Statistical Models & Computing Methods

Lecture 1: Introduction

Cheng Zhang

School of Mathematical Sciences, Peking University

September 16, 2021

General Information

- Class times:
  - Even Monday 1:00-2:50pm, Thursday 10:10am-12:00pm
  - Classroom Building No.3, Room 406
- Instructor:
  - Cheng Zhang: `chengzhang@math.pku.edu.cn`
- Teaching assistants:
  - Tianyu Xie: `tianyuxie@pku.edu.cn`
- Tentative office hours:
  - 315 Building No.20
  - Thursday 3:00-5:00pm or by appointment
- Website:
  `https://zcrabbit.github.io/courses/smcm-f21.html`

- A branch of mathematical sciences focusing on efficient numerical methods for statistically formulated problems
- The focus lies on computer intensive statistical methods and efficient modern statistical models.
- Developing rapidly, leading to a broader concept of computing that combines the theories and techniques from many fields within the context of statistics, mathematics and computer sciences.

Goals

- Become familiar with a variety of modern computational statistical techniques and knows more about the role of computation as a tool of discovery
- Develop a deeper understanding of the mathematical theory of computational statistical approaches and statistical modeling.
- Understand what makes a good model for data.
- Be able to analyze datasets using a modern programming language (e.g., python).

Textbook

- No specific textbook required for this course
- Recommended textbooks:
  - Givens, G. H. and Hoeting, J. A. (2005) Computational Statistics, 2nd Edition, Wiley-Interscience.
  - Gelman, A., Carlin, J., Stern, H., and Rubin, D. (2003). Bayesian Data Analysis, 2nd Edition, Chapman & Hall.
  - Liu, J. (2001). Monte Carlo Strategies in Scientific Computing, Springer-Verlag.
  - Lange, K. (2002). Numerical Analysis for Statisticians, Springer-Verlag, 2nd Edition.
  - Hastie, T., Tibshirani, R. and Friedman, J. (2009). The Elements of Statistical Learning, 2nd Edition, Springer.
  - Goodfellow, I., Bengio, Y. and Courville, A. (2016). Deep Learning, MIT Press.

Tentative Topics

- Optimization Methods
  - Gradient Methods
  - Expectation Maximization
- Approximate Bayesian Inference Methods
  - Markov chain Monte Carlo
  - Variational Inference
  - Scalable Approaches
- Applications in Machine Learning & Related Fields
  - Variational Autoencoder
  - Generative Adversarial Networks
  - Flow-based Generative Models
  - Bayesian Phylogenetic Inference

Familiar with at least one programming language (with python preferred!).

- All class assignments will be in python (and use numpy).
- You can find a good Python tutorial at

  `http://www.scipy-lectures.org/`

  You may find a shorter python+numpy tutorial useful at

  `http://cs231n.github.io/python-numpy-tutorial/`

Familiar with the following subjects

- Probability and Statistical Inference
- Stochastic Processes

Grading Policy

- 4 Problem Sets: $4 \times 15\% = 60\%$
- Final Course Project: 40%
  - up to 3 people for each team
  - Teams should be formed by the end of week 4
  - Midterm proposal: 5%
  - Oral presentation: 10%
  - Final write-up: 25%
- Late policy
  - 7 free late days, use them in your ways
  - Afterward, 25% off per late day
  - No more than 3 late days per PS
  - Does not apply to Final Course Project
- Collaboration policy
  - Finish your work independently, verbal discussion allowed

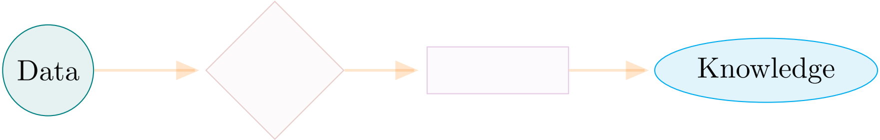
- Structure your project exploration around a general problem type, algorithm, or data set, but should explore around your problem, testing thoroughly or comparing to alternatives.
- Present a project proposal that briefly describe your teams' project concept and goals in one slide in class on 11/15.
- There will be in class project presentation at the end of the term. Not presenting your projects will be taken as voluntarily giving up the opportunity for the final write-ups.
- Turn in a write-up ($<$ 10 pages) describing your project and its outcomes, similar to a research-level publication.

Today's Agenda

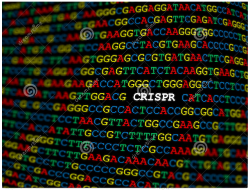
- A brief overview of statistical approaches

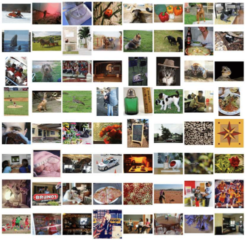
- Basic concepts in statistical computing

Statistical Pipeline

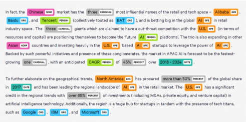


Statistical Pipeline



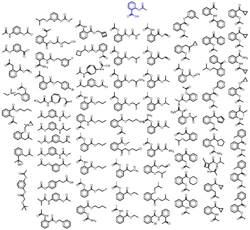


Data

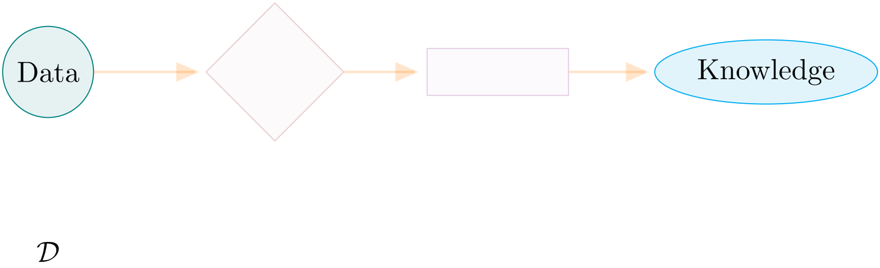


In fact, the Chinese NORP market has the three CARDINAL most influential names of the retail and tech space – Alibaba GPE , Baidu ORG , and Tencent PERSON (collectively touted as BAT ORG), and is betting big in the global AI GPE in retail industry space . The three CARDINAL giants which are claimed to have a cut-throat competition with the U.S. GPE (in terms of resources and capital) are positioning themselves to become the 'future AI PERSON platforms'. The trio is also expanding in other Asian NORP countries and investing heavily in the U.S. GPE based AI GPE startups to leverage the power of AI GPE . Backed by such powerful initiatives and presence of these conglomerates, the market in APAC AI is forecast to be the fastest-growing one CARDINAL , with an anticipated CAGR PERSON of 45% PERCENT over 2018 - 2024 DATE .

To further elaborate on the geographical trends, North America LOC has procured more than 50% PERCENT of the global share in 2017 DATE and has been leading the regional landscape of AI GPE in the retail market. The U.S. GPE has a significant credit in the regional trends with over 65% PERCENT of investments (including M&As, private equity, and venture capital) in artificial intelligence technology. Additionally, the region is a huge hub for startups in tandem with the presence of tech titans, such as Google ORG , IBM ORG , and Microsoft ORG .



Statistical Pipeline



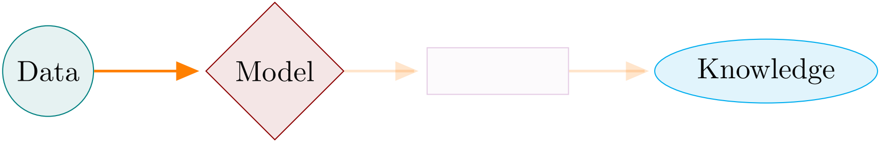
Statistical Pipeline

Linear Models

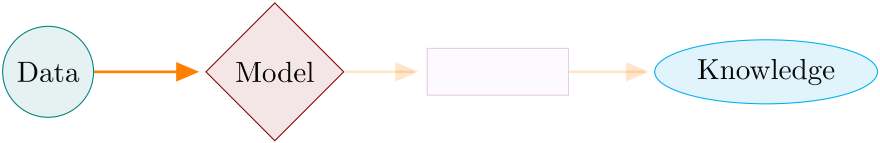
Neural Networks

Bayesian Nonparametric Models

Generalized Linear Models

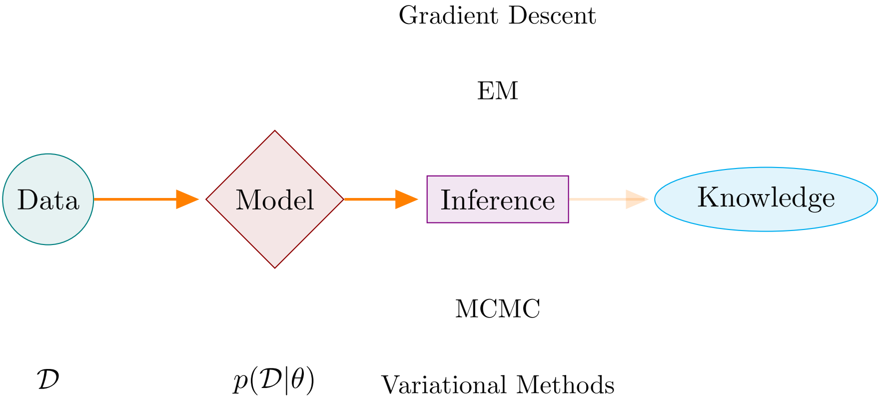


$\mathcal{D}$



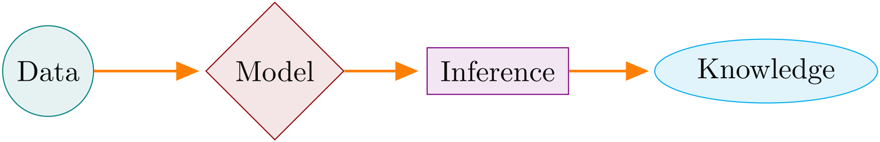
$\mathcal{D}$ $p(\mathcal{D}|\theta)$

Statistical Pipeline



Gradient Descent

EM



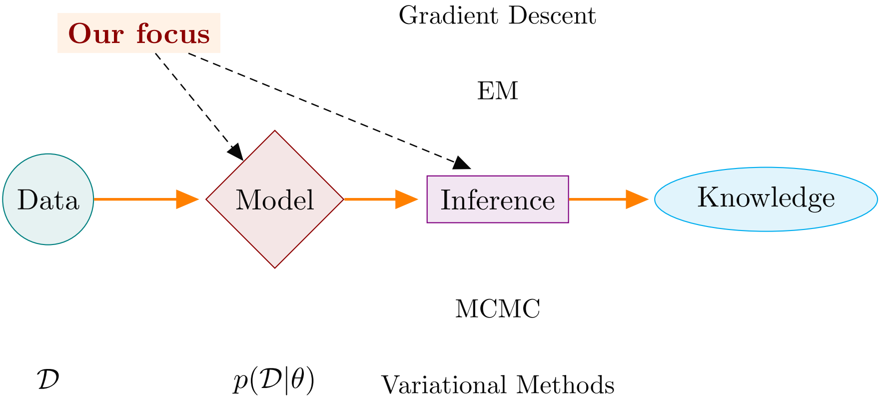
MCMC

$\mathcal{D}$

$p(\mathcal{D}|\theta)$

Variational Methods

Statistical Pipeline



北京大學 PEKING UNIVERSITY

"All models are wrong, but some are useful."

George E. P. Box

Models are used to describe the data generating process, hence prescribe the probabilities of the observed data $\mathcal{D}$

$$p(\mathcal{D}|\theta)$$

also known as the **likelihood**.

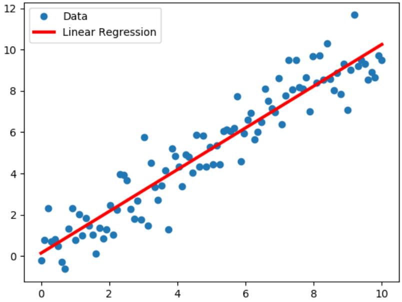
Examples: Linear Models

Data: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$

Model:

$$Y = X\theta + \epsilon, \ \epsilon \sim \mathcal{N}(0, \sigma^2 I_n)$$

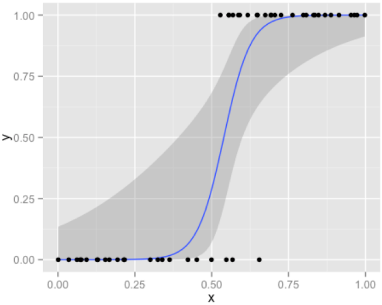
$$\Rightarrow Y \sim \mathcal{N}(X\theta, \sigma^2 I_n)$$



$$p(Y|X, \theta) = (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{\|Y - X\theta\|_2^2}{2\sigma^2}\right)$$

Data:

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n,\ y_i \in \{0, 1\}$$



Model:

$$Y \sim \text{Bernoulli}(p)$$

$$p = \frac{1}{1 + \exp(-X\theta)}$$

$$p(Y|X, \theta) = \prod_{i=1}^{n} p_i^{y_i}(1 - p_i)^{1-y_i}$$

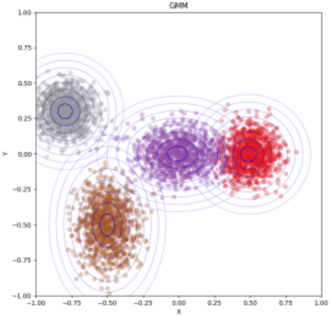
Examples: Gaussian Mixture Model

Data: $\mathcal{D} = \{y_i\}_{i=1}^n, \; y_i \in \mathbb{R}^d$

Model:

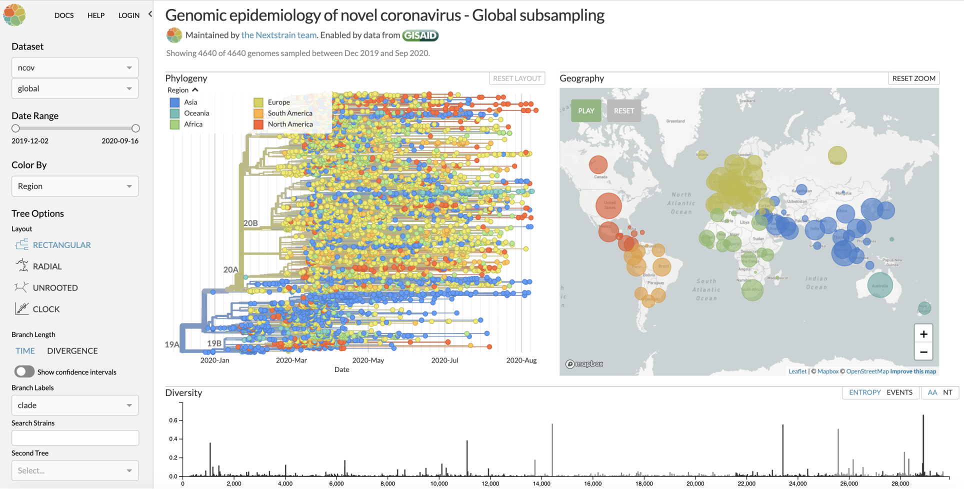
$$y|Z = z \sim \mathcal{N}(\mu_z, \sigma_z^2 I_d)$$

$$Z \sim \text{Categorical}(\alpha)$$



$$p(Y|\mu, \sigma, \alpha) = \prod_{i=1}^{n} \sum_{k=1}^{K} \alpha_k \; (2\pi\sigma_k^2)^{(-d/2)} \exp\left(-\frac{\|y_i - \mu_k\|_2^2}{2\sigma_k^2}\right)$$

Examples: Phylogenetic Model



Examples: Phylogenetic Model

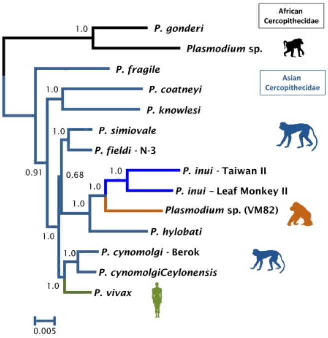
Data: DNA sequences $\mathcal{D} = \{y_i\}_{i=1}^n$

```
CTTTTCAAGG AGTATTTCCT ATGAACGAGT TAGACGGCAT
CATTGCAAAG GGAATAATCT ATGAACGCAA TAATTATTGA
CATTTTCAGG ATAACTTTCT ATGAAAGTAA ACTTAATACT
GAAAAGAAAT CGAGGCAAAA ATGAGCAAAG TCAGACTCGC
TGCAAAAAAA GGAAGACCAT ATGCTTGACG CTCAAACCAT
TTTTTGTGGA GAAGACGCGT GTGATTGTTA AACGACCCGT
GTTATTAAGG ATATGTTCAT ATGTTTTTCA AAAAGAACCT
TACCCACCGG ATTTTTACCC ATGCTCACCG TTAAGCAGAT
AATCAAAATG GAATAAAATC ATGCTACCAT CTATTTCAAT
ATCACAGGGG AAGGTGAGAT ATGCACTCTC AAATCTGGGT
ACATCCAGTG AGAGAGACCG ATGCATCCGA TGCTGAACAT
```

Examples: Phylogenetic Model

Data: DNA sequences $\mathcal{D} = \{y_i\}_{i=1}^{n}$

```
CTTTTCAAGG AGTATTTCCT ATGAACGAGT TAGACGGCAT
CATTGCAAAG GGAATAATCT ATGAACGCAA TAATTATTGA
CATTTTCAGG ATAACTTTCT ATGAAAGTAA ACTTAATACT
GAAAAGAAAT CGAGGCAAAA ATGAGCAAAG TCAGACTCGC
TGCAAAAAAA GGAAGACCAT ATGCTTGACG CTCAAACCAT
TTTTTGTGGA GAAGACGCGT GTGATTGTTA AACGACCCGT
GTTATTAAGG ATATGTTCAT ATGTTTTTCA AAAAGAACCT
TACCCACCGG ATTTTTACCC ATGCTCACCG TTAAGCAGAT
AATCAAAATG GAATAAAATC ATGCTACCAT CTATTTCAAT
ATCACAGGGG AAGGTGAGAT ATGCACTCTC AAATCTGGGT
ACATCCAGTG AGAGAGACCG ATGCATCCGA TGCTGAACAT
```

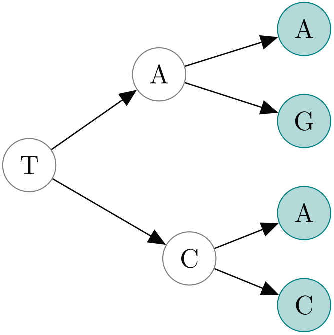


Data: DNA sequences $\mathcal{D} = \{y_i\}_{i=1}^n$

Model: Phylogenetic tree: (τ, q).
Substitution model:

- stationary distribution: $\eta(a_\rho)$.
- transition probability:

$$p(a_u \to a_v | q_{uv}) = P_{a_u a_v}(q_{uv})$$



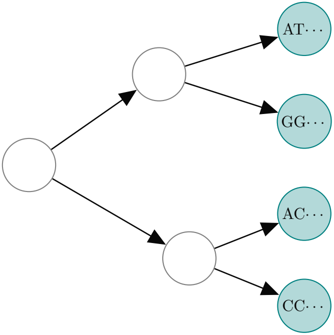
Examples: Phylogenetic Model

Data: DNA sequences $\mathcal{D} = \{y_i\}_{i=1}^n$

Model: Phylogenetic tree: (τ, q).
Substitution model:

- stationary distribution: $\eta(a_\rho)$.
- transition probability:

$$p(a_u \to a_v | q_{uv}) = P_{a_u a_v}(q_{uv})$$



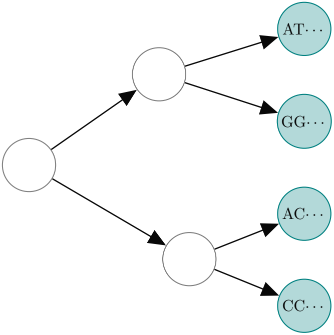
$$p(Y|\tau, q) = \prod_{i=1}^{n} \sum_{a^i} \eta(a_\rho^i) \prod_{(u,v)\in E(\tau)} P_{a_u^i a_v^i}(q_{uv})$$

Data: DNA sequences $\mathcal{D} = \{y_i\}_{i=1}^n$

Model: Phylogenetic tree: (τ, q).
Substitution model:

- stationary distribution: $\eta(a_\rho)$.
- transition probability:

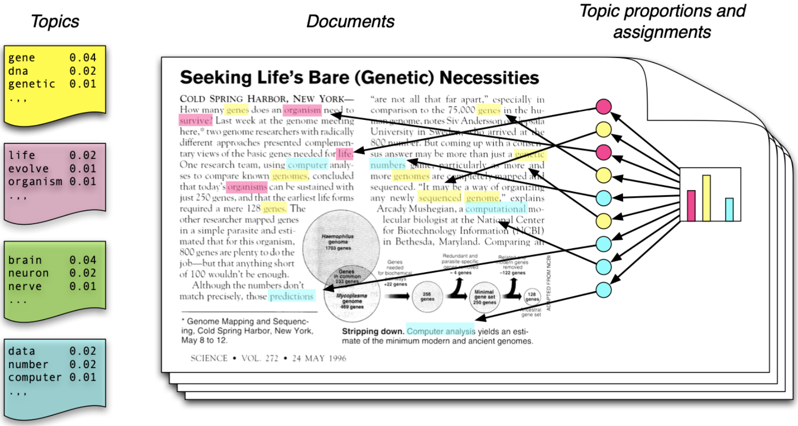
$$p(a_u \to a_v | q_{uv}) = P_{a_u a_v}(q_{uv})$$



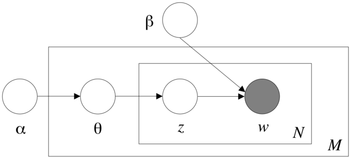
$$p(Y|\tau, q) = \prod_{i=1}^{n} \sum_{a^i} \eta(a_\rho^i) \prod_{(u,v)\in E(\tau)} P_{a_u^i a_v^i}(q_{uv})$$

where a^i agree with y_i at the tips

Examples: Latent Dirichlet Allocation

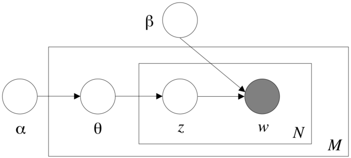


- Each topic is a distribution over words
- Documents exhibit multiple topics



Data: a corpus $\mathcal{D} = \{\boldsymbol{w}_i\}_{i=1}^{M}$

Model: for each document $\boldsymbol{w}$ in $\mathcal{D}$,



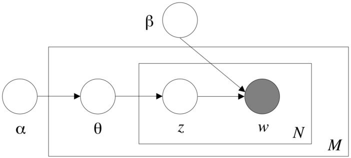
Data: a corpus $\mathcal{D} = \{\boldsymbol{w}_i\}_{i=1}^{M}$

Model: for each document $\boldsymbol{w}$ in $\mathcal{D}$,

- choose a mixture of topics $\theta \sim \mathrm{Dir}(\alpha)$

Examples: Latent Dirichlet Allocation

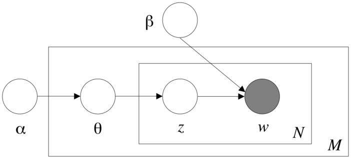
Data: a corpus $\mathcal{D} = \{\boldsymbol{w}_i\}_{i=1}^{M}$



Model: for each document $\boldsymbol{w}$ in $\mathcal{D}$,

- choose a mixture of topics $\theta \sim \mathrm{Dir}(\alpha)$
- for each of the N words w_n,

$$z_n \sim \mathrm{Multinomial}(\theta), \quad w_n | z_n, \beta \sim p(w_n | z_n, \beta)$$



Data: a corpus $\mathcal{D} = \{\boldsymbol{w}_i\}_{i=1}^M$

Model: for each document $\boldsymbol{w}$ in $\mathcal{D}$,

- choose a mixture of topics $\theta \sim \mathrm{Dir}(\alpha)$
- for each of the N words w_n,

$$z_n \sim \mathrm{Multinomial}(\theta), \quad w_n|z_n, \beta \sim p(w_n|z_n, \beta)$$

$$p(\mathcal{D}|\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) \prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \, d\theta_d$$

Many well-known distributions take the following form

$$p(y|\theta) = h(y) \exp\left(\phi(\theta) \cdot T(y) - A(\theta)\right)$$

- $\phi(\theta)$: natural/canonical parameters
- $T(y)$: sufficient statistics
- $A(\theta)$: log-partition function

$$A(\theta) = \log\left(\int_y h(y) \exp(\phi(\theta) \cdot T(y))\, dy\right)$$

Examples: Bernoulli Distribution

$Y \sim \text{Bernoulli}(\theta)$:

$$
\begin{aligned}
p(y|\theta) &= \theta^y (1-\theta)^{1-y} \\
&= \exp\left(\log\left(\frac{\theta}{1-\theta}\right) y + \log(1-\theta)\right)
\end{aligned}
$$

- $\phi(\theta) = \log\left(\frac{\theta}{1-\theta}\right)$
- $T(y) = y$
- $A(\theta) = -\log(1-\theta) = \log(1+e^{\phi(\theta)})$
- $h(y) = 1$

Examples: Gaussian Distribution

$Y \sim \mathcal{N}(\mu, \sigma^2)$:

$$
\begin{aligned}
p(y|\mu, \sigma^2) &= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(y-\mu)^2\right) \\
&= \frac{1}{\sqrt{2\pi}} \exp\left(\frac{\mu}{\sigma^2} y - \frac{1}{2\sigma^2} y^2 - \frac{\mu^2}{2\sigma^2} - \log \sigma\right)
\end{aligned}
$$

- $\phi(\theta) = [\frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2}]^T$
- $T(y) = [y, y^2]^T$
- $A(\theta) = \frac{\mu^2}{2\sigma^2} + \log \sigma$
- $h(y) = \frac{1}{\sqrt{2\pi}}$

$Y = \{y_i\}_{i=1}^n$, $y_i \sim p(y_i|\theta)$, the Log-likelihood

$$L(\theta; Y) = \sum_{i=1}^{n} \log p(y_i|\theta)$$

The gradient of L with respect to θ is called the **score**

$$s(\theta) = \frac{\partial L}{\partial \theta}$$

The expected value of the score is zero

$$\mathbb{E}(s) = \sum_{i=1}^{n} \int \frac{\partial \log p(y_i|\theta)}{\partial \theta} p(y_i|\theta)\ dy_i = \sum_{i=1}^{n} \frac{\partial}{\partial \theta} \int p(y_i|\theta)\ dy_i = 0$$

Fisher Information

Fisher information is the variance of the score.

$$\mathcal{I}(\theta) = \mathbb{E}(ss^T)$$

Under mild assumptions (e.g., exponential families),

$$\mathcal{I}(\theta) = -\mathbb{E}\left(\frac{\partial^2 L}{\partial\theta\partial\theta^T}\right)$$

Intuitively, Fisher information is a measure of the curvature of the Log-likelihood function. Therefore, it reflects the sensitivity of model about the parameter at its current value.

KL Divergence

- Kullback-Leibler divergence or KL divergence is a measure of statistical distance between two distributions $p(x)$ and $q(x)$

$$D_{KL}(q\|p) = \int q(x) \log \frac{q(x)}{p(x)}\, dx$$

- KL divergence is non-negative

$$D_{KL}(q\|p) = -\int q(x) \log \frac{p(x)}{q(x)} \geq -\log \int p(x)\, dx = 0$$

- Consider a family of distributions $p(x|\theta)$, Fisher information is Hessian of KL-divergence between two distributions $p(x|\theta)$ and $p(x|\theta')$ with respect to θ' at $\theta' = \theta$

$$\nabla^2_{\theta'} D_{KL}\left(p(x|\theta)\|p(x|\theta')\right)|_{\theta'=\theta} = \mathcal{I}(\theta)$$

Maximum Likelihood Estimate

$$\hat{\theta}_{MLE} = \arg\max_{\theta} L(\theta) \approx \arg\max_{\theta} \mathbb{E}_{y\sim p_{data}} \log \frac{p(y|\theta)}{p_{data}(y)}$$
$$= \arg\min_{\theta} D_{KL}(p_{data}(y)||p(y|\theta))$$

- **Consistency**. Under weak regularity condition, $\hat{\theta}_{MLE}$ is consistent: $\hat{\theta}_{MLE} \to \theta_0$ in probability as $n \to \infty$, where θ_0 is the "true" parameter
- **Asymptotical Normality**.

$$\hat{\theta}_{MLE} - \theta_0 \to \mathcal{N}(0, \mathcal{I}^{-1}(\theta_0))$$

See Rao 1973 for more details.

Example: Poisson Distribution

$$L(\theta; y_1, \ldots, y_n) = \sum_{i=1}^{n} y_i \log\theta - n\theta - \sum_{i=1}^{n} \log y_i!$$

$$s(\theta) = \frac{\sum_{i=1}^{n} y_i}{\theta} - n, \quad \mathcal{I}(\theta) = \frac{n}{\theta}$$

$$\hat{\theta}_{MLE} = \arg\max_{\theta} \sum_{i=1}^{n} y_i \log\theta - n\theta = \frac{\sum_{i=1}^{n} y_i}{n}$$

By the **Law of large numbers**

$$\hat{\theta}_{MLE} \xrightarrow{p} \theta_0$$

By **central limit theorem**

$$\hat{\theta}_{MLE} - \theta_0 \xrightarrow{d} \mathcal{N}\left(0, \frac{\theta_0}{n}\right)$$

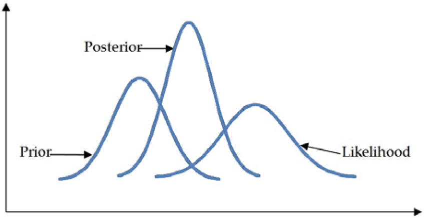
- Can we find an unbiased estimator with smaller variance than $\mathcal{I}^{-1}(\theta_0)$?
- **Cramér-Rao Lower Bound**: For any unbiased estimator $\hat{\theta}$ of θ_0 based on independent observations following the true distribution, the variance of the estimator is bounded by the reciprocal of the Fisher information

$$\mathbb{V}\text{ar}(\hat{\theta}) \geq \frac{1}{\mathcal{I}(\theta_0)}$$

- Sketch of proof: Consider a general estimator $T = t(X)$ with $\mathbb{E}(T) = \psi(\theta_0)$. Let s be the score function,

$$\mathbb{C}\text{ov}(T, s) = \mathbb{E}(Ts) = \psi'(\theta_0) \Rightarrow \mathbb{V}\text{ar}(T) \geq \frac{[\psi'(\theta_0)]^2}{\mathbb{V}\text{ar}(s)} = \frac{[\psi'(\theta_0)]^2}{\mathcal{I}(\theta_0)}$$

In Bayesian statistics, besides specifying a model $p(y|\theta)$ for the observed data, we also specify our **prior** $p(\theta)$ for the model parameters.



Bayes rule for inverse probability

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta) \cdot p(\theta)}{p(\mathcal{D})} \propto p(\mathcal{D}|\theta) \cdot p(\theta)$$

known as the **posterior**.

- uncertainty quantification, provides more useful information
- reducing overfitting. Regularization $\Longleftrightarrow$ Prior.

Prediction

$$p(x|\mathcal{D}) = \int p(x|\theta, \mathcal{D}) p(\theta|\mathcal{D}) d\theta$$

Model Comparison

$$p(m|\mathcal{D}) = \frac{p(\mathcal{D}|m)p(m)}{p(\mathcal{D})}$$

$$p(\mathcal{D}|m) = \int p(\mathcal{D}|\theta, m) p(\theta|m)\, d\theta$$

- **Subjective Priors**. Priors should reflect our beliefs as well as possible. They are subjective, but not arbitrary.
- **Hierarchical Priors**. Priors of multiple levels.

$$
\begin{aligned}
p(\theta) &= \int p(\theta|\alpha)p(\alpha)\, d\alpha \\
&= \int p(\theta|\alpha)\, d\alpha \int p(\alpha|\beta)p(\beta)\, d\beta
\end{aligned}
$$

- **Conjugate Priors**. Priors that ease computation, often used to facilitate the development of inference and parameter estimation algorithms.

Conjugate Priors

- **Conjugacy**: prior $p(\theta)$ and posterior $p(\theta|Y)$ belong to the same family of distribution
- Exponential family

$$p(Y|\theta) \propto \exp\left(\phi(\theta) \cdot \sum_i T(y_i) - nA(\theta)\right)$$

- Conjugate prior

$$p(\theta) \propto \exp\left(\phi(\theta) \cdot \nu - \eta A(\theta)\right)$$

- Posterior

$$p(\theta|Y) \propto \exp\left(\phi(\theta) \cdot (\nu + \sum_i T(y_i)) - (n + \eta)A(\theta)\right)$$

Example: Multinomial Distribution

Data: $\mathcal{D} = \{\boldsymbol{x}_i\}_{i=1}^m$. For each $\boldsymbol{x}$ in $\mathcal{D}$

$$p(\boldsymbol{x}|\theta) \propto \exp\left(\sum_{k=1}^{K} x_k \log \theta_k\right)$$

Use $\text{Dir}(\alpha)$ as the conjugate prior

$$p(\theta) \propto \exp\left(\sum_{k=1}^{K} (\alpha_k - 1) \log \theta_k\right)$$

$$p(\theta|\mathcal{D}) \propto \exp\left(\sum_{k=1}^{K}\left(\alpha_k - 1 + \sum_{i=1}^{M} x_{ik}\right) \log \theta_k\right)$$

Consider random variables $\{X_t\}, t = 0, 1, \ldots$ with state space $\mathcal{S}$

Markov Property

$$p(X_{n+1} = x | X_0 = x_0, \ldots, X_n = x_n) = p(X_{n+1} = x | X_n = x_n)$$

Transition Probability

$$P_{ij}^n = p(X_{n+1} = j | X_n = i), \quad i, j \in \mathcal{S}.$$

A Markov chain is called *time homogeneous* if $P_{ij}^n = P_{ij}, \forall n$.

A Markov chain is governed by its transition probability matrix.

- Stationary Distribution.

$$\pi^T P = \pi^T.$$

- Ergodic Theorem. If the Markov chain is irreducible and aperiodic, with stationary distribution π, then

$$X_n \xrightarrow{d} \pi$$

and for any function h

$$\frac{1}{n}\sum_{t=1}^{n} h(X_t) \to \mathbb{E}_\pi h(X), \quad n \to \infty$$

given $\mathbb{E}_\pi |h(X)|$ exists.

What's Next?

- In general, finding MLE and posterior analytically is difficult. We almost always have to resort to computational methods.
- In this course, we'll discuss a variety of computational techniques for numerical optimization and integration, approximate Bayesian inference methods, with applications in statistical machine learning, computational biology and other related field.

References

- J. Felsenstein. Evolutionary trees from DNA sequences: a maximum likelihood approach. J. Mol. Evol. 17, 368–376 (1981)
- D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. JMLR 3, 2003.
- C. R. Rao. Linear Statistical Inference and its Applications. 2nd edition. New York: Wiley, 1973.
- S. M. Ross. Introduction to Probability Models, 7th ed. Academic, 2000.