AI Theories

Lesson 2: Machine Learning

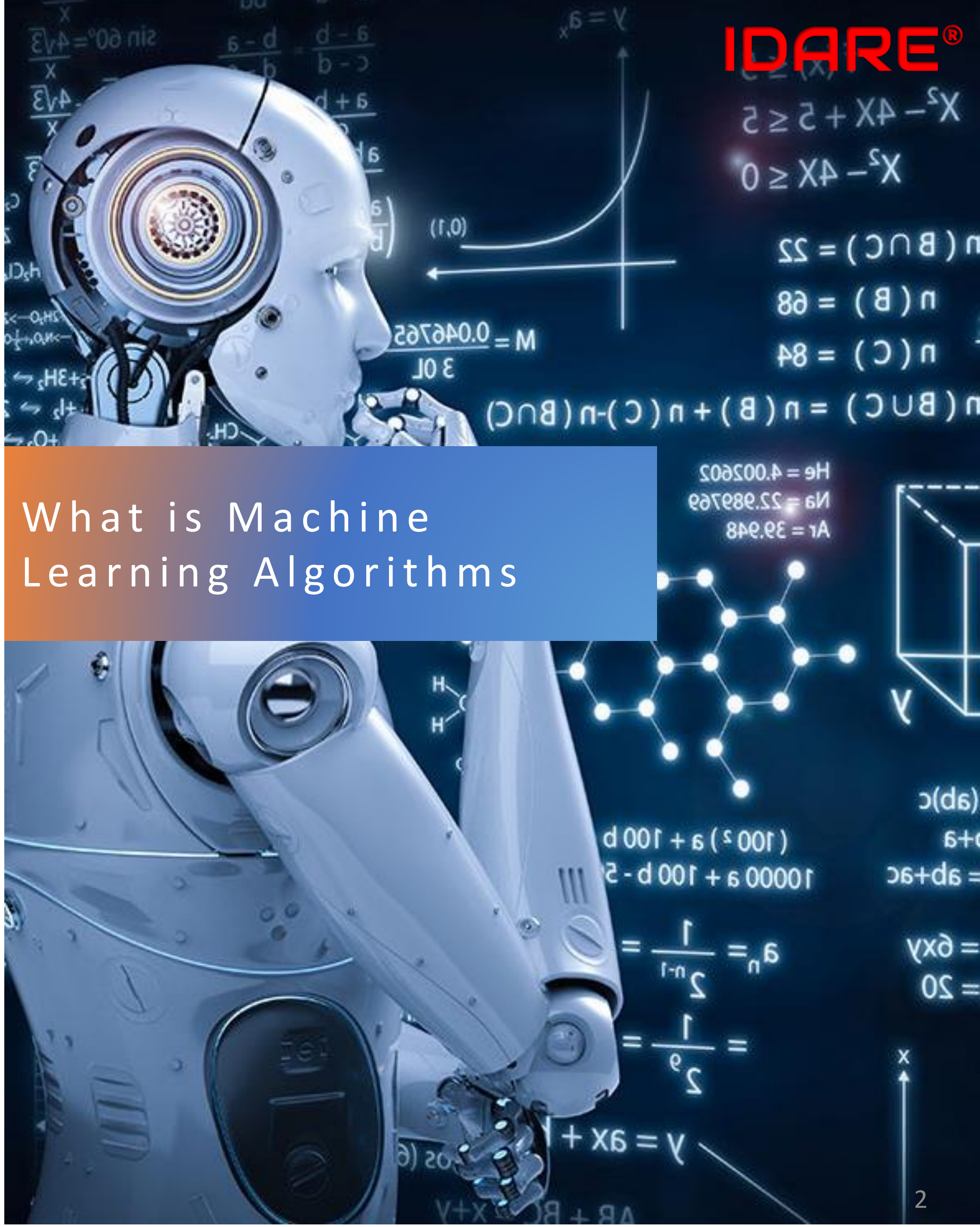
Topic 1 : Basic Concepts of AIML solutions

MACHINE LEARNING

ARE MATHEMATICAL ALGORITHMS THAT HELPS LEARNING FROM HISTORY, EXTRACT KNOWLEDGE THEN EITHER CLASSIFIES OR QUANTIFIES

Machine Learning (**ML**) is a collection of tools and techniques that transforms data into (hopefully good) decisions by making *classifications*, like whether or not someone will love a movie, or *quantitative predictions*, like how tall someone is.

It’s all about those two things. When we use machine learning to make *quantitative predictions*, we call it **Regression**. And when we *classify* things, we call it **Classification**.



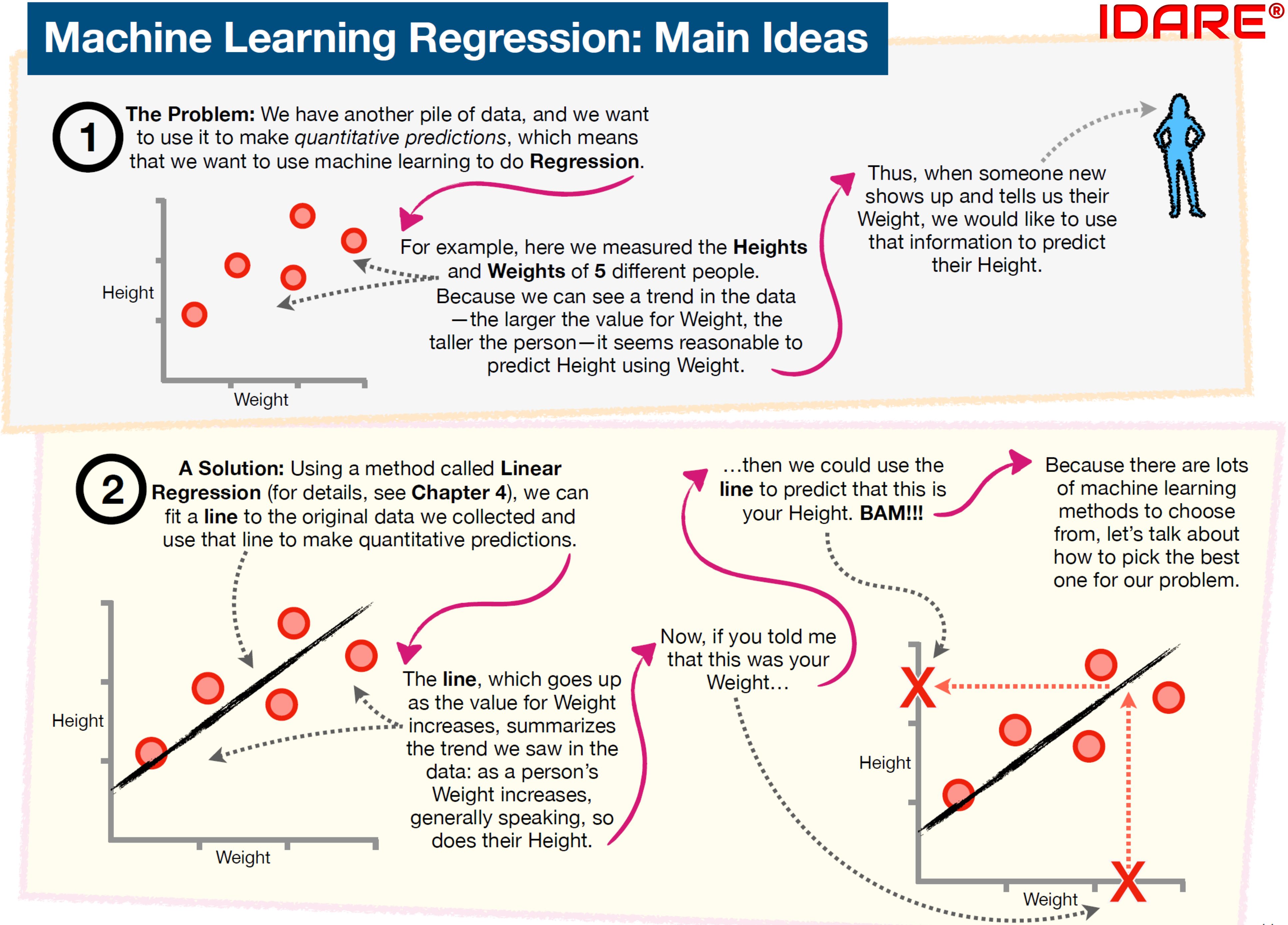
IDARE®

Types of AIML Solutions!!!

Concept of Regression (Quantify)

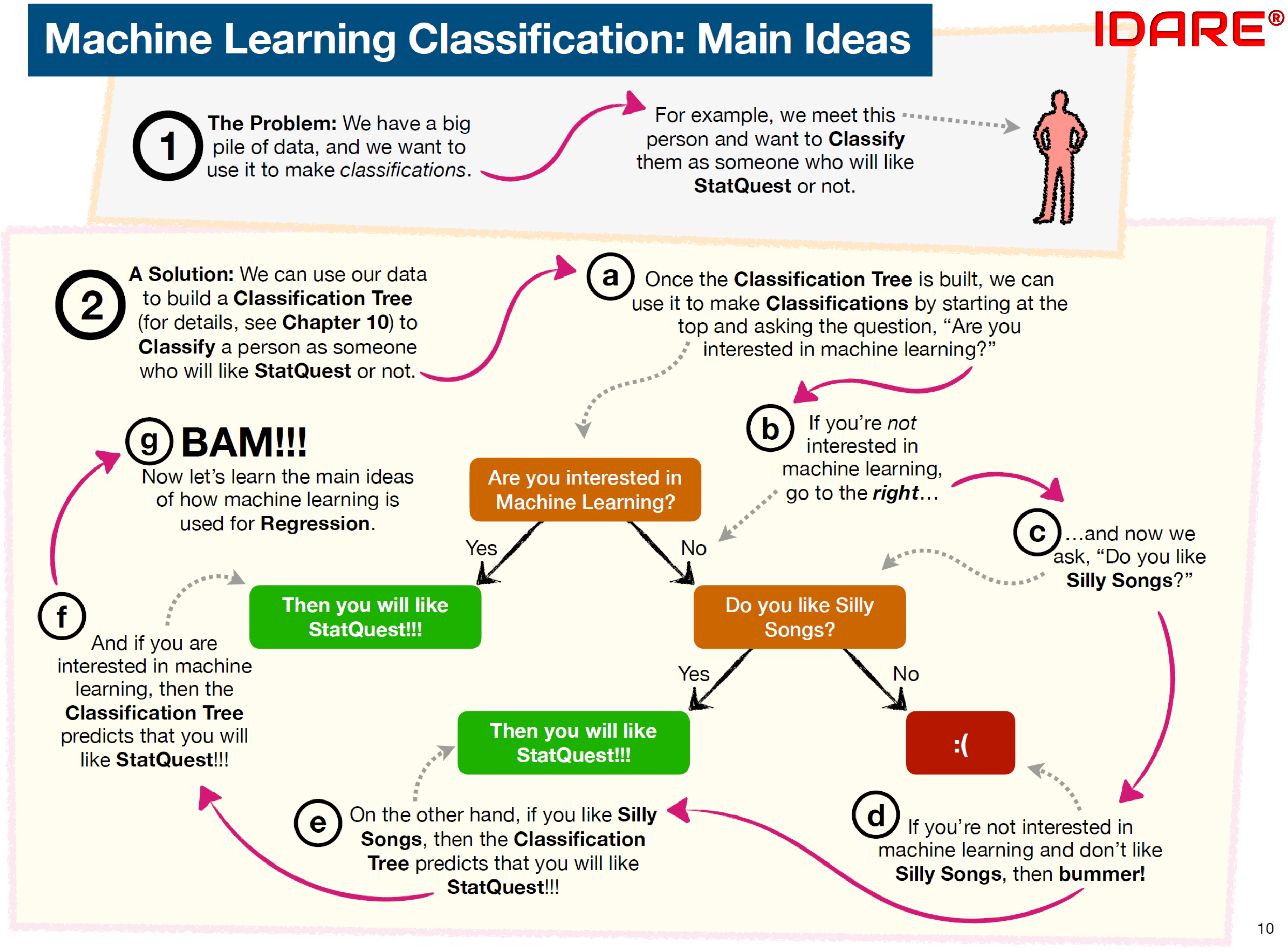
Concept of Classification (Classify)

When we want to **Quantify** to understand the magnitude, we call it **Regression**



IDARE®

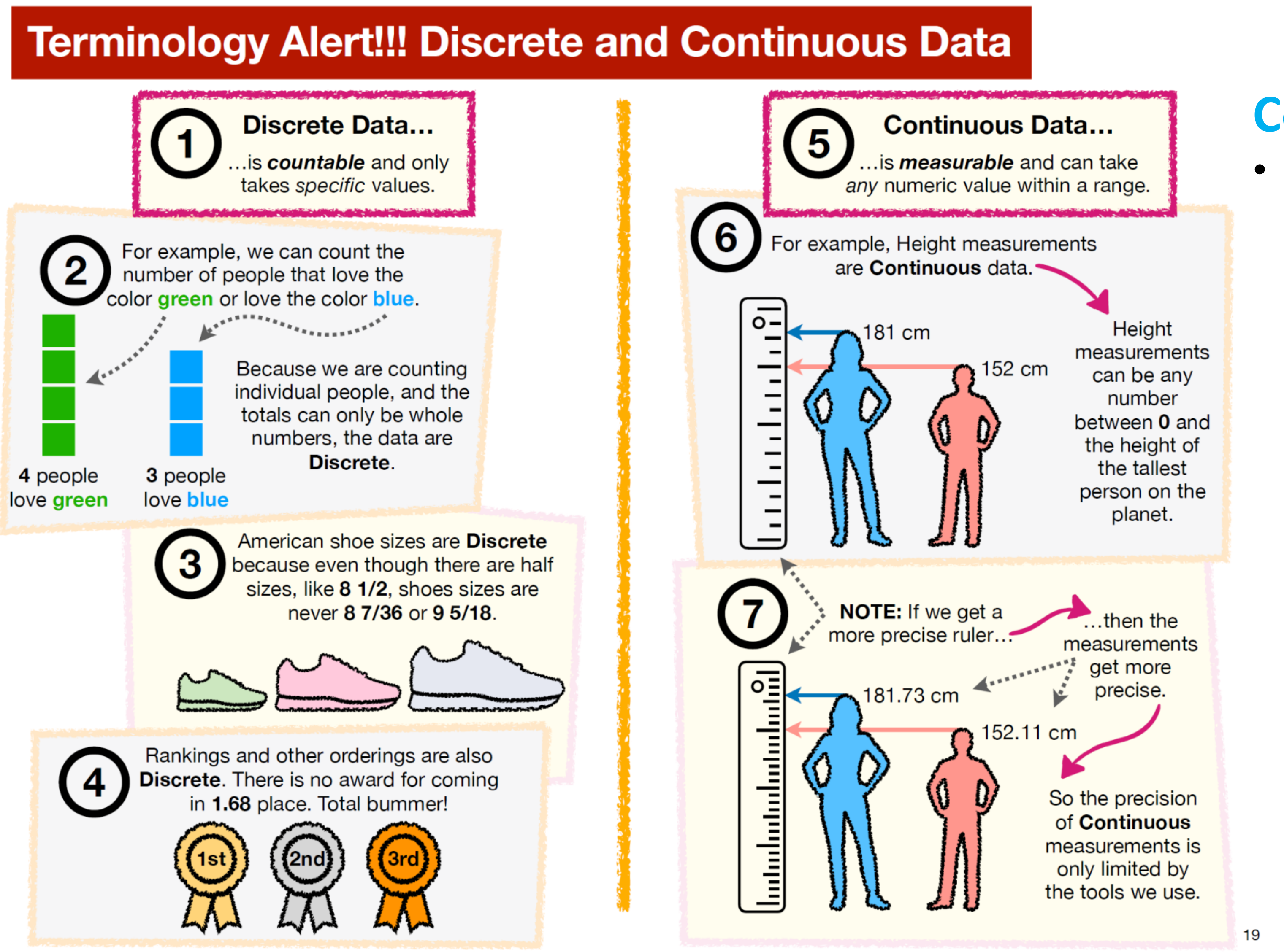
When we want to **Diagnose or Identify** to separate things out, we call it **Classification**



When to use which type?

Discrete Data

- Discrete Number i.e. 10,20 30
- Text or String i.e
  - Red, Blue
  - Tall, Long
  - Fail, No Fail



Continuous Data

- Any number between ranges can be even fractions

Classification

Regression

Types of Algorithm for Classification and Regression!!!

How to Assess the Solution or Compare

Types of Algorithms utilized in AI

Statistical Algorithm

- Designed understands relationship between variables
- Learn From Past and Predict Future
- Serves as the basis of explainability of the solution
- Useful when the problem is well understood

Linear Regression, logistic Reg. , SVR etc.

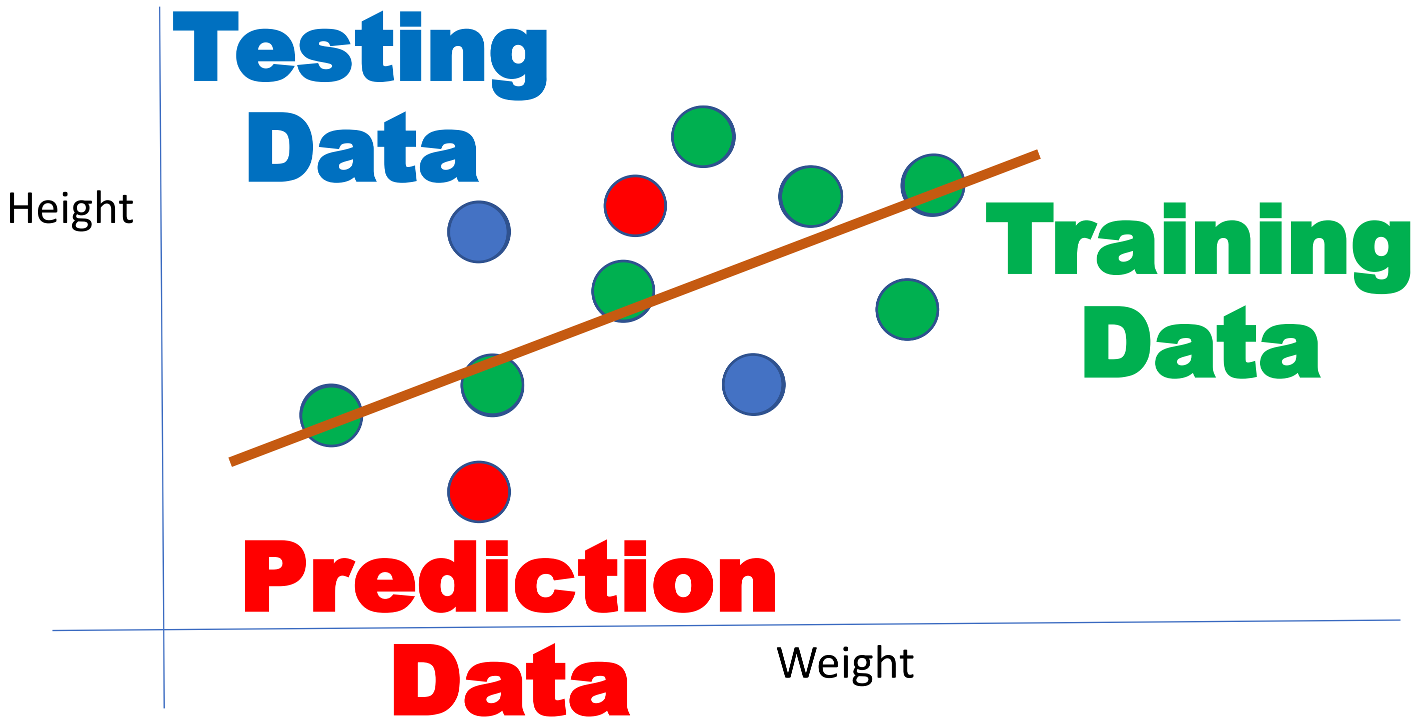
Machine Learning Algorithm

- Designed to provide accurate prediction
- Learn From Past's Mistake and Predict Future
- Useful when the problem is less understood

Decision Tree, Neural Network, Deep Learning, Gradient Boosting etc

How Machine Learning works with Data!!!

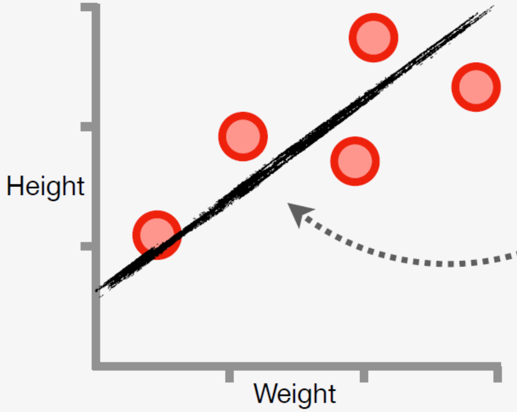
- Machine learning learns from Training Data set, then…
- Validation done on Unseen or Out of Sample Actual data, call Testing Data
- If Satisfied future prediction performs on Prediction Data



How to Verify or Validate AI solution!!!

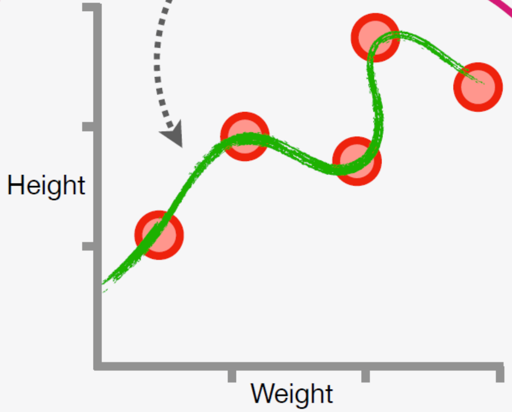
Comparing Machine Learning Methods: Main Ideas

1 **The Problem:** As we'll learn in this book, machine learning consists of a lot of different methods that allow us to make **Classifications** or **Quantitative Predictions**. How do we choose which one to use?



For example, we could use this **black line** to predict Height from Weight…

…or we could use this **green squiggle** to predict Height from Weight.

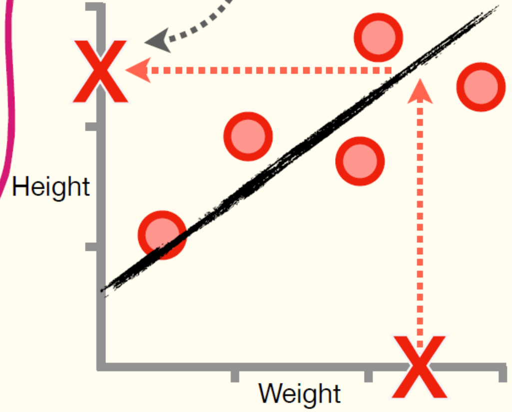


How do we decide to use the **black line** or the **green squiggle**?

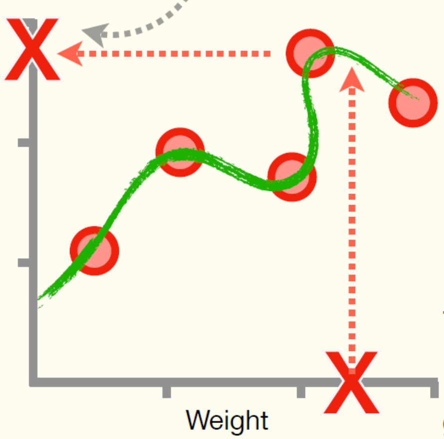
2 **A Solution:** In machine learning, deciding which method to use often means just trying it and seeing how well it performs.

For example, given this person's Weight…

…the **black line** predicts this Height.



In contrast, the **green squiggle** predicts that the person will be *slightly taller*.



We can compare those two predictions to the person's *actual* Height to determine the quality of each prediction.

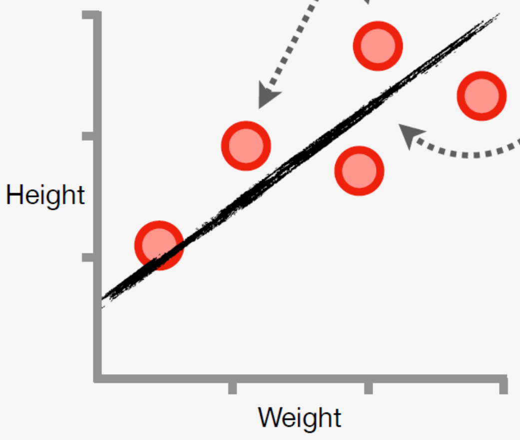
BAM!!!

Now that we understand the **Main Ideas** of how to compare machine learning methods, let's get a better sense of how we do this in practice.

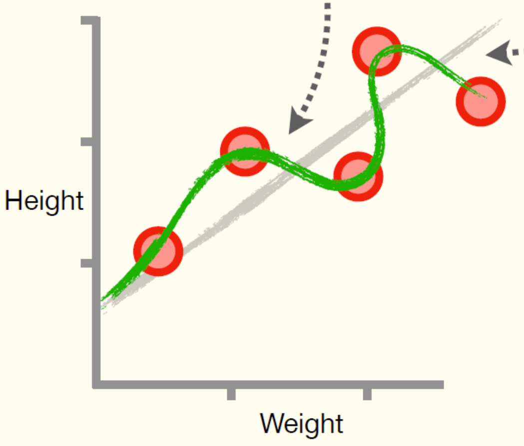
Comparing Machine Learning Methods: Intuition Part 1

1 The original data that we use to observe the trend and fit the **line** is called **Training Data**.

In other words, the **black line** is ***fit*** to the **Training Data**.

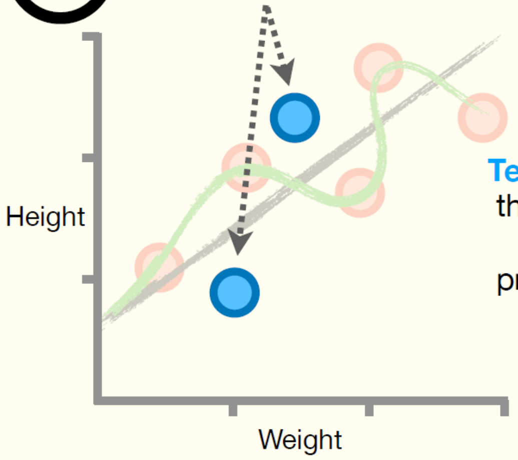


2 Alternatively, we could have fit a **green squiggle** to the **Training Data**.

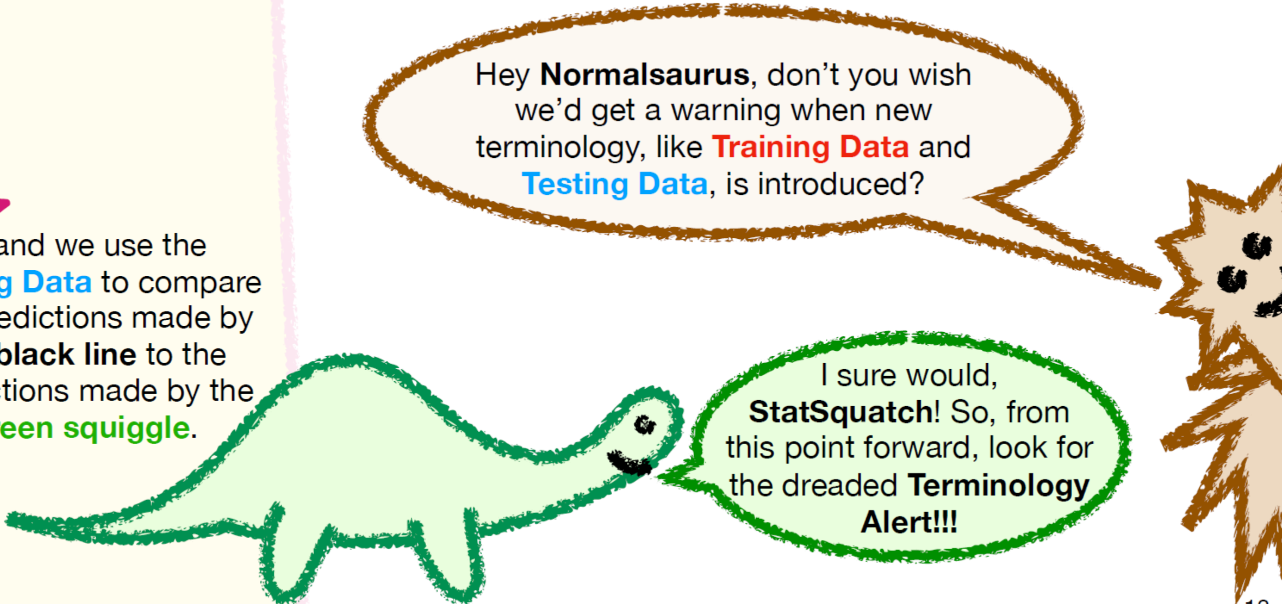


The **green squiggle** fits the **Training Data** better than the **black line**, but remember the goal of machine learning is to make ***predictions***, so we need a way to determine if the **black line** or the **green squiggle** makes better predictions.

3 So, we collect more data, called **Testing Data**…



…and we use the **Testing Data** to compare the predictions made by the **black line** to the predictions made by the **green squiggle**.



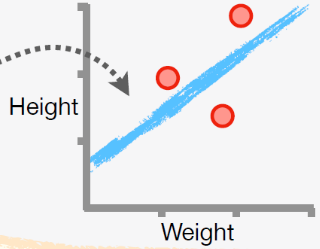
Basic Concepts of Machine Learning!!!

Error Metric

IDARE®

The Sum of the Squared Residuals: Main Ideas Part 1

1 **The Problem:** We have a model that makes predictions. In this case, we're using Weight to predict Height. However, we need to quantify the quality of the model and its predictions.

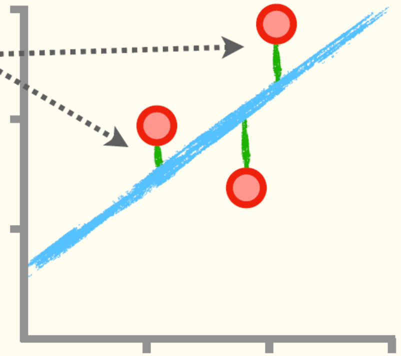


2 **A Solution:** One way quantify the quality of a model and its predictions is to calculate the **Sum of the Squared Residuals**.

As the name implies, we start by calculating **Residuals**, the differences between the **Observed** values and the values **Predicted** by the model.

Residual = Observed - Predicted

Visually, we can draw **Residuals** with these green lines.



Since, in general, the smaller the **Residuals**, the better the model fits the data, it's tempting to compare models by comparing the sum of their **Residuals**, but the **Residuals** below the blue line would cancel out the ones above it!!!

So, instead of calculating the sum of the **Residuals**, we square the **Residuals** first and calculate the **Sum of the Squared Residuals (SSR)**.

The Sum of Squared Residuals (SSR) is usually defined with fancy **Sigma** notation and the right-hand side reads: "The sum of all observations of the squared difference between the observed and predicted values."

$$\text{SSR} = \sum_{i=1}^{n} (\text{Observed}_i - \text{Predicted}_i)^2$$

n = the number of **Observations**.

i = the index for each **Observation**. For example, ***i* = 1** refers to the first **Observation**.

The **Sigma** symbol, **Σ**, tells us to do a **summation**.

NOTE: ***Squaring***, as opposed to taking the ***absolute value***, makes it easy to take the derivative, which will come in handy when we do **Gradient Descent** in **Chapter 5**.

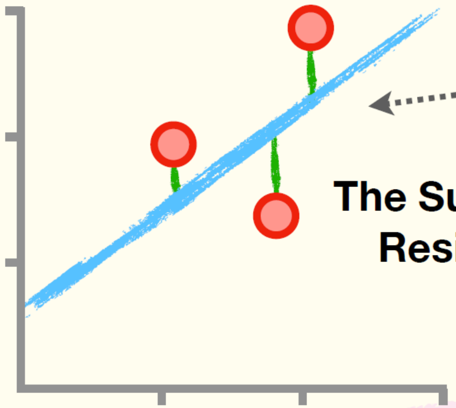
SSR: Step-by-Step

Observed =

Predicted =

Residual =

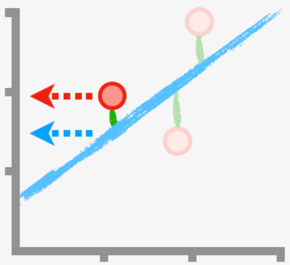
1 In this example, we have **3** Observations, so n **= 3**, and we expand the summation into **3** terms.



The Sum of Squared Residuals (SSR) $= \sum_{i=1}^{n} (\text{Observed}_i - \text{Predicted}_i)^2$

For i **= 1**, the term for the first Observation…

$(1.9 - 1.7)^2$

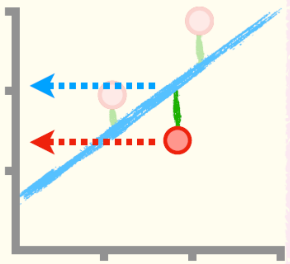


$$\textbf{SSR} = (\text{Observed}_1 - \text{Predicted}_1)^2 + (\text{Observed}_2 - \text{Predicted}_2)^2 + (\text{Observed}_3 - \text{Predicted}_3)^2$$

2 Once we expand the summation, we plug in the **Residuals** for each Observation.

For i **= 2**, the term for the second Observation…

$(1.6 - 2.0)^2$

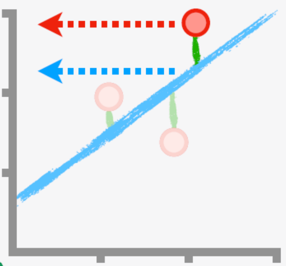


$$\textbf{SSR} = (1.9 - 1.7)^2 + (1.6 - 2.0)^2 + (2.9 - 2.2)^2 = 0.69$$

3 Now, we just do the math, and the final **Sum of Squared Residuals (SSR)** is **0.69**.

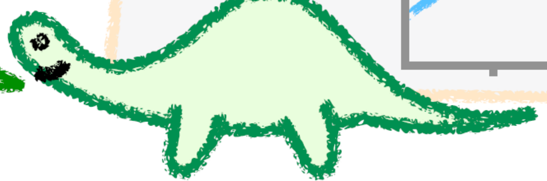
For i **= 3**, the term for the third Observation…

$(2.9 - 2.2)^2$



BAM!!!

Don't get me wrong, the **SSR** is awesome, but it has a pretty big problem that we'll talk about on the next page.



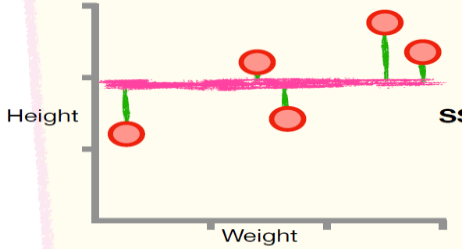
IDARE®

2 **A Solution:** One way to compare the two models that may be fit to different-sized datasets is to calculate the **Mean Squared Error** (**MSE**), which is simply the average of the **SSR**.

$$\textbf{Mean Squared Error (MSE)} = \frac{\textbf{The Sum of Squared Residuals (SSR)}}{\textbf{Number of Observations, } n} = \frac{\sum_{i=1}^{n}(\text{Observed}_i - \text{Predicted}_i)^2}{n}$$

R^2: Details Part 1

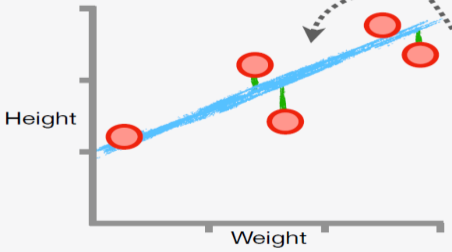
1 First, we calculate the **Sum of the Squared Residuals** for the **mean**. We'll call this **SSR** the **SSR(mean)**. In this example, the mean Height is **1.9** and the **SSR(mean) = 1.6**.



$$\textbf{SSR(mean)} = (2.3 - \textbf{1.9})^2 + (1.2 - \textbf{1.9})^2 + (2.7 - \textbf{1.9})^2 + (1.4 - \textbf{1.9})^2 + (2.2 - \textbf{1.9})^2$$

$$\boxed{= 1.6}$$

2 Then, we calculate the **SSR** for the **fitted line**, **SSR(fitted line)**, and get **0.5**.



NOTE: The smaller **Residuals** around the **fitted line**, and thus the smaller **SSR** given the same dataset, suggest the **fitted line** does a better job making predictions than the **mean**.

$$\textbf{SSR(fitted line)} = (1.2 - \textbf{1.1})^2 + (2.2 - \textbf{1.8})^2 + (1.4 - \textbf{1.9})^2 + (2.7 - \textbf{2.4})^2 + (2.3 - \textbf{2.5})^2$$

$$\boxed{= 0.5}$$

3 Now we can calculate the R^2 value using a surprisingly simple formula...

$$R^2 = \frac{\text{SSR(mean)} - \text{SSR(fitted line)}}{\text{SSR(mean)}}$$

$$= \frac{1.6 - 0.5}{1.6}$$

$$\boxed{= 0.7}$$

...and the result, **0.7**, tells us that there was a **70%** reduction in the size of the **Residuals** between the **mean** and the **fitted line**.

4 In general, because the numerator for R^2...

SSR(mean) - SSR(fitted line)

...is the amount by which the **SSRs** shrank when we fitted the line, R^2 values tell us the percentage the **Residuals** around the **mean** shrank when we used the **fitted line**.

When **SSR(mean) = SSR(fitted line)**, then both models' predictions are equally good (or equally bad), and R^2 **= 0**

$$\frac{\text{SSR(mean)} - \text{SSR(fitted line)}}{\text{SSR(mean)}} = \frac{0}{\text{SSR(mean)}} = 0$$

When **SSR(fitted line) = 0**, meaning that the **fitted line** fits the data perfectly, then R^2 **= 1**.

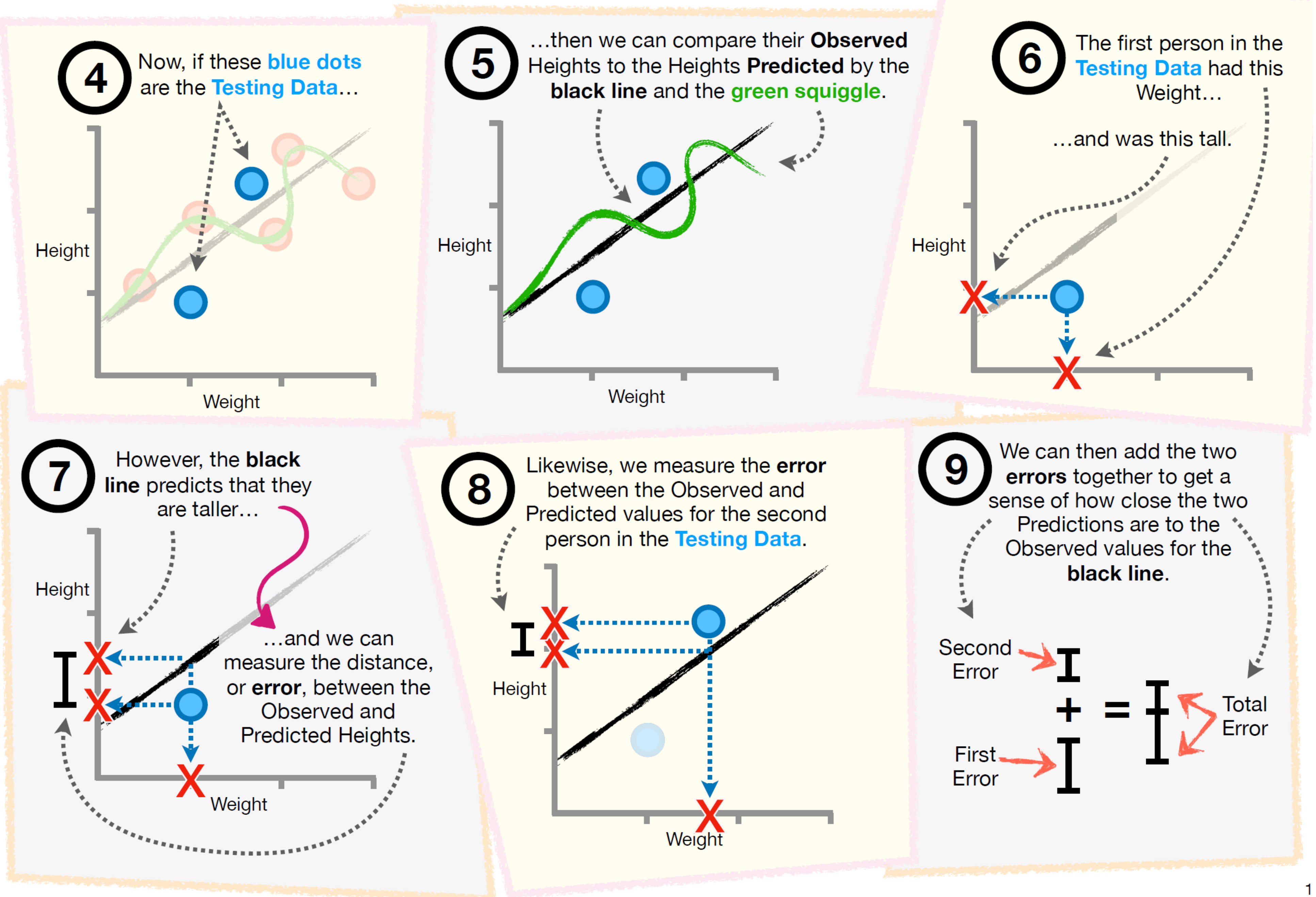
$$\frac{\text{SSR(mean)} - 0}{\text{SSR(mean)}} = \frac{\text{SSR(mean)}}{\text{SSR(mean)}} = 1$$

Basic Concepts of Machine Learning!!!

Assessing with multiple options

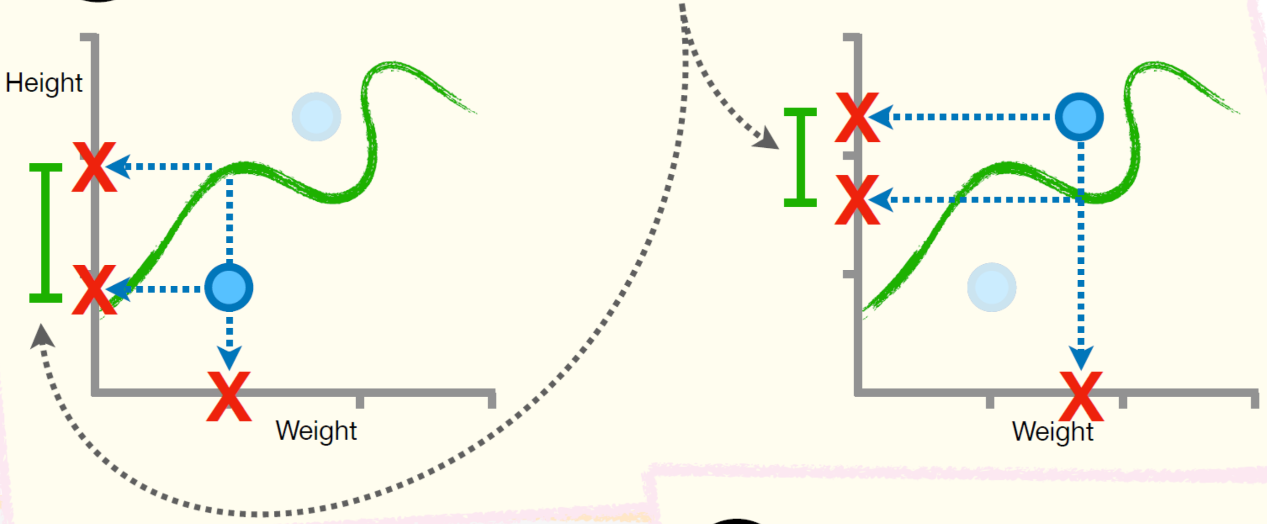
Comparing Machine Learning Methods: Intuition Part 2

IDARE®

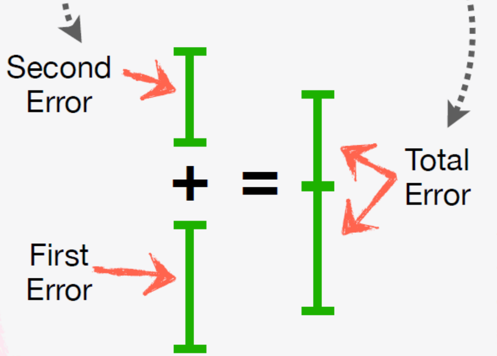


Comparing Machine Learning Methods: Intuition Part 3

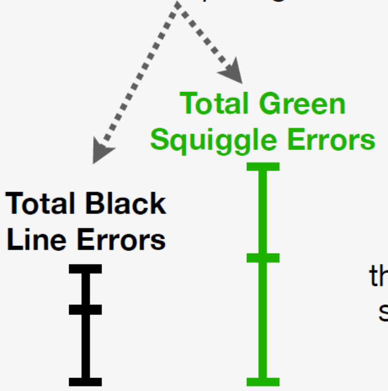
10 Likewise, we can measure the distances, or **errors**, between the Observed Heights and the Heights Predicted by the **green squiggle**.



11 We can then add the two errors together to get a sense of how close the Predictions are to the Observed values for the **green squiggle**.

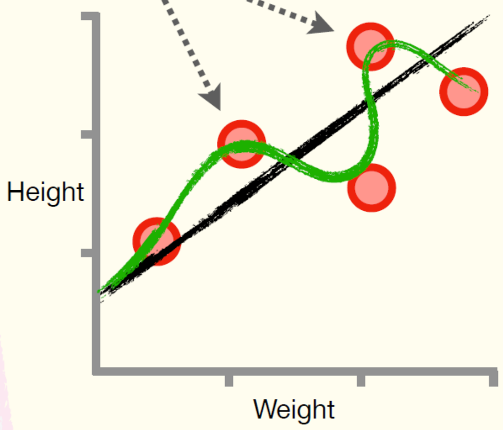


12 Now we can compare the predictions made by the **black line** to the predictions made by the **green squiggle** by comparing the sums of the **errors**.



And we see that the sum of the **errors** for the **black line** is shorter, suggesting that it did a better job making predictions.

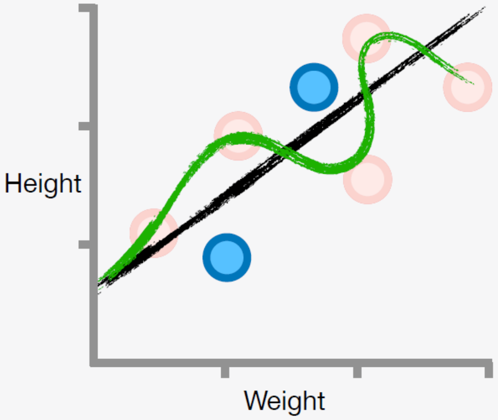
13 In other words, even though the **green squiggle** fit the **Training Data** way better than the **black line**…



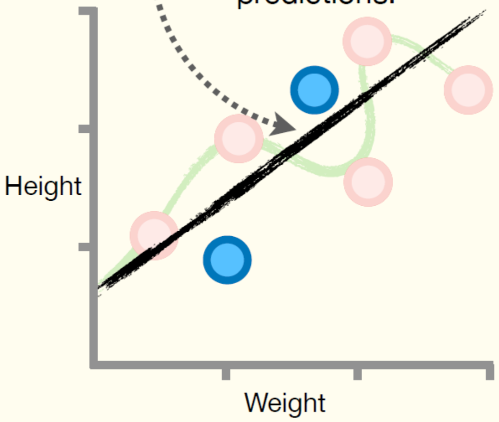
14 …the **black line** did a better job predicting **Height** with the **Testing Data**.

Comparing Machine Learning Methods: Intuition Part 4

15 So, if we had to choose between using the **black line** or the **green squiggle** to make predictions…

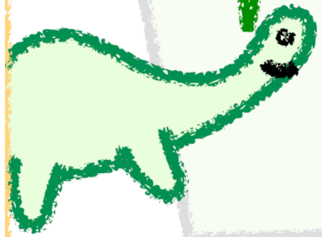


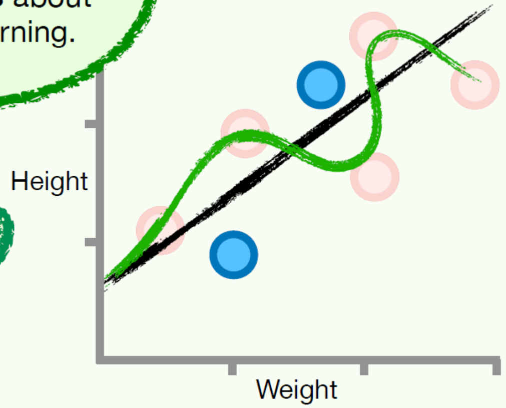
16 …we would choose the **black line** because it makes better predictions.



BAM!!!

The example we just went through illustrates **2 Main Ideas** about machine learning.





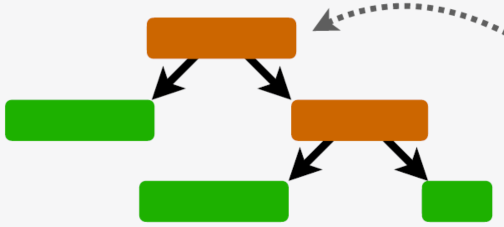
First, we use **Testing Data** to evaluate machine learning methods.

Second, just because a machine learning method fits the **Training Data** well, it doesn't mean it will perform well with the **Testing Data**.

TERMINOLOGY ALERT!!!

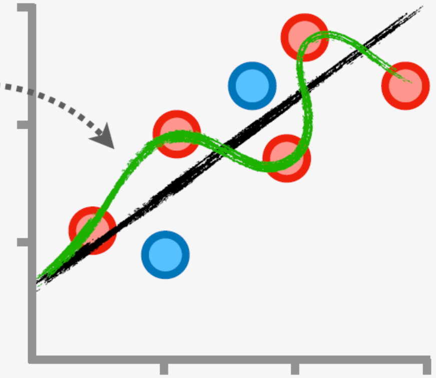
When a machine learning method fits the **Training Data** *really well* but makes *poor predictions*, we say that it is **Overfit** to the **Training Data**. **Overfitting** a machine learning method is related to something called the **Bias-Variance Tradeoff**, and we'll talk more about that later.

The Main Ideas of Machine Learning: Summary



Now, you may be wondering why we started this book with a super simple **Decision Tree**…

…and a simple **black line** and a silly **green squiggle** instead of a…



…**Deep Learning Convolutional Neural Network** or a **[insert newest, fanciest machine learning method here]**.

There are lots of cool machine learning methods. In this book, we'll learn about…

Regression

Logistic Regression

Naive Bayes

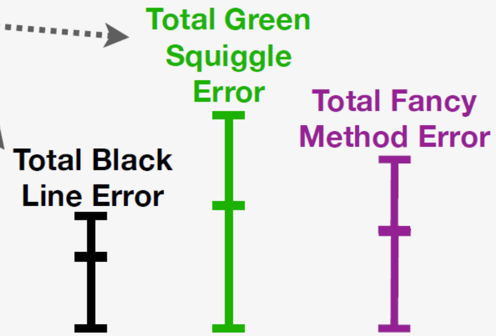
Classification Trees

Regression Trees

Support Vector Machines

Neural Networks

There are tons of fancy-sounding machine learning methods, like **Deep Learning Convolutional Neural Networks**, and each year something new and exciting comes along, but regardless of what you use, the most important thing is how it performs with the Testing Data.



BAM!!!

Now that we understand some of the main ideas of machine learning, let's learn some fancy terminology so we can sound smart when we talk about this stuff at dance parties.

Basic Concepts of Machine Learning!!!

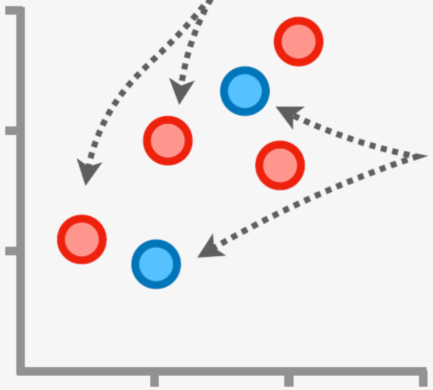
How to Make the Solution Robust and Stable

Cross Validation: Main Ideas

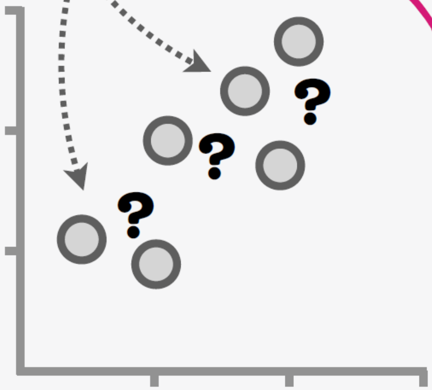
IDARE®

1 **The Problem:** So far, we've simply been told which points are the **Training Data**…

…and which points are the **Testing Data**.



However, usually no one tells us what is for **Training** and what is for **Testing**.



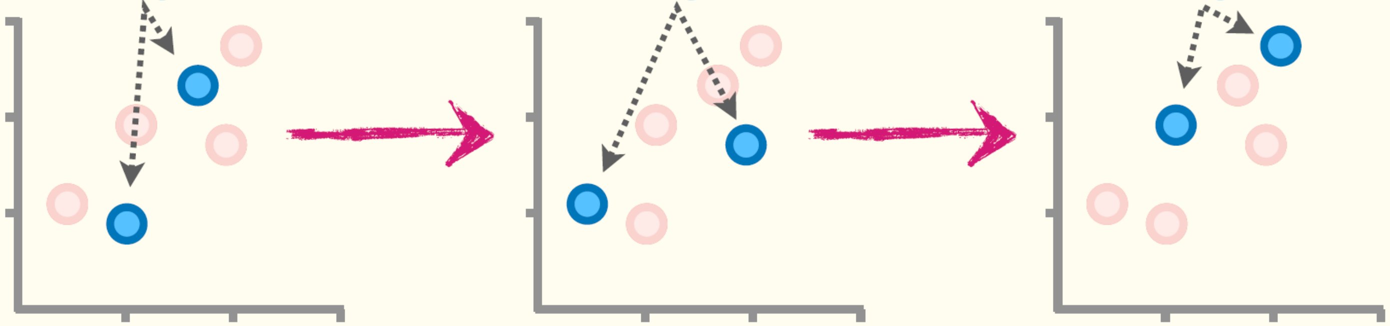
How do we pick the best points for **Training** and the best points for **Testing**?

2 **A Solution:** When we're not told which data should be used for **Training** and for **Testing**, we can use **Cross Validation** to figure out which is which in an unbiased way.

Rather than worry too much about which specific points are best for **Training** and best for **Testing**, **Cross Validation** uses *all* points for both in an *iterative* way, meaning that we use them in steps.

BAM!!!

Testing set **#1** | Testing set **#2** | Testing set **#3**

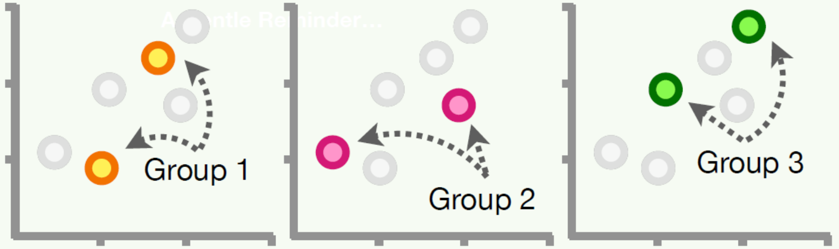


Cross Validation: Details Part 3

IDARE®

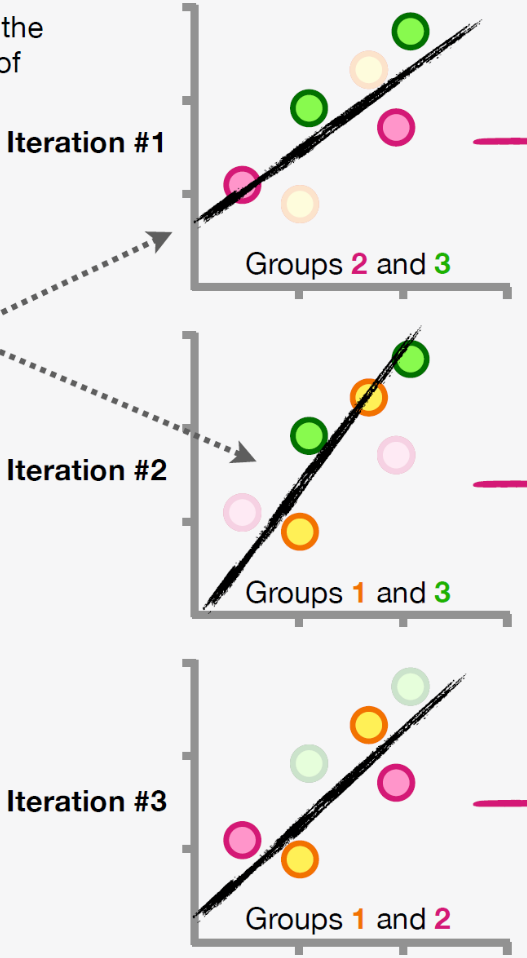
8 Because we have **3** groups of data points, we'll do **3** iterations, which ensures that each group is used for Testing. The number of iterations are also called **Folds**, so this is called **3-Fold Cross Validation**.

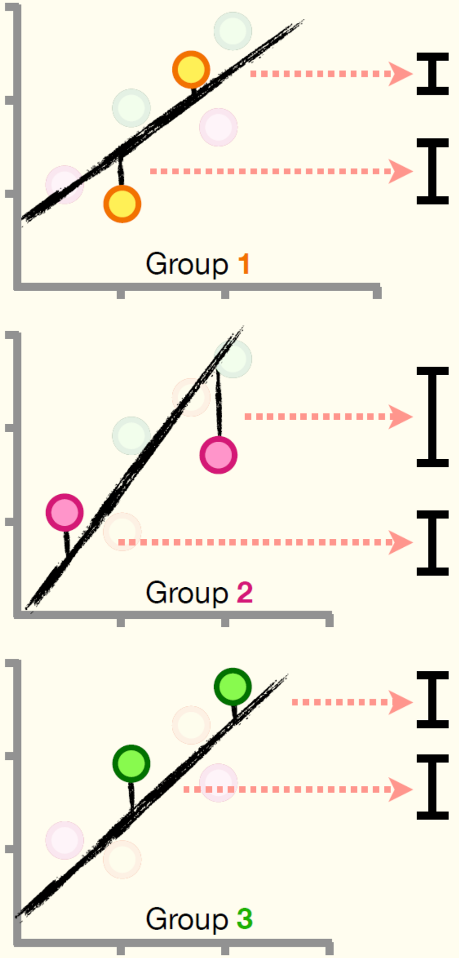
Gentle Reminder: These are the original **3** groups.



9 So, these are the **3** iterations of Training…

NOTE: Because each iteration uses a different combination of data for Training, each iteration results in a slightly different fitted **line**.





10 …and these are the **3** iterations of Testing.

A different fitted line combined with using different data for Testing results in each iteration giving us different prediction errors.

We can average these errors to get a general sense of how well this model will perform with future data…

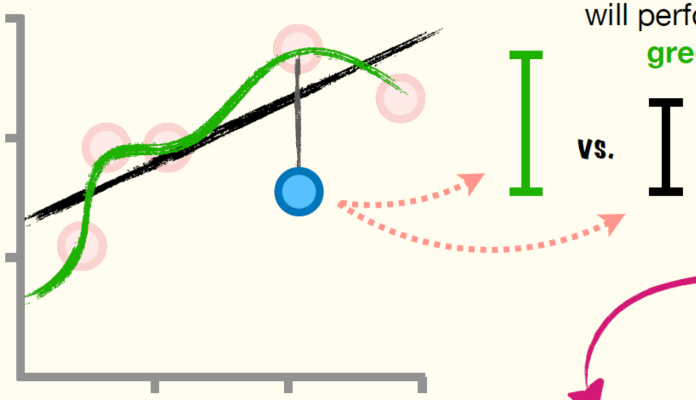
…or we can compare these errors to errors made by another method.

Cross Validation: Details Part 7

16 When we use **Cross Validation** to compare machine learning methods, for example, if we wanted to compare a **black line** to a **green squiggle**…

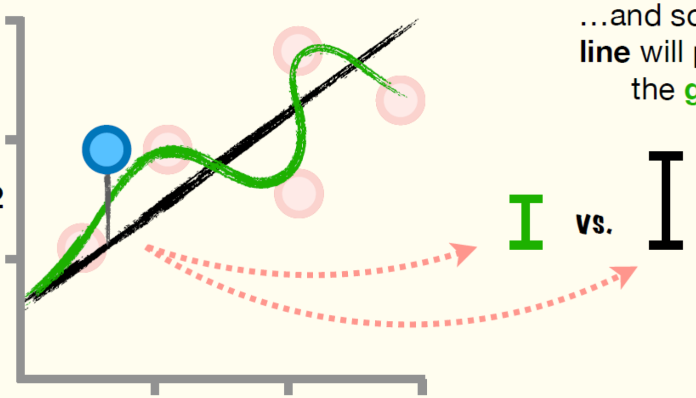
…sometimes the **black line** will perform *better* than the **green squiggle**…

Iteration #1

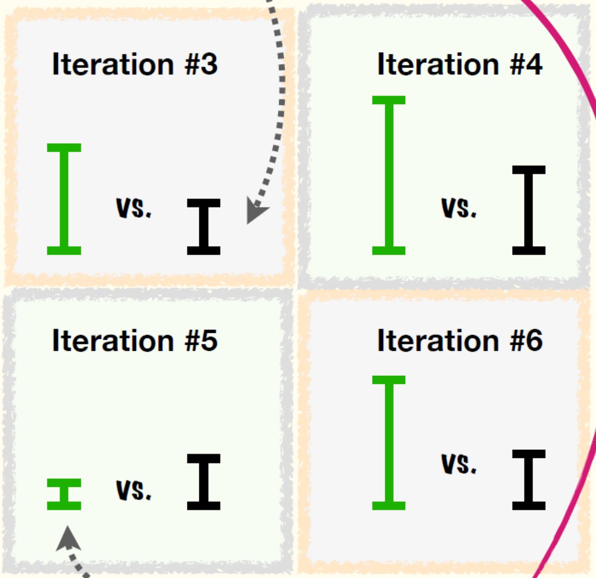


…and sometimes the **black line** will perform *worse* than the **green squiggle**.

Iteration #2



And, after doing all of the iterations, we're left with a variety of results, some showing that the **black line** is better…



…and some showing that the **green squiggle** is better.

When the results are mixed, how do we decide which method, if any, is better? Well, one way to answer that question is to use **Statistics**, and that's what we'll talk about in the next chapter.

TRIPLE BAM!!!

IDARE®