Trabajo Tema 1 MED (MUM)

Beatriz Coronado Sanz

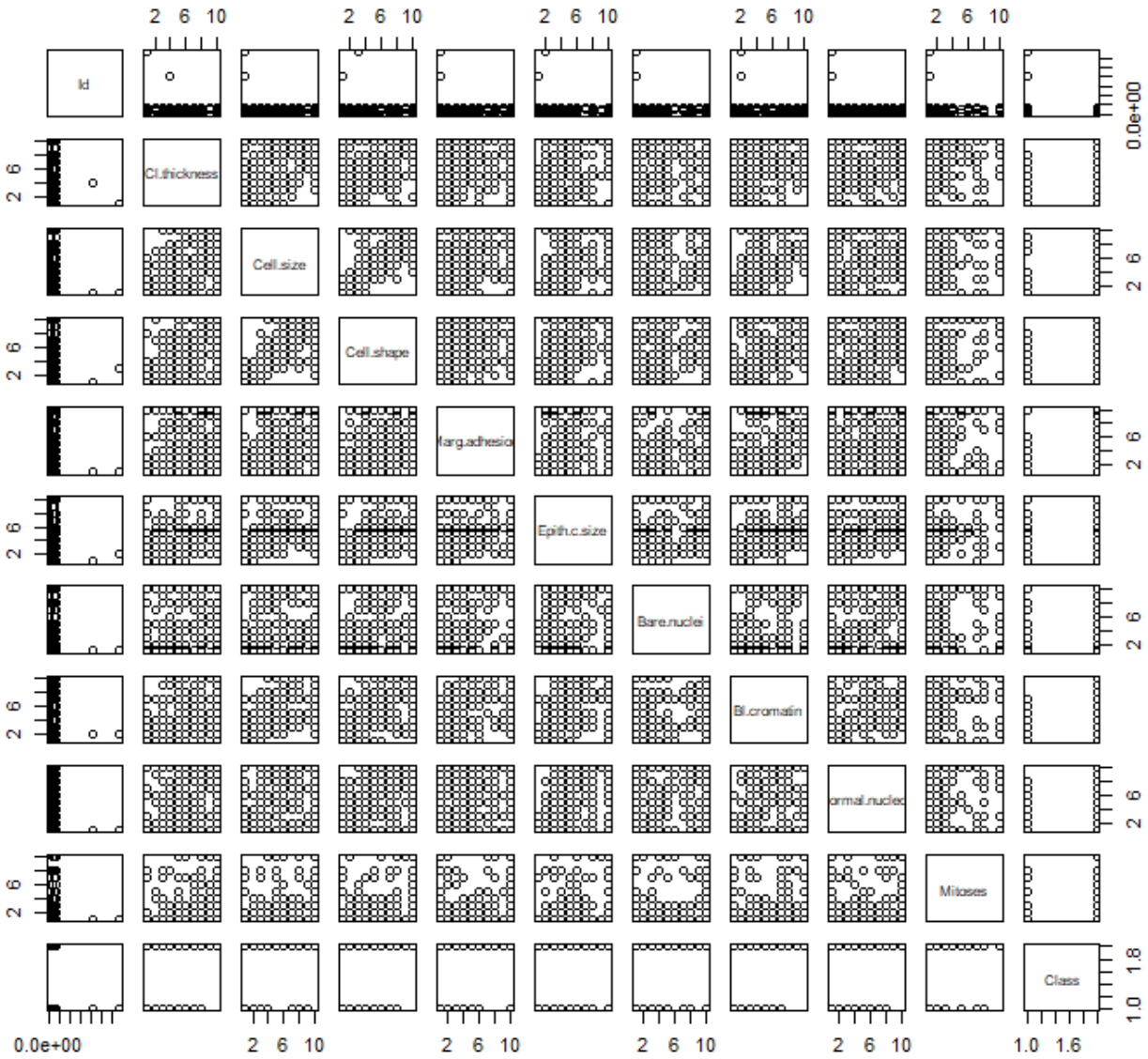
Curso 2018-2019

Índice

Datos de BreastCancer

- Conjunto de datos que contiene distintas características asociadas al tumor mamario de una serie de pacientes.
- El tumor de cada individuo se clasifica como benigno (no hay cáncer) o como maligno (hay cáncer).
- La finalidad es crear distintos modelos que puedan predecir si un tumor es benigno o maligno en función de sus características.

Representación de los datos de BreastCancer



Preprocesado

- Se divide el conjunto de datos en el conjunto de entrenamiento (70 %) y el conjunto de test (30 %).
- Se centran los datos del conjunto de entrenamiento (media nula y varianza $= 1$).
- Se aplica la transformación del conjunto de entrenamiento a los datos del conjunto de test.

Modelo y predicción para Naive Bayes

- Se construye el modelo con los datos de entrenamiento.
- Se predicen los datos del conjunto de test con el modelo.

Resultados para Naive Bayes

- Los resultados obtenidos son los siguientes:

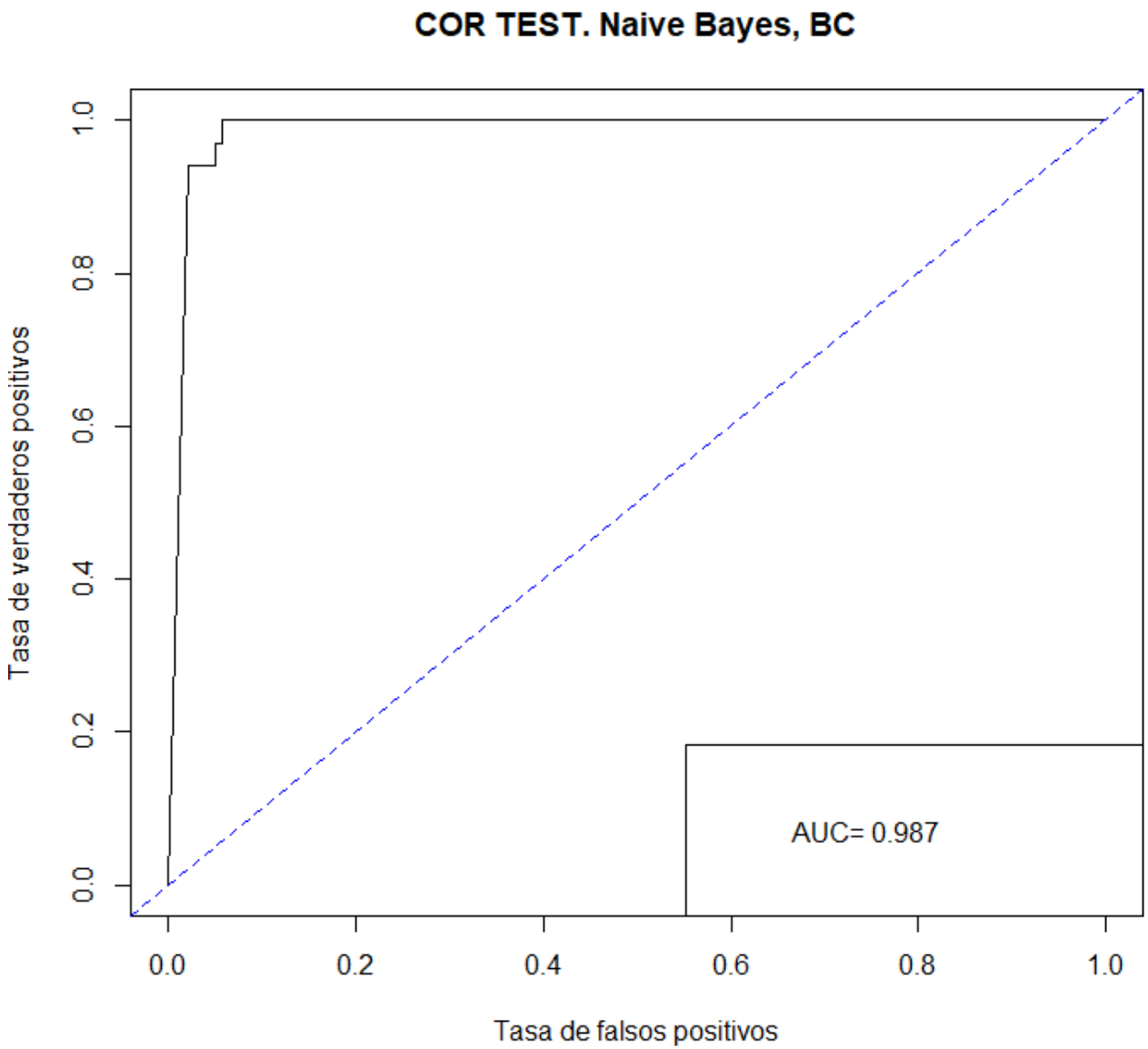
Real/Pred	Malignant	Benign
Malignant	67	0
Benign	8	130

- Tasa de acierto del test: 96.1 %
- Sensitividad del test: 100 %
- Especificidad del test: 94.2 %

Resultados para Naive Bayes

- Remarcamos que la sensitividad del test es del 100 %, es decir, que no se produce ningún falso negativo. Por lo tanto, todos los casos clasificados como benignos lo son de verdad.
- Tenemos 8 falsos positivos, es decir, que hemos clasificado 8 casos como malignos cuando en realidad son benignos.
- La tasa de acierto obtenida es muy alta, ronda el 96 %.

AUC para Naive Bayes



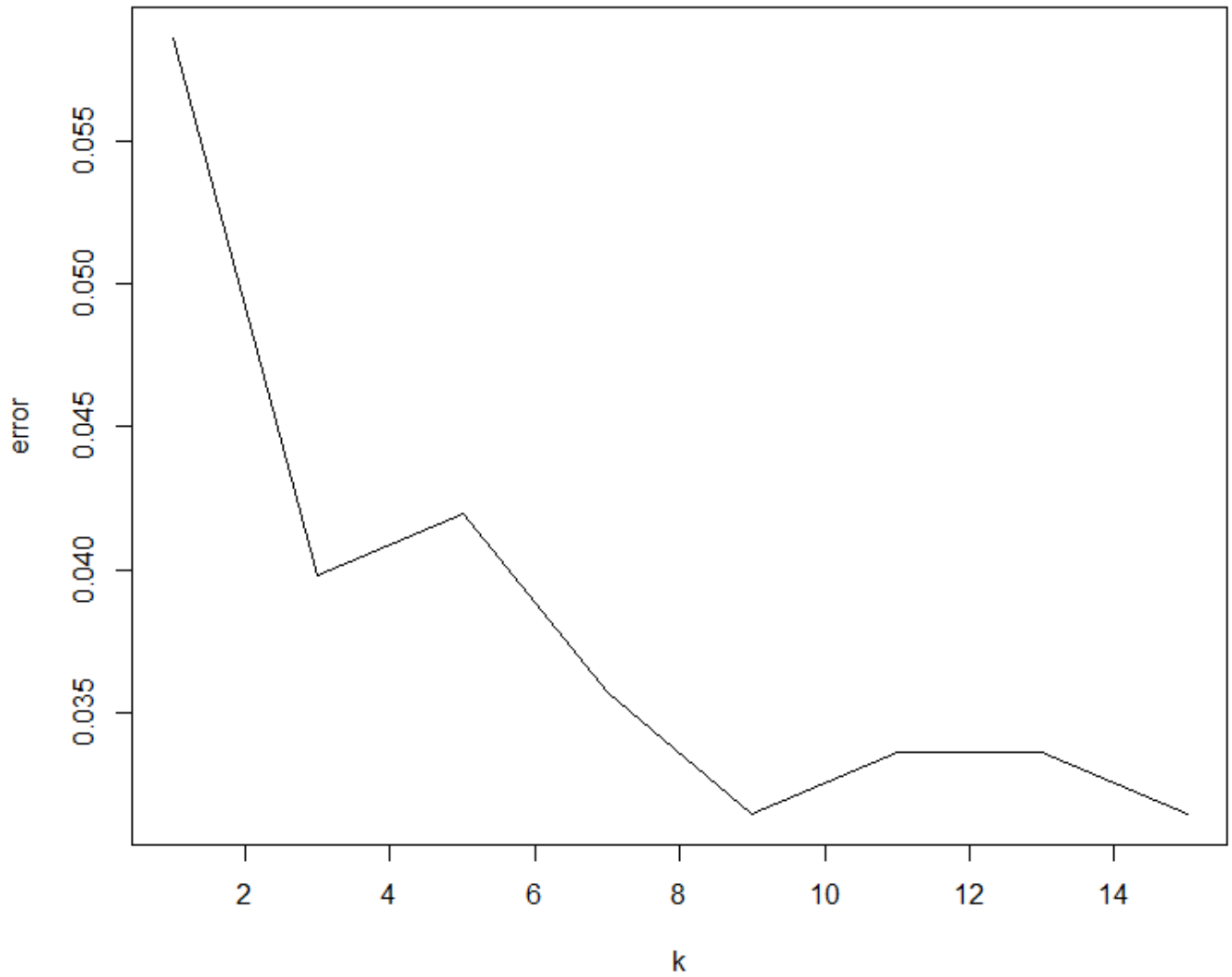
AUC para Naive Bayes

- El área bajo la curva que obtenemos se acerca mucho a 1 (vale 0.987).
- Por lo tanto, el rendimiento de este test es muy bueno.

Modelo y predicción para KNN

- Se evalúa KNN con la posibilidad de elegir un número de vecinos entre 1 y 15 para validación cruzada.
- Se construye el modelo de KNN en base al conjunto de entrenamiento y con el número de vecinos óptimo en la evaluación anterior (en nuestro caso el número de vecinos óptimo es 9).
- Se predicen los datos del conjunto de test con el modelo.

Gráfico con los errores de KNN en función del número de vecinos considerado



Resultados para KNN

- Los resultados obtenidos son los siguientes:

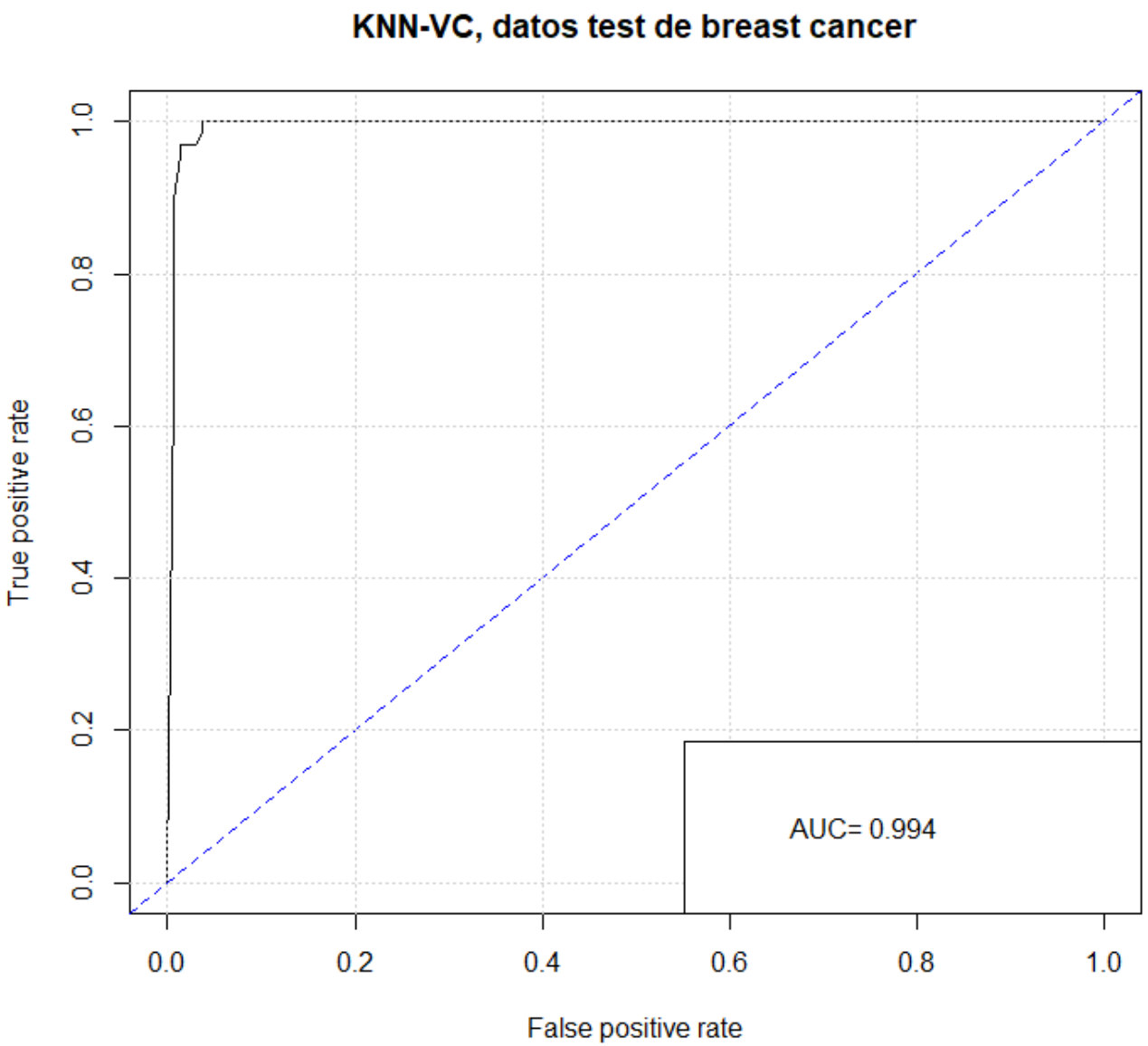
Real/Pred	Malignant	Benign
Malignant	65	2
Benign	4	134

- Tasa de acierto del test: 97.1 %
- Sensitividad del test: 97 %
- Especificidad del test: 97.1 %

Resultados para KNN

- En este caso tenemos falsos positivos y falsos negativos (4 y 2 respectivamente).
- Los valores de sensitividad y especificidad son muy altos, así como la tasa de acierto del test (todos rondan el 97 %).

AUC para KNN


KNN-VC, datos test de breast cancer
True positive rate
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0
AUC= 0.994

AUC para KNN

- El área bajo la curva que obtenemos se acerca mucho a 1 (vale 0.994).
- Por lo tanto, el rendimiento de este test es muy bueno.

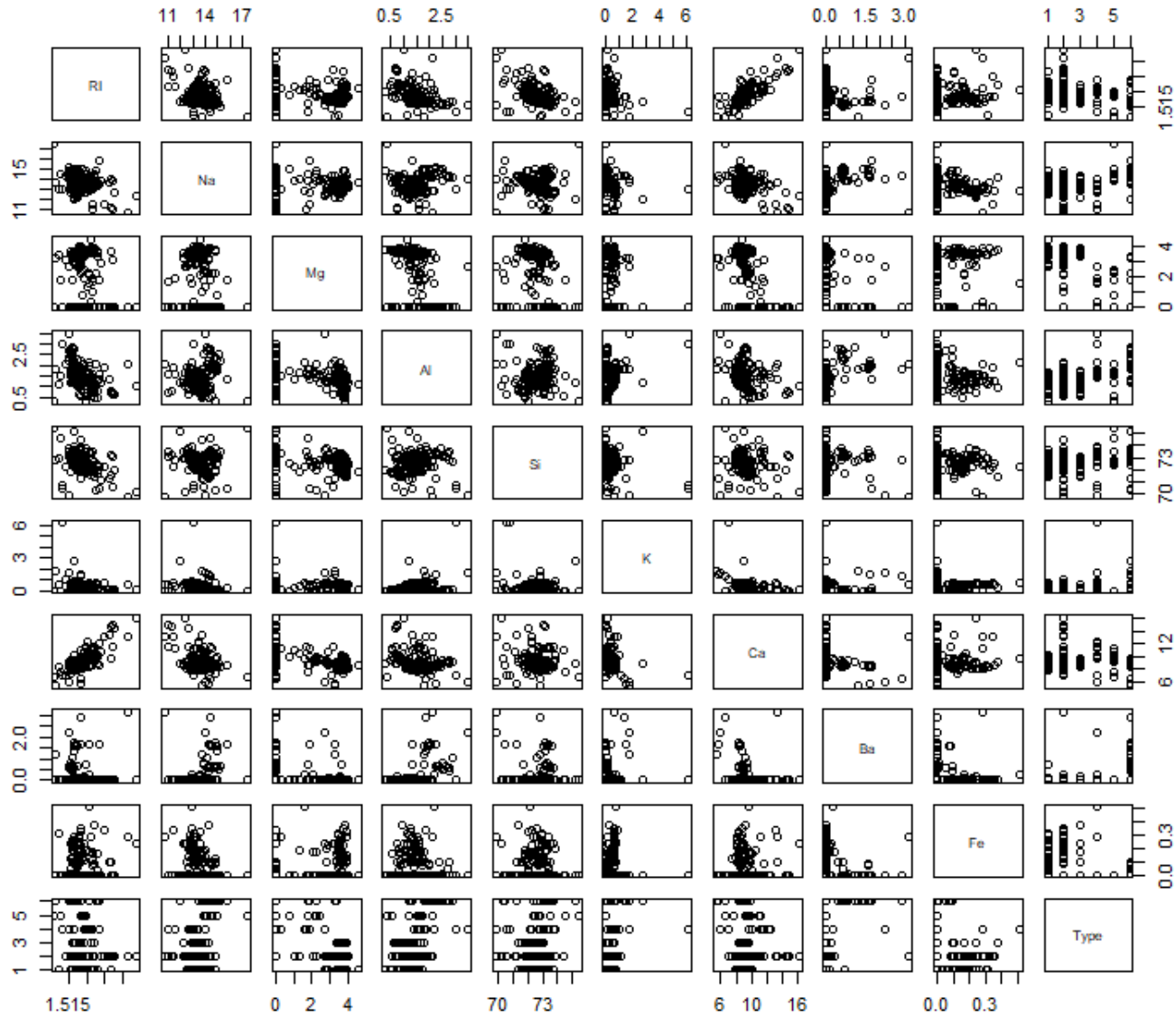
Comparación entre Naive Bayes y KNN para Breast Cancer

- Obtenemos dos modelos muy robustos con alta tasa de acierto para nuestro problema. El modelo KNN es ligeramente superior al de Naive Bayes.
- La especificicad es mejor en KNN pero la sensibilidad es mejor en Naive Bayes.
- La AUC es mejor en KNN (0.994) que en Naive Bayes (0.987), por lo que el rendimiento de KNN es mejor que el de Naive Bayes.
- Sin embargo, si tenemos en cuenta lo que representan los datos, es mejor el modelo de Naive Bayes porque nunca nos equivocamos al indicar a un paciente que tiene cáncer cuando de verdad tiene cáncer, aunque en teoría este modelo sea algo peor que KNN.

Datos de Glass

- Conjunto de datos que contiene distintas características de varios cristales.
- Los cristales se clasifican en 6 clases distintas según los elementos químicos que contienen.
- La finalidad es crear distintos modelos que puedan predecir a que clase pertenece un cristal dado.

Representación de los datos de Glass



Preprocesado

- Se divide el conjunto de datos en el conjunto de entrenamiento (70 %) y el conjunto de test (30 %).
- Se centran los datos del conjunto de entrenamiento (media nula y varianza $= 1$).
- Se aplica la transformación del conjunto de entrenamiento a los datos del conjunto de test.

Modelo y predicción para Naive Bayes

- Se construye el modelo con los datos de entrenamiento.
- Se predicen los datos del conjunto de test con el modelo.

Resultados para Naive Bayes

- Los resultados obtenidos son los siguientes:

Real/Pred	1	2	3	5	6	7
1	17	3	1	0	0	0
2	13	2	0	2	2	0
3	6	0	0	0	1	0
5	0	1	0	2	2	1
6	0	0	0	0	2	0
7	1	0	0	0	3	6

- Tasa de acierto del test: 44.62 %

Resultados por clase para Naive Bayes

- Tenemos los siguientes resultados por clases:

Clase	Precisión	Sensitividad	Especificidad
1	45.95	80.92	45.45
2	33.33	10.53	8.7
3	0	0	1.72
5	50	33.33	3.39
6	20	100	12.7
7	85.7	60	1.82

- Precisión: cuántos elementos de la clase *i* son de verdad de la clase *i*
- Sensitividad: cuántos elementos de la clase *i* hemos clasificado bien.
- Especificidad: proporción de falsos positivos de la clase *i* respecto a todos los datos sin contar los de la clase *i*

Resultados para Naive Bayes

- Clasificamos correctamente todos los elementos de la clase 6 y muchos de la clase 1, pero ninguno de la clase 3.
- Hay una alta tasa de falsos positivos para la clase 1. Esto puede indicar que hay bastantes clases para las que Naive Bayes no encuentra diferencias y las clasifica todas en una misma clase (en este caso las clasifica en la clase 1).
- Tenemos que señalar que el conjunto de test no es homogéneo pues tenemos 20 elementos para las clases 1 y 2 pero no superamos los 10 elementos para ninguna de las otras 4 clases. Esto puede justificar que la la tasa de acierto del test sea muy baja y no llegue al 50 %. Con el error bootstrap comprobaremos si el conjunto de test es representativo o no. Si lo es, quiere decir que en el conjunto de entrenamiento hay muchos más ejemplos de las clases 1 y 2 que del resto.

Error bootstrap para Naive Bayes

- Para cada muestra boostrap calculamos su modelo y su predicción para obtener su error. También predecimos el conjunto de elementos que no tomamos de la muestra (se corresponden con el conjunto test).
- Los principales errores que obtenemos son:
  - Error empírico: 51.87 %
  - Error bootstrap: 50.59 %
  - Error OOB: 60.06 %
  - Error bootstrap 0.632: 57.05 %
  - Error bootstrap 0.632+: 57,79 %

Error bootstrap para Naive Bayes

- El error empírico es el error del modelo que hemos obtenido para el conjunto de entrenamiento al clasificar todos los datos
- Observamos que el error bootstrap es parecido al error empírico, lo que quiere decir que el modelo de Naive Bayes no clasifica correctamente los datos con los que se ha creado el modelo.
- Observamos que clasifica peor los datos con los que no se ha creado el modelo.
- En este caso, debemos tener en cuenta que el modelo de Naive Bayes clasificaba mal desde el principio, sesgando las predicciones hacía la clase 1, por lo que los resultados se corresponden con los de este sesgo. Comprobamos que es irrelevante el conjunto de test que cojamos, pues el modelo de Naive Bayes clasifica siempre igual de mal para nuestro problema.

Modelo y predicción para KNN

- Se evalúa KNN con la posibilidad de elegir un número de vecinos entre 1 y 15 para validación cruzada.
- Se construye el modelo de KNN en base al conjunto de entrenamiento y con el número de vecinos óptimo en la evaluación anterior (en nuestro caso el número de vecinos óptimo es 1).
- Se predicen los datos del conjunto de test con el modelo.

Gráfico con los errores de KNN en función del número de vecinos considerado

- Los resultados obtenidos son los siguientes:

Real/Pred	1	2	3	5	6	7
1	15	4	2	0	0	0
2	2	16	0	0	1	0
3	4	0	3	0	0	0
5	0	0	0	3	0	3
6	0	0	0	0	1	1
7	0	0	1	0	0	9

- Tasa de acierto del test: 72,31 %

Resultados para KNN

- Tenemos los siguientes resultados por clases:

Clase	Precisión	Sensitividad	Especificicad
1	71,43	71,43	13,64
2	80	84,21	8,7
3	50	42,86	5,17
5	100	50	0
6	50	50	1.59
7	69,2	90	7,23

- Precisión: cuántos elementos de la clase *i* son de verdad de la clase *i*
- Sensitividad: cuántos elementos de la clase *i* hemos clasificado bien.
- Especificidad: proporción de falsos positivos de la clase *i* respecto a todos los datos sin contar los de la clase *i*

Resultados para KNN

- En este caso, la tasa de acierto del modelo es del 72 %, que es un valor aceptable.
- Observamos que ahora las clases están mejor distribuidas y que la precisión por clase no baja del 50 % Del mismo modo, la sensitividad por clase también tiene mejores resultados.
- Como dato reseñable vemos que no tenemos ningún falso positivo para la clase 5 (y, por tanto, su precisión es del 100 % y su especificidad del 0 %).

Error bootstrap para KNN

- Para cada muestra boostrap calculamos su modelo y su predicción para obtener su error. También predecimos el conjunto de elementos que no tomamos de la muestra (se corresponden con el conjunto test).
- Los principales errores que obtenemos son:
  - Error empírico: 8.41 %
  - Error bootstrap: 0 %
  - Error OOB: 30.06 %
  - Error bootstrap 0.632: 22.1 %
  - Error bootstrap 0.632+: 24.01 %

Error bootstrap para KNN

- En este caso, el error empírico es muy bajo (ronda el 8 %), por lo que nuestro modelo es bueno.
- El error bootstrap es 0, lo que quiere decir que clasifica correctamente los datos con los que se construye el modelo.
- El error OOB es del 30 %, por lo que 1 de cada 3 veces falla para un elemento nuevo (no considerado a la hora de calcular el modelo).

Comparación entre Naive Bayes y KNN para Glass

- Para la clasificación multiclase que tenemos, el modelo KNN es mucho mejor que el de Naive Bayes.
- El modelo Naive Bayes no distingue bien las clases y sesga sus predicciones hacia la clase 1, incluso con los datos del conjunto de entrenamiento. Esto puede ocurrir debido a que hay muchos más ejemplos de las dos primeras clases que del resto.
- El modelo KNN distingue razonablemente bien las clases para cualquier conjunto de test y no se equivoca para los datos con los que se ha construido el modelo.

Conclusiones finales

- Para clasificación binaria (conjunto de datos Breast Cancer) funcionan razonablemente bien tanto Naive Bayes como KNN.
- Para clasificación multiclase (conjunto de datos Glass) KNN funciona mucho mejor que Naive Bayes.
- Observamos que con el aumento de clases los métodos de clasificación funcionan peor.
- Podría ser interesante aplicar algún método de reducción de dimensión como PCA antes de aplicar Naive Bayes o KNN para ver si la clasificación resultante es mejor para el ejemplo multiclase.