**HNet: Graph learning using the Hypergeometric distribution.**

Erdogan Taskesen1\*

1Datalab, Ministry of Infrastructure and Water Management, Delft, 2622 HA, the Netherlands

ET: erdogan.taskesen@rws.nl

\* Corresponding author

# **Abstract**

**Background:** Real-world applications often require data sets that contain both measurements of continues and categorical values for the same sample. These so-called unstructured datasets have intensive pre-processing steps before addressing any hypothesis driven analysis or performing a data driven approach. Although such analysis is nowadays fastened with the availability of many libraries, determining the effect or causality of one variable on another, without any assumptions on the model form, remains a challenge because the search space of directed acyclic graphs (DAGs) patterns is super-exponential in the number of variables.

**Result:** We propose HNet, a method where relationships across variables in an unstructured data sets can be tested for significance. The result is a network-graph that consists of (partially) directed or undirected edges between the nodes (i.e., variables). We evaluated the accuracy of HNet using well known graphs with varying number of nodes and edges (i.e., sprinkler, lung cancer), for which the data is generated using Bayesian sampling. The performance is compared to Bayesian structure learning, and association rules for which we demonstrate that large scaling in nodes can still be computed in limited time.

**Conclusions:** We demonstrate that HNet allows detailed examination of the relationships across variables within a data set. HNet easily scales up to model thousands of variables and thereby overcomes some of the limitations of existing resampling-based methods.

**Key words:** hypergeometric test, structure learning, networks, unstructured data.

**Availability**: HNet is available at git XXX.

# **Introduction**

In recent years, there has been much effort in the progress of network-learning. The importance is stressed by many real-world applications that are complex interaction systems such as social networks, collaboration networks but also biological networks. By revealing the hidden patterns within the network, we can better understand the organizational and structural functions of network systems. In global, the field of network learning can be dissected into two major groups; *i)* The challenges where an existing network is provided (e.g., social network) with the goal to learn a network representation, and *ii)* the challenges where unstructured data is provided in a data matrix with the goal the learn the network structure or its associations (node links). In the first case, graph or knowledge embeddings can be used to transform node-links in a low-dimensional vector, which can then be used in applications with supervised or unsupervised models. Such an approach can capture the complex associations between node-links. Some popular methods are Splitter1, Deepwalk2, node2vec3 and LINE4. In the second group of network learning, unstructured data is an input into the model and the goal is to determine the representing network by means of strong relations between the variables. Questions can be addressed such as; does variable X (in)directly influence Y, or might they have a common cause? A representation can be learned using Bayesian Network structure learning5,6 which determines the directed acyclic graphs (DAG) given the data. Bayesian learning has been successfully applied in many fields such as insurance7, health8, and biological networks9. However, the search space of DAGs is super-exponential in the number of variables for which the typical scoring functions can result in a local suboptimum. This is especially the case for large data sets (e.g., with many node links to be determined) where an exhaustive search is intractable due to computational burden. As an example, an Bayesian approach has NP-complete11 complexity which requires a quantum solution12 for very large data sets. For small data sets, an exhaustive search for DAGs can be computed, whereas for medium data sets, the use of heuristics (e.g., hill-climbing10) in combination with Bayesian approaches can provide a good solutions. In addition to Bayesian learning, there are also rule-based machine learning techniques (association rules) to discover co-occurrence relationships between variables (item sets) in the large search space. The use of rule-based techniques, such as Apriori13, Eclat14 and FP-Growth, is successfully applied in many fields such as marketing (e.g., product placements, promotional pricing), retail (e.g., loyalty programs, sales promotions), security (e.g., intrusion detection15, malicious activities), web usage mining (e.g., advertisements). A drawback is however the risk of finding many spurious associations, and the limitation of only modelling categorical values (item lists).

With HNet, we aim to discover statistically significant associations across categorical as well as continues variables. We control for the risk of finding spurious associations by multiple test correction whereas the detected associations form the edges in a network. HNet does not force variables into static item sets but instead we created variable item sets that can be easily examined by our interactive network. To test HNet, we re-evaluated the detection of well-known networks, i.e., data set Alarm16, and Asia8 and demonstrate a good fit of our solutions. In addition we analysed a well-known unstructured data sets, the Titanic17. Here we demonstrate the goodness of fit, the ease of use and how to deeper examine the discovered associations.

# **Material and Methods**

**HNet.** Detection of significant nodes-links given a data set using HNet is a multi-step process (Figure 1, A-F). The first step (A) is feature typing, i.e., every feature is set either being categoric, numeric or is excluded. The typing can be user-defined or automatically determined on conditions. In the latter case, features are set to numerical if values are floating type or have more than a minimum number of unique elements (e.g., if the number of unique elements >20% of the total non-missing elements). Features are set to categoric if values are boolean, integer or string. The second step is encoding the categoric values into a one-hot dense array (Figure 1B). From the one-hot dense array we subsequently create combinatory features using *k* combinations over *n* features (without replacement, Figure 1C). By default k=1, meaning that the input matrix (*Xcategoric*) is the same as combinatory matrix (*Xcombination*). When k>1, *n* features are combined by multiplication for the *k* unique combinations. The new combinatoric feature (Xc) is then added to the dense array. To avoid high computational costs, mutual exclusive features are excluded, and features are excluded in case *Xi* contains less then ymin positive samples (default ymin=10).

The dense array (*Xcombination*) is subsequently used to assess significance with the categoric features (*Xcategoric*) (Figure 1D). Significance is tested by means of the hypergeometric distribution, where we test for over-representation of successes in sample *Xi*. The hypergeometric P-value between feature X*i* and feature Xc*j* (*Pc(i,j)*), is calculated as the probability of randomly drawing *x* or more successes from the population in *n* total draws with population size *N* containing *K* successes. For any *Xi* and *Xcj*, *Pc(i,j)* is computed as:

To assess significance across the numeric features (*Xnumeric*) in relation to the dense array (*Xcombination*) we utilized the Mann-Whitney-U test. Each numeric vector *Xni*, is split on categoric feature *Xci* versus ~*Xci*, and tested whether randomly selected value from *Xci* will be less than or greater than a randomly selected value from ~*Xci*.

All tested combinations, either categoric-categoric or categoric-numeric features, are stored in an adjacency matrix (*Padj*), and are corrected for multiple testing using False Discovery Rate (FDR)18 or Familywise error rate (FWER)19 (the default Multiple Test Method, MTM, is set to Holm20, Figure 1E).

The last step in HNet (Figure 1F) is declaring significance and the corresponding weights. An edge between two nodes is called significant when *P\*adj(i,j) < alpha*, where alpha is 0.05 (default). The edge-weight is set by:

The final output of HNet is an adjacency matrix containing the edge weights that indicate whether pairs of vertices are adjacent or not in the graph.

**d3graph** is a dynamic graph representation to deeper examine the detected associations. Just like static graphs, the dynamic graph consists out of nodes and edges with adjustable sizes and colours. However, with d3graph we create an interactive network that is stand-alone html file. The network includes collision and charge parameters to ensure that nodes do not overlap. We associate each node with a text-label, and nodes with its corresponding links can be highlighted when double clicked on it. We also allow to break links based on the edge weights using a slider. Each node involves a tooltip that can easily be adapted to display any of the underlying data. We developed d3graph as a python function which output custom java script file for the user defined parameters. The custom java script file on its turn uses functionalities from the d3 java script library (version 3). In its simplest form, the input for d3graph is an adjacency matrix for which the elements indicate pairs of vertices are adjacent or not in the graph.

**Bayesian structure learning.** We utilized the Bayesian structure learning algorithm to learn the optimal DAG using the data sets based on a score-based approach, and under the assumption that the data is complete (no missing values). The score-based model selection approach consists out of two main parts, the scoring function, and the search strategy. The scoring function maps models to a numerical score based on how well it fits to the given data set, whereas the search strategy enables selection of a model with optimal score across the search space of all possible models. We used Bayesian information criterion (BIC) as the scoring functions to measure the fit between model and data, and hill-climbing as search strategy. We ran Bayesian structure learning on data set containing the varying set of samples and selected the best scoring model.

# **Data**

**Data**. Directed Acyclic Graphs (DAG) of Sprinkler and Alarm16, and Asia8 are used to generate a data set by means Bayesian inference and forward sampling. Generated data sets vary in sample sizes; n=[100,1000,5000,10000]. The number of nodes for the Asia model contains 8 nodes, 8 arcs, and 18 parameters. The Alarm model contains 37 nodes, 46 arcs and 509 parameters. The titanic data set contains 891 samples with 12 feature columns.

# **Results**

**Titanic dataset.** Using HNet, we analysed the associations across the 12 input features for 891 samples. The first step is the typing of features using default settings. The resulting array contains in total 19 categoric features (i.e., Survived [1,0], Pclass [1,2,3], Sex [female,male], Sibsp [0,1,2,3,4], Parch [0,1,2], Cabin, Embarked [C,Q,S] ), and 2 numeric features (Fare and Age). The features Passenger-ID and Name are excluded from the model as the features did not fulfil the requirement of having at least 10 samples within a category or being of numeric type. HNet characterized in total 60 unique edges across 47 nodes under alpha=0.05 and multiple testing using Holm (Figure 2A). Note that the detected edges are not necessarily symmetric which can therefore be indicative for directionality, as an example no survival (survived=0) is significantly associated with males, but not the other way around. Although the ground truth of this data set is unknown, the strongest association is in line with intuitive expectations, i.e., first class passengers are significantly associated with High Fare (fare>60.3, P<2.87-79), whereas third class passengers are significantly associated with low fare (fare<8.1, P<4.99-73). The next best association is between passengers that are female and survived (P<4.79-57), followed by passengers that did not survived and are males (P=4.79-57). The graph is consistently formed across these two clusters. The male-no survival cluster is expanded with low fare, having no siblings, embarking position is S or Q whereas the females-survival cluster is expanded with high fare, having 1 sibling, and embarking position is C. Interestingly, directionality for passengers that did not survived is outwards whereas those that survived is mainly inwards. This may suggest that survived passengers are on coordinated actions, whereas it is not for passengers that did not survive.

**Alarm dataset.** Using HNet, we analysed the associations across the 12 input features for 891 samples. The

Based on our approach, the 25 unique tissues grouped into 22 clusters (Fig 2A). We characterized the clusters with a tissue label by means of the hypergeometric test with *P*<0.0001 (Fig 2A). Out of the 22 detected clusters, 17 clusters showed near one-to-one relationships with the tissue label, three clusters showed a collection of multiple different tissues, which is not unexpected as these tissues represent respectively hormone-sensitive female reproductive tissues in cluster 12 (Fallopian tube, Ovary and Uterus samples), tissues associated with the digestive system in cluster 13 (Colon, Kidney, Pancreas and Stomach samples), and mucous-membrane tissues in cluster 14 (Esophagus and Vagina samples). Note that, although these tissues are grouped together in the cluster analysis, a visual look in the t-SNE map (Fig 1) demonstrates separation of the tissues. Furthermore, two clusters contained major brain sub-regions; one cluster contained samples from the cerebellum, and the other cluster contained 11 other brain sub-regions (Fig 1B). Besides the separation of tissues, the t-SNE-map also revealed substructures in the brain, blood and skeleton muscle tissues. For the 313 samples in the brain regions we did re-cluster using the initial t-SNE coordinates, and could demonstrate clear grouping of Cerebellar/Cerebellum regions, Basal Ganglia regions, Cortex regions, hypothalamus, and a mixture of Hippocampus, Amygdala, Substantia Nigra, and Spinal Cord regions were seen (Fig 1B). In whole blood we detected separation of pre- and post-mortem samples, and novel substructures were seen in skeleton muscle (Fig 1A) for which no further tissue annotation is available. Note that minimal differences in clustering results are observed if different gene expression level filtering cut-off values [0.5,..,10] are used (S1 and S2 Fig).

Twenty-one clusters overlapped significantly between the t-SNE-map and HC approach on the high-dimensional data (*P*<0.0001) (Fig 2B). However, the HC approach grouped samples in larger clusters, e.g., all brain samples (HC cluster 2) or the ‘skeleton muscle’ and heart tissue samples (HC cluster 5). Contrarily, the t-SNE-map clearly separates tissue types into different clusters (see also dendrogram differences in Fig 1C). A comparison between the HC in the original data space versus HC in low data space yielded in a cophenetic correlation of 0.68, indicative of overlapping clusters.

To investigate the value of the t-SNE mapping further, we evaluated the results of different clustering algorithms (DBSCAN, HC, k-means, and Mixture of Gaussians) between the brain samples in the low dimensional t-SNE-map and in their original high dimensional representation. For all four clustering methods, the t-SNE-map resulted in higher significant cluster enrichment of the different brain regions (S1 Table), and lower Davies-Bouldin scores (S3 Fig). This demonstrates that a reduction of data complexity, by a transformation step of samples into a low dimensional space, is beneficial for follow-up analysis. As an example, with the use of HC we detected eight clusters among the brain samples in both the original and low dimensional space, but in the low dimensional map we detected clusters that are more representative for the different brain regions (Fig 1C).

Interestingly we also detected that the gene expression profiles of 31 samples do not map to the cluster with a matching tissue label. These samples are either outlier tissues that are more heterogeneous at the cellular level (Fig 1A and S2 Table), or may have been mislabelled. This discrepancy was left unnoticed in the RNA-seq analysis, but the t-SNE-map clearly addresses the issue. Twelve out of 31 outliers lie in the adipose cluster 9 which is known to be a strong contaminant of other tissues samples as described in the pathology notes. Out of the 12 samples, 8 samples originate from the neighbouring breast tissue cluster.

# **Discussion**

Taken together, we demonstrate that a 2D representation based on the t-SNE mapping of tissue-samples closely resembles the HC results and confirms the findings of the published GTEx studies. Our presented results are based on data that is derived from only the GTEx consortium. If multiple data sets from different consortiums are combined, it may require additional normalization steps or may even require batch-correction methods before making a joint analysis using t-SNE.

It should be noted that, besides the similarly clustered samples, differences between HC clustering in the original space versus the low dimensional space are also seen (Fig 1C). As an example, samples of the spinal cord are clustered in the original HC space (15 out of 16) whereas in the low dimensional space, these samples are divided into two clusters (Fig 1C). Interestingly, all samples (except for one) are in relative close proximity to each other in the 2D space (Fig 1B). Thus in terms of biological interpretation, there is one sample of the spinal cord which may be an outlier (clustered with Hypothalamus), and both approaches agree on that. In general, (small) differences between results are insurmountable when distinct approaches are used. A final biological interpretation remains an experts task for which our provided t-SNE-map allows a more detailed examination of local substructures across and within the human tissues.

# **Author Contributions**

ET designed the study, analysed the data, and drafted the manuscript.

# **Competing interests**

The author has declared that no competing interests exist.

# **References**

1. Epasto, A. & Perozzi, B. Is a single embedding enough? Learning node representations that capture multiple social contexts. in *The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019* (2019). doi:10.1145/3308558.3313660.

2. Qiu, J. *et al.* Network embedding as matrix factorization: Unifying DeepWalk, LINE, PTE, and node2vec. in *WSDM 2018 - Proceedings of the 11th ACM International Conference on Web Search and Data Mining* (2018). doi:10.1145/3159652.3159706.

3. Grover, A. & Leskovec, J. Node2vec: Scalable feature learning for networks. in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016). doi:10.1145/2939672.2939754.

4. Tang, J. *et al.* LINE: Large-scale information network embedding. in *WWW 2015 - Proceedings of the 24th International Conference on World Wide Web* (2015). doi:10.1145/2736277.2741093.

5. Friedman, N. & Koller, D. Being Bayesian about network structure. A Bayesian approach to structure discovery in Bayesian networks. *Mach. Learn.* (2003) doi:10.1023/A:1020249912095.

6. Braun, D. A., Mehring, C. & Wolpert, D. M. Structure learning in action. *Behavioural Brain Research* (2010) doi:10.1016/j.bbr.2009.08.031.

7. Binder, J., Koller, D., Russell, S. & Kanazawa, K. Adaptive Probabilistic Networks with Hidden Variables. *Mach. Learn.* (1997) doi:10.1023/A:1007421730016.

8. Lauritzen, S. L. & Spiegelhalter, D. J. Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems. *J. R. Stat. Soc. Ser. B* (1988) doi:10.1111/j.2517-6161.1988.tb01721.x.

9. Sachs, K., Perez, O., Pe’er, D., Lauffenburger, D. A. & Nolan, G. P. Causal protein-signaling networks derived from multiparameter single-cell data. *Science (80-. ).* (2005) doi:10.1126/science.1105809.

10. Tsamardinos, I., Brown, L. E. & Aliferis, C. F. The max-min hill-climbing Bayesian network structure learning algorithm. *Mach. Learn.* (2006) doi:10.1007/s10994-006-6889-7.

11. Chickering, D. M. Learning Bayesian Networks is NP-Complete. in (1996). doi:10.1007/978-1-4612-2404-4\_12.

12. O’Gorman, B., Babbush, R., Perdomo-Ortiz, A., Aspuru-Guzik, A. & Smelyanskiy, V. Bayesian network structure learning using quantum annealing. *European Physical Journal: Special Topics* (2015) doi:10.1140/epjst/e2015-02349-9.

13. Agrawal, R. & Srikant, R. Fast Algorithms for Mining Association Rules in Large Databases. in *Proc. of the 20th International Conference on Very Large Data Bases (VLDB’94)* (1994).

14. Zaki, M. J. Scalable algorithms for association mining. *IEEE Trans. Knowl. Data Eng.* (2000) doi:10.1109/69.846291.

15. Martellini, M. & Malizia, A. *Cyber and Chemical, Biological, Radiological, Nuclear, Explosives Challenges: Threats and Counter Efforts*. (2017).

16. Beinlich, I. A., Suermondt, H. J., Chavez, R. M. & Cooper, G. F. The ALARM Monitoring System: A Case Study with two Probabilistic Inference Techniques for Belief Networks. in (1989). doi:10.1007/978-3-642-93437-7\_28.

17. Singh, A., Saraswat, S. & Faujdar, N. Analyzing Titanic disaster using machine learning algorithms. in *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017* (2017). doi:10.1109/CCAA.2017.8229835.

18. Benjamini, Y., Drai, D., Elmer, G., Kafkafi, N. & Golani, I. Controlling the false discovery rate in behavior genetics research. in *Behavioural Brain Research* (2001). doi:10.1016/S0166-4328(01)00297-2.

19. Hochberg, Y. A sharper bonferroni procedure for multiple tests of significance. *Biometrika* (1988) doi:10.1093/biomet/75.4.800.

20. Holm, S. A simple sequentially rejective multiple test procedure. *Scand. J. Stat.* (1979).

# **Figures and Tables**

**Fig 1. The tissue landscape of the gene expression profiles illustrated in a two-dimensional map.**

(A) Projection of 1641 GTEx samples in a 2D-map: Each point represents a sample which is coloured according to the (sub)tissue label (45 in total). Samples that cluster outside the matching tissue subtype-cluster are circled in red. Sample clusters are illustrated by the 22 differentially coloured density maps. (B) the projection of 313 brain samples and their associated brain-tissue. (C) Comparison of brain samples clustered by the HC approach versus the t-SNE approach resulted in cophenetic correlation of 0.68. An edge links the sample ID positions as clustered by the HC and t-SNE approaches. Edge colours are based on the brain tissue regions. Clusters are labelled if a particular brain region was significantly overrepresented in the cluster (hypergeometric test with *P*<0.001).

**Fig 2. Cluster associations and comparison with the hierarchical clustering approach**. (A) Association of the t-SNE-map based clusters with tissue types. A star indicates significant overrepresentation (*P*<0.0001) of samples in a cluster with the respective tissue label. Red-coloured numbers depict samples outside the cluster that matches their tissue label. Coloured squares depict the percentage of samples that overlap. (B) Comparison of the t-SNE-map based clusters versus the HC approach based clusters. A star indicates significant over-representation (*P*<0.0001) of samples in a cluster of the t-SNE-map with samples in a cluster of the HC approach. Coloured squares depict the percentage of samples that overlap.

# **Supporting Information**

**S1 Fig. Cluster associations for different pre-processing steps of the gene expressions**. Cluster associations with the tissue labels are determined for 20 different gene filtering cut-offs, i.e., [0.5,..,10]. Each set of genes is reduced to 2 dimensions (by means of t-SNE) followed by DBSCAN cluster analysis to determine the optimal number of clusters. Each square depicts the -log10(P) enrichment of the cluster label with the tissue label, and shown when *P*<0.001. Results are ordered on tissue label.

**S1 Table: Comparison of four cluster algorithms in original data space and low dimensional space.** Four cluster algorithms are compared in the original data space and in the low dimensional space. Enrichment is computed for the detected cluster labels and the brain regions. Original space is by using all available RNA seq features. Low dimensional space is by first employing a t-SNE to two dimensions.