Gradient Descent

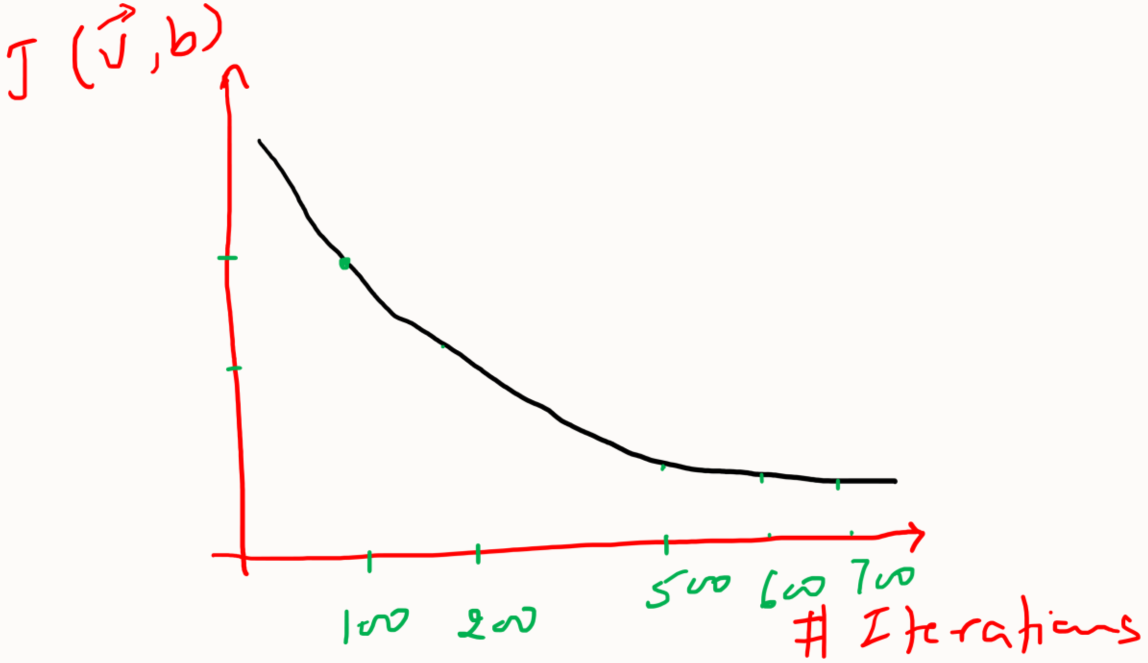
Learning Rate, Feature Engineering, Polynomial Regression

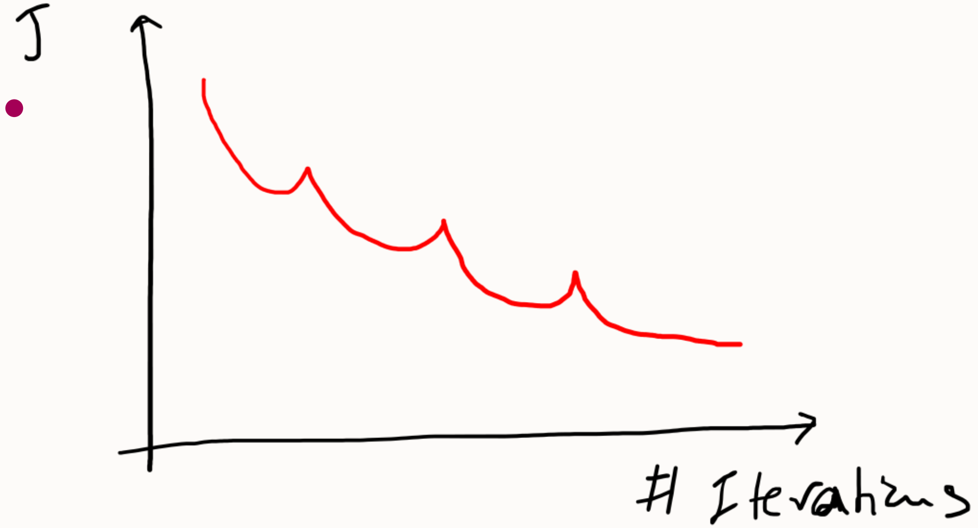
Convergence test

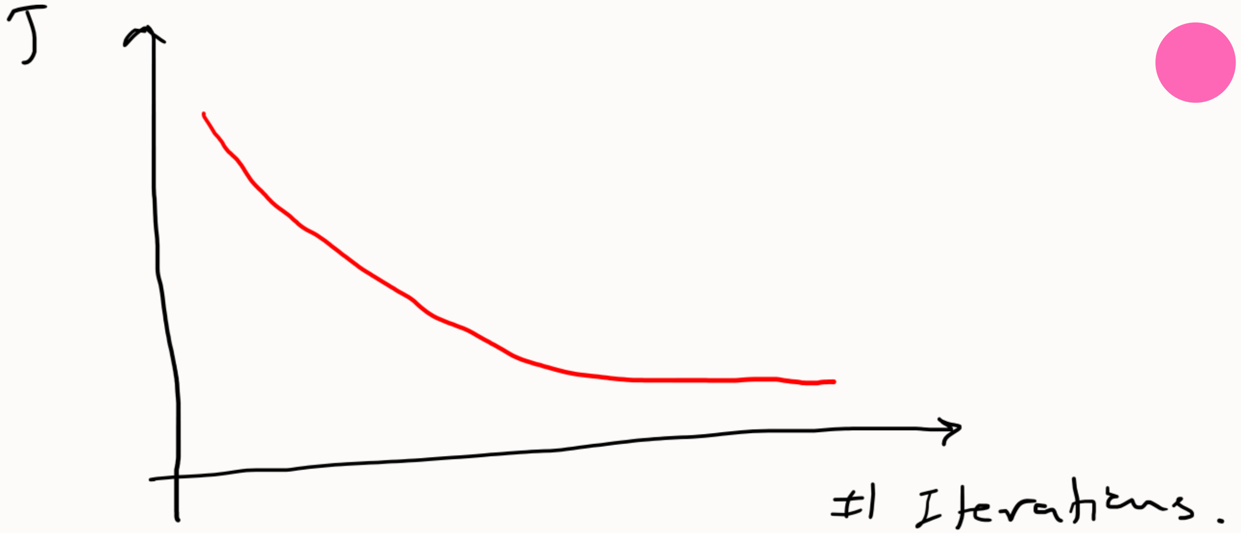
- Learning curve
- Automatic Convergence test

$$\varepsilon = 10^{-3}$$

$$\Delta J = J_{new} - J_{old} \leq \varepsilon$$

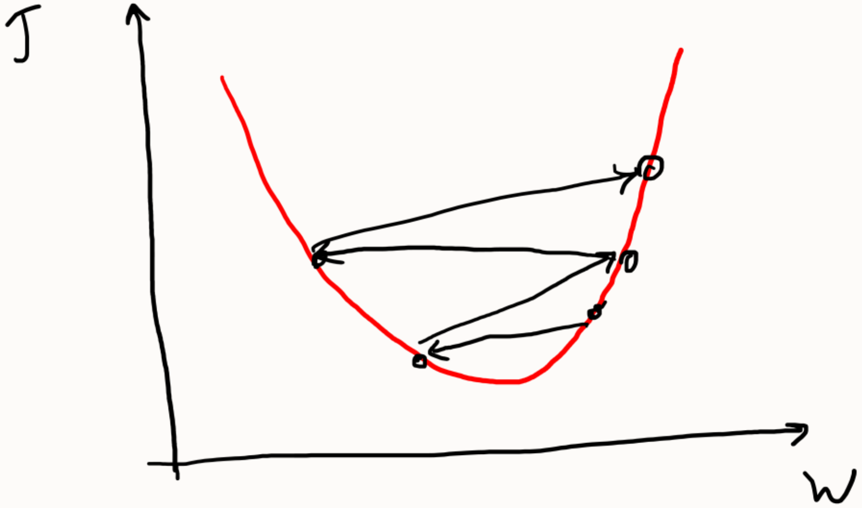


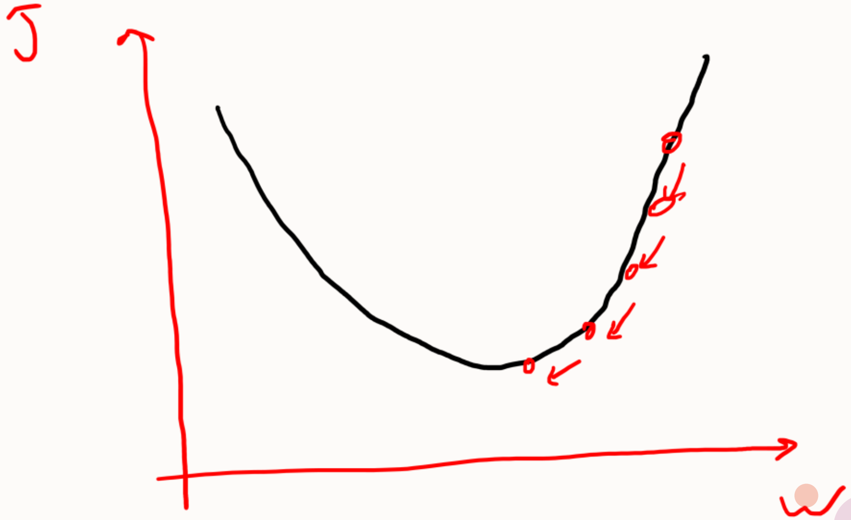




α too high.

α too small

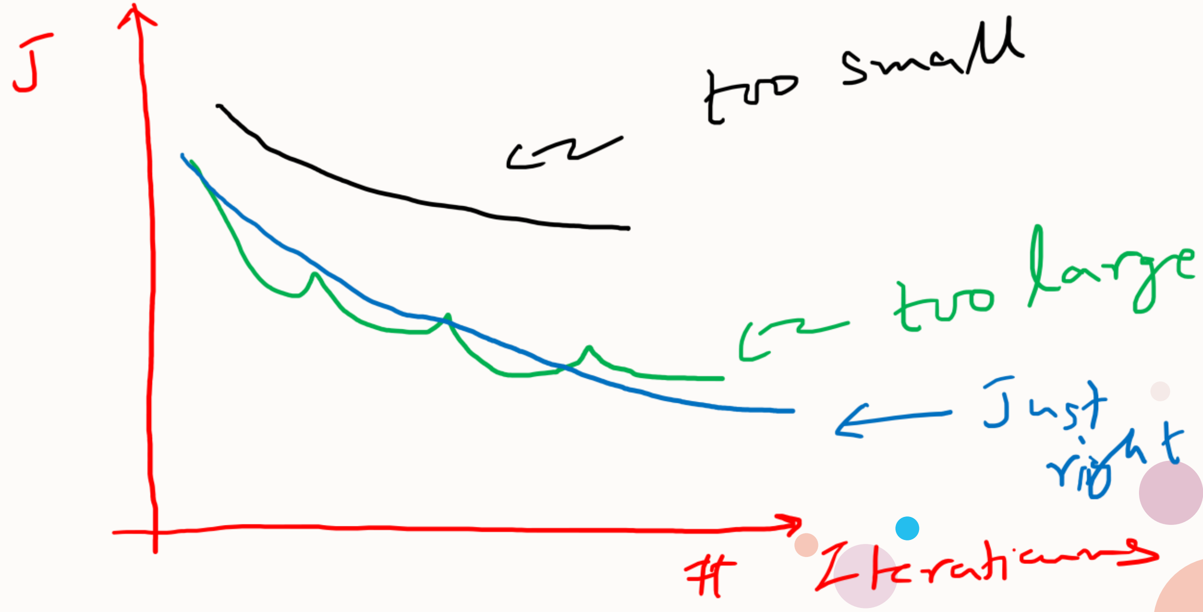




Learning rate (α)

0.001 0.003 0.01 0.03 0.1 0.3 1 3 10

÷3 ×3



Feature Engineering

Feature Scaling

Encoding Categorical Variables

Creating New Features

Employee_ID	Experience (Years)	Education Level	Age	Job Role	City	Working Hours/Week	Salary (₹)
1	2	Bachelor’s	25	Engineer	New Delhi	40	60000
2	5	Master’s	30	Manager	Mumbai	45	90000
3	8	PhD	35	Scientist	Banglore	50	120000
4	1	Bachelor’s	22	Engineer	Hyderabad	38	55000
5	10	Master’s	40	Manager	Kolkata	48	110000
6	NaN	PhD	45	Scientist	New Delhi	42	130000

Handling Missing Values

Encoding Categorical Variables

- Ordinal Encoding
- One-Hot Encoding
- Target Encoding

Ordinal Encoding

Maintains Order: PhD (2) > Master’s (1) > Bachelor’s (0), which reflects real-world hierarchy.
Reduces Complexity: Converts categorical values into numerical values for linear regression models.

- Increasing numbers to different levels of education

- Education Level (Bachelor’s, Master’s, PhD) → (0, 1, 2)

Employee_ID	Experience (Years)	Education Level (encoded)	Age	Job Role	City	Working Hours/Week	Salary (₹)
1	2	0	25	Engineer	New Delhi	40	60000
2	5	1	30	Manager	Mumbai	45	90000
3	8	2	35	Scientist	Banglore	50	120000
4	1	0	22	Engineer	Hyderabad	38	55000
5	10	1	40	Manager	Kolkata	48	110000
6	NaN	2	45	Scientist	New Delhi	42	130000

One-Hot Encoding

Works well for categories **without a meaningful order** (e.g., Job Role, City).
Ensures that categorical variables don't mislead linear regression models.
Prevents the model from assuming a ranking when there isn't one.

- One-Hot Encoding is used when categorical values do not have a meaningful order.
- Job role:
- Engineer [1 0 0], Manager [0 1 0], Scientist [0 0 1]

Employee_ID	Experience (Years)	Education Level	Age	Job_Engineer	Job_Manager	Job_Scientist	City	Working Hours/Week	Salary (₹)
1	2	Bachelor's	25	1	0	0	New Delhi	40	60000
2	5	Master's	30	0	1	0	Mumbai	45	90000
3	8	PhD	35	0	0	1	Banglore	50	120000
4	1	Bachelor's	22	1	0	0	Hyderabad	38	55000
5	10	Master's	40	0	1	0	Kolkata	48	110000
6	NaN	PhD	45	0	0	1	New Delhi	42	130000

Target Encoding

- Target encoding is an encoding technique where we replace categorical variables with the mean of the target variable
- Step 1: Compute the Mean Salary for Each City
- Step 2: Replace City with Its Corresponding Average Salary

City	Mean Salary (₹)
New Delhi	95000
Mumbai	90000
Banglore	120000
Hyderabad	55000
Kolkata	110000

Employee_ID	Experience (Years)	Education Level	Age	Job Role	City (encoded)	Working Hours/Week	Salary (₹)
1	2	Bachelor’s	25	Engineer	95000	40	60000
2	5	Master’s	30	Manager	90000	45	90000
3	8	PhD	35	Scientist	120000	50	120000
4	1	Bachelor’s	22	Engineer	55000	38	55000
5	10	Master’s	40	Manager	110000	48	110000
6	NaN	PhD	45	Scientist	95000	42	130000

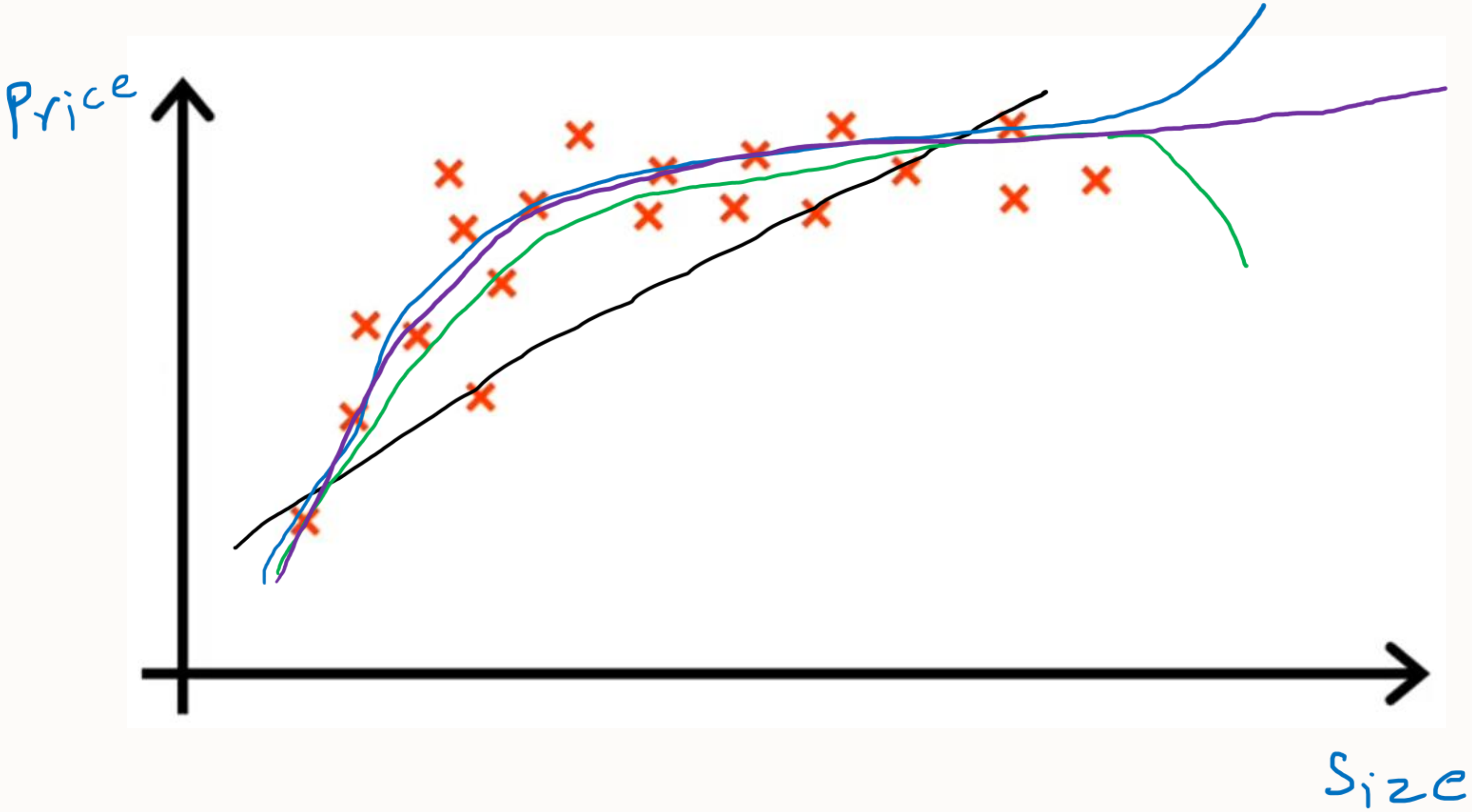
Creating New Features

- Seniority Level: If Experience (Years) > 7, create Is_Senior = 1, else 0
- Age Group: Convert Age into categories (Young: 20-30, Mid: 30-40, Senior: 40+)
- Work-Life Balance Score: Work_Hours_Per_Week / Age (higher values indicate more work pressure)

$$f = w_1 x_1 + w_2 x_2 + b$$

$$f = w_1 x_1 + w_2 x_2 + w_3 x_3 + b$$

Polynomial regression



$$f = v x + b$$

$$f = w_1 x + w_2 x^2 + b$$

$$f = w_1 x + w_2 x^2 + w_3 x^3 + b$$

$$10^3 \qquad 10^6 \qquad 10^9$$

$$f = v_1 x + w_2 \sqrt{x} + \dots + b$$