CMI LAB

UNIVERSITY OF WOLVERHAMPTON
KNOWLEDGE ▪ INNOVATION ▪ ENTERPRISE

6CS012 – Artificial Intelligence and Machine Learning

Ahsan Adeel

Theme lead, Conscious Multisensory Integration (CMI) Lab
Visiting fellow, MIT Synthetic Intelligence Lab, MIT
and Oxford Computational Neuroscience Lab, University of Oxford
Visiting EPSRC/MRC fellow, University of Stirling

http://www.cmilab.org/

http://www.cmilab.org/dr_ahsan_adeel.html#Home

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

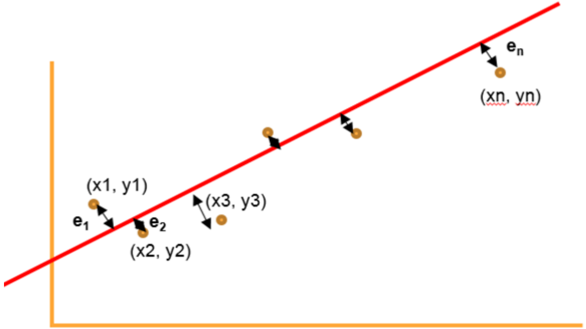
NUFFIELD DEPARTMENT OF SURGICAL SCIENCES
Medical Sciences Division

Lecture 3 – Gradient Descent

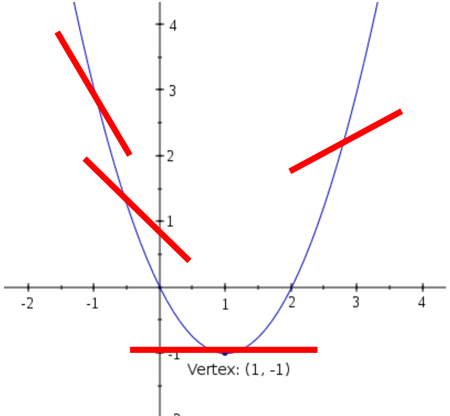
Lecture 2 Review:
Closed-Form Equation (Analytical Solution)

- SE = $(y_1 - (mx_1+b))^2 + (y_2 - (mx_2+b))^2 + \ldots\ldots, + (y_n - (mx_n+b))^2$
- Linear algebra application:

$$\boxed{SE = n\overline{y^2} - 2mn\,\overline{xy} - 2bn\bar{y} + m^2 n\overline{x^2} + 2mbn\bar{x} + nb^2}$$



- Find m and b that minimizes MSE
  - $\frac{d}{dm}$ (SE) = 0
  - $\frac{d}{db}$ (SE) = 0

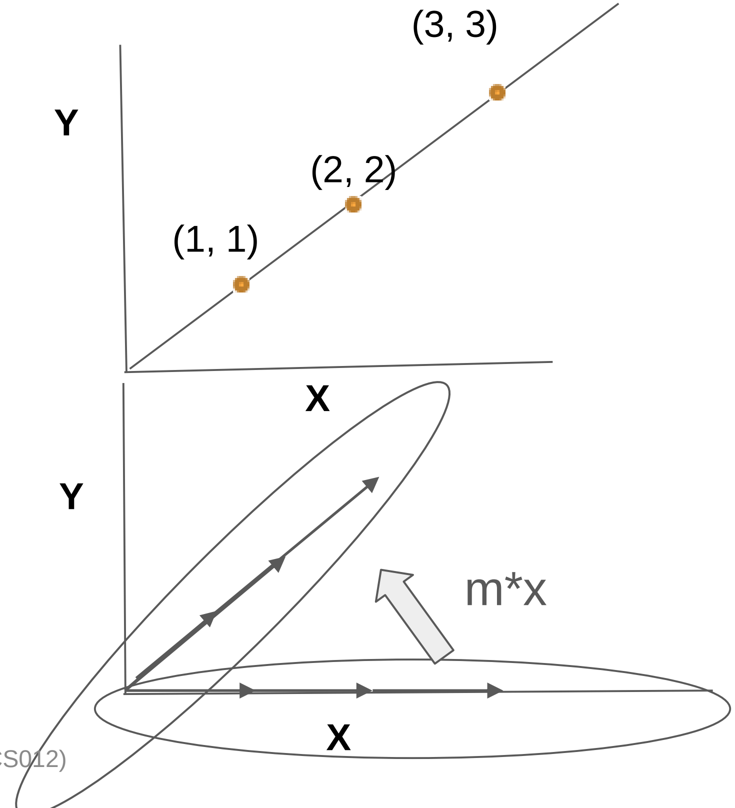


Lecture 2 Review:
Closed-Form Equation (Analytical Solution)

Linear Regression – WS task 3:

- Slope (m) = $(X^TX)^{-1} X^TY$
- $\hat{Y} = mx + c$

i.e. A linear transformation



Limitations of closed-form equation

1. Computational Complexity

- $m = (X^T X)^{-1} X^T Y$

- Here, we first calculate the matrix $X'X$ then invert it

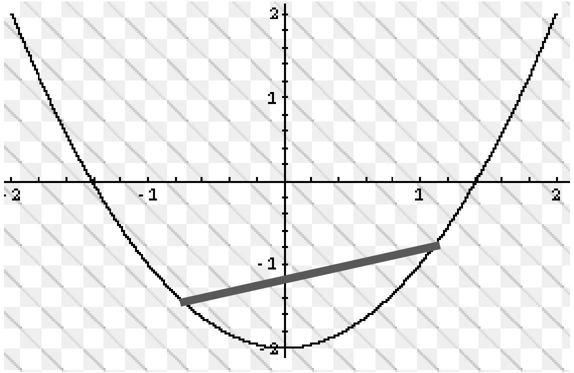
- If matrix X has K number of input variables (columns) and N rows of observations, it becomes an expensive calculation.

- In machine learning, we can end up with K>1000 and N>1,000,000.

Limitations of closed-form equation

2. Non-convex optimization

- The MSE cost function for a Linear Regression model is a convex function

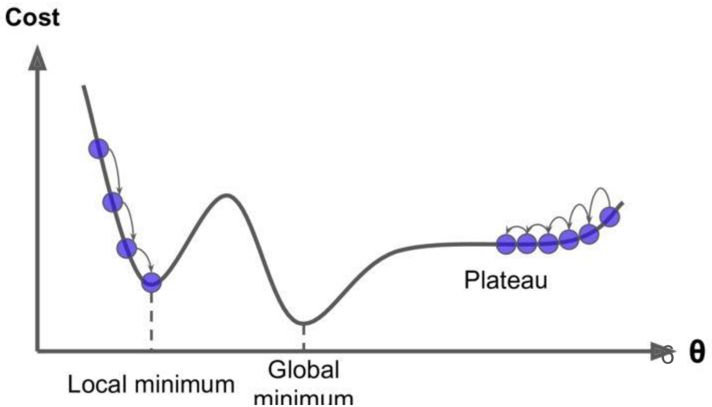


Convex function:

- If we pick any two points on the curve, the line segment joining them never crosses the curve.

- No local minima, just one global minimum.

Non-convex function: could have several local minima's



One of the Solutions: Stochastic Gradient Descent (SGD)

Lets first see what is GD?

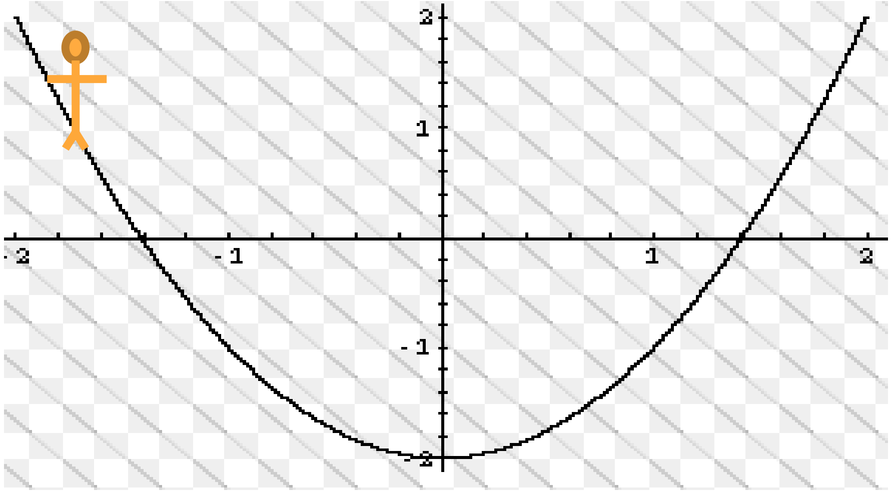
What’s GD?

“Imagine a person is stuck in the mountains and is trying to get down (i.e. trying to find the global minimum). There is heavy fog such that visibility is extremely low. Therefore, the path down the mountain is not visible, so they must use local information to find the minimum.”

https://en.wikipedia.org/wiki/Gradient_descent

What’s GD?

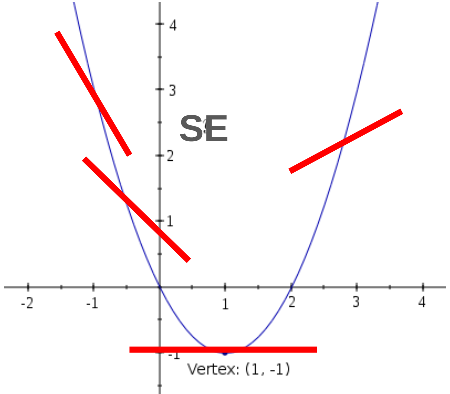
“They can use the method of gradient descent, which involves looking at the steepness of the hill at their current position, then proceeding in the direction with the steepest descent (i.e. downhill).”



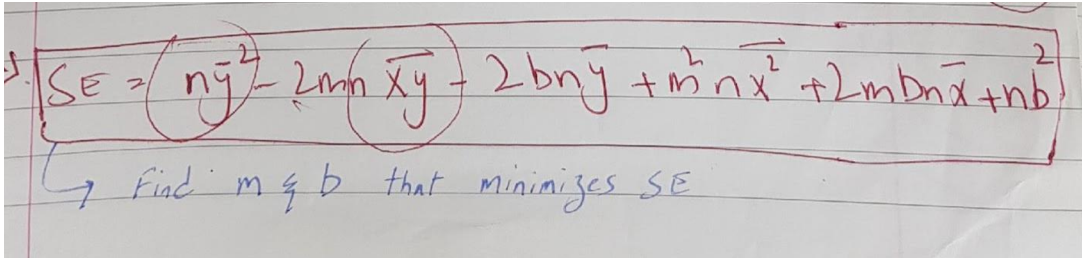
What’s GD?

“In this analogy, the person represents the algorithm, and the path taken down the mountain represents the sequence of parameter settings that the algorithm will explore.”

“The steepness of the hill represents the slope of the error surface at that point. The instrument used to measure steepness is differentiation (the slope of the error surface can be calculated by taking the derivative of the squared error function at that point).”



GD Mathematical Representation?



y = mx + b

y = **θ**x + b or y = wx + b

m, θ, and **w** are the same things

If we write MSE in terms of **θ**:

- $SE = e_1^2 + e_2^2 + e_3^2 +, \ldots\ldots., + e_n^2$
- $SE = (y_1 - (\boldsymbol{\theta}\, x_1 + b))^2 + (y_2 - (\boldsymbol{\theta}\, x_2 + b))^2 + \ldots\ldots., + (y_n - (\boldsymbol{\theta}\, x_n + b))^2$

$$\mathrm{MSE}\left(\mathbf{X}, h_{\boldsymbol{\theta}}\right) = \frac{1}{m} \sum_{i=1}^{m} \left(\boldsymbol{\theta}^T \mathbf{x}^{(i)} - y^{(i)}\right)^2$$

GD Mathematical Representation?

$$\text{MSE}(\mathbf{X}, h_{\boldsymbol{\theta}}) = \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{\theta}^T\mathbf{x}^{(i)} - y^{(i)}\right)^2$$

Vector of weights – recall vector of weights in lecture 2

For n input variables:

$$\nabla_{\boldsymbol{\theta}}\,\text{MSE}(\boldsymbol{\theta}) = \begin{pmatrix} \frac{\partial}{\partial\theta_0}\text{MSE}(\boldsymbol{\theta}) \\ \frac{\partial}{\partial\theta_1}\text{MSE}(\boldsymbol{\theta}) \\ \vdots \\ \frac{\partial}{\partial\theta_n}\text{MSE}(\boldsymbol{\theta}) \end{pmatrix} = \frac{2}{m}\mathbf{X}^T(\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$$

How many input variables and associated weights do we have in Boston housing dataset?

GD Mathematical Representation?

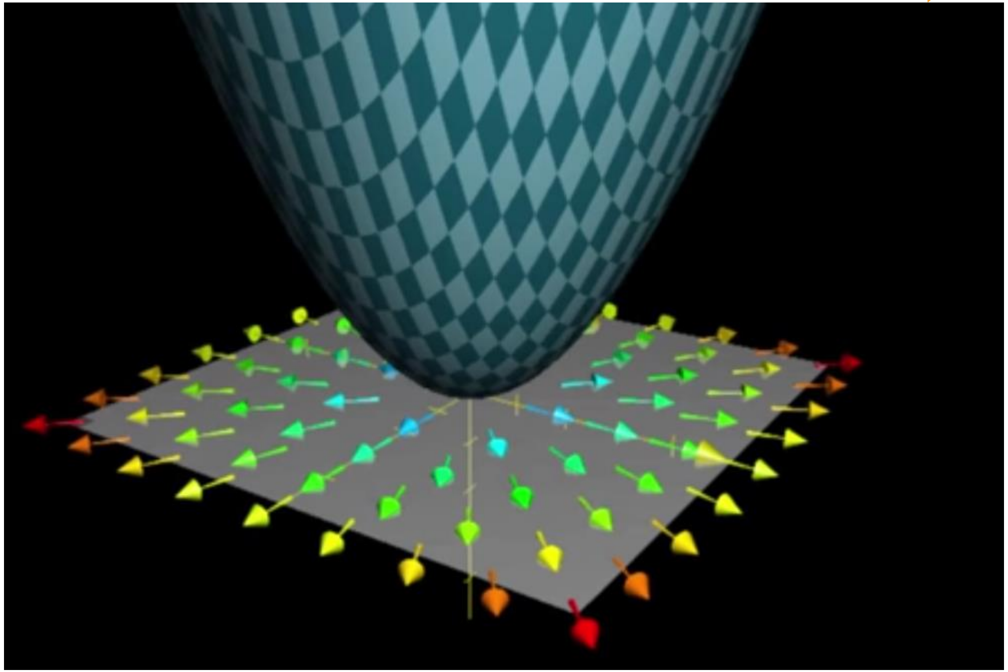
$$\text{MSE}(\mathbf{X}, h_{\boldsymbol{\theta}}) = \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{\theta}^T\mathbf{x}^{(i)} - y^{(i)})^2$$

Vector of weights – recall vector of weights in lecture 2

$$\nabla_{\boldsymbol{\theta}}\,\text{MSE}(\boldsymbol{\theta}) = \begin{pmatrix} \frac{\partial}{\partial\theta_0}\text{MSE}(\boldsymbol{\theta}) \\ \frac{\partial}{\partial\theta_1}\text{MSE}(\boldsymbol{\theta}) \\ \vdots \\ \frac{\partial}{\partial\theta_n}\text{MSE}(\boldsymbol{\theta}) \end{pmatrix} = \frac{2}{m}\mathbf{X}^T(\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$$

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

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



Example 1

$f(x) = x^2$

- Starting point: x= -2
- $\eta = 0.1$
- ∇ = [d/dx (f(x))] = 2x

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

X (next step) = X - $\boldsymbol{\eta}$ $\boldsymbol{\nabla}$f(X)

Iteration ‘1’

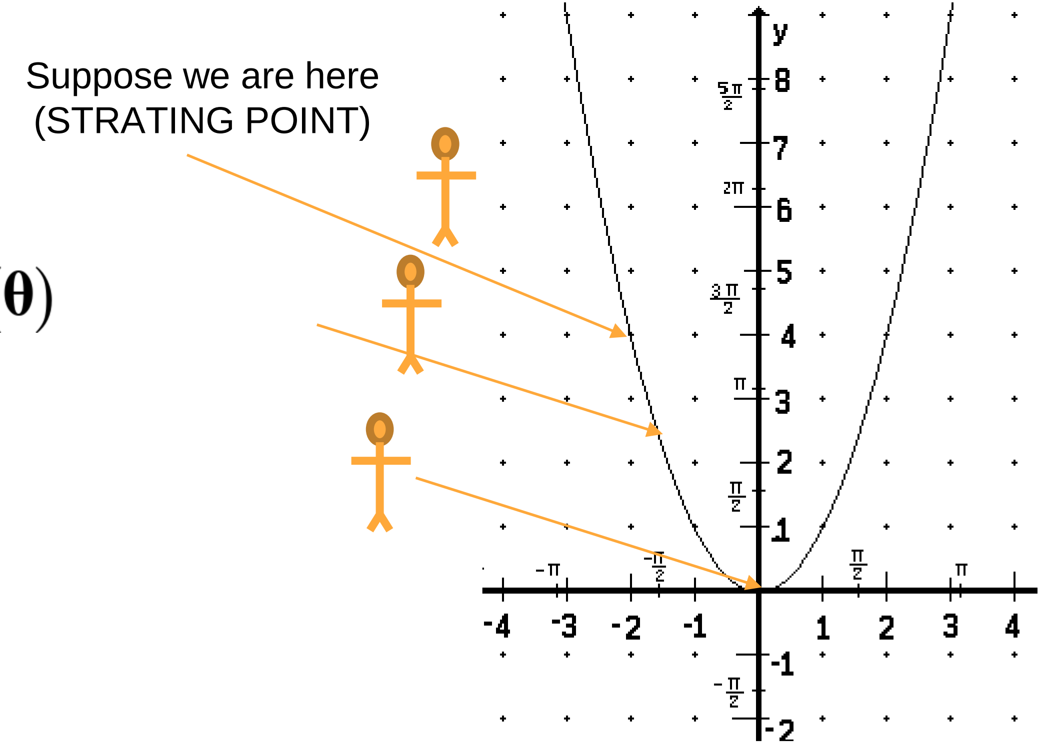
- X (next step) = (-2) – (0.1)(2(-2))
- X (next step) = -2 + 0.4 = -1.6

Iteration ‘2’

Iteration ‘3’

Iteration ‘4’

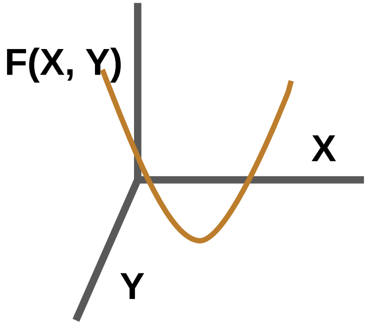
Iteration ‘n’

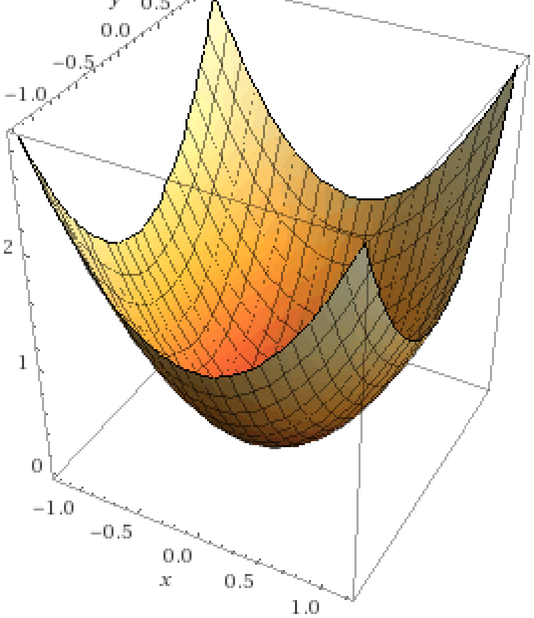


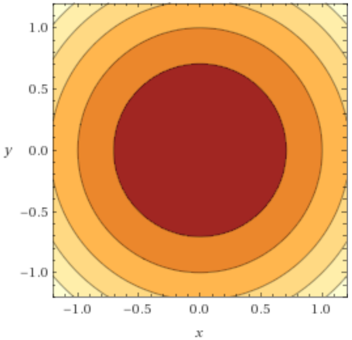
Example 2

- $f(x, y) = x^2 + y^2$
- $\nabla(f(x,y)) = \begin{pmatrix} \frac{d}{dx} f(x, y) \\ \frac{d}{dy} f(x, y) \end{pmatrix}$
- $\nabla(f(x,y)) = \begin{pmatrix} 2X \\ 2Y \end{pmatrix}$

X and Y (next step) = $\begin{pmatrix} X^{old} - \boldsymbol{\eta} 2X \\ Y^{old} - \boldsymbol{\eta} 2Y \end{pmatrix}$







WS-3 Tasks

1. **Task 1:**

a. **Explain the difference between closed-form analytical solution and GD?**
b. **Why the gradient = 0 in the closed-form analytical solution?**
c. **What's the difference between 'm' vector, 'θ' vector and '∇' vector.**
d. **What does ∇ vector tells you?**
e. **What does each element of the ∇ vector tells you?**
f. **Run example 1 on paper for 4 iterations**

2. Task 2: Run GD algorithm and find X and Y that minimizes the following function

f(x, y) = $x^2 + \log(\mathbf{SN})y^2$

- SN: last two digits of your student number

3. Task 3:

a. **Solve task 3 of WS-2 using GD**
b. **Compare both the solutions (equation and GD based)**

Stochastic Gradient Descent

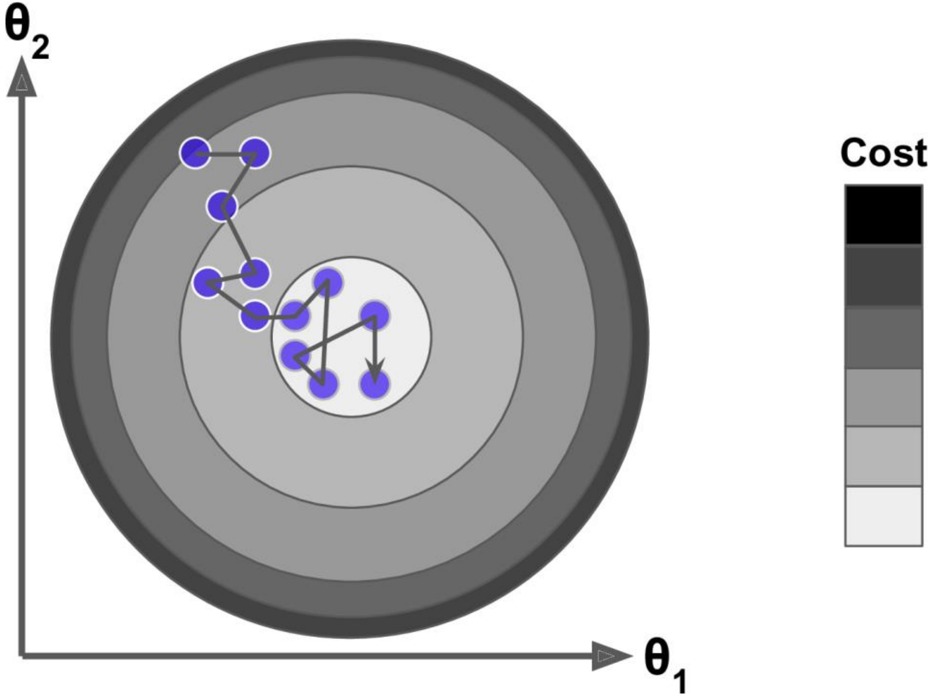
- Batch Gradient Descent Problem → uses the whole training set once to compute the gradient vector at every step.

- In contrast, the SGD randomly picks an instance from the training set at every step → instance based gradient vector calculation.

- SGD is much faster, since it has very little data to manipulate at every iteration.

- SGD is more feasible for training large Datasets, since only one instance needs to be in the memory at each iteration.

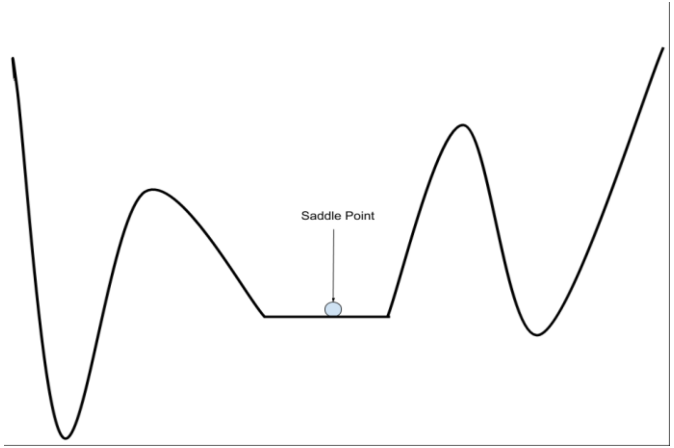
Stochastic Gradient Descent



Other issues

Other issues include:

- **Flat Regions (Saddle Points)**



- **High dimensional data**
    - Deep neural networks often have millions of parameters
    - High dementional data e.g. high resolution image, speech
    - Curse of dimensionality.

References

- https://en.wikipedia.org/wiki/Gradient_descent
- https://www.axonoptics.com/2015/06/light-triggers-migraines/
- Huff, Trevor, and Scott C. Dulebohn. "Neuroanatomy, Visual Cortex." (2017).
- Ian Goodfellow and Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 2016, url: http://www.deeplearningbook.org
- https://www.khanacademy.org/math/statistics-probabilit
- 3blue1brown: https://www.3blue1brown.com/
- http://neuralnetworksanddeeplearning.com/index.html