | 8.50 |
| 141 | 7.00 | 9.5 | 5.00 |
| 143 | 6.00 | 9.0 | 6.50 |
| 145 | 5.00 | 5.0 | 7.00 |
| 151 | 5.00 | 5.0 | 6.50 |

[102 rows x 3 columns]

```
[169]: from sklearn.discriminant_analysis import LinearDiscriminantAnalysis, QuadraticDiscriminantAnalysis
from sklearn.metrics import plot_confusion_matrix, precision_score, recall_score
from sklearn.svm import SVC
import seaborn as sns
```

```
[131]: D05.dropna(inplace=True)
X05 = D05.loc[:, ejercicios]
y05 = D05.loc[:, 'Class']
lda = LinearDiscriminantAnalysis()
qda = QuadraticDiscriminantAnalysis()
svc = SVC()
```

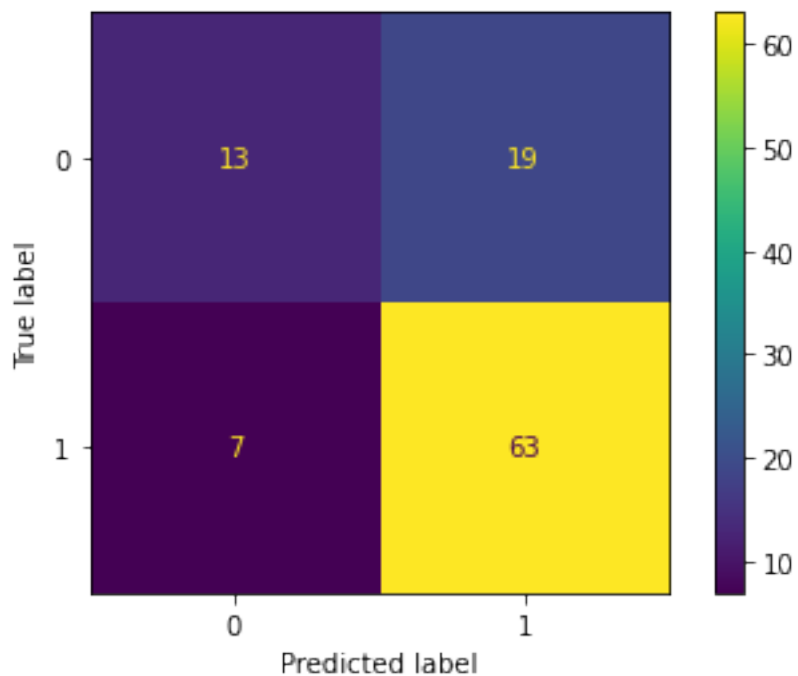
```
[132]: lda.fit(X=X05, y=y05 )
qda.fit(X=X05, y=y05 )
svc.fit(X=X05, y=y05 )
```

```
[132]: SVC()
```

```
[133]: y_pred_lda = lda.predict(X05)
y_pred_qda = qda.predict(X05)
y_pred_svc = svc.predict(X05)
```

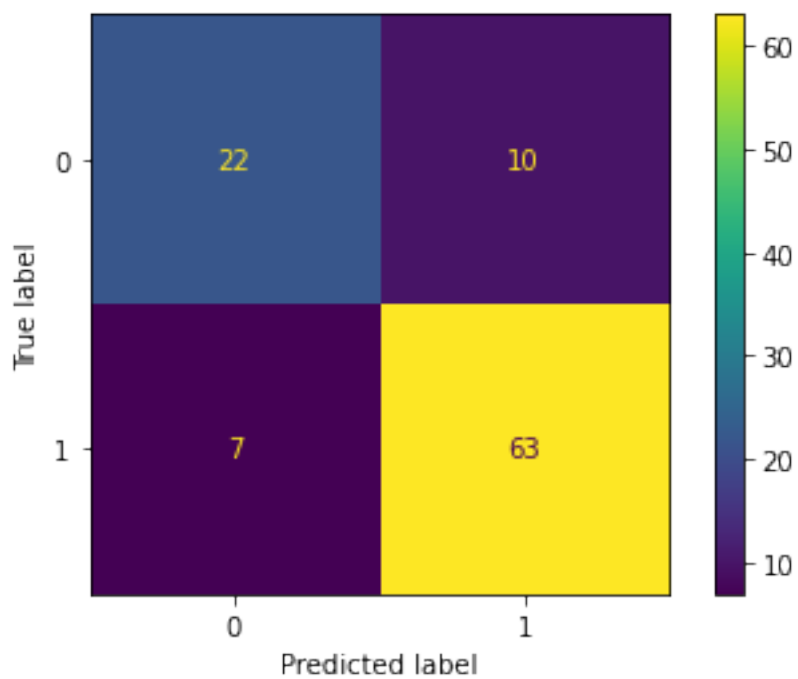
```
[124]: plot_confusion_matrix(lda, X=X05, y_true=y05)
```

```
[124]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7fe41f14ff10>
```



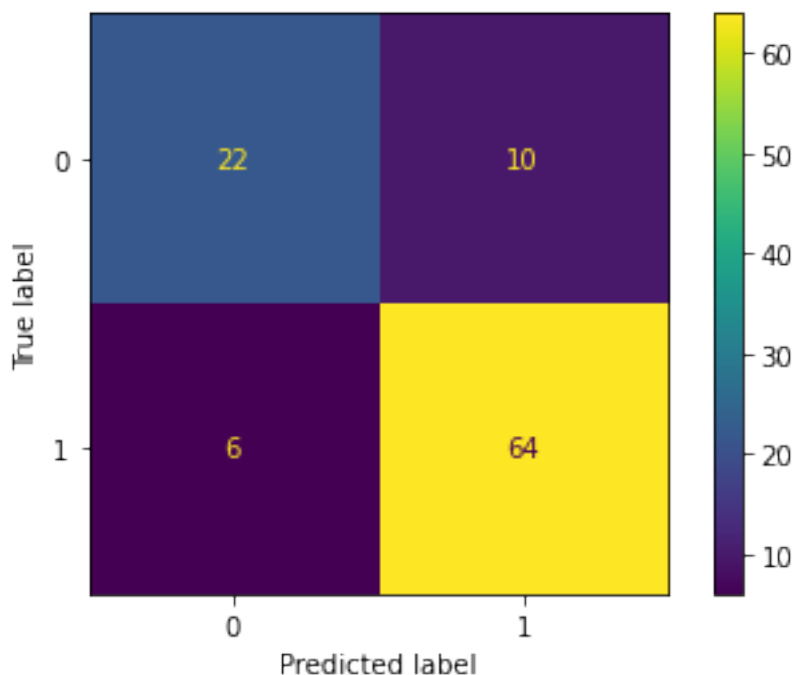
```
[125]: plot_confusion_matrix(qda, X=X05, y_true=y05)
```

```
[125]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7fe41efc5fd0>
```



```
[134]: plot_confusion_matrix(svc, X=X05, y_true=y05)
```

```
[134]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at  
0x7fe41f11ec10>
```



```
[150]: print(f'Precisi3n:{precision_score(y_true=y05, y_pred=y_pred_lda):.2f}, Recall:  
→{recall_score(y_true=y05, y_pred=y_pred_lda):.2f}')
```

Precisi3n:0.77, Recall:0.9:.2f

```
[154]: print(f'Precisi3n:{precision_score(y_true=y05, y_pred=y_pred_qda):.2f}, Recall:  
→{recall_score(y_true=y05, y_pred=y_pred_qda):.2f}')
```

Precisi3n:0.86, Recall:0.90

```
[153]: print(f'Precisi3n:{precision_score(y_true=y05, y_pred=y_pred_svc):.2f}, Recall:  
→{recall_score(y_true=y05, y_pred=y_pred_svc):.2f}')
```

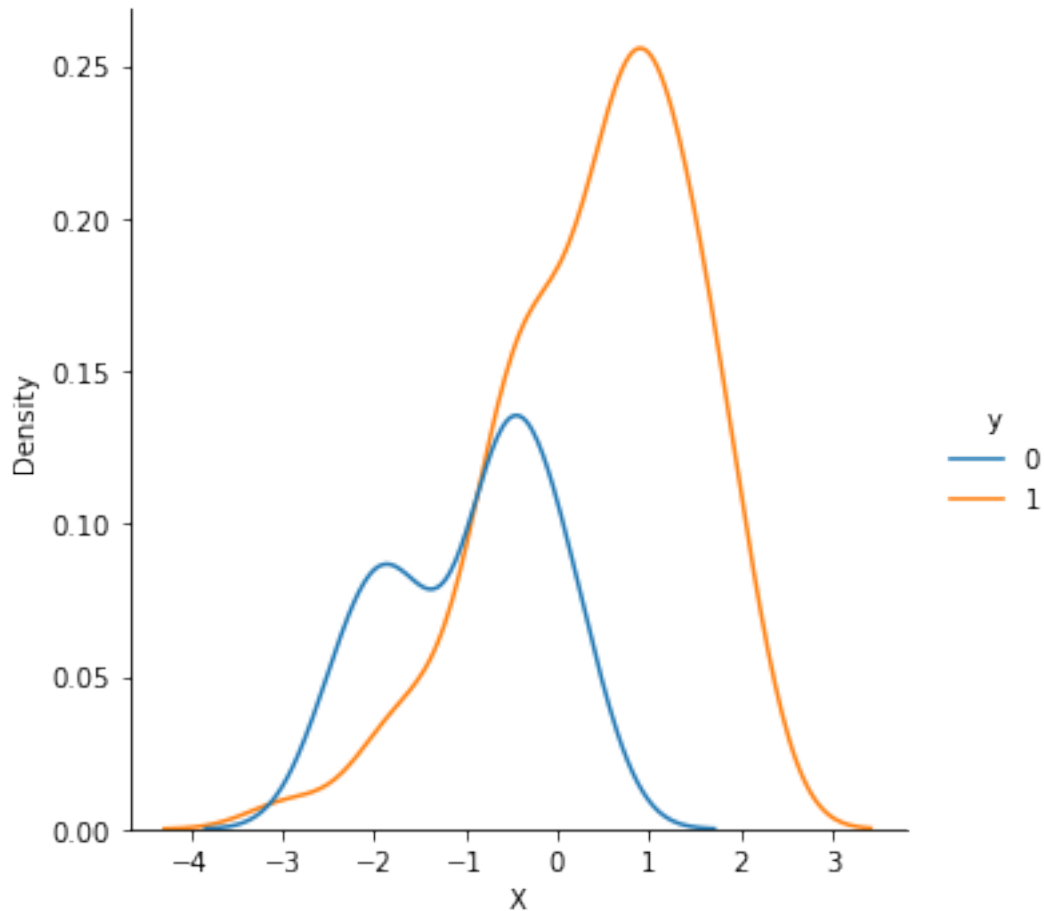
Precisi3n:0.86, Recall:0.91

```
[129]: recall_score(y_true=y05, y_pred=y_pred_qda)
```

```
[129]: 0.9
```

```
[155]: X05_lda = lda.transform(X05)
```

```
[182]: sns.displot(pd.DataFrame({'X':X05_lda.flatten(), 'y':y05.values}), x='X',  
    ↪ hue='y', kind="kde");
```



```
[192]: np.round((abs(lda.coef_)/abs(lda.coef_).sum()), 2)
```

```
[192]: array([[0.38, 0.02, 0.61]])
```

```
[193]: X05.columns
```

```
[193]: Index(['Cáncamo', 'Portaherramienta', 'Examen Conjunto'], dtype='object')
```

```
[ ]:
```