Interactive dependency visualizer: Understanding pairwise variable relationships, applied to single-cell RNA-seq data

Kevin Lin*, Taewan Kim†

*University of Pennsylvania, †University of Chicago

Abstract

Pairwise dependency graphs are useful tools in data science that summarize which pairs of variables among d variables are dependent, but it is difficult to choose a particular estimator for this task amongst so many possible estimators. This paper describes an interactive model diagnostics and visualization system to efficiently explore pairwise relationships in the dataset via scatter plots. By doing so, the scientist can make an informed decision on how to select the most appropriate method. First, our system learns how the scientist interprets whether or not two variables in the dataset are dependent by having her provide labels for a small number of scatter plots, and then automatically constructs a pairwise dependency graph based on these learned preferences. By comparing this pairwise dependency graph with those numerically estimated, the system can inform the scientist which estimator is best suited for the dataset at hand. We design each component of this system to be interpretable so this system provide evidence to explain why one estimator was chosen over another. We apply our system to illustrate the broad variety of bivariate relations among genes when analyzing certain immune cell types based on single-cell RNA-seq data. We also demonstrate how it is difficult to chose the appropriate dependency statistic a priori without having visually understood the types of patterns within a dataset, and how our visualization system can help scientists make informed decisions on how to study gene co-expression networks across different cell types.

1 Introduction

As data scientists and computational biologists develop increasingly nuanced numerical methods to tackle larger and more complex datasets, it becomes harder to decide when to use one method over another for the dataset at hand. Since each method potentially gives a drastically different result, it is important to use *model diagnostics* to assess how well the chosen method fits the current dataset. Specifically, we refer to model diagnostics as the "open-ended search for structure not captured by the fitted model" using graphical displays (Buja et al., 2009). While more and more statistical goodness-of-fits tests are developed to accompany the numerical methods, previous studies in lineup tests have found that graphical systems can overcome the limitations of these tests (Wickham et al., 2010):

> "Traditional statistical tests are well studied, well-formulated and work best when data is well-behaved, following a known distribution in relatively simple scenarios. But ... traditional statistical tests do not cover all of the complexities that arise when exploring data."

We continue this line of work of developing graphical systems by applying the ideas of previous systems to the task of estimating *pairwise dependency graphs*. Assume we observe n i.i.d. samples of a d-dimensional random variable $X = (X_1, \ldots, X_d)$. The pairwise dependency graph we want to estimate is defined as $G = (V, E)$ with vertices $V = \{1, \ldots, d\}$ and edge set E such that

$$(i, j) \notin E \iff X_i \text{ is independent of } X_j.$$

These graphs provide concise summaries of the relationship between the d variables. In our paper, we focus on understanding the dependency graphs among cells of a particular cell type from single-cell RNA-seq data. In practice, there are many ways to estimate these graphs in practice. For example, there are many ways to quantify the dependence between two variables, of which we mention a few:

- **Pearson correlation**: Perhaps the common measurement of dependence, Pearson correlation measures the linear dependence between X_i and X_j. Its population quantity is defined by
$$\frac{\text{Covariance}(X_i, X_j)}{\sqrt{\text{Var}(X_i)}\sqrt{\text{Var}(X_j)}}.$$
While a pair of independent variables must have a Pearson correlation of 0, the converse is not true.
- **Distance correlation**: Distance correlation, introduced in Székely et al. (2007), is related to how far apart the product of the marginal characteristic functions, $\phi_{X_i}\phi_{X_j}$, is from the joint characteristic function $\phi_{X_iX_j}$. Distance correlation equals 0 in population if and only if X_i and X_j are independent.
- **Mutual information**: Mutual information quantifies how much information is shared between X_i and X_j. Its population quantity is defined by
$$\int\int f_{X_i,X_j}(x_i, x_j) \log\Big(\frac{f_{X_i,X_j}(x_i, x_j)}{f_{X_i}(x_i) f_{X_j}(x_j)}\Big) dx_i dx_j.$$

Mutual information equals 0 in population if and only if X_i and X_j are independent.

We consider many other statistics designed to measure the relation between two random variables, such as Spearman's correlation, Kendall's τ and it's modified version τ^* (Bergsma and Dassios, 2014), Maximal Information Coefficient (MIC), normalized mutual information, Hoeffding's D (Hoeffding, 1994), Hilbert-Schmidt Independence Criterion (Gretton et al., 2005), Blomqvist's beta (Bukovšek et al., 2019), and Chatterjee's ξ (Chatterjee, 2021). For any one of the aforementioned statistics, the user can form a graph G by thresholding the dependency measure, using a significance test, etc. Likely, a different choice of statistic can yield a (potentially dramatic) different estimate of G.

There has been a flurry of statistical work describing the qualities of an "ideal" statistic of dependency between two random variables, and countless benchmarking papers have shown how each statistic is most powerful for a different family of a bivariate distributions (de Siqueira Santos et al., 2014; Deb et al., 2020; Cao and Bickel, 2020). Therefore, we operate under the belief that no method is universally most suitable for all datasets (Madigan et al., 2014). However, can the scientist learn which method is the most appropriate for her current dataset? We take the position in this paper that visual statistical methods (Buja et al., 2009) provide the key ingredient to answer this question. By showing a practitioner an carefully-crafted lineup of bivariate scatter plots from the dataset in question, our system can learn from her subjective choices what qualifies as "dependent random variables" and apply this rule to learn the an appropriately-tailor dependency statistic to the remainder of the dataset.

In this paper, we describe a visualization system to understand the gene coordination patterns based on G for certain immune cells using on single-cell RNA-seq data. Here, G is often called the *gene co-expression network*, and gene coordination refers to how certain groups of genes have expression that rise or fall in-sync with one another. Specifically, we analyze the CD8$^+$ effector T-cells from Stuart et al. (2019), an important cell-type that aids the immune system to acquire long-term memory to pathogens (Martin and Badovinac, 2018). There have been numerous works developing new methodology to estimate the cell-type specific gene co-expression network G for specifically single-cell RNA-seq data (Dai et al., 2019; Wang et al., 2021). Our work offers a complementary viewpoint on why estimating the dependency between two genes from single-cell data requires new methods – we demonstrate that there are many bivariate relations in single-cell data that many of the aforementioned "application-agnostic" dependency statistics do not appropriately model.

While our visualization system is applied for single-cell RNA-seq analysis in this paper, it can be used more broadly for any applied data analysis of a high-dimensional dataset's dependency network G. We believe that for most applied data analysis, our visualization system can help scientists to gain a better understanding of the diverse landscape of bivariate relations within their dataset and to pick the most appropriate dependency statistic.

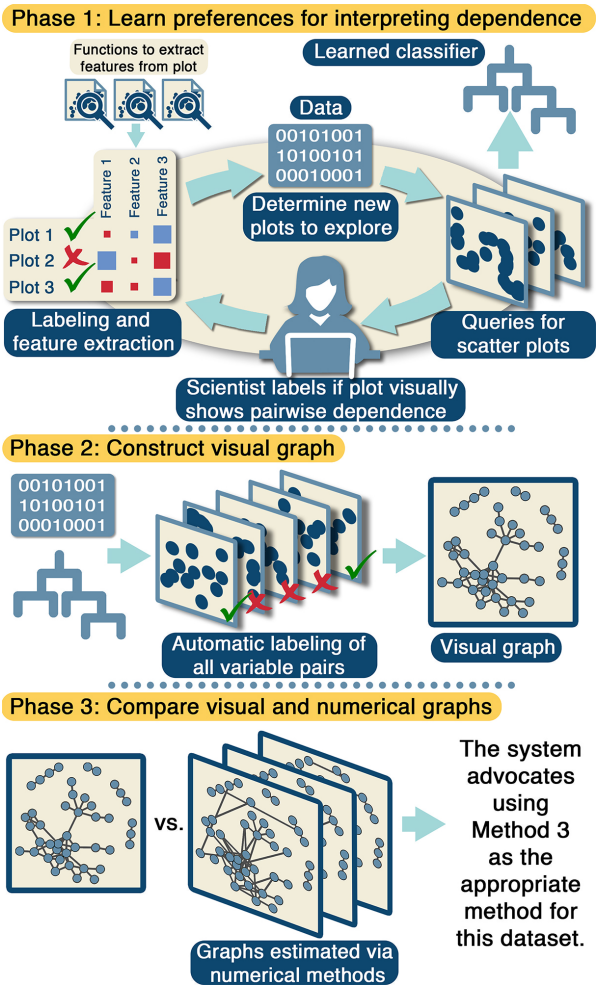

Figure 1: Visualization system to diagnose which method is most appropriate for the dataset. Phase 1 is described in Sections 3 and 4. Phase 2 and 3 are described in Section 5.

2 System overview

In this paper, we build a visualization system in `R` that revolves around scatter plots. Let $\mathbb{X} \in \mathbb{R}^{n \times d}$ denote the dataset in question, containing n samples and d variables. Suppose a scientist has K different methods that aim to estimate the dependency between two random variables (i.e., Pearson, distance correlation, etc.), where $G^{(K)}$ denotes the kth (of K) estimated dependency graph. Notably, while $G^{(1)}, \ldots, G^{(K)}$ are all estimated from the same dataset $\mathbb{X}$, they might vary numerically. These differences in graphs could potentially result in dramatically conclusions in a subsequent downstream analysis.

As mentioned in Wickham et al. (2010), to pick which graph is the most appropriate one, a scientist can visually interpret whether or not two variables are dependent from the scatter plot, and she can compare this against the numerical results for each of the k methods. However, in practice, when d becomes large, it is unrealistic to check all $\binom{d}{2}$ pairwise scatter plots against each pairwise dependency graph, $G^{(1)}, \ldots, G^{(K)}$. To resolve this, our visualization system learns how the scientist visually interprets if two variables are dependent based on her labelling a small subset of scatter plots chosen in an active learning manner (Phase 1). This allows the system to build a pairwise dependency graph based on what it thinks the scientist would perceive from unseen scatter plots (Phase 2), and finds the closest numerically estimated graph among $G^{(1)}, \ldots, G^{(K)}$ (Phase 3). In short, this system selects the numerical method that matches the most with the scientist's preferences and visual perception (Figure 1).

We design this system to be as interpretable as possible in each phase so that our system can provide supporting evidence on why one of the K methods was selected over another. The details of our system are described in the Appendix. While we focus on *model selection*, a specific task within model diagnostic, we note that the first two phases of our system can help with exploratory data analysis as well.

3 Feature extraction

As shown in Figure 1, our system revolves around learning how the scientist interprets scatter plots showing different variable pairs. The main engineering challenge for this task is extracting interpretable, numerical features from samples between X_i and X_j so the system can meaningfully quantify the differences among scatter plots. Of course, the dependency measures themselves (i.e., Pearson correlation, distance correlation, etc.) are possible functions to consider, but there are many other functions that could quantify how two variables are related. Figure 2 provides broad categories on what these functions could quantify. Here, we had denoised the single-cell RNA-seq data using SAVER (Huang et al., 2018) prior to visualization. After an appropriate variable screening, we are left with 324 genes. We aim to estimate the dependency network among these genes. See the Appendix for more details on our preprocessing of the data.

- **Clustering**: Some functions can quantify the number of modes in the two-dimensional density, or count the number of outliers in the sample.
- **Regression**: Some functions can quantify if there's a distinct linear or nonparametric relationship between the two variables.
- **Shape**: Some functions can quantify if the joint distribution is symmetric around some mean function, or if the distribution has long tails.
- **Variability**: Some functions can quantify whether most of the samples are spread out or highly concentrated within a small area, or how well the samples can be approximated by a bivariate Gaussian distribution.

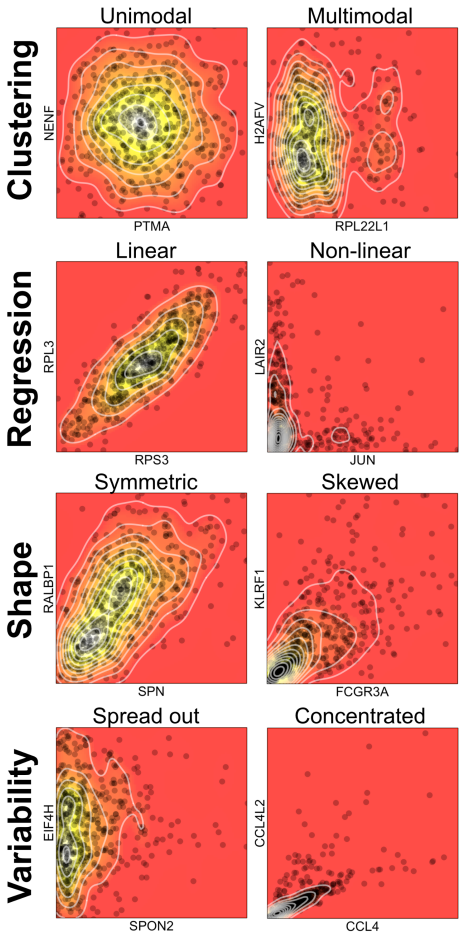

Figure 2: Example scatter plots shown by the visualizer system for the scientist to label. We plot the scatter plots against heatmaps visualizing the two-dimensional kernel density estimate to aid our interpretation (yellow denoting high density, red denoting low density), along with white contour lines. Each row denotes a different category of features, and the two columns denote examples of different extremes in that category.

As mentioned in Figure 1, the system learns how the scientist interprets the scatter plots in the first phase. Specifically, the system shows the scientist a scatter plot between a different variable pair, X_i and X_j for $i, j \in \{1, \ldots, d\}$ and $i \neq j$ in each iteration, and asks her to visually assess whether or not the two variables are dependent. This will become the training labels for our classifier, described in the next section. The system collects the training features by applying the functions described above on the samples between these two

variables, $\{\mathbb{X}_{1,i}, \mathbb{X}_{1,j}), \ldots, (\mathbb{X}_{n,i}, \mathbb{X}_{n,j})\}$. After labelling a small number of scatter plots (each of a different pair of variables X_i and X_j), the scientist can visualize the training labels she generated and training features collected as a heatmap, as shown in Figure 3. Based on this heatmap, the scientist can determine if she wants to change some of her previous labels or if she needs to label more scatter plots before proceeding to the next phase. Ideally, the scientist should be able to discern that some features can jointly accurately classify her labels, and that similar plots she came across were similarly labeled. In our genomic application, we labeled 30 scatter plots, but only 10 are shown in Figure 3.

4 Dependency discovery

After numerous scatter plots have been labeled and quantified, the system learns a random forest classifier that mimics the preferences of the scientist at the end of the first phase. One reason we choose to use a random forest classifier is to incorporate active learning in our system. By using an active learning scheme to select which variable pairs the scientist should label next, our system can show the scientist a diverse range of plots to accurately learn her preferences with as few scatter plots as possible. Specifically, after labelling a scatter plot, our system uses a batched uncertainty sampling scheme to determine the next scatter plot to show the scientist. The most basic version is outlined in the following pseudo-code, where we initialize $\mathcal{L} = \emptyset$ and $\mathcal{U} = \{(i,j) \in \{1,\ldots,d\}^2 : i \neq j\}$. Typically, the tuning parameter m is a small positive integer (for example, $m = 15$ by default).

Data: Labeled variable pairs $\mathcal{L}$, Unlabeled variable pairs $\mathcal{U}$, Tuning parameter m

Uniformly select m variable pairs $\mathcal{S} = \{(i_1, j_1), \ldots, (i_m, j_m)\} \in \mathcal{U}$;

Train a random forest classifier C on $\mathcal{L}$ (using the extracted numerical features and the user-provided labels) if $\mathcal{L} \neq \emptyset$;

for *each variable pair* $(i,j) \in \mathcal{S}$ **do**

  Extract the vector of numerical features $\boldsymbol{z}$ (described in Section 3) from $\{(\mathbb{X}_{i,1}, \mathbb{X}_{j,1}), \ldots, (\mathbb{X}_{i,n}, \mathbb{X}_{j,n})\}$;

  Compute the entropy among the predictions for $\boldsymbol{z}$ based on all the individual decision trees in C;

end

Have the user label the scatter plot corresponding with the variable pair $(i,j) \in \mathcal{S}$ with the largest entropy as "Dependent" or "Not dependent";

Add (i,j) to $\mathcal{L}$, and record the user-provided label and extracted numerical features. Also, remove (i,j) from $\mathcal{U}$;

We repeat the above procedure to actively learn which pair of variables should be labeled until the scientist is satisfied with the performance of the classifier C. Two additional reasons we choose to use a random forest classifier stem from recent visualization tools to interpret the classifier. These tools allow the scientist to gain a better understanding of the fitted random forest classifier's behavior and determine if the classifier captures meaningful signal.

- **Variable importance**: Roughly speaking, a variable is important if it partitions the samples by their classification labels within each decision tree of the random forest. This is often quantified by the mean decrease in the Gini index.
- **Tree approximation**: While it is hard to visualize a random forest due to its numerous decision trees, we can visualize the decision tree that best approximates the predictions of the random forest (Zhou and Hooker, 2016).

Lastly, random forest classifiers can output the predicted probability a particular variable pair is dependent. Hence, the system can show the user the scatter plots of the variable pairs that are most likely to not be dependent, most uncertain, or least likely to be dependent. After manually inspecting a few of the scatter plots (such as the scatter plots at each of the extremes), the scientist can proceed to the next phase. In Figure 4, we display these diagnostics after we manually labeled 30 scatter plots discovered using our visualization system. Here, as an illustration, we focused on a particular set of $d = 20$ genes among the CD8$^+$ effector T-cells.

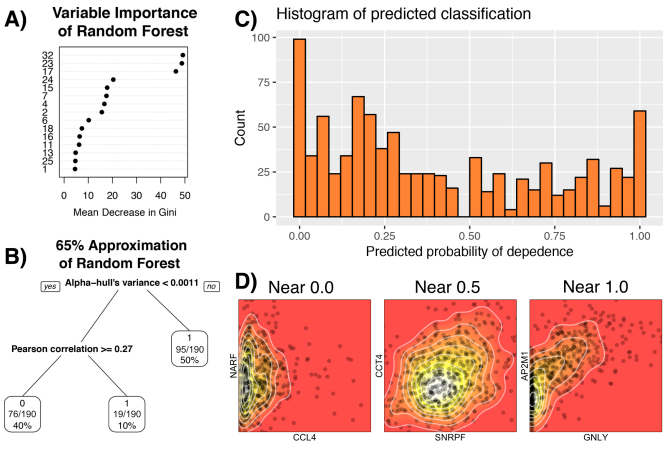

Figure 4: **A** and **B**) The variable importance and best decision tree of the random forest. Notice that variable 32, "Mutual's change for core" as seen in Figure 3 is the most important variable, but is not used when constructing the best approximate decision tree, which matches the predictions of the fitted random forest classifier on 65% of $\binom{20}{2}$ variable pairs. **C**) Histogram of predicted probability of dependence among all variable pairs. **D**) Example scatter plots associated with variable pairs that are strongly predicted to be not dependent (i.e., predicted probability near 0), ambiguous (near 0.5), and dependent (near 1).

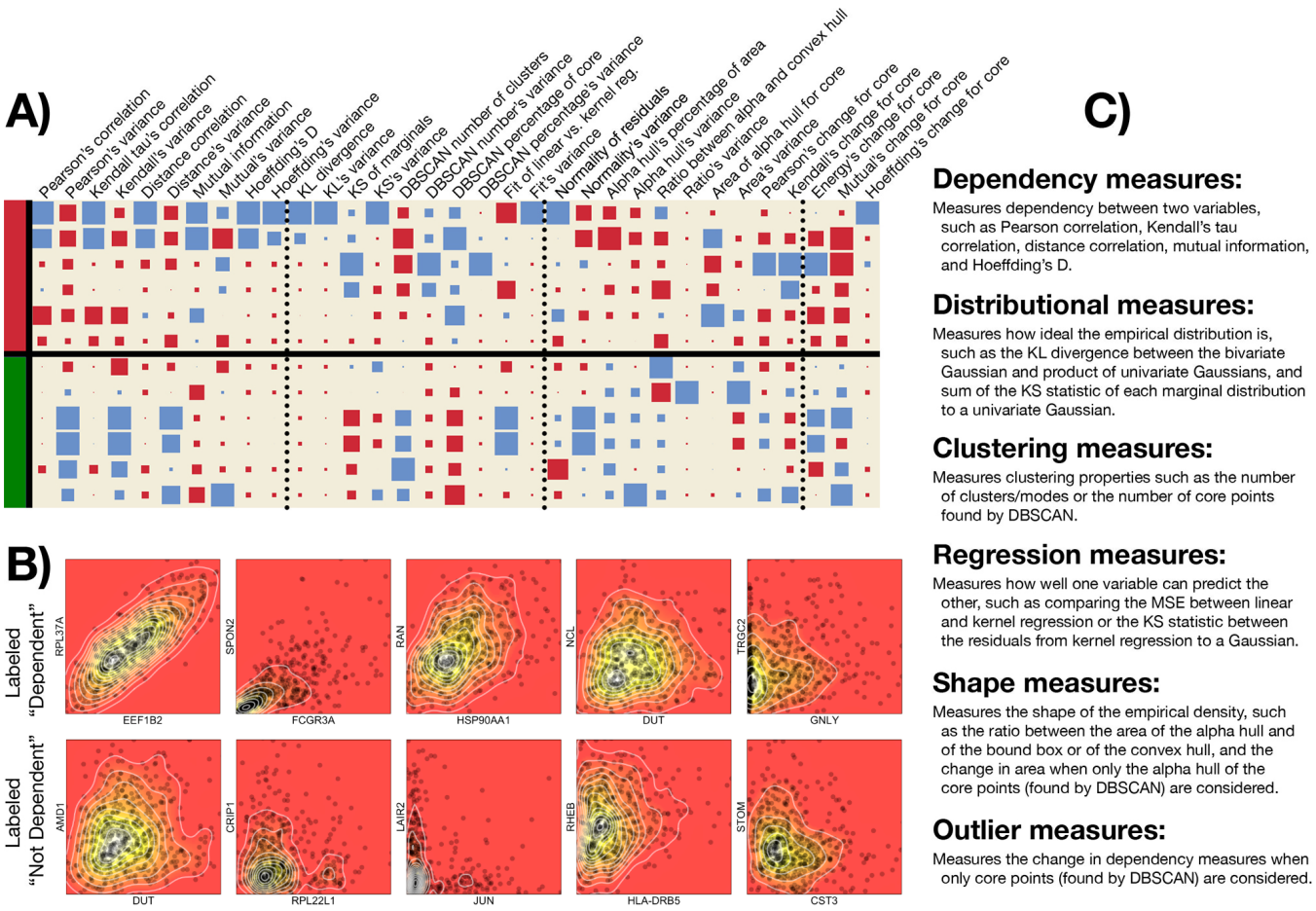

Figure 3: **A**) Heatmap showing the features of the labeled scatter plots. We adopt the plotting scheme developed in Buja et al. (2016) to allow an easier comparison of the features. Only 12 of the 30 labeled scatter plots, one per row, and 33 of the 50 features are shown here, one per column. In left-most column, the 6 red rows denotes scatter plots labeled as "Dependent," while the 6 green rows denotes those labeled as "Not Dependent." In all the remaining 33 columns, blue squares denote values above the mean value within that feature, and red squares denote values below the mean. The size of the square denotes the magnitude of the deviation from the mean value. The vertical dotted lines denote blocks of 10 features. Among the first 28 features, every even feature measures the variability of the odd feature in front of it over subsamples drawn uniformly. **B**) An example of 10 scatter plots used during the training process. We choose to show these specific 10 scatter plots among the 30 labeled scatter plots to demonstrate how the pairwise scatter plots in the dataset can display varied geometric properties, which necessitates an interactive visualization system such as ours to complement numerical analyses. **C**) A short verbal description of the 33 features, grouped by categories. The descriptions are in order of the named columns in the heatmap. Under the category of "Clustering measures," we use DBSCAN, a density-based clustering method, to cluster the bivariate data or to find the core points.

5 Graph comparison

With a classifier at hand that captures the preferences of the scientist, our system can automatically label all $\binom{d}{2}$ variable pairs to construct the *visual graph*, which we denote as $G^{(\text{vis})}$ (i.e., Phase 3 as depicted in Figure 1). This pairwise dependency graph approximates the scientist's preferences on which variable pairs are dependent based on scatter plots. This contrasts with the *numerical graphs*, which are pairwise dependency graphs numerically estimated using the statistical methods mentioned in Section 1. Among the K ways to estimate the dependency network G, we select the method most appropriate for the dataset by determining which numerical graph (among $G^{(1)}, \ldots, G^{(K)}$) is closest to $G^{(\text{vis})}$. While there are many ways to quantify the similarity between graphs, we use the the *shared information distance* between the node partitions of each numerical graph and that of the visual graph. The graph that achieves the smallest distance is the graph that our system advocates the scientist to use. This metric is most suitable for our genomic application, as described in the next section, but other metrics could be used as well.

The shared information distance is defined for two partitions of d elements. Let $\mathcal{P} = \{P_1, \ldots, P_k\}$ be a non-overlapping partition with k subsets, where $P_i \in \{1, \ldots, d\}$ denotes all the elements in the ith subset. We use spectral clustering (Lei and Rinaldo, 2015) to obtain this partitioning. Similarly, $\mathcal{Q} = \{Q_1, \ldots, Q_\ell\}$ is another partition with ℓ subsets. We define $p_i = |P_i|/d$, the fraction of elements in the ith partition of $\mathcal{P}$, and similarly for q_i. Lastly, let $r_{ij} = |P_i \cap Q_j|/d$, the fraction of elements in the ith partition of $\mathcal{P}$ and the jth partition of $\mathcal{Q}$. The shared information distance is then

$$d(\mathcal{P}, \mathcal{Q}) = -\sum_{i,j} r_{ij} \Big[\log(r_{ij}/p_i) + \log(r_{ij}/q_j) \Big].$$

Our graph comparison algorithm is as follows.

Data: Visual graph $G^{(\text{vis})}$, Numerical graphs $G^{(1)}, \ldots, G^{(K)}$

Estimate the partition $\mathcal{P}$ using community detection algorithm A on $G^{(\text{vis})}$;

for *each numerical graph* $G^{(k)}$ **do**

  Estimate the partition $\mathcal{Q}^{(k)}$ using A on $G^{(k)}$;

  Compute the shared information distance $d(\mathcal{P}, \mathcal{Q}^{(k)})$;

end

Select the graph $G^{(k)}$ with the smallest shared information distance;

For illustration, we consider the graphs formed by Pearson correlation, distance correlation and mutual information among $d = 20$ specific genes. We choose these particular genes as to emphasize the potential differences among the numerical graphs of different methods, and discuss the analysis among the full set of 324 genes in the Appendix. We tune the random forest classifier so the visual graph has 85% sparsity (meaning there are roughly $0.15\binom{d}{2}$ edges), and threshold each numerical graph appropriately to have the same number of edges. We then use spectral clustering to cluster the nodes in each of the four graphs into 5 different clusters. The resulting graphs are shown in Figure 5. Here, each graph is plotted to show all $d = 20$ genes (i.e., nodes) in the same layout. Using the shared information distance, among these three numerical graphs, our system states that the Kendall's tau graph is closest to our visual graph. Of course, for a more detailed single-cell RNA-seq analysis, some of the K methods could include domain-specific methods, such as the aforementioned methods in Dai et al. (2019) and Wang et al. (2021). However, to demonstrate the broad applicability of our method to a wide audience, we illustrate our visualization system using more domain-agnostic methods here.

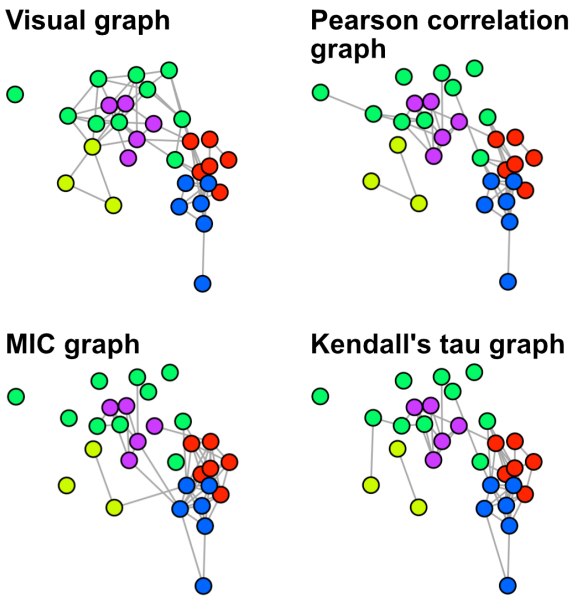

Figure 5: Visual graph and three numerical graphs. Each graph displays the same $d = 20$ genes in the same layout, and an edge denotes visually-perceived or statistically-estimated dependence between two genes. Here, the color arrangements among the $d = 20$ genes is shared among all four graphs, and is only to enable easier comparison between graphs.

Note in general, we caution scientists from using visual graphs directly for downstream analyses. This is because the visual graph $G^{(\text{vis})}$ is inherently not reproducible since it was constructed via the scientist's subjective preferences on what qualifies as statistical dependence. Instead, we believe $G^{(\text{vis})}$ should be mainly used as a model diagnostic to select among the K different statistical methods to assess statistical dependence.

Note that from Figure 5, we see certain pairs of genes that were estimated to be statistically dependent via

Pearson correlation, MIC, and Kendall's τ, but was not perceived to be statistically dependent by the learned preferences of the scientist. Similarly, we see certain pairs of genes that were perceived to be dependent by the scientist, but were not estimated to be statistically dependent via Pearson correlation, MIC, and Kendall's τ. We highlight such an example in Figure 6. In Figure 6A, based on the scientists' learned preferences, the shown scatter plot between the genes EIF5A and RAD23A is not predicted to demonstrate statistical dependence. However, among the the 11 dependency statistics, the shown relationship ranks above the 96% quantile among all $\binom{d}{2}$ bivariate relations. Perhaps the heavy-tailed nature of this bivariate distribution causes the discrepancy between the scientist's perceived lack of dependence and the numerically-estimated dependence. On the other hand, in Figure 6B, the shown scatter plot between the genes EIF5A and RAD23A is predicted to demonstrate statistical dependence. However, only HSIC and Normalized Mutual Information ranks the shown relationship above 85% among all $\binom{d}{2}$ bivariate relations. These examples further solidify our central theme – a carefully-designed visualization system can help scientists decide which statistical method to use when analyzing a particular dataset because no statistical method is expected to be the most appropriate for every dataset.

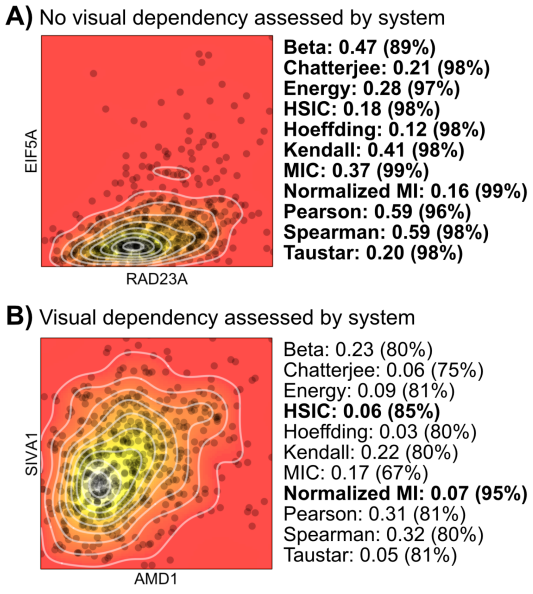

Figure 6: **A**) A scatter plot of two genes that was not predicted to be dependent, based on the learned preferences from the scientist's labelings. **B**) A scatter plot of two genes that was not predicted to be dependent, based on the learned preferences from the scientist's labelings. In both plots, for each of the $K = 11$ methods (i.e., Blomqvist's beta, Chatterjee's ξ, etc.), we display the estimated value of the statistic as well as its quantile when compared to the respective statistics among the remaining $\binom{d}{2}$ bivariate relations. The bolded methods denote the ones where the quantile is above 85% (i.e., our construction of $G^{(k)}$ would estimate dependency between the two genes).

We note that while our results here were based on a particular construction of numerical graphs $G^{(1)}, \ldots, G^{(K)}$ (i.e., a particular set of K dependency statistics and using the 85% quantile threshold to denote the presence or absence of dependency), there will always exist many discrepancies among numerical graphs regardless of what the specific construction is.

In the Appendix, we demonstrate the differences in the CD8$^+$ effector T-cells' gene co-expression networks among genes enriched for pathways relevant to T-cells for the visual graph (each generated by a different scientist) and other numerical methods, and also contrast the networks among different T-cell subtypes.

6 Conclusion

While numerical methods to estimate statistical dependency such as Pearson correlation, distance correlation, mutual information, and many others are precise and well understood mathematically, they do not always perform desirably under any conceivable dataset. Hence, it is important to use visualizations as model diagnostics so the user can make informed decisions on which method is most appropriate for her specific dataset. This paper describes an interactive visualization system tailored for pairwise dependency graphs, where we designed the system so the results of each phase are as interpretable as possible. By having the scientist provide subjective labelings for a carefully-crafted set of scatter plots learned via active learning, our system can deploy the learned preferences at scale in order to assess which statistical method best reflects the dataset's dependency structure.

7 Acknowledgements

We acknowledge Han Liu, Kathryn Roeder, Amy Tian, and Tianqi Zhao for useful suggestions for the methodology and data analysis throughout this paper.

References

Bergsma, W. and Dassios, A. (2014). A consistent test of independence based on a sign covariance related to kendall's tau. *Bernoulli*, 20(2):1006–1028.

Buja, A., Cook, D., Hofmann, H., Lawrence, M., Lee, E.-K., Swayne, D. F., and Wickham, H. (2009). Statistical inference for exploratory data analysis and model diagnostics. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 367(1906):4361–4383.

Buja, A., Krieger, A. M., and George, E. I. (2016). A visualization tool for mining large correlation tables: The association navigator. In *Handbook of Big Data*.

Bukovšek, D. K., Košir, T., Mojškerc, B., and Omladič, M. (2019). Relation between Blomqvist's beta and other measures of concordance of copulas. *arXiv preprint arXiv:1911.03467*.

Cao, S. and Bickel, P. J. (2020). Correlations with tailored extremal properties. *arXiv preprint arXiv:2008.10177*.

Chatterjee, S. (2021). A new coefficient of correlation. *Journal of the American Statistical Association*, 116(536):2009–2022.

Dai, H., Li, L., Zeng, T., and Chen, L. (2019). Cell-specific network constructed by single-cell RNA sequencing data. *Nucleic acids research*, 47(11):e62–e62.

de Siqueira Santos, S., Takahashi, D. Y., Nakata, A., and Fujita, A. (2014). A comparative study of statistical methods used to identify dependencies between gene expression signals. *Briefings in bioinformatics*, 15(6):906–918.

Deb, N., Ghosal, P., and Sen, B. (2020). Measuring association on topological spaces using kernels and geometric graphs. *arXiv preprint arXiv:2010.01768*.

Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. (2005). Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer.

Hoeffding, W. (1994). A non-parametric test of independence. In *The Collected Works of Wassily Hoeffding*, pages 214–226. Springer.

Huang, M., Wang, J., Torre, E., Dueck, H., Shaffer, S., Bonasio, R., Murray, J. I., Raj, A., Li, M., and Zhang, N. R. (2018). SAVER: Gene expression recovery for single-cell RNA sequencing. *Nature Methods*, 15(7):539–542.

Lei, J. and Rinaldo, A. (2015). Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1):215–237.

Madigan, D., Stang, P. E., Berlin, J. A., Schuemie, M., Overhage, J. M., Suchard, M. A., Dumouchel, B., Hartzema, A. G., and Ryan, P. B. (2014). A systematic statistical approach to evaluating evidence from observational studies. *Annual Review of Statistics and Its Application*, 1:11–39.

Martin, M. D. and Badovinac, V. P. (2018). Defining memory CD8 T cell. *Frontiers in immunology*, 9:2692.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck III, W. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, 177(7):1888–1902.

Székely, G. J., Rizzo, M. L., and Bakirov, N. K. (2007). Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794.

Wang, X., Choi, D., and Roeder, K. (2021). Constructing local cell-specific networks from single-cell data. *Proceedings of the National Academy of Sciences*, 118(51):e2113178118.

Wickham, H., Cook, D., Hofmann, H., and Buja, A. (2010). Graphical inference for infovis. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):973–979.

Zhou, Y. and Hooker, G. (2016). Interpreting models via single tree approximation. *arXiv preprint arXiv:1610.09036*.