Learning Mixed Multinomial Logits with Provable Guarantees

Yiqun Hu
David Simchi-Levi, MIT
Zhenzhen Yan, Nanyang Technological University

Introduction

Mixed Multinomial Logit (MMNL)

- Consider a set of $[m] = \{1, \ldots, m\}$ alternatives
- Population modeled by K MNL mixtures
- Each mixture k:
  - shares the same utility V_{kj}, $j \in [m]$
  - exhibits the same logit model

$$q_{kj} = \frac{\exp(V_{kj})}{\sum_{i \in [m]} \exp(V_{ki})} \qquad \forall j \in [m]$$

  - associates with a mixture weight α_k
  - Also known as the `softmax` function

Mixed Multinomial Logit (MMNL)

- Assume linear utility given observed candidate feature vector $\boldsymbol{z}_j \in \mathbb{R}^d, \forall\, j \in [M]$

$$V_{kj} = \boldsymbol{\beta}_k^\top \boldsymbol{z}_j,\ \forall\, k$$

- *Aggregated choice probability*:

$$\mathbf{g} = \sum_{k=1}^{K} \alpha_k \boldsymbol{q_k} \in \mathbb{R}^m$$

where $q_{kj} := q_j(\boldsymbol{\beta}_k) = \frac{\exp(\boldsymbol{\beta}_k^\top \boldsymbol{z}_j)}{\sum_{i\in[m]} \exp(\boldsymbol{\beta}_k^\top \boldsymbol{z}_i)}$

- Goal: estimate $\alpha_k, \boldsymbol{\beta}_k, k = 1, \ldots, K$

Literature Review

Learning MMNL

- Heuristics
  - EM algorithm (Train 09′)
- Learning algorithms
  - Uniform 2-MNL (Chierichetti et al 18′)
  - Arbitrary 2-MNL (Tang 20′)
- Hybrid (convergence only at aggregated level)
  - Frank-Wolfe (Jagabathula et al 20′)

Stochastic Subregion Frank-Wolfe (SSRFW)

Problem Formulation

- Data assumption
  - Population of size N
  - For each time period $t = 1, \ldots, T$:
    - Observe historical decision for each decision maker i: $Y_i^{(t)} \in \mathbb{R}^M$ with $Y_{ij}^{(t)} = \mathbb{1}_{[i \text{ chose } j \text{ at time } t]}$
    - Compute ***observed share***: $y_j^{(t)} = \frac{1}{N} \sum_{i=1}^{N} Y_{ij}^{(t)}$
- Learning objective:

$$\min_{g \in \mathrm{Conv}(\overline{\mathcal{P}})} \mathcal{L}(\boldsymbol{g}; \boldsymbol{y}) \quad \equiv \quad \min_{g \in \mathrm{Conv}(\overline{\mathcal{P}})} \frac{1}{2} \sum_{t=1}^{T} \left\| \boldsymbol{y}^{(t)} - \boldsymbol{g} \right\|^2$$

where $\boldsymbol{g} = \sum_{k=1}^{K} \alpha_k \boldsymbol{q}_{\boldsymbol{k}}(\boldsymbol{\beta}_k)$

Stochastic Subregion Frank-Wolfe

- Let $\mathcal{P}$ be the set of all logit vectors given $z_j, \ \forall j$
- Construct a candidate set $\mathcal{Q} \subset \mathcal{P}$ of logit vectors
- Require $\exists\, \pi : [L] \to [K], \left\| \boldsymbol{q}_\ell - \boldsymbol{q}^*_{\pi(\ell)} \right\| \leq \epsilon, \forall\, q_\ell \in \mathcal{Q}$
  L is the number of elements in $\mathcal{Q}$
- Each of the extreme points of $\mathrm{Conv}(\mathcal{Q})$ is close to some ground truth $\boldsymbol{q}_k$

Score Matrix

For decision maker i (of type k)

- Historical data: $Y_i^{(t)} \in \mathbb{R}^M, t = 1, \cdots, T$
- Define $X_i^{(t)}, t = 1, \ldots, T$ i.i.d random variable with pmf $\boldsymbol{q}_k^*$
  $X_i^{(t)} = j$ if $Y_{ij}^{(t)} = 1$
- Compute the empirical CDF

$$F_T(x; i) = \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}_{\{X_i^{(t)} \leq x\}}, x \in [M]$$

Pairwise score (dissimilarity) between i and j

$$s(i, j) = ||F_T(x; i) - F_T(x; j)||_\infty$$

$\mathcal{Q}$ Construction Algorithm

$$\mathcal{Q} = \left\{ \boldsymbol{q}_\ell \,\middle|\, \boldsymbol{q}_\ell = \frac{1}{nT} \sum_{i \in I_\ell} \sum_{t=1}^{T} Y_i^{(t)} \right\}_{\ell=1,\ldots,L}$$

Input: score matrix S, number of subsamples L, subsample size n
Initialization: $\mathcal{Q} = \texttt{set}()$

for $\ell \leftarrow 1$ **to** L **do**
    Choose seed: $i \sim U(0, N)$
    Initiate: $I = \texttt{set}()$
    while $|I| \neq n$ **do**
        $j \leftarrow \texttt{random\_sample}([N] \setminus I)$
        Generate $u \sim U(0, 1)$
        if $u < p_{j|i}$ **then**
            $I.\texttt{add}(j)$
        end
    end
    Compute $\boldsymbol{q}_\ell = \frac{1}{nT} \sum_{i \in I} \sum_{t=1}^{T} Y_i^{(t)}$
    $\mathcal{Q}.\texttt{add}(\boldsymbol{q}_\ell)$
end
Output: $\mathcal{Q}$

The SSRFW Algorithm

Input: data $\boldsymbol{y}$, $\mathcal{Q}$

Initialization: $k = 0$; $\boldsymbol{\alpha}^{(0)} = [1]$, a random $\boldsymbol{g}^{(0)}$

while *stopping condition not met* **do**

$k \leftarrow k + 1$

Compute $\boldsymbol{q} = \underset{\boldsymbol{v} \in \mathrm{Conv}(\mathcal{Q})}{\arg\min} \left\langle \nabla \mathcal{L}\left(\boldsymbol{g}^{(k-1)}; \boldsymbol{y}\right), \boldsymbol{v} - \boldsymbol{g}^{(k-1)} \right\rangle$

$\rightarrow$ *support finding step*

Compute $\boldsymbol{\alpha}^{(k)} = \underset{\boldsymbol{\alpha} \in \Delta_k}{\arg\min}\, \mathcal{L}\left(\alpha_0^{(k)} \boldsymbol{g}^{(0)} + \sum_{s=1}^{k} \alpha_s^{(k)} \boldsymbol{q}^{(s)}\right)$

$\rightarrow$ *proportions update step*

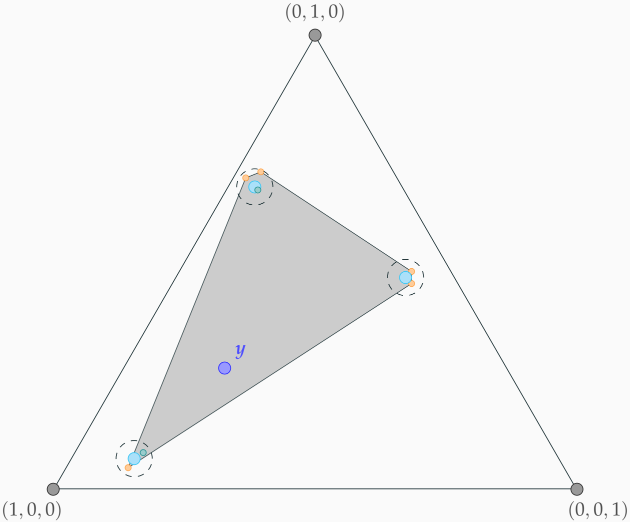
Update $\boldsymbol{g}^{(k)} := \alpha_0^{(k)} \boldsymbol{g}^{(0)} + \sum_{s=1}^{k} \alpha_s^{(k)} \boldsymbol{q}^{(s)}$

end

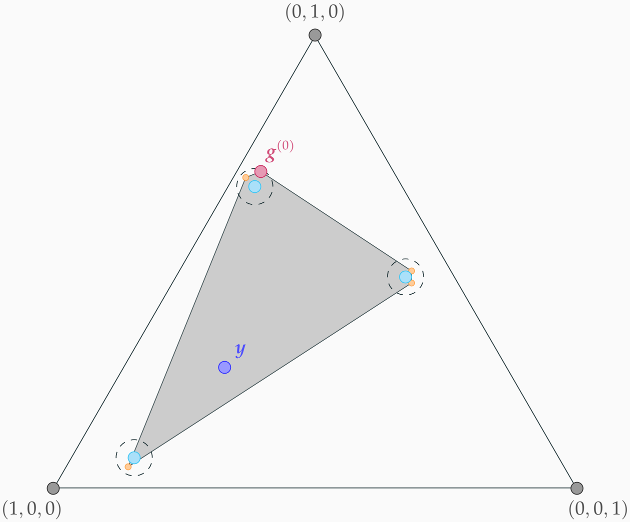
Output: choice prob. $\boldsymbol{q}^{(0)}, \ldots, \boldsymbol{q}^{(k)}$

mixture weights. $\boldsymbol{\alpha}^{(k)} \in \Delta_k \subset \mathbb{R}^{k+1}$

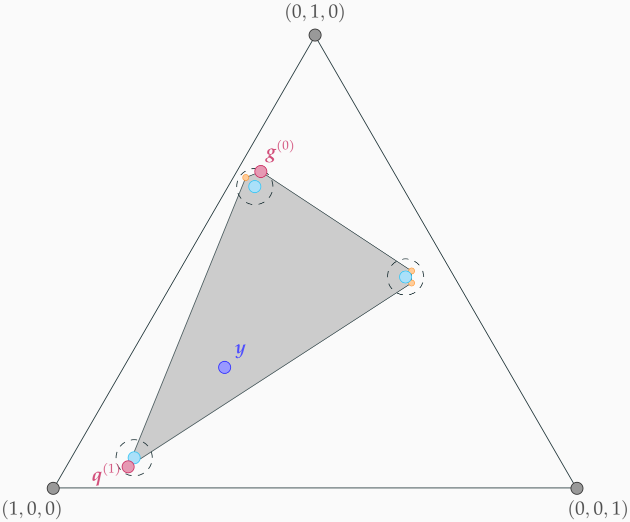
SSRFW Illustration



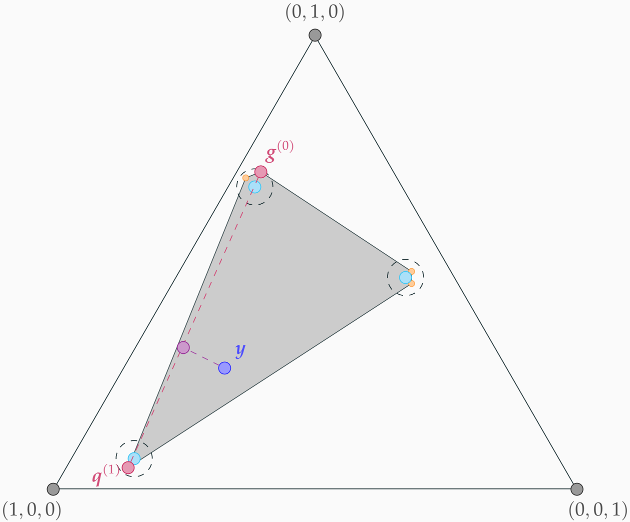
SSRFW Illustration



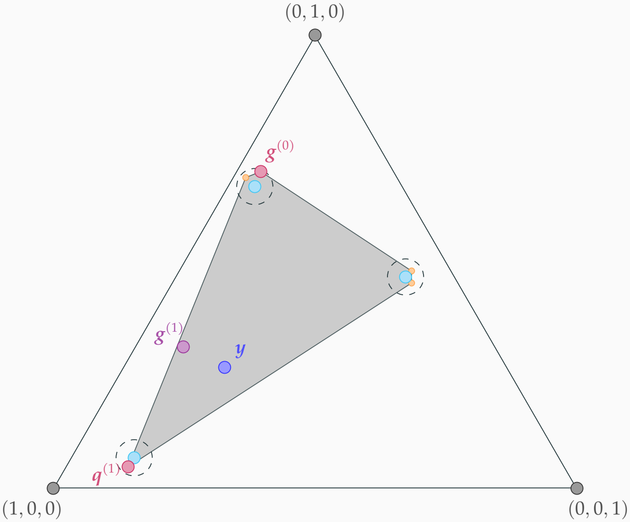
SSRFW Illustration



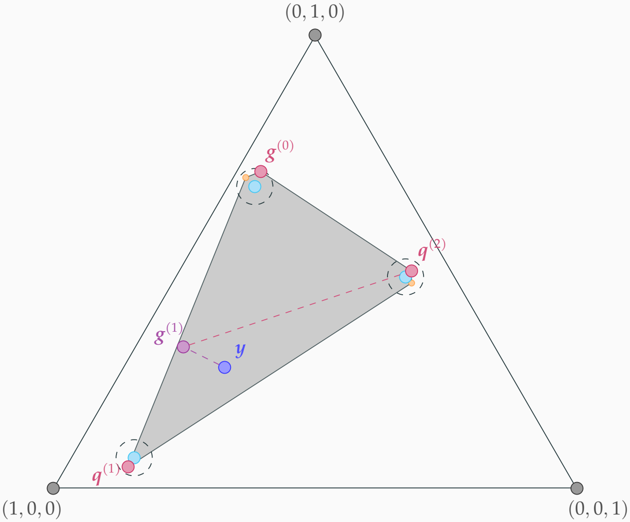
SSRFW Illustration



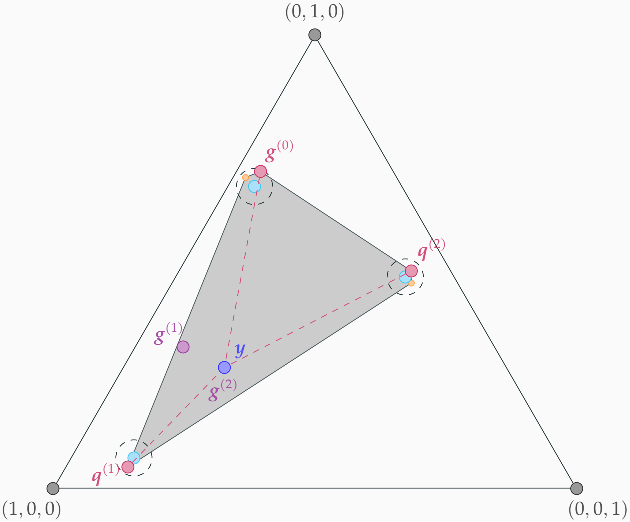
SSRFW Illustration



SSRFW Illustration



SSRFW Illustration



Advantages

- Utilize individual choice data
- No prior knowledge on the number of mixtures (K) needed - no model misspecification
- Provable convergence on the estimators based on the $\mathcal{Q}$ construction algorithm

Main Result

Theorem 1

Let $\mathbf{g} = \sum_{k=1}^{K} \alpha_k \boldsymbol{q}_k$ be a mixed multinomial logit (MMNL) model over a set of M items. Assume $M \geq K$. For any $\epsilon > 0$, $0 < \delta < 1$, `SSRFW` outputs an MMNL $\hat{\mathbf{g}} = \sum_{k=1}^{K'} \hat{\alpha}_k \hat{\boldsymbol{q}}_k$ where $K' \geq K$ such that, with probability $\geq 1 - \delta$, there exists a many-to-one mapping $\pi : j \mapsto i, j \in [K'], i \in [K]$ such that

$$\left\| \hat{\boldsymbol{q}}_j - \boldsymbol{q}_{\pi(j)} \right\| \leq \epsilon, \forall\, j \;\text{ and }\; \left| \sum_{j:\pi(j)=i} \hat{\alpha}_j - \alpha_i \right| \leq \epsilon, \forall\, i$$

The number of samples required $n(\epsilon, \delta)$ is polynomial in $\dfrac{1}{\epsilon}$ and $\dfrac{1}{\delta}$.

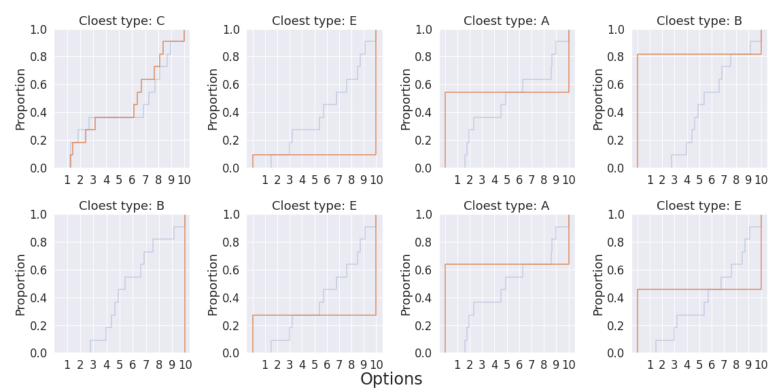
Numerical Experiments

Simulation Studies

Case Study on Nielsen Panel Data

$q^{(0)}, \ldots, q^{(K')}$ output from the algorithm

FW



Mixture Recovery

$\boldsymbol{q}^{(0)}, \ldots, \boldsymbol{q}^{(K')}$ output from the algorithm

Example Results (Nielsen Panel Data)

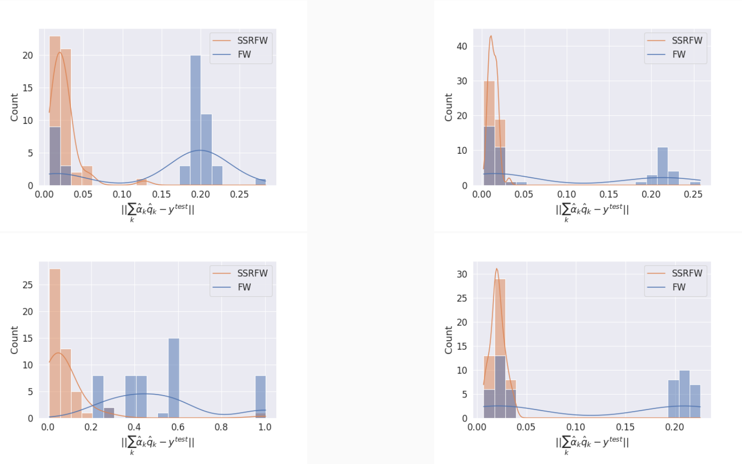


Figure 1: Categories: yogurt, pet food, candy, cereal

Conclusion

Conclusion

- Developed a new algorithm: `SSRFW`
  - based on the FW framework
  - utilize personal-level choice data
  - provide theoretical guarantees on the estimators
  - recover true model parameters
- Conducted various numerical experiments to compare `SSRFW` and the original FW
  - Simulation study to compare to the ground truth
  - Case studies on other Nielsen Consumer Panel Data

Thank You