A. Proof of Lemma 1

We begin by rewriting the loss ℓ_α as a cost-sensitive classification loss. First, we use the telescoping trick to obtain

$$\begin{aligned}\ell_\alpha(y,u) &= \ell\Big(\underline{y}, \lfloor u \rfloor_\alpha + \tfrac{\alpha}{2}\Big)\\ &= \ell(\underline{y}, \tfrac{\alpha}{2}) + \sum_{z\in\mathcal{Z}} \underbrace{\Big[\ell\Big(\underline{y}, z+\tfrac{\alpha}{2}\Big) - \ell\Big(\underline{y}, z-\tfrac{\alpha}{2}\Big)\Big]}_{c(\underline{y},z)/N} \mathbf{1}\{u \ge z\}.\end{aligned}$$

Now plugging in $u = \underline{f}(x)$ and using the fact that for $z \in \mathcal{Z}$, we have $\mathbf{1}\{\underline{f}(x) \ge z\} = \mathbf{1}\{f(x) \ge z\} = h_f(x,z)$, we obtain

$$\ell_\alpha(y, \underline{f}(x)) = \ell(\underline{y}, \tfrac{\alpha}{2}) + \frac{1}{N}\sum_{z\in\mathcal{Z}} c(\underline{y}, z) h_f(x,z). \tag{15}$$

Thus, ignoring the constant $\ell(\underline{y}, \frac{\alpha}{2})$, the loss ℓ_α can be viewed as the cost-sensitive error of the classifier h_f.

For $z \in \mathcal{Z}$, we can rewrite $\gamma_{a,z}(\underline{f})$ as

$$\begin{aligned}\gamma_{a,z}(\underline{f}) &= \mathbb{P}[\underline{f}(X) \ge z \mid A = a] - \mathbb{P}[\underline{f}(X) \ge z]\\ &= \mathbb{E}[h_f(X,z) \mid A = a] - \mathbb{E}[h_f(X,z)] \\ &= \gamma_{a,z}(h_f),\end{aligned} \tag{16}$$

completing the proof of the lemma.

B. Iteration Complexity of Algorithm 1

Theorem 4. *Algorithm 1 terminates in at most* $\frac{16B^2\log(2|\mathcal{A}|N+1)}{\nu^2}$ *iterations. Furthermore, if* Q *is any feasible point of* (11) *then the solution* $\widehat{Q}$ *returned by Algorithm 1 satisfies:*

$$\begin{aligned}\widehat{\text{cost}}(\widehat{Q}) &\le \widehat{\text{cost}}(Q) + 2\nu\\ \big|\widehat{\gamma}_{a,z}(\widehat{Q})\big| &\le \widehat{\varepsilon}_a + \frac{2+2\nu}{B} \quad \textit{for all } a \in \mathcal{A},\ z\in\mathcal{Z}.\end{aligned}$$

Proof. This result is essentially a corollary of Theorem 1, and Lemmas 2 and 3 of Agarwal et al. (2018). Specifically, we note that the constraints appearing in our problem (11) can be cast in their general framework along the lines of their Example 1, with a total of $2|\mathcal{A}|N$ constraints. Following their Example 3, we obtain that the maximal constraint violation ρ, needed in their Theorem 1, is at most 2. We further observe that the violation of the i.i.d. structure by explicit averaging over z values does not impact their optimization analysis in any way. Therefore, their Theorem 1 with $\rho = 2$ implies that our Algorithm 1 finds a ν-approximate saddle point of the Lagrangian in at most $\frac{16B^2\log(2|\mathcal{A}|N+1)}{\nu^2}$ iterations as desired.

To bound $\widehat{\text{cost}}(\widehat{Q})$ and $\big|\widehat{\gamma}_{a,z}(\widehat{Q})\big|$ we appeal to their Lemmas 2 and 3. First note that their approach applies to the objective of our problem (11) as long as the costs $c(y,z)$ are in $[0,1]$ (see their footnote 4). However, in our case, these can be in $[-1,1]$ (see Eq. 9). This does not affect their Theorem 1 and Lemma 2, but their Lemma 3 now holds with the right-hand side equal to $\frac{2+2\nu}{B}$ instead of $\frac{1+2\nu}{B}$. Their Lemma 2 immediately yields the bound $\widehat{\text{cost}}(\widehat{Q}) \le \widehat{\text{cost}}(Q) + 2\nu$, whereas the modified Lemma 3 yields the bound $\big|\widehat{\gamma}_{a,z}(\widehat{Q})\big| \le \widehat{\varepsilon}_a + \frac{2+2\nu}{B}$ for all a, z, finishing the proof. □

C. Proof of Theorem 2

Before proving the theorem, we recall a standard definition of the Rademacher complexity of a class of functions, which plays an important role in our deviation bounds. Let $\mathcal{G}$ be a class of functions $g : \mathcal{U} \to \mathbb{R}$ over some space $\mathcal{U}$. Then the (worst-case) Rademacher complexity of $\mathcal{G}$ is defined as:

$$R_n(\mathcal{G}) \coloneqq \sup_{u_1,\ldots,u_n\in\mathcal{U}} \mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}}\left|\frac{1}{n}\sum_{i=1}^{n}\sigma_i g(u_i)\right|\right], \tag{17}$$

where the expectation is over the i.i.d. random variables $\sigma_1, \ldots, \sigma_n$ with $\mathbb{P}(\sigma_i = 1) = \mathbb{P}(\sigma_i = -1) = 1/2$.

The Rademacher complexity of a class $\mathcal{G}$ can be used to obtain uniform bounds of any Lipschitz continuous transformations of $g \in \mathcal{G}$ as follows:

Lemma 2. *Let D be a distribution over a pair of random variables (S,U) taking values in $\mathcal{S}\times\mathcal{U}$. Let $\mathcal{G}$ be a class of functions $g:\mathcal{U}\to[0,1]$, and let $\varphi:\mathcal{S}\times[0,1]\to[-1,1]$ be a contraction in its second argument, i.e., for all $s\in\mathcal{S}$ and all $t,t'\in[0,1]$, $|\varphi(s,t)-\varphi(s,t')|\le|t-t'|$. Then with probability at least $1-\delta$, for all $g\in\mathcal{G}$,*

$$\left|\widehat{\mathbb{E}}\bigl[\varphi(S,g(U))\bigr]-\mathbb{E}\bigl[\varphi(S,g(U))\bigr]\right|\le 4R_n(\mathcal{G})+\frac{2}{\sqrt{n}}+\sqrt{\frac{2\ln(2/\delta)}{n}}\ ,$$

where the expectation is with respect to D and the empirical expectation is based on n i.i.d. draws from D. If φ is also linear in its second argument then a tighter bound holds, with $4R_n(\mathcal{G})$ replaced by $2R_n(\mathcal{G})$.

Proof. Let $\Phi:=\{\varphi_g\}_{g\in\mathcal{G}}$ be the class of functions $\varphi_g:(s,u)\mapsto\varphi(s,g(u))$. By Theorem 3.2 of Boucheron et al. (2005), we then have with probability at least $1-\delta$, for all g,

$$\left|\widehat{\mathbb{E}}\bigl[\varphi(S,g(U))\bigr]-\mathbb{E}\bigl[\varphi(S,g(U))\bigr]\right|=\left|\widehat{\mathbb{E}}[\varphi_g]-\mathbb{E}[\varphi_g]\right|\le 2R_n(\Phi)+\sqrt{\frac{2\ln(2/\delta)}{n}}\ . \tag{18}$$

We will next bound $R_n(\Phi)$ in terms of $R_n(\mathcal{G})$. For a fixed tuple $(s_1,u_1),\dots,(s_n,u_n)$, we have

$$\begin{aligned}\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}}\left|\sum_{i=1}^n\sigma_i\varphi(s_i,g(u_i))\right|\right]&\le\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}}\left|\sum_{i=1}^n\sigma_i\Bigl(\varphi(s_i,g(u_i))-\varphi(s_i,0)\Bigr)\right|\right]+\sqrt{n}\\&\le 2\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}}\left|\sum_{i=1}^n\sigma_i g(u_i)\right|\right]+\sqrt{n}\end{aligned}$$

where the first inequality follows from Theorem 12(5) of Bartlett & Mendelson (2002) and the last inequality follows from the contraction principle of Ledoux & Talagrand (1991), specifically their Theorem 4.12. Dividing by n and taking a supremum over $(s_1,u_1),\dots,(s_n,u_n)$ yields the bound

$$R_n(\Phi)\le 2R_n(\mathcal{G})+\frac{1}{\sqrt{n}}.$$

Together with the bound (18), this proves the lemma for an arbitrary contraction φ. If φ is linear in its second argument, we get a tighter bound by invoking Theorem 4.4 of Ledoux & Talagrand (1991) instead of their Theorem 4.12. □

Our proof largely follows the proof of Theorems 2 and 3 of Agarwal et al. (2018). We first use Lemma 2 to show that by solving the empirical problem (11), we also obtain an approximate solution of the corresponding population problem:

$$\min_{Q\in\Delta(\mathcal{H})}\operatorname{cost}(Q)\text{ s.t. }\bigl|\gamma_{a,z}(Q)\bigr|\le\varepsilon_a\quad\forall a\in\mathcal{A},z\in\mathcal{Z}. \tag{19}$$

The theorem will then follow by invoking the equivalence between problem (19) and the discretized fair regression (8), and adding up various approximation errors.

Bounding empirical deviations in the cost and constraints. To bound the deviations in the cost, we need to be a bit careful, because the definition of $\widehat{\text{cost}}$ mixes the empirical expectation over the data with the averaging over $z\in\mathcal{Z}$. For the analysis, we therefore define

$$\widehat{\text{cost}}_z(h)=\widehat{\mathbb{E}}\bigl[c(\underline{Y},z)h(X,z)\bigr]\quad\text{and}\quad\text{cost}_z(h)=\mathbb{E}\bigl[c(\underline{Y},z)h(X,z)\bigr].$$

Since $c(\underline{Y},z)\in[-1,1]$, we can invoke Lemma 2 with $S=c(\underline{Y},z)$, $U=(X,z)$, $\mathcal{G}=\mathcal{H}$, and $\varphi(s,t)=st$ to obtain that with probability at least $1-\delta/2$ for all $z\in\mathcal{Z}$ and all $h\in\mathcal{H}$

$$\left|\widehat{\text{cost}}_z(h)-\text{cost}_z(h)\right|\le 2R_n(\mathcal{H})+\frac{2}{\sqrt{n}}+\sqrt{\frac{2\ln(4N/\delta)}{n}}=\widetilde{O}(n^{-\beta}),$$

where the last equality follows by Assumption 1 and the setting $N\propto n^\beta$. Taking an average over $z\in\mathcal{Z}$ and a convex combination according to any $Q\in\Delta(\mathcal{H})$, we obtain by Jensen's inequality that with probability at least $1-\delta/2$ for all $Q\in\Delta(\mathcal{H})$

$$\left|\widehat{\text{cost}}(Q)-\text{cost}(Q)\right|=\widetilde{O}(n^{-\beta}). \tag{20}$$

To bound the deviations in the constraints, we invoke Lemma 2 with $S=1$, $U=(X,z)$, $\mathcal{G}=\mathcal{H}$, and $\varphi(s,t)=st$, but apply it to the data distribution conditioned on $A=a$. We thus obtain that with with probability at least $1-\delta/2$ for all

$a \in \mathcal{A}$, $z \in \mathcal{Z}$, and $h \in \mathcal{H}$

$$\left|\widehat{\gamma}_{a,z}(h) - \gamma_{a,z}(h)\right| \le 2R_{n_a}(\mathcal{H}) + \frac{2}{\sqrt{n_a}} + \sqrt{\frac{2\ln(4|\mathcal{A}|N/\delta)}{n_a}}.$$

By Jensen's inequality this also means that with probability at least $1-\delta/2$ for all $a \in \mathcal{A}$, $z \in \mathcal{Z}$, and $Q \in \Delta(\mathcal{H})$

$$\left|\widehat{\gamma}_{a,z}(Q) - \gamma_{a,z}(Q)\right| \le 2R_{n_a}(\mathcal{H}) + \frac{2}{\sqrt{n_a}} + \sqrt{\frac{2\ln(4|\mathcal{A}|N/\delta)}{n_a}}. \tag{21}$$

In the remainder of the analysis, we assume that Eqs. (20) and (21) both hold, which occurs with probability at least $1-\delta$ by the union bound.

Putting it all together. Given the settings of ν, B and N, we obtain by Theorem 4 that Algorithm 1 terminates in $O\big(n^{4\beta}\ln(n^\beta|\mathcal{A}|)\big)$ iterations, as desired, and returns a distribution $\widehat{Q}$ which compares favorably with any feasible point Q of the empirical problem (11), meaning that for any such Q, we have

$$\widehat{\text{cost}}(\widehat{Q}) \le \widehat{\text{cost}}(Q) + O(n^{-\beta}) \tag{22}$$

$$\left|\widehat{\gamma}_{a,z}(\widehat{Q})\right| \le \widehat{\varepsilon}_a + O(n^{-\beta}) \quad \text{for all } a \in \mathcal{A},\, z \in \mathcal{Z}. \tag{23}$$

Now bounding $\widehat{\text{cost}}(\widehat{Q})$ and $\widehat{\text{cost}}(Q)$ in Eq. (22) via the uniform convergence bound (20), we obtain

$$\text{cost}(\widehat{Q}) \le \text{cost}(Q) + \widetilde{O}(n^{-\beta}). \tag{24}$$

Similarly, bounding $\widehat{\gamma}_{a,z}(\widehat{Q})$ via the bound (21) and $\widehat{\varepsilon}_a \le \varepsilon_a + \widetilde{O}(n_a^{-\beta})$ via Assumption 1 and our setting of C', we obtain

$$\left|\gamma_{a,z}(\widehat{Q})\right| \le \varepsilon_a + \widetilde{O}(n_a^{-\beta}) \quad \text{for all } a \in \mathcal{A},\, z \in \mathcal{Z}. \tag{25}$$

Above, we assumed that Q was a feasible point of the empirical problem (11). However, assuming that Eq. (21) holds, any feasible solution of the population problem (19) is also feasible in the empirical problem (11) thanks to our setting of C'. Thus, Eqs. (24) and (25) show that the solution $\widehat{Q}$ is approximately feasible and approximately optimal in the population problem (19). It remains to relate $\widehat{Q}$ to the original fair regression problem (3).

First, by Lemma 1 and Eqs. (24) and (25), we can interpret $\widehat{Q}$ as a distribution over the set of discretized regressors $\underline{\mathcal{F}}$ and obtain that for all $\underline{Q} \in \Delta(\underline{\mathcal{F}})$ that are feasible in the discretized fair regression problem (8), we have

$$\text{loss}_\alpha(\widehat{Q}) \le \text{loss}_\alpha(\underline{Q}) + \widetilde{O}(n^{-\beta}) \tag{26}$$

$$\left|\gamma_{a,z}(\widehat{Q})\right| \le \varepsilon_a + \widetilde{O}(n_a^{-\beta}) \quad \text{for all } a \in \mathcal{A},\, z \in [0,1], \tag{27}$$

where in Eq. (27) we have expanded the domain $z \in \mathcal{Z}$ to $z \in [0,1]$ thanks to Eq. (7). Finally, by substituting the solution $\underline{Q}^*$ of problem (8) as $\underline{Q}$ in Eq. (26) and applying Theorem 1, we obtain that for any $Q^* \in \Delta(\mathcal{F})$ that is feasible in the discretized fair regression problem (8), we have

$$\text{loss}(\widehat{Q}) \le \text{loss}(Q^*) + \alpha + \widetilde{O}(n^{-\beta}) = \text{loss}(Q^*) + \widetilde{O}(n^{-\beta}), \tag{28}$$

where the last equality follows by our setting $\alpha = 1/N = O(n^{-\beta})$. The theorem now follows from Eqs. (28) and (27).

D. Algorithm for Fair Regression under Bounded Group Loss

In this section we provide a detailed pseudocode of our algorithm for fair regression under BGL, described at a high level in Section 5.

E. Proof of Theorem 3

The analysis of Algorithm 2 proceeds similarly to the analysis of Algorithm 1.

Iteration complexity. Similarly to our analysis of Algorithm 1 in Theorem 4, we can appeal to Theorem 1, and Lemmas 2 and 3 of Agarwal et al. (2018). While our objective and constraints are for the distributions Q over $[0,1]$-valued predictors $f \in \mathcal{F}$, whereas their analysis is for distributions over $\{0,1\}$-valued classifiers, we can still directly use their Theorem 1, and Lemmas 2 and 3, because they only rely on the bilinear structure of the Lagrangian with respect to Q and $\boldsymbol{\lambda}$ and the boundedness of the objective and constraints, which all hold in our setting. The maximal constraint violation, needed in their Theorem 1, is $\rho \le 1$. Therefore, their Theorem 1 implies that Algorithm 2 terminates in at most $4B^2\ln(|\mathcal{A}|+1)/\nu^2 = O(n^{4\omega}\ln|\mathcal{A}|)$ iterations and finds a ν-approximate saddle point $\widehat{Q}$ of problem (14), albeit sometimes

Algorithm 2 Fair regression with bounded group loss

```
Input: training examples $\{(X_i, Y_i, A_i)\}_{i=1}^n$, slacks in fairness constraint $\widehat{\zeta}_a \in [0,1]$, bound $B$, convergence threshold $\nu$
Define best-response functions:
    $\textsc{Best}_f(\boldsymbol{\lambda}) \coloneqq \arg\min_{f\in\mathcal{F}} L^{\text{BGL}}(f,\boldsymbol{\lambda})$
    $\textsc{Best}_{\boldsymbol{\lambda}}(Q) \coloneqq \arg\max_{\boldsymbol{\lambda}\ge 0,\,\|\boldsymbol{\lambda}\|_1\le B} L^{\text{BGL}}(Q,\boldsymbol{\lambda})$
Set learning rate $\eta = \nu/(2B)$
Set $\boldsymbol{\theta}_1 = \mathbf{0} \in \mathbb{R}^{|\mathcal{A}|}$
for $t = 1, 2, \ldots$ do
    for all $a$ do                                  // Compute $\boldsymbol{\lambda}_t$ from $\boldsymbol{\theta}_t$ and find the best response $f_t$
        $\lambda_{t,a} \leftarrow B\,\exp\{\theta_{t,a}\}/\left(1+\sum_a \exp\{\theta_{t,a}\}\right)$
    end for
    $f_t \leftarrow \textsc{Best}_f(\boldsymbol{\lambda}_t)$
    $\widehat{Q}_t \leftarrow \frac{1}{t}\sum_{t'=1}^t f_{t'}, \quad \widehat{\boldsymbol{\lambda}}_t \leftarrow \frac{1}{t}\sum_{t'=1}^t \boldsymbol{\lambda}_{t'}$          // Calculate the current approximate saddle point
    $\overline{\nu} \leftarrow L^{\text{BGL}}\big(\widehat{Q}_t, \textsc{Best}_{\boldsymbol{\lambda}}(\widehat{Q}_t)\big) - L^{\text{BGL}}(\widehat{Q}_t, \widehat{\boldsymbol{\lambda}}_t)$          // Check the suboptimality of $(\widehat{Q}_t, \widehat{\boldsymbol{\lambda}}_t)$
    $\underline{\nu} \leftarrow L^{\text{BGL}}(\widehat{Q}_t, \widehat{\boldsymbol{\lambda}}_t) - L^{\text{BGL}}\big(\textsc{Best}_f(\widehat{\boldsymbol{\lambda}}_t), \widehat{\boldsymbol{\lambda}}_t\big)$
    if $\max\{\overline{\nu}, \underline{\nu}\} \le \nu$ then          // Terminate if converged
        if $\widehat{\gamma}^{\text{BGL}}(\widehat{Q}_t) \le \widehat{\zeta}_a + \frac{1+2\nu}{B}$ for all $a \in \mathcal{A}$ then
            return $\widehat{Q}_t$
        else
            return null
        end if
    end if
    Set $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta\widehat{\boldsymbol{\gamma}}^{\text{BGL}}(f_t) - \eta\widehat{\boldsymbol{\zeta}}$          // Apply the exponentiated-gradient update
end for
```

it ends up returning *null* instead of $\widehat{Q}$. To prove the theorem we consider two cases.

Case I: There is a feasible solution Q^* to the original problem (4). Given the settings of ν and B, and using Lemmas 2 and 3 of Agarwal et al. (2018), we obtain that the ν-approximate saddle point $\widehat{Q}$ of the empirical problem (14) satisfies

$$\widehat{\text{loss}}(\widehat{Q}) \le \widehat{\text{loss}}(Q) + 2\nu \tag{29}$$

$$\widehat{\gamma}_a^{\text{BGL}}(\widehat{Q}) \le \widehat{\zeta}_a + \frac{1+2\nu}{B} \quad \text{for all } a \in \mathcal{A}, \tag{30}$$

for any distribution Q feasible in the empirical problem (13). Eq. (30) implies that in this case the algorithm returns $\widehat{Q} \ne$ *null*. It remains to argue that statements similar to (29) and (30) hold for true expectations rather than just empirical expectations.

To turn the statements (29) and (30) into matching population versions, we use the concentration result of Lemma 2 similarly as in the proof of Theorem 2. Specifically, we invoke Lemma 2 with $S = Y$, $U = X$, $\mathcal{G} = \mathcal{F}$, and $\varphi(s,t) = \ell(s,t)$, separately for the data distribution (with failure probability $\delta/2$) and the data distribution conditioned on each $A = a$ (with failure probabilities $\delta/(2|\mathcal{A}|)$). We thus obtain that with probability at least $1-\delta$, for all $Q \in \Delta(\mathcal{F})$,

$$\left|\widehat{\text{loss}}(Q) - \text{loss}(Q)\right| \le 4R_n(\mathcal{F}) + \frac{2}{\sqrt{n}} + \sqrt{\frac{2\ln(4/\delta)}{n}}$$

$$\left|\widehat{\gamma}_a^{\text{BGL}}(Q) - \gamma_a^{\text{BGL}}(Q)\right| \le 4R_{n_a}(\mathcal{F}) + \frac{2}{\sqrt{n_a}} + \sqrt{\frac{2\ln(4|\mathcal{A}|/\delta)}{n_a}} \quad \text{for all } a \in \mathcal{A}.$$

By Assumption 2 and our setting of C', this implies that with probability at least $1-\delta$, for all $Q \in \Delta(\mathcal{F})$,

$$\left|\widehat{\text{loss}}(Q) - \text{loss}(Q)\right| = \widetilde{O}(n^{-\omega}) \tag{31}$$

$$\left|\widehat{\gamma}_a^{\text{BGL}}(Q) - \gamma_a^{\text{BGL}}(Q)\right| \le C' n_a^{-\omega} \quad \text{for all } a \in \mathcal{A}. \tag{32}$$

We continue the analysis assuming that the uniform convergence bounds (31) and (32) both hold. Instantiating the bound (31) for $\widehat{\text{loss}}(Q)$ and $\widehat{\text{loss}}(\widehat{Q})$ in Eq. (29) yields, for any Q feasible in the empirical problem (13),

$$\text{loss}(\widehat{Q}) \le \text{loss}(Q) + 2\nu + \widetilde{O}(n^{-\omega}) = \text{loss}(Q) + \widetilde{O}(n^{-\omega}), \tag{33}$$

where the last equality follows by our setting of ν. Similarly, instatianting the bound (32) for $\widehat{\gamma}_a^{\text{BGL}}(\widehat{Q})$ in Eq. (30) yields

$$\gamma_a^{\text{BGL}}(\widehat{Q}) \le \widehat{\zeta}_a + \frac{1+2\nu}{B} + \widetilde{O}(n_a^{-\omega}) \le \zeta_a + \widetilde{O}(n_a^{-\omega}) \quad \text{for all } a \in \mathcal{A}, \tag{34}$$

where the last inequality follows by our setting of B, ν, and C' as well as the bound $\widehat{\zeta}_a \le \zeta_a + C' n_a^{-\omega}$ from Assumption 2.

Above, we assumed that Q was a feasible point of the empirical problem (13). However, assuming that Eq. (32) holds, any feasible solution of the population problem (4) is also feasible in the empirical problem (13) thanks to our setting of C'. Thus, Eqs. (33) and (34) hold for any Q^* feasible in (4), proving the theorem in this case.

Case II: There is no feasible solution Q^* to the original problem (4). In this case, the ν-approximate saddle point $\widehat{Q}$ that the algorithm finds may still satisfy

$$\widehat{\gamma}_a^{\text{BGL}}(\widehat{Q}) \le \widehat{\zeta}_a + \frac{1+2\nu}{B} \quad \text{for all } a \in \mathcal{A}, \tag{35}$$

in which case the algorithm returns $\widehat{Q}$ and the theorem holds vacuously since here is no feasible point Q^*. If the found approximate saddle point does not satisfy Eq. (35), then the algorithm returns *null* and the theorem also holds.

F. Details for Efficient Implementation of Algorithm 1

On the high level, Algorithm 1 keeps track of auxiliary vectors $\boldsymbol{\theta}_t^+, \boldsymbol{\theta}_t^- \in \mathbb{R}^{|\mathcal{A}|N}$. These are used to obtain the vector $\boldsymbol{\lambda}_t \in \mathbb{R}2|\mathcal{A}|N$ played by the $\boldsymbol{\lambda}$-player. The Q-player responds to $\boldsymbol{\lambda}_t$ by playing $h_t \in \mathcal{H}$ (Step 7), which is used to update $\boldsymbol{\theta}_t^\pm$. The average of $\boldsymbol{\lambda}_t$'s and h_t's played so far is the candidate solution for the saddle-point problem

$$\min_{Q \in \Delta} \max_{\boldsymbol{\lambda} \ge \mathbf{0},\, \|\boldsymbol{\lambda}\|_1 \le B} L(Q, \boldsymbol{\lambda}). \tag{36}$$

The algorithm checks the suboptimality of this solution and terminates once the convergence ν is reached (Steps 9–11).

The updates of $\boldsymbol{\theta}_t^+$ and $\boldsymbol{\theta}_t^-$ (Step 12) run in time $O(|\mathcal{A}|N)$ assuming that $\boldsymbol{\gamma}(h_t)$ has already been calculated. Similarly, the transformation of $\boldsymbol{\theta}_t^+$ and $\boldsymbol{\theta}_t^-$ to $\boldsymbol{\lambda}_t$ (Steps 4–6) runs in time $O(|\mathcal{A}|N)$, because the denominator is the same across all coordinates and so it needs to be computed only once. We next show that the remaining operations, except for the two $\textsc{Best}_h$ calls, run in time $O(n \log n + |\mathcal{A}|N)$.

Computation of $\widehat{\text{cost}}(h_f)$ and $\widehat{\boldsymbol{\gamma}}(h_f)$. This computation is implicit in the calculation of Lagrangian in Steps 9–10, and also in the updates of $\boldsymbol{\theta}_t^+$ and $\boldsymbol{\theta}_t^-$ in Step 12. By Eq. (15),

$$\widehat{\text{cost}}(h_f) = \widehat{\mathbb{E}}\Big[\ell_\alpha(Y, \underline{f}(X)) - \ell\big(Y,\ \tfrac{\alpha}{2}\big)\Big],$$

so it can be calculated in time $O(n)$ in a single pass over examples. To calculate $\widehat{\boldsymbol{\gamma}}(h_f)$, we keep the training examples partitioned into $|\mathcal{A}|$ disjoint sets according to their protected attribute A. Let n_a be the number of examples with $A = a$. We sort these examples according to $f(X)$ in time $O(n_a \log n_a)$. Now going through these examples from the largest $f(X)$ value to the smallest allows us to calculate the conditional expectations

$$\widehat{\mathbb{E}}[h_f(X, z) \mid A = a] = \widehat{\mathbb{P}}[f(X) \ge z \mid A = a]$$

across all z in time $O(n_a + N)$. Since we need to do this for all $a \in \mathcal{A}$, the overall runtime is $O(n \log n + |\mathcal{A}|N)$. Finally, to calculate $\widehat{\boldsymbol{\gamma}}$ we also need expectations $\widehat{\mathbb{E}}[h_f(X, z)]$, which can be obtained by taking weighted sums of $\widehat{\mathbb{E}}[h_f(X, z) \mid A = a]$ in time $O(|\mathcal{A}|N)$. Altogether, the running time of computing $\widehat{\text{cost}}(h_f)$ and $\widehat{\boldsymbol{\gamma}}(h_f)$ is therefore $O(n \log n + |\mathcal{A}|N)$.

Computation of $L(h_f, \boldsymbol{\lambda})$ and $L(\widehat{Q}_t, \boldsymbol{\lambda})$. Lagrangian is evaluated in Steps 9–10 to determine whether the algorithm has converged. Note that $L(Q, \boldsymbol{\lambda})$ depends on Q only through $\widehat{\text{cost}}(Q)$ and $\widehat{\boldsymbol{\gamma}}(Q)$. If we have already computed these, $L(Q, \boldsymbol{\lambda})$ can be evaluated in time $O(|\mathcal{A}|N)$. To calculate $L(h_f, \boldsymbol{\lambda})$ for an arbitrary h_f, we first need to calculate $\widehat{\text{cost}}(h_f)$ and $\widehat{\boldsymbol{\gamma}}(h_f)$, so the overall running time is $O(n \log n + |\mathcal{A}|N)$. To calculate $L(\widehat{Q}_t, \boldsymbol{\lambda})$, note that $\widehat{\text{cost}}(\widehat{Q}_t) = \frac{1}{t}\sum_{t'=1}^{t} \widehat{\text{cost}}(h_{t'})$, so we can obtain $\widehat{\text{cost}}(\widehat{Q}_t)$ from $\widehat{\text{cost}}(\widehat{Q}_{t-1})$ at the cost of evaluation of $\widehat{\text{cost}}(h_t)$ and similarly for $\widehat{\boldsymbol{\gamma}}(\widehat{Q}_t)$. Therefore, the first evaluation of the form $L(\widehat{Q}_t, \boldsymbol{\lambda})$ takes time $O(n \log n + |\mathcal{A}|N)$ and each consequent evaluation takes time $O(|\mathcal{A}|N)$.

Computation of $\textsc{Best}_{\boldsymbol{\lambda}}(\widehat{Q}_t)$. The best response of the $\boldsymbol{\lambda}$-player is used in Step 9 to determine the suboptimality of the current solution. Given an arbitrary Q, $\textsc{Best}_{\boldsymbol{\lambda}}(Q)$ returns $\boldsymbol{\lambda}$ maximizing $L(Q, \boldsymbol{\lambda})$ over $\boldsymbol{\lambda} \geq \mathbf{0}$, $\|\boldsymbol{\lambda}\|_1 \leq B$. By first-order optimality, the optimizing $\boldsymbol{\lambda}$ is either $\mathbf{0}$ or puts all of its mass on the most violated constraint among $\widehat{\gamma}_{a,z}(Q) \leq \widehat{\varepsilon}_a$, $\widehat{\gamma}_{a,z}(Q) \geq -\widehat{\varepsilon}_a$. In particular, let $\mathbf{e}^+_{a,z}$ and $\mathbf{e}^-_{a,z}$ denote the basis vectors corresponding to coordinates $\lambda^+_{a,z}$ and $\lambda^-_{a,z}$. The call to $\textsc{Best}_{\boldsymbol{\lambda}}(Q)$ first calculates

$$(a^*, z^*) = \operatorname*{arg\,max}_{(a,z)} \big[|\widehat{\gamma}_{a,z}(Q)| - \widehat{\varepsilon}_a \big]$$

and then returns

$$\begin{cases} B\mathbf{e}^+_{a^*,z^*} & \text{if } \widehat{\gamma}_{a^*,z^*}(Q) > \widehat{\varepsilon}_{a^*} \\ B\mathbf{e}^-_{a^*,z^*} & \text{if } \widehat{\gamma}_{a^*,z^*}(Q) < -\widehat{\varepsilon}_{a^*} \\ \mathbf{0} & \text{otherwise.} \end{cases}$$

Thus, $\textsc{Best}_{\boldsymbol{\lambda}}(\widehat{Q}_t)$ can be calculated in time $O(|\mathcal{A}|N)$ as long as we have $\widehat{\boldsymbol{\gamma}}(\widehat{Q}_t)$, whose computation we have already accounted for within the computation of the Lagrangian of the form $L(\widehat{Q}_t, \boldsymbol{\lambda})$.

F.1. Details for Reduction to Cost-sensitive Classification

By definition of $\widehat{\gamma}_{a,z}(h)$, we have

$$\begin{aligned} \sum_{a,z} \lambda_{a,z} \widehat{\gamma}_{a,z}(h) &= \sum_{a,z} \lambda_{a,z} \left(\frac{1}{p_a} \widehat{\mathbb{E}}\big[h(X,z)\mathbf{1}\{A=a\}\big] - \widehat{\mathbb{E}}\big[h(X,z)\big] \right) \\ &= \widehat{\mathbb{E}}\left[N\mathbb{E}_Z \left[\sum_a \lambda_{a,Z}\, h(X,Z) \left(\frac{\mathbf{1}\{A=a\}}{p_a} - 1 \right) \right] \right] \\ &= \widehat{\mathbb{E}}\left[N\mathbb{E}_Z \left[\left(\frac{\lambda_{A,Z}}{p_A} - \sum_a \lambda_{a,Z} \right) h(X,Z) \right] \right]. \end{aligned} \tag{37}$$

In particular, Eqs. (10) and (37) imply that the minimization of $L(h, \boldsymbol{\lambda})$ is indeed equivalent to minimizing the right-hand side of Eq. (12) as claimed.

F.2. Details for Reduction to Least-squares Regression

The construction of the least-squares regression data set begins with Eq. (12), which states that for $h_f \in \mathcal{H}$

$$\widehat{\text{cost}}(h_f) + \sum_{a,z} \lambda_{a,z} \widehat{\gamma}_{a,z}(h_f) \;=\; \frac{1}{n} \sum_{i \leq n} \sum_{z \in \mathcal{Z}} \frac{1}{N} c_{\boldsymbol{\lambda}}(\underline{Y}_i, A_i, z) h_f(X_i, z). \tag{38}$$

We will now rewrite the inner summation over z as a loss function for the predictor f, parameterized by $\boldsymbol{\lambda}$. More precisely, for any $a \in \mathcal{A}$ and $\tilde{y} \in \tilde{\mathcal{Y}}$, let the "loss" function for a prediction u be defined as

$$g_{\boldsymbol{\lambda}}(\tilde{y}, a, u) = \sum_{z \in \mathcal{Z}, z \leq u} \frac{1}{N} c_{\boldsymbol{\lambda}}(\tilde{y}, a, z).$$

Now consider a specific i in Eq. (38). Let $X \coloneqq X_i$, $A \coloneqq A_i$ and $\tilde{Y} \coloneqq \underline{Y}_i$. Then the summation over z for this specific i can be written as

$$\begin{aligned} \sum_{z \in \mathcal{Z}} \frac{1}{N} c_{\boldsymbol{\lambda}}(\tilde{Y}, A, z) h_f(X, z) \;&=\; \sum_{z \in \mathcal{Z}} \Big[g_{\boldsymbol{\lambda}}(\tilde{Y}, A, z) - g_{\boldsymbol{\lambda}}(\tilde{Y}, A, z - \alpha) \Big] \mathbf{1}\{f(X) \geq z\} \\ &=\; g_{\boldsymbol{\lambda}}(\tilde{Y}, A, \lfloor f(X) \rfloor_\alpha) \\ &=\; g_{\boldsymbol{\lambda}}(\tilde{Y}, A, f(X)). \end{aligned}$$

Plugging this back into Eq. (38), we see that the minimization of Eq. (38) over $h \in \mathcal{H}$ is equivalent to the minimization of the empirical loss under $g_{\boldsymbol{\lambda}}$ among $f \in \mathcal{F}$:

$$\min_{f \in \mathcal{F}} \left[\frac{1}{n} \sum_{i=1}^{n} g_{\boldsymbol{\lambda}}(\underline{Y}_i, A_i, f(X_i)) \right].$$

Solving this problem directly seems to require access to a generic optimization oracle. We instead use a heuristic, where we first pick

$$U_i \in \operatorname*{arg\,min}_{u \in [0,1]} g_{\boldsymbol{\lambda}}(\underline{Y}_i, A_i, u)$$

and then seek to solve the least-squares regression problem

$$\min \sum_{i \le n} (U_i - f(X_i))^2 .$$

To obtain the values U_i we first calculate the values $c_{\boldsymbol{\lambda}}(\tilde{y}, a, z)$ across all $\tilde{y}$, a, and z in the overall time $O(|\mathcal{A}||\tilde{\mathcal{Y}}|N)$. Then, using the definition of $g_{\boldsymbol{\lambda}}$, the minimizer of $g_{\boldsymbol{\lambda}}(\tilde{y}, a, u)$ over u can be found in time $O(N)$ for each specific value of $\tilde{y}$ and a, so all the minimizers can be precalculated in time $O(|\mathcal{A}||\tilde{\mathcal{Y}}|N)$. Thus, preparing the data set for the least-squares reduction takes time $O(|\mathcal{A}||\tilde{\mathcal{Y}}|N)$ and the resulting regression data set is of size n.

F.3. Details for Reduction to Risk Minimization under ℓ

The same reasoning that yielded the reduction to the least-squares regression can be used to derive a reduction to risk minimization under any loss $\ell(y, u)$ that is convex in u. We again begin with computing the minimizers U_i for each $g_{\boldsymbol{\lambda}}(\underline{Y}_i, A_i, f(X_i))$ term. Now suppose that there are two values $\tilde{Y}_{i,1}, \tilde{Y}_{i,2} \in [0,1]$ such that $\frac{\partial}{\partial u}\ell(\tilde{Y}_{i,1}, U_i) \le 0$ and $\frac{\partial}{\partial u}\ell(\tilde{Y}_{i,2}, U_i) \ge 0$ (if ℓ is non-smooth, we can pick arbitrary elements of the subdifferential set). If no such pair exists, then it is not possible to obtain $f(X_i) = U_i$ by minimizing ℓ over any distribution of examples since the gradient can never vanish at U_i. However, if such a pair exists, we can induce weights $W_{i,1}, W_{i,2} \in [0,1]$ such that $W_{i,1} + W_{i,2} = 1$ and

$$W_{i,1} \, \tfrac{\partial}{\partial u}\ell(\tilde{Y}_{i,1}, U_i) + W_{i,2} \, \tfrac{\partial}{\partial u}\ell(\tilde{Y}_{i,2}, U_i) = 0.$$

Hence, we create two weighted examples for each (X_i, A_i, Y_i) triple in our dataset, and solve

$$\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \Big[W_{i,1} \, \ell(\tilde{Y}_{i,1}, f(X_i)) + W_{i,2} \, \ell(\tilde{Y}_{i,2}, f(X_i)) \Big].$$

For instance, for logistic loss, we can always pick $\tilde{Y}_{i,1} = 0$, $\tilde{Y}_{i,2} = 1$ and $W_{i,2} = U_i$. The complexity of this reduction is identical to that of the least-squares reduction above, but the resulting risk minimization problem might be better aligned with the original problem as the experiments in Section 6 show.

G. Additional Experimental Results

In this section, we include further details on our experimental evaluation.

Evaluation on the training sets. In Figure 2 we include the training performances of our algorithm and the baseline methods, including the SEO method and the unconstrained regressors. Our method generally dominated or closely matched the baseline methods. The SEO method provided solutions that were not Pareto optimal on the law school data set.

Implementation of the cost-sensitive oracle. Given an instance of cost-sensitive classification problem, CS oracle optimizes the equivalent weighted binary classification problem on the data $\{W_i, X_i', Y_i\}_{i=1}^n$ with each $X_i' = (x_i, z_i)$ (see Section 3 for the transformation). The oracle aims to solve

$$\min_{h \in \mathcal{H}} \sum_{i=1}^{n} W_i \mathbf{1}\left\{ h(X_i') \ne Y_i \right\}. \tag{39}$$

where every function h in the class $\mathcal{H}$ is parameterized by a vector β and defined as $h(x, z) = \mathbf{1}\left\{\langle \beta, x\rangle \ge z\right\}$ for any input (x, z). Instead of optimizing over the objective in (39), we will consider the following minimization problem with hinge loss. It will be convenient to consider the labels Y_i take values $\{\pm 1\}$ and each predictor h predicts in $\{\pm 1\}$. Then the optimization becomes:

$$\min_{\beta} \sum_{i=1}^{n} W_i \max\left\{0, \tfrac{\alpha}{2} - \langle \beta, x_i\rangle Y_i\right\}$$

Furthermore, we will optimize over β with ℓ_∞ norm bounded by 1. Then we can encode the optimization problem as a linear program:

$$\begin{aligned} &\min_{\beta, t} \quad \sum_i t_i \\ &\text{for all } i \in [n]: \quad t_i \ge 0, \\ &\text{for all } i \in [n]: \quad t_i \ge \tfrac{\alpha}{2} - Y_i \left\langle \beta, x_i \right\rangle, \\ &\text{for all } j \in [d]: -1 \le \beta_j \le 1. \end{aligned}$$

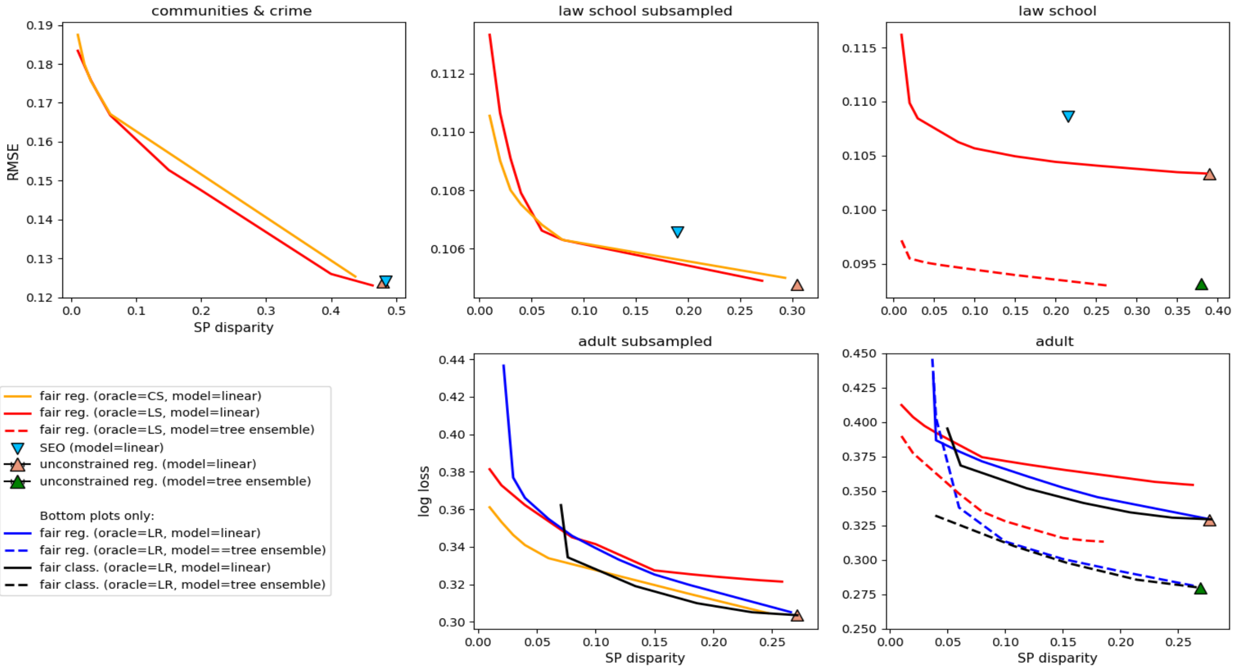

Figure 2. Training loss versus constraint violation with respect to DP. For our algorithm, we varied the fairness slackness parameter and plot the Pareto frontiers of the sets of returned predictors. For the logistic regression experiments, we also plot the Pareto frontiers of the sets of returned predictors given by fair classification reduction methods.

Oracle	Model class	Runtime per call (seconds)
CS	linear	18.94
LS	linear	3.48
LS	tree ensemble	3.60
LR	linear	3.62
LR	tree ensemble	3.69

Table 1. Runtime comparison on the oracles over different model classes. We ran the oracles on a sub-sampled law school data set with 1,000 examples, using a machine with a 2.7 GHz Intel processor and 16GB memory.

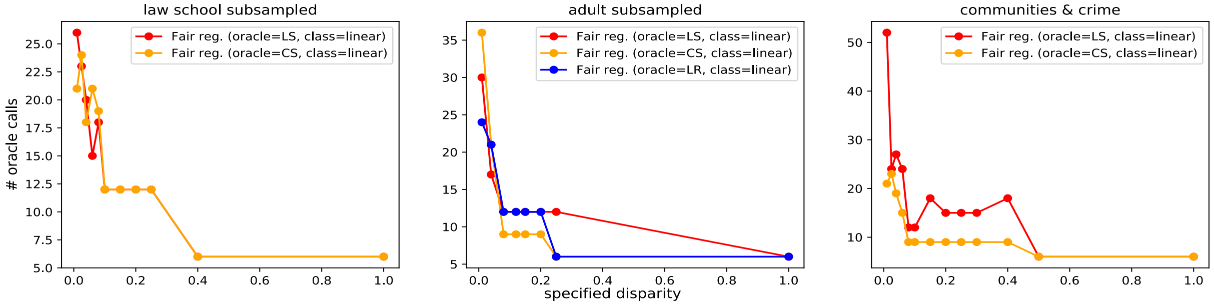

Figure 3. Number of oracle calls versus specified value of fairness slackness.

In our experiments, this optimization problem was solved with the Gurobi Optimizer (Gurobi Optimization, 2018).

Runtime comparison. We performed a comparison on the running time of a single call of the three supervised learning oracles. On a subsampled law school data set with 1,000 examples, we ran the oracles to solve an instance of the BEST_h problem, optimizing over either the linear models or tree ensemble models. The details are listed in Table 1. We also compare the number of oracle calls for different specified values of fairness slackness.