Latent/Missing Variables & Hidden Markov Models

Nicholas Ruozzi

University of Texas at Dallas

Unobserved Variables

- Latent or hidden variables in the model are never observed
  - We may or may not be interested in their values, but their existence is crucial to the model
- Some observations in a particular sample may be missing
  - Missing information on surveys or medical records (quite common)
  - We may need to model how the variables are missing

Missing Data

- Data can be missing from the model in many different ways
  - Missing completely at random: the probability that a data item is missing is independent of the observed data and the other missing data
  - Missing at random: the probability that a data item is missing can depend on the observed data
  - Missing not at random: the probability that a data item is missing can depend on the observed data and the other missing data

Modelling Missing Data

- Add additional binary variable m_i to the model for each possible observed variable x_i that indicates whether or not that variable is observed

$$p(x_{obs}, x_{mis}, m) = p(m|x_{obs}, x_{mis})p(x_{obs}, x_{mis})$$

Modelling Missing Data

- Add additional binary variable m_i to the model for each possible observed variable x_i that indicates whether or not that variable is observed

$$p(x_{obs}, x_{mis}, m) = \underbrace{p(m|x_{obs}, x_{mis})}\, p(x_{obs}, x_{mis})$$

Explicit model of the missing data
(missing not at random)

Modelling Missing Data

- Add additional binary variable m_i to the model for each possible observed variable x_i that indicates whether or not that variable is observed

$$p(x_{obs}, x_{mis}, m) = \underbrace{p(m|x_{obs})}_{\text{Missing at random}}p(x_{obs}, x_{mis})$$

Modelling Missing Data

- Add additional binary variable m_i to the model for each possible observed variable x_i that indicates whether or not that variable is observed

$$p(x_{obs}, x_{mis}, m) = \underbrace{p(m)}_{\text{Missing completely at random}} p(x_{obs}, x_{mis})$$

Modelling Missing Data

- Add additional binary variable m_i to the model for each possible observed variable x_i that indicates whether or not that variable is observed

$$p(x_{obs}, x_{mis}, m) = p(m)p(x_{obs}, x_{mis})$$

How can you model latent variables in this framework?

Learning with Missing Data

- In order to design learning algorithms for models with missing data, we will make two assumptions
  - The data is missing at random
  - The model parameters corresponding to the missing data (δ) are separate from the model parameters of the observed data (θ)
- That is

$$p(x_{obs}, m|\theta, \delta) = p(m|x_{obs}, \delta)p(x_{obs}|\theta)$$

- Derivation of the algorithm in this case then follows similarly to the previous discuss

Learning with Latent Variables

- Log-likelihood with latent variables:

$$\log l(\theta) = \sum_{i=1}^{N} \log p(x^{(i)}|\theta)$$
$$= \sum_{i=1}^{N} \log \sum_{y} p(x^{(i)}, y|\theta)$$

  - Again, this is typically not a concave function of θ
    - We will apply the same trick that we did with GMMs last lecture

Expectation Maximization

$$\log l(\theta) = \sum_{i=1}^{N} \log p(x^{(i)}|\theta)$$

$$= \sum_{i=1}^{N} \log \sum_{y} p(x^{(i)}, y|\theta)$$

$$= \sum_{i=1}^{N} \log \sum_{y} q_i(y) \cdot \frac{p\left(x^{(i)}, y\middle|\theta\right)}{q_{i(y)}}$$

$$\geq \sum_{i=1}^{N} \sum_{y} q_i(y) \log \frac{p\left(x^{(i)}, y\middle|\theta\right)}{q_{i(y)}}$$

Expectation Maximization

$$F(q,\theta) \equiv \sum_{i=1}^{N} \sum_{y} q_i(y) \log \frac{p\left(x^{(i)}, y \middle| \theta\right)}{q_i(y)}$$

- Maximizing F is equivalent to the maximizing the log-likelihood
- Maximize it using coordinate ascent

$$q^{t+1} = \arg \max_{q_1,\dots,q_K} F(q, \theta^t)$$

$$\theta^{t+1} = \operatorname*{argmax}_{\theta} F(q^{t+1}, \theta)$$

Expectation Maximization

$$\sum_{i=1}^{N} \sum_{y} q_i(y) \log \frac{p\left(x^{(i)}, y \middle| \theta^t\right)}{q_i(y)}$$

- Maximized when $q_i(y) = p\left(y \middle| x^{(i)}, \theta^t\right)$
- Can reformulate the EM algorithm as

$$\theta^{t+1} = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{N} \sum_{y} p(y|x^{(i)}, \theta^t) \log p\left(x^{(i)}, y \middle| \theta\right)$$

Latent Variable Models

- Many real-world models contain latent variables
- Because we will need to marginalize out over the latent variables in MLE, the presence of latent variables in the model can make performing MLE much harder
  - As before, we will make simplifying assumptions about the probability distribution of the latent variables

Markov Chains

- A Markov chain is a sequence of random variables $X_1, \dots, X_T \in S$ such that

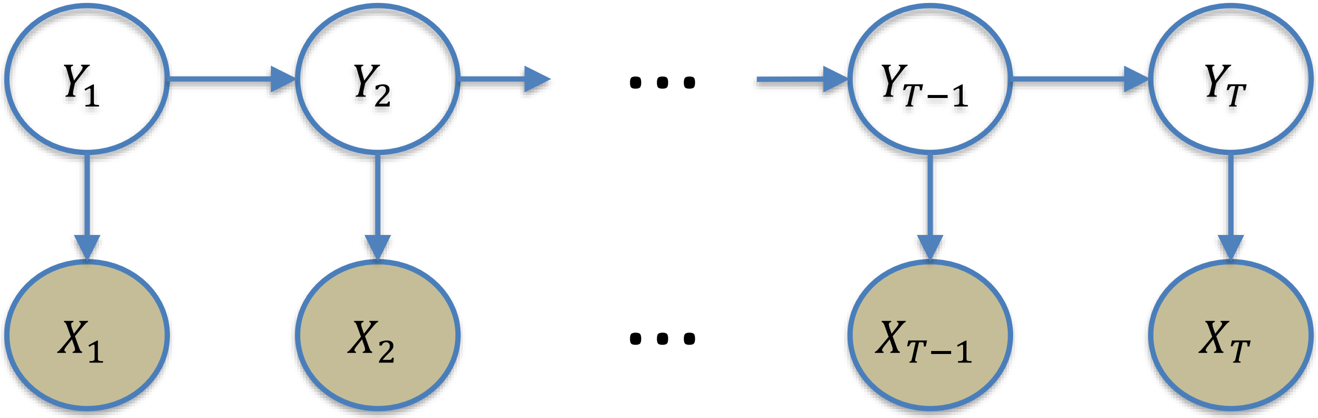
$$p(x_{t+1}|x_1, \dots, x_T) = p(x_{t+1}|x_t)$$

- The set S is called the state space, and $p(X_{t+1} = j|X_t = i)$ is the probability of transitioning from state i to state j at step t

Markov Chains

- When the probability of transitioning between two states does not depend on time, we call it a time homogeneous Markov chain
  - Represent it by a $|S| \times |S|$ transition matrix A
    - $A_{ij} = p(X_{t+1} = j | X_t = i)$
    - A is a **stochastic** matrix (all rows sum to one)

Hidden Markov Models



$$p(x_1, \dots, x_T, y_1, \dots, y_T) = p(y_1)p(x_1|y_1)\prod_t p(y_t|y_{t-1})p(x_t|y_t)$$

- X's are observed variables, Y's are latent/hidden
- Time homogenous: $p(y_t = j|y_{t-1} = i) = p(y_{t'} = j|y_{t'-1} = i)$
- For learning, we are given sequences of observations

Hidden Markov Models

- Well suited to problems/models that evolve over time
- Examples:
  - Observations correspond sizes of tree growth rings for one year, the latent variables correspond to average temperature
  - Observations correspond to noisy missile location, latent variables correspond to true missile locations

Learning HMMs

- A bit of notation:
  - $\pi_i = p(Y_1 = i)$
  - $A_{ij} = p(Y_t = j | Y_{t-1} = i)$
  - $b_j(x_t) = p(X_t = x_t | Y_t = j)$
- These parameters describe an HMM, $\theta = \{\pi, A, b\}$
  - We'll derive the updates in the case that the observations X_t are discrete random variables

Learning HMMs

$$\sum_{y} p(y|x,\theta^{s}) \log p(x,y|\theta) =$$

$$= \sum_{y} p(y|x,\theta^{s}) \log\left(p(y_1)p(x_1|y_1) \prod_{t=2}^{T} p(y_t|y_{t-1})p(x_t|y_t) \right)$$

$$= \sum_{y} p(y|x,\theta^{s}) \log\left(\pi_{y_1} b_{y_1}(x_1) \prod_{t=2}^{T} A_{y_t,y_{t-1}} b_{y_t}(x_t) \right)$$

$$= \sum_{y} p(y|x,\theta^{s}) \log \pi_{y_1} + \sum_{y} p(y|x,\theta^{s}) \left(\sum_{t=1}^{T} \log b_{y_t}(x_t) \right) + \sum_{y} p(y|x,\theta^{s}) \left(\sum_{t=2}^{T} \log A_{y_t,y_{t-1}} \right)$$

$$= \sum_{i} p(Y_1 = i|x,\theta^{s}) \log \pi_i + \sum_{t=1}^{T} \sum_{i} p(Y_t = i|x,\theta^{s}) \log b_i(x_t) + \sum_{t=2}^{T} \sum_{i} \sum_{j} p(Y_t = i, Y_{t-1} = j|x,\theta^{s}) \log A_{i,j}$$

Learning HMMs

$$p(y|x,\theta^s) = \pi_{y_1}^{s-1} b_{y_1}^{s-1}(x_1) \prod_{t=2}^{T} A_{y_t,y_{t-1}}^{s-1} b_{y_t}^{s-1}(x_t)$$

$$\pi_i^s = p(Y_1 = i|x,\theta^s)$$

$$b_i^s(k) = \frac{\sum_{t=1}^{T} p(Y_t = i|x,\theta^s)\delta(x_t = k)}{\sum_{t=1}^{T} p(Y_t = i|x,\theta^s)}$$

$$A_{ij}^s = \frac{\sum_{t=2}^{T} p(Y_t = i, Y_{t-1} = j|x,\theta^s)}{\sum_{t=2}^{T} p(Y_{t-1} = j|x,\theta^s)}$$

Prediction in HMMs

- Once we learn the model, given a new sequence of observations, $x_1, \ldots, x_T$, we want to predict y_T
  - In the tree application, this corresponds to finding the temperature at a specific time given the rings of a tree
  - In the missile tracking example, this corresponds to finding the position of the missile at a particular time
- Want to compute $p(y_T|x, \theta)$

Prediction in HMMs

- Want to compute $p(y_T|x,\theta) = p(x, y_T|\theta)/p(x|\theta)$
  - Direct approach:

$$p(x, Y_T = i|\theta) = \sum_{y_1,\dots,y_{T-1}} p(x, y_1, \dots, y_{T-1}, Y_T = i|\theta)$$

  - Dynamic programming approach:

$$\begin{aligned} p(x, Y_T = i|\theta) &= \sum_j p(x, Y_T = i, Y_{T-1} = j) \\ &= \sum_j p(x_1, \dots, x_{T-1}, Y_{T-1} = j) p(x_T, Y_T = i|x_1, \dots, x_{T-1}, Y_{T-1} = j) \\ &= \sum_j p(x_1, \dots, x_{T-1}, Y_{T-1} = j) p(x_T|Y_T = i) p(Y_T = i|Y_{T-1} = j) \end{aligned}$$

Prediction in HMMs

- Want to compute $p(y_T|x,\theta) = p(x,y_T|\theta)/p(x)$
  - Direct approach:

$$p(x, Y_T = i|\theta) = \sum_{y_1,\dots,y_{T-1}} p(x, y_1, \dots, y_{T-1}, Y_T = i|\theta)$$

  - Dynamic programming approach:

Called **filtering**: easy to implement using dynamic programming

$$p(x, Y_T = i|\theta) = \sum_j p(x, Y_T = i, Y_{T-1} = j)$$

$$= \sum_j p(x_1, \dots, x_{T-1}, Y_{T-1} = j) p(x_T, Y_T = i|x_1, \dots, x_{T-1}, Y_{T-1} = j)$$

$$= \sum_j p(x_1, \dots, x_{T-1}, Y_{T-1} = j) p(x_T|Y_T = i) p(Y_T = i|Y_{T-1} = j)$$

Latent Variables & EM

- Previous updates derived for a single observation (to simplify)
  - Can get the general updates for multiple sequences by adding sums in the appropriate places
- Same principle as EM for mixture models
  - Also suffers from the existence of lots of local optima