Apuntes Semana 4 - 13/03/2025

Jose Pablo Granados Siles
2022028503

***Abstract*—Este documento presenta un análisis formal de la regresión lineal, incluyendo su definición matemática, la función de pérdida utilizada, y las técnicas de optimización aplicadas. Se detalla el método de descenso del gradiente y se analiza la importancia de los hiperparámetros en el proceso de entrenamiento. Además, se comparan diferentes enfoques de optimización como el descenso del gradiente por lotes, estocástico y mini-lotes.**

I. Introducción

La regresión lineal es una técnica estadística fundamental en el análisis de datos y aprendizaje automático. Este método busca establecer una relación lineal entre una variable dependiente y una o más variables independientes.

II. Noticias Recientes en IA

Antes de profundizar en nuestro tema principal, mencionaremos algunas noticias relevantes en el campo de la inteligencia artificial:

- **Nuevo CEO de Intel:** Lip-Bu Tan.
- **Manus AI:** Un agente de IA general que integra pensamientos y acciones, utilizando LLMs avanzados con superior rendimiento en el benchmark GAIA.
- **Archon AI:** Un sistema capaz de crear otros agentes de IA mediante flujos de trabajo de codificación avanzados.
- **OpenAI:** Ha lanzado nuevas herramientas para desarrolladores y empresas para crear agentes de IA, incluyendo la API de Respuestas y herramientas integradas como búsqueda web y uso de computadora.

Actualmente, el enfoque principal en la industria está en el desarrollo de herramientas para crear agentes de IA, más que en los agentes mismos.

III. Fundamentos de Regresión Lineal

A. Definición Formal

La regresión lineal es un método estadístico que analiza la relación entre una variable dependiente Y y una o más variables independientes X. Formalmente, disponemos de un conjunto de datos etiquetados $\{(x_i, y_i)\}_{i=1}^{N}$, donde x_i representa el vector de variables independientes y y_i es la variable dependiente correspondiente.

El objetivo de la regresión lineal es construir un modelo que pueda predecir valores de Y a partir de X. Este modelo se define como:

$$f_{w,b}(x) = wx + b \tag{1}$$

donde:

- w es el vector de pesos o coeficientes
- b es el término independiente o sesgo
- x es el vector de variables independientes

Para problemas multidimensionales, w y x son vectores, y el producto wx representa el producto escalar $\sum_{j=1}^{d} w_j x_j$, donde d es la dimensión del espacio de características.

B. Ejemplo Práctico

Consideremos un modelo que predice las calificaciones de estudiantes basándose en sus horas de estudio:

Horas de estudio (x)	**Calificación (y)**
1	50
2	55
3	65
4	70
5	75

En este caso, el modelo de regresión lineal sería aproximadamente:

$$y = 5x + 45 \tag{2}$$

IV. Función de Pérdida y Optimización

A. Error Cuadrático Medio (MSE)

Para medir el rendimiento de nuestro modelo, utilizamos una función de pérdida que cuantifica la diferencia entre las predicciones y los valores reales. La función de pérdida más común en regresión lineal es el Error Cuadrático Medio (MSE):

$$L(w, b) = \frac{1}{N} \sum_{i=1}^{N} (f_{w,b}(x_i) - y_i)^2 \tag{3}$$

donde:

- N es el número total de ejemplos de entrenamiento
- $f_{w,b}(x_i)$ es la predicción del modelo para el ejemplo i
- y_i es el valor real para el ejemplo i

Es importante distinguir entre:

- **Función de pérdida (Loss function):** Mide el error para un solo ejemplo.
- **Función de costo (Cost function):** Mide el error promedio sobre todo el conjunto de datos.

B. Propiedades de la Función de Costo

La función de costo MSE tiene varias propiedades importantes:

- Es convexa, lo que garantiza un mínimo global único.
- Es diferenciable, permitiendo el uso de métodos basados en gradientes.
- Un valor grande de L indica un modelo impreciso.
- Si $L = 0$, el modelo predice perfectamente todos los puntos de datos.

C. Minimización de la Función de Costo

Para encontrar los valores óptimos de w y b, necesitamos minimizar la función de costo $L(w, b)$. Una función convexa tiene un punto mínimo global, mientras que las funciones no convexas pueden tener múltiples mínimos locales.

Una función $f(x)$ tiene un mínimo local en $x = a$ si existe un intervalo abierto $(a - \delta, a + \delta)$ tal que $f(a) \leq f(x)$ para todo x en el intervalo.

V. Descenso del Gradiente

A. Concepto y Metodología

El descenso del gradiente es un algoritmo de optimización que busca el mínimo de una función avanzando en la dirección opuesta al gradiente. Matemáticamente, actualiza los parámetros de la siguiente manera:

$$w := w - \alpha \frac{\partial L}{\partial w} \tag{4}$$

$$b := b - \alpha \frac{\partial L}{\partial b} \tag{5}$$

donde α es la tasa de aprendizaje (learning rate).

B. Repaso de Derivadas

Para aplicar el descenso del gradiente, necesitamos calcular las derivadas parciales de la función de costo respecto a los parámetros w y b. Recordemos algunas reglas de derivación:

- **Derivada de una constante:** Si $f(x) = c$, entonces $f'(x) = 0$
- **Derivada de la variable independiente:** Si $f(x) = x$, entonces $f'(x) = 1$
- **Derivada de un coeficiente multiplicado por la variable:** Si $f(x) = cx$, entonces $f'(x) = c$
- **Derivada de una potencia:** Si $f(x) = x^n$, entonces $f'(x) = nx^{n-1}$
- **Derivada de una suma:** Si $f(x) = u(x)+v(x)$, entonces $f'(x) = u'(x) + v'(x)$
- **Derivada de un producto:** Si $f(x) = u(x)v(x)$, entonces $f'(x) = u'(x)v(x) + u(x)v'(x)$

C. Ejemplo de Derivada

Consideremos la función $f(x) = 2x \cdot 3x$:

$$f(x) = 2x \cdot 3x \tag{6}$$

$$f'(x) = 2 \cdot 3x + 2x \cdot 3 \tag{7}$$

$$f'(x) = 6x + 6x \tag{8}$$

$$f'(x) = 12x \tag{9}$$

D. Derivadas Parciales

Las derivadas parciales representan la tasa de cambio de una función con respecto a una variable específica, manteniendo las demás variables constantes.

Si $f(x, y)$ es una función de dos variables, entonces:

- La derivada parcial respecto a x se denota como $\frac{\partial f}{\partial x}$
- La derivada parcial respecto a y se denota como $\frac{\partial f}{\partial y}$

E. Cálculo de Derivadas Parciales para la Función de Costo

Para la función de costo $L(w, b) = \frac{1}{N} \sum_{i=1}^{N} (f_{w,b}(x_i) - y_i)^2$ con $f_{w,b}(x) = wx + b$, calculamos:

$$\frac{\partial L}{\partial w} = \frac{\partial}{\partial w} \left[\frac{1}{N} \sum_{i=1}^{N} (wx_i + b - y_i)^2 \right] \tag{10}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \frac{\partial}{\partial w} \left[(wx_i + b - y_i)^2 \right] \tag{11}$$

$$= \frac{1}{N} \sum_{i=1}^{N} 2(wx_i + b - y_i) \cdot \frac{\partial}{\partial w} (wx_i + b - y_i) \tag{12}$$

$$= \frac{1}{N} \sum_{i=1}^{N} 2(wx_i + b - y_i) \cdot x_i \tag{13}$$

$$= \frac{2}{N} \sum_{i=1}^{N} x_i (wx_i + b - y_i) \tag{14}$$

De manera similar, para b:

$$\frac{\partial L}{\partial b} = \frac{\partial}{\partial b} \left[\frac{1}{N} \sum_{i=1}^{N} (wx_i + b - y_i)^2 \right] \tag{15}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \frac{\partial}{\partial b} \left[(wx_i + b - y_i)^2 \right] \tag{16}$$

$$= \frac{1}{N} \sum_{i=1}^{N} 2(wx_i + b - y_i) \cdot \frac{\partial}{\partial b} (wx_i + b - y_i) \tag{17}$$

$$= \frac{1}{N} \sum_{i=1}^{N} 2(wx_i + b - y_i) \cdot 1 \tag{18}$$

$$= \frac{2}{N} \sum_{i=1}^{N} (wx_i + b - y_i) \tag{19}$$

VI. Actualización de Parámetros

Con las derivadas parciales calculadas, podemos actualizar los parámetros w y b usando las fórmulas:

$$w := w - \alpha \frac{\partial L}{\partial w} \tag{20}$$

$$b := b - \alpha \frac{\partial L}{\partial b} \tag{21}$$

A. Selección de la Tasa de Aprendizaje

La elección del hiperparámetro α (tasa de aprendizaje) es crucial:

- Un α muy pequeño requiere muchas iteraciones para converger.
- Un α muy grande puede hacer que el algoritmo no converja o salte sobre el mínimo.

En la práctica, se suele comenzar con un valor relativamente grande y reducirlo gradualmente para encontrar el óptimo.

VII. Conceptos Avanzados

A. Epochs

Un epoch es una iteración completa sobre todos los datos del conjunto de entrenamiento. El número de epochs es un hiperparámetro que determina cuántas veces se procesará todo el conjunto de datos durante el entrenamiento.

B. Batch

Un batch es un subconjunto del total de datos de entrenamiento utilizado para calcular el gradiente. Existen diferentes enfoques de entrenamiento:

- **Batch Gradient Descent:** Calcula el error utilizando todo el conjunto de datos antes de actualizar los parámetros.
  - Ventajas: Convergencia y gradiente estables.
  - Desventajas: Requiere tener todo el dataset en memoria; puede ser lento para datasets grandes.
- **Stochastic Gradient Descent (SGD):** Actualiza los parámetros después de cada ejemplo individual.
  - Ventajas: Rápido; útil para datasets grandes.
  - Desventajas: Puede ser ruidoso e inestable.
- **Mini-Batch Gradient Descent:** Actualiza los parámetros después de procesar un subconjunto de ejemplos.
  - Ventajas: Combina beneficios de batch y SGD; más estable que SGD pero más rápido que batch.
  - Desventajas: Requiere ajustar el tamaño del batch.

C. ¿Por qué usar MSE en lugar de MAE?

El Error Absoluto Medio (MAE) es otra función de pérdida posible, definida como:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |f_{w,b}(x_i) - y_i| \tag{22}$$

Sin embargo, el MSE se prefiere sobre el MAE porque:

- El MSE es diferenciable en todos los puntos.
- El MAE no es diferenciable en el punto donde el error es cero, lo que complica la aplicación del descenso del gradiente.

VIII. Conclusión

La regresión lineal es una técnica fundamental en el análisis estadístico y el aprendizaje automático. A través de este documento, hemos explorado sus fundamentos matemáticos, metodología de optimización y consideraciones prácticas. Aunque simple, la regresión lineal proporciona una base sólida para entender métodos más complejos de aprendizaje automático.