Tarea 1: MLP (Parte teórica)

Entrega: Jueves 22 de Agosto, 23:59 (Gradescope)

Profesor:	Pablo Estévez V.
Auxiliar:	Ignacio Reyes J.
Semestre:	Primavera 2019

1. Pregunta 1

1. ¿Cuál es la ventaja del perceptrón multicapa respecto a un modelo de una sola capa? ¿Qué diferencias tienen las fronteras de decisión de dichos modelos?

   Sol: Un perceptrón de una sola capa tiene el problema de memorizar, no generalizar a nuevos ejemplos, sino cada unidad responde a un solo patrón. Los de multicapa son útiles en la aplicación de reconocimiento o clasificación de patrones, permite resolver problemas que no son linealmente separables, de forma que podremos resolver cualquier problema.

   El Perceptrón monocapa solo puede establecer dos regiones separadas por una frontera lineal en el espacio del patrón de entrada, donde habría un hiperplano. Divide las fronteras de decisión en mitades a través de hiperplanos. El perceptrón multicapa puede generar fronteras de decisión arbitrariamente complejas. Puede formar cualquier región convexa en este espacio. Las regiones convexas se forman por la intersección entre las regiones formadas por cada neurona de la capa anterior.

2. Explique los conceptos de accuracy, precision, recall y F1 score.

   Sol:

   Accuracy es la razón entre la observación predicha correctamente y el total de observaciones.

   Accuracy = $\frac{TP+TN}{TP+FP+FN+TN}$

   Precision es la razón de observaciones positivas predichas correctamente al total de observaciones positivas predichas

   Precision = $\frac{TP}{TP+FP}$

   Recall o Sensitivity es la proporción de observaciones positivas predichas correctamente a todas las observaciones en la clase real

   Recall = $\frac{TP}{TP+FN}$

F1 Score considera tanto la precision como el recall. Es la media armónica (promedio) de la precision y el recall. Por lo tanto, tiene en cuenta tanto los falsos positivos como los falsos negativos.

F1 Score $= \frac{2}{\frac{1}{Recall}+\frac{1}{Precision}} = \frac{2*(Recall*Precision)}{Recall+Precision}$

2. Pregunta 2

1. Determine la salida de la red neuronal (y) en función de los pesos y de la entrada.

   Sol:

$$a = sgm(w_{xa} * x)$$
$$b = sgm(w_{xb} * x)$$

   Debido a que la neurona de salida tiene una función de activación lineal:

$$y = c = w_{ac} * a + w_{bc} * b = w_{ac} * sgm(w_{xa} * x) + w_{bc} * sgm(w_{xb} * x)$$

2. Determine las reglas de actualización para cada uno de los pesos en función de y, la entrada, la tasa de aprendizaje μ y la salida deseada d.

   Sol:

   Para la capa de salida:

$$y = c = X^T W = \left[\begin{array}{cc} a & b \end{array}\right] \begin{bmatrix} w_{ac} \\ w_{bc} \end{bmatrix}$$
$$\varepsilon = (d-y)^2 = (d-c)^2 = (d - X^T W)^2$$
$$\nabla = \frac{\partial \varepsilon^2}{\partial W} = 2\varepsilon \frac{\partial \varepsilon}{\partial W} = -2\varepsilon X$$
$$\delta_y = -\frac{\partial \varepsilon^2}{\partial y} = -2\varepsilon$$
$$W_{k+1} = W_k - \mu \nabla_k = W_k + 2\mu\varepsilon_k X_k = W_k + \mu\delta_k X_k$$
$$W_{y,k+1} = \left[\begin{array}{c} w_{ac}^{k+1} \\ w_{bc}^{k+1} \end{array}\right] = \left[\begin{array}{c} w_{ac}^{k} \\ w_{bc}^{k} \end{array}\right] + \mu\delta_{yk} \left[\begin{array}{c} a \\ b \end{array}\right] = \left[\begin{array}{c} w_{ac}^{k} \\ w_{bc}^{k} \end{array}\right] + \mu * 2 * (d-y) * \left[\begin{array}{c} a \\ b \end{array}\right]$$
$$\left[\begin{array}{c} w_{ac}^{k+1} \\ w_{bc}^{k+1} \end{array}\right] = \left[\begin{array}{c} w_{ac}^{k} \\ w_{bc}^{k} \end{array}\right] + \mu * 2 * (d-y) * \left[\begin{array}{c} sgm(w_{xa} * x) \\ sgm(w_{xb} * x) \end{array}\right]$$

   Para la capa oculta:

$$w_{xa}^{k+1} = w_{xa}^{k} + \mu\delta_a x_k$$
$$w_{xb}^{k+1} = w_{xb}^{k} + \mu\delta_b x_k$$
$$\varepsilon_{T_k} = \sum_y \varepsilon_{y_k}^2$$
$$\delta_a = -\frac{\partial \varepsilon_T}{\partial a} = -\sum_y \frac{\partial \varepsilon_T}{\partial y}\frac{\partial y}{\partial a} = \delta_y w_{ac}^k$$

$$\frac{\partial y}{\partial a} = w_{ac}^k$$

$$-\frac{\partial \varepsilon_T}{\partial y} = \delta_y = 2 * (d - y)$$

Analogo para δ_b:

$$\delta_b = \delta_y w_{bc}^k$$

Lo que resulta en:

$$w_{xa}^{k+1} = w_{xa}^k + \mu * w_{ac}^k \delta_y x_k = w_{xa}^k + \mu * w_{ac}^k * 2 * (d - y) * x_k$$

$$w_{xb}^{k+1} = w_{xb}^k + \mu * w_{bc}^k \delta_y x_k = w_{xb}^k + \mu * w_{bc}^k * 2 * (d - y) * x_k$$

3. Realice una actualización de pesos al presentar el ejemplo x = 1 con salida deseada d = -3, suponiendo un vector de pesos inicial $\vec{w} = (w_{xa}^0, w_{xb}^0, w_{ac}^0, w_{bc}^0) = (2,\ 2,\ 1,\ 1)$ y una tasa de aprendizaje $\mu = 0{,}5$.

   Sol: A partir de los resultados anteriores:

$$a = sgm(w_{xa} * x) = sgm(2 * 1) = \frac{1}{1 + e^{-2}} = 0{,}88$$

$$b = sgm(w_{xb} * x) = sgm(2 * 1) = \frac{1}{1 + e^{-2}} = 0{,}88$$

$$y = c = w_{ac} * a + w_{bc} * b = 1 * 0{,}88 + 1 * 0{,}88 = 1{,}76$$

$$w_{ac}^1 = w_{ac}^0 + \mu * 2 * (d - y) * a = 1 + 0{,}5 * 2 * (-3 - 1{,}76) * 0{,}88 = -3{,}1888$$

$$w_{bc}^1 = w_{bc}^0 + \mu * 2 * (d - y) * b = 1 + 0{,}5 * 2 * (-3 - 1{,}76) * 0{,}88 = -3{,}1888$$

$$w_{xa}^1 = w_{xa}^0 + \mu * w_{ac}^0 * 2 * (d - y) * x = 2 + 0{,}5 * 2 * 2 * (-3 - 1{,}76) * 1 = -7{,}52$$

$$w_{xb}^1 = w_{xb}^0 + \mu * w_{bc}^0 * 2 * (d - y) * x = 2 + 0{,}5 * 2 * 2 * (-3 - 1{,}76) * 1 = -7{,}52$$