

Dimensionality Testing, Shape-Constrained Neural Network Estimation, and Equilibrium Approximation in Team Games

by

Zetian Zhang

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Applied Economics

Waterloo, Ontario, Canada, 2026

© Zetian Zhang 2026

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

Supervisor(s): Tao Chen
Associate Professor
Department of Economics
University of Waterloo

Internal Member: Thomas Parker
Associate Professor
Department of Economics
University of Waterloo

Clemens Possnig
Assistant Professor
Department of Economics
University of Waterloo

Internal-External Member: Bin Li
Professor
Department of Statistics and Actuarial Science
University of Waterloo

External Examiner: Tingting Ding
Assistant Professor
Department of Economics
College of Business
James Madison University

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

While Chapter 1 and 2 are co-authored with my supervisor, Dr. Tao Chen, from the Department of Economics at the University of Waterloo, I have made a major contribution to the work, including all of the proofs, the simulation study and the writing of manuscripts.

Abstract

Modern advancements in computational technology have catalyzed an explosion in data dimensionality, enabling researchers to model intricate relationships among multitudinous variables simultaneously; however, these advancements also challenge the application of many useful classical theories (for instance, the F -test which is one of the most widely used tests fails with increasing dimensions alongside sample growth). Although some modifications of classical theories, such as the corrected F -test, have been proposed to mitigate the high-dimensional bias, a more fundamental question about whether the dimension is “high” has not been addressed. In Chapter 1, we tackle this problem by proposing a test statistic, which is asymptotically normal, to distinguish high from non-high dimensionality for vectors. In large sample sizes, our test has the correct size when the dimension is “high”; and it is asymptotically powerful against all alternatives with “low” dimensions. A set of simulation studies demonstrates that our test achieves satisfactory size and power even in small to moderate sample sizes.

Deep neural networks (DNNs), as the most modern advancement, have demonstrated exceptional performance across a wide range of domains, such as image classification (Krizhevsky et al. 2012, He et al. 2016), speech recognition (Hinton et al. 2012), natural language processing (Mikolov et al. 2013, Vaswani et al. 2017) and game intelligence (Silver et al. 2016, Perolat et al. 2022). DNNs can be conceptualized as a generalization of classical sieve estimators, transitioning from “single-layer” to “multi-layer” architectures. This architectural evolution substantially expands the model’s parameter space, but diminishing the interpretability of parameters and the model itself. In Chapter 2, we provide the consistency and asymptotic normality of the DNN-based estimator for point estimation and extend these properties the case with prior knowledge of shapes.

The incorporation of prior knowledge into neural networks to enhance model performance has evolved into a fundamental methodology, being widely used in real-world applications. The abstract prior knowledge, particularly the shape restrictions such as monotonicity, convexity, and homogeneity, hold critical importance within economic modeling frameworks, where functional specifications like production and utility functions inherently require certain shape restrictions to ensure both mathematical and theoretical validity in economic analyses. However, current methods exhibit fundamental limitations in effectively embedding abstract knowledge, since those methods either implement loss function modifications that fail to guarantee the desired shape, or require specifically designed structures through restricting the number or type of layers to certain forms—a parametric imposition that essentially limits model expressiveness. Our shape-restricted approximator is shown

to be consistent and asymptotically normal under mild regularity conditions. Based on case studies, the proposed method outperforms the existing shape-restricted networks.

Two-team zero-sum games model a broad range of real-world competitive interactions more effectively than traditional two-player games, yet they remain relatively underexplored. Existing successful approaches for two-player games, including Go, poker, and Stratego, are difficult to extend directly to team settings, as they cannot simultaneously account for inter-team competition and intra-team cooperation. In chapter 3, we investigate two-team zero-sum games under the setting where teammates cannot communicate, with the goal of approximating the team Nash equilibrium. Unlike existing methods such as TMECor (McAleer et al. 2023) and GFXP (Xu et al. 2025a), which rely on communication or correlation devices by treating each team as a centralized joint player, our method directly tackles decentralized team settings without such coordination mechanisms. We propose a novel extension of Policy-Space Response Oracles (PSRO) (Lanctot et al. 2017) that incorporates multi-agent reinforcement learning (MARL) to compute joint cooperative best responses during population expansion. By leveraging shared team rewards, the proposed method captures cooperation within teams while preserving competitive optimization across teams. The proposed algorithm demonstrates a stronger tendency to converge to global Nash equilibria compared with classical PSRO based on some experimental results.

Acknowledgements

I would like to express my sincere gratitude to Dr. Tao Chen, Dr. Thomas Parker, Dr. Clemens Possnig, and Dr. Tingting Ding for their invaluable guidance, insightful feedback, and generous sharing of knowledge throughout this research. This thesis would not have been possible without their support and encouragement.

I am especially grateful to my supervisor, Dr. Tao Chen, whose mentorship has profoundly shaped my PhD journey. His rigorous academic guidance, deep expertise, and remarkable patience have continually inspired me to strive for excellence and become a better scholar. Under his supervision, I have gained invaluable insights not only into academic research, but also into approaches to life and personal growth that will remain with me far beyond my doctoral studies. It has been a great privilege and honour to be his student.

I would also like to extend my appreciation to the faculty, staff, and colleagues in the Department of Economics at the University of Waterloo for their support and kindness. Their assistance and companionship have made my time at Waterloo both rewarding and memorable.

Finally, I wish to express my deepest gratitude to my parents for their unconditional love, unwavering encouragement, and constant belief in me. Their support has been my greatest source of strength.

Dedication

To those who have loved and supported me, I dedicate this thesis.

Table of Contents

Examining Committee Membership	ii
Author's Declaration	iii
Statement of Contributions	iv
Abstract	v
Acknowledgements	vii
Dedication	viii
List of Figures	xii
List of Tables	xiii
Introduction	1
1 Test for Dimensionality of Estimated Vectors	4
1.1 Introduction	4
1.2 Test for Estimated Vectors from Linear Regressions	7
1.3 Test for Estimated Vectors from Nonlinear Regressions	10
1.4 Simulation Studies	14

1.4.1	Classical Theories in High Dimensions	16
1.5	Conclusion	18
2	Network Estimation under Shape Restrictions	19
2.1	Introduction	19
2.1.1	Monotonicity	23
2.1.2	Convexity	23
2.1.3	Homogeneity	24
2.2	Large Sample Properties	25
2.2.1	Rate of Convergence	25
2.2.2	Asymptotic Normality	28
2.3	Experiments	31
2.3.1	Case Studies	35
2.4	Conclusion	37
3	Nash Equilibrium Approximation in Team Games	38
3.1	Introduction	38
3.2	Game Setup	40
3.3	Literature Review	42
3.3.1	Reinforcement Learning based Best Response	42
3.3.2	Meta-Solver	44
3.3.3	Policy Population Initialization	45
3.4	Methodology	46
3.4.1	Team Equilibrium without Communication	46
3.4.2	Equilibrium Approximation Algorithm	48
3.5	Experiments	49
3.5.1	Practical Implementation	50
3.5.2	Matrix Game	50
3.5.3	Sequential Game	55
3.6	Conclusion	58

References	59
APPENDICES	70
A Proofs and Supplementary Text in Chapter 1	71
A.1 Proofs	71
A.1.1 Proof of Lemma 1.2.1	71
A.1.2 Proof of Lemma 1.2.2	72
A.1.3 Proof of Theorem 1.2.1	73
A.1.4 Proof of Lemma 1.3.1	74
A.1.5 Proof of Lemma 1.3.2	75
A.1.6 Proof of Theorem 1.3.1	77
A.2 Supplementary Text	77
B Proofs and Supplementary Text in Chapter 2	81
B.1 Proofs	81
B.1.1 Part I: Shape Constraints	81
B.1.2 Part II: Rate of Convergence and Asymptotic Normality	85
B.2 Supplementary Text	94
C Proofs and Supplementary Text/Figure in Chapter 3	96
C.1 Proof of Proposition 3.4.1 and 3.4.2	96
C.2 Supplementary Text/Figure	96
C.2.1 Team Kuhn Poker	96
C.2.2 Examples of Leaning Dynamics	98

List of Figures

2.1	Deep Neural Network	29
3.1	Classical PSRO's learning dynamics.	52
3.2	Proposed algorithm's learning dynamics.	53
3.3	Learning dynamics of classical PSRO and proposed algorithm. . .	58
B.1	2-hidden-layer network.	94
B.2	Distributions of point estimates.	95
C.1	Learning dynamics of classical PSRO and proposed algorithm Ex- ample I.	98
C.2	Learning dynamics of classical PSRO and proposed algorithm Ex- ample II.	99
C.3	Learning dynamics of classical PSRO and proposed algorithm Ex- ample III.	100

List of Tables

1.1	Empirical rejection rates for estimated vectors.	15
1.2	Empirical rejection rates for estimated vectors with different D/N	16
1.3	Empirical rejection rates of the F -test and modified F -test.	17
2.1	Results for monotonicity	33
2.2	Results for convexity	33
2.3	Results for homogeneity	34
2.4	Results for the combination of monotonicity and homogeneity	35
2.5	The results for benchmark, and our shape-restricted estimation	36
3.1	Classical PSRO's Nash equilibrium	55
3.2	Proposed algorithm's Nash equilibrium	56
A.1	Empirical rejection rates for estimated vectors from linear regressions	78
A.2	Empirical rejection rates for estimated vectors from nonlinear regressions	80

Introduction

The rapid development of modern data acquisition technologies, computing facilities, and machine learning methodologies has fundamentally transformed the landscape of statistical modeling and artificial intelligence research. This thesis studies several methodological and theoretical problems arising in high-dimensional statistics, shape-constrained neural network learning, and Nash equilibrium approximation in team game settings, respectively.

The first chapter of this thesis investigates the problem of determining whether the classical or high-dimensional theories for a given dataset or model are valid. Classical statistical theories were primarily developed under low-dimensional settings (Rowell & Walters 1976); however, contemporary applications increasingly involve high dimensions. Representative examples include large-scale neural networks with billions of parameters (Zhao et al. 2023). These developments have substantially expanded the applicability of modern statistical learning methods while simultaneously exposing the theoretical limitations of classical approaches due to the increasing dimension. A major challenge arising from these modern applications is that many conventional statistical theories become invalid when dimensionality grows with the sample size. Classical inference procedures often rely on assumptions that fail under high-dimensional settings (Dempster 1958, Bai & Saranadasa 1996, Calhoun 2011, Sur et al. 2019). Although some modified classical and high-dimensional theories have been proposed, these theories are typically derived under asymptotic assumptions involving high dimensionality, yet the practical question of whether a given dataset or model is sufficiently high-dimensional for these theories to apply remains largely unresolved. To address this issue, this thesis develops a general testing method to distinguish high from non-high dimensionality for estimated vectors, enabling scholars to assess the validity of classical or high-dimensional theories for a given dataset or model. The proposed methodology is motivated by the geometric properties of high-dimensional random vectors and develops test statistics based on the asymptotic behavior of vector norms. Furthermore, simulation studies demonstrate that the proposed test exhibits satisfactory performance, even under small to moderate sample sizes.

The second chapter focuses on incorporating structural prior knowledge into neural network models. Despite their remarkable empirical success, neural networks often lack effective mechanisms for embedding structural constraints and provide limited theoretical guarantees on their asymptotic properties. Structural prior knowledge, including monotonicity, convexity, and homogeneity constraints, plays a fundamental role in many scientific and economic applications. Existing approaches for enforcing such shape constraints either rely on penalty-based regularization, which cannot guarantee exact satisfaction of constraints, or require specialized network architectures that sacrifice flexibility, transferability, and computational efficiency (Sill 1997, Bamberger et al. 2023, Warin 2023, Amos et al. 2017, Calafiore et al. 2020, Tang et al. 2020). To overcome these limitations, this thesis proposes a flexible shape-constrained correction framework applicable to general neural networks without modifying their original structures. Building upon monotonization procedures and Euler’s homogeneous function theorem, the proposed method transforms unconstrained neural network estimators into shape-constrained estimators while preserving desirable asymptotic properties. In particular, the corrected estimators retain convergence rates comparable to those of the preliminary networks and are shown to satisfy asymptotic normality under mild regularity conditions.

The final chapter investigates equilibrium approximation in two-team zero-sum games. In contrast to traditional two-player games, two-team competitive settings provide a more expressive framework for modeling strategic interactions that involve both cooperative behavior within teams and competitive behavior across teams. Recent advances in reinforcement learning and game theory have demonstrated impressive performance in competitive environments such as Go, poker, and Stratego (Silver et al. 2016, Moravčík et al. 2017, Brown & Sandholm 2018, Perolat et al. 2022); however, extending these methodologies to more realistic multi-agent settings involving both intra-team cooperation and inter-team competition remains highly challenging. Existing equilibrium learning methods (McAleer et al. 2023, Xu et al. 2025a) for team games largely focus on communication-enabled settings and rely on correlated strategies that are infeasible in decentralized environments. This thesis therefore considers the more challenging setting in which any communication and correlation devices are not allowed and investigates the approximation of team Nash equilibrium. To address this problem, a novel extension of Policy-Space Response Oracles (PSRO) is proposed by integrating multi-agent reinforcement learning into the population expansion procedure. The resulting algorithm enables cooperative best-response learning within teams while maintaining competitive optimization between teams. The proposed framework converges to team Nash equilibrium without communication and empirically demonstrates stronger tendencies toward global equilibrium convergence than classical PSRO.

Overall, this thesis contributes to the theoretical and methodological foundations of modern statistical learning and artificial intelligence by addressing three important challenges: providing a general testing method to distinguish high from non-high dimensionality for estimated vectors, incorporating structural constraints into neural networks while preserving flexibility and asymptotic properties, and developing equilibrium learning algorithms for decentralized team-based competitive games.

Chapter 1

Test for Dimensionality of Estimated Vectors

1.1 Introduction

Prior to the mid-twentieth century, experimental designs were predominantly developed under settings characterized by relatively large sample sizes and low-dimensional measurement spaces (Rowell & Walters 1976). Consequently, classical statistical theories were largely established under the assumption that the dimension of variables remained fixed or grew slowly relative to the sample size. Over the past several decades, however, this paradigm has changed substantially due to rapid advances in data acquisition technologies and computing facilities. As pointed out by Lindsay et al. (2004), modern scientific and technological developments have fundamentally transformed the nature of practical problems, leading to increasingly complex datasets with a large number of variables. Accordingly, statistical analysis has progressively shifted toward scenarios involving high-dimensional data, where the dimension of observations and/or model parameters may grow substantially with the sample size. This transition is particularly evident in modern machine learning applications, such as large language models, where the number of trainable parameters can exceed hundreds of billions (Zhao et al. 2023).

The emergence of high-dimensional problems has substantially challenged many classical statistical theories. When the dimension increases relative to the sample size, traditional methods may lose their validity or exhibit severe distortions. For example, Dempster (1958) showed that the classical Hotelling’s T^2 test for the two-sample problem become undefined when the dimension approaches the degrees of freedom available for estimating

the covariance matrix. Although dimension-agnostic testing procedures that remain valid regardless of dimensionality have been developed for the one-sample problem (Kim & Ramdas 2024), the existing method for the two-sample problem (Bai & Saranadasa 1996) is only justified under high-dimensional assumption. Similar issues arise for likelihood-based inference and regression analysis. In particular, the classical F-test and likelihood-ratio test require modifications when the dimension is proportional to the sample size, leading to the corrected procedures developed by Calhoun (2011) and Sur et al. (2019), respectively. Furthermore, classical asymptotic theories for estimators, including M-estimators, have been revisited under increasing dimensionality (Huber 1973, Portnoy 1984, 1985, 1988, He & Shao 2000). Meanwhile, increasing dimensionality is not merely a source of statistical difficulty but also provides new theoretical opportunities. For example, Vershynin (2018) provided a comprehensive treatment of high-dimensional theories for random vectors and matrices, presenting concentration inequalities for various norms, some useful inequalities on isometry approximations, etc.

Although various modified classical and high-dimensional theories tailored for high-dimensional settings have been proposed, the fundamental question of whether the dimensionality is sufficiently high for these theories to be applicable remains largely unresolved. To investigate this problem, it is first necessary to clarify the meaning of dimensionality in modern statistical models. In general modelling frameworks, the concept of dimension can be grouped into two categories: the input/data dimension, which is the number of features of each observation, and the parameter dimension, which is the number of coefficients/parameters need to be estimated. Since the observations and coefficients are typically characterized as vectors not being estimated (called random vectors) and vectors being estimated (called estimated vectors), respectively, these two categories of dimension can be unified to vector dimensions. When either the data dimension or the parameter dimension (or both) increases with the sample size, many classical statistical theories are no longer valid. For random vectors, existing studies have begun to address the problem of identifying high-dimensional regimes. In particular, Xu (2024-08-19) proposed a test for determining whether the dimension of random vectors is sufficiently large. However, the corresponding problem for estimated vectors remains unresolved. Thus, the main contribution of this chapter is providing a general testing method to distinguish high from non-high dimensionality for estimated vectors, enabling scholars to assess the validity of classical or high-dimensional theories for a given dataset or model.

In particular, in this chapter, we propose a statistic to test the following:

$$H_0 : \text{The estimated vector is in high dimensions.}$$

The preceding discussions have introduced the notion of “high dimensions,” which we now

formally define as the scenario in which the high-dimensional theories hold, or the classical theories fail and need to be modified. It is important to elucidate this definition for further interpretation of testing results; thus, the typical example, the failure of the F -test for estimated vectors is presented. The classical F -test is asymptotically invalid and the test tends to over-reject (Calhoun 2011) when the estimated vectors are in high dimensions. Calhoun (2011) suggested to modify the classical statistic $\hat{F}_n$ to

$$\hat{G}_n = \hat{v}_n \hat{F}_n + (1 - \hat{v}_n),$$

where v_n depends on the sample size, number of regressors and restrictions. However, this modification induces a non-negligible deviation from the classical statistic $\hat{F}_n$ in low-dimensional settings, thereby constraining its applicability exclusively to high dimensions. Therefore, the defined “high dimensions” is the scenario, in which the high-dimensional theories or the modification of the classical theories should be applied, because the classical theories, as exemplified by the F -test, do not hold. Due to the abstract nature of null hypothesis, we need to carefully interpret the testing results. If we fail to reject the null hypothesis, the random or estimated vector is in high dimensions, which implies that the high-dimensional theories or the modified classical theories should be applied; otherwise, the vector falls into the non-high dimensions in which the classical theories should be applied.

The proposed test fundamentally evaluates the distribution of the norm of vectors, which is motivated by the fact that the standard normal random vector in high dimensions approximately follows the uniform distribution on a sphere whose radius depends on its norm (Vershynin 2018). This norm with proper adjustments is asymptotically normal as $D, N \rightarrow \infty$, then forming our test statistic. Such testing framework can be generalized to different estimators, provided that their consistency is satisfied. In particular, the least squares (LS) estimator utilized in our test requires the sample size N and dimension D satisfying $D < N$. When alternative estimators are employed, the required asymptotic relationship between N and D may vary depending on the estimator (Huber 1973, Portnoy 1984, 1985, 1988, He & Shao 2000, Sur et al. 2019, Sur & Candès 2019). Moreover, results from a set of simulation studies suggest that our test performs well even in small to moderate sample sizes.

1.2 Test for Estimated Vectors from Linear Regressions

Consider the linear regression model

$$Y_n = \mathbf{X}_n^\top \beta_0 + \epsilon_n, \quad n = 1, 2, \dots, N,$$

where Y_n is the scalar response, $\mathbf{X}_n$ is a D -dimensional vector of explanatory variables, β_0 is a D -dimensional vector of unknown parameters, and ϵ_n is the random error. As mentioned, we work with the i.i.d. samples. With proper conditions on some higher moments of the joint distribution of $\{\mathbf{X}_n, \epsilon_n\}$, heteroskedasticity or weak dependence among individuals can also be accommodated. Moreover, assume that $E[\epsilon_n | \mathbf{X}_n] = 0$ and $\text{Var}(\epsilon_n | \mathbf{X}_n) = \sigma^2$ for each $n = 1, 2, \dots, N$; and $\Sigma := \mathbb{E}[\mathbf{X}_n \mathbf{X}_n^\top]$ exists with a bounded spectral norm.

To test the dimensionality of the β_0 , an estimator is required at the first step. The LS estimator $\hat{\beta} = \arg \min_{\beta} \sum_{n=1}^N (Y_n - \mathbf{X}_n^\top \beta)^2$, as one of the most widely used estimators, is employed in our test. To ensure its consistency, the following two assumptions need to hold in probability.

Assumption 1.2.1 *For any $\epsilon > 0$, there is a positive constant B , such that for all n ,*

$$P\left(\frac{\mathbf{X}_n^\top \mathbf{X}_n}{D} > B\right) < \epsilon$$

Assumption 1.2.2 *There are positive constants b and c , such that the minimum and maximum eigenvalues of $\frac{\mathbb{X}^\top \mathbb{X}}{N}$ satisfy*

$$\lambda_{\min}\left(\frac{\mathbb{X}^\top \mathbb{X}}{N}\right) \geq b \quad \text{and} \quad \lambda_{\max}\left(\frac{\mathbb{X}^\top \mathbb{X}}{N}\right) \leq c,$$

where $\mathbb{X}$ is an $N \times D$ matrix with $\mathbf{X}_n^\top$ as each row.

Assumptions 1.2.1 and 1.2.2 follow from the $X1$ assumption (Portnoy 1985), which is technically required specifically for the LS estimator, which implies that these assumptions may require modification or could even be relaxed for alternative. Then, the following lemma can be established as follow.

Lemma 1.2.1 *Under Assumptions 1.2.1 and 1.2.2, $\|\hat{\beta} - \beta_0\|^2 = O_p(\frac{D}{N})$.*

Lemma 1.2.1 provides the property of convergence for the employed OLS estimator, which implies that our test does not strictly require consistent estimator, i.e. $\|\hat{\beta} - \beta_0\| = o_p(1)$. As mentioned in introduction, the asymptotics of the proposed test statistic is build under $D, N \rightarrow \infty$, and the asymptotic relationship between D and N are not directly assumed. Besides the assumptions for estimation, the asymptotics of the test statistic are established also under the following assumptions.

Assumption 1.2.3

$$\left\| \frac{\mathbb{X}^\top \mathbb{X}}{N} - \Sigma \right\| = o_p(D^{-1/2}).$$

Assumption 1.2.4 *There are constant e and a sequence of constants $B_N \geq 1$ such that*

- (a) $\frac{1}{N} \sum_{n=1}^N \mathbb{E} [(\mathbf{X}_n \epsilon_n)_d^2] \geq e$ *for all $d = 1, 2, \dots, D$;*
- (b) $\frac{1}{N} \sum_{n=1}^N \mathbb{E} [(\mathbf{X}_n \epsilon_n)_d^{2+k}] \leq B_N^k$ *for all $d = 1, 2, \dots, D$ and $k = 1, 2$;*
- (c) $\mathbb{E} [\exp(|(\mathbf{X}_n \epsilon_n)_d|/B_N)] \leq 2$ *for all $d = 1, 2, \dots, D$ and $n = 1, 2, \dots, N$;*
- (d) $\frac{B_N^2 \log^7(DN)}{N} = o_p(1)$.

Assumption 1.2.3 implies that the matrix estimation error, measured by the Frobenius norm, decreases as the dimension D increases. This is admittedly a relatively strong assumption, requiring the estimator to become increasingly accurate as the dimension grows. Nevertheless, the assumption is introduced primarily for theoretical convenience. The proposed framework is not restricted to a particular matrix estimator, and other estimators can also be incorporated provided that they satisfy this required bound. Assumption 1.2.4 ensues the application of high-dimensional CLT. Then, the following lemma is obtained.

Lemma 1.2.2 *Under Assumptions 1.2.1–1.2.4, there is*

$$t_{D,N} := \sqrt{\frac{D}{2}} \left(\frac{N}{D} \left\| \sigma^{-1} \left(\frac{\mathbb{X}^\top \mathbb{X}}{N} \right)^{1/2} (\hat{\beta} - \beta_0) \right\|^2 - 1 \right) \xrightarrow{d} \mathcal{N}(0, 1).$$

As mentioned in introduction, Lemma 1.2.2 provides the asymptotic normality similar to those in the case of general random vectors (Xu 2024-08-19); however, the LHS in the Lemma 1.2.2 cannot be the test statistic since the true value β_0 is typically unknown in practice. On the other hand, the tails of their asymptotic distributions under the null and alternative hypothesis do not differ significantly, which will lead less power. Therefore, a KS-type test statistic is constructed to address this issue, which leading in a double bootstrapped test. We use the mean of bootstrap estimates approximating the true value to construct a KS-type bootstrapped test, which is implemented as follows:

- (i) Estimate $\hat{\beta}$ and obtain $\hat{\epsilon}_n$ where $\hat{\epsilon}_n := Y_n - \mathbf{X}_n^\top \hat{\beta}$ for $n = 1, 2, \dots, N$.
- (ii) Independent of the observed data $D_N := \{(Y_n, \mathbf{X}_n) : 1 \leq n \leq N\}$, use a random number generator to generate $R_1, R_2, \dots, R_N \stackrel{iid}{\sim}$ Rademacher. Define $Y_n^* := \mathbf{X}_n^\top \hat{\beta} + R_n \hat{\epsilon}_n$ for $n = 1, 2, \dots, N$.
- (iii) Re-estimate β using the data $\{(Y_n^*, \mathbf{X}_n) : 1 \leq n \leq N\}$. Denote it by β^* .
- (iv) Repeat steps (ii)-(iii) B times. We calculate the following and denote it as t_k^*

$$t_k^* := \sqrt{\frac{D}{2}} \left(\frac{N}{D} \left\| \hat{\Sigma}^{-1/2} \sqrt{B} (\hat{\beta} - \bar{\beta}^*) \right\|^2 - 1 \right),$$

$$\text{where } \hat{\Sigma} = \left(\frac{\mathbf{X}^\top \mathbf{X}}{N} \right)^{-1} \left(\frac{\sum_{n=1}^N \hat{\epsilon}_n^2 \mathbf{X}_n \mathbf{X}_n^\top}{N} \right) \left(\frac{\mathbf{X}^\top \mathbf{X}}{N} \right)^{-1} \text{ and } \bar{\beta}^* = \frac{1}{B} \sum_{b=1}^B \beta_b^*.$$

- (v) Repeating steps (ii)-(iv) K times produces a sequence $\{t_k^*\}_{k=1}^K$. The test statistic is

$$T_{N,D,K,B}^* := K^{\frac{1}{2}} \sup_{z \in \mathcal{R}} \left| K^{-1} \sum_{k=1}^K \mathbf{1}_{(-\infty, z]}(t_k^*) - \Phi(z) \right|,$$

where $\Phi(\cdot)$ is the CDF of the standard normal distribution.

- (vi) The decision rule “Reject H_0 if the value of $T_{N,D,K,B}^*$ exceeds c_α ” then leads to a size- α test, where c_α is the critical value for the KS test of standard normality at significance level α .

In the testing procedure, for simplicity, we employ the Rademacher multiplier bootstrap. The proposed methodology is not restricted to this particular choice, and other multiplier bootstrap schemes may also be used under appropriate regularity conditions.

Theorem 1.2.1 Under Assumptions 1.2.1–1.2.4, if $\sqrt{K} \left(\frac{B_N^2 \log^7(DN)}{N} \right)^{1/6} = o_p(1)$ and $\frac{K}{D} = o_p(1)$, then there is

$$T_{N,D,K,B}^* \xrightarrow{d} \sup_{z \in R} |B(\Phi(z))|,$$

where $B(\cdot)$ is the Brownian bridge.

This theorem ensures that the test for estimated vectors from linear regressions has the correct size. Moreover, the test is asymptotically powerful against all fixed alternatives with low dimensions, since $t_{D,N}$ is asymptotically non-normal for fixed D . In practice, one can follow the testing procedure to obtain the corresponding statistic, if its value exceeds the critical value for the desired significance level α , the null hypothesis is rejected; otherwise, the vector is in high dimensions, which implies that the classical theories are invalid, so their modifications or the high-dimensional theories should be applied.

1.3 Test for Estimated Vectors from Nonlinear Regressions

The test statistic proposed in the previous section can be further extended to the nonlinear case. Consider the nonlinear regression model,

$$Y_n = f(\mathbf{X}_n, \beta_0) + \epsilon_n, \quad n = 1, 2, \dots, N,$$

where Y_n is the scalar response, $\mathbf{X}_n$ is a J -dimensional vector of explanatory variables, β_0 is a D -dimensional vector of unknown parameters, ϵ_n is the random error, and f is a known nonlinear function. To test the dimensionality of the D -dimensional parameter vector β_0 , besides the i.i.d. assumption, we assume that $E[\epsilon_n | \mathbf{X}_n] = 0$ and $\text{Var}(\epsilon_n | \mathbf{X}_n) = \sigma^2$ for each $n = 1, 2, \dots, N$; and $\Sigma := \mathbb{E}[\nabla f_n(\beta_0) \nabla^\top f_n(\beta_0)]$ exists with a bounded spectral norm, where $f_n(\beta)$ represents $f(\mathbf{X}_n, \beta)$. For estimation, we keep employing the LS estimator,

$$\hat{\beta} = \arg \min_{\beta} \sum_{n=1}^N (Y_n - f(\mathbf{X}_n, \beta))^2.$$

Its consistency is ensured under the following two assumptions, which hold in probability.

Assumption 1.3.1 *There are constants B and δ such that*

$$\sum_{d=1}^D \sum_{n=1}^N (\nabla f_{n,d}(\beta))^2 \leq BDN \text{ for all } \beta \text{ with } \|\beta - \beta_0\| \leq \delta,$$

where $\nabla f_{n,d}(\beta)$ is the d -th element of $\nabla f_n(\beta)$.

Assumption 1.3.2 *There are positive constants, δ_2 and c , such that*

$$\lambda_{\min} \left(\frac{1}{N} \sum_{n=1}^N \nabla f_{n,d}(\tilde{\beta}) \nabla^\top f_{n,d}(\beta) \right) > c$$

for all β with $\|\beta - \beta_0\| \leq \delta_2$ and $\tilde{\beta}$, between β_0 and β .

Assumptions 1.3.1 and 1.3.2 follow from the $X1$ assumption (Portnoy 1985), which is technically required for estimation. The consistency of $\hat{\beta}$ is obtained in the following lemma.

Lemma 1.3.1 *Under Assumptions 1.3.1 and 1.3.2, $\|\hat{\beta} - \beta_0\|^2 = O_p(\frac{D}{N})$.*

Based on the consistency, we follow the procedure in the linear case, and apply the high-dimensional CLT to the above squared norm under the following assumptions.

Assumption 1.3.3 *Suppose that there are constants b and a and a sequence of constants $B_N \geq 1$ such that*

$$\begin{aligned} (a) \quad & \frac{1}{N} \sum_{n=1}^N \mathbb{E} [t_{n,d}^2] \geq b && \text{for all } d = 1, 2, \dots, D; \\ (b) \quad & \frac{1}{N} \sum_{n=1}^N \mathbb{E} [t_{n,d}^{2+k}] \leq B_N^k && \text{for all } d = 1, 2, \dots, D \text{ and } k = 1, 2; \\ (c) \quad & \mathbb{E} \left[\exp \left(\frac{|t_{n,d}|}{B_N} \right) \right] \leq 2 && \text{for all } d = 1, 2, \dots, D \text{ and } n = 1, 2, \dots, N; \\ (d) \quad & \frac{B_N^2 \log^7(DN)}{N} = o_p(1), \end{aligned}$$

where $t_{n,d}$ is the d -th element of $\nabla f_n(\beta_0)\epsilon_n$.

Assumption 1.3.3 follows from the M and E conditions in Chernozhukov et al. (2017) to ensure the application of high-dimensional CLT.

Assumption 1.3.4 $\frac{1}{N} \sum_{n=1}^N \nabla f_n(\beta) \nabla^\top f_n(\beta)$ converges uniformly in β in an open neighborhood of β_0 with respect to the ℓ_2 -norm, which is

$$\left\| \frac{1}{N} \sum_{n=1}^N \nabla f_n(\beta) \nabla^\top f_n(\beta) - \frac{1}{N} \sum_{n=1}^N \nabla f_n(\beta_0) \nabla^\top f_n(\beta_0) \right\| = o_p(D^{-1/2})$$

for any β with $\|\beta - \beta_0\|^2 = O_p\left(\frac{D}{N}\right)$; and

$$\left\| \frac{1}{N} \sum_{n=1}^N \nabla f_n(\beta_0) \nabla^\top f_n(\beta_0) - \Sigma \right\| = o_p(D^{-1/2}).$$

Assumptions 1.3.4 and following 1.3.5 ensure the remainder term in the Taylor expansion applied for asymptotics is $o_p(1)$, which is further asymptotically negligible.

Assumption 1.3.5 For any β with $\|\beta - \beta_0\| = O_p\left(\frac{D}{N}\right)$, we have

$$\begin{aligned} \left\| \frac{1}{N} \sum_{n=1}^N [f_n(\beta_0) - f_n(\beta)] \mathbf{H}_{f_n(\beta)} \right\| &= o_p(D^{-1/2}), \\ \left\| \frac{1}{N} \sum_{n=1}^N \epsilon_n \mathbf{H}_{f_n(\beta)} \right\| &= o_p(D^{-1/2}), \end{aligned}$$

where $\mathbf{H}_{f(x)} = \nabla^2 f(x)$.

Assumptions 1.3.4 and 1.3.5 can be achieved directly under the regime where $D/N = o_p(1)$. If one requires that $D/N \rightarrow c \in (0, 1)$, these assumptions can also be achieved through constraints on the distribution of independent variables X_n and the non-linear function f . For instance, one can assume that $X_n \sim \mathcal{N}(0, N^{-1}I_D)$ in logistic or probit models (Sur & Candès 2019). Moreover, Assumptions 1.3.1, 1.3.2, 1.3.4 and 1.3.5 are formulated specifically for the LS estimator; when employing alternative estimation methods, these assumptions may require modification or could even be relaxed.

Lemma 1.3.2 Under Assumptions 1.3.1–1.3.5, there is

$$r_{D,N} := \sqrt{\frac{D}{2}} \left(\frac{N}{D} \left\| \sigma^{-1} \left(\frac{1}{N} \sum_{n=1}^N \nabla f_n(\hat{\beta}) \nabla^\top f_n(\hat{\beta}) \right)^{\frac{1}{2}} (\hat{\beta} - \beta_0) \right\|^2 - 1 \right) \xrightarrow{d} \mathcal{N}(0, 1).$$

Based on this lemma, we construct the KS-type bootstrapped test for non-linear regressions as follows:

- (i) Estimate $\hat{\beta}$ and obtain $\hat{\epsilon}_n$ where $\hat{\epsilon}_n := Y_n - f(\mathbf{X}_n, \hat{\beta})$, $n = 1, 2, \dots, N$.
- (ii) Independent of the observed data $D_n := \{(Y_n, \mathbf{X}_n) : 1 \leq n \leq N\}$, use a random number generator to generate $R_1, R_2, \dots, R_N \stackrel{iid}{\sim} \text{Rademacher}$. Define $Y_n^* := f(\mathbf{X}_n, \hat{\beta}) + R_n \hat{\epsilon}_n$, $n = 1, 2, \dots, N$.
- (iii) Re-estimate β using the data $\{(Y_n^*, \mathbf{X}_n) : 1 \leq n \leq N\}$. Denote it by β^* .
- (iv) Repeat steps (ii)-(iii) B times. Calculate the following and denote it as r_k^*

$$r_k^* := \sqrt{\frac{D}{2}} \left(\frac{N}{D} \left\| \hat{\Sigma}^{-1/2} \sqrt{B} (\hat{\beta} - \bar{\beta}^*) \right\|_2^2 - 1 \right)$$

where $\hat{\Sigma} = \left(\frac{\sum_{n=1}^N \nabla f_n(\hat{\beta}) \nabla^\top f_n(\hat{\beta})}{N} \right)^{-1} \left(\frac{\sum_{n=1}^N \hat{\epsilon}_n^2 \nabla f_n(\hat{\beta}) \nabla^\top f_n(\hat{\beta})}{N} \right) \left(\frac{\sum_{n=1}^N \nabla f_n(\hat{\beta}) \nabla^\top f_n(\hat{\beta})}{N} \right)^{-1}$,
and $\bar{\beta}^* = \frac{1}{B} \sum_{b=1}^B \beta_b^*$.

- (v) Repeating steps (ii)-(iv) K times provides a sequence of $\{r_k^*\}_{k=1}^K$. The test statistic is

$$R_{N,D,K,B}^* := K^{\frac{1}{2}} \sup_{z \in R} |K^{-1} \sum_{k=1}^K \mathbf{1}_{(-\infty, z]}(r_k^*) - \Phi(z)|,$$

where $\Phi(\cdot)$ is the CDF of the standard normal distribution.

- (vi) The decision rule “Reject H_0 if the value of $R_{N,D,K,B}^*$ exceeds c_α ” then leads to a size- α test, where c_α is the critical value for the KS test of standard normality at significance level α .

Theorem 1.3.1 *Under Assumptions 1.3.1–1.3.5, if $\sqrt{K} \left(\frac{B_N^2 \log^7(DN)}{N} \right)^{1/6} = o_p(1)$ and $\sqrt{\frac{K}{D}} = o_p(1)$, then there is*

$$R_{N,D,K,B}^* \xrightarrow{d} \sup_{z \in R} |B(\Phi(z))|,$$

where $B(\cdot)$ is the Brownian bridge.

Theorem 1.3.1 ensures the correct size of our test, which is asymptotically powerful against all fixed alternatives with low dimensions.

1.4 Simulation Studies

In this section, simulation studies are presented to illustrate the size and power of our proposed test. We constructed different designs for estimated vectors from linear regressions (see Appendix A.2 for the nonlinear case). Within each design, the dimension D is set from 2 to 100; and the sample size is set from 100 to 1000.

For linear regressions, we consider the model,

$$Y_n = \mathbf{X}_n^\top \beta_0 + \epsilon_n,$$

where $\beta_0 = (0, 0, \dots, 0)^\top$; and constructed two designs for the distributions of $\mathbf{X}_n$ and ϵ_n .

Design (i).	$\mathbf{X}_n \sim \mathcal{N}(0, I_{D \times D}),$	$\epsilon_n \sim N(0, 1).$
Design (ii).	$\mathbf{X}_{n,i} \sim \text{Cauchy}(1, 0),$	$\epsilon_n \sim t_5.$

Then, our statistic is applied to test the dimensionality of β_0 for each design. To keep the running time manageable, the corresponding results are based on 300 simulations.

For Design (i), where the independent variable $\mathbf{X}_n$ follows a normal distribution, the empirical rejection rates decrease from 1 to 0.05 when the dimension D increases to 100, which holds for all sample sizes (Table 1.1). For D/N , the empirical rejection rates decrease substantially as the ratio D/N increases (Table A.1). In particular, when $D/N = 0.01$, the rejection rates are extremely high, reaching 1.000 for $N = 100$ and $N = 200$, indicating strong power against low-dimensional settings. As D/N increases to 0.05 and 0.1, the rejection rates decline rapidly and gradually approach the nominal significance level of 0.05. When $D/N \geq 0.3$, the rejection rates remain close to 0.05 for sample sizes exceeding 300, suggesting that the proposed test achieves satisfactory size performance under high-dimensional regimes (Table A.1). Similar patterns are observed for Design (ii), where the independent variable $\mathbf{X}_n$ follows a Cauchy distribution. Overall, the results indicate that the proposed test exhibits good empirical size and power for estimated vectors even in small to moderate sample sizes.

Table 1.1: **Empirical rejection rates for estimated vectors.**

The columns report the fraction of simulations for which the p-value is less than 0.05 with different sample sizes.

D	Design (i)					Design (ii)				
	Sample size N					Sample size N				
	100	200	300	500	1000	100	200	300	500	1000
2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
3	0.607	0.597	0.597	0.600	0.623	0.580	0.550	0.577	0.580	0.593
4	0.467	0.450	0.427	0.470	0.460	0.440	0.403	0.427	0.463	0.490
5	0.387	0.357	0.390	0.363	0.407	0.387	0.347	0.357	0.433	0.307
6	0.297	0.300	0.287	0.313	0.287	0.303	0.297	0.303	0.333	0.287
7	0.293	0.293	0.260	0.260	0.243	0.280	0.253	0.230	0.313	0.273
8	0.213	0.227	0.280	0.227	0.193	0.230	0.237	0.233	0.297	0.283
9	0.250	0.210	0.207	0.250	0.250	0.217	0.250	0.203	0.193	0.267
10	0.223	0.207	0.200	0.190	0.203	0.163	0.183	0.227	0.200	0.237
20	0.160	0.156	0.160	0.110	0.117	0.137	0.103	0.100	0.090	0.123
30	0.133	0.090	0.127	0.110	0.130	0.103	0.110	0.107	0.067	0.087
40	0.080	0.087	0.087	0.090	0.080	0.090	0.093	0.070	0.090	0.087
50	0.080	0.043	0.060	0.083	0.113	0.110	0.073	0.063	0.103	0.087
60	0.077	0.097	0.070	0.100	0.087	0.113	0.073	0.057	0.077	0.070
70	0.070	0.080	0.060	0.070	0.073	0.050	0.073	0.070	0.067	0.053
80	0.073	0.077	0.070	0.067	0.050	0.080	0.070	0.063	0.080	0.067
90	0.050	0.073	0.063	0.050	0.047	0.053	0.050	0.067	0.043	0.060
100	-	0.050	0.050	0.047	0.057	-	0.053	0.053	0.057	0.053

Table 1.2: **Empirical rejection rates for estimated vectors with different D/N .**

D/N	Design (i)					Design (ii)				
	Sample size N					Sample size N				
	100	200	300	500	1000	100	200	300	500	1000
0.01	1.000	1.000	0.597	0.363	0.203	1.000	1.000	0.577	0.433	0.237
0.05	0.387	0.207	0.123	0.100	0.113	0.387	0.183	0.130	0.087	0.087
0.1	0.223	0.156	0.127	0.083	0.057	0.163	0.103	0.107	0.103	0.053
0.3	0.133	0.097	0.063	0.083	0.060	0.103	0.073	0.067	0.053	0.043
0.5	0.080	0.050	0.057	0.067	0.053	0.110	0.053	0.057	0.050	0.047

1.4.1 Classical Theories in High Dimensions

In this subsection, the F -test and modified F -test (Calhoun 2011) which have been briefly introduced, are examined carefully to show that our proposed test is consistent with their validity in high dimensions based on some simulation experiments. We check the validity of the F -test and modified F -test in the model

$$y_n = \beta_0 + \beta^\top x_n + \epsilon_n,$$

where β_0 and β are all zeros; each element of x_n is independently drawn from the distribution $\text{Cauchy}(1, 0)$, and ϵ_n is drawn from Student's t distribution with degree 5 for $n = 1, 2, \dots, N$. These two tests were applied to test the null hypothesis $H_0 : \beta = 0$ by setting $N = 100, 200, \dots, 500$ and $D = 2, 3, \dots, 100$; and then the empirical rejection rates were calculated at significance level 5% through 1000 repetitions. Table 1.3 shows that the F -test over-rejects as the dimension increases. The result of Design (ii), which has the same settings as the above model, shows that when the empirical rejection rates of our test converge to 0.05, the F -test over-rejects and the corresponding empirical rejection rates even increase up to 0.1 (Tables 1.1 & 1.3). On the other hand, the modified F -test exhibits substantial over-rejection in small-sample, high-dimensional settings, with an empirical rejection rate of 0.114 when $D = 60$ and $N = 100$. Therefore, it is less stable than the classical F -test in small-sample settings, although its empirical rejection rates approach the nominal size of 0.05 for moderate sample sizes. Overall, the unsatisfactory performance exhibited by these two tests in different scenarios highlights the necessity of our testing procedure.

Table 1.3: **Empirical rejection rates of the F -test and modified F -test.**

The columns report the fraction of simulations for which the p-value is less than 0.05 with different sample sizes.

D	F -test					Modified F -test				
	Sample size N					Sample size N				
	100	200	300	500	1000	100	200	300	500	1000
2	0.061	0.066	0.055	0.044	0.062	0.060	0.066	0.057	0.047	0.055
3	0.057	0.057	0.056	0.058	0.048	0.050	0.051	0.047	0.051	0.050
4	0.076	0.063	0.063	0.057	0.076	0.071	0.046	0.054	0.056	0.052
5	0.068	0.071	0.064	0.085	0.071	0.060	0.040	0.054	0.056	0.042
6	0.062	0.064	0.051	0.057	0.081	0.054	0.050	0.044	0.053	0.046
7	0.076	0.059	0.066	0.081	0.065	0.054	0.061	0.062	0.044	0.043
8	0.088	0.073	0.072	0.072	0.070	0.049	0.056	0.044	0.043	0.050
9	0.080	0.070	0.089	0.083	0.066	0.052	0.047	0.065	0.031	0.047
10	0.087	0.071	0.074	0.093	0.075	0.066	0.063	0.056	0.043	0.055
20	0.078	0.082	0.093	0.081	0.095	0.058	0.055	0.058	0.052	0.044
30	0.071	0.089	0.085	0.112	0.073	0.066	0.048	0.050	0.060	0.051
40	0.071	0.080	0.096	0.088	0.098	0.061	0.057	0.048	0.063	0.052
50	0.096	0.083	0.094	0.075	0.101	0.061	0.052	0.049	0.050	0.048
60	0.087	0.083	0.098	0.096	0.100	0.085	0.046	0.046	0.054	0.054
70	0.081	0.088	0.087	0.102	0.097	0.114	0.054	0.051	0.053	0.049
80	0.068	0.084	0.101	0.096	0.099	-	0.040	0.046	0.050	0.052
90	0.064	0.092	0.106	0.105	0.110	-	0.050	0.045	0.047	0.054
100	-	0.104	0.117	0.114	0.100	-	0.052	0.050	0.050	0.064

1.5 Conclusion

We have shown how to test for dimensionality of estimated vectors from both linear and nonlinear regressions, providing scholars with a methodological tool to determine whether classical or high-dimensional theories are valid for given data samples. Simulation results suggest that our test statistic has good size and power in small to moderate sample sizes. Since our testing framework is compatible with a broad class of estimators, adopting alternative estimators could further enhance the scope of our test.

Chapter 2

Network Estimation under Shape Restrictions

2.1 Introduction

Incorporating domain-specific priors or expert-derived knowledge into neural networks has emerged as a methodology with broad applicability, achieving widespread adoption across numerous fields (Liu, Zhang & Liu 2024, Fu et al. 2023, Gao et al. 2023, Chen et al. 2023, Lang et al. 2023, Yang & Fang 2023, Cole & Williamson 2019, Archer & Wang 1993, Liu et al. 2020, Minin et al. 2010, Feelders 2000). For example, the prior knowledge about the scattering centers and target-aspects has been utilized for radar target recognition (Liu, Zhang & Liu 2024); and the knowledge about the mapping relationship between medications and diagnoses has been used for medication combination prediction (Gao et al. 2023). Most of the prior knowledge are encodable and can be directly embedded into neural networks as additional information; however, the incorporation of abstract/non-encodable knowledge, which is equally critical, continues to pose fundamental implementation challenges.

The abstract prior knowledge predominantly encompasses fundamental structural properties governing functional relationships, including but not limited to monotonicity, convexity/concavity requirements with respect to specified input variables, and homogeneity conditions. These axiomatic regularities serve as critical constraints, for example, in economic models and biological systems analysis. Canonical applications manifest in economic theory through the imposition of monotonicity in utility functions and homogeneity in production system specifications, while biological applications typically impose monotonicity

constraints on growth functions with respect to some critical parameters. This kind of prior knowledge is too abstract to be encoded directly into neural networks, but they cannot be violated in many real-world scenarios. When a neural network is trained to approximate these functions with shape constraints based on raw data, the prescribed constraints cannot be ensured, since the noise present in the data may result in deviations from the intended shape during training.

Existing approaches for incorporating shape constraints can be grouped into two categories: penalty-based regularization and architectural constrained formulation. The former approach is implemented through the modification of loss functions with additional constraint terms during neural network optimization. Specifically, these modified loss functions linearly combine the classical loss function, such as the mean squared error loss, and an additional loss function which penalizes the violation of the required shape. However, minimizing these modified loss function cannot ensure that the penalty loss is zero, thus the required shape cannot be guaranteed. On the other hand, the latter category achieves strict enforcement of shape requirements through dedicated architectural designs. In contrast to conventional network architectures, these designs utilize some special connections between layers and even limit the number or types of layers to prespecified forms, confining the hypothesis class exclusively to shape-restricted functions (Sill 1997, Bamberger et al. 2023, Warin 2023, Amos et al. 2017, Calafiore et al. 2020, Tang et al. 2020). For example, the monotonic MIN-MAX neural network (Sill 1997) employs the combination of minimization and maximization to connect layers which guarantees that the network output is monotonic.

While these architectural designs successfully embed the required shape, their practical adoption faces three challenges: inconvenient implementation, time inefficiency and limited generalization capability. The first challenge stems from the divergence in forward propagation mechanisms between these shape-restricted networks and traditional convolutional architectures, which complicates the structural implementation. The architectural reconstruction with limited number or types of layers does not only require retraining, which is time consuming, but also leads limited generalization capability. Specifically, existing transfer learning techniques or methods aimed at enhancing model generalization (Hosna et al. 2022, Ribani & Marengoni 2019, Tan et al. 2018) encounter fundamental barriers within these shape-restricted frameworks, as their architectural constraints fundamentally restrict structural compatibility and parameter adaptability across different tasks. On the other hand, a lack of large sample properties for the existing shape-restricted networks results in infeasible implementation of many statistical inference methods. Many scholars have shed light on the properties of neural networks, such as the universal approximation property, but the corresponding large sample behaviors remain critically understudied,

which implies that these networks, especially with multiple hidden layers, are still like black-box systems.

This paper proposes a flexible shape-constrained correction method which can be applied to general neural networks without changing the original architecture to produce the shape-constrained estimation. Based on the monotonizing procedure (Dette et al. 2006, Dette & Scheder 2006, Birk & Dette 2007) and Euler’s homogeneous function theorem, our method applies a trained neural network as the preliminary estimate which is further corrected to satisfy different shape constraints simultaneously. More importantly, the proposed method preserves, to a considerable extent, the convergence rate of the preliminary network, and is shown to be asymptotically normal under mild regularity conditions. Compared to other shape-restricted networks, our method achieves comparable and even better performance based on some case studies.

Methods

Let $\mathcal{T} : \mathcal{X} \subseteq \mathcal{R}^D \rightarrow \mathcal{R}$ denote a neural network operating from the input space $\mathcal{X}$ to a scalar output space. In supervised learning frameworks (e.g., classification), given observational pairs (X, y) where $X = \{x_1, x_2, \dots, x_D\}$ represents the explanatory variables and y the response variable, the network aims to approximate an unknown target function $f_0 : \mathcal{X} \rightarrow \mathcal{R}$ such that $f_0(X) = E[y|X]$ governing the data generating process. In this paper, the input value is normalized to the interval $[0, 1]$, that is $\mathcal{X} = [0, 1]^D$, as exemplified by min-max or z-score normalization widely adopted in practice. Without any prior knowledge, the neural network $\mathcal{T}$ with initial weights is trained and fitted to the observed sample $\{y_i, X_i\}_{i=1}^n$. Given the prior knowledge about shapes of the function f_0 , our method achieves the shape-restricted estimation based on this trained network $\mathcal{T}$.

Before introducing the method, we need to formally define the shape. Let S be a list of pairs representing the shape, such that,

$$S = \{(r_1, s_1), (r_2, s_2), \dots, (r_P, s_P)\},$$

where $r_p \in \{\text{mon}, \text{con}, \text{hom}\}$. “Mon”, “con” and “hom” represent strictly monotonic, strictly convex and homogeneous, respectively. If $r_p \in \{\text{mon}, \text{con}\}$, s_p indicates the index of the corresponding input coordinate. If $r_p = \text{hom}$, s_p is degree of the homogeneity. Thus, S is a combination of different shapes on some input coordinates; and each coordinate can only have at most one shape. Therefore, there is $P \leq D + 1$ (All coordinates are shape-constrained, and homogeneity is imposed). Since the proposed shape-restricted estimators

for strictly increasing and decreasing are identical fundamentally, we only consider the case of strict increasing in the rest of the paper. In practice, “increasing” or “decreasing” should be specified.

The general procedure of our method requires two inputs, the shape, S , and a trained network, $\mathcal{T}$. Suppose that the underlying true function f_0 has the shape,

$$S = \{(r_1, s_1), (r_2, s_2), \dots, (r_P, s_P)\}.$$

For a trained network $\mathcal{T}$, each shape constraint (r_p, s_p) is iteratively imposed to it, as follows.

Step 1: For (r_1, s_1) , define

$$H_1(X; S, \mathcal{T}) := \mathbf{M}_{r_1}^{s_1}(X; \mathcal{T}).$$

$H_1(X; S, \mathcal{T})$ is the approximator which satisfies the shape constraint r_1 .

Step 2: For (r_2, s_2) , based on the result from Step 1, define

$$H_2(X; S, \mathcal{T}) := \mathbf{M}_{r_2}^{s_2}(X; H_1).$$

$H_2(X; S, \mathcal{T})$ satisfies both r_1 and r_2 simultaneously.

.....

Step P: For (r_P, s_P) , based on the result from Step $P - 1$, similarly define

$$H_P(X; S, \mathcal{T}) := \mathbf{M}_{r_P}^{s_P}(X; H_{P-1}).$$

$H_P(X; S, \mathcal{T})$ is the approximator which satisfies all shape constraints for the corresponding coordinates.

We can also write the above procedure as $\mathbf{M}_{r_P}^{s_P} \circ \mathbf{M}_{r_{P-1}}^{s_{P-1}} \circ \dots \circ \mathbf{M}_{r_2}^{s_2} \circ \mathbf{M}_{r_1}^{s_1}(X; \mathcal{T}) =: H(X; S, \mathcal{T})$. It is important to note that the final estimator can satisfy all shape requirements simultaneously regardless of the order of the steps (see Appendix B). However, changing the order of the steps does not yield the same value in finite samples. Each $\mathbf{M}_{r_p}^{s_p}$ is an approach which makes a given function satisfy the shape r_p with respect to the corresponding coordinate indexed by s_p . More importantly, our approximator is non-parametric, since it is based on the original unrestricted networks, which are regarded as non-parametric approximators (Horel & Giesecke 2020), and only make the shape-constrained correction. Different from our method, as mentioned, the existing shape-constrained networks have limited architectures, which implies that these approximators are parametric. Although it

is shown that these approximators with proper number of neuron nodes can be arbitrarily close to the underlying function with certain shapes, how to choose the proper number of nodes is still challenging. Specifically, insufficient nodes may not represent the underlying function, which is discussed in case studies.

The following subsections present the approaches, $\mathbf{M}_{\text{mon}}$, $\mathbf{M}_{\text{con}}$ and $\mathbf{M}_{\text{hom}}$, for monotonicity, convexity and homogeneity in details.

2.1.1 Monotonicity

The method of achieving monotonicity is motivated by the isotonization procedure (Dette et al. 2006, Dette & Scheder 2006), which is different from reconstructing loss functions or architectures (Archer & Wang 1993, Liu et al. 2020, Minin et al. 2010, Sill 1997, Runje & Shankaranarayana 2023, Gupta et al. 2019, Daniels & Velikova 2010, Sill & Abu-Mostafa 1996). This method applies an unconstrained estimate of a function as the preliminary estimate which is monotonized in each of the specified coordinates without changing the monotonicity of other coordinates. Our method utilizes the monotonizing procedure (Dette et al. 2006) to neural networks.

For a network $\mathcal{T} : [0, 1]^D \rightarrow \mathbf{R}$, if it is required to satisfy the shape restriction (mon, i) , the corresponding approach $\mathbf{M}_{\text{mon}}^i$ is the following:

$$\begin{aligned} \mathbf{M}^{-1}(y; \mathcal{T}) &= \frac{1}{Mh_d} \int_{-\infty}^y \sum_{m=1}^M K \left(\frac{\mathcal{T}(x_1, x_2, \dots, x_{i-1}, \frac{m}{M}, x_{i+1}, \dots, x_D) - u}{h_d} \right) du \\ &= \int_0^1 \mathbf{I}_{\{\mathcal{T}(x_1, x_2, \dots, x_{i-1}, z, x_{i+1}, \dots, x_D) \leq y\}} dz + o_p(1), \end{aligned}$$

where K is a kernel function and h_d is a bandwidth. Then the inverse of the above strictly increasing (monotonic) function $\mathbf{M}^{-1}(y; \mathcal{T})$ is calculated and denoted by $\mathbf{M}_{\text{mon}}^i(X; \mathcal{T})$ which is also strictly increasing with respect to the i^{th} coordinate (Dette et al. 2006, Dette & Scheder 2006). Note that there are two formulas presented above, but their difference converges to 0 in probability. In practice, the latter one is more computationally easier by simplifying the indicator function in the integral. However, the corresponding large sample properties are investigated based on the kernel one in this chapter.

2.1.2 Convexity

The proposed approach for convexity is monotonizing the derivative of an estimate. When a function is continuously differentiable, it is strictly convex if and only if its derivative is

strictly increasing. Therefore, in order to obtain the convex estimate based on a given neural network, we first construct the monotonic estimate of its derivative, and then integrate it. This requires that the neural network is continuously differentiable, which implies that the activation function should be continuously differentiable due to the chain rule. Based on the method from Birk & Dette (2007), for a network $\mathcal{T} : [0, 1]^D \rightarrow \mathbf{R}$, the approach $\mathbf{M}_{\text{con}}^i$ for a network is the following:

Firstly, for each $X_{-i} = \{x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_D\} \in [0, 1]^{D-1}$, define

$$\begin{aligned} \mathbf{C}^{-1}(y|X_{-i}; \mathcal{T}) &= \frac{1}{Mh_d} \int_{-\infty}^t \sum_{m=1}^M K \left(\frac{\frac{\partial \mathcal{T}}{\partial x_i}(x_1, x_2, \dots, x_{i-1}, \frac{m}{M}, x_{i+1}, \dots, x_D) - u}{h_d} \right) du \\ &= \int_0^1 \mathbf{I}_{\left\{ \frac{\partial \mathcal{T}}{\partial x_i}(x_1, x_2, \dots, x_{i-1}, z, x_{i+1}, \dots, x_D) \leq y \right\}} dz + o_p(1), \end{aligned}$$

where K is a kernel function and h_d is a bandwidth. Then calculate the inverse of the above strictly increasing function, and denote it as $\mathbf{C}(z|X_{-i}; \mathcal{T})$.

Secondly, define

$$\mathbf{M}_{\text{con}}^i(X; \mathcal{T}) = \mathcal{T}(x_1, x_2, \dots, x_{i-1}, u_0, x_{i+1}, \dots, x_D) + \int_{u_0}^{x_i} \mathbf{C}(z|X_{-i}; \mathcal{T}) dz$$

for any fixed $u_0 \in (0, 1)$. The estimate $\mathbf{M}_{\text{con}}^i(X; \mathcal{T})$ is strictly convex with respect to the i^{th} coordinate because its derivative $\mathbf{C}(z|X_{-i}; \mathcal{T})$ is strictly increasing by the construction in the first step (Birk & Dette 2007). This method can be extended to the case of concavity directly, since we only need to monotonize the corresponding derivative to decreasing.

2.1.3 Homogeneity

While monotonicity and convexity dominate the literature of shape-restricted estimation, homogeneity holds fundamental significance for some classical functions, such as neoclassical production functions in economics, and the solutions to many graph problems (Tang et al. 2020). Some statistical methods without utilizing neural networks have been proposed, but these methods cannot strictly achieve homogeneous estimation (Tripathi 1997, Tripathi & Kim 2003). On the other hand, graph neural networks (GNNs) which are modified to be a universal approximator of homogeneous functions (Tang et al. 2020) only works for degree one. Therefore, current approaches either failing to ensure homogeneous estimation or being limited to homogeneity degree one. Our method for homogeneity is motivated by Euler's homogeneous function theorem. If a function g is homogeneous of

degree r , define $h(\mathbf{s}) := g(\mathbf{s}\mathbf{X})$. Based on Euler's homogeneous function theorem, there is $\mathbf{s}h'(\mathbf{s}) = rh(\mathbf{s})$. As a result, the solutions of this linear differential equation have the form $h(\mathbf{s}) = C\mathbf{s}^r$, which can be used to obtain a homogeneous estimate. For a given input X , the general procedure is to fit the model $C\mathbf{s}^r$ with parameter C to a function $g(\mathbf{s}X)$. If the prior knowledge is that the unknown function f_0 is homogeneous of degree r , the approach $\mathbf{M}_{\text{hom}}^r$ is the following:

For a given $X = \{x_1, x_2, \dots, x_D\} \in [0, 1]^D$, find the best $\hat{C}$, such that $\hat{C}\mathbf{s}^r$ is close to $\mathcal{T}(\mathbf{s}X)$. That is

$$\mathbf{M}_{\text{hom}}^r(X; \mathcal{T}) := \hat{C} = \arg \min_C \sum_{j=1}^J (Cs_j^r - \mathcal{T}(s_j X))^2,$$

where the sequence $\{s_j\}_{j=1}^J$ is such that $s_j X \in [0, 1]^D$.

This method, intuitively, uses the model $C\mathbf{s}^r$ fits the preliminary network estimation $\mathcal{T}$ along the direction X , which is equivalent to that along the direction aX with $a > 0$, so the estimation is homogeneous of degree r (see Appendix B). However, there is still one exception. If $r > 0$, this method cannot ensure $\mathbf{M}_{\text{hom}}(0; \mathcal{T}) = 0$, because $\mathcal{T}(0) = 0$ is not guaranteed. Therefore, we define $\mathbf{M}_{\text{hom}}(0; \mathcal{T}) = 0$ for $r > 0$, additionally.

2.2 Large Sample Properties

The proposed estimator corrects the network-based estimation and preserves, to a considerable extent, the convergence properties of the original network. Under some mild regularity conditions, our estimator is shown to be asymptotically normal. In the next two sections, the rate of convergence and asymptotic normality properties are discussed in detail.

2.2.1 Rate of Convergence

We begin our asymptotic analysis by characterizing the rate of convergence of the proposed methods for three different shapes. In particular, we demonstrate that, our estimator achieves a convergence rate depending on the preliminary neural network. The following theorem is considered under the case where the input dimension is D and the shape constraint is imposed on the first coordinate.

Theorem 2.2.1 *If the monotonicity (i.e. increasing with respect to the first coordinate) is correctly specified, the underlying true function f_0 and the preliminary (network-based) estimator $\hat{f}$ satisfy:*

(1.a) f_0 is strictly increasing, and its partial derivative with respect to the first coordinate is bounded above and below by positive constants;

(1.b) The first derivative of $\hat{f}^{-1}$ is bounded above and below;

(1.c) $\int_{[0,1]^{D-1}} \left(\int_0^1 \left| \hat{f}(x_1, X_{-1}) - f_0(x_1, X_{-1}) \right|^p dx_1 \right) p_{-1}(X_{-1}) dX_{-1} = O \left(\left\| \hat{f} - f_0 \right\|_{L^p(\mu)} \right);$

then we have

$$\left\| \mathbf{M}_{mon}(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} = O \left(\left\| \hat{f} - f_0 \right\|_{L^p(\mu)}^{\frac{p}{p+1}} \right).$$

In 1.c, $p_{-1}(X_{-1})$ is the conditional density function of $X_2, \dots, X_D$ conditional on X_1 . The μ is a measure with which the sample X is drawn from the distribution associated, and $p_{-1}(X_{-1})$ is the joint marginal probability density function of X_{-1} . Theorem 1 shows that if the original network-based estimator converges at order C , after incorporating our correction, the convergence rate becomes $C^{\frac{p}{p+1}}$; and further monotonic correction (for example, on the second coordinate) will induce an additional reduction on this rate. Researchers typically focus on the convergence rate of the L^2 -norm of neural networks, for example, the order $O \left((\log(n))^6 \max_{(P,K) \in \mathcal{P}} n^{-\frac{2P}{2P+K}} \right)$ where $\mathcal{P}$ is the smoothness constraint (Kohler & Langer 2021), then the rate of convergence is diminished to the original rate raised to the $\frac{2}{3}$ power. This result differs from that in the classical monotonization method (Dette et al. 2006, Dette & Scheder 2006), where the root- n consistency is constrained by the choice of the preliminary estimator, namely the classical kernel estimator. Similar result can be found in the method of convexity.

Theorem 2.2.2 *If the convexity/concavity is correctly specified, the underlying true function f_0 and the preliminary (network-based) estimator $\hat{f}$ satisfy:*

(2.a) f_0 is strictly convex, and its second partial derivative with respect to the first coordinate is bounded above and below by positive constants;

(2.b) The first derivative of $\hat{f}'^{-1}$ is bounded above and below;

$$(2.c) \quad \left\| \hat{f}(u_0, X_{-1}) - f(u_0, X_{-1}) \right\|_{L^p(\mu)} = O \left(\left\| \hat{f} - f_0 \right\|_{L^p(\mu)}^{\frac{p}{p+1}} \right);$$

$$(2.d) \quad \int_{[0,1]^{D-1}} \left(\int_0^1 \left| \hat{f}'(x_1, X_{-1}) - f_0'(x_1, X_{-1}) \right|^p dx_1 \right) p_{-1}(X_{-1}) dX_{-1} = O \left(\left\| \hat{f} - f_0 \right\|_{L^p(\mu)} \right);$$

then we have

$$\left\| \mathbf{M}_{con}(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} = O \left(\left\| \hat{f} - f_0 \right\|_{L^p(\mu)}^{\frac{p}{p+1}} \right).$$

Since the convex procedure monotonizes the derivative of the preliminary estimator, under suitable conditions on its partial derivative and on the underlying true function, this method can also achieve the same $\frac{p}{p+1}$ power reduction as the monotonic case. On the other hand, the homogeneity method can fully preserve the convergence rate of the preliminary estimator, as demonstrated below.

Theorem 2.2.3 *If the homogeneity is correctly specified, the underlying true function f_0 and the preliminary (network-based) estimator $\hat{f}$ satisfy:*

$$(3.a) \quad \text{for any non-negative } a, \text{ there is a constant } c, \text{ such that } \left\| \hat{f}(aX) - f_0(aX) \right\|_{L^p(\mu)} \leq c \left\| \hat{f}(X) - f_0(X) \right\|_{L^p(\mu)};$$

$$(3.b) \quad \text{the non-negative sequence } \{s_t\}_{t=1}^T \text{ satisfies } \sum_{t=1}^T s_t^r / \sum_{t=1}^T s_t^{2r} \leq c \text{ for some constant } c > 0;$$

then there is

$$\left\| \mathbf{M}_{hom}^r(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} = O \left(\left\| \hat{f}(X) - f_0(X) \right\|_{L^p(\mu)} \right).$$

Theorem 2.2.3 implies that the homogeneity method holds the same convergence rate, which is achieved by strong assumptions on $\hat{f}$ and the choice of $\{s_t\}_{t=1}^T$.

Our proposed method serves as a post-processing procedure, rather than an alternative training architecture, so it still preserves some of the fundamental advantages of neural network architectures, including efficient compositional structures approximation, mitigating the curse of dimensionality, and efficient parameter optimization via backpropagation (Kohler & Langer 2021, Schmidt-Hieber 2020). Since our shape correction is applied only

at the output level without modifying the parameters or weights of the preliminary model, both the learned compositional representation and the backpropagation-based training mechanism remain fully preserved. The only cost is a moderate reduction in the convergence rate, which is the statistical price of enforcing shape constraints.

It is important to note that the shape-corrected estimator derived from the proposed method will diverge when the shape constraints are misspecified, which highlights the importance of reliable prior knowledge, since incorrect priors can negatively impact the estimation. On the other hand, if the preliminary estimator has captured the required shapes, our method will not modify the preliminary estimation.

2.2.2 Asymptotic Normality

Although the rate of convergence can be analyzed for general neural networks, rigorous investigation of the asymptotic normality property necessitates imposing specific regularity conditions on both the data-generating distribution and structural configurations of the network. Two significant requirements involve restricting all activation functions to polynomial forms, and limiting the number of trainable weights.

Assumption 2.2.1 *Activation functions in the preliminary network are polynomial.*

To approximate the function f_0 , we consider the deep neural network as shown in Figure 2.1. This network contains L hidden layers and a set of width $\mathbf{d} := (d_1, d_2, \dots, d_L)$ for each hidden layer. Specifically, d_l is the number of neural nodes in the hidden layer l . Since all activation functions are polynomials, a neural network is still a polynomial, and all of the coefficients are composited by its weights, which further implies that we can define the degree of the network as following.

Definition 2.2.1 *For a deep neural network, $\mathcal{T}$, which contains L hidden layers and a set of width $\mathbf{d} := (d_1, d_2, \dots, d_L)$ for each hidden layer, define the degree of $\mathcal{T}$ is*

$$\deg(\mathcal{T}) := \prod_{l=1}^L \max_{i=1, \dots, d_l} \deg(\sigma_l^{(i)}),$$

where $\sigma_l^{(i)}$ is the activation function at the i th node in the hidden layer l (node $H_l^{(i)}$).

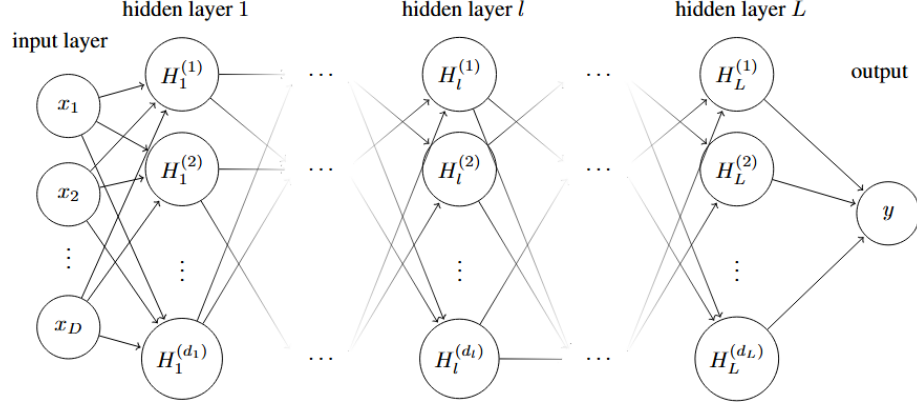


Figure 2.1: **Deep Neural Network.**

Before establishing the asymptotics of our shape-restricted estimator, we recall the traditional series estimator. For a series estimator, consider a vector of approximating functions

$$p^K(X) = (p_{1K}(X), \dots, p_{KK}(X))',$$

whose linear combination can approximate $f_0(X)$. Then a series estimator of $f_0(X)$ is

$$\begin{aligned}\hat{f}(X) &= p^K(X)' \hat{\beta}, \\ \hat{\beta} &= (P'P)^- P'Y,\end{aligned}$$

where $P = [p^K(X_1), \dots, p^K(X_n)]'$, $Y = [y_1, \dots, y_n]'$ and $(P'P)^-$ is the Moore-Penrose Generalized inverse of $P'P$. One particular series is the power series. Let $\lambda = (\lambda_1, \dots, \lambda_D)$ be a vector of non-negative integers, i.e. a multi-index, with norm $|\lambda| = \sum_{d=1}^D \lambda_d$, let $X^\lambda = \prod_{d=1}^D x_d^{\lambda_d}$. For a sequence of distinct such vectors, a power series approximation has $p_{kK}(X) = X^{\lambda^{(k)}}$. For example, if $D = 2$ and $K = 6$, then $p^6(x) = (1, x_1, x_2, x_1x_2, x_1^2, x_2^2)'$. Then, a deep neural network with polynomial activation functions can be represented as the form which is similar to the series estimator.

Lemma 2.2.1 *For a deep neural network, $\mathcal{T}$, with degree G , if there are $\bar{K} = \binom{G+D}{D}$ nodes in the last hidden layer, $\mathcal{T}(X)$ can be represented as*

$$\mathcal{T}(X) = p^{\bar{K}}(X)' \mathbf{W} \mathbf{V}^{\bar{K}},$$

where $\mathbf{W}$ is a $\bar{K} \times \bar{K}$ matrix whose entries are composed of the corresponding polynomials of all weights except v_k for $k = 1, 2, \dots, \bar{K}$. Here $V^{\bar{K}} = \{v_1, v_2, \dots, v_{\bar{K}}\}$ is a vector of $\bar{K}$ weights between the last hidden layer and the output.

Assumption 2.2.2 $(y_1, X_1), \dots, (y_n, X_n)$ are i.i.d. and $\text{Var}(y|X)$ is bounded.

Assumption 2.2.3 $E[(y - f_0(X))^4]$ is bounded and $\text{Var}(y|X)$ is bounded away from zero.

Assumptions 2.2.2 and 2.2.3 indicate that we will work with i.i.d. (independent and identically distributed) samples; and some moment conditions are required. Under Assumption 2.2.2, the asymptotics of neural networks can be investigated as the series estimator, as in the following lemma.

Lemma 2.2.2 Under Assumption 2.2.2, suppose that a deep network, $\mathcal{T}$, has degree G , then for any fixed weights except $\bar{K}$ weights after the last hidden layer, if either the matrix $\mathbf{W}$ is orthonormal or the matrices $\mathbf{W}$ and $P'P$ are invertible, the network estimator obtained by minimizing the mean squared loss is equivalent, in terms of point estimation, to a power series estimator.

The number of weights after the last hidden layer is not required to equal $\bar{K}$ which serves as the lower bound for Lemma 2.2.2; however, the equivalence necessitates exact $\bar{K}$ trainable weights (see Appendix B). Then, the asymptotics of neural networks can be established under the following assumptions.

Assumption 2.2.4 The support of X is a Cartesian product of compact connected intervals on which X has a probability density function that is bounded away from zero.

Assumption 2.2.5 The underlying function $f_0(X)$ is continuously differentiable of order s on $[0, 1]^D$.

Assumption 2.2.5 follows from the assumption by Newey (1997), that the approximation rate depends on s/D , which is related to the smoothness of the function $f_0(X)$ and the dimensionality of X . When $f_0(X)$ is analytic, s can be an arbitrarily large positive number.

Theorem 2.2.4 *Under Assumptions 2.2.1–2.2.5, suppose that a deep neural network, $\mathcal{T}$, has degree G and the underlying true function f_0 has the shape S , for any fixed weights except $\bar{K}$ weights after the last hidden layer, if either the matrix $\mathbf{W}$ is orthonormal or the matrices $\mathbf{W}$ and $P'P$ are invertible, $\sqrt{n}\bar{K}^{-s/D}$, $\bar{K}^6/n \rightarrow 0$ give that our shape-restricted approximator $H(X, S, \hat{\mathcal{T}})$ based on the unrestricted network estimator, $\hat{\mathcal{T}}$, is asymptotically normal for fixed X , that is*

$$\sqrt{n}\hat{V}^{-1/2} \left(H(X, S, \hat{\mathcal{T}}) - f_0(X) \right) \xrightarrow{d} \mathcal{N}(0, 1),$$

where $\hat{V}$ depends on the architecture of the network and the shape requirement S .

Theorem 2.2.4 provides the asymptotic normality of the network-based shape-restricted estimator; therefore, analogous to conventional nonparametric estimation, hypothesis testing frameworks or point-wise confidence intervals can be formally derived for our estimator. All of the restrictions on the activation functions or the number of trainable weights are primarily serve as theoretical tools to ensure the identifiability and enable the rigorous derivation of normality results, rather than a limitation for practical implementation. Similar ideas appear in some classical methods. For example, while trimmed functions are often used to control extreme values and simplify theoretical analysis, in practice, they are not always strictly applied; alternative approaches such as robust estimation or regularization are typically sufficient. In quantile regressions, the introduction of a kink in the check function is crucial for theoretical derivations, however, this modification is not strictly needed in practice.

2.3 Experiments

The following two subsections illustrate the performance of proposed method through simulation and case studies. In the simulation studies, four functions with distinct shapes were constructed as the underlying true functions. Three of them are monotonic, convex and homogeneous, respectively, and the fourth is both monotonic and homogeneous. For each case, a fully connected network containing 3 hidden layers (with 32, 16, 8 neurons, respectively) was trained as the benchmark¹, and then was imposed the corresponding shape constraints via our method. We also compare our method with the Deep Lattice Network (DLN) (You et al. 2017) for monotonicity, the Input Convex Neural Network

¹We used the Adam optimizer with a learning rate of 10^{-3} . The model is trained in 1000 epochs.

(ICNN) (Amos et al. 2017) for convexity, and the neural networks using RePU activation functions for homogeneity. In the case studies, the proposed method was empirically evaluated on two datasets.

Simulation Studies

We apply the proposed network estimation for the following three functions at first.

$$\begin{aligned} f(x_1, x_2) &= 2x_1^2 I_{\{x_1 \geq 0.5\}} + x_1 I_{\{x_1 < 0.5\}} - 3|x_2 - 0.5|, \\ f(x_1, x_2) &= e^{x_1} - |x_2 - 0.5|, \\ f(x_1, x_2) &= 5\sqrt{x_1 x_2}, \end{aligned}$$

where each input x_i is uniformly distributed on the interval $[0, 1]$. Moreover, errors following standard normal distribution are added. The first function is increasing with respect to the first coordinate, and it contains linear and non-linear part on the interval $[0, 0.5]$ and $[0.5, 1]$, respectively. The second one is convex with respect to the first coordinate, while the last one is homogeneous of degree one. For each function, we set multiple sample sizes, 100, 200, 300, 400, 500, 1000 and 5000 for training, and generate a set with size 500 for testing. Table 2.1, 2.2 and 2.3 present the corresponding results based on 500 simulations.

In Table 2.1, 97.4% of classical neural networks are not monotonic when the sample size is 100. This proportion increases when the sample size rises from 100 to 300 but eventually decreases as the sample size rises to 5000, which implies that the classical neural network (ReLU activation functions) without embedding any shape constraints cannot achieve the true shape of the underlying function, and increasing the sample size cannot resolve this problem. Compared to the benchmark and DLN, the proposed method has lower MSE on the test set. This difference between the benchmark/DLN and our method shrinks as the sample size increases.

In the convexity case, since the first-order derivative of the preliminary estimator is monotonized, we employed the SiLU activation function rather than ReLU. In Table 2.2, the non-convexity rate reaches 90.0% at a sample size of 100. This rate exhibits a non-monotonic pattern with increasing sample sizes, initially decreasing before subsequently increasing to 92.2% at larger sampling scales, which further implies that a larger sample size even increases the probability that the unrestricted network deviates from the true shape. Different from the case of monotonicity, the MSE on the test set from our method is larger than that from the benchmark and ICNN. This issue is caused by the method for convexity, especially, for small sample sizes. In the step 2 of convexity, $\mathbf{C}_1(x_1|X_{-1}; \mathcal{T})$

Table 2.1: **Results for monotonicity.** $\hat{\mathcal{T}}_{\text{ReLU}}$ represents the unrestricted network (benchmark) with ReLU activation function, $\hat{\mathcal{T}}_{\text{ReLU}_{\text{mon}}}$ represents our monotonic approximator, and $\hat{\mathcal{T}}_{\text{DLN}}$ is the Deep Lattice Network (DLN).

Sample Size	Rate of non-monotonicity		Average train MSE		Average test MSE		
	$\hat{\mathcal{T}}_{\text{ReLU}}$	$\hat{\mathcal{T}}_{\text{ReLU}_{\text{mon}}}$	$\hat{\mathcal{T}}_{\text{ReLU}}$	$\hat{\mathcal{T}}_{\text{ReLU}_{\text{mon}}}$	$\hat{\mathcal{T}}_{\text{ReLU}}$	$\hat{\mathcal{T}}_{\text{ReLU}_{\text{mon}}}$	$\hat{\mathcal{T}}_{\text{DLN}}$
n=100	0.974	0.000	0.928	0.943	1.117	1.109	1.253
n=200	0.974	0.000	0.889	0.922	1.094	1.077	1.187
n=300	0.978	0.000	0.914	0.938	1.073	1.059	1.145
n=400	0.972	0.000	0.929	0.948	1.056	1.044	1.118
n=500	0.968	0.000	0.967	0.972	1.029	1.027	1.102
n=1000	0.960	0.000	0.982	0.985	1.018	1.017	1.058
n=5000	0.946	0.000	0.992	0.994	1.009	1.008	1.012

Table 2.2: **Results for convexity.** $\hat{\mathcal{T}}_{\text{SiLU}}$ represents the unrestricted network (benchmark) with SiLU activation function, $\hat{\mathcal{T}}_{\text{SiLU}_{\text{con}}}$ represents our convex approximator with $u_0 = 0.5$, and $\hat{\mathcal{T}}_{\text{ICNN}}$ is the Input Convex Neural Network (ICNN).

Sample Size	rate of non-convexity		Average train MSE		Average test MSE		
	$\hat{\mathcal{T}}_{\text{SiLU}}$	$\hat{\mathcal{T}}_{\text{SiLU}_{\text{con}}}$	$\hat{\mathcal{T}}_{\text{SiLU}}$	$\hat{\mathcal{T}}_{\text{SiLU}_{\text{con}}}$	$\hat{\mathcal{T}}_{\text{SiLU}}$	$\hat{\mathcal{T}}_{\text{SiLU}_{\text{con}}}$	$\hat{\mathcal{T}}_{\text{ICNN}}$
n=100	0.900	0.000	0.867	1.628	1.221	1.862	1.072
n=200	0.584	0.000	0.956	1.021	1.05	1.103	1.038
n=300	0.548	0.000	0.972	1.002	1.035	1.058	1.025
n=400	0.556	0.000	0.978	0.998	1.029	1.043	1.022
n=500	0.584	0.000	0.981	0.986	1.020	1.023	1.021
n=1000	0.674	0.000	0.990	0.991	1.015	1.016	1.010
n=5000	0.922	0.000	0.996	1.004	1.010	1.014	1.001

contains the part $\mathcal{T}(u_0)$, with $u_0 = 0.5$ in our simulations. Consequently, the performance of proposed method is critically contingent on the predictive accuracy of the benchmark at this fixed point. Once the benchmark exhibits suboptimal fit at this singular point, the proposed method systematically propagates estimation biases throughout the shape-restricted framework. However, if we select the best u_0 from the list $[0.1, 0.3, 0.5, 0.7, 0.9]$, such that our method with this u_0 achieves the lowest MSE on the training set, then the corresponding MSE on the test set significantly decreases to 1.369. Moreover, the issue vanishes as the sample size increases, so this additional step of selecting u_0 is not required for moderate sample sizes.

Homogeneity is the most strict condition that the classical unrestricted neural networks can achieve among these three shapes, since the rate of non-homogeneity approaches 100% in Table 2.3. Compared to the benchmark and neural network using RePU(r) activation functions, our method achieves lower MSE on the test set.

Table 2.3: **Results for homogeneity.** $\hat{\mathcal{T}}_{\text{ReLU}}$ represents the unrestricted network (benchmark) with ReLU activation function, $\hat{\mathcal{T}}_{\text{ReLU_hom}}$ represents our homogeneous approximator, and $\hat{\mathcal{T}}_{\text{RePU}}$ is the neural network using RePU(r) as the activation functions.

Sample Size	rate of non-homogeneity		Average train MSE		Average test MSE		
	$\hat{\mathcal{T}}_{\text{ReLU}}$	$\hat{\mathcal{T}}_{\text{ReLU_hom}}$	$\hat{\mathcal{T}}_{\text{ReLU}}$	$\hat{\mathcal{T}}_{\text{ReLU_hom}}$	$\hat{\mathcal{T}}_{\text{ReLU}}$	$\hat{\mathcal{T}}_{\text{ReLU_hom}}$	$\hat{\mathcal{T}}_{\text{RePU}}$
n=100	1.000	0.000	0.919	0.943	1.104	1.062	1.066
n=200	1.000	0.000	0.951	0.977	1.058	1.034	1.030
n=300	1.000	0.000	0.967	0.988	1.045	1.026	1.046
n=400	1.000	0.000	0.987	0.996	1.050	1.024	1.042
n=500	0.998	0.000	0.986	0.998	1.051	1.022	1.040
n=1000	0.998	0.000	0.998	0.999	1.037	1.017	1.038
n=5000	1.000	0.000	0.998	1.002	1.008	1.004	1.025

Next, we consider the following function combining different shapes,

$$y = f(x_1, x_2) = 3x_1^2 + 3x_2^2,$$

which is homogeneous of degree 2 and is increasing with respect to the first coordinate. Therefore, both of the monotonicity and homogeneity were imposed to the unrestricted benchmark. We compared four different approximators, which are the unrestricted benchmark, the estimation with only monotonic constraint, the one with only homogeneous constraint and the one with both shapes. To keep the running time manageable, we set three sample sizes, namely, $n = 50, 100, 200$ for training and 50 samples for testing. The results in Table 2.4 are based on 100 simulations.

In Table 2.4, the rate of unsatisfied shape is 1 for all sample sizes, which also holds for the estimation with only monotonicity. However, if only homogeneity is imposed to the network, there are still above 80% of the corresponding networks satisfying monotonicity. For each sample size, the unrestricted network has the lowest MSE compared those with shape constraints, which is in line with the cases of single shape. As more shape constraints are imposed, the associated MSE on the test set demonstrates a progressive reduction.

Overall, the proposed method corrects the preliminary networks to satisfy the required shapes by trading off training-set approximation accuracy. By incorporating the true

Table 2.4: **Results for the combination of monotonicity and homogeneity.** $\hat{\mathcal{T}}_{\text{ReLU_mon}}$ represent the approximator with only monotonicity while $\hat{\mathcal{T}}_{\text{ReLU_hom}}$ represent one with only homogeneity. $\hat{\mathcal{T}}_{\text{ReLU_hom+mon}}$ represent the approximator with both shape constraints.

Sample Size	rate of unsatisfied shape			Average train MSE			Average test MSE		
	n=50	n=100	n=200	n=50	n=100	n=200	n=50	n=100	n=200
$\hat{\mathcal{T}}_{\text{ReLU}}$	1.000	1.000	1.000	0.891	0.960	0.959	1.183	1.123	1.080
$\hat{\mathcal{T}}_{\text{ReLU_mon}}$	1.000	1.000	1.000	0.906	0.969	0.968	1.178	1.120	1.074
$\hat{\mathcal{T}}_{\text{ReLU_hom}}$	0.210	0.270	0.210	0.955	0.989	0.984	1.114	1.065	1.041
$\hat{\mathcal{T}}_{\text{ReLU_hom+mon}}$	0.000	0.000	0.000	0.964	0.992	0.985	1.109	1.065	1.039

shape derived from prior knowledge, the shape-constrained correction guides the resulting estimation toward the topology of the underlying function, yielding substantial reductions in out-of-sample MSE. Although the ICNN achieves lowest MSE, the flexibility of our method allows employing the ICNN as a preliminary estimator, which can be further corrected for additional shape constraints, thereby improving estimation accuracy.

2.3.1 Case Studies

In this section, the proposed method was compared to the benchmark, as well as the Monotonic Multi-Layer Perceptron (MLP) (Zhang & Zhang 1999) and the Monotonic MIN-MAX (MM) (Sill 1997) network on two data sets which are taken from DaISy (De Moor 2005). In order to compare these networks, we followed the same process of data preparation from Minin et al. (2010). Each data set was randomly divided into a training set (70% of the data set) and a test set (30%). Then 20% of the training set was further selected as a validation set. The performance of all network estimations was evaluated on the test set based on two metrics, Root mean square (RMS) error and R^2 which is defined as

$$R^2 = 1 - \frac{\sum_n (y_n - \hat{y}_n)^2}{\sum_n (y_n - \bar{y})^2}$$

where y is the target output, $\hat{y}$ the network output, and $\bar{y}$ the average of the target y over the test set.

CD-player data set. This data set is based on the mechanical construction of a CD player arm (De Moor 2005). It has 2048 instances, and contains two attributes, the forces

of the mechanical actuators and the tracking accuracy of the arm. The larger the force the lower the tracking accuracy, which is the monotone dependency.

Ethane–ethylene distillation data set. This is a data set of a simulation related to the identification of an ethane-ethylene distillation column. It contains 90 instances, and each instance has 4 attributes, which include the top ethane composition, top pressure, input ethane composition and ratio between the reboiler duty and the feed flow. It is assumed that the larger the ratio between the distillate and the feed flow, the lower the top ethane composition (De Moor 2005).

Table 2.5: **The results for benchmark, and our shape-restricted estimation.** $\widehat{\mathcal{T}}_{\text{ReLU}}$, $\widehat{\mathcal{T}}_{\text{ReLU_mon}}$ and $\widehat{\mathcal{T}}_{\text{DLN}}$ represent the unrestricted network (benchmark), our shape-constrained estimator, and the Deep Lattice Network, respectively.

Data set	RMS error			R^2		
	$\widehat{\mathcal{T}}_{\text{ReLU}}$	$\widehat{\mathcal{T}}_{\text{ReLU_mon}}$	$\widehat{\mathcal{T}}_{\text{DLN}}$	$\widehat{\mathcal{T}}_{\text{ReLU}}$	$\widehat{\mathcal{T}}_{\text{ReLU_mon}}$	$\widehat{\mathcal{T}}_{\text{DLN}}$
CD-player	0.120	0.120	0.112	0.944	0.944	0.951
Distillation	0.366	0.321	0.975	0.724	0.788	-0.944

As shown in Table 2.5, on the CD-player dataset, our method demonstrates performance comparable to the MLP and MM networks, which attain RMS error and R^2 values of 0.09 and 0.90, and 0.55 and 0.96, respectively (Minin et al. 2010). The DLN achieves the best performance among all methods. However, on the Ethane–ethylene distillation data set, our method outperforms these three networks, as the MLP, MM and DLN approximators exhibit severe fitting issues, evidenced by their negative R^2 s (-0.944 , -1.0 and -1.1) (Minin et al. 2010). As we mentioned in the introduction section, the MLP and MM approximators are fundamentally parametric due to architectural constraints. While theoretically capable of universal monotonic approximation, their practical implementation necessitates explicit specification for hidden layer configurations. Suboptimal nodal cardinality—particularly underparameterization—induces fundamental representational capacity limitations in network architectures, precluding sufficient convergence toward the underlying function. Although DLN does not require explicit hidden-layer configuration, an unsuitable parameterization may still cause overfitting and consequently poor test-set performance.

2.4 Conclusion

This chapter introduces a novel shape-constrained correction method for general neural networks, which imposes multiple shape constraints simultaneously without requiring specific architectural modifications. Some case studies demonstrate that our method achieves comparable and even better performance than the existing shape-constrained networks. More importantly, the proposed method preserves the convergence rate of the preliminary (network-based) estimator to a considerable extent, and we further establish the asymptotic normality property. The derivation of such asymptotics remains predicated on specific architectural constraints and activation function regularity conditions, necessitating subsequent research efforts to establish large sample properties for more general network architectures.

Chapter 3

Nash Equilibrium Approximation in Team Games

3.1 Introduction

Two-team zero-sum games constitute a more realistic framework for modeling strategic interactions than conventional two-player games, yet they have received comparatively limited attention in the literature (Xu et al. 2025b). While many algorithms have achieved remarkable success in two-player games such as Go (Silver et al. 2016), poker (Moravčík et al. 2017, Brown & Sandholm 2018) and Stratego (Perolat et al. 2022), with some capable of approximating Nash equilibrium, extending these methods to two-team zero-sum games remains challenging.

In this chapter, we study two-team games where each player shares rewards with the other players in one of two teams and rewards between teams are zero-sum. This class of games can be viewed as the combination of two-player games and cooperative multi-player games (Foerster et al. 2018a), because the competition between teams and cooperation within teams need to be considered at the same time. A canonical example of two-team zero-sum games is bridge, but current algorithms still fail to compete with expert humans (McAleer et al. 2023). Among multiple solution concepts for team games, team maxmin equilibrium with correlation device (TMEC_{Cor}) is the most extensively studied (McAleer et al. 2023, Xu et al. 2025a). TMEC_{Cor} allows within-team players to communicate before play jointly correlating their strategies based on a distribution. Policy-Space Response Oracles (PSRO) (Lanctot et al. 2017) is a popular framework that generalizes the double oracle (DO) algorithm (McMahan et al. 2003) guaranteeing to converge to an approxi-

mate Nash equilibrium for multi-player games. Some straightforward extensions of PSRO to team games, such as team PSRO (McAleer et al. 2023) and Generalized Fictitious Cross-Play (GFXP) (Xu et al. 2025a) have been proposed to approximate the TMECor. These extensions treat each team as a centralized joint agent, thereby reducing the original two-team game to an equivalent two-player formulation and inheriting the associated convergence properties. However, such formulations rely on centralized coordination and become invalid when teammates act independently without communication or correlation devices. This motivates our focus on approximating the team Nash equilibrium in fully decentralized settings.

Motivated by this limitation, we extend PSRO to team games by employing multi-agent reinforcement learning (MARL) to expand the players’ policy populations through joint cooperative best responses. Since teammates share the same reward, the population expansion procedure corresponds to a cooperative multi-player learning problem (Foerster et al. 2018a), which theoretically induces an optimal joint best response. Unlike DO, which computes individual best responses independently and therefore overlooks the coordination among teammates, our framework estimates the best response at the team level while preserving decentralized individual policies. Meanwhile, unlike the original PSRO applied to team games, which treats a team as a single agent and allows potentially correlated joint teammate strategies, our approach decomposes the learned team-level response into individual policies, enabling coordinated decision-making without requiring communication among teammates. The contributions of this chapter include: 1) our algorithm is a straightforward extension of PSRO to team games converging to the team Nash equilibrium without any communication; 2) it exhibits a higher tendency to converge to global Nash equilibria compared with classical PSRO. Before presenting the proposed algorithm, we first introduce the game setup and notation, followed by a review of the key components upon which our framework is built, including reinforcement learning-based best-response computation, meta-solvers, and policy population initialization.

The remainder of this chapter is organized as follows. Section 3.2 introduces the game setup and notation. Section 3.3 reviews the key components underlying our framework. Section 3.4 presents the proposed algorithm, followed by the experimental results.

3.2 Game Setup

A zero-sum two-team game is often described as an extensive-form game, defined by a tuple

$$G = (J, C, V, \omega, \{\mathcal{I}_j\}_{j \in J}, \{A_j\}_{j \in J}, X, P, r),$$

where $J = \{\textcolor{red}{M}1, \textcolor{red}{M}2, \textcolor{blue}{N}1, \textcolor{blue}{N}2\}$ is the set of players; and each team $\textcolor{red}{M}/\textcolor{blue}{N}$ has 2 players playing cooperatively against the other. Player C is a special player, called the chance player, modelling exogenous stochasticity in the environment. V is a finite tree of histories divided into subsets of terminal nodes (or leaves) Z and decision nodes D . ω is a function assigning each decision node v to players. If $\omega(v) = C$, the action is controlled by Nature, with a set of possible outcomes $X(v)$ and probability distribution $P(.|v)$. $A_j(.)$ is the set of available actions at each decision node for each player $j \in J$. r is a reward function mapping each $z \in Z$ to a reward $\{r_j(z)\}_{j \in J}$ with $r_{\textcolor{red}{M}1} = r_{\textcolor{red}{M}2}$, $r_{\textcolor{blue}{N}1} = r_{\textcolor{blue}{N}2}$ and $\sum_{j \in J} r_j = 0$ due to the zero-sum setting. For simplicity, we write $r_{\textcolor{red}{M}}$ and $r_{\textcolor{blue}{N}}$ as the rewards for each team.

In the games of imperfect information, the private information is represented via information set (info set) $I \in \mathcal{I}_j$. In particular, the set of decision nodes D is partitioned into a collection $\mathcal{I}_j$ of non-empty sets, and each $I \in \mathcal{I}_j$ groups together nodes which are indistinguishable to player j . This implies that for any $v, w \in I$, nodes v and w have the same set of available actions. Perfect recall is assumed; that is, no player forgets information acquired earlier in the game. In this chapter, we focus on two types of extensive games: simultaneous game and sequential game. To model more sophisticated real-world scenarios, we enrich the extensive game with extra concepts: world state S , transition function $\mathcal{T}$ and observation function O .

Simultaneous game is a game where each player $j \in J$ simultaneously chooses an action $a_j \in A_j$ without knowledge of the actions chosen by other players, which implies that $\omega(v) = J$ at all decision nodes which are not acted by the chance player. These actions form a joint action $\mathbf{a} = (\mathbf{a}_{\textcolor{red}{M}}, \mathbf{a}_{\textcolor{blue}{N}}) = (a_{\textcolor{red}{M}1}, a_{\textcolor{red}{M}2}, a_{\textcolor{blue}{N}1}, a_{\textcolor{blue}{N}2}) \in \prod_{j \in J} A_j = \mathcal{A}$, inducing the transition in the environment from world state $s \in S$ to the other s' based on the state transition function $\mathcal{T}(s, \mathbf{a}) \in S$.

- The game eventually ends with finite playing times when a terminal state (or node in extensive-form games) is reached. Then a history at time t can be represented by a sequence of states and actions given by $h = (s^1, \mathbf{a}^1, s^2, \mathbf{a}^2, \dots, s^t)$.

- After each transition, each player $j \in J$ has a partial observation $o_j = O(s, \mathbf{a}, s')$ with $O(s, \mathbf{a}, s') : S \times \mathcal{A} \times S \rightarrow O_A \times O_S$ instead of fully observing the actions and states, where O_A and O_S are the space of the observed actions and states, respectively. As a result, an information state for player j , denoted by I_j , is a sequence of the corresponding observations up to time t , given by $I_j(h) = \{o_j^1, o_j^2, \dots, o_j^t\} \equiv I_j$. Moreover, the union of all players' observations reveals the true state and actions.
- A pure policy for player j specifies the action that player j selects at each information set, that is, $\pi_j(I_j) : \mathcal{I}_j \rightarrow A_j(I_j)$. The policy space is denoted by Π_j for each player j ; and write $\boldsymbol{\pi} = \{\pi_j\}_{j \in J}$. A mixed policy is a probability distribution $\sigma_j \in \Delta(\Pi_j)$ over the policy space Π_j . Given a policy profile $\boldsymbol{\sigma} := (\sigma_{M1}, \sigma_{M2}, \sigma_{N1}, \sigma_{N2}) \in \prod_{j \in J} \Delta(\Pi_j)$, the expected reward for players in team M is

$$V_M(\boldsymbol{\sigma}) := \mathbb{E}_{\boldsymbol{\pi} \sim \boldsymbol{\sigma}} [r_M];$$

while the players in team N obtain $V_N = -V_M$ due to the zero-sum setting.

Sequential game starts by an initial player randomly at time $t = 1$. Then, the action turns of players are based on ω . Specifically, $w(v) \in J$ at all decision nodes which are not acted by the chance player. Therefore, at each state, the corresponding player (i.e. j) chooses an action $a \in A_j$. This action induces the transition in the environment based on the state transition function $\mathcal{T}(s, a) \in S$.

- A history at time t can be represented by a sequence of states and actions given by $h = (s^1, a^1, s^2, a^2, \dots, s^t)$. In the sequential case, only one action is performed between states.
- Instead of fully observing the actions and states, each player $j \in J$ has a partial observation $o_j = O(s, a, s')$, which results in the information $I_j = \{o_j^1, o_j^2, \dots, o_j^t\}$.
- A pure policy for player j specifies the action selected for each observed information, given by $\pi_j(I_j) : \mathcal{I}_j \rightarrow A_j(I_j)$. A mixed policy is a probability distribution $\sigma_j \in \Delta(\Pi_j)$ over the policy space Π_j . Given a policy profile $\boldsymbol{\sigma} := (\sigma_{M1}, \sigma_{M2}, \sigma_{N1}, \sigma_{N2}) \in \prod_{j \in J} \Delta(\Pi_j)$, the expected reward for players in team M is

$$V_M(\boldsymbol{\sigma}) := \mathbb{E}_{\boldsymbol{\pi} \sim \boldsymbol{\sigma}} [r_M];$$

while the players in team N obtain $V_N = -V_M$ due to the zero-sum setting.

The difference between the simultaneous and sequential games includes the action turns, state transition, reward function and observed history; however, there is no huge difference for policy spaces except the history set. Therefore, the mixed policy is a probability distribution over the policy space in both cases. Each player aims to mix their policies to maximize the corresponding expected rewards. This setup can represent many games with finite steps, such as poker or chess.

3.3 Literature Review

The proposed algorithm builds upon three fundamental components that have been extensively studied in the literature: reinforcement learning for best-response computation, meta-solvers for equilibrium approximation, and policy population initialization. Since our method extends the PSRO framework by modifying these components for decentralized team games, we briefly review the relevant literature and introduce the techniques throughout this section.

3.3.1 Reinforcement Learning based Best Response

With the development of deep learning, reinforcement learning has been significantly advanced through its integration with deep neural networks, achieving impressive results in chess, Go, as well as some imperfect information games such as poker (Perolat et al. 2022, Silver et al. 2018, Moravčík et al. 2017, Brown & Sandholm 2018, 2019). Many of these successes are built upon actor-critic methods, a widely used class of reinforcement learning algorithms. This algorithm consists of two main components: value function, which works as the “critic” to estimate the value or advantage of the players’ action based on TD, Retrace, V-trace, etc (Sutton 1988, Munos et al. 2016, Espeholt et al. 2018); policy function, which works as the “actor” to determine which action to take given observed information. Both functions can be represented by deep neural networks once the game is encoded; and the policy function is updated toward maximizing the expected rewards based on the “critic”. In this section, we provide an example of the actor-critic algorithm for the multi-agent case corresponding to the team best response. As the player best response is a special case of the multi-agent one, its details is omitted.

The actor-critic algorithm starts from the estimation of the “critic” function. Some previous work (Oliehoek et al. 2008, Kraemer & Banerjee 2016, Foerster et al. 2016, Jorge et al. 2016, Foerster et al. 2018b) utilize this function by centralized training but decentralized execution; while some work (Perolat et al. 2022) train and execute it decentralizely.

This difference stems from the level at which the critic is evaluated. Due to the rewards sharing setting, we use the former one. Firstly, the centralized critic, action-value function,

$$Q(s^t, \mathbf{I}^t, \mathbf{a}^t) = \mathbb{E}_{\mathbf{a} \sim \sigma} [r | s^t, \mathbf{I}^t, \mathbf{a}^t],$$

is learned for the joint action $\mathbf{a}^t$ conditional on the state s^t and the joint observed information $\mathbf{I}^t$ by $TD(\lambda)$ (Foerster et al. 2018b), Retrace (Munos et al. 2016) and V-trace (Espeholt et al. 2018). For example, this function, evaluating the value of an action at each state, is estimated by the critic $Q^\theta(\cdot)$, which is trained on-policy by using a deep neural network version of $TD(\lambda)$ (Foerster et al. 2018b). In particular, in each iteration, the parameters θ are updated by gradient descent to minimize the loss:

$$\mathcal{L}^t(\theta) = [Q_{\text{target}}^t - Q^\theta(\cdot_t)]^2,$$

where $Q_{\text{target}}^t = (1 - \lambda) \sum_{n=1}^{\infty} \lambda^{n-1} G_{(n)}^t$, and $G_{(n)}^t = \sum_{l=1}^n \gamma^{l-1} r^{t+l} + \gamma^n Q^\theta(\cdot_{t+n})$. $G_{(n)}^t$ are calculated with bootstrapped values estimated by a target network with parameters copied periodically from θ . Then the action advantage function (Foerster et al. 2018b) for each player (i.e. player **M1**) is estimated, by comparing the Q -value for the current action $a_{\mathbf{M1}}$ to a counterfactual baseline while keeping the others' action fixed:

$$\begin{aligned} A_{\mathbf{M1}}(s^t, \mathbf{I}^t, \mathbf{a}^t) \\ = Q^\theta(s^t, \mathbf{h}^t, \mathbf{a}^t) - \sum_{a'_{\mathbf{M1}}} \pi_{\mathbf{M1}}(a'_{\mathbf{M1}} | \mathbf{I}_{\mathbf{M1}}^t) Q^\theta(\mathbf{I}^t, a'_{\mathbf{M1}}, a_{\mathbf{M2}}^t). \end{aligned}$$

The advantage function can also be estimated based on the state-value function V , which approximates the value of a state (Schulman et al. 2017). Specifically, in each iteration, the function V^θ , is updated by gradient descent to minimize the loss:

$$\mathcal{L}^t(\theta) = [V_{\text{target}}^t - V^\theta(\cdot_t)]^2,$$

where $V_{\text{target}}^t = \sum_{l=0}^{\infty} \gamma^l r^{t+l}$. Then the advantage function for each player (i.e. player **M1**) is computed by

$$A_{\mathbf{M1}}(s_t) = \delta_{\mathbf{M1}}^t + (\gamma\lambda)\delta_{\mathbf{M1}}^t + \dots + (\gamma\lambda)^{T-t+1}\delta_{\mathbf{M1}}^{T-1},$$

where $\delta_{\mathbf{M1}}^t = r^t \gamma V^\theta(s^{t+1}) - V^\theta(s^t)$.

At the end of each iteration, the player's policy can be updated by the gradient based on

$$\sum_{j \in \{\mathbf{M1}, \mathbf{M2}\}} \nabla_{\theta} \log \pi_j(a_j | I_j) A_j$$

at each iteration k . It is important to note that the trained policy is a behaviour policy, since behaviour policies provide a natural and tractable way to act within unknown environment and their differentiability is a prerequisite for applying gradient-based optimization methods. For this trained behaviour policy, there exists a reward-based equivalent mixed policy over the space of reduced-normal-form policies due to perfect recall (Kuhn & Tucker 1953). Therefore, one of the policies with positive probabilities is the required best response. On the other hand, our framework is not limited to the reinforcement learning; any method capable of estimating the best response policy can be employed for simultaneous games (Yu et al. 2022, Rashid et al. 2020, Lowe et al. 2017), or sequential games (Perolat et al. 2022, Silver et al. 2016).

3.3.2 Meta-Solver

Meta-solvers are a fundamental component of the PSRO framework, responsible for computing approximate equilibrium strategies over restricted policy populations. Rather than solving the original game directly, PSRO iteratively constructs a sequence of restricted normal-form games where players choose policies from the space Π , but this space is excessively large (i.e. $|A|^{|H|}$) in many games. For example, the well-researched imperfect information game, the no-limit Texas Hold'em poker, has 10^{164} states while the Stratego has even 10^{535} states (Perolat et al. 2022), implying the huge policy space. Although the corresponding ϵ -approximate equilibrium can be approximated theoretically, the required polynomial time is still intractable, that is PPAD-complete (Daskalakis et al. 2009). However, it is reasonable to expect many policies will never be used, so we follow the idea of DO and PSRO and tackle the original large game by decomposing it into a sequence of normal-form subgames, where meta-solvers play a pivotal role to solve each of these subgames. A meta-solver takes the normal-form game as input and produces a mixed policy σ^i for each player $i \in \mathcal{I}$ such that no player has incentive to deviate. The commonly used meta-solvers include linear programming, fictitious play (Brown 1951), regret-matching (Hart & Mas-Colell 2000, 2001), Hedge (Auer et al. 1995), and replicator dynamics (Taylor & Jonker 1978, Lanctot et al. 2017).

The general procedure of these methods starts with an initial mixed policies σ_j^0 for $j \in J$. Fictitious play updates the policy for each player via

$$\sigma_j^k = \left(1 - \frac{1}{k+1}\right) \sigma_j^{k-1} + \frac{1}{k+1} \sigma_j'^k$$

where $\sigma_j'^k = \arg \max_{\sigma_j \in \Delta(\Pi_j)} V_j(\sigma_j, \boldsymbol{\sigma}_{-j}^{k-1})$. In regret matching, for each player j , the regret

for policy π_j at iteration k is computed as

$$R_j^k(\pi_j) = \sum_{m=1}^k r^m(\pi_j)$$

with $r^m(\pi_j) = V_j(\pi_j, \sigma_{-j}^{m-1}) - V_j(\sigma^{m-1})$. Then the policy is updated according to the regret matching rule,

$$\sigma_j^{k+1}(\pi_j) = \frac{R_j^{k+}(\pi_j)}{\sum_{\pi'_j \in \Pi_j} R_j^{k+}(\pi'_j)},$$

where $x^+ = \max\{0, x\}$. If the denominator is zero, each policy is assigned with equal probability. Since these meta-solvers are straight-forward algorithms, we defer the details of other methods to Appendix C.

It is important to note that deep neural networks do not only empower the implementation of reinforcement learning but also support the traditional meta-solvers, such as Neural Replicator Dynamics (NeuRD) (Hennes et al. 2020), that is equivalent to the soft-max counterfactual regret minimization guaranteeing the convergence to an ϵ -approximate equilibrium (Hennes et al. 2020).

3.3.3 Policy Population Initialization

The initialization of the policy population plays a crucial role in the efficiency and stability of equilibrium approximation, as the initial set of policies determines the starting point of the iterative population expansion process. Since the equilibrium is approximated by solving a sequence of restricted normal-form games as the policy population expands, so different initial policy populations may have different effect on the procedure of equilibrium approximation. Some work starts the equilibrium approximation with the population consisting of random policies (Dönmez et al. 2024, Liu, Wang, Wang, Zhang, Yang, Zhang, An & Wen 2024, Zhang et al. 2021, Muller et al. 2019, Lanctot et al. 2017), while some work (Duan et al. 2021) show that starting with a proper initial population, in practice, can contribute to more efficient approximation of equilibrium, although the equilibrium can always be achieved with any initialization theoretically.

There are two general methods of initializing the policy population: pre-trained policies, and “imitation learning” policies from human experts. The pre-trained policy refers to a policy that is trained directly through reinforcement learning or analogous deep learning

methodologies. These policies have performed impressive results in many poker or chess games against humans (Perolat et al. 2022, Silver et al. 2018, Moravčík et al. 2017, Brown & Sandholm 2018, 2019), so they can be viewed as “proper” policies being added into the initial population. On the other hand, imitation learning, as a compelling approach for sequential games, aims to emulate the behaviour of the expert’s policy (Ng et al. 2000). One of the simplest ideas from Behavioral Cloning (BC) is to maximize the likelihood of the expert’s actions at each state, without reasoning about the consequences of actions (Pomerleau 1991), which causes the agent drift away from the demonstrated states especially in sequential games (Ross et al. 2011). To overcome this issue, recent methods based on inverse reinforcement learning (IRL) aim not only to imitate the actions, but also to visit demonstrated states (Fu et al. 2017, Reddy et al. 2019, Finn et al. 2016, Ng et al. 2000). Since the key idea of the proposed algorithm is the policy population expansion, we only employ the random policy initialization in the following experiments to keep the computational cost manageable. Nevertheless, alternative initialization strategies may further improve empirical performance and are left for future investigation.

3.4 Methodology

This section presents the proposed framework for approximating team Nash equilibria in fully decentralized environments without communication. We first formally define the team equilibrium concept, and then introduce the proposed equilibrium approximation algorithm,

3.4.1 Team Equilibrium without Communication

Definition 3.4.1 *A policy profile $\sigma = (\sigma_{M1}, \sigma_{M2}, \sigma_{N1}, \sigma_{N2})$ is a team equilibrium without communication if for each player $j \in J$, there is*

$$\max_{\pi_j \in \Pi_j} V_j(\pi_j, \sigma_{-j}) \leq V_j(\sigma),$$

where σ_{-j} is the joint policy except the player j .

This definition is different from the TMECor (McAleer et al. 2023), in which players within teams are allowed communication before the game starts. In particular, the equilibrium

search space in TMECor is $\Delta(\Pi_{\textcolor{red}{M}1} \times \Pi_{\textcolor{red}{M}2}) \times \Delta(\Pi_{\textcolor{blue}{N}1} \times \Pi_{\textcolor{blue}{N}2})$ while that space in our equilibrium is $\times_{j \in J} \Delta(\Pi_j)$. Furthermore, we can define the best response for the player j against the fixed policy σ_{-j} as

$$\mathbb{BR}(\sigma_{-j}) = \arg \max_{\pi_j \in \Pi_j} V_j(\pi_j, \sigma_{-j}).$$

Since the players within the same team share the rewards, we can also have the team best response (i.e. for team $\textcolor{red}{M}$) against the fixed team policy $\sigma_{\textcolor{blue}{N}}$,

$$\text{TBR}(\sigma_{\textcolor{blue}{N}}) = \arg \max_{\pi_{\textcolor{red}{M}} \in \Pi_{\textcolor{red}{M}1} \times \Pi_{\textcolor{red}{M}2}} V_{\textcolor{red}{M}}(\pi_{\textcolor{red}{M}}, \sigma_{\textcolor{red}{M}}).$$

To approximate the team equilibrium in definition 3.4.1, we first generalize the best response to any policy that achieves expected rewards against the others' policies within ϵ of optimal.

Definition 3.4.2 (*ϵ -best response*) A policy σ_j is an ϵ -best response for player j against the fixed policy profile σ_{-j} , if

$$V_j(\mathbb{BR}(\sigma_{-j}), \sigma_{-j}) \leq V^i(\sigma_j, \sigma_{-j}) + \epsilon.$$

We can further define

$$e_j(\sigma) := V_j(\mathbb{BR}(\sigma_{-j}), \sigma_{-j}) - V_j(\sigma)$$

as player j 's incentive to deviate under the policy profile σ . Then, the approximation of team equilibrium is defined by allowing each player's incentive to deviate, that is, the policy of each player is an ϵ -best response.

Definition 3.4.3 (*ϵ -approximate equilibrium*) A policy profile $\sigma = (\sigma_{\textcolor{red}{M}1}, \sigma_{\textcolor{red}{M}2}, \sigma_{\textcolor{blue}{N}1}, \sigma_{\textcolor{blue}{N}2})$ is an ϵ -approximate equilibrium if for each player $j \in J$, there is

$$e_j(\sigma) = V_j(\mathbb{BR}(\sigma_{-j}), \sigma_{-j}) - V_j(\sigma) \leq \epsilon.$$

This definition allows at most ϵ rewards gained from deviating for each player. Some scholars utilize $\sum_{j \in J} e_j(\sigma)$ as the metric to quantify the distance from the optimal profile; and they define $\sum_{j \in J} e_j(\sigma)/|J|$ as the exploitability (Timbers et al. 2020). If we set $\epsilon = 0$, this is just the team equilibrium in definition 3.4.1.

3.4.2 Equilibrium Approximation Algorithm

Direct application of the reinforcement learning (RL) for learning policies that maximize the expected rewards have achieved considerable success in both single or multi-agent games especially under the fixed environment (Sutton et al. 1999, Foerster et al. 2018a). However, such approaches generally do not guarantee convergence to Nash equilibria in competitive player-versus-player or team-versus-team games. Therefore, instead of training a single policy, Policy-Space Response Oracles (PSRO)(Lanctot et al. 2017) scales up the Double Oracles (DO) (McMahan et al. 2003) to large games by replacing the best response operator with reinforcement learning-based approximations. In recent years, DO or PSRO based methods have been widely used to approximate the Nash equilibrium for non-team games and approximate TMECor or CTME for team games (McAleer et al. 2023, Muller et al. 2019, Balduzzi et al. 2019, Yao et al. 2023, Lanctot et al. 2017, Xu et al. 2025a, Liu, Wang, Zhang, Yang, An & Wen 2024, Liu, Wang, Wang, Zhang, Yang, Zhang, An & Wen 2024). In this chapter, we propose a general algorithm framework to approximate the team equilibrium (Definition 3.4.1) for team games without any communication and correlation devices.

Our algorithm starts with two initial populations $\Pi_{M1}^0 = \{\pi_{M1}^0\}$, $\Pi_{M2}^0 = \{\pi_{M2}^0\}$, $\Pi_{N1}^0 = \{\pi_{N1}^0\}$, and $\Pi_{N2}^0 = \{\pi_{N2}^0\}$ for each player, respectively. In the k -th iteration, the restricted game $G(\Pi_{M1}^k, \Pi_{M2}^k, \Pi_{N1}^k, \Pi_{N2}^k)$ is constructed, and it is a zero-sum normal-form game where team M players take policies from the population Π_{M1}^k, Π_{M2}^k and team N players take policies from the population Π_{N1}^k, Π_{N2}^k . The “pure strategies” in the traditional normal-form game become policies in this constructed game. Since each population Π^k is finite, a mixed Nash equilibrium $(\sigma_{M1}, \sigma_{M2}, \sigma_{N1}, \sigma_{N2})$ of this restricted game $G(\Pi_{M1}^k, \Pi_{M2}^k, \Pi_{N1}^k, \Pi_{N2}^k)$ can be computed by meta-solvers because of the Nash’s theorem(Nash 2024). Next, two types of the best response to this NE (Nash equilibrium) in the original game is computed and added to the corresponding population. Finally, our algorithm terminates when the best response for each player does not gain more than the NE reward of the restricted game. Since no best response can improve the reward for each player at the termination, our algorithm converges to the meta-policy team equilibrium.

Proposition 3.4.1 *For the game with finite number of policies, our algorithm with exact best response and meta-solver converges to a team equilibrium without communication.*

Proposition 3.4.2 *For the game with finite number of policies, our algorithm with proper best response and meta-solver converges to an ϵ -approximate equilibrium.*

Algorithm 1 Proposed Equilibrium Approximation Algorithm

```
1: Input: Initialize the population of policies  $\Pi_{M1}^0, \Pi_{M2}^0, \Pi_{N1}^0, \Pi_{N2}^0$ 
2: Output: Equilibrium Approximate  $(\sigma_{M1*}, \sigma_{M2*}, \sigma_{N1*}, \sigma_{N2*})$ 
3: repeat  $k \leftarrow 0, 1, 2, \dots$ 
4:    $\sigma = (\sigma_{M1}, \sigma_{M2}, \sigma_{N1}, \sigma_{N2}) \leftarrow$  NE in game  $G(\Pi_{M1}^k, \Pi_{M2}^k, \Pi_{N1}^k, \Pi_{N2}^k)$  by meta-solvers
5:    $\pi_{TM1}, \pi_{TM2} \leftarrow \text{TBR}(\sigma_N)$ 
6:    $\pi_{TN1}, \pi_{TN2} \leftarrow \text{TBR}(\sigma_M)$ 
7:    $\Pi_{M1}^{k+1} \leftarrow \Pi_{M1}^k \cup \{\pi_{TM1}\}$ 
8:    $\Pi_{M2}^{k+1} \leftarrow \Pi_{M2}^k \cup \{\pi_{TM2}\}$ 
9:    $\Pi_{N1}^{k+1} \leftarrow \Pi_{N1}^k \cup \{\pi_{TN1}\}$ 
10:   $\Pi_{N2}^{k+1} \leftarrow \Pi_{N2}^k \cup \{\pi_{TN2}\}$ 
11: until  $V_j(\text{BR}(\sigma_{-j}), \sigma_{-j}) - V_j(\sigma) \leq \epsilon$  for each player  $j$ 
12: Return  $(\sigma_{M1}, \sigma_{M2}, \sigma_{N1}, \sigma_{N2})$ 
```

The existence of the meta-policy team equilibrium is ensured by the Nash’s theorem (Nash 2024) if the game with finite state and policy spaces. However, when multiple Nash equilibria exist, our framework cannot determine which equilibrium will be converged to. Indeed, the point of convergence is influenced by the policy population initialization. Although this algorithm must expand all policies in the worst case, it is reasonable to expect many policies will never be the best response, so this algorithm will terminate early in practice, like DO and PSRO (McMahan et al. 2003, Lanctot et al. 2017, McAleer et al. 2023). Different from DO and PSRO, the proposed framework provides a middle ground between independent learning and fully centralized team modeling. Specifically, it estimates the best response at the team level to account for cooperative interactions among teammates, but assigns the resulting policy components to individual agents rather than representing the team as a single joint policy. This differs from DO, which independently computes individual best responses without explicitly considering teammate coordination, and from PSRO, which models a team as a unified agent with potentially correlated teammate strategies.

3.5 Experiments

The proposed algorithm is evaluated on two classes of games: the matrix game, which is the simplest case of simultaneous game, and the sequential game. We first compare the learning dynamics of the proposed algorithm with those of the classical PSRO (Lanctot

et al. 2017) using two team-based competitive matrix games, *Team Rock-Paper-Scissors* (team RPS) game and *Occupation Game*. We then evaluate both the proposed algorithm and classical PSRO in the sequential game, *Kuhn Poker*.

3.5.1 Practical Implementation

For the two matrix games, we employed the proximal policy optimization (PPO) algorithm (Schulman et al. 2017) to approximate the team best response policy in the proposed algorithm and individual best response policy in classical PSRO. Each player’s policy is represented by a two-hidden-layer (32 nodes in each layer) fully connected network, which is trained in 1000 epochs with a learning rate of 10^{-3} . In each iteration, the regret-matching (Hart & Mas-Colell 2000, 2001) algorithm is applied to compute the Nash equilibrium of the corresponding subgame. Moreover, both algorithms are initialized with all possible initial policy population.

For the sequential game, we continue to employ the proximal policy optimization (PPO) algorithm to approximate best-response policies and apply the regret-matching algorithm to compute Nash equilibrium under the same training settings used in the matrix-game experiments.

3.5.2 Matrix Game

Team Rock-Paper-Scissors (Team RPS) game extends the two-player zero-sum game Rock-Paper-Scissors (RPS) to 4-player setting, in which 4 players are divided into two teams playing against each other. Each player can choose action either 0 or 1. The joint action from two players within the same team determines the final action of team. If both players in the same team choose action 0, then the team plays Rock; if both choose 1, the team plays Scissors; otherwise, the team plays Paper. Since the players cannot communicate with the teammate or jointly choose the action, the classical NE where each team plays Rock, Paper and Scissors with equal probability cannot be achieved. However, there is a NE where all players choose action 1.

We run both the proposed algorithm and classical PSRO with 5 different initial policy

populations, as following,

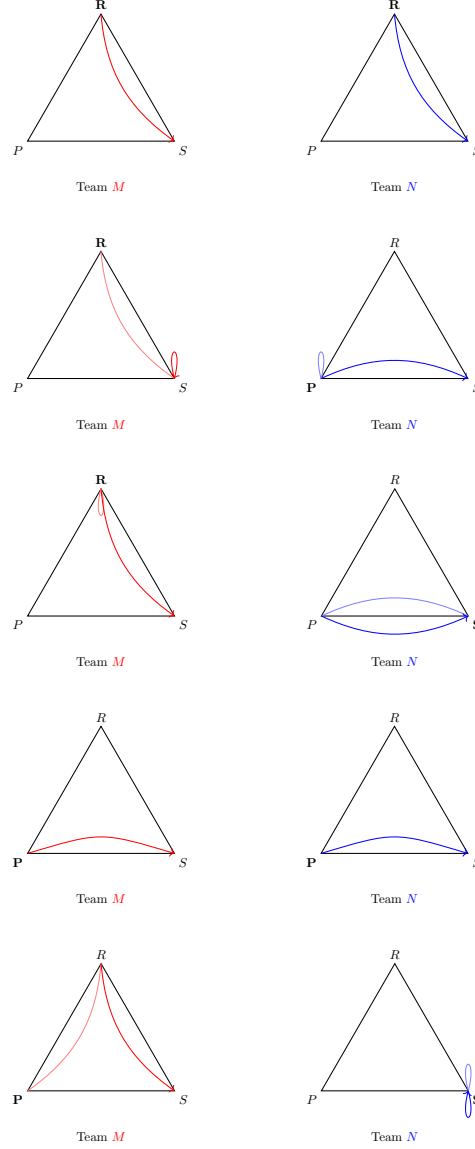
$$\begin{aligned}
\Pi_{M1}^0, \Pi_{M2}^0, \Pi_{N1}^0, \Pi_{N2}^0 &= \{0\}, \{0\}, \{0\}, \{0\}; \\
&= \{0\}, \{0\}, \{0\}, \{1\}; \\
&= \{0\}, \{0\}, \{1\}, \{1\}; \\
&= \{1\}, \{0\}, \{1\}, \{0\}; \\
&= \{0\}, \{1\}, \{1\}, \{1\}.
\end{aligned}$$

As shown in Figures 3.1 and 3.2, both the proposed algorithm and classical PSRO converge to the Nash equilibrium corresponding to the Scissors strategy within several iterations. However, classical PSRO converges more rapidly, requiring fewer iterations to reach the equilibrium.

Nevertheless, the proposed algorithm enables simultaneous strategic adaptation among teammates, thereby allowing a richer set of potentially beneficial strategies to be incorporated into the policy population. As a result, the algorithm is better suited to approximating global Nash equilibria in which each team's strategy is unexploitable¹. The advantage of this cooperative adaptation mechanism becomes more apparent in the following game.

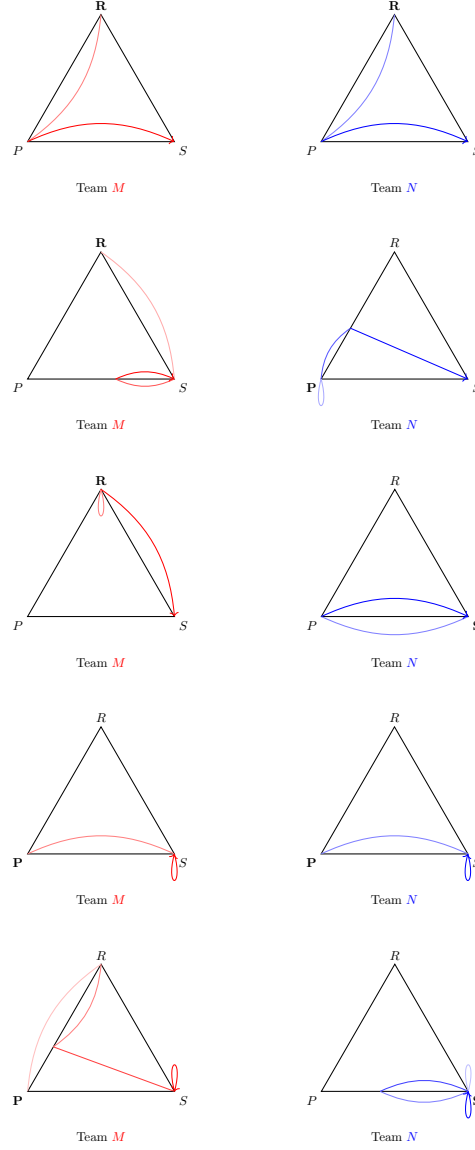
¹An unexploitable policy is a strategy for which no opponent strategy can achieve a higher expected payoff by exploiting it.

Figure 3.1: Classical PSRO's learning dynamics with 5 different initial policy populations².



²Teams are initialized at the bolded vertices of the triangular simplex, corresponding to pure strategies with 100% probability of playing Rock, Paper, or Scissors. Each arrow indicates the evolutionary trajectory of the strategy distribution, with colour intensity increasing from light to dark as the strategy converges.

Figure 3.2: Proposed algorithm's learning dynamics with 5 different initial policy populations³.



³Teams are initialized at the bolded vertices of the triangular simplex, corresponding to pure strategies with 100% probability of playing Rock, Paper, or Scissors. Each arrow indicates the evolutionary trajectory of the strategy distribution, with colour intensity increasing from light to dark as the strategy converges.

Occupation Game. We consider a 4-player zero-sum occupation game where four players, $J = \{\textcolor{red}{M}1, \textcolor{red}{M}2, \textcolor{blue}{N}1, \textcolor{blue}{N}2\}$, are divided into two teams $\textcolor{red}{M}$ & $\textcolor{blue}{N}$ playing against each other. Players within the same team receive the same team payoff. The game is played over a finite set of four regions, denoted by $R = \{r_1, r_2, r_3, r_4\}$, each of which can be occupied by at most one team. A region is considered controlled by a team if the number of team members selecting it is larger than that of opposing team; otherwise, the region is contested and yields no payoff to either team. The team payoff is determined by the number of controlled regions. Formally, let $C_{\textcolor{red}{M}} \subseteq R$ and $C_{\textcolor{blue}{N}} \subseteq R$ denote the sets of regions controlled by Team $\textcolor{red}{M}$ and Team $\textcolor{blue}{N}$, respectively. The payoff to team $\textcolor{red}{M}$ is $u_{\textcolor{red}{M}} = |C_{\textcolor{red}{M}}| - |C_{\textcolor{blue}{N}}|$, and team $\textcolor{blue}{N}$ receives $u_{\textcolor{blue}{N}} = -u_{\textcolor{red}{M}}$.

In this game, we run the proposed algorithm and classical PSRO with 5 different initial policy populations,

$$\begin{aligned} \Pi_{\textcolor{red}{M}1}^0, \Pi_{\textcolor{red}{M}2}^0, \Pi_{\textcolor{blue}{N}1}^0, \Pi_{\textcolor{blue}{N}2}^0 &= \{r_1\}, \{r_1\}, \{r_1\}, \{r_1\}; \\ &= \{r_1\}, \{r_1\}, \{r_1\}, \{r_2\}; \\ &= \{r_1\}, \{r_1\}, \{r_2\}, \{r_2\}; \\ &= \{r_1\}, \{r_2\}, \{r_3\}, \{r_4\}; \\ &= \{r_1\}, \{r_1\}, \{r_2\}, \{r_3\}. \end{aligned}$$

Since the occupation game admits multiple Nash equilibria, although both algorithms exhibit convergence to a Nash equilibrium under different initial settings, the particular equilibrium to which each algorithm converges may differ. In general, the proposed algorithm (5.4 iterations on average) requires more iterations to achieve the NE than the classical PSRO (4.4 iterations on average), since the proposed one expands the policy population allowing the within-team players to deviate simultaneously. Across all five initialization settings, the proposed algorithm successfully approximated four global Nash equilibria associated with unexploitable strategies. By comparison,, the classical PSRO failed to identify any global Nash equilibrium under the same experimental settings. These findings suggest that, in the absence of intra-team communication, the proposed algorithm exhibits a stronger tendency to converge toward global Nash equilibria, thereby demonstrating improved robustness in decentralized environments.

Table 3.1: Classical PSRO’s Nash equilibrium with 5 different initial policy populations.

Initial policy population	Nash equilibrium
$\{r_1\}_{\textcolor{red}{M1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_1\}_{\textcolor{red}{M2}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_1\}_{\textcolor{blue}{N1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_1\}_{\textcolor{blue}{N2}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_1\}_{\textcolor{red}{M1}}$	$(r_1, r_2, r_3, r_4) = (0, 1, 0, 0)$
$\{r_1\}_{\textcolor{red}{M2}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{2}, 0, \frac{1}{2}, 0\right)$
$\{r_1\}_{\textcolor{blue}{N1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0\right)$
$\{r_2\}_{\textcolor{blue}{N2}}$	$(r_1, r_2, r_3, r_4) = \left(0, \frac{1}{2}, 0, \frac{1}{2}\right)$
$\{r_1\}_{\textcolor{red}{M1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{3}, \frac{1}{3}, 0, \frac{1}{3}\right)$
$\{r_1\}_{\textcolor{red}{M2}}$	$(r_1, r_2, r_3, r_4) = (0, 0, 1, 0)$
$\{r_2\}_{\textcolor{blue}{N1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{2}, 0, \frac{1}{2}, 0\right)$
$\{r_2\}_{\textcolor{blue}{N2}}$	$(r_1, r_2, r_3, r_4) = \left(0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$
$\{r_1\}_{\textcolor{red}{M1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_1\}_{\textcolor{red}{M2}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_2\}_{\textcolor{blue}{N1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_3\}_{\textcolor{blue}{N2}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_1\}_{\textcolor{red}{M1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_2\}_{\textcolor{red}{M2}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_3\}_{\textcolor{blue}{N1}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$
$\{r_4\}_{\textcolor{blue}{N2}}$	$(r_1, r_2, r_3, r_4) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$

3.5.3 Sequential Game

Kuhn Poker. We consider a variant of Kuhn Poker (see Appendix C.2.1) extended to a four-player, two-team setting, which is a finite, sequential, imperfect-information team game without any communication. Four players are partitioned into two teams: Team **M** consisting of players **M1** and **M2**, and Team **N** consisting of players **N1** and **N2**. Each player receives a private card and acts sequentially (**M1**, **N1**, **M2**, **N2**) with check/bet and call/fold options. Team utility equals the sum of its two members’ net winnings, making the game zero-sum between the two teams.

Due to the complexity of *Kuhn Poker*, we cannot exactly obtain the Nash equilibrium, so the learning dynamics using exploitability as the metric is employed to show the performance of proposed algorithm and classical PSRO (Lanctot et al. 2017). Suppose that $\sigma = (\sigma_{\textcolor{red}{M1}}, \sigma_{\textcolor{red}{M2}}, \sigma_{\textcolor{blue}{N1}}, \sigma_{\textcolor{blue}{N2}})$ is the policy profile output, the exploitability is defined as the

Table 3.2: Proposed algorithm's Nash equilibrium with 5 different initial policy populations.

Initial policy population	Nash equilibrium
$\{r_1\}_{M1}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{2}, 0, \frac{1}{2}, 0)$
$\{r_1\}_{M2}$	$(r_1, r_2, r_3, r_4) = (0, \frac{1}{2}, 0, \frac{1}{2})$
$\{r_1\}_{N1}$	$(r_1, r_2, r_3, r_4) = (0, 1, 0, 0)$
$\{r_1\}_{N2}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{3}, 0, \frac{1}{3}, \frac{1}{3})$
$\{r_1\}_{M1}$	$(r_1, r_2, r_3, r_4) = (0, 1, 0, 0)$
$\{r_1\}_{M2}$	$(r_1, r_2, r_3, r_4) = (1, 0, 0, 0)$
$\{r_1\}_{N1}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{2}, 0, \frac{1}{2}, 0)$
$\{r_2\}_{N2}$	$(r_1, r_2, r_3, r_4) = (0, 1, 0, 0)$
$\{r_1\}_{M1}$	$(r_1, r_2, r_3, r_4) = (0, 1, 0, 0)$
$\{r_1\}_{M2}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{3}, 0, \frac{1}{3}, \frac{1}{3})$
$\{r_2\}_{N1}$	$(r_1, r_2, r_3, r_4) = (0, 1, 0, 0)$
$\{r_2\}_{N2}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$
$\{r_1\}_{M1}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{3}, \frac{2}{3}, 0, 0)$
$\{r_1\}_{M2}$	$(r_1, r_2, r_3, r_4) = (0, 0, \frac{1}{2}, \frac{1}{2})$
$\{r_2\}_{N1}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{2}, \frac{1}{2}, 0, 0)$
$\{r_3\}_{N2}$	$(r_1, r_2, r_3, r_4) = (0, 0, \frac{1}{2}, \frac{1}{2})$
$\{r_1\}_{M1}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{2}, 0, \frac{1}{2}, 0)$
$\{r_2\}_{M2}$	$(r_1, r_2, r_3, r_4) = (0, \frac{1}{2}, 0, \frac{1}{2})$
$\{r_3\}_{N1}$	$(r_1, r_2, r_3, r_4) = (\frac{1}{2}, 0, \frac{1}{2}, 0)$
$\{r_4\}_{N2}$	$(r_1, r_2, r_3, r_4) = (0, \frac{1}{2}, 0, \frac{1}{2})$

following

$$\text{exploitability} = \sum_{j \in \{M1, M2, N1, N2\}} V_j(\mathbb{BR}(\sigma_{-j}), \sigma_{-j}) - V_j(\sigma),$$

which measures the total incentives by deviating. The exploitability is defined as the sum of the profits of all players to deviate unilaterally from the current policy profile. Intuitively, it measures how much each player can improve their payoff by switching to a best response while the other players keep their strategies fixed. Therefore, a smaller exploitability value indicates that the current policy profile is closer to a Nash equilibrium, while exploitability equal to zero implies that no player has an incentive to deviate, i.e., the profile is a Nash equilibrium. Since both methods are evaluated under the same payoff scale, raw exploitability values are reported directly, where smaller values indicate closer

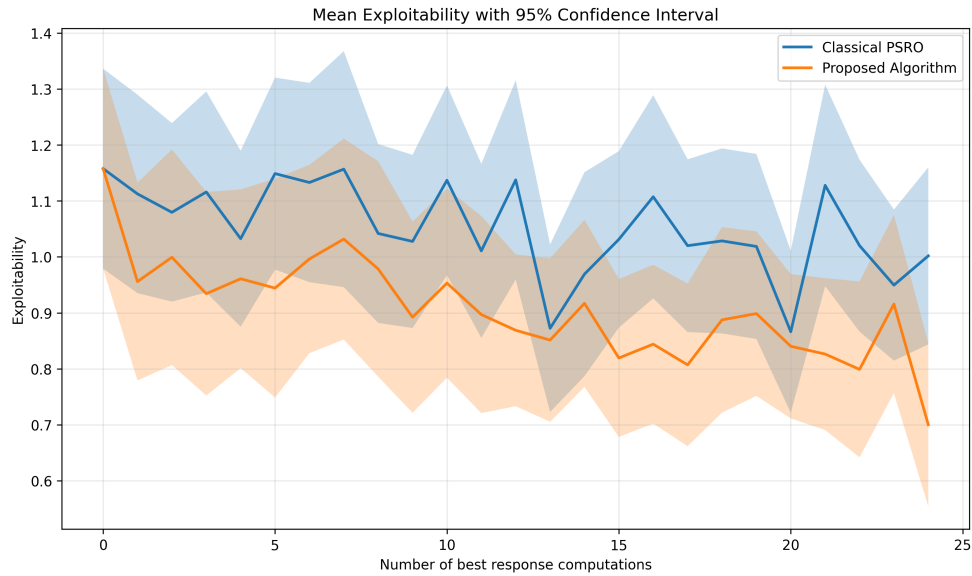
approximation to Nash equilibrium.

We approximate the exploitability by PPO in 50 repetitions. To keep the running time manageable, the policy population is expanded for 25 times, that is, the number of best response computations is 25 for each approximation. We then report the evolution of the mean exploitability (shaded regions represent the corresponding 95% confidence intervals) over iterations, averaged across 75 simulations with different initial policy populations in Figure 3.3. Three examples among these simulations are also presented in Appendix C.2.2.

In Figure 3.3, classical PSRO (blue curve) exhibits persistent oscillations in exploitability throughout the learning process. In contrast, the proposed algorithm (orange curve) demonstrates a more consistent downward trend in exploitability, despite occasional minor fluctuations. Notably, throughout all iterations, the proposed method maintains exploitability at a consistently lower level than classical PSRO, suggesting superior convergence stability and more reliable equilibrium approximation. Overall, these results demonstrate that the proposed algorithm provides both improved stability and stronger robustness in approximating equilibrium strategies.

The different convergence behaviour stems from the policy population expansion mechanism. Classical PSRO expands each player’s policy population using individual best responses, which maximise personal payoff, so such self-interested strategy often conflicts with team utility in the cooperative setting. In contrast, the proposed algorithm approximates a team best response—a joint policy that maximises team utility—and uses it to expand the policy population of each team member, which ensures that added policies are aligned with team-level objectives.

Figure 3.3: Learning dynamics of classical PSRO and proposed algorithm



3.6 Conclusion

We propose a novel algorithm for approximating team Nash equilibrium in fully decentralized settings without communication and correlation devices. Unlike the classical Policy Space Response Oracles (PSRO) framework, the proposed method expands the policy population by incorporating team best-response policies. Experimental results on the matrix game demonstrate that our algorithm exhibits a higher tendency to converge to global Nash equilibria compared with classical PSRO. Furthermore, simulation studies on the sequential game indicate that the proposed approach achieves a more stable reduction in exploitability throughout the training process than the classical PSRO.

References

- Amos, B., Xu, L. & Kolter, J. Z. (2017), Input convex neural networks, *in* ‘International conference on machine learning’, PMLR, pp. 146–155.
- Andrews, D. (1991), ‘Asymptotic normality of series estimators for nonparametric and semiparametric regression models’, *Econometrica* **59**(2), 307–345.
- Archer, N. P. & Wang, S. (1993), ‘Application of the back propagation neural network algorithm with monotonicity constraints for two-group classification problems’, *Decision sciences* **24**(1), 60–75.
- Auer, P., Cesa-Bianchi, N., Freund, Y. & Schapire, R. E. (1995), Gambling in a rigged casino: The adversarial multi-armed bandit problem, *in* ‘Proceedings of IEEE 36th annual foundations of computer science’, IEEE, pp. 322–331.
- Bai, Z. & Saranadasa, H. (1996), ‘Effect of high dimension: by an example of a two sample problem’, *Statistica Sinica* pp. 311–329.
- Balduzzi, D., Garnelo, M., Bachrach, Y., Czarnecki, W., Perolat, J., Jaderberg, M. & Graepel, T. (2019), Open-ended learning in symmetric zero-sum games, *in* ‘International Conference on Machine Learning’, PMLR, pp. 434–443.
- Bamberger, S., Heckel, R. & Krahmer, F. (2023), ‘Approximating positive homogeneous functions with scale invariant neural networks’.
- Bellman, R. & Kalaba, R. (1957), ‘Dynamic programming and statistical communication theory’, *Proceedings of the National Academy of Sciences - PNAS* **43**(8), 749–751.
- Birk, M. & Dette, H. (2007), ‘Estimating a convex function in nonparametric regression’, *Scandinavian journal of statistics* **34**(2), 384–404.

- Blake, C. & Merz, C. (1998), ‘Uci repository of machine learning databases’.
URL: <http://www.ics.uci.edu/mlearn>
- Brown, G. W. (1951), ‘Iterative solution of games by fictitious play’, *Act. Anal. Prod Allocation* **13**(1), 374.
- Brown, N. & Sandholm, T. (2018), ‘Superhuman ai for heads-up no-limit poker: Libratus beats top professionals’, *Science* **359**(6374), 418–424.
- Brown, N. & Sandholm, T. (2019), ‘Superhuman ai for multiplayer poker’, *Science* **365**(6456), 885–890.
- Calafiore, G. C., Gaubert, S. & Possieri, C. (2020), ‘Log-sum-exp neural networks and posynomial models for convex and log-log-convex data’, *IEEE transaction on neural networks and learning systems* **31**(3), 827–838.
- Calhoun, G. (2011), ‘Hypothesis testing in linear regression when k / n is large’, *Journal of econometrics* **165**(2), 163–174.
- Chen, X. & White, H. (1999), ‘Improved rates and asymptotic normality for nonparametric neural network estimators’, *IEEE transactions on information theory* **45**(2), 682–691.
- Chen, Y., Ma, J., Gu, N., Wang, X. & He, M. (2023), Mining and injecting legal prior knowledge to improve the generalization ability of neural networks in chinese judgments, in ‘Artificial Neural Networks and Machine Learning - ICANN 2023’, Vol. 14258 of *Lecture Notes in Computer Science*, Springer, Switzerland, pp. 350–362.
- Chernozhukov, V., Chetverikov, D. & Kato, K. (2017), ‘Central limit theorems and bootstrap in high dimensions’, *The Annals of probability* **45**(4), 2309–2352.
- Chi, C.-M., Vossler, P., Fan, Y. & Lv, J. (2022), ‘Asymptotic properties of high-dimensional random forests’, *The Annals of statistics* **50**(6), 3415–.
- Cole, G. W. & Williamson, S. A. (2019), ‘Avoiding resentment via monotonic fairness’.
- Daniels, H. & Velikova, M. (2010), ‘Monotone and partially monotone neural networks’, *IEEE transactions on neural networks* **21**(6), 906–917.
- Daskalakis, C., Goldberg, P. W. & Papadimitriou, C. H. (2009), ‘The complexity of computing a nash equilibrium’, *Communications of the ACM* **52**(2), 89–97.

- Davidson, R. & MacKinnon, J. G. (2006), ‘The power of bootstrap and asymptotic tests’, *Journal of Econometrics* **133**(2), 421–441.
- De Moor, B. (2005), ‘Daisy: Database for the identification of systems’.
URL: <https://homes.esat.kuleuven.be/~smc/daisy/>
- Dempster, A. P. (1958), ‘A high dimensional two sample significance test’, *The Annals of mathematical statistics* **29**(4), 995–1010.
- Dette, H., Neumeyer, N. & Pilz, K. F. (2006), ‘A simple nonparametric estimator of a strictly monotone regression function’, *Bernoulli : official journal of the Bernoulli Society for Mathematical Statistics and Probability* **12**(3), 469–490.
- Dette, H. & Scheder, R. (2006), ‘Strictly monotone and smooth nonparametric regression for two or more variables’, *Canadian journal of statistics* **34**(4), 535–561.
- Dönmez, A., Arslantaş, Y. & Sayin, M. (2024), ‘Team-fictitious play for reaching team-nash equilibrium in multi-team games’, *Advances in Neural Information Processing Systems* **37**, 111515–111543.
- Donoho, D. L. et al. (2000), ‘High-dimensional data analysis: The curses and blessings of dimensionality’, *AMS math challenges lecture* **1**(2000), 32.
- Duan, Z., Huang, W., Zhang, D., Du, Y., Wang, J., Yang, Y. & Deng, X. (2021), ‘Is nash equilibrium approximator learnable?’, *arXiv preprint arXiv:2108.07472*.
- Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I. et al. (2018), Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures, *in* ‘International conference on machine learning’, PMLR, pp. 1407–1416.
- Feelders, A. J. (2000), Prior knowledge in economic applications of data mining, *in* ‘European Conference on Principles of Data Mining and Knowledge Discovery’, Springer, pp. 395–400.
- Finn, C., Levine, S. & Abbeel, P. (2016), Guided cost learning: Deep inverse optimal control via policy optimization, *in* ‘International conference on machine learning’, PMLR, pp. 49–58.
- Fisher, R. A. & Tippett, L. H. C. (1928), ‘Limiting forms of the frequency distribution of the largest or smallest member of a sample’, *Mathematical proceedings of the Cambridge Philosophical Society* **24**(2), 180–190.

- Foerster, J., Assael, I. A., De Freitas, N. & Whiteson, S. (2016), ‘Learning to communicate with deep multi-agent reinforcement learning’, *Advances in neural information processing systems* **29**.
- Foerster, J., Farquhar, G., Afouras, T., Nardelli, N. & Whiteson, S. (2018a), Counterfactual multi-agent policy gradients, in ‘Proceedings of the AAAI conference on artificial intelligence’, Vol. 32.
- Foerster, J., Farquhar, G., Afouras, T., Nardelli, N. & Whiteson, S. (2018b), Counterfactual multi-agent policy gradients, in ‘Proceedings of the AAAI conference on artificial intelligence’, Vol. 32.
- Fu, J., Luo, K. & Levine, S. (2017), ‘Learning robust rewards with adversarial inverse reinforcement learning’, *arXiv preprint arXiv:1710.11248*.
- Fu, R., Huang, T., Li, M., Sun, Q. & Chen, Y. (2023), ‘A multimodal deep neural network for prediction of the driver’s focus of attention based on anthropomorphic attention mechanism and prior knowledge’, *Expert systems with applications* **214**, 119157–.
- Gao, C., Yin, S., Wang, H., Wang, Z., Du, Z. & Li, X. (2023), ‘Medical-knowledge-based graph neural network for medication combination prediction’, *IEEE transaction on neural networks and learning systems* **PP**, 1–12.
- Gromov, M. & Milman, V. D. (1983), ‘A topological application of the isoperimetric inequality’, *American Journal of Mathematics* **105**(4), 843–854.
- Gupta, A., Shukla, N., Marla, L., Kolbeinsson, A. & Yellepeddi, K. (2019), ‘How to incorporate monotonicity in deep networks while preserving flexibility?’.
- Hart, S. & Mas-Colell, A. (2000), ‘A simple adaptive procedure leading to correlated equilibrium’, *Econometrica* **68**(5), 1127–1150.
- Hart, S. & Mas-Colell, A. (2001), A reinforcement procedure leading to correlated equilibrium, in ‘Economics Essays: A Festschrift for Werner Hildenbrand’, Springer, pp. 181–200.
- He, K., Zhang, X., Ren, S. & Sun, J. (2016), Deep residual learning for image recognition, in ‘Proceedings of the IEEE conference on computer vision and pattern recognition’, pp. 770–778.
- He, X. & Shao, Q.-M. (2000), ‘On parameters of increasing dimensions’, *Journal of multivariate analysis* **73**(1), 120–135.

- Hennes, D., Morrill, D., Omidshafiei, S., Munos, R., Perolat, J., Lanctot, M., Gruslys, A., Lespiau, J.-B., Parmas, P., Duéñez-Guzmán, E. et al. (2020), Neural replicator dynamics: Multiagent learning via hedging policy gradients, *in* ‘Proceedings of the 19th international conference on autonomous agents and multiagent systems’, pp. 492–501.
- Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N. et al. (2012), ‘Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups’, *IEEE Signal processing magazine* **29**(6), 82–97.
- Horel, E. & Giesecke, K. (2020), ‘Significance tests for neural networks’, *Journal of Machine Learning Research* **21**(227), 1–29.
- Horowitz, J. L. (2019), ‘Bootstrap methods in econometrics’, *Annual Review of Economics* **11**, 193–224.
- Hosna, A., Merry, E., Gyalmo, J., Alom, Z., Aung, Z. & Azim, M. A. (2022), ‘Transfer learning: a friendly introduction’, *Journal of Big Data* **9**(1), 102.
- Huber, P. J. (1973), ‘Robust regression: Asymptotics, conjectures and monte carlo’, *The Annals of statistics* **1**(5), 799–821.
- Jimenez, L. & Landgrebe, D. (1998), ‘Supervised classification in high-dimensional space: geometrical, statistical, and asymptotical properties of multivariate data’, *IEEE transactions on systems, man and cybernetics. Part C, Applications and reviews* **28**(1), 39–54.
- Jorge, E., Kågebäck, M., Johansson, F. D. & Gustavsson, E. (2016), ‘Learning to play guess who? and inventing a grounded language as a consequence’, *arXiv preprint arXiv:1611.03218*.
- Kim, I. & Ramdas, A. (2024), ‘Dimension-agnostic inference using cross u-statistics’, *Bernoulli* **30**(1), 683–711.
- Kohler, M. & Langer, S. (2021), ‘On the rate of convergence of fully connected deep neural network regression estimates’, *The Annals of statistics* **49**(4), 2231–2249.
- Kraemer, L. & Banerjee, B. (2016), ‘Multi-agent reinforcement learning as a rehearsal for decentralized planning’, *Neurocomputing* **190**, 82–94.
- Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012), Imagenet classification with deep convolutional neural networks, *in* F. Pereira, C. Burges, L. Bottou & K. Weinberger,

- eds, ‘Advances in Neural Information Processing Systems’, Vol. 25, Curran Associates, Inc.
- URL:** https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf
- Kuhn, H. W. & Tucker, A. W. (1953), *Contributions to the Theory of Games*, number 28, Princeton University Press.
- Lanctot, M., Zambaldi, V., Gruslys, A., Lazaridou, A., Tuyls, K., Pérolat, J., Silver, D. & Graepel, T. (2017), ‘A unified game-theoretic approach to multiagent reinforcement learning’, *Advances in neural information processing systems* **30**.
- Lang, J., Yang, L.-Z. & Li, H. (2023), ‘Tsp-gnn: a novel neuropsychiatric disorder classification framework based on task-specific prior knowledge and graph neural network’, *Frontiers in neuroscience* **17**, 1288882–1288882.
- Lehmann, E. L. E. L. (1999), *Elements of large-sample theory*, Springer texts in statistics., Springer, New York.
- Lindsay, B. G., Kettenring, J. & Siegmund, D. O. (2004), ‘A report on the future of statistics’, *Statistical science* **19**(3), 387–407.
- Liu, N., Wang, M., Wang, X., Zhang, W., Yang, Y., Zhang, Y., An, B. & Wen, Y. (2024), ‘Computing ex ante equilibrium in heterogeneous zero-sum team games’, *arXiv preprint arXiv:2410.01575*.
- Liu, N., Wang, M., Zhang, Y., Yang, Y., An, B. & Wen, Y. (2024), ‘Leveraging team correlation for approximating equilibrium in two-team zero-sum games’, *arXiv preprint arXiv:2403.00255*.
- Liu, Q., Zhang, X. & Liu, Y. (2024), ‘A prior-knowledge-guided neural network based on supervised contrastive learning for radar hrrp recognition’, *IEEE transactions on aerospace and electronic systems* **60**(3), 2854–2873.
- Liu, X., Han, X., Zhang, N. & Liu, Q. (2020), ‘Certified monotonic neural networks’, *Advances in Neural Information Processing Systems* **33**, 15427–15438.
- Lowe, R., Wu, Y. I., Tamar, A., Harb, J., Pieter Abbeel, O. & Mordatch, I. (2017), ‘Multi-agent actor-critic for mixed cooperative-competitive environments’, *Advances in neural information processing systems* **30**.

- McAleer, S., Farina, G., Zhou, G., Wang, M., Yang, Y. & Sandholm, T. (2023), ‘Team-psro for learning approximate tmcpr in large team games via cooperative reinforcement learning’, *Advances in Neural Information Processing Systems* **36**, 45402–45418.
- McMahan, H. B., Gordon, G. J. & Blum, A. (2003), Planning in the presence of cost functions controlled by an adversary, in ‘Proceedings of the 20th International Conference on Machine Learning (ICML-03)’, pp. 536–543.
- Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013), ‘Efficient estimation of word representations in vector space’, *arXiv preprint arXiv:1301.3781*.
- Minin, A., Velikova, M., Lang, B. & Daniels, H. (2010), ‘Comparison of universal approximators incorporating partial monotonicity by structure’, *Neural networks* **23**(4), 471–475.
- Moravčík, M., Schmid, M., Burch, N., Lisý, V., Morrill, D., Bard, N., Davis, T., Waugh, K., Johanson, M. & Bowling, M. (2017), ‘Deepstack: Expert-level artificial intelligence in heads-up no-limit poker’, *Science* **356**(6337), 508–513.
- Muller, P., Omidshafiei, S., Rowland, M., Tuyls, K., Perolat, J., Liu, S., Hennes, D., Marris, L., Lanctot, M., Hughes, E. et al. (2019), ‘A generalized training approach for multiagent learning’, *arXiv preprint arXiv:1909.12823*.
- Munos, R., Stepleton, T., Harutyunyan, A. & Bellemare, M. (2016), ‘Safe and efficient off-policy reinforcement learning’, *Advances in neural information processing systems* **29**.
- Nash, J. F. (2024), Non-cooperative games, in ‘The Foundations of Price Theory Vol 4’, Routledge, pp. 329–340.
- Newey, W. K. (1997), ‘Convergence rates and asymptotic normality for series estimators’, *Journal of econometrics* **79**(1), 147–168.
- Ng, A. Y., Russell, S. et al. (2000), Algorithms for inverse reinforcement learning., in ‘Icml’, Vol. 1, p. 2.
- Oliehoek, F. A., Spaan, M. T. & Vlassis, N. (2008), ‘Optimal and approximate q-value functions for decentralized pomdps’, *Journal of Artificial Intelligence Research* **32**, 289–353.

- Ortega, J. M. & Rheinboldt, W. C. (1970), *Iterative solution of nonlinear equations in several variables*, Computer Science and Applied Mathematics, Academic Press, Inc., San Diego, California ;.
- Perolat, J., De Vylder, B., Hennes, D., Tarassov, E., Strub, F., de Boer, V., Muller, P., Connor, J. T., Burch, N., Anthony, T. et al. (2022), ‘Mastering the game of stratego with model-free multiagent reinforcement learning’, *Science* **378**(6623), 990–996.
- Pomerleau, D. A. (1991), ‘Efficient training of artificial neural networks for autonomous navigation’, *Neural computation* **3**(1), 88–97.
- Portnoy, S. (1984), ‘Asymptotic behavior of m -estimators of p regression parameters when p^2/n is large. i. consistency’, *The Annals of statistics* **12**(4), 1298–1309.
- Portnoy, S. (1985), ‘Asymptotic behavior of m estimators of p regression parameters when p^2/n is large; ii. normal approximation’, *The Annals of statistics* **13**(4), 1403–1417.
- Portnoy, S. (1988), ‘Asymptotic behavior of likelihood methods for exponential families when the number of parameters tends to infinity’, *The annals of statistics* pp. 356–366.
- Rashid, T., Samvelyan, M., De Witt, C. S., Farquhar, G., Foerster, J. & Whiteson, S. (2020), ‘Monotonic value function factorisation for deep multi-agent reinforcement learning’, *Journal of Machine Learning Research* **21**(178), 1–51.
- Reddy, S., Dragan, A. D. & Levine, S. (2019), ‘Sqil: Imitation learning via reinforcement learning with sparse rewards’, *arXiv preprint arXiv:1905.11108* .
- Ribani, R. & Marengoni, M. (2019), A survey of transfer learning for convolutional neural networks, in ‘2019 32nd SIBGRAPI conference on graphics, patterns and images tutorials (SIBGRAPI-T)’, IEEE, pp. 47–57.
- Ross, S., Gordon, G. & Bagnell, D. (2011), A reduction of imitation learning and structured prediction to no-regret online learning, in ‘Proceedings of the fourteenth international conference on artificial intelligence and statistics’, JMLR Workshop and Conference Proceedings, pp. 627–635.
- Rowell, J. G. & Walters, D. E. (1976), ‘Analysing data with repeated observations on each experimental unit’, *The Journal of agricultural science* **87**(2), 423–432.
- Runje, D. & Shankaranarayana, S. M. (2023), Constrained monotonic neural networks, in ‘International Conference on Machine Learning’, PMLR, pp. 29338–29353.

- Schmidt-Hieber, J. (2020), ‘Rejoinder: Nonparametric regression using deep neural networks with relu activation function’, *The Annals of statistics* **48**(4), 1916–1921.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. (2017), ‘Proximal policy optimization algorithms’.
URL: <https://arxiv.org/abs/1707.06347>
- Sill, J. (1997), ‘Monotonic networks’, *Advances in neural information processing systems* **10**.
- Sill, J. & Abu-Mostafa, Y. (1996), ‘Monotonicity hints’, *Advances in neural information processing systems* **9**.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M. et al. (2016), ‘Mastering the game of go with deep neural networks and tree search’, *nature* **529**(7587), 484–489.
- Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T. et al. (2018), ‘A general reinforcement learning algorithm that masters chess, shogi, and go through self-play’, *Science* **362**(6419), 1140–1144.
- Sur, P. & Candès, E. J. (2019), ‘A modern maximum-likelihood theory for high-dimensional logistic regression’, *Proceedings of the National Academy of Sciences - PNAS* **116**(29), 14516–14525.
- Sur, P., Chen, Y. & Candès, E. J. (2019), ‘The likelihood ratio test in high-dimensional logistic regression is asymptotically a rescaled chi-square’, *Probability theory and related fields* **175**(1-2), 487–558.
- Sutton, R. S. (1988), ‘Learning to predict by the methods of temporal differences’, *Machine learning* **3**(1), 9–44.
- Sutton, R. S., McAllester, D., Singh, S. & Mansour, Y. (1999), ‘Policy gradient methods for reinforcement learning with function approximation’, *Advances in neural information processing systems* **12**.
- Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C. & Liu, C. (2018), A survey on deep transfer learning, in ‘Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27’, Springer, pp. 270–279.

- Tang, H., Huang, Z., Gu, J., Lu, B.-L. & Su, H. (2020), ‘Towards scale-invariant graph-related problem solving by iterative homogeneous gnns’, *Advances in Neural Information Processing Systems* **33**, 15811–15822.
- Taylor, P. D. & Jonker, L. B. (1978), ‘Evolutionary stable strategies and game dynamics’, *Mathematical biosciences* **40**(1-2), 145–156.
- Timbers, F., Bard, N., Lockhart, E., Lanctot, M., Schmid, M., Burch, N., Schrittwieser, J., Hubert, T. & Bowling, M. (2020), ‘Approximate exploitability: Learning a best response in large games’, *arXiv preprint arXiv:2004.09677*.
- Tripathi, G. (1997), Semiparametric efficiency bounds and testing in models with shape restrictions, PhD thesis, Northwestern University.
- Tripathi, G. & Kim, W. (2003), ‘Nonparametric estimation of homogeneous functions’, *Econometric theory* **19**(4), 640–663.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. & Polosukhin, I. (2017), ‘Attention is all you need’, *Advances in neural information processing systems* **30**.
- Vershynin, R. (2018), *High-dimensional probability : an introduction with applications in data science*, Cambridge series in statistical and probabilistic mathematics ; 47, Cambridge University Press, Cambridge.
- Vershynin, R. (2020), ‘High-dimensional probability’, *University of California, Irvine* **10**, 11.
- Warin, X. (2023), ‘The groupmax neural network approximation of convex functions’, *IEEE transaction on neural networks and learning systems* **PP**, 1–5.
- Xu, W. (2024-08-19), ‘Applications of machine learning on econometrics for two-stage regression, bias-adjusted inference with unobserved confounding, and test for high dimensionality’, *Dissertation, University of Waterloo*.
- Xu, Z., Yu, C., Liang, Y., Wu, Y. & Wang, Y. (2025a), ‘Learning global nash equilibrium in team competitive games with generalized fictitious cross-play’, *Journal of Machine Learning Research* **26**(44), 1–30.
- Xu, Z., Yu, C., Liang, Y., Wu, Y. & Wang, Y. (2025b), ‘Learning global nash equilibrium in team competitive games with generalized fictitious cross-play’, *Journal of Machine Learning Research* **26**(44), 1–30.

- Yang, D. & Fang, H. (2023), ‘A low-latitude three-dimensional ionospheric electron density model based on radio occultation data using artificial neural networks with prior knowledge’, *Space Weather* **21**(1).
- Yao, J., Liu, W., Fu, H., Yang, Y., McAleer, S., Fu, Q. & Yang, W. (2023), ‘Policy space diversity for non-transitive games’, *Advances in Neural Information Processing Systems* **36**, 67771–67793.
- You, S., Ding, D., Canini, K., Pfeifer, J. & Gupta, M. (2017), ‘Deep lattice networks and partial monotonic functions’, *Advances in neural information processing systems* **30**.
- Yu, C., Velu, A., Vinitzky, E., Gao, J., Wang, Y., Bayen, A. & Wu, Y. (2022), ‘The surprising effectiveness of ppo in cooperative multi-agent games’, *Advances in neural information processing systems* **35**, 24611–24624.
- Zhang, H. & Zhang, Z. (1999), Feedforward networks with monotone constraints, in ‘IJCNN’99. International Joint Conference on Neural Networks. Proceedings (Cat. No.99CH36339)’, Vol. 3, pp. 1820–1823 vol.3.
- Zhang, Y., An, B. & Černý, J. (2021), Computing ex ante coordinated team-maxmin equilibria in zero-sum multiplayer extensive-form games, in ‘Proceedings of the AAAI Conference on Artificial Intelligence’, Vol. 35, pp. 5813–5821.
- Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.-Y. & Wen, J.-R. (2023), ‘A survey of large language models’.

APPENDICES

Appendix A

Proofs and Supplementary Text in Chapter 1

A.1 Proofs

A.1.1 Proof of Lemma 1.2.1

Firstly, define $F(\beta) = \sum_{n=1}^N \mathbf{X}_n (\epsilon_n - \mathbf{X}_n^\top \beta)$. Without loss of generality, assume $\beta = 0$. Then there is a root $\hat{\beta}$ of the equation $F(\beta) = 0$ satisfying $\|\hat{\beta}\|^2 = O_p(D/N)$ if the result 6.3.4 of Ortega & Rheinboldt (1970) holds. Thus, it is equivalent to show that $\beta^\top F(\beta) < 0$ for $\|\beta\|^2 = BD/N$ (Portnoy 1984), where

$$\beta^\top F(\beta) = \sum_{n=1}^N \mathbf{X}_n^\top \beta (\epsilon_n - \mathbf{X}_n^\top \beta) = \sum_{n=1}^N \mathbf{X}_n^\top \beta \epsilon_n - \beta^\top \mathbf{X}_n \mathbf{X}_n^\top \beta =: M_1 - M_2.$$

We know that $M_1 \leq \|\beta\| \|\sum_{n=1}^N \mathbf{X}_n \epsilon_n\|$. Under Assumption 1.2.1, there is

$$\mathbb{E} \left[\left\| \sum_{n=1}^N \mathbf{X}_n \epsilon_n \right\|^2 \right] = \sum_{d=1}^D \sum_{n=1}^N \mathbf{X}_{n,d}^2 \mathbb{E} [\epsilon_n^2] \leq BND.$$

Therefore, Chebychev's inequality implies that for any $\epsilon > 0$ there is a constant C_1 , such that for all N

$$\mathbb{P} \left\{ M_1 \leq C_1 \sqrt{ND} \|\beta\| \text{ for all } \beta \right\} \geq 1 - \epsilon.$$

For M_2 , Assumption 1.2.2 implies that there is a constant δ such that

$$\sum_{n=1}^N \beta^\top \mathbf{X}_n \mathbf{X}_n^\top \beta \geq bN \|\beta\|^2$$

for all β with $\|\beta\| \leq \delta$. As a result, there is N_0 such that for $N \geq N_0$

$$\mathbb{P}\{M_1 - M_2 \leq C_1 \sqrt{ND} \|\beta\| - bN \|\beta\|^2 \text{ for all } \beta \text{ with } \|\beta\| \leq \delta\} \geq 1 - \epsilon.$$

Let $\sqrt{C} = 2C_1/b$ and choose $N'_0 > N_0$ such that $C(D/N) \leq \delta^2$, for $N \geq N'_0$, then

$$\begin{aligned} & \mathbb{P}\{M_1 - M_2 \leq 0 \text{ for all } \beta \text{ with } \|\beta\|^2 = CD/N\} \\ & \geq \mathbb{P}\left\{M_1 - M_2 \leq -\frac{1}{2}BbD \text{ for all } \beta \text{ with } \|\beta\|^2 = CD/N\right\} \geq 1 - \epsilon. \end{aligned}$$

□

A.1.2 Proof of Lemma 1.2.2

After moving N inside the squared norm, we then rewrite

$\sigma^{-1} (N^{-1} \mathbb{X}^\top \mathbb{X})^{\frac{1}{2}} \sqrt{N} (\hat{\beta} - \beta_0)$ as the following

$$\begin{aligned} & \sigma^{-1} (N^{-1} \mathbb{X}^\top \mathbb{X})^{\frac{1}{2}} \sqrt{N} (\hat{\beta} - \beta_0) = \sigma^{-1} (N^{-1} \mathbb{X}^\top \mathbb{X})^{-1/2} \frac{1}{\sqrt{N}} \sum_{n=1}^N \mathbf{X}_n \epsilon_n \\ & = \sigma^{-1} \Sigma^{-1/2} \frac{1}{\sqrt{N}} \sum_{n=1}^N \mathbf{X}_n \epsilon_n + \sigma^{-1} \left((N^{-1} \mathbb{X}^\top \mathbb{X})^{-1/2} - \Sigma^{-1/2} \right) \frac{1}{\sqrt{N}} \sum_{n=1}^N \mathbf{X}_n \epsilon_n \\ & = : (a) + (b). \end{aligned}$$

For (a), under Assumption 1.2.3, high-dimensional CLT implies that $\frac{1}{\sqrt{N}} \sum_{n=1}^N \mathbf{X}_n \epsilon_n$ converges in distribution to a normal random vector with the covariance matrix $\sigma^2 \Sigma$, so we have

$$\frac{\|(a)\|^2 - D}{\sqrt{2D}} \xrightarrow{d} \mathcal{N}(0, 1),$$

by continuous mapping theorem. Then, by writing

$$\sqrt{\frac{D}{2}} \left(\frac{N}{D} \left\| \sigma^{-1} (N^{-1} \mathbb{X}^\top \mathbb{X})^{1/2} (\hat{\beta} - \beta_0) \right\|_2^2 - 1 \right) = \frac{\|(a)\|^2 - D}{\sqrt{2D}} + \frac{\|(b)\|^2 + (a)^\top (b)}{\sqrt{D}}.$$

We are left to show $\|e\|^2 = o_p(D^{1/2})$. Assumption 1.2.3 and 1.2.4 provide $\|(b)\|^2 = o_p(D^{1/2})$ and $(a)^\top (b) \leq \|(a)\| \|(b)\| = o_p(D^{1/2})$, hence we obtain the desired result. □

A.1.3 Proof of Theorem 1.2.1

Rewrite the statistic $T_{N,D,K,B}^*$ as following.

$$\begin{aligned}
T_{N,D,K,B}^* &= \sqrt{K} \sup_{z \in R} \left(\frac{1}{K} \sum_{k=1}^K \mathbf{1}_{(-\infty, z]}(t_k^*) - \Phi(z) \right) \\
&= \sqrt{K} \sup_{z \in R} \left(\frac{1}{K} \sum_{k=1}^K \mathbf{1}_{(-\infty, z]}(t_k^*) - \mathbb{E}(\mathbf{1}_{(-\infty, z]}(t_k^*)) + \mathbb{E}(\mathbf{1}_{(-\infty, z]}(t_k^*)) - \Phi(z) \right) \\
&= \sqrt{K} \sup_{z \in R} \left(\frac{1}{K} \sum_{k=1}^K \mathbf{1}_{(-\infty, z]}(t_k^*) - \mathbb{E}(\mathbf{1}_{(-\infty, z]}(t_k^*)) \right) \\
&\quad + \sqrt{K} \sup_{z \in R} (\mathbb{E}(\mathbf{1}_{(-\infty, z]}(t_k^*)) - \Phi(z)) \\
&=: (3.a) + (3.b)
\end{aligned}$$

Firstly, show that $t_k^* \xrightarrow{d} \mathcal{N}(0, 1)$. For each β_b^* , we know that

$$\beta_b^* = (\mathbb{X}^\top \mathbb{X})^{-1} \mathbb{X}^\top \mathbf{Y}_b^* = (\mathbb{X}^\top \mathbb{X})^{-1} \mathbb{X}^\top (\mathbb{X} \hat{\beta} + R_N^b \hat{\epsilon}) = \hat{\beta} + (\mathbb{X}^\top \mathbb{X})^{-1} \sum_{n=1}^N \mathbf{X}_n R_n^b \hat{\epsilon}_n.$$

Then, conditional on $D_N := \{\{\mathbf{Y}_n, \mathbf{X}_n\} : n = 1, 2, \dots, N\}$, there is $\mathbb{E}^*[\beta_b^*] = \hat{\beta}$ and

$$\begin{aligned}
\text{var}(\beta_b^*) &= \mathbb{E}^* \left[(\mathbb{X}^\top \mathbb{X})^{-1} \mathbb{X}^\top R_N^b \hat{\epsilon} \hat{\epsilon}^\top R_N^{b\top} \mathbb{X} (\mathbb{X}^\top \mathbb{X})^{-1} \right] \\
&= (\mathbb{X}^\top \mathbb{X})^{-1} \mathbb{E}^* \left[\sum_{n=1}^N (R_n^b)^2 \hat{\epsilon}_n^2 \mathbf{X}_n \mathbf{X}_n^\top \right] (\mathbb{X}^\top \mathbb{X})^{-1} \\
&= (\mathbb{X}^\top \mathbb{X})^{-1} \left(\sum_{n=1}^N \hat{\epsilon}_n^2 \mathbf{X}_n \mathbf{X}_n^\top \right) (\mathbb{X}^\top \mathbb{X})^{-1}.
\end{aligned}$$

Since the high-dimensional CLT (Chernozhukov et al. 2017) implies that

$$[\text{var}(\beta_b^*)]^{-1/2} \sqrt{B} \left(\frac{1}{B} \sum_{b=1}^B \beta_b^* - \hat{\beta} \right) \xrightarrow{d} W,$$

where $W \sim \mathcal{N}(0, I_{D \times D})$, therefore, we have $t_k^* \xrightarrow{d} N(0, 1)$ by continuous mapping theorem.

For (3.a), since the class $\mathbf{C} = \{(-\infty, z] : z \in R\}$ is a Donsker class. It is known that (3.a) converges weakly in $l^\infty(R)$ to a Brownian bridge $B(\mathbb{P}\{t_k^* < z\})$, and so converges to $B(\Phi(z))$ as $D, B \rightarrow \infty$ because $\mathbb{P}\{t_k^* < z\} \rightarrow \Phi(z)$.

For (3.b), we are left to show (3.b) = $o_p(1)$ for any z . Since we know that $\mathbb{P}\{t_k^* < z\} - \Phi(z) \rightarrow 0$ as $D, B \rightarrow \infty$, once the convergence rate is obtained, K can be set comparatively small to achieve our goal. Given

$$\begin{aligned} & \sqrt{K} (\mathbb{P}\{t_k^* < z\} - \mathbb{P}\{t^0 < z\} + \mathbb{P}\{t^0 < z\} - \Phi(z)) \\ &= \sqrt{K} (\mathbb{P}\{t_k^* < z\} - \mathbb{P}\{t^0 < z\}) + \sqrt{K} (\mathbb{P}\{t^0 < z\} - \Phi(z)) \\ &=: (3.b.1) + (3.b.2), \end{aligned}$$

where $t_0 = \sqrt{\frac{D}{2}} \left(\frac{1}{D} \sum_{d=1}^D Z_d^2 - 1 \right)$ with $Z_d \sim \mathcal{N}(0, 1)$ for $d = 1, 2, \dots, D$. Based on Berry-Esseen theorem, we have

$$\sup_{z \in R} (3.b.2) \leq C_1 \sqrt{\frac{K}{D}},$$

for all z . For (3.b.1), based on the high-dimensional CLT (Chernozhukov et al. 2017), there is

$$\sup_{z \in R} (3.b.1) = O_p \left(\sqrt{K} \left(\frac{B_N^2 \log^7(DN)}{N} \right)^{1/6} \right).$$

Therefore, we have (3.b) = $o_p(1)$ with $\sqrt{K} \left(\frac{B_N^2 \log^7(DN)}{N} \right)^{1/6} \rightarrow 0$ and $\frac{K}{D} \rightarrow 0$, as required. $\square$

A.1.4 Proof of Lemma 1.3.1

Without loss of generality, assume $\beta_0 = 0$. Similar to Lemma 1.2.1, it suffices to show that $\beta^\top F(\beta) < 0$ for $\|\beta\|^2 = BD/N$, where

$$\begin{aligned} \beta^\top F(\beta) &= \sum_{n=1}^N \nabla^\top f_n(\beta) \beta (\epsilon_n + f(\mathbf{X}_n, 0) - f(\mathbf{X}_n, \beta)) \\ &= \sum_{n=1}^N \nabla^\top f_n(\beta) \beta \epsilon_n - \sum_{n=1}^N \nabla^\top f_n(\beta) \beta (f(\mathbf{X}_n, \beta) - f(\mathbf{X}_n, 0)) \\ &=: M_1 - M_2. \end{aligned}$$

For M_1 , $M_1 \leq \|\beta\| \left\| \sum_{n=1}^N \nabla f_n(\beta) \epsilon_n \right\|$, and under Assumption 1.3.1, there is

$$\mathbb{E} \left\| \sum_{n=1}^N \nabla f_n(\beta) \epsilon_n \right\|^2 = \sum_{d=1}^D \sum_{n=1}^N (\nabla f_n(\beta)_d)^2 \mathbb{E} \epsilon_n^2 \leq BND.$$

Therefore, Chebychev's inequality implies that for any $\epsilon > 0$ there is a constant C_1 such that

$$\mathbb{P}\{M_1 \leq C_1 \sqrt{ND} \|\beta\| \text{ for all } \beta\} \geq 1 - \epsilon.$$

For M_2 ,

$$\begin{aligned} M_2 &= \sum_{n=1}^N \nabla^\top f_n(\beta) \beta (f(\mathbf{X}_n, \beta) - f(\mathbf{X}_n, 0)) = \sum_{n=1}^N (\nabla^\top f_n(\beta) \beta) \left(\nabla^\top f_n(\tilde{\beta}) \beta \right) \\ &= \sum_{n=1}^N \beta^\top \nabla f_n(\beta) \nabla^\top f_n(\tilde{\beta}) \beta \geq C_2 N \|\beta\|^2 \text{ (Assumption 1.3.2)} \end{aligned}$$

for all β with $\|\beta\| \leq \delta_2$. After combining the result of M_1 and M_2 , there is N_0 , such that for $n \geq N_0$

$$\mathbb{P}\{M_1 - M_2 \leq C_1 \sqrt{ND} \|\beta\| - C_2 N \|\beta\|^2 \text{ for all } \beta \text{ with } \|\beta\| \leq \delta\} \geq 1 - 2\epsilon.$$

Let $\sqrt{C} = 2C_1/C_2$ and choose $N_1 > N_0$, so that $C(D/N_1) \leq \delta^2$ for $n \geq N_1$. Then we have

$$\begin{aligned} &\mathbb{P}\{M_1 - M_2 \leq 0 \text{ for all } \beta \text{ with } \|\beta\|^2 = CD/N\} \\ &\geq \mathbb{P}\{M_1 - M_2 \leq -1/2BC_2D \text{ for all } \beta \text{ with } \|\beta\|^2 = CD/N\} \geq 1 - 2\epsilon, \end{aligned}$$

as required. □

A.1.5 Proof of Lemma 1.3.2

Recall $S_N(\beta) = \sum_{n=1}^N (Y_n - f(\mathbf{X}_n, \beta))^2$. By taking Taylor expansion for $\frac{\partial S_N}{\partial \beta} \Big|_{\hat{\beta}}$ around the true value β_0 , there is

$$\frac{\partial S_N}{\partial \beta} \Big|_{\hat{\beta}} = \frac{\partial S_N}{\partial \beta} \Big|_{\beta_0} + \frac{\partial^2 S_N}{\partial \beta \partial \beta^\top} \Big|_{\beta^*} (\hat{\beta} - \beta_0),$$

where β_0 is the true value. Since the $\hat{\beta}$ is the minimizer of $S_N(\beta)$, the left hand side of the above equation is zero. Therefore, we obtain

$$-\frac{1}{N} \frac{\partial^2 S_N}{\partial \beta \partial \beta^\top} \Big|_{\beta^*} \sqrt{N} (\hat{\beta} - \beta_0) = \frac{1}{\sqrt{N}} \frac{\partial S_N}{\partial \beta} \Big|_{\beta_0}.$$

Let's first consider $\frac{1}{N} \frac{\partial^2 S_N}{\partial \beta \partial \beta^\top} \Big|_{\beta^*}$. Differentiating it with respect to β yields

$$\frac{1}{N} \frac{\partial^2 S_N}{\partial \beta \partial \beta^\top} = \frac{2}{N} \sum_{n=1}^N \nabla f_n(\beta) \nabla^\top f_n(\beta) - \frac{2}{N} \sum_{n=1}^N [f_n(\beta_0) - f_n(\beta)] \mathbf{H}_{f_n(\beta)} - \frac{2}{N} \sum_{n=1}^N \epsilon_n \mathbf{H}_{f_n(\beta)}.$$

Then, we can rewrite $\frac{1}{N} \frac{\partial^2 S_N}{\partial \beta \partial \beta^\top} \Big|_{\beta^*}$ as $2\mathbf{\Sigma} + (4.a) + (4.b) + (4.c) + (4.d)$ in which

$$(4.a) = 2 \frac{1}{N} \sum_{n=1}^N \nabla f_n(\beta_0) \nabla^\top f_n(\beta_0) - 2\mathbf{\Sigma},$$

$$(4.b) = 2 \frac{1}{N} \sum_{n=1}^N \nabla f_n(\beta^*) \nabla^\top f_n(\beta^*) - 2 \frac{1}{N} \sum_{n=1}^N \nabla f_n(\beta_0) \nabla^\top f_n(\beta_0),$$

$$(4.c) = -\frac{2}{N} \sum_{n=1}^N [f_n(\beta_0) - f_n(\beta^*)] \mathbf{H}_{f_n(\beta^*)},$$

$$(4.d) = -\frac{2}{N} \sum_{n=1}^N \epsilon_n \mathbf{H}_{f_n(\beta^*)},$$

which implies that

$$\sqrt{N} (\hat{\beta} - \beta_0) = (2\mathbf{\Sigma})^{-1} \frac{1}{\sqrt{N}} \frac{\partial S_N}{\partial \beta} \Big|_{\beta_0} + e,$$

where $e = (2\mathbf{\Sigma})^{-1} ((4.a) + (4.b) + (4.c) + (4.d)) \sqrt{N} (\hat{\beta} - \beta_0)$.

With $\hat{\mathbf{\Sigma}} = \frac{1}{N} \sum_{n=1}^N \nabla f_n(\hat{\beta}) \nabla^\top f_n(\hat{\beta})$, $\sigma^{-1} \hat{\mathbf{\Sigma}}^{1/2} \sqrt{N} (\hat{\beta} - \beta_0)$ can be written as (4.e) +

(4.f) + (4.g), where

$$\begin{aligned}(4.e) &= \sigma^{-1} \boldsymbol{\Sigma}^{1/2} (2\boldsymbol{\Sigma})^{-1} \frac{1}{\sqrt{N}} \frac{\partial S_N}{\partial \beta} \Big|_{\beta_0}, \\(4.f) &= \sigma^{-1} \left(\hat{\boldsymbol{\Sigma}}^{1/2} - \boldsymbol{\Sigma}^{1/2} \right) (2\boldsymbol{\Sigma})^{-1} \frac{1}{\sqrt{N}} \frac{\partial S_N}{\partial \beta} \Big|_{\beta_0}, \\(4.g) &= \sigma^{-1} \boldsymbol{\Sigma}^{1/2} e.\end{aligned}$$

Therefore, the main part of Lemma 1.3.2 becomes

$$\sqrt{\frac{D}{2}} \left(\frac{1}{D} \| (4.e) \|^2 - 1 \right) + \frac{e^*}{\sqrt{D}},$$

where e^* consists of the sum of squared norms of (4.f) and (4.g) plus the sum of the inner products $| (4.e)^\top (4.f) |$, $| (4.e)^\top (4.g) |$, and $| (4.f)^\top (4.g) |$. Under Assumption 1.3.3, the high-dimensional CLT (Chernozhukov et al. 2017) and continuous mapping theorem give that

$$\sqrt{\frac{D}{2}} \left(\frac{1}{D} \| (4.e) \|^2 - 1 \right) \xrightarrow{d} \mathcal{N}(0, 1).$$

Finally, Assumptions 1.3.4 and 1.3.5 are sufficient to yield $e^* = o_p(\sqrt{D})$ as required. $\square$

A.1.6 Proof of Theorem 1.3.1

The proof of Theorem 1.3.1 follows by directly combining the arguments in Theorem 1.2.1 and Lemma 1.3.2. $\square$

A.2 Supplementary Text

In this section, we provide one extra design for the linear case and the design for the nonlinear case. For the linear case, we consider the design,

$$\begin{aligned}\text{Design (iii). } \mathbf{X}_n &\sim \mathcal{N}(0, I_{D \times D}) && \text{for all } n < \frac{N}{2}, \\ \mathbf{X}_n &\sim \chi_1^2 && \text{for all } n \geq \frac{N}{2}.\end{aligned}$$

Table A.1: Empirical rejection rates for estimated vectors from **linear** regressions

D	Design (iii)			
	Sample size N			
	100	500	1000	2000
2	1.000	1.000	1.000	1.000
3	0.570	0.607	0.603	0.550
4	0.470	0.467	0.463	0.423
5	0.380	0.400	0.380	0.377
6	0.300	0.283	0.290	0.337
7	0.283	0.290	0.287	0.223
8	0.233	0.217	0.207	0.233
9	0.200	0.230	0.223	0.223
10	0.260	0.197	0.200	0.270
20	0.117	0.177	0.107	0.143
30	0.093	0.093	0.083	0.097
40	0.080	0.130	0.063	0.080
50	0.063	0.073	0.070	0.093
60	0.077	0.083	0.063	0.053
70	0.067	0.043	0.060	0.040
80	0.050	0.060	0.057	0.073
90	0.030	0.053	0.053	0.060
100	-	0.057	0.047	0.053

In this design, we combine the distribution of $\mathbf{X}_n$ in Design (i) and the chi-squared distribution in Design (ii), the result is still consistent, that is, the empirical rejection rates decrease from 1 to 0.05 when the dimension D increases up to 100.

For the nonlinear case, since the restrictions on the nonlinear function and the regressors are strict, we only consider one design, **Design (iv)**, in which

$$Y_n = \frac{1}{1 + e^{-\mathbf{X}_n^\top \beta_0}} + \epsilon_n$$

where $\mathbf{X}_n \sim \mathcal{N}(0, \frac{1}{N})$, $\epsilon_n \sim \mathcal{N}(0, \frac{1}{N})$ and $\beta_0 = (0, 0, \dots, 0)^\top$.

The result in Table A.2 show the empirical rejection rates for estimated vectors from nonlinear regressions. In Design (iii), the result is different from that in the linear case (Table 1.1) when the sample size is 100. Specifically, when $N = 100$, the rejection rate decreases first but increases rapidly up to 1 as D rises beyond 30. This issue is caused by the least square estimator, which does not provide good estimates for such small sample sizes and large dimensions. As we mentioned, one can try other estimators as extensions of our test. However, when the sample size is increased appropriately, this issue no longer exists and the desired result is obtained.

Table A.2: Empirical rejection rates for estimated vectors from **nonlinear** regressions

D	Design (iv)			
	Sample size N			
	100	500	1000	2000
2	1.000	1.000	1.000	1.000
3	0.583	0.573	0.553	0.568
4	0.403	0.436	0.413	0.447
5	0.307	0.360	0.390	0.354
6	0.213	0.317	0.330	0.341
7	0.190	0.233	0.260	0.268
8	0.123	0.230	0.257	0.245
9	0.077	0.200	0.247	0.172
10	0.097	0.223	0.160	0.128
20	0.083	0.130	0.140	0.156
30	1.000	0.113	0.137	0.079
40	1.000	0.067	0.083	0.075
50	1.000	0.083	0.077	0.068
60	1.000	0.090	0.087	0.061
70	1.000	0.047	0.063	0.065
80	1.000	0.057	0.063	0.057
90	1.000	0.060	0.053	0.044
100	-	0.053	0.037	0.041

Appendix B

Proofs and Supplementary Text in Chapter 2

B.1 Proofs

This section consists of two parts. The first part contains the proof that the proposed homogeneous-restricted estimation is homogeneous, as well as the proof that our method can achieve the combination of different shapes regardless of the order. The second part contains the proof of the lemmas, Theorem [2.2.1](#), [2.2.2](#), [2.2.3](#) and [2.2.4](#).

B.1.1 Part I: Shape Constraints

Homogeneity

Fix non-zero $X, aX \in [0, 1]^D$, define two sequences $\{s_j\}_{j=1}^J$ and $\{e_j\}_{j=1}^J$ such that $s_j X = e_j aX$ for $j = 1, 2, \dots, J$. Based on our method, we know that the homogeneous-restricted estimation is

$$\begin{aligned}\hat{y}_1 &= \arg \min_C \sum_{j=1}^J (Cs_j^r - \mathcal{T}(s_j X))^2, \\ \hat{y}_2 &= \arg \min_C \sum_{j=1}^J (Ce_j^r - \mathcal{T}(e_j aX))^2.\end{aligned}$$

We aim to show that $a^r \hat{y}_1 = \hat{y}_2$. Minimizing the above two gives us

$$\hat{y}_1 = \frac{\sum_{j=1}^J s_j^r \mathcal{T}(s_j X)}{\sum_{j=1}^J (s_j^r)^2} \quad \text{and} \quad \hat{y}_2 = \frac{\sum_{j=1}^J e_j^r \mathcal{T}(e_j a X)}{\sum_{j=1}^J (e_j^r)^2}.$$

Since we know that $s_j = e_j a$ for $j = 1, 2, \dots, J$, there is

$$a^r \hat{y}_1 = \frac{\sum_{j=1}^J s_j^r \mathcal{T}(s_j X)}{\sum_{j=1}^J (s_j^r)^2} = \frac{\sum_{j=1}^J (a^r)^2 e_j^r \mathcal{T}(e_j a X)}{\sum_{j=1}^J (e_j^r a^r)^2} = \frac{\sum_{j=1}^J 2e_j^r \mathcal{T}(e_j a X)}{\sum_{j=1}^J (e_j^r)^2} = \hat{y}_2,$$

as required. It means that we can flexibly set the sequence $\{s_j\}_{j=1}^J$ such that $\{s_j a X\}_{j=1}^J$ is located at some fixed points along the direction X for different a . Otherwise, we can increase the size of the sequence so that it can fill up the interval along the direction X . $\square$

Combination of Different Shape Restrictions

We know that monotonicity and convexity are the shape for a single input coordinate, so the combinations within these two can be achieved by the proposed method Dette & Scheder (2006). Now, we consider the combination of monotonicity/convexity and homogeneity. Without loss of generality, suppose that the underlying function f_0 is monotonic/convex with respect the first coordinate, and homogeneous of degree r .

Monotonicity $\rightarrow$ Homogeneity

Based on our method, $\mathbf{M}_{\text{mon}}^1(X; \mathcal{T})$ is increasing with respect to the first coordinate. After applying the method of homogeneity, $\mathbf{M}_{\text{hom}}^r$, for the monotonized approximator $\mathbf{M}_{\text{mon}}^1(X; \mathcal{T})$, the composited approximator is homogeneous of degree r for sure. Therefore, we are left to show that $\mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{mon}}^1(X; \mathcal{T})$ is still increasing with respect to the first coordinate.

For any X_a, X_b such that $X_a = (x_a, x_2, \dots, x_D)$ and $X_b = (x_b, x_2, \dots, x_D)$ where $x_b \geq x_a$, our method provides

$$\begin{aligned} \hat{y}_a &:= \mathbf{M}_{\text{hom}} \circ \mathbf{M}_{\text{mon}}^1(X_a; \mathcal{T}) = \arg \min_C \sum_{j=1}^J (C s_j^r - \mathbf{M}_{\text{mon}}^1(s_j x_a, s_j x_2, \dots, s_j x_D; \mathcal{T}))^2, \\ \hat{y}_b &:= \mathbf{M}_{\text{hom}} \circ \mathbf{M}_{\text{mon}}^1(X_b; \mathcal{T}) = \arg \min_C \sum_{j=1}^J (C s_j^r - \mathbf{M}_{\text{mon}}^1(s_j x_b, s_j x_2, \dots, s_j x_D; \mathcal{T}))^2. \end{aligned}$$

We aim to show $\hat{y}_b \geq \hat{y}_a$. Since $\mathbf{M}_{\text{mon}}^1(X; \mathcal{T})$ is increasing with respect to the first coordinate, we know that $\mathbf{M}_{\text{mon}}^1(s_j x_b, s_j x_2, \dots, s_j x_D; \mathcal{T}) \geq \mathbf{M}_{\text{mon}}^1(s_j x_a, s_j x_2, \dots, s_j x_D; \mathcal{T})$ for all j . Let

$$\begin{aligned} S_{a,j} &:= \mathbf{M}_{\text{mon}}^1(s_j x_a, s_j x_2, \dots, s_j x_D; \mathcal{T}), \\ S_{b,j} &:= \mathbf{M}_{\text{mon}}^1(s_j x_b, s_j x_2, \dots, s_j x_D; \mathcal{T}), \end{aligned}$$

then we have $S_{b,j} \geq S_{a,j}$ for $j = 1, 2, \dots, J$. Eventually, there is

$$\hat{y}_a = \frac{\sum_{j=1}^J s_j^r S_{a,j}}{\sum_{j=1}^J (s_j^r)^2} \text{ and } \hat{y}_b = \frac{\sum_{j=1}^J s_j^r S_{b,j}}{\sum_{j=1}^J (s_j^r)^2}.$$

Since $s_j \geq 0$ for all j based on our method, we have $\hat{y}_b \geq \hat{y}_a$, as required. $\square$

Homogeneity $\rightarrow$ Monotonicity

We aim to show that $\mathbf{M}_{\text{mon}}^1 \circ \mathbf{M}_{\text{hom}}^r(kX; \mathcal{T}) = k^r \mathbf{M}_{\text{mon}}^1 \circ \mathbf{M}_{\text{hom}}^r(X; \mathcal{T})$. We firstly consider the case $k \leq 1$. Let y represent $\mathbf{M}_{\text{mon}}^1 \circ \mathbf{M}_{\text{hom}}^r(kX; \mathcal{T})$. Based on the construction of the method of monotonicity, there is

$$\begin{aligned} & \int_0^1 I_{\{H(kx_1, kx_2, \dots, kx_{i-1}, z, kx_{i+1}, \dots, kx_D) \leq y\}} dz = kx_i, \\ \Rightarrow & \int_0^1 \frac{1}{k} I_{\{H(kx_1, kx_2, \dots, kx_{i-1}, z, kx_{i+1}, \dots, kx_D) \leq y\}} dz = x_i, \\ \Rightarrow & \int_0^1 \frac{1}{k} I_{\{H(x_1, x_2, \dots, x_{i-1}, \frac{z}{k}, x_{i+1}, \dots, x_D) \leq \frac{y}{k^r}\}} dz = x_i, \\ \Rightarrow & \int_0^{\frac{1}{k}} I_{\{H(x_1, x_2, \dots, x_{i-1}, t, x_{i+1}, \dots, x_D) \leq \frac{y}{k^r}\}} dt = x_i \end{aligned}$$

The above implies that $\mathbf{M}_{\text{mon}}^1 \circ \mathbf{M}_{\text{hom}}^r(X; \mathcal{T}) = \frac{y}{k^r} = \mathbf{M}_{\text{mon}}^1 \circ \mathbf{M}_{\text{hom}}^r(kX; \mathcal{T}) / k^r$. The case of $k > 1$ is almost similar. $\square$

Convexity $\rightarrow$ Homogeneity

Here, we need to show that $\mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{con}}^1(X; \mathcal{T})$ is still convex with respect to the first coordinate. For any X_a, X_b and $\theta > 0$ such that $X_a = (x_a, x_2, \dots, x_D)$ and $X_b =$

$(x_b, x_2, \dots, x_D)$ where $x_b \geq x_a$, our method provides

$$\hat{y}_a := \mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{con}}^1(X_a; \mathcal{T}) = \arg \min_C \sum_{j=1}^J (Cs_j^r - \mathbf{M}_{\text{con}}^1(s_j x_a, s_j x_2, \dots, s_j x_D; \mathcal{T}))^2,$$

$$\hat{y}_b := \mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{con}}^1(X_b; \mathcal{T}) = \arg \min_C \sum_{j=1}^J (Cs_j^r - \mathbf{M}_{\text{con}}^1(s_j x_b, s_j x_2, \dots, s_j x_D; \mathcal{T}))^2,$$

$$\hat{y}_{a,b} := \mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{con}}^1(X_b; \mathcal{T}) = \arg \min_C \sum_{j=1}^J (Cs_j^r - \mathbf{M}_{\text{con}}^1(s_j(\theta x_a + (1-\theta)x_b), s_j x_2, \dots, s_j x_D; \mathcal{T}))^2.$$

Aim to show that $\theta \hat{y}_a + (1-\theta) \hat{y}_b \geq \hat{y}_{a,b}$. Firstly, we can solve the above equations and obtain the closed form of $\hat{y}$.

$$\begin{aligned} \hat{y}_a &= \frac{\sum_{j=1}^J s_j^r \mathbf{M}_{\text{con}}^1(s_j x_a, s_j x_2, \dots, s_j x_D; \mathcal{T})}{\sum_{j=1}^J (s_j^r)^2}, \\ \hat{y}_b &= \frac{\sum_{j=1}^J s_j^r \mathbf{M}_{\text{con}}^1(s_j x_b, s_j x_2, \dots, s_j x_D; \mathcal{T})}{\sum_{j=1}^J (s_j^r)^2}, \\ \hat{y}_{a,b} &= \frac{\sum_{j=1}^J s_j^r \mathbf{M}_{\text{con}}^1(s_j(\theta x_a + (1-\theta)x_b), s_j x_2, \dots, s_j x_D; \mathcal{T})}{\sum_{j=1}^J (s_j^r)^2}. \end{aligned}$$

Due to the convexity of $\mathbf{M}_{\text{con}}^1$, there is

$$\begin{aligned} &\theta \hat{y}_a + (1-\theta) \hat{y}_b - \hat{y}_{a,b} \\ &= \frac{1}{\sum_{j=1}^J (s_j^r)^2} \left[\sum_{j=1}^J s_j^r \theta \mathbf{M}_{\text{con}}^1(s_j x_a, s_j x_2, \dots, s_j x_D; \mathcal{T}) + \sum_{j=1}^J s_j^r (1-\theta) \mathbf{M}_{\text{con}}^1(s_j x_b, s_j x_2, \dots, s_j x_D; \mathcal{T}) \right] \\ &\quad - \frac{1}{\sum_{j=1}^J (s_j^r)^2} \left[\sum_{j=1}^J s_j^r \mathbf{M}_{\text{con}}^1(s_j(\theta x_a + (1-\theta)x_b), s_j x_2, \dots, s_j x_D; \mathcal{T}) \right] \\ &= : \frac{1}{\sum_{j=1}^J (s_j^r)^2} \left[\sum_{j=1}^J s_j^r (\theta \mathbf{M}_{\text{con},a}^1 + (1-\theta) \mathbf{M}_{\text{con},b}^1 - \mathbf{M}_{\text{con},a,b}^1) \right]. \end{aligned}$$

Since $\mathbf{M}_{\text{con}}^1$ is convex with respect to the first coordinate, it is clear that

$$(\theta \mathbf{M}_{\text{con},a}^1 + (1-\theta) \mathbf{M}_{\text{con},b}^1 - \mathbf{M}_{\text{con},a,b}^1) \geq 0.$$

As a result, we have $\hat{y}_a + (1 - \theta)\hat{y}_b - \hat{y}_{a,b} \geq 0$. $\square$

Homogeneity $\rightarrow$ Convexity

Need to show that $\mathbf{M}_{\text{con}}^i \circ \mathbf{M}_{\text{hom}}^r(kX; \mathcal{T}) = k^r \mathbf{M}_{\text{con}} \circ \mathbf{M}_{\text{hom}}^r(X; \mathcal{T})$. Firstly, it know that $\frac{\partial \mathbf{M}_{\text{hom}}^r(kX; \mathcal{T})}{\partial \mathbf{x}_1} = k^{r-1} \frac{\partial \mathbf{M}_{\text{hom}}^r(X; \mathcal{T})}{\partial \mathbf{x}_1}$ which implies that $\frac{\partial \mathbf{M}_{\text{hom}}^r(X; \mathcal{T})}{\partial \mathbf{x}_i}$, denoted as $H(X)$, is homogeneous of degree $r - 1$. Based on the construction of our method of convexity and the result from the above sections, there is

$$\mathbf{M}_{\text{con}}^1 \circ \mathbf{M}_{\text{hom}}^r(X; \mathcal{T}) = \mathbf{M}_{\text{hom}}^r(u_0, x_2, \dots, x_D; \mathcal{T}) + \int_{u_0}^{x_1} \mathbf{M}_{\text{mon}}^1 \circ H(z, x_2, \dots, x_D) dz.$$

Then there is

$$\begin{aligned} & \mathbf{M}_{\text{con}} \circ \mathbf{M}_{\text{hom}}^r(kX; \mathcal{T}) \\ &= \mathbf{M}_{\text{hom}}^r(u_0, kx_2, \dots, kx_D; \mathcal{T}) + \int_{u_0}^{kx_1} \mathbf{M}_{\text{mon}}^1 \circ H(z, kx_2, \dots, kx_D) dz \\ &= k^r \mathbf{M}_{\text{hom}}^r(u_0/k, x_2, \dots, x_D; \mathcal{T}) + \int_{u_0}^{kx_1} k^{r-1} \mathbf{M}_{\text{mon}}^1 \circ H(z/k, x_2, \dots, x_D) dz \\ &= k^r \mathbf{M}_{\text{hom}}^r(u_0/k, x_2, \dots, x_D; \mathcal{T}) + k^r \int_{u_0/k}^{x_1} \mathbf{M}_{\text{mon}}^1 \circ H(t, x_2, \dots, x_D) dt \\ &= k^r \mathbf{M}_{\text{con}}^1 \circ \mathbf{M}_{\text{hom}}^r(X; \mathcal{T}) + k^r \epsilon \end{aligned}$$

where

$$\epsilon = \mathbf{M}_{\text{hom}}^r(u_0/k, x_2, \dots, x_D; \mathcal{T}) - \mathbf{M}_{\text{hom}}^r(u_0, x_2, \dots, x_D; \mathcal{T}) + \int_{u_0/k}^{u_0} \mathbf{M}_{\text{mon}}^1 \circ H(t, x_2, \dots, x_D) dt$$

Besides, u_0 can be set arbitrarily small to ignore the error term. $\square$

B.1.2 Part II: Rate of Convergence and Asymptotic Normality

Proof of Theorem 2.2.1

Let's start with $D = 1$. Based on the construction of monotonization, for any x , there is

$$\mathbf{M}_{\text{mon}}(x, \hat{f}) - f_0(x) = \Delta_N^{(1)} + \frac{1}{2} \Delta_N^{(2)},$$

where

$$\Delta_N^{(1)} = -\frac{\partial}{\partial x} f_0(x) \left(\hat{f}^{-1} - f_0^{-1} \right) \circ f_0(x),$$

and for some $\lambda \in [0, 1]$,

$$\begin{aligned} \Delta_N^{(2)} = & 2 \frac{\left(\hat{f}^{-1} - f_0^{-1} \right) \frac{\partial}{\partial x} \left(\hat{f}^{-1} - f_0^{-1} \right)}{\frac{\partial}{\partial x} \left(f_0^{-1} + \lambda \left(\hat{f}^{-1} - f_0^{-1} \right) \right)} \circ \left(f_0^{-1} + \lambda \left(\hat{f}^{-1} - f_0^{-1} \right) \right)^{-1}(x) \\ & - \frac{\left(\hat{f}^{-1} - f_0^{-1} \right)^2 \frac{\partial}{\partial x} \left(f_0^{-1} + \lambda \left(\hat{f}^{-1} - f_0^{-1} \right) \right)}{\left[\frac{\partial}{\partial x} \left(f_0^{-1} + \lambda \left(\hat{f}^{-1} - f_0^{-1} \right) \right) \right]^3} \\ & \circ \left(f_0^{-1} + \lambda \left(\hat{f}^{-1} - f_0^{-1} \right) \right)^{-1}(x). \end{aligned}$$

For both $\Delta_N^{(1)}$ and $\Delta_N^{(2)}$, the key point of investigating the rate of convergence is $\hat{f}^{-1} - f_0^{-1}$, which is defined as

$$\int_0^1 I_{\hat{f}(x) \leq t} dx - \int_0^1 I_{f_0(x) \leq t} dx = \int_0^1 I_{\hat{f}(x) \leq t} dx - I_{f_0(x) \leq t} dx,$$

which can be further bounded as

$$\begin{aligned} \left| \int_0^1 I_{\hat{f}(x) \leq t} dx - \int_0^1 I_{f_0(x) \leq t} dx \right| & \leq \int_0^1 \left| I_{\hat{f}(x) \leq t} - I_{f_0(x) \leq t} \right| dx \\ & \leq \int_0^1 I_{|f_0(x) - t| \leq \delta} dx + \int_0^1 I_{\delta < |\hat{f}(x) - f_0(x)|} dx, \end{aligned}$$

since for any $\delta > 0$, there is

$$\left| I_{\hat{f}(x) \leq t} - I_{f_0(x) \leq t} \right| \leq I_{|f_0(x) - t| \leq [\hat{f}(x) - f_0(x)]} \leq I_{|f_0(x) - t| \leq \delta} + I_{\delta < |\hat{f}(x) - f_0(x)|}.$$

If the function f_0 is strictly increasing and its derivative has a lower bound $c > 0$, then we have

$$\int_0^1 I_{|f_0(x) - t| \leq \delta} dx \leq \frac{2\delta}{c}.$$

On the other hand, the Markov's inequality implies that

$$\int_0^1 I_{\delta < |\hat{f}(x) - f_0(x)|} dx \leq \frac{1}{\delta^p} \int_0^1 \left| \hat{f}(x) - f_0(x) \right|^p dx = \frac{\left\| \hat{f} - f_0 \right\|_{L^p}^p}{\delta^p}.$$

Overall, we can obtain the upper bound $2\delta/c + \|\hat{f} - f_0\|_{L^p}^p/\delta^p$ for $|\hat{f}^{-1} - f_0^{-1}|$. Then setting $\delta = (\frac{c}{2}\|\hat{f} - f_0\|_{L^p}^{\frac{p}{p+1}})^{\frac{1}{p+1}}$ gives us

$$\left| \int_0^1 I_{\hat{f}(x) \leq t} dx - \int_0^1 I_{f_0(x) \leq t} dx \right| = O\left(\left\| \hat{f} - f_0 \right\|_{L^p}^{\frac{p}{p+1}}\right)$$

for any t .

Next, with loss of generality, we assume that the dimension is $d = 2$, and the monotonic restriction is imposed to the first coordinate. Based on the result from case with $d = 1$, we know that

$$\begin{aligned} \left| \mathbf{M}_{\text{mon}}(X, \hat{f}) - f_0(X) \right| &= \left| \mathbf{M}_{\text{mon}}(x_1, x_2, \hat{f}) - f_0(x_1, x_2) \right| \\ &\leq C \left(\int_0^1 \left| \hat{f}(x_1, x_2) - f_0(x_1, x_2) \right|^p dx_1 \right)^{\frac{p}{p+1}} \end{aligned}$$

for any X and a constant C . Then, there is

$$\begin{aligned} &\left\| \mathbf{M}_{\text{mon}}(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)}^p \\ &= \int_{[0,1]^2} \left| \mathbf{M}_{\text{mon}}(x_1, x_2, \hat{f}) - f_0(x_1, x_2) \right|^p d\mu(X) \\ &\leq C^p \int_{[0,1]^2} \left(\int_0^1 \left| \hat{f}(x_1, x_2) - f_0(x_1, x_2) \right|^p dx_1 \right)^{\frac{p}{p+1}} d\mu(X) \\ &\leq C^p \left(\int_{[0,1]^2} \left(\int_0^1 \left| \hat{f}(x_1, x_2) - f_0(x_1, x_2) \right|^p dx_1 \right) d\mu(X) \right)^{\frac{p}{p+1}} \end{aligned}$$

The last inequality comes from the Jensen's inequality for concave function $\phi(x) = x^{\frac{p}{p+1}}$. If μ is the probability measure induced by X with the probability density $p_X(x_1, x_2)$, we have

$$\left\| \mathbf{M}_{\text{mon}}(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)}^p \leq C^p \left(\int_0^1 \left(\int_0^1 \left| \hat{f}(x_1, x_2) - f_0(x_1, x_2) \right|^p dx_1 \right) p_2(x_2) dx_2 \right)^{\frac{p}{p+1}},$$

where $p_2(x_2)$ is the marginal probability density function for x_2 . If the preliminary estimator $\hat{f}$ satisfies

$$\int_0^1 \left(\int_0^1 \left| \hat{f}(x_1, x_2) - f_0(x_1, x_2) \right|^p dx_1 \right) p_2(x_2) dx_2 = O \left(\left\| \hat{f} - f \right\|_{L^p(\mu)} \right),$$

we know that Therefore, we know that

$$\left\| \mathbf{M}_{\text{mon}}(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} = O \left(\left\| \hat{f} - f \right\|_{L^p(\mu)}^{\frac{p}{p+1}} \right).$$

□

This result holds for the Lebesgue measure directly.

Proof of Theorem 2.2.2

Firstly, let's define the operator ψ on $C([0, 1])$ s.t. $\psi(g) : R(g) \rightarrow [0, 1]$ with

$$\psi(g)(t) = \int_0^1 I_{g(x) \leq t} dx$$

where $R(g)$ is the range of g . With loss of generality, assume that $d = 2$, and the convex restriction is imposed to the first coordinate. Based on the construction of method for convexity, there is

$$\begin{aligned} & \mathbf{M}_{\text{con}}(X, \hat{f}) - f_0(X) \\ &= \hat{f}(u_0, x_2) - f(u_0, x_2) + \int_{u_0}^{x_1} \left(\int_0^1 I_{\frac{\partial}{\partial x_1} \hat{f}(z, x_2) \leq t} dz - \int_0^1 I_{\frac{\partial}{\partial x_1} f_0(z, x_2) \leq t} dz \right) dt. \end{aligned}$$

We denote $\frac{\partial}{\partial x_1} g(X)$ by $g'(X)$ for simplicity in the following. The case of monotonicity shows that

$$\left| \int_0^1 I_{\hat{f}'(z, x_2) \leq t} dz - \int_0^1 I_{f'_0(z, x_2) \leq t} dz \right| \leq C \left(\int_0^1 \left| \hat{f}'(x_1, x_2) - f'_0(x_1, x_2) \right|^p dx_1 \right)^{\frac{p}{p+1}}$$

for a constant C , which implies that

$$\begin{aligned} & \left\| \mathbf{M}_{\text{con}}(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} \\ & \leq \left\| \hat{f}(u_0, x_2) - f_0(u_0, x_2) \right\|_{L^p(\mu)} + C \left\| \left(\int_0^1 \left| \hat{f}'(x_1, x_2) - f'_0(x_1, x_2) \right|^p dx_1 \right)^{\frac{p}{p+1}} \right\|_{L^p(\mu)}. \end{aligned}$$

If the preliminary estimator $\hat{f}$ satisfies

$$\left\| \hat{f}(u_0, x_2) - f(u_0, x_2) \right\|_{L^p(\mu)} = O \left(\left\| \hat{f} - f_0 \right\|_{L^p(\mu)}^{\frac{p}{p+1}} \right),$$

and its partial derivative satisfies

$$\int_0^1 \left(\int_0^1 \left| \hat{f}'(x_1, x_2) - f_0'(x_1, x_2) \right|^p dx_1 \right) p_2(x_2) dx_2 = O \left(\left\| \hat{f} - f \right\|_{L^p(\mu)} \right),$$

then we can have the corresponding convergence rate

$$\left\| \mathbf{M}_{\text{con}}(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} = O \left(\left\| \hat{f} - f \right\|_{L^p(\mu)}^{\frac{p}{p+1}} \right).$$

□

Proof of Theorem 2.2.3

Based on the construction of our method, we know that, for all X , there is

$$\begin{aligned} \mathbf{M}_{\text{hom}}^r(X, \hat{f}) - f_0(X) &= \frac{\sum_{t=1}^T s_t^r \left(\hat{f}(s_t X) - f_0(s_t X) \right)}{\sum_{t=1}^T (s_t^r)^2} \\ &= \frac{\sum_{t=1}^T (s_t^r)}{\sum_{t=1}^T (s_t^r)^2} \sum_{t=1}^T \frac{s_t^r}{\sum_{t=1}^T s_t^r} \left(\hat{f}(s_t X) - f_0(s_t X) \right) \\ &= \frac{\sum_{t=1}^T (s_t^r)}{\sum_{t=1}^T (s_t^r)^2} \sum_{t=1}^T w_t \left(\hat{f}(s_t X) - f_0(s_t X) \right). \end{aligned}$$

$\sum_{t=1}^T w_t \left(\hat{f}(s_t X) - f_0(s_t X) \right)$ is a weighted sum of $\hat{f}(\cdot) - f_0(\cdot)$, which depends on the non-negative sequence of $\{s_t\}_{t=1}^T$. Then there is

$$\left\| \mathbf{M}_{\text{hom}}^r(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} \leq \frac{\sum_{t=1}^T (s_t^r)}{\sum_{t=1}^T (s_t^r)^2} \sum_{t=1}^T w_t \left\| \hat{f}(s_t X) - f_0(s_t X) \right\|_{L^p(\mu)}.$$

If $\|\hat{f}(aX) - f_0(aX)\|_{L^p(\mu)}$ is dominated by $\|\hat{f}(X) - f_0(X)\|_{L^p(\mu)}$ for any non-negative a , we can directly obtain that

$$\left\| \mathbf{M}_{\text{hom}}^r(X, \hat{f}) - f_0(X) \right\|_{L^p(\mu)} = O\left(\left\| \hat{f}(X) - f_0(X) \right\|_{L^p(\mu)}\right),$$

once the choice of the sequence $\{s_t\}_{t=1}^T$ has bounded $\sum_{t=1}^T (s_t^r) / \sum_{t=1}^T (s_t^r)^2$. $\square$

Proof of Theorem 2.2.4

Asymptotic Normality

Monotonicity

WLOG, we set $D = 1$ for the following. The case of $D > 1$ can be easily derived with the similar argument from Dette & Scheder (2006). Suppose that the underlying function f_0 is increasing. From Newey (1997) & Andrews (1991), we have that

$$\sqrt{n}\hat{V}^{-1/2} \left(\mathbf{M}_{\text{mon}}^1(X, \hat{f}) - f_0(X) \right) \xrightarrow{D} \mathcal{N}(0, 1).$$

where

$$\hat{V} = \hat{A}' \hat{Q}^{-1} \hat{\Sigma} \hat{Q}^{-1} \hat{A}.$$

Specifically,

$$\begin{aligned} \hat{A} &= \left(f_0'(X) \frac{1}{h_d} \int K_d \left(\frac{f_0(t) - f_0(X)}{h_d} \right) p_{1,K}(t) dt, \dots, f_0'(X) \frac{1}{h_d} \int K_d \left(\frac{f_0(t) - f_0(X)}{h_d} \right) p_{K,K}(t) dt \right)' \\ &= (p_{1,K}(X), p_{2,K}(X), \dots, p_{K,K}(X))', \end{aligned}$$

$$\hat{Q} = P'P/n,$$

$$\hat{\Sigma} = \sum_{i=1}^n P^K(X_i) P^K(X_i)' \left[y_i - \mathbf{M}_{\text{mon}}^1(X_i, \hat{f}) \right]^2 / n.$$

$\square$

Homogeneity

Suppose that the underlying function f_0 is homogeneous of degree r . Based on our construction, there is

$$\mathbf{M}_{\text{hom}}^r(X, \hat{f}) - f_0(X) = \frac{\sum_{j=1}^J s_j^r \left(\hat{f}(s_j X) - f_0(s_j X) \right)}{\sum_{j=1}^J (s_j^r)^2}.$$

This is just a weighted sum of some random variables which are asymptotic normal. Thus, we define γ such that for any θ

$$\gamma(\theta) = \frac{\sum_{j=1}^J s_j^r \theta(s_j X)}{\sum_{j=1}^J (s_j^r)^2}.$$

Let $P^K(x) = (P_{1,K}(x), P_{2,K}(x), \dots, P_{K,K}(x))'$ and $\mathbf{P} = [P^K(x_1), P^K(x_2), \dots, P^K(x_n)]'$. Based on Theorem 5 from Newey (1997), we have

$$\sqrt{n} \hat{V}^{-1/2} \left(\mathbf{M}_{\text{hom}}^r(X, \hat{f}) - f_0(X) \right) \xrightarrow{d} \mathcal{N}(0, 1),$$

where

$$\hat{V} = \hat{A}' \hat{Q}^{-1} \hat{\Sigma} \hat{Q}^{-1} \hat{A}.$$

Specifically,

$$\begin{aligned} \hat{A} &= \frac{\sum_{j=1}^J s_j^r P^K(s_j X)}{\sum_{j=1}^J (s_j^r)^2}, \quad \hat{Q} = P' P / n, \\ \hat{\Sigma} &= \sum_{i=1}^n P^K(X_i) P^K(X_i)' \left[y_i - \mathbf{M}_{\text{hom}}^r(X_i, \hat{f}) \right]^2 / n. \end{aligned}$$

□

Combination of Different Shape Restrictions

Suppose that the underlying function f_0 is homogeneous of degree r and is increasing. WLOG, we only consider the case in which monotonicity is imposed at first. Based on the construction of our method, we obtain $\mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{mon}}^1(X; \hat{f})$ as the final estimation for the input X . Aim to show that $\mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{mon}}^1(X; \hat{f})$ is asymptotic normal.

Firstly, we write $\mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{mon}}^1(X; \hat{f}) - f_0$ as

$$\frac{\sum_{j=1}^J s_j^r \left(\mathbf{M}_{\text{mon}}^1(s_j X; \hat{f}) - f_0(s_j X) \right)}{\sum_{j=1}^J (s_j^r)^2}.$$

Since $\mathbf{M}_{\text{mon}}^1(s_j X; \hat{f}) - f_0(s_j X)$ can be written as the following

$$\frac{\partial}{\partial \mathbf{x}} f_0(s_j X) \int \frac{1}{h_d} K_d \left(\frac{f_0(s_j t) - f_0(s_j X)}{h_d} \right) \left(\hat{f}(s_j t) - f_0(s_j t) \right) ds_j t + \Delta_n,$$

there is

$$\mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{mon}}^1(X; \hat{f}) - f_0 = \gamma'_f[\hat{f} - f_0] + O_p(J\Delta_n).$$

where

$$\gamma'_f[g] = \sum_{j=1}^J \frac{(s_j^r)^2 \frac{\partial}{\partial \mathbf{x}} f_0(X)}{\sum_{j=1}^J (s_j^r)^2} \int \frac{1}{h_d} K_d \left(\frac{f_0(s_j t) - f_0(s_j X)}{h_d} \right) g(s_j t) dt.$$

For the error term, it suffices to ignore it if we set $\frac{J\sqrt{nh_d^2}}{K}, \frac{Jh_d}{K\sqrt{n}}, \frac{JK}{\sqrt{n}} \rightarrow 0$. Similar, we have

$$\sqrt{n}\hat{V}^{-1/2} \left(\mathbf{M}_{\text{hom}}^r \circ \mathbf{M}_{\text{mon}}^1(X; \hat{f}) - f_0 \right) \xrightarrow{d} \mathcal{N}(0, 1).$$

In this case, only the $\hat{A}$ is different. It is

$$\hat{A} = \left(\sum_{j=1}^J \frac{(s_j^r)^2 P_{1,K}(s_j X)}{\sum_{j=1}^J (s_j^r)^2}, \dots, \sum_{j=1}^J \frac{(s_j^r)^2 P_{K,K}(s_j X)}{\sum_{j=1}^J (s_j^r)^2} \right)'.$$

For the reverse case, $\mathbf{M}_{\text{mon}}^1 \circ \mathbf{M}_{\text{hom}}^r(X; \hat{f})$, the homogeneity-constrained estimator can be represented as a combination of several local polynomial estimators. Consequently, for a general well-trained neural network, imposing the homogeneity constraint prior to the monotonicity/convexity constraint also preserves the corresponding large-sample properties. $\square$

Proof of Lemma 2.2.1

Since there are $\bar{K}$ weights between the last hidden layer and the output, we can write $\mathcal{T}(X)$ as $\sum_{k=1}^{\bar{K}} v_k H_L^{(k)}(X)$, where v_k is the weight connecting the k -th nodes to the final output and $H_L^{(k)}(X)$ is the result at the k -th nodes in the last hidden layer.

We know that the degree of $\mathcal{T}$ is G , then each $H_L^{(k)}(X)$ can be written as $p^{\bar{K}}(X)' w_L^{(k)}$ for a vector $w_L^{(k)}$. Then we have

$$\begin{aligned}\mathcal{T}(X) &= \sum_{k=1}^{\bar{K}} p^{\bar{K}}(X)' w_L^{(k)} v_k = p^{\bar{K}}(X)' \sum_{k=1}^{\bar{K}} w_L^{(k)} v_k \\ &= p^{\bar{K}}(X)' \left(\sum_{k=1}^{\bar{K}} w_L^{(k)} v_k \right) \\ &= p^{\bar{K}}(X)' \left(\left(w_L^{(1)}, w_L^{(2)}, \dots, w_L^{(\bar{K})} \right)' V^{\bar{K}} \right).\end{aligned}$$

Letting $\mathbf{W} = \left(w_L^{(1)}, w_L^{(2)}, \dots, w_L^{(\bar{K})} \right)'$ provides the required result. $\square$

Proof of Lemma 2.2.2

Based on Lemma 2.2.1, we know that $\mathcal{T}(X) = p^{\bar{K}}(X)' \mathbf{W} V^{\bar{K}}$. Training network $\mathcal{T}$ is to minimize the distance

$$\min_{V^{\bar{K}}} \sum_i \text{dist} \left(y_i, p^{\bar{K}}(X_i)' \mathbf{W} V^{\bar{K}} \right).$$

On the other hand, the power series estimation is to minimize

$$\min_{\beta} \sum_i \text{dist} \left(y_i, p^{\bar{K}}(X_i)' \beta \right).$$

Since $\mathbf{W}$ is invertible, the network estimator is equivalent to a power series estimator in terms of the point estimation. Suppose the loss function is the mean squared error. The minimization becomes

$$\min_{V^{\bar{K}}} \sum_{i=1}^n \left(y_i - p^{\bar{K}}(X_i)' \mathbf{W} V^{\bar{K}} \right)^2.$$

We can obtain the estimator of $V^{\bar{K}}$, denoted by $\hat{V}$, as

$$\hat{V} = (\mathbf{W}' P' P \mathbf{W})^{-1} \mathbf{W}' P' Y$$

Therefore, there is

$$\begin{aligned}
\hat{\mathcal{T}}(X) &= p^{\bar{K}}(X)' \mathbf{W} \hat{V} \\
&= p^{\bar{K}}(X)' \left(\mathbf{W} (\mathbf{W}' P' P \mathbf{W})^{-1} \mathbf{W}' P' Y \right) \\
&= p^{\bar{K}}(X)' \left(\mathbf{W} \mathbf{W}^{-1} (P' P)^{-1} (\mathbf{W}')^{-1} \mathbf{W}' P' Y \right) \\
&= p^{\bar{K}}(X)' \left((P' P)^{-1} P' Y \right) \\
&= p^{\bar{K}}(X)' \hat{\beta}
\end{aligned}$$

where $p^{\bar{K}}(X)' \hat{\beta}$ is the point estimation based the power series estimator. $\square$

B.2 Supplementary Text

For lemma 2.2.2, we provide a simple simulation for the distribution of the difference between the point estimates and the truth at point $(0.5, 0.5)$ based on the network in figure B.1. The corresponding result is based on 1000 sample size and 1500 simulations. In this case, two partially frozen networks with different fixed matrices $\mathbf{W}$ and one network with all trainable weights are compared. We also drew a normal distribution with the sample mean and variance of the results from one of the frozen networks. Figure B.2 shows that these two frozen networks produce the similar distribution of the point estimates and this distribution is almost normal, which is consistent with our theorem. The network with all trainable weights seem to provide the similar distribution; however, we have no theory that its asymptotic distribution is normal. This could be a potential research topic.

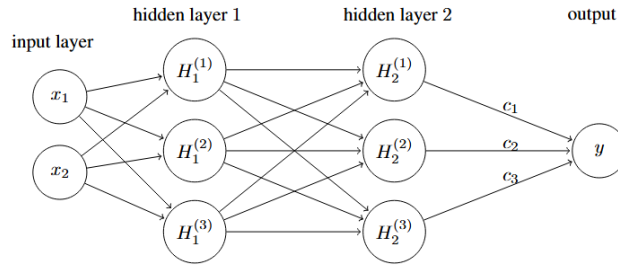


Figure B.1: 2-hidden-layer network.

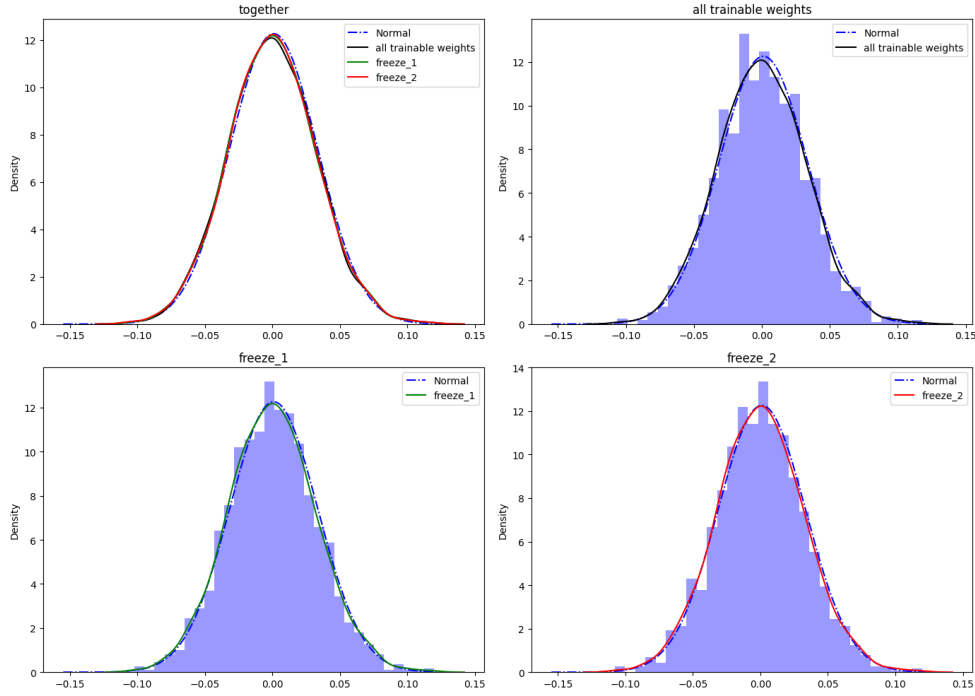


Figure B.2: **Distributions of point estimates.**

The histograms illustrate the distribution of point estimates from two partially frozen networks and one network with all trainable weights, respectively. The colored solid lines are the corresponding kernel density estimation curves. The dotted line is the Gaussian probability density function curve.

Appendix C

Proofs and Supplementary Text/Figure in Chapter 3

C.1 Proof of Proposition 3.4.1 and 3.4.2

Proof: Note that by Definition 3.4.3 and the stopping condition in algorithm 1, if the ϵ is 0, the returned policy profile $(\sigma_{M1}, \sigma_{M2}, \sigma_{N1}, \sigma_{N2})$ is the team equilibrium without communication (Definition 3.4.1). We are left to show that the algorithm eventually terminates. Algorithm 1 terminates eventually because it either adds a new policy for at least one player's policy population in each iteration or terminates. For a game with finite number of policies, when no new policy are added for every players, there is no incentive for players to deviate from the returned Nash equilibrium against the opponent, so the condition in Definition 3.4.1 is satisfied and the algorithm 1 terminates in team equilibrium without communication. The proof of Proposition 3.4.2 proceeds in an analogous manner and is omitted to avoid redundancy. $\square$

C.2 Supplementary Text/Figure

C.2.1 Team Kuhn Poker

Consider a deck consisting of five distinct cards endowed with a strict total order. At the beginning of each round, one card is dealt uniformly at random to each of the four players without replacement, leaving one card undealt. Each player privately observes only

their own card and receives no information regarding the cards held by any other player, including their teammate. Communication between teammates is prohibited throughout the game.

The game is played sequentially according to the fixed action order: **M1**, **N1**, **M2**, and **N2**. Each player acts exactly once. The available actions depend on the current betting history. If no prior **bet** has been made, the acting player may either **check** or **bet**. Once a **bet** has occurred, all subsequent players may only **call** or **fold**. At most one **bet** may be initiated during the game. At the start of game, every player contributes an ante of one unit to the pot. A player who chooses to bet or call contributes one additional unit. Players who fold relinquish all claims to the pot and take no further actions in the round. After all actions have been completed, a showdown takes place among the players who have not folded. The player holding the highest-ranked card among the remaining active players wins the showdown. The entire pot is then awarded to the winning player's team.

Utilities are defined at the team level. The total pot is divided equally between the two members of the winning team. Each player's individual utility is given by their share of the awarded pot minus their own monetary contribution (i.e., ante plus any additional wager). The team utility is defined as the sum of the utilities of its two members, thereby inducing a two-team zero-sum game.

C.2.2 Examples of Learning Dynamics

Figure C.1: Learning dynamics of classical PSRO and proposed algorithm with random initial policy population 1

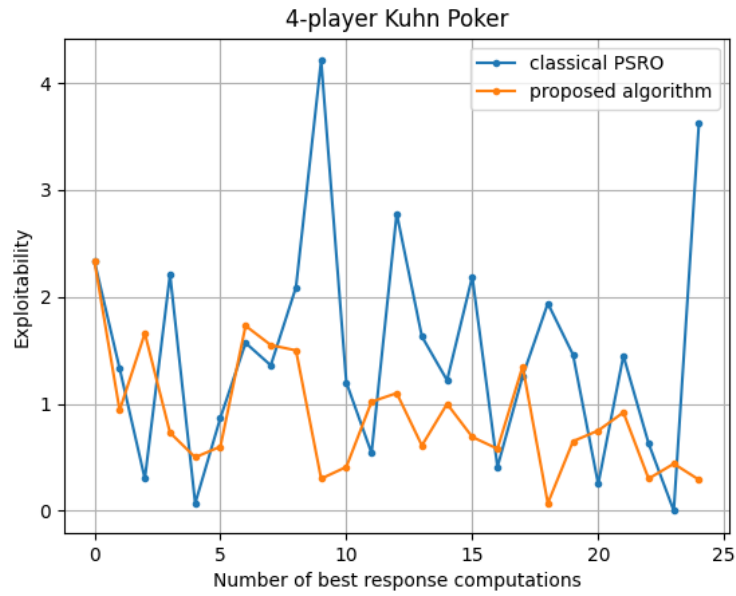


Figure C.2: Learning dynamics of classical PSRO and proposed algorithm with random initial policy population 2

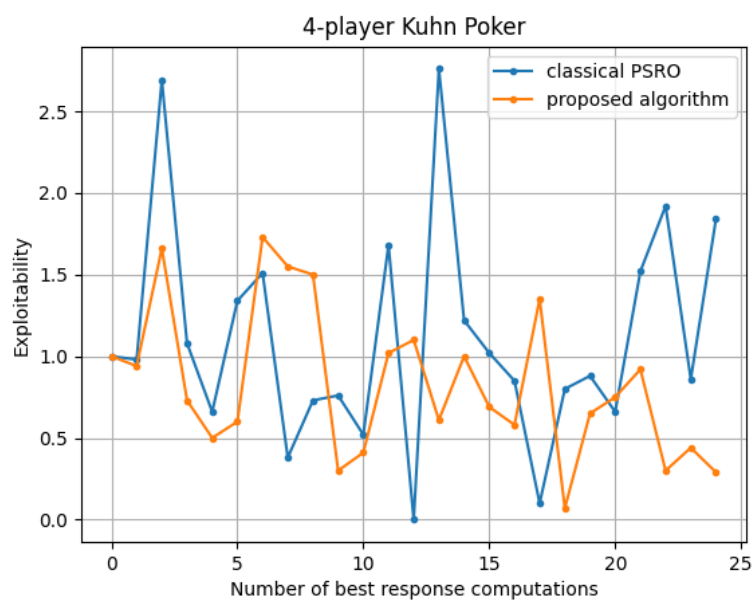


Figure C.3: Learning dynamics of classical PSRO and proposed algorithm with random initial policy population 3

