Building Blocks of Supervised Machine Learning II

Swati Mishra

Applications of Machine Learning (4AL3)

Fall 2024

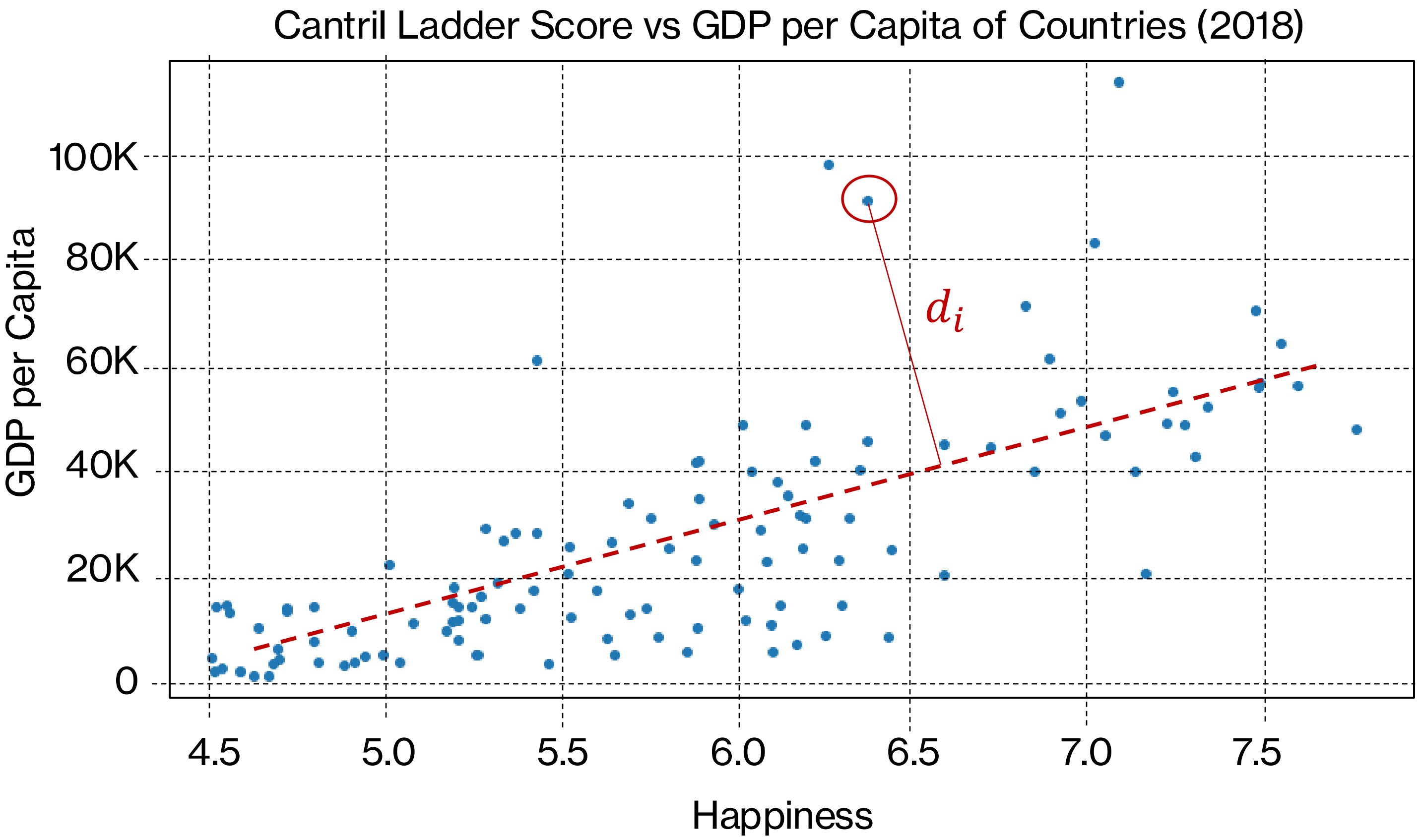
McMaster University

ENGINEERING

Review

- Recipe of Supervised Learning (Dataset + Cost Function + Optimizer + Model)
- Terminology in Machine Learning
- Parametric and Non-parametric models
- Linear Regression – Ordinary Least Squares

Review Linear Regression - OLS



Step 2:

Hypothesize a linear model

$$y' = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p$$

Step 3:

Select a Loss Function

$$\sum_{i=0}^{n} {d_i}^2 \quad => \quad MSE = \frac{1}{n} \sum_{n=1}^{n} (y_i - y_i')^2$$

Step 4:

Find β such that it minimizes loss function

$$\beta' = (X^T X)^{-1} X^T y$$

Review Linear Regression - OLS

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i')^2$

“Find β such that it minimizes the loss” means that

1. find derivative of K w.r.t. β
2. make it equal to 0

$$\nabla_\beta \mathrm{K}(\beta) = 0 \Rightarrow \cdots \Rightarrow \nabla_\beta \frac{1}{2}(X\beta - y)^T(X\beta - y) = 0 \Rightarrow \cdots \Rightarrow \beta = (X^T X)^{-1} X^T y$$

(Derived Value of $\beta = \beta$’, it may not be the true β, it is an approximation)

3. predicted value of $\beta' = (X^T X)^{-1} X^T y$

Review Linear Regression - OLS

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i')^2$

"Find β such that it minimizes the loss" means that

1. find derivative of K w.r.t. β
2. make it equal to 0

$$\nabla_\beta K(\beta) = 0 \Rightarrow \cdots \Rightarrow \nabla_\beta \frac{1}{2}(X\beta - y)^T(X\beta - y) = 0 \Rightarrow \cdots \Rightarrow \beta = (X^T X)^{-1} X^T y$$

(Derived Value of $\beta = \beta'$, it may not be the true β, it is an approximation)

3. predicted value of $\beta' = (X^T X)^{-1} X^T y$

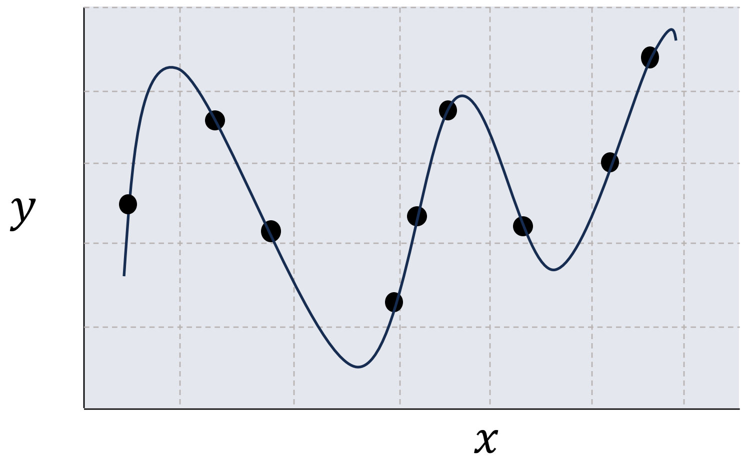
What if there are too many variables?

Gradient Descent

It is a very popular optimization algorithm used in machine learning to determine how the change in the input is impacting the output.

Gradient Descent

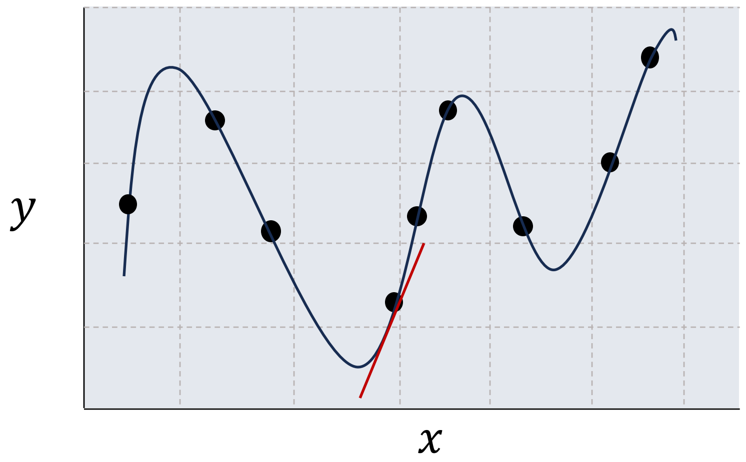
- Assume there is a function $y = f(x)$



Gradient Descent

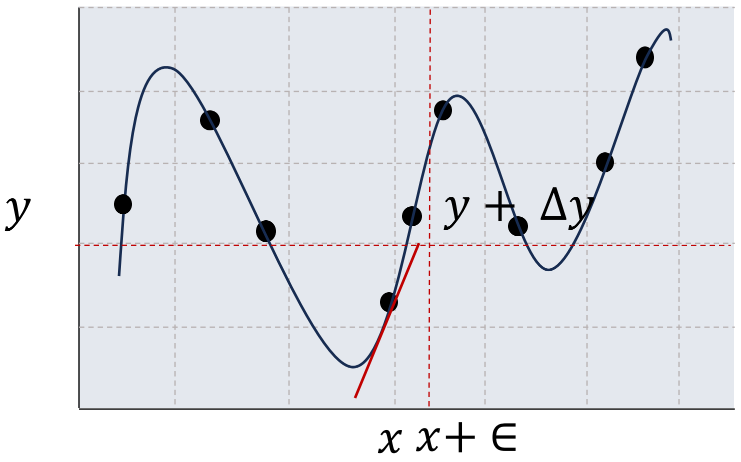
- Assume there is a function $y = f(x)$

- Derivative of y = $f'(x)$ = $\frac{dy}{dx}$
  - $f'(x)$ gives the slope of $f(x)$ at point x



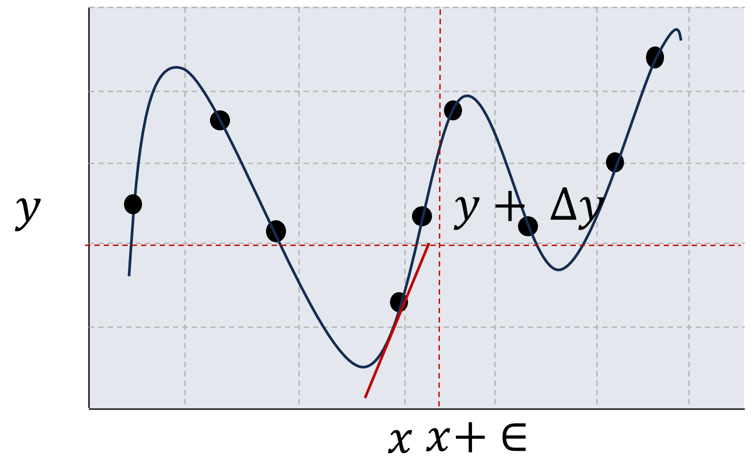
Gradient Descent

- Assume there is a function $y = f(x)$
- Derivative of y , $f'(x) = \frac{dy}{dx}$
  - $f'(x)$ gives the slope of $f(x)$ at point x
- $f'(x + \epsilon) \approx f(x) + \epsilon f'(x)$
  - This provides information on how do we change x such that we can influence small improvement in y



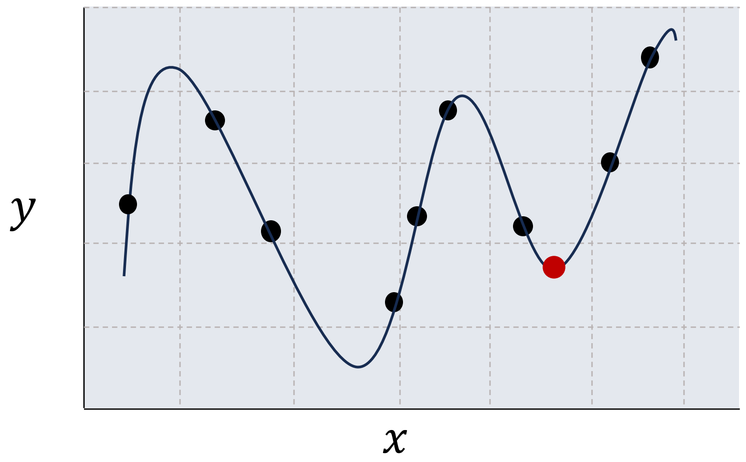
Gradient Descent

- Assume there is a function $y = f(x)$
- Derivative of y , $f'(x) = \frac{dy}{dx}$
  - $f'(x)$ gives the slope of $f(x)$ at point x
- $f'(x + \epsilon) \approx f(x) + \epsilon f'(x)$
  - This provides information on how do we change x such that we can influence small improvement in y
- $f'(x)$=0
  - This is the **stationary point**, because now changing x does not give information on change in y .



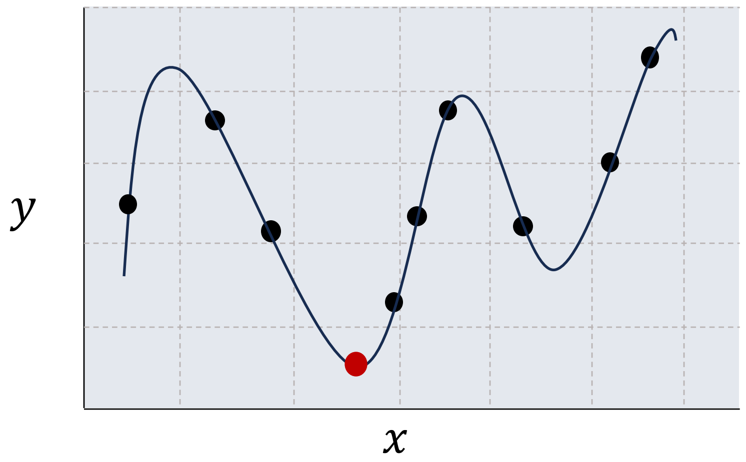
Gradient Descent

- Assume there is a function $y = f(x)$
- Derivative of y , $f'(x) = \frac{dy}{dx}$
  - $f'(x)$ gives the slope of $f(x)$ at point x
- $f'(x+\epsilon) \approx f(x) + \epsilon f'(x)$
  - This provides information on how do we change x such that we can influence small improvement in y
- $f'(x)=0$
  - This is the **stationary point**, because now changing x does not give information on change in y .
- **Local minimum** = $f(x)$ is lower than all neighboring points



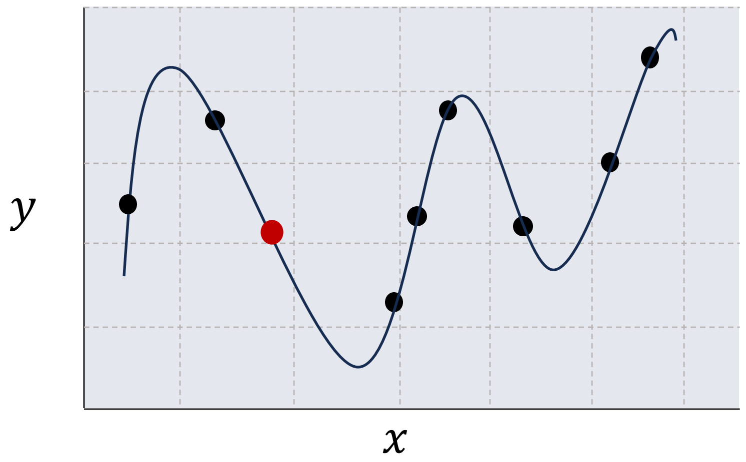
Gradient Descent

- Assume there is a function $y = f(x)$
- Derivative of y , $f'(x) = \frac{dy}{dx}$
  - $f'(x)$ gives the slope of $f(x)$ at point x
- $f'(x + \epsilon) \approx f(x) + \epsilon f'(x)$
  - This provides information on how do we change x such that we can influence small improvement in y
- $f'(x)$=0
  - This is the **stationary point**, because now changing x does not give information on change in y .
- **Local minimum** = $f(x)$ is lower than all neighboring points
- **Global minimum** = $f(x)$ is lower than all neighboring points



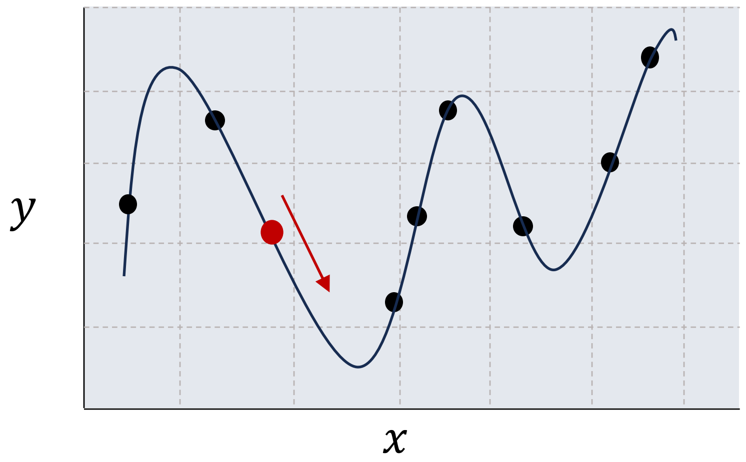
Gradient Descent

- Assume there is a function $y = f(x)$
- Derivative of y , $f'(x) = \frac{dy}{dx}$
  - $f'(x)$ gives the slope of $f(x)$ at point x
- $f'(x + \epsilon) \approx f(x) + \epsilon f'(x)$
  - This provides information on how do we change x such that we can influence small improvement in y
- $f'(x)$=0
  - This is the **stationary point**, because now changing x does not give information on change in y .
- **Local minimum** = $f(x)$ is lower than all neighboring points
- **Global minimum** = $f(x)$ is lower than all neighboring points
- **Saddle points** = points which are neither minimum, nor maximum



Gradient Descent

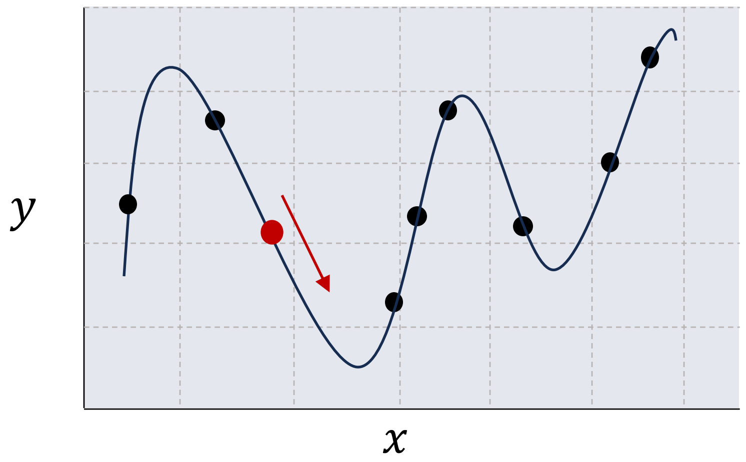
- Assume there is a function $y = f(x)$
- Derivative of y , $f'(x) = \frac{dy}{dx}$
  - $f'(x)$ gives the slope of $f(x)$ at point x
- $f'(x + \epsilon) \approx f(x) + \epsilon f'(x)$
  - This provides information on how do we change x such that we can influence small improvement in y
- $f'(x)$=0
  - This is the **stationary point**, because now changing x does not give information on change in y .
- **Local minimum** = $f(x)$ is lower than all neighboring points
- **Global minimum** = $f(x)$ is lower than all neighboring points
- **Saddle points** = points which are neither minimum, nor maximum
- In gradient decent we continuously check for local minimum and head in the direction of negative gradient.



Gradient Descent

To find the minimum value of Loss Function K

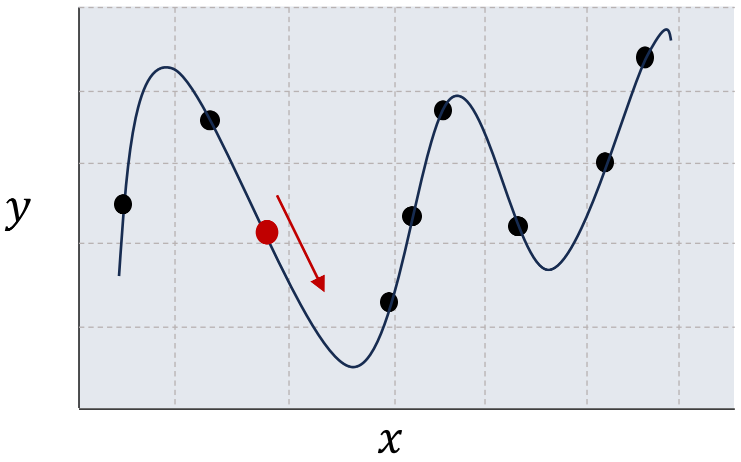
- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill



Gradient Descent

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent $= \nabla_{\beta}\ K$
  - Step 2c: Compute new value of β_{new} = β - $\alpha\nabla_{\beta}K$
  - Go to 2b and Repeat

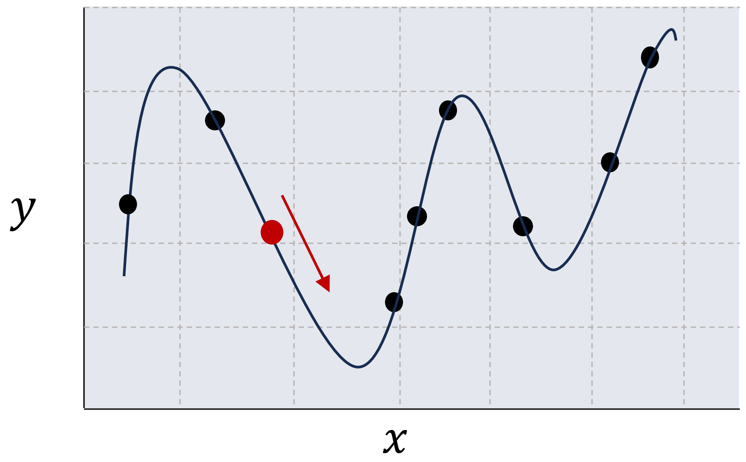


In gradient descent , the goal is to decrease K by moving in the direction of the negative gradient.

Gradient Descent

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent $= \nabla_\beta \ K$
  - Step 2c: Compute new value of $\beta_{new} = \beta - \alpha \nabla_\beta K$
  - Go to 2b and Repeat



What if there are too many variables?

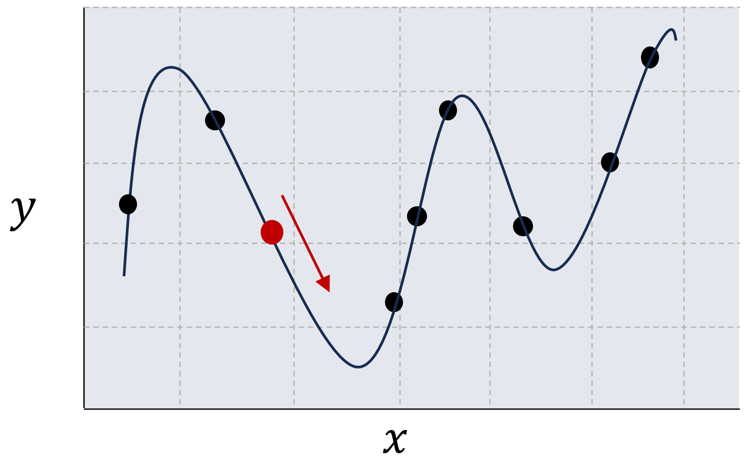
Gradient Descent

$$y' = \mathrm{f(x)} = \beta_0 + \beta_1 * \mathrm{x}_1 + \beta_2 * \mathrm{x}_2 + \cdots + \beta_p * \mathrm{x}_p$$

Computing Partial Derivative w.r.t. β

$$\nabla_\beta \mathrm{f(x)} = \begin{vmatrix} \frac{\partial \mathrm{f(x)}}{\partial \beta_1} \\ \frac{\partial \mathrm{f(x)}}{\partial \beta_2} \\ \vdots \\ \frac{\partial \mathrm{f(x)}}{\partial \beta_p} \end{vmatrix}$$

$\frac{\partial \mathrm{f(x)}}{\partial \beta_1} =$ partial derivate of $f(x)$ w.r.t. β_1 while all other β_i is constant.



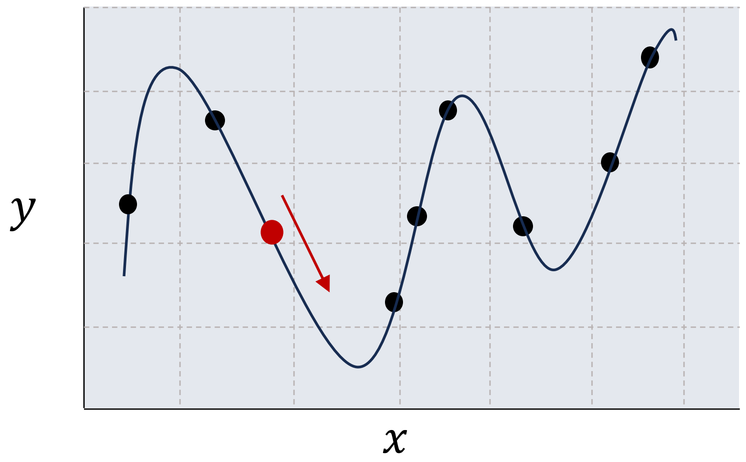
In gradient descent , the goal is to decrease f by moving in the direction of the negative gradient.

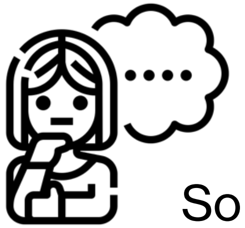
Gradient Descent

$$y' = f(x) = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_p * x_p$$

Computing Partial Derivative w.r.t. β

$$\nabla_\beta f(x) = \begin{vmatrix} \frac{\partial f(x)}{\partial \beta_1} \\ \frac{\partial f(x)}{\partial \beta_2} \\ \vdots \\ \frac{\partial f(x)}{\partial \beta_p} \end{vmatrix}$$





So, what does the algorithm look like?

In gradient descent , the goal is to decrease f by moving in the direction of the negative gradient.

Gradient Descent

Loss Function (K)

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent $= \nabla_{\beta}\ K$ (partial derivative of K)
  - Step 2c: Compute new value of $\beta_{new} = \beta - \alpha\nabla_{\beta}K$
  - Go to 2b and Repeat

Gradient Descent

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i')^2$

For linear regression task

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
    - Step 2a: Start with any arbitrary value of β
    - Step 2b: Find the steepest descent $= \nabla_\beta K$ (partial derivative of K)
    - Step 2c: Compute new value of $\beta_{new} = \beta - \alpha\nabla_\beta K$
    - Go to 2b and Repeat

McMaster University

Gradient Descent

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i{'})^2 \Rightarrow \frac{1}{n}\sum_{n=1}^{n}(y_i - \beta . x_i)^2$ Why? $Y = \beta . X$

For linear regression task

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent $= \nabla_\beta\ K$ (partial derivative of K)
  - Step 2c: Compute new value of $\beta_{new} = \beta - \alpha\nabla_\beta K$
  - Go to 2b and Repeat

Gradient Descent

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i')^2 \Rightarrow \frac{1}{n}\sum_{n=1}^{n}(y_i - \beta.x_i)^2$

For linear regression task

$$\nabla_\beta K = \nabla_\beta\left(\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta.x_i)^2\right) \Rightarrow \cdots \Rightarrow \frac{2}{n}(\beta.x - y)\cdot \boldsymbol{x^T}$$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent = $\nabla_\beta\ K$ (partial derivative of K)
  - Step 2c: Compute new value of β_{new} = β - α$\nabla_\beta K$
  - Go to 2b and Repeat

Gradient Descent

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i')^2 \Rightarrow \frac{1}{n}\sum_{n=1}^{n}(y_i - \beta.x_i)^2$

For linear regression task

$$\nabla_\beta K = \nabla_\beta\left(\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta.x_i)^2\right) \Rightarrow \cdots \Rightarrow \frac{2}{n}(\beta.x - y)\cdot \boldsymbol{x^T}$$

Predicted value

Actual value

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
    - Step 2a: Start with any arbitrary value of β
    - Step 2b: Find the steepest descent $= \nabla_\beta\ K$ (partial derivative of K)
    - Step 2c: Compute new value of β_{new} = β - α$\nabla_\beta K$
    - Go to 2b and Repeat

Gradient Descent

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i')^2 \Rightarrow \frac{1}{n}\sum_{n=1}^{n}(y_i - \beta . x_i)^2$

For linear regression task

$$\nabla_\beta K = \nabla_\beta\left(\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta . x_i)^2\right) \Rightarrow \cdots \Rightarrow \frac{2}{n}(\beta . x - y) \cdot \boldsymbol{x^T}$$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
    - Step 2a: Start with any arbitrary value of β
    - Step 2b: Find the steepest descent $= \nabla_\beta\ K$ (partial derivative of K)
    - Step 2c: Compute new value of β_{new} = β - α$\nabla_\beta K$
    - Go to 2b and Repeat

Gradient Descent

Loss Function (K) $= \frac{1}{n}\sum_{n=1}^{n}(y_i - y_i')^2 \Rightarrow \frac{1}{n}\sum_{n=1}^{n}(y_i - \beta . x_i)^2$

For linear regression task

$$\nabla_\beta K = \nabla_\beta\left(\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta . x_i)^2\right) \Rightarrow \cdots \Rightarrow \frac{2}{n}(\beta . x - y) \cdot \boldsymbol{x^T}$$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent $= \nabla_\beta K$ (partial derivative of K)
  - Step 2c: Compute new value of β_{new} = β - α$\nabla_\beta K$
  - Go to 2b and Repeat

Steepest descent converges when every element of the gradient is zero $\nabla_\beta K$ =0

Gradient Descent

For linear regression task $\nabla_{\beta} K = \frac{2}{n}(\beta . x - y) \cdot \boldsymbol{x}^{\boldsymbol{T}}$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent = $\nabla_{\beta}\ K$
  - Step 2c: Compute new value of β_{new} = β - $\alpha\nabla_{\beta}K$
  - Go to 2b and Repeat

```
beta = np.random.randn(2,1)  # random initialization
```

```
gradients = 2/n * (X.T).dot(X_.dot(beta) - Y)
```

```
beta = beta - alpha * gradients
```

McMaster University

Gradient Descent

For linear regression task $\nabla_\beta K = \frac{2}{n}(\beta . x - y) \cdot \boldsymbol{x}^{\boldsymbol{T}}$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent = $\nabla_\beta\ K$
  - Step 2c: Compute new value of β_{new} = β - $\alpha\nabla_\beta K$
  - Go to 2b and Repeat

α = learning rate

```
beta = np.random.randn(2,1)  # random initialization
```

```
gradients = 2/n * (X.T).dot(X_.dot(beta) - Y)
```

```
beta = beta - alpha * gradients
```

Design Considerations

For linear regression task $\nabla_\beta K = \frac{2}{n}(\beta . x - y) \cdot \boldsymbol{x^T}$

$What\ should\ be\ the\ learning\ rate?$

$What\ if\ it's\ learning\ rate\ is\ high/low?$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent $= \nabla_\beta\ K$
  - Step 2c: Compute new value of β_{new} = β - $\alpha \nabla_\beta K$
  - Go to 2b and Repeat

Design Considerations

For linear regression task $\nabla_\beta K = \frac{2}{n}(\beta . x - y) \cdot \boldsymbol{x^T}$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
  - Step 2a: Start with any arbitrary value of β
  - Step 2b: Find the steepest descent = $\nabla_\beta\ K$
  - Step 2c: Compute new value of β_{new} = β - α$\nabla_\beta K$
  - Go to 2b and Repeat → $How\ long\ should\ we\ run\ this?$

Design Considerations

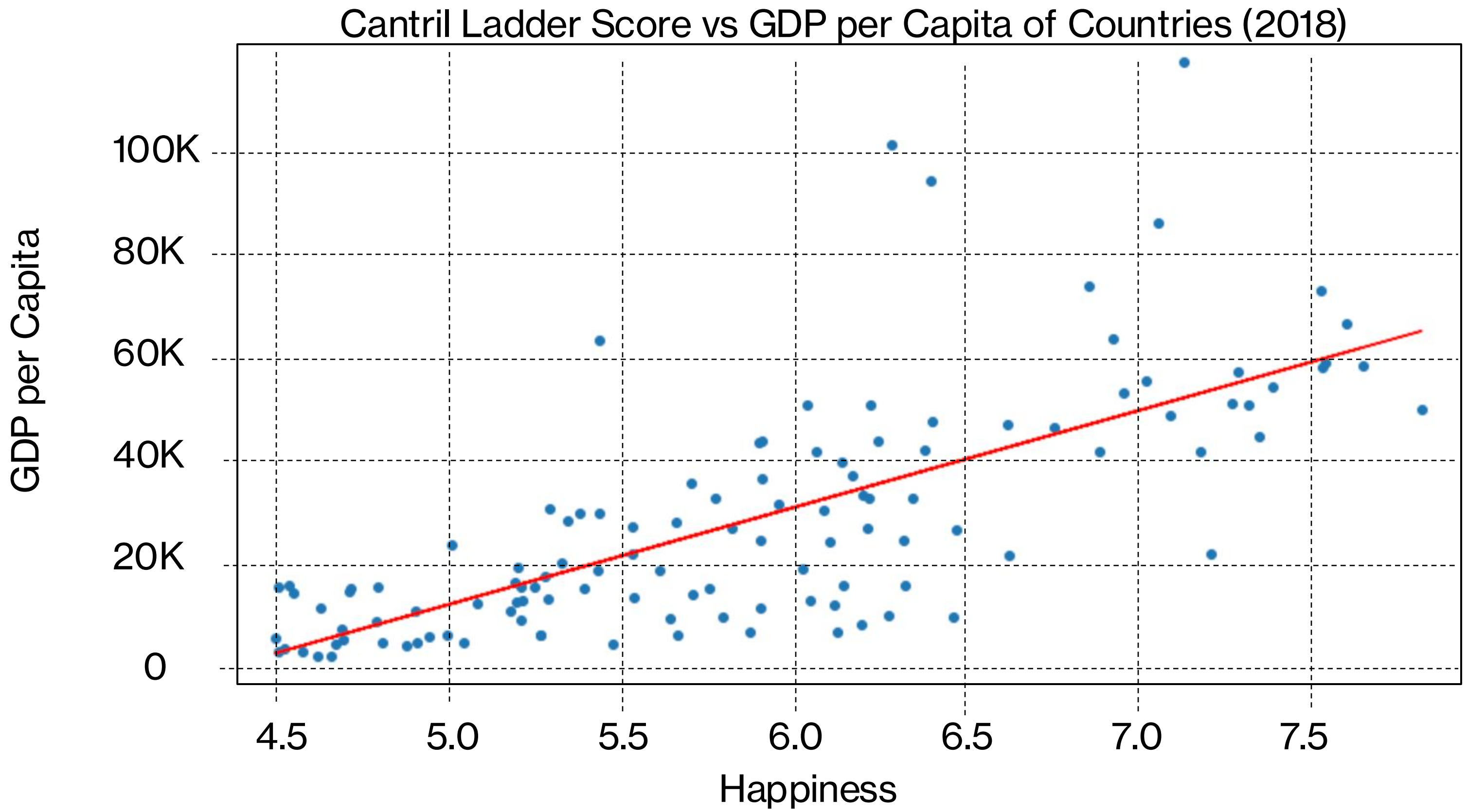
For linear regression task $\nabla_\beta K = \frac{2}{n}(\beta . x - y) \cdot \boldsymbol{x^T}$

To find the value β that minimizes a Loss Function K

- Step 1: Obtain the gradient of K
- Step 2: Follow this gradient downhill:
    - Step 2a: Start with any arbitrary value of β
    - Step 2b: Find the steepest descent $= \nabla_\beta \; K$
    - Step 2c: Compute new value of β_{new} = β - α$\nabla_\beta K$
    - Go to 2b and Repeat → $How\ long\ should\ we\ run\ this?$

$What\ if\ it's\ too\ high\ or\ too\ low?$

Linear Regression



Thank You

McMaster University