Cooperative multi-view integration with Scalable and Interpretable Model Explainer

Jerome J. Choi[1,2], Noah Cohen Kalafut[2,4], Tim Gruenloh[3], Corinne D. Engelman[1], Tianyuan Lu[1,3,◇] & Daifeng Wang[2,3,4,◇,*]

[1]*Department of Population Health Sciences, University of Wisconsin-Madison School of Medicine and Public Health*
[2]*Waisman Center, University of Wisconsin-Madison*

[3]*Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison School of Medicine and Public Health*
[4]*Department of Computer Sciences, University of Wisconsin-Madison*

[◇]*These authors jointly supervised this work.*

[*]*Corresponding author. daifeng.wang@wisc.edu*

Abstract

Single-omics approaches often provide a limited perspective on complex biological systems, whereas multi-omics integration enables a more comprehensive understanding by combining diverse data views. However, integrating heterogeneous data types and interpreting complex relationships between biological features—both within and across views—remains a major challenge. To address these challenges, we introduce COSIME (Cooperative Multi-view Integration with a Scalable and Interpretable Model Explainer). COSIME applies the backpropagation of a learnable optimal transport algorithm to deep neural networks, thus enabling the learning of latent features from several views to predict disease phenotypes. It also incorporates Monte Carlo sampling to enable interpretable assessments of both feature importance and pairwise feature interactions for both within and across views. We applied COSIME to both simulated and real-world datasets—including single-cell transcriptomics, spatial transcriptomics, epigenomics and metabolomics—to predict Alzheimer's disease-related phenotypes. Benchmarking of existing methods demonstrated that COSIME improves prediction accuracy and provides interpretability. For example, it reveals that synergistic interactions between astrocyte and microglia genes associated with Alzheimer's disease are more likely to localize at the edges of the middle temporal gyrus. Finally, COSIME is also publicly available as an open source tool.

Main

Single-omics approaches offer limited insight into complex biological systems, as each molecular layer (e.g., genomics, transcriptomics, or metabolomics) functions in coordination with others. Multi-view data integration combines multiple layers to identify novel biomarkers, understand disease mechanisms, and improve phenotype prediction[1,2]. Machine learning can

efficiently analyze large-scale multi-view data, modeling complex cross-view interactions that influence phenotypes.

Several methods have been developed for multi-view data integration. Cooperative learning[3] is a supervised learning method that combines squared-error loss with an agreement penalty encouraging feature similarity across views. The penalty strength is fixed and manually tuned, limiting dynamic adaptation and end-to-end learning of cross-view interactions. DIABLO[4] identifies shared structure across omic layers through supervised feature selection, but assumes homogeneity in their association with the outcome, which may not hold in heterogeneous datasets. MOFA+[5] is an unsupervised factor analysis method that models latent linear structures across views, but it lacks mechanisms for interaction analysis or interpretability of prediction features. Furthermore, these models typically assume linear relationships and do not leverage deep learning or optimal transport.

Interpretability is another major challenge. SHAP (SHapley Additive exPlanations)[6] quantifies feature attribution but is computationally expensive for complex models and can assign interaction contributions only for tree-based models. LIME[7] provides local interpretability for individual predictions but lacks global explanations and does not account for feature interactions.

To address these limitations, we introduce Cooperative Multi-view Integration with Scalable and Interpretable Model Explainer (COSIME). COSIME consists of two components: (1) an integration component that uses deep neural encoders and Learnable Optimal Transport (LOT) to align views in a joint latent space and predict disease phenotypes; (2) an interpretation component that estimates feature importance (FI) and pairwise interactions using Shapley values and Shapley-Taylor indices. We evaluate COSIME on simulated and real-world datasets to predict Alzheimer's disease (AD) phenotyes using transcriptomics, metabolomics, epigenomics, and spatial transcriptomics, and we demonstrate its strong predictive performance, flexibility and interpretability.

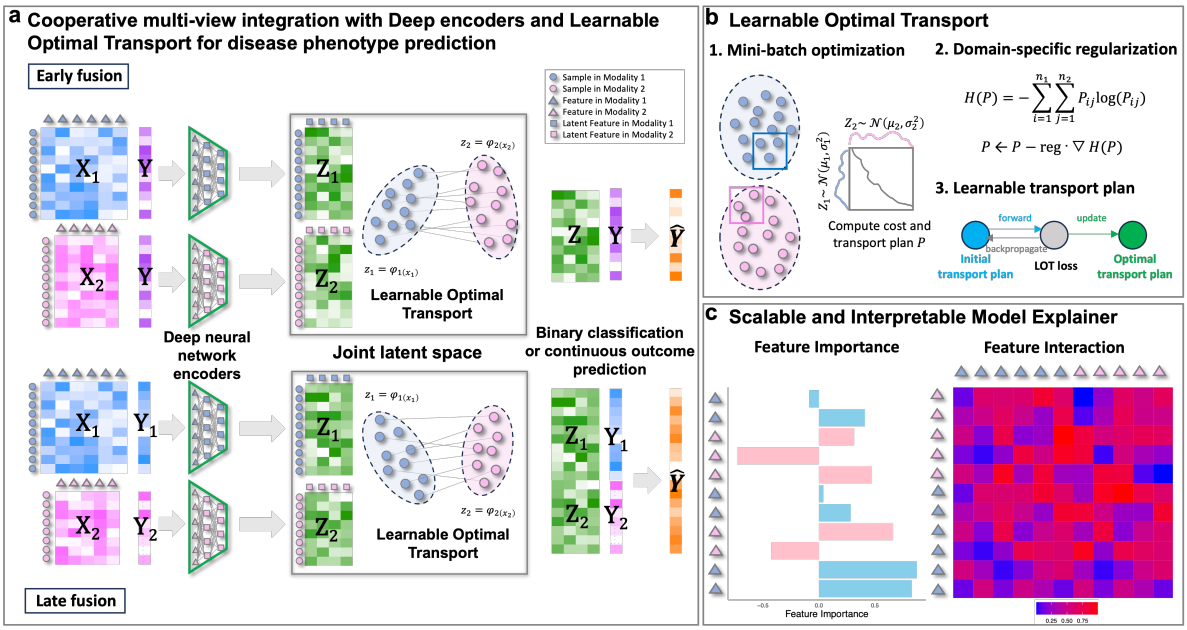
Results

Overview of COSIME

COSIME framework.

Fig. 1 provides an overview of COSIME. The first component involves integrating multi-view data for disease phenotype prediction by leveraging deep learning-based encoders and learnable optimal transport (LOT), which model both linear and nonlinear relations more flexibly than traditional linear methods. COSIME effectively captures the complex, multi-layered interactions between different omic modalities—such as transcriptomics and epigenomics, transcriptomics and spatial transcriptomics, and transcriptomics and metabolomics—while preserving the distinct characteristics of each data type.

a, COSIME integrates multi-view data for disease phenotype prediction through a three-step process: (1) Each data view is passed through separate deep neural network encoders (deep encoders). (2) The learnable optimal transport (LOT) aligns and merges these features into a joint latent space. (3) The integrated latent representation is then used to predict disease phenotypes. **b,** LOT integrates heterogeneous datasets by offering three key advantages: (1) a learnable transport plan that adapts during training, (2) mini-batch processing for scalability, and (3) domain-specific regularization to enhance alignment between source and target distributions. **c,** The scalable and interpretable model explainer interprets model predictions by identifying important features and feature interactions. It quantifies both individual feature importance (FI) and pairwise feature interactions to reveal synergistic or antagonistic effects on the model's output.



LOT offers three main advantages over conventional optimal transport methods. First, it uses a learnable transport plan, enabling dynamic alignment of heterogeneous data. Second, LOT supports mini-batch processing, making it scalable for large datasets while maintaining computational efficiency. Finally, LOT can incorporate domain-specific regularization, allowing improved handling of view-specific differences and robustness over traditional methods.

The second component of COSIME focuses on computing feature importance (FI) values for each view, as well as pairwise interaction values for both within-view and across-view interactions. To achieve this, COSIME employs the Scalable and Interpretable Model Explainer, which leverages the Shapley values and Shapley-Taylor indices[8] to compute both feature importance and pairwise feature interactions, respectively. Monte Carlo sampling and batch processing enable scalable computation and enhance interpretability. Additionally, it allows for the identification of the directionality of interactions—whether they exhibit synergism (complementary effects) or antagonism (conflicting effects)—offering deeper insights into how features interact in the model.

COSIME achieves strong predictive performance by modeling across-view relationships within a shared latent space. It integrates heterogeneous data and identifies feature importance and interactions to improve accuracy and interpretability. While these two main components of COSIME can be used together for both enhanced prediction and interpretability, they are also designed to be used independently depending on the specific needs of the analysis. Together, they address the challenges of heterogeneous data and complex feature interactions, enabling biomarker discovery and mechanistic understanding.

In the following sections, the performance metrics were compared across different methods

using the mean and ±1.96 times the standard deviation from 5-fold cross-validation. For the best-trained models across the different multi-view datasets, we computed feature importance for each data view and feature interaction values both within and across views. All results for the multi-view predictions using COSIME models and other methods are provided in the **Supplementary Data 1** and **2**. Feature importance and interactions for the simulated data can be found in the **Supplementary Data 3–6**.

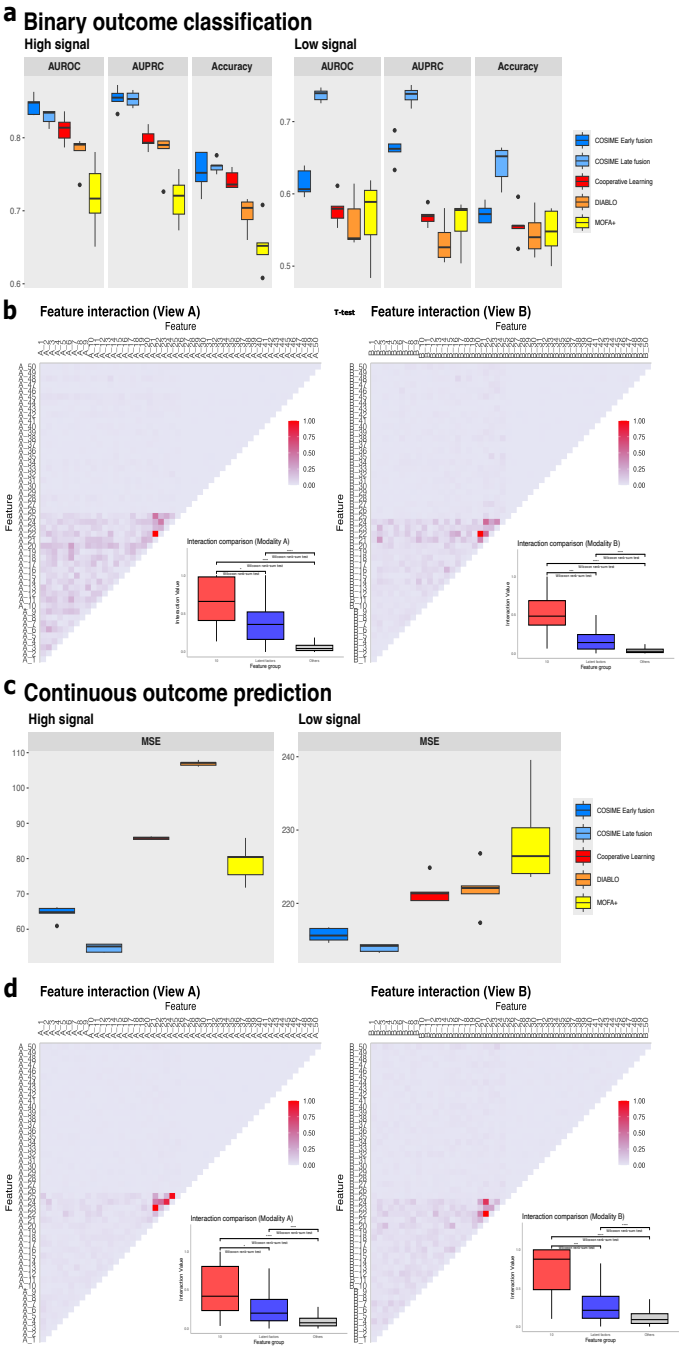
Simulation study

Multi-view data were generated under different signal levels (high and low) and outcome types (binary and continuous). COSIME early fusion performed best on high-signal binary data, outperforming CL[3], DIABLO[4], and MOFA+[5] with logistic regression (AUROC: 0.845 ± 0.026, AUPRC: 0.854 ± 0.029, accuracy: 0.754 ± 0.054). COSIME late fusion also outperformed benchmarks (AUROC: 0.828 ± 0.021, AUPRC: 0.853 ± 0.021, accuracy: 0.761 ± 0.019) (**Fig. 2a**). For low-signal binary data, late fusion yielded the best results (AUROC: 0.737 ± 0.017, AUPRC: 0.736 ± 0.025, accuracy: 0.640 ± 0.052), and early fusion still surpassed all benchmarks.

Predictive performance and feature interaction for simulated data.

a, Predictive performance for binary outcome classification using COSIME early fusion, COSIME late fusion, CL, DIABLO, and MOFA+ with logistic regression on both high-signal and low-signal multi-view datasets. **b,** Heatmaps depicting the pairwise interactions of the first 50 features from views A and B for binary outcome prediction, along with box plots and two-sided Wilcoxon rank-sum tests for three interaction groups: (Group 1) 10 interaction terms artificially introduced during data generation, (Group 2) 1,790 interactions involving latent features, and (Group 3) 3,150 other interactions. The box plots display the distribution of interaction values, including the first quartile, median, and third quartile. Asterisks indicate statistical significance: * for p–value < 0.05 and **** for p–value < 0.0001. **c,** Prediction performance for continuous outcome prediction using COSIME early fusion, COSIME late fusion, CL, DIABLO, and MOFA+ with regression on both high-signal and low-signal multi-view datasets. **d,** Heatmaps depicting the pairwise interactions of the first 50 features from views A and B for continuous outcome prediction, along with box plots and pairwise two-sided t-tests for three interaction groups: (Group 1) 10 interaction terms artificially introduced during data generation, (Group 2) 1,790 interactions involving latent features, and (Group 3) 3,150 other interactions. The box plots display the distribution of interaction values, including the first quartile, median, and third quartile. Asterisks indicate statistical significance: *** p–value < 0.001 and **** for p–value < 0.0001.

Fig. 1: Prediction performance and feature interaction from simulated data.



Feature interaction values from COSIME early fusion (high-signal binary data) were grouped as follows: (1) artificially introduced, (2) involving latent features and (3) others. Scaled interactions for the first 50 features are shown in **Fig. 2b**. Wilcoxon rank-sum tests show Group 1 > Group 2 ($p < 0.05$), Group 1 > Group 3 ($p < 0.0001$), and Group 2 > Group 3 ($p < 0.0001$), in both views A and B. For high-signal continuous outcomes, COSIME achieved the lowest MSE (54.701 ± 2.391), outperforming CL, DIABLO, and MOFA+ (MSE: 64.490 ± 4.130) (**Fig.**

2c). On low-signal data, COSIME early fusion again performed best (MSE: 215.707 ± 1.833), while late fusion outperformed benchmarks (MSE: 218.425 ± 1.099). Interaction analysis for high-signal continuous data (top 50 features in **Fig. 2d**) shows consistent results: Group 1 > Group 2 ($p < 0.05$) and Group 3 ($p < 0.0001$); Group 2 > Group 3 ($p < 0.0001$) across both views. Overall, COSIME consistently outperforms all benchmarking methods and recovers synthetic interactions introduced during data generation.

To confirm the superiority of COSIME for multi-view integration and outcome prediction, we compared its performance against other methods. The results are presented in **Supplementary Table 1** and **Supplementary Figs. 2 and 3**. COSIME with both early and late fusion outperforms its single-view methods, single-view regressions and benchmark methods, including aligned OT with regression and single-cell multi-omic integration with optimal transport (SCOT) with regression. These findings highlight the strength of the COSIME integration and show that LOT in COSIME surpasses other OT-based approaches.

To illustrate the computational efficiency of COSIME, we report runtimes for various dataset sizes in **Supplementary Table 3**.

To justify the superiority of COSIME for multi-view integration and outcome prediction, we compared its performance against other methods. Results are presented in the **Supplementary Table 1** and **Supplementary Figs. 2** and **3**. COSIME with both early and late fusion outperformed its single-view methods, single-view regressions, and benchmark methods including Aligned OT with regression and SCOT OT with regression. These findings highlight the strength of COSIME's integration and show that LOT in COSIME surpasses other optimal transport-based approaches.

To provide guidance on the computational efficiency of COSIME, we report run times for varying dataset sizes in the **Supplementary Table 3**.

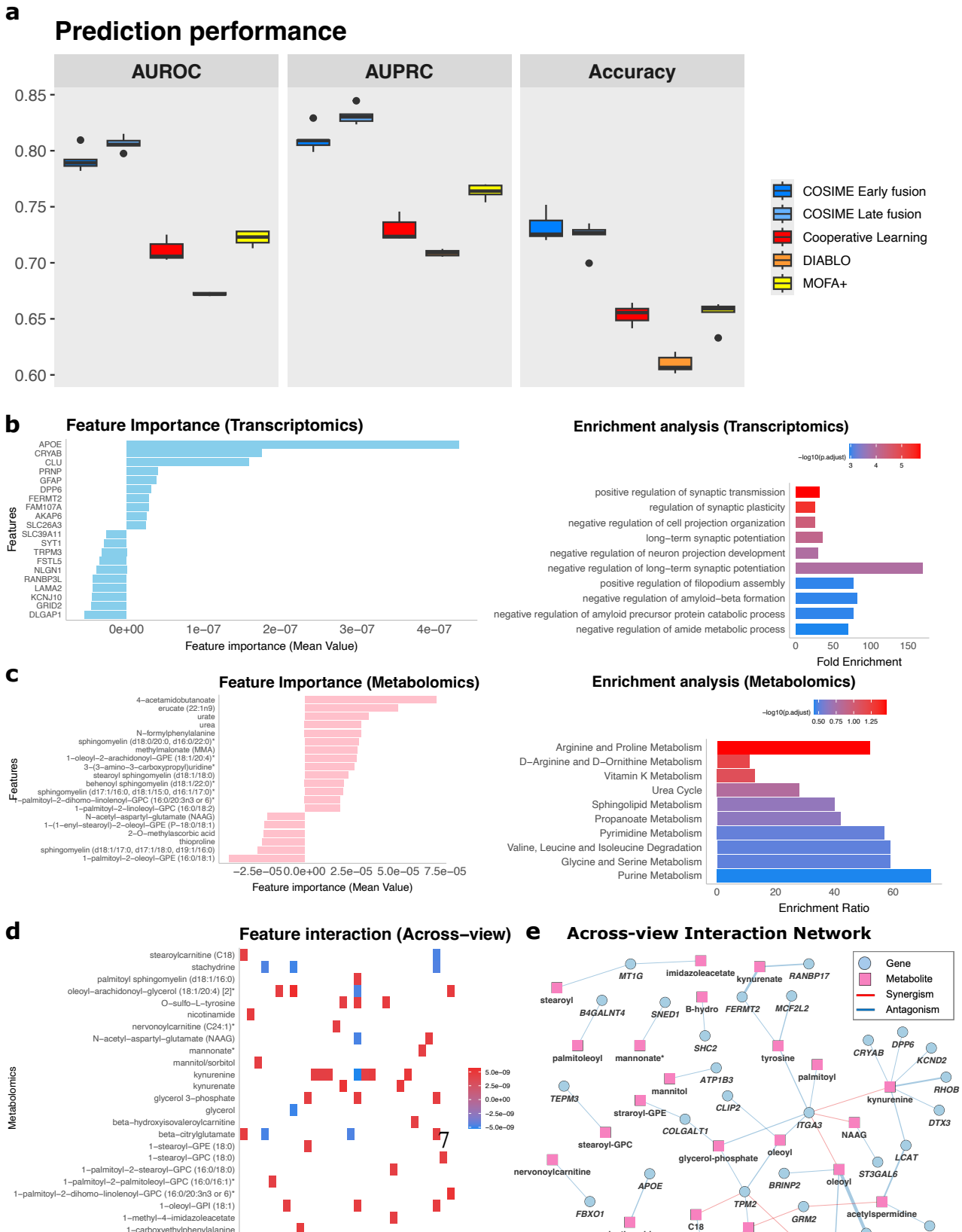
Cognitive diagnosis from transcriptomic and metabolomic data

Multi-view data from scRNA-seq of astrocytes and metabolomics were used to test the prediction performance of COSIME models. COSIME early fusion with matched samples (n=2,286) achieved the best performance for binary classification of Alzheimer's disease (AD) cognitive diagnosis using multi-view datasets (astrocytes and metabolomics) (AUROC: 0.792 ± 0.021, AUPRC: 0.810 ± 0.022, accuracy: 0.732 ± 0.025). To leverage all available samples (4292 metacells in transcriptomics and 2286 samples in metabolomics) across views, we also applied COSIME late fusion for unmatched samples, which yielded comparable results (AUROC: 0.807 ± 0.013, AUPRC: 0.832 ± 0.016, accuracy: 0.723 ± 0.027) (**Fig. 3a**).

Fig. 3

Classifying cognitive diagnoses for AD from transcriptomic (astrocytes) and metabolomic data.

a, Predictive performance using COSIME early fusion, COSIME late fusion, CL, DIABLO, and MOFA+ with logistic regression. **b,** Bar plots showing the top 20 prioritized genes (absolute feature importance values) and enrichment analysis for those genes. **c,** Bar plots showing the top 20 prioritized metabolites (absolute feature importance values) and enrichment analysis for those metabolites. **d,** Heatmap showing the top 50 pairwise across-view (gene-metabolite) interactions. **e,** Across-view interaction network showing relationships between the top 50 pairwise across-view (gene-metabolite) interactions.



We also compared its performance against other methods. Overall, COSIME with both early and late fusion outperformed its single-view methods, single-view regressions, and existing benchmark methods including Aligned OT with regression and SCOT OT with regression. Results are presented in the **Supplementary Table 2** and **Supplementary Fig. 4**.

FI values were used to prioritize features. The top 20 most important genes (based on absolute values) from the transcriptomic data are displayed in **Fig. 3b**, and enrichment analysis reveals that these genes are strongly associated with AD. Notably, processes such as synaptic transmission, amyloid-beta ($A\beta$) formation, and synaptic plasticity are strongly implicated in AD. They represent crucial aspects of AD pathophysiology[9,10].

Fig. 3c shows the top 20 most important metabolites (ranked by absolute feature importance values) from the metabolomic data. Enrichment analysis of these metabolites highlights metabolic pathways such as arginine and proline metabolism vitamin K metabolism, and sphingolipid metabolism. These pathways may contribute to AD by exacerbating the brain's energy deficit, promoting oxidative damage, and driving neuroinflammation[11].

Pairwise feature interaction values are computed, and the top 50 across-view feature interaction values are shown in **Fig. 3d**. *APOE* has synergistic effects with nicotinamide. The *APOE* $\varepsilon 4$ allele is the strongest genetic risk factor for late-onset AD, promoting amyloid-beta aggregation, neuroinflammation, and synaptic dysfunction[12]. Nicotinamide, a form of vitamin B3 and a precursor to NAD^+, plays a critical role in mitochondrial function, oxidative stress response, and DNA repair. Altered nicotinamide metabolism in AD reflects disrupted energy homeostasis and neurodegenerative stress. Together, elevated *APOE* expression and dysregulated nicotinamide metabolism may synergistically exacerbate AD pathology. *APOE*-driven neuroinflammation and amyloid deposition may increase cellular demand for NAD^+, while impaired nicotinamide pathways may fail to meet this demand, leading to mitochondrial dysfunction and neuronal vulnerability. Their combined dysregulation amplifies the biological signals of neurodegeneration, contributing to a stronger predictive signature for AD in machine learning models.

ATP5PF shows antagonistic effects with stachydrine (**Fig. 3d**). *ATP5PF* is a subunit of mitochondrial ATP synthase essential for cellular energy production through oxidative phosphorylation. In AD, reduced *ATP5PF* expression reflects mitochondrial dysfunction and impaired energy metabolism, both of which are key features of neurodegeneration[13]. Stachydrine is a plant-derived metabolite known for its antioxidant and mitochondrial protective effects. It has been shown to reduce oxidative stress, preserve mitochondrial function, and support cellular energy homeostasis in various disease models[14]. They may exhibit an antagonistic relationship in AD, where high levels of stachydrine may counteract the detrimental effects of *ATP5PF* downregulation. This suggests that stachydrine acts as a compensatory factor against mitochondrial dysfunction, potentially buffering the impact of energy failure in the brain.

Furthermore, *TPM2* and *ITGA3* may serve as important hub genes in the across-view network (**Fig. 3e**). These two genes are likely involved in astrocyte function under both normal and pathological conditions, particularly through their roles in inflammation, actin remodeling, and neurovascular interactions. Kynurenine may be a hub metabolite, interacting with genes. Through its role in neuroinflammation and oxidative stress, kynurenine may antagonize the function of key astrocyte genes that support neuronal survival, metabolic stability, and synaptic homeostasis. In AD, elevated kynurenine may disrupt astrocyte-mediated neuroprotection, leading to an inverse or antagonistic relationship with homeostatic astrocyte genes[15].

Feature importance and interactions for the ROSMAP binary classification early fusion are in the **Supplementary Data 7**.

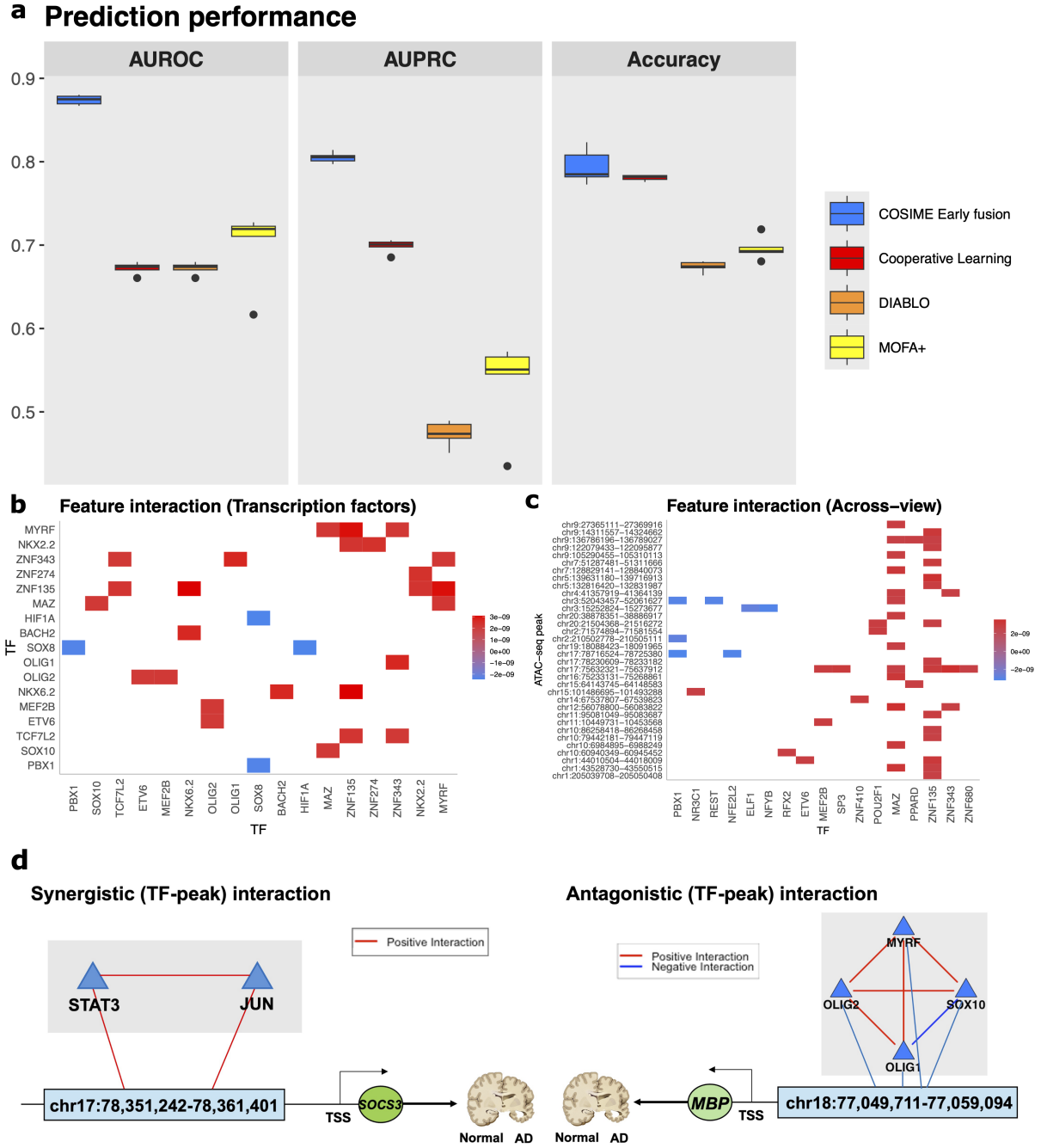
Cognitive status from transcriptomic and epigenomic data

Multi-view data from both scRNA-seq and scATAC-seq of oligodendrocytes were utilized to evaluate the predictive performance of COSIME models. COSIME early fusion performed the best for binary classification of cognitive status (dementia) using multi-view datasets (oligodendrocytes) (AUROC: 0.874 ± 0.011, AUPRC: 0.805 ± 0.013, accuracy: 0.794 ± 0.041) (**Fig. 4a**).

Fig. 4

Classifying dementia from transcriptomic (oligodendrocyte) and epigenomic (oligodendrocyte) data.

a, Predictive performance using COSIME early fusion, CL, DIABLO, and MOFA+ with logistic regression. **b,** Heatmap showing the top 50 pairwise TF interactions. **c,** Heatmap showing the top 50 pairwise across-view (TF-ATAC peak) interactions. **d,** Diagrams illustrating examples of synergistic and antagonistic relationships between a TF-ATAC peak regulating a target gene (TG) at their transcription start site (TSS) and the AD phenotype.



We investigated pairwise interactions between the ten known key TFs involved in oligodendrocytes[16]. For instance, synergistic interactions were identified between SOX10 and MAZ as

well as between MYRF and MAZ. Specifically, SOX10 is critical for oligodendrocyte differentiation[17]. MAZ is involved in regulating genes related to cell proliferation, differentiation, and survival[18]. Their co-expression may suggest a compensatory mechanism for myelin repair in dementia. MYRF is another key player in myelination[19]. Its synergism with MAZ may similarly support remyelination in disease contexts (**Fig. 4b**).

Moreover, antagonistic interaction is identified between SOX8 and HIF1A. SOX8 regulates development and differentiation[20], while HIF1A is a stress-related factor[21] that could suppress differentiation and thereby contribute to dementia pathology. SOX8 also shows antagonism with PBX1, a TF involved in neural cell development[22,23]. This interaction may reflect a regulatory imbalance affecting myelin integrity (**Fig. 4b**).

Based on the top fifty across-view absolute interaction values, we identified several TFs interacting with multiple oligodendrocyte-specific regulatory regions. MAZ showed synergistic interactions with ATAC peak regions mapping to genes like *PTPRF*[24], *ERBB3*[25], *GRID1*[26], *LIMCH1*[27], *FLNC*[28], and *DUSP7*[29]. These interactions suggest MAZ may influence gene networks relevant to neurodegeneration. ZNF135 also showed synergism with peaks linked to *NKX2-2*[30], *PTPRF*[24], *CNTN2*[31], *PSD2*[32], and *SLC6A9*[33], associated with neural development and synaptic function (**Fig. 4c**).

In contrast, PBX1 antagonized open chromatin regions and may reduce accessibility, impairing activation of protective genes such as *CPS1*[34] and *CYTH1*[35]. This may reflect loss of protective oligodendrocyte mechanisms (**Fig. 4c**).

STAT3 and JUN exhibit synergistic activity in the chr17:78,351,242-78,361,401 region containing *SOCS3*. STAT3 regulates JAK-STAT signaling, immune response, and differentiation[36,37], while JUN regulates stress and inflammatory gene expression[38,39]. Both TFs are known to regulate *SOCS3*, a gene upregulated in dementia[40,41]. Their coordination may contribute to *SOCS3* dysregulation and downstream inflammation (**Fig. 4d**).

The four key TFs — SOX10[17], MYRF[17], OLIG1[42], and OLIG2[42] — show synergistic relationships, except SOX10 and OLIG1, which are antagonistic. These TFs regulate oligodendrocyte differentiation and myelination. Their synergy may contribute to oligodendrocyte dysfunction and myelin damage in dementia. All four TFs antagonistically interact with an ATAC peak near *MBP* (chr18:77,049,711-77,059,094), a key myelin gene[43], suggesting suppressed *MBP* expression (**Fig. 4d**).

Feature importance and interactions for the SEA-AD scRNA-seq and scATAC-seq (oligodendrocytes) early fusion are included in the **Supplementary Data 8**.

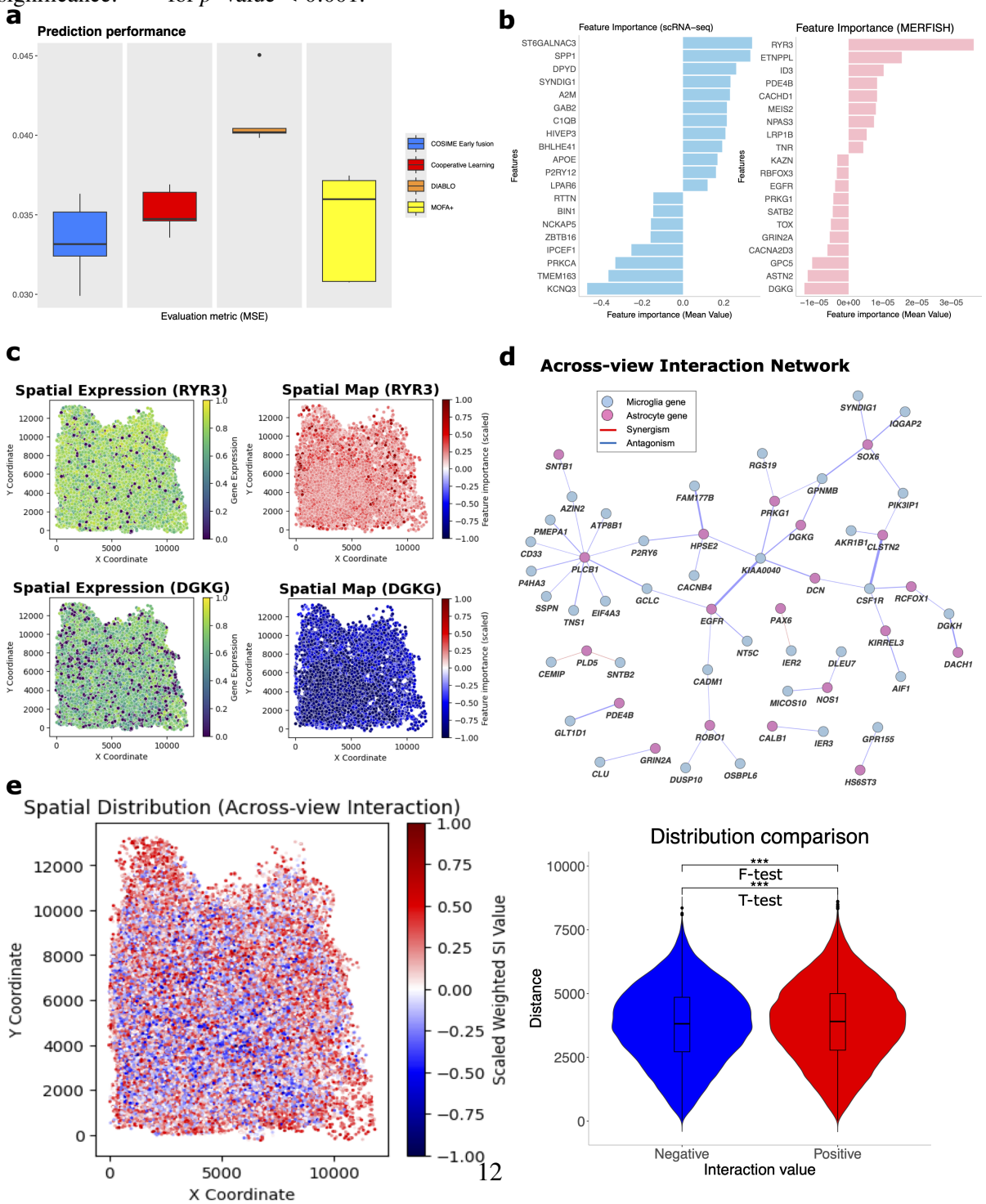
Predicting AD progression scores from transcriptomic data

To assess the predictive capability of COSIME models, multi-view data combining scRNA-seq of microglia and single-cell spatial transcriptomics (MERFISH) of astrocytes were applied. COSIME early fusion performed better than the other benchmarking methods for continuous outcome prediction of Alzheimer's disease progression scores using multi-view datasets (microglia and astrocytes) (MSE:0.033 ± 0.005) (**Fig. 5a**).

Fig. 5

Predicting AD progression scores from transcriptomic (microglia) and spatial transcriptomic MERFISH (astrocytes) data.

a, Predictive performance using COSIME early fusion, CL, DIABLO, and MOFA+ with regression. **b,** Bar plots showing the top 20 prioritized genes (absolute feature importance values) in each view. **c,** Spatial plots showing aexpression level and importance values across cells for *RYR3* and *DGKG*. **d,** Across-view interaction network showing relationships between the top 50 pairwise across-view (gene-gene) interactions. **e,** Spatial distribution of all across-view interaction values colored by the directionality and the distribution of 80,072 positive and 68,619 negative interaction values, with statistical tests (two-tailed t-test and two-tailed F-test) applied to assess differences between two groups. The violin plots including box plots display the distribution of interaction values in the two groups. Asterisk indicate statistical significance: *** for p–value < 0.001.



In microglia, several genes with positive feature importance are linked to AD through roles in neuroinflammation, clearance, and activation. *SPP1* is a microglial marker involved in inflammatory response and amyloid plaque phagocytosis[44]. *APOE*, particularly the E4 allele, is a major AD risk gene affecting clearance and neuroinflammation[12,45]. *C1QB*, part of the complement system, tags damaged neurons for microglial clearance[46,47]. Genes with negative feature importance like *BIN1*, involved in synaptic regulation and endocytosis[48], may be downregulated in AD microglia. **Fig. 5b** also shows that in astrocytes, genes with negative importance like *RYR3* may reflect impaired synaptic regulation via calcium signaling[49,50]. *DGKG*, with positive importance, is involved in lipid metabolism and inflammation[51]. Its dysregulation may affect glutamate signaling, contributing to synaptic damage. Top enriched pathways (**Supplementary Fig. 1**) are largely AD-relevant. We examined spatial distribution of cells based on *RYR3* expression and importance values (**Fig. 5c**). Cells with top 25% *RYR3* feature importance values are significantly central (t-test). Similarly, negatively important *DGKG* cells are also more central (t-test). Together, this suggests central enrichment of downregulated genes and peripheral activity of upregulated genes (**Fig. 5c**). We constructed a gene-gene interaction network from the top 50 across-view interaction values. Astrocyte genes *PLCB1* and *HPSE2* are key hubs linking multiple microglial genes. *PLCB1* regulates phosphatidylinositol signaling and inflammation[52]. *HPSE2* modulates extracellular matrix remodeling and immune responses[53] (**Fig. 5d**). A spatial map based on across-view interactions was generated (**Fig. 5e**). Both t-test and F-test reject the null (t-stat: 18.783, F-stat: 1.242,), indicating distinct spatial distributions. Synergistic interactions are more peripheral, while antagonistic ones are central (**Fig. 5e**). In AD, the middle temporal gyrus (MTG) is vulnerable to early pathological changes including amyloid and tau accumulation[54,55]. Peripheral MTG regions may experience stronger glial activation and synaptic stress[56,57], making them potential early hotspots for AD-related gene activity. While microglia and astrocytes are both essential for maintaining brain function, their interactions at the gene level can be either synergistic or antagonistic[58–60].

FI and interactions for the SEA-AD scRNA-seq (microglia) and MERFISH (astrocytes) early fusion are shown in the **Supplementary Data 9**.

Discussion

We introduce COSIME (Cooperative Multiview Integration with Scalable Interpretable Model Explainer) and apply it to various multi-view datasets, including simulated data, transcriptomics-metabolomics, transcriptomics-epigenomics, and transcriptomics-spatial transcriptomics. COSIME has two key components: first, it integrates multi-view data for disease phenotype prediction by leveraging deep encoders and LOT techniques, combining both unsupervised and supervised learning. Second, it computes feature importance for each view and pairwise interactions for both within- and across-view features.

COSIME addresses several challenges in the existing literature. First, it handles both linear and nonlinear relationships and captures feature interactions. The deep encoders generate embeddings without relying on linear assumptions, which can then be used with linear or nonlinear models. Second, COSIME supports matched and unmatched samples via separate deep encoders, applying early or late fusion accordingly. LOT is a specialized technique that aligns embeddings into a joint latent space. Third, LOT is a unique technique that aligns latent distributions using a learnable transport plan dynamically optimized during training. This allows flexible adaptation to data structure and improves downstream prediction tasks. Fourth, COSIME efficiently computes FI and feature interactions via Monte Carlo sampling

and batch processing. Lastly, COSIME identifies the directionality of interactions—synergism or antagonism—enhancing interpretability and providing intuitive output matrices.

COSIME incorporates LOT, which is developed for integrating two heterogeneous multi-view datasets in predictive modeling. This capability is particularly important given the non-linear nature of omics data—such as transcriptomics, metabolomics, epigenomics, and spatial transcriptomics. Unlike existing methods such as Cooperative Learning[3], DIABLO[4], and MOFA+[5], which are not specifically designed to capture nonlinear relations, COSIME (both early and late fusion) effectively captures nonlinear dependencies and consistently outperforms these methods in phenotype prediction. In addition, COSIME outputs both FI and interaction values from any complex models, whereas SHAP[6] provides interaction matrices only for tree-based models and LIME[7] is limited to computing feature attributions only from simpler models. Beyond these analytical capabilities, COSIME is user-friendly, allowing users to train a model, compute feature relevance with a pretrained model, or combine both, while supporting flexibility in fusion strategy and output type.

We show COSIME outperforms well-known multi-omics models in integration and prediction, while also providing interpretable insights. However, there is room for further development. First, at present, COSIME computes only pairwise feature interactions; higher-order interactions may yield richer explanations but would require more advanced approaches. Second, selecting the number of hidden layers and the size of the latent space is not trivial. Although the latent dimensionality was fine-tuned, optimizing the network architecture will be explored in future work. Third, while our framework currently supports two data views, it can be extended to more views by adding separate encoders and aligning them in a shared latent space. For example, integrating transcriptomics, chromatin accessibility, and DNA methylation[61] would involve applying LOT pairwise (e.g., transcriptomics–epigenomics, epigenomics–methylation, and transcriptomics–methylation) and jointly optimizing their alignment. This also enables integration of modalities such as Hi-C[62] with suitable representations. Lastly, COSIME could also be applied to other brain-related diseases, such as neuropsychiatric disorders[63,64], when large-scale multi-omics datasets become available. It can also integrate with methods for multi-view imputation[65–67], enhancing its capacity to handle missing data and broadening its applicability.

Methods

COSIME overview

Cooperative Multiview Integration with Scalable and Interpretable Model Explainer (COSIME) is a machine learning model that integrates multi-view data for disease phenotype prediction and computes feature importance and interaction scores. By leveraging deep learning-based encoders, COSIME effectively captures the complex, multi-layered interactions between different omic modalities while preserving the unique characteristics of each data type. The integration of LOT techniques aligns and merges heterogeneous datasets, improving the accuracy of modeling across-view relationships in the joint latent space. In addition, COSIME leverages the Shapley-Taylor Interaction Index[8] to compute feature importance and interaction values, allowing for a deeper understanding of how individual features and their interactions contribute to the predictions (**Fig. 1**).

Simulated Data

We generated synthetic data by creating two distinct datasets, $\mathbf{X}_1$ and $\mathbf{X}_2$, each with a specified number of features, to simulate a scenario for multi-view analysis. Latent factors, which are unobserved variables that affect the data, were introduced with specified standard deviations (std, std_1, and std_2) and were either correlated or uncorrelated depending on the experimental design. The strength of the influence of these latent factors on the multi-view datasets was controlled by the scaling factors scale_1 and scale_2. Additionally, interaction effects between the latent factors of $\mathbf{X}_1$ and $\mathbf{X}_2$ were optionally included to introduce non-linear relationships, with the weight of these interactions controlled by the interaction_weight parameter.

Latent Factors and Dataset Generation

We generate the latent factors for the datasets $\mathbf{X}_1$ and $\mathbf{X}_2$ based on the correlation setting. If the latent factors are correlated, we use a shared latent factor matrix $\mathbf{U}$ while if the latent factors are uncorrelated, we use separate latent factor matrices $\mathbf{U}_1$ and $\mathbf{U}_2$.

Latent Factor Generation: The latent factors for both $\mathbf{X}_1$ and $\mathbf{X}_2$ are drawn from normal distributions:

$$\mathbf{U}_1 \sim \mathcal{N}(0, \sigma_1^2)_{n \times f_{\text{latent}}} \tag{1}$$

$$\mathbf{U}_2 \sim \mathcal{N}(0, \sigma_2^2)_{n \times f_{\text{latent}}} \tag{2}$$

Where:

- n is the number of samples.
- f_{latent} is the number of latent factors.
- σ_1^2, σ_2^2 are the variances of the latent factors affecting $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively.

Correlated Latent Factors: If the latent factors in the two data views are correlated, we use a shared latent factor matrix $\mathbf{U}$, and both datasets $\mathbf{X}_1$ and $\mathbf{X}_2$ are generated as:

$$\mathbf{X}_1 = \mathbf{U} \times \beta_1 + \epsilon_1 \tag{3}$$

$$\mathbf{X}_2 = \mathbf{U} \times \beta_2 + \epsilon_2 \tag{4}$$

Where:

- $\mathbf{U}$ is the shared latent factor matrix.
- β_1, β_2 are the weights for $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively.
- ϵ_1, ϵ_2 represent the noise for $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively, and follow:

$$\epsilon_1 \sim \mathcal{N}(0, \sigma_{\epsilon_1}^2) \tag{5}$$

$$\epsilon_2 \sim \mathcal{N}(0, \sigma_{\epsilon_2}^2) \tag{6}$$

Uncorrelated Latent Factors: If the latent factors in the two data views are uncorrelated, we generate separate latent factor matrices $\mathbf{U}_1$ and $\mathbf{U}_2$ for $\mathbf{X}_1$ and $\mathbf{X}_2$, and the datasets are generated as:

$$\mathbf{X}_1 = \mathbf{U}_1 \times \beta_1 + \epsilon_1 \tag{7}$$

$$\mathbf{X}_2 = \mathbf{U}_2 \times \beta_2 + \epsilon_2 \tag{8}$$

Where:

- $\mathbf{U}_1$ and $\mathbf{U}_2$ are the uncorrelated latent factor matrices for $\mathbf{X}_1$ and $\mathbf{X}_2$.
- β_1, β_2 are the weights for $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively.
- ϵ_1, ϵ_2 represent the noise for $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively, and follow:

$$\epsilon_1 \sim \mathcal{N}(0, \sigma^2_{\epsilon_1}) \tag{9}$$

$$\epsilon_2 \sim \mathcal{N}(0, \sigma^2_{\epsilon_2}) \tag{10}$$

Interaction Effects

Correlated Latent Factors: When the latent factors in the two data views are correlated, the interaction term is generated by considering pairwise combinations of the latent factors. Specifically, the interaction term $\mathbf{z}_{\text{interaction}}$ is calculated as:

$$\mathbf{z}_{\text{interaction}} = \sum_{i=f_{\text{latent}}+1}^{f_{\text{latent}}+5} \sum_{j=f_{\text{latent}}+1}^{f_{\text{latent}}+5} w U_i U_j \tag{11}$$

Where:

- U_i and U_j are the elements of the latent factor matrix $\mathbf{U}$.
- The interaction term involves the product of latent factors U_i and U_j for each pair.
- The sum is performed over the indices i and j, which range from $f_{\text{latent}} + 1$ to $f_{\text{latent}} + 5$, where the interaction is modeled between latent factors.
- w is the interaction weight, a constant that determines the strength of the interaction between latent factors.

The interaction term $\mathbf{z}_{\text{interaction}}$, which is given by equation 11, is then added to the feature matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ as follows:

$$x_{1,i} = x_{1,i} + \mathbf{z}_{\text{interaction}} \tag{12}$$

$$x_{2,j} = x_{2,j} + \mathbf{z}_{\text{interaction}} \tag{13}$$

Where equations 12 and 13 show how the interaction term $\mathbf{z}_{\text{interaction}}$ is added to the features x_1 and x_2 in the correlated case.

Outcome Variable

The outcome variables $\mathbf{y}_1$ and $\mathbf{y}_2$ are generated based on the feature matrices $\mathbf{X}_1$ and $\mathbf{X}_2$, taking into account whether interaction terms are included or not.

Correlated Latent Factors For the case where the latent factors are correlated (i.e., a shared latent factor matrix $\mathbf{U}$ is used for both $\mathbf{X}_1$ and $\mathbf{X}_2$), the outcome generation is as follows:

- **With Interaction:** When interaction terms are included, the expected outcome μ_{all} is computed as the linear combination of the latent factors $\mathbf{U}$ and the latent factor weights β, along with the pairwise interaction terms between the additional latent factors. The interaction term $\mathbf{z}_{\text{interaction}}$, given by equation 11 is used, which is defined as:

$$\mu_{\text{all}} = \mathbf{U}\beta + \mathbf{z}_{\text{interaction}} \tag{14}$$

- **Without Interaction:** When no interaction terms are included, the expected outcome is simply the linear combination of the latent factors $\mathbf{U}$ weighted by β:

$$\mu_{\text{all}} = \mathbf{U}\beta \tag{15}$$

Uncorrelated Latent Factors For the case where the latent factors are uncorrelated (i.e., separate latent factor matrices $\mathbf{U}_1$ and $\mathbf{U}_2$ are used for $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively), the outcome generation is as follows:

- The expected outcome is the sum of the linear combinations of $\mathbf{U}_1$ and $\mathbf{U}_2$ with their respective weights $\beta_{\mathbf{U}_1}$ and $\beta_{\mathbf{U}_2}$:

$$\mu_{\text{all}} = \mathbf{U}_1\beta_{\mathbf{U}_1} + \mathbf{U}_2\beta_{\mathbf{U}_2} \tag{16}$$

Outcome Variable Generation: The outcome variables $\mathbf{y}_1$ and $\mathbf{y}_2$ are generated using the expected outcomes μ_{all} computed above, along with Gaussian noise for continuous outcomes or using a logistic function for binary outcomes.

- **Continuous Outcomes:** For continuous outcomes, the outcome variables $\mathbf{y}_1$ and $\mathbf{y}_2$ are generated by adding Gaussian noise to the expected values:

$$\mathbf{y}_1 = \mu_{\text{all}} + \epsilon_1, \quad \epsilon_1 \sim \mathcal{N}(0, \sigma^2) \tag{17}$$

$$\mathbf{y}_2 = \mu_{\text{all}} + \epsilon_2, \quad \epsilon_2 \sim \mathcal{N}(0, \sigma^2) \tag{18}$$

  Where ϵ_1 and ϵ_2 are Gaussian noise terms with mean 0 and variance σ^2.

- **Binary Outcomes:** For binary outcomes, the predicted probabilities $\hat{\mathbf{y}}_1$ and $\hat{\mathbf{y}}_2$ are computed using the logistic function:

$$\hat{\mathbf{y}}_1 = \frac{1}{1 + \exp(-\mu_{\text{all}})} \tag{19}$$

$$\hat{\mathbf{y}}_2 = \frac{1}{1 + \exp(-\mu_{\text{all}})} \tag{20}$$

The outcome variables are then generated based on Bernoulli trials with added Gaussian noise to the predicted probabilities:

$$\mathbf{y}_1 \sim \text{Bernoulli}(p = \hat{\mathbf{y}}_1 + \epsilon_3), \quad \epsilon_3 \sim \mathcal{N}(0, \sigma^2) \tag{21}$$

$$\mathbf{y}_2 \sim \text{Bernoulli}(p = \hat{\mathbf{y}}_2 + \epsilon_4), \quad \epsilon_4 \sim \mathcal{N}(0, \sigma^2) \tag{22}$$

Where ϵ_3 and ϵ_4 are Gaussian noise terms with mean 0 and variance σ^2.

Experimental Conditions

We generated a total of eight experimental conditions, divided between binary and continuous outcome variables. Each condition included two distinct datasets to facilitate multi-view analysis. The conditions were based on two factors: strength of signal in the data (high vs. low) and the fusion method (early vs. late). For each dataset, we used different random seeds to introduce variability while keeping all other parameters consistent, particularly the latent factor structure **U**. In all datasets, 10 interaction terms were introduced within each view to capture the dependencies between selected features. These interactions model the relationships between certain features within the same view.

The experimental conditions were as follows:

- **Signal Strength and Fusion Type**
  - **High Signal (Early Fusion)**: Data were generated with a strong latent factor influence, including interaction effects between features. The latent factor structure **U** and interaction terms remained consistent across both datasets.
  - **High Signal (Late Fusion)**: The same strong latent factor influence was used as in the early fusion condition. However, the datasets were generated with different samples, while maintaining consistency in latent factor structure **U** and interaction effects.
  - **Low Signal (Early Fusion)**: Data were generated with a weaker latent factor influence. Interaction effects were included, and the latent factor structure **U** and interaction terms remained consistent across both datasets.
  - **Low Signal (Late Fusion)**: Data were generated with weak latent factor influence. The two datasets were generated with different samples, while maintaining consistency in latent factor structure **U** and interaction effects.
- **Outcome Type**
  - **Binary Outcomes**: Interaction effects were included, and the latent factor structure **U** and interaction terms remained consistent across datasets, based on the signal strength and fusion type.
  - **Continuous Outcomes**: Similar to binary outcomes, but the outcome variables were generated using Gaussian noise as detailed in the main text.

Parameters used to generate the datasets:

The following are the parameters used to generate the datasets, based on the notations introduced in the previous sections.

- Sample size: $n = 1000$
- Number of features: $f_{\mathbf{X}_1} = f_{\mathbf{X}_2} = 100$
- Number of latent factors: $f_{\text{latent}} = 25$ (Including five for interacting features)
- Scaling factors: $\text{scale}_1 = \text{scale}_2 = 5$ for binary high signal and continuous high signal; $\text{scale}_1 = \text{scale}_2 = 10$ for binary low signal and continuous low signal
- Latent factor standard deviations: std = 1 (shared standard deviation for latent factors affecting both views if correlated), $\text{std}_1 = 1$ (standard deviation for latent factors influencing $\mathbf{X}_1$ if uncorrelated), $\text{std}_2 = 1$ (standard deviation for latent factors influencing $\mathbf{X}_2$ if uncorrelated)
- Factor strength: latent_strength = 2 for high signal, latent_strength = 1 for low signal
- Noise standard deviation: $\sigma = 5$ for strong signal conditions, $\sigma = 10$ for weak signal conditions (binary outcomes), $\sigma = 15$ for weak signal conditions (continuous outcomes)
- Correlation between two datasets: TRUE if $\mathbf{X}_1$ and $\mathbf{X}_2$ are correlated, FALSE if they are uncorrelated
- Data type: Binary or continuous outcomes for $\mathbf{y}_1$ and $\mathbf{y}_2$
- Interaction effects: TRUE if interactions between features of $\mathbf{X}_1$ and $\mathbf{X}_2$ are included, with interaction weight $w = 10$
  - Interaction effects between the 21st and 25th features within each view ($\mathbf{X}_1$ and $\mathbf{X}_2$) were artificially introduced.

Where:

- $\mathbf{X}_1$ and $\mathbf{X}_2$ represent the feature matrices for the two datasets.
- $\mathbf{U}$ is the shared latent factor matrix when the latent factors are correlated, and $\mathbf{U}_1$ and $\mathbf{U}_2$ are the separate latent factor matrices for the uncorrelated case.
- β, $\beta_{\mathbf{U}_1}$, and $\beta_{\mathbf{U}_2}$ are the latent factor weights used to generate the expected outcome variables.
- latent_strength indicates the strength of the latent factors' influence on the outcome variables, with higher values corresponding to stronger signal conditions.
- σ denotes the standard deviation of the noise added to the outcome variables.
- w is the interaction weight, a constant that controls the influence of interaction terms between features from $\mathbf{X}_1$ and $\mathbf{X}_2$, particularly for the 21st to 25th features in each view.

Additional datasets:

To compare the computing time across datasets of different sizes, we additionally generated the following datasets.

- 5000 samples with 100 features in each view
- 5000 samples with 250 features in each view
- 10000 samples with 500 features in each view

We followed the same settings but proportionally increased the number of latent factors based on the number of features in each scenario.

Real Data

Transcriptomics-Metabolomics (ROSMAP)

The *Religious Orders Study* and *Rush Memory and Aging Project* (ROSMAP) are ongoing longitudinal clinical-pathologic cohort studies of aging and AD[68]. The ROSMAP data include a wealth of clinical, cognitive, neuroimaging, genetic, and neuropathological information from older adults, providing invaluable insights into the early stages of AD, cognitive decline, and related neurodegenerative processes.

Transcriptomics Single-cell RNA-seq (scRNA-seq) for astrocytes was downloaded from a published study[69]. From 149,558 cells and 17,817 genes, we projected 4,292 metacells and 16,718 genes for AD pathologic diagnosis using *Metacell-2*[70]. Then, we performed differential expression testing for Alzheimer's disease (AD) pathological diagnosis using *Seurat* and selected the top 280 differentially expressed genes (p-value adjusted < 0.1 and $\log_2$(fold change) > 0.2). Additionally, 25 genes that are either AD-risk or astrocyte-specific (*APOE*[12,45], *CLU*[71], *SORL1*[72], *TREM2*[72], *ABCA7*[72], *PICALM*[72], *BIN1*[48], *PLD3*[72], *CD33*[73], *CD2AP*[72], *EPHA1*[72], *INPP5D*[72], *FERMT2*[72], *CR1*[72], *PTK2B*[72], *SLC24A4*[72], *CASS4*[72], *ACE*[73], *SORCS1*[72], *GAB2*[74], *CDK5R1*[75], *PRNP*[76], *IL1RAP*[77], *CNP*[53], *TOMM40*[78]) were also selected.

Metabolomics ROSMAP metabolomic data processed by *Metabolon HD4* was downloaded from *AD knowledge Portal* - backend (syn10235592). The dataset comprises a total of 1,055 biochemicals, 971 compounds of known identity (named biochemicals) and 84 compounds of unknown structural identity (unnamed biochemicals) for 514 brain samples. We selected 305 biochemicals that do not have any missingness in their samples.

Multi-view Data For early fusion, we merged transcriptomic and metabolomic data and achieved 2,286 metacells (samples) and 305 genes and 305 metabolites in each view. Metabolites were matched to the donors for metacells in transcriptomic data. For late fusion, we used full samples for each view whose AD pathologic diagnosis is known. 4,292 metacells and 2,286 samples for transcriptomics and metabolomics, respectively were used.

Transcriptomics-Epigenomics (SEA-AD)

The *Seattle Alzheimer's Disease Brain Cell Atlas* (SEA-AD) consortium studies deep molecular and cellular understanding of the early pathogenesis of AD[79]. By integrating neuropathological data, single-cell and spatial genomics, and longitudinal clinical metadata, SEA-AD serves as a unique resource for exploring the mechanisms underlying Alzheimer's disease and related dementias.

Transcriptomics scRNA-seq data for oligodendrocytes from middle temporal gyrus (111,194 cells) was downloaded from the *Registry of Open Data on AWS* as AnnData objects (h5ad format). 3,927 metacells and 17,368 genes were projected for cognitive status (dementia). We selected expressions for the 206 transcription factors that are known to co-bind or interact each other in oligodendrocytes[16].

Epigenomics Single-cell ATAC-seq (scATAC-seq) data for all nuclei from middle temporal gyrus was downloaded from the *Registry of Open Data on AWS* as AnnData objects (h5ad format). There were 35,925 cells and 218,882 peaks for oligodendrocytes. Among those peaks, we chose 702 peaks that are overlapped with the known oligodendrocyte-specific peaks[16].

Multi-view Data We merged the scRNA-seq and scATAC-seq datasets by cell. The 35,925 cells from the scATAC-seq data were then grouped according to the metacells they belonged to in the scRNA-seq data, and the average counts for each metacell were computed.

Transcriptomics and Spatial Transcriptomics (SEA-AD)

Transcriptomics scRNA-seq data for microglia was downloaded from *CellxGene*[80] for a published study[79]. From 40,000 cells and 18,279 genes, we projected 1,009 metacells and 17,348 genes for AD progression score using *Metacell-2*. Then, we implemented differential expression testing for AD progression score using the Differential Gene Expression analysis based on the negative binomial distribution (*DESeq2*)[81] and selected 400 differentially expressed genes (p-value adjusted < 0.05 and $\log_2$(fold change) > 0.75). Additionally, 28 genes that are either AD-risk or microglia-specific were identified. (*CSF1R*[82], *TREM2*[72], *SORL1*[72], *CD68*[82], *TNF*[83], *CLEC7A*[82], *IL6*[108], *APOE*[12,45], *C1QA*[82], *C1QB*[82], *C1QC*[82], *SPP1*[82], *MAPK8*[84], *PTGS2*[85], *VEGFA*[86], *FOS*[82], *CLU*[71], *CR1*[72], *PICALM*[72], *CD33*[73], *SORCS1*[72], *GAB2*[74], *CASS4*[72], *CDK5R1*[82], *PRNPL*[76], *IL1RAP*[82], *CNP*[53], *TOMM40*[78]) were also selected.

Spatial Transcriptomics (MERFISH) Astrocyte single-cell MERFISH data for the whole taxonomy collected from the middle temporal gyrus was obtained through the *Open Data Registry on AWS* as AnnData objects (h5ad format) for a published study[79]. From 1,321,191 cells and 135 genes, 8,693 metacells and 134 genes were projected for AD progression score using *Metacell-2*. 1,016 metacells were projected for astrocytes.

Multi-view Data Among the 1,016 metacells in the MERFISH data, we randomly selected 1,009 metacells. The AD progression score was highly correlated between the scRNA-seq and MERFISH datasets (r = 0.986). For each dataset, we computed the mean AD progression score, which was highly correlated with the corresponding scores in both datasets (r = 0.997 for scRNA-seq and r = 0.996 for MERFISH).

Adjusting for confounding

Since age and sex could act as confounders, we tested whether they were associated with both the exposure and the outcome. Specifically, we examined the following associations:

$$\text{AD outcome} \sim \text{age} + \text{sex} \tag{23}$$

$$\text{Feature}_i \sim \text{age} + \text{sex} \tag{24}$$

where i indexes each feature in two data views.

In the ROSMAP dataset used for binary classification (**Fig. 3**), sex was the only variable significantly associated with the outcome. Additionally, 230 out of 305 genes and 234 out of 305 metabolites were significantly associated with sex (FDR $<$ 0.05). Therefore, we adjusted for sex by regressing out its confounding effect and used the adjusted data for prediction. All panels in **Fig. 3** have been updated to reflect the use of the adjusted input data.

In contrast, for the SEA-AD dataset used for continuous outcome prediction (**Fig. 5**), neither age nor sex showed a significant association with the outcome. Thus, they did not meet the criteria for confounding and were not adjusted for.

In the case of the multiome dataset used for binary classification (**Fig. 4**), since the samples were derived from metacells comprising cells from multiple donors, individual-level confounder adjustment was not applicable.

Model Design

Let $\mathbf{X}_1 \in \mathbb{R}^{n \times p_{\mathbf{X}_1}}$ and $\mathbf{X}_2 \in \mathbb{R}^{n \times p_{\mathbf{X}_2}}$ represent two data views, and $\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^n$ be real-valued labels that represent the same disease phenotype corresponding to $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively. COSIME inputs each data view into a deep neural network encoder (deep encoder), performs dimensionality reduction to learn K-dimensional features, and creates embeddings in an unsupervised fashion. Then, the Learnable Optimal Transport (LOT) aligns the distributions of two latent spaces in a joint latent space. We use the joint latent vector to classify or predict disease phenotypes.

Deep Neural Network Encoders (Deep encoders)

COSIME is designed to process dual-view data using separate probabilistic deep encoders for each view. Each view $x_1 \in X_1$ and $x_2 \in X_2$ is passed through its own deep encoder to learn a latent representation. These encoders approximate the posterior distributions $p(z_1|x_1)$ and $p(z_2|x_2)$, where z_1 and z_2 are the latent variables corresponding to views 1 and 2, respectively. COSIME utilizes backpropagation of Learnable Optimal Transport (LOT) to these deep neural networks, enabling the learning of latent features from multiple views, which is then used for predicting disease phenotypes.

The deep encoders output the mean (μ_i) and log-standard deviation ($\log(\sigma_i)$) for the posterior distribution of each view. The encoder for each view consists of multiple fully connected layers with LeakyReLU activations. Batch Normalization is applied to stabilize learning and accelerate convergence, while Dropout is used as a regularization technique to mitigate overfitting. The final output of each deep encoder is a latent vector z_i that represents the compressed information from the corresponding input data of a view.

To regularize the learned posterior distributions and prevent overfitting, the model incorporates the Kullback-Leibler divergence (KLD) loss during the training of the deep encoders. This

loss encourages the posterior distributions of each view to remain close to a standard Gaussian prior distribution $p(z_i) = \mathcal{N}(0, I)$, which helps improve the ability of the model to generalize and prevents overfitting. The KLD loss for each view is computed as:

$$\mathcal{L}_{\text{KLD}_1} = D_{\text{KLD}_1}(\psi_1(z_1|x_1) \parallel p(z_1)) \tag{25}$$

$$\mathcal{L}_{\text{KLD}_2} = D_{\text{KLD}_2}(\psi_2(z_2|x_2) \parallel p(z_2)) \tag{26}$$

where ψ_1 and ψ_2 are the encoder functions for views A and B, respectively, D_{KLD_A} and D_{KLD_B} are the KLD measures for the latent representations of views A and B, respectively, and $p(z_1)$ and $p(z_2)$ are the prior distributions for the latent variables z_1 and z_2 corresponding to views A and B. The KLD loss for each view i is defined as:

$$D_{\text{KLD}_i}(\psi_i(z_i|x_i) \parallel p(z_i)) = \frac{1}{2}\left(\sigma_i^2 + \mu_i^2 - \log\sigma_i^2 - 1\right) \tag{27}$$

where μ_i and σ_i are the mean and standard deviation of the posterior distribution for view i.

Learnable Optimal Transport (LOT)

COSIME utilizes Learnable Optimal Transport (LOT), a unique method designed to align the latent space distributions of two distinct data views. There are several key advantages of LOT.

First, unlike traditional optimal transport methods, Learnable Optimal Transport (LOT) treats the transport plan as a learnable parameter, optimized dynamically during training rather than relying on a fixed or pre-defined plan. This allows LOT to adapt more flexibly to complex data structures and align latent spaces in a joint latent space. By learning the transport plan iteratively, LOT can adjust its alignment strategy to the specific characteristics of the data, making it particularly effective for downstream tasks such as prediction and classification.

Second, LOT can scale effectively with large datasets. By incorporating mini-batch Sinkhorn[87] updates, LOT accelerates the optimization process, reduces memory usage, and ensures efficient computation. This makes LOT particularly well-suited for high-dimensional data from multiple sources, enabling it to handle large-scale tasks without sacrificing performance.

Third, LOT integrates domain-specific regularization through the coupling matrix, which allows the model to incorporate prior knowledge about the relationship between the views. This is particularly beneficial when working with heterogeneous data in the multi-view analysis, where domain-specific relationships can enhance the quality of the learned latent space. We apply entropy regularization to stabilize the optimization process, ensuring smooth convergence and better-converging embeddings, which are crucial for downstream tasks.

Learnable Optimal Transport (LOT) Process LOT treats the transport plan as a learnable parameter, which is iteratively updated through backpropagation. This enables LOT to adaptively align the latent spaces of different views in a shared representation space.

Both the source and target distributions in LOT are modeled as Gaussian distributions, each parameterized by their means (μ) and standard deviations (σ). The transport plan is learned by minimizing the pairwise transport cost between these distributions. The cost reflects the distance between their latent vectors, and the transport plan is updated during training to reduce this cost, aligning the distributions in a shared latent space.

The process involves the following steps:

1. **Step 1: Mini-Batching Processing**

   Before performing the computation, the source and target distributions are split into mini-batches to ensure efficient processing, especially for large datasets. Let the total number of source samples be N_{src} and the total number of target samples be N_{tgt}. In the mini-batching procedure, we split the source distribution $\{\mu_{\text{src}}, \sigma_{\text{src}}\}$ and the target distribution $\{\mu_{\text{tgt}}, \sigma_{\text{tgt}}\}$ into smaller batches of size B_{src} and B_{tgt}, respectively. Denote the i-th mini-batch of the source and target distributions as:

$$\mathcal{B}^i_{\text{src}} = \left\{\mu^i_{\text{src}}, \sigma^i_{\text{src}}\right\}, \quad \mathcal{B}^i_{\text{tgt}} = \left\{\mu^i_{\text{tgt}}, \sigma^i_{\text{tgt}}\right\} \tag{28}$$

   for $i = 1, 2, \ldots, N_{\text{src}}/B_{\text{src}}$ and $j = 1, 2, \ldots, N_{\text{tgt}}/B_{\text{tgt}}$, respectively. This mini-batch processing reduces memory consumption and allows the transport plan to be optimized iteratively.

   Each mini-batch is processed separately, and the transport plan is updated incrementally based on these smaller subsets of the source and target distributions.

2. **Step 2: Pairwise Distance Computation**

   The pairwise distances between the means μ^i_{src} and μ^j_{tgt} and the standard deviations σ^i_{src} and σ^j_{tgt} of the mini-batches are computed using the Euclidean distance:

$$\text{dist}_\mu(i, j) = \|\mu^i_{\text{src}} - \mu^j_{\text{tgt}}\|_2, \quad \text{dist}_\sigma(i, j) = \|\sigma^i_{\text{src}} - \sigma^j_{\text{tgt}}\|_2 \tag{29}$$

   where i and j refer to the samples from the source and target mini-batches, respectively.

   These pairwise distances are used to construct the cost matrix for each mini-batch, which will then be used in the subsequent steps.

3. **Step 3: Cost Matrix Construction**

   After computing the pairwise distances, the total cost matrix for each mini-batch is constructed as follows:

$$\text{cost}^i_{total}(i, j) = \text{dist}_\mu(i, j) + \text{dist}_\sigma(i, j) + \epsilon \tag{30}$$

   where ϵ is a small constant for numerical stability.

4. **Step 4: Sinkhorn Algorithm (Learnable Transport Plan)**

   The transport matrix $\mathbf{P}^i$ for each mini-batch is computed iteratively using the Sinkhorn algorithm. The transport plan is initialized as a learnable parameter, $\mathbf{P}^i$, where each element of $\mathbf{P}^i$ represents the transport probability between a sample from the source and a sample from the target. The transport plan is updated during each iteration to minimize the total transportation cost.

   The update rule for the transport plan $\mathbf{P}^i(i, j)$ for each mini-batch is:

$$\mathbf{P}^i(i, j) = \frac{\exp\left(-\frac{\text{cost}^i_{total}(i,j)}{\text{reg}}\right)}{Z^i_{\text{src}} \cdot Z^i_{\text{tgt}}} \tag{31}$$

where Z^i_{src} and Z^i_{tgt} are the normalizing factors for the source and target distributions, respectively:

$$Z^i_{\text{src}} = \sum_j \exp\left(-\frac{\text{cost}^i_{total}(i,j)}{\text{reg}}\right), \quad Z^i_{\text{tgt}} = \sum_i \exp\left(-\frac{\text{cost}^i_{total}(i,j)}{\text{reg}}\right) \tag{32}$$

Here, the learnable transport plan allows gradients to flow through the transport matrix, enabling the optimization process using backpropagation.

Handling Different Sample Sizes The Sinkhorn algorithm can effectively handle source and target distributions with differing sample sizes, making it particularly useful for COSIME late fusion. To ensure that transport plans remain valid, we apply normalization to the weights of the source and target distributions during each mini-batch update as follows:

$$\mathbf{A}(i) = \frac{\mathbf{A}(i)}{\sum_i \mathbf{A}(i)}, \quad \mathbf{B}(j) = \frac{\mathbf{B}(j)}{\sum_j \mathbf{B}(j)} \tag{33}$$

This adjustment ensures that the transport plan remains meaningful even when the source and target distributions contain different numbers of samples.

The Sinkhorn algorithm iteratively refines the transport plan $\mathbf{P}^i$ until convergence or until a predefined number of iterations.

5. **Step 5: Dual Variables and Marginal Constraints**

   Dual variables $\mathbf{u}^i$ and $\mathbf{v}^i$ are introduced to enforce the marginal constraints of the transport problem. These variables ensure that the transport plan satisfies the required marginal distributions for the source and target distributions. The dual variables are updated iteratively for each mini-batch:

$$\mathbf{u}^i(i) = \frac{\mathbf{A}(i)}{Z^i_{\text{src}}} \cdot \exp\left(-\frac{\text{cost}^i_{total}(i,j)}{\text{reg}}\right) \tag{34}$$

$$\mathbf{v}^i(j) = \frac{\mathbf{B}(j)}{Z^i_{\text{tgt}}} \cdot \exp\left(-\frac{\text{cost}^i_{total}(i,j)}{\text{reg}}\right) \tag{35}$$

   where **reg** denotes the entropic regularization parameter in Sinkhorn's algorithm, and the dual variables enforce that the transport plan aligns with the marginal distributions of the source and target.

6. **Step 6: Optimizing the Transport Plan**

   During each mini-batch, the transport plan is optimized by performing backpropagation to minimize the total loss. This involves computing the gradients of the transport plan with respect to the loss function and updating the transport plan parameters via gradient descent.

7. **Step 7: Final LOT Loss Calculation**

   After optimizing the transport plan, the final LOT loss is computed as the sum of the product of the cost matrix and the transport plan:

$$\mathcal{L}_{\text{LOT}} = \sum_{i,j} \text{cost}_{total}(i, j) \cdot \mathbf{P}(i, j) \tag{36}$$

   This loss function quantifies the minimum cost required to transform the source distribution into the target distribution, thus aligning the two views in a joint latent space.

Phenotype Prediction/Classification

COSIME employs the aligned latent representations from each view, obtained via separate probabilistic deep encoders. After the deep encoders generate the latent vectors for each view, LOT aligns these embeddings into a joint latent space. The model then uses these aligned embeddings for phenotype prediction.

Two fusion strategies are employed to combine the embeddings from the two views: early fusion and late fusion. Both fusion strategies are applied for binary outcome classification and continuous outcome prediction tasks.

Early Fusion The fused representation is obtained by taking the mean of the representations $z_1 \in \mathbb{R}^d$ and $z_2 \in \mathbb{R}^d$ from each view:

$$z_{\text{fusion}} = \frac{z_1 + z_2}{2} \tag{37}$$

Late Fusion The latent representations $z_1 \in \mathbb{R}^d$ and $z_2 \in \mathbb{R}^d$ from each view are vertically concatenated to form the fused representation z_{fusion}:

$$z_{\text{fusion}} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \tag{38}$$

where $z_{\text{fusion}} \in \mathbb{R}^{2d}$ is the fused vector obtained by stacking z_1 and z_2 vertically.

Binary Outcome Prediction The fused representation z_{fusion} is then passed through a linear layer followed by a sigmoid activation function to compute the predicted probability $\hat{p}$:

$$\hat{p} = \sigma(W_{\text{binary}} \cdot z_{\text{fusion}} + b) \tag{39}$$

where $\sigma(\cdot)$ is the sigmoid function, and W_{binary} and b are the learned weights and bias parameters of the prediction layer.

Loss function

The **BCEWithLogitsLoss** is computed for the early fusion setup, which combines the sigmoid activation and binary cross-entropy in a single, more numerically stable operation. This loss function is defined as:

$$\mathcal{L}_{\text{BCE}}^{\text{early}} = -\frac{1}{n} \sum_{i=1}^{n} [y_i \log(\sigma(\hat{p}_i)) + (1 - y_i) \log(1 - \sigma(\hat{p}_i))] \tag{40}$$

where $\hat{p}_i$ is the predicted probability of the i-th sample belonging to the positive class (i.e., the probability output by the model), and y_i is the true binary label for the i-th sample, where $y_i \in \{0, 1\}$.

Continuous Phenotype Prediction For continuous phenotype prediction, the fused representation z_{fusion} is passed through a linear transformation:

$$\hat{y} = W_{\text{reg}} \cdot z_{\text{fusion}} + b \tag{41}$$

where W_{reg} and b are the learned weights and bias of the regression layer.

Loss for Regression The Mean Squared Error (MSE) loss for continuous outcome prediction using a linear model is computed as:

$$\mathcal{L}_{\text{MSE}}^{\text{early}} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{42}$$

where $\hat{y}_i$ is the predicted continuous value for the i-th sample, representing the model's prediction, and y_i is the true continuous value for the i-th sample, which is the actual target value the model aims to predict. The loss measures the squared difference between the predicted and true values for each sample and averages this over all N samples.

Loss for Neural Network A neural network (NN) model can be used for continuous outcome prediction, the output $\hat{y}_i$ is predicted by the neural network, and the loss function remains MSE:

$$\mathcal{L}_{\text{MSE}}^{\text{NN, early}} = \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \hat{y}_i^{\text{NN}}\right)^2 \tag{43}$$

where $\hat{y}_i^{\text{NN}}$ is the continuous value predicted by the neural network for the i-th sample, which is the output of the neural network model for that particular input, and y_i is the true continuous value for the i-th sample, which is the actual target value that the model is attempting to predict. The loss measures the squared difference between the predicted value and the true value for each sample, and the average squared error is computed over all N samples.

Prediction Loss The prediction loss term $\mathcal{L}_{\text{pred}}$ is defined based on the task at hand:

$$\mathcal{L}_{\text{pred}} = \begin{cases} \mathcal{L}_{\text{BCE}}^{\text{early}}, & \text{for binary outcome classification (early fusion)} \\ \mathcal{L}_{\text{MSE}}^{\text{early}}, & \text{for continuous outcome prediction (early fusion, regression)} \\ \mathcal{L}_{\text{MSE}}^{\text{NN, early}}, & \text{for continuous outcome prediction (early fusion, NN)} \\ \mathcal{L}_{\text{BCE}}^{\text{late}}, & \text{for binary outcome classification (late fusion)} \\ \mathcal{L}_{\text{MSE}}^{\text{late}}, & \text{for continuous outcome prediction (late fusion, regression)} \\ \mathcal{L}_{\text{MSE}}^{\text{NN, late}}, & \text{for continuous outcome prediction (late fusion, NN)} \end{cases}$$

Total Weighted Loss

During the training phase, the contributions of each loss term can vary in magnitude, depending on the relative scale of the individual losses. To balance the influence of each term on the total

loss, we normalize the loss terms dynamically. This dynamic normalization ensures that no individual loss term has an outsized influence on the total loss, allowing for a more balanced optimization process.

We first compute the inverse of each loss term:

$$\text{Inv}_{\mathcal{L}_{\text{KLD}_{\mathcal{A}}}} = \frac{1}{\mathcal{L}_{\text{KLD}_A}(\theta) + 1e^{-8}} \tag{44}$$

$$\text{Inv}_{\mathcal{L}_{\text{KLD}_{\mathcal{B}}}} = \frac{1}{\mathcal{L}_{\text{KLD}_B}(\theta) + 1e^{-8}} \tag{45}$$

$$\text{Inv}_{\mathcal{L}_{\text{LOT}}} = \frac{1}{\mathcal{L}_{\text{LOT}}(\theta) + 1e^{-8}} \tag{46}$$

$$\text{Inv}_{\mathcal{L}_{\text{pred}}} = \frac{1}{\mathcal{L}_{\text{pred}}(\theta) + 1e^{-8}} \tag{47}$$

Next, we compute the total inverse magnitude:

$$\text{Total_Inverse} = \text{Inv}_{\mathcal{L}_{\text{KLD}_{\mathcal{A}}}} + \text{Inv}_{\mathcal{L}_{\text{KLD}_{\mathcal{B}}}} + \text{Inv}_{\mathcal{L}_{\text{LOT}}} + \text{Inv}_{\mathcal{L}_{\text{pred}}} \tag{48}$$

Now, we normalize the inverse magnitudes to get their relative adjusted weights:

$$\text{Norm}_{\mathcal{L}_{\text{KLD}_{\mathcal{A}}}} = \frac{\text{Inv}_{\mathcal{L}_{\text{KLD}_{\mathcal{A}}}}}{\text{Total_Inverse}} \tag{49}$$

$$\text{Norm}_{\mathcal{L}_{\text{KLD}_{\mathcal{B}}}} = \frac{\text{Inv}_{\mathcal{L}_{\text{KLD}_{\mathcal{B}}}}}{\text{Total_Inverse}} \tag{50}$$

$$\text{Norm}_{\mathcal{L}_{\text{LOT}}} = \frac{\text{Inv}_{\mathcal{L}_{\text{LOT}}}}{\text{Total_Inverse}} \tag{51}$$

$$\text{Norm}_{\mathcal{L}_{\text{pred}}} = \frac{\text{Inv}_{\mathcal{L}_{\text{pred}}}}{\text{Total_Inverse}} \tag{52}$$

Finally, the total weighted loss $\mathcal{L}_{\text{Total_weighted}}(\theta)$ is computed as:

$$\begin{aligned}\mathcal{L}_{\text{Total_weighted}}(\theta) = & KLD_{A_{\text{weight}}} \cdot \text{Norm}_{\mathcal{L}_{\text{KLD}_{\mathcal{A}}}} \cdot \mathcal{L}_{\text{KLD}_A}(\theta) + \\ & KLD_{B_{\text{weight}}} \cdot \text{Norm}_{\mathcal{L}_{\text{KLD}_{\mathcal{B}}}} \cdot \mathcal{L}_{\text{KLD}_B}(\theta) + \\ & OT_{\text{weight}} \cdot \text{Norm}_{\mathcal{L}_{\text{LOT}}} \cdot \mathcal{L}_{\text{LOT}}(\theta) + \\ & CL_{\text{weight}} \cdot \text{Norm}_{\mathcal{L}_{\text{pred}}} \cdot \mathcal{L}_{\text{pred}}(\theta)\end{aligned} \tag{53}$$

where λ_{KLD_A}, λ_{KLD_B}, λ_{LOT}, and λ_{pred} are the weights that control the importance of each normalized loss term, specifically $\text{Norm}_{\mathcal{L}_{\text{KLD}_{\mathcal{A}}}}$, $\text{Norm}_{\mathcal{L}_{\text{KLD}_{\mathcal{B}}}}$, $\text{Norm}_{\mathcal{L}_{\text{LOT}}}$, and $\text{Norm}_{\mathcal{L}_{\text{pred}}}$, respectively. These normalized loss terms are the contributions of each individual loss term $\mathcal{L}_{\text{KLD}_A}(\theta)$, $\mathcal{L}_{\text{KLD}_B}(\theta)$, $\mathcal{L}_{\text{LOT}}(\theta)$, and $\mathcal{L}_{\text{pred}}(\theta)$ to the total loss, which are normalized to ensure that no single loss dominates the optimization process.

To optimize the model parameters, we minimize the total weighted loss:

$$\theta^* = \underset{\theta}{\text{argmin}}\, \mathcal{L}_{\text{Total_weighted}}(\theta) \tag{54}$$

This method prevents overemphasis on any one loss component, allowing the model to effectively balance view alignments and prediction accuracy. It enables flexible adjustment of task importance, which is crucial when working with multi-view data, where different objectives (e.g., alignment, prediction) may compete for optimization.

Model training process

The input data for each view was split into 75% for training and 25% for testing (holdout). 5-fold cross-validation was applied to the training set. The best-performing model was selected based on the minimum classification (or prediction) loss during the training phase and was then evaluated on the testing set. The **Algorithm 1: COSIME Model Training** is detailed in the **Supplementary Information**.

The prediction performance of the models was evaluated using holdout data with metrics such as Area Under the Receiver Operating Characteristic Curve (AUROC), Area Under the Precision-Recall Curve (AUPRC), and accuracy for binary outcome classification, and Mean Squared Error (MSE) for continuous outcome prediction. The performance metrics were compared across different methods using the mean and ±1.96 times the standard deviation. For the best-trained models across the different multi-view datasets, we computed feature importance for each data view and interaction values both within and across views.

Hyperparameter tuning

We tuned several hyperparameters such as the learning rate, learning rate decay (gamma), loss weights, dropout rate, joint latent space dimensionality, early stopping patience, minimum change for early stopping, and weight decay in the optimizer. Hyperparameter optimization was performed using *Ray Tune*[88] with Distributed Asynchronous Hyperparameter Optimization (*Hyperopt*)[89]. The Adam optimizer was employed to minimize the total loss, with a maximum of 300 epochs. Early stopping was used to prevent overfitting, based on performance improvements on the validation set in the training phase.

Optimization and Loss Minimization

During the training phase, the model updates the hyper-parameters by minimizing the total loss. The optimization process involves computing gradients with respect to the total loss function and updating the parameters using gradient descent or its variants.

By minimizing the total loss, COSIME simultaneously learns to:

- Regularize the posterior distributions of the latent representations for each view through the KLD losses,
- Align the embeddings from both views into a shared latent space using LOT,
- Predict or classify the disease phenotype by minimizing classification (or prediction) loss.

Evaluating the Performance of the Model Predictions

To evaluate the performance of the best model selected in the training phase for both binary outcome classification and continuous outcome prediction, several standard metrics are employed:

Binary Outcome Classification

1. **Area Under the Receiver Operating Characteristic Curve (AUROC):** AUROC measures the ability of the model to discriminate between the positive and negative classes. It is calculated by assessing the True Positive Rate (TPR) against the False Positive Rate

(FPR) at various threshold settings. The AUROC value ranges from 0 to 1, where 1 indicates perfect classification, and 0.5 is the expected performance of a random classifier. The formula for AUROC is:

$$\text{AUROC} = \int_0^1 \text{TPR}(f)\, d\text{FPR}(f) \tag{55}$$

where $\text{TPR}(f)$ and $\text{FPR}(f)$ are the True Positive Rate and False Positive Rate at each decision threshold f.

2. **Area Under the Precision-Recall Curve (AUPRC):** AUPRC is another useful metric for evaluating binary classification performance, particularly when dealing with imbalanced datasets. It measures the area under the curve created by plotting precision against recall at various threshold settings. The formula for AUPRC is:

$$\text{AUPRC} = \int_0^1 \text{Precision}(r)\, d\text{Recall}(r) \tag{56}$$

where $\text{Precision}(r)$ and $\text{Recall}(r)$ are the precision and recall at each decision threshold r.

3. **Accuracy:** Accuracy is the proportion of correctly classified instances (both true positives and true negatives) out of the total number of instances. It is calculated as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{57}$$

where TP is True Positives, TN is True Negatives, FP is False Positives, and FN is False Negatives.

Continuous Outcome Prediction

Mean Squared Error (MSE): MSE measures the average squared difference between the true values and the predicted values. It is a common loss function for regression tasks and is used to quantify the error in continuous prediction tasks. The formula for MSE is:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{58}$$

where y_i is the true continuous label for the i-th sample, $\hat{y}_i$ is the predicted value for the i-th sample, and n is the total number of samples.

These evaluation metrics assess the effectiveness of the COSIME model in both binary and continuous phenotype prediction tasks. Higher AUROC and AUPRC values, along with higher accuracy and lower MSE, indicate better model performance.

FI and feature interaction

COSIME introduces an advanced approach for quantifying both FI and feature interactions in predictive modeling, building upon the Shapley-Taylor framework[8]. By using Monte Carlo sampling and a batching process, COSIME efficiently approximates Shapley values and Shapley-Taylor indices, making it highly scalable and interpretable for large, high-dimensional datasets while maintaining memory efficiency. This approach ensures COSIME is well-suited for complex real-world applications.

FI is simply the difference between a model's prediction when using all features and its prediction when one specific feature is left out. Feature interaction refers to the additional effect on the prediction when two features are used together, beyond the sum of their individual effects. Note that pairwise feature interaction is different from traditional correlation between two features, which describes their relationship in the input data but does not reflect their influence on the model's prediction.

COSIME employs Monte Carlo sampling to efficiently approximate both Shapley values and feature interactions. This approach notably enhances its scalability, interpretability, and flexibility, especially when handling large datasets and high-dimensional feature spaces. Instead of relying on exact calculations, which become computationally expensive as the number of features or data points increases, Monte Carlo sampling allows COSIME to estimate Shapley values and Shapley-Taylor indices efficiently. This method also offers precise and interpretable calculations of pairwise feature interactions by directly simulating feature combinations. By estimating these interactions in this way, COSIME avoids complex approximations or indirect assumptions, providing clearer insights into how combinations of features jointly affect predictions. Furthermore, COSIME is better equipped to handle complex feature interactions and nonlinear relationships more effectively.

Additionally, COSIME is memory-efficient due to its batching process, which breaks the data into smaller subsets to minimize memory usage. This enables COSIME to handle large datasets on systems with limited memory, making it suitable for real-world applications with resource constraints. The batching process ensures that COSIME can scale efficiently while maintaining performance across high-dimensional feature spaces.

COSIME can also calculate signed feature interactions, which indicate whether pairs of features interact synergistically (positive interaction) or antagonistically (negative interaction). This is achieved by directly simulating feature combinations during the Monte Carlo sampling process, allowing COSIME to estimate both the magnitude and direction of interactions. The sign of an interaction is determined by comparing the joint effect of the features to their individual effects, revealing whether their combined influence on the prediction. This enhancement adds a important layer of interpretability, enabling users to understand not only the strength but also the directionality of feature interactions.

Finally, COSIME allows users to customize the Monte Carlo sample size and maximum memory usage, providing flexibility in the computational process. When the memory usage is specified, COSIME automatically determines the optimal batch size, making the method both user-friendly and adaptable to different system configurations. The details of **Algorithm 2: COSIME Feature Importance and Interaction Computation** are provided in the **Supplementary Information**.

Shapley Value (Feature Importance Computation)

The Shapley value ϕ_i for a given feature i is computed by considering the marginal contribution of feature i across all possible combinations of other features. This is approximated using Monte

Carlo sampling over multiple iterations. The Shapley value for each feature i and each sample s is computed as:

$$\phi_p(s) = \frac{1}{M} \sum_{m=1}^{M} \left(f(\mathbf{x}_s^{(p-1)}) - f(\mathbf{x}_s^{(p)}) \right) \tag{59}$$

where:

- $f(\mathbf{x}_s)$ represents the prediction using all features,
- $f(\mathbf{x}_s^{(p)})$ represents the prediction when feature p is masked,
- $f(\mathbf{x}_s^{(p-1)})$ represents the prediction when all features except for feature p are masked.

The Shapley values for all features are stored in a matrix $\mathcal{S} \in \mathbb{R}^{N \times F}$, where N is the number of samples and F is the number of features.

$$\mathcal{S}_{ij} = \phi_j(s_i) \tag{60}$$

where i indexes the i-th sample ($i \in \{1, 2, \ldots, N\}$) and j indexes the j-th feature ($j \in \{1, 2, \ldots, F\}$), and $\phi_j(s_i)$ represents the Shapley value for feature j with respect to sample i.

Feature Interaction Effects

The interaction effect between two features p and q quantifies the joint contribution of features p and q compared to when they are individually absent. The interaction effect $\mathcal{I}_{pq}$ is computed as:

$$\mathcal{I}_{pq}(s) = f(\mathbf{x}_s) - f(\mathbf{x}_s^{(p)}) - f(\mathbf{x}_s^{(q)}) + f(\mathbf{x}_s^{(pq)}) \tag{61}$$

where:

- $f(\mathbf{x}_s)$: the prediction of the model using all features,
- $f(\mathbf{x}_s^{(p)})$ and $f(\mathbf{x}_s^{(q)})$: the predictions of the model with features p and q individually masked,
- $f(\mathbf{x}_s^{(pq)})$: the prediction of the model when both features p and q are masked.

For pairwise interactions, the interaction effect is averaged over M Monte Carlo iterations:

$$\mathcal{I}_{ij}(s) = \frac{1}{M} \sum_{m=1}^{M} \left[f(\mathbf{x}_s) - f(\mathbf{x}_s^{(i)}) - f(\mathbf{x}_s^{(j)}) + f(\mathbf{x}_s^{(ij)}) \right] \tag{62}$$

The interaction effects for all pairs of features are stored in the interaction matrix $\mathcal{I} \in \mathbb{R}^{F \times F}$, where each entry $\mathcal{I}_{ij}$ represents the interaction between features i and j.

$$\mathcal{I}_{ij} = \mathcal{I}_{ji} \tag{63}$$

Monte Carlo Approach and Interaction Algorithm

Given the model f, input data $X \in \mathbb{R}^{N \times F}$ (where N is the number of samples and F is the number of features), and target labels $y \in \mathbb{R}^N$, we compute the Shapley values and interaction effects as follows:

- **Shapley Values**: For each feature i, compute the marginal contribution to each sample over M Monte Carlo iterations. Store the results in the matrix $\mathcal{S}$.
- **Interaction Effects**: For each pair of features (i, j), compute the interaction effect using the formula above, averaging over M Monte Carlo iterations. Store the results in the interaction matrix $\mathcal{I}$.

The algorithm processes the data in batches to manage memory efficiently. If the batch size is not provided, it is dynamically computed based on the available memory. The number of batches B is given by:

$$B = \left\lceil \frac{N}{\text{batch size}} \right\rceil \tag{64}$$

The batch processing ensures that we do not exceed the maximum memory usage during computation. Considering the available resources and estimated computing time, we used 100 Monte Carlo iterations for the simulated multi-view data and 10 iterations for the real multi-view data. For each multi-view dataset, the test (holdout) data was used.

Memory Management

To ensure that the memory usage remains within a specified limit, we compute the memory required for each batch as:

$$\text{Batch memory (GB)} = \frac{\text{batch size} \times \text{element size of X} \times F}{1 \times 10^9} \tag{65}$$

If the calculated batch memory exceeds the available memory, the batch size is adjusted accordingly. The dynamic batch size calculation ensures that the computation does not exceed the memory limit.

Output

The Shapley values are stored in a matrix $\mathcal{S}$ with dimensions $N \times F$, and the interaction effects are stored in a matrix $\mathcal{I}$ with dimensions $F \times F$. These matrices represent the individual contributions of each feature and their pairwise interactions in the prediction process.

In summary, our innovative approach provides a comprehensive methodology for assessing both individual feature importance (Shapley values) and pairwise feature interactions in machine learning models. By leveraging these tools, we enable a deeper understanding of how features individually and collectively influence model predictions. The computational framework we present, along with the accompanying code, facilitates the application of this methodology to various prediction tasks, enhancing model interpretability and providing valuable insights into feature relationships.

Downstream analyses

Gene Enrichment Analysis

The function `enrichGO` from the `clusterProfiler`[90] R package, a universal enrichment tool for interpreting omics data, was used to perform enrichment analyses for the 20 prioritized genes by absolute feature importance (FI) values in each view of real multi-view datasets. The top 10 categories were displayed by $\log_{10}$(FDR) (**Fig. 3b** and **Supplementary Fig. 1**).

Metabolite Enrichment Analysis

`MetaboAnalyst 6.0`[91] was used to implement a metabolite set enrichment analysis for 20 prioritized metabolites by absolute feature FI values in the metabolomic data for classifying cognitive diagnosis for Alzheimer's disease (**Fig. 3c**).

Interaction Network

For the top 50 across-view interaction values, the edges and nodes for the interacting genes (and genes and metabolites) were formatted in R, and the network was generated using `Cytoscape 3.10.3`[92] (**Fig. 3e** and **Fig. 5d**).

Spatial mapping and statistical tests

Spatial expression plots for *RYR3* and *DGKG* were generated in Python, employing `matplotlib` and `seaborn` (**Fig. 5c**). To assess whether there are significant spatial differences between cells with high FI values and low feature importance values for each gene, the following steps were taken:

- Compute the midpoint of the bounding box: It is calculated by finding the midpoint of the bounding box defined by the x and y coordinates. Specifically, the center is determined by averaging the maximum and minimum values of both the x and y coordinates:

$$\text{center}_x = \frac{\max(x) + \min(x)}{2}, \quad \text{center}_y = \frac{\max(y) + \min(y)}{2} \tag{66}$$

  This central point serves as the reference from which the distances of each point are measured.

- Compute the Euclidean distance from each point to the midpoint of the bounding box: Once the midpoint of the bounding box is defined, the Euclidean distance from each point (x_i, y_i) to the computed midpoint of the bounding box $(\text{center}_x, \text{center}_y)$ is calculated using the formula:

$$\text{distance}_i = \sqrt{(x_i - \text{center}_x)^2 + (y_i - \text{center}_y)^2} \tag{67}$$

  This step computes a scalar value for each point that quantifies how far each point is from the center.

- Perform a two-sample two-tailed t-test: The dataset is divided into two groups based on the feature importance:

- Group 1: Points with FI values greater than or equal to the 75th percentile (i.e., the top 25%).
- Group 2: Points with FI values below the 75th percentile.

We performed a two-sample two-tailed t-test to evaluate whether the distances from the midpoint of the bounding box differed between the top 25% of cells with the highest feature importance values and the remaining 75% of cells. The null hypothesis (H_0) for the two-tailed t-test is:

$$H_0 : \mu_{\text{Group1}_{\text{FI}}} = \mu_{\text{Group2}_{\text{FI}}}$$

where $\mu_{\text{Group1}_{\text{FI}}}$ and $\mu_{\text{Group2}_{\text{FI}}}$ represent the means of the distances for the high and low FI groups, respectively. The alternative hypothesis is:

$$H_1 : \mu_{\text{Group1}_{\text{FI}}} \neq \mu_{\text{Group2}_{\text{FI}}}$$

The test statistic t and the associated p–value are calculated to determine whether the difference in means is statistically significant.

Similarly, the spatial distribution of across-view interaction values was visualized using the same tools (**Fig. 5e**). The following steps were taken to generate the spatial distribution of across-view interaction values and assess whether there are significant spatial differences between cells with positive feature interaction values and negative feature interaction values, the subsequent steps were carried out:

- Distance calculation: The distance of each cell from the center of the tissue sample was calculated using its spatial coordinates (X, Y). This distance quantifies how far each cell is from the central point of the tissue section. The Euclidean distance from the center is computed as:

$$\text{Distance to center} = \sqrt{(X - X_{\text{center}})^2 + (Y - Y_{\text{center}})^2} \tag{68}$$

  where X_{center} and Y_{center} are the coordinates of the center of the tissue.

- Cell-level interaction score: For each cell, the interaction score was computed using the MERFISH gene expression levels and pairwise across-view interaction values. Let $\mathbf{E}^{(i)}_{\text{MERFISH}}$ denote the gene expression vector for cell i, which is of length G_{MERFISH}. The interaction matrix $\mathbf{M}$ has dimensions $G_{\text{MERFISH}} \times G_{\text{scRNA-seq}}$, where each element M_{g_1,g_2} represents the pairwise interaction between MERFISH gene g_1 and scRNA-seq gene g_2. The cell-level interaction score $S^{(i)}$ for each cell i was calculated as the dot product of the gene expression vector for that cell and the interaction matrix:

$$S^{(i)} = \mathbf{E}^{(i)}_{\text{MERFISH}} \cdot \mathbf{M} \tag{69}$$

  In expanded form, the score for cell i with respect to scRNA-seq gene j is:

$$S^{(i)}_j = \sum_{g_1=1}^{G_{\text{MERFISH}}} \mathbf{E}^{(i)}_{\text{MERFISH}}[g_1] \cdot M_{g_1,j} \tag{70}$$

  This score quantifies the interaction between the MERFISH gene expression profile of cell i and the scRNA-seq gene expression profile, enabling grouping of cells based on

their interaction scores. Cells were then grouped based on the sign of their interaction scores: those with positive scores were assigned to the Positive group, while those with negative scores were assigned to the Negative group. This grouping allowed for further comparison of spatial distributions and other characteristics between the two groups.

- Perform a two-sample two-tailed t-test: To determine whether there were significant differences in the distances of cells from the center between the two groups, an independent two-sample t-test was performed. The hypotheses tested were:
  - Null hypothesis (H_0): The mean distance from the center of the Positive and Negative groups are equal, i.e.,
  $$H_0 : \mu_{\text{Positive}} = \mu_{\text{Negative}}$$
  - Alternative hypothesis (H_1): The mean distance from the center of the Positive and Negative groups are not equal, i.e.,
  $$H_1 : \mu_{\text{Positive}} \neq \mu_{\text{Negative}}$$
- Perform a two-tailed F-test: In addition to comparing the means, an F-test was performed to compare the variances of the two groups. The hypotheses tested were:
  - Null hypothesis (H_0): The variances of the Positive and Negative groups are equal, i.e.,
  $$H_0 : \sigma^2_{\text{Positive}} = \sigma^2_{\text{Negative}}$$
  - Alternative hypothesis (H_1): The variances of the Positive and Negative groups are not equal, i.e.,
  $$H_1 : \sigma^2_{\text{Positive}} \neq \sigma^2_{\text{Negative}}$$

Benchmarking methods

Cooperative Learning

Cooperative learning is a framework for fitting machine learning models to multi-view data[3]. It presents a loss function that minimizes the prediction error across all views, while encouraging the predictions from the views to agree. This framework with linear regression, which is what we use for benchmarking.

We implemented the Cooperative Learning framework using the `multiview` R package. Following the examples provided on the Cooperative Learning GitHub repository, we preprocessed the input data using the *preProcess* function to center and scale it. For model evaluation, we conducted 5-fold cross-validation using the `cv.multiview` function, with a maximum of 300 iterations. For binary outcome classification, we set `family = binomial()` and `type.measure = "class"`. For continuous outcome prediction, we used the default `family = gaussian()` and `type.measure = "mse"`. The hyperparameter values for the regularization parameter were chosen from the set $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$, while the lambda values were left at their default settings.

Data Integration Analysis for Biomarker discovery using Latent components (DIABLO)

Data Integration Analysis for Biomarker discovery using Latent components (DIABLO) is a supervised extension of canonical correlation analysis[4]. It aims to identify linear combinations of variables from each dataset that maximize the correlation between datasets.

We implemented DIABLO using the `mixOmics` R package. Following the tutorial on the mixOmics website, we created a data block using the matrix function. For model training, we performed 5-fold cross-validation using the `block.splsda` function. The number of variables was tested within the set $\{10, 20, 30, 40, 50\}$.

Multi-Omics Factor Analysis v2 (MOFA+)

Multi-Omics Factor Analysis v2 (MOFA+) is an unsupervised matrix decomposition method that can be viewed as a generalization of sparse principal components analysis.

We used the `MOFA2` R package to obtain latent representations. Following the tutorial on the MOFA+ downstream analysis in R, we trained a MOFA model using the `run_mofa` function. To project unseen data from a new view into the latent space Z_i, we computed the pseudo-inverse of the weight matrix W_i. The latent representations Z_i for all views were then concatenated. For prediction, we applied logistic regression for binary outcomes and linear regression for continuous outcomes. To classify and predict new, unseen samples, we used the concatenated latent representation matrix Z in PyTorch (Python 3.10). We evaluated different latent factor dimensions from the set $\{1, 5, 10, 15\}$.

cha

Aligned Optimal Transport

Aligned optimal transport (Aligned OT)[93] serves as a baseline OT loss, where the OT matrix is assumed to be diagonal. Therefore, the final cost is the sum of the diagonal elements of the cost matrix, which we use in place of the LOT loss for our benchmark. Aligned OT thus captures only the alignment between the two latent spaces.

To compute this loss, we used the Aligned OT implementation from the Python Optimal Transport (POT)[93] library. For variability analysis, we quantified the dispersion of cost values by computing the standard deviation of the cost matrices in both source and target modalities relative to their means.

Single-cell Multi-omic Integration with Optimal Transport

The Single-cell Multi-omic Integration with Optimal Transport (SCOT)[94] OT loss is more complex leverages Gromov-Wasserstein transport to capture structural relationships between the source and target cost matrices. The resulting transport plan is combined with the cost matrices through element-wise multiplication and summation to yield the final OT loss. This approach uses Euclidean distances to define costs and seeks to minimize distances between matched representations across modalities.

We implemented the OT loss computation by referring to SCOT methods. To assess variability in transport costs, we again measured the standard deviation of the cost matrices with respect to their means in each modality.

Data Availability

Simulated data and all data supporting the results are included in the Supplementary Data 1–9 and are publicly available on GitHub (https://github.com/daifengwanglab/COSIME). ROSMAP scRNA-seq was sourced from a website (compbio.mit.edu/ad_aging_brain). Metabolomic data can be downloaded from the AD Knowledge Portal - backend (syn10235592). SEA-AD scATAC-seq, scRNA-seq, and MERFISH data were obtained from SEA-AD: Seattle Alzheimer's Disease Brain Cell Atlas (cellxgene.cziscience.com/collections/...).

Code Availability

The entire COSIME code is implemented in Python and available via Zenodo at https://doi.org/10.5281/zenodo.15428769[95]. The COSIME Python package and examples can be accessed publicly at https://github.com/daifengwanglab/COSIME.

References

1. Hasin, Y., Seldin, M. & Lusis, A. Multi-omics approaches to disease. en. *Genome Biology* **18,** 83. ISSN: 1474-760X. `https://genomebiology.biomedcentral.com/articles/10.1186/s13059-017-1215-1` (2024) (Dec. 2017).
2. Subramanian, I., Verma, S., Kumar, S., Jere, A. & Anamika, K. Multi-omics Data Integration, Interpretation, and Its Application. eng. *Bioinformatics and Biology Insights* **14,** 1177932219899051. ISSN: 1177-9322 (2020).
3. Ding, D. Y., Li, S., Narasimhan, B. & Tibshirani, R. Cooperative learning for multiview analysis. en. *Proceedings of the National Academy of Sciences* **119,** e2202113119. ISSN: 0027-8424, 1091-6490. `https://pnas.org/doi/full/10.1073/pnas.2202113119` (2024) (Sept. 2022).
4. Singh, A. *et al.* DIABLO: an integrative approach for identifying key molecular drivers from multi-omics assays. en. *Bioinformatics* **35** (ed Birol, I.) 3055–3062. ISSN: 1367-4803, 1367-4811. `https://academic.oup.com/bioinformatics/article/35/17/3055/5292387` (2024) (Sept. 2019).
5. Argelaguet, R. *et al.* MOFA+: a statistical framework for comprehensive integration of multi-modal single-cell data. en. *Genome Biology* **21,** 111. ISSN: 1474-760X. `https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-02015-1` (2024) (Dec. 2020).
6. Lundberg, S. & Lee, S.-I. *A Unified Approach to Interpreting Model Predictions* Version Number: 2. 2017. `https://arxiv.org/abs/1705.07874` (2024).
7. Ribeiro, M. T., Singh, S. & Guestrin, C. *"Why Should I Trust You?": Explaining the Predictions of Any Classifier* Version Number: 3. 2016. `https://arxiv.org/abs/1602.04938` (2024).
8. Dhamdhere, K., Agarwal, A. & Sundararajan, M. *The Shapley Taylor Interaction Index* Version Number: 2. 2019. `https://arxiv.org/abs/1902.05622` (2024).
9. O'Brien, R. J. & Wong, P. C. Amyloid precursor protein processing and Alzheimer's disease. eng. *Annual Review of Neuroscience* **34,** 185–204. ISSN: 1545-4126 (2011).

10. Hampel, H. *et al.* The Amyloid- Pathway in Alzheimer's Disease. en. *Molecular Psychiatry* **26,** 5481–5503. ISSN: 1359-4184, 1476-5578. https://www.nature.com/articles/s41380-021-01249-0 (2024) (Oct. 2021).

11. Kinney, J. W. *et al.* Inflammation as a central mechanism in Alzheimer's disease. en. *Alzheimer's & Dementia: Translational Research & Clinical Interventions* **4,** 575–590. ISSN: 2352-8737, 2352-8737. https://alz-journals.onlinelibrary.wiley.com/doi/10.1016/j.trci.2018.06.014 (2024) (Jan. 2018).

12. Ferrari-Souza, J. P. *et al.* *APOE* 4 associates with microglial activation independently of A plaques and tau tangles. en. *Science Advances* **9,** eade1474. ISSN: 2375-2548. https://www.science.org/doi/10.1126/sciadv.ade1474 (2024) (Apr. 2023).

13. You, Y. *et al.* Alzheimer's Disease Associated AKAP9 I2558M Mutation Alters Posttranslational Modification and Interactome of Tau and Cellular Functions in CRISPR-edited Human Neuronal Cells. *Aging Cell* **21,** e13617. ISSN: 1474-9726. PMID: 35567427 (June 2022).

14. Sun, X., Zhou, M., Pu, J. & Wang, T. Stachydrine exhibits a novel antiplatelet property and ameliorates platelet-mediated thrombo-inflammation. en. *Biomedicine & Pharmacotherapy* **152,** 113184. ISSN: 07533322. https://linkinghub.elsevier.com/retrieve/pii/S075333222200573X (2024) (Aug. 2022).

15. Gulaj, E., Pawlak, K., Bien, B. & Pawlak, D. Kynurenine and Its Metabolites in Alzheimer's Disease Patients. *Advances in Medical Sciences* **55,** 204–211. ISSN: 1898-4002. PMID: 20639188 (2010).

16. Choi, J. J., Svaren, J. & Wang, D. CoTF-reg Reveals Cooperative Transcription Factors in Oligodendrocyte Gene Regulation Using Single-Cell Multi-Omics. *Communications Biology* **8,** 181. ISSN: 2399-3642. PMID: 39910206 (Feb. 5, 2025).

17. Hornig, J. *et al.* The Transcription Factors Sox10 and Myrf Define an Essential Regulatory Network Module in Differentiating Oligodendrocytes. en. *PLoS Genetics* **9** (ed Barres, B. A.) e1003907. ISSN: 1553-7404. https://dx.plos.org/10.1371/journal.pgen.1003907 (2024) (Oct. 2013).

18. Triner, D. *et al.* Myc-Associated Zinc Finger Protein Regulates the Proinflammatory Response in Colitis and Colon Cancer via STAT3 Signaling. eng. *Molecular and Cellular Biology* **38,** e00386–18. ISSN: 1098-5549 (Nov. 2018).

19. Bujalka, H. *et al.* MYRF Is a Membrane-Associated Transcription Factor That Autoproteolytically Cleaves to Directly Activate Myelin Genes. en. *PLoS Biology* **11** (ed ffrench-Constant, C.) e1001625. ISSN: 1545-7885. https://dx.plos.org/10.1371/journal.pbio.1001625 (2024) (Aug. 2013).

20. Schmidt, K., Glaser, G., Wernig, A., Wegner, M. & Rosorius, O. Sox8 Is a Specific Marker for Muscle Satellite Cells and Inhibits Myogenesis. en. *Journal of Biological Chemistry* **278,** 29769–29775. ISSN: 00219258. https://linkinghub.elsevier.com/retrieve/pii/S0021925820842434 (2024) (Aug. 2003).

21. Sarkar, A. & Hochedlinger, K. The sox family of transcription factors: versatile regulators of stem and progenitor cell fate. eng. *Cell Stem Cell* **12,** 15–30. ISSN: 1875-9777 (Jan. 2013).

22. McGettrick, A. F. & O'Neill, L. A. The Role of HIF in Immunity and Inflammation. en. *Cell Metabolism* **32,** 524–536. ISSN: 15504131. https://linkinghub.elsevier.com/retrieve/pii/S1550413120304150 (2024) (Oct. 2020).

23. Hau, A.-C. *et al.* Transcriptional cooperation of PBX1 and PAX6 in adult neural progenitor cells. eng. *Scientific Reports* **11,** 21013. ISSN: 2045-2322 (Oct. 2021).

24. Lee, J.-R. Protein tyrosine phosphatase PTPRT as a regulator of synaptic formation and neuronal development. eng. *BMB reports* **48,** 249–255. ISSN: 1976-670X (May 2015).

25. Sithanandam, G. & Anderson, L. M. The ERBB3 receptor in cancer and cancer gene therapy. eng. *Cancer Gene Therapy* **15,** 413–448. ISSN: 1476-5500 (July 2008).

26. Fossati, M. *et al.* Trans-Synaptic Signaling through the Glutamate Receptor Delta-1 Mediates Inhibitory Synapse Formation in Cortical Pyramidal Neurons. eng. *Neuron* **104,** 1081–1094.e7. ISSN: 1097-4199 (Dec. 2019).

27. Lin, Y.-H. *et al.* LIMCH1 regulates nonmuscle myosin-II activity and suppresses cell migration. eng. *Molecular Biology of the Cell* **28,** 1054–1065. ISSN: 1939-4586 (Apr. 2017).

28. Zhou, A.-X., Hartwig, J. H. & Akyürek, L. M. Filamins in cell signaling, transcription and organ development. en. *Trends in Cell Biology* **20,** 113–123. ISSN: 09628924. `https://linkinghub.elsevier.com/retrieve/pii/S0962892409002876` (2024) (Feb. 2010).

29. Kondoh, K. & Nishida, E. Regulation of MAP kinases by MAP kinase phosphatases. en. *Biochimica et Biophysica Acta (BBA) - Molecular Cell Research* **1773,** 1227–1237. ISSN: 01674889. `https://linkinghub.elsevier.com/retrieve/pii/S0167488906004538` (2024) (Aug. 2007).

30. Briscoe, J. *et al.* Homeobox gene Nkx2.2 and specification of neuronal identity by graded Sonic hedgehog signalling. en. *Nature* **398,** 622–627. ISSN: 0028-0836, 1476-4687. `https://www.nature.com/articles/19315` (2024) (Apr. 1999).

31. Chatterjee, M. *et al.* Contactin-2, a synaptic and axonal protein, is reduced in cerebrospinal fluid and brain tissue in Alzheimer's disease. eng. *Alzheimer's Research & Therapy* **10,** 52. ISSN: 1758-9193 (June 2018).

32. Citri, A. & Malenka, R. C. Synaptic Plasticity: Multiple Forms, Functions, and Mechanisms. en. *Neuropsychopharmacology* **33,** 18–41. ISSN: 0893-133X, 1740-634X. `https://www.nature.com/articles/1301559` (2024) (Jan. 2008).

33. Marques, B. L. *et al.* Neurobiology of glycine transporters: From molecules to behavior. en. *Neuroscience & Biobehavioral Reviews* **118,** 97–110. ISSN: 01497634. `https://linkinghub.elsevier.com/retrieve/pii/S0149763420304826` (2024) (Nov. 2020).

34. McGowan, M. *et al.* The Application of Neurodiagnostic Studies to Inform the Acute Management of a Newborn Presenting With Sarbamoyl Shosphate Synthetase 1 Deficiency. eng. *Child Neurology Open* **8,** 2329048X20985179. ISSN: 2329-048X (2021).

35. Rak, J. *et al.* Cytohesin 1 regulates homing and engraftment of human hematopoietic stem and progenitor cells. eng. *Blood* **129,** 950–958. ISSN: 1528-0020 (Feb. 2017).

36. Steelman, A. J. *et al.* Activation of oligodendroglial Stat3 is required for efficient remyelination. eng. *Neurobiology of Disease* **91,** 336–346. ISSN: 1095-953X (July 2016).

37. Jain, M. *et al.* Role of JAK/STAT in the Neuroinflammation and its Association with Neurological Disorders. eng. *Annals of Neurosciences* **28,** 191–200. ISSN: 0972-7531 (July 2021).

38. Zhou, C. *et al.* JUN is a key transcriptional regulator of the unfolded protein response in acute myeloid leukemia. eng. *Leukemia* **31,** 1196–1205. ISSN: 1476-5551 (May 2017).

39. Caprariello, A. V., Mangla, S., Miller, R. H. & Selkirk, S. M. Apoptosis of oligodendrocytes in the central nervous system results in rapid focal demyelination. eng. *Annals of Neurology* **72,** 395–405. ISSN: 1531-8249 (Sept. 2012).

40. Yan, M. *et al.* SOCS modulates JAK-STAT pathway as a novel target to mediate the occurrence of neuroinflammation: Molecular details and treatment options. en. *Brain Research Bulletin* **213,** 110988. ISSN: 03619230. https://linkinghub.elsevier.com/retrieve/pii/S0361923024001217 (2024) (July 2024).

41. Dominguez, E., Mauborgne, A., Mallet, J., Desclaux, M. & Pohl, M. SOCS3-mediated blockade of JAK/STAT3 signaling pathway reveals its major contribution to spinal cord neuroinflammation and mechanical allodynia after peripheral nerve injury. eng. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* **30,** 5754–5766. ISSN: 1529-2401 (Apr. 2010).

42. Zhou, Q. & Anderson, D. J. The bHLH Transcription Factors OLIG2 and OLIG1 Couple Neuronal and Glial Subtype Specification. en. *Cell* **109,** 61–73. ISSN: 00928674. https://linkinghub.elsevier.com/retrieve/pii/S0092867402006773 (2024) (Apr. 2002).

43. Galiano, M. *et al.* Myelin basic protein functions as a microtubule stabilizing protein in differentiated oligodendrocytes. en. *Journal of Neuroscience Research* **84,** 534–541. ISSN: 0360-4012, 1097-4547. https://onlinelibrary.wiley.com/doi/10.1002/jnr.20960 (2024) (Aug. 2006).

44. Mendiola, A. S. *et al.* Defining blood-induced microglia functions in neurodegeneration through multiomic profiling. en. *Nature Immunology* **24,** 1173–1187. ISSN: 1529-2908, 1529-2916. https://www.nature.com/articles/s41590-023-01522-0 (2024) (July 2023).

45. Yin, Z. *et al.* APOE4 impairs the microglial response in Alzheimer's disease by inducing TGF-mediated checkpoints. en. *Nature Immunology* **24,** 1839–1853. ISSN: 1529-2908, 1529-2916. https://www.nature.com/articles/s41590-023-01627-6 (2024) (Nov. 2023).

46. Schäfer, M. K.-H. *et al.* Complement C1q Is Dramatically Up-Regulated in Brain Microglia in Response to Transient Global Cerebral Ischemia. en. *The Journal of Immunology* **164,** 5446–5452. ISSN: 0022-1767, 1550-6606. https://journals.aai.org/jimmunol/article/164/10/5446/32714/Complement-C1q-Is-Dramatically-Up-Regulated-in (2024) (May 2000).

47. Fraser, D. A., Pisalyaput, K. & Tenner, A. J. C1q enhances microglial clearance of apoptotic neurons and neuronal blebs, and modulates subsequent inflammatory cytokine production. eng. *Journal of Neurochemistry* **112,** 733–743. ISSN: 1471-4159 (Feb. 2010).

48. Voskobiynyk, Y. *et al.* Alzheimer's disease risk gene BIN1 induces Tau-dependent network hyperexcitability. eng. *eLife* **9,** e57354. ISSN: 2050-084X (July 2020).

49. Matyash, M., Matyash, V., Nolte, C., Sorrentino, V. & Kettenmann, H. Requirement of functional ryanodine receptor type 3 for astrocyte migration. en. *The FASEB Journal* **16,** 1–25. ISSN: 0892-6638, 1530-6860. https://onlinelibrary.wiley.com/doi/10.1096/fj.01-0380fje (2024) (Jan. 2002).

50. Zhang, Y. *et al.* Purification and Characterization of Progenitor and Mature Human Astrocytes Reveals Transcriptional and Functional Differences with Mouse. eng. *Neuron* **89,** 37–53. ISSN: 1097-4199 (Jan. 2016).

51. Massart, J. & Zierath, J. R. Role of Diacylglycerol Kinases in Glucose and Energy Homeostasis. en. *Trends in Endocrinology & Metabolism* **30,** 603–617. ISSN: 10432760. https://linkinghub.elsevier.com/retrieve/pii/S1043276019301201 (2024) (Sept. 2019).

52. Lorente-Gea, L., García, B., Martín, C., Quirós, L. M. & Fernández-Vega, I. Heparan sulfate proteoglycans and heparanases in Alzheimer's disease: current outlook and potential therapeutic targets. eng. *Neural Regeneration Research* **12,** 914–915. ISSN: 1673-5374 (June 2017).

53. Sadick, J. S. *et al.* Astrocytes and oligodendrocytes undergo subtype-specific transcriptional changes in Alzheimer's disease. en. *Neuron* **110,** 1788–1805.e10. ISSN: 08966273. https://linkinghub.elsevier.com/retrieve/pii/S0896627322002446 (2024) (June 2022).

54. Pandya, V. A. & Patani, R. Region-specific vulnerability in neurodegeneration: lessons from normal ageing. eng. *Ageing Research Reviews* **67,** 101311. ISSN: 1872-9649 (May 2021).

55. Chen, S. *et al.* Spatially resolved transcriptomics reveals genes associated with the vulnerability of middle temporal gyrus in Alzheimer's disease. en. *Acta Neuropathologica Communications* **10,** 188. ISSN: 2051-5960. https://actaneurocomms.biomedcentral.com/articles/10.1186/s40478-022-01494-6 (2024) (Dec. 2022).

56. Davey, J. *et al.* Exploring the role of the posterior middle temporal gyrus in semantic cognition: Integration of anterior temporal lobe with executive processes. eng. *NeuroImage* **137,** 165–177. ISSN: 1095-9572 (Aug. 2016).

57. Harrison, N. A., Doeller, C. F., Voon, V., Burgess, N. & Critchley, H. D. Peripheral inflammation acutely impairs human spatial memory via actions on medial temporal lobe glucose metabolism. eng. *Biological Psychiatry* **76,** 585–593. ISSN: 1873-2402 (Oct. 2014).

58. Liddelow, S. A. *et al.* Neurotoxic Reactive Astrocytes Are Induced by Activated Microglia. *Nature* **541,** 481–487. ISSN: 0028-0836, 1476-4687. https://www.nature.com/articles/nature21029 (2025) (Jan. 2017).

59. Gao, C., Jiang, J., Tan, Y. & Chen, S. Microglia in Neurodegenerative Diseases: Mechanism and Potential Therapeutic Targets. *Signal Transduction and Targeted Therapy* **8,** 359. ISSN: 2059-3635. https://www.nature.com/articles/s41392-023-01588-0 (2025) (Sept. 22, 2023).

60. Schartz, N. D. *et al.* C5aR1 Antagonism Suppresses Inflammatory Glial Responses and Alters Cellular Signaling in an Alzheimer's Disease Mouse Model. *Nature Communications* **15,** 7028. ISSN: 2041-1723. https://www.nature.com/articles/s41467-024-51163-6 (2025) (Aug. 15, 2024).

61. Liang, W.-W. *et al.* Integrative Multi-Omic Cancer Profiling Reveals DNA Methylation Patterns Associated with Therapeutic Vulnerability and Cell-of-Origin. *Cancer Cell* **41,** 1567–1585.e7. ISSN: 1878-3686. PMID: 37582362 (Sept. 11, 2023).

62. Lim, J. *et al.* Advances in Single-Cell Omics and Multiomics for High-Resolution Molecular Profiling. *Experimental & Molecular Medicine* **56,** 515–526. ISSN: 2092-6413. https://www.nature.com/articles/s12276-024-01186-2 (2025) (Mar. 5, 2024).

63. Emani, P. S. *et al.* Single-cell genomics and regulatory networks for 388 human brains. en. *Science* **384,** eadi5199. ISSN: 0036-8075, 1095-9203. https://www.science.org/doi/10.1126/science.adi5199 (2025) (May 2024).

64. Gupta, C. *et al. Network-based drug repurposing for psychiatric disorders using single-cell genomics* en. Dec. 2024. http://medrxiv.org/lookup/doi/10.1101/2024.12.01.24318008 (2025).

65. Chandrashekar, P. B. *et al.* DeepGAMI: deep biologically guided auxiliary learning for multimodal integration and imputation to improve genotype–phenotype prediction. en. *Genome Medicine* **15,** 88. ISSN: 1756-994X. https://genomemedicine.biomedcentral.com/articles/10.1186/s13073-023-01248-6 (2025) (Oct. 2023).

66. Cohen Kalafut, N., Huang, X. & Wang, D. Joint variational autoencoders for multimodal imputation and embedding. en. *Nature Machine Intelligence* **5,** 631–642. ISSN: 2522-5839. https://www.nature.com/articles/s42256-023-00663-z (2025) (May 2023).

67. Alatkar, S. A. & Wang, D. CMOT: Cross-Modality Optimal Transport for multimodal inference. en. *Genome Biology* **24,** 163. ISSN: 1474-760X. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-023-02989-8 (2025) (July 2023).

68. Bennett, D. A. *et al.* Religious Orders Study and Rush Memory and Aging Project. eng. *Journal of Alzheimer's disease: JAD* **64,** S161–S189. ISSN: 1875-8908 (2018).

69. Mathys, H. *et al.* Single-cell atlas reveals correlates of high cognitive function, dementia, and resilience to Alzheimer's disease pathology. en. *Cell* **186,** 4365–4385.e27. ISSN: 00928674. https://linkinghub.elsevier.com/retrieve/pii/S009286742300973X (2024) (Sept. 2023).

70. Ben-Kiki, O., Bercovich, A., Lifshitz, A. & Tanay, A. Metacell-2: a divide-and-conquer metacell algorithm for scalable scRNA-seq analysis. en. *Genome Biology* **23,** 100. ISSN: 1474-760X. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-022-02667-1 (2024) (Dec. 2022).

71. Endo, F. *et al.* Molecular basis of astrocyte diversity and morphology across the CNS in health and disease. en. *Science* **378,** eadc9020. ISSN: 0036-8075, 1095-9203. https://www.science.org/doi/10.1126/science.adc9020 (2024) (Nov. 2022).

72. Karch, C. M., Cruchaga, C. & Goate, A. M. Alzheimer's Disease Genetics: From the Bench to the Clinic. en. *Neuron* **83,** 11–26. ISSN: 08966273. https://linkinghub.elsevier.com/retrieve/pii/S0896627314004851 (2024) (July 2014).

73. Malik, M. *et al.* CD33 Alzheimer's Risk-Altering Polymorphism, CD33 Expression, and Exon 2 Splicing. en. *The Journal of Neuroscience* **33,** 13320–13325. ISSN: 0270-6474, 1529-2401. https://www.jneurosci.org/lookup/doi/10.1523/JNEUROSCI.1224-13.2013 (2024) (Aug. 2013).

74. Reiman, E. M. *et al.* GAB2 Alleles Modify Alzheimer's Risk in APOE 4 Carriers. en. *Neuron* **54,** 713–720. ISSN: 08966273. https://linkinghub.elsevier.com/retrieve/pii/S0896627307003790 (2024) (June 2007).

75. Wilkaniec, A., Gassowska-Dobrowolska, M., Strawski, M., Adamczyk, A. & Czapski, G. A. Inhibition of cyclin-dependent kinase 5 affects early neuroinflammatory signalling in murine model of amyloid beta toxicity. en. *Journal of Neuroinflammation* **15,** 1. ISSN: 1742-2094. https://jneuroinflammation.biomedcentral.com/articles/10.1186/s12974-017-1027-y (2024) (Dec. 2018).

76. Victoria, G. S., Arkhipenko, A., Zhu, S., Syan, S. & Zurzolo, C. Astrocyte-to-neuron intercellular prion transfer is mediated by cell-cell contact. en. *Scientific Reports* **6,** 20762. ISSN: 2045-2322. https://www.nature.com/articles/srep20762 (2024) (Feb. 2016).

77. Lau, S.-F., Cao, H., Fu, A. K. Y. & Ip, N. Y. Single-nucleus transcriptome analysis reveals dysregulation of angiogenic endothelial cells and neuroprotective glia in Alzheimer's disease. en. *Proceedings of the National Academy of Sciences* **117,** 25800–25809. ISSN: 0027-8424, 1091-6490. https://pnas.org/doi/full/10.1073/pnas.2008762117 (2024) (Oct. 2020).

78. Roses, A. *et al.* Understanding the genetics of *APOE* and *TOMM40* and role of mitochondrial structure and function in clinical pharmacology of Alzheimer's disease. en. *Alzheimer's & Dementia* **12,** 687–694. ISSN: 1552-5260, 1552-5279. https://alz-journals.onlinelibrary.wiley.com/doi/10.1016/j.jalz.2016.03.015 (2024) (June 2016).

79. Gabitto, M. I. *et al.* Integrated multimodal cell atlas of Alzheimer's disease. en. *Nature Neuroscience.* ISSN: 1097-6256, 1546-1726. https://www.nature.com/articles/s41593-024-01774-5 (2024) (Oct. 2024).

80. Megill, C. *et al. cellxgene: a performant, scalable exploration platform for high dimensional sparse matrices* en. Apr. 2021. http://biorxiv.org/lookup/doi/10.1101/2021.04.05.438318 (2024).

81. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. en. *Genome Biology* **15,** 550. ISSN: 1474-760X. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0550-8 (2024) (Dec. 2014).

82. Chen, Y. & Colonna, M. Microglia in Alzheimer's disease at single-cell level. Are there common patterns in humans and mice? eng. *The Journal of Experimental Medicine* **218,** e20202717. ISSN: 1540-9538 (Sept. 2021).

83. Bhaskar, K. *et al.* Microglial derived tumor necrosis factor- drives Alzheimer's disease-related neuronal cell cycle events. en. *Neurobiology of Disease* **62,** 273–285. ISSN: 09699961. https://linkinghub.elsevier.com/retrieve/pii/S0969996113002829 (2024) (Feb. 2014).

84. Ajoolabady, A., Lindholm, D., Ren, J. & Pratico, D. ER stress and UPR in Alzheimer's disease: mechanisms, pathogenesis, treatments. en. *Cell Death & Disease* **13,** 706. ISSN: 2041-4889. https://www.nature.com/articles/s41419-022-05153-5 (2024) (Aug. 2022).

85. Rangaraju, S. *et al.* Identification and therapeutic modulation of a pro-inflammatory subset of disease-associated-microglia in Alzheimer's disease. en. *Molecular Neurodegeneration* **13,** 24. ISSN: 1750-1326. https://molecularneurodegeneration.biomedcentral.com/articles/10.1186/s13024-018-0254-8 (2024) (Dec. 2018).

86. Sun, N. *et al.* Human microglial state dynamics in Alzheimer's disease progression. en. *Cell* **186,** 4386–4403.e29. ISSN: 00928674. https://linkinghub.elsevier.com/retrieve/pii/S0092867423009716 (2024) (Sept. 2023).

87. Damodaran, B. B., Flamary, R., Seguy, V. & Courty, N. An Entropic Optimal Transport loss for learning deep neural networks under label noise in remote sensing images. en. *Computer Vision and Image Understanding* **191,** 102863. ISSN: 10773142. https://linkinghub.elsevier.com/retrieve/pii/S1077314219301559 (2024) (Feb. 2020).

88. Liaw, R. *et al. Tune: A Research Platform for Distributed Model Selection and Training* Version Number: 1. 2018. https://arxiv.org/abs/1807.05118 (2025).

89. Bergstra, J., Yamins, D. & Cox, D. D. Hyperopt: Distributed asynchronous hyper-parameter optimization. *Astrophysics Source Code Library.* ADS Bibcode: 2022ascl.soft05008B, ascl:2205.008. https://ui.adsabs.harvard.edu/abs/2022ascl.soft05008B (2025) (May 2022).

90. Yu, G. Thirteen years of clusterProfiler. en. *The Innovation* **5,** 100722. ISSN: 26666758. https://linkinghub.elsevier.com/retrieve/pii/S2666675824001607 (2025) (Nov. 2024).

91. Pang, Z. *et al.* MetaboAnalyst 6.0: towards a unified platform for metabolomics data processing, analysis and interpretation. en. *Nucleic Acids Research* **52,** W398–W406. ISSN: 0305-1048, 1362-4962. https://academic.oup.com/nar/article/52/W1/W398/7642060 (2025) (July 2024).

92. Shannon, P. *et al.* Cytoscape: a software environment for integrated models of biomolecular interaction networks. eng. *Genome Research* **13,** 2498–2504. ISSN: 1088-9051 (Nov. 2003).

93. Flamary, R. *et al.* POT: Python Optimal Transport. *Journal of Machine Learning Research* **22,** 1–8. http://jmlr.org/papers/v22/20-451.html (2021).

94. Demetci, P., Santorella, R., Sandstede, B., Noble, W. S. & Singh, R. SCOT: Single-Cell Multi-Omics Alignment with Optimal Transport. *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology* **29,** 3–18. ISSN: 1557-8666. PMID: 35050714 (Jan. 2022).

95. Choi, J. *COSIME code* https://doi.org/10.5281/zenodo.15428769. Zenodo. 2025.

Acknowledgements

J.J.C. was supported by National Institutes of Health (NIH) grants under funding codes R21NS128761, RF1MH128695, R01AG067025, and P50HD105353, all awarded to D.W.; the National Science Foundation CAREER Award 2144475, also to D.W.; the Wisconsin Distinguished Graduate Fellowship (WDGF) from the Waisman Center; and an NIH grant under funding code RF1AG054047 awarded to C.D.E. N.C.K. and T.G. were supported by the same NIH and NSF grants awarded to D.W. (R21NS128761, RF1MH128695, R01AG067025, P50HD105353, and NSF CAREER Award 2144475). T.L. was supported by NIH grant RF1AG054047 awarded to C.D.E.

Author Information

Authors and Affiliations

Department of Population Health Sciences, University of Wisconsin-Madison, Madison, WI 53726, USA
Jerome J. Choi, Corinne D. Engelman, and Tianyuan Lu

Waisman Center, University of Wisconsin-Madison, Madison, WI 53705, USA
Jerome J. Choi, Noah Cohen Kalafut, and Daifeng Wang

Department of Computer Sciences, University of Wisconsin-Madison, Madison, WI 5706, USA
Noah Cohen Kalafut and Daifeng Wang

Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, Madison, WI 53076, USA
Tim Gruenloh and Daifeng Wang

Contributions

Conceptualization, J.J.C. and D.W.; Methodology, J.J.C. and N.C.K.; Formal Analysis, J.J.C. and T.G.; Investigation, J.J.C., T.L., and D.W.; Writing–Original Draft, J.J.C.; Writing–Review and Edit, J.J.C., N.C.K., T.G., C.D.E., T.L., and D.W.; Supervision, T.L. and D.W.; Funding Acquisition, C.D.E. and D.W..

Corresponding Authors

Correspondence to Daifeng Wang.

Ethics Declarations

Competing interests

The authors declare no competing interests.