Active Learning and Covering Problems with Precedence

author names withheld

Editor: Under Review for COLT 2026

Abstract

In the Bayesian Active Learning a hidden hypothesis is required to be uncovered. To do so, the learner is allowed to perform tests, each of which reveals partial information about the hidden hypothesis. Upon receiving this information, the learner adaptively selects the next test to be performed. The goal is to uncover the hidden hypothesis while performing as few tests as possible in the worst or average case.

In the covering problems, we are given a set of items and a collection of subsets that cover these items. The objective is to select a sequence of subsets that covers all items, which minimizing the worst or average covering cost.

For both types of problems, a natural constraint may arise that some tests can only be performed only after certain other tests (or some subsets can only be selected after selecting certain other subsets). We model such constraints using directed acyclic graphs (DAGs) that impose precedence on the tests or subsets. This paper explores the connection of active learning and covering problems under such constraints.

We show that given any bicriteria $(O(1), \alpha)$-approximation ratio for the Precedence Constrained Set Cover, we can obtain an $O(\alpha \cdot \log n)$-approximation ratio for the Worst Case Active Learning with precedence constraints, where n is the number of hypothesis. Similarly, we prove that given any $O(\beta)$-approximation ratio for the Precedence Constrained Min-Sum Set Cover, we can obtain an $O(\beta \cdot \log n)$-approximation ratio for the Average Case Active Learning with Precedence Constraints. Finally, we provide several approximation algorithms for the Set Cover and Min-Sum Set Cover problems with various types of precedence constraints.

Keywords: Bayesian active learning, Set cover, Precedence constraints, Approximation Algorithms, Decision Trees

Ogolne uwagi:

- **wszelkie uwagi pisze jako komenda ... dzieki czemu na koniec latwo sie pozbyc; nie bede raczej uwag dorzucal w mailach, aby nie zniknelo; wszystko co ponizej oczywiscie do dyskusji, a gdy bedzie zgoda i bedzie zaimplementowane, to bede usuwal artefakty.**

- **przejrzawszy sporo papierow z poprzedniego COLT, mam obserwacje, ze dobrym/typowym ukladem papieru jest: intro; our contribution; related work; outline(opcjonalnie); preliminaries; wyniki; appendix.**

- **intro zwykle nie jest zbyt rozlekle oraz prawie zawsze pozbawione lania wody. Czesto od razu definicja problemu, aby wprowadzic pojecia, aby moc szybko formalnie podac wyniki (our contribution)**

- **"front" artykulu, czyli wszystko do preliminaries to typowo 4-5 stron,**

- **front we wszystkich miejcach zawiera zwykle odnosniki do literatury a sekcja "related work" jest czesto tytulowana "other related work" lub cos w tym rodzaju**

- czytelnik powiniem poza dowodami rozumiec baze przeczytawszy front (czyli rozumiec wyniki, widziec co papier robi) a jesli chce sie dowiedziec jak/dlaczego (dowody) to idzie dalej. Czesto recenzent jest leniwy i nie zajrzy dalej niz front... niestety.

- W zwiazku z powyzszym sekcja "our contribution" (potencjalnie w tytule dodamy "and techniques" jak Michal sugeruje) powinna sie pochwalic takze jakimis ciekawszymi trickami lub technikami uzytymi pozniej w dowodach.

1. Introduction

pomysl na intro: zdefiniowac dwa glowne problemy: learning + set cover; zapowiedziec, ze sa powiazane ze soba i celem papieru jest przestudiowanie tych zaleznosci plus usyskanie konkretnych wynikow; pytanie/do sprawdzenia: czy ktores wyniki przypadkiem poprawiaja lub sa tozsame z najlepszymi znanymi bez precedensow; ewentualne inna "marketingowe" uwagi.

Consider a set $\mathcal{H}$ of n *hypotheses*, a set $\mathcal{T}$ of m *tests* and an unknown *target hypothesis* $h^* \in \mathcal{H}$ that needs to be discovered through an adaptive learning process. Each test $t \in \mathcal{T}$ is a partition of $\mathcal{H}$, that is, t consists of subsets of $\mathcal{H}$ such that $x \cap y = \emptyset$ for any $x.y \in t$ and $\bigcup t = \mathcal{H}$. As a result of executing a test $t \in \mathcal{T}$, questioner recives a *reply* that reveals $x \in t$ such that $h^* \in x$. That is, the questioner learns which subset of $\mathcal{H}$ that belongs to t contains the target. Each subsequent test is selected by questioner by taking into acocunt replies from all test to date. Without formally stating an optimization criterion we refer to the above as the *Adaptive Learing Process* (AL). (Another widely used name in the literature is the decision tree construction). The goal for the questioner is to output h^*.

Consider an arbitrary partial order $(\mathcal{T}, \preceq)$ that introduces a precedence relation between tests. This leads us to the two adaptive learning problems in which order to perform a test t, all its predecesors had to be performed previously. Hence we have the *Worst Case Adaptive Learing with Precedences* (WCALP) in which the goal is to compute the AL that respects the precedence constraints and outputs the target h^* by performing the minimum number of tests in the worst case. Similarly, in the *Average Case Adaptive Learing with Precedences* (ACALP) the optimization criterion changes to minimizing the number of queries done on average.

In this work we study connections between adaptive lerning with precedences and the covering problems defined as follows. We are given a set $\mathcal{U}$ of n items, a collection $\mathcal{S}$ of m subsets of $\mathcal{U}$, such that $\bigcup \mathcal{S} = \mathcal{U}$, an arbitrary partial order $(\mathcal{S}, \preceq)$ on these subsets and an integer k. We say that a subfamily $\mathcal{C} \subseteq \mathcal{S}$ *covers* at least k items from $\mathcal{U}$ if $|\bigcup \mathcal{C}| \geq k$. We ask for a $\mathcal{C} \subseteq \mathcal{S}$ that covers at least k items from $\mathcal{U}$ and for each $x \in \mathcal{C}$ and each $y \in \mathcal{S}$ such that $y \preceq x$ it holds $y \in \mathcal{C}$. In the *Precedence Constrained Set Cover* (PCSC) the goal is to minimize $|\mathcal{C}|$. A permutation $C_1, \ldots, C_k)$ of the elements in $\mathcal{C}$ is *consistent* with the partial order $(\mathcal{S}, \preceq)$ if for any C_i and C_j such that $C_i \preceq C_j$ it holds $i < j$. The *coverage time* of a $x \in \bigcup \mathcal{C}$ is the minimum index i such that $x \in C_i$. In the *Precedence Constrained Min-Sum Set Cover* (PCMSSC) the goal is to find a sequence $(C_1, \ldots, C_k)$ that minimizes the total coverage time of all items in $C_1 \cup \cdots \cup C_k$.

1.1. Our contribution

Pomysl na rozdzial:

- **zajawka, że wprpwadzimy nowe inne problemy (raz - jak pomocnicze; dwa - jako dopelnienie obrazu roznych rzeczy z litearatury)**

- **zdefiniowac pozostałe problemy z "diagramu" zaleznosci miedzy nimi**

- **diagram**

- **najwazniejsze twierdzenia**

- **tabelka na podsumowanie**

Consider following problems:

- The *Precedence Constrained Bayesian Active Learning Problem* consists a set of $\mathcal{H}$ of n hypothesis, a set $\mathcal{T}$ of m tests and a DAG (directed acyclic graph) $\mathcal{F} = \{\mathcal{T}, \preceq\}$ encoding the precedence constraints between available tests. Among $\mathcal{H}$ a hidden hypothesis is required to be encovered. To do so, the learner is allowed to perform tests, each of which reveals partial information about the hidden hypothesis. Upon receiving this information, the learner adaptively selects the next test to be performed. Importantly, in order to perform such test the learner needs to perform all of its predecesors in $\mathcal{F}$ first. The goal is to uncover the hidden hypothesis while performing as few tests as possible. Depending on the chosen criterion we distinguish between the *Precedence Constrained Worst Case Active Learning* (PCWCAL) and *Precedence Constrained Average Case Active Learning* (PCACAL) problems.

- The *Precedence Constrained Covering Problem* consists of a set of n items $\mathcal{U}$, a collection $\mathcal{S}$ of m subsets of $\mathcal{U}$ that cover these items, and a DAG $\mathcal{F} = \{\mathcal{S}, \preceq\}$ encoding the precedence constraints between available subsets. The goal is to select a sequence of tests that covers at least K items. Depending on the chosen criterion we distinguish between the *Precedence Constrained Set Cover* (PCSC) and *Precedence Constrained Min-Sum Set Cover* (PCMSSC) problems. In the first we are only interested in minimizing the number of selected subsets, while in the second we want to minimize the average time it takes to cover an item.

1.2. Our results and techniques

precedence/problem	PCSC	PCMSSC	PCWCAL	PCACAL
none	$O(\log n)$	4	$O(\log n)$	$O(\log n)$
inforest	$O(\log n)$*	4	$O(\log n)$*	$O(\log n)$*
outforest	$O(\log^2 n)$**	$O(\log n)$**	$O(\log^2 n)$*	$O(\log^2 n)$*
general	$O(\sqrt{n}\log n)$*	$O(\sqrt{n})$	$O(\sqrt{n}\log n)$*	$O(\sqrt{n}\log n)$*

Table 1: Approximation algorithms for various covering and active learning problems under different precedence constraints. (* denotes new results, ** denotes previously unmentioned corollaries of known results)

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}
h_1	0	0	0	0	0	0	2	0	0	0	0	1
h_2	0	0	0	0	0	1	0	0	0	0	0	0
h_3	0	1	0	1	0	0	0	0	0	0	0	0
h_4	0	0	0	0	1	0	0	1	0	0	0	0
h_5	0	0	0	0	0	0	1	0	0	1	1	0
h_6	1	0	0	2	0	0	0	0	0	1	0	0
h_7	0	0	0	0	0	2	0	0	1	2	0	0
h_8	0	0	1	0	0	0	0	0	0	3	0	0
h_9	0	0	0	0	0	0	0	1	1	0	2	0
h_{10}	0	0	0	1	0	0	0	0	0	0	0	2

(a) Hypotheses and tests table

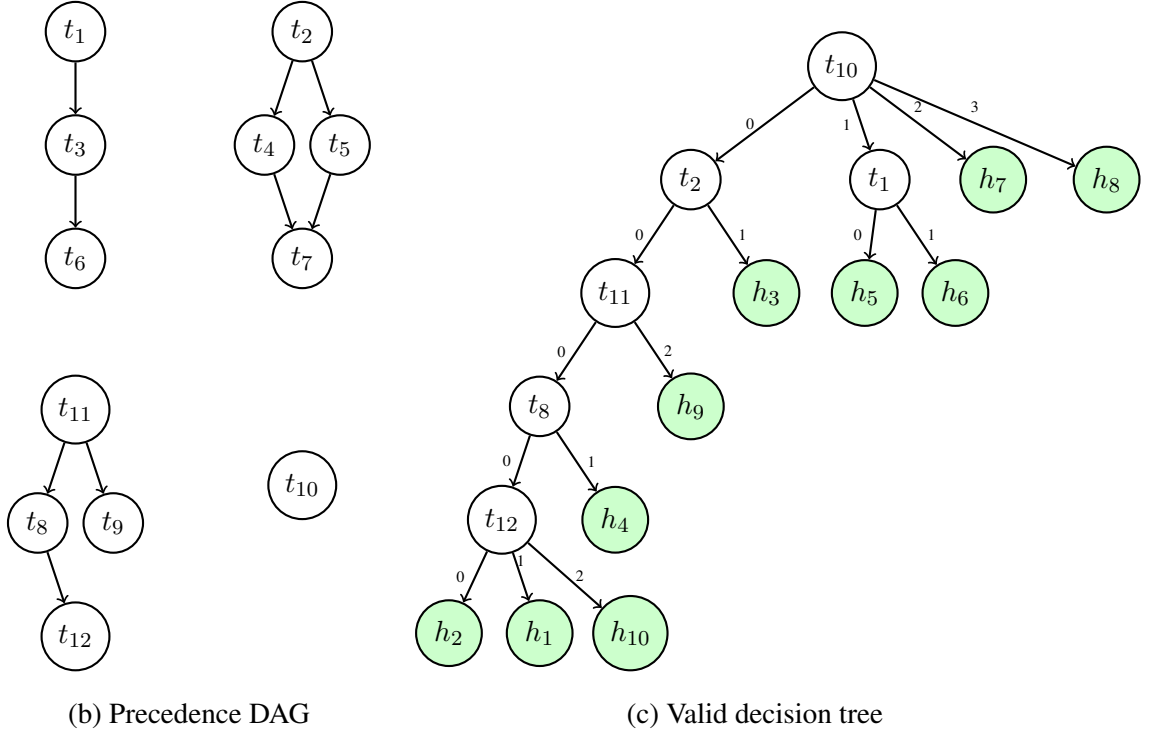(b) Precedence DAG

(c) Valid decision tree

Figure 1: Example of a PCAL instance with 10 hypotheses and 12 tests. (a) Hypotheses-tests table. (b) Precedence DAG with four components. (c) A valid decision tree solution respecting precedence constraints.

(a) Universe and covering sets

(b) Precedence DAG

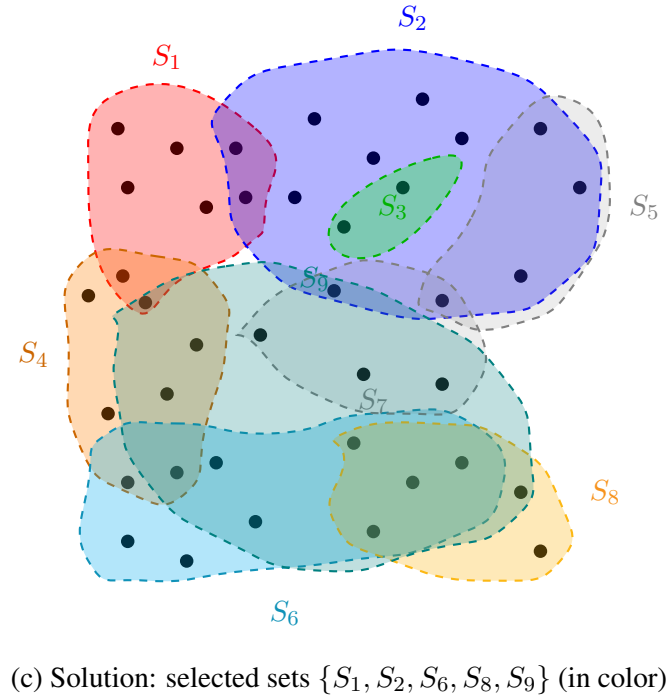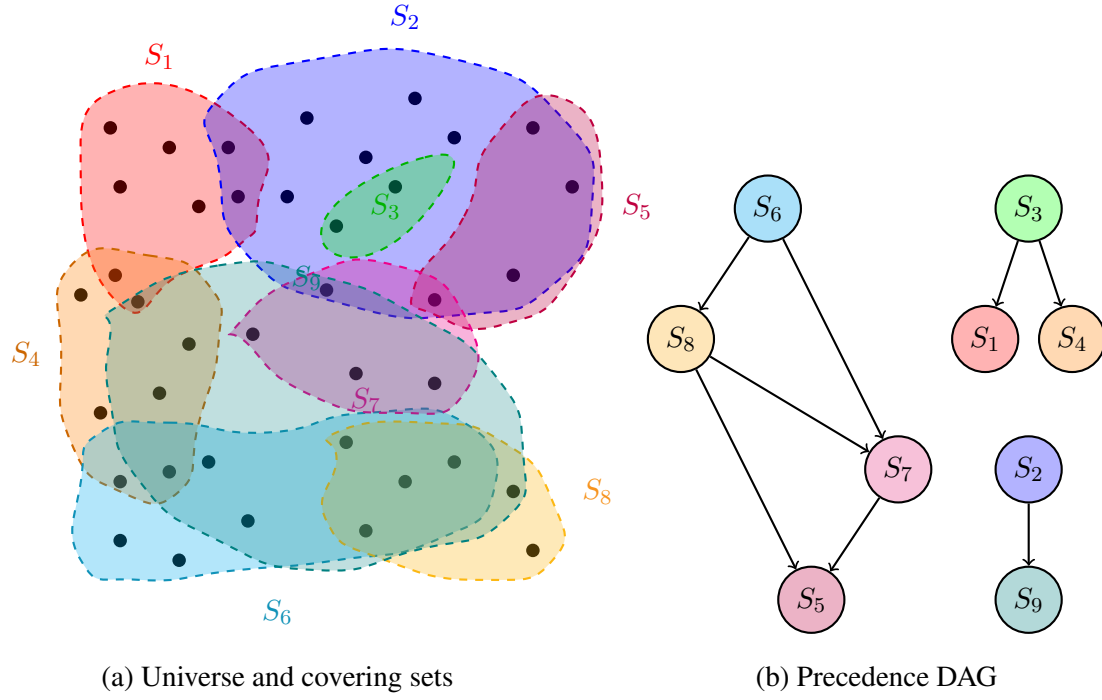(c) Solution: selected sets $\{S_1, S_2, S_6, S_8, S_9\}$ (in color)

Figure 2: Example of a PCCP instance with 39 elements and 9 covering sets. (a) Universe with covering sets. (b) Precedence DAG with three components. (c) Solution using 7 selected sets (colored).
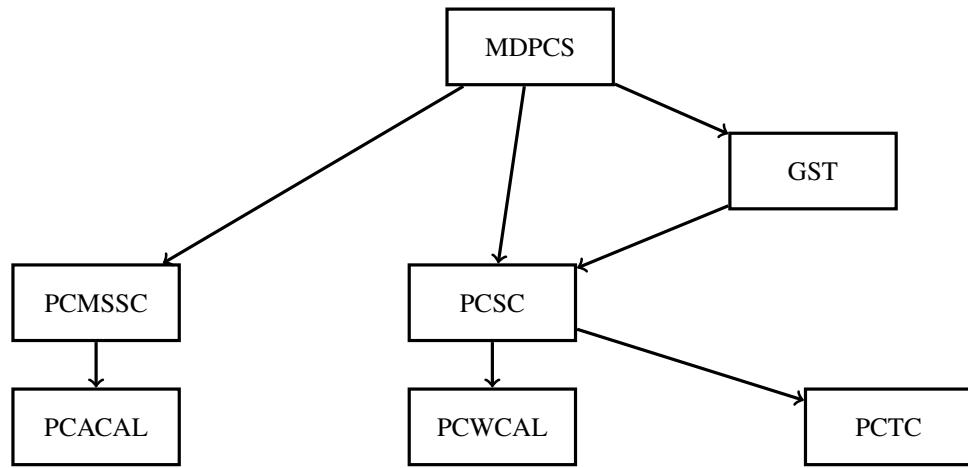
Figure 3: Relationships between covering and active learning problems, $\Pi_1 \rightarrow \Pi_2$ denotes that an approximation algorithm for problem Π_1 implies an approximation algorithm for problem Π_2.

2. Preliminaries

Definition 1 (Precedence constrained set cover (PCSC)) *Given a universe $\mathcal{U}$ of n items, a collection $\mathcal{S}$ of m subsets of $\mathcal{U}$, a DAG $\mathcal{F} = \{\mathcal{S}, \preceq\}$ encoding precedence constraints, and a coverage requirement K, find a precedence-closed subfamily $\mathcal{C} \subseteq \mathcal{S}$ that covers at least K items while minimizing $|\mathcal{C}|$.*

Definition 2 (Precedence constrained min-sum set cover (PCMSSC)) *Given a universe $\mathcal{U}$ of n items, a collection $\mathcal{S}$ of m subsets of $\mathcal{U}$, a DAG $\mathcal{F} = \{\mathcal{S}, \preceq\}$ encoding precedence constraints, and a coverage requirement K, find a precedence-closed sequence of sets that covers at least K items while minimizing the average time (position in sequence) at which items are covered.*

Definition 3 (Precedence constrained test cover (PCTC)) *Given a set $\mathcal{H}$ of n hypotheses, a set $\mathcal{T}$ of m tests, and a DAG $\mathcal{F} = \{\mathcal{T}, \preceq\}$ encoding precedence constraints, find a precedence-closed subfamily of tests that distinguishes all pairs of hypotheses.*

Definition 4 (Precedence constrained worst case active learning (PCWCAL)) *Given a set $\mathcal{H}$ of n hypotheses, a set $\mathcal{T}$ of m tests, and a DAG $\mathcal{F} = \{\mathcal{T}, \preceq\}$ encoding precedence constraints, construct a decision tree respecting precedence constraints that identifies any hypothesis from $\mathcal{H}$ while minimizing the worst-case depth of the tree.*

Definition 5 (Precedence constrained average case active learning (PCACAL)) *Given a set $\mathcal{H}$ of n hypotheses, a set $\mathcal{T}$ of m tests, a DAG $\mathcal{F} = \{\mathcal{T}, \preceq\}$ encoding precedence constraints, and a probability distribution over $\mathcal{H}$, construct a decision tree respecting precedence constraints that identifies any hypothesis from $\mathcal{H}$ while minimizing the expected depth (average case cost).*

Definition 6 (Group Steiner Tree (GST)) *Given an undirected graph $G = (V, E)$ with edge costs, a root vertex $r \in V$, and groups $g_1, \ldots, g_k \subseteq V$, find a minimum-cost tree T rooted at r that contains at least one vertex from each group g_i.*

For a subfamily $\mathcal{A} \subseteq \mathcal{F}$, we define the coverage as:

$$\text{cov}(\mathcal{A}) \equiv \bigcup_{A \in \mathcal{A}} A$$

For a subset X of the universe, the coverage on X is:

$$\text{cov}(\mathcal{A}, X) = \text{cov}(\mathcal{A}) \cap X$$

The density Δ of a nonempty subfamily $\mathcal{A}$ on subset X is:

$$\Delta(\mathcal{A}, X) \equiv \frac{|\text{cov}(\mathcal{A}, X)|}{|\mathcal{A}|}$$

For convenience, we define $\Delta(\emptyset, X) < 0$.

Definition 7 (Max-Density Precedence-Closed Subfamily (MDPCS)) *Given a family of m sets $\mathcal{G}$, a precedence relation $\prec$, and a set of n items to be covered $R \subseteq \text{cov}(\mathcal{G})$, the MDPCS problem asks to find a precedence-closed subfamily $\mathcal{A} \subseteq \mathcal{G}$ that maximizes $\Delta(\mathcal{A}, R)$.*

For $S \in \mathcal{G}$, let $P[S]$ denote the minimal precedence-closed subfamily of $\mathcal{G}$ containing S (i.e., the ancestors of S including S itself).

3. Active Learning via Covering Problems

We begin with the following folklore lemma concerning both worst and average case learning.

Lemma 8 *Let* $I = (\mathcal{H}, \mathcal{T}, \mathcal{F})$ *be any PCAL instance. Let* $\mathcal{H}' \subseteq \mathcal{H}$. *Then* $\text{OPT}(\mathcal{H}', \mathcal{T}, \mathcal{F}) \leq \text{OPT}(I)$.

3.1. Worst Case

Definition 9 (Pairsep) *Let* D *be any decision tree for the PCWCAL problem instance* $\mathcal{I} = (\mathcal{H}, \mathcal{T}, \mathcal{F})$. *We define a sequence of tests* P_D *called* pairsep *as follows. Initially,* P_D *is empty and* $\mathcal{H}' = \mathcal{H}$. *While* $\binom{|\mathcal{H}'|}{2} > \binom{|\mathcal{H}|}{2}/2$, *we append to* P_D *the test* $r(D_{\mathcal{H}'})$ *and update* $\mathcal{H}'$ *to be the set of hypotheses corresponding to the child of* $D_{\mathcal{H}'}$ *that contains the most hypotheses. If* $\text{COST}(D) = \text{OPT}(\mathcal{I})$, *then we denote* $P^*(\mathcal{I}) = P_D$ *(ties broken arbitrarily).*

It should be remarked that P_D is well-defined, as each test in P_D can have at most one child associated with more than half of the pairs hypotheses in $\mathcal{H}'$. Since P_D is a subpath of D, we also have the following simple observation.

Observation 1 *Let* I *be any instance of PCWCAL. Then* $|P^*(I)| \leq \text{OPT}(I)$.

This allows to use $|P^*(I)|$ as a lower bound on $\text{OPT}(I)$ in the analysis of the approximation algorithm for PCWCAL. We have the following lemma:

Lemma 10 *Let* $I = (\mathcal{H}, \mathcal{T}, \mathcal{F})$ *be any PCWCAL instance. Let* S^* *be the optimal solution for the PCSC on instance* $(\mathcal{U}, \mathcal{T}, \mathcal{F})$ *with* $K = n/2$, *where* $\mathcal{U} = \{(h, j) \mid h, j \in \mathcal{H}\}$ *and a test* t *covers* (h, j) *if it distinguishes* h *and* j. *Then,* $|S^*| \leq |P^*(I)|$.

Proof We show that $P^*(I)$ is a feasible solution for the PCSC instance $(\mathcal{U}, \mathcal{T}, \mathcal{F})$ with $K = n/2$. By definition, for every $\mathcal{H}' \in \mathcal{H} - P^*(I)$, we have $\binom{|\mathcal{H}'|}{2} \leq \binom{|\mathcal{H}|}{2}/2$. Therefore the number of pairs covered by tests in $P^*(I)$ is at least:

$$\binom{|\mathcal{H}|}{2} - \sum_{\mathcal{H}' \in \mathcal{H} - P^*(I)} \binom{|\mathcal{H}'|}{2} \geq \binom{|\mathcal{H}|}{2} - \frac{\binom{|\mathcal{H}|}{2}}{2} = \frac{\binom{|\mathcal{H}|}{2}}{2}.$$

Therefore, by the optimality of S^*, we have $|S^*| \leq |P^*(I)|$ as required. ■

Theorem 11 *If there is an* (γ, α)*-bicriteria approximation algorithm for PCSC then there is an* $O\left(\frac{\alpha}{\log\left(\frac{2\gamma}{2\gamma-1}\right)} \cdot \log n\right)$*-approximation algorithm for PCWCAL. In particular when* $\gamma = O(1)$, *the approximation is* $O(\alpha \cdot \log n)$.

Proof The following observation follows by Lemmas 8 and 14:

Observation 2 *Let* D_S *be the decision tree built on tests from* S *closed under* $\mathcal{F}$. *Then,* $\text{COST}(D_S) \leq \alpha \cdot |P^*(I)|$.

Algorithm 1: The $O(\alpha \cdot \log n)$-approximation algorithm for the PCWCAL
procedure WORSTDECISIONTREE$(\mathcal{H}, \mathcal{T}, \mathcal{F})$
    $\mathcal{U} \leftarrow \{(h, j) \mid h, j \in \mathcal{H}\}$
    foreach $t \in \mathcal{T}$ **do**
        | Mark t as covering $(h, j) \in \mathcal{U}$ if t distinguishes h and j
    end
    $S \leftarrow$ Run the α-approximation algorithm for PCSC on instance $(\mathcal{U}, \mathcal{T}, \mathcal{F})$ with $K = |\mathcal{U}|/2$
    $D \leftarrow D_S \leftarrow$ any decision tree built on tests from S closed under $\mathcal{F}$
    foreach $\mathcal{H}' \in \mathcal{H} - S$ **do**
        $D' \leftarrow$ WORSTDECISIONTREE$(\mathcal{H}', \mathcal{T} - S, \mathcal{F} - S)$
        Attach D' to the leaf of D corresponding to $\mathcal{H}'$
    end
    return D

We are now ready to prove the theorem.

Lemma 12 *Let D be the decision tree returned by* WORSTDECISIONTREE *on input $I = (\mathcal{H}, \mathcal{T}, \mathcal{F})$. Then,* $\text{COST}(D) \leq \dfrac{2\alpha}{\log\left(\frac{2\gamma}{2\gamma-1}\right)} \cdot \log n \cdot OPT(I)$.

Proof We prove the lemma by induction on $p = \binom{|\mathcal{H}|}{2}$. The base case when $n = 1$ and $p = 2$ is trivial since there are no pairs to cover. Assume by induction that for every $I' = (\mathcal{H}', \mathcal{T}, \mathcal{F})$ such that $\mathcal{H}' \in \mathcal{H} - S$ and $n' = |\mathcal{H}'|$ we have $\text{COST}(D') \leq \dfrac{\alpha}{\log\left(\frac{2\gamma}{2\gamma-1}\right)} \cdot \log \binom{n'}{2} \cdot \text{OPT}(I')$, where D' is the decision tree returned by WORSTDECISIONTREE on input I'. We have that:

$$
\begin{aligned}
\text{COST}(D) &\leq \text{COST}(D_S) + \max_{\mathcal{H}' \in \mathcal{H} - S} \text{COST}(D') \\[2mm]
&\leq \alpha \cdot |S^*| + \max_{\mathcal{H}' \in \mathcal{H} - S} \frac{\alpha}{\log\left(\frac{2\gamma}{2\gamma-1}\right)} \cdot \log \binom{n'}{2} \cdot \text{OPT}(I') \\[2mm]
&\leq \alpha \cdot |P^*(I)| + \frac{\alpha}{\log\left(\frac{2\gamma}{2\gamma-1}\right)} \cdot \log \left(\left(\frac{2\gamma-1}{2\gamma}\right) \cdot \binom{n}{2} \right) \cdot \text{OPT}(I) \\[2mm]
&= \alpha \cdot \text{OPT}(I) + \frac{\alpha}{\log\left(\frac{2\gamma}{2\gamma-1}\right)} \cdot \left(\log \binom{n}{2} \right) \cdot \text{OPT}(I) - \alpha \cdot \text{OPT}(I) \\[2mm]
&= \frac{\alpha}{\log\left(\frac{2\gamma}{2\gamma-1}\right)} \cdot \log \binom{n}{2} \cdot \text{OPT}(I)
\end{aligned}
$$

Since $\log \binom{n}{2} \leq 2\log n$, the lemma follows. ∎

∎

3.2. Average Case

Definition 13 (Coversep) *Let D be any decision tree for the PCACAL problem instance $\mathcal{I} = (\mathcal{H}, \mathcal{T}, \mathcal{F})$. We define a sequence of tests C_D called* coversep *as follows. Initially, C_D is empty*

and $\mathcal{H}' = \mathcal{H}$. While $|\mathcal{H}'| > |\mathcal{H}|/2$, we append to C_D the test $r(D_{\mathcal{H}'})$ and update $\mathcal{H}'$ to be the set of hypotheses corresponding to the child of $D_{\mathcal{H}'}$ that contains the most hypotheses. If $\mathrm{COST}(D) = \mathrm{OPT}(\mathcal{I})$, then we denote $C^*(\mathcal{I}) = C_D$ (ties broken arbitrarily).

It should be remarked that C_D is well-defined, as each test in C_D can have at most one child associated with more than half of the hypotheses in $\mathcal{H}'$. Since C_D is a subpath of D, we also have the following simple observation.

Observation 3 *Let I be any instance of PCACAL. Then $\mathrm{COST}(C^*(I)) \le \mathrm{OPT}(I)$.*

This allows to use $\mathrm{COST}(C^*(I))$ as a lower bound on $\mathrm{OPT}(I)$ in the analysis of the approximation algorithm for PCACAL. We have the following lemma:

Lemma 14 *Let $I = (\mathcal{H}, \mathcal{T}, \mathcal{F})$ be any PCACAL instance. Let S^* be the optimal solution for the PCMSSC on instance $(\mathcal{U}, \mathcal{T}, \mathcal{F})$ with $K = n/4$, where $\mathcal{U} = \mathcal{H}$ and a test t element $u \in \mathcal{U}$ if for $u \in U_{t,j}$, $|U_{t,j}| \le \frac{3}{4} \cdot |\mathcal{U}|$. Then, $\mathrm{COST}(S^*) \le \mathrm{COST}(C^*(I))$.*

Theorem 15 *If there is a β-approximation algorithm for PCMSSC then there is an $O(\beta \cdot \log n)$-approximation algorithm for PCACAL.*

Algorithm 2: The $O(\beta \cdot \log n)$-approximation algorithm for the PCACAL

procedure AVERAGEDECISIONTREE($\mathcal{H}, \mathcal{T}, \mathcal{F}$)

    $\mathcal{U} \leftarrow \mathcal{H}$

    foreach $t \in \mathcal{T}$ **do**

        Set t to cover $u \in \mathcal{U}$ if for $u \in U_{t,j}$, $|U_{t,j}| \le \frac{3}{4} \cdot |\mathcal{U}|$

    end

    $S \leftarrow$ Run the α-approximation algorithm for PCMSSC on instance $(\mathcal{U}, \mathcal{T}, \mathcal{F})$ with $K = n/4$

    $D \leftarrow D_S \leftarrow$ any decision tree built on tests from S closed under $\mathcal{F}$

    foreach $\mathcal{H}' \in \mathcal{H} - S$ **do**

        $D' \leftarrow$ AVERAGEDECISIONTREE($\mathcal{H}', \mathcal{T} - S, \mathcal{F} - S$)

        Attach D' to the leaf of D corresponding to $\mathcal{H}'$

    end

    return D

4. Set covering with constraints

4.1. Max-Density Precedence-Closed Subfamily (MDPCS)

The key to solce PCSC and PCMSSC is to solve the MDPCS problem. An approximation algorithm for MDPCS can be used as an essential subroutine in our algorithms for PCSC and PCMSSC. By **?**, the following greedy algorithm achieves an $O(\sqrt{m})$-approximation for MDPCS:

Algorithm 3: The greedy algorithm for MDPCS
procedure MDPCS-GREEDY$(\mathcal{G}, \prec, R)$
> $\mathcal{A} \leftarrow \mathcal{G}$
> **foreach** $S \in \mathcal{G}$ **do**
> > **if** $\Delta(P[S], R) > \Delta(\mathcal{A}, R)$ **then**
> > > $\mathcal{A} \leftarrow P[S]$
> >
> > **end**
>
> **end**
> **return** $\mathcal{A}$

Let $\delta = \max_{S \in \mathcal{G}} \Delta(P[S], R)$. When $\delta \geq 1$, then the approximation factor of the greedy can also be bounded by $O(\sqrt{n})$. We show that if we enforce a certain condition on the input called ϵ-*shallow ancestry*, then for $\epsilon < 1$ the greedy algorithm achieves an $O(n^\epsilon)$-approximation .

For each $S \in \mathcal{G}$, let $p(S) = |P[S]|$ and $c(S, R) = |\text{cov}(P[S], R)|$.

Theorem 16 *Suppose there exists a constant $C > 0$ and $\epsilon \in (0, 1)$ such that for all $S \in \mathcal{G}$, $p(S) \leq C \cdot c(S)^\epsilon$ (ϵ-shallow ancestry). Then MDPCS-Greedy provides an $O(n^\epsilon)$-approximation.*

Proof Let $\mathcal{A}^*$ be an optimal solution consisting of sets $S_1, \ldots, S_k$. There are two cases:

1. If $\delta \geq n^{1-\epsilon}$, then, we observe that $\Delta(\mathcal{A}, R) = \delta \geq n^{1-\epsilon}$. Since we can cover at most n elements with at least one set, we have $\Delta(\mathcal{A}^*, R) \leq n$. Therefore:
$$\frac{\Delta(\mathcal{A}^*, R)}{\Delta(\mathcal{A}, R)} \leq \frac{n}{n^{1-\epsilon}} = n^\epsilon$$

2. Else, if $\delta \leq n^{1-\epsilon}$, we proceed as follows: By definition of density, for any $S \in \mathcal{G}$, $c(S) \leq \delta \cdot p(S)$. Combining this with the ϵ-shallow ancestry condition, we have that for all $S \in \mathcal{G}$, $c(S) \leq \delta \cdot C \cdot c(S)^\epsilon$. Rearraning this inequality, we get that $c(S) \leq (\delta \cdot C)^{\frac{1}{1-\epsilon}}$. We have that:
$$|\text{cov}(\mathcal{A}^*)| = \left| \bigcup_{j=1}^{k} \text{cov}(S_j, R) \right| \leq \sum_{j=1}^{k} c(S_j) \leq k \cdot (\delta \cdot C)^{\frac{1}{1-\epsilon}}$$

   Therefore:
$$\Delta(\mathcal{A}^*, R) = \frac{|\text{cov}(\mathcal{A}^*, R)|}{k} \leq (\delta \cdot C)^{\frac{1}{1-\epsilon}}$$

   By the greedy choice, $\Delta(\mathcal{A}, R) \geq \delta$ and by assumption $\delta \leq n^{1-\epsilon}$. Thus:
$$\frac{\Delta(\mathcal{A}^*, R)}{\Delta(\mathcal{A}, R)} \leq \frac{(\delta \cdot C)^{\frac{1}{1-\epsilon}}}{\delta} = C^{\frac{1}{1-\epsilon}} \cdot \delta^{\frac{\epsilon}{1-\epsilon}} \leq C^{\frac{1}{1-\epsilon}} \cdot (n^{1-\epsilon})^{\frac{\epsilon}{1-\epsilon}} = C^{\frac{1}{1-\epsilon}} \cdot n^\epsilon$$

Since C is constant, the theorem follows. ∎

4.2. Precedence constrained set cover

We show the following:

Theorem 17 *If there exists an γ approximation algorithm for the MDPCS problem, then there exists an $(H_K + 1) \cdot \gamma$ - approximate algorithm for the PCSC problem.*

Proof ∎

Algorithm 4: The γ-greedy algorithm for PCSC

procedure PCSC$(\mathcal{U}, \mathcal{S}, \mathcal{F}, K)$

$\quad \mathcal{C} \leftarrow \emptyset$

$\quad$ **while** $|\mathrm{cov}(\mathcal{C}, \mathcal{U})| < K$ **do**

$\quad\quad \mathcal{A} \leftarrow$ Run the γ-approx. algorithm for MDPCS on $(\mathcal{U} - \mathcal{C}, \mathcal{S} - \mathcal{C}, \mathcal{F} - \mathcal{C}, m)$

$\quad\quad$ **if** $|\mathrm{cov}(\mathcal{C} \cup \mathcal{A}, \mathcal{U})| \geq K$ **then**

$\quad\quad\quad$ Find the minimum budget $B \in [|\mathcal{A}|]$, such that the γ-approx. algorithm for MDPCS on $(\mathcal{U} - \mathcal{C}, \mathcal{S} - \mathcal{C}, \mathcal{F} - \mathcal{C}, B)$ returns a set $\mathcal{B}$ with $|\mathrm{cov}(\mathcal{B}, \mathcal{U} - \mathcal{C})| \geq \frac{K - \mathrm{cov}(\mathcal{C}, \mathcal{U})}{\alpha}$

$\quad\quad\quad$ **while** $|\mathrm{cov}(\mathcal{C}, \mathcal{U})| < K$ **do**

$\quad\quad\quad\quad \mathcal{B} \leftarrow$ Run the γ-approx. algorithm for MDPCS on $(\mathcal{U} - \mathcal{C}, \mathcal{S} - \mathcal{C}, \mathcal{F} - \mathcal{C}, B)$

$\quad\quad\quad\quad \mathcal{C} \leftarrow \mathcal{C} \cup \mathcal{B}$

$\quad\quad\quad$ **end**

$\quad\quad\quad$ **return** $\mathcal{C}$

$\quad\quad$ **end**

$\quad\quad$ **foreach** $u \in \mathrm{cov}(\mathcal{A}, \mathcal{U} - \mathcal{C})$ **do**

$\quad\quad\quad c(u) \leftarrow \Delta(\mathcal{A}, \mathcal{U} - \mathcal{C})$

$\quad\quad$ **end**

$\quad\quad \mathcal{C} \leftarrow \mathcal{C} \cup \mathcal{A}$

$\quad$ **end**

Theorem 18 *If the precedence constraints form an outforest, then there exists an bicriteria $(4, O(\log n))$-approximation algorithm for PCSC which be converted to an $O(\log^2 n)$ approximation algorithm.*

4.3. Precedence constrained min sum set cover

By **?**:

Theorem 19 *If there exists an γ approximation algorithm for the MDPCS problem, then there exists an $4 \cdot \gamma$ - approximate algorithm for the PCMSSC problem.*

Theorem 20 *If the precedence constraints form an outforest, then there exists an $(O(\log n))$-approximation algorithm for PCMSSC.*

5. Hardness

Theorem 21 *PCWCAL with outforest precedence constraints is NP-hard to approximate within a factor of $O\left(\log^2 n\right)$ unless P = NP.*

6. Conclusions and Future Work

Appendix A. My Proof of Theorem 1

This is a boring technical proof.

Appendix B. My Proof of Theorem 2

This is a complete version of a proof sketched in the main text.