Thank you to all three reviewers for their thoughtful reviews. Please find below responses to specific comments.

Reviewer 1. Thank you for the overall positive review.

- *lines 155 to 159: this is a bit of a repeat, and somewhat ill-placed:* We will modify as suggested.
- *indicate whether there are other interesting measures one could consider:* Another measure is the distance to monotonicity, which corresponds to the difference between n and the longest increasing sequence. Considering other measures of sortedness is an interesting direction for future work.
- *R should be defined; r_1 should be defined as 0:* We will add a line in the algorithm for initializing R. By definition of $r_t, r_1 = 0$ since there is no $t' < 1$.
- *line 192 how are ties within the argmin dealt with?* Ties for argmin can be broken arbitrarily.
- *In the introduction of the event $\mathcal{O}_\sigma$, ...:* The conditional event in the probability is for a fixed permutation over $t-1$ elements. We will fix the text to "conditionally uniformly distributed" as suggested.
- *line 227: what bound do you refer to at the end of the sentence?* The bound refers to the expression for the conditional probability between lines 223 and 224, we will make this clearer.
- *line 247: shouldn't we say: is popular at time t w.p. at most $e^{-\Omega(\alpha)}$?* Yes, this should be Ω instead of O.
- *line 277: should $O(n\sqrt{n})$ be $\Omega(n\sqrt{n})$?* Yes, we will fix this, thank you for catching these typos.

Reviewer 4. It seems that a main concern is that "the total number of inversions is quite high" which makes it "hard to get excited about the results." A better way to look at the result is to see what fraction of the secretaries are incorrectly ranked with respect to a randomly picked secretary. By allocating at random, each secretary is incorrectly ranked with respect to $1/2$ fraction of remaining secretaries. With our algorithm, each secretary is incorrectly ranked with respect to only a $1/\sqrt{n}$ fraction of remaining secretaries.

- *Is there a different natural approach that does not work as well?*
  One of our first attempts to this problem was an inductive bucketing algorithm where we use the first s elements to partition the positions within buckets in which we add future elements. The non-inductive and inductive bucketing algorithms obtain $O(n^{9/5})$ and $O(n^{5/3})$ inversions respectively.
- *Does the algorithm perform worse without using noise, or is it just helpful for the analysis?*
  This is a good question. We do not know what is the number of inversions obtained by the algorithm without noise, analyzing this algorithm would require new non-trivial techniques. Adding noise is a central idea needed for our analysis.

Reviewer 5. The two additions to the model that are mentioned in the review are very interesting questions that indeed provide a closer account for issues in real-world applications. We can extend our results to account for both of these additions. In the final version of the paper, we will add these additional results. We do also note that this is the first paper studying the secretary ranking problem and that we believe there is value in providing a simple model to start with and to which further additions can then be added.

- Improving the number of comparisons: It is possible to obtain $O(n^{3/2})$ inversions with $O(n \log n)$ comparisons, which is optimal up to lower order terms. This is because the algorithm maintains an ordering of the elements seen so far. Thus, when a new element arrives, the algorithm can find the position of this new element in the current ordering by binary search.
- Noisy pairwise comparisons: we can extend our results to a noisy model where each comparison is, independently, correct w.p. $1-\epsilon$ and wrong w.p. ϵ. The impact of the noise in the analysis is an additional ϵn additive term per element in the estimation cost and no additional term for the assignment cost. The total cost is then $O(n^{3/2} + \epsilon n^2)$.
  The reason this can be done is that the analysis is modular and decomposes the error in an estimation and assignment terms. If we estimate the new rank using line 3 of Algorithm 1 but with noisy comparisons, we still have an unbiased estimator of the rank (but with larger variance) and Proposition 3.3 still holds. Those are the only requirements for the analysis of the assignment cost, so it remains unchanged.
- If we have noisy comparisons and also want to improve the number of comparisons, the technique in (a) for reducing the number of comparisons no longer holds, but we can still reduce the number of comparisons by estimating $\tilde{r}$ in line 3 of Algorithm 1 by sampling only a few of the previous elements and paying the corresponding cost in increased variance. The same analysis as in (b) still holds.