Cuestionario 1

Aprendizaje Automático

José Manuel Pérez Lendínez

1

Identificar, para cada una de las siguientes tareas, cual es el problema, que tipo de aprendizaje es el adecuado (supervisado, no supervisado, por refuerzo) y los elementos de aprendizaje $(\mathcal{X}, f, \mathcal{Y})$ que deberíamos usar en cada caso. Si una tarea se ajusta a más de un tipo, explicar como y describir los elementos para cada tipo.

a) Clasificación automática de cartas por distrito postal.

Utilizaría aprendizaje supervisado puesto que el problema se basa en ser capaz de reconocer en la dirección de la carta el numero que corresponderá al distrito. Esto seria un reconocedor de dígitos que podrían estar manuscritos o realizado por ordenador e impresos.

También tendría que ser capaz de reconocer y separar la parte de la dirección (calle, piso, numero y el código de distrito.) de la parte del receptor de la carta (nombre y apellidos). Teniendo acceso al dataset de distritos de las ciudades podríamos clasificarlos como perteneciente a uno de estos registros. Se podría añadir un reconocimiento en la dirección de envió y ver si esta corresponde al distrito escrito por el usuario, con esto se podrían reconocer errores en las cartas.

- **X:** Correspondería con las características de una imagen de la etiqueta de la carta.
- **f:** Tendría que ser capaz de reconocer los dígitos del distrito y separarlos del resto de la información.
- **Y:** Daría como salida el distrito al que pertenece la clase clasificándola para este distrito.

b) Decidir si un determinado indice del mercado de valores subirá o bajará dentro de un periodo de tiempo fijado.

Dejando fuera el método de refuerzo que no seria muy útil en este caso podríamos usar tanto supervisado como no supervisado. El problema en este caso es que el entorno de los mercados es muy amplio y depende de muchas variables que no podremos medir. Tiene mucha dependencia de políticas económicas, declaraciones de los propios directivos de las empresas que no podremos medir o tener en cuenta y muchas mas variables difíciles de obtener.

Otro problema seria que lo que paso anterior mente para un valor del mercado podria no darse de la misma manera en el instante actual al depender de muchas características.

Se podría utilizar aprendizaje supervisado revisando una gran cantidad de datos anteriores para ver cuando subieron y cuando bajaron los valores. El problema seria que no podríamos asegurar que lo pasado anteriormente se replicara igual en el presente.

Otra opción seria intentar utilizar el aprendizaje no supervisado para encontrar características comunes en valores que hicieron a estos subir o bajar en

el pasado y ver si en el presente se están cumpliendo o no utilizando posteriormente un sistema de aprendizaje supervisado para entrenarlo con estas nuevas características.

En el caso de el aprendizaje supervisado necesitaremos los siguiente:

- **X:** Como entrada para etiquetar los datos podriamos utilizar por ejemplo:
  - Valor actual.
  - Valor máximo alcanzado.
  - Valor mínimo alcanzado.
  - Media del valor en los últimos meses.
  - Valor contable y de mercado para la empresa
  - Valor contable y de mercado para la acción.

El problema es el explicado anteriormente. Es muy dificil recoger toda la información necesaria para saber si una acción subirá o bajara ya que depende de muchas opciones(política, escándalos de directivos, declaraciones de directivos, problemas en la empresa, ...)

- **F:** Necesitaría ser capaz de reconocer mediante la información dada patrones que haga probable que el valor suba o baje.
- **Y:** : Clasificaría en crece, se maniten o decrece el precio.

c) Hacer que un dron sea capaz de rodear un obstáculo.

Utilizaría aprendizaje por refuerzo, usando para ello un simulador que contenga las características que queremos que el dron enfrente en la realidad. Daremos un refuerzo positivo a las acciones que aportan un buen resultado al movimiento alrededor del objeto a rodear.

d) Dada una colección de fotos de perros, posiblemente de distintas razas, establecer cuantas razas distintas hay representadas en la colección.

Utilizaríamos aprendizaje no supervisado al no tener las fotos etiquetadas y no saber ni siquiera el numero de clases que podríamos tener. Al no tener esta información buscaremos características comunes en las fotos mediante un algoritmos de aprendizaje no supervisado. Con esto podríamos obtener características que identificaran una o varias clases. Los únicos datos necesario serian las fotos de los perros.

2

¿Cuales de los siguientes problemas son más adecuados para una aproximación por aprendizaje y cuales más adecuados para una aproximación por diseño? Justificar la decisión

a) Determinar si un vertebrado es mamífero, reptil, ave, anfibio o pez.

Usaría diseño puesto que las características que diferencian a los tipos de vertebrados son conocidas y no necesitamos buscar una función nueva para ver como clasificarlos.

b) Determinar si se debe aplicar una campaña de vacunación contra una enfermedad.

Se podria usar cualquiera de las dos opciones.

- **Diseño:** A partir de los datos de los últimos años decidir si la enfermedad puede ir a peor y seria necesario una camapaña de vacunación para controlarla o si la enfermedad no es tan grave para llevar a cabo esta campaña. Se podrían utilizar datos como numero de afectado en los últimos años, probabilidad de contagio de la enfermedad, gravedad de la enfermedad o resitencia a la vacuna de la enfermedad.

- **Aprendizaje:** Si tenemos información suficiente de campañas anteriores y como la campaña consiguió o no controlar la enfermedad. Con esta enfermedad podríamos usar el aprendizaje para que evalué una nueva posible campaña basándose en los datos de las campañas anteriores y si estas fueron útiles para controlarla o no.

c) Determinar perfiles de consumidor en una cadena de supermercados.

Utilizaría aprendizaje puesto al no conocer los posibles perfiles de consumidor que puede tener el supermercado o no saber todas las características que son importantes para clasificar como un perfil u otro.

d) Determinar el estado anímico de una persona a partir de una foto de su cara. Utilizaría una aproximación por aprendizaje, debido a que lo que necesitamos es ser capaces a partir de una foto el estado de animo de esa personal. Si tenemos fotos etiquetadas con el estado de animo de la persona podríamos crear un modelo de aprendizaje que nos devuelva como salida uno de los estados de ánimos que utilizamos. Por tanto estaríamos buscando una función que no conocemos para reconocer el estado de animo de la persona.

e) Determinar el ciclo óptimo para las luces de los semáforos en un cruce con mucho tráfico.

Yo usaría una aproximación por diseño en el que mediría la cantidad de trafico de las calles que llegan al cruce controlado por los semáforos. Con esto daría mas tiempo de luz verde a las calles con mas congestión de trafico.

3

Construir un problema de aprendizaje desde datos para un problema de clasificación de fruta en una explotación agraria que produce mangos, papayas y guayabas. Identificar los siguientes elementos formales X , Y, D, f del problema. Dar una descripción de los mismos que pueda ser usada por un computador. ¿Considera que en este problema estamos ante un caso de etiquetas con ruido o sin ruido? Justificar las respuestas.

Nuestra etiqueta de clase es la dada por el enunciado. También se podrían utilizar etiquetas numéricas. Por ejemplo -1 para papaya, 0 para mango y 1 para guayaba

$$Y = \{papaya, mango, guayaba\}$$

La entrada seria un vector de las características medidas a cada fruta. Entre estas podríamos utilizar por ejemplo, tamaño, peso, color, forma, textura , etc...

$$X = x_1, x_2, ..., x_n, y$$

El conjunto de datos para el entrenamiento estaría compuesto por los vectores de características de todas las frutas que hemos catalogado. Seria una matriz con n+1 columnas (numero de características por fruta y la clase a la que pertenece) y m numero de filas que sera el número de frutas medidas.

$$D = \begin{bmatrix} x_{1,1} & x_{1,2} & .... & x_{1,n} & y_1 \\ . & . & .... & . & . \\ . & . & .... & . & . \\ x_{m,1} & x_{m,2} & .... & x_{m,n} & y_n \end{bmatrix}$$

Mediante estos datos intentaremos aproximarnos a una función f que consiga mediante una instancia de datos reconocer a que fruta pertenece.

$$f(x_{i,1}x_{i,2}....x_{i,n}) = y_i$$

En cuanto el ruido puede ser que se introduzca en la muestra de datos si a la hora de realizar las medidas de las características de cada fruta no se hace de forma correcta y se cometen errores en algunas muestras. También se podría tener ruido dependiendo de si se cogen buenos ejemplares para la muestra o ejemplares que podrían no representar totalmente a la fruta por defectos de esta.

4

Suponga una matriz cuadrada A que admita la descomposición $A = X^T X$ para alguna matriz X de números reales. Establezca una relación entre los valores singulares de las matriz A y los valores singulares de X.

5

Sean x e y dos vectores de características de dimensión Mx1. La expresión

$$\mathrm{cov}(\mathbf{x},\mathbf{y}) = \frac{1}{M}\sum_{i=1}^{M}(x_i-\overline{x})(y_i-\overline{y})$$

define la covarianza entre dichos vectores, donde $\overline{z}$ representa el valor medio de los elementos de z. Considere ahora una matriz X cuyas columnas representan vectores de características. La matriz de covarianzas asociada a la matriz $\mathrm{X} = (\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_N)$ es el conjunto de covarianzas definidas por cada dos de sus vectores columnas. Es decir, $\mathrm{cov(X)} = \begin{pmatrix} \mathrm{cov}(\mathbf{x}_1,\mathbf{x}_1) & \mathrm{cov}(\mathbf{x}_1,\mathbf{x}_2) & \cdots & \mathrm{cov}(\mathbf{x}_1,\mathbf{x}_N) \\ \mathrm{cov}(\mathbf{x}_2,\mathbf{x}_1) & \mathrm{cov}(\mathbf{x}_2,\mathbf{x}_2) & \cdots & \mathrm{cov}(\mathbf{x}_2,\mathbf{x}_N) \\ \cdots & \cdots & \cdots & \cdots \\ \mathrm{cov}(\mathbf{x}_N,\mathbf{x}_1) & \mathrm{cov}(\mathbf{x}_N,\mathbf{x}_2) & \cdots & \mathrm{cov}(\mathbf{x}_N,\mathbf{x}_N) \end{pmatrix}$

Sea $1_M^T = (1, 1, \cdots, 1)$ **un vector** $M \times 1$ **de unos. Mostrar que representan las siguientes expresiones**

$a) E1 = 11^T\mathrm{X}$

Vamos por partes. Primero tenemos la multiplicacion de $1 * 1^{(t)}$ obtenemos una matriz de 1 con tamaño mxm.

$$\begin{bmatrix} 1_{1,1} & 1_{1,2} & .... & 1_{1,m} \\ . & . & .... & . \\ . & . & .... & . \\ 1_{m,1} & 1_{m,2} & .... & 1_{m,m} \end{bmatrix}$$

Al multiplicar esta matriz por $\mathrm{X} = (\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_N)$ obtenemos

$$\begin{bmatrix} \sum_{i=1}^m x_{1,i} & \sum_{i=1}^m x_{2,i} & .... & \sum_{i=1}^m x_{m,i} \\ . & . & .... & . \\ . & . & .... & . \\ \sum_{i=1}^m x_{1,i} & \sum_{i=1}^m x_{2,i} & .... & \sum_{i=1}^m x_{m,i} \end{bmatrix}$$

Donde en cada posición de la matriz tenemos la sumatoria de las caracteristicas de la matriz X para esa columna.

b) $E2 = \left(\mathrm{X} - \frac{1}{M}E1\right)^T \left(\mathrm{X} - \frac{1}{M}E1\right)$ Vamos a a calcular primero $X - \frac{1}{M}E1$. Al multiplicarlos por $\frac{1}{M}$ nos quedaremos en cada numero de la matriz con la media de esa columna de características.

$$\frac{1}{M}\begin{bmatrix} \sum_{i=1}^m x_{1,i} & \sum_{i=1}^m x_{2,i} & .... & \sum_{i=1}^m x_{n,i} \\ . & . & .... & . \\ . & . & .... & . \\ \sum_{i=1}^m x_{1,i} & \sum_{i=1}^m x_{2,i} & .... & \sum_{i=1}^m x_{n,i} \end{bmatrix} = \begin{bmatrix} \overline{x_1} & \overline{x_2} & .... & \overline{x_m} \\ . & . & .... & . \\ . & . & .... & . \\ \overline{x_1} & \overline{x_2} & .... & \overline{x_m} \end{bmatrix}$$

A continuación le restamos la matriz X.

$$\begin{bmatrix} x_{1,1} & x_{1,2} & .... & x_{1,m} \\ . & . & .... & . \\ . & . & .... & . \\ x_{m,1} & x_{m,2} & .... & x_{m,m} \end{bmatrix} - \begin{bmatrix} \overline{x_1} & \overline{x_2} & .... & \overline{x_m} \\ . & . & .... & . \\ . & . & .... & . \\ \overline{x_1} & \overline{x_2} & .... & \overline{x_m} \end{bmatrix} = \begin{bmatrix} x_{1,1} - \overline{x_1} & x_{1,2} - \overline{x_2} & .... & x_{1,m} - \overline{x_m} \\ . & . & .... & . \\ . & . & .... & . \\ x_{m,1} - \overline{x_1} & x_{m,2} - \overline{x_2} & .... & x_{m,m} - \overline{x_m} \end{bmatrix}$$

Con esto ya tenemos la parte interior de los paréntesis ahora tenemos que transpone la primara y multiplicarla por la segunda.

$$\begin{bmatrix} x_{1,1} - \overline{x_1} & .... & x_{m,1} - \overline{x_1} \\ x_{1,2} - \overline{x_2} & .... & x_{m,2} - \overline{x_2} \\ . & .... & . \\ x_{1,m} - \overline{x_m} & .... & x_{m,m} - \overline{x_m} \end{bmatrix} * \begin{bmatrix} x_{1,1} - \overline{x_1} & x_{1,2} - \overline{x_2} & .... & x_{1,m} - \overline{x_m} \\ . & . & .... & . \\ . & . & .... & . \\ x_{m,1} - \overline{x_1} & x_{m,2} - \overline{x_2} & .... & x_{m,m} - \overline{x_m} \end{bmatrix}$$

Vamos a analizar que se pondría en la posición (1,1) de la matriz resultante de multiplicarlas.

$$(x_{1,1} * \overline{x1}) * (x_{1,1} * \overline{x1}) + (x_{2,1} * \overline{x1}) * (x_{2,1} * \overline{x1}) + ... + (x_{m,1} * \overline{x1}) * (x_{m,1} * \overline{x2})$$

Con esto se ve claramente que la matriz E2 es igual al nominador de la formula de la covarianza($\mathrm{Cov}(X,Y) = \frac{\sum_{i=1}^{M}(x_i - \overline{x})(y_i - \overline{y})}{M}$). Como nosotros no dividimos por M en la matriz E2 tendremos lo mismos que con M*cov(x).

6

Considerar la matriz hat definida en regresión $\hat{\mathrm{H}} = \mathrm{X}\left(\mathrm{X}^{\mathrm{T}}\mathrm{X}\right)^{-1}\mathrm{X}^{\mathrm{T}}$ **donde X es la matriz de observaciones de dimensión** $N \times (d+1), \mathrm{y}\mathrm{X}^{\mathrm{T}}\mathrm{X}$ **es invertible.**

a) ¿Que representa la matriz $\hat{\mathrm{H}}$ **en un modelo de regresión?**

La matriz hat representa la matriz de proyección con los valores de las observaciones para la variable Y en nuestro modelo X que contiene las observación para cada una de las variables en la regresión. Tenemos que tener en cuenta que tendremos una diferencia entre los valores para Y dados por la matriz hat y los valores reales de Y. Esta diferencia en los valores llamada residuos también podemos obtenerlas mediante la matriz hat.

b) Identifique la propiedad más relevante de dicha matriz en relación con regresión lineal.

La matriz $H = X(X^TX)^-1X^T$

Idempotente: Elevada por un numero es igual a si misma.

$$H * H = X(X^TX)^{-1}X^TX(X^TX)^{-1}X^T$$

Como se ve estamos multiplicando $X^TX(X^TX)^-1$ que nos da la matriz identidad y por tanto nos quedamos con $X(X^TX)^{-1}X^T = H$

Me parece la propiedad mas importante porque nos esta asegurando que si predecimos H y queremos predecir H^2 o a cualquier potencia estaríamos prediciendo lo mismo que en H. Por tanto nos tendría que dar los mimos resultados para dicha predicción

7

La regla de adaptación de los pesos del Perceptron ($w_{new} = W_{old} + yx$) **tiene la interesante propiedad de que mueve el vector de pesos en la dirección adecuada para clasificar x de forma correcta. Suponga el vector de pesos w de un modelo y un datos x(t) mal clasificado respecto de dicho modelo. Probar matemáticamente que el movimiento de la regla de adaptación de pesos siempre produce un movimiento de w en la dirección correcta para clasificar bien x(t)**

Tenemos que tener en cuenta que si $\mathbf{w}^T(n)\mathbf{x}(n) \geq 0$ y $\mathbf{x}(n)$ que pertenezca a la clase ζ_1 y si $\mathbf{w}^T(n)\mathbf{x}(n) < 0$ para cada vector de entrada $\mathbf{x}$ que pertenezca a la clase ζ_2 Vamos a ver los tres casos posibles.

- x(t) estaría bien clasificado: No se llevaría a cabo ninguna corrección en el vector y se tendría que $w_{new} = W_{old}$

- si $\mathbf{w}^T(n)\mathbf{x}(n) \geq 0$ y $\mathbf{x}(n)$ que pertenezca a la clase ζ_2. En este caso tendríamos un error y se tendría que mover el vector.

  $$\mathbf{w}(new) = \mathbf{w}(old) + y(n)\mathbf{x}(n)$$

  Como y(n) daría un -1 con esto se movería hacia el lado correcto para clasificarlo

- si $\mathbf{w}^T(n)\mathbf{x}(n) < 0$ y $\mathbf{x}(n)$ que pertenezca a la clase ζ_J: Este caso seria igual al anterior pero cambiando en este caso nos daría y(n) el valor 1 por lo cual se movería hacia el lado contrario al caso anterior y moviéndose hacia el x(n)

8

Sea un problema probabilistico de clasificación binaria con etiquetas $\{0, 1\}$**, es decir** $\mathrm{P(Y = 1)} = h(\mathrm{x})\mathrm{yP(Y = 0)} = 1 - h(\mathrm{x})$**, para una funcion** $h()$ **dependiente de la muestra.**

1. **Considere una muestra i.i.d. de tamaño** $\mathrm{N\,(x1, \cdots, xN)}$**. Mostrar que la funcion** h **que maximiza la verosimilitud de la muestra es la misma que minimiza**

$$E_{\mathrm{in}}(\mathbf{w}) = \sum_{n=1}^{N} [y_n = 1] \ln \frac{1}{h\,(\mathbf{x}n)} + [yn = 0] \ln \frac{1}{1 - h\,(\mathbf{x}_n)}$$

donde $[\cdot]$ vale 1 o 0 según que sea verdad o falso respectivamente la expresión en su interior.

2. **Para el caso $h(x) = \sigma\left(\mathbf{w}^T\mathbf{x}\right)$ mostrar que minimizar el error de la muestra en el apartado anterior es equivalente a minimizar el error muestral.**

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} \ln\left(1 + e^{-y_n \mathbf{w}^T \mathbf{x}_n}\right)$$

9

Derivar el error E_{in} para mostrar que en regresión logística se verifica:

$$\nabla E_{\text{in}}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \frac{y_n \mathbf{x} n}{1 + e^{yn\mathbf{w}^T\mathbf{x}n}} = \frac{1}{N}\sum n = 1^N - y_n \mathbf{x} n \sigma\left(-yn\mathbf{w}^T\mathbf{x}_n\right)$$

Argumentar sobre si un ejemplo mal clasificado contribuye al gradiente más que un ejemplo bien clasificado.

10

Definamos el error en un punto (x_n, y_n) por

$$\mathbf{e}n(\mathbf{w}) = \max\left(0, -yn\mathbf{w}^T\mathbf{x}_n\right)$$

Argumentar si con esta función de error el algoritmo PLA puede interpretarse como SGD sobre e_n con tasa de aprendizaje $v = 1$.

Cambiamos en la función para calcular el gradiente descendente estocástico los datos por los dados en el enunciado. $w_j = w_j - 1\frac{\partial(max(0, -y_n\mathbf{w}^T\mathbf{x}_n))}{\partial w_j}$.

Para esto hay que derivar $-y_n\mathbf{w}^T\mathbf{x}_n$ respecto a w. Esto nos daría $max(0, -y_n x_n)$ y por tanto se quedaría que $wj = wj - 1 * max(0, -y_n x_n)$. Con esto tenemos que si el elemento se clasifica bien el valor seria 0 y no se modificaría el vector de peso y si esta mal clasificado nos quedaríamos con $-y_n x_n$. La formula para modificar el peso en el vector es

$$\mathbf{w}(new) = \mathbf{w}(old) + y(n)\mathbf{x}(n)$$

y esta coincide con lo que hemos visto anteriormente y se usa solo cuando se clasifica mal.

Con esto podemos decir que el SGD con estos parámetros si coinciden con PLA