Tercera parte: investigación de modelos supervisados

Random forest classifier

Es un modelo que se utiliza en machine learning para aprendizaje supervisado.

Se usa tanto en problemas de clasificación como en regresión (para predecir valores númericos).

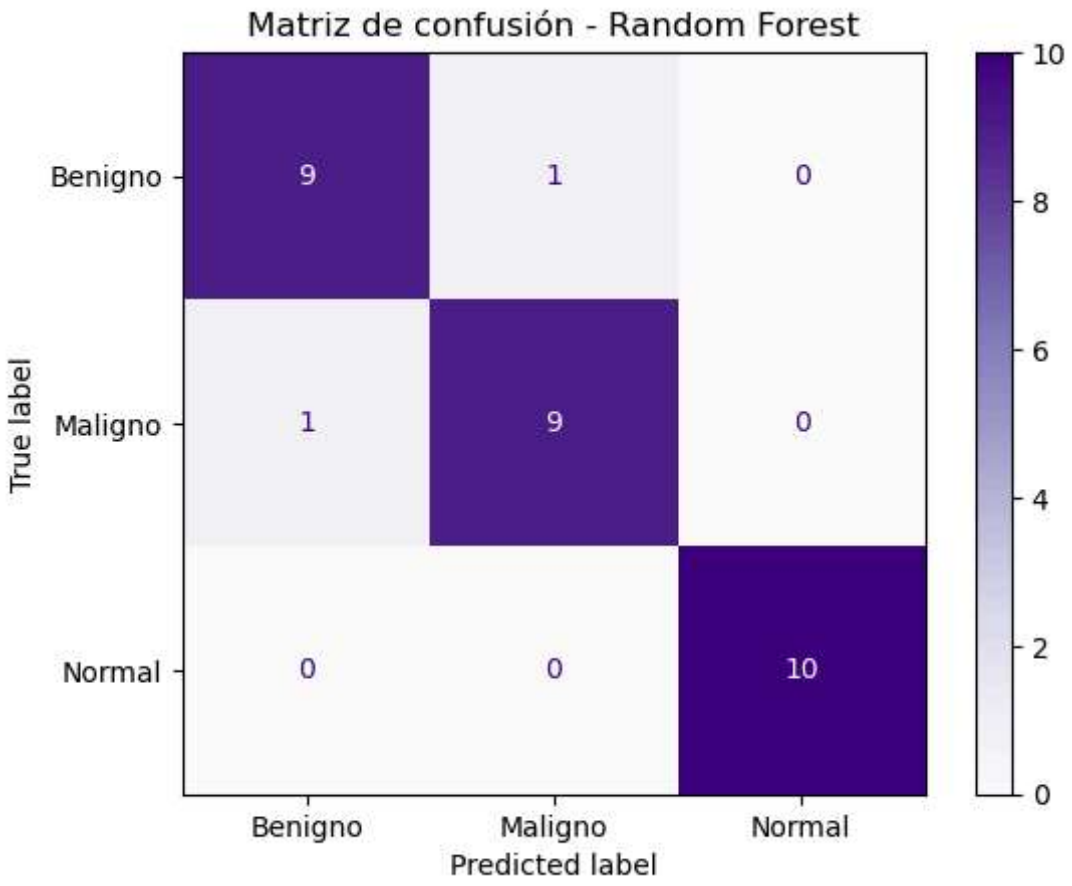
Random Forest crea muchos árboles de decisión, lo que soluciona el sobreajuste de un solo árbol de decisión que tiende a memorizar los datos de entrenamiento y fallar al ver datos nuevos.

Se asegura de que cada árbol sea diferente a los demás. Si todos los árboles fueran iguales, no ganaríamos nada.

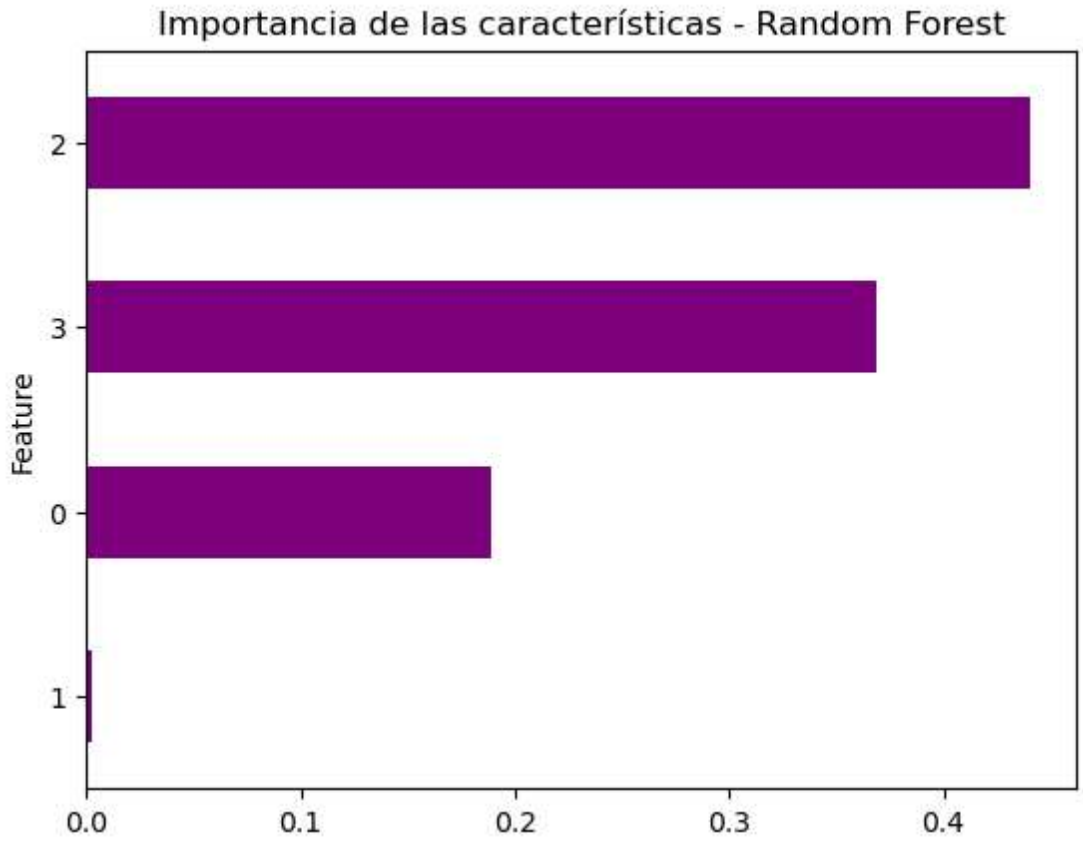
```python
# TERCERA PARTE: INVESTIGACIÓN DE MODELOS SUPERVISADOS

# dividimos el conjunto de datos
X_train_rf, X_test_rf, y_train_rf, y_test_rf = train_test_split(
    X, Y,
    test_size=0.2,
    stratify=Y,
    random_state=SEED,
    shuffle = True
)

# creamos el modelo
model_rf = random_forest_model(350, SEED, 1)

# validación cruzada
skf = StratifiedKFold(
    n_splits=5, # 80/20
    shuffle=True,
    random_state=SEED
)

cv_scores = cross_val_score(model_rf, X_train_rf, y_train_rf, cv=skf, scoring='recall_macro')

print('\n---Resultados de la validación cruzada---')
print(f'Puntuación media: {cv_scores.mean()}')
print(f'Desviación estándar: {cv_scores.std()}')

# entranimiento del modelo
model_rf.fit(X_train_rf, y_train_rf)

# predicción
y_pred_rf = model_rf.predict(X_test_rf)

# evaluación
print('\n---Resultados del modelo (Random Forest)---')
print(f'Accuracy: {accuracy_score(y_test_rf, y_pred_rf)}')
print(classification_report(y_test_rf, y_pred_rf))

# visualización
cm = confusion_matrix(y_test_rf, y_pred_rf, labels=model_rf.classes_)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=model_rf.classes_)
disp.plot(cmap=plt.cm.Purples)
plt.title('Matriz de confusión - Random Forest')
plt.show()

importances = model_rf.feature_importances_
feature_names = df.columns.drop(4)
feature_df = pd.DataFrame({"Feature": feature_names, "Importance": importances})
feature_df = feature_df.sort_values(by="Importance", ascending=True)
feature_df.plot(kind="barh", x="Feature", y="Importance", color='purple', legend=False)
plt.title("Importancia de las caracteristicas - Random Forest")
plt.show()
```

A partir de este modelo, hemos conseguido los siguientes resultados:

```
---Resultados de la validación cruzada---
Puntuación media: 0.95
Desviación estándar: 0.0311804782231162

---Resultados del modelo (Random Forest)---
Accuracy: 0.9333333333333333
              precision    recall  f1-score   support

     Benigno       0.90      0.90      0.90        10
     Maligno       0.90      0.90      0.90        10
      Normal       1.00      1.00      1.00        10

    accuracy                           0.93        30
   macro avg       0.93      0.93      0.93        30
weighted avg       0.93      0.93      0.93        30
```



La siguiente gráfica muestra que variables de las 4 son las que más peso tienen a la hora de hacer una predicción:


Importancia de las características - Random Forest
Feature
2
3
0
1
0.0
0.1
0.2
0.3
0.4