RESPONSE TO REVIEWERS

We thank both reviewers and the editor for their helpful and insightful comments. We respond to each of the comments below. For ease, we italicize the reviewer comments. Quotes from the revision are in red. We are grateful to have the opportunity to resubmit this new and improved version of our manuscript.

Reviewer 1

- *The authors rule out many potential methods for this problem based on the disered criteria of "intepretability". However, they conclude by noting a weakness of their method (based on simulated data) is the unreliability of intepreting the results with respect to specific edges. Therefore, there is arguably an inconsistency between what is promised and what has been achieved.*
  We have tried to clarify in two ways. First, we more clearly explain the desideratum in the §1:
  For prognostic and diagnostic purposes, merely being able to differentiate groups of brain-graphs from one another is sufficient. However, for treatment, it is desirable to know which vertices and/or edges are malfunctioning, such that therapy can be targeted to those locations. This is the motivating application for our work.
  Unfortunately, this desideratum is not achieved with such a small sample size and data distributed according to our synthetic data analysis. We have highlighted this by adding the below line to §5.1.1:
  Despite our stated desideratum of interpretability of the resulting classifier in terms of correctly identifying the signal-edges and vertices, for data sampled from this assumed distribution, sample sizes of < 50 seem to be insufficient.
- *Moreover, from a general point of view, it would greatly enhance the value of the paper if comparison to some "less intepretable" methods were offered; in general, one might imagine practitioners making some trade-off between performance and interpretability, and there is an opportunity here to offer valuable data to aid in that decision. It would be useful to know how the method proposed in the paper compares to some standard baseline other than Naive Bayes, such as those mentioned in the third paragraph in the introduction. Another simple baseline would be logistic regression, using edge/no-edge between vertices as the features, with L1 regularization on the weights, which should be quite similar to the incoherent estimator.*
  We have now compared the performance of our classifiers both in simulation and on the real data. §4.3 now includes the following text:
  An important aspect of any algorithm is compute time, both of training and testing. The signal subgraph classifiers that we developed are very fast. Computations essentially amount to computing a test-statistic for all $|\mathcal{E}|$ edges, then sorting them. The parameter estimates of the likelihood and prior terms come directly from the same test-statistics used to obtain the significance of each edge. Thus, obtaining those estimates amounts to essentially computing a mean. On the other hand, the lasso classifier, which yields *worse* signal detection and misclassification rates than both our classifiers, requires an iterative algorithm for each value on the hyper-parameter path [20]. Despite that efficient computational schemes have been developed for searching the whole regularization path [21], such iterative algorithms should be much slower than our classifiers.
  Indeed, the lower panel of Figure 3 demonstrates that our MATLAB implementation of the signal subgraph classifiers are approximately 10 times faster than MATLAB's lasso implementation. All the results shown in Figure 3 include errorbars computed from 100 trials, each with 100 held-out samples, demonstrating that for these simulation parameters, the differences are highly significant. Although the quantitative results may vary for different implementations and different parameter settings, our expectation is that the qualitative results should be consistent. Thus, because our classifiers have lower risk, better signal identification, and run an order of magnitude faster than the standard, we do not consider lasso in further simulations.
  Moreover, we added Table 1, which compares the performance of a variety of "interpretable" and "less interpretable" classifiers on real data, and the below text:
  We next compare the performance of our classifiers on this MR connectome sex classification data set to several other classifiers. First, a standard parametric classifier: lasso. We chose the regularization parameter via a 10-fold cross-validation. Second, a non-parametric (distribution free) classifier: k_n-nearest neighbor (kNN), which operates directly on graphs [26]. This kNN classifier uses the Frobenius norm distance metric. We tried all $k \in [n]$ and simply report the best performance. The universal consistency of this kNN classifier is useful in assessing the algorithm complexity supported by this data. In particular, given enough samples, kNN will achieve optimal performance. Less than optimal performance therefore indicates that the sample size is not sufficiently large for this kNN classifier. Third, a graph invariant based classifier. We computed six graph invariants for each graph: size, max degree, scan statistic, number of triangles,

clustering coefficient, and average path length, normalized each to have zero mean and unit variance, and then used a kNN with ℓ_2 distance metric on the invariants. These particular invariants were chosen based on their desirable statistical properties [27–29].
Despite the small sample size, Table 1 demonstrates that the signal-subgraph classifier is significantly better than all the others, as assessed via a one-sided McNemar's test.

- *The hyper-parameter section is unclear, and therefore the experimental results are unconvincing. Specifically, on the non-synthetic experiment for classification of sex based on MR connectome, cross-validated misclassification was used to judge performance.*
  We have added "§2.4.5 Hyper-Parameter Selection" to clarify this issue.
  The signal-subgraph estimators require specifying the number of signal-edges s, as well as the number of signal-vertices m for the coherent classifier. In both cases, the number of possible values of finite. In particular, $s \in [d_V]$ and $m \in [V]$. Thus, to select the best hyper-parameters we implement cross-validation procedures (see Section 2.5.2 for details), iterating over $(s, m) \in \vec{s} \times \vec{m} \subseteq [d_V] \times [V]$. Note that when $m = V$, the coherent signal subgraph estimator reduces to the incoherent signal subgraph estimator. For all simulated data, we compare hyper-parameter performance via a training and held-out set. For the real data application, we decided to use a leave-one-out cross-validation procedure due to the small sample size.
- *(By the way, it is only mentioned in the abstract and Figure 6 caption that the task is actually gender classification; this could be made more clear in the text.)*
  Thank you, we have added the word "sex" all over the manuscript, in particular, in the title of "§5 MR Connectome Sex Classification", and throughout the following paragraphs.
- *These numbers only reflect performance when the hyper-parameters are set optimally for the data being tested, therefore it's not surprising that the more complex models did better. To compute how well the model would perform when the hyper-parameters are not known, they must be set on some held-out data, and then applied to the test data.*
  Thank you for pointing out this mistake. We no longer compare the coherent and incoherent classifier using McNemar's test, as the assumptions underlying that test are badly inaccurate.
- *In the top panel of Figure 4, for small numbers of training samples, the coherent estimator has a larger miss rate than the incoherent estimator, thus it is stated that to pick the right estimator, the number of samples must be known. However, how many samples were used to generate the plot? Is there a confidence interval, and is the difference between estimators when the number of training samples is small statistically significant?*
  We now ran this simulation for 200 trials to estimate the mean and standard error in those plots. Both the main text and caption now states this clearly. The errorbars indicate the the differences are likely significant.
  The top panel shows the mean and standard error of the missed-edge rate—the fraction of edges incorrectly identified—averaged over 200 trials.
- *In most applications, obtaining subgraph information (connectivity) about specific vertices is easier than obtainining information about all vertices. Can this procedure be used to provide a principled method to estimate important vertices and then focus additional data acquisition selectively to maximize information regarding some class identity?*
  Thank you for pointing out this omission! We have added a sentence to the discussion mentioning our work on this interesting and very related topic:
  Specifically, extension to situations for which none of the vertices are labeled [38; 39], only some subset of vertices are labeled [40; 41], or data are otherwise errorfully observed [42], are all avenues of future investigation.
- *It would be helpful if the authors could comment more on the issue of assuming the vertices are labeled. In many important applications, vertices will not be labeled and a vertex matching procedure will have to be performed. Is this truly a distinct problem, or would it make more sense in such a situation to think of both matching & classification being performed simultaneously?*
  Again, thank you for asking this particular question. We have added the following sentences to the Introduction:
  The field of *connectomics* (the study of brain-graphs), however, is ripe with many examples of brain-graphs with vertex labels. In invertebrate brain-graphs, for example, often each neuron is named, such that one can compare neurons across individuals of the same species [7]. In vertebrate neurobiology, while neurons are rarely named, "neuron types" [8] and neuroanatomical regions [9] are named.
  Indeed, we have two other manuscript pending at PAMI right now addressing the issue of how to classify when vertex labels are unknown [38; 39].

Reviewer 2

- *The application to real data does not include a comparison with known truth or a gold standard and no comparison with other state-of-the art graph classification methods [b].*
  We have now rectified this previous omission by including Table 1 and associated text in §5. The text is pasted above in response to Reviewer 1.
- *Technical Correctness*
  The model that you propose seems to be a very minor variant of the model we have proposed. In particular, you assume that $\mathcal{S}$, the set of edges in the signal subgraph, is a *random (latent) variable*. On the other hand, we assume that $\mathcal{S}$ is a *parameter* to be estimated. We have discussed the utility of this complementary "Bayesian" view of our proposed model and appreciate many of its benefits. Indeed, we believe that in many situations, assuming that $\mathcal{S}$ is a random variable and assuming a prior over its distribution will likely yield improvements in error, although at the cost of computation. We hope to explore this extension in future work. To clarify the distinction, we have now added the following sentence to §2.2:
  While it may be natural to treat $\mathcal{S}$ as a prior, we treat it as a parameter of the model; the constraints, s and m, are considered *hyper-parameters*.
- *Due to the introduction of the random variable S and the prior P(S), this model is more specific than the framework of hybrid generative-discriminative models in [a].*
  Thank you for pointing out the related work on hybrid generative-discriminative models. We have removed some of the text in §2.2 and added the citation you mentioned.
- *It would be appropriate to devote one section to related work, e.g. some of the methods described in [b]. Without any references except [1,2] the following statements remain unjustified: This graph-model based approach is qualitatively different from most previous approaches which utilize only vertex labels or graph structure*
  We have now added several references to support this claim:
  In the latter case, computing a set of graph invariants (such as clustering coefficient), and then classifying using only these invariants ignores vertex labels [1; 5; 6].
- *Nonetheless, the classifiers described below still significantly outperform the benchmarks.*
  We have now added a table comparing the results of our coherent classifier with several other state-of-the-art algorithms, as discussed above.
- *Due to the existence of the encoding mentioned above, the following statement should be avoided.*
  Because the encoding mentioned above is a slightly "Bayesianized" variant of our method, as mentioned above, we have decided to keep this sentence in the manuscript.
- *to get rid of the passages that recite textbook knowledge and instead focus on the original contributions. The additional space should be devoted to an informal, intuitive introduction of these contributions and a discussion of related work.*
  We have followed this advice. In addition to the additions, we have removed several paragraphs of text.
- *to drastically reduce the amount of mathematical rigor and to use a more common mathematical notation where necessary, e.g. only (1) from this review and possibly the graphical model in Fig. 1.*
  We have removed much of the extraneous text, and clarified the mathematics.
- *to get rid of Fig. 1 and shorten Section 2.4 as far as it refers to this figure. It is sufficient to say that the number of all possible graphs with n nodes, even with the constraining properties, is too large to be explored by exhaustive search.*
  We added this figure only after presenting this material informally at a number of invited talks, and discovered that people's intuition about the size of the search spaces was way off.
- *to start with a more pronounced and intuitive motivation for graph clas- sification in the context of MR connectomics.*
  We have added a paragraph in the introduction to this end:
  The field of *connectomics* (the study of brain-graphs), however, is ripe with many examples of brain-graphs with vertex labels. In invertebrate brain-graphs, for example, often each neuron is named, such that one can compare neurons across individuals of the same species [7]. In vertebrate neurobiology, while neurons are rarely named, "neuron types" [8] and neuroanatomical regions [9] are named. Moreover, a widely held view is that many psychiatric issues are fundamentally "connectopathies" [10; 11]. For prognostic and diagnostic purposes, merely being able to differentiate groups of brain-graphs from one another is sufficient. However, for treatment, it is desirable to know which vertices and/or edges are malfunctioning, such that therapy can be targeted to those locations. This is the motivating application for our work.
- *to use the term class-conditional subgraph consistently instead of signal subgraph. The latter is potentially misleading because it does not refer to a signal as in signal processing as some readers of this journal will assume but to signalling neurons.*

We have given extensive thought to this change. In the end, we like signal subgraph for a number of reasons. First, it *does* refer to a signal, as in signal processing. The point is that the signal is context dependent, in this case, the signal is which class are you in. In other words, they are the signaling neurons: they signal which class the subject is in. There is a propensity within neurosciences to talk about "signal" without first specifying an specific exploitation task, as if "signal" has some meaning ex nihilo. Our choice of using "signal subgraph" is, in part, to emphasize that signal is (necessarily) defined by exploitation task.

- *to replace the term coherent graph that has many connotations unrelated to its meaning here and should therefore be avoided.*
  We have tried to motivate and clarify our use of the word "coherent" in a the §1:
  Moreover, borrowing a term from the compressive sensing literature [3; 4], we are interested in learning to what extent this signal is *coherent*; that is, to what extent are the signal-subgraph edges incident to a relatively small set of vertices. In other words, if the signal is sparse in the edges, then the signal-subgraph is incoherent, if it is also sparse in the vertices, then the signal-subgraph is coherent (we formally define these notions below).
  And in §6:
  Our signal-subgraph classifiers represent somewhat of a departure from previous work. Most graph classification algorithms come from the "structural pattern recognition" school of thought [1], lacking an explicit statistical model and associated provable properties. On the other hand, most work on "statistical pattern recognition" begins by assuming the data to be classified are Euclidean vectors [31]. Our work is a unification of the two. Moreover, because the sufficient statistics are essentially encoded in a matrix, our work can be related to recent developments in matrix decompositions. For example, sparse and low-rank matrix decompositions are close in spirit to our coherent signal subgraph estimators [32–34]. Note, however, that our coherent estimator is robust to signal-vertices having a subset of its edges highly non-significant; that is, the coherent signal-subgraph estimator can be thought of as a *local* sparse and low-rank decomposition.
- *Moreover, it is sufficient to say, (i) instead of Section 2.4.1.1 (Section 2.4.1.2) that a threshold c is selected minimally such that $>= s$ edges have a significance level $< c$ ($>= s$ nodes exists such that each has an incident edge with a significance level $< c$). (ii) instead of 3.1 that the standard methods for estimating the terms of the model are consistent.*
  We agree that the text explained the signal-subgraph estimators was poor. We have revised these sections extensively. Please see text for details.
- *Examples of textbook style:....*
  We have removed or revised each of these examples.
- *Since the model is defined for undirected graphs...*
  The model is defined for both directed and undirected graphs. We have now clarified in §2.2:
  Fourth, although the following theory and algorithms are valid for both directed and undirected graphs, for concreteness, assume that the graphs are *simple* graphs; that is, undirected, with binary edges, and lacking (self-) loops (so $\mathcal{E} = \binom{V}{2}$).