A Convex Optimization Approach for Computing Correlated Choice Probabilities with Many Alternatives*

Selin Damla Ahipasaoglu† Xiaobo Li‡ Karthik Natarajan§

May 25, 2016

Abstract

A popular discrete choice model that incorporates correlation information is the Multinomial Probit (MNP) model where the random utilities of the alternatives are chosen from a multivariate normal distribution. Computing the choice probabilities is challenging in the MNP model when the number of alternatives is large and simulation is a popular technique used to approximate the choice probabilities. Mishra, Natarajan and Teo (2012) have recently proposed a semidefinite optimization approach to compute choice probabilities for the distribution of the random utilities that maximizes expected agent utility given only the mean, variance and covariance information. Their model is referred to as the Cross Moment (CMM) model. Computing the choice probabilities with many alternatives is challenging in the CMM model since one needs to solve large scale semidefinite programs. We develop a simpler formulation as a representative agent model by maximizing over the choice probabilities in the unit simplex where the objective function is the sum of the expected utilities and a strongly concave perturbation function. By characterizing the perturbation function for the CMM model and its gradient, we develop a simple first order gradient method with inexact line search to compute choice probabilities. We establish local linear convergence of this algorithm under mild assumptions on the choice probabilities. An implication of our results is that inverting the choice probabilities to compute

*This project was partly funded by the SUTD-MIT International Design Center grant IDG31300105 on 'Optimization for Complex Discrete Choice' and the MOE Tier 2 grant MOE2013-T2-2-168 on 'Distributional Robust Optimization for Consumer Choice in Transportation Systems'.

†Engineering Systems and Design, Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372. Email: ahipasaoglu@sutd.edu.sg

‡Industrial and Systems Engineering, University of Minnesota, Minneapolis, MN 55455. Email: lixx3195@umn.edu

§Engineering Systems and Design, Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372. Email: karthik_natarajan@sutd.edu.sg

the mean utilities is straightforward given any positive definite covariance matrix. Numerical experiments show that this method can compute choice probabilities for a large number of alternatives within a reasonable amount of time while explicitly capturing the correlation information. Comparisons with simulation methods for MNP and semidefinite programming methods for CMM indicate the efficacy of the method.

1 Introduction

Consider a population of agents, each of whom chooses their most preferred alternative from a finite but mutually exclusive set of alternatives $[n] = \{1, 2, ..., n\}$. Our goal in this paper is to compute for each alternative, the probability of being chosen by agents in this population. Random utility maximization is a popular approach used to study such a discrete choice problem which we discuss next.

Notation: Throughout the paper, we use ordinary characters such as x, μ *to denote scalars, boldfaced lowercase characters such as* $\boldsymbol{x}, \boldsymbol{\mu}$ *to denote vectors, boldfaced uppercase characters such as* $\boldsymbol{X}, \boldsymbol{\Sigma}$ *to denote matrices and characters with the tilde notation such as* $\tilde{u}, \tilde{\epsilon}$ *to denote random variables. We use* $\boldsymbol{x}^T$ *and* $\boldsymbol{X}^T$ *to denotes the transpose of a vector* $\boldsymbol{x}$ *and a matrix* $\boldsymbol{X}$*.*

1.1 Random Utility Maximization

In the additive random utility maximization model, the utility of alternative i is specified as:

$$\tilde{u}_i = \mu_i + \tilde{\epsilon}_i, \quad \forall i \in [n], \tag{1}$$

where μ_i is the deterministic component of the utility and $\tilde{\epsilon}_i$ is the random component of the utility of alternative i. Let $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^T$ denote the vector of the deterministic component of the utility which is common across the agents and $\tilde{\boldsymbol{\epsilon}} = (\tilde{\epsilon}_1, \dots, \tilde{\epsilon}_n)^T$ denote the vector of the random components with a joint probability distribution $\theta(\cdot)$. Each realization of the random component vector results in a utility vector for an agent in the population who chooses the alternative with the highest utility. Assuming no ties, the probability that i is the most preferred alternative in this population of agents is given by:

$$p_i = P\left(i = \underset{k \in [n]}{\operatorname{argmax}} \ (\mu_k + \tilde{\epsilon}_k)\right). \tag{2}$$

Let $\boldsymbol{p} = (p_1, ..., p_n)^T$ denote the vector of choice probabilities. In the Multinomial Logit (MNL) model, the random terms $\tilde{\epsilon}_i$ are assumed to be identically and independently distributed Gumbel random variables. The MNL model was popularized by Luce [28] and McFadden [29] and has the following choice probability formula:

$$p_i^{\text{mnl}} = \frac{e^{\mu_i}}{\sum\limits_{k \in [n]} e^{\mu_k}}, \quad \forall i \in [n].$$

While the MNL choice probability is known in closed form and possesses desirable properties such as concavity of the log-likelihood function, it also suffers from drawbacks. One of the well-known properties of MNL is the Independence of Irrelevant Alternatives (IIA) property which implies that the ratio of the choice probabilities for any two alternatives is independent of the utilities of the other alternatives:

$$\frac{p_i^{\text{mnl}}}{p_j^{\text{mnl}}} = e^{\mu_i - \mu_j}, \quad \forall i \neq j.$$

When the alternatives have correlated utilities, the IIA property of MNL gives rise to misleading choice predictions. Correlation information can be captured using the Generalized Extreme Value (GEV) model (see McFadden [30]), which is a generalization of MNL model. While the choice probabilities in the GEV model still have a closed form expression, the model does not allow for all possible correlation structures among the random terms.

A model that accounts for any valid correlation matrix is the Multinomial Probit (MNP) model in which $\tilde{\boldsymbol{\epsilon}}$ is assumed to be normally distributed with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}$, namely $\tilde{\boldsymbol{u}} \sim \text{Normal}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The MNP model is flexible in terms of modeling dependence and does not possess the IIA property. However the choice probabilities do not have a closed form expression with Monte Carlo simulation being the most commonly used method to find the choice probabilities. The reader is referred to Hajivassiliou, McFadden and Ruud [18] for an in-depth discussion of simulation techniques used to approximate the choice probabilities in MNP models with the Geweke-Hajivassiliou-Keane (GHK) simulator being the most commonly used technique among them (see Geweke [14], Hajivassiliou and McFadden [17], Keane [25]) .

1.2 A Representative Agent Model

An alternative derivation of the MNL choice probability that is particularly relevant to this paper is the "representative agent" model from economics (see Anderson, Palma and Thisse [2, 1]). In this model, the aggregate behavior of a population of agents is described through the choices made by a single representative agent who has a preference for diversity and randomizes choice. Consider a representative agent who chooses a probability vector in the unit $(n-1)$-simplex:

$$\mathbf{\Delta}_{n-1} = \left\{ \boldsymbol{x} \in \Re_n^+ \;\middle|\; \boldsymbol{e}^T \boldsymbol{x} = 1 \right\},$$

where $\boldsymbol{e}$ is a vector of all ones. Then, it is easy to verify that the MNL choice probabilities are the optimal decision variables to an entropy maximization problem of the form:

$$\boldsymbol{p}^{\text{mnl}} = \operatorname{argmax} \left\{ \boldsymbol{\mu}^T \boldsymbol{x} - \sum_{i \in [n]} \left(x_i \ln x_i - x_i \right) \;\middle|\; \boldsymbol{x} \in \mathbf{\Delta}_{n-1} \right\}. \tag{3}$$

The optimization problem (3) is solved by a representative agent who decides on the probability vector $\boldsymbol{x}$ to get an expected utility of $\boldsymbol{\mu}^T \boldsymbol{x}$ plus a entropy based perturbation function $\sum_i \left(-x_i \ln x_i + x_i \right)$ that rewards the agent for randomizing.

Recently, Fudenberg, Iijima and Strzalecki [13] have proposed the use of a general additive perturbation function in the representative agent model inspired from the entropy formulation for the MNL model as a simple and tractable approach to model choice under uncertainty. Under this approach, the choice probability vector is defined as the optimal solution to the problem:

$$\boldsymbol{p} = \operatorname{argmax} \left\{ \boldsymbol{\mu}^T \boldsymbol{x} - \sum_{i \in [n]} V_i(x_i) \;\middle|\; \boldsymbol{x} \in \mathbf{\Delta}_{n-1} \right\}, \tag{4}$$

where the functions $V_i(x_i)$ are assumed to be strictly convex in the interval $[0, 1]$, continuously differentiable in $(0, 1)$ with $\lim_{x_i \to 0} V_i'(x_i) = -\infty$. The entropy maximization problem is a special case of this model. Under the assumptions on the functions $V_i(\cdot)$, the choice probabilities are unique and lie in the relative interior of the simplex (see Rockafellar [37]). Furthermore, they provide an axiomatic justification of formulation (4) by showing its equivalence to two conditions, one condition generalizing an acyclicity condition derived from the strong axiom of revealed preferences and the second condition generalizing Luce's IIA condition. A weaker form of the perturbation function by relaxing the condition that $\lim_{x_i \to 0} V_i'(x_i) = -\infty$ is also studied in [13], which allows for choice probabilities to take a value of 0. The class of representative agent models with additive perturbation functions in (4) is in general not equivalent to the class of random utility models. Namely, such models rule out some random utility models even with iid. random terms and might allow for choices that do not admit a random utility model representation (see [13] for examples). Natarajan, Song and Teo [33] have provided an alternate justification for formulation (4) by relaxing the standard assumption that the joint distribution of the random components in the utility model is known. Let $\tilde{\boldsymbol{\epsilon}} \sim_\theta (F_1, \ldots, F_n)$ denote the set of probability distributions θ

for $\tilde{\boldsymbol{\epsilon}}$ with a fixed marginal distribution $F_j(\cdot)$ for each random term $\tilde{\epsilon}_j$. The model which they refer to as the Marginal Distribution (MDM) model evaluates the maximum expected agent utility over all joint distributions of the random components with the given marginal distributions and is formulated as follows:

$$\text{(MDM)} \quad Z^*_{\text{mdm}} = \max_{\tilde{\boldsymbol{\epsilon}} \sim_\theta (F_1,\ldots,F_n)} \mathbb{E}_\theta \left(\max_{i \in [n]} (\mu_i + \tilde{\epsilon}_i) \right). \tag{5}$$

They show that the maximum expected agent utility Z^*_{mdm} in the MDM model is the optimal objective value to the following concave maximization problem over the unit simplex:

$$\text{(MDM)} \quad Z^*_{\text{mdm}} = \max \left\{ \boldsymbol{\mu}^T \boldsymbol{x} + \sum_{i \in [n]} \int_{1-x_i}^{1} F_i^{-1}(t)dt \;\middle|\; \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \right\}, \tag{6}$$

with the choice probabilities given by the optimal decision variables. Clearly (6) is a special case of (4).

Hofbauer and Sandholm [21] have developed an extension of the representative agent formulation to allow for more general distributions for the random components as follows. Let $\boldsymbol{p} = P(\boldsymbol{\mu}) : \Re_n \to \boldsymbol{\Delta}_{n-1}$ be the mapping defined in (2) from the mean utility vector to the choice probability vector where the random vector $\tilde{\boldsymbol{\epsilon}}$ is assumed to admit a strictly positive density on $\Re_n$ and is sufficiently smooth such that the function $P(\cdot)$ is continuously differentiable. Then they show that there exists an deterministic function $V(\cdot) : \boldsymbol{\Delta}_{n-1} \to \Re$ such that:

$$\boldsymbol{p} = \text{argmax} \left\{ \boldsymbol{\mu}^T \boldsymbol{x} - V(\boldsymbol{x}) \;\middle|\; \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \right\}, \tag{7}$$

where the function $V(\cdot)$ is a continuously differentiable, strictly convex function in the simplex and becomes infinitely steep near the relative boundary of the simplex. Under these assumptions, the choice probability is unique and lies in the relative interior of the simplex. Feng, Li and Wang [12] have recently shown that the representative agent model in (7) is equivalent to a "semi-parametric" choice model, where the choice probability is evaluated for a distribution (or a sequence of distributions) in a prescribed set of distributions Θ that maximize expected utility:

$$Z^* = \max_{\tilde{\boldsymbol{\epsilon}} \sim_\theta \Theta} \mathbb{E}_\theta \left(\max_{i \in [n]} (\mu_i + \tilde{\epsilon}_i) \right). \tag{8}$$

In typical applications, choice probabilities are either known in closed form (examples include logit, nested logit and GEV models) or estimated using simulation (examples include probit and mixed logit models). Formulations (4) and (7) are interesting in their generality since the choice probability vectors can be computed as the solution to a convex optimization problem and thus opens up an alternate computational approach to find choice probabilities. Specific choices of perturbation functions that have been studied include the quadratic function $V(\boldsymbol{x}) = \sum_i x_i^2$, Tsallis entropy $V(\boldsymbol{x}) = \frac{1}{q(1-q)}\sum_i x_i - x_i^q$ for $q > 0$, Renyi entropy $V(\boldsymbol{x}) = -\frac{1}{1-q}\ln\sum_i x_i^q$ for $q \in (0,1)$ and the logarithm function $V(\boldsymbol{x}) = -\sum_i \ln x_i$ (see Mertikopoulos and Sandholm [31]). For the logarithm perturbation function, however there is no random utility model which yields the same choice probabilities as the representative agent model (see Hofbauer and Sandholm [21]). However, it seems difficult to find an explicit computationally tractable representation for the perturbation function $V(\cdot)$ in terms of the distribution function $\theta(\cdot)$, particularly with correlated utilities. To the best of our knowledge, random utility maximization models for which the perturbation function $V(\cdot)$ is explicitly known are the multinomial logit model, the nested logit model (see Verboven [44]) and the GEV model (see Swait and Marley [40]). These correspond to instances where the choice probabilities are known in closed form. For example in the MNP model with a general correlation structure among the random terms, no explicit representation of the function $V(\cdot)$ seems to be known.

The main contributions and the structure of the paper are summarized next:

(a) In Section 2, we review the Cross Moment (CMM) model introduced by Mishra, Natarajan and Teo [32]. In this choice model, the distribution of the random utilities is assumed to have mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, but the assumption of normality from the MNP model is dropped. The distribution in the CMM model is chosen as the one that maximizes the expected agent utility. Our main result in this section is to provide a representative agent formulation for the CMM model with an explicit computationally tractable representation of the perturbation function $V(\cdot)$. We also prove strong concavity of the objective function for the CMM model under the assumption that the covariance matrix is strictly positive definite. Our results make extensive use of properties of the square root function of matrices from matrix analysis.

(b) In Section 3, we evaluate the gradient of the objective function in the representative agent formulation for the CMM model and provide the optimality conditions. As an application, we show that under the assumption of strict positive definiteness of the covariance matrix, there is a one-to-one mapping between the mean utilities and the choice probabilities (or market shares) in the CMM model. Inferring the mean utilities from the choice probabilities (or inverting the market shares) is easy in the CMM model without having to go through simulation methods.

(c) In Section 4, we develop a simple first order gradient ascent method with inexact line search to compute choice probabilities. This is particularly appealing since it transforms the solution of a semidefinite program (SDP) for the CMM model to a gradient based method which makes it possible to solve large instances. We prove the algorithm is locally linearly convergent.

(d) In Section 5, we provide computational results for the CMM model. The computational results indicate that the decrease in the running time obtained from the gradient method is potentially of significant value when solving for choice probabilities with many alternatives since solving large scale semidefinite programs remains a challenge. While the choice probabilities in the CMM model are computed using convex optimization, the choice probabilities in the MNP model are computed using simulation. We provide numerical experiments to illustrate the efficacy of the model.

2 A Representative Agent Formulation for the Cross Moment (CMM) Model

We begin this section by reviewing the CMM model from Mishra, Natarajan and Teo [32].

2.1 Cross Moment Model

In this choice model, the joint distribution of $\tilde{\boldsymbol{\epsilon}}$ is assumed to be only partially specified to the modeler. Specifically, the available information on the joint distribution is the first two moments of $\tilde{\boldsymbol{\epsilon}}$. Let $\tilde{\boldsymbol{\epsilon}} \sim_\theta (\mathbf{0}, \boldsymbol{\Sigma})$ denote the set of probability distributions for $\tilde{\boldsymbol{\epsilon}}$ that satisfies the following two conditions: $E_\theta[\tilde{\boldsymbol{\epsilon}}] = \mathbf{0}$ and $Cov_\theta[\tilde{\boldsymbol{\epsilon}}] = \boldsymbol{\Sigma}$. The modeler is then assumed to solve the optimization problem:

$$\text{(CMM)} \quad Z^*_{\text{cmm}} = \max_{\tilde{\boldsymbol{\epsilon}} \sim_\theta (\mathbf{0}, \boldsymbol{\Sigma})} \mathbb{E}_\theta \left(\max_{i \in [n]} (\mu_i + \tilde{\epsilon}_i) \right). \tag{9}$$

The outer optimization in (9) is over all joint distributions of the random components that are consistent with the two moment information. Hence, problem (9) is equivalent to finding a joint distribution for the random components that maximizes the expected agent utility[1].

[1] The problem of finding the joint distribution of the random components that minimizes the expected agent utility with the first two moment information reduces to Jensen's bound. This is uninteresting from a discrete choice modelling perspective since all the agents then choose the alternative with the highest mean.

Mishra, Natarajan and Teo [32] solved the moment problem (9) by reformulating it as the following semidefinite program:

$$
\begin{aligned}
\text{(CMM)} \quad Z^*_{\text{cmm}} = \max \quad & \sum_{i\in[n]} \boldsymbol{e_i}^T \boldsymbol{y_i} \\
\text{s.t.} \quad & \sum_{i\in[n]} \begin{pmatrix} \boldsymbol{W_i} & \boldsymbol{y_i} \\ \boldsymbol{y_i}^T & x_i \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{\mu} \\ \boldsymbol{\mu}^T & 1 \end{pmatrix} \\
& \begin{pmatrix} \boldsymbol{W_i} & \boldsymbol{y_i} \\ \boldsymbol{y_i}^T & x_i \end{pmatrix} \succeq 0 \qquad \forall i \in [n],
\end{aligned}
\tag{10}
$$

where $\boldsymbol{e_i}$ is a vector with 1 in the ith position and 0 otherwise. Let $\{\boldsymbol{W_i^*}, \boldsymbol{y_i^*}, x_i^*\}$ for $i \in [n]$ be an optimal solution to the semidefinite program (10). The joint distribution of the random utilities $\tilde{\boldsymbol{u}}$ that maximizes the expected agent utility is a mixture of multivariate normal distributions given as:

$$
\tilde{\boldsymbol{u}} = \left\{ \text{Normal}\left(\frac{\boldsymbol{y_i^*}}{x_i^*}, \frac{\boldsymbol{W_i^*}}{x_i^*} - \frac{\boldsymbol{y_i^*}}{x_i^*}\frac{\boldsymbol{y_i^{*T}}}{x_i^*} \right), \quad \text{with probability } x_i^*, \quad \forall i \in [n]. \right. \tag{11}
$$

More importantly, they showed that the optimal decision variables $\boldsymbol{x}^*$ in the SDP formulation are the choice probabilities for the mixture of multivariate normal distributions in (11) which maximizes the expected agent utility. Mishra, Natarajan and Teo [32] provide applications of this formulation to problems in route choice, random walk theory and product line selection with the number of alternatives up to hundred. Numerical experiments in [32] showed that the CMM model captures correlation information in predicting choices and provides insights often qualitatively similar to MNP. The semidefinite programs have been further reduced in size by Natarajan and Teo [34] as follows:

$$
\begin{aligned}
\text{(CMM)} \quad Z^*_{\text{cmm}} = \max \quad & \text{trace}(\boldsymbol{Y}) \\
\text{s.t.} \quad & \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \\
& \begin{pmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{Y}^T & \boldsymbol{\mu} \\ \boldsymbol{Y} & \text{Diag}(\boldsymbol{x}) & \boldsymbol{x} \\ \boldsymbol{\mu}^T & \boldsymbol{x}^T & 1 \end{pmatrix} \succeq 0,
\end{aligned}
\tag{12}
$$

where trace($\boldsymbol{Y}$) is the trace of the matrix $\boldsymbol{Y}$ and Diag($\boldsymbol{x}$) is a diagonal matrix with the entries of $\boldsymbol{x}$ along the diagonal. For completeness, we provide the proof of the equivalence of the semidefinite programs (10) and (12) in the Appendix. In formulation (12), the optimal x_i^* variables represent a lower bound on the choice probability for the distribution θ^* that maximizes the expected agent utility given the first two moments:

$$P_{\theta^*}\left(i = \underset{k\in[n]}{\text{argmax}}\ (\mu_k + \tilde{\epsilon}_k)\right) \geq x_i^*, \quad \forall i \in [n],$$

where equality holds if there are no ties. Since the multivariate normal distribution is a feasible distribution in the CMM formulation, Z^*_{cmm} is an upper bound on the expected agent utility in MNP. Computationally these models differ in the way the choice probabilities are computed. In the MNP model, simulation techniques are used to compute the choice probabilities. On the other hand, the CMM model uses convex optimization techniques to solve the semidefinite program.

There has been an increasing interest in the literature on discrete choice models that deal with a large number of alternatives. Examples that have been studied includes the choice of lake recreation sites in the state of Wisconsin involving 589 alternatives (see Parsons and Kealy [36]), choice of car models involving 689 alternatives (see Brownstone, Bunch and Train [8]) and choice of messenger bags involving 3584 alternatives (see Toubia et al. [42]). Models that treat products as bundles of characteristics with an additive error term that accounts for variation in the taste for the products in conjunction with variation in taste for the characteristics of the products results in choices where the number of products (alternatives) is exponential in the number of characteristics. In a recent paper, Ahipasaoglu et. al. [38] used the CMM model as an alternative to MNP for computing choice probabilities in a traffic equilibrium problem and showed that it provides a practical alternative to MNP in estimating traffic flows. The correlation information in their model arises from origin-destination paths (alternatives) sharing common roads (characteristics). The number of paths in such networks might be exponential in the number of roads. In our computational experiments, we have found that solving the semidefinite program (12) using state of art interior point method based solvers such as SDPT3 version 4 (see [43, 41]) in MATLAB R2014 on a laptop with an Intel(R) i7-5600U CPU processor (2.6 GHz) with 4GB RAM works well when the number of alternatives is up to two hundred roughly. Solving large semidefinite programs with matrix size up to a few thousands still remains a computational challenge and is a subject of intense research in the optimization community. One such algorithm that is able to solve large scale SDPs is the research software SDPNAL+ version 0.3 that has been recently developed by Sun, Toh, Yang and Zhao (see [47] and [46])[2]. In contrast to such general purpose codes that solve large scale SDPs,

[2]This code has kindly been made freely available at http://www.math.nus.edu.sg/∼ mattohkc/SDPNALplus.html by the authors and is based on a semismooth Newton-Conjugate gradient augmented Lagrangian method coupled with a alternating direction method of multipliers.

we develop a specialized method based on gradient ascent with inexact line search by deriving a representative agent reformulation of the CMM model. Numerical results in Section 5 show that such a method is suitable when the number of alternatives is large.

2.2 Optimization over the Unit Simplex

In this section, we develop a representative agent formulation for the CMM model that transforms the semidefinite program to a nonlinear maximization problem over the unit simplex.

Theorem 1. *Assume that* $\boldsymbol{\Sigma} \succ 0$. *Then the maximum expected agent utility* Z^*_{cmm} *in the CMM model is the optimal objective value to the following nonlinear optimization problem over the unit simplex:*

$$(CMM) \quad Z^*_{cmm} = \max \left\{ \boldsymbol{\mu}^T \boldsymbol{x} + trace\left(\left(\boldsymbol{\Sigma}^{1/2}\ \boldsymbol{S}(\boldsymbol{x})\ \boldsymbol{\Sigma}^{1/2} \right)^{1/2} \right) \ \Big|\ \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \right\}, \tag{13}$$

where $\boldsymbol{S}(\boldsymbol{x}) = Diag(\boldsymbol{x}) - \boldsymbol{x}\boldsymbol{x}^T \succeq 0$ *and* $\boldsymbol{B} = \boldsymbol{A}^{1/2}$ *is the unique positive semidefinite square root of a matrix* $\boldsymbol{A} \succeq 0$ *such that* $\boldsymbol{A} = \boldsymbol{B}^2$. *Furthermore the optimal decision variables* $\boldsymbol{x}^*$ *are the choice probabilities for the distribution that maximizes the expected agent utility.*

Proof:

Applying Schur's lemma to the positive semidefinite matrix in formulation (12), we obtain the equivalent nonlinear semidefinite program:

$$\begin{aligned} Z^*_{\text{cmm}} = \max \quad & \text{trace}(\boldsymbol{Y}) \\ \text{s.t.} \quad & \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \\ & \begin{pmatrix} \boldsymbol{\Sigma} & \boldsymbol{Y}^T - \boldsymbol{\mu}\boldsymbol{x}^T \\ \boldsymbol{Y} - \boldsymbol{x}\boldsymbol{\mu}^T & \text{Diag}(\boldsymbol{x}) - \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \succeq 0. \end{aligned} \tag{14}$$

Define a transformation of the variables by letting $\hat{\boldsymbol{Y}} = \boldsymbol{Y} - \boldsymbol{x}\boldsymbol{\mu}^T$. Then, $\text{trace}(\hat{\boldsymbol{Y}}) = \text{trace}(\boldsymbol{Y}) - \boldsymbol{\mu}^T\boldsymbol{x}$. This transforms the problem to the equivalent nonlinear semidefinite programming formulation:

$$\begin{aligned} Z^*_{\text{cmm}} = \max \quad & \boldsymbol{\mu}^T\boldsymbol{x} + \text{trace}(\hat{\boldsymbol{Y}}) \\ \text{s.t.} \quad & \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \\ & \begin{pmatrix} \boldsymbol{\Sigma} & \hat{\boldsymbol{Y}}^T \\ \hat{\boldsymbol{Y}} & \boldsymbol{S}(\boldsymbol{x}) \end{pmatrix} \succeq 0, \end{aligned} \tag{15}$$

where $\boldsymbol{S}(\boldsymbol{x}) = \text{Diag}(\boldsymbol{x}) - \boldsymbol{x}\boldsymbol{x}^T$. The matrix $\boldsymbol{S}(\boldsymbol{x})$ is positive semidefinite for all $\boldsymbol{x} \in \boldsymbol{\Delta}_{n-1}$ since:

$$\begin{aligned} \boldsymbol{v}^T\boldsymbol{S}(\boldsymbol{x})\boldsymbol{v} &= \sum_{i\in[n]} v_i^2 x_i - \left(\sum_{i\in[n]} v_i x_i\right)^2, \\ &\geq 0, \qquad\qquad \forall \boldsymbol{v} \in \Re_n, \end{aligned}$$

where the last inequality comes from $\mathbb{E}(\tilde{v}^2) \geq \mathbb{E}(\tilde{v})^2$ where the random variable $\tilde{v}$ is defined to take value v_i with probability x_i for $i \in [n]$. The semidefinite program in (15) can be reformulated as a two-stage optimization problem of the form:

$$Z^*_{\text{cmm}} = \max\left\{\boldsymbol{\mu}^T\boldsymbol{x} - V(\boldsymbol{x}) \;\middle|\; \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1}\right\}, \tag{16}$$

where $\boldsymbol{x} \in \boldsymbol{\Delta}_{n-1}$ is the first stage decision vector and $V(\boldsymbol{x})$ is the optimal value to the following second stage problem where $\hat{\boldsymbol{Y}}$ is the second stage matrix decision variable:

$$\begin{aligned} V(\boldsymbol{x}) = \min \quad & -\text{trace}(\hat{\boldsymbol{Y}}) \\ \text{s.t.} \quad & \begin{pmatrix} \boldsymbol{\Sigma} & \hat{\boldsymbol{Y}}^T \\ \hat{\boldsymbol{Y}} & \boldsymbol{S}(\boldsymbol{x}) \end{pmatrix} \succeq 0. \end{aligned} \tag{17}$$

The second stage semidefinite program in (17) for a given value of $\boldsymbol{x}$ has a closed form solution (see Dowson and Landau [10], Olkin and Pukelsheim [35] and Shapiro [39]). Applying this result since $range(\boldsymbol{S}(\boldsymbol{x})) \subseteq range(\boldsymbol{\Sigma})$, the optimal second stage solution is given as:

$$\hat{\boldsymbol{Y}}^{*T} = \boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\left(\left(\boldsymbol{S}(\boldsymbol{x})^{1/2}\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right)^{1/2}\right)^{\dagger}\boldsymbol{S}(\boldsymbol{x})^{1/2}, \tag{18}$$

where $(\cdot)^\dagger$ denotes the Moore-Penrose pseudoinverse of a matrix. Hence, the optimal value of formulation (17) is:

$$\begin{aligned} V(\boldsymbol{x}) = \; & -\text{trace}\left(\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\left(\left(\boldsymbol{S}(\boldsymbol{x})^{1/2}\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right)^{1/2}\right)^{\dagger}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right), \\ = \; & -\text{trace}\left(\left(\boldsymbol{S}(\boldsymbol{x})^{1/2}\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right)\left(\left(\boldsymbol{S}(\boldsymbol{x})^{1/2}\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right)^{1/2}\right)^{\dagger}\right), \\ = \; & -\text{trace}\left(\left(\boldsymbol{S}(\boldsymbol{x})^{1/2}\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right)^{1/2}\right), \\ = \; & -\text{trace}\left(\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2}\right)^{1/2}\right), \end{aligned} \tag{19}$$

where the second equality comes from the invariance of the trace under cyclic permutations, the third equality comes from the property of the pseudo-inverse that $\boldsymbol{A}(\boldsymbol{A}^{1/2})^\dagger = \boldsymbol{A}^{1/2}\boldsymbol{A}^{1/2}(\boldsymbol{A}^{1/2})^\dagger =$

$\boldsymbol{A}^{1/2}$ and the last equality comes from the observation that for any $n \times n$ real square matrix $\boldsymbol{A}$, the matrices $\boldsymbol{A}\boldsymbol{A}^T$ and $\boldsymbol{A}^T\boldsymbol{A}$ have the same set of eigenvalues (see Horn and Johnson [22]). By substituting into (16), we obtain:

$$Z^*_{\text{cmm}} = \max\left\{ \boldsymbol{\mu}^T\boldsymbol{x} + \text{trace}\left(\left(\boldsymbol{\Sigma}^{1/2}\ \boldsymbol{S}(\boldsymbol{x})\ \boldsymbol{\Sigma}^{1/2} \right)^{1/2} \right) \ \middle|\ \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \right\}.$$

■

Remark 1. *The second stage problem in (17) has been studied in [10, 35, 39] in the following context: Given two n-dimensional random vectors with covariance matrices $\boldsymbol{\Sigma}$ and $\boldsymbol{S}(\boldsymbol{x})$, find the cross moment matrix $\hat{\boldsymbol{Y}}^T$ between the two random vectors that minimizes the expected L_2 distance between the vectors. In the proof of Theorem 1, the two vectors in the second stage corresponds to the random component of the utility vector $\tilde{\boldsymbol{\epsilon}}$ and the random choice vector which chooses $\boldsymbol{e}_i$ (alternative i) with probability x_i. The first stage problem corresponds to finding the best probability vector $\boldsymbol{x}$. For completeness, we provide a proof of the optimality of the solution in (18) in the Appendix.*

2.3 Strong Concavity of the Objective Function

In this section, we prove strong concavity of the objective function in the representative agent formulation for the CMM model. The result is based on the following definitions of functions of (positive semidefinite) matrices. Consider a symmetric positive semidefinite matrix $\boldsymbol{A}$ with an eigendecomposition $\boldsymbol{Q}\text{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^T$ where $\boldsymbol{Q}$ is an orthonormal matrix and $\boldsymbol{\lambda}$ is the vector of nonnegative eigenvalues. Given a function $h(\cdot) : [0, \infty) \to [0, \infty)$, the matrix function is defined as $h(\boldsymbol{A}) = \boldsymbol{Q}\text{Diag}(h(\boldsymbol{\lambda}))\boldsymbol{Q}^T$ where the function $h(\cdot)$ is applied to the eigenvalues in the diagonal matrix. As is the convention, we use $\boldsymbol{A} \succeq \boldsymbol{B}$ to denote $\boldsymbol{A} - \boldsymbol{B} \succeq 0$.

Definition 1. *Consider a function $h : [0, \infty) \to [0, \infty)$.*

(a) The function h is operator monotone if for all $\boldsymbol{A}, \boldsymbol{B} \succeq 0$:

$$\boldsymbol{A} \succeq \boldsymbol{B} \Longrightarrow h(\boldsymbol{A}) \succeq h(\boldsymbol{B}).$$

(b) The function h is operator concave if for all $\boldsymbol{A}, \boldsymbol{B} \succeq 0$ and $\lambda \in [0, 1]$:

$$h\left((1-\lambda)\boldsymbol{A} + \lambda\boldsymbol{B}\right) \succeq (1-\lambda)h(\boldsymbol{A}) + \lambda h(\boldsymbol{B}).$$

An example of a matrix function that is both operator monotone and operator concave is the square root function.

Theorem 2. *(Special case of the Löwner-Heinz Theorem [27, 19])*
The function $h(t) = t^{1/2}$ is both operator monotone and operator concave.

Before we introduce the key result of this section, we recall the definition of *strong convexity.*

Definition 2. *A function $V(\boldsymbol{x}) : \mathcal{D} \to \Re$ where $\mathcal{D}$ is a convex subset of $\Re_n$ is strongly convex if for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{D}$ and $\lambda \in (0,1)$, there exists a constant $m > 0$ such that*

$$V(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda V(\boldsymbol{x}) + (1-\lambda)V(\boldsymbol{y}) - \frac{m}{2}\lambda(1-\lambda)\|\boldsymbol{x}-\boldsymbol{y}\|^2.$$

This bring us to the following result for the objective function in formulation (13).

Theorem 3. *The function $V(\boldsymbol{x}) = -trace\left(\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2}\right)^{1/2}\right)$ defined on the unit simplex $\boldsymbol{x} \in \boldsymbol{\Delta}_{n-1}$ is strongly convex for $\boldsymbol{\Sigma} \succ 0$.*

Proof: For all $\boldsymbol{x}, \boldsymbol{y} \in \boldsymbol{\Delta}_{n-1}$ with $\boldsymbol{x} \neq \boldsymbol{y}$ and $\lambda \in (0,1)$, we have:

$$\begin{aligned}
\boldsymbol{S}(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}) &= \text{Diag}\left(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}\right) - \left(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}\right)\left(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}\right)^T, \\
&= \lambda\text{Diag}\left(\boldsymbol{x}\right) + (1-\lambda)\text{Diag}\left(\boldsymbol{y}\right) - \lambda^2\boldsymbol{x}\boldsymbol{x}^T - (1-\lambda)^2\boldsymbol{y}\boldsymbol{y}^T - \lambda(1-\lambda)(\boldsymbol{x}\boldsymbol{y}^T + \boldsymbol{y}\boldsymbol{x}^T), \\
&= \lambda\left(\text{Diag}\left(\boldsymbol{x}\right) - \boldsymbol{x}\boldsymbol{x}^T\right) + (1-\lambda)\left(\text{Diag}\left(\boldsymbol{y}\right) - \boldsymbol{y}\boldsymbol{y}^T\right) + \lambda(1-\lambda)(\boldsymbol{x}-\boldsymbol{y})(\boldsymbol{x}-\boldsymbol{y})^T, \\
&= \lambda\boldsymbol{S}(\boldsymbol{x}) + (1-\lambda)\boldsymbol{S}(\boldsymbol{y}) + \lambda(1-\lambda)(\boldsymbol{x}-\boldsymbol{y})(\boldsymbol{x}-\boldsymbol{y})^T.
\end{aligned}$$

Pre-multiplying and post-multiplying by $\boldsymbol{\Sigma}^{1/2}$ implies that:

$$\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y})\boldsymbol{\Sigma}^{1/2} = \boldsymbol{\Sigma}^{1/2}\left(\lambda\boldsymbol{S}(\boldsymbol{x}) + (1-\lambda)\boldsymbol{S}(\boldsymbol{y}) + \lambda(1-\lambda)(\boldsymbol{x}-\boldsymbol{y})(\boldsymbol{x}-\boldsymbol{y})^T\right)\boldsymbol{\Sigma}^{1/2}. \quad (20)$$

Now let $\boldsymbol{A} = \boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\lambda\boldsymbol{x}+(1-\lambda)\boldsymbol{y})\boldsymbol{\Sigma}^{1/2}$, $\boldsymbol{B} = \lambda\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2} + (1-\lambda)\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{y})\boldsymbol{\Sigma}^{1/2}$, $\rho = \lambda(1-\lambda)$ and $\boldsymbol{w} = \boldsymbol{\Sigma}^{1/2}(\boldsymbol{x}-\boldsymbol{y})$. Using this notation, we can rewrite the equation (20) as:

$$\boldsymbol{A} = \boldsymbol{B} + \rho\boldsymbol{w}\boldsymbol{w}^T.$$

Let $\lambda_1(\boldsymbol{A}) \leq \lambda_2(\boldsymbol{A}) \cdots \leq \lambda_n(\boldsymbol{A})$ denote the eigenvalues of $\boldsymbol{A}$ (and respectively $\lambda_i(\boldsymbol{B})$ for $\boldsymbol{B}$). For $\rho > 0$ with a rank one perturbation, Bunch, Nilsen and Sorensen [9] have shown that:

$$\lambda_i(\boldsymbol{A}) \geq \lambda_i(\boldsymbol{B}), \quad \forall i \in [n].$$

Let $a = \rho \boldsymbol{w}^T \boldsymbol{w}$. Then there exists a vector $\boldsymbol{\beta} \geq 0$ with $\sum_i \beta_i = a$ such that

$$\lambda_i(\boldsymbol{A}) = \lambda_i(\boldsymbol{B}) + \beta_i, \quad \forall i \in [n].$$

Hence a lower bound on the sum of the square roots of the eigenvalues of the matrix $\boldsymbol{A}$ is obtained by solving the optimization problem:

$$\begin{aligned}
\sum_{i\in[n]} \lambda_i(\boldsymbol{A})^{1/2} \geq \quad \min_{\boldsymbol{\beta}} \quad & \sum_{i\in[n]} (\lambda_i(\boldsymbol{B}) + \beta_i)^{1/2} \\
\text{s.t.} \quad & \sum_{i\in[n]} \beta_i = a, \\
& \beta_i \geq 0, \qquad \forall i \in [n].
\end{aligned}$$

The right hand side of the above inequality corresponds to minimizing a concave function over a simplex, therefore the minimizer must be attained by at least one of the vertices of the simplex. This gives:

$$\begin{aligned}
\sum_{i\in[n]} \lambda_i(\boldsymbol{A})^{1/2} \quad & \geq \quad \min_{j\in[n]} \left\{ \sum_{i\neq j} \lambda_i(\boldsymbol{B})^{1/2} + (\lambda_j(\boldsymbol{B}) + a)^{1/2} \right\}, \\
& \geq \quad \min_{j\in[n]} \left\{ \sum_{i\in[n]} \lambda_i(\boldsymbol{B})^{1/2} + \frac{a}{2\sqrt{\lambda_j(\boldsymbol{B}) + a}} \right\}, \\
& = \quad \sum_{i\in[n]} \lambda_i(\boldsymbol{B})^{1/2} + \frac{a}{2\sqrt{\lambda_n(\boldsymbol{B}) + a}},
\end{aligned}$$

where the second inequality is from the supergradient inequality for the concave square root function. Clearly there exists a positive number M_1, such that $a = \lambda(1-\lambda)(\boldsymbol{x}-\boldsymbol{y})^T \boldsymbol{\Sigma}(\boldsymbol{x}-\boldsymbol{y}) \leq M_1$ for all $\lambda \in (0,1)$ and $\boldsymbol{x}, \boldsymbol{y} \in \boldsymbol{\Delta}$. Similarly, there exists a positive number M_2, such that $\lambda_n(\boldsymbol{B}) = \max_i \lambda_i(\boldsymbol{B}) \leq M_2$, for all $\lambda \in (0,1)$ and $\boldsymbol{x}, \boldsymbol{y} \in \boldsymbol{\Delta}$. Letting $\alpha = \frac{1}{\sqrt{M_1+M_2}}$, we have

$$\sum_{i\in[n]} \lambda_i(\boldsymbol{A})^{1/2} \geq \sum_{i\in[n]} \lambda_i(\boldsymbol{B})^{1/2} + \frac{\alpha}{2}\lambda(1-\lambda)(\boldsymbol{x}-\boldsymbol{y})^T \boldsymbol{\Sigma}(\boldsymbol{x}-\boldsymbol{y}). \tag{21}$$

Since $\mathrm{trace}(\boldsymbol{A}^{1/2}) = \sum_{i\in[n]} \lambda_i(\boldsymbol{A})^{1/2}$, by (21) we obtain

$$\begin{aligned}
& \mathrm{trace}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y})\boldsymbol{\Sigma}^{1/2}\right)^{1/2} \\
\geq \quad & \mathrm{trace}\left(\lambda\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2} + (1-\lambda)\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{y})\boldsymbol{\Sigma}^{1/2}\right)^{1/2} + \frac{\alpha}{2}\lambda(1-\lambda)(\boldsymbol{x}-\boldsymbol{y})^T\boldsymbol{\Sigma}(\boldsymbol{x}-\boldsymbol{y}), \\
\geq \quad & \lambda\mathrm{trace}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2}\right)^{1/2} + (1-\lambda)\mathrm{trace}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{y})\boldsymbol{\Sigma}^{1/2}\right)^{1/2} + \frac{\alpha}{2}\lambda(1-\lambda)(\boldsymbol{x}-\boldsymbol{y})^T\boldsymbol{\Sigma}(\boldsymbol{x}-\boldsymbol{y}),
\end{aligned}$$

where the last inequality is from the concavity of the square root function. Let $\lambda_{min}(\boldsymbol{\Sigma})$ be the smallest eigenvalue of $\boldsymbol{\Sigma}$. Since $\boldsymbol{\Sigma}$ is positive definite, then $\lambda_{min}(\boldsymbol{\Sigma}) > 0$ and

$$\|\boldsymbol{\Sigma}^{1/2}(\boldsymbol{x} - \boldsymbol{y})\|^2 \geq \lambda_{min}(\boldsymbol{\Sigma})\|\boldsymbol{x} - \boldsymbol{y}\|^2.$$

Then by the definition of the function $V(\cdot)$, we obtain

$$V(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda V(\boldsymbol{x}) + (1-\lambda)V(\boldsymbol{y}) - \frac{\alpha}{2}\lambda_{min}(\boldsymbol{\Sigma})\lambda(1-\lambda)\|\boldsymbol{x} - \boldsymbol{y}\|^2,$$

and therefore the function $V(\boldsymbol{x})$ is strongly convex on its domain for $\boldsymbol{\Sigma} \succ 0$, where the strong convexity parameter depends on the matrix $\boldsymbol{\Sigma}$. ■

3 Optimality Conditions and Its Implications

One of the key advantages in developing the representative agent formulation for the CMM model is that it transforms the semidefinite program to a nonlinear strongly concave maximization problem over the unit simplex. In this section, we provide a characterization of the directional derivatives of the objective function and provide optimality conditions for the model. In addition, we show that as we approach the boundary of the feasible region from its interior, the (projected) gradient of the objective function blows to infinity. These results have important implications that we discuss in detail in this section.

3.1 Projected Gradient of the Objective Function

Consider a vector $\boldsymbol{x}$ in the relative interior of the simplex and restrict the direction of the perturbation to be in the tangent space of $\boldsymbol{\Delta}_{n-1}$ defined as $\overline{\boldsymbol{\Delta}}_{n-1} = \left\{\boldsymbol{v} \in \Re_n \;\middle|\; \boldsymbol{e}^T\boldsymbol{v} = 0\right\}$. Let $||\boldsymbol{v}||_2 = 1$. Then the directional derivative of $V(\boldsymbol{x})$ in the direction $\boldsymbol{v}$ is defined as:

$$\boldsymbol{\nabla}_{\boldsymbol{v}} V(\boldsymbol{x}) = \lim_{\epsilon \to 0} \frac{V(\boldsymbol{x} + \epsilon \boldsymbol{v}) - V(\boldsymbol{x})}{\epsilon}.$$

To compute the directional derivative, observe that:

$$\begin{aligned} V(\boldsymbol{x} + \epsilon\boldsymbol{v}) &= -\text{trace}\left(\boldsymbol{\Sigma}^{1/2}\ \boldsymbol{S}(\boldsymbol{x} + \epsilon\boldsymbol{v})\ \boldsymbol{\Sigma}^{1/2}\right)^{1/2}, \\ &= -\text{trace}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2} + \epsilon\boldsymbol{\Sigma}^{1/2}\left(\text{Diag}(\boldsymbol{v}) - \boldsymbol{x}\boldsymbol{v}^T - \boldsymbol{v}\boldsymbol{x}^T\right)\boldsymbol{\Sigma}^{1/2} - \epsilon^2\boldsymbol{\Sigma}^{1/2}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\right)^{1/2}, \\ &= -\text{trace}\left(\boldsymbol{T}(\boldsymbol{x}) + \boldsymbol{E}_{\boldsymbol{v}}(\epsilon, \boldsymbol{x})\right)^{1/2}, \end{aligned}$$

where:

$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2} \text{ and } \boldsymbol{E_v}(\epsilon, \boldsymbol{x}) = \epsilon\boldsymbol{\Sigma}^{1/2}\left(\mathrm{Diag}(\boldsymbol{v}) - \boldsymbol{x}\boldsymbol{v}^T - \boldsymbol{v}\boldsymbol{x}^T\right)\boldsymbol{\Sigma}^{1/2} - \epsilon^2\boldsymbol{\Sigma}^{1/2}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}.$$

The next lemma provides a characterization of the null space of the matrix $\boldsymbol{T}(\boldsymbol{x})$ and its relation to the null space of the matrix $\boldsymbol{E_v}(\epsilon, \boldsymbol{x})$. This lemma is needed for the main result of this and the next section.

Lemma 1.

(a) *Let* $\boldsymbol{x} = \binom{\boldsymbol{0}}{\underline{\boldsymbol{x}}} \in \boldsymbol{\Delta}_{n-1}$, *where* $\boldsymbol{0}$ *is a vector of* r *zeros and* $\underline{\boldsymbol{x}} \in \Re^{++}_{n-r}$ *is a strictly positive vector for some integer* r *such that* $0 \le r \le n-1$. *Then the null space of the matrix* $\boldsymbol{T}(\boldsymbol{x})$ *is given as:*

$$\mathit{Null}(\boldsymbol{T}(\boldsymbol{x})) = \left\{ k\boldsymbol{\Sigma}^{-1/2}\boldsymbol{z} \;\middle|\; \boldsymbol{z} = \begin{pmatrix} \boldsymbol{z_1} \\ \boldsymbol{e} \end{pmatrix},\ \boldsymbol{z_1} \in \Re_r \right\},$$

where $\boldsymbol{e}$ *is a vector of ones of dimension* $n-r$ *and* $\mathit{rank}(\boldsymbol{T}(\boldsymbol{x})) = n-r-1$.

(b) *Particularly, when* $\boldsymbol{x}$ *is in the relative interior of the simplex denoted by* $\mathit{rint}(\boldsymbol{\Delta}_{n-1})$, *then* $\mathit{rank}(\boldsymbol{T}(\boldsymbol{x})) = n-1$ *and* $\mathit{Null}(\boldsymbol{T}(\boldsymbol{x})) \subseteq \mathit{Null}(\boldsymbol{E_v}(\epsilon, \boldsymbol{x}))$.

Proof:

(a) Let $\boldsymbol{A} \circ \boldsymbol{B}$ denote the Hadamard product of two matrices of the same dimension. Any vector $\boldsymbol{z} \in \Re_n$ can be expressed as $\boldsymbol{z} = \boldsymbol{\Sigma}^{-1/2}\binom{\boldsymbol{z_1}}{\boldsymbol{z_2}}$, for some $\boldsymbol{z_1} \in \Re_r$ and $\boldsymbol{z_2} \in \Re_{n-r}$. Then:

$$\begin{aligned}\boldsymbol{T}(\boldsymbol{x})\boldsymbol{z} &= \boldsymbol{\Sigma}^{1/2}\left(\mathrm{Diag}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2}\boldsymbol{z} - \boldsymbol{x}\boldsymbol{x}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{z}\right), \\ &= \boldsymbol{\Sigma}^{1/2}\boldsymbol{x} \circ \left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{z} - \boldsymbol{x}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{z}\boldsymbol{e}\right),\end{aligned}$$

where the last equality comes from the observation that $\mathrm{Diag}(\boldsymbol{x})(\boldsymbol{\Sigma}^{1/2}\boldsymbol{z}) = \boldsymbol{x} \circ (\boldsymbol{\Sigma}^{1/2}\boldsymbol{z})$ and $\boldsymbol{x}(\boldsymbol{x}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{z}) = \boldsymbol{x} \circ (\boldsymbol{x}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{z})\boldsymbol{e}$. This is equivalent to

$$\boldsymbol{T}(\boldsymbol{x})\boldsymbol{z} = \boldsymbol{\Sigma}^{1/2}\begin{pmatrix} \boldsymbol{0} \\ \underline{\boldsymbol{x}} \end{pmatrix} \circ \left(\begin{pmatrix} \boldsymbol{z_1} \\ \boldsymbol{z_2} \end{pmatrix} - \begin{pmatrix} \boldsymbol{0} \\ \underline{\boldsymbol{x}} \end{pmatrix}^T \begin{pmatrix} \boldsymbol{z_1} \\ \boldsymbol{z_2} \end{pmatrix}\boldsymbol{e}\right).$$

Since $\underline{\boldsymbol{x}} > \boldsymbol{0}$, $\boldsymbol{T}(\boldsymbol{x})\boldsymbol{z} = \boldsymbol{0}$ implies that:

$$\boldsymbol{z_2} - (\underline{\boldsymbol{x}}^T\boldsymbol{z_2})\boldsymbol{e} = \boldsymbol{0}.$$

Solving this equation gives $\boldsymbol{z} \in \mathrm{Null}(\boldsymbol{T}(\boldsymbol{x})) = \{k\boldsymbol{\Sigma}^{-1/2}\boldsymbol{z} \mid \boldsymbol{z} = \binom{\boldsymbol{z_1}}{\boldsymbol{e}}\}$ where $\boldsymbol{z_1} \in \Re_r$. Therefore, $\mathrm{rank}(\mathrm{Null}(\boldsymbol{T}(\boldsymbol{x}))) = r+1$ and the rank-nullity theorem implies that $\mathrm{rank}(\boldsymbol{T}(\boldsymbol{x})) = n-r-1$.

(b) For $\boldsymbol{x} \in \text{rint}(\boldsymbol{\Delta}_{n-1})$, all the entries are nonzero. To show that $\boldsymbol{\Sigma}^{-1/2}\boldsymbol{e}$ lies in the null space of the matrix $\boldsymbol{E_v}(\epsilon, \boldsymbol{x})$, observe that:

$$\begin{aligned} \boldsymbol{E_v}(\epsilon,\boldsymbol{x})\boldsymbol{\Sigma}^{-1/2}\boldsymbol{e} &= \epsilon\boldsymbol{\Sigma}^{1/2}\left(\text{Diag}(\boldsymbol{v}) - \boldsymbol{x}\boldsymbol{v}^T - \boldsymbol{v}\boldsymbol{x}^T\right)\boldsymbol{\Sigma}^{1/2}\boldsymbol{\Sigma}^{-1/2}\boldsymbol{e} - \epsilon^2\boldsymbol{\Sigma}^{1/2}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{\Sigma}^{-1/2}\boldsymbol{e}, \\ &= \epsilon\boldsymbol{\Sigma}^{1/2}\left(\text{Diag}(\boldsymbol{v})\boldsymbol{e} - \boldsymbol{x}\boldsymbol{v}^T\boldsymbol{e} - \boldsymbol{v}\boldsymbol{x}^T\boldsymbol{e}\right) - \epsilon^2\boldsymbol{\Sigma}^{1/2}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{e}, \\ &= \boldsymbol{0}, \end{aligned}$$

where the final equality comes from the observation that $\boldsymbol{e}^T\boldsymbol{x} = 1$ and $\boldsymbol{v}^T\boldsymbol{e} = 0$. Hence, $\text{Null}(\boldsymbol{T}(\boldsymbol{x})) \subseteq \boldsymbol{E_v}(\epsilon, \boldsymbol{x})$. ■

To prove the main theorem of this section, we make use of the Fréchet derivative of a matrix function which is defined as follows.

Definition 3. *The Fréchet derivative of a real matrix function $g : \Re_{n\times n} \mapsto \Re_{n\times n}$ at $\boldsymbol{X} \in \Re_{n\times n}$ is a linear mapping $L_g : \Re_{n\times n} \mapsto \Re_{n\times n}$ such that $g(\boldsymbol{X}+\boldsymbol{E}) - g(\boldsymbol{X}) - L_g(\boldsymbol{X},\boldsymbol{E}) = o(||\boldsymbol{E}||)$ for all $\boldsymbol{E} \in \Re_{n\times n}$.*

The Fréchet derivative, if exists, is known to be unique. The Fréchet derivative for the matrix square root function, which exists when $\boldsymbol{X}$ is positive definite, is the unique solution to the Sylvester equation (refer to Kenney and Laub [26], Higham [20]):

$$\boldsymbol{X}^{1/2}L_{1/2}(\boldsymbol{X},\boldsymbol{E}) + L_{1/2}(\boldsymbol{X},\boldsymbol{E})\boldsymbol{X}^{1/2} = \boldsymbol{E}. \tag{22}$$

Next, we derive the first order derivative of $V(\cdot)$.

Theorem 4. *Define:*

$$\boldsymbol{g}(\boldsymbol{x}) = -\tfrac{1}{2}\,diag\left(\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^\dagger\boldsymbol{\Sigma}^{1/2}\right) + \boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^\dagger\boldsymbol{\Sigma}^{1/2}\boldsymbol{x}, \tag{23}$$

where $diag(\cdot)$ is the column vector formed by the diagonal elements of the matrix and $\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2}$. The directional derivative of $V(\boldsymbol{x})$ at $\boldsymbol{x} \in rint(\boldsymbol{\Delta}_{n-1})$ in the direction $\boldsymbol{v} \in \overline{\boldsymbol{\Delta}}_{n-1}$ is $\boldsymbol{\nabla_v}V(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{x})^T\boldsymbol{v}$, and its projected gradient on the tangent space is:

$$\overline{\boldsymbol{\nabla}}V(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{x}) - \tfrac{1}{n}\boldsymbol{e}^T\boldsymbol{g}(\boldsymbol{x})\boldsymbol{e}. \tag{24}$$

Proof: Lemma 1 implies that for the given symmetric matrices $\boldsymbol{T}(\boldsymbol{x})$ and $\boldsymbol{E_v}(\epsilon,\boldsymbol{x})$, there exists an orthogonal matrix $\boldsymbol{P}$ with $\boldsymbol{P}\boldsymbol{P}^T = \boldsymbol{P}^T\boldsymbol{P} = \boldsymbol{I}$ such that

$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \overline{\boldsymbol{\Lambda}}(\boldsymbol{x}) \end{pmatrix}\boldsymbol{P}^T \text{ and } \boldsymbol{E_v}(\epsilon,\boldsymbol{x}) = \boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}) \end{pmatrix}\boldsymbol{P}^T,$$

where $\overline{\mathbf{\Lambda}}(\boldsymbol{x})$ is a diagonal matrix of size $(n-1)\times(n-1)$ containing the non-zero eigenvalues of $\boldsymbol{T}(\boldsymbol{x})$ and $\boldsymbol{P}$ is the matrix of eigenvectors of matrix $\boldsymbol{T}(\boldsymbol{x})$ with the first eigenvector equal to $\frac{\mathbf{\Sigma}^{-1/2}\boldsymbol{e}}{||\mathbf{\Sigma}^{-1/2}\boldsymbol{e}||_2}$. The matrix $\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})$ is however not necessarily diagonal. Thus, we obtain:

$$\begin{aligned}
V(\boldsymbol{x}+\epsilon\boldsymbol{v})-V(\boldsymbol{x}) &= -\text{trace}\left((\boldsymbol{T}(\boldsymbol{x})+\boldsymbol{E}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}))^{1/2}\right)+\text{trace}\left(\boldsymbol{T}(\boldsymbol{x})^{1/2}\right),\\
&= -\text{trace}\left(\boldsymbol{P}\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\mathbf{\Lambda}}(\boldsymbol{x})+\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\end{pmatrix}\boldsymbol{P}^T\right)^{1/2}+\text{trace}\left(\boldsymbol{P}\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\mathbf{\Lambda}}(\boldsymbol{x})\end{pmatrix}\boldsymbol{P}^T\right)^{1/2},\\
&= -\text{trace}\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\mathbf{\Lambda}}(\boldsymbol{x})+\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\end{pmatrix}^{1/2}+\text{trace}\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\mathbf{\Lambda}}(\boldsymbol{x})\end{pmatrix}^{1/2},\\
&= -\text{trace}\left(\left(\overline{\mathbf{\Lambda}}(\boldsymbol{x})+\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)^{1/2}-\overline{\mathbf{\Lambda}}^{1/2}(\boldsymbol{x})\right).
\end{aligned}$$

To evaluate the last expression, let $L_{1/2}(\overline{\mathbf{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}))$ denote the Fréchet derivative for the matrix square root which is the unique solution to the Sylvester equation:

$$\overline{\mathbf{\Lambda}}^{1/2}(\boldsymbol{x})L_{1/2}(\overline{\mathbf{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}))+L_{1/2}(\overline{\mathbf{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}))\overline{\mathbf{\Lambda}}^{1/2}(\boldsymbol{x})=\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}).$$

The existence of solution is guaranteed since $\overline{\mathbf{\Lambda}}(\boldsymbol{x})\succ 0$. The Sylvester equation can then be expressed as:

$$L_{1/2}(\overline{\mathbf{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}))+\overline{\mathbf{\Lambda}}^{-1/2}(\boldsymbol{x})L_{1/2}(\overline{\mathbf{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}))\overline{\mathbf{\Lambda}}(\boldsymbol{x})^{1/2}=\overline{\mathbf{\Lambda}}^{-1/2}(\boldsymbol{x})\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}).$$

Hence:

$$\text{trace}\left(L_{1/2}(\overline{\mathbf{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x}))\right)=\frac{1}{2}\text{trace}\left(\overline{\mathbf{\Lambda}}^{-1/2}(\boldsymbol{x})\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right).$$

Using the definition of the Fréchet derivative we have:

$$\begin{aligned}
V(\boldsymbol{x}+\epsilon\boldsymbol{v})-V(\boldsymbol{x}) &= -\frac{1}{2}\text{trace}\left(\overline{\mathbf{\Lambda}}^{-1/2}(\boldsymbol{x})\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)+o\left(||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||\right),\\
&= -\frac{1}{2}\text{trace}\left(\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\mathbf{\Lambda}}^{-1/2}(\boldsymbol{x})\end{pmatrix}\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\end{pmatrix}\right)+o\left(||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||\right),\\
&= -\frac{1}{2}\text{trace}\left(\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\mathbf{\Lambda}}^{-1/2}(\boldsymbol{x})\end{pmatrix}\boldsymbol{P}^T\boldsymbol{E}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\boldsymbol{P}\right)+o\left(||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||\right),\\
&= -\frac{1}{2}\text{trace}\left(\boldsymbol{P}\begin{pmatrix}0 & \mathbf{0}^T\\ \mathbf{0} & \overline{\mathbf{\Lambda}}^{-1/2}(\boldsymbol{x})\end{pmatrix}\boldsymbol{P}^T\boldsymbol{E}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)+o\left(||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||\right),\\
&= -\frac{\epsilon}{2}\text{trace}\left((\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\mathbf{\Sigma}^{1/2}\left(\text{Diag}(\boldsymbol{v})-\boldsymbol{x}\boldsymbol{v}^T-\boldsymbol{v}\boldsymbol{x}^T\right)\mathbf{\Sigma}^{1/2}\right)+o\left(||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||\right),\\
&= -\frac{\epsilon}{2}\left(\text{diag}\left(\mathbf{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\mathbf{\Sigma}^{1/2}\right)^T-2\boldsymbol{x}^T\mathbf{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\mathbf{\Sigma}^{1/2}\right)\boldsymbol{v}+o\left(||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||\right).
\end{aligned}$$

Hence, we obtain the expression for the directional derivative in the direction $\boldsymbol{v} \in \overline{\boldsymbol{\Delta}}_{n-1}$ as:

$$\begin{aligned}
\boldsymbol{\nabla}_{\boldsymbol{v}} V(\boldsymbol{x}) &= \lim_{\epsilon \to 0} \frac{V(\boldsymbol{x}+\epsilon \boldsymbol{v}) - V(\boldsymbol{x})}{\epsilon}, \\
&= -\frac{1}{2}\left(\operatorname{diag}\left(\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\right)^T - 2\boldsymbol{x}^T\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\right)\boldsymbol{v}, \\
&= \boldsymbol{g}(\boldsymbol{x})^T \boldsymbol{v},
\end{aligned}$$

where $\boldsymbol{g}(\boldsymbol{x}) = -\frac{1}{2}\left(\operatorname{diag}\left(\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\right) - 2\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\boldsymbol{x}\right)$. Since this is true for all $\boldsymbol{v} \in \overline{\boldsymbol{\Delta}}_{n-1}$, we obtain $\overline{\boldsymbol{\nabla}} V(\boldsymbol{x})$ by projecting $\boldsymbol{g}(\boldsymbol{x})$ onto the tangent space $\overline{\boldsymbol{\Delta}}_{n-1}$. [3] ■

3.2 Optimality Conditions

The representative agent formulation for the CMM model is:

$$Z^* = \max\left\{ f(\boldsymbol{x}) \;\middle|\; \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1} \right\} \text{ where } f(\boldsymbol{x}) = \boldsymbol{\mu}^T\boldsymbol{x} - V(\boldsymbol{x}).$$

Since the objective function is strongly concave and given the first-order derivatives of the objective function in Theorem 4, we can now write down the first-order optimality conditions for the CMM model as follows:

$$\overline{\boldsymbol{\nabla}} f(\boldsymbol{x}) = \left(\boldsymbol{\mu} - \frac{1}{n}\boldsymbol{e}^T\boldsymbol{\mu}\boldsymbol{e}\right) - \left(\boldsymbol{g}(\boldsymbol{x}) - \frac{1}{n}\boldsymbol{e}^T\boldsymbol{g}(\boldsymbol{x})\boldsymbol{e}\right) = 0 \quad \text{and} \quad \boldsymbol{x} \in \boldsymbol{\Delta}_{n-1}, \tag{25}$$

where $\overline{\boldsymbol{\nabla}} f(\boldsymbol{x})$ is the projected gradient of f onto the tangent space of the feasible region. Next, we will discuss some of the implications of these optimality conditions.

3.3 Mapping between Mean Utilities and Choice Probabilities

In this section, we show a one-to-one correspondence between the mean utilities $\boldsymbol{\mu}$ under appropriate normalization and the choice probabilities $\boldsymbol{x}$ in the relative interior of the simplex in the CMM model. This is important from an modeling viewpoint since it shows that the CMM model is capable of generating all the choice probabilities in the relative interior of the unit simplex. Furthermore, this is important in identification and estimation of demand parameters (see Berry [4]). We show

[3] We abuse the notation slightly here since the gradient of $V(\boldsymbol{x})$ does not exist outside the feasible region. For all theoretical and algorithmic purposes, the mathematical quantity that we calculate, i.e., $\overline{\boldsymbol{\nabla}} V(\boldsymbol{x})$, behaves as the projected gradient. To be mathematically precise, one would embed the function into the affine subspace $\boldsymbol{e}^T\boldsymbol{x} = 1$ by substituting for one of the decision variables, but this approach is notationally cumbersome in exposition.

that under mild assumptions on the covariance matrix, inverting the choice probabilities in the CMM model is fairly easy. Towards this, we first prove the following lemma that characterizes the gradient of the objective function in the representative agent formulation of the CMM model near the relative boundary of the simplex.

Theorem 5. *Assume that* $\boldsymbol{\Sigma} \succ 0$. *As* $\boldsymbol{x}$ *approaches the relative boundary of the unit simplex, the projected gradient of* $V(\cdot)$ *blows up to* $+\infty$.

Proof: Suppose that the sequence of interior points $\{\boldsymbol{x}_k\}_{k=1,\ldots\infty}$ approaches a point $\hat{\boldsymbol{x}}$ on the relative boundary of the unit simplex, along the direction $-\boldsymbol{z} \in \Re_n$. Assume that $\hat{\boldsymbol{x}}$ has exactly m zeros. Any such $\boldsymbol{z}$ must satisfy $\boldsymbol{e}^T\boldsymbol{z} = 0$ and $\hat{x}_i = 0 \Rightarrow z_i > 0$. We first prove that:

$$\lim_{t\to 0+} \frac{V(\hat{\boldsymbol{x}}) - V(\hat{\boldsymbol{x}} + t\boldsymbol{z})}{t} = +\infty.$$

Let $z_0 = \min_{i:\hat{x}_i=0} z_i$ and $\sigma_1 = \lambda_1(\boldsymbol{\Sigma})$. We have:

$$\begin{aligned} \boldsymbol{T}(\hat{\boldsymbol{x}} + t\boldsymbol{z}) &= \boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\hat{\boldsymbol{x}} + t\boldsymbol{z})\boldsymbol{\Sigma}^{1/2}, \\ &= \boldsymbol{\Sigma}^{1/2}\mathrm{Diag}(\hat{\boldsymbol{x}} + t\boldsymbol{z})\boldsymbol{\Sigma}^{1/2} - \boldsymbol{\Sigma}^{1/2}(\hat{\boldsymbol{x}} + t\boldsymbol{z})(\hat{\boldsymbol{x}} + t\boldsymbol{z})^T\boldsymbol{\Sigma}^{1/2}. \end{aligned}$$

It is clear that $\min_i\{\hat{x}_i + tz_i\} = \min_{i:\hat{x}_i=0} tz_i = tz_0$ if t is sufficiently small. From Lemmas 4 and 5, both given in the Appendix, this implies that

$$\lambda_2(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\hat{\boldsymbol{x}} + t\boldsymbol{z})\boldsymbol{\Sigma}^{1/2}) \geq \lambda_1(\boldsymbol{\Sigma}^{1/2}\mathrm{Diag}(\hat{\boldsymbol{x}} + t\boldsymbol{z})\boldsymbol{\Sigma}^{1/2}) \geq t\sigma_1 z_0 > 0$$

since $\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\hat{\boldsymbol{x}} + t\boldsymbol{z})\boldsymbol{\Sigma}^{1/2}$ is a rank one update of $\boldsymbol{\Sigma}^{1/2}\mathrm{Diag}(\hat{\boldsymbol{x}} + t\boldsymbol{z})\boldsymbol{\Sigma}^{1/2}$. Therefore, together with Lemma 1, we have:

$$\lambda_1(\boldsymbol{T}(\hat{\boldsymbol{x}} + t\boldsymbol{z})) = 0,\ \lambda_2(\boldsymbol{T}(\hat{\boldsymbol{x}} + t\boldsymbol{z})) \geq t\sigma_1 z_0,\ \text{and}\ \lambda_3(\boldsymbol{T}(\hat{\boldsymbol{x}} + t\boldsymbol{z})), \ldots, \lambda_n(\boldsymbol{T}(\hat{\boldsymbol{x}} + t\boldsymbol{z})) > 0.$$

Recall also that Lemma 1 gives:

$$\lambda_1(\boldsymbol{T}(\hat{\boldsymbol{x}})) = \cdots = \lambda_{m+1}(\boldsymbol{T}(\hat{\boldsymbol{x}})) = 0 \text{ and } \lambda_{m+2}(\boldsymbol{T}(\hat{\boldsymbol{x}})), \ldots, \lambda_n(\boldsymbol{T}(\hat{\boldsymbol{x}})) > 0.$$

Furthermore, from standard results for the continuity of the matrix eigenvalue function (see [16]), we know that $\|\lambda_j(\boldsymbol{T}(\hat{\boldsymbol{x}} + t\boldsymbol{z})) - \lambda_j(\boldsymbol{T}(\hat{\boldsymbol{x}}))\| \leq O(t)$, for all j. Combining above facts,

$$\begin{aligned} \lim_{t\to 0_+} \frac{V(\hat{\boldsymbol{x}}) - V(\hat{\boldsymbol{x}} + t\boldsymbol{z})}{t} &= \lim_{t\to 0_+} \frac{\displaystyle\sum_{j=1}^{n} \left(\sqrt{\lambda_j(\boldsymbol{T}(\hat{\boldsymbol{x}} + t\boldsymbol{z}))} - \sqrt{\lambda_j(\boldsymbol{T}(\hat{\boldsymbol{x}}))}\right)}{t} \\ &\geq \lim_{t\to 0_+} \frac{\sqrt{t\sigma_1 z_0} + \displaystyle\sum_{j=m+2}^{n} \left(\sqrt{\lambda_j(\boldsymbol{T}(\hat{\boldsymbol{x}})) + O(t)} - \sqrt{\lambda_j(\boldsymbol{T}(\hat{\boldsymbol{x}}))}\right)}{t} \\ &\to +\infty, \end{aligned}$$

since the ratio $\sqrt{t}/t$ diverges to $+\infty$ as t approaches 0 and $\lim_{t\to 0_+} O(t)/t = 0$.

We next show that

$$\lim_{t\to 0+} |\boldsymbol{\nabla}_{\boldsymbol{z}} V(\hat{\boldsymbol{x}} + t\boldsymbol{z})| = \lim_{t\to 0+} \lim_{s\to 0^+} \left(\frac{V(\hat{\boldsymbol{x}} + t\boldsymbol{z}) - V(\hat{\boldsymbol{x}} + (t+s)\boldsymbol{z})}{s} \right) = +\infty.$$

Suppose this is not the case. Since $V(\hat{\boldsymbol{x}})$ is convex, $\boldsymbol{\nabla}_{\boldsymbol{z}} V(\hat{\boldsymbol{x}} + t\boldsymbol{z})$ is monotone in t, which implies that

$$\liminf_{t\to 0+} \lim_{s\to 0^+} \left(\frac{V(\hat{\boldsymbol{x}} + t\boldsymbol{z}) - V(\hat{\boldsymbol{x}} + (t+s)\boldsymbol{z})}{s} \right) = \limsup_{t\to 0+} \lim_{s\to 0^+} \left(\frac{V(\hat{\boldsymbol{x}} + t\boldsymbol{z}) - V(\hat{\boldsymbol{x}} + (t+s)\boldsymbol{z})}{s} \right).$$

Therefore, there has to be M such that

$$\lim_{s\to 0^+} \left(\frac{V(\hat{\boldsymbol{x}} + t\boldsymbol{z}) - V(\hat{\boldsymbol{x}} + (t+s)\boldsymbol{z})}{s} \right) \leq M, \quad \forall t > 0 \text{ with } \hat{\boldsymbol{x}} + t\boldsymbol{z} \in \ rint(\Delta_{n-1}). \tag{26}$$

However, since $\lim_{t\to 0+} \frac{V(\hat{\boldsymbol{x}}) - V(\hat{\boldsymbol{x}} + t\boldsymbol{z})}{t} = +\infty$, there exists $t_0 > 0$ such that $\frac{V(\hat{\boldsymbol{x}}) - V(\hat{\boldsymbol{x}} + t_0\boldsymbol{z})}{t_0} > 2M$. In addition, since V is continuous on Δ_{n-1}, it is also uniformly continuous on Δ_{n-1}. Therefore, there exists $\delta > 0$ such that $|V(\hat{\boldsymbol{x}}) - V(\hat{\boldsymbol{x}} + s\boldsymbol{z})| < t_0 M$ for all $s \in [0, \delta)$. It is without loss of generality to assume that $\delta < t_0$. Therefore,

$$\begin{aligned} \lim_{s\to 0^+} \left(\frac{V(\hat{\boldsymbol{x}} + \frac{\delta}{2}\boldsymbol{z}) - V(\hat{\boldsymbol{x}} + (\frac{\delta}{2} + s)\boldsymbol{z})}{s} \right) &\geq \left(\frac{V(\hat{\boldsymbol{x}} + \frac{\delta}{2}\boldsymbol{z}) - V(\hat{\boldsymbol{x}} + t_0\boldsymbol{z})}{t_0 - \frac{\delta}{2}} \right), \\ &\geq \left(\frac{V(\hat{\boldsymbol{x}} + \frac{\delta}{2}\boldsymbol{z}) - V(\hat{\boldsymbol{x}} + t_0\boldsymbol{z})}{t_0} \right), \\ &> \left(\frac{V(\hat{\boldsymbol{x}}) - t_0 M - V(\hat{\boldsymbol{x}} + t_0\boldsymbol{z})}{t_0} \right) > M. \end{aligned}$$

This is in contradiction with equation (26), which completes the proof. ■

We are now ready to prove the main result of this section. Hofbauer and Sandholm [21] have shown that given any joint distribution of the noise terms, the mapping from the deterministic components of the utilities $\boldsymbol{\mu}$ (under appropriate normalization) to the set of choice probabilities in the relative interior of the simplex is surjective, namely any vector of choice probabilities can be obtained by selecting suitable mean values. We show in the next theorem that under mild assumptions on the covariance matrix, there is a one-to-one correspondence between the mean utilities under the normalization condition $\mu_1 = 0$ and the choice probabilities in the relative interior of the simplex for the CMM model.

Theorem 6. *Assume that* $\mathbf{\Sigma} \succ 0$. *Without loss of generality, set* $\mu_1 = 0$. *Let* $\boldsymbol{p} = P(\boldsymbol{\mu}) : \{0\} \times \Re_{n-1} \to \mathbf{\Delta}_{n-1}$ *be the mapping from the mean utilities to the choice probabilities in the CMM model. Then* $P(\cdot)$ *is a bijection between* $\{0\} \times \Re_{n-1}$ *and the relative interior of the simplex* Δ_{n-1}, *namely there is a one-to-one correspondence between the mean utilities and the choice probabilities.*

Proof:

(a) We first show that every mean vector in $\{0\} \times \Re_{n-1}$ in the CMM model results in a unique vector of choice probabilites in the relative interior of the unit simplex. From the strong concavity of the objective function in the representative agent formulation of the CMM model (Theorems 1 and 3) and the observation that the gradient of the objective function blows up to infinity near the relative boundary of the simplex (Theorem 5), the choice probability vector in the CMM model lies strictly in the relative interior of the simplex and is unique.

(b) We next show that every choice probability vector in the relative interior of the simplex maps to a unique mean vector in $\{0\} \times \Re_{n-1}$ in the CMM model. From the optimality conditions in (25) and with $\mu_1 = 0$, by multiplying with the vector $\boldsymbol{e}_1$ we have:

$$\frac{1}{n}\boldsymbol{e}^T\boldsymbol{\mu} = \frac{1}{n}\boldsymbol{e}^T\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}_1(\boldsymbol{x}).$$

Plugging in back to the optimality conditions, we obtain the mean utilities from the choice probabilities as follows:

$$\boldsymbol{\mu} = \boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}_1(\boldsymbol{x})\boldsymbol{e}. \tag{27}$$

Taken together, this implies there is a one-to-one correspondence between the set of deterministic utilities in $\{0\} \times \Re_{n-1}$ and the set of choice probabilities in the relative interior of the unit simplex. ■

For the MNL model, the mean utilities are uniquely identified from the following simple formula:

$$\mu_i = \ln(p_i^{\text{mnl}}) - \ln(p_1^{\text{mnl}}), \quad \forall i \in [n].$$

A similar result exists for identifying the mean utilities from the nested logit model (see Berry [4]). For the CMM model, the mean utilities are uniquely identified from the simple calculation in (27) where $\boldsymbol{g}(\boldsymbol{x}) = -\frac{1}{2}\left(\text{diag}\left(\mathbf{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\mathbf{\Sigma}^{1/2}\right) - 2\mathbf{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\mathbf{\Sigma}^{1/2}\boldsymbol{x}\right)$. To the best of our knowledge, no such easily computable formula is available for the MNP model.

4 Calculating the Choice Probabilities in the CMM model

4.1 The Gradient Ascent Algorithm

In this section we present a projected gradient ascent method [4] to calculate the choice probabilities in the CMM model. The algorithm is given in Algorithm 1 in which stepsizes are chosen according to the well-known Armijo's line search rule.

Input: $\boldsymbol{\mu}, \boldsymbol{\Sigma}$, starting point $\boldsymbol{x_0}$, initial step size $\alpha_0 \in (0, 1]$, $\beta \in (0, 1)$, $\tau \in (0, 1)$, tolerance $\epsilon > 0$.

Output: Optimal solution $\boldsymbol{x}^*$.

Initialize stopping criteria: *criteria* $\leftarrow \epsilon + 1$;

while *criteria* $> \epsilon$ **do**

  $\alpha \leftarrow \alpha_0$

  $\boldsymbol{x} \leftarrow \boldsymbol{x}_0 + \alpha \overline{\boldsymbol{\nabla}} f(\boldsymbol{x}_0)$,

  while $\boldsymbol{x} \notin \text{int}(\boldsymbol{\Delta}_{n-1})$ *or* $f(\boldsymbol{x}) < f(\boldsymbol{x}_0) + \tau\alpha \|\overline{\boldsymbol{\nabla}} f(\boldsymbol{x}_0)\|^2$ **do**

    $\alpha \leftarrow \beta\alpha$

    $\boldsymbol{x} \leftarrow \boldsymbol{x}_0 + \alpha \overline{\boldsymbol{\nabla}} f(\boldsymbol{x}_0)$,

  end

  $\boldsymbol{x}_0 \leftarrow \boldsymbol{x}$

  criteria $\leftarrow \|\boldsymbol{x} - \boldsymbol{x_0}\|$.

end

Algorithm 1: Projected gradient ascent algorithm with Armijo search

From Theorem 5, we know that the optimal solution lies in $\text{int}(\boldsymbol{\Delta}_{n-1})$. The algorithm presented in Algorithm 1 converges to the optimal solution (see [23]). While the objective function has a nice curvature (it is strongly concave), it does not have a Lipschitz continuous gradient near the relative boundary. In fact, the function itself does not satisfy the Lipschitz continuity condition near the relative boundary (see Theorem 5). In the next section, we show that if $\boldsymbol{x}$ is sufficiently far away from the relative boundary of the feasible region, then the algorithm converges linearly for appropriately chosen parameters within a local neighborhood. As we show numerically in Section 5, this helps explains the good behavior of the algorithm in most cases while for some ill-conditioned

[4] Note that the presentation here is slightly different than the one in classical references such as Section 2.3 of Bertsekas [5], but the algorithm is the same since the projection is onto an affine subspace, i.e., $\text{Proj}_{\boldsymbol{Ax}=\boldsymbol{b}}(\hat{\boldsymbol{x}} + \nabla f) = \hat{\boldsymbol{x}} + \text{Proj}_{\boldsymbol{Ax}=\boldsymbol{0}}(\nabla f)$ for $\boldsymbol{A}\hat{\boldsymbol{x}} = \boldsymbol{b}$.

problems where for example one of the choice probabilities is very low, the algorithm tends to be slower.

4.2 Local Linear Convergence of the Algorithm

We first show that with $\tau \in [0.5, 1)$, the distance between the solution at successive iterations and the optimal solution is non-increasing.

Lemma 2. *Let $d_k = ||\boldsymbol{x}^k - \boldsymbol{x}^*||$, where $\boldsymbol{x}^*$ is the optimal solution and $\boldsymbol{x}^k$ is the k-th iterate. Then $d_k \leq d_{k-1}$ for all $k > 0$ if $\tau \geq 0.5$.*

Proof: From the definition of d_k, we have

$$d_{k+1}^2 = ||\boldsymbol{x}^{k+1}-\boldsymbol{x}^*||^2 = ||\boldsymbol{x}^k+\alpha_k\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)-\boldsymbol{x}^*||^2 = d_k^2+\alpha_k^2||\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)||^2-2\alpha_k\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)^T(\boldsymbol{x}^*-\boldsymbol{x}^k). \quad (28)$$

Since $f(\cdot)$ is concave, we have

$$\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)^T(\boldsymbol{x}^* - \boldsymbol{x}^k) \geq f(\boldsymbol{x}^*) - f(\boldsymbol{x}^k) = \epsilon_k.$$

Note that $\epsilon_k \geq 0$ by definition. Combined with inequality (28), we have

$$d_{k+1}^2 \leq d_k^2 - \alpha_k(2\epsilon_k - \alpha_k||\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)||^2).$$

From the Armijo's rule, we have

$$f(\boldsymbol{x}^{k+1}) \geq f(\boldsymbol{x}^k) + \tau\alpha_k\|\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)\|^2.$$

Therefore,

$$\alpha_k\|\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)\|^2 \leq \frac{1}{\tau}(f(\boldsymbol{x}^{k+1}) - f(\boldsymbol{x}^k)) = \frac{1}{\tau}(\epsilon_k - \epsilon_{k+1}).$$

Hence, we have

$$\begin{aligned} d_{k+1}^2 &\leq d_k^2 - \alpha_k(2\epsilon_k - \alpha_k||\overline{\boldsymbol{\nabla}}f(\boldsymbol{x}^k)||^2), \\ &\leq d_k^2 - \alpha_k\left(2\epsilon_k - \frac{1}{\tau}(\epsilon_k - \epsilon_{k+1})\right), \\ &\leq d_k^2 - \alpha_k\left(2\epsilon_k - 2(\epsilon_k - \epsilon_{k+1})\right), \\ &\leq d_k^2 - 2\alpha_k\epsilon_{k+1}, \\ &\leq d_k^2, \end{aligned}$$

where the third inequality uses the fact that $\tau \geq 0.5$. ■

Note that the above proof is very similar to the proof of Proposition 9.1.2 in [3] for the unconstrained convex case.

We are now ready to discuss the rate of convergence of the algorithm by choosing the parameter τ, carefully.

Theorem 7. *If there exists a $\gamma \in (0,1)$ such that $||\boldsymbol{x}^0 - \boldsymbol{x}^*|| \leq \gamma x^*_{\min}$ where $x^*_{\min} = \min_i x^*_i$ and if $\tau = 0.5$, then*

$$f(\boldsymbol{x}^*) - f(\boldsymbol{x}^k) \leq \theta^k \left(f(\boldsymbol{x}^*) - f(\boldsymbol{x}^0) \right),$$

where $\theta = 1 - \min\{m, \beta m/L\}$ with m defined as the strong convexity constant in the proof of Theorem 3 and

$$L = \frac{9n}{4(1-\gamma)x^*_{\min}\sigma_1} \left\| \boldsymbol{\Sigma}^{1/2} \right\|^4 + \frac{n}{((1-\gamma)x^*_{\min}\sigma_1)^{1/2}} \left\| \boldsymbol{\Sigma}^{1/2} \right\|^2,$$

where $\sigma_1 = \lambda_1(\boldsymbol{\Sigma})$.

Proof: From Lemma 2, we know that $||\boldsymbol{x}^k - \boldsymbol{x}^*|| \leq \gamma x^*_{\min}$ for all $k > 0$. So we can restrict the feasible region to $\mathcal{X} = \boldsymbol{\Delta}_{n-1} \cap \{\boldsymbol{x} : ||\boldsymbol{x} - \boldsymbol{x}^*|| \leq \gamma x^*_{\min}\}$. It is easily seen that for all $\boldsymbol{x} \in \mathcal{X}$, $\min_i x_i \geq (1-\gamma)x^*_{\min}$ and, therefore, $\mathcal{X} \subset \text{int}(\boldsymbol{\Delta}_{n-1})$. Applying Theorem 8 given in the Appendix, $\overline{\boldsymbol{\nabla}} f(\boldsymbol{x}^k)$ is Lipschitz continuous over $\mathcal{X}$ with Lipschitz constant

$$L = \frac{9n}{4(1-\gamma)x^*_{\min}\sigma_1} \left\| \boldsymbol{\Sigma}^{1/2} \right\|^4 + \frac{n}{((1-\gamma)x^*_{\min}\sigma_1)^{1/2}} \left\| \boldsymbol{\Sigma}^{1/2} \right\|^2.$$

The result follows from the linear convergence rate result (see Boyd and Vandenberghe [7]) for $\tau \leq 0.5$, with $\theta = 1 - \min\{2m\tau, 2\beta\tau m/L\}$, where m is the strong convexity constant. In our case, $\tau = 0.5$, so we have

$$f(\boldsymbol{x}^*) - f(\boldsymbol{x}^k) \leq \theta^k \left(f(\boldsymbol{x}^*) - f(\boldsymbol{x}^0) \right),$$

where $\theta = 1 - \min\{m, \beta m/L\}$ with m being the constant in the proof of Theorem 3. ■

Since the algorithm converges globally (see Iusem [23]), Theorem 7 shows that there exists a large enough integer M, such that, after M iterations the algorithm converges linearly. The algorithm is thus locally linearly convergent. The typical behavior of the algorithm is presented in Figure 1. We provide a more detailed computational study regarding the convergence of the algorithm in the next section.

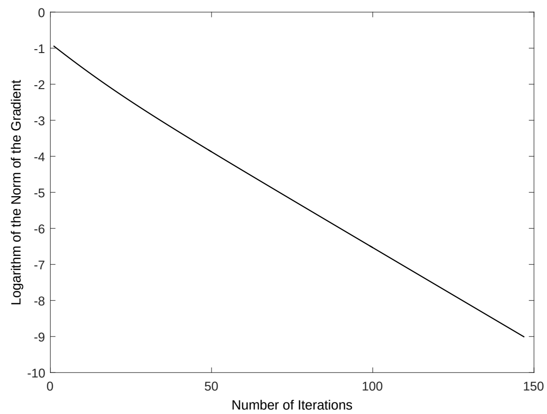

Figure 1: Local convergence of the algorithm for a random instance with $n = 100$.

5 Computational Results

In the first set of numerical experiments, we compare the computational times and accuracy of the gradient ascent method developed in this paper for the CMM model with an SDP solver that is suitable for solving large scale SDPs. In the second set of experiments, we numerically test the convergence of the gradient method for the CMM model. In the third set of experiments, we compare the choice probabilities from the CMM and MNP model. The results help illustrate the efficacy of the model.

5.1 Comparison of SDP and the Gradient Method for the CMM Model

The SDP in (12) was solved using the SDPNAL+ version 0.3 (beta) while the code for the gradient ascent method was developed in MATLAB R2014[5]. The computational experiments were run on a laptop with an Intel(R) i7-5600U CPU processor (2.6 GHz) with 4GB RAM.

The number of alternatives n were varied in the set $\{100, 200, \ldots, 1000\}$. The mean of the utilities was randomly generated in $[0, 1]^n$. The covariance matrix $\boldsymbol{\Sigma} = \boldsymbol{V}\text{Diag}(\boldsymbol{d})\boldsymbol{V}^T$ was randomly generated by choosing the eigenvalues in the vector $\boldsymbol{d}$ uniformly in $(0, 1]^n$ and the eigenvectors in $\boldsymbol{V}$ using an orthogonalization of a random n by n matrix with each entry in $[-1, 1]$. For each size n, 10 instances were randomly generated. In the computational experiments, we used the default settings for SPDNAL+ version 0.3. For the gradient method, the parameters were set as $\alpha_0 = 0.1$, $\tau = 0.5$, $\beta = 0.6$ with $\epsilon =$ 1e-4. To compare the accuracy of the methods, we evaluated the error

[5]The code for the gradient method and the test instances can be obtained from the webpages of the authors.

measured in L_2-distance between the choice probability vectors obtained from the SDP solver and the gradient ascent method:

$$\text{error}_{\text{prob}} = \|\boldsymbol{x}^*_{sdp} - \boldsymbol{x}^*_{grd}\|_2,$$

where $\boldsymbol{x}^*_{sdp}$ and $\boldsymbol{x}^*_{grd}$ are the solutions obtained from the SDP solver and the gradient ascent method, respectively. We also evaluated the difference in the optimal objective value as follows:

$$\text{error}_{\text{obj}} = |f(\boldsymbol{x}^*_{sdp}) - f(\boldsymbol{x}^*_{grd})|.$$

The results are provided in Table 1 which clearly indicates that both the methods are very close in terms of choice probabilities and the objective value. In Table 2, the computational times for the two methods are provided which illustrates that the gradient method converges much faster for this set of instances in comparison to the SDP solver.

n	error	1	2	3	4	5	6	7	8	9	10
100	Prob	1.63e-5	2.46e-5	3.69e-5	0.743e-5	3.08e-5	0.84e-5	2.92e-5	2.04e-5	2.47e-5	3.17e-5
	Obj	0.0056	0.0071	0.0055	0.0020	0.0054	0.0076	0.0067	0.0056	0.0057	0.0077
200	Prob	2.67e-5	2.53e-5	8.76e-5	2.36e-4	2.109e-4	2.672e-4	1.867e-4	2.06e-5	3.32e-5	2.39e-5
	Obj	0.0071	0.0077	0.0077	0.0060	0.0050	0.0067	0.0046	0.0075	0.0085	0.0076
300	Prob	9.40e-5	1.53e-4	0.88e-5	1.14e-4	2.34e-4	1.19e-4	1.04e-4	1.53e-4	2.08e-4	1.22e-4
	Obj	0.0033	0.0021	0.0065	0.0018	0.0016	0.0004	0.0028	0.0004	0.0052	0.0020
400	Prob	1.73e-4	2.89e-4	7.14e-5	0.95e-5	4.17e-5	3.78e-5	5.23e-5	1.43e-4	3.75e-5	6.08e-5
	Obj	0.0031	0.0028	0.0006	0.0055	0.0028	0.0041	0.0026	0.0039	0.0033	0.0041
500	Prob	8.79e-5	5.30e-5	2.95e-5	1.80e-5	3.54e-4	3.01e-5	4.29e-4	3.28e-5	1.51e-4	5.40e-5
	Obj	0.0027	0.0099	0.0026	0.0070	0.0086	0.0065	0.0104	0.0073	0.0024	0.0004
600	Prob	2.14e-05	2.10e-05	2.21e-06	2.82e-05	2.52e-05	2.70e-03	2.13e-05	2.27e-05	3.70e-05	4.12e-05
	Obj	0.0055	0.0042	0.0021	0.0190	0.0044	0.1066	0.0047	0.0046	0.0047	0.0161
700	Prob	7.16e-05	4.61e-05	1.45e-04	3.99e-05	2.31e-05	3.63e-05	1.82e-04	2.60e-05	5.60e-05	3.38e-05
	Obj	0.0015	0.0012	0.0016	0.0010	0.0066	0.0019	0.0051	0.0074	0.0093	0.0017
800	Prob	4.49e-05	1.78e-04	3.01e-05	2.19e-04	4.32e-04	4.51e-05	2.84e-04	6.64e-04	5.08e-04	7.62e-05
	Obj	0.0015	0.0000	0.0010	0.0058	0.0166	0.0017	0.0163	0.0013	0.0113	0.0132
900	Prob	1.12e-04	4.53e-05	4.08e-05	7.41e-04	4.86e-05	3.72e-04	3.06e-05	2.79e-04	6.29e-05	5.20e-05
	Obj	0.0137	0.0108	0.0055	0.0299	0.0120	0.0171	0.0009	0.0038	0.0008	0.0117
1000	Prob	3.88e-05	3.60e-04	3.62e-05	2.53e-04	7.74e-05	3.45e-05	5.56e-05	1.16e-04	6.13e-05	3.95e-05
	Obj	0.0019	0.0088	0.0084	0.0004	0.0045	0.0024	0.0158	0.0036	0.0098	0.0091

Table 1: Comparison of SDPNAL+ and the gradient method in terms of accuracy for 10 instances for each n.

5.2 Convergence of the Gradient Method for the CMM model

In the second set of numerical experiments, we study the convergence behavior of the algorithm. Theorem 7 shows that the region of linear convergence for the algorithm depends on $x^*_{\min}$, which is the distance between the optimal solution and the relative boundary. To study the effect of $x^*_{\min}$, we plot the number of iterations versus the level of accuracy achieved by the algorithm within those

n	Time (sec)	1	2	3	4	5	6	7	8	9	10
100	Grad	0.40	0.34	0.34	0.34	0.37	0.37	0.34	0.37	0.37	0.34
	SDP	11.82	6.78	8.23	65.49	6.22	7.84	6.35	8.22	5.85	8.29
200	Grad	1.43	1.17	1.23	1.38	1.27	1.35	1.29	1.26	1.21	1.17
	SDP	31.31	30.38	29.53	33.97	34.25	31.63	32.85	29.79	30.84	31.85
300	Grad	2.51	2.68	2.27	2.60	2.24	2.46	2.71	2.58	2.34	2.26
	SDP	114.45	99.74	101.27	114.67	116.59	121.58	113.42	111.21	120.35	109.49
400	Grad	4.35	3.60	3.47	3.86	3.83	3.49	3.65	5.24	4.21	7.44
	SDP	274.00	314.95	282.51	256.59	271.39	266.01	284.29	297.44	176.16	299.89
500	Grad	6.44	8.39	5.75	5.17	4.96	5.05	5.44	4.91	5.30	5.66
	SDP	617.63	548.72	527.94	477.75	585.20	467.12	521.08	462.04	499.21	492.89
600	Grad	13.61	14.24	12.62	12.94	13.35	13.35	13.49	14.22	19.07	13.36
	SDP	715.00	683.00	17903.00	2864.00	755.00	1829.00	655.00	649.00	891.00	658.00
700	Grad	23.16	23.49	20.87	19.92	21.38	20.51	19.84	20.73	20.87	21.30
	SDP	1220.30	1564.10	1692.80	1423.60	1163.10	1484.60	1264.40	1569.90	1186.40	1302.30
800	Grad	33.08	33.83	33.86	38.14	39.65	43.21	39.65	29.87	33.80	39.68
	SDP	2304.20	1880.30	2330.00	2091.30	2812.10	2325.40	2717.60	2715.70	2600.10	1862.70
900	Grad	49.18	52.07	53.41	48.36	52.50	48.95	53.83	48.90	47.92	48.81
	SDP	3665.10	3595.60	3663.40	3809.30	3728.10	3406.50	3571.90	3890.80	3233.50	3443.50
1000	Grad	60.60	71.76	71.93	66.79	63.91	61.58	63.19	82.55	79.03	62.07
	SDP	4846.90	5220.90	5325.90	5398.60	4999.20	4840.20	4954.00	4815.50	5220.80	5406.00

Table 2: Comparison of SDP solver SDPNAL+ and the gradient method in terms of computational times for 10 instances for each n.

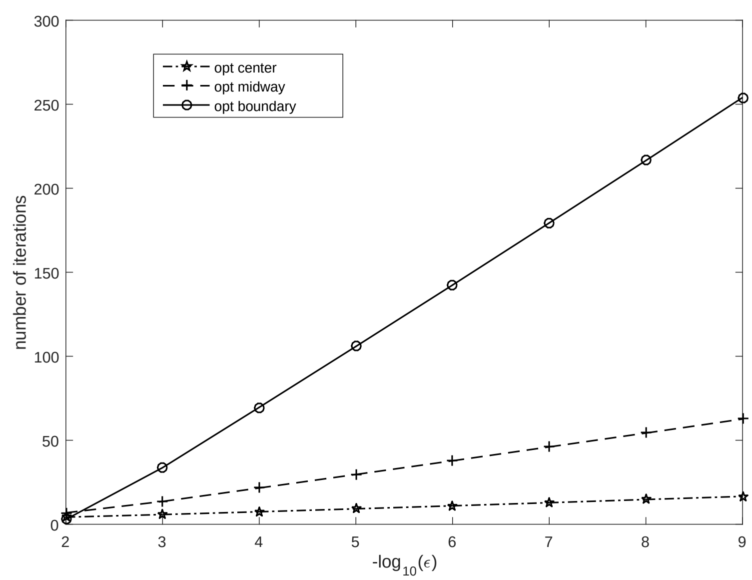

Figure 2: Average behavior of the algorithm with different $x^*_{\min}$.

iterations, i.e., the tolerance ϵ, in Figure 2. To plot Figure 2, we picked $n = 100$ and randomly generated a covariance matrix $\boldsymbol{\Sigma}$. Next, we chose a facet which is the convex combination of 10 randomly picked extreme points, and let the center of the facet be $\boldsymbol{x}^{\text{bd}}$. We considered three scenarios: In the 'opt center' scenario, we let $\boldsymbol{x}^* = 0.9\boldsymbol{x}^{\text{ct}} + 0.1\boldsymbol{x}^{\text{bd}}$, where $\boldsymbol{x}^{\text{ct}} = \{1/100, \ldots, 1/100\}$ is the center of the unit simplex; in the 'opt midway' scenario, we let $\boldsymbol{x}^* = 0.5\boldsymbol{x}^{\text{ct}} + 0.5\boldsymbol{x}^{\text{bd}}$; in the 'opt boundary' scenario, we let $\boldsymbol{x}^* = 0.1\boldsymbol{x}^{\text{ct}} + 0.9\boldsymbol{x}^{\text{bd}}$. Note that we can choose

$$\boldsymbol{\mu} = \boldsymbol{g}(\boldsymbol{x}^*) = -\frac{1}{2}\text{diag}\left(\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}^*))^\dagger\boldsymbol{\Sigma}^{1/2}\right) + \boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}^*))^\dagger\boldsymbol{\Sigma}^{1/2}\boldsymbol{x}^*,$$

to ensure that the optimal solution is $\boldsymbol{x}^*$. Clearly, $x^*_{\min} = 0.009$, 0.005 and 0.001 for these three scenarios. We randomly generate a starting point $\boldsymbol{x}^0$. We then vary ϵ from 10^{-2} to 10^{-9} and record the corresponding number of iterations. Figure 2 is obtained by averaging the results of 20 independent replications. From the plot, we can clearly observe the local linear convergence behavior for all three scenarios. As the optimal solution approaches to the boundary, the slope of the plot increases indicating that the constant in the linear convergence rate result increases as $x^*_{\min}$ decreases. This is also predicted by the theoretical results.

To study the influence of the starting point, we plot the average number of iterations and computation times versus the location of the starting point in the Figure 3. To obtain Figure 3, we

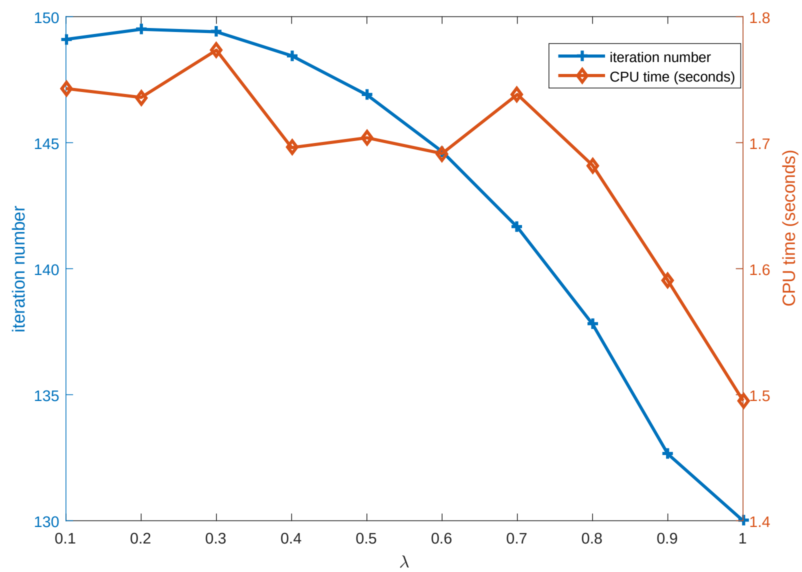

Figure 3: Average number of iterations and CPU time versus the location of the initial point.

the settings as in Figure 2. Instead of varying the optimal solution, we randomly generate $\boldsymbol{\mu}$ once to use in all experiments but choose various starting points on a line segment between a boundary point and the center of the simplex by setting $\boldsymbol{x}^0 = \lambda\boldsymbol{x}^{\text{ct}} + (1-\lambda)\boldsymbol{x}^{\text{bd}}$ and varying the parameter

λ. We fix the tolerance to $\epsilon = 10^{-6}$. From the figure, we find that required number of iterations to achieve the fixed tolerance level and the corresponding CPU times do not change much with respect to the starting point. The figure indicates that the location of the optimal solution seems to play a more important role for convergence than the location of the initial starting point.

5.3 Comparison of the CMM model and MNP model

In the last set of numerical experiments, we compare the choice probabilities for the MNP model obtained with simulation and for the CMM model obtained with the gradient ascent method.

5.3.1 Small size examples from Börsch-Supan and Hajivassiliou [6]

We first provide a comparison of the MNP and CMM choice probabilities for small size examples taken from Börsch-Supan and Hajivassiliou [6]. A popular alternative to the simple frequency simulator for MNP is the GHK simulator (see Geweke [14], [15] , Hajivassiliou [17], and Keane [24], [25]). The GHK simulator makes use of draws from truncated univariate normal distributions and requires evaluation of univariate integrals. Börsch-Supan and Hajivassiliou [6] have provided four examples with 5 alternatives to show that the GHK simulator produces probability estimates with substantially smaller variances than the simple frequency simulator. The details of the examples are provided in Table 3. Example 1 involves mild correlations and has a small choice probability for the first alternative, Example 2 has slightly higher correlations, Example 3 has some large correlation coefficients while Example 4 has a choice probability close to 0.5 with mild correlations. Comparison of the choice probabilities obtained from the GHK simulator for the MNP model and the gradient ascent method for the CMM model are provided in Table 3. The results indicate that the choice probability estimate for alternative 1 from the two models are fairly close to each other though developed under different assumptions on the utilities. For examples 1 and 2, where the choice probability of alternative 1 is small, the CMM model gives a higher choice probability for alternative 1 to be the most preferred one in comparison with the MNP model. On the other hand for examples 3 and 4, where the choice probability of alternative 1 is larger. The CMM model gives a slightly lower chance for alternative 1 to be the most preferred one in comparison with the MNP model.

Example	Parameters	MNP	CMM
1	$\Delta\boldsymbol{\mu} = \begin{pmatrix} -1.00 \\ -0.75 \\ -0.50 \\ -0.20 \end{pmatrix}, \Delta\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.2 & 0.3 & 0.1 \\ 0.2 & 1 & 0.4 & 0.3 \\ 0.3 & 0.4 & 1 & 0.5 \\ 0.1 & 0.3 & 0.5 & 1 \end{pmatrix}$	0.02409 (0.00068)	0.05366
2	$\Delta\boldsymbol{\mu} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \Delta\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.2 & 0.2 & 0.2 \\ 0.2 & 1 & 0.4 & 0.4 \\ 0.2 & 0.4 & 1 & 0.6 \\ 0.2 & 0.4 & 0.6 & 1 \end{pmatrix}$	0.15037 (0.00444)	0.15668
3	$\Delta\boldsymbol{\mu} = \begin{pmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \end{pmatrix}, \Delta\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.9 & 0 & 0 \\ 0.9 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.95 \\ 0 & 0 & 0.95 & 1 \end{pmatrix}$	0.64773 (0.00773)	0.63789
4	$\Delta\boldsymbol{\mu} = \begin{pmatrix} 1.50 \\ 0.75 \\ 0.50 \\ 0.75 \end{pmatrix}, \Delta\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.5 & 0.2 & 0.1 \\ 0.5 & 1 & 0.5 & 0.2 \\ 0.2 & 0.5 & 1 & 0.5 \\ 0.1 & 0.2 & 0.5 & 1 \end{pmatrix}$	0.49716 (0.01394)	0.47787

Table 3: Comparison of the choice probability for alternative 1 between the MNP and CMM model. $\Delta\boldsymbol{\mu}$ and $\Delta\boldsymbol{\Sigma}$ are the mean and the covariance matrix for the utilities $(\tilde{u}_1 - \tilde{u}_2, \tilde{u}_1 - \tilde{u}_3, \tilde{u}_1 - \tilde{u}_4, \tilde{u}_1 - \tilde{u}_5)^T$. The number in parenthesis indicates the standard deviation of the estimator.

5.3.2 Larger example from Jester rating dataset

In this example, we compare the choice probabilities from the CMM and the MNP model where data is available regarding the utilities of a large number of alternatives. We use the rating dataset from the Jester Online Joke Recommender System, in particular Dataset 2+[6]. The data consists of more than 2 million continuous ratings for 150 jokes collected from over 50000 individuals. Each individual provides ratings between -10 and 10 for a subset of the jokes, 10 of the jokes have never been rated and therefore excluded from the dataset.

To generate the utility parameter, we capture the data in a matrix of size 50000 by 140, whose the $(i, j)^{th}$ entry corresponds to the rating of individual i for joke j, if it exists. For the ratings that are incomplete, we use the standard Collaborative Filtering (CF) method, which is widely used in recommendation engines. The user-based version of CF estimates a missing rating from individual i for joke j based on existing ratings for joke j from a set of individuals who are similar to individual i. Alternatively, the item-based version of CF uses the existing ratings of individual i for other items. We use the item-based CF in our application, since it is more suitable in situations where the number of items is significantly smaller than the number of individuals (see Ekstrand, Riedl, and Konstan [11] for a recent exhaustive survey on the topic).

To begin with, let us provide the details of the item-based CF. Let r_j denote the j^{th} column

[6]http://eigentaste.berkeley.edu/dataset/

of the data matrix and $r(i,j)$ the existing rating from individual i for joke j. We calculate the *estimated* rating $\hat{r}(i,j)$ for individual i and joke j as follows:

$$\hat{r}(i,j) = \frac{\sum_{k \in J_i} w(j,k) r(i,k)}{\sum_{k \in J_i} |w(j,k)|},$$

where J_i is the set of jokes that have been rated by individual i and $w(j,k)$ is a measure of similarity between jokes j and k. Although there are other similarity measures in the literature, we use the cosine similarity, defined as $w(j,k) = \frac{r_j \cdot r_k}{\|r_j\|_2 \|r_k\|_2}$, for its simplicity, popularity, and good predictive properties. Similarly, the ratings can be estimated with alternative methods as well, nevertheless, the weighted average approach is a popular choice.

We use the *completed* data matrix to estimate the mean ratings $\boldsymbol{\mu} \in \Re^{140}$ over all users and the corresponding covariance matrix $\boldsymbol{\Sigma} \in \Re^{140 \times 140}$. Using these two parameter values, we calculate the choice probabilities, i.e., the probability that a joke j is the most prefered among all jokes,

1. Using the CMM and the gradient ascent algorithm developed in this paper with tolerance level $\epsilon = 10^{-3}$ and the rest of the parameters as in previous section.

2. Using the MNP model and the GHK simulator described above with 50000 samples.

We also calculate a basic in-sample statistic corresponding to the number of times a joke has the highest rating divided by the number of individuals. (Whenever there is a tie between l jokes for the highest rating, the count is incremented by $1/l$ instead of 1.) The choice probabilities from the CMM and MNP models together with the in-sample probabilities are provided in Figure 4. From the figure, we observe that the alternatives with very small choice probabilities in MNP take on higher choice probability values in the CMM model. On the other hand, the alternatives with larger choice probabilities in MNP take on smaller choice probability values in the CMM model. These results mirror the observations from the previous section. A possible explanation for this observation is that the distribution of the random utilities that maximizes expected agent utility in the CMM model is a mixture of multivariate normal distributions. The mixture of normals is a fat-tailed distribution and tends to give higher probabilities to the events that are low probability events in the standard normal distribution. In terms of the trend, however the results clearly indicate that the alternatives that are more preferred in one model are also more preferred in the other model.

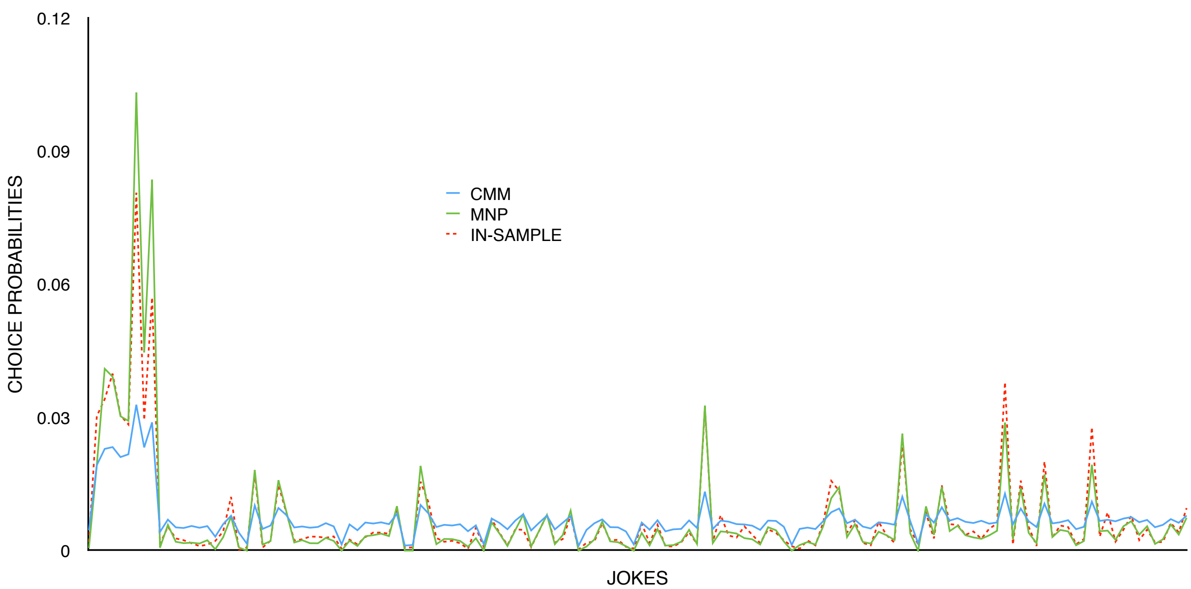

Figure 4: Choice Probabilities from CMM, MNP, and In-Sample for the Jester Dataset

6 Conclusion

In this paper, we have described a convex optimization approach to compute choice probabilities with correlated utilities. The choice model is derived for the joint distribution of the random utilities that maximizes expected agent utility given only the mean, variance and covariance information. Unlike MNP, the assumption of normality is dropped in this model. In contrast to MNP models where the choice probabilities are evaluated through simulation, we use a simple gradient ascent method to find the choice probabilities. The biggest advantage of the convex optimization approach is that one can compute choice probabilities for many alternatives with correlated utilities in a reasonable amount of time. In this era with consumers having more and more alternative options and increasing amounts of information, this paper proposes a new approach to computing choice probabilities that scales well with size. The next research question is to develop efficient inference techniques for the CMM model.

Acknowledgement

The authors would like to thank Kristin Wood (SUTD), Teo Chung Piaw (NUS), Zheng Zhichao (SMU) and Rudabeh Meskarian (SUTD) for valuable discussions on this paper.

Appendix

Proof [Equivalence of formulations (10) and (12)]

Step 1: To show that optimal value of (10) $\leq$ optimal value of (12).

Consider an optimal solution to the semidefinite program in (10) denoted by $\{\boldsymbol{W_i^*}, \boldsymbol{y_i^*}, x_i^*\}$ for $i \in [n]$. We consider the case with all the x_i^* values being strictly positive. Let $x_i = x_i^*$ and $\boldsymbol{Y}^T\boldsymbol{e_i} = \boldsymbol{y_i^*}$ for all i. We start by verifying that the following matrix in (12) is positive semdefinite:

$$\begin{pmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{Y}^T & \boldsymbol{\mu} \\ \boldsymbol{Y} & \text{Diag}(\boldsymbol{x}) & \boldsymbol{x} \\ \boldsymbol{\mu}^T & \boldsymbol{x}^T & 1 \end{pmatrix} \succeq 0,$$

To see this, observe that:

$$\begin{aligned}
\begin{pmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{\mu} \\ \boldsymbol{\mu}^T & 1 \end{pmatrix} - \begin{pmatrix} \boldsymbol{Y}^T \\ \boldsymbol{x}^T \end{pmatrix} \text{Diag}(\boldsymbol{x})^{-1} \begin{pmatrix} \boldsymbol{Y} & \boldsymbol{x} \end{pmatrix} &= \begin{pmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T - \boldsymbol{Y}^T \text{Diag}(\boldsymbol{x})^{-1}\boldsymbol{Y} & \boldsymbol{\mu} - \boldsymbol{Y}^T\boldsymbol{e} \\ \boldsymbol{\mu}^T - \boldsymbol{e}^T\boldsymbol{Y} & 1 - \boldsymbol{e}^T\boldsymbol{x} \end{pmatrix}, \\
&= \begin{pmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T - \sum_i \frac{\boldsymbol{y_i^*}\boldsymbol{y_i^*}^T}{x_i^*} & \boldsymbol{\mu} - \sum_i \boldsymbol{y_i^*} \\ \boldsymbol{\mu}^T - \sum_i \boldsymbol{y_i^*}^T & 1 - \sum_i x_i^* \end{pmatrix}, \\
&= \begin{pmatrix} \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T - \sum_i \frac{\boldsymbol{y_i^*}\boldsymbol{y_i^*}^T}{x_i} & \boldsymbol{0} \\ \boldsymbol{0}^T & 0 \end{pmatrix}, \\
&\succeq 0,
\end{aligned}$$

where the third equality comes from the feasibility condition in (10) and the positive semidefiniteness of the matrix in the last step follows from Schur's lemma since $\boldsymbol{W_i^*} \succeq \boldsymbol{y_i^*}\boldsymbol{y_i^*}^T/x_i$ for $i \in [n]$. Thus the solution $\{\boldsymbol{Y}, \boldsymbol{x}\}$ is feasible to the semidefinite program (12) with the same objective value. The case with some of the variables $x_i^* = 0$ is handled similarly by dropping the rows and columns corresponding to the zero entries.

Step 2: To show that optimal value of (10) $\geq$ optimal value of (12).

Consider an optimal solution to the semidefinite program in (12) denoted by $\{\boldsymbol{Y^*}, \boldsymbol{x^*}\}$. Consider the case where all the x_i^* values are strictly positive. From Schur's lemma, the positive semidefiniteness of the matrix in (12) is equivalent to following two conditions:

$$\begin{aligned}
\boldsymbol{\Lambda} \;=\; \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^T - \boldsymbol{Y^*}^T \text{Diag}(\boldsymbol{x^*})^{-1}\boldsymbol{Y^*} &\succeq 0, \\
\boldsymbol{Y^*}^T\boldsymbol{e} &= \boldsymbol{\mu}.
\end{aligned}$$

Define:

$$\begin{pmatrix} \boldsymbol{W_i} & \boldsymbol{y_i} \\ \boldsymbol{y_i}^T & x_i \end{pmatrix} = \begin{pmatrix} \boldsymbol{Y^*}^T\boldsymbol{e_i}\boldsymbol{e_i}^T\boldsymbol{Y^*}/x_i^* + \boldsymbol{\Lambda}/n & \boldsymbol{Y^*}^T\boldsymbol{e_i} \\ \boldsymbol{e_i}^T\boldsymbol{Y^*} & x_i^* \end{pmatrix}, \quad \forall i \in [n].$$

This is a feasible solution to the semidefinite program (10) with the same objective value. As before, the case with some of the $x_i^* = 0$ can be handled by dropping the rows and columns corresponding to the zero entries. Taken together, we obtain the desired result. ■

Proof [Optimality of (18) for formulation (17)]

The dual formulation for the semidefinite program (17) is given as:

$$\begin{aligned} V_D^*(\boldsymbol{x}) = \min \quad & \boldsymbol{\Sigma} \cdot \boldsymbol{Y_1} + \boldsymbol{S}(\boldsymbol{x}) \cdot \boldsymbol{Y_2} \\ \text{s.t.} \quad & \begin{pmatrix} \boldsymbol{Y_1} & -\boldsymbol{I}/2 \\ -\boldsymbol{I}/2 & \boldsymbol{Y_2} \end{pmatrix} \succeq 0, \end{aligned}$$

where $\boldsymbol{I}$ is an identity matrix of size $n \times n$. The optimality conditions for the primal and dual semidefinite programs are given as:

1. Primal feasibility:
$$\begin{pmatrix} \boldsymbol{\Sigma} & \hat{\boldsymbol{Y}}^{*T} \\ \hat{\boldsymbol{Y}}^* & \boldsymbol{S}(\boldsymbol{x}) \end{pmatrix} \succeq 0.$$

2. Dual feasibility:
$$\begin{pmatrix} \boldsymbol{Y_1^*} & -\boldsymbol{I}/2 \\ -\boldsymbol{I}/2 & \boldsymbol{Y_2^*} \end{pmatrix} \succeq 0.$$

3. Complementary slackness:
$$\begin{pmatrix} \boldsymbol{\Sigma} & \hat{\boldsymbol{Y}}^{*T} \\ \hat{\boldsymbol{Y}}^* & \boldsymbol{S}(\boldsymbol{x}) \end{pmatrix} \begin{pmatrix} \boldsymbol{Y_1^*} & -\boldsymbol{I}/2 \\ -\boldsymbol{I}/2 & \boldsymbol{Y_2^*} \end{pmatrix} = 0.$$

Expanding the complementary slackness condition, we get

$$\begin{aligned} &\text{(a)} \quad \boldsymbol{\Sigma} \boldsymbol{Y_1^*} - \hat{\boldsymbol{Y}}^{*T}/2 = 0 \\ &\text{(b)} \quad -\boldsymbol{\Sigma}/2 + \hat{\boldsymbol{Y}}^{*T} \boldsymbol{Y_2^*} = 0 \\ &\text{(c)} \quad \hat{\boldsymbol{Y}}^* \boldsymbol{Y_1^*} - \boldsymbol{S}(\boldsymbol{x})/2 = 0 \\ &\text{(d)} \quad -\hat{\boldsymbol{Y}}^*/2 + \boldsymbol{S}(\boldsymbol{x}) \boldsymbol{Y_2^*} = 0 \end{aligned}$$

From conditions (a) and (b), we get the equality:

$$\boldsymbol{\Sigma} \boldsymbol{Y_1^*} \boldsymbol{Y_2^*} = \boldsymbol{\Sigma}/4.$$

The matrices $\boldsymbol{Y_1^*}$ and $\boldsymbol{Y_2^*}$ are hence nonsingular related through an inverse: $\boldsymbol{Y_1^*} = \boldsymbol{Y_2^*}^{-1}/4$. Condition (a) implies that $\boldsymbol{Y_1^*} = \boldsymbol{\Sigma}^{-1} \hat{\boldsymbol{Y}}^{*T}/2$. Using condition (c), we obtain the matrix equality:

$$\hat{\boldsymbol{Y}}^* \boldsymbol{Y_1^*} = \boldsymbol{S}(\boldsymbol{x})/2 = \hat{\boldsymbol{Y}}^* \boldsymbol{\Sigma}^{-1} \hat{\boldsymbol{Y}}^{*T}/2.$$

The solution to this quadratic matrix equation is given as:

$$\hat{\boldsymbol{Y}}^{*T} = \boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\left(\left(\boldsymbol{S}(\boldsymbol{x})^{1/2}\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right)^{1/2}\right)^{\dagger}\boldsymbol{S}(\boldsymbol{x})^{1/2}.$$

Also,

$$\boldsymbol{Y}_{\boldsymbol{1}}^{*} = \boldsymbol{Y}_{\boldsymbol{2}}^{*-1}/4 = \frac{1}{2}\boldsymbol{S}(\boldsymbol{x})^{1/2}\left(\left(\boldsymbol{S}(\boldsymbol{x})^{1/2}\boldsymbol{\Sigma}\boldsymbol{S}(\boldsymbol{x})^{1/2}\right)^{1/2}\right)^{\dagger}\boldsymbol{S}(\boldsymbol{x})^{1/2}.$$

Lemmas for Sections 3 and 4

Lemma 3. *Suppose* $\boldsymbol{x} > \boldsymbol{0}$ *and let* $x_{\min} = \min_i x_i$, $x_{\max} = \max_i x_i$. *Then:*

$$||L_{1/2}(Diag(\boldsymbol{x}), \boldsymbol{E})|| \le \frac{||\boldsymbol{E}||}{2x_{\min}^{1/2}},$$

and

$$||L_{-1/2}(Diag(\boldsymbol{x}), \boldsymbol{E})|| \le \frac{n||\boldsymbol{E}||}{2x_{\min}^{3/2}}.$$

Proof: The Fréchet derivative for the matrix inverse function is given as:

$$L_{-1}(\boldsymbol{X}, \boldsymbol{E}) = -\boldsymbol{X}^{-1}\boldsymbol{E}\boldsymbol{X}^{-1}.$$

The Fréchet derivative for the matrix square root function, which exists when $\boldsymbol{X}$ is positive definite, is the unique solution to the Sylvester equation (refer to Kenney and Laub [26], Higham [20]):

$$\boldsymbol{X}^{1/2}L_{1/2}(\boldsymbol{X}, \boldsymbol{E}) + L_{1/2}(\boldsymbol{X}, \boldsymbol{E})\boldsymbol{X}^{1/2} = \boldsymbol{E}. \tag{29}$$

Following the chain rule (Theorem 3.4 in Higham [20]), we have

$$L_{-1/2}(\boldsymbol{X}, \boldsymbol{E}) = L_{1/2}(\boldsymbol{X}^{-1}, L_{-1}(\boldsymbol{X}, \boldsymbol{E})) = L_{1/2}(\boldsymbol{X}^{-1}, -\boldsymbol{X}^{-1}\boldsymbol{E}\boldsymbol{X}^{-1}),$$

and therefore,

$$\boldsymbol{X}^{-1/2}L_{-1/2}(\boldsymbol{X}, \boldsymbol{E}) + L_{-1/2}(\boldsymbol{X}, \boldsymbol{E})\boldsymbol{X}^{-1/2} = -\boldsymbol{X}^{-1}\boldsymbol{E}\boldsymbol{X}^{-1}. \tag{30}$$

Combining equations (29) and (30), we have

$$L_{-1/2}(\boldsymbol{X}, \boldsymbol{E}) = -\boldsymbol{X}^{-1/2}L_{1/2}(\boldsymbol{X}, \boldsymbol{E})\boldsymbol{X}^{-1/2}. \tag{31}$$

Define $\boldsymbol{L} = L_{1/2}(\mathrm{Diag}(\boldsymbol{x}), \boldsymbol{E})$. From equation (29), we have:

$$\mathrm{Diag}(\boldsymbol{x})^{1/2}\boldsymbol{L} + \boldsymbol{L}\mathrm{Diag}(\boldsymbol{x})^{1/2} = \boldsymbol{E},$$

which implies that

$$L_{i,j} = \frac{E_{i,j}}{x_i^{1/2} + x_j^{1/2}} \leq \frac{E_{i,j}}{2x_{\min}^{1/2}}.$$

Therefore, we have

$$||L|| = \sqrt{\sum_{i,j} L_{i,j}^2} \leq \sqrt{\sum_{i,j} \frac{E_{i,j}^2}{4x_{\min}}} = \frac{||\boldsymbol{E}||}{2x_{\min}^{1/2}}.$$

In addition, by equation (31)

$$||L_{-1/2}(\mathrm{Diag}(\boldsymbol{x}), \boldsymbol{E})|| = ||\mathrm{Diag}(\boldsymbol{x})^{-1/2}\boldsymbol{L}\mathrm{Diag}(\boldsymbol{x})^{-1/2}|| \leq ||\mathrm{Diag}(\boldsymbol{x})^{-1/2}||^2||\boldsymbol{L}|| = \frac{n||\boldsymbol{E}||}{2x_{\min}^{3/2}}.$$

■

Lemma 4. *Let* $\boldsymbol{B} = \boldsymbol{A} - \boldsymbol{u}\boldsymbol{u}^T$. *Then* $\lambda_1(\boldsymbol{B}) \leq \lambda_1(\boldsymbol{A})$, *and*

$$\lambda_{i-1}(\boldsymbol{A}) \leq \lambda_i(\boldsymbol{B}) \leq \lambda_i(\boldsymbol{A}), \quad \forall i = 2, \ldots, n.$$

Proof: The proof can be found on page 97-98 of [45].

Lemma 5. *Let* $\boldsymbol{D} = Diag(d_1, ..., d_n)$ *be a diagonal matrix with* $d_i > 0, \forall i \in [n]$. *Let* $\boldsymbol{\Sigma}$ *be a positive definite matrix. Then* $\lambda_1(\boldsymbol{\Sigma}^{1/2}\boldsymbol{D}\boldsymbol{\Sigma}^{1/2}) \geq \lambda_1(\boldsymbol{\Sigma}) \min_i\{d_i\}$.

Proof: Since $eig(AA^T) = eig(A^TA)$, we have

$$\lambda_1(\boldsymbol{\Sigma}^{1/2}\boldsymbol{D}\boldsymbol{\Sigma}^{1/2}) = \lambda_1(\boldsymbol{D}^{1/2}\boldsymbol{\Sigma}\boldsymbol{D}^{1/2}).$$

But

$$\boldsymbol{D}^{1/2}\boldsymbol{\Sigma}\boldsymbol{D}^{1/2} \succeq \lambda_1(\boldsymbol{\Sigma})\boldsymbol{D}^{1/2}\boldsymbol{I}\boldsymbol{D}^{1/2} = \lambda_1(\boldsymbol{\Sigma})\boldsymbol{D}.$$

Therefore, for all $\boldsymbol{v}$ with $||\boldsymbol{v}|| = 1$, we have

$$\boldsymbol{v}^T\boldsymbol{D}^{1/2}\boldsymbol{\Sigma}\boldsymbol{D}^{1/2}\boldsymbol{v} \geq \boldsymbol{v}^T\lambda_1(\boldsymbol{\Sigma})\boldsymbol{D}\boldsymbol{v} \geq \lambda_1(\boldsymbol{\Sigma}) \min_i\{d_i\}.$$

■

Theorem 8. *Assume that* $\boldsymbol{\Sigma} \succ 0$. *For any feasible direction* $\boldsymbol{v} \in \overline{\boldsymbol{\Delta}}_{n-1}$ *with* $\|\boldsymbol{v}\| = 1$ *and* $\boldsymbol{x} \in rint(\boldsymbol{\Delta}_{n-1})$,

$$|f''_{\boldsymbol{x},\boldsymbol{v}}(0)| \leq \frac{9n}{4x_{\min}\sigma_1} \left\|\boldsymbol{\Sigma}^{1/2}\right\|^4 + \frac{n}{(x_{\min}\sigma_1)^{1/2}} \left\|\boldsymbol{\Sigma}^{1/2}\right\|^2,$$

where $f_{\boldsymbol{x},\boldsymbol{v}}(t) = V(\boldsymbol{x} + t\boldsymbol{v}) > 0$, $\sigma_1 = \lambda_1(\boldsymbol{\Sigma})$ *and* $x_{\min} = \min_i \ x_i$.

Proof: Let $\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{g}_1(\boldsymbol{x}) + \boldsymbol{g}_2(\boldsymbol{x})$ where

$$\boldsymbol{g}_1(\boldsymbol{x}) = -\frac{1}{2}\text{diag}\left(\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\right), \quad \boldsymbol{g}_2(\boldsymbol{x}) = \boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\boldsymbol{x}.$$

From the proof of Theorem 4, we know that:

$$\begin{aligned}
& \boldsymbol{g}_1(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v} - \boldsymbol{g}_1(\boldsymbol{x})^T\boldsymbol{v} \\
= \; & -\frac{1}{2}\boldsymbol{v}^T\text{diag}\left(\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}+\epsilon\boldsymbol{v}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\right) + \frac{1}{2}\boldsymbol{v}^T\text{diag}\left(\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\boldsymbol{\Sigma}^{1/2}\right) \\
= \; & -\frac{1}{2}\boldsymbol{v}^T\text{diag}\left(\boldsymbol{\Sigma}^{1/2}\left((\boldsymbol{T}^{1/2}(\boldsymbol{x}+\epsilon\boldsymbol{v}))^{\dagger} - (\boldsymbol{T}^{1/2}(\boldsymbol{x}))^{\dagger}\right)\boldsymbol{\Sigma}^{1/2}\right) \\
= \; & -\frac{1}{2}\boldsymbol{v}^T\text{diag}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}) + \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)^{-1/2} - \overline{\boldsymbol{\Lambda}}(\boldsymbol{x})^{-1/2} \end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\right) \\
= \; & -\frac{1}{2}\boldsymbol{v}^T\text{diag}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}), \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right) + o\left(||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||\right) \end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\right) \\
= \; & -\frac{1}{2}\boldsymbol{v}^T\text{diag}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}), \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right) \end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\right) + o(\epsilon).
\end{aligned}$$

Therefore, we have:

$$
\begin{aligned}
& ||\boldsymbol{g}_1(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v} - \boldsymbol{g}_1(\boldsymbol{x})^T\boldsymbol{v}|| \\
= \; & || -\frac{1}{2}\boldsymbol{v}^T\text{diag}\left(\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}), \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)\end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\right) + o(\epsilon)|| \\
\leq \; & \frac{1}{2}||\boldsymbol{v}|| \cdot \left\|\boldsymbol{\Sigma}^{1/2}\right\|^2 \cdot \left\|\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}), \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)\end{pmatrix}\boldsymbol{P}^T\right\| + o(\epsilon) \\
\leq \; & \frac{1}{2}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2 \cdot \left\|\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}), \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)\end{pmatrix}\boldsymbol{P}^T\right\| + o(\epsilon) \\
\leq \; & \frac{1}{2}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2 \cdot \left\|L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}), \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)\right\| + o(\epsilon) \\
\leq \; & \frac{1}{2}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2 \cdot \frac{n||\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||}{2\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))} + o(\epsilon) \\
= \; & \frac{1}{2}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2 \cdot \frac{n||\boldsymbol{E}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})||}{2\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))} + o(\epsilon) \\
= \; & \frac{1}{2}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2 \cdot \frac{n||\epsilon\boldsymbol{\Sigma}^{1/2}\left(\text{Diag}(\boldsymbol{v}) - \boldsymbol{x}\boldsymbol{v}^T - \boldsymbol{v}\boldsymbol{x}^T\right)\boldsymbol{\Sigma}^{1/2}||}{2\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))} + o(\epsilon) \\
\leq \; & \frac{\epsilon}{2}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^4 \cdot \frac{n||\boldsymbol{v}||(1+2||\boldsymbol{x}||)}{2\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))} + o(\epsilon) \\
\leq \; & \frac{\epsilon}{2}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^4 \cdot \frac{3n}{2\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))} + o(\epsilon).
\end{aligned}
$$

Note that the last inequality holds since $||\boldsymbol{x}|| \leq 1$ for all $\boldsymbol{x} \in \boldsymbol{\Delta}_{n-1}$. On the other hand,

$$
\begin{aligned}
& \boldsymbol{g}_2(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v} - \boldsymbol{g}_2(\boldsymbol{x})^T\boldsymbol{v} \\
= \; & \boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}+\epsilon\boldsymbol{v}))^\dagger\boldsymbol{\Sigma}^{1/2}(\boldsymbol{x}+\epsilon\boldsymbol{v}) - \boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}(\boldsymbol{T}^{1/2}(\boldsymbol{x}))^\dagger\boldsymbol{\Sigma}^{1/2}\boldsymbol{x} \\
= \; & \boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}) + \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)^{-1/2}\end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}(\boldsymbol{x}+\epsilon\boldsymbol{v}) - \boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \overline{\boldsymbol{\Lambda}}(\boldsymbol{x})^{-1/2}\end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{x} \\
= \; & \boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}), \overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)\end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{x} + \epsilon\boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \overline{\boldsymbol{\Lambda}}(\boldsymbol{x})^{-1/2}\end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{v} + o(\epsilon).
\end{aligned}
$$

Therefore, we have

$$
\begin{aligned}
& \left\|\boldsymbol{g}_2(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v}-\boldsymbol{g}_2(\boldsymbol{x})^T\boldsymbol{v}\right\| \\
\leq\ & \left\|\boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)\end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{x}\right\| + \epsilon\left\|\boldsymbol{v}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{P}\begin{pmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \overline{\boldsymbol{\Lambda}}(\boldsymbol{x})^{-1/2}\end{pmatrix}\boldsymbol{P}^T\boldsymbol{\Sigma}^{1/2}\boldsymbol{v}\right\| + o(\epsilon) \\
\leq\ & \|\boldsymbol{x}\|\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2\left\|L_{-1/2}\left(\overline{\boldsymbol{\Lambda}}(\boldsymbol{x}),\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\right)\right\| + \epsilon\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2\left\|\overline{\boldsymbol{\Lambda}}(\boldsymbol{x})^{-1/2}\right\| + o(\epsilon) \\
\leq\ & \left\|\boldsymbol{\Sigma}^{1/2}\right\|^2\cdot\frac{n\|\overline{\boldsymbol{E}}_{\boldsymbol{v}}(\epsilon,\boldsymbol{x})\|}{2\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))} + \epsilon\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2\frac{n}{\lambda_2(\boldsymbol{T}(\boldsymbol{x}))^{1/2}} + o(\epsilon) \\
\leq\ & \epsilon\left\|\boldsymbol{\Sigma}^{1/2}\right\|^4\cdot\frac{3n}{2\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))} + \epsilon\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2\frac{n}{\lambda_2(\boldsymbol{T}(\boldsymbol{x}))^{1/2}} + o(\epsilon).
\end{aligned}
$$

In addition, from Lemma 4 and Lemma 5, we have

$$\lambda_2(\boldsymbol{T}(\boldsymbol{x})) = \lambda_2(\boldsymbol{\Sigma}^{1/2}\boldsymbol{S}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2}) \geq \lambda_1(\boldsymbol{\Sigma}^{1/2}\mathrm{Diag}(\boldsymbol{x})\boldsymbol{\Sigma}^{1/2}) \geq x_{\min}\sigma_1 > 0.$$

Therefore, we have

$$
\begin{aligned}
\left|f''_{\boldsymbol{x},\boldsymbol{v}}(0)\right| &= \left|\lim_{\epsilon\to 0^+}\frac{\boldsymbol{g}_1(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v}-\boldsymbol{g}_1(\boldsymbol{x})^T\boldsymbol{v}+\boldsymbol{g}_2(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v}-\boldsymbol{g}_2(\boldsymbol{x})^T\boldsymbol{v}}{\epsilon}\right| \\
&\leq \left|\lim_{\epsilon\to 0^+}\frac{\boldsymbol{g}_1(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v}-\boldsymbol{g}_1(\boldsymbol{x})^T\boldsymbol{v}}{\epsilon}\right| + \left|\lim_{\epsilon\to 0^+}\frac{\boldsymbol{g}_2(\boldsymbol{x}+\epsilon\boldsymbol{v})^T\boldsymbol{v}-\boldsymbol{g}_2(\boldsymbol{x})^T\boldsymbol{v}}{\epsilon}\right| \\
&\leq \frac{9n}{4\lambda_2^{3/2}(\boldsymbol{T}(\boldsymbol{x}))}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^4 + \frac{n}{\lambda_2(\boldsymbol{T}(\boldsymbol{x}))^{1/2}}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2 \\
&\leq \frac{9n}{4(x_{\min}\sigma_1)^{3/2}}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^4 + \frac{n}{(x_{\min}\sigma_1)^{1/2}}\left\|\boldsymbol{\Sigma}^{1/2}\right\|^2.
\end{aligned}
$$

■

Remark 2. *Theorem 8 develops an upper bound for the absolute value of the second order derivatives in direction* $\boldsymbol{v} \in \overline{\boldsymbol{\Delta}}_{n-1}$. *The significance of the theorem is that the second order derivative of* $V(\boldsymbol{x})$ *is bounded for all points* $\boldsymbol{x} \in rint(\boldsymbol{\Delta}_{n-1})$, *with the bound associated with the minimum components of* $\boldsymbol{x}$.

Bibliography

[1] S. Anderson, A. Palma, and J.-F. Thisse, *Discrete Choice Theory of Product Differentiation*, Cambridge: MIT Press, 1992.

[2] S. Anderson, A. D. Palma, and J. Thisse, *A representative consumer theory of the logit model*, International Economic Review, 29 (1988), pp. 461–466.

[3] A. Ben-Tal and A. Nemirovski, *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*, vol. 2, Siam, 2001.

[4] S. T. Berry, *Estimating discrete-choice models of product differentiation*, RAND Journal of Economics, 25 (1994), pp. 242–262.

[5] D. P. Bertsekas, *Nonlinear programming*, (1999).

[6] A. Börsch-Supan and V. A. Hajivassiliou, *Smooth unbiased multivariate probability simulators for maximum likelihood estimation of limited dependent variable models*, Journal of Econometrics, 58 (1993), pp. 347–368.

[7] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[8] D. Brownstone, D. S. Bunch, and K. Train, *Joint mixed logit models of stated and revealed preferences for alternative-fuel vehicles*, Transportation Research B, 34 (2000), pp. 315–338.

[9] J. Bunch, C. Nilsen, and D. Sorensen, *Rank-one modification of the symmetric eigenproblem*, Numerische Mathematik, 31 (1978), pp. 31–48.

[10] D. C. Dowson and B. V. Landau, *The fréchet distance between multivariate normal distributions*, Journal of Multivariate Analysis, 12 (1982), pp. 450–455.

[11] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, *Collaborative filtering recommender systems*, Foundations and Trends in Human-Computer Interaction, 4 (2010), pp. 81–173.

[12] G. Feng, X. Li, and Z. Wang, *Analysis of discrete choice models: A welfare-based framework*, Available at SSRN 2574466, (2015).

[13] I. R. Fudenberg, D. and T. Strzalecki, *Stochastic choice and revealed perturbed utility*, Econometrica, 83 (2015), pp. 2371–2409.

[14] J. Geweke, *Bayesian inference in econometric models using monte carlo integration*, Econometrica, 57 (1989), pp. 1317–1339.

[15] J. Geweke, *Efficient Simulation from the Multivariate Normal and Student-t Distributions Subject to Linear Constraints*, Defense Technical Information Center, 1992.

[16] G. H. Golub and C. F. V. Loan, *Matrix Computations*, John Hopkins University Press, 1996.

[17] V. A. Hajivassiliou and D. L. McFadden, *The method of simulated scores for the estimation of ldv models*, Econometrica, 66 (1998), pp. 863–896.

[18] V. A. Hajivassiliou, D. L. McFadden, and P. Ruud, *Simulation of multivariate normal rectangle probabilities and their derivatives: Theoretical and computational results*, Journal of Econometrics, 72 (1996), pp. 85–134.

[19] E. Heinz, *Beiträge zur Störungstheorie der Spektralzerlegung*, Math. Ann., 123 (1951), pp. 415–438.

[20] N. J. Higham, *Functions of Matrices: Theory and Computation*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2008.

[21] J. Hofbauer and W. H. Sandholm, *On the global convergence of stochastic fictitious play*, Econometrica, 70 (2002), pp. 2265–2294.

[22] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 1985.

[23] A. Iusem, *On the convergence properties of the projected gradient method for convex optimization*, Computational & Applied Mathematics, 22 (2003), pp. 37–52.

[24] M. P. Keane, *Four Essays in Empirical Macro and Labor Economics*, Brown University, 1990.

[25] M. P. Keane, *A computationally practical simulation estimator for panel data*, Econometrica, 62 (1994), pp. 95–116.

[26] C. Kenney and A. J. Laub, *Condition estimates for matrix functions*, SIAM Journal on Matrix Analysis and Applications, 10 (1989), pp. 191 – 209.

[27] K. Löwner, *Über monotone matrixfunktionen*, Mathematische Zeitschrift, 38 (1934), pp. 177–216.

[28] R. D. Luce, *Individual choice behavior*, John Wiley, New York, 1959.

[29] D. McFadden, *Conditional logit analysis of qualitative choice behavior. in paul zarembka, editor.*, Frontiers in Econometrics, (1974), pp. 105–142.

[30] D. McFadden, A. Karlqvist, L. Lundqvist, F. Snickars, and E. J. Weibull, *Modelling the choice of residential location*, Proc. Spatial Interaction Theory Planning Models, (1978), pp. 75–96.

[31] P. Mertikopoulos and W. H. Sandholm, *Learning in games via reinforcement and learning*, To appear in Mathematics of Operations Research, (2016).

[32] V. Mishra, K. Natarajan, H. Tao, and C.-P. Teo, *Choice prediction with semidefinite optimization when utilities are correlated*, IEEE Transactions on Automatic Control, 57 (2012), pp. 2450–2463.

[33] K. Natarajan, M. Song, and C.-P. Teo, *Persistency model and its applications in choice modeling*, Management Science, 55 (2009), pp. 453–469.

[34] K. Natarajan and C.-P. Teo, *On reduced semidefinite programs for second order moment bounds with applications*, To appear in Mathematical Programming, (2016).

[35] I. Olkin and F. Pukelsheim, *The distance between two random vectors with given dispersion matrices*, Linear Algebra and its Applications, 48 (1982), pp. 257 – 263.

[36] G. Parsons and M. Kealy, *Randomly drawn opportunity sets in a random utility model of lake recreation*, Land Economics, 68 (1992), pp. 93–206.

[37] R. T. Rockafellar, *Convex Analysis*, vol. 28, 1970.

[38] S. D. Selin Damla Ahipasaoglu, R. Meskarian, L. Magnanti, Thomas, and K. Natarajan, *Beyond normality: A cross moment-stochastic user equilibrium model*, Transportation Research Part B: Methodological, 81 (2015), pp. 333–354.

[39] A. Shapiro, *Extremal problems on the set of nonnegative definite matrices*, Linear Algebra and its Applications, 67 (1985), pp. 7 – 18.

[40] J. Swait and A. A. J. Marley, *Probabilistic choice (models) as a result of balancing multiple goals*, Journal of Mathematical Psychology, 57 (2013), pp. 1–14.

[41] K. C. Toh, M. J. Todd, and R. H. Tutuncu, *Sdpt3 - a matlab software package for semidefinite programming*, Optimization Methods and Software, 1 (1999), pp. 545–581.

[42] O. Toubia, D. I. Simester, J. R. Hauser, and E. Dahan, *Fast polyhedral adaptive conjoint estimation*, Marketing Science, 22 (2003), pp. 273–303.

[43] R. H. Tutuncu, K. C. Toh, and M. J. Todd, *Solving semidefinite-quadratic-linear programs using sdpt3*, Mathematical Programming Series B, 95 (2003), pp. 189–217.

[44] F. Verboven, *The nested logit model and representative consumer theory*, Economics Letters, 50 (1996), pp. 57–63.

[45] J. H. Wilkinson, *The algebraic eigenvalue problem*, vol. 87, Clarendon Press Oxford, 1965.

[46] L. Q. Yang, D. Sun, and K. C. Toh, *Sdpnal+: a majorized semismooth newton-cg augmented lagrangian method for semidefinite programming with nonnegative constraints*, Mathemtical Programming Computation, 7 (2015), pp. 331–366.

[47] X.-Y. Zhao, D. Sun, K. C. Toh, and M. J. Todd, *A newton-cg augmented lagrangian method for semidefinite programming*, SIAM Journal of Optimization, 20 (2010), pp. 1737–1765.