Machine Learning Flow (Any algorithm)

*refers to Linear_Regression_v2.ipynb

A — Find Data → B — Load Data → C — Data preprocessing → D — Split data → E — Train model on data → F — Evaluate model

B:
1.1 Load data

C:
C1
1.2 Explore data
1.3 Selecting variable

C2 — Exploratory Data Analysis

C3 — Feature Engineering → missing values, Encoding

One HOT ENCODING

C4 — Feature Scaling

D:
- Train data
- Testing data

1.3 Prepare data

E:
- Algorithm

1.4 Train

math's here

F:
-

1.5 Evaluate

C3. Feature Engineering

One Hot Encoding

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.90	0	yes	southwest	16884.9240
11	62	female	26.29	0	yes	southeast	27808.7251
14	27	male	42.13	0	yes	southeast	39611.7577
19	30	male	35.30	0	yes	southwest	36837.4670
23	34	female	31.92	1	yes	northeast	37701.8768

Sex Encoding

- 0 : Female
- 1 : Male

Children Encoding

0	1	2	3	4	5	Number of child
0	0	0	0	0	0	0
0	1	0	0	0	0	1
0	0	1	0	0	0	2
0	0	0	1	0	0	3
0	0	0	0	1	0	4
0	0	0	0	0	1	5

age	sex	bmi	OHE_1	OHE_2	OHE_3	OHE_4	OHE_5
19	0	27.90	0	0	0	0	0
34	0	31.92	1	0	0	0	0

D) Split the data

1. First shuffle the rows of the date

	BMI	Charges
1	27	10,000
	⋮	
9337		
9338		
1338	30	50,000

Train data

2. X_train is 0.7 × 1338 = 9337 rows

Y_train is .. = 9337 rows

Test data

3. X_test is 0.3 × 1338 = 401 rows

y_test is 0.3 × = 401 rows

Smoker only has 274 rows

X_train = 0.7 × 274 = 191

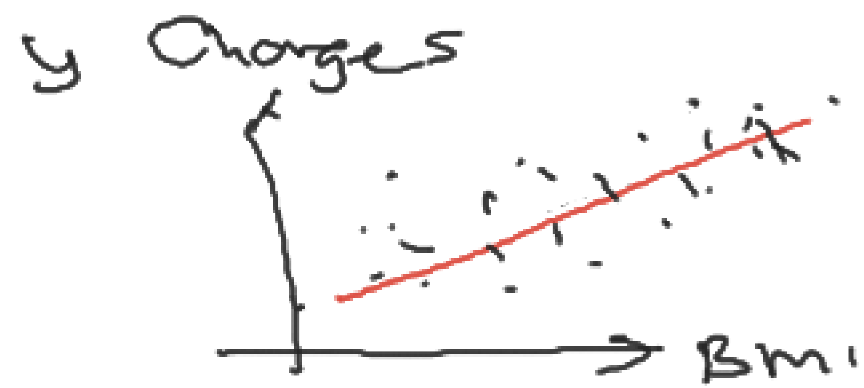
X_test = 0.3 × 274 = 82

Linear Regression (LR)

↳ 1) One variable LR

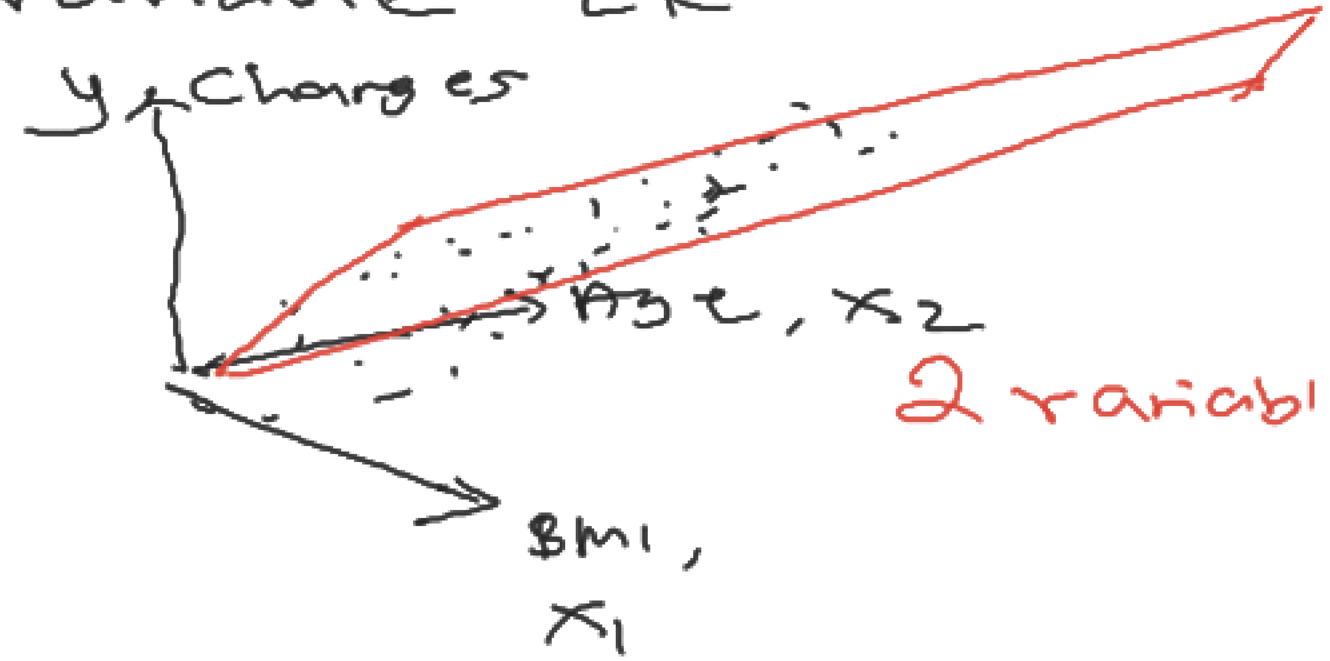
2) multi variable LR

1- variable



$\boxed{y = mx + c}$

$h_\theta(x) = \boxed{\theta_0} + \boxed{\theta_1} x_1$



2 variable

$\boxed{y = m_1 x_1 + m_2 x_2 + c}$

$h_\theta(x) = \boxed{\theta_0} + \boxed{\theta_1} x_1 + \boxed{\theta_2} x_2$

Q How about 11 variables? What does the eqn looks like?

A) $h_\theta(x) = \boxed{\theta_0} + \overset{m_1}{\boxed{\theta_1}} x_1 + \overset{m_2}{\boxed{\theta_2}} x_2 + \overset{m_3}{\boxed{\theta_3}} x_3 + \dots + \overset{m_{11}}{\boxed{\theta_{11}}} x$

Linear Regression (LR)

- We are given this insurance.csv dataset.
- We want to do some LR on this dataset.
- For example. Can we predict how much a person has to pay for his/her insurance based on their BMI?

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

How to look at dataset

(input) Features, variable, dimension, attribute

target output,

rows = observations / samples

ith observation

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

Continuous variable

1. age
2. Bmi
3. Charges

Categorical variable

1. Sex
2. Children
3. Smoker
4. region

Revision of Linear Algebra

x , $x = 5$: scalar

$$\mathbf{x} , \mathbf{x} = \begin{array}{c} \overbrace{\quad\downarrow \quad \downarrow \quad \downarrow\quad}^{3\ col} \\ [1 \quad 2 \quad 3] \\ \xrightarrow{\hspace{2em}} \ 1\ row \end{array} \rightarrow \mathbf{x} \text{ is a } \overset{row \times col}{1 \times 3} \text{ row vector}$$

$$\mathbf{x} : \begin{array}{c} \\ \text{3 rows} \end{array} \left\{ \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \right. \text{ (column) vector}$$

1 col

$\mathbf{x}$ is a $\overset{row \times col}{3 \times 1}$ column vector

$$\mathbf{X} : \begin{bmatrix} 1 & 10 & 100 \\ 2 & 20 & 200 \\ 3 & 30 & 300 \end{bmatrix} \downarrow 3\ col$$

3 row

$\mathbf{X}$ is a 3×3 matrix

Combine

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad \mathbf{x}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix} \quad \mathbf{x}_3 = \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 10 & 100 \\ 2 & 20 & 200 \\ 3 & 30 & 300 \end{bmatrix}$$

Now when we see a dataset, lets imagine it as vectors and matrices

	x_1	x_2	x_3	x_4	x_5	x_6	y
	age	**sex**	**bmi**	**children**	**smoker**	**region**	**charges**
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

x_i : the i^{th} feature/attributes of the dataset. x_i is a col vector

y : a col vector of the target output

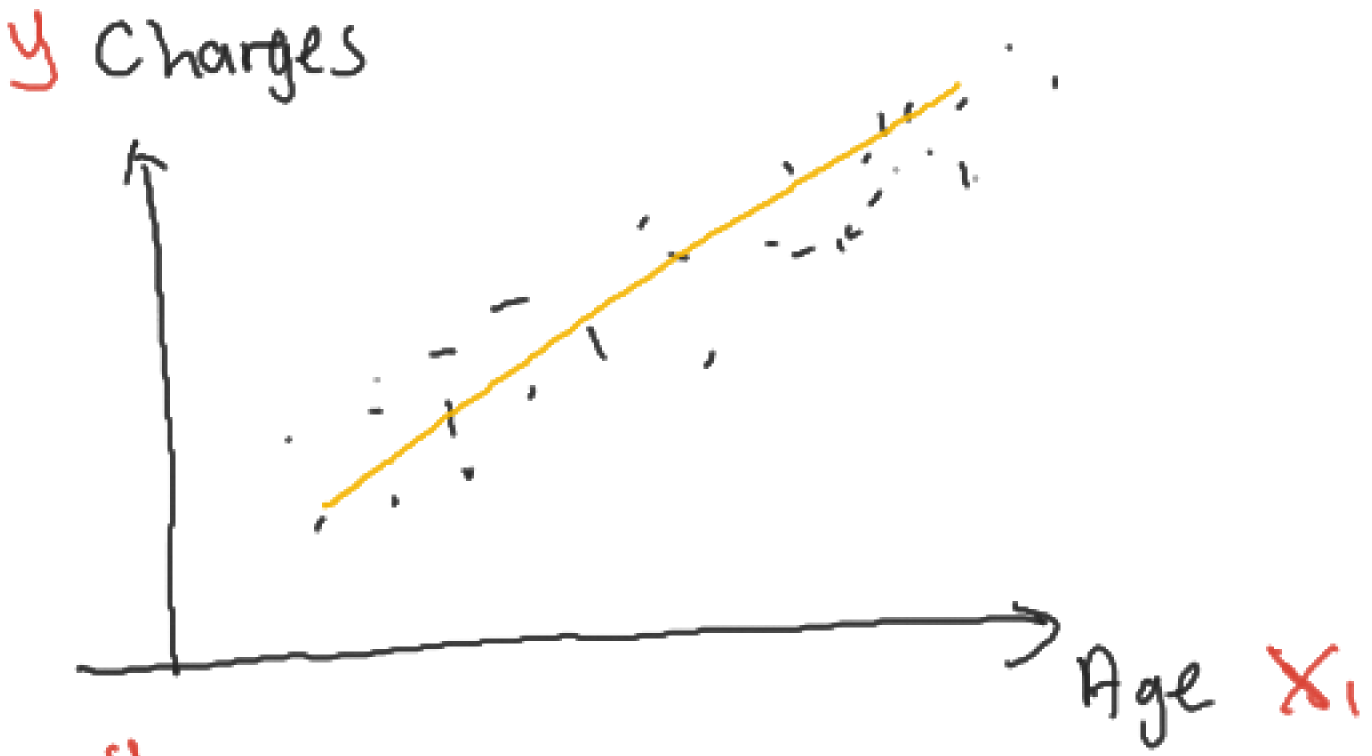
$$X = \begin{array}{c} 0 \\ \vdots \\ 1337 \end{array} \begin{bmatrix} x_1 & x_2 & \cdots & x_6 \\ 19 & \vdots & & \vdots \\ 18 & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \end{bmatrix}$$

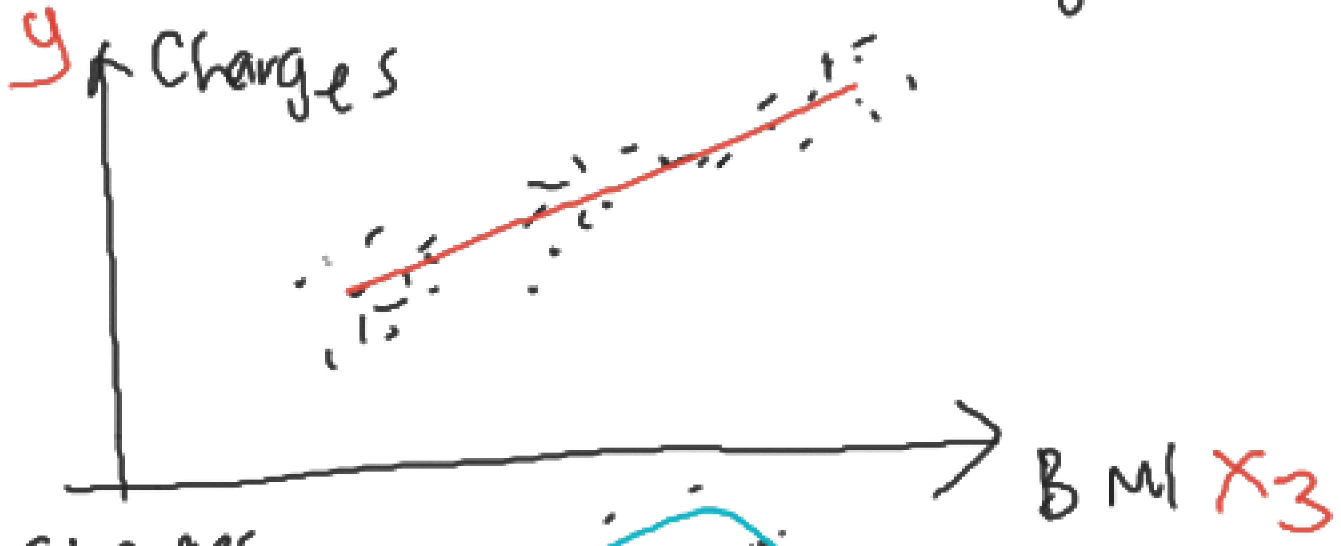
- we have 6 features. But how many rows of observation/samples do we have?

cols = 6
rows = 1338

X is a 1338×6 matrix #

Our 'intuition'





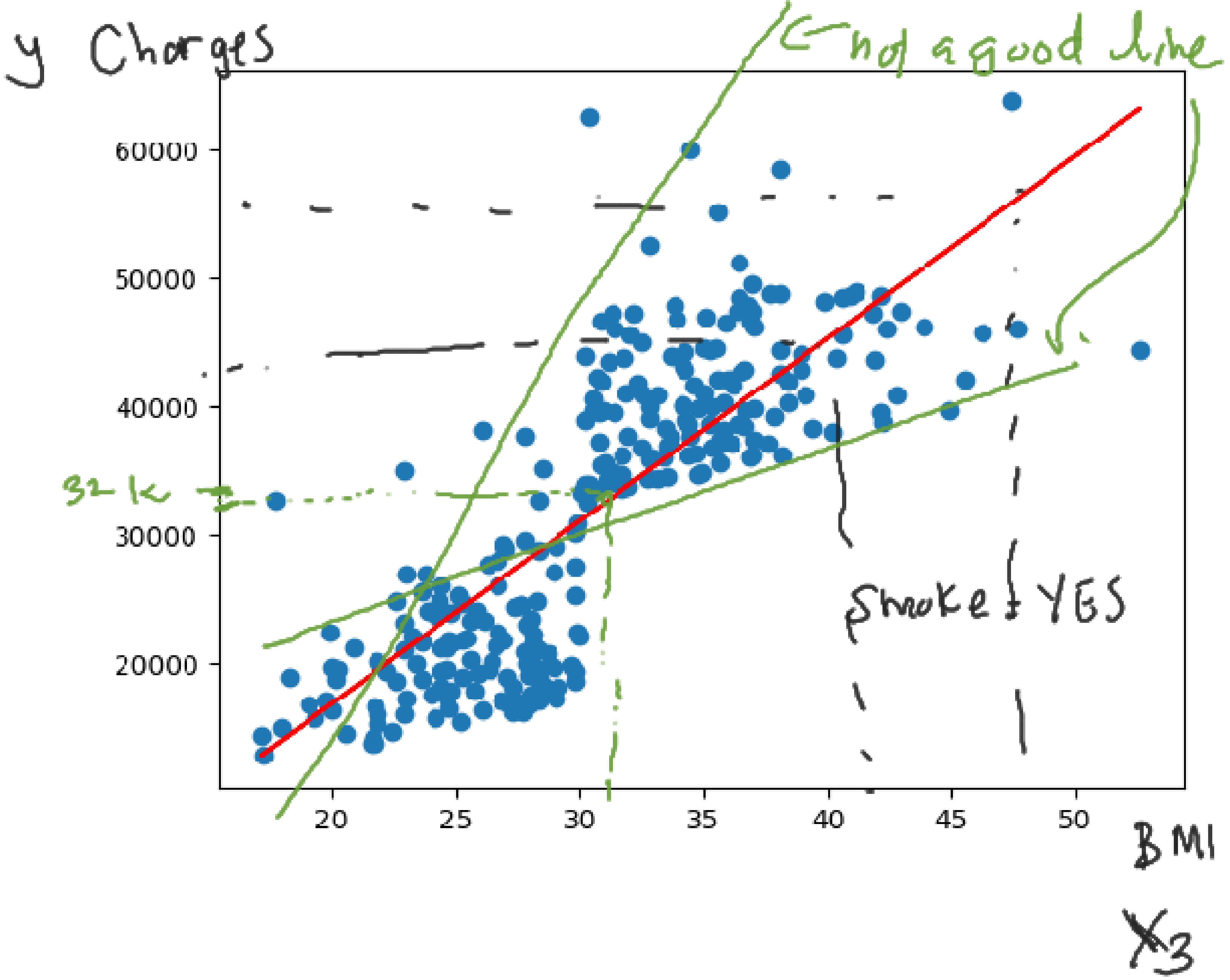
	X_1	X_2	X_3	X_4	X_5	X_6	y
	age	**sex**	**bmi**	**children**	**smoker**	**region**	**charges**
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

X: features/attributes, independent variable

y: target output, dependant variable

Linear Regression with one variable

	X_3 BMI	y Charges
1	29	50000
2	30	60000
⋮		
1338	25	70000
1337	31	?



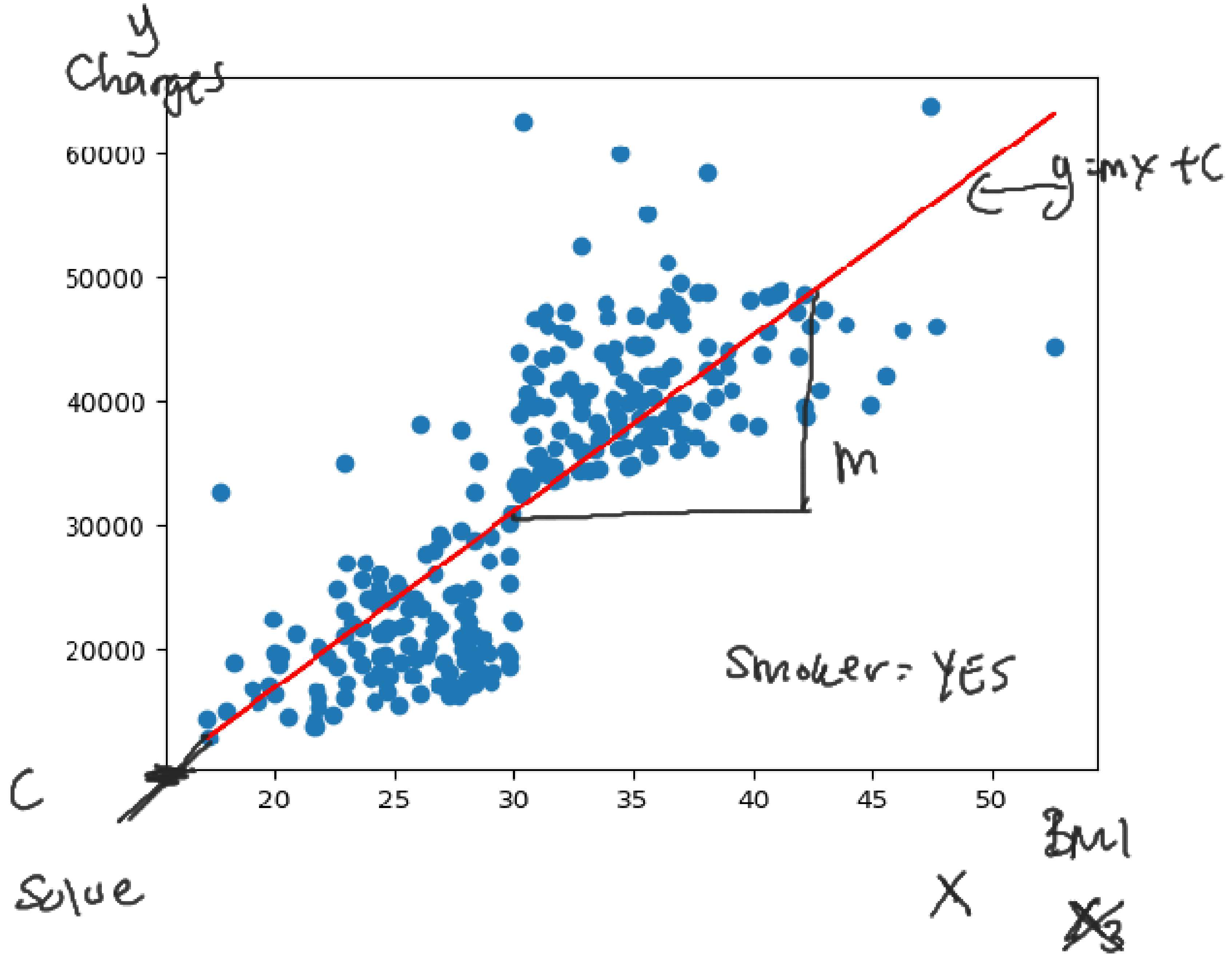
- Linear regression is to find a linear line that fits the dataset.
- We can make predictions for unseen observation. For example what is the insurance charges for a person with BMI = 31 ?

How do we find the best line (red) that fits the data?

$$y = mx + c$$

$$h(x) = \theta_0 + \theta_1 x$$

c: intercept m: slope



- So our job (for one variable) is to solve for m and c.

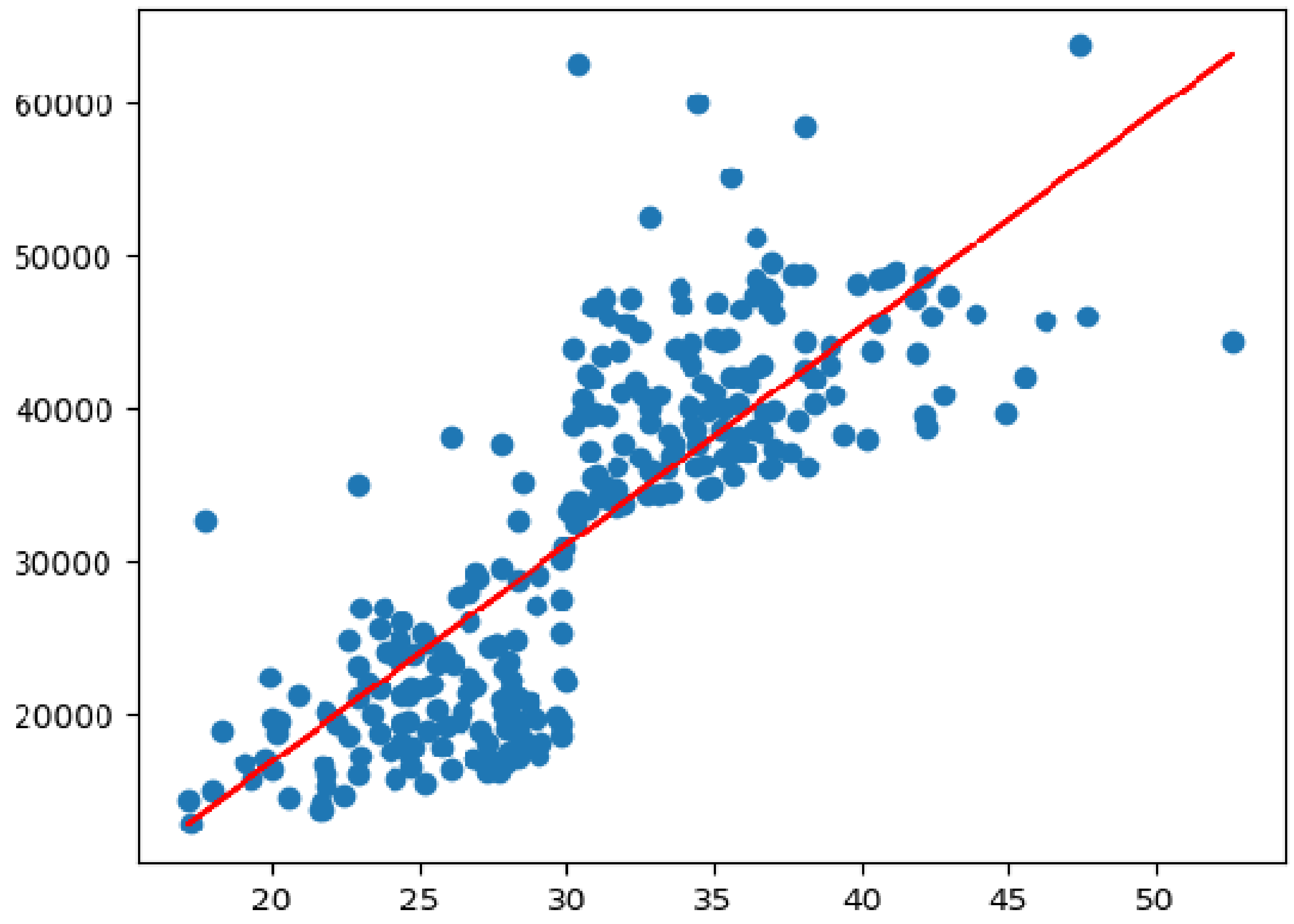
How do we find the 'best' m and c?

$$y = mx + c$$

Two methods

A) Solve using equation

B) Use gradient descent



A) Solve using equation

Ordinary Least Square method (from stats)

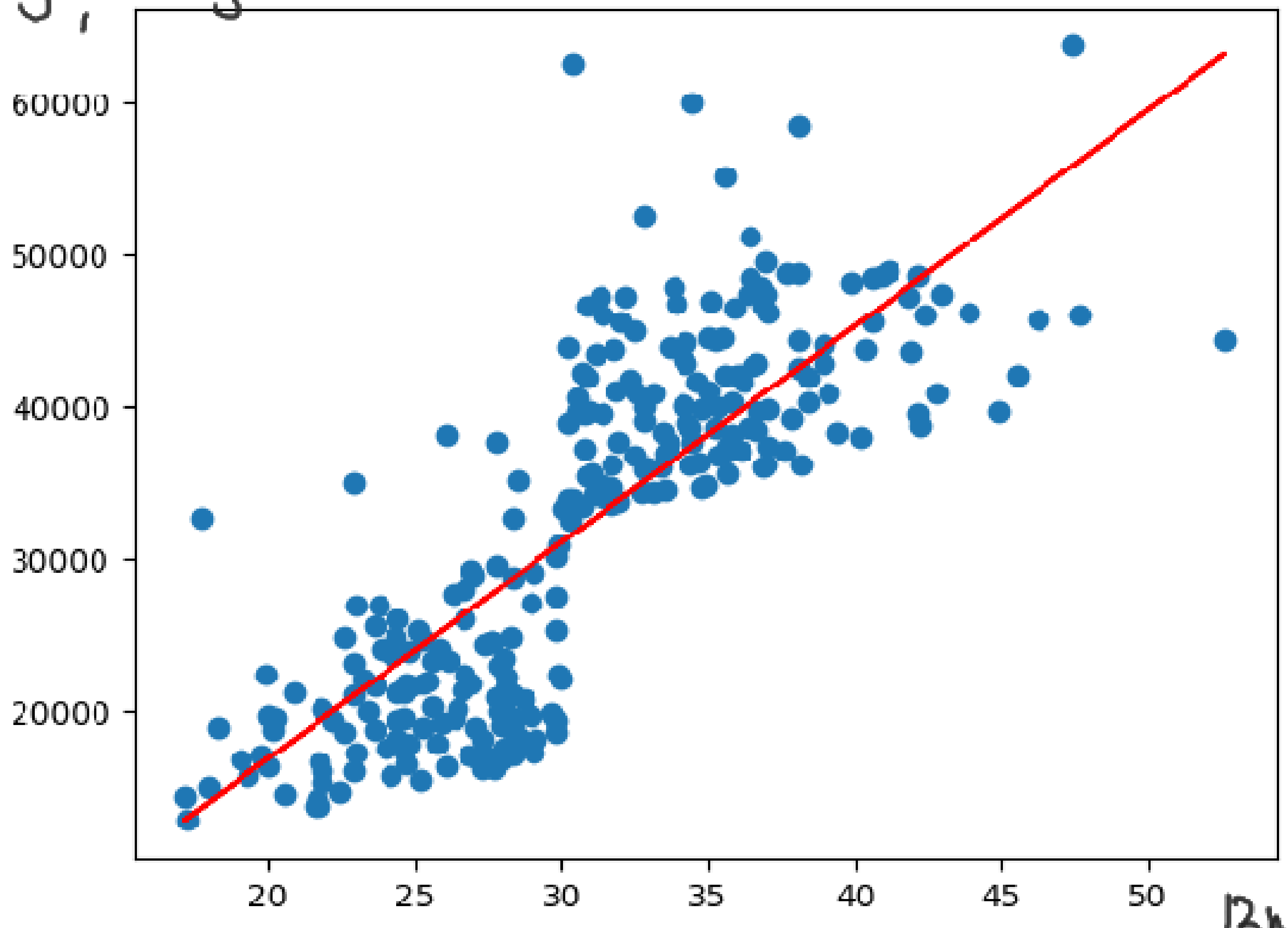
c, $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x};$

m, $\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2},$

what is this?

$y = mx + c$

$y = \beta_0 + \beta_1 x$



From our dataset

index	x_3 BMI	y Charges
1	19	
2	23	
i	$x_i = 25$	$y_i = 20000$

$n = 1338$

$\bar{x}$: averge of x $= \frac{1}{N}\sum_{i=1}^{N} x_i$

$\bar{y}$ = " of y $= \frac{1}{N}\sum_{i=1}^{N} y_i$

b) Use gradient Descent

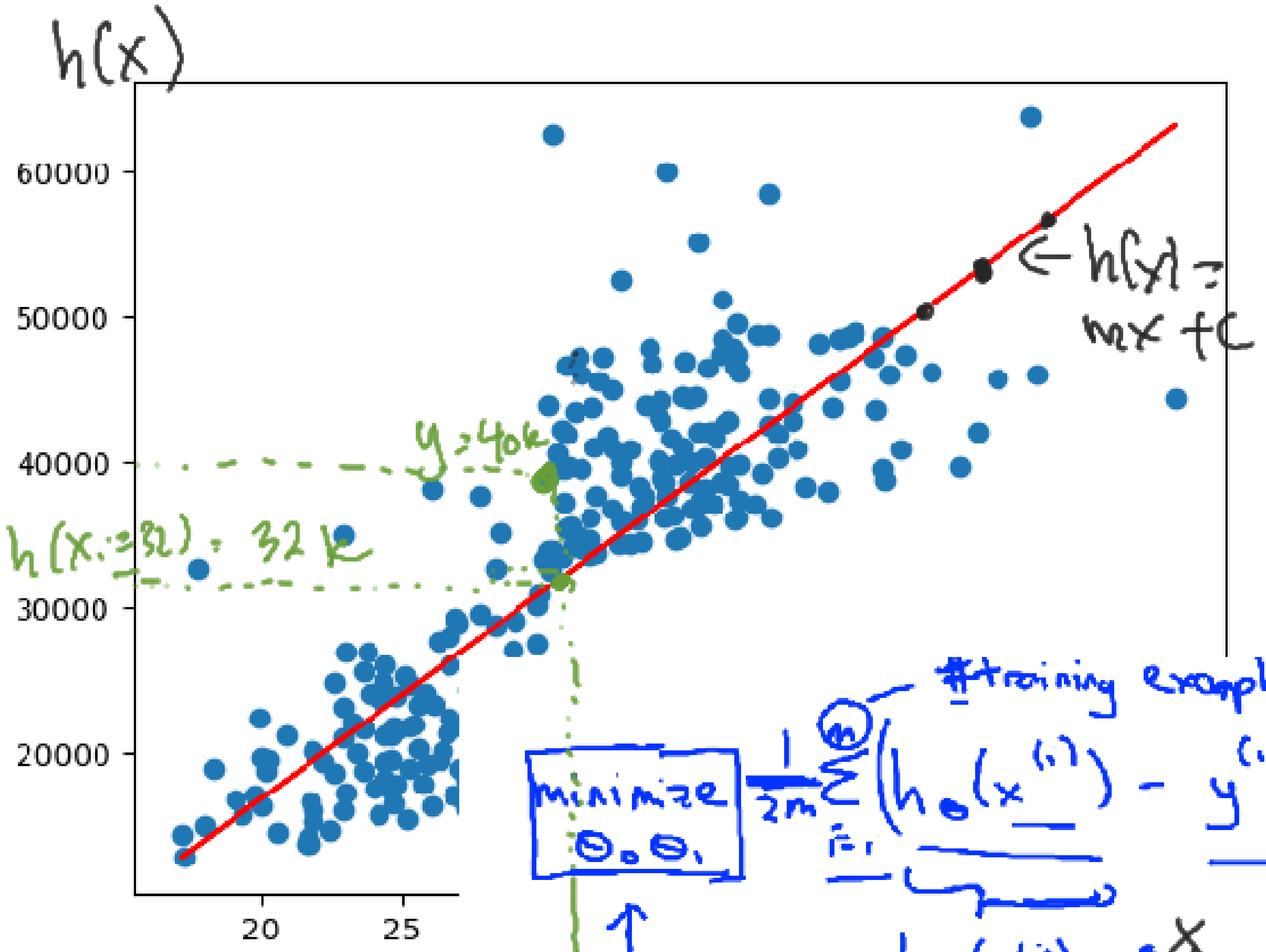
(gradient of the cost function)

We need to define a cost function

$h(x) = mx + c$ ← our red line

• : our data   • : our prediction (line)

$$\boxed{h(x^i) - y^i}^2$$



$$\underset{\theta_0, \theta_1}{\text{minimize}} \ \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

m: # training examples

$$h_\theta(x^{(i)}) = \theta_0 + \theta_1 x^{(i)}$$

∴ So our cost function is

$$\boxed{J(c,m) = \frac{1}{2N}\sum_{i=1}^{N}\left(h(x^i) - y^i\right)^2}$$

index	X BMI	y Charges	h(x)
1	27	40000	
2	31	45,000	
⋮			
i	x^i = 32	y_i = 40,000	32,000
⋮			
N = 1338	25	38 000	

(i))

at

or our

, y)

$$J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

$$\underset{\theta_0, \theta_1}{\text{minimize}} \ \underbrace{J(\theta_0, \theta_1)}_{\text{Cost function}}$$

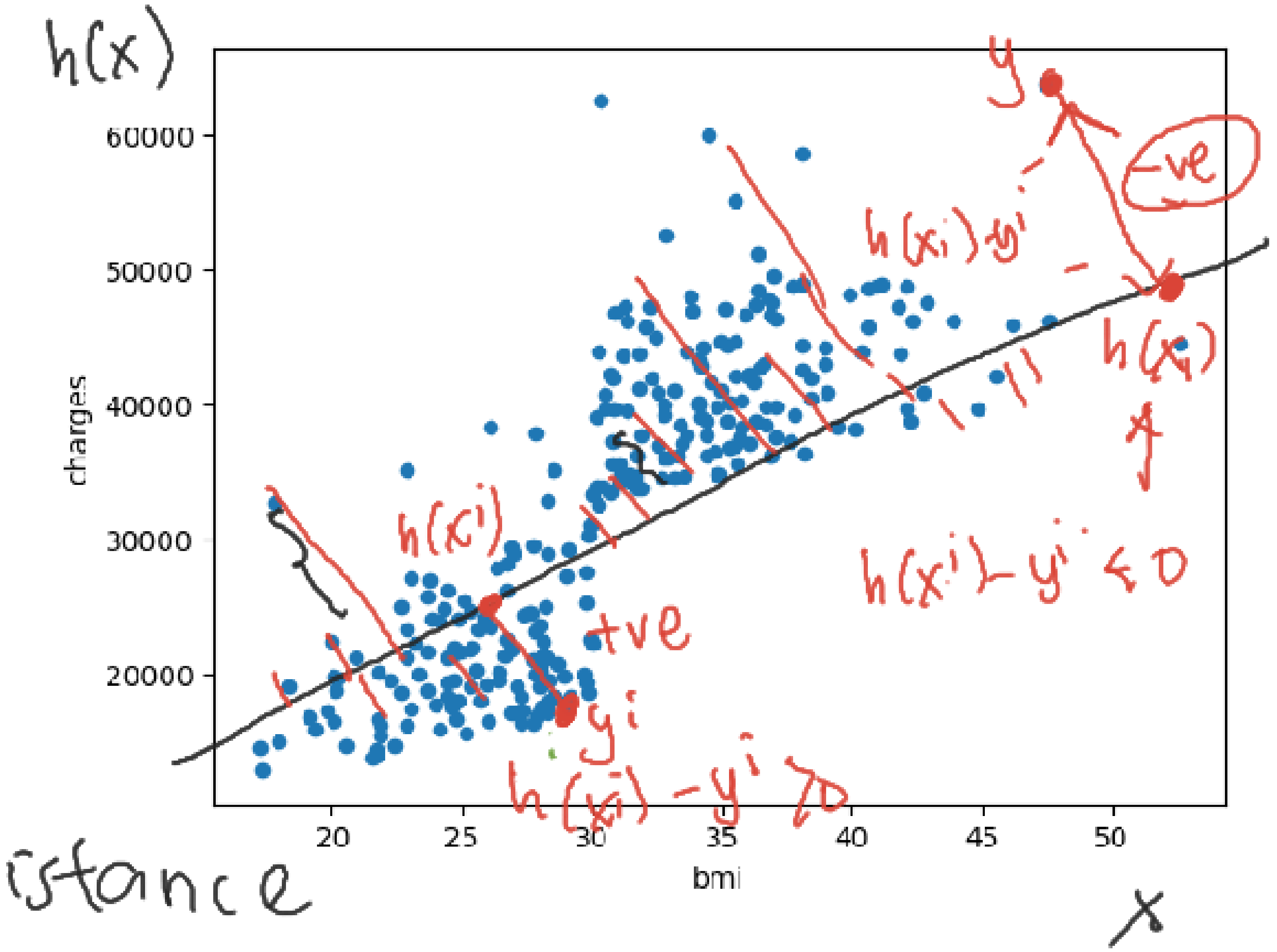
Squared error function

B.) Use gradient Descent

(gradient of the cost function)

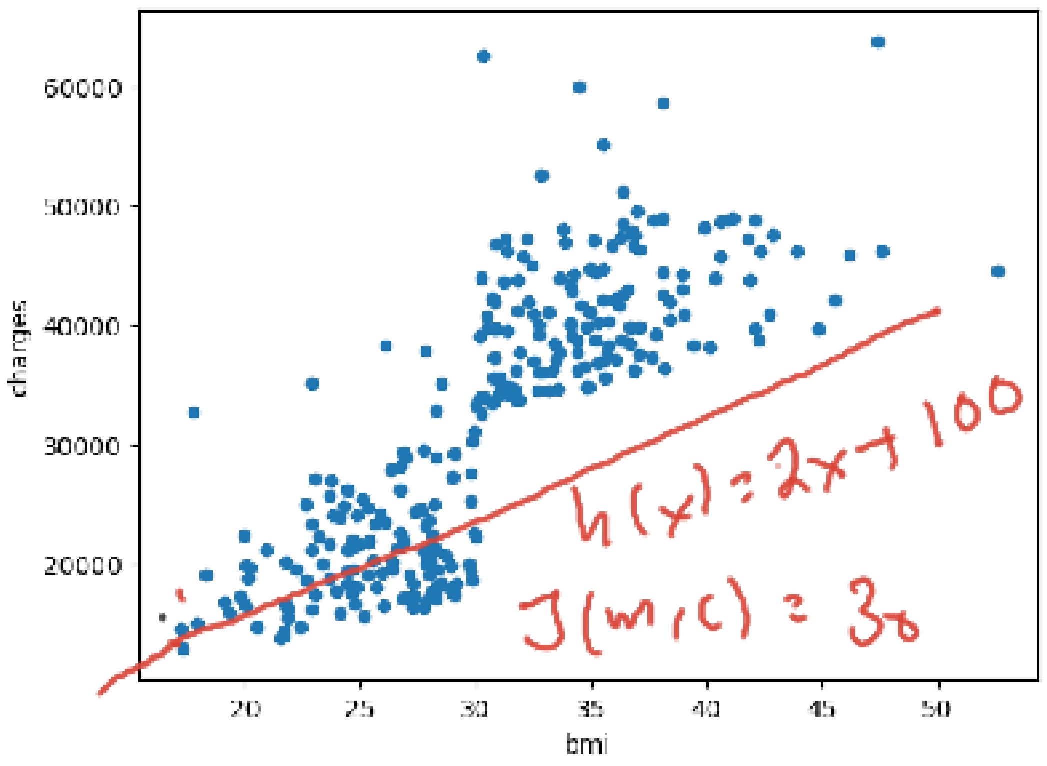
Cost function $h(x) = mx + c$

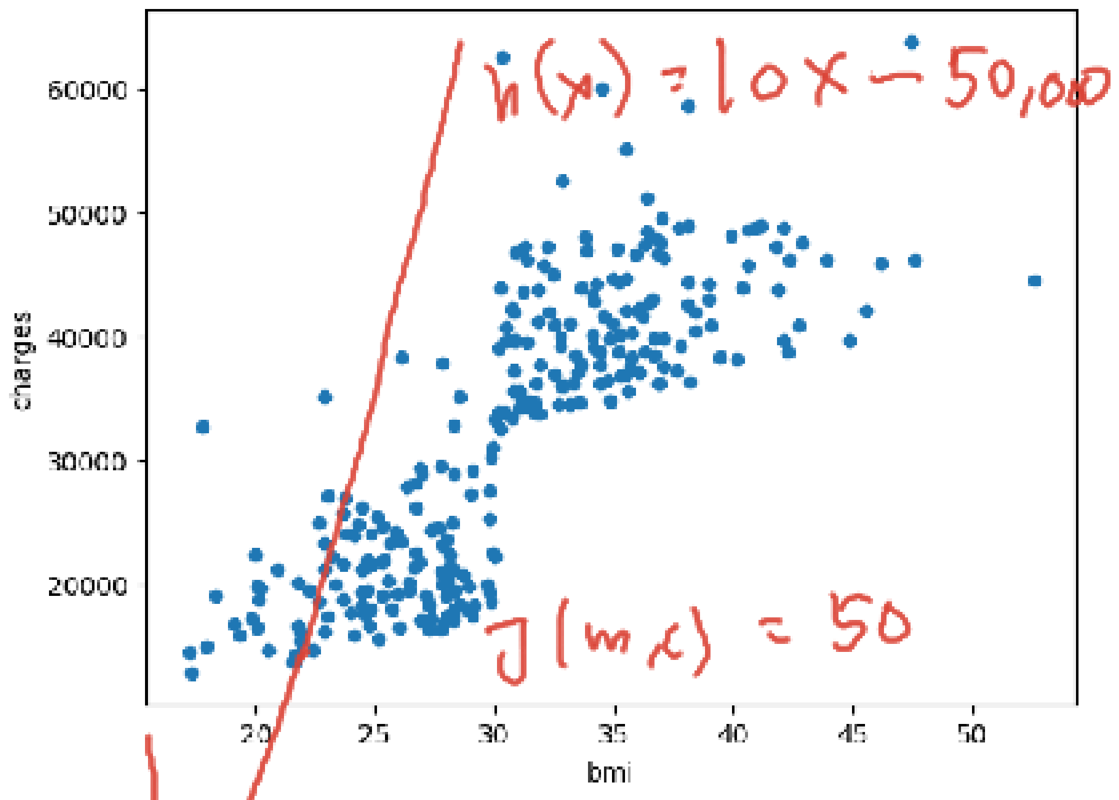
$$J(m,c) = \frac{1}{2N} \sum_{i=1}^{N=1338} \left(h(x^i) - y^i \right)^2$$

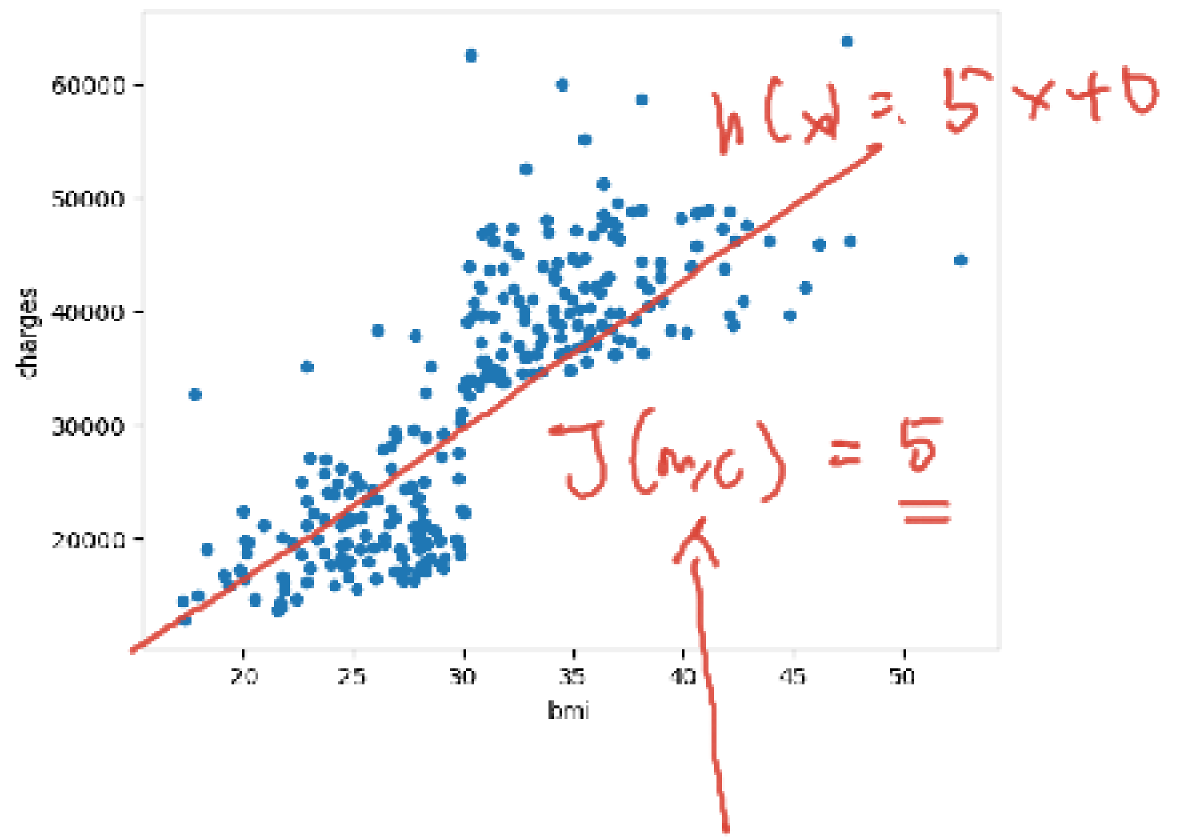


- The cost function calculates the distance between our prediction $h(x)$ and the original data.

- So then we need to find the values of m and c that minimizes the cost function J







$R^2 = 0.1$ bad

$R^2 = 0.01$ really bad

m=5, c=0
yields the lowest
error/cost function

$R^2 = 0.7$ good ✓

How do we find the best m and c that minimizes the cost function?

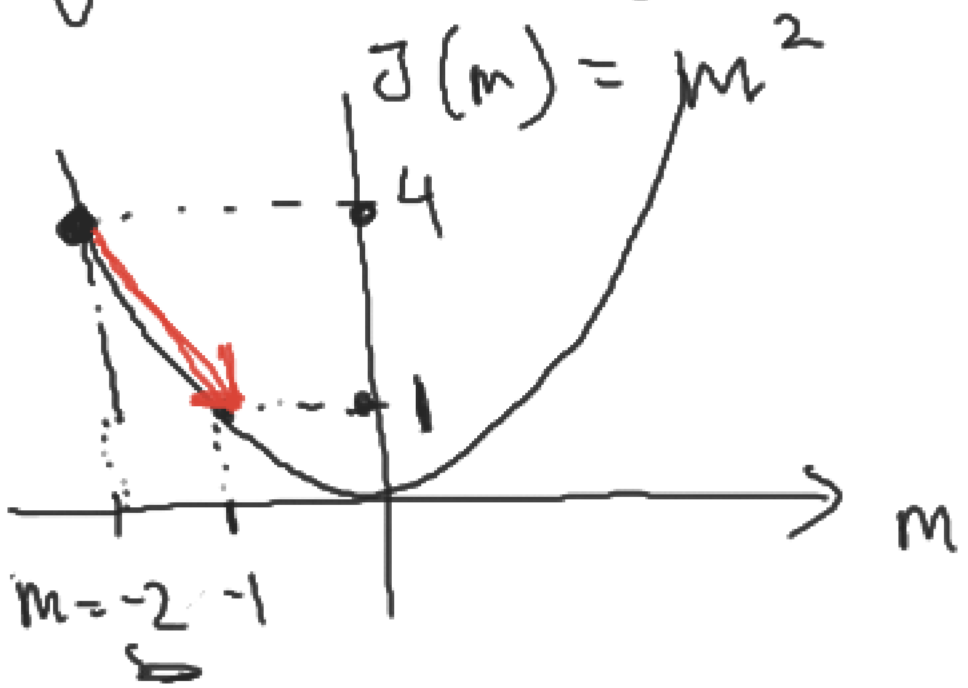
$$J(m,c) = \frac{1}{2N} \sum_{i=1}^{N=1338} (h(x^i) - y^i)^2$$

What does this cost function looks like?

(I don't know)

How do we find the best m and c that minimizes the cost function?

Let's just imagine our cost function looks like this



α: learning rate/step = 1/4

Update using gradient descent

1st iter

$$m_{new} = m_{old} - \alpha \frac{dJ(m)}{dm}$$

$$= -2 - \frac{1}{\cancel{4}} \cdot \cancel{2}(-\cancel{2})$$

$$= -1$$

$$Y_{new} = M_{new}^2 = 1$$

Derivative/gradient

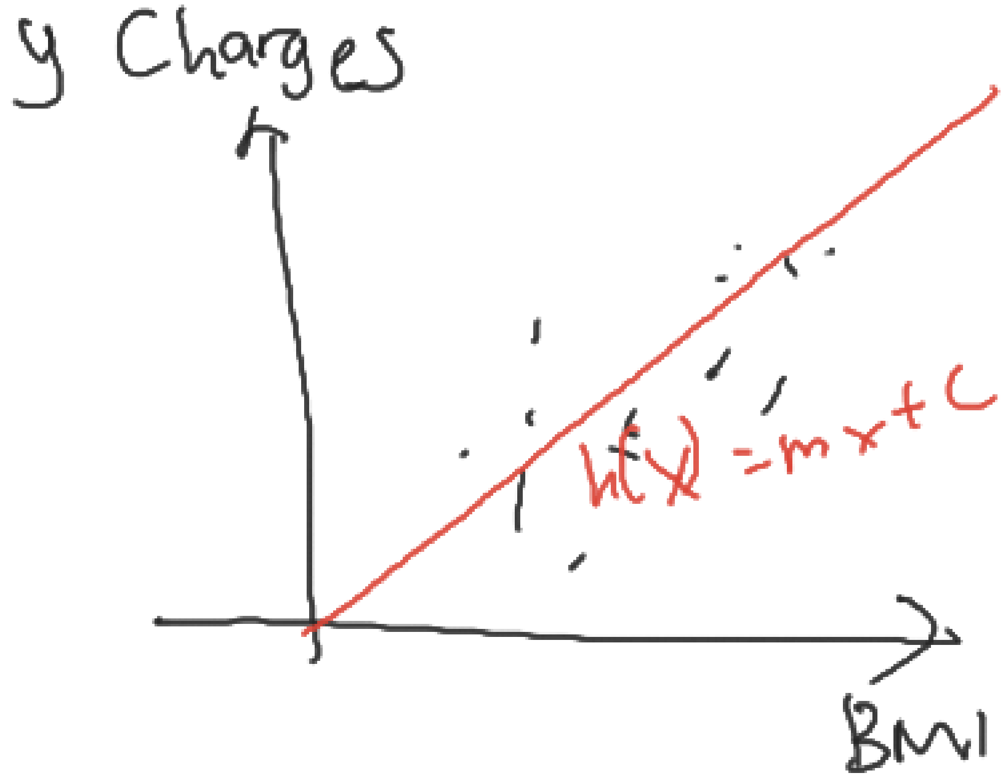
$$\boxed{J(m) = m^2 \qquad \frac{dJ(m)}{dm} = 2m}$$

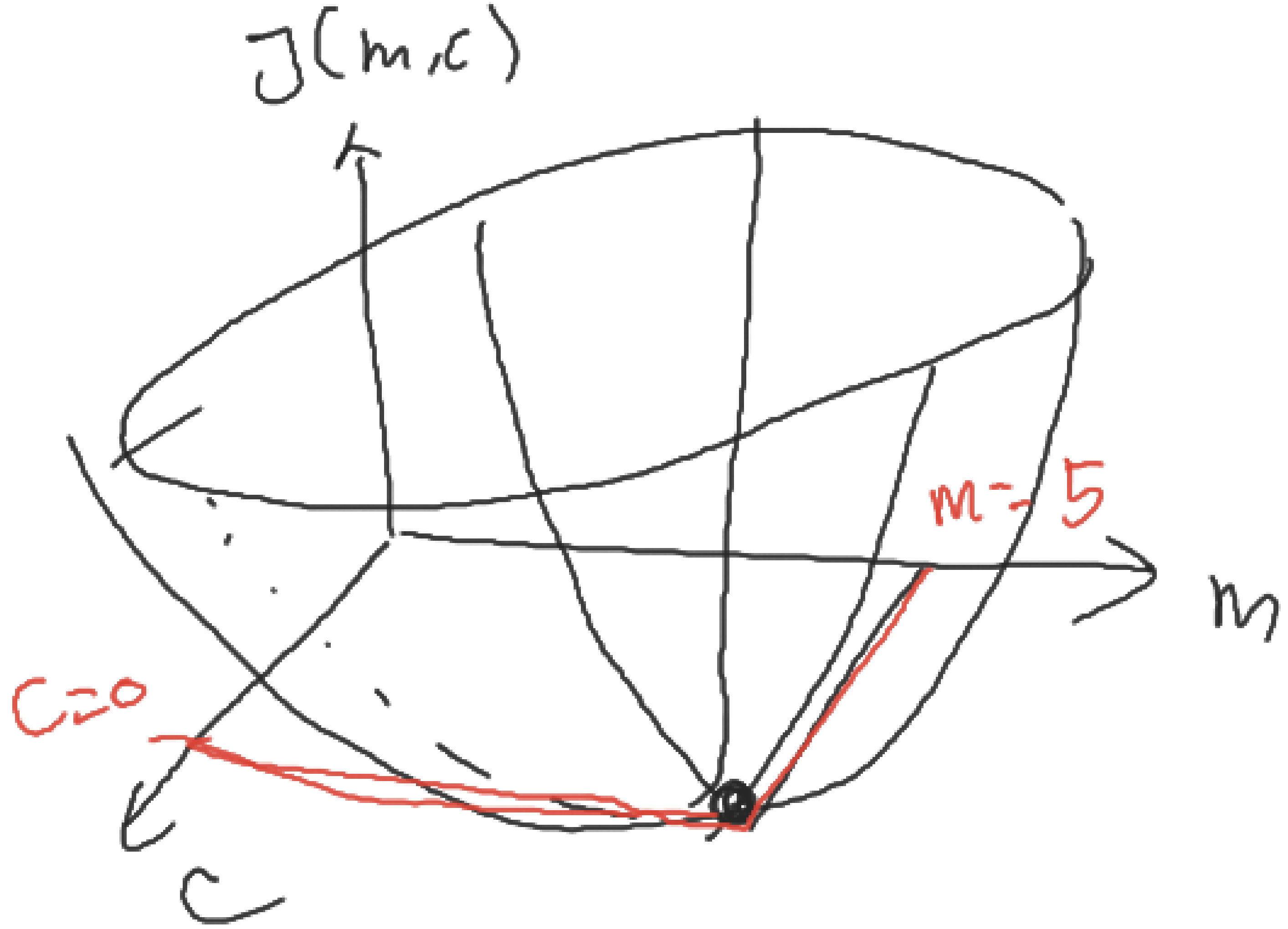
THIS IS HOW WE GO DOWN THE GRADIENT OF THE COST FUNCTION

We define the cost function

$N = 1338$

① $J(m,c) = \frac{1}{2N} \sum_{i=1}^{N} (h(x^i) - y^i)^2$





② we find the best value of m and c which minimizes cost function J.

③ This is done by going down the derivative/gradient of J

$M_{new} = M_{old} - \alpha \frac{dJ(m,c)}{dm}$

$C_{new} =$