We thank all reviewers for their insightful and constructive comments.

Response to R#2 Q2.1:...The main weakness is novelty ... the novel contributions are in data scheduler and negative sample strategy, which in my opinion do not constitute a contribution worth publication at NeurIPS... While R#2 well noticed our new techniques, our contribution of *bringing a simple and effective line back to the sight for unsupervised learning* may be overlooked. It breaks the inherent belief that parametric instance classification has intrinsic limitation [32]. Such knowledge is new to the community and is better valued independent of tech contribution. As for the two novel techniques, we respect the reviewer's personal opinion, but we would greatly appreciate if the reviewer could also take the following facts into consideration: 1) "a simple baseline" (by R#1, R#3, R#4) which does not "require special handling to avoid data leakage and can be easily implemented" (by R#4); 2) simple and effective solutions to address *crucial* issues in parametric instance classification (by R#1 ,R#3 ,R#4): the sliding window scheduler addresses the extremely infrequent instance visiting issue and the *corrected updating technique* enables applying the PIC framework to unlimited data scale.

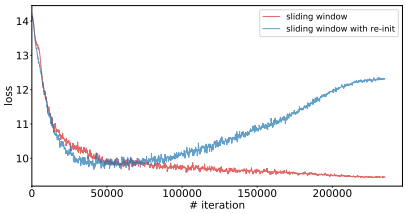
Q2.2: ...Another weakness of this paper is on the empirical side...when SimCLR is trained for 1000 epochs it outperforms the proposed method... SimCLR performs worse than our approach in longer training settings (69.3% vs 70.8% in Table 6). Nevertheless, the main goal of this work is not to achieve higher accuracy number, but more importantly to recall another line of unsupervised learning, which could increase the diversity of research.

Q2.3: ...notation details not properly introduced... We would check it carefully and improve the writing accordingly.

Q2.4: ...missing references on cross-level relationship modeling... The listed papers perform unsupervised learning via pairwise similarity learning. We will add discussion of these papers in the revision.

Response to R#3 Q3.1: ... make it clear in introduction that the proposed solution is a "fix" of [12] ... Thanks for the suggestion. We will clarify it in the introduction like "our paper is basically a revisit of [12]" (L184-188).

Q3.2: ...does the improvement in training comes only from being able to "learn well" a single instance class before moving to another one? ... Yes, the improvement in training should come from being able to "learn well" a single instance class because the opposite effect could not help training.



Q3.3: How about the opposite effect like forgetting this instance class [1*] ... To answer this question, we consider an experiment by re-initializing all the instance class weights which have not been seen for a long time (12.5% in each window). The figure illustrates its loss curve compared to that of the original sliding-window based training. At early steps, explicitly forgetting the instances performs similarly well with the original sliding-window based method, indicating the forgetting issue does not affect optimization. This is probably because the large learning rate enables learning classification weights well even when given random initialization for the long-period-not-visited instances. At later steps, the standard sliding-window has steadily reduced loss while explicitly forgetting the instances results in much poorer performance. This indicates that the weights for long-period-not-visited instances may still effect well, although their weights are not updated for a long period, probably because of the small learning rates. On the whole, the forgetting of the *minority* may not be a serious issue, and the benefit of well learning the *majority* weights overcomes such disadvantages. We will add discussion in the revision.

Q3.4: ...it would be much more valuable if we could see a comparison of visualizations and statistics (Fig. 3) with other methods such as MoCo and SimCLR... Thanks for the suggestion. We conducted similar experiments for MoCo and SimCLR and their behavior is similar as PIC. Actually, the goal of this section is not to compare PIC with MoCo/SimCLR, but to understand PIC itself, where the conclusion may be generalized to other instance discrimination based methods. Note we also perform experiments distinct to PIC, shown in Appendix Sec. F and Fig. 4, which build a smooth transition between PIC and supervised method.

Q3.5: ...how to count epochs for sliding window sampler... As stated in the footnote of page 5, we use the term "epoch" to indicate the equivalent training length with that of epoch-based scheduler, to simplify the description.

Response to R#4 Q4.1: ...results on larger backbone (e.g., deeper or wider)... ResNet-50 ($2\times$) with 200-epoch pre-training achieves 71.2% top-1 accuracy, which is 3.9% better than that of the standard ResNet-50 ($1\times$, 67.3%).

Q4.2: ...if projection head is dropped... Yes, we follow [5,15,22] to drop the projection head in downstream tasks.

Q4.3: ...comment on why cosine softmax brings such a significant improvement in the current setting... The only difference between cosine softmax loss and standard softmax loss is that the standard softmax loss accounts vector *magnitudes* for similarity computation, in addition to the *angles* between two vectors used by cosine softmax. We think the significant accuracy drop by standard softmax loss is due to the significantly worse generalization ability of vector *magnitudes* than *angles*. In fact, similar behavior has been widely observed in numerous applications of metric learning, including face recognition [9, 30], person re-ID ("Deep Cosine Metric Learning for Person Re-Identification , WACV-18") and few-shot learning ("MatchingNet-NeurIPS16", "Improving Generalization via Scalable Neighborhood Component Analysis, ECCV-18", "A Closer Look at Few-shot Classification, ICLR-20"). To our knowledge, strict proof of such behavior is still an open question.