Modelo Lineal de Regresión

Aprendizaje Automático INF-393 II-2018

Ricardo Ñanculef

UTFSM Campus San Joaquín

Table of contents

Regresión en el Modelo Supervisado

Modelo Supervisado

Dados un espacio de entrada $\mathbb{X}$, un espacio de salida $\mathbb{Y}$, una función de pérdida (loss) $L : \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}$, y un conjunto de ejemplos $S = \{(x^{(i)}, y^{(i)})\}_{i=1}^{n} \subset \mathbb{X} \times \mathbb{Y}$, un problema de aprendizaje supervisado consiste en encontrar una función $f : \mathbb{X} \to \mathbb{Y}$ que minimice *el error medio de predicción*:

$$R(f) = \mathbb{E}(L(f(x), y)) = \sum_{x,y} L(f(x), y)p(x, y) \ . \tag{1}$$

Notemos que acá, $p(x, y)$ es la función de probabilidad sobre $\mathbb{X} \times \mathbb{Y}$ con la cuál se generan los ejemplos.

Si no genera confusión escribiremos $\sum_{x,y} L(f(x), y)p(x, y)$ en vez de $\int L(f(x), y)dP(x, y)$ con el fin de hacer la notación más intuitiva.

Modelo Supervisado

Como en la práctica $p(x, y)$ no se conoce, muchos métodos buscan f en modo de minimizar el denominado *error de entrenamiento*:

$$\hat{R}(f) = \frac{1}{n} \sum_{i=1}^{n} L(f(x^{(i)}), y^{(i)}) \,. \tag{2}$$

Cuando un bajo error de entrenamiento no se corresponde con un bajo error de predicción decimos que el método *sobre-ajusta*.

Modelo Supervisado

Para implementar una solución, el learner elige un determinado espacio de soluciones posibles que se denomina *espacio de hipótesis*:

$$\mathcal{H} = \{f(x;\alpha) : \alpha \in \Lambda\} \ . \tag{3}$$

La posibilidad de sobre-ajuste depende fuertemente de la complejidad de $\mathcal{H}$ (dimensión VC) relativa al número de ejemplos disponibles.

Regresión

En general, llamaremos regresión, a cualquier problema en que $\mathbb{Y} \subset \mathbb{R}$ es continuo y denso en $\mathbb{R}^o$. Por simplicidad, asumiremos en adelante que $o = 1$.

Modelo Lineal de Regresión

Asumiendo que los datos han sido representados de modo que $\mathbb{X} \subset \mathbb{R}^d$, un método muy simple para abordar problemas de regresión consiste en elegir como espacio de hipótesis es espacio de todas las "líneas" en $\mathbb{R}^d$, es decir, todas las funciones de la forma

$$f(x; \beta) = \sum_{i=1}^{d} \beta_i x_i + \beta_0 = \beta^T x + \beta_0 \,, \tag{4}$$

y adoptar la función de pérdida denominada *quadratic loss*,

$$L(f(x), y) = (f(x) - y)^2 \,. \tag{5}$$

Modelo Lineal de Regresión

Notemos que con este espacio de hipótesis, nuestras aproximaciones $f(x)$ a la respuesta deseada (y) simplemente "pesan" los distintos atributos que hayamos decidido utilizar para representar x y suman los efectos obtenidos:

$$f(x;\beta) = \sum_{i=1}^{d} \beta_i x_i + \beta_0 = \beta^T x + \beta_0 \,. \tag{6}$$

En otras palabras, el modelo es lineal en β. Notemos sin embargo, que si los datos disponibles (antes de la representación en $\mathbb{X}$), son de la forma $\{(z^{(i)}, y^{(i)})\}_{i=1}^{n}$, el modelo podría no ser lineal en z.

Entrenamiento del Modelo Lineal

Formulación Clásica

En la formulación clásica del modelo lineal, el entrenamiento se traduce en encontrar $\beta_0, \beta_1, \ldots, \beta_n$ en modo de minimizar la siguiente función objetivo

$$J(\beta) = \sum_{i=1}^{n} (y^{(i)} - f(x^{(i)}))^2 \propto \hat{R}(f)\,. \tag{7}$$

Una posibilidad consiste en encontrar un punto crítico de J y chequear que se trata de un mínimo. Esto implica resolver el sistema de ecuaciones

$$\frac{\partial J}{\partial \beta_i} = 0 \,\forall i = 0, \ldots, d\,. \tag{8}$$

Notación Matricial

Para resolver ese sistema, es conviente expresar el problema matricialmente, definiendo las siguientes matrices:

- Las respuestas deseadas $\{y^{(i)}\}_{i=1}^{n}$ se organizan como filas de una matriz $Y \in \mathbb{R}^{n \times 1}$.
- Las incógnitas $\beta_0, \beta_1, \ldots, \beta_n$ se organizan en un vector $\underline{\beta} \in \mathbb{R}^{d+1}$.
- Los ejemplos de entrada $\{x^{(i)}\}_{i=1}^{n}$ se expanden en una dimensión, definiendo $\underline{x}^{(i)} = (x^{(i)}, 1)$.
- Los ejemplos $\{\underline{x}^{(i)}\}_{i=1}^{n}$ se organizan como filas de una matriz $X \in \mathbb{R}^{n \times d+1}$ denominada matriz de diseño.

Por comodidad definimos $p = d + 1$.

Notación Matricial

Notemos que con esas definiciones, las predicciones del modelo correspondientes a los ejemplos de entrada $\{x^{(i)}\}_{i=1}^{n}$ se pueden obtener como

$$\hat{Y} = X\underline{\beta}. \tag{9}$$

La función objetivo toma entonces la forma,

$$J(\underline{\beta}) = \|\hat{Y} - Y)\|^2 = (X\underline{\beta} - Y)^T(X\underline{\beta} - Y)\,. \tag{10}$$

Con esta notación encontrar los puntos críticos de $J(\underline{\beta})$ es sencillo ...

Solución Clásica

En efecto, asumiendo $X^T X \in \mathbb{R}^{p \times p}$ invertible,

$$\frac{\partial J}{\partial \underline{\beta}} = 0 \Leftrightarrow 2X^T(X\underline{\beta} - Y) = 0 \Leftrightarrow \underline{\beta} = (X^T X)^{-1} X^T Y \qquad (11)$$

Además,

$$\frac{\partial^2 J}{\partial \underline{\beta}^2} = 0 \Leftrightarrow 2X^T X \succ 0\,.$$

Lo que muestra que el modelo se puede "entrenar" en tiempo entre cúbico y cuadrático en el número de atributos.

Solución Clásica

Notemos que las predicciones del modelo entrenado se pueden obtener fácilmente como

$$\hat{Y} = X\underline{\beta} = X(X^T X)^{-1} X^T Y = HY \tag{12}$$

con $H = X(X^T X)^{-1} X^T$. Notemos que la matriz H es un proyector, i.e., $H^n = H$. No es difícil mostrar que H proyecta en efecto al sub-espacio generado por las columnas de X, correspondientes a los distintos atributos. Los errores de predicción se obtienen como

$$\hat{Y} - Y = HY - Y = MY\,, \tag{13}$$

donde $M = (I - H)$ es también un proyector, ortogonal a H.

Entrenamiento vía Gradiente

Una forma alternativa de entrenar un learner lineal, consiste en usar un método de optimización ampliamente empleado en machine learning denominado *gradiente descendente*. Dado un criterio de entrenamiento de la forma

$$\min_{\underline{\beta} \in \mathbb{R}^p} J(\underline{\beta}), \tag{14}$$

con $J(\underline{\beta})$ diferenciable, este método genera una sucesión de soluciones $\underline{\beta}^{(0)}, \underline{\beta}^{(1)}, \ldots, \underline{\beta}^{(t)}$ que se obtienen iterando la siguiente regla

$$\underline{\beta}^{(t+1)} = \underline{\beta}^{(t)} - \eta_t \left. \frac{\partial J}{\partial \underline{\beta}} \right|_{\underline{\beta} = \underline{\beta}^{(t)}}, \tag{15}$$

donde $\eta_t \in \mathbb{R}$ se denomina *la tasa de aprendizaje*. Con frecuencia se usa un valor fijo $\eta_t = \eta \in (0, 1)$.

Entrenamiento vía Gradiente

En el caso que estamos estudiando

$$J(\underline{\beta}) = \|X\underline{\beta} - Y)\|^2 = \sum_{i=1}^{n} (\underline{\beta}^T \underline{x}^{(i)} - y^{(i)})^2 \tag{16}$$

de modo que

$$\frac{\partial J}{\partial \underline{\beta}} = 2X^T(X\underline{\beta} - Y) = 2\sum_{i=1}^{n} (\underline{\beta}^T \underline{x}^{(i)} - y^{(i)})\underline{x}^{(i)} = 2\sum_{i=1}^{n} e^{(i)}\underline{x}^{(i)} \tag{17}$$

donde $e^{(i)} = (y^{(i)} - f(x^{(i)}))$ es el error de predicción.

Entrenamiento vía Gradiente

Combinando estos cálculos con el algoritmo genérico, obtenemos que una forma alternativa de entrenar el modelo lineal consiste en iterar la siguiente regla

$$\underline{\beta}^{(t+1)} = \underline{\beta}^{(t)} - \eta_t \sum_{i=1}^{n} e^{(i)} \underline{x}^{(i)} , \tag{18}$$

que, en cada iteración, mueve el modelo hacia los ejemplos que se están prediciendo con mayor error $e^{(i)}$.

Gradiente versus Solución Clásica

- Notemos que una iteración del algoritmo anterior es lineal en el número de atributos y lineal en el número de ejemplos.
- La solución clásica es en cambio cuadrática en el número de datos y prácticamente cúbica en el número de atributos.
- La ventaja de entrenar via gradiente es entonces que en problemas con muchos atributos podemos obtener rápidamente una solución aproximada al problema, que se va refinando gradualmente y que podemos detener en el caso de superar un límite de tiempo.

Gradiente Estocástico

Una variante aún más flexible del método del gradiente, consiste en usar lo que se denomina *gradiente descendente estocástico*.

Supongamos que en cada iteración t, construimos una aproximación $\tilde{G}_t$ del verdadero gradiente $G_t = \partial J / \partial \underline{\beta}\big|_{\underline{\beta}^{(t)}}$ tal que

$$\mathbb{E}(\tilde{G}_t) = G_t \tag{19}$$

Bajo condiciones razonables, es posible demostrar que si iteramos la regla

$$\underline{\beta}^{(t+1)} = \underline{\beta}^{(t)} - \eta_t \tilde{G}_t \,, \tag{20}$$

obtenemos una sucesión convergente al óptimo de la función objetivo.

Gradiente Estocástico

En nuestro problema, debiésemos aproximar el gradiente

$$G_t = \frac{\partial J}{\partial \underline{\beta}} = 2 \sum_{i=1}^{n} e^{(i)} \underline{x}^{(i)} \tag{21}$$

Una forma sencilla se hacerlo es elegir B ejemplos de entrenamiento al azar $\{(x^{(i^*)}, y^{(i^*)})\}_{i^*=1}^{B}$ y estimar el gradiente como

$$\tilde{G}_t = \frac{2n}{B} \sum_{i^*=1}^{B} e^{(i^*)} x^{(i^*)} \tag{22}$$

No es difícil demostrar que:

$$\mathbb{E}(\tilde{G}_t) = G_t. \tag{23}$$

Gradiente Estocástico versus Solución Clásica

- Notemos que una iteración del algoritmo anterior es lineal en el número de atributos e independiente del número de ejemplos.
- La solución clásica es en cambio cuadrática en el número de datos y prácticamente cúbica en el número de atributos.
- La ventaja de entrenar via gradiente estocástico es entonces que en problemas con muchísimos datos y muchos atributos podemos obtener rápidamente una solución aproximada al problema, refinarla gradualmente, y eventualmente detener el entrenamiento con una solución válida si el tiempo disponible para entrenamiento se agota.

¿Porqué o Cuándo usar la Función de Pérdida Cuadrática?

Tipos de Error en Aprendizaje

En todo método de aprendizaje desde ejemplos, hay que tener 3 fuentes de error en consideración

1. **Error de Estimación**: La diferencia entre lo que aprenderíamos de un conjunto infinito de ejemplos versus lo que aprendemos desde un conjunto finito de datos.
2. **Error de Aproximación**: La diferencia entre lo que aprenderíamos en un espacio de hipótesis sin límites (todas las funciones de $\mathbb{X}$ a $\mathbb{Y}$) versus lo que podemos aprender en el espacio de hipótesis seleccionado.
3. **Error de Optimización**: La diferencia entre el verdadero óptimo del criterio de entrenamiento y aquello logramos encontrar con nuestro algoritmo de entrenamiento (quizás aproximado o truncado).

Análisis de la Pérdida Cuadrática

Para analizar el comportamiento de una determinada función de pérdida, es útil ignorar estos errores y pensar qué obtendríamos con dicha función si pudiésemos buscar en un espacio de hipótesis ilimitado, con conocimiento perfecto de $p(x, y)$ (infinitos ejemplos) y con un solver óptimo.

En nuestro caso, nos interesa saber qué obtendríamos usando la pérdida cuadrática si pudiésemos minimizar

$$\mathbb{E}(L(f(x), y)) = \sum_{x,y} (f(x) - y)^2 p(x, y)\,. \tag{24}$$

Descomposición del Error Cuadrático

Sea $r(x) = \mathbb{E}(y|x) = \sum_y y\, p(y|x)$. Notemos que

$$\sum_{x,y}(f(x)-y)^2 p(x,y) = \sum_{x,y}(f(x)-r(x)+r(x)-y)^2 p(x,y) \quad (25)$$

$$= \sum_{x,y}(f(x)-r(x))^2 p(x,y) + \sum_{x,y}(r(x)-y)^2 p(x,y)$$

$$-2\sum_{x,y}(f(x)-r(x))(r(x)-y)p(x,y)\,.$$

Además,

$$\sum_{x,y}(f(x)-r(x))(r(x)-y)p(x,y) = \sum_{x,y}(f(x)-r(x))(r(x)-y)p(y|x)p(x)$$

$$= \sum_x (f(x)-r(x))p(x)\left(\sum_y (r(x)-y)p(y|x)\right)$$

$$= \sum_x (f(x)-r(x))p(x)\left(r(x)-\sum_y y\, p(y|x)\right)$$

$$= \sum_x (f(x)-r(x))p(x)\,(r(x)-r(x)) = 0\,.$$

Descomposición del Error Cuadrático

Entrenar $f(x)$ con la pérdida cuadrática es entonces equivalente a minimizar

$$\sum_{x,y}(f(x)-r(x))^2 p(x,y)+\sum_{x,y}(r(x)-y)^2 p(x,y)\,. \tag{26}$$

Sin embargo, $r(x)$ no depende de $f(x)$, por lo entrenar $f(x)$ con la pérdida cuadrática es equivalente a minimizar

$$R(f)=\sum_{x,y}(f(x)-r(x))^2 p(x,y)=\sum_{x}\sum_{y}(f(x)-r(x))^2 p(y|x)p(x) \tag{27}$$

$$=\sum_{x}(f(x)-r(x))^2 p(x)\,. \tag{28}$$

Es claro que $f(x)=r(x)=\mathbb{E}(y|x)$ es un mínimo de $R(f)$.

Conclusión

¿Porqué o cuándo usar la función de pérdida cuadrática?

Entrenar con la pérdida cuadrática tiene sentido cuando nos interesa aproximar el valor esperado de la respuesta (y) frente a un determinado input x.

Ejemplo

Supongamos que deseamos aprender la demanda por bicicletas en Ñuñoa a partir de una serie de atributos que incluyen tipo de día (feriado o laboral), hora del día y posición geográfica.

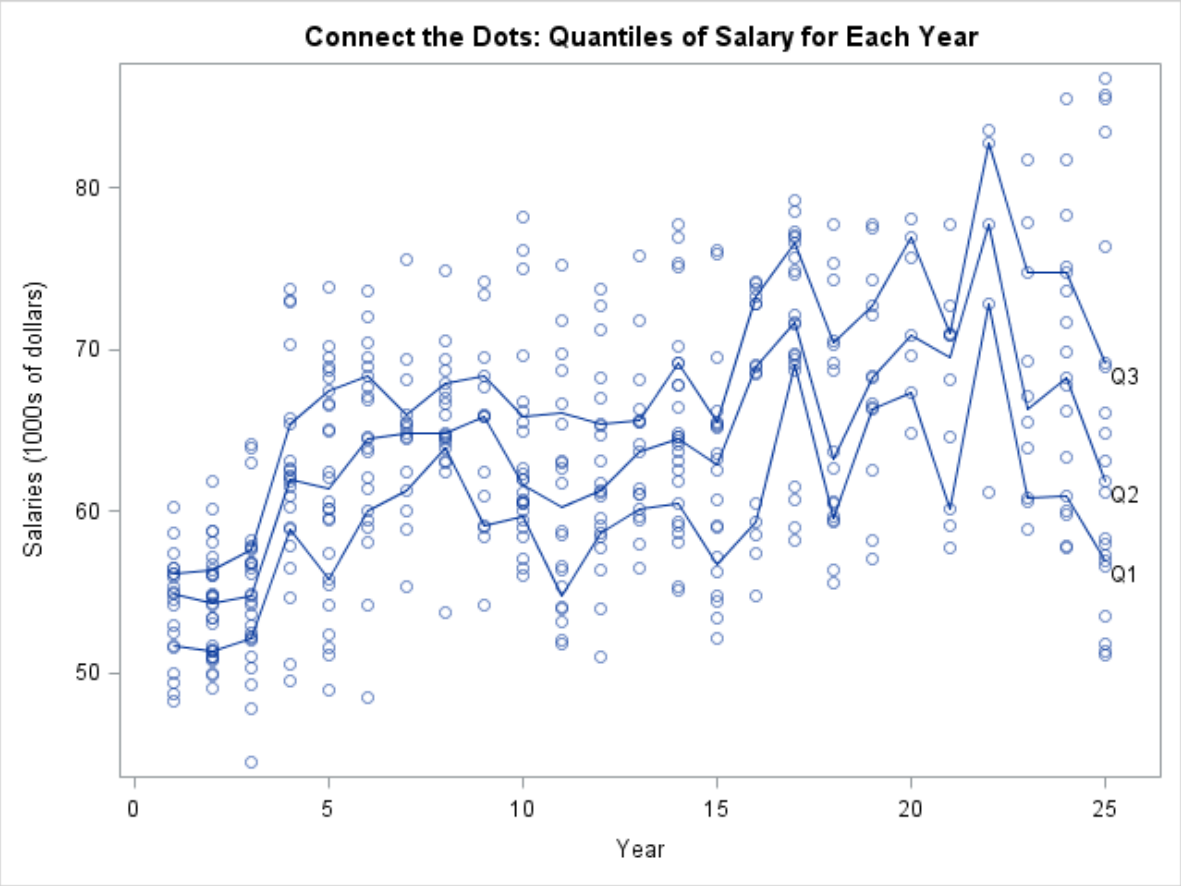
Conclusión

Ejemplo

- Es claro que para una misma configuración de atributos (tipo de día, hora del día y posición geográfica) la demanda puede variar.
- Lo que sucede es que y es una variable aleatoria que se mantiene aleatoria aún si conocemos x, es decir, existe una determinada distribución de probabilidad $p(y|x)$.
- Aproximar $\mathbb{E}(y|x)$ consiste en aproximar la demanda media de bicicletas para un tipo de día, hora del día y posición geográfica. En otras palabras, el valor esperado de $p(y|x)$.

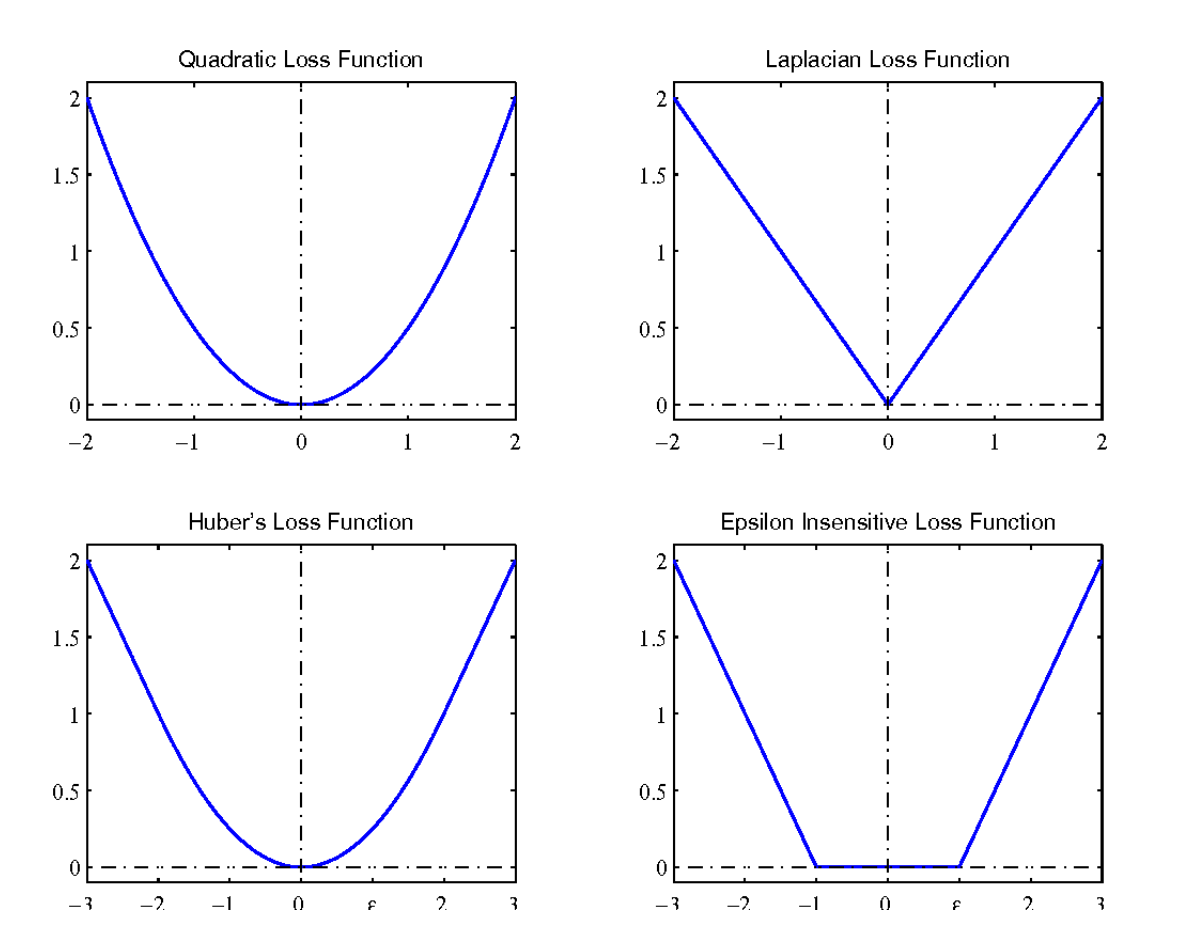
Quantile Regression

Tarea: ¿Cómo modificar la función de costo si nos interesa aproximar un quantil de $y|x$ en vez de la media?



Otras Funciones de Pérdida

Aún si decidimos modelar la media de $y|x$, podría no ser conveniente emplear la función de pérdida cuadrática.



Punto de Vista Clásico

El punto de vista clásico (como se presenta el modelo lineal en textos de estadística) consiste en adoptar el supuesto de que la v.a. y se puede descomponer aditivamente de la forma

$$y = f^*(x) + \epsilon \tag{29}$$

donde $f^*(x)$ es una función determinista de x, desconocida, que se busca aproximar, y ϵ es una v.a. independiente de x. A continuación, se asume que $\mathbb{E}(\epsilon) = 0$. Este supuesto lleva a concluir que

$$\mathbb{E}(y|x) = f^*(x)\,, \tag{30}$$

es decir, aquello que se busca aproximar es justamente el valor esperado de y dado x.

Normalidad

Con frecuencia, la justificación clásica sobre el uso de la función cuadrática en el modelo lineal requiere suponer que $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Notemos que en este caso, se obtiene inmediatamente que

$$y|x \sim \mathcal{N}(f^*(x), \sigma^2)\,. \tag{31}$$

Es decir, modelo lineal $f(x; \underline{\beta})$ está aproximando el valor esperado de la distribución de y condicional a x.

Máxima Verosimilitud

Este punto de vista es interesante porque permite demostrar que entrenar el modelo lineal utilizando la función cuadrática es equivalente a maximizar la función de verosimilitud condicional $\mathcal{L}(\underline{\beta}) = P(\underline{Y}|\underline{\beta}, \underline{X})$, donde $\underline{X} = \{x^{(i)}\}_{i=1}^{n}$ e $\underline{Y} = \{y^{(i)}\}_{i=1}^{n}$.

En efecto, asumiendo que los ejemplos son independientes

$$\mathcal{L}(\underline{\beta}) = \prod_{i=1}^{n} p(y^{(i)}|x^{(i)}, \underline{\beta}) \,. \tag{32}$$

Como $y^{(i)}|x^{(i)} \sim \mathcal{N}(f(x^{(i)}; \underline{\beta}), \sigma^2)$, obtenemos que

$$\mathcal{L}(\underline{\beta}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ \frac{-(y^{(i)} - f(x^{(i)}; \underline{\beta}))^2}{2\sigma^2} \right\} . \tag{33}$$

Máxima Verosimilitud

Se sigue fácilmente, que la función de log-verosimilitud toma la forma

$$\ell(\underline{\beta}) = \sum_{i=1}^{n} \log p(y^{(i)}|x^{(i)}, \underline{\beta}) \tag{34}$$
$$= -\sum_{i=1}^{n} \frac{-(y^{(i)} - f(x^{(i)}; \underline{\beta}))^2}{2\sigma^2} - \log \sqrt{2\pi\sigma^2}\,.$$

Como σ^2 es independiente de $\underline{\beta}$, obtenemos que maximizar la verosimilitud es equivalente a minimizar

$$J(\underline{\beta}) = \sum_{i=1}^{n} (y^{(i)} - f(x^{(i)}; \underline{\beta}))^2\,. \tag{35}$$

Conclusión

¿Porqué o cuándo usar la función de pérdida cuadrática?

Si asumimos un modelo aditivo $y = f^*(x) + \epsilon$ con ruido gausiano $\epsilon \sim \mathcal{N}(0, \sigma^2)$, entrenar el modelo lineal es equivalente a maximizar la función de verosimilitud (condicional) $\mathcal{L}(\underline{\beta}) = P(\underline{Y}|\underline{\beta}, \underline{X})$.

Importante

Si el ruido ϵ sigue otra distribución de probabilidad y se desea entrenar el modelo para maximizar la función de verosimilitud, la función de pérdida a utilizar podría no ser la pérdida cuadrática.

Conclusión

Ejemplo

Si ϵ sigue una distribución de Laplace

$$p(\epsilon) = \frac{1}{2\sigma} \exp\left\{ \frac{-|\epsilon|}{\sigma} \right\}, \tag{36}$$

y se desea entrenar el modelo para maximizar la función de verosimilitud, la función de pérdida a utilizar debe ser

$$L(f(x), y) = |f(x) - y|. \tag{37}$$

Interpretación del Modelo e Inferencias Clásicas

Interpretación del Modelo e Inferencias Clásicas

En preparación. Favor ver apuntes en moodle.

Resumen

1. Regresión en el Modelo Supervisado
2. Entrenamiento del Modelo Lineal
3. Discusión sobre la Pérdida Cuadrática
4. Interpretación del Modelo e Inferencias Clásicas