**Clasificadores Candidatos Analizados**

1. **Random Forest**
   * Random Forest utiliza múltiples árboles de decisión y combina sus predicciones para mejorar la generalización y reducir el sobreajuste. Es robusto frente a ruido, trabaja bien con variables categóricas y numéricas, y no requiere normalización o codificación extensiva.
   * Referencia académica: Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer. ISBN: 9780387848570.
2. **Redes Neuronales Artificiales (ANN)**
   * ANN es ideal para datos complejos con relaciones no lineales. Sin embargo, requiere preprocesamiento intensivo, como normalización mediante escalado Min-Max, y grandes volúmenes de datos para evitar problemas de sobreajuste. Dado que el dataset incluye clases desbalanceadas, podría ser más desafiante entrenar una red neuronal de manera efectiva.
   * Referencia académica: Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. ISBN: 9780262035613.
3. **Máquinas de Soporte Vectorial (SVM)**
   * Las SVM son eficaces en datasets con un número moderado de características, pero presentan limitaciones al escalar a grandes volúmenes de datos y múltiples clases. Además, requieren normalización previa, lo que agrega complejidad al preprocesamiento.
   * Referencia académica: Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297. DOI: 10.1007/BF00994018.
4. **K-Nearest Neighbors (KNN)**
   * Aunque es simple de implementar, KNN es sensible a las escalas de las características y a los datos desbalanceados. Además, su complejidad aumenta significativamente con el tamaño del dataset, lo que lo hace menos eficiente para grandes volúmenes de datos.
   * Referencia académica: Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27. DOI: 10.1109/TIT.1967.1053964.
5. **Naive Bayes**
   * Aunque Naive Bayes es eficaz para datos categóricos y texto, la fuerte suposición de independencia entre características limita su aplicabilidad en datos con correlaciones. Para datasets complejos como el actual, no es la opción más adecuada.
   * Referencia académica: Zhang, H. (2004). The optimality of Naive Bayes. *AAAI Conference on Artificial Intelligence*. DOI: 10.1609/aaai.v17i1.12356.

**Justificación de la Selección: Random Forest**

**Características de Random Forest**

1. **Capacidad para Manejar Datos Mixtos:**
   * Random Forest trabaja de manera eficiente con datos categóricos y numéricos sin requerir preprocesamientos intensivos como normalización o codificación. Esto lo hace particularmente útil para datasets heterogéneos como el actual.
   * Referencia: Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32. DOI: 10.1023/A:1010933404324.
2. **Robustez Frente a Datos Desbalanceados:**
   * Con el parámetro class\_weight='balanced', Random Forest ajusta el peso de cada clase automáticamente, lo que permite manejar clases desbalanceadas sin necesidad de sobremuestreo o submuestreo extensivo.
   * Ventaja: Esto ayuda a mejorar métricas como el recall y F1-score en clases minoritarias.
3. **Escalabilidad:**
   * Random Forest es capaz de manejar grandes volúmenes de datos debido a su enfoque basado en árboles paralelizados, donde cada árbol se construye de manera independiente. Esto garantiza un tiempo de entrenamiento eficiente incluso en datasets grandes.
4. **Interpretabilidad:**
   * Random Forest permite analizar la importancia de las características (feature\_importances\_), proporcionando información valiosa sobre qué factores contribuyen más a la clasificación. Esto es crucial para interpretar los resultados y justificar el modelo ante un público no técnico.

**Ventajas Comparativas**

1. **Versatilidad y Robustez:**
   * A diferencia de modelos como SVM o ANN, Random Forest no requiere supuestos específicos sobre las distribuciones de los datos ni relaciones lineales o no lineales. Es un modelo "todo terreno" que funciona bien en la mayoría de los escenarios.
   * Referencia: James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning: With Applications in R*. Springer. ISBN: 9781461471370.
2. **Menor Sensibilidad a Ruido y Outliers:**
   * Los algoritmos basados en vecinos cercanos (KNN) o las ANN son más susceptibles al ruido en los datos. Random Forest, gracias a su método de votación mayoritaria entre árboles, es menos sensible a valores atípicos.
3. **Eficiencia en Multiclase:**
   * Random Forest maneja múltiples clases de manera nativa, a diferencia de SVM, que necesita estrategias adicionales como "one-vs-rest" o "one-vs-one".