COMPUTER SCIENCE
INDIANA UNIVERSITY
School of Informatics and Computing
Bloomington

Machine Learning

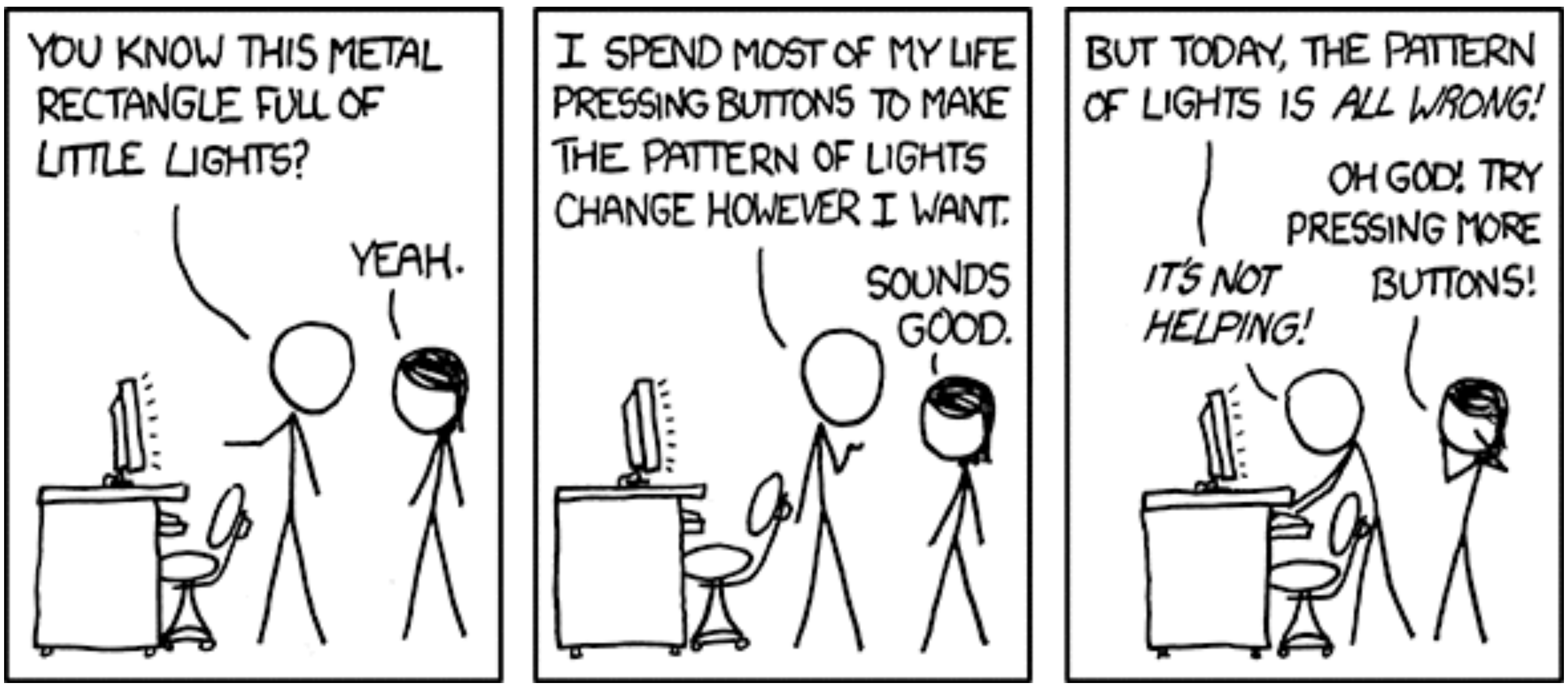
CSCI-B 555
Fall 2015
Martha White

What is this course about?

- The world is full of information and data
- Much of that data is noisy and has an element of uncertainty
  - We have incomplete knowledge of the environment
  - Actions of other actors not provided
- **Goal**: understand machine learning algorithms to analyze this data by deriving them from the beginning
  - our focus will be on prediction (on new data)

What is this course about?

- **One line goal:** demystify the seemingly huge collection of machine learning algorithms by exploring their mathematical foundation



What is this course about?

- The focus is on fundamental data analysis/statistics concepts, and not necessarily on application of these algorithms
  - though you will get to do that too
- Application of algorithms is simpler when you understand their development and underlying assumptions

Basic information

Class meets:
Time: MW 11:15pm – 12:30pm
Place: Informatics East, Room 130

Instructor:
Martha White
Office: Lindley Hall 401E
Email: martha@indiana.edu
Web: http://homes.soic.indiana.edu/martha/

Office Hours:
Time: M 2:00pm-4:00pm (is this a good time?)
or by appointment
Place: Lindley Hall 401E

Class Web Site:
https://iu.instructure.com/courses/1480234

Associate Instructors (Teaching Assistants)

Shantanu Jain

Email: shajain@indiana.edu
Office: LH 401A
Times: M 1:30pm-3:00pm

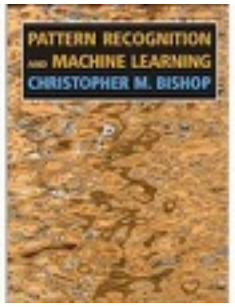
Zeeshan Ali Sayyed

Email: zasayyed@indiana.edu
Office: LH 401A
Times: W 1:00pm-2:30pm

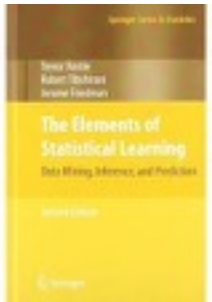
Textbook information

- **Main notes** provided in Canvas
    - written by Predrag Radivojac and myself (about 130 pages)

- **Additional reference material/Recommended readings:**
    - Pattern Recognition and Machine Learning
      by C. M. Bishop, Springer 2006.



    - The Elements of Statistical Learning
      by T. Hastie, R. Tibshirani, and J. Friedman, 2009



Grading policy

- 40%: Assignments (4), mostly mathematical with some small programming exercises
- 25%: Class project, (likely) solving a problem on Kaggle
- 25%: Final exam, December 16, 12:30 p.m. - 2:30 p.m.
- 10%: Thought questions (4), show that you're reading and thinking about the material

More on grading

- Top performers will get A+
  - This course is not curved; I decide the letter thresholds (Note that I don't enjoy having to assign you a grade, but it helps you learn)
- I'll try to maintain a distribution of scores; Canvas should show you at least your grades and the average
- **All assignments are individual**; you can chat about the material, but follow the "in your head" rule: leave with everything in your head, and not recorded/written down
- Acknowledge sources (classmates, websites, books) in your typed up documents
  - assignments are in LaTex; I have posted examples/tutorials on Canvas

Submission policy

- Homework assignments are due on the date specified in Canvas
  - always Wednesday, at 11:59 p.m.
- Submit assignments, thought questions and project in Canvas
- Project and thought questions cannot be submitted late
- Late assignments will be accepted according to the following rules
  - **points** **(on time)** **recommended!**
  - points x 0.9 (1 day late)
  - points x 0.7 (2 days late)
  - points x 0.5 (3 days late)
  - points x 0.3 (4 days late)
  - points x 0.1 (5 days late)
  - 0 (after 5 days)

  not recommended!
- There can be legitimate issues; if something is wrong, talk to me early

Academic honesty

- I have to report every cheating incident to the university
- Code of student rights, responsibilities and conduct: http://www.indiana.edu/~code/
  - e.g. Students are responsible to "facilitate the learning environment and the process of learning, including attending class regularly, completing class assignments, and coming to class prepared".
- You are in graduate school, so I hope one of your priorities is to learn and become a more independent thinker

My expectations

- Basic mathematical skills
    - calculus
    - probabilities
    - linear algebra
- You are hardworking and motivated to learn (machine learning)
- You are motivated to succeed in class
- You are motivated to think beyond the material and ask open-ended questions (one of the goals for graduate school)

What to expect from this course

- I know you have expectation of me too
  - I will try to be transparent in marking and course choices
  - I am here to help you learn; I will treat you with respect and listen thoughtfully to your questions (I love to answer questions and give advice!)
  - Feel free to give me feedback (e.g., Miss Martha, you are talking too fast)
  - I will provide an mechanism for anonymous feedback
- This course will be quite mathematical, with derivations of details
  - sometimes I'll say "we need to eat our broccoli" or "pain is good for you"
- By the end of the course, you should have a good grasp of fundamental concepts in ML and algorithm derivation for ML

Thought questions

- Graduate school is about thinking and asking questions
  - maybe surprisingly, research is actually about asking good questions
  - a question does not have to have an answer, but it can be thought-provoking or make you think about a topic differently
- “Thought questions” correspond to (short) readings in the notes, and should demonstrate you’ve read and thought about the topics
- There are no stupid questions (so ask any questions in class/office hours/email), but “thought questions” are to demonstrate insight and so I have some requirements

Examples of “good” thought questions

- After reading about independence, I wonder how one could check in practice if two variables are independent, given a database of samples? Is this even possible?
- Where do the axioms of probability come from? Why does p have to satisfy these properties and what would it mean to add other ones? What would it mean to drop a required property on the probability function?
- “In the given example about expected utility, the parameters (+10 for correct and -20 for incorrect) seems a bit arbitrary. What are common decision making tools practitioners use once outcomes are known?” — student from B554
- “Why not always use gradient descent, as we did to learn the hyperparameters for Bayesian linear models? Why do we ever use EM?” — student from B554
- Why is PCA so widely used? It seems simple and I would not expect it to be able to encode complex properties.

Examples of “bad” thought questions

- I don’t understand linear regression. Could you explain it again? (i.e. a request for me to explain something, without any insight)
- Derive the maximum likelihood approach for a Gaussian. (i.e., an exercise question from a textbook)
- What is the difference between a probability mass function and a probability density function? (i.e., a question that could easily be answered from reading the definitions in the notes)
- How are Boltzmann machines and feedforward neural networks different? (i.e., again a definition)
    - But the following modification would be good: “I can see that Boltzmann machines and feedforward neural networks are different, in that the first is undirected and the second directed. How does this difference impact modeling properties and accuracy of estimation in practice?”

What is machine learning?

- We could label it in many different ways including
  - Data-driven approach to artificial intelligence
  - Applied statistics

- Some keywords associated with machine learning
  - Prediction, probability, samples, data, function, optimization, stochastic, …

ICML 2015
word cloud

Where did machine learning come from?

- Let's first step back to the goals of artificial intelligence
- Many original AI approaches were expert-based
  - logic approaches for theorem proving
  - expert systems
- Machine learning arose as a data-based approach to solve artificial intelligence problems
  - why the shift? increased computation, availability of data and efficacy of data driven approaches (largely driven by availability of data)

What are the ultimate goals?

- The focus for many ML researchers has shifted from AI towards generally solving important (practical) problems
  - computer vision, speech recognition, clustering, modeling temporal data, …
- This includes a focus on understanding intrinsic properties of a learning problem
  - is it difficult to learn (e.g., NP-hard)?
  - how can it be formulated in a precise way? (e.g., convex objective for sparse coding, preference for “simpler” hypotheses)
  - how many samples are needed to learn the model (epsilon) accurately?
  - how well does the learned model generalize to new samples?

How do learning problems differ?

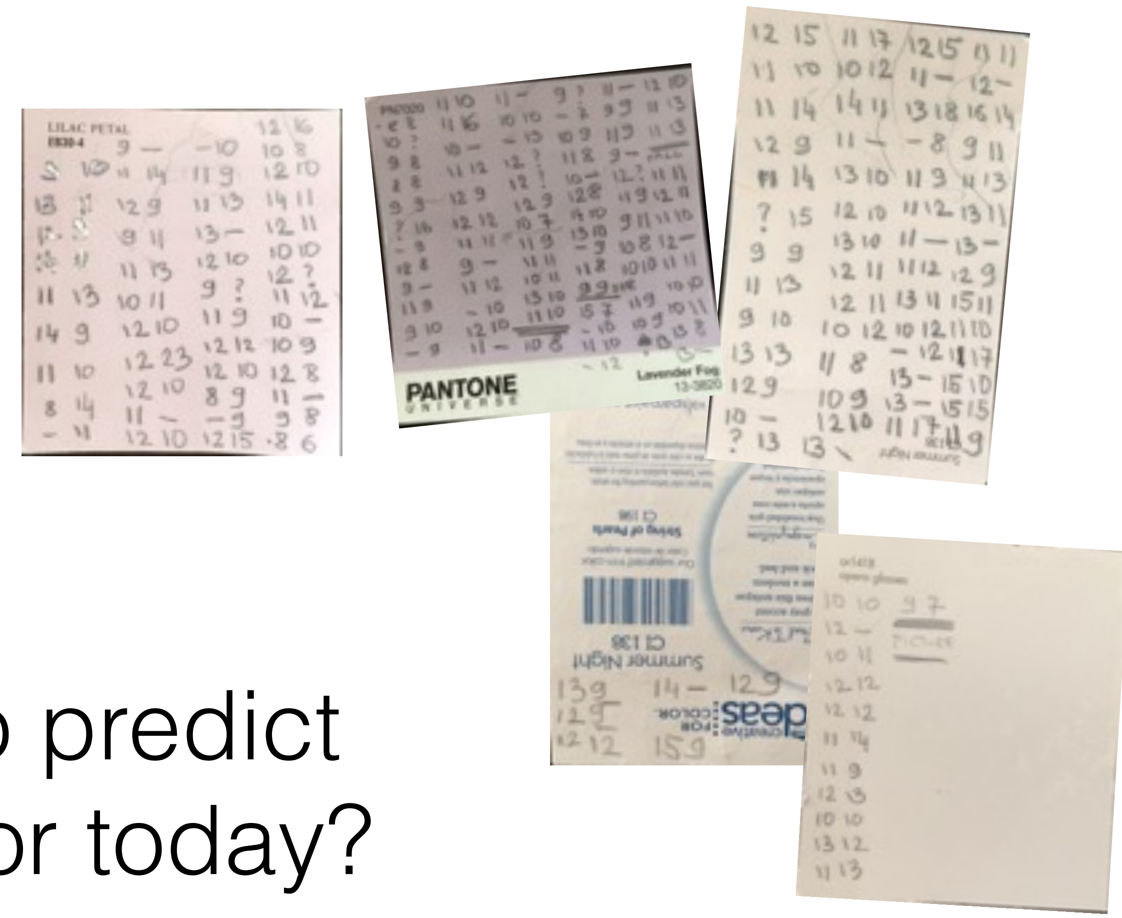
- They can be categorized across several dimensions
  - **Control versus prediction:** though a control algorithm will likely use predictions to improve decision-making (e.g., reinforcement learning)
  - **Supervised and unsupervised:** supervised learning is for prediction, unsupervised learning is usually for visualization or representation learning; some algorithms combine these two important components
  - **How they are used:** empower decision-making of end user OR autonomously control system
  - … there are more, but these are some main ones

- **Algorithms** also differ in many ways, even for similar problems
  - e.g., process data incrementally (as stream) or in batch; low computation (or memory) versus heavy computation (or memory); approximate or exact

Let’s look at some fun examples!

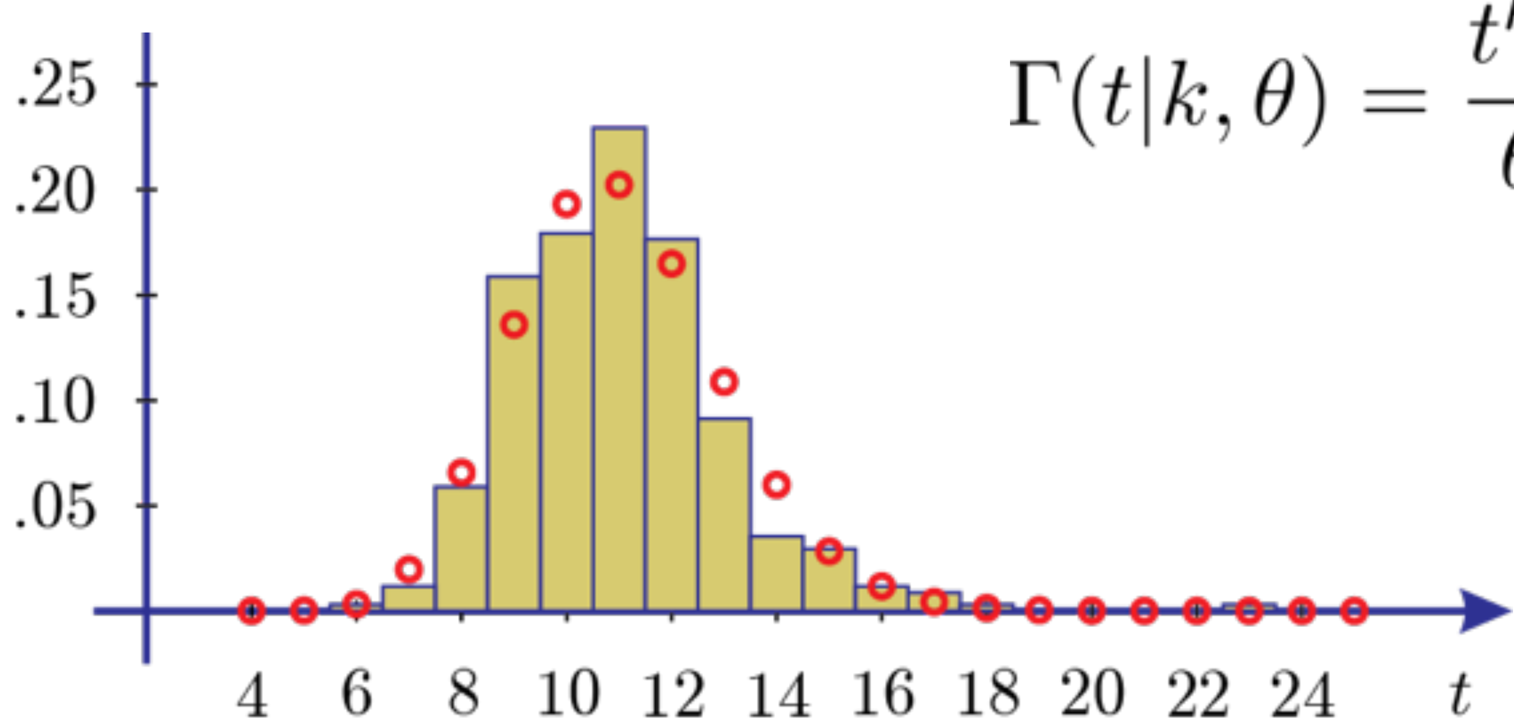
- Commute times (mostly independent data, iid data)
- Weather prediction (temporally connected data)
    - machine learning is often used for time series, but in the specific case of weather, mostly expert models appear to be used (for now…)
- Robotics and octopus arm simulator (machine learning for control)
    - as a caveat, we will not look at control algorithms; however, their development uses the fundamental concepts in this course
    - check out my Stochastic optimization for ML seminar in Spring for control

Commute times



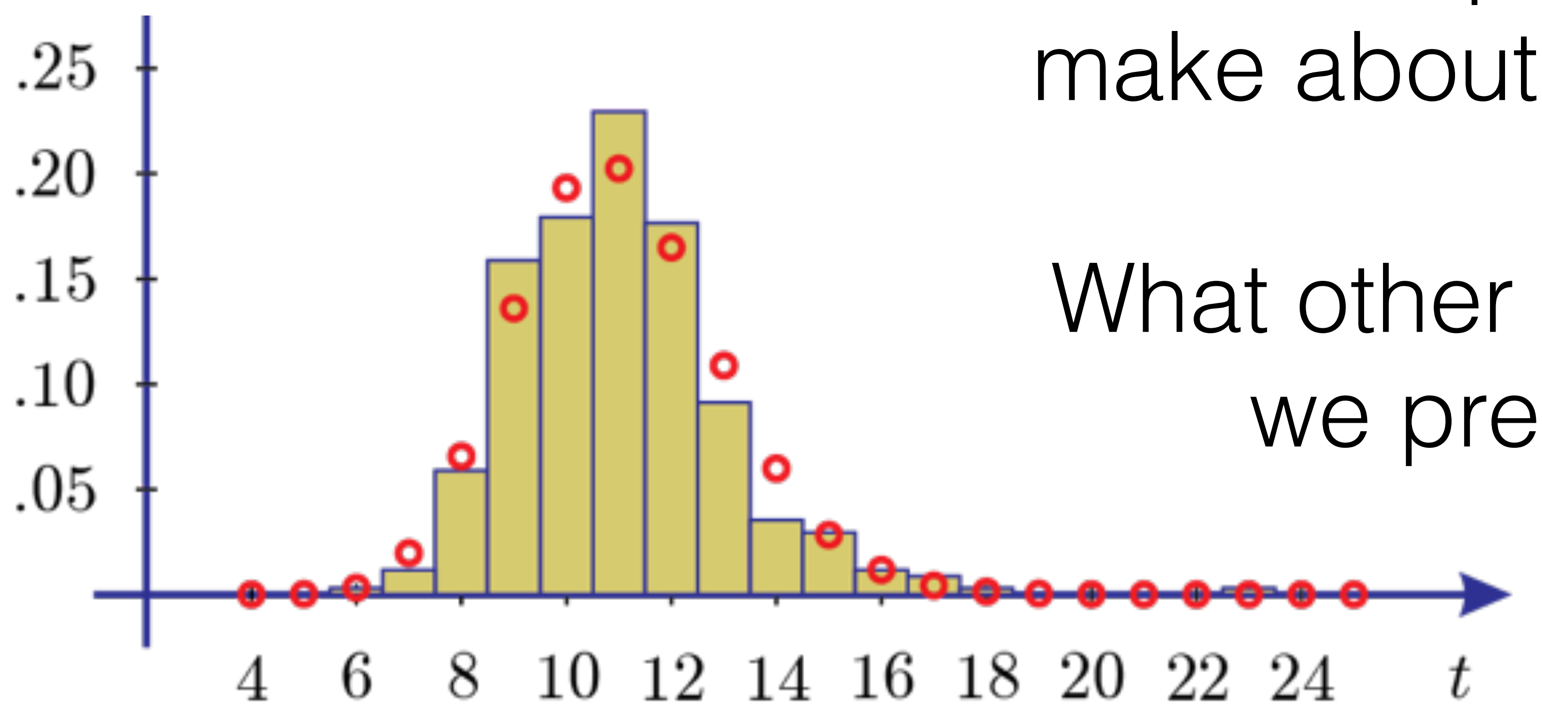
How might you try to predict your commute time for today?

Commute times (2)



$$\Gamma(t|k,\theta) = \frac{t^{k-1}e^{-\frac{t}{\theta}}}{\theta^k\Gamma(k)}$$

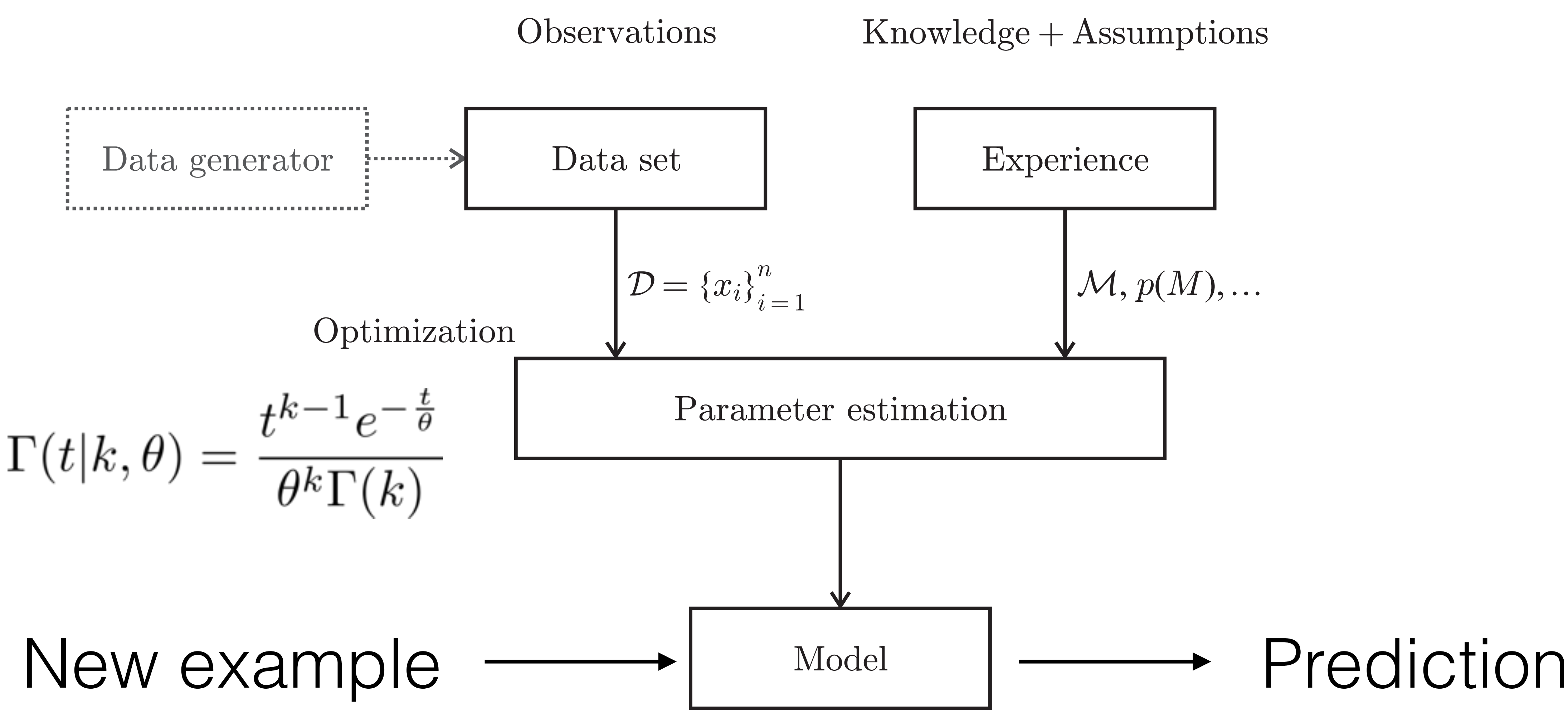
Commute times (3)



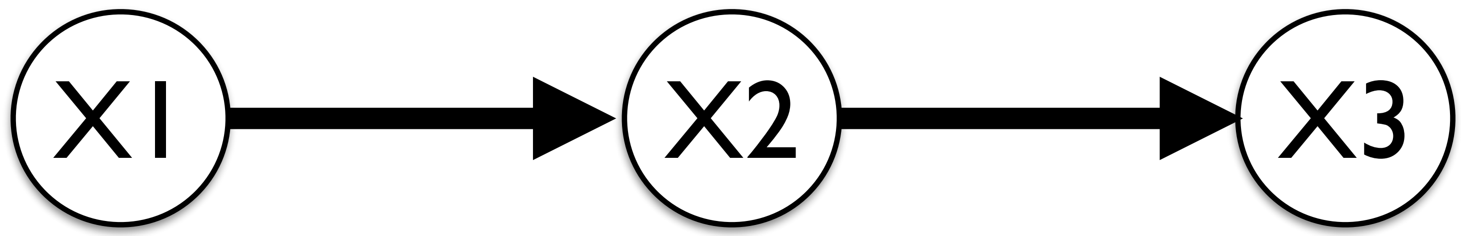
What assumptions can we make about the data?

What other ways can we predict?

Summarized flow



Weather prediction (time series)

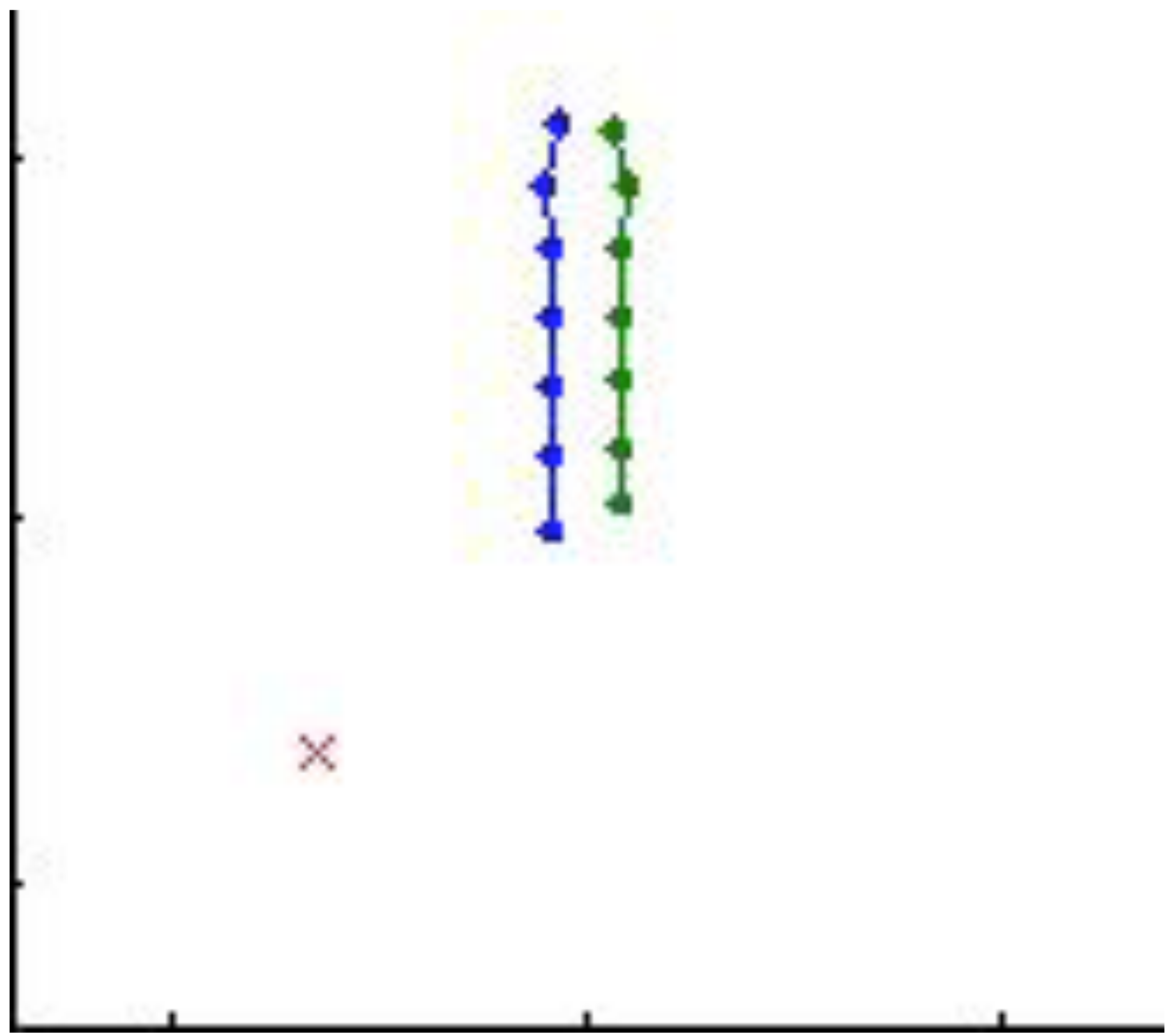


- Imagine we want to predict the probability of rain tomorrow, 2 days from now, 3 days, …
- One common strategy for time series is to use the last p points as features to predict the next point, 2 points into future, etc.
- What other strategies can you imagine?
- How do you predict a probability value, rather than say a binary value (0 or 1) or a real value?
  - Hint: these are things we will learn

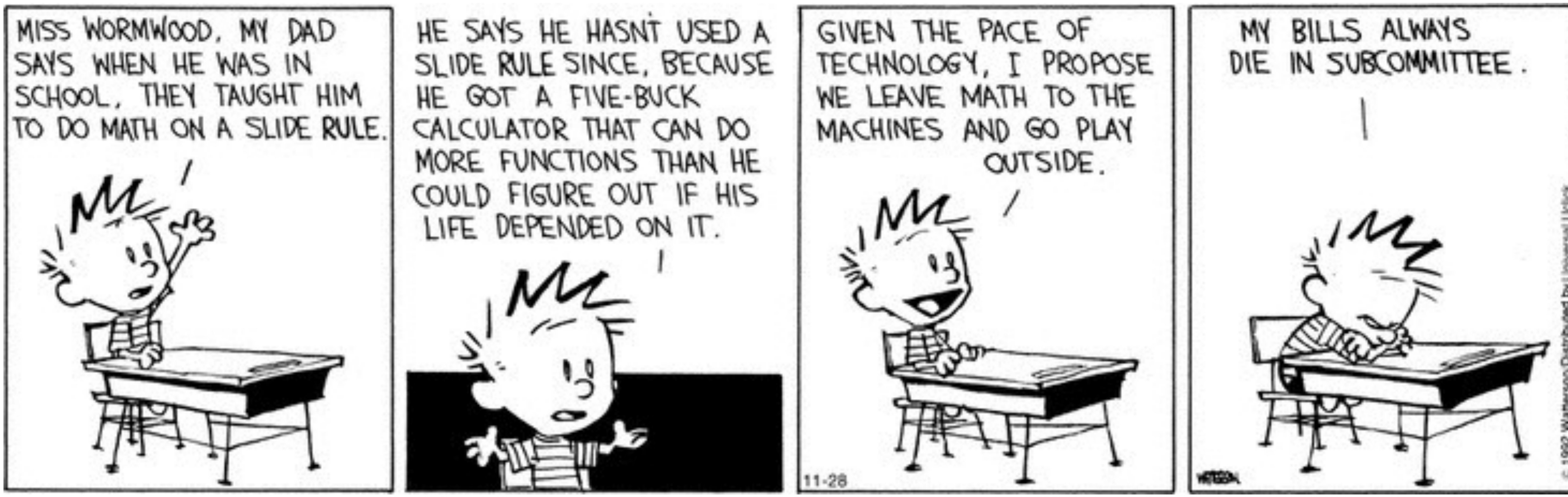
Simple robot (control)

- https://www.youtube.com/watch?v=CkTftoNFeGY
  - time 2:50 to 4:10
- An example of a simple implementation of RL algorithms (not heavily engineered, heavily data-based)
  - something you could likely implement after an intro RL course
  - goal to work towards more autonomous systems
- Though simple, does input a relatively large number of sensors

Octopus arm (control)



Next: crash course in probability



- Probabilities underly much of machine learning
    - enable precise modeling of uncertainty

Topic overview

- Parameter estimation and prediction problems (Chapters 2 and 3)
    - the core background for modeling in machine learning
- Linear regression (Chapter 3)
- Generalized linear models (Chapter 4)
- Linear classifiers (Chapter 5)
- Representations for ML (Chapter 6)
    - help make linear predictors more powerful (e.g., neural networks)
- Empirical evaluation of learning algorithms (Chapter 7)
- … and any other topics listed on the syllabus if we have time (e.g., boosting)