This page is intentionally blank. Proper e-companion title page, with INFORMS branding and exact metadata of the main paper, will be produced by the INFORMS office when the issue is being assembled.

Appendices to *Dynamic Data-Driven Estimation of Non-Parametric Choice Models*

Appendix A: Proofs

Proof of Proposition 1. Note that $D(x,p) = h(x-p) = D(p,x)$. Furthermore, $h(z)$ is clearly convex in z, and hence $h(0) \le h(z)/2 + h(-z)/2 = h(z)$ for any z, so $x = p$ is an optimal solution. □

Proof of Proposition 2. It is easy to see that

$$D(x,p) = \sum_{j\in[m]} \max_{i\in A_j}(x_{ij} - p_{ij}).$$

Now, when $x = p$, $D(x,p) = 0$. However, when $x \neq p$, there exists at least one $j \in [m]$ and $i \in A_j$ such that $x_{ij} \neq p_{ij}$. If $x_{ij} > p_{ij}$, then we know $D(x,p) > 0$. If $x_{ij} < p_{ij}$ then since both $\{x_{ij}\}_{i\in A_j}, \{p_{ij}\}_{i\in A_j} \in \Delta_{|A_j|}$, we have $\sum_{i\in A_j}(x_{ij} - p_{ij}) = 0$, so there must exist some other $i' \in A_j$ such that $x_{i'j} > p_{i'j}$, hence $D(x,p) > 0$ also. □

Proof of Theorem 1. First, notice that $\{\theta_t/\Theta_T\}_{t\in[T]}$ form a set of convex combination weights. Thus, using the convex-concave structure of $\Psi(\cdot,\cdot;p)$ and the definition of $\epsilon^{\Psi}_{\text{sad}}(\bar{x}^\theta_T, \bar{y}^\theta_T; p)$ in (11), we get the standard bound

$$\epsilon^{\Psi}_{\text{sad}}\left(\bar{x}^\theta_T, \bar{y}^\theta_T; p\right) \le \max_{y\in Y} \frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x_t,y;p) - \min_{x\in X}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x,y_t;p).$$

Let us examine the first term on the right hand side. Adding and subtracting $\Psi(x_t,y;p_t)$ for each term in the sum, we can bound this term by

$$\begin{aligned}
\max_{y\in Y}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x_t,y;p) &= \max_{y\in Y}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\left[\Psi(x_t,y;p_t) + \Psi(x_t,y;p) - \Psi(x_t,y;p_t)\right] \\
&= \max_{y\in Y}\left\{\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x_t,y;p_t) + \frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\left[\Psi(x_t,y;p) - \Psi(x_t,y;p_t)\right]\right\} \\
&\le \max_{y\in Y}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x_t,y;p_t) + \max_{y\in Y}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\left[\Psi(x_t,y;p) - \Psi(x_t,y;p_t)\right] \\
&= \max_{y\in Y}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x_t,y;p_t) + \epsilon^{\circ}\left(\{x_t;p_t,\theta_t\}_{t\in[T]};p\right).
\end{aligned}$$

Using a similar strategy for the second term $\min_{x\in X}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x,y_t;p)$, we can get

$$\min_{x\in X}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x,y_t;p) \ge \min_{x\in X}\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\Psi(x,y_t;p_t) - \epsilon^{\bullet}\left(\{y_t;p_t,\theta_t\}_{t\in[T]};p\right).$$

Subtracting this lower bound from the upper bound on the first term then gives us the result. □

Proof of Theorem 2. First observe that for any $t \geq 1$, $(x_t, y), (x, y_t) \in X \times Y$ and p, by Assumption 1,

$$|\Psi(x_t, y; p) - \Psi(x_t, y; p_t)| \leq L_\Psi \|p - p_t\|, \quad |\Psi(x, y_t; p_t) - \Psi(x, y_t; p)| \leq L_\Psi \|p_t - p\|.$$

This implies that

$$\epsilon^\circ \left(\{x_t; p_t, \theta_t\}_{t\in[T]}; p \right) + \epsilon^\bullet \left(\{y_t; p_t, \theta_t\}_{t\in[T]}; p \right) \leq \frac{2L_\Psi}{\Theta_T} \sum_{t\in[T]} \theta_t \|p_t - p\|.$$

We now show the following:

$$a_t \to 0 \implies \frac{1}{\Theta_T} \sum_{t\in[T]} \theta_t a_t \to 0.$$

To get our result, we apply this to the sequence $a_t = 2L_\Psi \|p_t - p\|$, which converges to 0 since $p_t \to p$. Fix some $\epsilon > 0$, and choose $S(\epsilon) \in \mathbb{N}$ sufficiently large such that for $t \geq S(\epsilon)$, $|a_t| \leq \epsilon/3$. Furthermore, choose T sufficiently large such that $\left| \frac{1}{\Theta_T} \sum_{t\in[S(\epsilon)]} \theta_t a_t \right| \leq \epsilon/2$ and $\left| \frac{1}{\Theta_T} \sum_{t\in[S(\epsilon)]} \theta_t \right| \leq 1/2$. We have

$$\left| \frac{1}{\Theta_T} \sum_{t\in[T]} \theta_t a_t \right| \leq \left| \frac{1}{\Theta_T} \sum_{t\in[S(\epsilon)]} \theta_t a_t \right| + \left| \frac{1}{\Theta_T} \sum_{t=S(\epsilon)+1}^{T} \theta_t a_t \right|.$$

The first term is $\leq \epsilon/2$ by our choice of T, and also the second term satisfies

$$\left| \frac{1}{\Theta_T} \sum_{t=S(\epsilon)+1}^{T} \theta_t a_t \right| \leq \frac{1}{\Theta_T} \sum_{t=S(\epsilon)+1}^{T} \theta_t |a_t| \leq \frac{\epsilon}{3\Theta_T} \sum_{t=S(\epsilon)+1}^{T} \theta_t = \frac{\epsilon}{3} \left(1 - \frac{1}{\Theta_T} \sum_{t=1}^{S(\epsilon)} \theta_t \right) \leq \frac{\epsilon}{2}.$$

□

Proof of Proposition 3. We have $|\Psi(x, y; p) - \Psi(x, y; p')| = |\langle B(p - p'), y \rangle| \leq \|y\|_* \|B(p - p')\| \leq \|y\|_* \|B\| \|p - p'\| \leq G_Y \|B\| \|p - p'\|$. □

Proof of Lemma 1. This is immediate since for any $x \in X$, $\Psi(x_t, y_t; p_t) = \min_{x'\in X} \Psi(x', y_t; p_t) \leq \Psi(x, y_t; p_t)$. □

Proof of Corollary 1. Theorem 1 and Corollary 1 immediately imply that $\epsilon^\Psi_{\text{sad}}(\bar{x}^\theta_T, \bar{y}^\theta_T; p)$ is bounded by the right hand side in the result. To get the left hand side, notice that

$$\epsilon^\Psi_{\text{sad}}(\bar{x}^\theta_T, \bar{y}^\theta_T; p) = f(\bar{x}^\theta_T; p) - \min_{x\in X} f(x; p) + \max_{y\in Y} g(y, p) - g(\bar{y}^\theta_T; p) \geq f(\bar{x}^\theta_T; p) - \min_{x\in X} f(x; p)$$

since the optimality gap for the dual problem $\mathcal{D}(p)$ in (9) is always non-negative. □

Proof of Theorems 3, 4 and 5. Theorem 3 follows almost directly from Ho-Nguyen and Kılınç-Karzan (2019, Theorem 1), with minor modifications. Theorem 4 follows from Ho-Nguyen and Kılınç-Karzan (2019, Theorem 2). Theorem 5 follows from Shalev-Shwartz and Kakade (2008, Theorem 2) which, after taking $\ell_t = \theta_t \alpha \omega + \theta_t (h_t - \alpha \omega)$ and $\theta_t = t$, gives the regret bound

$$\sum_{t\in[T]} \theta_t h_t(z_t) - \min_{z\in Z} \sum_{t\in[T]} \theta_t h_t(z) \leq \frac{1}{2} \sum_{t\in[T]} \frac{\theta_t^2 G^2}{\alpha \sum_{s\in[t]} \theta_s} = \frac{1}{\alpha} \sum_{t\in[T]} \frac{tG^2}{t+1} \leq \frac{TG^2}{\alpha}.$$

Dividing by $\Theta_T = T(T+1)/2$ gives the result. □

Proof of Proposition 4 Note that $\nabla_z h_x(z) = B(p-x) + \alpha\omega(z)$, hence $\|\nabla_z h_x(z)\|_* \le \|B(p-x)\|_* + G'$ is uniformly bounded over p, x since these come from a bounded set, so Assumption 2 is satisfied. Assumption 4 holds trivially. $\square$

Appendix B: Existing Approaches to Non-Parametric Choice Estimation

In this appendix, we examine the existing approaches to learn the non-parametric choice model, i.e., infer an appropriate probability vector λ using the data collected via the process outlined in Section 2.1, and demonstrate how they are particular instantiations of our general model. For a fixed subset $\mathcal{A}_j$, $j \in [m]$, we denote the collection of associated choice probabilities as $A_j\lambda = \{\mathbb{P}_\lambda[i \mid \mathcal{A}_j]\}_{i\in\mathcal{A}_j} \in \Delta_{|\mathcal{A}_j|}$.

B.1. Revenue Prediction Approach

Let r_i be the revenue of item $i \in [n]$. Then the expected revenue of an assortment $\mathcal{A} \subset [n]$ under distribution λ is $\sum_{i\in\mathcal{A}} r_i \mathbb{P}_\lambda[i \mid \mathcal{A}]$. Farias et al. (2013) seek to find the worst-case expected revenue from a distribution λ consistent with the given data in the sense that the theoretical probabilities $\mathbb{P}_\lambda[i \mid \mathcal{A}_j] = \langle a_{ij}, \lambda\rangle$ are precisely consistent with their empirical estimates p_{ij}. Since the probabilities $\mathbb{P}_\lambda[i \mid \mathcal{A}]$ are linear in λ, this can be formulated as a linear program (LP)

$$\min_\lambda \left\{ \sum_{i\in\mathcal{A}} r_i \mathbb{P}_\lambda[i \mid \mathcal{A}] : \ A\lambda = p, \ \lambda \in \Delta_{n!} \right\}.$$

We first make a few observations related to this model of Farias et al. (2013). In fact, when $\mathcal{A} = \mathcal{A}_j$ for some $j \in [m]$, we have $\mathbb{P}_\lambda[i \mid \mathcal{A}] = \langle a_{ij}, \lambda\rangle = p_{ij}$ due to the constraints $A\lambda = p$, hence the objective is constant. Thus, the LP becomes a feasibility problem

$$\text{find} \quad \lambda \in \Delta_{n!} \quad \text{s.t.} \quad A\lambda = p. \tag{EC.1}$$

That said, (EC.1) is still computationally intractable even for moderate values of n because it involves $n!$ variables. Nonetheless, the dual of (EC.1) admits the following robust LP interpretation:

$$\max_{\beta,\nu} \left\{ \langle \beta, p\rangle - \nu : \ \max_{\sigma\in S_n} \langle \beta, a(\sigma)\rangle \le \nu \right\}. \tag{EC.2}$$

Note that verifying the feasibility of a solution with respect to the robust constraint in (EC.2), i.e.,

$$\max_{\sigma\in S_n} \langle \beta, a(\sigma)\rangle = \max_\sigma \left\{ \sum_{j\in[m]} \sum_{i\in\mathcal{A}_j} \beta_{ij} a_{ij}(\sigma) : \ \sigma \in S_n \right\} \le \nu \tag{EC.3}$$

is a combinatorial problem of the exact same form as (21). Farias et al. (2013) suggests solving (EC.2) either using the constraint sampling technique (Calafiore and Campi 2005) or by building an approximation to its robust counterpart obtained from approximating the uncertainty sets with an efficiently representable polyhedron.

In fact, (EC.1) can be seen as choosing $\lambda \in \Delta_{n!}$ to minimize a (very harsh) distance measure:

$$\min_{\lambda \in \Delta_{n!}} D(A\lambda, p), \quad D(A\lambda, p) = \begin{cases} 0, & A\lambda = p \\ \infty, & \text{otherwise.} \end{cases} \tag{EC.4}$$

In general, and specifically when the observations are noisy, there is no guarantee that there exists $\lambda \in \Delta_{n!}$ to fit the data p exactly, i.e., $A\lambda = p$. To remedy this, van Ryzin and Vulcano (2015) and Mišić (2016) examine approaches that use less harsh distance measures $D(\cdot,\cdot)$.

B.2. Maximum Likelihood Estimation Approach

van Ryzin and Vulcano (2015) propose the following method to learn λ via maximum likelihood estimation (MLE). We next describe their method and provide an alternative interpretation of their approach as the minimization of a particular distance measure, namely Kullback-Leibler (KL) divergence, between the true distributions $A_j\lambda$ and their empirical estimates p_j. Note that given two positive vectors $p, x \in \mathbb{R}^n$, their KL divergence is $\mathrm{KL}(p, x) := \sum_{i\in[n]} p_i \log(x_i/p_i)$.

By (1), each item-assortment pair $i \in \mathcal{A}_j$ is seen Kq_{ij} times amongst the observations $\{i^k, \mathcal{A}^k\}_{k=1}^K$. Based on this, the log-likelihood of the observation set $\{i^k, \mathcal{A}^k\}_{k=1}^K$ is $\sum_{j\in[m]} \sum_{i\in\mathcal{A}_j} Kq_{ij} \log(\langle a_{ij}, \lambda\rangle)$. Thus, ignoring the constant K factor, the MLE problem is

$$\max_{\lambda} \left\{ \sum_{j\in[m]} \sum_{i\in\mathcal{A}_j} q_{ij} \log(\langle a_{ij}, \lambda\rangle) : \lambda \in \Delta_{n!} \right\}. \tag{EC.5}$$

Throughout, we use the convention that when $q_{ij} = \langle a_{ij}, \lambda\rangle = 0$, we set $q_{ij}\log(\langle a_{ij}, \lambda\rangle) = 0$. This implies that if the optimal solution λ to (EC.5) has $\mathbb{P}_\lambda[i \mid \mathcal{A}_j] = \langle a_{ij}, \lambda\rangle = 0$, then we must have $q_{ij} = 0$ also, i.e., we did not observe any choices of i from $\mathcal{A}_j$ in our data either.

Like (EC.1), the problem (EC.5) is very large, with $n!$ variables. A column generation technique is suggested in van Ryzin and Vulcano (2015) to get around this, i.e., solve (EC.5) on a subset of the variables, and use the optimality conditions to add variables as needed. The MLE column generating subproblem is constructed as

$$\max_{\sigma} \left\{ \sum_{j\in[m]} \sum_{i\in\mathcal{A}_j} \frac{q_{ij} a_{ij}(\sigma)}{\langle a_{ij}, \lambda(S)\rangle} : \sigma \in S_n \right\}. \tag{EC.6}$$

The solution $\lambda(S)$ is optimal if (EC.6) $\le K$, otherwise the column σ^* maximizing (EC.6) is added to the set S, and the process is repeated. Note that (EC.6) has the same form as (21) and (EC.3).

We next demonstrate that the MLE problem (EC.5) admits a nice interpretation between the empirical estimates $\{p_j\}_{j\in[m]}$ and the distributions $\{A_j\lambda\}_{j\in[m]}$. To observe this, let us rewrite the objective in (EC.5) as

$$\sum_{j\in[m]} \sum_{i\in\mathcal{A}_j} q_{ij} \log(\langle a_{ij}, \lambda\rangle) = \sum_{j\in[m]} q_j \sum_{i\in\mathcal{A}_j} p_{ij} \log(\langle a_{ij}, \lambda\rangle)$$

$$= -\sum_{j\in[m]} q_j \underbrace{\sum_{i\in\mathcal{A}_j} p_{ij}\log\left(\frac{p_{ij}}{\langle a_{ij},\lambda\rangle}\right)}_{=\mathrm{KL}(p_j, A_j\lambda)} + \underbrace{\sum_{j\in[m]} q_j \sum_{i\in\mathcal{A}_j} p_{ij}\log(p_{ij})}_{=\text{constant}}$$

where $\mathrm{KL}(a,b)$ is the KL divergence between two probability distributions a and b. Hence, (EC.5) is equivalent to solving

$$\min_{\lambda}\left\{\sum_{j\in[m]} q_j\,\mathrm{KL}(p_j, A_j\lambda):\ \lambda\in\Delta_{n!}\right\}. \tag{EC.7}$$

Thus, by defining $D(A\lambda,p)=\sum_{j\in[m]} q_j\,\mathrm{KL}(p_j,A_j\lambda)$, we see that the MLE approach is equivalent to (EC.4) but with a different distance measure $D(\cdot,\cdot)$.

B.3. Norm-Minimization Approach

As opposed to the approaches outlined in Appendix B.1 and B.2, in order to estimate a non-parametric choice model λ, Mišić (2016) suggest minimizing the ℓ_1-norm of $p-A\lambda$ by solving

$$\min_{\lambda}\left\{\|p-A\lambda\|_1:\ \lambda\in\Delta_{n!}\right\}. \tag{EC.8}$$

In fact, (EC.8) can be cast as an LP, but it is still computationally intractable since the dimension of λ is $n!$. Similar to van Ryzin and Vulcano (2015), Mišić (2016) addresses this computational difficulty via a column generation approach. Again, (EC.8) is of the same form as (EC.4) where the distance measure $D(\cdot,\cdot)$ is selected to be $D(A\lambda,p)=\|p-A\lambda\|_1$. Furthermore, the resulting column generating subproblem is of the form

$$\max_{\sigma}\left\{\sum_{j\in[m]}\sum_{i\in\mathcal{A}_j}\beta_{ij}(S)a_{ij}(\sigma)-\nu(S):\ \sigma\in S_n\right\}, \tag{EC.9}$$

where $\beta(S)$ and $\nu(S)$ are from the dual solution to solving (EC.8) on a subset of columns $\sigma\in S\subset S_n$. Again, this subproblem has the same form as (21), (EC.3) and (EC.6).

Appendix C: Convergence Rates for Error Terms $\epsilon^\circ,\epsilon^\bullet$

In Table EC.1, we state the convergence rate of $\frac{2L}{\Theta_T}\sum_{t\in[T]}\theta_t\|p_t-p\|\to 0$ for different possible rates of $\|p_t-p\|\to 0$, as well as two common choices for θ_t, namely $\theta_t=1$ and $\theta_t=t$. In Section 4, we discuss the effect of the choice of θ on the regret bounds for $\widehat{\epsilon}$.

Proposition EC.1. *The convergence rates in Table EC.1 hold.*

Proof of Proposition EC.1 First, we analyze $S(r,T):=\sum_{t\in[T]}\frac{1}{t^r}$ for $r\neq 1,2$. Observe that

$$\frac{1}{1-r}\left(\frac{1}{(T+1)^{r-1}}-1\right)=\int_{t=1}^{T+1}\frac{1}{t^r}dt\le S(r,T)\le\left(1+\int_{t=1}^{T}\frac{1}{t^r}dt\right)=\frac{1}{1-r}\left(\frac{1}{T^{r-1}}-r\right).$$

rate at which $\frac{2L_\Psi}{\Theta_T}\sum_{t\in[T]}\theta_t\|p_t-p\|\to 0$	$\theta_t=1,\ \Theta_T=T$	$\theta_t=t,\ \Theta_T=\frac{T(T+1)}{2}$
$\|p_t-p\|=O(1/t^r),\ r\in(0,1),$	$\sim 1/T^r$	$\sim 1/T^r$
$\|p_t-p\|=O(1/t)$	$\sim \log(T)/T$	$\sim 1/T$
$\|p_t-p\|=O(1/t^r),\ r\in(1,2),$	$\sim 1/T$	$\sim 1/T^r$
$\|p_t-p\|=O(1/t^2)$	$\sim 1/T$	$\sim \log(T)/T^2$
$\|p_t-p\|=O(1/t^r),\ r>2$	$\sim 1/T$	$\sim 1/T^2$
$\|p_t-p\|=O(\beta^t),\ \beta\in(0,1)$	$\sim 1/T$	$\sim 1/T^2$

Table EC.1 Convergence rate of bound for $\epsilon^\circ+\epsilon^\bullet$.

- We consider the case $\theta_t=1$, $\Theta_T=T$, $\|p_t-p\|=O(1/t^r)$, $r>0$. In this case we have

$$\frac{1}{1-r}\left(\frac{1}{(T+1)^r}-\frac{1}{T+1}\right)\leq\frac{2L}{\Theta_T}\sum_{t\in[T]}\theta_t\|p_t-p\|\sim\frac{1}{T}S(r,T)\leq\frac{1}{1-r}\left(\frac{1}{T^r}-\frac{r}{T}\right).$$

When $r<1$, $1-r>0$ and $1/T=O(1/T^r)$, hence the lower and upper bounds are $\sim 1/T^r$. When $r>1$, $1-r<0$ and $1/T^r=O(1/T)$, hence the lower and upper bounds are $\sim 1/T$.

- Now consider the case $\theta_t=t$, $\Theta_T=2/(T(T+1))$, $\|p_t-p\|=O(1/t^r)$, $r>0$. In this case we have

$$\frac{2}{2-r}\left(\frac{1}{(T+1)^r}-\frac{1}{(T+1)^2}\right)\leq\sim\frac{1}{\Theta_T}\sum_{t\in[T]}\theta_t\|p_t-p\|$$
$$\sim\frac{2}{T(T+1)}S(r-1,T)\leq\frac{2}{2-r}\left(\frac{1}{T^{r-1}(T+1)}-\frac{r-1}{T(T+1)}\right).$$

When $r<2$, we have $2-r>0$ and $1/(T(T+1))=O(1/(T^{r-1}(T+1)))=O(1/T^r)$, hence the lower and upper bounds are $\sim 1/T^r$. When $r>2$, $2-r<0$ and $1/(T^{r-1}(T+1))=O(1/(T(T+1)))=O(1/T^2)$, hence the lower and upper bounds are $\sim 1/T^2$.

- Now consider the case $\theta_t=1$, $\Theta_T=T$, $\|p_t-p\|=O(1/t)$. Then

$$\frac{2L}{\Theta_T}\sum_{t\in[T]}\theta_t\|p_t-p\|\sim\frac{1}{T}\sum_{t\in[T]}\frac{1}{t}\sim\frac{\log(T)}{T}.$$

- Now consider the case $\theta_t=t$, $\Theta_T=2/(T(T+1))$, $\|p_t-p\|=O(1/t^2)$. Then

$$\frac{2L}{\Theta_T}\sum_{t\in[T]}\theta_t\|p_t-p\|\sim\frac{1}{T(T+1)}\sum_{t\in[T]}\frac{1}{t}\sim\frac{\log(T)}{T^2}.$$

Finally, consider the case $\|p_t-p\|=O(\beta^t)$ for $\beta\in(0,1)$. When $\theta_t=1$, $\Theta_T=T$, we have

$$\frac{2L}{\Theta_T}\sum_{t\in[T]}\theta_t\|p_t-p\|=\frac{2L}{\Theta_T}\sum_{t\in[T]}\beta^t=\frac{2L\beta(1-\beta^T)}{T(1-\beta)}\sim 1/T.$$

When $\theta_t=t$, $\Theta_T=2/(T(T+1))$, we have

$$\frac{4L}{T(T+1)}\sum_{t\in[T]}\theta_t\|p_t-p\|=\frac{4L}{T(T+1)}\sum_{t\in[T]}t\beta^t=\frac{4L\beta(1-(T+1)\beta^T+T\beta^{T+1})}{T(T+1)(1-\beta)^2}\sim 1/T^2.$$

□

Appendix D: Online Convex Optimization

In the standard OCO setting, we are given a convex domain Z and a finite time horizon T. In each time period $t \in [T]$, the following takes place:

- we make a decision $z_t \in Z$ based on *past* information from time steps $1, \dots, t-1$ only.
- Then, a convex loss function $h_t : Z \to \mathbb{R}$ is revealed, we suffer loss $h_t(z_t)$ and get some feedback typically in the form of first-order information $\nabla h_t(z_t)$.

It is usually assumed that the functions h_t are chosen possibly by an all-powerful adversary that has full knowledge of our learning algorithm—and we know of only the general class of these functions. As such, it is unreasonable to compare the loss of the player across the time horizon to the best possible loss, which would require full knowledge of h_t in advance of choosing x_t. Instead, the player's sequence of decisions z_t is evaluated against the best fixed decision in hindsight, and the (average) difference is defined to be the *regret*:

$$\frac{1}{T} \sum_{t \in [T]} h_t(z_t) - \min_{z \in Z} \frac{1}{T} \sum_{t \in [T]} h_t(x). \tag{EC.10}$$

The goal in OCO is to design efficient *regret minimizing* algorithms that generate x_t so that the regret tends to zero as T increases. Thus, in OCO we seek algorithms to choose x_t that ensure

$$\frac{1}{T} \sum_{t \in [T]} h_t(z_t) - \min_{z \in Z} \frac{1}{T} \sum_{t \in [T]} h_t(z) \le r(T), \quad \lim_{T \to \infty} r(T) = 0,$$

and the performance of our algorithms is measured by how quickly $r(T)$ tends to 0.

For our work, we consider two simple modifications to the standard OCO setting: *lookahead decisions* and *weighted regret*. More precisely, lookahead decisions allow for the possibility of choosing z_t *with knowledge of* the loss function h_t, while weighted regret modifies (EC.10) to instead be a weighted average

$$\frac{1}{\Theta_T} \sum_{t \in [T]} \theta_t h_t(z_t) - \min_{z \in Z} \frac{1}{\Theta_T} \sum_{t \in [T]} \theta_t h_t(z), \tag{EC.11}$$

where (as before) $\{\theta_t\}_{t \in [T]}$ is a collection of non-negative weights, and $\Theta_T = \sum_{t \in [T]} \theta_t$. The reason for examining weighted regret is clear: by appropriately defining h_t, the two terms in (17) and (18) are actually weighted regret terms. We examine lookahead decisions because the h_t that we define to interpret (17) and (18) as two regret terms, which are $-\Psi(x_t, \cdot; p_t)$ and $\Psi(\cdot, y_t; p_t)$ respectively, actually depend on the decisions x_t, y_t and the data p_t, which we have some control over. In particular, in our setting, it is possible to make lookahead decisions for *one* of the regret terms; however, it is not possible to do it for both, because we must choose either x_t before y_t or x_t before y_t. The primal oracle algorithms that we introduce in Section 4.1 exactly choose x_t after y_t, thus y_t must be chosen in a non-anticipative manner.

Note that Ho-Nguyen and Kılınç-Karzan (2019), Wang and Abernethy (2018), Abernethy et al. (2018) have all examined lookahead and weighted regret in OCO before. In fact, Ho-Nguyen and Kılınç-Karzan (2019) have used these concepts for *standard* JEO. Our work, however, aims to provide an avenue for using OCO to solve the *saddle point* JEO problem (10), where one of the domains involved faces an additional high dimensionality challenge. All of these developments are motivated by the problem of dynamic non-parametric estimation of a choice model. In Sections 2-4, we have introduced this problem formally and discuss the derivation of efficient algorithms for this problem specifically using our general SP JEO framework.

Appendix E: Interpretations of Algorithms from Section 4.2

We discuss interpretations of our methods when applied to $Z = Y$, $z_t = y_t$ for $t \in [T]$ and $h_t(\cdot) = -\Psi(x_t, \cdot; p_t)$.

Remark EC.1. For Theorems 3 and 4, observe that since $x_t = \arg\min_{x\in X} \Psi(x, y_t; p_t)$, $\nabla h_t(z_t) = -\nabla_{y_t}\Psi(x_t, y_t; p_t) = -\nabla\left(\min_{x\in X}\Psi(x, y_t; p_t)\right) = -\nabla g(y_t; p_t)$. Thus, the gradients we compute are simply the gradients of the (negative of) the dual function $g(y;p_t)$ from (9). In other words, if $p_t = p$ for $t\in[T]$, then using Theorem 3 or 4 to minimize the dual regret $\mathcal{R}_y$ is simply performing the well-known Mirror Descent algorithm to solve the dual problem (9). When $p_t \neq p$, these update rules are simply performing Mirror Descent on approximate versions of (9), with built-in guarantees on the error when $p_t \neq p$. ∎

Remark EC.2. For Theorem 5, let us consider the case when $p_t = p$ for $t \in [T]$. Then y_t are computed as

$$y_{t+1} = \arg\max_{y\in Y}\left\{\frac{1}{\Theta_t}\sum_{s\in[t]}\theta_s\Psi(x_s, y; p)\right\}.$$

If, furthermore, $\Psi(x,y;p)$ is linear in x, i.e., it is of the form $\Psi(x,y;p) = \langle x, \boldsymbol{\Psi}(y;p)\rangle - \alpha\omega(y)$, then we can push the sum into the inner product, and since $f(x;p)$ from (8) is of the form $f(x;p) = \max_{y\in Y}\Psi(x,y;p) = \max_{y\in Y}\{\langle x, \boldsymbol{\Psi}(y;p)\rangle - \alpha\omega(y)\}$, by letting $\bar{x}_t^\theta = \frac{1}{\Theta_t}\sum_{s\in[t]}\theta_s x_s$ and using the convex envelope theorem we have

$$y_{t+1} = \arg\max_{y\in Y}\left\{\left\langle \bar{x}_t^\theta, \boldsymbol{\Psi}(y;p)\right\rangle - \alpha\omega(y)\right\}, \quad \boldsymbol{\Psi}(y_{t+1};p) = \nabla f(\bar{x}_t^\theta;p).$$

Now, recalling that

$$x_{t+1} = \arg\min_{x\in X}\Psi(x, y_{t+1}; p) = \arg\min_{x\in X}\langle x, \boldsymbol{\Psi}(y_{t+1};p)\rangle = \arg\min_{x\in X}\langle x, \nabla f(\bar{x}_t^\theta;p)\rangle,$$

$$\text{we deduce}\quad \bar{x}_{t+1}^\theta = (1-\gamma_{t+1})\,\bar{x}_t^\theta + \gamma_{t+1}x_{t+1}, \quad \gamma_{t+1} = \frac{\theta_{t+1}}{\Theta_{t+1}} \in [0,1].$$

Note that this is exactly a Frank-Wolfe update for the current average point $\bar{x}_t^\theta$ on the primal function $f(x;p)$. Therefore, when Ψ is linear in x and $p_t = p$ for all $t \in [T]$, using Theorem 5 to minimize $\mathcal{R}_y$ is equivalent to using the Frank-Wolfe (F-W) algorithm to solve the primal problem (8). See also Abernethy et al. (2018) for an equivalent observation in the case of a particular type of Ψ arising from the convex conjugate of f. Thus, within the general context of the JEO problem (10), we can think of using Theorem 5 as a generalization of the F-W algorithm to the dynamic setup, with built-in error guarantees for $p_t \neq p$. ∎

Appendix F: Rates and Comparison to Frank-Wolfe Methods

In our choice model estimation problem, the high-dimensionality challenge of the domain X necessitates the use of projection/prox-free algorithms. In this respect, our developments for the JEO problem (4) can be compared against the classical Frank-Wolfe (F-W) algorithm (see Jaggi (2013), Freund and Grigas (2016)), which admits guarantees when using *approximate gradients*. Indeed, under certain assumptions which we will carefully examine, we can think of using $p_t \approx p$ as an approximate gradient method, i.e., $\nabla_x D(x,p_t) \approx \nabla_x D(x,p)$. Alternatively, Devolder et al. (2014) considers projection-type first-order methods for smooth functions, and provides guarantees on using approximate gradient oracles within such algorithms. Since projecting onto X defined in (2) for the choice model estimation problem is difficult due to the high dimensionality of the domain X, we will not discuss the methods of Devolder et al. (2014) and instead focus on the F-W algorithm with approximate gradient oracles in Freund and Grigas (2016).

In this appendix, we will do the following.

- We first examine the applicability of the standard F-W method due to the issue of the non-smoothness of the objective function for the norm-based distance measures $D(x,p)$.
- We then show how online Mirror Descent (MD, Theorem 3) can circumvent the non-smoothness and give the corresponding rates obtainable.
- We then present a simple technique to smooth an ℓ_q-norm for $q \in [2,\infty)$ by squaring it. We examine naïvely applying the F-W method to solve (4) and compare the guarantees for F-W with approximate gradients from Freund and Grigas (2016) to the guarantees from using MD in our framework. We find that the data error terms from the naïve F-W method are worse than the ones from using MD in our framework, while the regret bound terms are comparable asymptotically, but involve worse constant factors.
- Finally, we present the so-called 'Nesterov smoothing' technique for more general norms. We examine naïvely applying the F-W method to solve (4) using Nesterov smoothing, and compare the guarantees for F-W with approximate gradients from Freund and Grigas (2016) to the guarantees from using a F-W algorithm derived from our framework (see Remark EC.2). We find that while

the regret bound terms are comparable in both settings, the data error terms for the naïve F-W method require a certain rate of convergence of $\|p_t - p\| \to 0$ to vanish; this is not a problem for the F-W method derived from our framework.

F.1. Smoothness Requirement for the Frank-Wolfe Algorithm

It is known that the F-W method in general *does not* converge on non-smooth objectives; see Nesterov (2018, Example 1) that demonstrates this on a max-type objective function. In the JEO problem (4) when $D(x,p) = \|x-p\|$, e.g., as in Appendix B.3, $D(x,p)$ is non-smooth due to the norm. In addition, the usual convergence of the F-W algorithm relies on a finite curvature constant assumption that related to the smoothness properties of the function. In particular, the curvature constant C_D of a function $D(x,p)$ of the variable x is defined as

$$C_D := \sup_{\substack{x,s\in X\\ \alpha\in[0,1]}} \frac{1}{\alpha^2}\left(D((1-\alpha)x+\alpha s,p) - D(x,p) - \alpha\langle s-x, \nabla_x D(x,p)\rangle\right). \tag{EC.12}$$

It is well-known that when the function is smooth and the domain is bounded, the associated curvature constant is finite; see e.g., Jaggi (2013, Lemma 7). Nevertheless, distance measures $D(x,p)$ of interest in the case of non-parametric choice estimation problem, e.g., from Appendix B are non-smooth. Moreover, we next show that when $D(x,p)$ is set up based on the norm (see Appendix B.3) or the KL divergence as in van Ryzin and Vulcano (2015) (see Appendix B.2) the associated curvature constant of $D(x,p)$ is infinite as well.

Proposition EC.2. *Suppose $n > 2$. For any $q \in [1,\infty]$, the function $D(x,p) = \|x-p\|_q$ has infinite curvature constant (EC.12) for any $p \in X$. Furthermore, when the MLE based weighted KL divergence is used, i.e., $D(x,p) = \sum_{j\in[m]} w_j \,\mathrm{KL}(p_j, x_j)$ for any positive weights w_j, and $p \in X$ such that $p_{ij} > 0$ for all $i \in A_j$, the curvature constant is infinite.*

Proof. We will first show that the curvature constant C_D defined in (EC.12) of $D(x,p) = \|x-p\|$ is infinite for any $p \in X$. Let us choose $x = p$, reserving the choice of $\alpha \in [0,1]$ and $s \in X$ for later. Then $D(x,p) = 0$, $D((1-\alpha)x+\alpha s,p) = \alpha\|s-p\|$, and the subgradients of $D(x,p)$ are $\{y : \|y\|_* \le 1\}$. Thus, for any selection of subgradient mapping $y(\hat{x}) \in \nabla_x D(\hat{x},p)$ we have

$$\begin{aligned}\frac{1}{\alpha^2}\Big[D((1-\alpha)x+\alpha s,p) - D(x,p) - \alpha\langle s-x, y(x)\rangle\Big] &= \frac{1}{\alpha^2}\Big[\alpha\|s-p\| - \alpha\langle s-p, y(x)\rangle\Big]\\ &= \frac{1}{\alpha}\Big[\|s-p\| - \langle s-p, y(x)\rangle\Big].\end{aligned}$$

Note that whenever there is a choice $s \in X$ with $\|s-p\| - \langle s-p, y(p)\rangle > 0$, we can send $\alpha \to 0$ and conclude that the curvature constant C_D is infinite.

To choose the appropriate s, we denote the set of subgradients of $\|\cdot\|_q$ at $s-p$ as $G_{\|\cdot\|}(s-p)$. Observe that for a norm $\|\cdot\|$, if $y \in G_{\|\cdot\|}(s-p)$ then $\|y\|_* \le 1$ and $\langle s-p, y\rangle = \|s-p\|$. Thus, we need to choose $s \in X$ such that $y(x) \notin G_{\|\cdot\|}(s-p)$. To do this, we exploit the following property of ℓ_q-norms. It is simple to check that for $q \in [1,\infty]$ and $y \in G_{\|\cdot\|_q}(s-p)$, we have the property that $y_{ij} > 0 \implies s_{ij} - p_{ij} > 0$. For our selection $y(x)$, first suppose that there exists $i \in \mathcal{A}_j$ such that $y(x)_{ij} > 0$. Then a ranking σ that ranks i last will have $a(\sigma)_{ij} = 0$, so $a(\sigma)_{ij} - p_{ij} \le 0$ because $p_{ij} \ge 0$. We cannot have $p = a(\sigma)$ for all $(n-1)!$ rankings σ that ranks i last (note that $n > 2$); hence, there exists one σ such that $a(\sigma) \neq p$, and we choose $s = a(\sigma)$. This implies that $y(x)_{ij} > 0$ while $s_{ij} - p_{ij} \le 0$, hence $y(x) \notin G_{\|\cdot\|_q}(s-p)$. Now suppose that $y(x)_{ij} \le 0$ for all item-subset pairs (i,j). If $y(x) = 0$, then the result follows trivially by choosing any $s \neq p$. Suppose now there exists some $y(x) < 0$. It is again simple to check that for $q \in [1,\infty]$ and $y \in G_{\|\cdot\|_q}(s-p)$, we have the property that $y_{ij} < 0 \implies s_{ij} - p_{ij} < 0$. Then a ranking σ that ranks i first will have $a(\sigma)_{ij} = 1$, so $a(\sigma)_{ij} - p_{ij} \ge 0$ because $p_{ij} \le 1$. We cannot have $p = a(\sigma)$ for all $(n-1)!$ rankings σ that ranks i first, so there exists one such that $a(\sigma) \neq p$, and we choose $s = a(\sigma)$. This implies that $y(x)_{ij} < 0$ while $s_{ij} - p_{ij} \ge 0$, hence $y(x) \notin G_{\|\cdot\|_q}(s-p)$. Thus, in all cases for $y(x)$, we can choose the appropriate $s \in X$.

Now consider the weighted KL-divergence $D(x,p) = -\sum_{j\in[m]} w_j \sum_{i\in\mathcal{A}_j} p_{ij} \log(x_{ij}/p_{ij})$. We can assume that $p_{ij} > 0$ by simply ignoring terms in the sum for which $p_{ij} = 0$. Choose $x = p$, which ensures that $D(\cdot,p)$ is differentiable at x with $\nabla_x D(x,p)_{ij} = -w_j/x_{ij}$. Then we have

$$
\begin{aligned}
&\frac{1}{\alpha^2}\Big[D((1-\alpha)x+\alpha s,p) - D(x,p) - \alpha\langle s-x, \nabla_x D(x,p)\rangle\Big] \\
&= -\frac{1}{\alpha^2}\sum_{j\in[m]} w_j \sum_{i\in\mathcal{A}_j} p_{ij}\log\left(1-\alpha+\alpha\frac{s_{ij}}{p_{ij}}\right) + \frac{1}{\alpha}\sum_{j\in[m]} w_j \sum_{i\in\mathcal{A}_j}\left(\frac{s_{ij}}{p_{ij}} - 1\right).
\end{aligned}
$$

Note that the second term is bounded by $\frac{1}{\alpha}\left(\sum_{j\in[m]} w_j\right)(\max_{i,j} 1/p_{ij} - 1)$. Choose $s_{ij} = a(\sigma)$ for any $\sigma \in S_n$. Then there exists some i,j such that $s_{ij} = 0$. Sending $\alpha \to 1$ results in $\log\left(1-\alpha+\alpha\frac{s_{ij}}{p_{ij}}\right) \to \infty$, and the second term is bounded, so the curvature constant C_D is infinite. $\square$

Note that our framework proposes to handle the non-smoothness of $D(x,p) = \|x-p\|$ due to the norm by defining $\Psi(x,y;p)$ appropriately, and suggests to utilize a regret-minimizing algorithm to bound the dual regret $\mathcal{R}_y$, from which primal optimality gap bounds can be inferred through (11) (since the dual optimality gap is always non-negative).

F.2. Basic Convergence Rate Using the Mirror Descent Algorithm

We first discuss convergence rates that we can derive within our framework. Since norms are non-smooth, this immediately suggests the utilization of the Mirror Descent algorithm (Theorem 3). Recall that our assumption on D is that it has a representation (5):

$$
D(x,p) = \max_{y\in Y}\left\{\langle B(x-p), y\rangle - \alpha\omega(y)\right\}.
$$

q	Y	B	$\omega(y)$	$\|\cdot\|_\omega$	Ω	G	$\sqrt{\frac{2\Omega G^2}{T}}$
1	$\{\|y\|_\infty \le 1\} \subset \mathbb{R}^N$	I_N	$\frac{1}{2}\|y\|_2^2$	$\|\cdot\|_2$	$N/2$	$\sqrt{2m}$	$\sqrt{\frac{2mN}{T}}$
$1 < q < 2$	$\{\|y\|_{\frac{q}{q-1}} \le 1\} \subset \mathbb{R}^N$	I_N	$\frac{1}{2}\|y\|_2^2$	$\|\cdot\|_2$	$N^{\frac{2-q}{q}}/2$	$\sqrt{2m}$	$\sqrt{\frac{2mN^{\frac{2-q}{q}}}{T}}$
$2 \le q < \infty$	$\{\|y\|_{\frac{q}{q-1}} \le 1\} \subset \mathbb{R}^N$	I_N	$\frac{q-1}{2}\|y\|_{\frac{q}{q-1}}^2$	$\|\cdot\|_{\frac{q}{q-1}}$	$(q-1)/2$	$(2m)^{1/q}$	$\sqrt{\frac{(2m)^{2/q}(q-1)}{T}}$
∞	$\Delta_{2N} \subset \mathbb{R}^{2N}$	$[I_N \ -I_N]$	$\sum_{k\in[2N]} y_k \log(y_k)$	$\|\cdot\|_1$	$\log(2N)$	1	$\sqrt{\frac{2\log(2N)}{T}}$

Table EC.2 Different q-norms and their Mirror Descent constants (Theorem 3).

The first row of Table 1 shows that any norm $\|\cdot\|$ can be written in the form (5) with $\alpha = 0$, $B = I_N$ and $Y = \{y \in \mathbb{R}^N : \|y\|_* \le 1\}$. Define

$$\tilde{X} := \left\{ x \ge 0 : \sum_{i\in A_j} x_{ij} = 1, \ j \in [m] \right\} \subset \mathbb{R}^N,$$

and note that $X \subset \tilde{X}$, $p_t \in \tilde{X}$ for all $t \in [T]$ (by (1)). Thus, assuming that ω is 1-strongly convex with respect to some possibly different norm $\|\cdot\|_\omega$, and defining

$$\Omega = \max_{\|y\|_* \le 1} \omega(y) - \min_{\|y\|_* \le 1} \omega(y), \quad G \ge \max_{x\in X, p' \in \tilde{X}} \|x - p'\|_{\omega,*},$$

using Theorem 3 and Proposition 3 we get the sub-optimality bound

$$\|\bar{x}_T^\theta - p\| - \min_{x\in X} \|x - p\| \le \sqrt{\frac{2\Omega G^2}{T}} + \frac{G_Y \|B\|}{T} \sum_{t\in[T]} \|p_t - p\|, \tag{EC.13}$$

where G_Y is as defined in Proposition 3, which in our case will be 1.

Table EC.2 gives a summary of possible choices of ω for different q-norms, together with the associated constants Ω, G, and the sub-optimality bound. In general, ω should be chosen so that ΩG^2 is small, and the operation $\arg\min_{y\in Y} \{\langle z, y\rangle + \omega(y)\}$ is easy to compute for any z.

F.3. Convergence Guarantees of the Frank-Wolfe Method for Smooth Minimization

For a function $D(x,p)$ which is L-smooth with respect to a norm $\|\cdot\|$, the standard convergence rate for the F-W method, fixing $p_t = p$, is given in Freund and Grigas (2016, Bound 3.1, Eq. 8):

$$D(\bar{x}_T, p) - \min_{x\in X} D(x,p) \le \frac{2L}{T+4} \max_{x,x'\in X} \|x - x'\|.$$

Therefore, if we can use an alternative function D instead of a norm then relate the optimality gap back to the original norm, we can get convergence guarantees for the norm.

In the dynamic setting, however, we need to use an approximate F-W method, since $p_t \neq p$. Essentially, this means that at each iteration t, we have an approximate gradient $\nabla_x D(x,p_t) \approx \nabla_x D(x,p)$, which we require to satisfy

$$\max_{x,x'\in X} |\langle \nabla_x D(x,p_t) - \nabla_x D(x,p), x - x'\rangle| \le \delta_t. \tag{EC.14}$$

We update according to

$$x_t = \arg\min_{x' \in X} \langle \nabla_x D(\bar{x}_t, p_t),\ x' \rangle, \quad \bar{x}_t = (1-\gamma_t)\bar{x}_{t-1} + \gamma_t x_t. \tag{EC.15}$$

When working with approximate gradients, if (EC.14) holds and we choose γ_t appropriately, Freund and Grigas (2016, Theorem 5.1, Proposition 5.1) provides a convergence rate of

$$D(\bar{x}_T, p) - \min_{x \in X} D(x,p) \le \frac{2L}{T+4} \max_{x,x' \in X} \|x - x'\|^2 + \frac{4}{(T+1)(T+2)} \sum_{t \in [T]} (t+1)\delta_t. \tag{EC.16}$$

Since norms are non-smooth, we cannot apply the F-W algorithm directly; recall that the F-W method in general does not converge on non-smooth objectives (see Nesterov (2018, Example 1)). We next discuss two alternative techniques to build smooth approximations to norms.

F.3.1. Smoothing via Squaring the Norm A simple way we can smooth a norm is to square it, i.e., $D(x,p) = \|x-p\|^2$. This does not work for arbitrary norms, but it is known that for ℓ_q-norms, $\frac{1}{2}\|\cdot\|_q^2$ is $(q-1)$-smooth for $q \in [2,\infty)$.

Convergence Rate with Approximate Gradients. Using (EC.16), the naïve F-W algorithm using approximate gradients achieves a bound of (with $D_X := \max_{x,x' \in X} \|x - x'\|_q = (2m)^{1/q}$):

$$\begin{aligned}\|\bar{x}_T - p\|_q - \min_{x \in X} \|x - p\|_q &\le \sqrt{\|\bar{x}_T - p\|_q^2 - \min_{x \in X} \|x-p\|_q^2} \\ &\le \sqrt{\frac{8(2m)^{2/q}(q-1)}{T+4}} + \sqrt{\frac{4(2m)^{1/q}(q-1)}{(T+1)(T+2)} \sum_{t \in [T]} (t+1)\|p_t - p\|_q}. \end{aligned} \tag{EC.17}$$

The first term of (EC.17) is asymptotically comparable to the first term of (EC.13) (third row of Table EC.2), but with worse constants. From Appendix C, we see that the second (data error) term of (EC.17) is asymptotically worse than the data error term in (EC.13) for a variety of rates of convergence of $\|p_t - p\| \to 0$, and also has worse constants.

The data error term is derived as follows. The smoothness of $\|x-p\|_q^2$ is derived from the smoothness of $d(z) = \frac{1}{2}\|z\|_q^2$, which is equivalent to

$$\|\nabla d(z) - \nabla d(z')\|_* \le (q-1)\|z - z'\|.$$

Now observe that

$$\|\nabla_x D(x,p_t) - \nabla_x D(x,p)\|_{q,*} = \|\nabla d(x - p_t) - \nabla d(x-p)\|_{q,*} \le (q-1)\|p_t - p\|_q.$$

Therefore, we have

$$\begin{aligned}\delta_t &:= \max_{x,x' \in X} |\langle \nabla_x D(x,p_t) - \nabla_x D(x,p),\ x - x' \rangle| \\ &\le \max_{x,x' \in X} \{\|\nabla_x D(x,p_t) - \nabla_x D(x,p)\|_{q,*}\, \|x - x'\|_q\} \\ &\le (q-1)\|p_t - p\|_q \max_{x,x' \in X} \|x - x'\|_q = (q-1)\|p_t - p\|_q D_X. \end{aligned}$$

The technique of squaring the norm only works for particular $q \in [2,\infty)$. We next propose a more generally applicable method for smoothing.

F.3.2. Smooth Approximations to General Norms We can smooth more general norms by utilizing the 'Nesterov smoothing' framework outlined in Beck and Teboulle (2012). Interestingly, this also allows us to make use of other algorithms within our framework such as Theorems 4 and 5.

Given a function $g : Z \to \mathbb{R}$, we say that $g_\beta : Z \to \mathbb{R}$ is an β-approximation of g if there exists $\beta_1, \beta_2 \geq 0$, $\beta_1 + \beta_2 = 1$ such that the following holds:

$$\forall z \in Z, \quad g(z) - \beta_1\beta \leq g_\beta(z) \leq g(z) + \beta_2\beta. \tag{EC.18}$$

LEMMA EC.1. *Suppose g, g_β satisfies (EC.18). Then for any $\bar{z} \in Z$,*

$$g(\bar{z}) - \min_{z\in Z} g(z) \leq g_{L,\beta}(\bar{z}) - \min_{z\in Z} g_{L,\alpha}(z) + \beta.$$

Proof of Lemma EC.1. Observe that $g(\bar{z}) \leq g_\beta(\bar{z}) + \beta_1\alpha$. Second, observe that $\min_{z\in Z} g(z) \geq \min_{z\in Z} g_\beta(z) - \beta_2\alpha$. Subtracting the appropriate terms gives us the result. □

Thus, through approximating the non-smooth distance measures $D(x,p)$ with smooth functions, we can use the F-W algorithm and get sub-optimality bounds. For this, we follow the so-called 'Nesterov smoothing' technique outlined in Beck and Teboulle (2012, Section 4.3): by setting $\alpha > 0$ instead of $\alpha = 0$ in (5) and choosing ω to be strongly convex, we get a smooth approximation of the norm with the following guarantee.

LEMMA EC.2. *Let*

$$D(x,p) = \max_{y\in Y} \langle B(x-p), y\rangle, \quad D_\alpha(x,p) = \max_{y\in Y} \left\{\langle B(x-p), y\rangle - \alpha\omega(y)\right\}.$$

Suppose that $\omega(y) \geq 0$ for all $y \in Y$. Then

$$D(x,p) - \alpha \max_{y\in Y} \omega(y) \leq D_\alpha(x,p) \leq D(x,p).$$

Similarly, if $\omega(y) \leq 0$ for all $y \in Y$, then

$$D(x,p) \leq D_\alpha(x,p) \leq D(x,p) + \alpha \max_{y\in Y} |\omega(y)|.$$

Proof of Lemma EC.2. When $\omega(y) \geq 0$ for all $y \in Y$, we have

$$\langle B(x-p), y\rangle \geq \langle B(x-p), y\rangle - \alpha\omega(y) \geq \langle B(x-p), y\rangle - \alpha \max_{y'\in Y} \omega(y').$$

Taking the maximum over $y \in Y$ of all sides gives the first result. The second result is proved similarly. □

q	Y	B	$\omega(y)$	$\max_{y\in Y}\lvert\omega(y)\rvert$	$D_\alpha(x,p)$
1	$\{\lVert y\rVert_\infty \le 1\}\subset\mathbb{R}^N$	I_N	$\frac{1}{2}\lVert y\rVert_2^2$	$N/2$	$\sum_{j\in[m]}\sum_{i\in\mathcal{A}_j} H_\alpha(x_{ij}-p_{ij})$
$1<q<2$	$\{\lVert y\rVert_{\frac{q}{q-1}} \le 1\}\subset\mathbb{R}^N$	I_N	$\frac{1}{2}\lVert y\rVert_2^2$	$N^{\frac{2-q}{q}}/2$	n/a
$2\le q<\infty$	$\{\lVert y\rVert_{\frac{q}{q-1}} \le 1\}\subset\mathbb{R}^N$	I_N	$\frac{q-1}{2}\lVert y\rVert_{\frac{q}{q-1}}^2$	$(q-1)/2$	$H_{\alpha(q-1)}(\lVert x-p\rVert_q)$
∞	$\Delta_{2N}\subset\mathbb{R}^{2N}$	$[I_N\ -I_N]$	$\sum_{k\in[2N]} y_k\log(y_k)$	$\log(2N)$	$\alpha\log\left(\sum_{j\in[m]}\sum_{i\in\mathcal{A}_j} 2\cosh\left(\frac{x_{ij}-p_{ij}}{\alpha}\right)\right)$

Table EC.3 Different q-norms and their smoothings.

Example EC.1. Let $D(x,p)=\lVert x-p\rVert=\max_y\{\langle x-p,y\rangle : \ : \lVert y\rVert_*\le 1\}$. Define the Huber function

$$H_\alpha(r)=\begin{cases}\frac{1}{2\alpha}r^2, & r<\alpha\\ |r|-\frac{\alpha}{2}, & r\ge\alpha.\end{cases} \tag{EC.19}$$

Setting $\omega(y)=\frac{1}{2}\lVert y\rVert_*^2$, we have

$$\begin{aligned} D_\alpha(x,p) &= \max_{y:\lVert y\rVert_*\le 1}\left\{\langle x-p,y\rangle-\frac{\alpha}{2}\lVert y\rVert_*^2\right\}=\max_{\substack{\gamma\in[0,1]\\ y:\lVert y\rVert_*=1}}\left\{\gamma\langle x-p,y\rangle-\frac{\alpha}{2}\gamma^2\right\}=\max_{\gamma\in[0,1]}\left\{\gamma\lVert x-p\rVert-\frac{\alpha}{2}\gamma^2\right\}\\ &=\begin{cases}\frac{1}{2\alpha}\lVert x-p\rVert^2, & \lVert x-p\rVert\le\alpha\\ \lVert x-p\rVert-\frac{\alpha}{2}, & \lVert x-p\rVert>\alpha\end{cases}\\ &=H_\alpha(\lVert x-p\rVert).\end{aligned}$$

This guarantees that $H_\alpha(\lVert x-p\rVert)$ is a $\alpha/2$-approximation of $\lVert x-p\rVert$. Note that $\omega(y)=\frac{1}{2}\lVert y\rVert_*^2$ is not strongly convex in general, which we need in order to use $\omega(y)$ in Theorems 4 or 5. However, if we consider q-norms $\lVert\cdot\rVert=\lVert\cdot\rVert_q$ for $2\le q<\infty$, then it is well-known that $\frac{1}{2}\lVert y\rVert_*^2=\frac{1}{2}\lVert y\rVert_{\frac{q}{q-1}}^2$ is strongly convex (with respect to $\lVert\cdot\rVert_{\frac{q}{q-1}}$) with parameter $(q-1)^{-1}$.

For $q=1$, we can take $\omega(y)=\frac{1}{2}\lVert y\rVert_2^2$ to also get an $N/2$-approximation. For $1<q<2$, we use the same ω to get a $N^{\frac{2-q}{q}}/2$-approximation (which is because $\max_{\lVert y\rVert_{\frac{q}{q-1}}\le 1}\lVert y\rVert_2=N^{\frac{1}{2}-\frac{q-1}{q}}$). However, notice that there is no closed form for D_α, but this is not a problem for our framework as long as we have the max-type representation (5) for D_α. For $q=\infty$, we can use the lifted representation of the ℓ_1-ball into the $2N$-simplex, and set $\omega(y)$ to be the negative entropy, which gets us a $\log(2N)$-approximation. A summary of this is given in Table EC.3; notice that the choices of ω and associated constants are quite similar to Table EC.2. ∎

Convergence Rate with Approximate Gradients. Using a similar argument to the one in Appendix F.3.1, we can derive the following guarantee on the approximate gradients $\nabla_x D_\alpha(x,p_t)$ for functions (5):

$$\begin{aligned}\delta_t &:= \max_{x,x'\in X}|\langle\nabla_x D_\alpha(x,p_t)-\nabla_x D_\alpha(x,p),x-x'\rangle|\\ &\le \max_{x,x'\in X}\lVert\nabla_x D_\alpha(x,p_t)-\nabla_x D_\alpha(x,p)\rVert_*\lVert x-x'\rVert\\ &\le \frac{1}{\alpha}\lVert p_t-p\rVert\max_{x,x'\in X}\lVert x-x'\rVert.\end{aligned}$$

q	Y	$\omega(y)$	$\max_{y\in Y}\lvert\omega(y)\rvert$	$G=D_X$	α^*	$\frac{2G^2}{\alpha^*(T+1)}+\alpha^*\max_{y\in Y}\lvert\omega(y)\rvert$
1	$\{\lVert y\rVert_\infty\le 1\}\subset\mathbb{R}^N$	$\frac{1}{2}\lVert y\rVert_2^2$	$N/2$	$\sqrt{2m}$	$\sqrt{\frac{8m}{N(T+1)}}$	$\sqrt{\frac{8mN}{T+1}}$
$1<q<2$	$\{\lVert y\rVert_{\frac{q}{q-1}}\le 1\}\subset\mathbb{R}^N$	$\frac{1}{2}\lVert y\rVert_2^2$	$N^{\frac{2-q}{q}}/2$	$\sqrt{2m}$	$\sqrt{\frac{8m}{N^{\frac{2-q}{q}}(T+1)}}$	$\sqrt{\frac{8mN^{\frac{2-q}{q}}}{T+1}}$
$2\le q<\infty$	$\{\lVert y\rVert_{\frac{q}{q-1}}\le 1\}\subset\mathbb{R}^N$	$\frac{q-1}{2}\lVert y\rVert_{\frac{q}{q-1}}^2$	$(q-1)/2$	$(2m)^{1/q}$	$\sqrt{\frac{4(2m)^{2/q}}{(q-1)(T+1)}}$	$\sqrt{\frac{4(2m)^{1/q}(q-1)}{T+1}}$
∞	$\Delta_{2N}\subset\mathbb{R}^{2N}$	$\sum_{k\in[2N]} y_k\log(y_k)$	$\log(2N)$	1	$\sqrt{\frac{2}{\log(2N)(T+1)}}$	$\sqrt{\frac{8\log(2N)}{T+1}}$

Table EC.4 Different q-norms and their (smoothed) convergence rates.

Thus, by denoting $D_X := \max_{x,x'\in X}\lVert x-x'\rVert$, (EC.16) gives the following guarantee from using the naïve F-W method for the JEO problem (4):

$$D_\alpha(\bar{x}_T,p)-\min_{x\in X} D_\alpha(x,p)\le \frac{2D_X^2}{\alpha(T+4)}+\frac{4D_X}{\alpha(T+1)(T+2)}\sum_{t\in[T]}(t+1)\lVert p_t-p\rVert.$$

Translating this back into a bound on the norm (Lemmas EC.1, EC.2), we have

$$\lVert\bar{x}_T-p\rVert-\min_{x\in X}\lVert x-p\rVert\le\frac{2D_X^2}{\alpha(T+4)}+\alpha\max_{y\in Y}\lvert\omega(y)\rvert+\frac{4D_X}{\alpha(T+1)(T+2)}\sum_{t\in[T]}(t+1)\lVert p_t-p\rVert. \quad \text{(EC.20)}$$

Let us contrast this guarantee with what is achievable in our primal-dual framework by using Theorem 5 and Proposition 3. Recall from Remark EC.2 that this is actually a variant of F-W. By defining

$$G:=\max_{x\in X,p'\in\tilde{X}}\lVert x-p'\rVert,$$

we get the bound

$$\begin{aligned}\lVert\bar{x}_T^\theta-p\rVert-\min_{x\in X}\lVert x-p\rVert&\le D_\alpha(\bar{x}_T^\theta,p)-\min_{x\in X}D_\alpha(x,p)+\alpha\max_{y\in Y}\lvert\omega(y)\rvert\\&\le\frac{2G^2}{\alpha(T+1)}+\alpha\max_{y\in Y}\lvert\omega(y)\rvert+\frac{2G_Y\lVert B\rVert}{T(T+1)}\sum_{t\in[T]}t\lVert p_t-p\rVert,\end{aligned} \quad \text{(EC.21)}$$

where $G_Y,\lVert B\rVert$ are defined as in Proposition 3; for norms we have $G_Y=1$, $\lVert B\rVert\le 2$. Comparing (EC.20) and (EC.21), we notice that $G\approx D_X$, hence, the first terms are quite comparable. Also, except for an additive factor $\alpha\max_{y\in Y}\lvert\omega(y)\rvert$, these $O(1/T)$ rates in theory are faster than the $O(1/\sqrt{T})$ rates we get from the non-smooth Mirror Descent. However, if we choose α optimally taking this term into account explicitly, we get rates that are asymptotically comparable with those of the non-smooth Mirror Descent, but with worse constants. We summarize this in Table EC.4.

The main difference between the bounds for the naïve F-W method and the variant from our framework displayed in (EC.20) and (EC.21) respectively is, since our continuity requirement on Ψ (Assumption 1) is different to (EC.14), α does not appear in the third data error term of (EC.21). This is significant because choosing $\alpha\propto 1/\sqrt{T}$ to minimize the first two terms now makes the third data error term $\frac{1}{\sqrt{T}(T+1)}\sum_{t\in[T]} t\|p-p_t\|$, which may diverge even if $\|p-p_t\|\to 0$. Thus, to ensure convergence using the naïve F-W method on the smoothed norm with approximate gradients, we need $\|p-p_t\|\to 0$ *sufficiently fast*. In contrast, our primal-dual framework avoids this obstacle completely. We note also that for norms, $G_Y\|B\|\le 2$, so we can also see from Table EC.4 that the constants in the error terms of (EC.21) are better than (EC.20).

Appendix G: Computational Study Details and Supplementary Results

We describe in detail the experimental setup of our computational study. All experiments are conducted on a server with 2.8 GHz processor and 64GB memory, using Python 3.6. Gurobi 8.0 (with default Gurobi settings except we limit the number of threads to 2) is used to solve the integer programming subproblems.

Test Instances. We employ a setup similar to Mišić (2016, Chapter 4.5.3). Our ground truth choice model over $n=10$ items (plus one no-choice option) is a mixed MNL model with K segments. Given mixing probabilities $w\in\Delta_K$ and K sets of utilities $\{u_{i,k}\}_{i\in\{0\}\cup[n]}$, $k\in[K]$, the mixed MNL model chooses an item $i\in\mathcal{A}\subseteq[n]$ with probability

$$\mathbb{P}[i\mid\mathcal{A}]=\sum_{k\in[K]} w_k\frac{u_{i,k}}{u_{0,k}+\sum_{i'\in\mathcal{A}}u_{i',k}}.$$

For each $k\in[K]$, we generate $n+1$ parameters $q_{i,k}\sim U(0,1)$, $i\in\{0\}\cup[n]$ (recall that 0 denotes the no-choice option present in each subset). The utilities $u_{i,k}$ are then set as follows: four randomly chosen $i\in\{0\}\cup[n]$ are set to $u_{i,k}=Lq_{i,k}$ while the rest are set to $u_{i,k}=q_{i,k}/10$. The mixing probabilities $\{w_k\}_{k\in[K]}$ are chosen randomly from the $(K-1)$-dimensional simplex. We test on 100 randomly generated instances of this ground truth model. In the main discussion presented before, we showed results in the dynamic setting for $K=L=5$. In Appendix G.1, we also provide results in the static setting under various parameter regimes $K\in\{1,5,10\}$ and $L\in\{5,10,100\}$, and thus test the effect of different ground truth models on the conclusions drawn. We observe that in these different ground truth models, the conclusions are in general in line with the ones from $K=L=5$ setting; this supports that the conclusions drawn from the dynamic data experiments with this particular choice of ground truth choice model are likely to be valid for other ground truth choice models as well.

We tested each algorithm on 100 different instances of a ground truth model for each parameter combination.

Distance measures. In our main discussion, we consider distance measures $D(x,p)$ based on the ℓ_2-norm. Specifically, we tested $D(x,p)$ from the first and the third rows of Table 1, as well as $D(x,p)=\frac{1}{2}\|x-p\|_2^2$, which has max-type representation $\frac{1}{2}\|x-p\|_2^2=\max_y\left\{\langle x-p,y\rangle-\frac{1}{2}\|y\|_2^2\right\}$. In this appendix, we also consider different choices of distance measures $D(\cdot,\cdot)$ arising from ℓ_1- and

$\|\cdot\|$	Y	B	$\omega(y)$	$D(x,p)$
$\|\cdot\|_1$	$\{\|y\|_\infty \le 1\} \subset \mathbb{R}^N$	I_N	$\frac{1}{2}\|y\|_2^2$	$\sum_{j\in[m]}\sum_{i\in A_j} H_\alpha(x_{ij}-p_{ij})$
$\|\cdot\|_2$	$\{\|y\|_2 \le 1\} \subset \mathbb{R}^N$	I_N	$\frac{1}{2}\|y\|_2^2$	$H_\alpha(\|x-p\|_2)$
$\|\cdot\|_\infty$	$\Delta_{2N} \subset \mathbb{R}^{2N}$	$\begin{bmatrix} I_N & -I_N \end{bmatrix}$	$\sum_{k\in[2N]} y_k \log(y_k)$	$\alpha \log\left(\sum_{j\in[m]}\sum_{i\in\mathcal{A}_j} 2\cosh\left(\frac{x_{ij}-p_{ij}}{\alpha}\right)\right)$

Table EC.5 Smoothed norms used in experiments, where $H_\alpha(r) = \frac{1}{2}r^2$ when $r < \alpha$ and $H_\alpha(r) = |r| - \alpha/2$ otherwise is the Huber function.

ℓ_∞-norms. Table EC.5 shows the distance measures we used in our experiments for ℓ_1-, ℓ_2- and ℓ_∞-norms respectively. Note that $\alpha = 0$ recovers the case for the underlying standard norm. Also, when we used a smoothed norm, we tuned the parameter α to minimize the suboptimality gap bound for the *non-smooth* norm. For example, if we have $D_\alpha(x,p)$ as the smoothed version of the ℓ_∞-norm, then using the F-W algorithm on the static problem, after T iterations we get the bound

$$D_\alpha(\bar{x}_T^\theta, p) - \min_{x\in X} D_\alpha(x,p) \le \frac{16}{\alpha T}.$$

However, translating this back into a sub-optimality gap bound for the ℓ_∞-norm incurs an additional $\log(2N)\alpha$ additive error, thus we have

$$\|\bar{x}_T^\theta - p\|_\infty - \min_{x\in X}\|x-p\|_\infty \le \frac{16}{\alpha T} + \log(2N)\alpha = 8\sqrt{\frac{\log(2N)}{T}},$$

where the last equality is setting $\alpha = 4/\sqrt{\log(2N)T}$ to minimize the upper bound. See Beck and Teboulle (2012) for further details on how this is done for other norms.

Algorithm implementation. We implemented the following solution methods based on our primal-dual framework. For smooth distance measures, i.e., when we have $\alpha > 0$ and ω is strongly convex in the representation (5), we use the following algorithms:

- the naïve Frank-Wolfe (F-W) algorithm updating

$$x_{t+1} = \arg\min_{x'\in X}\langle \nabla_x D(\bar{x}_t^\theta, p_t), x'\rangle, \quad \bar{x}_{t+1}^\theta = \left(1 - \frac{\theta_{t+1}}{\Theta_{t+1}}\right)\bar{x}_t^\theta + \frac{\theta_{t+1}}{\Theta_{t+1}} x_{t+1},$$

with $\theta_t = t$. Note that for smooth distances D, convergence is guaranteed by our discussion in Appendix F.

- The modified F-W algorithm obtained by using Theorem 5 for the dual updates (see Remark EC.2). This is equivalent to updating

$$x_{t+1} = \arg\min_{x\in X}\langle \nabla_x D(\bar{x}_t^\theta, \bar{p}_t^\theta), x\rangle, \quad \bar{p}_t^\theta = \frac{1}{\Theta_t}\sum_{t\in[T]}\theta_t p_t, \quad \bar{x}_{t+1}^\theta = \left(1 - \frac{\theta_{t+1}}{\Theta_{t+1}}\right)\bar{x}_t^\theta + \frac{\theta_{t+1}}{\Theta_{t+1}} x_{t+1},$$

for $\theta_t = t$. Note that in the static case, when $p_t = p$ for every t, this is equivalent to the naïve F-W approach.

- The dual Mirror Descent (MD) algorithm from Theorem 4.

When D is non-smooth, i.e., it is the norm, we employed the dual MD algorithm from Theorem 3. Note that Theorem 3 employed constant step size policies based on constants Ω_Y, G and the maximum iteration count T; computing these constants depends on the particular norm and ω chosen for the domain Y (see Assumption 3), but are not difficult to obtain. For each of these methods and norms, we set a maximum iteration limit of $T = 10,000$.

In the dynamic setting, we use the algorithms outlined above, but only examine the non-parametric estimation model where the distance measure $D(\cdot,\cdot)$ is based on ℓ_2-norm, and the case of $m = 20$ subsets. In the static setting, we examine the effect of using different distance measures based on ℓ_1, ℓ_2, and ℓ_∞ norms as well as the number subsets $m \in \{10, 20, 50\}$. In Appendix G.1, we provide results for all of these cases for the static setting. The conclusions from these static setup experiments are in line with the base case we discuss here; therefore, we defer the results for distance measures based on other norms and the case of $m \in \{10, 50\}$ to Appendix G.1.

Our non-parametric estimation procedure is as follows. We first generate m subsets of $[n]$ of maximum size $\lfloor n/2 \rfloor$ uniformly at random. We append the no-choice option 0 to all of these (consequently the dimension of the domain X is $\dim(X) = 11! \approx 40,000,000$). Using the ground truth model, we compute the p_{train} vector, where $p_{\text{train},ij} = \mathbb{P}[i \mid A_j]$, and A_j is a subset from our training set. In the static setup, we set $p_t = p_{\text{train}}$ at each iteration. In the dynamic setup, we generated a sequence of $p_t \to p_{\text{train}}$, and at each iteration we supply p_t to the algorithm. We initially generate 2000 random choice observations (i, j), where A_j is one of the training subsets chosen randomly, and $i \in A_j$ is chosen with the probability $p_{\text{train},ij}$. We then compute p_1 using these observations according to (1). For $t \geq 2$, we generate $\kappa \in \mathbb{N}$ new observations, then update p_{t-1} with these new observations. We tested various choices of κ between 50 and 1000.

For both data regimes, we terminate training according to the mean absolute error (MAE), defined as $\text{MAE}(p, p') = \frac{1}{\text{length}(p)} \sum_{i,j} |p_{ij} - p'_{ij}|$, where length$(p)$ is the length of the vector p. In the static setup, we terminate training when $\text{MAE}(\bar{x}_t^\theta, p_{\text{train}}) \leq 0.001$, where $\bar{x}_t^\theta$ is the vector of choice probabilities (for our current estimated model) on the subsets used in training after t iterations. In the dynamic setup, we terminate training when $\text{MAE}(\bar{x}_t^\theta, \bar{p}_t^\theta) \leq 0.001$.

Performance metrics. We compare the effectiveness of our methods using three criteria: model fit, sparsity, and algorithm efficiency.

To evaluate model fit, we examine the mean absolute error of choice probabilities on subsets generated independently from the training set. Specifically, we generate 100 subsets of $[n]$ of maximum size $\lfloor n/2 \rfloor$ uniformly at random (independently to the training subsets), and append the no-choice option 0 to each of them. We compute the vector of choice probabilities p_{test} using our ground truth model. Letting $\bar{x}$ be the choice probabilities on the test subsets computed from the estimated choice model at training termination, we calculate $\mathrm{MAE}(\bar{x}, p_{\text{test}})$.

To evaluate sparsity of our estimated model, we examine the number of different rankings σ with positive probability $\lambda(\sigma) > 0$ in our estimated model. Sparsity is very much desired for non-parametric models, since choice probabilities for sparser models can be computed more efficiently.

To evaluate algorithm efficiency, we examine the number of iterations until the termination criterion is reached. While we could have used solution time as another metric for this purpose, we observed in the static setup that solution time is highly correlated with the number of iterations. Runtimes are affected by how fast the combinatorial subproblem (21) is solved, but the focus of our work is not on this aspect, hence in our discussions we focused on the number of iterations as a more accurate representation of algorithm efficiency for our purposes.

G.1. Static Estimation Results

We use the static setup to compare the effect of different parameters (e.g., K, L, and the norm) on algorithm performance. We compare the F-W algorithm (recall that the naïve and modified versions, i.e., $\mathrm{FW}_{\text{naïve}}$ and FW_{dyn}, are equivalent in the static setup, and so we simply refer to it as F-W in this subsection), as well as the dual MD algorithm for both original non-smoothed norm MD_{ns} and its smoothing $\mathrm{MD}_{\text{smth}}$.

Figures EC.1, EC.2 and EC.3 plot the test MAE, the average number of rankings and the average number of iterations for each of the three solution methods when using different norms, varying $K \in \{1,5,10\}$ and fixing L to respectively $5, 10$ and 100 while fixing $m = 20$. We observe that the three methods have roughly the same test MAE for each of the norms, with perhaps the smoothed dual MD method performing slightly better when D is based on certain norms, but whenever this occurs, it terminates in more iterations and with a denser model. On the other hand, the non-smooth dual MD method clearly learns a sparser model, and clearly terminates in less number of iterations than the other two methods, which are similar in these two metrics. Therefore, we conclude that, regardless of the type of norm used in the estimation procedure, the non-smooth dual MD method is superior in the static setting, since it manages to learn a sparser model more efficiently, while maintaining the same model fit. This conclusion holds for all combinations of $K \in \{1,5,10\}$ and $L = \{5,10,100\}$. In particular, they are consistent with our findings for the dynamic setting results shown in Figure 1.

In Figure EC.4, we examine the effect of m by varying $m \in \{10,20,50\}$ while fixing $K = L = 5$. We observe that as m increases, the test MAE goes down, but the model sparsity and the number of iterations to convergence increases across all different approaches and norms used. This is as expected, since having more training subsets should allow us to fit better models, but increases the dimension of the choice probability set X. Our conclusions regarding the comparison of different approaches remain essentially the same: the non-smooth dual approach still outperforms the others.

Figure EC.5 shows the average solution times and the average subproblem times for each method and norm, fixing $m = 20$ and $K = L = 5$. From Figure EC.5 we conclude that the number of iterations and the overall solution time is strongly correlated (contrast it with Figure EC.4(b)) and the non-smooth dual approach is still outperforming the other two with respect to overall solution time. We do not believe that the variation in average subproblem time is the result of any inherent property of the methods used or norms. Moreover, the average subproblem solution times are quite small, and thus the variations in subproblem solution times are relatively small.

G.2. Additional Remarks

In our numerical experiments, we observed that the average number of rankings and iterations are highly correlated. In fact, the Spearman correlation between these two metrics in the static setting with $K = L = 5$ is ≈ 0.922, thus we conclude that the average number of iterations is a good proxy for model sparsity. This can be seen in the theory: all of our algorithms start with one ranking, and at each iteration they add at most one ranking to the estimated model, which provides an explicit bound on the sparsity of the estimated choice they provide at the end.

Finally, one can argue that the algorithms $\mathrm{FW}_{\text{naïve}}$, $\mathrm{FW}_{\mathrm{dyn}}$, and $\mathrm{MD}_{\mathrm{smth}}$ are much simpler to implement than the non-smooth MD algorithm ($\mathrm{MD}_{\mathrm{ns}}$), since there is essentially no parameter tuning aside from computing the smoothing parameters α for D in Table EC.5. On the other hand, $\mathrm{MD}_{\mathrm{ns}}$ algorithm additionally requires tuning the selection of step size, knowing the time horizon T, and computing the constants Ω_Y, G (which in turn affects the smoothing parameters). However, for our particular choice model estimation problem, these quantities are quite straightforward to compute, and our analysis and numerical results are based on such 'textbook' constant step size policies derived from these, which worked quite well. In terms of performance, we see that the extra sophistication in non-smooth MD ($\mathrm{MD}_{\mathrm{ns}}$) can significantly outperform F-W ($\mathrm{FW}_{\text{naïve}}$ or $\mathrm{FW}_{\mathrm{dyn}}$) and smoothed MD ($\mathrm{MD}_{\mathrm{smth}}$).

References

Abernethy J, Lai KA, Levy KY, Wang JK (2018) Faster rates for convex-concave games. *COLT 31*, 1595–1625.

Beck A, Teboulle M (2012) Smoothing and first order methods: A unified framework. *SIAM Journal on Optimization* 22(2):557–580.

Calafiore G, Campi M (2005) Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming* 102(1):25–46.

Devolder O, Glineur F, Nesterov Y (2014) First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming* 146(1):37–75.

Farias V, Jagabathula S, Shah D (2013) A nonparametric approach to modeling choice with limited data. *Management Science* 59(2):305–322.

Freund RM, Grigas P (2016) New analysis and results for the frank–wolfe method. *Mathematical Programming* 155(1):199–230.

Ho-Nguyen N, Kılınç-Karzan F (2019) Exploiting problem structure in optimization under uncertainty via online convex optimization. *Mathematical Programming* 177(1):113–147.

Jaggi M (2013) Revisiting Frank-Wolfe: Projection-free sparse convex optimization. *Proceedings of the 30th International Conference on Machine Learning*, volume 28(1), 427–435.

Mišić VV (2016) *Data, models and decisions for large-scale stochastic optimization* (Chapter 4). Ph.D. thesis, Massachusetts Institute of Technology.

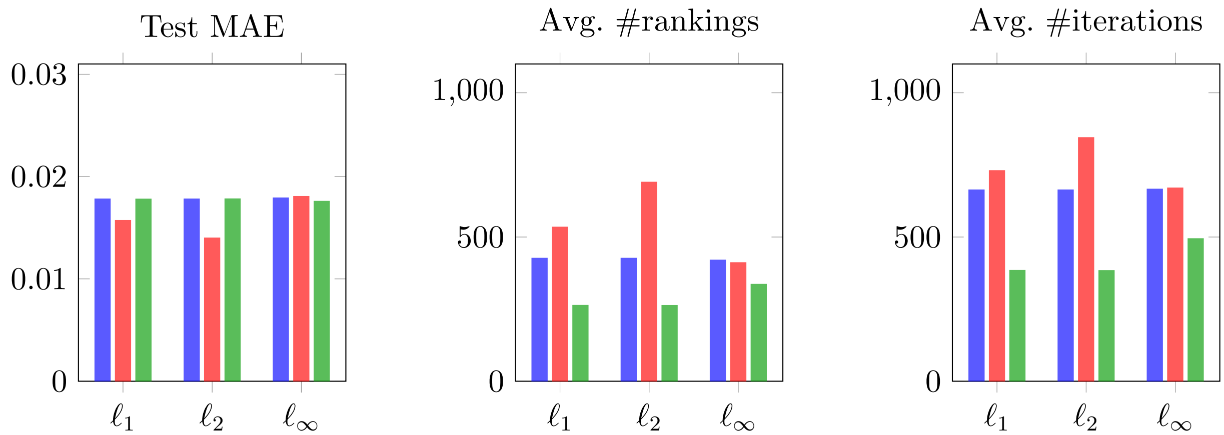
Nesterov Y (2018) Complexity bounds for primal-dual methods minimizing the model of objective function. *Mathematical Programming* 171(1):311–330.

Shalev-Shwartz S, Kakade SM (2008) Mind the duality gap: Logarithmic regret algorithms for online optimization. *NeurIPS 21*, 1457–1464.

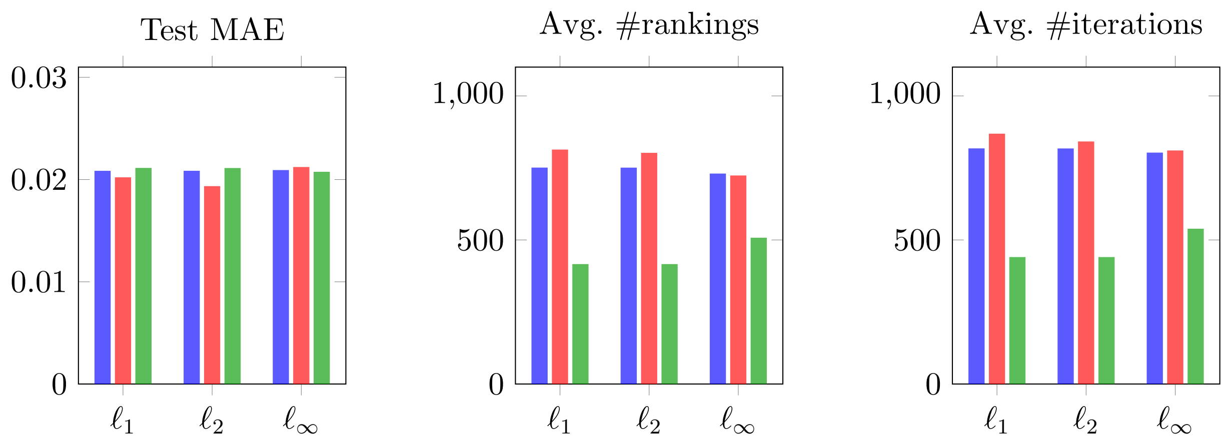
van Ryzin G, Vulcano G (2015) A market discovery algorithm to estimate a general class of nonparametric choice models. *Management Science* 61(2):281–300.

Wang JK, Abernethy JD (2018) Acceleration through optimistic no-regret dynamics. *NeurIPS 31*, 3824–3834.

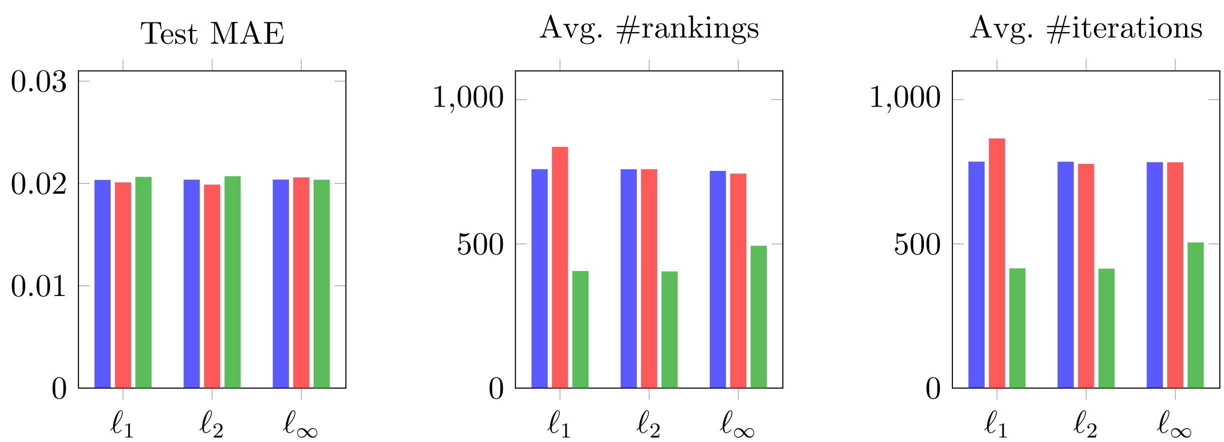
Figure EC.1 Performance metrics in the static setup for different K, fixing $L = 5, m = 20$.



(a) $K = 1$

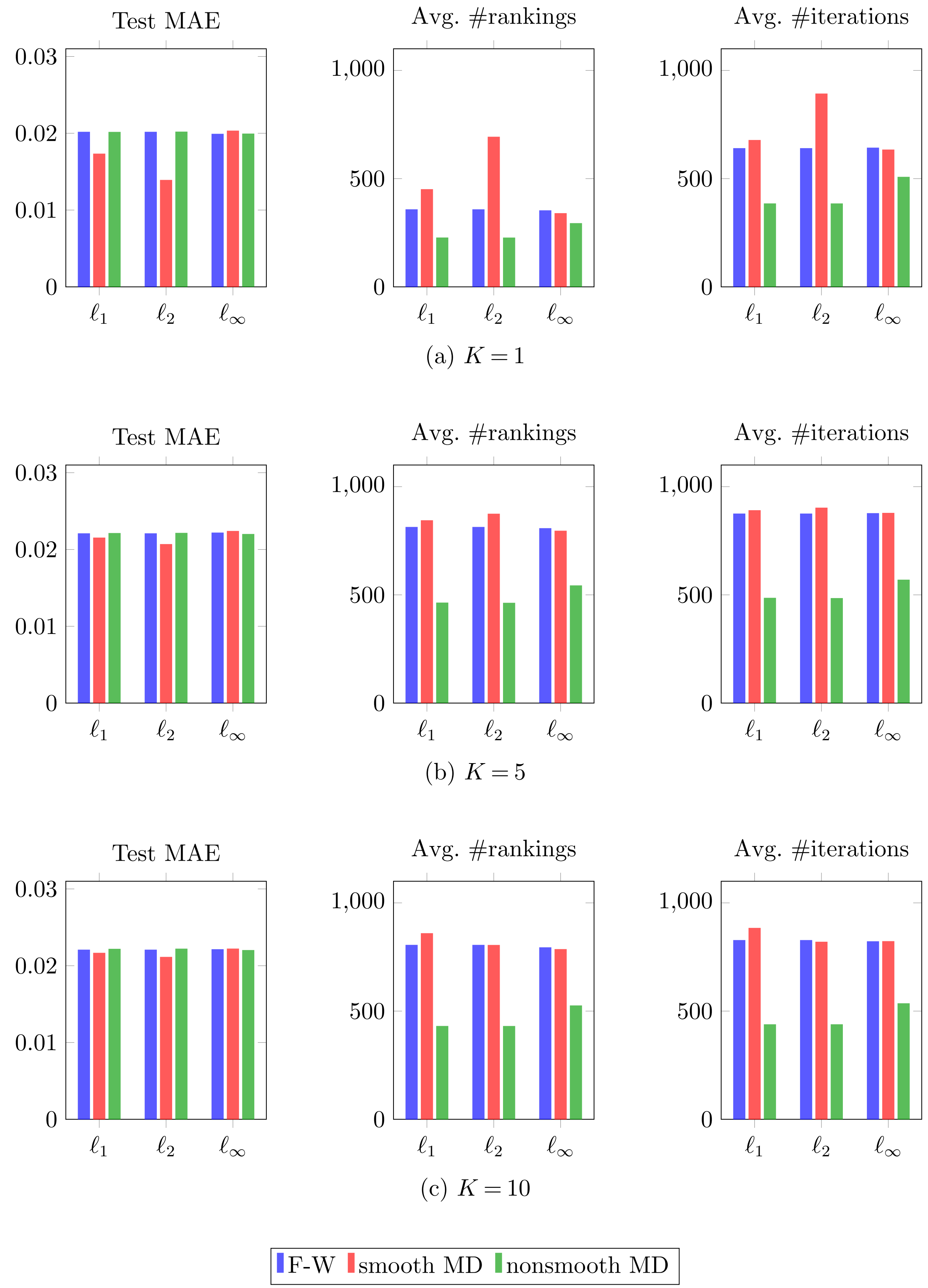


(b) $K = 5$



(c) $K = 10$

Figure EC.2 Performance metrics in the static setup for different K, fixing $L = 10, m = 20$.

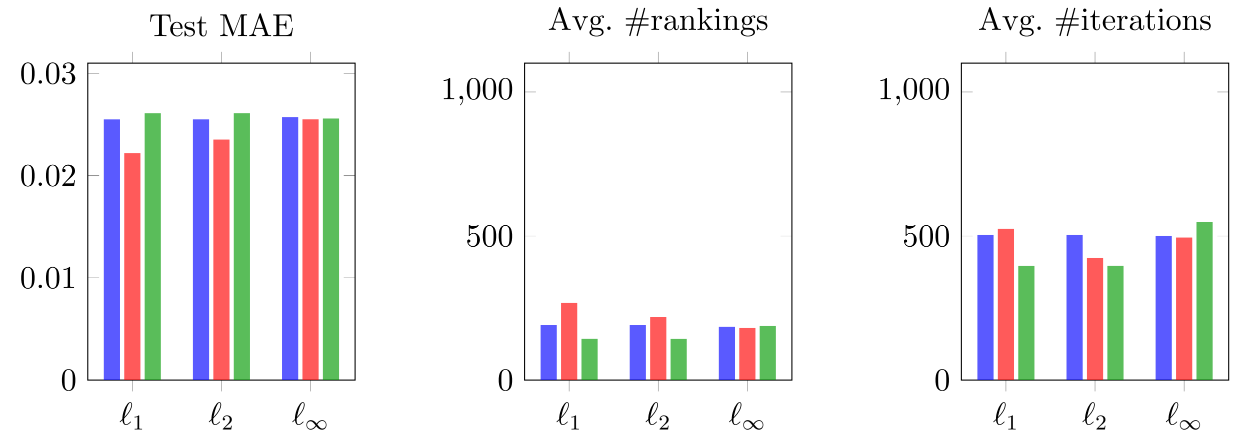


(a) $K = 1$

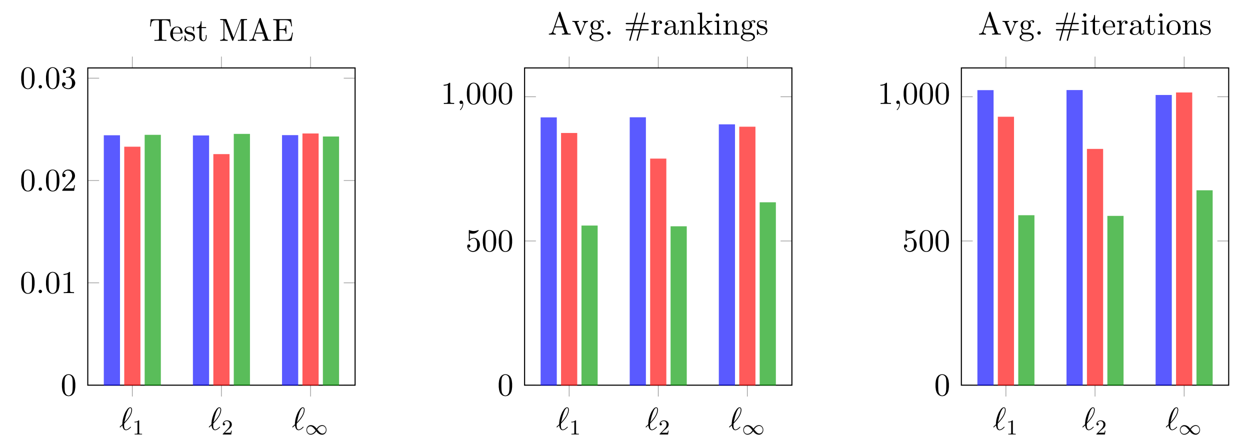
(b) $K = 5$

(c) $K = 10$

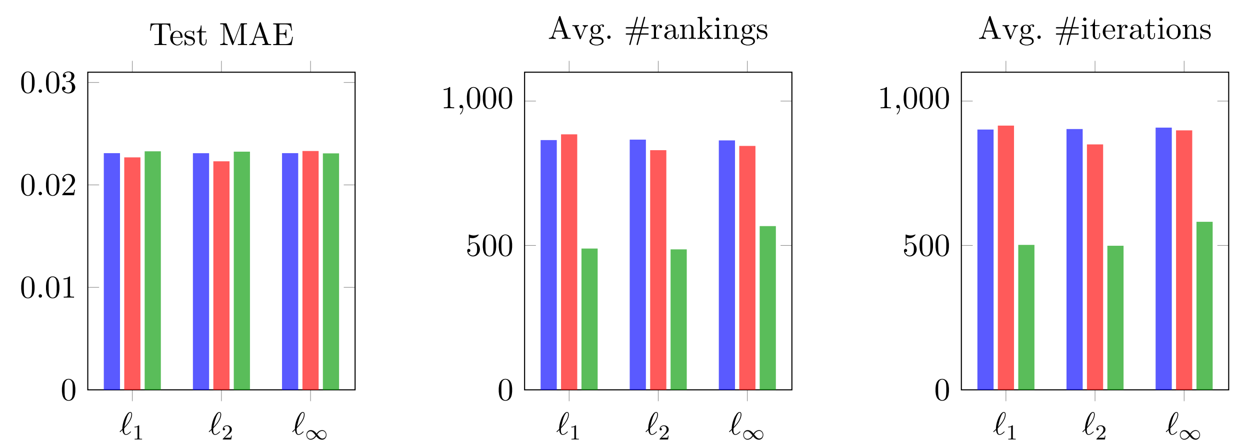
Figure EC.3 Performance metrics in the static setup for different K, fixing $L = 100, m = 20$.



(a) $K = 1$

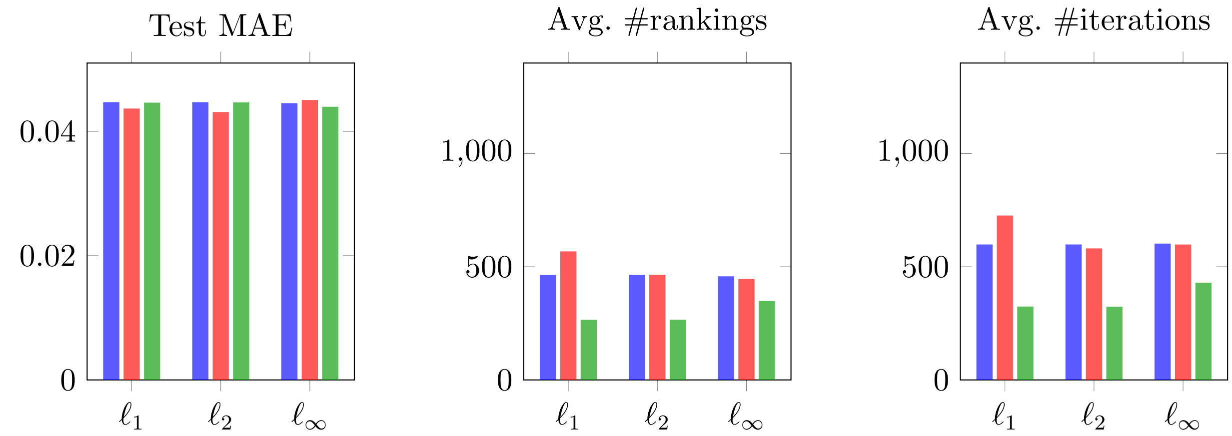


(b) $K = 5$

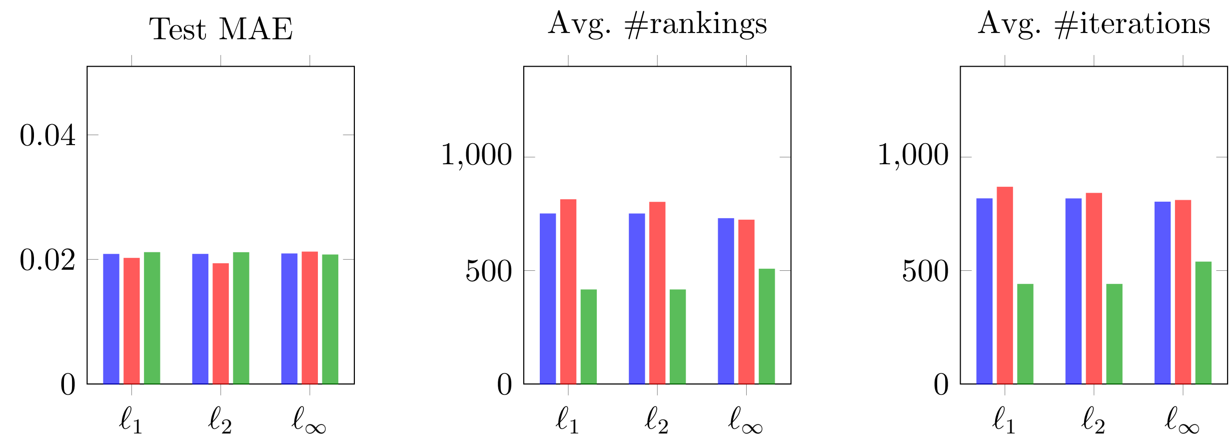


(c) $K = 10$

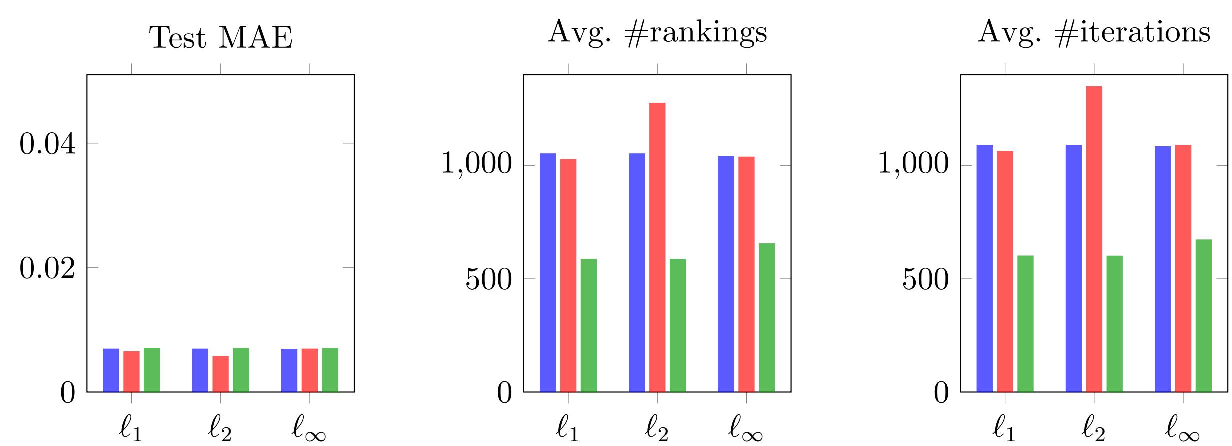
Figure EC.4 Performance metrics in the static setup for different m, fixing $K = L = 5$.



(a) $m = 10$



(b) $m = 20$



(c) $m = 50$

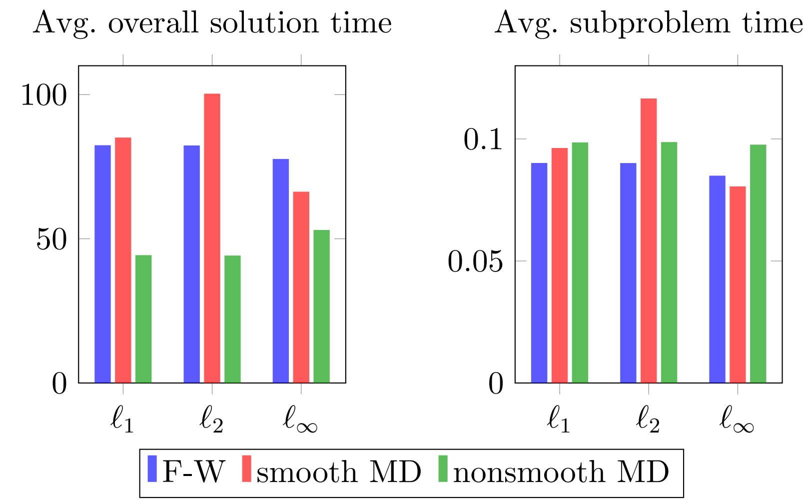


Figure EC.5 Overall solution times and subproblem times per iteration (both in seconds) in the static setup for $m = 20$ and $K = L = 5$.