Óscar Fernández Sánchez
Sergio Orejón Pérez
Jordy Ivan San Martín Ponce
Carlos Garrido Junco

1) **Describir brevemente en qué consiste el problema ampliando la información dada más arriba**

   Dada una base de datos de diagnóstico de cáncer de mama, a partir de los parámetros recogidos(radio medio,peor simetría,error de concavidad, etc), predecir un futuro diagnóstico utilizando los métodos de k-nn, árboles de decisión y con naïve Bayes

2) **Describir la muestra según un criterio razonable**

   Hemos decidido dividir los datos en un set de entrenamiento y otro de test ya que uno lo utilizaremos para entrenar nuestro modelo que será el conjunto de entrenamiento y con el de test comprobaremos la precisión del modelo.

3) **Emplear un modelo de k-nn seleccionando los hiper-parámetros óptimos mediante algún procedimiento**

   Para este caso, no es necesario cambiar el hiper parámetro ya que no se produce un sobre ajuste, sin embargo podemos seleccionarlo utilizando un bucle en el cual vamos comparando la precisión obtenida de calcular el modelo knn con los diferentes números vecinos próximos, para realizar esto aplicamos el algoritmo de validación cruzada(**cross validation)** el cual va guardando en un array las distintas precisiones del modelo knn con un hiper parametro determinado, y una vez terminado, escogemos el número de vecinos proximos que tiene el valor de precisión más alto dentro del array para asi obtener el hiper parametro mas optimo.

4) **Hacer lo mismo para un modelo de árboles**

   Para seleccionar los hiper parámetros utilizamos el concepto de entropía, que a su vez utiliza el concepto de la ganancia de información, y esta mide a su vez el grado de sorpresa de un elemento.
   Esto se puede implementar muy fácilmente desde sklearn a partir de un parámetro en el método de “DecisionTreeClassifier”,en el cual se introduce como “criterion=’entropy’”

5) **Hacer lo mismo para un modelo de naïve Bayes**

   En este caso hemos seguido los mismos pasos que en los dos ejercicios anteriores, pero a la hora de la corrección del modelo hemos probado el método de Gauss y el método de Laplace, que solucionan el error de la probabilidad cero, que puede afectar de forma negativa a la predicción. Los resultados nos han mostrado que usando el método de Gauss obtenemos un 5% más de acierto en el modelo de entrenamiento y un 3% más en el modelo de test.

6) **Comparar, con algún criterio (explicando por qué) los resultados obtenidos con los tres modelos**

Hemos comparado los tres modelos mediante la “precisión”, con la cual podemos ver que el modelo knn muestra una precisión en su fase de entrenamiento del 0,94 y en el test también 0,94, siendo así un resultado muy bueno para este modelo, mientras que en los otros dos modelos propuestos muestran una precisión menor en cuanto a relación entre los datos de entrenamiento y de test, siendo en el modelo de naive bayes una precisión de 0,94 y en los de test 92, esto se debe a que en el modelo de Bayes no conocemos con exactitud la probabilidad a priori y por eso da mas error que el de Knn, y en el árbol de clasificación es en los datos de entrenamiento vemos un 100 y en los datos de test un 70,39, esto se debe a que nos encontramos un sobreajuste, esto produce que este modelo no sea muy preciso .

7) **Plantear posibles críticas o mejoras de cada uno de los modelos o del procedimiento seguido, tratando de evitar la repetición de lo mencionado en clase**

Para knn hemos considerado utilizar la validación cruzada y además utilizar un bucle en el cual vamos comparando el número de vecinos, hasta llegar al número de vecinos que minimiza el error de la validación cruzada.

Para los árboles, una mejora a implementar sería el tema del prunning ya que observamos que en nuestro modelo nos da un 100% de precision en la fase de entrenamiento, y esto se debe hay un sobreajuste y por lo tanto habría que “podarlo” para evitar la diferencia que hay entre la precision de los datos de entrenamiento y de test.

Y por último, para el modelo de Naive Bayes, una mejora sería implementar la corrección de Laplace mediante el modelo de MultinomialNB() evitar el error de probabilidad cero.

8) **Replicar todos estos resultados, empleando obligatoriamente un código similaral utilizado en los puntos anteriores, sobre otro problema médico de características similares (clasificación binaria)**

El csv hay que cargarlo al correr el bloque de código correspondiente a esta pregunta, que pedirá importar un archivo y se le deberá pasar el archivo **heart.csv.**

Para este ejercicio hemos usado el csv de Heart Disease UCI, que proporciona datos médicos sobre pacientes y determina si el paciente tiene problemas de corazón. Se pasan en el csv 14 columnas para determinarlo. Para diferenciar este ejercicio del resto de la práctica, al método de vecinos próximos le hemos aplicado un algoritmo de validación cruzada para arreglar los problemas del sobreajuste y que

no aparezcan outliers en el modelo, ya que al ser un modelo no testado cogido de internet pueden aparecer valores que contaminen la muestra de datos que escogemos. Y para el árbol de selección, nos hemos planteado ‘’podar’’ el árbol, para eliminar nodos que tuviesen un acierto similar y de esta forma simplificar el algoritmo. Aplicando los mismos métodos que en los ejercicios anteriores hemos llegado a los siguientes resultados:

Modelo Knn antes de reparar el sobreajuste:
Precision set entrenamiento: 0.74
Precision set de test: 0.68
El numero optimo de vecinos es: 23
Rehacer el modelo KNN con el hiperparametro optimizado
Precision set de entrenamiento: 0.71
Precision set de test:: 0.67

Arbol de clasificacion
Porcentaje de predicciones correctas (training): 100.0
Porcentaje de predicciones correctas (test): 72.36842105263158

Bayes
Porcentaje de predicciones correctas (training): 87.19008264462809
Porcentaje de predicciones correctas (test): 85.24590163934425934425