Course Logistics, Intro to Probabilistic Modeling and Inference

Piyush Rai

Topics in Probabilistic Modeling and Inference (CS698X)

Jan 7, 2019

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - “TPMI” or just “PMI”
- **Timing and Venue:** M/W 17:10-18:25, KD-101

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - “TPMI” or just “PMI”
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - "TPMI" or just "PMI"
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - "TPMI" or just "PMI"
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`
- **Gradescope (for assignment submissions):** `https://tinyurl.com/cs698x-s18g`

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - “TPMI” or just “PMI”
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`
- **Gradescope (for assignment submissions):** `https://tinyurl.com/cs698x-s18g`
  - Assignments must be typeset in LaTeX

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - "TPMI" or just "PMI"
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`
- **Gradescope (for assignment submissions):** `https://tinyurl.com/cs698x-s18g`
  - Assignments must be typeset in LaTeX
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - "TPMI" or just "PMI"
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`
- **Gradescope (for assignment submissions):** `https://tinyurl.com/cs698x-s18g`
  - Assignments must be typeset in LaTeX
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in; office: RM-502)

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - "TPMI" or just "PMI"
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`
- **Gradescope (for assignment submissions):** `https://tinyurl.com/cs698x-s18g`
    - Assignments must be typeset in LaTeX
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in; office: RM-502)
    - Prefix email subject by CS698X (better alternative: Piazza private message to instructor)

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - "TPMI" or just "PMI"
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`
- **Gradescope (for assignment submissions):** `https://tinyurl.com/cs698x-s18g`
  - Assignments must be typeset in LaTeX
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in; office: RM-502)
  - Prefix email subject by CS698X (better alternative: Piazza private message to instructor)
  - Office Hours: Friday 18:00-19:00 (by appointment)

Course Logistics

- **Course name:** Topics in Probabilistic Modeling and Inference (CS698X) - “TPMI” or just “PMI”
- **Timing and Venue:** M/W 17:10-18:25, KD-101
- **Course website:** `https://tinyurl.com/cs698x-s19w` (slides/readings etc will be posted here)
- **Piazza discussion site:** `https://tinyurl.com/cs698x-s19p`
- **Gradescope (for assignment submissions):** `https://tinyurl.com/cs698x-s18g`
  - Assignments must be typeset in LaTeX
- Course-related announcements will be sent on the **class mailing list** (and also on Piazza)
- **Instructor:** Piyush Rai (Email: piyush@cse.iitk.ac.in; office: RM-502)
  - Prefix email subject by CS698X (better alternative: Piazza private message to instructor)
  - Office Hours: Friday 18:00-19:00 (by appointment)
- Auditing? Don't need formal permission from me. Send me email to be added to the mailing list

The TA Team

- TA office hours/locations and contact details will be posted on Piazza


Shivam Bansal


Dhanajit Brahma


Sunabha Chatterjee


Abhishek Kumar


Siddhartha Saxena


Vinay Kumar Verma

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%
  - Final exam: 25%

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%
  - Final exam: 25%
  - Note: Both exams will be closed-book (you will be provided a cheat-sheet)

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%
  - Final exam: 25%
  - Note: Both exams will be closed-book (you will be provided a cheat-sheet)
- Class Project: 20%

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%
  - Final exam: 25%
  - Note: Both exams will be closed-book (you will be provided a cheat-sheet)
- Class Project: 20%
  - Research project, to be done in groups of 3

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%
  - Final exam: 25%
  - Note: Both exams will be closed-book (you will be <u>provided</u> a cheat-sheet)
- Class Project: 20%
  - Research project, to be done in groups of 3
  - More details will be shared very soon

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%
  - Final exam: 25%
  - Note: Both exams will be closed-book (you will be provided a cheat-sheet)
- Class Project: 20%
  - Research project, to be done in groups of 3
  - More details will be shared very soon
- Top 10% students, based only on exams+quiz $\Rightarrow$ straight A grade

Grading Scheme

- 4-5 homework assignments: 30%
  - Written questions + some programming in Python/MATLAB
- 2 quizzes: 10%
- 2 exams: 40%
  - Midterm exam: 15%
  - Final exam: 25%
  - Note: Both exams will be closed-book (you will be provided a cheat-sheet)
- Class Project: 20%
  - Research project, to be done in groups of 3
  - More details will be shared very soon
- Top 10% students, based only on exams+quiz $\Rightarrow$ straight A grade
- Outstanding, publishable work in class project $\Rightarrow$ straight A grade

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- Must write your own solution in your own words (same goes for coding assignments)

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- Must 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

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- Must 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
- Other things that will lead to punishment
    - Use of unfair means in the exams
    - Fabricating experimental results in assignments/project

Collaboration vs Cheating

- Collaboration is encouraged. Cheating/copying will lead to strict punishments.
- Feel free to discuss homework assignments with your classmates.
- Must 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
- 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

Course Policies

- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules

Course Policies

- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules
- Requests for homework extensions won't be entertained
  - Can submit homeworks upto 3 late days with 10% penalty per day
  - Every student entitled for ONE late homework submission without penalty (use it wisely)

Course Policies

- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules
- Requests for homework extensions won't be entertained
  - Can submit homeworks upto 3 late days with 10% penalty per day
  - Every student entitled for ONE late homework submission without penalty (use it wisely)
- No attendance policy enforced but I expect you to attend classes regularly

Course Policies

- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules
- Requests for homework extensions won't be entertained
  - Can submit homeworks upto 3 late days with 10% penalty per day
  - Every student entitled for ONE late homework submission without penalty (use it wisely)
- No attendance policy enforced but I expect you to attend classes regularly
- Use Piazza actively and responsibly

Course Policies

- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules
- Requests for homework extensions won't be entertained
  - Can submit homeworks upto 3 late days with 10% penalty per day
  - Every student entitled for ONE late homework submission without penalty (use it wisely)
- No attendance policy enforced but I expect you to attend classes regularly
- Use Piazza actively and responsibly
  - Limited to discussions related to class

Course Policies

- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules
- Requests for homework extensions won't be entertained
  - Can submit homeworks upto 3 late days with 10% penalty per day
  - Every student entitled for ONE late homework submission without penalty (use it wisely)
- No attendance policy enforced but I expect you to attend classes regularly
- Use Piazza actively and responsibly
  - Limited to discussions related to class
  - Allowed to remain anonymous to classmates but not to instructors

Course Policies

- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules
- Requests for homework extensions won't be entertained
  - Can submit homeworks upto 3 late days with 10% penalty per day
  - Every student entitled for ONE late homework submission without penalty (use it wisely)
- No attendance policy enforced but I expect you to attend classes regularly
- Use Piazza actively and responsibly
  - Limited to discussions related to class
  - Allowed to remain anonymous to classmates but not to instructors
  - Avoid asking questions privately (so that everyone can benefit from the question/answer)

Course Policies

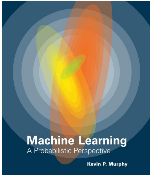
- Repeat: Absolutely ZERO tolerance for cheating
  - Punishable as per institute's/department's rules
- Requests for homework extensions won't be entertained
  - Can submit homeworks upto 3 late days with 10% penalty per day
  - Every student entitled for ONE late homework submission without penalty (use it wisely)
- No attendance policy enforced but I expect you to attend classes regularly
- Use Piazza actively and responsibly
  - Limited to discussions related to class
  - Allowed to remain anonymous to classmates but not to instructors
  - Avoid asking questions privately (so that everyone can benefit from the question/answer)
  - Questions should not be attempts to get/verify answers to homework problems

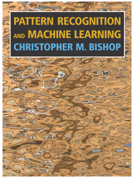
Textbook and Readings

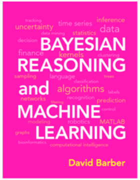
- **Textbook:** No official textbook required
- Required reading material will be provided

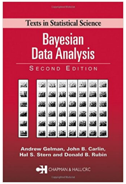
Textbook and Readings

- **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









Background Expected (Important)

- Basic concepts from probability theory (also refer to the prob-stats refresher on course webpage)
  - Random variables, various discrete/continuous distributions
  - PDF, CDF, expectation, variance, mutual information, entropy, Kullback-Leibler (KL) divergence
  - Basic methods for parameter estimation for probability distributions (e.g., maximum likelihood)

Background Expected (Important)

- Basic concepts from probability theory (also refer to the prob-stats refresher on course webpage)
  - Random variables, various discrete/continuous distributions
  - PDF, CDF, expectation, variance, mutual information, entropy, Kullback-Leibler (KL) divergence
  - Basic methods for parameter estimation for probability distributions (e.g., maximum likelihood)
- Familiarity with basic probabilistic models in machine learning, e.g.,
  - Probabilistic view of linear regression, logistic regression, generative classification
  - Latent variable models (e.g., Gaussian mixture model, probabilistic PCA)

Background Expected (Important)

- Basic concepts from probability theory (also refer to the prob-stats refresher on course webpage)
  - Random variables, various discrete/continuous distributions
  - PDF, CDF, expectation, variance, mutual information, entropy, Kullback-Leibler (KL) divergence
  - Basic methods for parameter estimation for probability distributions (e.g., maximum likelihood)
- Familiarity with basic probabilistic models in machine learning, e.g.,
  - Probabilistic view of linear regression, logistic regression, generative classification
  - Latent variable models (e.g., Gaussian mixture model, probabilistic PCA)
- Familiarity with standard machine learning models, e.g.,
  - Nearest neighbors, kernel methods, logistic regression, SVM
  - Standard algos for clustering, dimensionality reduction, matrix factorization

Background Expected (Important)

- Basic concepts from probability theory (also refer to the prob-stats refresher on course webpage)
  - Random variables, various discrete/continuous distributions
  - PDF, CDF, expectation, variance, mutual information, entropy, Kullback-Leibler (KL) divergence
  - Basic methods for parameter estimation for probability distributions (e.g., maximum likelihood)
- Familiarity with basic probabilistic models in machine learning, e.g.,
  - Probabilistic view of linear regression, logistic regression, generative classification
  - Latent variable models (e.g., Gaussian mixture model, probabilistic PCA)
- Familiarity with standard machine learning models, e.g.,
  - Nearest neighbors, kernel methods, logistic regression, SVM
  - Standard algos for clustering, dimensionality reduction, matrix factorization
- Familiarity with basic optimization methods, e.g.,
  - Gradient descent, stochastic gradient descent, alternating optimization
  - Basic optimization algos for latent variable models (e.g., expectation maximization)

Probabilistic Modeling and Inference

(or living happily with uncertainty)

Why a Probabilistic Approach?

- We may want probabilistic predictions (e.g., probability that a transaction is fraud)

Why a Probabilistic Approach?

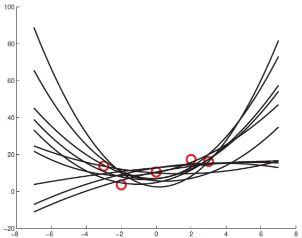
- We may want probabilistic predictions (e.g., probability that a transaction is fraud)
- We may have imprecise/noisy data. Need to model the noise/uncertainty explicitly

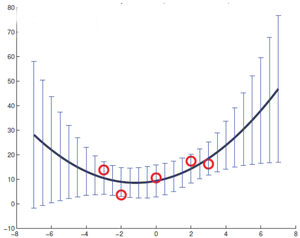
Why a Probabilistic Approach?

- We may want probabilistic predictions (e.g., probability that a transaction is fraud)
- We may have imprecise/noisy data. Need to model the noise/uncertainty explicitly
  - Can do it using appropriate probability distributions

Why a Probabilistic Approach?

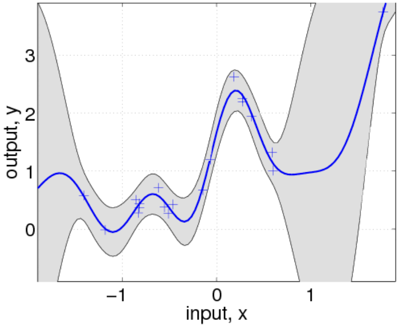
- We may want probabilistic predictions (e.g., probability that a transaction is fraud)
- We may have imprecise/noisy data. Need to model the noise/uncertainty explicitly
  - Can do it using appropriate probability distributions
- Due to data scarcity, there may be uncertainty in the estimated model parameters and predictions
  - Can do so by learning a probability distribution over parameters and predictions





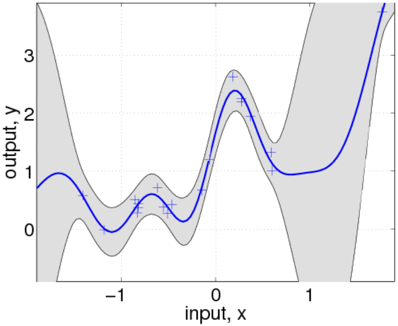
Why a Probabilistic Approach (Contd)?

- Sequential decision-making: 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?



Why a Probabilistic Approach (Contd)?

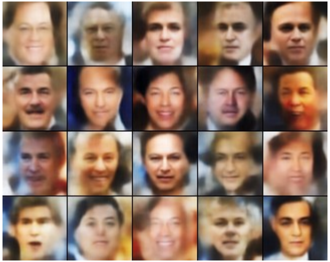
- Sequential decision-making: Estimate of model's uncertainty can "guide" us, e.g.
  - Given the current estimate of a function uncertainty over the input space, where should we acquire the next observation?



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

Why a Probabilistic Approach (Contd)?

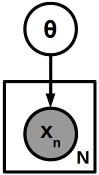
- Sometimes we may be interested in learning the underlying probability distribution of data
- Learning the distribution can enable us to understand and also generate new data!



Modeling Data Probabilistically: A Simplistic View

- Assume data $\mathbf{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ generated from a probabilistic model with unknown parameters θ

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \sim p(\boldsymbol{x}|\theta)$$

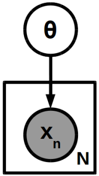


- The above picture denotes a simplistic "plate notation" graphical model

Modeling Data Probabilistically: A Simplistic View

- Assume data $\mathbf{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ generated from a probabilistic model with unknown parameters θ

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \sim p(\boldsymbol{x}|\theta)$$

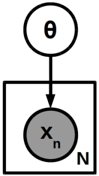


- The above picture denotes a simplistic "plate notation" graphical model
- Note: Shaded nodes = observed; unshaded nodes = unknown/unobserved

Modeling Data Probabilistically: A Simplistic View

- Assume data $\mathbf{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ generated from a probabilistic model with unknown parameters θ

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \sim p(\boldsymbol{x}|\theta)$$

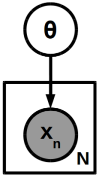


- The above picture denotes a simplistic "plate notation" graphical model
- Note: Shaded nodes = observed; unshaded nodes = unknown/unobserved
- Goal: To estimate the unknowns of the model (θ in this case), given the observed data $\mathbf{X}$

Modeling Data Probabilistically: A Simplistic View

- Assume data $\mathbf{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ generated from a probabilistic model with unknown parameters θ

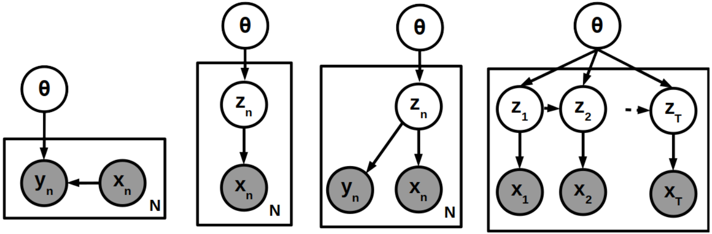
$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \sim p(\boldsymbol{x}|\theta)$$



- The above picture denotes a simplistic "plate notation" graphical model
- Note: Shaded nodes = observed; unshaded nodes = unknown/unobserved
- Goal: To estimate the unknowns of the model (θ in this case), given the observed data $\mathbf{X}$
- Can use the learned model to make predictions
  - E.g., the probability $p(\boldsymbol{x}_*|\theta)$ or $p(\boldsymbol{x}_*|\mathbf{X})$ of a new input $\boldsymbol{x}_*$ under this model

Modeling Data Probabilistically

- This basic problem set-up can be generalized in various ways



A Simple Supervised Learning Model

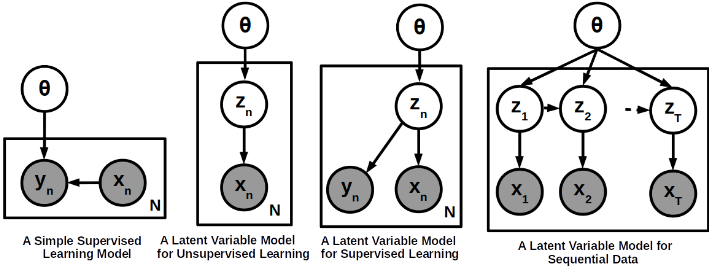
A Latent Variable Model for Unsupervised Learning

A Latent Variable Model for Supervised Learning

A Latent Variable Model for Sequential Data

Modeling Data Probabilistically

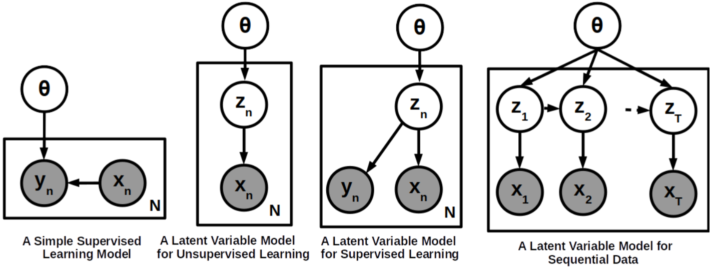
- This basic problem set-up can be generalized in various ways



- Any node (even if observed) that we are uncertain about is modeled by a probability distribution
  - These nodes become the random variables of the model

Modeling Data Probabilistically

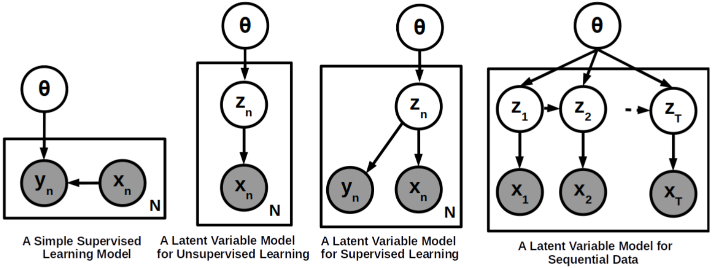
- This basic problem set-up can be generalized in various ways



- Any node (even if observed) that we are uncertain about is modeled by a probability distribution
    - These nodes become the random variables of the model
- The full model is specified via a joint prob. distribution over all random variables

Modeling Data Probabilistically

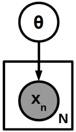
- This basic problem set-up can be generalized in various ways



- Any node (even if observed) that we are uncertain about is modeled by a probability distribution
  - These nodes become the random variables of the model
- The full model is specified via a joint prob. distribution over all random variables
- The goal is to infer the unknowns of the model, given the observed data

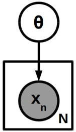
Modeling Data Probabilistically

- Specification of probabilistic models requires two key ingredients: Likelihood and prior



Modeling Data Probabilistically

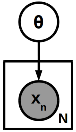
- Specification of probabilistic models requires two key ingredients: Likelihood and prior



- Likelihood function $p(\boldsymbol{x}|\theta)$ or the "observation model" specifies how data is generated
  - Measures data fit (or "loss") w.r.t. the given parameter θ

Modeling Data Probabilistically

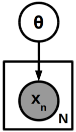
- Specification of probabilistic models requires two key ingredients: Likelihood and prior



- Likelihood function $p(\boldsymbol{x}|\theta)$ or the "observation model" specifies how data is generated
  - Measures data fit (or "loss") w.r.t. the given parameter θ
- Prior distribution $p(\theta)$ specifies how likely different parameter values are *a priori*
  - Also corresponds to imposing a "regularizer" over θ

Modeling Data Probabilistically

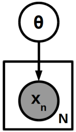
- Specification of probabilistic models requires two key ingredients: Likelihood and prior



- Likelihood function $p(\boldsymbol{x}|\theta)$ or the "observation model" specifies how data is generated
  - Measures data fit (or "loss") w.r.t. the given parameter θ
- Prior distribution $p(\theta)$ specifies how likely different parameter values are *a priori*
  - Also corresponds to imposing a "regularizer" over θ
- Domain knowledge can help in the specification of the likelihood and the prior

Parameter Estimation/Inference in Probabilistic Models

- Perhaps the simplest way is to find θ that makes the observed data most likely or most probable

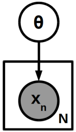


- Formally, find θ that maximizes the probability of the observed data

$$\hat{\theta} \quad = \quad \arg\max_{\theta} \; \log p(\mathbf{X}|\theta)$$

Parameter Estimation/Inference in Probabilistic Models

- Perhaps the simplest way is to find θ that makes the observed data most likely or most probable



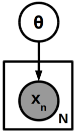
- Formally, find θ that maximizes the probability of the observed data

$$\hat{\theta} = \arg\max_{\theta} \log p(\mathbf{X}|\theta)$$

- However, this gives a single "point" estimate of θ. Doesn't tell us about the uncertainty in θ

Parameter Estimation/Inference in Probabilistic Models

- Perhaps the simplest way is to find θ that makes the observed data most likely or most probable



- Formally, find θ that maximizes the probability of the observed data

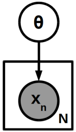
$$\hat{\theta} = \arg\max_{\theta} \log p(\mathbf{X}|\theta)$$

- However, this gives a single "point" estimate of θ. Doesn't tell us about the uncertainty in θ
- We can estimate the full posterior distribution over θ to get the uncertainty

$$p(\theta|\mathbf{X}) = \frac{p(\mathbf{X}|\theta)p(\theta)}{p(\mathbf{X})} \propto \text{Likelihood} \times \text{Prior}$$

Parameter Estimation/Inference in Probabilistic Models

- Perhaps the simplest way is to find θ that makes the observed data most likely or most probable



- Formally, find θ that maximizes the probability of the observed data

$$\hat{\theta} = \arg\max_{\theta} \log p(\mathbf{X}|\theta)$$

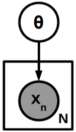
- However, this gives a single "point" estimate of θ. Doesn't tell us about the uncertainty in θ
- We can estimate the full posterior distribution over θ to get the uncertainty

$$p(\theta|\mathbf{X}) = \frac{p(\mathbf{X}|\theta)p(\theta)}{p(\mathbf{X})} \propto \text{Likelihood} \times \text{Prior}$$

- This is called **Bayesian inference**. The posterior distribution captures the uncertainty in θ

Parameter Estimation/Inference in Probabilistic Models

- Perhaps the simplest way is to find θ that makes the observed data most likely or most probable



- Formally, find θ that maximizes the probability of the observed data

$$\hat{\theta} = \arg\max_{\theta} \log p(\mathbf{X}|\theta)$$

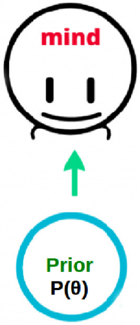
- However, this gives a single "point" estimate of θ. Doesn't tell us about the uncertainty in θ
- We can estimate the full posterior distribution over θ to get the uncertainty

$$p(\theta|\mathbf{X}) = \frac{p(\mathbf{X}|\theta)p(\theta)}{p(\mathbf{X})} \propto \text{Likelihood} \times \text{Prior}$$

- This is called **Bayesian inference**. The posterior distribution captures the uncertainty in θ
- We will study both point estimation and Bayesian inference methods (and hybrids!)

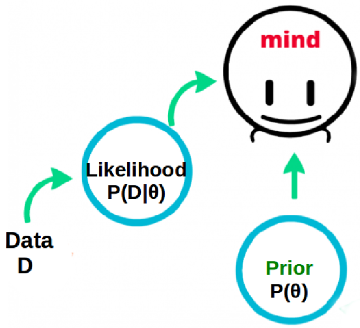
Bayesian Inference

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



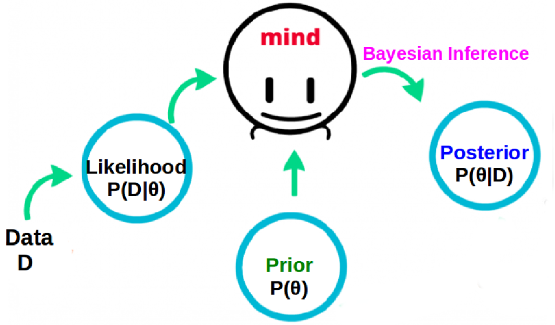
Bayesian Inference

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



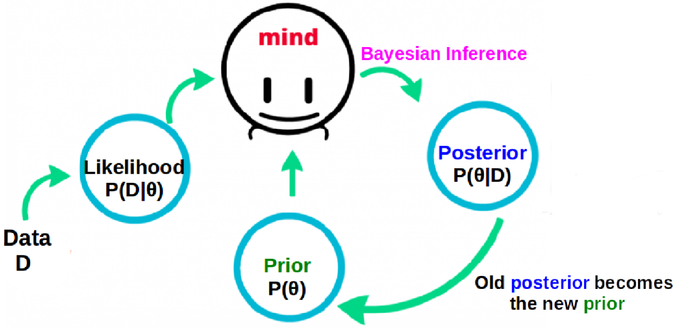
Bayesian Inference

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



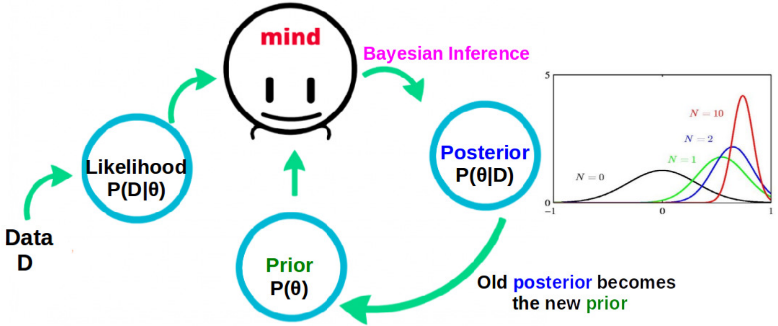
Bayesian Inference

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



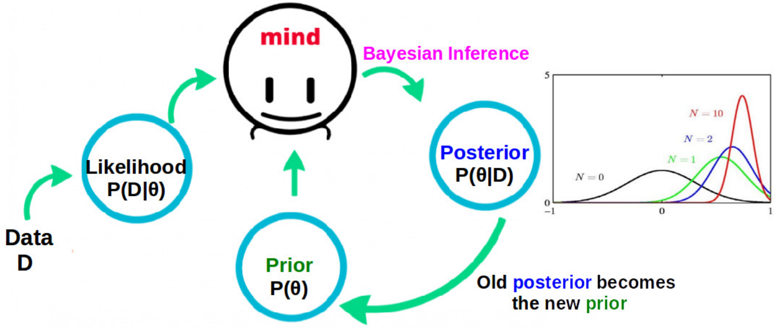
Bayesian Inference

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



Bayesian Inference

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

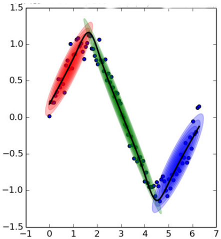


- Our belief about θ keeps getting updated as we see more and more data

Some Other Benefits of the Probabilistic Approach

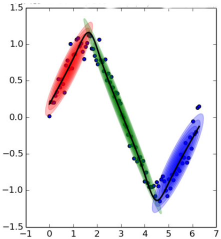
Modular Construction of Complex Models

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



Modular Construction of Complex Models

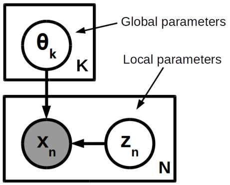
- Can easily construct combinations of multiple simple probabilistic models to learn complex patterns
- An example: Can perform nonlinear classification using a mixture of linear classifiers
  - It is a simple yet powerful combination of two models - one that performs clustering of the data and the other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



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

Generative Models

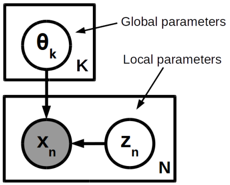
- Generative models of data can be naturally specified in a probabilistic framework



- Each data point $\boldsymbol{x}_n$ is associated with latent variables $\boldsymbol{z}_n$
- Latent variables can be used a compact representation or an "encoding" of the data
- Such models are used in many problems, especially unsupervised learning: Gaussian mixture model, probabilistic principal component analysis, topic models, deep generative models, etc.

Generative Models

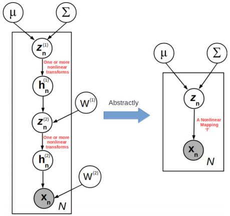
- Generative models of data can be naturally specified in a probabilistic framework



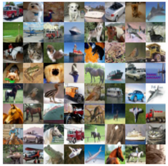
- Each data point $\boldsymbol{x}_n$ is associated with latent variables $\boldsymbol{z}_n$
- Latent variables can be used a compact representation or an "encoding" of the data
- Such models are used in many problems, especially unsupervised learning: Gaussian mixture model, probabilistic principal component analysis, topic models, deep generative models, etc.
- Can also use the latent variables to infer missing data or relevance of each data point

(Deep) Generative Models

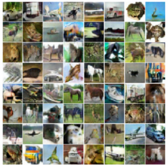
- Deep Generative Models for extremely popular nowadays (e.g., Variational Auto-encoders and Generative Adversarial Networks)



- Once learned, these models can also synthesize realistic looking "new" data from random $\boldsymbol{z}$'s



Real images (CIFAR-10)



Generated images

Averaging Over Posterior Distribution

- Can use the posterior distribution over parameters to compute "averaged prediction", e.g.,

$$p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \mathbf{X}, \boldsymbol{y}) = \int p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \theta)p(\theta|\mathbf{X}, \boldsymbol{y})d\theta$$

- $p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \mathbf{X}, \boldsymbol{y})$ with θ "integrated out" is called posterior predictive distribution

Averaging Over Posterior Distribution

- Can use the posterior distribution over parameters to compute "averaged prediction", e.g.,

$$p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \mathbf{X}, \boldsymbol{y}) = \int p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \theta)p(\theta|\mathbf{X}, \boldsymbol{y})d\theta$$

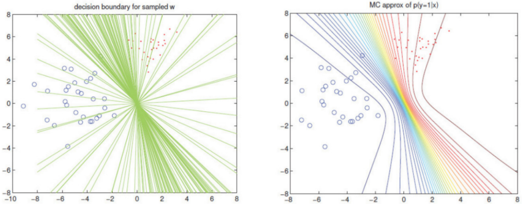
- $p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \mathbf{X}, \boldsymbol{y})$ with θ "integrated out" is called posterior predictive distribution
- Without a posterior, we can only compute $p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \theta)$ using a "single best" estimate of θ

Averaging Over Posterior Distribution

- Can use the posterior distribution over parameters to compute "averaged prediction", e.g.,

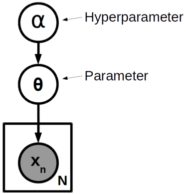
$$p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \mathbf{X}, \boldsymbol{y}) = \int p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \theta)p(\theta|\mathbf{X}, \boldsymbol{y})d\theta$$

- $p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \mathbf{X}, \boldsymbol{y})$ with θ "integrated out" is called posterior predictive distribution
- Without a posterior, we can only compute $p(\boldsymbol{y}_* = 1|\boldsymbol{x}_*, \theta)$ using a "single best" estimate of θ
- Averaging leads to more robust predictions (and prevents overfitting)



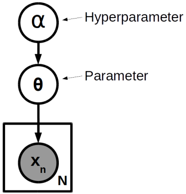
Hyperparameter Estimation

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



Hyperparameter Estimation

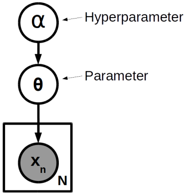
- Every model invariably has certain hyperparameters, e.g., regularization hyperparater in a linear regression model, or kernel hyperparameters in nonlinear regression of kernel SVM, etc.



- The probabilistic approach enables learning the hyperparam. from data (without cross-validation)

Hyperparameter Estimation

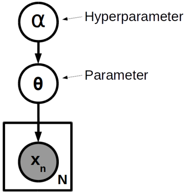
- Every model invariably has certain hyperparameters, e.g., regularization hyperparater in a linear regression model, or kernel hyperparameters in nonlinear regression of kernel SVM, etc.



- The probabilistic approach enables learning the hyperparam. from data (without cross-validation)
  - Can put priors on the hyperparameters and infer the posterior distribution

Hyperparameter Estimation

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

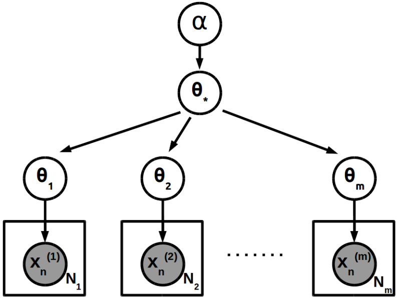


- The probabilistic approach enables learning the hyperparam. from data (without cross-validation)
    - Can put priors on the hyperparameters and infer the posterior distribution
    - Can do point estimation for hyperparameters by maximizing the marginal likelihood

$$\hat{\alpha} = \arg\max_{\alpha} \log P(\mathbf{X}|\alpha)$$

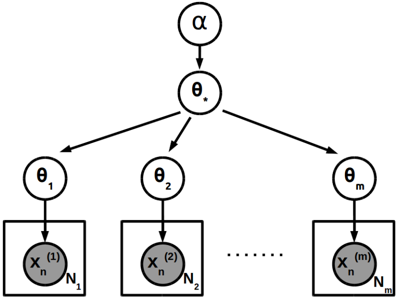
Multitask and Transfer Learning

- Allows joint learning across multiple data sets (known as multitask learning or transfer learning)



Multitask and Transfer Learning

- Allows joint learning across multiple data sets (known as multitask learning or transfer learning)



- Enables different but related models to "share statistical strength"

Model Comparison

- Suppose we have a number of models to choose from
- Let's compute the posterior probability of each candidate model, again using Bayes rule

$$P(m|\mathbf{X}) = \frac{P(m)P(\mathbf{X}|m)}{P(\mathbf{X})}$$

Model Comparison

- Suppose we have a number of models to choose from
- Let's compute the posterior probability of each candidate model, again using Bayes rule

$$P(m|\mathbf{X}) = \frac{P(m)P(\mathbf{X}|m)}{P(\mathbf{X})}$$

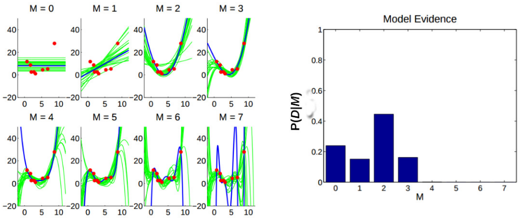
- Assuming each model is equally likey to be chosen *a priori*, we can ignore the prior $P(m)$

Model Comparison

- Suppose we have a number of models to choose from
- Let's compute the posterior probability of each candidate model, again using Bayes rule

$$P(m|\mathbf{X}) = \frac{P(m)P(\mathbf{X}|m)}{P(\mathbf{X})}$$

- Assuming each model is equally likey to be chosen *a priori*, we can ignore the prior $P(m)$
  - Just choose the model m that has the highest marginal likelihood $P(\mathbf{X}|m)$

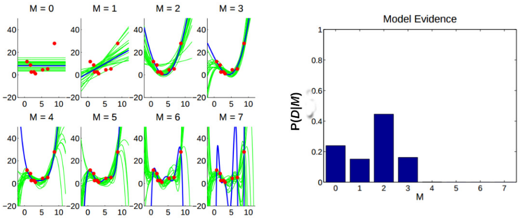


Model Comparison

- Suppose we have a number of models to choose from
- Let's compute the posterior probability of each candidate model, again using Bayes rule

$$P(m|\mathbf{X}) = \frac{P(m)P(\mathbf{X}|m)}{P(\mathbf{X})}$$

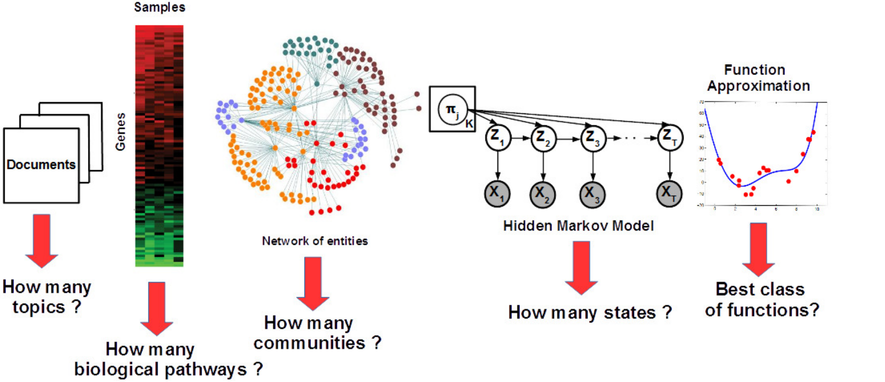
- Assuming each model is equally likey to be chosen *a priori*, we can ignore the prior $P(m)$
  - Just choose the model m that has the highest marginal likelihood $P(\mathbf{X}|m)$



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

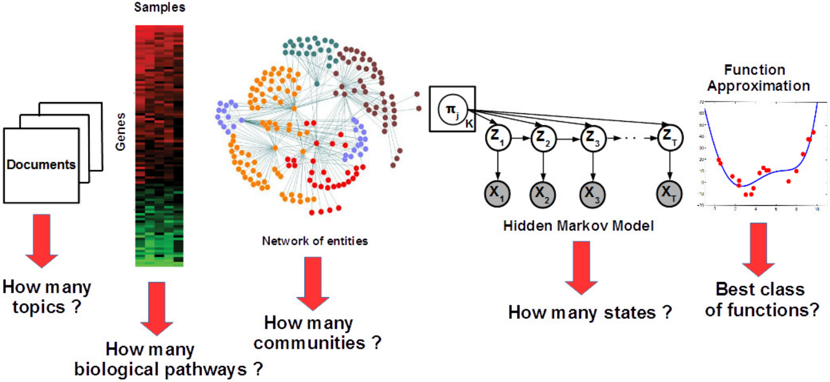
Nonparametric Bayesian Modeling

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



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 Outline

- Basics of probabilistic modeling and inference
  - Common probability distributions
  - Basic point estimation (MLE and MAP)
- Bayesian inference (simple and not-so-simple cases)
- Probabilistic models for regression and classification
- Probabilistic Graphical Models
- Gaussian Processes (probabilistic modeling meets kernels)
- Latent Variable Models (for i.i.d., sequential, and relational data)
- Approximate Bayesian inference (EM, variational inference, sampling, etc)
- Nonparametric Bayesian methods
- Recent Advances, e.g., deep generative models, black-box inference, etc