* Cost function

- It is average of loss functions over the entire dataset.
- It help us reach the optimal solution.
- It is techique to evaluate the performance of our algorithm.
- Our strategy would be to minimize the cost.

$$\text{cost} \propto 1/\text{Accuracy of model}.$$

- Their are various cost functions.

[A] Regression Cost function ($J(\theta_0, \theta_n)$)

- Used for regression models.

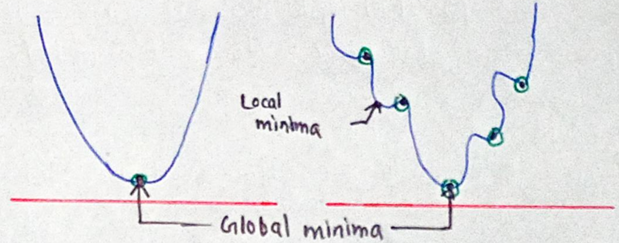
[A.1] Mean Squared Error (L2 Loss)

$$\boxed{J(\theta_0,\theta_n) = \frac{1}{n}\sum_{i=1}^{n}[h_\theta(x)^i - y^i]^2}$$

- Measured sum of squared differences of predicted and actual values
- Basically

$$\boxed{J(\theta_0,\theta_n) = \text{sum squared error} / n}$$

- As it is squared, it penlizes even small deviations in predictions, which means this cost function has only one global minima. i.e a convex function.



Due to local minima regressor could think it is best fit because neighbours are higher although a better option global mima is available.

- Not robust to outliers, as if outliers it will square the error leading to less accuracy.
- As it penalizes the error it squares the unit.

[A.2] Mean Absolute Error (L1 Loss)

$$\boxed{J(\theta_0,\theta_n) = \frac{1}{n}\sum_{i=1}^{n} |h_\theta(x)^i - y^i|}$$

- Measured sum of modulo of differences between predicted and actual value.
- Basically,

$$\boxed{J(\theta_0,\theta_n) = \text{sum modulo error} / n}$$

- As it dont square the errors its robust to outliers and units also dont get squared.
- But convergence usually take more time optimization. (Time consuming)

[A.3] Root mean squared error

$$\boxed{J(\theta_0,\theta_n) = \left[\frac{1}{n}\sum_{i=1}^{n}[h_\theta(x)^i - y^i]^2\right]^{1/2}}$$

- Measured root of square mean error
- Basically,

$$\boxed{J(\theta_0,\theta_n) = (L_2\,\text{Loss})^{1/2}}$$

- Doesnt penalize the error as much as L2Loss
- It is time optimized.