General: we thank the reviewers for their valuable feedback. We first address questions shared by most reviewers:
• **Alg. 1.** We clarify what happens during meta-train and -test respectively for Alg. 1. *Meta-train:* Alg. 1 inverses the regularized kernel matrix $(\mathbf{K} + \lambda I)^{-1}$, costing $O(N^3)$ for N meta-train tasks. Unconditional meta-learning is recommended (not mandatory, see App. C.1) as a warm start for TASML. *Meta-test:* The kernel vector $v(D)$ and weights $\alpha(D)$ are computed, costing $O(N^2)$ operations. $\alpha(D)$ are used in eq. (7) (or 9) to perform adaptation.
• **Inference Time.** For model adaptation, TASML takes $\sim 0.23s$ (computing $\alpha(D)$ for $N = 30k$) and $\sim 6s$ (see line 290) optimizing (9). In applications where model accuracy has the priority (e.g. AutoML services), it can be reasonable to trade-off time for accuracy. The adaptation cost is also amortized over all future queries in the adapted model.
• **Experiments on CIFAR-FS.** We chose the same settings as those used to obtain Tab. 2 in our paper. For 1-shot, TASML ($\mathbf{74.6 \pm 0.7}$), Leo (71.2 ± 0.6), and MAML (68.8 ± 0.7). For 5-shot, TASML ($\mathbf{85.1 \pm 0.4}$), Leo (82.0 ± 0.4), and MAML (83.7 ± 0.7). TASML significantly outperforms the baselines, in line with findings in the paper.

R2. *Multi-task Learning (MTL)* While MTL may be used for meta-learning as a heuristic, it does not prioritize performance of target tasks, nor prevent negative transfer towards it. In contrast, TASML only selects the most relevant tasks for adaptation in a principled way. We implemented Kendall et. al on miniImagnet. The results are 56.8 ± 1.4 (1-shot) and 68.7 ± 1.2 (5-shot), under-performing TASML. Critically, each target task's performance swing widely when trained with the MTL loss, which makes learning unstable. Our additional experiments suggest negative transfer as a main issue with applying MTL to meta-learning.
• *Eq. (9) and MAML.* Both Eq. (7) and (9) are task-specific objectives with near-identical implementation. (9) is a variant of (7), where (9) also exploits (few) labeled samples from target task during training. Remark 1 applies to both.
• *Different architectures in baselines.* Tab. 1 cited results from previous papers. For fairness, Tab. 2 reports results for MAML with WRN-28-10 (i.e. LEO's feature), and that structured prediction can improve both MAML and LEO.

R4. Clarify (9)**.** The additional term can be interpreted as a special task where support and query sets coincide. The term regularizes models to focus on relevant features from selected tasks, in order to perform well on target tasks.
• *Mini-batches and kernel evaluation.* For each target task D, we first compute $\alpha(D)$ against the entire meta-train set (see **Alg. 1** and **Inference Time** above). Each mini-batch samples k tasks and their weights from M-filtered meta-train set to optimize eq. (9) restricted to the mini-batch.
• *Whether meta-representation is learned* Yes. the parameters of the meta-representation are learned.
• *Tab. 3.* LEO is slower during meta-train due to network complexity and having to learn task-conditional initialization. TASML's network is simpler and more efficient to train, but diverts task-conditioning to test time (see **Inference Time**).

R5. *Is structured prediction (SP) necessary?* SP is not the only way to formalize the problem, and our paper reviewed several existing conditional meta-learning methods. Rather, SP offers a principled strategy for conditional meta-learning, for which we can study the statistical properties. These qualities make such perspective appealing.
• *"Handcrafted architecture" and motivation gap.* We will improve the phrasing: for most previous conditional methods, the network design is ad-hoc to implement the specific conditional principles (e.g. task clustering). In contrast, TASML uses kernel to implicitly captures task similarity, and yields weighted loss functions, which are more likely to generalize to different application settings, and augments existing methods (see Tab. 2).
• *On the inequality* $\mathcal{E}(\tau_*) \leq \mathcal{E}(\theta_*)$. Conditional meta-learning minimizes $\mathcal{E}(\cdot)$ over $\mathcal{T}$ (all measurable functions $\tau : \mathcal{D} \to \Theta$), a significantly larger set than Θ (all *constant* functions from $\mathcal{D}$ to Θ). Hence $\min_{\mathcal{T}} \mathcal{E}(\tau) \leq \min_{\Theta} \mathcal{E}(\theta)$.
• *Does warm-start affect rates?* No, Thm.1 does not make assumptions about the initial model parameters.
• *Choice of Kernel.* We compared and discussed the impact of different kernels options in App. C.4.
• *Using least-squares (LS) loss* For few-shot classification task, there appears to be no drawback in our experiments. On the contrary, LS enables efficient meta-gradient computation and speed up learning.
• *What makes TASML "tick"?* Top-3 factors: $1)$ feature pre-training; $2)$ structured prediction, and $3)$ least-squares loss.

R6. *On the term "structure".* We agree with R6 that the term "structure" can be confused with existing methods. We will differentiate such literature and TASML, and clarify that the term denotes the use of structured prediction.
• *Access training tasks at test time.* We agree that non-parametric methods could be challenging for low-resources settings (e.g. mobile devices). However, in settings such as AutoML services in data centers, access to some past data is common, and trading-off model adaptation time to achieve better performance is a valid use case (e.g. AutoGluon by Erikson et. al 2020). Further, top-M filtering (see line 180) already limits memory usage, and we plan to further mitigate the requirement in the future, e.g. by storing only salient tasks (e.g. Sparse GP by Seeger et al. 2003).
• *Clarify conditional meta-learning* Conditional meta-learning refers to methods that condition initial model parameters on target tasks, followed by gradient-based model adaptation. Methods such as MAML is unconditional as it learns a shared initial parameters for all tasks.
• *Relation with works [1,2,3,4,5] from R6.* We note that [2,3] were discussed in the paper (lines 86-97, 188). [1] assumes knowledge about task hierarchy and not directly comparable to TASML. We will include [4, 5] as conditional methods. However, they do not belong to our paper's focus of gradient-based meta-learning methods. We note that TASML achieves better results against [2, 4]'s reported results (the rest didn't perform the same benchmark).