BrainQuest: Perception-Guided Brain Network Comparison

Lei Shi
SKLCS, Institute of Software
Chinese Academy of Sciences
Beijing, China
shil@ios.ac.cn

Hanghang Tong
School of Computing
Arizona State University
Phoenix, USA
htong6@asu.edu

Xinzhu Mu
Academy of Arts & Design
Tsinghua University
Beijing, China
mxz12@mails.tsinghua.edu.cn

***Abstract*—Why are some people more creative than others? How do human brain networks evolve over time? A key stepping stone to both mysteries and many more is to compare weighted brain networks. In contrast to networks arising from other application domains, the brain network exhibits its own characteristics (e.g., high density, indistinguishability), which makes any off-the-shelf data mining algorithm as well as visualization tool sub-optimal or even mis-leading.**

In this paper, we propose a shift from the current mining-then-visualization paradigm, to jointly model these two core building blocks (i.e., mining and visualization) for brain network comparisons. The key idea is to integrate the human perception constraint into the mining block earlier so as to guide the analysis process. We formulate this as a multi-objective feature selection problem; and propose an integrated framework, BrainQuest, to solve it. We perform extensive empirical evaluations, both quantitatively and qualitatively, to demonstrate the effectiveness and efficiency of our approach.

I. Introduction

In recent decades, revolutionary neuroimaging techniques (e.g., multimodal MRI) have advanced the fundamental understandings of the neural connection and co-functioning of *in vivo* human brains, known as the brain network [1] or connectome [2]. The high-resolution measurement of brain networks opens the door to many data mining problems. In this paper, we focus on the comparative mining of weighted brain networks among labeled populations [3]. For example, what is the difference between the brain networks of a highly creative group and a normal group? How do brain networks evolve over time, in the aftermath of a major surgery?

At the first glance, it seems that many matured data mining techniques could conveniently lend themselves to this task. For example, feature selection and frequent graph mining which optimize quantitative performance measures, including the label classification accuracy, precision/recall, etc. However, we argue that, in the context of the brain network comparison, the *interpretability* of mining results for end users is at least as important as their quantitative performance measures. First, the current data generation process in both brain imaging and network creation is error-prone, and there is no generic comparative pattern on brain networks among different population groups. These factors lead to the significant uncertainties in the patterns detected by algorithms. Such patterns would be worthless without the cross-examination with historical records and the manual confirmation by domain experts. Second, the domain experts (e.g., neurologists and doctors) are not necessarily data mining experts with the knowledge of the full detail of mining algorithms. Instead, they might depend on visual interfaces (e.g., graphs drawn in Figure 2) to analyze the cortical difference. Third, on such interfaces, the mechanism for human users to discover comparative patterns and interpret the mining results is significantly different from a fully automatic algorithm. In fact, human users are largely governed by the perception theory of the vision system.

Applying the interpretability constraint by the human perception, most relevant data mining techniques in their current forms are sub-optimal for the brain network comparison task, if not infeasible at all. In particular, feature selection methods such as statistical hypothesis testing and sparse regression models [4][5] identify individual and/or collections of network connections that are discriminative among outcome groups (e.g., high/low IQ scores). However, the comparative pattern on the selected features at the perception-level is not often noticeable by the end users. On the other hand, when interaction effects among features are strong, feature selection methods might fail to detect the subgraph patterns that have been shown to be prevalent in brain networks [6].

The key innovation of this work is the joint modeling of the *discriminative objective* in data mining and the *interpretability constraint* in visualization guided by the *human perception mechanism*. We present BrainQuest, an integrated comparison framework on brain networks, that achieves effectiveness and efficiency from both data analytics and domain user's perspectives. Our major contributions can be summarized as below.

- *A Novel Problem Definition* based on an empirical study of real-world brain network characteristics (Section II), that integrates multiple objectives into a coherent feature selection formulation. We propose a new constraint on human perception which has not been studied before (Section III);
- *A Perception-Guided Modeling and Algorithm* namely the prioritized sparse group lasso, to fulfill our design

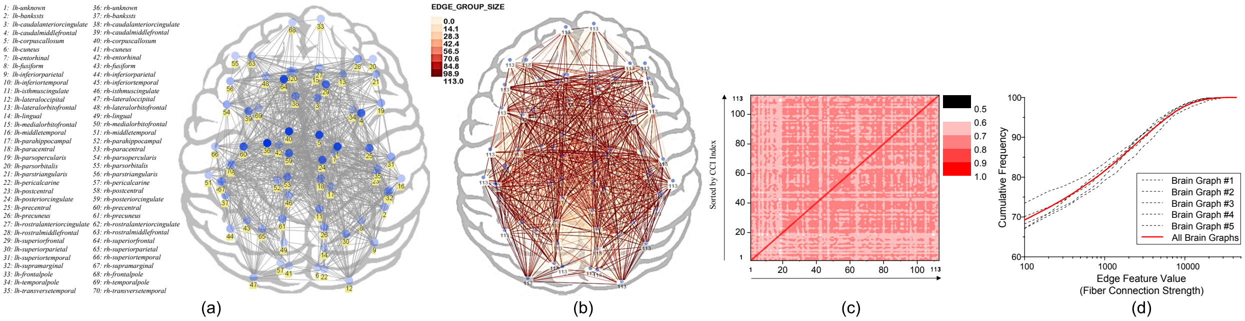
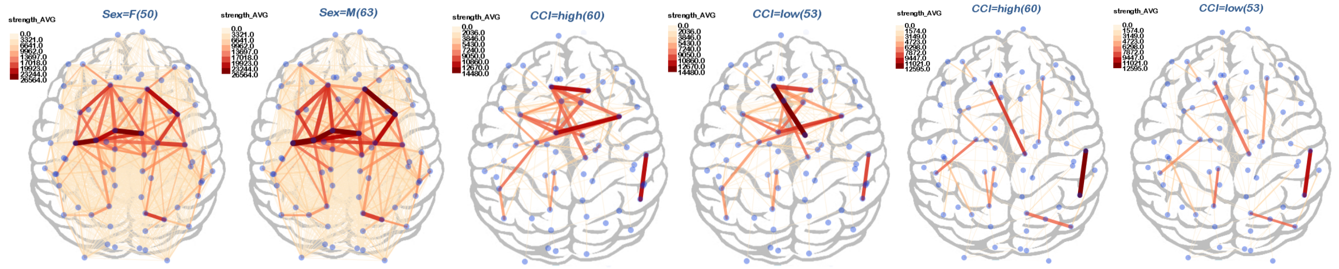


Figure 1. (a) One subject's brain network, nodes are placed at the center of each cerebral region, edges indicate fiber connections. The node label gives the region index, with a full list on the left. The node color shows its degree. (b) The aggregated brain network of all subjects, nodes are grouped by region index. The node label gives the number of aggregated regions and the edge color shows the number of original edges in the aggregation. (c) The correlation coefficient matrix of all subjects' unweighted topology vector. (d) The CDF of fiber connection strengths of 5 single subjects and all subjects.



(a) Comparison on gender with all features (b) Comparison on CCI with significant features (c) Comparison on CCI with features by lasso

Figure 2. Brain network comparison between two groups of subjects. Both edge thickness and color saturation indicate the average fiber strength in each group: (a) Female v.s. male in purely visual comparison, all edges are displayed. (b) High CCI v.s. low CCI, 129 edges are displayed, all with significant difference between groups ($p < 0.05$). (c) 83 edge features selected by lasso are displayed.

objectives simultaneously. The perception constraint is satisfied through the experiment-driven model calibration and a novel usage of the priority criterion for regularized, group-based feature selections (Section IV, V);

- *Comprehensive Evaluations* by both numeric experiments on quantitative measures linking to the design objectives, and the user study on comparison tasks in a practical scenario (Section VI).

II. Empirical Study

We studied the brain network of 113 subjects provided by the Open Connectome project [2]. The data is measured by multimodal MRI and processed in an automated pipeline [7]. Both gyral-based region-level small graphs and voxel-based low-level big graphs are estimated from the raw MRI data. Due to the wide acceptance of the gyral-based human brain division atlas [8], we focus on the region-level small graph in our study. On each subject, the small graph consists of 70 nodes, corresponding to cerebral regions in one human brain (35 in each hemisphere). The edge between a pair of nodes represents the fiber connection between cerebral regions, where the edge strength indicates the degree of connectivity. Each subject is recorded with rich demographic information, including their gender, age, and measures on Full-Scale IQ (FSIQ), Composite Creativity Index (CCI) and the Big Five personality traits. We classify the value of each measure into a few classes for the ease of comparison. For example, both FSIQ and CCI are divided into two classes: the high class with FSIQ (or CCI)≥100 and the low class with FSIQ (or CCI)<100. Three interesting properties are found on these brain networks, posing new challenges on the comparison task studied here.

High density. On each subject, the 70-node brain network has 800∼1208 edges, leading to a high graph density of 0.33∼0.5. Figure 1(a) illustrates the network of one random subject. The central regions are almost fully connected. Figure 1(b) further shows an aggregated brain network of all subjects by grouping nodes of the same region together. The aggregated network has 2016 edges and 50% edges are shared by at least a half of subjects.

Indistinguishability. The networks among subjects are similar in both topology and connection strength. To show that, we build the Pearson's correlation coefficient matrix of the unweighted topology vector of all subjects. Each topology vector has a length of 2415, including all binary fiber connections of a subject. Figure 1(c) depicts the matrix, almost every pair of subjects has a topology correlation larger than 0.6, and the average correlation is close to 0.7. The weighted topology correlations are even higher, with an average of 0.9. On the fiber connection strength, Figure 1(d) shows the CDF of connection strengths in 5 random subjects and also the CDF from all subjects. Both the percentage of weak connections (<100 in strength) and the distribution of stronger connections are quite similar among individuals.

Limitation of feature selection. In comparing weighted brain networks among subject groups, the inherent high graph density and similarity in topology make it difficult for a pure visualization-based approach. Figure 2(a) shows an example comparing female and male subjects, using the edge color and thickness to show the average connection strength. People can discover some differences on individual edges, but it is difficult to extract comparative subgraph patterns. In fact, on the full graph level, the attribute of subjects has little correlation with the overall topology. We order the correlation coefficient matrix in Figure 1(c) by subject's CCI index. The figure reveals no significant clustering pattern, except that the top 20 creative subjects have a little bit different topology from the others. These findings suggest using computational edge feature selection methods in the brain comparison task. Unfortunately, two baseline feature selection methods are shown to be ineffective in our initial studies. First, we conduct unpaired t-test on each edge connection between comparing groups. Only the edges with significant difference ($p<0.05$) are selected. Figure 2(b) shows an example in comparing high v.s. low CCI groups, in which 129 selected edges are displayed and trivial edges (average strength below 100) are removed. Though the comparison exhibits clear differences, it is shown that the selected edges may not directly contribute to the difference in outcome. We input these 129 edge features into a standard logistic regression model to predict the CCI group index. The average prediction accuracy under 10-fold cross-validations reaches 52.28%, even worse than that of a null model (53.1%). In the second trial, we apply L1 regularization with elastic net [5] on logistic regressions. The best prediction accuracy (85.5%) is achieved on $\alpha = 1$, corresponding to the lasso regularization [4]. Figure 2(c) depicts the 83 edges selected by lasso. These edges scatter uniformly over the graph, some even without noticeable difference in the visual comparison. It suggests that the success in predicting the outcome does not necessarily lead to an interpretable pattern in comparing brain networks.

III. Problem

We first introduce the notations used throughout the problem definition, as listed in Table I. The raw input is the brain network of N subjects under study, represented by undirected graphs $G_1, \cdots, G_N$. Each graph G_i is composed of a same number of nodes, denoted by n. Each node represents one gyral-based region covering thousands of adjacent MRI imaging voxels. There is an edge between each pair of nodes if fiber connections are detected between their regions. All edges are weighted by one continuous measure $\mathcal{X}$, normally the fiber connection strength. For simplicity, we assume each graph to have the same number of edges: $e_1, \cdots, e_p$, where $p = \frac{n(n-1)}{2}$. On G_i, the edge weight vector by $\mathcal{X}$ is denoted by $\boldsymbol{x_i} = (x_{i1}, \cdots, x_{ip})'$. For those edges that do not have fiber connection, we set their weight components to zero.

Table I
Notations.

SYMBOL	DESCRIPTION
N, G_i	# of subjects and their brain graphs
n, p, e_j	# of nodes, # of edges and each edge in the brain graph
$\mathcal{X}$, X, $\boldsymbol{x_i}$, x_{ij}	edge weight variable, weight matrix on all subjects, weight vector on G_i and the component on e_j
$\mathcal{Y}$, $\boldsymbol{y}$, y_i	outcome variable on subjects, value on all subjects and G_i
K, S_k, V_k	# of levels for the outcome variable, the subset of subjects for each level, and their aggregation views for comparison
$\mathcal{R}$, $\boldsymbol{r_k}$, r_{kj}	transfer function on edge aggregations, edge weight on V_k and e_j
$\boldsymbol{\gamma}$, γ_j	edge feature selection vector and the component for e_j
$\boldsymbol{X_\gamma}$, $V_k(\boldsymbol{\gamma})$	partial edge weight matrix, the view after feature selection

At the network level, each subject and their brain graph is associated with a discrete outcome variable $\mathcal{Y}$, e.g., the high/low CCI group of subjects by their CCI index. The value of $\mathcal{Y}$ on N subjects is denoted by the vector $\boldsymbol{y} = (y_1, \cdots, y_N)'$, where y_i has K possible levels. This outcome variable classifies all subjects into K disjoint subsets, $S_1, \cdots, S_K$. The brain graphs in each subset are aggregated into one view by the region index, generating K views for the visual comparison, denoted by $V_1, \cdots, V_K$. Due to the homogeneity of brain graphs, each view still has n nodes and $p = \frac{n(n-1)}{2}$ edges. The edge weight by $\mathcal{X}$ on each view is determined by a transfer function $\mathcal{R}$ over individual edge weights. By default, we apply the mean function which is used in standard visualization tools to illustrate the average brain connectivity of a group. The edge weight vector on the view of V_k is denoted by $\boldsymbol{r_k} = (r_{k1}, \cdots, r_{kp})'$. In this work, without loss of generality, we target the pairwise comparison ($K = 2$) between two views (V_1, V_2) aggregating brain networks by a binary label, e.g., the high/low CCI class.

Problem 1: Pairwise Brain Network Comparison

Given: (1) *the edge weight matrix $\boldsymbol{X}$ on a set of brain connectivity graphs (design matrix)*; (2) *the vector $\boldsymbol{y}$ of a binary label on these graphs (response vector)*; (3) *the transfer function $\mathcal{R}$ to aggregate edge weights onto the group-based views for visual comparison*;

Select: *the collection of useful edge features for comparison, represented by the feature selection vector $\boldsymbol{\gamma} = \{0, 1\}^p$*;

By optimizing four design objectives:

D1. **Discriminative** power by maximizing the binary classification accuracy on the label $\mathcal{Y}$ with selected features: $\max \mathrm{P}(\hat{y}_i = y_i | \boldsymbol{X_\gamma}, \boldsymbol{y})$,
where $\boldsymbol{X_\gamma}$ denotes the partial design matrix after feature selection, $\hat{y}_i$ is the predicted label on graph G_i;

D2. **Sparsity** by bounding the number of selected features: $\sum_{j=1}^{p} \gamma_j \leq t$,
where t is the parameter to control the sparsity. This is to avoid overfitting in learning brain network labels

because we have $p \gg N$, i.e., a fat design matrix;

D3. **Grouping** effect by maximizing the clustering coefficient of selected edge features in the aggregated views: $\max \sum_{k=1}^{K} \text{ClusterCoeff}(V_k(\boldsymbol{\gamma}))$, $V_k(\boldsymbol{\gamma})$ denotes the kth view after feature selection;

D4. **Visibility** of feature differences by a lower bound on the ratio of visible differences for comparison: $\text{P}(|r_{1j} - r_{2j}| \geq JND | \gamma_j = 1) \geq \xi$, where ξ is the visibility threshold, JND is the just noticeable difference in perception (Section V-B).

IV. Model and Algorithm

A. Prioritized Sparse Group Lasso

We propose an integrated model based on the regularization idea of lasso [4] to fulfill the four design objectives (i.e., D1∼D4 in Problem 1). The goal is to choose the optimal feature weight vector $\boldsymbol{w}$ (which determines the feature selection vector by $\gamma_j = \mathbf{1}_{(\mathbf{0},+\infty)}(w_j)$) that minimizes:

$$\underbrace{NLL(\boldsymbol{w})}_{D1} + \underbrace{\alpha\lambda||\boldsymbol{w}||_1}_{D2} + \underbrace{(1-\alpha)\lambda \sum_{m=1}^{M} \sqrt{p_m}\, \overbrace{\theta_m}^{D4}\, ||\boldsymbol{w}^{(\boldsymbol{m})}||_2}_{D3} \tag{1}$$

where $NLL(\boldsymbol{w}) = \sum_{i=1}^{N} log(1 + e^{-y_i \boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x_i}})$ denotes the Negative Log Likelihood (NLL) for the weight vector $\boldsymbol{w}$. Edge features are partitioned into M groups with size $p_1, \cdots, p_M$, splitting the weight vector $\boldsymbol{w}$ into sub-vectors $\boldsymbol{w}^{(\mathbf{1})}, \cdots, \boldsymbol{w}^{(\boldsymbol{M})}$.

The first term of this model is the NLL of a logistic regression model. Minimizing NLL leads to an optimization of the prediction accuracy, which meets the objective of discriminative power (D1). The second term excluding α is the standard L1 norm penalty to ensure feature sparsity (D2), where the parameter λ is to control the degree of sparsity. The third term is mostly derived from the group lasso penalty [9] to select subgraph patterns based on an existing grouping of edge features (D3). The parameter α is added to balance the groupwise sparsity and the within-group sparsity.

The modeling to satisfy the design objectives of D1∼D3 is well-known to the data mining community as variants of lasso methods [4][9][10], but a key challenge remains open, i.e., how to meet D4, the perception-level visibility of differences. Our major contribution in modeling is to propose a new prioritized mechanism on the group feature selection. The intuition is to encourage the selection of group of features with a higher visibility than the desired threshold; and suppress the selection of other less visible groups. This is achieved by introducing a priority parameter, denoted by θ_m, for each group of features. This model adaptation seems straightforward, but the optimization of these priorities is nontrivial. First, the selection/de-selection of groups of features is a complex process, which is coupled with the other parameters as well as the input data. We provide a theoretical analysis on this process to support our optimization-based solution (Section IV-C). Second, the exact modeling of visible differences for human in the brain network comparison is unsettled, which requires user experiments to calibrate the model (Section VI-A).

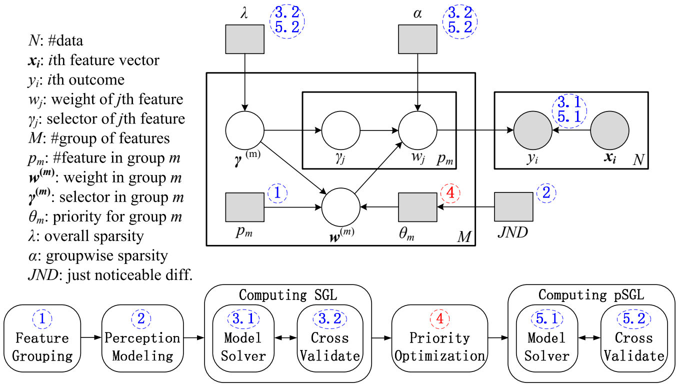


Figure 3. Graphical model of the framework and the solution pipeline.

The entire model is named the prioritized Sparse Group Lasso (pSGL). Figure 3 gives an explanation from the perspective of bayesian graphical model. Filled boxes are model parameters and input data, while hollow boxes are the variables to compute. This is similar to the modeling of group lasso and elastic net (Chapter 13.5 of [11]) except for the introduction of θ_m and JND. To solve this joint model, we propose a five-stage solution pipeline (Figure 3): (1) All edge features are grouped by existing categories or clustering algorithms (Section V-A); (2) The human perception model is established to compute the visibility of differences in the comparison (Section V-B); (3) A basic Sparse Group Lasso (SGL) model without priority is solved by the latest algorithm, and cross-validated to determine the best sparsity parameters (Section V-C); (4) The priority for each feature group is computed by an optimization algorithm (Section IV-B); (5) The pSGL model with priorities is solved to meet the visibility objective.

B. Optimization Algorithm for Priorities

In our solution, the key stage is to compute the priority θ_m for each group of features (Stage 4 in Figure 3), based on the result in solving the unprioritized SGL model (Stage 3 in Figure 3). The objectives in this stage are two-fold: (1) satisfy the visibility of difference constraint (D4) in the pSGL model; and (2) minimize the variance to the unprioritized SGL model. This can be formulated as:

$$\begin{aligned} &\min \sum_{m=1}^{M} \sqrt{p_m} |\theta_m - 1| \cdot ||\hat{\boldsymbol{w}}^{(m)}||_2 \\ &s.t. \quad \frac{\sum_{m=1}^{M} p_m \xi_m (\gamma^{(m)} + \Delta\gamma^{(m)})}{\sum_{m=1}^{M} p_m (\gamma^{(m)} + \Delta\gamma^{(m)})} \geq \xi \end{aligned} \tag{2}$$

where $\hat{\boldsymbol{w}}^{(m)}$ denotes the weight sub-vector on feature group m solved for the unprioritized SGL model, $\gamma^{(m)} \in \{0, 1\}$

indicates whether any feature in group m is selected in the unprioritized SGL model, $\Delta\gamma^{(m)}$ denotes the change of feature selection on group m after applying priorities. $\Delta\gamma^{(m)} \in \{-1, 0, 1\}$ indicates group de-selection, unchanged and selection, respectively. ξ_m denotes the ratio of visible differences in feature group m.

We show that θ_m can be computed as having the minimal change to enable $\Delta\gamma^{(m)}$ (see the analysis in Section IV-C):

$$\theta_m = \begin{cases} \frac{||S(-\frac{\partial NLL}{\partial \boldsymbol{w}^{(m)}}(\boldsymbol{\hat{w}}), \alpha\lambda)||_2}{(1-\alpha)\lambda\sqrt{p_m}} & |\Delta\gamma^{(m)}| = 1 \\ 1 & \Delta\gamma^{(m)} = 0 \end{cases} \tag{3}$$

Substituting with (3), the optimization problem becomes

$$\min \sum_{m=1}^{M} ||\boldsymbol{\hat{w}}^{(m)}||_2 \cdot |\frac{||S(-\frac{\partial NLL}{\partial \boldsymbol{w}^{(m)}}(\boldsymbol{\hat{w}}), \alpha\lambda)||_2}{(1-\alpha)\lambda} - \sqrt{p_m}| \cdot |\Delta\gamma^{(m)}|$$

$$s.t. \sum_{m=1}^{M} p_m(\xi_m - \xi)\Delta\gamma^{(m)} \geq \sum_{m=1}^{M} p_m(\xi - \xi_m)\gamma^{(m)} \tag{4}$$

This turns out to be a constraint linear programming problem over $\Delta\gamma^{(m)}$, given the weight vector $\boldsymbol{\hat{w}}$ solved for the unprioritized SGL model. We propose a budget optimization algorithm in Algorithm 1 to solve the problem. The idea is to treat the right side of the constraint in (4) as the fixed budget to spend, and the left side terms as investments to meet the budget. The objective in (4) is to minimize the total cost by each investment of $\Delta\gamma^{(m)} \neq 0$. The algorithm sorts all feasible investments by the investment/cost efficiency and spends the budget by this rank until no budget is left.

Algorithm 1: Optimization Algorithm for θ_m.

```
Input  : $\hat{w}$, $\hat{w}^{(m)}$, $p_m$, $\xi$, $\xi_m$, $\alpha$, $\lambda$
Output: $\Delta\gamma^{(m)}, \theta_m$ for $m = 1, \cdots, M$
begin
    $F \leftarrow \emptyset$, $budget \leftarrow 0$
    for $m \leftarrow 1$ to $M$ do        // initialization
        $\gamma^{(m)} \leftarrow \mathbf{1}_{(0,+\infty)}(\hat{w}^{(m)}), \Delta\gamma^{(m)} \leftarrow 0, \theta_m \leftarrow 1$
        $cost_m \leftarrow ||\hat{w}^{(m)}||_2|\frac{||S(-\frac{\partial NLL}{\partial w^{(m)}}(\hat{w}),\alpha\lambda)||_2}{(1-\alpha)\lambda} - \sqrt{p_m}|$
        $invest_m \leftarrow p_m(\xi_m - \xi)$
        $budget \leftarrow budget + p_m(\xi - \xi_m)\gamma^{(m)}$
        if $(\gamma^{(m)} - 0.5) \cdot invest_m < 0$ then
            $F \leftarrow F \cup \{m\}$   // feasible groups
    for $m \in F$ do // invest/cost efficiency
        $efficiency_m \leftarrow \frac{\min(|invest_m|, budget)}{cost_m}$
    Sort $F$ by $efficiency_F$ decreasingly
    while $budget > 0$ && $F \neq \emptyset$ do
    // iterations
        $s = F(1)$     // most efficient group
        $\Delta\gamma^{(s)} \leftarrow 1 - 2 \cdot \gamma^{(s)}$
        $\theta_s \leftarrow \frac{||S(-\frac{\partial NLL}{\partial w^{(s)}}(\hat{w}),\alpha\lambda)||_2}{(1-\alpha)\lambda\sqrt{p_s}}$
        $budget \leftarrow budget - |invest_s|$
        $F \leftarrow F - \{s\}$
end
```

C. Theoretical Analysis

Correctness Analysis. We first discuss how the proposed model regularizes the weight vector towards zero. The objective function in (1) is not differentiable when $w_j = 0$ due to the L1 penalty, so this is a non-smooth optimization problem. However, Equation 1 is clearly convex so that the optimality condition can be obtained through subgradient equations. Denote the objective in (1) by $F(\boldsymbol{w})$, its subgradient $\boldsymbol{g}$ at $\boldsymbol{w_0}$ satisfies

$$F(\boldsymbol{w}) - F(\boldsymbol{w_0}) \geq \boldsymbol{g}^T(\boldsymbol{w} - \boldsymbol{w_0}), \forall \boldsymbol{w} \in \mathcal{R}^p \tag{5}$$

Consider a particular group of features with the weight subvector $\boldsymbol{\hat{w}}^{(m)}$, and the entire weight vector is denoted by $\boldsymbol{\hat{w}}$. This group will be zeroed out when $\boldsymbol{\hat{w}}^{(m)} = 0$ is one of the subgradient satisfying (5). Taking gradients on (1) within the group m, the condition in (5) becomes:

$$\frac{\partial NLL}{\partial \boldsymbol{w}^{(m)}}(\boldsymbol{\hat{w}})\Delta\boldsymbol{\hat{w}}^{(m)} + \alpha\lambda||\Delta\boldsymbol{\hat{w}}^{(m)}||_1 + (1-\alpha)\lambda\sqrt{p_m}\theta_m||\Delta\boldsymbol{\hat{w}}^{(m)}||_2 \geq 0$$

where $\Delta\boldsymbol{\hat{w}}^{(m)}$ denotes an arbitrary small change from $\boldsymbol{\hat{w}}^{(m)} = 0$, NLL is the first term of (1). With a few reductions and analysis, the inequation translates to the form of soft thresholding in lasso, indicating the groupwise sparsity condition:

$$||S(-\frac{\partial NLL}{\partial \boldsymbol{w}^{(m)}}(\boldsymbol{\hat{w}}), \alpha\lambda)||_2 \leq (1-\alpha)\lambda\sqrt{p_m}\theta_m \tag{6}$$

where $(S(z, \alpha\lambda))_j = (|z_j| - \alpha\lambda)_+$. It is clear that θ_m controls whether the features in group m should be deselected entirely, which leads to (3). Using a similar analysis, we can derive the within-group sparsity condition for $\hat{w}_j = 0$ in group m:

$$|\frac{\partial NLL}{\partial w_j}(\boldsymbol{\hat{w}})| \leq \alpha\lambda \tag{7}$$

In determining the priorities for the pSGL model, for feature groups not selected in the unprioritized SGL model ($\boldsymbol{\hat{w}}^{(m)} = 0$), we need to replace $\boldsymbol{\hat{w}}^{(m)}$ in (4) with an estimated cost of selecting the group, denoted by $\boldsymbol{\hat{w}}_+^{(m)}$. Using the subgradient analysis in (6)(7), each $\hat{w}_j$ in $\boldsymbol{\hat{w}}_+^{(m)}$ satisfies:

$$\frac{\partial NLL}{\partial w_j}(\boldsymbol{\hat{w}})sign(\hat{w}_j) + (\alpha - \alpha\sqrt{p_m} + \sqrt{p_m})\lambda = 0 \tag{8}$$

Notice that $\frac{\partial NLL}{\partial w_j}(\boldsymbol{\hat{w}})$ is non-decreasing when $\hat{w}_j$ increases. Newton-Raphson method can be applied to solve (8) for each $\hat{w}_j$, and finally compute $\boldsymbol{\hat{w}}_+^{(m)}$.

Complexity Analysis. On the algorithm scalability, the proposed Algorithm 1 scales well as the problem size grows.

The most costly step is the sorting of the list of feasible groups, which has a complexity of $O(Mlog(M))$. In the case of brain networks, the number of groups grows linearly with the number of regions, having $O(M) \sim O(\sqrt{p})$. Also, the algorithm to solve SGL has a complexity of $O(p)$ [12], linear to the number of features. The total computation complexity holds linear to the number of features. This is validated by experiments in Section VI-D.

V. Implementation Detail

A. Feature Grouping

In the first stage of our solution, edge features are grouped and input to the SGL model. Existing feature categories can be applied as the group, e.g., the functional classification of brain regions. In addition, two clustering methods are supported. The first is the node clustering on the aggregated brain graph of all subjects, again by the mean transfer function. $M-1$ node groups are obtained by optimizing the clustering objective on weight graphs. Then M edge feature groups are derived, $M-1$ groups correspond to the subgraphs by the node clustering and the other group contains all inter-cluster edges. The second method directly clusters edge features by translating the aggregated brain graph into the corresponding line graph, where each node refers to one edge in the brain graph. The edge weight on a line graph is computed by the similarity between adjacent edge features on the brain graph or their weight multiplication [13].

B. Perception Model

After the feature grouping, we need to determine whether each group of features is visible in the comparison by human. Here we introduce the Just-Noticeable Difference (JND) model [14] in the perception theory. The concept of JND is defined as the minimal amount of perception magnitude that something must be changed for human to notice the difference. Formally, given a reference stimulus with value I on certain perception channel, the JND profile, denoted as $JND(I)$, quantifies a minimally increased stimulation $I+JND(I)$, at which just $P\%$ of people can detect changes from the previous stimulation intensity. Normally P takes a value of 50, so that a half of people will sense the change at least as large as JND. By Weber's Law, JND is proportional to the original intensity: $JND(I) = k \cdot I$. The factor k takes a constant value, but varies across different user bases and modalities of the human perception (e.g., sound, vision).

For the scenario of visual comparison, the closest JND model has been proposed on the image processing domain [14]. There are two additional factors except for the intensity difference: (1) background luminance adaptation; (2) spatial masking. In this work, we adopt an extended JND model from the image perception domain to the subgraph-level JND on node-link graphs. On the visual comparison of a subgraph G, each edge is said to be noticeable if its difference between groups is no smaller than $JND(G)$.

$$JND(G) = \beta_0 + \beta_1 \cdot E(G) + \beta_2 \cdot STD(G) \qquad (9)$$

where $E(G)$ denotes the weighted average of edge color saturation by edge length/space, $STD(G)$ denotes the standard deviation of edge color saturations. More detail and the rationale of this perception model is explained in the extended technical report [15]. Note that the model parameters will be calibrated through the user experiment in Section VI-A.

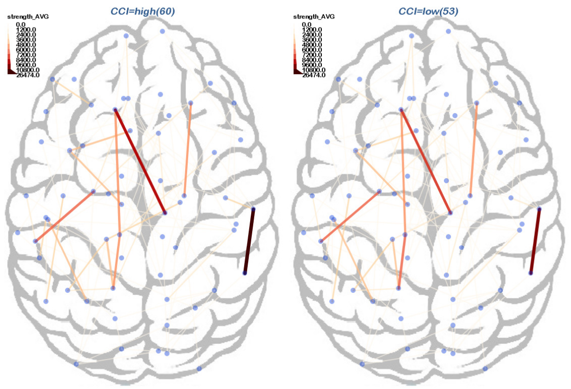


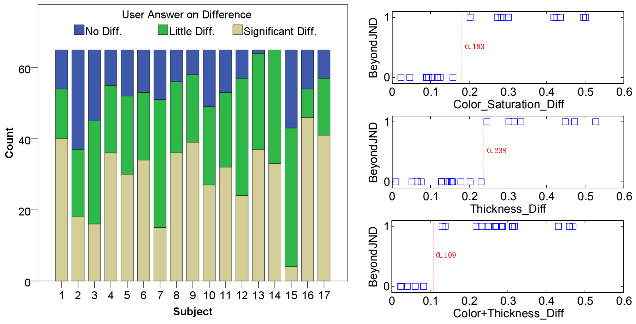
Figure 4. Brain network comparative visualization applying color palette, feature capping and redundant coding mechanisms. 83 features are selected by lasso with a 85.5% prediction accuracy.

C. Model Estimation

In solving SGL and pSGL models with fixed priority parameters, we apply Moreau-Yosida regularization based algorithm in [12]. To determine the sparsity parameter of α and λ, we first try a list of value in $\alpha \in [0,1]$. For each α, the overall sparsity λ takes logarithmically spaced values within the feasible range for nonzero weight vectors. The best λ is determined as the one with the highest prediction accuracy. Note that the prediction accuracy is calculated in a 10-fold cross-validation by a random partition of the data.

D. Visual Design

In complement to the algorithmic framework, we propose a customized visual design for the comparison of brain networks, as shown in Figure 4: (1) Color palette. Beyond the linear mapping from the edge feature to the color saturation, we introduce data binning with 9 sequential color classes. Here the color palette follows the suggestion in ColorBrewer [16], the number of classes is determined by the result in Section VI-A. (2) Feature value capping. In our empirical study, it is found that only a few edge features have noticeable difference (>10.9%) in the comparison. We develop the feature capping method to amplify small differences to be more visible. For example, in our case with a capping value of 10800 (Figure 4), the visible difference threshold is reduced to 1200 from 3000. For edges with weight exceeding this cap, we use a single upper-bounded color to draw, which makes up the augmented 10-color



(a) (b)

Figure 5. The user study results to calibrate the subgraph JND model: (a) Distribution of user's answer on visual difference. (b) 50% JND of three visual coding methods: color saturation, line thickness, color+thickness. The ratio of difference in X axis is measured as the absolute value difference divided by the maximal edge value in its subgraph (the background luminance), therefore these ratios are not uniformly distributed on [0,1].

palette. (3) Redundant coding. By the result in Section VI-A, using both color saturation and line thickness can significantly improve user's performance in visual comparison. This is due to the redundant coding effect that leverages more visual channels to display the difference.

VI. Evaluation

Experiments are designed to answer the following questions: (Q1) Whether the proposed subgraph JND model captures the user's performance in identifying visual differences among brain networks? (Q2) How well does the proposed model perform in optimizing the design objectives, both individually and collectively? (Q3) How does our method scale?

A. JND Experiment

We conducted a controlled user experiment on brain network comparison to estimate the subgraph JND model in (9).

Design. We recruited 17 subjects after they passed the color-blindness test. The experiment followed the within-subject design and each subject entered a total of 65 tasks independently. The first 5 tasks were designed for training and the following 60 tasks were the test phase, divided into 20 tasks for each of three visual difference coding methods: (1) using color saturation, (2) using line thickness, and (3) using the redundant coding on both channels.

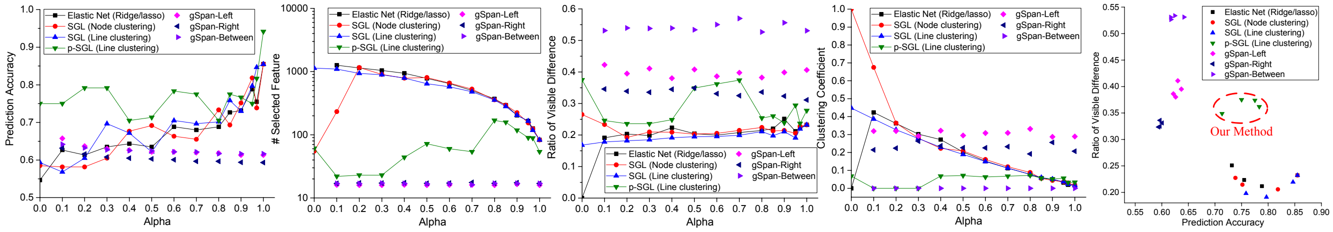
Data and Tasks. In each task, we asked users to compare two views: one with the original brain network data, and the other with planned differences added on the edges of a subgraph. All original views applied the average of 113 brain networks in our data set. The other view with difference was generated like this. First, we randomly chose a nontrivial subgraph with a varying size by the pre-computed graph clustering. Then each edge in this subgraph was selected with some probability (0.5 by default). All selected edges were increased/decreased in feature value by a ratio of their original values. The ratio is uniformly controlled between 5% and 100% (20 samples). One of the three visual coding methods is applied to display the difference. We ensured a balanced design so that the full space of each method can be explored. On each task, users were asked to choose from three difference levels between the two views: (1) No difference, (2) Little difference (random noise), (3) Significant difference. We recorded both user's choice and their completion time.

Results. As shown in Figure 5(a), users indicate 82.7% tasks to have at least little difference and 48.5% with a significant difference. Based on this distribution, we choose significant v.s. non-significant as the boundary for noticeable difference. Also note that user #15 has largely skewed answers from the others, so we have dropped his entries in the analysis. On each of 60 tasks in the testing phase, we check whether there is at least 50% users answering with significant difference. This corresponds to whether the task setting is beyond the JND or not. Then the model in (9) can be fitted with logistic regression. Somehow surprisingly, the estimated model in our scenario is quite simple. As depicted in Figure 5(b), the above/below the JND outcome can be perfectly classified by the ratio of difference on visual channels. The regression analysis obtains a boundary ratio of 0.183 for using color saturation to visualize the difference, 0.238 for using line thickness, and 0.109 for color+line thickness coding. These results demonstrate that the line color coding for visual comparison is better perceived than the line thickness coding, while the redundant coding of both gains the best performance.

B. Performance Comparison

We evaluate the performance of following feature selection methods: lasso (Elastic Net under $\alpha = 1$), ridge regression (Elastic Net under $\alpha \rightarrow 0$) and Elastic Net; sparse group lasso (SGL) under $\theta_m = 1$ with both node and line clustering by edge strength or strength difference between comparative views; prioritized SGL (pSGL) under a ratio of visible difference threshold (ξ) of 0.25; group lasso (SGL under $\alpha = 0$). We focus on the scenario of comparing brain networks between the high CCI group (60 subjects) and the low CCI group (53 subjects). The groupwise sparsity α varies from 0 to 1 to cover the full space of lasso-based methods. The statistical hypothesis testing, which only selects features with significant differences ($p < 0.05$), predicts the CCI class even worse than a null model (53.1% accuracy), so we drop this model in the comparison. For SGL models with different clustering algorithms, we choose the node/line clustering with the best prediction accuracy.

We also compare with the frequent graph mining methods in the experiment. In particular, we choose one of the most popular methods, gSpan [17]. In our scenario, we take a two-step approach. First, frequent subgraphs among all brain networks are generated as candidates using gSpan. Second, the extracted subgraphs are treated as the input features of a



(a) Prediction Accuracy (b) #Selected Feature (c) Visibility of Difference (d) Clustering Coefficient (e) Accuracy v.s. Visibility

Figure 6. Performance comparison of feature selection methods on design objectives.

standard logistic regression to train a binary classifier for the two CCI classes. It is very time consuming to run gSpan on the entire brain networks, largely due to their high densities (see Section II). To address this issue, we divide the 70-node brain network into three parts, left-brain, right-brain and the left-right connection subgraphs. gSpan is executed separately on each set of regional networks.

All methods are compared on the objectives defined in Section III. Figure 6 summarizes the result on discriminative power, sparsity, visibility and grouping effect. As shown in Figure 6(a), all feature selection methods achieve better prediction accuracies with a larger α. This is because in our setting, we have $N \ll p$. Thus, as we increase α and stress more on the overall sparsity, fewer features are selected (Figure 6(b)). Consequently, the prediction performance is improved thanks to the less overfitting. Notice that, SGL with an appropriate clustering could outperform Elastic Net without an explicit grouping. The proposed pSGL model further improves the prediction, mainly because it selects an even smaller number of discriminative features and thus reduces the overfitting. In the extreme case, gSpan would only select a single small subgraph, which degrades the prediction accuracy.

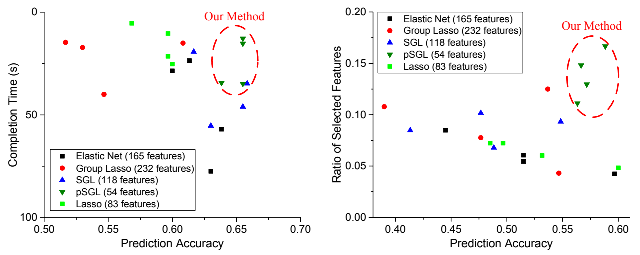
In terms of the visibility of difference, as illustrated in Figure 6(c), the proposed pSGL model raises the visibility to a level close to or above the specified threshold (0.25) and is better than most of the lasso-based methods. The gSpan algorithms achieve the best visibility, mainly because only a small number of features are selected. As for the clustering coefficient, Figure 6(d) shows that all lasso-based methods have a better clustering effect with a smaller α, which is consistent with their heuristics. The proposed pSGL model leads to a smaller clustering coefficient because it selects fewer edge features. Nonetheless, with a medium α (0.4$\sim$0.7), the pSGL model still has a better clustering than other methods with $\alpha > 0.9$, when all methods achieve a comparable prediction accuracy at 80%. Frequent subgraph mining algorithms produce clustered subgraphs by their design.

In summary, the proposed pSGL model achieves the best overall performance on the four design objectives of our problem. Figure 6(e) illustrates the trade-off between the prediction accuracy and the visibility of difference on a scatterplot. Four representative plots of the pSGL model lie on the upper-right corner, indicating a balance between the prediction accuracy and the visibility. On the other hand, the existing lasso methods stay at the lower-right corner, suffering from poor visibility for comparison; and frequent subgraph mining methods stay at upper-left corner, falling short on the prediction accuracy.

Our results are better demonstrated with the visual comparison in Figure 7 (the result by lasso is given in Figure 4). By lasso, the best prediction accuracy of 85.3% is reached, but the selected features are scattered out and hard to compare by humans. Elastic Net (Figure 7(a)) and SGL with line clustering (Figure 7(c)) both obtain the best prediction under $\alpha \rightarrow 1$. The selected features are more clustered, but still the comparative pattern is not significant for humans to interpret. Group lasso with node clustering (Figure 7(b)) shows perfectly clustered view, however, the prediction accuracy is poor (58.2%) and there are too many features to compare. The result by the proposed pSGL model ($\alpha = 0.7$ for the best visibility) is show in Figure 7(d). Our method extracts more focused, clustered and visible patterns for the human interpretation, in the meanwhile producing a good prediction accuracy (77.5%). We can infer that the connections of region #64 (rh-superiorfrontal) and #39 (rh-caudalmiddlefrontal) are important for the CCI difference. There is strong accordance to our findings in neuroscience literature. The superior frontal region is involved in self-awareness [18] while there is theory that self-awareness strongly influences creativity [19]. At a higher level, both regions of #64 and #39 are in the right hemisphere, and in Figure 7(d), the high CCI group has stronger connections than the low CCI group between #64/#39 and several regions in the left hemisphere. This difference can be further augmented by introducing binary edge filters that hide weak connections below a threshold. Figure 7(e)(f) are the consequences filtering over Figure 7(d). Only 12 and 8 features are kept in the high CCI group while much less features stay in the low CCI group. On the graph mining algorithms (Figure 7(g)(h)), visual differences can be perceived, though they are not as discriminative as those from our pSGL model.

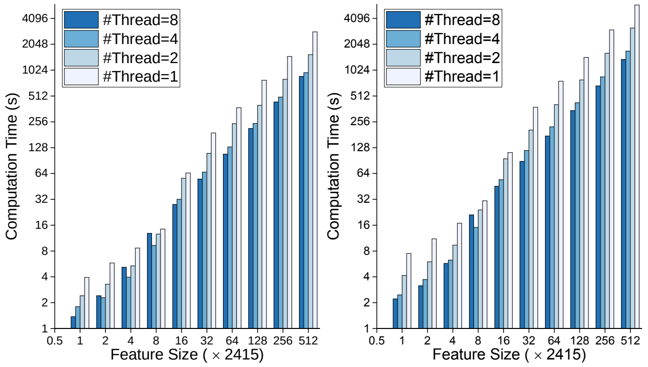
C. User Studies

Design. We follow up with a controlled user study to evaluate the user performance of feature selection methods. 12 subjects were recruited, all with basic knowledge of



(a) Accuracy v.s. Completion Time (b) Accuracy v.s. Selected Features

Figure 8. User study result comparing feature selection methods.



(a) SGL without Priorities (b) pSGL with Priority Optimization

Figure 9. The algorithm scalability when the feature size is increased exponentially. X/Y axis in both figures are under the same log scale.

graph and network. Each subject was required to complete 6 tasks, corresponding to 5 visualization results in Figure 4 and Figure 7(a-d), and one sample task at the beginning for training. On each task, the subject was asked to select all edges that have a significant difference between the comparative view. They were instructed to work in a best-effort manner. We recorded all the edges they selected and the time for completion. By the end of the study, we input the user selected features into a standard logistic regression model and calculate the prediction accuracy for each subject×task setting.

Result. Figure 8(a) presents the performance of top subjects finishing with best prediction accuracies over each feature selection method. The scatterplot shows the trade-off between the prediction accuracy and the completion time, corresponding to the *model effectiveness* for users. Figure 8(b) depicts the performance of top subjects selecting most features over existing models (by the ratio of manually selected features in those selected by models/algorithms). This corresponds to the *model efficiency* for users. In both figures, it is shown that pSGL lies on the upper-right area, i.e., our method achieves the best performance in both user effectiveness and efficiency.

D. Scalability

We test the scalability of the proposed method by synthesizing larger brain networks. To upgrade the network into $K \times 2415$ features, $(\lceil\sqrt{K}\rceil - 1)$ new dummy nodes are replicated from each of 70 source network nodes/regions. Between each pair of new nodes, an edge is created with a probability of $\frac{K-1}{(\lceil\sqrt{K}\rceil-1)^2}$. The edge weight is determined by the corresponding feature in the 70-region network, with white noise in a 5% range. The number of feature groups grows $\lceil\sqrt{K}\rceil$ times, linear to the number of network nodes.

Ten synthesized brain networks are used in the test, $K = 2^0, 2^1, \cdots, 2^9$ times of the original network, until reaching a million edge features. Two algorithms are applied, the first with the baseline solver in [12] for the SGL model, and the other is our proposed algorithm for the pSGL model. All experiments are carried out on a commodity desktop as the server and 8 external Intel Xeon 2.67GHz computing nodes running Matlab parallel computing toolbox.

Results are summarized in Figure 9 in the uniform log-log scale. By both algorithms, the computation time grows linearly with the number of features. This is demonstrated by the slopes of time bars in Figure 9(a)(b), which are close to one in each setting. Compared with the baseline algorithm in Figure 9(a), the priority optimization increases the computation time by 30% $\sim$ 110%. This corresponds well with our solution pipeline that solves the SGL model twice. The latter solver can be faster when the priorities are optimized to only select feature groups that increase the visibility ratio towards the threshold.

VII. Related Work

Brain Network Analysis emerges as a compelling topic in data mining research due to the maturation of non-invasive neuroimaging techniques [20]. The raw neuroimaging data is modeled as a high-order tensor, e.g., by three-dimensional image and time. On these tensor data, fundamental problems are defined [21], including the node discovery that detects brain areas with coordinated activities, edge discovery that creates weighted relationship between nodes, and the verification of network strength. Both the tensor and brain networks can be trained by learning methods (tensor decomposition, feature selection) to infer the relationship with certain outcomes, e.g., Alzheimer's Disease [22][23]. In another thread, studies on the subgraph extraction and analysis on brain networks are also popular, where the challenge lies in the modeling and subgraph mining of uncertain brain networks [24][6]. Though solid progress has been made on this area, the problem of jointly optimizing data mining and perception-level objectives has never been studied before.

Feature Selection algorithms are widely applied in the study of bioinformatic data, because of its tendency to carry much more features (e.g. genes, biological pathway) than the data sample. On regression analysis, the regularization-based sparse learning has attracted intensive studies for decades. The seminal work by Tibshirani [4] introduced the lasso (aka L1 regularization), which adds the L1 norm penalty to encourage zero weights for sparsity. In many scenarios, lasso can be too aggressive to identify correlated features. Therefore, Elastic Net [5] was proposed to exploit the grouping effect in feature selection, which applies a combination of L1 and L2 penalty. With a similar goal, group lasso [9] was introduced, which allows specifying the group of correlated features. The latest work on the sparse group lasso [10] further combined the group lasso with L1 penalty, to provide flexibility in controlling both groupwise and within-group sparsities. On graph analysis, the graph-guided fused lasso [25] was designed to cluster selected features together. Compared to the existing work on feature selection, we consider the novel perspective of human perception and propose a new model in this objective.

Network Visualization has been well-studied to display networks and graphs [26]. Due to the unique characteristic of brain networks (e.g., high density), existing visualization designs are often inadequate for brain networks (e.g., Figure 1(a-b)). Moreover, the task of visual comparison on brain networks is largely unexplored. The work in [3] might be one of the sparse literature on this subject. They studied the effectiveness of two visual representations on weighted graph comparison. This work is more a design study and they do not consider data mining objectives.

VIII. Conclusions

This paper presents BrainQuest, an integrated mining and visualization framework for the comparison of brain networks. We consider statistical and perception constraints on: (1) discriminative power; (2) sparsity; (3) grouping effect; (4) visibility of differences. BrainQuest achieves these goals by a multi-objective feature selection model. Notably, the new constraint on perception is calibrated through user experiment and optimized by a novel usage of the priority criterion on lasso-based models. We propose scalable algorithms to implement the framework and conduct comprehensive evaluations in both quantitative experiment and user study. The mining result corresponds well to neuroscience findings, which demonstrates our success.

Acknowledgment

The dataset is publicly available at OpenConnectome [2]. The source code can be found here [15].

References

[1] O. Sporns, *Networks of the Brain*. MIT press, 2011.

[2] "Open connectome, http://openconnectomeproject.org."

[3] B. Alper, B. Bach, N. Henry Riche, T. Isenberg, and J.-D. Fekete, "Weighted graph comparison techniques for brain connectivity analysis," in *CHI*, 2013, pp. 483–492.

[4] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B*, pp. 267–288, 1996.

[5] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[6] X. Kong, P. S. Yu, X. Wang, and A. B. Ragin, "Discriminative feature selection for uncertain graph classification," in *SDM*, 2013, pp. 82–93.

[7] W. R. Gray, J. A. Bogovic, J. T. Vogelstein, B. A. Landman, J. L. Prince, and R. Vogelstein, "Magnetic resonance connectome automated pipeline: an overview," *IEEE Pulse*, vol. 3, no. 2, pp. 42–48, 2012.

[8] R. S. Desikan, F. Ségonne, B. Fischl, B. T. Quinn, B. C. Dickerson, D. Blacker, R. L. Buckner *et al.*, "An automated labeling system for subdividing the human cerebral cortex on mri scans into gyral based regions of interest," *Neuroimage*, vol. 31, no. 3, pp. 968–980, 2006.

[9] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society: Series B*, vol. 68, no. 1, pp. 49–67, 2006.

[10] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013.

[11] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[12] J. Liu and J. Ye, "Moreau-yosida regularization for grouped tree structure learning," in *NIPS*, 2010, pp. 1459–1467.

[13] U. Kang, S. Papadimitriou, J. Sun, and H. Tong, "Centralities in large networks: Algorithms and observations," in *SDM*, 2011, pp. 119–130.

[14] C.-H. Chou and Y.-C. Li, "A perceptually tuned subband image coder based on the measure of just-noticeable- distortion profile," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 467–476, 1995.

[15] "BrainQuest project." [Online]. Available: http://lcs.ios.ac.cn/%7eshil/projects/BrainQuest/

[16] M. Harrower and C. A. Brewer, "Colorbrewer. org: an online tool for selecting colour schemes for maps," *The Cartographic Journal*, vol. 40, no. 1, pp. 27–37, 2003.

[17] X. Yan and J. Han, "gspan: Graph-based substructure pattern mining," in *ICDM*, 2002, pp. 721–724.

[18] "Superior frontal gyrus. http://en.wikipedia.org/wiki/superior_frontal_gyrus."

[19] P. J. Silvia and A. G. Phillips, "Self-awareness, self-evaluation, and creativity," *Personality and Social Psychology Bulletin*, vol. 30, no. 8, pp. 1009–1017, 2004.

[20] X. Kong and P. S. Yu, "Brain network analysis: a data mining perspective," *SIGKDD Explorations*, vol. 15, no. 2, pp. 30–38, 2014.

[21] I. Davidson, S. Gilpin, O. Carmichael, and P. Walker, "Network discovery via constrained tensor analysis of fmri data," in *KDD*, 2013, pp. 194–202.

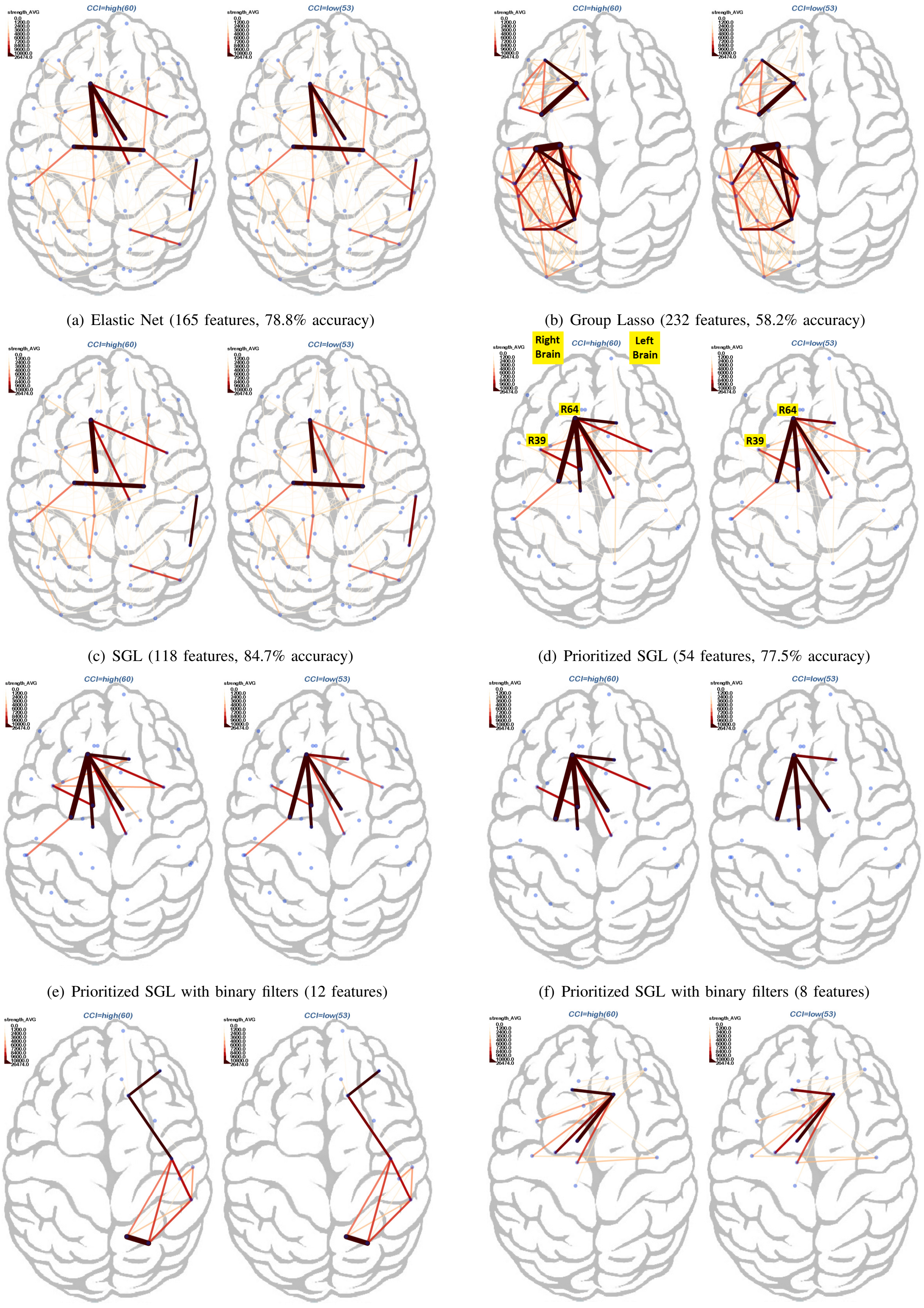
[22] L. Sun, R. Patel, J. Liu, K. Chen, T. Wu, J. Li, E. Reiman, and J. Ye, "Mining brain region connectivity for alzheimer's disease study via sparse inverse covariance estimation," in *KDD*, 2009, pp. 1335–1344.

[23] S. Huang, J. Li, J. Ye, A. Fleisher, K. Chen, T. Wu, and E. Reiman, "Brain effective connectivity modeling for alzheimer's disease by sparse gaussian bayesian network," in *KDD*, 2011, pp. 931–939.

[24] Z. Zou, H. Gao, and J. Li, "Discovering frequent subgraphs over uncertain graph databases under probabilistic semantics," in *KDD*, 2010, pp. 633–642.

[25] X. Chen, S. Kim, Q. Lin, J. G. Carbonell, and E. P. Xing, "Graph-structured multi-task regression and an efficient optimization method for general fused lasso," *arXiv preprint:1005.3579*, 2010.

[26] G. D. Battista, P. Eades, R. Tamassia, and I. G. Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall PTR, 1998.



(a) Elastic Net (165 features, 78.8% accuracy)

(b) Group Lasso (232 features, 58.2% accuracy)

(c) SGL (118 features, 84.7% accuracy)

(d) Prioritized SGL (54 features, 77.5% accuracy)

(e) Prioritized SGL with binary filters (12 features)

(f) Prioritized SGL with binary filters (8 features)

(g) gSpan on left brain (16 features, 62.2% accuracy)

(h) gSpan on left-right connections (17 features, 63.3% accuracy)

Figure 7. Visual comparison of feature selection results (best viewed in color and high resolution).