Relational Blocking for Causal Discovery

Matthew J. H. Rattigan and **Marc Maier** and **David Jensen**
Department of Computer Science
University of Massachusetts Amherst
140 Governors Drive
Amherst, MA 01003

Abstract

Blocking is a technique commonly used in manual statistical analysis to account for confounding variables. However, blocking is not currently used in automated learning algorithms. These algorithms rely solely on statistical conditioning as an operator to identify conditional independence. In this work, we present *relational blocking* as a new operator that can be used for learning the structure of causal models. We describe how blocking is enabled by relational data sets, where blocks are determined by the links in the network. By blocking on entities rather than conditioning on variables, relational blocking can account for both measured and unobserved variables. We explain the mechanism of these methods using graphical models and the semantics of *d*-separation. Finally, we demonstrate the effectiveness of relational blocking for use in causal discovery by showing how blocking can be used in the causal analysis of two real-world social media systems.

1 Introduction

Conditional independence is a central concept for learning and reasoning with causal models (Pearl 2000; Spirtes, Glymour, and Scheines 2000). Explicit tests for conditional independence are the basic operators used in many algorithms for learning the structure of such models. These tests identify conditional independence by explicitly evaluating the impact of conditioning on specific sets of one or more observed variables.

In this paper, we present *relational blocking*, a fundamentally new algorithmic operator for learning conditional independence by exploiting relational structure among data entities. Relational blocking behaves in ways that differ fundamentally from conditioning. Specifically, it adjusts for sets of both observed and latent variables when they act as confounders. Yet it does not induce dependence when these variables are common effects.

Relational blocking formalizes approaches commonly used in the social sciences (Trochim 2006). Despite its widespread use in other fields, it has not been used in algorithms for learning the structure of causal models such as PC (Spirtes, Glymour, and Scheines 2000) or RPC (Maier et al. 2010). We describe relational blocking using recently introduced formalisms for describing directed graphical models of relational data (Heckerman, Meek, and Koller 2007), and we use these formalisms to show how blocking is distinct from simple conditioning. We demonstrate the effectiveness of relational blocking by showing how it reduces variability and adjusts for entire classes of observed and latent confounders. Finally, we examine the frequency with which relational blocking can be applied to discover causal dependencies in data describing social media systems.

2 Example

Consider the problem of understanding the operation of Wikipedia, a peer-produced encyclopedia of general knowledge.[1] Wikipedia articles, or pages, are produced collectively by thousands of volunteer users. Pages are created and modified by users, and users often organize themselves into groups called "projects," each of which covers a general topic. Within a project, individual pages are assessed by editors for "quality," an objective evaluation of key criteria.

One of the most persistent claims about Wikipedia is that its high quality stems from the large number of users that collaborate to write each article (Kittur and Kraut 2008). We call this the *many-eyes hypothesis*: The more users that revise an article, the higher the quality of that article. If we knew that this association was causal, then we could increase the quality of an article by directing more users to revise it. However, to determine that a causal dependence exists between editor count and article quality, we must eliminate other plausible alternative models that could explain an observed dependence.

A naive approach to this question would examine a large number of pages at a given point in time and estimate the dependence between the number of editors E and the quality of the page Q. This method tests the assumptions encoded in the graphical model shown in Figure 1a. Given this design, the variables are highly correlated: We sampled twenty random Wikipedia pages from ten projects, and found that a chi-square test yields χ^2=101.83 (n=189, since not all pages had Q and E values; DOF=12; $p = 2.44 \times 10^{-16}$), and approximately 66% of the variance of page quality could be attributed to the number of editors. This approach is quite

[1] http://www.wikipedia.org

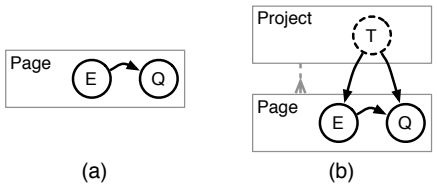

Figure 1: *(a) A simple graphical model describes the dependence between the number of editors E and quality Q of an article, but it does not account for common causes. (b) A more complex graphical model incorporates latent common causes T associated with project.*

similar to those conducted by many algorithms in machine learning—it identifies a statistical association between two variables, but that association is insufficient to establish a causal dependence. The observed dependence could stem from a common cause, such as the general topic area T. It is plausible that pages on topics of high interest to Wikipedians may be edited by a disproportionately large number of users (that is, T causes E). Additionally, that same interest in topic could drive editors to exert special care when editing, thereby improving quality (T causes Q). If T is a cause of both E and Q, then E and Q will be marginally dependent even if their dependence is not directly causal.

Unfortunately, since topic T is not a measured variable, we cannot account for its influence on E and Q through simple conditioning. However, since project structure is based on topic, we can adjust for this potential common cause by blocking. Projects govern pages that are thematically similar, so blocking on project can factor out the latent influence of topic. This alternative approach helps to differentiate between the graphical model shown in Figure 1a and the model in Figure 1b.

Figure 1, and other figures in the rest of this paper, are expressed as directed acyclic probabilistic entity relationship (DAPER) models (Heckerman, Meek, and Koller 2007). DAPER models combine the graphical conventions of entity-relationship (ER) diagrams, graphical models, and plate diagrams to show sets of probabilistic dependencies that can span the multiple entity types present in a relational data set. The DAPER model in Figure 1a contains only a single entity type and thus is equivalent to a conventional Bayesian network. However, the DAPER model in Figure 1b shows dependencies that span two entity types in which an instance of one entity (Project) typically connects to more than one instance of a second entity type (Page). A given Project instance is related to several Page instances, each of which contains an instance of the E variable. Each of those E variables has the same parent variable T on the given Project instance.

When we use project links to arrange pages into groups, we find that the average correlation between editor count and page quality decreases. A Cochran-Mantel-Haenszel test yields M^2=82.33 (n=189; DOF=12; $p = 1.48 \times 10^{-12}$). Although lower, this value is still highly significant, and roughly 53% of the variance would now be attributed to the number of editors. The effect size has dropped, but it is still statistically significant.

However, using this approach allows a stronger claim regarding the source of the association because we have plausibly factored out at least one potential (unmeasured) common cause. The ability to factor out multiple variables, observed or latent, is a key benefit of blocking. After ruling out several plausible common causes of variation, we now have much stronger evidence that the dependence between editor count and page quality is causal and that the many-eyes hypothesis is valid.

The example above highlights three concepts whose intersection forms the basis of this work. First, the Wikipedia data set is relational, made up of heterogeneous, interrelated data instances drawn from a relational network. Second, the question being investigated is causal. While there is a marginal association between editor count and quality, we are trying to establish a more powerful claim. Lastly, we were able to adjust for confounding factors (and draw a stronger causal conclusion) by using blocking as a complement to traditional conditioning.

3 Relational Blocking

At its core, blocking[2] is a data grouping strategy used to reduce variation and factor out common causes. The block design, originating in the agricultural experimental design work of Fisher (1935), divides data instances into disjoint groups, or blocks, according to the value of one or more blocking criteria. Within each block, confounding factors (often called "nuisance factors") associated with the blocking variable are held constant, reducing any variability in the outcome (effect) variable that is due to these factors. For example, the analysis of a drug trial might block on the hospital where the treatment was administered, allowing experimenters to control for any environmental factors associated with the facility.

In a network setting, units can be blocked using network structure as well. Relational blocking groups entities that share links with a common neighbor, called the *blocking entity*. Blocking in this manner can be used to facilitate causal discovery in network data sets consisting of entities (e.g., people, events, or places) that share some type of relationship or action among them. For example, papers written by common authors, or movies produced by the same studio, may form blocks. In this work, we focus on bipartite data sets, where entities are related in a one-to-many manner, and leave the analysis of alternative network structures for future exploration.

The use of relational structure to block by entities rather than attributes can be thought of as an extension of the classic twin design, in which pairs of twins are blocks. For more than a century, researchers have relied on twin data to account for whole classes of (often unmeasurable) attributes related to family environment and heredity (Boomsma, Busjahn, and Peltonen 2002).

[2]The term blocking is overloaded in the statistical sciences. In this paper, blocking refers to instance grouping, and is distinct from the concept of path blocking found in the graphical models literature.

The benefit of relational blocking is twofold. The first is statistical: By organizing experimental units into groups such that variability within each block is reduced, we can improve statistical power. Relational blocks hold constant any attribute associated with the blocking entity. In this respect, blocking serves the same purpose as conditioning. However, unlike conditioning, blocking can simultaneously adjust for the influence of several (even latent) variables. When applied to hierarchical domains (such as the synthetic domains described in Section 4), relational blocking serves a similar purpose to multilevel modeling, where the influence of factors associated with common group or entity is modeled within the appropriate regression equation associated with each level of the hierarchy (Goldstein 1995).

The second benefit relates to causal reasoning. The *causal sufficiency assumption* (Spirtes, Glymour, and Scheines 2000) states that any possibly confounding variables are observed. When blocking, factors that are held constant within each block can be eliminated as possible common causes of treatment and outcome, enabling stronger claims of causal sufficiency and pruning the space of alternative causal models. By eliminating entire classes of potential common causes, including both measured and latent variables, the causal sufficiency assumption is relaxed, in that confounding factors can be accounted for even if they are unobserved.

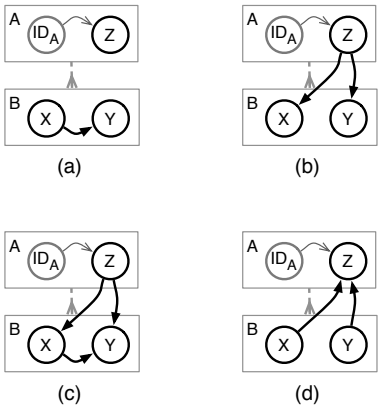

Figure 2: *Different generative models for bipartite one-to-many data. In case (a), X directly influences Y. In (b), X and Y have a common cause (Z), and blocking and conditioning will both render them conditionally independent. In (c), blocking and conditioning are able to factor out the influence of confounder Z, but the two remain conditionally dependent. Case (d) depicts Z as a common effect of X and Y; here, X and Y are rendered dependent when conditioned on Z (Berkson's paradox), yet remain independent when Z is held constant through blocking using entities of type A. In all models, the thin gray arrows represent the deterministic dependence between ID_A and Z.*

4 Blocking vs. Conditioning

It may be tempting to view blocking merely as a form of conditioning. While the two serve common purposes—reducing variability and adjusting for common causes—they do not produce the same statistical results. To illustrate this point, we generate synthetic bipartite data and compare the results of blocking and conditioning for different generative models of attribute structure. Each data set consists of entities of two types, A and B, connected in a one-to-many manner. In all cases, there are 10,000 B entities, with the number of A entities varying between different experiments. Each A entity carries two attributes, Z and H, with the former considered measured and the latter latent. The B entities also have two attributes, X and Y, both of which are measured.

In each experiment, the goal is to assess the dependence between X and Y while either blocking on A or conditioning on Z. Note that Z is generated as a continuous variable; in each experiment it is discretized to a fixed number of levels in order to compare the results of blocking and conditioning using the same hypothesis test (we use Guo's weighted Pearson's r correlation (2003)). While not presented here, we found that the results of experiments using partial correlation with an untransformed Z were qualitatively similar.

To represent blocking with a graphical model, we introduce an identity variable ID_A (Rattigan and Jensen 2010). The models in Figure 2 depict bipartite, one-to-many models with the identifier variable included. With this framework, we can formally define relational blocking:

Definition 1 *Let A and B be two entity sets in a k-partite network. A block contains a set of B entities that link to a common A entity. Let ID be the unique identifier of a block, and let X and Y be two attributes of B.* ***Relational blocking*** *is a process that evaluates the conditional independence of X and Y given ID by grouping B entities into disjoint blocks.*

The thin gray directed edge connecting ID_A and Z denotes a *deterministic* dependence between the two. Certainly, ID_A determines Z, since the value of ID_A indicates the value of Z with a simple lookup. The reverse is not true, however, as several A entities may share the same value of Z while having different identifiers.

Despite being common in real data sets, the consequences of determinism in graphical models is rarely discussed in the machine learning literature. The presence of deterministic dependence slightly complicates the rules of d-separation.[3] The following definition is adapted from Spirtes et al. (2000), and Geiger (1989):

Definition 2 *Let $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{W}$ be three disjoint sets of vertices in DAG G. Let Det($\mathbf{V}$) be the set of all variables determined by $\mathbf{V}$. Then, $\mathbf{X}$ and $\mathbf{Y}$ are* ***d-separated*** *by $\mathbf{W}$ if and only if for all undirected paths P between $\mathbf{X}$ and $\mathbf{Y}$ either (1) $\exists v \in colliders(P)$ such that $v \wedge descendants(v) \notin \mathbf{W}$ or (2) $\exists v \in noncolliders(P)$ such that $v \in Det(\mathbf{W})$.*

4.1 Common Causes

The first set of experiments simulate the scenario outlined in the introduction. Figures 2a and 2b represent two generative

[3] Some authors use the term *D-separation* (with a capital 'D') to denote d-separation with determinism; in this work, we will not rely on this typographical convention.

models where X and Y are marginally dependent, denoted $X \not\perp\!\!\!\perp Y$. In the first case, X has direct influence on Y; in the second, their marginal dependence is due to a common cause.

Under the framework of d-separation (Pearl 1988), this marginal dependence is evident from the existence of a collider-free path connecting X and Y in either case. From data, we can differentiate the two models with a conditional independence test. Conditioning on Z has no effect on the independence relationship between X and Y in model 2a, but interrupts the d-connecting path in model 2b, rendering X and Y conditionally independent: $X \perp\!\!\!\perp Y \mid Z$.

The data for model 2b are generated such that $X, Y = \beta_Z Z + \epsilon$. For all values of β_Z, blocking is comparable to conditioning in terms of Type I error, maintaining an error level of less than 7% for $\alpha = 0.05$, with conditioning less than 6%.

This similarity in performance can be explained by the semantics of d-separation and the observation that, as defined above, blocking is equivalent to conditioning on ID_A. When conditioning on variable Z, data are grouped such that the value of Z is held constant within each group. Similarly, blocking holds constant the entity A within each group. In model 2b, Z lies on the only d-connecting path between X and Y. Per the above definition, conditioning on Z or any set of variables that determines Z will render X and Y conditionally independent. Since ID_A fully determines Z, conditioning on it (that is, blocking) will d-separate X and Y.

4.2 Common Effects

An additional case is described by the model shown in Figure 2d. In this case, X and Y are marginally independent, while Z is generated such that $Z = \beta X' + \beta Y' + \epsilon$, where X' and Y' are the sums of the values of the X and Y values for each related B entity. This case presents an example of Berkson's paradox (Berkson 1946), where conditioning on a common effect (i.e., collider) will induce dependence between marginally independent variables. Here, blocking and conditioning lead to opposite conclusions. As expected, conditioning on Z does indeed induce dependence between X and Y; however, blocking on A does not, even though doing so effectively adjusts for variable Z as in the conditioning case.

These effects can be seen in Figure 3. Conditioning produces the expected result: As we increase the strength of effect parameter β, conditioning induces a dependence between X and Y more frequently. Blocking, on the other hand, does not produce any of the conditional dependence described by Berkson's paradox. The d-separation criteria stated above agree with our empirical results—conditioning on the collider Z creates a d-connecting path, while blocking (conditioning on ID_A) does not.

The differences between blocking and conditioning cannot be attributed to statistical power. For the case presented above, the block size (10 instances) is an order of magnitude smaller than the conditioning groups (100). To compensate for this difference, we randomly split each conditioning group into subgroups of 10 instances (labeled as "split" in Figure 3). Even with conditioning groups of equal size

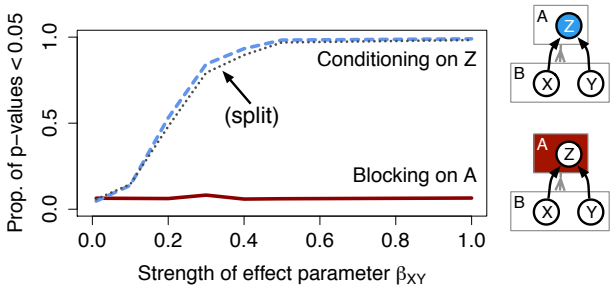

Figure 3: *Unlike conditioning, blocking does not induce conditional dependence when holding constant a common effect of two marginally independent variables. The line labelled* split *indicates a conditioning analysis with statistical power identical to the blocking analysis.*

to the blocks, the proportion of significant p-values is unchanged.

These results clearly indicate that blocking and conditioning are fundamentally different operations. The difference between blocking and conditioning is illustrated in Figure 4. For a small dataset generated under the model in Figure 2d, the data have been stratified into contingency tables for both blocking and conditioning. Even for this illustrative example, the results of statistical tests can differ, as the p-value for the conditioning case is 0.009 (indicating significance at the 0.01 level), compared to 0.033 for blocking (not significant at 0.01).

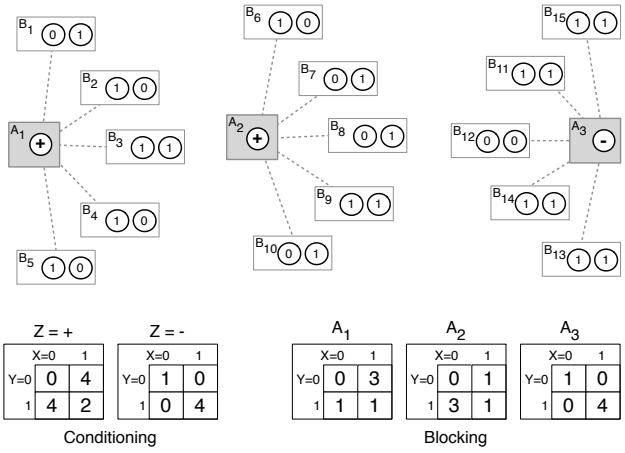

Figure 4: *Blocking and conditioning are distinct operations, as they stratify the data in different ways. For the above relational data set, conditioning groups the data into two strata, yielding a combined χ^2 value of 9.44 (p=0.009) while blocking groups the data into three strata, producing a χ^2 value of 8.75 (p=0.033).*

4.3 Latent Confounders

Conditioning and blocking do not perform equivalently in the presence of latent variables. Figures 5a and 5b depict generative models for data with a latent variable H acting

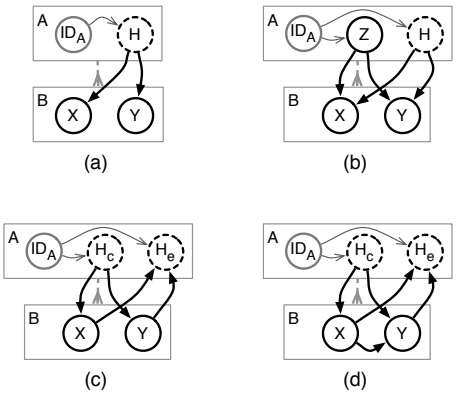

Figure 5: *Models for bipartite data with latent variables. Models (a) and (b) depict cases where a latent common cause H exerts influence on X and Y. In these cases, blocking is able to render X and Y conditionally independent, while conditioning does not. In models (c) and (d), X and Y have both a latent common cause H_c and a latent common effect H_e. Here, blocking will distinguish between the two models.*

as a common cause of both X and Y. Since H is unobserved, conditioning is impossible for model 5a, while blocking performs as if it is controlling for an observable variable. In the case of model 5b, both a measured (Z) and latent (H) variable exert influence on X and Y, such that $X, Y = \beta_Z Z + \beta_H H + \epsilon$. The plot in Figure 6 depicts Type I error rate at the α=0.05 level with β_Z held constant at 0.5, and β_H varying from 0 to 0.5. Since blocking accounts for all confounders, it can be used to establish conditional independence in the presence of unmeasured factors. Thus, in cases where two variables are marginally dependent, conditioning alone is inadequate for ruling out alternative models such as those in models 5a or 5b.

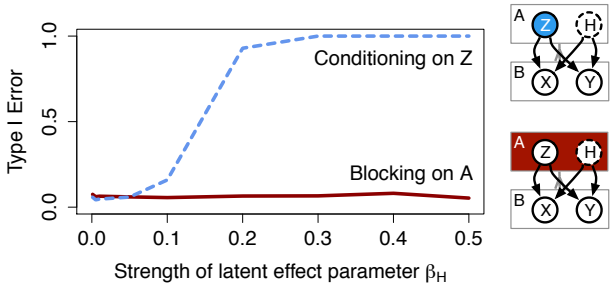

Figure 6: *The effects of blocking and conditioning differ for data generated under the models shown in Figure 5b. Conditioning can only adjust for measured variable Z, and is susceptible to high rates of Type I error as the strength of the latent effect β_H increases. Blocking accounts for both H and Z, it is not affected by β_H.*

The models depicted in 5c and 5d show cases where X and Y have both a latent common cause (H_c) and latent common effect (H_e). In both cases, X and Y are marginally dependent. Blocking renders X and Y conditionally independent for model 5c, but not 5d. As a result, any finding that $X \not\perp\!\!\!\perp Y \mid ID_A$ cannot be "explained away" by the presence of (latent) common effects when blocking (this property follows directly from the results detailed in Section 4.2). Thus, while blocking can adjust for multiple latent confounders, it introduces no threat to causal conclusions in the presence of latent common effects.

4.4 Power

The small example in Figure 4 illustrates another distinction between blocking and conditioning: Since identifiers and variables are related in a non-injective manner, blocking necessarily stratifies the data into smaller groups. To investigate the effects of the smaller groupings on statistical power, we generated synthetic data using the model found in Figure 2a such that $Y = \beta_X X + \epsilon$. Figure 7 depicts statistical power as a function of effect size, sample size, and block size. In each case, blocking does slightly decrease statistical power, which is expected given the smaller strata. However, given the large size of many modern relational data sets such as Wikipedia, these effects of this decrease are minimal.

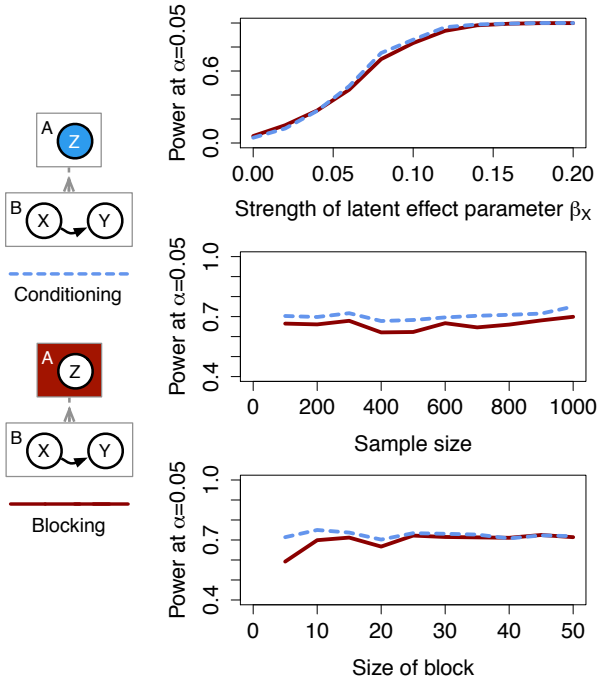

Figure 7: *Although relational blocking groups the data into smaller strata than conditioning, there is little effect on statistical power.*

5 Blocking in Practice

To assess the practical utility of relational blocking, we analyzed two domains derived from the peer-production systems Wikipedia and Stack Overflow. Each data set was comprised of multiple related entity types and attributes. The data schema for each can be found in Figure 8. Blocking was applicable to 80% of the questions identified by practitioners

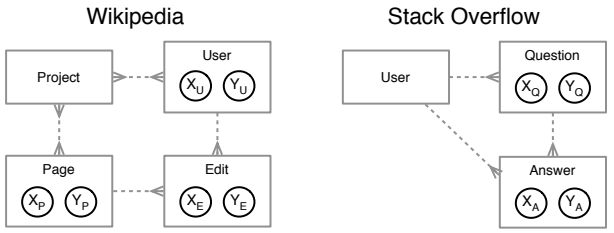

Figure 8: *Data schemata for Wikipedia and Stack Overflow. Each pair of X and Y variables within to the same entity can be tested for dependence, and related parent entities can be used for blocking. For example, Wikipedia Page.Quality and Page.Edits can be blocked through Project or User, while Stack Overflow Question.Score and Question.Length can only be blocked through User.*

as the most interesting, and blocking produced substantial changes in results in 28% of the quantitative assessments of actual causal dependencies.

Table 1: Details of Wikipedia data

Entity	Attributes	Block Ents.
Page	Adopted by Project, Age, Assessment, Editors, Edits, Featured, Importance, Length Notice, Number of Links, Protected, Quality, Views	Project, User
User	Role, Edits, Membership in Project	Page, Project
Edit	Size, Vandalism, Minor, Reverted	Page, User

5.1 Wikipedia

Although Wikipedia has been the subject of several recent studies (e.g., Kittur 2008), we know very little about how it functions, particularly from a causal standpoint. These aspects make Wikipedia an ideal candidate for studying the applicability and utility of relational blocking.

Our version of the data contained User entities and Edit events in addition to the Pages and Projects discussed in Section 2. The details of the entity types and associated attributes can be found in Table 1. In all, there are twenty attributes that are applied to three target entity types (Projects lack intrinsic attributes of their own, and are only used as blocking entities). This schema allows for 174 different relationships apropos to the bipartite models illustrated in Figure 2, for which 348 blocking schemes are available (each X, Y attribute pair can be blocked with two different entities).

We took a qualitative approach to determining the applicability of relational blocking. We surveyed ten people, each with a bachelors or masters degree in Computer Science, to obtain a sample of interesting causal questions in the Wikipedia domain. Respondents were given a simple list of attributes and asked to indicate ten pairs of causes and effects they found compelling for study (attributes were presented in one of five random orderings to eliminate biases associated with presentation). The group generated a list of 99 causal discovery tasks (one respondent provided only 9 tasks), 71 of which were unique. Of these, 57 (80%) can be addressed with a simple relational blocking approach such as the one outlined in Section 2. While not definitive, these results indicate that relational blocking can be readily applied to the types of problems that interest practitioners.

Table 2: Details of Stack Overflow data

Entity	Attributes	Block Ents.
Question	Ans. Count, Mean Ans. Score, Mean Ans. Comment Count, Mean Ans. Length, Comment Count, Favorite Count, Has Accepted Ans., Length, Score, View Count	User
Answer	Accepted, Comment Count, Score, Length	Question, User

5.2 Stack Overflow

In addition to the Wikipedia data set discussed in Section 2, we examined data from Stack Overflow, an online technical resource that allows users to pose questions as well as answer others' questions.[4] For our study, we examined dependence between attributes on Questions (blocking on Users) as well as attributes on Answers (blocking on Users or Questions), and found a significant change in effect size in 28% of all cases. The complete list of attributes is found in Table 2.

For each of the 57 same-entity attribute pairs, we assessed their marginal and conditional independence using all available data for the month of March 2010. For pairs of continuous attributes (e.g. Score, Comment Count), we utilized a blocked Pearson's r statistic (Guo 2003); for nominal attributes, we applied a Cochran-Mantel-Haenszel test. When one attribute was continuous and the other nominal, we discretized the continuous attribute to five levels using agglomerative clustering). In all cases, experiments involving Question entities had a sample size greater than 50k, while those involving Answer entities had samples of over 100k. Given these large samples, p-values for even the smallest effect sizes were significant, so we focused on associations with marginal effect sizes greater than 0.1 (the effect size for both statistics can be measured on a scale of 0.0–1.0).

Of the 57 attribute pairs, twenty exhibited a marginal association greater than 0.1. Of these, sixteen (28%) demonstrated a strong reduction in the size of effect when blocking, suggesting a dependence structure similar to the model found in Figure 2c (albeit with a latent Z). For instance, Question Score and View Count exhibit an effect size of $r^2 = 0.51$ in the marginal case, but this drops to 0.12 in the conditional case (the associated z-scores are 214.16 and 59.49, respectively; both p-values are significant at the 1×10^{-8} level). This result suggests that while Score and

[4] http://stackoverflow.com

View Count are associated, latent attributes on the Question author (e.g., expertise, writing style) are a common cause for both and explain most of the variation.

Four attribute pairs exhibited little change in effect size when blocking was applied, which provides evidence for the model in Figure 2a. For instance, the Score of a provided Answer is highly associated with Accepted status. Authors of Stack Overflow Questions can optionally "accept" a good Answer from among those provided; since many choose to accept the one with the highest score, this result is not surprising.

6 Conclusions and Future Work

In this work, we have presented relational blocking as a technique to facilitate learning the structure of causal models. Blocking is similar in function to simple conditioning in its ability to reduce variability and increase statistical power. However, unlike conditioning, blocking does not induce dependence when accounting for common effects. Blocking is able to adjust for whole classes of confounders simultaneously, whether observed or latent, effectively relaxing the causal sufficiency assumption and strengthening causal conclusions.

We have illustrated the use of blocking using synthetic data and found our approach to perform well in terms of Type I and Type II error. Furthermore, by explaining our results using the graphical models framework and d-separation criteria, we are able to provide a theoretic understanding of a commonly used technique employed in the social sciences. In addition, we have demonstrated the utility of blocking on two real world data sets.

Our approach is currently limited to relational data sets with one-to-many relationships. We currently see two possible methods for extending the work to more complex network structures containing many-to-many relationships: (1) By sampling links we can create a one-to-many projection of any graph, allowing us to block; (2) By grouping *blocking* entities as a preliminary step, we create blocks that share common *sets* of neighbors rather than a single parent entity. Blocking can also extend to data incorporating spatial and temporal dynamics in addition to dyadic relationships. Finally, we have described blocking as a new algorithmic operator, and the next logical step is to incorporate it into a constraint-based system for fully automated causal discovery in complex data sets.

7 Acknowledgements

We thank Cynthia Loiselle for her assistance in preparing this manuscript. In addition, we thank Peter Spirtes for his insights into the mechanisms of d-separation under determinism. This material is based on research sponsored by the Air Force Research Laboratory under agreement number FA8750-09-2-0187. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusion contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory or the U.S. Government.

References

Berkson, J. 1946. Limitations of the application of fourfold table analysis to hospital data. *Biometrics Bulletin* 47–53.

Boomsma, D.; Busjahn, A.; and Peltonen, L. 2002. Classical twin studies and beyond. *Nature Reviews Genetics* 3:872–882.

Fisher, R. A. 1935. *The Design of Experiments*. Edinburgh: Oliver and Boyd.

Geiger, D.; Verma, T.; and Pearl, J. 1989. *Identifying Independence in Bayesian Networks*. Wiley Online Library.

Goldstein, H. 1995. *Multilevel Statistical Models*. Arnold London.

Guo, J. 2003. Four correlation coefficients with a third blocking variable: Their efficacy, relative efficiency, and test statistics. *Communications in Statistics: Theory and Methods* 32(9):1835–1858.

Heckerman, D.; Meek, C.; and Koller, D. 2007. Probabilistic entity-relationship models, PRMs, and plate models. *Introduction to Statistical Relational Learning* 201–238.

Kittur, A., and Kraut, R. E. 2008. Harnessing the wisdom of crowds in Wikipedia: Quality through coordination. In *Proceedings of the ACM 2008 Conference on Computer Supported Cooperative Work*, 37–46.

Maier, M.; Taylor, B.; Oktay, H.; and Jensen, D. 2010. Learning causal models of relational domains. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Pearl, J. 2000. *Causality: Models, Reasoning, and Inference*. New York, NY: Cambridge University Press.

Rattigan, M., and Jensen, D. 2010. Leveraging d-separation for relational data sets. In *2010 IEEE International Conference on Data Mining*, 989–994. IEEE.

Spirtes, P.; Glymour, C.; and Scheines, R. 2000. *Causation, Prediction and Search*. Cambridge, MA: MIT Press, 2nd edition.

Trochim, W. M. 2006. The Research Methods Knowledge Base, 2nd Edition. www.socialresearchmethods.net/kb/.