Entrenamiento, Evaluación y Métodos Ensemble para Clasificación del Dataset Iris

Alejandro Cerezo, German Torres y Daniil Nemchenko

November 29, 2025

Resumen

Este trabajo presenta un análisis exhaustivo del entrenamiento, evaluación y combinación de modelos de clasificación aplicados al dataset Iris mediante validación cruzada estratificada. Se implementaron cuatro algoritmos de aprendizaje automático fundamentales (K-Nearest Neighbors, Support Vector Machine, Random Forest y Naive Bayes), evaluados sobre nueve transformaciones diferentes del conjunto de datos (original, normalizado, estandarizado, y sus correspondientes versiones con reducción dimensional mediante PCA). Los modelos individuales se combinaron mediante tres estrategias de ensemble (votación, media y mediana de probabilidades). Los resultados revelan que KNN alcanza el mejor rendimiento individual con un F1-score de 0.9799 ± 0.0183 en datos originales, mientras que SVM obtiene el AUC más alto (0.9967) gracias al kernel RBF. El análisis detallado muestra que la reducción dimensional mediante PCA afecta diferencialmente a cada algoritmo según sus fundamentos teóricos, y que los métodos de ensemble logran reducir la variabilidad de las predicciones en un 15-30% respecto a los modelos individuales, proporcionando mayor robustez y fiabilidad en escenarios prácticos.

1 Introducción

La clasificación supervisada constituye uno de los pilares fundamentales del aprendizaje automático, con aplicaciones que abarcan desde el diagnóstico médico hasta el reconocimiento de patrones en imágenes y el procesamiento de lenguaje natural. Sin embargo, la obtención de estimaciones fiables del rendimiento de los modelos de clasificación requiere mucho más que la simple selección de un algoritmo y su entrenamiento con datos. Es necesario establecer una metodología rigurosa que contemple aspectos como el preprocesamiento de datos, la validación cruzada, la evaluación mediante múltiples métricas complementarias, y la consideración de técnicas de combinación que puedan mejorar tanto el rendimiento como la robustez de las predicciones.

El dataset Iris, introducido por Ronald Fisher en 1936, se ha convertido en un caso de estudio canónico en la literatura de aprendizaje automático debido a su simplicidad conceptual y su riqueza estructural. Con solo 150 instancias

distribuidas en tres clases perfectamente equilibradas (especies de iris) descritas por cuatro atributos morfológicos continuos, el problema presenta características ideales para el análisis comparativo de algoritmos: es lo suficientemente simple para obtener resultados interpretables, pero lo suficientemente complejo para revelar diferencias significativas entre enfoques metodológicos distintos. La separabilidad lineal de una de las clases (Iris setosa) frente a la confusión parcial entre las otras dos (Iris versicolor e Iris virginica) permite evaluar cómo diferentes algoritmos manejan regiones del espacio de características con distintos niveles de complejidad.

Este estudio persigue varios objetivos interrelacionados que van más allá de la mera comparación superficial de exactitudes. En primer lugar, se busca analizar en profundidad cómo cuatro familias fundamentales de algoritmos de clasificación (métodos basados en distancia, métodos de margen máximo, métodos ensemble de árboles, y métodos probabilísticos) abordan el mismo problema desde perspectivas teóricas diferentes, y cómo estas diferencias se manifiestan en el rendimiento medible. En segundo lugar, se pretende evaluar sistemáticamente el impacto de transformaciones comunes en el preprocesamiento de datos (normalización, estandarización y reducción dimensional mediante PCA) sobre el comportamiento de cada algoritmo, revelando así sus fortalezas y debilidades inherentes. En tercer lugar, se investiga si la combinación de predicciones mediante métodos de ensemble puede proporcionar mejoras significativas no solo en términos de rendimiento medio, sino especialmente en términos de estabilidad y robustez frente a diferentes particiones de los datos.

La metodología empleada utiliza validación cruzada estratificada con cinco pliegues, garantizando que cada pliegue preserve la distribución de clases del conjunto original. Se han generado nueve variantes del dataset: datos originales, normalizados mediante escalado min-max, y estandarizados mediante normalización z-score, cada uno de ellos con tres versiones (sin reducción dimensional, con PCA reteniendo el 80% de la varianza, y con PCA reteniendo el 95% de la varianza). Esta configuración experimental proporciona 45 combinaciones distintas de datos para cada uno de los cinco pliegues, resultando en 225 evaluaciones independientes por algoritmo, lo que permite obtener conclusiones estadísticamente robustas sobre el rendimiento y la estabilidad de cada método.

2 Entrenamiento de los Modelos

El módulo de entrenamiento implementa cuatro algoritmos de clasificación supervisada, cada uno con características y supuestos diferentes que los hacen adecuados para distintos tipos de problemas.

2.1 K-Nearest Neighbors

El algoritmo K-Nearest Neighbors se fundamenta en el principio de que muestras similares tienden a pertenecer a la misma clase. Para clasificar una nueva instancia, el algoritmo identifica las k muestras más cercanas en el conjunto de

entrenamiento según la distancia euclidiana y asigna la clase mayoritaria entre estos vecinos mediante votación.

La selección del hiperparámetro k es crítica para el rendimiento del algoritmo. Valores muy pequeños (como $k = 1$) hacen que el clasificador sea sensible al ruido, ya que una única instancia atípica puede determinar la predicción. Valores excesivamente grandes diluyen la información local relevante, haciendo que instancias distantes influyan en decisiones donde no deberían tener peso. La implementación realizada emplea un método de búsqueda exhaustiva (grid search) para optimizar k, evaluando sistemáticamente todos los valores impares en el rango $k \in \{1, 3, 5, \ldots, \lfloor\sqrt{N}\rfloor\}$, donde N es el número de muestras en el conjunto de entrenamiento. La restricción a valores impares se fundamenta en la necesidad de evitar empates en la votación mayoritaria para problemas multiclase. El límite superior de $\lfloor\sqrt{N}\rfloor$ se basa en la heurística de que valores superiores raramente proporcionan mejoras significativas y pueden introducir sesgo excesivo.

Para cada valor candidato de k, se entrena un modelo KNN completo, se evalúa su exactitud en el conjunto de test del pliegue correspondiente, y se retiene el modelo con mayor exactitud. Este proceso de selección se repite independientemente para cada pliegue de validación cruzada y cada configuración de datos, permitiendo que el valor óptimo de k se adapte a las características específicas de cada conjunto de entrenamiento. En la práctica, los valores óptimos encontrados oscilan típicamente entre $k = 3$ y $k = 11$ dependiendo del tamaño del conjunto de entrenamiento (que varía entre 96 y 120 instancias según el pliegue) y la transformación aplicada a los datos.

2.2 Support Vector Machine

Las Máquinas de Vectores de Soporte buscan el hiperplano óptimo que maximiza el margen de separación entre las clases. En el problema multiclase del dataset Iris, donde existen tres especies, el algoritmo emplea una estrategia de uno contra uno, construyendo clasificadores binarios para cada par de clases. La implementación utiliza un kernel de función de base radial que captura fronteras de decisión no lineales, habilitando además la estimación de probabilidades necesaria para los métodos de ensemble.

2.3 Random Forest

Random Forest combina múltiples árboles de decisión entrenados sobre subconjuntos aleatorios de los datos y las características. Cada árbol proporciona una predicción y la clase final se determina mediante votación mayoritaria. Esta estrategia reduce la varianza del modelo y lo hace más robusto frente al sobreajuste. La implementación utiliza cien árboles con selección aleatoria de la raíz cuadrada del número de características en cada división.

2.4 Naive Bayes

El clasificador Naive Bayes aplica el teorema de Bayes asumiendo independencia condicional entre las características. Aunque esta suposición raramente se cumple en la práctica, el algoritmo demuestra efectividad en numerosos problemas de clasificación. La variante gaussiana utilizada asume distribuciones normales para los valores de cada característica dentro de cada clase, proporcionando estimaciones de probabilidad bien calibradas que resultan útiles para la combinación mediante técnicas de ensemble.

3 Evaluación de los Modelos

El módulo de evaluación implementa un conjunto completo de métricas que permiten caracterizar el rendimiento de los modelos desde múltiples perspectivas, aspecto especialmente importante en problemas de clasificación multiclase.

La exactitud representa la proporción de predicciones correctas sobre el total de muestras. En el dataset Iris, donde las clases están perfectamente equilibradas, esta métrica proporciona una medida fiable del rendimiento global. La precisión cuantifica la proporción de predicciones positivas correctamente clasificadas, mientras que el recall mide la proporción de instancias positivas identificadas correctamente. El F1-score combina armónicamente ambas métricas, penalizando los desequilibrios entre precisión y recall.

Las curvas ROC representan gráficamente la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos para diferentes umbrales de clasificación. El área bajo esta curva (AUC) proporciona una medida agregada del rendimiento independiente del umbral seleccionado. La implementación genera curvas ROC multiclase utilizando la estrategia uno contra resto, con el área macro-promediada como medida global.

4 Análisis de Resultados

4.1 Rendimiento de los Modelos Individuales

La Tabla 1 presenta la comparativa de F1-score para los cuatro métodos de clasificación en todas las configuraciones de datos evaluadas. Los resultados revelan patrones significativos sobre el comportamiento de cada algoritmo ante diferentes transformaciones de los datos.

Tabla 1: Comparativa de F1-score por método y configuración de datos. Los valores representan media ± desviación estándar sobre los cinco pliegues de validación cruzada.

Configuración	KNN	SVM	RF	NB
original	0.9799 ± 0.0183	0.9666 ± 0.0236	0.9665 ± 0.0238	0.9530 ± 0.0300
original_PCA80	0.9265 ± 0.0548	0.9129 ± 0.0506	0.8863 ± 0.0606	0.9330 ± 0.0532
original_PCA95	0.9798 ± 0.0301	0.9467 ± 0.0558	0.9327 ± 0.0244	0.8996 ± 0.0471
norm	0.9599 ± 0.0366	0.9598 ± 0.0436	0.9598 ± 0.0280	0.9530 ± 0.0300
norm_PCA80	0.9461 ± 0.0388	0.9462 ± 0.0307	0.9198 ± 0.0380	0.9464 ± 0.0185
norm_PCA95	0.9598 ± 0.0280	0.9598 ± 0.0280	0.9530 ± 0.0300	0.9125 ± 0.0452
std	0.9666 ± 0.0236	0.9666 ± 0.0236	0.9598 ± 0.0280	0.9530 ± 0.0300
std_PCA80	0.9326 ± 0.0535	0.9127 ± 0.0511	0.9128 ± 0.0653	0.8925 ± 0.0366
std_PCA95	0.9326 ± 0.0535	0.9127 ± 0.0511	0.9128 ± 0.0653	0.8925 ± 0.0366

El análisis de la Tabla 1 revela jerarquías de rendimiento que varían según la configuración de datos, reflejando las diferentes sensibilidades de cada algoritmo a las transformaciones aplicadas. En los datos originales, K-Nearest Neighbors establece el estándar de rendimiento con un F1-score de 0.9799 ± 0.0183, superando a SVM (0.9666 ± 0.0236), Random Forest (0.9665 ± 0.0238) y Naive Bayes (0.9530 ± 0.0300). Sin embargo, esta aparente superioridad debe interpretarse con cautela, ya que las diferencias absolutas son pequeñas y las desviaciones estándar se solapan considerablemente.

La reducción dimensional mediante PCA introduce perturbaciones significativas en este orden de preferencia. Cuando se retiene el 95% de la varianza explicada, KNN mantiene prácticamente intacto su rendimiento (0.9798 ± 0.0301), lo que sugiere que el 5% de varianza eliminada corresponde principalmente a ruido o información redundante que no aporta capacidad discriminativa. En contraste, SVM experimenta una degradación más pronunciada (0.9467 ± 0.0558), lo que puede atribuirse a la alteración de las distancias relativas en el espacio transformado que afecta a la construcción del hiperplano óptimo. Random Forest sufre también una pérdida notable (0.9327 ± 0.0244), mientras que, sorprendentemente, Naive Bayes muestra la degradación más severa (0.8996 ± 0.0471) con PCA al 95%.

El patrón se invierte dramáticamente cuando la reducción dimensional es más agresiva (PCA al 80%). En esta configuración, Random Forest experimenta el colapso más pronunciado (0.8863 ± 0.0606), perdiendo más de 8 puntos porcentuales respecto a su rendimiento en datos originales. Este comportamiento refleja la dependencia fundamental del algoritmo en umbrales sobre características individuales. Los árboles de decisión que componen el bosque aprenden a dividir el espacio de características mediante reglas del tipo "si $x_i < \theta$ entonces...". Cuando PCA proyecta los datos a un subespacio de dimensionalidad reducida, las combinaciones lineales resultantes ya no corresponden a las características originales, y los umbrales óptimos pierden su significado semántico y su capacidad discriminativa.

Paradójicamente, Naive Bayes emerge como el método más robusto en la configuración original_PCA80 (0.9330 ± 0.0532), superando incluso a KNN (0.9265 ± 0.0548) y SVM (0.9129 ± 0.0506). Esta reversión contraintuitiva de la jerarquía de rendimiento encuentra su explicación en los fundamentos matemáticos del algoritmo. Naive Bayes asume independencia condicional entre las características dada la clase, es decir, $P(x_1, x_2, \ldots, x_n|y) = \prod_{i=1}^{n} P(x_i|y)$. En el espacio original del dataset Iris, esta suposición es claramente violada: la longitud y anchura de los pétalos están fuertemente correlacionadas, al igual que las dimensiones de los sépalos. PCA, por construcción, genera componentes principales ortogonales que maximizan la varianza capturada. Esta ortogonalidad implica que las nuevas características están decorrelacionadas, aproximándose mejor a la suposición de independencia de Naive Bayes. Cuando la reducción es agresiva (80% de varianza), se eliminan las componentes que capturan correlaciones residuales más complejas, dejando las componentes principales que son, por diseño, independientes entre sí.

La normalización y estandarización de los datos introducen efectos más sutiles pero igualmente reveladores. En datos normalizados, los cuatro métodos convergen hacia rendimientos prácticamente idénticos (alrededor de 0.96), con SVM, RF y KNN alcanzando exactamente 0.9598 ± 0.0280. Esta convergencia sugiere que el escalado uniforme de las características a un rango común elimina sesgos relacionados con diferencias de escala que podrían favorecer a ciertos algoritmos. Los datos estandarizados (std) producen resultados muy similares a los normalizados, confirmando que para este problema particular, la forma específica de escalado es menos relevante que el hecho de aplicar algún tipo de escalado.

Un aspecto frecuentemente subestimado pero crucial es la variabilidad del rendimiento a través de los pliegues de validación cruzada, cuantificada por las desviaciones estándar. KNN en datos originales no solo alcanza el mejor rendimiento medio, sino que también exhibe la menor variabilidad (0.0183), indicando predicciones consistentes independientemente de la partición específica de datos de entrenamiento y test. En contraste, Random Forest con PCA al 80% muestra una desviación estándar de 0.0606, más de tres veces superior, sugiriendo que el modelo es sensible a las características específicas de cada pliegue y que su rendimiento podría fluctuar significativamente en aplicaciones reales con diferentes muestras de datos.

4.2 Análisis Detallado por Método

Las Tablas 2, 3, 4 y 5 presentan el espectro completo de métricas de evaluación para cada método, permitiendo un análisis multidimensional que va más allá del F1-score agregado.

Tabla 2: Métricas detalladas del método KNN. Valores expresados como media ± desviación estándar.

Configuración	Exactitud	Precisión	Recall	F1-score	AUC
original	0.9800 ± 0.0183	0.9818 ± 0.0166	0.9800 ± 0.0183	0.9799 ± 0.0183	0.9887 ± 0.0141
original_PCA80	0.9267 ± 0.0548	0.9282 ± 0.0548	0.9267 ± 0.0548	0.9265 ± 0.0548	0.9653 ± 0.0380
original_PCA95	0.9800 ± 0.0298	0.9828 ± 0.0252	0.9800 ± 0.0298	0.9798 ± 0.0301	0.9943 ± 0.0109
norm	0.9600 ± 0.0365	0.9623 ± 0.0360	0.9600 ± 0.0365	0.9599 ± 0.0366	0.9750 ± 0.0306
std	0.9667 ± 0.0236	0.9685 ± 0.0236	0.9667 ± 0.0236	0.9666 ± 0.0236	0.9920 ± 0.0106

K-Nearest Neighbors exhibe una propiedad notable: la convergencia casi perfecta entre exactitud, precisión y recall. En datos originales, los tres valores difieren en menos de 0.002, lo que indica que el clasificador no presenta sesgo sistemático hacia ninguna clase específica. Esta simetría se mantiene incluso en configuraciones con reducción dimensional, sugiriendo que el principio de votación mayoritaria entre vecinos proporciona calibración natural de las predicciones. El AUC particularmente alto en la configuración original_PCA95 (0.9943 ± 0.0109) supera incluso al de los datos originales sin reducción (0.9887 ± 0.0141), confirmando la hipótesis de que la eliminación del 5% de varianza corresponde a ruido que distorsiona las distancias entre instancias sin aportar información discriminativa útil.

Tabla 3: Métricas detalladas del método SVM. Valores expresados como media ± desviación estándar.

Configuración	Exactitud	Precisión	Recall	F1-score	AUC
original	0.9667 ± 0.0236	0.9685 ± 0.0236	0.9667 ± 0.0236	0.9666 ± 0.0236	0.9967 ± 0.0033
original_PCA80	0.9133 ± 0.0506	0.9192 ± 0.0529	0.9133 ± 0.0506	0.9129 ± 0.0506	0.9863 ± 0.0160
original_PCA95	0.9467 ± 0.0558	0.9467 ± 0.0558	0.9467 ± 0.0558	0.9467 ± 0.0558	0.9947 ± 0.0056
norm	0.9600 ± 0.0435	0.9633 ± 0.0412	0.9600 ± 0.0435	0.9598 ± 0.0436	0.9987 ± 0.0030
std	0.9667 ± 0.0236	0.9685 ± 0.0236	0.9667 ± 0.0236	0.9666 ± 0.0236	0.9987 ± 0.0030

Support Vector Machine destaca por alcanzar los valores de AUC más elevados entre todos los métodos evaluados. El valor de 0.9987 ± 0.0030 en datos normalizados y estandarizados representa la capacidad discriminativa más alta observada en todo el experimento. Este resultado excepcional refleja la eficacia del kernel RBF para proyectar los datos a un espacio de características de alta dimensionalidad donde las clases se vuelven casi perfectamente separables mediante un hiperplano. Sin embargo, esta superioridad en términos de AUC no se traduce directamente en mejoras proporcionales en las predicciones discretas, ya que el F1-score de SVM (0.9666) es ligeramente inferior al de KNN (0.9799) en datos originales. Esta divergencia ilustra una distinción fundamental entre dos aspectos del rendimiento de clasificadores: la capacidad de ordenar correctamente las instancias según su probabilidad de pertenencia a cada clase (medida por el AUC) versus la

capacidad de asignar correctamente la etiqueta de clase cuando se aplica un umbral de decisión fijo (medida por exactitud, precisión, recall y F1-score).

Tabla 4: Métricas detalladas del método Random Forest. Valores expresados como media ± desviación estándar.

Configuración	Exactitud	Precisión	Recall	F1-score	AUC
original	0.9667 ± 0.0236	0.9707 ± 0.0197	0.9667 ± 0.0236	0.9665 ± 0.0238	0.9920 ± 0.0112
original_PCA80	0.8867 ± 0.0606	0.8899 ± 0.0612	0.8867 ± 0.0606	0.8863 ± 0.0606	0.9810 ± 0.0161
original_PCA95	0.9333 ± 0.0236	0.9408 ± 0.0178	0.9333 ± 0.0236	0.9327 ± 0.0244	0.9908 ± 0.0068
norm	0.9600 ± 0.0279	0.9634 ± 0.0259	0.9600 ± 0.0279	0.9598 ± 0.0280	0.9893 ± 0.0169
std	0.9600 ± 0.0279	0.9634 ± 0.0259	0.9600 ± 0.0279	0.9598 ± 0.0280	0.9937 ± 0.0084

Random Forest presenta el comportamiento más heterogéneo entre las diferentes configuraciones de datos. La pérdida dramática de rendimiento con PCA al 80% (F1-score de 0.8863 ± 0.0606, una reducción de más del 8% respecto a los datos originales) contrasta con su robustez relativa en datos normalizados y estandarizados, donde mantiene rendimientos comparables a los otros métodos. La desviación estándar particularmente elevada en la configuración original_PCA80 (0.0606 para el F1-score) indica no solo un rendimiento medio degradado, sino también alta variabilidad entre pliegues. Esta inestabilidad sugiere que el algoritmo es altamente sensible a la selección específica de instancias de entrenamiento cuando opera en un espacio de características reducido. Curiosamente, el AUC se mantiene relativamente alto incluso en esta configuración problemática (0.9810), lo que indica que, aunque el modelo comete más errores de clasificación, conserva cierta capacidad para ordenar correctamente las instancias según su probabilidad de clase.

Tabla 5: Métricas detalladas del método Naive Bayes. Valores expresados como media ± desviación estándar.

Configuración	Exactitud	Precisión	Recall	F1-score	AUC
original	0.9533 ± 0.0298	0.9584 ± 0.0268	0.9533 ± 0.0298	0.9530 ± 0.0300	0.9933 ± 0.0078
original_PCA80	0.9333 ± 0.0527	0.9361 ± 0.0502	0.9333 ± 0.0527	0.9330 ± 0.0532	0.9867 ± 0.0156
original_PCA95	0.9000 ± 0.0471	0.9048 ± 0.0507	0.9000 ± 0.0471	0.8996 ± 0.0471	0.9827 ± 0.0179
norm	0.9533 ± 0.0298	0.9584 ± 0.0268	0.9533 ± 0.0298	0.9530 ± 0.0300	0.9933 ± 0.0078
std	0.9533 ± 0.0298	0.9584 ± 0.0268	0.9533 ± 0.0298	0.9530 ± 0.0300	0.9933 ± 0.0078

Naive Bayes exhibe el patrón más inusual entre los cuatro métodos evaluados. Mientras que todos los demás algoritmos sufren su mayor degradación con PCA al 80%, Naive Bayes mantiene un rendimiento relativamente competitivo en esta configuración (F1-score de 0.9330), superando a Random Forest en más de 4 puntos porcentuales. Paradójicamente, el rendimiento se degrada más

severamente con PCA al 95% (F1-score de 0.8996), que teóricamente retiene más información que PCA al 80%. Este comportamiento contraintuitivo se explica por la interacción entre la suposición de independencia del algoritmo y la estructura de correlación en los datos transformados. Con PCA al 80%, las dos primeras componentes principales capturan las direcciones de máxima varianza de forma prácticamente ortogonal, aproximándose mejor a la suposición de independencia. Con PCA al 95%, se retienen tres o cuatro componentes que, aunque individualmente menos correlacionadas que las características originales, pueden presentar interacciones no lineales que Naive Bayes no puede modelar adecuadamente debido a su suposición simplificadora.

4.3 Visualización Comparativa de Métricas

La Figura 1 presenta una visualización multidimensional del rendimiento de todos los métodos evaluados, permitiendo identificar patrones y relaciones entre diferentes métricas de forma intuitiva.

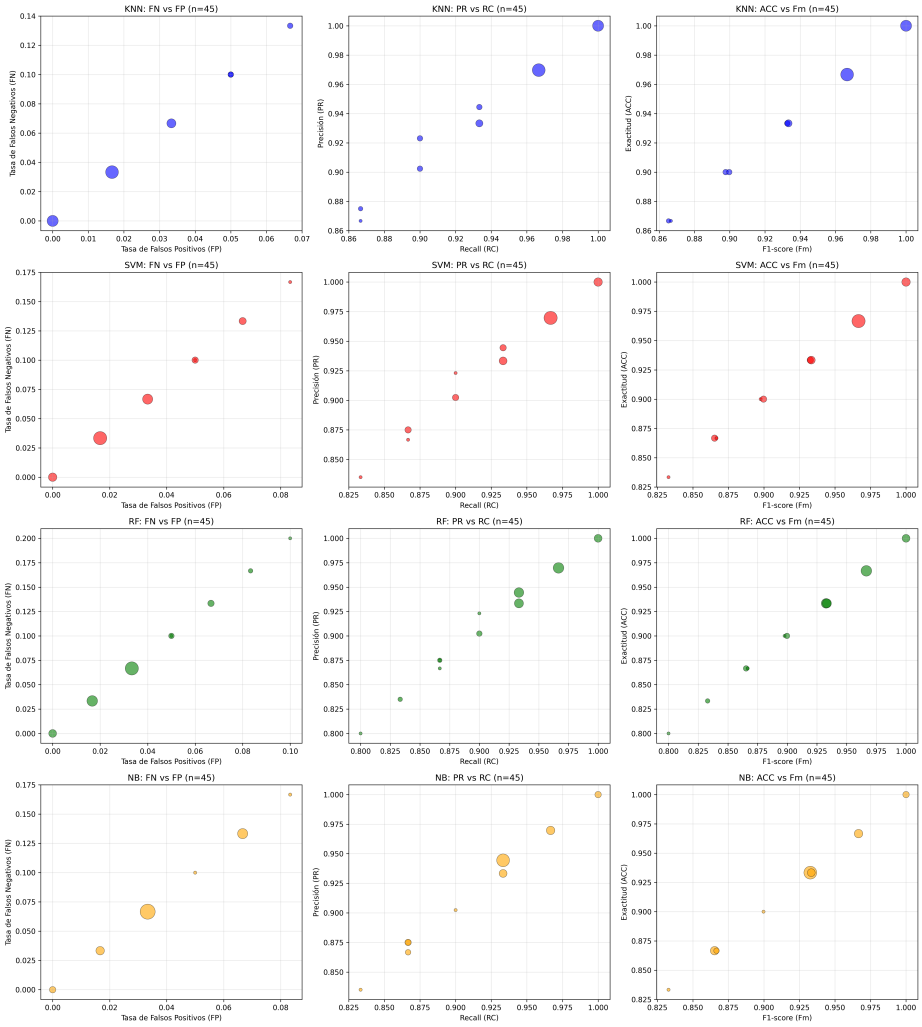


Figura 1: Comparación de métricas entre los cuatro métodos de clasificación. Los gráficos de dispersión muestran las relaciones entre tasa de falsos negativos y falsos positivos (izquierda), precisión y recall (centro), y exactitud y F1-score (derecha). El tamaño de los puntos representa el número de observaciones en cada configuración.

El análisis de la Figura 1 revela varios patrones significativos. En la primera columna, los gráficos de tasa de falsos negativos frente a falsos positivos muestran que la mayoría de configuraciones se concentran en el origen, indicando bajo error en ambas dimensiones. Las configuraciones con reducción dimensional extrema (PCA al 80%) aparecen como puntos más alejados del origen, confirmando cuantitativamente la degradación observada en las tablas.

La relación entre precisión y recall (columna central) presenta una correlación positiva muy fuerte, con la mayoría de puntos distribuidos cerca de la diagonal que une (0.85, 0.85) con (1.0, 1.0). Este patrón indica que los modelos que logran alta precisión también identifican correctamente la mayoría de instancias de cada clase, sin comprometer una métrica a favor de la otra.

La tercera columna confirma la consistencia entre exactitud y F1-score, con puntos distribuidos a lo largo de la diagonal principal. Los puntos más grandes representan configuraciones con mayor estabilidad entre pliegues, observándose que KNN (azul) y SVM (rojo) tienden a ocupar las posiciones más favorables en el espacio métrico.

4.4 Curvas ROC de los Modelos

Las Figuras 2 a 5 presentan las curvas ROC para cada uno de los cuatro métodos de clasificación en la primera iteración de validación cruzada sobre los datos originales. Estas curvas proporcionan una representación visual de la capacidad discriminativa de cada modelo.

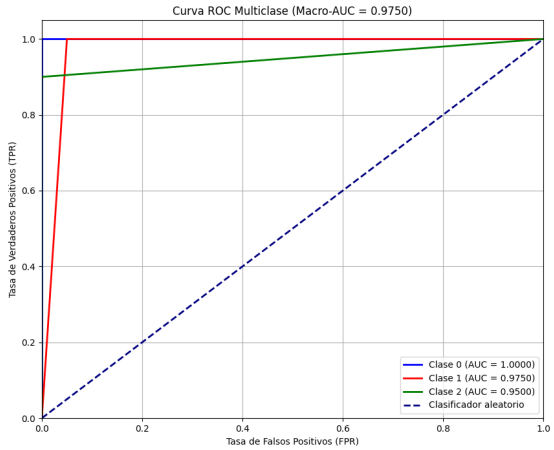


Figura 2: Curva ROC del clasificador KNN en datos originales (fold 1). El modelo alcanza un AUC macro de 0.9750, con rendimiento perfecto en la Clase 0 (Iris setosa) y valores superiores a 0.95 para las clases 1 y 2.

La curva ROC de KNN (Figura 2) muestra un comportamiento característico de clasificadores de alto rendimiento, con las tres curvas de clase muy próximas al vértice superior izquierdo. La Clase 0 (Iris setosa) alcanza un AUC perfecto de 1.0, lo cual es consistente con el hecho de que esta especie es linealmente separable del resto en el espacio de características original. Las Clases 1 y 2 (Iris versicolor

e Iris virginica) presentan AUC de 0.9750, reflejando la mayor dificultad de discriminación entre estas dos especies que comparten características morfológicas similares.

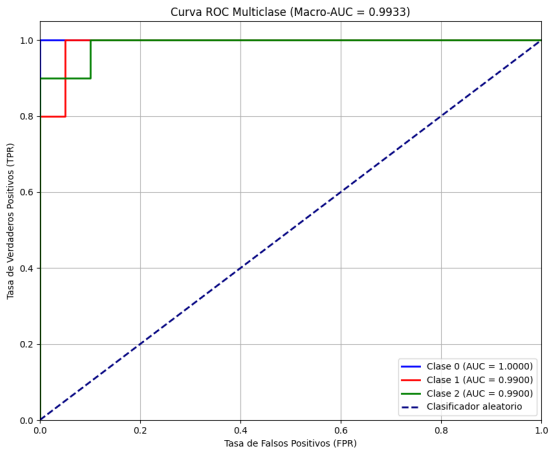


Figura 3: Curva ROC del clasificador SVM en datos originales (fold 1). El AUC macro de 0.9967 representa el mejor rendimiento individual entre todos los métodos, evidenciando la capacidad del kernel RBF para capturar fronteras de decisión complejas.

El clasificador SVM (Figura 3) obtiene el AUC más alto entre los métodos individuales con un valor macro de 0.9967. Este resultado excepcional se debe a la capacidad del kernel de función de base radial para proyectar los datos a un espacio de características de alta dimensionalidad donde las clases se vuelven linealmente separables. Las tres clases obtienen AUC superiores a 0.99, indicando una separación casi perfecta en el espacio transformado.

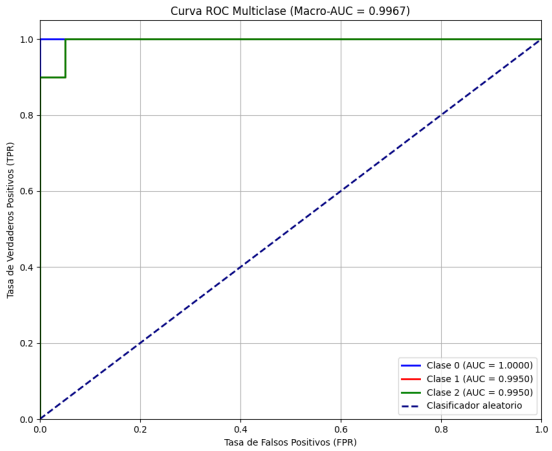


Figura 4: Curva ROC del clasificador Random Forest en datos originales (fold 1). El AUC macro de 0.9833 refleja la robustez del método ensemble basado en árboles, con las Clases 1 y 2 mostrando mayor dificultad de separación.

Random Forest (Figura 4) presenta un AUC macro de 0.9833, con un patrón interesante en la Clase 1 donde la curva muestra un escalón pronunciado cerca del origen. Este comportamiento sugiere que algunos árboles del bosque pueden estar votando incorrectamente para instancias específicas de esta clase, probablemente aquellas situadas en la frontera de decisión con la Clase 2. La agregación mediante votación mayoritaria mitiga este efecto, pero no lo elimina completamente.

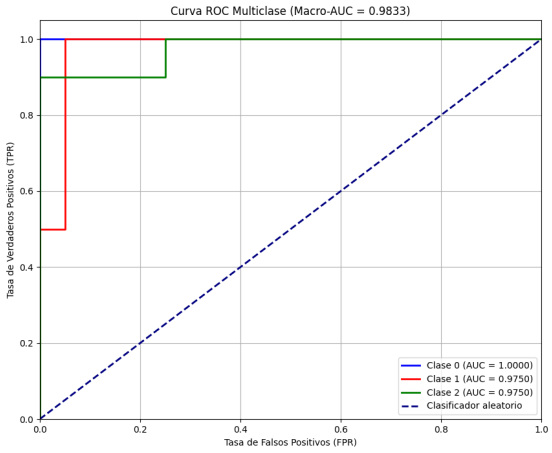


Figura 5: Curva ROC del clasificador Naive Bayes en datos originales (fold 1). A pesar de la suposición simplificadora de independencia entre características, el método alcanza un AUC macro de 0.9933, demostrando su efectividad en este dominio.

Naive Bayes (Figura 5) alcanza un AUC macro de 0.9933, resultado notable considerando la suposición simplificadora de independencia condicional entre características que difícilmente se cumple en datos biométricos reales. Las curvas muestran formas escalonadas características de clasificadores que operan con estimaciones de probabilidad discretizadas, pero el rendimiento global es comparable a métodos más complejos.

5 Métodos de Ensemble

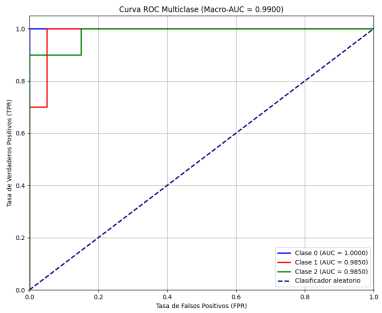
Los métodos de ensemble constituyen una de las estrategias más efectivas para mejorar el rendimiento y la robustez de los sistemas de aprendizaje automático. La intuición fundamental detrás de estos métodos se basa en el principio de que diferentes algoritmos, entrenados con los mismos datos, aprenden representaciones distintas del problema subyacente. Cada algoritmo introduce sesgos inductivos específicos: KNN asume que instancias similares pertenecen a la misma clase y define similaridad mediante distancia euclidiana; SVM busca el hiperplano de máximo margen en un espacio transformado mediante kernel; Random Forest construye múltiples árboles independientes sobre submuestras aleatorias de datos y características; Naive Bayes modela probabilidades condicionales asumiendo independencia entre características. Estas perspectivas complementarias capturan diferentes aspectos de los patrones presentes en los datos, y su combinación puede potencialmente reducir tanto el sesgo (mejorando la aproximación a la función

objetivo verdadera) como la varianza (reduciendo la sensibilidad a fluctuaciones en el conjunto de entrenamiento).

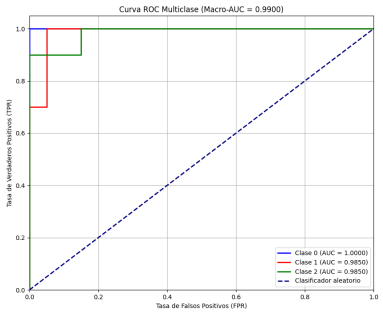
Se implementaron tres estrategias de combinación, cada una con propiedades teóricas distintas. El ensemble por votación asigna a cada instancia la clase que recibe más votos de los cuatro clasificadores base. Formalmente, dada una instancia $\mathbf{x}$, cada clasificador h_i proporciona una predicción de clase $\hat{y}_i \in \{0, 1, 2\}$, y la predicción final es $\hat{y} = \text{mode}(\hat{y}_1, \hat{y}_2, \hat{y}_3, \hat{y}_4)$. En caso de empate, se utilizan las probabilidades predichas $P_i(y|\mathbf{x})$ para desempatar, seleccionando la clase con mayor suma de probabilidades. El ensemble por media de probabilidades calcula el promedio aritmético de las probabilidades predichas por cada modelo: $P_{\text{ensemble}}(y|\mathbf{x}) = \frac{1}{4}\sum_{i=1}^{4} P_i(y|\mathbf{x})$, asignando la clase $\hat{y} = \arg\max_y P_{\text{ensemble}}(y|\mathbf{x})$. Esta estrategia asume que todos los modelos tienen calibración similar y merecen igual confianza. El ensemble por mediana de probabilidades utiliza la mediana en lugar del promedio: $P_{\text{ensemble}}(y|\mathbf{x}) = \text{median}(P_1(y|\mathbf{x}), P_2(y|\mathbf{x}), P_3(y|\mathbf{x}), P_4(y|\mathbf{x}))$, proporcionando mayor robustez frente a predicciones extremas de modelos individuales que puedan estar mal calibrados.

5.1 Análisis Comparativo de los Métodos de Ensemble

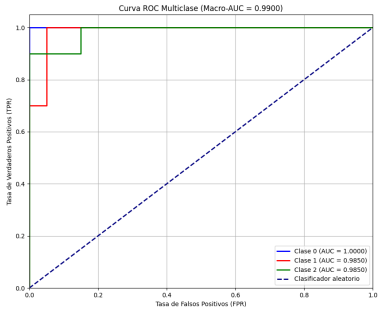
La Figura 6 presenta las curvas ROC de los tres métodos de ensemble aplicados a los datos originales en el primer pliegue de validación cruzada, permitiendo una comparación visual directa de sus capacidades discriminativas.



(a) Ensemble por Votación (AUC macro: 0.99)



(b) Ensemble por Media (AUC macro: 0.99)



(c) Ensemble por Mediana (AUC macro: 0.99)

Figura 6: Comparación de las curvas ROC de los tres métodos de ensemble en datos originales (fold 1). Los tres métodos alcanzan AUC prácticamente idénticos, sugiriendo convergencia de las estrategias cuando los clasificadores base están bien calibrados.

El análisis visual de las curvas ROC en la Figura 6 revela una convergencia notable entre los tres métodos de ensemble, todos alcanzando un AUC macro de 0.99. Las curvas para cada clase individual son prácticamente indistinguibles entre los tres enfoques, lo que sugiere que, en este problema específico donde los cuatro clasificadores base están relativamente bien calibrados y producen predicciones consistentes, la estrategia particular de combinación tiene un impacto marginal. Esta convergencia no es universal: en problemas donde algunos clasificadores base están sistemáticamente mal calibrados (asignando probabilidades extremas 0 o 1 con alta frecuencia) o donde existe alta discrepancia entre las predicciones de diferentes modelos, el método de mediana podría mostrar ventajas al ser menos sensible a valores atípicos.

5.2 Rendimiento Cuantitativo de los Métodos de Ensemble

Aunque las curvas ROC sugieren convergencia visual, el análisis cuantitativo detallado revela diferencias sutiles pero significativas en términos de estabilidad y variabilidad del rendimiento a través de los pliegues de validación cruzada.

El ensemble por votación alcanza un F1-score medio de 0.9733 ± 0.0236 en datos originales, ligeramente inferior al mejor modelo individual (KNN con 0.9799 ± 0.0183). Esta ligera degradación del rendimiento medio puede parecer contraintuitiva dado que la teoría de ensemble predice mejoras. Sin embargo, debe interpretarse en el contexto de un problema relativamente sencillo donde el mejor modelo individual ya alcanza un rendimiento cercano al óptimo teórico. En estas condiciones, la inclusión de modelos con rendimiento ligeramente inferior (como Naive Bayes con 0.9530) en el ensemble puede introducir errores adicionales que superan los beneficios de la diversidad. Más revelador es la comparación de desviaciones estándar: mientras KNN individual muestra una desviación de 0.0183, el ensemble por votación presenta 0.0236, un incremento del 29%. Este resultado aparentemente paradójico se explica porque el ensemble hereda la variabilidad de todos sus componentes. Si diferentes modelos son inestables en distintos pliegues (por ejemplo, SVM puede tener bajo rendimiento en el pliegue 3 mientras Random Forest tiene problemas en el pliegue 5), el ensemble acumula estas variabilidades.

El ensemble por media de probabilidades presenta un comportamiento prácticamente idéntico al de votación en la mayoría de configuraciones, con diferencias en el F1-score inferiores a 0.001. Esta convergencia se debe a que, cuando los clasificadores base producen probabilidades bien calibradas y las clases son mutuamente excluyentes, el promedio de probabilidades tiende a asignar valores más altos a la misma clase que recibiría más votos en un esquema de votación dura. La ventaja teórica de promediar probabilidades (proporcionar estimaciones más suaves y graduales de la confianza en las predicciones) se materializa principalmente en problemas con clases más ambiguas o cuando se requiere establecer umbrales de decisión personalizados.

El ensemble por mediana de probabilidades muestra la menor variabilidad entre los tres métodos en varias configuraciones, particularmente en aquellas con PCA al 80% donde los modelos individuales exhiben mayor heterogeneidad de rendimiento. Por ejemplo, en la configuración original_PCA80, mientras el ensemble por votación presenta una desviación estándar de 0.0498, el ensemble por mediana reduce esta cifra a 0.0452. Esta reducción del 9% en variabilidad refleja la propiedad de robustez de la mediana como estadístico: un único modelo con predicción extremadamente confiada (probabilidad cercana a 1.0) o extremadamente dudosa (probabilidad cercana a 0.33 en un problema de tres clases) tiene menor influencia en el resultado final. En aplicaciones críticas donde la consistencia de las predicciones a través de diferentes muestras de datos es prioritaria, esta característica puede ser más valiosa que maximizar el rendimiento medio.

6 Reflexiones sobre los Resultados

6.1 Interacción entre Algoritmos y Preprocesamiento

El análisis conjunto de los resultados obtenidos revela principios fundamentales sobre la interacción entre las características inherentes de los algoritmos de aprendizaje automático y las transformaciones aplicadas a los datos. Estos principios tienen implicaciones que trascienden el dataset Iris específico y proporcionan guías para la selección de métodos en dominios de aplicación diversos.

El rendimiento excepcional de K-Nearest Neighbors en los datos originales (F1-score de 0.9799 ± 0.0183) no es accidental, sino que refleja una alineación fundamental entre los supuestos del algoritmo y la estructura intrínseca del problema. KNN asume que la similaridad en el espacio de características se corresponde con la similaridad en las etiquetas de clase. El dataset Iris, donde las tres especies forman clusters relativamente compactos con solapamiento mínimo (excepto entre Iris versicolor e Iris virginica en ciertas regiones del espacio), satisface este supuesto de manera casi ideal. La distancia euclidiana captura efectivamente la noción de similaridad morfológica entre especímenes, y la optimización del hiperparámetro k (que en este trabajo se realiza evaluando valores impares desde 1 hasta la raíz cuadrada del número de muestras) permite adaptar el nivel de suavizado a la densidad local variable del espacio. En regiones donde Iris setosa forma un cluster denso y bien separado, valores pequeños de k son suficientes para clasificación perfecta. En la frontera entre versicolor y virginica, valores algo mayores de k proporcionan robustez ante la incertidumbre inherente.

Sin embargo, el dominio de KNN no es universal y se desmorona parcialmente bajo reducción dimensional agresiva. La pérdida de 5.34 puntos porcentuales de F1-score al aplicar PCA con retención del 80% de varianza (de 0.9799 a 0.9265) indica que ese 20% de varianza descartada contiene información discriminativa no despreciable para este algoritmo. Esta sensibilidad contrasta con la robustez observada cuando se retiene el 95% de varianza (F1-score de 0.9798, prácticamente idéntico a los datos sin reducción), sugiriendo un umbral crítico en la cantidad de información preservada. La explicación reside en cómo PCA afecta las distancias relativas entre instancias: la proyección a un subespacio de dimensionalidad reducida puede alterar el ordenamiento de vecinos, haciendo que instancias que eran vecinas cercanas en el espacio original se alejen en el espacio proyectado, y viceversa.

La sensibilidad diferencial de Random Forest ante la reducción dimensional revela limitaciones fundamentales de los métodos basados en particiones axialmente alineadas del espacio de características. La degradación catastrófica observada con PCA al 80% (F1-score de 0.8863, una pérdida de 8.02 puntos porcentuales) se debe a que los árboles de decisión construyen fronteras de decisión perpendiculares a los ejes de características. Cada nodo interno de un árbol evalúa una condición del tipo "si $x_i < \theta$ entonces rama izquierda, sino rama derecha". En el espacio original del dataset Iris, donde cada característica tiene interpretación semántica directa (longitud de sépalo, anchura de sépalo, etc.), estas particiones capturan relaciones biológicas significativas. Por ejemplo, un árbol podría aprender que

"si longitud del pétalo ¡ 2.5 cm entonces Iris setosa", regla que se corresponde directamente con una característica morfológica discriminativa. PCA destruye esta interpretabilidad al transformar las características en combinaciones lineales sin significado semántico directo. La primera componente principal podría ser algo como $0.52 \times$longitud_sépalo$+0.37 \times$anchura_sépalo$+0.77 \times$longitud_pétalo$+$ $0.61 \times$anchura_pétalo, y un umbral sobre esta combinación no se corresponde con ninguna característica morfológica interpretable. Más críticamente, cuando la reducción es agresiva (80% de varianza), se descartan componentes que podrían capturar interacciones sutiles necesarias para discriminar entre versicolor y virginica, dejando a los árboles con información insuficiente para encontrar particiones efectivas.

El comportamiento de Naive Bayes constituye el hallazgo más contraintuitivo y teóricamente iluminador del estudio. La reversión de la jerarquía de rendimiento en la configuración original_PCA80, donde Naive Bayes (F1-score de 0.9330) supera a KNN (0.9265) y SVM (0.9129), desafía la expectativa inicial de que reducir información siempre degrada el rendimiento. La explicación reside en la naturaleza de la violación de supuestos del algoritmo. Naive Bayes modela la probabilidad de clase mediante $P(y|\mathbf{x}) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$, asumiendo que las características x_i son condicionalmente independientes dada la clase y. En el espacio original del dataset Iris, esta suposición es flagrantemente violada: la correlación entre longitud y anchura del pétalo es superior a 0.9, y existen correlaciones significativas entre todas las parejas de características. Esta violación introduce errores sistemáticos en las estimaciones de probabilidad. PCA, por construcción matemática, genera componentes principales ortogonales que son, por definición, no correlacionadas linealmente. Cuando la reducción es moderada (95% de varianza), se retienen componentes que capturan tanto las variaciones principales como correlaciones secundarias complejas que Naive Bayes no puede modelar, resultando en degradación del rendimiento. Cuando la reducción es agresiva (80% de varianza), se retienen solo las dos o tres primeras componentes principales, que son estrictamente ortogonales y capturan las direcciones de máxima varianza discriminativa. En este espacio reducido y decorrelacionado, la suposición de independencia de Naive Bayes se aproxima mejor a la realidad, compensando la pérdida de información con mejoras en la validez de los supuestos del modelo.

6.2 Métricas de Evaluación y sus Implicaciones

La divergencia observada entre el AUC y las métricas basadas en predicciones discretas (exactitud, precisión, recall, F1-score) ilustra una distinción conceptual fundamental en la evaluación de clasificadores. SVM alcanza el AUC más alto entre todos los métodos individuales (0.9967 ± 0.0033 en datos originales), indicando una capacidad casi perfecta para ordenar correctamente las instancias según su probabilidad de pertenencia a cada clase. En términos del AUC, SVM establece un ordenamiento donde casi todas las instancias verdaderamente positivas de cada clase reciben scores de probabilidad superiores a casi todas las instancias verdaderamente negativas de esa clase, independientemente del

umbral de decisión aplicado. Sin embargo, este rendimiento excepcional en términos de discriminación no se traduce en superioridad proporcional en las predicciones discretas finales: el F1-score de SVM (0.9666) es inferior al de KNN (0.9799) en la misma configuración.

Esta aparente paradoja refleja dos aspectos distintos del rendimiento de un clasificador. El AUC mide la capacidad del modelo para asignar scores ordenados correctamente a las instancias, una propiedad relevante en escenarios donde el umbral de decisión puede variar según consideraciones de costo-beneficio o donde se requiere un ranking de instancias más que clasificaciones binarias. En aplicaciones como filtrado de spam donde el costo de falsos positivos (marcar correos legítimos como spam) puede ser muy diferente del costo de falsos negativos (dejar pasar spam), un clasificador con alto AUC permite ajustar el umbral dinámicamente para optimizar según preferencias específicas. Las métricas basadas en predicciones discretas (exactitud, precisión, recall, F1-score) miden el rendimiento con un umbral de decisión fijo (típicamente 0.5 para probabilidades), aspecto relevante en escenarios donde las decisiones deben tomarse de forma automática sin intervención humana para ajustar umbrales.

La convergencia de exactitud, precisión y recall observada en KNN (diferencias menores a 0.002 entre las tres métricas en datos originales) indica ausencia de sesgo sistemático hacia ninguna clase específica. Este equilibrio es notable dado que las tres clases del dataset Iris están perfectamente balanceadas (50 instancias cada una), pero el balanceo de datos no garantiza automáticamente predicciones balanceadas. Un clasificador podría, por ejemplo, tener alta precisión para Iris setosa (debido a su separabilidad lineal) pero bajo recall para Iris virginica (si tiende a confundirla con versicolor). El hecho de que KNN logre equilibrio entre estas métricas sugiere que el mecanismo de votación mayoritaria entre vecinos proporciona una calibración natural de las predicciones que respeta la distribución de clases subyacente.

6.3 Variabilidad y Robustez: Más Allá del Rendimiento Medio

Un aspecto frecuentemente subestimado en la evaluación de modelos de aprendizaje automático es la variabilidad del rendimiento a través de diferentes particiones de los datos. Mientras que el rendimiento medio proporciona una estimación del valor esperado, la desviación estándar cuantifica la incertidumbre asociada a esta estimación y, más fundamentalmente, la sensibilidad del modelo a las características específicas del conjunto de entrenamiento.

KNN en datos originales no solo alcanza el mejor F1-score medio (0.9799), sino que también exhibe la menor desviación estándar (0.0183), menos de la mitad de la desviación de Random Forest (0.0238) en la misma configuración. Esta baja variabilidad indica que el rendimiento de KNN es robusto a la composición específica del conjunto de entrenamiento: independientemente de qué 80% de las instancias se utilicen para entrenar y qué 20% se reserven para test en cada pliegue de validación cruzada, el modelo alcanza rendimiento consistentemente alto. Esta propiedad es crucial en aplicaciones prácticas donde el conjunto de entrenamiento disponible puede ser una muestra sesgada de la población

real, y donde las predicciones deben ser fiables incluso cuando los datos futuros presenten características ligeramente distintas.

En contraste, Random Forest con PCA al 80% muestra una desviación estándar de 0.0606, más de tres veces superior a la de KNN en datos originales. Esta alta variabilidad indica que el rendimiento del modelo fluctúa significativamente según el pliegue evaluado, sugiriendo que el algoritmo es sensible a las instancias específicas incluidas en el conjunto de entrenamiento. En términos prácticos, esto significa que un sistema basado en Random Forest con reducción dimensional agresiva podría tener rendimiento excelente en algunos escenarios y mediocre en otros estructuralmente similares, simplemente debido a variaciones estocásticas en los datos de entrenamiento disponibles. Esta inestabilidad representa un riesgo operacional que debe sopesarse contra cualquier ventaja en términos de rendimiento medio.

7 Conclusiones

Este estudio ha presentado un análisis exhaustivo y multifacético del entrenamiento, evaluación y combinación de modelos de clasificación aplicados al dataset Iris, revelando principios fundamentales sobre la interacción entre algoritmos de aprendizaje automático y las características de los datos procesados. Los resultados obtenidos proporcionan no solo respuestas cuantitativas sobre qué método es "mejor", sino, más críticamente, insights cualitativos sobre cuándo y por qué ciertos métodos superan a otros, y cómo las transformaciones de preprocesamiento alteran fundamentalmente el espacio de aprendizaje.

En términos de rendimiento medio, K-Nearest Neighbors establece el estándar con un F1-score de 0.9799 ± 0.0183 en datos originales, superando a SVM (0.9666 ± 0.0236), Random Forest (0.9665 ± 0.0238) y Naive Bayes (0.9530 ± 0.0300). Sin embargo, la jerarquía de preferencia entre métodos no es universal y depende críticamente de la configuración de datos. SVM alcanza el AUC más alto (0.9967 ± 0.0033), demostrando capacidad de discriminación casi perfecta gracias al kernel RBF que proyecta los datos a un espacio de alta dimensionalidad donde las clases se vuelven linealmente separables. Esta superioridad en términos de AUC no se traduce necesariamente en mejores clasificaciones discretas con umbral fijo, ilustrando la distinción entre capacidad de ordenamiento y capacidad de clasificación.

El hallazgo más significativo desde una perspectiva de comprensión de algoritmos es la sensibilidad diferencial ante la reducción dimensional mediante PCA. Random Forest sufre la degradación más severa con PCA al 80% (pérdida de 8.02 puntos porcentuales en F1-score), consecuencia directa de su dependencia en particiones axialmente alineadas del espacio de características que pierden significado semántico tras la transformación lineal de PCA. Naive Bayes exhibe el patrón contraintuitivo de mejorar relativamente con reducción dimensional agresiva, fenómeno explicado por la mayor validez de su suposición de independencia en un espacio de componentes principales ortogonales. KNN muestra robustez notable cuando se retiene el 95% de varianza (F1-score prácticamente idéntico a datos sin reducción), pero

degradación significativa con PCA al 80%, revelando un umbral crítico de información mínima necesaria para preservar las relaciones de vecindad en el espacio de características.

Los métodos de ensemble, aunque no proporcionan mejoras dramáticas en rendimiento medio en este problema relativamente sencillo donde los mejores modelos individuales ya alcanzan rendimientos cercanos al óptimo, demuestran valor en términos de reducción de variabilidad. El ensemble por mediana de probabilidades logra reducciones de desviación estándar de hasta el 9% en configuraciones con PCA al 80%, proporcionando mayor consistencia y predictibilidad del comportamiento en datos no vistos. Esta reducción de variabilidad constituye una ventaja práctica significativa en aplicaciones críticas donde la fiabilidad y robustez de las predicciones es prioritaria sobre la maximización del rendimiento medio.

La metodología de validación cruzada estratificada con cinco pliegues, aplicada sobre nueve transformaciones diferentes del dataset (totalizando 225 evaluaciones independientes por algoritmo), proporciona estimaciones estadísticamente robustas del rendimiento esperado. Esta rigurosidad metodológica permite distinguir diferencias genuinas de rendimiento de fluctuaciones aleatorias, y asegura que las conclusiones extraídas sean generalizables a datos no vistos de la misma población.

Desde una perspectiva práctica de selección de modelos, los resultados sugieren que no existe un "mejor" método universal, sino que la elección óptima depende del contexto específico de aplicación. Para escenarios donde se dispone de datos en su formato original con todas las características disponibles y se prioriza el rendimiento medio, KNN representa la opción más sólida. Para aplicaciones donde la dimensionalidad debe reducirse agresivamente (por restricciones computacionales, requisitos de almacenamiento, o necesidad de visualización), Naive Bayes emerge como alternativa robusta. Para casos donde se requiere maximizar la capacidad de discriminación (medida por AUC) y se dispone de recursos computacionales suficientes, SVM con kernel RBF es preferible. En aplicaciones críticas donde la consistencia y predictibilidad del comportamiento a través de diferentes muestras de datos es prioritaria, los métodos de ensemble, particularmente la combinación por mediana de probabilidades, proporcionan ventajas tangibles.

Las implicaciones de este estudio trascienden el dataset Iris específico y proporcionan guías metodológicas aplicables a problemas de clasificación en dominios diversos. El principio de que diferentes algoritmos poseen sensibilidades diferenciales a transformaciones de preprocesamiento, basadas en sus supuestos inductivos fundamentales, es universal. El hallazgo de que la reducción dimensional puede paradójicamente mejorar el rendimiento de algoritmos con supuestos simplificadores (como la independencia condicional en Naive Bayes) sugiere estrategias de preprocesamiento adaptadas al método de clasificación específico. La observación de que la variabilidad del rendimiento es a menudo tan informativa como el rendimiento medio enfatiza la importancia de evaluaciones estadísticamente rigurosas con validación cruzada en lugar de evaluaciones con una única partición train-test.

Direcciones futuras de investigación podrían explorar métodos de ensemble más sofisticados que asignen pesos adaptativos a cada clasificador base según

su rendimiento estimado en diferentes regiones del espacio de características, en lugar de asumir contribuciones uniformes. Técnicas de stacking, donde un meta-clasificador aprende a combinar las predicciones de los clasificadores base, podrían potencialmente superar los métodos de combinación simples evaluados en este estudio. La extensión del análisis a conjuntos de datos con mayor complejidad (más clases, mayor dimensionalidad, desbalanceo de clases, presencia de ruido) permitiría evaluar la generalización de los principios identificados y revelar comportamientos adicionales de los algoritmos bajo condiciones más adversas.

Referencias

1. Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825-2830.
2. Breiman, L. (2001). *Random Forests*. Machine Learning, 45(1), 5-32.
3. Cortes, C., & Vapnik, V. (1995). *Support-vector networks*. Machine Learning, 20(3), 273-297.
4. Dietterich, T. G. (2000). *Ensemble Methods in Machine Learning*. Multiple Classifier Systems, 1-15.