Machine Learning

15-110 – Wednesday 11/18

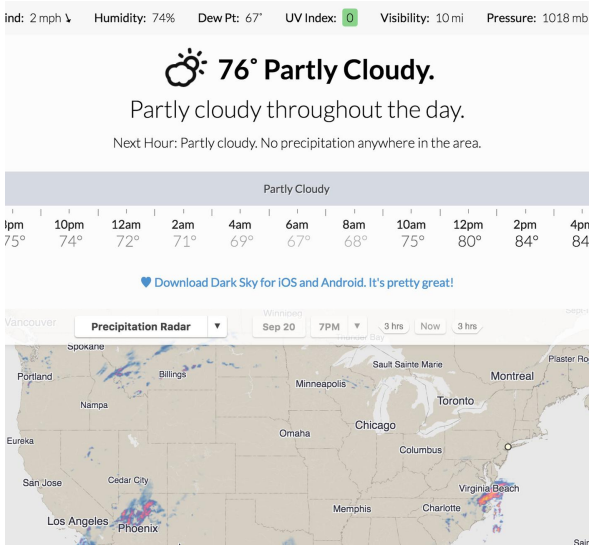
Learning Goals

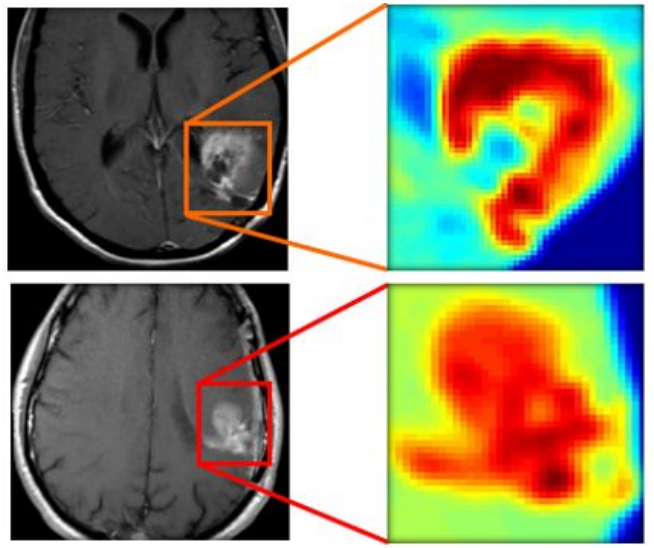
- Identify three major categories of reasoning used with machine learning – **classification, regression,** and **clustering** – and decide which is the best fit for a problem
- Given a dataset, identify **categorical, ordinal,** and **numerical features** which may help predict the correct output for a given input
- Identify how **training data, validation data,** and **testing data** are used in machine learning to produce an accurate reasoner and measure its performance

Machine Learning Overview

Machine Learning Is Used In Many Contexts

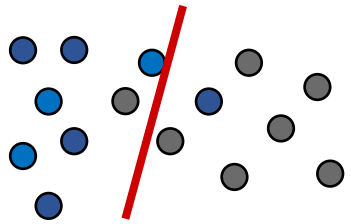
Machine Learning is a process used to **reason** over data and find patterns. It's used in hundreds of contexts across the world, including speech recognition, weather prediction, and medical diagnosis.



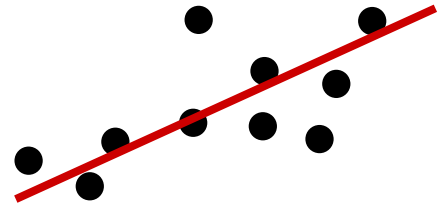


Some Types of Reasoning Associated with ML

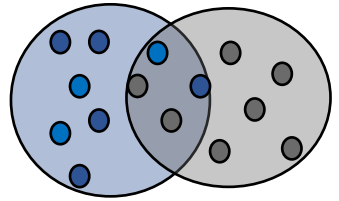
1. Classification
   - Assign an input to one of a fixed set of classes.
   - Examples: “Is this image a dog or a cat?” Or “Is this email spam or not spam?”



2. Regression
   - Predict the numeric value of a function on a novel input.
   - Example: given data about a house, estimate its market value.



3. Clustering
   - Group data points into clusters based on similarity.
   - Example: propose a set of plant species by measuring characteristics of actual plants in a region and grouping similar ones together.



Reasoning Models

A **reasoning model** is an algorithm for performing a reasoning task. There may be many algorithms that could be used for a task.

For example, for classification tasks one could use:

- A decision tree
- A linear discriminator
- A neural network
- A k-nearest neighbor classifier

Reasoning models are **produced** by machine learning algorithms.

Key Concepts of Machine Learning

AI4K12.org sets out four key concepts of machine learning:

1. Machine learning allows a computer to acquire reasoning behaviors **without people explicitly programming** those behaviors.

2. Learning new behaviors is brought about by changes in the parameters of a **reasoning model**, such as a decision tree or a neural network.

Key Concepts of Machine Learning

3. **Large amounts of training data** are required to narrow down the learning algorithm's choices when the reasoning model is capable of a great variety of behaviors.

4. The reasoning model constructed by the machine learning algorithm can be **applied to new data** to solve problems or make decisions.

Main Types of Machine Learning Algorithms

Machine Learning algorithms are divided into three main types. Which type you use depends on the kind of problem you are trying to solve.

Supervised learning: Used when the data is labeled. The goal is to learn to predict the output (label) for unlabeled data by **training** on the labeled data.

Unsupervised learning: Used when the training data is **not** labeled. The goal is to infer the natural structure of the data.

Reinforcement learning: Used for sequential decision problems, where the computer learns from its own experience. Not covered in this lecture.

Supervised Learning: Choose, Train, then Test

To apply supervised learning to a classification or regression problem, you can follow a simple process.

First: **choose** which learning algorithm you want to use and which features you'll train on.

Second: use the algorithm to **train** a reasoning model based on data you provide. The algorithm will 'learn' from the data the same way a student learns by going over worked examples.

Third: **test** the reasoning model on a different set of data. This helps determine how accurate the model actually is.

Training

Training Identifies Key Features

To use machine learning, we must break down a complex data set into a collection of **features**. During the training process, the algorithm identifies which features contribute the most to the underlying pattern.

If you have a table of data, the features would be the **columns**. For example, housing data might have columns for number of rooms, total square footage, size of yard, etc..

You can create and add new features, the same way you would add new columns in data analysis.

Three Types of Features

When we work with simple table data (in data analysis or machine learning), that data often falls into one of three types.

Categorical: Data fall into one of several categories. Those categories are separate and cannot be compared.

Example: style of house (ranch, split-level, two-story, duplex, Victorian, etc.)

Ordinal: Data fall into separate categories, but those categories **can** be compared – they have a specific **order**.

Example: what is the condition of the house? (poor, fair, good, excellent, new)

Numerical: Data are **numbers**. We can perform mathematical operations on it and compare it to other data.

Example: how many square feet does the house have?

Type of Reasoning is Based on Feature Type

To determine what type of reasoning model you need to create to answer a given question, consider the **type** of feature that you need it to produce.

If you need to predict a **categorical** or **ordinal** class, then you need a **classification** model.

If you need to predict a **numerical** value, you need a **regression** model.

If you don't know you're predicting and want to find patterns in the data, you need a **clustering** model.

Example: Is It a Dog?

???

Dog Features:

- Ear type = pointy
- Has Fur = True
- Screen in background = True

Example: Is It a Dog?

Dog Features:

- Ear type = pointy
  - Has Fur = True
  - ~~Screen in background = True~~

Dog Features:

- Ear type = pointy
  - Has Fur = True
  - Nose length < 6in

Example: Is It a Dog?

Dog Features:

- Ear type = pointy
  - Has Fur = True
  - Nose length < 6in

???

Feature Takeaways

It's rare for a machine learning algorithm to identify a single feature that can definitively be used to answer a question.

Usually, the algorithm uses a **combination of several features**, which are weighted based on how well they correlate with the correct answer.

The algorithm needs to learn from a **lot** of examples to get a good sense of what the real pattern is.

Activity: Features for Dog Breeds

You do: say you wanted to make a machine learning algorithm that could identify the breed of a dog based on a set of features. What are some important features you would include?

Try to come up with features of all three types: categorical, ordinal, and numerical.

Demo: Try it yourself!

If the machine learning algorithm has already been implemented, you can train a reasoning model without writing code!

Teachable Machine uses a neural network reasoning model to classify images.

Build an image classifier model here:
https://teachablemachine.withgoogle.com/train/image

A Simple ML Example: Linear Approximator

Given the age of a child in years (x), estimate their height in inches (y).

We will use a linear equation as our reasoning model:

$$y = mx + b$$

This reasoning model has just two parameters: m and b.

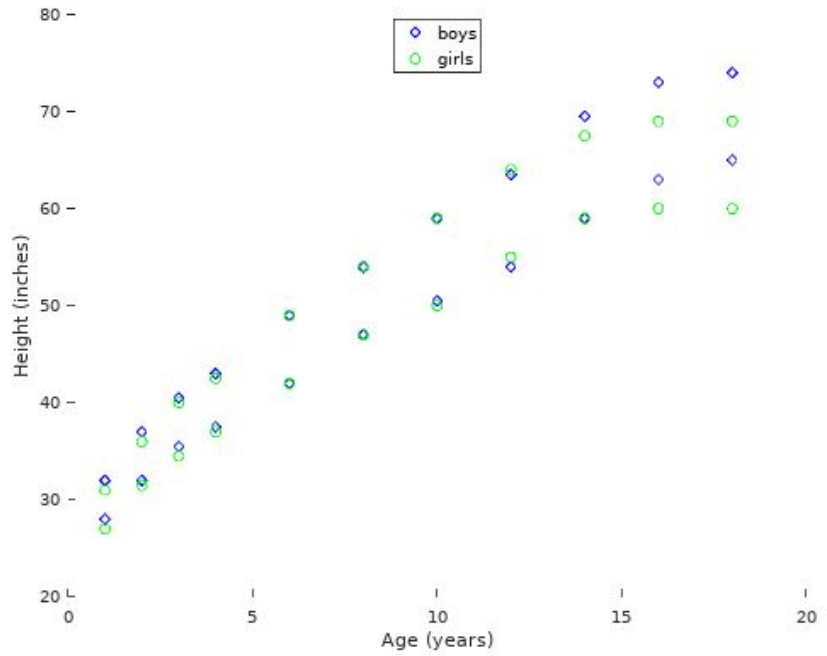
m is the *slope*

b is the *y-intercept*

Training Data

Normal growth data courtesy of CDC and Cincinnati Children's Hospital

https://www.cincinnatichildrens.org/health/g/normal-growth



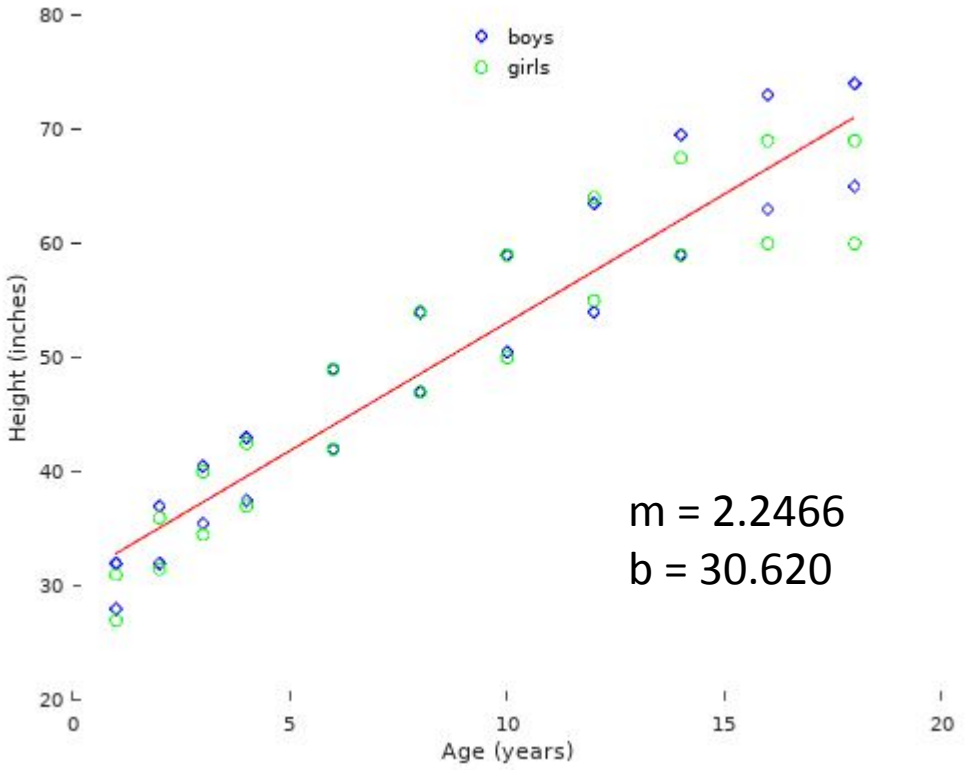
Age	Height Females in Inches	Height Males in Inches
1	27 to 31	28 to 32
2	31.5 to 36	32 to 37
3	34.5 to 40	35.5 to 40.5
4	37 to 42.5	37.5 to 43
6	42 to 49	42 to 49
8	47 to 54	47 to 54
10	50 to 59	50.5 to 59
12	55 to 64	54 to 63.5
14	59 to 67.5	59 to 69.5
16	60 to 68	63 to 73
18	60 to 68.5	65 to 74

Learning Algorithm: Linear Regression

Linear regression has a straightforward formula for estimating m and b from the training data:

$$m = \frac{n\sum xy - (\sum x)(\sum y)}{n\sum x^2 - (\sum x)^2}$$

$$b = \frac{\sum y - m\sum x}{n}$$



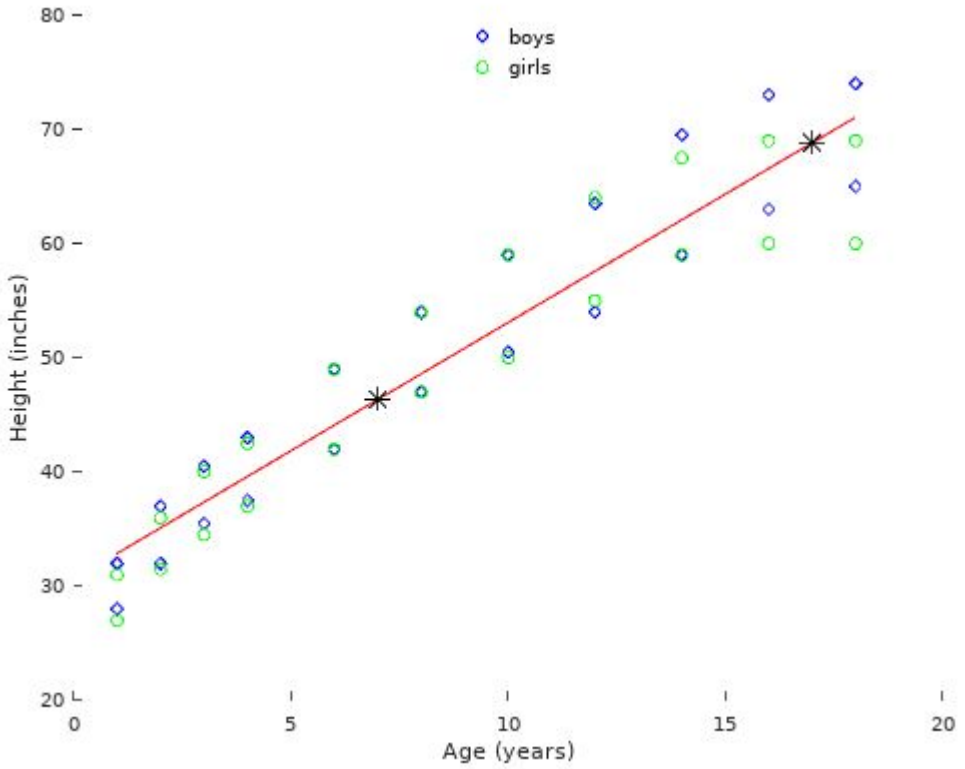
m = 2.2466
b = 30.620

Applying the Trained Reasoning Model

height = 2.2466 × age + 30.620

What is the predicted height of a 7 year old? 46.3 inches

What is the predicted height of a 17 year old? 68.8 inches



Linear Regression Demos: Try It Yourself

Here are several online linear regression demos you can try:

http://www.shodor.org/interactivate/activities/Regression/

https://www.desmos.com/calculator/jwquvmikhr

http://digitalfirst.bfwpub.com/stats_applet/stats_applet_5_correg.html

What If The Data Isn't Linear?

If a line gives a poor fit to the data, you can try a more complex model, such as a quadratic, cubic, quartic, or other type of equation:

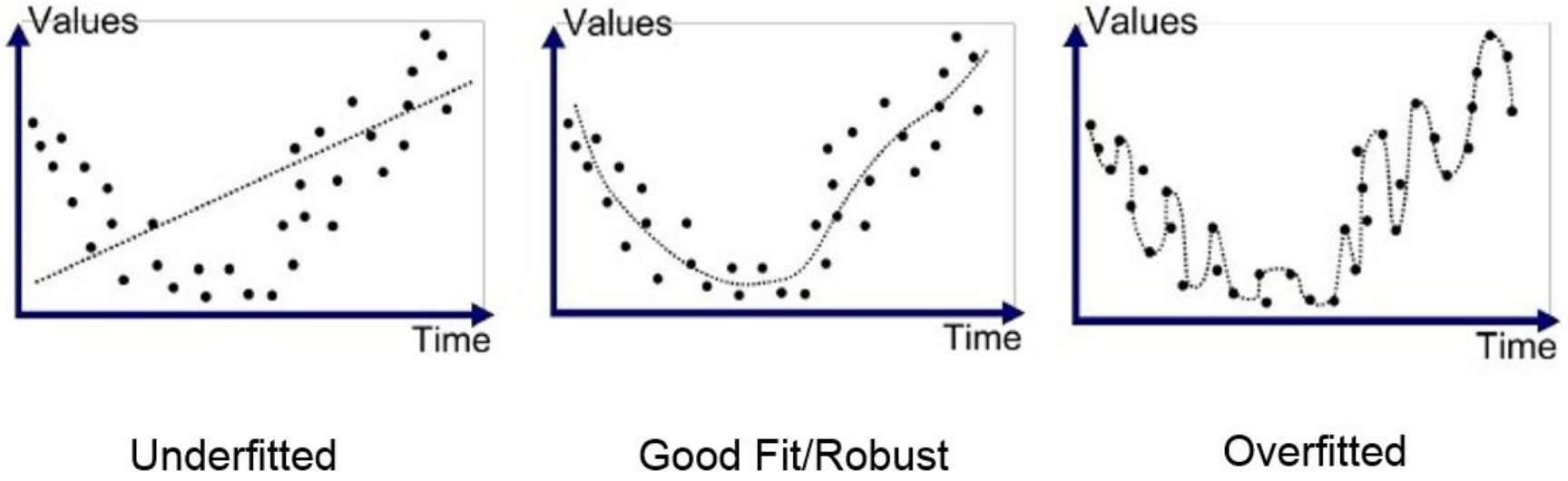
$$y = ax^4 + bx^3 + cx^2 + dx + e$$

There may not be a simple formula for calculating the parameters of a complex model. Instead, we use a learning algorithm called **gradient descent** that adjusts the parameters gradually, in tiny steps, to try to reduce the error. (The error is the difference between the calculated result and the correct result.)

Are Complex Models Better?

Complex models have greater freedom to match the data. But…

- More parameters requires more data, and more work to train.
- Complex models can **overfit** and generate bizarre results:



Validation Data Can Prevent Overfitting

Many machine learning algorithms start out with simple models that become more complex over time as the algorithm tries to eliminate every last bit of error.

We can keep a separate set of labeled data called the **validation set,** not used in training. Then as the model starts becoming too complex, we'll see the validation set error go up due to overfitting even as the training set error continues to go down. When this happens, we should stop learning.

A common technique used in machine learning is **cross-validation**. One dataset is used for both training and validation data; it is split up in a different way on each training run, so that the model is not always evaluated on the same data. This avoids overfitting to the validation data.

Testing

Should You Trust Your Reasoning Model?

Once we've trained a reasoning model, we can use that model to make predictions about new data.

We don't want the model to **only** work well on the data we provided originally. We want it to work well on the new data too.

When you build a reasoning model with a machine learning algorithm, you need to separate your data into three groups: **training** data, **validation** data, and **testing** data. This will let you evaluate your model on 'new' data once it is done training. The training data is usually the majority (70% to 80%) of the data set.

Testing Data Provides Final Results

When the algorithm thinks it's achieved an optimal model, the **testing data** is used to determine how accurate that model actually is. This is a portion of the data (maybe 10-15%) that was set aside at the beginning and never used during the training process.

Unlike the validation data, which is evaluated multiple times, the model is run on the test data **once**. We measure how close its predicted results are to the actual results. That score is the accuracy of the model.

You cannot train on your testing data if you want a fair test of the model!!!

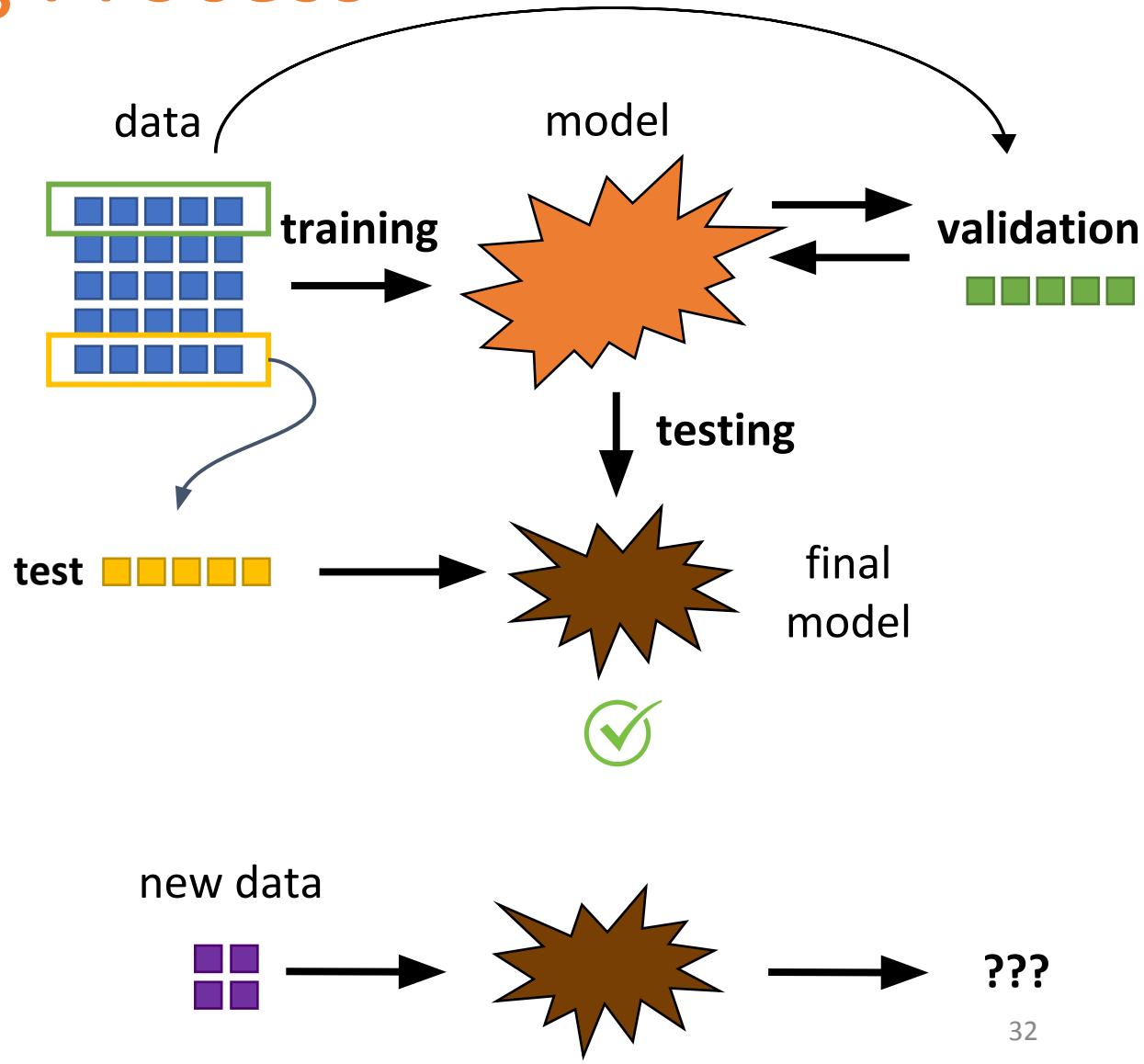
Example: Bad Training Process

What happens if we train on our test data?

The algorithm will get the opportunity to observe patterns in the test data. It will optimize the model to include those patterns.

When the model is tested, it will of course be accurate, because the model was optimized to notice the correct patterns.

But if we try to use the model on new, unlabeled data later on, the patterns may no longer be valid. We don't know for sure, because all of our labeled data was used for testing.



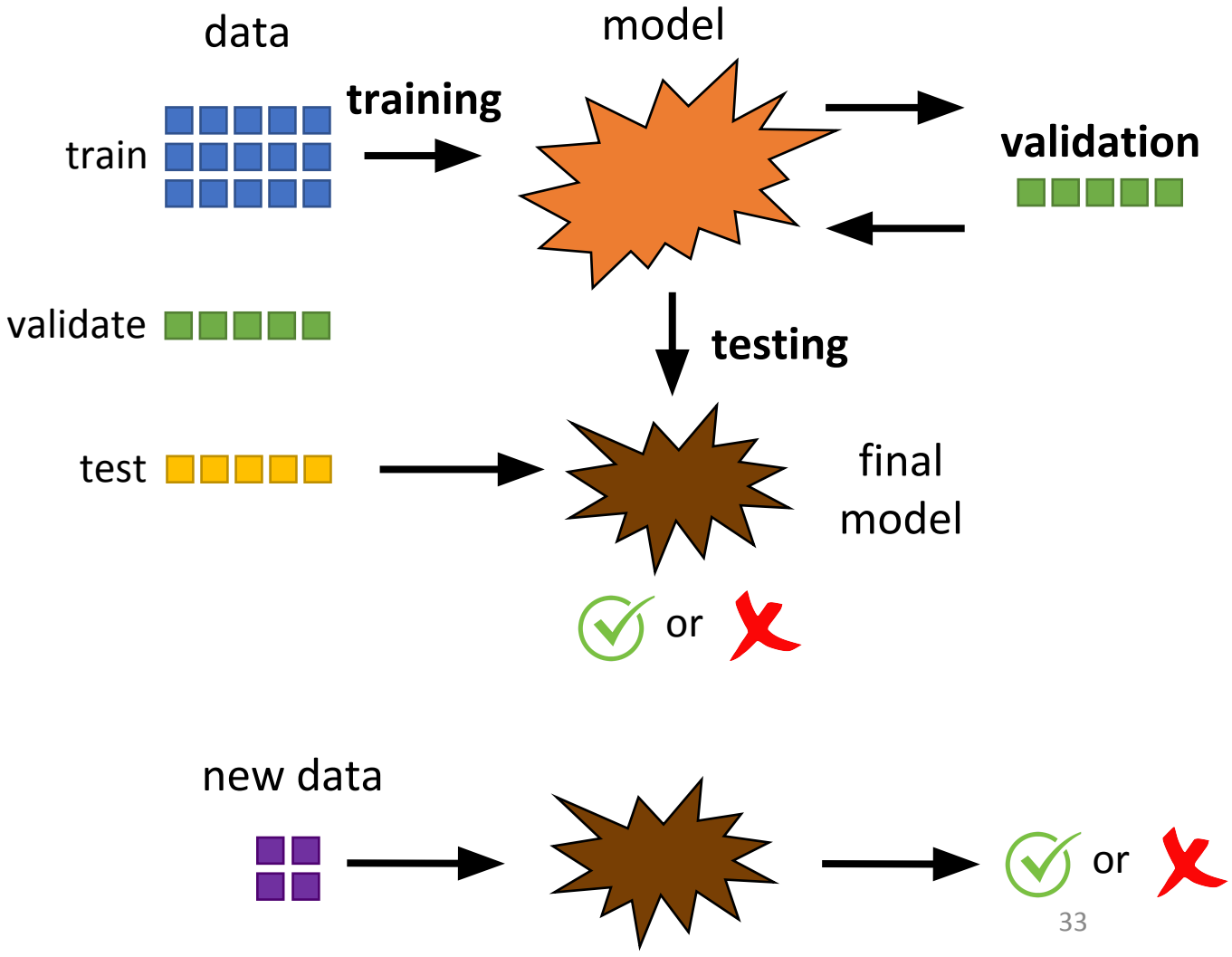
Example: Good Training Process

A better process: split the data into training, validation, and testing sets.

We'll train on the training set, and repeatedly test on the validation set. This should remove some of the overfitting from the training data.

When we're done, we'll test on the test set once. That produces our final result. It might be good, or it might be bad; it depends on how the model turned out.

However, the new data should have about the same accuracy as the test data, since the model never saw the test data before.

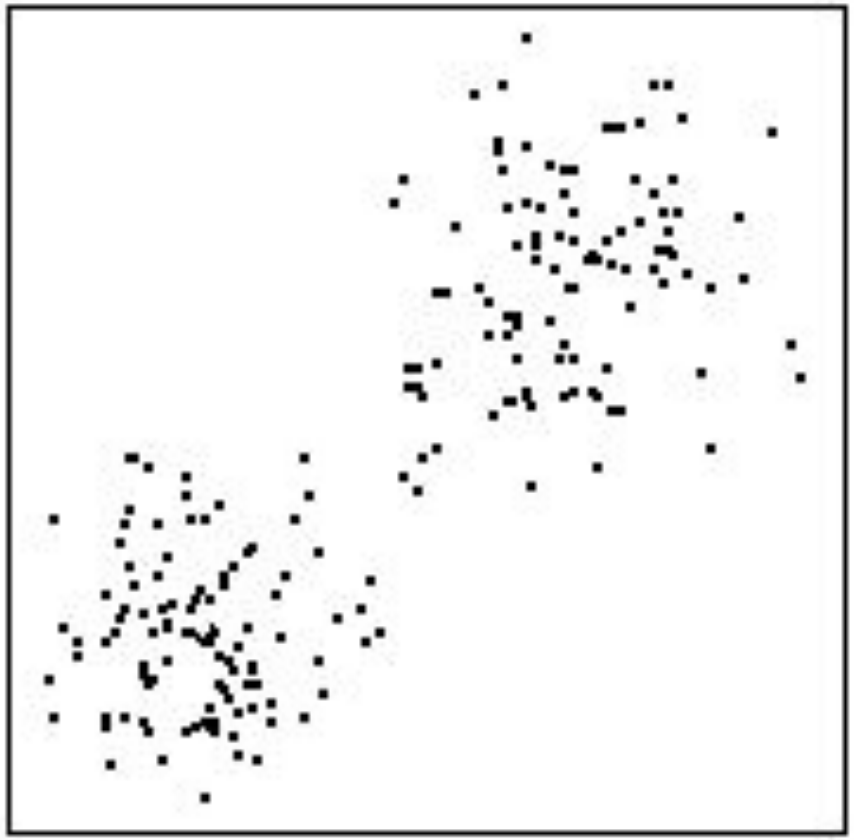


Unsupervised Learning

Unsupervised Learning Does Clustering

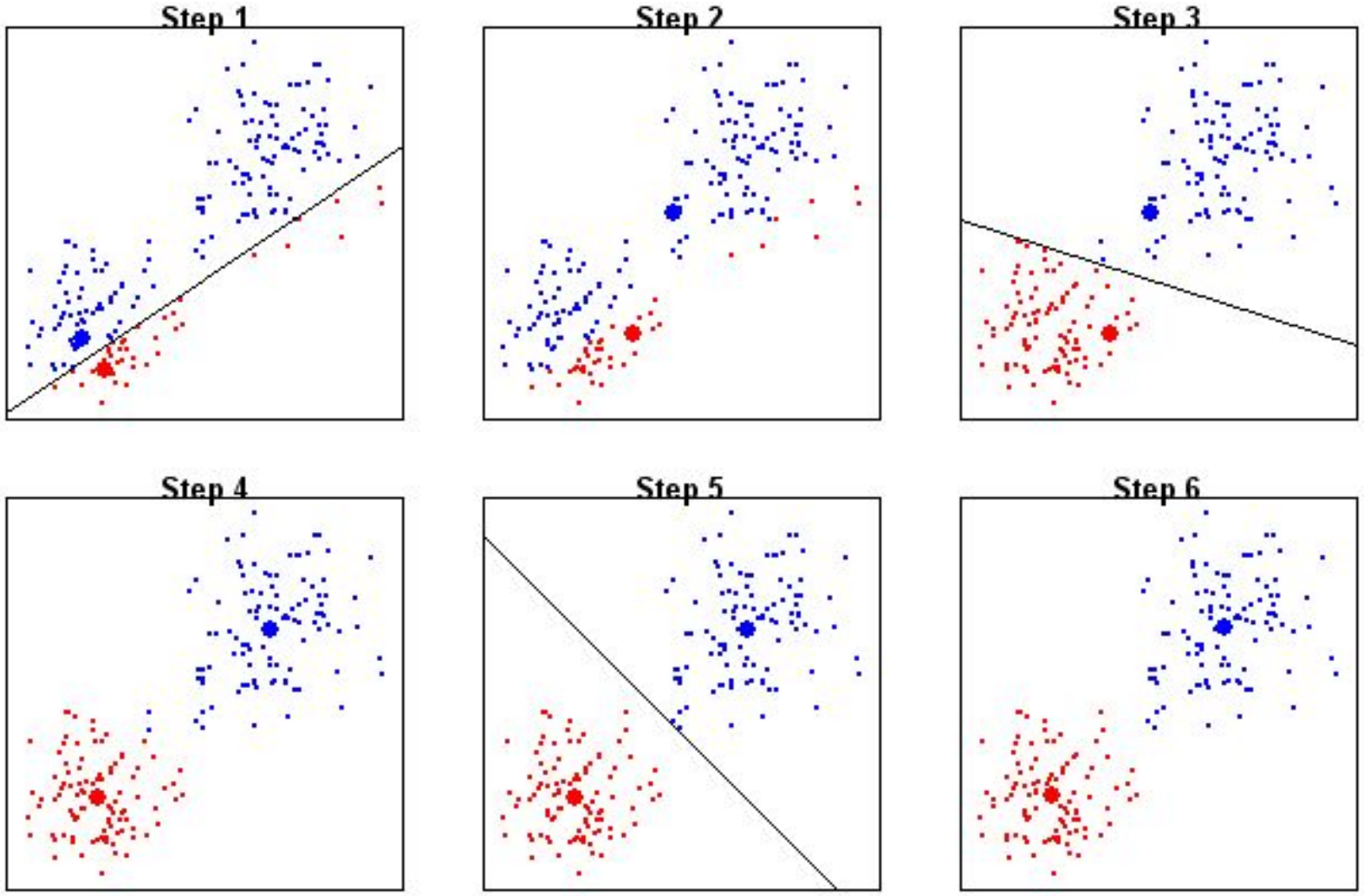
Unsupervised learning discovers classes in a data set by clustering similar data points together.

Find the Clusters in This Data Set



k-Means Clustering Algorithm

0. Generate random cluster centers.
1. Label each point based on the closest cluster center.
2. Recalculate cluster centers as the means of the points they captured.
3. Repeat steps 1-2 until no change.



http://sherrytowers.com/2013/10/24/k-means-clustering/

Learning Goals

- Identify three major categories of reasoning used with machine learning – **classification, regression,** and **clustering** – and decide which is the best fit for a problem
- Given a dataset, identify **categorical, ordinal,** and **numerical features** which may help predict the correct output for a given input
- Identify how **training data, validation data,** and **testing data** are used in machine learning to support **testing**
- **Feedback:** https://bit.ly/110-feedback