Universidad de Ingeniería y Tecnología - UTEC
Departamento de Ingeniería Mecatrónica
Curso: Tópicos de Ingeniería Mecatrónica (2020 -2)

Tema 2
Clasisficación: Regresión Logística

Prof. Oscar E. Ramos Ponce

La clasificación consiste en predecir a qué clase pertenece cada uno de los datos de entrada. A veces las clases se denominan *etiquetas* (*labels*), categorías, o *targets*. Un ejemplo de clasificación binaria es la detección de correo no deseado (*spam*) dado que solo hay dos posibles clases: correo no deseado, o correo deseado (*spam* o no *spam*). En este caso un clasificador utiliza los datos de entrenamiento para tratar de elaborar un modelo que encuentre características semejantes en el correo no deseado.

La clasificación es un problema que pertenece a la categoría de aprendizaje supervisado, donde se proporciona tanto los datos como las etiquetas o clases a las que pertenecen estos datos.

Se puede distinguir dos tipos de clasificadores:

- *Clasificadores binarios (de dos clases)*- Dado un conjunto de datos, buscan clasificar una única categoría, pero al hacerlo, indirectamente clasifican además todo aquello que no pertenece a la categoría. En el ejemplo anterior, al clasificar *spam*, indirectamente se está también clasificando lo que no es *spam*. Por este motivo se les llama de dos clases. Las etiquetas de estos clasificadores son una variable $y \in \{0, 1\}$ con solamente dos posibilidades, donde 1 típicamente indica la clase que se busca (clase positiva), y 0 el resto (clase negativa). Sin embargo, los clasificadores $h_w(\boldsymbol{x})$ suelen tener como salida un valor real entre 0 y 1 ($0 \leq h_w(\boldsymbol{x}) \leq 1$) que se interpreta como una probabilidad: mientras más cerca esté la predicción de 1, más seguro está el clasificador sobre la pertenencia a la clase 1, y viceversa.

- *Clasificadores multiclase.* En este caso se busca clasificar más de una clase de manera explícita. Si se desea clasificar n clases, las etiquetas son $y = \{0, 1, 2, \cdots, n-1\}$ o $y = \{1, 2, 3, \cdots, n\}$, dependiendo de la convención que se esté utilizando.

Existen bastantes métodos que se utilizan para realizar clasificación. Algunos de estos métodos son los siguientes:

- Regresión logística
- Redes neuronales
- Support Vector Machines (SVMs)

- Naive Bayes
- Redes Bayesianas
- Árboles de decisión
- K-nearest neighbor

En este tema solamente se verá la regresión logística, pero posteriormente se verá otros métodos de clasificación.

1. Función de Hipótesis

Debido a que la *regresión logística* es un método de clasificación, se desea que la función de hipótesis se encuentre entre 0 y 1; es decir, $h_w(\boldsymbol{x}) \in [0, 1]$. Mientras más cercano se encuentre el valor a 1, más será la certidumbre de que se tiene la clase 1, y mientras más cercano se encuentre el valor a 0, más seguro se estará de que se tiene la clase 0. En regresión logística, la función de hipótesis se define como

$$h_w(\boldsymbol{x}) = g(\boldsymbol{w}^T\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w}^T\boldsymbol{x}}} \tag{1}$$

donde la función g se denomina función *sigmoidea* o *logística* y se define como

$$g(z) = \frac{1}{1 + e^{-z}}.$$

La forma de esta función sigmoidea o logística se muestra en la Fig. 1. Como se observa, se encuentra definida en el rango de 0 a 1, interseca al eje vertical en 0.5, y tiene como asíntotas precisamente a 0 y a 1.

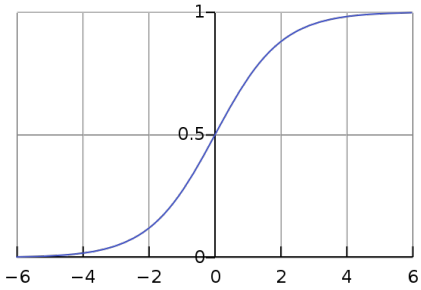

Figura 1: Función logística $g(z)$

Como muestra (1), en la clasificación a través de regresión logística, primero se realiza un ajuste lineal $\boldsymbol{w}^T\boldsymbol{x}$, cuyo valor abarca todo $\mathbb{R}$, y luego a esta regresión lineal se le aplica la función logística para obtener valores entre 0 y 1.

De manera probabilística, la función de hipótesis dada en (1) se puede interpretar como

$$h_w(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}; \boldsymbol{w}),$$

que es la probabilidad que se tenga la clase $y = 1$ dado el valor de entrada $\boldsymbol{x}$, utilizando la parametrización $\boldsymbol{w}$. En la expresión anterior se usa ";" para denotar que no existe condicionamiento en $\boldsymbol{w}$, debido a que $\boldsymbol{w}$ no es una variable aleatoria, sino solo se utiliza para representar un modelo determinado. De igual manera, utilizando leyes probabilísticas se tiene $p(y = 0|\boldsymbol{x}; \boldsymbol{w}) + p(y = 1|\boldsymbol{x}; \boldsymbol{w}) = 1$, de donde se obtiene

$$p(y = 0|\boldsymbol{x}; \boldsymbol{w}) = 1 - p(y = 1|\boldsymbol{x}; \boldsymbol{w}).$$

1.1. Predicción

Con regresión logística la predicción de la clase 0 o 1 se realizará de la siguiente manera:

- Se predice $y = 1$ si $h_w(\boldsymbol{x}) \geq 0.5$ o equivalentemente si $\boldsymbol{w}^T\boldsymbol{x} \geq 0$.
- Se predice $y = 0$ si $h_w(\boldsymbol{x}) < 0.5$ o equivalentemente si $\boldsymbol{w}^T\boldsymbol{x} < 0$.

La equivalencia entre $h_w(\boldsymbol{x})$ y $\boldsymbol{w}^T\boldsymbol{x}$ se puede deducir por observación del gráfico de la función sigmoidea. Dadas las dos condiciones anteriores, la predicción se basa en la recta que determina $\boldsymbol{w}^T\boldsymbol{x}$: por encima de esta recta se tiene una clase, y por debajo de la recta se tiene otra clase. Debido a que esta recta determina el límite de la clasificación, se le suele denominar *frontera de decisión* (*decision boundary*).

Ejemplo. Si los datos de entrenamiento tienen dos atributos, $\boldsymbol{x} \in \mathbb{R}^2$, la función de hipótesis está dada por $h_w(\boldsymbol{x}) = g(w_0 + w_1x_1 + w_2x_2)$. En este caso, el argumento de la función sigmoidea $w_0 + w_1x_1 + w_2x_2 = 0$, define una frontera de decisión lineal que separa al plano $x_1 - x_2$ en dos partes.

1.2. Extensiones

Así como era posible generalizar la regresión lineal para contener una combinación lineal de términos no lineales, la regresión logística puede también ser generalizada para contener términos genéricos. Con este fin, se define la función de hipótesis usando bases no lineales como

$$h_w(\boldsymbol{x}) = g\left(\boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x})\right)$$

donde

$$\phi(\boldsymbol{x}) = \begin{bmatrix} \phi_0(\boldsymbol{x}) \\ \phi_1(\boldsymbol{x}) \\ \vdots \\ \phi_n(\boldsymbol{x}) \end{bmatrix}$$

es el vector de bases no lineales, y usualmente se asigna $\phi_0(\boldsymbol{x}) = 1$ para que w_0 sea el término de *bias*. Estas bases no lineales pueden ser términos polinomiales, trigonométricos, entre otras funciones no lineales. Por ejemplo, para el caso polinómico se podría tener términos como $x_1^2x_2$, $x_1^2x_2^2$, $x_1^3x_2$, entre muchas otras posibilidades.

Ejemplo. Si se utiliza la función de hipótesis $h_w(\boldsymbol{x}) = g(w_0 + w_1x_1^2 + w_2x_2^2)$ para datos que tienen dos atributos (x_1, x_2), se obtiene una frontera de decisión descrita por una circunferencia. Lo que se encuentra dentro de la circunferencia se clasifica como $y = 0$, y lo que se encuentra fuera como $y = 1$.

2. Función de Costo

2.1. Para una Instancia

La función de costo ℓ para una sola instancia se puede inicialmente definir como

$$\ell(h_w(\boldsymbol{x}), y) = \begin{cases} -\log(h_w(\boldsymbol{x})), & y = 1 \\ -\log(1 - h_w(\boldsymbol{x})), & y = 0 \end{cases}$$

donde log representa el logaritmo neperiano. La justificación de esta función de costo se puede realizar analizando qué sucede con cada una de las clases por separado.

- Si se tiene la clase 1; es decir, $y = 1$, la función de costo será $\ell(h_w(\boldsymbol{x}), y) = -\log(h_w(\boldsymbol{x}))$. Se puede verificar que los límites de esta función, cuando h_w tiende a cero y a 1, son:

$$\lim_{h_w(\boldsymbol{x})\to 0} -\log(h_w(\boldsymbol{x})) = \infty, \qquad \lim_{h_w(\boldsymbol{x})\to 1} -\log(h_w(\boldsymbol{x})) = 0,$$

  lo cual implica que a medida que la hipótesis se acerca al valor real de 1, el costo se aproxima a cero, y a medida que la hipótesis se aleja de este valor (se aproxima a cero), el costo se hace infinitamente grande.

- Si $y = 0$, se tiene la función de costo $\ell(h_w(\boldsymbol{x}), y) = -\log(1 - h_w(\boldsymbol{x}))$. Se puede verificar que los límites de esta función, cuando h_w tiende a cero y cuando tiende a 1, son:

$$\lim_{h_w(\boldsymbol{x})\to 0} -\log\left(1 - h_w(\boldsymbol{x})\right) = 0, \qquad \lim_{h_w(\boldsymbol{x})\to 1} -\log(1 - h_w(\boldsymbol{x})) = \infty.$$

  De este modo, a medida que la hipótesis se acerca al valor real de 0, el costo también se aproxima a cero, y a medida que la hipótesis se aleja de este valor (se aproxima a 1), el costo se hace infinitamente grande.

Este análisis de cada uno de los casos justifica, de manera informal, la elección de esta función de costo.

Por facilidad de notación, debido a que $y \in \{0, 1\}$, la función de costo para una sola instancia $\ell(h_w(\boldsymbol{x}), y)$ se puede escribir de manera equivalente como

$$\ell(h_w(\boldsymbol{x}), y) = -y \log(h_w(\boldsymbol{x})) - (1 - y) \log(1 - h_w(\boldsymbol{x})). \tag{2}$$

Para verificarlo, se debe considerar que y solo puede tomar dos valores: 0 o 1. Si $y = 1$, el segundo término se anula, quedando solo el primero; y si $y = 0$, el primer término se anula, quedando solo el segundo término.

2.2. Para Todas las Instancias

La función de costo dada en (2) fue definida para una sola instancia. Considerando todas las n instancias, esta función se convierte en la siguiente sumatoria:

$$J(\boldsymbol{w}) = \frac{1}{n} \sum_{i=1}^{n} \ell(h_w(\boldsymbol{x}^{(i)}), y^{(i)}),$$

la cual se puede expresar de manera explícita como

$$J(\boldsymbol{w}) = -\frac{1}{n} \sum_{i=1}^{n} y^{(i)} \log(h_w(\boldsymbol{x}^{(i)})) + (1 - y^{(i)}) \log(1 - h_w(\boldsymbol{x}^{(i)})). \tag{3}$$

Esta función es la función de costo que se utiliza para la regresión logística.

2.3. Optimización usando Descenso del Gradiente

Para poder encontrar la hipótesis $h_w(\boldsymbol{x})$ que clasifique mejor los datos de entrenamiento, se debe minimizar la función costo. Así, el problema que se busca resolver es

$$\min_{w} J(\boldsymbol{w}),$$

cuya solución determinará los parámetros $\boldsymbol{w}$ que hagan que la predicción binaria $h_w(\boldsymbol{x})$ sea lo más cercana a la realidad y.

La derivada de la función de costo para la regresión logística, dada en (3), es

$$\frac{\partial}{\partial w_j} J(\boldsymbol{w}) = \frac{1}{n} \sum_{i=1}^{n} \left(h_w(\boldsymbol{x}^{(i)}) - y^{(i)} \right) x_j^{(i)},$$

la cual, coincidentemente, es semejante a la derivada de la función de costo para la regresión lineal. Utilizando esta derivada, el método del descenso de gradiente se puede representar como

$$w_j := w_j - \alpha \frac{1}{n} \sum_{i=1}^{n} \left(h_w(\boldsymbol{x}^{(i)}) - y^{(i)} \right) x_j^{(i)}, \tag{4}$$

donde se debe actualizar simultáneamente cada $j = 0, \cdots, d$, asumiendo que la entrada $\boldsymbol{x}$ tiene d atributos (se inicia desde 0 para considerar al *bias* w_0).

De manera alternativa, se puede realizar la actualización de todos los parámetros a la vez utilizando el gradiente de la función de costo $\nabla_{\boldsymbol{w}} J(\boldsymbol{w})$. Utilizando este gradiente, la actualización de los parámetros está dada por

$$\boldsymbol{w} := \boldsymbol{w} - \alpha \frac{1}{n} \sum_{i=1}^{n} \left(h_w(\boldsymbol{x}^{(i)}) - y^{(i)} \right) \boldsymbol{x}^{(i)}. \tag{5}$$

Nuevamente esta actualización es similar, coincidentemente, a la actualización que se realiza para la regresión lineal.

2.4. Justificación Probabilística de la Función de Costo

Debido a que la función sigmoidea utilizada para la regresión logística tiene una salida entre 0 y 1, la hipótesis se puede interpretar como una probabilidad. Así, la probabilidad del clasificador se puede determinar como

$$p(y = 1|\boldsymbol{x}; \boldsymbol{w}) = h_w(\boldsymbol{x})$$
$$p(y = 0|\boldsymbol{x}; \boldsymbol{w}) = 1 - h_w(\boldsymbol{x}).$$

Debido a que $y \in \{0, 1\}$, ambas probabilidades se pueden escribir de manera más compacta como

$$p(y|\boldsymbol{x}; \boldsymbol{w}) = (h_w(\boldsymbol{x}))^y \left(1 - h_w(\boldsymbol{x})\right)^{1-y}. \tag{6}$$

Esta equivalencia se puede fácilmente demostrar reemplazando $y = 1$ y $y = 0$, que son los dos únicos posibles valores de y para el problema de clasificación binaria.

Verosimilitud. La función de *verosimilitud* (en inglés, *likelihood*) es una función de los parámetros $\boldsymbol{w}$ del modelo dados los datos observados. Consiste en ver a la probabilidad $p(y|\boldsymbol{x}; \boldsymbol{w})$ como una función de $\boldsymbol{w}$ manteniendo fijos los demás valores. Así, se define como

$$\mathscr{L}(\boldsymbol{w}) = p(y|\boldsymbol{x}; \boldsymbol{w}).$$

De forma muy poco rigurosa, representa la probabilidad de que los parámetros $\boldsymbol{w}$ modelen adecuadamente los datos observados; en este caso, la relación entre $\boldsymbol{x}$ y y. Se puede extender la función de verosimilitud a todas las instancias del conjunto de entrenamiento multiplicando las respectivas probabilidades:

$$\mathscr{L}(\boldsymbol{w}) = \prod_{i=1}^{n} p(y^{(i)}|\boldsymbol{x}^{(i)}; \boldsymbol{w}),$$

Reemplazando (6) en esta función de verosimilitud de todas las instancias se obtiene

$$\mathscr{L}(\boldsymbol{w}) = \prod_{i=1}^{n} \left(h_w(\boldsymbol{x}^{(i)})\right)^{y^{(i)}} \left(1 - h_w(\boldsymbol{x}^{(i)})\right)^{1-y^{(i)}}.$$

Por facilidad de cálculo, para reemplazar los productos por sumas, se suele tomar el logaritmo natural de esta función, obteniendo

$$\log \mathscr{L}(\boldsymbol{w}) = \sum_{i=1}^{n} y^{(i)} \log(h_w(\boldsymbol{x}^{(i)})) + (1 - y^{(i)}) \log(1 - h_w(\boldsymbol{x}^{(i)})). \tag{7}$$

Dado que la función logaritmo es monótona, su aplicación no genera inconvenientes al momento de tratar de maximizar la función de verosimilitud.

Máxima Verosimilitud. Para obtener los mejores parámetros $\boldsymbol{w}$, que estadísticamente modelen mejor los datos observados, se debe maximizar la función de verosimilitud, a lo que se conoce como el estimador de *máxima verosimilitud* (en inglés, *maximum likelihood*). Debido a que el logaritmo es monótono, resulta conceptualmente equivalente maximizar $\mathscr{L}(\boldsymbol{w})$ o maximizar su logaritmo $\log \mathscr{L}(\boldsymbol{w})$. Se puede observar que el logaritmo de la verosimilitud, dado en (7), es el negativo de la función de costo (3), escalado por una constante. Por este motivo, maximizar la función de verosimilitud, o su logaritmo $\log \mathscr{L}(\boldsymbol{w})$, es equivalente a minimizar la función de costo $J(\boldsymbol{w})$ definida anteriormente. En conclusión, el estimador de máxima verosimilitud contiene parámetros $\boldsymbol{w}$ que se pueden igualmente obtener minimizando la función de costo $J(\boldsymbol{w})$.

Nota: la función de costo $J(\boldsymbol{w})$ para la regresión lineal también maximiza la verosimilitud asumiendo una distribución normal del error de predicción (considerando que las variables son independientes e idénticamente distribuidas).

3. Método de Newton-Raphson para Minimización

El método de Newton-Raphson, a veces llamado solamente método de Newton, se utiliza para encontrar los puntos en los que una función se hace cero. Su aplicación para la minimización se da encontrando los ceros de la derivada de una función, ya que los puntos donde esta derivada se hace cero indican un mínimo (o máximo) local de la función original.

3.1. Para Funciones de una Variable

Primero se considerará el caso de una función real de variable real $f(w) : \mathbb{R} \to \mathbb{R}$.

Ceros de la función. El problema consiste en encontrar w tal que $f(w) = 0$. En este caso se comienza con un valor inicial w_0. Luego se calcula la pendiente $f'(w_0)$ en este punto. Si se prolonga esta pendiente, intersecará al eje horizontal en w_1, de tal modo que la pendiente se podrá definir como

$$f'(w_0) = \frac{f(w_0)}{\Delta}$$

donde $\Delta = w_0 - w_1$. Reordenando los términos se llega a $w_1 = w_0 - \frac{f(w_0)}{f'(w_0)}$. Luego, utilizando este valor de w_1 se puede repetir el proceso hasta llegar a un valor suficientemente cercano a cero. Generalizando esta idea, para encontrar el valor de w que hace cero a la función

$f(w)$, se comienza con un valor inicial w_0, y luego se itera, de tal modo que la iteración $k+1$ queda dada por

$$w_{k+1} = w_k - \frac{f(w_k)}{f'(w_k)}.$$

La iteración terminará cuando de verifique que $f(w_{k+1})$ se encuentra lo suficientemente cercana a cero, siendo w_{k+1} la solución.

Mínimo de la función. El método de Newton también puede ser utilizado para minimizar una función $f(w) : \mathbb{R} \to \mathbb{R}$. En este caso, el mínimo (en realidad, el extremo de la función que también podría ser un máximo) corresponde a puntos donde la derivada se hace cero; es decir, $f'(w) = 0$. Por tanto, se usa el método de Newton para calcular los valores que hacen cero a la derivada $f'(w)$. Al igual que en el caso anterior, se inicia con un valor inicial w_0 y se itera de tal modo que la iteración $k+1$ está dada por

$$w_{k+1} = w_k - \frac{f'(w_k)}{f''(w_k)}.$$

El valor w_{k+1} que lleve a $f''(w_{k+1})$ a cero será un punto crítico, donde podría haber un mínimo o un máximo (en esta parte se considera solo mínimos debido a que las funciones que se utiliza son convexas).

3.2. Para Funciones de Varias Variables

El cálculo iterativo del mínimo de una función real de variable real anterior usando el método de Newton puede ser generalizado para el caso de una función real de varias variables, como la función de costo $J(\boldsymbol{w}) : \mathbb{R}^n \to \mathbb{R}$. En este caso, la derivada está dada por el gradiente $\nabla_{\boldsymbol{w}} J(\boldsymbol{w})$ y la segunda derivada está dada por la matriz Hessiana $H(\boldsymbol{w})$ cuyos elementos h_{ij} son

$$h_{ij} = \frac{\partial^2 J(\boldsymbol{w})}{\partial w_i \partial w_j}.$$

Así, el método iterativo consiste en comenzar con valor inicial $\boldsymbol{w}_0$ e ir iterando, de tal modo que la iteración $k+1$ está dada por

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - H(\boldsymbol{w}_k)^{-1} \nabla_{\boldsymbol{w}} J(\boldsymbol{w}).$$

La ventaja de utilizar este método, con respecto al descenso de gradiente (de tipo *batch*) consiste en su mayor rapidez de convergencia. Sin embargo, debido al cálculo de la matriz Hessiana, y a su inversión, una iteración de este método puede ser más computacionalmente costosa que una iteración del descenso del gradiente.

A la aplicación del método de Newton-Raphson a la regresión logística se le suele denominar el método *Fisher Scoring.*

4. Clasificación Multiclase

La regresión logística se aplica directamente para clasificar dos clases. La forma más sencilla de extender la clasificación a más de dos clases consiste en entrenar un clasificador para cada una de las clases, y luego aplicar dicho clasificador a cada dato de entrada, escogiendo la clase cuyo valor de hipótesis se encuentre más cercano a 1. A esto se conoce como el método *one vs all*, uno contra todos, o uno contra el resto. En esta forma de clasificación, dadas K clases, se define un clasificador $h_w^{(k)}(\boldsymbol{x}) = p(y = k|\boldsymbol{x}; \boldsymbol{w})$ para cada

clase $k = \{1, 2, \cdots, K\}$. Luego, dado un valor de entrada $\boldsymbol{x}$, la predicción será la clase j tal que

$$j = \arg\max_k h_w^{(k)}(\boldsymbol{x}).$$

Es decir, se busca la clase j, de entre todas las clases posibles, cuya predicción sea la más confiable (más cercana a 1).

Por otro lado, es posible generalizar la regresión logística a varias clases. A esto se suelde denominar *regresión logística multinomial* o *multiclase*. Otros nombres comunes son *regresión softmax* o *clasificador softmax*. A continuación se brindará detalles sobre la función de hipótesis y la función de costo para la regresión logística multinomial.

4.1. Función de Hipótesis

En la clasificación multiclase se considerará que existen en total K clases, de tal modo que la salida toma algún valor entre 1 y K; es decir, $y = \{1, 2, \cdots, K\}$. Al igual que para la regresión logística, se asumirá que el vector de entrada $\boldsymbol{x}$ tiene d atributos, tal que $\boldsymbol{x} \in R^d$. Sin embargo, por facilidad, se adiciona un atributo $x_0 = 1$, quedando el vector de entrada de tamaño $\boldsymbol{x} \in R^{d+1}$. En este caso, los parámetros asociados con la clase k se denotarán por $\boldsymbol{w}_k \in \mathbb{R}^{d+1}$.

En regresión multinomial, dada una entrada $\boldsymbol{x}$, la probabilidad de que su salida asociada y sea de la clase k está dada por

$$p(y = k|\boldsymbol{x}; \boldsymbol{w}) = \frac{e^{\boldsymbol{w}_k^T \boldsymbol{x}}}{\sum_{j=1}^{K} e^{\boldsymbol{w}_j^T \boldsymbol{x}}}, \tag{8}$$

donde la sumatoria $\sum_{j=1}^{K} e^{\boldsymbol{w}_j^T \boldsymbol{x}}$ se realiza para todas las clases. El uso de esta sumatoria es únicamente para normalizar la distribución, de tal modo que las probabilidades de todas las clases sumen 1. De manera matemática, debido a esta normalización, se tendrá que

$$p(y = 1|\boldsymbol{x}; \boldsymbol{w}) + p(y = 2|\boldsymbol{x}; \boldsymbol{w}) + \cdots + p(y = K|\boldsymbol{x}; \boldsymbol{w}) = 1.$$

Esto, a su vez, muestra que en realidad hay una restricción, y por tanto una de las clases queda completamente definida si se predicen las otras $K - 1$ clases. A la función descrita por (8) se le conoce como *softmax*, y por eso a esta forma de clasificación también se le conoce con el nombre de regresión softmax.

La función de hipótesis h_w para este caso multiclase, se define como un vector que contiene cada una de las probabilidades de que la salida pertenezca a alguna de las clases; es decir,

$$h_w(\boldsymbol{x}) = \begin{bmatrix} p(y = 1|\boldsymbol{x}; \boldsymbol{w}) \\ p(y = 2|\boldsymbol{x}; \boldsymbol{w}) \\ \vdots \\ p(y = K|\boldsymbol{x}; \boldsymbol{w}) \end{bmatrix}$$

Utilizando esta definición de la función de hipótesis, se puede reemplazar la probabilidad de cada clase, dada por (8), para obtener

$$h_w(\boldsymbol{x}) = \frac{1}{\sum_{j=1}^{K} e^{\boldsymbol{w}_j^T \boldsymbol{x}}} \begin{bmatrix} e^{\boldsymbol{w}_1^T \boldsymbol{x}} \\ e^{\boldsymbol{w}_2^T \boldsymbol{x}} \\ \vdots \\ e^{\boldsymbol{w}_K^T \boldsymbol{x}} \end{bmatrix} \tag{9}$$

donde cada parámetro $\boldsymbol{w}_j$ se asocia a la clase j.

4.2. Función de Costo

Antes de definir la función de costo, es necesario definir el *corchete de Iverson* $[\![P]\!]$, también llamado *función indicatriz* $\mathbb{1}(P)$, como

$$[\![P]\!] = \mathbb{1}(P) = \begin{cases} 1, & \text{si } P \text{ es verdadero} \\ 0, & \text{si } P \text{ es falso.} \end{cases}$$

Por ejemplo, la expresión $[\![y^{(i)} = k]\!]$ será igual a 1 solo cuando $y^{(i)}$ sea k (pertenezca a la clase k); de lo contrario será 0. Utilizando esta notación, se define la función de costo multinomial como

$$J(\boldsymbol{w}) = -\sum_{i=1}^{n}\sum_{k=1}^{K}[\![y^{(i)} = k]\!]\log(p(y^{(i)} = k|\boldsymbol{x}^{(i)};\boldsymbol{w})).$$

Notar que la sumatoria externa se aplica a todas las instancias del conjunto de entrenamiento, y la sumatoria interna a cada una de las clases. De este modo, debido al uso del corchete de Iverson, para cada instancia solamente uno de los elementos de la sumatoria interna será diferente de cero: aquél que satisfaga $y^{(i)} = k$. Utilizando (8), la función de costo se puede escribir de manera explícita como

$$J(\boldsymbol{w}) = -\sum_{i=1}^{n}\sum_{k=1}^{K}[\![y^{(i)} = k]\!]\log\frac{e^{\boldsymbol{w}_k^T\boldsymbol{x}^{(i)}}}{\sum_{j=1}^{K}e^{\boldsymbol{w}_j^T\boldsymbol{x}^{(i)}}}.$$

La minimización de esta función de costo se realiza utilizando algún método de optimización, como el descenso del gradiente que fue utilizado para la regresión lineal y logística. En este caso, el gradiente con respecto a los parámetros $\boldsymbol{w}_k$ de la función de costo, representado como $\nabla_{\boldsymbol{w}_k}J(\boldsymbol{w})$ es

$$\nabla_{\boldsymbol{w}_k}J(\boldsymbol{w}) = -\sum_{i=1}^{n}\Big([\![y^{(i)} = k]\!] - p(y^{(i)} = k|\boldsymbol{x}^{(i)};\boldsymbol{w})\Big)\,\boldsymbol{x}^{(i)}$$

Se debe considerar que $\nabla_{\boldsymbol{w}_k}J(\boldsymbol{w})$ es un vector, y actualiza a todos los parámetros $\boldsymbol{w}_k$ a la vez. Se debe aplicar este gradiente para cada uno de los parámetros $\boldsymbol{w}_k$ en cada iteración del descenso del gradiente.

4.3. Sobreparametrización de la Regresión Multinomial

Considérese un vector $\boldsymbol{\theta} \in \mathbb{R}^{d+1}$ de la misma dimensión que $\boldsymbol{w}_k$, y supóngase que este vector arbitrario se resta a cada parámetro $\boldsymbol{w}_j$, donde $j = 1, 2, \cdots, K$. Bajo esta suposición, y utilizando (8), la predicción que la salida pertenece a una clase k quedará dada por

$$p(y = k|\boldsymbol{x};\boldsymbol{w}) = \frac{e^{(\boldsymbol{w}_k-\boldsymbol{\theta})^T\boldsymbol{x}}}{\sum_{j=1}^{K}e^{(\boldsymbol{w}_j-\boldsymbol{\theta})^T\boldsymbol{x}}} = \frac{e^{\boldsymbol{w}_k^T\boldsymbol{x}}e^{-\boldsymbol{\theta}^T\boldsymbol{x}}}{\sum_{j=1}^{K}e^{\boldsymbol{w}_j^T\boldsymbol{x}}e^{-\boldsymbol{\theta}^T\boldsymbol{x}}} = \frac{e^{\boldsymbol{w}_k^T\boldsymbol{x}}}{\sum_{j=1}^{K}e^{\boldsymbol{w}_j^T\boldsymbol{x}}}.$$

Esto muestra que a pesar de sustraer $\boldsymbol{\theta}$ de cada $\boldsymbol{w}_k$, las predicciones que realiza la función de hipótesis no se modifican. Debido a este comportamiento, se concluye que los parámetros de la regresión multinomial son redundantes; es decir, para cualquier hipótesis que se tenga, hay múltiples parámetros posibles que darían el mismo resultado. A esto se conoce como la propiedad de *sobreparametrización* de la regresión multinomial.

Utilizando esta propiedad, se puede escoger $\boldsymbol{\theta} = \boldsymbol{w}_K$, convirtiendo al parámetro $\boldsymbol{w}_K$ en $\boldsymbol{w}_K - \boldsymbol{\theta} = \boldsymbol{w}_K - \boldsymbol{w}_K = 0$. El efecto de esta opción de $\boldsymbol{\theta}$ es eliminar al vector $\boldsymbol{w}_K$ sin afectar el resultado que la función de hipótesis predice. Más aún, se puede eventualmente aprovechar esta propiedad para solamente optimizar los parámetros $\boldsymbol{w}_1$ al $\boldsymbol{w}_{K-1}$. Debe resultar claro que en lugar de escoger $\boldsymbol{\theta} = \boldsymbol{w}_K$ se podría escoger cualquier otro parámetro correspondiente a otra clase y el resultado sería similar.

Debido a esta propiedad, si los parámetros $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_K$ minimizan la función de costo $J(\boldsymbol{w})$, los parámetros $\boldsymbol{w}_1 - \boldsymbol{\theta}, \boldsymbol{w}_2 - \boldsymbol{\theta}, \cdots, \boldsymbol{w}_K - \boldsymbol{\theta}$, con cualquier $\boldsymbol{\theta}$ arbitrario, también minimizan a la función de costo. Un detalle de la función de costo es que es convexa a pesar de esta sobreparametrización. Por este motivo, el método del descenso del gradiente no encontrará mínimos locales sino el mínimo global. Sin embargo, el Hessiano de $J(\boldsymbol{w})$ es singular, lo cual conlleva a que una implementación directa del método de Newton-Raphson presente problemas numéricos y, por tanto, no sea recomendable.

4.4. Relación con la Regresión Logística

La regresión logística es un caso especial de la regresión multinomial cuando $K = 2$. En este caso, la función de hipótesis dada en (9) se reduce a

$$h_w(\boldsymbol{x}) = \frac{1}{e^{\boldsymbol{w}_1^T \boldsymbol{x}} + e^{\boldsymbol{w}_2^T \boldsymbol{x}}} \begin{bmatrix} e^{\boldsymbol{w}_1^T \boldsymbol{x}} \\ e^{\boldsymbol{w}_2^T \boldsymbol{x}} \end{bmatrix}.$$

Utilizando la propiedad de la sobreparametrización de la regresión multinomial, se puede escoger $\boldsymbol{\theta} = \boldsymbol{w}_2$, de tal modo que $\boldsymbol{w}_1$ se reemplaza por $\boldsymbol{w}_1 - \boldsymbol{w}_2$ y $\boldsymbol{w}_2$ se reemplaza por un vector de ceros 0. Más aún, para simplificar la notación, se puede utilizar una variable $\boldsymbol{w}$ tal que $\boldsymbol{w} = \boldsymbol{w}_1 - \boldsymbol{w}_2$. Así, la función de hipótesis queda como

$$h_w(\boldsymbol{x}) = \frac{1}{e^{(\boldsymbol{w}_1 - \boldsymbol{w}_2)^T \boldsymbol{x}} + e^{0\boldsymbol{x}}} \begin{bmatrix} e^{(\boldsymbol{w}_1 - \boldsymbol{w}_2)^T \boldsymbol{x}} \\ e^{0\boldsymbol{x}} \end{bmatrix} = \frac{1}{e^{\boldsymbol{w}^T \boldsymbol{x}} + 1} \begin{bmatrix} e^{\boldsymbol{w}^T \boldsymbol{x}} \\ 1 \end{bmatrix}.$$

Separando cada uno de los términos se tiene

$$h_w(\boldsymbol{x}) = \begin{bmatrix} \frac{e^{\boldsymbol{w}^T \boldsymbol{x}}}{e^{\boldsymbol{w}^T \boldsymbol{x}} + 1} \\ \frac{1}{e^{\boldsymbol{w}^T \boldsymbol{x}} + 1} \end{bmatrix} = \begin{bmatrix} 1 - \frac{1}{e^{\boldsymbol{w}^T \boldsymbol{x}} + 1} \\ \frac{1}{e^{\boldsymbol{w}^T \boldsymbol{x}} + 1} \end{bmatrix}.$$

Si se asume que la primera fila predice la clase 0, y la segunda fila la clase 1, se llega exactamente a la misma predicción que se obtiene con la regresión logística. Esto muestra que la regresión multinomial o softmax es una generalización de la regresión logística.