Contents

Supplementary

A key distinction from traditional deconvolution tasks

Specifically, in the NanoString CosMx spatial single-cell dataset of non-small cell lung cancer (NSCLC), the proportion of each cell's *k*nearest neighbors sharing the same niche label (niche kNN purity) and the proportion sharing the same cell-type label (cell-type kNN purity) were computed. The results (Fig. 1a) showed that local niche consistency was substantially lower than that of cell types (niche kNN purity [1] = 50.96%; cell-type kNN purity = 83.66%). This discrepancy suggests that the dominant axes of variation captured by gene expression are more strongly aligned with cell-type–associated transcriptional programs than with niche-level signals, which represent multicellular cooperative units. Because a given niche typically comprises multiple cell types coexisting within defined spatial neighborhoods, niche labels inherently correspond to mixed expression distributions at the single-cell level across cell types, rather than a pseudo–cell type that forms a tight cluster in expression space and can be separated by simple linear unmixing.

Therefore, recovering niche abundances from conventional tissue-level data is not a trivial substitution of label sets, but a substantially more challenging identification problem: it requires disentangling cross–cell-type coordinated functional patterns—driven by spatial neighborhood constraints and intercellular interactions—from mixed signals dominated by cell-type variation. Moreover, most existing deconvolution methods do not explicitly leverage structural priors carried by spatial single-cell data, such as local continuity, spatial autocorrelation, and neighborhood relationships. As a consequence, the strategy of first annotating niches and then applying conventional deconvolution is face intrinsic challenges in statistical identifiability and robustness. Motivated by these considerations, NicheDECODE is proposed to explicitly integrate spatial structural evidence and functional module priors during both training and inference, improving the separability of niche-level signals and enabling reliable estimation of niche abundances from conventional tissue-level datasets.

[1] To quantify local label consistency, we defined k-nearest neighbor (kNN) purity based on the neighborhood graph constructed from gene expression profiles. For each cell, we identified its k nearest neighbors from the connectivity graph and computed the proportion of neighbors sharing the same label (e.g., niche label or cell-type label) as the query cell. The kNN purity was then defined as the average of these proportions across all cells. A higher kNN purity indicates stronger local clustering of cells with the same label in expression space, reflecting greater separability and homogeneity of the corresponding biological category.

Further validation on ROSMAP cohort

To further evaluate robustness, NicheDECODE models were independently trained by two distinct spatial reference datasets (Chen et al. and Maynard et al.) and subsequently applied to ROSMAP cohort. Notably, both training settings yielded highly consistent trends: white matter (WM) abundance was positively correlated with Alzheimer's disease (AD) progression, whereas

gray matter abundance showed a negative correlation with Braak stage (Fig S3). These results indicate that NicheDECODE reliably recovers spatial structural alterations from tissue-level data and remains stable across reference sources.

Using the DLPFC single-cell atlas from Song *et al.* as reference, we performed cell-type deconvolution of the ROSMAP cohort using CIBERSORTx (nine cell types; Fig. S5a). To assess predictive relevance, we trained multi-layer perceptron (MLP) models to predict Braak stage using three feature sets separately: (i) bulk gene expression, (ii) inferred cell-type proportions, and (iii) inferred niche proportions (Fig. S5b). Across multiple hyperparameter settings, models based on niche proportions consistently achieved the lowest mean squared error (MSE), indicating that niche-level composition provides stronger predictive signal for AD progression than either gene expression or cell-type proportions alone.

Furthermore, linear regression analysis (linregress function of SciPy) revealed that none of the individual cell-type proportions were significantly associated with Braak stage (Fig S5c), reinforcing the notion that niche-level organization captures disease-related variation beyond cell-type abundance alone.

Methods

Mapping METABRIC samples to TCGA-defined niche subtypes

First, a sample-level niche feature matrix was constructed in the TCGA-BRCA cohort using the niche abundance profiles of individual samples. For patients with multiple samples, the abundance of each niche component was averaged across samples to obtain a single patient-level profile. The resulting TCGA niche features were then standardized using z-score transformation. K-means clustering was performed on the standardized profiles to classify TCGA patients into two niche-defined subtypes. The number of clusters was set to (k=2), the random seed was fixed at 2026, and multiple random initializations were used to improve clustering stability.

To map the METABRIC cohort onto the two niche subtypes defined in TCGA, the METABRIC niche abundance profiles were first standardized using the means and standard deviations estimated from the TCGA cohort. The standardized METABRIC profiles were then supplied to the K-means model fitted in TCGA. When the prediction yielded both subtypes, each METABRIC sample was assigned a TCGA-like niche subtype according to its nearest TCGA niche centroid.

In practice, direct nearest-centroid assignment classified all METABRIC samples into a single TCGA niche subtype. We therefore used a centroid-axis projection strategy as an alternative mapping approach. In the standardized TCGA niche feature space, the vector extending from the TCGA C1 centroid to the TCGA C2 centroid was defined as the C1-to-C2 projection axis. Each standardized METABRIC sample was projected onto this axis to obtain a continuous projection score. Because the axis was oriented from C1 toward C2, a higher projection score indicated a more TCGA C2-like niche state, whereas a lower score indicated a more TCGA C1-like state. Finally, METABRIC samples were divided into C1-like and C2-like groups using the median projection score within the METABRIC cohort as the cutoff. Samples with projection scores below the median were classified as C1-like, whereas those with scores greater than or equal to the median were classified as C2-like.

Simulation of cfDNA methylation profiles and prediction of niche abundances

We first identified niche-associated methylation features using paired samples from the TCGA-BRCA cohort. Samples with available RNA-sequencing data were used to match tissue HM450 methylation beta-value profiles with the seven niche abundances inferred by NicheDECODE. For each niche state, samples were ranked according to their inferred abundance, and the top and bottom quartiles were defined as the high- and low-abundance groups, respectively. Differentially methylated CpG sites between the two groups were identified using a two-sided Wilcoxon rank-sum test. CpG sites with ($P < 0.05$) and an absolute difference in mean beta value greater than 0.1 were retained. For each niche state, up to 5,000 positively associated and 5,000 negatively associated CpG sites were selected.

To generate simulated cfDNA methylation profiles, the selected tumour tissue methylation profiles were linearly mixed with healthy cfDNA methylation backgrounds. The healthy-background fraction was randomly sampled between 0.2 and 0.6, and the tumour fraction was defined as its complement. Each healthy-background profile was generated by randomly combining three healthy cfDNA samples and introducing mild multiplicative noise. The resulting mixed CpG beta values were used as model inputs, whereas the niche abundance profiles inferred from the corresponding TCGA transcriptomic samples were retained as reference labels.

A multilayer perceptron model was implemented in PyTorch to predict the relative abundances of the seven niche states from the simulated cfDNA methylation profiles. Samples were divided equally into training and test sets using stratified sampling based on the previously defined TCGA ($k=2$) niche subtypes. Model performance was evaluated in the independent test set by calculating Pearson correlation coefficients between predicted and reference abundances for each niche state. The predicted niche abundance profiles in the test set were subsequently standardized and grouped into two clusters using spectral clustering. The prognostic relevance of the resulting groups was evaluated using Kaplan–Meier analysis, the log-rank test and Cox proportional-hazards regression.

Compatibility constraints of methods with proteomics data

In both proteomics datasets, CIBERSORTx failed to detect significant differential proteins across niches. This is likely attributable to the comparatively weaker inter-niche molecular contrast in proteomic data relative to transcriptomic data, together with the substantially smaller number of available molecular features.

RCTD could not be successfully applied to the CITE-seq dataset for similar reasons, as insufficient discriminative molecular features were identified. In the CODEX dataset, RCTD was further incompatible because the measured protein intensities are not provided in integer count format, which is required by the method.

UCDSelect, as a large pretrained model developed on transcriptomic data, relies on modality-specific learned representations that do not readily generalize to proteomics data.

For these reasons, the three methods were excluded from comparative analyses in the proteomics benchmarking experiments.

Details of Comparison Methods

ScpDeconv

ScpDeconv is a deep learning-based method that employs autoencoders to effectively explore the relationships between tissue proteomics data and single-cell proteomics data. It can interpolate features with high information content that are not present in the single-cell reference, thereby enhancing the reliability of the deconvolution results. In this study, when using ScpDeconv, the training data were generated from pseudo-tissue samples randomly drawn from a uniform distribution. The tissue data is scaled to the range of [0, 1] using the MinMaxScaler() class from the Sklearn preprocessing module. When employing this method, refer to the example workflow provided in its GitHub repository and adhere to its default parameters.

CIBERSORTx

CIBERSORTx is a deconvolution method based on support vector regression that can derive gene expression features from single-cell reference data and estimate cell type proportions. CIBERSORTx is a user-friendly web tool that requires feature alignment of the single-cell reference matrix and the data to be deconvoluted. Users should structure the data according to the tutorial examples and run the method using the default parameters.

Tangram

Tangram is a deep learning-based spatial mapping deconvolution algorithm that constructs a probability matrix for mapping cells to spatial locations, aligning single-cell and spatial transcriptomic expression using cosine similarity loss to achieve deconvolution of cell type composition in spatial data. For configuration, we use the deconvolution examples provided by Tangram on GitHub and apply cluster mode (mode='clusters') to map and output normalized cell type density matrices. The single-cell data is normalized using Scanpy's pp.normalize_total, and cell types with fewer than 2 samples are filtered out. We obtain the top 200 marker genes for each cell type using Scanpy's tl.rank_genes_groups, then intersect these with the spatial data to get common genes. Genes with no information are removed through tg.pp_adatas. During mapping, we use the default parameters from the Tangram authors' tutorial without Squidpy to ultimately output the deconvolution results of spatial points and niche types.

Ucdselect

UCD is a deconvolution algorithm based on a pre-trained model. The model is pre-trained using 10 million pseudotissue samples from the author's integrated scRNA-seq training database. When using UCD, the first step is to match the gene expression features of the input reference single-cell data and spatial mixed data with the feature dictionary of the pre-trained data, ensuring that they share common features with the pre-trained model. Since the deconvolution results of the UCD_base mode refer to cell types from the pre-trained dataset, this paper employs the UCD_select mode. In this mode, niche type pairs required for UCD_select. Subsequently, the API is called to invoke the cloud-based pre-trained model, which performs deconvolution on the input data and returns the abundance matrix of niche types.

RCTD

RCTD is a spatial transcriptomics deconvolution method implemented in R, based on a mixed linear model framework that corrects for batch effects. It optimizes model parameters through maximum likelihood iterative optimization to compute the posterior probability distribution of niche type composition at each spatial point, decomposing the gene counts of that spatial point into a weighted

sum of the contributions from each niche type. This paper follows the Bioconductor tutorial, constructing a puck that combines spatial expression data and spatial location information. The createRCTD() function is then used to preprocess the puck data and single-cell reference. This function filters spots and genes based on thresholds such as UMI counts and identifies differentially expressed genes. It creates niche type profiles from the reference, setting CELL_MIN_INSTANCE=20 in the createRCTD() function to filter out low-abundance niche types. Finally, it specifies doublet_mode="full" in the run.RCTD() function to enable a complete mode that allows for an unrestricted number of niche types, resulting in the deconvolution outcomes.

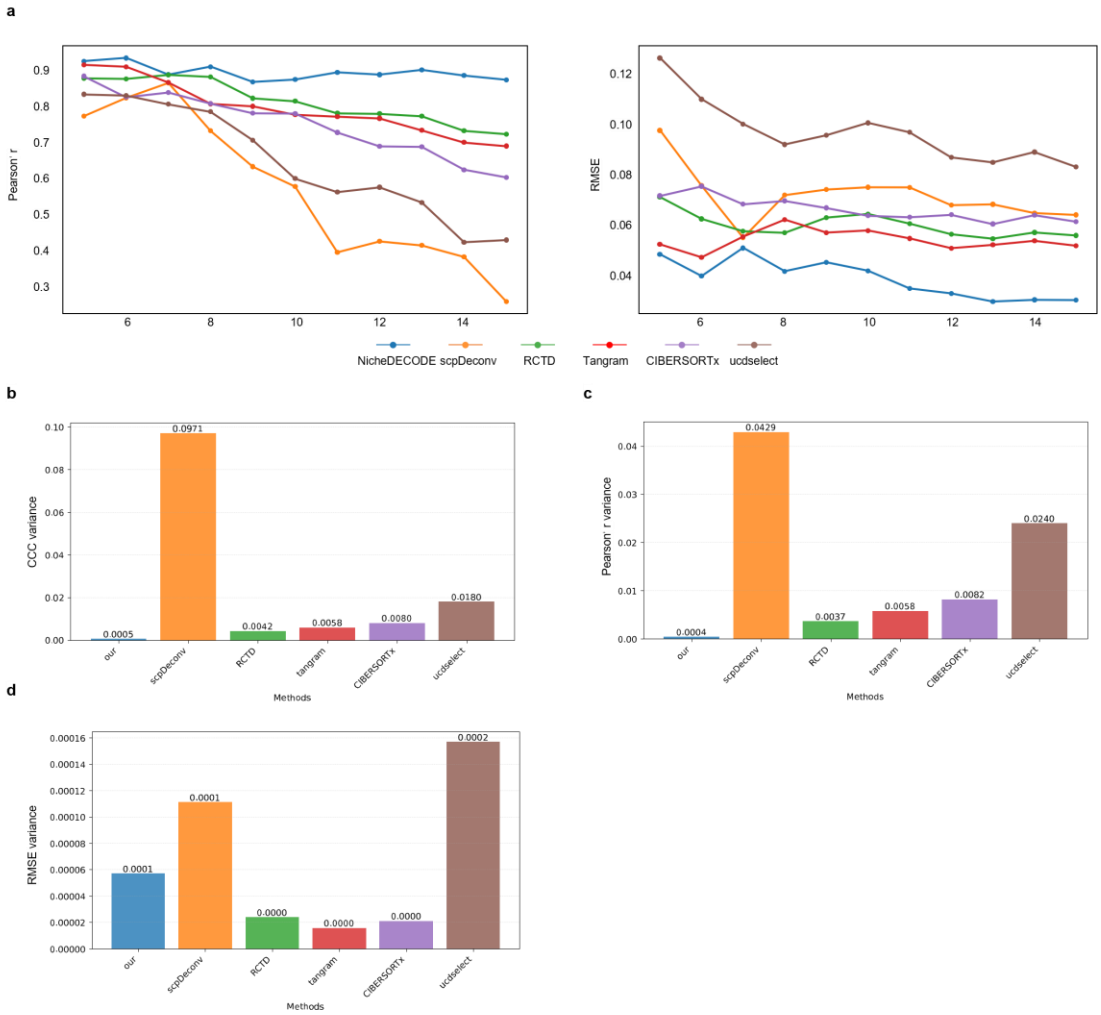

Fig. S1| Additional results on CosMx NSCLC data. (a) Line plots illustrating Pearson's r (left) and RMSE (right) of NicheDECODE, scpDeconv, RCTD, Tangram, CIBERSORTx, and ucdselect across niche resolutions ranging from 5 to 15 clusters. (b-d) Variance of CCC (b), Pearson's r (c), and RMSE (d) across niche resolutions (5 – 15 clusters) for each method.

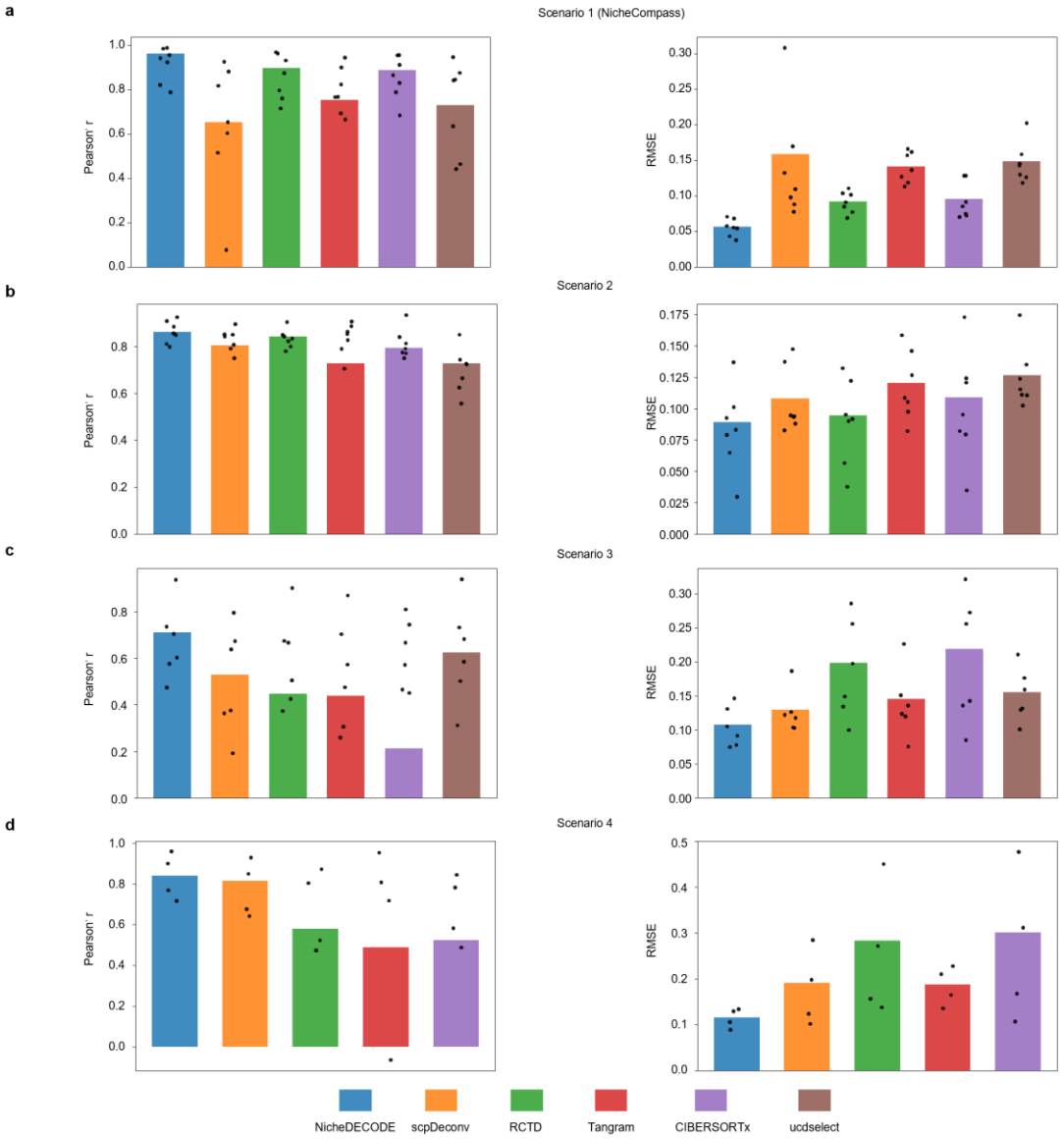

Fig. S2| Pearson'r and RMSE of different methods across four generalization scenarios. Pearson'r and RMSE results for NicheDECODE, scpDeconv, RCTD, Tangram, CIBERSORTx, and ucdselect across Scenario 1 (a), Scenario 2 (b), Scenario 3 (c), and Scenario 4 (d).

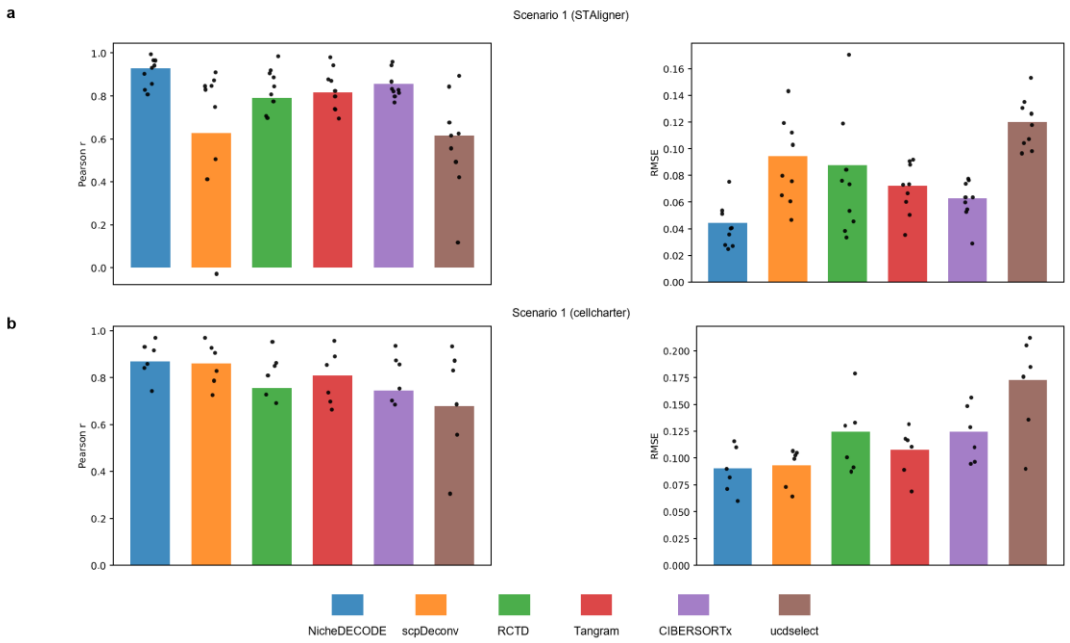

Fig. S3| Pearson'r and RMSE of different methods across scenario 1. Pearson'r and RMSE results for NicheDECODE, scpDeconv, RCTD, Tangram, CIBERSORTx, and ucdselect across Scenario 1 STAligner (a), Scenario 1 cellcharter (b).

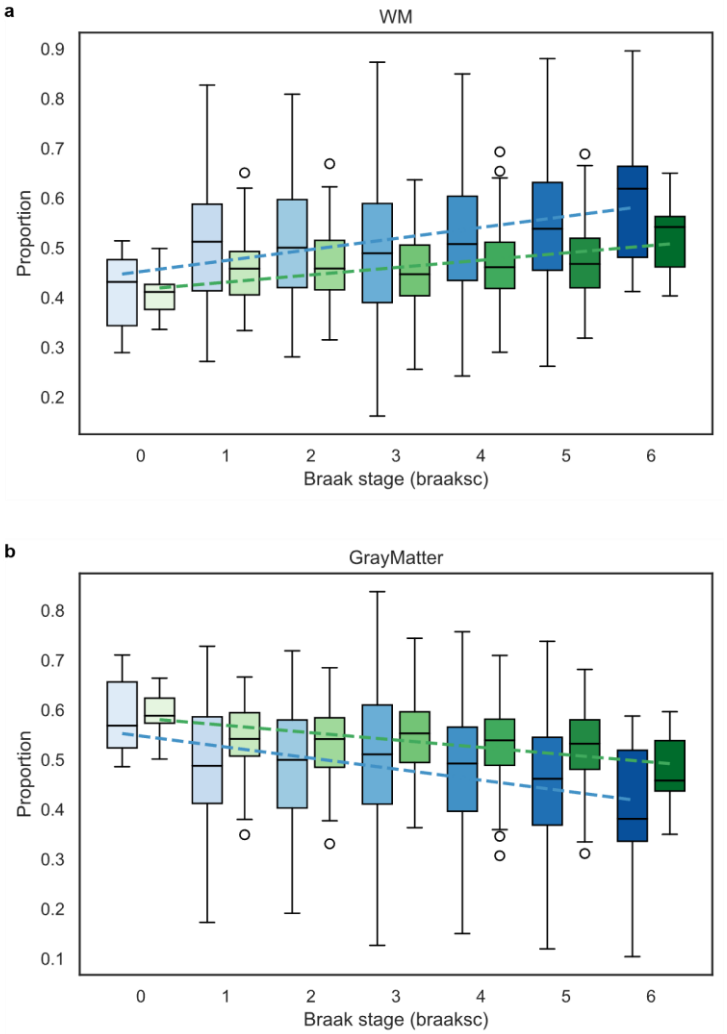

Fig. S4| Box plots showing changes in white matter (a) and gray matter (b) abundances across Alzheimer's disease Braak stages. AD cortex (blue) and healthy cortex (green) references. Notations: Box plots show the median (centre line), the 25th–75th percentiles (bounds of the box), and the minimum and maximum values (whiskers); dots indicate outliers.

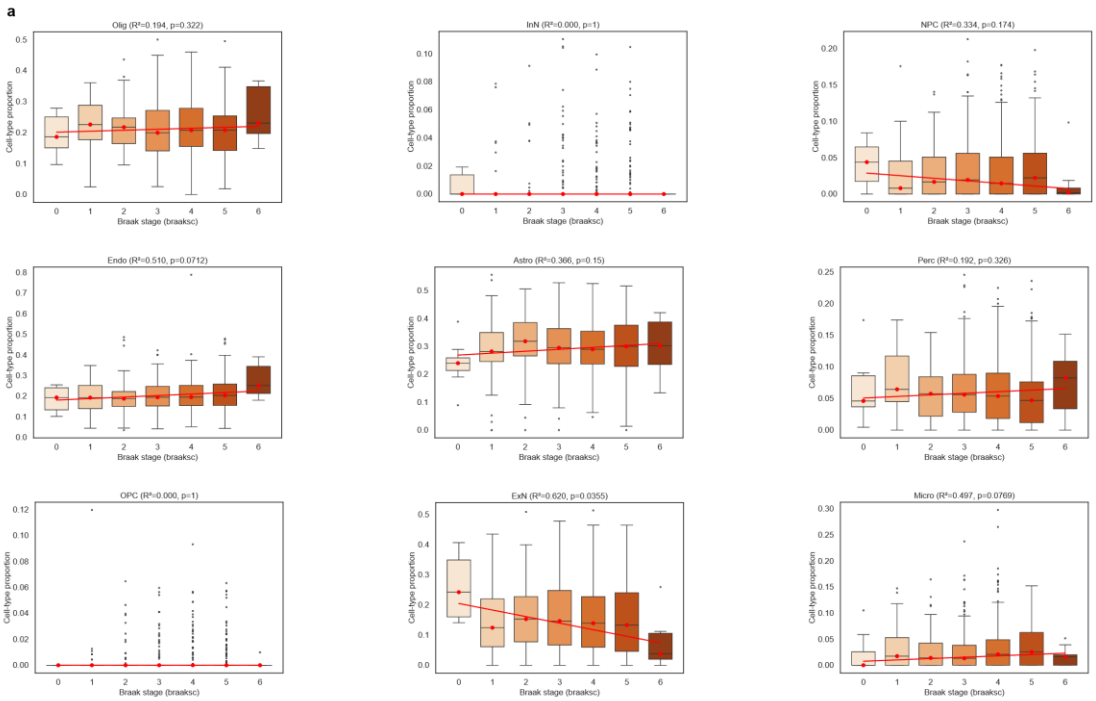

Fig. S5| Cell-type deconvolution results on the ROSMAP cohort. (a) Box plots of cell-type abundances in the ROSMAP cohort inferred by CIBERSORTx. The x-axis represents samples from different AD Braak stages, and the y-axis represents the cell-type proportions. (stage-wise n = 10, 53, 76, 201, 208, 275, and 9). Regression lines were fitted by ordinary least squares using the median abundance within each Braak stage, Accordingly, both R^2 and the p value for the slope (from the stage-median regression) were computed based on these stage medians using the linregress-function from the SciPy library. Notations: Box plots show the median (centre line), the 25th–75th percentiles (bounds of the box), and the minimum and maximum values (whiskers); dots indicate outliers.

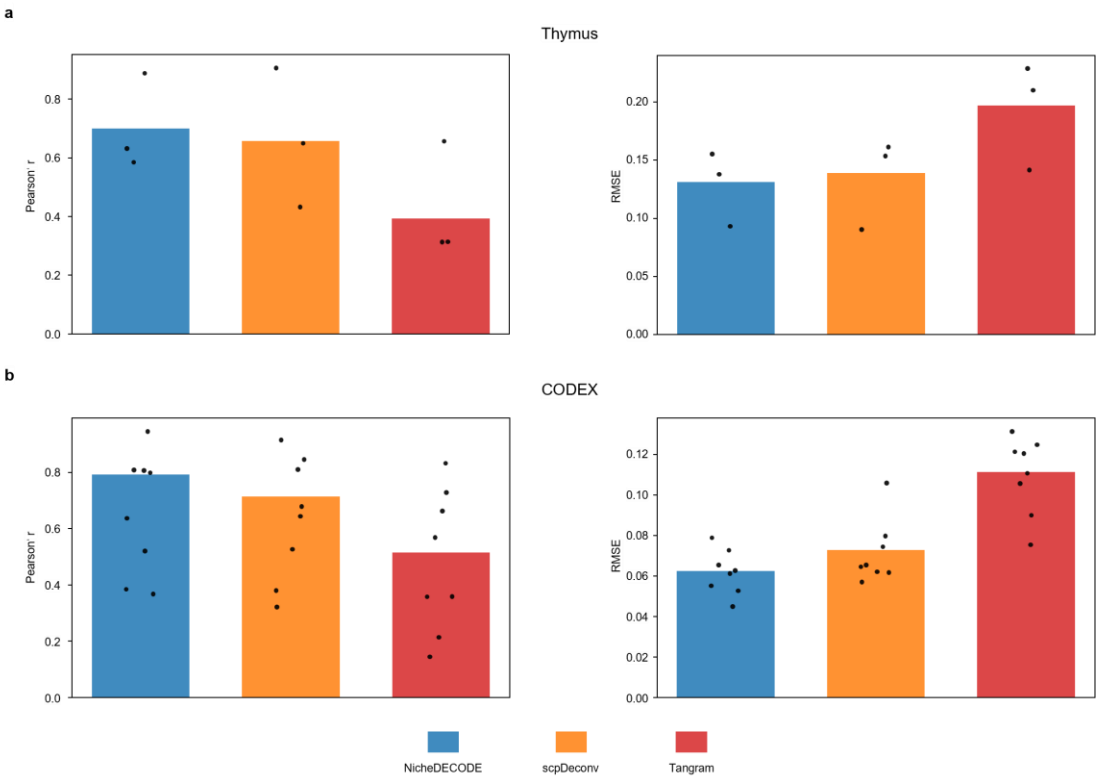

Fig. S6| Pearson'r and RMSE of different methods across proteomics data. Pearson'r and RMSE results for NicheDECODE, scpDeconv, and Tangram on the Thymus (a) and CODEX (b) datasets.