Regresión lineal por gradiente descendente

Gamaliel Moreno Chávez

MCPI

Evaluación
2021

Contenido

- Aprendizaje supervisado
- Regresión lineal
- Descenso por gradiente

Aprendizaje supervisado

Aprendizaje supervisado. métodos entrenados con:

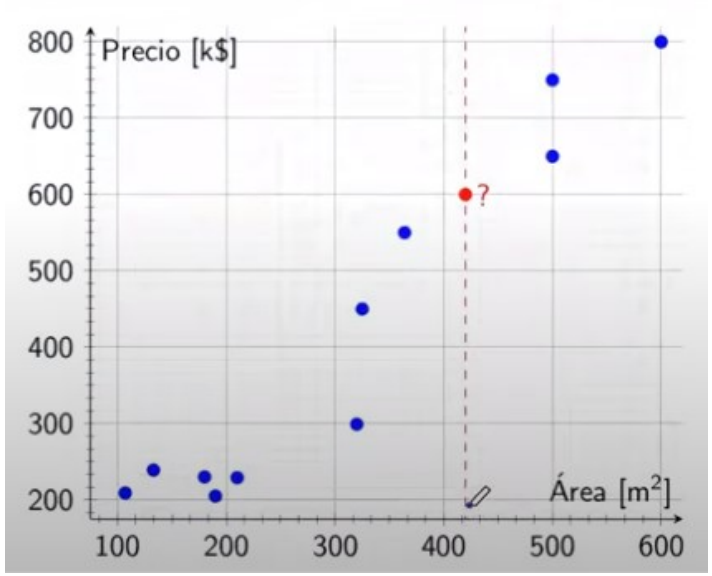
- Conjunto de entrenamiento con pares ordenados $(\mathbf{x}^{(i)}, y^{(i)})$
- $\mathbf{x}^{(i)}$ es el i-ésimo vector de entrada
- $y^{(i)}$ es la correspondiente etiqueta (label) correcta que se desea predecir posteriormente
- Descenso por gradiente

Es el tipo de aprendizaje más común.

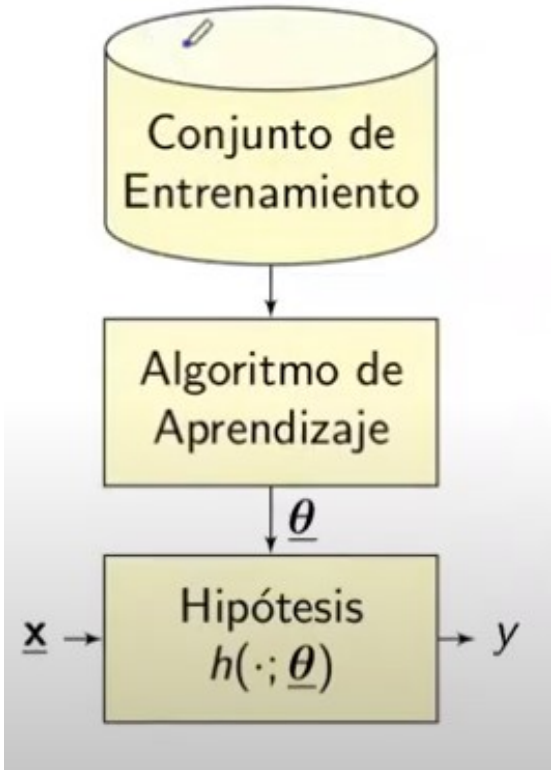
Problema

Dado un conjunto de entrenamiento como el que se muestra en la figura, ¿Cómo se puede encontrar la relación de salida en términos de la entrada?

Área [m^2]	Plantas	Hab.	Precio [k$]
600	3	5	800
190	2	2	205
210	2	2	229
364	2	2	550
325	2	4	450
180	2	2	230
133	2	2	239
500	2	3	650
107	1	2	209
320	2	3	299
500	2	4	750



Notación y modelo supervisado



- m: números de muestras de entrenamiento
- $\mathbf{x}$: datos de entrada
- n : dimensión de la entrada $\mathbf{x}$ (número de características)
- y: variable de salida u objetivo (target)
  - Clasificación: $y \in \{C_1, \ldots, C_k\} k \in \mathbb{N}$
  - Regresión: $y \in \mathbb{R}$
- $(\mathbf{x}^{(i)}, y^{(i)})$: i-ésima muestra de entrenamiento
- $\boldsymbol{\theta}$: parámetros
- $y = h(\mathbf{x}; \boldsymbol{\theta})$: hipótesis

Hipótesis para regresión lineal

- Ejemplo de hipótesis: regresión lineal

$$y = h(\mathbf{x}; \boldsymbol{\theta}) = h_{\boldsymbol{\theta}}(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n$$

- Ejemplo con n=3
  - x_1: área de casa
  - x_2: # de habitaciones
  - x_3: # de pisos
- Convención para simplificar notación: $x_0 = 1$

$$y = h(\mathbf{x}; \boldsymbol{\theta}) = h_{\boldsymbol{\theta}}(\mathbf{x}) = \theta_0 x_0 + \theta_1 x_1 + \cdots + \theta_n x_n = \sum_{i=0}^{n} \theta_i x_i$$

$$= \boldsymbol{\theta}^T \mathbf{x} = \langle \boldsymbol{\theta}, \mathbf{x} \rangle = \boldsymbol{\theta} \cdot \mathbf{x}$$

$$\boldsymbol{\theta} = [\theta_0, \theta_1, \ldots, \theta_n]^T$$

$$\boldsymbol{x} = [x_0, x_1, \ldots, x_n]^T$$

Función objetivo y minimización de cuadrados

Para encontrar $\boldsymbol{\theta}$ minimizamos la función de error $J(\boldsymbol{\theta})$ con

$$J(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{m}(h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) - y^{(i)})^2$$

- $r^i = h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) - y^{(i)}$ se le llama residuo
- El factor $\frac{1}{2}$ se coloca por conveniencia
- Planteamos problemas de optimización de mínimos cuadrados ordinarios

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \boldsymbol{\theta}$$

Objetivo: Se buscan parámetros $\boldsymbol{\theta}$ que producen el menor valor de $\boldsymbol{\theta}$

Ejemplo de regresión de precios de casa

El caso general de regresión **lineal** minimiza entonces a

$$J(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{m}(h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) - y^{(i)})^2$$

$$= \frac{1}{2}\sum_{i=1}^{m}(\boldsymbol{\theta}^T\mathbf{x} - y^{(i)})^2$$

y para el caso de precio= f(area)

$$J(\theta_0, \theta_1) = \frac{1}{2}\sum_{i=1}^{m}(\theta_0 + \theta_1 x_1^{(i)} - y^{(i)})^2$$

Minimización de la función objetivo

- Hay varias posibilidades para minimizar $J(\boldsymbol{\theta})$
- En general, las técnicas de aprendizaje
  - Toma un valor inicia de $\boldsymbol{\theta}$
  - Modifican iterativamente $\boldsymbol{\theta}$ para reducir $J(\boldsymbol{\theta})$

Gradiente descendente

Caso particular descenso por gradiente

1. Tome un valor $\boldsymbol{\theta}^{(0)}$ inicial, t=0
2. Calcule en $\boldsymbol{\theta}^{(t)}$ el gradiente (máxima dirección de cambio)

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}^{(t)}) = \left[\frac{\partial J}{\partial \theta_0} \frac{\partial J}{\partial \theta_1} \cdots \frac{\partial J}{\partial \theta_0} \right]^T$$

3. Calcule la nueva posición

$$\boldsymbol{\theta}^{(t+1)} := \boldsymbol{\theta}^{(t)} - \alpha \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}^{(t)})$$

o de forma equivalente para cada $\theta_j, j \in 1, \ldots, n$

$$\theta_j^{(t+1)} := \theta_j^{(t)} - \alpha \frac{\partial J(\boldsymbol{\theta}^{(t)})}{\partial \theta_j}$$

Estrategias de parada

Para detener la búsqueda de mínimo:

- Usualmente se utilizan tasas de cambio
- Primera opción : $J(\boldsymbol{\theta}^t) - J(\boldsymbol{\theta}^{t+1}) < \epsilon$
- segunda opción : $||\boldsymbol{\theta}^t - \boldsymbol{\theta}^{t+1}|| < \epsilon$
- tercera opción : Número máximo de iteraciones
- opción usual : combinación de anteriores

Normalización de datos

- Si el gradiente es fuertemente asimétrico (como en el caso actual), la tasa de aprendizaje α debe elegirse muy pequeña y proceso necesitará demasiadas iteraciones para converger
- Datos deben normalizarse (preprocesamiento) para evitar estos problemas

Descenso de gradiente para regresión lineal 1

Partiendo del caso concreto

$$J(\theta_0, \theta_1) = \frac{1}{2} \sum_{i=1}^{m} (\theta_0 + \theta_1 x_1^{(i)} - y^{(i)})^2$$

podemos calcular el gradiente fácilmente

$$\nabla_{\boldsymbol{\theta}} J(\theta_0, \theta_1) = \begin{bmatrix} \frac{\partial J(\theta_0, \theta_1)}{\partial \theta_0} \\ \frac{\partial J(\theta_0, \theta_1)}{\partial \theta_1} \end{bmatrix},$$

$$= \begin{bmatrix} \sum_{i=1}^{m} (\theta_0 + \theta_1 x_1^{(i)} - y^{(i)}) \cdot 1 \\ \sum_{i=1}^{m} (\theta_0 + \theta_1 x_1^{(i)} - y^{(i)}) \cdot x_1^{(i)} \end{bmatrix}$$

Descenso de gradiente para regresión lineal 2

Observe que para el caso general de regresión lineal se tiene

$$J(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{m}(\boldsymbol{\theta}^T \boldsymbol{x}^{(i)} - y^{(i)})^2$$
$$= \frac{1}{2}\sum_{i=1}^{m}\left((\theta_0 x_0 + \theta_1 x_1 + \cdots + \theta_n x_n) - y^{(i)}\right)^2$$

La j-ésima componente del gradiente $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$ es

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} = \sum_{i=1}^{m}(\boldsymbol{\theta}^T \boldsymbol{x}^{(i)} - y^{(i)}) \cdot x_j^{(i)}$$

Descenso de gradiente para regresión lineal 3

Lo que finalmente implica que el gradiente es

$$\nabla J(\boldsymbol{\theta}) = \sum_{i=1}^{m} (\boldsymbol{\theta}^T \boldsymbol{x}^{(i)} - y^{(i)}) \cdot \boldsymbol{x}^{(i)}$$

Descenso de gradiente por lotes

Combinando todos los resultados anteriores tenemos el algoritmo de descenso de gradiente por lotes

1. Inicialice $t := 0$ y $\boldsymbol{\theta}^{(0)}$
2. **repeat**
3. $\boldsymbol{\theta}^{(t+1)} := \boldsymbol{\theta}^{(t)} - \alpha \sum_{i=1}^{m} (\boldsymbol{\theta^{(t)}}^T \boldsymbol{x}^{(i)} - y^{(i)}) \cdot \boldsymbol{x}^{(i)}$
4. $t := t + 1$
5. **go to 2 until converge**

- Lotes: cada paso usa todo el conjunto de entrenamiento
- α tasa de aprendizaje, su ajuste es delicado
  - Si α es muy grande oscila alrededor del mínimo
  - Si α es muy pequeña, necesita muchos pasos para converger
- El descenso de gradiente converge a extremos locales, que dependen del punto inicial

Descenso por gradiente estocástico

Descenso por gradiente estocástico o incremental usa un ejemplo del conjunto de entrenamiento a la vez:

1. Inicialice $t := 0$ y $\boldsymbol{\theta}^{(0)}$
2. **repeat**
3. **for each** $(\boldsymbol{x}^{(i)}, y^{(i)})$ in training set do
4. $\boldsymbol{\theta}^{(t+1)} := \boldsymbol{\theta}^{(t)} - \alpha(\boldsymbol{\theta}^{(t)\,T}\boldsymbol{x}^{(i)} - y^{(i)}) \cdot \boldsymbol{x}^{(i)}$
5. $t := t + 1$
6. **end for**
7. **go to 2 until converge**

- No asegura convergencia
- Trayectoria hacia el mínimo divaga pero en general se acerca al mínimo
- Útil para conjuntos de entrenamiento gigantescos

Descenso por gradiente estocástico

- Una época es la presentación de los m datos de entrenamiento
- Método estocásticos
  - Produce soluciones acertadas más pronto
  - Ruido de trayectorias puede sacar trayectoria de tascos
  - Una época requiere m pasos
- Métodos por lotes
  - Es trivialmente paralelizable
  - Tiende a estancarse más fácil
  - Si el conjunto de datos es muy grande, un paso es caro de calcular
  - Si la superficie de error es convexa converge rápido y suave
  - Cada paso es a la vez una época
- Pueden combinarse: mini-lotes
  - Compromiso entre los dos esquemas
  - Puede hacerse
    - Con reemplazo: Preferido si conjuntos de datos son gigantescos
    - Sin reemplazo (permutación): posible sólo con conjuntos pequeños
  - Si mini-lote tiene tamaño v, entonces una época tiene m/v pasos