STATS 406 Fall 2015
Final Review

December 10, 2015

- Inversion method:
  - Uses CDF, NOT PDF.
  - For simplicity, you can memorize the discrete case separately.
- Rejection sampling:
  - Uses PDF, NOT CDF.
  - The enveloping distribution must be dominant EVERYWHERE.

Random number generation

- Inversion method: If CDF $F(x)$ is known, we can sample $X \overset{\text{CDF}}{\sim} F$ by $F^{-1}(U)$, where $U \sim$ Uniform$(0, 1)$.
- **Question 1.(a):** Sample standard Cauchy: $F(x) = \frac{1}{\pi}\arctan(x) + \frac{1}{2}$.
- **Answer:** First figure out the inverse function F^{-1}:

$$y = \frac{1}{\pi}\arctan(x) + \frac{1}{2}$$

which gives $x = \tan\left(\pi\left(y - \frac{1}{2}\right)\right)$, that is $F^{-1}(t) = \tan\left(\pi\left(t - \frac{1}{2}\right)\right)$. We can sample X by $X := \tan\left(\pi\left(U - \frac{1}{2}\right)\right)$.

Random number generation

- Inversion method: If CDF $F(x)$ is known, we can sample $X \overset{\text{CDF}}{\sim} F$ by $F^{-1}(U)$, where $U \sim$ Uniform$(0,1)$.
- **Question 1.(b):** Sample Geometric(p):

$$\mathbb{P}(X = k) = p(1-p)^{k-1}$$

- **Answer:** The discrete version of the inversion method is a stick breaking algorithm:
  1. Sample $U \sim$ Uniform$(0,1)$. Set $k = 1$, $v = p$.
  2. while(U>v){
     k = k + 1;
     v = v + p*(1-p)^(k-1);
     }
  3. Return k.

Random number generation

- Rejection sampling: Want to sample from PDF $f(x)$, know: 1. how to sample from PDF $g(x)$; 2. for a constant M, $f(x) \leq Mg(x)$ for all x.
  - M doesn't have to be its optimal choice.
  - The domination of Mg over f must hold for all x.
- **Question 1.(c):** Given CDF $F(x) = \sin(\pi x)$ on $\left[0, \frac{1}{2}\right]$, sample from F.
- **Answer:** First derive the corresponding PDF: $f(x) = F'(x) = \pi\cos(\pi x)$. $f(x)$ ranges from π to 0 on $\left[0, \frac{1}{2}\right]$. So we can use the uniform distribution on $\left[0, \frac{1}{2}\right]$ to dominate $f(x)$ with the choice of $M = \frac{\pi}{2}$.

Algorithm:

1. Sample $U \sim \text{Uniform}(0, \frac{1}{2})$.
2. Accept U with probability $\dfrac{\pi\cos(\pi U)}{\frac{\pi}{2}\cdot 2} = \cos(\pi U)$.

Monte-Carlo integration

- Remember and understand the rewriting-of-integral. It is the starting point of all methods taught in this course.

Monte-Carlo integration

- All Monte-Carlo integration techniques start with the common insight:

$$I = \int f(x)\mathrm{d}x = \int \frac{f(x)}{\pi(x)}\pi(x)\mathrm{d}x = \mathbb{E}\left[\frac{f(X)}{\pi(X)}\right]$$

  where $X \overset{\text{PDF}}{\sim} \pi(x)$.

  They only differ in choices of $\pi(x)$ and/or ways to compute $\mathbb{E}\left[\frac{f(X)}{\pi(X)}\right]$.

Monte-Carlo integration

- Plain Monte-Carlo: Use a uniform distribution as $\pi(x)$.
- **Question 2.(a):** Use Uniform$(1, 3)$.

Algorithm:

1. Sample $X_1, \ldots, X_n \sim$ Uniform$(1, 3)$.
2. Estimate I by

$$\hat{I} = \frac{2}{n} \sum_{i=1}^{n} \frac{1}{X_i^2}$$

Where does the factor 2 come from?

Monte-Carlo integration

- Importance sampling: To improve efficiency, choose $\pi(x)$ that mimics the shape of $f(x)$.
- **Question 2.(b):** Compute $\mathbb{E}[Y]$, where $Y = X^3 \mathbb{1}[X > 0]$, $X \sim N(0,1)$.
- **Answer:** First write the expectation in integration form:

$$I = \int_0^{+\infty} x^3 \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \mathrm{d}x = \int_0^{+\infty} \frac{x^3 \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}}{\pi(x)} \pi(x) \mathrm{d}x = \mathbb{E}\left[\frac{X^3 \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{X^2}{2}}}{\pi(X)} \right]$$

for $X \overset{\text{PDF}}{\sim} \pi(x)$. As required by the question, choose $\pi(x)$ to be the PDF of the standard exponential distribution, that is, $\pi(x) = e^{-x}$ for $x > 0$.

Algorithm:

1. Sample $X_1, \ldots, X_n$ from standard exponential distribution.
2. Estimate I by

$$\hat{I} = \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{X_i^3 \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{X_i^2}{2}}}{e^{-X_i}} \right\}$$

Monte-Carlo integration

- Importance sampling (self-normalized): Ordinary importance sampling requires knowing $f(x)$ exactly. When $f(x)$ is only known up to a constant, the self-normalized version of importance sampling should be used.
- **Question 2.(c):** Compute $\mathbb{E}[X]$, where $X \overset{\text{PDF}}{\sim} f(x) \propto e^{-x^{3/2}}$.
- **Answer:** Recall the derivation of the self-normalized importance sampling:

$$I = \mathbb{E}[X] = \int x f(x) \mathrm{d}x = \int \frac{x f(x)}{\pi(x)} \pi(x) \mathrm{d}x = \mathbb{E}\left[\frac{X f(X)}{\pi(X)}\right] = \frac{\mathbb{E}\left[\frac{X f(X)}{\pi(X)}\right]}{\mathbb{E}\left[\frac{f(X)}{\pi(X)}\right]}$$

where $X \overset{\text{PDF}}{\sim} \pi(x)$. With $X_1, \ldots, X_n$ generated from PDF $\pi(x)$, we use the self-normalized importance sampling:

$$\hat{I} = \frac{\frac{1}{n} \sum_{i=1}^n \frac{X_i f(X_i)}{\pi(X_i)}}{\frac{1}{n} \sum_{i=1}^n \frac{f(X_i)}{\pi(X_i)}} = \frac{\sum_{i=1}^n \frac{X_i f_0(X_i)}{\pi(X_i)}}{\sum_{i=1}^n \frac{f_0(X_i)}{\pi(X_i)}}$$

where $f_0(x) := e^{-x^{3/2}} \propto f(x)$ with unknown constant. Blue part: the ordinary importance sampling estimator if $f(x)$ is fully known.

Monte-Carlo performance evaluation and bootstrap

- First understand the Monte-Carlo performance evaluation, whose logic is very straightforward.
- Then bootstrap is all about *replacement*: we don't know the needed ingredients in the Monte-Carlo performance evaluation, so we use their estimations instead.

Monte-Carlo performance evaluation and bootstrap

- Definition of mean-squared error(MSE).
- **Question 3.(a) answer:** $\text{MSE}(\widehat{\mu^2}) = \mathbb{E}\left[\left(\widehat{\mu^2} - \mu^2\right)^2\right]$.

Monte-Carlo performance evaluation and bootstrap

- Monte-Carlo performance evaluation: if we know the true values of the parameter, we can generate simulated data from the population. Draw many samples to evaluate the accuracy of an estimator.
- **Question 3.(b):** Consider $N(\mu, 1)$ and the estimator $\widehat{\mu^2} = (\bar{X})^2$. If μ is known how to compute $\text{MSE}(\widehat{\mu^2})$?
- **Answer:** Very straightforward:
  1. Generate m independent samples $X^{(1)}, \ldots, X^{(m)}$, each of size n.
  2. Compute the square of each sample mean: $\widehat{\mu^2}^{(i)} = (\text{mean}(X^{(i)}))^2$.
  3. Estimate the MSE: $\widehat{\text{MSE}}(\widehat{\mu^2}) = \frac{1}{m}\sum_{i=1}^m \left((\bar{X}^{(i)})^2 - \mu^2 \right)^2$
- **Remark:** Rigorously speaking, the MSE here depends on the sample size. We should have stated that $\widehat{\mu^2}$ is estimating the MSE at sample size n.
- **Remark:** Other tasks, e.g. testing, CI, ... only differ in Step 3.

Monte-Carlo performance evaluation and bootstrap

- Bootstrap: When the true parameter value is unknown, the population distribution becomes unknown, too. We carry out the evaluation procedures referring to the replacement chart as follows:

Monte-Carlo evaluation	Bootstrap
true parameter value	estimated parameter value
population distribution	sample distribution
independent samples	resamples
sample statistics	sample statistics of resamples

Monte-Carlo performance evaluation and bootstrap

- Bootstrap:
- **Question 3.(c):** Consider $N(\mu, 1)$ and the estimator $\widehat{\mu^2} = (\bar{X})^2$. If μ is unknown how to estimate MSE$(\widehat{\mu^2})$?
- **Answer:** Following the replacement chart to derive the algorithm:
  1. Use the estimated parameter in place of the true parameter: $\widehat{\mu^2} = (\bar{X})^2$
  2. Draw resamples $X^{*(1)}, \ldots, X^{*(m)}$, each of size n.
  3. Compute the square of each sample mean: $\widehat{\mu^2}^{*(i)} = (\mathrm{mean}(X^{*(i)}))^2$.
  4. Estimate the MSE: $\widehat{\mathrm{MSE}}^*(\widehat{\mu^2}) = \frac{1}{m}\sum_{i=1}^m \left((\bar{X}^{*(i)})^2 - \widehat{\mu^2}\right)^2$.

SQL

- The order to read and write SQL scripts.
- When joining tables, specify table names when citing variables appearing in more than one tables.

SQL

- Recall the order in which we read an SQL script. This is also the order in which we write an SQL script.
  1. FROM (including INNER JOIN)
  2. WHERE
  3. GROUP BY
  4. HAVING
  5. SELECT
  6. ORDER BY

SQL

- **Question 4.(a):** Query all *pianists* from *Soviet*. Only report *pianist* and *country*.
- **Answer:**

```
SELECT Pianist, Country
FROM Pianists
WHERE Country="Soviet"
```

SQL

- **Question 4.(b):** Query the table *Works* and summarize the number of *works* performed by *pianist*. Only report *pianist* and the number of works performed.
- **Answer:**
```
SELECT Pianists, Count(Title) as NumberOfWorksPerformed
FROM Works
GROUP BY Pianists
```
- **Remark:** also correct:
```
SELECT Pianists, Count(Pianists) as NumberOfWorksPerformed
```

SQL

- **Question 4.(c):** Combine tables *Works* and *Pianists* and query works played by *European(including Soviet) pianists*. Only report *title*, *composer* and *pianist*.
- **Answer:**
  SELECT Title, Composer, Pianists.Pianist AS Pianist
  FROM
  Works INNER JOIN Pianists
  ON Works.Pianist = Pianists.Pianist
  WHERE Pianists.Pianist = "Soviet" OR Pianists.Pianist = "Germany"
  OR Pianists.Pianist = "Austria"

SQL

- **Question 4.(d):** Combine all three tables and query works composed by *German composers* and performed by *Soviet pianists*. Only report *title*, *composer* and *pianist*.
- **Answer:**
  First combine the first two tables:

  SELECT Title, Composers.Composer AS Composer, Works.Pianist AS Pianist
  FROM Composers INNER JOIN Works
  ON Composers.Composer = Works.Composer
  WHERE Composers.Country = "Germany"

SQL

- **Question 4.(d):** Combine all three tables and query works composed by *German composers* and performed by *Soviet pianists*. Only report *title*, *composer* and *pianist*.
- **Answer:**
  Then combine this table (in blue) with the third table:

```
SELECT Title, T1.Composer AS Composer, T1.Pianist AS Pianist
FROM
(
SELECT Title, Composers.Composer AS Composer, Works.Pianist AS
Pianist
FROM Composers INNER JOIN Works
ON Composers.Composer = Works.Composer
WHERE Composers.Country = "Germany"
) AS T1
INNER JOIN
Pianists
ON T1.Pianist = Pianists.Pianist
WHERE Pianists.Country = "Soviet"
```

XML

- **Question 5:** Rewrite the following entry, transforming the attributes into children:

```
<book Title="The Return of The Native"
Author="Thomas Hardy" PublishedYear="1878" />
```

  Consider the rewritten version: write an R command (assume the package "XML" is loaded and *root* points to the *book* tag) to query the content of the *PublishedYear* tag. The returned value must be numeric.

- **Answer:** Rewrite the entry:

```
<book>
<Title>The Return of The Native</Title>
<Author>Thomas Hardy</Author>
<PublishedYear>1878</PublishedYear>
</book>
```

  Notice: 1. no quote marks needed; 2. remember to close each tag; 3. XML is case-sensitive.

  Query the *PublishedYear* tag:

```
as.numeric(xmlValue(root[["PublishedYear"]]))
```

Optimization

- Set derivative to 0 and solve. Lagrangian multiplier.
- Gradient methods:
  - Tell the scenario and determine the correct tool to use (descend or ascend?).
- Newton's method:
  - Hessian matrix
  - (Sometimes useful) matrix inversion

Optimization

- Use the illustration in Lab_11.pdf to help you memorize the formulations of gradient methods and Newton's method.
- Those illustrations are univariate, but once you have the formulation, it's easy to extend them to the multivariate case.

Optimization

- **Question 6.(a):** Optimize $f(x, y) = x^2 + 4(y-1)^2$, starting at $(x_0, y_0) = (2, 3)$. Use gradient method and Newton's method.
- **Answer:** First compute the gradient: $\nabla f(x, y) = (2x, 8(y-1))^T$.

Gradient method: minimization $\Rightarrow$ gradient descend.

$$\begin{aligned}(x_1, y_1) &= (x_0, y_0) - \text{StepSize} \cdot \nabla f(x_0, y_0) \\ &= (2, 3) - 0.1 \cdot (4, 16) = (1.6, 1.4) \\ (x_2, y_2) &= (x_1, y_1) - \text{StepSize} \cdot \nabla f(x_1, y_1) \\ &= (1.6, 1.4) - 0.1 \cdot (3.2, 3.2) = (1.28, 1.08)\end{aligned}$$

Newton's method: calculate Hessian: $\frac{\partial^2 f}{\partial x^2} = 2$, $\frac{\partial^2 f}{\partial x \partial y} = 0$ and $\frac{\partial^2 f}{\partial y^2} = 8$, so

$$\begin{pmatrix} x_{t+1} \\ y_{t+1} \end{pmatrix} = \begin{pmatrix} x_t \\ y_t \end{pmatrix} - H^{-1}(x_t, y_t) \cdot \nabla f(x_t, y_t) = \begin{pmatrix} x_t \\ y_t \end{pmatrix} - \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{8} \end{pmatrix} \begin{pmatrix} 2x_t \\ 8(y_t - 1) \end{pmatrix}$$

$$= \begin{pmatrix} x_t \\ y_t \end{pmatrix} - \begin{pmatrix} x_t \\ y_t - 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Newton's method converges to the optimum after only one iteration, and will stay there forever.

Optimization

- **Question 6.(b):** Poisson mixture: 3 Poisson distributions with λ_k and mixing probabilities π_k, $k = 1, 2, 3$. Observe $X_1, \ldots, X_n$. Write down and maximize the incomplete log-likelihood.
- **Answer:** By total probability theorem, we have the likelihood:

$$\mathbb{P}(X_i = k) = \sum_{j=1}^{3} \mathbb{P}(X_i = k | Z_i = j)\, \mathbb{P}(Z_i = j) = \sum_{j=1}^{3} \frac{\lambda_j^k}{k!} e^{-\lambda_j} \pi_j$$

Therefore, the log-likelihood function is:

$$l(\Theta|X) = \sum_{i=1}^{n} \log \left(\sum_{j=1}^{3} \frac{\lambda_j^k}{k!} e^{-\lambda_j} \pi_j \right)$$

The gradient method we should use here is a gradient ascend as follows:

$$\Theta_{t+1} = \Theta_t + \mathrm{StepSize} \cdot \nabla l(\Theta_t | X)$$

EM algorithm

- Complete log-likelihood
- Conditional probability and conditional expectation
- Bayes formula and total probability theorem

EM algorithm

- **Question 7:** Poisson mixture: 3 Poisson distributions with λ_k and mixing probabilities π_k, $k = 1, 2, 3$. Observe $X_1, \ldots, X_n$. Write down and maximize the incomplete log-likelihood.
- **Answer:** The question is not yet completely ready to apply EM algorithm upon. We need to first finish modeling by introducing latent random variables.
  Set $Z_i \in \{1, 2, 3\}$ to be a categorical random variable that indicates which Poisson distribution generates X_i. The ith term in the complete likelihood function is:
  $$\mathbb{P}(X_i = k, Z_i = j) = \frac{\lambda_j^k}{k!} e^{-\lambda_j} \cdot \pi_j$$
  Employing the indicator function $\mathbb{1}[Z_i = j]$, the log-likelihood function is:
  $$\begin{aligned} l_c(\Theta; X, Z) &= \sum_{i=1}^{n} \left\{ \sum_{j=1}^{3} \mathbb{1}[Z_i = j] \log \left(\frac{\lambda_j^{X_i}}{X_i!} e^{-\lambda_j} \cdot \pi_j \right) \right\} \\ &= \sum_{i=1}^{n} \left\{ \sum_{j=1}^{3} \mathbb{1}[Z_i = j] \left(X_i \log \lambda_j - \lambda_j + \log \pi_j \right) \right\} + \text{constant} \end{aligned}$$

EM algorithm

- **Answer(continued):** The complete log-likelihood:

$$l_c(\Theta; X, Z) = \sum_{i=1}^{n} \left\{ \sum_{j=1}^{3} \mathbb{1}[Z_i = j] \left(X_i \log \lambda_j - \lambda_j + \log \pi_j \right) \right\} + \text{constant}$$

- E-step: calculate $\mathbb{E}\left[l_c(\Theta; X, Z) | \Theta_t, X\right]$. Notice that here l is linear in $\mathbb{1}[Z_i = j]$, which only depends on X_i, it suffices to evaluate $\mathbb{E}\left[\mathbb{1}[Z_i = j] | \Theta_t, X_i\right]$. By Bayes formula and total probability theorem:

$$\mathbb{E}\left[\mathbb{1}[Z_i = j] | \Theta_t, X_i\right] = \mathbb{P}\left(Z_i = j | \Theta_t, X_i\right) \overset{\text{B.F.}}{=} \frac{\mathbb{P}\left(X = X_i | Z_i = j, \Theta_t\right) \mathbb{P}\left(Z_i = j | \Theta_t\right)}{\mathbb{P}\left(X = X_i | \Theta_t\right)}$$

$$\overset{\text{T.P.F}}{=} \frac{\mathbb{P}\left(X = X_i | Z_i = j, \Theta_t\right) \mathbb{P}\left(Z_i = j | \Theta_t\right)}{\sum_{\tilde{j}=1}^{3} \mathbb{P}\left(X = X_i | Z_i = \tilde{j}, \Theta_t\right) \mathbb{P}\left(Z_i = \tilde{j} | \Theta_t\right)}$$

$$= \frac{\dfrac{\left(\lambda_j^{(t)}\right)^{X_i} e^{-\lambda_j^{(t)}}}{X_i!} \cdot \pi_j^{(t)}}{\sum_{\tilde{j}=1}^{3} \left\{ \dfrac{\left(\lambda_{\tilde{j}}^{(t)}\right)^{X_i} e^{-\lambda_{\tilde{j}}^{(t)}}}{X_i!} \cdot \pi_{\tilde{j}}^{(t)} \right\}} =: \langle \mathbb{1}[Z_i = j] \rangle$$

EM algorithm

- **Answer(continued):**
- M-step: replacing all $\mathbb{1}[Z_i = j]$ in the complete log-likelihood by $\langle \mathbb{1}[Z_i = j] \rangle$, we have

$$\mathbb{E}\left[l_c(\Theta; X, Z) | \Theta_t, X\right] = \sum_{i=1}^{n} \left\{ \sum_{j=1}^{3} \langle \mathbb{1}[Z_i = j] \rangle \left(X_i \log \lambda_j - \lambda_j + \log \pi_j \right) \right\} + \text{constant}$$

By taking the derivative of $\mathbb{E}\left[l_c(\Theta; X, Z) | \Theta_t, X\right]$ over each λ_j respectively and setting it to zero, we immediately have:

$$\lambda_j^{(t+1)} = \frac{\sum_{i=1}^{n} \langle \mathbb{1}[Z_i = j] \rangle X_i}{\sum_{i=1}^{n} \langle \mathbb{1}[Z_i = j] \rangle}$$

for $j = 1, 2, 3$.

EM algorithm

- **Answer(continued):**
- M-step(continued): recall that

$$\mathbb{E}\left[l_c(\Theta; X, Z)|\Theta_t, X\right] = \sum_{i=1}^{n}\left\{\sum_{j=1}^{3}\langle\mathbb{1}[Z_i = j]\rangle\left(X_i \log\lambda_j - \lambda_j + \log\pi_j\right)\right\} + \text{constant}$$

Obtaining the update for π_i's is slightly harder due to the constraint $\sum_{j=1}^{3}\pi_j =$. We consider the corresponding terms plus the Lagrangian multiplier:

$$\sum_{i=1}^{n}\left\{\sum_{j=1}^{3}\left(\mathbb{1}[Z_i = j]\log\pi_j\right)\right\} - \alpha\left(\pi_1 + \pi_2 + \pi_3 - 1\right)$$

and set its derivative to zero. We have

$$\alpha = \frac{\sum_{i=1}^{n}\mathbb{1}[Z_i = 1]}{\pi_1} = \frac{\sum_{i=1}^{n}\mathbb{1}[Z_i = 2]}{\pi_2} = \frac{\sum_{i=1}^{n}\mathbb{1}[Z_i = 3]}{\pi_3}$$

Therefore

$$\pi_j^{(t+1)} = \frac{\sum_{i=1}^{n}\mathbb{1}[Z_i = j]}{n}$$

Final tips

- Keep your local lecture/lab notes up to date.
- Looking for more practice questions? Redo the homework.
- Make a “cheating sheet”. This helps yourself summarize the content and realize what is important. I always found this very helpful.
  The exam is closed-book, DO NOT use any cheating sheet in the exam!!