Efficient computation of maximum likelihood estimators of hierarchical subspace models

Hisayuki Hara *
Faculty of Economics, Niigata University, Niigata, Japan
hara@econ.niigata-u.ac.jp

Abstract

In this paper we discuss efficient computation of the maximum likelihood estimator (MLE) for general log linear models for contingency tables. In general an iterative algorithm, such as the iterative proportional fitting (IPF), is required for computing the MLE of a log linear model. For a hierarchical model, it is well known that the updating rule for IPF is localized according to the separation or decomposition of the graph describing conditional independence structure among variables of the model and then the computational cost is reduced (e.g. [1]). Here we extend the algorithm for a hierarchical model to the hierarchical subspace model (HSM) introduced by [2] and propose an efficient algorithm of updating rule of IPF for HSM.

Keywords : chordal extension, graphical model, hierarchical model, information propagation, iterative proportional fitting

1 Introduction

Consider an m-way contingency table $\{x(\boldsymbol{i}), \boldsymbol{i} \in \mathcal{I}\}$ with the set of variables $[m] = \{1, \ldots, m\}$, where $\boldsymbol{i}$ denotes each cell and $\mathcal{I}$ denotes the set of cells. $x(\boldsymbol{i})$ is a frequency for a cell $\boldsymbol{i}$. For a subset $D \subset [m]$, denote by $\boldsymbol{i}_D$ a marginal cell for D. Let $\mathcal{I}_D$ be the set of marginal cells for D. Let Δ be a simplicial complex with the set of vertices $[m]$. Then hierarchical model defined by Δ is expressed by

$$\log p(\boldsymbol{i}) = \sum_{D \in \Delta} \mu_D(\boldsymbol{i}_D). \tag{1}$$

Models (1) with linear restrictions among natural parameters is extensively used in practical data analysis (e.g. [3, 4]). The canonical form of such a model is written by

$$\log p(\boldsymbol{i}) = \sum_{D \in \tilde{\Delta}} \phi_D(\boldsymbol{i}_D)\boldsymbol{\beta}_D, \tag{2}$$

where $\phi_D(\boldsymbol{i}_D) = \{\phi_D^k(\boldsymbol{i}_D)\}_{k=1,\ldots,K_D}$ is $1 \times K_D$ vector of known functions of $\boldsymbol{i}_D$ and $\boldsymbol{\beta}_D$ is $K_D \times 1$ parameter vector. Since the model (2) is the models (1) with linear constraint, these two models have the same interaction terms corresponding to Δ. In canonical form (2), however, we note that $\tilde{\Delta}$ is also a simplicial complex but not equal to Δ in general (e.g. [2]). The model (2) is still an exponential family and therefore a linear subspace of hierarchical model (1). The model (2) is also called hierarchical subspace model (HSM, [2]).

In this paper we discuss an efficient computation of maximum likelihood estimator (MLE) of HSM (2). Define

$$t_D := \sum_{\boldsymbol{i}_D \in \mathcal{I}_D} \phi_D(\boldsymbol{i}_D) x(\boldsymbol{i}_D).$$

Then the sufficient statistic t for HSM (2) is

$$t = \{t_D \mid D \in \tilde{\Delta}\}.$$

Then the likelihood equations are written by

$$t_D = E[t_D] = n \sum_{\boldsymbol{i}_D \in \mathcal{I}_D} \phi_D(\boldsymbol{i}_D) p(\boldsymbol{i}_D), \quad D \in \tilde{\Delta}.$$

In general MLE of HSM (2) is not explicitly obtained and iterative computation is required to compute MLE. Iterative proportional fitting algorithm (IPF) is one of the popular algorithms for computing MLE. IPF for HSM is described as follows.

Algorithm 1 (IPS for HSM)**.**

Step 0 $t \leftarrow 0$, $p^{(t)}(\boldsymbol{i}) \leftarrow 1/n$

Step 1 For all $D \in \tilde{\Delta}$ and $k = 1, \ldots, K_D$, update $p^{(t)}(\boldsymbol{i})$ as

$$p^{(t+1)}(\boldsymbol{i}) \leftarrow \frac{\sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) x(\boldsymbol{i}_D)}{n \sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) p^{(t)}(\boldsymbol{i}_D)} \cdot p^{(t)}(\boldsymbol{i}), \quad \boldsymbol{i} \in \mathcal{I}. \tag{3}$$

Step 2 If $p^{(t+1)}(\boldsymbol{i})$ converges for all $\boldsymbol{i}$, output $p^{(t+1)}(\boldsymbol{i})$. If not, go back to Step 1.

It is well known that the convergence of this algorithm to MLE is guaranteed (e.g. [5]). Step 1 requires $O(|\mathcal{I}|)$ times updates. For a hierarchical model (1), however, it is known that the updating rule for IPF is localized according to the separation or decomposition of the graph describing conditional independence structure among variables of the model and then the computational cost can be reduced (e.g. [1]). The main purpose of this paper is to extend the argument in hierarchical model to HSM. As discussed in [2], decomposition of an HSM does not always correspond to that of its conditional independence graph G_Δ and hence the algorithm of [1] cannot be applied directly to HSM. In this paper we show that by defining the decomposition of a HSM properly, we can apply the algorithm of [1] also to HSM.

2 Main Results

2.1 Decomposition of HSM

As discussed the previous section, the canonical form of HSM (2) is defined by a simplicial complex $\tilde{\Delta}$. In general $\tilde{\Delta}$ is not equal to Δ. As mentioned in the previous section, the model (2) and the models (1) have the same interaction terms corresponding to Δ. Let H_Δ be a hypergraph with the set of vertices $[m]$ and the set of hyperedges Δ. Then H_Δ also describes the conditional independence structure of both (1) and (2). Let $\mathcal{C}$ and $\mathcal{S}$ be the set of compact components and dividers of H_Δ (see [1]). Let $\nu(S)$ be the multiplicity of $S \in \mathcal{S}$ (e.g. [5]). Then $p(\boldsymbol{i})$ is expressed by

$$p(\boldsymbol{i}) = \frac{\prod_{C \in \mathcal{C}} p(\boldsymbol{i}_C)}{\prod_{S \in \mathcal{S}} p(\boldsymbol{i}_S)^{\nu(S)}}. \tag{4}$$

When $\phi_D(\boldsymbol{i}_D)\boldsymbol{\beta}_D = \mu_D(\boldsymbol{i}_D)$, $\forall \boldsymbol{i}_D \in \mathcal{I}_D$, in (2), D is called saturated in (2). For any hierarchical model (1), every S is known to be saturated in (1). For HSM, however, S is

not always saturated. Furthermore, if S is not saturated in (2), marginal models $p(\boldsymbol{i}_C)$ and $p(\boldsymbol{i}_S)$ are no longer full exponential families and therefore they are not HSMs in general. If there is a linear constraint of parameters across more than one facet, $\tilde{\Delta} \supset \Delta$ in the canonical form (2). This fact indicates that the decomposition of HSM does not necessarily correspond to that of H_Δ or G_Δ.

Let $H_{\tilde{\Delta}}$ be a hypergraph with the set of vertices $[m]$ and the set of hyperedges $\tilde{\Delta}$. Let $\mathcal{S}^*$ be the set of dividers of $H_{\tilde{\Delta}}$ which are saturated in (2). Let $\mathcal{C}^*$ be the set of compact component defined by S^*. The elements of $\mathcal{C}^*$ are also called extended compact components ([2]). Then $p(\boldsymbol{i})$ satisfies

$$p(\boldsymbol{i}) = \frac{\prod_{C^* \in \mathcal{C}^*} p(\boldsymbol{i}_{C^*})}{\prod_{S^* \in \mathcal{S}^*} p(\boldsymbol{i}_{S^*})^{\nu(S^*)}} \tag{5}$$

and marginal models $p(\boldsymbol{i}_{C^*})$ satisfy

$$p(\boldsymbol{i}_{C^*}) = \sum_{C^* \cap D, D \in \tilde{\Delta}} \phi_D(\boldsymbol{i}_D)\boldsymbol{\beta}_D.$$

Hence $p(\boldsymbol{i}_{C^*})$ is still an HSM ([2]). Since $S \in \mathcal{S}^*$ is saturated in (2), the MLE of $p(\boldsymbol{i}_{S^*})$ is the corresponding relative marginal frequency $x(\boldsymbol{i}_{S^*})/n$, where n is total sample size. Therefore the MLE of $p(\boldsymbol{i})$ is expressed by

$$\hat{p}(\boldsymbol{i}) = \frac{\prod_{C^* \in \mathcal{C}^*} \hat{p}(\boldsymbol{i}_{C^*})}{\prod_{S^* \in \mathcal{S}^*} (x_S(\boldsymbol{i}_{S^*})/n)^{\nu(S^*)}}. \tag{6}$$

This fact shows that in order to compute the MLE of HSM (2) by IPF, it suffices to apply IPF to each marginal model $p(\boldsymbol{i}_{C^*})$ $C^* \in \mathcal{C}^*$. Then the computational cost for a update is reduced to $O(\sum_{C^* \in \mathcal{C}^*} |\mathcal{I}_{C^*}|)$.

2.2 Information propagation algorithm for HSM

In this section we assume that an HSM (2) satisfies $\mathcal{S}^* = \emptyset$. Such an HSM is called prime. For a prime HSM, we can directly apply the information propagation algorithm of [1]. Let $\bar{G}_\Delta$ be a chordal extension of conditional independence model G_Δ. Let $\bar{\mathcal{C}}$ and $\bar{\mathcal{S}}$ be the set of maximal cliques and minimal vertex separators of $\bar{G}_\Delta$. Let $\bar{\mathcal{T}}_C$ be a directed clique tree of G_Δ with a root $\bar{C}$. Then the algorithm is described as follows.

Algorithm 2 (Updating rule for localized IPS at time $t+1$)**.**

Step 1 For $D \in \Delta$ and k, choose any $\bar{C} \in \bar{\mathcal{C}}$ satisfying $D \subset \bar{C}$.

Step 2 Update $p(\boldsymbol{i}_{\bar{C}})$, $\boldsymbol{i} \in \mathcal{I}$ by

$$p^{(t+1)}(\boldsymbol{i}_{\bar{C}}) \leftarrow \frac{\sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) x(\boldsymbol{i}_D)}{n \sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) p^{(t)}(\boldsymbol{i}_D)} \cdot p^{(t)}(\boldsymbol{i}_{\bar{C}})$$

Step 3 For all descendants $\bar{C}'$ of $\bar{C}$ on $\bar{\mathcal{T}}_C$, update $p(\boldsymbol{i}_{\bar{C}'})$ in the following way.

3-1. Assume that all ancestors of C' are already updated. From a parent $\bar{C}''$ of $\bar{C}'$, receive $p^{(t+1)}(\boldsymbol{i}_{\bar{S}})$, where $\bar{S} = \bar{C}' \cap \bar{C}''$.

3-2. Update $p(\boldsymbol{i}_{\bar{C}'})$ by

$$p^{(t+1)}(\boldsymbol{i}_{\bar{C}'}) \leftarrow \frac{\sum_{i_D \in I_D} p^{(t+1)}(\boldsymbol{i}_{\bar{S}})}{\sum_{i_D \in I_D} p^{(t)}(\boldsymbol{i}_{\bar{S}})} \cdot p^{(t)}(\boldsymbol{i}_{\bar{C}'}).$$

The computational cost of the above algorithm for updating $p(\boldsymbol{i})$ is $O(\sum_{\bar{C} \in \bar{\mathcal{C}}} |\mathcal{I}_{\bar{C}}|)$. In general $\sum_{\bar{C} \in \bar{\mathcal{C}}} |\mathcal{I}_{\bar{C}}| < |\mathcal{I}|$.

Theorem 1. *The output of Algorithm 2 is the same as the output of the updating rule (3).*

Proof. $p^{(t)}(\boldsymbol{i})$ is written by

$$p^{(t)}(\boldsymbol{i}) = \frac{\prod_{\bar{C} \in \bar{\mathcal{C}}} p^{(t)}(\boldsymbol{i}_{\bar{C}})}{\prod_{\bar{S} \in \bar{\mathcal{S}}} p^{(t)}(\boldsymbol{i}_{\bar{S}})}.$$

The updating rule for $p^{(t)}(\boldsymbol{i}_{\bar{C}})$ in Step 2 is

$$p^{(t+1)}(\boldsymbol{i}_{\bar{C}}) = \frac{\sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) x(\boldsymbol{i}_D)}{n \sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) p^{(t)}(\boldsymbol{i}_D)} \cdot p^{(t)}(\boldsymbol{i}_{\bar{C}}),$$

The updating rule for $p^{(t)}(\boldsymbol{i}_{\bar{C}'})$ in Step 3 is

$$p^{(t+1)}(\boldsymbol{i}_{\bar{C}'}) \leftarrow \frac{\sum_{i_D \in I_D} p^{(t+1)}(\boldsymbol{i}_{\bar{S}})}{\sum_{i_D \in I_D} p^{(t)}(\boldsymbol{i}_{\bar{S}})} \cdot p^{(t)}(\boldsymbol{i}_{\bar{C}'}).$$

Noting that

$$p^{(t+1)}(\boldsymbol{i}_S) = \frac{p^{(t+1)}(\boldsymbol{i}_S)}{p^{(t)}(\boldsymbol{i}_S)} p^{(t)}(\boldsymbol{i}_S),$$

we obtain

$$p^{(t+1)}(\boldsymbol{i}) = \frac{\sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) x(\boldsymbol{i}_D)}{n \sum_{i_D \in I_D} \phi_D^k(\boldsymbol{i}_D) p^{(t)}(\boldsymbol{i}_D)} \cdot p^{(t)}(\boldsymbol{i}).$$

□

References

[1] J. H. Badsberg and F. M. Malvestuto. An implementaition of the iterative proportional fitting procecure by propagation trees. *Comput. Statist. Data. Anal.*, 37:297–322, 2001.

[2] Hisayuki Hara, Tomonari Sei, and Akimichi Takemura. Hierarchical subspace model for contingency tables. *Journal of Multivariate Analysis*, 103:19–34, 2010.

[3] Chihiro Hirotsu. Two-way change-point model and its application. *Australian Journal of Statistics*, 39(2):205–218, 1997.

[4] Søren Højsgaard. Split models for contingency tables. *Comput. Statist. Data. Anal.*, 42:621–645, 2003.

[5] Steffen L. Lauritzen. *Graphical Models*. Oxford University Press, Oxford, 1996.