We would like to thank the reviewers for their detailed reading of our work and constructive comments. We will take into account all their valuable remarks on the general presentation and readability of the paper. We now address their more specific concerns separately for each reviewer in order of appearance.

Reviewer #1. We will clarify the definition of d in line 111 by defining $d = \max_{e \in \mathcal{U}} |\mathcal{F}(e)|$ in a separate equation.

Please clarify where you assume that $w_S >= 1$ (line 449). We use this assumption in the inequality just below where we can assume that $(1 + 1/w_S)^{w_S} \geq 2$ since $w_S \geq 1$.

Why does Alg 4 stop after at most N updates if $N^{pred} <= B$ (proof of Thm 2) ? Alg 4 always stops after at most N updates, regardless of the prediction. This is because the true instance contains N skiing days so N new constraints are revealed online. After the last constraint has arrived, the algorithm always stops.

In the proof of Lem 15 between line 651 and 652, the second equality is actually an inequality. Also clarify why the prior inequality holds. Agreed, the second equality is an inequality. For the prior inequality, recall that $S(k)$ is the value of $\sum_{l=t_0}^{\infty} x_l$ at time t after k big updates. Thus, $S(k) = \sum_{l=t_0}^{t} x_l$. Since we are considering an update due to a request j acknowledged at time t_0 by the predicted solution, it must be that $t(j) \leq t_0$ hence we have that $\sum_{l=t(j)}^{t} x_l \geq \sum_{l=t_0}^{t} x_l$. Noting this, we can lower bound the first line replacing $\sum_{l=t(j)}^{t} x_l$ by $\sum_{l=t_0}^{t} x_l = S(k)$.

Reviewer #2. *I am not sure I would call the approach black box in the strict sense, as there is still a fair amount of tailor made analysis for each setting (and the algorithm modification, although intuitive, still requires a deep understanding of the problem).* The term black-box was used to underline the fact that the designed algorithms do not make any assumption on the quality of the prediction and use it in a “black-box manner”. We will clarify this by avoiding the term black-box in the final version of the paper. Moreover, we agree that modifying the primal dual algorithm requires a deep understanding of the algorithm itself. However, for all problems we considered, only the additive term in the update of the primal variables needs to be modified (in a similar vein). Hence it is true that the approach is not a black box in the strict sense but the basic idea of the framework is a great help in obtaining the result.

Covering problems are generally very well understood in the online literature. Are there any interesting covering problems left for others to apply this technique? There are actually a lot of covering problems that are solved via the online primal-dual technique like for instance the weighted caching and load balancing problems (see [4] in our bibliography). In addition to that, we suspect that our work might also provide insights not only for covering problems but also for packing problems. A concrete example of that is the revenue maximization in ad auctions problem which also has an optimal $\frac{e}{e-1}$-competitive algorithm and whose dual is similar to the TCP acknowledgement problem (see [5]). Moreover, another interesting direction may be to incorporate predictions into the primal dual technique when it is used to solve covering problems where the objective function is non linear (e.g. convex). We will add a separate paragraph discussing these future directions in the final version of the paper.

Reviewer #3. *how do you recover optimal integral solutions, from the fractional solutions you get with your algorithms?* We did not emphasize this point since it already appeared in previous literature (except for the Bahncard problem). However, we agree that this is a very crucial point and deserves more emphasis. To obtain our results, we only modify the algorithm that builds the fractional solution but not the rounding scheme. For the ski rental, Bahncard and TCP acknowledgement problems it is possible to round (randomly) the solution in an online manner such that the expected cost of the integral solution obtained is equal to the cost of the fractional solution. The rounding scheme for ski rental and TCP can be found in the reference [5] and in Lemma 14 of our paper for the Bahncard problem. For set cover a multiplicative $O(\log n)$ factor is lost during the rounding due to the integrality gap of the linear program. The rounding scheme for the online set cover problem is described by Alon et al. in [1].

do standard datasets exist for benchmarking online algorithms, likely with data coming from real industrial applications? To the best of our knowledge there are no standard datasets to benchmark online algorithms. In the new field of learning augmented online algorithms the only datasets used so far were Brighkite and CitiBikeNYC by Lykouris and Vassilvitskii in [19]. The aforementioned datasets were not used in our work because the access requests were too sparse to model the TCP requests of a large server. Although we could not find a real dataset we would like to underline that our artificial dataset was created to mimic a real TCP packet instance following common assumptions regarding packet distribution as noted in [11,20,22,30].

could the authors benefit from distribution errors coming from the predictors? We would like to emphasize that the main challenge that we address is to design a learning augmented algorithm which uses the prediction without any assumption on its quality and on the type of prediction errors. That being said, once additional assumptions are made, it is very possible that one can twist our results to obtain better guarantees. We believe that this is an intriguing direction with possibly interesting results. However, this direction is complementary to the area of learning augmented algorithms as it approaches the problem under a stochastic optimization point of view.