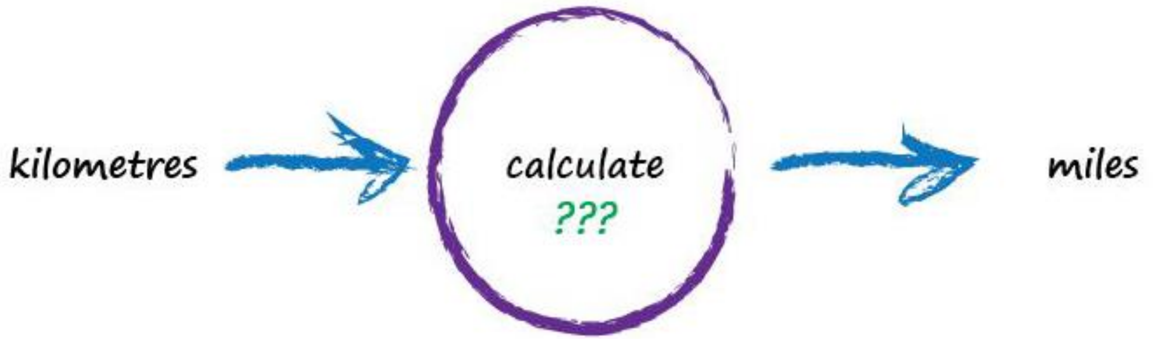
NN Through Simple Example

A Simple Predicting Machine

- How to predict without formula
- Relationship is linear
- Or if we double the number in miles, the same distance in kilometers is also doubled
- OR miles = km * c but c? Randomly choose c =0.5

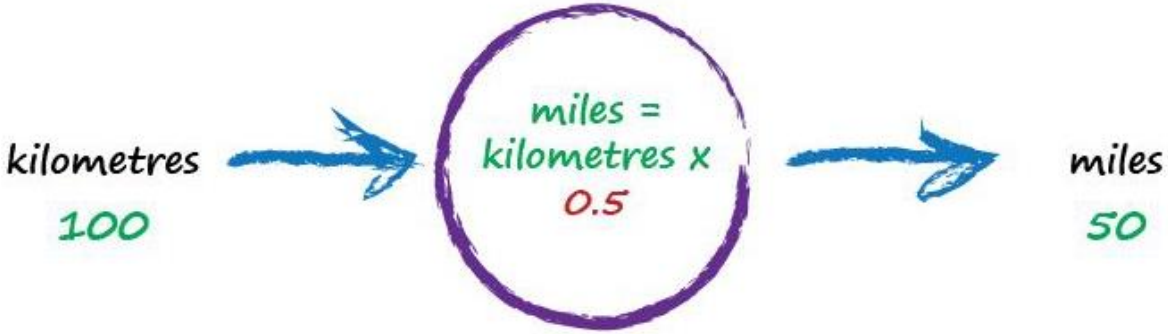


Truth Example	Kilometres	Miles
1	0	0
2	100	62.137

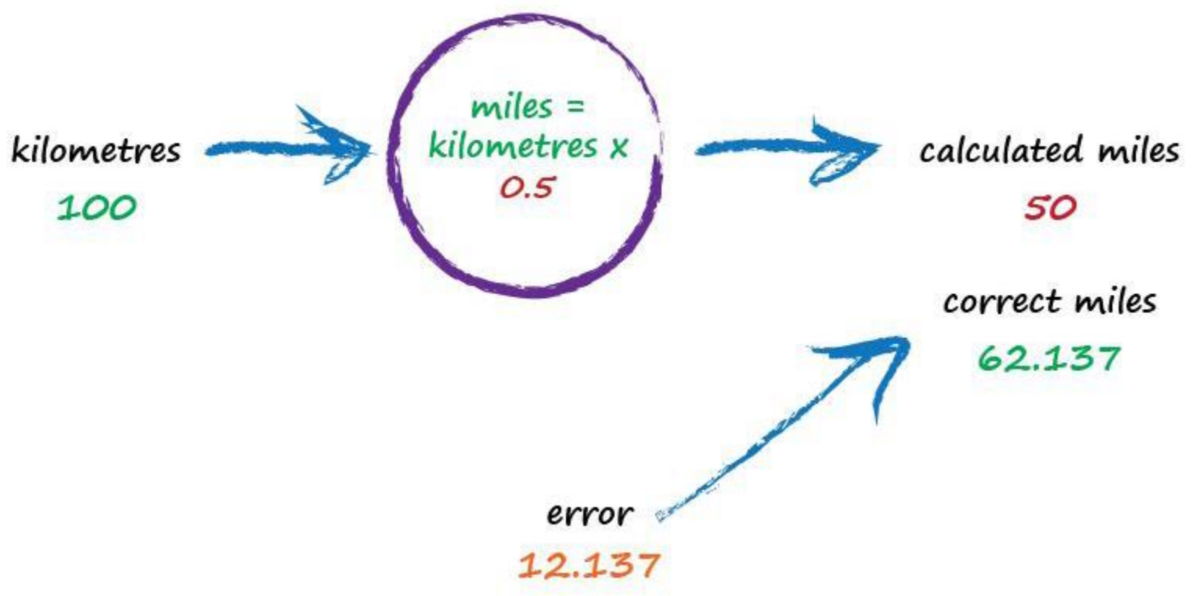
- answer should be 62.137
- **We get error**
- error = truth calculated

= 62.137 - 50

= 12.137



- How much we nudge the value of c?
- Correction is a fraction of the error.
- Big error means a bigger correction is needed, and a tiny error means the teeniest of nudges to **c** .
- This is the very core process of learning in a
- neural network to train the machine to get better and better at giving the right answer
- This process is **iterative,** and it means repeatedly improving an answer bit by bit



miles = km * c

Increase c to c=0.6

Error = 2.137

Increase to 0.7 c=0.7

Error = -7.863(overshot)

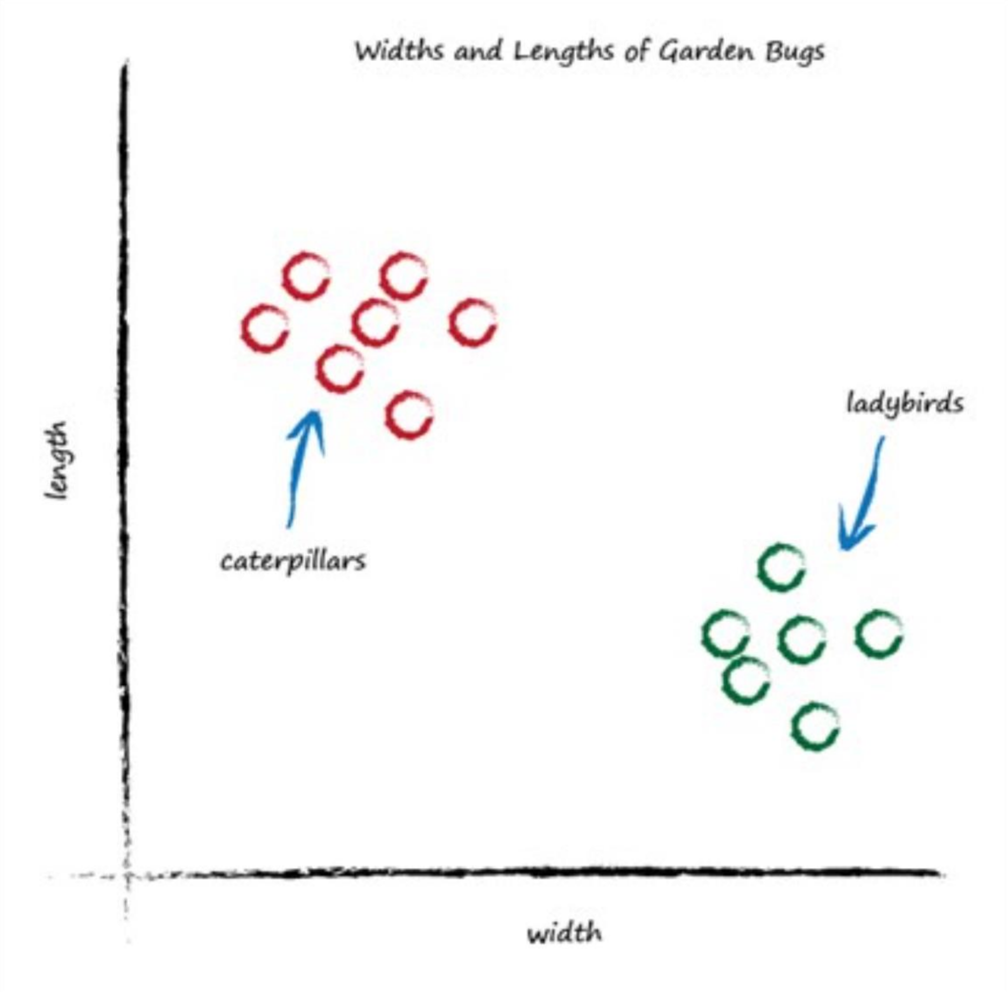
Now make it between 0.6 and 0.7

0.61 c= 0.61

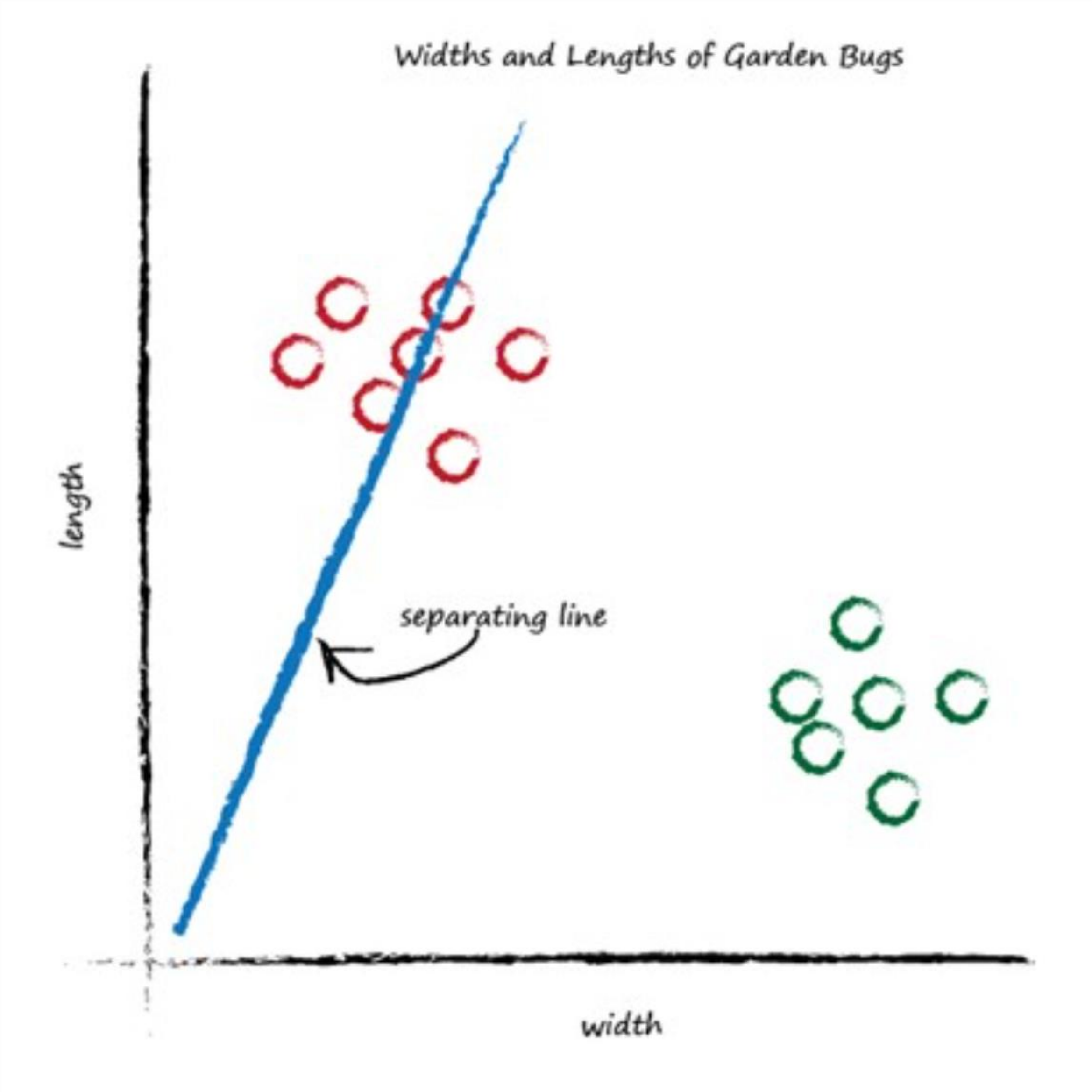
Error = 1.137

arden Bugs

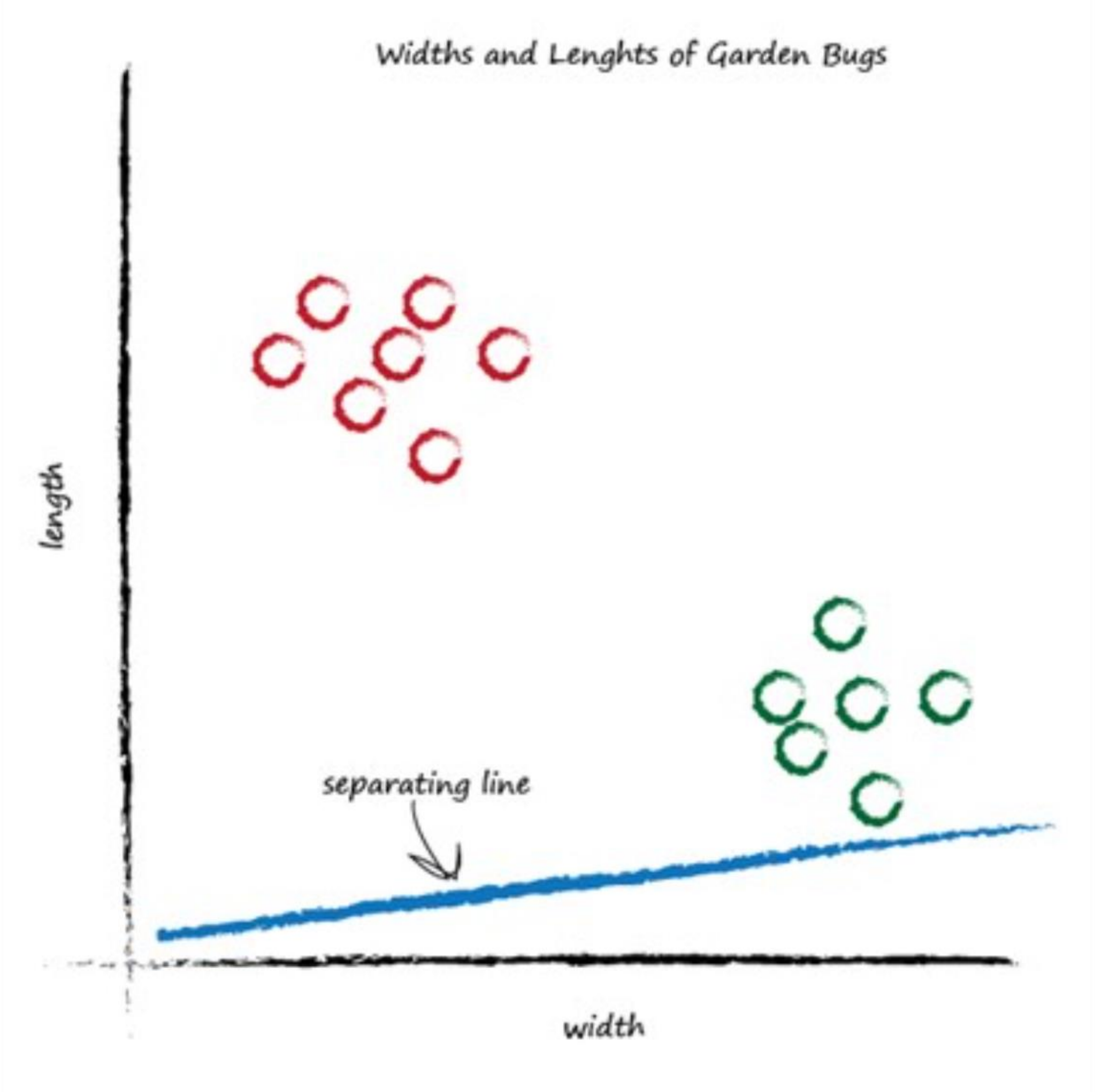
Training a Classifier for Garden Bug



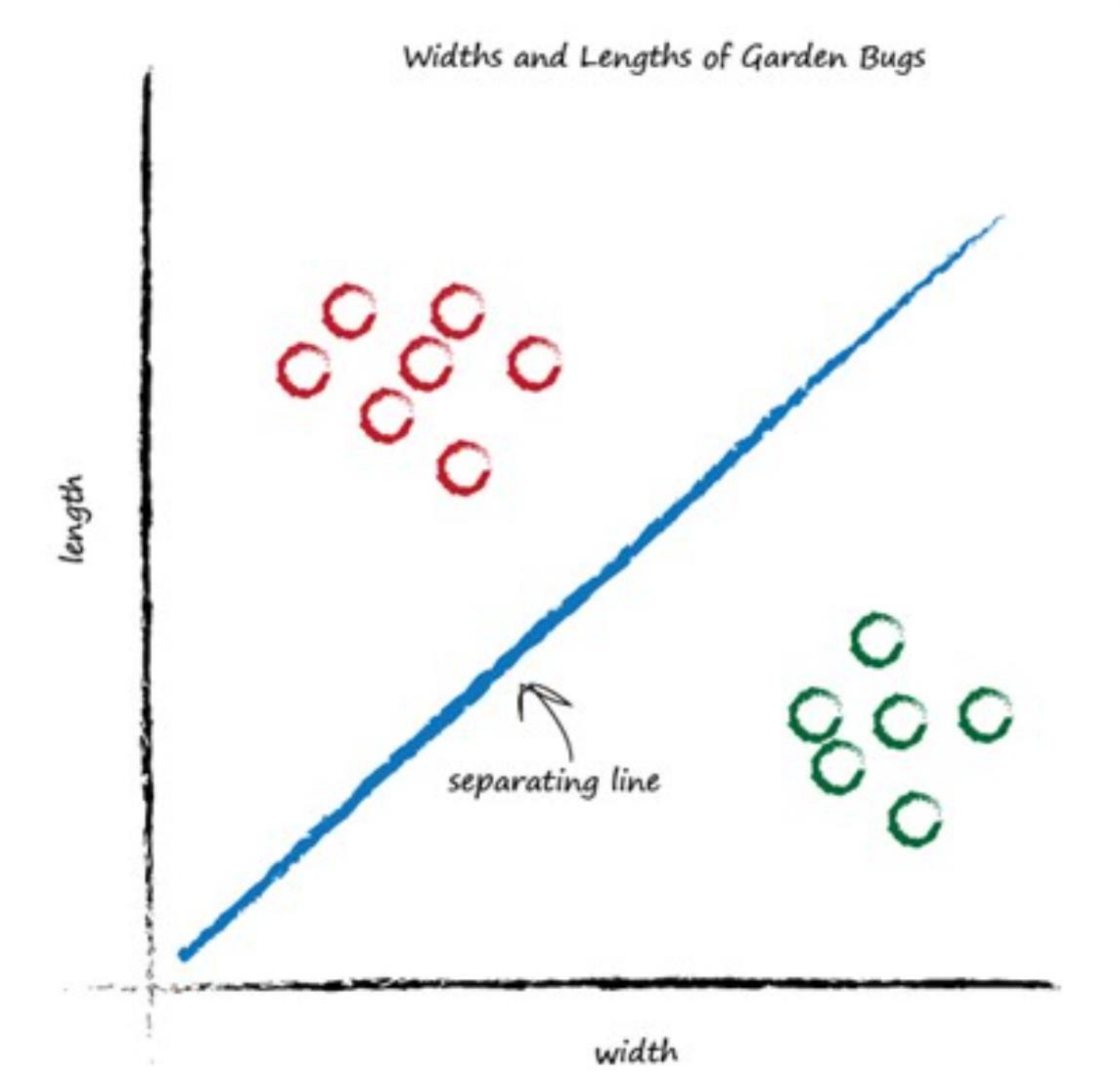
Classifying Bugs



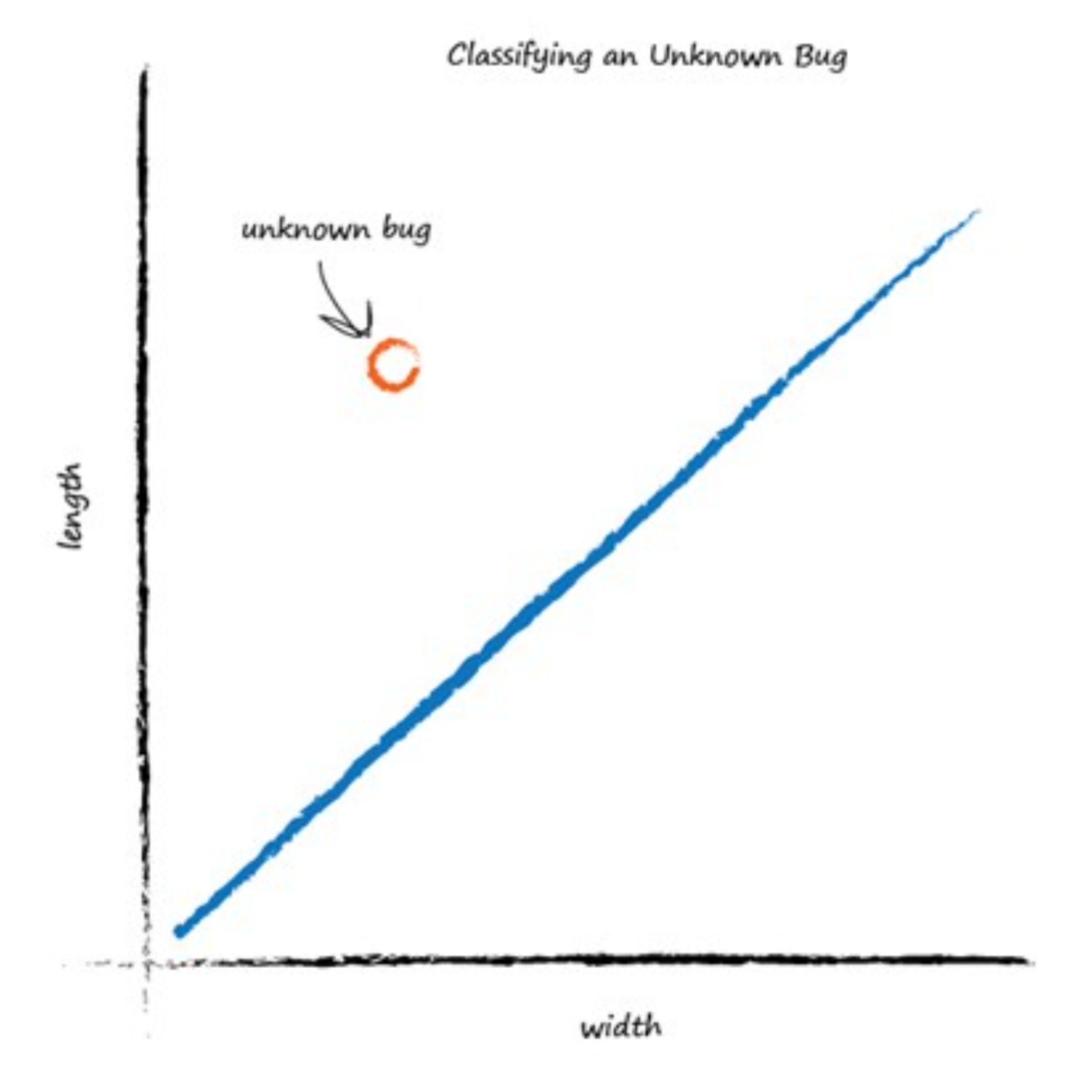
Classifying Bugs



Classifying Bugs



Classifying Bugs

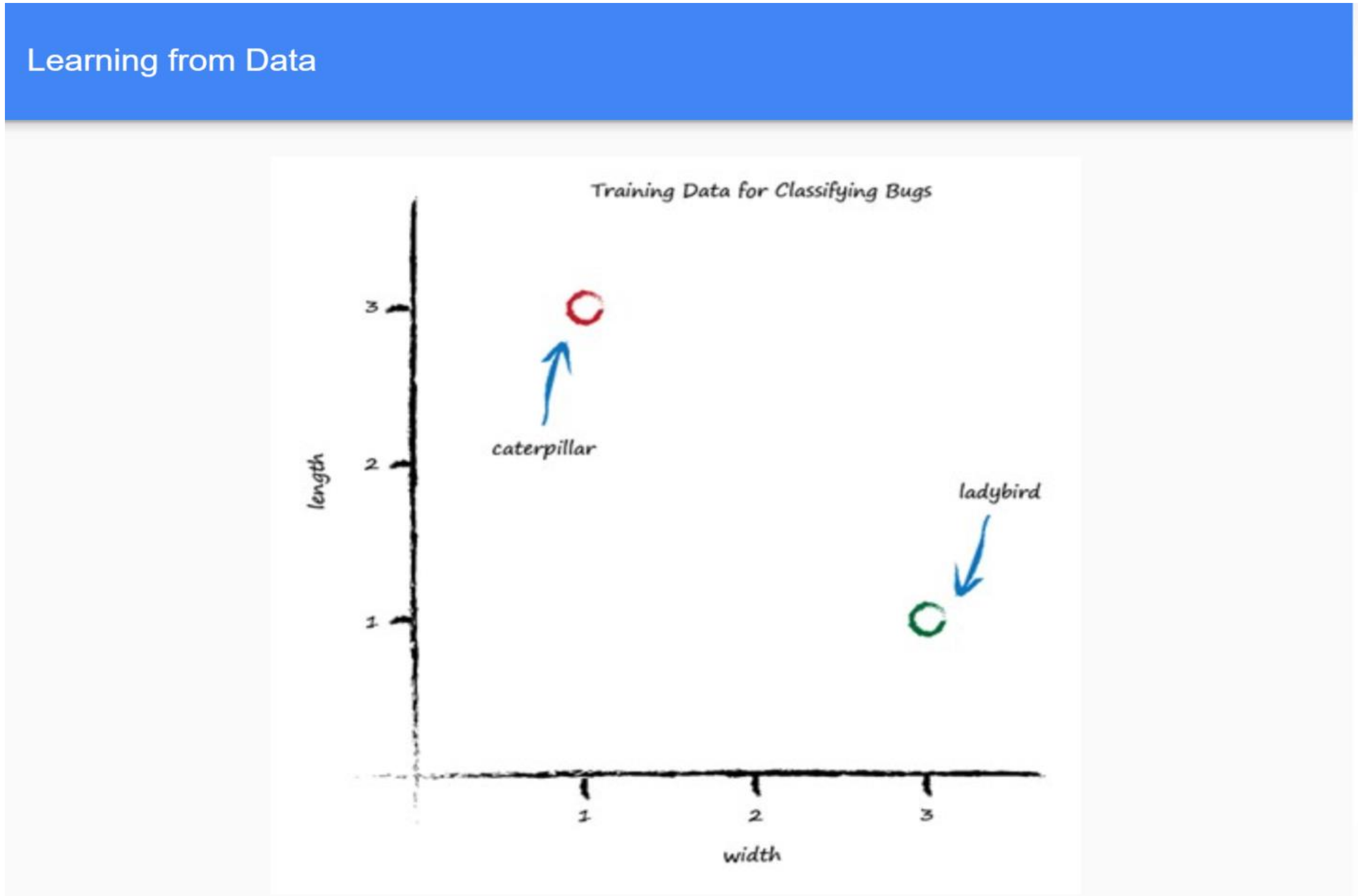


Training Dataset

- Examples of truth used to teach a predictor, or a classifier are called the **training data** .

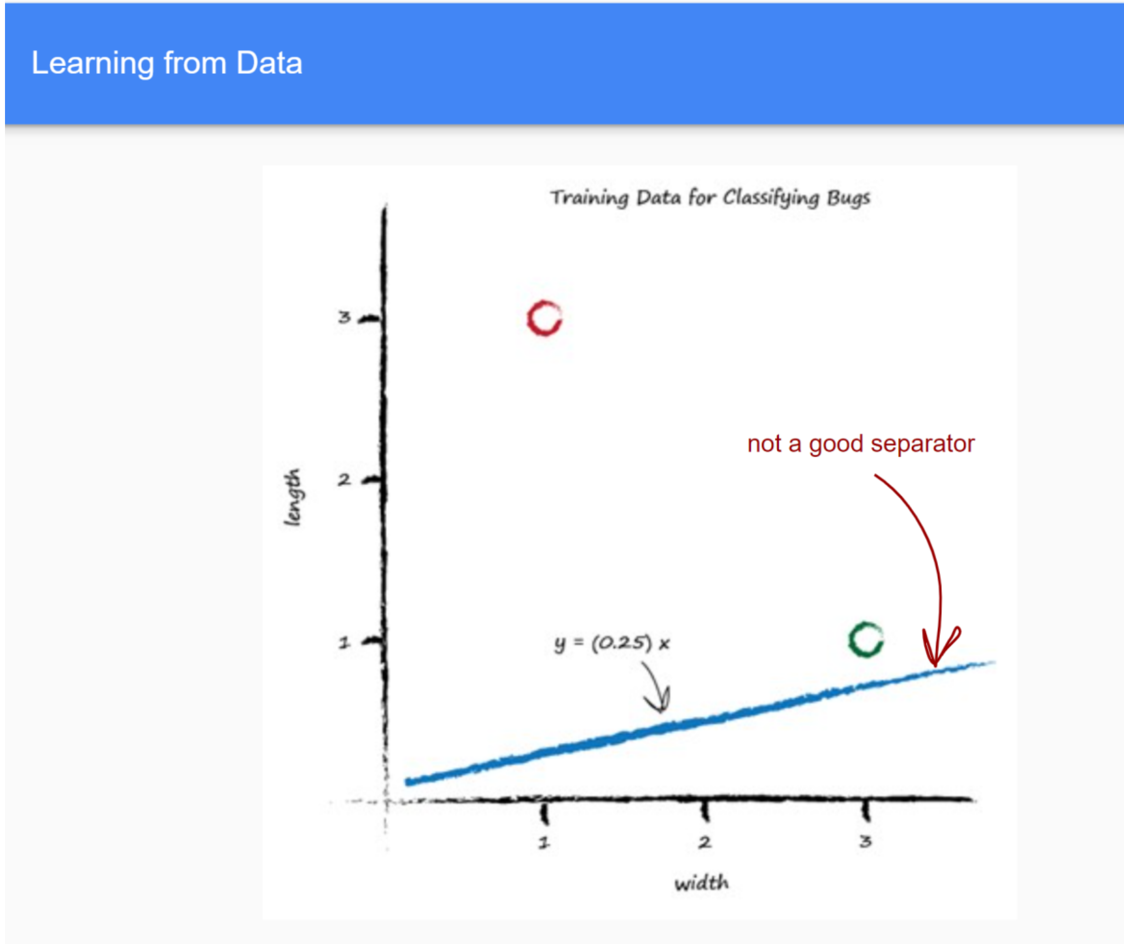
Example	Width	Length	Bug
1	3.0	1.0	ladybird
2	1.0	3.0	caterpillar

- A linear function as dividing line is straight Line so we can say that $y = Ax$
- Why the names **y** and **x** instead of length and width as done in km and miles example:
- Because the line is not a predictor here. It doesn't convert width to length, but it is a dividing line, a classifier.
- The parameter **A** controls the slope of the line.
- The larger **A** is the larger the slope.



How Training Data works

- **A** is 0.25 to get started. The dividing line is **y = 0.25x**
- **y = Ax**
- **How to adjust line:**
- **Step of repeated instructions or** an **algorithm.**
- **FOR EXAMPLE,**
- The width is 3.0 and length is 1.0 for a ladybird. If we
- tested the **y = Ax** function with this example where **x** is 3.0,
- `y = (0.25) * (3.0) = 0.75`
- The parameter **A = 0.25, (**initial randomly chosen)
- So for a bug of width 3.0, the length should be 0.75.
- **that's too small**
- As the training data example tells us it must be a length of 1.0
- There is an **error (**this error is used to inform how to adjust the parameter **A** .**)**   BACK



- This line seems to be a predictor of a bug's length given its width
- The line is required to be a dividing line between ladybirds
- and caterpillars, not a predictor of a bug's length given

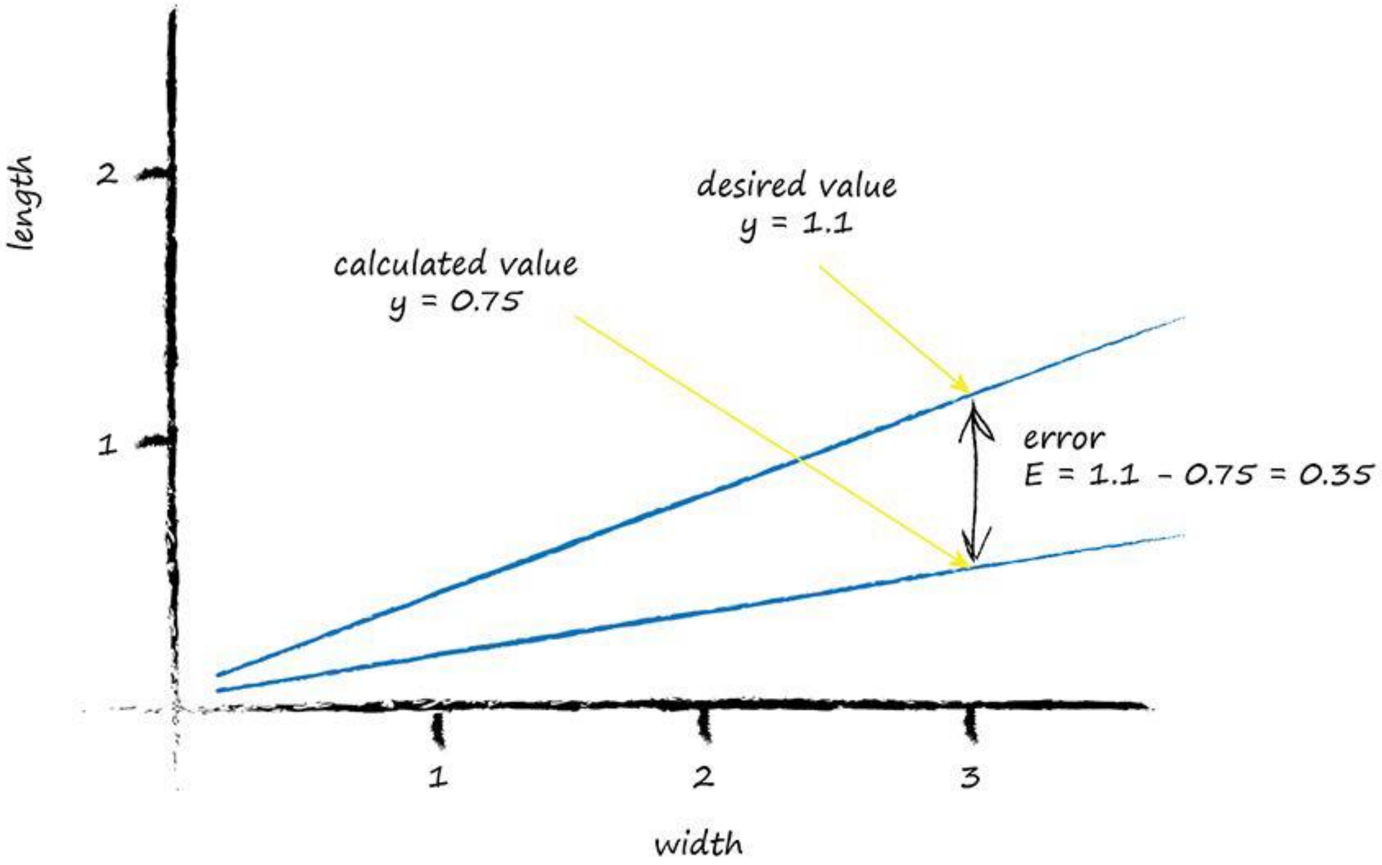
its width.

Another Example y = 1.1 when **x** = 3.0

Error = (desired target – actual output)

E = 1.1 - 0.75 = 0.35

Visual Representation of
`Error = (desired target - actual output)`

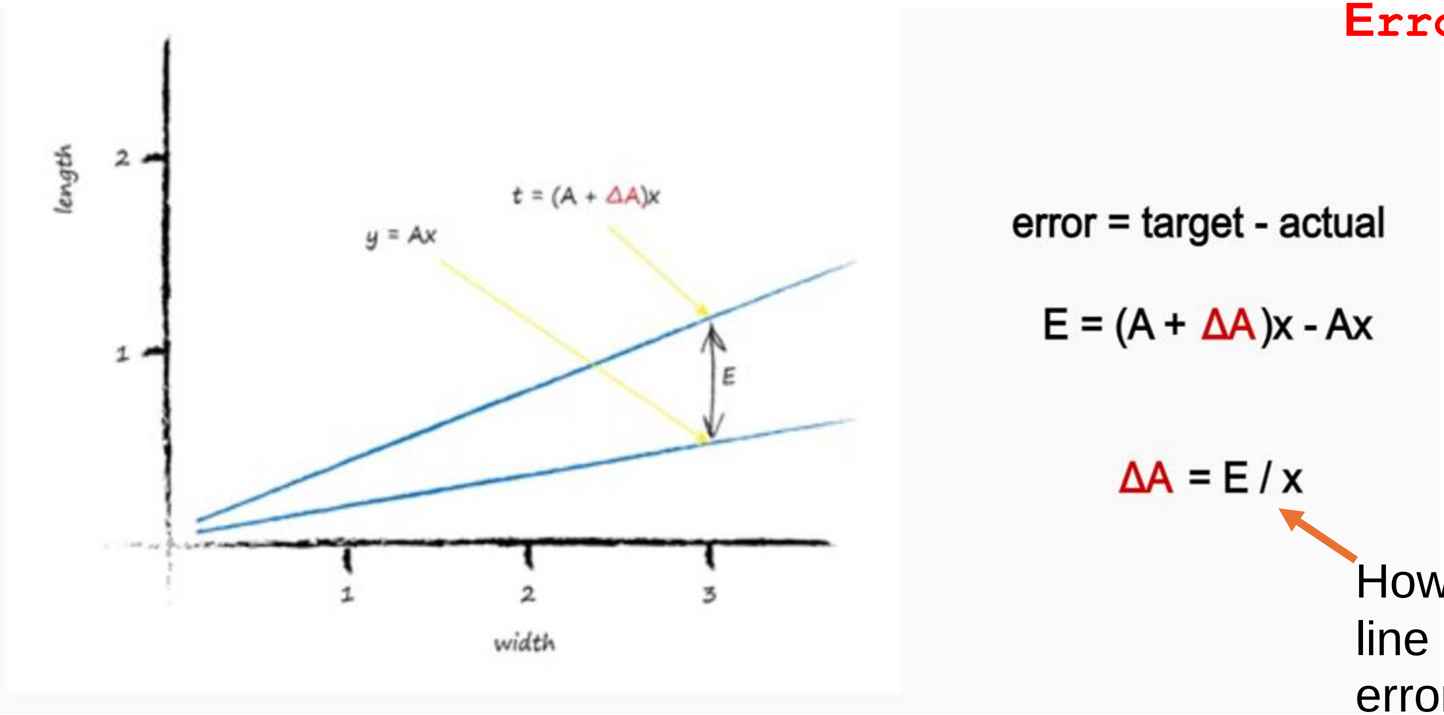


Updating Parameter

- **E** to guide us to a better refined parameter **A but how?**
- the linear function for the classifier: **Y = Ax**

Consider correct desired value for target is t = (A+ΔA)x

Where Δ means small change or delta, i.e. required for adjusting A



Error = (desired target - actual output)

$$t - y = (A + \Delta A)x \;\; Ax$$

$$E = t - y = Ax + (\Delta A)x \;\; Ax$$

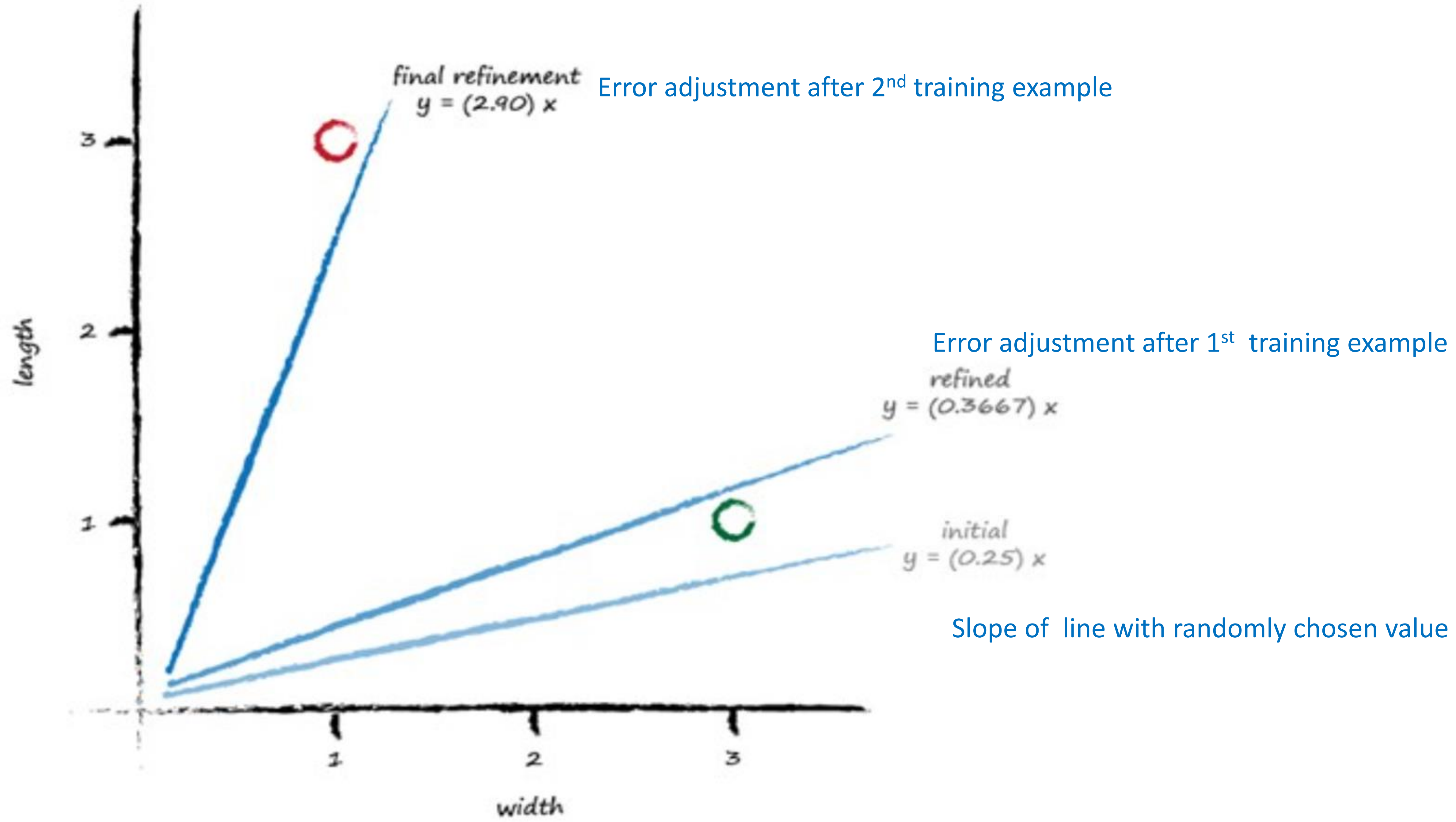
$$E = (\Delta A)x$$

How much to adjust **A** by to improve the slope of the line so it is a better classifier, being informed by the error **E** .

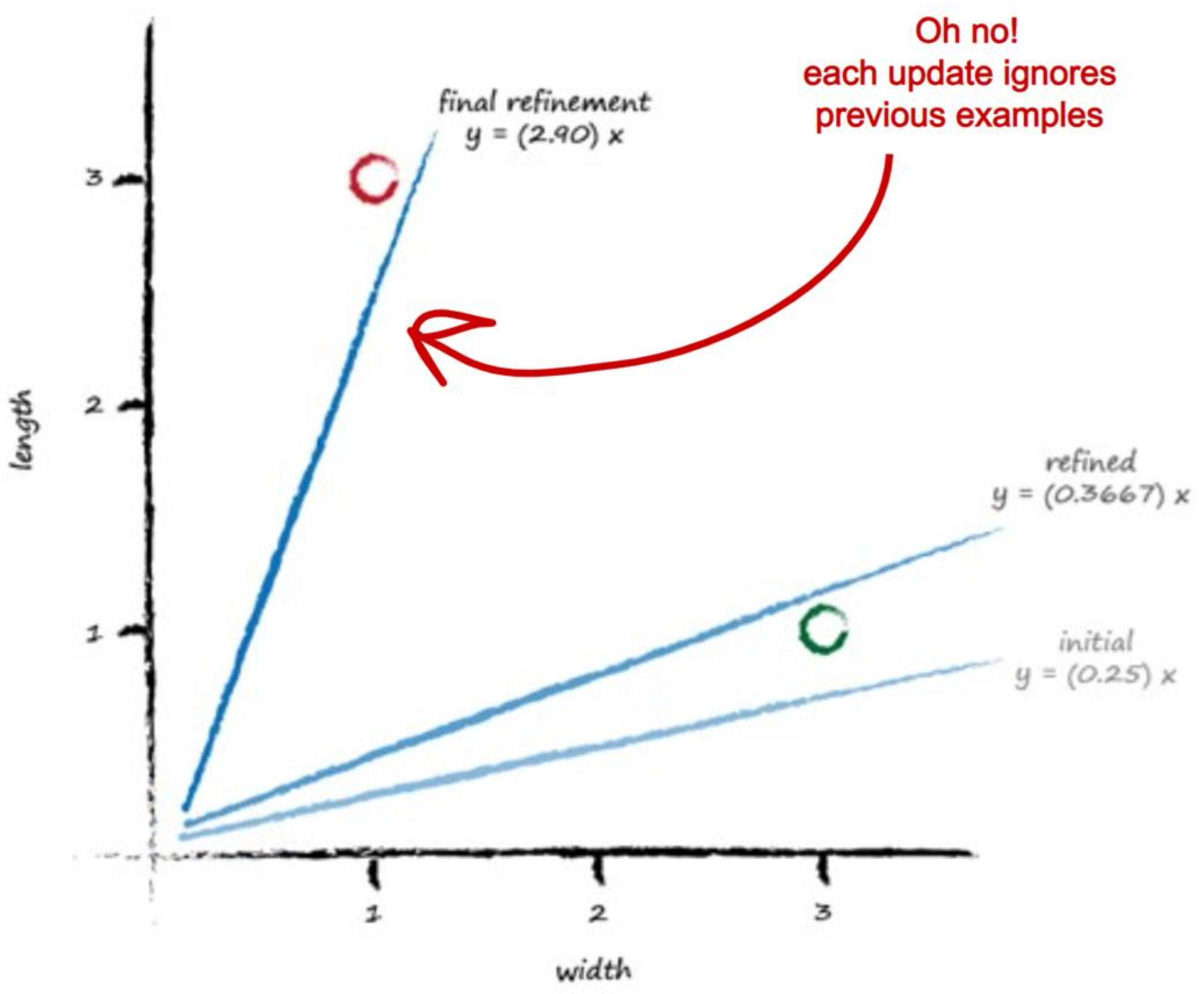
Therefore, visiting our example Back .

Updating Parameter

- The error was 0.35 and the **x** was 3.0.
- That gives Δ **A** = **E** / **x** as 0.35 / 3.0 = 0.1167.
- So, to change the current **A** = 0.25 by 0.1167. That means the new improved value for **A** is
- (**A** + Δ **A**) which is 0.25 + 0.1167 = 0.3667.
- Finally, the calculated value of **y** with this new **A** is 1.1
- **Another Example # 2 caterpillar**
- **x =1.0 , y = 3.0** and we have updated value (**A** + Δ **A**) = 0.3667, so **y** = 0.3667 * 1.0 = 0.3667. but desired target is **y = 3.0**
- **Let's set** desired target value at 2.9. The error **E** is (2.9 0.3667) = 2.5333.
- The Δ **A** is **E** / **x** which is 2.5333 / 1.0 = 2.5333. The newer **A** is 0.3667 + 2.5333 = 2.9



If Updating for each training data example, the final update simply matches the last training example closely



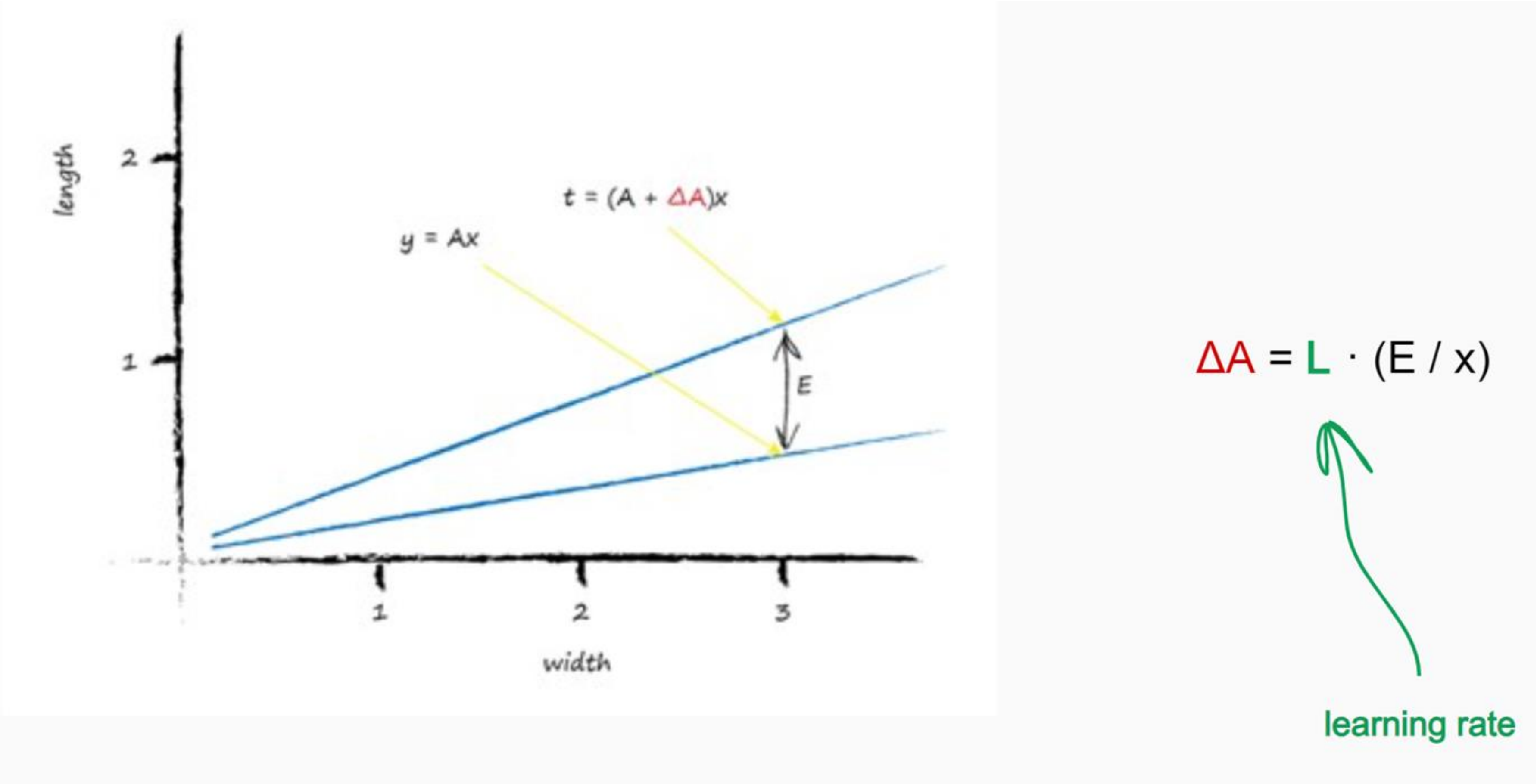
So, to keep in context all learning example, We do not take every change in A(Δ **A**) as whole but fraction or part of it.
This is term as **moderation** an important concept in ML.
This concept can also be observes in mile to KM example(changing parameter c by small amount).

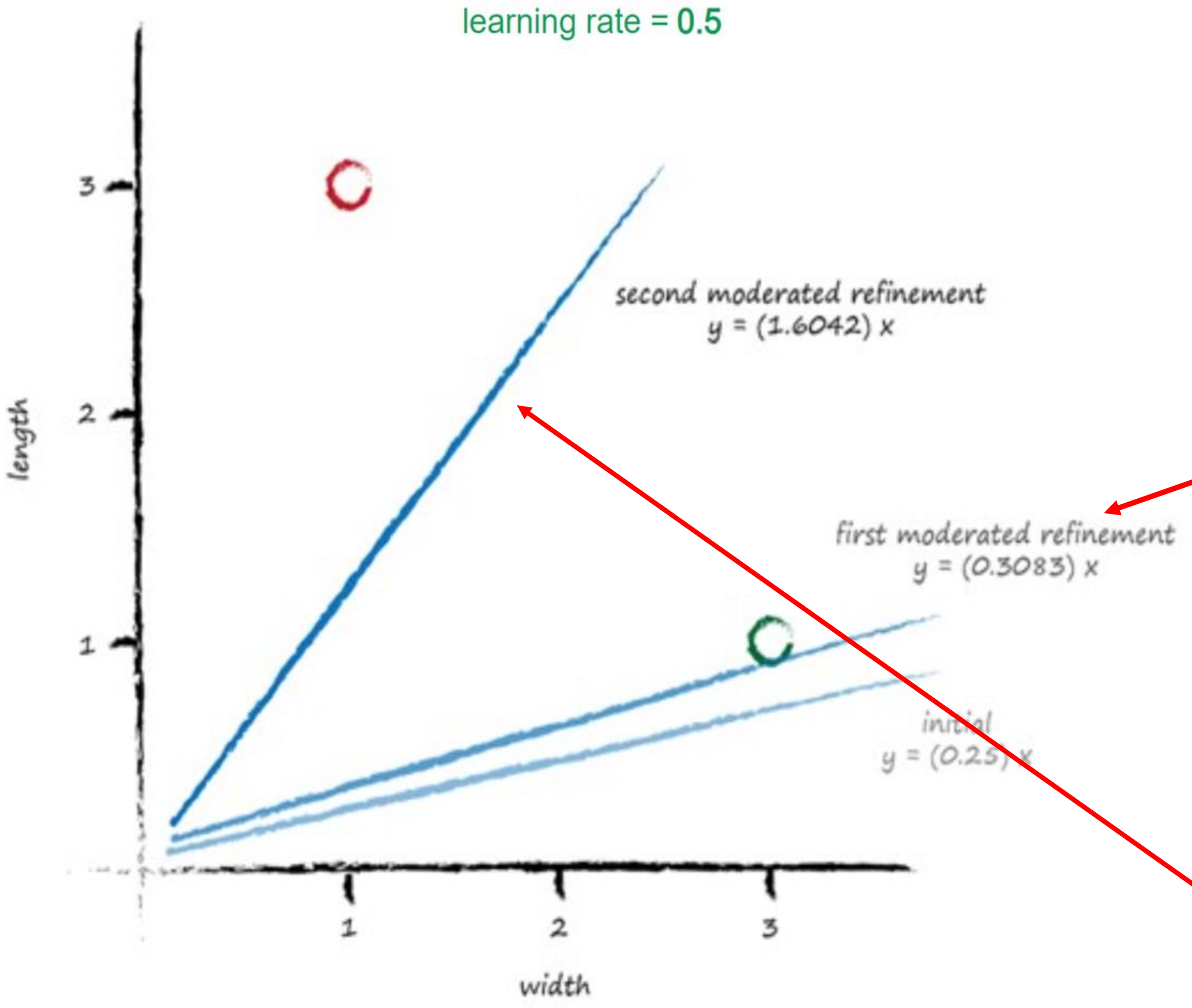
$$\Delta A = L\ (E\ /\ x\ )$$

Where L is moderation or learning rate.

If we take L = 0.5, only update half as much as would have done without moderation.

Calm down or moderate Learning Rate

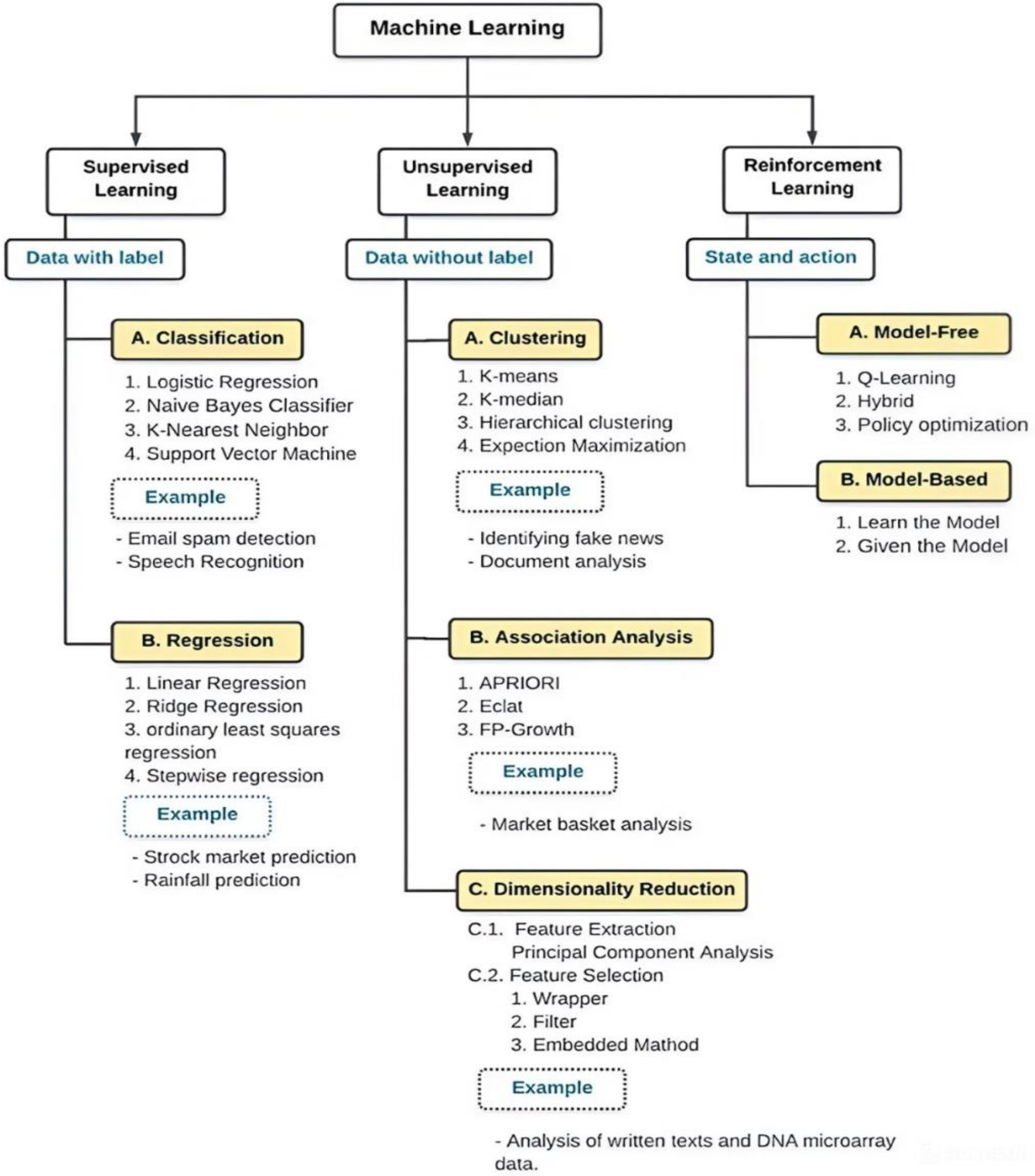




The Learning Rate

- 1) First training : **y** = 0.25 * 3.0 = 0.75. A desired value of 1.1 gives us an error of 0.35.
  - The Δ **A** = **L** (**E** / **x**) = **0.5** * 0.35 / 3.0 = 0.0583.
  - The updated **A** is 0.25 + 0.0583 = 0.3083.
  - **x** = 3.0 gives **y** = 0.3083 * 3.0 = 0.9250(**below 1.1**)
- 2) Second Example:
  - **x** = 1.0. Using **A** = **0.3083** → y = 0.3083 * 1.0 = **0.3083**.
  - The desired value was 2.9 so the error is (2.9 - 0.3083)
  - = **2.5917**.
  - Δ **A** = **L** (**E** / **x**) = 0.5 * 2.5917 / 1.0 = 1.2958. The even newer **A** is now 0.3083 + 1.2958 = 1.6042.

- The dividing line is y=1.6042x, learned through an automated classification method
- Moderating updates limits the impact of noisy or erroneous real-world training examples.


Machine Learning
Supervised Learning
Unsupervised Learning
Reinforcement Learning
Data with label
Data without label
State and action
A. Classification
1. Logistic Regression
2. Naive Bayes Classifier
3. K-Nearest Neighbor
4. Support Vector Machine
Example
- Email spam detection
- Speech Recognition
B. Regression
1. Linear Regression
2. Ridge Regression
3. ordinary least squares regression
4. Stepwise regression
Example
- Strock market prediction
- Rainfall prediction
A. Clustering
1. K-means
2. K-median
3. Hierarchical clustering
4. Expection Maximization
Example
- Identifying fake news
- Document analysis
B. Association Analysis
1. APRIORI
2. Eclat
3. FP-Growth
Example
- Market basket analysis
C. Dimensionality Reduction
C.1. Feature Extraction
Principal Component Analysis
C.2. Feature Selection
1. Wrapper
2. Filter
3. Embedded Mathod
Example
- Analysis of written texts and DNA microarray data.
A. Model-Free
1. Q-Learning
2. Hybrid
3. Policy optimization
B. Model-Based
1. Learn the Model
2. Given the Model