Algorithmic Machine Learning (9 pages) Spring 2014

Fourier Theoretic Probabilistic Inference over Permutations

Instructor:Bobby Kleinberg Sung Min Park (sp765)

1 Introduction

This is a survey of the main ideas from "Fourier Theoretic Probabilistic Inference over Permutations" by Huang et al.

2 Motivation

Permutations naturally arise in many real world problems. For instance, a voting preference for a set of candidates is a permutation. Another example that motivates this study is that of *identity management problem*. There are n tracks, each of which is occupied by exactly one of n people. When people pass near each other, confusion can arise and we may lose track of who is on which track. The task is to maintain a probability distribution over mapping from object tracks to identities (which is over of permutation of n elements).

Of course, what makes maintaining probability distribution over a permutation group S_n hard is that S_n becomes large for even very small values of n. Common heuristics for distributions, such as graphical models, cannot capture the mutual exclusability of constraints of a permutation.

2.1 Random walks and the Forward algorithm

First, we describe our model for the above identity management problem. The transition model is that of random walks over the permutation group. At each time step t, we have a a new *model* $Q^{(t)}$, which is a probability distribution that models some event. We sample a random permutation $\pi^{(t)}$ from $Q^{(t)}$, and generate the new state by $\sigma^{(t+1)} = \pi^{(t)}\sigma^{(t)}$.

We also make observations $z^{(t)}$. We assume that we know the transition models $P(\sigma^{(t)}|\sigma^{(t-1)})$ for each time step, as well as the observation models $P(z^{(t)}|\sigma^{(t)})$ (for example, this might reflect the distribution over the color of clothing for each individual).

Based on this information, we can compute the likelihood of any permutation after a sequence of mixing events and observations by iterating the following two steps:

1. *prediction/roll up*: $P(\sigma^{(t+1)}) = \sum_{\sigma^{(t)}} P(\sigma^{(t+1)}|\sigma^{(t)}) \cdot P(\sigma^{(t)})$
2. *conditioning*: $P(\sigma|z) = \frac{1}{Z}P(z|\sigma) \cdot P(\sigma)$ (this is just application of Bayes' rule followed by normalization using Z)

Naively, this algorithm runs in $O((n!)^2$, which is impractical.

3 Fourier Transform on Finite Groups

Their first goal is to find a way to more compactly represent distributions over permutations (a distribution is just a function from the permutation group to the reals).

Recall that the familiar Fourier transform is used to decompose a function whose domain is the real line into a spectrum of frequencies. Often, the advantage of representing the function in the frequency domain is that most of the energy of the function is concentrated in a few low order frequencies; hence, we can approximate the function well by keeping track of a fewer number of coefficients.

There is a group theoretic generalization, which look at functions whose domain is a group G. The hope is that we can represent the probability distribution on G more compactly if we could somehow do the analog of Fourier transform on the real line. But first, we must introduce some concepts from representation theory.

3.1 Group Representation

A **representation** of a group G on a vector space V over a field K is a homomorphism $\rho : G \to GL(V)$, i.e. $\forall \sigma_1, \sigma_2 \in G, \rho(\sigma_1\sigma_2) = \rho(\sigma_1) \cdot \rho(\sigma_2)$. $GL(V)$ is the general linear group, the group of all bijective linear transformations $V \to V$, with functional composition as group operation. For our purposes, we focus on the case where V has finite dimension d_ρ, so $GL(V)$ is the set of invertible $d_\rho \times d_\rho$ matrices. d_ρ is called the *degree* of the representation.

We focus most of our discussion on the finite group S_n. A few examples of representation on S_n:

1. The trivial representation $\rho_{(n)}$ maps each element of the symmetric group to 1. While this representation may not not so interesting or useful, it will later play its role later.
2. The first-order permutation representation $\tau_{(n-1,1)}$, given by the matrix $[\tau_{(n-1,1)}(\sigma)]_{ij} = \mathbb{1}\{\sigma(j) = i\}$. The images are the familiar $n \times n$ permutation matrices.
3. The alternating representation of S_n, $\rho_{(1,\ldots,1)}$, which maps σ to $+1$ if σ can be written as the composition of an even number of transpositions, and -1 otherwise.

In the above example ρ indicates that that the representation is *irreducible*, whereas τ indicates that it is not. We define *irreducibility* below.

Irreducible representations
A linear subspace $W \subset V$ is called G**-invariant** if $gw \in W \ \forall g \in G, \forall w \in W$. The restriction of ρ to a G-invariant subspace is called a **subrepresentation**. A representation ρ is said to be **irreducible** if it has only trivial subrepresentations (subrepresentations based on trivial subspaces, V and $\{0\}$). Otherwise, the representation is **reducible** and can be factored into a direct sum of irreducible representations. The direct sum of two representations ρ_1 and ρ_2 is a new representation defined as:

$$\rho_1 \oplus \rho_2(\sigma) \triangleq \begin{pmatrix} \rho_1(\sigma) & 0 \\ 0 & \rho_2(\sigma) \end{pmatrix}$$

For a finite group, there are only a finite number of irreducible representation up to equivalence. That is, for any representation τ, there $\exists$ a matrix C s.t.

$$C^{-1} \cdot \tau(\sigma) \cdot C = \bigoplus_{\rho \in \Gamma} \bigoplus_{j=1}^{z_\rho} \rho$$

where Γ is the set of all distinct irreducible representations of group G, and z_ρ is the number of times ρ appears in the direct sum. C is sometimes referred to as a *similarity transform*.

We leave it as an exercise to the reader to decompose $\rho_{(2,1)}$ from our example into a direct sum of two irreducible representations.

Now, we explain the reason for the subscripts in the represenations above.

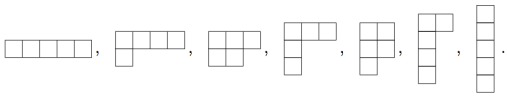
3.2 Ferrer's diagrams

The subscript in the above representations stands for a partition. A *partition* of n is a tuple of positive integers $(\lambda_1, ..., \lambda_\ell)$ that sum to n. For convenience, assume $\lambda_1 \geq \cdots \geq \lambda_\ell$.

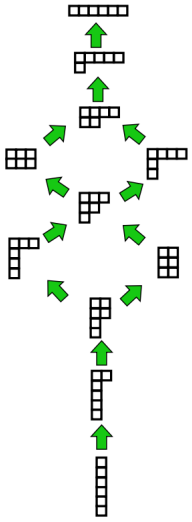
We will find it useful to visualize a partition using a *Ferrer's diagram*. For example, for partitions of $n = 5$,

(5), (4,1), (3,2), (3,1,1), (2,2,1), (2,1,1,1) (1,1,1,1,1)

their respective Ferrers diagrams are



Paritions of n form a partial order given the following dominance ordering: $\lambda \trianglerighteq \mu$ if for each i, $\sum_{k=1}^{i} \lambda_k \geq \sum_{k=1}^{i} \mu_k$. This order will become useful later when we consider the Fourier transform. The ordering for $n = 6$ is shown as an example:



A *Young tabloid* is an assignment of numbers $1, ..., n$ to the boxes of a Ferrers diagram; each row is considered an unordered set. A variation where each row is an ordered tuple is called a *Young tableaux*. A *Young tableaux* is *standard* if its entries are increasing to the right along rows and down columns.

These combinatorial objects are useful for the following reason: Every irreducible representation of S_n corresponds to some partition of n. There is in fact an algorithm to compute the irreducible representation corresponding to a given partition; this makes use of standard Young tableaux for the given partition. The idea of the algorithm is to first compute the irreducible representation at adjacent transpositions [1]; as every permutation is a composition of adjacent transpositions, we can then simply multiply the corresponding representation matrices since representation is homomorphic.

3.3 Fourier transform

We are now ready to define the Fourier transform on a finite group. We define the Fourier transform of $f : G \to \mathbb{R}$ at the representation ρ to be

$$\hat{f}_\rho = \sum_\sigma f(\sigma)\rho(\sigma)$$

Note that now a Fourier transform coefficient is not necessarily a single number, but rather a matrix.

Just like on the real line, the value of f at any element of G can be represented using the Fourier transform coefficients as follows:

$$f(\sigma) = \frac{1}{|G|} \sum_\lambda d_{\rho_\lambda} Tr[\hat{f}^T_{\rho_\lambda} \cdot \rho_\lambda(\sigma)]$$

where λ indexes over the collection of all irreducibles of G.

We revisit some of our examples of representation earlier to give some intuition of what the coefficients represent:

[1] For more details, see Appendix B of the paper

1. The trivial representation $\rho_{(n)}$ corresponds to the constant basis function. Recall that for functions on the real line the Fourier transform at the zero frequency gives the DC component of a signal. The analog here is that the Fourier transform of f at the trivial representation $\rho_{(n)}$ is $\hat{f}_{\rho_{(n)}} = \sum_{\sigma} f(\sigma)$. If f is a probability distribution, then $\hat{f}_{\rho_{(n)}} = 1$.

2. The first-order permutation representation $\tau_{(n-1,1)}$ has the following Fourier transform:

$$[\hat{f}_{\tau_{(n-1,1)}}]_{ij} = \sum_{\sigma \in S_n} f(\sigma)[\tau_{(n-1,1)}]_{ij} = \sum_{\sigma \in S_n} \mathbb{1}\{\sigma(j) = i\} = \sum_{\sigma : \sigma(j) = i} f(\sigma)$$

   The (i, j)-th element of the transform is the marginal probability that a random permutation drawn from the original distribution maps element j to i. (and maps the other $n-1$ elements to among them, but this is redundant since permutation is bijective)

   Similarly, if we consider representations $\tau_{(n-2,2)}$, corresponding to tabloids of shape $\lambda = (n-2, 2)$, we can find second-order marginals such as the marginal probability that $\{1, 2\}$ maps to $\{2, 4\}$. This particular example is *unordered*, but if we want to instead answer *ordered* questions such as what is the marginal probability that $\{1\}$ maps to $\{2\}$ and $\{2\}$ maps to $\{4\}$, we can instead look at the representations corresponding to tabloids of shape $\lambda = (n-2, 1, 1)$. This can be visualized as follows:

$$P\left(\sigma : \sigma\left(\left\{\begin{array}{|c|c|c|c|}\hline 3&4&5&6\\\hline 1\\\cline{1-1} 2\\\cline{1-1}\end{array}\right\}\right) = \left\{\begin{array}{|c|c|c|c|}\hline 1&3&5&6\\\hline 2\\\cline{1-1} 4\\\cline{1-1}\end{array}\right\}\right)$$

3. The representation $\tau_{(1,\ldots,1)}$ (note, this is *not* the alternating representation $\rho_{(1,\ldots,1)}$) exactly recovers the original probabilities $P(\sigma)$.

In general, Fourier coefficients of a representation τ_μ can be viewed as the marginal probabilities of Young tabloids of shape λ. Say that we fixed an ordering[2] on the set of possible Young tabloids of shape λ, $\{t_1\}, \{t_2\}, \ldots$ and define the permutation representation $\tau_\lambda(\sigma)$ as the following matrix:

$$[\tau_\lambda(\sigma)]_{ij} = \begin{cases} 1 & \text{if } \sigma(\{t_j\}) = \{t_i\} \\ 0 & \text{if not} \end{cases}$$

Then, if $P(\sigma)$ is a probability distribution, then

$$\begin{aligned} \left[\hat{P}_{\tau_\lambda}\right]_{ij} &= \sum_{\sigma \in S_n} P(\sigma)[\tau_\lambda(\sigma)]_{ij} \\ &= \sum_{\sigma : \sigma(\{t_j\}) = \{t_i\}} \\ &= P(\sigma : \sigma(\{t_j\}) = \{t_i\}) \end{aligned}$$

Hence, marginal probabilities corresponding to Young tabloids of shape λ are given by the Fourier transform at the representation τ_λ.

Now, it is often possible to reconstruct lower order marginals by summing over the appropriate higher order marginal probabilities, so there is some redudancy in the above representations τ_λ. But we can construct a partially ordered set of representations so that ρ_λ captures all the information at the 'frequency' λ which was not already captured at lower frequency representations ρ_μ where $\mu \trianglerighteq \lambda$ in the partial order given by the

[2] It is unclear from the paper how one might define such an ordering that is useful

dominance hierarchy. These representations ρ_λ turn out to be exactly the irreducible representations of the group S_n.

This idea is precisely presented in the following:

Theorem 1. For each partition λ there exists a matrix C_λ s.t. $C_\lambda^T \cdot \tau_\lambda(\sigma) \cdot C_\lambda = \bigoplus_{\mu \trianglerighteq \lambda} \bigoplus_{\ell=1}^{K_{\lambda\mu}} \rho_\mu(\sigma)$

The multiplicities $K_{\lambda\mu}$ are called the *Kostka numbers* and can be computed using Young's rule. It follows easily from the definition that we can compute the Fourier transform at τ_λ as follows:

$$\hat{f}_{\tau_\lambda} = C_\lambda \cdot \left[\bigoplus_{\mu\lambda} \bigoplus_{\ell=1}^{K_{\lambda\mu}} \hat{f}_{\rho_\mu} \right] \cdot C_\lambda^T$$

The dimensions of ρ_λ is polynomial for fixed k; roughly $O(n^{2k})$ storage is required to maintain kth order marginals. Hence, it is possible to compactly summarize distributions over permutations by saving only the low-frequency terms of the Fourier transform.

4 Algorithms in the frequency domain

We have found a way to compactly represent distributions over permutations in the Fourier domain. However, if we were to naively use inference algorithms, we would have to constantly transform the coefficients back and forth into the Fourier domain, and even with FFT, this is prohibitively expensive as the transform runs in $O(n! \log n!)$.

We revisit the forward algorithm from section 1, but this time we express its operations entirely in the frequency domain, which will allow us to avoid the above transformation cost.

4.1 Prediction/Rollup

The prediction/rollup step can be rewritten as a convolution[3] of two distributions:

$$\begin{aligned} P(\sigma^{(t+1)}) &= \sum_{\sigma^{(t)}} P(\sigma^{(t+1)}|\sigma^{(t)}) \cdot P(\sigma^{(t)}) \\ &= \sum_{(\sigma^{(t)},\pi^{(t)}):\sigma^{(t+1)}=\pi^{(t)}\cdot\sigma^{(t)}} Q^{(t)}(\pi^{(t)}) \cdot P(\sigma^{(t)}) \\ &= \sum_{\sigma^{(t)}} Q^{(t)}(\sigma^{(t+1)}(\,)) \cdot P(\sigma^{(t)}) \\ &= [Q^{(t)} * P](\sigma^{(t+1)}) \end{aligned}$$

where $Q^{(t)} * P$ is defined as the convolution. By the convolution theorem, $\widehat{[Q * P]}_\rho = \widehat{Q}_\rho \cdot \widehat{P}_\rho$, we can simply update $\widehat{P}_\rho^{(t+1)} \leftarrow \widehat{Q}_\rho^{(t)} \cdot \widehat{P}_\rho^{(t)}$ in the Fourier domain.

[3]This is an extension of the familiar notion of convolutions, where addition and subtraction become function composition and inverse

Computing transforms for particular models
One reasonable question from an implementer's point of view is how we might compute the Fourier transforms for a given Q. Using the definition, $\widehat{Q}_\rho = \sum_\sigma Q(\sigma) \cdot \rho(\sigma)$. The concern is that in order to compute the transform at a representation, we have to sum over all elements of S_n; this is exponential, and was in fact the very reason why we started studying this problem!

Luckily, for particular models Q we can more easily compute $\widehat{Q}_\rho$ without iterating over all elements of S_n. One common model is the *pairwise mixing model*, defined as

$$Q_{ij}(\pi) = \begin{cases} r & \text{if } \pi = \varepsilon \\ 1-r & \text{if } \pi = (i,j) \\ 0 & \text{if not} \end{cases}$$

This models the event where elements i and j swap their identities with probability $1-r$. The Fourier coefficient for ρ_λ is simply $[\widehat{Q}_{ij}]_{\rho_\lambda} = rI + (1-r)\rho_\lambda((i,j))$. There are several other common models that arise in practice that the paper discusses.

One nice property of convolution operation is that it acts pointwise, meaning that the value of convolution of two distributions at an element is affected only by the corresponding values of each distribution at the element. This is nice because if we only keep the low-order transforms they will propagate only to those same levels; we say that the prediction/rollup operation does not increase the representational complexity.

4.2 Conditioning

Unfortunately, we don't have this nice property for the conditioning step. To get an intuition for why, as an example consider the first-order marginal probabilities:

$$\Pr(\text{Alice is at Track 1 or Track 2}) = 0.5$$
$$\Pr(\text{Bob is at Track 1 or Track 2}) = 0.5$$

If we then make the following first-order observation:

$$\Pr(\text{Cathy is at Track 1 or Track 2}) = 1$$

then we can infer that Alice and Bob cannot possibly both occupy Tracks 1 and 2. That is,

$$\Pr(\{\text{Alice, Bob}\} \text{ are on Tracks } \{1, 2\}) = 1$$

After conditioning on the new observation, we are left with a second-order marginal despite that both the prior and likelihood functions were known up to first order. The example shows that conditioning can smear information from low-order Fourier coefficients to high-order coefficients.

Recall that the conditioning operations is: $P(\sigma|z) = \frac{1}{Z} P(z|\sigma) \cdot P(\sigma)$. In order to express this operation in the Fourier domain, we need to be able to translate the multiplication of two functions into the Fourier domain. The idea is to manipulate the function $f(\sigma)g(\sigma)$ so that it looks like the result of an inverse Fourier transform. After some manipulations using Kronecker products of matrices, we arrive at the following result [4]

[4] For proof, see Proposition 10, p. 1024 of Huang et al.

Theorem 2. Let $\hat{f}, \hat{g}$ be the Fourier transforms of functions f and g, and for each ordered pair of irreducibles (ρ_λ, ρ_μ), define $A_{\lambda\mu} \triangleq C_{\lambda\mu}^{-1} \cdot (\hat{f}_{\rho_\mu} \oplus \hat{g}_{\rho_\mu}) \cdot C_{\lambda\mu}$. Here, $C_{\lambda\mu}$ is a similarity transform s.t. for any $\sigma \in G$,

$$C_{\lambda\mu}^{-1} \cdot [\rho_\lambda \otimes \rho_\mu](\sigma) \cdot C_{\lambda\mu} = \bigoplus_{\nu \in \Gamma} \bigoplus_{\ell=1}^{z_{\lambda\mu\nu}} \rho_\nu(\sigma)$$

Then, the Fourier transform of the pointwise product fg is:

$$[\widehat{fg}]_{\rho_\nu} = \frac{1}{d_{\rho_\nu}|G|} \sum_{\lambda\mu} d_{\rho_\lambda} d_{\rho_\mu} \sum_{\ell=1}^{z_{\lambda\mu\nu}} A_{\lambda\mu}^{(v,\ell)}$$

In order to apply the above theorem, we need to know $z_{\lambda\mu\nu}$, which are called the Clebsch-Gordan series, and the similarity transforms $C_{\lambda\mu}$, which are called the Clebsch-Gordan coefficients. There are no known analytical formulas for finding the entire Clebsch-Gordan series and coefficients. Luckily, we only need to compute them once for a particular group, and just store them in a look up table.

4.3 Bandlimiting and projection

For efficiency, we can maintain the Fourier transform at a reduced set of coefficients during the above inference process. If the conditioning step introduces new higher order terms, we simply discard them; this is where error is introduced into the system. Empirical evidence shows that if the distribution is relatively 'smooth' most of its energy is stored in the low-order Fourier coefficients, so we don't accumulate much error.

One concern is that the Fourier transform we have at the end of the inference process might correspond to marginal probabilities that are inconsistent or even negative. Let us refer to the space of coefficients corresponding to consistent joint distributions as the *marginal polytope*.

In order to deal with this issue, after each conditioning step, we project the approximate distribution f to the nearest function in a relaxed marginal polytope. that is closest in the L_2 norm, by solving the following quadratic program:

$$\begin{aligned}
\text{minimize} \quad & \sum_{\lambda \in \Lambda} d_\lambda \mathrm{Tr}\left[(\hat{f} - \hat{f}^{proj})_{\rho_\lambda}^T (\hat{f} - \hat{f}^{proj})_{\rho_\lambda}\right] \\
\text{subject to} \quad & \left[\hat{f}^{proj}\right]_{(n)} = 1 \\
& \left[C_{\lambda^{MIN}} \cdot \left(\bigoplus_{\mu \trianglerighteq \lambda^{MIN}} \bigoplus_{\ell=1}^{K_{\lambda^{MIN}\mu}} \hat{f}_{\rho_\mu}^{proj}\right)\right]_{ij} \geq 0 \quad \forall (i,j)
\end{aligned}$$

The objective expresses the distance between our original function f and our projection f^{proj} in terms of their Fourier coefficients using the Plancherel Theorem. The first constraint requires that f^{proj} is a valid probability distribution (i.e. it sums to 1), and the second set of constraints states that $\hat{f}_{\rho_{\lambda^{MIN}}}^{proj}$ has all nonnegative entries. From our discussion in section 3.3, this means that marginal probabilities are nonnegative.

Even after projection, there might not necessarily exist a joint probability distribution on S_n consistent with those marginals.

5 Conclusion

Huang et al. was able to efficiently perform inference operations over probabilitiy distribution on the symmetric group by expressing their operations entirely in the Fourier domain, and maintaining only a limited set of low-order terms. The mathematical machinery used in this paper (most of which were actually introduced before this paper) are intriguing, but are quite heavy duty. In particular, translating their algorithms into a usable implementation appears to be a highly non-trivial task, even considering that they provide several algorithms for working with representation matrices.

Also, the paper could definitely provide more detail on how the resulting output of their inference operations is actually used to predict the identity mapping for their real camera network experiment, given the fact that there is not necessarily exist a joint probability distribution on S_n consistent with the final marginals.

References

Images are from the original paper: `http://jmlr.org/papers/volume10/huang09a/huang09a.pdf`