Bioinformatics: Biology X

Bud Mishra

Room 1002, 715 Broadway, Courant Institute, NYU, New York, USA

Model Building/Checking, Reverse Engineering, Causality

Outline

“Where (or of what) one cannot speak, one must pass over in silence.”
–Ludwig Wittgenstein, *Tractatus Logico-Philosophicus*, 1921.

Summary of the lecture / discussion points

1

Outline

Conditional Probabilities

- Suppose that A_1 and A_2 are two events such that $P(A_2) \neq 0$. Then the conditional probability that the event A_1 occurs, given that event A_2 occurs, denoted by $P(A_1|A_2)$ is given by the formula

$$P(A_1|A_2) = \frac{P(A_1 \& A_2)}{P(A_2)}.$$

Bayes Rule

- Suppose that A_1 and A_2 are two events such that $P(A_1) \neq 0$ and $P(A_2) \neq 0$. Then

$$P(A_2|A_1) = \frac{P(A_2)P(A_1|A_2)}{P(A_1)}.$$

Markov Models

- Suppose there are n states $S_1, S_2, \ldots, S_n$. And the probability of moving to a state S_j from a state S_i depends only on S_i, but not the previous history. That is:

$$
\begin{aligned}
& P(s(t+1) = S_j | s(t) = S_i, s(t-1) = S_{i_1}, \ldots) \\
= \;& P(s(t+1) = S_j | s(t) = S_i).
\end{aligned}
$$

Then by Bayes rule:

$$
\begin{aligned}
& P(s(0) = S_{i_0}, s(1) = S_{i_1}, \ldots, s(t-1) = S_{i_{t-1}}, s(t) = S_{i_t}) \\
= \;& P(s(0) = S_{i_0}) P(S_{i_1} | S_{i_0}) \cdots P(S_{i_t} | S_{i_{t-1}}).
\end{aligned}
$$

HMM: Hidden Markov Models

Defined with respect to an **alphabet** Σ

- A set of (hidden) **states** Q,
- A $|Q| \times |Q|$ matrix of **state transition probabilities** $A = (a_{kl})$, and
- A $|Q| \times |\Sigma|$ matrix of **emission probabilities** $E = (e_k(\sigma))$.

States

Q is a set of states that emit symbols from the alphabet Σ. Dynamics is determined by a state-space trajectory determined by the state-transition probabilities.

A Path in the HMM

- Path $\Pi = \pi_1 \pi_2 \cdots \pi_n$ = a sequence of states $\in Q^*$ in the hidden markov model, M.
- $x \in \Sigma^* =$ sequence generated by the path Π determined by the model M:

$$P(x|\Pi) = P(\pi_1) \left[\prod_{i=1}^{n} P(x_i|\pi_i) \cdot P(\pi_i|\pi_{i+1}) \right]$$

A Path in the HMM

- Note that

$$
\begin{aligned}
P(x|\Pi) &= P(\pi_1)\left[\prod_{i=1}^{n} P(x_i|\pi_i)\cdot P(\pi_i|\pi_{i+1})\right] \\
P(x_i|\pi_i) &= e_{\pi_i}(x_i) \\
P(\pi_i|\pi_{i+1}) &= a_{\pi_i,\pi_{i+1}}
\end{aligned}
$$

- Let π_0 and π_{n+1} be the initial ("begin") and final ("end") states, respectively

$$P(x|\Pi) = a_{\pi_0,\pi_1} e_{\pi_1}(x_1) a_{\pi_1,\pi_2} e_{\pi_2}(x_2) \cdots e_{\pi_n}(x_n) a_{\pi_n,\pi_{n+1}}$$

i.e.

$$P(x|\Pi) = a_{\pi_0,\pi_1} \prod_{i=1}^{n} e_{\pi_i}(x_i) a_{\pi_i,\pi_{i+1}}.$$

Decoding Problem

- For a given sequence x, and a given path π, the model (Markovian) defines the probability $P(x|\Pi)$
- In a casino scenario: the dealer knows Π and x, the player knows x but not Π.
- "The path of x is hidden."
- **Decoding Problem**: Find an optimal path π^* for x such that $P(x|\pi)$ is maximized.

$$\begin{aligned}\pi^* &= \arg\max_{\pi} P(\pi|x). \\ &= \arg\max_{\pi} P(x|\pi)P(\pi)/P(x).\end{aligned}$$

Assume uniform non-infromative priors for $P(x)$ and $P(\pi)$. Then, we can optimize the following:

$$\pi^* = \arg\max_{\pi} P(x|\pi).$$

Dynamic Programming Approach

Principle of Optimality

Optimal path for the $(i+1)$-prefix of x

$$x_1 x_2 \cdots x_{i+1}$$

uses a path for an i-prefix of x that is optimal among the paths ending in an unknown state $\pi_i = k \in Q$.

Dynamic Programming Approach

Recurrence: $s_k(i) =$ the probability of the most probable path for the i-prefix ending in state k

$$\forall_{k \in Q} \forall_{1 \leq i \leq n} \qquad s_k(i) = e_k(x_i) \cdot \max_{l \in Q} s_l(i-1) a_{lk}.$$

Dynamic Programming

- $i = 0$, Base case

$$s_{begin}(0) = 1, s_k(0) = 0, \forall_{k \neq begin}.$$

- $0 < i \leq n$, Inductive case

$$s_l(i+1) = e_l(x_{i+1}) \cdot \max_{k \in Q}[s_k(i) \cdot a_{kl}]$$

- $i = n + 1$

$$P(x|\pi^*) = \max_{k \in Q} s_k(n) a_{k,end}.$$

Viterbi Algorithm

- Dynamic Programing with "**log-score**" function

$$S_l(i) = \log s_l(i).$$

- Space Complexity = $O(n|Q|)$.
- Time Complexity = $O(n|Q|)$.
- Additive formula:

$$S_l(i+1) = \log e_l(x_{i+1}) + \max_{k \in Q}[S_k(i) + \log a_{kl}].$$

Bayesian Interpretation

- Probability $P(e) \mapsto$ our certainty about whether event e is true or false in the real world. (Given whatever information we have available.)
- "**Degree of Belief.**"
- More rigorously, we should write

Conditional probability $P(e|L) \mapsto$ *Represents a degree of belief with respect to* L *— The background information upon which our belief is based.*

Probability as a Dynamic Entity

- We update the “degree of belief” as more data arrives: using **Bayes Theorem**:

$$P(e|D) = \frac{P(D|e)P(e)}{P(D)}.$$

  Posterior is proportional to the prior in a manner that depends on the data $P(D|e)/P(D)$.
- **Prior Probability**: $P(e)$ is one’s belief in the event e before any data is observed.
- **Posterior Probability**: $P(e|D)$ is one’s updated belief in e given the observed data.
- **Likelihood**: $P(D|e) \mapsto$ Probability of the data under the assumption e

Dynamics

- **Note:**

$$P(e|D_1, D_2) = \frac{P(D_2|D_1, e)P(e|D_1)}{P(D_2|D_1)} = \frac{P(D_2|D_1, e)P(D_1|e)P(e)}{P(D_2 D_1)}$$

- **Further, note:** *The effects of prior diminish as the number of data points increase.*
- **The Law of Large Number:**

  With large number of data points, Bayesian and frequentist viewpoints become indistinguishable.

Parameter Estimation

- Functional form for a model M
  1. Model depends on some parameters Θ
  2. What is the best estimation of Θ?
- Typically the parameters Θ are a set of real-valued numbers
- Both prior $P(\Theta)$ and posterior $P(\Theta|D)$ are defining probability density functions.

MAP Method: Maximum A Posteriori

- Find the set of parameters Θ
  1. Maximizing the posterior $P(\Theta|D)$ or minimizing a score $-\log P(\Theta|D)$

$$\begin{aligned} E'(\Theta) &= -\log P(\Theta|D) \\ &= -\log P(D|\Theta) - \log P(\Theta) + \log P(D) \end{aligned}$$

  2. Same as minimizing

$$E(\Theta) = -\log P(D|\Theta) - \log P(\Theta)$$

  3. If prior $P(\Theta)$ is uniform over the entire parameter space (i.e., uninformative)

$$\min \arg_{\Theta} E_L(\Theta) = -\log P(D|\Theta).$$

Maximum Likelihood Solution

Outline

Information theory

- Information theory is based on probability theory (and statistics).
- **Basic concepts**: *Entropy* (the information in a random variable) and *Mutual Information* (the amount of information in common between two random variables).
- The most common unit of information is the **bit** (based log 2). Other units include the **nat**, and the **hartley**.

Entropy

- The entropy H of a discrete random variable X is a measure of the amount uncertainty associated with the value X.
- Suppose one transmits 1000 bits (0s and 1s). If these bits are known ahead of transmission (to be a certain value with absolute probability), logic dictates that no information has been transmitted. If, however, each is equally and independently likely to be 0 or 1, 1000 bits (in the information theoretic sense) have been transmitted.

Entropy

- Between these two extremes, information can be quantified as follows.
- If **X** is the set of all messages x that X could be, and $p(x)$ is the probability of X given x, then the **entropy of** X is defined as

$$H(x) = E_X[I(x)] = -\sum_{x \in X} p(x) \log p(x).$$

Here, $I(x)$ is the self-information, which is the entropy contribution of an individual message, and E_X is the expected value.

- An important property of entropy is that it is maximized when all the messages in the message space are equiprobable $p(x) = 1/n$, i.e., most unpredictable, in which case $H(X) = \log n$.
- The binary entropy function (for a random variable with two outcomes $\in \{0, 1\}$ or $\in \{H, T\}$:

$$H_b(p, q) = -p \log p - q \log q, \quad p + q = 1.$$

Joint entropy

- The joint entropy of two discrete random variables X and Y is merely the entropy of their pairing: $\langle X, Y \rangle$.
- Thus, if X and Y are independent, then their joint entropy is the sum of their individual entropies.

$$H(X, Y) = E_{X,Y}[-\log p(x,y)] = -\sum_{x,y} p(x,y) \log p(x,y).$$

- For example, if (X,Y) represents the position of a chess piece — X the row and Y the column, then the joint entropy of the row of the piece and the column of the piece will be the entropy of the position of the piece.

Conditional Entropy or Equivocation

- The conditional entropy or conditional uncertainty of X given random variable Y (also called the equivocation of X about Y) is the average conditional entropy over Y:

$$\begin{aligned} H(X|Y) &= E_Y[H(X|y)] \\ &= -\sum_{y \in Y} p(y) \sum_{x \in X} p(x|y) \log p(x|y) \\ &= -\sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(y)} \end{aligned}$$

- A basic property of this form of conditional entropy is that:

$$H(X|Y) = H(X,Y) - H(Y).$$

Mutual Information (Transinformation)

- Mutual information measures the amount of information that can be obtained about one random variable by observing another.
- The mutual information of X relative to Y is given by:

$$I(X;Y) = E_{X,Y}[SI(x,y)] = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}.$$

where SI (**Specific mutual Information**) is the pointwise mutual information.

- A basic property of the mutual information is that

  $$I(X;Y) = H(X) - H(X|Y) = H(X) + H(Y) - H(X,Y) = I(Y;X).$$

  That is, knowing Y, we can save an average of $I(X;Y)$ bits in encoding X compared to not knowing Y. Note that mutual information is **symmetric**.
- It is important in communication where it can be used to maximize the amount of information shared between sent and received signals.

Kullback-Leibler Divergence (Information Gain)

- The Kullback-Leibler divergence (or information divergence, information gain, or relative entropy) is a way of comparing two distributions: a “true” probability distribution $p(X)$, and an arbitrary probability distribution $q(X)$.

$$\begin{aligned} D_{KL}(p(X)\|q(X)) &= \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)} \\ &= \sum_{x \in X} [-p(x) \log q(x)] - [-p(x) \log p(x)] \end{aligned}$$

- If we compress data in a manner that assumes $q(X)$ is the distribution underlying some data, when, in reality, $p(X)$ is the correct distribution, the Kullback-Leibler divergence is the number of average additional bits per datum necessary for compression.
- Although it is sometimes used as a ‘distance metric,’ it is not a true metric since it is not symmetric and does not satisfy the triangle inequality (making it a semi-quasimetric).

- Mutual information can be expressed as the average Kullback-Leibler divergence (information gain) of the posterior probability distribution of X given the value of Y to the prior distribution on X:

$$\begin{aligned} I(X;Y) &= E_{p(Y)}[D_{KL}(p(X|Y=y)\|p(X)] \\ &= D_{KL}(p(X,Y)\|p(X)p(Y)). \end{aligned}$$

  In other words, mutual information $I(X,Y)$ is a measure of how much, on the average, the probability distribution on X will change if we are given the value of Y. This is often recalculated as the divergence from the product of the marginal distributions to the actual joint distribution.

- Mutual information is closely related to the log-likelihood ratio test in the context of contingency tables and the multinomial distribution and to Pearson's χ^2 test.

Source theory

- Any process that generates successive messages can be considered a source of information.
- A memoryless source is one in which each message is an independent identically-distributed random variable, whereas the properties of ergodicity and stationarity impose more general constraints. All such sources are stochastic.

Information Rate

- **Rate** Information rate is the average entropy per symbol. For memoryless sources, this is merely the entropy of each symbol, while, in the case of a stationary stochastic process, it is

$$r = \lim_{n \to \infty} H(X_n | X_{n-1}, X_{n-2} \ldots)$$

- In general (e.g., nonstationary), it is defined as

$$r = \lim_{n \to \infty} \frac{1}{n} H(X_n, X_{n-1}, X_{n-2} \ldots)$$

- In information theory, one may thus speak of the "rate" or "entropy" of a language.

Rate Distortion Theory

- $R(D) =$ Minimum achievable rate under a given constraint on the expected distortion.
- $X =$ random variable; $T =$ alphabet for a compressed representation.
- If $x \in X$ is represented by $t \in T$, there is a distortion $d(x, t)$

$$
\begin{aligned}
R(D) &= \min_{\{p(t|x):\langle d(x,t)\rangle \le D\}} I(T, X). \\
\langle d(x,t)\rangle &= \sum_{x,t} p(x,t) d(x,t) \\
&= \sum_{x,t} p(x) p(t|x) d(x,t)
\end{aligned}
$$

- Introduce a Lagrange multiplier parameter β and
- Solve the following **variational problem**

$$\mathcal{L}_{min}[p(t|x)] = I(T;X) + \beta \langle d(x,t) \rangle_{p(x)p(t|x)}.$$

- We need

$$\frac{\partial \mathcal{L}}{\partial p(t|x)} = 0.$$

Since

$$\mathcal{L} = \sum_x p(x) \sum_t p(t|x) \log \frac{p(t|x)}{p(t)} + \beta \sum_x p(x) \sum_t p(t|x) d(x,t),$$

we have

$$p(x) \left[\log \frac{p(t|x)}{p(t)} + \beta d(x,t) \right] = 0.$$

$$\Rightarrow \frac{p(t|x)}{p(t)} \propto e^{-\beta d(x,t)}.$$

Summary

- In summary,

$$p(t|x) = \frac{p(t)}{Z(x,\beta)} e^{-\beta d(x,t)} \qquad p(t) = \sum_x p(x) p(t|x).$$

$Z(x,\beta) = \sum_t p(t) \exp[-\beta d(x,t)]$ is a Partition Function.

- The Lagrange parameter in this case is positive; It is determined by the upper bound on distortion:

$$\frac{\partial R}{\partial D} = -\beta.$$

Redescription

- Some hidden object may be observed via two views X and Y (two random variables.)
- Create a common descriptor T
- Example $X =$ words, $Y =$ topics.

$$
\begin{aligned}
R(D) &= \min_{p(t|x):I(T:Y)\geq D} I(T;X) \\
\mathcal{L} &= I(T:X) - \beta I(T;Y)
\end{aligned}
$$

- Proceeding as before, we have

$$
\begin{aligned}
p(t|x) &= \frac{p(t)}{Z(x,\beta)} e^{-\beta D_{KL}[p(y|x)\|p(y|t)]} \\
p(t) &= \sum_x p(x) p(t|x) \\
p(y|t) &= \frac{1}{p(t)} \sum_x p(x,y) p(t|x) \\
p(y|x) &= \frac{p(x,y)}{p(x)}
\end{aligned}
$$

- **Information Bottleneck** $= T$.

Blahut-Arimoto Algorithm

- Start with the basic formulation for RDT; Can be changed *mutatis mutandis* for IB.
- **Input:** $p(x)$, T, and β
- **Output:** $p(t|x)$

Step 1. Randomly initialize $p(t)$

Step 2. **loop until** $p(t|x)$ converges (to a fixed point)

Step 3. $p(t|x) := \frac{p(t)}{Z(x,\beta)} e^{-\beta d(x,t)}$

Step 4. $p(t) := \sum_x p(x) p(t|x)$

Step 5. **endloop**

Convex Programming: Optimization of a convex function over a convex set $\mapsto$ Global optimum exists!

[End of Lecture #??]

See you next week!