

Provable Data-driven Projection Method for Quadratic Programming

Anh Tuan Nguyen^{1*}, Viet Anh Nguyen²

¹Carnegie Mellon University

²The Chinese University of Hong Kong
atnguyen@cs.cmu.edu, nguyenv@se.cuhk.edu.hk

Abstract

Projection methods aim to reduce the dimensionality of the optimization instance, thereby improving the scalability of high-dimensional problems. Recently, Sakaue and Oki (2024) proposed a data-driven approach for linear programs (LPs), where the projection matrix is learned from observed problem instances drawn from an application-specific distribution of problems. We analyze the generalization guarantee for the data-driven projection matrix learning for convex quadratic programs (QPs). Unlike in LPs, the optimal solutions of convex QPs are not confined to the vertices of the feasible polyhedron, and this complicates the analysis of the optimal value function. To overcome this challenge, we demonstrate that the solutions of convex QPs can be localized within a feasible region corresponding to a special active set, utilizing Caratheodory’s theorem. Building on such observation, we propose the unrolled active set method, which models the computation of the optimal value as a Goldberg-Jerrum algorithm with bounded complexities, thereby establishing learning guarantees. We then extend our analysis further to other settings, including learning to match the optimal solution and an input-aware setting, where we learn to map QP problem instances to projection matrices.

Extended version — <https://arxiv.org/pdf/2509.04524>

1 Introduction

Linear programs (LPs) and the more general quadratic programs (QPs) are simple forms of convex optimization problems, yet they play crucial roles in many industrial (Gass 2003; Dostál 2009) and scientific domains (Amos 2022). Practical LP and QP instances are often computationally intensive to solve due to their enormous problem size, which can encompass millions of variables and constraints. As a result, accelerating solving approaches for large-scale LPs and QPs are important directions in the operations research literature. Two of the most prominent approaches include accelerated solvers and dimensionality reduction methods. Accelerated solvers focus on improving the speed of widely used solvers on large-scale problems via parallelization, randomization, or wisely leveraging cheap first-order (i.e., gradient)

information, among other approaches. Some of the recent advances include parallelized simplex methods (Huangfu and Hall 2018), randomized interior-point methods (Chowdhury et al. 2022), and primal-dual hybrid gradient methods (Applegate et al. 2021).

Another complementary, solver-agnostic approach for large-scale LPs and QPs is the dimensionality reduction technique, which generalizes the idea of reducing the size of problem instances while preserving the properties of the objective values and variables. A promising approach is through random projections (d’Ambrosio et al. 2020; Vu et al. 2019; Vu, Poirion, and Liberti 2018), where a random projection matrix is used to map the variables and feasible regions of the original problem instances into a low-dimensional space, forming projected problem instances that can be solved much faster. The solutions of projected problem instances can then be mapped back to the original space, in the hope that their quality is sufficiently comparable to that of the optimal solution of the original problem instances. Importantly, this solver-agnostic approach can be combined with accelerated solvers to further improve the solving of large-scale LPs and QPs.

However, random projection matrices overlook the geometric properties of the problem instances, and this oversight potentially leads to inferior solution quality of the projected problem instances compared to that of the original problem instances. Recently, Sakaue and Oki (2024) proposed a data-driven approach for learning the projection matrix, specifically targeting LPs. Assume that there are not one, but multiple LPs $\pi_{\text{LP}} = (\mathbf{c}, \mathbf{A}, \mathbf{b}) \in \Pi_{\text{LP}} \subset \mathbb{R}^n \times \mathbb{R}^{n \times m} \times \mathbb{R}^m$ that have to be solved in the form

$$\text{OPT}(\pi_{\text{LP}}) = \min_{\mathbf{x} \in \mathbb{R}^n} \{\mathbf{c}^\top \mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}.$$

The parameters π_{LP} are drawn from some application-specific and potentially unknown problem distribution \mathcal{D}_{LP} over Π_{LP} . Sakaue and Oki (2024) proposed to learn the projection matrix $\mathbf{P} \in \mathcal{P} \subset \mathbb{R}^{n \times k}$, where $k \ll n$ is the dimensionality of the projection space, by minimizing the expected optimal objective of the projected LPs $\mathbb{E}_{\pi_{\text{LP}} \sim \mathcal{D}_{\text{LP}}}[\ell_{\text{LP}}(\mathbf{P}, \pi_{\text{LP}})]$, where

$$\ell_{\text{LP}}(\mathbf{P}, \pi_{\text{LP}}) = \min_{\mathbf{y} \in \mathbb{R}^k} \{\mathbf{c}^\top \mathbf{P}\mathbf{y} \mid \mathbf{A}\mathbf{P}\mathbf{y} \leq \mathbf{b}\}$$

is the optimal objective value of the projected LP. Because \mathcal{D} is unknown, minimizing $\mathbb{E}_{\pi_{\text{LP}} \sim \mathcal{D}_{\text{LP}}}[\ell_{\text{LP}}(\mathbf{P}, \pi_{\text{LP}})]$

*Work done while visiting TTIC.

is intractable, and instead we learn P through *empirical risk minimization (ERM)* using LP problem instances drawn from \mathcal{D}_{LP} . It is easy to see that $\ell_{LP}(P, \pi_{LP})$ upper-bounds $\text{OPT}(\pi_{LP})$, and therefore the smaller $\mathbb{E}_{\pi_{LP} \sim \mathcal{D}_{LP}}[\ell_{LP}(P, \pi_{LP})]$ is, the closer the quality of solutions of projected problem instances to that of original problem instances. Along with promising empirical results, Sakaue and Oki (2024) provided generalization guarantees for learning P via ERM by analyzing the learning-theoretic complexity (i.e., pseudo-dimension (Pollard 1984)) of the corresponding loss function class $\mathcal{L}_{LP} = \{\ell_P : \Pi_{LP} \rightarrow [-H, 0] \mid P \in \mathcal{P}\}$, where $\ell_P(\pi_{LP}) := \ell_{LP}(P, \pi_{LP})$, and H is some real-valued upper-bound for the function class.

Inspired by this success, a natural direction is to extend this framework to convex QPs. Similarly, given QP problem instances $\pi = (Q, c, A, b) \in \Pi \subset \mathbb{R}^{n \times n} \times \mathbb{R}^n \times \mathbb{R}^{m \times n} \times \mathbb{R}^n$ coming from an application-specific, unknown problem distribution \mathcal{D} over Π , the idea is to learn a projection matrix $P \in \mathcal{P} \subset \mathbb{R}^{n \times k}$ with $k \ll n$ that achieves small population loss $\mathbb{E}_{\pi \sim \mathcal{D}}[\ell(P, \pi)]$ via ERM, where

$$\ell(P, \pi) = \min_{y \in \mathbb{R}^k} \left\{ \frac{1}{2} y^\top P^\top Q P y + c^\top P y \mid A P y \leq b \right\}.$$

Again, to ensure the generalization guarantee for P learned via ERM, we need to analyze the function class $\mathcal{L} = \{\ell_P : \Pi \rightarrow [-H, 0] \mid P \in \mathcal{P}\}$, where $\ell_P(\pi) := \ell(P, \pi)$.

At first glance, the extension to QPs may seem straightforward because the previous ideas seem readily applicable to the form of QPs, and the gradient update can also be derived using the envelope theorem. However, the optimal solutions of QPs exhibit fundamentally different geometrical structures, and it turns out that extending the existing theoretical framework to QPs requires developing new tools tailored to these specific problems.

Contributions. We formalize the data-driven projection method for convex QPs and analyze the generalization guarantees of learning the projection matrix. Our contributions can be summarized as follows:

1. We establish generalization guarantees for the data-driven learning projection matrix for QPs in Theorem 5.7. Our new result is more general and strictly tighter than the previous bound proposed by Sakaue and Oki (2024), which applies only to LPs. For completeness, we also instantiate a lower bound for the convex QPs case in Proposition 5.8.
2. We propose and analyze a novel learning scenario, where the goal is to match the optimal solution in Section 6.1. This setting is particularly useful in practical applications where the focus is on the solution to be implemented. The guarantee result is presented in Theorem 6.1.
3. We consider input-aware settings, where we learn a neural network that maps a convex QP to a tailored projection matrix, as described in Section 6.2. The guarantee result is presented in Theorem 6.2.

Technical challenges and overviews. For any LP with parameters $\pi_{LP} = (c, A, b)$ and any projection matrix P , the solution of the projected LP is one of the vertices of the

feasible polyhedron. Leveraging such observation, Sakaue and Oki (2024) describes the computation of the projected LP’s optimal value $\ell_{LP}(P, \pi_{LP})$ by enumerating all potential vertices, and identifies the vertex y^* that yields the lowest objective $c^\top P y^*$. The computation of $\ell_{LP}(P, \pi_{LP})$ can then be described by a bounded number of distinct conditional statements involving polynomials in the entries of P ; see Section 3.2 for details.

This favorable property, however, does not extend to QPs, as the solution of QPs can be anywhere within the feasible polyhedron, not just at its vertices. This makes directly locating the solution and calculating the optimal objective $\ell(P, \pi)$ very challenging. To overcome this issue, we propose a four-step analytical approach. First, we will construct a perturbed objective $\ell_{P, \gamma}(\pi)$ that is well-behaved and can approximate $\ell(P, \pi)$ with arbitrarily precision (Lemma 5.1, Proposition 5.2). Second, we leverage the structure of this perturbed problem to develop the *unrolled active set method*, an algorithm that exactly computes its optimal value (Lemma 5.4). Third, we demonstrate that our method can be framed as a GJ algorithm with bounded complexities (Lemma 5.5), which enables us to bound the pseudo-dimension of the perturbed function class. Finally, by relating the perturbed objective to the original, we extend this bound to the original QP loss function, thereby proving our main generalization guarantee (Theorem 5.7).

2 Related Works

Projection methods for LPs and QPs. Projection methods aim to accelerate the solution of LPs and QPs by reducing the size of the problem instances. Previous works have investigated random projection to reduce the number of constraints (Vu et al. 2019; Poirion, Lourenco, and Takeda 2023) and variables (Akchen and Misis 2025). Recently, Sakaue and Oki (2024) and Iwata and Sakaue (2025) considered a data-driven approach, learning the projection matrix for a specific distribution of problems instead of a random projection, specifically targeting LPs. Our paper extends this framework to convex QPs.

Learning to optimize. Learning to optimize leverages machine learning to develop optimization methods, i.e., by predicting an initial solution for the exact algorithm, approximating the exact solution directly, or adapting specific components of optimization algorithms (Chen et al. 2022; Amos et al. 2023; Bengio, Lodi, and Prouvost 2021). Learning to project for LPs (Sakaue and Oki 2024; Iwata and Sakaue 2025) and convex QPs (this work) belongs to this broad category, where the learned projection matrices are used to accelerate off-the-shelf solvers and produce approximate solutions that are guaranteed to be feasible, unlike prior methods that approximate optimal solutions directly using neural networks.

Data-driven algorithm design. Data-driven algorithm design (Balcan 2020; Gupta and Roughgarden 2020) is an emerging algorithm design paradigm that proposes adapting algorithms by configuring their hyperparameters or internal components to the specific set of problem instances they

must solve, rather than considering the worst-case problem instances. Assuming that there is an application-specific, potentially unknown problem distribution from which the problem instances are drawn, data-driven algorithm design aims to maximize its empirical performance using the observed problem instances, with the hope that the adapted algorithm will perform well on future problem instances drawn from the same problem distribution. Data-driven algorithm design is an active research direction in both empirical validation and theoretical analysis across various domains, including sketching and low-rank approximation (Indyk, Vakilian, and Yuan 2019; Bartlett, Indyk, and Wagner 2022; Li et al. 2023), (mixed) integer linear programming (Balcan et al. 2018; Li et al. 2023), tuning regularization hyperparameters (Balcan et al. 2022; Balcan, Nguyen, and Sharma 2023), and other general frameworks for theoretical analysis in data-driven settings Bartlett, Indyk, and Wagner (2022); Balcan, Nguyen, and Sharma (2025a,b). Data-driven projection methods for LPs (Sakaue and Oki 2024) and QPs are specific instances of data-driven algorithm design.

3 Backgrounds on Learning Theory

3.1 Pseudo-dimension

We recall the notion of *pseudo-dimension*, the primary learning-theoretic complexity measure of this work.

Definition 1 (Pseudo-dimension, Pollard 1984). *Consider a real-valued function class \mathcal{L} , of which each function ℓ takes input π in Π and output $\ell(\pi) \in [-H, 0]$. Given a set of inputs $S = (\pi_1, \dots, \pi_N) \subset \Pi$, we say that S is shattered by \mathcal{L} if there exists a set of real-valued threshold $r_1, \dots, r_N \in \mathbb{R}$ such that $|\{\text{sign}(\ell(\pi_1) - r_1), \dots, \text{sign}(\ell(\pi_N) - r_N) \mid \ell \in \mathcal{L}\}| = 2^N$. The pseudo-dimension of \mathcal{L} , denoted as $\text{Pdim}(\mathcal{L})$, is the maximum size N of a input set that \mathcal{L} can shatter.*

It is widely known from the learning theory literature that if a real-valued function class has bounded pseudo-dimension, then it is PAC-learnable with ERM.

Theorem 3.1 (Pollard 1984). *Consider a real-valued function class \mathcal{F} , of which each function \mathcal{L} takes input π in Π and output $\ell(\pi) \in [-H, 0]$. Assume that $\text{Pdim}(\mathcal{L})$ is finite. Then given $\epsilon > 0$ and $\delta \in (0, 1)$, for any $N \geq M(\delta, \epsilon)$, where $M(\delta, \epsilon) = \mathcal{O}\left(\frac{H^2}{\epsilon^2}(\text{Pdim}(\mathcal{L}) + \log(1/\delta))\right)$, with probability at least $1 - \delta$ over the draw of $S = (\pi_1, \dots, \pi_N) \sim \mathcal{D}^N$, where \mathcal{D} is a distribution over Π , we have*

$$\mathbb{E}_{\pi \sim \mathcal{D}}[\hat{\ell}_S(\pi)] \leq \inf_{\ell \in \mathcal{L}} \mathbb{E}_{\pi \sim \mathcal{D}}[\ell(\pi)] + \epsilon.$$

Here, $\hat{\ell}_S \in \arg \min_{\ell \in \mathcal{L}} \frac{1}{N} \sum_{i=1}^N \ell(\pi_i)$ is the ERM minimizer.

3.2 Goldberg-Jerrum Framework

Goldberg-Jerrum (GJ) framework is originally proposed by Goldberg and Jerrum (1993), and is later refined by Bartlett, Indyk, and Wagner (2022). It is a convenient framework to establish a pseudo-dimension upper-bound for parameterized function classes, of which the computation can be described by a *GJ algorithm* using conditional statements, intermediate values, and outputs involving rational functions

of their parameters. The formal definition of the GJ algorithm can be described as follows.

Definition 2 (GJ algorithm, (Bartlett, Indyk, and Wagner 2022)). *A **GJ algorithm** Γ operates on real-valued inputs, and can perform two types of operations:*

- *Arithmetic operators of the form $v'' = v \odot v'$, where $\odot \in \{+, -, \times, \div\}$, and*
- *Conditional statements of the form “if $v \geq 0 \dots$ else \dots ”.*

In both cases, v and v' are either inputs or values previously computed by the algorithm.

The immediate values v, v', v'' computed by the GJ algorithm are rational functions (fractions of two polynomials) of its parameters. The complexities of the GJ algorithm are measured by the highest degree of rational functions it computes and the number of distinct rational functions that appear in the conditional statements. The formal definition of its complexities is as follows.

Definition 3 (Complexities of GJ algorithm, (Bartlett, Indyk, and Wagner 2022)). *The **degree** of a GJ algorithm is the maximum degree of any rational function that it computes of the inputs. The **predicate complexity** of a GJ algorithm is the number of distinct rational functions that appear in its conditional statements. Here, the degree of rational function $f(\mathbf{x}) = \frac{g(\mathbf{x})}{h(\mathbf{x})}$, where g and h are two polynomials in \mathbf{x} , is $\deg(f) = \max\{\deg(g), \deg(h)\}$.*

The following theorem asserts that if any function class for which the function’s computation can be described by a GJ algorithm with bounded degree and predicate complexities, then the pseudo-dimension of that function class is also bounded.

Theorem 3.2 (Bartlett, Indyk, and Wagner (2022, theorem 3.3)). *Suppose that each function $\ell_{\mathbf{P}} \in \mathcal{L}$ is specified by n real parameters $\mathbf{P} \in \mathbb{R}^n$. Suppose that for every $\pi \in \Pi$ and $r \in \mathbb{R}$, there is a GJ algorithm $\Gamma_{\pi, r}$ that, given $\ell_{\mathbf{P}} \in \mathcal{L}$, returns “true” if $\ell_{\mathbf{P}}(\pi) \geq r$ and “false” otherwise. Assume that $\Gamma_{\pi, r}$ has degree Δ and predicate complexity Λ . Then, $\text{Pdim}(\mathcal{L}) = \mathcal{O}(n \log(\Delta \Lambda))$.*

Note that the GJ algorithm $\Gamma_{\pi, r}$ described above corresponds to each fixed input π and threshold value r . The input of the GJ algorithm $\Gamma_{\pi, r}$ is the hyperparameters \mathbf{P} (the projection matrix in our case) parameterizing $\ell_{\mathbf{P}}$, and the intermediate values and conditional statements involve in rational functions of \mathbf{P} . Moreover, the GJ framework only serves as a tool for analyzing *learning-theoretic complexity* (e.g., pseudo-dimension) of the parameterized function class \mathcal{L} , and does not describe how the function $\ell_{\mathbf{P}}(\pi)$ is computed in practice. In our framework, the computation of $\ell_{\mathbf{P}}(\pi)$ utilizes our proposed *unrolled active set method* in Algorithm 1, which we show to be a GJ algorithm with bounded complexities in Lemma 5.5. In practice, it might be computed using the *active set method* (Nocedal and Wright 2006) or interior-point method (Dikin 1967) for computational efficiency; however, these methods cannot be cast as GJ algorithms.

4 Problem Settings

This section formalizes the problem of learning the projection matrix for QPs in the data-driven setting.

4.1 Original QPs and Projected QPs

Consider the *original QPs* (OQPs) $\pi = (\mathbf{Q}, \mathbf{c}, \mathbf{A}, \mathbf{b}) \in \Pi \subset \mathbb{R}^{n \times n} \times \mathbb{R}^n \times \mathbb{R}^{m \times n} \times \mathbb{R}^m$ with inequality constraints:

$$\text{OPT}(\pi) = \min_{\mathbf{x} \in \mathbb{R}^n} \left\{ \frac{1}{2} \mathbf{x}^\top \mathbf{Q} \mathbf{x} + \mathbf{c}^\top \mathbf{x} \mid \mathbf{A} \mathbf{x} \leq \mathbf{b} \right\}, \quad (\text{OQP})$$

where \mathbf{Q} is a positive semi-definite (PSD) matrix, while n and m are the number of variables and constraints, respectively. Here, we assume that the variable size n and the number of constraints m are large, and solving the OQP is a computationally expensive task. The core idea of the projection method evolves around a *full column-rank* projection matrix $\mathbf{P} \in \mathcal{P} \subset \mathbb{R}^{n \times k}$, where $k \ll n$ is the projection dimension. Setting $\mathbf{x} = \mathbf{P} \mathbf{y}$, we obtain the *projected QPs* (PQPs) corresponding to the OQPs π and the projection matrix \mathbf{P}

$$\ell(\mathbf{P}, \pi) = \min_{\mathbf{y} \in \mathbb{R}^k} \left\{ \frac{1}{2} \mathbf{y}^\top \mathbf{P}^\top \mathbf{Q} \mathbf{P} \mathbf{y} + \mathbf{c}^\top \mathbf{P} \mathbf{y} \mid \mathbf{A} \mathbf{P} \mathbf{y} \leq \mathbf{b} \right\}. \quad (\text{PQP})$$

Similar to prior works (Sakaue and Oki 2023; Vu et al. 2019), we make the following assumptions for the OQPs.

Assumption 1 (Regularity conditions). The OQPs:

- (1) take inequality-constrained form as (OQP),
- (2) have $\mathbf{0}_n \in \mathbb{R}^n$ as a feasible point,
- (3) have the feasible region is bounded by R in the sense that $\|\mathbf{x}\|_2 \leq R$ for any feasible \mathbf{x} , and
- (4) have bounded optimal objective value from $[-H, 0]$, for some constant positive H .

Remark 1. As discussed in previous work (Sakaue and Oki 2024; Vu et al. 2019), Assumption 1 is not restrictive. First, any QPs that also have equality assumptions can also be converted into the inequality form (Assumption 1.1 by considering the null space of the equality constraints (see Appendix C, (Sakaue and Oki 2024) for details). For Assumption 1.2, one can instead assume that there exists a feasible point \mathbf{x}_0 , and linearly translate the feasible region so that \mathbf{x}_0 coincides with $\mathbf{0}_n$, without changing the form of QPs. Assumption 1.3 is standard from the optimization literature (Vu et al. 2019). Lastly, Assumption 1.4 is simply a consequence of Assumption 1.1 and 1.2.

Under Assumption 1, the PQPs also have a favorable structure, which can be formalized as follows.

Proposition 4.1. *Under Assumption 1, then for any OQP π and projection matrix \mathbf{P} , (i) the corresponding PQP has $\mathbf{0}_k$ as a feasible point, and (ii) $\ell(\mathbf{P}, \pi)$ is lower bounded by $\text{OPT}(\pi)$, and therefore takes a value between $[-H, 0]$.*

Proof. Since $\mathbf{0}_n$ is a feasible point of OQP π , $\mathbf{y} = \mathbf{0}_k$ satisfies $\mathbf{A} \mathbf{P} \mathbf{y} \leq \mathbf{b}$, meaning that $\mathbf{0}_k$ is a feasible point of PQP. Moreover, let \mathbf{y}^* be an optimal solution of PQP, then $\mathbf{x}^* = \mathbf{P} \mathbf{y}^*$ is a feasible point of OQP, thus $\text{OPT}(\pi) \leq \ell(\mathbf{P}, \pi)$. \square

4.2 Data-driven Learning of the Projection Matrix

In the data-driven setting, we assume that there is an application-specific and potentially unknown problem distribution \mathcal{D} over the set of QPs Π . The optimal projection matrix \mathbf{P} minimizes the population PQPs' optimal objective value

$$\mathbf{P}_{\mathcal{D}}^* \in \arg \min_{\mathbf{P} \in \mathcal{P}} \mathbb{E}_{\pi \sim \mathcal{D}} [\ell(\mathbf{P}, \pi)].$$

From Proposition 4.1, we know that the smaller $\mathbb{E}_{\pi \sim \mathcal{D}} [\ell(\mathbf{P}, \pi)]$, the closer the optimal objective value of PQP $\ell(\mathbf{P}, \pi)$ to $\text{OPT}(\pi)$, and the better \mathbf{P} . However, since \mathcal{D} is unknown, we instead learn \mathbf{P} using the observed PQPs $S = \{\pi_1, \dots, \pi_N\}$ drawn i.i.d. from \mathcal{D} via ERM

$$\hat{\mathbf{P}}_S \in \arg \min_{\mathbf{P} \in \mathcal{P}} \frac{1}{N} \sum_{i=1}^N \ell(\mathbf{P}, \pi_i).$$

Object of study. We aim to answer the standard generalization guarantee question: given a tolerance $\epsilon > 0$ and a failure probability $\delta \in (0, 1)$, what is the sample complexity $M(\epsilon, \delta)$ such that with probability at least $1 - \delta$ over the draw of problem instances $S = \{\pi_1, \dots, \pi_N\}$, where $N \geq M(\epsilon, \delta)$, we have $\mathbb{E}_{\pi \sim \mathcal{D}} [\ell(\hat{\mathbf{P}}_S, \pi)] \leq \mathbb{E}_{\pi \sim \mathcal{D}} [\ell(\mathbf{P}_{\mathcal{D}}^*, \pi)] + \epsilon$. Consider the function class $\mathcal{L} = \{\ell_{\mathbf{P}} : \Pi \rightarrow [-H, 0] \mid \mathbf{P} \in \mathcal{P}\}$, where $\ell_{\mathbf{P}}(\pi) = \ell(\mathbf{P}, \pi)$. Theorem 3.1 suggests that the generalization guarantee is achievable by bounding the pseudo-dimension of \mathcal{L} .

5 Generalization Guarantee for Data-driven Input-Agnostic Projection Method for QPs

This section provides the generalization guarantee for data-driven learning of the projection matrix \mathbf{P} for QPs.

5.1 Regularizing via Perturbing OQPs and the Perturbed Function Class

There are two main obstacles to analyzing the generalization guarantee for data-driven learning the projection matrix in QPs. First, the optimal solution of QPs can lie arbitrarily anywhere in the feasible polyhedron. Second, when the matrix \mathbf{Q} is singular, there can be infinitely many optimal solutions. To address this issue, we first introduce the perturbed function class \mathcal{L}_γ , constructed by adding Tikhonov's regularization to the objective function of OQPs. This perturbation is equivalent to perturbing the matrix \mathbf{Q} of the original input. After the perturbation: (i) the objective function of the perturbed OQPs and any perturbed PQPs becomes strictly convex, which favorably helps us localize the unique optimal solution and constructing the unrolled active set method; and (2) the perturbed function class \mathcal{L}_γ can approximate \mathcal{L} with arbitrary precision, and therefore analyzing the \mathcal{L}_γ can recover the guarantee for \mathcal{L} .

Lemma 5.1. *Given a OQP $\pi = (\mathbf{Q}, \mathbf{c}, \mathbf{A}, \mathbf{b})$, then there exists $\gamma > 0$ that is independent on \mathbf{P} , such that for the perturbed OQP $\pi_\gamma = (\mathbf{Q}_\gamma, \mathbf{c}, \mathbf{A}, \mathbf{b})$, where $\mathbf{Q}_\gamma = \mathbf{Q} + \gamma \mathbf{I}_n$ and for any projection matrix $\mathbf{P} \in \mathcal{P}$, we have*

$$0 \leq \ell(\mathbf{P}, \pi_\gamma) - \ell(\mathbf{P}, \pi) \leq \frac{\gamma R^2}{2},$$

where R comes from Assumption 1.3.

Proof. Let $\mathbf{y}^*(\mathbf{P})$ is an optimal solution of the PQP, that is, $\ell(\mathbf{P}, \boldsymbol{\pi}) = \frac{1}{2} \mathbf{y}^*(\mathbf{P})^\top \mathbf{P}^\top \mathbf{Q} \mathbf{P} \mathbf{y}^*(\mathbf{P}) + \mathbf{c}^\top \mathbf{P} \mathbf{y}^*(\mathbf{P})$, and let

- (i) $f_{\mathbf{P}}(\mathbf{y}) = \frac{1}{2} \mathbf{y}^\top \mathbf{P}^\top \mathbf{Q} \mathbf{P} \mathbf{y} + \mathbf{c}^\top \mathbf{P} \mathbf{y}$ be the objective function of the PQP, and
- (ii) $f_{\mathbf{P}, \gamma}(\mathbf{y}) = \frac{1}{2} \mathbf{y}^\top \mathbf{P}^\top (\mathbf{Q} + \gamma \mathbf{I}_n) \mathbf{P} \mathbf{y} + \mathbf{c}^\top \mathbf{P} \mathbf{y} = f_{\mathbf{P}}(\mathbf{y}) + \frac{\gamma}{2} \|\mathbf{P} \mathbf{y}\|_2^2$ be the objective function of the perturbed PQP.

Then $f_{\mathbf{P}}(\mathbf{y}^*(\mathbf{P})) = \ell(\mathbf{P}, \boldsymbol{\pi})$ by definition, and note that $\mathbf{y}^*(\mathbf{P})$ is a feasible point of the perturbed PQP $\boldsymbol{\pi}_\gamma$, meaning that $f_{\mathbf{P}, \gamma}(\mathbf{y}^*(\mathbf{P})) \geq \ell(\mathbf{P}, \boldsymbol{\pi}_\gamma)$. Besides, PQP and perturbed PQP have the same feasible region, with the objective of PQP $f_{\mathbf{P}}(\mathbf{y})$ is smaller than that of perturbed PQP $f_{\mathbf{P}, \gamma}(\mathbf{y})$, meaning that $\ell(\mathbf{P}, \boldsymbol{\pi}_\gamma) \geq \ell(\mathbf{P}, \boldsymbol{\pi})$. Combining the facts above, we have

$$\begin{aligned} 0 &\leq \ell(\mathbf{P}, \boldsymbol{\pi}_\gamma) - \ell(\mathbf{P}, \boldsymbol{\pi}) \leq f_{\mathbf{P}, \gamma}(\mathbf{y}^*(\mathbf{P})) - f_{\mathbf{P}}(\mathbf{y}^*(\mathbf{P})) \\ &= \frac{\gamma}{2} \|\mathbf{P} \mathbf{y}^*(\mathbf{P})\|_2^2 \leq \frac{\gamma R^2}{2}, \end{aligned}$$

where the final inequality comes from the fact that $\mathbf{P} \mathbf{y}^*(\mathbf{P})$ is a feasible point of OQP, and the feasible region is bounded by R by Assumption 1.3. \square

Proposition 5.2. *Any perturbed PQP of a perturbed OQP $\boldsymbol{\pi}_\gamma$ under a projection matrix \mathbf{P} has a unique optimal solution.*

Proof. First, notice that the matrix $\mathbf{P}^\top \mathbf{Q}_\gamma \mathbf{P}$ is positive definite. To see that, for any $\mathbf{y} \in \mathbb{R}^k$ and $\mathbf{y} \neq \mathbf{0}$, we have $\mathbf{P} \mathbf{y} \neq \mathbf{0}_n$ since \mathbf{P} is a full-column rank matrix. Therefore $\mathbf{y}^\top \mathbf{P}^\top \mathbf{Q}_\gamma \mathbf{P} \mathbf{y} = (\mathbf{P} \mathbf{y})^\top \mathbf{Q}_\gamma (\mathbf{P} \mathbf{y}) > 0$ as $\mathbf{Q}_\gamma = \mathbf{Q} + \gamma \mathbf{I}_n$ is a positive definite matrix. This implies that the objective value of the perturbed PQP $\boldsymbol{\pi}_\gamma$ is also strictly convex. Moreover, the perturbed PQP is feasible (admitting $\mathbf{0}_k$ as a feasible point by Proposition 4.1) and bounded as $-H \leq \text{OPT}(\boldsymbol{\pi}) \leq \ell(\mathbf{P}, \boldsymbol{\pi}) \leq \ell(\mathbf{P}, \boldsymbol{\pi}_\gamma)$. Therefore, the perturbed PQP has a unique optimal solution. \square

We now formally define the perturbed function class \mathcal{L}_γ .

Definition 4 (Perturbed function class). *Given $\gamma > 0$, the perturbed function class \mathcal{L}_γ is defined as*

$$\mathcal{L}_\gamma = \{\ell_{\mathbf{P}, \gamma} : \Pi \rightarrow [-H, 0] \mid \mathbf{P} \in \mathcal{P}\},$$

where $\ell_{\mathbf{P}, \gamma}(\boldsymbol{\pi}) = \ell(\mathbf{P}, \boldsymbol{\pi}_\gamma)$, $\boldsymbol{\pi}_\gamma = (\mathbf{Q} + \gamma \mathbf{I}_n, \mathbf{c}, \mathbf{A}, \mathbf{b})$, and $\ell(\mathbf{P}, \boldsymbol{\pi}_\gamma)$ is the optimal objective value of the perturbed PQP corresponding to the perturbed OQP $\boldsymbol{\pi}_\gamma$ and projection matrix \mathbf{P} .

Next, we analyze the pseudo-dimension of \mathcal{L}_γ , and later use the bound on the pseudo-dimension of \mathcal{L}_γ to bound the pseudo-dimension of \mathcal{L} . Toward this goal, we need to introduce several auxiliary results: In Section 5.2, we simplify the representation of the optimal solution using a localization technique, and in Section 5.3, we devise a GJ algorithm to compute the optimal value of the perturbed PQP. These two results are the foundation for the bounding the pseudo-dimension in Section 5.4.

5.2 Localizing the Solution of Perturbed PQPs

Consider the perturbed PQP corresponding to a projection matrix \mathbf{P} and the perturbed OQP $\boldsymbol{\pi}_\gamma = (\mathbf{Q}_\gamma, \mathbf{c}, \mathbf{A}, \mathbf{b})$, and for convenience, let $\tilde{\mathbf{Q}} = \mathbf{P}^\top \mathbf{Q}_\gamma \mathbf{P}$, $\tilde{\mathbf{c}} = \mathbf{P}^\top \mathbf{c}$, and $\tilde{\mathbf{A}} = \mathbf{A} \mathbf{P}$. We first prove that the solution of the perturbed PQP can be described using a simpler equality-constrained QP, of which the constraint matrix $\tilde{\mathbf{A}}_{\mathcal{B}}$, extracted from the constraint matrix $\tilde{\mathbf{A}}$ of the perturbed PQP, has linearly independent rows. This serves as a localizing scheme for the optimal solution of a perturbed PQP, and is the foundation for the unrolled active set method (Algorithm 1) that we describe later.

Lemma 5.3. *Let \mathbf{y}^* be the (unique) optimal solution of perturbed PQP with the corresponding active set $\mathcal{A}(\mathbf{y}^*) = \{i \in \{1, \dots, m\} \mid \tilde{\mathbf{A}}_i \mathbf{y}^* = \mathbf{b}_i\}$. Then there exists a subset $\mathcal{B} \subseteq \mathcal{A}(\mathbf{y}^*)$ such that:*

1. *The matrix $\tilde{\mathbf{A}}_{\mathcal{B}}$ has linearly independent row. Here $\mathbf{A}_{\mathcal{B}}$ is the matrix formed by the row i^{th} row of \mathbf{A} for $i \in \mathcal{B}$.*
2. *\mathbf{y}^* is the unique solution for the equality-constrained QP $\min_{\mathbf{y} \in \mathbb{R}^k} \left\{ \frac{1}{2} \mathbf{y}^\top \tilde{\mathbf{Q}} \mathbf{y} + \tilde{\mathbf{c}}^\top \mathbf{y} \mid \tilde{\mathbf{A}}_{\mathcal{B}} \mathbf{y} = \mathbf{b}_{\mathcal{B}} \right\}$.*

Proof sketch. The detailed proof can be found in the appendix. Using KKT conditions, we claim that $-(\tilde{\mathbf{Q}} \mathbf{y}^* + \tilde{\mathbf{c}}) = \sum_{i \in \mathcal{A}(\mathbf{y}^*)} \lambda_i^* \cdot \tilde{\mathbf{A}}_i$. Note that $\sum_{i \in \mathcal{A}(\mathbf{y}^*)} \lambda_i^* \cdot \tilde{\mathbf{A}}_i$ is a conic combination, and using Conic's Carathéodory theorem, we claim that there exists a subset $\mathcal{B} \subseteq \mathcal{A}(\mathbf{y}^*)$ such that there exists $\mu_i \geq 0$ for $i \in \mathcal{B}$ such that $-(\tilde{\mathbf{Q}} \mathbf{y}^* + \tilde{\mathbf{c}}) = \sum_{j \in \mathcal{B}} \mu_j \cdot \tilde{\mathbf{A}}_j \Leftrightarrow \tilde{\mathbf{Q}} \mathbf{y}^* + \tilde{\mathbf{c}} + \tilde{\mathbf{A}}_{\mathcal{B}}^\top \boldsymbol{\mu}_{\mathcal{B}} = \mathbf{0}$, and $\tilde{\mathbf{A}}_{\mathcal{B}}$ has linearly independent rows. Finally, we will show that \mathbf{y}^* is the unique solution of the equality-constrained QP $\min_{\mathbf{y} \in \mathbb{R}^k} \left\{ \frac{1}{2} \mathbf{y}^\top \tilde{\mathbf{Q}} \mathbf{y} + \tilde{\mathbf{c}}^\top \mathbf{y} \mid \tilde{\mathbf{A}}_{\mathcal{B}} \mathbf{y} = \mathbf{b}_{\mathcal{B}} \right\}$ by showing that $(\mathbf{y}^*, \boldsymbol{\mu}_{\mathcal{B}})$ is a KKT point of that problem. \square

Remark 2. In Lemma 5.3, since $\tilde{\mathbf{A}}_{\mathcal{B}}$ has linearly independent row and $\tilde{\mathbf{Q}}$ is positive definite, one can easily show that the KKT matrix $\mathbf{K} = \begin{bmatrix} \tilde{\mathbf{Q}} & \tilde{\mathbf{A}}_{\mathcal{B}}^\top \\ \tilde{\mathbf{A}}_{\mathcal{B}} & \mathbf{0} \end{bmatrix}$ corresponding to the equality-constrained QP is invertible. Therefore, there exists $\lambda_{\mathcal{B}}$ such that $\begin{bmatrix} \mathbf{y}^* \\ \lambda_{\mathcal{B}}^* \end{bmatrix} = \mathbf{K}^{-1} \begin{bmatrix} -\tilde{\mathbf{c}} \\ \mathbf{b}_{\mathcal{B}} \end{bmatrix}$. This point is helpful in designing the *unrolled active set method* for computing $\ell(\mathbf{P}, \boldsymbol{\pi}_\gamma)$ in the next section.

5.3 The Unrolled Active Set Method

We now introduce the *unrolled active set method*, a GJ algorithm with bounded complexities that exactly computes the optimal objective value of the perturbed PQP $\ell(\mathbf{P}, \boldsymbol{\pi}_\gamma)$ corresponding to the perturbed OQP $\boldsymbol{\pi}_\gamma$ and the projection matrix \mathbf{P} .

Intuition. The details of the unrolled active set method are demonstrated in Algorithm 1. Here, the algorithm is defined for each perturbed OQP $\boldsymbol{\pi}_\gamma$ and takes the projection matrix \mathbf{P} as the input. The general idea is to check all the potential active subsets \mathcal{A} of rows of $\tilde{\mathbf{A}} = \mathbf{A} \mathbf{P}$ up to $\min\{m, k\}$ elements. If we find \mathcal{A} such that KKT matrix

$K = \begin{bmatrix} \tilde{Q} & \tilde{A}_{\mathcal{A}}^{\top} \\ \tilde{A}_{\mathcal{A}} & \mathbf{0} \end{bmatrix}$ is invertible, then we can use it to calculate the potential optimal solution \mathbf{y}_{cand} and Lagrangian λ_{cand} . We then check if $(\mathbf{y}_{\text{cand}}, \lambda_{\text{cand}})$ is a KKT point of the perturbed PQP corresponding to the perturbed OQP π_{γ} and the projection matrix P . If yes, then \mathbf{y}_{cand} is the optimal solution for the perturbed PQP, and we output the optimal objective value; else we move on to the next potential active subset \mathcal{A} .

Correctness and GJ complexities. In this section, we demonstrate that the unrolled active set method yields the optimal solution for the perturbed PQP. Then, we will show that the algorithm is also a GJ algorithm, and we will bound its predicate complexity and degree.

Lemma 5.4. *Given a perturbed OQP π_{γ} , the algorithm $\Gamma_{\pi_{\gamma}}$ described by Algorithm 1 will output $\ell(P, \pi_{\gamma})$.*

Proof sketch. The detailed proof can be found in the appendix. To prove the existence, showing that the algorithm guarantees to find an optimal solution \mathbf{y}^* , we have to use Lemma 5.3, saying that there exists a subset $\mathcal{B} \subset \mathcal{A}(\mathbf{y}^*)$ such that \mathbf{y}^* is the solution of the equality constrained QP corresponding to $\tilde{A}_{\mathcal{B}}$ with linearly independent rows. Then, we notice that the algorithm will check all subsets of $\{1, \dots, m\}$ of at most $\min(m, k)$ elements, hence it will eventually check $\mathcal{A} = \mathcal{B}$. When $\mathcal{A} = \mathcal{B}$, we verify that the candidate \mathbf{y}_{cand} and λ_{cand} will pass all the primal and dual feasibility checks, and \mathbf{y}_{cand} is the optimal solution. For the correctness part, we will show that any \mathbf{y}_{cand} output by the algorithm is the optimal solution, by showing that $(\mathbf{y}_{\text{cand}}, \lambda)$, where $\lambda_{\mathcal{A}} = \lambda_{\text{cand}}$ and $\lambda_{\bar{\mathcal{A}}} = \mathbf{0}$, is indeed a KKT point. \square

Lemma 5.5. *Given a perturbed OQP π_{γ} , the algorithm $\Gamma_{\pi_{\gamma}}$ described by Algorithm 1 is a GJ algorithm with degree $\mathcal{O}(m+k)$ and predicate complexity $\mathcal{O}(m \min(2^m, (\frac{em}{k})^k))$.*

Proof sketch. First, note that $\tilde{Q} = P^{\top} Q_{\gamma} P$ is a matrix of which each entry is a polynomial in (the entries of) P of degree at most 2. Similarly, each entry of $\tilde{A} = AP$ and $\tilde{c} = P^{\top} c$ is a polynomial in P of degree at most 1. We show that we have to check at most $\min(2^m, (em/k)^k)$ potential active sets. For each potential active set \mathcal{A} , we show that the number of distinct predicates is $\mathcal{O}(m)$ and the maximum degree of each predicate is $\mathcal{O}(m+k)$. Combining those facts gives the result. \square

5.4 Pseudo-dimension Upper-bound Recovery for the Original Function Class

Using Lemma 5.5, we now give a concrete upper-bound for the pseudo-dimension of the perturbed function class \mathcal{L}_{γ} .

Lemma 5.6 (Pseudo-dimension of \mathcal{L}_{γ}). *We have $\text{Pdim}(\mathcal{L}_{\gamma}) = \mathcal{O}(nk \min(m, k \log m))$ for any $\gamma > 0$.*

Using the bound on the pseudo-dimension of perturbed function class $\text{Pdim}(\mathcal{L}_{\gamma})$ and the connection between \mathcal{L}_{γ} and \mathcal{L} via Lemma 5.1, we can bound the the pseudo-dimension of the original function class \mathcal{L} as follows.

Algorithm 1: The unrolled active set method $\Gamma_{\pi_{\gamma}}$ corresponding to the perturbed OQP $\pi_{\gamma} = (Q_{\gamma}, c, A, b)$

Input: Projection matrix $P \in \mathbb{R}^{n \times k}$

Output: Optimal value of the perturbed PQP.

```

1: Set  $\tilde{Q} = P^{\top} Q_{\gamma} P$ ,  $\tilde{A} = AP$ , and  $\tilde{c} = P^{\top} c$ .
2: for potential active set  $\mathcal{A} \subset \{1, \dots, m\}$ ,  $|\mathcal{A}| \leq \min\{m, k\}$  do
3:   Construct KKT matrix  $K = \begin{bmatrix} \tilde{Q} & \tilde{A}_{\mathcal{A}}^{\top} \\ \tilde{A}_{\mathcal{A}} & \mathbf{0} \end{bmatrix}$ 
4:   if  $\det(K) \neq 0$  then
5:     Compute  $\begin{bmatrix} \mathbf{y}_{\text{cand}} \\ \lambda_{\text{cand}} \end{bmatrix} = K^{-1} \begin{bmatrix} -\tilde{c} \\ \mathbf{b}_{\mathcal{A}} \end{bmatrix}$ .
6:     /* Checking feasibility of potential solution  $\mathbf{y}_{\text{cand}}$  */
7:      $yFeasible = True$ 
8:     for  $j \notin \mathcal{A}$  do
9:       if  $\tilde{A}_j^{\top} \mathbf{y}_{\text{cand}} > \mathbf{b}_j$  then
10:         $yFeasible = False$ 
11:       break
12:     end if
13:   end for
14:   if  $yFeasible$  then
15:     /* Checking validation of Lagrangian  $\lambda_{\text{cand}}$  */
16:      $lambdaValid = True$ 
17:     for  $j \in \mathcal{A}$  do
18:       if  $\lambda_{\text{cand},j} < 0$  then
19:         $lambdaValid = False$ 
20:       break
21:     end if
22:   end for
23:   if  $lambdaValid$  then
24:     return  $\frac{1}{2} \mathbf{y}_{\text{cand}}^{\top} \tilde{Q} \mathbf{y}_{\text{cand}} + \tilde{c}^{\top} \mathbf{y}_{\text{cand}}$ 
25:   end if
26: end if
27: end for

```

Theorem 5.7 (Pseudo-dimension of \mathcal{L}). *We have $\text{Pdim}(\mathcal{L}) = \mathcal{O}(nk \min(m, k \log m))$.*

Proof. First, we claim that $0 \leq \text{fatdim}_{\gamma R^2/2} \mathcal{L} \leq \text{Pdim}(\mathcal{L}_{\gamma})$ for any $\gamma > 0$, where $\text{fatdim}_{\alpha}(\mathcal{L})$ is the fat-shattering dimension of \mathcal{L} at scale α . To see that, assume $S = \{\pi_1, \dots, \pi_N\}$ is $\frac{\gamma R^2}{2}$ fat-shattered by \mathcal{L} , meaning that there exists real-valued thresholds $r_1, \dots, r_N \in \mathbb{R}$ such that for any $I \subset \{1, \dots, N\}$, there exists $\ell_P \in \mathcal{L}$ such that

$$f_P(\pi_i) > r_i + \frac{\gamma R^2}{2} \text{ for } i \in I, \quad \text{and}$$

$$f_P(\pi_j) < r_j - \frac{\gamma R^2}{2} \text{ for } j \notin I.$$

From Lemma 5.1, we have $0 \leq \ell_{P,\gamma}(\pi) - \ell_P(\pi) \leq \frac{\gamma R^2}{2}$ for any π and any $P \in \mathcal{P}$. This implies that $f_{P,\gamma}(\pi_i) > r_i$ if and only if $i \in I$. Therefore, S is also pseudo-shattered by \mathcal{L}_{γ} , which implies $0 \leq \text{fatdim}_{\gamma R^2/2}(\mathcal{L}) \leq \text{Pdim}(\mathcal{L}_{\gamma})$.

From Lemma 5.6, $\text{Pdim}(\mathcal{L}_\gamma) = \mathcal{O}(nk \min(m, k \log m))$ for any $\gamma > 0$, therefore $0 \leq \text{fatdim}_{\gamma R^2/2}(\mathcal{L}) \leq C \cdot nk \min(m, k \log m)$ for any $\gamma > 0$ and some fixed constant C . Taking the limit $\gamma \rightarrow 0^+$ and using properties of fat-shattering dimension, we have $0 \leq \text{Pdim}(\mathcal{L}) \leq C \cdot nk \min(m, k \log m)$, or $\text{Pdim}(\mathcal{L}) = \mathcal{O}(nk \min(m, k \log m))$. \square

Note that Theorem 5.7 is also applicable for data-driven learning projection matrix for LPs, as LP is a sub-problem of QP. Compared to the upper bound $\text{Pdim}(\mathcal{L}_{\text{LP}}) = \mathcal{O}(nk^2 \log mk)$ by Sakaue and Oki (2024, Theorem 4.4), our bound in Theorem 5.7 is strictly tighter and applicable to both QPs and LPs.

5.5 Lower Bound of Pseudo-dimension

For completeness, we also present the lower-bound for $\text{Pdim}(\mathcal{L})$, of which the construction is inspired by the construction of learning projection matrix for LPs (Sakaue and Oki 2024).

Proposition 5.8. *We have $\text{Pdim}(\mathcal{L}) = \Omega(nk)$.*

6 Extension to Other Settings

6.1 Learning to Match the Optimal Solution

In many cases, the decision-maker is interested in concrete solutions to implement. In these situations, the optimal value is less important than the optimal solution. We now consider the setting where we want to learn the projection matrix P such that the optimal solution of the PQP is close to that of the OQP. Such an approximative solution can also be used to warm-start an exact solver and accelerate the solving process. In this section, we propose an alternative objective value for learning P . First, we assume the strict convexity of the problem instance, so that the optimal solution of the OQP is well-defined (unique).

Assumption 2. For any $\pi = (Q, c, A, b) \in \Pi$, the matrix Q is positive definite.

Under such assumption, we seek the projection matrix P such that the recovered solution is close to the optimal solution of the PQP in expectation, i.e., $P_{\mathcal{D}}^* \in \arg \min_{P \in \mathcal{P}} \mathbb{E}_{\pi \sim \mathcal{D}} [\ell_{\text{match}}(P, \pi)]$, where $\ell_{\text{match}}(P, \pi) = \|\mathbf{x}_\pi^* - P\mathbf{y}^*(P, \pi)\|_2^2$ is the matching loss, $\mathbf{x}_\pi^* = \arg \min_{\mathbf{x} \in \mathbb{R}^n} \{\frac{1}{2}\mathbf{x}^\top Q\mathbf{x} + c^\top \mathbf{x} \mid A\mathbf{x} \leq \mathbf{b}\}$ is the optimal solution of the OQP, and $\mathbf{y}^*(P, \pi) = \arg \min_{\mathbf{y} \in \mathbb{R}^k} \{\frac{1}{2}\mathbf{y}^\top P^\top Q P \mathbf{y} + c^\top P \mathbf{y} \mid A P \mathbf{y} \leq \mathbf{b}\}$ is the optimal solution of the PQP. Again, since \mathcal{D} is unknown, we are instead given N problem instances $S = \{\pi_1, \dots, \pi_N\}$ drawn i.i.d. from \mathcal{D} , and learn P via ERM, i.e., $\hat{P}_S \in \arg \min_{P \in \mathcal{P}} \frac{1}{N} \sum_{i=1}^N \ell_{\text{match}}(P, \pi_i)$. Let $\mathcal{L}_{\text{match}} = \{\ell_{\text{match}, P} : \Pi \rightarrow [-H, 0]\} \mid P \in \mathcal{P}$, where $\ell_{\text{match}, P}(\pi) := \ell_{\text{match}}(P, \pi)$. The following result provides the upper-bound for the pseudo-dimension of $\mathcal{L}_{\text{match}}$.

Theorem 6.1. *Assuming that all the QPs satisfies Assumption 2 so that \mathbf{x}_π^* is defined uniquely. Then $\text{Pdim}(\mathcal{L}_{\text{match}}) = \mathcal{O}(nk \min(m, k \log m))$.*

Proof sketch. Given π , the general idea is using a variant of the unrolled active set method (Algorithm 1) to calculate the optimal solution $\mathbf{y}^*(P, \pi)$. Then $\ell_{\text{match}}(P, \pi)$ can also be calculated with a GJ algorithm with a bounded predicate complexity and degree, based on the GJ algorithm calculating $\mathbf{y}^*(P, \pi)$. Finally, Theorem 3.2 gives us the final guarantee. \square

6.2 Input-aware Learning of Projection Matrix

In this section, we consider the setting of input-aware data-driven learning of the projection matrix for QPs, recently proposed by Iwata and Sakaue (2025) in the context of LP. Here, instead of learning a single projection matrix P , we learn a mapping $f_\theta : \Pi \rightarrow \mathcal{P}$, e.g., a neural network, that takes a problem instance π drawn from \mathcal{D} and outputs the corresponding projection matrix $P_\pi = f_\theta(\pi)$. With some computational trade-off for generating the projection matrix, this method has shown promising results, generating a better input-aware projection matrix that achieves better performance than an input-agnostic projection matrix while using the same projection dimension k .

Network architecture. Inspired by Iwata and Sakaue (2025), we assume that f_θ is a neural network parameterized by $\theta \in \Theta \subset \mathbb{R}^W$, where W is the number of parameters of the neural network that takes the input π_{flat} of size $n^2 + n + nm + m$ that is formed by flattening Q, c, A, b in π . Let L be the number of hidden layers, and let f_θ be the network of $L + 2$ layers, with the number of neurons of input layer is $W_0 = m^2 + n + mn + m$, that of the output layer is $W_{L+2} = nk$, and that of i^{th} layer is W_i for $i \in \{1, \dots, L\}$. Each hidden layer uses ReLU as the non-linear activation function, and let $U = \sum_{i=1}^N W_i$ be the number of hidden neurons. Consider the function class $\mathcal{L}_{\text{ia}} = \{\ell_\theta : \Pi \rightarrow [-H, 0] \mid \theta \in \Theta\}$, where $\ell_\theta(\pi) := \ell(f_\theta(\pi_{\text{flat}}), \pi)$. Then we have the following result, which bounds the pseudo-dimension of \mathcal{L}_{ia} .

Theorem 6.2. *Assume that the output $f_\theta(\pi)$ has full column rank, then $\text{Pdim}(\mathcal{L}_{\text{ia}, \gamma}) = \mathcal{O}(W(L \log(U + mk) + \min(m, k \log m)))$.*

7 Conclusion and Future Works

We introduced the task of data-driven learning of a projection matrix for convex QPs. By a novel analysis approach, we establish the first upper bound on the pseudo-dimension of the learning projection matrix in QPs. Compared to the previous bound of (Sakaue and Oki 2024), our new result is more general because it applies to both QPs and LPs and is strictly tighter. We further extend our analysis to learning to match the optimal solution and the input-aware setting. Our analysis opens many interesting directions for extension, including the conic programming and (mixed) integer programming.

Acknowledgments

Viet Anh Nguyen gratefully acknowledges the support from the CUHK's Improvement on Competitiveness in Hiring New Faculties Funding Scheme, UGC ECS Grant 24210924, and UGC GRF Grant 14208625.

References

- Akchen, Y.-C.; and Misić, V. V. 2025. Column-randomized linear programs: Performance guarantees and applications. *Operations Research*, 73(3): 1366–1383.
- Amos, B. 2022. Tutorial on amortized optimization for learning to optimize over continuous domains. *arXiv preprint arXiv:2202.00665*.
- Amos, B.; et al. 2023. Tutorial on amortized optimization. *Foundations and Trends® in Machine Learning*, 16(5): 592–732.
- Applegate, D.; Díaz, M.; Hinder, O.; Lu, H.; Lubin, M.; O’Donoghue, B.; and Schudy, W. 2021. Practical large-scale linear programming using primal-dual hybrid gradient. *Advances in Neural Information Processing Systems*, 34: 20243–20257.
- Balcan, M.-F. 2020. Data-driven algorithm design. *arXiv preprint arXiv:2011.07177*.
- Balcan, M.-F.; Dick, T.; Sandholm, T.; and Vitercik, E. 2018. Learning to branch. In *International conference on machine learning*, 344–353. PMLR.
- Balcan, M. F.; Nguyen, A. T.; and Sharma, D. 2025a. Algorithm configuration for structured Pfaffian settings. *Transactions on Machine Learning Research*.
- Balcan, M. F.; Nguyen, A. T.; and Sharma, D. 2025b. Sample complexity of data-driven tuning of model hyperparameters in neural networks with structured parameter-dependent dual function. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Balcan, M.-F. F.; Khodak, M.; Sharma, D.; and Talwalkar, A. 2022. Provably tuning the ElasticNet across instances. *Advances in Neural Information Processing Systems*, 35: 27769–27782.
- Balcan, M.-F. F.; Nguyen, A.; and Sharma, D. 2023. New bounds for hyperparameter tuning of regression problems across instances. *Advances in Neural Information Processing Systems*, 36: 80066–80078.
- Bartlett, P.; Indyk, P.; and Wagner, T. 2022. Generalization bounds for data-driven numerical linear algebra. In *Conference on Learning Theory*, 2013–2040. PMLR.
- Bengio, Y.; Lodi, A.; and Prouvost, A. 2021. Machine learning for combinatorial optimization: A methodological tour d’horizon. *European Journal of Operational Research*, 290(2): 405–421.
- Chen, T.; Chen, X.; Chen, W.; Heaton, H.; Liu, J.; Wang, Z.; and Yin, W. 2022. Learning to optimize: A primer and a benchmark. *Journal of Machine Learning Research*, 23(189): 1–59.
- Chowdhury, A.; Dexter, G.; London, P.; Avron, H.; and Drineas, P. 2022. Faster randomized interior point methods for tall/wide linear programs. *Journal of Machine Learning Research*, 23(336): 1–48.
- Dikin, I. I. 1967. Iterative solution of problems of linear and quadratic programming. In *Doklady Akademii Nauk*, volume 8, 674–675.
- Dostál, Z. 2009. *Optimal Quadratic Programming Algorithms: With Applications to Variational Inequalities*, volume 23. Springer Science & Business Media.
- d’Ambrosio, C.; Liberti, L.; Poirion, P.-L.; and Vu, K. 2020. Random projections for quadratic programs. *Mathematical Programming*, 183(1): 619–647.
- Gass, S. I. 2003. *Linear programming: methods and applications*. Courier Corporation.
- Goldberg, P.; and Jerrum, M. 1993. Bounding the Vapnik-Chervonenkis dimension of concept classes parameterized by real numbers. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory*, 361–369.
- Gupta, R.; and Roughgarden, T. 2020. Data-driven algorithm design. *Communications of the ACM*, 63(6): 87–94.
- Huangfu, Q.; and Hall, J. J. 2018. Parallelizing the dual revised simplex method. *Mathematical Programming Computation*, 10(1): 119–142.
- Indyk, P.; Vakilian, A.; and Yuan, Y. 2019. Learning-based low-rank approximations. *Advances in Neural Information Processing Systems*, 32.
- Iwata, T.; and Sakaue, S. 2025. Learning to generate projections for reducing dimensionality of heterogeneous linear programming problems. In *Forty-second International Conference on Machine Learning*.
- Li, Y.; Lin, H.; Liu, S.; Vakilian, A.; and Woodruff, D. 2023. Learning the Positions in CountSketch. In *The Eleventh International Conference on Learning Representations*.
- Nocedal, J.; and Wright, S. J. 2006. *Numerical Optimization*. Springer.
- Poirion, P.-L.; Lourenco, B. F.; and Takeda, A. 2023. Random projections of linear and semidefinite problems with linear inequalities. *Linear Algebra and its Applications*, 664: 24–60.
- Pollard, D. 1984. *Convergence of Stochastic Processes*. Springer Series in Statistics.
- Sakaue, S.; and Oki, T. 2023. Improved generalization bound and learning of sparsity patterns for data-driven low-rank approximation. In *International Conference on Artificial Intelligence and Statistics*, 1–10. PMLR.
- Sakaue, S.; and Oki, T. 2024. Generalization bound and learning methods for data-driven projections in linear programming. *Advances in Neural Information Processing Systems*, 37: 12825–12846.
- Vu, K.; Poirion, P.-L.; d’Ambrosio, C.; and Liberti, L. 2019. Random projections for quadratic programs over a Euclidean ball. In *International Conference on Integer Programming and Combinatorial Optimization*, 442–452. Springer.
- Vu, K.; Poirion, P.-L.; and Liberti, L. 2018. Random projections for linear programming. *Mathematics of Operations Research*, 43(4): 1051–1071.