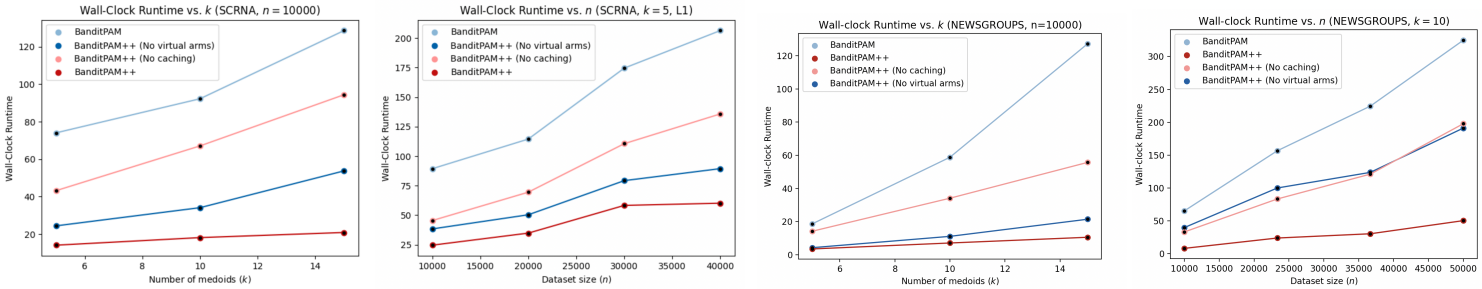

Figure 1 (scRNA & 20 Newsgroups): Wall-clock runtime versus scaling with k and n for the `SCRNA` ($L1$ distance) and `20 Newsgroups` (cosine distance) datasets. We observe speedups of up to $6.4\times$ and $8.5\times$ speedup when examining the scaling of each algorithm with k and $3.2\times$ and $7\times$ speedup for scaling of n for the two datasets, respectively.

Table 1: Average Runtime Speedup Summary

Dataset	BUILD + SWAP	SWAP only	BUILD only
MNIST	$\times 4.75$ | $\times 8.23$ | $\times 10.77$	$\times 6.20$ | $\times 9.33$ | $\times 12.18$	$\times 1.58$ | $\times 1.83$ | $\times 2.16$
CIFAR	$\times 5.05$ | $\times 9.24$ | $\times 12.52$	$\times 6.29$ | $\times 11.8$ | $\times 15.03$	$\times 2.49$ | $\times 3.27$ | $\times 3.65$
SCRNA	$\times 3.95$ | $\times 5.12$ | $\times 6.81$	$\times 4.74$ | $\times 6.01$ | $\times 7.94$	$\times 2.67$ | $\times 2.91$ | $\times 3.44$
20 Newsgroups	$\times 5.19$ | $\times 7.62$ | $\times 9.18$	$\times 6.67$ | $\times 8.98$ | $\times 10.27$	$\times 2.52$ | $\times 2.90$ | $\times 3.16$

Table 1 (Average Speedup Table): Wall-clock speedup of BanditPAM++ compared to BanditPAM on the four datasets MNIST, CIFAR, SCRNA, and 20 Newsgroups. Results were averaged over four different dataset sizes $n =$ 10k, 20k, 30k, 50k for settings BUILD + SWAP, SWAP only, and BUILD only. The BUILD only setting leverages permutation-invariant caching only, whereas the other two settings also leverage Virtual Arms. The three speedup values in each cell correspond to experiments where $k = 5, 10$, and 15 respectively.

Table 2: Relative Loss with Varying δ

Dataset	10^{-2}	10^{-3}	10^{-5}	10^{-10}
MNIST	1.0	1.0	1.0	1.0
20 Newsgroups	1.0	1.0	1.0	1.0

Table 2 (Loss with δ): Loss of BanditPAM++ over BanditPAM wtih δ values ranging from 10^{-2}, 10^{-3}, 10^{-5}, and 10^{-10}. BanditPAM++ has the exact same clustering loss with BanditPAM for various values δ.

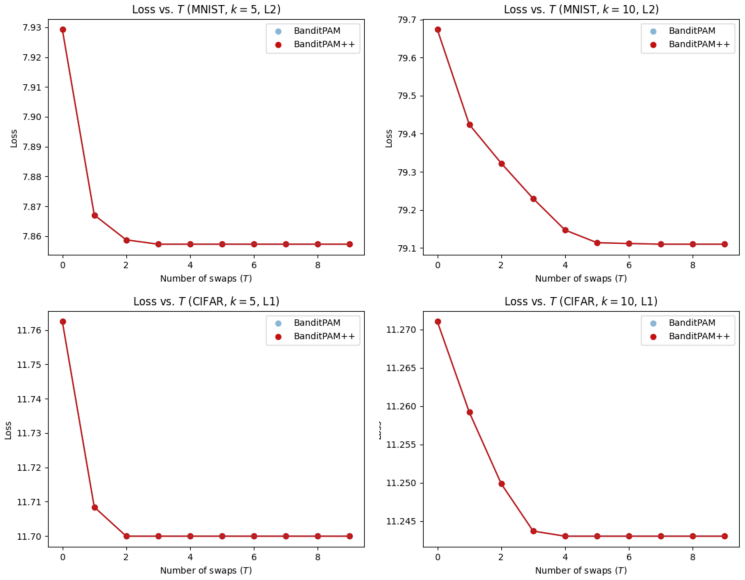

Figure 2 (Clustering Loss with Varying T): Clustering loss with increasing T for the MNIST and CIFAR datasets for $k = 5$ and $k = 10$. Beyond $T = k$, the loss shows very little change. BanditPAM++ and BanditPAM have the same loss for all T making them trace the exact same optimization trajectory with increasing T.