CSCI567 Machine Learning (Spring 2021)

Sirisha Rambhatla

University of Southern California

Jan 15, 2021

Outline

1. About this course
2. Overview of machine learning
3. Mathematical Foundations

Outline

1. About this course
2. Overview of machine learning
3. Mathematical Foundations

Overview

Nature of this course

- Covers standard statistical machine learning methods (supervised learning, unsupervised learning, etc.)
- Particular focuses are on the conceptual understanding and derivation of these methods

Learning objectives:

- Hone skills on grasping abstract concepts and thinking critically to solve problems with machine learning techniques
- Solidify your knowledge with hand-on programming tasks
- Prepare you for studying advanced machine learning techniques

Teaching logistics

We will divide the allotted time on WF 10:00-11:50 AM as follows:

Lectures: WF 10:00-11:10 AM

Discussions: WF 11:10-11:50 AM (by TAs)

DEN@Viterbi/D2L:

- Use "Virtual Meetings" tab at the CSCI-567 page at https://courses.uscden.net to access the meeting link
- feel free to unmute and ask questions (avoid chat box)
- be patient if connection is lost
- let me know if you have any comments

Online platforms

Course website: `https://courses.uscden.net`

- general information (schedule, slides, etc.)
- homework release and submissions
- recorded lectures/discussions
- submit written assignments
- grade posting

Piazza: `https://piazza.com/class/kjkinvvwzi12mp`

- Also on DEN@Viterbi/D2L platform
- main discussion forum
- everyone has to enroll

Kaggle (for course project)

Teaching staff

2 TAs (lecture/discussion, quiz, etc.)

- Liyu Chen `liyuc@usc.edu`
- Karishma Sharma `krsharma@usc.edu`

2 CPs (homework, project, etc.)

- Dhiti Thakkar `dhitisam@usc.edu`
- Prateek Jain `jainp@usc.edu`

Office hours are on Piazza→Resources→Staff

Prerequisites

- Undergraduate level training in **probability and statistics, linear algebra, (multivariate) calculus**
- Programming: Python and necessary packages (e.g. numpy)

  not an intro-level CS course, no training of basic programming skills.

Slides and readings

Lectures
Lecture slides/handouts will be posted before the class (and possibly updated after)[1].

Readings

- No required textbooks
- Main recommended readings:
  - Machine Learning: A Probabilistic Perspective by Kevin Murphy
  - Elements of Statistical Learning by Hastie, Tibshirani and Friedman
- More: see course website

[1] Special thanks to Prof. Haipeng Luo and Prof. Yan Liu for the course material!

Grade

Structure:

- 40%: 5 written assignments
- 30%: 2 quizzes
- 30%: 1 Kaggle-based course project

Initial cut-offs (for A and B):

- B- = [70,75), B = [75, 80), B+ = [80, 85)
- A- = [85, 90), A = [90, 100]

Important: final cut-offs will NOT be released. If adjusted they could only be LOWER.

Homework

5 written assignments (problem sets):

- submit one pdf to D2L (scanned copy or typeset with LaTeX etc.)
- graded based on correctness
- collaboration is permitted at high-level but must be stated (each member has to make a separate submission)
- Copying solutions from any sources → *zero grade*.
- 3 late days in total, at most *one* can be used for each assignment
- A two-day window for re-grading (regarding *factual errors*)

Course Project

Done on **Kaggle**

- Groups of 2-3 students (we randomly assign you team-mates)
- Same project assigned to all groups
- Deliverables: A progress update, and a final 4 page write-up
- Grading based on number of submissions, ranking and the deliverables.
- More details to come as the semester progresses.

Quizzes

First one on **03/03**, second one on **05/10** (final).

- Quiz 1: in class, 10:00-11:50 AM,
- Quiz 2 (final), scheduled for 8:00-10:00 AM; see `https://classes.usc.edu/term-20211/finals/`.
- open-book, no collaboration or consultation from others allowed
- Details will be discussed closer to the quiz date, **the dates are tentative**.

Academic integrity

Plagiarism and other unacceptable violations

- neither ethical nor in your self-interest
- zero-tolerance
- check `https://viterbischool.usc.edu/academic-integrity/` for a complete list

Outline

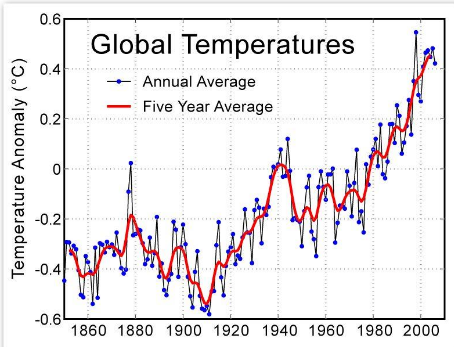
What is machine learning?

One possible definition[2]
a set of methods that can automatically *detect patterns* in data, and then use the uncovered patterns to *predict future data*, or to perform other kinds of decision making *under uncertainty*

cf. Murphy's book

Example: detect patterns

How the temperature has been changing?

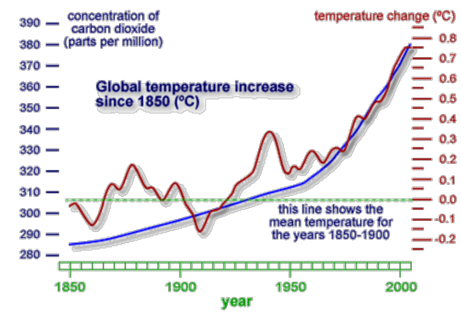


Patterns

- Seems going up
- Repeated periods of going up and down.

How do we describe the pattern?

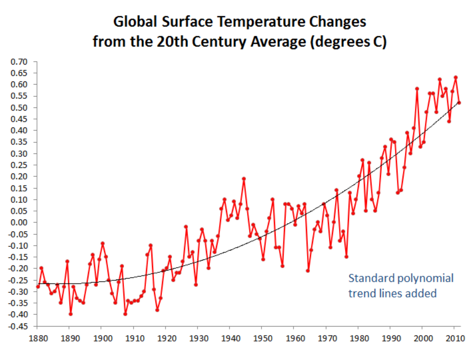
Build a model: fit the data with a polynomial function



- The model is not accurate for individual years
- But collectively, the model captures the major trend

Predicting future

What is temperature of 2010?



- Again, the model is not accurate for that specific year
- But then, it is close to the actual one

What we have learned from this example?

Key ingredients in machine learning

- Data
  collected from past observation (we often call them *training data*)
- Modeling
  devised to capture the patterns in the data
  - The model does not have to be true — "All models are wrong, but some are useful" by George Box.
- Prediction
  apply the model to forecast what is going to happen in future

A rich history of applying statistical learning methods

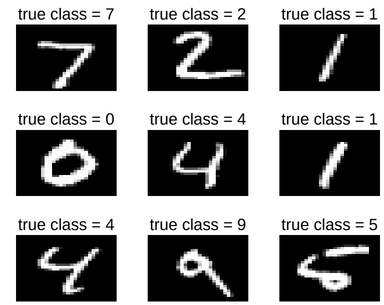
Recognizing flowers (by R. Fisher, 1936)

Types of Iris: setosa, versicolor, and virginica



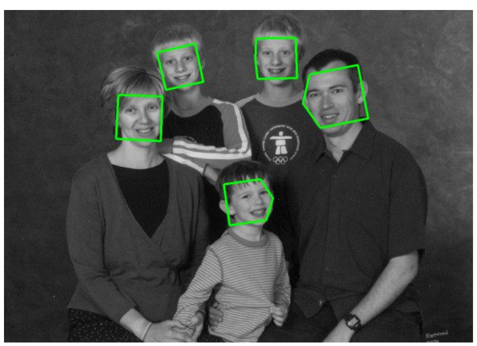
Huge success 30 years ago

Recognizing handwritten zipcodes (AT&T Labs, late 1990s)



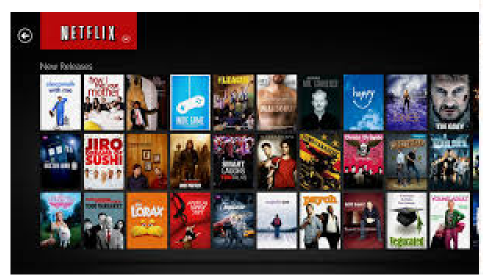
More modern ones, in your social life

Recognizing your friends on Facebook



It might be possible to know about you than yourself

Recommending what you might like



Why is machine learning so hot?

- **Tons of consumer applications**:
  - speech recognition, information retrieval and search, email and document classification, stock price prediction, object recognition, biometrics, etc
  - Highly desirable expertise from industry: Google, Facebook, Microsoft, Uber, Twitter, IBM, Amazon, $\cdots$
- **Enable scientific breakthrough**
  - Climate science: understand global warming cause and effect
  - Biology and genetics: identify disease-causing genes and gene networks
  - Social science: social network analysis; social media analysis
  - Business and finance: marketing, operation research
  - Emerging ones: healthcare, energy, $\cdots$

What is in machine learning?

Different flavors of learning problems

- Supervised learning
  Aim to predict (as in previous examples)
- Unsupervised learning
  Aim to discover hidden and latent patterns and explore data
- Decision making (e.g. reinforcement learning)
  Aim to act optimally under uncertainty
- Many other paradigms

The focus and goal of this course

- Supervised learning (before Quiz 1)
- Unsupervised learning (after Quiz 1)

Outline

How to grasp machine learning well

Three pillars to machine learning[3]

- Probability, Statistics and Information Theory
- Linear Algebra and Matrix Analysis
- Optimization

Resources

- Suggested Reading:
  - All of Statistics Page 21-89
  - Murphy's textbook
  - The Matrix Cookbook (a great resource!)
    `www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf`
- There are other great resources/visualizations available online
- If you find a neat explanation for something be sure to share with all of us in the "useful links" thread on piazza

Quote from Prof. Michael I. Jordan

Probability: basic definitions

Sample Space: a set of all possible outcomes or realizations of some random trial.

Example: Toss a coin twice; the sample space is $\Omega = \{HH, HT, TH, TT\}$.

Event: A subset of sample space

Example: the event that at least one toss is a head is $A = \{HH, HT, TH\}$.

Probability: We assign a real number $P(A)$ to each event A, called the probability of A.

Probability Axioms: The probability P must satisfy three axioms:

1. $P(A) \geq 0$ for every A;
2. $P(\Omega) = 1$;
3. If $A_1, A_2, \ldots$ are disjoint, then $P(\cup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P(A_i)$

Random Variables

Definition: A random variable is a measurable function that maps from a probability space to a measurable space, i.e. $X : \Omega \to R$, that assigns a real number $X(\omega)$ to each outcome $\omega \in \Omega$.

Two Types: Discrete (e.g. Bernoulli in Coin toss) and Continuous (e.g. Gaussian)

Data and Statistics The data are specific realizations of random variables; A statistic is just any function of the data or random variables, e.g. mean, variance etc.

Distribution Function

Definition: Suppose X is a random variable and x is a specific value that it can take, then

For discrete r.v. X, the *probability mass function* is defined as

$$f_X(x) = P(X = x)$$

For continuous r.v. X, $f_X(x) \geq 0$ is the *probability density function* if for every $a \leq b$

$$P(a \leq X \leq b) = \int_a^b f(x)dx$$

where $\int_{-\infty}^{\infty} f(x)dx = 1$. Note: for continuous distributions $P(X = x) = 0$!

Cumulative distribution function (CDF) of X : $F_X(x) = P(X \leq x)$. If $F(x)$ is differentiable everywhere, $f(x) = F'(x)$.

Expectation

Expected Values

- Of a function $g(\cdot)$ of a discrete random variable X,

$$E[g(X)] = \sum_{x \in \mathcal{X}} g(x)f(x);$$

- Of a function $g(\cdot)$ of a continuous random variable X,

$$E[g(X)] = \int_{-\infty}^{\infty} g(x)f(x).$$

Mean and Variance $\mu = E[X]$ is the mean; $var[X] = E[(X - \mu)^2]$ is the variance. We also have $var[X] = E[X^2] - \mu^2$.

Multivariate Distributions

Definition:

$$F_{X,Y}(x,y) := P(X \leq x, Y \leq y),$$

and

$$f_{X,Y}(x,y) := \frac{\partial^2 F_{X,Y}(x,y)}{\partial x \partial y},$$

Marginal Distribution of X (Discrete case):

$$f_X(x) = P(X = x) = \sum_y P(X = x, Y = y) = \sum_y f_{X,Y}(x,y)$$

or $f_X(x) = \int_y f_{X,Y}(x,y)dy$ for continuous variable.

Conditional probability of X given $Y = y$ is

$$f_{X|Y}(x|y) = P(X = x|Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)} = \frac{f_{X,Y}(x,y)}{f_Y(y)}$$

Bayes Rule

Law of total Probability: X takes values $x_1, \ldots, x_n$ and y is a value of Y, we have

$$f_Y(y) = \sum_j f_{Y|X}(y|x_j) f_X(x_j)$$

Bayes Rule:
(Simple Form)

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$$

(Discrete Random Variables)

$$f_{X|Y}(x_i|y) = \frac{f_{Y|X}(y|x_i) f_X(x_i)}{\sum_j f_{Y|X}(y|x_j) f_X(x_j)}$$

(Continuous Random Variables)

$$f_{X|Y}(x|y) = \frac{f_{Y|X}(y|x) f_X(x)}{\int_x f_{Y|X}(y|x) f_X(x) dx}$$

Independence

Independent Variables X and Y are *independent* if and only if:

$$P(X = x, Y = y) = P(X = x)P(Y = y)$$

or $f_{X,Y}(x,y) = f_X(x)f_Y(y)$ for all values x and y.

IID variables: *Independent and identically distributed* (IID) random variables are drawn from the same distribution and are all mutually independent.

If $X_1, \ldots, X_n$ are independent, we have

$$E[\prod_{i=1}^n X_i] = \prod_{i=1}^n E[X_i], \quad var[\sum_{i=1}^n a_i X_i] = \sum_{i=1}^n a_i^2 var[X_i]$$

Linearity of Expectation: Even if $X_1, \ldots, X_n$ are not independent,

$$E[\sum_{i=1}^n X_i] = \sum_{i=1}^n E[X_i].$$

Correlation

Covariance

$$cov(X,Y) = E[(X - \mu_x)(Y - \mu_y)],$$

Correlation coefficients

$$corr(X,Y) = Cov(X,Y)/\sigma_x \sigma_y$$

Independence $\Rightarrow$ Uncorrelated ($corr(X,Y) = 0$).

However, the reverse is generally not true.
The important special case: multi-variate Gaussian distribution.

Statistics

Suppose $X_1, \ldots, X_n$ are random variables:
Sample Mean:

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

Sample Variance:

$$S_{N-1}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \bar{X})^2.$$

If X_i are iid:

$$E[\bar{X}] = E[X_i] = \mu,$$
$$Var(\bar{X}) = \sigma^2/N,$$
$$E[S_{N-1}^2] = \sigma^2$$

Point Estimation

Definition The *point estimator* $\hat{\theta}_N$ is a function of samples $X_1, \ldots, X_N$ that approximates a parameter θ of the distribution of X_i.

Sample Bias: The bias of an estimator is

$$bias(\hat{\theta}_N) = E_\theta[\hat{\theta}_N] - \theta$$

An estimator is *unbiased estimator* if $E_\theta[\hat{\theta}_N] = \theta$

Standard error The standard deviation (i.e. the square-root of variance) of $\hat{\theta}_N$ is called the *standard error*

$$se(\hat{\theta}_N) = \sqrt{Var(\hat{\theta}_N)}.$$

Review of Information Theory

Suppose X can have one of the m values: $x_1, \ldots, x_m.$, and the probability $P(X = x_i) = p_i$.

Entropy is the average amount of *surprise* in a r.v.'s outcome.

$$H(X) = -\sum_{j=1}^{m} p_i \log p_i$$

- "High entropy" means X is from a uniform (boring) distribution;
- "Low entropy" means X is from varied (peaks and valleys) distribution.

Information Theory

Conditional Entropy is the remaining entropy of a random variable Y given that the value of another random variable X is known.

$$H(Y|X) = \sum_{i=1}^{m} p(X = x_i)H(Y|X = x_i) = -\sum_{i=i}^{m}\sum_{j=1}^{n} p(x_i, y_j)\log p(y_j|x_i)$$

Mutual Information: if Y must be transmitted, how many bits on average would be saved if both ends of the line knew X?

$$I(Y; X) = H(Y) - H(Y|X).$$

Notice that $I(Y; X) = I(X; Y) = H(X) + H(Y) - H(X, Y)$

Kullback-Leibler divergence is a measure of distance between two distributions: a "true" distribution $p(X)$, and an arbitrary distribution $q(X)$.

$$\mathsf{KL}(p||q) = \sum_x p(x)\log\frac{p(x)}{q(x)}$$

We can write $I(X; Y) = KL(p(x, y)||p(x)p(y))$.

Optimization

Definition: Optimization refers to choosing the best element from some set of available alternatives. A general form is as follows:

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, i = 1, \ldots, m \\ & h_i(x) = 0, i = 1, \ldots, p. \end{aligned} \tag{1}$$

Difficulties:

1. Local or global optimum?
2. Difficulty to find a feasible point,
3. Stopping criteria,
4. Poor convergence rate,
5. Numerical issues

Convex Optimization

Convex Functions: if for any two points $x_1, x_2 \in X$ and any $t \in [0, 1]$,

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2).$$

A function f is said to be *concave* if $-f$ is convex.

Convex Set a set S is convex if and only if for any $x_1, x_2 \in S$, $tx_1 + (1-t)x_2 \in S$ for any $t \in [0, 1]$,

Convex Optimization is minimization (maximization) of a convex (concave) function over a convex set, e.g., Linear Programming (LP), Quadratic Programming (QP), and Semi-Definite Programming (SDP).

Popular convex optimization algorithms:

- Gradient descent
- Conjugate gradient
- Newton's method
- Quasi-Newton method
- Subgradient method