Open Access Library Journal
2023, Volume 10, e10615
ISSN Online: 2333-9721
ISSN Print: 2333-9705

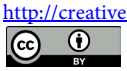
Heavy-Head Sampling Strategy of Graph Convolutional Neural Networks for *q*-Consistent Summary-Explanations with Application to Credit Evaluation Systems

Xinrui Dou

School of Communication and Electronic Engineering, Jishou University, Jishou, China
Email: douxinrui@stu.jsu.edu.cn

How to cite this paper: Dou, X.R. (2023) Heavy-Head Sampling Strategy of Graph Convolutional Neural Networks for *q*-Consistent Summary-Explanations with Application to Credit Evaluation Systems. *Open Access Library Journal*, **10**: e10615.
https://doi.org/10.4236/oalib.1110615

Received: August 14, 2023
Accepted: September 12, 2023
Published: September 15, 2023



Abstract

Machine learning systems have found extensive applications as auxiliary tools in domains that necessitate critical decision-making, such as healthcare and criminal justice. The interpretability of these systems' decisions is of paramount importance to instill trust among users. Recently, there have been developments in globally-consistent rule-based summary-explanation and its max-support (MSGC) problem, enabling the provision of explanations for specific decisions along with pertinent dataset statistics. Nonetheless, globally-consistent summary-explanations with limited complexity tend to have small supports, if any. In this study, we propose a more lenient variant of the summary-explanation, namely the *q*-consistent summary-explanation, which strives to achieve greater support at the expense of slightly reduced consistency. However, the challenge lies in the fact that the max-support problem of the *q*-consistent summary-explanation (MSqC) is significantly more intricate than the original MSGC problem, leading to extended solution times using standard branch-and-bound (B & B) solvers. We improve the B & B solving process by replacing time-consuming heuristics with machine learning (ML) models and apply a heavy-head sampling strategy for imitation learning of MSqC problems by exploiting the heavy-head maximum depth distribution of B & B solution trees. Experimental results show that using the heavy-head sampling strategies, the final evaluation results of trained strategies on MSqC problems are significantly improved compared to previous studies using uniform sampling strategies.

Subject Areas

Credit Evaluation Systems, Artificial Intelligence

Keywords

Summary-Explanation, q-Consistent, Branch-and-Bound, Heavy-Head Sampling Strategy

1. Introduction

Amidst the progressions within the realm of machine learning and the proliferation of applications within artificial intelligence, the imperative of transparency surrounding machine learning models has grown markedly pronounced across myriad contexts. Sectors necessitating discerning determinations, notably healthcare and the criminal justice system have commenced integrating machine learning systems in a supplementary capacity [1]. Nonetheless, models boasting elevated accuracies frequently assume the form of enigmatic black boxes, characterized by opacity and the veiling of their decisional rationales. This phenomenon is largely inadvertent and inexorable, as the construction of these models (or algorithms) does not stem from direct human coding but rather emanates from intricate hypothesis sets—neural networks or support vector machines, for instance, possessing a profusion of parameters, forged through machine learning facilitated by copious volumes of training data.

To cultivate heightened transparency, the traditional modus operandi involves the deployment of global explicatory techniques, harnessed to elicit proximate interpretable models from the recesses of black-box machine learning counterparts, thus emulating their decisional logic. As an illustration, neural networks can yield extracted classification rules [2], while tree ensembles can yield derived decision trees [3]. Notwithstanding, each of these conventional global explication methodologies is tailored to specific model categories, encompassing neural networks and tree ensembles, thereby eschewing model-agnosticism. Moreover, given the intersection of intricate models and expansive datasets, the resultant interpretable models extracted may not effectively approximate the original intricacies.

The method most closely akin to this study is the globally-consistent rule-based summative explication posited by Rudin *et al.* [4], wherein IP optimization dilemmas are tackled to ascertain the rule-based summative explication of a given target observation with minimal intricacy (the MCGC conundrum) or augmented substantiation (the MSGC dilemma). The substantiation of a rule-based summative explication is contingent upon the frequency of instances within the dataset that align with the rule's IF-condition. Illustratively, a globally-consistent summative explication, provided by the MaxSupport algorithm for an observation from the FICO dataset [5], is elucidated as follows, with a substantiation of 594: Among the 594 individuals for whom $ExternalRiskEstimate \leq 63$ and $AverageMinFile \leq 48$, the entirety exhibited default prognoses. Manifestly, explications endowed with substantial substantiations serve to instill confidence in

users' vis-à-vis the explicatory framework.

Nevertheless, owing to the constraint of global consistency, the application of the MSGC problem to voluminous datasets frequently engenders rules of elevated intricacy or negligible substantiation, if not unviable altogether. In myriad pragmatic contexts, explications characterized by substantial substantiations wield considerable influence over users' acquiescence, while trifling incongruity might prove tolerable. Piqued by this premise, the present study addresses the quandary of maximizing substantiation with q-consistency (MSqC), where $q \in (0,1]$, a paradigm that can markedly amplify the substantiation of the explicatory rule. A concrete instance of a 0.85-consistent summative explication, boasting a substantiation of 2000, is portrayed as follows: "Surpassing 85% of the 2000 individuals, whose $ExternalRiskEstimate \le 63$ and $AverageMinFile \le 48$, were prognosticated to default."

While the extension may appear uncomplicated, it transpires that the task of identifying a q-consistent summative elucidation is substantially more intricate than seeking its globally-consistent counterpart. The intricacy arises from the necessity to optimize q-consistent substantiation, which can encompass an incongruence fraction of up to $1-q$ among matched observations. This mandates the inclusion of all observations yielding divergent outcomes from the target observation within the MSqC formulation, diverging from the MSGC formulation. Furthermore, conventional integer programming (IP) solvers employing the branch-and-bound (B & B) precise approach exhibit limited efficiency in the context of MSGC. For instance, as reported in [4], a 60-second time constraint was imposed to achieve solution termination, and in our replication, complete resolution of MSGC employing the SCIP solver [6] necessitated an average of 101 seconds for a dataset size $|\mathcal{N}| = 1K$ and 1852 seconds for $|\mathcal{N}| = 10K$. The performance degradation is markedly exacerbated when tackling the more intricate MSqC, as substantiated in the experimental assessment.

In recent years, in order to automate the tuning of B & B algorithms for different types of problems, machine learning (ML) methods have been developed to replace the hand-crafted expert heuristics, also known as branch-and-bound methods based on machine learning (ML-B & B), the ML-B & B model is trained using the imitation learning model expert strategy strong branching rules

The main contribution of this paper is to use the heavy-head feature of the MSqC problem and apply the heavy-head sampling strategy, which improves the accuracy and efficiency of solving the MSqC problem and can better apply the MSqC problem to the field of credit evaluation systems.

2. Background

The initial body of explainability literature primarily delves into model-specific global elucidation techniques, aimed at delineating the overarching behavior of particular models. As an illustration, decision trees [7], and classification rules [2] [8] have been enlisted to expound the global aspects of neural networks. Diverg-

ing from these conventional methods, there has been a surge in interest toward model-agnostic local explanation strategies in recent years, focusing on elucidating the rationales underlying specific observations or predictions. This trajectory's inception is frequently attributed to the seminal contribution by Ribeiro *et al.* [9], wherein the LIME algorithm was introduced. The realization surfaced that model-level (*i.e.*, global) explanation is not universally viable due to the potential arbitrary complexity of black-box models; nonetheless, within a data instance's vicinity, the decision boundary should invariably exhibit sufficient simplicity to be captured by an interpretable model. This premise is veracious for nearly all pragmatic models, rendering local explanations broadly applicable.

Belonging to the class of model-available-agnostic methodologies, the LIME algorithm operates under the assumption that the black-box model is accessible for localized data generation pertaining to the target observation, albeit its specifics remain unknown. Specifically, for any arbitrary classifier, the LIME algorithm [9] commences by generating perturbed data proximate to the target observation, and then it fits an interpretable model (a sparse linear model) onto the created local dataset to facilitate model-agnostic local elucidation. Subsequent studies have engendered extensions to this approach. For instance, in a subsequent endeavor [10], Ribeiro *et al.* proposed replacing the sparse linear model in LIME with if-then rules, designated as Anchors, to yield explanations that are both more intuitive and accurate. In another work [11], Guidotti *et al.* harnessed a genetic algorithm to generate the local dataset, consequently deriving decision rules and counterfactual rules from the trained local interpretable predictor. Addressing the data shift and instability stemming from random perturbations in LIME, Zafar *et al.* introduced Deterministic Local Interpretable Model-Agnostic Explanations (DLIME) [12]. Additionally, Huang *et al.* devised a comprehensive interpretative framework for general graph neural network models, termed Graph-LIME [13], enabling localized interpretation of model outputs through nonlinear feature selection strategies such as the Hilbert-Schmidt Independence Criterion (HSIC) Lasso.

Instances exist wherein the black-box model remains elusive for the generation of synthetic data, leaving historical or furnished data as the sole fount of knowledge. A prominent illustration of this circumstance is encapsulated in the FICO explainable machine learning challenge [5], wherein a dataset engendered by FICO's proprietary black-box model is at disposal, yet the model itself remains beyond the reach of researchers. In situations governed solely by data availability, a prevailing strategy for elucidation involves training an alternate predictive model founded upon the proffered dataset, serving as a surrogate to the original model. Two avenues unfold before us: the surrogate model can either assume the form of an inherently interpretable model, or it can prioritize precision by adopting a black-box model for emulation. To synopsize, both avenues mandate the cultivation of a secondary model to mirror the original model, with the requirement of minimizing approximation error for trustworthiness of explanations. Moreover, irrespective of the chosen trajectory, neither can furnish

explanations that encompass data-relative statistics, such as support information as manifested in [4].

Charting an alternate course, Rudin *et al.* [4] introduced the globally-consistent rule-based summative explication technique. This approach sidesteps the necessity of generating local data via an accessible model. Instead, it engenders a rule-based summative explication through the resolution of an integer programming (IP) optimization dilemma. This endeavor hinges solely upon the dataset and can be steered towards either minimizing the intricacy of the rule or maximizing its substantiation. The substantiation of a rule-based summative explication is gauged by the number of instances within the dataset adhering to the rule's IF-condition. Exemplifying this approach, an illustrative summary-explanation declaration concerning an observation extracted from the FICO dataset resonates as follows: "For the entirety of 594 individuals exhibiting an $ExternalRiskEstimate \leq 63$ and $AverageMinFile \leq 48$, unanimous prognostications of default transpired." The substantiation for this summary-explanation rests at 594, an informative metric eluding model-available-agnostic methodologies reliant on data generation.

An advantageous attribute of Rudin *et al.*'s technique [4] resides in the summary-explanations it produces, being endowed with a certifiable global consistency. In other words, for all instances adhering to the IF-condition within the dataset, the THEN-statement holds true. Nonetheless, procuring globally-consistent rules can present challenges. When confronted with sizable practical datasets, the method outlined in [4] frequently yields rules marked by heightened intricacy or limited substantiation, or even infeasibility. Indeed, for many pragmatic scenarios, explanations possessing substantial substantiation aid in cultivating users' confidence in the explicatory system, while minor incongruities might prove tolerable, contingent upon certain thresholds. Anchored in this notion, this study embarks on an extension of the MaxSupport framework outlined in [4], wherein a slight leeway for inconsistencies within the explanation is introduced. This augmentation effectively augments the substantiation of the explicatory rule. The improved problem is called the MSqC problem.

The MSqC challenge epitomizes a form of MILP dilemma. Existing open-source solvers, exemplified by SCIP, are encumbered by substantial computational burdens when addressing such quandaries. In tandem with the progression of machine learning, the nexus of MILP solution strategies and machine learning (ML) approaches has manifested. A paradigmatic instance is the machine learning-based branch-and-bound (ML-B & B) technique, wherein supervised learning expedites the computation of selection priority indices during MILP resolution, hastening variable branch and node selection decisions within the branch-and-bound framework. Alvarez *et al.* introduced machine learning methodologies to swiftly compute the variable branch priority strong branching (SB) value [14]; subsequently, an imitation learning (IL)-grounded B & B method surfaced [15]. This IL-oriented approach directly learns the relative priority sequence of

diverse decisions, obviating the intricate index computation process and further augmenting B & B efficacy. Capitalizing on Graph Convolutional Neural Networks (GCNN), a model is devised to assimilate B & B selection tactics, leveraging bipartite graph representations of ongoing B & B states, and harnessing GCNN to distill insightful facets. The GCNN architecture exploits MILP's bipartite graph formulation and shared parameterization to model dilemmas across varying dimensions. Trained via IL, this model approximates the proficiency of resource-intensive expert-heuristic SB. Gasse *et al.*'s method [16] has proven notably efficacious across diverse MILP benchmark scenarios, prompting subsequent explorations and extensions [17] [18]. Building on this foundation, the endeavor to enhance the ML-B & B approach for the efficacious resolution of the MSqC quandary holds both consequential practical and scientific implications.

3. Problem Formulation

3.1. Global Consistency Summary Explanation

First convert the data set into a $|\mathcal{P}|$ -dimensional binary sample data set $\{(x_i, y_i), i \in \mathcal{N}\}$, where $x_i \in \{0,1\}^{|P|}$, $\mathcal{N}$ is the sample index set, and $\mathcal{N}$ is the binary feature function index set. In general, such a binary dataset can be obtained from an ordered feature set $\{\tilde{x}_{e,1} \geq 0, \tilde{x}_{e,1} \geq 50, \tilde{x}_{e,2} \geq 0\}$ Binarize it to $x_e = \{\delta_{e,1}, \delta_{e,2}, \delta_{e,3}\} = \{1,0,1\}$. The following terms "characteristic" and "characteristic function" are used interchangeably.

Let b represent a joint clause formed by the logical AND ($\wedge$) operation of multiple conditions, with each condition corresponding to a binary feature denoted by F_p. In other words, for a subset of features $\mathcal{P}' \subseteq \mathcal{P}$, the joint clause is defined as $b(\cdot) = \wedge_{p \in \mathcal{P}'} F_p(\cdot)$. Taking the given example, b could be either $\tilde{x}_{e,1} \geq 0$ or $(\tilde{x}_{e,1} \geq 50) \wedge (\tilde{x}_{e,2} \geq 0)$.

An overview explanation is represented by a rule $b(\cdot) \to y$, which defines a binary classifier as follows:

$$h^{b(\cdot)\to y}(x) = \begin{cases} y & \text{if } b(x) = \wedge_{p \in \mathcal{P}'} F_p(x) = 1 \\ 1-y & \text{else} \end{cases}$$

In the context of a sample value (x_e, y_e), a globally consistent overview explanation is an overview explanation $b(\cdot) \to y_e$ that satisfies the following properties:

1) Relevancy, *i.e.*, $b(x_e) = 1$;

2) Consistency, which states that for all sample values $i \in \mathcal{N}$, if $b(x_i) = 1$, then $y_i = y_e$.

A more lucid mode of expression posits that a globally consistent overview elucidating the mapping $b(\cdot) \to y_e$ may be cast in the following form: For all sample values (e.g., individuals or customers) for which $b(\cdot)$ holds true, the corresponding outcome (e.g., predicted risk or decision) is y_e, identical to the value of sample *e*. Such an elucidation establishes a correlation between the current sample value (x_e, y_e) and pre-existing records in the dataset, rendering it

more persuasive to users in domains such as credit evaluation. It is noteworthy that general global (model-level) rule-based explanations typically adopt lists of rules represented in either disjunctive normal form (DNF) or conjunction normal form (CNF). Suppose a DNF rule list $b^{(1)}(\cdot)\vee b^{(2)}(\cdot)\vee\cdots$ serves as the global interpretation of the black box; in that case, it can be inferred that for a given specific sample value (x_e, y_e), a globally consistent summary interpretation can be viewed as a clause $b(\cdot)$ in the DNF global rule list, activated on x_e, meaning $b(x_e)=1$.

The quality of the clause $b(\cdot)$ is assessed by two key metrics:

1) Complexity $|b|$, defined as the number of conditions in b;

2) Support $|\mathcal{S}_{\mathcal{N}}(b)|$, that is, the size of the support set $\mathcal{S}_{\mathcal{N}}(b)$, which is defined as the sample value set (or index set) that satisfies the clause b, that is, $S_N(b)=\{i\in\mathcal{N}: b(x_i)=1\}$.

As a matter of convention, the term "support" may pertain to either the set $\mathcal{S}_{\mathcal{N}}(b)$ or its cardinality $|\mathcal{S}_{\mathcal{N}}(b)|$. Moreover, for convenience, when employing set notation, one may refer to either the indexed set or the original set, provided that such usage does not give rise to confusion.

3.2. Minimize Complexity and Maximize Support

In [4], the paper introduces two significant problems related to the Global Consistency of the SE $b(\cdot)\to y_e$, namely the Global Consistency Minimizing Complexity (MCGC) and Global Consistency Maximizing Support (MSGC) problems. The primary objective of the MCGC problem is to identify the SE b with the lowest possible complexity. This objective can be precisely formulated as an Integer Programming (IP) model, which can be stated as follows:

$$\min_{b}\sum_{p\in P^e} b_p \tag{1}$$

$$\text{s.t.}\sum_{p\in P^e} b_p\left(1-\delta_{i,p}\right)\geq 1,\quad \forall i\in\mathcal{N}\setminus\mathcal{N}^e \tag{2}$$

$$b_p\in\{0,1\},\quad \forall p\in\mathcal{P}^e \tag{3}$$

where the binary decision variable $b_p=1$ indicates that the feature p appears in the resulting clause $b(\cdot)=\wedge_{p\in\mathcal{P}'}F_p(\cdot)$, *i.e.* $p\in\mathcal{P}'$; and $b_p=0$ otherwise. As a result, b_p is also referred to as a feature variable. Moreover, the binary variable $\delta_{i,p}\in\{0,1\}$ serves to indicate whether the observation value i satisfies the binary feature p, expressed as $\delta_{i,p}=F_p(x_i)$. The set $\mathcal{P}^e$, known as the activation feature set of the observation value e, consists of the feature subset that the observation value e fulfills, formally represented as $\mathcal{P}^e=\{p\in\mathcal{P}:\delta_{e,p}=1\}$. Furthermore, $\mathcal{N}^e$ designates the collection of consistent observations, *i.e.*, $\mathcal{N}^e=\{i\in\mathcal{N}: y_i=y_e\}$, while $\mathcal{N}\setminus\mathcal{N}^e$ denotes the set of inconsistent observations where $y_i\neq y_e$ for all $i\in\mathcal{N}\setminus\mathcal{N}^e$. Correlation is ensured by restricting the selection of features to $\mathcal{P}^e$, while consistency is guaranteed by satisfying condition (2), ensuring that any observation for $y_i\neq y_e$ will have $b(x_i)=0$. Although the MC model is expeditious to solve due to its simplicity, it does not

guarantee a large support.

The MSGC problem can be regarded as an extension of the MCGC problem, with its objective being to discover the SE b that maximizes the support $\left|s_N(b)\right|$, while still adhering to the complexity constraint $|b| < M_c$. In this context, the paper adopts a reasonable complexity threshold of $M_c = 4$ for SE in credit evaluation. The formulation of the MSGC problem is as follows:

$$\max_{b,r} \sum_{i \in \mathcal{N}^e} r_i \tag{4}$$

$$\text{s.t.} \sum_{p \in \mathcal{P}^e} b_p \left(1 - \delta_{i,p}\right) \geq 1, \quad \forall i \in \mathcal{N} \setminus \mathcal{N}^e \tag{5}$$

$$\sum_{p \in \mathcal{P}^e} b_p \left(1 - \delta_{i,p}\right) \leq M\left(1 - r_i\right), \quad \forall i \in \mathcal{N}^e \tag{6}$$

$$\sum_{p \in \mathcal{P}^e} b_p \leq M_c, \tag{7}$$

$$b_p, r_i \in \{0,1\}, \quad \forall p \in \mathcal{P}^e, \forall i \in \mathcal{N}^e \tag{8}$$

Among these variables, the decision variable $r_i \in \{0,1\}$ indicates whether the observation i belongs to the support of the clause $b(\cdot)$; specifically, $b(x_i) = 1$ if and only if $r_i = 1$. Furthermore, the constant M is introduced, satisfying the condition $M \geq M_c$, where M_c is the complexity threshold. Equation (6) ensures that for an observation $i \in \mathcal{N}^e$ to qualify as a support for $b(\cdot)$, it must fulfill all the conditions specified by $b(\cdot)$. The support of the MSGC SE is typically much more extensive compared to that of the MCGC SE. However, owing to its increased complexity, the computational process to solve the MSGC models (4)-(8) is significantly slower.

3.3. Model Based on q-Consistency Improvement

A globally consistent SE $b(\cdot) \to y$ can be interpreted as a 1-consistent or 100% consistent rule, as it necessitates the consistency property to hold for all observations $i \in N$. However, locating such a 1-consistent rule can prove to be challenging. Particularly, in the context of significantly large datasets, the MCGC and MSGC models often yield rules with high complexity or limited support, or even infeasible solutions. The underlying reason is straightforward: for intricate datasets, the existence of a 1-consistency rule (e.g., $M_c \leq 4$) with reasonable complexity is improbable. Hence, it is natural to relax the requirement of 1-consistency to a lower agreement level, such as 0.9 or 0.8 agreement. Such a relaxation should be deemed acceptable for SE in numerous practical domains, including credit evaluation.

Let us introduce the concept of q-consistency as a property of SE $b(\cdot) \to y$ in the following manner: For at least a proportion of q of observations $i \in \mathcal{N}$, the condition $y_i = y$ holds true if $b(x_i) = 1$. To further clarify, we define $S_N(b, y)$ as the consistent support set, denoted by $S_N(b, y) = \{i \in \mathcal{N} : b(x_i) = 1, y_i = y\}$. The consistency level of the rule $b(\cdot) \to y$ is formally expressed as $c_{\mathcal{N}}(b, y) = \left|\mathcal{S}_{\mathcal{N}}(b, y)\right| / \left|\mathcal{S}_{\mathcal{N}}(b)\right|$. In simpler terms, the

q-consistency property asserts that $c_N(b,y) \geq q$; this implies that the rule is consistent for at least a proportion of q observations. For observations (x_e, y_e), q aligns with the SE $b(\cdot) \to y$, and it can be paraphrased as follows: "For a subset of observations equal to or greater than q, where $b(\cdot)$ holds true, the predicted outcome is y_e." As an illustrative example in credit assessment, the observed q-consistent SE for the FICO dataset is as follows: "For all 100 individuals with $ExternalRiskEstimate \leq 63$ and $AverageMinFile \leq 48$, the model predicts a high risk of default."

Indeed, the concept of q-consistency naturally extends the q-consistency support maximization problem (MSqC) from the framework of MSGC problems (4)-(8). The fundamental objective of the MSqC problem is to maximize the support of SE $b(\cdot) \to y$, while adhering to the q-consistency constraint $c_N(b,y) \geq q$. Formally, the MSqC problem can be articulated as follows:

$$\max_{b,r} \sum_{i \in \mathcal{N}} r_i \tag{9}$$

$$\text{s.t.} \sum_{p \in \mathcal{P}^e} b_p \left(1 - \delta_{i,p}\right) + r_i \geq 1, \quad \forall i \in \mathcal{N} \setminus \mathcal{N}^e \tag{10}$$

$$\sum_{p \in \mathcal{P}^e} b_p \left(1 - \delta_{i,p}\right) \leq M\left(1 - r_i\right), \quad \forall i \in \mathcal{N} \tag{11}$$

$$\sum_{p \in \mathcal{P}^e} b_p \leq M_c, \tag{12}$$

$$\sum_{i \in \mathcal{N}} \left(a_i - q\right) r_i \geq 0, \tag{13}$$

$$b_p, r_i \in \{0,1\}, \quad \forall p \in \mathcal{P}^e, \forall i \in \mathcal{N} \tag{14}$$

Compared to the MSGC model (4)-(8), four modifications have been made:

1) Binary variables r_i, which indicate supportive observations, are now defined for all observations $i \in \mathcal{N}$, instead of being limited to $i \in \mathcal{N}^e$.

2) Constraints (10) have been adapted to include r_i, enabling the representation of inconsistent support ($r_i = 1$)) for observations $i \in \mathcal{N} \setminus \mathcal{N}^e$.

3) Constraints (11) have been extended to cover all observations $i \in \mathcal{N}$ instead of being restricted to $\mathcal{N}^e$.

4) Constraints (13) represents the q-consistency constraint (equivalent to $c_N(b,y) \geq q$), where binary constants $a_i = 1$ if and only if $i \in N^e$.

4. Methodology

The resolution of MCGC and MSGC models hinges on a branch and bound solver. Nevertheless, conventional branch-and-bound methodologies encounter significant challenges when grappling with expansive datasets. The applicability of branch and bound techniques to these models is fraught with inefficacy and inefficiency. While the MCGC model offers swift and uncomplicated resolution, its branch-and-bound exact solution support remains wanting. In contrast, the MSGC and MSqC models strive to optimize support, yet their solution time proves untenable for pragmatic applications. The MSqC problem is a special kind of

MILP problem. In recent years, there have been many ML-B & B methods to solve this kind of problem. This paper studies how the distribution of the sample branch data of the MSqC problem affects the performance of the ML-B & B model, so as to improve the solution to the MSqC problem's efficiency.

4.1. Optimization of the Sample Distribution

Contemporary investigations [16] [19] frequently adopt uniform sampling of branching decisions across nodes. Consequently, the resultant ML-policies exhibit akin approximation accuracies per node in a given B & B iteration. In contrast, our proposition herein is to endow crucial nodes with elevated sampling probability, thereby elevating approximation accuracies for pivotal choices. It is widely acknowledged that choices within shallower nodes bear greater significance than those in deeper nodes. Hence, our approach chiefly relies on depth-related insights to gauge the sampling probability per node. This strategy is then appraised by assessing the performance of the trained models.

4.2. Problems of the *q*-Consistent Summary-Explanation Solution Depth and Nodes Characteristics

Prior to delving into sampling strategies, it proves instructive to scrutinize the B & B solution process through the lens of depth and node attributes. **Figure 1** delineates the distribution of maximum depths and visited nodes for ten thousand solutions encompassing diverse MSqC instances. Evident from **Figure 1(a)**, a substantial majority of MSqC instances converge within a B & B tree depth of 100. The distribution of maximum visited nodes exhibits a prolonged tail, with the most extensive node visits reaching 25,746. As depicted in **Figure 1(b)**, the upper echelon (the least populous) 90% of visited node counts illustrates a similar trend, reflecting a characteristic of pronounced skewness in the distribution of visited nodes.

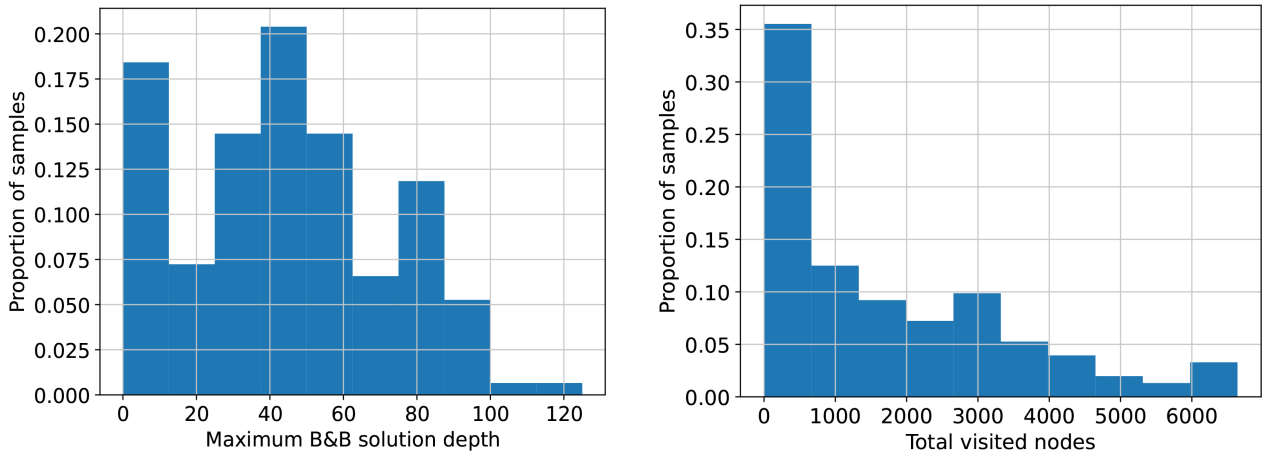


Figure 1. The distributions of the maximum depths and visited nodes of the B & B solution trees for two hundred MSqC instances. (a) Max depth; (b) Max visited nodes (top 90%).

4.3. Sampling Strategies

Within this subsection, an array of sampling strategies is elucidated. Primarily, the conventional uniform sampling approach, employed in earlier inquiries, is expounded upon. Following this, the innovative heavy-head sampling strategy is introduced, primarily directed towards shallower-depth nodes and nodes accessed early in the process. Figure 2 serves to exemplify the conceptual framework of these distinct sampling strategies, acknowledging a minor divergence from actuality due to the general incompleteness inherent in a B & B tree's depiction.

4.3.1. Uniform Sampling

The precedent uniform sampling strategy finds broad application in antecedent investigations related to variable selection in ML-B & B [16] [19]. Within uniform sampling, a fixed probability p is utilized to record the state of the B & B process, along with the scores and actions furnished by the strong branching policy, prior to any branching decision on a given node. Following the resolution of a MILP instance, a fresh B & B solution episode commences with the sampler incorporating another instance. Notably, the sampling probability, often set at $p = 0.05$ during implementation, serves to uphold the diversity of amassed samples for subsequent use as training data.

4.3.2. Heavy-Head Sampling

In light of the preexisting understanding that decisions of shallower depths tend to hold heightened significance, the formulated heavy-head sampling strategy is meticulously crafted to bolster the representation of such pivotal determinations within the amassed samples. Under the heavy-head sampling paradigm, a branching decision is subject to sampling with a probability exceeding p if its depth does not surpass K_{depth} and the count of visited nodes (within the ongoing solution episode) remains within the confines of K_{nodes}. Alternatively, should these conditions not be met, the decision is subjected to sampling with a consistent probability p, analogous to the established uniform sampling method.

5. Computer Experiments

Figure 3 portrays the schematic illustration of our experimental framework, a derivation from Gasse *et al.*'s learn-to-branch project [16]. Notably, our experimental methodology diverges through the following distinct aspects:

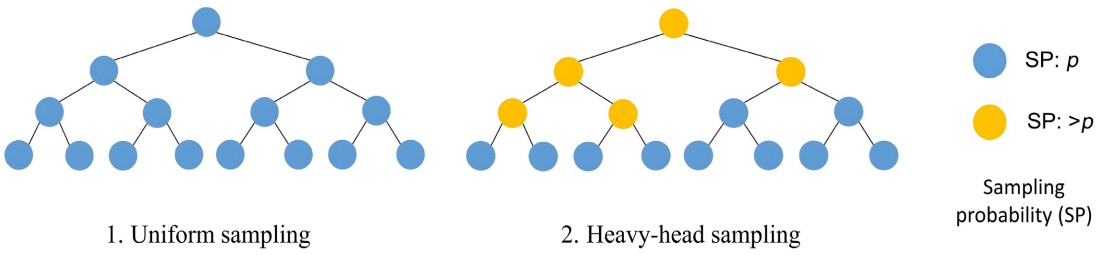


Figure 2. Sample distributions resulted from different sampling strategies.

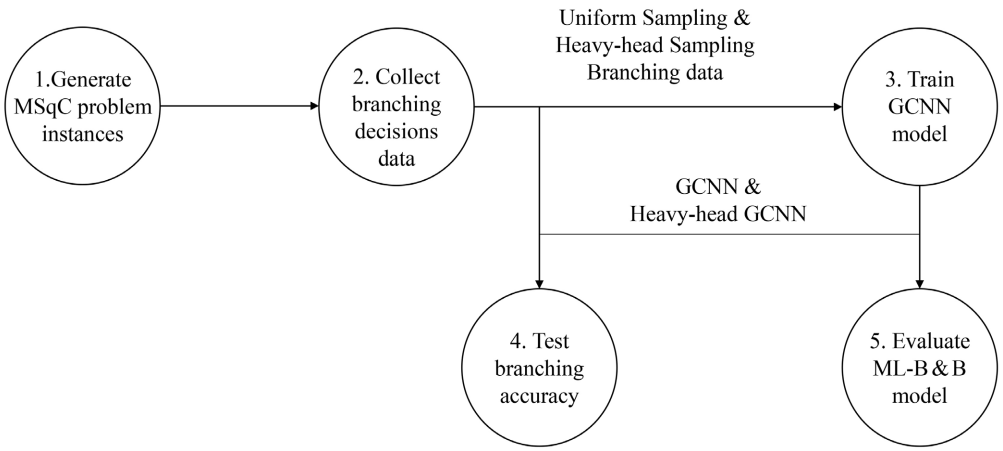


Figure 3. A schematic representation of the experimental framework.

1) Divergent sampling strategies are deployed to amass branching data: the uniform sampling strategy and the heavy-head sampling strategy;

2) The testing of GCNNs' branching accuracies extends across two distinct sample distributions, thus facilitating a more pronounced observation of the approximation bias inherent in the trained GCNN model.

5.1. Experimental Framework

This work's experiments unfold through a sequence of five pivotal stages.

1) Generate MSqC problem instances: The instances set use the FICO datasets. It contains training examples, validation examples, test examples and three transfer examples with different complexity, namely Easy, Hard;

2) Gather Branching Data: Employing SCIP 7.0 [6], we solve the generated instances while obtaining branching decisions through the uniform and heavy-head sampling strategies. This results in distinct training, validation, and testing datasets for each approach;

3) Train GCNN Models: We adopt the GCNN model from [16], training it using the branching data derived from the two distinct sampling strategies. As a consequence, we develop two distinct GCNN models—GCNN and heavy-head GCNN;

4) Assess Branching Accuracy: Our investigation entails evaluating the trained GCNNs' branching accuracy across two distinct sample distributions—node-uniform and heavy-head. This aids in a more nuanced observation of the trained GCNN model's approximation bias;

5) Evaluate ML-B & B Efficiency: The yardstick for comparing the sampling strategies resides in the efficiency of the final Machine Learning Branch and Bound (ML-B & B) model for MSqC problem solving. By integrating the trained GCNNs into SCIP's B & B solution process and replacing the default SCIP brancher, we finally test the effectiveness of the sampling strategies.

We perform experiments with two different setups. The training scheme is the default setup used in previous studies [16] [19], Specifically, 10 K (1 K = 1000)

branching samples are extracted from 200 instances for training, 2 K branching samples from 50 instances for validation, and the same for testing. The training process uses a batch size of 8, epoch size (number of batches per epoch) of 312, and max epochs of 1 K. The experiments are performed separately for both the uniform sampling strategy and the heavy-head sampling strategy.

5.2. Comparison of Sample Distributions

We generate MSqC problems of varying complexities and subsequently solve the training instances using SCIP 7.0 [6]. Branching data is acquired through two distinct sampling strategies: uniform sampling and heavy-head sampling. For ease of reference, we denote the branching data sampled via uniform sampling as following a node-uniform distribution. This characterization indicates that selecting a branching decision from this distribution is equivalent to a uniform random choice from all nodes across B & B solution trees. Similarly, branching data collected through heavy-head sampling is said to follow a heavy-head distribution. **Figure 4** visually demonstrates the depth distribution in these sample distributions. Notably, the heavy-head strategy significantly emphasizes collecting branching data from shallow nodes (depth in [0, 19]), in contrast to the uniform strategy, which yields fewer data from deeper nodes.

5.3. Comparison of Branching Accuracies

This section assesses the GCNN models' branching accuracy within the framework of dataset training schemes. We examine models trained through uniform and heavy-head sampling strategies for their branching accuracy. To maintain consistency with [16], we adopt four metrics for measuring accuracy: acc@1, acc@3, acc@5, and acc@10.

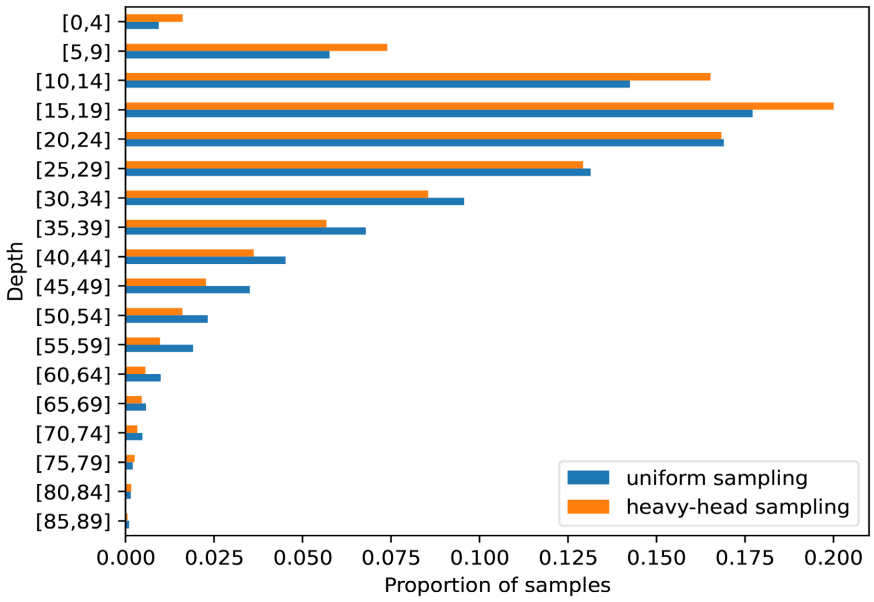


Figure 4. Distributions of the collected branching data over B & B node depths.

Table 1 presents a comprehensive comparison, revealing that the heavy-head GCNN exhibits superior overall branching accuracy when contrasted with both the GCNN and TREES models [20]. The observed enhancement in accuracy underscores the statistically significant advantages associated with the heavy-head sampling strategy.

5.4. Comparison of Problem-Solving Efficiency

In this subsection, MSqC instances of different difficulties are generated. For consistency with [16], the metrics for problem-solving Efficiency are the employed metrics encompass: 1) the normalized 1-shifted geometric mean of solving times (Time), 2) the normalized final node counts of solutions (Node), and 3) the count of instances where each branching policy yields the fastest solving time among 100 attempts (Win). Considering the large variance in performance over different problem instances, the means of Time and Node metrics are normalized against uniform-GCNN to facilitate comparison between different GCNN models. In the form of "mean r + std * 100%", "r" represents the 1-shifted geometric mean for GCNN. The ML-B & B models are obtained by embedding trained models into the SCIP's B & B solution process to replace the default SCIP brancher. To evaluate each problem difficulty (Easy, Hard), five training seeds are used for 20 new instances, leading to 100 solving attempts for each difficulty. As shown in Table 2, the results show that the heavy-head sampling strategy performs better on all of the performance metrics.

Table 1. ML-B & B model branch accuracy comparison.

Problem	Accuracy Level	Model		
		Trees	GCNN	Heavy-Head GCNN
MSqC	acc@1	0.2530 ± 0.0000	0.4839 ± 0.0063	**0.5147 ± 0.0073**
	acc@3	0.4665 ± 0.0000	0.7083 ± 0.0090	**0.7488 ± 0.0045**
	acc@5	0.5945 ± 0.0000	0.8008 ± 0.0111	**0.8375 ± 0.0042**
	acc@10	0.7325 ± 0.0000	0.8905 ± 0.0081	**0.9151 ± 0.0044**

Table 2. ML-B & B solving problem efficiency comparison.

Problem	Type	Model	Node	Time	Wins	T-Stats (p-Value)
MSqC	Easy	Trees	2.7985r ± 30.88%	1.5641r ± 25.19%	6	12.58 (0.0000)
		GCNN	1.0000r ± 68.24%	1.0000r ± 59.10%	30	0.00 (1.0000)
		Heavy-head GCNN	**0.8456r ± 50.41%**	**0.7338r ± 40.91%**	**64**	**−3.87 (0.0001)**
	Hard	Trees	2.8580r ± 38.31%	1.5915r ± 34.08%	7	7.59 (0.0000)
		GCNN	1.0000r ± 71.66%	1.0000r ± 63.64%	29	0.00 (1.0000)
		Heavy-head GCNN	**0.8353r ± 48.63%**	**0.7386r ± 40.57%**	**64**	**−4.56 (0.0000)**

6. Conclusion

In this paper, we have advanced the proposition that demanding absolute 100% consistency from summary-explanations is prone to excessiveness within pragmatic contexts. We contend that minor inconsistencies are inherent and acceptable within summary-explanations for substantial, practical datasets, and that these inconsistencies can indeed be harnessed to enhance their overall support. Building upon this rationale, we introduce the MSqC problem, aiming to enhance support while allowing modest concessions in consistency. Subsequently, we discern that addressing the MSqC problem proves notably more intricate than its MSGC counterpart, rendering traditional B & B solvers for exact solutions unsuitable. Consequently, we harness the heavy-head distribution of maximum depths within the MSqC problem and employ a heavy-head sampling strategy for imitation learning in B & B for MSqC. Empirical findings substantiate that this strategy substantially heightens the accuracy and efficiency of solving MSqC issues, as demonstrated through statistical testing.

Fund Project

This work was supported in part by the Natural Science Foundation of China under Grants 62006095, in part by the Hunan Natural Science Foundation of China under Grant 2021JJ40441, and in part by the Research Foundation of Education Bureau of Hunan Province, China, under Grants 20B470, by the National Innovation and Entrepreneurship Training Project S202010531027, and by the Jishou University Graduate Research Innovation Project JGY2023073.

Conflicts of Interest

The author declares no conflict of interest.

References

[1] Qiao, L.T., Wang, W.J. and Lin, B. (2021) Learning Accurate and Interpretable Decision Rule Sets from Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, **35**, 4303-4311. https://doi.org/10.1609/aaai.v35i5.16555

[2] Lawless, C., Dash, S., Gunluk, O. and Wei, D. (2021) Interpretable and Fair Boolean Rule Sets via Column Generation. arXiv Preprint arXiv: 2111.08466. https://doi.org/10.48550/arXiv.2111.08466

[3] Liu, W.N., Fan, H. and Xia, M. (2022) Credit Scoring Based on Tree-Enhanced Gradient Boosting Decision Trees. *Expert Systems with Applications*, **189**, Article 116034. https://doi.org/10.1016/j.eswa.2021.116034

[4] Rudin, C. and Shaposhnik, Y. (2023) Globally-Consistent Rule-Based Summary-Explanations for Machine Learning Models: Application to Credit-Risk Evaluation. *Journal of Machine Learning Research*, **24**, 1-44. https://doi.org/10.2139/ssrn.3395422

[5] Rudin, C. and Radin, J. (2019) Why Are We Using Black Box Models in Ai When We Don't Need to? A Lesson from an Explainable AI Competition. *Harvard Data Science Review*, **1**, 1-9. https://doi.org/10.1162/99608f92.5a8a3a3d

[6] Gamrath, G., Anderson, D., Bestuzheva, K., Chen, W.K., Eifler, L., Gasse, M., Ge-

mander, P., Gleixner, A., Gottwald, L., Halbig, K., *et al.* (2020) The SCIP Optimization Suite 7.0. ZIB-Report 20-10, Zuse Institute Berlin, Berlin, Germany. https://opus4.kobv.de/opus4-zib/files/7802/scipopt-70.pdf

[7] Chen, Y.Z., Chen, W., Chandra Pal, S., Saha, A., Chowdhuri, I., Adeli, B., Janizadeh, S., Dineva, A.A., Wang, X.J. and Mosavi, A. (2022) Evaluation Efficiency of Hybrid Deep Learning Algorithms with Neural Network Decision Tree and Boosting Methods for Predicting Groundwater Potential. *Geocarto International*, **37**, 5564-5584. https://doi.org/10.1080/10106049.2021.1920635

[8] Ribeiro, M.T., Singh, S. and Guestrin, C. (2016) "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings of the* 2016 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Demonstrations*, New York, 13 August 2016, 1135-1144. https://doi.org/10.18653/v1/n16-3020

[9] Bücker, M., Szepannek, G., Gosiewska, A. and Biecek, P. (2022) Transparency, Auditability, and Explainability of Machine Learning Models in Credit Scoring. *Journal of the Operational Research Society*, **73**, 70-90. https://doi.org/10.1080/01605682.2021.1922098

[10] Ribeiro, M.T. Singh, S. and Guestrin, C. (2018) Anchors: High-Precision Model-Agnostic Explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, New Orleans, 2-7 February 2018. https://doi.org/10.1609/aaai.v32i1.11491

[11] Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F. and Giannotti, F. (2018) Local Rule-Based Explanations of Black Box Decision Systems. arXiv Preprint arXiv: 1805.10820. https://doi.org/10.48550/arXiv.1805.10820

[12] Zafar, M.R. and Khan, N. (2021) Deterministic Local Interpretable Model-Agnostic Explanations for Stable Explainability. *Machine Learning and Knowledge Extraction*, **3**, 525-541. https://doi.org/10.32920/22734320.v1

[13] Huang, Q., Yamada, M., Tian, Y., Singh, D. and Chang, Y. (2022) Graphlime: Local Interpretable Model Explanations for Graph Neural Networks. *IEEE Transactions on Knowledge and Data Engineering*, **35**, 6968-6972. https://doi.org/10.1109/tkde.2022.3187455

[14] Alvarez, A.M., Louveaux, Q. and Wehenkel, L. (2017) A Machine Learning-Based Approximation of Strong Branching. *INFORMS Journal on Computing*, **29**, 185-195. https://doi.org/10.1287/ijoc.2016.0723

[15] Osa, T., Pajarinen, J., Neumann, G., Bagnell, J.A., Abbeel, P., Peters, J., *et al.* (2018) An Algorithmic Perspective on Imitation Learning. Foundations and Trends in Robotics, Boston. https://doi.org/10.1561/9781680834116

[16] Gasse, M., Chételat, D., Ferroni, N., Charlin, L. and Lodi, A. (2019) Exact Combinatorial Optimization with Graph Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, **32**, 15580-15592.

[17] Zhou, J., Cui, G.Q., Hu, S.D., Zhang, Z.Y., Yang, C., Liu, Z.Y., Wang, L.F., Li, C.C. and Sun, M.S. (2020) Graph Neural Networks: A Review of Methods and Applications. *AI open*, **1**, 57-81. https://doi.org/10.1016/j.aiopen.2021.01.001

[18] Buffelli, D., Liò, P. and Vandin, F. (2022) Sizeshiftreg: A Regularization Method for Improving Size-Generalization in Graph Neural Networks. 36*th Conference on Neural Information Processing Systems*, New Orleans, 28 November-9 December 2022, 31871-31885.

[19] Gupta, P., Gasse, M., Khalil, E., Mudigonda, P., Lodi, A. and Bengio, Y. (2020) Hybrid Models for Learning to Branch. 34*th Conference on Neural Information Processing Systems* (*NeurIPS* 2020), Vancouver, 6-12 December 2020, 18087-18097.

https://doi.org/10.48550/arXiv.2006.15212

[20] Geurts, P., Ernst, D. and Wehenkel, L. (2006) Extremely Randomized Trees. *Machine Learning*, **63**, 3-42. https://doi.org/10.1007/s10994-006-6226-1