Daniela Jiménez Téllez - A01654798

Clasificación de email de spam: pre-procesamiento y baselines

En este reporte se comparará el desempeño de los modelos de clasificación entrenados en la actividad de clasificación de email. Se usaron 4 clasificadores diferentes: Regresión Logística, Support Vector Machine (SVM), Random Forest, y Gradient Boosting Machine. A continuación se presentan los resultados de cada uno:

Regresión Logística

```
DEBUG:: Los labels completos de regresión logística son:
[0 1 0 0 0 0 1 0 1 1 1 0 0 0 1 0 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 0 0 1 0 0 1
 0 0 1 0 1 1 0 0 1 1 1 1 0 0 0 1 0 0 1 0 1 1 0 1 1 0 1 1 0 1 0 1 0 0 0 1 1
 0 0 1 1 0 1 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1 1 0 1 0 0 0 0 1 1 1 1 0 1 1 1 1
 0 1 0 1 0 1 0 0 0 0 1 0 1 1 1 0 1 0 0 0 1 0 0 0 1 1 1 1 0 0 1 1 0 1 1 1 0
 1 1 0 1 0 1 1 0 1 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0 1 1 1 1 1 1 1 0 0 0
 0 1 0 0 1 1 0 1 0 0 1 1 0 0 0 1 0 0 1 0 0 0 0 0 1 0 1 0 1 0 1 1 1 0 0 1 0
 1 1 1 1 0 1 0 0 1 0 0 0 1 0 1 0 0 1 0 0 1 1 0 1 1 1 0 0 1 0 1 1 1 1 1 0 0
 1 1 1 0 0 1 1 1 0 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 1 1 0 0 0 1 1 1 1 0
 1 0 1 0 1 0 0 1 0 0 1 1 0 1 0 1 1 0 0 0 1 0 0 0 0 0 0 1 0 1 1 0 0 1 1 1 1 1
 1 0 1 1 1 1 1 1 1 0 0 1 0 1 0 0 1 0 0 0 0 1 0 0 1 1 0 1 1 1 0 0 1 0 1 0 0 0
 1 0 1 0 1 1 1 0 1 0 0 1 1 1 1 0 0 0 1 0 1 0 1 1 1 1 1 1 1 0 1 1 1 0 0 0 1 0 1
 0 1 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 1 1 0 1 1 1 0
 1 0 1 1 1 0 0 1 0 0 1 1 1 0 1 1 0 0 0 0 1 0 0 1 1 1 0 1 0 1 0 0 1 0 0 0 1
 1 0 0 1 0 1 1 0 1 0 0 1 1 1 0 0 1 0 0 1 0 0 1 0 0 1 1 1 0 1 0 1 1 1 1 1 1
 0 0 1 1 1 1 0 1 1 0 0 1 0 1 1 1 0 1 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 1 1 1
 0 0 1 0 0 1 0 1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 0 0 1 1 0 1 1 1 1 1 1 0 0 1
 0 0 0 1 1 0 1 0]

   DEBUG::El accuracy score de regresión logística es::
   0.9866666666666667
```

La Regresión Logística es un modelo de clasificación que es utilizado para predecir el resultado de una variable categórica. En este caso, este modelo mostró gran eficiencia en la clasificación de email de spam, teniendo un accuracy de 0.9867, lo que indica que sí sabe diferenciar entre el spam y los correos deseados. A comparación de los otros modelos, en este caso no se tiene el tiempo exacto de ejecución, sin embargo, este modelo es conocido por su rápida ejecución y eficiencia computacional.

Support Vector Machine (SVM)

```
Entrenar el Clasificador SVC tomó  82 segundos
DEBUG::Las labels del Clasificador SVC son::
[1 1 1 0 0 0 1 1 1 1 1 1 0 0 1 1 1 1 0 1 0 0 1 1 0 0 1 1 1 1 0 0 1 1 0 0 1
 0 0 1 0 1 1 1 0 1 1 1 1 0 0 0 1 0 0 1 0 1 1 1 1 1 1 1 1 0 1 0 1 0 0 0 1 1
 1 1 1 1 0 1 1 1 0 1 1 0 0 1 1 0 1 1 1 1 1 1 1 0 1 1 0 0 0 1 1 1 1 0 1 1 1 0
 0 1 0 1 0 1 0 1 1 0 1 0 1 1 1 1 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1 1 0 1 1 1 0
 1 1 0 1 0 1 1 0 1 0 0 0 0 0 1 0 1 1 1 0 1 1 0 0 1 1 1 1 1 1 1 1 0 1 1 1 0 0
 0 1 0 0 1 0 0 1 1 0 1 1 1 1 0 1 0 1 1 1 0 0 1 0 1 0 1 1 1 0 1 1 1 0 0 1 0
 1 1 1 1 1 1 1 1 1 0 0 1 0 1 0 0 1 0 0 1 1 0 1 1 1 0 0 1 0 1 1 1 1 1 0 0
 0 1 1 1 0 1 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 0 1 1 1 0 1 1 1 0 0 0 1 1 1 1 0
 1 0 1 0 1 1 0 1 0 0 1 1 0 1 0 1 1 0 0 0 1 0 0 0 0 0 0 1 0 1 1 0 0 1 1 1 1 1
 1 1 1 1 1 1 1 1 0 0 1 0 1 0 0 1 0 0 0 0 1 0 1 1 1 1 1 1 1 1 0 1 1 1 0 0 0
 1 0 1 0 1 1 1 1 1 0 1 1 1 1 1 0 0 1 0 1 1 1 1 1 1 1 1 0 1 1 1 0 0 0 1 0 1
 1 1 0 0 1 0 1 1 1 1 0 0 0 0 1 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 1 0
 1 0 1 0 1 0 0 1 0 0 1 1 1 0 1 1 0 0 1 1 1 0 0 1 1 1 0 1 0 1 0 0 1 0 0 1 1
 1 0 0 1 0 1 1 1 1 1 1 1 1 0 1 1 1 0 1 0 0 1 0 0 0 1 1 1 1 0 1 1 1 1 1 1
 0 0 1 1 1 1 0 1 1 0 0 1 0 1 1 0 0 1 0 1 0 1 0 1 1 0 1 0 1 1 0 0 1 0 1 1 1
 0 0 1 1 1 1 0 1 0 0 0 1 1 0 1 1 0 0 0 0 1 1 1 0 0 1 1 0 1 1 1 1 0 1 1 0 1
 0 0 1 1 1 0 1 0]
DEBUG::El accuracy score del Clasificador SVC es::
0.8333333333333334
```

Support Vector Machine es un tipo de aprendizaje supervisado que se utiliza para problemas de clasificación. Lo que hace este modelo es que busca un hiperplano que separe datos difíciles de separar. En este caso, se puede observar que el modelo no fue tan eficiente en comparación de los demás. El clasificador tomó 82 segundos en ejecutar y tuvo un accuracy de 0.8334. Esto nos dice que el SVM no pudo capturar las características entre los datos de spam y los emails normales. Es por eso que no fue eficiente en el contexto de este problema.

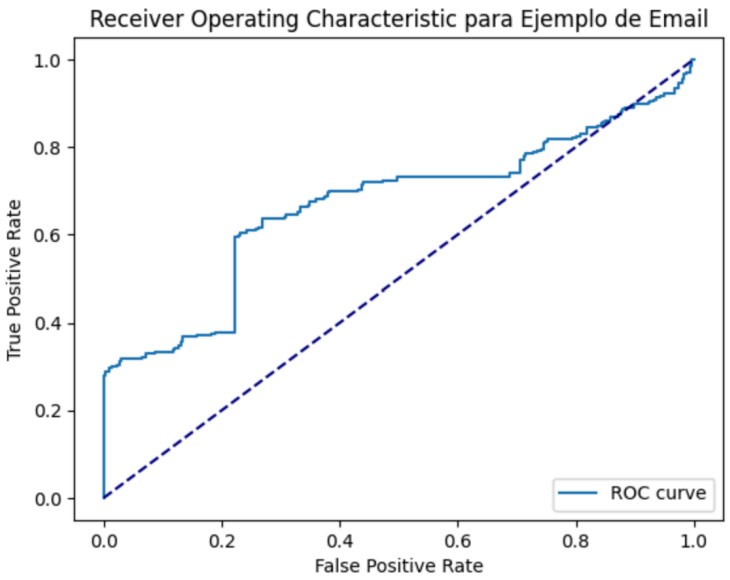
Random Forest

```
Entrenar el Random Forest Classifier tomó   2 segundos
DEBUG::Las etiquetas RF predecidas son::
[0 0 0 0 0 0 1 0 1 1 1 1 0 0 0 1 0 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 0 0 1 0 0 1
 0 0 1 0 1 1 0 0 1 1 1 1 1 0 0 0 1 0 0 1 0 1 1 0 1 1 0 1 1 0 1 0 1 0 0 0 1 1
 0 0 1 1 0 1 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1 1 0 1 0 0 0 0 1 1 1 1 0 1 1 1 1
 0 1 0 1 0 1 0 0 0 0 1 0 1 1 1 0 1 0 0 0 1 0 0 0 1 1 1 1 0 0 1 1 0 1 1 1 1 0
 1 1 0 1 0 1 1 0 1 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0 1 1 0 1 1 1 1 1 0 0 0
 0 1 0 0 1 1 0 1 0 0 1 1 0 0 0 1 0 0 1 0 0 0 0 0 1 0 1 0 1 0 1 1 1 0 0 1 0
 1 1 1 1 0 1 0 0 1 0 0 0 1 0 1 0 0 0 0 0 1 1 0 1 1 1 0 0 1 0 1 1 1 1 1 0 0
 1 1 1 0 0 1 1 1 0 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 1 1 0 0 0 1 1 1 1 0
 1 0 1 0 1 0 0 1 0 0 1 1 0 1 0 1 1 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 1 1 1 1 1
 1 0 1 1 1 1 1 1 0 0 1 0 1 0 0 1 0 0 0 0 1 0 0 1 1 0 1 1 1 0 0 1 0 1 0 0 0
 1 0 1 0 1 1 0 1 0 0 1 1 1 1 0 0 0 1 0 1 0 1 1 1 1 1 1 0 1 1 1 0 0 0 1 0 1
 0 1 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 1 1 0 1 1 1 0
 1 0 1 1 1 0 0 1 0 0 1 1 1 0 1 1 0 0 0 0 1 0 0 1 1 1 0 1 0 1 0 1 1 0 0 0 1
 1 0 0 1 0 1 1 0 1 0 0 1 1 1 0 0 1 0 0 1 0 0 1 0 0 1 1 1 0 1 0 1 1 1 1 1 1
 0 0 1 1 1 1 0 1 1 0 0 1 0 1 1 1 0 1 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 1 1 1
 0 0 1 0 0 1 0 1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0 1 0 0 1 1 0 1 1 1 1 1 1 0 0 1
 0 0 0 1 1 0 1 0]
DEBUG::El RF testing accuracy score es::
0.99
```

Al igual que SVM, Random Forest es un modelo de aprendizaje supervisado el cual crea varios árboles de decisión para encontrar una mejor predicción y clasificación. En el contexto de este problema se puede observar que sirvió muy bien. Este modelo corrió en 2 segundos, lo que nos dice que es muy eficiente computacionalmente hablando. Igualmente, tuvo un accuracy de 0.99, el cual es muy alto. Casi perfecto. Muchas veces, esto puede indicar sobre ajuste, pero generalmente Random Forest reacciona muy bien ante esto.

Gradient Boosting Machine

```
Model Report
Accuracy : 0.9979
AUC Score (Train): 0.998916
CV Score : Mean - 0.9933672 | Std - 0.004898276 | Min - 0.9857136 | Max - 0.9982652
El entrenamiento del Gradient Boosting Classifier tomó 255 segundos
DEBUG::Los labels predecidos de Gradient Boosting son::
[0 0 0 0 0 0 1 0 1 1 1 0 0 0 1 0 1 1 0 1 0 0 1 1 0 0 1 1 0 1 0 0 0 1 0 0 1
 0 0 1 0 1 1 0 0 1 1 1 1 0 0 0 1 0 0 1 0 1 1 0 1 1 0 1 1 0 1 0 1 0 0 0 1 1
 0 0 1 1 0 1 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1 0 1 0 0 0 0 1 1 1 1 0 1 1 1 1
 0 1 0 1 0 1 0 0 0 0 1 0 1 1 1 0 1 0 0 0 1 0 0 0 1 1 1 1 0 0 1 1 0 1 1 1 0
 1 1 0 1 0 1 1 0 1 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0 1 1 0 1 1 1 1 0 0 0
 0 1 0 0 1 1 0 1 0 0 1 1 0 0 0 1 0 0 1 0 0 0 0 0 1 0 1 0 1 0 1 1 1 0 0 1 0
 1 1 1 1 0 1 0 0 1 0 0 0 1 0 1 0 0 1 0 0 1 1 0 1 1 1 0 0 1 0 1 1 1 1 1 0 0
 1 1 1 0 0 1 1 1 0 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 1 1 0 0 0 1 1 1 1 0
 1 0 1 0 1 0 0 1 0 0 1 1 0 1 0 1 1 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 1 1 1 1 1
 1 0 1 1 1 1 1 1 0 0 1 0 1 0 0 1 0 0 0 0 1 0 0 1 1 0 1 1 1 0 0 1 0 1 0 0 0
 1 0 1 0 1 1 0 1 0 0 1 1 1 1 0 0 0 1 0 1 0 1 1 1 1 1 1 0 1 1 1 0 0 0 1 0 1
 0 1 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 1 1 1 1 1 0 1 1 1 0 0 1 1 1 1 0 1 1 1 0
 1 0 1 1 1 0 0 1 0 0 1 1 0 0 1 1 0 0 0 0 1 0 0 1 1 1 0 1 0 1 0 0 1 0 0 0 1
 1 0 0 1 0 1 1 0 0 0 0 1 1 1 0 0 1 0 0 1 0 0 1 0 0 1 1 1 0 1 0 1 1 1 1 1 1
 0 0 1 1 1 1 0 1 1 0 0 1 0 1 1 1 0 1 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 1 1 1
 0 0 1 0 0 1 0 1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 0 0 1 1 0 1 1 1 1 1 1 0 0 1
 0 0 0 1 1 0 1 0]
DEBUG::El testing accuracy score de Gradient Boosting es::
0.9833333333333333
```



Finalmente, el último modelo es Gradient Boosting Machine, el cual se basa en la técnica de boosting. Esta técnica toma diferentes modelos para crear un mejor modelo. En esta actividad tuvo uno de los mejores scores, con 0.98334 de accuracy. Sin embargo, tomó mucho tiempo en ejecutarse, con 255 segundos, se puede decir que este modelo no es computacionalmente eficiente. Esta es una de las desventajas de este, ya que al crear múltiples árboles de decisión, puede tardar más en correr, lo que no lo hace la opción más viable.

Conclusión

Para concluir se puede decir que el mejor modelo para el clasificador de email de spam es Random Forest, ya que no solo mostró un resultado alto en la accuracy (0.99), sino que también tomó 2 segundos en ejecutarse. En el caso de los otros modelos, hay defectos que por muy pequeños que sean, no los vuelven el mejor modelo. Para Regresión Logística, realmente no hubo mucho problema, sino que Random Forest fue mejor. Por otro lado, en SVM, el resultado no fue bueno ni en el accuracy, ni en el tiempo de ejecución. Finalmente, en GBM, se tuvo un muy buen accuracy, pero el costo computacional es muy alto debido al tiempo que tarda en ejecutarse. Por lo tanto, Random Forest es una solución eficiente y precisa en el contexto de un clasificador de email de spam.