STAT 339
Probabilistic Modeling and Machine Learning

October 4, 2021

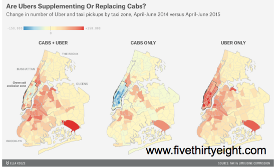
Colin Reimer Dawson

Outline

Some Cool Things you can do with data

Recommendation Systems

Data-Driven Journalism

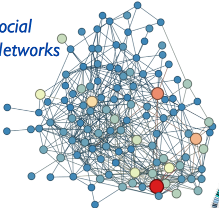


Competitive State Summary

State	EV	O	R	Forecast		90% Prediction Interval	%
NM	5	183	355	Obama +8.9	+/- 6.6		98%
ME-2	1	184	354	Obama +8.2	+/- 8.9		94%
OR	7	191	347	Obama +7.9	+/- 6.3		98%
MI	16	207	331	Obama +7.2	+/- 5.4		98%
MN	10	217	321	Obama +7.0	+/- 5.4		98%
PA	20	237	301	Obama +5.1	+/- 5.0		96%
WI	10	247	291	Obama +4.0	+/- 5.6		88%
NV	6	253	285	Obama +3.5	+/- 5.6		85%
IA	6	259	279	Obama +2.9	+/- 6.3		78%
NH	4	263	275	Obama +2.8	+/- 6.8		75%
OH	18	281	257	Obama +2.6	+/- 5.2		80%
CO	9	290	248	Obama +1.1	+/- 5.8		63%
VA	13	303	235	Obama +0.8	+/- 4.8		61%
FL	29	332	206	Romney +0.7	+/- 5.3		59%
NC	15	347	191	Romney +2.6	+/- 5.0		81%
NE-2	1	348	190	Romney +6.5	+/- 6.8		94%
AZ	11	359	179	Romney +7.8	+/- 6.7		97%
MO	10	369	169	Romney +9.0	+/- 5.5		99%
GA	16	385	153	Romney +9.7	+/- 5.5		100%
MT	3	388	150	Romney +9.7	+/- 7.4		98%

Political Science

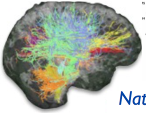
Social Networks



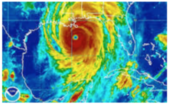
GIS / Development / Public Policy

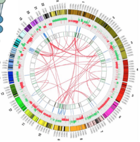
Sports

Finance



Meteorology / Climate Science





Natural Sciences

Thanks to David Shuman at Macalester College for this slide

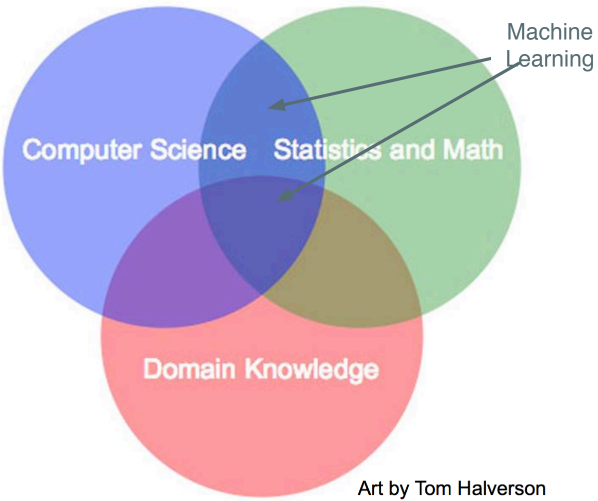
What is Machine Learning?

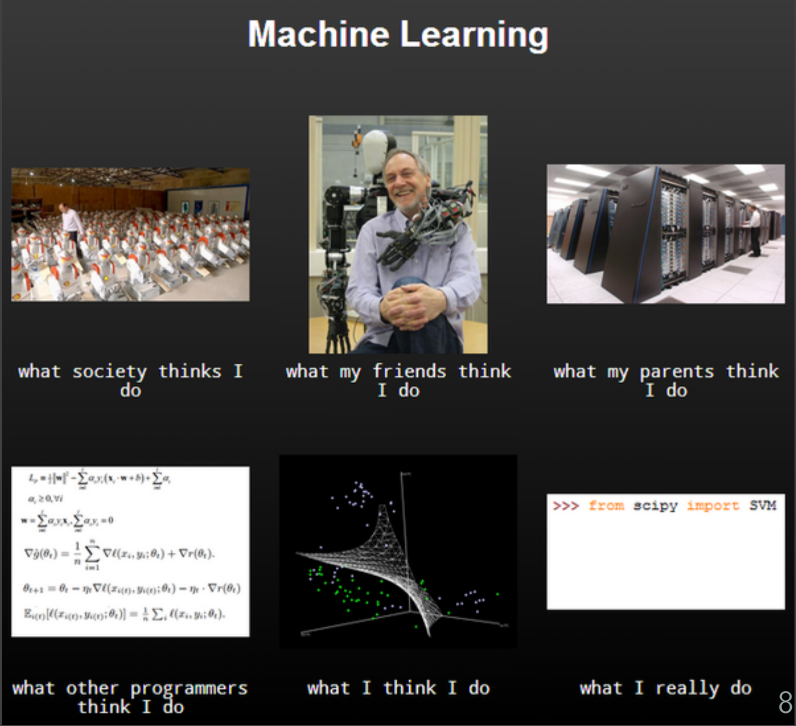
> *"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."*
> *— Tom Mitchell*

What is Machine Learning?

> *"[Machine Learning is a] field of study that gives computers the ability to learn without being explicitly programmed."*
> *— Arthur Samuel*

Statistics, Computer Science, and Machine Learning

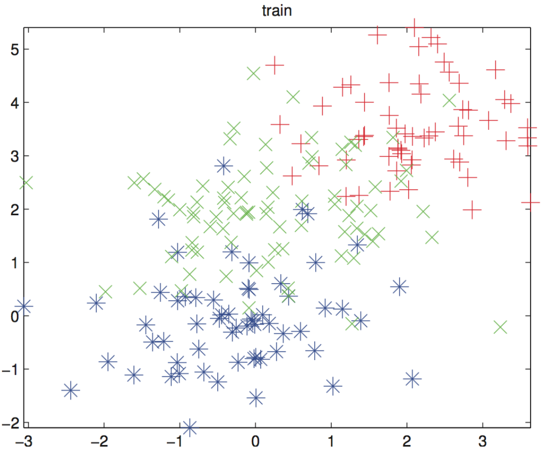


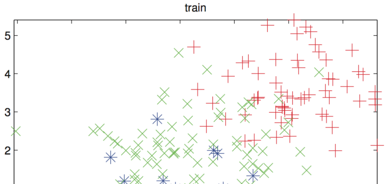

Machine Learning
what society thinks I do
what my friends think I do
what my parents think I do
what other programmers think I do
what I think I do
>>> from scipy import SVM
what I really do

Types of Learning

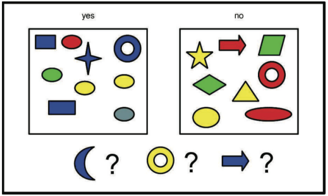
- Supervised Learning: Learning to make predictions when you have many examples of "correct answers"
  - Classification: answer is a category / label
  - Regression: answer is a number
- Unsupervised Learning: Finding structure in unlabeled data
- Reinforcement Learning: Finding actions that maximize long-run reward (not part of this course)

Supervised Learning





Supervised Learning with a Probabilistic Model



(a)

D features (attributes)

N cases

Color	Shape	Size (cm)
Blue	Square	10
Red	Ellipse	2.4
Red	Ellipse	20.7

Label
1
1
0

(b)

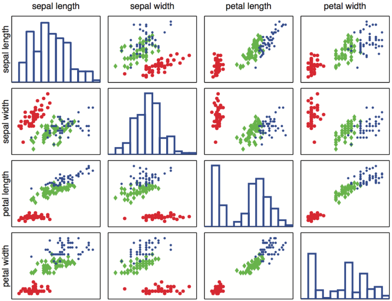
- Training data: $\{(t_i, \mathbf{x}_i)\}_{i=1}^{n}$; t_i = label, $\mathbf{x}_i$ = features.
- Fit a model of all of the features: $P(\mathbf{x}, t)$, or $P(\mathbf{x}|t)$
- Testing: Assign $P(t_{new}|\mathbf{x}_{new},$ Model)

Data in Higher Dimensions

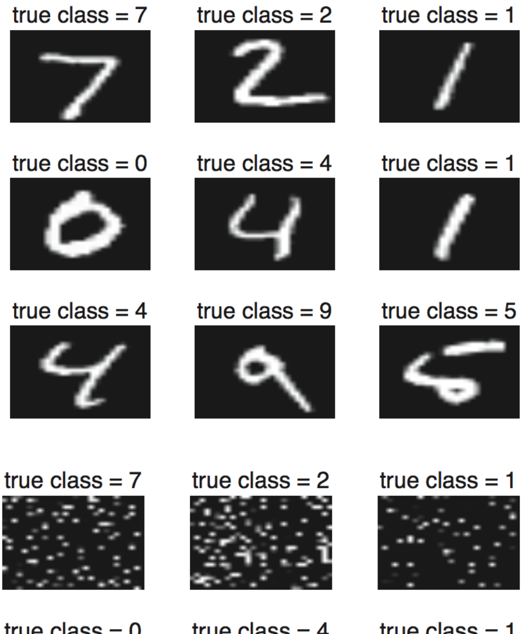
(a)

(b)

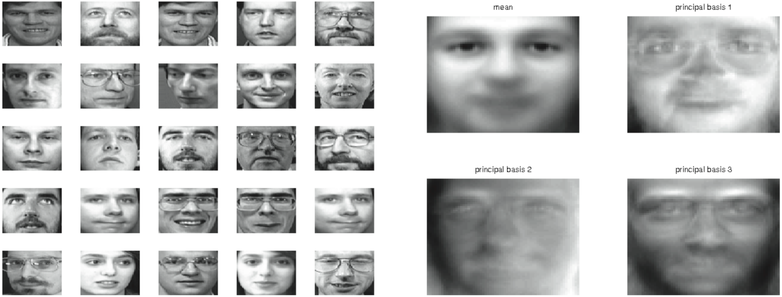
(c)



Data in Very High Dimensions

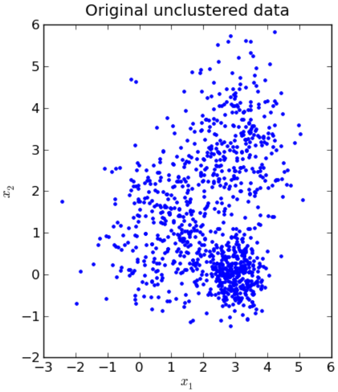


Aside: Feature Extraction (“Eigenfaces”)



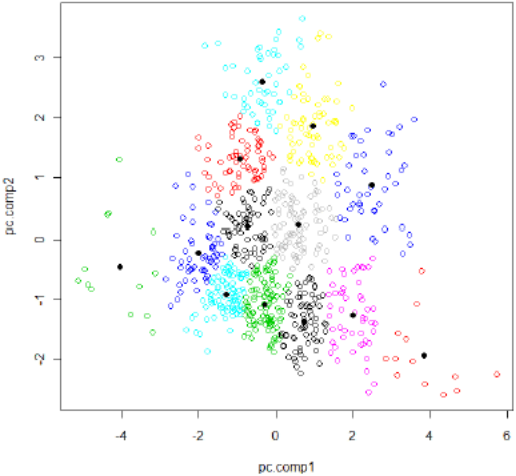
(a) (b)

Finding Clusters



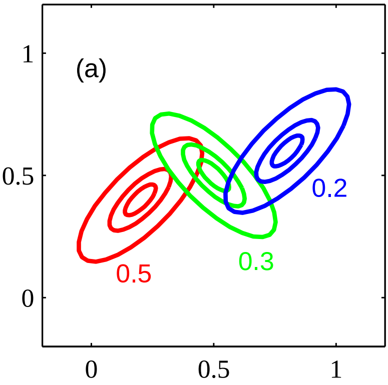
- Clustering: Grouping data into categories without any "ground truth" information
- Example Application: Modeling people's taste in movies

Model-Free Clustering



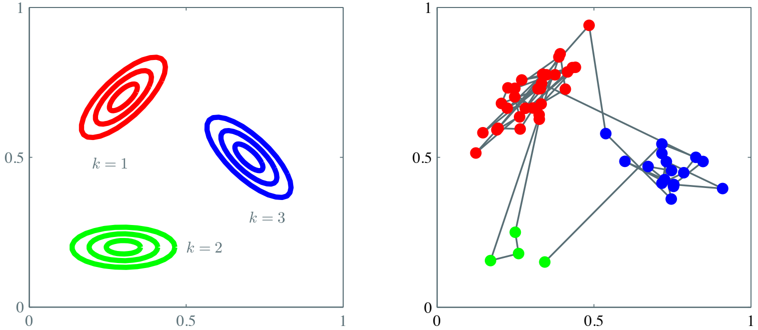
Model-free example: Given a distance metric, maximize distances among cluster centers; then assign points to closest center.

Clustering with a Probabilistic Model



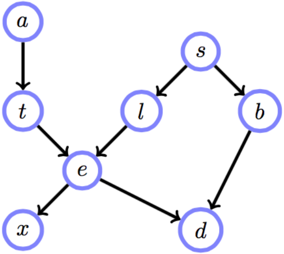
Output: A set of cluster weights and a probability distribution for each cluster

Clustering With Time



- We can combine a model of clusters with a model of how observations "transition" between clusters.
- Example: Speech recognition

Probabilistic Graphical Models



$x =$ Positive X-ray
$d =$ Dyspnea (Shortness of breath)
$e =$ Either Tuberculosis or Lung Cancer
$t =$ Tuberculosis
$l =$ Lung Cancer
$b =$ Bronchitis
$a =$ Visited Asia
$s =$ Smoker

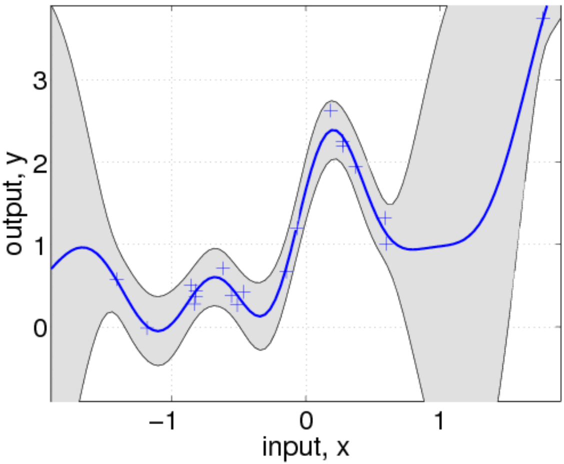
Figure: Probabilistic Graphical Model for Medical Diagnosis.

- "Probabilistic Graphical Models": learn joint distribution of several variables using a graph of relationships to impose structure
- Example: Medical diagnosis

Parametric Vs. Nonparametric Models

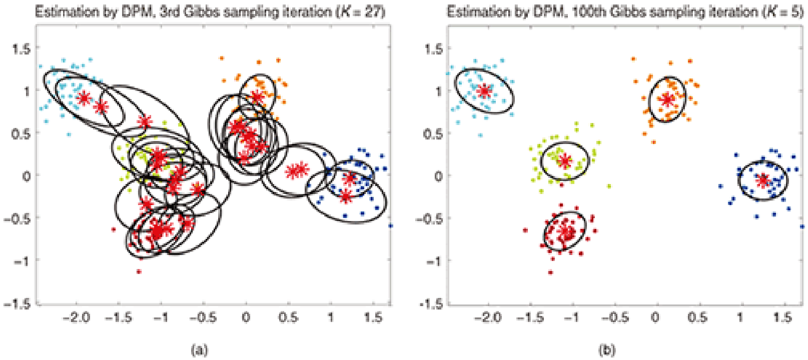
- A **parametric** model has a fixed degree of complexity, regardless of the amount of data
  - Examples: linear regression, clustering w/ fixed # of clusters, neural networks
- A **nonparametric** model can adaptively "grow" its complexity as the amount of data grows (effectively they have "infinite" complexity)
  - Examples: "Nearest neighbors" classification, Gaussian Process regression, clustering w/ unknown number of clusters

Gaussian Process Regression



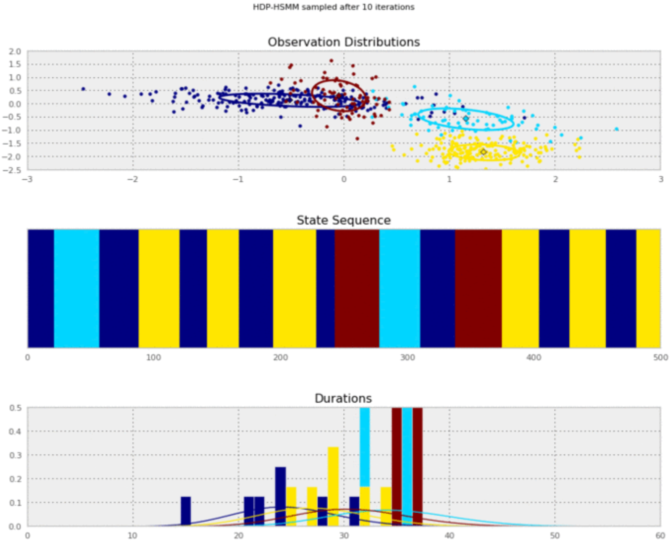
GP Regression: Assume a “smooth” function, but allow the amount of wiggliness adapt to the data

“Infinite” Clustering Model



Infinite Mixture Model: Assume “infinitely many” clusters, and figure out which ones appear in the data

Infinite Dynamic Clustering Model



Course Outline

- Course Website: `http://colindawson.net/stat339`
- Syllabus, slides, schedule, assignments, resources available there
- Electronic submission of assignments via GitHub (with or without actually using `git`
- HW Solutions will also be posted to GitHub

Course Outline

- Part I: Basic ML Ideas / Supervised Learning (2 weeks)
- Part II: Probabilistic Modeling Foundations (3 weeks)
- Part III: Probabilistic Inference Foundations (2 weeks)
- Part IV: Probabilistic Supervised Learning (2 weeks)
- Part V: Unsupervised Learning (3 weeks)
- Part VI: Nonparametric Models (time permitting)

Graded Components

- (Mostly) Weekly Problem Sets (50% across ~ 9 assignments)
- One Take-home Exam (20%; due 12/08)
- Group Project and Presentation (20%)
- Participation and Engagement (10%)

See the syllabus for Honor Code guidelines

Prerequisite Skills/Knowledge

- Key math background: Partial derivatives, vectors, chain rule (MATH 231)
- Basic programming skills (CS 150), preferably comfort with Python
- Different from CS 374: greater emphasis on **models and probabilistic reasoning**; less emphasis on data structures and coding
- We will definitely get "into the weeds" of math/stats derivations of formulas/algorithms
- Coding will be at a medium level of abstraction **close to the math** (not too low-level, but no "black boxes" either)

Homework 0 (Optional)

- Do online tutorials to familiarize yourself with a programming language (preferably Python). See course website for resources.
- First problem set will be posted Wednesday; due the following Wednesday night
- Chance to get up to speed with/review calculus, coding, a bit of linear algebra basics
- You will need to **look things up for yourself frequently**; helpful links/references on the website