



Universidad Simón Bolívar

Departamento de Computación y Tecnología de la información

Arturo Toro 12-10796

Proyecto - Heart Disease (Cleveland)

Resumen

Con los datos Heart Disease Data Set se construyen dos redes que permiten clasificar los datos entre aquellos en los cuales hay enfermedades del corazón y en los que no. Para ello, se utiliza la data procesada disponible en el repositorio y se realizan pruebas con una MLP (Perceptrón multicapas) y una SVM (Máquina de vector de soportes), siendo la primera la que mejores resultados obtuvo con un 100% de precisión sobre el conjunto de validación para los mejores parámetros encontrados en el caso de múltiples clases.

Preprocesamiento

Antes de seleccionar los clasificadores a utilizar y realizar la entonación de los parámetros, se obtuvo información relacionada con la naturaleza de los datos:

- En primer lugar, el dataset cuenta con 303 ejemplos, de los cuales 7 presentan datos faltantes.
- Aunque en el repositorio especifican que la investigación se ha centrado en la construcción de clasificadores binarios (Separar presencia de ausencia de la enfermedad), debido a que se existen 5 clases, se decidió construir y entonar clasificadores para ambos casos.
- Para el caso del clasificador binario, puede decirse que la proporción entre las clases es balanceada debido a que 54.1% de los datos corresponden a “Ausencia de la enfermedad” mientras que un 45.9% corresponde a “Presencia de la enfermedad”
- Los datos de los atributos no están normalizados, lo cual podría afectar negativamente la convergencia de los algoritmos

A partir de la información obtenida se tomaron las siguientes decisiones:

1. Datos faltantes: En el caso de los 7 ejemplos del conjunto de datos que contenían datos faltantes, se decidió completar esos valores utilizando el promedio de la columna en la que se encuentra dicho valor. De esta forma, es posible trabajar con esos ejemplos pero el valor del atributo no influirá de forma significativa en el entrenamiento de la red. Cabe destacar que para utilizar las funciones de la librería seleccionada que permiten realizar esto de forma rápida, se sustituyeron todos los caracteres '?' del conjunto de datos por 'NaN' mediante un editor de texto.
2. Normalización de los datos: Para normalizar los datos se utiliza un "escalamiento min-max" sobre cada una de las columnas, dejando los datos en un intervalo de 0 a 1 mediante la expresión:

$$X_{normal} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Clasificadores utilizados

Debido a que uno de los intereses del proyecto es construir un clasificador multiclases se descartó la opción de utilizar un perceptrón para realizar la tarea. En su lugar, se decidió realizar experimentos con una MLP y con una SVM, debido a la capacidad que demostraron para clasificar datos de distintas dimensiones a lo largo del curso. Las decisiones tomadas para la implementación y entonación de cada una de las redes seleccionadas fueron:

MLP: La red utiliza Backpropagation como algoritmo de entrenamiento y se establece un límite de 3000 épocas para el entrenamiento. Se realizaron pruebas con 1 y 2 capas para el caso del clasificador binario, mientras que para el caso de múltiples clases se realizaron pruebas con 1, 2 y 3 capas. En la sección de Experimentos y Resultados se proporcionan detalles relacionados con el número de neuronas en cada capa. Adicionalmente, los valores de las tasas de aprendizajes utilizados fueron 0.001, 0.005, 0.01 y 0.05.

En cuanto a las funciones de activación, se utiliza softmax para la capa de salida, mientras que para las capas ocultas se experimentó con relu y tangente hiperbólica

(Se utiliza la misma función de transferencia en todas las neuronas de la capa oculta).

- Tangente hiperbólica ($\tanh(x)$): $f(x) = \tanh(x)$
- Relu: $f(x) = \max(0, x)$

Por último, todas las pruebas se realizaron con un momentum de 0.5 para intentar que el clasificador no se quedara atrapado en mínimos locales. Esta decisión se produjo debido a que en las pruebas iniciales (sin momentum), la precisión de la red era baja para tasas de aprendizaje pequeñas (alrededor de 50%).

SVM: Se realizan pruebas con kernel sigmoide y rbf, mientras que el coeficiente (gamma) del kernel es 1/13 (13 atributos).

Experimentos y Resultados

Para realizar los experimentos, se utilizó la técnica de validación cruzada (K-Fold Cross validation) con una variante de estratificación que busca mantener la proporción entre las clases para todas las particiones generadas. En este caso, se utilizó 10-fold cross validation, lo cual produce particiones con 28,29,31,32 y 33 para el conjunto de validación. Esta diferencia de tamaño en cada una de las particiones se debe a la estratificación que busca mantener la proporción entre clases. El resto de los ejemplos van al conjunto de entrenamiento.

Es importante destacar que los datos son desordenados antes de comenzar los experimentos para evitar que la actualización de los pesos se vea afectada por el orden original.

En el caso de la MLP, se presenta una tabla para las pruebas más relevantes en donde:

- $F(x)$: Función de transferencia en la capa oculta.
- # Capas: Número de capas ocultas
- # Neuronas: Número de neuronas por capa, de izquierda a derecha separado por comas (,).
- tasa apr. : Tasa de aprendizaje
- P. Entrena: Precisión sobre el conjunto de entrenamiento (%)
- P. Valida: Precisión sobre el conjunto de validación (%)

Para la SVM se reporta:

- kernel: Kernel utilizado durante la prueba
- P. Entrena: Precisión sobre el conjunto de entrenamiento (%)
- P. Valida: Precisión sobre el conjunto de validación (%)

Es importante destacar que en este informe sólo se mostrarán algunos de los resultados considerados como los mas relevantes para justificar las decisiones tomadas e ilustrar el desempeño de las redes. Los resultados completos están disponibles en el repositorio de git con el mismo formato que se muestra en las tablas presentadas a continuación.

El mejor resultado se resalta en azul en cada tabla. En caso de empate, se selecciona la red menos compleja (Menor cantidad de neuronas/capas).

Clasificador binario

En este caso, sólo se clasifica la presencia o ausencia de enfermedades en el corazón, para ello todos los valores esperados que representan enfermedades del corazón (Del 1 al 4), se mapean a 1, mientras que la ausencia de enfermedades se deja con el valor 0. Los resultados obtenidos fueron los siguientes:

MLP (Sin momento)					
F(x)	# Capas	# Neuronas	tasa apr.	P. Entrena.	P. Valida.
tanh	2	(4, 3)	0.001	50.2	49.45
tanh	2	(4, 3)	0.05	98.79	97.03
tanh	2	(4, 3)	0.01	100.0	99.33
tanh	2	(4, 3)	0.05	100.0	100.0
relu	2	(3, 2)	0.01	54.13	54.12
relu	2	(3, 2)	0.05	92.89	91.41
relu	2	(3, 2)	0.01	100.0	100.0
relu	2	(3, 2)	0.05	100.0	100.0

Tabla 1: Resultados MLP sin momento. Caso binario

Aunque se tuvo éxito utilizando una MLP sin momentum como clasificador binario, se presentan los resultados utilizando momentum:

MLP (Con momento)					
F(x)	# Capas	# Neuronas	tasa apr.	P. Entrena.	P. Valida.
tanh	2	(3, 2)	0.001	54.13	54.44
tanh	2	(3, 2)	0.005	100.0	99.67
tanh	2	(3, 2)	0.01	100.0	100.0
tanh	2	(3, 2)	0.05	100.0	100.0
relu	2	(5, 2)	0.001	88.03	86.77
relu	2	(5, 2)	0.005	100.0	99.0
relu	2	(5, 2)	0.01	100.0	100.0
relu	2	(5, 2)	0.05	100.0	100.0

Tabla 2: Resultados MLP con momento. Caso binario

Es importante destacar que, aunque ambas variantes logran clasificar todos los datos correctamente, al agregar momentum se logra que más configuraciones de la red alcancen el óptimo, lo cual indica que se agrega flexibilidad a la red utilizando esta técnica.

SVM		
Kernel	P. Entrena.	P. Valida.
sigmoide	89.88	88.4
rbf	91.57	91.05

Tabla 3: Resultados SVM. Caso binario

Como se puede apreciar, los resultados utilizando una SVM son mucho peores que los mejores resultados obtenidos utilizando una MLP, por lo cual es posible que la elección del kernel no sea la adecuada.

Clasificador con múltiples clases

En este caso, se clasifican 4 tipos de enfermedades del corazón y la ausencia de ellas. Algunos de los resultados obtenidos fueron los siguientes:

MLP (Sin momentum)					
F(x)	# Capas	# Neuronas	tasa apr.	P. Entrena.	P. Valida.
tanh	2	(5, 3)	0.001	53.80	54.0
tanh	2	(5, 3)	0.05	100.0	99.7
tanh	1	-3	0.001	54.49	54.65
tanh	1	-3	0.005	73.49	72.68
relu	2	(5, 3)	0.01	94.5	94.48
relu	2	(5, 3)	0.05	96.52	95.81
relu	2	(3, 2)	0.001	54.13	54.30
relu	2	(3, 2)	0.005	66.71	65.61

Tabla 4: Resultados MLP sin momento. Múltiples clases

Notar que para valores “altos” de la tasa de aprendizaje, se obtienen mejores resultados, mientras que para valores más bajos el algoritmo pareciera quedar atrapado en mínimos locales. Este comportamiento se repite para las distintas configuraciones seleccionadas en las capas y es lo que motivó a utilizar momentum en los experimentos posteriores.

MLP (Con momentum)					
F(x)	# Capas	# Neuronas	tasa apr.	P. Entrena.	P. Valida.
tanh	2	(5, 3)	0.001	59.20	58.91
tanh	2	(5, 3)	0.005	98.46	98.02
tanh	2	(5, 3)	0.01	99.82	99.35
tanh	2	(5, 3)	0.05	100.0	100.0
relu	2	(5, 3)	0.005	94.43	94.48
relu	1	-4	0.05	100.0	99.64
relu	2	(3, 2)	0.001	54.13	54.30
relu	2	(3, 2)	0.005	82.62	80.61

Tabla 5: Resultados MLP con momento. Múltiples clases

Con estos ejemplos puede notarse que, aunque para una tasa de aprendizaje de 0.001 el desempeño de la red aún es deficiente, presenta una mejoría con respecto al modelo que no incluía el momentum.

SVM		
Kernel	P. Entrena.	P. Valida.
sigmoide	56.87	56.92
rbf	69.54	66.31

Tabla 6: Resultados SVM. Múltiples clases

Como se puede apreciar, los resultados utilizando una SVM son mucho peores que los mejores resultados obtenidos utilizando una MLP, por lo cual es posible que la elección del kernel no sea la adecuada.

Detalles del mejor caso

A continuación se muestran los mejores resultados obtenidos sobre el conjunto de validación para cada caso con cada uno de los clasificadores utilizados:

	Comparación de clasificadores (Mejor caso)	
Problema	Precisión MLP (%)	Precisión SVM (%)
Binario	100	91.05
5 Clases	100	66.31

Tabla 1: Comparación de resultados (Conjunto de validación)

Se considera que los mejores resultados obtenidos para el caso de un clasificador multiclases se logran con una red MLP con 2 capas ocultas con (5,3) neuronas respectivamente y función de activación tangente hiperbólica, con una precisión de 100% sobre el conjunto de entrenamiento y 100% para el conjunto de validación. Cabe destacar que para lograr clasificar correctamente todos los ejemplos se utilizó momentum de 0.5 y una tasa de aprendizaje de 0.05. En general, se obtuvieron mejores resultados para las tasas de aprendizaje altas.

Por otro lado, al agregar momento en el caso de un clasificador binario es posible clasificar correctamente todos los datos utilizando múltiples configuraciones para una red MLP, mientras que también fue posible clasificar correctamente todos los datos sin utilizar momento. Una de las configuraciones exitosas consta de 2 capas ocultas con (3,2) neuronas respectivamente, con una precisión de 100% sobre el conjunto de entrenamiento y 100% para el conjunto de validación.

Así pues, se considera que una MLP permite resolver el problema de forma satisfactoria. Utilizar el momentum parece aportar flexibilidad a la red y le permite alcanzar mejores resultados.