AC and all Reviewers: We thank all reviewers. To summarize, all the reviewers acknowledged the novelty of our transductive Mutual Information (MI) loss, the novelty of the ADM optimizer and the speed-up it brings, as well as the SOTA results over 5 benchmarks, by significant margins. The criticisms are essentially based on: discussions on recent prior works and motivation (R1), a misunderstanding of the standard transductive setting in few-shot learning (R3), and the use of a label marginal prior (R2, R4), which is just a convenient generalization of our MI, but not really the main contribution of the paper (our prior-free MI loss obtains SOTA results over all benchmarks).

(R2-R4) Concerns about the use of a prior π over label marginals Y: First, we concede that writing Eq. (3) as we did may have conveyed that the prior itself is crucial to the well functioning of the method, and needs to be estimated accurately from the support set. We discuss the motivation of using $\mathcal{S}$-based prior later. Regardless, this does not alter the main contribution of the work, which is the MI (uniform prior). As a matter of fact, R4 also brought up this point, yet gave an accept score (7). The main utility of the term $\mathcal{D}_{\text{KL}}$ is actually to prevent the trivial solutions of conditional-entropy minimization, as discussed in L.229-232, rather than to impose exact label proportions. Therefore, the exactitude of the prior is not crucial. In fact, using any prior at all (i.e., non-uniform) is optional in our formulation, as it merely represents a generalization of the standard MI. We emphasize that the MI (no prior) alone is our contribution, and achieves SOTA results over 4 standard benchmarks by wide margins (it is also competitive on iNat). Imposing a prior when available (could come from any source) is application-dependent and can indeed lead to enhanced performances, but is, again, not necessary. All in all, the issue could be easily fixed by writing CE + MI as our main objective (3), and merely proposing the prior-aware MI version as an extension when a prior is available. **Why using a support-based prior?** The very setting of FSL assumes that $\mathcal{S}$ and $\mathcal{Q}$ are sampled from the same underlying joint distribution $p(x, y)$ (Section 1.2 in [Wang et al. "Generalizing from a few examples: A survey on few-shot learning." ACM Computing Surveys]), which uniquely determines the marginal $p(y)$ (as can be seen by simply marginalizing out x). Hence, empirical marginals should be close $\hat{p_{\mathcal{S}}}(y) \approx \hat{p_{\mathcal{Q}}}(y)$. Therefore, we concede to R2-R4 that exactly having $\hat{p_{\mathcal{S}}}(y) = \hat{p_{\mathcal{Q}}}(y)$ as in standard benchmarks may appear somewhat artificial. Typically, the $\mathcal{S}$-based prior does not exactly match the true query marginal as in the more *realistic* iNat task, but still provides a better estimate of it (as shown by our results on iNat).

(R1) Clarification of the motivation : We are maximizing the MI between the query features and predictions (i.e., between inputs and outputs), not between the query and support sets. We mentioned DeepInfoMax as an example of a deep-learning instance of the original InfoMax principle. We use the latter principle in a different way: The inputs are considered as extracted features of pretrained network, and output are predictions. Such idea is motivated by and relates to the MI in classical clustering works (e.g., [16]), which we cite/discuss in L.62-64. **Significant differences with references [1, 2, 3]**: We start by emphasizing that [1], [2] and [3] design meta-learning methods for both training and inference. Our method is used for inference only, and can work on top of any pre-trained feature extractor. As for the objective functions, Ref. [1] doesn't deal with the MI. The self-labeling in [1] can be viewed as a minimization of the min-entropy of query predictions (we will cite/discuss this work). Also, the MI measures in [2, 3] evaluate quantities that are completely different from us because [2,3] learn conditional distributions over classifier's weights, while we learn task-specific weights by direct optimization. Specifically, [2] maximizes the MI between features and weights while we maximize MI between the query features and labels. The former is intractable as both variables are continuous with no access to the underlying distributions, and requires a variational approximation, while the latter (ours) is tractable ($Y_{\mathcal{Q}}$ is low-dimensional/discrete + we have access to $p(y|f_\phi(x))$). Objective (7) in [3] uses a cross-entropy (CE) on query samples while our objective uses CE on support and an unsupervised MI on query samples. As for the KL term in (7), it encourages the posterior over weights to match a prior p_ψ, while the KL in our objective (3) encourages the label marginals to be close to a prior. Notice that all 3 papers require additional modules to train (soft-weighting network in [1], reconstruction in [2] and gradient synthetizer in [3]), while our method doesn't (fewer parameters/hyper-parameters). **W-updates:** W are updated with all samples (both support and query) at once (possible because we work directly on low-dimensional extracted features). **Is the method limited to uniform distribution?** No, we can use any prior in Eq. (2). While π is uniform for all the standard benchmarks, for iNat, we showed that a π estimated from the labeled support samples yields improvements (Table 4). **1-shot case, is the one-hot encoded label used for π in Eq. (3)**: No. In all our 1-shot experiments, π is uniform (i.e., no prior – we minimize the MI).

(R3) Mis-understanding of the transductive few-shot setting: The model **does not see test data** during training (only base classes are seen during training). At test-time, it only sees **one test task at a time** . As discussed in lines 37-56, this transductive setting is now a standard in few-shot learning, as evidenced by the large number of recent major-conference publications in this setting, e.g., [6, 13, 14, 18, 23, 30], among many others.

(R2) (Implementation details:) As mentioned in lines 196-200, we used the standard training in [42] for the base classes. We also used standard image data sets. This is why we did not provide those training and image-size details.

(R4) ADM vs. GD: All results are averaged over 10'000 runs. For the CE row in Table 3, results differ because TIM-GD performs GD based finetuning, while TIM-ADM uses the updates in Table 9 (appendix). **Application to domain adaptation:** Yes, applying our method to domain adaption might be very interesting. Thanks for the suggestion!