

RESUMEN DEL ARTICULO:

"Comparing THE MOORE - PENROSE PSEUDOINVERSE AND
GRADIENT DESCENT FOR SOLVING LINEAR REGRESSION PROBLEMS
A PERFORMANCE ANALYSIS"

El artículo presenta una investigación exhaustiva sobre métodos para resolver regresión lineal por mínimos cuadrados ordinarios ols, cuyo objetivo es encontrar el vector de parámetros β que minimiza la suma de errores al cuadrado, expresada como $\arg \min_{\beta} \|X\beta - y\|_2^2$.

1: La Pseudoínsvera de Moore - Penrose (pinv):

- Es una solución analítica directa y cerrada. Generaliza la inversa de una matriz permitiendo encontrar una solución incluso si la matriz X no es cuadrada.
- La solución se obtiene con $\beta_{\text{pinv}} = X^T y$.

2: EL DESCENSO DE GRADIENTE (GD) :

Es un método iterativo de optimización que encuentra la solución de forma incremental, actualizando los parámetros para minimizar errores.

Para realizar la comparación se emplean:

- Se generaron conjuntos de datos donde se controlan sistemáticamente el número de muestras (n), el número de características (d) y factor de condición de matriz X . Un factor de condición bajo que simula datos mal condicionados.
- Se usó conjunto de datos "California Housing" ($n \approx 20,640$, $d = 8$) y "Oci diabetes" ($n = 492$, $d = 10$) para validar resultados.
- El rendimiento se midió usando el tiempo de ejecución, error cuadrático medio para medir precisión, iteraciones hasta la convergencia.

RESULTADOS

3 EFICIENCIA COMPUTACIONAL (TIEMPO EJECUCION)

- Para los largos protocolo hasta $n=5000$, $d=50$ la pseudoinversa fue consistentemente mas rápida a menudo por ordenes de magnitud, completando los milisegundos
- El tiempo de ejecución del descenso de gradiente (batchGD) fue notablemente mas lento, ya que su costo por iteración escala con el numero de muestras (n) y características (d).

4 PRECISION Y ESTABILIDAD NUMERICA

la pseudoinversa proporciona una solución exacta con MSE bajo y estable en todas las condiciones y el descenso de gradiente solo alcanzo una precision comparable (MSE bajo) cuando los datos estan bien acondicionados.

5 SENSIBILIDAD DEL CONDICIONAMIENTO

- En el descenso de gradiente demostró ser altamente sensible al condicionamiento de datos
- Los datos mal acondicionada haciendo GD requeriría mas iteraciones y fallara en encontrar solucion precisa mientras que la pseudoinversa al ser solucion analitica no se vio afectada de la misma manera y encontró la solución convexa de minimos cuadrado de norma minima

CONCLUSION

Usar la Pseudoinversa para conjunto de datos pequeño donde n y d son Grandes, la pseudoinversa de Moore-Penrose es generalmente preferible

Usar descenso gradiente: los métodos iterativos en el descenso de gradiente se vuelven "inexactos" o "no prácticos" para conjunto de datos extremadamente grande

Reporte de Comparación de Rendimiento

Se realizo un análisis comparativo entre la función LM y una implementación de gradiente evaluando tiempo de ejecución y uso de memoria de 100 a 1000 000 usando benchmarks y siendo perfilado por Profvis para análisis de los métodos.

| Tamaño | LM | Gradiente |
|----------|-----------|------------|
| 100 | 0,001357 | 0,002312 |
| 1000 | 0,0014827 | 0,0220196 |
| 10,000 | 0,0074827 | 0,1641378 |
| 100,000 | 0,0553629 | 1,4426852 |
| 500,000 | 0,2189633 | 7,1380826 |
| 1000,000 | 0,5600950 | 14,8881908 |

- LM() muestra escalabilidad lineal con pendiente suave
- Gradiente descendente escala de forma menos eficiente
- La brecha aumenta a medida que n aumenta

USO DE MEMORIA (BYTES)

| Tamaño | LM | Gradiente |
|----------|-----------|-------------|
| 100 | 64680 | 1830776 |
| 1000 | 619048 | 17619200 |
| 10,000 | 6213928 | 176019200 |
| 100,000 | 61851432 | 1760019200 |
| 500,000 | 308197160 | 8800019200 |
| 1000,000 | 616391464 | 17600019200 |

Contrario a intuición inicia el gradiente descendente usa 28x mas memoria que LM durante la ejecución

File Edit Code View Plots Session Build Debug Profile Tools Help

Source or Save Run Source Environment History Connections Tutorial

Untitled Profile

```

 09 #> ## USO DE MEMORIA
10 p2 <- ggplot(resultados, aes(x = Tamaño, y = Memoria_B / 1e6, color = Método)) +
11   geom_line(linewidth = 1.3) + # CORREGIDO: Linewidth en lugar de size
12   scale_x_log10(labels = scales::comma) +
13   labs(title = "Comparación de uso de memoria",
14        x = "Tamaño de muestra (escala logarítmica)",
15        y = "Memoria (MB)") +
16   theme_minimal(base_size = 13)
17
18 print(p2)
19
20
21 # ----- Análisis detallado con profvis -----
22 n <- 1000000
23 X <- matrix(rnorm(n * 10), ncol = 10)
24 Y <- rnorm(n)
25
26 cat("\n===== PERFIANDO CON PROFVIS (n = 1,000,000) =====\n")
27
28 p <- profvis(c(
29   modelo_lm <- lm(Y ~ X),
30   gradienre_desc(X, Y)
31 ))
32
33 print(p)
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94

```

93:11 Análisis detallado con profvis

Console Terminal Background Jobs

R • R 4.5.1 · -j

