Universidad Simón Bolívar

Departamento de Computación y Tecnología de la Información

CI5438 – Inteligencia Artificial II

Enero-Marzo 2017

Integrantes:

Leonardo Martínez Carné: 11-10567

Nicolás Mañán Carné: 06-39883

Joel Rivas Carné: 11-10866

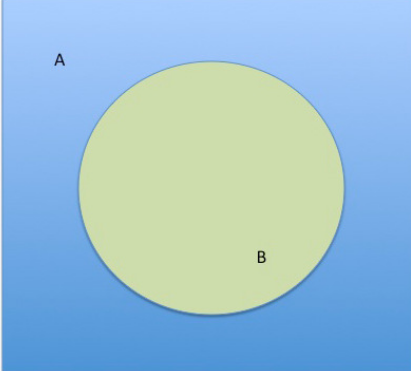
**Proyecto 2 – Redes Neuronales**

**Resumen**

En el siguiente proyecto se realizaran 3 ejercicios relacionados sobre redes neuronales, siendo este tema una base importante para el desarrollo de IA (Inteliencia Artificial).

Ejercicio 1: Se busca construir un algoritmo que implemente backpropagation en una red multicapa feedforward.

Ejercicio 2: Se realizara una serie de pruebas a la red neuronal construida anteriormente. El experimento trata sobre la clasificación de patrones, en donde se pretende lograr una clasificación de puntos en un plano perteneciente a dos regiones predeterminadas (A y B). Estas dos regiones corresponden a figuras geométricas, el área de A es el cuadrado menos el área que ocupa B, siendo el área de B limitada por una circunferencia centrada en (10,10) con radio 6, incrustada al interior del área de A. Como se muestra en el siguiente dibujo:

 Cuya ecuación canónica de esta circunferencia es:

(x − 10) 2 + (y − 10) 2 = 36

Se entiende por patrón un punto (x,y) dentro del rectángulo

y el área al que este pertenece (A o B). Se suministraron tres

conjuntos de entrenamiento de 500,100 y 2000 patrones ya

clasificados, en donde se entrenara la red. Adicionalmente, se

generara otro trío de conjuntos de datos de igual tamaño.

Se tomaron un conjunto de prueba de 10201 puntos de un barrido completo de la región cuadrada correctamente etiquetados. Se evaluaran, mostraran y analizaran las configuraciones en base a: error en entrenamiento, error en prueba, falsos positivos y falsos negativos.

Ejercicio 3: Se entrenara la red neuronal para construir dos clasificadores sobre los datos del conjunto *Iris Data Set* (<http://archive.ics.uci.edu/ml/datasets/Iris>). Uno que separe las “Iris Setosa” del resto (Clasificador Binario) y el otro que separa cada una de las clases. Estos conjuntos de datos se probaran con redes de 4 a 10 neuronas en la capa intermedia, usando como conjunto de entrenamiento los siguientes porcentajes de los datos para cada uno de ellos: 50%,60%,70%,80%,90%.

**Detalles de implementación/experimentación**

Ejercicio 1: Se uso una lista de lista, donde cada lista interna representa una capa de la red neuronal y cada capa contiene a su vez una lista de diccionarios que representan las neuronas.

Ejercicio 2: Antes que nada, se hizo un pre procesamiento de los datos, aplicando el método de normalización Min-Max. Posteriormente, se ejecuto la red neuronal con un alpha de 0.1 y 20000 iteraciones para la convergencia.

Por otra parte, se creo un script de python3 para generar los 10201 puntos para el conjunto de prueba. Igualmente, otro script para generar el trío adicinal de conjuntos de entrenamiento de 500,1000 y 2000.

Ejercicio 3: En principio, se aplico el método de normalización Min-Max y se transformaron los valores nominales a valores numéricos. Luego, se implemento un script en python3 para crear los dos conjuntos: Uno que separé los “Iris Setosa” del resto (Clasificador binario) y uno que separe cada una de las 3 clases. Y por ultimo, se seleccionaron los conjuntos de datos entrenamiento de 50%,60%,70%,80% y 90% manualmente.

**Presentación y discusión de los resultados**

**Ejercicio 2:**

Conjunto de entrenamiento 1, con N=500 y ALPHA = 0.1**.**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nº Neuronas** | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Costo Mínimo** | 121.4593 | 10.445 | 3.25517 | 3.54716 | 0.95166 | 2.05435 | 1.01744 | 0.85874 | 0.53607 |
| **Costo Máximo** | 202.9669 | 197.88 | 240.082 | 245.900 | 247.079 | 218.338 | 232.31 | 241.218 | 241.549 |
| **Error de Prueba (MSE)** | 1238.5 | 137.5 | 71.5 | 89.0 | 85.5 | 93.5 | 92.0 | 92.0 | 90.0 |
| **Falsos Positivos** | 1684 | 87 | 79 | 130 | 102 | 114 | 107 | 115 | 99 |
| **Falsos Negativos** | 793 | 188 | 64 | 48 | 69 | 73 | 77 | 69 | 81 |
| **% Total de Predicciones Falsas** | 23 % | 2,696% | 0% | 1% | 0% | 1% | 1% | 1% | 0% |

**Ejercicio 3:**

Primer conjunto de datos, con EPOCHS = 500 ALPHA = 0.1

Conjunto de entrenamiento de 50%

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Nº Neuronas** | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Costo Minimo** | 0.031647 | 0.032323 | 0.026179 | 0.033217 | 0.026074 | 0.028804 | 0.023701 |
| **Costo Maximo** | 23.529952 | 32.139896 | 29.403197 | 23.848601 | 32.337585 | 35.362520 | 36.414706 |
| **Error de Prueba (MSE)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Positivos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Negativos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **% Total de Predicciones Falsas** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Conjunto de entrenamiento de 60%

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Nº Neuronas** | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Costo Minimo** | 0.035502 | 0.035601 | 0.030731 | 0.036882 | 0.030367 | 0.032104 | 0.028115 |
| **Costo Maximo** | 27.772692 | 37.980620 | 34.435932 | 28.472308 | 38.095737 | 42.209713 | 43.629493 |
| **Error de Prueba (MSE)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Positivos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Negativos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **% Total de Predicciones Falsas** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Conjunto de entrenamiento de 70%

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Nº Neuronas** | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Costo Minimo** | 0.029401 | 0.030713 | 0.024606 | 0.031029 | 0.024495 | 0.027130 | 0.022214 |
| **Costo Maximo** | 31.542141 | 43.570894 | 38.866699 | 32.302637 | 43.626981 | 49.022311 | 50.837242 |
| **Error de Prueba (MSE)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Positivos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Negativos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **% Total de Predicciones Falsas** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Conjunto de entrenamiento de 80%

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Nº Neuronas** | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Costo Minimo** | 0.031215 | 0.032218 | 0.027113 | 0.032654 | 0.026743 | 0.028823 | 0.024852 |
| **Costo Maximo** | 35.302040 | 48.631755 | 42.955022 | 36.292927 | 48.427737 | 55.691535 | 57.977840 |
| **Error de Prueba (MSE)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Positivos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Negativos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **% Total de Predicciones Falsas** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Conjunto de entrenamiento de 90%

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Nº Neuronas** | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Costo Minimo** | 0.031715 | 0.032685 | 0.027884 | 0.033032 | 0.027475 | 0.029469 | 0.025691 |
| **Costo Maximo** | 38.087858 | 53.055956 | 45.690604 | 38.93293 | 52.094009 | 62.411200 | 65.069206 |
| **Error de Prueba (MSE)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Positivos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Falsos Negativos** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **% Total de Predicciones Falsas** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Segundo conjunto de datos,

**Conclusiones**