We thank all the reviewers (**R1**, **R2**, and **R4**) for their thoughtful comments, and respond to questions below.

Hiring as a motivating application & multiple arm pulls per stage (R1, R4): We were initially motivated by the graduate admissions system run at our university. Here, at every stage, it is possible for *multiple* independent reviewers to look at an applicant. Indeed, our admissions committee strives to hit at least two written reviews per application package, before potentially considering one or more Skype/Hangouts calls with a potential applicant. (In our data, for instance, some applicants received up to 6 independent reviews per stage.)

While motivated by academic admissions, we believe our model is of broad interest to industry as well. For example, in tech, it is common to allocate more (or fewer) 30-minute one-on-one interviews on a visit day, and/or multiple pre-visit programming screening teleconference calls. Similarly, in management consulting (cf. the McKinsey citation in our submission), it is common to repeatedly give independent "case study" interviews to borderline candidates.

Information gain (R1): Information gain s relates to the confidence of accept/reject from an interview vs review. As stages get more expensive, the estimates of utility become more precise, i.e. the estimate comes with a distribution with a lower variance. In practice, a resume review may make a candidate seem much stronger than they are, or a badly written resume could severely underestimate their abilities. However, in-person interviews give better estimates. A strong arm pull with information gain s is equivalent to s weak pulls because it is equivalent to pulling from a distribution with a $\sigma/\sqrt{s}$ sub-Gaussian tail, i.e. it is equivalent to getting a (probably) closer estimate.

Hardness (R1, R2, R4): Hardness is defined in Eq. 3 as the sum of inverse squared gaps. We will be sure to point to Eq. 3 every time hardness is discussed; the example to be added in response to **R2** will help with intuition as well.

Figure 1 (R4): Fig. 1 is plotted using fixed values for K_1, δ, and ϵ. Hardness is defined in Eq. 3 (see above). We did, however, run experiments to see how the figure shifts when increasing/decreasing each of these values (to quantify the intuition given by Thm. 1). In order to save space we described the effects of the variation on the figure. In the camera ready version we can include a full page of varied graphs in the appendix, the data for which already exists.

Exposition clarification for T and budget (R1): The definition of T changes based on the section of the paper; we will change this. In Thm. 1 and Alg. 1 the reviewer is correct to assume that T is the cost divided by the information gain. In Thm. 2 and Alg. 2, as presently written, T becomes the budget. We will change T to $\tilde{T}$ in Sec. 4 for clarity.

Budget and K_i (R1): The reviewer is correct in suggesting that choosing budget and K_i simultaneously is difficult. The policy maker should look at past decisions to estimate gap scores (Eq. 2) and hardness (Eq. 3). There is a clear trade-off between information gain and cost. If the policy maker assumes (based on past data) that the gap scores will be high (it is easy to differentiate between applicants) then the lower stages should have a high K_i, and a budget to match the relevant cost j_i. If the gap scores are all low (it is hard to differentiate between applicants) then more decisions should be made in the higher, more expensive stages. By looking at the ratio of small gap scores to high gap scores, or by bucketing gap scores, a policy maker will be able to set each K_i.

Oracle and suboptimality (R4): The algorithm assumes that the oracle is correct (which is a very light assumption for the theory section, which assumes a linear objective, but less trivial for general monotone submodular objectives like the diversity-promoting objective used in some of the experiments). However, we find that even with this assumption the algorithm performs well empirically under the submodular function given.

Correctness of Alg. 2 (R1): Thanks for pointing this out! It is a typo. $\tilde{K}_i$ should be $n - \sum_{a=0}^{i-1} \tilde{K}_a$, i.e., the number of arms remaining. In our code we implemented K_i in that correct way. (See grid test on line 482 in `cut_neurips.py`).

Cost (R1): The confidence radius was chosen due to a Hoeffding bound. See Appendix B1 for further information.

BRUTaS and CACO differences (R2): The CACO algorithm is a fixed confidence algorithm, meaning that each stage ends after some (user-supplied) confidence interval is satisfied. When a stage ends, only K_i *arms* move on to the next stage. The BRUTaS algorithm is a fixed budget algorithm where each stage uses a certain amount of *budget*. All arms are pulled $\tilde{T}_{i,t}$ times before a decision to keep or reject a single arm is made. An apples-to-apples example comparing the two algorithms is a bit tricky due to taking different inputs (per-round budget vs. number of arms); we will add two "similar" illustrative examples in the 3-arms/2-stage case to help compare and contrast the two algorithms, though.

Greedy subset selection (R4): We will run against the greedy algorithm, as suggested. We find the most relevant comparison to be the previous work ("SWAP") from Schumann et. al. 2019. Our method has theoretical guarantees, as does theirs. (Note that we also compare to the actual admissions decision process, which is fairly complex.)

Anonymized data release (R4): Unfortunately, we are prohibited via FERPA regulations from releasing the real dataset. That said, we are working (i) with our IRB for an anonymized release option as suggested (e.g., by fitting a blanket parameterized model) and (ii) to release through our multi-university research consortium a general set of scripts that mimic our application scraping (e.g., running OCR on SoPs and rec. letters, parsing structured data such as GRE/GPA) and feed that generic data into our algorithms run on their servers. This is a longer process that will require rounds of validation, but our long-term goal is to collect aggregate statistics and push practical impact in this way.