Predicción de Clase de Animal mediante Clasificadores Bayesianos, Estimadores No Paramétricos y k-NN

Jordi Blasco Lozano

Abstract

Este trabajo aborda la práctica 2 en la asignatura de Fundamentos del Aprendizaje Automático empleando el dataset Zoo de UCI, aplicando seis algoritmos de clasificación: Naive Bayes Gaussiano, MLE Multivariante, Histogram Bayes, Parzen Windows, k-NN Density Bayes y k-NN Rule. Se analiza su rendimiento en una tarea real de clasificación multiclase (7 clases) con características binarias y clases minoritarias. Los resultados revelan que Naive Bayes, Parzen Windows y k-NN Rule logran clasificación perfecta, mientras que métodos más complejos fallan debido a la maldición de la dimensionalidad y la escasez de datos en clases minoritarias.

1. Introducción y justificación del dataset

He seleccionado el dataset Zoo de UCI porque cumple con los requerimientos del enunciado: clasificación multiclase (7 clases: mamífero, ave, reptil, pez, anfibio, invertebrado, insecto), 16 atributos binarios interpretables, y tamaño adecuado (101 instancias originales) para observar comportamientos de métodos paramétricos y no paramétricos.

Dataset balanceado: Dado el desbalance original (41 mamíferos vs 4 anfibios), generé sintéticamente nuevos animales mediante investigación profunda usando IA para los nuevos animales y sus características, alcanzando 287 instancias con 41 ejemplares por clase. Esto permite evaluar cómo el balanceo afecta el rendimiento de cada modelo, especialmente aquellos sensibles a datos escasos (MLE Full).

2. Dataset y Análisis Exploratorio

Dataset original: 101 instancias, 16 atributos binarios, 7 clases desbalanceadas (mamíferos: 41, aves: 20, reptiles: 5, peces: 13, anfibios: 4, invertebrados: 8, insectos: 10). Dataset balanceado: 287 instancias con 41 ejemplares por clase, generados sintéticamente mediante investigación de características específicas de cada tipo de animal.

Las 16 características binarias incluyen: pelo, huevos, vuela, acuático, depredador, dientes, columna vertebral, respira, venenoso, aletas, patas, cola, doméstico, tamaño. No se requiere normalización. Las etiquetas de clase se codifican numéricamente (0-6).

3. Metodología: Particionado y Validación

División Train-Test (80%-20%): División estratificada manteniendo proporciones de clases (original: 80 train/21 test; balanceado: 229 train/58 test).

Validación Cruzada: 5-fold estratificada solo en entrenamiento para optimizar hiperparámetros (bandwidth en Parzen, k en k-NN), evitando contaminación.

Métricas: F1-macro (métrica principal por desbalance), accuracy, precision, recall, matrices de confusión.

4. Modelos Implementados

Seis clasificadores: (1) Naive Bayes Gaussiano, (2) MLE Multivariante con covarianza completa, (3) Histogram Bayes, (4) Parzen Windows con kernel Gaussiano, (5) k-NN Density Bayes, (6) k-NN Rule.

5. Resultados

He evaluado los seis modelos en el conjunto de prueba de 21 muestras. La Table 1 muestra los resultados principales de tests y la Table 2 muestra los resultados del entrenamiento, para la tabla de tests y modelos parametricos he usado el hiperparametro con más precisión que me ha dado el entrenamiento.

Table 1. Resultados en conjunto de prueba

	Original		Balanceado	
Modelo	Acc	F1	Acc	F1
Naive Bayes	1.00	1.00	0.91	0.91
MLE Full	0.71	0.46	0.95	0.95
Histogram Bayes	0.38	0.08	0.16	0.07
Parzen (h=0.1)	1.00	1.00	0.95	0.95
k-NN Density (k=11)	0.48	0.57	0.74	0.68
k-NN Rule (k=1)	1.00	1.00	0.95	0.95

Table 2. Validación cruzada 5-fold en entrenamiento

	Original		Balanceado	
Modelo	F1	std	F1	std
Naive Bayes	0.85	0.14	0.84	0.04
MLE Full	0.53	0.10	0.87	0.01
Histogram Bayes	0.25	0.13	0.06	0.02
Parzen (h=0.1)	0.86	–	0.94	–
k-NN Density (k=11)	0.57	–	0.70	–
k-NN Rule (k=1)	0.83	–	0.94	–

5.1. Análisis detallado por modelo

5.1.1. Naive Bayes Gaussiano

Original: Accuracy = 1.0, F1-macro = 1.0
Balanceado: Accuracy = 0.91, F1-macro = 0.91

Análisis: En el dataset original desbalanceado, Naive Bayes logra clasificación perfecta (21/21 muestras correctas). Sin embargo, con el dataset balanceado (287 muestras, 41 por clase), el rendimiento baja ligeramente a 91%, revelando que la perfección inicial se debía en parte al pequeño tamaño del conjunto de prueba. A pesar de la asunción "naive" de independencia entre características, el modelo funciona excepcionalmente bien en ambos casos. La matriz de confusión muestra que las principales confusiones ocurren entre reptiles-anfibios y anfibios-insectos, lo cual es razonable dado que comparten características físicas similares. El modelo estima solo d parámetros por clase ($\mu_{y,i}$ y $\sigma^2_{y,i}$ para cada característica), lo que lo hace robusto incluso con clases minoritarias.

5.1.2. MLE Multivariante (Full Bayesian Gaussian)

Original: Accuracy = 0.71, F1-macro = 0.46
Balanceado: Accuracy = 0.95, F1-macro = 0.95

Análisis: Este modelo muestra la mejora más dramática al balancear los datos. En el dataset original, falla significativamente con clases minoritarias (reptiles, anfibios, invertebrados), clasificándolas erróneamente como mamíferos. La causa es que con solo 3-6 ejemplos en entrenamiento, la estimación de la matriz de covarianza completa 16×16 (136 parámetros únicos por clase) es imposible, resultando en matrices singulares o mal condicionadas. Con el dataset balanceado (41 ejemplos por clase), el modelo puede estimar correctamente las covarianzas y alcanza 95% de rendimiento, igualando a los mejores modelos. Esto demuestra que los modelos más complejos necesitan más datos para funcionar correctamente. La covarianza completa captura las correlaciones entre características, pero requiere $O(d^2)$ muestras por clase.

5.1.3. Histogram Bayes

Original: Accuracy = 0.38, F1-macro = 0.08
Balanceado: Accuracy = 0.16, F1-macro = 0.07

Análisis: El peor rendimiento de todos los modelos en ambos escenarios, evidenciando claramente la maldición de la dimensionalidad. Con 16 características binarias, existen $2^{16} = 65,536$ posibles combinaciones (bins). Incluso con 287 muestras en el dataset balanceado, la mayoría de bins permanecen vacíos, resultando en estimaciones de densidad nulas ($p(\mathbf{x}|y) = 0$) para la mayoría de regiones. El modelo colapsa prediciendo solo la clase mayoritaria por defecto. Los histogramas requieren que el número de muestras crezca exponencialmente con la dimensionalidad ($N \propto b^d$), lo cual es inviable en la práctica. Este resultado confirma lo estudiado en teoría sobre las limitaciones de los histogramas en espacios de alta dimensión.

5.1.4. Parzen Windows

Original: Accuracy = 1.0, F1-macro = 1.0
Balanceado: Accuracy = 0.95, F1-macro = 0.95

Análisis: Parzen windows con kernel Gaussiano mantiene excelente rendimiento en ambos escenarios. Con bandwidth $h = 0.1$ óptimo (obtenido por validación cruzada), el modelo produce estimaciones de densidad continuas y suaves que funcionan bien incluso con datos escasos. A diferencia de histogramas, el suavizado del kernel Gaussiano compensa la falta de muestras, evitando las discontinuidades artificiales de los bins. En el dataset balanceado, obtiene el mejor F1-macro en validación cruzada (0.94), superior a Naive Bayes (0.84), confirmando que el bandwidth optimizado generaliza correctamente. Las confusiones ocurren principalmente entre clases con características solapadas (anfibios-reptiles), lo cual es esperado. El método mitiga efectivamente la maldición de la dimensionalidad mediante suavizado continuo.

5.1.5. k-NN Density Bayes

Original: Accuracy = 0.48, F1-macro = 0.57
Balanceado: Accuracy = 0.74, F1-macro = 0.68

Análisis: Rendimiento mediocre que mejora con el balanceo pero sigue siendo inferior a otros métodos. Con $k = 11$ óptimo, los 11 vecinos más cercanos diluyen la señal local, resultando en estimación de densidad por volumen local inestable. El modelo adapta el volumen $V_k(\mathbf{x})$ localmente (pequeño en regiones densas, grande en dispersas), pero esta estrategia resulta problemática con clases que tienen distribuciones espaciales dispersas. En el dataset original, confunde masivamente mamíferos como reptiles. Con el balanceo mejora, pero las confusiones persisten entre reptiles-peces y insectos distribuidos entre múltiples clases. Comparado con Parzen (que fija bandwidth globalmente), la adaptación local de volumen es menos efectiva en este dataset. La estimación explícita de densidad añade complejidad sin beneficio claro.

5.1.6. k-NN Rule

Original: Accuracy = 1.0, F1-macro = 1.0
Balanceado: Accuracy = 0.95, F1-macro = 0.95

Análisis: El método clásico de k-NN mantiene excelente rendimiento con $k = 1$ óptimo (vecino más cercano). En el dataset original logra clasificación perfecta, indicando que las clases están bien separadas espacialmente. Con el dataset balanceado, mantiene 95% de accuracy, igualando a MLE Full y Parzen. Con $k = 1$, el método simplemente asigna la clase del vecino más cercano, lo cual es óptimo cuando los datos están bien separados y tienen poco ruido. La simplicidad del método (votación directa sin estimar densidades explícitas) resulta ventajosa: menos pasos intermedios significan menos fuentes de error. Las confusiones coinciden exactamente con las de Parzen (anfibios-reptiles), confirmando que ambos métodos capturan correctamente la estructura espacial. Valores mayores de k diluirían innecesariamente la señal local.

6. Discusión y Conclusiones

6.1. Comparación general de enfoques

Modelos paramétricos: Naive Bayes perfecto (1.0/1.0), asunción de independencia no penaliza; MLE Full falla con clases minoritarias (0.71/0.46), más parámetros $\neq$ mejor con datos limitados.

Modelos no paramétricos: Parzen perfecto (1.0/1.0, h=0.1), suavizado continuo supera histogramas; k-NN Rule perfecto (1.0/1.0, k=1), simplicidad efectiva; Histogram colapsa (0.38/0.08), maldición dimensionalidad; k-NN Density mediocre (0.48/0.57), volumen local inestable.

6.2. Impacto de clases minoritarias

El dataset Zoo tiene clases con solo 3-4 ejemplos en entrenamiento. Naive Bayes maneja bien clases minoritarias estimando solo 16 parámetros/clase, mientras que MLE Full necesita estimar 136 parámetros/clase (matriz 16×16) y falla. Parzen y k-NN Rule no estiman parámetros por clase, usan similitud directa y funcionan bien si ejemplos están separados.

6.3. Métricas: Accuracy vs F1-macro

En dataset multiclase con distribución desbalanceada (40% mamíferos vs 4% anfibios), el F1-macro es esencial. Histogram Bayes tiene 38% accuracy pero solo 7.9% F1-macro, revelando que predice solo mamíferos. Los tres modelos perfectos (Naive Bayes, Parzen, k-NN Rule) tienen Accuracy = F1-macro = 1.0, confirmando clasificación genuinamente balanceada.

6.4. Conclusiones finales

En dataset original: Naive Bayes, Parzen (h=0.1) y k-NN Rule (k=1) perfectos (1.0/1.0); MLE Full falla con clases minoritarias (matrices singulares); Histogram colapsa (2^{16} bins, 101 muestras). Con balanceo: MLE Full mejora dramáticamente (0.71→0.95), validando que necesita $O(d^2)$ muestras/clase.

Lecciones: Asunción de independencia (Naive Bayes) no siempre penaliza; modelos complejos necesitan más datos; histogramas sufren curse of dimensionality, Parzen lo mitiga; k-NN simple supera k-NN density (menos pasos, menos errores). Complejidad $\neq$ mejor rendimiento.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning education by demonstrating the practical application of fundamental classification algorithms on real-world datasets. There are no negative societal consequences anticipated from this educational work.

References