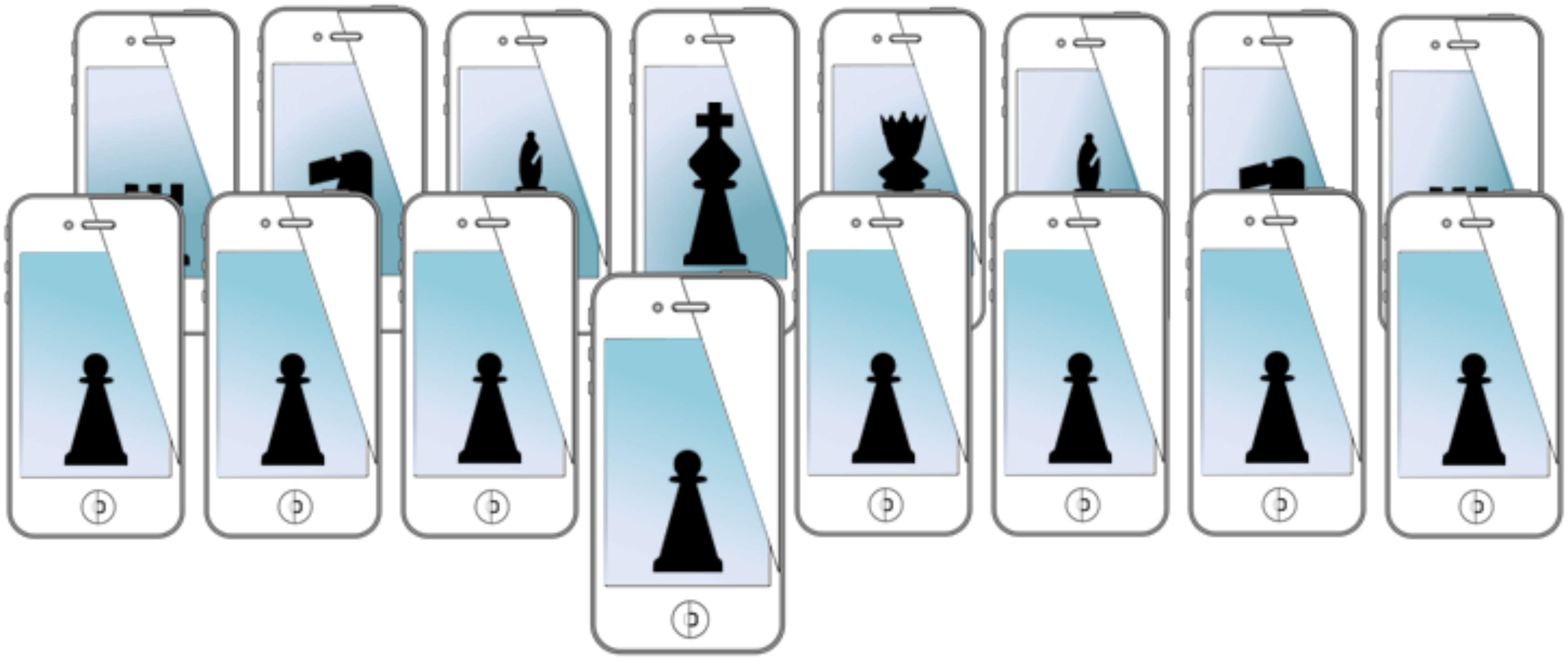
MOBILE SENSING LEARNING



CS5323 & 7323

Mobile Sensing and Learning

machine learning crash course

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

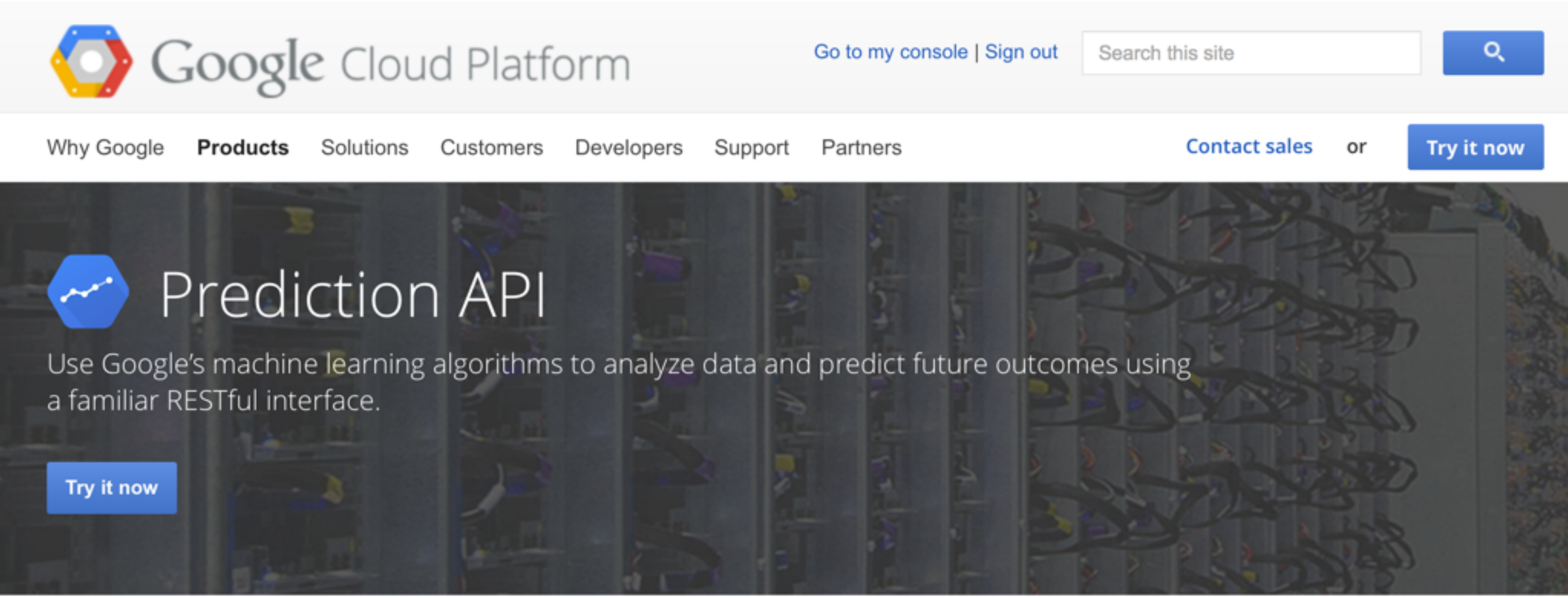
course logistics

- grading update
- next time: final flipped assignment
  - you can run server from your laptop (or cloud)
- **A5** is due soon
  - try to make it a first draft of your final project
- **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!

machine learning



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

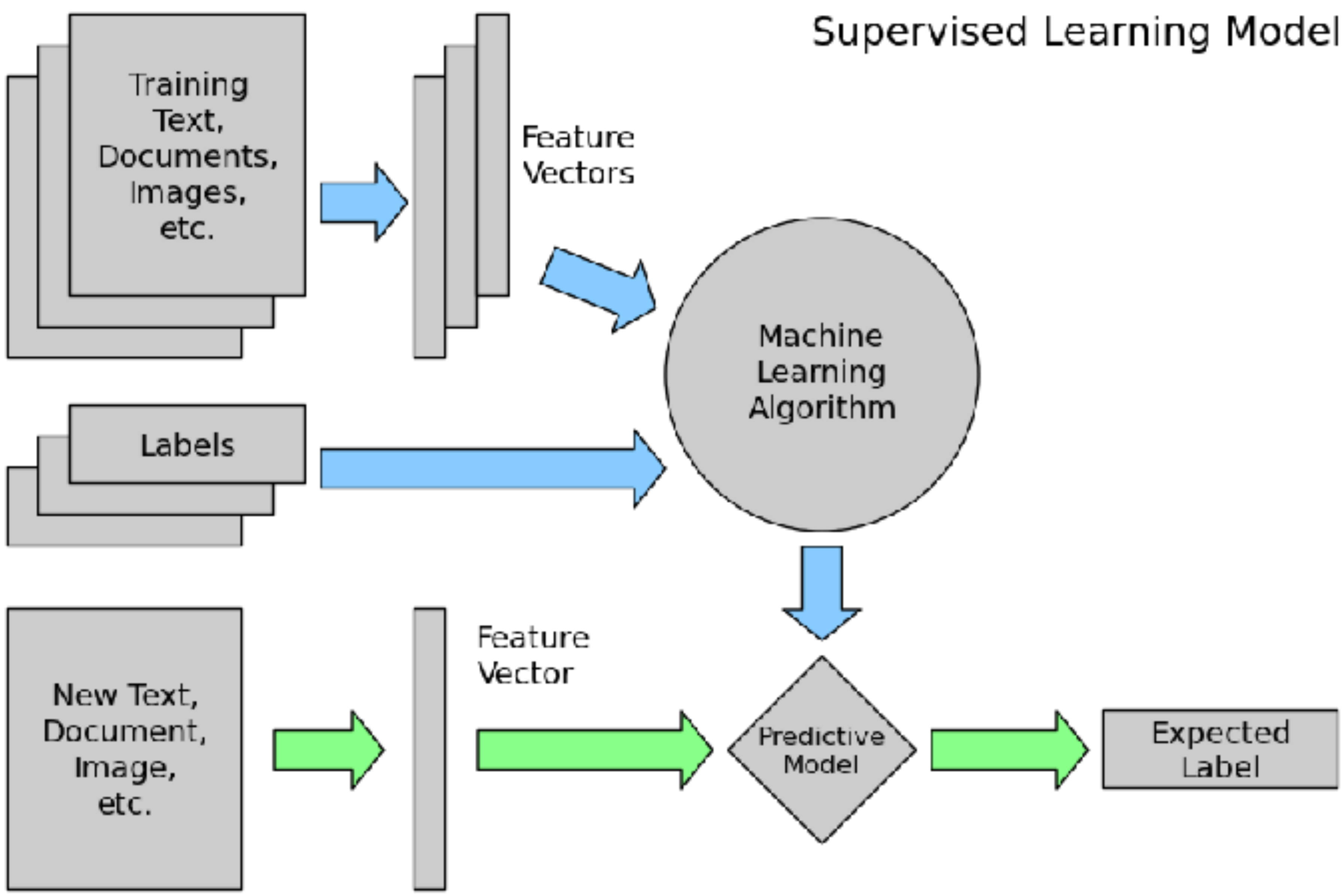
noun: **learning**

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;

teach a computer to learn something **through experiences**

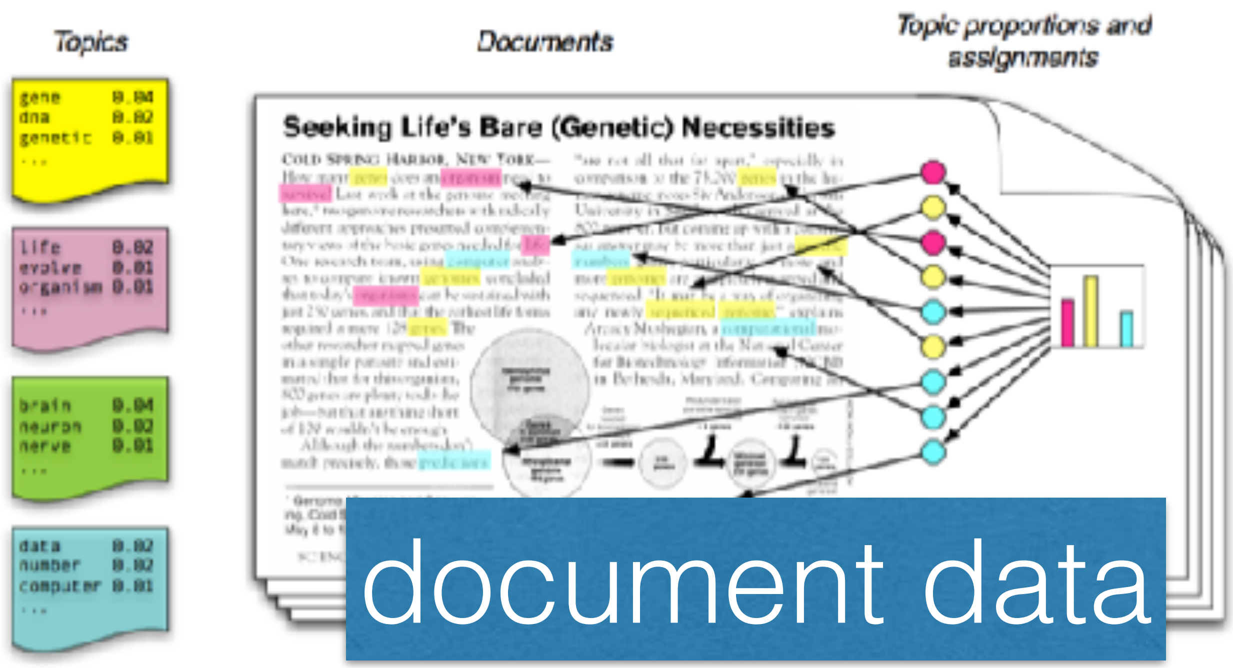
pattern recognition

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



document data

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

table data

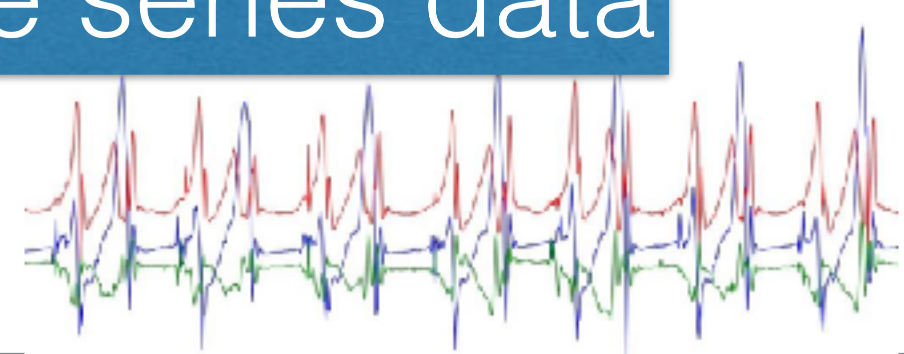
Attributes, columns, variables, fields, characteristics, **Features**

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

image data

time series data



features and labels

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

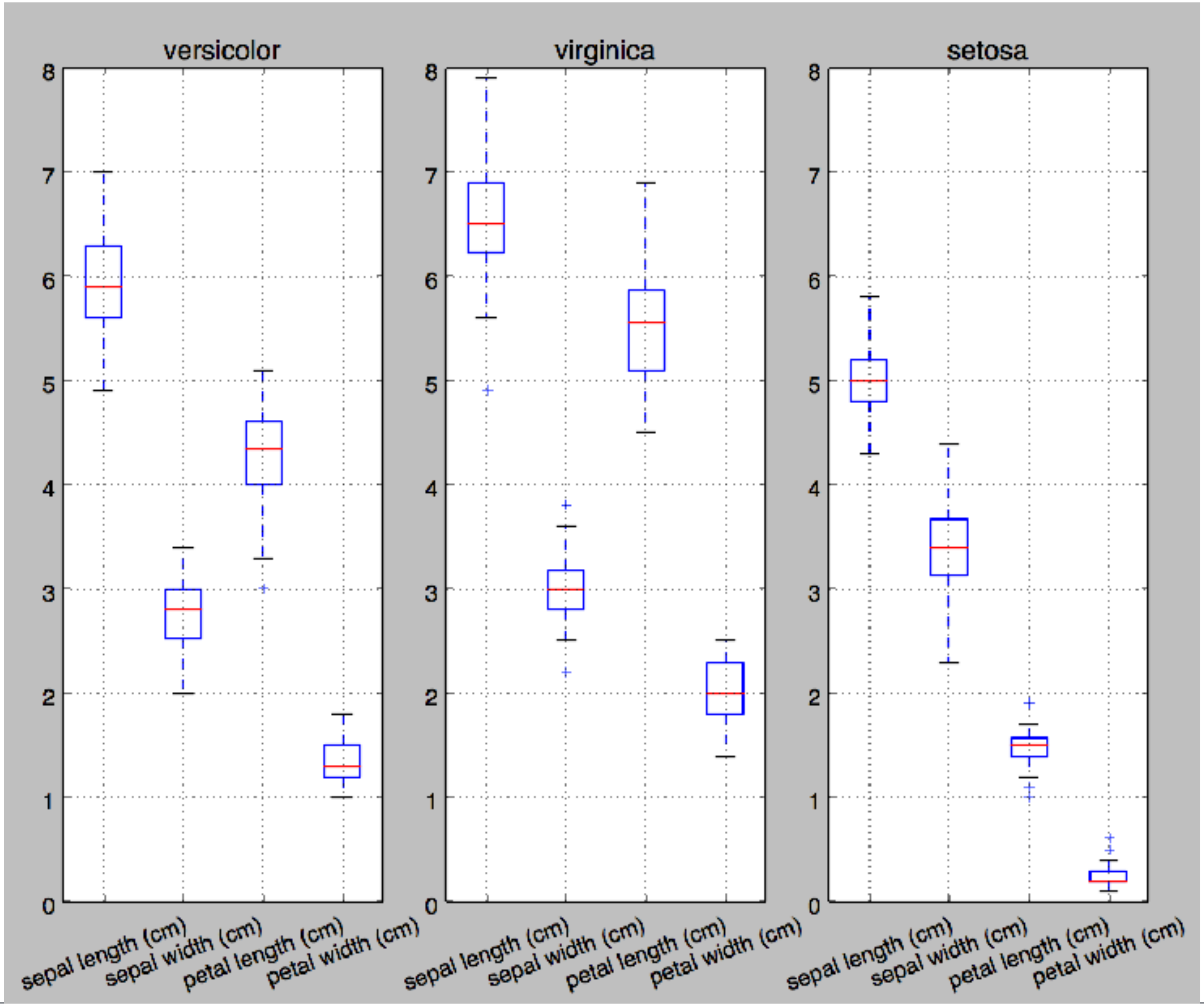
setosa versicolor virginica

- 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
```

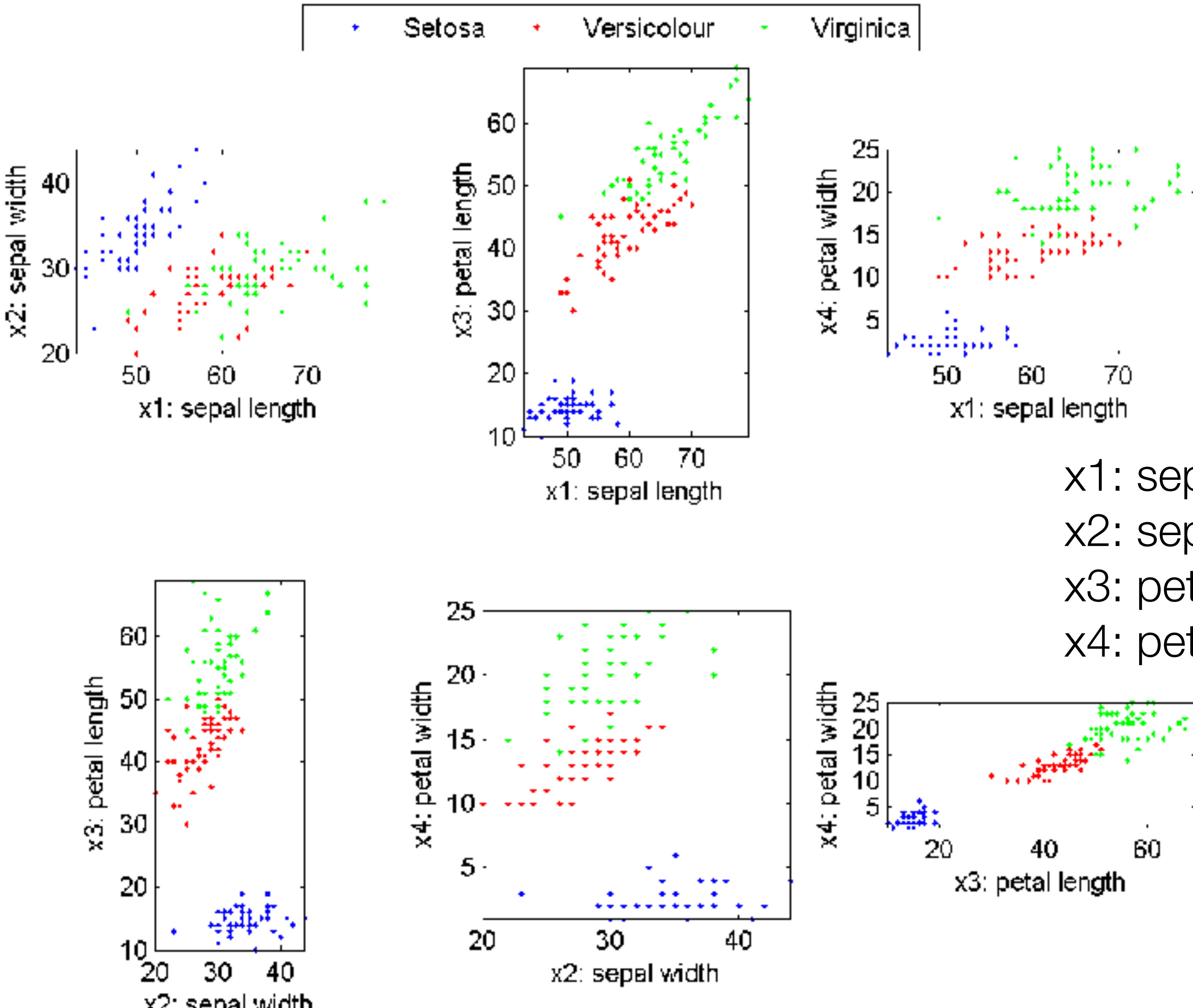
50 examples each

visualizing features



separability

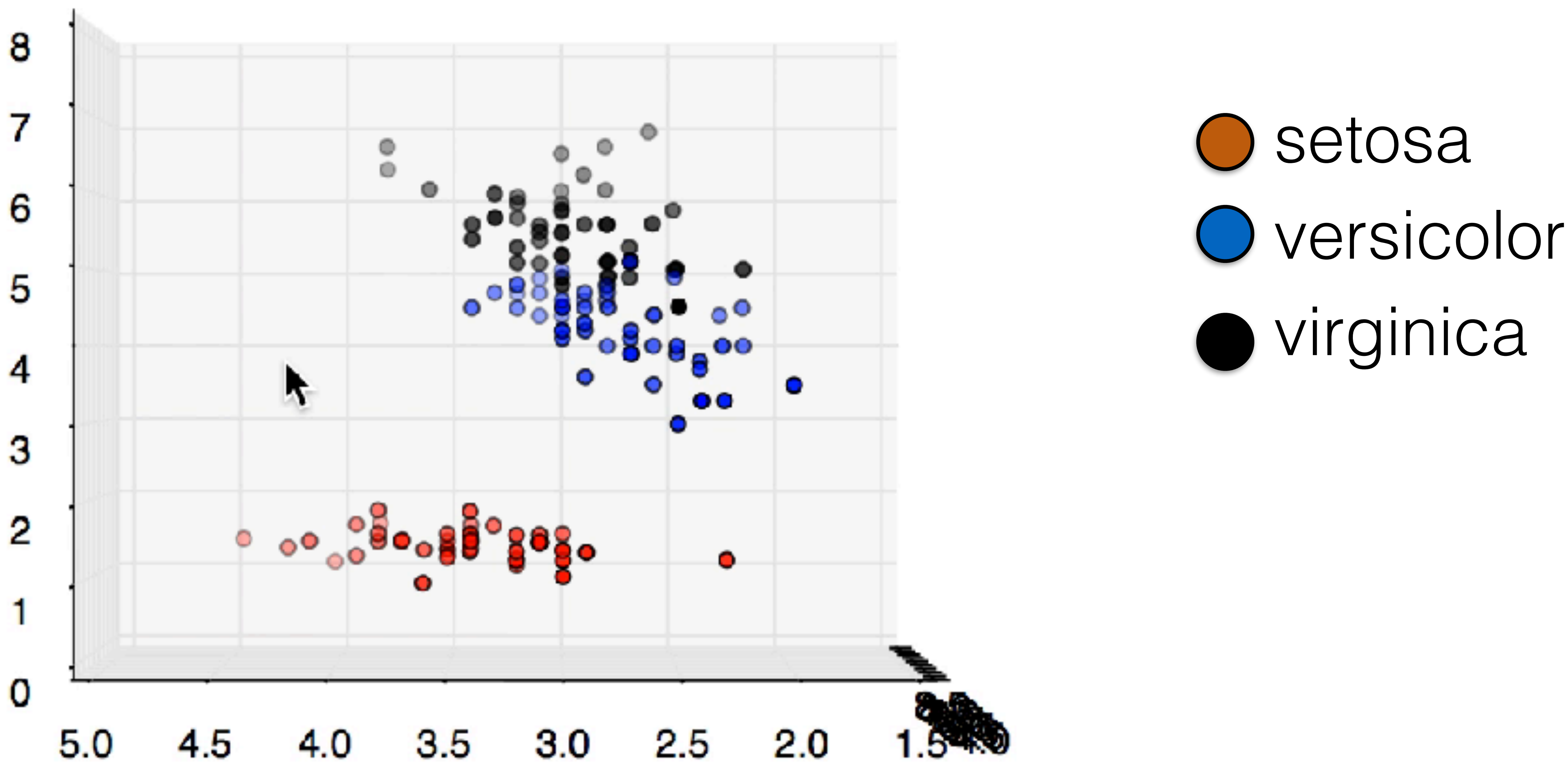
visualize features



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!

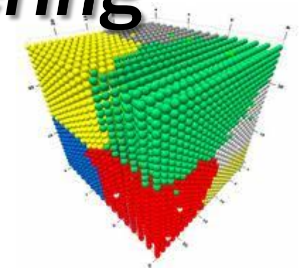
categorizing ML algorithms

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

machine learning: categorize

- categorize by availability of labels

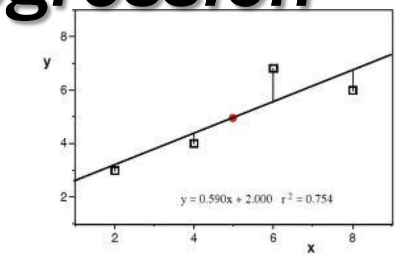
Clustering



unlabeled

unsupervised

Regression



Classification

Ordinal Reg.

labeled

supervised

machine learning: categorize

`[5.1, 3.5, 1.4, 0.2]` setosa

`[5.7, 2.8, 4.5, 1.3]` versicolor

`[7.6, 3. , 6.6, 2.1]` 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

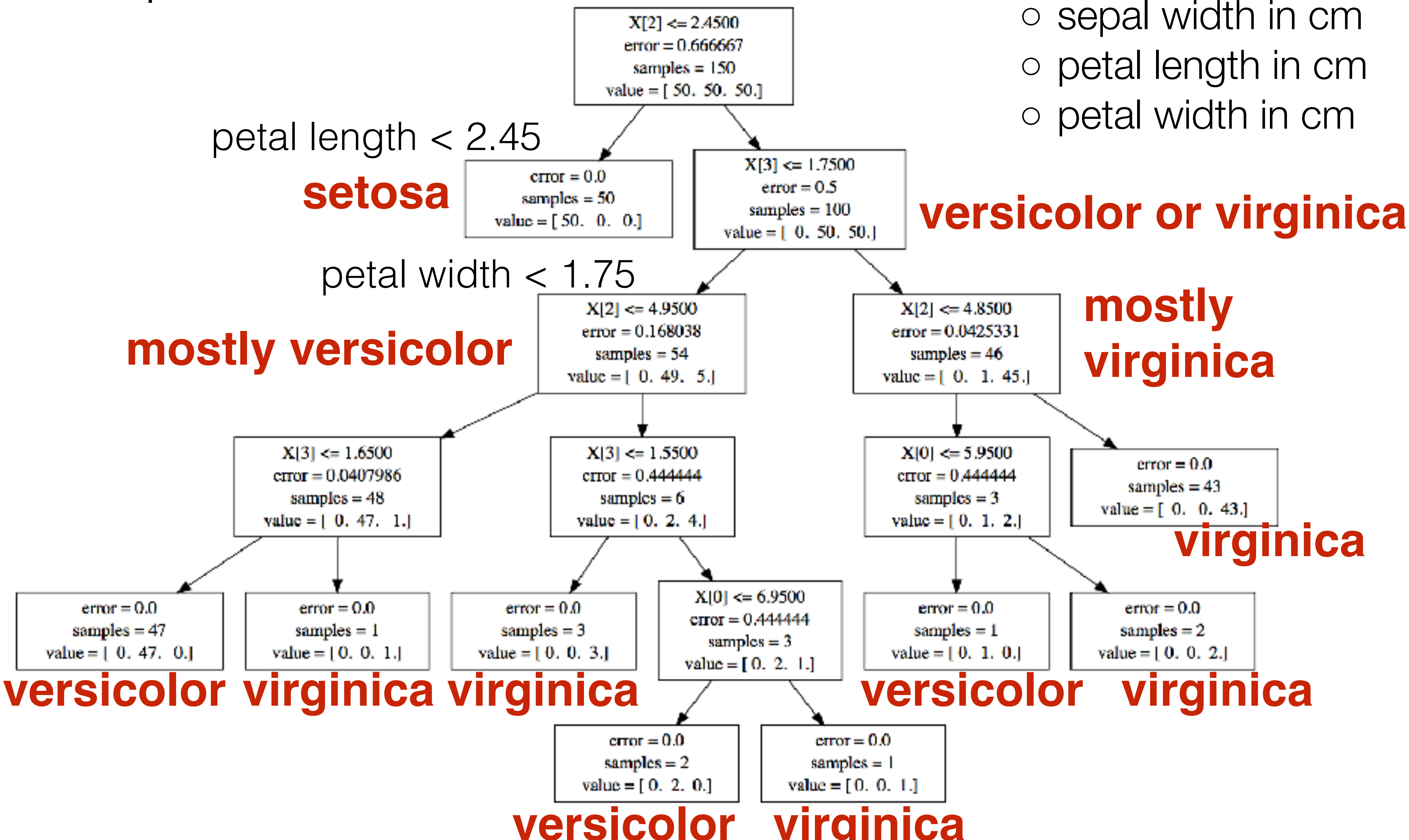
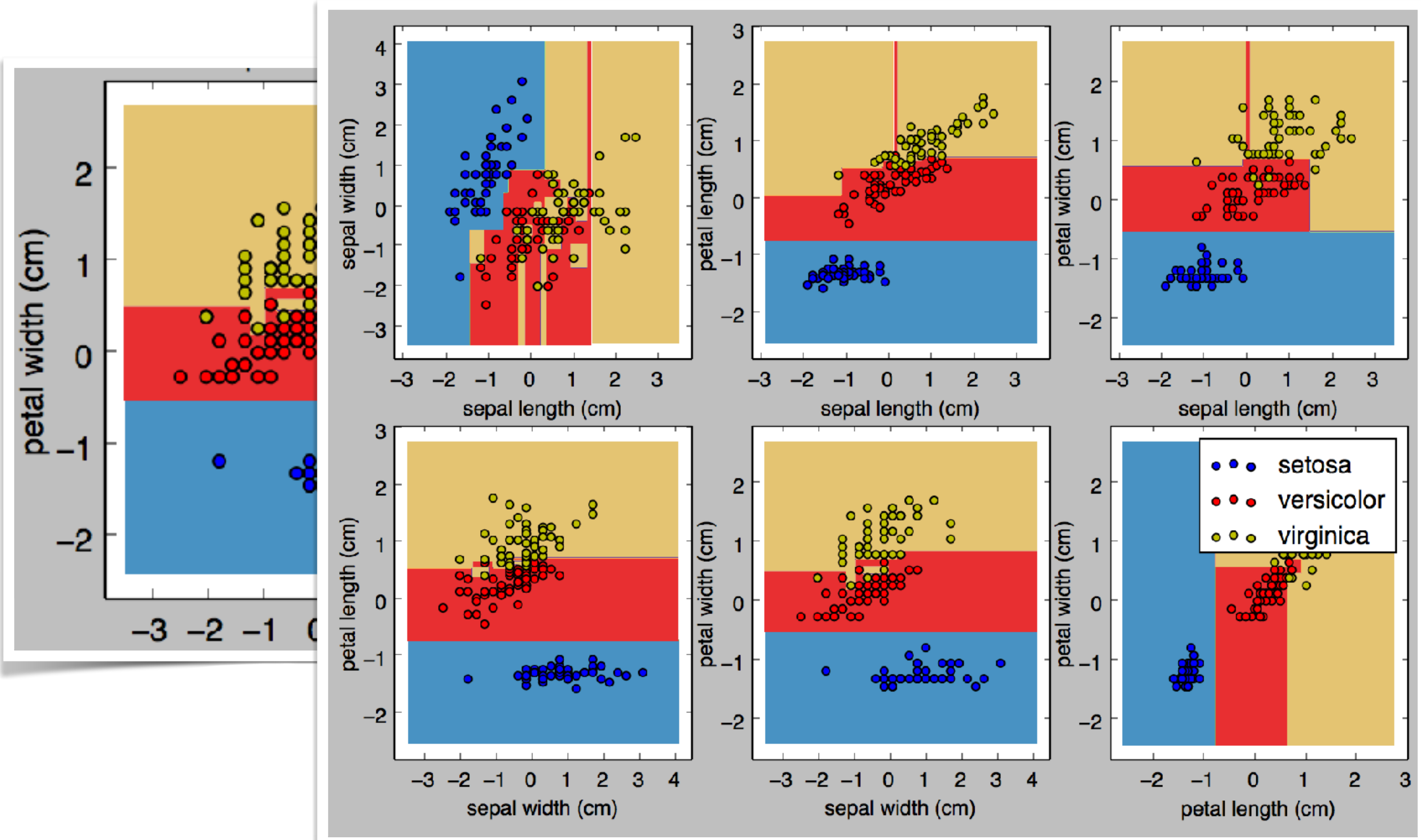


image: scikit-learn

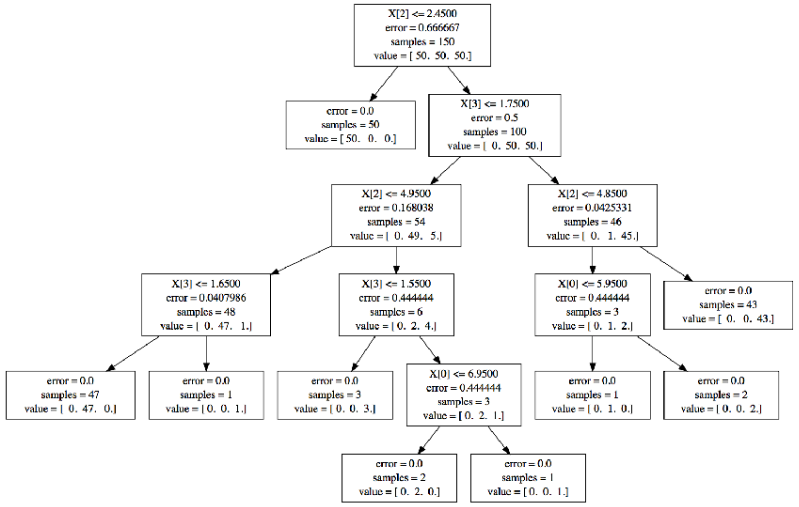
decision tree class boundaries



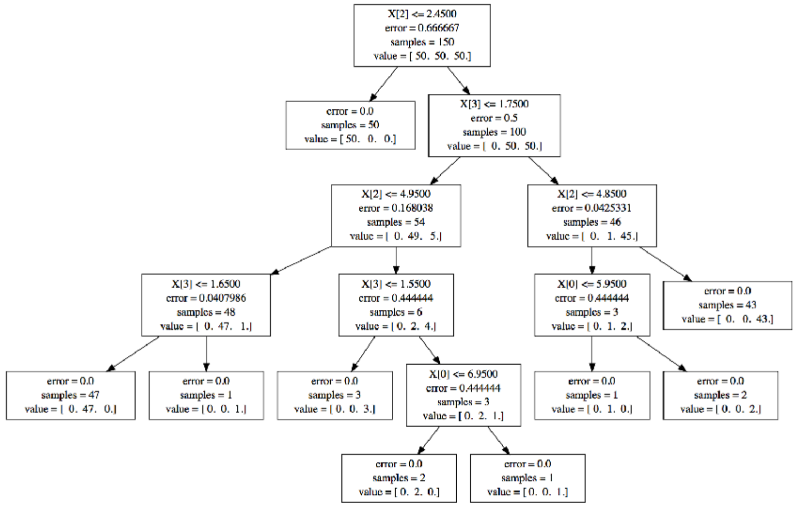
random forests

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

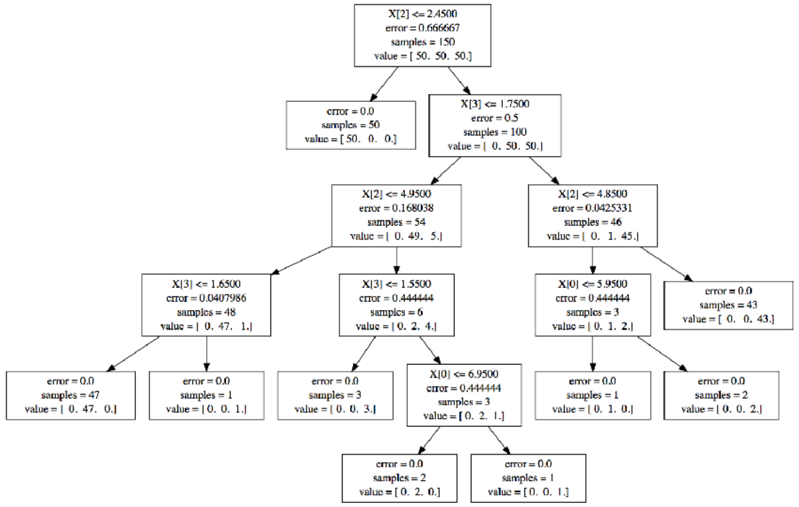
120 examples
3 features



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

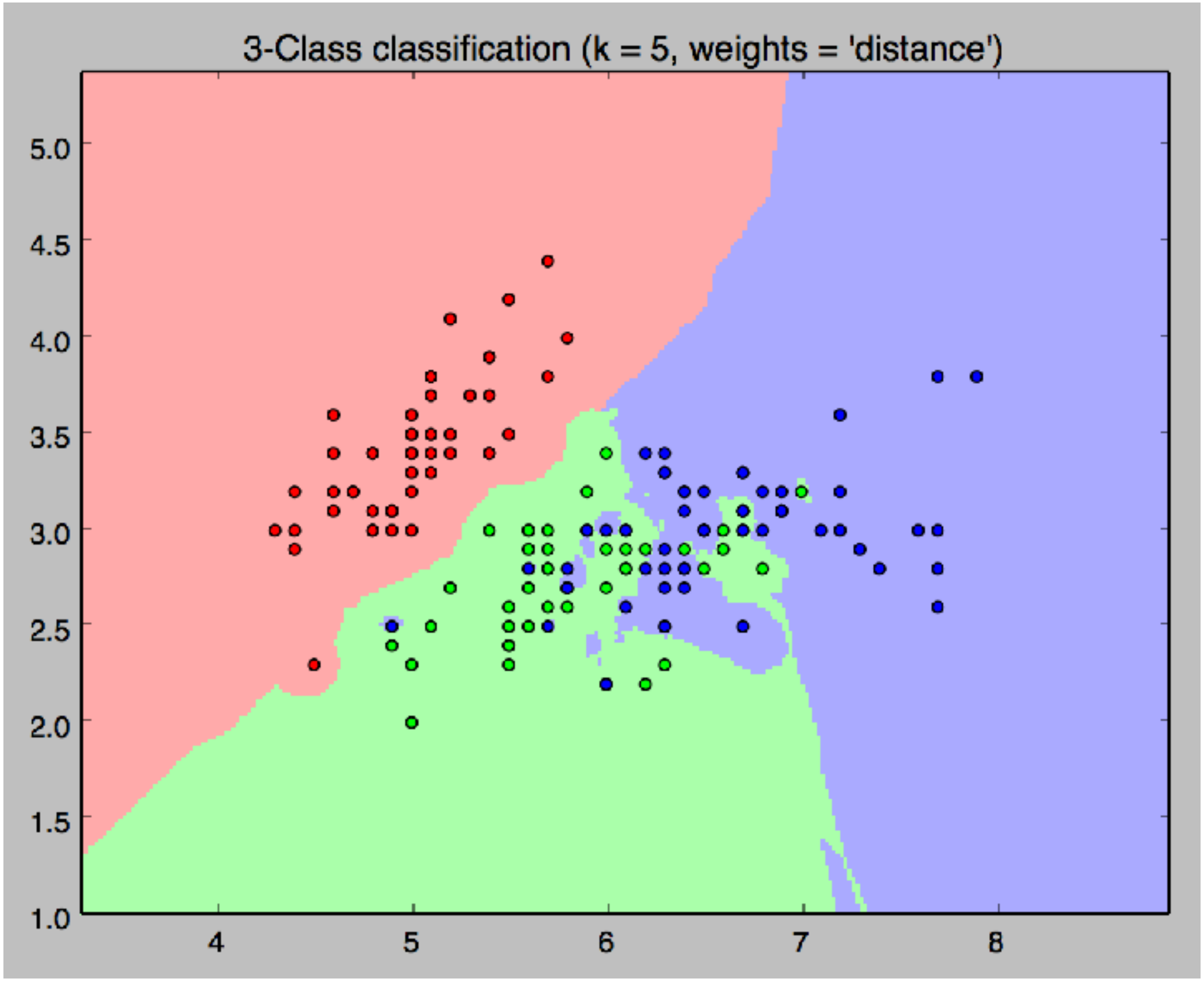
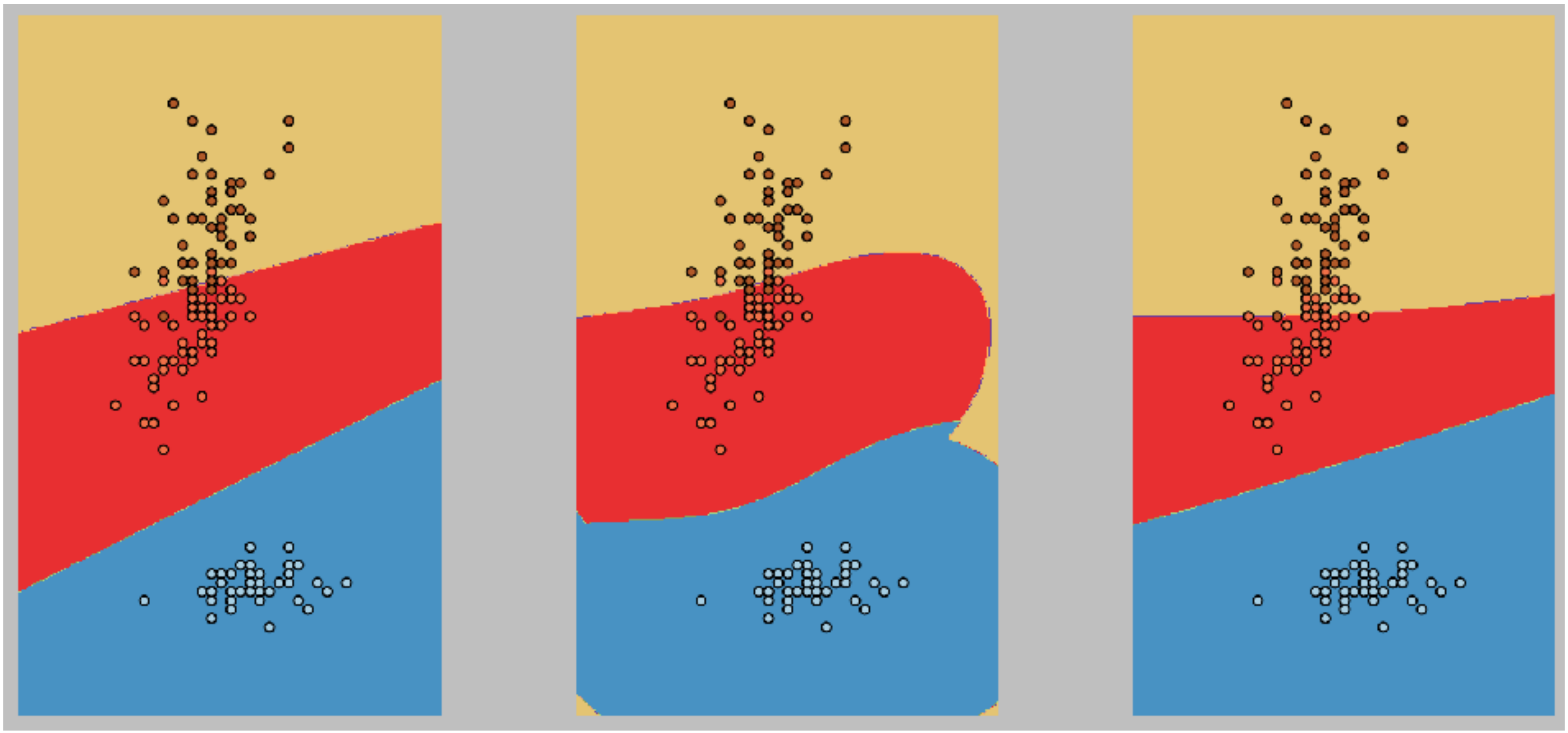


image: datasciencerules.org

support vector machines

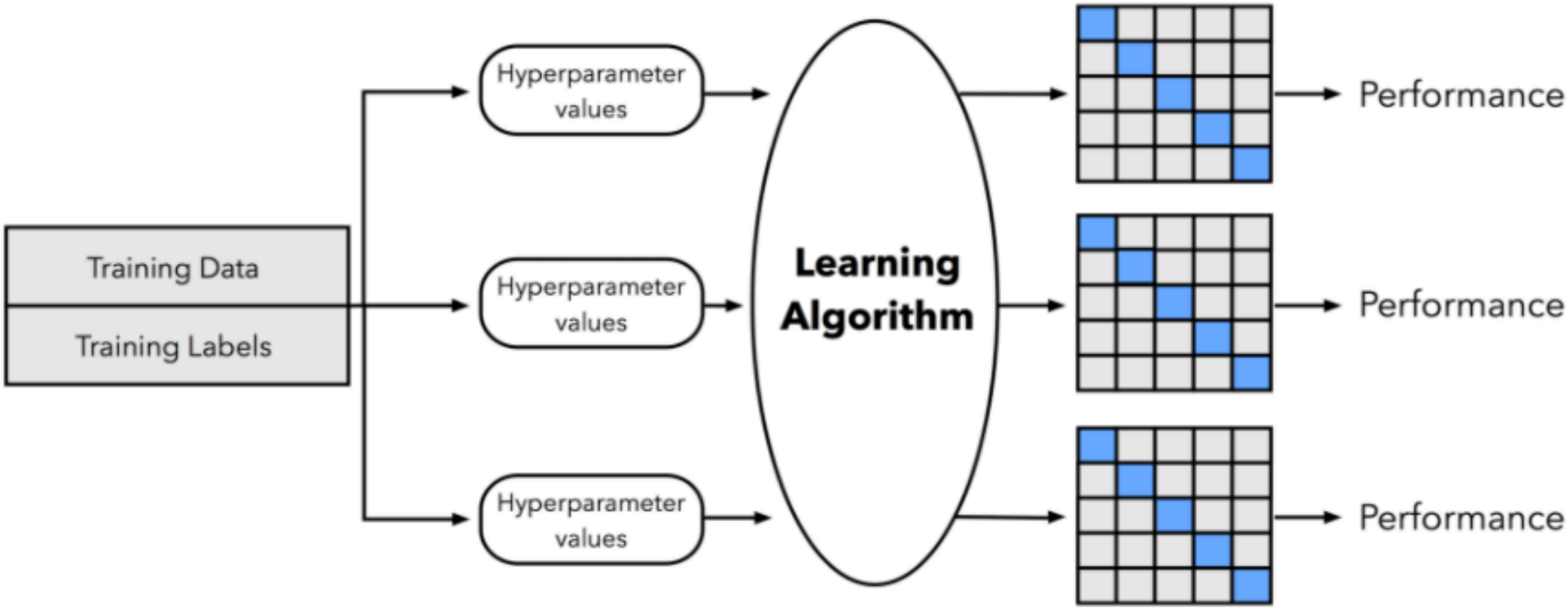
- find a vector that separates the training data



linear vector gaussians sigmoid

finding the best ML model

- try a bunch of stuff until it works well enough

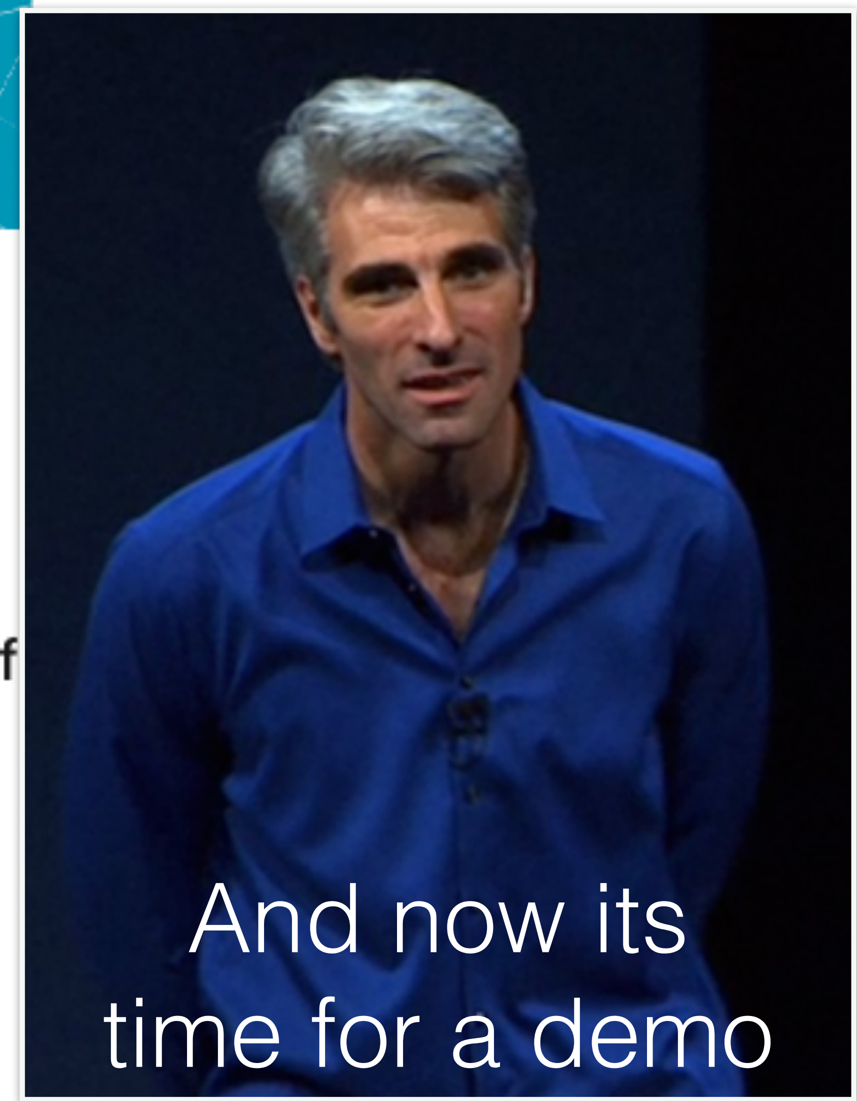


http://ethen8181.github.io/machine-learning/model_selection/model_selection.html

turi create and scikit-learn

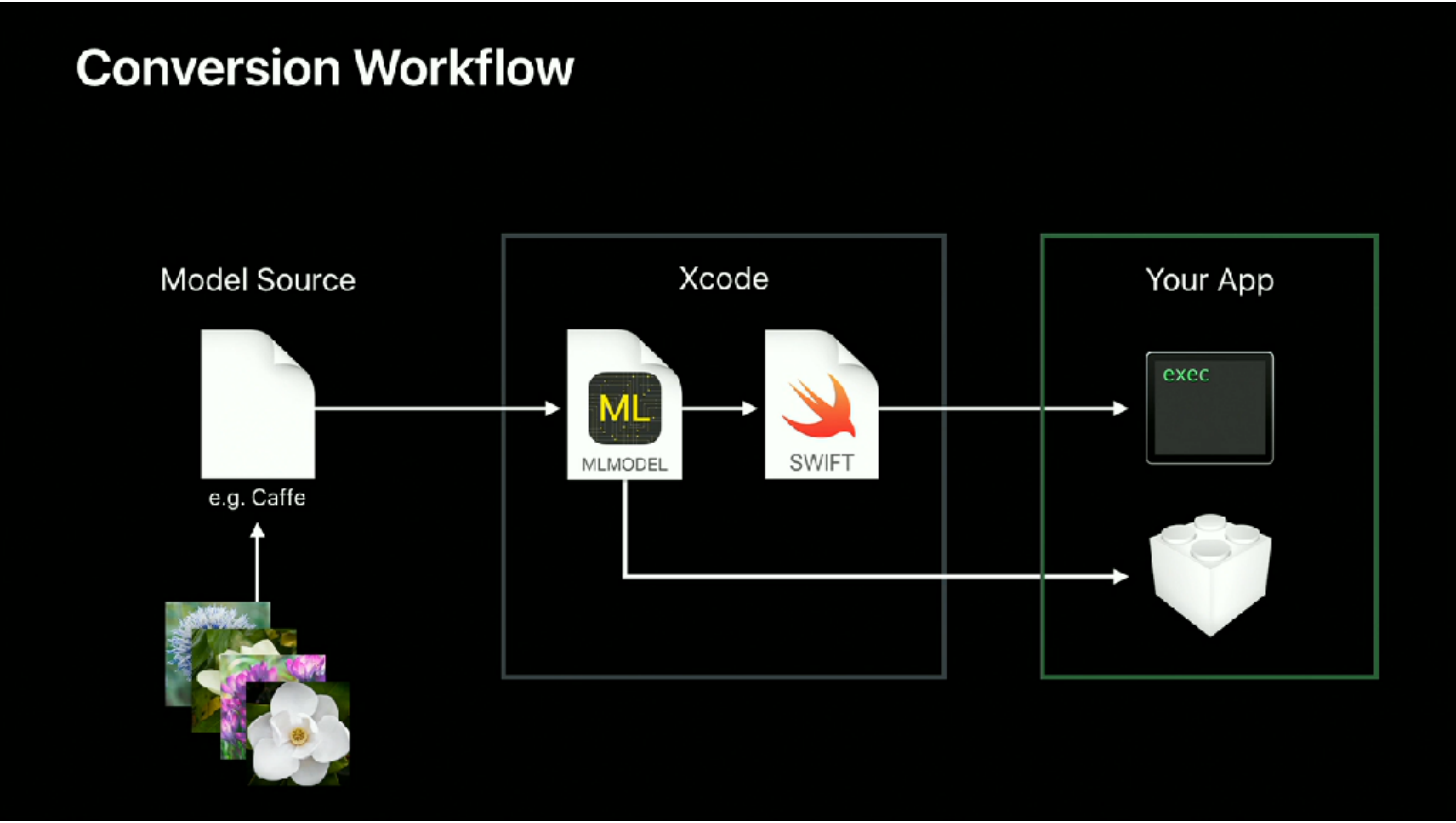
Carlos Guestrin · 2nd

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

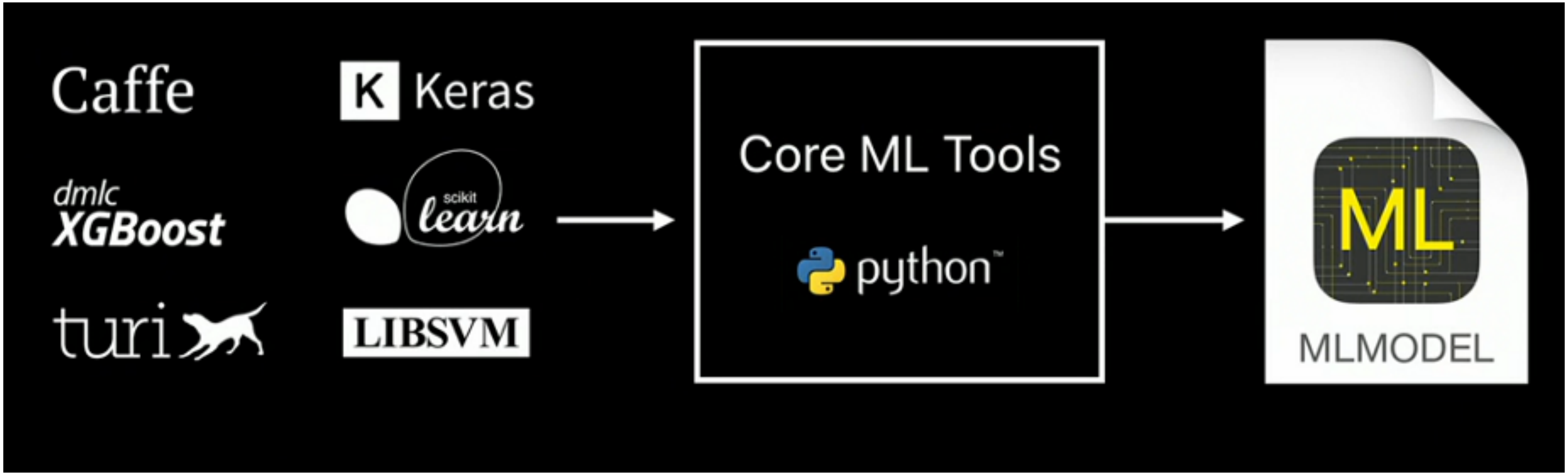


python_short_examples > TuriExample.ipynb

CoreML

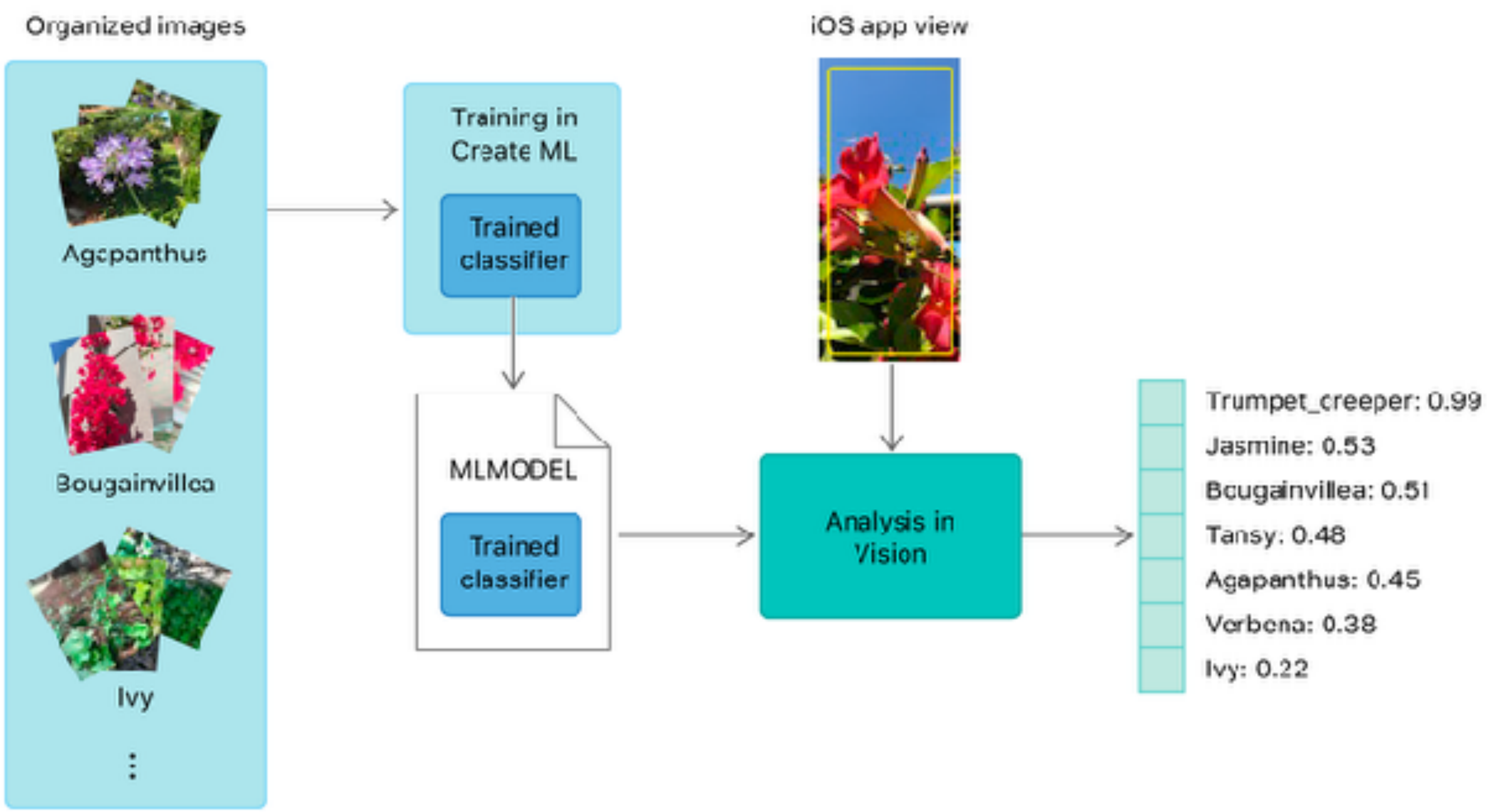

Conversion Workflow
Model Source
e.g. Caffe
Xcode
ML
MLMODEL
SWIFT
Your App
exec

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

```swift
// 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

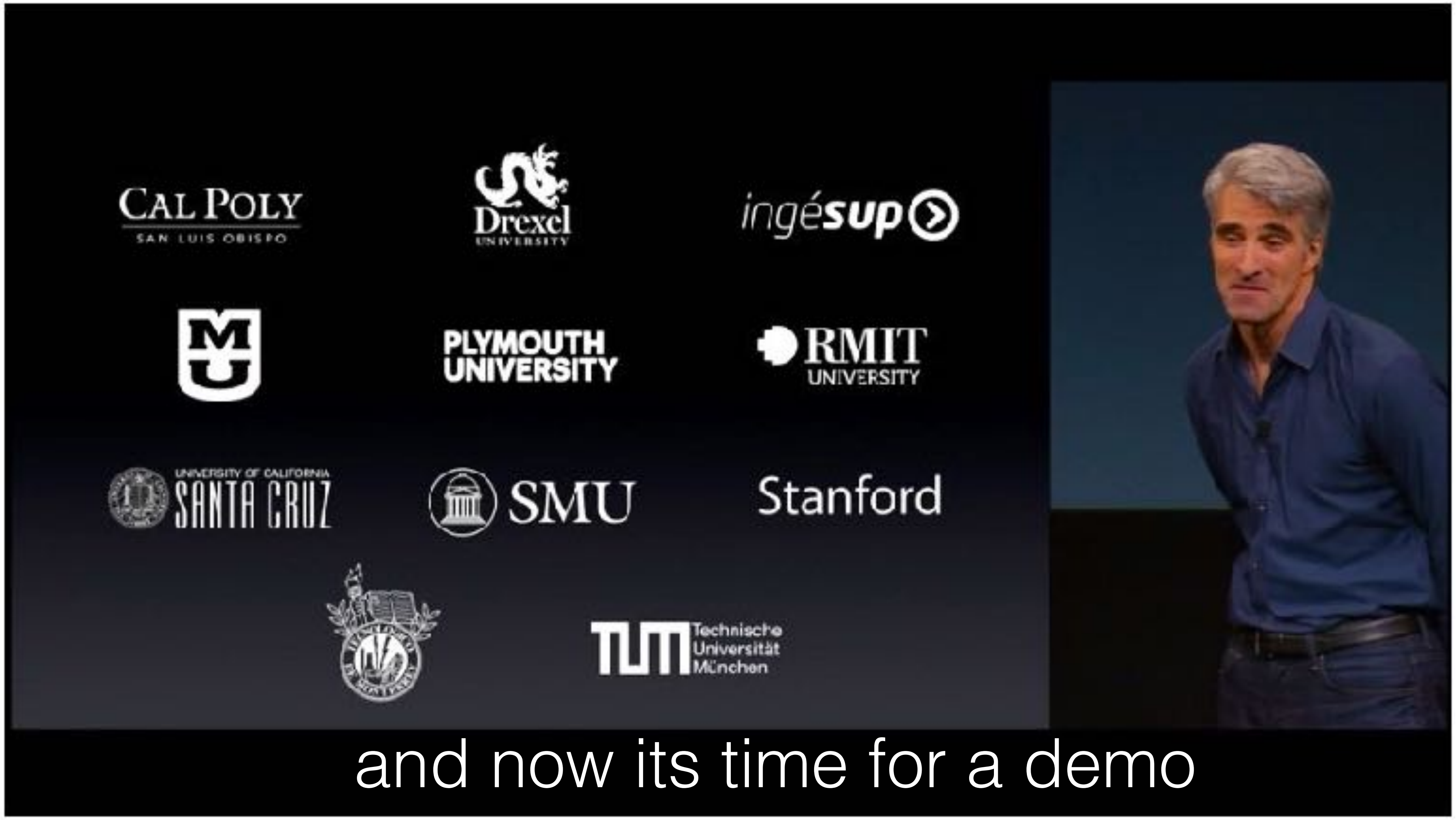
add data to request(s)

perform request

interpret request results

CoreML, a taste

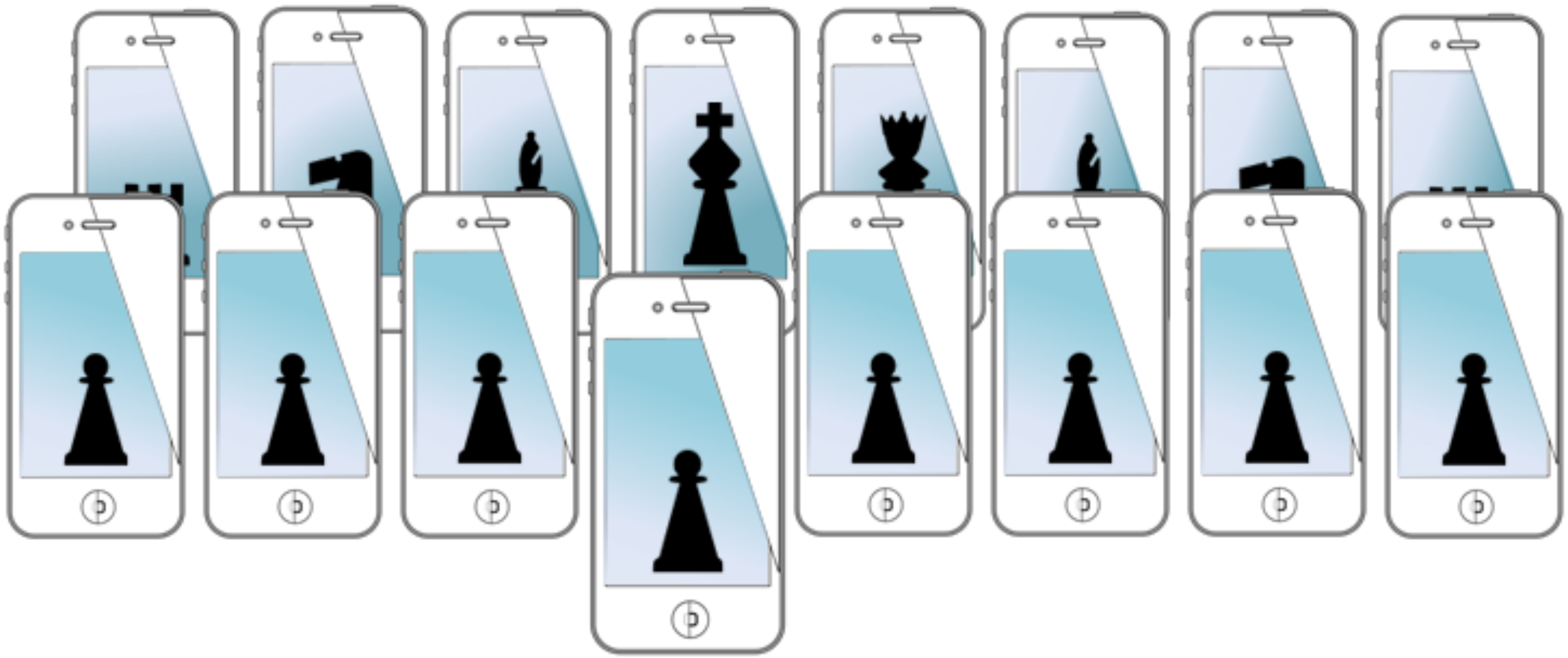
- demo using vision API



for next time…

- flipped module for using Turi and iOS
- even if you are in (or have taken) 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