SPARSE NEURAL ADDITIVE MODEL: INTERPRETABLE DEEP LEARNING WITH FEATURE SELECTION VIA GROUP SPARSITY

Shiyun Xu, Zhiqi Bu
Department of AMCS
University of Pennsylvania
{shiyunxu,zbu}@sas.upenn.edu

Pratik Chaudhari
Department of Electrical and Systems Engineering
University of Pennsylvania
pratikac@seas.upenn.edu

Ian J. Barnett
Department of Biostatistics, Epidemiology, and Informatics
University of Pennsylvania
ibarnett@pennmedicine.upenn.edu

ABSTRACT

Interpretable machine learning has demonstrated impressive performance while preserving explainability. In particular, neural additive models (NAM) offer the interpretability to the black-box deep learning and achieve state-of-the-art accuracy among the large family of generalized additive models. In order to empower NAM with feature selection and improve the generalization, we propose the sparse neural additive models (SNAM) that employ the group sparsity regularization (e.g. Group LASSO), where each feature is learned by a sub-network whose trainable parameters are clustered as a group. We study the theoretical properties for SNAM with novel techniques to tackle the non-parametric truth, thus extending from classical sparse linear models such as the LASSO, which only works on the parametric truth.

Specifically, we show that the estimation error of SNAM vanishes asymptotically as $n \to \infty$. We also prove that SNAM, similar to LASSO, can have exact support recovery, i.e. perfect feature selection, with appropriate regularization. Moreover, we show that the SNAM can generalize well and preserve the 'identifiability', recovering each feature's effect. We validate our theories via extensive experiments and further testify to the good accuracy and efficiency of SNAM.

1 INTRODUCTION

Deep learning has shown dominating performance on learning complex tasks, especially in high-stake domains such as finance, healthcare and criminal justice. However, most neural networks are not naturally as interpretable as decision trees or linear models. Even to answer fundamental questions like "what is the exact effect on the output if we perturb the input?", neural networks oftentimes rely on complicated and ad-hoc methods to explain the model behavior, with additional training steps and loose theoretical guarantee. As a result, the black-box nature of neural networks renders difficult and risky for human to trust deep learning models or at least to understand them.

There is a long line of work studying the interpretable machine learning. At high level, existing methods can be categorized into two classes: (1) model-agnostic methods, and (2) innately interpretable models. On one hand, model-agnostic methods aim to explain the predictions of models that are innately black-box, via the feature importance and local approximation, which include Shapley values (Shapley, 2016; Strumbelj & Kononenko, 2014; Lundberg & Lee, 2017) and LIME (Ribeiro et al., 2016) as the representatives. On the other hand, directly interpretable models such as the decision-tree-based models and the generalized additive models (GAM), including the generalized linear models (GLM, (Nelder & Wedderburn, 1972)) as sub-cases, are the most widely applied and demonstrate amazing performance. Recently, the neural additive model (NAM) (Agarwal et al., 2020) introduces a new member into the GAM family, which applies sub-networks to learn f_j effectively, making accurate predictions while preserving the explainable power.

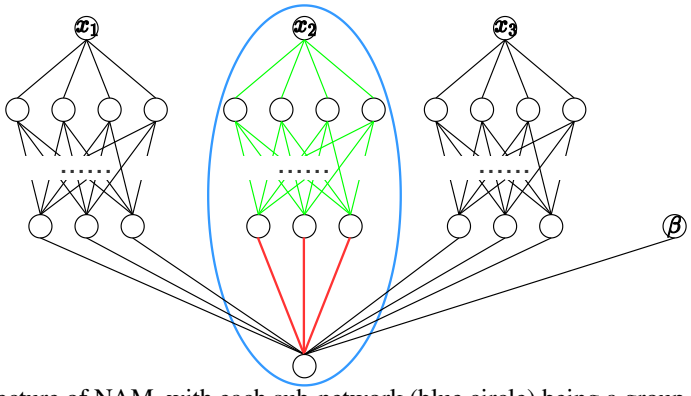

Figure 1: Architecture of NAM, with each sub-network (blue circle) being a group for Group LASSO regularization in SNAM. Note that in multi-class, multi-label, and multi-task problems, the last layer can have multiple neurons.

Yet, theoretical results about NAM on some important questions are missing: Does the convergence of NAM behave nicely? Does NAM guarantee to learn the true additive model consistently, as sample size increases? How to modify NAM such as to select features and whether the feature selection is accurate? Can we expect each sub-network in NAM to recover each f_j?

In this paper, we answer these questions in the affirmative. We study the sparse NAM with specific group sparsity regularization, especially the Group LASSO (Meier et al., 2008; Friedman et al., 2001), which reduces to NAM when the penalty is zero. We highlight that our SNAM is the first innately interpretable model that simultaneously uses neural networks and allows feature selection.

For theoretical analysis, we focus on the setting

$$\mathbf{y} = \sum_{j=1}^{p} f_j\left(\mathbf{X}_j\right) + \boldsymbol{\epsilon} \tag{1}$$

where i.i.d. samples $\mathbf{X}_j \sim \mathcal{X}_j$ for $j \in [p]$ where $\mathcal{X}_j$ is some distribution and the noise $\boldsymbol{\epsilon} \sim SG(\sigma^2)$ where SG means sub-Gaussian with variance σ^2.

2 SNAM: Model and Linearization Regime

To analyze To analyze SNAM under the regularization, for the j-th sub-network, we write the trainable parameters of as $\boldsymbol{\Theta}_j$ (visualized in Figure 1 by the blue circle) and the output as h_j. Then we write the SNAM output as

$$h(\mathbf{X}, \boldsymbol{\Theta}) = \sum\nolimits_j h_j(\mathbf{X}_j, \boldsymbol{\Theta}_j) + \beta$$

With these notations in place, we can learn the model via the following SNAM optimization problem with the Group LASSO regularization and an arbitrary loss $\mathcal{L}$:

$$\min_{\boldsymbol{\Theta}, \beta} \mathcal{L}\big(\mathbf{y}, \sum_j h_j(\mathbf{X}_j, \boldsymbol{\Theta}_j) + \beta\big) + \lambda \sum_j \|\boldsymbol{\Theta}_j\|_2. \tag{2}$$

Notably, the group structure defined on sub-networks is the key to feature selection in SNAM: it explicitly penalizes $\boldsymbol{\Theta}_j$ so that the entries in $\boldsymbol{\Theta}_j$ are either all non-zero or all zero. The latter case happens when λ is large, resulting in the j-th feature to be not selected as $h_j = 0$.

In fact, if each sub-network has only a single parameter β_j and no hidden layers at all, then the Group LASSO penalty is equivalent to the LASSO penalty: $\|\beta_j\|_2 = |\beta_j|$. Therefore, we view LASSO as the simplest version of SNAM with Group LASSO regularization. This connection leads to the theoretical findings in this work, since we will analyze the linearization of SNAM via the random feature (RF):

$$h^{\text{RF}}(\mathbf{X}, \boldsymbol{\theta}) = \sum_{j=1}^{p} h_j^{\text{RF}}(\mathbf{X}_j, \boldsymbol{\theta}_j) = \sum_{j=1}^{p} \mathbf{G}_j \boldsymbol{\theta}_j \tag{3}$$

where $\boldsymbol{\theta} := [\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_p]$ is the last layer, $\mathbf{w} := [\mathbf{w}_1, \cdots, \mathbf{w}_p]$ are hidden layers, and the random feature map $\mathbf{G}_j := g_j(\mathbf{X}_j, \mathbf{w}(0)) \in \mathbb{R}^{n \times m}$ is the forward propagation of the j-th sub-network until the output layer. Therefore, the corresponding optimization for the RF network is

$$\hat{\boldsymbol{\theta}}^{\text{RF}} := \text{argmin}_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{y}, \mathbf{G}\boldsymbol{\theta}) + \lambda \sum_j \|\boldsymbol{\theta}_j\|_2 \tag{4}$$

where $\mathbf{G} := [\mathbf{G}_1, \cdots, \mathbf{G}_p]$ is the concatenation of $\mathbf{G}_j$. In this regime, SNAM is linear in trainable parameters $\boldsymbol{\theta}$ (though non-linear in input $\mathbf{X}$) and is indeed a kernel regression, a topic with rich theoretical understanding.

3 Non-Asymptotic Analysis of SNAM

We study the primal problem

$$\hat{\boldsymbol{\theta}} := \text{argmin}_{\boldsymbol{\theta}} \frac{1}{2} \|\mathbf{y} - \sum\nolimits_j \mathbf{G}_j \boldsymbol{\theta}_j\|_2^2 + \lambda \sum\nolimits_j \|\boldsymbol{\theta}_j\|_2 \tag{5}$$

and equivalently the dual problem

$$\hat{\boldsymbol{\theta}} := \text{argmin}_{\boldsymbol{\theta}: \sum_j \|\boldsymbol{\theta}_j\|_2 \le \mu} \frac{1}{2} \|\mathbf{y} - \sum\nolimits_j \mathbf{G}_j \boldsymbol{\theta}_j\|_2^2 \tag{6}$$

We point out that although the analysis of SNAM is similar to that of LASSO at high level, our analysis is technically more involved and requires novel tools, due to the fact that the true model (1) is non-parametric (unlike the LASSO whose true model is parametric).

3.1 Slow Rate with Group LASSO Penalty

Similar to the analysis of slow rate for the LASSO (Wainwright, 2009), our analysis needs SNAM to overfit the training data under the low-dimensional $\mathbf{G}$ regime.

Theorem 3.1. *Under Assumption A.1 and Assumption A.2, supposing $|f_j|$ is upper bounded by constant c_j and noise $\epsilon \sim SG(\sigma^2)$, then with probability at least $1 - \delta_1 - \delta_2$, we have for $\hat{\theta}$ in* (6)*,*

$$\frac{1}{n} \|\sum_j (\mathbf{f}_j - \mathbf{G}_j \hat{\boldsymbol{\theta}}_j)\|_2^2 \le \frac{2\sigma}{\sqrt{n}} \left(\sum_j c_j / \sqrt{\delta_2} + \mu \max_j \sqrt{\mathbb{E} g_j(\mathcal{X}_j, \mathbf{w}_j(0))^2} \sqrt{2 \log(m_j / \delta_1)} \right)$$

where m_j is the width of output layer in the j-th sub-network and μ is the penalty coefficient.

The MSE $\frac{1}{n} \|\sum_j (\mathbf{f}_j - \mathbf{G}_j \hat{\boldsymbol{\theta}}_j)\|_2^2$ converges to zero with rate $1/\sqrt{n}$ as $n \to \infty$. We note that the convergence rate of SNAM has the same order as that of LASSO, but SNAM requires two probability quantities δ_1, δ_2 due to the non-parametric true model (1), whereas the LASSO only needs δ_1.

3.2 Exact Support Recovery

The support recovery for parametric models like LASSO is defined on the parameters, e.g. $supp(\hat{\boldsymbol{\beta}}) = \{j : \hat{\beta}_j \neq 0\}, supp(\boldsymbol{\beta}) = \{j : \beta_j \neq 0\}$(Bühlmann & Van De Geer, 2011; Wainwright, 2009; Tibshirani & Wasserman, 2017). For non-parametric models like SPAM and our SNAM, the support is instead defined on the functions $S = supp(f) = \{j : f_j \neq 0\}$. We prove in Appendix B that, with proper Group LASSO regularization, the SNAM recovers the true $supp(f)$ exactly.

Theorem 3.2. *Under a few assumptions (Assumption A.2, Assumption A.3 and Assumption A.4 in Appendix A), then*

$$\lambda > \max_{j \notin S} \|\mathbf{G}_j^\top\|_\infty \|\mathbf{y}\|_\infty / \gamma$$

guarantees that the SNAM solution $\hat{\boldsymbol{\theta}}$ in (5) *has the exact support recovery, i.e. $supp(h) = supp(f)$.*

4 Asymptotic Analysis of SNAM

In this section, we study the asymptotic consistency of SNAM and hence indicate its good generalization behavior. Our results build on top of the asymptotic zero loss given by the slow rate in Theorem 3.1. The proofs can be found in Appendix B.

4.1 CONSISTENCY

We show in Theorem 4.1 that the SNAM h_n, when trained on n samples, converges to the unknown true model f in a probability measure.

Theorem 4.1. *Under the assumptions in Theorem 3.1, we have the convergence in probability measure:*

$$\lim_{n\to\infty} \rho(\{x \in \mathcal{X} : |f(x) - h_n(x)| \geq \varepsilon\}) = 0$$

for arbitrarily small $\epsilon > 0$. Here ρ is the probability measure of $\mathcal{X}$, the joint distribution of data $\mathbf{X}$. In words, the prediction function h_n converges to the true model f.

4.2 EFFECT IDENTIFIABILITY

Another more difficult challenge in the generalized additive models is the identifiability of individual effects, in the sense that we want to have $h_j \to f_j$ for all $j \in [p]$.

Theorem 4.2 (Effect Identifiability)**.** *Assuming $h_n \to f$ in probability measure of $\mathcal{X}$ as $n \to \infty$, if $\mathcal{X}_j$ is independent of $\mathcal{X}_{-j}$, then $\lim_{n\to\infty} h_{n,j}(x)$ converges to $f_j(x)$ in probability up to a constant.*

5 EXPERIMENTS

We conduct multiple experiments on both synthetic and real datasets. We emphasize that here SNAM is not RF SNAM, i.e. we train all parameters in sub-networks. We use MSE loss for regression, cross-entropy (CE) loss for classification, and wall-clock time for all tasks. Furthermore, we compare SNAM to other sparse interpretable methods: NAM, ℓ_1 linear support vector machine (SVM), LASSO and SPAM (Ravikumar et al., 2009).

5.1 SYNTHETIC DATASETS

To validate our statistical analysis on SNAM, i.e. the feature selection (or support recovery), the estimation consistency and the effect identifiability, we experiment on synthetic regression and classification datasets. We emphasize that, it is necessary to work with synthetic data instead of real-world ones, since we need access to the truth f_j for our performance measures.

5.1.1 DATA GENERATION

We generate a data matrix $\mathbf{X} \in \mathbb{R}^{3000\times 24}$ and denote the j-th column of $\mathbf{X}$ as $\mathbf{X}_j$. $\mathbf{y}$ is generated by the following additive model for a regression task:

$$\mathbf{y} = f_1(\mathbf{X}_1) + \cdots + f_{24}(\mathbf{X}_{24}) + \mathcal{N}(0, 1)$$

where all f_j are zero functions except

$$\begin{aligned} f_1(x) &= 2x^2 \tanh x, \\ f_2(x) &= \sin x \cos x + x^2, \\ f_3(x) &= 20/(1 + e^{-5\sin x}), \\ f_4(x) &= 20 \sin^3 2x - 6\cos x + x^2 \end{aligned}$$

5.1.2 PERFORMANCE MEASURES

Denote the output of each sub-network as $\hat{f}_j$. To illustrate the performance on the support recovery, we use precision and recall to compare $\hat{f}_j$ and truth f_j. In particular, we use ℓ_2 norm of a sub-network's weights to indicate whether $\hat{f}_j = 0$. We define the identification error (iden. error),

$$\min_{c_j\in\mathbb{R}} \frac{1}{n} \|\hat{f}_j(\mathbf{X}_j) - f_j(\mathbf{X}_j) - c_j\|_2^2.$$

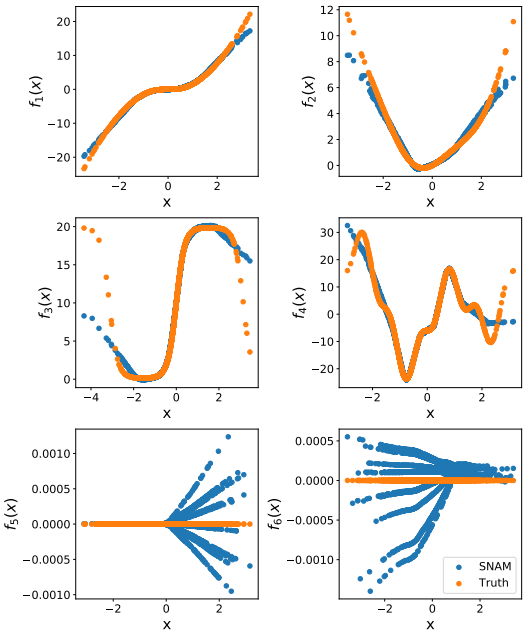

Figure 2: Individual effect learned by SNAM on synthetic regression. Blue dots are prediction $\hat{f}_j(\mathbf{X}_j)$ and orange dots are truth $f_j(\mathbf{X}_j)$, with $j = 1, \cdots, 6$.

	ℓ_1 SVM	LASSO	SPAM	SNAM
MSE loss	140.7	139.7	25.75	**10.61**
Precision	0.17	**1.00**	0.17	**1.00**
Recall	**1.00**	**1.00**	**1.00**	**1.00**
Iden. error	5.90	6.09	3.07	**0.69**
Time (sec)	0.005	0.007	152.1	48.52
#. Feature	24	4	4	4
#. Param	24	4	-	127201

Table 1: Performance of sparse interpretable methods on synthetic regression.

5.1.3 RESULTS

In Table 1, for regression task, SNAM domintes existing sparse interpretable methods in all measures. Especially, SNAM (which includes LASSO as a sub-case) is the only method that achieves exact support recovery, obtaining perfect precision and recall scores. When facing complicated target functions, SNAM, as a non-linear model, significantly outperforms linear models like linear SVM and LASSO, in terms of test loss and identification error. In contrast to SPAM, another non-linear model that achieves low loss, SNAM outperforms in both loss and efficiency, with a 3 times speed-up. We further visualize the effects learned by SNAM in Figure 2, demonstrating the strong approximation offered by the neural networks.

5.2 COMPAS CLASSIFICATION

COMPAS is a widely used commercial tool to predict the recidivism risk based on defendants' features and it is known for its racial bias against the black defendants. The ProPublica released the recidivism dataset (Angwin et al., 2016), that includes the characteristics of defendants in Broward County, Florida, and the predictions on reoffending by the COMPAS algorithm. This dataset has 6172 examples and 13 features [1].

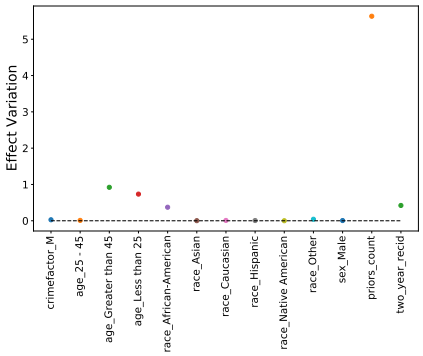

Figure 3: Variation of effects learned by SNAM on COMPAS dataset.

A closer look at Figure 3 describes the relations between features and the variation of effect, which is gap between the minimum recidivism risk and the maximum one among all individual samples for a particular feature, i.e. $\max_i \hat{f}_j(X_{ij}) - \min_i \hat{f}_j(X_{ij})$. If the variation of an effect is large, then SNAM indicates the feature is significant. Indeed, the top 5 features selected by SNAM are prior counts, ages, two year recidivism and whether the defendant is African American. The last feature clearly demonstrates SNAM's explanability of the COMPAS algorithm's racial bias. In short, the features selected by SNAM are consistent with NAM's selection based on shape functions.

6 DISCUSSION

For future directions, one may further extend SNAM's theory to the fast convergence rate (Van De Geer & Bühlmann, 2009) in sample size, or to the jointly trained SNAM in terms of time. We believe the theoretical analysis and empirical evaluation can be explored for a whole family of interesting SNAMs. For example, while SNAM with Group LASSO penalty contains LASSO as sub-case, we can view SNAM with Group SLOPE (Brzyski et al., 2019) penalty as extension of SLOPE (Bogdan et al., 2015). Other possible extensions of elastic net (Zou & Hastie, 2005), adaptive LASSO (Zou, 2006), K-level SLOPE (Zhang & Bu, 2021; Bu et al., 2021) are also possible with SNAM.

[1] The data preprocessing follows `https://github.com/propublica/compas-analysis`.

REFERENCES

Rishabh Agarwal, Nicholas Frosst, Xuezhou Zhang, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. *arXiv preprint arXiv:2004.13912*, 2020.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. propublica, may 23, 2016, 2016.

Małgorzata Bogdan, Ewout Van Den Berg, Chiara Sabatti, Weijie Su, and Emmanuel J Candès. Slope—adaptive variable selection via convex optimization. *The annals of applied statistics*, 9(3): 1103, 2015.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

Damian Brzyski, Alexej Gossmann, Weijie Su, and Małgorzata Bogdan. Group slope–adaptive selection of groups of predictors. *Journal of the American Statistical Association*, 114(525): 419–433, 2019.

Zhiqi Bu, Jason Klusowski, Cynthia Rush, and Weijie J Su. Characterizing the slope trade-off: A variational perspective and the donoho-tanner limit. *arXiv preprint arXiv:2105.13302*, 2021.

Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, 2011.

Rick Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pp. 4768–4777, 2017.

Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008.

John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384, 1972.

Pradeep Ravikumar, John Lafferty, Han Liu, and Larry Wasserman. Sparse additive models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(5):1009–1030, 2009.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Lloyd S Shapley. *17. A value for n-person games*. Princeton University Press, 2016.

Erik Strumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665, 2014.

Ryan Tibshirani and Larry Wasserman. Sparsity, the lasso, and friends. *Lecture notes from "Statistical Machine Learning," Carnegie Mellon University, Spring*, 2017.

Sara A Van De Geer and Peter Bühlmann. On the conditions used to prove oracle results for the lasso. *Electronic Journal of Statistics*, 3:1360–1392, 2009.

Martin J Wainwright. Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1 -constrained quadratic programming (lasso). *IEEE transactions on information theory*, 55(5): 2183–2202, 2009.

Yiliang Zhang and Zhiqi Bu. Efficient designs of slope penalty sequences in finite dimension. In *International Conference on Artificial Intelligence and Statistics*, pp. 3277–3285. PMLR, 2021.

Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429, 2006.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

A Assumptions of Main Results

Assumption A.1 (Overfitting of SNAM)**.** *Denoting the truth* $\mathbf{f}_j := f_j(\mathbf{X}_j)$*, we assume there exists* μ *such that*

$$\frac{1}{n}\|\mathbf{y} - \sum_j \mathbf{G}_j\hat{\boldsymbol{\theta}}_j\|_2^2 \leq \frac{1}{n}\|\mathbf{y} - \sum_j \mathbf{f}_j\|_2^2 = \frac{1}{n}\|\boldsymbol{\epsilon}\|_2^2.$$

To guarantee a unique solution of SNAM, we assume that the SNAM feature map $\mathbf{G}$ has full rank.

Assumption A.2 (Full rank of feature map)**.** $\mathbf{G} \in \mathbb{R}^{n\times M}$ *has full column rank* M *and thus* $\mathbf{G}^\top\mathbf{G} \in \mathbb{R}^{M\times M}$ *is invertible.*

Here M is the sum of numbers of neurons at the last hidden layer of each sub-network[2]. Our first result is the slow rate of the SNAM convergence $h(\mathbf{X}, \hat{\boldsymbol{\theta}}) \to f(\mathbf{X})$ as $n \to \infty$.

Assumption A.3 (Mutual incoherence)**.** *For some* $\gamma > 0$*, we have*

$$\left\| \left(\mathbf{G}_S^\top\mathbf{G}_S\right)^{-1}\mathbf{G}_S^\top\mathbf{G}_j \right\|_2 \leq 1 - \gamma, \text{ for } j \notin S \tag{7}$$

where $\mathbf{G}_S$ *is the concatenation of* $\mathbf{G}_j$ *for all* $j \in S$*.*

Next, we assume that the regularization is not too large to omit significant features.

Assumption A.4 (Maximum regularization)**.** *The Group LASSO penalty coefficient* λ *in* (5) *is small enough so that the following solution is dense*

$$\tilde{\boldsymbol{\theta}}_S := \text{argmin}_{\boldsymbol{\theta}_S} \frac{1}{2}\|\mathbf{y} - \sum_{j\in S}\mathbf{G}_j\boldsymbol{\theta}_j\|_2^2 + \lambda\sum_{j\in S}\|\boldsymbol{\theta}_j\|_2 \tag{8}$$

B Proofs of Main Results

B.1 Proof of Theorem 4.3

Proof. By the Lagrange duality, for any penalty $\lambda > 0$, there exists some $\mu > 0$ such that the optimization problem

$$\min_{\boldsymbol{\theta}} \frac{1}{2}\|\mathbf{y} - \sum_j \mathbf{G}_j\boldsymbol{\theta}_j\|_2^2 + \lambda\sum_j\|\boldsymbol{\theta}_j\|_2 \equiv \min_{\boldsymbol{\theta}} \frac{1}{2}\|\mathbf{y} - \sum_j \mathbf{G}_j\boldsymbol{\theta}_j\|_2^2 \quad \text{s.t. } \sum_j\|\boldsymbol{\theta}_j\|_2 \leq \mu$$

From Assumption A.1, the minimizer $\hat{\boldsymbol{\theta}}$ satisfies that

$$\frac{1}{n}\|\boldsymbol{\epsilon} + \sum_j(\mathbf{f}_j - \mathbf{G}_j\hat{\boldsymbol{\theta}}_j)\|_2^2 = \frac{1}{n}\|\mathbf{y} - \sum_j \mathbf{G}_j\hat{\boldsymbol{\theta}}_j\|_2^2 \leq \frac{1}{n}\|\mathbf{y} - \sum_j \mathbf{f}_j\|_2^2 = \frac{1}{n}\|\boldsymbol{\epsilon}\|_2^2. \tag{9}$$

Expanding the left-most term,

$$\frac{1}{n}\|\boldsymbol{\epsilon} + \sum_j(\mathbf{f}_j - \mathbf{G}_j\hat{\boldsymbol{\theta}}_j)\|_2^2 = \frac{1}{n}\|\boldsymbol{\epsilon}\|_2^2 + \frac{1}{n}\|\sum_j(\mathbf{f}_j - \mathbf{G}_j\hat{\boldsymbol{\theta}}_j)\|_2^2 + \frac{2}{n}\left\langle \boldsymbol{\epsilon}, \sum_j(\mathbf{f}_j - \mathbf{G}_j\hat{\boldsymbol{\theta}}_j)\right\rangle$$

Substituting back to (9) and after some rearranging, we get:

$$\begin{aligned}
&\frac{1}{n}\|\sum_j(\mathbf{f}_j - \mathbf{G}_j\hat{\boldsymbol{\theta}}_j)\|_2^2 \leq \frac{2}{n}\sum_j\left\langle \boldsymbol{\epsilon}, \mathbf{G}_j\hat{\boldsymbol{\theta}}_j - \mathbf{f}_j\right\rangle \leq \frac{2}{n}\sum_j(\|\boldsymbol{\epsilon}^\top\mathbf{G}_j\hat{\boldsymbol{\theta}}_j\|_2 + \|\boldsymbol{\epsilon}^\top\mathbf{f}_j\|_2)\\
&\leq \frac{2}{n}(\sum_j\|\mathbf{G}_j^\top\boldsymbol{\epsilon}\|_\infty\|\hat{\boldsymbol{\theta}}_j\|_2 + \sum_j\|\mathbf{f}_j\|_\infty\|\boldsymbol{\epsilon}\|_2) \leq \frac{2}{n}(\sum_j\|\mathbf{G}_j^\top\boldsymbol{\epsilon}\|_\infty\|\hat{\boldsymbol{\theta}}_j\|_2 + \sum_j c_j\|\boldsymbol{\epsilon}\|_2)
\end{aligned}$$

where the third inequality follows by the triangular inequality and the second last inequality holds by the Holder's inequality. Note that $\|\mathbf{G}_j^\top\boldsymbol{\epsilon}\|_\infty = \max_{k=1,2,\cdots,m}|(\mathbf{G}_j^\top)_k\boldsymbol{\epsilon}|$ is a maximum of m

[2]When all sub-networks have the same architecture, we write $M = mp$ where the last hidden layer width m. More generally, suppose the j-th sub-network has last hidde layer width m_j, then $M = \sum_j m_j$.

Gaussians. Here $(\mathbf{G}_j^\top)_k \in \mathbb{R}^n$ is the k-th feature fed into the output layer of the j-th sub-network. For each k, $(\mathbf{G}_j^\top)_k \boldsymbol{\epsilon}$ has mean zero and variance

$$\text{Var}((\mathbf{G}_j^\top)_k \boldsymbol{\epsilon}) = \sigma^2 \mathbb{E}((\mathbf{G}_j)_k^\top (\mathbf{G}_j)_k) = n\sigma^2 \mathbb{E} g_j(\mathcal{X}_j, \mathbf{w}_j(0))^2$$

By the maximal sub-Gaussian inequality Boucheron et al. (2013), for any $\delta_1 > 0$, with probability at least $1 - \delta_1$:

$$\|\mathbf{G}_j^\top \boldsymbol{\epsilon}\|_\infty = \max_{k=1,2,\cdots,m} |(\mathbf{G}_j)_k \boldsymbol{\epsilon}| \leq \sigma \sqrt{n \mathbb{E} g_j(\mathcal{X}_j, \mathbf{w}_j(0))^2} \sqrt{2 \log(m_j/\delta_1)}.$$

Furthermore, by Markov's inequality, with probability at least $1 - \delta_2$, we have $\|\boldsymbol{\epsilon}\|_2^2 \leq \mathbb{E}(\|\boldsymbol{\epsilon}\|_2^2)/\delta_2 = n\sigma^2/\delta_2$. In summary, we obtain

$$\begin{aligned} &\frac{1}{n} \| \sum_j (\mathbf{f}_j - \mathbf{G}_j \hat{\boldsymbol{\theta}}_j) \|_2^2 \leq \frac{2}{n} (\sum_j \|\mathbf{G}_j^\top \boldsymbol{\epsilon}\|_\infty \|\hat{\boldsymbol{\theta}}_j\|_2 + \sum_j c_j \|\boldsymbol{\epsilon}\|_2) \\ &\leq \frac{2\sigma}{\sqrt{n}} (\mu \max_j \sqrt{\mathbb{E} g_j(\mathcal{X}_j, \mathbf{w}_j(0))^2} \sqrt{2 \log(m_j/\delta_1)} + \sum_j c_j / \sqrt{\delta_2}) \end{aligned}$$

□

B.2 Proof of Theorem 4.7

We assume each sub-network has the same architecture, with last layer width m.

Proof. We construct and study a specific vector $\tilde{\boldsymbol{\theta}} \in \mathbb{R}^{|S|m \times 1}$ by setting $\tilde{\boldsymbol{\theta}}_S$ as in (8) and $\tilde{\boldsymbol{\theta}}_j = \mathbf{0}$ for $j \notin S$: denoting the complement set of S as S^C), we have:

$$\tilde{\boldsymbol{\theta}}_S = \text{argmin}_{\boldsymbol{\theta}_S} \frac{1}{2} \|\mathbf{y} - \sum_{j \in S} \mathbf{G}_j \boldsymbol{\theta}_j\|_2^2 + \lambda \sum_{j \in S} \|\boldsymbol{\theta}_j\|_2 \quad \text{and} \quad \tilde{\boldsymbol{\theta}}_{S^C} = \mathbf{0}.$$

From Assumption A.4 (maximum regularization), we have that $\tilde{\boldsymbol{\theta}}_S$ is dense, i.e. $\tilde{\boldsymbol{\theta}}_j \neq \mathbf{0}$ for all $j \in S$. Therefore, if the constructed $\tilde{\boldsymbol{\theta}}$ is indeed the SNAM solution $\hat{\boldsymbol{\theta}}$ in (5), then $supp(h) \supseteq supp(f)$. Further, $\tilde{\boldsymbol{\theta}}_{S^C} = \mathbf{0}$ leads to $supp(h) = S = supp(f)$.

Next, we check that the constructed $\tilde{\boldsymbol{\theta}}$ is indeed the solution of SNAM in (5) via the KKT condition, which requires that for all $j \in [p]$,

$$\mathbf{G}_j^\top (\sum_{l=1}^p \mathbf{G}_l \tilde{\boldsymbol{\theta}}_l - \mathbf{y}) + \lambda \mathbf{s}_j = \mathbf{G}_j^\top (\mathbf{G}_S \tilde{\boldsymbol{\theta}}_S - \mathbf{y}) + \lambda \mathbf{s}_j = 0 \tag{10}$$

Here $\mathbf{s}_j$ is the subgradient of $\|\tilde{\boldsymbol{\theta}}_j\|_2$, which is $\tilde{\boldsymbol{\theta}}_j / \|\tilde{\boldsymbol{\theta}}_j\|_2$ if $\tilde{\boldsymbol{\theta}}_j \neq \mathbf{0}$ and otherwise within a unit sphere. The first equality of (10) follows by the construction $\tilde{\boldsymbol{\theta}}_{S^C} = \mathbf{0}$. We break (10) into the support set S and its complement S^C,

$$\mathbf{G}_S^\top \left(\mathbf{y} - \mathbf{G}_S \tilde{\boldsymbol{\theta}}_S \right) = \lambda \mathbf{s}_S \tag{11}$$

$$\mathbf{G}_{S^C}^\top \left(\mathbf{y} - \mathbf{G}_S \tilde{\boldsymbol{\theta}}_S \right) = \lambda \mathbf{s}_{S^C} \tag{12}$$

Notice that if both KKT conditions (11) and (12) are satisfied by $\tilde{\boldsymbol{\theta}}$, then $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}$. For $j \in S$, the KKT condition in (11) is the same as that of (8) and hence satisfied by the definition of $\tilde{\boldsymbol{\theta}}_S$. For $j \notin S$, our goal is to show $\|\mathbf{s}_j\|_2 < 1$, which is a sufficient condition to guarantee $\tilde{\boldsymbol{\theta}}_{S^C} = \mathbf{0}$ and thus to satisfy the KKT condition (12).

To show $\|\mathbf{s}_j\|_2 < 1$, we can solve $\tilde{\boldsymbol{\theta}}_S$ from (11), leveraging the full rank of $\mathbf{G}_S^\top \mathbf{G}_S \in \mathbb{R}^{|S|m \times |S|m}$ from Assumption A.2, and obtain

$$\tilde{\boldsymbol{\theta}}_S = \left(\mathbf{G}_S^\top \mathbf{G}_S \right)^{-1} \left(\mathbf{G}_S^\top \mathbf{y} - \lambda \mathbf{s}_S \right)$$

Substituting the formula of $\tilde{\boldsymbol{\theta}}_S$ into (12) and denoting $\mathbf{P}_S := \mathbf{I} - \mathbf{G}_S \left(\mathbf{G}_S^\top \mathbf{G}_S \right)^{-1} \mathbf{G}_S^\top$, we get

$$\mathbf{s}_{S^C} = \frac{1}{\lambda} \mathbf{G}_{S^C}^\top \mathbf{P}_S \mathbf{y} + \mathbf{G}_{S^C}^\top \mathbf{G}_S \left(\mathbf{G}_S^\top \mathbf{G}_S \right)^{-1} \mathbf{s}_S$$

For $j \notin S$, taking the ℓ_2 norm and applying the triangular inequality give

$$\|\mathbf{s}_j\|_2 \leq \frac{1}{\lambda}\left\|\mathbf{G}_j^\top \mathbf{P}_S \mathbf{y}\right\|_2 + \left\|\mathbf{G}_j^\top \mathbf{G}_S \left(\mathbf{G}_S^\top \mathbf{G}_S\right)^{-1} \mathbf{s}_S\right\|_2 \tag{13}$$

Applying the Holder's inequality to the second term in (13) gives

$$\left\|\mathbf{G}_j^\top \mathbf{G}_S \left(\mathbf{G}_S^\top \mathbf{G}_S\right)^{-1} \mathbf{s}_S\right\|_2 \leq \left\|\mathbf{G}_j^\top \mathbf{G}_S \left(\mathbf{G}_S^\top \mathbf{G}_S\right)^{-1}\right\|_2 \|\mathbf{s}_S\|_\infty < 1-\gamma$$

where the inequality follows from Assumption A.3 (mutual incoherence).

Regarding the first term in (13), unlike in the LASSO support recovery analysis Wainwright (2009) where the maximal inequality is directly applicable, we seek new tools since $\left\{\left\|\mathbf{G}_j^\top \mathbf{P}_S \mathbf{y}\right\|_2\right\}$ are non-centered random variables. We apply the Holder's inequality to the first term in (13),

$$\frac{1}{\lambda}\left\|\mathbf{G}_j^\top \mathbf{P}_S \mathbf{y}\right\|_2 \leq \frac{1}{\lambda}\left\|\mathbf{G}_j^\top\right\|_\infty \left\|\mathbf{P}_S\right\|_2 \left\|\mathbf{y}\right\|_\infty \leq \frac{1}{\lambda}\left\|\mathbf{G}_j^\top\right\|_\infty \left\|\mathbf{y}\right\|_\infty$$

in which the last inequality follows from the fact that $\mathbf{P}_S$ is a projection matrix with $\|\mathbf{P}_S\|_2 \leq 1$.

All in all, we have

$$\max_{j \notin S} \|\mathbf{s}_j\|_2 \leq \frac{1}{\lambda} \max_{j \notin S} \left\|\mathbf{G}_j^\top\right\|_\infty \left\|\mathbf{y}\right\|_\infty + 1 - \gamma$$

and therefore, if $\lambda > \max_{j \notin S} \|\mathbf{G}_j^\top\|_\infty \|\mathbf{y}\|_\infty / \gamma$, then SNAM recovers the true support exactly. □

B.3 Proofs in Section 5

Proof of Theorem 5.1. From Theorem 3.1, we see that $\frac{1}{n}\|f(\mathbf{x}) - h_n(\mathbf{x})\|_2^2 = O_p(1/\sqrt{n}) = o_p(1)$. To prepare the proof of the convergence in probability measure, we consider the probaility space consisting of $(\mathcal{X}, E, \rho)$, where $\mathcal{X}$ is the sample space, E is the event space, and ρ is the probability measure. Defining the events $S_n := \{x \in \mathcal{X} : |f(x) - h_n(x)| \geq \epsilon\}$, we have $S_n \in E$.

We will prove the theorem by contradiction. If there exists an $\epsilon > 0$ such that for any $N, \delta > 0$, there is some $n_N > N$ such that $\rho(\{x \in \mathcal{X} : |f(x) - h_n(x)| \geq \epsilon\}) > \delta$.

However, since

$$\begin{aligned}\frac{1}{n}\|f(\mathbf{x}) - h_n(\mathbf{x})\|_2^2 &= \frac{1}{n}\sum_{i=1}^n (f(\mathbf{x}_i) - h_n(\mathbf{x}_i))_2^2 \geq \frac{1}{n}\sum_{\mathbf{x}_i \in S_n} (f(\mathbf{x}_i) - h_n(\mathbf{x}_i))_2^2 \\ &= \frac{1}{n}\sum_{i=1}^n \mathbb{I}(\mathbf{x}_i \in S_n)(f(\mathbf{x}_i) - h_n(\mathbf{x}_i))_2^2 \geq \frac{\epsilon^2}{n}\sum_{i=1}^n \mathbb{I}(\mathbf{x}_i \in S_n)\end{aligned}$$

Denote each random variable $\mathbb{I}(\mathbf{x}_i \in S_n) := Z_{n,i}$. Together they constitute a row-wise i.i.d. triangular array. Since $\sup_n \mathbb{E}(Z_{n,i}^2) \leq 1 < \infty$, by applying the weak law of large number for triangular array (Durrett, 2019, Theorem 2.2.11), we obtain

$$\frac{1}{n}\|f(\mathbf{x}) - h_n(\mathbf{x})\|_2^2 \geq \frac{\epsilon^2}{n}\sum_{i=1}^n \mathbb{I}(\mathbf{x}_i \in S_n) \xrightarrow{p} \epsilon^2 \mathbb{P}(x \in S_n) > \epsilon^2 \delta$$

This contradicts with the asymptotic zero estimation MSE, i.e. $\frac{1}{n}\|f(\mathbf{x}) - h_n(\mathbf{x})\|_2^2 \xrightarrow{p} 0$. □

Proof of Theorem 5.2 . Following the proof of Theorem 4.1, we know for any $\epsilon > 0, \delta > 0$, there exists N such that for any $n_N > N$, we have $\rho(\{x \in \mathcal{X} : |f(x) - h_n(x)| \geq \epsilon\}) < \delta$ and denote $S_n(\epsilon) := \{x \in \mathcal{X} : |f(x) - h_n(x)| \geq \epsilon\}$. We further denote $S_{n,j}^C := \{x_{-j} : (x_j, x_{-j}) \in S_n^C\}$ where S_n^C is the complement of S_n.

Under the condition that $\mathcal{X}_j$ is independent of $\mathcal{X}_{-j}$, we take the expectation with respect to $\mathcal{X}_{-j}$, using the marginal density as p_{-j}:

$$\int_{S_{n,j}^C} f(\mathcal{X}) p_{-j}(u) du = \int_{S_{n,j}^C} (f_{n,j}(\mathcal{X}_j) + f_{n,-j}(u)) p_{-j}(u) du = f_{n,j}(\mathcal{X}_j) + c_{j,1}$$

Notice that this integral is also bounded between $\int_{S_{n,j}^C} (h_{n,j}(\mathcal{X}_j) + h_{n,-j}(u) \pm \epsilon) p_{-j}(u) du = \mathbb{P}(\mathcal{X}_{-j} \in S_{n,j}^C)(h_{n,j}(\mathcal{X}_j) \pm \epsilon) + c_{j,2}$. The probability $\mathbb{P}(\mathcal{X}_{-j} \in S_{n,j}^C)$ goes to 1 as $\delta \to 0$. Further, as $\epsilon \to 0$, we have $h_{n,j}(\mathcal{X}_j) \xrightarrow{p} f(\mathcal{X}_j) + c_j$ for some constant c_j. □