Supporting Information for "Multiresolution categorical regression for interpretable cell type annotation"

Aaron J. Molstad
School of Statistics, University of Minnesota, Minneapolis, MN, U.S.A.

Keshav Motwani
Department of Biostatistics, University of Washington
Seattle, WA, U.S.A.

A Proofs from Section 4

For ease of display, we define $M_l = M(\mathcal{A}_l)$ and $D_l = D(a_l)$ for each $l \in [L]$. Recall that $\|\cdot\|_2$ denotes the Euclidean norm of a vector. Throughout, we use the notation $v_{\mathcal{A}_l} \in \mathbb{R}^{a_l}$ to denote the subvector of v with components indexed by the elements of $\mathcal{A}_l$.

Proof of Lemma 1. The result of Lemma 1 can be established by applying a similar series of arguments as the proof of Lemma 1 from the Supplementary Materials of Molstad and Rothman (2023). Recall, we are concerned with the solution to

$$\underset{\nu \in \mathbb{R}^K}{\arg\min} \left\{ \frac{1}{2} \|\nu - \eta\|_2^2 + \tilde{\gamma}\|\nu\|_2 + \tilde{\lambda} \sum_{l=1}^{L} w_l \|M_l \nu\|_2 \right\} \tag{12}$$

where M_l is as defined after (9) in the main text. Note that M_l is symmetric so that $M_l^\top = M_l$. We will first show that with $\hat{\nu}_{0,\tilde{\lambda}}$, the solution to (12) with $\tilde{\gamma} = 0$, we can obtain the solution to (12) through the equality

$$\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = \max\left(1 - \frac{\tilde{\gamma}}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2}, 0\right) \hat{\nu}_{0,\tilde{\lambda}}.$$

First, notice that the zero subgradient equation for (12) can be expressed

$$0 = \hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} - \eta + \tilde{\gamma} v + \tilde{\lambda} \sum_{l=1}^{L} M_l \phi^{(l)}$$

where $v = \hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} / \|\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}\|_2$ if $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} \neq 0$ and $\|v\|_2 \leq 1$ otherwise; and each $\phi^{(l)} = M_l \hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} / \|M_l \hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}\|_2$ if $M_l \hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} \neq 0$ and otherwise, $\|\phi^{(l)}\|_2 \leq 1$ with $\phi^{(l)}_k = 0$ and $\phi^{(l)} \in \mathbb{R}^K$ for all $k \notin \mathcal{A}_l$ otherwise.

Before we proceed, we make two notes about the $\phi^{(l)}$. First, note that here, the superscript does not represent an iteration counter (as in the main manuscript), but simply an index (i.e., we have L of the $\phi^{(l)} \in \mathbb{R}^K$). Second, note that whether $M_l \hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = 0$ or not, $\phi^{(l)}_k = 0$ for all $k \notin \mathcal{A}_l$, for each $l \in [L]$.

To proceed, note that the zero subgradient equation for (12) with $\tilde{\gamma} = 0$ is

$$0 = \hat{\nu}_{0,\tilde{\lambda}} - \eta + \tilde{\lambda} \sum_{l=1}^{L} M_l \tilde{\phi}^{(l)}$$

where each $\tilde{\phi}^{(l)} = M_l \hat{\nu}_{0,\tilde{\lambda}} / \ \|M_l \hat{\nu}_{0,\tilde{\lambda}}\|_2$ if $M_l \hat{\nu}_{0,\tilde{\lambda}} \neq 0$ and otherwise, $\|\tilde{\phi}^{(l)}\|_2 \leq 1$ with $\tilde{\phi}^{(l)}_k = 0$ for $k \notin \mathcal{A}_l$. We will show that these first order conditions imply those for (12) as long as $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}$ is as defined in Lemma 1.

- Suppose $\|\hat{\nu}_{0,\tilde{\lambda}}\|_2 > \tilde{\gamma}$. We then will show that $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = (1 - \tilde{\gamma}/\|\hat{\nu}_{0,\tilde{\lambda}}\|_2)\hat{\nu}_{0,\tilde{\lambda}}$ satisfies the first order conditions for (12). By definition of $\hat{\nu}_{0,\tilde{\lambda}}$,

$$\begin{aligned}
& 0 = \hat{\nu}_{0,\tilde{\lambda}} - \eta + \tilde{\lambda} \sum_{l=1}^{L} M_l \tilde{\phi}^{(l)} \\
\implies\ & 0 = \hat{\nu}_{0,\tilde{\lambda}} - \eta + \tilde{\lambda} \sum_{l=1}^{L} M_l \tilde{\phi}^{(l)} + (1 - \tilde{\gamma}/\|\hat{\nu}_{0,\tilde{\lambda}}\|_2)\hat{\nu}_{0,\tilde{\lambda}} - (1 - \tilde{\gamma}/\|\hat{\nu}_{0,\tilde{\lambda}}\|_2)\hat{\nu}_{0,\tilde{\lambda}} \\
\implies\ & 0 = -\eta + \tilde{\gamma} \frac{\hat{\nu}_{0,\tilde{\lambda}}}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} + \tilde{\lambda} \sum_{l=1}^{L} M_l \tilde{\phi}^{(l)} + \underbrace{(1 - \tilde{\gamma}/\|\hat{\nu}_{0,\tilde{\lambda}}\|_2)\hat{\nu}_{0,\tilde{\lambda}}}_{\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}} \\
\implies\ & 0 = \hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} - \eta + \tilde{\gamma} \frac{\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}}{\|\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}\|_2} + \tilde{\lambda} \sum_{l=1}^{L} M_l \tilde{\phi}^{(l)},
\end{aligned}$$

where $\frac{\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}}{\|\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}\|_2} = \frac{\hat{\nu}_{0,\tilde{\lambda}}}{\|\hat{\nu}_{0\tilde{\lambda}}\|_2}$ because $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} \propto \hat{\nu}_{0,\tilde{\lambda}}$ when $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = (1 - \tilde{\gamma}/\|\hat{\nu}_{0,\tilde{\lambda}}\|_2)\hat{\nu}_{0,\tilde{\lambda}}$. The final equality above is exactly the first order conditions for (12) as long as $\tilde{\phi}^{(l)}$ is a subgradient of $\|M_l\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}\|_2$ for each $l \in [L]$, i.e., $\phi^{(l)} = \tilde{\phi}^{(l)}$. However, this is immediate: if $M_l\hat{\nu}_{0,\tilde{\lambda}} \neq 0$, this implies $M_l\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = (1 - \tilde{\gamma}/\|\hat{\nu}_{0,\tilde{\lambda}}\|_2)(M_l\hat{\nu}_{0,\tilde{\lambda}}) \neq 0$, so $\phi^{(l)} = M_l\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}/\|M_l\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}}\|_2 = M_l\hat{\nu}_{0,\tilde{\lambda}}/\|M_l\hat{\nu}_{0,\tilde{\lambda}}\|_2 = \tilde{\phi}^{(l)}$; whereas if $M_l\hat{\nu}_{0,\tilde{\lambda}} = 0$, then by the same logic $M_l\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = 0$, so we may again take $\phi^{(l)} = \tilde{\phi}^{(l)}$ for all $l \in [L]$.

- Suppose $0 < \|\hat{\nu}_{0,\tilde{\lambda}}\|_2 \leq \tilde{\gamma}$. We will now show that this implies $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = 0$. Recall that the first order conditions for (12) when $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = 0$ are

$$0 = -\eta + \tilde{\gamma}v + \tilde{\lambda}\sum_{l=1}^{L} M_l\phi^{(l)}$$

for some $\|v\|_2 \leq 1$ and for some $\|\phi^{(l)}\|_2 \leq 1$ with $\phi_k^{(l)} = 0$ for $k \notin \mathcal{A}_l$ for each $l \in [L]$. Let $z_1 \geq 0$ be a scalar such that $1 = \frac{\tilde{\gamma}}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} - z_1$. By definition, there exists subgradients $\tilde{\phi}^{(l)}$ such that

$$\begin{aligned}
&0 = \hat{\nu}_{0,\tilde{\lambda}} - \eta + \tilde{\lambda}\sum_{l=1}^{L} M_l\tilde{\phi}^{(l)} \\
\implies\ &0 = \hat{\nu}_{0,\tilde{\lambda}}\left(\frac{\tilde{\gamma}}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} - z_1\right) - \eta + \tilde{\lambda}\sum_{l=1}^{L} M_l\tilde{\phi}^{(l)} \\
\implies\ &0 = \tilde{\gamma}\hat{\nu}_{0,\tilde{\lambda}}\left(\frac{1}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} - \frac{z_1}{\tilde{\gamma}}\right) - \eta + \tilde{\lambda}\sum_{l=1}^{L} M_l\tilde{\phi}^{(l)}
\end{aligned}$$

so that we need only argue that

$$\|\hat{\nu}_{0,\tilde{\lambda}}\|_2\left(\frac{1}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} - \frac{z_1}{\tilde{\gamma}}\right) \leq 1$$

in which case we could take $v = \hat{\nu}_{0,\tilde{\lambda}}\left(\frac{1}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} - \frac{z_1}{\tilde{\gamma}}\right)$ and $\tilde{\phi}^{(l)} = \phi^{(l)}$. Notice

$$\|\hat{\nu}_{0,\tilde{\lambda}}\|_2\left(\frac{1}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} - \frac{z_1}{\tilde{\gamma}}\right) = \left(1 - \frac{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2(\tilde{\gamma}/\|\hat{\nu}_{0,\tilde{\lambda}}\|_2 - 1)}{\tilde{\gamma}}\right) = \frac{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2}{\tilde{\gamma}},$$

which is no greater than one by assumption. Hence, for $v = \hat{\nu}_{0,\tilde{\lambda}}\left(\frac{1}{\|\hat{\nu}_{0,\tilde{\lambda}}\|_2} - \frac{z_1}{\tilde{\gamma}}\right)$, we have $\|v\|_2 \leq 1$ so that

$$\implies 0 = \tilde{\gamma}v - \eta + \tilde{\lambda}\sum_{l=1}^{L} M_l \tilde{\phi}^{(l)}.$$

Finally, we need to argue that $\|\tilde{\phi}^{(l)}\|_2 \leq 1$, in which case we could take $\tilde{\phi}^{(l)} = \phi^{(l)}$ for each $l \in [L]$. However, $\|\tilde{\phi}^{(l)}\|_2 \leq 1$ regardless of whether $M_l \hat{\nu}_{0,\tilde{\lambda}} = 0$ or not, so we can take $\phi^{(l)} = \tilde{\phi}^{(l)}$ for each $l \in [L]$.

- Suppose $\|\hat{\nu}_{0,\tilde{\lambda}}\|_2 = 0$. In this case, we have $0 = -\eta + \tilde{\lambda}\sum_{l=1}^{L} M_l \tilde{\phi}^{(l)}$ where $\|\tilde{\phi}^{(l)}\| \leq 1$ for each $l \in [L]$. Thus, taking $v = 0$, we also have $0 = -\eta + \tilde{\gamma}v + \tilde{\lambda}\sum_{l=1}^{L} M_l \tilde{\phi}^{(l)}$. Since $\|v\|_2 \leq 1$ and each $\|\tilde{\phi}^{(l)}\|_2 \leq 1$, these are exactly the first order conditions for (12) with $\hat{\nu}_{\tilde{\gamma},\tilde{\lambda}} = 0$.

Putting these three cases together completes the proof. ■

Proof of Theorem 1. Recall, based on the result of Lemma 1, we are concerned with computing

$$\underset{\nu \in \mathbb{R}^K}{\arg\min} \left\{ \frac{1}{2}\|\nu - \eta\|_2^2 + \tilde{\lambda}\sum_{l=1}^{L} w_l \|D_l \nu_{\mathcal{A}_l}\|_2 \right\}. \tag{13}$$

First, notice that because the $\mathcal{A}_l$ are nonoverlapping (by assumption) (13) can be solved separately over each set $\mathcal{A}_l$, so it suffices to focus on

$$\underset{\nu_l \in \mathbb{R}^{a_l}}{\text{minimize}} \left\{ \frac{1}{2}\|\nu_l - \eta_{\mathcal{A}_l}\|_2^2 + \tilde{\lambda} w_l \|D_l \nu_l\|_2 \right\}. \tag{14}$$

Let $\bar{\nu}_{l,\tilde{\lambda}}$ denote the argument minimizing (14). To solve 14, we consider its dual problem

$$\underset{\zeta \in \mathbb{R}^{a_l}}{\text{minimize}} \ \frac{1}{2}\|\eta_{\mathcal{A}_l} - D_l \zeta\|_2^2 \quad \text{subject to} \quad \|\zeta\|_2 \leq w_l \tilde{\lambda}, \tag{15}$$

which can be derived using a similar series of arguments as those used to derive (13) in Tibshirani and Taylor (2011). With a minimizer of (15), say $\hat{\zeta}$, the minimizer of (14) is given by $\eta_{\mathcal{A}_l} - D_l \hat{\zeta}$. Thus, we focus on (15). We consider two cases: (i) $\|(D_l^\top D_l)^- D_l^\top \eta_{\mathcal{A}_l}\|_2 \leq w_l \tilde{\lambda}$ and (ii) $\|(D_l^\top D_l)^- D_l^\top \eta_{\mathcal{A}_l}\|_2 > w_l \tilde{\lambda}$ where $(D_l^\top D_l)^-$ denotes the Moore-Penrose pseudoinverse of $D_l^\top D_l = D_l$. Note that because $(D_l^\top D_l)^- D_l^\top = D_l$, we could rewrite (i) as $\|D_l \eta_{\mathcal{A}_l}\|_2 \leq w_l \tilde{\lambda}$, and similarly for (ii).

If (i) holds, one solution to (15) is given by $\hat{\zeta} = (D_l^\top D_l)^- D_l^\top \eta_{\mathcal{A}_l} = D_l \eta_{\mathcal{A}_l}$ since the constraint in (15) is then satisfied by a global minimizer. Note that because D_l is rank deficient, the dual problem can have many solutions. When (i) holds, the set of solutions is given by $\{D_l\eta_{\mathcal{A}_l}+c\colon\ c\in\mathbb{R}^{a_l},\ \|D_l\eta_{\mathcal{A}_l}+c\|_2\leq w_l\tilde{\lambda},\ D_lc=0\}$. For any $\tilde{\zeta}$, an element of the set of solutions, we thus have

$$\bar{\nu}_{l,\tilde{\lambda}}=\eta_{\mathcal{A}_l}-D_l\tilde{\zeta}=\eta_{\mathcal{A}_l}-D_lD_l\eta_{\mathcal{A}_l}=\eta_{\mathcal{A}_l}-D_l\eta_{\mathcal{A}_l}=1_{a_l}(1_{a_l}^\top\eta_{\mathcal{A}_l})/a_l,$$

which is identical for all solutions $\tilde{\zeta}$. Of course, the primal problem is strictly convex, and has a unique solution. The primal problem is also strictly feasible, so strong duality holds. Thus, we conclude $\bar{\nu}_{l,\tilde{\lambda}}=1_{a_l}(1_{a_l}^\top\eta_{\mathcal{A}_l})/a_l$ is the minimizer of the primal problem when $\|D_l\eta_{\mathcal{A}_l}\|_2\leq w_l\tilde{\lambda}$.

Now let us consider (ii). Because $\|(D_l^\top D_l)^-D_l^\top\eta_{\mathcal{A}_l}\|_2> w_l\tilde{\lambda}$, we know that the argument minimizing (15) is not an unconstrained solution. This follows from the fact that $(D_l^\top D_l)^-D_l^\top\eta_{\mathcal{A}_l}$ is the minimizer of $g^\star$ with the minimum Euclidean norm among all minimizers. Hence, in this case, the dual problem can be expressed

$$\underset{\zeta\in\mathbb{R}^{a_l}}{\text{minimize}}\|\eta_{\mathcal{A}_l}-D_l\zeta\|_2^2\quad\text{subject to}\quad\|\zeta\|_2^2=w_l^2\tilde{\lambda}^2.\tag{16}$$

Because there is a one-to-one correspondence between ridge regression in its Lagrangian form and its constrained form (as above), we know the argument minimizing (16) corresponds to

$$\bar{\zeta}(\tau)=\arg\min_{\zeta\in\mathbb{R}^{a_l}}\|\eta_{\mathcal{A}_l}-D_l\zeta\|_2^2+\tau\|\zeta\|_2^2$$

for a choice of $\tau>0$ such that $\|\bar{\zeta}(\tau)\|_2^2=w_l^2\tilde{\lambda}^2$. Due to the fact that $\bar{\zeta}(\tau)=(D_l^\top D_l+\tau I_{a_l})^{-1}D_l^\top\eta_{\mathcal{A}_l}$, we need only determine τ such that

$$\eta_{\mathcal{A}_l}^\top D_l(D_l^\top D_l+\tau I_{a_l})^{-1}(D_l^\top D_l+\tau I_{a_l})^{-1}D_l^\top\eta_{\mathcal{A}_l}=\|(D_l^\top D_l+\tau I_{a_l})^{-1}D_l^\top\eta_{\mathcal{A}_l}\|_2^2=w_l^2\tilde{\lambda}^2.$$

Let $U\Psi U^\top=D_l$ be the eigendecomposition of $D_l^\top D_l=D_l$. By construction, Ψ is a diagonal matrix with nonnegative diagonal entries denoted $\psi_1,\dots,\psi_{a_l}$. Then, we must find a τ such that

$$\eta_{\mathcal{A}_l}^\top D_l(D_l^\top D_l+\tau I_{a_l})^{-1}(D_l^\top D_l+\tau I_{a_l})^{-1}D_l^\top\eta_{\mathcal{A}_l}=w_l^2\tilde{\lambda}^2$$

or restated, letting $v=U^\top\eta_{\mathcal{A}_l}\in\mathbb{R}^{a_l}$,

$$\eta_{\mathcal{A}_l}^\top U\Psi(\Psi+\tau I)^{-2}\Psi U^\top\eta_{\mathcal{A}_l}=w_l^2\tilde{\lambda}^2\iff\sum_{k=1}^{a_l}\frac{v_k^2\psi_k^2}{(\psi_k+\tau)^2}=w_l^2\tilde{\lambda}^2.$$

Next, letting $v_{-a_l} = (v_1, \ldots, v_{a_l-1})^\top$ and using that $\psi_j = 1$ for $j \in \{1, \ldots, a_l - 1\}$ and $\psi_{a_l} = 0$,

$$\sum_{k=1}^{a_l} \frac{v_k^2 \psi_k^2}{(\psi_k + \tau)^2} = w_l^2 \tilde{\lambda}^2 \iff \sum_{k=1}^{a_l - 1} \frac{v_k^2}{(1 + \tau)^2} = w_l^2 \tilde{\lambda}^2 \iff \tau = \frac{\|v_{-a_l}\|_2}{w_l \tilde{\lambda}} - 1.$$

It can be easily verified that $D_l(D_l^\top D_l + \tau)^{-1} D_l^\top = D_l/(\tau + 1)$, so plugging this back into our expression for $\bar{\nu}_{l,\tilde{\lambda}}$, we have

$$\begin{aligned}
\bar{\nu}_{l,\tilde{\lambda}} &= \eta_{\mathcal{A}_l} - D_l(D_l^\top D_l + \tau)^{-1} D_l^\top \eta_{\mathcal{A}_l} \\
&= \eta_{\mathcal{A}_l} - \eta_{\mathcal{A}_l}/(\tau + 1) + 1_{a_l}(1_{a_l}^\top \eta_{\mathcal{A}_l})/(\tau a_l + a_l) \\
&= \{1 - (\tau + 1)^{-1}\}\eta_{\mathcal{A}_l} + (\tau + 1)^{-1}(1_{a_l}^\top \eta_{\mathcal{A}_l}/a_l) 1_{a_l}.
\end{aligned}$$

Finally, using that $\|v_{-a_l}\|_2 = \|D_l \eta_{\mathcal{A}_l}\|_2$, which implies $\tau = \frac{\|v_{-a_l}\|_2}{w_l \tilde{\lambda}} - 1 = \frac{\|D_l \eta_{\mathcal{A}_l}\|_2}{w_l \tilde{\lambda}} - 1$, we conclude

$$\bar{\nu}_{l,\tilde{\lambda}} = \left(1 - \frac{w_l \tilde{\lambda}}{\|D_l \eta_{\mathcal{A}_l}\|_2}\right) \eta_{\mathcal{A}_l} + \frac{w_l \tilde{\lambda}}{\|D_l \eta_{\mathcal{A}_l}\|_2} (1_{a_l}^\top \eta_{\mathcal{A}_l}/a_l) 1_{a_l},$$

which completes the proof. ■

B Proofs and definitions from Section 5

B.1 Definition of $\mathcal{C}(\widetilde{\mathcal{S}}, \phi)$

First, we define the set $\mathcal{C}(\widetilde{\mathcal{S}}, \phi)$ which our restricted eigenvalue condition, **A2**, depends on. For completeness, let us remind the reader of the quantities needed for this definition. Recall that $\mathcal{S} \subset [p]$ is the set of predictors which are relevant (i.e., $\beta_{j,:}^\dagger \neq 0$ for $j \in \mathcal{S}$) and $\mathcal{S}^c = [p] \backslash \mathcal{S}$. By definition of $\beta^\dagger$, the kth predictor is irrelevant if $\beta_{k,:}^\dagger = 0$. Similarly, for each $\mathcal{A}_l$, the set $\mathcal{S}_l = \{k \in [p] : \beta_{k,\mathcal{A}_l}^\dagger \neq c 1_{a_l}$ for any $c \in \mathbb{R}\}$ is the set of predictors which distinguish between fine categories belonging to the lth coarse category. Consequently, $\mathcal{S}_l^c = [p] \setminus \mathcal{S}_l$, i.e., $\mathcal{S}_l^c$ is the set of predictors which do not distinguish between the fine categories in coarse category l in the sense that $\beta_{k,\mathcal{A}_l}^\dagger = c 1_{a_l}^\top$ for some $c \in \mathbb{R}$, including $c = 0$.

With these quantities, we are ready to define the set $\mathcal{C}(\widetilde{\mathcal{S}}, \phi)$ which will depend on constants $\phi = (\phi_1, \phi_2) \in (1, \infty) \times (0, \infty) =: \mathcal{T}$ and the collection of sets $\widetilde{\mathcal{S}} := \{\mathcal{S}, \mathcal{S}_1, \ldots, \mathcal{S}_L\}$.

Specifically, letting $\Delta_{\mathcal{S},:}$ be the submatrix of Δ consisting of the rows indexed by $\mathcal{S}$, define

$$\mathcal{C}(\widetilde{\mathcal{S}},\phi)=\Bigg\{\Delta\in\mathbb{R}^{p\times K}:\phi_1\phi_2\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2+(\phi_1+1)\|\Delta_{\mathcal{S},:}\|_{1,2}\ \geq$$
$$\phi_1\phi_2\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2+(\phi_1-1)\|\Delta_{\mathcal{S}^c,:}\|_{1,2}\Bigg\}.$$

B.2 Proof of Theorem 2

In order to prove the result from Section 5, we use essentially the same proof technique as in Molstad and Rothman (2023). First, we provide the key lemmas, which we prove in a later section. In this section, we define $\|A\|_{\infty,2}=\max_{j\in[a]}\|A_{j:}\|_2$ and $\|A\|_{1,2}=\sum_{j=1}^{a}\|A_{j:}\|_2$ where $\|A_{j:}\|_2$ is the Euclidean norm of the jth row of a matrix $A\in\mathbb{R}^{a\times b}$.

Lemma 2. *Suppose* ***C1****,* ***A1****, and* ***A2*** *hold. Let $\epsilon>0$, $c>2$, $\phi_1>1$, and $\phi_2>0$ be fixed constants. Define $\rho_1=c(\phi_1+1)$, $\rho_2=c\phi_1\phi_2$, $d_n=\sqrt{6}\max_{i\in[n]}\|x_i\|_2$, and*

$$\gamma=\frac{\phi_1\epsilon\kappa(\tilde{\mathcal{S}},\phi)}{\rho_1\sqrt{|\mathcal{S}|}+\rho_2\Psi_{\mathcal{A}}(\tilde{\mathcal{S}})}.$$

If $\gamma\geq\phi_1\|\nabla\mathcal{G}(\beta^\dagger)\|_{\infty,2}$, $\lambda=\phi_2\gamma$, and $d_n\epsilon$ is sufficiently close to zero such that $e^{d_n\epsilon}+d_n\epsilon-d_n^2\epsilon^2/c-1>0$, then $\|\hat{\beta}-\beta^\dagger\|_F\leq\epsilon$.

Lemma 3. *(Molstad and Rothman, 2023, Lemma 7) Under condition* ***C1*** *and assumption* ***A1****, for a given $\alpha\in(0,1)$*

$$\Pr\left\{\|\nabla\mathcal{G}(\beta^\dagger)\|_{\infty,2}\leq\sqrt{\frac{K}{4n}}+\sqrt{\frac{\log(p/\alpha)}{n}}\right\}\geq1-\alpha.$$

A complete proof of Lemma 3 can be found in Molstad and Rothman (2023). Finally, with these two lemmas in hand, we are able to prove the main result.

Proof of Theorem 2. Let

$$\epsilon=\frac{\rho_1\sqrt{|\mathcal{S}|}+\rho_2\Psi_{\mathcal{A}}(\widetilde{\mathcal{S}})}{\kappa(\widetilde{\mathcal{S}},\phi)}\left(\sqrt{\frac{K}{4n}}+\sqrt{\frac{\log(p/\alpha)}{n}}\right).$$

Then, define

$$\gamma = \phi_1 \left(\sqrt{\frac{K}{4n}} + \sqrt{\frac{\log(p/\alpha)}{n}} \right).$$

Taking $c = 3$ and assuming $d_n s^* \gamma \to 0$, it thus follows that for n sufficiently large, $e^{d_n \epsilon} + d_n \epsilon - d_n^2 \epsilon^2 / c - 1 > 0$ (since $d_n \epsilon \to 0$ is implied by $d_n s^* \gamma \to 0$), in which case Lemma 2 and Lemma 3 imply

$$\Pr \left\{ \|\hat{\beta} - \beta^\dagger\|_F \leq \frac{\rho_1 \sqrt{|\mathcal{S}|} + \rho_2 \Psi_{\mathcal{A}}(\widetilde{\mathcal{S}})}{\kappa(\widetilde{\mathcal{S}}, \phi)} \left(\sqrt{\frac{K}{4n}} + \sqrt{\frac{\log(p/\alpha)}{n}} \right) \right\}$$
$$\geq \Pr \left\{ \|\nabla \mathcal{G}(\beta^\dagger)\|_{\infty,2} \leq \sqrt{\frac{K}{4n}} + \sqrt{\frac{\log(p/\alpha)}{n}} \right\} \geq 1 - \alpha. \quad ■$$

Remark 1. *A more precise version of Theorem 2 could be obtained by requiring that $d_n \epsilon$ is sufficiently small such that $e^{d_n \epsilon} + d_n \epsilon - d_n^2 \epsilon^2 / c - 1 > 0$. For example, if we set $c = 3$, then as long as*

$$3 d_n \left[\frac{(\phi_1 + 1) \sqrt{|\mathcal{S}|} + \phi_1 \phi_2 \Psi_{\mathcal{A}}(\widetilde{\mathcal{S}})}{\kappa(\widetilde{\mathcal{S}}, \phi)} \right] \left(\sqrt{\frac{K}{4n}} + \sqrt{\frac{\log(p/\alpha)}{n}} \right) \in (0, 1.36),$$

it would follow that $\|\hat{\beta} - \beta^\dagger\|_F \leq \epsilon$ with probability at least $1 - \alpha$.

B.3 Proofs of lemmas

In order to prove the main lemma, Lemma 2, we need the following.

Lemma 4. *(Molstad and Rothman, 2023, Lemmas 3 and 4) With $d_n = \sqrt{6} \max_{i \in [n]} \|x_i\|_2$,*

$$\mathcal{G}(\beta^\dagger + \Delta) - \mathcal{G}(\beta^\dagger) \geq \mathrm{tr}\{\Delta^\top \nabla \mathcal{G}(\beta^\dagger)\} + \frac{\mathrm{vec}(\Delta)^\top \nabla^2 \mathcal{G}(\beta^\dagger) \mathrm{vec}(\Delta)}{d_n^2 \|\Delta\|_F^2} (e^{-d_n \|\Delta\|_F} + d_n \|\Delta\|_F - 1)$$

for any $\Delta \in \mathbb{R}^{p \times K}$.

Lemma 4 is a consequence of the 2-self concordance of the multinomial negative log-likelihood (Bach, 2010; Tran-Dinh et al., 2015). Next, we need a lemma which establishes that when γ is chosen appropriately, $\hat{\beta} - \beta^\dagger$ belongs to $\mathcal{C}(\widetilde{\mathcal{S}}, \phi)$.

Lemma 5. *If $\lambda = \phi_2 \gamma$ and $\gamma \geq \phi_1 \|\nabla \mathcal{G}(\beta^\dagger)\|_{\infty,2}$, then $\hat{\beta} - \beta^\dagger \in \mathcal{C}(\widetilde{\mathcal{S}}, \phi)$.*

We are now ready to prove the main lemma, Lemma 2.

Proof of Lemma 2. Let $\mathcal{F}_{\lambda,\gamma}$ denote the objective function from (3) (with all rows of β included in both penalties). Because $\mathcal{F}_{\lambda,\gamma}$ is convex and $\hat{\beta}$ is its minimizer, to establish the result it is sufficient to show that $\inf_{\Delta\in\mathcal{B}_{\epsilon,\phi}}\{\mathcal{F}_{\lambda,\gamma}(\beta^\dagger+\Delta)-\mathcal{F}_{\lambda,\gamma}(\beta^\dagger)\}>0$ where $\mathcal{B}_{\epsilon,\phi}=\{\Delta\in\mathbb{R}^{p\times K}:\|\Delta\|_F=\epsilon,\Delta\in\mathcal{C}(\widetilde{\mathcal{S}},\phi)\}$. For a proof of this fact, see Lemma 4 of Negahban et al. (2012). First, define $\mathcal{H}(\Delta)=\mathcal{F}_{\lambda,\gamma}(\beta^\dagger+\Delta)-\mathcal{F}_{\lambda,\gamma}(\beta^\dagger)$ so that

$$\begin{aligned}\mathcal{H}(\Delta)=&\underbrace{\mathcal{G}(\beta^\dagger+\Delta)-\mathcal{G}(\beta^\dagger)}_{T_1}+\underbrace{\gamma\|\beta^\dagger+\Delta\|_{1,2}-\gamma\|\beta^\dagger\|_{1,2}}_{T_2}\\&\underbrace{+\lambda\sum_{j=1}^{p}\sum_{l=1}^{L}\|D(a_l)(\beta_{j,\mathcal{A}_l}+\Delta_{j,\mathcal{A}_l})\|_2-\lambda\sum_{j=1}^{p}\sum_{l=1}^{L}\|D(a_l)\beta_{j,\mathcal{A}_l}\|_2}_{T_3}.\end{aligned}$$

We will bound each term T_1, T_2, and T_3. Starting with T_2, since $\beta^\dagger_{\mathcal{S}^c,:}=0$, we see that

$$\|\beta^\dagger+\Delta\|_{1,2}-\|\beta^\dagger\|_{1,2}=\|\beta^\dagger_{\mathcal{S},:}+\Delta_{\mathcal{S},:}\|_{1,2}+\|\Delta_{\mathcal{S}^c,:}\|_2-\|\beta^\dagger_{\mathcal{S},:}\|_{1,2}\geq\|\Delta_{\mathcal{S}^c,:}\|_{1,2}-\|\Delta_{\mathcal{S},:}\|_{1,2}$$

by the triangle inequality so that $T_2\geq\gamma\|\Delta_{\mathcal{S}^c,:}\|_{1,2}-\gamma\|\Delta_{\mathcal{S},:}\|_{1,2}$. Next, dealing with T_3, since $D(a_l)\beta_{k,\mathcal{A}_k}=0_{a_l}$ for $k\in\mathcal{S}_l^c$, by a similar argument,

$$\begin{aligned}&\sum_{j=1}^{p}\sum_{l=1}^{L}\|D(a_l)(\beta_{j,\mathcal{A}_l}+\Delta_{j,\mathcal{A}_l})\|_2-\sum_{j=1}^{p}\sum_{L=1}^{L}\|D(a_l)\beta_{j,\mathcal{A}_l}\|_2\\&\qquad=\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)(\beta_{k,\mathcal{A}_l}+\Delta_{k,\mathcal{A}_l})\|_2+\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2-\sum_{L=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\beta_{k,\mathcal{A}_l}\|_2\\&\qquad\geq\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2-\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2\end{aligned}$$

so that we have

$$T_3\geq\lambda\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2-\lambda\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2.$$

To bound T_1, we apply Lemma 4 which implies

$$T_1 \geq \operatorname{tr}\{\Delta^\top \nabla \mathcal{G}(\beta^\dagger)\} + \frac{\operatorname{vec}(\Delta)^\top \nabla^2 \mathcal{G}(\beta^\dagger)\operatorname{vec}(\Delta)}{d_n^2\|\Delta\|_F^2}(e^{-d_n\|\Delta\|_F} + d_n\|\Delta\|_F - 1).$$

Putting these three bounds together and using $\lambda = \phi_2\gamma$—along with the fact that $\operatorname{tr}\{\Delta^\top \nabla \mathcal{G}(\beta^\dagger)\} \geq -\|\Delta\|_{1,2}\|\nabla \mathcal{G}(\beta^\dagger)\|_{\infty,2}$ by Hölder's inequality—we have that

$$\begin{aligned}\mathcal{H}(\Delta) \geq &-\|\Delta\|_{1,2}\|\nabla \mathcal{G}(\beta^\dagger)\|_{\infty,2} + \frac{\operatorname{vec}(\Delta)^\top \nabla^2 \mathcal{G}(\beta^\dagger)\operatorname{vec}(\Delta)}{d_n^2\|\Delta\|_F^2}(e^{-d_n\|\Delta\|_F} + d_n\|\Delta\|_F - 1)\\ &+ \gamma\|\Delta_{\mathcal{S}^c,:}\|_{1,2} - \gamma\|\Delta_{\mathcal{S},:}\|_{1,2} + \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2 - \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2.\end{aligned}$$

Then, since we are assuming $\gamma \geq \phi_1\|\nabla \mathcal{G}(\beta^\dagger)\|_{\infty,2}$

$$\begin{aligned}\mathcal{H}(\Delta) \geq\ & -\|\Delta\|_{1,2}\frac{\gamma}{\phi_1} + \frac{\operatorname{vec}(\Delta)^\top \nabla^2 \mathcal{G}(\beta^\dagger)\operatorname{vec}(\Delta)}{d_n^2\|\Delta\|_F^2}(e^{-d_n\|\Delta\|_F} + d_n\|\Delta\|_F - 1)\\ &+ \gamma\|\Delta_{\mathcal{S}^c,:}\|_2 - \gamma\|\Delta_{\mathcal{S},:}\|_2 + \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2 - \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2\\ \geq\ & \frac{\operatorname{vec}(\Delta)^\top \nabla^2 \mathcal{G}(\beta^\dagger)\operatorname{vec}(\Delta)}{d_n^2\|\Delta\|_F^2}(e^{-d_n\|\Delta\|_F} + d_n\|\Delta\|_F - 1) + \gamma\frac{(\phi_1 - 1)}{\phi_1}\|\Delta_{\mathcal{S}^c,:}\|_2 \qquad (17)\\ &- \gamma\frac{(\phi_1+1)}{\phi_1}\|\Delta_{\mathcal{S},:}\|_2 + \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2 - \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2\end{aligned}$$

so that applying **A2**, i.e., $\frac{\operatorname{vec}(\Delta)^\top \nabla^2 \mathcal{G}(\beta^\dagger)\operatorname{vec}(\Delta)}{\|\Delta\|_F^2} \geq \kappa(\widetilde{\mathcal{S}}, \phi)$ for all $\Delta \in \mathcal{B}_{\epsilon,\phi}$, we further have

$$\begin{aligned}\mathcal{H}(\Delta) \geq\ & \frac{\kappa(\widetilde{\mathcal{S}}, \phi)}{d_n^2}(e^{-d_n\|\Delta\|_F} + d_n\|\Delta\|_F - 1) + \gamma\frac{(\phi_1 - 1)}{\phi_1}\|\Delta_{\mathcal{S}^c,:}\|_2\\ &- \gamma\frac{(\phi_1+1)}{\phi_1}\|\Delta_{\mathcal{S},:}\|_2 + \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2 - \phi_2\gamma\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2\\ \geq\ & \frac{\kappa(\widetilde{\mathcal{S}}, \phi)}{d_n^2}(e^{-d_n\|\Delta\|_F} + d_n\|\Delta\|_F - 1) - \gamma\left(\frac{(\phi_1+1)}{\phi_1}\|\Delta_{\mathcal{S},:}\|_2 - \phi_2\sum_{l=1}^{L}\sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2\right).\end{aligned}$$

Using $\|\Delta_{\mathcal{S},:}\|_{1,2}\leq \sqrt{|\mathcal{S}|}\|\Delta\|_F$ and $\Psi_{\mathcal{A}}(\widetilde{\mathcal{S}})\|\Delta\|_F\geq \sum_{l=1}^L \sum_{k\in\mathcal{S}_l}\|D(a_l)\Delta_{k,\mathcal{A}_l}\|_2$, it follows from the above that with $\|\Delta\|_F= \epsilon$,

$$\mathcal{H}(\Delta) \geq \frac{\kappa(\widetilde{\mathcal{S}},\phi)}{d_n^2}(e^{-d_n\epsilon}+d_n\epsilon-1)-\gamma\epsilon\left\{\frac{(\phi_1+1)}{\phi_1}\sqrt{|\mathcal{S}|}+\phi_2\Psi_{\mathcal{A}}(\widetilde{\mathcal{S}})\right\}.$$

Thus, by taking

$$\gamma = \frac{\phi_1\epsilon\kappa(\widetilde{\mathcal{S}},\phi)}{c(\phi_1+1)\sqrt{|\mathcal{S}|}+c\phi_1\phi_2\Psi_{\mathcal{A}}(\widetilde{\mathcal{S}})},$$

it follows that

$$\mathcal{H}(\Delta) \geq \frac{\kappa(\widetilde{\mathcal{S}},\phi)}{d_n^2}(e^{-d_n\epsilon}+d_n\epsilon-1)-\frac{\epsilon^2\kappa(\widetilde{\mathcal{S}},\phi)}{c} = \frac{\kappa(\widetilde{\mathcal{S}},\phi)}{d_n^2}\left(e^{-d_n\epsilon}+d_n\epsilon-\frac{d_n^2\epsilon^2}{c}-1\right)$$

which is positive—thus establishing the result—if

$$\left(e^{-d_n\epsilon}+d_n\epsilon-\frac{d_n^2\epsilon^2}{c}-1\right)>0,$$

which will occur when $d_n\epsilon$ is sufficiently close to zero. ■.

Proof of Lemma 5. Since $\mathcal{F}_{\gamma,\lambda}$ is convex and $\hat{\beta}$ is its minimizer, we know that $\mathcal{H}(\hat{\Delta})\leq 0$ where $\hat{\Delta}=\hat{\beta}-\beta^{\dagger}$. Thus, by the inequality in (17),

$$\begin{aligned}0 \geq \mathcal{H}(\hat{\Delta}) \geq \gamma\frac{(\phi_1-1)}{\phi_1}\|\hat{\Delta}_{\mathcal{S}^c,:}\|_{1,2} &- \gamma\frac{(\phi_1+1)}{\phi_1}\|\hat{\Delta}_{\mathcal{S},:}\|_{1,2}\\ &+\phi_2\gamma\sum_{l=1}^L\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\hat{\Delta}_{k,\mathcal{A}_l}\|_2-\phi_2\gamma\sum_{l=1}^L\sum_{k\in\mathcal{S}_l}\|D(a_l)\hat{\Delta}_{k,\mathcal{A}_l}\|_2\end{aligned}$$

from which it follows that

$$\begin{aligned}\gamma\frac{(\phi_1+1)}{\phi_1}\|\hat{\Delta}_{\mathcal{S},:}\|_{1,2}&+\phi_2\gamma\sum_{l=1}^L\sum_{k\in\mathcal{S}_l}\|D(a_l)\hat{\Delta}_{k,\mathcal{A}_l}\|_2\\ &\geq \gamma\frac{(\phi_1-1)}{\phi_1}\|\hat{\Delta}_{\mathcal{S}^c,:}\|_{1,2}+\phi_2\gamma\sum_{l=1}^L\sum_{k\in\mathcal{S}_l^c}\|D(a_l)\hat{\Delta}_{k,\mathcal{A}_l}\|_2,\end{aligned}$$

or, restated,

$$(\phi_1 + 1)\|\hat{\Delta}_{\mathcal{S},:}\|_{1,2} + \phi_1\phi_2 \sum_{l=1}^{L} \sum_{k\in\mathcal{S}_l} \|D(a_l)\hat{\Delta}_{k,\mathcal{A}_l}\|_2$$

$$\geq (\phi_1 - 1)\|\hat{\Delta}_{\mathcal{S}^c,:}\|_{1,2} + \phi_1\phi_2 \sum_{l=1}^{L} \sum_{k\in\mathcal{S}_l^c} \|D(a_l)\hat{\Delta}_{k,\mathcal{A}_l}\|_2. \quad \blacksquare$$

C Algorithms

In this section, we present formal statements of the algorithms described in Section 4.

First, we present the accelerated proximal gradient descent algorithm for computing the estimator in Algorithm 1. How to solve Step 4.1.2 is discussed in Sections 4.2 and 4.3. In the case of overlapping coarse categories, we can solve Step 4.1.2 of Algorithm 1 using Algorithm 2. Note that we can actually solve the proximal operator with overlapping coarse categories more efficiently than as described in Algorithm 2. Specifically, if any $\mathcal{A}_l$ do not intersect with any other $\mathcal{A}_{l'}$ (including $\mathcal{A}_l$ which are singletons), then we can solve for $\nu_{\mathcal{A}_l}$ from (8) using exactly the expression from Theorem 1. overlap can be

Recovery	p	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Exact	100	0.9933	0.8067	0.6156	0.4594	0.3439	0.5072
	200	0.9961	0.8033	0.6144	0.4706	0.3506	0.5533
	500	0.9928	0.8078	0.5994	0.4500	0.3994	0.5861
	1000	0.9961	0.8089	0.6144	0.4561	0.4072	0.6289
Partial	100	0.9933	0.9728	0.9439	0.9306	0.9528	0.9683
	200	0.9961	0.9683	0.9428	0.9339	0.9539	0.9694
	500	0.9928	0.9733	0.9217	0.9106	0.9522	0.9628
	1000	0.9961	0.9722	0.9411	0.9044	0.9350	0.9611

Table 1: The average proportion of important predictors where the true structure was estimated exactly ("Exact"), and the average proportion of important predictors where the true structure was coarser (or identical) to the estimated structure ("Partial").

Algorithm 1 Accelerated proximal gradient descent for computing (3)

```
Initialize $\beta^{(0)} = \beta^{(1)} \in \mathbb{R}^{p \times K}$, $\alpha^{(0)} = \alpha^{(1)} = 1$, $\tau^{(1)} = \tau_0 > 0$, and $L_0 = \|X\|_F^2 \sqrt{K}/n$
  0. Set $t = 1$ and proceed to 1
  1. Set $\Gamma = \beta^{(t)} + \{(\alpha^{(t-1)} - 1)/\alpha^{(t)}\}(\beta^{(t)} - \beta^{(t-1)})$
  2. Set $\eta = \Gamma - \tau^{(t)} \nabla \mathcal{G}(\Gamma)$
  3. Set $\bar{\beta}_{1,:} = \eta_{1,:}$
  4. For $j \in \{2, \dots, p\}$ in parallel
        4.1. If $\|\eta_{j,:}\|_2 \leq \tau^{(t)}\gamma$
                4.1.1. Set $\bar{\beta}_{j,:} = 0$
             Else
                4.1.2. Compute $\nu_j = \arg\min_{\nu \in \mathbb{R}^K} \{\frac{1}{2}\|\eta_{j,:} - \nu\|_2^2 + \tau^{(t)} \lambda \sum_{l=1}^{L} w_l \|D(a_l)\nu_{\mathcal{A}_l}\|_2\}$
                4.1.3. Set $\bar{\beta}_{j,:} = \max(1 - \tau^{(t)}\gamma/\|\nu_j\|_2, 0)\nu_j$
  5. If $\mathcal{G}(\bar{\beta}) \leq \mathcal{G}(\Gamma) + \mathrm{tr}\{\nabla \mathcal{G}(\Gamma)^\top (\bar{\beta} - \Gamma)\} + \|\bar{\beta} - \Gamma\|_F^2 / 2\tau^{(t)}$
        5.1. Set $\beta^{(t+1)} = \bar{\beta}$, $\alpha^{(t+1)} = \frac{1 + \sqrt{1 + 4(\alpha^{(t)})^2}}{2}$, $\tau^{(t+1)} = \tau^{(t)}$, $t = t + 1$ and return to 1
     Else
        5.2. Replace $\tau^{(t)} = \max(\tau^{(t)}/2, 1/L_0)$ and return to 2
  6. If relative change in objective function is less than $\epsilon$ for last three iterations, terminate
     Else, return to 1
```

D Additional results from Section 6

D.1 Results with alternative performance metrics

In Figures 7 and 8, we display Kullback-Leibler divergences and classification errors, respectively, for the simulation settings described in Section 6. In Figure 7, we see that all relative performances essentially mirror those from Figures 2 of the main manuscript. We discuss results presented in Figure 8 in the Section D.2.

D.2 Comparison to additional competitors

In this section, we compare the methods considered in the main manuscript to additional competitors. Specifically, we compare to the multiclass sparse discriminant analysis method (`MSDA`) proposed by Mai et al. (2019), to "vanilla" random forests (`RF`), and hierarchical random forests (`HRF`) proposed by Kaymaz et al. (2021). Because these methods do not estimate response category probabilities directly, we restrict our attention to a comparison in terms of classification accuracy.

Algorithm 2 Blockwise coordinate descent for solving (8) with overlapping $\mathcal{A}_l$

```
Initialize $(\zeta^{(1)}_{:,1}, \dots, \zeta^{(1)}_{:,L})$ such that $\zeta^{(1)}_{:,l}$ belongs to the feasible set for (9) for each $l \in [L]$
  0. Set $\tilde{\eta} = \eta - \sum_{j=1}^{L} M(\mathcal{A}_l)\zeta_l^{(1)}$, $r = 1$, and proceed to 1
  1. For $l \in \{1, 2, \dots, L\}$ in order
       1.1. Update $\tilde{\eta} = \tilde{\eta} + M(\mathcal{A}_l)\zeta_l^{(r)}$
       1.2. Compute $\zeta_l^{(r+1)}$ according to (11)
       1.3. Update $\tilde{\eta} = \tilde{\eta} - M(\mathcal{A}_l)\zeta_l^{(r+1)}$
  2. If not converged, set $r = r + 1$ and return to 2
     Else, return the solution to (9), $\hat{\nu}_{0,\tilde{\lambda}} = \eta - \sum_{l=1}^{L} M(\mathcal{A}_l)\zeta_l^{(r+1)}$
```

In Figure 8, we present this comparison under the data generating models considered in Section 6. We see that in general, the methods relying on the multinomial logistic regression model perform best, though MSDA can perform better than the L_1-penalized multinomial logistic regression estimator in certain scenarios. The two random forest variants always perform worse than the other competitors. When p is small, RF outperforms HRF, though when few predictors distinguish between coarse categories and $p = 1000$, we see HRF significantly outperform RF.

D.3 Support and effect resolution recovery

A fundamentally important aspect of our estimator is the degree to which it recovers both the relevant set of predictors, as well as the resolution at which the relevant predictors affect the response category probabilities. To assess this question, we provide results on effect resolution recovery in Table 1, and variable selection accuracy in Table 2. In Table 1, we provide the average proportion of important predictors where the true structure was estimated exactly ("exact"), and the average proportion of important predictors where the true structure was coarser (or identical) to the estimated structure ("partial"). The latter is interesting because even if we do not recover the blockwise (effect resolution) structure exactly, we can often approximate it well without exact equality. For example, in one simulation scenario, the row for a particular important predictor, denoted by k, was

$\beta^*_{k,:} =$ 1.903	1.903	1.903	-3.168	-3.168	-3.168	1.248	1.248	1.248	0.018	0.018	0.018
$\widehat{\beta}_{k,:} =$ 0.802	0.802	0.802	**-1.222**	**-1.171**	**-1.173**	0.294	0.294	0.294	0.093	0.093	0.093

In the example above, we see that our estimate nearly recovers the blockwise structure, but for the fourth through sixth coefficients, fails to enforce exactly equality. In Table 1, this would not count as an "exact" recovery, but it would count as a "partial" recovery since our method estimates the effects to occur at a resolution finer than the truth. Notable about this estimate—and many others we observed to be "partial" recoveries—is that coefficients which do not have the exact blockwise equality of the truth have approximately the blockwise structure. In the example above, in practice $\widehat{\beta}_{k,:}$ will have an effect nearly indistinguishable from a version with coefficients four through six being exactly equal.

Note that if we failed to include a relevant predictor in the model, this would count as neither "exact" nor "partial" recovery.

We also provide true positive (TPR) and true negative (TNR) variable selection rates, as well as model size (defined as the total number of distinct coefficients estimated by each method) in Table 2. We see that our method, `mrMLR`, and the group lasso penalized multinomial logistic regression estimator, `Group`, have nearly identical TPR and TNR. However, as the number of predictors effecting response categories at a coarse resolution increases, (i.e., going from Model 1 to Model 6), we see that the model sizes begin to differ substantially. In particular, by Model 6, where only 3 of the 18 relevant predictors affect the response category probabilities at the finest resolution, our method estimates near half as many distinct coefficients as `Group`. This partially explains the improvement in efficiency our method provides.

		Model 1		Model 2		Model 3		Model 4		Model 5		Model 6	
		`Group`	`mrMLR`	`Group`	`mrMLR`	`Group`	`mrMLR`	`Group`	`mrMLR`	`Group`	`mrMLR`	`Group`	`mrMLR`
TPR	100	1.000	1.000	1.000	0.999	1.000	1.000	0.999	0.999	0.997	0.998	0.995	0.997
	200	1.000	1.000	0.998	0.998	0.997	0.999	0.996	0.998	0.991	0.997	0.987	0.994
	500	1.000	1.000	0.999	0.999	0.994	0.996	0.990	0.993	0.987	0.994	0.972	0.987
	1000	0.999	0.999	0.996	0.997	0.995	0.996	0.988	0.992	0.979	0.992	0.967	0.986
TNR	100	0.203	0.203	0.189	0.185	0.197	0.178	0.194	0.182	0.201	0.168	0.218	0.229
	200	0.384	0.381	0.378	0.372	0.384	0.373	0.385	0.362	0.403	0.343	0.428	0.414
	500	0.612	0.609	0.612	0.608	0.617	0.608	0.620	0.600	0.632	0.584	0.652	0.622
	1000	0.740	0.738	0.739	0.737	0.739	0.728	0.747	0.738	0.755	0.716	0.778	0.750
Model size	100	1000.2	997.0	1013.9	979.2	1005.7	926.8	1008.6	783.8	1001.5	741.5	984.7	509.1
	200	1560.8	1563.4	1574.8	1527.4	1559.9	1409.0	1557.7	1215.6	1517.5	1171.1	1461.6	751.7
	500	2460.2	2463.2	2457.7	2377.5	2427.8	2174.8	2409.8	1886.4	2344.0	1795.4	2219.9	1202.4
	1000	3282.4	3289.6	3289.3	3188.3	3285.2	3035.6	3189.6	2481.3	3102.2	2330.4	2827.3	1554.6

Table 2: Average true positive rate, true negative rates, and total number of distinct coefficients estimated by both `Group` and `mrMLR` under the data generating models described in Section 6.

D.4 Computing times

In Tables 3 and 4, we present averages of the time taken to compute the entire solution path in the simulation study scenarios. Note that we compute the solution path over both pairs of tuning parameters (γ, λ), of which we consider a grid of 100×10 (i.e., this is the time to compute our estimator roughly 1000 times). Also note that we use extremely strict convergence tolerance ϵ[1] in the simulation studies and the real data analysis. Every simulation study replicate required 2GB of memory or less and was performed on a single core on HiperGator 3.0 at the University of Florida (`https://www.rc.ufl.edu/about/hipergator`).

ϵ	p	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
10^{-9}	100	6.10	9.02	12.24	13.93	13.03	12.72
	200	9.58	12.19	11.12	17.85	14.96	15.94
	500	11.01	19.78	15.58	28.06	31.42	36.55
	1000	18.06	34.32	32.13	49.62	67.77	80.96
10^{-7}	100	2.68	4.22	5.54	6.34	5.81	5.44
	200	4.61	5.80	5.35	8.39	6.93	7.27
	500	5.76	10.12	7.82	14.00	15.09	17.42
	1000	9.45	17.50	16.37	25.34	33.92	39.63
10^{-5}	100	1.69	2.55	3.38	3.90	3.47	3.22
	200	2.84	3.51	3.24	5.06	4.09	4.17
	500	3.48	5.95	4.58	8.00	8.47	9.57
	1000	5.51	10.27	9.93	14.62	17.04	20.14

Table 3: Average time (in minutes) to compute the entire solution path over 100×10 candidate tuning parameter pairs (γ, λ) in the nonoverlapping simulation study scenarios. Note that these times represent computation on a single core with 2GB memory.

p	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
100	46.24	69.50	72.06	75.10	75.35	77.35
200	68.70	107.95	107.90	114.20	100.38	97.45
500	99.73	163.14	204.94	242.51	194.73	161.60
1000	175.41	289.16	292.32	385.86	422.03	316.73

Table 4: Average time (in minutes) to compute the entire solution path over 100×10 candidate tuning parameter pairs (γ, λ) in the simulation study scenarios described in Section D.5 with $\epsilon = 10^{-9}$.

[1] We claim convergence when the objective function changes less than ϵ% for three consecutive iterations.

We were able to run our analysis of the single-cell data from Hao et al. (2021) (Section 7) on cores with 8GB of memory on HiperGator 3.0 at the University of Florida. Only when $n = 50000$ or $p \geq 1500$ did we have have to increase the memory to 12GB per core. Thus, in principle, these analyses could be performed on most modern laptop computers. Computing times in the real data analyses were highly dependent on the number of candidate tuning parameters and convergence tolerance. Fitting a 100×5 grid of candidate tuning parameters with $\epsilon = 10^{-7}$ took 40 hours on average (over 50 replicates) with $n = 20000$ and $p = 500$. To achieve the same high-accuracy solution with `glmnet` took 10 hours on average for the same replicates. This makes sense since `glmnet` is solving the same problem our algorithm solves with a 100×1 grid when $\lambda = 0$, so the computing time of our algorithm is roughly in line with the state-of-the-art on this problem.

The authors intend to continually update the R package `HierMultinom`, so one could expect these times will decrease as new versions are released. Please visit `https://github.com/ajmolstad/HierMultinom` to download the most recent version.

Our method can require long computing times. This is a consequence of working with the multinomial negative log-likelihood. The computation of the gradient, which is required in every iteration of our algorithm, has calculations requiring $O(Knp)$ and $O(n^2K)$ flops where n is the number of cells in the training data, K the number of response categories, and p the number of candidate predictors. However, in practice, there are many ways one could significantly reduce the computing time, were this a concern. These include

- **Relaxing the accuracy of solution.** As shown in Table 3, a user can relax the convergence tolerance and expect far shorter computing times. We found that in general, the performance was relatively unaffected as long as $\epsilon \leq 10^{-5}$: we provide evidence of this in Figure 9.

- **Parallelization.** In the simulation studies and real data analyses, we computed the entire solution path on a single core. However, this computation could be easily parallelized. Our code is structured so that one computes a solution path across all candidate γ with λ held fixed. Thus, when we compute the solution path for 100×10 candidate tuning parameters, if we have 10 cores available to us, each 100×1 solution path could be computed in parallel. This would correspond roughly to dividing each of the computing times in the table by 10. In practice, we expect practitioners would implement this approach in any real application, so rather than taking 40 hours, it would take closer to 8 hours.

- **Number of candidate tuning parameters.** Because timely computation was not our main concern in the simulation studies, we considered a large number of candidate tuning parameters. One could easily consider far fewer if computing time is a priority.

D.5 Overlapping $\mathcal{A}_l$

In this subsection, we perform a similar set of simulation settings as in Section 6, but with $\mathcal{A}_1 = \{1, 2, 3\}$, $\mathcal{A}_2 = \{4, 5, 6\}$, $\mathcal{A}_3 = \{7, 8, 9\}$, $\mathcal{A}_4 = \{10, 11, 12\}$, and $\mathcal{A}_5 = \{1, 2, \dots, 6\}$. Just as in the nonoverlapping case considered in the main manuscript, we consider six models for β^*, Model 1–6. For each model, we first select 18 important predictors, then randomly select s of the important predictors only distinguish between the coarse categories defined by $\mathcal{A}_3$, $\mathcal{A}_4$, and $\mathcal{A}_5$. The remaining $18 - s$ are useful for distinguishing between all fine categories. All other $p - 18$ predictors are irrelevant. Just as in the nonoverlapping case, for $j \in \{1, \dots, 6\}$, Model j is defined by taking $s = 3(j - 1)$. As before, nonzero values of β^* are drawn independently $\mathrm{N}(0, 5)$. We consider $p \in \{100, 200, 500, 1000\}$ for each model.

In this scenario, we consider two versions of `Approx`. In this setting, the method `Approx` uses $\mathcal{A}_1, \dots, \mathcal{A}_4$ (as defined in the Section 6) whereas `Coarse-Approx` uses $\mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_5$. Both versions of `Approx` are inspired by the existing methods of de Kanter et al. (2019) and Bernstein et al. (2021), who use conditional models to account for the multiresolution structure of cell type.

We present Hellinger distance results for settings with overlapping $\mathcal{A}_l$ in Figure 10. We see relative performances vary across models in a similar pattern as in the simulations performed in Section 6. Specifically, under Model 1 when all 18 important predictors distinguish between all fine categories, we see that `mrMLR` and `Group` perform best. From Models 2–6, we gradually see `mrMLR` outperform `Group`. By Model 4, `mrMLR` the interquartile range for both methods no longer overlap. Of course, this agrees with intuition since `mrMLR` can exploit that a large number of the important predictors only distinguish between some coarse categories to improve efficiency. The competitors which make use of this information, `Approx` and `Coarse-Approx`, perform nearly as well as `mrMLR` under Model 6, but perform very poorly for Models 1–3, and significantly worse than `mrMLR` in Models 4 and 5. Using Kullback–Leibler divergence as a performance metric, in Figure 11 we again see a similar pattern as in Figure 7.

Finally, in Figure 12, we display classification error results under the data generating model with overlapping coarse categories. As before, we also consider the competitors `MSDA`, RF, and HRF. We see the same general patterns as in the nonoverlapping coarse category case (Figure 8), though in this setting HRF never significantly outperforms RF. In all settings, our method mrMLR and Group outperform MSDA, RF, and HRF.

E Generating β^* in the simulation studies

To provide some additional insight as to how we generate β^* in the simulation studies, we provide the R code below. Not that this code applies for Models 2–6 in the nonoverlapping coarse category setting. Generating the coefficients under Model 1 is trivial.

```
groups <- list(1:3, 4:6, 7:9, 10:12)
s <- 3 + (Model - 2)*3
beta <- matrix(0, nrow=p, K)
coarse.inds <- sample(1:p, s)
for (kk in 1:4) {
  gen.coefs <- rnorm(length(coarse.inds), sd = sqrt(5))%*%t(rep(1, length(groups[[kk]])))
  beta[coarse.inds, groups[[kk]]] <- gen.coefs
}
fine.inds <- sample(c(1:p)[-coarse.inds], 18 - s)
for (kk in 1:4) {
  for (ll in 1:length(fine.inds)) {
    beta[fine.inds[ll], groups[[kk]]] <- rnorm(length(groups[[kk]]), sd = sqrt(5))
  }
}
```

F Additional details from Section 7

F.1 Data preparation

Note that many of the details provided in this section are identical to those from Motwani et al. (2023).

To prepare the data for our analysis, we first removed low-quality cells based on the percentage of mitochondrial reads and number of genes expressed (with nonzero counts) in each cell. Let $\widetilde{X}^c$ be the full $\widetilde{n} \times \widetilde{p}$ gene expression count matrix. Define $s_j = \sum_{g=1}^{\widetilde{p}} \widetilde{X}^c_{j,g}$, be the number of total counts for the jth cell. Also, let $\mathcal{M} \subset \{1, \dots, \widetilde{p}\}$ be the set of mitochondrial genes. Define the percentage of mitochondrial reads to be

$$m_j = 100 \cdot \sum_{g \in \mathcal{M}} \frac{\widetilde{X}^c_{j,g}}{s_j}.$$

Furthermore, define the number of expressed genes to be $e_j = \sum_{g=1}^{\widetilde{p}} \mathbf{1}(\widetilde{X}^c_{(j,g} > 0)$. Let $\mathcal{V}$ be the set of cells with no more than 5 percent mitochondrial reads and at least 200 genes expressed

$$\mathcal{V} = \{j : m_j < 5\} \cap \{k : e_k > 200\}$$

and define $n = |\mathcal{V}|$ and $\widetilde{X} = \widetilde{X}^c_{\mathcal{V},:}$ to be the filtered count matrix.

With low-quality cells removed, we then construct the normalized count matrix. Specifically, the normalized matrix X is defined by

$$X_{j,g} = \log\left(\frac{10^4 \cdot \widetilde{X}^c_{j,g}}{s_j} + 1\right), \quad j \in \mathcal{V}, \quad g \in [\widetilde{p}].$$

This is the standard log-normalization used in the software `Seurat`.

Finally, we rank genes according to their variability after adjusting for their expected expression. Specifically, to rank genes for screening purposes we use the `FindVariableFeatures` function in Seurat with `selection.method = "vst"` on the normalized matrix X and rank genes according to the `vst.variance.standardized` column in descending order for each dataset (Stuart et al., 2019). Given the ranking of all $\widetilde{p}$ genes, $\mathcal{G}$ (where the first element of $\mathcal{G}$ is most variable, and the pth element the least), when varying the number of genes p, we take the first p genes from this ordered list, and reassign $X = X_{:,\mathcal{G}_{1:p}}$.

F.2 Investigating selected genes

In this section, we display the expression of two genes which were estimated to distinguish between certain cell types at a coarse resolution. Specifically, based on the estimated regression coefficient matrix in Figure 6, our method estimates that MS4A1 is able to distinguish between B cell subtypes, but not between any other coarse cell type. In the top panel of Figure 14, we see that the distribution of MS4A1 expression differs across B cell subtypes—it appears B intermediate has somewhat higher expression than B memory and B naive—but does not appear to be expressed much in any of the other coarse cell types. In the bottom panel of Figure 14, we see that the expression of XCL2, which our method estimated to distinguish between the two types of NK cells, clearly separates NK and NK_CD56bright. Interestingly, we also see that XCL2 was estimated to distinguish between the fine CD8$^+$ cells. In Figure 14, we see that some CD8$^+$ TEM cells have higher log counts than any of the other fine CD8$^+$ cells.

Code for creating these plots—which can be modified to examine any gene—is available with the code portion of the Supporting Information.

As mentioned in the main manuscript, we also include a tree-based visualization of the cell type hierarchy characterized in Table 1. This plot was created using the `PlotTopoTree` function from the `HierFIT` R package (Kaymaz et al., 2021).

F.3 Comparison to `MSDA`, `RF`, and `HRF`

In this section, we compare our method `mrMLR` and `Group` to `MSDA`, `RF`, and `HRF` in the single-cell RNA-seq data analysis from Section 7.2. Because only `mrMLR` and `Group` model cell type probabilities directly, we do not consider deviance here, but rather, focus exclusively on classification accuracy. Results under the same setup described in Section 7.2 are provided in Figure 16. As described in the main manuscript, both `Group` and `mrMLR` outperform the competitors across every scenario we considered.

In addition to the five competitors mentioned above, we also used `mrMLR-Or` and `RF-Or`. These are the methods `mrMLR` and `RF` but with the candidate genes restricted to only those which were identified as marker genes in Hao et al. (2021). For `mrMLR-Or`, we set $\gamma = 0$ so that all the marker genes from Hao et al. (2021) are included in the model, but they may affect the cell type probabilities at various resolutions. Because this gene set was partially used to actually label these cells, this constitutes “oracle” information—hence the designation `-Or`.

In Figure 16, we see that both oracle methods perform worse than `mrMLR`, which performs both gene selection and model fitting simultaneously. Notably, `RF-Or` performs better than `RF`. This can be explained by the fact that `RF` does not perform any type of variable selection, so restricting attention to the marker gene set from Hao et al. (2021) improves performance. However, both variants perform worse than both `mrMLR-Or` and `mrMLR`.

F.4 Additional figures

In Figure 6, we present a heatmap displaying the entire estimated coefficient matrix. This is the matrix $\hat{\beta}$, for which we display a submatrix in Figure 5 of the main manuscript.

G Existing methods and extensions

G.1 Comparison of mrMLR to Motwani et al. (2023) and Molstad and Rothman (2023)

One method related to our own, proposed by Motwani et al. (2023), also uses the multinomial logistic regression model for cell type annotation, but in the context of integrative cell type annotation. Their motivation was to fit the multinomial logistic regression model in a setting where the training data consist of multiple datasets from distinct studies whose response labels are available at different resolutions. For example, a practitioner may have one dataset where the responses labels are the fine cell types from Table 1 of the main manuscript (of which there are 28 types) and another dataset where the response labels are only the most coarse from Table 1 (of which there are 9). Motwani et al. (2023) use the multiresolution structure of cell types to define the observed data log-likelihood and thus fit a multinomial logistic regression model at the finest resolution possible. However, their fitted model does not have the interpretability afforded by mrMLR. If one applied the estimator of Motwani et al. (2023) to a single dataset (as in our motivating data analysis), this would be equivalent to fitting mrMLR with $\lambda = 0$, i.e., the standard group lasso penalized multinomial logistic regression (Vincent and Hansen, 2014). Our new multiresolution penalty could be incorporated into the method of Motwani et al. (2023), though this would require substantial modification of their computational algorithm. see Additional details about Motwani et al. (2023) are provided in Web Appendix E.

There are notable methodological similarities between mrMLR and the estimator proposed in Molstad and Rothman (2023), who focused on fitting the multivariate categorical response regression model (i.e., regression model with multiple distinct categorical response variables). In effect, the method of Molstad and Rothman (2023) shrinks a linear combination of regression coefficients from a multinomial logistic regression model using the penalty $\lambda\Phi(\beta) = \lambda \sum_{j=2}^{p} \|P^\top \beta_{j,:}\|_2$ where P is a matrix constructed so that each row of $P^\top \beta_{j,:}$ corresponds to the jth predictor's effect on the log-odds ratio between different categorical responses. This allowed for fitted models where some predictors could be interpreted as affecting only the marginal distributions of the responses. While our proposed estimator also penalizes the Euclidean norm of a linear combination of coefficients within a row of β, the matrices P and $D(a_l)$ have inherently distinct structures. For example, in the setting where the multinomial logistic regression model corresponds to a bivariate binary categorical response, $P = (1, -1, -1, 1)^\top \in \mathbb{R}^4$. Moreover, unlike Φ, $\Omega_{\mathcal{A}}$ allows for multiple penalties to be applied to each row of β and allows for penalties on overlapping sets of coefficients. The new multiresolution penalty thus requires novel computational and theoretical results, and is needed to address a fundamentally different problem than that considered in Molstad and Rothman (2023).

G.2 Application of `mrMLR` to integrative cell type annotation

In Motwani et al. (2023), the authors propose a method for fitting the multinomial logistic regression model using multiple datasets from distinct studies whose response labels are available at different resolutions. For more details on this setting, see Motwani et al. (2023). While the multiresolution structure of the response is used in defining the observed data log-likelihood, their method does not lead to the fitted model interpretability that the new penalty described in this article provides. The purpose of this section is to describe how we could, in principle, combine the main ideas proposed by Motwani et al. (2023) and this paper.

We first recall some of the key notation from Motwani et al. (2023), ignoring their extension to account for batch effects (i.e., setting $\rho = 0$, named as `IBMR-NG`) for brevity (and computational feasibility). Suppose we observe $J \geq 1$ datasets with single-cell gene expression profiles and cell types manually annotated. Let $\mathcal{C}_j$ denote the set of labels used to annotate the jth dataset for $j \in [J] = \{1, \ldots, J\}$ and let $\mathcal{C}$ denote the set of labels at the desired finest resolution across all datasets. Let $Y_{(j)i}$ and $\tilde{Y}_{(j)i}$ be the random variables corresponding to the annotated cell type and true (according to the finest resolution label set) cell type of the ith cell in the jth dataset for $j \in [J]$, $i \in [n_j] = \{1, \ldots, n_j\}$, with supports $\mathcal{C}_j$ and $\mathcal{C}$, respectively. Define the user-specified binning function $f_j : \mathcal{C} \to \mathcal{C}_j$ which maps a finest resolution category to the label used to describe that category in the jth dataset. Also, define the "unbinning" function $g_j = f_j^{-1}$ (inverse image) where $g_j(k) = f_j^{-1}(k) = \{l \in \mathcal{C} : f_j(l) = k\}$ for $k \in \mathcal{C}_j$. We also now define the relationship between $Y_{(j)i}$ and $\tilde{Y}_{(j)i}$ through the following equivalence of events

$$\{Y_{(j)i} = k\} = \bigcup_{l \in g_j(k)} \{\tilde{Y}_{(j)i} = l\}, \quad j \in [J], \quad k \in \mathcal{C}_j.$$

We thus have that

$$\Pr(Y_{(j)i} = k \mid x_{(j)i}) = \sum_{l \in g_j(k)} \Pr(\tilde{Y}_{(j)i} = l \mid x_{(j)i}), \quad j \in [J], \quad k \in \mathcal{C}_j \tag{18}$$

since the events $\{\tilde{Y}_{(j)i} = l\}$ and $\{\tilde{Y}_{(j)i} = l'\}$ with $l, l' \in g_j(k)$ and $l \neq l'$ are mutually exclusive as a cell can only be of one cell type. Under the multinomial logistic regression model, the log-likelihood contribution for the ith cell in the kth dataset can be expressed as

$$\ell_{(j)i}^{\mathcal{C},g}(\beta) = \sum_{k \in \mathcal{C}_j} \mathbf{1}(y_{(j)i} = k) \log \left(\sum_{l \in g_j(k)} \frac{\exp(x_{(j)i}^\top \beta_{:,l})}{\sum_{v \in \mathcal{C}} \exp(x_{(j)i}^\top \beta_{:,v})} \right)$$

for $j \in [J]$ and $i \in [n_j]$, where $\mathbf{1}$ denotes the indicator function. Motwani et al. (2023) therefore define the (scaled by $1/N$) negative log-likelihood as

$$\mathcal{L}_{\mathcal{C},g}(\beta) = -\frac{1}{N} \sum_{j=1}^{J} \sum_{i=1}^{n_j} \ell_{(j)i}^{\mathcal{C},g}(\beta),$$

where $N = \sum_{j=1}^{J} n_j$ is the total sample size. The `IBMR-NG` estimator proposed by Motwani et al. (2023) is defined as

$$\arg\min_{\beta} \left\{ \mathcal{L}_{\mathcal{C},g}(\beta) + \lambda \sum_{j=2}^{p} \|\beta_{j,:}\|_2 \right\}. \tag{19}$$

However, as mentioned earlier, this estimator does not provide interpretability in terms of which genes help to distinguish between which subsets of cell types. Therefore, we can use the penalty proposed in this paper, which would yield the estimator

$$\arg\min_{\beta} \left\{ \mathcal{L}_{\mathcal{C},g}(\beta) + \lambda \sum_{j=2}^{p} \|\beta_{j,:}\|_2 + \gamma \Omega_{\mathcal{A}}(\beta) \right\}, \tag{20}$$

recalling that

$$\Omega_{\mathcal{A}}(\beta) = \lambda \sum_{j=2}^{p} \sum_{l=1}^{L} w_l \left(\min_{c \in \mathbb{R}} \|\beta_{j,\mathcal{A}_l} - c 1_{a_l}\|_2 \right).$$

The computational developments in Section 4 can be used to compute (20). The theoretical results in Section 5, however, will not apply to (20) since there is, in effect, a loss of information when some of the responses are labeled at a coarser resolution. We leave a rigorous study of (20) – computationally, theoretically, and empirically – as a direction for future research.

G.3 Hierarchical classification and tree-based effect aggregation

Finally, we note that the problem of classification with response categories organized in a known hierarchy is well-studied in machine learning: see, for example, Wang et al. (2009, 2011); Silla and Freitas (2011). Broadly speaking, the majority of methods for hierarchical classification are not model-based, and thus classifiers are often difficult to interpret. To the best of our knowledge, no existing method is designed to yield fitted model that can be interpreted in the way afforded by `mrMLR`. Moreover, many existing approaches fit distinct conditional classifiers at each level of the hierarchy, which can be less efficient and more difficult to interpret than fitting a single unified model: e.g., see the performance of `Coarse` and `Coarse-Approx` in our simulation studies.

There is also work on tree-structured effect-aggregation in the context of regression (Yan and Bien, 2021; Shao et al., 2021), an in Gaussian graphical modelling (Wilms and Bien, 2022). Most relevant to our work, Yan and Bien (2021) proposed a method for encouraging coefficient equality in the context of rare predictor aggregation for linear regression. In brief, they require that predictors to be aggregated (i.e., predictors whose corresponding coefficients are equivalent) are related through a tree-structure. Our method would not require such a hierarchy, but could be computationally burdensome to implement in their context. The notion of false split rate, introduced in Shao et al. (2021), could be adopted in the context of multiresolution categorical regression: we leave this extension as future work.

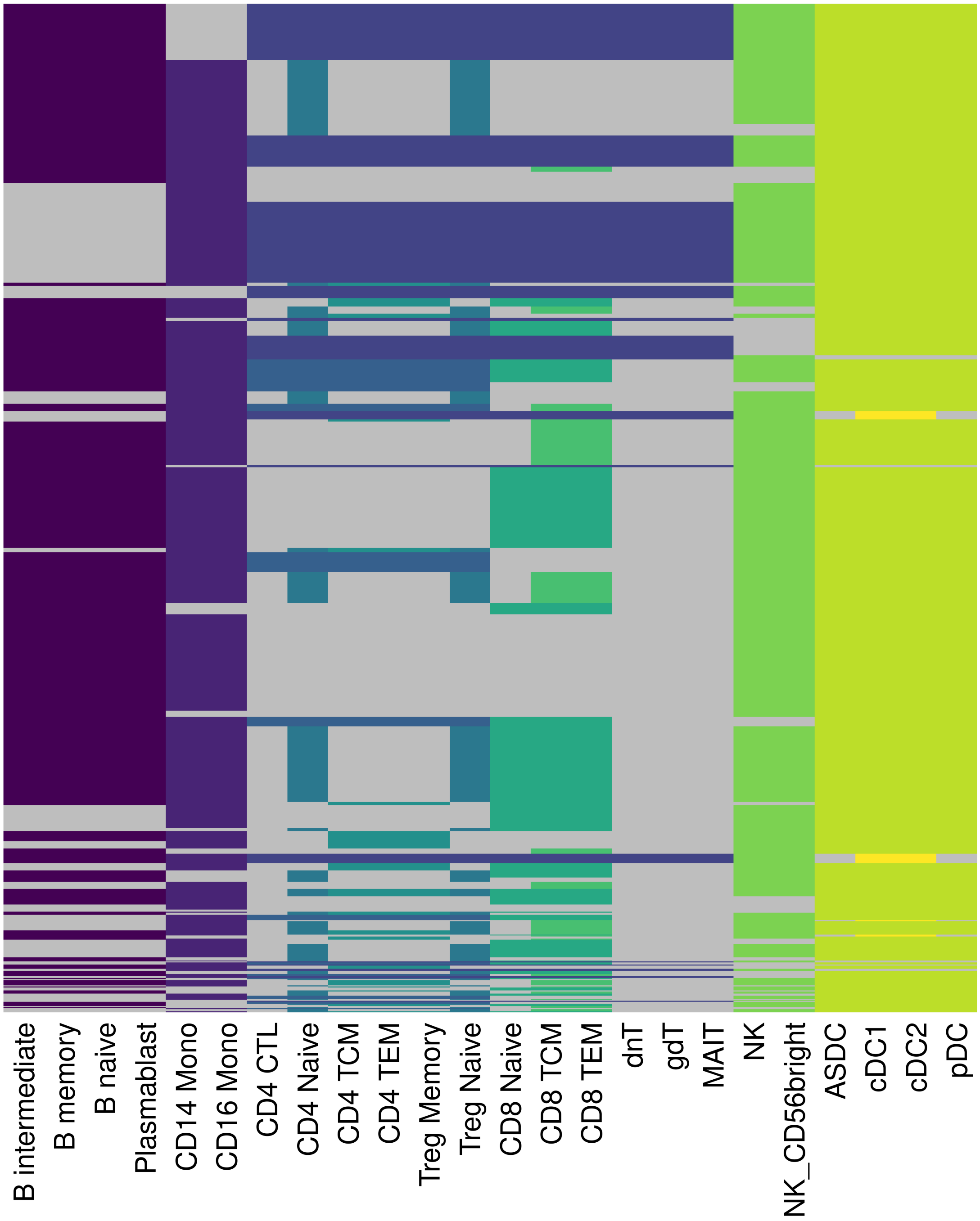

Figure 6: The entire matrix $\hat{\beta}$ from Section 7.3. Grey cells are those with estimated coefficient values distinct from all others in their row, whereas cells of the same (non-grey) color have identical coefficient values within a row.

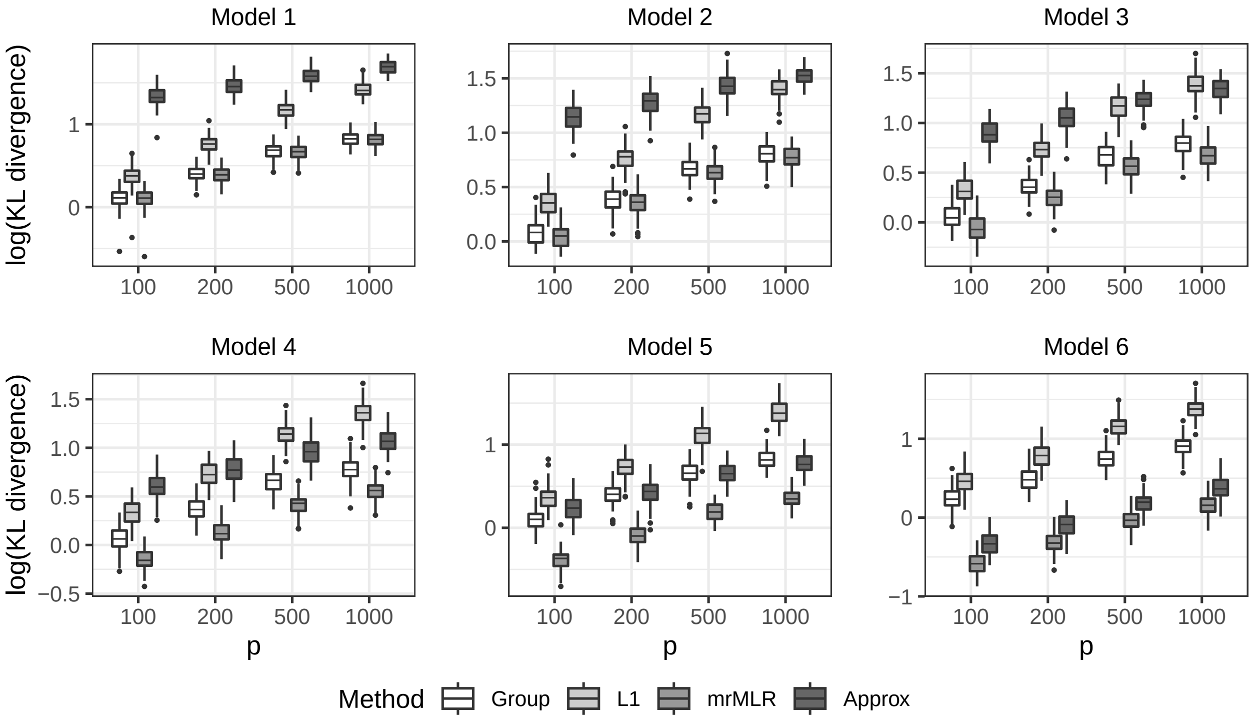

Figure 7: Kullback-Leibler divergences over 100 independent replications under Models 1–6 with $p \in \{100, 200, 500, 1000\}$ and $\mathcal{A}_l = \{3(l-1)+1, 3(l-1)+2, 3l\}$ for $l \in [4]$.

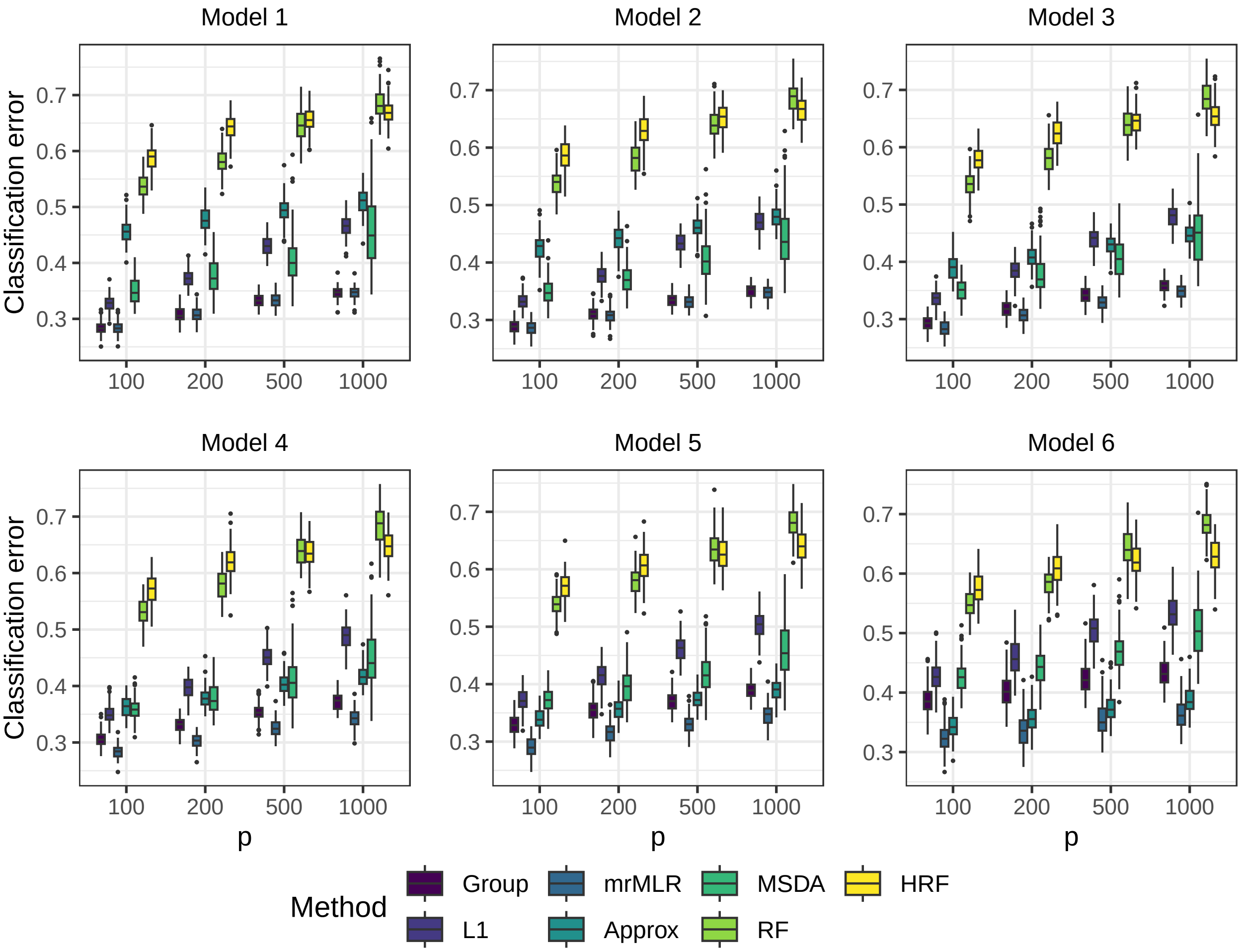

Figure 8: Classification errors over 100 independent replications under Models 1–6 with $p \in \{100, 200, 500, 1000\}$ and $\mathcal{A}_l = \{3(l-1)+1, 3(l-1)+2, 3l\}$ for $l \in [4]$.

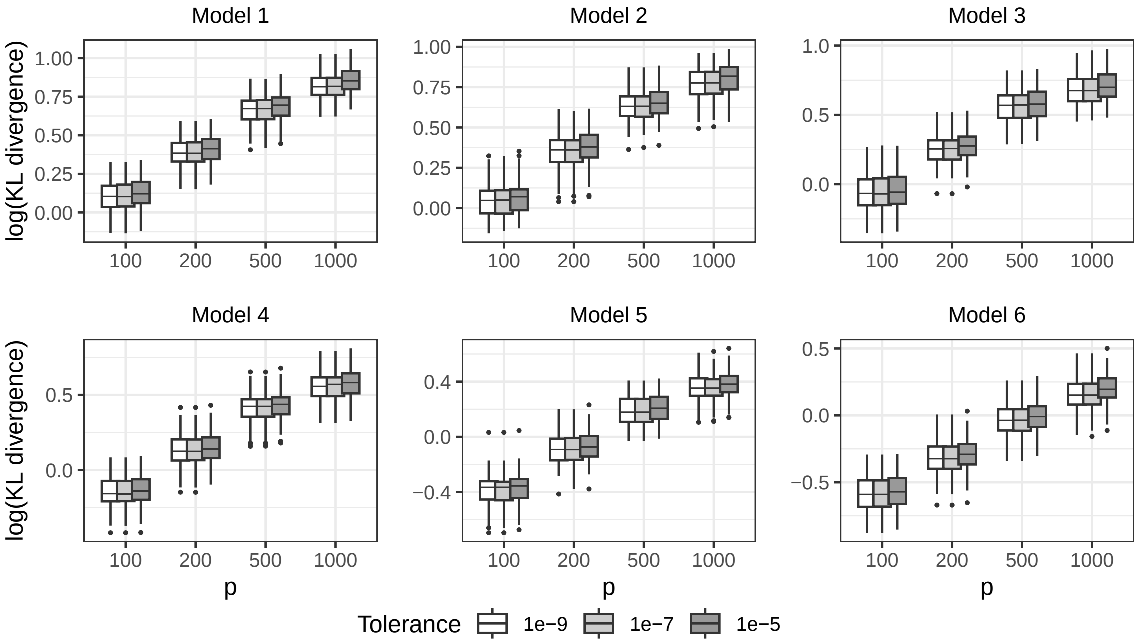

Figure 9: Kullback-Leibler divergences of mrMLR with convergence tolerances $\epsilon \in \{10^{-9}, 10^{-7}, 10^{-5}\}$ under the same data generating models as described in Section 6.

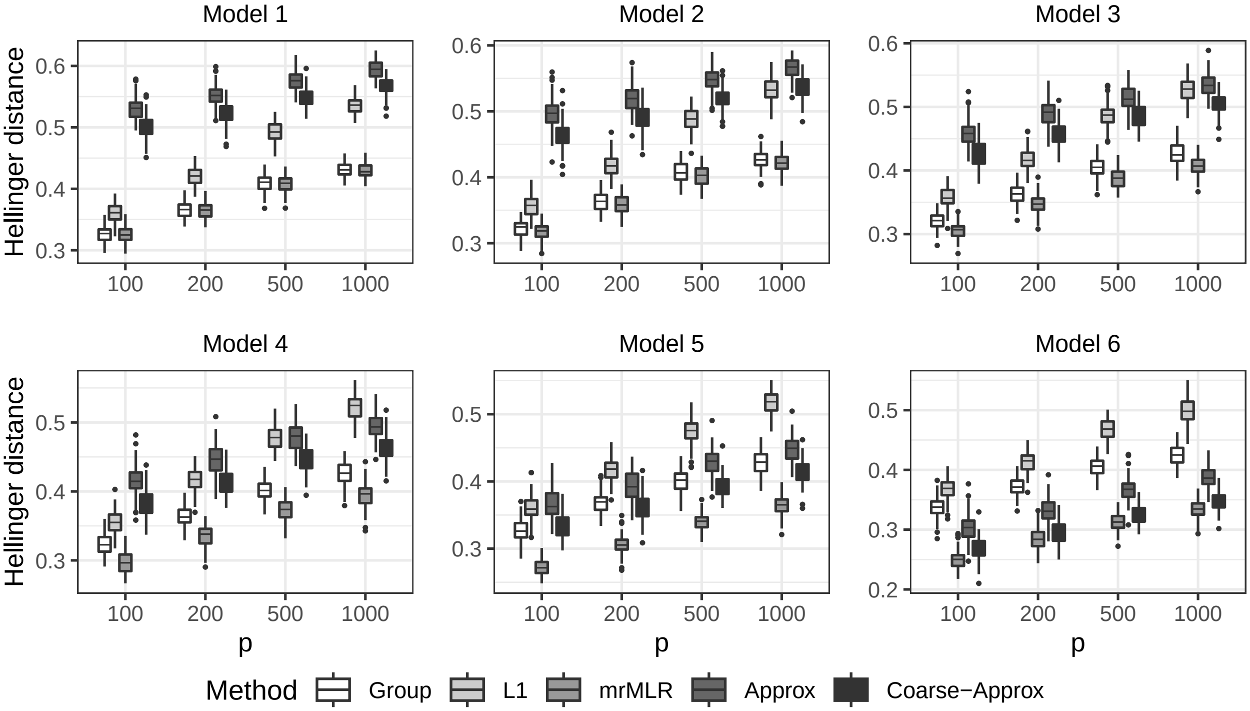

Figure 10: Hellinger distance over 100 independent replications under Models 1–6 with $p \in \{100, 200, 500, 1000\}$, $\mathcal{A}_l = \{3(l-1)+1, 3(l-1)+2, 3l\}$ for $l \in \{1, 2, 3, 4\}$, and $\mathcal{A}_5 = \{1, \ldots, 6\}$.

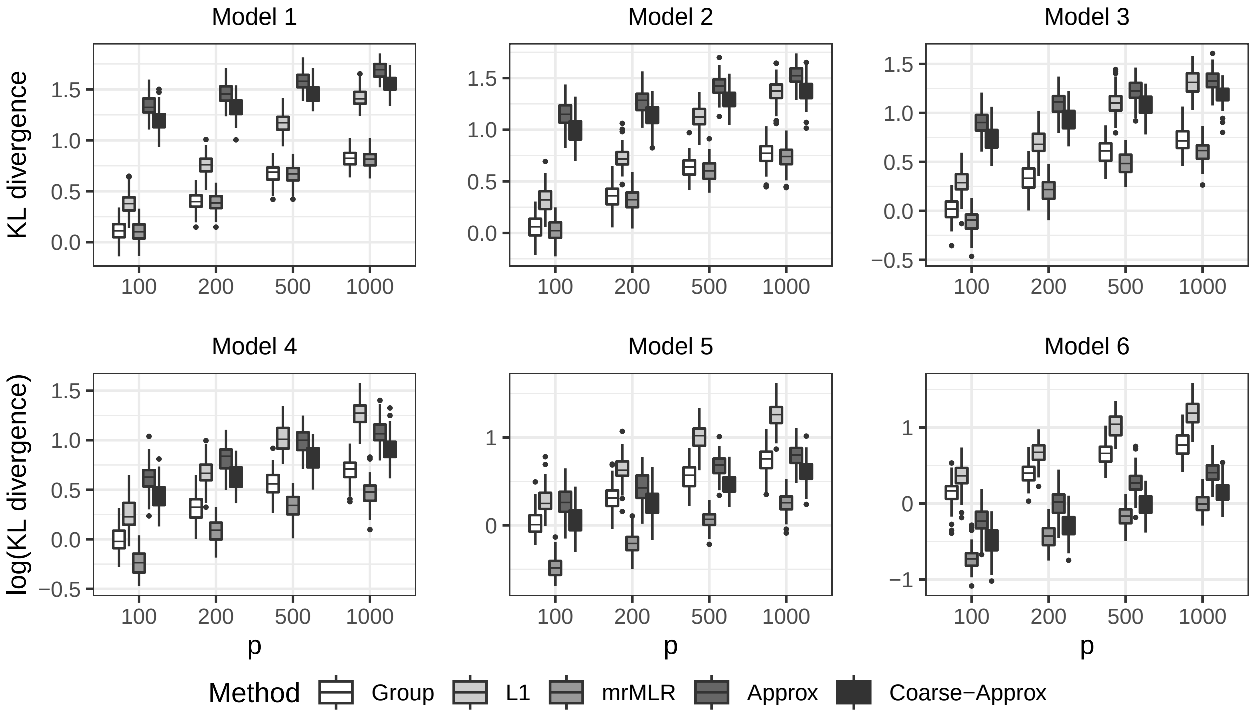

Figure 11: Kullback-Leibler divergences over 100 independent replications under Models 1–6 with $p \in \{100, 200, 500, 1000\}$, $\mathcal{A}_l = \{3(l-1)+1, 3(l-1)+2, 3l\}$ for $l \in [4]$, and $\mathcal{A}_5 = [6]$.

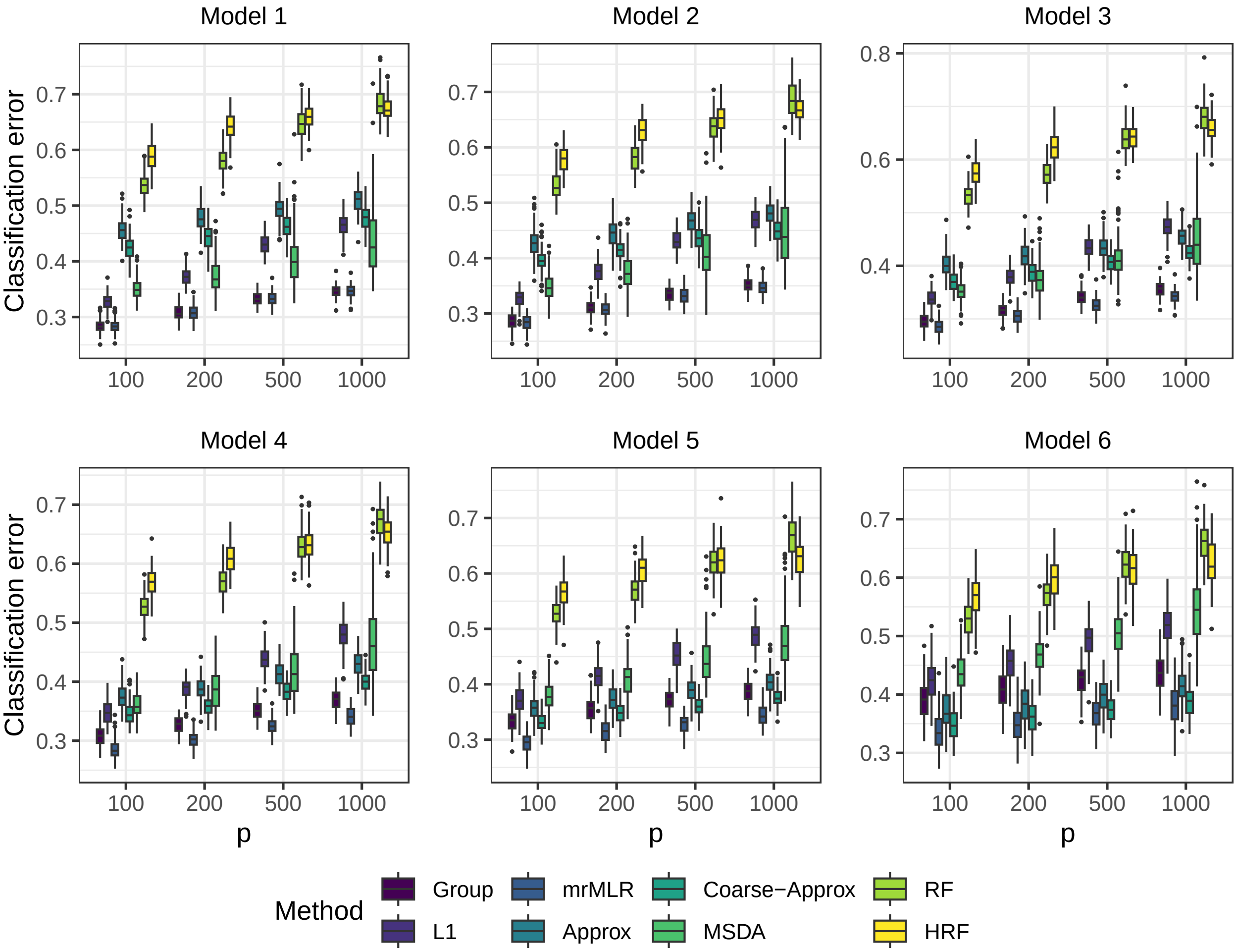

Figure 12: Classification errors over 100 independent replications under Models 1–6 with $p \in \{100, 200, 500, 1000\}$, $\mathcal{A}_l = \{3(l-1)+1, 3(l-1)+2, 3l\}$ for $l \in [4]$, and $\mathcal{A}_5 = [6]$.

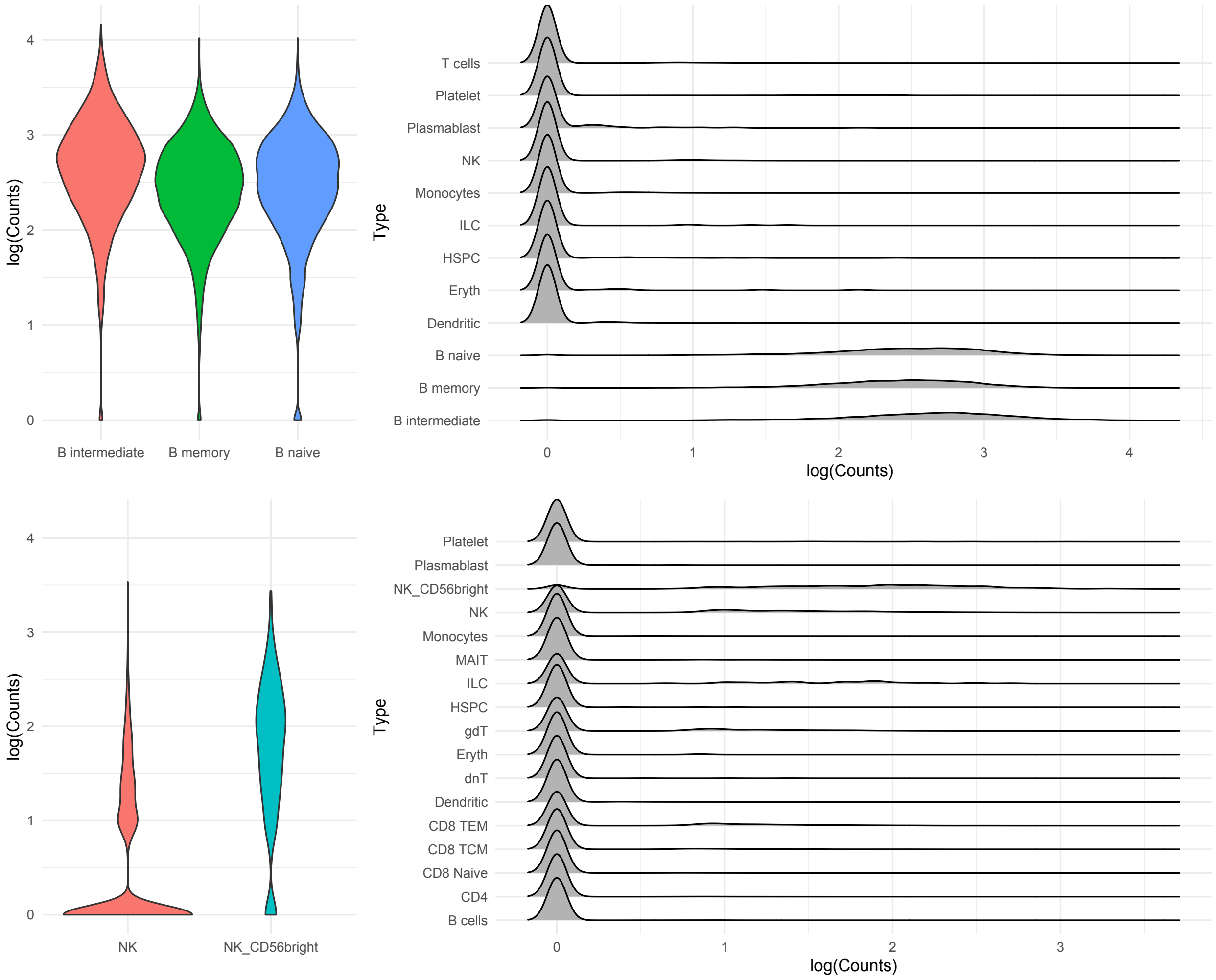

Figure 13: Expression of (top) MS4A1 and (middle) XCL2 aggregated over (left) fine cell types the gene was estimated to distinguish between. In the right panel, we show normalized log RNA-seq counts over the cell types which the gene was estimated to be able to distinguish between.

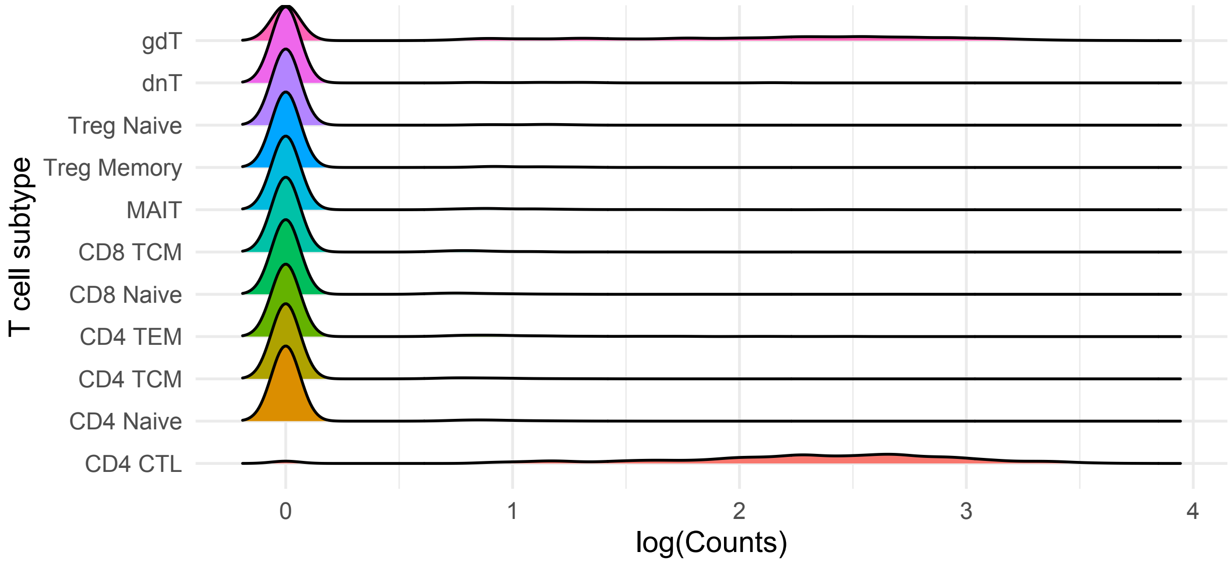

Figure 14: Expression of GZMH over T cell subtypes (excluding dnT, gdT, and MAIT) in the complete data from Hao et al. (2021).

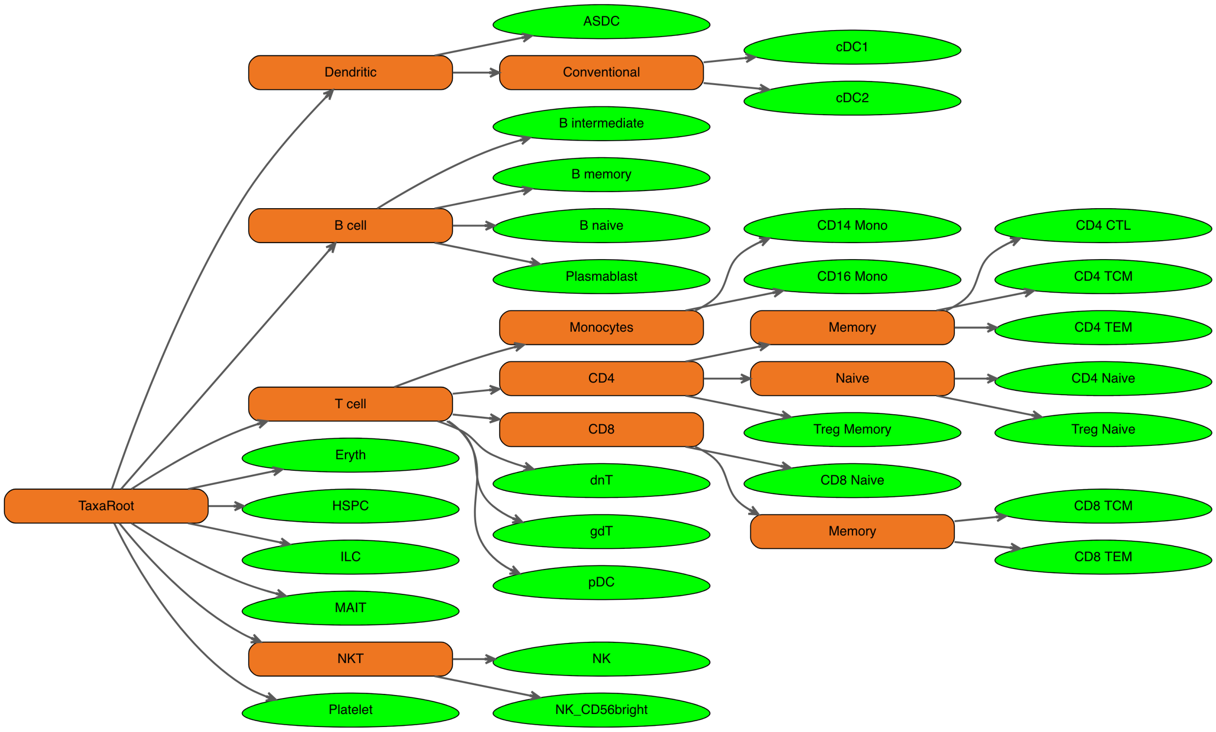

Figure 15: A visualization of the cell-type hierarchy characterized in Table 1 of the main manuscript.

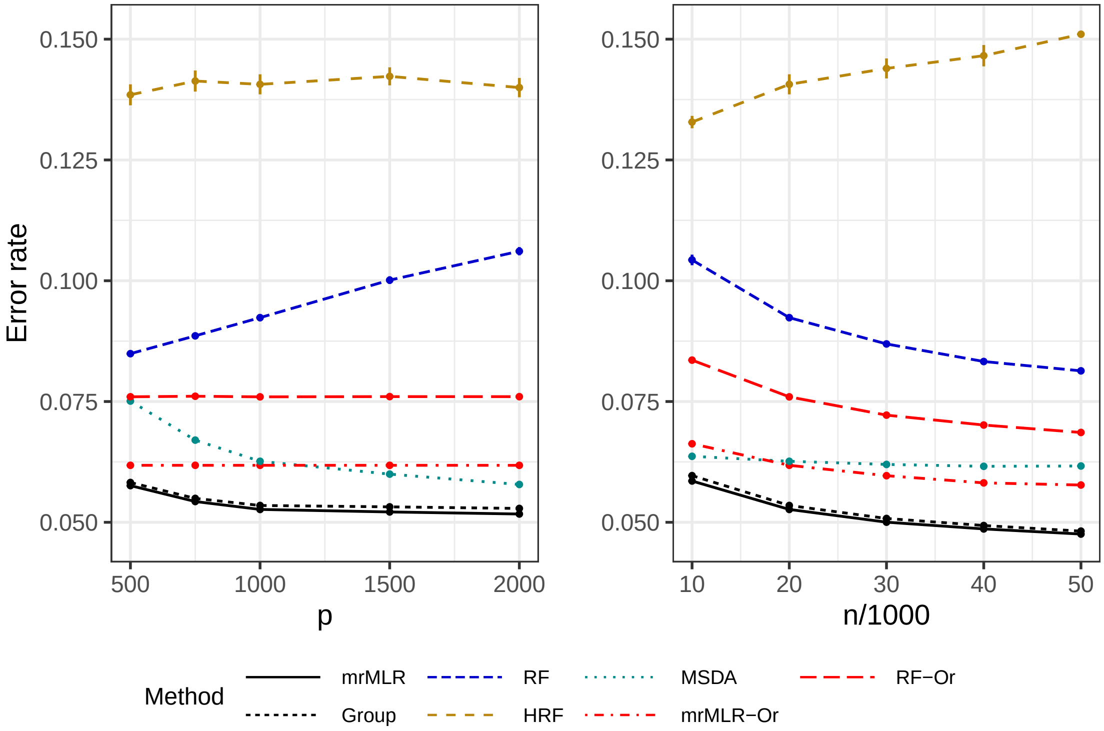

Figure 16: Average classification error rates on the single-cell RNA-seq dataset from Hao et al. (2021) with (left) $n = 20000$ fixed and $p \in \{500, 1000, 1500, 2000\}$ and (right) $p = 1000$ fixed and $n \in \{10000, 20000, 30000, 40000, 50000\}$.

References

Bach, F. (2010). Self-concordant analysis for logistic regression. *Electronic Journal of Statistics*, 4:384–414.

Bernstein, M. N., Ma, Z., Gleicher, M., and Dewey, C. N. (2021). CellO: Comprehensive and hierarchical cell type classification of human cells with the cell ontology. *Iscience*, 24(1).

de Kanter, J. K., Lijnzaad, P., Candelli, T., Margaritis, T., and Holstege, F. C. (2019). CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. *Nucleic Acids Research*, 47(16):e95–e95.

Hao, Y., Hao, S., Andersen-Nissen, E., Mauck III, W. M., Zheng, S., Butler, A., Lee, M. J., Wilk, A. J., Darby, C., and Zager, M. (2021). Integrated analysis of multimodal single-cell data. *Cell*, 184(13):3573–3587.

Kaymaz, Y., Ganglberger, F., Tang, M., Haslinger, C., Fernandez-Albert, F., Lawless, N., and Sackton, T. B. (2021). Hierfit: a hierarchical cell type classification tool for projections from complex single-cell atlas datasets. *Bioinformatics*, 37(23):4431–4436.

Mai, Q., Yang, Y., and Zou, H. (2019). Multiclass sparse discriminant analysis. *Statistica Sinica*, 29(1):97–111.

Molstad, A. J. and Rothman, A. J. (2023). A likelihood-based approach for multivariate categorical response regression in high dimensions. *Journal of the American Statistical Association*, 118(542):1402–1414.

Motwani, K., Bacher, R., and Molstad, A. J. (2023). Binned multinomial logistic regression for integrative cell type annotation. *Annals of Applied Statistics.*

Negahban, S. N., Ravikumar, P., Wainwright, M. J., and Yu, B. (2012). A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557.

Shao, S., Bien, J., and Javanmard, A. (2021). Controlling the false split rate in tree-based aggregation. *arXiv preprint arXiv:2108.05350.*

Silla, C. N. and Freitas, A. A. (2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22:31–72.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck, W. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, 177(7):1888–1902.

Tibshirani, R. J. and Taylor, J. (2011). The solution path of the generalized lasso. *The Annals of Statistics*, 39(3):1335 – 1371.

Tran-Dinh, Q., Li, Y.-H., and Cevher, V. (2015). Composite convex minimization involving self-concordant-like cost functions. In *Modelling, Computation and Optimization in Information Systems and Management Sciences*, pages 155–168. Springer.

Vincent, M. and Hansen, N. R. (2014). Sparse group lasso and high dimensional multinomial classification. *Computational Statistics and Data Analysis*, 71:771–786.

Wang, H., Shen, X., and Pan, W. (2011). Large margin hierarchical classification with mutually exclusive class membership. *Journal of Machine Learning Research*, 12(9).

Wang, J., Shen, X., and Pan, W. (2009). On large margin hierarchical classification with multiple paths. *Journal of the American Statistical Association*, 104(487):1213–1223.

Wilms, I. and Bien, J. (2022). Tree-based node aggregation in sparse graphical models. *The Journal of Machine Learning Research*, 23(1):11078–11113.

Yan, X. and Bien, J. (2021). Rare feature selection in high dimensions. *Journal of the American Statistical Association*, 116(534):887–900.