Causal Rule Sets for Identifying Subgroups with Enhanced Treatment Effect

Tong Wang University of Iowa tong-wang@uiowa.edu Cynthia Rudin Duke University cynthia@duke.cs.edu

Abstract

We introduce a novel generative model for interpretable subgroup analysis for causal inference applications, Causal Rule Sets (CRS). A CRS model uses a small set of short rules to capture a subgroup where the average treatment effect is elevated compared to the entire population. We present a Bayesian framework for learning a causal rule set. The Bayesian framework consists of a prior that favors simpler models and a Bayesian logistic regression that characterizes the relation between outcomes, attributes and subgroup membership. We find maximum a posteriori models using discrete Monte Carlo steps in the joint solution space of rules sets and parameters. We provide theoretically grounded heuristics and bounding strategies to improve search efficiency. Experiments show that the search algorithm can efficiently recover a true underlying subgroup and CRS shows consistently competitive performance compared to other state-of-the-art baseline methods.

1 Introduction

Throughout society, we are paying increasing attention to personalized therapies, targeted advertising, and other types of personalized recommendations. When estimating the efficacy of a treatment, researchers are not only interested in conclusions that apply to the whole population but are also looking for subgroups on which the treatment is especially effective. This is because a treatment may have heterogeneous effect on different groups of people, which is very common in many real-world problems. One often-cited example in oncology is trastuzumab which has been shown to be an effective treatment for breast cancer only when the tumor is HER2 positive [1]. This example is illustrative of the necessity to identify such subgroups if they exist.

Problems related to this are referred to as subgroup analysis or subgroup-treatment effect interactions [2]. One challenge faced in subgroup analyses is multiplicity, due to a large number of subgroups typically examined. The problem of multiplicity is more severe when there is a large number of predictors/attributes. If a subgroup is

defined by a rule (a conjunction of conditions), then the number of possible subgroups increases exponentially with the number of attributes. This can potentially pose a serious problem when working with data of large dimensions.

Two classes of methods have been developed recently for identifying subgroups with enhanced treatment response. One class of methods called moderated regression analysis [3], fit statistical regression models that include possible treatment-feature interactions. However, the methods mainly test the hypotheses of known subgroups instead of discovering unknown subgroups from data. It relies on clear prior hypotheses about which subgroups could be involved in the interactions. Such a priori hypotheses may not exist in many applications, and we would want to learn these subgroups from data. Particularly especially when the number of features is large and it becomes harder for the domain expert to identify which possible subgroups to consider. The other class of widely adopted approaches uses tree-based recursive partitioning. They partition data into smaller subsets until a stopping criterion is met. Recursive partitioning methods are a natural way to analyze a large number of features that have potentially complicated interactions. But they usually partition data greedily, and the splits are not based on the treatment effect further down in the tree. The downfall of being greedy is that there is no direct optimization of a global objective.

To address the shortcomings of previous methods, we propose a model called Causal Rule Sets (CRS) that captures a subgroup with a set of rules: an observation is in the subgroup if it satisfies at least one rule in the set. An example of a CRS model is shown below where the treatment is a coupon, and the outcome is whether the customer buys a product.

if a customer (is older than 40 AND has children AND annual income <\$50,000)

OR (is female AND married AND used the coupon before) **then**

the customer is more highly influenced by the coupon than the general population.

end if

The model consists of two rules, each rule being a conjunction of conditions. The rules capture a subgroup where a customer satisfies at least one of the rules.

Instead of subgroups being defined by one rule in previous recursive partitioning models, a rule set defines a more flexible concept of a subgroup. illustrate it using the example below. Imagine data distributed on a 2D plane and using one rule places an implicit constraint on identified subgroups: they are limited to points that fit into one rectangle. For example, " $0.2 \le X_1 \le 0.4 \text{ AND } 0.3 \le X_2 \le 0.6$." is represented be the red rectangle in Figure 1(a). If the subgroup has other irregular shapes, for example, distributed along an arc, as represented by black dots in Figure 1(a), it cannot be captured by a single rule. However, subgroups of irregular shapes can be identified by allowing multiple rules. It is easy to find a set of smaller rectangles that collectively cover the subgroup as shown in Figure 1(b). As the shape becomes more irregular, it becomes a question of how to place a set of rectangles to cover the points. Allowing more rectangles will cover the points more accurately but at the cost of model simplicity, which can be controlled in our model via a prior. A rule set model brings greater freedom in defining a subgroup and can potentially uncover more complicated interactions among features.

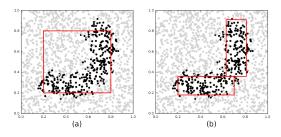


Figure 1: An example of a subgroup. Black dots represent a true subgroup, and gray dots represent the rest of the population. (a) corresponds to one rule. (b) corresponds to a model of a rule set with two rules.

In this work, we use a Bayesian framework to learn a CRS model for its appealing property that it provides a single unified framework to jointly optimize fitting to the data and model complexity without directly "hard" controlling either. We propose an efficient inference method for learning a maximum a priori (MAP) model. The algorithm consists of two steps to effectively reduce computation. First, a rule space is constructed via rule mining. Then, the search algorithm iterate over states that consist of rule sets and the corresponding pa-

rameters until convergence and applies an explorationwith-exploitation strategy to improve search efficiency. To further reduce computation, a simpler metric is developed to replace the more computationally heavy objective during the search.

Experiments were conducted to test the ability of CRS to retrieve true subgroups. We also compared CRS model with other baseline models using real-world data sets. CRS demonstrated competitive out-of-sample performance compared to baseline methods while offering greater flexibility of obtaining subgroups with various support and average treatment effect.

2 Related Work

There has been extensive research on subgroup analysis in the context of causal inference [4–9]. An earlier and classic type of methods pertains to the situation where clear *a priori* hypotheses exist about which subgroups are involved in the interactions, and then statistical methods are applied to assess the hypotheses [10, 11].

More recent work performs post hoc analysis that automatically identifies subgroups from data. These methods are more useful for discovering unusual or unexpected results. Most work on post hoc analysis is tree-structured recursive partitioning methods, which partition data greedily with various heuristics proposed in different works. These methods induce subgroups involved in a treatment-subgroup interaction from the data. A few recent and widely cited works include: Subgroup Identification based on Differential Effect Search (SIDES) [12], Virtual Twins (VT) [13], and Qualitative Interaction Trees (QUINT) [14]. SIDES greedily partitions a database into two subgroups at each parent group, such that the treatment effect within one of the two subgroups is maximized compared with the other one. QUINT builds a binary tree to partition data into three subgroups, where a treatment is better, worse, or similar to the alternative. QUINT and SIDES differ mainly in their partitioning criteria. VT methods first use two random forests to estimate the probability of positive outcomes in treatment and control groups respectively and then use the difference of the two as a target variable to build a regression or classification tree. These methods are baselines in this paper. The all produce subgroups captured by one leaf/rule. Neither builds interpretable models as in the example model provided above.

3 Preliminaries

We work with data $S = \{(\mathbf{x}_i, y_i, T_i)\}_{i=1}^n$ comprised of n instances, each of which corresponds to an observation described by a covariate vector $\mathbf{x}_i \in \mathbb{R}^{J+1}$ that consists of J features and a constant to account for the intercept

in the Bayesian logistic regression. Assume ignorability and SUTVA [15]. Let $T_i \in \{0,1\}$ denote the treatment assignment for instance i. The outcome is represented by $y_i \in \{0,1\}$. We use the potential outcomes framework [16] with potential outcomes $y_i(1), y_i(0) \in \{0,1\}$ under treatment and control, and define the treatment effect at x as

(3.1)
$$\tau(x) = \mathbb{E}[y_i(1) - y_i(0)|\mathbf{x}_i = x].$$

Given a dataset S, our goal is to find a set of rules A to capture a subgroup that demonstrates an enhanced treatment effect. A rule is a conjunction of conditions and the number of conjunctions is referred to the length of the rule. For example, "female AND age > 50 AND married" is a rule of length 3. Let $a(\mathbf{x}_i) \in \{0,1\}$ represent if instance i satisfies rule a or as we also call it, "covered" by rule a. Let A denote a rule set. An instance satisfies the rule set (or is "covered" by a rule set) if it satisfies at least one rule in the set, represented as below.

(3.2)
$$A(\mathbf{x}_i) = \begin{cases} 1 & \exists a \in A, a(\mathbf{x}_i) = 1 \\ 0 & \text{otherwise.} \end{cases}$$

Let I represent the indices for data S and I_A represent indices for instances in the subgroup defined by A,

(3.3)
$$I_A = \{i \in I | A(\mathbf{x}_i) = 1\}.$$

4 Causal Rule Sets

We propose a Bayesian framework for learning a causal rule set A. The Bayesian approach turns this problem into finding a MAP solution P(A|S,H), given data S and a set of hyperparameters denoted by H. The model consists of a prior p(A) and a Bayesian logistic regression $\Theta(S;A,\mathbf{w})$ for modeling the conditional likelihood of data. We now describe them in detail.

4.1 Prior We use a generative process described in [17] for the prior that favors a simple model, i.e., a *small* set of *short* rules. A small model is easy to interpret since it contains fewer conditions to comprehend. Meanwhile, the shorter a rule, the larger the support, which naturally avoids overfitting. Let \mathcal{A} denote a rule space partitioned into pools of rules with equal lengths, indexed by rule length l. $\mathcal{A} = \mathcal{A}_1 \cup \cdots \mathcal{A}_L$, L being the maximum rule length a user allows.

Assume the interpretability of a rule is only associated with its length. Then rules in the same pool A_l are drawn uniformly randomly with probability p_l , on which we place a beta prior, yielding

(4.4) A rule selected from
$$A_l \sim \text{Bernoulli}(p_l)$$
,

(4.5) For
$$l \in \{1, ..., L\}, p_l \sim \text{Beta}(\alpha_l, \beta_l)$$
.

Let M_l notate the number of rules drawn from pool A_l . Then the prior for A is

(4.6)

$$p(A) = \prod_{l=1}^{L} p(M_l; \alpha_l, \beta_l) \propto \prod_{l=1}^{L} B(M_l + \alpha_l, |\mathcal{A}_l| - M_l + \beta_l).$$

 $B(\cdot)$ represents the Beta function. $\{\alpha_l, \beta_l\}_{l=1}^L$ jointly control the expected number of rules in A. We usually choose $\alpha_l \ll \beta_l$ so that the model tends to choose fewer rules from each pool.

4.2 Conditional Likelihood We use a Bayesian Logistic Regression model to control for confounding covariates and characterize the effect of receiving the treatment and being in the subgroup, yielding:

$$p(y_i = 1 | \mathbf{x}_i, T_i) = \sigma \Big(\mathbf{v} \mathbf{x}_i + \gamma^{(0)} T_i + \gamma^{(1)} A(\mathbf{x}_i) + \gamma^{(2)} T_i A(\mathbf{x}_i) \Big),$$
(4.7)

where $\sigma(\cdot)$ is a sigmoid function. The exponent in formula (4.7) models the interaction between the attributes, treatment assignment T_i and the rule set A. v is a vector of coefficients for attributes including an intercept. $\mathbf{v}\mathbf{x}_i$ captures the baseline contribution from the attributes towards getting a good outcome $y_i = 1$, regardless of whether receiving a treatment. $\gamma^{(1)}$ can be regarded as a measurement of additional treatment effect for being in subgroup A. $\gamma^{(0)}T_i$ captures the baseline treatment effect on the entire population. $\gamma^{(1)}A(\mathbf{x}_i)$ models the effect of being in subgroup I_A on the outcome, which accounts for the potential confounding effect of being in subgroup I_A . $\gamma^{(2)}T_iA(\mathbf{x}_i)$ implies that the additive contribution to the outcome exists only when instance i receives the treatment, i.e., $T_i = 1$, and it belongs to the subgroup, i.e., $A(\mathbf{x}_i) = 1$.

Let $\mathbf{w} = \{\mathbf{v}, \gamma^{(0)}, \gamma^{(1)}, \gamma^{(2)}\}$. Assuming data is i.i.d, the conditional likelihood of data S is

(4.8)

$$\mathcal{L}(S; A, \mathbf{w}) = \prod_{i=1}^{n} p(y_i = 1 | \mathbf{x}_i, T_i)^{y_i} \cdot p(y_i = 0 | \mathbf{x}_i, T_i)^{(1-y_i)},$$

parameterized by:

(4.9)
$$A \sim \text{Prior}(A) \text{ and } \mathbf{w} \sim \mathcal{N}(\mu, \Sigma).$$

We define this partial posterior that does not include the prior information for A:

(4.10)
$$\log \Theta(S; A, \mathbf{w}) = \log \mathcal{L}(S; A, \mathbf{w}) + \log p(\mathbf{w}),$$

which considers both the fit for the data and the regularization. If μ is set to 0, the above form is reduced to logistic regression with l_2 regularization.

We cast this as a machine learning problem, which means our goal is for the CRS model to perform well with respect to the likelihood on data drawn from the population distribution $p(\mathbf{x})$ (out of sample).

5 Model Fitting

We describe how to find an optimal rule set A^* that maximizes the posterior p(A|H,S), where H denotes a set of hyperparameters, $H = \{A, L, \{\alpha_l, \beta_l\}_{l=1}^L, \Sigma, \mu\}$. This is equivalent to maximizing:

(5.11)
$$F(A, \mathbf{w}; S, H) = \log \Theta(S; A, \mathbf{w}) + \log p(A; H).$$

Given a rule set A, the corresponding parameters are obtained by maximizing $F(A, \mathbf{w})$, which is equivalent to maximizing $\Theta(A; \mathbf{w})$ since only $\Theta(A; \mathbf{w})$ in $F(A, \mathbf{w})$ depends on \mathbf{w} . Let \mathbf{w}_A denote the optimal parameters for a given rule set A. \mathbf{w}_A is estimated via

(5.12)
$$\mathbf{w}_A = \max_{\mathbf{w}} \Theta(A, \mathbf{w})$$

Our goal is to find an optimal pair (A^*, \mathbf{w}_{A^*}) , such that

(5.13)
$$A^*, \mathbf{w}_{A^*} = \max_{A, \mathbf{w}} F(A, \mathbf{w}).$$

The main search procedure follows the steps of simulated annealing which generates a sampling chain that starts from a random state and proposes the next state by selecting from the neighbors. In the context of our model, each state is defined as $s^{[t]} = (A^{[t]}, \mathbf{w}_{A^{[t]}})$, where t is the time stamp. Given a rule set $A^{[t]}$, a âĂIJneighboringâĂİ solution is a rule set whose edit distance is 1 to the current rule set (one of the rules is different). Therefore, $A^{[t+1]}$ is generated by adding, removing or replacing a rule from $A^{[t]}$. Then $\mathbf{w}_{A^{[t+1]}}$ is obtained via formula (5.12). Given a temperature schedule function over time steps, $T(t) = T_0^{1-\frac{t}{N_{\text{iter}}}}$, the proposed state $(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}})$ is accepted and assigned to $s^{[t+1]}$ with probability $\min(1, \exp\{\frac{F(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}}) - F(s^{[t]})}{T(t)}\})$. The chain converges to the optimal solution as the temperature cools down.

Inference for rule set models is challenging since the model space grows exponentially with the number of rules and the number of rules grows exponentially with the number of conditions. Classic simulated annealing that randomly proposes a state converges very slowly, making the problem difficult to solve in practice.

We describe an inference algorithm for efficiently searching for a MAP Model. First, we generate a set of candidate rules \mathcal{A} that contains promising rules with non-negligible support and restrict our search within only this meaningful set. Then we run a simulated annealing that applies various strategies to improve the search efficiency. We describe them in detail.

5.1 Candidate Rule Generation Our rule generation approach follows and modifies Virtual Twins (VT) methods [13]. First, two random forest [18] are built to predict $p(y_i = 1)$ in control and treatment groups, respectively.

$$\hat{p}_i(1) = p(y_i = 1|T_i = 1, \mathbf{x}_i),$$

 $\hat{p}_i(0) = p(y_i = 1|T_i = 0, \mathbf{x}_i).$

 $\hat{p}_i(1) - \hat{p}_i(0)$ can be regarded as an estimate of the treatment effect on instance i. In VT [13], it is then directly used as a classification or regression label to build a tree and get subgroups. However, this approach produces suboptimal solutions in our model since our goal is to find an optimal set of rules that collectively capture a subgroup. A single rule with high estimated treatment effect is not necessarily selected since it might work poorly with other selected rules. Nonetheless, this estimation is useful in narrowing down the search space, since it is unlikely that a collectively good set of rules contains a rule that is particularly bad. Therefore, we need only find rules with a high estimated treatment effect.

In order to find these rules, we define a binary variable to indicate if $\hat{p}_i(1) - \hat{p}_i(0)$ is non-negative:

$$Z_i = \mathbb{1}(\hat{p}_i(1) - \hat{p}_i(0) > 0).$$

 $Z_i=1$ represents instances with non-negative estimated treatment effects from which we mine frequent itemsets. There are many off-the-shelf techniques that can mine rules sufficiently fast. In this particular work, we used FP-Growth [19]. FP-Growth takes binary-coded data where each column represents whether the attributes satisfy a condition. A condition can refer to either a continuous attribute within a range (for example, age is between 10 to 20) or a categorical attribute within a specific category (for example, gender is male). Since the rules are mined only from the positive class, they cover observations with larger estimated treatment effect than the general population.

Validity check To control for confounding, a candidate rule need to cover treatment and control groups with similar distributions. Therefore, we apply a two-sample Student's t-test on the treatment and control groups covered by each rule. It checks if two distributions have the same mean. If it does, then we can assume the two distributions are approximately similar. We only keep rules with a p-value significantly small. In our work we used p < 0.05.

In practice, to further reduce computation, one can use the average treatment effect within each rule as a secondary criterion to further reduce the search space. In our work, we use the implementation from [20] to evaluate the average treatment effect to further screen rules.

These rules become the rule space \mathcal{A} from where we will find an optimal set A^* .

5.2 A Theoretically Grounded Heuristic Simulated annealing proposes a state by randomly choosing from neighbors. It takes tens of thousands of iterations to converge according to [21,22] for even a medium-sized dataset, which is not practical for a large real system. Wang et al. [17] proposed a more efficient algorithm that converges much faster but needs an evaluation on every neighbor at each iteration. It is computationally impractical for CRS model since evaluating $F(A, \mathbf{w})$ includes fitting a Bayesian logistic regression. We would like to develop a simpler heuristic to avoid computing $F(A, \mathbf{w})$ while not hurting the optimality of the output.

A key observation in our model is that the selected rule set will try to "cover" instances whose $y_i(1) - y_i(0) = 1$. (All other instances have a non-positive treatment effect.) While we are unable to observe both potential outcomes to find out exactly which examples to include, we are able to determine, by observing one of the potential outcomes and the treatment assignment, which instances to exclude. We divide the outcome space into four regions based on y_i and T_i : $I = \mathcal{E}_0 \cup \mathcal{E}_1 \cup \mathcal{U}$, and

(5.14)
$$\mathcal{E}_c = \{i | T_i = c, y_i = 1 - c\}, c \in \{0, 1\}$$

$$(5.15) \qquad \mathcal{U} = \{i | T_i = y_i\}.$$

We observe that when an instance i is in \mathcal{U} , it is unknown whether the treatment is effective without knowing the other potential outcome. It needs to be inferred by the model. However, if an example is in \mathcal{E}_c , the treatment effect is non-positive. \mathcal{E}_0 represents a group of instances that already have good outcomes $(y_i = 1)$ in the control group so treatment is not necessary. \mathcal{E}_1 represents a group that shows bad outcomes $(y_i = 0)$ under treatment so the treatment is not helpful. Therefore a good subgroup should contain less or none of the \mathcal{E}_c area. From this intuition we define an $Ideal\ Rule\ Set$.

DEFINITION 1. Given a data set $S = \{(\mathbf{x}_i, y_i, T_i)\}_{i=1}^n$, an Ideal Rule Set \bar{A} is defined as:

$$\bar{A}(\mathbf{x}_i) = \mathbb{1}(i \in \mathcal{U}).$$

An ideal rule set "covers" only \mathcal{U} . We call such a rule set ideal because it achieves maximum conditional likelihood with prior on \mathbf{w} . We show in Theorem 1 that $\Theta(S; \bar{A}, \mathbf{w}_{\bar{A}})$ is the upper bound on the Bayesian logistic regression on dataset S given any rule set A and its corresponding parameters \mathbf{w}_A , i.e.,

THEOREM 1. $\forall A.$ Let $\mathbf{w}_A = \max_{\mathbf{w}} \Theta(A, \mathbf{w})$ and notate the elements in \mathbf{w}_A as $\mathbf{w}_A = {\{\mathbf{v}_A, \gamma_A^{(0)}, \gamma_A^{(0)}, \gamma_A^{(2)}\}}$. If $\gamma_A^{(1)} \leq 0, \gamma_A^{(1)} + \gamma_A^{(2)} \geq 0$,

$$\Theta(A, \mathbf{w}_A) \leq \Theta(\bar{A}, \mathbf{w}_{\bar{A}}).$$

(See the supplementary material for the proof.) This theorem states that an Ideal Rule Set is an optimal solution to $F(A, \mathbf{w})$, if ignoring the prior probability $\operatorname{Prior}(A)$. Note that this Theorem only provides a mathematically achievable upper bound on $\Theta(\cdot)$. It does not guarantee that an Ideal Rule Set is a MAP model since there might not exist a rule set that only covers examples in $\mathcal U$ at all. However, Theorem 1 illuminates on criteria for evaluating a rule set, that a good rule set needs to cover much of $\mathcal U$ and little of $\mathcal E_0 \cup \mathcal E_1$. Following this intuition, We define $\operatorname{precision}$ of a rule set.

Definition 2. Given a data set S indexed by I, the precision of a rule set A is

(5.16)
$$Q(A) = \frac{|I_A \cap \mathcal{U}|}{|\mathcal{U}|}.$$

5.3 Exploration-with-Exploitation We use the metric (5.16) to evaluate and choose neighbors generated from adding, cutting or replacing a rule. To further reduce the number of candidate neighbors, we would like to first determine which action to take by learning from a current model. Define

(5.17)
$$\epsilon^{[t]} = \{i | i \in \mathcal{E}_0 \cup \mathcal{E}_1, A^{[t]}(\mathbf{x}_i) = 1\},\$$

(5.18)
$$u^{[t]} = \{i | i \in \mathcal{U}, A^{[t]}(\mathbf{x}_i) = 0\}.$$

At iteration t, an example k is drawn uniformly from $\epsilon^{[t]} \cup u^{[t]}$. Let $\mathcal{R}_1(\mathbf{x}_k)$ represent a set of rules that \mathbf{x}_k satisfies and $\mathcal{R}_0(\mathbf{x}_k)$ represent a set of rules that \mathbf{x}_k does not satisfy. If $\mathbf{x}_k \in \epsilon^{[t]}$, it means $A^{[t]}$ covers wrong data and we then find a neighboring rule set that covers less, by removing or replacing a rule from $A^{[t]} \cap \mathcal{R}_0(\mathbf{x}_k)$. If $\mathbf{x}_k \in u^{[t]}$, then as explained previously, it is not sure if \mathbf{x}_k should or should not be covered. Therefore, the new rule set is proposed by randomly choosing from adding a rule from $\mathcal{R}_1(\mathbf{x}_k)$, or removing or replacing a rule from $A^{[t]} \cap \mathcal{R}_0(\mathbf{x}_k)$.

After determining the best action, we choose a rule $z^{[t]}$ to perform the action on. We evaluate $Q(\cdot)$ on all neighbors produced by performing the selected action. Then a choice is made between exploration, choosing a random rule, and exploitation, choosing the best rule. We denote the probability of exploration as q. This randomness helps avoid local minima and helps the Markov Chain to converge to the global optima. We detail the three actions below.

- ADD: 1) With probability q, draw $z^{[t]}$ randomly from $\mathcal{R}_1(\mathbf{x}_k)$; with probability 1-q, $z^{[t]}=\arg\max_{a\in\mathcal{R}_1(\mathbf{x}_k)}Q(A^{[t]}\cup a)$. 2) Then $A^{[t+1]}\leftarrow A^{[t]}\cup z$.
- REMOVE: 1) With probability q, draw $z^{[t]}$ randomly from $A^{[t]} \cap \mathcal{R}_0(\mathbf{x}_k)$; with probability 1-q, $z^{[t]} = \underset{a \in A^{[t]} \cap \mathcal{R}_0(\mathbf{x}_k)}{\arg \max} Q(A^{[t]} \setminus a)$. 2). Then $A^{[t+1]} \leftarrow A^{[t]} \setminus z$.
- REPLACE: 1) REMOVE, 2) ADD.

The proposal strategy assesses the current model and evaluates all neighbors in order to make sure that the selected action improves the current model, and the selected rule makes maximizes the improvement (coordinate descent). This is significantly more practically efficient than proposing moves uniformly at random.

Constraint-Region Search We wish to further reduce search complexity by utilizing properties derived from the model. In our proposed framework, the Bayesian prior places a preference on particular sizes of models (we choose the parameters so that smaller models are preferred over large models), which means the MAP solution is more likely to appear in certain regions than others. If we locate the more promising regions defined by prior parameters, we can restrain the search within a much smaller space and will, therefore, find the MAP models more quickly. Here we wish to derive an upper bound on the size of a MAP model; furthermore, as we obtain better solutions with the search, we would like the bound to become tighter so that the "appropriate" region becomes more defined as we get close to the MAP model.

Let $v^{[t]}$ denote the maximum objective value that we have seen at or before time t,

$$v^{[t]} = \max_{\tau \le t} F(A^{[\tau]}, \mathbf{w}_{A^{[\tau]}}).$$

Let M_l^* represent the number of rules of length l in the MAP model A^* and $m_l^{[t]}$ represent the derived upper bound on M_l^* at time t. $m_l^{[t]}$ decreases monotonically with t and is updated via the following theorem.

Theorem 2. On data S, apply a Causal Rule Set model with parameters

$$H = \{\mathcal{A}, L, \{\alpha_l, \beta_l\}_{l=1}^L, \Sigma, \mu\},\$$

where $L, \{\alpha_l, \beta_l\}_{l=1,...L} \in \mathbb{N}^+$. Define $\{A^*, \mathbf{w}^*\} \in \arg\min_{A,\mathbf{w}} F(A,\mathbf{w}) \text{ and } \mathbf{w}_{A^*} = \{\mathbf{v}_{A^*}, \gamma_{A^*}^{(0)}, \gamma_{A^*}^{(1)}, \gamma_{A^*}^{(2)}, \}$. If $\alpha_l < \beta_l, \gamma_{A^*}^{(1)} \leq 0, \gamma_{A^*}^{(1)} + \gamma_{A^*}^{(2)} \geq 0$, we have:

$$|A^*| \le \sum_{l=1}^{L} m_l^{[t]},$$

where

$$m_l^{[t]} = \frac{\log \Theta(\bar{A}, \mathbf{w}_{\bar{A}}) + \log p(\emptyset) - v^{[t]}}{\log \left(\frac{|\mathcal{A}_l| + \beta_l - 1}{m_l^{[t-1]} + \alpha_l - 1}\right)}$$

for
$$t \in \mathbb{N}^+$$
 and $m_l^{[0]} = \frac{\log \Theta(\bar{A}, \mathbf{w}_{\bar{A}}) - \log \Theta(\emptyset, \mathbf{w}_{\emptyset})}{\log \left(\frac{|A_l| + \beta_l - 1}{|A_l| + \alpha_l - 1}\right)}$.

The smaller $\frac{\alpha_l}{\beta_l}$, the tighter the bound, which is consistent with the intuition of selecting rules of length l with smaller p_l . At time 0, we use an empty rule set as a benchmark, i.e. $v^{[0]} = \log \Theta(\emptyset, \mathbf{w}_{\emptyset}) + \log p(\emptyset), m_l^{[-1]} \leq |\mathcal{A}_l|$, yielding $m_l^{[0]}$ as above. If an empty set is a good approximation, i.e., $\Theta(\emptyset, \mathbf{w}_{\emptyset})$ is close to $\Theta(\bar{A}, \mathbf{w}_{\bar{A}})$, then $|A^*|$ is small. This bound agrees with intuition; if an empty set already achieves good performance, then adding more rules will likely hurt the outcome.

As the search continues, $v^{[t]}$ becomes larger, and the upper bound $m_l^{[t]}$ decreases accordingly, pointing to a more defined area of smaller sets of rules where our search algorithm should emphasize and explore more. Therefore in our algorithm, the probability of adding rules decay exponentially as $m_l^{[t]}$ decreases. The smaller this bound, the more often a CUT action should be selected instead of an ADD or REPLACE, in order to reduce the model size. This bound drags the sampling chain towards regions of smaller models and helps to find the MAP model much quicker in practice.

See the algorithm in Algorithm 1.

Algorithm 1 Search Algorithm for CRS

1: Initialize: $A^{[0]}, v^{[0]} \leftarrow \emptyset, F(\emptyset, \mathbf{w}_{\emptyset})$

2: **for** t = 0, ..., T **do**

```
\mathbf{x}_i \leftarrow \text{ an example randomly drawn from } \epsilon^{[t]} \cup u^{[t]} \ (\epsilon^{[t]})
         and u^{[t]} are from Formula (5.17) and (5.18)).
  4:
                           A^{[t+1]} = \begin{cases} \text{CUT}(A^{[t]}) \text{ with prob } \frac{1}{2} \\ \text{REPLACE}(A^{[t]}) \text{ with prob } \frac{1}{2} \end{cases}
  5:
                       A^{[t+1]} = \begin{cases} \text{CUT}(A^{[t]}) \text{ w/ prob } \frac{2}{3} - \frac{1}{3}e^{-\frac{\sum_{l=1}^{L} m_{l}^{[t]}}{C}} \\ \text{ADD}(A^{[t]}) \text{ w/ prob } \frac{1}{3}e^{-\frac{\sum_{l=1}^{L} m_{l}^{[t]}}{C}} \\ \text{REPLACE}(A^{[t]}) \text{ with prob } \frac{1}{3} \end{cases}
  6:
  8:
                  \mathbf{w}_{A[t+1]} = \arg\max_{A} \Theta(A^{[t+1]}, \mathbf{w})
  9:
                  v^{[t+1]} = \max\{v^{[t]}, F(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}})\}
10:
                  if F(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}}) \ge v^{[t]} then
11:
                           Update A^*, \mathbf{w}_{A^*}, v^{[t+1]}
12:
13:
         \begin{array}{ll} (A^{[t+1]},\mathbf{w}_{A^{[t+1]}}) &= (A_t,\mathbf{w}_{A^{[t]}}) \quad \text{with} \quad \text{probability} \\ \exp\left(\frac{F(A^{[t+1]},\mathbf{w}_{A^{[t+1]}}) - F(A^{[t]},\mathbf{w}_{A^{[t]}})}{T_0(t)}\right) \end{array}
15: end for
16: return A^*, \mathbf{w}_{A^*}
```

6 Experimental Evaluation

In this section, we discuss the detailed experimental evaluation of causal rule sets using synthetic and real-world datasets and compare CRS with various state-of-the-art baseline methods.

6.1 Recovery performance on synthetic data We would like to test how accurately and how quickly the subgroups are identified by CRS. We apply CRS on simulated datasets where true subgroups with enhanced treatment effect are predefined, and then compare true subgroups with subgroups recovered by CRS.

We first generate features $\{\mathbf{x}_i\}_{i=1}^n$ randomly drawn from normal distributions and randomly assigned binary treatment $\{T_i\}_{i=1}^n$. Next, $y_i(0)$ are generated from a logistic regression function with parameters \mathbf{v} randomly drawn from normal distributions. $y_i(0) = 1$ if $\sigma(\mathbf{v}x_i) \geq 0.5$ and $y_i(0) = 0$ otherwise. Then we mine rules from $\{\mathbf{x}_i\}_{i=1}^n$ and randomly choose five rules to form a true rule set A^* . Assuming the treatment is effective only on subgroup I_{A^*} and non-effective everywhere else, we set $y_i(1) = 1$ if $A^*(\mathbf{x}_i) = 1$, otherwise $y_i(1) = y_i(0)$. Then for each instance \mathbf{x}_i , the observed label is generated by

$$y_i = (1 - T_i)y_i(0) + T_iy_i(1).$$

Thus we created simulated dataset $\{(\mathbf{x}_i, y_i, T_i)\}_i^n$ with a varying number of instances n and number of features J. For each simulated dataset, 30% was held out for testing. On the remaining 70%, we first mined rules with a minimum support of 5% (we are only interested in subgroups with non-negligible sizes) and L=3. We then used average treatment effect as a secondary criterion to select the best m rules as search space A. (The average treatment effect was computed using propensity score matching. Here we used the implementation from [20]). We set the expected means μ tot 0 and variance Σ to an identity matrix. Since we favor small models, we set $\alpha_l = 1$ and $\beta_l = |A_l|$ for all l. Finally, we ran the search algorithm till 150 steps.

The search algorithm returned an output rule set for each simulated dataset, denoted as \hat{A} . Here, we evaluate the closeness of the recovered subgroup captured by \hat{A} to the true subgroup captured by A^* at each iteration. The closeness is measured in accuracy by comparing the labels of instances (1 means being in the subgroup and 0 otherwise) determined by A^* and those that are determined by \hat{A} .

We conducted three sets of experiments, varying n, J and m. For each set of parameters, we repeated the experiment 100 times and recorded the error rate on the hold-out data. We plot the average and standard deviation of the error rate in Figure 2, together with

their runtime in seconds.

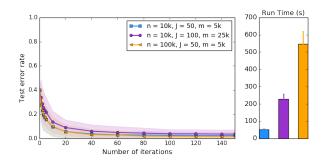


Figure 2: Convergence and runtime analysis of CRS.

Figure 2 shows that CRS recovers the true subgroup with high probability and the convergence happens within 100 iterations. Blue and yellow curves represent two sets of experiments on datasets with 50 features and different numbers of instances. We notice that these two curves almost entirely overlap. This is because the algorithm searches within the same size of rule space, 5000. The number of instances does not have much impact on the speed of convergence or the performance of the output model. On the other hand, if the search space increases, as it did for the third set of experiments where m = 25,000, the algorithm achieved slightly higher error rate since it becomes more difficult for the algorithm to search within a much larger space. While all three sets of experiments ran for 150 steps, their run time differed since the dimension of the matrix the algorithm works on was different, needing different processing time for each iteration.

6.2 Experiments on real-world data We then evaluate the performance of CRS on real-world datasets from diverse domains including juvenile study, consumer analysis, and court decisions.

Dataset The first data set is In-vehicle Recommender System collected from Amazon Mechanical Turk via a survey $[23]^1$. Turkers were asked whether they would accept a coupon in different driving scenarios characterized by passenger, weather, destination, time, current location, etc. For this experiment, our goal is to understand the treatment effect of factors, i) price range of average expenditure(<\$20 or \ge \$20) and ii) direction of the venue for using the coupon (on the same direction as the current direction or opposite), and we identify subgroups where the treatment shows an enhanced treatment effect on drivers' acceptance of a

Two made the data available at https://www.researchgate.net/publication/318645502_in-vehicle_coupon_recommendation

Table 1: Summary of datasets

	Treatment/Control	n	Descriptions	Y
In-vehicle recommender system	Direction: same/opposite	12,684	User's attributes: gender, age, marital status, etc; Contextual attributes: destination,	accept the coupon
	Price: <\$20/≥\$20		weather, time, etc; Coupon's attributes: time before it expires	
Juvenile	Exposure to violence: Y/N	4,023	gender, age, questions about witnessing violence, friends delinquency, family drug or alcohol history	delinquency
Sentencing	Prior conviction: Y/N	5,562	race, sex, plea, number of previous years in prison sex of the judge, race of the judge, etc	prison sentence

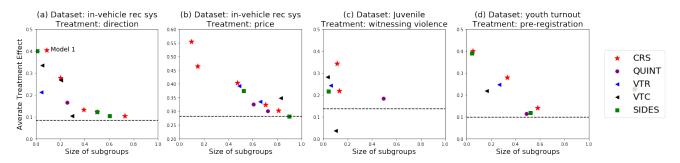


Figure 3: Average treatment effect and the size of the subgroups discovered by CRS and baseline method

coupon. The second data set is **Juvenile** dataset [24], which was to study the effect of juvenile exposure to violence on committing delinquency. The data was collected via a survey sent to juveniles which asks several questions about witnessing violence in real life (community, school, family, etc), friends' delinquency (drug use, alcohol use, etc) and family drug or alcohol problems. The third dataset is **Sentencing** [25], which was to examine the impact of defendants' prior criminal records on the current sentencing. See Table 1 for a summary of the datasets.

To compare with CRS, we chose the following stateof-the-arts as baselines: SIDES [12], VT [13] with a classification tree (VTC), VT with a regression tree (VTR), and QUINT [14]. Subgroups obtained by different methods are not directly comparable due to their differing sizes and treatment effects - it cannot be determined if a smaller subgroup with a higher treatment effect is better or worse than a larger subgroup with a smaller treatment effect. Thus, we wish to obtain a list of models for each method such that they cover a larger range of subgroup size and treatment effect for easier comparison. To do this, for CRS, we varied parameters α_l, β_l to favor smaller or larger set of rules.. For baseline methods, since they are tree-structured, we varied the maximum depth of a tree from 2 to 6 to generate a list of models.

We partition each data set into 60% training, 20% validation and 20% testing. We applied CRS and

baseline methods on the training set and chose the list of models on the frontier of the treatment effect-subgroup size curve evaluated on the validation set. We found the corresponding subgroups on the test set and plot their size and average treatment effect in Figure 3. The dashed line represents the baseline average treatment effect on the test data, computed using propensity score matching [26] methods to account for possible selection bias in the data. We used the implementation from [20] for the actual computation.

We present an example of a CRS model (Model 1 in Figure 3a), obtained from the in-vehicle recommender system dataset with treatment "direction".

weather = snowy AND destination \neq No immediate destination AND age ≤ 45

OR weather = snowy AND passenger \neq friends

This subset has a support of 8.9% and an average treatment of 0.40. It is interesting to notice that the two rules capture situations when people are reluctant to take a detour, due to weather condition (snowy) or/and when the driver is going to some immediate destination. In these cases, the relative location of a venue is of critical importance in creating successful adoption of the coupon than general contexts.

In our experiments, CRS achieved consistently competitive performance compared to baselines. This is unsurprising, given that it aims to globally optimize

its objective, whereas baseline methods use greedy approaches. While baseline methods cover a smaller range of subgroup sizes, CRS offers greater flexibility of tuning the model to generate a much larger range of subgroups of varying sizes and average treatment effects. This makes it possible for the model to suit the need of different problems in real applications.

7 Conclusions

We present a new model, Causal Rule Sets, for identifying a subgroup by a small set of short rules. To the best of our knowledge, this is the first work to use a set of rules to capture a subgroup with enhanced treatment effect. Previous recursive partitioning methods use one rule to capture a group and are not flexible in covering subgroups with irregular "shapes". CRS is to capture subgroups of various shapes and sizes. We use a Bayesian framework to learn rules from data. The inference method uses theoretically grounded heuristics and bounding strategies to improve search efficiency. evaluated the recovery performance of CRS using synthetic data generated from underlying true subgroups. and our algorithm converged quickly to the true subgroups. We demonstrated the effectiveness of our model on real-world datasets from diverse domains.

References

- [1] Jose Baselga, Edith A Perez, Tadeusz Pienkowski, and Richard Bell. Adjuvant trastuzumab: a milestone in the treatment of her-2-positive early breast cancer. *The* oncologist, 11(Supplement 1):4–12, 2006.
- [2] Peter M Rothwell. Subgroup analysis in randomised controlled trials: importance, indications, and interpretation. *The Lancet*, 365(9454):176–186, 2005.
- [3] Jacob Cohen, Patricia Cohen, Stephen G West, and Leona S Aiken. Applied multiple regression/correlation analysis for the behavioral sciences. Routledge, 2013.
- [4] James O Berger, Xiaojing Wang, and Lei Shen. A bayesian approach to subgroup identification. *Journal of biopharmaceutical statistics*, 24(1):110–129, 2014.
- [5] Kosuke Imai, Marc Ratkovic, et al. Estimating treatment effect heterogeneity in randomized program evaluation. The Annals of Applied Stats, 7(1), 2013.
- [6] Xiaogang Su, Chih-Ling Tsai, Hansheng Wang, David M Nickerson, and Bogong Li. Subgroup analysis via recursive partitioning. *Journal of Machine Learn-ing Research*, 10(Feb):141–158, 2009.
- [7] Fulton Wang and Cynthia Rudin. Causal falling rule lists. arXiv preprint arXiv:1510.05189, 2015.
- [8] Patrick M Schnell, Qi Tang, Walter W Offen, and Bradley P Carlin. A bayesian credible subgroups approach to identifying patient subgroups with positive treatment effects. *Biometrics*, 72(4):1026–1036, 2016.
- [9] Wei-Yin Loh, Xu He, and Michael Man. A regression tree approach to identifying subgroups with differential treatment effects. *Stats in medicine*, 34, 2015.

- [10] Susan F Assmann, Stuart J Pocock, Laura E Enos, and Linda E Kasten. Subgroup analysis and other (mis) uses of baseline data in clinical trials. The Lancet, 355(9209):1064–1069, 2000.
- [11] Stephen W Lagakos. The challenge of subgroup analyses-reporting without distorting. New England Journal of Medicine, 354(16):1667, 2006.
- [12] Ilya Lipkovich, Alex Dmitrienko, Jonathan Denne, and Gregory Enas. Subgroup identification based on differential effect searchâĂŤa recursive partitioning method for establishing response to treatment in patient subpopulations. Statistics in medicine, 30(21):2601–2621, 2011.
- [13] Jared C Foster, Jeremy MG Taylor, and Stephen J Ruberg. Subgroup identification from randomized clinical trial data. Statistics in medicine, 30(24):2867– 2880, 2011.
- [14] Elise Dusseldorp and Iven Van Mechelen. Qualitative interaction trees: a tool to identify qualitative treatment–subgroup interactions. Statistics in medicine, 33(2):219–237, 2014.
- [15] Joshua D Angrist, Guido W Imbens, and Donald B Rubin. Identification of causal effects using instrumental variables. *Journal of the American statistical Associa*tion, 91(434):444–455, 1996.
- [16] Donald B Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974.
- [17] Tong Wang, Cynthia Rudin, Velez-Doshi Finale, YImin Liu, Erica Klampfl, and Perry MacNeille. Bayesian rule sets for interpretable classification. In *ICDM*, 2016.
- [18] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [19] Christian Borgelt. An implementation of the fp-growth algorithm. In *Proc of the 1st interl workshop on open source data mining: frequent pattern mining implementations*, pages 1–5. ACM, 2005.
- [20] Jasjeet S Sekhon. Multivariate and propensity score matching software with automated balance optimization: the matching package for r. 2011.
- [21] Benjamin Letham, Cynthia Rudin, Tyler H. Mc-Cormick, and David Madigan. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. Annals of Applied Statistics, 2015. accepted with minor revision.
- [22] Fulton Wang and Cynthia Rudin. Falling rule lists. In AISTATS, 2015.
- [23] Tong Wang, Cynthia Rudin, F Doshi, Yimin Liu, Erica Klampfl, and Perry MacNeille. A bayesian framework for learning rule sets for interpretable classification. *Journal of Machine Learning Research*, 18(70):1–37, 2017.
- [24] Joy D Osofsky. The effect of exposure to violence on young children. American Psychologist, 50(9):782, 1995.
- [25] S Welch and C Spohn. Effects of prior record in sentencing research in a large northeastern city, 1968-79. 1988.
- [26] Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, pages 41–55, 1983.

8 Appendix

Proof. (of Theorem 1) We notate elements in $\mathbf{w}_{\bar{A}}$ as $\mathbf{w}_{\bar{A}} = \{\mathbf{v}_{\bar{A}}, \gamma_{\bar{A}}^{(0)}, \gamma_{\bar{A}}^{(1)}, \gamma_{\bar{A}}^{(2)}\}$. Since $\bar{A}(\mathbf{x}_i) = 1$ when $i \in I \setminus \mathcal{E}$, we rewrite $\bar{A}(\mathbf{x}_i)$ as

(8.19)
$$\bar{A}(\mathbf{x}_i) = T_i y_i + (1 - T_i)(1 - y_i)$$

Expanding $\Theta(\bar{A}, \mathbf{w}_{\bar{A}})$ using formula (4.10) and plugging in (8.19) yields

$$\log \Theta(\bar{A}, \mathbf{w}_{\bar{A}}) = \log p(\mathbf{w}_{\bar{A}}) + \sum_{u_i = 1} \log \sigma \left(\mathbf{v}_{\bar{A}} \mathbf{x}_i + \gamma_{\bar{A}}^{(0)} T_i + \gamma_{\bar{A}}^{(1)} T_i + \gamma_{\bar{A}}^{(2)} T_i \right) +$$

(8.20)
$$\sum_{u_i=0} \log \left[1 - \sigma \left(\mathbf{v}_{\bar{A}} \mathbf{x}_i + \gamma_{\bar{A}}^{(0)} T_i + \gamma_{\bar{A}}^{(1)} (1 - T_i) \right) \right]$$

We then upper bound the conditional likelihood of data S and the prior of parameters given any rule set A and \mathbf{w}_A .

$$\log \Theta(A, \mathbf{w}_A) = \sum_{i=1}^n \log P(y_i | \mathbf{x}_i, T_i; A, \mathbf{v}_A) + \log p(\mathbf{w}_A) = \log p(\mathbf{w}_A)$$

$$\sum_{u_i=1} \log \sigma \left(\mathbf{v}_A \mathbf{x}_i + \gamma_A^{(0)} T_i + \gamma_A^{(1)} A(\mathbf{x}_i) + \gamma_A^{(2)} T_i A(\mathbf{x}_i) \right) \tag{U_1}$$

(8.21)
$$+ \sum_{y_i=0} \log \left[1 - \sigma \left(\mathbf{v}_A \mathbf{x}_i + \gamma_A^{(0)} T_i + \gamma_A^{(1)} A(\mathbf{x}_i) + \gamma_A^{(2)} T_i A(\mathbf{x}_i) \right) \right], \tag{U_2}$$

where

(8.22)
$$\gamma_A^{(1)} A(\mathbf{x}_i) + \gamma_A^{(2)} T_i A(\mathbf{x}_i) \le (\gamma_A^{(1)} + \gamma_A^{(2)}) T_i A(\mathbf{x}_i)$$

(8.22) follows since $\gamma^{(1)} \leq 0$ and $T_i \in \{0,1\}$, so $\gamma_A^{(1)} A(\mathbf{x}_i) \leq \gamma_A^{(1)} T_i A(\mathbf{x}_i)$. (8.23) follows because $\gamma^{(1)} + \gamma^{(2)} \geq 0$ and $A(\mathbf{x}_i) \in \{0,1\}$. Since $\sigma(x)$ increases monotonically with x, we get

(8.24)
$$U_1 \le \sum_{y_i=1} \log \sigma \left(\mathbf{v}_A \mathbf{x}_i + \gamma_A^{(0)} T_i + \gamma_A^{(1)} T_i + \gamma_A^{(2)} T_i \right)$$

Meanwhile, since $\gamma_A^{(2)} \ge -\gamma_A^{(1)}$,

$$\gamma_A^{(1)} A(\mathbf{x}_i) + \gamma_A^{(2)} T_i A(\mathbf{x}_i) \ge \gamma_A^{(1)} A(\mathbf{x}_i) - \gamma_A^{(1)} T_i A(\mathbf{x}_i)$$

$$= \gamma^{(1)} (1 - T_i) A(\mathbf{x}_i)$$

$$\ge \gamma^{(1)} (1 - T_i)$$
(8.25)

(8.25) follows because $\gamma^{(1)} \leq 0$ and $A(\mathbf{x}) \in \{0,1\}$. Thus

(8.26)
$$U_2 \le \sum_{u_i=0} \log \left[1 - \sigma \left(\mathbf{v}_A \mathbf{x}_i + \gamma_A^{(0)} T_i + \gamma_A^{(1)} (1 - T_i) \right) \right]$$

Plugging (8.24) and (8.26) back into inequality (8.21) yields

$$\log \Theta(A, \mathbf{w}_{A}) \leq \log p(\mathbf{w}_{A}) + \sum_{y_{i}=1} \log \sigma \left(\mathbf{v}_{A} \mathbf{x}_{i} + \gamma_{A}^{(0)} T_{i} + \gamma_{A}^{(1)} T_{i} + \gamma_{A}^{(2)} T_{i} \right) +$$

$$\sum_{y_{i}=0} \log \left[1 - \sigma \left(\mathbf{v}_{A} \mathbf{x}_{i} + \gamma_{A}^{(0)} T_{i} + \gamma_{A}^{(1)} (1 - T_{i}) \right) \right]$$

$$= \log \Theta(\bar{A}, \mathbf{w}_{A}) \leq \log \Theta(\bar{A}, \mathbf{w}_{\bar{A}})$$

$$(8.27)$$

(8.27) follows since $\mathbf{w}_{\bar{A}} = \max_{\mathbf{w}} \Theta(\bar{A}, \mathbf{w})$.

Proof. (Of Theorem 2) Let \emptyset denote an empty set where there are no rules and \mathbf{w}_{\emptyset} is the optimal parameter corresponding to \emptyset . Since $\{A^*, \mathbf{w}_{A^*}\} \in \arg\min_{A, \mathbf{w}} F(A, \mathbf{w})$, then $F(A^*, \mathbf{w}_{A^*}) \geq v^{[t]}$, i.e.

(8.28)
$$\log \Theta(S; A^*, \mathbf{w}_{A^*}) + \log p(A^*) \ge v^{[t]}.$$

Now we find conditions where the above inequality always holds. We do it in the proceeding two steps. According to Theorem 1, we have

$$\Theta(S; A^*, \mathbf{w}_{A^*}) \le \Theta(\bar{A}, \mathbf{w}_{\bar{A}})$$

The prior probability of selecting A^* is

$$(8.30) p(A^*) = \prod_{l}^{L} \mathcal{L}_l h(M_l),$$

We write out $h(M_l)$ as the following, multiplying by 1 in disguise:

$$h(M_l) = \Gamma(\alpha_l)\alpha_l \dots (\alpha_l + M_l - 1) \frac{\Gamma(|\mathcal{A}_l| + \beta_l - M_l)(|\mathcal{A}_l| + \beta_l - M_l) \dots (|\mathcal{A}_l| + \beta_l - 1)}{(|\mathcal{A}_l| + \beta_l - M_l) \dots (|\mathcal{A}_l| + \beta_l - 1)}$$

$$= \frac{\Gamma(\alpha_l)\Gamma(|\mathcal{A}_l| + \beta_l)\alpha_l \dots (\alpha_l + M_l - 1)}{(|\mathcal{A}_l| + \beta_l - M_l) \dots (|\mathcal{A}_l| + \beta_l - 1)}$$

$$\leq \Gamma(\alpha_l)\Gamma(|\mathcal{A}_l| + \beta_l) \left(\frac{\alpha_l + m_l^{[t]} - 1}{|\mathcal{A}_l| + \beta_l - 1}\right)^{M_l}$$
(8.31)

We also observe

$$(8.32) h(M_l) \le h(0)$$

for all M_l . To obtain this, we take the second derivative of $h(M_l)$ with respect to M_l , as follows:

$$h''(M_l) = h(M_l) \left(\left(\sum_{k=1}^{\infty} \frac{1}{k + |\mathcal{A}_l| + \beta_l - M_l - 1} - \frac{1}{k + M_l + \alpha_l - 1} \right)^2 + \left(\sum_{k=1}^{\infty} \frac{1}{k + |\mathcal{A}_l| + \beta_l - M_l - 1} \right)^2 + \left(\sum_{k=1}^{\infty} \frac{1}{k + M_l + \alpha_l - 1} \right)^2 \right)$$
> 0.

Therefore $h(M_l)$ is strictly convex, and $h(M_l) \le \max\{h(0), g(|\mathcal{A}_l|)\} = h(0)$. Combining (8.30) with (8.31) and (8.32) we have

$$p(A^*) = \mathcal{L}_{l'}h(M_{l'}) \prod_{l=1,\dots L, l \neq l'} \mathcal{L}_{l}h(M_{l})$$

$$\leq \mathcal{L}_{l'}\Gamma(\alpha_{l})\Gamma(|\mathcal{A}_{l}| + \beta_{l}) \left(\frac{|\mathcal{A}_{l}| + \alpha_{l} - 1}{|\mathcal{A}_{l}| + \beta_{l} - 1}\right)^{M_{l}} \prod_{l=1,\dots L, l \neq l'} \mathcal{L}_{l}h(0)$$

$$= \left(\frac{M_{l} + \alpha_{l'} - 1}{|\mathcal{A}_{l}| + \beta_{l'} - 1}\right)^{M_{l'}} \cdot \prod_{l} \mathcal{L}_{l}h(0)$$

$$= \left(\frac{M_{l} + \alpha_{l'} - 1}{|\mathcal{A}_{l}| + \beta_{l'} - 1}\right)^{M_{l'}} \cdot p(\emptyset),$$

$$(8.33)$$

which means

(8.34)
$$\log p(A^*) \le \log p(\emptyset) + M_{l'} \log \left(\frac{M_l + \alpha_{l'} - 1}{|\mathcal{A}_l| + \beta_{l'} - 1} \right)$$

Now we apply inequality (8.28) and substituting in (8.29) and (8.33), we find

(8.35)
$$\Theta(\bar{A}, \mathbf{w}_{\bar{A}}) + \log p(\emptyset) + M_{l'} \log \left(\frac{M_l + \alpha_{l'} - 1}{|\mathcal{A}_l| + \beta_{l'} - 1} \right) \ge v^{[t]}.$$

Since $\alpha_{l'} < \beta_{l'}$, so $\log \left(\frac{|M_l| + \alpha_{l'} - 1}{|A_l| + \beta_{l'} - 1} \right) < 0$, therefore

(8.36)
$$m_l^{[t]} = \frac{\log \Theta(\bar{A}, \mathbf{w}_{\bar{A}}) + \log p(\emptyset) - v^{[t]}}{\log \left(\frac{|\mathcal{A}_l| + \beta_l - 1}{m_l^{[t-1]} + \alpha_l - 1}\right)}$$

which holds for each $l \in \{1, 2..., L\}$. Thus, the total number of rules in A^* is bounded by

$$(8.37) |A^*| \le \sum_{l=1}^{L} m_l^{[t]}.$$