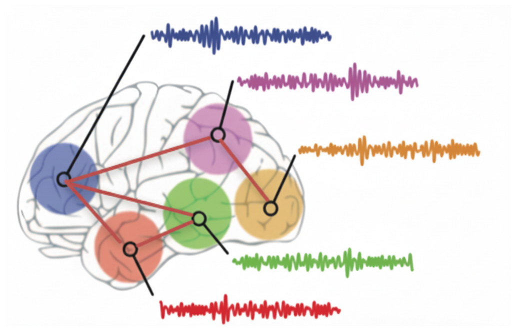
Precision Neural Networks: Joint Graph and Relational Learning

Andrea Cavallo*, **Samuel Rey**@†, Antonio G. Marques†, Elvin Isufi*

@samuel.rey.escudero@urjc.es, *Delft University of Technology, Delft, Netherlands, †King Juan Carlos University, Madrid, Spain

Introduction

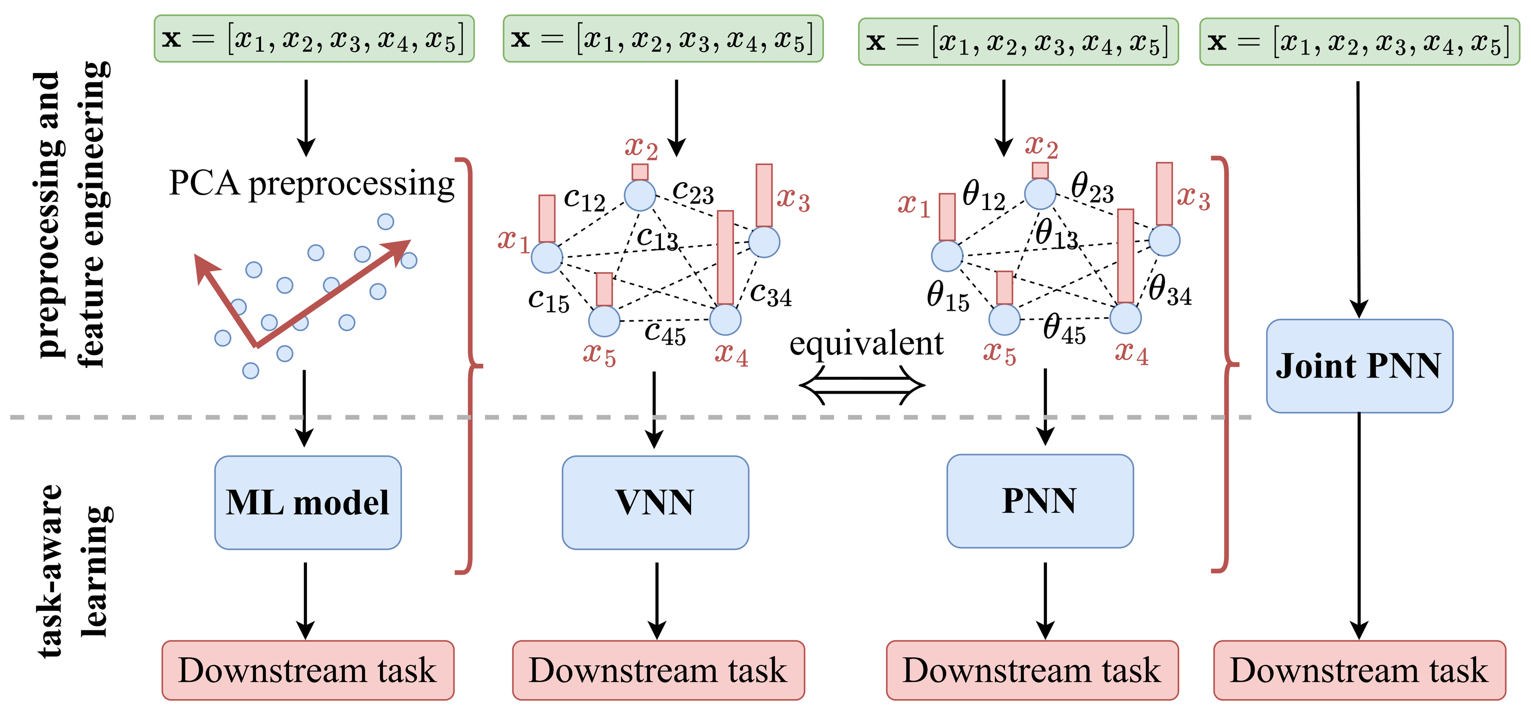
Consider observations $\mathbf{X} \in \mathbb{R}^{N\times T}$ (e.g., brain signals) and targets $\mathbf{y} \in \mathbb{R}^{T}$ (e.g., a patient's age).



The covariance matrix $\mathbf{C}$, or precision matrix $\mathbf{\Theta}_0 = \mathbf{C}^{-1}$, can serve as an inductive bias to predict $\mathbf{y}$ from $\mathbf{X}$:

(1) PCA projects data $\mathbf{x} \in \mathbb{R}^N$ on the (principal) covariance eigenvectors: $\tilde{\mathbf{x}} = \mathbf{V}^{\mathsf{T}}\mathbf{x}$, with $\mathbf{C} = \mathbf{V}\mathbf{W}\mathbf{V}^{\mathsf{T}}$.

(2) coVariance Neural Networks (VNNs) process data via nonlinear filter banks based on sample covariance $\hat{\mathbf{C}}$.



Limitations

(1) PCA is unstable to finite-sample estimation errors, task-agnostic and has limited expressivity.

(2) VNNs operate on the task-agnostic sample covariance.

Contribution

Precision Neural Networks (PNNs) jointly estimate a task-aware precision matrix in a stable and expressive manner.

Method

A Precision Neural Network (PNN) Φ is a VNN operating on matrix $\mathbf{\Theta}$:

$$\mathbf{x}_f^{(\ell)} = \sigma\left(\sum_{j=1}^{F_{\text{in}}} \mathbf{H}_{fj}^{(\ell)}(\mathbf{\Theta})\mathbf{x}_j^{(\ell-1)}\right), \quad \mathbf{H}_{fj}^{(\ell)}(\mathbf{\Theta}) = \sum_{k=0}^{K} h_{kfj}^{(\ell)}\mathbf{\Theta}^k.$$

It is trained to optimize a loss combining the downstream task, a graphical lasso penalty $\mathcal{L}_{\text{GL}}(\mathbf{X}_{\text{tr}}, \mathbf{\Theta}) = \text{tr}(\mathbf{C}\mathbf{\Theta}) - \text{logdet}(\mathbf{\Theta} + \epsilon\mathbf{I}) + \lambda\|\mathbf{\Theta}_{\bar{\mathcal{D}}}\|_1$ and a closeness penalty:

$$\min_{\mathbf{h},\tilde{\mathbf{\Theta}},\mathbf{\Theta}} \quad \alpha\mathcal{L}_{\text{task}}(\mathbf{y}_{\text{tr}}, \Phi(\mathbf{X}_{\text{tr}}, \tilde{\mathbf{\Theta}}, \mathbf{h})) + (1-\alpha)\mathcal{L}_{\text{GL}}(\mathbf{X}_{\text{tr}}, \mathbf{\Theta}) + \frac{\gamma}{2}\|\mathbf{\Theta} - \tilde{\mathbf{\Theta}}\|_F^2 \quad \text{s.t.} \quad \mathbf{\Theta} \succeq 0, \quad \|\mathbf{\Theta}\|_2 \leq M.$$

Alternating optimization (AO)

Step 1. Update $\mathbf{\Theta}$

$$\mathbf{\Theta}^{(i+1)} = \underset{\mathbf{\Theta}}{\text{argmin}} \quad (1-\alpha)\mathcal{L}_{\text{GL}}(\mathbf{X}_{\text{tr}}, \mathbf{\Theta}) + \frac{\gamma}{2}\|\mathbf{\Theta} - \tilde{\mathbf{\Theta}}^{(i)}\|_F^2 \quad \text{s.t.} \quad \mathbf{\Theta} \succeq 0, \quad \|\mathbf{\Theta}\|_2 \leq M.$$

Step 2. Update $\tilde{\mathbf{\Theta}}$

$$\tilde{\mathbf{\Theta}}^{(i+1)} = \underset{\tilde{\mathbf{\Theta}}}{\text{argmin}} \quad \alpha\mathcal{L}_{\text{task}}(\mathbf{y}_{\text{tr}}, \Phi(\mathbf{X}_{\text{tr}}, \tilde{\mathbf{\Theta}}, \mathbf{h}^{(i)})) + \frac{\gamma}{2}\|\mathbf{\Theta}^{(i+1)} - \tilde{\mathbf{\Theta}}\|_F^2.$$

Step 3. Update $\mathbf{h}$

$$\mathbf{h}^{(i+1)} = \underset{\mathbf{h}}{\text{argmin}} \quad \alpha\mathcal{L}_{\text{task}}(\mathbf{y}_{\text{tr}}, \Phi(\mathbf{X}_{\text{tr}}, \tilde{\mathbf{\Theta}}^{(i+1)}, \mathbf{h})).$$

The AO strategy comes with several benefits:

⇒ All steps can be solved via first-order methods.

⇒ Step 1 is **strongly convex**; global $\mathbf{\Theta}$ can be found.

Theorem

With high probability, there exist constants $m_1, m_2 > 0$ such that

$$\|\mathbf{\Theta}^{(i)} - \mathbf{\Theta}_0\|_F \leq m_1\sqrt{\frac{(N+S)\log N}{T}} + m_2\|\mathbf{\Theta}_0 - \tilde{\mathbf{\Theta}}^{(i)}\|_*^{1/2}.$$

⇒ PNNs recover $\mathbf{\Theta}_0$ at the rate $\mathcal{O}(T^{-1/2})$.

⇒ Sparser $\mathbf{\Theta}_0$ (lower S) is easier to recover.

⇒ Task labels affect recovery via $\tilde{\mathbf{\Theta}}^{(i)}$.

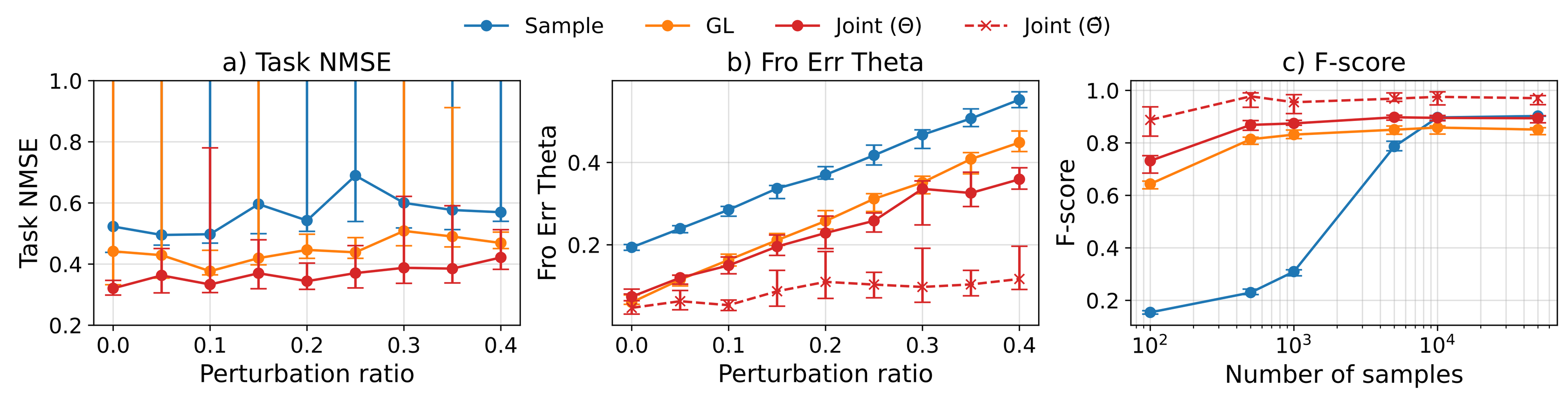
Experiments

Synthetic data

Generate sparse $\mathbf{\Theta}_0$ from an ER graph with 50 nodes

⇒ Draw 1000 observations $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Theta}_0^{-1})$.

⇒ Target signal $\mathbf{y} = \mathbf{H}(\tilde{\mathbf{\Theta}})\mathbf{x} + \mathbf{z}$ from perturbed $\tilde{\mathbf{\Theta}}$.



Takeaways. Task-aware PNN outperform the alternatives

⇒ Task can serve as a prior to improve graph estimation

Real data

Observations $\mathbf{x}$ are cortical thickness measurements; regression targets $\mathbf{y}$ are patients' biological ages.

Method	ADNI		ABIDE	
	MAE	#Zeros	MAE	#Zeros
PCA	10.3±0.7	–	4.41±0.34	–
VNN	6.06±0.20	–	5.81±0.11	–
Sample	5.78±0.10	0±0	5.94±0.02	0±0
GL	5.81±0.06	4546±0	5.80±0.11	2232±0
Joint	**5.31±0.18**	1225±1126	**3.64±0.65**	2771±976

Takeaways. Joint PNNs achieve best downstream MAE.

⇒ Recover better and sparser precision matrices

Conclusion

PNNs jointly estimate a sparse precision matrix and a GNN, improving task performance and precision reconstruction quality.

Paper