MODELOS Y APRENDIZAJE
ENTREGABLE FINAL 24/02/2024

Autor: Carlos Rosero.

OBJETIVO Y ALCANCE

Supongamos que usted trabaja en el servicio de salud y recibe muestras que provienen de mujeres con cáncer de mama. Los médicos han extraído características y las han anotado, su trabajo es crear un modelo que sea capaz de identificar si un paciente tiene o no cáncer. El dataset y su descripción aparecen aquí:

https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

Recordemos que un falso positivo no es tan preocupante como un falso negativo, ya que en el futuro se les hacen más pruebas a las pacientes y hay oportunidades de descubrir que estábamos en un error. Sin embargo, un falso negativo puede llevar a que el cáncer se desarrolle sin supervisión durante más tiempo del necesario y podría llevar a daños más graves o incluso la muerte de la paciente.

Teniendo esto en cuenta, desarrolla un modelo que funcione lo mejor posible y explica qué decisiones has tomado en su elaboración y por qué.

Entregar un informe que contenga:

- Link a repositorio público de Github con la siguiente información en el readme:
- Un archivo en Jupyter Notebook reseteado con todas las celdas ejecutadas en orden.
- En el notebook debe aparecer el preprocesado de datos desde los archivos originales a ser posible.
- En el notebook debe se tiene que probar al menos con 3 modelos, evaluarlos y decidir cuál es el mejor, justificando la respuesta en base a las matrices de confusión que aparecen al evaluar el error en training y en test.
- Un archivo Readme.md en el que se explica el proyecto y el ejercicio.
- Una carpeta data con el dataset.
- Análisis crítico de los resultados del modelo.
- Interpretación del modelo y características más relevantes.

DESARROLLO

La ruta donde se encuentra el documento en formato Jupyter Notebook (.ipynb) es:

https://github.com/CarlosRoseroC/TareaFinalMyA/tree/main

Descripción del dataset.

En base al link se revisa el sitio y se descarga el dataset; al revisar el dataset se encuentra que no tiene una cabecera con los nombres de los campos y para ello se añade en la primera fila los campos que se indican en la página.

Tenemos varios campos como ID, Diagnosis, radius1, texture1, perimeter1, area1…fractal_dimension3. Un total de 32 campos y 569 registros.

Se hace la carga del dataset, se revisan los campos y sus valores.

- Se elimina el campo “ID” porque no le agrega valor a los modelos.
- En el histograma del campo “Diagnosis” vemos que hay mas registros benignos que malignos (M = maligno, B = benigno).
- No hay valores nulos.
- El campo “Diagnosis” solo tiene valor M o B.
- El resto de los campos tienen valores numéricos de tipo flotante

Para el proceso de aprendizaje se separa el dataset en TRAIN y TEST con los parámetros test_size = 0.2 y random_state = 42

Los modelos que se usan para este ejercicio son:

Random_Forest.

Se elige este modelo por ser robusto y en los ejercicios realizados se ha comportado muy bien. Se configuran los siguientes parámetros en el modelo:

```
modelo_rf = RandomForestClassifier(n_estimators=33, random_state=42,
max_depth = 5, max_features=3)
```

Se movieron varios parámetros y el que dio mejores resultados es n_estimators con valores de 9, 29 y 33; el max_depth y el max_features con valores algo bajos para mantener un equilibrio en la diversidad de los árboles y la cantidad de información.

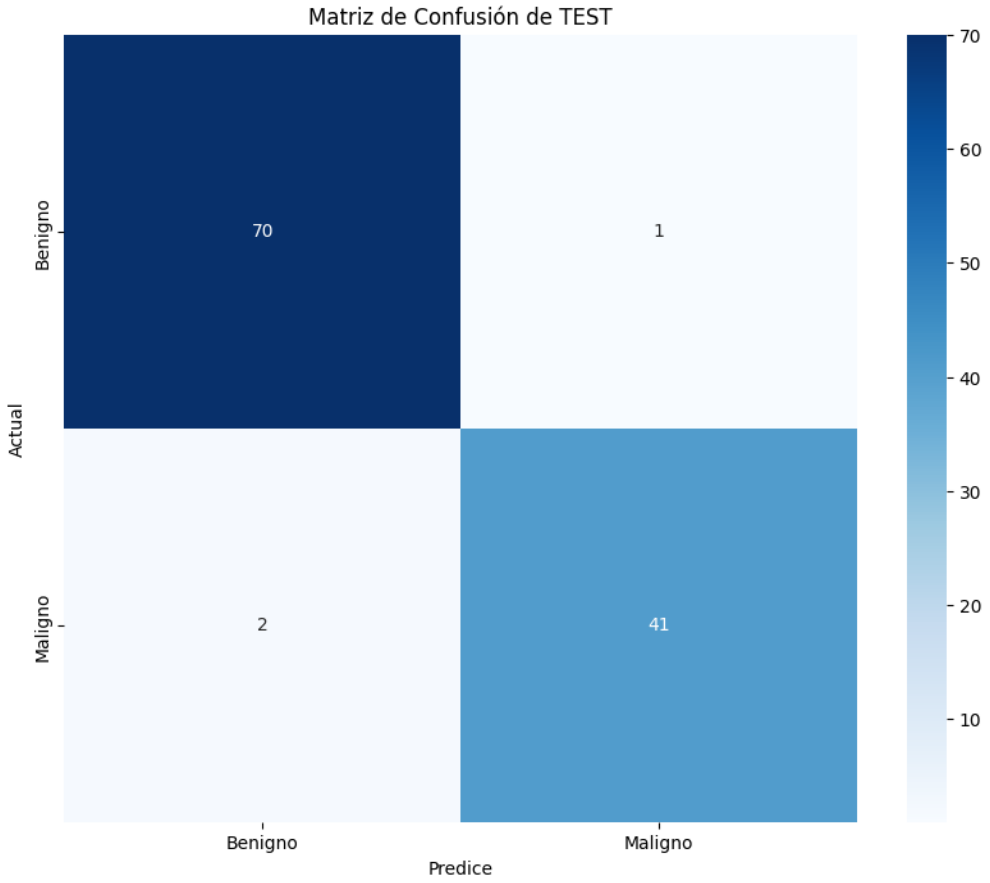
Los indicadores del modelo TEST y su matriz de confusión es:

```
              precision    recall  f1-score   support

           B       0.97      0.99      0.98        71
           M       0.98      0.95      0.96        43

    accuracy                           0.97       114
   macro avg       0.97      0.97      0.97       114
weighted avg       0.97      0.97      0.97       114

Accuracy (exactitud): 0.9736842105263158
[[70  1]
 [ 2 41]]
```



El accuracy(exactitud) está en un 97.368 %; la precisión para detectar los casos M está en un 98%, realmente es un modelo bastante robusto para este dataset. Es importante aclarar que para este modelo los parámetros se ajustaron para que exista la menor cantidad de falsos-negativos

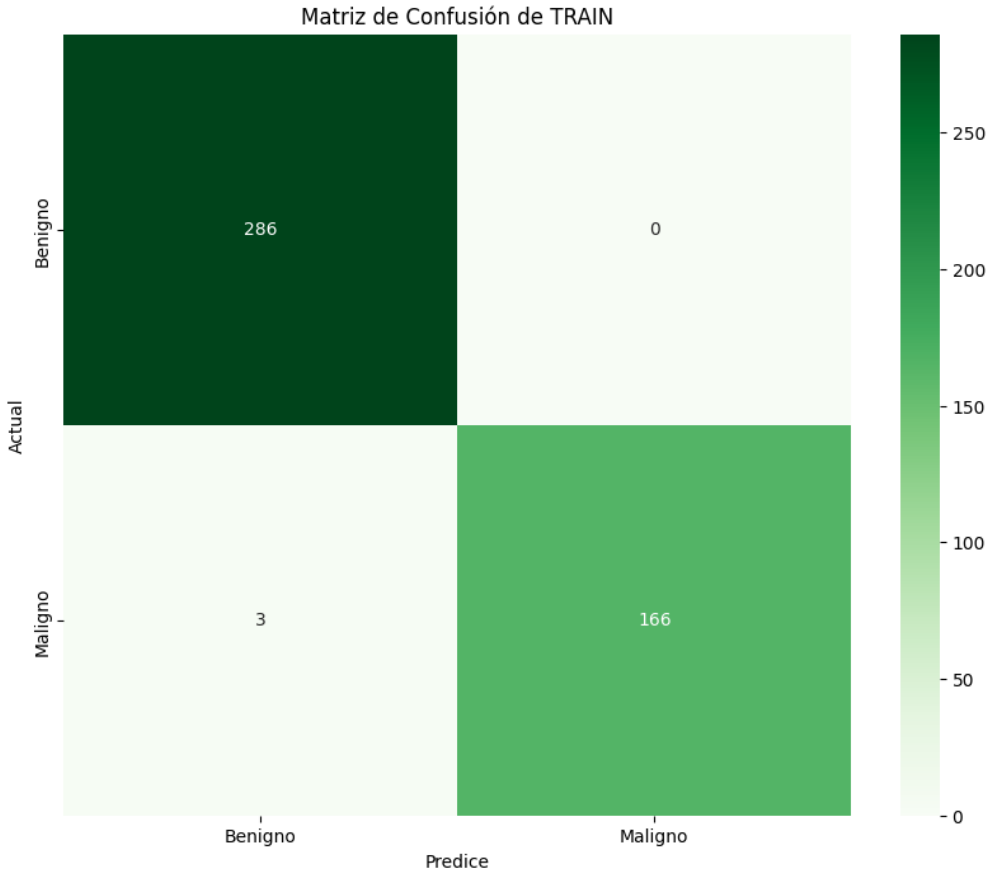
Los indicadores del modelo TRAIN y su matriz de confusión es:

```
              precision    recall  f1-score   support

           B       0.99      1.00      0.99       286
           M       1.00      0.98      0.99       169

    accuracy                           0.99       455
   macro avg       0.99      0.99      0.99       455
weighted avg       0.99      0.99      0.99       455

Accuracy (exactitud): 0.9934065934065934
[[286   0]
 [  3 166]]
```



En la matriz de confusión de TRAIN existen 3 falsos-negativos, y como se comentaba al variar lo parámetros es lo mejor afinamiento que se pudo tener en este modelo..

SVM(Support Vector Machines).

SVM fue utilizado en uno de los ejercicios y dio ciertos resultados aceptables; es importante anotar que este algoritmo ha sido diseñado para funcionar de mejor manera en conjunto de datos algo desbalanceados.

Los parámetros utilizados para este modelo son:

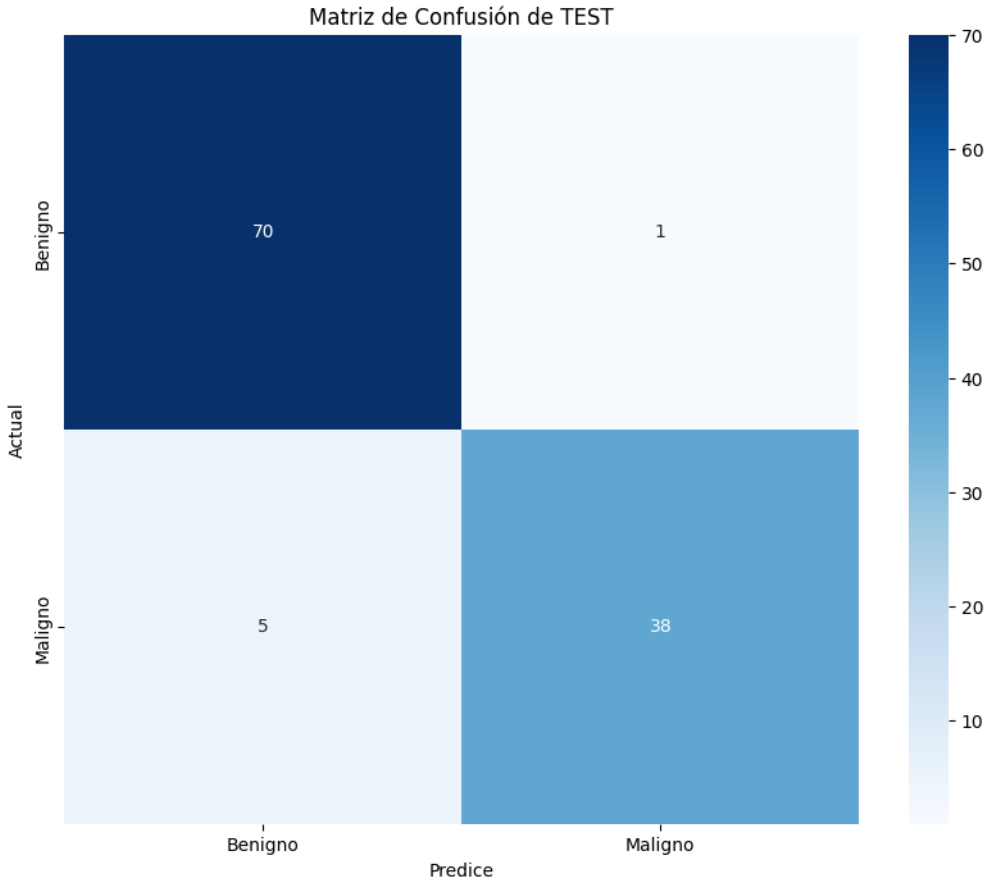
```
modelo_svc = SVC(C=1.0, kernel='poly', gamma = 0.001)
```

El parámetro de regularización C = 1; se usa valor pequeño de gamma(entre 0.001 y 0.1) para que el modelo sea más sensible a los patrones locales en los datos. y esto se complementa con el kernel = ‘poly’; al ejecutar este modelo si le toma un poco más de tiempo que los otros modelos y es por el kernel polinómico que usa el modelo. Otros tipos de kernel daban un accuracy demasiado bajo.

Los indicadores del modelo TEST y su matriz de confusión es:

	precision	recall	f1-score	support
B	0.93	0.99	0.96	71
M	0.97	0.88	0.93	43
accuracy			0.95	114
macro avg	0.95	0.93	0.94	114
weighted avg	0.95	0.95	0.95	114

Accuracy (exactitud): 0.9473684210526315
[[70 1]
[5 38]]



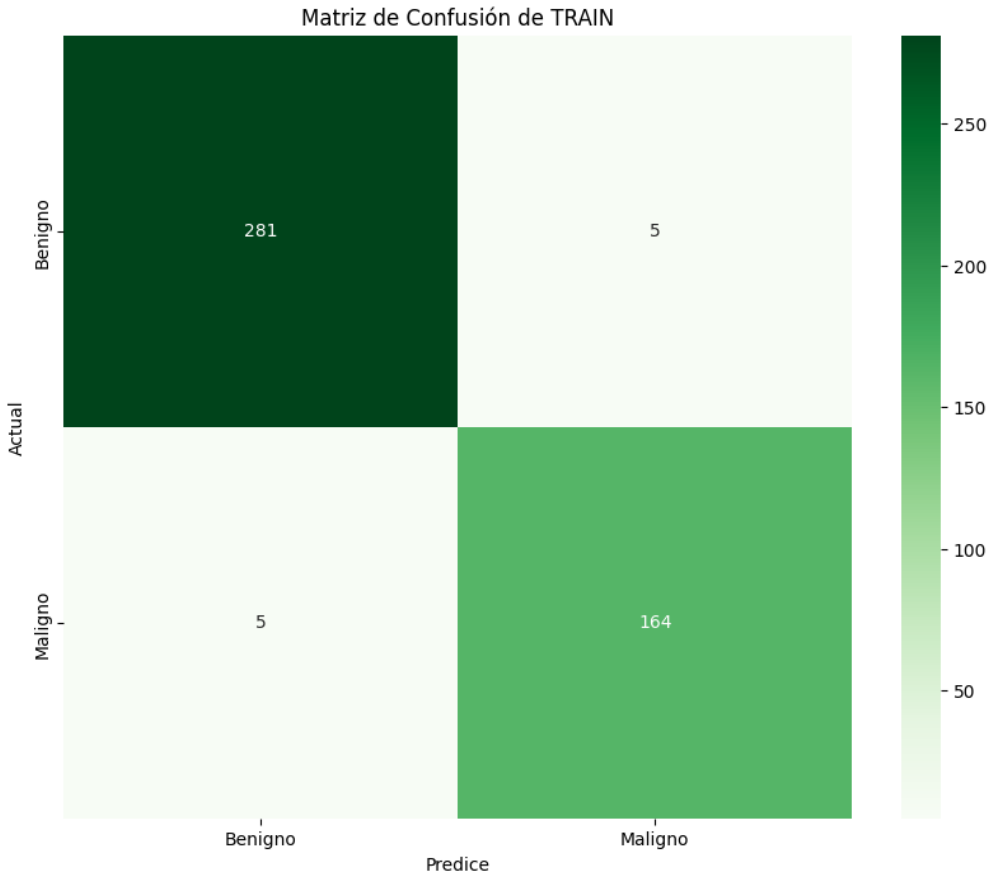
El accuracy(exactitud) está en un 94.736 %; la precisión para detectar los casos M está en un 97%, acá se visualiza que tenemos un falso-positivo y cinco falsos-negativos. Es importante aclarar que para este modelo los parámetros se ajustaron para que exista la menor cantidad de falsos-negativos.

Los indicadores del modelo TRAIN y su matriz de confusión es:

	precision	recall	f1-score	support
B	0.98	0.98	0.98	286
M	0.97	0.97	0.97	169
accuracy			0.98	455
macro avg	0.98	0.98	0.98	455
weighted avg	0.98	0.98	0.98	455

Accuracy (exactitud): 0.978021978021978

```
[[281   5]
 [  5 164]]
```



En la matriz de confusión de TRAIN existen cinco falsos-positivos y cinco falsos-negativos; como se comentaba al variar lo parámetros es el mejor afinamiento que se pudo tener para este modelo.

Regresión Logisitica.

LogisticRegression es un modelo con el que se inicia el aprendizaje de machine-learning y sus conceptos son sencillos de entender y puede ser efectivo si para este modelo los patrones se ajustan a este algoritmo.

Los parámetros utilizados para este modelo son:

```
modelo_rl = LogisticRegression(C=0.01, max_iter=1000)
```

Al especificar C = 0.01 se está aplicando una regularización más fuerte al modelo, esto puede ser útil para evitar el sobreajuste; existe un máximo de iteraciones que llega a 1000 para que logre una convergencia.

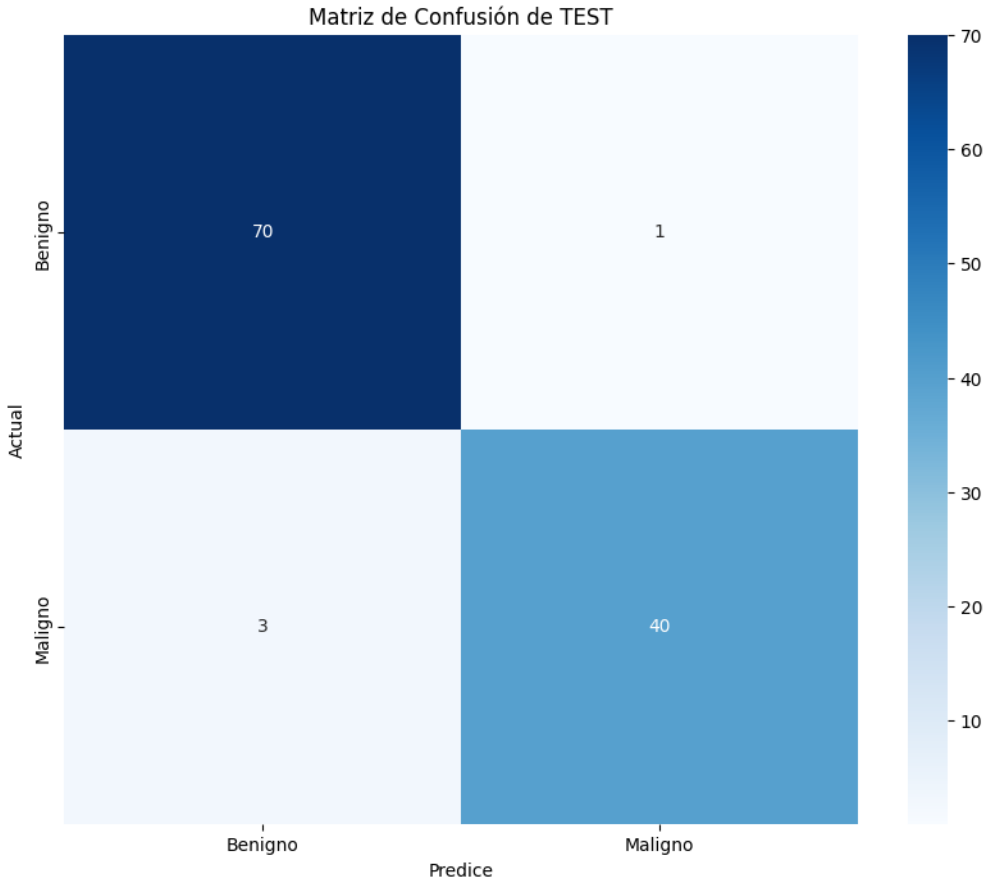
Los indicadores del modelo TEST y su matriz de confusión es:

```
              precision    recall  f1-score   support

           B       0.96      0.99      0.97        71
           M       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

Accuracy (exactitud): 0.9649122807017544
[[70  1]
 [ 3 40]]
```



El accuracy(exactitud) está en un 96.491 %; la precisión para detectar los casos M está en un 98%, acá se visualiza que tenemos un falso-positivo y tres falsos-negativos. Es

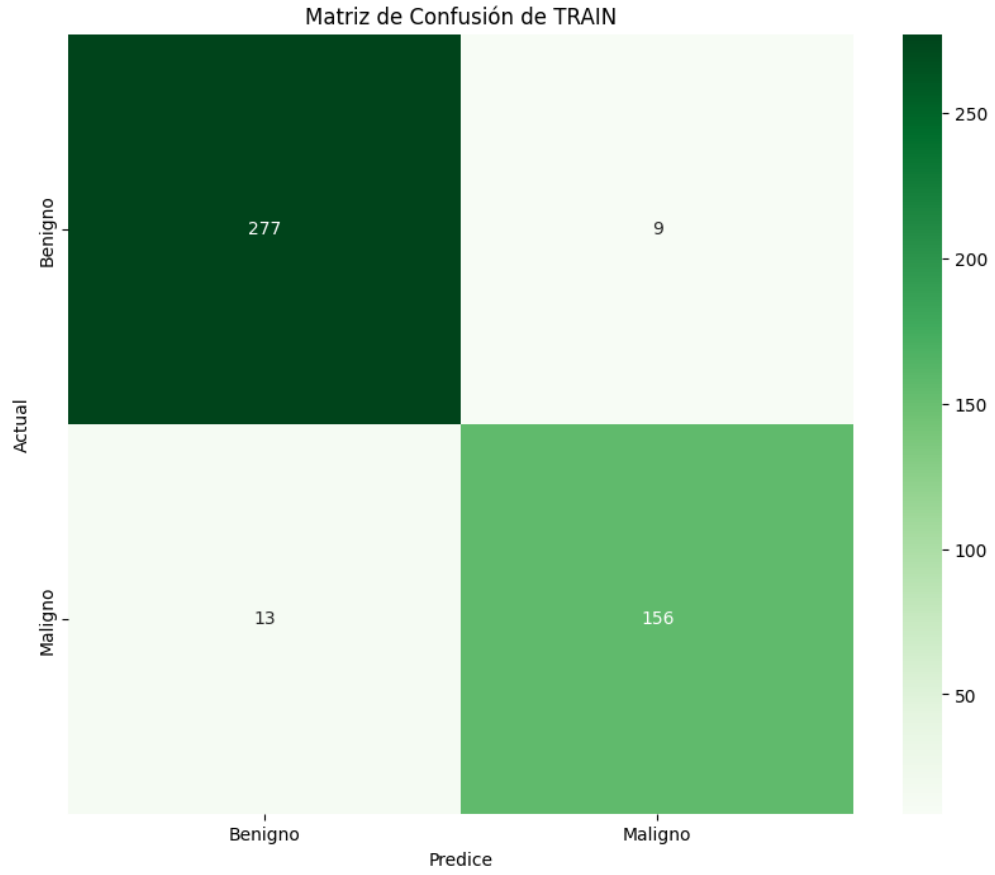
importante aclarar que para este modelo los parámetros se ajustaron para que exista la menor cantidad de falsos-negativos.

Los indicadores del modelo TRAIN y su matriz de confusión es:

	precision	recall	f1-score	support
B	0.96	0.97	0.96	286
M	0.95	0.92	0.93	169
accuracy			0.95	455
macro avg	0.95	0.95	0.95	455
weighted avg	0.95	0.95	0.95	455

Accuracy (exactitud): 0.9516483516483516

```
[[277   9]
 [ 13 156]]
```



En la matriz de confusión de TRAIN existen nueve falsos-positivos y trece falsos-negativos; como se comentaba al variar lo parámetros es el mejor afinamiento que se pudo tener para este modelo.

CONCLUSIÓN

Luego de revisar cada modelo sus métricas y matrices de confusión podemos concluir que el que tiene una mayor accuracy(97.368 %) y menor cantidad de falsos-negativos tanto en TRAIN(3) como TEST(2) es el modelo RandomForest, y este sería el recomendado para este ejercicio de cáncer de mama.