باسمه تعالی

فاز اول پروژه‌ی درس مقدمه ای بر یادگیری ماشین

مدل های گرافی و ماشین بولتزمن

استاد درس

دکتر محمّدحسین یاسائی میبدی

دانشکده‌ی مهندسی برق
دانشگاه صنعتی شریف

پاییز ۱۴۰۳

آخرین مهلت تحویل:
۱۷ دی ۱۴۰۳

فهرست مطالب

حتما قبل از شروع پروژه بخش ۵ را مطالعه نمایید.

۱ مدل‌های گرافی

مدل های گرافی ابزارهای قدرتمندی برای توصیف توزیع‌های احتمالی هستند. این مدل‌ها ارتباط بین متغیرها را به خوبی بیان میکنند و به طور خاص برای بررسی این ارتباطات مناسب هستند . زیرا این امکان را میدهند تا با ابزار های نظریه گراف این همبستگی ها و ارتباطات را تحلیل کنیم. همچنین میتوان محاسبات پیچیده را با الگوریتم های گراف با محاسبات کمتری انجام دهیم. انواع متفاوتی از این مدل‌ها وجود دارند برای آشنایی با این مدل ها ابتدا با چند تعریف در گراف آشنا میشویم.

۱.۱ چند تعریف

۱.۱.۱ گراف بدون جهت

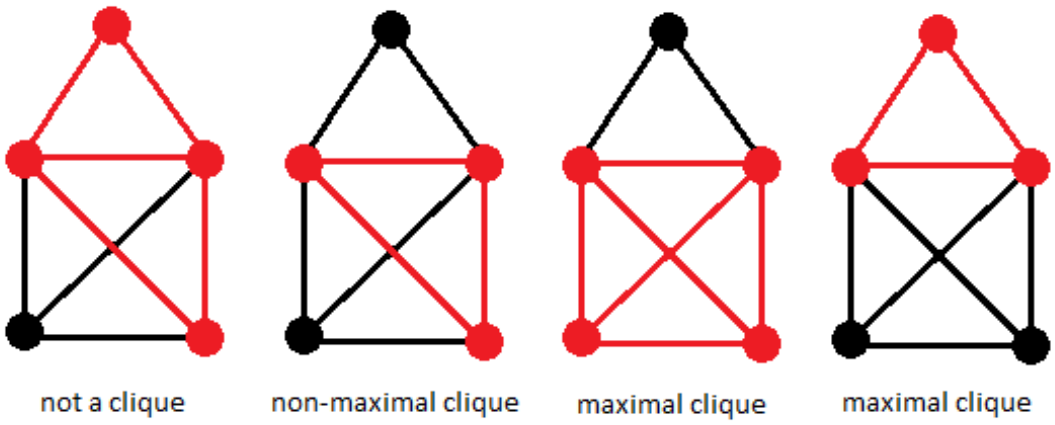
گراف بدون جهت به تاپل دوتایی $G = (V, E)$ میگوییم که در V نشان دهنده گره ها و E نشان دهنده یال هاست به گونه ای که اگر یالی بین دو نود v, w داشته باشیم داریم $v, w \in E$ و میگوییم w در همسایگی v است و بالعکس. برخلاف گراف بدون جهت در گراف با جهت ترتیب w, v اهمیت دارد چون هر یال جهت نیز دارد اما در گراف بدون جهت، جهت نداریم.

۲.۱.۱ گروه خاص (clique)

به زیر مجموعه ای از گره‌های یک گراف که که یک گراف کامل را تشکیل میدهند گروه خاص میگوییم.

۳.۱.۱ گروه خاص ماکسیمال (Maximal clique)

گروه خاص ماکسیمال به گروه خاصی میگوییم که هیچ گره ی دیگری در گراف نباشد که با اضافه کردنش گروه همچنان خاص بماند. در شکل ۱ مثال هایی را مشاهده می‌کنید.



شکل ۱: Maximal clique

مجموعه تمام گروه‌های خاص ماکسیمال یک گراف را با C نشان میدهیم.

۴.۱.۱ مسیر

به دنباله‌ای از یال‌های $v_1, v_2, v_3, \ldots, v_m \in V$ که برای $i = 1, 2, 3, \ldots, m-1$ داشته باشیم $\{v_i, v_{i+1}\} \in E$ یک مسیر از v_1 به v_m می‌گوییم. به مجموعه گره‌های $\mathcal{V} \subset V$ جداکننده دو گره $v \notin$

$\mathcal{V}, w \notin \mathcal{V}$ می‌گوییم اگر هر مسیر از v به w شامل گره‌ای از $\mathcal{V}$ باشد.

۲.۱ مدل گرافی

مدل های گرافی توزیع های احتمالاتی و روابط وابستگی بین آنها را با یک ساختار گراف بیان میکنند، به گونه ای که هر گره گراف نماینده یک متغیر تصادفی است و یال ها بیانگر وابستگی بین متغیر ها هستند. انواع متفاوتی از مدل های گرافی وجود دارند که در این جا با میدان تصادفی مارکف آشنا میشوید.

۱.۲.۱ متغیر های مستقل شرطی (conditionally independent variables)

میگوییم دو متغیر $X_۱, X_۲$ به شرط $X_۳$ مستقل هستند هرگاه داشته باشیم.

$$p(X_۱, X_۲ \mid X_۳) = p(X_۱ \mid X_۳) \cdot p(X_۲ \mid X_۳)$$

۲.۲.۱ فضای حالت

فضای حالت در یک گراف مارکوف به مجموعه‌ای از تمام حالات ممکن که یک سیستم می‌تواند در آنها باشد اشاره دارد. در یک گراف، متغیرهای تصادفی به هر گره اختصاص داده می‌شوند و فضای حالت شامل تمامی ترکیب های ممکنی است که برای این متغیرهای تصادفی قابل تصور است. توجه شود فضای حالت بسیار به فضای نمونه ای نزدیک است فضای حالت یک مفهوم کلی تر از فضای نمونه ای است و بیانگر تمام مقادیری است که برای یک متغیر و یا حالت سیستم قابل تصور است در فضای نمونه ای علاوه بر حالت ها یک تابع باید تعریف شود تا به هر حالت ممکن یک احتمال نسبت دهد.

۳.۲.۱ میدان تصادفی مارکف

فرض کنید به هر گره از گراف $G = (V, E)$ یک متغیر تصادفی نسبت دهیم که مقادیری از فضای حالت Λ_v را اختیار میکند. برای سادگی فرض میکنیم تمام متغیر ها فضای حالت معادل Λ دارند. متغیرهای تصادفی $\mathbf{X} = (X_v)_{v \in V}$ را میدان احتمالاتی مارکف مینامیم اگر توزیع احتمالاتی مشترک p خاصیت جهانی مارکف را نسبت به گراف براورده کند. این یعنی برای هر زیر مجموعه $\mathcal{A}, \mathcal{B}, \mathcal{S} \subset V$ که در آن تمام گره های $\mathcal{A}$ و $\mathcal{B}$ توسط $\mathcal{S}$ از هم جدا شده اند داشته باشیم: $\mathbf{X} = (X_a)_{a \in \mathcal{A}}$ و $\mathbf{X} = (X_b)_{b \in \mathcal{B}}$ با داشتن $\mathbf{X} = (X_s)_{s \in \mathcal{S}}$ به صورت شرطی مستقل اند. یا به عبارتی داریم :

$$p\left((x_a)_{a \in A} \mid (x_t)_{t \in S \cup B}\right) = p\left((x_a)_{a \in A} \mid (x_t)_{t \in S}\right)$$

به تعبیری دیگر در یک میدان مارکف توزیع گره v به شرط همسایه هایش مستقل از بقیه گره هاست.

۴.۲.۱ پوشش مارکف

مجموعه‌ای از گره‌ها $\mathrm{MB}(v)$ را پوشش مارکوف گره v می‌نامند، اگر برای هر مجموعه گره‌های B با $v \notin B$ داشته باشیم

$$p(v \mid \mathrm{MB}(v), B) = p(v \mid \mathrm{MB}(v)).$$

در یک میدان مارکف همسایه های گره v پوشش مارکف آن هستند.

۵.۲.۱ توزیع به صورت سخت مثبت

یک توزیع احتمال p به صورت سخت مثبت است اگر و تنها اگر برای هر x در فضای حالت Λ داشته باشیم:

$$p(x) > ۰$$

به عبارت دیگر، این بدان معناست که هیچ مقدار از x وجود ندارد که احتمال آن صفر باشد و تمامی حالت‌های ممکن دارای احتمال غیرصفر هستند.

Hammersley-Clifford Theorem ۶.۲.۱

یک توزیع نسبت به یک گراف بدون جهت G با گروه های خاص ماکسیمال C فاکتوریزه می‌شود اگر مجموعه‌ای از توابع غیرمنفی $\{\psi_C\}_{C\subset C}$ که توابع پتانسیل نامیده می‌شوند وجود داشته باشند به طوری که

$$\forall \mathbf{x}, \hat{\mathbf{x}} \in \Lambda^{|V|} : (\mathbf{x}_c)_{c\in C} = (\hat{\mathbf{x}}_c)_{c\in C} \Rightarrow \psi_C(\mathbf{x}) = \psi_C(\hat{\mathbf{x}})$$

و

$$p(\mathbf{x}) = \frac{1}{Z}\prod_{c\in C}\psi_c(\mathbf{x}) = \frac{1}{Z}\prod_{c\in C}\psi_c(\mathbf{x_c})$$

در اینجا: $\mathbf{x}_c$ گره‌هایی از $\mathbf{x}$ هستند که جزو گروه خاص c هستند - $\psi_C(\mathbf{x})$ توابع پتانسیل هستند. - Z ثابت نرمال‌سازی است. - $\Lambda^{|V|}$ فضای حالت است. و $p(\mathbf{x})$ توزیع مشترک تمام متغیر های تصادفی مربوط به گره های G هستند. به طور کلی، فاکتوریزه کردن یک توزیع احتمال به معنای تجزیه آن به ضرب تعدادی توابع کوچکتر است که به بخش‌های مختلف یک گراف مربوط می‌شوند. این فرآیند باعث ساده‌تر شدن محاسبات و فهم بهتر ساختار توزیع می‌شود.

ثابت نرمال‌سازی Z به صورت

$$Z = \sum_x \prod_{c\in C}\psi_c(x_c)$$

است و تابع پارتیشن نامیده می‌شود. اگر p به صورت سخت مثبت باشد، این موضوع برای توابع پتانسیل نیز صادق است. بنابراین می‌توانیم بنویسیم:

$$p(x) = \frac{1}{Z}\prod_{c\in C}\psi_c(x_c) = \frac{1}{Z}\exp\left(\sum_{c\in C}\ln\psi_c(x_c)\right) = \frac{1}{Z}\exp(-E(x))$$

که در اینجا $E := \sum_{c\in C}\ln\psi_c(x_c)$ به عنوان تابع انرژی نامیده می‌شود. بنابراین، توزیع احتمال هر میدان تصادفی مارکوف (MRF) می‌تواند به شکل ارائه شده در بالا بیان شود که به آن توزیع گیبس نیز می‌گویند.

پرسش تئوری ۱. سعی کنید درباره مفهوم توابع پتانسیل در یک میدان مارکف کمی توضیح بدهید و یک مثال از یک میدان مارکف به همراه توابع پتانسیل اش بزنید. توجه کنید مثال میتواند بسیار ساده باشد اما حتما باید مقادیر عددی تعیین گردد و توضیحات هم تا حد ممکن کوتاه و هدفمند باشند.

Unsupervised Learning ۳.۱

به یادگیری (ویژگی های مهمی) از یک توزیع نامعلوم بر اساس سمپل های آن Unsupervised learning میگوییم. یادگیری هرگونه ارتباط در بین داده ها به طور مثال کاهش بعد، دسته بندی داده های نزدیک به هم و کشف الگو ها در دیتا نمونه هایی از Unsupervised learning هستند.

مسئله در میدان تصادفی مارکف ۱.۳.۱

فرض کنید تعدادی متغیر تصادفی دارید که فضای حالت یکسانی دارند و یک میدان تصادفی مارکف تشکیل می‌دهند. چند نمونه از حالت‌های سیستم دارید. یک نمونه می‌تواند به صورت زیر باشد:

$$x_{v_1}, x_{v_2}, \ldots, x_{v_n}$$

همچنین فرض کنید ساختار کلی گراف را هم داریم. حال اگر فرض کنیم که توابع انرژی گراف می‌توانند از خانواده خاصی باشند که این خانواده از توابع با پارامتر Θ پارامتریزه شده‌اند. می‌توان با استفاده از نمونه‌ها، پارامترها را یاد گرفت.

پرسش تئوری ۲. ۳ نوع از توابع پتانسیل را نام برده و پارامتر های آنها را مشخص کنید.

ML ۲.۳.۱

برای پیدا کردن پارامتر ها از قائده درست نمایی بیشینه استفاده میکنیم. با فرض داده های مستقل از هم به صورت زیر :

$$S = \{\mathbf{x_1}, \ldots, \mathbf{x_m}\}$$

،

$$\mathbf{x_1} = \{x_{v1}, \ldots, x_{vn}\}$$

داریم:

$$L(\theta \mid S) = \prod_{i=1}^{m} p(\mathbf{x_i} \mid \theta)$$

عبارت بالا احتمال مشاهده دیتاست به شرط پارامترهاست.
توجه کنید که تنها $\mathbf{x_i}$ ها از هم مستقل اند و در یک سمپل مقادیر گره های مختلف باهم همبستگی دارند و هدف ما یافتن همین همبستگی ها میباشد.

پرسش تئوری ۳. به علت اینکه پیدا کردن آنالیزی پارامترها برای بیشینه سازی عبارت بالا به طور عمومی ممکن نیست در خیلی از مواقع از روش $GradientAscent$ استفاده میشود . در کمتر از سه خط این روش را توضیح بدهید.

Latent Variables ۳.۳.۱

پیشنهاد میشود قبل از شروع این بخش **این ویدیو** را ببینید.
همانطور که مشاهده کردید برای مدل کردن یک توزیع مجهول با m متغیر میتوان گرافی با تعداد بیشتری گره در نظر گرفت که m گره در آنها مشاهده پذیر اند. اگر $\mathbf{X} = (X_v)_{v \in V}$ متغیر های مربوط به یک میدان تصادفی مارکف باشند که یک زیر مجموعه m تایی از آنها یعنی $\mathbf{V} = (V_1, ..V_m)$ متغیر های مشاهده پذیر اند، به متغیر های باقی مانده $\mathbf{H} = (H_1, ..H_m)$ متغیر های Latent یا پنهان میگوییم. در این شرایط توزیع مشترک میدان مجهول مارکف به صورت زیر است:

$$p(v) = \sum_h p(v, h) = \frac{1}{Z} \sum_h e^{-E(v,h)}$$

$$Z = \sum_{v,h} e^{-E(v,h)}$$

همانطور که در ویدیو مشاهده کردید علت استفاده از متغیر های Latent مدل کردن و کشف ارتباطات پیچیده تر بین متغیر های مشاهده پذیر است.

محاسبه گرادیان لگاریتم درست نمایی میدان تصادفی کارکف با متغیر پنهان ۴.۳.۱

لگاریتم درست نمایی برای میدان تصادفی مارکف با متغیر پنهان به صورت زیر است:

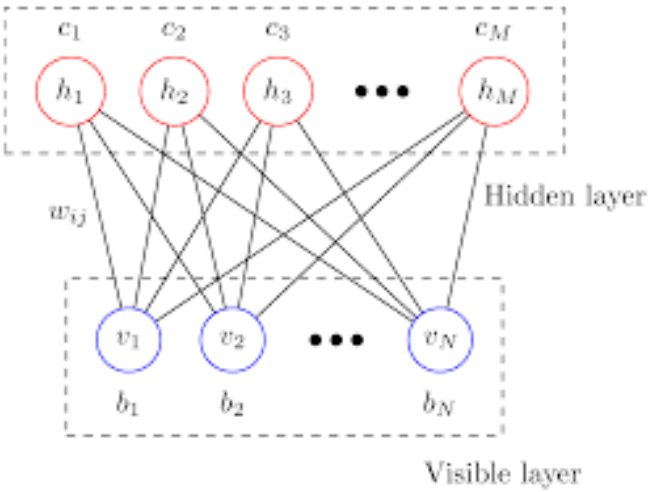
$$\ln L(\theta \mid v) = \ln p(v \mid \theta) = \ln \frac{1}{Z} \sum_h e^{-E(v,h)} = \ln \sum_h e^{-E(v,h)} - \ln \sum_{v,h} e^{-E(v,h)}$$

پرسش تئوری ۴. نشان دهید:

$$\frac{\partial \ln L(\theta \mid v)}{\partial \theta} = -\sum_{h} p(h \mid v) \frac{\partial E(v,h)}{\partial \theta} + \sum_{v,h} p(v,h) \frac{\partial E(v,h)}{\partial \theta}$$

۲ ماشین بولتزمن

در این پروژه قصد داریم که با ساختار ماشین بولتزمن Boltzmann Machines آشنا بشویم.
ماشین های بولتزمن سعی بر این امر دارد که بتواند توزیع نمونه های ورودی را تخمین بزند و اینگونه می تواند در بخش های مختلف یادگیری ماشین استفاده بشود. ابتدا با ساختار آن آشنا می شویم تا درک بهتری از نوع عملکرد آن داشته باشیم.
ماشین بولتزمن شامل دو لایه نورون هست که شامل یک لایه نمایان بوده و یک لایه پنهان.(مطابق شکل زیر)



همانطور که می بینید تمام نورون های لایه نمایان به تمام نورون های لایه پنهان متصل هستند و بالعکس و هیچ دو نورون در یک لایه به یکدیگر متصل نیستند که این باعث مستقل شدن نورون های پنهان به شرط یک نورون نمایان می شود.
این لایه پنهان سعی بر این دارد که توزیع ورودی را تخمین بزند.بدین معنی که با استخراج ویژگی های مهم از ورودی می تواند توزیع ورودی را بسازد.

پرسش تئوری ۵. با توجه به توضیحات داده شده در مورد نحوه عملکرد ماشین بولتزمن، این ساختار در چه بخش هایی از یادگیری ماشین می تواند استفاده شود؟

حال بایستی یک توزیع مشترک بین تمام حالاتی که نورون های نمایان و پنهان نسبت بدهیم.(توجه کنید که مقدار تمام نورون ها ۰ یا ۱ می باشد)
به همین علت از توزیع گیبس Gibbs distribution استفاده می کنیم که فرمول آن به صورت زیر می باشد.

$$p(v,h) = \frac{1}{Z} e^{-E(v,h)}$$

v متناظر با نورون های نمایان و h متناظر با نورون های پنهان می باشد.

$$E(v,h) = -\sum_{i=1}^{n} \sum_{j=1}^{m} w_{ij} h_i v_j - \sum_{j=1}^{m} b_j v_j - \sum_{i=1}^{n} c_i h_i.$$

که w_ij وزن تخصیص داده شده به یال بین نورون v_j و h_i می باشد و b_j و c_i بایاس های متناظر با نورون نمایان i و نورون پنهان j می باشد.
همانطور که می دانید برای آموزش دادن مدل بایستی بتوانیم یک سری گرادیان و عبارات را حساب کنیم که در این بخش به آن می پردازیم.

پرسش تئوری ۶. ثابت کنید:

$$p(H_i = ۱ \mid \mathbf{v}) = \sigma\left(\sum_{j=۱}^{m} w_{ij} v_j + c_i\right)$$

و

$$p(V_j = ۱ \mid \mathbf{h}) = \sigma\left(\sum_{i=۱}^{n} w_{ij} h_i + b_j\right).$$

در این قسمت می خواهیم بحث کنیم که توزیع هایی که در پرسش قبل بدست آوردید به چه درد می خورد. در بخش بعدی به طور مفصل نحوه به روزرسانی پارامتر ها را فراخواهید گرفت اما در این قسمت مفاهیم اولیه آن را با یکدیگر مرور خواهیم کرد.

در بخش بعد خواهید دید که ما نیازداریم با وجود دانستن h مقدار نورون v و بالعکس را به صورت تصادفی با توزیعی که بدست آوردید مقدار دهی کنیم . طبیعتا بایستی به گونه ای نمونه برداریم که توزیع نمونه های به وجود امده نزدیک به توزیع بدست آمده باشد و برای این مهم از مفاهیم پایه فرآیند های مارکوف و نمونه برداری استفاده می کنیم که شرح زیر می باشد.

فرایندهای مارکوف و نمونه‌برداری گیبس

فرایندهای مارکوف

فرایند مارکوف یک فرآیند تصادفی است که در آن حالت فعلی سیستم تنها به حالت قبلی وابسته است و مستقل از توالی حالت‌های گذشته است. به طور دقیق، اگر $\{X(k) \mid k \in \mathbb{N}_۰\}$ یک زنجیره مارکوف باشد، آنگاه برای همه $k \geq ۰$ و حالت‌های i, j داریم:

$$P(X(k+۱) = j \mid X(k) = i, X(k-۱), \ldots, X(۰)) = P(X(k+۱) = j \mid X(k) = i).$$

این ویژگی که به عنوان خاصیت مارکوف شناخته می‌شود، به این معناست که آینده تنها به وضعیت فعلی بستگی دارد و از تاریخچه گذشته مستقل است.

اگر احتمال انتقال از یک حالت به حالت دیگر ثابت باشد، زنجیره مارکوف همگن نامیده می‌شود. در این حالت، ماتریس انتقال P تعریف می‌شود که در آن عنصر p_{ij} احتمال انتقال از حالت i به حالت j است.

زنجیره‌های مارکوف ویژگی‌های مهمی دارند که برای الگوریتم‌های نمونه‌برداری بسیار مفید هستند:

- **ایستایی (Stationarity):** یک توزیع π به عنوان توزیع پایا تعریف می‌شود اگر:

$$\pi^T = \pi^T P,$$

به این معنا که اعمال ماتریس انتقال بر روی π تغییری در آن ایجاد نمی‌کند. اگر زنجیره مارکوف برای زمان کافی اجرا شود، به این توزیع پایا همگرا خواهد شد.

- **همگرایی:** اگر زنجیره مارکوف غیرقابل کاهش (یعنی از هر حالت بتوان به هر حالت دیگری رسید) و غیرتناوبی باشد، توزیع حالت‌های زنجیره در نهایت به توزیع پایا نزدیک می‌شود، مستقل از توزیع اولیه.

نمونه‌برداری گیبس

نمونه‌برداری گیبس یکی از روش‌های نمونه‌برداری مونت کارلو مبتنی بر زنجیره مارکوف (MCMC) است که برای تولید نمونه‌هایی از توزیع‌های احتمال پیچیده مانند توزیع گیبس در ماشین بولتزمن استفاده می‌شود.

ایده اصلی این روش به شرح زیر است:

- فرض کنید توزیع مشترک متغیرهای تصادفی $\{X_۱, X_۲, \ldots, X_N\}$ به صورت $\pi(x) = \frac{۱}{Z} e^{-E(x)}$ تعریف شده است، که در آن $E(x)$ انرژی سیستم و Z یک مقدار نرمال‌سازی است.
- به جای نمونه‌برداری مستقیم از $\pi(x)$، که ممکن است پیچیده باشد، هر متغیر X_i به نوبت و بر اساس توزیع شرطی $\pi(X_i \mid X_{-i})$ به‌روزرسانی می‌شود، که در آن X_{-i} مجموعه سایر متغیرها است.

در هر گام از الگوریتم:

- یک متغیر X_i انتخاب می‌شود.
- مقدار جدید X_i از توزیع شرطی $P(X_i \mid X_{-i})$ نمونه‌برداری می‌شود، که به طور مؤثر احتمال تغییر این متغیر را به حالت‌های مختلف در نظر می‌گیرد.

چرا از نمونه‌برداری گیبس استفاده کنیم؟

در مدل‌های پیچیده‌ای مانند ماشین بولتزمن محدود (RBM)، توزیع احتمال نهایی سیستم (توزیع گیبس) معمولاً بسیار پیچیده است و نمی‌توان مستقیماً از آن نمونه‌برداری کرد. برای مثال، توزیع گیبس به صورت زیر تعریف می‌شود:

$$P(v,h) = \frac{1}{Z} e^{-E(v,h)},$$

که در آن $E(v,h)$ تابع انرژی سیستم است. برای آموزش مدل و یا ارزیابی آن، نیاز داریم نمونه‌هایی از این توزیع به دست آوریم.
زنجیره مارکوف و روش نمونه‌برداری گیبس ابزارهایی را برای این کار فراهم می‌کنند:

- **ساخت یک زنجیره مارکوف:** ابتدا زنجیره‌ای طراحی می‌کنیم که توزیع پایای آن همان توزیع گیبس باشد.
- **تضمین همگرایی:** اگر زنجیره مارکوف غیرقابل کاهش و غیرتناوبی باشد، تضمین می‌شود که با گذشت زمان به توزیع گیبس همگرا شود.
- **بازنمایی داده‌ها:** نمونه‌هایی که از زنجیره مارکوف پس از همگرایی به دست می‌آیند، به طور موثری نماینده توزیع گیبس هستند و می‌توانند برای آموزش یا تحلیل مدل استفاده شوند.

به بیان ساده، نمونه‌برداری گیبس به ما این امکان را می‌دهد که از پیچیدگی توزیع اصلی عبور کرده و نمونه‌هایی تولید کنیم که ساختار مدل را به خوبی بازنمایی می‌کنند. این روش نه تنها محاسبات را ساده می‌کند، بلکه امکان رسیدن به حالت پایدار را نیز فراهم می‌آورد.

پرسش تئوری ۷. برای به روز رسانی پارامتر ها، نیاز داریم مشتق های جزئی تابع هدف را نسبت به پرامتر ها حساب کنیم.
مشتق های جزئی زیر را بدست آورده و تا حد امکان آنها را ساده کنید.

- $\frac{\partial}{\partial w_{ij}} \ln \mathcal{L}(\boldsymbol{\theta} \mid \boldsymbol{v})$
- $\frac{\partial}{\partial b_j} \ln \mathcal{L}(\boldsymbol{\theta} \mid \boldsymbol{v})$
- $\frac{\partial}{\partial c_i} \ln \mathcal{L}(\boldsymbol{\theta} \mid \boldsymbol{v})$

پرسش تئوری ۸. نشان دهید پاسخ شما به صورت زیر میتواند نوشته شود:

$$\frac{\partial \ln \mathcal{L}(\theta|v)}{\partial w_{ij}} = \frac{1}{l} \sum_{v \in \mathcal{S}} [\mathbb{E}_{p(h|v)}[v_i h_j] - \mathbb{E}_{p(h,v)}[v_i h_j]] = \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \quad (۱)$$

---

۳ تقریب گرادیان، حل مشکل

در قسمت های قبل گرادیان را نسبت به پارامتر های مدل محاسبه کردید، به طور مثال برای وزن w_{ij} در ماشین بولتزمن، خواهیم داشت.

$$\frac{\partial \ln \mathcal{L}(\theta|v)}{\partial w_{ij}} = \frac{1}{l}\sum_{v\in\mathcal{S}}[\mathbb{E}_{p(h|v)}[v_i h_j] - \mathbb{E}_{p(h,v)}[v_i h_j]] = \langle v_i h_j\rangle_{\text{data}} - \langle v_i h_j\rangle_{\text{model}} \qquad (۲)$$

با تمامی زیبایی ریاضیاتی خود، معادله ۲ غیر قابل محاسبه است جمله دوم نیاز به محاسبه $\mathbb{E}_{\text{model}}$ دارد، و جمع روی تعداد حالت های v, h نیاز به محاسبه $2^m, 2^n$ حالت مختلف است. یک ایده محاسباتی ساده، این است که همانطور که جمله اول با استفاده از Monte-Carlo estimation از سمپل های داده محسابه میشود، جمله دوم را نیز همینطور تخمین بزنیم.

$$\mathcal{S} = \{\tilde{v}_s\}_{s=1}^{S} \to \mathbb{E}_{p(h,v)}[v_i h_j] = \mathbb{E}_{p(v)}\mathbb{E}_{p(h|v)}[v_i h_j] \approx \frac{1}{S}\sum_{s=1}^{S}\mathbb{E}_{p(h|v)}[\tilde{v}_i h_j]$$

برای تخمین زدن جمله دوم با استفاده از روش های مونته-کارلو، نیاز به سمپل های زیادی از مدل $p(v,h)$ دارند، که همانطور که صحبت شد خود نیاز به الگوریتم MCMC دارد. در نتیجه برای دقیق شدن تقریب خود، نیاز به تعداد زیادی نمونه تولید شده توسط یک الگوریتم MCMC هستیم، که خود به اندازه کافی زنجیر طولانی ای بوده که توزیع $p(v)$ را تخمین بزند باشد. در نتیجه این الگوریتم، خیلی در عمل مناسب نیست.
ولی الگوریتم هایی وجود دارند که به این فرایند سرعت بسیاری میدهند، و تنها به k مرحله در الگوریتم MCMC ($k = 1$ در بسیاری از کاربرد ها) نیاز خواهیم داشت. و اینگونه contrastive-divergence معرفی میشود. حال جمله دوم رو تنها با استفاده از یک نمونه تقریبی از مدل با استفاده از k مرحله MCMC، با مقدار اولیه v انجام میشود، به صورتی که v یکی از نمونه های داده باشد.

$$\text{CD}_k(\theta, v_\circ) = -\sum_h p(h|v_\circ)\frac{\partial E(v_\circ, h)}{\partial\theta} + \sum_h p(h|v_k)\frac{\partial E(v_k, h)}{\partial\theta} \qquad (۳)$$

که در اینجا v_k نمونه تولید شده بعد از k مرحله MCMC بر روی نمونه اولیه داده $v_\circ$ است.
توزیع نمونه ها پس از k مرحله، را با p_θ^k مشخص میکنیم.

$$v_k \sim p_\theta^k \iff v_k \sim MCMC(p_\theta, k, \text{initial:}v_\circ)$$

مشخص است که $p_\theta^\circ = p_{\text{data}}, v_\circ \sim p_{\text{data}}$ و در حالت نهایی، $p_\theta^\infty = p_\theta, v^\infty \sim p_\theta$، پایداری الگوریتم فوق به صورت عملی، و تئوری اثابات شده است.

پرسش تئوری ۹. نشان دهید که بهینه کردن $\text{CD}_k(\theta, v_\circ)$ معادل با بهینه سازی $D_{\text{KL}}(p_{\text{data}}\|p_\theta) - D_{\text{KL}}(p_\theta^k\|p_\theta)$ است، که در آن:

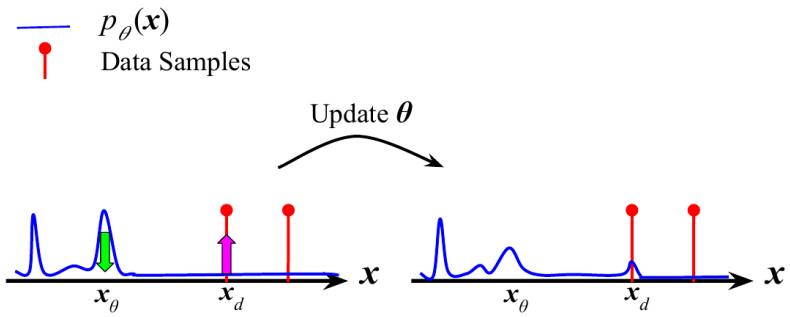
$$D_{\text{KL}}(P\|Q) = \sum_x P(x)\log\left(\frac{P(x)}{Q(x)}\right) = \mathbb{E}_{x\sim P}\left[\log\frac{P(x)}{Q(x)}\right]$$

نشان دهید که با افزایش k بهینه کردن CD معادل با تخمینگر بیشینه درست نمایی است.

با بهینه کردن CD دو اتفاق همزمان رخ میدهد.

$$\theta \longleftarrow \theta - \nabla_\theta CD \Rightarrow \begin{cases} p_\theta(x)\uparrow & x\sim p_{\text{data}} \\ p_\theta^k(x)\downarrow & x\sim p_\theta \end{cases}$$

به صورت خلاصه، حال یک الگوریتم کلی برای یادگیری RBM به صورت شهودی، فرایند به شکل زیر است.



شکل ۲: الگوریتم شهودی معمولی RBM

به صورت الگوریتمی، به شکل زیر است.

Algorithm 1. k-step contrastive divergence

Input: RBM $(V_1, \ldots, V_m, H_1, \ldots, H_n)$, training batch S
Output: gradient approximation Δw_{ij}, Δb_j and Δc_i for $i = 1, \ldots, n$, $j = 1, \ldots, m$

init $\Delta w_{ij} = \Delta b_j = \Delta c_i = 0$ for $i = 1, \ldots, n$, $j = 1, \ldots, m$
forall the $\boldsymbol{v} \in S$ **do**
  $\boldsymbol{v}^{(0)} \leftarrow \boldsymbol{v}$
  for $t = 0, \ldots, k-1$ **do**
    for $i = 1, \ldots, n$ **do** sample $h_i^{(t)} \sim p(h_i \mid \boldsymbol{v}^{(t)})$
    for $j = 1, \ldots, m$ **do** sample $v_j^{(t+1)} \sim p(v_j \mid \boldsymbol{h}^{(t)})$
  for $i = 1, \ldots, n$, $j = 1, \ldots, m$ **do**
    $\Delta w_{ij} \leftarrow \Delta w_{ij} + p(H_i = 1 \mid \boldsymbol{v}^{(0)}) \cdot v_j^{(0)} - p(H_i = 1 \mid \boldsymbol{v}^{(k)}) \cdot v_j^{(k)}$
    $\Delta b_j \leftarrow \Delta b_j + v_j^{(0)} - v_j^{(k)}$
    $\Delta c_i \leftarrow \Delta c_i + p(H_i = 1 \mid \boldsymbol{v}^{(0)}) - p(H_i = 1 \mid \boldsymbol{v}^{(k)})$

شکل ۳: االگوریتم بهینه سازی اولیه برای RBM

پرسش تئوری ۱۰. یافته های قسمت های قبل خود را، با الگوریتم k-step contrastive divergence توجیح کنید، در مورد پیچیدگی این الگوریتم، و انتظار شما از خروجی مدل بعد از اعمال یادگیری با استفاده از این الگوریتم بحث کنید.

پرسش تئوری ۱۱. همانطور که تا الان مشاهده کردید، ماشین بولتزمن مناسب کار کردن با داده های باینری است، و v و h هر دو متغیر های باینری هستند، آیا ممکن است این مدل ها را برای داده های پیوسته استفاده کرد؟ پیشنهاد شما چیست، برای $v, h \in \mathbb{R}$ و همچنین $v, h \in [0, 1]$ آیا با الگوریتم های مشابه با الگوریتم قبل میتوان این مدل ها را آموزش داد؟

۴ بخش عملی

۱.۴ بخش اول: پیش‌پردازش داده‌ها

- داده‌های MNIST را بارگذاری کرده و آن‌ها را به حالت باینری تبدیل کنید (مثلاً با اعمال یک آستانه روی شدت پیکسل‌ها).
- نمونه‌هایی از داده‌های باینری‌شده را نمایش دهید تا با داده‌ها آشنا شوید.

۲.۴ بخش دوم: پیاده‌سازی الگوریتم و یادگیری

- الگوریتم k-step contrastive divergence را برای یادگیری مدل خود پیاده‌سازی کنید.
- به ازای مقادیر مختلف k (مثلاً $k = ۱, ۵, ۱۰$) عملیات یادگیری را انجام دهید.
- نمونه‌های تولیدی مدل را به ازای kهای مختلف تولید کرده و نمایش دهید.

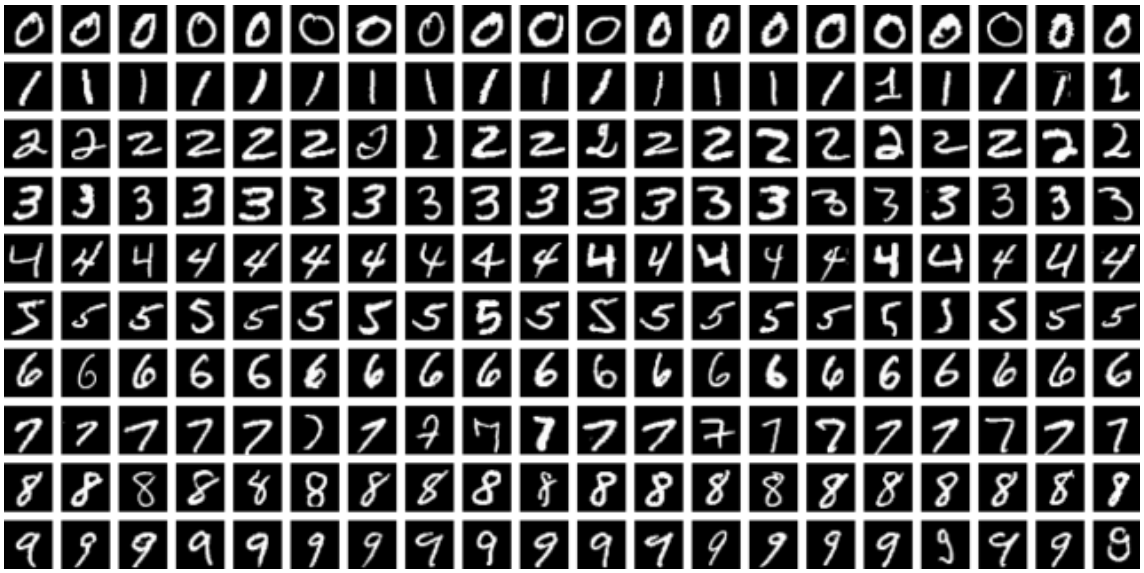
۳.۴ بخش سوم: نمایش روند نمونه‌سازی

پویانمایی‌ای از نمونه‌های تولیدی در طول اجرای الگوریتم MCMC ($v_۰, v_۱, \ldots, v_k$) بسازید و روند آن را نمایش دهید. این پویانمایی باید نشان دهد که چگونه نمونه‌ها در طول تکرارها بهبود می‌یابند.

۴.۴ بخش چهارم: کنترل روی نمونه‌های تولیدی

بررسی کنید که آیا می‌توان از مدل برای تولید نمونه‌هایی از اعداد مشخص (مثلاً ۰ یا ۱) استفاده کرد یا خیر.

- پیشنهاد خود را برای ایجاد کنترل بر خروجی مدل توصیف کنید.
- اگر امکان دارد، راهکار خود را پیاده‌سازی کرده و نمونه‌های کنترل‌شده تولیدی را نمایش دهید.



شکل ۴: نمونه هایی از داده های MNIST

۵ نکات مهم

لطفاً به نکات زیر دقّت کنید:

۱. پروژه شامل دو فاز خواهد بود.

۲. پروژه را میتوانید به صورت انفرادی یا به شکل گروه های دو نفره انجام دهید. دقت کنید چه به شکل انفرادی و چه به صورت گروهی باید تمام بخش های پروژه را انجام دهید و انجام انفرادی آن امتیاز اضافه ای برای شما نخواهد داشت.

۳. دو فاز این پروژه در مجموع ۲ تا ۳ نمره از نمره درس را تشکیل می دهند.

۴. پس از پایان پروژه یک روز برای تحویل حضوری پروژه در نظر گرفته می شود و باید کد ها و خروجی های خود را در حضور دستیاران آموزشی ارائه دهید و به پرسش های دستیاران پاسخ دهید. دقت کنید که تمام اعضای گروه باید به تمام بخش های پروژه مسلط باشند. در نهایت برای تمام اعضای گروه یک نمره در نظر گرفته خواهد شد.

۵. برای فاز نخست پروژه میتوانید حداکثر ۳ روز تاخیر مجاز استفاده نمایید اما به دلیل وجود ددلاین ثبت نمرات ددلاین فاز دوم پروژه سخت خواهد بود و امکان استفاده از تاخیر برای آن وجود ندارد.

۶. تمام بخش های هر دو فاز پروژه اجباری هستند و نمره امتیازی برای آن در نظر گرفته نشده است.

۷. تمامی شبیه‌سازی‌ها باید با کمک زبان Python انجام شود. همچنین مجاز هستید از تمام کتابخانه هایی که در طول تمرین ها از آنها استفاده کرده اید مانند `numpy`، `scipy` و `pytorch` استفاده نمایید اما دقت کنید پیاده سازی الگوریتم ها باید توسط شما انجام شده باشد و نمیتوانید از کتابخانه هایی که الگوریتم را به صورت آماده پیاده سازی کرده اند استفاده نمایید.

۸. تحویل پروژه به صورت گزارش و کدهای نوشته‌شده است. گزارش باید شامل پاسخ پرسش‌ها، تصاویر و نمودارها و نتیجه‌گیری‌های لازم باشد. در نهایت یک فایل شامل کد ها و یک گزارش به فرمت pdf را در سامانه CW آپلود نمایید. آپلود کردن پروژه توسط یکی از اعضای گروه کافی میباشد.

۹. اگر برای پاسخ به پرسش‌ها، از منبعی (کتاب، مقاله، سایت و...) کمک گرفته‌اید، حتماً به آن ارجاع دهید.

۱۰. درصورت مشاهده‌ی تقلّب، نمره‌ی هردو فرد صفر منظور خواهد شد.

۱۱. مسئول پروژه آقای سلیمان بیگی میباشد و در صورت داشتن مشکلاتی در گروهبندی، زمان تحویل حضوری و ... به ایشان (@amirr62a) پیام دهید.

۱۲. در صورت داشتن پرسش در بخش ۱ به @amirrezazameni، در بخش ۲ به @Mahdi_h721 و در بخش های ۳ و ۴ به @BornaKhodabandeh پیام دهید.

موفّق باشید!