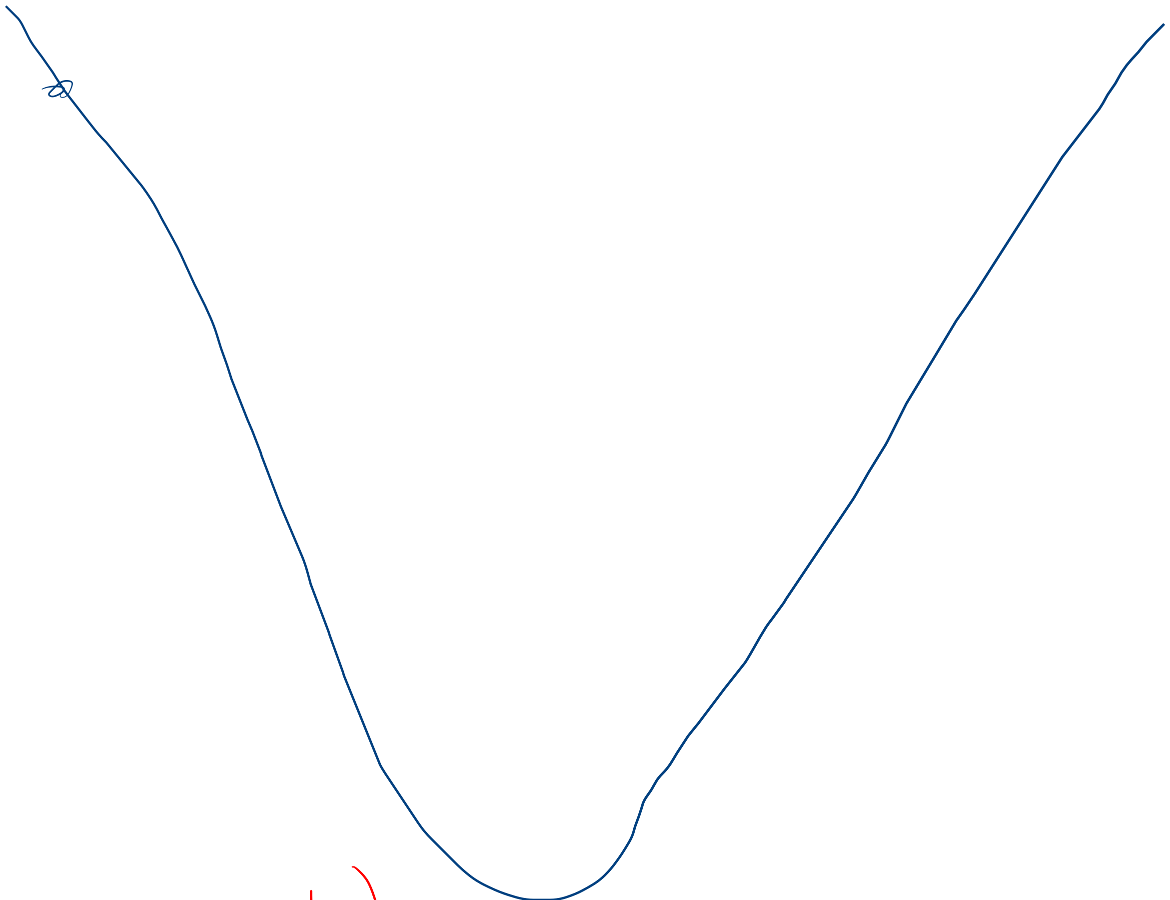
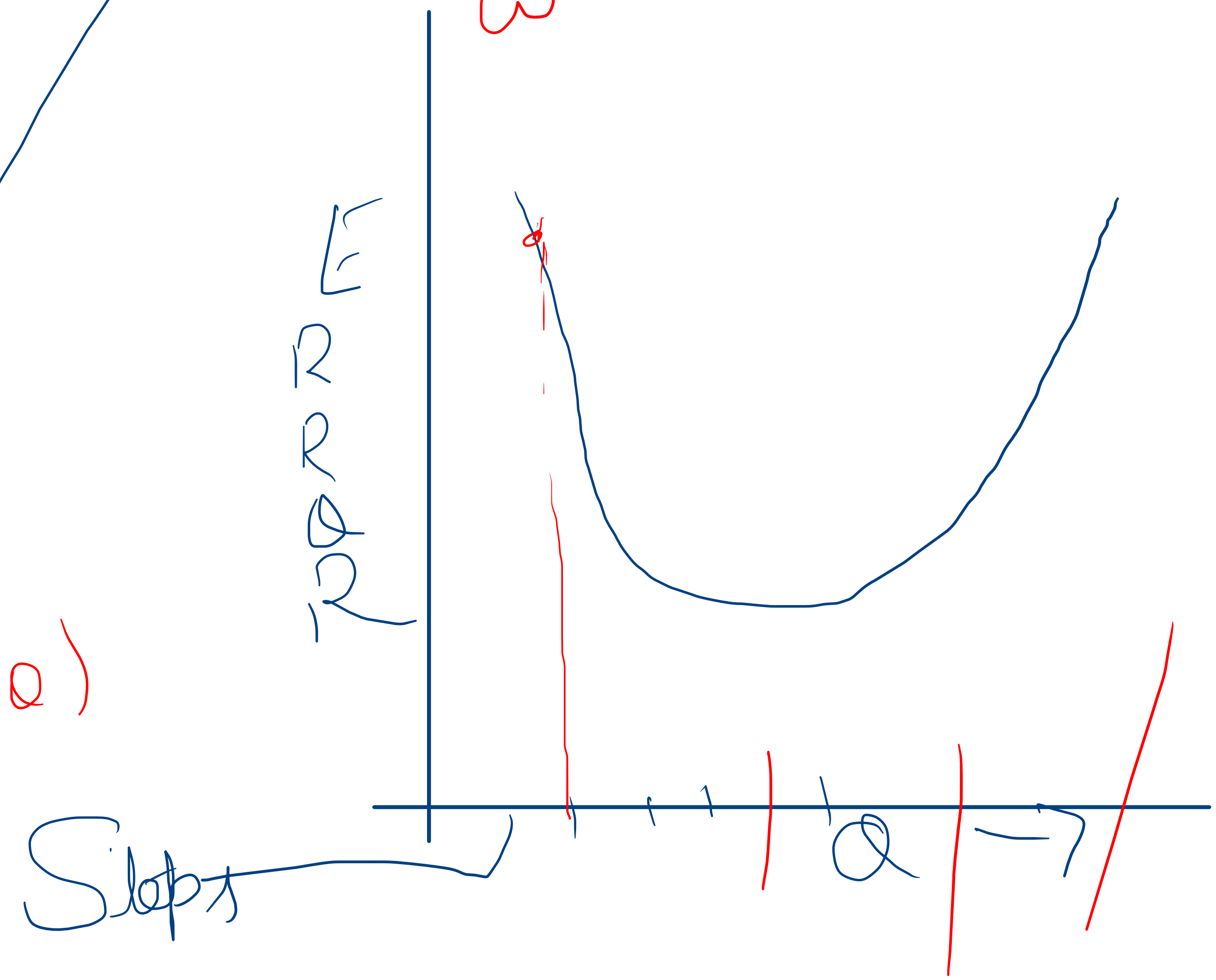


$\nabla_\theta(MSE(\theta)) =$ Change in MSE with respect to θ



$$\theta^{(\text{next step})} = \theta - \eta \nabla_\theta MSE(\theta)$$

$$\theta^{(new)} = \theta - \eta \nabla_\theta MSE(\theta)$$

If we are going in right direction the MSE is -ve

$$= \theta - (\eta)(-MSE)$$

$$= \theta + \underline{\eta MSE}$$

If I am going in wrong direction then

$$= \theta - (\eta)(MSE)$$

η = what should we put the value of η as it has to be manually input by us.

η = 1, .01, .00001, (.)

= $\boxed{.1}$

$$\hat{\theta} = \boxed{\left(X^T \cdot X\right)^{-1}} \cdot X^T \cdot y$$

Best parameters 1 go

x has n ~~fea~~ instances

That means $(X^T \cdot X)$ will give

$n \times n$ matrix.

To calculate inverse of $n \times n$ matrix
it requires $O(n^{2.4})$ time.

$O(n^{2.4})$ Time.

$O(n^3)$

Suppose we have a dataset of 1 feature & 1000 instances.

matrix size

(1000×1000)

$(1000)^3 = (10)^9$

We know python can do 10^7 calculations in 1 sec.

So to solve equation for 100 rows.

we will take $\frac{(10)^9}{10^7}$ seconds

$= 100$ sec.

$$(10{,}000)^3 = \frac{(10)^{12}}{10^7} \text{ sec.}$$

$$= \boxed{\frac{10{,}0000}{3600}} \text{ hour.}$$

$= 27.7$ hours

Tolerance :- (± 5)

if: d^{ment} < tolerance) then stop

$\boxed{800}$ [200] → $\boxed{0}$

y target

$\boxed{100}$ ✓

95 or 105

0^{rows} ≥ < (± 5)

For batch gradient descent with fixed learning rate the convergence rate of MSE is $O\left(\frac{1}{\text{iterations}}\right)$

ε = tolerance.

$\boxed{\text{Stochastic}}$