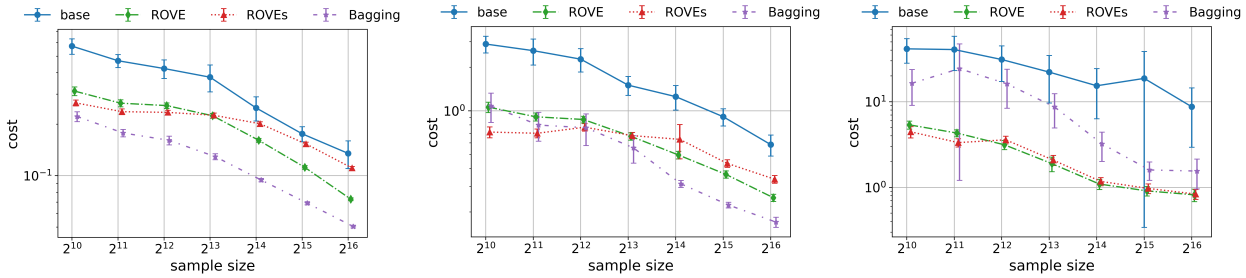
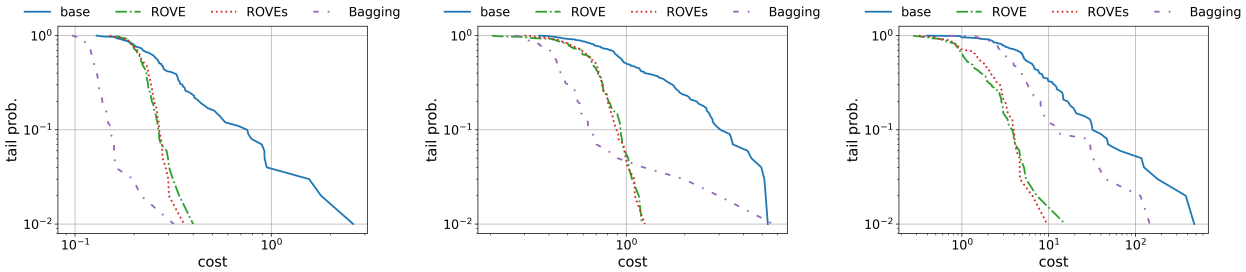


(a) Average perf. for shape $= 2.1$. (b) Average perf. for shape $= 1.5$. (c) Average perf. for shape $= 1.1$.



(d) Tail perf. for shape $= 2.1, n = 2^{13}$. (e) Tail perf. for shape $= 1.5, n = 2^{13}$. (f) Tail perf. for shape $= 1.1, n = 2^{13}$.

Figure 1: Comparison with bagging on synthetic data in terms of expected out-of-sample costs (MSE) with 95% confidence intervals under varying degrees of tail heaviness. Data generation is the same as described in lines 308-313 with Pareto noise of varying shape parameters. (a)(b)(c): Average test performance with 95% CIs. (d)(e)(f): Tail probability of test performance at the sample size $n = 2^{13}$. The base training algorithm is the multilayer perceptron with 4 hidden layers trained with Adam and early stopping. Hyperparameters for ROVE and ROVEs: $k_1 = \max(30, n/2), k_2 = \max(30, n/1000), B_1 = 50, B_2 = 1000$. In each case, the same subsample size k_1 and ensemble size B_1 are used for bagging.

We consider comparison with bagging that resembles our method most closely among existing ensemble methods as both involve repeated training on randomly drawn subsamples. We implement bagging, or subagging to be precise, by training base models on randomly drawn subsamples and then averaging the outputs of all the base models in the ensemble to make the final prediction. The same subsample size and ensemble size are used for both our method and bagging to compare their generalization performance under similar computation costs.

Figure 1 shows that whether bagging or our method outperforms the other depends on the tail heaviness: ROVE and ROVEs exhibit relatively inferior test performance when the noise has a shape of 2.1, but outperforms bagging as the tail of the noise gets heavier towards a shape of 1.1.