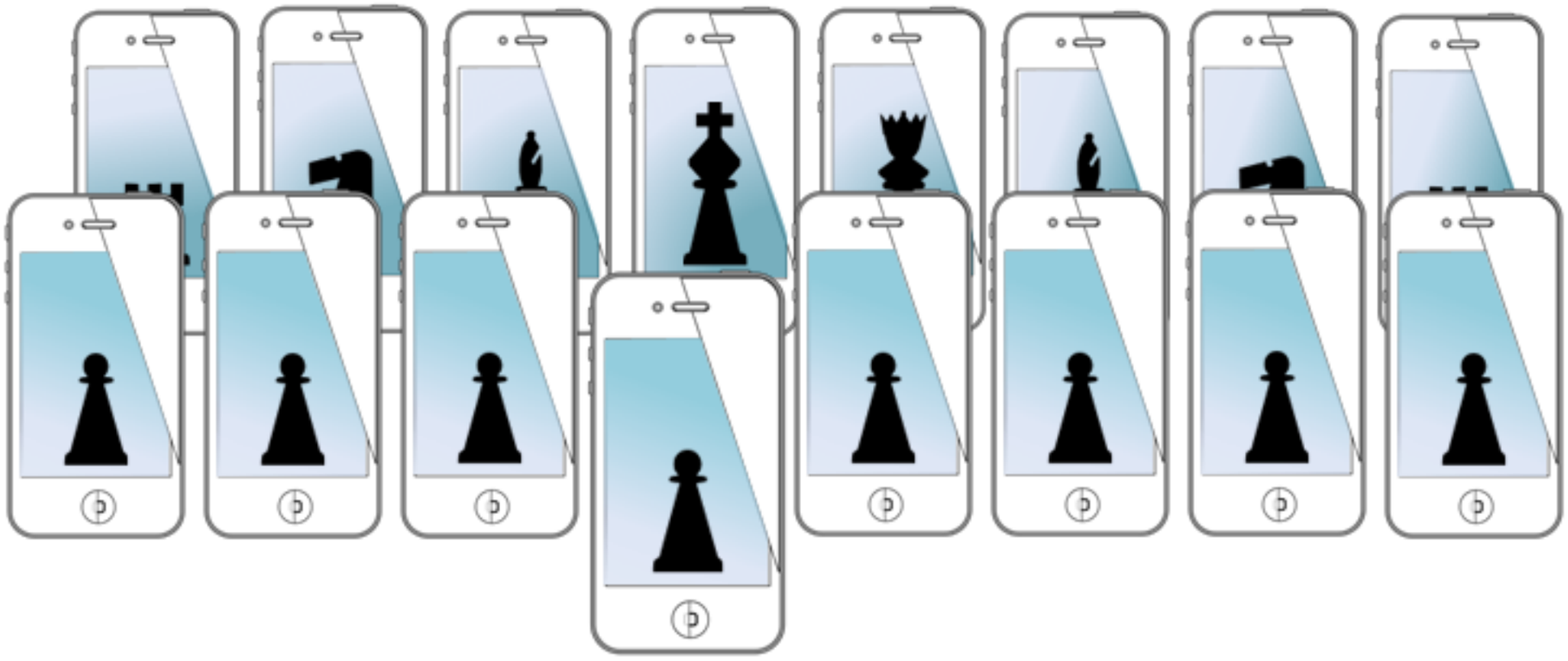
MOBILE SENSING LEARNING



CS5323 & 7323

Mobile Sensing and Learning

machine learning crash course

Eric C. Larson, Lyle School of Engineering,
Computer Science, Southern Methodist University

course logistics

- **A4** is due this week
- **A5** is due in two weeks
  - you can run server from your laptop (or cloud)
  - try to make it a first draft of your final project
    - seriously—make things easier on yourself
- **final project proposal** is also due at same time
  - feel free talk to me about it

assignment 5

- try to make this the first iteration of the final project!

Assignment Five - Machine Learning as a Service

Module A

Create an iOS application using the HTTPSwiftExample that:

- Collects some form of **low throughput (sampling rate > 1s)** feature data for processing: audio, video, motion, or from the micro-controller
- Uploads **labeled feature data** to a server via HTTP POST requests
  - you can run the server from your laptop or mac mini
  - Alternatively you can use a virtual machine, AWS, or other cloud service
- Trains a model from the labeled data (e.g., KNN, SVM, Random Forest, etc.)
- **Requests predictions** from the server by uploading unknown feature vectors
  - can be periodically or initiated by user
- Note that the server code given to you will automatically save any feature data you upload and train a machine learning model, given the correct POST/GET request commands

You should not need to update the server for any of the given functionality. However, the predictions from the server may not be sufficient without updating the training parameters or the type of model used. Verify the functionality of the application to the instructor during lab time or office hours (or scheduled via email).

assignment 5

- try to make this the first iteration of the final project!

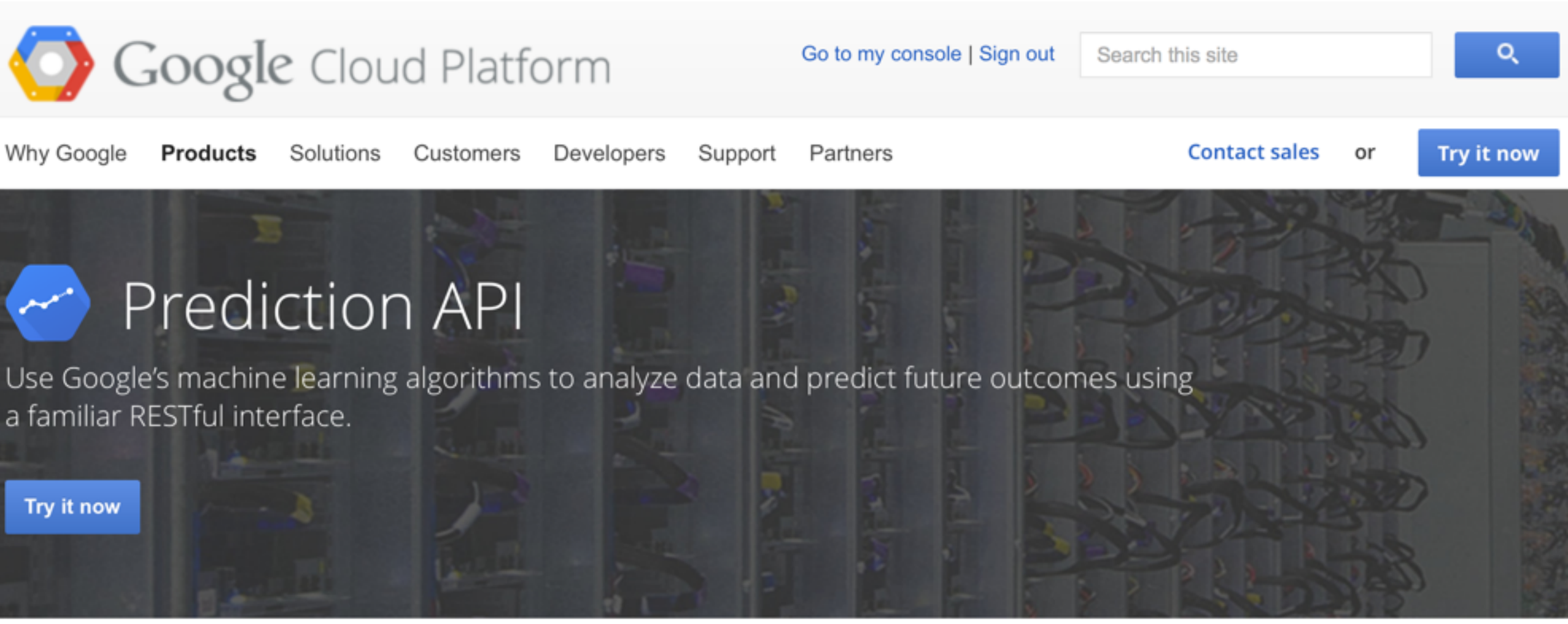
Assignment Five - Machine Learning as a Service

Module B

Update the HTTPExample and the tornado web server to:

- Specify the type of model to use in the Machine Learning
  - at least **two different types** of machine learning models (***e.g.***, SVM and KNN)
- Compare the efficacy of two or more different models
  - **send parameters** to use in the machine learning models from the phone (***e.g.***, number of neighbors to use in KNN)
- **Exceptional Work:** 7000 Level Students Choose ONE of the following:
  - make the training of the model non-blocking to the tornado IOLoop
    - (requires co-routines and manipulation of sklearn fit function)
  - implement **authentication** in tornado and in your iOS application
    - cookie secret is fine
    - or third party
  - Use CoreML to export your custom trained machine learning model and run the machine learning prediction locally on the iOS app (NOTE: the **CoreML model must be exported from the data you create** on your HTTPServer)

machine learning



machine learning

BigML

1

Set up a Source

To create a new source, just drag and drop your data file onto BigML's interface, or select the file you want to use with the **upload** icon. Sources can be created from almost any tabular data (csv or arff files). You can gzip (.gz) or compress (**.bz2**) them to save bandwidth. You can also create sources from remote locations using protocols such as **HTTP(S)**, **s3**, **azure**, or **odata**. You can upload files of up to **64GB** or up to **5TB** if you use remote S3 buckets. Once your source has been created, you can use a configuration panel to update types, names, **labels**, **descriptions**, and other parsing preferences.

How to create a source

3

Create a Model

To create a model, just use the **1-cl**
If you want to use all active fields an
generate a model that predicts (i.e.,
column in the input data. You can al
panel to select a different **objective**
your dataset using multitude of opt
processed to build a predictive mod
the most relevant patterns in your d
generate predictions for new data in

How to create a model

2

Create a Dataset

To create a new dataset, just use the **1-click dataset** button from a source view if you want to include all the fields and the complete source, or use the **configure dataset** panel to select a few specific fields or limit the total size of data to analyze. BigML will start computing the distribution of values for each of the fields in your dataset. This process can take from a few seconds to a few hours depending on the size of your data. As your dataset is being created, BigML shows you a visualization that gives you immediate feedback about your data.

How to create a dataset

4

Generate Predictions

To create a prediction, you can eith
generated web form or a question h
click the **predict** or **predict questio**
presented with the corresponding i
of input fields is big or you want to
predictions you can use our API. Sta
High Performance Prediction Server
generate thousands of predictions

How to make predictions

agenda

- intro to machine learning
- numpy, scipy, and turi-create
    - using iOS with python:
        - applepy?
- by the end of this lesson I want you to know:
    - what is machine learning
    - know the keywords
    - know enough about ML toolkit to approach it
    - lose whatever barrier you had to ML

machine learning

ma·chine

/məˈSHēn/

noun

noun: **machine**; plural noun: **machines**

1. an apparatus using or applying mechanical power and having several parts, each with a definite function and together performing a particular task.
   "a fax machine"
   synonyms: apparatus, appliance, device, contraption, contrivance, mechanism, engine, gadget, tool More
   - a coin-operated dispenser.
     "a candy machine"
   - *technical*
     any device that transmits a force or directs its application.

learn·ing

/ˈlərniNG/

noun

noun: **learning**

1. the acquisition of knowledge or skills through experience, study, or by being taught.
   "these children experienced difficulties in learning"
   synonyms: study, studying, education, schooling, tuition, teaching, academic work; More
   - knowledge acquired through experience, study, or being taught.
     "I liked to parade my learning in front of my sisters"
     synonyms: scholarship, knowledge, education, erudition, intellect, enlightenment, illumination, edification, book learning, information, understanding, wisdom More
     antonyms: ignorance

teach a computer to learn something
through experiences

the hierarchy

One Small Piece of Artificial Intelligence

Data Mining

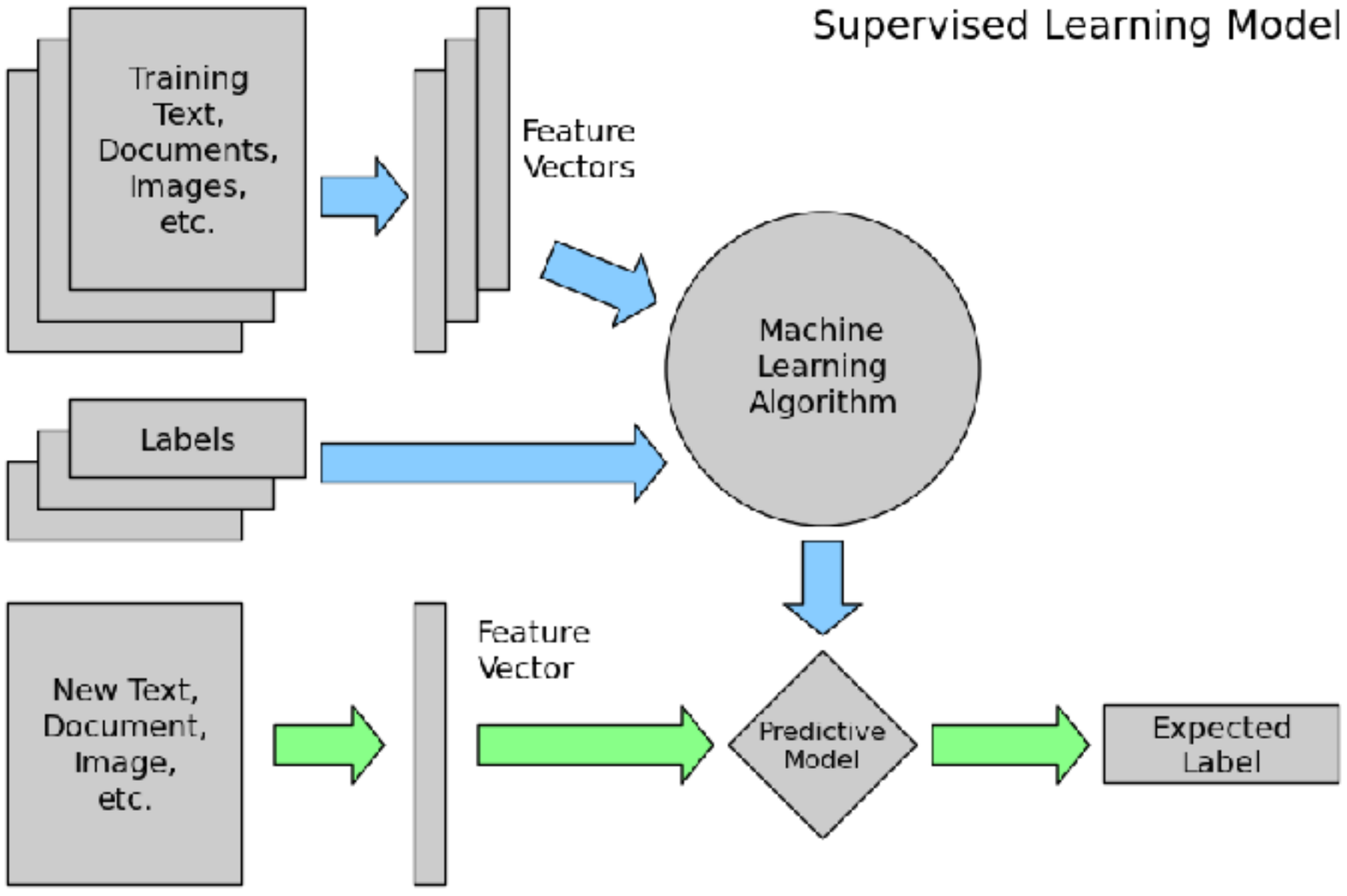
ML

- Prediction Methods
  - Use some variables to predict unknown or future values of other variables
- Description Methods
  - Find human-interpretable patterns that describe the data.

ML

- Classification
- Regression
- Deviation Detection
- Clustering
- Association Rule Discovery
- Sequential Pattern Discovery

machine learning models



- *Training* Instances: Features + Labels
- Find a *model* mapping class from values of features.
- Goal: Assign guessed label to previously unseen instances

types of data

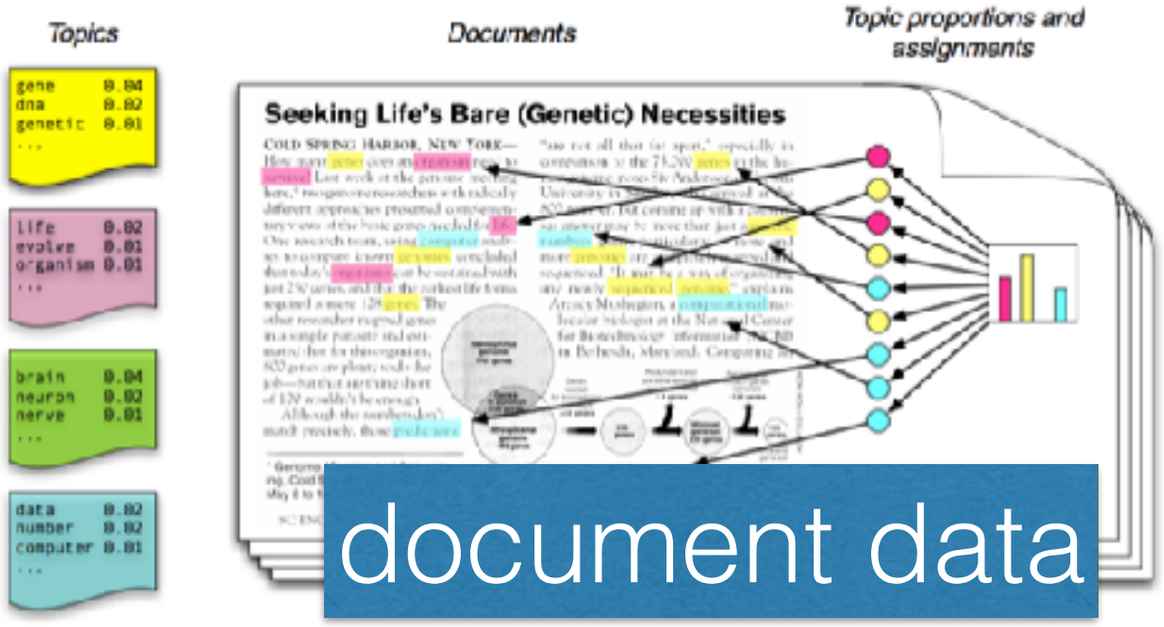


Figure source: Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4), 77-84.

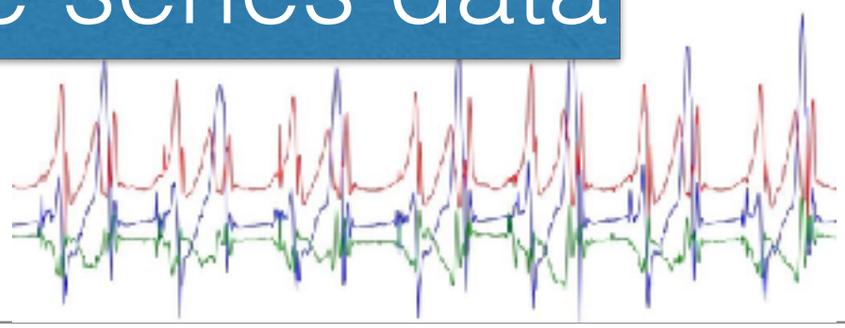
table data

Attributes, columns, variables, fields, characteristics, **Features**

Objects, records, rows, points, **samples**, cases, entities, **instances**

TID	Pregnant	BMI	Age	Diabetes
1	Y	33.6	41-50	positive
2	N	26.6	31-40	negative
3	Y	23.3	31-40	positive
4	N	28.1	21-30	negative
5	N	43.1	31-40	positive
6	Y	25.6	21-30	negative
7	Y	31.0	21-30	positive
8	Y	35.3	21-30	negative
9	N	30.5	51-60	positive
10	Y	37.6	51-60	positive

time series data



features and labels

- actually, feature vectors
- **classic example**: the iris dataset—table data

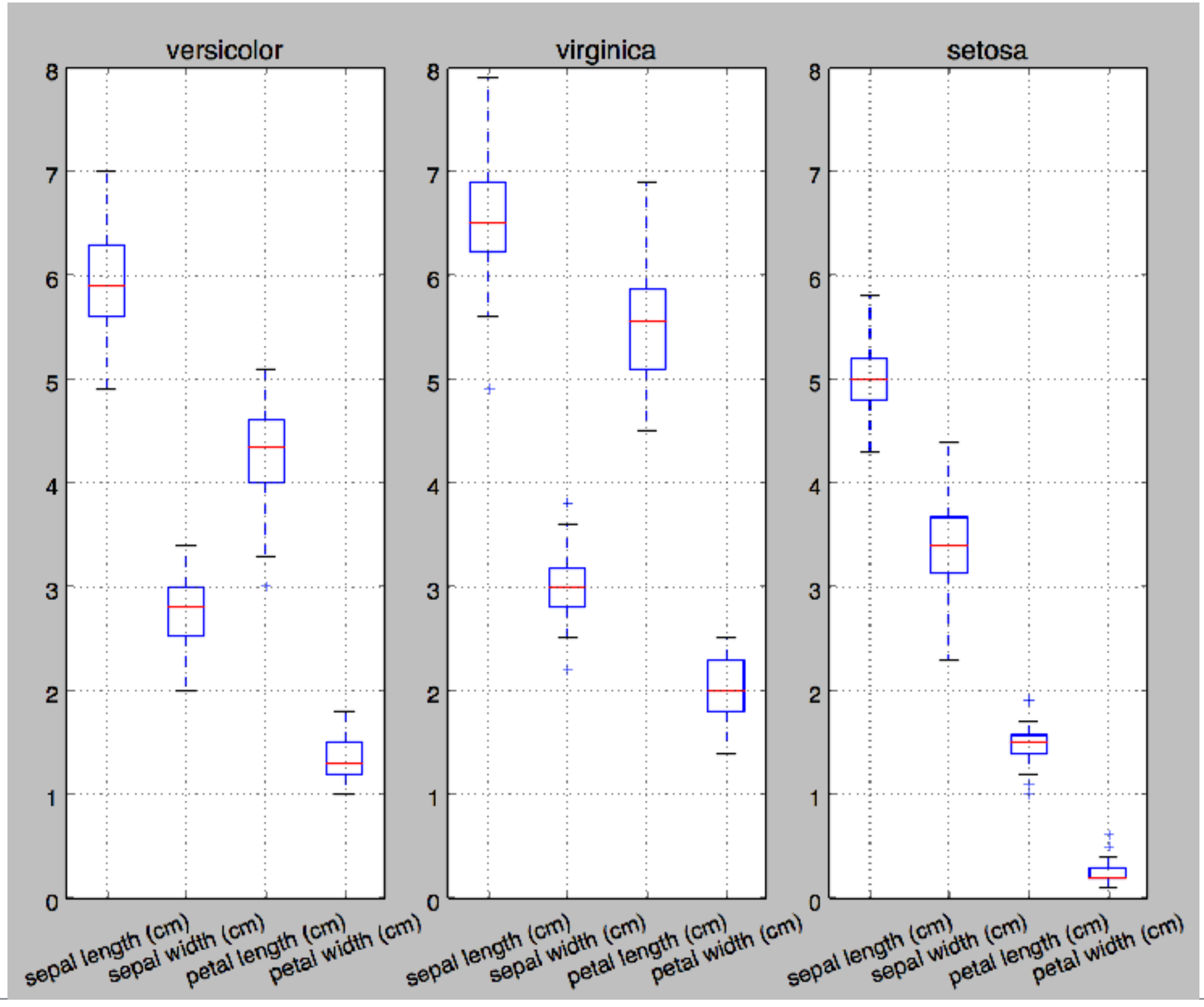
setosa versicolor virginica

- 4 features
  - sepal length in cm
  - sepal width in cm
  - petal length in cm
  - petal width in cm

```
[5.1,  3.5,  1.4,  0.2]  setosa
[5.7,  2.8,  4.5,  1.3]  versicolor
[7.6,  3. ,  6.6,  2.1]  virginica
```

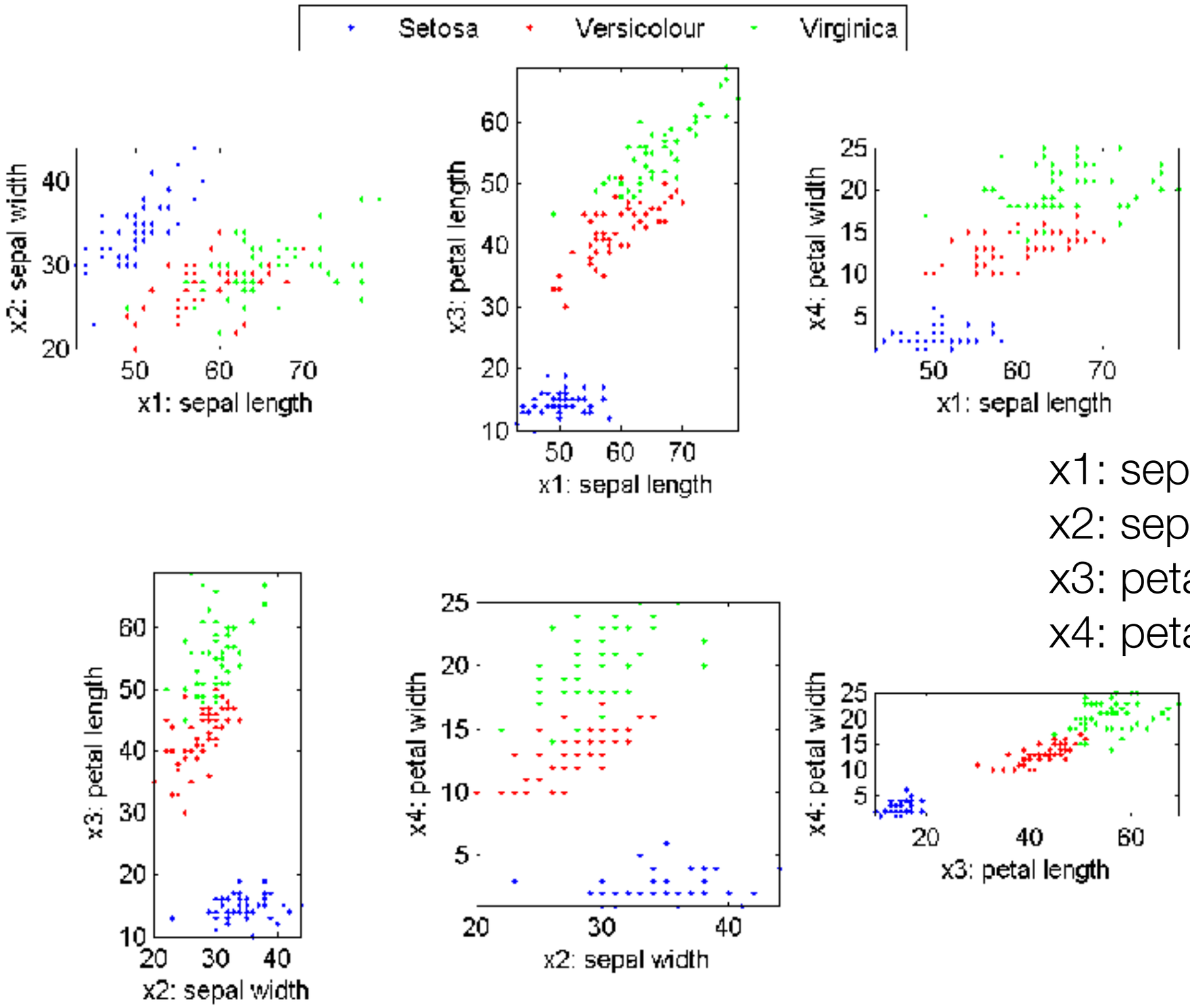
50 examples each

visualizing features



separability

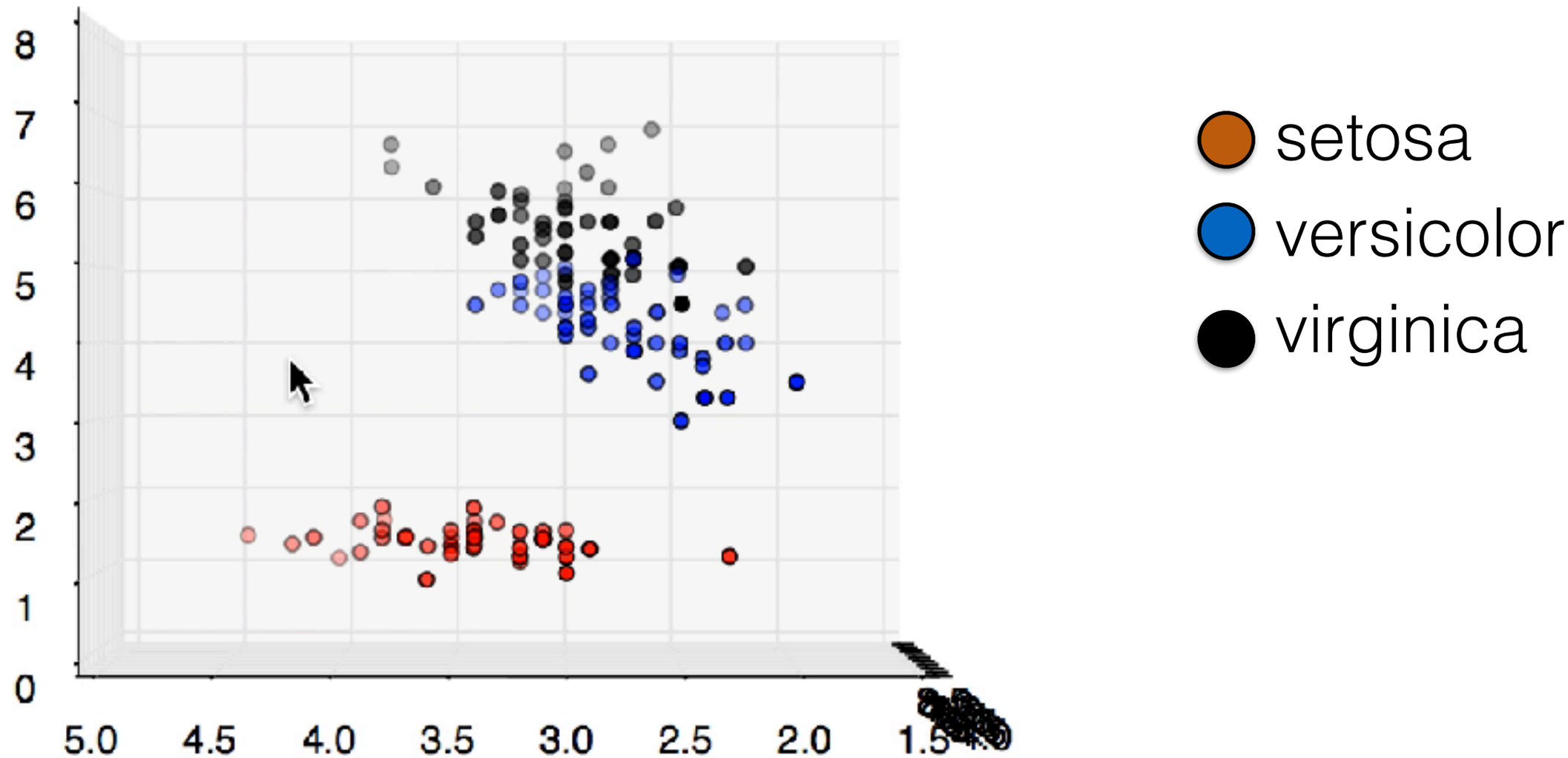
visualize features



x1: sepal length in cm
x2: sepal width in cm
x3: petal length in cm
x4: petal width in cm

image source:
mirlab.org

visualizing features



what about **four** dimensions?

features

- most common is numeric
- vector quantization
- bag of words
  - term frequency: percentage of a times a word appears in a type of document
  - inverse document frequency
- graphs
  - used to quantize

take data mining!
or python machine learning!

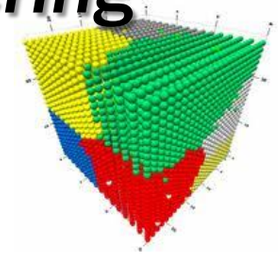
categorizing ML algorithms

- availability of labels
- structure of features and labels
- parameters saved after training

machine learning: categorize

- categorize by availability of labels

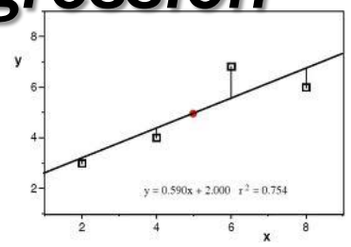
Clustering



unlabeled

unsupervised

Regression



Classification

Ordinal Reg.

labeled

supervised

machine learning: categorize

```
[5.1,  3.5,  1.4,  0.2]
[5.7,  2.8,  4.5,  1.3]
[7.6,  3. ,  6.6,  2.1]
```

setosa

versicolor

virginica

unstructured

predict *one*

Classification

The quick brown fox jumped over the lazy dog.

ia adj adj noun verb prep. ia adj noun

structured **predict an object of items**

machine learning: categorize

Classification

unknown

`[7.5, 3.7 , 5.0, 2.7]`

`*[1,0,1,1] =15.2`

setosa	`[5.1, 3.5, 1.4, 0.2]*[1,0,1,1] =5.7`	**<7**	
versicolor	`[5.7, 2.8, 4.5, 1.3]*[1,0,1,1] =11.7`	**else**	
virginica	`[7.6, 3. , 6.6, 2.1]*[1,0,1,1] =14.3`	**>13**	

store all labeled examples **nonparametric**

method 1: closest example in training set

store parameters: [1, 0, 1, 1] **parametric**

method 2: sum greater than value for each class

common ML algorithms

nonparametric

- nearest neighbor
- k-nearest neighbor (KNN)
- kernel density estimator

parametric

- decision tree
- random forest/boosted trees
- logistic regression
- neural networks
- gaussian mixtures

- support vector machines
- and many many more…

decision trees

- example via iris data

- **4 features**
  - sepal length in cm
  - sepal width in cm
  - petal length in cm
  - petal width in cm

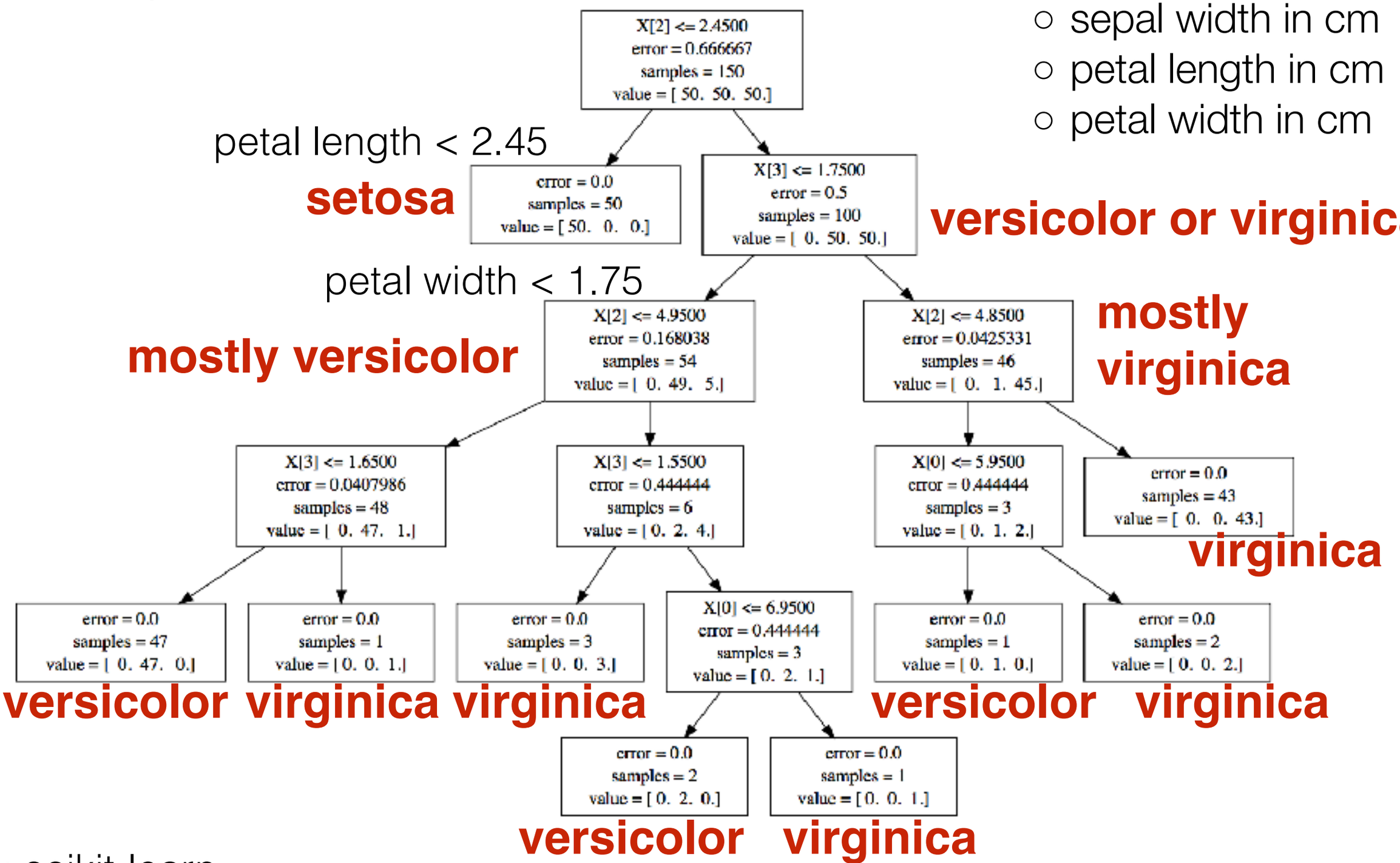
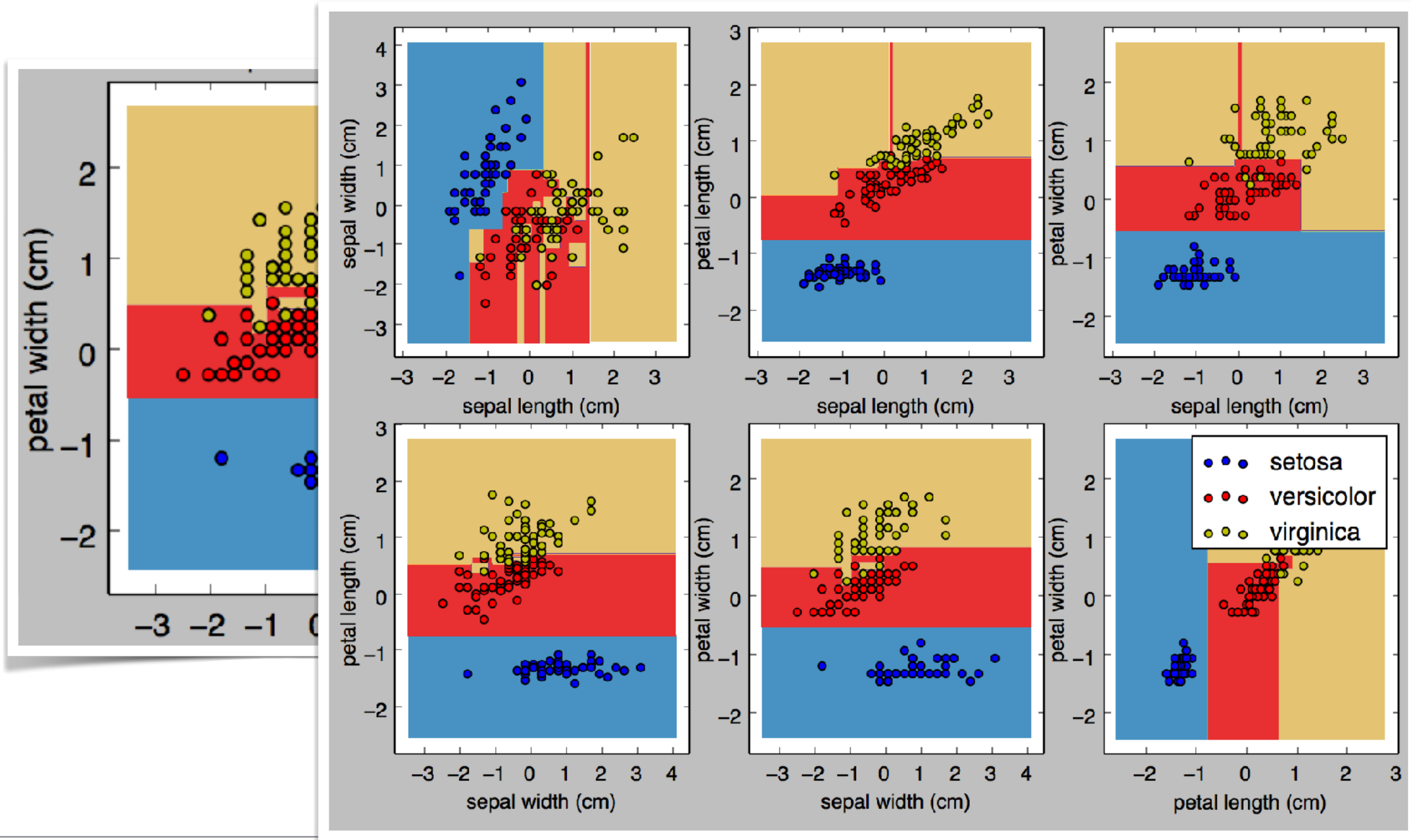


image: scikit-learn

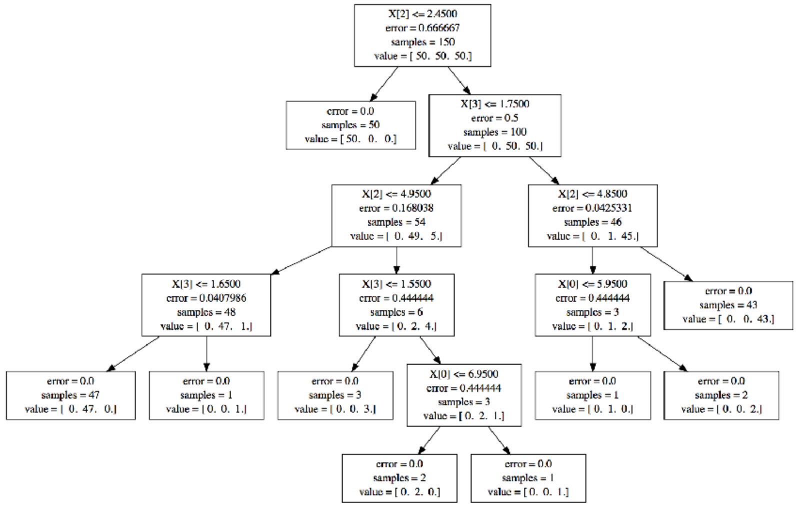
decision tree class boundaries



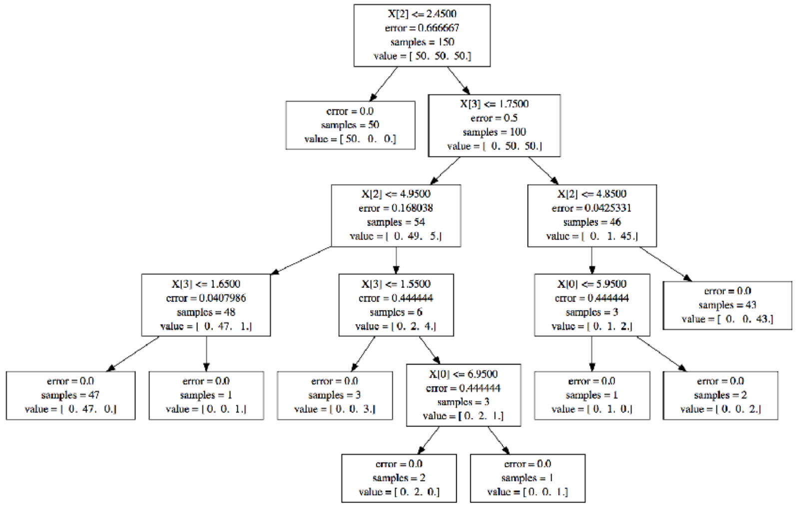
random forests

- make a bunch of trees
  - each tree made with a random subset all training data
  - and a random subset of the features

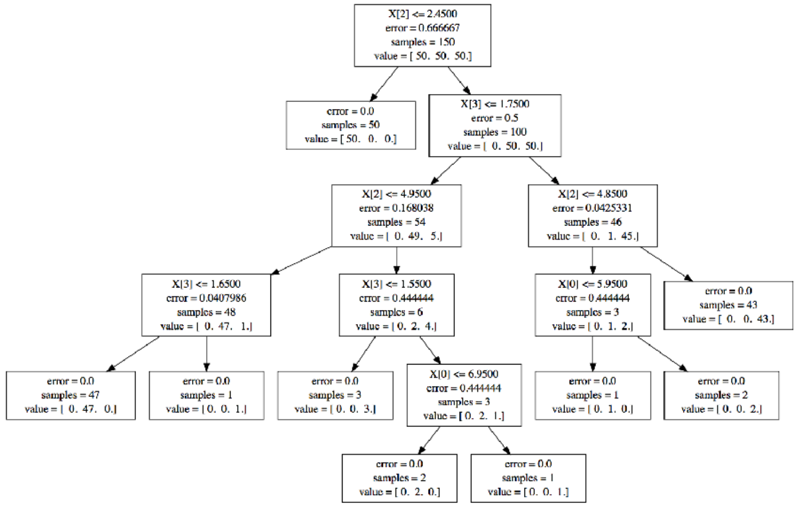
120 examples
3 features



120 examples
3 features



120 examples
3 features



x 200 trees

unknown vector?
query each tree

95% say setosa
4% say versicolor
1% say virginica

k-nearest neighbor

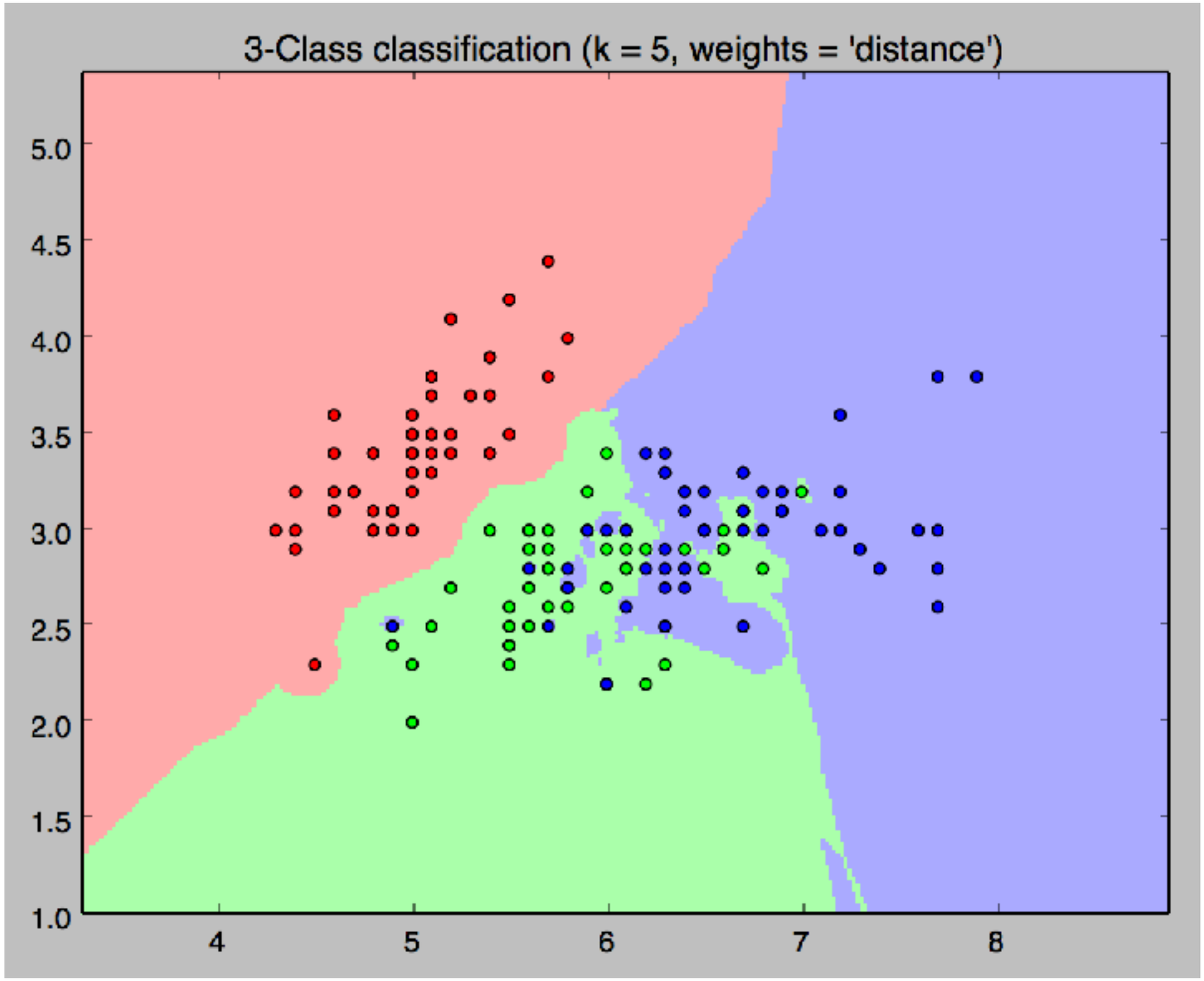
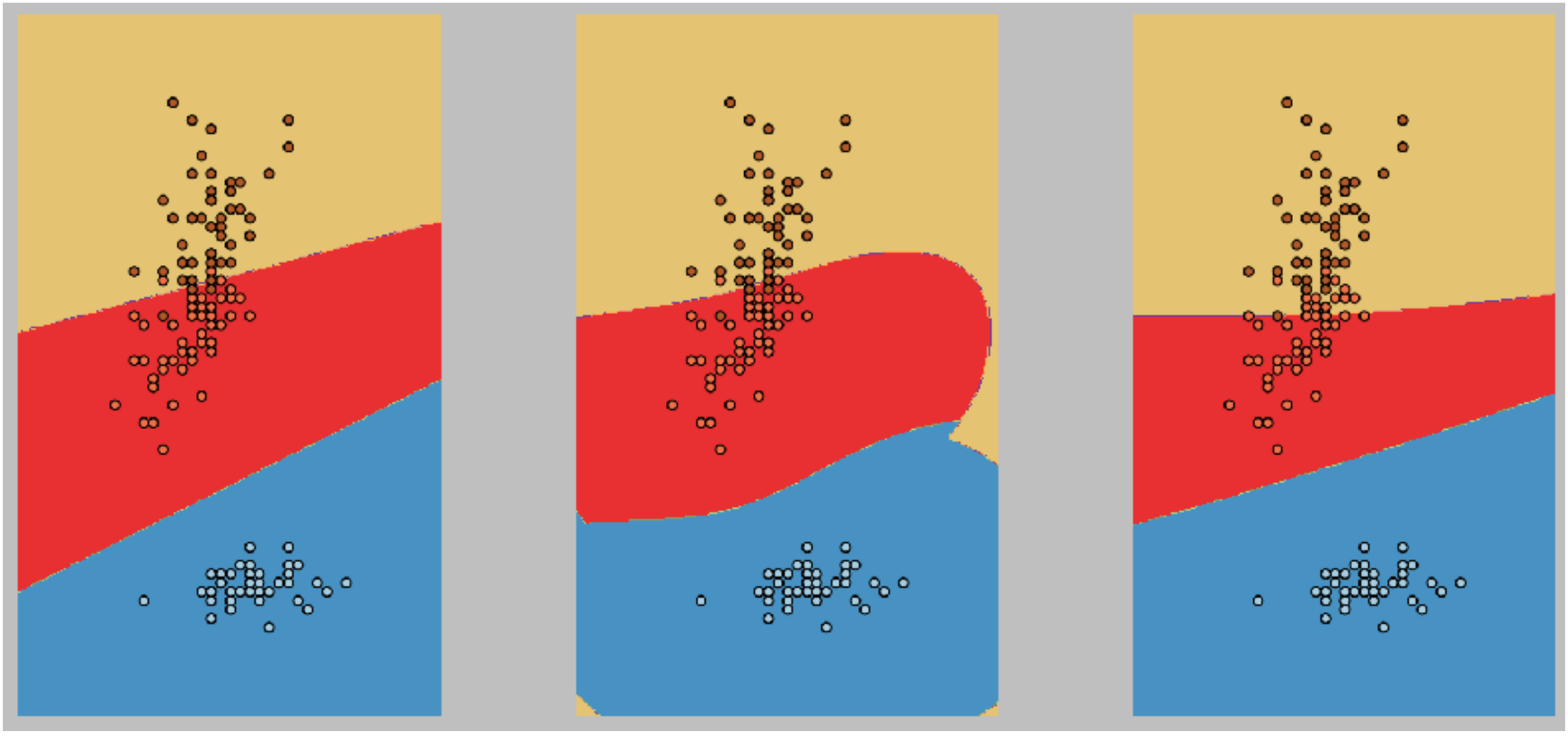


image:
datasciencerules.org

support vector machines

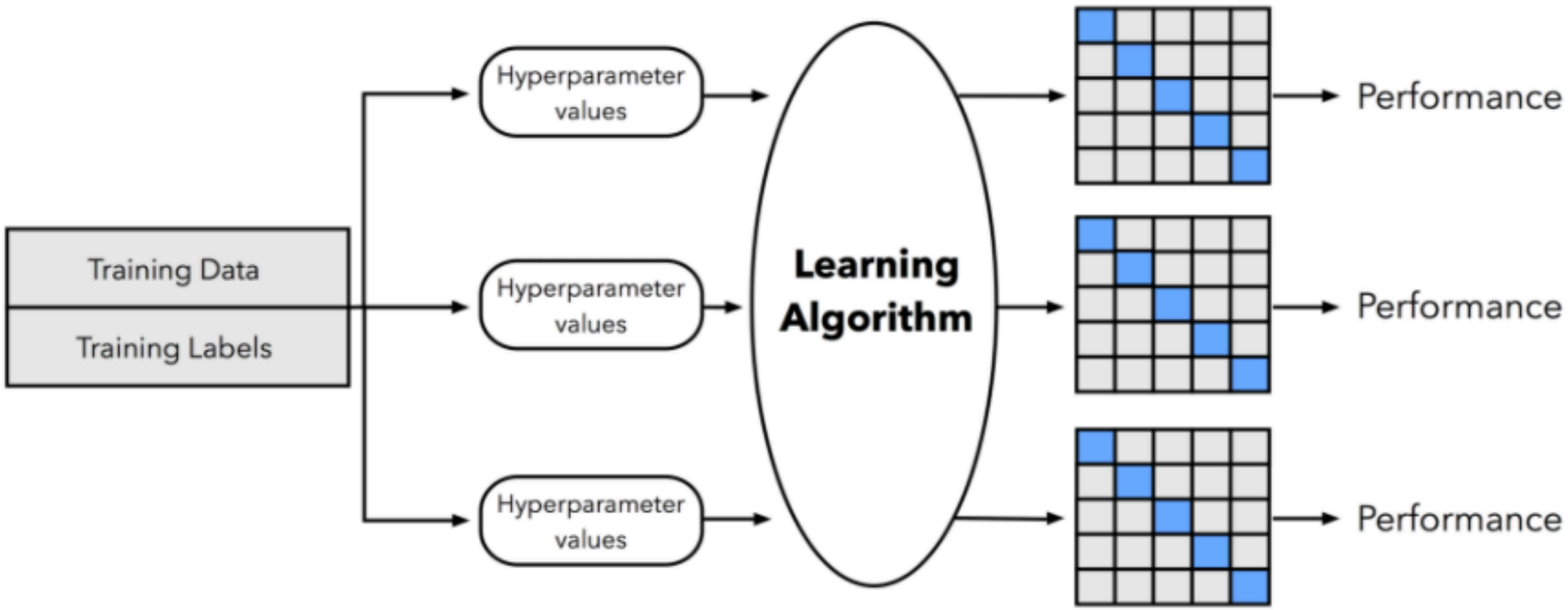
- find a vector that separates the training data



linear vector | gaussians | sigmoid

finding the best ML model

- try a bunch of stuff until it works well enough

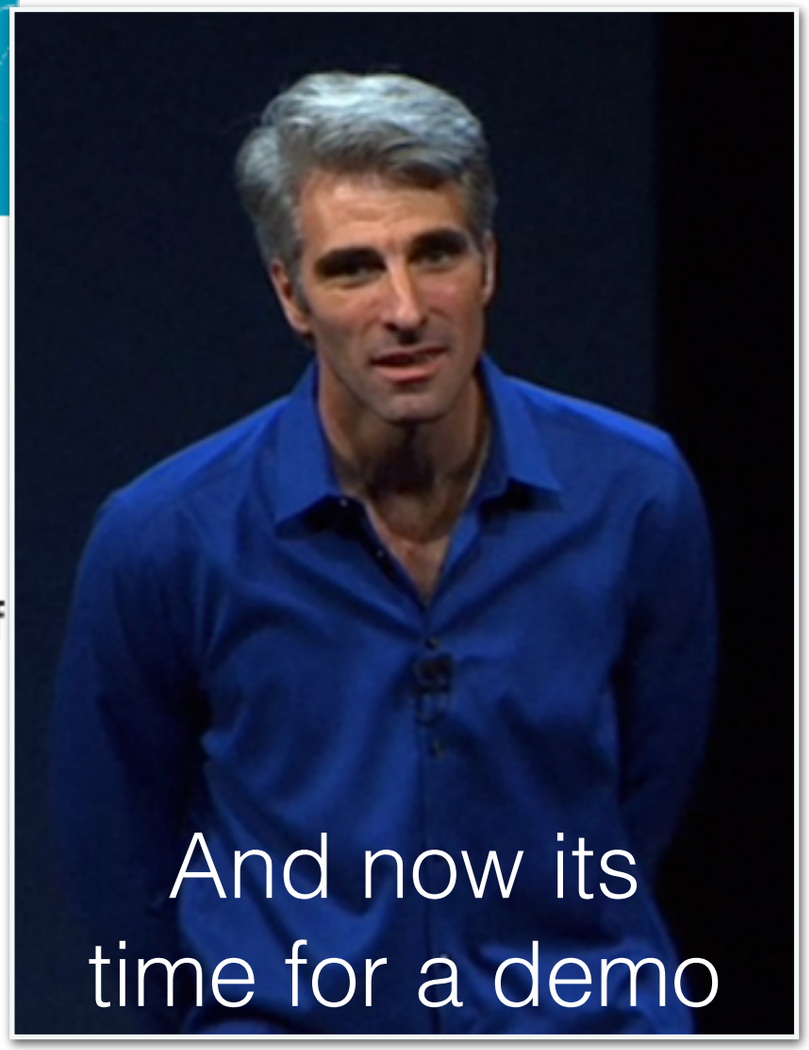


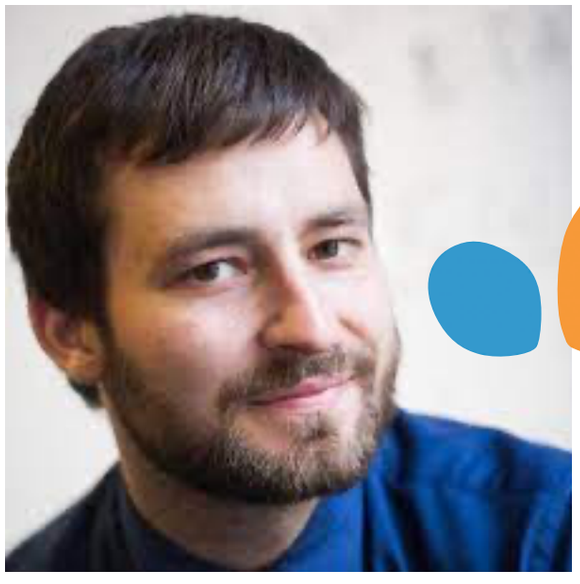
http://ethen8181.github.io/machine-learning/model_selection/model_selection.html

turi create and scikit-learn

Carlos Guestrin · 2nd

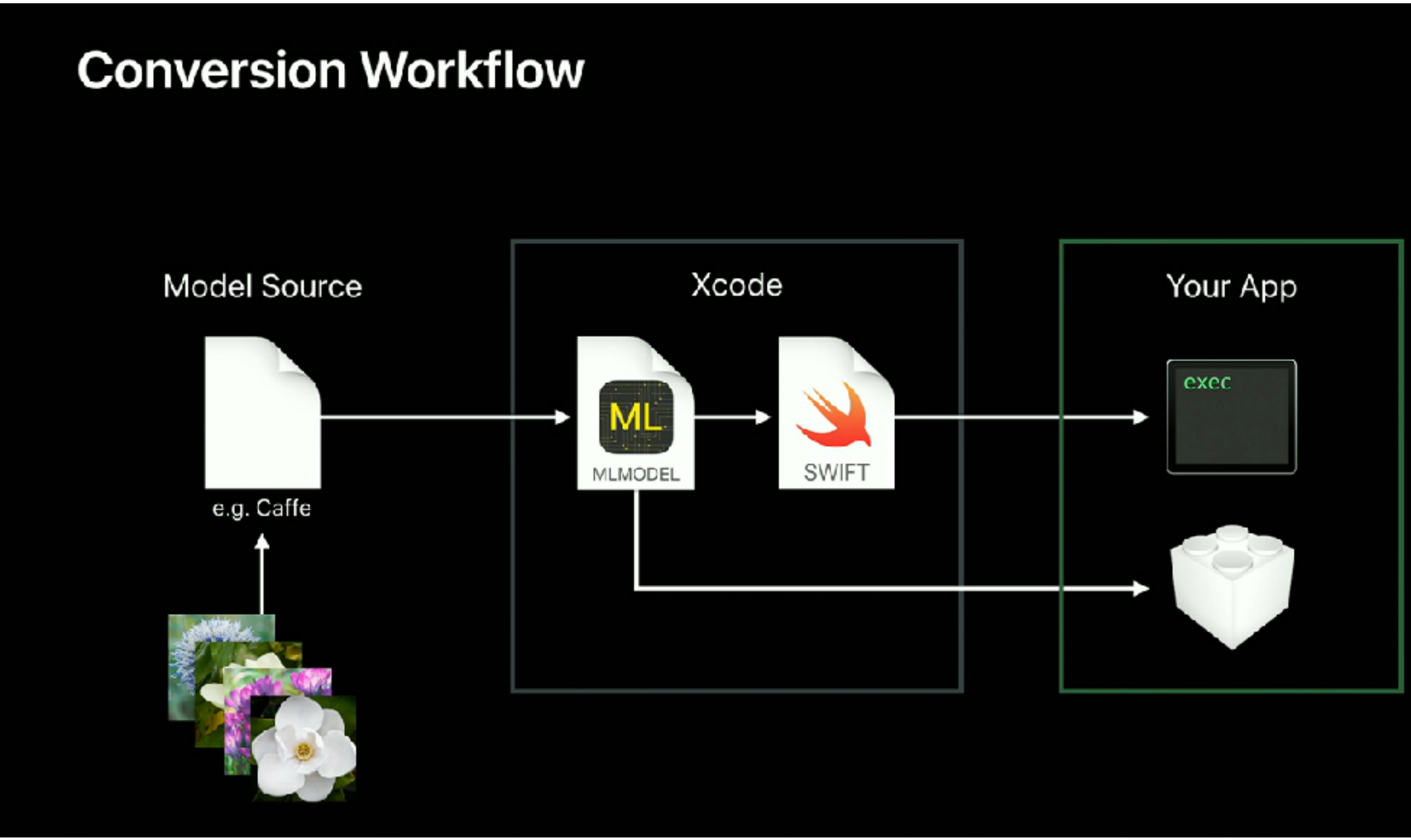
Senior Director of AI and Machine Learning at Apple & Amazon Professor of Machine Learning at University of Washington



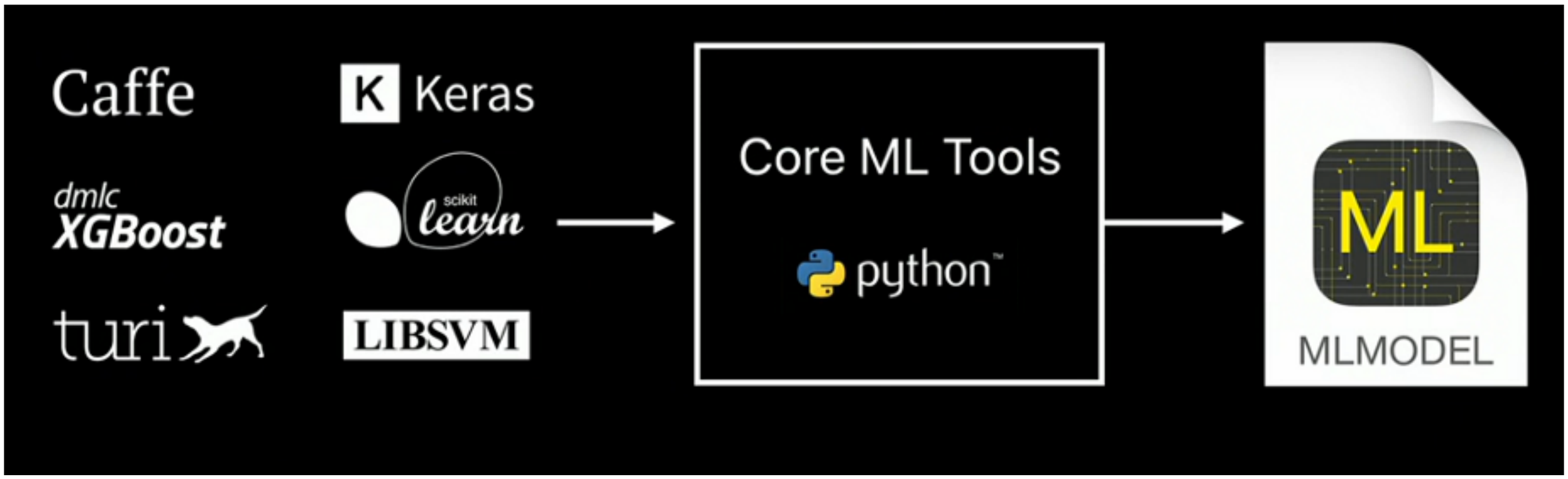


python_short_examples > TuriExample.ipynb

CoreML

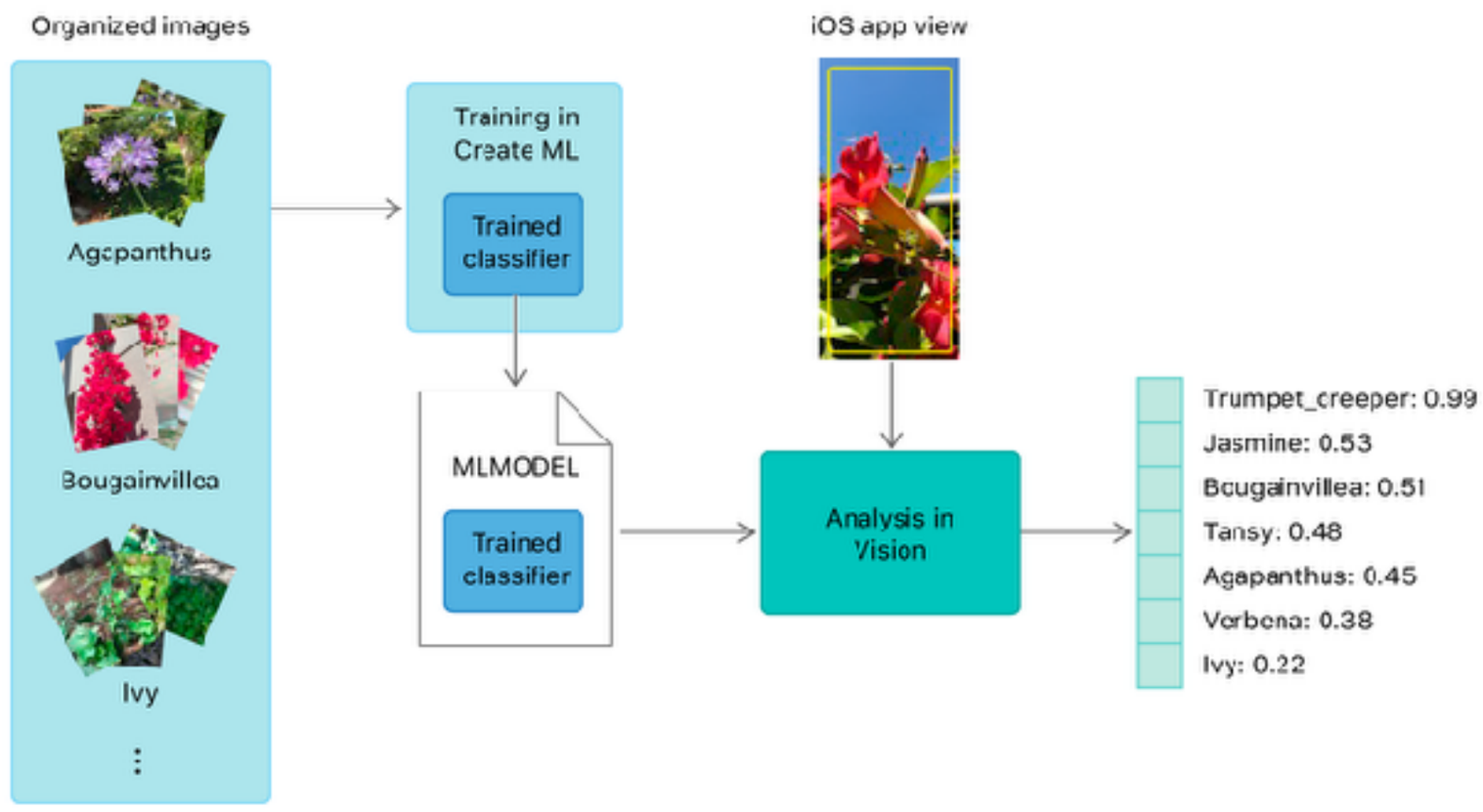

Conversion Workflow
Model Source
Xcode
Your App
e.g. Caffe
ML
MLMODEL
SWIFT
exec

CoreML


Caffe
Keras
dmlc
XGBoost
scikit
learn
turi
LIBSVM
Core ML Tools
python
ML
MLMODEL

CoreML with vision API

- load ml model in Xcode
- wrap model
- create vision request
- wait for result in completion handler



the vision API

```
// generate request for vision and ML model
let request = VNCoreMLRequest(model: self.model,
                              completionHandler: resultsMethod)

// add data to vision request handler
let handler = VNImageRequestHandler(cgImage: cgImage!, options: [:])

// now perform classification
do{
     try handler.perform([request])
}catch _{
     …
}

func resultsMethod(request: VNRequest, error: Error?) {
        guard let results = request.results as? [VNClassificationObservation]
            else { fatalError() }

        for result in results {
            if(result.confidence > 0.05){
                print(result.identifier,result.confidence)
            }
        }
    }
```

setup vision request with completion handler and ML model

add data to request(s)

perform request

interpret request results

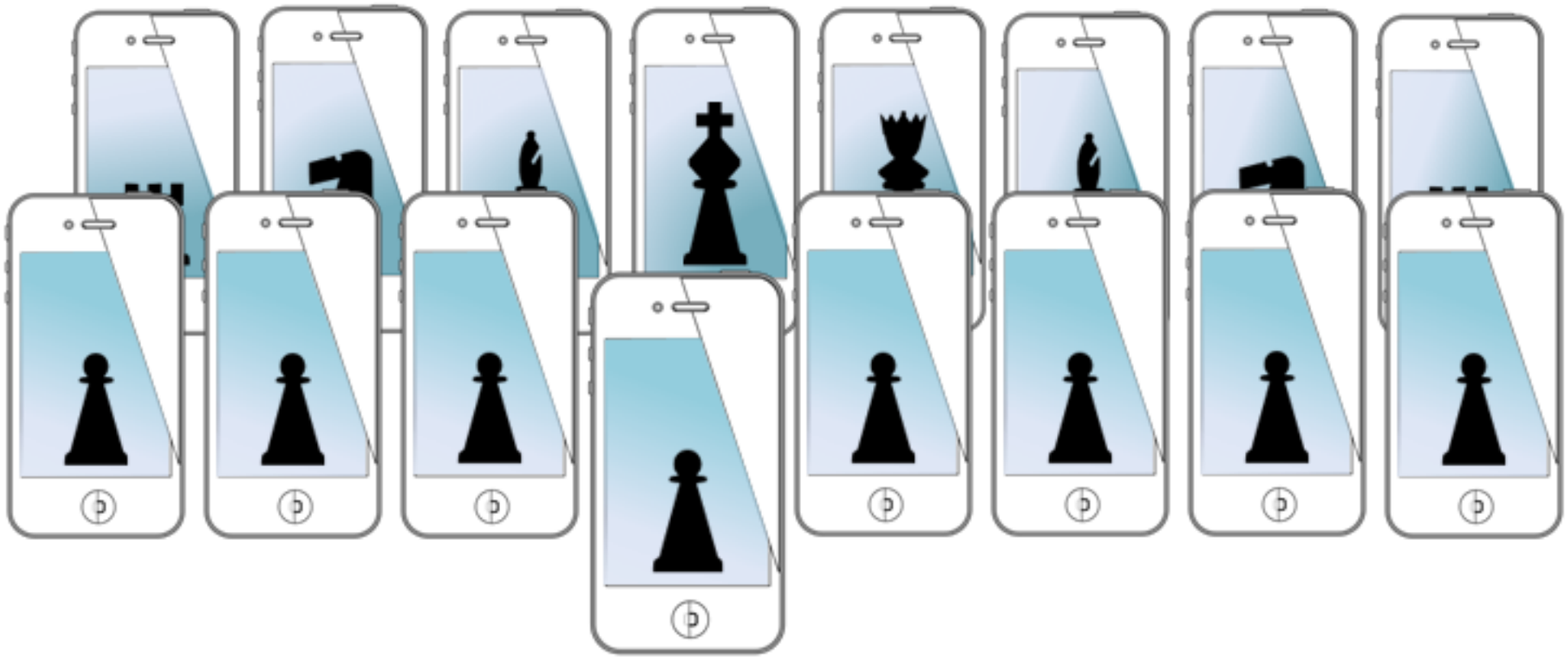
CoreML, a taste

- demo using vision API

for next time…

- flipped module for using Turi and iOS
- even if you are in Data Mining or Python ML, take a look
  - maybe skip over parts you know
  - we will have a near production level ML as a service platform by the end of the video lecture

MOBILE SENSING LEARNING



CS5323 & 7323

Mobile Sensing and Learning

machine learning crash course

Eric C. Larson, Lyle School of Engineering,
Computer Science, Southern Methodist University