RESEARCH STATEMENT

ZI YANG

My research interests are optimization, especially polynomial and convex optimization, tensor computation, and their applications in machine learning and data science. The following is a summary of projects that are done during Ph.D. study.

The project *Detecting Copositivity of Tensors*[5] uses techniques in polynomial optimization to detect copositivity of symmetric tensors. In the project *The Saddle Point Problem of Polynomials*[3], we develop a first practical numerical algorithm to solve the general non convex-concave saddle point problem of polynomials. *Hermitian Tensor Decompositions*[4] studies properties of Hermitian tensors and their decompositions which have important applications in quantum physics. *Learning Diagonal Gaussian Mixture Models and Incomplete Tensor Decomposition*[2] applies tensor decomposition techniques to recover unknown parameters of the Gaussian mixture model from observed samples. *Separability of Hermitian Tensors and PSD Decompositions* [1] reformulates the problem as a truncated moment problem which can be solved by semidefinite relaxations. Moreover, techniques from tensor decompositions are also used to certify separability for Hermitian tensors.

1. Detection of Copositive Tensors and Matrices

The first work focuses on detecting copositivity for matrices and tensors. Copositive matrices and tensors have broad applications, including complementarity problems, vacuum stability, etc. However, how to detect copositivity was an open question in prior work. We proposed the first algorithm that can detect copositivity for all matrices and tensors.

A symmetric tensor $\mathcal{A}$ of order m and dimension n is a multi-dimensional array $\mathcal{A} := (\mathcal{A}_{i_1 \dots i_m})$ and $\mathcal{A}_{i_1 i_2 \dots i_m} = \mathcal{A}_{j_1 j_2 \dots j_m}$ whenever $(i_1, i_2, \dots, i_m)$ is a permutation of $(j_1, j_2, \dots, j_m)$. Each symmetric tensor $\mathcal{A}$ is determined by the polynomial $\mathcal{A}(x) := \sum_{1 \le i_1, i_2, \cdots, i_m \le n} \mathcal{A}_{i_1 i_2 \cdots i_m} x_{i_1} x_{i_2} \cdots x_{i_m}$. A symmetric tensor $\mathcal{A}$ is called copositive if $\mathcal{A}(x) \ge 0$ for all $x \ge 0$. $\mathcal{A}$ is copositive if and only if $v^* \ge 0$ where v^* is the minimum value of the following optimization problem.

$$\left\{ \begin{array}{ll} v^* := \min & \mathcal{A}(x) \\ \quad\quad\;\; \text{s.t.} & e^T x = \sum_{i=1}^n x_i = 1, \, (x_1, \dots, x_n) \ge 0. \end{array} \right. \tag{1.1}$$

The problem (1.1) is a polynomial optimization problem. A standard approach for solving it is to apply classical Lasserre relaxations. However classical Lasserre relaxations may not have finite convergence and certifying its convergence is also hard. To avoid these issues, we express Lagrange multipliers by polynomials and additionally add an ball constraint to improve the efficiency to get the new problem

$$\left\{ \begin{array}{ll} v^* := \min & \mathcal{A}(x) \\ \quad\quad\;\; \text{s.t} & e^T x - 1 = p_1(x) x_1 = \cdots = p_n(x) x_n = 0, \\ & 1 - \|x\|^2 \ge 0, \, x_i \ge 0, \, p_i(x) \ge 0 \, (i = 1, \dots, n). \end{array} \right. \tag{1.2}$$

where e is the all one vector and $p_i := \frac{\partial \mathcal{A}(x)}{\partial x_i} - m\mathcal{A}(x)$ $(i = 1, 2, \ldots, n)$ are Lagrange multipliers in polynomial form. The above problem (1.2) is in fact equivalent to problem (1.1). Classical Lasserre relaxations are applied to solve problem (1.2). Let v_k be the minimum value of the kth order relaxation of problem (1.2). We proved that $v_1 \leq v_2 \leq \cdots \leq v_k = v^*$ for all k big enough. Thus if $v_k \geq 0$ for some k, then $\mathcal{A}$ is copositive. But even if $v_k < 0$ for some k, we cannot conclude $v^* < 0$ since it is possible that $v_k < v^*$ and it can be very hard to verify if $v_k = v^*$. Therefore, we proposed the problem (1.3)

$$\left\{ \begin{array}{ll} \min & \xi^T [x]_m \\ \text{s.t} & e^T x - 1 = 0,\ x \geq 0, v_k - \mathcal{A}(x) \geq 0. \end{array} \right. \tag{1.3}$$

where ξ is randomly chosen vector. Suppose $\mathcal{A}$ is not copositive. For big enough k, the minimizer of the kth order relaxation of problem (1.3) can give us a point $u \geq 0$ such that $\mathcal{A}(u) < 0$ which refutes copositivity of $\mathcal{A}$.

Theorem 1.1 ([5]). *Let $\mathcal{A}$ be a symmetric tensor. Our algorithm has the properties:*

(i) *If $\mathcal{A}$ is copositive, then our algorithm must stop with $v_k \geq 0$, when k is sufficiently large.*
(ii) *If $\mathcal{A}$ is not copositive, then our algorithm must return a point $u \geq 0$ with $f(u) < 0$, when k is sufficiently large.*

This work gives a complete semidefinite algorithm for detecting tensor copositivity. Our algorithm converts the problem of detecting copositivity into a sequence of semidefinite programming problems. It is proven that the algorithm must terminate in finite steps, so the copositivity can be detected exactly in finitely many iterations. This is the first algorithm that can detect copositivity in finitely many iterations for all matrices and tensors. This work has been published on *SIAM Journal on Optimization.*

2. The Saddle Point Problem

The second work is about the challenging saddle point problem. They are of fundamental importance in min-max optimization, game theory, etc. The saddle point problem is a classically open question. We proposed the first efficient numerical algorithm to solve the saddle point problem of polynomials.

Let $F(x, y)$ be a polynomial in (x, y) and X, Y are two semialgebraic sets defined by polynomials. (x^*, y^*) is said to be saddle point of $F(x, y)$ over $X \times Y$ if

$$F(x^*, y) \leq F(x^*, y^*) \leq F(x, y^*) \quad \forall x \in X, y \in Y$$

The above implies that

$$\min_{x\in X} \max_{y\in Y} F(x, y) \;=\; F(x^*, y^*) \;=\; \max_{y\in Y} \min_{x\in X} F(x, y). \tag{2.1}$$

Therefore all saddle points share the same objective value.

Suppose (x^*, y^*) is a saddle point, then x^* is a minimizer of $F(x, y^*)$ over X and y^* is a maximizer of $F(x^*, y)$ over Y. It involves two separate optimization problems that must be solved simultaneously. KKT conditions and Lagrange multipliers can be applied to connect these two problems. In our work, Lagrange multipliers are expressed by polynomials in term of original variables. Let $\phi(x, y), \psi(x, y)$ be equality and inequality polynomials respectively in KKT conditions. Every saddle point must satisfy those KKT conditions, so it leads to the following problem

$$\begin{array}{rl} \min_{x\in X, y\in Y} & F(x,y) \\ \text{s.t.} & \phi(x,y)=0, \psi(x,y)\geq 0 \end{array} \tag{2.2}$$

Each saddle point must be feasible to the above problem. Thus solving the problem can give us candidate saddle points. Let (x^*, y^*) be a minimizer of the above problem. If x^* is a minimizer of $F(x, y^*)$ over X and y^* is a maximizer of $F(x^*, y)$ over Y, then (x^*, y^*) is a saddle point; otherwise, such (x^*, y^*) is not a saddle point i.e., there exists $u \in X$ and/or there exists $v \in Y$ such that $F(u, y^*) - F(x^*, y^*) < 0$ and/or $F(x^*, v) - F(x^*, y^*) > 0$. The points u, v can be used to add new constraints $F(u, y) - F(x, y) \geq 0$ and/or $F(x, y) - F(x, v) \geq 0$ to problem (2.2). All saddle points must satisfy the newly added constraints, so the new problem (2.2) will exclude (x^*, y^*) while not excluding any saddle point. By repeatedly adding new constraints to problem (2.2), we will finally find a saddle point or certify the nonexistence of saddle points. It results in the following Algorithm 2.1.

Algorithm 2.1 ([3])**.** *Let X, Y be two semialgebraic sets and $F, \phi_x, \psi_x, \phi_y, \psi_y$ be polynomials as in problem (2.2). Let $K_1 = K_2 = \mathcal{S}_a := \emptyset$.*

Step 1: *If (2.2) is infeasible, then F does not have a saddle point over $X \times Y$ and stop; otherwise, solve (2.2) for a set K^0 of minimizers. Let $k := 0$.*

Step 2: *For each $(x^*, y^*) \in K^k$, do the following:*

- (a): *Solve $\min_{x\in X} F(x, y^*)$ to get a set of minimizers $S_1(y^*)$ with minimum value $\vartheta_1(y^*)$. If $F(x^*, y^*) > \vartheta_1(y^*)$, update $K_1 := K_1 \cup S_1(y^*)$.*
- (b): *Solve $\max_{y\in Y} F(x^*, y)$ to get a set of maximizers $S_2(x^*)$ with maximum value $\vartheta_2(x^*)$. If $F(x^*, y^*) < \vartheta_2(x^*)$, update $K_2 := K_2 \cup S_2(x^*)$.*
- (c): *If $\vartheta(y^*) = F(x^*, y^*) = \vartheta(x^*)$, update $\mathcal{S}_a := \mathcal{S}_a \cup \{(x^*, y^*)\}$.*

Step 3: *If $\mathcal{S}_a \neq \emptyset$, then each point in $\mathcal{S}_a$ is a saddle point and stop; otherwise go to Step 4.*

Step 4: *Solve the minimization problem (2.3)*

$$\begin{array}{rl} \min_{x\in X, y\in Y} & F(x,y) \\ s.t. & \phi(x,y)=0, \psi(x,y)\geq 0 \\ & F(u,y)-F(x,y)\geq 0\,(\forall\, u\in K_1), \\ & F(x,v)-F(x,y)\leq 0\,(\forall\, v\in K_2). \end{array} \tag{2.3}$$

If (2.3) is infeasible, then F has no saddle point and stop; otherwise, compute a set K^{k+1} of optimizers for (2.3). Let $k := k + 1$ and go to Step 2.

All optimization problems in Algorithm 2.1 can be solved by semidefinite relaxations. The convergence of Algorithm 2.1 is shown as follows.

Theorem 2.2 ([3])**.** *For generic polynomial F and semialgebraic sets X, Y, Algorithm 2.1 must terminate after finitely many iterations. Moreover, if $\mathcal{S}_a \neq \emptyset$, then each $(x^*, y^*) \in \mathcal{S}_a$ is a saddle point. If $\mathcal{S}_a = \emptyset$, then there is no saddle point.*

In conclusion, this work basically solved the saddle point problem of polynomials. We constructed Algorithm 2.1 for computing saddle points. For almost all polynomial saddle point problems, our algorithm can either compute a saddle point or detect the nonexistence of saddle points. This is the first efficient numerical algorithm that can solve general saddle point problems of polynomials. This work is under minor revision of *Foundations of Computational Mathematics.*

3. Hermitian Tensor Decompositions

Hermitian tensors are natural extension of Hermitian matrices to higher order, but their properties are very different. Every quantum mixed stated can be expressed by Hermitian tensors, so Hermitian tensors have important applications in quantum physics. Hermitian tensor decompositions, real Hermitian tensors, flattenings, positivity, and separability are studied in the work.

A tensor $\mathcal{H} = (\mathcal{H}_{i_1 \ldots i_m j_1 \ldots j_m}) \in \mathbb{C}^{n_1 \times \cdots \times n_m \times n_1 \times \cdots \times n_m}$ is said to be *Hermitian* if $\mathcal{H}_{i_1 \ldots i_m j_1 \ldots j_m} = \overline{\mathcal{H}_{j_1 \ldots j_m i_1 \ldots i_m}}$ for all $i_1, ..., i_m$ and $j_1, ..., j_m$ in the range, where $\bar{a}$ is the conjugate of a. The set of all such Hermitian tensors is denoted as $\mathbb{C}^{[n_1,\ldots,n_m]}$.

A rank-1 Hermitian tensor in $\mathbb{C}^{[n_1,\ldots,n_m]}$ must be in the form

$$[v^1, v^2, \cdots, v^m]_{\otimes h} := v^1 \otimes v^2 \cdots \otimes v^m \otimes \overline{v^1} \otimes \overline{v^2} \cdots \otimes \overline{v^m}. \tag{3.1}$$

Every Hermitian tensor is a real linear combination of rank-1 Hermitian tensors, i.e. $\mathcal{H} = \sum_{i=1}^{r} \lambda_i [u_i^1, \ldots, u_m^i]_{\otimes h}$ for some real scalars λ_i and complex vectors u_i^j. The smallest such r is called the *Hermitian rank* of $\mathcal{H}$, denoted by $\operatorname{hrank}(\mathcal{H})$. The corresponding decomposition is called a *Hermitian rank decomposition.*

Hermitian tensors form a vector space over the real field. It is natural to consider the canonical basis of the vector space. For $c \in \mathbb{C}$ and tuples $I := (i_1, \ldots, i_m)$, $J := (j_1, \ldots, j_m)$, denote by $\mathcal{E}^{IJ}(c)$ the Hermtian tensor in $\mathbb{C}^{[n_1,\ldots,n_m]}$ such that

$$\left(\mathcal{E}^{IJ}(c)\right)_{i_1 \cdots i_m j_1 \cdots j_m} = \overline{\left(\mathcal{E}^{JI}(c)\right)}_{j_1 \cdots j_m i_1 \cdots i_m} = c$$

and all other entries are zeros. When $c = 1, \sqrt{-1}$, these tensors form the canonical basis. Hermitian ranks of all basis tensors and their Hermitian rank decompositions are determined in our work. For cleanness of the statement, explicit decompositions are omitted. Hermitian ranks of basis tensors are given as follows.

Theorem 3.1 ([4]). *Assume $n_1, \ldots, n_m \geq 2$, $I = (i_1, \ldots, i_m), J = (j_1, \ldots, j_m)$, and $c \neq 0$. If $I = J$, then* $\operatorname{hrank} \mathcal{E}^{IJ}(c) = 1$*; if $I \neq J$, then* $\operatorname{hrank} \mathcal{E}^{IJ}(c) = 2d$ *where d is the number of nonzero entries of $I - J$.*

An explicit decomposition is give in the following example.

Example 3.2. *For $I = (1, 2)$, $J = (3, 4)$ and $c \neq 0$, the basis tensor $\mathcal{E}^{(12)(34)}(c) \in \mathbb{C}^{[4,4]}$ has the Hermitian rank 4, with the following Hermitian rank decomposition (in the following $i := \sqrt{-1}$)*

$$\frac{1}{4}\left[\begin{pmatrix} c \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}\right]_{\otimes h} + \frac{1}{4}\left[\begin{pmatrix} c \\ 0 \\ -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix}\right]_{\otimes h} - \frac{1}{4}\left[\begin{pmatrix} c \\ 0 \\ i \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ i \end{pmatrix}\right]_{\otimes h} - \frac{1}{4}\left[\begin{pmatrix} c \\ 0 \\ -i \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ -i \end{pmatrix}\right]_{\otimes h}.$$

Real Hermitian tensors are Hermitian tensors that all entries are real. The subspace of real Hermitian tensors in $\mathbb{C}^{[n_1,\ldots,n_m]}$ is denoted by

$$\mathbb{R}^{[n_1,\ldots,n_m]} := \mathbb{C}^{[n_1,\ldots,n_m]} \cap \mathbb{R}^{n_1 \times \cdots \times n_m \times n_1 \times \cdots \times n_m}.$$

For real Hermitian tensors, we are interested in their real decompositions. $\mathcal{H} \in \mathbb{R}^{[n_1,\ldots,n_m]}$ is called $\mathbb{R}$-*Hermitian decomposable* if $\mathcal{H} = \sum_{i=1}^{r} \lambda_i [u_i^1, \ldots, u_i^m]_{\otimes h}$ for real vectors $u_i^j \in \mathbb{R}^{n_j}$ and real scalars $\lambda_i \in \mathbb{R}$. Not every real Hermitian tensor is $\mathbb{R}$-Hermitian decomposable which is very different from the complex case. We characterize when a tensor is $\mathbb{R}$-Hermitian decomposable in the following theorem.

Theorem 3.3 ([4]). *$\mathcal{A} \in \mathbb{R}^{[n_1,\ldots,n_m]}$ is $\mathbb{R}$-Hermitian decomposable if and only if*

$$\mathcal{A}_{i_1\ldots i_m j_1\ldots j_m} = \mathcal{A}_{k_1\ldots k_m l_1\ldots l_m} \tag{3.2}$$

for all labels such that $\{i_s, j_s\} = \{k_s, l_s\}$, $s = 1, \ldots, m$.

Matrix flattening is a significant tool while studying tensors. Hermitian flattening and Kronecker flattening are two special flattenings for Hermitian tensors. For the Hermitian tensor $\mathcal{H}$ with the decomposition $\mathcal{H} := \sum_{i=1}^{r} \lambda_i [u_i^1, \ldots, u_i^m]_{\otimes h}$, its Hermitian flattening matrix is

$$\mathfrak{m}(\mathcal{H}) = \sum\nolimits_{i=1}^{r} \lambda_i \, (u_i^1 \boxtimes \cdots \boxtimes u_i^m)(u_i^1 \boxtimes \cdots \boxtimes u_i^m)^*.$$

where $\boxtimes$ is the Kronecker product and a^* is the conjugate transpose of a. The canonical Kronecker flattening of $\mathcal{H}$ is

$$\kappa(\mathcal{H}) := \sum_{i=1}^{r} (u_i^1 \boxtimes \overline{u_i^1} \boxtimes \cdots \boxtimes u_i^{m-1} \boxtimes \overline{u_i^{m-1}})(u_i^m \boxtimes \overline{u_i^m})^T$$

$\operatorname{rank}(\mathfrak{m}(\mathcal{H}))$ and $\operatorname{rank}(\kappa(\mathcal{H}))$ are both lower bounds for $\operatorname{hrank}(\mathcal{H})$. However, the bounds can be very different.

Example 3.4. *For $m = 2$ and $n > 1$, consider the Hermitian tensor in $\mathbb{R}^{[n,n]}$*

$$\mathcal{H} = \sum\nolimits_{i,j=1}^{n} e_i \otimes e_i \otimes e_j \otimes e_j = \big(\sum\nolimits_{i=1}^{n} e_i \otimes e_i\big) \otimes \big(\sum\nolimits_{i=1}^{n} e_i \otimes e_i\big).$$

Then $\operatorname{hrank}(\mathcal{H}) \geq \operatorname{rank} \kappa(\mathcal{H}) = n^2$ *while* $\operatorname{rank} \mathfrak{m}(\mathcal{H}) = 1$.

A Hermitian tensor $\mathcal{H} \in \mathbb{H}^{[n_1,\ldots,n_m]}$ can be uniquely determined by the conjugate multi-quadratic polynomial $\mathcal{H}(x, \overline{x}) := \langle \mathcal{H}, [x_1, \ldots, x_m]_{\otimes_h} \rangle$, in the tuple $x := (x_1, \ldots, x_m)$ of complex vector variables $x_i \in \mathbb{C}^{n_i}$.

Let $\mathbb{F} = \mathbb{C}$ or $\mathbb{R}$. $\mathcal{H} \in \mathbb{F}^{[n_1,\ldots,n_m]}$ is called $\mathbb{F}$-*positive semidefinite* ($\mathbb{F}$-psd) if $\mathcal{H}(x, \overline{x}) \geq 0, \forall x_i \in \mathbb{F}^{n_i}$. Denote the cone of $\mathbb{F}$-psd Hermitian tensors by $\mathscr{P}_{\mathbb{F}}^{[n_1,\ldots,n_m]}$. Separable Hermitian tensors are closely related to psd Hermitian tensors. $\mathcal{H} \in \mathbb{F}^{[n_1,\ldots,n_m]}$ is called $\mathbb{F}$-*separable*if $\mathcal{H}$ has the decomposition $\mathcal{H} = \sum_{i=1}^{r} [u_i^1, \ldots, u_i^m]_{\otimes h}$ for some $u_i^j \in \mathbb{F}^{n_j}$. Denote the cone of $\mathbb{F}$-separable Hermitian tensors by $\mathscr{S}_{\mathbb{F}}^{[n_1,\ldots,n_m]}$.

The following theorem characterizes properties of $\mathscr{S}_{\mathbb{F}}^{[n_1,\ldots,n_m]}, \mathscr{P}_{\mathbb{F}}^{[n_1,\ldots,n_m]}$ and their duality relationship.

Theorem 3.5 ([4]). *$\mathscr{P}_{\mathbb{C}}^{[n_1,\ldots,n_m]}$ and $\mathscr{S}_{\mathbb{C}}^{[n_1,\ldots,n_m]}$ are proper cones, i.e. they are closed, convex, solid, and pointed. $\mathscr{P}_{\mathbb{R}}^{[n_1,\ldots,n_m]}$ and $\mathscr{S}_{\mathbb{R}}^{[n_1,\ldots,n_m]}$ are closed and convex. However, $\mathscr{P}_{\mathbb{R}}^{[n_1,\ldots,n_m]}$ is solid but not pointed; $\mathscr{S}_{\mathbb{R}}^{[n_1,\ldots,n_m]}$ is pointed but not solid. Moreover, $\mathscr{S}_{\mathbb{F}}^{[n_1,\ldots,n_m]}$ and $\mathscr{P}_{\mathbb{F}}^{[n_1,\ldots,n_m]}$ are dual to each other for $\mathbb{F} = \mathbb{R}, \mathbb{C}$.*

In the above, we talked about basis Hermitian tensors, real Hermitian tensors, flattenings, psd Hermitian tensors and separability. Besides these topics, this work also discussed Hermitian eigenvalues, Hermitian/conjugate sum of squares, hierarchy of SOS representations, reformulation of separability, and e.t.c.. The work has been published on *SIAM Journal on Matrix Analysis and Applications.*

4. Learning Diagonal Gaussian Mixture Models and Incomplete Tensor Decomposition

A Gaussian mixture model is a mixture of several Gaussian distributions. Learning Gaussian mixture models has applications across numerous fields, including speech recognition, economics, social science, and biology. In this work, we proposed a novel algorithm to learn diagonal Gaussian models from its moment tensors.

Consider a diagonal Gaussian mixture model with k components. For $i \in [k]$, let ω_i be the proportion of each component i ($\omega_i > 0$ and $\sum_{i=1}^k \omega_i = 1$) and each component distribution is a normal distribution $\mathcal{N}(\mu_i, \Sigma_i)$, where $\mu_i \in \mathbb{R}^d$ is the mean vector and $\Sigma_i = \operatorname{diag}(\sigma_{i1}^2, \ldots, \sigma_{id}^2) \in \mathbb{R}^{d\times d}$ is the diagonal covariance matrix. Given samples drawn from the Gaussian mixture model, we aim to estimate the unknown parameters $\{(\omega_i, \mu_i, \Sigma_i) : i \in [k]\}$ of the Gaussian mixture model.

We first show the moment structure hidden in model.

Theorem 4.1. *Let $M_3 := E(x \otimes x \otimes x)$ be the third order moment tensor where x is the random variable of dimension d for the Gaussian mixture model with parameters $\{(\omega_i, \mu_i, \Sigma_i) : i \in [k]\}$. Then*

$$M_3 = \sum_{i=1}^k \omega_i \mu_i \otimes \mu_i \otimes \mu_i + \sum_{j=1}^d (a_j \otimes e_j \otimes e_j + e_j \otimes a_j \otimes e_j + e_j \otimes e_j \otimes a_j)$$

where $a_j := \sum_{i=1}^k \omega_i \sigma_{ij}^2 \mu_i$ for $j = 1, \cdots, d$.

Note that the part $\sum_{j=1}^d (a_j \otimes e_j \otimes e_j + e_j \otimes a_j \otimes e_j + e_j \otimes e_j \otimes a_j)$ only contains indices (i_1, i_2, i_3) such that at least two of them are equal. Let $\mathcal{F} := \sum_{i=1}^k \omega_k \mu_i \otimes \mu_i \otimes \mu_i$. It holds that $\mathcal{T}_{i_1 i_2 i_3} = (M_3)_{i_1 i_2 i_3}$ whenever i_1, i_2, i_3 are distinct. M_3 can be estimated by samples of the Gaussian mixture model, so part of entries of $\mathcal{F}$ are known. From these known entries of $\mathcal{F}$, we developed a brand new algorithm to recover the decomposition of $\mathcal{F}$ by using generating polynomial. Since the decomposition of $\mathcal{F}$ is unique when k is small, the recovered decomposition can be used to find weights ω_i and mean vectors μ_i. Finally

$$\sum_{j=1}^d (a_j \otimes e_j \otimes e_j + e_j \otimes a_j \otimes e_j + e_j \otimes e_j \otimes a_j) = M_3 - \mathcal{T}$$

is known. The above equation is a linear system of covariances σ_{ij}^2. Therefore covariance matrices $\Sigma_1, \ldots, \Sigma_k$ are obtained by solving the above linear equations.

In practice, M_3 is never known exactly, but it can be estimated from samples. When the estimation of M_3 is sufficiently close to the ground truth, our algorithm can get good estimations of parameters. The following theorem characterizes the estimation quality.

Theorem 4.2. *Consider the d-dimensional diagonal Gaussian mixture model with parameters $\{(\omega_i, \mu_i, \Sigma_i) : i \in [r]\}$ where $r \le \frac{d}{2} - 1$. Let $\widehat{M_1}$ and $\widehat{M_3}$ be the estimations of $M_1 := \mathbb{E}(x)$ and $M_3 := \mathbb{E}(x \otimes x \otimes x)$ respectively. $\{(\omega_i^{opt}, \mu_i^{opt}, \Sigma_i^{opt}) : i \in [r]\}$ are estimations of parameters obtained from our algorithm with input $\widehat{M_1}, \widehat{M_3}$. If $\epsilon := \max(\|M_3 - \widehat{M_3}\|, \|M_1 - \widehat{M_1}\|)$ is small enough, then*

$$\|\mu_i - \mu_i^{opt}\| = O(\epsilon), \|\omega_i - \omega_i^{opt}\| = O(\epsilon), \|\Sigma_i - \Sigma_i^{opt}\| = O(\epsilon).$$

In conclusion, we proposed a new algorithm to learn Gaussian mixture models based on its moment tensors. First, we find the decomposition of a symmetric incomplete tensor generated from the third order moment. Then the decomposition is used to recover all unknown parameters of the Gaussian mixture model. Numerical experiments on both synthetic and real-world datasets demonstrate the outstanding performance of our algorithm compared to the traditional EM algorithm.

5. Separability of Hermitian Tensors and PSD Decompositions

As mentioned in section 3, detecting separability of Hermitian tensors is of great importance in quantum physics. In this work, we reformulated the problem as a truncated moment problem which can be solved by semidefinite programming. For separable Hermitian tensors with low $\mathbb{C}$-psd rank, the separability can be detected by using tensor decomposition.

Recall that every separable Hermitian tensor must be in the form of $\mathcal{H} = \sum_{i=1}^{r}[u_i^1, \dots, u_i^m]_{\otimes h}$. Let $x_i = x_i^{re} + \sqrt{-1}x_i^{im}$, then $[x_1, \dots, x_m]_{\otimes h} = R(\tilde{x}) + \sqrt{-1}I(\tilde{x})$ where $R(\tilde{x}), I(\tilde{x})$ are polynomials in $\tilde{x} := (x_1^{re}, x_1^{im}, \dots, x_m^{re}, x_m^{im})$. $\mathcal{H}$ can be rewritten as

$$\mathcal{H} = \int R(\tilde{x}) + \sqrt{-1}I(\tilde{x})d\mu \tag{5.1}$$

where $\mu := \sum_{i=1}^{r} \delta_{(\mathrm{Re}(u_i^1),\mathrm{Im}(u_i^1),\dots,\mathrm{Re}(u_i^m),\mathrm{Im}(u_i^m))}$ is a sum of Dirac measure and $\mathrm{Re}(a), \mathrm{Im}(a)$ denote the real part and imaginary part of a respectively. Thus we get the following result.

Proposition 5.1. *The Hermitian tensor $\mathcal{H}$ is separable if and only if there exists a measure μ supported on $\mathbb{S} := \{\tilde{x} = (x_1^{re}, x_1^{im}, \dots, x_m^{re}, x_m^{im}) | \, \|x_j^{re}\|^2 + \|x_j^{im}\|^2 = 1, x_j^{re}, x_j^{im} \in \mathbb{R}^{n_j}\}$ such that equation (5.1) holds.*

The above proposition reformulates detecting separability as a truncated moment problem. Let $F(\tilde{x})$ be a random sum of squares polynomial. Consider the truncated moment problem

$$\begin{array}{rl} \min_\mu & \int F(\tilde{x})d\mu \\ \text{s.t.} & \mathrm{Re}(\mathcal{H}) = \int R(\tilde{x})d\mu \\ & \mathrm{Im}(\mathcal{H}) = \int I(\tilde{x})d\mu \\ & \mu \text{ is a measure supported on } \mathbb{S} \end{array} \tag{5.2}$$

$\mathcal{H}$ is separable if and only if the above problem is feasible. The above problem can be solved by standard semidefinite relaxations. See our work [1] for more details. The convergence results are shown as follows.

Theorem 5.2. *Let $\mathcal{H}$ be a Hermitian tensor, then the following holds.*

- (i) *If $\mathcal{H}$ is separable, then under some assumptions, the kth order relaxation of (5.2) will give a measure that solves (5.2) for some k big enough.*
- (ii) *If $\mathcal{H}$ is not separable, then the kth order relaxation of (5.2) must be infeasible for some k big enough.*

We also proposed a new faster approach based on tensor decomposition since the previous method can be very slow for big $\mathcal{H}$.

$\mathcal{H}$ is separable if and only if there exist psd matrices B_{ij} such that $\mathfrak{m}(\mathcal{H}) = \sum_{i=1}^{s} B_{i1} \boxtimes \cdots \boxtimes B_{im}$. The smallest such s is called $\mathbb{C}$-psd rank of $\mathcal{H}$, denoted by $\text{hrank}_{psd}(\mathcal{H})$. Directly finding the $\mathbb{C}$-psd rank is usually very hard. Suppose $\mathcal{H} = \sum_{i=1}^{r} \lambda_i [u_i^1, \ldots, u_i^m]_{\otimes h}$, define its associated non-symmetric tensor

$$\mathbb{T}(\mathcal{H}) := \sum_{i=1}^{r} \lambda_i \mathfrak{v}(u_i^1 (u_i^1)^*) \otimes \cdots \otimes \mathfrak{v}(u_i^m (u_i^m)^*)$$

where $\mathfrak{v}(A)$ is the vectorization of matrix A. Decomposing $\mathbb{T}(\mathcal{H})$ can certify the separability of $\mathcal{H}$ and determine its $\mathbb{C}$-psd rank in some cases. If $\mathbb{T}(\mathcal{H})$ has rank s with the decomposition

$$\mathbb{T}(\mathcal{H}) = \sum_{i=1}^{s} \mathfrak{v}(B_{i1}) \otimes \cdots \otimes \mathfrak{v}(B_{im}), \tag{5.3}$$

with B_{ij} being a psd Hermitian matrix, then $\mathcal{H}$ is separable and $\text{hrank}_{psd}(\mathcal{H}) = s$.

We further proved that when $\text{hrank}_{psd}(\mathcal{H})$ is small, then $\mathbb{T}(\mathcal{H})$ will have the decomposition in the form of (5.3). In such case, separability can be verified by decomposing $\mathbb{T}(\mathcal{H})$. Decompositions of nonsymmetric tensors can be found by some implemented algorithms. It is usually much faster than the previous truncated moment approach and can handle bigger Hermitian tensors.

6. Future Work

In the future, I will certainly continue my work on optimization and tensor computation. Here is a brief plan for the future work.

- Polynomial and convex optimization has broad applications in various fields, including game theory, decision making, control theory, tensor computation. A big part of my future plan is to explore more techniques in polynomial optimization and apply them to solve related problems.
- Tensor computation has gained more and more interest because of its power in machine learning and data science. However, there are still lots of unsolved problems in this relatively new research area. In the future, I will continue to conduct research on tensor problems, like tensor decomposition, tensor completion, tensor eigenvalues, and so on.
- Tensors have broad applications in real-world and data science, including tensor regression, tensor PCA, tensor neural network, e.t.c. Work on these applications is also part of my future research.

References

[1] M. Dressler, J. Nie and Z. Yang, Separability of Hermitian Tensors and PSD Decompositions. Submitted to *Linear and Multilinear Algebra.*

[2] B. Guo, J. Nie and Z. Yang, Learning Diagonal Gaussian Mixture Models and Incomplete Tensor Decomposition. *In preparation.*

[3] J. Nie, Z. Yang and G. Zhou, The Saddle Point Problem of Polynomials. Submitted to *Foundations of Computational Mathematics*, under minor revision.

[4] J. Nie and Z. Yang, Hermitian Tensor Decompositions. *SIAM Journal on Matrix Analysis and Applications*, 41(3), pp. 1115-1144, 2020.

[5] J. Nie, Z. Yang and X. Zhang, A Complete Semidefinite Algorithm for Detecting Copositive Matrices and Tensors. *SIAM Journal on Optimization*, 28(4), pp. 2902-2921, 2018.