Linear Regression

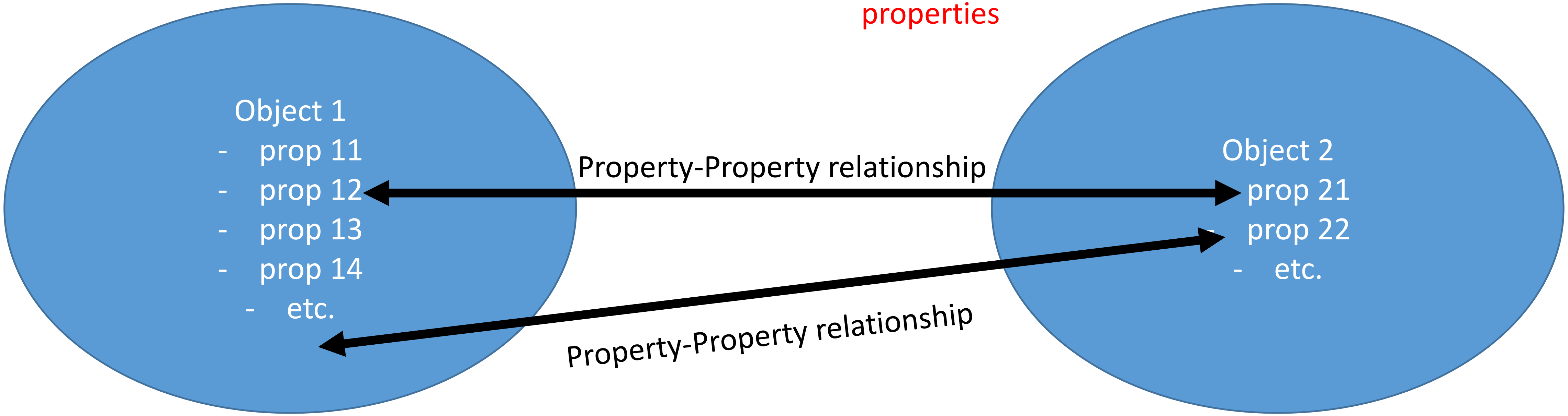
Dr. Kalidas Y., IIT Tirupati

In this lecture you will learn about LOSS Function and SOLVER formulation

Objects can be ANY… literally ANY..! concept in the world!!

(1) Identify objects and properties within each object

(2) A rough understanding of which properties relate to which other properties



(3) Vector representation of objects

Convert <noun/pronoun/etc> → <Noun>objects,
e.g. car -> CarObject, he -> PersonObject etc.

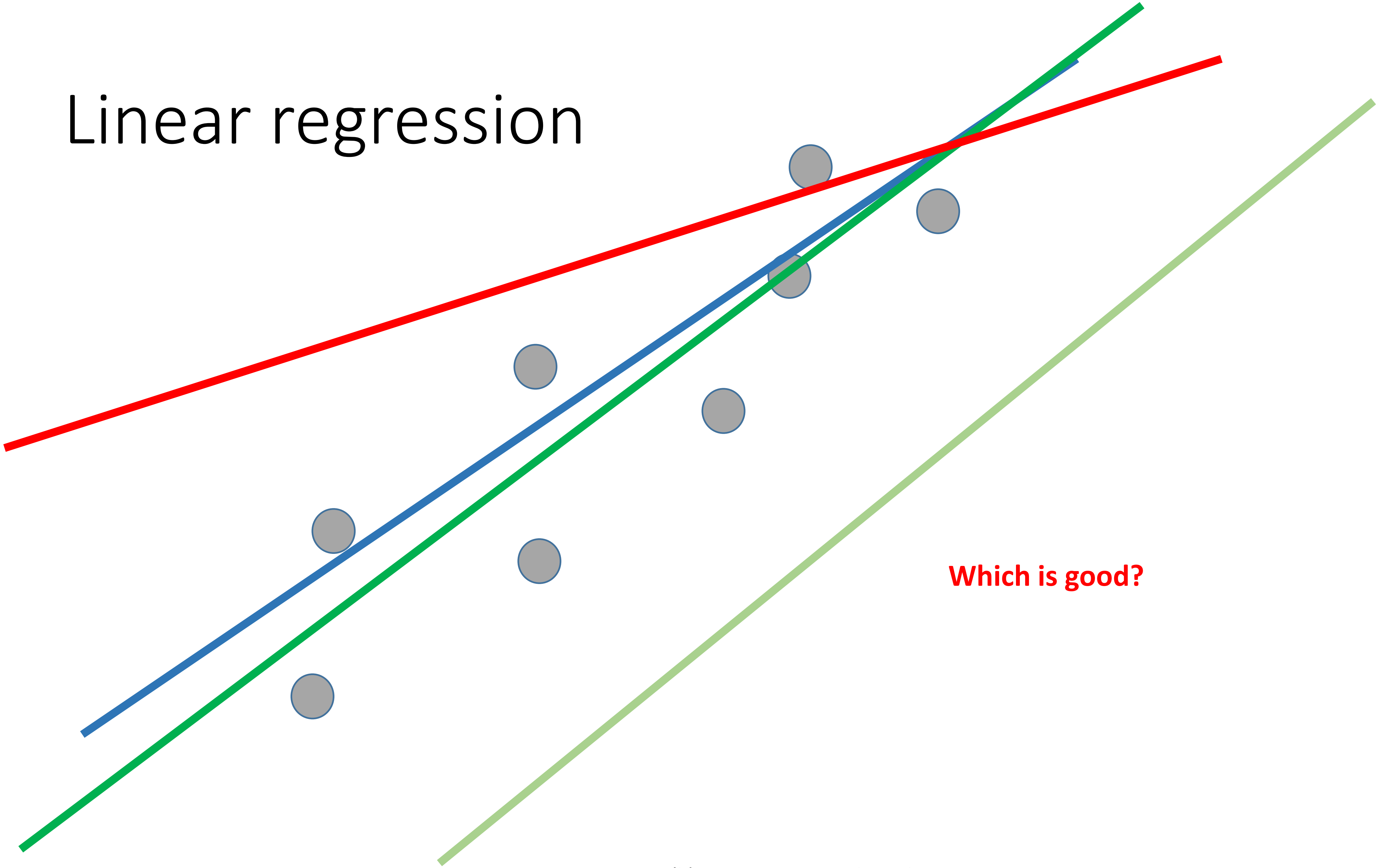
<verb → <Verb>-er, <Verb>-able objects,
e.g. walk -> Walk-er or Walk-able, talk -> Talk-er, Talk-able


AgeObject
- age value
HeightObject
- height value

AudioRecordingTimeDurationObject
- time value

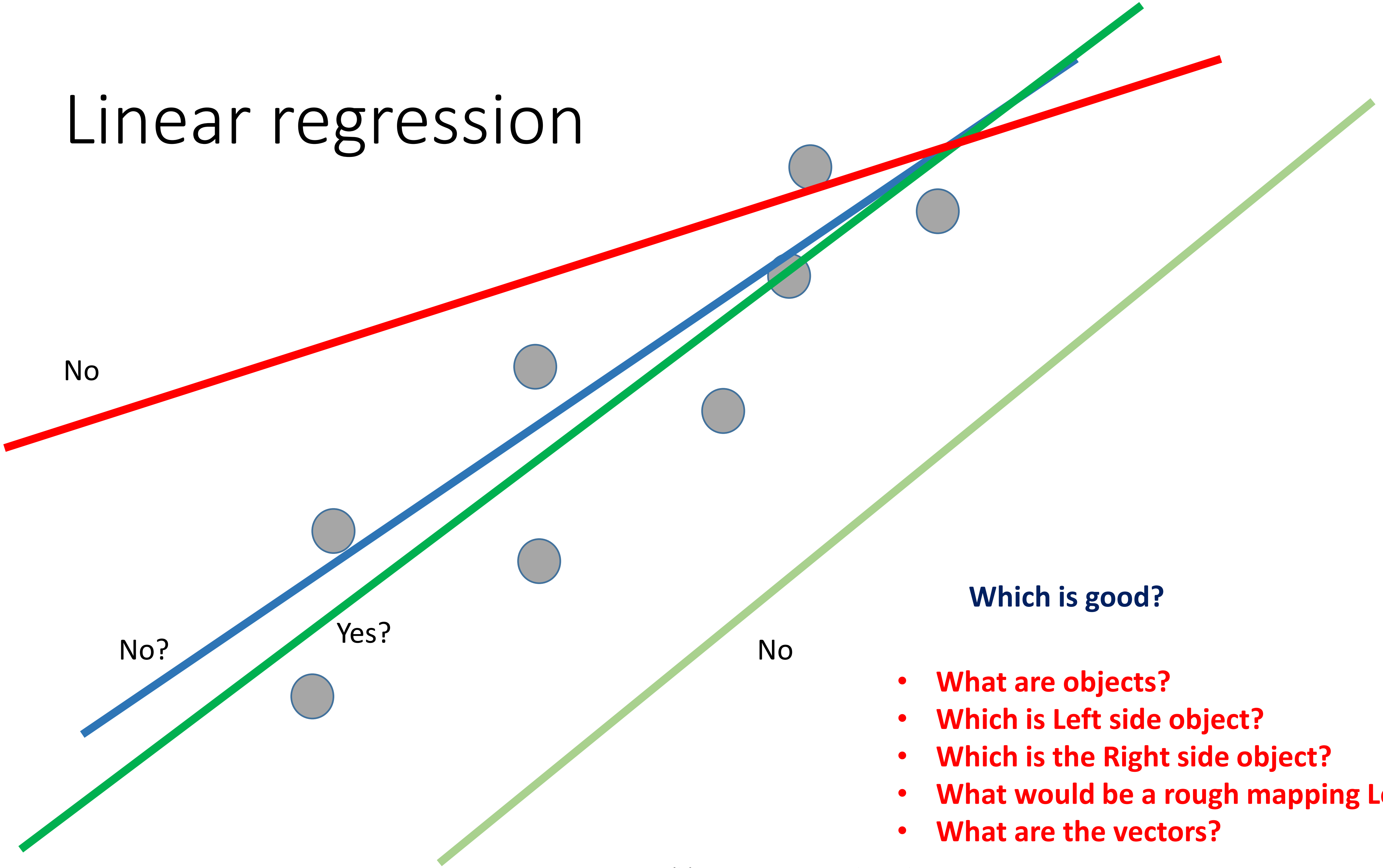
AudioRecordingFileSizeObject
- file size value

Linear regression



Which is good?

Linear regression



Which is good?

- **What are objects?**
- **Which is Left side object?**
- **Which is the Right side object?**
- **What would be a rough mapping Left to Right?**
- **What are the vectors?**

Linear regression

Where are objects?

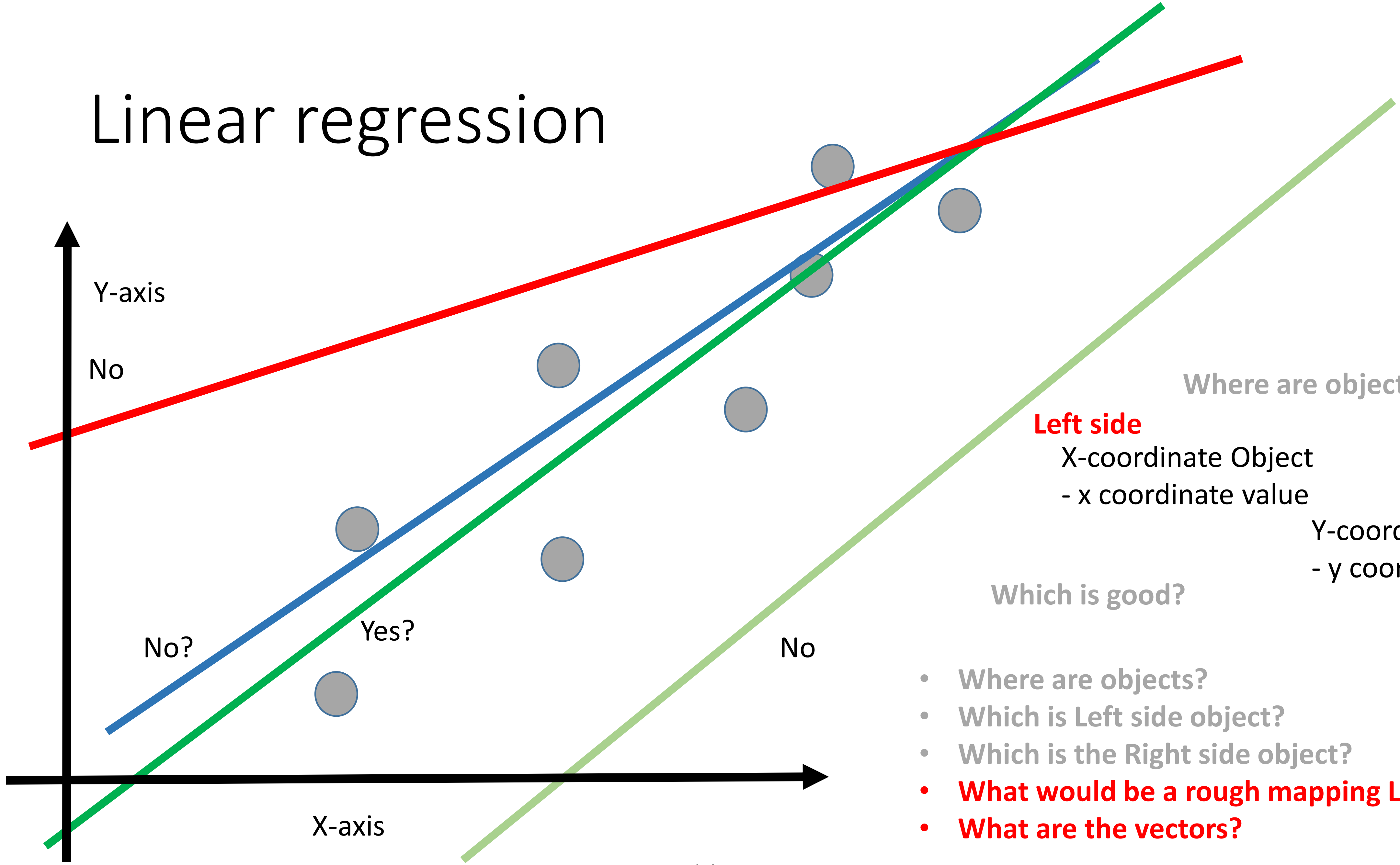
X-coordinate Object
- x coordinate value

Y-coordinate Object
- y coordinate value

Which is good?

- **Where are objects?**
- **Which is Left side object?**
- **Which is the Right side object?**
- **What would be a rough mapping Left to Right?**
- **What are the vectors?**

Linear regression



Where are objects?

Left side

X-coordinate Object

- x coordinate value

Right side

Y-coordinate Object

- y coordinate value

Which is good?

- Where are objects?
- Which is Left side object?
- Which is the Right side object?
- **What would be a rough mapping Left to Right?**
- **What are the vectors?**

Linear regression

Where are objects?

Left side

X-coordinate Object

- x coordinate value

Right side

Y-coordinate Object

- y coordinate value

Which is good?

- Where are objects?
- Where is Left side object?
- Where is the Right side object?
- What would be a rough mapping Left to Right?
- **What are the vectors?**

Linear regression

Where are objects?

Left side

X-coordinate Object

- x coordinate value

vector of 1 element

Right side

Y-coordinate Object

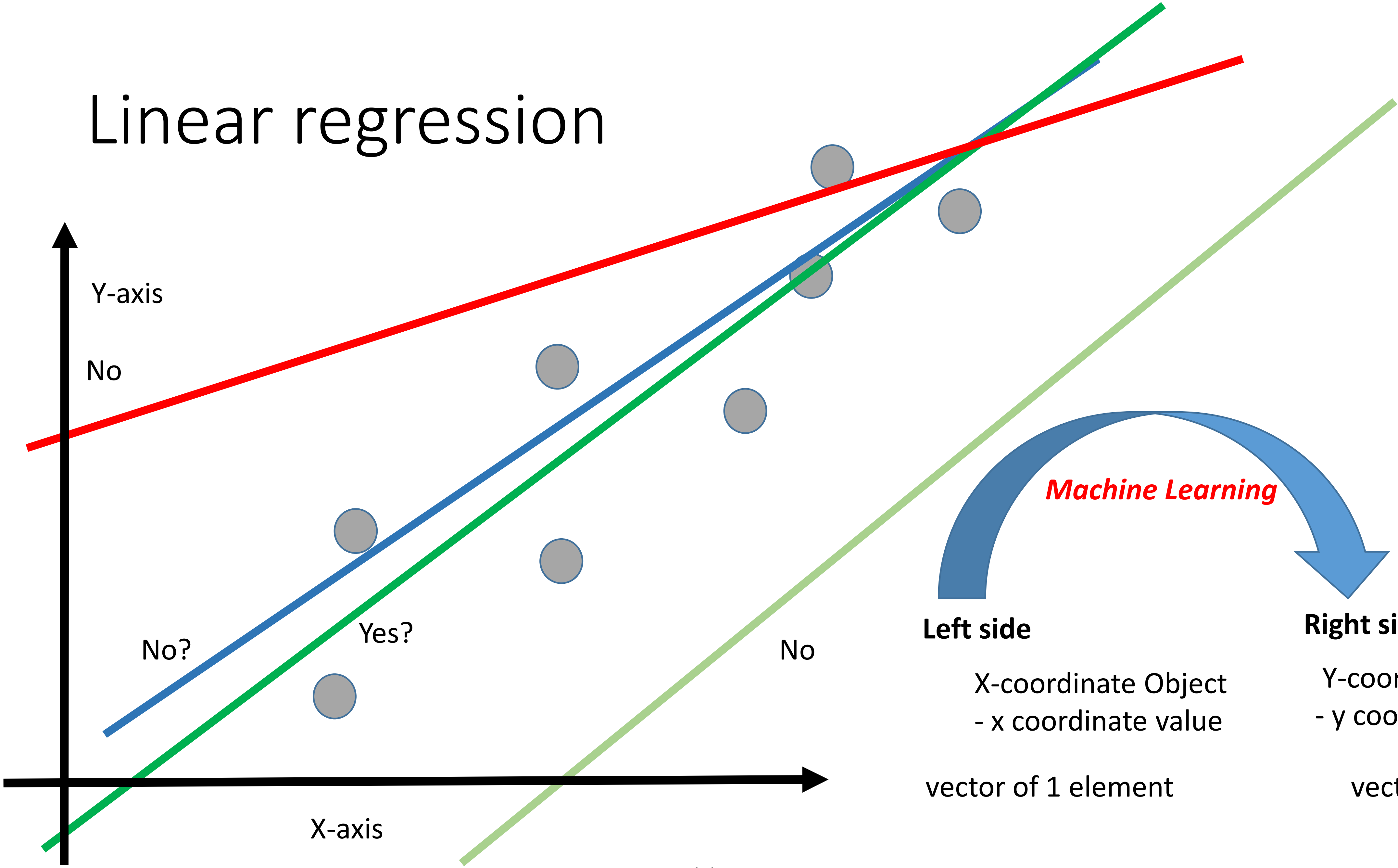
- y coordinate value

vector of 1 element

Which is good?

- Where are objects?
- Which is Left side object?
- Which is the Right side object?
- What would be a rough mapping Left to Right?
- What are the vectors?

Linear regression



Left side

X-coordinate Object
- x coordinate value

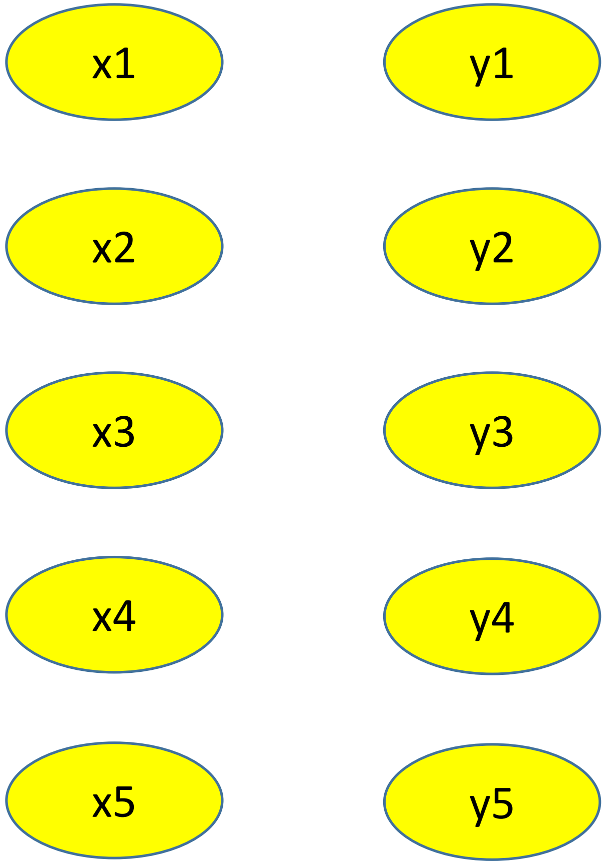
vector of 1 element

Right side

Y-coordinate Object
- y coordinate value

vector of 1 element

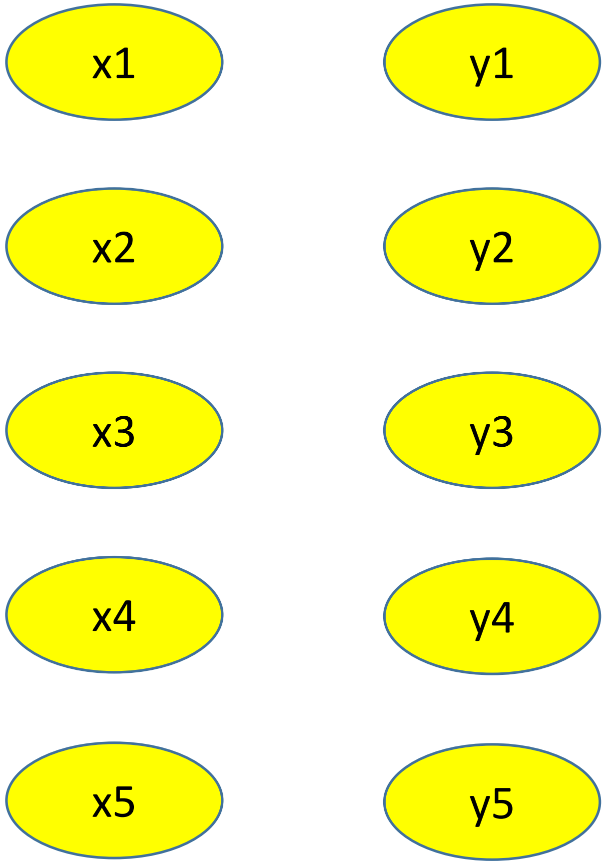
Representation



Mapping

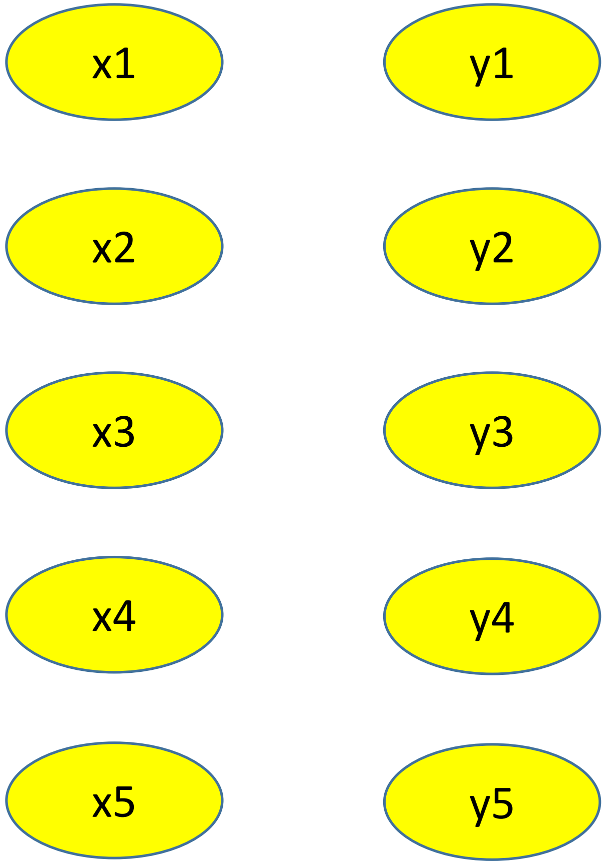
x_i vector to y_i vector

Representation



X	**Y** **Target or Label**
x1	y1
x2	y2
x3	y3
x4	y4
x5	y5

Representation



Mapping

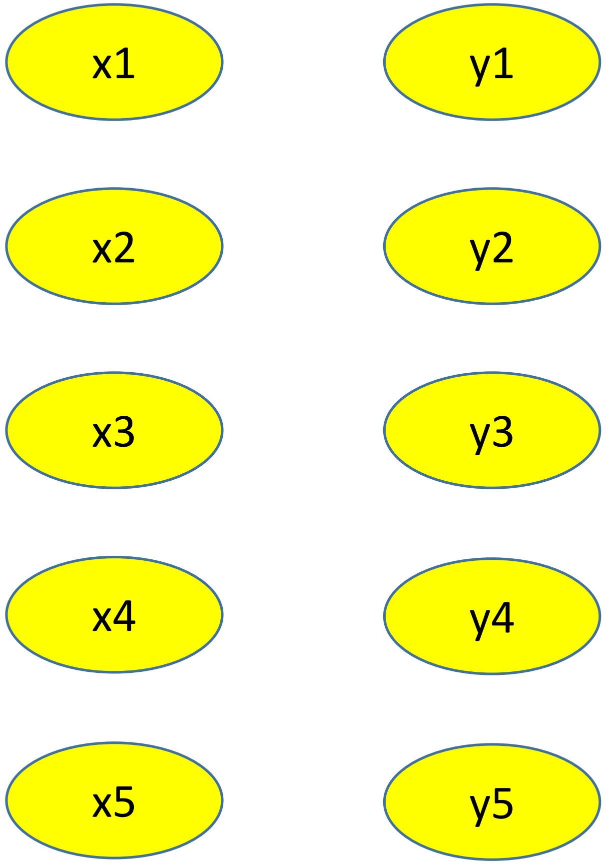
x_i vector to y_i vector

List notation

$$X = [x_i]_{i \in [1..5]}$$

$$Y = [y_i]_{i \in [1..5]}$$

Representation



Mapping

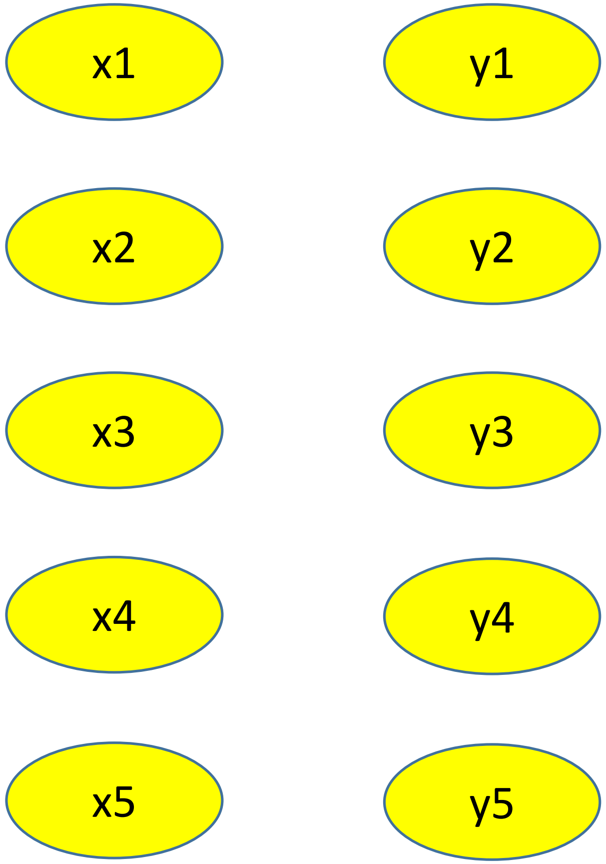
x_i vector to y_i vector

List notation

X

Y

Representation



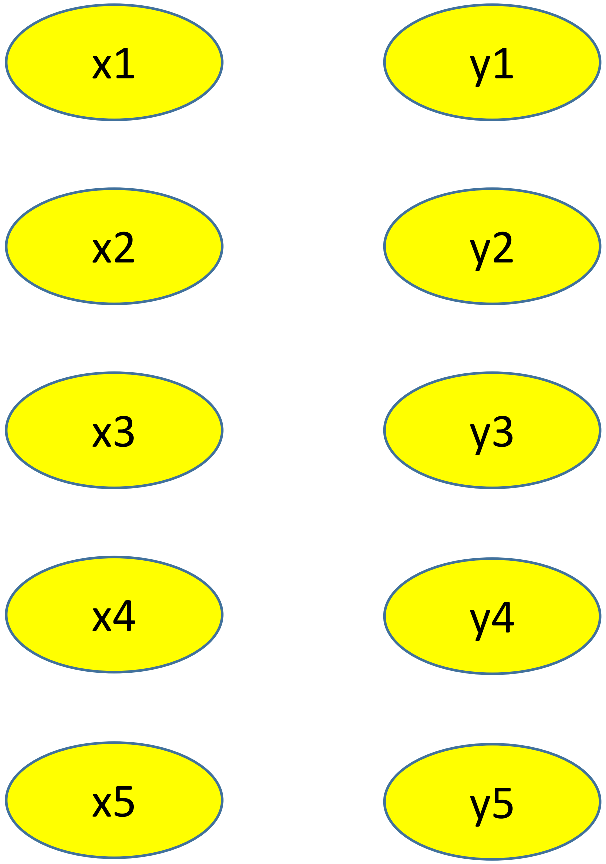
Mapping

x_i vector to y_i vector

List notation

$$Y = f(X)$$

Representation



Mapping

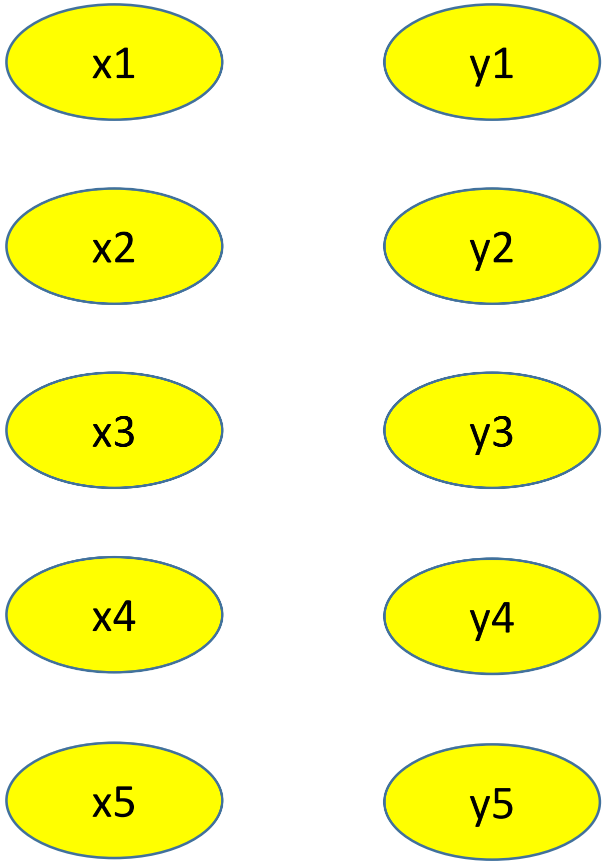
x_i vector to y_i vector

List notation

$Y = f(X)$

f() is called model

Representation



Mapping

x_i vector to y_i vector

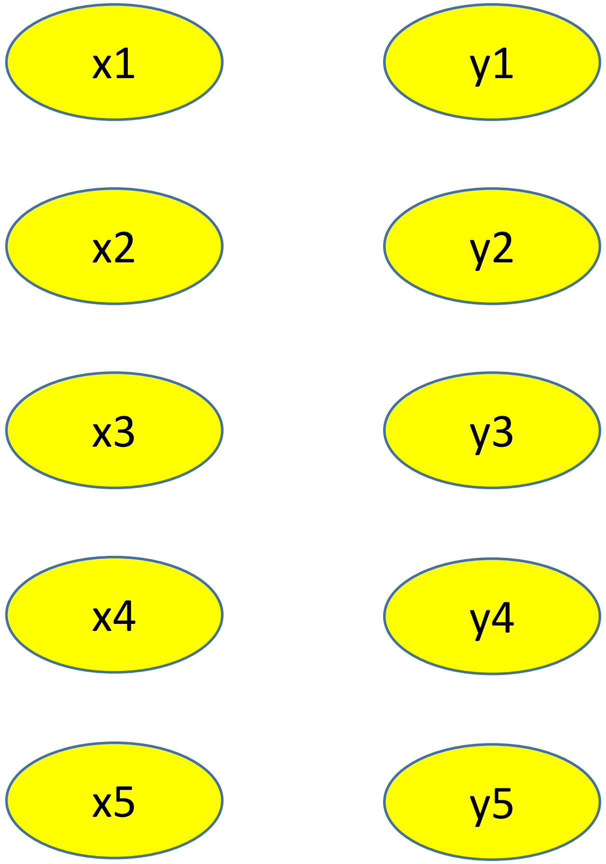
List notation

$Y = f(X)$

f() is called model

model's fit(X,Y)

Representation



Mapping

x_i vector to y_i vector

List notation

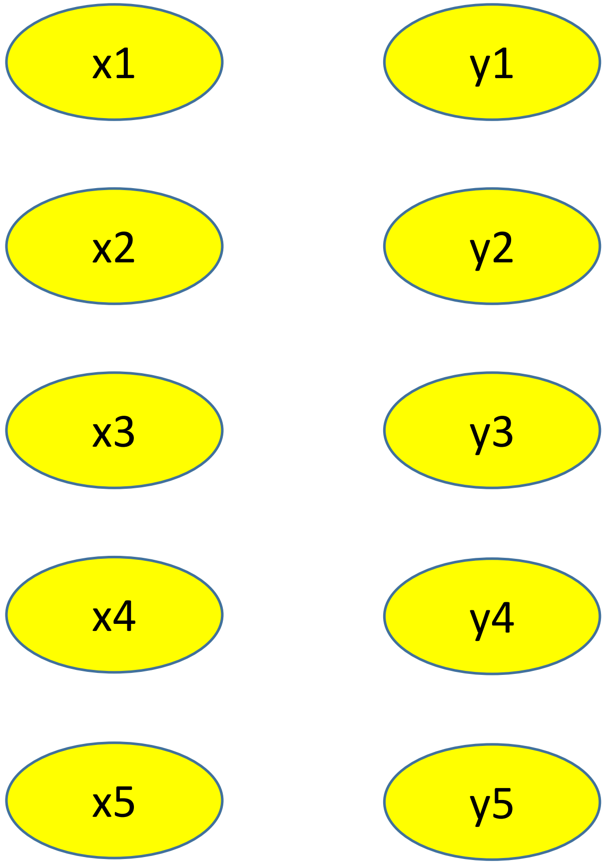
$Y = f(X)$

f() is called model

model's fit(X,Y)

model's predict($\mathbf{X}^{\mathbf{new}}$)

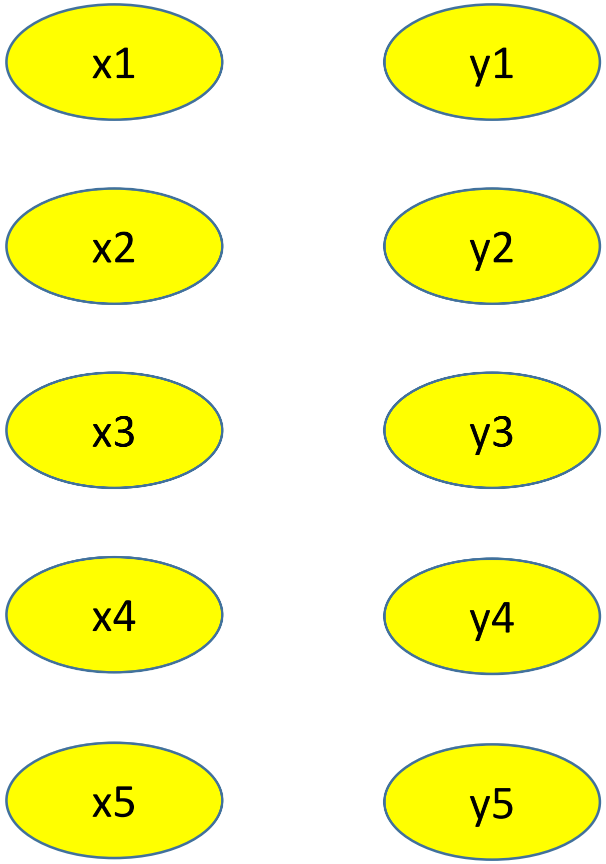
Left side (X) → Machine Learning → Right side (Y)



PREDICT..

Predict Y (right side), Given X (left side)

Left side (X) → Machine Learning → Right side (Y)



$y = x^2$? or..

$y = x^3$? or..

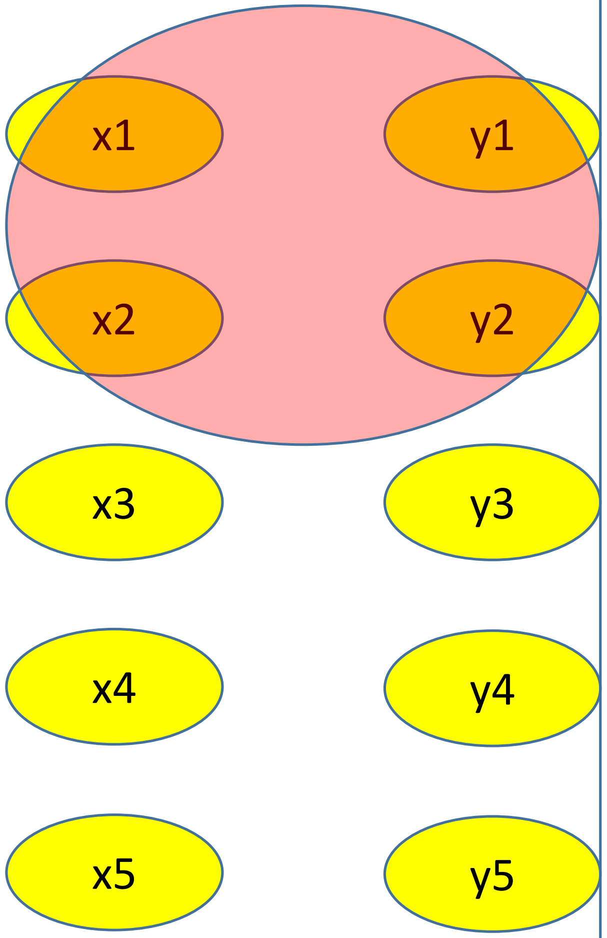
$y = 3\,x + 34$? or..

$y = -67\,x^2 + 45$? or..

????

????

Left side (X) → Machine Learning → Right side (Y)



Line…

passing through two points (x1,y1) and (x2,y2)

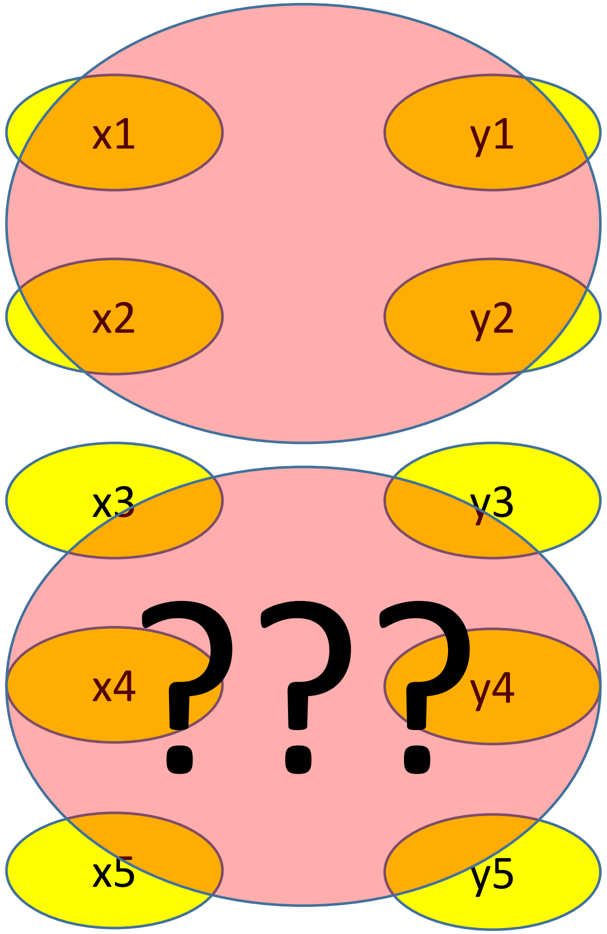
$$y = m\,x + c$$

$$\frac{y - y1}{x - x1} = \frac{y2 - y1}{x2 - x1}$$

$$y = \frac{y2 - y1}{x2 - x1} * \mathrm{x} + \left(\mathrm{y1} - \mathrm{x1} * \frac{y2 - y1}{x2 - x1}\right)$$

Linear Regression

Left side (X) → Machine Learning → Right side (Y)



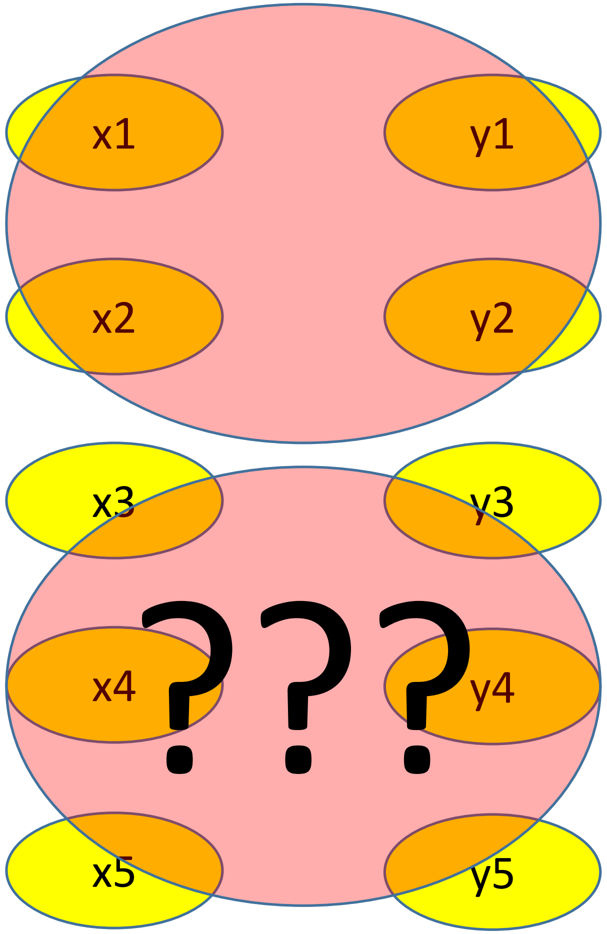
Line passing through two points (x1,y1) and (x2,y2)

$$y = m\,x + c$$

$$\frac{y - y1}{x - x1} = \frac{y2 - y1}{x2 - x1}$$

$$y = \frac{y2 - y1}{x2 - x1} * \mathrm{x} + \left(\mathrm{y1}\ - \mathrm{x1}\ * \ \frac{y2 - y1}{x2 - x1}\right)$$

Left side (X) → Machine Learning → Right side (Y)



Line passing through two points (x1,y1) and (x2,y2)

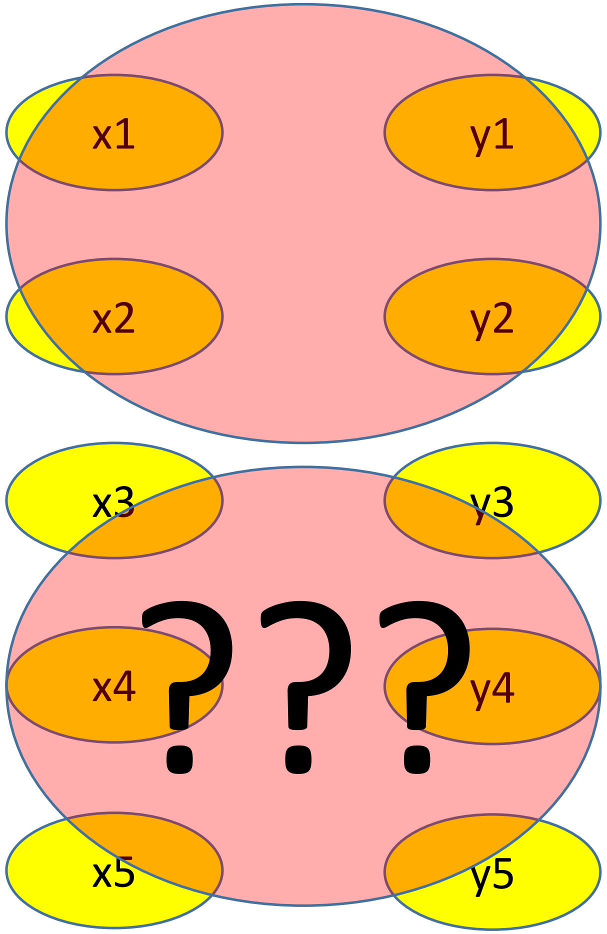
$$y = \frac{y2 - y1}{x2 - x1} * \mathrm{x} + \left(\mathrm{y1} \; - \mathrm{x1} \; * \; \frac{y2 - y1}{x2 - x1} \right)$$

Substitute (x,y) = (x3,y3)

Substitute (x,y) = (x4,y4)

…and then do what!!??

Left side (X) → Machine Learning → Right side (Y)



Line passing through two points (x1,y1) and (x2,y2)

$$y = \frac{y2 - y1}{x2 - x1} * \mathrm{x} + \left(\mathrm{y1} - \mathrm{x1} * \frac{y2 - y1}{x2 - x1}\right)$$

~~Substitute (x,y) = (x3,y3)~~
~~Substitute (x,y) = (x4,y4)~~

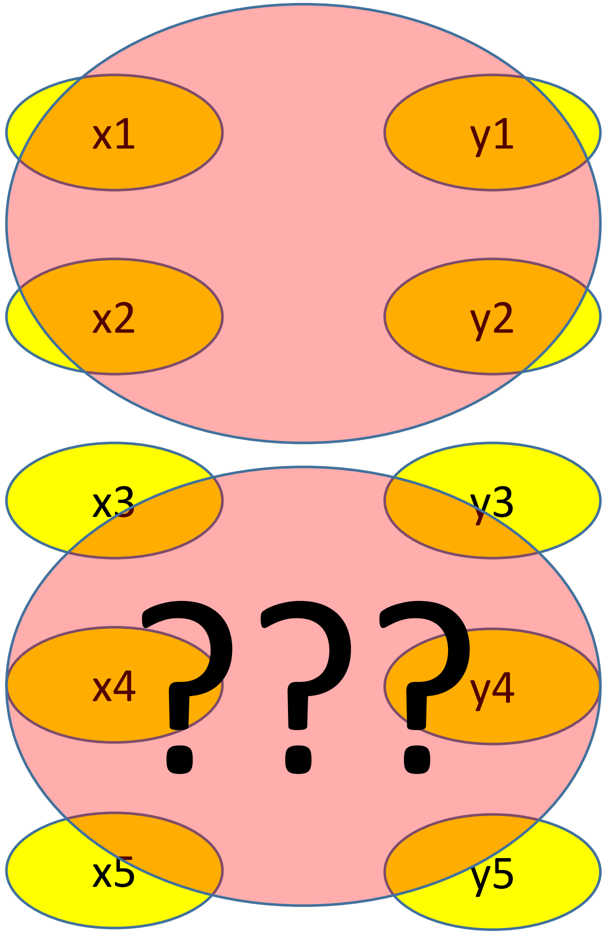
Need goodness measure!

Need badness measure!

Need quality measure!

Linear Regression

Left side (X) → Machine Learning → Right side (Y)

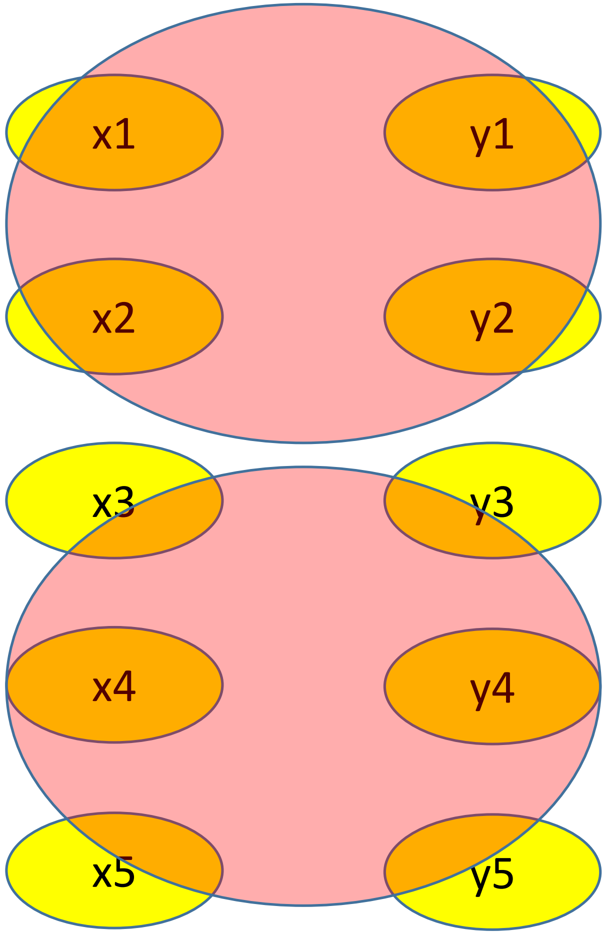


Line passing ~~through two points (x1,y1) and (x2,y2)~~

$$\cancel{y = \frac{y2 - y1}{x2 - x1} * \mathrm{x} + \left(\mathrm{y1} - \mathrm{x1} * \frac{y2 - y1}{x2 - x1}\right)}$$

Linear Regression

Left side (X) → Machine Learning → Right side (Y)

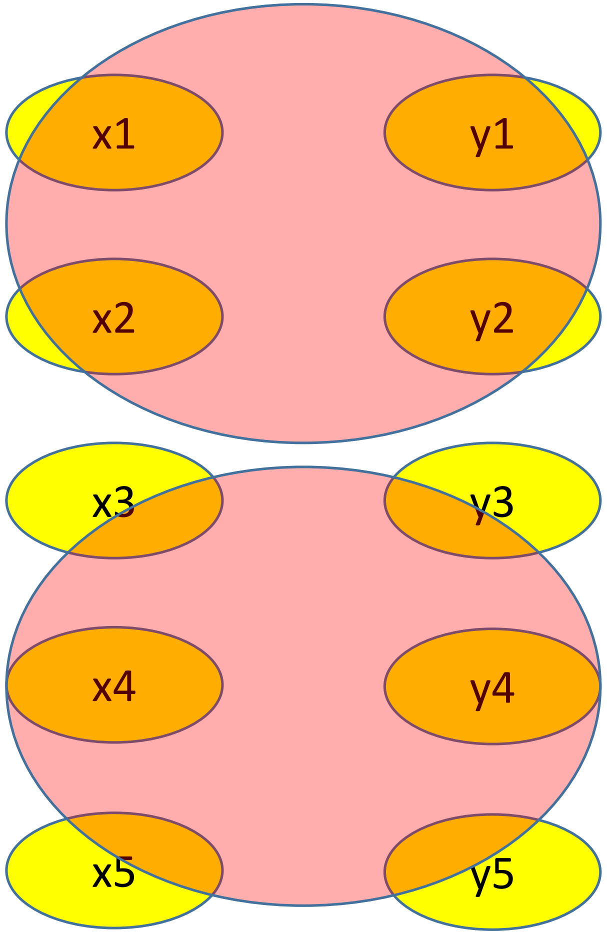


Equation of Line :- $y = m * x + c$

Need goodness or badness measure!

Linear Regression

Left side (X) → Machine Learning → Right side (Y)

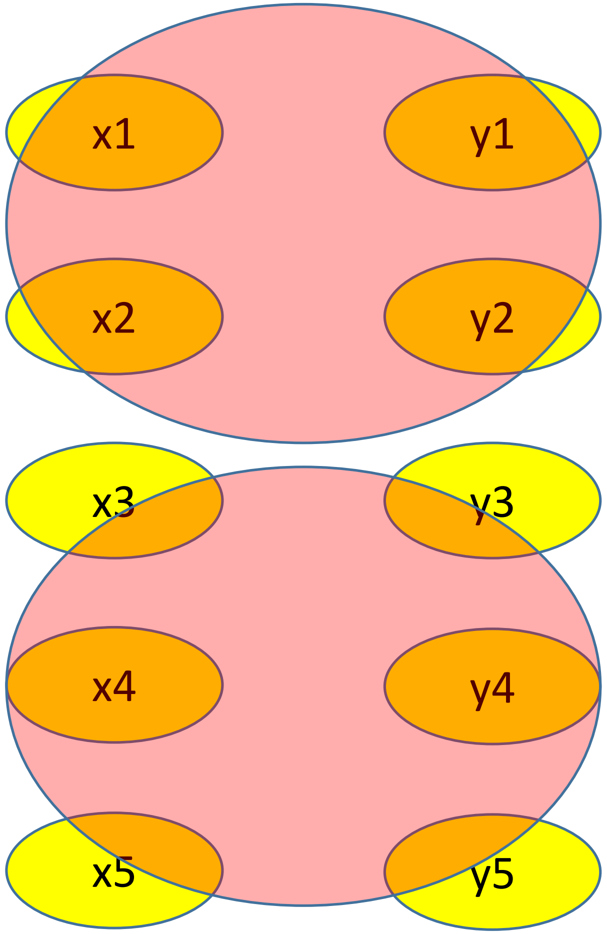


Equation of Line :- $\mathrm{y} = \mathrm{m} * x + c$

$$y1 = m * x1 + c\ ?$$
$$y2 = m * x2 + c\ ?$$
$$\ldots$$
$$y5 = m * x5 + c\ ?$$

???
Left side = Exactly same as = Right side ???
Exactly can they be same?
For any value of m and c
What is learning here?

Left side (X) → Machine Learning → Right side (Y)



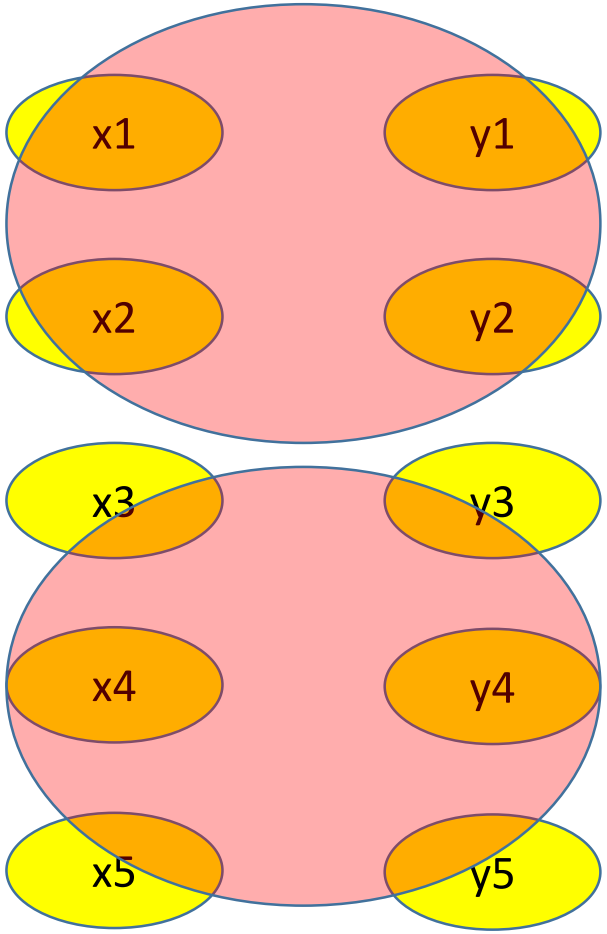
Equation of Line :- $y = m * x + c$

$$y1 = m * x1 + c\ ?$$
$$y2 = m * x2 + c\ ?$$
$$\ldots$$
$$y5 = m * x5 + c\ ?$$

What could be a badness measure?...

Linear Regression

Left side (X) → Machine Learning → Right side (Y)



Equation of Line :- $y = m * x + c$

Badness for (x1,y1) :- $(y1 - (m * x1 + c))^2$

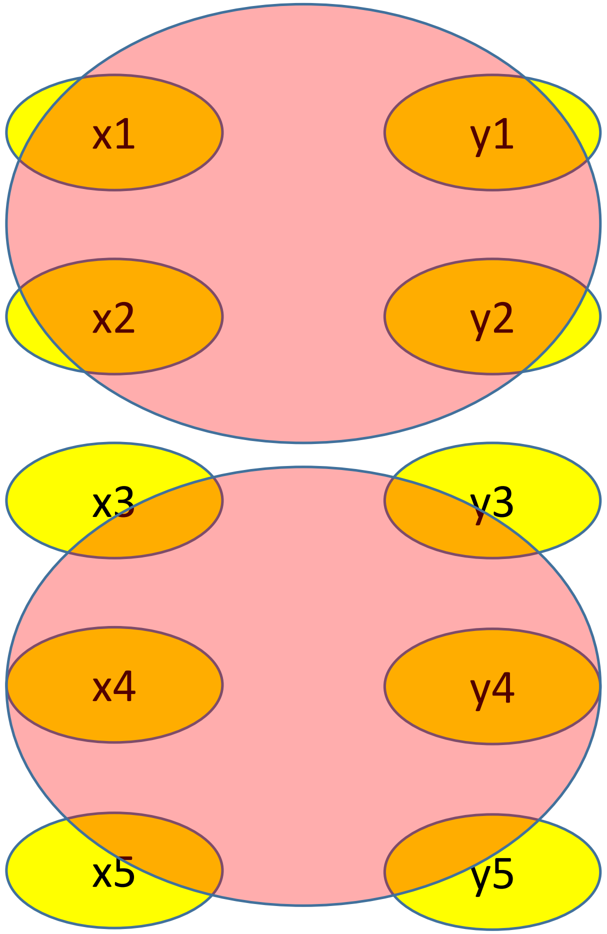
Badness for (x2,y2) :- $(y2 - (m * x2 + c))^2$

...

Badness for (x5,y5) :- $(y5 - (m * x5 + c))^2$

Linear Regression

Left side (X) → Machine Learning → Right side (Y)



Equation of Line :- $\mathrm{y} = \mathrm{m} \; * x + c$

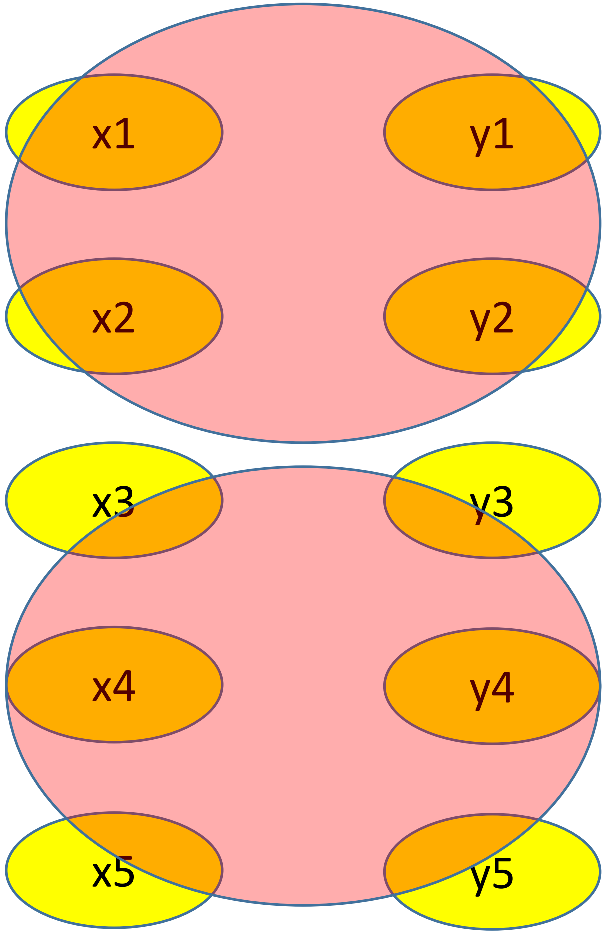
$$\mathrm{L}(\mathrm{m}, \mathrm{c}, \mathrm{X}, \mathrm{Y}) = \sum_{i=1}^{i=5} (y_i - (m * x_i + c))^2$$

LOSS FUNCTION

ERROR FUNCTION

Linear Regression

Left side (X) → Machine Learning → Right side (Y)



Equation of Line :- $y = \mathrm{m} * x + c$

Actual.. Ground truth..

Prediction

$$\mathrm{L}(\mathrm{m}, \mathrm{c}) = \sum_{i=1}^{i=5} \left(y_i - (m * x_i + c)\right)^2$$

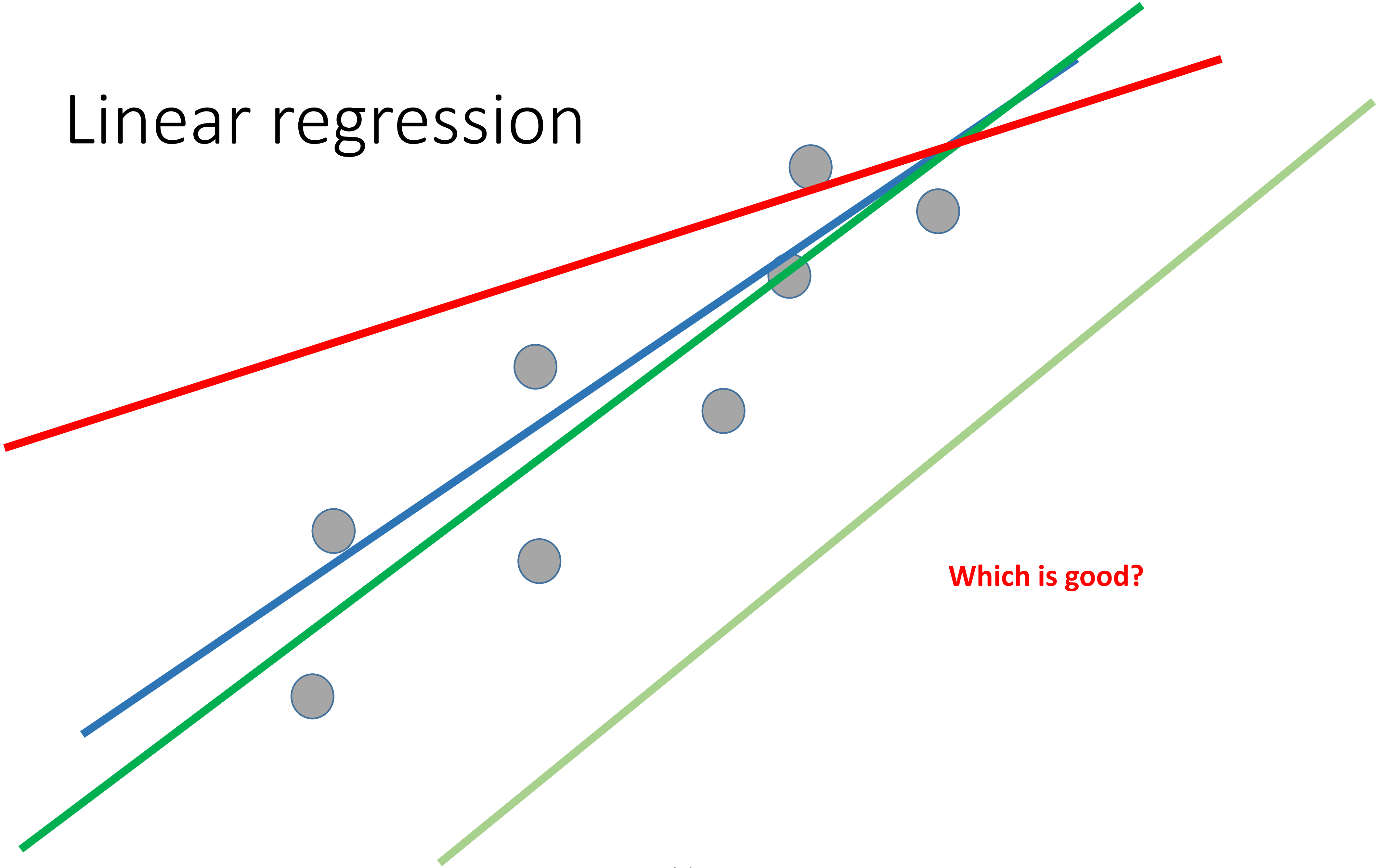
REGRESS..

prediction to return to.. actual

LOSS FUNCTION

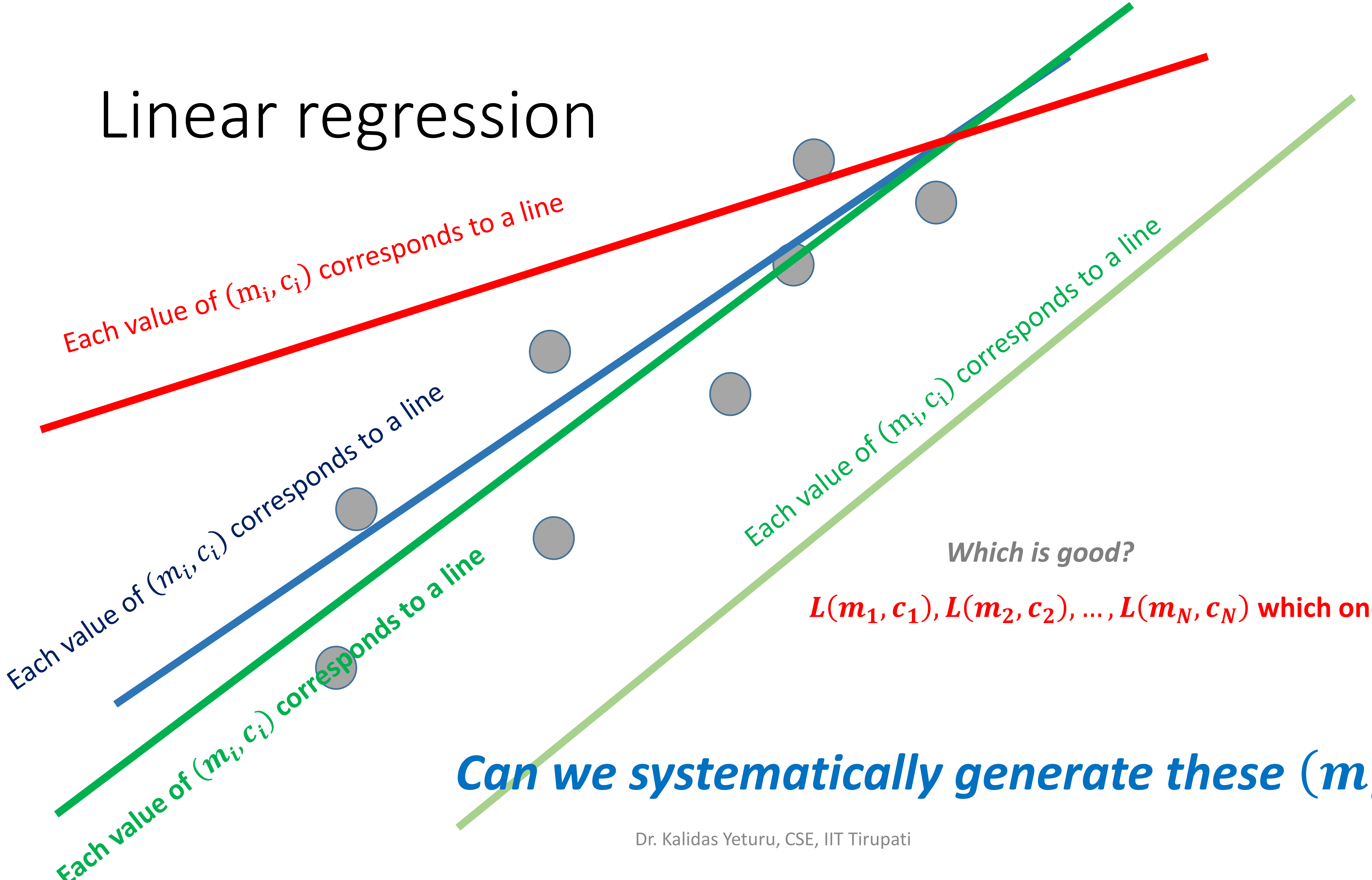
PARAMETERS

Linear regression



Which is good?

Linear regression



Which is good?

$L(m_1, c_1), L(m_2, c_2), \ldots, L(m_N, c_N)$ **which one is better?**

Can we systematically generate these (m_i, c_i)?

What are the different ways?

2nd argument is *the list of Parameters*

BruteForceSolver(L, [m1,c1])

1st argument is the *Loss function*

$\boldsymbol{minval = +10000.0}$

FOR $\mathbf{m} \in [\mathbf{-100, \dots, +100}]$

FOR $\mathbf{c} \in [\mathbf{-100, \dots, +100}]$

Compute $\boldsymbol{v = L(m, c)}$

IF $\boldsymbol{v < minval}$**:**

$\boldsymbol{v = minval}$

$\boldsymbol{m1 = m,}$

$\boldsymbol{c1 = c}$

Questions…

- *-100 to +100, who gave the range?*
- *What is the step size?*
- *What if the solution is highly fine, (3.451, -89.1123)*
- *Can we speed up?*

What are the different ways?

2nd argument is *the* ***list of Parameters***

RandomSolver(L, [m1,c1])

1st argument is the Loss function

$\boldsymbol{minval = +10000.0}$

FOR ITER= 1:100000

FOR $\mathbf{m} = \boldsymbol{RAND(-100, 100)}$

FOR $\boldsymbol{c = RAND(-100, 100)}$

Compute $\boldsymbol{v = L(m, c)}$

IF $\boldsymbol{v < minval}$**:**
$\boldsymbol{v = minval}$
$\boldsymbol{m1 = m,}$
$\boldsymbol{c1 = c}$

Questions…

- *-100 to +100, who gave the range?*
- *What is the step size?*
- *What if the solution is highly fine, (3.451, -89.1123)*
- *Can we speed up?*

What are the different ways?

2nd argument is *the list of Parameters*

GradiantSolver(L, [m1,c1], gradL)

1st argument is the Loss function

3rd argument is the gradient function