به نام خدا

گزارش شماره 1 (4 فروردین سال 1399)

عنوان پروژه: بررسی مسئله درهم‌تنیدگی کوانتومی برای سیستم‌های دوذره‌ای توسط روش‌های یادگیری ماشین

اعضا: سیده فاطمه دهقان، مهکامه سلیمی، سحر تغیر، محمدمهدی ماستری فراهانی

فهرست مطالب

تعمیم ماتریس‌های پاولی(ماتریس‌های گلمان(GellMann matrix)) :

برای نمایش حالت سیستم‌های کوانتومی بسیار مناسب است که به جای استفاده از بردار حالت ،که یکی از بردارهای متعلق به فضای هیلبرت است، از ماتریس چگالی (ماتریس حالت) استفاده کنیم، چراکه توصیف کامل‌تری از سیستم ارایه می‌دهد. ماتریس چگالی یا عملگر چگالی یکی از اعضای فضای هیلبرت عملگرها است و این عملگرهای چگالی یک زیر فضا در این فضای هیلبرت می‌سازند[0].

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i| \quad \rightarrow \quad p_i \geq 0 \quad , \quad \sum_i p_i = 1$$

بنابراین واضح است که عملگر چگالی یک عملگر هرمیتی، مثبت و رد آن برابر با 1 است.

$$\rho = \rho^\dagger \quad , \quad Tr(\rho) = 1 \quad , \quad \langle\varepsilon_i|\rho|\varepsilon_i\rangle > 0 \quad \forall\varepsilon_i$$

برای توصیف سیستم‌های دوترازی کوانتومی، به عنوان مثال کیوبیت‌ها، می‌توان پایه‌هایی برای فضای هیلبرت عملگرهای حالتشان پیدا کرد و این پایه‌ها همان ماتریس‌های پاولی به همراه ماتریس واحد است.

$$basis = \{I, \sigma_1, \sigma_2, \sigma_3\}$$

که:

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \ , \ \sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \ , \ \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \ , \ \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

به عنوان یک توضیح مختصر از اینکه چرا می‌توان همواره هر عملگر چگالی را در این پایه بسط داد، کافی است به مثال زیر توجه کنید:

از آنجا که عملگر چگالی یک عملگر هرمیتی است می‌توان آنرا به صورت زیر نوشت:

$$\rho = \begin{pmatrix} a & b - ic \\ b + ic & d \end{pmatrix}$$

حال چون رد آن باید برابر با واحد باشد، داریم:

$$\rho = \begin{pmatrix} 1 + a & b - ic \\ b + ic & 1 - a \end{pmatrix} = I + b\sigma_1 + c\sigma_2 + a\sigma_3$$

بنابراین در حالت کلی توانستیم یک عملگر چگالی را بر حسب ماتریس‌های چگالی بسط دهیم. نمایش ریاضی آن اینگونه خواهد بود:

$$\rho = \sum_i a_i \sigma_i$$

حال می‌خواهیم حالت یک سیستم دوذره‌ای $(bipatite)$ را برحسب این پایه‌ها بنویسیم. می‌دانیم که حالت‌های دوذره‌ای در فضای هیلبرت بزرگتری هستند که از ضرب تانسوری دو فضای هیلبرت هر کدام از ذره‌ها به وجود آمده‌اند. به راحتی می‌توان آن‌ها را اینگونه نشان داد:

$$\rho_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$$

و اگر بخواهیم آن‌ها را بر حسب پایه‌های فضا (ماتریس‌های پاولی) بسط دهیم به راحتی می‌توان نوشت:

$$\rho = \sum_{i,j} a_{ij}\ \sigma_i \otimes \sigma_j$$

و بنابراین می‌توان هر حالت را با ضرایب بسط آن مشخص کرد. چون ماتریس‌ها پاولی هرمیتی هستند و فیزیکی هستند، این ضرایب، که در واقع از اندازه‌گیری سیستم به دست می‌آیند، باید حقیقی باشند.

اگر بخواهیم از سیستم‌های دوترازی به سیستم‌های d ترازی برویم، طبیعی به نظر می‌رسد که ماتریس‌های پاولی را تعمیم دهیم. ماتریس‌های پاولی به دو روش تعمیم داده می‌شوند که اولی ماتریس‌هایی می‌سازد که هرمیتی و بدون رد هستند (درست مانند ماتریس‌های پاولی) و دومی ماتریس‌های غیر هرمیتی می‌سازد. ما اولی را برای کارمان انتخاب می‌کنیم، چون این ماتریس‌ها هرمیتی هستند و بنابراین فیزیکی‌اند. روش تعمیم‌اشان اینگونه است که چون ماتریس‌های پاولی مولدهای گروه $SU(2)$ هستند، ماتریس‌های تعمیم یافته نیز مولدهای گروه $SU(d)$ باشند. روش ساخت ماتریس‌ها به این صورت است:

$$f_{k,j}^{d} = \begin{cases} E_{kj} + E_{jk} & for\ \ (k < j) \\ -i(E_{jk} - E_{kj}) & for\ \ (k > j) \end{cases}$$

$$h_k^d = \begin{cases} I_d & (k=1) \\ h_k^{d-1} \oplus 0 & (1 < k < d) \\ \sqrt{\frac{2}{d(d-1)}} \left(I_{d-1} \oplus (1-d)\right) & (k=d) \end{cases}$$

که در آن d بعد سیستم است. به ازای $d=2$ ماتریس‌های پاولی بدست می‌آیند و به ازای $d=3$ ماتریس‌ها گلمان $(GellMann\ matrix)$. به ازای بعدهای دیگر ماتریس‌های گلمان تعمیم یافته، ساخته خواهند شد. واضح است که به ازای هر d ما d^2 ماتریس به عنوان پایه‌های فضا به دست خواهیم آورد. یکی از آن‌ها ماتریس واحد است و بنابراین d^2-1 ماتریس گلمان به دست خواهیم آورد[1].

به ازای $d=3$ داریم:

$$g_0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \ , \ g_1 = \begin{pmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \ , \ g_2 = \begin{pmatrix} 0 & 0 & -i \\ 0 & 0 & 0 \\ i & 0 & 0 \end{pmatrix}$$

$$g_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \ , \ g_4 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -2 \end{pmatrix} \ , \ g_5 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -i \\ 0 & i & 0 \end{pmatrix}$$

$$g_6 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \ , \ g_7 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \ , \ g_8 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

دلیل انتخاب این نوع ترتیب، یکسانی آن با کد پایتون مربوطه می‌باشد.

در کد تولید داده‌های یادگیری، یک تابع تعریف شده است که، از کتابخانه $pysme$ برداشته شده است و ماتریس‌ها را در بعد دلخواه برای ما تولید می‌کند. تابع $gellmann(j,k,d)$ سه ورودی می‌گیرد که d همان بعد سیستم است. این تابع، ماتریس گلمان j,k را به ما می‌دهد. سپس در هنگامی که می‌خواهیم ماتریس‌های گلمان را آماده‌سازی کنیم از ضرایب j,k استفاده نخواهیم کرد، بلکه همانند بالا آن‌ها را با یک شماره مرتب می‌کنیم. تابع $gellmann\_basis(d)$ که از تابع بالا استفاده می‌کند و توسط ما تعریف شده است، ماتریس‌های گلمان d بعدی را می‌سازد و درنهایت آن‌ها را تحت عنوان یک دیکشنری $(dictionary)$ ذخیره می‌کند، که کلیدهای آن شماره ماتریس و اعضای آن خود ماتریس‌های گلمان

هستند. علت ذخیره کردن آن تحت عنوان دیکشنری اینست که بعدا کار با این نوع داده برای محاسبه ضرایب، شهودی‌تر و واضح تر است.

با این تعاریف می‌توانیم حالت یک سیستم d حالته را بر حسب ماتریس‌های گلمان تعمیم یافته، بسط دهیم:

$$\rho = \sum_i a_i \, g_i$$

و برای سیستم‌های دو ذره‌ای:

$$\rho = \sum_{i,j} a_{ij} \, g_i \otimes g_j$$

از این به بعد به این بسط، بسط گلمان می‌گوییم.

نحوه ساختن ماتریس چگالی تصادفی:

برای پروژه ما نیاز داریم که تعدادی داده از پیش تعیین شده را به ماشین بدهیم تا از روی داده‌ها یاد بگیرد. برای تولید داده نیاز داریم که ابتدا تعدادی ماتریس تصادفی تولید کنیم تا از روی آن‌ها داده‌های خود را بسازیم.

برای تولید داده‌های مورد استفاده، از ماتریس‌های چگالی تصادفی استفاده شده است که توصیف‌گر حالت‌های دو سیستم سه ترازه است. حالت‌های درهم‌تنیده یا جداپذیر این ماتریس های چگالی مشخص نیست.

برای ساخت این ماتریس‌ها از کتابخانه $Qutip$ پایتون استفاده شده است[2]. که اساس کار این کتابخانه ساخت ماتریس‌های چگالی بر اساس آنسامبل‌های هیلبرت–اشمیت است. برای ساخت این ماتریس‌ها که خواص یک ماتریس چگالی را داشته باشد، لازم است ابتدا یک ماتریس مختلط تصادفی که از آنسامبل $Ginibre$ پیروی می‌کند را انتخاب کرده و با استفاده از رابطه زیر یک ماتریس چگالی هرمیتی, مثبت و نرمال شده به دست می‌آید[3].

$$\square_{hs} = \frac{AA^t}{tr(AA^t)}$$

نحوه بدست آوردن ضرایب بسط گلمان از روی یک ماتریس چگالی تصادفی:

مناسب است که نحوه بدست آوردن ضرایب بسط گلمان را از روی یک ماتریس چگالی تصادفی بدانیم، چراکه برای پروژه، ما ابتدا ماتریس‌های چگالی تصادفی را می‌سازیم و بعد آن را توسط ضرایب بسط گلمان مشخص می‌کنیم.

برای محاسبه ضرایب بسط گلمان از خصوصیت تعامد آن‌ها استفاده می‌کنیم. در حقیقت می‌دانیم که :

$$Tr(g_i g_j) = 2\delta_{ij}$$

حال داریم:

$$\left(\rho = \sum_{i,j} a_{ij}\, g_i \otimes g_j\right) . g_k \otimes g_l$$

$$\rightarrow \quad \rho(g_k \otimes g_l) = \sum_{i,j} a_{ij}\, g_i g_k \otimes g_j g_l$$

حال از طرفین رد می‌گیریم:

$$Tr(g_i g_k \otimes g_j g_l) = Tr(g_i g_k) \otimes Tr(g_j g_l) = 4\delta_{ik} \otimes \delta_{jl}$$

بنابراین داریم:

$$a_{ij} = \frac{1}{4} Tr\left(\rho(g_i \otimes g_j)\right)$$

واضح است که به ازای سیستم ۳×3 بعدی ما 81 ضریب خواهیم داشت که البته یکی از آن‌ها بدیهی است. چراکه به ازای ماتریس‌های واحد، ضریب برابر 0.25 است.

$$a_{88} = \frac{1}{4} Tr(\rho(I \otimes I)) = \frac{1}{4} Tr(\rho) = 0.25$$

تشخیص اولیه حالت‌های درهم‌تنیده و جداپذیر:

برای اینکه داده‌هایی که آماده می‌کنیم، برچسب مناسب برای مسئله موردنظر را کسب کنند، نیاز داریم تعیین کنیم که داده‌های ما جداپذیر هستند یا درهم‌تنیده. البته که تعیین دقیق این حالت‌ها خود جواب مسئله است، ولی می‌توان با روش‌های موجود تعدادی را مشخص کرد و تعدادی نیز نامشخص باقی می‌مانند.

در اینجا برای تشخیص اولیه حالت‌های درهم‌تنیده, از میان روش های بسیاری که وجود دارند، روش پیرز–هورودوکی $(positive\ partial\ transpose - PPT)$ انتخاب شده است تا برای برچسب زدن اولیه داده‌ها، به طور دقیق حالت‌های درهم‌تنیده معین شود.

تعریف *PPT*: [4]

روش پیرز– هوردوکی روش مهمی برای تشخیص درهم‌تنیدگی ماتریس‌هایی که معین کننده دو زیرسیستم A, B هستند، می‌باشد. برای ماتریس‌های با ابعاد 2×3 و 2×2 استفاده از این روش برای تشخیص حالت‌های جدا پذیر و درهم‌تنیده کافی است و جواب مسئله را به طور دقیق معین می‌کند. اما برای ابعاد بالاتر، از این روش به عنوان شرط کافی نمی‌توان استفاده کرد و برای تعیین دقیق حالت‌های جداپذیر باید از روش‌هایی چون شاهدها $(Witness)$ استفاده کرد[5].

اگر عملگر چگالی رو فضای $\mathcal{H}_A \otimes \mathcal{H}_B$ عمل کند، داریم:

$$\square = \sum_{ijk} p_{kl}^{ij} |i\rangle\langle j| \otimes |k\rangle\langle l|$$

برای حالت‌های جداپذیر می‌توان ماتریس چگالی را اینگونه نوشت :

$$\square = \sum_{i} p_i \, \square_i^A \otimes \square_i^B$$

که در آن:

$$p_i > 0 \quad , \quad \sum_i p_i = 1$$

این یک ترکیب محدب $(Convex\ Combination)$ از عملگر چگالی‌های زیر سیستم است.

با اعمال ترانهاده جزیی بر روی حالت B داریم :

$$\square^{T_B} = \sum_i p_i \square_i^A \otimes (\square_i^B)^T$$

این نکته لازم به ذکر است که اگر ترانهاده جزیی بر روی زیرسیستم A نیز انجام می‌شد نتیجه مشابهی به همراه داشت.

حال چون می‌دانیم تمامی ویژه مقادیر ماتریس چگالی زیرسیستم B حتی در صورت ترانهاده شدن، همان ویژه مقادیر قبل هستند، پس انتظار می‌رود که ویژه مقادیر ماتریس چگالی کل بعد از اعمال ترانهاده جزیی کماکان مثبت باقی بماند و در صورتی که این ویژه مقادیر منفی شود نمایانگر این است که فرض اولیه برای نوشتن ماتریس چگالی به عنوان ترکیب محدب ماتریس های چگالی زیرسیستم‌های A, B (فرض جدا پذیر بودن) نقض می‌شود و می‌توان نتیجه گرفت که حالت‌های موجود درهم‌تنیده هستند[4]. اما در صورت مثبت بودن ضرایب، چون PPT شرط کافی نیست، (برای سیستم‌های با ابعاد بیشتر از 2×3) نمی‌توان نتیجه گرفت که حالت مد نظر جداپذیر است و حالت ناشناخته باقی می‌ماند.

برای پیاده سازی این الگوریتم در برنامه، یک تابع PPT را تعریف کردیم. سپس با استفاده توابع موجود در کتابخانه $Qutip$ از قبیل $partial\_transpose$ از ماتریس چگالی تصادفی خود یک ترانهاده جزیی گرفتیم (نسبت به یکی از زیر سیستم‌ها). واضح است که برای الگوریتم فوق، فرقی ندارد که نسبت به کدام زیر سیستم ترانهاده جزیی بگیریم. سپس این ماتریس چگالی بدست آمده یک $Qobj$ است که در واقع یکی از اشیای کتابخانه $Qutip$ است. این شی یک متد دارد به نام $eigenstates$ که خروجی آن به صورت یک لیست است که خود شامل دو لیست درونی است. لیست اول ویژه‌مقادیر مربوط به ترانهاده جزیی ماتریس چگالی است. این لیست خود به خود از کوچک به بزرگ مرتب شده است و بنابراین کافی است که فقط عضو ابتدایی آن مشخص شود. اگر منفی بود که حالت درهم‌تنیده است و در غیر اینصورت ناشناخته باقی می‌ماند.

همچنین برای ساختن ماتریس‌های چگالی تصادفی جدایی پذیر ما از تعریف استفاده کردیم. از آنجا که هر حالت جدایی‌پذیر را می‌توان بر حسب یک ترکیب محدب از زیر سیستم‌های مختلف نوشت، ما می‌توانیم تعدادی از این ماتریس‌ها را بسازیم. ابتدا نیاز داریم که تعدادی ماتریس چگالی تصادفی بسازیم که بُعد آن‌ها برابر با بُعد زیرسیستم‌ها باشد. سپس باید تعدادی از آن‌ها را به صورت تصادفی انتخاب کنیم. اینکار را تابع $choice$ از کتابخانه $random$ انجام می‌دهد. حال باید تعدادی عدد مثبت بیابیم که مجموع همه آن‌ها برابر 1 باشند. در حقیقت این اعداد برای ما نقش ضرایب احتمال را بازی می‌کنند. مناسب است که از تابع $np.random.dirichlet$ استفاده کنیم که در واقع در کتابخانه $numpy$ یافت می‌شود. بعد از تمام این‌ها کافی است که ماتریس‌ها را به ترتیب در هم ضرب تانسوری کرده و با اعداد تصادفی تولید شده ضرب کرده و در نهایت همه را با هم جمع بزنیم. برای این منظور ما تابع $Convex\_Combination$ را در کد خود تعریف کردیم.

برای مشخص کردن حالات روش دیگری موسوم به روش CHA $(Convex\ Hull\ approximation)$ وجود دارد که توضیحات مربوط به آن در ادامه می‌آید.

پیاده‌سازی الگوریتم تشخیص درهم‌تنیدگی بدون استفاده از شاهدها یا PPT:

برای درک این الگوریتم و پیاده‌سازی آن از مقاله‌ی [6] بهره بردیم.

در اولین قدم برای آن که بتوانیم از روش‌های یادگیری ماشین برای تشخیص درهم‌تنیدگی سیستم استفاده کنیم، لازم است تعداد زیادی ماتریس چگالی تصادفی تولید کرده و با الگوریتم‌هایی مشخص کنیم که این حالت‌ها درهم‌تنیده هستند یا جدایی‌پذیر. سپس ماشین را با این ماتریس‌ها $learn$ می‌کنیم تا بتواند حالت‌های نامشخص برای ما را هم تشخیص دهد.

باید ابتدا ماتریس‌های تصادفی را برچسب بزنیم تا بتوانیم فرآیند یادگیری را شروع کنیم. برای این منظور از این الگوریتم استفاده کردیم:

می‌دانیم حالت‌های جدایی‌پذیر یک $convex\ hull$ تشکیل می‌دهند که مرز آن حالت‌های خالص جدایی‌پذیر هستند[7].

بنابراین ابتدا با مجموعه‌ای از حالت‌های خالص جدایی‌پذیر، یک $convex\ hull$ ساختیم برای آن که بتوانیم به گونه‌ای این زیرفضا را شبیه‌سازی کنیم. بعد از آن هر ماتریس چگالی تصادفی را با بررسی این که آیا در این $convex\ hull$ هست یا خیر برچسب می‌زنیم. بودن یک حالت در این $convex\ hull$ معادل جدایی‌پذیر بودن آن است.

سیستم متشکل از دو کیوتریت:

لازم به ذکر است که ما در فضای $Feature\ vector$ های هر ماتریس چگالی این کار را انجام می‌دهیم. $Feature\ vector$ های حالت‌های مختلف، همان ضرایب بسط گلمان به دست آمده برایشان هستند. بُعد این فضا برای یک سیستم دو کیوتریتی ۸۱ است.

برای آن‌که بتوانیم یک $convex\ hull$ بسازیم، پس از تولید ضرایب بسط گلمان ۸۲ ماتریس چگالی خالص، از توابع $convex\ hull$ و $Delaunay$ در کتابخانه‌ی $Scipy$ استفاده کردیم.

با استفاده از تابعی که تشخیص می‌دهد آیا نقطه‌ای در یک $convex\ hull$ هست یا نه، تلاش کردیم سیستم دوکیوتریتی را مورد بررسی قرار دهیم اما دچار مشکلاتی شد که نتوانستیم برنامه را اجرا کنیم.

سیستم متشکل از دو کیوبیت:

به دلیل مشکل پیش آمده تصمیم گرفتیم ابعاد سیستم را کم کنیم تا بتوانیم کد را تست کنیم، بنابراین با دو کیوبیت کار کردیم. فضای حالت در این صورت 16 بعدی است.

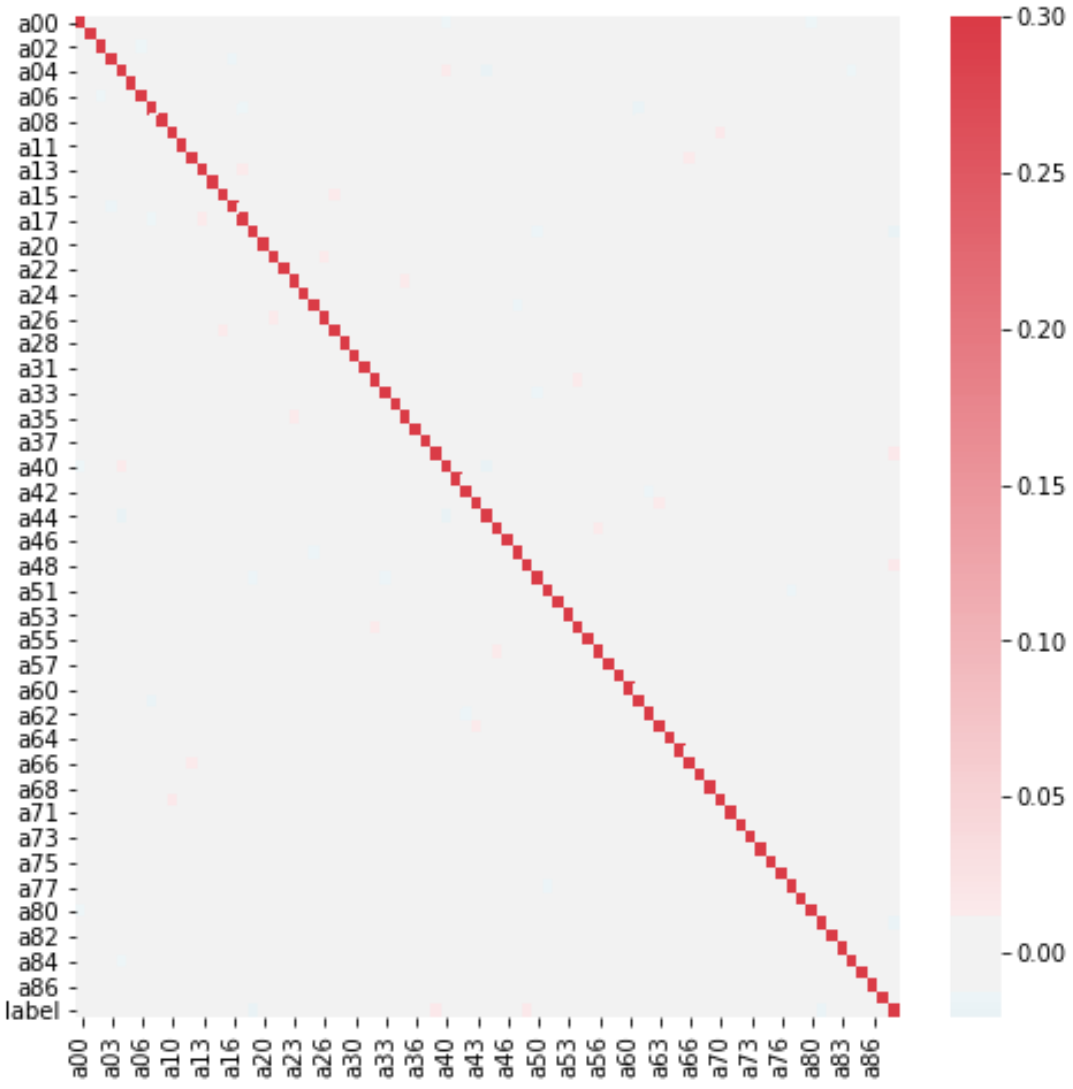
تعدادی ماتریس چگالی خالص 4 در 4 تولید کردیم و ضرایب بسط آن‌ها برحسب ماتریس‌های پائولی را به دست آوردیم.

$convex\ hull$ را با استفاده از $Feature\ vector$ این ماتریس‌ها ساختیم و با تابع $Delaunay$ بررسی کردیم که آیا یک حالت آزمایشی شناخته شده در $convex\ hull$ هست یا خیر. خروجی تابع نهایی $true$ برای حالت جدایی‌پذیر و $false$ برای حالت درهم‌تنیده است.

بررسی همبستگی مقادیر اندازه‌گیری شده داده‌ها:

برای 200 000 دیتا فریم به دست آمده که شامل ضرایب بسط ماتریس‌های گلمان هستند و برچسب گذاری شده‌اند، تابع همبستگی را چک کرده و نمایش داده‌ایم.

با توجه به $heatmap$ رسم شده می‌توان دریافت که همبستگی بین ضرایب و برچسب گذاری‌ها صفر است. همچنین با افزایش تعداد داده‌های مورد بررسی می‌توان مشاهده کرد که مطابق انتظار به طور کامل این همبستگی صفر می‌شود.



تصویر 1 : در این تصویر، که یک نقشه حرارتی از میزان همبستگی ویژگی‌های داده‌ها با هم می‌باشد، مشاهده می‌کنید که هیچ ویژگی از داده‌ها با هم همبستگی ندارد و از آن مهم‌تر اینکه ویژگی‌های داده‌ها با $label$ ها نیز همبستگی ندارد.

داده های برچسب گذاری شده :

لینک dropbox زیر حاوی 1 میلیون و 100 هزار داده برچسب گذاری شده است .

https://www.dropbox.com/sh/ciwxuttoxbox4dq/AAAl4rc4O4LPalQDwigRgtTDa?dl=0

منابع:

[0] Phys. Rev. A 70, 060303(R) (2004)

[1] https://en.wikipedia.org/wiki/Generalizations_of_Pauli_matrices

[2] qutip.org/docs/4.1/apidoc/functions.html

[3] Karol ˙Zyczkowski, Generating random density matrices.5

[4] A. Peres, Phys. Rev. Lett. **76** (1997) 1413.

[5] M. Horodecki, P. Horodecki and R. Horodecki Phys. Lett. A 22 (1996) 1.

[6] Sirui Lu,A Separability-Entanglement Classifier via Machine Learning. (2017)

[7] M. A. Jafarizadeh,Two-qutrit Entanglement Witnesses and Gell-Mann Matrices. (2008)