Universidad Torcuato Di Tella

Métodos Computacionales

Segundo Trabajo Práctico

Optimización y Análisis de Datos

Integrantes:

Martina Mariño
Kiara Martínez
Sol Camus

Índice

1. Modelado y cuadrados mínimos **2**

1.1. **Escribir el sistema normal $A^\top A\,\beta = A^\top y$** . . . 2

1.2. **Función en Python que resuelve el sistema** . . . 5

1.3. **Gráfico de datos y curva ajustada** . . . 6

1.4. **Cálculo del error cuadrático medio (MSE)** . . . 7

2. Interpretación mediante SVD **8**

2.1. **Cálculo de la SVD de A** . . . 8

2.2. **Verificación numérica de la ortogonalidad de U** . . . 9

2.3. **Comparación de soluciones obtenidas por SVD y por ecuaciones normales** . . . 10

2.4. **Mal condicionamiento y colinealidad** . . . 12

3. Análisis cuadrático **13**

3.1. **Gráfico de $Q(\beta_1, \beta_2)$ para β_0 fijo** . . . 13

3.2. **Convexidad de $Q(\beta)$ y mínimo en la solución de mínimos cuadrados** 15

4. Optimización numérica **15**

4.1. **Descenso por gradiente para minimizar $Q(\beta)$** . . . 15

4.2. **Experimentos con tasas de aprendizaje y tolerancias** . . . 17

4.3. **(Opcional) Convergencia respecto de la solución por SVD** . . . 18

5. Discusión y extensiones **19**

5.1. **Relación entre los distintos métodos de resolución** . . . 19

5.2. **Efecto de agregar términos cúbicos o ruido en el ajuste** . . . 20

6. Aplicación a un caso real: predicción de salario **22**

6.1. **Descripción del conjunto de datos** . . . 22

6.2. **Preprocesamiento y formulación matricial** . . . 22

6.3. **Problemas de colinealidad y mal condicionamiento** . . . 23

6.4. **Ajuste mediante SVD y descenso por gradiente** . . . 23

6.5. **Análisis de residuos e interpretación** . . . 24

1. Modelado y cuadrados mínimos

Se dispone del conjunto de datos:

$$(x_i, y_i) = \{(0,1), (1,2), (2,2{,}8), (3,3{,}6), (4,4{,}5)\}$$

Queremos ajustar un modelo cuadrático de la forma:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2$$

usando el método de los mínimos cuadrados.

1.1. Escribir el sistema normal $A^\top A\,\beta = A^\top y$

1) Cálculo de A, $A^\top$, y, β

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix}, \qquad A^\top = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 \\ 0 & 1 & 4 & 9 & 16 \end{bmatrix},$$

$$y = \begin{bmatrix} 1 \\ 2 \\ 2.8 \\ 3.6 \\ 4.5 \end{bmatrix}, \qquad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}.$$

2) Cálculo de $A^\top A$

$$A^\top A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 \\ 0 & 1 & 4 & 9 & 16 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix} = \boxed{\begin{bmatrix} 5 & 10 & 30 \\ 10 & 30 & 100 \\ 30 & 100 & 354 \end{bmatrix}}$$

3) Cálculo de $(A^\top A)^{-1}$

Para calcular la inversa de $A^\top A$ construimos la matriz aumentada

$$\left[\, A^\top A \;\middle|\; I \,\right] = \left[\begin{array}{ccc|ccc} 5 & 10 & 30 & 1 & 0 & 0 \\ 10 & 30 & 100 & 0 & 1 & 0 \\ 30 & 100 & 354 & 0 & 0 & 1 \end{array}\right],$$

y aplicamos el método de Gauss–Jordan.

El procedimiento consiste en realizar operaciones elementales por filas hasta transformar la parte izquierda en la matriz identidad.

Al finalizar, obtenemos

$$\left[\, I \;\middle|\; (A^{\mathsf{T}}A)^{-1} \,\right],$$

lo que nos permite leer directamente la matriz inversa buscada.

Paso 1 (pivote en fila 1): $R_1 \leftarrow \frac{1}{5}R_1$.

$$\left[\begin{array}{ccc|ccc} 1 & 2 & 6 & \frac{1}{5} & 0 & 0 \\ 10 & 30 & 100 & 0 & 1 & 0 \\ 30 & 100 & 354 & 0 & 0 & 1 \end{array}\right]$$

Paso 2 (anular debajo del pivote):

$R_2 \leftarrow R_2 - 10R_1, \quad R_3 \leftarrow R_3 - 30R_1.$

$$\left[\begin{array}{ccc|ccc} 1 & 2 & 6 & \frac{1}{5} & 0 & 0 \\ 0 & 10 & 40 & -2 & 1 & 0 \\ 0 & 40 & 174 & -6 & 0 & 1 \end{array}\right]$$

Paso 3 (pivote en fila 2): $R_2 \leftarrow \frac{1}{10}R_2$.

$$\left[\begin{array}{ccc|ccc} 1 & 2 & 6 & \frac{1}{5} & 0 & 0 \\ 0 & 1 & 4 & -\frac{1}{5} & \frac{1}{10} & 0 \\ 0 & 40 & 174 & -6 & 0 & 1 \end{array}\right]$$

Paso 4 (anular arriba del segundo pivote): $R_1 \leftarrow R_1 - 2R_2$.

$$\left[\begin{array}{ccc|ccc} 1 & 0 & -2 & \frac{3}{5} & -\frac{1}{5} & 0 \\ 0 & 1 & 4 & -\frac{1}{5} & \frac{1}{10} & 0 \\ 0 & 40 & 174 & -6 & 0 & 1 \end{array}\right]$$

Paso 5 (pivote en fila 3 sobre la tercera columna):

$R_3 \leftarrow \frac{1}{174}R_3$.

$$\left[\begin{array}{ccc|ccc} 1 & 0 & -2 & \frac{3}{5} & -\frac{1}{5} & 0 \\ 0 & 1 & 4 & -\frac{1}{5} & \frac{1}{10} & 0 \\ 0 & \frac{20}{87} & 1 & -\frac{1}{29} & 0 & \frac{1}{174} \end{array}\right]$$

Paso 6 (anular la tercera columna arriba):

$R_2 \leftarrow R_2 - 4R_3, \quad R_1 \leftarrow R_1 + 2R_3.$

$$\left[\begin{array}{ccc|ccc} 1 & \frac{40}{87} & 0 & \frac{77}{145} & -\frac{1}{5} & \frac{1}{87} \\ 0 & \frac{7}{87} & 0 & -\frac{9}{145} & \frac{1}{10} & -\frac{2}{87} \\ 0 & \frac{20}{87} & 1 & -\frac{1}{29} & 0 & \frac{1}{174} \end{array}\right]$$

Paso 7 (normalizar segundo pivote y eliminar en otras filas):
$R_2 \leftarrow \frac{87}{7}R_2, \quad R_1 \leftarrow R_1 - \frac{40}{87}R_2, \quad R_3 \leftarrow R_3 - \frac{20}{87}R_2.$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{31}{35} & -\frac{27}{35} & \frac{1}{7} \\ 0 & 1 & 0 & -\frac{27}{35} & \frac{87}{70} & -\frac{2}{7} \\ 0 & 0 & 1 & \frac{1}{7} & -\frac{2}{7} & \frac{1}{14} \end{array}\right]$$

Por lo tanto,

$$\boxed{(A^\intercal A)^{-1} = \begin{bmatrix} \frac{31}{35} & -\frac{27}{35} & \frac{1}{7} \\ -\frac{27}{35} & \frac{87}{70} & -\frac{2}{7} \\ \frac{1}{7} & -\frac{2}{7} & \frac{1}{14} \end{bmatrix}}$$

$$\Rightarrow \quad (A^\intercal A)^{-1} \approx \begin{bmatrix} 0.886 & -0.771 & 0.143 \\ -0.771 & 1.243 & -0.286 \\ 0.143 & -0.286 & 0.071 \end{bmatrix}$$

4) Cálculo de $A^\intercal y$

$$A^\intercal y = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 \\ 0 & 1 & 4 & 9 & 16 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 2.8 \\ 3.6 \\ 4.5 \end{bmatrix} = \boxed{\begin{bmatrix} 13.9 \\ 36.4 \\ 117.6 \end{bmatrix}}$$

5) Cálculo de $\boldsymbol{\beta} = (A^\mathsf{T}A)^{-1}(A^\mathsf{T}y)$

$$\beta = \begin{bmatrix} \frac{31}{35} & -\frac{27}{35} & \frac{1}{7} \\ -\frac{27}{35} & \frac{87}{70} & -\frac{2}{7} \\ \frac{1}{7} & -\frac{2}{7} & \frac{1}{14} \end{bmatrix} \begin{bmatrix} 13.9 \\ 36.4 \\ 117.6 \end{bmatrix} \approx \boxed{\begin{bmatrix} 1.031 \\ 0.917 \\ -0.014 \end{bmatrix}}.$$

6) Modelo ajustado

Reemplazando los valores obtenidos, el modelo cuadrático ajustado por mínimos cuadrados es:

$$\hat{y} \;=\; 1{,}031 \;+\; 0{,}917\,x \;-\; 0{,}014\,x^2.$$

Justificación

El vector β obtenido minimiza la suma de cuadrados de los errores entre los valores observados y y los valores estimados $\hat{y} = A\beta$.

Geométricamente, $\hat{y}$ es la proyección ortogonal de y sobre el subespacio columna de A, lo que garantiza la mejor aproximación en la norma euclidiana.

1.2. Función en Python que resuelve el sistema

A partir del sistema normal obtenido en el apartado anterior:

$$(A^\mathsf{T}A)\,\beta = A^\mathsf{T}y,$$

implementamos en `Python` una función que resuelve este sistema utilizando el método de eliminación por Gauss.

Para ello definimos primero una rutina genérica que, dada una matriz M y un vector b, resuelve el sistema lineal

$$Mx = b$$

mediante el algoritmo de eliminación progresiva seguido de sustitución regresiva, garantizando la estabilidad numérica mediante la selección del pivote máximo en cada columna.

Posteriormente, construimos la matriz de diseño del modelo cuadrático,

$$A = \begin{bmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix},$$

y calculamos los términos del sistema normal:

$$A^\mathsf{T}A \qquad \text{y} \qquad A^\mathsf{T}y.$$

Finalmente resolvemos el sistema lineal $(A^\mathsf{T}A)\beta = A^\mathsf{T}y$ utilizando la función implementada.

Al ejecutar el código con los datos del enunciado se obtiene:

$$\beta \approx \begin{bmatrix} 1.031 \\ 0.917 \\ -0.014 \end{bmatrix},$$

en acuerdo con la solución analítica calculada en el apartado 1.1.

1.3. Gráfico de datos y curva ajustada

Para visualizar el ajuste obtenido, utilizamos los coeficientes β calculados mediante el sistema normal $(A^\mathsf{T}A)\beta = A^\mathsf{T}y$.

Con estos valores construimos la función estimada

$$\hat{y}(x) = \beta_0 + \beta_1 x + \beta_2 x^2,$$

y la evaluamos sobre un conjunto denso de puntos.

Simultáneamente, graficamos los datos originales (x_i, y_i) provistos en la consigna.

El resultado se muestra en la siguiente figura, donde se observa que la curva cuadrática obtenida por mínimos cuadrados sigue la tendencia general de los datos, pasando muy cerca de todos los puntos y confirmando que el modelo captura bien la relación entre x e y.

Este gráfico permite verificar visualmente la calidad del ajuste y complementa el análisis algebraico desarrollado en los apartados anteriores.

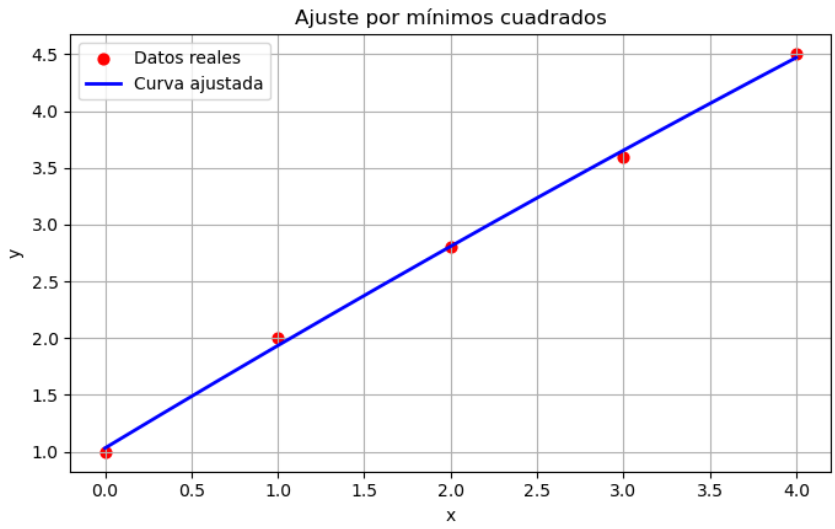

Figura 1: Datos originales y curva cuadrática ajustada.

1.4. Cálculo del error cuadrático medio (MSE)

Una vez obtenidos los coeficientes β y la correspondiente curva ajustada $\hat{y} = A\beta$, evaluamos la calidad del modelo mediante el error cuadrático medio (MSE), definido como

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2.$$

Dado que

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix}, \qquad \beta = \begin{bmatrix} 1.031 \\ 0.917 \\ -0.014 \end{bmatrix},$$

la predicción se obtiene como

$$\hat{y} = A\beta = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix} \begin{bmatrix} 1.031 \\ 0.917 \\ -0.014 \end{bmatrix} = \begin{bmatrix} 1.031 \\ 1.934 \\ 2.809 \\ 3.656 \\ 4.475 \end{bmatrix}.$$

Estos valores corresponden a la evaluación exacta del modelo cuadrático ajustado.

Debido al redondeo y truncamiento numérico en la computadora, el resultado impreso por el código es:

$$\hat{y} \approx \begin{bmatrix} 1.031 \\ 1.934 \\ 2.809 \\ 3.654 \\ 4.471 \end{bmatrix}.$$

El MSE mide la discrepancia promedio entre los datos observados y_i y los valores estimados $\hat{y}_i$.

Un MSE pequeño indica que el modelo logra una buena aproximación a los datos.

En nuestro caso, utilizando el vector de valores observados y y los valores estimados $\hat{y}$, se obtiene:

$$\text{MSE} \approx 0.001829.$$

El valor obtenido es muy pequeño, lo que confirma que el modelo cuadrático ajustado representa adecuadamente la tendencia de los datos suministrados.

2. Interpretación mediante SVD

Sea la SVD de A: $A = U\,\Sigma\,V^{\mathsf{T}}$, con $U \in \mathbb{R}^{5\times5}$, $\Sigma \in \mathbb{R}^{5\times3}$ diagonal y $V^{\mathsf{T}} \in \mathbb{R}^{3\times3}$.

2.1. Cálculo de la SVD de A

Matriz de diseño A:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix}$$

Autovalores y valores singulares para la SVD:

Para una matriz rectangular, los *valores singulares* σ_i se obtienen como las raíces cuadradas de los *autovalores* de $A^{\mathsf{T}}A$.

Calculamos

$$A^{\mathsf{T}}A = \begin{bmatrix} 5 & 10 & 30 \\ 10 & 30 & 100 \\ 30 & 100 & 354 \end{bmatrix}$$

Los autovalores λ_i de $A^{\mathsf{T}}A$ son las raíces de

$\det(A^{\mathsf{T}}A - \lambda I) = 0$:

$$\lambda_1 = 0.524, \qquad \lambda_2 = 3.471, \qquad \lambda_3 = 385.005.$$

Entonces, los valores singulares son:

$$\sigma_i = \sqrt{\lambda_i} \quad \Rightarrow \quad \sigma_1 = 19.622, \;\; \sigma_2 = 1.863, \;\; \sigma_3 = 0.724.$$

Luego, la Σ completa es:

$$\Sigma = \begin{bmatrix} 19.622 & 0 & 0 \\ 0 & 1.863 & 0 \\ 0 & 0 & 0.724 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Autovectores:

Los vectores propios normalizados de $A^\mathsf{T}A$ asociados a $\lambda_3, \lambda_2, \lambda_1$ (ordenados para que coincidan con $\sigma_1 \geq \sigma_2 \geq \sigma_3$) forman las columnas de V.

Numéricamente,

$$V = \begin{bmatrix} -0.083 & 0.789 & 0.608 \\ -0.272 & 0.569 & -0.776 \\ -0.959 & -0.23 & 0.168 \end{bmatrix}$$

Cálculo de U:

Para cada $i = 1, 2, 3$, definimos $u_i = \dfrac{1}{\sigma_i} A\, v_i$.

Completamos U con dos vectores ortonormales u_4, u_5 para tener $U \in \mathbb{R}^{5\times 5}$.

Y así obtenemos la base ortonormal numérica:

$$U = \begin{bmatrix} -0.004 & 0.424 & 0.840 & -0.026 & -0.337 \\ -0.067 & 0.606 & 0.000 & 0.281 & 0.741 \\ -0.227 & 0.541 & -0.376 & -0.688 & -0.202 \\ -0.486 & 0.229 & -0.288 & 0.637 & -0.472 \\ -0.841 & -0.329 & 0.264 & -0.204 & 0.27 \end{bmatrix}$$

SVD final:

Con las tres matrices anteriores se verifica:

$$\boxed{A = U\,\Sigma\,V^\mathsf{T}}$$

2.2. Verificación numérica de la ortogonalidad de U

Sea

$$U = \begin{bmatrix} -0.004 & 0.424 & 0.840 & -0.026 & -0.337 \\ -0.067 & 0.606 & 0.000 & 0.281 & 0.741 \\ -0.227 & 0.541 & -0.376 & -0.688 & -0.202 \\ -0.486 & 0.229 & -0.288 & 0.637 & -0.472 \\ -0.841 & -0.329 & 0.264 & -0.204 & 0.27 \end{bmatrix}$$

$$u_i = \text{columna } i \text{ de } U.$$

Verificamos que $u_i^\mathsf{T} u_j \approx 0$ para $i \neq j$ realizando las sumas término a término (5 términos por producto interno):

$$\begin{aligned} u_1 \cdot u_2 &= (-0.004)(0.424) + (-0.067)(0.606) + (-0.227)(0.541) \\ &\quad + (-0.486)(0.229) + (-0.841)(-0.329) = -3.73 \times 10^{-7} \approx 0. \end{aligned}$$

$$\begin{aligned} u_1 \cdot u_3 &= (-0.004)(0.840) + (-0.067)(0.000) + (-0.227)(-0.376) \\ &\quad + (-0.486)(-0.288) + (-0.841)(0.264) = 6.77 \times 10^{-8} \approx 0. \end{aligned}$$

$$\begin{aligned} u_1 \cdot u_4 &= (-0.004)(-0.026) + (-0.067)(0.281) + (-0.227)(-0.688) \\ &\quad + (-0.486)(0.637) + (-0.841)(-0.204) = -4.59 \times 10^{-7} \approx 0. \end{aligned}$$

$$\begin{aligned} u_1 \cdot u_5 &= (-0.004)(-0.337) + (-0.067)(0.741) + (-0.227)(-0.202) \\ &\quad + (-0.486)(-0.472) + (-0.841)(0.27) = -3.67 \times 10^{-8} \approx 0. \end{aligned}$$

$$\begin{aligned} u_2 \cdot u_3 &= (0.424)(0.840) + (0.606)(0.000) + (0.541)(-0.376) \\ &\quad + (0.229)(-0.288) + (-0.329)(0.264) = 7.25 \times 10^{-7} \approx 0. \end{aligned}$$

$$\begin{aligned} u_2 \cdot u_4 &= (0.424)(-0.026) + (0.606)(0.281) + (0.541)(-0.688) \\ &\quad + (0.229)(0.637) + (-0.329)(-0.204) = -2.04 \times 10^{-7} \approx 0. \end{aligned}$$

$$\begin{aligned} u_2 \cdot u_5 &= (0.424)(-0.337) + (0.606)(0.741) + (0.541)(-0.202) \\ &\quad + (0.229)(-0.472) + (-0.329)(0.27) = 5.17 \times 10^{-7} \approx 0. \end{aligned}$$

$$\begin{aligned} u_3 \cdot u_4 &= (0.840)(-0.026) + (0.000)(0.281) + (-0.376)(-0.688) \\ &\quad + (-0.288)(0.637) + (0.264)(-0.204) = -2.81 \times 10^{-7} \approx 0. \end{aligned}$$

$$\begin{aligned} u_3 \cdot u_5 &= (0.840)(-0.337) + (0.000)(0.741) + (-0.376)(-0.202) \\ &\quad + (-0.288)(-0.472) + (0.264)(0.27) = -1.205 \times 10^{-6} \approx 0. \end{aligned}$$

$$\begin{aligned} u_4 \cdot u_5 &= (-0.026)(-0.337) + (0.281)(0.741) + (-0.688)(-0.202) \\ &\quad + (0.637)(-0.472) + (-0.204)(0.27) = -9.83 \times 10^{-7} \approx 0. \end{aligned}$$

2.3. Comparación de soluciones obtenidas por SVD y por ecuaciones normales

Métodos:

Para el modelo cuadrático con matriz de diseño $A = [1,\, x,\, x^2]$ y vector de observaciones y, la solución por ecuaciones normales se obtiene como

$$\beta_{\text{EN}} \;=\; (A^{\mathsf{T}}A)^{-1}A^{\mathsf{T}}y,$$

mientras que la solución basada en la descomposición en valores singulares, si $A = U\Sigma V^{\mathsf{T}}$, viene dada por

$$\beta_{\text{SVD}} \;=\; V\,\Sigma^{+}\,U^{\mathsf{T}}y,$$

donde Σ^{+} es la pseudoinversa de Σ, obtenida invirtiendo únicamente los valores singulares no nulos.

Resultados numéricos:

En este problema, ambos métodos producen exactamente los mismos coeficientes:

$$\beta_{\text{EN}} \;=\; \beta_{\text{SVD}} \;=\; \begin{bmatrix} 1.031 \\ 0.917 \\ -0.014 \end{bmatrix}.$$

Las predicciones coinciden punto por punto:

$$\hat{y} = \begin{bmatrix} 1.031 \\ 1.934 \\ 2.809 \\ 3.654 \\ 4.471 \end{bmatrix}.$$

El error cuadrático medio es idéntico en ambos casos:

$$\text{MSE}_{\text{EN}} \;=\; \text{MSE}_{\text{SVD}} \;=\; 0.002.$$

Las diferencias relativas son prácticamente nulas:

$$\frac{\|\beta_{\text{EN}} - \beta_{\text{SVD}}\|}{\|\beta_{\text{EN}}\|} = 2.525 \times 10^{-15}, \qquad \frac{\|\hat{y}_{\text{EN}} - \hat{y}_{\text{SVD}}\|}{\|\hat{y}_{\text{EN}}\|} = 3.941 \times 10^{-16}.$$

Discusión:

La coincidencia entre ambos métodos se debe a que la matriz A tiene rango completo (tres columnas linealmente independientes). En este caso, la solución de mínimos cuadrados es única y ambas expresiones la recuperan exactamente.

Sin embargo, la SVD es más robusta desde el punto de vista numérico, ya que no requiere invertir la matriz $A^{\mathsf{T}}A$, la cual puede ser mal condicionada si las columnas de A son casi colineales.

2.4. Mal condicionamiento y colinealidad

Concepto de mal condicionamiento:

Un problema de mínimos cuadrados se considera mal condicionado cuando pequeñas perturbaciones en los datos (A o y) producen grandes variaciones en la solución β.

Esto ocurre cuando la matriz $A^\intercal A$ es casi singular, es decir, cuando presenta valores propios muy pequeños o muy desbalanceados entre sí. En tales casos, la inversión de $A^\intercal A$ amplifica el error numérico y vuelve inestable el cálculo de β.

Relación con la colinealidad:

La causa principal del mal condicionamiento es la colinealidad (o casi colinealidad) entre las columnas de A.

Si una columna puede aproximarse como combinación lineal de otras, las direcciones en el espacio de parámetros dejan de ser independientes.

Geométricamente, esto significa que las columnas pierden ángulo entre sí y se "aplanan" hacia un mismo subespacio, reduciendo la información efectiva del modelo.

Como consecuencia, pequeñas variaciones en los datos pueden producir cambios desproporcionados en la estimación de β, afectando la estabilidad y la interpretación del modelo.

Interpretación mediante SVD:

La descomposición $A = U\Sigma V^\intercal$ permite analizar este fenómeno con claridad.

Los valores singulares σ_i cuantifican la magnitud de la transformación lineal de A en cada dirección principal.

Un sistema está mal condicionado cuando alguno de estos valores es muy pequeño en comparación con el mayor.

El número de condición

$$\kappa(A) = \frac{\sigma_{\text{máx}}}{\sigma_{\text{mín}}}$$

resume esta relación: valores grandes de $\kappa(A)$ indican fuerte desbalance y, por lo tanto, sensibilidad a perturbaciones y mala estabilidad numérica.

Aplicación al problema:

En nuestro caso, los valores singulares obtenidos fueron

$$\sigma_1 \approx 19.622, \qquad \sigma_2 \approx 1.863, \qquad \sigma_3 \approx 0.724,$$

de donde se obtiene

$$\kappa(A) = \frac{19.622}{0.724} \approx 27.1.$$

Este valor indica un condicionamiento moderado.

No existe una inestabilidad severa, pero sí puede observarse cierta correlación entre las

columnas de A, especialmente entre x y x^2, ya que ambas crecen con la variable independiente.

Esto explica que, aunque las soluciones por ecuaciones normales y por SVD coinciden casi exactamente, el método basado en SVD sea conceptualmente preferible cuando el crecimiento de las columnas del diseño es más pronunciado o cuando se incorporan modelos polinomiales de mayor grado.

3. Análisis cuadrático

Sea la forma cuadrática asociada al error:

$$Q(\beta) = \|A\beta - y\|^2 = \beta^\mathsf{T}(A^\mathsf{T}A)\beta - 2(A^\mathsf{T}y)^\mathsf{T}\beta + y^\mathsf{T}y.$$

3.1. Gráfico de $Q(\beta_1, \beta_2)$ para β_0 fijo

Para visualizar la forma cuadrática $Q(\beta)$ asociada al error, fijamos el valor del parámetro β_0 en el valor obtenido por mínimos cuadrados en la sección anterior $\beta_0 \approx 1{,}031$, y consideramos Q como función de las variables β_1 y β_2:

$$(\beta_1, \beta_2) \longmapsto Q(\beta_0, \beta_1, \beta_2) = \|A\beta - y\|^2.$$

Numéricamente se genera una malla de valores (β_1, β_2) en un rectángulo que contiene al punto óptimo $(\beta_1^*, \beta_2^*) \approx (0{,}917, -0{,}014)$, y para cada punto de la malla se evalúa $Q(\beta_0, \beta_1, \beta_2)$.

Con estos valores se construyen:

- Un mapa de curvas de nivel de $Q(\beta_1, \beta_2)$, donde las líneas representan niveles constantes de error. Estas curvas resultan elipses concéntricas alrededor del mínimo.
- Una superficie tridimensional de la función $(\beta_1, \beta_2) \mapsto Q(\beta_1, \beta_2)$, que muestra la típica forma de "paraboloide" asociada a una forma cuadrática convexa.

En ambas representaciones se marca el punto

$$(\beta_1^*, \beta_2^*) \approx (0{,}917,\ -0{,}014),$$

correspondiente a los coeficientes obtenidos por mínimos cuadrados.

Visualmente se observa que este punto coincide con el mínimo global de Q en el plano (β_1, β_2), y que el error crece de manera suave y aproximadamente elíptica al alejarse de dicho punto, lo cual es consistente con la estructura cuadrática de la función de error.

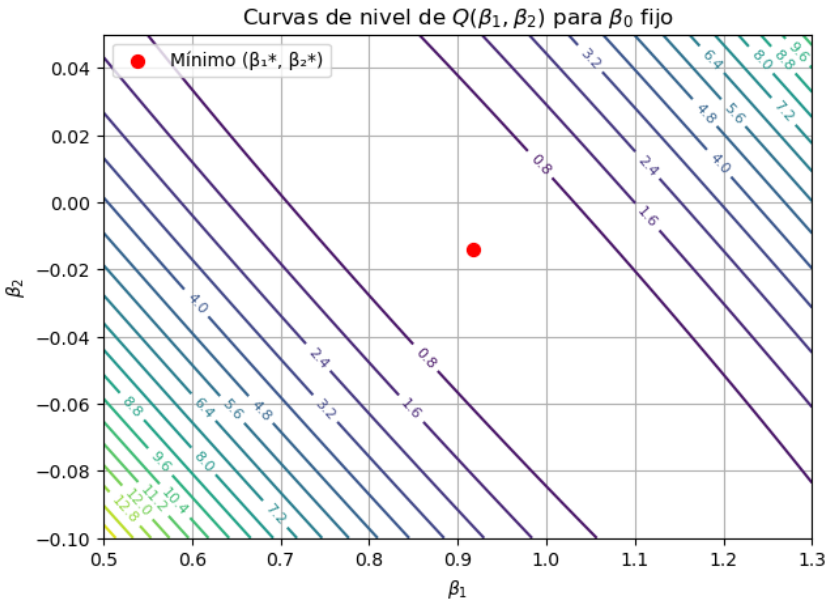

Figura 2: Curvas de nivel de $Q(\beta_1, \beta_2)$ con β_0 fijo.

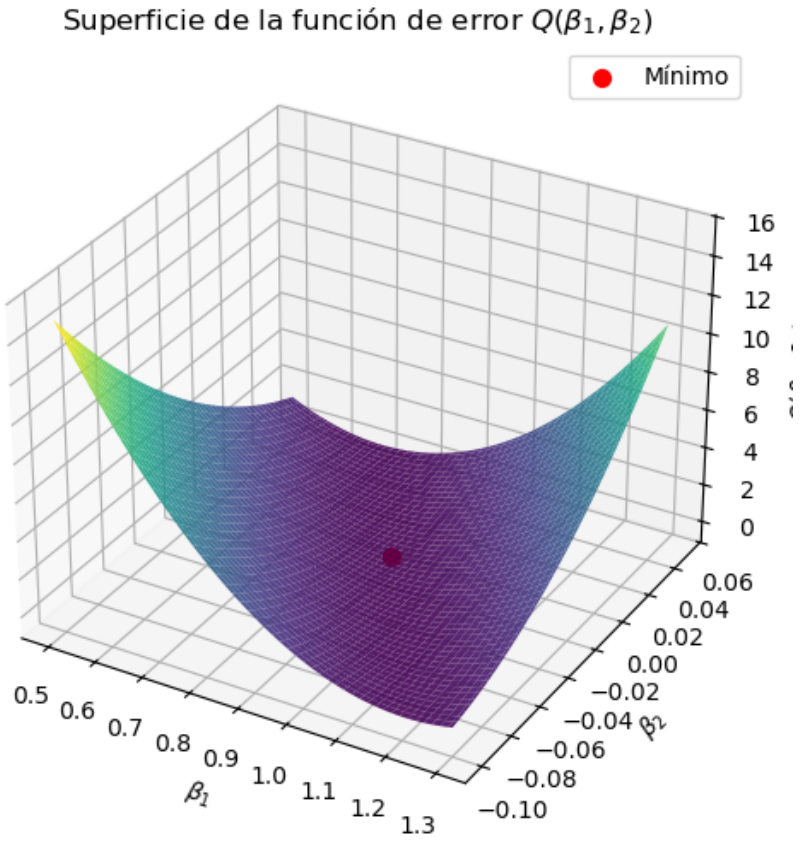

Figura 3: Superficie de la función de error $Q(\beta_1, \beta_2)$ con β_0 fijo.

3.2. Convexidad de $Q(\beta)$ y mínimo en la solución de mínimos cuadrados

Para analizar la convexidad de la función de error y localizar su mínimo, calculamos el gradiente y el Hessiano de Q respecto de β.

Partiendo de la expresión

$$Q(\beta) = \|A\beta - y\|^2 = (A\beta - y)^\mathsf{T}(A\beta - y),$$

obtenemos:

$$\nabla_\beta Q(\beta) = 2A^\mathsf{T}(A\beta - y), \qquad \nabla^2_\beta Q(\beta) = 2A^\mathsf{T}A.$$

La matriz $A^\mathsf{T}A$ es simétrica y, para todo vector z,

$$z^\mathsf{T}(A^\mathsf{T}A)z = (Az)^\mathsf{T}(Az) = \|Az\|^2 \geq 0.$$

Esto implica que $A^\mathsf{T}A$ es semidefinida positiva.

En consecuencia, la matriz Hessiana $\nabla^2 Q(\beta) = 2A^\mathsf{T}A$ también es semidefinida positiva y, por lo tanto, $Q(\beta)$ es una función convexa.

Si, además, las columnas de A son linealmente independientes, entonces $\|Az\|^2 = 0$ solo cuando $z = 0$, lo que implica que $A^\mathsf{T}A$ es definida positiva.

En este caso, $Q(\beta)$ es estrictamente convexa y posee un único mínimo global.

El punto crítico se obtiene anulando el gradiente:

$$\nabla_\beta Q(\beta) = 0 \quad \Longrightarrow \quad A^\mathsf{T}A\,\beta = A^\mathsf{T}y.$$

Este sistema corresponde a las ecuaciones normales del problema de mínimos cuadrados.

Si $A^\mathsf{T}A$ es invertible, su solución viene dada por

$$\beta^* = (A^\mathsf{T}A)^{-1}A^\mathsf{T}y.$$

Dado que $Q(\beta)$ es convexa (y estrictamente convexa cuando $A^\mathsf{T}A$ es definida positiva), este punto crítico β^* es el mínimo global de la función de error.

Por lo tanto, la solución de mínimos cuadrados coincide exactamente con el punto que minimiza la forma cuadrática $Q(\beta)$.

4. Optimización numérica

4.1. Descenso por gradiente para minimizar $Q(\beta)$

Planteo:

El método de descenso por gradiente busca minimizar la función de error

$$Q(\beta) = \|A\beta - y\|^2,$$

mediante actualizaciones iterativas de los parámetros en la dirección de máximo descenso:

$$\beta_{k+1} = \beta_k - \alpha_k \nabla Q(\beta_k), \qquad \nabla Q(\beta) = 2A^\mathsf{T}(A\beta - y).$$

Como Q es cuadrática y estrictamente convexa cuando $A^\mathsf{T}A$ es definida positiva, el método garantiza convergencia al mínimo global.

Implementación:

Se implementó una función `descenso_gradiente_Q` que permite utilizar dos esquemas de paso:

(i) un paso fijo $\alpha_k = \eta$, y

(ii) un paso óptimo, dado por

$$\alpha_k = \frac{\nabla Q_k^\mathsf{T} \nabla Q_k}{2\, \nabla Q_k^\mathsf{T} (A^\mathsf{T}A) \nabla Q_k},$$

que es la elección que minimiza Q a lo largo de la dirección del gradiente en cada iteración.

En este trabajo se empleó el paso óptimo, con tolerancia tol $= 10^{-12}$ y un máximo de 10^4 iteraciones.

Resultados:

El método converge al vector de parámetros:

$$\beta_{\text{GD}} = \begin{bmatrix} 1.031 \\ 0.917 \\ -0.014 \end{bmatrix}$$

$$\text{MSE}_{\text{final}} = 0.002$$

$$\text{Iteraciones} = 265$$

$$\|\nabla Q\| \approx 9.85 \times 10^{-13}$$

Discusión:

La solución obtenida coincide numéricamente con la calculada mediante ecuaciones normales y mediante SVD, lo cual es consistente con la convexidad estricta de $Q(\beta)$.

El uso del paso óptimo hace que el algoritmo sea eficiente y estable, evitando la necesidad de ajustar manualmente la tasa de aprendizaje.

Dado que Q es una función cuadrática, el descenso por gradiente no solo converge al mínimo global, sino que lo hace de forma predecible, reduciendo la norma del gradiente a niveles prácticamente nulos.

4.2. Experimentos con tasas de aprendizaje y tolerancias

Barrido de tasas de aprendizaje (η).

Se evaluó el comportamiento del descenso por gradiente con paso constante para distintos valores de η:

$$\eta \in \{10^{-4},\, 5\times10^{-4},\, 10^{-3},\, 5\times10^{-3},\, 10^{-2},\, 5\times10^{-2}\}.$$

El algoritmo se ejecutó con un máximo de 50 000 iteraciones y tolerancia $\texttt{tol} = 10^{-10}$.

Los resultados obtenidos fueron:

η	Iteraciones	MSE final	$\|\nabla Q\|$
1×10^{-4}	50 000	1.8286×10^{-3}	4.827×10^{-4}
5×10^{-4}	39 375	1.8286×10^{-3}	9.995×10^{-11}
1×10^{-3}	19 682	1.8286×10^{-3}	9.999×10^{-11}
5×10^{-3}	50 000	`nan`	`nan`
1×10^{-2}	50 000	`nan`	`nan`
5×10^{-2}	50 000	`nan`	`nan`

Para $\eta \leq 10^{-3}$ el método converge correctamente al mismo mínimo obtenido anteriormente.

Para valores mayores, el gradiente crece sin control y la iteración diverge, lo que se refleja en la aparición de valores `nan`. Esto ilustra la sensibilidad del descenso por gradiente a la elección de la tasa de aprendizaje.

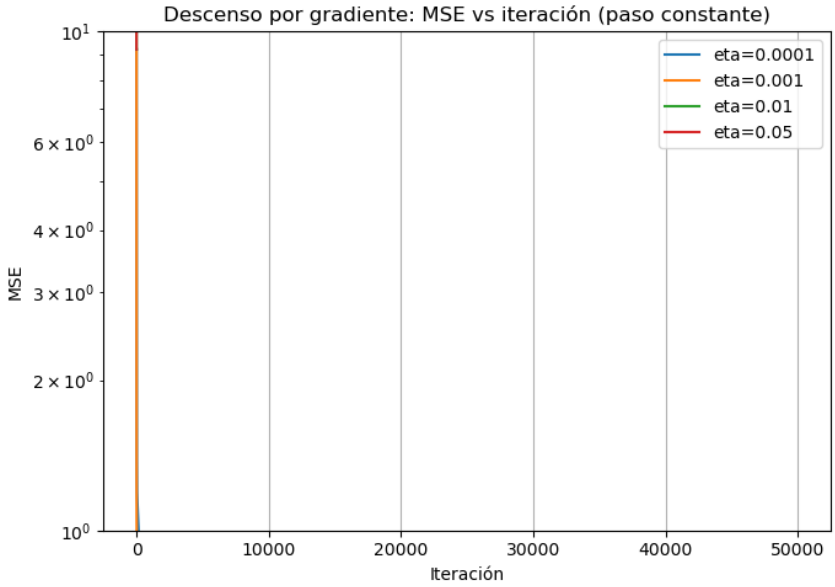

Figura 4: Evolución del MSE en función de las iteraciones del descenso por gradiente.

Barrido de tolerancias.

Con el paso óptimo, se estudió la influencia de la tolerancia de parada sobre la precisión y el número de iteraciones requeridas:

tol	Iteraciones	MSE final	$\|\nabla Q\|$
10^{-6}	123	1.8286×10^{-3}	9.770×10^{-7}
10^{-8}	173	1.8286×10^{-3}	8.522×10^{-9}
10^{-10}	221	1.8286×10^{-3}	8.962×10^{-11}
10^{-12}	265	1.8286×10^{-3}	9.854×10^{-13}
10^{-14}	100 000	1.8286×10^{-3}	3.352×10^{-14}

En todos los casos el valor de la función de error converge al mismo mínimo, pero una tolerancia más estricta exige más iteraciones.

En el caso extremo tol $= 10^{-14}$, el algoritmo alcanza la cota máxima de iteraciones estipulada (100 000), a pesar de que la norma del gradiente ya es extremadamente pequeña.

Discusión:

Se observa que el método converge siempre al mismo vector de parámetros

$\beta \approx [1.031, 0.917, -0.014]$, pero el número de iteraciones depende fuertemente de la elección de η y de la tolerancia.

Una tasa de aprendizaje demasiado grande produce inestabilidad numérica, mientras que una tolerancia excesivamente exigente solo incrementa el costo computacional sin mejorar de manera apreciable la calidad del ajuste.

En contraste, el descenso por gradiente con paso óptimo ofrece un buen trade-off entre velocidad y precisión, convergente al mínimo global en un número moderado de iteraciones.

4.3. (Opcional) Convergencia respecto de la solución por SVD

Planteo:

Se compararon las soluciones obtenidas mediante:

(i) ecuaciones normales, $\beta_{\text{NE}} = (A^\top A)^{-1} A^\top y$;

(ii) descomposición en valores singulares (SVD), $\beta_{\text{SVD}} = V \Sigma^+ U^\top y$; y

(iii) descenso por gradiente con paso óptimo.

Se evaluaron los coeficientes, las predicciones $\hat{y} = A\beta$, el MSE y las diferencias relativas entre los métodos.

Resultados numéricos:

Los tres procedimientos convergen al mismo vector de parámetros:

$$\beta_{\text{NE}} = \beta_{\text{SVD}} = \beta_{\text{GD}} \approx \begin{bmatrix} 1.031 \\ 0.917 \\ -0.014 \end{bmatrix}.$$

Los valores de MSE coinciden (hasta cierto redondeo):

$$\text{MSE}_{\text{NE}} \approx 1.829 \times 10^{-3}, \quad \text{MSE}_{\text{SVD}} \approx 1.829 \times 10^{-3}, \quad \text{MSE}_{\text{GD}} \approx 1.829 \times 10^{-3}.$$

Diferencias relativas:

Las diferencias relativas entre los vectores de parámetros son del orden de la precisión de máquina:

$$\frac{\|\beta_{\text{NE}} - \beta_{\text{SVD}}\|}{\|\beta_{\text{NE}}\|} = 2.525 \times 10^{-15}$$

$$\frac{\|\beta_{\text{NE}} - \beta_{\text{GD}}\|}{\|\beta_{\text{NE}}\|} = 6.819 \times 10^{-13}$$

$$\frac{\|\beta_{\text{SVD}} - \beta_{\text{GD}}\|}{\|\beta_{\text{SVD}}\|} = 6.844 \times 10^{-13}$$

Del mismo modo, las diferencias relativas entre las predicciones son insignificantes:

$$\frac{\|\hat{y}_{\text{NE}} - \hat{y}_{\text{SVD}}\|}{\|\hat{y}_{\text{NE}}\|} = 3.941 \times 10^{-16}$$

$$\frac{\|\hat{y}_{\text{NE}} - \hat{y}_{\text{GD}}\|}{\|\hat{y}_{\text{NE}}\|} = 1.004 \times 10^{-13}$$

$$\frac{\|\hat{y}_{\text{SVD}} - \hat{y}_{\text{GD}}\|}{\|\hat{y}_{\text{SVD}}\|} = 1.007 \times 10^{-13}$$

Verificación:

Dado que todas las diferencias relativas son $\ll 10^{-12}$, se verifica que:

$$\boxed{\beta_{\text{NE}} = \beta_{\text{SVD}} = \beta_{\text{GD}} \quad \text{y} \quad A\beta_{\text{NE}} = A\beta_{\text{SVD}} = A\beta_{\text{GD}}}$$

dentro de la tolerancia numérica del redondeo de la computadora.

Este resultado confirma que, al tratarse de una función cuadrática estrictamente convexa, los tres métodos (ecuaciones normales, SVD y descenso por gradiente con paso óptimo) encuentran exactamente el mismo mínimo global, tal como predice la teoría.

5. Discusión y extensiones

5.1. Relación entre los distintos métodos de resolución

Los tres métodos analizados (ecuaciones normales, descomposición en valores singulares (SVD) y descenso por gradiente) abordan el mismo problema de mínimos cuadrados, pero lo hacen desde perspectivas conceptuales y numéricas diferentes. A pesar de esto, cuando la matriz de diseño A posee rango completo, los tres procedimientos convergen al mismo vector de parámetros β^*, que minimiza la función de error $Q(\beta) = \|A\beta - y\|^2$.

Ecuaciones normales:

Se obtiene el minimizador imponiendo la condición $\nabla Q(\beta) = 0$, lo cual produce el sistema lineal $(A^\mathsf{T}A)\beta = A^\mathsf{T}y$.

Este método es eficiente para matrices pequeñas, pero su estabilidad numérica depende del condicionamiento de $A^\mathsf{T}A$.

Si las columnas de A son casi colineales, el sistema puede volverse inestable y amplificar errores.

SVD:

La descomposición $A = U\Sigma V^\mathsf{T}$ proporciona una forma numéricamente robusta de resolver problemas de mínimos cuadrados, ya que $A^+ = V\Sigma^+U^\mathsf{T}$.

El SVD separa las direcciones principales de variación, permitiendo detectar explícitamente la presencia de valores singulares pequeños, responsables del mal condicionamiento.

Por esta razón, es el método más estable y el recomendado en aplicaciones de mayor escala o cuando la matriz de diseño presenta colinealidad.

Descenso por gradiente:

A diferencia de los métodos anteriores, no requiere invertir matrices ni factorizar A.

En lugar de ello, aproxima iterativamente el minimizador mediante movimientos en la dirección del gradiente del error.

Para funciones cuadráticas como la estudiada, el método converge al mínimo global con paso óptimo.

Si bien puede ser menos eficiente para problemas pequeños, es fundamental en contextos donde A es muy grande o dispersa, donde factorizar o invertir matrices resulta impracticable.

En esos casos, el descenso por gradiente es el método estándar.

Conclusión:

Aunque conceptualmente distintos, los tres métodos conducen al mismo minimizador cuando A tiene rango completo:

$$\beta_{\text{NE}} = \beta_{\text{SVD}} = \beta_{\text{GD}}.$$

La diferencia radica en la forma en que cada método enfrenta la estructura algebraica y el condicionamiento del problema.

Ecuaciones normales son eficientes pero sensibles al mal condicionamiento; SVD es el método más estable y revelador desde el punto de vista geométrico; y el descenso por gradiente ofrece una alternativa escalable y flexible para problemas de gran tamaño.

5.2. Efecto de agregar términos cúbicos o ruido en el ajuste

Mayor complejidad del modelo (término cúbico) sin ruido:

A partir de los datos originales

$$(x_i, y_i) = \{(0, 1), (1, 2), (2, 2.8), (3, 3.6), (4, 4.5)\},$$

se compararon dos modelos:

$$\text{Cuadrático: } y \approx \beta_0 + \beta_1 x + \beta_2 x^2, \qquad \text{Cúbico: } y \approx \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3.$$

Ajustando por ecuaciones normales, se obtienen:

$$\beta^{(2)} = \begin{bmatrix} 1.032 \\ 0.917 \\ -0.014 \end{bmatrix}, \qquad \text{MSE}^{(2)}_{\text{train}} = 1.83 \times 10^{-3},$$

$$\beta^{(3)} = \begin{bmatrix} 1.001 \\ 1.132 \\ -0.164 \\ 0.025 \end{bmatrix}, \qquad \text{MSE}^{(3)}_{\text{train}} = 2.86 \times 10^{-5}.$$

El modelo cúbico logra un error de entrenamiento mucho menor, debido a su mayor flexibilidad: posee un parámetro adicional y es capaz de ajustarse casi exactamente a los cinco puntos disponibles.

Efecto del ruido - evaluación fuera de la muestra:

Para estudiar el impacto del ruido, se tomó como "verdad subyacente" el modelo cuadrático ajustado y se generaron datos perturbados de la forma:

$$y_i^{(\text{ruido})} = y_{\text{true}}(x_i) + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma^2).$$

En cada repetición se ajustaron ambos modelos (cuadrático y cúbico) y se evaluaron sobre una grilla de 200 puntos en $[0, 4]$, comparando contra $y_{\text{true}}(x)$.

Se realizaron 500 experimentos para cada nivel de ruido σ.

σ	RMSE test (cuadrático)	RMSE test (cúbico)
0.02	0.013	0.015
0.05	0.031	0.038
0.10	0.062	0.075
0.20	0.122	0.149

Discusión:

Mientras que el modelo cúbico reduce fuertemente el error de entrenamiento (sobreajuste deliberado a los datos), su desempeño fuera de la muestra se deteriora cuando existe ruido: el RMSE de test es sistemáticamente mayor que el del modelo cuadrático para todos los valores de σ.

Esto indica que el término cúbico captura variaciones generadas por el ruido.

En contraste, el modelo cuadrático presenta un error in–sample algo mayor, pero obtiene un RMSE de test consistentemente menor, lo que refleja una mejor capacidad de generalización.

Este comportamiento ilustra el clásico compromiso entre sesgo y varianza:

- el modelo cúbico tiene menor sesgo pero mayor varianza;
- el modelo cuadrático tiene mayor sesgo pero menor varianza.

En presencia de ruido, la menor varianza del modelo cuadrático conduce a un desempeño superior sobre datos no vistos (datos de testeo, que no son con los que se entrenó)

6. Aplicación a un caso real: predicción de salario

6.1. Descripción del conjunto de datos

En esta sección aplicamos los métodos desarrollados previamente a un conjunto de datos reales (*Salary Prediction Dataset*), cuyo objetivo es predecir el salario anual de un empleado a partir de un conjunto de características personales y laborales. Las variables disponibles son:

- `Age`: edad del empleado.
- `Gender`: género (*Male* o *Female*).
- `Education_level`: nivel educativo alcanzado.
- `Job_title`: puesto de trabajo.
- `Years_of_experience`: años de experiencia laboral.
- `Salary`: salario anual (variable objetivo).

Nuestro objetivo es construir un modelo lineal que permita estimar el salario utilizando los métodos de ajuste, factorización matricial y optimización desarrollados en las secciones anteriores del informe.

6.2. Preprocesamiento y formulación matricial

Partimos de las variables numéricas (`Age`, `Years_of_experience`) y de las variables categóricas (`Gender`, `Education_level`, `Job_title`). Tratamos las variables categóricas, obteniendo una matriz de diseño

$$X \in \mathbb{R}^{n \times p},$$

donde cada columna representa una característica numérica estandarizada o una variable binaria indicadora correspondiente a una categoría.

El modelo lineal a ajustar es

$$y \approx X\beta,$$

donde $y \in \mathbb{R}^n$ es el vector de salarios y $\beta \in \mathbb{R}^p$ contiene el término independiente y los coeficientes asociados a cada predictor.

Dividimos los datos en un conjunto de entrenamiento (80 %) y otro de test (20 %).

Las variables numéricas se estandarizan utilizando la media y el desvío estándar del conjunto de entrenamiento, y se agrega una columna de unos para el intercepto, obteniendo finalmente la matriz

$$A = \begin{bmatrix} 1 & X_{\text{std}} \end{bmatrix}.$$

6.3. Problemas de colinealidad y mal condicionamiento

La inclusión de un número elevado de variables binarias indicadoras genera alta colinealidad en la matriz de diseño. El número de condición estimado para A_{train} es

$$\text{cond}(A_{\text{train}}) \approx 2.6 \times 10^{15},$$

valor que indica un mal condicionamiento severo. Como consecuencia, al intentar resolver el sistema normal

$$(A^{\mathsf{T}} A)\, \beta = A^{\mathsf{T}} y$$

mediante la eliminación de Gauss implementada en la Sección 1, la matriz $A^{\mathsf{T}} A$ resulta numéricamente singular y el método falla.

Por tal motivo, para el caso real recurrimos exclusivamente a los métodos basados en SVD y descenso por gradiente, que son numéricamente robustos frente a colinealidad extrema.

6.4. Ajuste mediante SVD y descenso por gradiente

Aplicamos la descomposición SVD de A_{train}:

$$A_{\text{train}} = U\, \Sigma\, V^{\mathsf{T}},$$

a partir de la cual obtenemos la solución de mínimos cuadrados mediante la seudoinversa:

$$\beta_{\text{SVD}} = V\, \Sigma^{+} U^{\mathsf{T}} y_{\text{train}}.$$

En paralelo, utilizamos el método de descenso por gradiente definido en la Sección 4, empleando paso óptimo. Como se verificó en la parte teórica del informe, este método converge al mismo mínimo que la solución obtenida mediante SVD.

En ambos casos se obtienen coeficientes prácticamente idénticos y métricas de desempeño coincidentes. En la Tabla 1 se resumen los resultados obtenidos en entrenamiento y test.

Los valores de coeficiente de determinación obtenidos son

Método	MSE (train)	MSE (test)
SVD	3.79×10^7	2.27×10^8
Descenso por gradiente	3.79×10^7	2.27×10^8

Cuadro 1: Métricas de desempeño en el conjunto real de salarios.

$$R^2_{\text{train}} \approx 0.98, \qquad R^2_{\text{test}} \approx 0.89,$$

lo que indica que el modelo captura una porción significativa de la variabilidad del salario, con un nivel moderado de sobreajuste esperable por la alta dimensionalidad del conjunto de datos.

6.5. Análisis de residuos e interpretación

En la Figura 1 comparamos los salarios reales frente a los salarios predichos para el conjunto de test. La mayoría de los puntos se alinean en torno a la diagonal $y = \hat{y}$, aunque se observan errores más grandes en los salarios más altos.

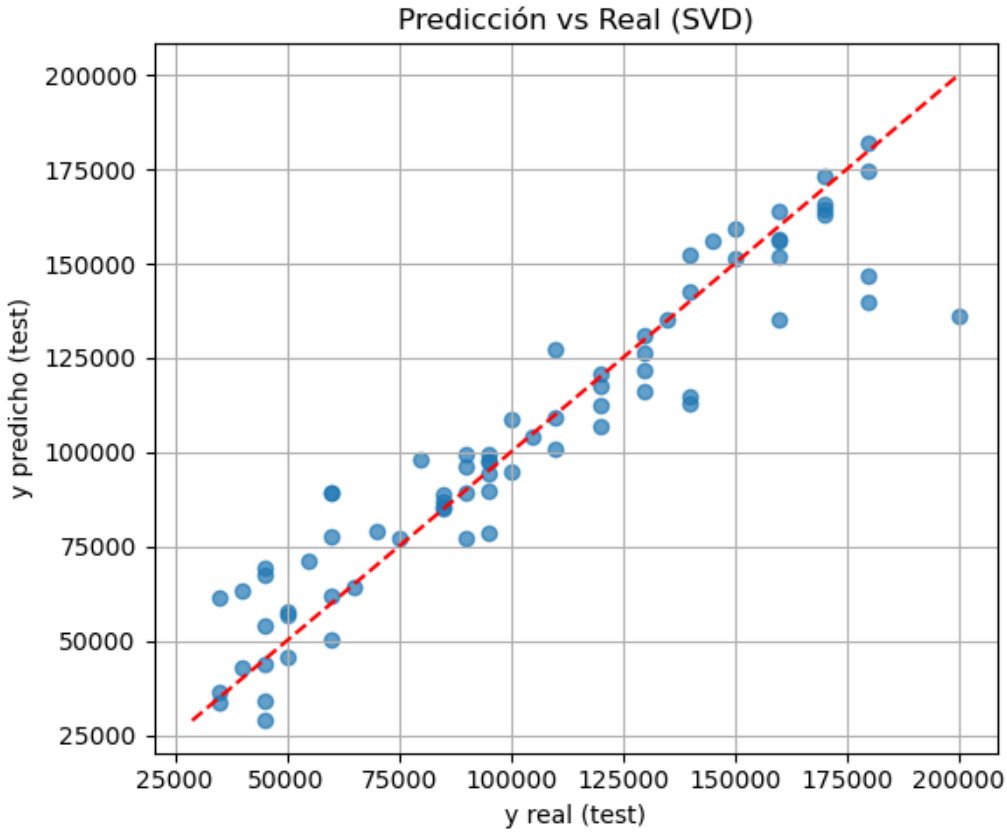

Figura 5: Comparación entre los valores reales y predichos de salario en el conjunto de test utilizando SVD.

La Figura 2 muestra el histograma de los residuos en test, que se distribuyen aproximadamente de forma simétrica alrededor de cero, con colas moderadas.

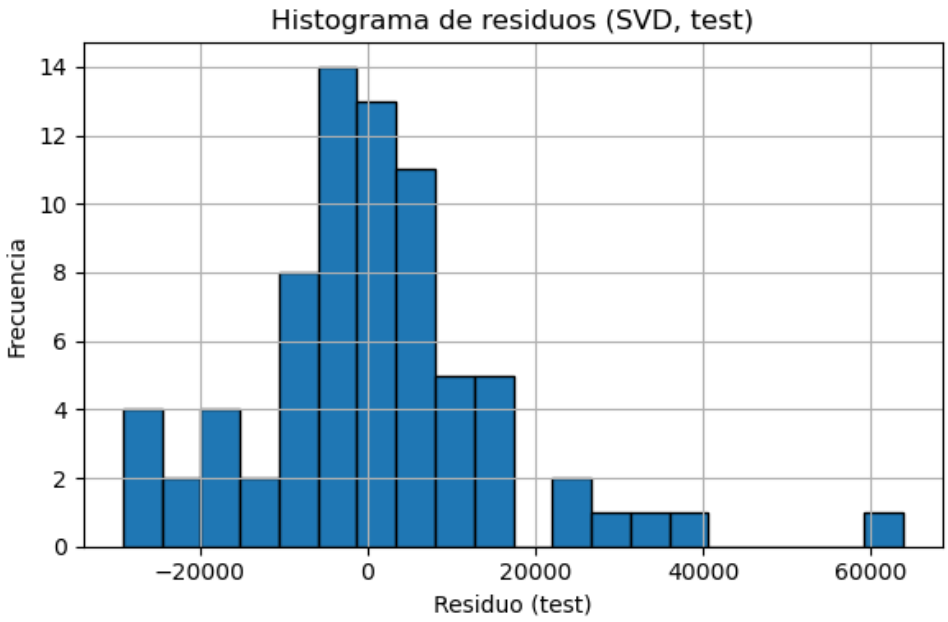

Figura 6: Histograma de residuos del modelo obtenido mediante SVD en el conjunto de test.

En cuanto a la interpretación de los parámetros, se observa que las variables numéricas `Age` y `Years_of_experience` presentan coeficientes positivos, lo cual indica que, en promedio, el salario aumenta con la edad y la experiencia laboral.

Las variables binarias indicadoras correspondientes al nivel educativo y al puesto de trabajo capturan diferencias sistemáticas entre grupos, reflejando que ciertos cargos y niveles académicos están asociados a salarios significativamente mayores.

Finalmente, el mal condicionamiento de la matriz de diseño remarca la importancia de utilizar métodos numéricamente estables como SVD o descenso por gradiente en problemas reales con muchas variables categóricas, donde las ecuaciones normales estándar pueden volverse inestables o directamente inviables.