

---

# Hidden Markov Model II

---

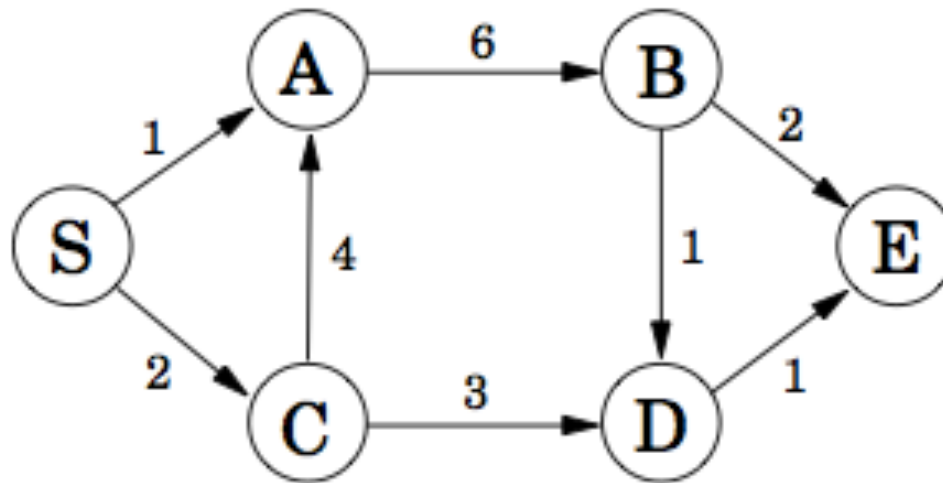
September 7, 2020

- HMM is used to model sequential data. Observed data are assumed to be “emitted” from hidden states, where the hidden states is a “Markov chain”.
- A HMM is characterized by initial/emission/transition probabilities.
- Difference between HMM and mixture model is the correlations between hidden states.
- The goals of HMM include (1) parameter estimation; (2) underlying states estimation; (3) determine the best path.
- Use EM with “forward-backward” algorithm for parameter estimation.
- We will cover **dynamic programming** and **Viterbi algorithm** in this lecture.

*“Two sledgehammers of the algorithms craft: dynamic programming and linear programming”*

- DP is a general optimization algorithm.
- Breaking the overall optimization problem into overlapping smaller problems.
- Solve each sub-problem once, and reuse the results, thus reducing the computing cost (dramatically).
- Often working backward.

Find the shortest path from S to E in the directed acyclic graph below.

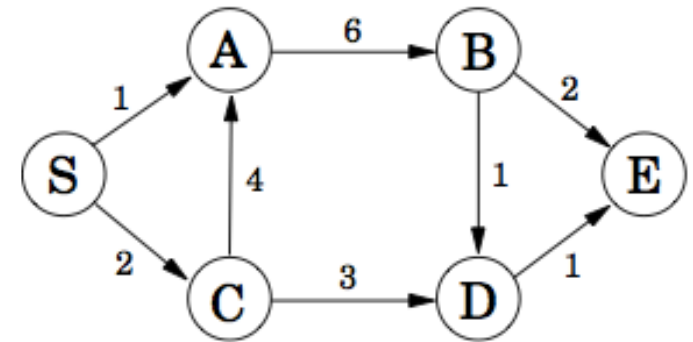


The problem can be solved backward. Take node D as an example. The way to get to D is through B or C. So,  $\text{dist}(D) = \min\{\text{dist}(B) + 1, \text{dist}(C) + 3\}$ .

Then work our way backward, we can find the best path.

Using exhaustive search , one has to do:

- SABE:  $1+6+2$
- SCABE:  $2+4+6+2$
- SCDE:  $2+3+1$
- SCABDE:  $2+4+6+1+1$



Total is 11 additions. The complexity grows exponentially with the size of graph

Using DP, do:

- $\text{Dist}(A) = \min(1, 2+4) = 1$
- $\text{Dist}(C) = 2$
- $\text{Dist}(B) = \text{dist}(A) + 6 = 1 + 6 = 7$
- $\text{Dist}(D) = \min(\text{dist}(B) + 1, \text{dist}(C) + 3) = \min(7 + 1, 2 + 3) = 5$
- $\text{Dist}(E) = \min(\text{dist}(B) + 2, \text{dist}(D) + 1) = \min(7 + 2, 5 + 1) = 6$

Total is 6 additions. The complexity grows linearly with the size of graph.

Under the notations:

- Observed data:  $\mathbf{u} = \{u_1, u_2, \dots, u_T\}$ .
- Hidden states:  $\mathbf{s} = \{s_1, s_2, \dots, s_T\}$
- Model parameters:  $\lambda = \{\pi_k, b_k(u), a_{k,l}\}$ .

We want to find the most possible “path”:  $\hat{\mathbf{s}} = \operatorname{argmax}_{\mathbf{s}} \Pr(\mathbf{s}|\lambda, \mathbf{u})$ . This is called the rule of *Maximum A Posteriori* (MAP) (mode of the posterior probability).

Since we have:

$$\Pr(\mathbf{s}|\lambda, \mathbf{u}) = \frac{\Pr(\mathbf{s}, \mathbf{u}|\lambda)}{\Pr(\mathbf{u}|\lambda)}$$

The denominator doesn't involve  $\mathbf{s}$ . So

$$\operatorname{argmax}_{\mathbf{s}} \Pr(\mathbf{s}|\lambda, \mathbf{u}) = \operatorname{argmax}_{\mathbf{s}} \Pr(\mathbf{s}, \mathbf{u}|\lambda)$$

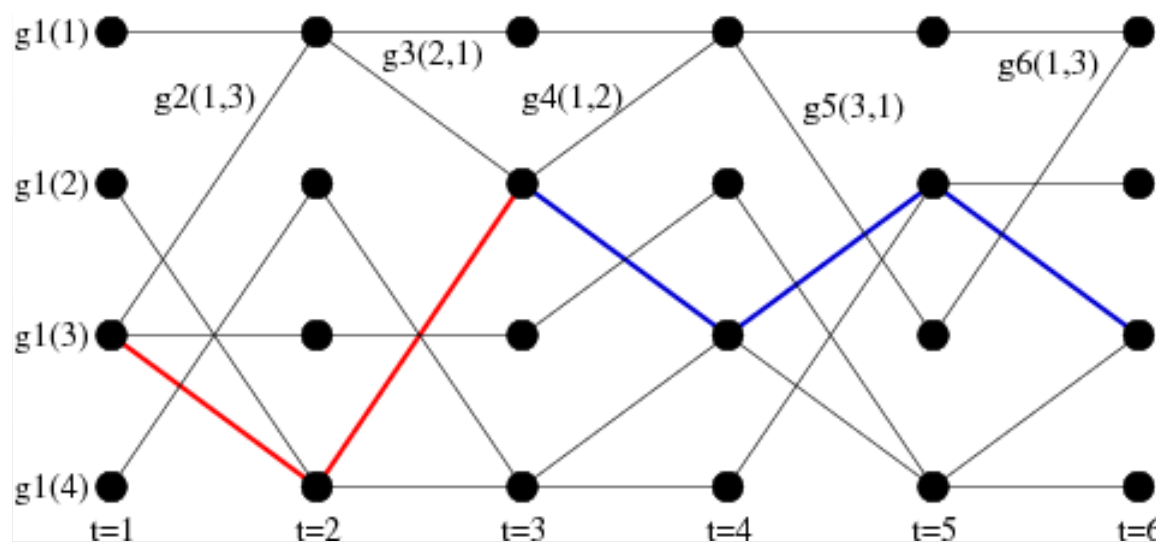
In other words, to maximize the conditional probability, we can simply maximize the joint probability.

- The Viterbi algorithm maximizes an objective function  $G(s)$ , where  $s = \{s_1, \dots, s_T\}$  is a sequence of categorical values:  $s_t \in \{1, \dots, M\}$ .
- $G(s)$  satisfies following special property:

$$G(s) = g_1(s_1) + g_2(s_2, s_1) + g_3(s_3, s_2) + \dots + g_T(s_T, s_{T-1}).$$

So the objective function can be expressed as sum of functions depending one state and its preceding one.

- In a diagram, let  $g_t(k, l)$  be the distance from state  $l$  at  $t - 1$  to state  $k$  at  $t$ . At the starting node, use  $g_1(k)$  for state  $k$ . The optimal path can be found through DP.



In a HMM, the distance between nodes are the transition probabilities. But we still need to consider emission probabilities.

Remember we want to find optimal sequence  $\mathbf{s}^*$ :

$$\mathbf{s}^* = \operatorname{argmax}_{\mathbf{s}} Pr(\mathbf{s}, \mathbf{u} | \lambda).$$

The objective function can be expressed as:

$$\begin{aligned} G(\mathbf{s}) &= \log Pr(\mathbf{s}, \mathbf{u} | \lambda) = \log[\pi_{s_1} b_{s_1}(u_1) a_{s_1, s_2} b_{s_2}(u_2) a_{s_2, s_3} \dots a_{s_{T-1}, s_T} b_{s_T}(u_T)] \\ &= [\log \pi_{s_1} + \log b_{s_1}(u_1)] + [\log a_{s_1, s_2} + \log b_{s_2}(u_2)] + \dots + [\log a_{s_{T-1}, s_T} + \log b_{s_T}(u_T)] \end{aligned}$$

If we define

$$\begin{aligned} g_1(s_1) &= \log \pi_{s_1} + \log b_{s_1}(u_1) \\ g_t(s_t, s_{t-1}) &= \log a_{s_{t-1}, s_t} + \log b_{s_t}(u_t) \end{aligned}$$

then  $G(\mathbf{s}) = g_1(s_1) + \sum_{t=2}^T g_t(s_t, s_{t-1})$ , and Viterbi algorithm can be applied.



- Notice that the Viterbi algorithm requires that the model parameters  $\lambda$  are known.
- “Viterbi training” algorithm can be applied to estimate  $\lambda$ . The steps are:
  1. Choose initial values of  $\lambda$ .
  2. Under current  $\lambda$ , find the optimal path  $s^*$ .
  3. Let  $L_k(t) = \mathbb{1}(s_t^* = k)$  and  $H_{k,l}(t) = \mathbb{1}(s_{t-1} = k)\mathbb{1}(s_t = l)$ , then update  $\lambda$  using the same M-step procedures derived before.
- Viterbi training replaces the step of computing forward and backward probabilities by finding the optimal path  $s^*$  under the current parameters using Viterbi algorithm.
- Basically, it uses “hard” classification (0/1) to replace the “soft” classification (probabilities).

- This is a model selection problem.
- Since the whole data likelihood  $P(u)$  is available, this can be done by using BIC/AIC.
- With one more state, there are more parameters from initial probability, transition probabilities, and emission probability.
- However, based on my experience, BIC tends to select large  $M$  in real data, especially when the chain is long.
- Sometimes have to use arbitrary criteria.

- So far we have discussed univariate HMM, e.g.,  $u_t$  is a scalar.
- When observation is a random vector, it can be modeled as a multivariate HMM.
- The emission probability  $b_k(u)$  becomes a multivariate distribution.
- There are more parameters need to be estimated, but the procedure is the same.

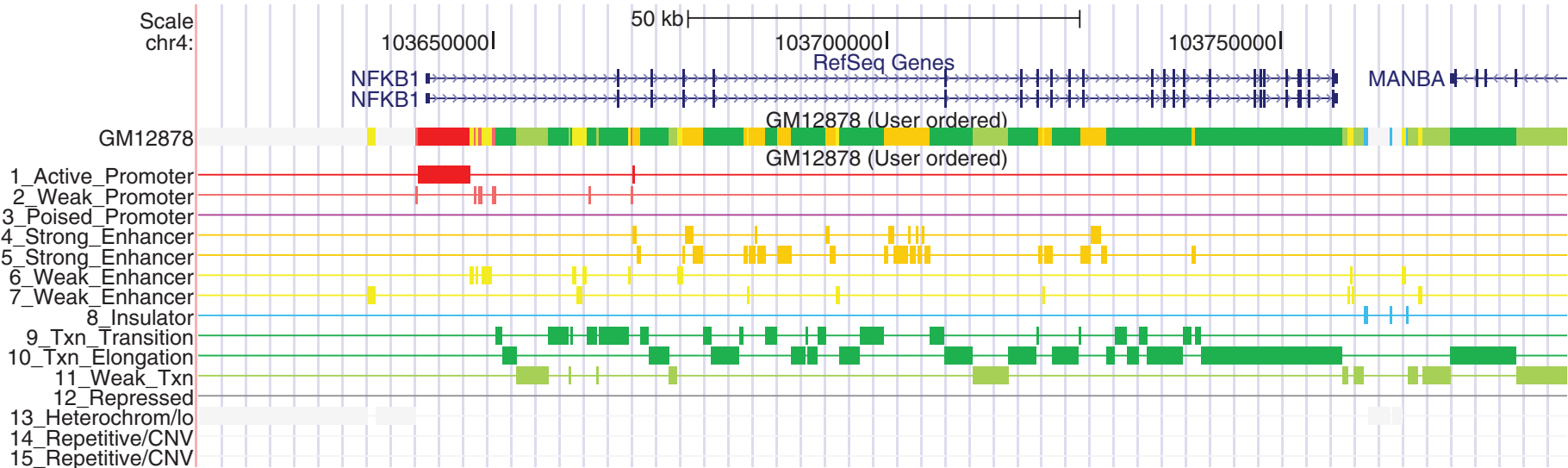
Ernst & Kellis, **Nature Method** 2012

---

## **ChromHMM: automating chromatin-state discovery and characterization**

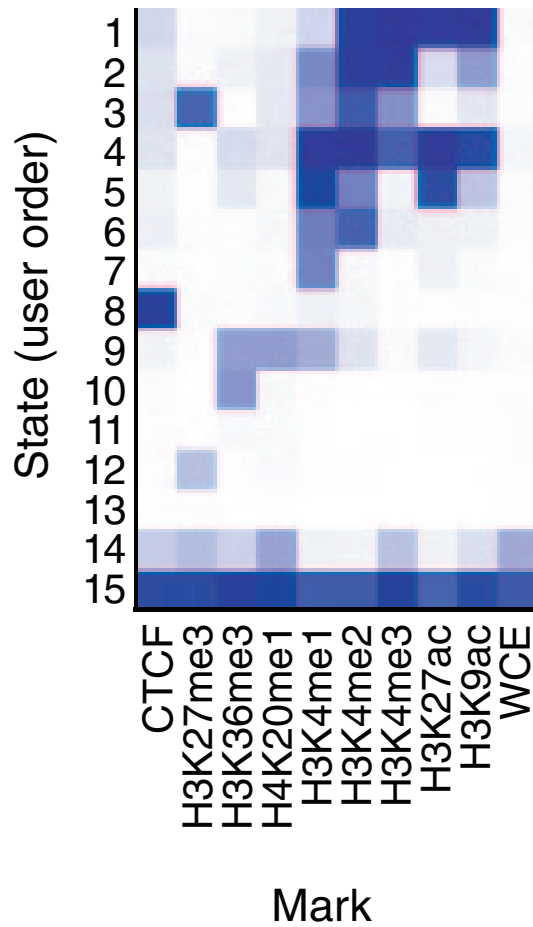
- The observed data are multiple ChIP-seq datasets profiling histone modification and protein binding strengths.
- The data are measurements from 200 bp bins genome-wide. There are around 10 million bins (chain is 10 million long).
- The goal is to segment the whole genome into a number of “states”.
- They run multivariate HMM, assuming the outcomes are independent.
- Result segments the genome into 15 states.

chromHMM genome segmentation result on a gene:

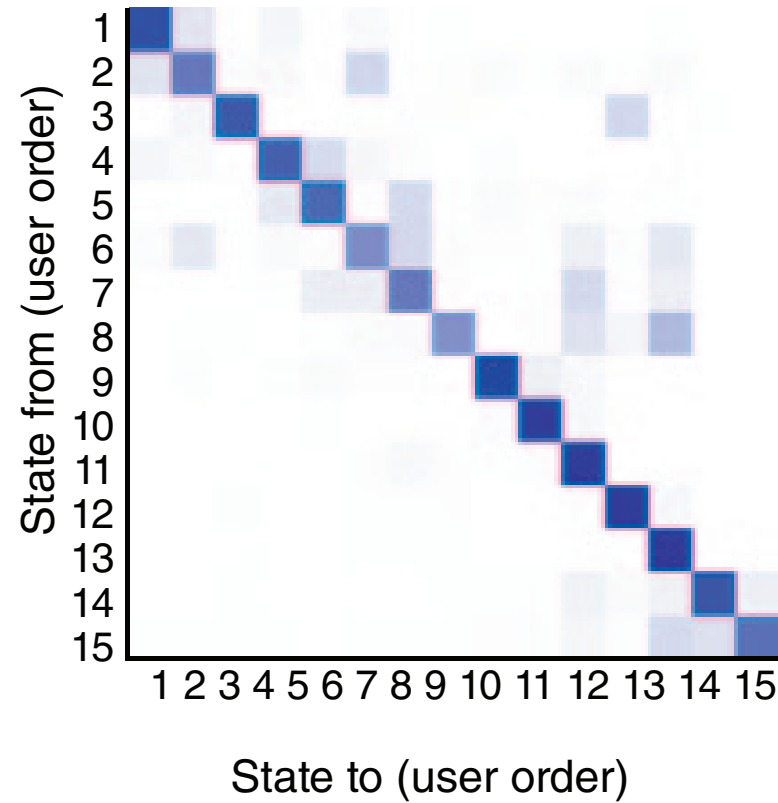


chromHMM emission and transition probabilities:

## Emission parameters



## Transition parameters

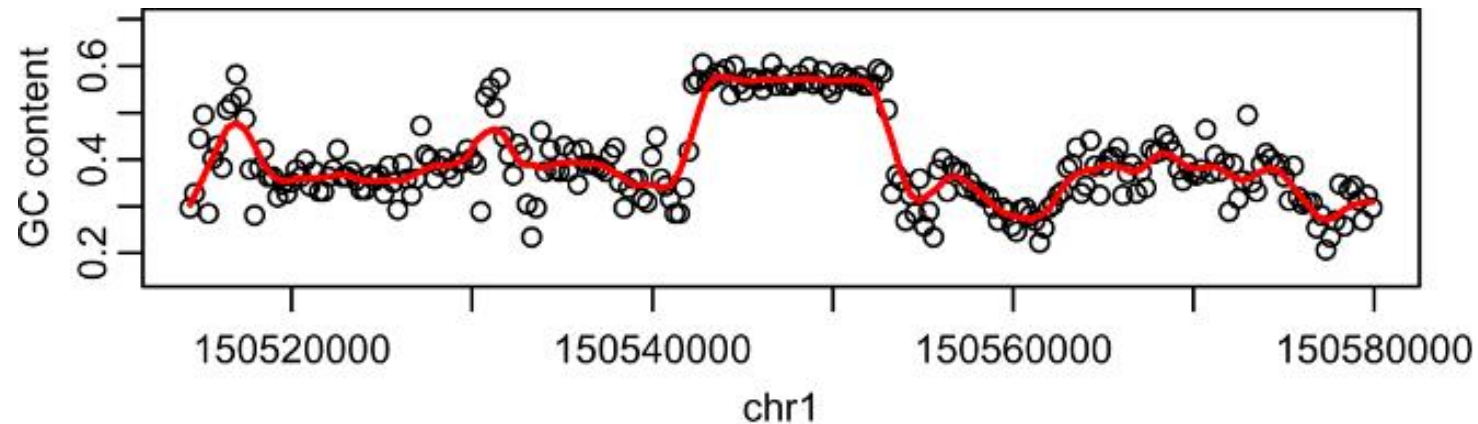


- The observations within a state have spatial correlations (remember HMM assumes observations are independent conditional on state).
- There are different ways to model the spatial correlations within a state, such as AR model or smoothing.
- Advantage over HMM: avoid over-fitting. The state segmentation of the chain is smoother.

- DNA sequence is a long (3 billion for human) string of four letters: A, C, G, T.
- The appearance of “CG” is rare, due DNA methylation, mutations and selections.
- However, there are regions where “CG” appears more frequently compared with overall. Such regions are called “CpG islands”, and they often mark important regions (such as gene promoters).
- The CpG islands can be detected by modeling the DNA sequence using HMM.
- Observed data are C+G content and CG appearance in 16bp bins.



GC content plot in a small region:



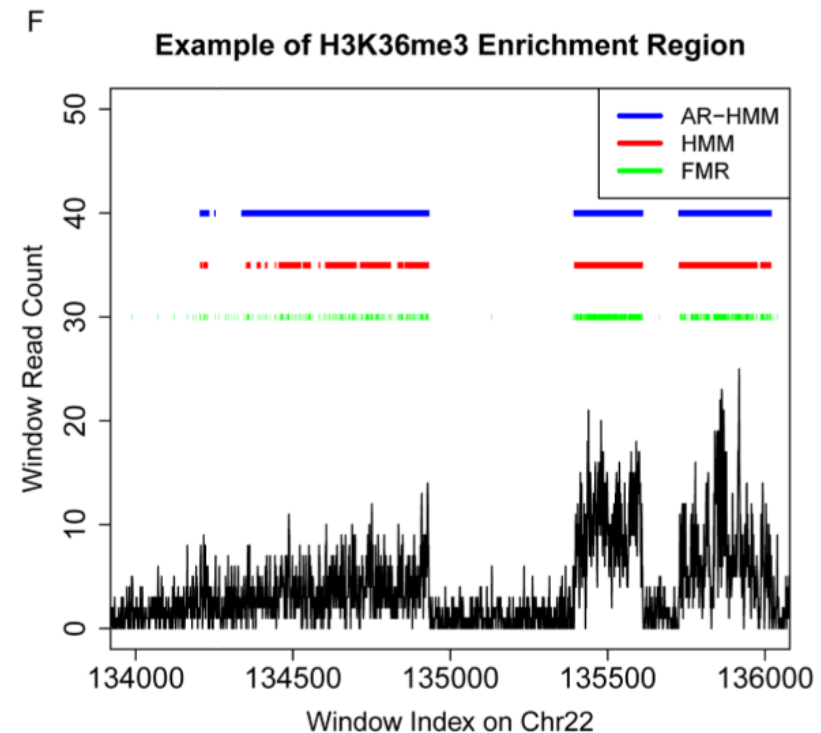
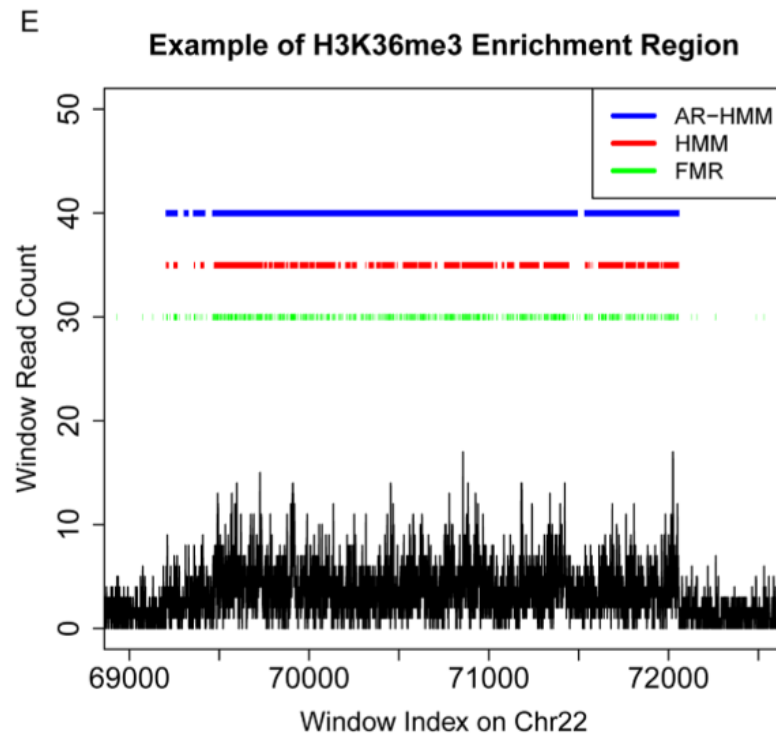
Our way to model this: the observation follows a smooth curve within each segmentation.

$$p(s) \mid s \in S_j \text{ and } X(M_j) = i \sim \text{Normal}\{c_i + f(s), \tau^2\},$$

For details, read Wu *et al.* (2010) *Redefining CpG islands using hidden Markov models*. Biostatistics.

Rashid *et al.* (2014) *Some Statistical Strategies for DAE-seq Data Analysis: Variable Selection and Modeling Dependencies among Observations*. JASA

- The goal is to detect long range histone modified regions.
- We want the regions to be long, but directly fitting a HMM often gives overly fragmented, short regions.
- Use AR model for the spatial dependence.



## Non-homogeneous HMM

Transition probability varies along the chain. Have to impose some constraints on the transition probabilities so that they can be estimated.

## Higher-order HMM

Assume the hidden states are from a higher order Markov chain, e.g., the current state depends on several previous states.

## Hierarchical HMM (HHMM)

Each state of the HHMM is itself an HHMM, e.g., the states of the HHMM emit sequences of observation that follows another HHMM.

## 2D HMM

Used in image segmentation. Inputs are 2D data emitted from a **Markov random field**. Need to model the transition from one observation to its neighbor. However a fully connected 2D HMM is NP-hard (computationally unsolvable). So different approximation is used (Pseudo 2D HMM).

- HMM is used to model sequential data. Observed data are assumed to be “emitted” from hidden states, where the hidden states is a “Markov chain”.
- A HMM is characterized by initial/emission/transition probabilities.
- Difference between HMM and mixture model is that HMM assumes correlations between hidden states, whereas mixture model assumes independence.
- The goals of HMM include (1) parameter estimation; (2) underlying states estimation; (3) determine the best path.
- Parameter estimation can be done by EM with “forward-backward” algorithm.
- Dynamic program (DP) is a general optimization method to find the shortest path in a directed acyclic graph.
- Viterbi algorithm is a DP algorithm applied in finding the optimal path for a HMM.
- Viterbi training is a simplified version of forward-backward algorithm. It uses hard classification to replace soft classification.