

Stochastic Graphical Models and Applications

Basilis Gidas
Brown University
Short Course on
Mathematical Methods in Speech and Image Analysis
Institute of Mathematics and It's Applications
University of Minnesota*

September 11th-13th, 2000

1 Markov Random Fields and Their Gibbs Representations

1.1 Markov Random Fields (MRF)

Consider the indexing set $\Lambda = \{1, 2, \dots, N\}$ (note that this is a finite countable set) that indexes the following random arrays,

$$X = \{X_i : i \in \Lambda\} \quad (1)$$

where the $X_i \in R$ and R is finite. For example the X_i 's can represent the intensity of a sound wave for speech recognition, or a pixel intensity in grey scale for a picture. Furthermore $P(\mathcal{R}^\Lambda)$ is the set of probability measures on $\Omega = \mathcal{R}^\Lambda$, which just represents the probability measures on the arrays. Consider the following graphs $\mathcal{G} = \{\Lambda, \xi\}$ where Λ represents the vertices of the graph and ξ represents the edges of the graph.

The connection of the neighborhood system is given by,

$$\mathcal{N}_i = \{j \in \Lambda : j \neq i, X_j \longleftrightarrow X_i\} \quad (2)$$

where $X_j \longleftrightarrow X_i$ denotes the sites j that are directly connected with site i . For example in a speech pattern a neighborhood would consist of the points to the left and right of the i^{th} index point. In a picture this neighborhood can be considered either the 4 points to the vertical or horizontal points away from the i^{th} points. Or the 8 points in vertical, horizontal or diagonal space away from the i^{th} point. We also define the set $\mathcal{N} = \{\mathcal{N}_i\}$ as the set of all neighborhood systems for \mathcal{G} .

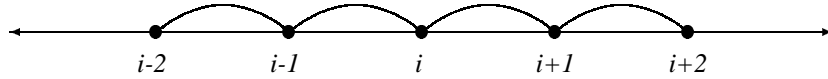
*Thanks to Brad Love for typing up these notes, his trip was supported by the Max Planck Institute for Demographic Research, located in Rostock, Germany.

A $p \in P(\mathcal{R}^\Lambda)$ is said to be a Markov Random Field on \mathcal{G} if,

$$P(X_i | X) = P(X_i | X_{\mathcal{N}_i}) \quad (3)$$

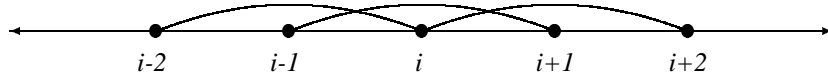
where ${}_iX = \{X_j : j \in \Lambda, j \neq i\}$ and $X_{\mathcal{N}_i} = \{X_j : j \in \mathcal{N}_i\}$ with the additional technical assumption that $P(X) > 0, \forall X \in \mathcal{R}^\Lambda$.

Example 1. A first order MRF on the linear graph has the following relationships,



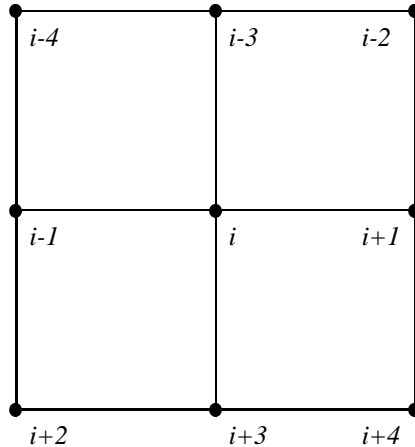
The characterization of the MRF above is $P(X_i | X) = P(X_i | X_{i-1}, X_{i+1})$.

Example 2. A 2nd order MRF of the above is,



The characterization of the MRF above is $P(X_i | X) = P(X_i | X_{i-2}, X_{i+2})$.

Example 3. Consider the following regular lattice,



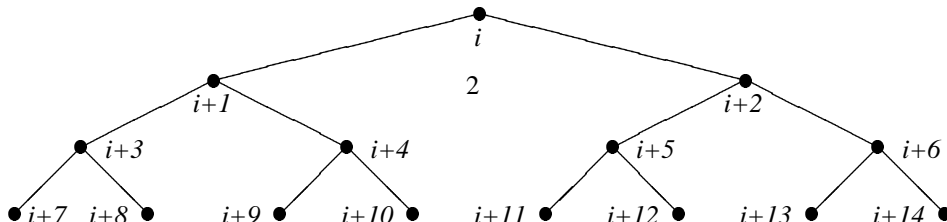
The characterization of the MRF above is

$$P(X_i | X) = P(X_i | X_{i-3}, X_{i-1}, X_{i+1}, X_{i+3}) \quad (4)$$

Example 4. We can further tweak Example 3, by adding into the conditional information that is provided by the diagonal, this can be expressed as,

$$P(X_i | X) = P(X_i | X_{i-4}, X_{i-3}, X_{i-2}, X_{i-1}, X_{i+1}, X_{i+2}, X_{i+3}, X_{i+4}) \quad (5)$$

Example 5. We can consider the following tree-structure,



One characterization of the MRF above is

$$P(X_{i+1}|X_i, X_{i+3}, X_{i+4}) \quad (6)$$

further probabilities can be seen by the connected sites.

1.2 Gibbs Distributions

For the Gibbs Distribution we need to define a set of clicks, where a click is defined as

$$\begin{aligned} \mathcal{C} &= \{\text{set of clicks}\} \\ &= \{c : c = \text{a fully connected subset of sites in } \Lambda\} \end{aligned} \quad (7)$$

this includes $c = \{i\}$ a singleton or $c = \{i_1, \dots, i_m\}$ with any two sites directly connected. It should be noted that any subset of click is also a click.

So with these definitions we are ready to define a Gibbs distribution. $P \in P(\mathcal{R}^\Lambda)$ is Gibbs with respect to $\mathcal{G} = \{\Lambda, \xi\}$ if P can be written as

$$P(X) = \prod_{c \in \mathcal{C}} F_c(X_c) \quad (8)$$

where $X_c = \{X_j : j \in c\}$ and we need the assumption that $F_c(X_c) > 0, \forall X_c \in \mathcal{R}^c$.

A Gibbs distribution can be written as

$$P(X) = \frac{1}{2} e^{\sum_{c \in \mathcal{C}} f_c(X_c)} \quad (9)$$

The representation is not unique since subset of clicks are also clicks, but if we look at maximal size click, then we have the following,

Theorem 1 (Clifford-Hamilton). *The probability measure \mathcal{P} is a Markov Random Field if and only if it is a Gibbs Distribution.*

Additional references are:

- Winkler, G, "Image Analysis, Random Fields and Dynamical Monte Carlo", 1995.
- Geman, D. Lecture Notes.

Additionally it is good to note that the functions F are not necessarily a cdf. \mathcal{P} Gibbs implies \mathcal{P} Markov Random Field is easy to show. For linear graphs \mathcal{P} Markov Random Field implies \mathcal{P} Markov. Additionally possibly higher order implications, i.e. 2nd Markov implies 1st ordered Markov. The other way is also true with additional minimal assumptions. The condition that we need finiteness of the X_i 's and that we need a lattice can be loosened with the addition of higher level mathematics.

2 Hidden Markov Random Fields

Consider the following data, an unobserved random arrays, denoted as $X_i = \{X_i \in \Lambda_x\}$ and the observed data, $Y = \{Y_k : k \in \Lambda_y\}$. Often we want to say something about X , given the observed data Y is contained in probability distributions $P(X|Y)$.

A Hidden Markov Random Field, or HMRF, has the following two properties that (X, Y) are MRF $\mathcal{G}_{X,Y} = \{\Lambda_x \cup \Lambda_y, \xi_{X,Y}\}$ and X is MRF $\mathcal{G}_X = \{\Lambda_X, \xi_X\}$, where ξ_x is a subset of $\xi_{X,Y}$. Here Y is a convolution of the observed X some other and component. This component can be accounting for random noise, i.e. the observed data is effected by noise and the actual data. It is significant to note that $P(X|Y)$, $P(X, Y)$ and $P(X)$ are all MRF although $P(Y)$ is not.

Consider just the observations, i.e. Y and suppose that this is the true distribution, then there is no reason to assume that Y has a Markov distribution. The distribution, although, of the observed data can come through a HMRF.

3 Computerized Characterization of HMRF

Here X , with $P(X)$ the posterior distribution, and (X, Y) , with $P(X, Y) = P(X)P(Y|X)$, where the posterior distribution are the major distributions of interest denoted as $P(X|Y)$.

Three major points of interest of this type of question is the inference problem or the decision problem, i.e.,

$$\hat{X}_{\text{map}} = \arg \max_X P(X|Y). \quad (10)$$

and is to calculate the mean of the posterior distribution $\int X P(X|Y) dx$. The second problem of interest is evaluating the distribution of the unobserved data, i.e. $P(Y)$. The final problem is given by learning, we have two parameters of interest, if we assume a parametric distribution for the two distributions $P(X)$ and $P(Y|X)$. Therefore we have $P_{\theta_1}(X)$ and $P_{\theta_2}(Y|X)$ and the goal is to learn how to estimate (θ_1, θ_2) , by maximizing,

$$\hat{\theta}(Y) = \arg \max_{\theta} P_{\theta}(Y) \quad (11)$$

3.1 The Problem of Inference

Consider the following conditional distribution where,

$$P(X|Y) = \prod_{c \in \mathcal{C}_x} F_c(X_c) \quad (12)$$

maximizing this function is equivalent to minimizing the following,

$$\begin{aligned} -\log(P(X|Y)) &= -\sum_c \log(F_c(X_c)) \\ &= \sum_{c \in \mathcal{C}_x} f_c(X_c) \end{aligned} \quad (13)$$

Consider a linear graph, then the above maximization scheme boils down to minimizing,

$$-\log(P(X, Y)) = f_{1,2}(X_1, X_2) + f_{2,3}(X_2, X_3) + \dots + f_{N-1,N}(X_{N-1}, X_N) \quad (14)$$

This by minimizing the above function by X_1 first, then, do the same thing for X_2 etc.. Once the function is minimized for each value, then repeat this process until convergence of the minimization process.

In general,

$$C(X) = \sum_{c \in \mathcal{C}} f_c(X_c) \quad (15)$$

with $\mathcal{G} = \{\Lambda, \xi\}$ and recall that \mathcal{C} is the set of clicks. Here we need to choose a visitation scheme, which in essence is a permutation of the points, for notational ease we will call these $\sigma(1), \dots, \sigma(N)$ where $|\Lambda| = N$ i.e. the cardinality of Λ is equal to N .

We can deform $C(X)$ as follows,

$$\begin{aligned} C(X) &= \sum_{c \in \mathcal{C}} f_c(X_c) \\ &= \sum_{c: \sigma(1) \in \mathcal{C}} f_c(X_c) + \sum_{c: \sigma(1) \notin \mathcal{C}} f_c(X_c) \end{aligned} \quad (16)$$

then minimize over $X_{\sigma(1)}$.

For ease of notation we will define the visitation of sites by,

$$\begin{aligned} \mathcal{B}_{\sigma(1)} &= \mathcal{N}_{\sigma(1)} \\ &= \{j \in \Lambda : j \neq \sigma(1), j \text{ is connected to } \sigma(1)\} \end{aligned} \quad (17a)$$

are the number of sites that are visited from site $\sigma(1)$ continuing in this fashion we have,

$$\mathcal{B}_{\sigma(2)} = \{j \in \Lambda : j \neq \sigma(1), \sigma(2) \text{ } j \in \mathcal{N}_{\sigma(2)} \text{ or } j \in \mathcal{B}_{\sigma(1)} \text{ if } \sigma(2) \in \mathcal{B}_{\sigma(1)}\} \quad (17b)$$

and iteratively we have,

$$\mathcal{B}_{\sigma(k)} = \{j \in \Lambda : j \neq \sigma(1), \dots, \sigma(k) \text{ } j \in \mathcal{N}_{\sigma(k)} \text{ or } j \in \mathcal{B}_{\sigma(k-1)} \text{ if } \sigma(k) \in \mathcal{B}_{\sigma(k-1)}\} \quad (17c)$$

for $k = 2, 3, \dots, N$. Minimizing the first term of (16) with respect to $X_{\sigma(1)}$ to get $\hat{X}_{\sigma(1)}(\{X_j : j \in \mathcal{N}_{\sigma(1)}\})$ and $C_{\sigma(1)}(\{X_j : j \in \mathcal{B}_{\sigma(1)}\})$. Thus we can re-express (16) as,

$$\begin{aligned} C(X) &= C_{\sigma(1)}(\{X_j : j \in \mathcal{B}_{\sigma(1)}\}) + \sum_{c: \sigma(1) \notin \mathcal{C}} f_c(X_c) \\ &= C_{\sigma(1)}(\{X_j : j \in \mathcal{B}_{\sigma(1)}\}) + \sum_{c: \sigma(1) \notin \mathcal{C}, \sigma(2) \in \mathcal{C}} f_c(X_c) + \sum_{c: \sigma(1), \sigma(2) \notin \mathcal{C}} f_c(X_c) \end{aligned} \quad (18)$$

This process is continued. The overall complexity is given by defining a new relation $|B_\sigma| = \max\{|B_{\sigma(1)}|, \dots, |B_{\sigma(N)}|\}$ and then the overall complexity is given by $O(|\Lambda|\mathcal{R}^{|B_\sigma|+1})$.

3.2 Evaluating the Distribution of Unobserved Data

For learning we have the following:

- X is MRF $\mathcal{G}_X = (\Lambda, \xi)$
- (X, Y) is a MRF for ...
- $P(X|Y)$ is a MRF with respect to $G = (\Lambda, \xi)$, where we can express $P(X|Y) = \prod_C F_C(X_C) = \frac{1}{2}e^{-\sum_c f_c(X_c)} = \frac{1}{2}e^{-H(X)}$

So we have that $X_i \in R$ and $X \in R^\Lambda$. We need to solve the following differential equation,

$$\frac{dX}{dt} = -\nabla \cdot H(X) + \sqrt{2T} \frac{dw}{dt} \quad (19)$$

with initial condition $X(0) = X_0$, w is the Brownian motion. We can solve this equation by multiply both sides by dt , then we have,

$$dX(t) = -\nabla \cdot H(X(t))dt + \sqrt{2T(t)}dw(t) \quad (20)$$

Also note that,

$$\pi(t, x) = \frac{e^{-\frac{1}{T(t)}H(x)}}{Z(t)} \rightarrow \delta(x - \underline{x}) \quad (21)$$

and that,

$$P(x, t) = P(X(t) = x | X(0) = x_0) \rightarrow \delta(x - \underline{x}) \quad (22)$$

if $T(t) \sim \frac{C}{\log(t)}$.

Consider the following arrays, $X^{(0)} \rightarrow X^{(1)} \rightarrow X^{(2)} \rightarrow \dots$ that converge to the correct answer. Then $X \in R^{|\Lambda|} = \Omega$ where $T_1 \geq T_2 \geq \dots \geq T_N \rightarrow 0$. Now you start with $X^{(0)}$ and propose a new $\tilde{X}^{(1)}$. Calculate the hamiltonians of each $X^{(0)}$ and $\tilde{X}^{(1)}$. If the hamiltonian of $\tilde{X}^{(1)}$ is less than or equal to the hamiltonian of $X^{(0)}$ then $X^{(0)} = \tilde{X}^{(1)}$. Otherwise choose $X^{(1)} = \tilde{X}^{(1)}$ with probability $e^{-\frac{1}{T}(H(\tilde{X}^{(1)})-H(X^{(0)}))}$.

4 The Learning Problem

Recall that we had the following $P(X|Y) = P(X)P(Y|X)$. so we had the following probability density function $P(X) = \frac{1}{2}e^{-\sum_c f_c(X_c)}$. There are additional choices in models such as,

$$P_{\theta_1}(X) = \frac{1}{z(\theta)} e^{-\sum_{i \in \Lambda} \sum_{\alpha=1}^M \theta_\alpha V^{(\alpha)}(x_i, X_{N_I^{(\alpha)}})} \quad (23)$$

One of the easiest model is $Y_i = X_i + \epsilon$ where $\epsilon \sim N(m, \sigma^2)$. Two ways to solve this method are using the Pseudo-likelihood method or the variational method.

When looking at $P_{\theta_2}(Y|X)$ where solving for $P_{\theta}(Y)$ where $\theta = (\theta_1, \theta_2)$ and this can be estimated using either the EM algorithm or Variation techniques.

Consider , $\max_{\theta} P_{\theta}(Y)$, and look at the following

$$\begin{aligned}
-\log(P_{\theta}(Y)) &= -\sum_X (\log(P_{\theta}(Y))) P_{\theta'}(X|Y) & (24) \\
&= -\sum_X \left(\log\left(\frac{P_{\theta}(X, Y)}{P_{\theta}(X|Y)}\right) \right) P_{\theta'}(X|Y) \\
&= -\sum_X (\log(P_{\theta}(X, Y))) P_{\theta'}(X|Y) + \sum_X (\log(P_{\theta}(X|Y))) P_{\theta'}(X|Y) \\
&= R(Y : \theta, \theta') - H(Y; \theta, \theta')
\end{aligned}$$

Then this can be done by dynamic programming and updating plugging in the solutions and resolving to get a new solution. The E-step is to compute $R(Y : \theta, \theta^{(n)})$ and the M-Step is to find the $\arg \min_{\theta} R(Y : \theta, \theta^{(n)}) = \theta^{(n+1)}$. Continue until convergence. The following Lemma can be shown,

Lemma 1. $-\log(P_{\theta^{(n)}}(Y)) \geq -\log(P_{\theta^{(n+1)}}(Y))$

The above outlined EM algorithm can be done with linear graphs and tree graphs but with Lattice graphs is computational hard.

The pseudo-likelihood solution is solved by finding the θ from

$$\max_{\theta} \prod_i P_{\theta}(X_i | X_{N_i}) \tag{25}$$

5 Tracking and Detection

Applications of tracking and detection is traffic observation (the difficulty being that the number of cars is random and changing) and subway surveillance.

There are four basic elements that go into the algorithm, they are the representation of objects, representation of the dynamics (you should some how articulate generic properties), we need to use the actual collected data. If we combine these first three components then this is a nonlinear filter problem, so the 4th element is the filtering algorithm.

The first filtering algorithm used is the Monte Carlo filter. One of the nicest filtering ideas in the last 40 years.

5.1 Representation of Objects

The form of the templates and the hierarchical models that are used. The templates used for the deformation move in real time. They are not tracking in 3-D rather the movement in the 2-D as the image is captured by the camera.

The first deformation is the Rigid deformation using location and rotation. The elastic deformation, are in principle infinite dimensional, but for practical purpose must be brought down to a finite dimension. Both models involve parameters that are generally used to calculate the generalized coordinates.

Consider the following details in the deformation of the templates and using generalized coordinates. First you use deformation, then relation and then translation.

A person could try to be very ambitious and try to faithfully represent the trajectories of a moving object. But the this not the goal, the tracking is actually being down by the data. So a simple model will do a good job, when accentuated by the actual observed data. Thus using linear models may be justified, even though these types of models do not allow dicontinuites. But if we go beyond linear models, we can handle the dicontinuites, but make the equations more difficult to solve.

We will start by writing down differential equations that accurately describe the motion. Let,

$$q = \begin{pmatrix} b \\ \theta \\ q_d \end{pmatrix} \quad (26)$$

where q is a function of time or the travel, θ is the rotation and q_d is the deformation parameter, thus a possible model might be,

$$x = b + R\xi R s(\xi) q_d \quad (27)$$

where the kinetic energy is given by,

$$T = \frac{1}{2} \int_B \Gamma(\xi) \cdot \dot{X}^t \dot{X} d\xi = \frac{1}{2} \dot{q}^t \mu(\xi) \dot{q} \quad (28)$$

the dissipation energy is given by

$$D = \frac{1}{2} \int_B \gamma(\xi) \dot{X}^t \dot{X} d\xi = \frac{1}{2} \dot{q}^t C(q) \dot{q} \quad (29)$$

the stress energy is given by

$$U = \frac{1}{2} \int_B \epsilon^t(\xi) \cdot E(\xi) \cdot \epsilon(\xi) d\xi = \frac{1}{2} \epsilon^t(q) \cdot \kappa \cdot \epsilon(q) \quad (30)$$

and here we use Lamé's Constant for κ . Finally we have the lagrangian dynamic is given by,

$$Q = \frac{d}{dt} \cdot \left(\frac{\partial T}{\partial \dot{q}} \right) - \frac{\partial T}{\partial q} + \frac{\partial D}{\partial \dot{q}} + \frac{\partial U}{\partial q} \quad (31)$$

where Q is the external force. We express the state value,

$$X(t) = \begin{pmatrix} q(t) \\ \dot{q}(t) \end{pmatrix} \quad (32)$$

and a more generalized form is given by

$$X(t) = f_k(X_k) + c \quad (33)$$

in the end we use a MRF.

In general we can construct a data matrix, i.e. the image, that frame k is given by $y_k = h_k(X_k, \nu_k)$ where k is a positive integer that relates to a particular frame and ν_k is the random noise. Here y_k is the raw data, but it can also be considered as the data of the frame and the previous frames. Finally it can be considered a highly nonlinear equation of the data.

Now we consider that X_k is a markov model. Y_k is a

$$P(Y_0, Y_1, \dots, Y_k | X_0, X_1, \dots, X_k) = \prod_j P(Y_j | X_j) \quad (34)$$

this is of course a hidden markov model.

Here the algorithm we look at $P(X_k | Y_0, \dots, Y_k)$ and we can generate N samples, around 5000 typically, call these $X_k^{(1)}, \dots, X_k^{(N)}$ and then statistics are calculated on these samples, typically either the mean or the median. It turns out, through the dynamics, then a random generation of the $k - 1$ frame will look very similar to the k frame. We then use the data in the k frame to move around this estimate.

This type of thought goes through two steps. The first step is a predictive step, i.e. trying to predict the next step, i.e. X_k the true frame, from the first $k - 1$ observed frames, i.e. Y_1, \dots, Y_{k-1} . Thus,

$$\begin{aligned} \tilde{X}_k^{(n)} &= f_{k-1} \left(X_{k-1}^{(n)}, w_k^{(n)} \right) \\ &\sim P(X_k | Y_0, \dots, Y_{k-1}) \end{aligned} \quad (35)$$

We then update by taking $\tilde{X}_k^{(1)}, \tilde{X}_k^{(2)}, \dots, \tilde{X}_k^{(N)}$ where,

$$q_k^{(n)} = \frac{P \left(Y_k | X_k = \tilde{X}_k^{(n)} \right)}{\sum_{m=1}^N P \left(Y_k | X_k = \tilde{X}_k^{(m)} \right)} \quad (36)$$

we then get $q = (q_k^{(0)}, q_k^{(1)}, \dots, q_k^{(N)})$. We then sample q to generate

$$\left(X_k^{(1)}, X_k^{(2)}, \dots, X_k^{(N)} \right) \quad (37)$$

It can be shown that $X_k^{(N)} \sim P(X_k | Y_0, \dots, Y_k)$.

We can calculate the cost function and call this $c(Y_k, X_k)$ where a high cost is a predicted picture that is far from the observed and vice-versa. To implement the algorithm (MCF) one needs not to specify $P(Y_k | X_k)$ but only $q_k^{(n)}$. In the implementation we take

$$q_k^{(n)} = \frac{c \left(Y_k | \tilde{X}_k^{(n)} \right)}{\sum_{m=1}^N c \left(Y_k | \tilde{X}_k^{(m)} \right)} \quad (38)$$