

**BỘ GIÁO DỤC
VÀ ĐÀO TẠO**

**VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM**

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ



Nguyễn Khánh Huyền

**BÀI TOÁN CỰC TIỂU CHUẨN NGUYÊN TỬ
CỦA MA TRẬN**

LUẬN VĂN THẠC SĨ: TOÁN HỌC

Hà Nội - 2022

**BỘ GIÁO DỤC
VÀ ĐÀO TẠO**

**VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM**

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ



Nguyễn Khánh Huyền

**BÀI TOÁN CỰC TIỂU CHUẨN NGUYÊN TỬ
CỦA MA TRẬN**

Chuyên ngành: Toán ứng dụng

Mã số: 8460112

LUẬN VĂN THẠC SĨ : TOÁN HỌC

NGƯỜI HƯỚNG DẪN KHOA HỌC :

TS. Lê Hải Yến

Hà Nội - 2022

LỜI CAM ĐOAN

Tôi xin cam đoan những gì viết trong luận văn là do quá trình tìm hiểu, học hỏi, trau dồi kiến thức của bản thân dưới sự hướng dẫn tận tình của TS. Lê Hải Yến. Mọi kết quả nghiên cứu cũng như ý tưởng của tác giả khác, nếu có đều được trích dẫn cụ thể trong luận văn. Đề tài luận văn này cho đến nay chưa được bảo vệ tại bất kì một hội đồng bảo vệ luận văn thạc sĩ nào. Tôi xin chịu trách nhiệm về những lời cam đoan.

Hà Nội, tháng 10 năm 2022

Học viên

Nguyễn Khánh Huyền

LỜI CẢM ƠN

Đầu tiên, tôi xin được bày tỏ lòng biết ơn sâu sắc nhất của mình tới TS. Lê Hải Yến, người đã trực tiếp hướng dẫn và giúp đỡ tôi xác định đề tài Luận văn chất lượng, cho tôi định hình được hướng nghiên cứu trong tương lai. Luận văn này được hoàn thành dưới sự hướng dẫn tận tình, tâm huyết của cô. Cô đã luôn quan tâm, giúp đỡ, động viên tôi rất nhiều trong suốt quá trình học tập và nghiên cứu để tôi có thể hoàn thành Luận văn.

Tôi xin gửi lời cảm ơn đến các thầy cô, những người đã trực tiếp giảng dạy cho tôi các kiến thức trong quá trình học tập cũng như nghiên cứu. Tôi xin cảm ơn tới Trung tâm Quốc tế Đào tạo và Nghiên cứu Toán học, Viện Toán học và cơ sở đào tạo Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam đã tạo điều kiện thuận lợi cho tôi về môi trường học tập trong quá trình thực hiện Luận văn này. Tôi cũng xin chân thành cảm ơn Quỹ VINIF đã hỗ trợ tài chính cho tôi năm nhất Cao học, giúp tôi có điều kiện tốt cũng như động lực hơn trong con đường nghiên cứu khoa học.

Tôi xin gửi lời cảm ơn TS. Đỗ Đức Hạnh, trưởng phòng nghiên cứu AI và Toán học ở Smartlog, cựu nghiên cứu viên ở Viện Toán học, đã nhiệt tình chỉ bảo tôi các kiến thức thực tế cũng như góc nhìn và tiềm năng ứng dụng thực tiễn của bài toán trong Luận văn. Tôi xin cảm ơn anh Võ Duy Trung, TS. Vũ Minh Tâm và các đồng nghiệp ở công ty Smartlog đã hết lòng giúp đỡ tôi trong quá trình làm Luận văn. Tôi cũng xin cảm ơn anh Kurt Bình - Tổng giám đốc công ty Smartlog đã tạo ra một môi trường nghiên cứu toán học tuyệt vời trong doanh nghiệp, cung cấp điều kiện cho tôi nghiên cứu, và áp dụng các bài toán lý thuyết trong ứng dụng thực tiễn doanh nghiệp, giúp kéo gần lại khoảng cách giữa toán học hàn lâm và xã hội.

Bên cạnh đó, trong quá trình học tập, nghiên cứu và thực hiện Luận văn, tôi còn nhận được nhiều sự quan tâm, hỗ trợ từ các quý thầy cô, bạn

bè trong Viện Toán học, đặc biệt là anh Nguyễn Xuân Quý và các bạn lớp cao học K2020B, Toán ứng dụng.

Cuối cùng, tôi xin bày tỏ lòng biết ơn tới gia đình tôi, những người đã luôn yêu thương và khích lệ tôi trong suốt quá trình học tập, nghiên cứu.

Danh sách hình vẽ

2.1	Biểu diễn bài toán Netflix dưới dạng ma trận.	17
2.2	Quan hệ giữa người dùng, các đặc điểm và bộ phim	18
3.1	Kết quả chạy thử nghiệm với kích thước ma trận cỡ 30×30 , số lượng quan sát được chiếm từ 20% đến 80%.	39
3.2	Kết quả chạy thử nghiệm với kích thước ma trận cỡ 30×30 , số lượng quan sát lớn chiếm từ 68% đến 80%.	39
3.3	Kết quả chạy thử nghiệm với kích thước ma trận cỡ 50×50 , số lượng quan sát được chiếm từ 20% đến 80%	40
3.4	Kết quả về sai số và thời gian của hai thuật toán điểm trong và thuật toán FISTA	41
3.5	Logo MIT.	42
3.6	Ví dụ cho việc phục hồi ảnh với tỉ lệ số điểm ảnh đã biết chiếm 5%.	42
3.7	Ví dụ cho việc phục hồi ảnh với tỉ lệ số điểm ảnh đã biết chiếm 10%.	43
3.8	Ví dụ cho việc phục hồi ảnh với tỉ lệ số điểm ảnh đã biết chiếm 20%.	43
3.9	Dữ liệu đầu vào của bài toán Netflix.	45
3.10	Đọc kết quả dữ liệu Netflix với người dùng có ID là "1".	45

Mục lục

Lời cam đoan	i
Lời cảm ơn	ii
Danh mục các hình vẽ	iv
Mục lục	v
Mở đầu	1
1 KIẾN THỨC CHUẨN BỊ	3
1.1 Khai triển SVD	3
1.2 Một số chuẩn ma trận	7
1.3 Hàm liên hợp	12
2 BÀI TOÁN CỰC TIỂU CHUẨN NGUYÊN TỬ	16
2.1 Mô hình thực tế và bài toán cực tiểu hàm hạng ma trận . . .	16
2.2 Bao lồi của hàm hạng ma trận	20
2.3 Điều kiện RIP	24
3 THUẬT TOÁN TỐI ƯU CHO BÀI TOÁN CỰC TIỂU CHUẨN NGUYÊN TỬ	29
3.1 Đưa bài toán cực tiểu chuẩn nguyên tử về dạng quy hoạch nửa xác định dương	30
3.2 Thuật toán proximal gradient	33
3.2.1 Toán tử Proximal	33
3.2.2 Thuật toán Proximal	34
3.3 Thử nghiệm số	37
3.3.1 Thử nghiệm với ma trận kích thước bé	38
3.3.2 Xử lý ảnh với thuật toán FISTA	41

3.3.3 Bài toán Netflix	43
Kết luận và kiến nghị	48
Tài liệu tham khảo	57

Mở đầu

Chúng ta đã biết nhiều khái niệm trong toán học cũng như trong thực tế như độ phức tạp của bài toán hay số chiều của ma trận có thể được biểu diễn thông qua hạng của một ma trận nào đó. Thông thường, những ma trận này có hạng rất thấp so với số chiều của dữ liệu.

Do đó, bài toán cực tiểu hàm hạng trên một tập lồi là một bài toán tối ưu quan trọng và xuất hiện nhiều trong việc lựa chọn các mô hình trong thực tế, ví dụ như bài toán phục hồi ma trận (matrix completion), bài toán nén ảnh, các bài toán trong hệ gợi ý, ... Trong một số trường hợp đặc biệt, bài toán cực tiểu hàm hạng có thể giải bằng cách sử dụng phân tích giá trị kì dị. Tuy nhiên, trong trường hợp tổng quát, đây là một bài toán tối ưu không lồi và yêu cầu thời gian tính toán mũ.

Một trong những phương pháp hiệu quả nhất để giải bài toán cực tiểu hàm hạng là thông qua việc giải bài toán cực tiểu chuẩn nguyên tử. Chuẩn nguyên tử của một ma trận chữ nhật được định nghĩa bằng tổng của các giá trị kì dị của ma trận đó. Chuẩn nguyên tử đã được chứng minh là bao lồi của hàm hạng trên hình cầu đơn vị theo chuẩn phổ và có thể được cực tiểu hóa bằng cách sử dụng các thuật toán tối ưu. Phương pháp này đã được đưa ra bởi Fazel [1] như là một phương pháp heuristic hiệu quả để tìm nghiệm của bài toán cực tiểu hàm hạng. Đối với các ma trận đường chéo, chuẩn nguyên tử chính bằng tổng trị tuyệt đối (chuẩn l_1) của các giá trị trên đường chéo. Cực tiểu hóa chuẩn l_1 là một phương pháp heuristic nổi tiếng và đã được sử dụng từ những năm 1970 bởi các nhà địa lí trong khi nghiên cứu các hoạt động địa chấn. Kể từ đó đến nay, phương pháp này đã được sử dụng trong nhiều lĩnh vực như khử nhiễu hình ảnh [2], lựa

chọn mô hình trong thống kê [3], xấp xỉ thưa và tối ưu hóa danh mục đầu tư [4],...

Trong luận văn này, chúng tôi sẽ nghiên cứu bài toán cực tiểu chuẩn nguyên tử và các điều kiện giới hạn isometry (RIP) để phương pháp này cho ta nghiệm của bài toán cực tiểu hàm hạng tương ứng. Đồng thời, chúng tôi cũng quan tâm đến các thuật toán tối ưu: phương pháp điểm trong, phương pháp proximal gradient và phiên bản tăng tốc của nó để giải quyết các bài toán cực tiểu chuẩn nguyên tử. Cụ thể:

- Chương 1: chúng tôi sẽ trình bày các kiến thức chuẩn bị về phân tích giá trị kì dị của ma trận, các tính chất quan trọng của ma trận và chuẩn ma trận, kiến thức về hàm liên hợp để phục vụ cho phát biểu các định lý chính.
- Chương 2: chúng tôi trình bày bài toán cực tiểu hàm hạng dựa trên mô hình thực tế, từ đó đưa bài toán cực tiểu hàm hạng về bài toán cực tiểu chuẩn nguyên tử, trình bày định lý về chuẩn nguyên tử là bao lồi của hàm hạng trên hình cầu đơn vị theo chuẩn phổ và các điều kiện RIP đảm bảo mối liên hệ của bài toán cực tiểu chuẩn nguyên tử và bài toán cực tiểu hàm hạng tương ứng.
- Chương 3: chúng tôi trình bày về hai phương pháp giải bài toán cực tiểu chuẩn nguyên tử và thử nghiệm số để so sánh hai phương pháp và ứng dụng bài toán chuẩn nguyên tử trong các bài toán thực tế: xử lý ảnh và bài toán Netflix.

Chúng tôi cũng nói về các khía cạnh phát triển của luận văn, và các phương hướng để có thể áp dụng lý thuyết này vào giải quyết các bài toán thực tế của doanh nghiệp.

CHƯƠNG 1

KIẾN THỨC CHUẨN BỊ

Trong chương này, chúng tôi sẽ nhắc lại về phương pháp phân tích giá trị kì dị của một ma trận (singular value decomposition) thường được gọi tắt là SVD và các định lý quan trọng của SVD ([5]). Tiếp theo, chúng tôi sẽ trình bày về các tính chất quan trọng của ma trận và các chuẩn ma trận cần thiết cho các kết quả chính, đồng thời cũng chỉ ra mối liên hệ giữa lý thuyết chuẩn vector và chuẩn ma trận. Cuối cùng là các kiến thức liên quan về hàm liên hợp.

1.1 Khai triển SVD

Việc phân tích một ma trận ra thành tích của nhiều ma trận đặc biệt khác có ứng dụng trong rất nhiều lĩnh vực: giảm số chiều dữ liệu, nén dữ liệu, tìm hiểu các đặc tính của dữ liệu, giải phương trình tuyến tính, phân cụm và nhiều ứng dụng khác. Ở đây, ta sẽ nhắc lại một trong những phương pháp phân tích ma trận rất đẹp của Đại số tuyến tính: phân tích giá trị kì dị. Ta có thể thấy, mọi ma trận, không nhất thiết là vuông, đều có thể được phân tích thành tích của ba ma trận đặc biệt.

Định nghĩa 1.1. ([5]) Cho ma trận $A \in \mathbb{R}^{m \times n}$. Khi đó, một phân tích giá trị kì dị của A được xác định bởi:

$$A = U\Sigma V^T,$$

trong đó

- $U \in \mathbb{R}^{m \times m}$ là ma trận trực giao,
- $V \in \mathbb{R}^{n \times n}$ là ma trận trực giao,
- $\Sigma \in \mathbb{R}^{m \times n}$ là ma trận đường chéo.

với các phần tử trên ma trận đường chéo σ_j của Σ là không âm và được sắp xếp theo thứ tự không tăng, nghĩa là $(\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0)$ với $p = \min(m, n)$.

Định lý 1.2. ([5]) Cho ma trận $A \in \mathbb{R}^{m \times n}$. Khi đó, luôn tồn tại khai triển SVD của ma trận A .

Chứng minh. Đặt $\sigma_1 = \|A\|_2 = \sup_{v \in \mathbb{R}^n, \|v\|_2=1} \|Av\|_2$, trong đó kí hiệu $\|x\|_2$ với $x \in \mathbb{R}^k$ là chuẩn Euclid của x trong \mathbb{R}^k . Theo định nghĩa supremum, tồn tại dãy $\{x_n\} \subseteq \mathbb{R}^n$ sao cho $\|x_n\|_2 = 1$ và $\lim_{n \rightarrow \infty} \|Ax_n\|_2 = \|A\|_2$.

Vì $\{x_n\}$ bị chặn, nên tồn tại dãy con $\{x_{n_k}\}_k \subset \{x_n\}_n$ và $v_1 \in \mathbb{R}^n$ sao cho $\lim_{n_k \rightarrow \infty} x_{n_k} = v_1$. Từ tính chất $\|x_n\|_2 = 1$ với mọi $n \in \mathbb{N}$ ta được $\|v_1\|_2 = 1$.

Điều đó dẫn đến

$$\|A\|_2 = \lim_{n \rightarrow \infty} \|Ax_n\|_2 = \lim_{k \rightarrow \infty} \|Ax_{n_k}\|_2 = \|Av_1\|_2.$$

Đặt $u_1 = \frac{Av_1}{\|Av_1\|_2}$. Khi đó $\|u_1\|_2 = \|v_1\|_2 = 1$ và $Av_1 = \sigma_1 u_1$. Bổ sung các vector vào v_1 để tạo ra hệ vector cơ sở trực chuẩn $\{v_1, v_2, \dots, v_n\}$ trong \mathbb{R}^n và bổ sung các vector vào u_1 để tạo ra hệ cơ sở trực chuẩn $\{u_1, u_2, \dots, u_m\}$ trong \mathbb{R}^m . Kí hiệu U_1, V_1 là các ma trận trực giao với các cột u_j, v_j tương ứng. Khi đó

$$\begin{aligned} S &= U_1^T A V_1 = \begin{bmatrix} u_1^T \\ \dots \\ u_m^T \end{bmatrix} A \begin{bmatrix} v_1 & | & v_2 & | & \dots & | & v_n \end{bmatrix} \\ &= \begin{bmatrix} u_1^T \\ \dots \\ u_m^T \end{bmatrix} \begin{bmatrix} Av_1 & | & Av_2 & | & \dots & | & Av_n \end{bmatrix} \end{aligned}$$

$$= \begin{bmatrix} \sigma_1 & w^T \\ 0 & B \end{bmatrix},$$

với 0 là vector cột có số chiều là $m - 1$, w^T là vector hàng có số chiều $n - 1$ và B là ma trận cỡ $(m - 1) \times (n - 1)$. Ngoài ra,

$$\left\| \begin{bmatrix} \sigma_1 & w^T \\ 0 & B \end{bmatrix} \begin{bmatrix} \sigma_1 \\ w \end{bmatrix} \right\|_2 \geq \sigma_1^2 + w^T w = \sqrt{\sigma_1^2 + w^T w} \left\| \begin{bmatrix} \sigma_1 \\ w \end{bmatrix} \right\|_2$$

Điều đó suy ra

$$\|S\|_2 \geq \sqrt{\sigma_1^2 + w^T w}.$$

Vì U_1, V_1 là ma trận trực giao, nên $\|S\|_2 = \|A\|_2 \geq \sqrt{\sigma_1^2 + w^T w}$. Do $\|A\| = \sigma_1$ nên $w = 0$.

Chứng minh định lý bằng quy nạp theo số chiều của ma trận:

Nếu $n = 1$ hoặc $m = 1$: chứng minh trên.

Trường hợp còn lại: ma trận con B biểu thị tác động của ma trận A lên không gian trực giao với v_1 . Theo giả thuyết quy nạp B có SVD, $B = U_2 \Sigma_2 V_2^T$. Vậy

$$U_1^T A V_1 = \begin{bmatrix} \sigma_1 & w^T \\ 0 & B \end{bmatrix} = \begin{bmatrix} \sigma_1 & w^T \\ 0 & U_2 \Sigma_2 V_2^T \end{bmatrix}.$$

Suy ra

$$\begin{aligned} A &= U_1 \begin{bmatrix} \sigma_1 & w^T \\ 0 & U_2 \Sigma_2 V_2^T \end{bmatrix} V_1^T \\ &= U_1 \begin{bmatrix} 1 & 0 \\ 0 & U_2 \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & V_2 \end{bmatrix}^T V_1^T \end{aligned}$$

là SVD của A vì các ma trận sau đều là ma trận trực giao.

$$U_1, \begin{bmatrix} 1 & 0 \\ 0 & U_2 \end{bmatrix}, V_1, \begin{bmatrix} 1 & 0 \\ 0 & V_2 \end{bmatrix},$$

và ma trận $\begin{bmatrix} \sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}$ là ma trận đường chéo. Định lý được chứng minh. \square

Chú ý: Khai triển SVD của ma trận A là duy nhất ([5]). Các $\sigma_1, \sigma_2, \dots, \sigma_p$ với $p = \min(m, n)$ được gọi là các **giá trị kì dị** của A .

Ví dụ 1.3. Xét ma trận $A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

Một phân tích giá trị kì dị của A được xác định:

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \\ \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix}.$$

Định lý 1.4. ([5]) Cho ma trận $A \in \mathbb{R}^{m \times n}$, $\text{rank}(A)$ là số giá trị kì dị khác 0 của A .

Chứng minh. Ta có, hạng của ma trận đường chéo bằng số phần tử có giá trị khác 0. Mà $A = U\Sigma V^T$, U, V là các ma trận có hạng đầy đủ. Do đó, $\text{rank}(A) = \text{rank}(\Sigma) = r$. \square

Định lý 1.5. ([5]) Các giá trị kì dị khác 0 của A là căn bậc hai của các giá trị riêng khác 0 của $A^T A$ và AA^T (hai ma trận này có cùng các giá trị riêng khác 0).

Chứng minh. Từ SVD của A ,

$$A^T A = (U\Sigma V^T)^T (U\Sigma V^T) = V\Sigma^T U^T U\Sigma V^T = V\Sigma^T \Sigma V^T.$$

Vậy $A^T A$ đồng dạng với $\Sigma^T \Sigma$ và do đó có cùng giá trị riêng. Các giá trị riêng của $\Sigma^T \Sigma$ là $\sigma_1^2, \sigma_2^2, \dots, \sigma_r^2$ và $n - r$ giá trị riêng bằng 0.

Lý luận như trên với AA^T , ta có kết quả tương tự. \square

Hệ quả 1.6. ([5]) Cho $A \in \mathbb{R}^{m \times n}$. Ma trận A có SVD là $U\Sigma V^T$, với $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$ là các ma trận trực giao, $\Sigma \in \mathbb{R}^{m \times n}$ là ma trận

đường chéo. Không mất tính tổng quát, giả sử $m \geq n$, thì $Av_i = \sigma_i u_i$ và $A^T u_i = \sigma_i v_i$ với $1 \leq i \leq n$. Từ hệ quả này, ta có thể thấy

$$\begin{aligned} A^T Av_i &= \sigma_i^2 v_i, \\ AA^T u_i &= \sigma_i^2 u_i. \end{aligned}$$

Điều này cho thấy mối quan hệ mật thiết giữa SVD của ma trận A và hệ riêng của các ma trận đối xứng AA^T và $A^T A$. Trong trường hợp A là ma trận đối xứng nửa xác định dương thì ta có thể thấy các giá trị riêng của A cũng là các giá trị kì dị của nó.

1.2 Một số chuẩn ma trận

Xét trong không gian \mathbb{R}^n . Ta có thể thấy rằng, về mặt topo các chuẩn trên không gian vector hữu hạn chiều đều tương đương nhau, nhưng trên thực tế ta cần phải dùng các chuẩn khác nhau để tính toán với các mục đích khác nhau. Điều này đặc biệt đúng trong bài toán về tối ưu hay machine learning, việc chọn các chuẩn khác nhau sẽ dẫn tới tốc độ hội tụ và quá trình hội tụ khác nhau, do đó nó có tác động rất lớn tới nghiệm của bài toán tối ưu,... Vì vậy, việc hiểu cấu trúc của các loại chuẩn sẽ đưa đến việc thiết kế các mô hình tối ưu phù hợp hơn với thực tế. Ở đây ta sẽ nhắc lại định nghĩa chuẩn và một số chuẩn thường gặp.

Định nghĩa 1.7. Chuẩn $\|\cdot\|$ trên không gian vector \mathbb{R}^n là một hàm $\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}$ thoả mãn các tính chất:

1. $\|x\| \geq 0$ cho bất kì $x \in \mathbb{R}^n$ và $\|x\| = 0$ khi và chỉ khi $x = \mathbf{0}$.
2. $\|\lambda x\| = |\lambda| \cdot \|x\|$ cho bất kì $x \in \mathbb{R}^n$ và $\lambda \in \mathbb{R}$.
3. $\|x + y\| \leq \|x\| + \|y\|$ cho bất kì $x, y \in \mathbb{R}^n$.

Với $p \geq 1$, chuẩn l_p trên \mathbb{R}^n được đưa ra bởi công thức:

$$\|x\|_p = \sqrt[p]{\sum_{i=1}^n |x_i|^p}.$$

Nhận thấy khi $p \rightarrow 0$ thì biểu thức trên trở thành số các phần tử khác 0 của x . Khi $p = 0$ được gọi là giả chuẩn (pseudo-norm) $\|\cdot\|_0$. Nó không phải là chuẩn vì nó không thỏa mãn điều kiện 2 và 3 của chuẩn. Giả-chuẩn này, thường được ký hiệu là $\|x\|_0$, khá quan trọng trong Machine Learning vì trong nhiều bài toán, chúng ta cần có ràng buộc “thưa”, tức là đại đa số các thành phần của x là khác không. Đối với một vector thưa thì việc lưu trữ chúng có thể thực hiện dễ dàng hơn thông qua các thư viện làm việc với ma trận thưa.

Có một vài chuẩn l_p thường được dùng:

1. Khi $p = 1$:

$$\|x\|_1 = \sum_{i=1}^n |x_i|.$$

2. Khi $p = 2$:

$$\|x\|_2 = \sqrt{\sum_{i=1}^n x_i^2}.$$

3. Khi $p = \infty$:

$$\|x\|_\infty = \max_{i=1,2,\dots,n} |x_i|.$$

Cho $X \in \mathbb{R}^{m \times n}$, theo Định lí 1.2, luôn tồn tại một phép phân tích SVD $X = U\Sigma V^T$, trong đó Σ là ma trận đường chéo gồm các $\sigma_i : \sigma_1 \geq \sigma_2 \geq \dots \geq 0$. Ta có thể hiểu SVD tạo ra một ánh xạ từ $\mathbb{R}^{m \times n} \rightarrow \mathcal{O} = \{\sigma = (\sigma_1, \sigma_2, \dots, \sigma_p) \in \mathbb{R}^p : \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0\}$, với $p = \min(m, n)$. Từ đây, ta có thể tổng quát hoá các chuẩn vector lên thành chuẩn các ma trận.

Cho ma trận $X \in \mathbb{R}^{m \times n}$, kí hiệu $\sigma_i(X)$ là giá trị kì dị lớn thứ i của X và bằng căn bậc hai giá trị riêng lớn thứ i của XX^T . Hạng của X là r bằng với số giá trị kì dị khác 0 của ma trận X .

Ta định nghĩa tích vô hướng trên không gian các ma trận $\mathbb{R}^{m \times n}$ như sau:

$$\langle X, Y \rangle := \text{Tr}(X^T Y) = \sum_{i=1}^m \sum_{j=1}^n X_{ij} Y_{ij}.$$

Định nghĩa 1.8. (*Chuẩn ma trận*)

Chuẩn tương ứng với tích vô hướng trên không gian các ma trận $\mathbb{R}^{m \times n}$ được gọi là chuẩn Frobenius (hoặc Hilbert-Schmidt) $\|\cdot\|_F$

$$\|X\|_F := \sqrt{\langle X, X \rangle} = \sqrt{\text{Tr}(X^T X)} = \left(\sum_{i=1}^m \sum_{j=1}^n X_{ij}^2 \right)^{\frac{1}{2}} = \left(\sum_{i=1}^r \sigma_i^2 \right)^{\frac{1}{2}}.$$

Chuẩn toán tử (phổ) của ma trận là bằng giá trị kì dị lớn nhất của nó:

$$\|X\| := \sigma_1(X).$$

Chuẩn nguyên tử của ma trận là bằng tổng các giá trị kì dị của nó:

$$\|X\|_* := \sum_{i=1}^r \sigma_i(X).$$

Vì các giá trị kì dị luôn dương nên chuẩn nguyên tử cũng là chuẩn l_1 của vector các giá trị kì dị. Ta có mối liên hệ bất đẳng thức cho bất kì ma trận X nào có hạng tối đa là r :

$$\|X\| \leq \|X\|_F \leq \|X\|_* \leq \sqrt{r} \|X\|_F \leq r \|X\|. \quad (1.1)$$

Chúng ta nhấn mạnh rằng, một vector x trong \mathbb{R}^n có thể coi một ma trận đường chéo $\text{diag}(x)$ có kích thước n . Do đó, các phép toán thông thường của vector (cộng, nhân với một số, nhân hai vector với nhau theo từng thành phần ...) có thể được phiên dịch một cách toàn vẹn sang cho lớp các ma trận đường chéo. Nói cách khác, ta có thể nhúng lớp các vector vào lớp các ma trận, thông qua các ma trận đường chéo.

Sự khác biệt lớn ở đây, đó là trong trường hợp tổng quát, thì phép nhân của hai ma trận là không giao hoán, khác với trường hợp đặc biệt khi lấy hạn chế trên lớp các ma trận đường chéo thì phép nhân là giao hoán. Nói cách khác, các ma trận có thể coi là sự tương tự không giao hoán của các vector, và khi các ma trận này suy biến trở thành ma trận đường chéo, thì chúng ta thu được lý thuyết cổ điển về vector.

Từ quan điểm của phân tích SVD $X = U\Sigma V^T$, tính không giao hoán của phép nhân ma trận có thể được xem như việc các toán tử trực giao

U và V xoắn các thành phần trên đường chéo với nhau. Lẽ tất nhiên, nếu các toán tử trực giao của phân tích SVD là tầm thường, thì ta thu được lý thuyết giao hoán tương ứng.

Chúng tôi muốn nhấn mạnh rằng, đây là một tư tưởng triết học rất quan trọng của toán học trong thế kỷ 20. Việc đưa một đối tượng giao hoán trở thành một đối tượng không giao hoán thường được gọi một cách văn hoa là Lượng Tử Hóa và trở thành một công cụ rất mạnh của vật lý toán hiện đại và trong quan điểm này người ta mong muốn đưa những công cụ toán học đã phát triển và được hiểu rõ cho trường hợp giao hoán sang cho trường hợp không giao hoán. Do khuôn khổ của luận văn, chúng tôi không đi sâu vào các khía cạnh này, và mong độc giả quan tâm có thể tìm hiểu thêm về suy rộng trực tiếp của lý thuyết ma trận ở đây (trong trường hợp hữu hạn chiều) sang trường hợp vô hạn chiều là lý thuyết không gian L_p không giao hoán [6], lý thuyết đại số Von Neuman cũng như một số tài liệu tham khảo liên quan tới tư tưởng "không giao hoán" hóa toán học trong cuốn sách hình học không giao hoán của Alain Connes [7].

Mặc dù vậy, đối với các độc giả quan tâm, chúng tôi muốn đề cập rằng, đằng sau tất cả, bài toán về tối ưu chuẩn nguyên tử có thể coi là phiên bản không giao hoán của lý thuyết compressed sensing của Candes và Terence Tao [8, 9]. Và không phải ngẫu nhiên, hầu hết các kết quả tương ứng của lý thuyết tối ưu chuẩn nguyên tử và ứng dụng trong hệ gợi ý đều bắt chước lại hầu hết các kết quả trong lý thuyết giao hoán cổ điển.

Qua đây, ta thấy rõ hơn sự liên hệ tương ứng giữa chuẩn vector và chuẩn ma trận ở bảng 1.1.

Định nghĩa 1.9. ([10]) (Chuẩn đối ngẫu) Với bất kì chuẩn $\|\cdot\|$ nào trong không gian tích vô hướng, tồn tại một chuẩn đối ngẫu $\|\cdot\|_d$ được định nghĩa

$$\|X\|_d := \sup \{ \langle X, Y \rangle : Y \in \mathbb{R}^{m \times n}, \|Y\| \leq 1 \}. \quad (1.2)$$

Trong không gian \mathbb{R}^n , chuẩn đối ngẫu của chuẩn l_p (với $1 < p < \infty$) là chuẩn l_q với $\frac{1}{p} + \frac{1}{q} = 1$. Điều này có thể chứng minh dựa vào bất đẳng thức Holder. Tương tự, chuẩn đối ngẫu của chuẩn l_∞ là chuẩn l_1 . Đối với trường hợp vector, chuẩn l_2 là tự đối ngẫu, nên ta cũng có thể phỏng đoán điều

Vector	Ma trận
Giao hoán	Không giao hoán
Cardinality	rank
Chuẩn Euclidean	Chuẩn Frobenius
Chuẩn l_1	Chuẩn nguyên tử
Chuẩn l_∞	Chuẩn toán tử

Bảng 1.1: Mối liên hệ giữa chuẩn vector và chuẩn ma trận

tương tự cũng đúng với trường hợp ma trận, chuẩn đối ngẫu của chuẩn Frobenius của ma trận là chuẩn Frobenius. Điều này có thể được chứng minh bởi các biến đổi thông thường hoặc bất đẳng thức Cauchy-Schwarz, từ

$$\sup \{ \text{Tr}(X'Y) : Y \in \mathbb{R}^{m \times n}, \text{Tr}(Y'Y) \leq 1 \},$$

bằng $\|X\|_F$ nếu $Y = X/\|X\|_F$. Hoàn toàn tương tự với trường hợp giao hoán, đối ngẫu của chuẩn $\|\cdot\|_\infty$ là chuẩn $\|\cdot\|_1$, dưới đây ta sẽ chỉ ra chuẩn đối ngẫu của chuẩn toán tử là chuẩn nguyên tử.

Định lý 1.10. (Định lý Von Neumann [11]) Cho $A, B \in \mathbb{R}^{m \times n}$, $q = \min\{m, n\}$ với $\sigma(A) = \sigma_1(A) \geq \dots \geq \sigma_q(A)$ và $\sigma(B) = \sigma_1(B) \geq \dots \geq \sigma_q(B)$, thì

$$\text{Tr}(AB) \leq \sum_{i=1}^q \sigma_i(A)\sigma_i(B).$$

Định lý 1.11. ([10]) Chuẩn đối ngẫu của chuẩn toán tử $\|\cdot\|$ trên $\mathbb{R}^{m \times n}$ là chuẩn nguyên tử $\|\cdot\|_*$.

Chứng minh. Theo định nghĩa của chuẩn đối ngẫu, với $X \in \mathbb{R}^{m \times n}$,

$$\begin{aligned} \|X\|_d &= \sup \{ \langle X, Y \rangle : Y \in \mathbb{R}^{m \times n}, \|Y\| \leq 1 \} \\ &= \sup \left\{ \frac{\langle X, Y \rangle}{\|Y\|}, \|Y\| \neq 0, Y \in \mathbb{R}^{m \times n} \right\} \\ &= \sup \left\{ \frac{\text{Tr}(X^T Y)}{\|Y\|}, \|Y\| \neq 0, Y \in \mathbb{R}^{m \times n} \right\}. \end{aligned}$$

Theo định lý Von Neuman ta có

$$\text{Tr}(X^T Y) \leq \sum_{i=1}^q \sigma_i(X) \sigma_i(Y),$$

trong đó $q = \min\{m, n\}$ và $\sigma_i(X)$ là các giá trị kì dị của ma trận X , $\sigma_i(Y)$ là các giá trị kì dị của ma trận Y .

Hơn nữa, $\|Y\| = \sigma_1(Y)$, $\sigma_1(Y)$ là giá trị kì dị lớn nhất của Y .

$$\frac{\text{Tr}(X^T Y)}{\|Y\|} \leq \frac{\sum_{i=1}^q \sigma_i(X) \sigma_i(Y)}{\sigma_1(Y)} \leq \sum_{i=1}^q \sigma_i(X) = \|X\|_*.$$

Nếu ta có phân tích SVD của X : $X = U \Sigma V^T$, dấu bằng xảy ra khi $Y = UV^T$. Vậy chuẩn đối ngẫu của chuẩn toán tử $\|\cdot\|$ trên $\mathbb{R}^{m \times n}$ là chuẩn nguyên tử $\|\cdot\|_*$. \square

1.3 Hàm liên hợp

Hàm liên hợp đóng vai trò rất quan trọng trong lý thuyết tối ưu lồi. Trực quan của nó, thay vì một hàm số, ta quan tâm tới lớp các độ dốc của các tiếp tuyến của hàm số đấy, và do đó có thể quy về một hàm được định nghĩa trên không gian liên hợp.

Định nghĩa 1.12. (Hàm liên hợp [12]) Xét hàm $f : \mathbb{R}^n \rightarrow (-\infty, \infty]$. Hàm liên hợp của f là hàm $f^* : \mathbb{R}^n \rightarrow (-\infty, \infty]$ được xác định bởi

$$f^*(y) = \sup_{x \in \mathbb{R}^n} \{\langle x, y \rangle - f(x)\} \quad \text{với } y \in \mathbb{R}^n.$$

Dưới đây là một số ví dụ về hàm liên hợp của một số hàm thông dụng:

Ví dụ 1.13. (Hàm liên hợp của hàm chỉ)

Cho hàm số

$$f = \delta_C = \begin{cases} 0 & \text{nếu } x \in C, \\ \infty & \text{nếu } x \notin C, \end{cases}$$

với $C \subseteq \mathbb{R}^n$ khác rỗng, hàm δ_C được gọi là hàm chỉ của tập hợp C . Khi đó với bất kỳ $y \in \mathbb{R}^n$, ta có

$$\begin{aligned} f^*(y) &= \sup_{x \in \mathbb{R}^n} \{ \langle x, y \rangle - \delta_C(x) \} \\ &= \sup_{x \in C} \{ \langle x, y \rangle \} \\ &= \sigma_C(y). \end{aligned}$$

Ví dụ 1.14. (Hàm liên hợp của chuẩn l_1)

Cho $f : \mathbb{R}^n \rightarrow \mathbb{R}$ được cho bởi $f(x) = \|x\|_1$. Với $y \in \mathbb{R}^n$, ta được

$$\begin{aligned} f^*(y) &= \sup_{x \in \mathbb{R}^n} \{ \langle x, y \rangle - f(x) \} \\ &= \sup_{x \in \mathbb{R}^n} \left\{ \sum_{i=1}^n x_i y_i - \sum_{i=1}^n |x_i| \right\} \\ &= \sup_{x \in \mathbb{R}^n} \sum_{i=1}^n (x_i y_i - |x_i|) \\ &= \begin{cases} 0 & \text{nếu } \|y\|_\infty \leq 1, \\ \infty & \text{còn lại.} \end{cases} \end{aligned}$$

Ví dụ 1.15. Cho $f : \mathbb{R}^n \rightarrow \mathbb{R}$ được cho bởi $f(x) = \|x\|_\infty$.

Với $a \in \mathbb{R}^n$:

$$\begin{aligned} f^*(a) &= \sup_{x \in \mathbb{R}^n} \left\{ \sum_{i=1}^n a_i x_i - \max_i |x_i| \right\} \\ &\leq \sup_{x \in \mathbb{R}^n} \left\{ \sum_{i=1}^n |a_i| |x_i| - \max_i |x_i| \right\} \\ &\leq \sup_i \{ |x_i| \|a\|_1 - \max_i |x_i| \} \\ &\leq \sup \{ \|x\|_\infty (\|a\|_1 - 1) \} \\ &= \begin{cases} 0 & \text{nếu } \|a\|_1 \leq 1, \\ \infty & \text{còn lại.} \end{cases} \end{aligned}$$

Nếu $\|a\|_1 \leq 1$, $\langle 0, a \rangle - \|0\|_\infty = 0$, $f^*(a) \geq 0$ trong trường hợp này.

Nếu $\|a\|_1 > 1$, do đó $\langle x, a \rangle - \|x\|_\infty$ là không bị chặn. Do đó,

$$f^*(a) = \begin{cases} 0 & \text{nếu } \|a\|_1 < 1, \\ \infty & \text{còn lại.} \end{cases}$$

Người ta cũng biết rằng, liên hợp của một hàm f nhận giá trị trên $(-\infty, +\infty]$ luôn là đóng và lồi ([13]).

Định lý 1.16. (Bất đẳng thức Fenchel [12]) Cho $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$. Khi đó với bất kì $x, y \in \mathbb{R}^n$,

$$f(x) + f^*(y) \geq \langle y, x \rangle.$$

Chứng minh. Theo định nghĩa của hàm liên hợp ta có với bất kì $x \in \mathbb{R}^n$ và $y \in \mathbb{R}^n$:

$$f^*(y) \geq \langle y, x \rangle - f(x).$$

Do đó

$$f(x) + f^*(y) \geq \langle y, x \rangle - f(x) + f(x)$$

hay

$$f(x) + f^*(y) \geq \langle y, x \rangle.$$

□

Ta định nghĩa song liên hợp (biconjugate) của hàm $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$

$$f^{**}(x) = (f^*)^* = \sup_{y \in \mathbb{R}^n} \{ \langle y, x \rangle - f^*(y) \}.$$

Mệnh đề 1.17. ([12]) Cho hàm số $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$. Khi đó $f(x) \geq f^{**}(x)$ với mọi $x \in \mathbb{R}^n$.

Chứng minh. Từ định nghĩa hàm liên hợp, ta có với mọi $x, y \in \mathbb{R}^n$,

$$f^*(y) \geq \langle y, x \rangle - f(x).$$

Do đó,

$$f(x) \geq \langle x, y \rangle - f^*(y)$$

nghĩa là,

$$f(x) \geq \sup_{y \in \mathbb{R}^n} \{\langle y, x \rangle - f^*(y)\} = f^{**}(x).$$

Mệnh đề được chứng minh. □

Định nghĩa 1.18. ([12]) Hàm số $f : \mathbb{R}^n \rightarrow [-\infty, +\infty]$ gọi là chính thường nếu nó không chứa giá trị $-\infty$ và tồn tại ít nhất một giá trị $x \in \mathbb{R}$ sao cho $f(x) < \infty$.

Định nghĩa 1.19. ([12]) Hàm số $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ là hàm đóng nếu $\text{epi}(f) = \{(x, y) : f(x) \leq y, x \in \mathbb{R}, y \in \mathbb{R}\}$ là đóng.

Định nghĩa 1.20. ([12]) Hàm số $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ là hàm đóng nếu $\text{epi}(f) = \{(x, y) : f(x) \leq y, x \in \mathbb{R}, y \in \mathbb{R}\}$ là tập lồi.

Định lý 1.21. ([12]) Cho hàm số $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ là một hàm đóng, chính thường và lồi. Khi đó $f \equiv f^{**}$

Định nghĩa 1.22. ([12]) Cho hàm số $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Bao lồi của hàm $f(x)$ là hàm lồi lớn nhất $g : \mathbb{R}^n \rightarrow \mathbb{R}$ thỏa mãn $g(x) \leq f(x)$ với mọi $x \in \mathbb{R}^n$

Mệnh đề 1.23. ([14]) Cho hàm số $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$. Khi đó f^{**} là bao lồi đóng của hàm $f(x)$.

CHƯƠNG 2

BÀI TOÁN CỰC TIỂU CHUẨN

NGUYÊN TỬ

2.1 Mô hình thực tế và bài toán cực tiểu hàm hạng ma trận

Hệ thống gợi ý (Recommender System) ngày càng trở nên vô cùng quan trọng của các sản phẩm điện tử có nhiều người dùng. Trong lĩnh vực hệ gợi ý, người dùng gửi các đánh giá về một tập hợp các sản phẩm trong cơ sở dữ liệu và nhà cung cấp sản phẩm sẽ đưa ra các đề xuất dựa trên sự ưa thích của người dùng. Hệ thống này xuất hiện rất nhiều trong thực tế, ví dụ như: khi bạn xem Youtube, Youtube sẽ tự động chuyển các clip liên quan đến clip bạn đang xem, nó cũng tự gợi ý những clip mà bạn có thể thích; hoặc khi bạn mua hàng trên Shoppe, hệ thống sẽ tự động gợi ý những mặt hàng mà những người đã mua món này sẽ thường mua kèm những món kia hay những mặt hàng có cùng tính chất đặc trưng với món hàng bạn đang xem; hoặc Facebook hiển thị quảng cáo những sản phẩm có liên quan đến từ khoá bạn vừa tìm kiếm,...

Ở đây, ta sẽ xét một ví dụ đặc biệt của bài toán này là vấn đề Netflix nổi tiếng. Netflix đã từng tổ chức một cuộc thi từ năm 2007 đến 2009 mong muốn các nhà khoa học dữ liệu phát triển các thuật toán để tăng khả năng

dự đoán chính xác của các lượt đánh giá phim. Người dùng (tương ứng với hàng của ma trận dữ liệu) sẽ xếp hạng các bộ phim (các cột của ma trận dữ liệu), nhưng thường người dùng chỉ đánh giá rất ít các bộ phim, do đó có rất ít dữ liệu được quan sát và nằm rải rác trên các ma trận này, được minh họa ở Hình 2.1. Và nhà cung cấp Netflix muốn khôi phục ma trận này để có thể giới thiệu cho các người dùng cụ thể các bộ phim mà họ có thể thích.



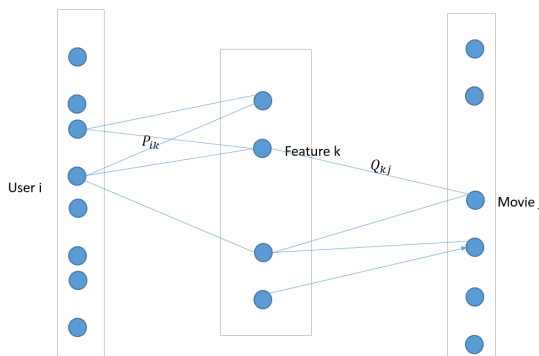
Hình 2.1: Biểu diễn bài toán Netflix dưới dạng ma trận.

<https://www.semanticscholar.org/paper/Low-rank-models-1-Matrix-completion-1-.-1-The/193fac04adbe29f57ee34f6b90ddb74a7d572a86/figure/0>

Theo ngôn ngữ toán học, vấn đề có thể được biểu diễn lại như sau: ta có ma trận dữ liệu $M \in \mathbb{R}^{m \times n}$, và ta chỉ biết một ít phần tử trong ma trận này là M_{ij} , với $(i, j) \in \Omega$, $1 \leq i \leq m$, $1 \leq j \leq n$, trong đó Ω là tập các phần tử quan sát được. Và ta mong muốn biết chính xác các phần tử còn lại trong ma trận này.

Rõ ràng, đây là một vấn đề khó, và không có cách nào có thể dự đoán các phần tử còn lại trong ma trận mà không có bất kỳ giả thuyết nào. Có một giả thiết phổ biến trong lĩnh vực này là giả sử rằng ma trận chưa

biết M có hạng thấp hoặc xấp xỉ hạng thấp. Trong hệ gợi ý, điều này có ý nghĩa bởi vì thông thường, chỉ có một số yếu tố tác động vào sở thích của người dùng. Cụ thể hơn, ta giả sử một người dùng thông thường i với $1 \leq i \leq m$ có các yếu tố đánh giá sở thích theo các tiêu chí đặc tính j với $1 \leq j \leq r$. Sự đánh giá này được thể hiện thông qua một ma trận thể hiện mức độ trọng số $P \in \mathbb{R}^{m \times r}$ nào đó. Tương tự, ta giả sử một bộ phim k với $1 \leq k \leq n$ có các đặc điểm đặc tính j nào đó được thể hiện qua ma trận $Q \in \mathbb{R}^{r \times n}$. Chúng ta chú ý rằng, tất cả các đặc điểm, các ma trận P và Q đều không được biết một cách chính xác, nhưng cách thể hiện ra ngoài thông qua việc đánh giá lại có thể quan sát được thông qua việc phân tích dữ liệu.



Hình 2.2: Quan hệ giữa người dùng, các đặc điểm và bộ phim

Khi đó, ta hi vọng ma trận thể hiện sự quan tâm của người dùng sẽ có dạng $M = P.Q$. Bởi vì rằng, các đặc tính của phim là hữu hạn, và thường thì con người ta chỉ dựa vào một số lượng nhỏ r các yếu tố để quyết định sự yêu thích đối với một đối tượng nào đó, nên ta có r rất nhỏ so với m và n . Khi đó, hiển nhiên $\text{rank}(M) = r$ và do đó ta hi vọng có thể tìm được M trong lớp các ma trận có hạng đủ nhỏ.

Theo [15], các tác giả đã chỉ ra rằng tiên đề này đã thay đổi hoàn toàn vấn đề, làm cho việc đi tìm nghiệm cho bài toán này có ý nghĩa. Ở đây, chúng tôi cũng muốn nhấn mạnh rằng việc khôi phục ma trận hạng thấp từ một mẫu cho trước các phần tử, xuất hiện trong nhiều lĩnh vực khác. Ví dụ, nó xuất hiện trong thị giác máy tính - một lĩnh vực rất nóng hiện tại. Ở đó, nhiều pixel có thể bị thiếu trong kỹ thuật số do tắc hoặc lỗi trong

việc quan sát các chuỗi video. Việc khôi phục một cảnh và suy ra chuyển động của camera từ một chuỗi hình ảnh là một bài toán hoàn thành ma trận được gọi là bài toán structure-from-motion [16, 17],...

Nếu số lượng quan sát đủ lớn và nếu các phần tử trong ma trận được quan sát phân bố đều, người ta có thể hy vọng rằng chỉ có một ma trận hạng thấp với các phần tử này và cũng chính là ma trận cần tìm. Nếu điều này là đúng, người ta sẽ muốn khôi phục ma trận dữ liệu bằng cách giải quyết bài toán tối ưu:

$$\begin{aligned} & \text{minimize} && \text{rank}(X) \\ & \text{thỏa mãn} && X_{ij} = M_{ij} \quad (i, j) \in \Omega, \end{aligned} \tag{2.1}$$

trong đó X là biến quyết định và $\text{rank}(X)$ là hạng của ma trận X . Bài toán (2.1) với một cách tiếp cận thông thường đơn giản là tìm kiếm lời giải thích đơn giản nhất phù hợp với dữ liệu quan sát được. Nếu chỉ có duy nhất một ma trận hạng thấp phù hợp với dữ liệu, việc giải bài toán (2.1) sẽ giúp khôi phục M . Điều này rất tiếc là ít được sử dụng thực tế vì bài toán tối ưu (2.1) là bài toán NP-hard, tất cả các thuật toán đã biết cung cấp lời giải chính xác có độ phức tạp tính toán tăng theo cấp số lũy thừa theo số chiều n của ma trận cả lý thuyết và thực hành [18]. Cho tới thời điểm này, chưa có thuật toán nào được biết đến có khả năng giải quyết bài toán này trong thực tế với $\max(m, n) \geq 10$, theo [19].

Trong [15], Candès và Recht đã chỉ ra rằng việc phục hồi chính xác một ma trận có thể giải quyết được dưới một số điều kiện bằng bài toán quy hoạch lồi, cụ thể hơn là từ bài toán cực tiểu chuẩn nguyên tử. Trong luận văn này, chúng tôi cũng sẽ chỉ ra rằng hàm chuẩn nguyên tử là bao lồi của hàm rank trên hình cầu đơn vị theo chuẩn phổ và chỉ ra điều kiện đảm bảo mối liên hệ của bài toán cực tiểu chuẩn nguyên tử và bài toán cực tiểu hàm hạng tương ứng. Chúng tôi muốn lý giải rằng, lý thuyết tối ưu lồi trên tập hợp lồi đã được phát triển từ lâu và đã được hiểu rõ một phần, cũng như các công cụ tính toán rất mạnh để giải gần đúng bài toán đã được xây dựng. Vì vậy, bằng việc xấp xỉ hàm tối ưu bằng bao lồi của nó, chúng ta hi vọng sẽ tiến gần hơn tới lời giải của bài toán tối thiểu hàm hạng.

Bài toán cực tiểu chuẩn nguyên tử tương ứng với (2.1) được phát biểu như sau:

$$\begin{aligned} & \text{minimize} \quad \|X\|_* \\ & \text{sao cho} \quad X_{ij} = M_{ij} \quad (i, j) \in \Omega. \end{aligned} \tag{2.2}$$

Trong [15] chỉ ra rằng nếu chúng ta quan sát k phần tử được chọn ngẫu nhiên từ ma trận cần khôi phục M là ma trận hạng thấp, nếu số quan sát k thoả mãn điều kiện:

$$k \geq C\ell^{6/5} \cdot r \log(\ell), \tag{2.3}$$

với một hằng số C dương nào đó, thì với xác suất rất cao, hầu hết ma trận $m \times n$ hạng r ($\ell = \max(m, n)$) có thể được khôi phục hoàn hảo bằng cách giải bài toán (2.2).

Dạng tổng quát hoá của bài toán cực tiểu chuẩn nguyên tử được phát biểu như sau:

$$\begin{aligned} & \text{minimize} \quad \|X\|_* \\ & \text{thoả mãn} \quad \mathcal{A}(X) = b, \end{aligned} \tag{2.4}$$

trong đó $X \in \mathbb{R}^{m \times n}$ là biến quyết định, ánh xạ tuyến tính $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ và $b \in \mathbb{R}^p$ được cho trước.

2.2 Bao lồi của hàm hạng ma trận

Chúng ta biết rằng, bao lồi của một hàm số $f : \mathbb{R}^n \rightarrow \mathbb{R}$ là hàm lồi lớn nhất $g : \mathbb{R}^n \rightarrow \mathbb{R}$ thoả mãn $g(x) \leq f(x)$ với mọi $x \in \mathbb{R}^n$.

Lý thuyết tối ưu lồi là lý thuyết đã được phát triển nhiều năm, và đã có rất nhiều công cụ mạnh, nhưng lý thuyết tối ưu không lồi, tối ưu rời rạc cho tới nay vẫn là các bài toán rất khó. Chúng ta thay vì giải trực tiếp bài toán tối ưu không lồi, tối ưu rời rạc thì có thể lựa chọn giải bài toán tối ưu lồi với bao lồi của hàm chi phí để có một nghiệm xấp xỉ đủ tốt, và do đó có thể sử dụng các công cụ khổng lồ của tối ưu lồi.

Từ (1.1), ta có $\text{rank}(X) \geq \frac{\|X\|_*}{\|X\|}$ với mọi $X \neq 0$. Với mọi ma trận X mà $\|X\| \leq 1$, ta có $\text{rank}(X) \geq \|X\|_*$, nên chuẩn nguyên tử là bao lồi của hàm hạng trên hình cầu đơn vị. Điều này được thể hiện trong định lý dưới đây.

Định lý 2.1. ([1]) Bao lồi của hàm hạng trên tập $\{X \in \mathbb{R}^{n \times n} : \|X\| \leq 1\}$ là chuẩn nguyên tử $\|\cdot\|_*$.

Chứng minh.

Phần 1: Hàm liên hợp của hàm hạng: $\text{rank}^*(X)$.

Liên hợp của hàm hạng rank , trên tập các ma trận với chuẩn toán tử ≤ 1 , là:

$$\text{rank}^*(Y) = \sup_{\|X\| \leq 1} (\text{Tr}(Y^T X) - \text{rank}(X)), \quad (2.5)$$

với $\langle Y, X \rangle = \text{Tr}(Y^T X)$ là tích vô hướng trên $\mathbb{R}^{m \times n}$.

Đặt $q = \min\{m, n\}$, theo định lý Von Neumann:

$$\text{Tr}(Y^T X) \leq \sum_{i=1}^q \sigma_i(Y) \sigma_i(X), \quad (2.6)$$

trong đó $\sigma_i(X)$ là giá trị kì dị lớn nhất thứ i của X . Nếu Y là ma trận cho trước, giả sử $X = U_X \Sigma_X V_X^T$ và $Y = U_Y \Sigma_Y V_Y^T$ là hai phân tích SVD tương ứng X và Y , $U_X, U_Y \in \mathbb{R}^{m \times m}$ là các ma trận trực giao, $V_X, V_Y \in \mathbb{R}^{n \times n}$ là các ma trận trực giao; $\Sigma_X = \text{diag}(\sigma_1(X), \dots, \sigma_n(X))$; $\Sigma_Y = \text{diag}(\sigma_1(Y), \dots, \sigma_n(Y))$ đẳng thức trong (2.6) xảy ra nếu U_X và V_X lần lượt bằng U_Y và V_Y .

Ta thấy rằng $\text{rank}(X)$ ở (2.5) là độc lập với U_X và V_X . Đặt $U_X = U_Y$ và $V_X = V_Y$, ta được

$$\text{rank}^*(Y) = \sup_{\|X\| \leq 1} \left(\sum_{i=1}^q \sigma_i(Y) \sigma_i(X) - \text{rank } X \right).$$

Nếu $X = 0$, ta có $\text{rank}^*(Y) = 0$ với mọi ma trận Y . Nếu $\text{rank}(X) = r$, $1 \leq r \leq q$ thì $\text{rank}^*(Y) = \sum_{i=1}^r \sigma_i(Y) - r$. Do đó, $\text{rank}^*(Y)$ có thể được biểu diễn

$$\text{rank}^*(Y) = \max \left\{ 0, \sigma_1(Y) - 1, \dots, \sum_{i=1}^r \sigma_i(Y) - r, \dots, \sum_{i=1}^q \sigma_i(Y) - q \right\}.$$

Nhận thấy rằng, thành phần lớn nhất trên tập này là tổng các phần tử $(\sigma_i(Y) - 1)$ dương. Ta được

$$\begin{aligned} \text{rank}^*(Y) &= \begin{cases} 0 & \text{nếu } \|Y\| \leq 1, \\ \sum_{i=1}^r \sigma_i(Y) - r & \text{nếu } \sigma_r(Y) > 1 \text{ và } \sigma_{r+1}(Y) \leq 1, \end{cases} \quad (2.7) \\ &= \sum_{i=1}^q (\sigma_i(Y) - 1)_+, \end{aligned}$$

trong đó a_+ là kí hiệu các phần dương của a .

Phân 2: Tính $\text{rank}^{**}(X)$.

Hàm liên hợp của rank^* được xác định như sau

$$\text{rank}^{**}(Z) = \sup_Y (\text{Tr } Z^T Y - \text{rank}^*(Y)),$$

với mọi $Z \in \mathbb{R}^{m \times n}$

Xét $Y = U_Y \Sigma_Y V_Y^T; Z = U_Z \Sigma_Z V_Z^T$ lần lượt là các SVD của Y và Z . $U_Y, U_Z \in \mathbb{R}^{m \times m}$ là các ma trận trực giao, $V_Y, V_Z \in \mathbb{R}^{n \times n}$ là các ma trận trực giao; $\Sigma_Y = \text{diag}(\sigma_1(Y), \dots, \sigma_n(Y)); \Sigma_Z = \text{diag}(\sigma_1(Z), \dots, \sigma_n(Z))$

Trước hết, ta chọn $U_Y = U_Z$ và $V_Y = V_Z$ để nhận được

$$\text{rank}^{**}(Z) = \sup_Y \left(\sum_{i=1}^q \sigma_i(Z) \sigma_i(Y) - \text{rank}^*(Y) \right).$$

Ta sẽ xét 2 trường hợp $\|Z\| > 1$ và $\|Z\| \leq 1$:

- Nếu $\|Z\| > 1$, ta có thể chọn $\sigma_1(Y)$ đủ lớn sao cho $\text{rank}^{**}(Z) \rightarrow \infty$.
Thật vậy,

$$\text{rank}^{**}(Z) = \sup_Y \left(\sum_{i=1}^q \sigma_i(Z) \sigma_i(Y) - \left(\sum_{i=1}^r \sigma_i(Y) - r \right) \right),$$

hệ số của $\sigma_1(Y)$ là $(\sigma_1(Z) - 1) > 0$.

- Nếu $\|Z\| \leq 1$. Nếu $\|Y\| \leq 1$, thì $\text{rank}^*(Y) = 0$ và cực đại đạt được với $\sigma_i(Y) = 1, i = 1, \dots, q$, dẫn đến

$$\text{rank}^{**}(Z) = \sum_{i=1}^q \sigma_i(Z) = \|Z\|_*.$$

Ta chỉ ra rằng nếu $\|Y\| > 1$, điểm cực đại luôn nhỏ hơn giá trị cho ở trên. Thật vậy, bằng cách cộng và trừ số hạng $\sum_{i=1}^q \sigma_i(Z)$ và sắp xếp lại các số hạng, ta nhận được

$$\begin{aligned} & \sum_{i=1}^q \sigma_i(Y)\sigma_i(Z) - \sum_{i=1}^r (\sigma_i(Y) - 1) \\ &= \sum_{i=1}^q \sigma_i(Y)\sigma_i(Z) - \sum_{i=1}^r (\sigma_i(Y) - 1) - \sum_{i=1}^q \sigma_i(Z) + \sum_{i=1}^q \sigma_i(Z) \\ &= \sum_{i=1}^r (\sigma_i(Y) - 1)(\sigma_i(Z) - 1) + \sum_{i=r+1}^q (\sigma_i(Y) - 1)\sigma_i(Z) + \sum_{i=1}^q \sigma_i(Z) \\ &< \sum_{i=1}^q \sigma_i(Z). \end{aligned}$$

Ta có bất đẳng thức cuối bởi vì hai tổng đầu trên dòng thứ 3 luôn có giá trị âm.

Các trường hợp đều dẫn đến

$$\text{rank}^{**}(Z) = \|Z\|_*$$

trên tập $\{Z \in R^n \mid \|Z\| \leq 1\}$. Do đó, trên tập này, $\|Z\|_*$ là bao lồi của hàm $\text{rank}(Z)$. \square

Cho X_0 là nghiệm của bài toán cực tiểu hàm hạng $\text{rank}(X) = b$. Đặt $M = \|X_0\|$. Bao lồi của hàm hạng trên tập $\mathcal{C} = \{X \in \mathbb{R}^{m \times n} : \|X\| \leq M\}$ là $\|X\|_*/M$. Gọi X_* là nghiệm của bài toán cực tiểu chuẩn nguyên tử với ràng buộc $\mathcal{A}(X) = b$. Khi đó ta có

$$\|X_*\|_*/M \leq \text{rank } X_0 \leq \text{rank}(X_*),$$

là chặn trên và chặn dưới của bài toán tối ưu hàm hạng khi chuẩn của nghiệm tối ưu đã biết. Hơn nữa, đây là giới hạn dưới chặt chẽ nhất trong số tất cả các giới hạn dưới lồi của hàm hạng trên tập \mathcal{C} .

Trong không gian vector, ta có một bất đẳng thức tương tự. Cho $\text{card}(x)$ là hàm lực lượng của x đếm số phần tử khác không trong vector x ; khi đó ta có $\text{card}(x) \geq \|x\|_1/\|x\|_\infty$. Không có gì ngạc nhiên khi chuẩn l_1 cũng là bao lồi của hàm lực lượng trên tập $\{x \in \mathbb{R}^n : \|x\|_\infty \leq 1\}$. Kết quả này có thể xem như là trường hợp đặc biệt của định lý trên.

2.3 Điều kiện RIP

Cho $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ là ánh xạ tuyến tính và X_0 là ma trận có hạng r . Đặt $b := \mathcal{A}(X_0)$ và định nghĩa

$$X^* := \operatorname{argmin}_X \|X\|_* \quad \text{sao cho } \mathcal{A}(X) = b. \quad (2.8)$$

Trong phần này, ta sẽ chỉ ra điều kiện để đảm bảo rằng $X^* = X_0$. Các điều kiện sẽ được xác định bởi các giá trị của một chuỗi tham số δ_r lượng hóa hành vi của ánh xạ tuyến tính \mathcal{A} khi bị giới hạn trong tập các ma trận có hạng r . Định nghĩa dưới đây sẽ tổng quát hoá về định nghĩa của tham số "restricted isometry property" từ các vector lên ma trận.

Định nghĩa 2.2. ([10]) Cho $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ là ánh xạ tuyến tính. Không mất tính tổng quát, giả sử $m \leq n$. Với mọi số nguyên r với $1 \leq r \leq m$, định nghĩa hằng số giới hạn isometry (RIP) là số nhỏ nhất $\delta_r(\mathcal{A})$ sao cho

$$(1 - \delta_r(\mathcal{A}))\|X\|_F \leq \|\mathcal{A}(X)\| \leq (1 + \delta_r(\mathcal{A}))\|X\|_F, \quad (2.9)$$

với mọi ma trận X có hạng lớn nhất là r .

Từ định nghĩa, ta có nhận xét $\delta_r(\mathcal{A}) \leq \delta_{r'}(\mathcal{A})$ với mọi $r \leq r'$.

RIP cho các vector thưa đã được phát triển bởi Candès và Tao [8]. Định nghĩa 2.2 đã phát triển khái niệm cho các ma trận, với chuẩn Euclid được thay thế bởi chuẩn Frobenius và lực lượng của vector được thay thế bởi hạng. Do đó, với các ma trận đường chéo, chuẩn Frobenius bằng chuẩn Euclid của đường chéo, định nghĩa này đưa về với định nghĩa RIP ban đầu của [8] trong trường hợp ma trận đường chéo.

Hai định lý dưới đây sẽ chỉ ra điều kiện đảm bảo mối liên hệ của bài toán cực tiểu chuẩn nguyên tử và bài toán cực tiểu hàm hạng tương ứng. Cả hai định lý là sự tổng quát hoá từ trường hợp vector thưa lên trường hợp ma trận hạng thấp.

Định lý 2.3. ([10]) Giả sử rằng $\delta_{2r} < 1$ với số nguyên $r \geq 1$ thì X_0 là ma trận duy nhất có hạng tối đa là r thoả mãn $\mathcal{A}(X) = b$.

Chứng minh. Ta sử dụng phương pháp phản chứng, giả sử tồn tại ma trận X có hạng r thoả mãn $\mathcal{A}(X) = b$. Đặt $Z := X_0 - X$ là ma trận khác không, có hạng tối đa là $2r$.

Ta có

$\mathcal{A}(Z) = \mathcal{A}(X_0 - X) = \mathcal{A}(X_0) - \mathcal{A}(X) = 0$ (do \mathcal{A} là ánh xạ tuyến tính). Theo định nghĩa, $(1 - \delta_{2r})\|Z\|_F \leq \|\mathcal{A}(Z)\|$. Tuy nhiên, $(1 - \delta_{2r})\|Z\|_F > 0$ vì $\delta_{2r} < 1$. Từ đó dẫn đến vô lý. \square

Mệnh đề 2.4. ([10]) Cho A và B là các ma trận có cùng số chiều. Nếu $AB^T = 0$ và $A^T B = 0$, thì $\|A + B\|_* = \|A\|_* + \|B\|_*$.

Chứng minh. Giả sử $A, B \in \mathbb{R}^{m \times n}$. Xét phân tích SVD rút gọn của A và B :

$$A = U_A \Sigma_A V_A^T, \quad B = U_B \Sigma_B V_B^T,$$

với $U_A \in \mathbb{R}^{m \times r_1}$, $V_A \in \mathbb{R}^{n \times r_1}$, $\Sigma_A \in \mathbb{R}^{r_1 \times r_1}$;

$U_B \in \mathbb{R}^{m \times r_2}$, $V_B \in \mathbb{R}^{n \times r_2}$, $\Sigma_B \in \mathbb{R}^{r_2 \times r_2}$.

Ta có:

$$AB^T = U_A \Sigma_A V_A^T V_B \Sigma_B^T U_B^T = 0.$$

Vì U_A và U_B là ma trận khả nghịch trái nên $V_A^T V_B = 0$.

Tương tự, $A^T B = 0$ thì $U_A^T U_B = 0$.

Vì $U_A^T U_B = 0$ nên $\begin{bmatrix} U_A & U_B \end{bmatrix}$ là ma trận trực giao; tương tự $\begin{bmatrix} V_A & V_B \end{bmatrix}$. Do đó, ta có một phân tích giá trị kỳ dị hợp lệ cho $A + B$:

$$A + B = \begin{bmatrix} U_A & U_B \end{bmatrix} \begin{bmatrix} \Sigma_A & \\ & \Sigma_B \end{bmatrix} \begin{bmatrix} V_A & V_B \end{bmatrix}^T.$$

Ta đã chứng minh được $\|A + B\|_* = \|A\|_* + \|B\|_*$. \square

Ta xét bổ đề sau.

Bổ đề 2.5. Cho A và B là các ma trận có cùng số chiều. Khi đó, tồn tại các ma trận B_1 và B_2 sao cho:

1. $B = B_1 + B_2$.

2. $\text{rank } B_1 \leq 2 \text{rank } A$.
3. $AB_2^T = 0$ và $A^T B_2 = 0$.
4. $\langle B_1, B_2 \rangle = 0$.

Chứng minh. Xét phân tích giá trị kỳ dị đầy đủ của $A_{m \times n}$

$$A = U \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} V^T.$$

Đặt $\hat{B} = U^T B V$

$$\hat{B} = \begin{bmatrix} \hat{B}_{11} & \hat{B}_{12} \\ \hat{B}_{21} & \hat{B}_{22} \end{bmatrix}.$$

Hai ma trận B_1 và B_2 được định nghĩa bởi

$$B_1 = U \begin{bmatrix} \hat{B}_{11} & \hat{B}_{12} \\ \hat{B}_{21} & 0 \end{bmatrix} V^T \text{ và } B_2 = U \begin{bmatrix} 0 & 0 \\ 0 & \hat{B}_{22} \end{bmatrix} V^T$$

Ta dễ dàng có thể thấy rằng B_1 và B_2 thoả mãn các điều kiện từ (1),(3) và (4). Điều kiện (2) được thoả mãn, bởi vì hạng của ma trận B_1 không vượt quá tổng hạng của $\begin{bmatrix} \hat{B}_{11} & 0 \\ \hat{B}_{21} & 0 \end{bmatrix}$ và $\begin{bmatrix} 0 & \hat{B}_{12} \\ 0 & 0 \end{bmatrix}$. Trong đó hạng của mỗi ma trận trên không vượt quá $\text{rank}(A)$. \square

Định lý 2.6. ([10]) *Giả sử rằng $r \geq 1$ sao cho $\delta_{5r} < 1/10$. Khi đó $X^* = X_0$.*

Chứng minh. Từ tính tối ưu của X^* , ta có $\|X_0\|_* \geq \|X^*\|_*$.

Đặt $R := X^* - X_0$. Áp dụng Bổ đề 2.5 với ma trận X_0 và R , tồn tại các ma trận R_0 và R_c sao cho $R = R_0 + R_c$, $\text{rank } R_0 \leq 2 \text{rank } X_0$, và $X_0 R_c^T = 0$ và $X_0^T R_c = 0$. Do đó,

$$\begin{aligned} \|X_0\|_* &\geq \|X_0 + R\|_* = \|X_0 + R_0 + R_c\|_* \geq \|X_0 + R_c\|_* - \|R_0\|_* \\ &= \|X_0\|_* + \|R_c\|_* - \|R_0\|_*, \end{aligned} \quad (2.10)$$

trong đó bất đẳng thức thứ ba là do bất đẳng thức trong tam giác, bất đẳng thức cuối là theo Mệnh đề 2.3.

Như vậy,

$$\|R_0\|_* \geq \|R_c\|_*. \quad (2.11)$$

Tiếp theo, ta chia R_c thành các ma trận R_1, R_2, \dots (các ma trận này có hạng tối đa là $3r$).

Đặt $R_c = U \text{diag}(\sigma) V^T$ là SVD của R_c . Cho mỗi $i \geq 1$, xác định tập chỉ số $I_i = \{3r(i-1) + 1, \dots, 3ri\}$ và đặt $R_i := U_{I_i} \text{diag}(\sigma_{I_i}) V_{I_i}^T$ (chú ý rằng $\langle R_i, R_j \rangle = 0$ nếu $i \neq j$). Từ cấu trúc như vậy, ta có

$$\sigma_k \leq \frac{1}{3r} \sum_{j \in I_i} \sigma_j \quad \forall k \in I_{i+1},$$

bao gồm $\|R_{i+1}\|_F^2 \leq \frac{1}{3r} \|R_i\|_*^2$. Ta có thể tính

$$\sum_{j \geq 2} \|R_j\|_F \leq \frac{1}{\sqrt{3r}} \sum_{j \geq 1} \|R_j\|_* = \frac{1}{\sqrt{3r}} \|R_c\|_* \leq \frac{1}{\sqrt{3r}} \|R_0\|_* \leq \frac{\sqrt{2r}}{\sqrt{3r}} \|R_0\|_F, \quad (2.12)$$

với đẳng thức cuối từ (1.1) ($\|R_0\|_* \leq \|R_0\|_F$) và sự thật rằng $\text{rank } R_0 \leq 2r$. Cuối cùng, chú ý rằng, hạng của $R_0 + R_1$ tối đa là $5r$, nên ta có thể bỏ chúng cùng nhau như là

$$\begin{aligned} \|\mathcal{A}(R)\| &\geq \|\mathcal{A}(R_0 + R_1)\| - \sum_{j \geq 2} \|\mathcal{A}(R_j)\| \\ &\geq (1 - \delta_{5r}) \|R_0 + R_1\|_F - (1 + \delta_{3r}) \sum_{j \geq 2} \|R_j\|_F \\ &\geq \left((1 - \delta_{5r}) - \sqrt{\frac{2}{3}} (1 + \delta_{3r}) \right) \|R_0\|_F \\ &\geq \left((1 - \delta_{5r}) - \frac{9}{11} (1 + \delta_{3r}) \right) \|R_0\|_F. \end{aligned} \quad (2.13)$$

Từ giả thuyết $\mathcal{A}(R) = \mathcal{A}(X^* - X_0) = 0$ nên nếu các thành phần ở bên phải tay phải là dương, $R_0 = 0$, điều này suy ra $R_c = 0$ từ (2.11), và do đó $X^* = X_0$. Ta có thể thấy vế phải dương khi $9\delta_{3r} + 11\delta_{5r} < 2$. Từ $\delta_{3r} \leq \delta_{5r}$, ta có $X^* = X_0$ nếu $\delta_{5r} < \frac{1}{10}$.

□

Số tỉ lệ $\frac{9}{11}$ ở trong chứng minh định lý được chọn để dễ tính toán, và rõ ràng nó không tối ưu. Có thể đạt được ràng buộc chặt hơn khi làm việc trực tiếp với dòng thứ hai đến dòng cuối cùng trong (2.13). Điểm quan trọng nhất là δ_{5r} bị chặn trên bởi một hằng số độc lập với m, n, r và p .

CHƯƠNG 3

THUẬT TOÁN TỐI ƯU CHO BÀI TOÁN CỰC TIỂU CHUẨN NGUYÊN TỬ

Trong chương này chúng tôi sẽ trình bày hai phương pháp giải bài toán cực tiểu chuẩn nguyên tử. Ở phần 3.1, phương pháp thứ nhất là đưa bài toán cực tiểu chuẩn nguyên tử về bài toán quy hoạch nửa xác định dương, và sử dụng phương pháp điểm trong để giải bài toán này. Đây là một phương pháp hiệu quả cho các bài toán kích thước nhỏ, cho ta các kết quả có độ chính xác rất cao. Tuy nhiên, độ phức tạp của thuật toán lớn, nó chỉ có thể xử lý tốt khi dữ liệu có số chiều khoảng 50 ([10]).

Vì vậy, trong phần 3.2, chúng tôi sẽ giới thiệu một phương pháp đưa bài toán về dạng bài toán tối ưu không ràng buộc và sử dụng thuật toán proximal gradient và phiên bản tăng tốc của nó (FISTA). Phương pháp này có thể giải quyết các bài toán thực tế có kích thước lớn. Các kết quả thử nghiệm của hai phương pháp trên được trình bày trong phần 3.3. Có rất nhiều phương pháp tối ưu khác có thể giải hiệu quả cho bài toán chuẩn nguyên tử trên không gian của các ma trận, tuy nhiên chúng tôi chưa có đủ thời gian để nghiên cứu, cũng như không gian để trình bày hết vào luận văn này.

3.1 Đưa bài toán cực tiểu chuẩn nguyên tử về dạng quy hoạch nửa xác định dương

Đối với các bài toán tương đối nhỏ đòi hỏi độ chính xác cao, phương pháp điểm trong cho bài toán quy hoạch nửa xác định dương có thể được áp dụng giải trực tiếp cho bài toán cực tiểu chuẩn nguyên tử. Từ bài toán cực tiểu chuẩn nguyên tử, ta có thể dễ dàng đưa về bài toán quy hoạch nửa xác định dương như dưới đây.

Theo Định lí 1.11, chuẩn đối ngẫu của chuẩn toán tử $\|\cdot\|$ trên $\mathbb{R}^{m \times n}$ là chuẩn nguyên tử $\|\cdot\|_*$, nên

$$\|X\|_* = \sup\{\text{Tr}(X^T Y) : Y \in \mathbb{R}^{m \times n}, \|Y\| \leq 1\},$$

hay $\|X\|_*$ là nghiệm của bài toán

$$\begin{aligned} & \text{maximize} && \text{Tr}(X^T Y) \\ & \text{thỏa mãn} && \|Y\| \leq 1. \end{aligned} \quad (3.1)$$

Ta có:

$$\begin{aligned} \|Y\| \leq 1 &\Leftrightarrow \sigma_1(Y) \leq 1 \quad (\text{với } \sigma_1(Y) \text{ là giá trị kì dị lớn nhất của } Y) \\ &\Leftrightarrow \lambda_1(Y Y^T) \leq 1 \quad (\text{với } \lambda_1(Y Y^T) \text{ là giá trị riêng lớn nhất của } Y Y^T) \\ &\Leftrightarrow \begin{pmatrix} 1 - \lambda_1 & 0 & \dots & 0 \\ 0 & 1 - \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 - \lambda_n \end{pmatrix} \succcurlyeq 0, \end{aligned} \quad (3.2)$$

với $\lambda_1, \lambda_2, \dots, \lambda_n$ là các giá trị riêng của $Y Y^T$. Vì $Y Y^T$ là ma trận đối xứng, nên có thể phân tích:

$$Y Y^T = P \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{pmatrix} P^T, \text{ trong đó } P \text{ là ma trận trực giao.}$$

Cho $v \in \mathbb{R}^n$ bất kì, ta có 3.2 tương đương

$$\begin{aligned}
& vP \begin{pmatrix} 1 - \lambda_1 & 0 & \dots & 0 \\ 0 & 1 - \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 - \lambda_n \end{pmatrix} P^T v^T \succcurlyeq 0 \\
& \Leftrightarrow v(I - YY^T)v^T \succcurlyeq 0 \\
& \Leftrightarrow I - YY^T \succcurlyeq 0.
\end{aligned} \tag{3.3}$$

Theo Bổ đề Schur, (3.3) tương đương với $\begin{bmatrix} I_m & Y \\ Y^T & I_n \end{bmatrix} \succcurlyeq 0$, hay

$$\|Y\| \leq 1 \Leftrightarrow \begin{bmatrix} I_m & Y \\ Y^T & I_n \end{bmatrix} \succcurlyeq 0. \tag{3.4}$$

Từ đây ta có bài toán nửa xác định dương của 3.1:

$$\begin{aligned}
& \text{maximize} && \text{Tr}(X^T Y) \\
& \text{thỏa mãn} && \begin{bmatrix} I_m & Y \\ Y^T & I_n \end{bmatrix} \succeq 0.
\end{aligned} \tag{3.5}$$

Ta có thể viết lại dưới dạng

$$\text{minimize} \quad -\text{Tr}(X^T Y) \tag{3.6}$$

$$\text{thỏa mãn} \quad -\begin{bmatrix} I_m & Y \\ Y^T & I_n \end{bmatrix} \preceq 0. \tag{3.7}$$

Xét hàm Lagrange của bài toán 3.6, với $\begin{bmatrix} W_{11} & W_{12} \\ W_{12}^T & W_{22} \end{bmatrix} \succeq 0$ là hệ số Lagrange:

$$\begin{aligned}
L(Y, W) &= -\langle X, Y \rangle - \left\langle \begin{bmatrix} I_m & Y \\ Y^T & I_n \end{bmatrix}, \begin{bmatrix} W_{11} & W_{12} \\ W_{12}^T & W_{22} \end{bmatrix} \right\rangle \\
&= -\langle X, Y \rangle + \langle -I_m, W_{11} \rangle + \langle -Y, W_{12} \rangle + \langle -Y^T, W_{12}^T \rangle + \langle -I_n, W_{22} \rangle.
\end{aligned}$$

Hàm đối ngẫu Lagrange:

$$g(W) = \inf_Y L(Y, W).$$

Khi đó

$$\nabla_Y L(Y, W) = -X - W_{12} - W_{12} = 0 \rightarrow X = -2W_{12}.$$

$$\text{Nên } g(W) = \begin{cases} -\text{Tr}(W_{11}) - \text{Tr}(W_{22}) & \text{nếu } X = -2W_{12}, \\ -\infty & \text{còn lại.} \end{cases}$$

Từ đây, bằng cách đặt $W_1 = 2W_{11}$, $W_2 = 2W_{22}$ ta có bài toán đối ngẫu nửa xác định dương của bài (3.5) :

$$\begin{aligned} & \text{minimize} && \frac{1}{2} (\text{Tr}(W_1) + \text{Tr}(W_2)) \\ & \text{sao cho} && \begin{bmatrix} W_1 & -X \\ -X^T & W_2 \end{bmatrix} \succeq 0. \end{aligned} \quad (3.8)$$

Do đó, từ bài toán chuẩn nguyên tử

$$\begin{aligned} & \text{minimize} && \|X\|_* \\ & \text{sao cho} && \mathcal{A}(X) = b. \end{aligned} \quad (3.9)$$

ta có thể đưa về bài toán quy hoạch nửa xác định dương như sau:

$$\begin{aligned} & \text{minimize} && \frac{1}{2} (\text{Tr}(W_1) + \text{Tr}(W_2)) \\ & \text{sao cho} && \begin{bmatrix} W_1 & -X \\ -X^T & W_2 \end{bmatrix} \succeq 0 \\ & && \mathcal{A}(X) = b. \end{aligned} \quad (3.10)$$

Với các ma trận có kích thước không quá lớn, chúng ta có thể dùng phương pháp điểm trong cho bài toán nửa xác định dương để giải. Hiện tại đã có các công cụ thư viện để giải sẵn các bài toán này như SeDuMi [20] hay SDPT3 [21]. Trong luận văn này, chúng tôi sẽ sử dụng thư viện CVXOPT (<https://cvxopt.org/>), một phần mềm khá mạnh để giải các

bài toán tối ưu lồi, và cụ thể hơn trong trường hợp đây là giải bài toán chuẩn nguyên tử bằng phương pháp điểm trong cho bài toán nửa xác định dương. Chúng tôi sẽ trình bày chi tiết về thư viện CVXOPT cũng như cách thực hiện ở phần thử nghiệm số 3.3.

3.2 Thuật toán proximal gradient

Như chúng ta có thể thấy, để tìm nghiệm của bài toán cực tiểu chuẩn nguyên tử với việc tính chính xác ràng buộc $\mathcal{A}(X) = b$ mất rất nhiều chi phí tính toán, mà trong thực tế thì việc tính chính xác tuyệt đối là không cần thiết hay không mang nhiều ý nghĩa. Lí do là, các phần tử quan sát được có thể chứa nhiễu, và có thể không tồn tại ma trận hạng thấp thỏa mãn điều kiện $\mathcal{A}(X) = b$. Do vậy, thay vì giải bài toán cực tiểu chuẩn nguyên tử, ta có thể xét bài toán tối ưu không ràng buộc sau:

$$\text{minimize}_{X \in \mathbb{R}^{m \times n}} \quad \frac{1}{2} \|\mathcal{A}(X) - b\|_2^2 + \mu \|X\|_*, \quad (3.11)$$

với μ là tham số. Mô hình 3.11 đều được nghiên cứu bởi nhiều nhà toán học [22, 23],... Trong luận văn này, chúng tôi xin trình bày thuật toán proximal gradient và phiên bản tăng tốc để giải bài toán phục hồi ma trận với kích thước ma trận lớn ([24]).

3.2.1 Toán tử Proximal

Định nghĩa 3.1. ([12]) Cho hàm $f : \mathbb{R}^n \rightarrow (-\infty, \infty]$. Toán tử proximal được định nghĩa bởi:

$$\text{prox}_f(x) = \text{argmin}_{u \in \mathbb{R}^n} \left\{ f(u) + \frac{1}{2} \|u - x\|^2 \right\} \quad \text{với } x \in \mathbb{R}^n.$$

Định lý 3.2. ([12]) Cho $f : \mathbb{R}^n \rightarrow (-\infty, \infty]$ là hàm đóng và lồi. Khi đó $\text{prox}_f(x)$ là duy nhất với mọi $x \in \mathbb{R}^n$ bất kỳ.

Ví dụ 3.3. Nếu $f \equiv c$ với $c \in \mathbb{R}$ nào đó, thì

$$\text{prox}_f(x) = \text{argmin}_{u \in \mathbb{R}^n} \left\{ c + \frac{1}{2} \|u - x\|^2 \right\} = x.$$

Ví dụ 3.4. Cho $f(x) = \langle a, x \rangle + b$, với $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$.

$$\begin{aligned} \text{prox}_f(x) &= \operatorname{argmin}_{u \in \mathbb{R}^n} \left\{ \langle a, u \rangle + b + \frac{1}{2} \|u - x\|^2 \right\} \\ &= \operatorname{argmin}_{u \in \mathbb{R}^n} \left\{ \langle a, x \rangle + b - \frac{1}{2} \|a\|^2 + \frac{1}{2} \|u - (x - a)\|^2 \right\} \\ &= x - a. \end{aligned}$$

Ví dụ 3.5. Cho $g : \mathbb{R} \rightarrow \mathbb{R}$, được xác định bởi $g(x) = \lambda|x|$, với $\lambda \in \mathbb{R}_+$. $\text{prox}_g(x)$ là nghiệm cực tiểu của hàm:

$$h(u) = \begin{cases} h_1(u) \equiv \lambda u + \frac{1}{2}(u - x)^2, & u > 0, \\ h_2(u) \equiv -\lambda u + \frac{1}{2}(u - x)^2, & u \leq 0. \end{cases}$$

Nếu $u > 0$, thì $0 = h'_1(u) = \lambda + u - x$, hay $u = x - \lambda$. Do đó, nếu $x > \lambda$, thì $\text{prox}_g(x) = x - \lambda$.

Nếu $u \leq 0$, tương tự ta có $x < -\lambda$, lúc này $\text{prox}_g(x) = x + \lambda$. Nếu $|x| \leq \lambda$, thì $\text{prox}_g(x) = 0$.

Vậy $\text{prox}_g(x) = \operatorname{sgn}(x)[|x| - \lambda]_+$.

3.2.2 Thuật toán Proximal

Xét mô hình tổng quát

$$\operatorname{minimize}_{x \in \mathbb{R}^n} \{F(x) := f(x) + g(x)\}, \quad (3.12)$$

trong đó:

- $g : \mathbb{R}^n \rightarrow R$ là hàm lồi, có thể không trơn.
- $f : \mathbb{R}^n \rightarrow R$ là hàm lồi, L-trơn, nghĩa là f khả vi liên tục và ∇f thoả mãn:

$$\|\nabla f(x) - \nabla f(y)\|_F \leq L_f \|x - y\|, \quad \forall x, y \in \mathbb{R}^n,$$

với $L_f > 0$.

Để hiểu ý tưởng đằng sau thuật toán giải (3.12), ta sẽ bắt đầu từ thuật toán chiếu gradient để giải bài toán 3.12 trong trường hợp $g = \delta_C$ với C là tập đóng và lồi, Trong trường hợp này, bài toán đưa về dạng

$$\text{minimize } \{f(x) : x \in C\}. \quad (3.13)$$

Bước cập nhật tổng quát của phương pháp chiếu gradient cho bài toán (3.13) là

$$x^{k+1} = P_C(x^k - t_k \nabla f(x^k)),$$

với t_k là độ dài bước tại vòng lặp k . Dễ dàng, ta có thể viết lại bước cập nhật như sau

$$x^{k+1} = \operatorname{argmin}_{x \in C} \left\{ f(x^k) + \langle \nabla f(x^k), x - x^k \rangle + \frac{1}{2t_k} \|x - x^k\|^2 \right\}.$$

Tức là, giá trị lần lặp tiếp theo là nghiệm cực tiểu trên C của tổng tuyến tính của phần hàm trơn tại bước lặp hiện tại cộng với hạng tử gần kề bậc hai.

Quay lại mô hình tổng quát 3.12, thật tự nhiên khi tổng quát hoá ý tưởng trên và xác định giá trị của bước lặp tiếp theo như nghiệm cực tiểu của tổng tuyến tính của f tại x^k , hàm không trơn g và hạng tử gần kề bậc hai:

$$x^{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ f(x^k) + \langle \nabla f(x^k), x - x^k \rangle + g(x) + \frac{1}{2t_k} \|x - x^k\|^2 \right\}. \quad (3.14)$$

Sau một số bước biến đổi đại số và lược bỏ các số hạng không đổi, ta có thể viết lại (3.14) thành

$$x^{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ t_k g(x) + \frac{1}{2} \|x - (x^k - t_k \nabla f(x^k))\|^2 \right\}. \quad (3.15)$$

Ta có thể viết lại 3.15 theo định nghĩa của toán tử proximal:

$$x^{k+1} = \operatorname{prox}_{t_k g}(x^k - t_k \nabla f(x^k)).$$

Từ đây, ta có thể mô tả thuật toán proximal gradient với kích thước bước nhảy $t_k = 1/L_k$ như sau:

Thuật toán proximal gradient

Khởi tạo: Chọn $x^0 \in \text{int}(\text{dom}(f))$.

Cho $k = 0, 1, 2, \dots$, các bước được thực hiện như sau:

Bước 1: Chọn $L_k \in \left(\frac{L_f}{2}, \infty\right)$.

Bước 2: Đặt $x^{k+1} = \text{prox}_{\frac{1}{L_k}} g \left(x^k - \frac{1}{L_k} \nabla f(x^k)\right)$.

Tốc độ hội tụ của thuật toán proximal gradient được chứng minh là $O(1/k)$ [12].

Tiếp theo, chúng tôi sẽ trình bày phương pháp cải tiến để có thể tăng tốc độ hội tụ của thuật toán, có thể đạt đến $O(1/k^2)$, thường được gọi là "FISTA", được đưa ra bởi Beck và Teboulle [25]. Thuật toán FISTA được trình bày như sau:

Thuật toán proximal gradient tăng tốc (FISTA)

Khởi tạo: Chọn $x^0 \in \mathbb{R}^n$. Đặt $y_0 = x_0$ và $t_0 = 1$. Chọn $L_k = L_f$
Cho $k = 0, 1, 2, \dots$, tạo x^{k+1} từ x^k bởi các bước lặp sau:

Bước 1 $x^{k+1} = \text{prox}_{\frac{1}{L_k g}} \left(y^k - \frac{1}{L_k} \nabla f(y^k)\right)$;

Bước 2 $t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$

Bước 3 $y^{k+1} = x^{k+1} + \left(\frac{t_k - 1}{t_{k+1}}\right) (x^{k+1} - x^k)$

Tính đúng đắn của thuật toán và tốc độ hội tụ của thuật toán có thể tham khảo ở [25].

Bây giờ, thuật toán FISTA được áp dụng cho bài toán chuẩn nguyên tử như sau

$$\min_{X \in \mathbb{R}^{m \times n}} \frac{1}{2} \|\mathcal{A}(X) - b\|_2^2 + \mu \|X\|_*, \quad (3.16)$$

với $f(X) = \frac{1}{2} \|\mathcal{A}(X) - b\|_2^2$ và $g(X) = \mu \|X\|_*$.

Ta có $\nabla f(Y) = \mathcal{A}^T(\mathcal{A}(Y) - b)$

Nên

$$\begin{aligned} X^{k+1} &= \text{prox}_{\frac{1}{L_k}g} \left(Y^k - \frac{1}{L_k} \nabla f(Y^k) \right) \\ &= \text{prox}_{\frac{1}{L_k}g} \left(Y^k - \frac{1}{L_k} \mathcal{A}^T(\mathcal{A}(Y) - b) \right). \end{aligned}$$

Theo [23], thì

$$X^{k+1} = U \cdot \text{diag}((\sigma - \frac{\mu}{L_k})_+) V^T,$$

với U, V và σ là từ phân tích SVD của

$$Y^k - (L_k)^{-1} \mathcal{A}^T(\mathcal{A}(Y) - b) = U \cdot \text{diag}(\sigma) V^T.$$

Từ đây, ta có thể giải bài toán chuẩn nguyên tử bằng phương pháp FISTA với các bước như sau:

FISTA cho (3.11)

Khởi tạo: Chọn $X^0 \in \mathbb{R}^{m \times n}$, đặt $Y_0 = X_0$ và $t_0 = 1$, chọn $L_k > 0$
Cho $k = 0, 1, 2, \dots$, tạo X^{k+1} từ X^k bởi các bước lặp sau:

Bước 1 Đặt $G^k = Y^k - \frac{1}{L_k} \mathcal{A}^T(\mathcal{A}(Y^k) - b)$

Tính SVD của G_k : $G_k = U \cdot \text{diag}(\sigma) V^T$.

Bước 2 $X^{k+1} = U \text{diag}((\sigma - \frac{\mu}{L_k})_+) V^T$.

Bước 3 $t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$.

Bước 4 $Y^{k+1} = X^{k+1} + \left(\frac{t_k - 1}{t_{k+1}} \right) (X^{k+1} - X^k)$.

3.3 Thử nghiệm số

Ở trong phần này, ta sẽ trình bày các kết quả về thử nghiệm số để giải bài toán phục hồi ma trận thông qua bài toán chuẩn nguyên tử. Ở thử nghiệm số 1, ta sẽ so sánh thuật toán điểm trong cho bài toán nửa xác định dương với thuật toán FISTA với ma trận được tạo ngẫu nhiên theo phân phối chuẩn với số chiều nhỏ. Ở thử nghiệm số 2, ta sẽ giải

quyết bài toán xử lý ảnh, cụ thể là phục hồi ảnh bằng thuật toán FISTA và đánh giá độ hiệu quả của nó. Ở thử nghiệm số 3, ta sẽ giải quyết bài toán với kích thước ma trận lớn, cụ thể là giải bài toán Netflix với bộ dữ liệu MovieLens 100k (<https://grouplens.org/datasets/movielens/100k/>). Ngôn ngữ phần mềm chúng tôi thực hiện ở đây bằng PYTHON. Ở thuật toán FISTA, các lần thử nghiệm trong luận văn này, chúng tôi đặt $L_f = 1, \mu = 10^{-4} \cdot \|A^T(b)\|_2$, dựa theo kết quả nghiên cứu [24].

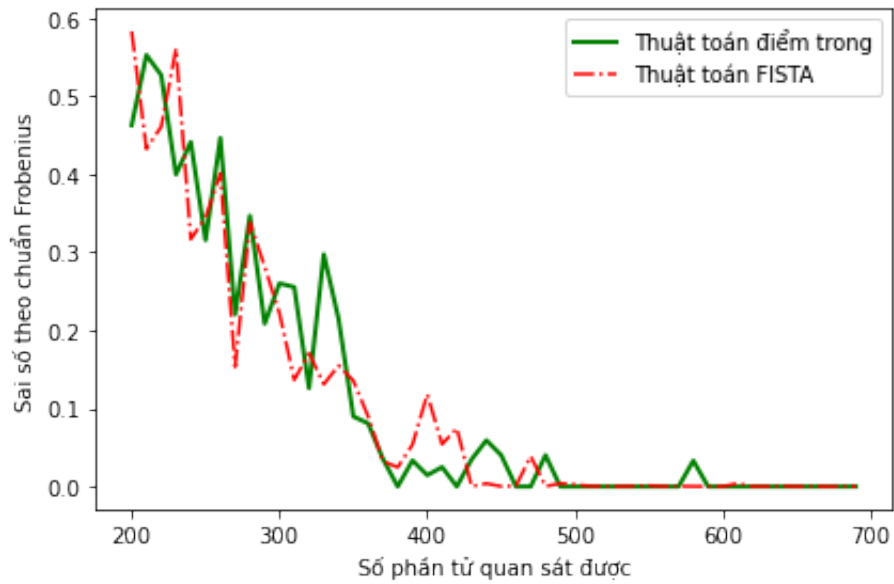
3.3.1 Thử nghiệm với ma trận kích thước bé

Với thuật toán điểm trong cho bài toán nửa xác định dương, chúng tôi sử dụng thư viện CVXOPT (<https://cvxopt.org/>) để giải. CVXOPT là một thư viện miễn phí được viết trên ngôn ngữ lập trình Python giúp giải rất nhiều các bài toán tối ưu lồi, được phát triển bởi Martin Andersen, Joachim Dahl, và Lieven Vandenberghe. Trong thư viện này cung cấp gói lệnh để giải bài toán chuẩn nguyên tử viết theo ý tưởng của [26].

Để so sánh hai thuật toán điểm trong cho bài toán nửa xác định dương và thuật toán FISTA, ở thí nghiệm đầu chúng tôi tạo một ma trận ngẫu nhiên M có kích thước 30×30 , tương ứng 900 phần tử, hạng bằng 3, bằng cách lấy hai ma trận cỡ 30×3 là M_L và M_R với các phần tử được lấy ngẫu nhiên độc lập tuân theo theo luật phân phối Gaussian và đặt $M = M_L \cdot M_R^T$. Để đảm bảo điều kiện có thể phục hồi ma trận ma trận, ta lấy ngẫu nhiên tập Ω các phần tử quan sát được theo phân phối đều. Trong thí nghiệm này, chúng tôi chạy chương trình với số phần tử quan sát được từ [200, 700], tương ứng với tỉ lệ số phần tử đã biết chiếm từ 20% đến 80% để đánh giá tốc độ hội tụ, khả năng phục hồi chính xác ma trận và thời gian chạy.

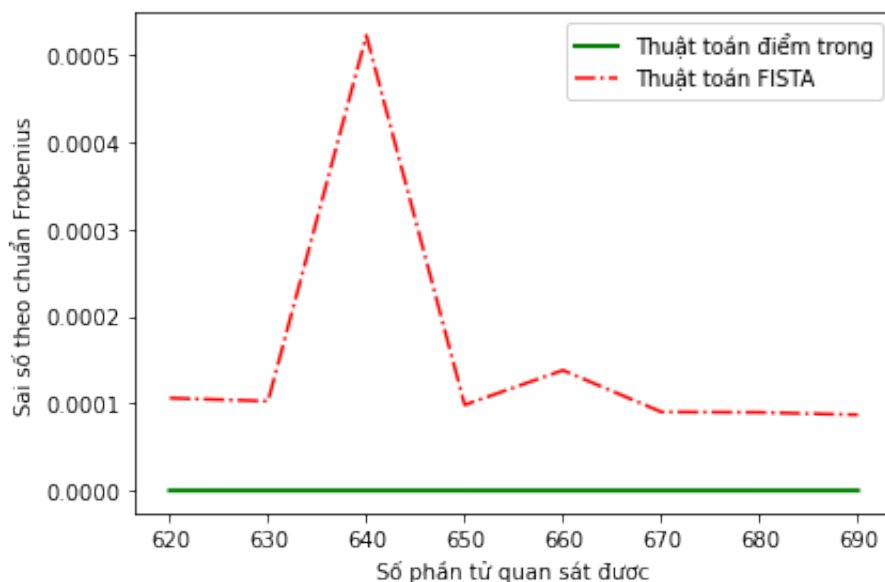
Ta sẽ đánh giá sai số của thuật toán bằng cách tính $\epsilon = \|X_{opt} - M\|_F / \|M\|_F$, với X_{opt} là kết quả thuật toán trả về, là ma trận sau khi được phục hồi. Nếu ϵ càng bé thì kết quả phục hồi ma trận càng tốt, nghĩa là ma trận sau khi phục hồi sẽ càng chính xác so với ma trận ban đầu.

Kết quả chạy ở thí nghiệm đầu tiên được thể hiện ở Hình 3.1 dưới đây:



Hình 3.1: Kết quả chạy thử nghiệm với kích thước ma trận cỡ 30×30 , số lượng quan sát được chiếm từ 20% đến 80%.

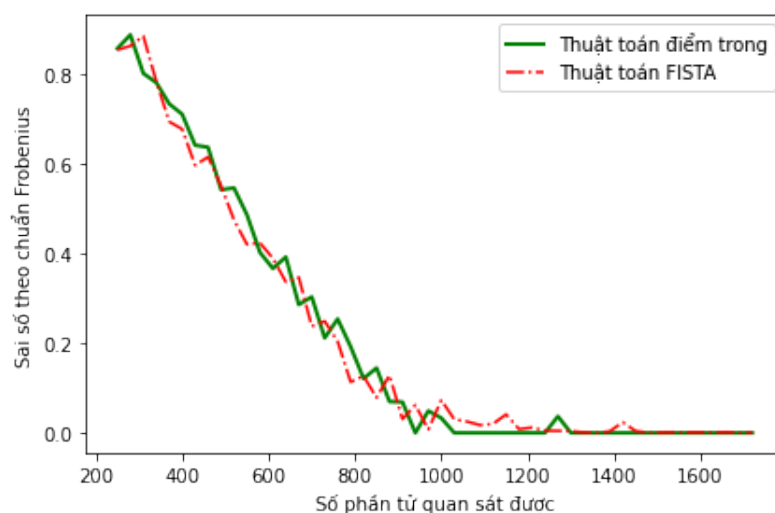
Nhìn vào kết quả thực nghiệm, ta có thể thấy về tốc độ hội tụ và khả năng phục hồi ma trận của hai thuật toán là như nhau. Thời gian chạy của hai thuật toán trong trường hợp kích thước nhỏ cũng xấp xỉ nhau. Tuy nhiên, ta xét kĩ hơn về sai số sau khi hai thuật toán đã hội tụ ở Hình 3.2.



Hình 3.2: Kết quả chạy thử nghiệm với kích thước ma trận cỡ 30×30 , số lượng quan sát lớn chiếm từ 68% đến 80%.

Ta có thể thấy rằng, khi sử dụng phương pháp điểm trong cho bài toán nửa xác định dương, nghiệm thu được thoả mãn ràng buộc trong bài toán cực tiểu chuẩn nguyên tử, còn trong khi sử dụng thuật toán (FISTA), thì sai số xấp xỉ 10^{-4} .

Chúng tôi tiếp tục thực nghiệm thứ 2, với việc tạo ma trận ngẫu nhiên tương tự với thí nghiệm trên, tuy nhiên kích thước ma trận thay đổi. Kích thước ma trận với lần thử nghiệm này là 50×50 . Ta có thể quan sát kết quả chạy với Hình 3.3 dưới đây:



Hình 3.3: Kết quả chạy thử nghiệm với kích thước ma trận cỡ 50×50 , số lượng quan sát được chiếm từ 20% đến 80%

Có thể thấy rằng về tốc độ hội tụ và sai số thì ta có thể đánh giá nó giống với thí nghiệm đầu tiên. Tuy nhiên, điều chúng tôi muốn nói ở đây là về mặt thời gian. Sau khi tăng kích thước ma trận từ 30×30 lên 50×50 thì thuật toán điểm trong cho bài toán nửa xác định dương đã chạy lâu hơn gấp 3-5 lần so với thuật toán FISTA. Ta có thể xem kết quả về thời gian chạy của hai thuật toán như Hình 3.4.

Cũng như đã nói ở phần 3.1, thì thuật toán điểm trong cho bài toán nửa xác định dương không phù hợp cho các bài toán toán có số chiều lớn. Vậy nên, với ma trận có kích thước bé, và mong muốn cần phục hồi một cách chính xác đến mức hoàn hảo các ma trận thì ta sẽ dùng thuật toán điểm trong, còn với những bài toán thực tế thì chúng ta nên sử dụng thuật

Thuật toán	n	Hạng	Tỉ lệ quan sát	Sai số	Thời gian (giây)
Thuật toán điểm trong	30	3	0.3	0.396	1.16
			0.4	0.030	1.3
			0.6	4.4e-8	0.68
FISTA	30	3	0.3	0.350	0.86
			0.4	0.160	1.76
			0.6	0.0003	0.97
Thuật toán điểm trong	50	4	0.3	0.326	10.03
			0.4	0.072	9.09
			0.6	7.59e-8	4.66
FISTA	50	4	0.3	0.227	1.39
			0.4	0.024	1.55
			0.6	0.001	1.47

Hình 3.4: Kết quả về sai số và thời gian của hai thuật toán điểm trong và thuật toán FISTA

toán FISTA. Trong các lần thực nghiệm tiếp theo, chúng tôi sẽ dùng thuật toán FISTA để xử lý các bài toán thực tế, cụ thể là bài toán xử lý ảnh và bài toán Netflix.

3.3.2 Xử lý ảnh với thuật toán FISTA

Để chỉ ra các ứng dụng của bài toán chuẩn nguyên tử, trong thí nghiệm này chúng tôi áp dụng thuật toán FISTA trong việc xử lý ảnh. Xử lý ảnh là một phân ngành trong xử lý số tín hiệu với tín hiệu xử lý là ảnh. Đây là một phân ngành khoa học mới rất phát triển trong những năm gần đây. Xử lý ảnh gồm 4 lĩnh vực chính: xử lý nâng cao chất lượng ảnh, nhận dạng ảnh, nén ảnh và truy vấn ảnh. Sự phát triển của xử lý ảnh đem lại rất nhiều lợi ích cho cuộc sống của con người.

Ngày nay xử lý ảnh đã được áp dụng rất rộng rãi trong đời sống như: photoshop, nén ảnh, nén video, nhận dạng biển số xe, nhận dạng khuôn mặt, nhận dạng chữ viết, xử lý ảnh thiên văn, ảnh y tế,....

Trong thí nghiệm này, chúng tôi sẽ minh họa cho việc phục hồi ảnh từ một bức ảnh nhiễu. Một bức ảnh có thể biểu diễn dưới dạng các ma trận. Thế nên việc phục hồi một bức ảnh nhiễu, tương đương với bài toán phục hồi ma trận với một tỉ lệ phần tử đã biết.

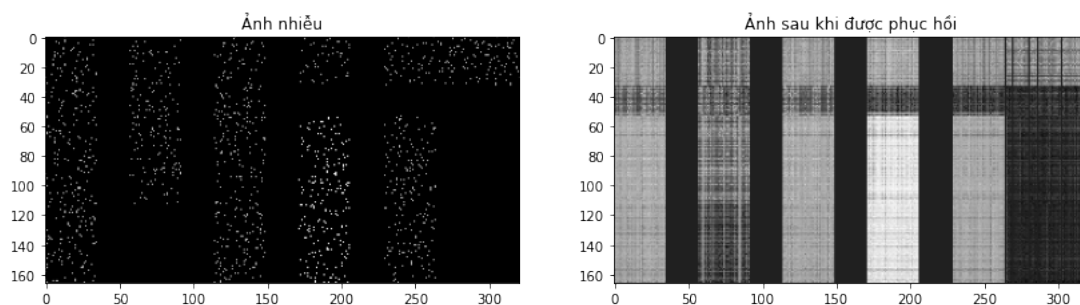
Thí nghiệm được thực hiện ở các bước như sau: Đầu tiên, chúng tôi

sử dụng bức ảnh là logo của MIT, được lấy từ trang web https://vi.m.wikipedia.org/wiki/T%E1%BA%ADp_tin:MIT_logo.svg Bức ảnh gốc ban đầu có hình dạng như sau:



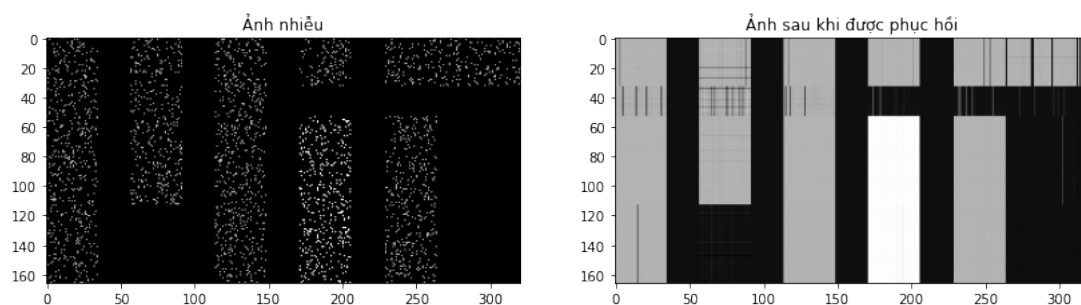
Hình 3.5: Logo MIT.

Đây là bức ảnh có kích thước 166×321 (53286 phần tử), hạng của ma trận là 4. Rõ ràng ta thấy đây là một ma trận hạng thấp. Để đơn giản hoá, chúng tôi sẽ thực hiện với bức ảnh xám, tức là bức ảnh được chuyển về thành một lớp ma trận. Chúng tôi sẽ minh hoạ việc phục hồi ma trận bằng thuật toán FISTA. Các kết quả thử nghiệm được thể hiện dưới đây:

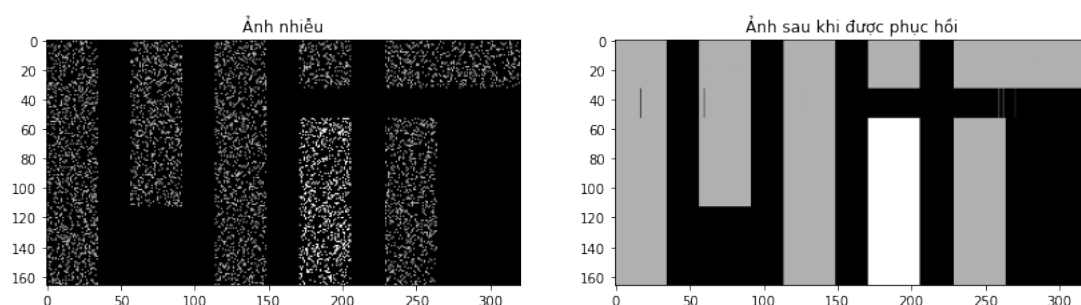


Hình 3.6: Ví dụ cho việc phục hồi ảnh với tỉ lệ số điểm ảnh đã biết chiếm 5%.

Ta có thể thấy rằng, đối với ma trận hạng thấp, mặc dù chúng ta chỉ biết 5% điểm ảnh nhưng chúng ta có thể khôi phục được bức ảnh có thể nhìn khá rõ. Còn với tỉ lệ đã biết chiếm 20% thì ta gần như đã khôi phục được hoàn hảo bức ảnh. Điều này cho ta thấy bài toán chuẩn nguyên tử đã giải quyết rất tốt cho bài toán xử lý ảnh.



Hình 3.7: Ví dụ cho việc phục hồi ảnh với tỉ lệ số điểm ảnh đã biết chiếm 10%.



Hình 3.8: Ví dụ cho việc phục hồi ảnh với tỉ lệ số điểm ảnh đã biết chiếm 20%.

3.3.3 Bài toán Netflix

Chúng tôi sẽ sử dụng bộ cơ sở dữ liệu MovieLens 100k bởi GroupLens (<https://grouplens.org/datasets/movielens/>). Trong này, người ta cung cấp các bộ dữ liệu có kích thước từ nhỏ đến lớn: 100.000, 1.000.000, 10.000.000, 25.0000.000 lượt đánh giá. Trong luận văn này, nhằm mục đích để minh họa cho thuật toán, chúng tôi đã sử dụng bộ dữ liệu nhỏ nhất với 100.000 lượt đánh giá được công bố vào tháng 4/1998.

Trong bộ dữ liệu 100.000 lượt đánh giá này có các tệp tin như sau:

- **u.data:** đây là tệp chứa 100.000 lượt đánh giá cho 1682 bộ phim từ 943 người dùng. Nhà cung cấp cũng cho biết thêm, sẽ có ít nhất 20 bộ phim được đánh giá từ mỗi người dùng. Các lượt đánh giá sẽ nhận giá trị từ 1 đến 5 sao.
- **ua.base, ua.test, ub.base, ub.test:** Ở đây để thuận lợi cho việc

áp dụng các thuật toán, chia bộ dữ liệu thành hai tập con để một tập là huấn luyện và một tập để đánh giá kết quả. Trong phạm vi luận văn này, chúng tôi sẽ dùng bộ *ua.base* và *ua.test*.

- **u.user:** tệp tin này chứa các thông tin người dùng, bao gồm: id, tuổi, giới tính, nghề nghiệp, vùng miền vì người ta tin rằng các thông tin này có thể ảnh hưởng tới sở thích các người dùng. Tuy nhiên, trong thuật toán của chúng tôi đang sử dụng, chúng tôi coi các người dùng là bình đẳng nhau, nên chúng tôi không sử dụng thông tin này.
- **u.genre:** tệp tin này chứa tên của 19 thể loại phim. Các thể loại bao gồm: unknown, Action, Comedy, Crime, Adventure, Animation, Children's, Documentary, Drama, War, Fantasy, Sci-Fi, Thriller, Film-Noir, Horror, Musical, Mystery, Romance, Western. Tương tự như phần người dùng, chúng tôi coi các bộ phim là bình đẳng nhau, không có mối liên hệ hay đặc tính chung giữa các bộ phim với nhau, nên thông tin này chúng tôi cũng không sử dụng.

Vì trong bài toán phục hồi ma trận, chúng tôi đã coi các người dùng và các bộ phim là độc lập nhau, không có sự tương quan bất kì giữa hai người dùng hay hai bộ phim nào với nhau, nên ở đây, chúng tôi chỉ quan tâm đến lượt đánh giá của người dùng đến các bộ phim, không quan tâm đến thông tin người dùng cũng như tính chất các bộ phim. Như đã nói ở trên, chúng tôi sẽ dùng bộ *ua.base* để huấn luyện, bộ *ua.test* là bộ dùng để kiểm tra đánh giá độ chính xác của thuật toán. Bộ *ua.base* và *ua.test* được chia ra từ bộ dữ liệu gốc với tỉ lệ 9:1. Dữ liệu đầu vào tập huấn luyện có 943 người dùng, 1682 bộ phim với 90570 lượt đánh giá. Có nghĩa ở đây, số phần tử đã biết trước chỉ chiếm 5.7% Dữ liệu đầu vào có dạng như sau:

	user_id	movie_id	rating	unix_timestamp
0	1	1	5	874965758
1	1	2	3	876893171
2	1	3	4	878542960
3	1	4	3	876893119
4	1	5	3	889751712
...
90565	943	1047	2	875502146
90566	943	1074	4	888640250
90567	943	1188	3	888640250
90568	943	1228	3	888640275
90569	943	1330	3	888692465

Hình 3.9: Dữ liệu đầu vào của bài toán Netflix.

Trong bảng dữ liệu này, người ta xếp theo thứ tự ID của người dùng, ở mỗi người dùng sẽ thể hiện lượt đánh giá của họ tương ứng với các ID của phim theo thứ tự. Ta có thể thấy, người dùng thứ nhất xem phim một và đánh giá là 5 sao, người dùng thứ nhất xem phim hai đánh giá 3 sao, người dùng thứ 943 xem phim thứ 1228 đánh giá 3 sao,... Người dùng thứ 943 không xem phim hoặc không đánh giá phim có ID là 1229, nên không hiện lượt đánh giá ở trong ma trận dữ liệu đầu vào. Tương tự như thế với các người dùng và các phim còn lại. Ví dụ, như người dùng có ID là 1, ta có thể thấy người dùng có ID là "1" đã đánh giá những bộ phim nào:

```
Các bộ phim đã đánh giá:
[ 0  9 13 18 24 99 110 126 236 241 254 256 257 268 271 272 273 274
 275 276 277 278 281 282 283 284 285 286 287 288 290 292 293 294 295 297
 298 299 300 301 302 303 304 305 306 307 308 309 310 312 314 315]
Các lượt đánh giá tương ứng với mỗi bộ phim:
[4 2 4 3 4 5 4 5 4 5 4 4 3 4 5 4 3 5 4 4 3 4 4 5 4 5 4 3 3 3 3 4 1 4 3 3 4
 4 4 5 4 4 3 4 3 3 1 4 5 5 1 5]
```

Hình 3.10: Đọc kết quả dữ liệu Netflix với người dùng có ID là "1".

Vì để phục hồi ma trận dữ liệu đầu vào, hay nói cách khác là dự đoán các lượt đánh giá của người dùng đối với các bộ phim, thì với bài phục hồi

ma trận, ta cần các hàng và các cột có tối thiểu ít nhất một phần tử. Đơn giản là nếu ta không có một tí thông tin gì về người dùng hay bộ phim thì ta không có một cơ sở nào để có thể dự đoán về người dùng đó sẽ đánh giá cao bộ phim nào, hay bộ phim đó sẽ phù hợp với người nào. Nó giống như kiểu để dự đoán trong một kì thi bất kì, bạn A được bao nhiêu điểm, nhưng ta không có bất kì thông tin nào về bạn A thì ta không thể nào dự đoán được bạn A đạt bao nhiêu điểm.

Vậy nên, trước khi áp dụng thuật toán, chúng tôi có bước xử lý dữ liệu với ma trận đầu vào. Vì trong bộ dữ liệu đã chắc chắn mỗi người dùng đánh giá ít nhất 20 bộ phim rồi nên chúng tôi chỉ cần kiểm tra xem có bộ phim nào không có lượt đánh giá không. Với các bộ phim nào chưa có lượt đánh giá, thường chúng ta sẽ có hai cách là bỏ bộ phim đó ra khỏi bộ dữ liệu, hoặc điền ngẫu nhiên một lượt đánh giá của bộ phim tương ứng với người dùng nào đó.

Sau khi xử lý dữ liệu xong, chúng tôi sẽ dùng thuật toán FISTA để xử lý thuật toán, với bài toán chuẩn nguyên tử như sau:

$$\begin{aligned} & \text{minimize} \quad \text{rank}(\mathbf{X}) \\ & \text{thỏa mãn} \quad X_{ij} = M_{ij} \quad (i, j) \in \Omega. \end{aligned} \tag{3.17}$$

với M_{ij} là ma trận dữ liệu đầu vào không đầy đủ, Ω là tập các chỉ số ID các người dùng i đã đánh giá các bộ phim có ID là j . Vì một số phần tử trong ma trận M không có thông tin, nên ta không thể tính toán sai số như cách tính ở phần thí nghiệm như chúng ta đã làm với các ma trận được tạo ngẫu nhiên trong phần 3.3.1.

Trong cuộc thi Netflix, thì người ta đánh giá sai số theo công thức RMSE, tức căn bậc hai của trung bình cộng bình phương của lỗi. RMSE càng bé thì cho ta thấy kết quả dự đoán càng tốt. RMSE được tính như sau:

$$RMSE = \sqrt{\sum_{(i,j) \in T} \frac{(M_{ij} - X_{ij})^2}{|T|}}$$

với X_{ij} là ma trận sau khi ta đã tính toán các lượt đánh giá của người dùng i đối với phim j tương ứng; T là tập các chỉ số mà có lượt đánh giá trên

tập kiểm tra.

Sau khi chạy với thuật toán FISTA, ta có kết quả sai số RMSE là: **0.9077**. Như vậy, sai số vào khoảng 0.9 sao. Kết quả này được đánh giá như nào? Theo như tìm hiểu của chúng tôi, nếu chỉ sử dụng các mô hình hồi quy tuyến tính đơn giản thì $RMSE = 1.0540$. Bài toán phục hồi ma trận trong trường hợp bài toán Netflix này tương ứng với các bài toán hệ gợi ý, mà mục tiêu của các bài toán hệ gợi ý chỉ cần chỉ ra được danh sách các bộ phim được đánh giá cao nhất tương ứng với mỗi người dùng, có thể là top 5 bộ phim, hay 10 bộ phim,...

Và bên ban tổ chức cuộc thi cũng đã nghiên cứu và tuyên bố rằng ngay cả khi những cải thiện nhỏ giúp RMSE giảm đi 1% đều dẫn đến sự khác biệt đáng kể trong việc lấy 10 bộ phim được đánh giá cao nhất tương ứng với mỗi người dùng. Vậy nên, thuật toán FISTA đã giúp cải thiện giảm RMSE 13.8%, thì chúng tôi cho rằng đây là một kết quả khá tốt.

KẾT LUẬN VÀ KIẾN NGHỊ

Bài toán cực tiểu chuẩn nguyên tử có ứng dụng rất đa dạng trong xử lý ảnh, hệ thống gợi ý, tài chính thống kê, xử lý tín hiệu địa chấn,... Trong luận văn này chúng tôi đã tìm hiểu và nghiên cứu được:

- Bài toán cực tiểu hàm hạng thông qua mô hình thực tế là bài toán điền các phần tử còn thiếu của ma trận (matrix completion), cụ thể trong bài toán Netflix ứng dụng trong Hệ thống gợi ý, từ đó đưa bài toán cực tiểu hàm hạng về bài toán cực tiểu chuẩn nguyên tử, dựa vào tính lồi của bài toán cực tiểu chuẩn nguyên tử.
- Các điều kiện RIP đảm bảo mối liên hệ của bài toán cực tiểu chuẩn nguyên tử và bài toán cực tiểu hàm hạng tương ứng.
- Hai phương pháp giải bài toán cực tiểu chuẩn nguyên tử là phương pháp điểm trong cho bài toán quy hoạch nửa xác định dương và phương pháp tăng tốc proximal gradient(FISTA). Chúng tôi cũng đã thử nghiệm số để so sánh hai phương pháp trên, và ứng dụng nó để giải bài toán xử lý ảnh và bài toán Netflix.

Hướng nghiên cứu tiếp theo

Chúng tôi mong muốn có thể xây dựng các thuật toán sử dụng trong hệ gợi ý, với số lượng người dùng có kích thước lớn từ vài triệu đến vài tỷ. Chúng tôi nhận thấy một số vấn đề trong quá trình nghiên cứu làm luận văn như sau:

1. Khi có người đánh giá về một bộ phim/ một mặt hàng nào đó, thì câu hỏi nên được đặt ra rằng sự đánh giá đó có phù hợp hay không? Ví dụ, nếu như một người ở rừng rậm, chưa từng tiếp xúc với thế giới văn minh và đánh giá về chất lượng của xe điện Tesla, thì sự đánh

giá đó không nên có ý nghĩa, cũng như hệ thống gợi ý đủ tốt cần phải loại trừ các trường hợp này.

- Chúng ta đã giả sử rằng, tất cả các người dùng, các bộ phim/hàng hóa,... đều bình đẳng với nhau thông qua việc loại bỏ đi tất cả các thông tin liên quan khác, ngoại trừ việc đánh giá. Trong khi đó, các thông tin quan trọng khác như sự quan hệ của những người đánh giá với nhau, sự tương đồng của các mặt hàng/ các bộ phim cũng nên đóng một vai trò lớn.

Thực tiễn chứng minh, nếu hầu hết tất cả những bạn bè của một người đều thích một mặt hàng nào đó, thì xác suất rất cao người đó cũng sẽ thích. Điều này cũng tương tự như trong lý thuyết mạng xã hội, hai người có cùng một lớp các bạn bè với nhau thì xác suất rất cao cũng sẽ là bạn bè quen biết với nhau, và do đó hệ thống gợi ý kết bạn nên gợi ý bạn bè chung.

- Đối với các bài toán thực tế có kích thước lớn như hiện nay (bigdata), với số người dùng có thể lên đến cả triệu, cả tỷ người dùng trong khi thuật toán FISTA chỉ đang giải tốt với ma trận có số chiều khoảng 10^5 [24], vậy nên việc tính toán với các bài toán bigdata có thể gặp nhiều khó khăn.

Câu hỏi không tầm thường đặt ra, làm cách nào để có thể xây dựng được một lý thuyết về gợi ý mà có thể hoạt động cho một số lượng cực lớn người dùng?

Một cách hiển nhiên, hệ thống như vậy cần phải được thiết kế để có thể chạy song song trên nhiều nhân của bộ vi xử lý, hoặc được thiết kế để có thể chạy trên nhiều máy tính của một hệ thống tính toán phân tán. Do đó, nếu như dữ liệu của chúng ta có thể được phân cụm một cách độc lập, thì vấn đề có thể coi như được giải quyết. Vậy cách phân cụm nên được diễn ra như thế nào?

Phụ lục

Ở đây là chúng tôi xin trình bày những ý tưởng nghiên cứu của định hướng phát triển đã nêu ở phía trên.

1. Cần phải có một cách nào đó để đưa thông tin của quan hệ giữa người dùng với nhau, cũng như của các bộ phim/mặt hàng với nhau vào trong hàm tối ưu.

Được khích lệ bởi lý thuyết mạng xã hội và phân tích dữ liệu topo, chúng tôi nghĩ đến sử dụng lý thuyết toán tử Laplace trên đồ thị.

Chúng ta nhắc lại rằng, cho một đồ thị $G = (V, E)$ có n đỉnh, và một ma trận $\Delta = (\delta_{i,j})$ thể hiện quan hệ giữa các đối tượng i và j . Cho $f : V \rightarrow \mathbb{R}^n$ là một hàm định nghĩa trên G , $\gamma : E \rightarrow \mathbb{R}$ là hàm trọng số.

Khi đó, toán tử Laplace được định nghĩa

$$\Delta f(i) = \sum_{j \in V} (f(i) - f(j)) \cdot \gamma_{i,j}$$

với mọi $i \in V$.

Ý nghĩa thực tế của toán tử này, đó là đo đặc mức độ vượt trội của một người so với bạn bè thân thiết của họ.

Chúng ta trang bị cho không gian các người dùng cấu trúc của một đồ thị với tác động của một toán tử Laplace rời rạc. Cụ thể hơn, giả sử P_i là phép chiếu chính tắc từ $\mathbb{R}^{m \times n}$ xuống \mathbb{R}^n , đưa X trở thành hàng thứ i của X , khi đó $P_i(X)$ trở thành một ma trận hàng. Liên hệ

giữa hai người dùng i và j , sẽ được trở thành liên hệ giữa hai vector $P_i(X)$ và $P_j(X)$ này.

Ta mong muốn, hai người i và j có hàm liên hệ $\gamma_{i,j}$, để nếu $\gamma_{i,j}$ rất lớn, thì hệ gợi ý nên gợi ý hai người dùng này rất gần nhau, và cộng thêm $\sum_{(i,j) \in \Gamma} \|P_i(X) - P_j(X)\|_2^2 \cdot \gamma_{i,j}$ vào hàm mục tiêu. Ở đây, V là tập những người dùng, $\Gamma \subset V \times V$ là tập con gồm các cặp người dùng có quan hệ với nhau. Đối với mỗi $(i, j) \in \Gamma$, ta cho tương ứng một số thực không âm $\gamma_{i,j}$ thể hiện quan hệ giữa hai người dùng. $\gamma_{i,j}$ bằng 0 nếu như hai người dùng i và j không có quan hệ với nhau, và nhận giá trị rất lớn nếu i và j có quan hệ mật thiết.

Câu chuyện xảy ra cũng hoàn toàn tương tự, ta có thể thêm yếu tố liên hệ vào các bộ phim/mặt hàng, trong đó Q là các phép chiếu lên trục bộ phim/mặt hàng của ma trận X , I là tập các bộ phim/mặt hàng, $\Lambda \subset I \times I$ là tập con gồm các cặp bộ phim/mặt hàng có quan hệ với nhau. Khi đó, hàm mục tiêu được viết dưới dạng

$$f = \|AX - b\|_2^2 + \mu \|X\|_* + \sum_{(i,j) \in \Gamma} \|P_i(X) - P_j(X)\|_2^2 \cdot \gamma_{i,j} \\ + \sum_{(k,l) \in \Lambda} \|Q_k(X) - Q_l(X)\|_2^2 \cdot \lambda_{k,l}$$

Hiển nhiên, nếu các tham số $\gamma_{i,j}$ và $\lambda_{k,l}$ suy biến, ta quay trở lại bài toán gốc thông thường. Chúng ta cũng có thể tính được ma trận $C \in \mathbb{R}^{m \times n}$, $d \in \mathbb{R}^k$ sao cho

$$f = \|CX - d\|_2^2 + \mu \|X\|_*$$

với C cho bởi

$$C^T C = A^T A + \sum_{(i,j) \in E} \gamma_{ij} (P_i - P_j)^T \cdot (P_i - P_j) + \sum_{(k,l) \in I.I} (\beta_{k,l} Q_k - Q_l)^T \cdot (Q_k - Q_l)$$

và d cho bởi

$$d = (C^T)^{-1} A^T b.$$

Tuy nhiên, cái quan trọng hơn, chúng ta có thể áp dụng định lý này để đưa bài toán gợi ý cho mạng xã hội về bài toán trong chương 3

một cách rất đơn giản mà gần như không phải thay đổi chương trình/ thuật toán.

Sau khi viết xong luận văn, chúng tôi thấy trong bài báo viết bởi Kim-Chuan Toh và Sangwoon Yun, 2009 ([24]), có thảo luận một vấn đề gần tương tự. Cụ thể hơn, hai tác giả này mong muốn có một quan hệ giữa người dùng i và mặt hàng thứ k xấp xỉ với một hằng số θ_{ik} cho trước. Và do đó, một hạng tử bậc hai khác cũng được cộng thêm vào cost function. Với ký hiệu của chúng tôi, hạng tử đó có dạng $|P_i(Q_k(X)) - \theta_{i,k}^2|^2$.

2. Chúng ta cũng biết rằng, về mặt thực tế, chỉ có thể giải bài toán hệ gợi ý ở khía cạnh địa phương do hạn chế về khả năng tính toán của máy tính. Hơn nữa, nên gom các người dùng có các thuộc tính nào đó gần với nhau, vì độ chính xác của thuật toán sẽ cao hơn.

Do đó, rất cần thiết để phân cụm người dùng thành các cụm, rồi sau đó mới áp dụng thuật toán gợi ý cho từng cụm. Và do đó, điều này có thể dễ dàng thực hiện được với một mạng máy tính phân tán.

Chú ý rằng, khi ta hạn chế thuật toán hệ gợi ý xuống một tập con O nào đó đồ thị, vấn đề phát sinh sẽ là có một cột ứng với một số lượng item nào đó không có đầy đủ thông tin. Chúng ta sẽ khắc phục điều này thông qua việc loại bỏ tạm thời các bộ phim/mặt hàng mà không có một đánh giá nào trong tập O .

Một điều ta có thể nhận xét thấy rằng, khi địa phương hoá hệ gợi ý trên một tập đủ nhỏ $O \subset G$ tập người dùng, thì hạng của ma trận được xây dựng nói chung sẽ nhỏ hơn hạng của ma trận được xây dựng trên toàn bộ đồ thị người dùng. Ý nghĩa thực tế của điều này, đó là mỗi tập con người dùng có một số tiêu chí riêng đối với các loại bộ phim/mặt hàng, và một phép toán gợi ý đầy đủ trên tập tất cả các người dùng sẽ nên có đầy đủ các tiêu chí của tất cả người dùng.

Vậy, vấn đề được đặt ra, khi nào thì các hạng địa phương sẽ có thể được ghép nối chung với nhau, trở thành một hạng toàn cục? Có trở ngại nào ngăn cản điều này? Và các ma trận đầy đủ này, khi ghép nối

với nhau có sự vênh nhau trên phần giao của các tập mở hay không?

Vấn đề được đặt ra ngược trở lại:

1. Nếu từ một tập mở U , ta liên tục thác triển giải tích toán tử làm đầy ma trận lên toàn bộ tập người dùng, thì sau một vòng và quay trở lại tập mở U , hạng của toán tử gợi ý cũng như ma trận được điền vào có sự thay đổi hay không?
2. Nếu từ một tập mở U , ta liên tục thác triển giải tích toán tử làm đầy ma trận lên toàn bộ tập người dùng, thì sau một vòng và quay trở lại tập mở U , hạng của ma trận được điền vào có sự thay đổi hay không?
3. Một mặt chúng ta có thể định nghĩa các tập mở U đủ nhỏ và nghiên cứu cách ghép nối lại để tạo ra đồ thị lớn, và nghiên cứu các hố, các lỗ trong đồ thị lớn này, và do đó có thể nghiên cứu quan hệ giữa topological data analysis với bài toán hệ gợi ý hoạt động ra sao?
4. Mặt khác chúng ta có thể hay không từ đây phát triển một mô hình nói về cách một ma trận có hạng r được mở rộng từ một tập mở sang toàn cục của đồ thị người dùng?
5. Mặt khác, bài toán này cung cấp một điều kiện để có thể ghép nối dữ liệu học được trong hệ thống gợi ý phân tán, để đưa ra một nghiệm toàn cục để có thể ứng dụng trong thực tế. Và cách thức triển khai tương ứng?

Chúng tôi tin rằng, đây là một bài toán có ý nghĩa, vì từ lý thuyết bài toán phục hồi ma trận hạng thấp, chúng ta có thể giải quyết bài toán ánh xạ một đồ thị vào một không gian vector có số chiều thấp thông qua kết quả của Candès-Terence Tao. Và vấn đề tương tự về quan hệ giữa địa phương và toàn cục cũng xuất hiện khi chúng ta giải quyết bài toán ở mức độ Large Scale trong công nghiệp. Và trên một đồ thị, người ta quan tâm tới các bài toán tối ưu rời rạc (ví dụ tối ưu vận tải VRP, hệ gợi ý, phân cụm ...) nên thông qua luận văn này, người ta hi vọng có thể đưa về bài toán tối ưu trên không gian Euclide (chứ không chỉ có bản đồ phi Euclide),

và từ đó có thể áp dụng thêm các công cụ của giải tích hiện đại vào các bài toán tối ưu rời rạc.

Tài liệu tham khảo

- [1] M. Fazel, H. Hindi, and S.P. Boyd. A rank minimization heuristic with application to minimum order system approximation. *Proceedings of the American Control Conference*, 6:4734–4739, 2001.
- [2] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259–268, 1992.
- [3] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, 73:273–282, 2011.
- [4] M. S. Lobo, M. Fazel, and S. Boyd. Portfolio optimization with linear and fixed transaction. *Annals of Operations Research*, 152:341–365, 2007.
- [5] L.N. Trefethen and D. Bau III. *Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, 1997.
- [6] Ricardo Correa da Silva. Lecture notes on noncommutative lp-spaces. <https://arxiv.org/abs/1803.02390>, 2018.
- [7] A. Connes. *Noncommutative Geometry*. Academic Press, San Diego, 1994.
- [8] E.J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51:4203–4215, 2005.
- [9] E.J. Candes and M. B. Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, 2008.

- [10] B. Recht, M. Fazel, and P.A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52:471–501, 2010.
- [11] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, USA, 2 edition, 2013.
- [12] A. Beck. *First-Order Methods in Optimization*, volume 25. Society for Industrial and Applied Mathematics, Philadelphia, USA, 2017.
- [13] T. Stromberg. *A study of the operation of infimal convolution*. Doctoral thesis, Sweden, 1994.
- [14] Hoang Tuy. *Convex Analysis and Global Optimization Volume 22*. Springer International Publishing AG, Cham, Switzerland, 2016.
- [15] E.J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9:717–772, 2009.
- [16] P. Chen and D. Suter. Recovering the missing components in a large noisy low-rank matrix: application to sfm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:1051–1063, 2004.
- [17] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9:137–154, 1992.
- [18] M.P. Chytil and V. Koubek. *Mathematical Foundations of Computer Science*. Springer Berlin, Heidelberg, 1984.
- [19] E.J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56:2053–2080, 2010.
- [20] J.F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11:625–653, 1999.

- [21] K.C. Toh, R.H. Tutuncu, and M.J. Todd. On the implementation of sdpt3 (version 3.1) - a matlab software package for semidefinite-quadratic-linear programming. *IEEE International Conference on Robotics and Automation*, pages 290–296, Taiwan, 2004.
- [22] S. Ma, D. Goldfarb, and L. Chen. Fixed point and bregman iterative methods for matrix rank minimization. *Mathematical Programming*, 128:321–353, 2009.
- [23] J. Cai, E.J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20:1956–1982, 2010.
- [24] K. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. *Pacific Journal of Optimization*, 6:615–640, 2010.
- [25] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2:183–202, 2009.
- [26] Z. Liu and L. Vandenberghe. Interior-point method for nuclear norm approximation with application to system identification. *SIAM Journal on Matrix Analysis and Applications*, 31:1235–1256, 2010.