Supplementary Information

Performance of SNeCT

Training and testing RMSE

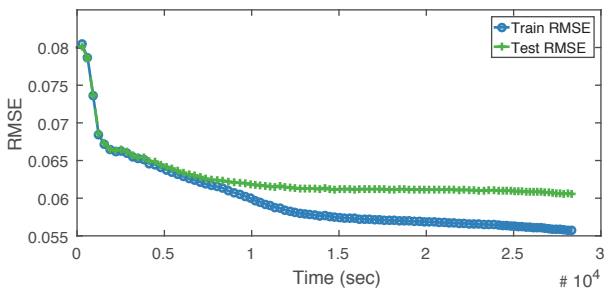

Fig. 1. Training and testing RMSEs of SNeCT with graph constraint of $\lambda = 1$.

Performance comparison

Table 1. Comparison of time complexity (per iteration) and memory usage of SNeCT with existing network-regularized HOSVD algorithm of Narita et al. (2012). SNeCT shows lower time complexity and memory usage. For simplicity, we assume that data tensor $\mathcal{X}$ has order of N, all modes are of size I, of rank J, and one mode has network constraint. P is the number of parallel cores.

	Time complexity (per iter.)	**Memory usage**
SNeCT	$\mathcal{O}(\lvert\Omega_{\mathcal{X}}\rvert J^N N/P + \lvert\Omega_{\mathbf{Y}}\rvert J/P)$	$\mathcal{O}(J^N P)$
Narita *et al.* (2012)	$\mathcal{O}(\lvert\Omega_{\mathcal{X}}\rvert J^N N^2 + \lvert\Omega_{\mathbf{Y}}\rvert J)$	$\mathcal{O}(J^{N-1} I)$

There is an existing work for network-regularized tensor decomposition of Narita *et al.* (2012) which follows gradient descent approach which is hardly parallelizable and takes high memory requirement due to the batch calculation of large matrices for gradients. Table 1 summarizes the comparison between SNeCT and Narita *et al.* (2012). SNeCT outperforms the existing method in terms of both time complexity and memory usage. SNeCT achieves low time complexity by direct parallelization of SGD updates while caching intermediate data tensor and low memory usage by avoiding batch matrix calculation used by Narita *et al.* (2012).

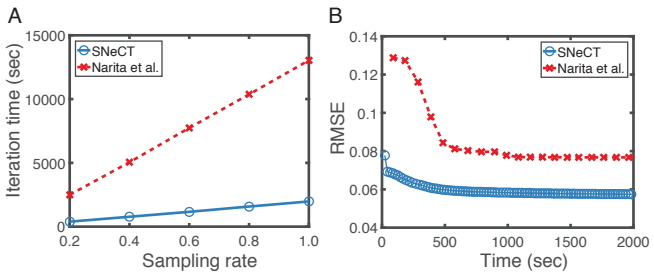

Fig. 2. Running time (A) and convergence (B) of SNeCT and Narita et al. (2012).

We experimentally evaluate SNeCT by measuring its accuracy and speed of decomposition and comparing them with the existing method (Narita *et al.*, 2012). We compare running time (for one iteration) of SNeCT with its competitor, (Narita *et al.*, 2012). We create sampled datasets by randomly selecting part of patients with certain ratio to verify the efficiency of SNeCT for 'big data' scenario. Figure 2(a) shows the running time of two methods for one iteration. Both methods scale linearly as the sampling rate (number of patients) increases. SNeCT takes less time than its competitor. That is, for original dataset, running time of SNeCT is 1974s, 6.6× faster than that of Narita *et al.* (2012), 13036s. Not only the running time but also the accuracy of decomposition is a critical part of the performance of our method. A factored result can intuitively be evaluated by how well the factored components can reconstruct back the validation tensor. We use the Root Mean Squared Error (RMSE) as the measure for reconstruction error. We use a randomly sampled tensor, 10% of patients and 10% of genes to evaluate convergence property of two methods. Figure 2(b) shows the convergence property of two methods. A lower RMSE for the same time means the faster convergence. SNeCT converges to a local minimum within a few decades of iterations while Narita *et al.* (2012) fails to converge within the observed time range.

Supplementary results

Gap statistics result

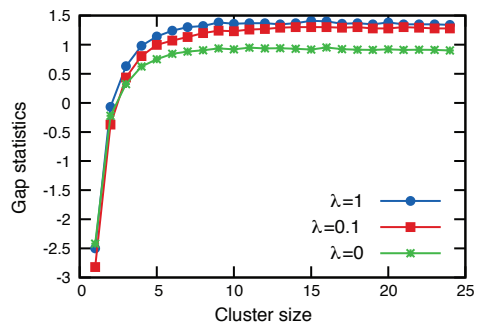

Fig. 3. Gap statistics on patient profiles ($U^{(1)}$). X-axis is the number of clusters and Y-axis is the gap statistics value of varying cluster sizes on factor matrix $U^{(1)}$ generated from SNeCT with graph regularization values of $\lambda = 1$, $\lambda = 0.1$, and $\lambda = 0$.

Figure 3 shows gap statistic values.

Cluster analysis

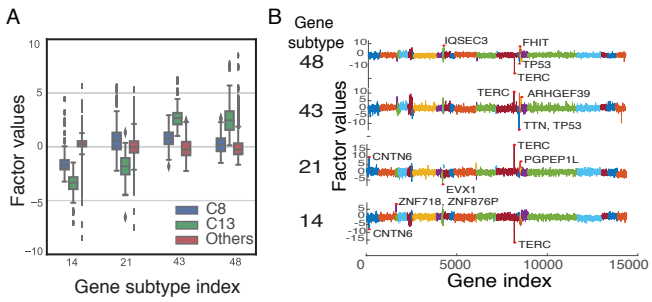

Fig. 4. Subtype analysis of two OV clusters, C8 and C13. A shows significant gene subtypes and patient factor norms distribution mapped to gene subtypes. B shows factor values of the significant gene subtypes.

Figure 4 shows distribution of factor values of significant gene subsets.

Figure 5 shows survival curves of BLCA, COAD, HNSC, LUAD, and LUSC cohorts.

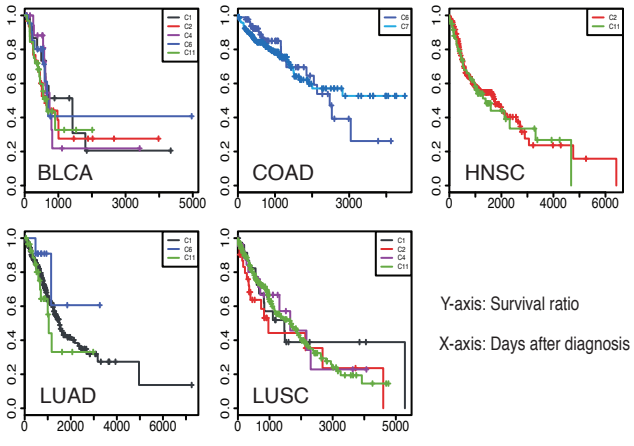

Fig. 5. Survival analysis of clusters formed for cohorts BLCA, COAD, HNSC, LUAD, and LUSC.