Introduccion

Se estudiara la base de datos de Cardiopatia de Cleveland, veremos como seria una mejor entonacion para varios modelos de clasificacion que usan redes neuronales.

Consideraciones:

Tener en cuenta que el dataset que se esta usando (Processed Cleveland Database) es pequeño, poco mas de 300 muestras, aun cuando hay varios dataset de otras ciudades, estas estan con muchos campos faltantes, por lo que seria perjudicial para el entrenamiento. Por otro lado tenemos que el dataset original tiene 76 atributos, pero todos los experiementos relacionados solo utilizan 14 de ellos que son los datos ya proporcionados en el Dataset que estamos usando, ademas de que la database de Cleveland es la unica que ha sido usada en investigaciones en el campo de Inteligencia Artificial.[2]

Vamos a fijar ciertos parametros, como las iteraciones maximas a 3000, mas de aqui consideramos sobreentrenar la red; La cantidad de neuronas por capa oculta a 20, se podria usar una neurona por parametro de entrada, que serian 13 neuronas pero agregamos 7 mas (un tercio mas aproximadamente) para tener un poco mas de holgura al momento de clasificar; 5 neuronas en la capa de salida, que representan los 5 estados posibles en la respuesta deseada; Los pesos sinapticos se inicializan de forma aleatoria.

Obetenmos los datos de prueba del dataset, ya que no disponemos de mas data que se ajuste a las data ingresada.

Usamos un metodo de aprendizaje para el caso del perceptron multicapa, llamado "Adam", el cual es una optimizacion para el descenso del gradiente estocastico.[1]

Esquema del DataSet

1. age: age in years
    - max value: 77
    - min value: 29
2. sex: sex (1 = male; 0 = female)
    - Value 0: famale
    - Value 1: male
3. cp: chest pain type
    - Value 1: typical angina
    - Value 2: atypical angina
    - Value 3: non-anginal pain
    - Value 4: asymptomatic
4. trestbps: resting blood pressure (in mm Hg on admission to the hospital)
    - max value: 200.0
    - min value: 94.0
5. chol: serum cholestoral in mg/dl
    - max value: 564.0
    - min value: 126.0
6. fbs: (fasting blood sugar > 120 mg/dl)
    - Value 0: false
    - Value 1: true
7. restecg: resting electrocardiographic results
    - Value 0: normal
    - Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV)
    - Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria
8. thalach: maximum heart rate achieved
    - max value: 202.0
    - min value: 71.0
9. exang: exercise induced angina
    - Value 0: no
    - Value 1: yes
10. oldpeak = ST depression induced by exercise relative to rest
    - max value: 6.2
    - min value: 0.0
11. slope: the slope of the peak exercise ST segment
    - Value 1: upsloping
    - Value 2: flat
    - Value 3: downsloping
12. ca: number of major vessels (0-3) colored by flourosopy
13. thal: 3 = normal; 6 = fixed defect; 7 = reversable defect
    - Value 3: normal
    - Value 6: fixed defect
    - Value 7: reversable defect
14. num: diagnosis of heart disease (angiographic disease status)
    - Value 0: < 50% diameter narrowing
    - Value 1: > 50% diameter narrowing

Pruebas

3 pruebas globales, cada una con un metodo de preprocesamiento de datos.

Estructura de las pruebas: 1. Sin procesar los datos de entrada. 1. Perceptron Multicapa 1. Metodo de entrenamiento: Estocastico 2. Metodo de entrenamiento: Adam (*) 2. Maquina de vectores de soporte

2. Escalando los datos entre 0 y 1 dividiendo entre el max de cada campo.

    1. Perceptron Multicapa
        1. Metodo de entrenamiento: Estocastico
        2. Metodo de entrenamiento: Adam (*)
    2. Maquina de vectores de soporte

3. Escalando los datos entre 0 y 1 usando la tecnica del MinMax de cada campo.

1. Perceptron Multicapa
    1. Metodo de entrenamiento: Estocastico
    2. Metodo de entrenamiento: Adam (*)
2. Maquina de vectores de soporte

Cada prueba consiste en entrenar con el dataset completo(Usando el metodo "fit"). luego verificar que tan bueno fue el entrenamiento con una porcion del dataset (Usando el metodo "score").

1.1.1. Datos sin procesar, MLP, SGD.

Para la entonacion inicial, elegimos unos parametros estandar, por ejemplo: - neuronas en la capa oculta=20. - Funcion de Activacion = Logistica. - Metodo de Aprendizaje = Descenso del Gradiente Estocastico - Tasa de aprendizaje = 0.01 - Tasa de momentum = 0.9

Para esta configuracion dio una mala clasificacion, al rededor de 39% de los datos bien clasificados.

Agregando otra capa oculta parece dar peores resultados, al rededor de 35% de los datos bien clasificados, por lo que no probaremos mas con este parametro.

Volvemos con la configuracion inicial pero bajandole la tasa de aprendizaje, obtenemos resultados parecidos.

Probamos bajandole el tasa de momentum a 0.1 y nos da una aproximacion mejor, al rededor del 55% de los datos bien clasificados. Pero si lo seguimos bajando, parece que ya baja el porcentaje de datos bien clasificados.

Tambien probamos con otra funcion de activacion, en este caso con la ReLU, ya que al no tener un preprocesamiento de datos puede funcionar mejor. Obtenemos resultados similares, al rededor de 55% de los datos bien clasificados, sin embargo, no podemos considerar esto una buena clasificacion.

1.1.2 Datos sin procesar, MLP, Adam

Probamos con otro metodo de aprendizaje (Adam), y obtenemos mejores resultados con la configuracion inicial, al rededor de 49% de los datos bien clasificados, nuevamente, no es suficiente. Bajando la tasa de aprendizaje a 0.001 obtenemos mejores resultados, ademas de tambien bajar el tasa de momentum a 0.1 y con funcion de activacion ReLU obteniendo mejores resultados, al rededor de 52% de los datos bien clasificados, sin embargo, con esta misma configuracion se obtuvo mejores resultados con el metodo de aprendizaje Descenso del Gradiente Estocastico.

1.2.1 Datos sin procesar, SVM, Gaussiano.

2.1.1 Procesamiento MaxAbs, MLP, SGD

Para la entonacion inicial probamos los mismos parametros que la en la prueba pasada. Notamos algo inusual y es que aun cuando variamos la cantidad de capas ocultas, la tasa de aprendizaje y la tasa de momentum tenemos el mimso resultado para cada caso, que representa al rededor de 55% de los datos bien clasificados. Probando bajar la tasa de aprendizaje tenemos que ni se llega al indice de tolerancia preestablecido por el Clasificador, de igual forma obtenemos un resultado similar.

Probando con otra funcion de activacion, en este caso la tangente hiperbolica(tanh), ya que si tenemos los datos escalados, si que tenemos resultados diferentes, pero no mejores, por lo que podemos decir que funciona mejor la funcion de activacion logistica.

2.1.2 Procesamiento MaxAbs, MLP, Adam

Para este caso si que vemos nuevos resultados, al rededor de 53% de los datos bien clasificados, con la funcion de activacion logistica, tasa de aprendizaje en 0.001 y tasa de momentum en 0.1. Pero probando con los mismo parametros pero con funcion de activacion tanh, obetemos un 60% de los datos bien clasificados, al parecer esta optimizacion resulta mas favorable con esta funcion de activacion.

3.1.1 Procesamiento MinMax, MLP, SGD

Nuevamente, para la entonacion inicial probamos los mismos parametros que la en la prueba pasada. Para esta nueva prueba obtenemos resultados mas interesantes, para los parametros iniciales si que tenemos una aproximacion del 51%, ahora, aumentando la cantidad de capas ocultas tenemos un resultado similar a varias de las pruebas pasadas, estamos hablando de un 55% de los datos bien clasificados, asi seguimos obteniendo resultados similares variando la tasa de aprendizaje y la tasa de momentum.

Aqui es que viene lo diferente, probando con la funcion de activacion tanh si que tenemos unos muy buenos resultados, un 93% de los datos bien clasificados, aun cuando no se haya alcanzado el indice de tolerancia, nuevamente preestablecido por el Clasificador.

3.1.2 Procesamiento MinMax, MLP, Adam

Cambiamos al metodo optimizado Adam, manteniendo los parametros de la prueba anterior y si que podemos ver una mejora, donde clasifica bien el 98% de los datos. Ahora, probando con la funcion de activacion logistica, vemos que nos da una clasificacion del 100%, aun cuando no se haya alcanzado el indice de tolerancia.

Conclusiones

Lo primero, que podemos notas es que la preparacion de los datos de entrada es muy importante, tener datos escalados parece ser muy determinante a la hora de entrenar una red multicapa, vemos una mejora en la clasificacion a medida que procesamos mas los datos de entrada.

Lo segundo, pareciera que con solo una capa oculta bastaba, sin embargo los mejores resultados fueron con dos capas ocultas de 20 neuronas cada una, con la entonacion correcta de los demas parametros, el numero de capas ocultas agarra importancio en el clasificador.

Tercero, el metodo de entrenamiento Adam, en general, si dio mejores resultado que el Descenso del Gradiente Estocastico.

Por ultimo, los datos no escalados se estimaba que la funcion de activacion ReLU iba a ser mas efectiva, sin embargo, no fue asi. Para los datos que estan escalados entre 0 y 1, se estimaba que la mejor funcion de activacion iba a ser la logistica, en este casi si se obtuvo mejores resultados que con la tangente hiperbolica.

Referencias:

1. Kingma, Diederik, and Jimmy Ba. “Adam: A method for stochastic optimization.”
2. Gennari, J.H., Langley, P, & Fisher, D. (1989). Models of incremental concept formation.