A. Proof of Proposition 1

Reminder of Proposition 1 OSLO's optimization problem, being defined in (6) as:

$$\begin{aligned} \max_{\boldsymbol{\mu},\boldsymbol{Z},\boldsymbol{\xi}} \quad & \mathcal{L}_{\text{O}}(\boldsymbol{Z},\boldsymbol{\xi},\boldsymbol{\mu}) + \mathcal{L}_{\text{soft}}(\boldsymbol{Z},\boldsymbol{\xi}) \\ \text{s.t} \quad & \boldsymbol{z}_i \in \Delta^K, \quad \xi_i \in [0,1] \quad \forall\, i \\ & \boldsymbol{z}_i = \boldsymbol{y}_i, \quad \xi_i = 1, \quad i \leq |\mathbb{S}| \end{aligned}$$

can be minimized by alternating the following updates:

$$\xi_i^{(t+1)} = \begin{cases} 1 \quad \text{if } i \leq |\mathbb{S}| \\ \sigma\left(\frac{1}{\lambda_\xi}\sum_{k=1}^{K} z_{ik}^{(t)} \log p(\boldsymbol{x}_i|k;\boldsymbol{\mu}^{(t)})\right) \text{ else} \end{cases}$$

$$\boldsymbol{z}_i^{(t+1)} \propto \begin{cases} \boldsymbol{y}_i \quad \text{if } i \leq |\mathbb{S}| \\ \exp\left(\frac{\xi_i^{(t+1)}}{\lambda_z} \log p(\boldsymbol{x}_i|\ \cdot\ ;\boldsymbol{\mu}^{(t)})\right) \text{ else} \end{cases}$$

$$\boldsymbol{\mu}_k^{(t+1)} = \frac{1}{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i^{(t+1)} z_{ik}^{(t+1)}} \sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i^{(t+1)} z_{ik}^{(t+1)} \phi_\theta(\boldsymbol{x}_i)$$

where σ denotes the sigmoid operation.

Proof. We denote by $\nabla_\cdot(\mathcal{L}_{\text{O}} + \mathcal{L}_{\text{soft}})$ the partial derivative of OSLO's optimization problem. We calculate the updates of ξ_i and z_{ik} for $i > |S|$, and of $\boldsymbol{\mu}_k$, by finding the annulation point of their partial derivative.

$$\begin{aligned} & \nabla_{\xi_i}(\mathcal{L}_{\text{O}} + \mathcal{L}_{\text{soft}}) = 0 \\ \Leftrightarrow \quad & \sum_{k=1}^{K} z_{ik} \log\left(p(\boldsymbol{x}_i,k;\boldsymbol{\mu})\right) \\ & \quad = \lambda_\xi\left((\log\xi_i + 1) - (\log(1-\xi_i)+1)\right) \\ \Leftrightarrow \quad & \frac{1}{\lambda_\xi}\sum_{k=1}^{K} z_{ik} \log\left(p(\boldsymbol{x}_i,k;\boldsymbol{\mu})\right) = \log\left(\frac{\xi_i}{1-\xi_i}\right) \\ \Leftrightarrow \quad & \xi_i = \sigma\left(\frac{1}{\lambda_\xi}\sum_{k=1}^{K} z_{ik} \log\left(p(\boldsymbol{x}_i,k;\boldsymbol{\mu})\right)\right) \end{aligned}$$

$$\begin{aligned} & \nabla_{z_{ik}}(\mathcal{L}_{\text{O}} + \mathcal{L}_{\text{soft}}) = 0 \\ \Leftrightarrow \quad & \xi_i \log\left(p(\boldsymbol{x}_i,k;\boldsymbol{\mu})\right) = \lambda_z\left(\log z_{ik} + 1\right) \\ \Rightarrow \quad & z_{ik} \propto \exp\left(\frac{\xi_i}{\lambda_z}\log\left(p(\boldsymbol{x}_i,k;\boldsymbol{\mu})\right)\right) \end{aligned}$$

$$\begin{aligned} & \nabla_{\boldsymbol{\mu}_k}(\mathcal{L}_{\text{O}} + \mathcal{L}_{\text{soft}}) = 0 \\ \Leftrightarrow \quad & \sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}\left(\phi_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - \boldsymbol{\mu}_k\right) = 0 \\ \Leftrightarrow \quad & \boldsymbol{\mu}_k = \frac{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik} \phi_{\boldsymbol{\theta}}(\boldsymbol{x}_i)}{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}} \end{aligned}$$

□

B. Normalizing centroids

Because we work with normalized features, we state in our implementation details that we found normalizing $||\boldsymbol{\mu}||$ after each update helps. Here we show that this "projected step" is actually the exact solution to the optimization problem Eq. (6) when adding the constraint $\boldsymbol{\mu} \in \mathcal{B}_2$, where $\mathcal{B}_2 = \{\boldsymbol{x} : ||\boldsymbol{x}||_2 = 1\}$ is the unit hypersphere.

Specifically, adding the constraint $\boldsymbol{\mu} \in \mathcal{B}_2$ modifies the Lagrangian by infinitely penalizing $\boldsymbol{\mu}_k$ for being outside the unit hypersphere. Without loss of generality, we only consider the part of the Lagrangian pertaining to $\boldsymbol{\mu}_k$ for some $k \in [1,K]$, which we refer to as $\mathcal{L}_k$:

$$\mathcal{L}_k(\boldsymbol{\mu}_k) = \sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik} ||\boldsymbol{\mu}_k - \phi_\theta(\boldsymbol{x}_i)||^2 + \mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k)$$

where $\mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k)$ equals 0 if $\boldsymbol{\mu}_k \in \mathcal{B}_2$ and ∞ otherwise. Because $\mathcal{L}_k$ is no longer differentiable, we introduce the subdifferential operator $\partial_\cdot(\cdot)$, which generalizes the standard notion of differentiability. Akin to the standard case, we look for $\boldsymbol{\mu}_k$ such that:

$$0 \in \partial_{\boldsymbol{\mu}_k} \mathcal{L}_k(\boldsymbol{\mu}_k),$$

which amounts to:

$$\begin{aligned} \Leftrightarrow \quad & 0 \in \{\nabla_{\boldsymbol{\mu}_k} \sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik} ||\boldsymbol{\mu}_k - \phi_\theta(\boldsymbol{x}_i)||^2\} + \partial_{\boldsymbol{\mu}_k} \mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k) \\ \Leftrightarrow \quad & \sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}\ \phi_\theta(\boldsymbol{x}_i) - \boldsymbol{\mu}_k(\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}) \in \partial_{\boldsymbol{\mu}_k} \mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k) \\ \Leftrightarrow \quad & \frac{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}\ \phi_\theta(\boldsymbol{x}_i)}{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}} - \boldsymbol{\mu}_k \in \partial_{\boldsymbol{\mu}_k} \frac{1}{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}} \mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k) \\ \Leftrightarrow \quad & \frac{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}\ \phi_\theta(\boldsymbol{x}_i)}{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}} - \boldsymbol{\mu}_k \in \partial_{\boldsymbol{\mu}_k} \mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k) \\ \Leftrightarrow \quad & \boldsymbol{\mu}_k = \text{Proj}_{\mathcal{B}_2}\left(\frac{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}\ \phi_\theta(\boldsymbol{x}_i)}{\sum_{i=1}^{|\mathbb{S}|+|\mathbb{Q}|} \xi_i z_{ik}}\right) \end{aligned}$$

where the penultimate step holds because $\lambda\mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k) = \mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k)$ by definition of $\mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k)$, and the last step holds because the projection operator $\text{Proj}_{\mathcal{B}_2}(\boldsymbol{\mu}_k) = \frac{\boldsymbol{\mu}_k}{||\boldsymbol{\mu}_k||}$ is the proximity operator of the constraint function $\mathcal{L}_{\mathcal{B}_2}(\boldsymbol{\mu}_k)$.

C. Metrics

Here we provide some details about the metrics used in Section 5

Acc: the classification accuracy on the closed-set instances of the query set (*i.e.* $y^q \in \mathbb{C}_{\mathbb{S}}$).

AUROC: the area under the ROC curve is an almost mandatory metric for any OOD detection task. For a set of outlier predictions in $[O, 1]$ and their ground truth (0 for inliers, 1 for outliers), any threshold $\gamma \in [O, 1]$ gives a true positive rate *TP*(γ) (*i.e.* recall) and a false positive rate *FP*(γ). By rolling this threshold, we obtain a plot of *TP* as a function of *FP i.e.* the ROC curve. The area under this curve is a measure of the discrimination ability of the outlier detector. Random predictions lead to an AUROC of 50%.

AUPR: the area under the precision-recall (PR) curve is also a common metric in OOD detection. With the same principle as the ROC curve, the PR curve plots the precision as a function of the recall. Random predictions lead to an AUPR equal to the proportion of outliers in the query set *i.e.* 50% in our set-up.

Prec@0.9: the precision at 90% recall is the achievable precision on the few-shot open-set recognition task when setting the threshold allowing a recall of 90% for the same task. While AUROC and AUPR are global metrics, *Prec@0.9* measures the ability of the detector to solve a specific problem, which is the detection of almost all outliers (*e.g.* for raising an alert when open-set instances appear so a human operator can create appropriate new classes). Since all detectors are able to achieve high recall with a sufficiently permissive threshold γ, an excellent way to compare them is to measure the precision of the predictor at a given level of recall (*i.e.* the proportion of false alarms that the human operator will have to handle). Random predictions lead to a *Prec@0.9* equal to the proportion of outliers in the query set *i.e.* 50% in our set-up.

D. Effects of the inlier latent on closed-set model parameters

We reported in Tab. 2 an ablation on the effect of introducing $\boldsymbol{\xi}$ (Eq. (4)) on the obtained $\boldsymbol{Z}$ (latent class assignments). Here we go further into this ablation by illustrating in Figure 5 how leveraging $\boldsymbol{\xi}$ yields better estimates of both $\boldsymbol{Z}$ the prototypes $\boldsymbol{\mu}$. The latter is measured by the similarity between $\boldsymbol{\mu}$ obtained after optimization and the ground-truth prototypes (using the support and query labels of each task). These results indicate that leveraging the inlier latent consistently improves the parametric model $\boldsymbol{\mu}$ across all benchmarks. Interestingly, this does not result in better latent class-assignments $\boldsymbol{Z}$ in the cross-domain scenarios.

E. Broad Open-Set setting

As we state in Section 5, in the standard FSOSR setting [16, 21]:

- support sets contain $|\mathbb{C}_{\text{CS}}| = 5$ closed-set classes with 1 or 5 instances, or *shots*, per class;
- query sets are formed by sampling 15 instances per class, from a total of ten classes:
  - the five closed-set classes $\mathbb{C}_{\text{CS}}$;
  - an additional set of $|\mathbb{C}_{\text{OS}}| = 5$ open-set classes.

This is a very strong assumption on the distribution of open-set samples. While this will not affect inductive method, it is likely to impact the performance of transductive methods like OSLO. In this section, we provide additional results in a more realistic setting. In this new setting, the query set is formed by sampling 15 instances for each of the 5 closed-set classes, plus $5 \times 15 = 75$ open-set instances, which are sampled indifferently from all remaining classes in the test set.

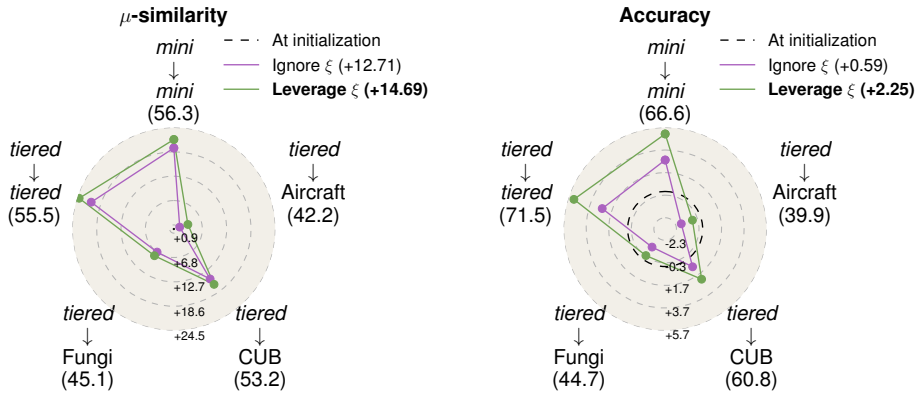
Results in Figure 6 show that the distribution of open-set queries is indeed a major factor in both closed-set and open-set performances for most transductive methods. Interestingly enough, some methods like Laplacian Shot [52] or BDCSPN [23] benefit from this relaxation of the previous open-set assumption. However, while OSLO's closed-set accuracy increases in the new setting, its open-set recognition ability decreases (while still achieving the best results across the benchmark).

F. The difficulty of FSOSR

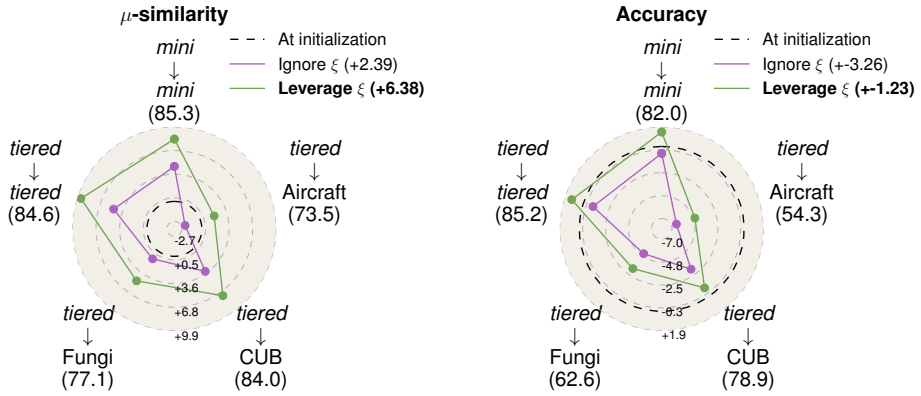
As stated in Section 3, our method follows the model-agnostic setting. Therefore, we perform Few-Shot Open-Set Recognition on features lying in a feature space $\mathcal{Z}$ and extracted by a frozen model $\phi_{\boldsymbol{\theta}} : \mathcal{X} \rightarrow \mathcal{Z}$, whose parameters $\boldsymbol{\theta}$ were trained on some large dataset $\mathcal{D}_{base} = \{(\boldsymbol{x}_i^b, y_i^b)\}_{i=1\ldots|\mathcal{D}_{base}|}$ such that for all i, $y_i^b \in \mathbb{C}_{base}$ with $\mathbb{C}_{base} \cap \mathbb{C}_{\text{CS}} = \mathbb{C}_{base} \cap \mathbb{C}_{\text{OS}} = \emptyset$.

While model-agnosticity is a very strong selling point for a few-shot learning method, it also comes with very difficult challenges, especially for an outlier detection task. In this section, we aim at providing a better understanding of the difficulty of FSOSR with both a qualitative and quantitative study of the clusters formed by novel classes $\mathbb{C}_{\text{OS}}$ when embedded by a feature extractor $\phi_{\boldsymbol{\theta}}$ untrained on $\mathbb{C}_{\text{OS}}$.

Measuring the difficulty of outlier detection on novel classes. As an anomaly detection problem, open-set recognition consists in detecting samples that differ from the population that is known by the classification model. However,



(a) 1-shot



(b) 1-shot

Figure 5. Effects of leveraging the inlier latent $\boldsymbol{\xi}$ on the quality of the closed-set parameters $\boldsymbol{Z}$ (measured with the accuracy) and $\boldsymbol{\mu}$ (measured with the cosine similarity between $\boldsymbol{\mu}$ and the ground truth prototypes computed as the average of all support and query embeddings for each class). We compare the full OSLO method from Eq. (4) (Leverage $\boldsymbol{\xi}$) with the standard likelihood objective from Eq. (3) (Ignore $\boldsymbol{\xi}$) and no optimization (At initialization). This figure follow the same logic as Figure. 2.

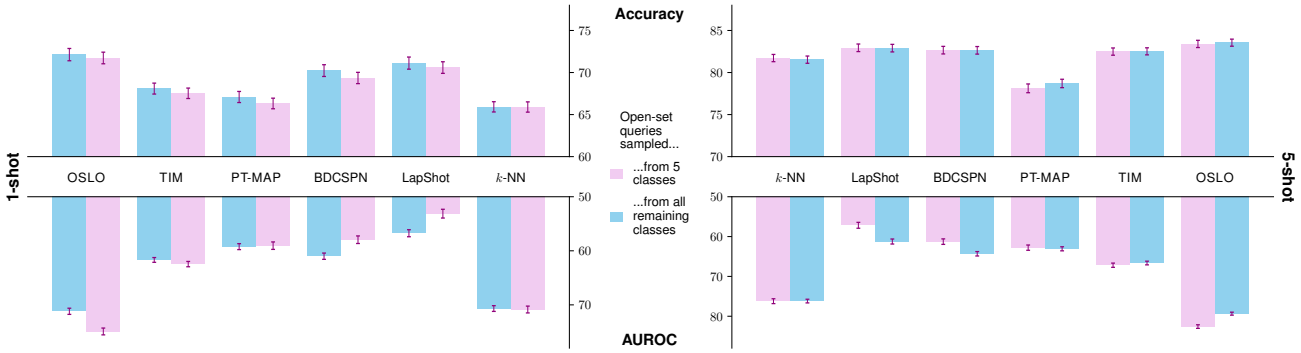


Figure 6. **Performance of transductive methods in the broad open-set setting.** We study the closed-set (accuracy) and open-set (AUROC) performance of transductive methods depending on the size of the query set on *mini*-ImageNet in the 1-shot and 5-shot settings. We add the inductive method k-NN + SimpleShot to compare with a method that is by nature independent to the number of queries.

in FSOSR, neither closed-set classes nor open-set classes have been seen during the training of the feature extractor *i.e.* $\mathbb{C}_{base} \cap \mathbb{C}_{\text{CS}} = \mathbb{C}_{base} \cap \mathbb{C}_{\text{OS}} = \emptyset$. In that sense, both the inliers and the outliers of our problem can be considered outliers from the perspective of the feature extractor. Intuitively, this makes it harder to detect open-set instances, since the model doesn't know well the distribution from which they are supposed to diverge. Here we empirically

Figure 7. 2-dimensional reduction with T-SNE of feature extracted from ImageNet's validation set using a ResNet12 trained on *mini*ImageNet. (Left): images from 20 randomly selected classes represented in *mini*ImageNet's base set. (Right): Images from the 20 classes represented in *mini*ImageNet's test set. Each color corresponds to a distinct class.

demonstrate and quantify the difficulty of OSR in a setting where closed-set classes have not been represented in the training set. Specifically, we estimate the gap in terms of quality of the classes' definition in the feature space, between classes that were represented during the training of the feature extractor *i.e.* $\mathbb{C}_{base}$, and the classes of the test set, which were not represented in the training set. To do so, we introduce the novel Mean Imposture Factor measure and use the intra-class to inter-class variance ratio ρ as a complementary measure. Note that the following study is performed on whole datasets, *not* few-shot tasks.

Mean Imposture Factor (MIF). Let $\mathcal{D}_{\phi_\theta} \subset \mathcal{Z} \times \mathbb{C}$ be a labeled dataset of extracted feature vectors, with $\phi_{\boldsymbol{\theta}}$ a fixed feature extractor and $\mathbb{C}$ a finite set of classes. For any feature vector $\boldsymbol{z}$ and a class k to which $\boldsymbol{z}$ does not belong, we define the Imposture Factor $IF_{\boldsymbol{z}|k}$ as the proportion of the instances of class k in $\mathcal{D}_{\phi_\theta}$ that are further than $\boldsymbol{z}$ from their class centroid. Then the MIF is the average IF over all instances in $\mathcal{D}_{\phi_\theta}$.

$$\boxed{MIF = \frac{1}{|\mathbb{C}|} \sum_k \frac{1}{|\mathcal{D}_{\phi_\theta} \backslash \mathcal{D}_k|} \sum_{\boldsymbol{z} \notin \mathcal{D}_k} IF_{\boldsymbol{z}|k}} \qquad (7)$$

$$\text{with } IF_{\boldsymbol{z}|k} = \frac{1}{|\mathcal{D}_k|} \sum_{\boldsymbol{z}' \in \mathcal{D}_k} \mathbb{1}_{\|\boldsymbol{z}' - \boldsymbol{\mu}_k\|_2 > \|\boldsymbol{z} - \boldsymbol{\mu}_k\|_2}$$

with $\mathcal{D}_k$ the set of instances in $\mathcal{D}_{\phi_\theta}$ with label k, and $\mathbb{1}$ the indicator function. The MIF is a measure of how perturbed the clusters corresponding to the ground truth classes are. A MIF of zero means that all instances are closer to their class centroid than any outsider. Note that MIF $= 1 -$ AUROC(ψ) where AUROC(ψ) is the area under the ROC curve for an outlier detector ψ that would assign to each instance an outlier score equal to the distance to the ground truth class centroid. To the best of our knowledge, the MIF is the first tool allowing to measure the class-wise integrity of a projection in the feature space. As a sanity check for MIF, we also report the intra-class to inter-class variance ratio ρ, used in previous works [12], to measure the compactness of a clustering solution.

Base classes are better defined than test classes. We experiment on three widely used Few-Shot Learning benchmarks: *mini*ImageNet [41], *tiered*ImageNet [29], and ImageNet $\rightarrow$ Aircraft [25]. We use the validation set of ImageNet in order to obtain novel instances for ImageNet, *mini*ImageNet, and *tiered*ImageNet's base classes. We also use it for test classes for consistency. In Figure 7, we present a visualization of the ability of a ResNet12 trained on *mini*ImageNet to project images of both base and test classes into clusters. While we are able to obtain well-separated clusters for base classes after the 2-dimensional T-SNE reduction, this is clearly not the case for test classes, which are more scattered and overlapping. Such results are quantitatively corroborated by Table 3, which shows that both MIF and ρ are systematically lower for base classes across 3 benchmarks and 5 feature extractors. This demonstrates the difficulty of defining in the feature space the distribution of a class that was not seen during the training of the feature extractor, and therefore the difficulty of defining clear boundaries between inliers and outliers *i.e.* closed-set images and open-set images, all the more when only a few samples are available.

G. Additional results

In this section we provide more complete versions of plots included in the main paper. Fig. 8 shows the results depending on the size of the query set for *mini*-ImageNet. Furthermore, 9 and 10 complete Fig. 2, showing the additional

Table 3. Contrast between datasets made of images from classes represented (*base*) or not represented (*test*) in the feature extractor's training set, on three benchmarks and with several backbones (RN12: ResNet12, WRN: WideResNet1810, ViT, RN50: ResNet50, and MX: MLP-Mixer), following the MIF (in percents) and the variance ratio (ρ). Best result for each column is shown in bold.

Classes	miniImageNet				tieredImageNet				ImageNet → Aircraft					
	ρ		MIF (%)		ρ		MIF (%)		ρ			MIF (%)		
	RN12	WRN	RN12	WRN	RN12	WRN	RN12	WRN	ViT	RN50	MX	ViT	RN50	MX
base	**0.93**	**0.84**	**0.89**	**1.03**	**1.09**	**0.78**	**0.78**	**0.81**	**0.96**	**1.36**	**2.54**	**0.09**	**0.29**	**0.31**
test	2.10	2.07	5.56	7.36	2.10	1.54	4.39	5.18	3.20	4.88	5.35	18.08	21.58	17.27

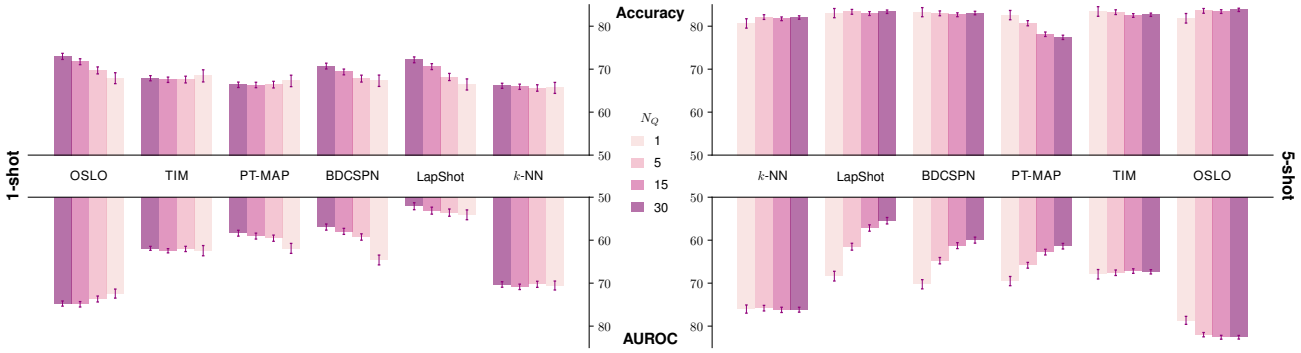


Figure 8. Version of Fig. 3 on *mini*-ImageNet.

Prec@0.9 metric, along with the results on the WRN2810 provided by [48].

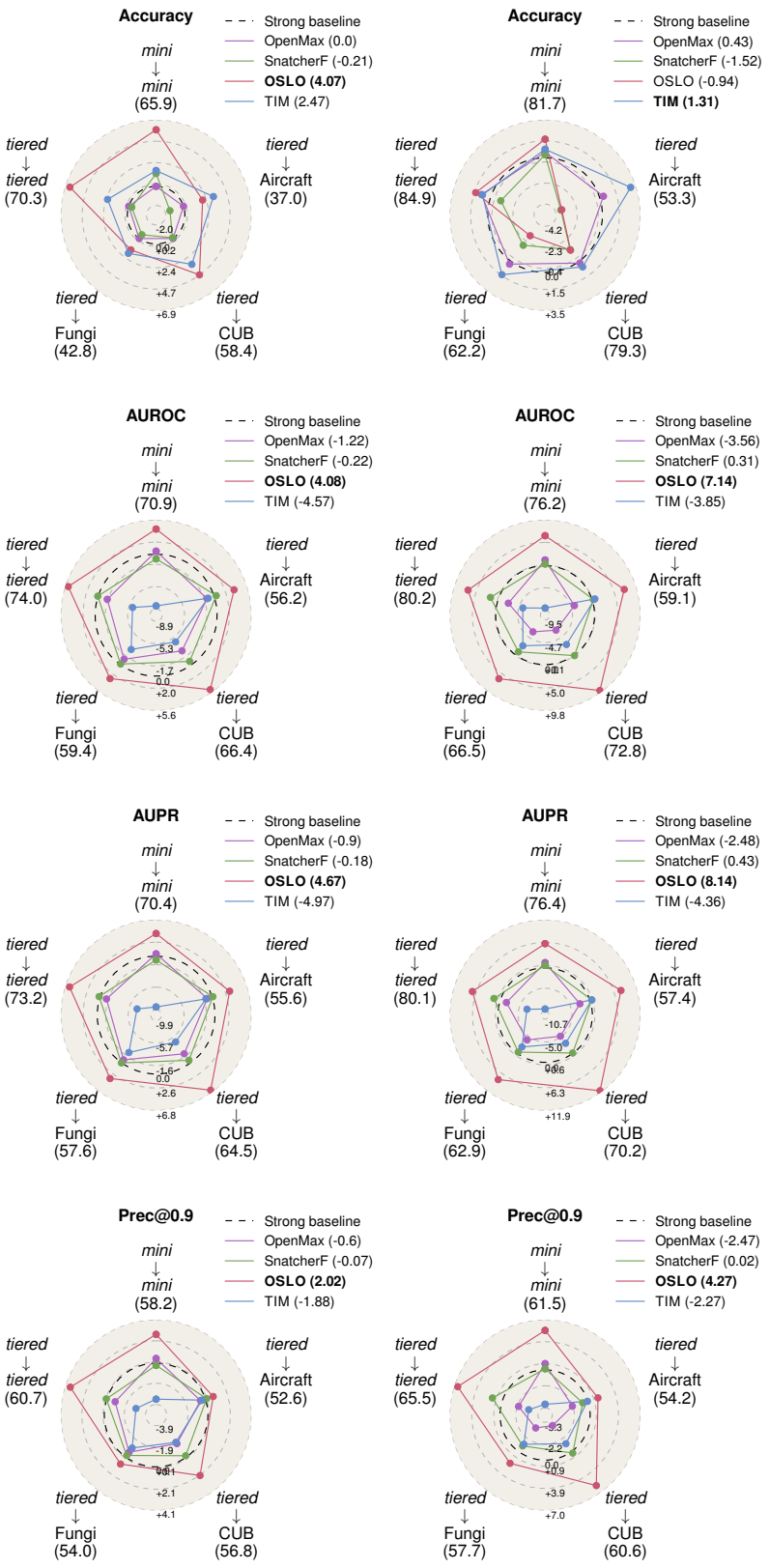


Figure 9. Complete version of Fig. 2 with a ResNet-12. (Left column): 1-shot. (Right column): 5-shot.

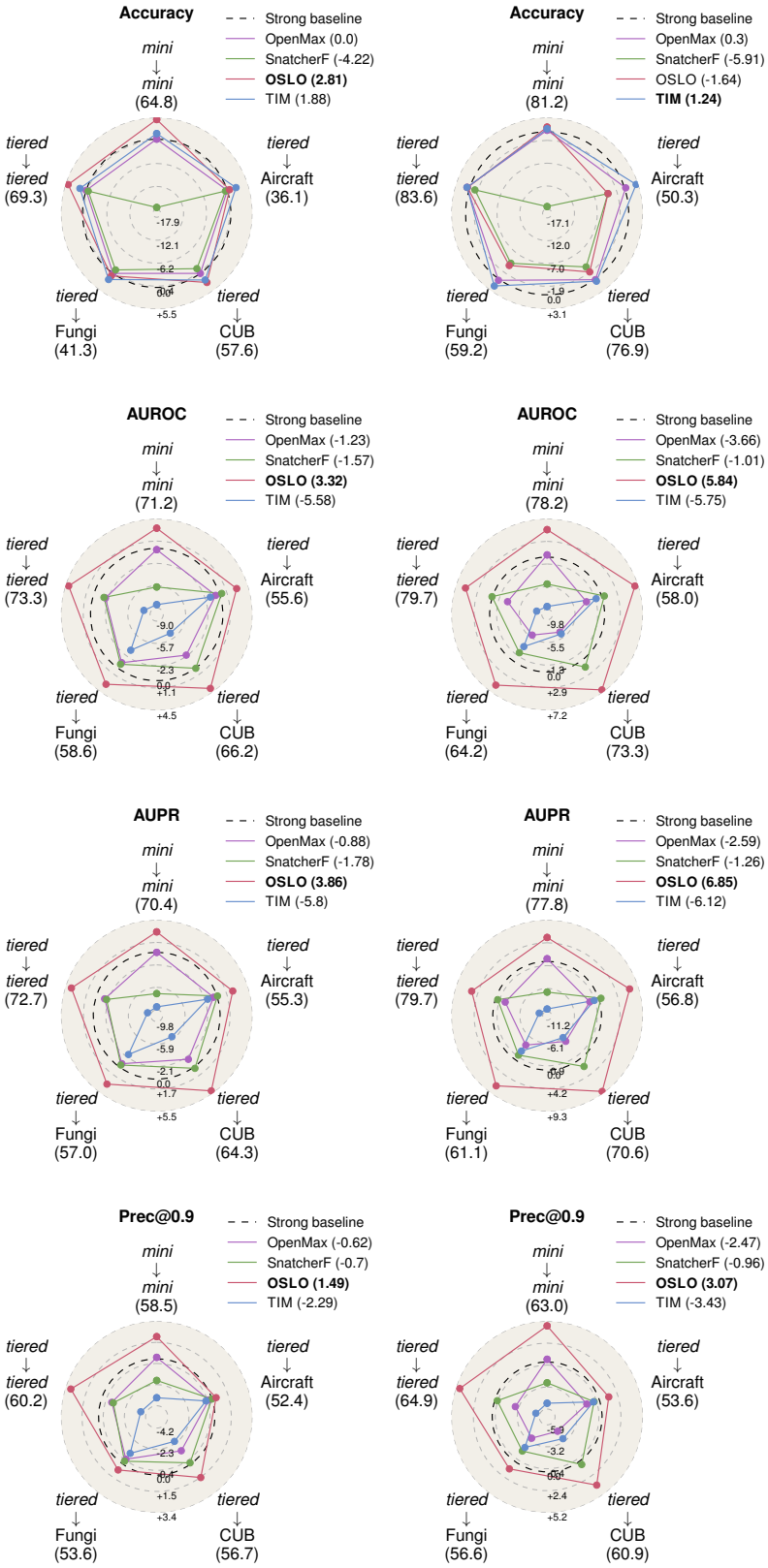


Figure 10. Complete version of Fig. 2 with a WideResNet 28-10. (Left column): 1-shot. (Right column): 5-shot. SnatcherF was not included in this plot because a yet misdiagnosed problem occurred with the provided *tiered*-ImageNet checkpoint.