Speech and Language Processing

Lecture 2
Probability Distributions, Markov Model, Hidden Markov Model

Information and Communications Engineering Course

Takahiro Shinozaki

Lecture Plan (Shinozaki's part)

I gives the first 6 lectures about speech recognition. Through these lectures, the backbone of the latest speech recognition techniques is explained.

1. 10/4 (remote)
   Introduction and Preparation
2. 10/4 (remote)
   Probability Distributions, Markov Models, Samplings
3. 10/6 (remote)
   Maximum Likelihood Estimation and EM Algorithm
4. 10/6 (remote)
   Bayesian Networks and Bayesian Inference
5. 10/7 (remote)
   Neural networks
6. 10/7 (remote)
   Reinforcement Learning

ASR by Statistical Modeling

- Uses probability distributions to model speech sounds

W : Symbols (Categories) to recognize
e.g. phone, word, word sequence, etc.

O: Input acoustic features

$$\hat{W} = \underbrace{\underset{W}{\operatorname{argmax}}}_{\text{Speech recognizer}} \underbrace{P(W|O)}_{\text{Speech model}}$$

Two Modeling Approaches

- Discriminative model
  - Directly models the conditional probability $P(W|O)$ of a recognition symbol W given an observation O

$$\hat{W} = \underset{W}{\operatorname{argmax}}\, P(W|O)$$

- Generative model
  - Uses the Bayes' theorem and models the generation process

$$\hat{W} = \underset{W}{\operatorname{argmax}}\, P(W|O) \qquad P(W|O) = \frac{P(O|W)P(W)}{P(O)}$$

Acoustic Model and Language Model

- In the generative modeling approach, the probability is decomposed into two parts
- P(*O*) is independent of the maximization of *W*, and can be ignored

$$\hat{W} = \underset{W}{\operatorname{argmax}} P(W|O)$$

$$= \underset{W}{\operatorname{argmax}} \frac{P(O|W)P(W)}{P(O)}$$

$$= \underbrace{\underset{W}{\operatorname{argmax}}}_{\text{Speech recognizer}} \underbrace{P(O|W)}_{\text{Acoustic model (AM)}} \underbrace{P(W)}_{\text{Language model (LM)}}$$

Problem Settings

- Frame-wise vowel recognition
  - ***O***: feature vector of a single frame
  - ***W***: One of the vowels (For Japanese: a,i,u,e,o)
- Isolated phone recognition
  - ***O***: A sequence of feature vectors of a segment of phone utterance
  - ***W***: One of the phones
- Isolated word recognition
  - ***O***: A sequence of feature vectors of a segment of word utterance
  - ***W***: One of the words in a vocabulary
- Continuous word recognition
  - ***O***: A sequence of feature vectors of an utterance
  - ***W***: Sequence of words

How to Models the Probabilities?

- When ***O*** is a feature vector and ***W*** is a vowel:

$P(W|O)$ $\quad$ $P(O|W)$ $\quad$ $P(W)$

W : categorical variable

O : real valued vector variable

- When ***O*** is a sequence of feature vectors and ***W*** is a sequence of phones or words:

$P(W|O)$ $\quad$ $P(O|W)$ $\quad$ $P(W)$

W : a sequence of categorical variables

O : a sequence of real valued vector variables

Categorical Distribution

- The distribution is represented by a table
- The probability distribution of a skewed die is an example of categorical distribution

Vowel	a	i	u	e	o
Probability	0.3	0.1	0.2	0.1	0.3

1-of-K Representation

- The same probability as the table description can be represented as an equation by using 1-of-K representation

Value W	1-of-K representation $W=(w_1, w_2, w_3, w_4, w_5)$	Probability $\boldsymbol{\rho}=(\rho_1, \rho_2, \rho_3, \rho_4, \rho_5)$
1 （a）	1,0,0,0,0	$\Pr(W=1)=\rho_1=0.3$
2 （i）	0,1,0,0,0	$\Pr(W=2)=\rho_2=0.1$
3 （u）	0,0,1,0,0	$\Pr(W=3)=\rho_3=0.2$
4 （e）	0,0,0,1,0	$\Pr(W=4)=\rho_4=0.1$
5 （o）	0,0,0,0,1	$\Pr(W=5)=\rho_5=0.3$

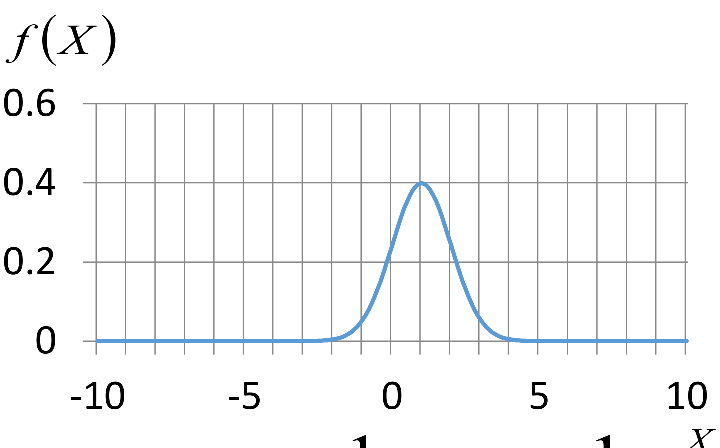
$$p(W)=\prod_{k=1}^{K}\rho_k^{\,w_k}$$

Gaussian Distribution

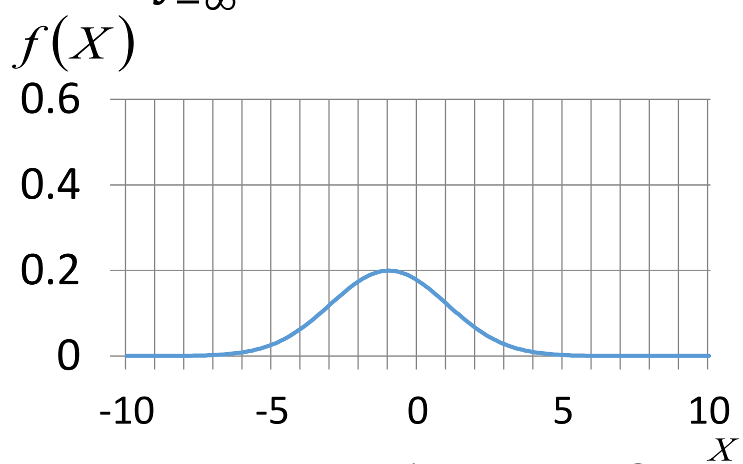
- Defined by two parameters mean μ and standard deviation σ (σ^2 is variance)

$$N(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$

It satisfies: $$0 < N(x|\mu,\sigma^2), \quad \int_{-\infty}^{\infty} N(x|\mu,\sigma^2)\,dx = 1$$



$\mu = 1, \quad \sigma = 1$



$\mu = -1, \quad \sigma = 2$

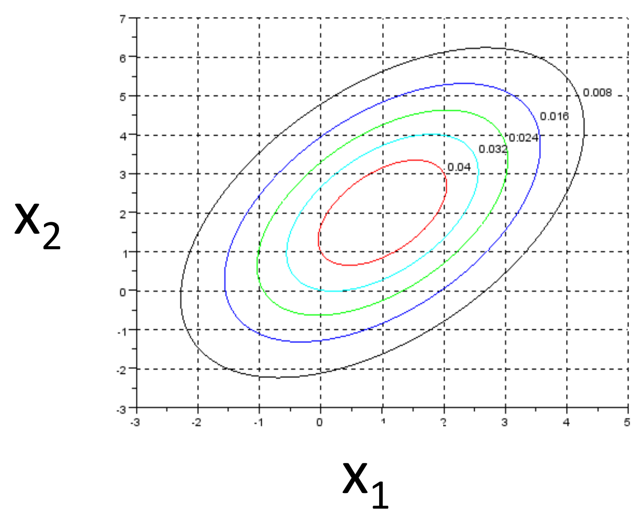
Multivariate Gaussian Distribution

- For D-dimensional vector **x**, it is defined using a mean vector **μ** and a covariance matrix **Σ**:

$$N(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D|\boldsymbol{\Sigma}|}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

| **Σ** | : determinant of **Σ**

Contour plot of an example of a two dimensional Gaussian distribution



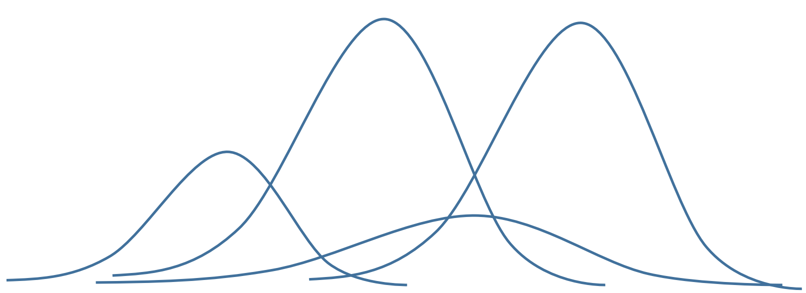
Gaussian Mixture Model (GMM)

- By mixing multiple Gaussian distributions, a complex distribution can be expressed
  → Useful to improve recognition performance

$$GMM(X) = \sum_i w_i N_i(X|\mu_i, S_i) \qquad \sum_{m=1}^{M} w_k = 1.0$$

w_i : Mixture weight

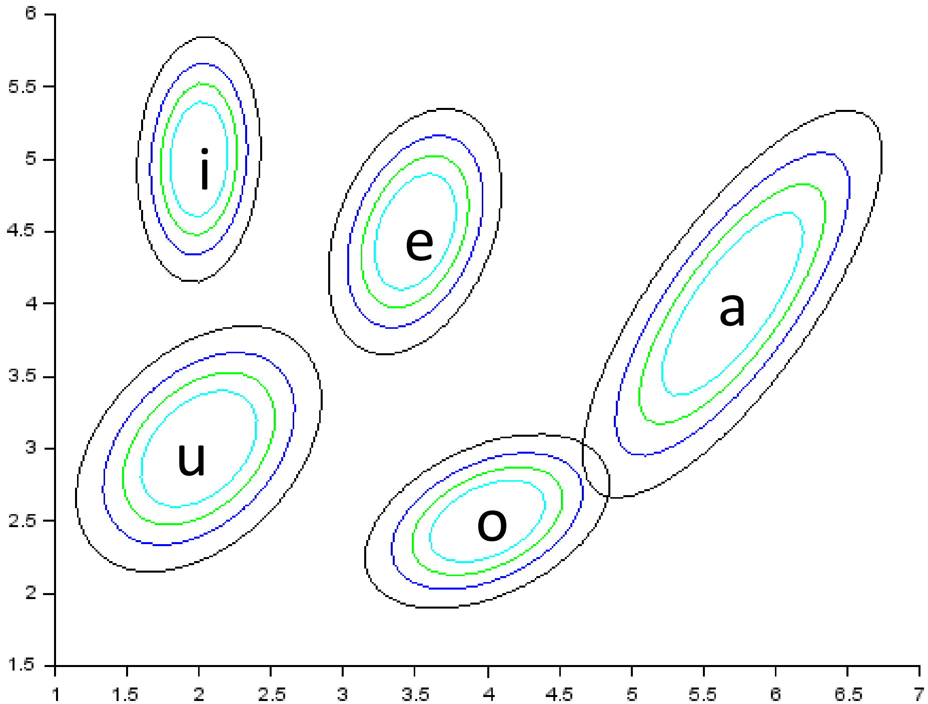
N_i : Component Gaussian distribution with mean μ_i and covariance S_i



Gaussian Distribution based AM

- When ***W*** is categorical (e.g. a vowel)**,** we can fit a Gaussian distribution for each category

$$P_w(O) = N(x|\mu_w, {\sigma_w}^2) = \frac{1}{\sqrt{2\pi {\sigma_w}^2}} \exp\left\{-\frac{1}{2{\sigma_w}^2}(x-\mu_w)^2\right\}$$



How to fit the distributions?

We will consider this problem later in the lecture of maximum likelihood estimation

Example of Frame-wise Vowel Recognition

- Gaussian distribution based acoustic model
- Categorical distribution based language model

$$\hat{W} = \underset{W \in \{a,i,u,e,o\}}{\operatorname{argmax}} P(W|O)$$

$$= \underset{W \in \{a,i,u,e,o\}}{\operatorname{argmax}} N(O|\mu_W, \Sigma_W) \prod_k \rho_k^{w_k}$$

Modeled by a set of Gaussian distributions fitted for each W

Modeled by a categorical distribution

Markov Model

A probability model $\mathrm{P}(S)$ of a sequence of states $S = s_1, s_2, \cdots$ where Markov property $\mathrm{P}(s_{t+1}, s_{t+2}, \cdots \mid s_1, s_2, \cdots, s_t) = \mathrm{P}(s_{t+1}, s_{t+2}, \cdots \mid s_t)$ holds

Future depends only on the current state

$$P(S) = P(s_1, s_2, s_3, \cdots, s_T)$$

$$= P(s_1)P(s_2|s_1)P(s_3, s_4, \cdots, s_T|s_1, s_2)$$

Product-rule

$$= P(s_1)P(s_2|s_1)P(s_3, s_4, \cdots, s_T|s_2)$$

Markov property

$$= P(s_1)P(s_2|s_1)P(s_3|s_2)P(s_4, \cdots, s_T|s_2, s_3)$$

Product-rule

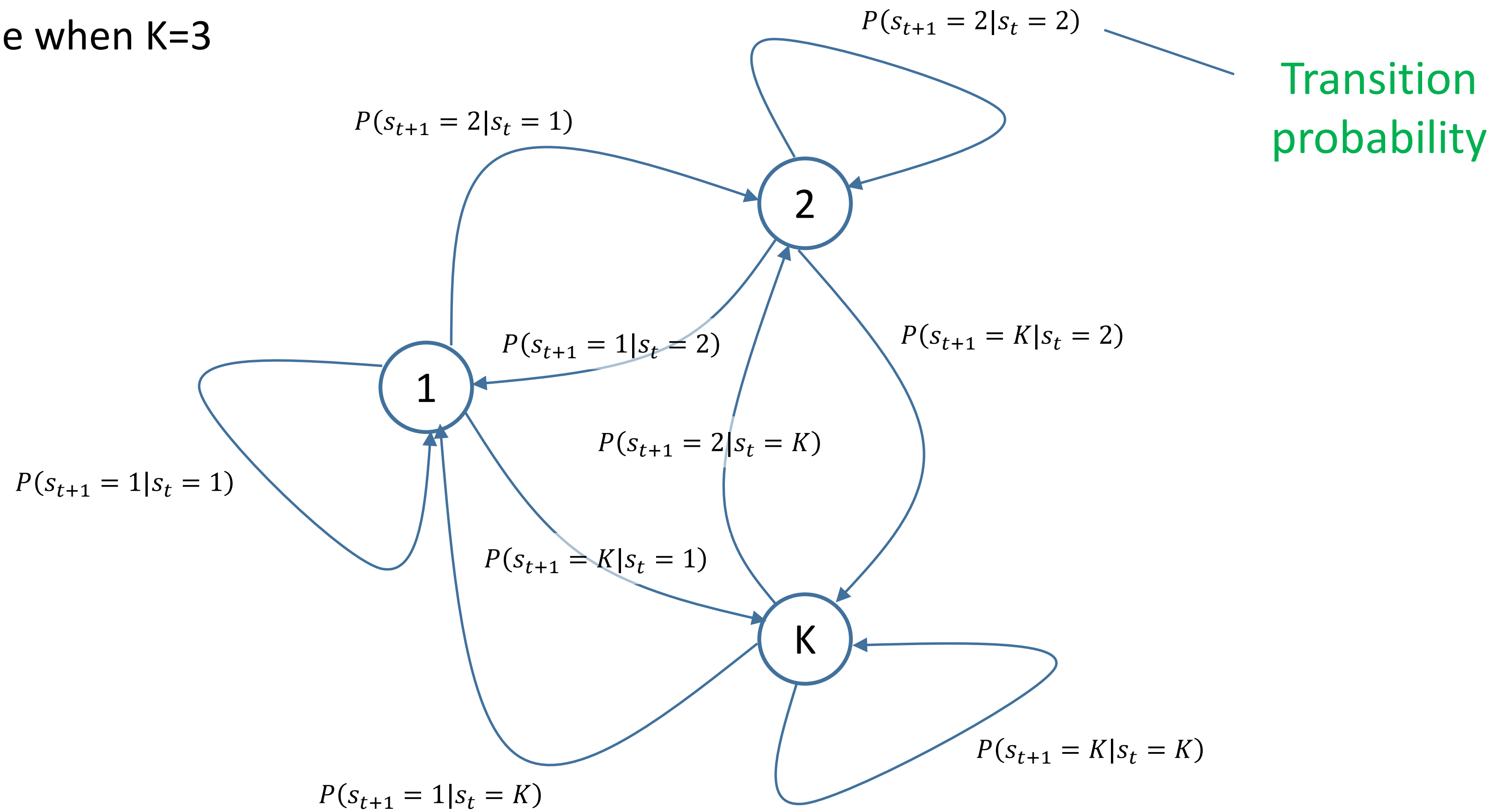
$$\vdots$$

$$= P(s_1)P(s_2|s_1)P(s_3|s_2)P(s_4|s_3) \cdots P(s_T|s_{T-1})$$

Graph Representation of Markov Model

Assumes s_t is a categorical variable taking one of $\{1,2,\cdots,K\}$

Example when K=3



N-gram Word Model

- Assumes that the appearance of a word in an utterance depends on at max N-1 preceding words as context
- A Markov model for a word sequence
- Represented by a set of categorical distributions prepared for each context

E.g. When N=2

$$P(w_1 w_2 w_3 \cdots w_T)$$

$$= P(w_1)P(w_2|w_1)P(w_3|w_1 w_2)P(w_4|w_1 w_2 w_3) \cdots P(w_T|w_1 w_2 \cdots w_{T-1})$$

$$\approx P(w_1)P(w_2|w_1)P(w_3|w_2)P(w_4|w_3) \cdots P(w_T|w_{T-1})$$

Unigram

- N=1
  - Do not consider the history at all
  - Same as the product of individual word probabilities

$$P(w_1 w_2 w_3 \cdots w_T)$$
$$\approx P(w_1)P(w_2)P(w_3)P(w_4) \cdots P(w_T)$$
$$= \prod_{t=1}^{T} P(w_t)$$

Example：

P(“Today is a sunny day”) =
P(“today”)P(“is”)P(“a”)P(“sunny”)P(“day”)

Bi-gram

- N=2
  - Consider only a previous word as the history

$$P(w_1 w_2 w_3 \cdots w_T)$$
$$\approx P(w_1)P(w_2|w_1)P(w_3|w_2)P(w_4|w_3)\cdots P(w_T|w_{T-1})$$
$$= P(w_1)\prod_{t=2}^{T} P(w_t|w_{t-1})$$

Example：

P("Today is a sunny day") =

P(today)P(is|today)P(a|is)P(sunny|a)P(day|sunny)

Tri-gram

- N=3
  - Consider two previous words as the history
  - Popular in speech recognition

$$P(w_1 w_2 w_3 \cdots w_T)$$
$$\approx P(w_1)P(w_2|w_1)P(w_3|w_1 w_2)P(w_4|w_2 w_3)\cdots P(w_T|w_{T-2}w_{T-1})$$
$$= P(w_1)P(w_2|w_1)\prod_{t=3}^{T} P(w_t|w_{t-2}, w_{t-1})$$

Example：

P(“Today is a sunny day”) =

P(today)P(is|today)P(a|today, is)P(sunny|is, a)P(day|a, sunny)

Hidden Markov Model (HMM)

A joint probability model $\mathrm{P}(S, O)$ of a sequence of states $S = s_1, s_2, \cdots$ and a sequence of observations $O = o_1, o_2, \cdots$ where Markov property $\mathrm{P}(s_{t+1}, s_{t+2}, \cdots \mid s_1, s_2, \cdots, s_t) = \mathrm{P}(s_{t+1}, s_{t+2}, \cdots | s_t)$ holds with the state sequence

$$P(S, O) = \prod_{t=1}^{T} P(s_t|s_{t-1}) P\,(o_t|s_t)$$

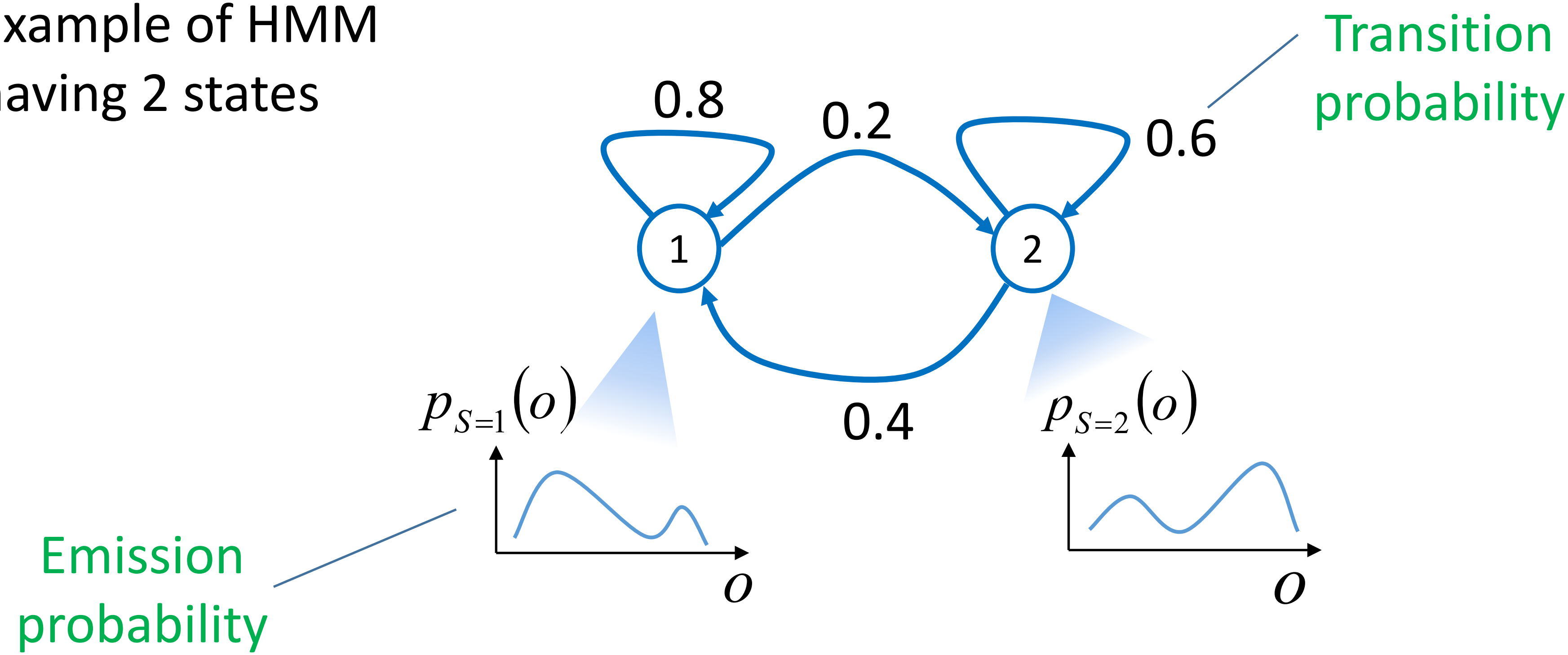
$$P(s_1|s_{-1}) \equiv P(s_1)$$

$$s_t \in \{1, 2, \cdots, K\}$$

$$P(O) = \sum_{S} P(S, O) = \sum_{S} \prod_{t=1}^{T} P(s_t|s_{t-1}) P\,(o_t|s_t)$$

Graph Representation of HMM

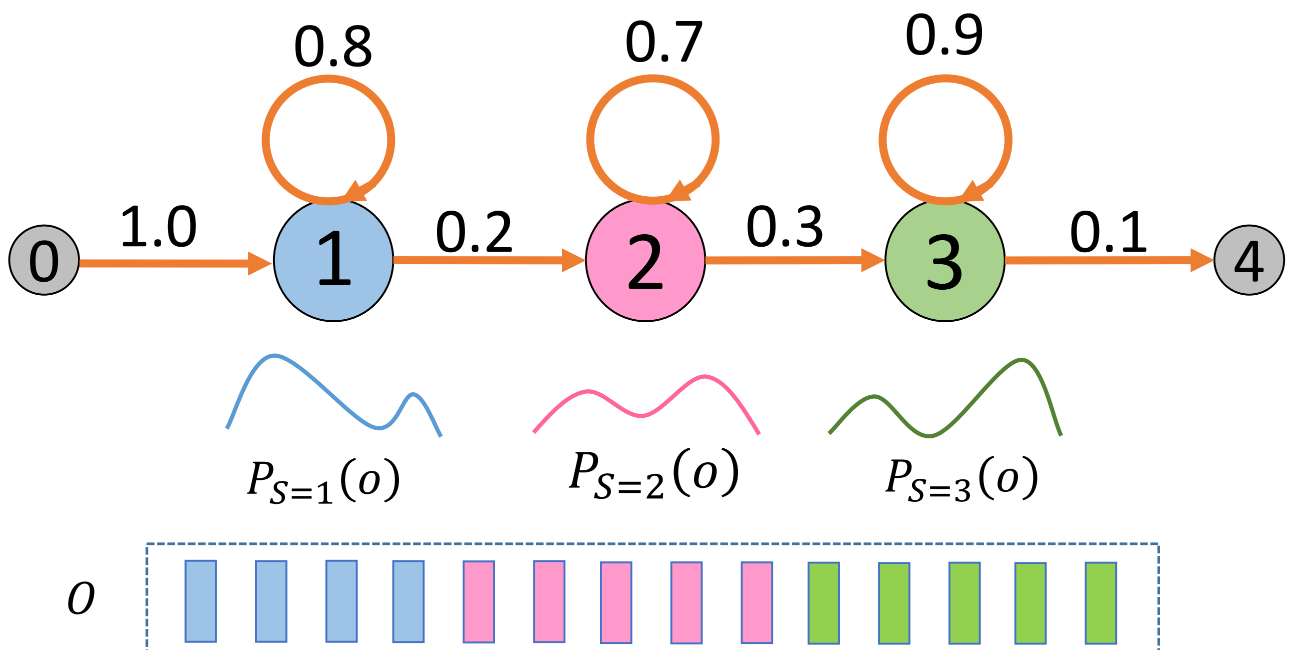
Example of HMM
having 2 states



Left-to-Right HMM

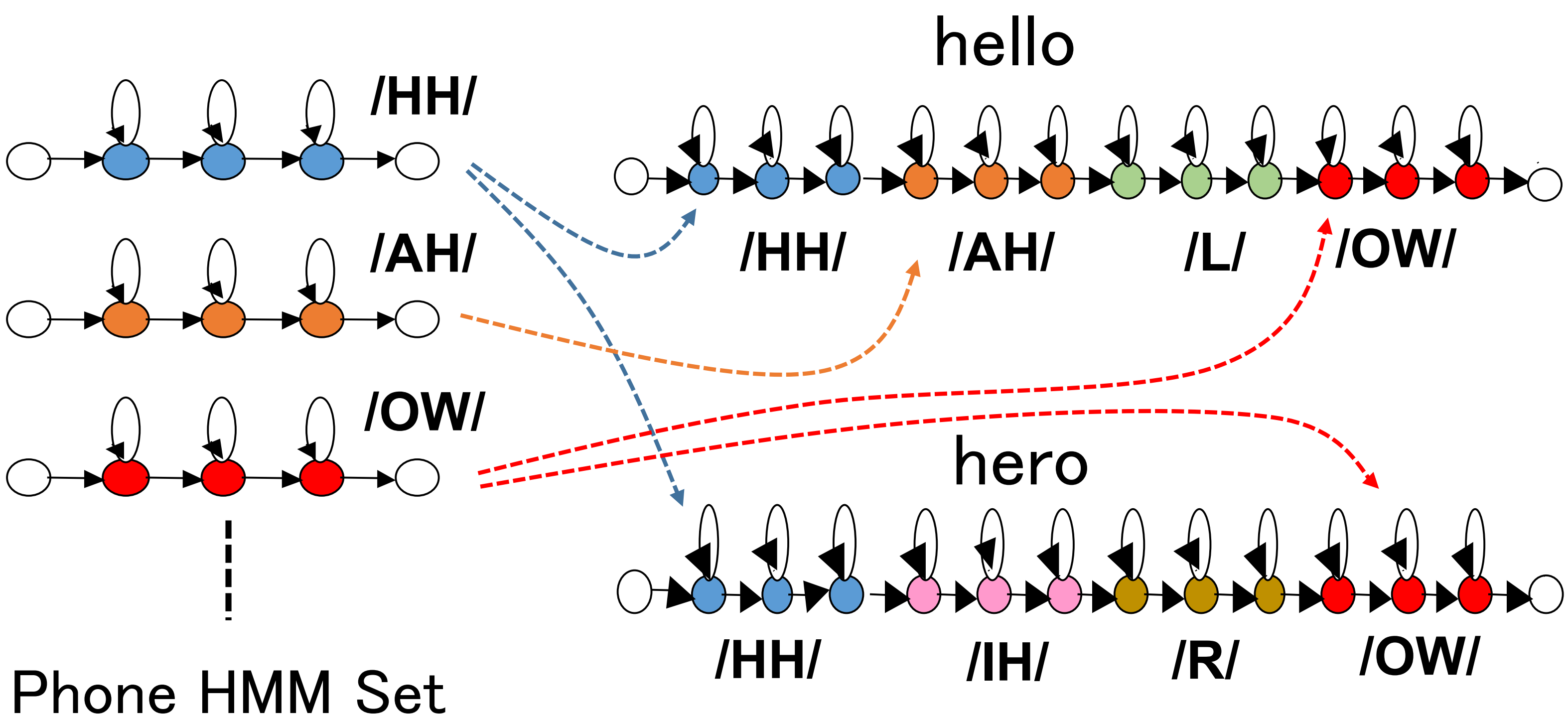
- Popular HMM design for acoustic modeling
  - Left-to-right transition structure
  - Non-emitting start and end states

Example of 3 state (K=3)
left-to-right HMM



Phone based Word/Utterance Modeling

- Since # of words/utterances is large, it is often not realistic to model each word/utterance by an HMM
- Phone based modeling composes arbitral word/utterance models by concatenating phone models



Example of Continuous Word Recognition

- HMM based acoustic model
- N-gram based language model

$$\hat{W} = \underset{W \in \{utterances\}}{\arg\max} \; P(W \mid O)$$

$$= \underset{W \in \{utterances\}}{\arg\max} \; HMM(O \mid \lambda_W)Ngram(W)$$

Phone based utterance HMM

N-gram model e.g. Tri-gram

Problem: How to Perform argmax?

- For continuous word recognition, # utterance is huge
  - E.g. If the vocabulary size V is 10000, and the utterance length L is 10, # utterances is $10000^{10} = 10^{40}$
- Enumerating all the utterances is impossible!

$$\hat{W} = \underset{W \in \{utterances\}}{\arg\max} \; HMM(O \mid \lambda_W) Ngram(W)$$

How to do this?

Apply dynamic programming

Exercise 2.1

Suppose W is a vowel and O is a MFCC feature vector. Suppose that $P_{AM}(O|W)$ is an acoustic model and $P_{LM}(W)$ is a language model. Obtain a vowel $\widehat{W}$ that maximizes $P(W|O)$ when the acoustic and language model log likelihoods are given as in the following table.

$$\hat{W} = \underset{W \in \{a, i, u, e, o\}}{\arg\max} \{P(W \mid O)\}$$

Vowel V	a	i	u	e	o
log(P(O|V))	-13.4	-10.5	-30.1	-15.2	-17.0
log(P(V))	-1.61	-2.30	-1.61	-1.39	-1.39

Exercise 2.2

- The following table defines a Bi-gram P(Word|Context)

C＼W	today	is	sunny	End
Start	0.6	0.1	0.2	0.1
today	0.1	0.5	0.3	0.1
is	0.1	0.1	0.7	0.1
sunny	0.1	0.1	0.2	0.6

*P(Start)=1.0

Example：

Start today is sunny End

1.0 0.6 0.5 0.7 0.6

1.0x0.6x0.5x0.7x0.6=0.126

Exercise 2.2 (Cont.)

- Based on the bigram definition of the previous slide, compute the probability of the following sentences

1) P("Start today sunny today sunny End")

=

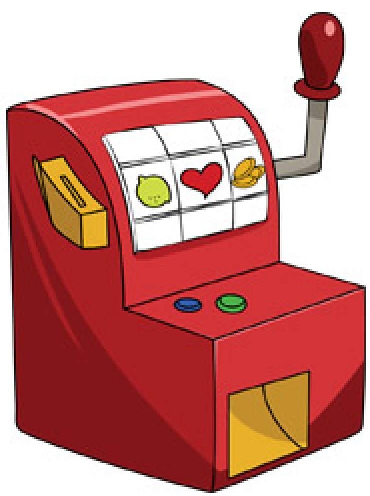
2) P("Start today today sunny sunny End")

=

Sampling Methods

Pseudo Random Generator

- On digital computer, everything is deterministically calculated and there is no randomness
- However, sometimes we want random numbers
- Most programing languages have a pseudo random generator function



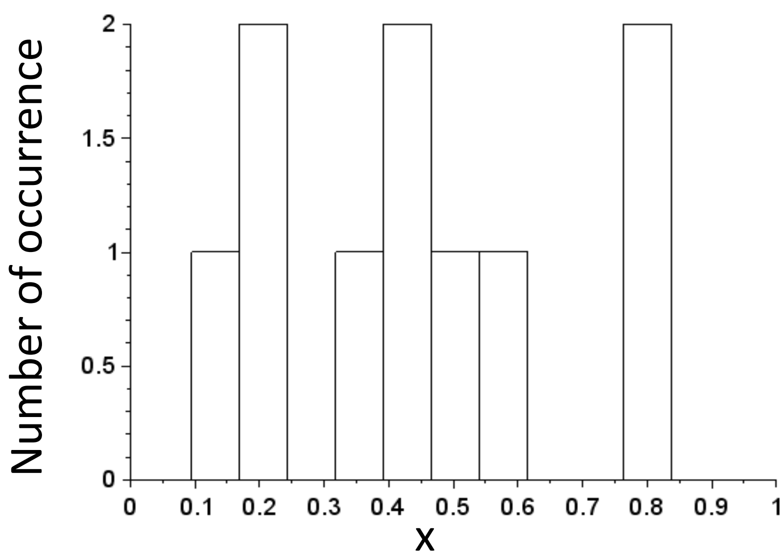
```
Python 2.6
> import random
> random.random()
0.89388901900395423
> random.random()
0.98563591571989639
> random.random()
0.53054443555684372
```

Sampling From a Uniform Distribution

- Random numbers distributed uniformly over some region

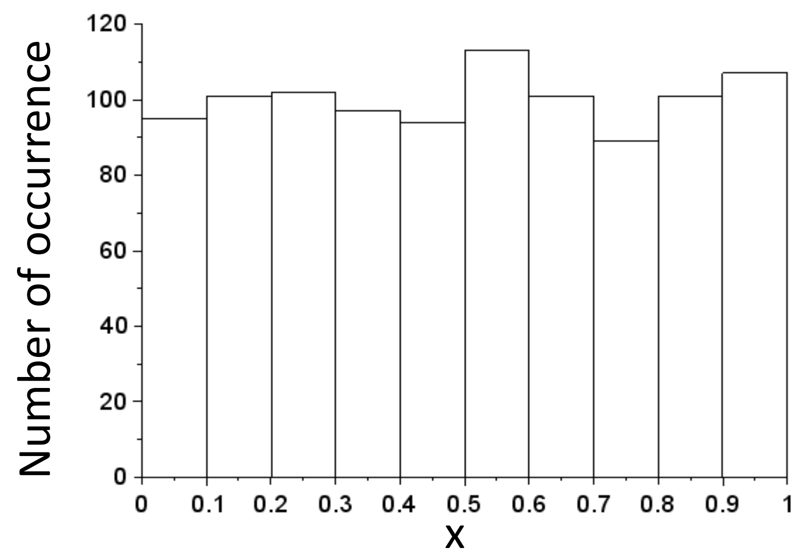
Example:
Histogram of samples obtained from a uniform distribution over (0, 1)
(To make the graph, scilab was used)

10 samples



histplot(10,rand(1:10), normalization=%f)

1000 samples



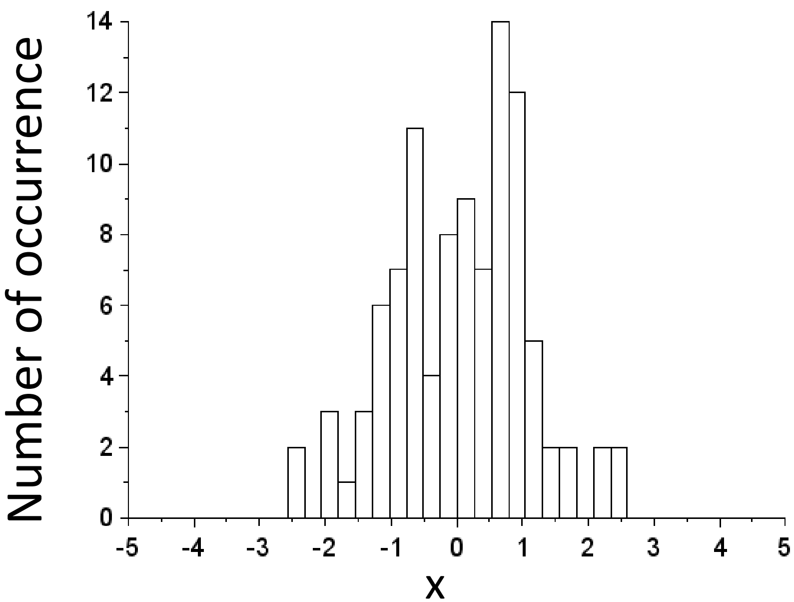
histplot(10,rand(1:1000), normalization=%f)

Sampling From a Gaussian Distribution

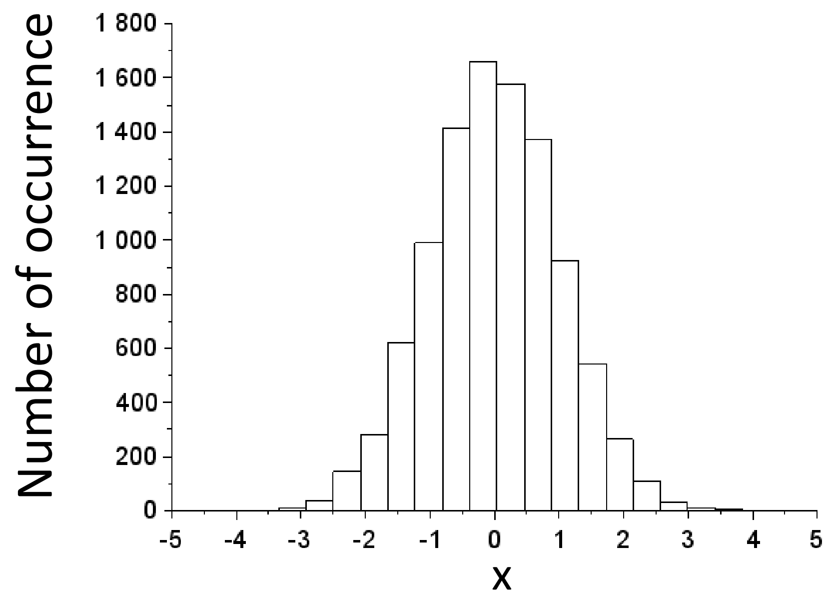
- Standard normal (Gaussian) distribution has a mean 0.0 and a variance 1.0

Example:

100 samples



10000 samples



Transform of Random Variable

- Let x be a random variable and f be a function $y = f(x)$. When x follows $p(x)$, y follows the following distribution $q(y)$

$$q(y) = p(x)\left|\frac{dx}{dy}\right|$$

Example

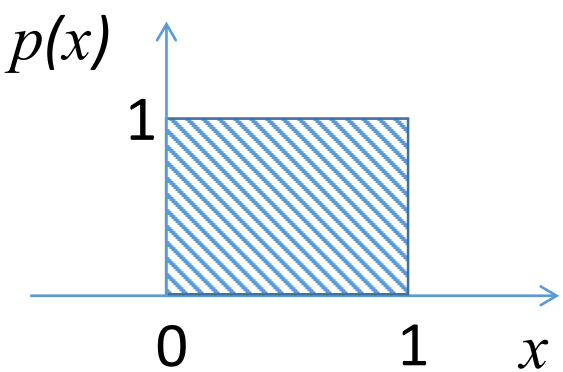
- When $p(x)$ and $y = f(x)$ are given as follows, obtain distribution $q(y)$

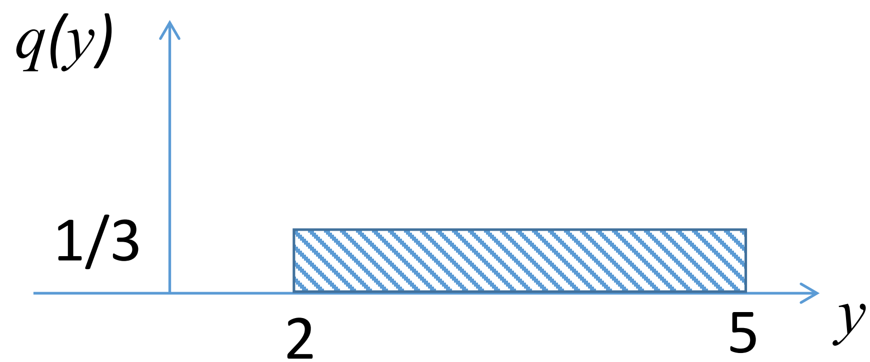
$$p(x) = 1 \quad x \in (0, 1)$$

$$y = 3x + 2$$

Answer

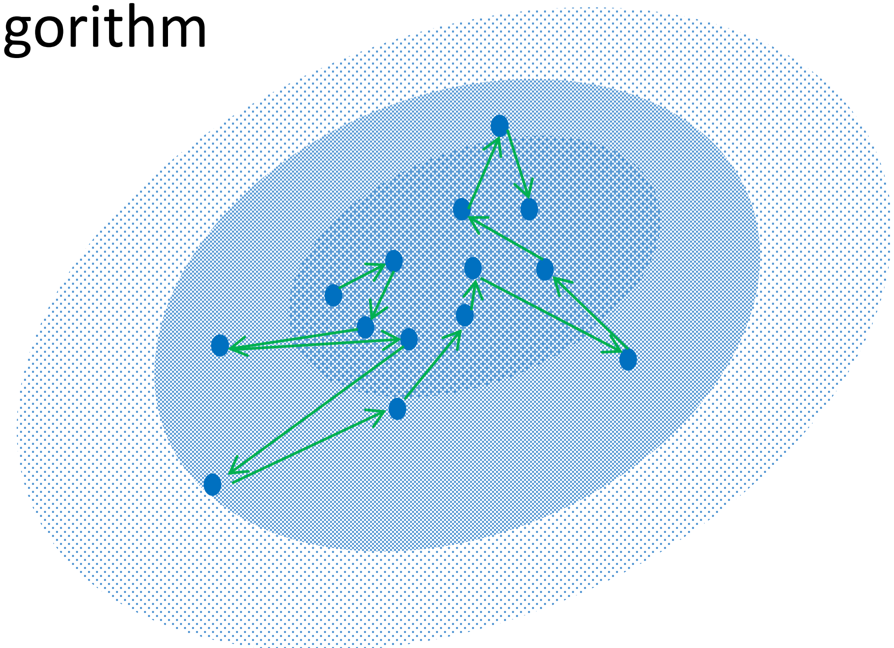
$$x = \frac{y-2}{3}, \qquad \left|\frac{dx}{dy}\right| = \frac{1}{3} \quad \Rightarrow \quad q(y) = \frac{1}{3} \quad y \in (2,\ 5)$$





Markov Chain Monte Carlo

- General and powerful framework for sampling
  - Scales well with the dimensionality of the sample space
- Maintains a state that forms a Markov chain. The set of the states follows the desired distribution
  - Metropolis algorithm
  - Metropolis-Hastings algorithm
  - Gibbs Sampling



Metropolis Algorithm

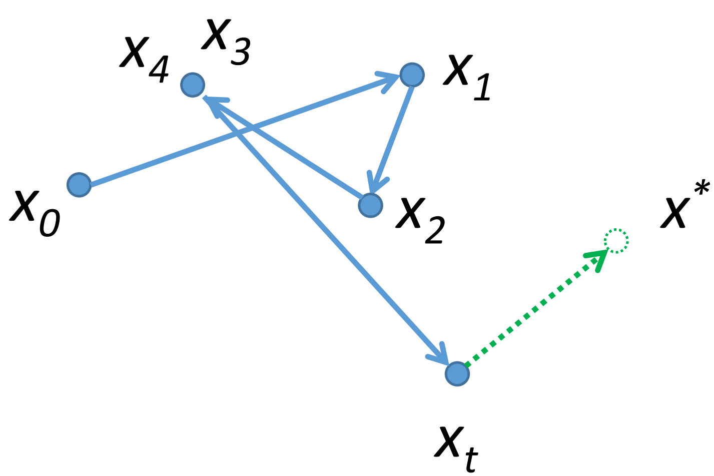
- Assumptions:
  - We want samples from a distribution $p(X)$

$$p(X) = \frac{1}{Z}\tilde{p}(X)$$

  - The normalization constant Z may be unknown
- Initialization:
  1. Prepare a symmetric proposal distribution $q(x_A|x_B)$ that satisfy $q(x_A|x_B)=q(x_B|x_A)$
  2. Prepare an initial state x_0

Procedure of Metropolis Algorithm

- Algorithm:
  1. Get a candidate sample x^* from the proposal distribution $q(x|x_t)$ based on the current state x_t
  2. Accept the candidate with probability $A(x^*, x_t) = \min\left(1, \frac{\widetilde{p}(x^*)}{\widetilde{p}(x_t)}\right)$ or discard it
  3. If the candidate is accepted, save it as the next state x_{t+1}. If it is discarded, then set x_{t+1} equals to x_t
  4. Goto step 3



Gibbs Sampling

- Problem:
  - We want samples from a joint distribution p(x_1, x_2,…, x_M)
- Algorithm:
  1. Prepare an initial state X_0 = <x_1,x_2,…,x_M>$_0$
  2. Select one of the variables x_i in order or at random
  3. Get a sample from $p(x_i \mid X \setminus i)$ and update x_i with that value
  4. Goto step 2. After enough iterations, the distribution of x_t follows p(X)

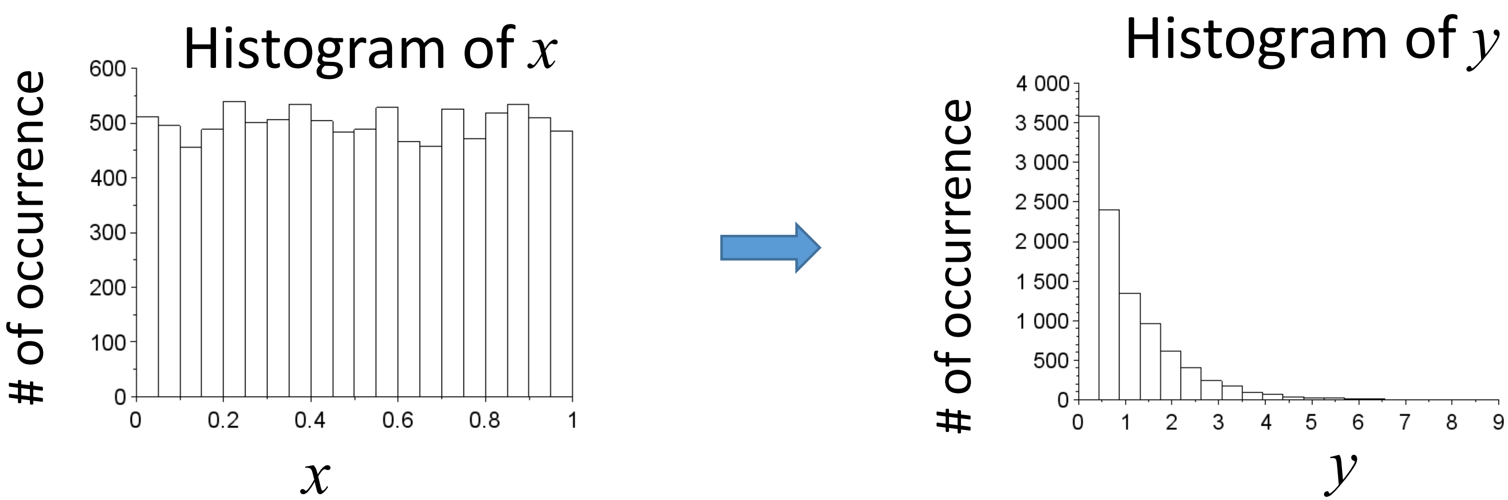
Compared to the Metropolis algorithm:

- Sampling from conditional distribution of x_i given all other variables is required
- There is no rejection step

Exercise 2.3

- When $p(x)$ and $y = f(x)$ are given as follows, obtain distribution $q(y)$

$$p(x) = 1 \quad x \in (0, 1), \qquad y = -\log(1 - x)$$



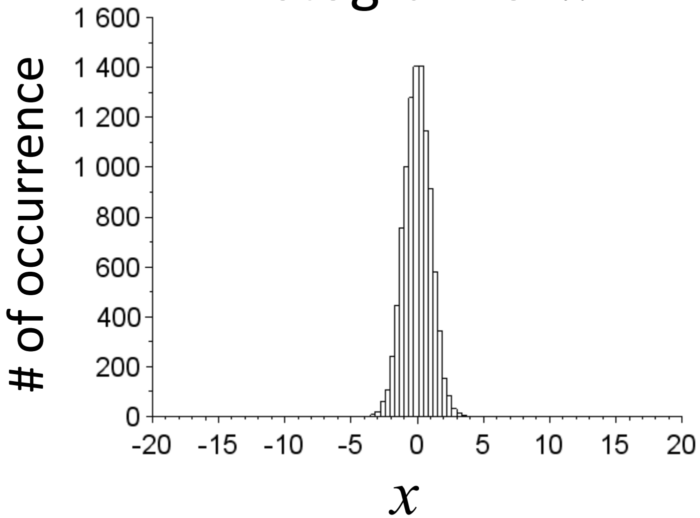
Exercise 2.4

- When $p(x)$ and $y = f(x)$ are given as follows, obtain distribution q(y)

$$p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2} \right) = N(x \mid 0,1) \quad x \in (-\infty, \infty), \qquad y = 3x + 4$$

The answer becomes a Gaussian distribution. Report its mean and variance.

Histogram of x



Histogram of y