# Measures for the Evaluation and Comparison of Graphical Model Structures

Gabriel Kronberger[1] (0000-0002-3012-3189), Bogdan Burlacu[1,2], Michael Kommenda[1,2], Stephan Winkler[1], and Michael Affenzeller[1,2]

[1] Heuristic and Evolutionary Algorithms Laboratory
University of Applied Sciences Upper Austria, Softwarepark 11, 4232 Hagenberg
gabriel.kronberger@fh-hagenberg.at

[2] Institute for Formal Models and Verification
Johannes Kepler University, Altenbergerstr. 69, 4040 Linz

**Abstract.** Structure learning is the identification of the structure of graphical models based solely on observational data and is NP-hard. An important component of many structure learning algorithms are heuristics or bounds to reduce the size of the search space. We argue that variable relevance rankings that can be easily calculated for many standard regression models can be used to improve the efficiency of structure learning algorithms. In this contribution, we describe measures that can be used to evaluate the quality of variable relevance rankings, especially the well-known normalized discounted cumulative gain (NDCG). We evaluate and compare different regression methods using the proposed measures and a set of linear and non-linear benchmark problems.

**Keywords:** Graphical Models, Structure Learning, Regression

## 1 Introduction

We aim to define an efficient algorithm for learning the structure of a graphical model solely from observational data. The algorithm should work for processes with continuous variables, non-linear dependencies, and noisy measurements.

Graphical models are not only useful for visualization purposes but can be facilitated for online control. Consider for example a system, which has multiple continuous inputs and dependent outputs as well as internal variables. In such applications, the system parameters are often interrelated. Therefore, a controller cannot set parameter values independently of each other. An explicit and complete model of all variable dependencies allows a controller to set valid values for all parameters and achieve a stable process.

This technical report represents material published in *Computer Aided Systems Theory – EUROCAST 2017* [14]. The original publication is available at https://link.springer.com/chapter/10.1007/978-3-319-74718-7_34. 

Many variants of structure learning algorithms have been proposed, including exact as well as approximate algorithms. However, most of the algorithms work only for discrete variables or impose constraints on the type of dependencies (e.g. only linear dependencies). Our intention is to use standard regression algorithms to learn models for each variable and calculate a variable relevance ranking for each model. The relevance ranking can later be used to guide a heuristic structure learning algorithm.

In this paper, we focus on the sub-problem of evaluating the quality of variable relevance rankings, i.e. we try to answer the research question: *How can we quantify the accuracy of variable relevance rankings produced by regression algorithms?*

## 2 Related Work

Structure learning for Bayesian networks is NP-hard [2]. In the past, a large number of algorithms for structure learning have been formulated including exact methods as well as approximate algorithms (cf. [11]). Exact methods such as the algorithms described in [9] and [17] use dynamic programming and work for problems up to 30 variables [1]. Approximate algorithms can handle much larger networks [1]. The PC algorithm [18] is a classical exact algorithm for structure learning which assumes an oracle for determining whether pairs are (conditionally) independent. An improved variant is the MMPC algorithm [19]. Many of the published algorithms work only for discrete variables. A simple approach for continuous variables and linear systems is to fit a lasso regression using each variable as the response and the others as predictors [15]. A more systematic approach is the graphical lasso [4] which optimizes a global penalized likelihood and can solve sparse problems with 1000 nodes in less than a minute [4]. The "ideal-parent" algorithm [3] can be used for structure learning for non-linear systems with continuous variables. However, the dependencies between variables are limited to generalized linear models with non-linear link functions. Artificial neural networks as well as Gaussian process networks have been used for structure learning for non-linear and continuous Bayesian Networks [5, 7].

Variable interaction networks can be used to visualize dependencies between variables of a system [12, 16]. Variable interaction networks are specific types of graphical models [10] in which nodes represent variables of the system and directed edges represent dependencies between variables. So far, variable interaction networks have been used primarily for visualization with the aim to gain a better understanding of complex systems (see e.g. [12, 13, 16, 20]). In these applications, empirical models (e.g. symbolic regression models) have been used to identify and describe statistical dependencies between continuous variables.

## 3 Methods

We use standard regression modeling methods to generate a ranking of input variables by relevance for each dependent variable e.g. using the explained variance measure.

Assuming multiple different methods are available for generating graphical models from observational data, we want to determine which of those produces the best structure. This can be determined only if the optimal system structure is known. Therefore, we use a set of synthetic problem instances where the data generating process is known.

### 3.1 Measures for the Quality of Variable Relevance Rankings

We propose to use one of three indicators to evaluate and compare variable rankings: (1) Gini coefficient [6], (2) Spearman's rank correlation, and (3) normalized discounted cumulative gain (NDCG) [8]. The Gini coefficient can be used to quantify how well a modeling method is able to discriminate between the actually necessary inputs and unnecessary inputs and is closely related to the AUC measure for classification problems [6]. The Gini coefficient can be calculated even if no information on the actual importance ranking of variables is available, as it only depends on how well the modeling method discriminates between necessary and irrelevant variables.

Spearman's rank correlation coefficient and NDCG allow comparison of the estimated ranks with an ideal ranking. The former weights all elements of the ranking equally and is therefore strongly influenced by the relative ranking of irrelevant variables if there are only a few actually relevant variables. The later uses an exponential weighting scheme to assign more weight on the most important variables. Therefore, NDCG is an ideal measure for the evaluation of variable relevance rankings for structure learning where it is most important that the top-most variables are correctly identified while the ordering of irrelevant variables should not have a strong impact.

### 3.2 Empirical Evaluation of Variable Relevance Measures

For the empirical evaluation we use four different regression algorithms: linear regression (LR), Gaussian process regression (GPR), random forest regression (RF), and symbolic regression (SR) with genetic programming. Each of those four methods is used to estimate regression models for all dependent variables and the relevance of all input variables is determined. The resulting rankings are compared to the actual variable relevance rankings and the quality of the ranking is calculated. Finally, the arithmetic mean of all ranking qualities is determined to produce an overall quality for each method and problem instance.

### 3.3 Problem Instances

We have generated data by sampling from random processes where all dependencies between variables are known. Linear as well as non-linear systems are

considered. For the linear instances all dependent variables are described by randomly generated linear models. For the non-linear systems all dependent variables are described by randomly sampled Gaussian processes. We have generated problem instances with different dimensionality $d \in \{10, 20, 50, 100\}$ and different noise levels $\in \{0\%, 1\%, 5\%, 20\%\}$. All instances represent graphical models in which the variables can be assigned to one of four levels. The relative number of variables in each level is fixed. The first and second level contain 33% of the variables, the third level contains 20% of the variables and the fourth level contains the remaining 14%. Variables in the first level are sampled independently from a zero-mean unit-variance Gaussian distribution. The variables in the other levels are sampled from generative models which use input variables randomly chosen from all lower levels.

The effective dimensionality $d_{\text{eff}}$ is the number of input variables which are actually used in each model and is sampled according to the following expressions:

$$u \sim \text{uniform}(0, 1) \tag{1}$$
$$r = -2\log(1 - u) \tag{2}$$
$$d_{\text{eff}} = \lfloor 1.5 + r \rfloor \tag{3}$$

Variable values are randomly sampled starting with the variables in the first level. After the values in the first level have been assigned, the sampling procedure continues with the variables in the next higher level. Considering only one dependent variable $y$, the values $y_i$ $(1 \leq i \leq N)$ are generated using the following expression, where $\boldsymbol{x}_i$ contains the values of the randomly selected input variables for this dependent variable.

$$y_i \overset{i.i.d}{\sim} N\left(\mu = f(\boldsymbol{x}_i), \sigma = \sqrt{\frac{\text{ratio}_{\text{noise}}}{1.0 - \text{ratio}_{\text{noise}}}}\right) \tag{4}$$

We used $N = 250$ samples for each problem instance. Correspondingly, the high-dimensional problems are more difficult as it is more likely to observe spurious strong correlations of irrelevant variables.

For the linear instances we sampled random models $f(\boldsymbol{x}_i)$ using the following:

$$f(\boldsymbol{x}_i) \;=\; g(\boldsymbol{x}_i, \boldsymbol{w}) = \boldsymbol{x}_i^T \boldsymbol{w} \tag{5}$$
$$\boldsymbol{w}_j \;=\; \frac{\lambda_j^2}{\text{Var}(x_j)},\ 1 < j \leq d_{\text{eff}} \tag{6}$$
$$\lambda_j \overset{i.i.d}{\sim} N(\mu = 0, \sigma = 1) \tag{7}$$

For the non-linear instances we sampled random models $f(\boldsymbol{x})$ from a zero-mean Gaussian process prior with squared exponential covariance function with randomly sampled length scales $\boldsymbol{\ell}$ for each dimension to determine the relevance

of each variable.

$$f(\boldsymbol{x}) \sim \mathcal{GP}(0, k(\boldsymbol{x}, \boldsymbol{x}')) \tag{8}$$

$$k(\boldsymbol{x}_p, \boldsymbol{x}_q) = \exp\left(-(\boldsymbol{x}_p - \boldsymbol{x}_q)^T(\boldsymbol{\ell}I)^{-1}(\boldsymbol{x}_p - \boldsymbol{x}_q)\right) \tag{9}$$

$$\boldsymbol{\ell}_j \overset{i.i.d}{\sim} \text{uniform}(0.5, 2.5) \tag{10}$$

## 4 Results

Table 1 shows a comparison of the average accuracies of the variable rankings calculated with the three proposed measures for all benchmark instances with $\text{ratio}_{\text{noise}} = 5\%$. As expected, linear regression works better for linear systems and is not able to identify relevant variables correctly for the non-linear systems. For the non-linear instances, RF produces the best variable rankings.

Interestingly, the ordering of the methods similar, regardless of the measure, which is used to compare the methods. For example, for the Gaussian process system with 20 variables the ordering of methods is the same for all three measures.

For a sensitivity analysis we generated problem instances with different noise levels. Table 2 shows the NDCG values for all problem instances and noise ratios. The linear instances without noise (0%) are ill-conditioned because indirect and direct dependencies cannot be distinguished by the standard regression algorithms. Therefore, all methods produce better variable rankings when at least a small amount of noise is present. RF produces the best NDCG values for the non-linear instances and is able to identify to top-ranked variables even for high dimensionality.

## 5 Discussion and Conclusions

Our overall aim is to use estimated variable relevance rankings for guiding structure learning algorithms for graphical models with continuous variables and non-linear dependencies. In previous work we have used a similar approach to generate so-called variable interaction networks [12]. However, variable interaction networks are useful only for visualization purposes. Instead, we would like to use graphical models also for example for online predictive control where multiple interrelated process variables must be controlled through multiple parameters which might also be interrelated.

We have described how the quality of variable relevance rankings can be measured relative to a gold standard and have used three different measures (i) Spearman's rank correlation coefficient, (ii) the Gini coefficient, and (iii) the normalized discounted cumulative gain (NDCG). We have demonstrated how the measures can be used to compare the quality of the variable relevance rankings for different regression algorithms. For an empirical evaluation, we have used linear as well as non-linear benchmark problems.

**Table 1.** Comparison of variable relevance rankings. Values are averages over the values for all dependent variables as well as the ranks of the values in each row. Noise level is 5% for all problem instances. All three measures often produce the same rankings of methods. The best NDCG values for each problem are highlighted using bold font. SR works best for the linear problem instances and RF for non-linear instances.

| | | | Average Value | | | | Rank | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Problem type | $d$ | Measure | LR | SR | RF | GPR | LR | SR | RF | GPR |
| Linear | 10 | Gini | 0.98 | 0.83 | 0.69 | 0.96 | 1 | 3 | 4 | 2 |
| | | NDCG | **0.99** | 0.93 | 0.85 | 0.97 | 1 | 3 | 4 | 2 |
| | | Spearman | 0.74 | 0.57 | 0.37 | 0.72 | 1 | 3 | 4 | 2 |
| | 20 | Gini | 0.95 | 0.93 | 0.89 | 0.93 | 1 | 3 | 4 | 2 |
| | | NDCG | 0.94 | **0.95** | 0.90 | 0.91 | 2 | 1 | 4 | 3 |
| | | Spearman | 0.49 | 0.47 | 0.43 | 0.48 | 1 | 3 | 4 | 2 |
| | 50 | Gini | 0.98 | 0.93 | 0.97 | 0.98 | 2 | 4 | 3 | 1 |
| | | NDCG | **0.96** | **0.96** | **0.96** | 0.97 | 3 | 4 | 2 | 1 |
| | | Spearman | 0.31 | 0.33 | 0.30 | 0.32 | 3 | 1 | 4 | 2 |
| | 100 | Gini | 0.98 | 0.98 | 0.93 | 0.98 | 3 | 2 | 4 | 1 |
| | | NDCG | 0.93 | **0.96** | 0.91 | 0.92 | 2 | 1 | 4 | 3 |
| | | Spearman | 0.24 | 0.32 | 0.23 | 0.26 | 3 | 1 | 4 | 2 |
| Gaussian process | 10 | Gini | 0.34 | 0.88 | 0.83 | 0.85 | 4 | 1 | 3 | 2 |
| | | NDCG | 0.71 | **0.95** | **0.95** | 0.94 | 4 | 1 | 2 | 3 |
| | | Spearman | 0.29 | 0.70 | 0.65 | 0.67 | 4 | 1 | 3 | 2 |
| | 20 | Gini | 0.77 | 0.85 | 0.94 | 0.80 | 4 | 2 | 1 | 3 |
| | | NDCG | 0.79 | 0.94 | **0.95** | 0.88 | 4 | 2 | 1 | 3 |
| | | Spearman | 0.39 | 0.47 | 0.49 | 0.41 | 4 | 2 | 1 | 3 |
| | 50 | Gini | 0.71 | 0.74 | 0.92 | 0.76 | 4 | 3 | 1 | 2 |
| | | NDCG | 0.64 | 0.85 | **0.89** | 0.73 | 4 | 2 | 1 | 3 |
| | | Spearman | 0.23 | 0.32 | 0.30 | 0.26 | 4 | 1 | 2 | 3 |
| | 100 | Gini | 0.50 | 0.69 | 0.80 | 0.58 | 4 | 2 | 1 | 3 |
| | | NDCG | 0.53 | 0.74 | **0.79** | 0.63 | 4 | 2 | 1 | 3 |
| | | Spearman | 0.12 | 0.26 | 0.20 | 0.14 | 4 | 1 | 2 | 3 |

**Table 2.** NDCG values for the variable relevance rankings reached for all instances. The best values for each problem instance are highlighted using bold font. SR works best for the linear instances, RF works best for the non-linear instances. The values for 5% noise are shown in Table 1.

**Linear instances**

| | Noise = 0% | | | | Noise = 1% | | | |
|---|---|---|---|---|---|---|---|---|
| $d$ | LR | SR | RF | GPR | LR | SR | RF | GPR |
| 10 | 0.68 | **0.96** | 0.93 | 0.70 | **1.00** | **1.00** | 0.97 | **1.00** |
| 20 | 0.64 | **0.91** | 0.87 | 0.64 | **0.96** | **0.96** | 0.88 | **0.96** |
| 50 | 0.54 | 0.80 | **0.85** | 0.55 | 0.92 | **0.95** | 0.84 | 0.91 |
| 100 | 0.59 | **0.87** | 0.85 | 0.57 | 0.88 | **0.89** | 0.84 | 0.84 |
| | Noise = 10% | | | | Noise = 20% | | | |
| 10 | **1.00** | **1.00** | 0.99 | **1.00** | **1.00** | **1.00** | 0.99 | **1.00** |
| 20 | 0.93 | 0.89 | **0.94** | 0.93 | 0.97 | **0.98** | 0.95 | 0.97 |
| 50 | 0.93 | 0.93 | **0.94** | **0.94** | **0.97** | **0.97** | 0.96 | **0.97** |
| 100 | **0.97** | 0.95 | 0.93 | 0.96 | 0.94 | **0.96** | 0.94 | 0.93 |

**Non-linear instances**

| | Noise = 0% | | | | Noise = 1% | | | |
|---|---|---|---|---|---|---|---|---|
| $d$ | LR | SR | RF | GPR | LR | SR | RF | GPR |
| 10 | 0.90 | 0.93 | **0.98** | 0.91 | 0.73 | 0.93 | **0.99** | 0.96 |
| 20 | 0.70 | 0.89 | **0.90** | 0.88 | 0.73 | 0.84 | **0.95** | 0.90 |
| 50 | 0.74 | 0.86 | **0.90** | 0.78 | 0.58 | 0.77 | **0.82** | 0.71 |
| 100 | 0.55 | 0.81 | **0.83** | 0.64 | 0.55 | 0.74 | **0.76** | 0.64 |
| | Noise = 10% | | | | Noise = 20% | | | |
| 10 | 0.81 | **0.93** | 0.92 | 0.90 | 0.75 | 0.93 | **0.96** | 0.94 |
| 20 | 0.68 | 0.80 | **0.86** | 0.82 | 0.78 | 0.86 | 0.85 | **0.88** |
| 50 | 0.63 | 0.80 | **0.83** | 0.74 | 0.61 | **0.81** | 0.79 | 0.67 |
| 100 | 0.58 | 0.78 | **0.81** | 0.66 | 0.55 | 0.74 | **0.80** | 0.59 |

Analysis of the results shows that the order of regression methods is frequently similar for all three measures. For the linear problem instances, symbolic regression most often produced the best variable relevance ranking whereas for the non-linear instances random forest regression consistently produced the best relevance rankings.

An interesting result is that Gaussian process regression performed worse than random forest regression on the non-linear instances even though the data for these instances have been generated by sampling from Gaussian processes. The effect is especially visible for the high-dimensional problem instances. This indicates that our approach of maximum likelihood learning with automatic relevance determination does not work well for the identification of the relevant input variables in this case.

**Acknowledgements**

The authors gratefully acknowledge financial support by the Austrian Research Promotion Agency (FFG) and the Government of Upper Austria within the COMET Project #843532 Heuristic Optimization in Production and Logistics (HOPL).

## References

1. Campos, C.P.d., Ji, Q.: Efficient structure learning of Bayesian networks using constraints. Journal of Machine Learning Research 12(Mar), 663–689 (2011)
2. Chickering, D.M.: Learning Bayesian Networks is NP-Complete, pp. 121–130. Springer-Verlag (January 1996)
3. Elidan, G., Nachman, I., Friedman, N.: "Ideal Parent" structure learning for continuous variable Bayesian networks. Journal of Machine Learning Research 8(8), 1799–1833 (2007)
4. Friedman, J., Hastie, T., Tibshirani, R.: Sparse inverse covariance estimation with the graphical lasso. Biostatistics 9(3), 432–441 (2008)
5. Friedman, N., Nachman, I.: Gaussian process networks. In: Proceedings of the Sixteenth conference on Uncertainty in Artificial Intelligence (UAI). pp. 211–219. Morgan Kaufmann Publishers (2000)
6. Hand, D.J., Till, R.J.: A simple generalisation of the area under the ROC curve for multiple class classification problems. Machine Learning 45(2), 171–186 (2001), http://dx.doi.org/10.1023/A:1010920819831
7. Hofmann, R., Tresp, V.: Discovering structure in continuous variables using Bayesian networks. Advances in Neural Information Processing Systems (NIPS) pp. 500–506 (1996)
8. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems (TOIS) 20(4), 422–446 (2002)
9. Koivisto, M., Sood, K.: Exact Bayesian structure discovery in Bayesian networks. Journal of Machine Learning Research 5(May), 549–573 (2004)
10. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning. The MIT Press (2009)
11. Koski, T.J., Noble, J.: A review of Bayesian networks and structure learning. Mathematica Applicanda 40(1), 51–103 (2012)

12. Kronberger, G.: Symbolic Regression for Knowledge Discovery - Bloat, Overfitting, and Variable Interaction Networks. Reihe C: Technik und Naturwissenschaften, Trauner Verlag (2011)
13. Kronberger, G., Fink, S., Kommenda, M., Affenzeller, M.: Macro-economic Time Series Modeling and Interaction Networks, pp. 101–110. Springer Berlin Heidelberg (2011)
14. Kronberger G., Burlacu B., Kommenda M., Winkler S., Affenzeller M.: Measures for the Evaluation and Comparison of Graphical Model Structures. In: Moreno-Díaz R., Pichler F., Quesada-Arencibia A. (eds) Computer Aided Systems Theory – EUROCAST 2017. EUROCAST 2017. Lecture Notes in Computer Science, vol 10671, pp 283–290. Springer, Cham (2018)
15. Meinshausen, N., Bühlmann, P.: High-dimensional graphs and variable selection with the lasso. The annals of statistics pp. 1436–1462 (2006)
16. Rao, R., Lakshminarayanan, S.: Variable interaction network based variable selection for multivariate calibration. Analytica Chimica Acta 599(1), 24 – 35 (2007)
17. Singh, A.P., Moore, A.W.: Finding optimal Bayesian networks by dynamic programming. Tech. Rep. CMU-CALD-05-1062, School of Computer Science, Carnegie Mellon University (June 2005)
18. Spirtes, P., Glymour, C., Scheines, R.: Causation, Prediction, and Search. Springer-Verlag, New-York (1993)
19. Tsamardinos, I., Brown, L.E., Aliferis, C.F.: The max-min hill-climbing Bayesian network structure learning algorithm. Machine learning 65(1), 31–78 (2006)
20. Winker, S., Affenzeller, M., Kronberger, G., Kommenda, M., Wagner, S., Jacak, W., Stekel, H.: Variable interaction networks in medical data. In: Proceedings of the 24th European Modeling and Simulation Symposium EMSS 2012, pp. 265–270. Dime Universitá di Genova (2012)