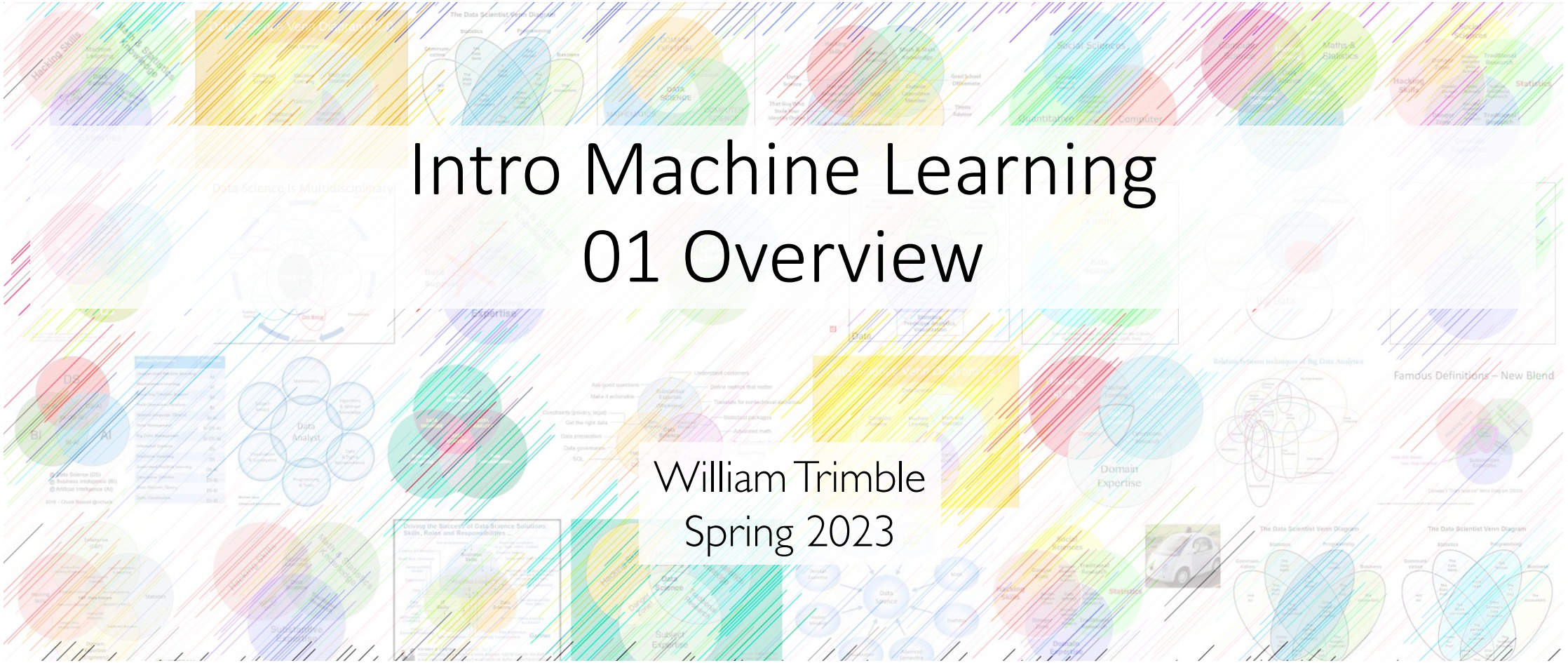


Intro Machine Learning
01 Overview

William Trimble

Spring 2023

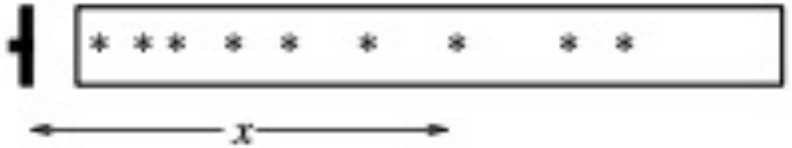
THE UNIVERSITY OF CHICAGO

Administrivia

- Week 1: read chapters 2 and 3 of Information Theory, Inference, and Learning Algorithms.
- Today: whirlwind overview
- Thursday: Bayesian estimation of continuous parameters
- Homework 1, heavy on prerequisites (calculus, linear algebra, some programming) is posted.

Exercise 3.3.[3, p.48] Inferring a decay constant

Unstable particles are emitted from a source and decay at a distance x, a real number that has an exponential probability distribution with characteristic length λ. Decay events can be observed only if they occur in a window extending from $x = 1\,\text{cm}$ to $x = 20\,\text{cm}$. N decays are observed at locations $\{x_1, \ldots, x_N\}$. What is λ?



Administrivia

- Grading: 70% homework, 30% group project
- Lowest homework dropped, no questions asked.
- Group project: ~7th week. Open-ended; choose a dataset and find something interesting if you can.

Who are we?

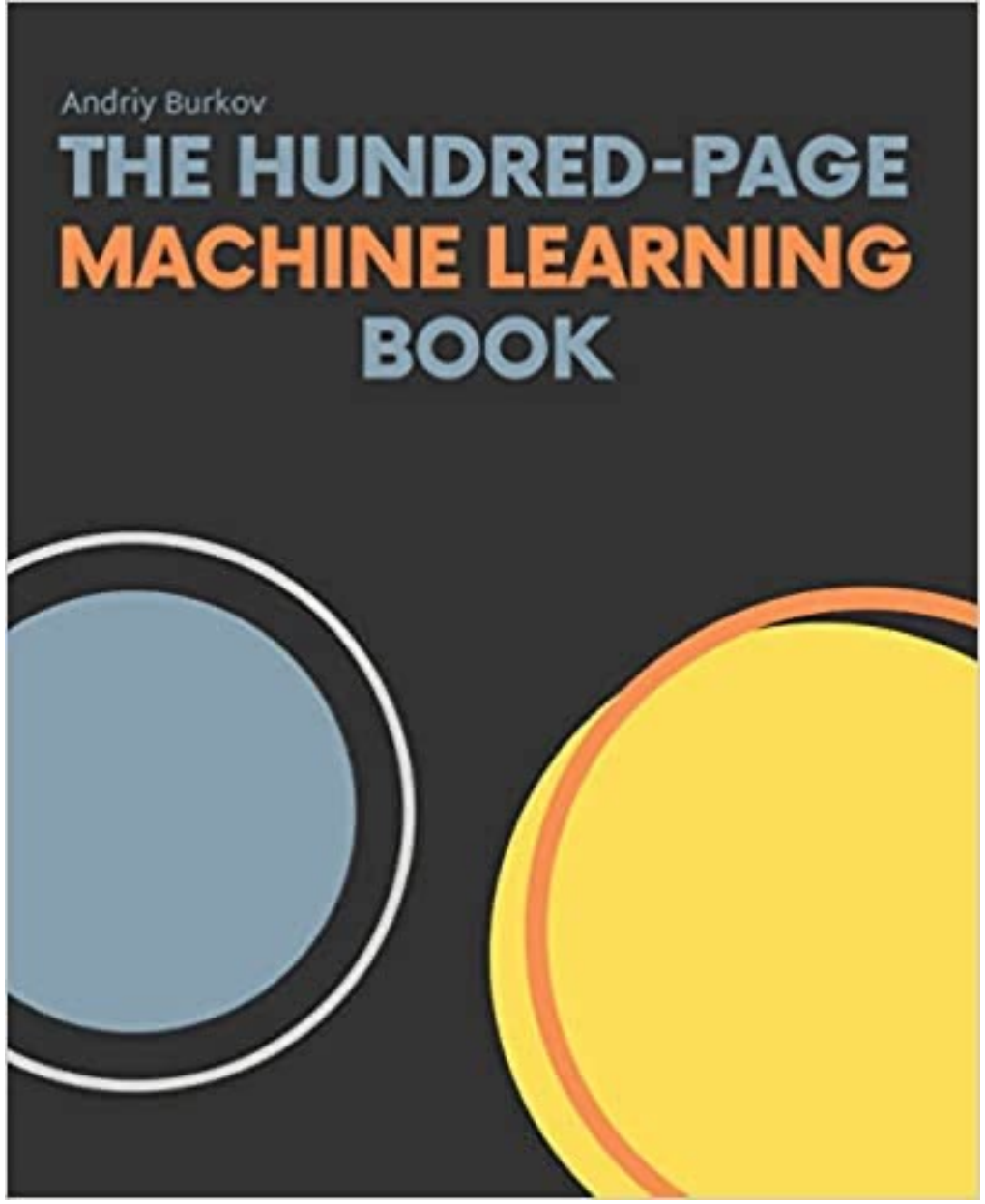
Instructors:

- William Trimble wltrimbl@uchicago.edu

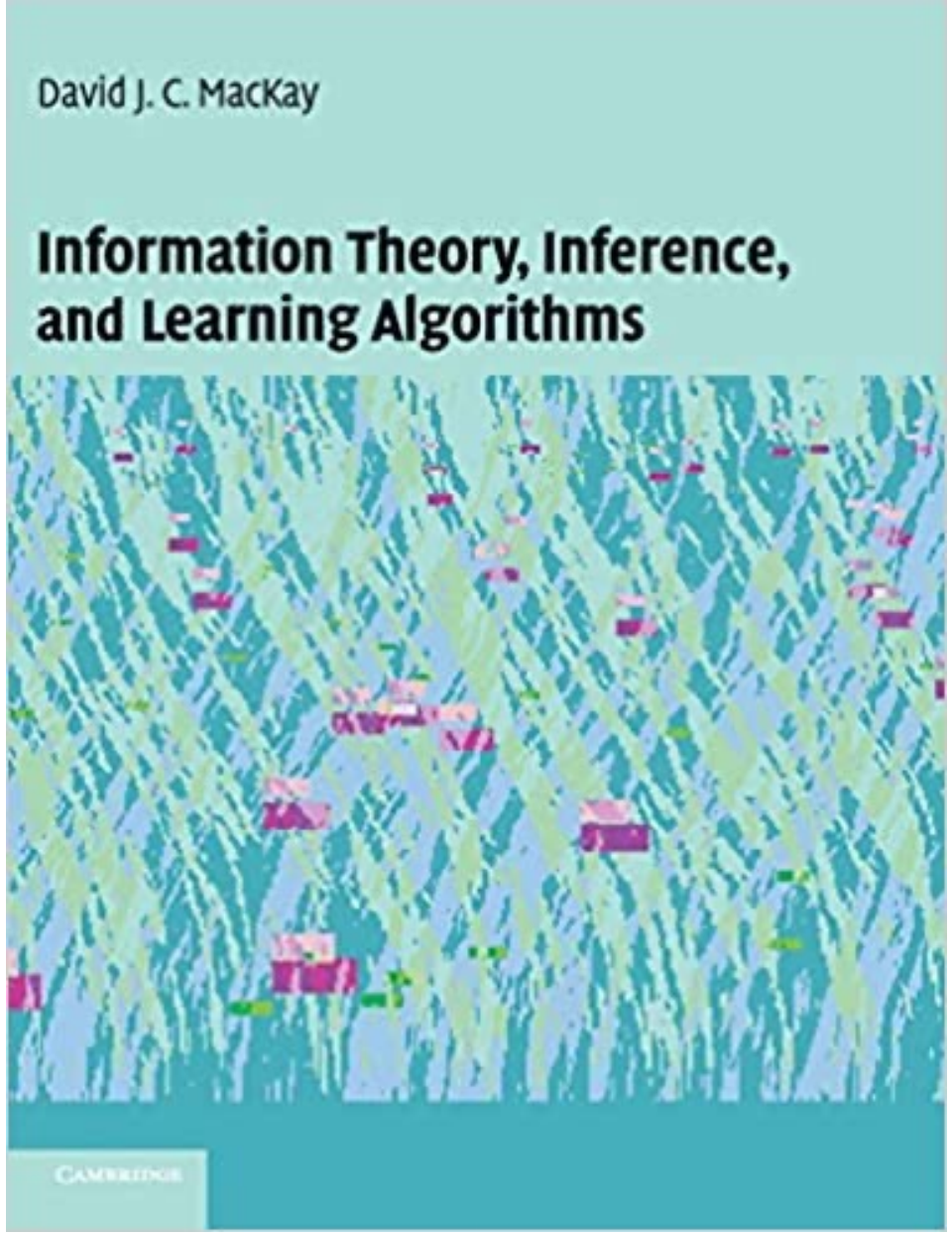
TA's:

Richard Huang <rrhuang@uchicago.edu>

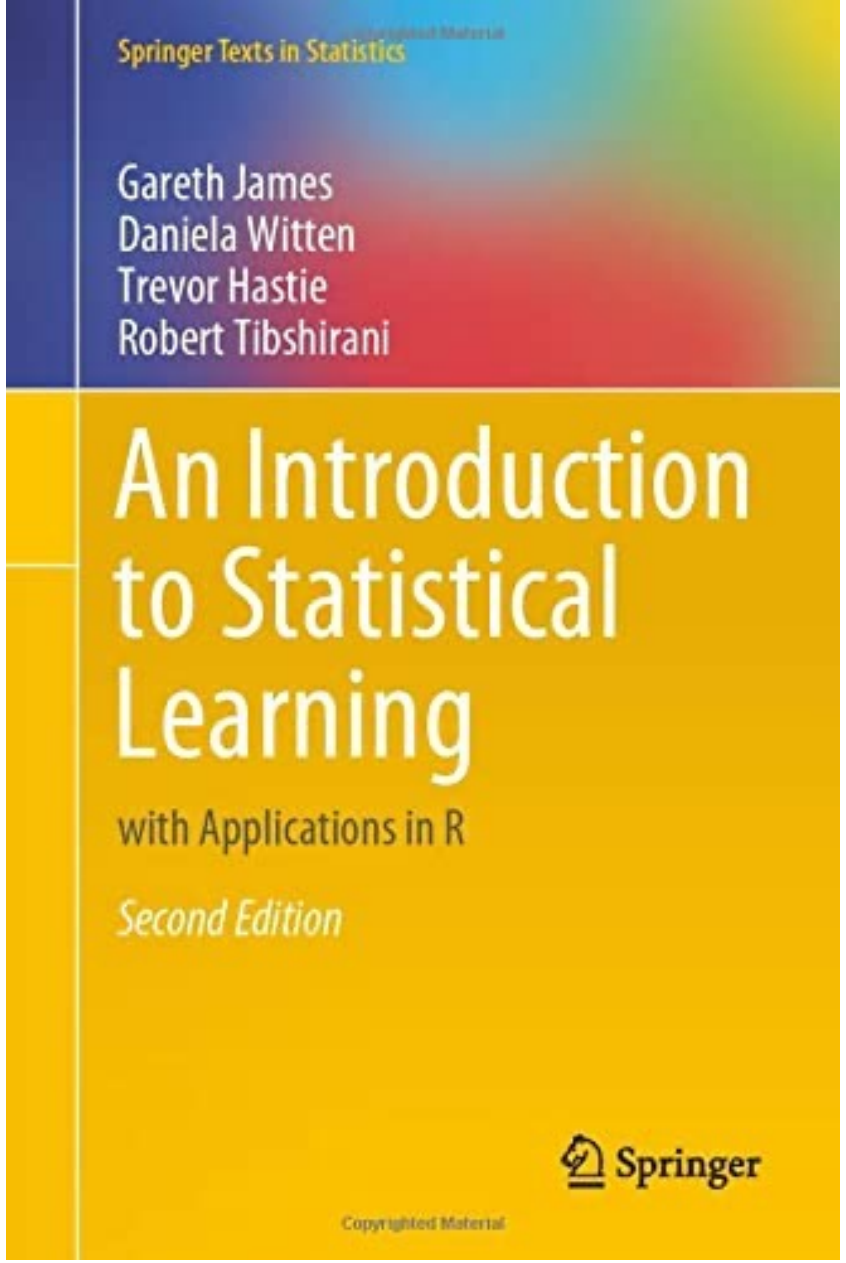
TA2: Melissa Adrian <@uchicago.edu> PhD student in Statistics

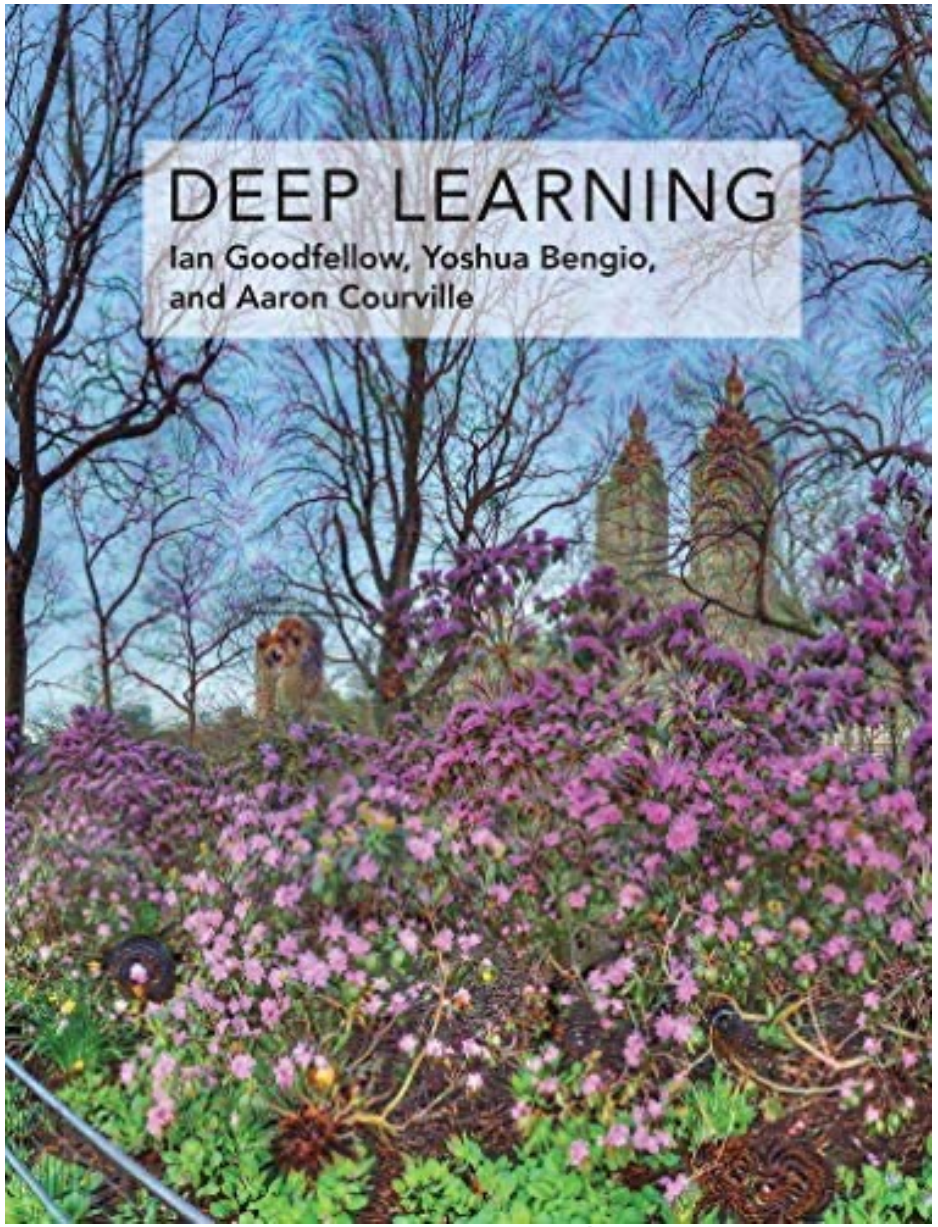


Andriy Burkov



David MacKay

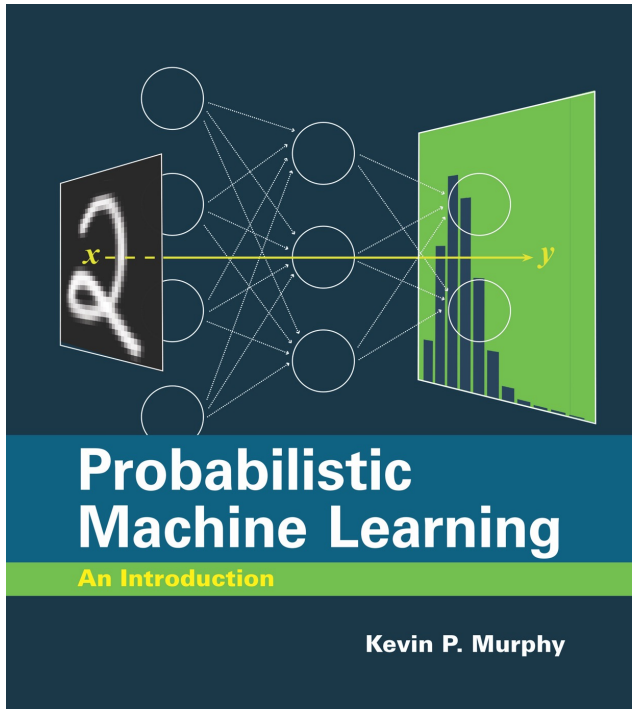




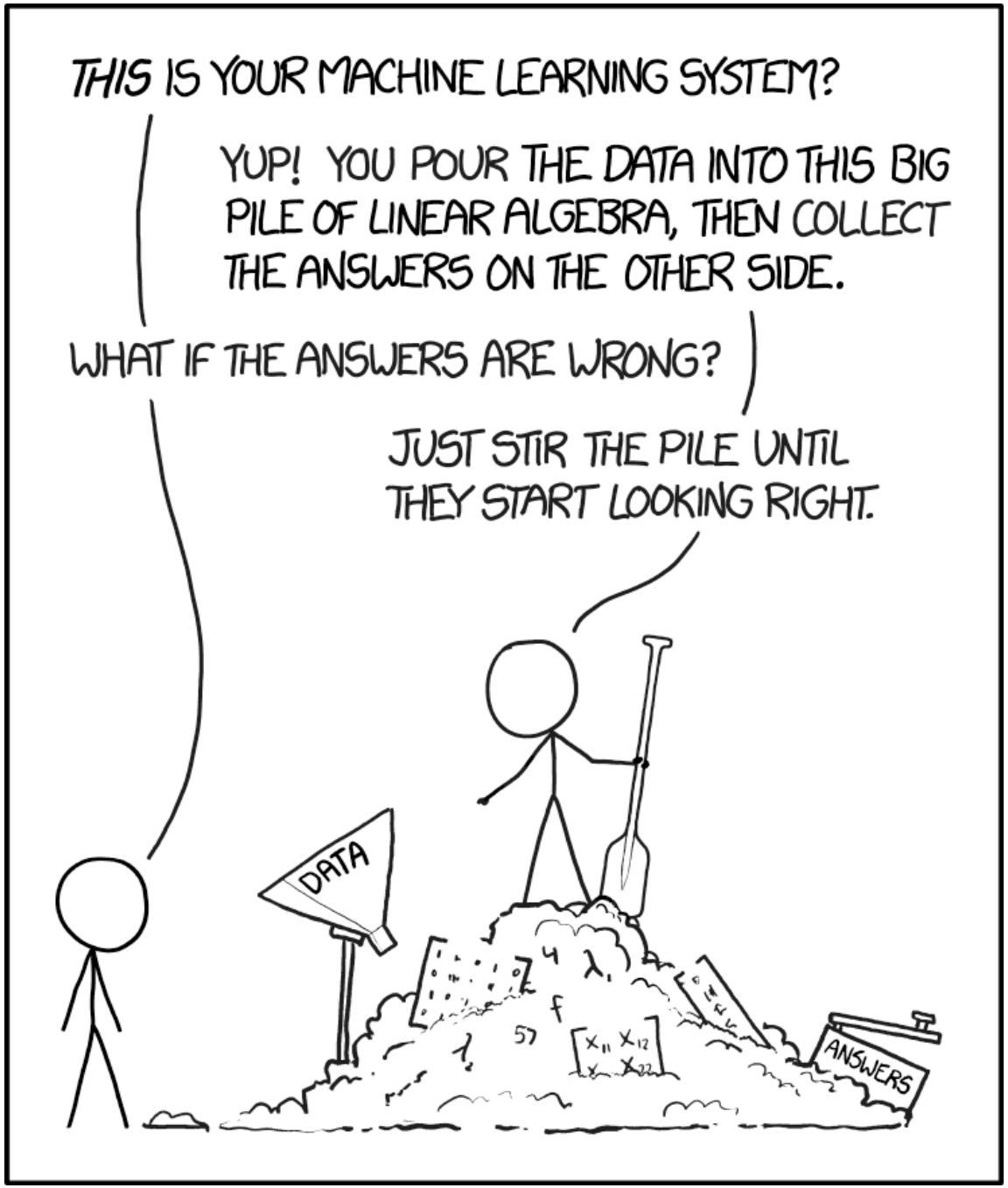
Gareth James

Ian Goodfellow

CS profs tend to be generous with their IP…



https://probml.github.io/pml-book/book1.html

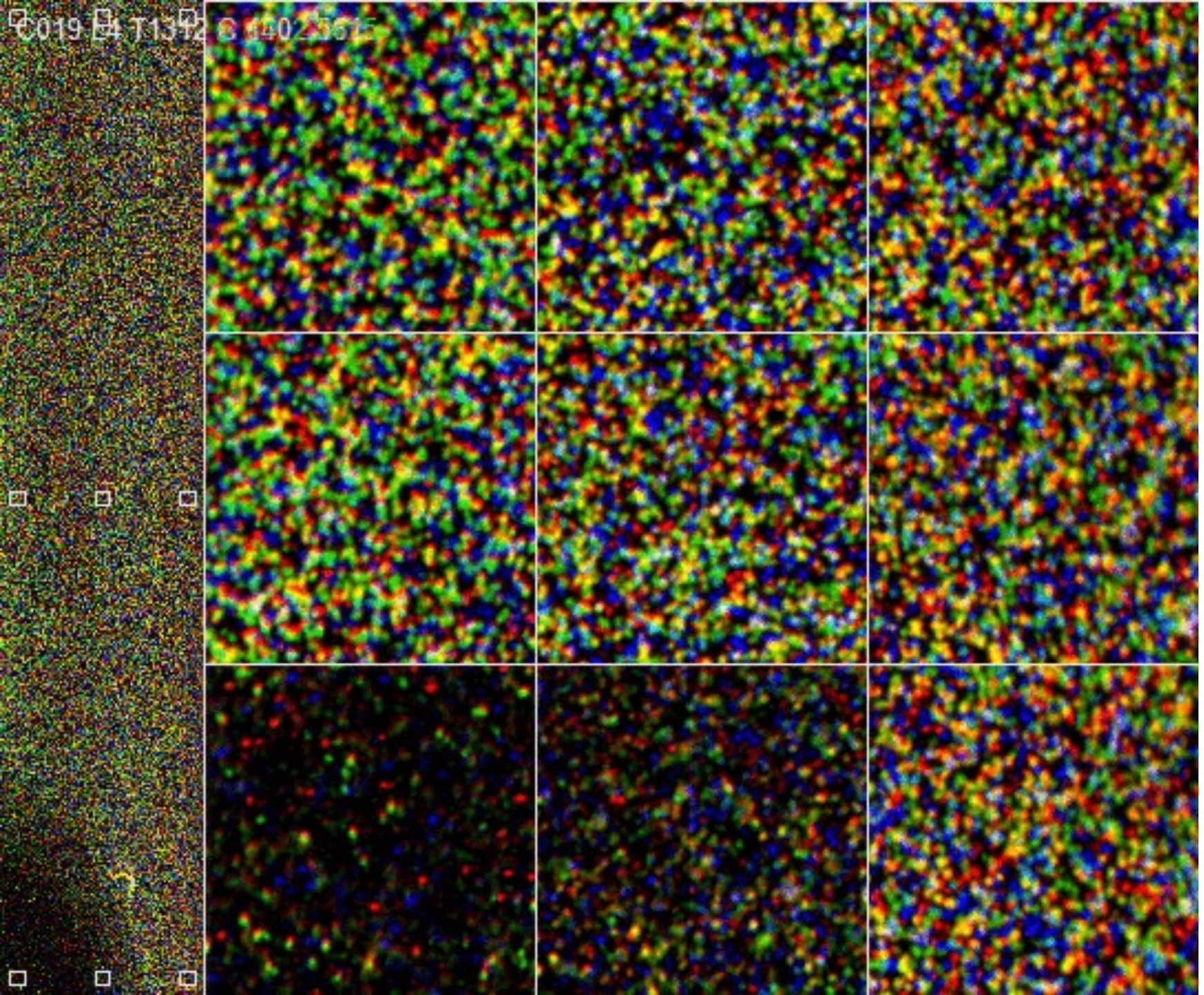


Why is this funny?

What is Machine Learning?

- That part of data analysis concerned with automated decisionmaking
- ”Modeling” “parameter estimation” are related but narrower
- Why data? If the task can be solved without data (say, by applying thresholds) you don’t need ML

This machine does not learn



- So-called NGS sequencing data: one of the data firehoses

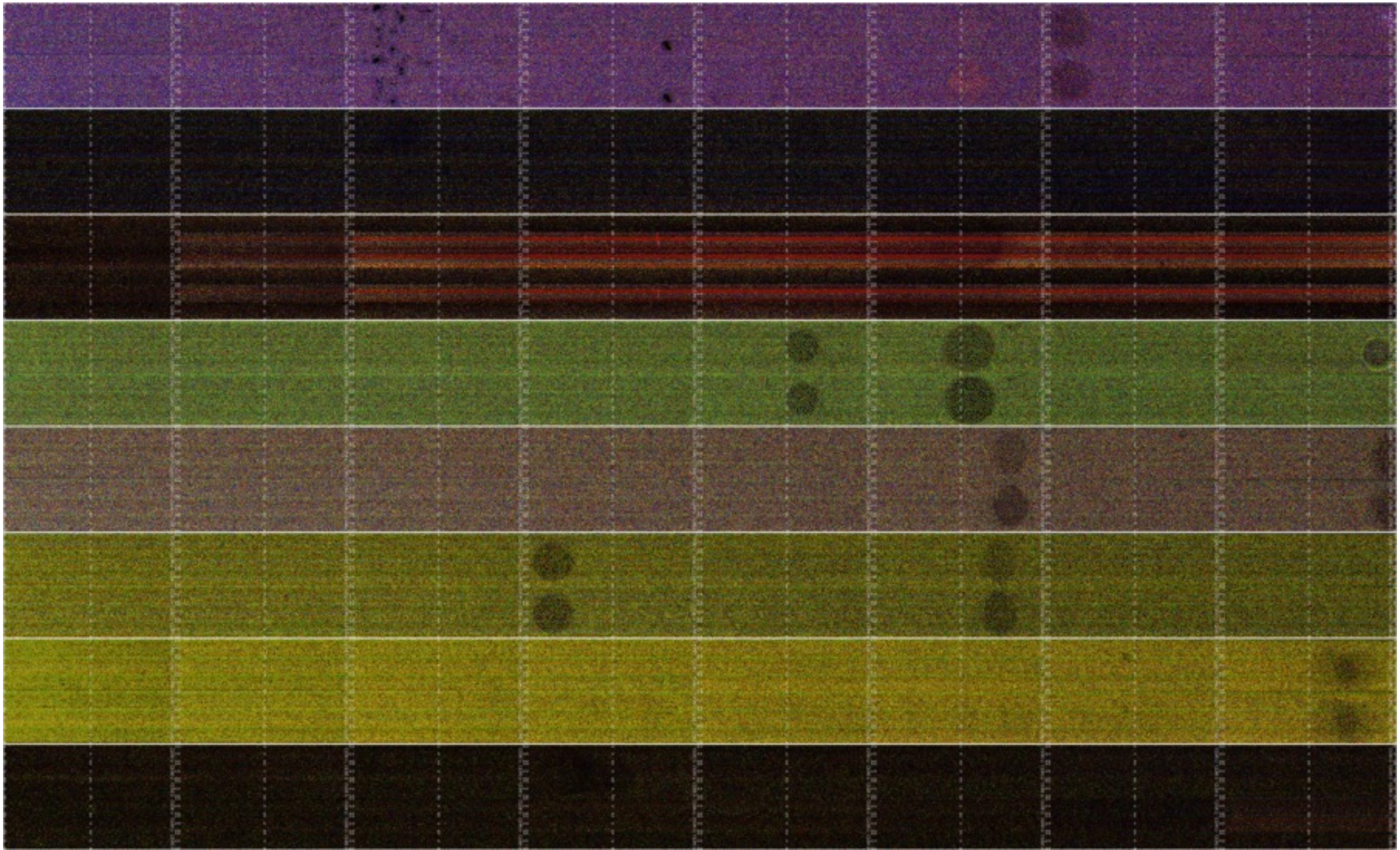


FIGURE 2.1. *A classification example in two dimensions. The classes are coded as a binary variable—*BLUE $= 0$, ORANGE $= 1$*—and then fit by linear regression. The line is the decision boundary defined by* $x^T\hat{\beta} = 0.5$*. The orange shaded region denotes that part of input space classified as* ORANGE*, while the blue region is classified as* BLUE*.*

Hastie, ESL, Ch 2

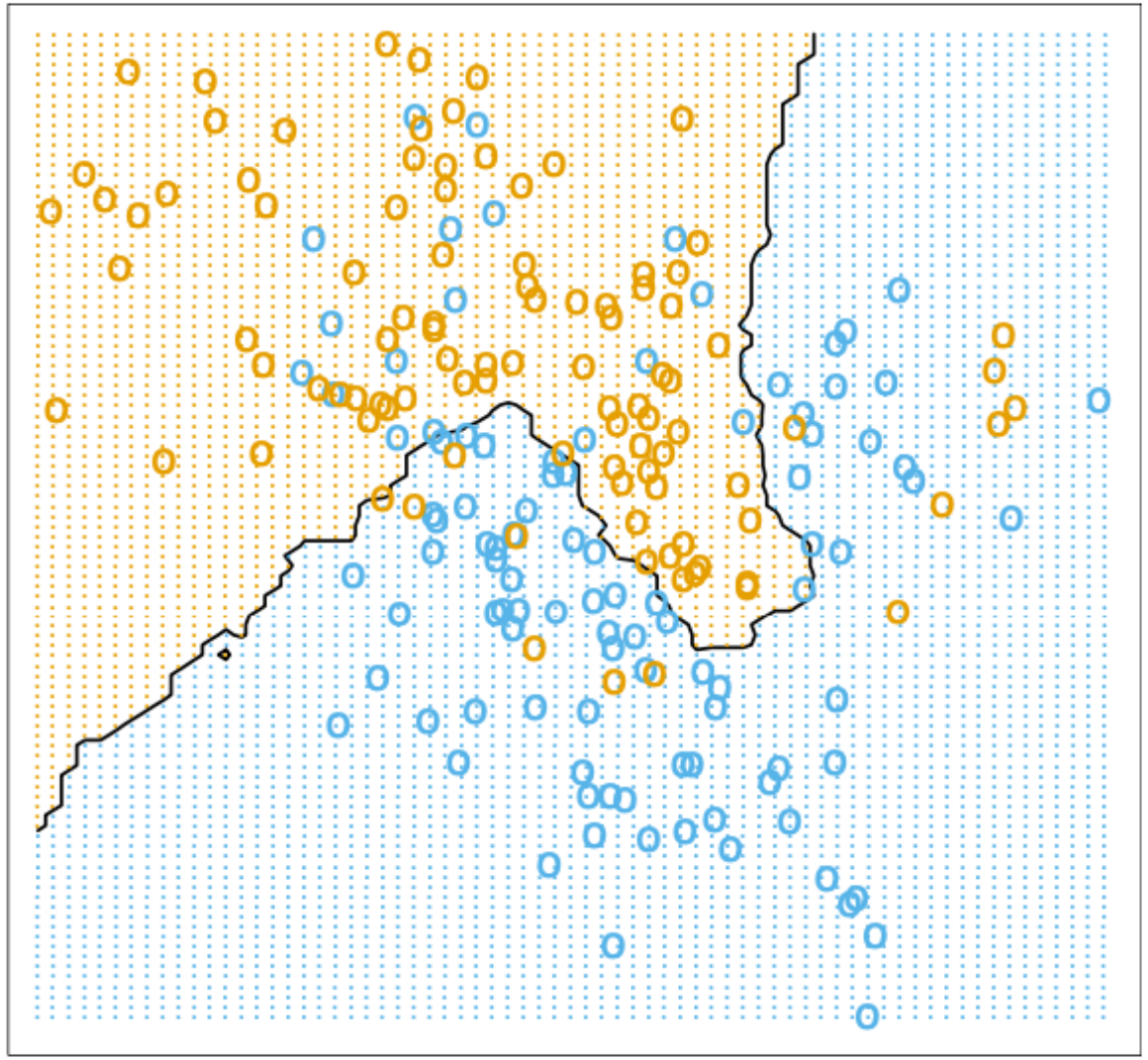


FIGURE 2.2. *The same classification example in two dimensions as in Figure 2.1. The classes are coded as a binary variable* (BLUE $= 0$, ORANGE $= 1$) *and then fit by* 15*-nearest-neighbor averaging as in (2.8). The predicted class is hence chosen by majority vote amongst the* 15*-nearest neighbors.*

Hastie, ESL, Ch 2

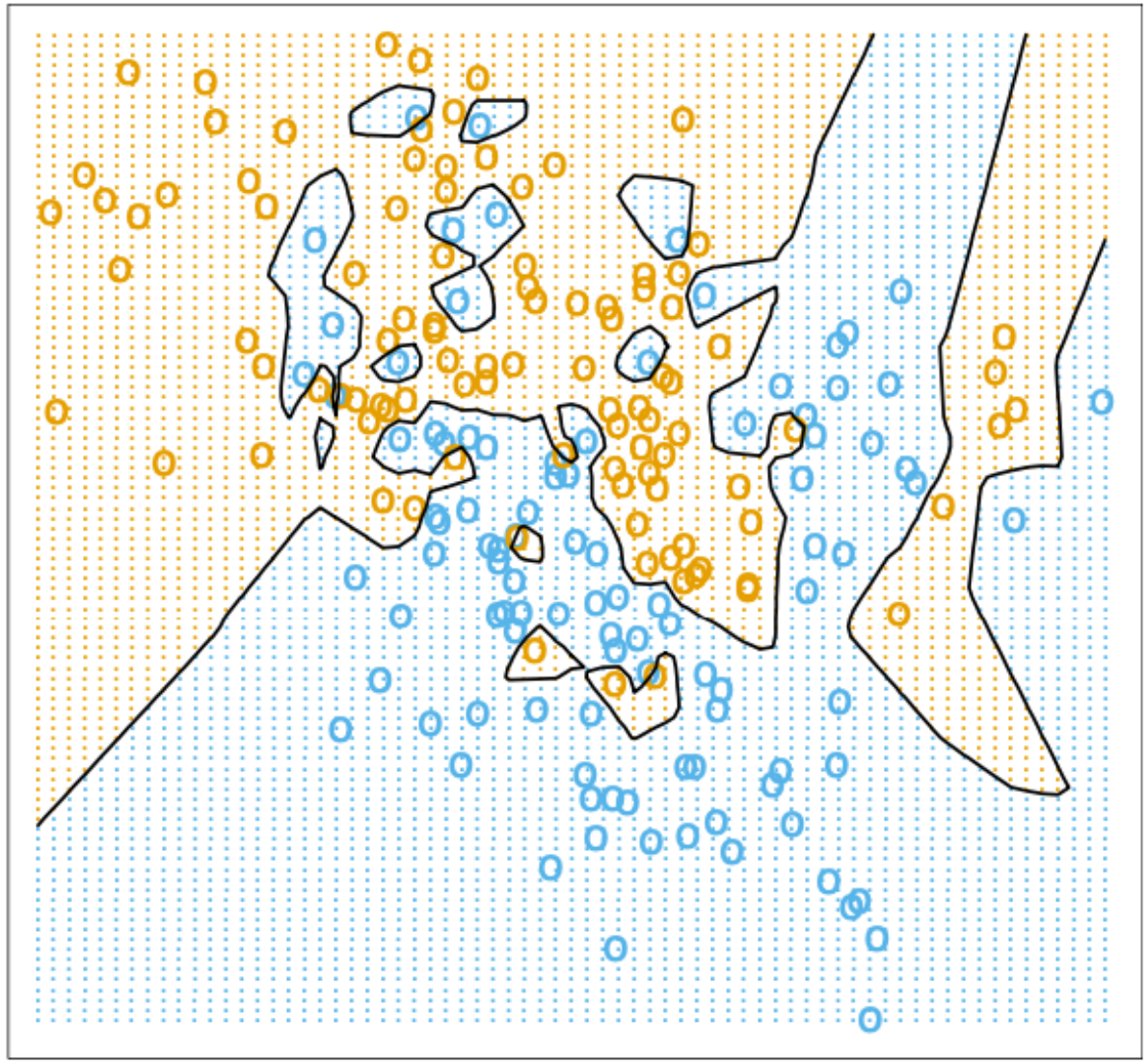
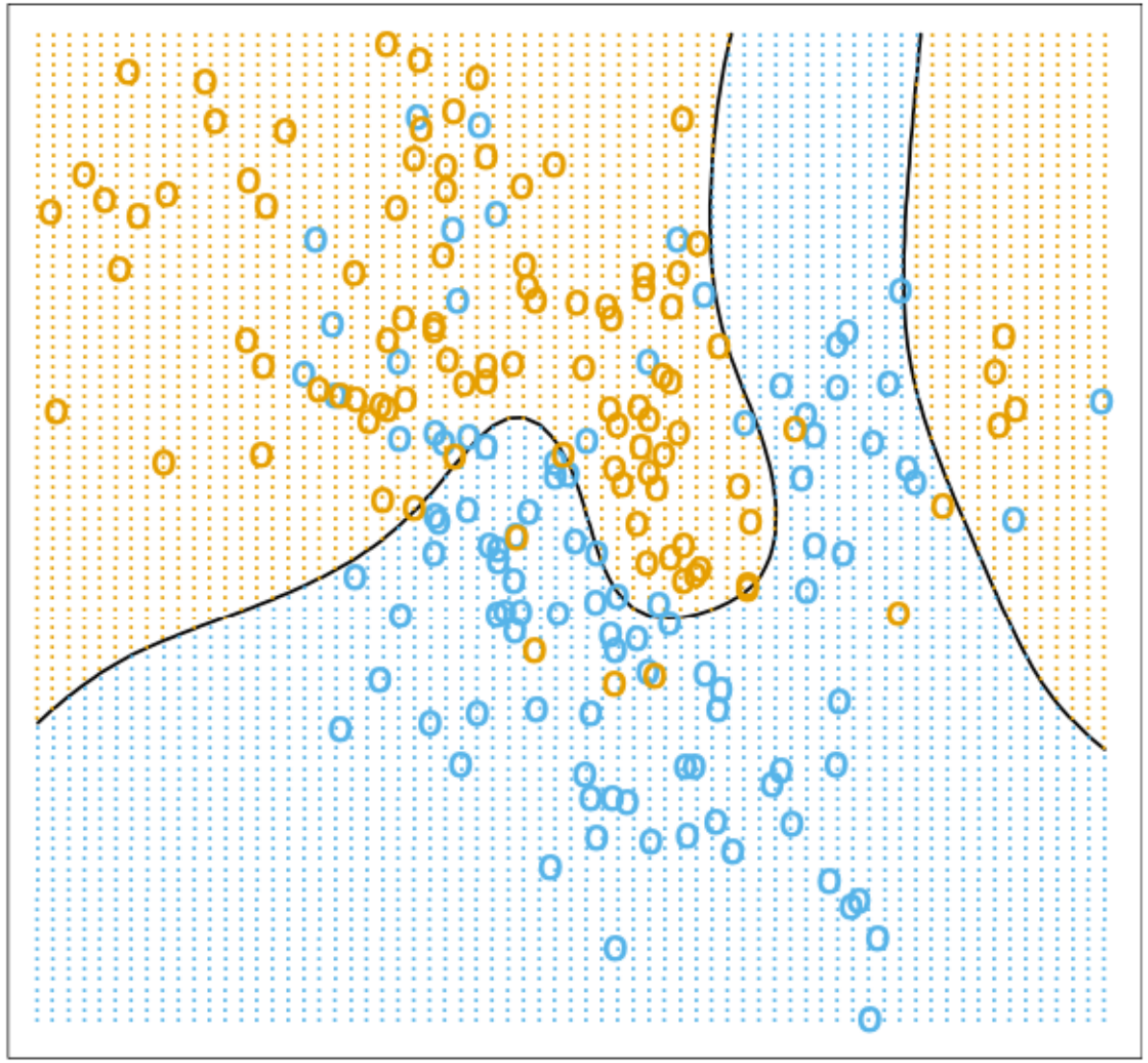


FIGURE 2.3. *The same classification example in two dimensions as in Figure 2.1. The classes are coded as a binary variable* (BLUE $= 0$, ORANGE $= 1$)*, and then predicted by* 1*-nearest-neighbor classification.*

Hastie, ESL, Ch 2

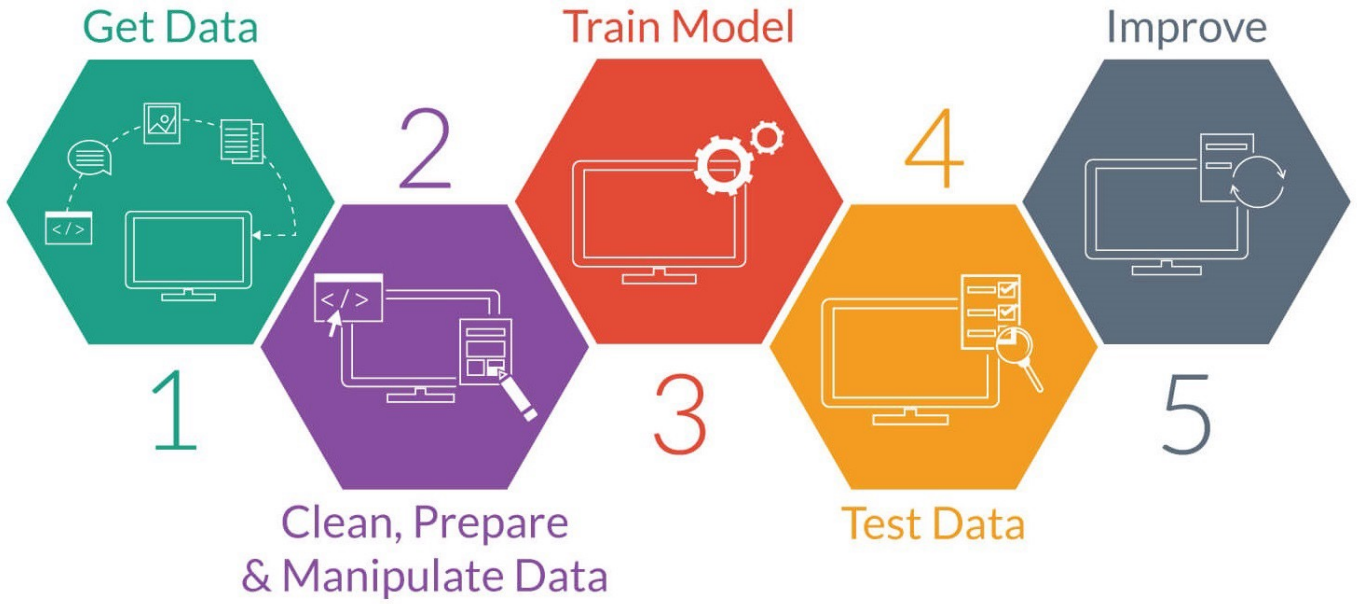


The optimal Bayes decision boundary for the simulation example. Since the generating density is known for each class, this boundary can be calculated exactly.

Hastie, ESL, Ch 2

The steps

- Define the Task (input space and output space)
- Data preprocessing (feature extraction & wrangling)
- Measure performance
- Train (or “learn”) the model
- Victory ?!?



There are myriad tasks

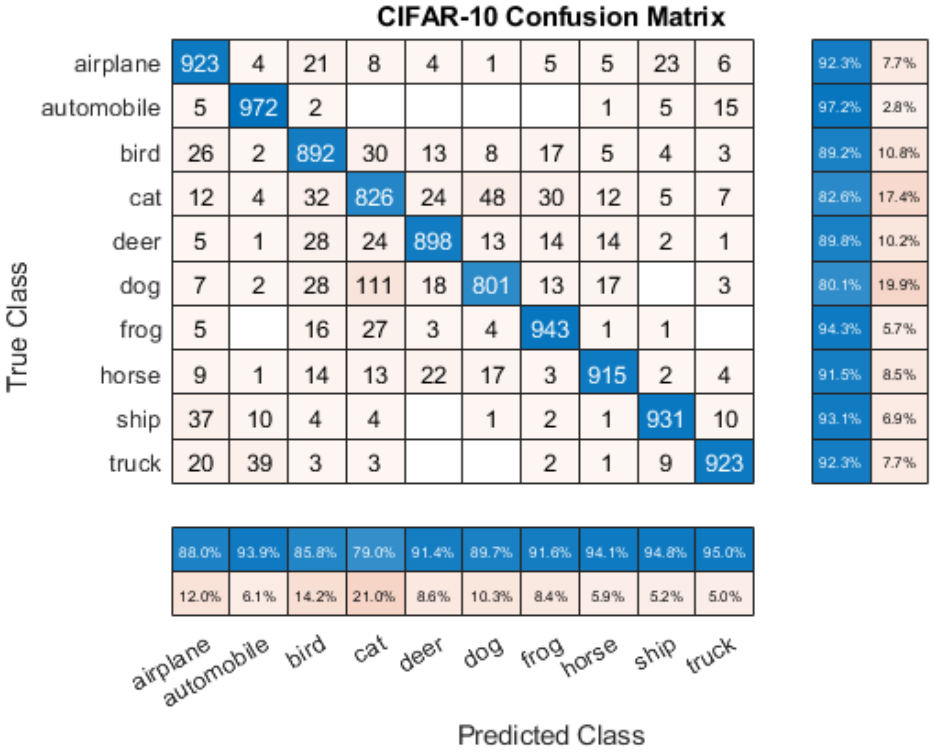
- Regression -- predicting some values of interest
  - Imputation of missing values
- Classification – put data points into categories
- Classification with missing inputs
  - collection of many classifiers, each for different set of missing data items?
  - generate a probability density for everything, marginalize over missing inputs
- Transcription –encode symbols in an image or audio waveform
- Structured output – answer very specific questions, like map annotation, write computer programs or captions
- Anomaly detection – fraud detection, for instance
- Noise detection / removal / error correction
- Density estimation; fitting (parameters of interest)

Goodfellow, Ch 5

There are myriad tasks

- Regression -- predicting some values of interest — Advertising, prediction
  - Imputation of missing values
- Classification – put data points into categories — Medical diagnosis
- Classification with missing inputs — Recommender algorithms
  - collection of many classifiers, each for different set of missing data items?
  - generate a probability density for everything, marginalize over missing inputs
- Transcription –encode symbols in an image or audio — Machine translation
- Structured output – answer very specific questions, like map annotation, write computer programs or captions — generation of textures, waveforms
- Anomaly detection – fraud detection, for instance — Fake news; security applications
- Noise detection / removal / error correction — Geodesy; remote sensing; astronomy
- Density estimation; fitting (parameters of interest); matching — fingerprints, DNA

Measure Performance



- How well does the model answer our question?
- Measure “accuracy”
- Maybe fold in costs for different sorts of errors

to guide decision thresholds; manage our training

in light of expected frequencies

- Measure accuracy on the training set and on data not used for training “testing” or “holdout” sets.
- Calculus helps us for continuous variables / continuous inputs
- Brute force or Markov-chain Monte Carlo for discrete variables
- Space of parameters is often exceedingly large and complex; we usually have to resort to randomized algorithms and truncated representations.

Goodfellow, Ch 5

Train (or “learn”) the model

- Unsupervised algorithms usually try to distill the empirical probability density into something that can be succinctly communicated.
  - This the essence of compression
  - Clustering, finding groups of similar items, recommendation
  - Dimension reduction, embedding, big data visualization
- Supervised algorithms: training data with gold-standard labels or desired outcomes
- Semi-supervised: we can only afford to label some of the data, but have access to a larger population of unlabeled data for modeling its density

Goodfellow, Ch 5

Linear classifier

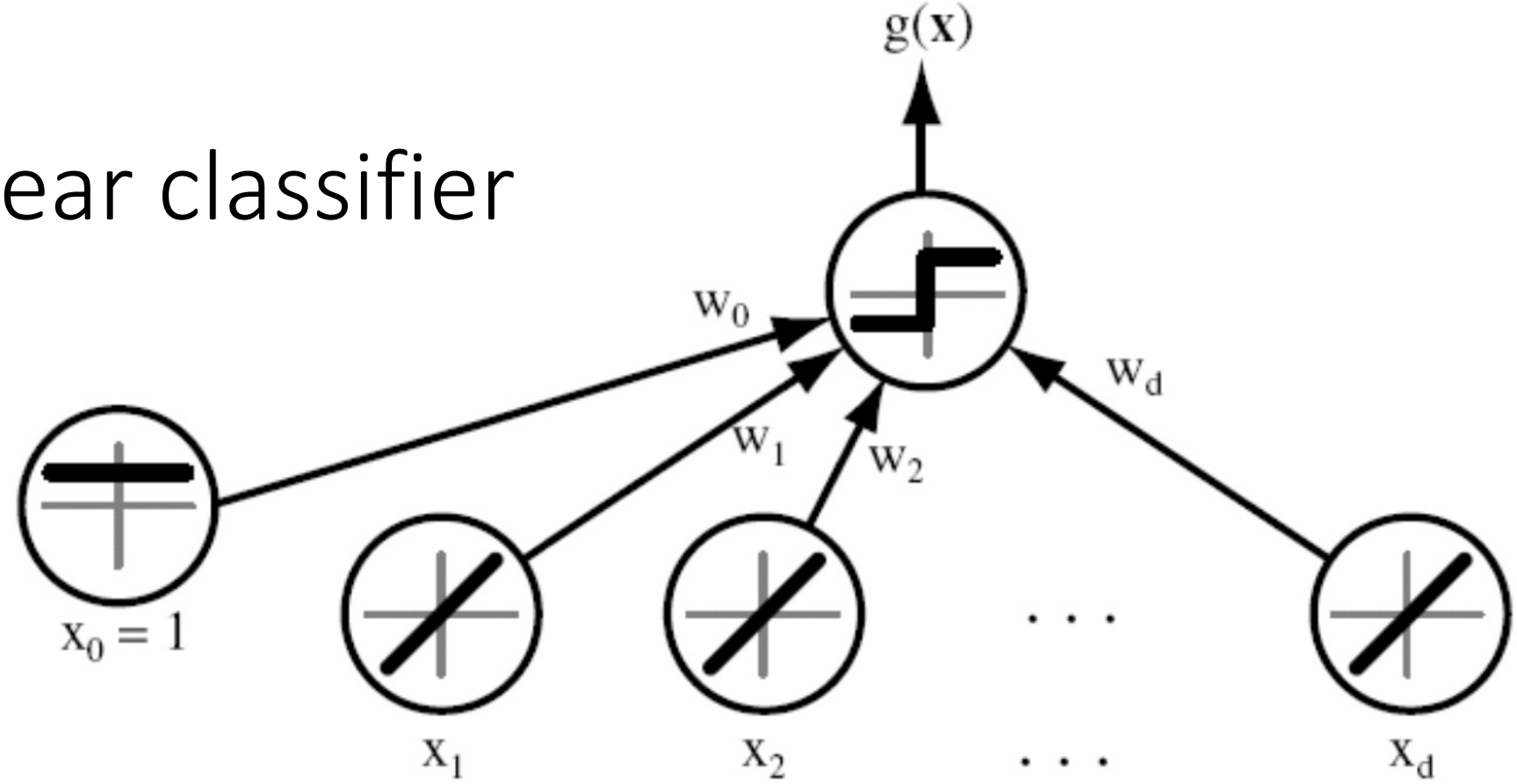
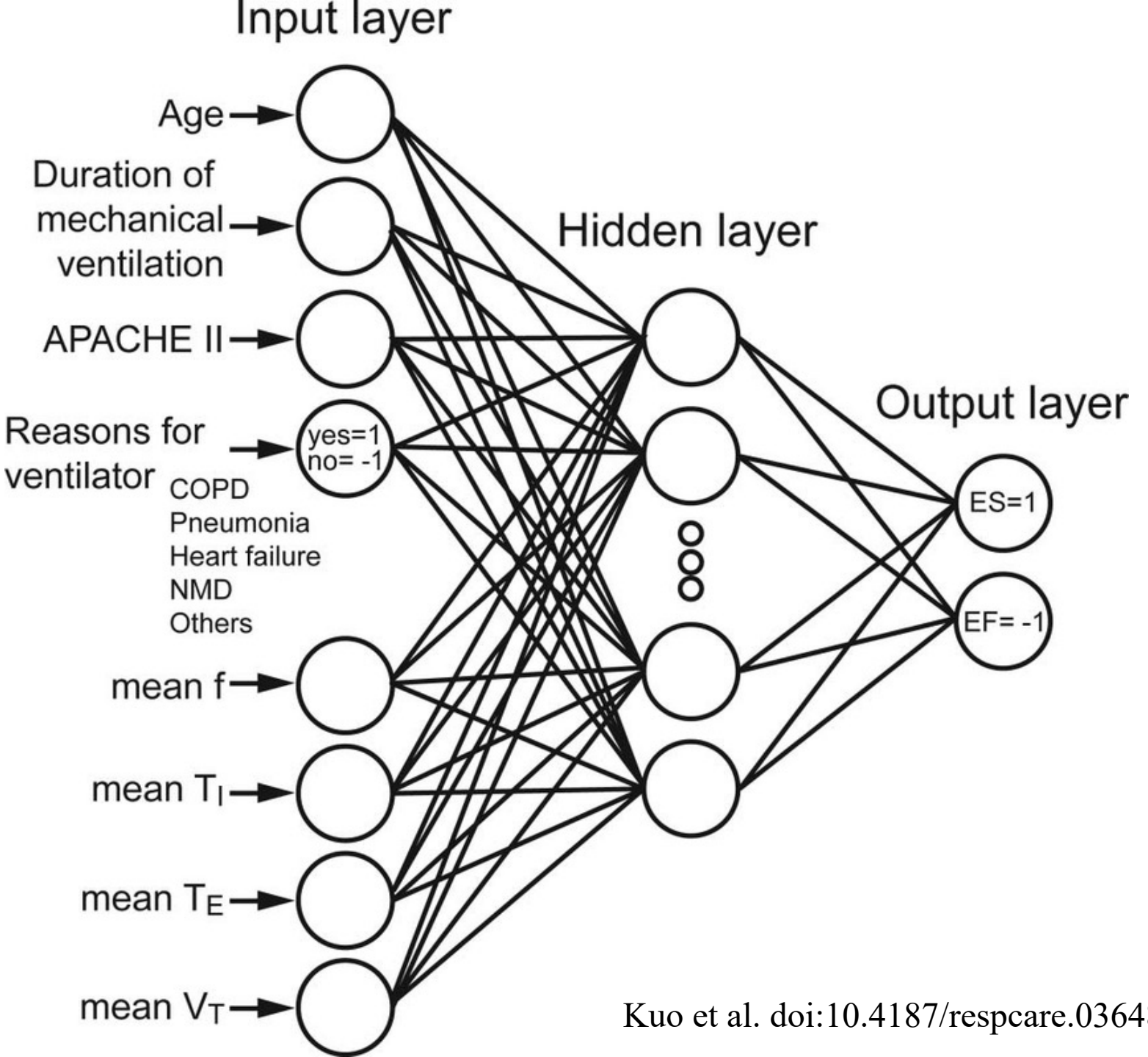


Figure 5.1: A simple linear classifier having d input units, each corresponding to the values of the components of an input vector. Each input feature value x_i is multiplied by its corresponding weight w_i; the output unit sums all these products and emits a $+1$ if $\mathbf{w}^t\mathbf{x} + w_0 > 0$ or a -1 otherwise.

Duda, Pattern Classification

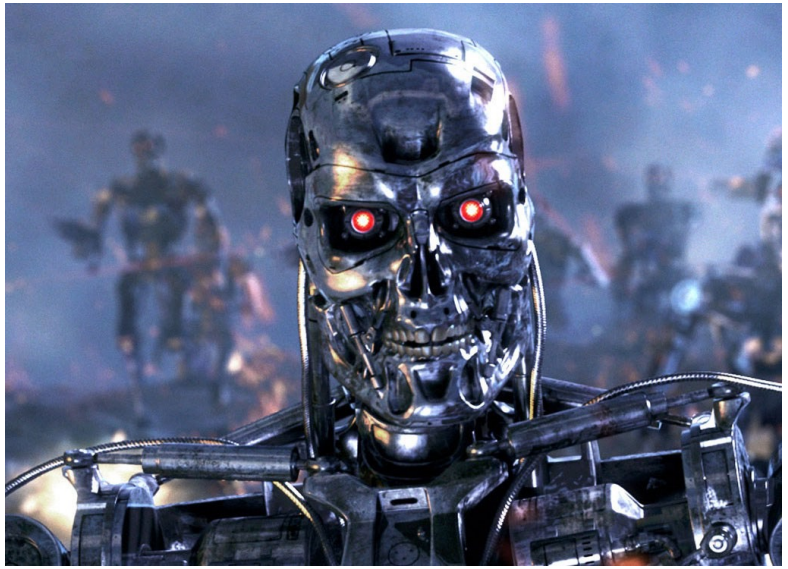
Neural networks

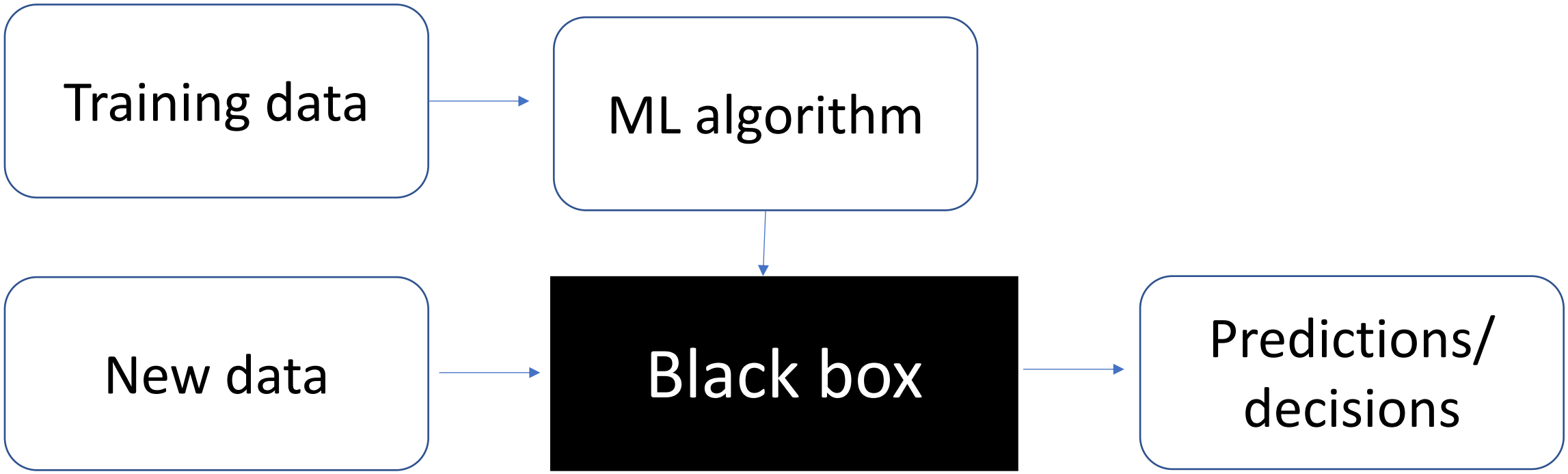
Linear classifiers
+nonlinear sauce
= arbitrary function machines

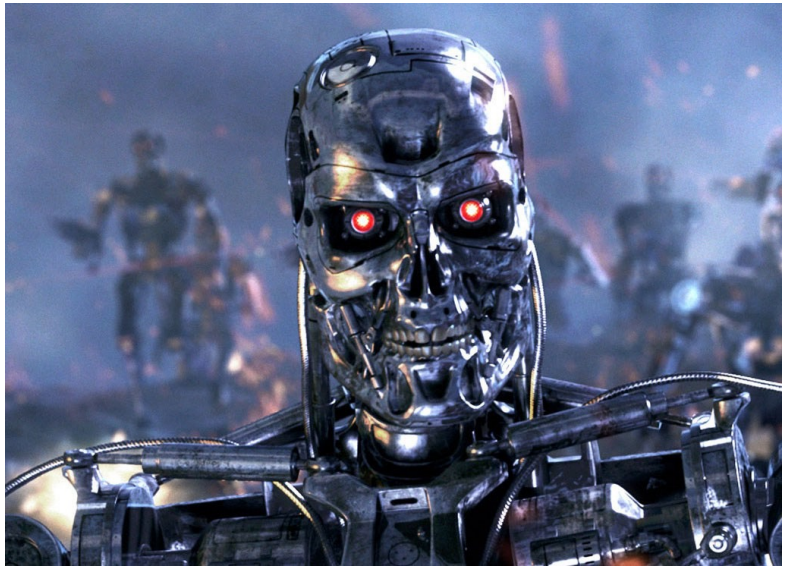


Kuo et al. doi:10.4187/respcare.03648

New data → **Black box** → Predictions/ decisions




Training data
ML algorithm
New data
Black box
Predictions/
decisions



Unlabeled data

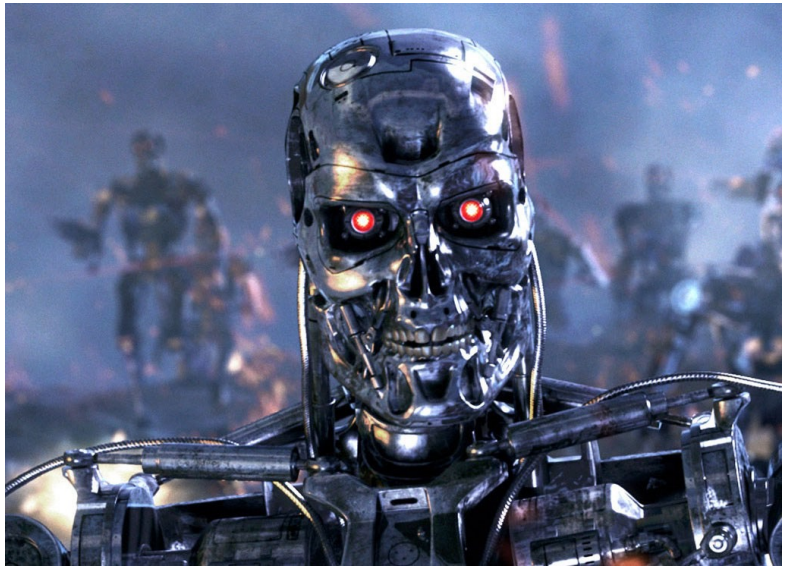
Training data

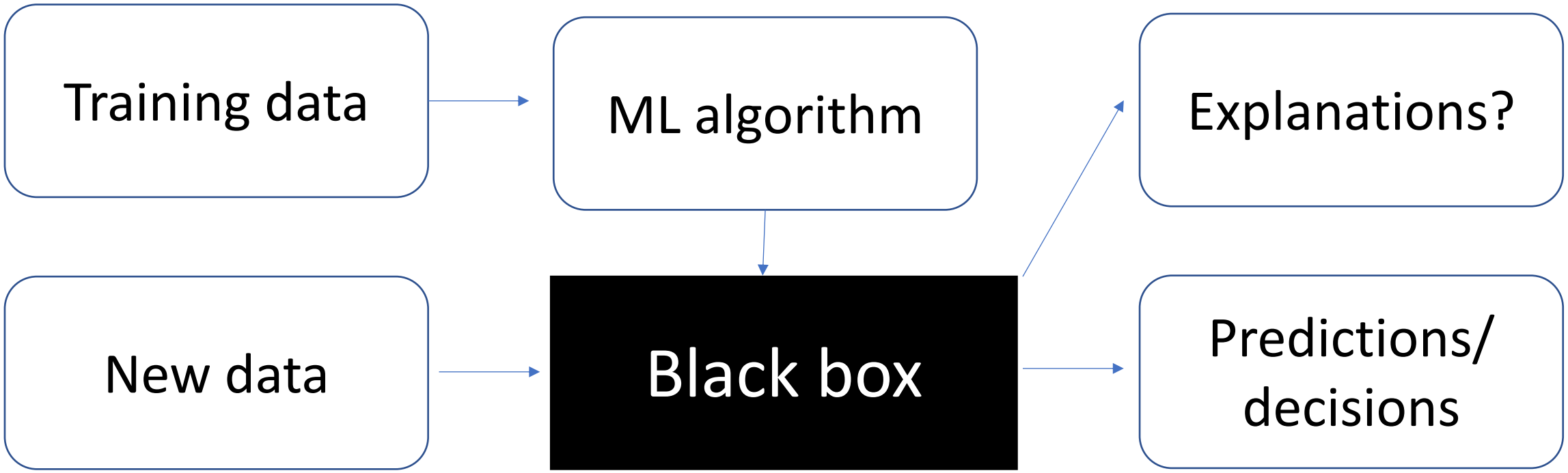
New data

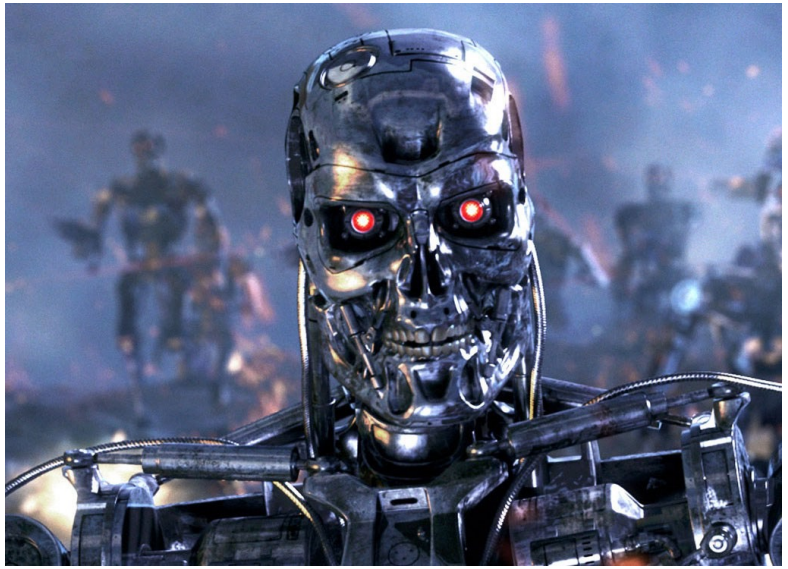
ML algorithm

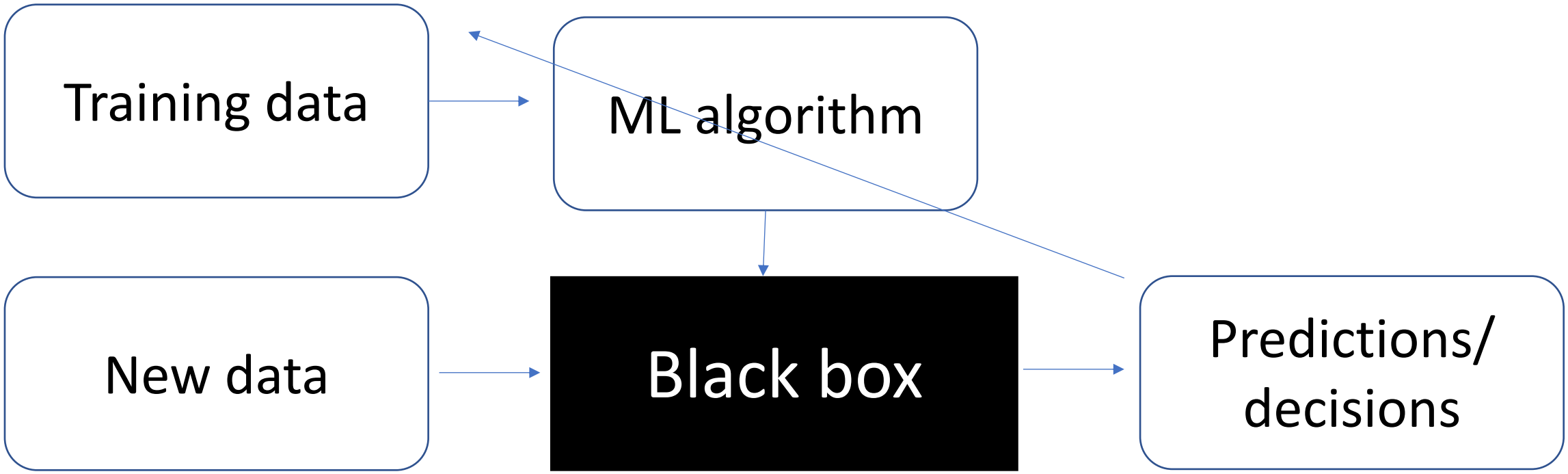
Black box

Predictions/
decisions

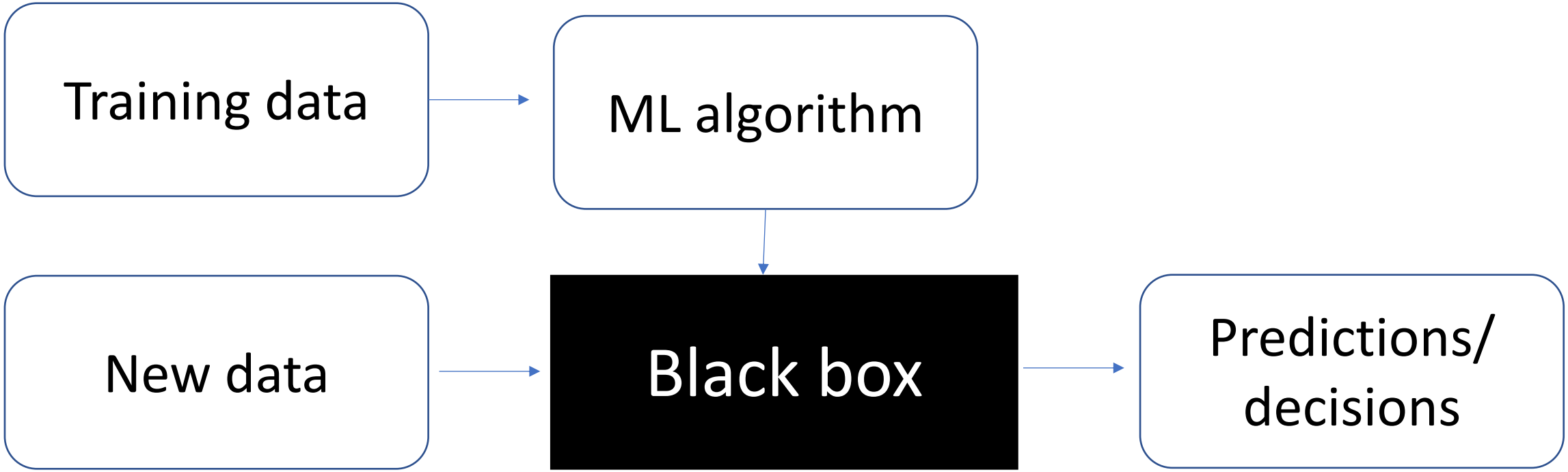



Training data
ML algorithm
Explanations?
New data
Black box
Predictions/
decisions




Training data
ML algorithm
New data
Black box
Predictions/
decisions

Where do the costs lie?



Where do the costs lie?

Big Data Borat
@BigDataBorat

In Data Science, 80% of time spent prepare data, 20% of time spent complain about need for prepare data.

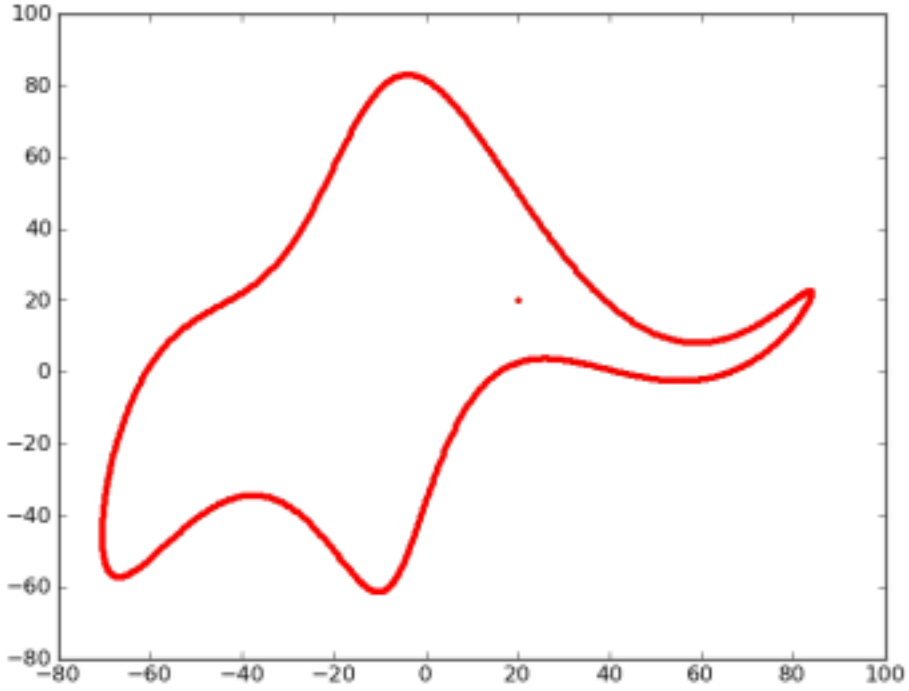
8:47 PM · Feb 26, 2013 · Twitter Web Client

544 Retweets **25** Quote Tweets **402** Likes

Most of the cost is in getting (locating, collecting, preparing) worthwhile, relevant, usable training data.

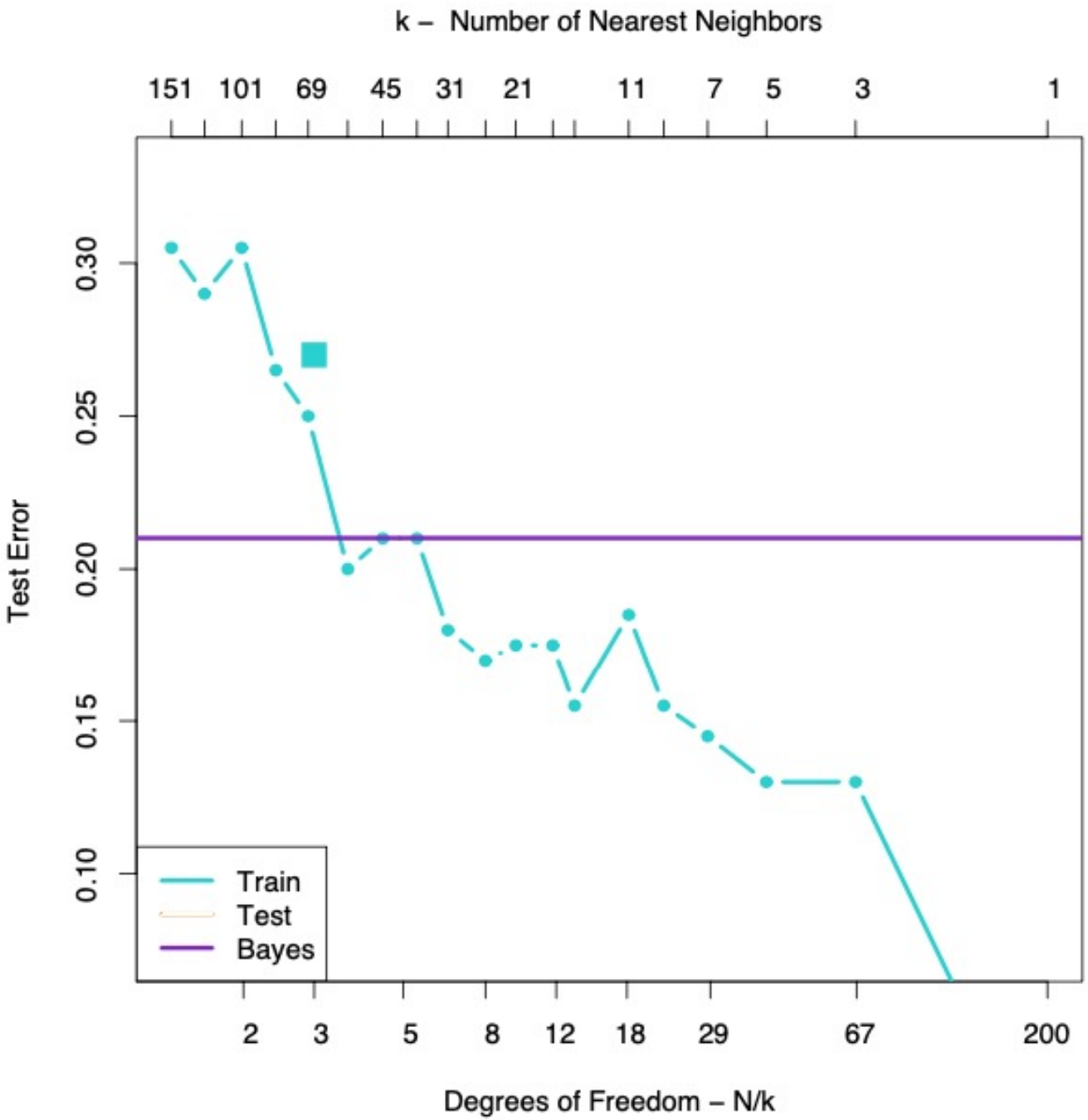
With an ocean of free parameters, we can fit anything.

"**with four parameters I can fit an elephant**, and with five I can make him wiggle his trunk."
– John von Neumann, as told to Enrico Fermi, as told by Freeman Dyson.



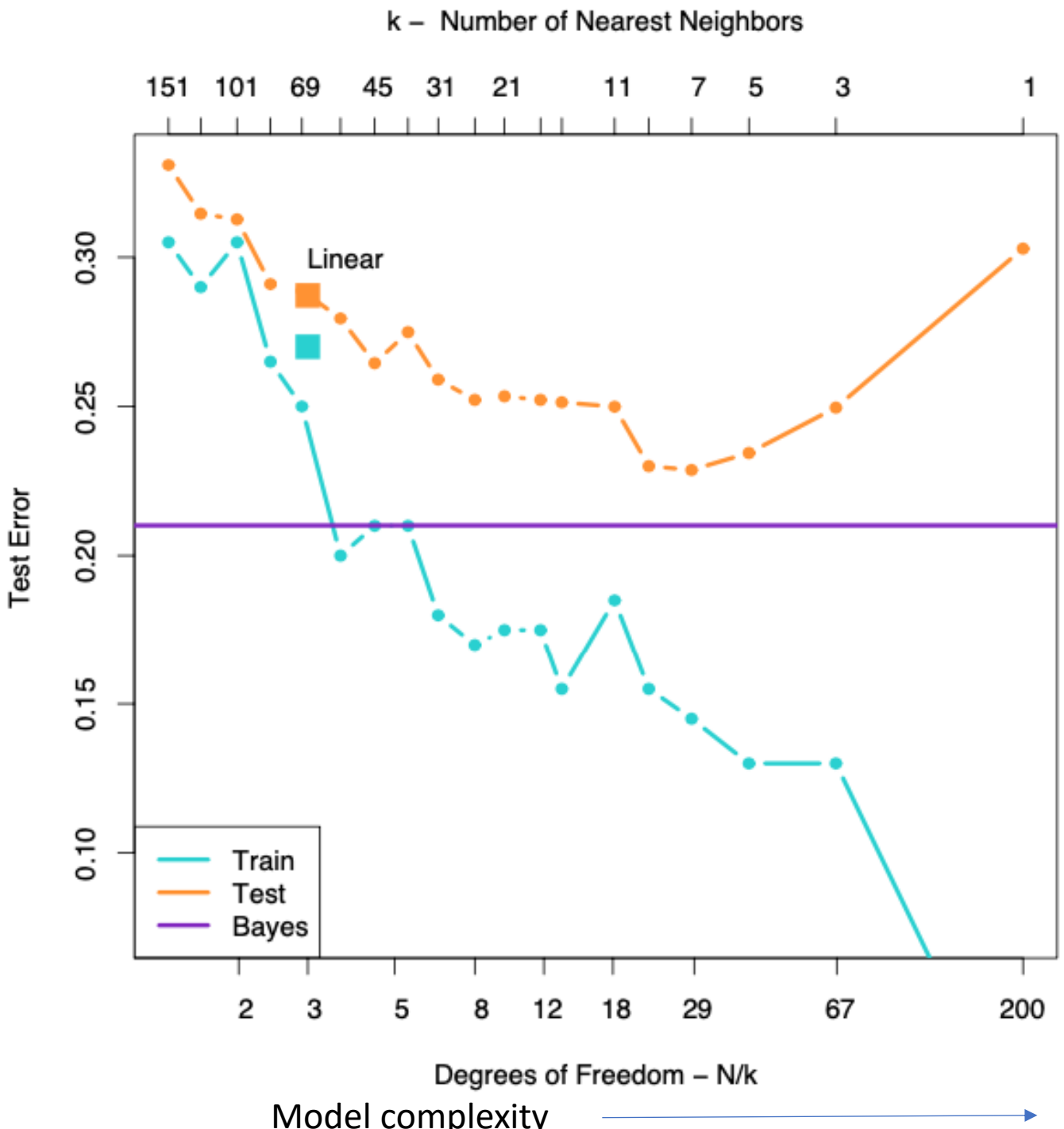
doi://10.1038/427297a

doi://10.1119/1.3254017



But explaining my data perfectly does not confer utility.

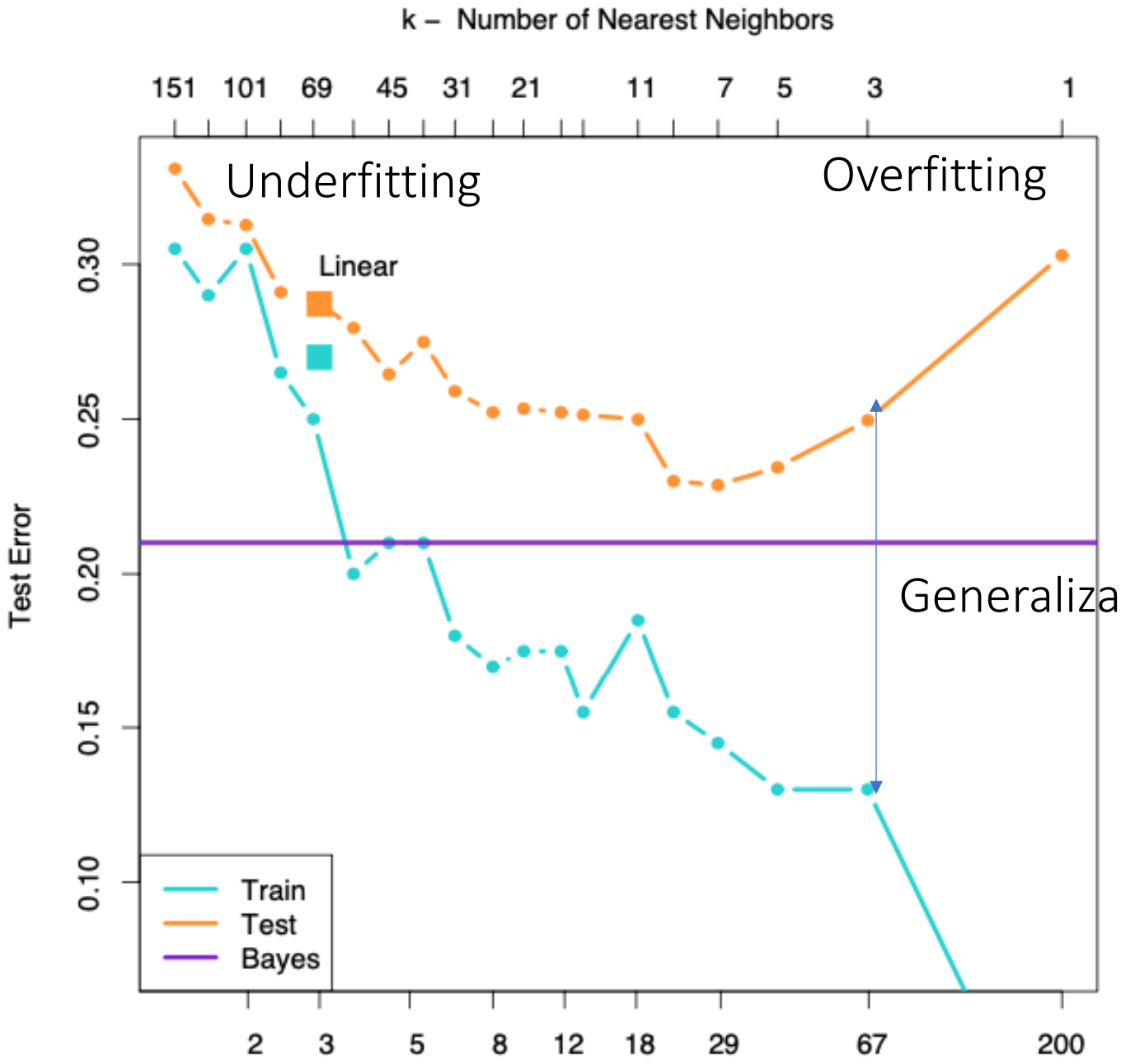
Hastie, ESL, Ch 2



Models that can be perfect on the training data fail miserably on data that the model has not seen.

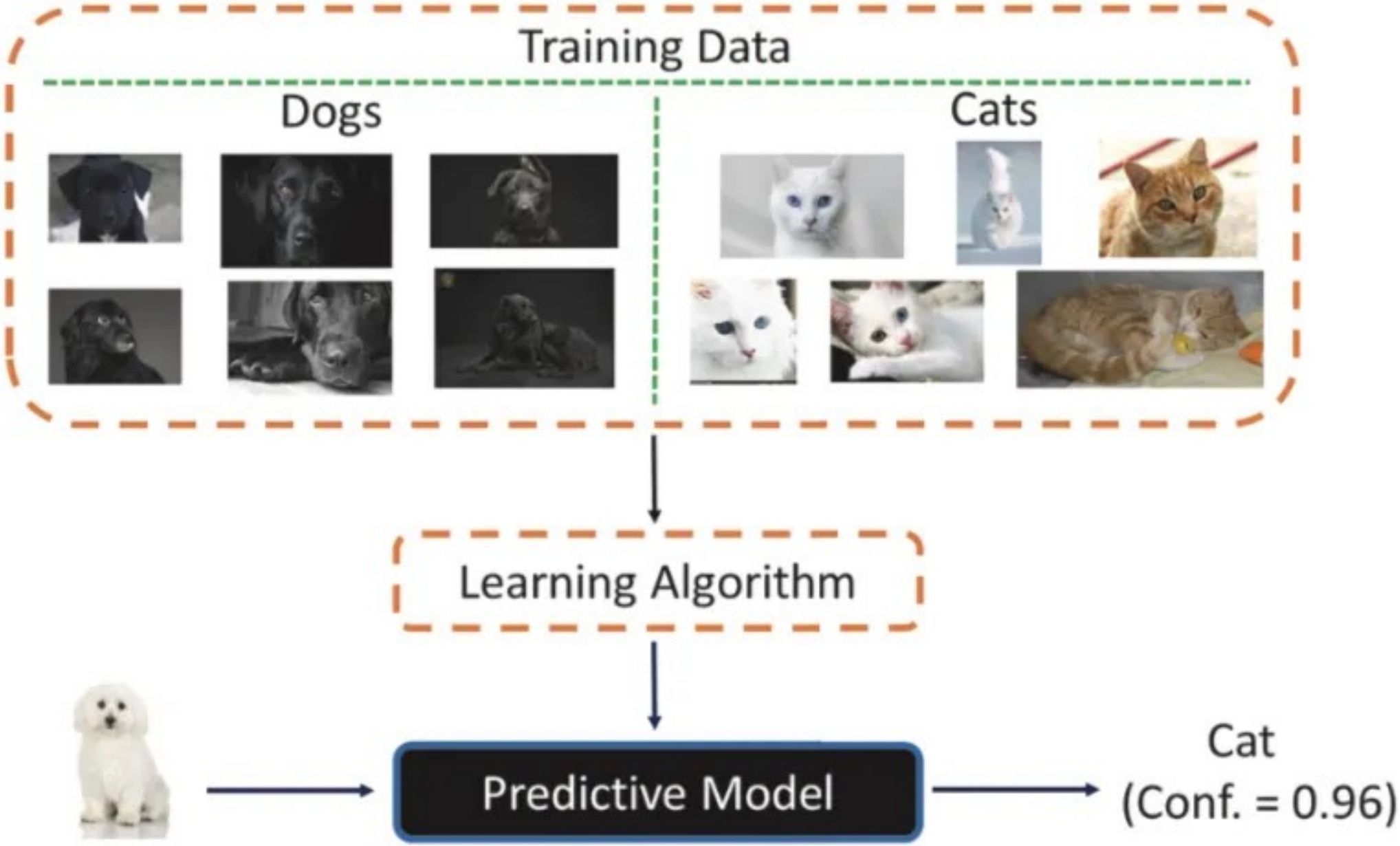
Hastie, ESL, Ch 2

Generalization error



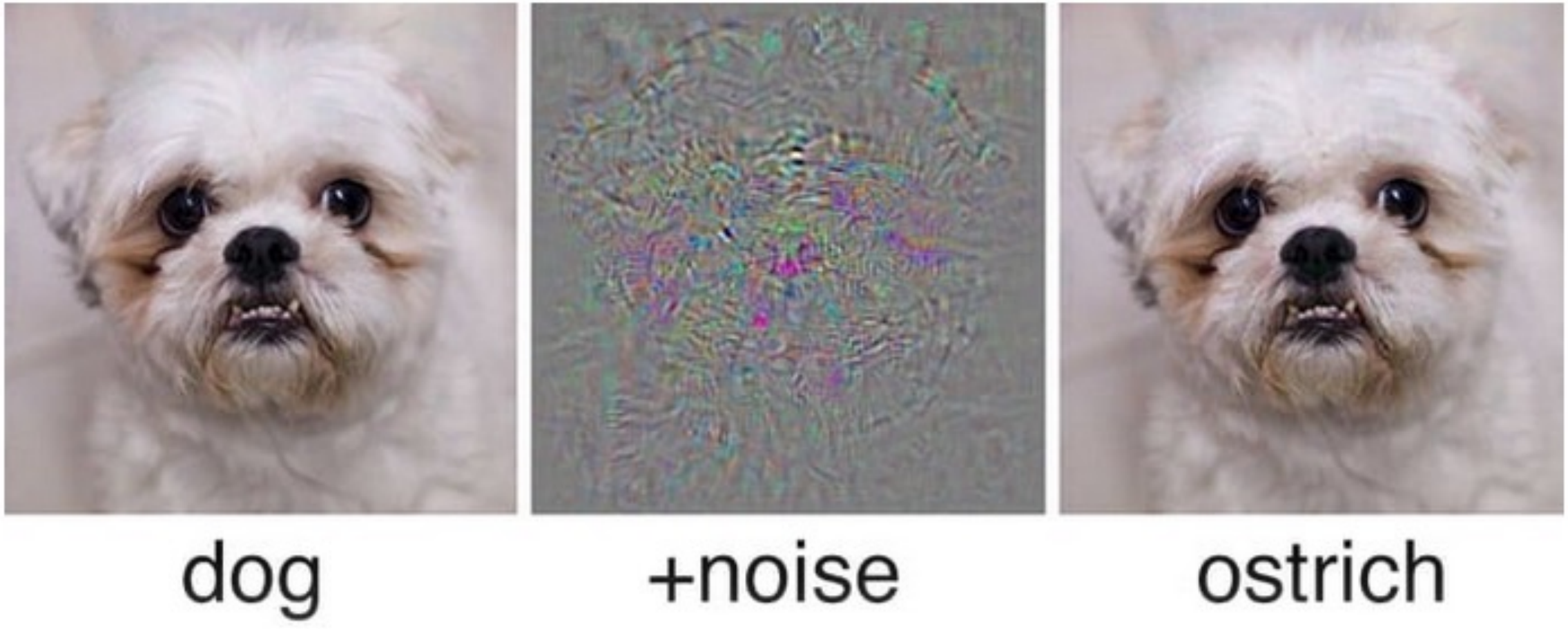
Generalization error

Hastie, ESL, Ch 2

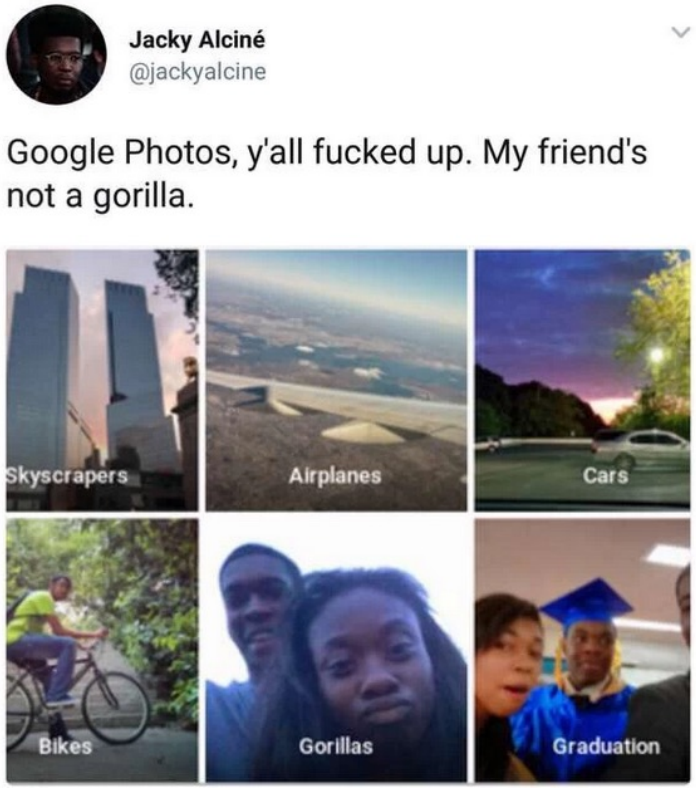

Training Data
Dogs
Cats
Learning Algorithm
Predictive Model
Cat
(Conf. = 0.96)

ML systems don’t make mistakes the way that we do…

Maliciously constructed inputs can lead machine-decision makers astray



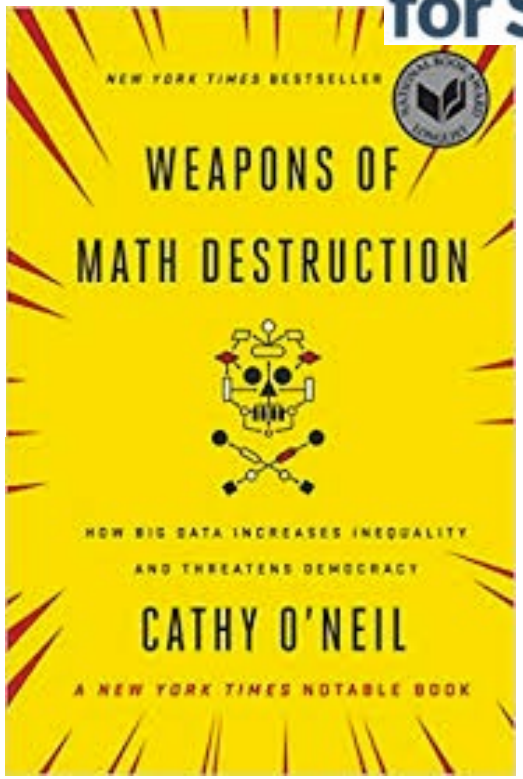
Data, ML systems are tools that benefit some more than others



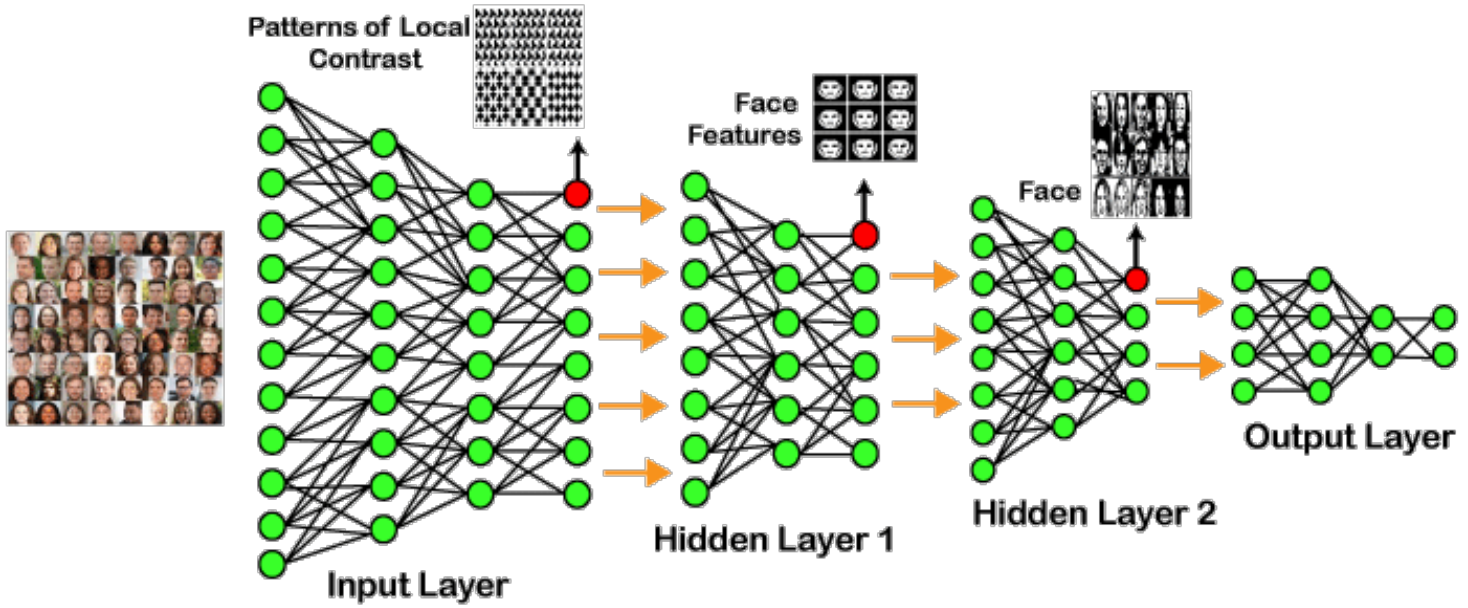
Researchers Tape Speed Limit Sign to Make Teslas Accelerate to 85 MPH

By Ryan Whitwam on February 19, 2020 at 1:01 pm | Comments

Data Science for Social Good

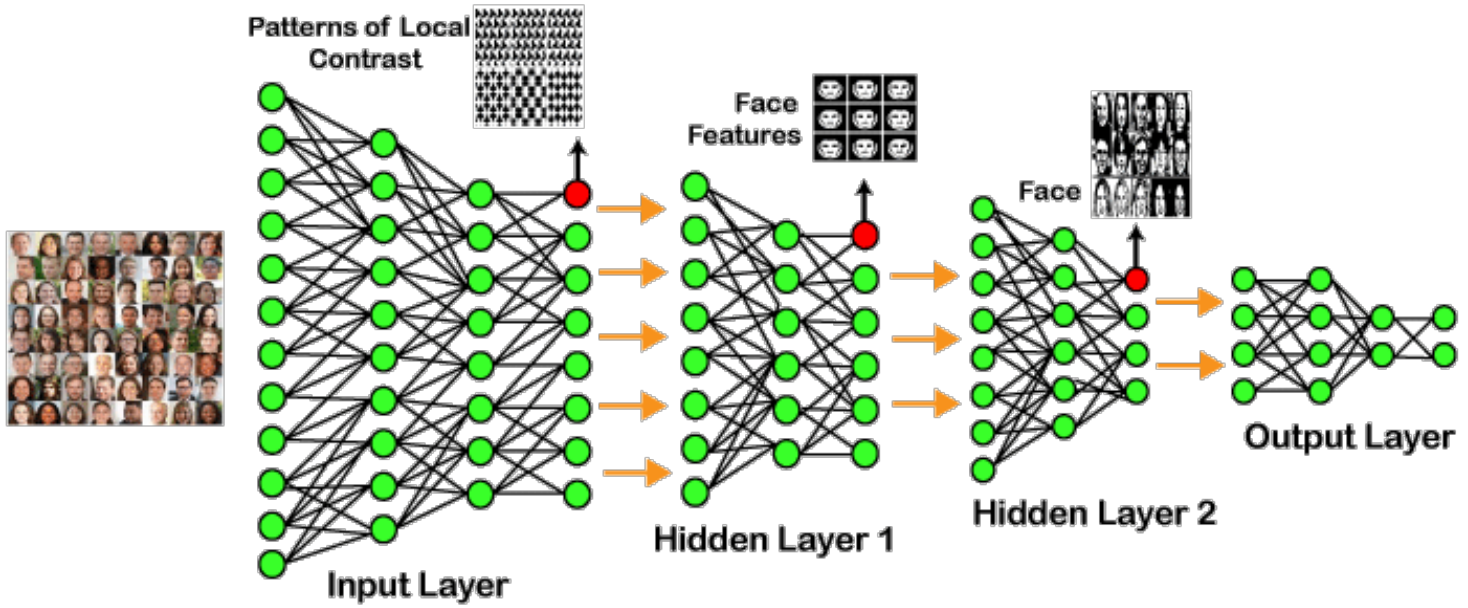


What is good?



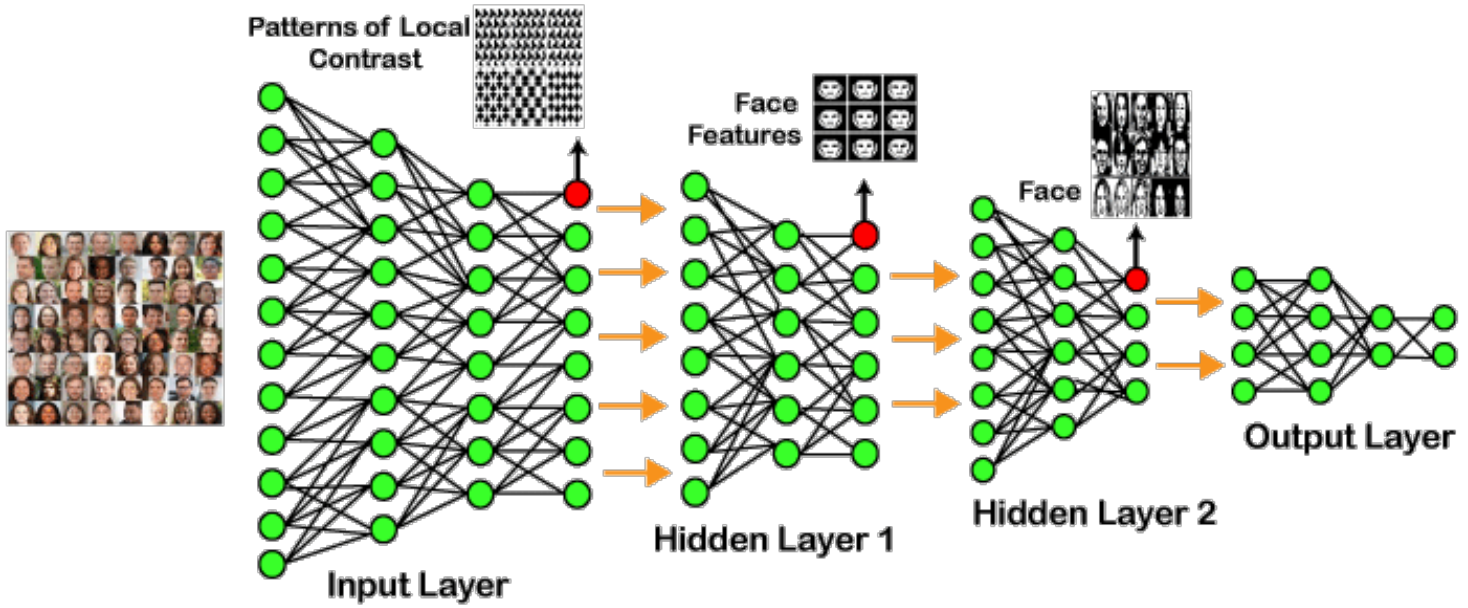
- What makes a good algorithm / model / procedure?

What is good?



- What is a good algorithm / model / procedure?
- Accuracy on out-of-bag data?

What is good?



- What is a good algorithm / model / procedure?
- Accuracy on out-of-bag data?
- Desired properties: train / retrain without great expense?
- Desired properties: return answers extremely quickly?
- Desired properties: generalize to data that perhaps aren't like the training data ???
- Desired properties: get good accuracy with comparatively little training data. !!!

Where do advances come form?

- The past 10 years has seen remarkable advances in computational decisionmaking. Where do these come from?
  - More data + more compute (Moore's law + AWS)
  - Better models: models that are well-adapted to the data can get better accuracy for the same amount of training data. (CNN, BERT)
  - New approaches (Generative Adversarial Networks; reinforcement learning)

The toolbox

The toolbox has different kinds of things.

“Traditional” algorithms that work (E-M, backpropagation, simulated annealing)

Learning procedures sometimes correspond to Bayesian inference. Often can only draw samples via Markov Chain Monte Carlo sampling.

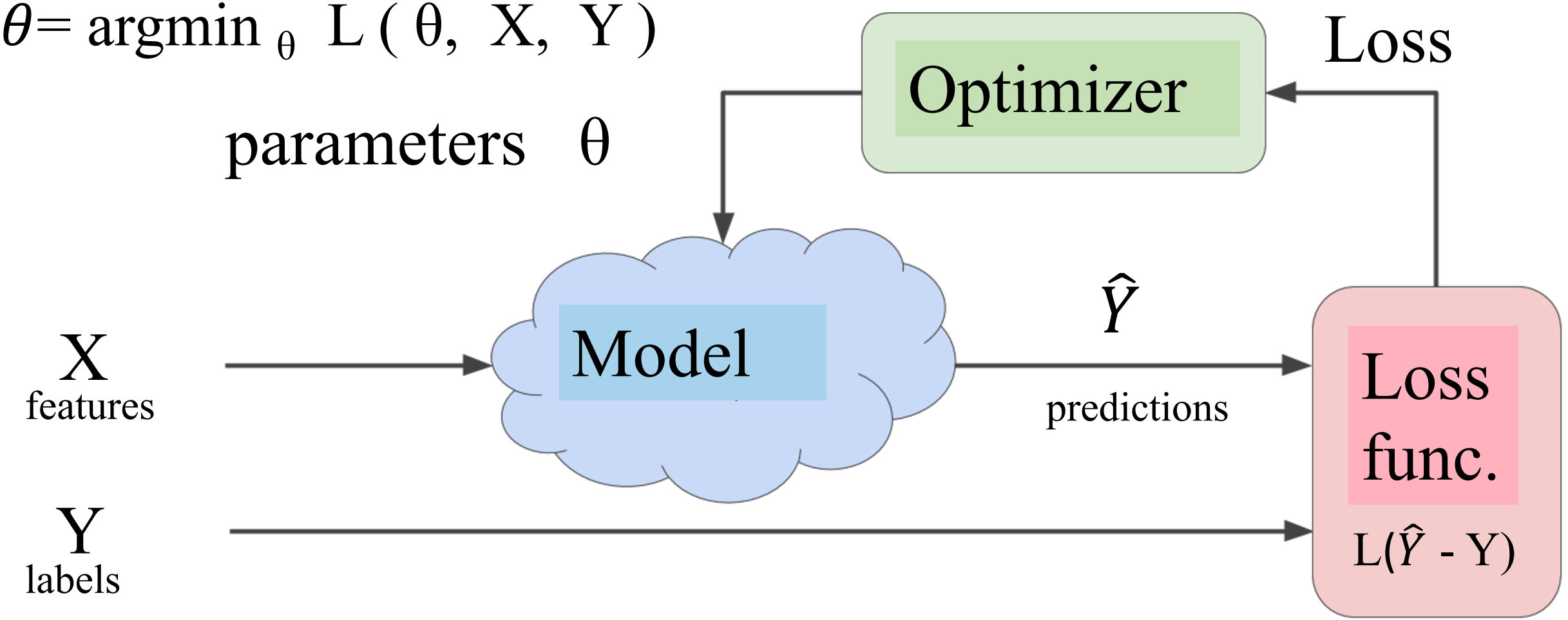
“Nonparametric” methods that look a little like adhockery

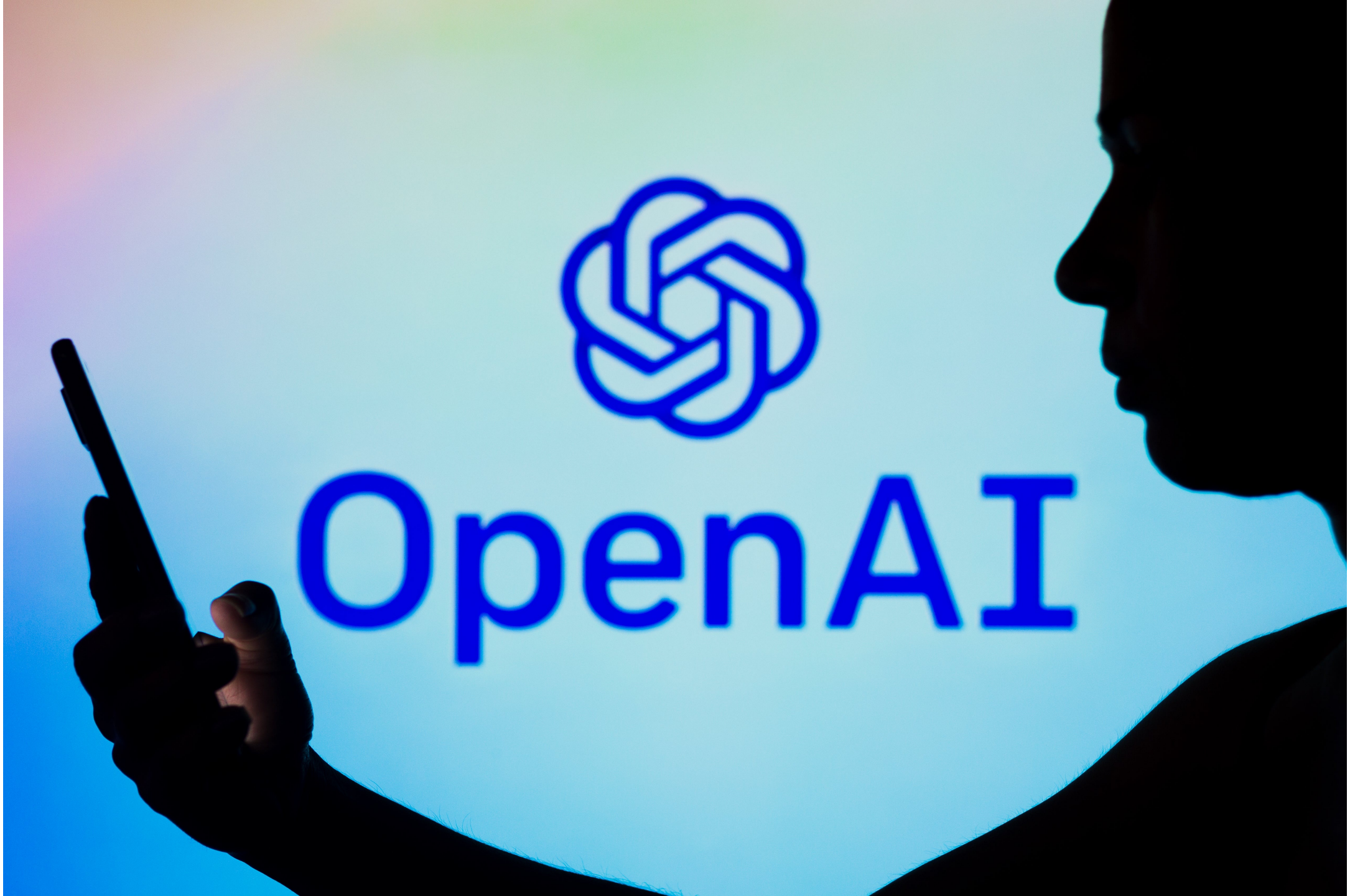
Optimization, stochastic gradient descent

Fairy dust picture of optimization

$$\hat{\theta} = \text{argmin}_{\theta}\ L(\theta, X, Y)$$

parameters θ

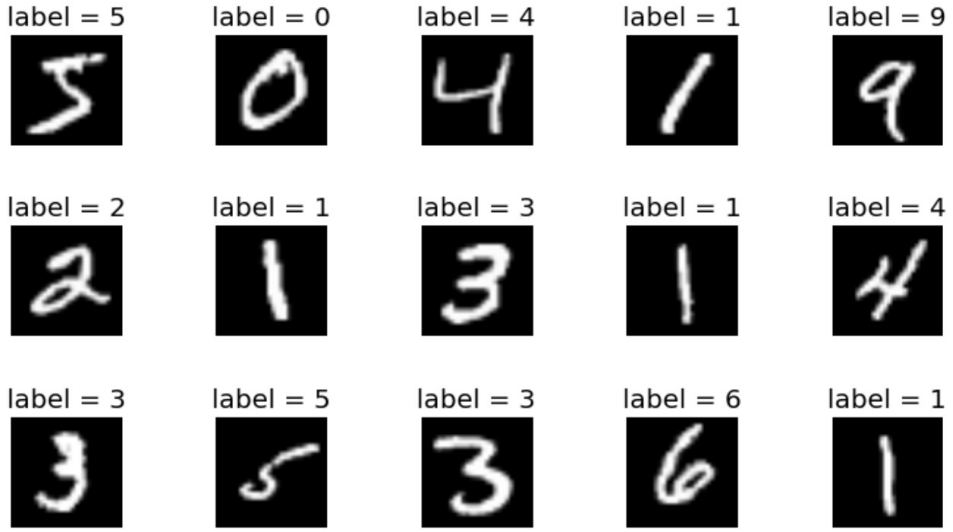



OpenAI

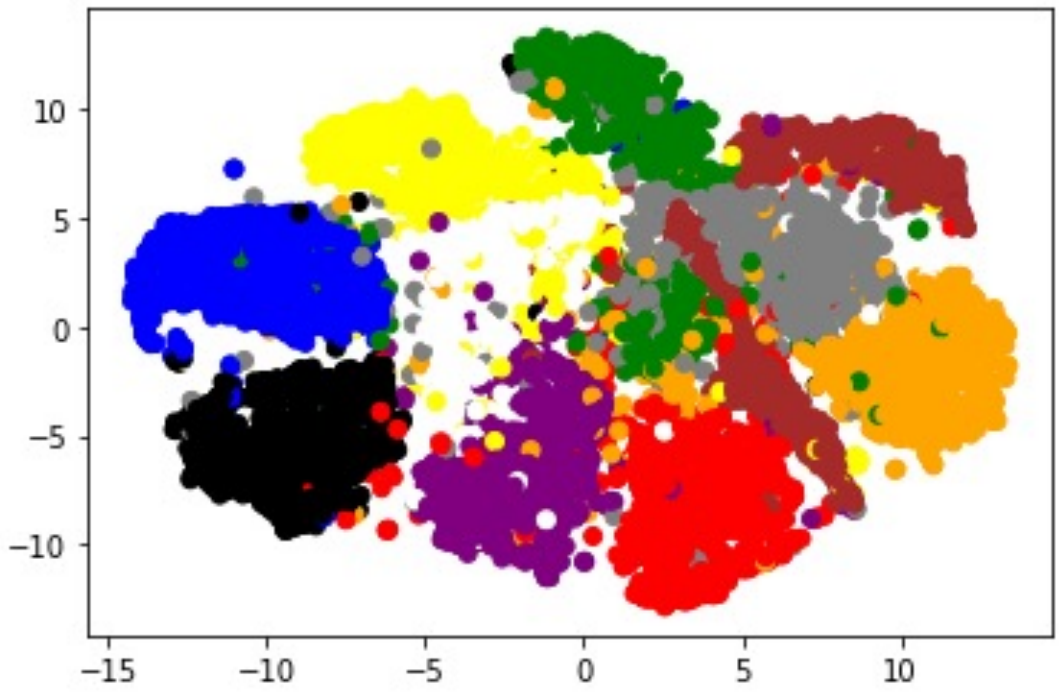
The Plan

- Parameter estimation
- Linear classifiers
- Nonparametric methods, clustering
- Optimization
- Evaluation, choosing algorithm
- Overfitting and regularization
- Neural networks
- Intro Deep Learning

- I will shoot for half math, half hacking, mostly with sklearn



MNIST data set



HW1

Exercise 3.3.[3, p.48] Inferring a decay constant

Unstable particles are emitted from a source and decay at a distance x, a real number that has an exponential probability distribution with characteristic length λ. Decay events can be observed only if they occur in a window extending from $x = 1\,\text{cm}$ to $x = 20\,\text{cm}$. N decays are observed at locations $\{x_1, \ldots, x_N\}$. What is λ?