# Logistics and Introduction to the Course

Piyush Rai

Topics in Probabilistic Modeling and Inference (CS698X)

Jan 4, 2018

# Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - "TPMI" or just "PMI"
- **Timing and Venue:** Tue/Th 5:10-6:30pm, KD-101
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in; please prefix email subject with CS698X)
- **Instructor Office Hours:** Wednesday 11:00-12:00pm (RM-502), or by appointment.
- **Course website:** `https://goo.gl/iJBkh7` (where course material and readings will be posted)
- **Discussion site:** Piazza (`https://goo.gl/UkbX1m`). Please register and participate actively.
- **Attendance Policy:** None, but missing too many classes will make it hard for you to catch up with the material.
- **Auditing?** Please let me know your email id to be added to the mailing list. You may also submit homeworks and appear for exams (we'll try to grade but cannot guarantee)

# TAs and Course/Project Mentors

- **TAs:** Shivam Bansal, Smrithi Prabhu, Vinay Verma



- **Course/Project Mentors:** Gundeep Arora, Nikhil Mehta, Govind Gopakumar, Archit Sharma, Nishit Asnani, Rishabh Goyal

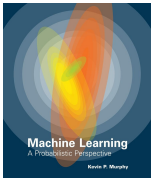


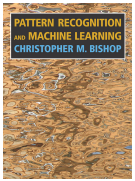



- TA/mentor office hours/locations and contact details will soon be posted on class webpage/Piazza

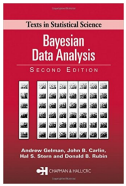
# Textbook and Readings

- **Textbook:** No official textbook required. Required reading material will be provided.
- Some books that you may use as reference
  - Kevin Murphy, Machine Learning: A Probabilistic Perspective (MLAPP), The MIT Press, 2012.
  - Christopher Bishop, Pattern Recognition and Machine Learning (PRML), Springer, 2007.
  - David Barber. Bayesian Reasoning and Machine Learning (BRML), Cambridge Univ. Press, 2012.
  - Andrew Gelman *et al*. Bayesian Data Analysis (BDA), Chapman & Hall/CRC, 2013








- Several other monographs on some specific topics will be provided for readings

# Grading Policy

- 5 homework assignments: 30%
  - Pen and paper based + some programming in MATLAB/Python
- Midterm exam: 20%
- Final exam: 30%
- Class Project: 20%
  - To be done in groups of 3
  - More details forthcoming (by next week)
  - Exceptional work in the course project may earn you a direct A grade
- Note: Exams will be closed-book (you will be provided a cheat-sheet)

# Course Project

- A list of possible project ideas will soon be shared
  - Many options: Research project, implementation of research papers, survey of a topic, etc.
- Project mentors can be another source for project suggestions
- You can also propose your own idea (but discuss the scope/feasibility with me first)
- Some deadlines
  - Project proposal: Worth 5%. Due on Feb 1
  - In-class final presentation: Worth 5%. During last week of classes
  - Final report: Worth 10%. Due end of semester.

# Collaboration vs Cheating

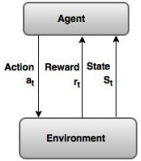
- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- However, you are supposed to write your own solution in your own words (same goes for coding assignments)
- Plagiarism from other sources (for assignments/project) will also lead to strict punishment unless you duly credit the original source(s)
- Other things that will lead to punishment
  - Use of unfair means in the exams
  - Fabricating experimental results in assignments/project
- Important: Both copying as well as helping someone copy will be equally punishable

# Probabilistic Modeling and Inference

# Why Probabilistic Modeling?

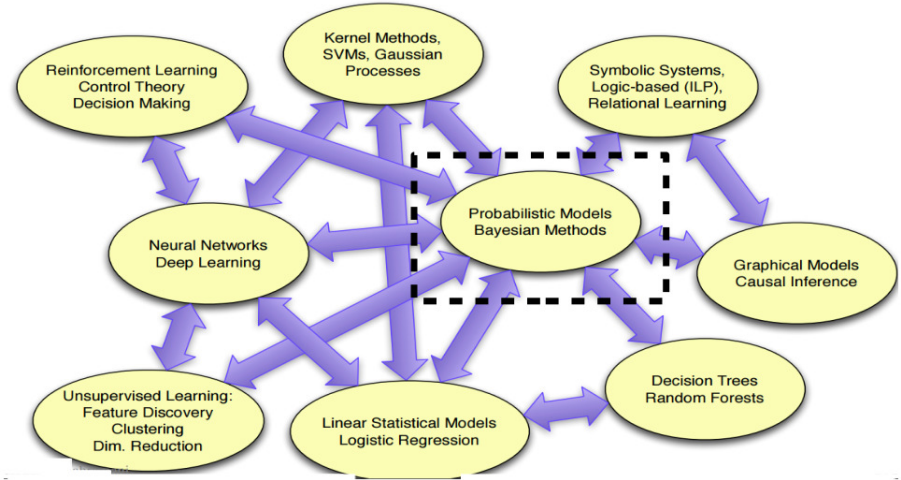
- Real-world data is noisy and uncertain (and often limited if data acquisition is expensive/difficult)

| | Item 1 | Item 2 | Item 3 | ... | Item n |
|---|---|---|---|---|---|
| User 1 | 2 | 3 | ? | ... | 5 |
| User 2 | ? | 4 | 3 | ... | ? |
| User 3 | 3 | 2 | ? | ... | 3 |
| ... | ... | ... | ... | ... | ... |
| User m | 1 | ? | 5 | ... | 4 |




- Consequently, inferences drawn or predictions made using such data will also be uncertain
- Important to take these aspects into account while modeling the data and making inferences
- Probabilistic modeling provides a "language" to handle this naturally by:
  - Modeling the observed data and all other unknowns via a joint probability distribution
  - Using rules of probability to make inferences about the unknowns given the observed data
- Broadly applicable to a wide range of domains, such as Vision, NLP, computational biology, robotics, finance, signal processing, and many more..

# Why Probabilistic Modeling?




Original figure courtesy: Zoubin Ghahramani

# The Simple Framework behind Probabilistic Modeling & Inference

- Based essentially on two simple rules of probability: sum rule and product rule

$$P(a) = \sum_b P(a,b) \quad \text{(Sum Rule)}$$
$$P(a,b) = P(a)P(b|a) = P(b)P(a|b) \quad \text{(Product Rule)}$$

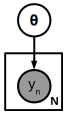
- Note: For continuous random variables, sum is replaced by integral: $P(a) = \int P(a,b)db$
- Another rule is the Bayes rule (can be easily obtained from the above two rules)

$$P(b|a) = \frac{P(b)P(a|b)}{P(a)} = \frac{P(b)P(a|b)}{\int P(a,b)db} = \frac{P(b)P(a|b)}{\int P(b)P(a|b)db}$$

- All of probabilistic modeling and inference is based on consistently applying these two simple rules

# Starting off: A Toy Probabilistic Model

- $N$ observations $\boldsymbol{y} = \{y_1, y_2, \ldots, y_N\}$, generated by a probabilistic model with unknown params $\theta$




- Let's treat both $\boldsymbol{y}$ and $\theta$ as random variables with a joint probability distribution $p(\boldsymbol{y}, \theta)$
- Note that we can factorize this joint distribution as

$$p(\boldsymbol{y}, \theta) = p(\boldsymbol{y}|\theta)p(\theta)$$

- $p(\boldsymbol{y}|\theta)$ is our posited observation/generation model for data, also called the likelihood function
  - Also helps in modeling the noise/uncertainty in the observed data
- $p(\theta)$ can encode our prior beliefs about the unknowns, called the prior distribution
- Can choose appropriate likelihood/prior to inject our domain knowledge about the problem/data

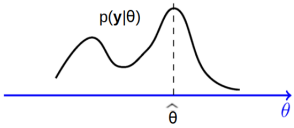
# The Inference Task

- Usually, two ways to perform inference for the unknowns $\theta$
  - Find a point estimate, e.g., the "most likely" $\theta$ that might have generated the observed data $\boldsymbol{y}$

$$\hat{\theta} = \arg\max_{\theta} p(\boldsymbol{y}|\theta) : \text{maximum likelihood (ML) estimation}$$

(i.e., find $\theta$ for which data looks most probable)

(ignores the prior $p(\theta)$)

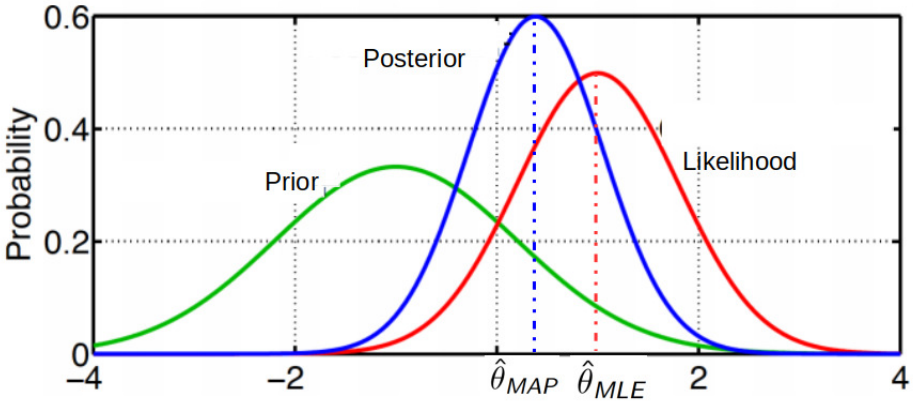



  - Perform fully Bayesian inference. Find the posterior distribution of $\theta$ using the Bayes rule

$$p(\theta|\boldsymbol{y}) = \frac{p(\theta)p(\boldsymbol{y}|\theta)}{p(\boldsymbol{y})} = \frac{p(\theta)p(\boldsymbol{y}|\theta)}{\int p(\boldsymbol{y},\theta)d\theta} = \frac{p(\theta)p(\boldsymbol{y}|\theta)}{\int p(\theta)p(\boldsymbol{y}|\theta)d\theta}$$ (takes into account the prior)

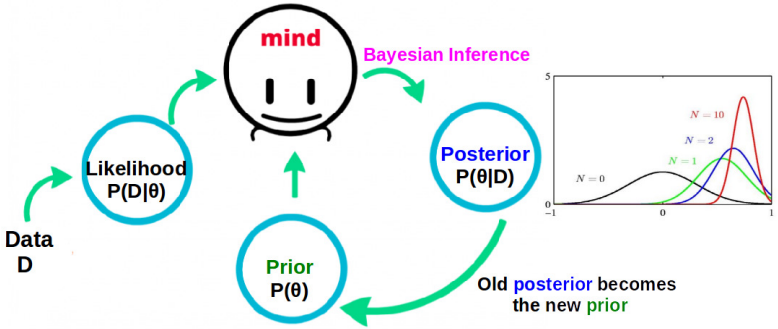
- The posterior distribution provides a more holistic view of $\theta$, not just a "point estimate"
  - Caveat: Also a much harder task in general as compared to finding the point estimate
- Another cheaper alternative is to directly find the maxima (mode) of the posterior distribution
  - Known as maximum-a-posteriori (MAP) estimation

# The Inference Task (A Pictorial Illustration)



# The Bayesian Inference Cycle

- Bayesian inference fits naturally into an "online" learning setting




- Our belief about $\theta$ keeps getting updated as we see more and more data

# Making Predictions

- Can use the inferred $\theta$ to predict other unknown quantities of interest
- E.g., for a new observation $y_*$, we can infer its predictive distribution $p(y_*|\boldsymbol{y})$
- This too can be done in two ways
  - Using the MLE/MAP point estimate ($\hat{\theta}$) of $\theta$, compute the "plug-in" predictive distribution

$$p(y_*|\boldsymbol{y}) = p(y_*|\hat{\theta})$$

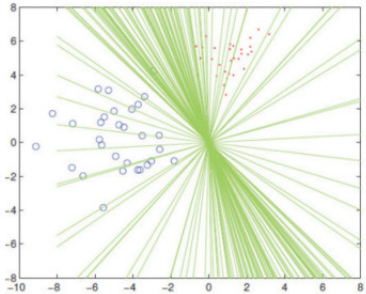
  - Compute the posterior predictive distribution by posterior-weighted averaging over all values of $\theta$

$$\begin{aligned} p(y_*|\boldsymbol{y}) &= \int p(y_*, \theta|\boldsymbol{y}) d\theta \\ &= \int p(y_*|\theta, \boldsymbol{y}) p(\theta|\boldsymbol{y}) d\theta \\ &= \int p(y_*|\theta) p(\theta|\boldsymbol{y}) d\theta \qquad \text{(Assumes that, given } \theta, \ y_* \text{ is independent of } \boldsymbol{y}) \end{aligned}$$

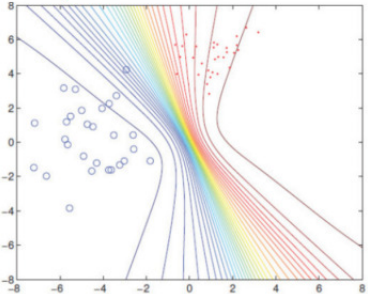
- Posterior averaged prediction is more robust (and also more informative; more on this later)
  - Caveat: Also much harder to compute as compared to the plug-in prediction

# Posterior Predictive (A Pictorial Illustration)

Samples from posterior distribution of a linear model for binary classification

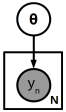



Equal probability contours of the resulting posterior predictive distribution

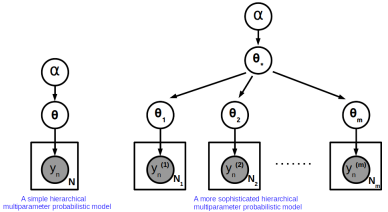


# Single Parameter Model vs Multiparameter Model

- Single parameter models have only a single unknown parameter
  - This parameter does not depend on any other unknowns




  - What it means: The prior $p(\theta)$ doesn't depend on any other unknown(s)
- Multiparameter models have multiple unknown parameters, e.g., hierarchical models




- Multiparameter models are more commonly encountered, more powerful/flexible/expressive. But also usually harder to do inference on (require methods such as EM, MCMC, VB inference, etc.)

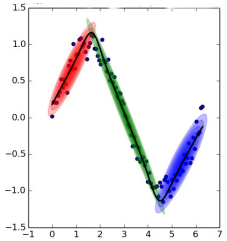
# Probabilistic Modeling = Density Estimation

- Any probabilistic modeling/inference problem ultimately boils down to doing density estimation
- Reason: All of the following quantities are essentially probability densities (continuous/discrete)
  - The likelihood distribution $p(\boldsymbol{y}|\theta)$
  - The prior distribution $p(\theta)$
  - The posterior distribution $p(\theta|\boldsymbol{y})$
  - The plug-in predictive distribution $p(y_*|\hat{\theta})$
  - The posterior predictive distribution $p(y_*|\boldsymbol{y})$
- Note: In prob. modeling, even supervised learning problems reduce to doing density estimation!
  - Here the observations are of the form $(\mathbf{X}, \boldsymbol{y}) = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$
  - The ultimate goal is to compute plug-in predictive $p(y_*|\boldsymbol{x}_*, \hat{\theta})$ or posterior predictive $p(y_*|\boldsymbol{x}_*, \mathbf{X}, \boldsymbol{y})$
  - Can compute it by modeling $p(\mathbf{X}, \boldsymbol{y}|\theta)$, or directly modeling $p(\boldsymbol{y}|\mathbf{X}, \theta)$. Both involve density est.

# Some Other Benefits of the Probabilistic Approach

# Modular Construction of Complex Models

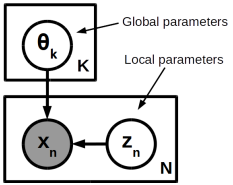
- Can easily construct combinations of multiple simple probabilistic models to learn complex patterns
- An example: Can perform nonlinear regression using a mixture of linear regression models
  - It is a simple yet powerful combination of two models - one that performs clustering of the data and the other that learns a linear regression model within each cluster (both learned jointly)




- More generally, these are called "mixture of experts" models

# Latent Variables and Generative Models

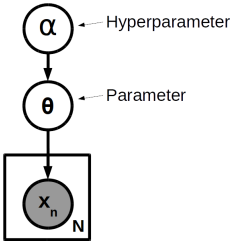
- Can endow the generative distribution $p(\boldsymbol{x}_n|\theta)$ with a data-point specific latent variable $\boldsymbol{z}_n$
- Each latent variable represents an "encoding" of the corresponding data point
- Such models are used in many problems, especially unsupervised learning: Gaussian mixture model, probabilistic principal component analysis, topic models, modeling sequential/time-series data, etc.




- Learning latent variable models has two benefits: (1) Each latent variable $\boldsymbol{z}_n$ can be used as a new representation of the data $\boldsymbol{x}_n$, and (2) We can synthesize new data using by generating a $\boldsymbol{z}$ randomly and mapping it to $\boldsymbol{x}$ via the generative model $p(\boldsymbol{x}|\boldsymbol{z}, \theta)$ from the learned model
- Note: LVMs are another example of multiparameter models.Usually harder to do inference on, as compared to models that have no latent variables (even point estimation such as MLE is hard!)

# Hyperparameter Estimation

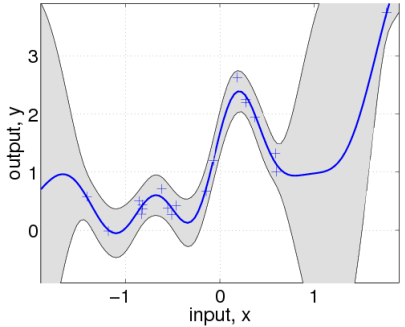
- Every model invariably has certain hyperparameters, e.g., regularization hyperparater in a linear regression model, or kernel hyperparameters in kernel regression or kernel SVM, etc.




- This is again an example of a multiparameter model.
- Estimating the hyperparameter becomes a subroutine of the overall model's inference problem
  - Both point estimation and fully Bayesian inference possible for hyperparameters
  - Usually a hard problem but many ways to solve it

# Sequential Learning and Decision-Making

- The estimate of model's uncertainty can "guide" us, e.g.
  - Given the current estimate of a function uncertainty over the input space, where should we acquire the next observation?



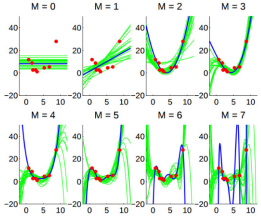

- This can many applications in active learning, reinforcement learning, Bayesian optimization, etc.

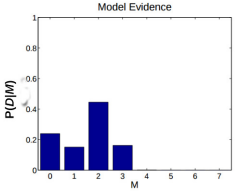
# Model Comparison

- Suppose we have a number of models $m = 1, 2, 3, \ldots, M$ to choose from
- We can compute the posterior probability of each candidate model, again using Bayes rule

$$\boxed{P(m|\boldsymbol{y}) = \frac{P(m)P(\boldsymbol{y}|m)}{P(\boldsymbol{y})}} \qquad [\text{where } p(\boldsymbol{y}|m) = \int p(\boldsymbol{y}, \theta|m)d\theta = \int p(\boldsymbol{y}|\theta, m)p(\theta|m)d\theta \ ]$$

- Assuming each model is equally likey to be chosen *a priori*, we can ignore the prior $P(m)$
  - Then model selection would just choose the model $m$ with largest marginal likelihood $P(\boldsymbol{y}|m)$

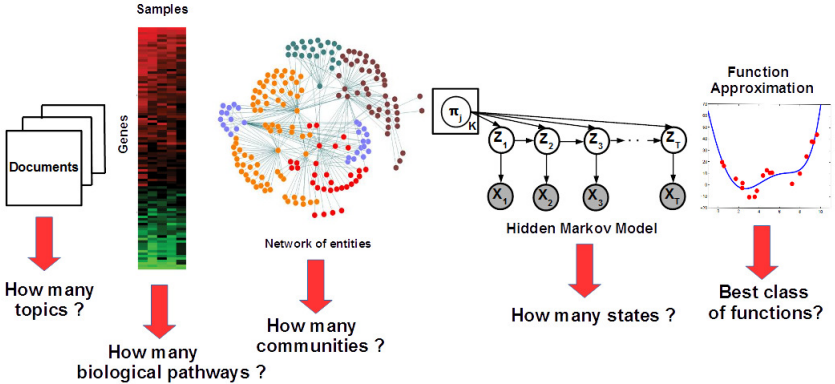



- It doesn't require a cross-validation set (can be done even for unsupervised learning problems)

# Nonparametric Bayesian Modeling

- **Nonparametric Bayesian Modeling:** A principled way to learn "right" model size/complexity




- The model size can grow with data (especially desirable for online learning settings)

# Tentative List of Topics

- Basics of probabilistic modeling and inference for single and multiparameter/hierarchical models
  - This covers estimating std. probability distributions from data, and regression/classification problems
- Basics of probabilistic graphical models
- Latent variable models for unsupervised learning and supervised learning
- Approximate inference methods (MCMC and Variational Inference, including recent advances)
- Nonparametric Bayesian models (Gaussian Process, Dirichlet Process, Beta Process)
- Latent variable models for text data and relational data (e.g., social networks)
- Latent variable models for sequential and time-series data
- Bayesian Optimization and sequential decision making problems
- Probabilistic Programming (implementing probabilistic models using Stan/Edward)
- Other topics based on students' interest

# Course Goals

- Learn the basics of probabilistic modeling of data
- Learn how to formulate complex data modeling problems via probabilistic models
- Learn the implications of making specific choices of likelihood/prior distributions and how these choices influence the model's properties and inference
- Understand the maths as well as the intuition behind the inference methods used for estimating the unknowns in such models
- Learn how to implement such models (both from scratch, as well as using support from some existing software frameworks such as Stan/Edward)
- Get acquainted with recent developments in the field of probabilistic modeling and inference
- Use the course project to reinforce the topics you learn about in the class, and explore other topics beyond what is covered in the class

# Pre-requisites and My Expectations from You

- Need to be comfortable with the following
  - Standard (discrete/continuous) probability distributions and their properties
  - Various other probability related concepts such as expectation, KL divergence, etc.
  - Linear algebra and calculus on vectors/matrices
  - Basic optimization techniques such as gradient and stochastic gradient methods
  - Note: I'll provide some pointers to quick refreshers on these (plus a tutorial session next week)
- Should try to work out (as an exercise) any derivations that I might skip in the class but request you to try yourself (if they are not immediately obvious to you)
  - In case of difficulty, get help from the instructor, TAs, or your classmates
- Regularly go through the provided reading materials
- Attend classes regularly (even though the attendance is not enforced)