Regresión Logística

Carlos Malanche

10 de abril de 2018

1. Clasificación

No todos los problemas son sobre predecir valores sobre la recta real. En algunos casos, vamos a tener medidas en que los elementos de la serie de datos $D = \{\underset{\rightarrow}{x}_i, y_i\}_{i=1}^{n}$ cumplen $\underset{\rightarrow}{x}_i \in \mathbb{R}^m$, $y_i \in \{\text{Clase } 1, \text{Clase } 2\}$ (por ejemplo al tratar de caracterizar especies, los biólogos deben utilizar categorías). Vamos a comenzar por intentar resolver el problema para dos categorías.

1.1. Clasifiación binaria

El primer paso (intuitivamente) es convertir las clases a valores que un modelo lineal pueda predecir, es decir, números. Digamos pues que $y_i \in \{0, 1\}$, Qué ocurre si entrenamos un modelo lineal para predecir esta variable? Recordemos que la predicción de un modelo lineal con vector de parámetros $\underset{\rightarrow}{c}$, para una nueva observación $\underset{\rightarrow}{x}_0$ se obtiene así

$$y_0 = \langle \underset{\rightarrow}{x}_0, \underset{\rightarrow}{c} \rangle \tag{1}$$

No es muy difícil de ver que bastará con que $\underset{\rightarrow}{c}$ tenga al menos dos entradas distintas de cero para obtener y_0 no sólo diferente a 0 o a 1, si no posiblemente fuera del rango de estas dos. Si el valor obtenido estuviera limitado al intervalo $[0, 1]$, podríamos pensar en interpretar el valor de y como una probabilidad... vaya, si tan sólo hubiera una manera de mapear el intervalo $(-\infty, \infty)$ (rango del operador $\langle \cdot, \cdot \rangle$) al $(0, 1)$...

1.2. Regresión Logística

Definamos como función sigmoide (a veces llamada función logística) la siguiente función

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

Si han leído la sección anterior, sabrán qué busca cumplir esta función: Mapear el rango del producto interno al intervalo abierto $(0, 1)$.

Ahora podemos interpretar el resultado de nuestro predictor lineal, al pasar por la función sigmoide, como una probabilidad de que la medición $\underset{\rightarrow}{x}_0$ pertenece a una de las dos clases (es decir, $P(y_0 \in \text{Clase } 2) = \sigma(\langle \underset{\rightarrow}{x}_0, \underset{\rightarrow}{c} \rangle$ y por ser una probabilidad binomial, $P(y_0 \in \text{Clase } 1) = 1 - \sigma(\langle \underset{\rightarrow}{x}_0, \underset{\rightarrow}{c} \rangle)$.

Todo muy bien hasta ahora, pero hay que preguntarnos: Si vamos a hacer esto... no hay que entrenar el modelo de manera diferente? La respuesta es sí, pues hemos modificado la función de costo:

$$J(\underset{\rightarrow}{c}) = \frac{1}{2n} \sum_{i=1}^{n} (\sigma(\underset{\rightarrow}{x}_i \cdot \underset{\rightarrow}{c}) - y_i)^2 \tag{3}$$

Noten que otra manera de derivar esta función de costo es utilizar máxima verosimilitud.

$$\begin{aligned} P(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{c}) &= \prod_{n=1}^{m} p(y_n, \boldsymbol{x}_n) \\ &= \prod_{n:y_n=1} p(y_n|\boldsymbol{x}_n) \prod_{n:y_n=0} p(y_n|\boldsymbol{x}_n) \\ &= \prod_{n=1}^{N} \sigma(\boldsymbol{x}_n^T \boldsymbol{c})^{y_n} (1 - \sigma(\boldsymbol{x}_n^T \boldsymbol{c}))^{1-y_n} \end{aligned}$$

Usando un truco similar al de la vez pasada, convertimos esto una función compuesta de sumandos al aplicar un logaritmo

$$
\begin{aligned}
\nabla \mathcal{L}(\boldsymbol{x}) = -\log(\prod_{n=1}^{N} \sigma(\boldsymbol{x}_n^T \boldsymbol{c})^{y_n}(1-\sigma(\boldsymbol{x}_n^T \boldsymbol{c}))^{1-y_n}) &= -\sum_{n=1}^{N} (y_n)\log(\sigma(\boldsymbol{x}_n^T \boldsymbol{c})) + (1-y_n)\log(1-\sigma(\boldsymbol{x}_n^T \boldsymbol{c})) \\
&= -\sum_{n=1}^{N} \log(1-\sigma(\boldsymbol{x}_n^T \boldsymbol{c})) + y_n \log(\frac{1-\sigma(\boldsymbol{x}_n^T \boldsymbol{c})}{\sigma(\boldsymbol{x}_n^T \boldsymbol{c})}) \\
&= \sum_{n=1}^{N} \log(1+e^{\boldsymbol{x}_n^T \boldsymbol{c}}) - y_n \boldsymbol{x}_n^T \boldsymbol{c}
\end{aligned}
$$

Para poder encontrar un mínimo, es necesario derivar y encontrar el gradiente. Notemos que

$$
\frac{\partial}{\partial x}\log(1+e^x) = \frac{e^x}{1+e^x} = \sigma(x) \tag{4}
$$

Con ello, la derivada de la función objetivo es

$$
\begin{aligned}
\nabla \mathcal{L}(\boldsymbol{c}) &= \sum_{n=1}^{N} \boldsymbol{x}_n \sigma(\boldsymbol{x}_n^T \boldsymbol{c}) - y_n \boldsymbol{x}_x \\
&= \sum_{n=1}^{N} \boldsymbol{x}_n (\sigma(\boldsymbol{x}_n^T \boldsymbol{c}) - y_n) \\
&= \boldsymbol{X}^T(\sigma(\boldsymbol{X}\boldsymbol{c})) - \boldsymbol{y})
\end{aligned}
$$

En donde al final se utilizó notación matricial para quitar la suma (noten queσ aplicada a un vector es lo mismo que aplicar la función en cada componente).

Igualar a cero este gradiente nos deja con un problema que no tiene solución analítica. Se puede usar un descenso de gradiente pues la función es convexa... lo es?

1.3. Convexidad

Observemos que una línea es convexa (pero no estrictamente convexa). La suma de funciones convexas es convexa. Podemos decir que las últimas N funciones forman una función convexa.

Ahora, para el logaritmo: para la función $\log(1+e^x)$ tenemos que su primer derivada es la función sigmoide. Su derivada es

$$
\frac{\partial}{\partial x}\sigma(x) = \frac{\partial}{\partial x}(1+e^{-x})^{-}1 = (1+e^{-x})^{-2}e^{-x} = \sigma(x)\frac{e^{-x}}{1+e^{-x}} = \sigma(x)(1-\sigma(x)) \tag{5}
$$

valor que siempre es positivo. Esto implica que la función es convexa. Por último, el argumento de la exponencial es una función lineal, y la composición de dos funciones convexas es convexa si la externa es no-decreciente. (la primer derivada es mayor o igual a cero). Todos los términos de la función objetivo son convexos!