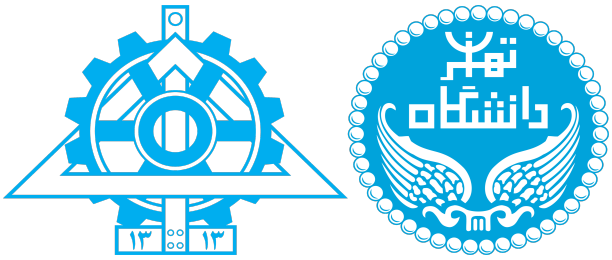


دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر

تمرین شماره: ۱

مدل‌های مولد عمیق

نام و نام‌خانوادگی: آرتین توسلی

شماره دانشجویی: ۸۱۰۱۰۲۵۴۳

نیم‌سال اول

سال تحصیلی ۴۰۵-۴۰۴

فهرست مطالب

فهرست تصاویر

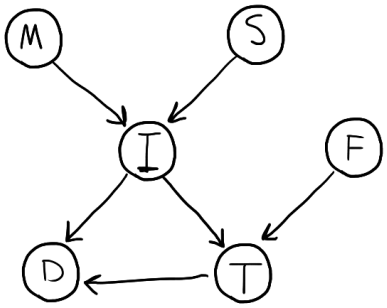
فهرست جداول

سوال ۱

PGM

۱.۱ شبکه بیزی

۱.۱.۱ رسم شبکه بیزی



شکل ۱.۱: شبکهٔ بیزی

۲.۱.۱ توزیع احتمالِ توأم

قاعده‌ی فاکتورگیری در شبکه‌های بیزی می‌گوید برای مجموعهٔ متغیرها $X_1, \ldots, X_n$ داریم:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P\big(X_i \mid \mathrm{Pa}(X_i)\big),$$

که در آن $\mathrm{Pa}(X_i)$ مجموعهٔ والدهای X_i در گراف است.

با توجه به شبکهٔ بیزی این سؤال، توزیع توأم به‌صورت زیر فاکتور می‌شود:

$$\boxed{P(M,S,F,I,T,D) = P(M)\,P(S)\,P(F)\,P(I \mid M,S)\,P(T \mid I,F)\,P(D \mid I,T)}.$$

۳.۱.۱ بررسی استقلال

طبق الگوریتم D-Seperation عمل میکنیم (در این الگوریتم اگه تمامی مسیر ها بلاک بشوند آن موقع مستقل میشود اما اگر تنها یک مسیر باز بماند پس مستقل میشوند)

آ. نادرست است؛ مسیر $F \to T \to D$ باز است، پس F و D وابسته‌اند.

ب. درست است؛ همه مسیر های S به D از I عبور میکند و با دانستن I تمام این مسیر ها به صورت حالت head-to-tail بلاک میشود.

ج. درست است؛ مسیر $M \to I \to T \leftarrow F$ در نقطه T بسته هست طبق حالت head-to-head و همینطور مسیر $M \to I \to D \leftarrow T \leftarrow F$ نیز در نقطه D به دلیل همین حالت بسته میباشد بسته است؛ در نتیجه M و F مستقل‌اند.

د. نادرست است؛ با دانستن نقطه T مسیر $M \to I \to T \leftarrow F$ را باز می‌کند، (حالت head-to-head) بنابراین M و F وابسته می‌شوند.

ه. درست است؛ مسیر $M \to I \to T$ با دانستن I توسط حالت head-to-tail بسته شده است اما با اینکه مسیر $M \to I \to D \leftarrow T$ توسط D باز شده است جون I مسیر را قطع کرده بود در کل مسیر قطع میشود بنابراین M و T مستقل می‌شوند.

۲.۱ شبکه مارکوف ۱

۱.۲.۱ توزیع احتمالِ توأم

$$P(C, O, A, S, T, B, M) = P(C)\, P(O)\, P(A)\, P(S \mid O)\, P(T \mid O, A) \times P(B \mid S)\, P(M \mid B, T, A).$$

۲.۲.۱ مارکوف بلنکت

مارکوف‌بلنکتِ یک گره شامل سه دستهٔ گره است:

- والدها (Parents)
- فرزندان (Children)
- والدهای مشترکِ فرزندان (Co-Parents)

طبق تعریف، مارکوف بلنکت یک گره یعنی کوچکترین مجموعه از گره هایی که اگر آنها را بدانیم، تمامی گره های دیگر گراف نسبت به آن گره مستقل خواهند بود. بدیهی است که یک گره به فرزندان و والد هایش وابسته هست پس حتما باید آنها را بدانیم اما زمانی که بچه ای را میدانیم، آن موقع باید حواسمون باشد که حالت V-structure (head-to-head) رخ ندهد، چون میدانیم با دانستن فرزند Z اولیا X،Y با هم وابسته میشوند پس حتما باید تمامی والد های مشترک فرزندان را نیز بدانیم.

در گراف این سؤال، والدهای T برابرند با:

$$\mathrm{Pa}(T) = \{O, A\}$$

فرزندِ T گرهٔ زیر است:

$$\mathrm{Ch}(T) = \{M\}$$

والدهای مشترکِ گرهٔ M (به‌جز خودِ T):

$$\{A, B\}$$

پس مارکوف‌بلنکتِ T به‌صورت زیر است:

$$\boxed{\mathrm{MB}(T) = \{O, A, M, B\}}$$

Perfect I-Map ۳.۲.۱

اگر G یک گراف و P توزیعِ روی همان متغیرها باشد، با $\mathcal{I}(G)$ مجموعهٔ استقلال‌های شرطیِ القاشده توسط جداسازی‌های گرافی و با $\mathcal{I}(P)$ مجموعهٔ استقلال‌های شرطیِ واقعیِ توزیع را نشان می‌دهیم.

P برای I-map $G \iff \mathcal{I}(G) \subseteq \mathcal{I}(P)$ (هر استقلالی که گراف القا کند در P هم درست است).

P برای D-map $G \iff \mathcal{I}(P) \subseteq \mathcal{I}(G)$ (هر استقلال واقعیِ P را گراف هم نشان می‌دهد).

P برای perfect I-map $G \iff \mathcal{I}(G) = \mathcal{I}(P)$ (هم I-map و هم D-map).

در این سوال فرض شده است که:

$$\mathcal{I}(P) = \mathcal{I}\big(G_{\text{Bayes}}\big)$$

حال در گراف بیزی طبق الگوریتم D-seperation داریم:

$$S \perp T \mid O.$$

استدلال: در مسیر S به T گره های O ، M قطع میباشند و در نتیجه تمامی مسیر ها بلاک شدند و این دو مستقل هستند (گره O در این مسیر بنابر حالت tail-to-tail قطع شده است و گره M در این مسیر بنابر حالت head-to-head قطع شده است)

$$S \perp T \mid O \in \mathcal{I}(P).$$

اما در گرافِ مارکوفِ داده‌شده، با separation می‌بینیم که با دانستن O هنوز مسیرِ

$$S - B - T$$

باز است؛ بنابراین

$$S \not\perp T \mid O \implies S \perp T \mid O \notin \mathcal{I}(G).$$

نتیجه. چون یک استقلال واقعیِ توزیع $(S \perp T \mid O)$ در گراف مارکوف نشان داده نشده است،

$$\mathcal{I}(P) \nsubseteq \mathcal{I}(G),$$

پس گراف مارکوف *D-map* نیست؛ در نتیجه نمی‌تواند *perfect I-map* نیز باشد.

۴.۲.۱ گراف وتری

گراف وتری گرافی است که هر دور به طول چهار یا بیشتر از آن شامل وتر باشد.

در گرافِ مارکوفِ دادهشده ، دورِ چهاررأسیِ زیر وجود دارد:

$$O - S - B - T - O.$$

وتر های این دور $O-B$ و $S-T$ هستند.پس گراف وتری نیست.

۵.۲.۱ توزیعِ توأم بر اساس کلیک بیشینه

کلیک های بیشینه، زیرگراف های کاملی هستند که با اضافه کردن گره دیگری در گراف، زیر گراف کامل بزرگتری ایجاد نمیشود. پس در گراف سوال داریم:

$$\{O, A, T\}, \quad \{A, T, M\}, \quad \{B, T, M\}, \quad \{O, S\}, \quad \{S, B\}, \quad \{C\}.$$

بنابراین توزیعِ توأم به‌شکل زیر نوشته می‌شود:

$$P(A, B, C, O, S, T, M) = \frac{1}{Z}\, \phi_1(O, A, T)\, \phi_2(A, T, M)\, \phi_3(B, T, M) \times \phi_4(O, S)\, \phi_5(S, B)\, \phi_6(C).$$

۶.۲.۱ حذفِ متغیرِ از توزیعِ توأم

۱) گرافِ بیزی. طبق زیربخش اول این قسمت داریم:

$$P(C,O,A,S,T,B,M) = P(C)\,P(O)\,P(A)\,P(S \mid O)\,P(T \mid O,A) \times P(B \mid S)\,P(M \mid B,T,A).$$

حذفِ C یعنی مجموع‌گیری(یا در حالت پیوسته انتگرال‌گیری) روی C:

$$\begin{aligned} P(O,A,S,T,B,M) &= \sum_C P(C,O,A,S,T,B,M) \\ &= \Big(\sum_C P(C)\Big)\; P(O)\,P(A)\,P(S \mid O)\,P(T \mid O,A)\,P(B \mid S)\,P(M \mid B,T,A) \\ &= 1 \times P(O)\,P(A)\,P(S \mid O)\,P(T \mid O,A)\,P(B \mid S)\,P(M \mid B,T,A), \end{aligned}$$

۲) گرافِ مارکوف. طبق زیربخش پنجم این قسمت داریم:

$$P(A,B,C,O,S,T,M) = \frac{1}{Z}\;\phi_1(O,A,T)\;\phi_2(A,T,M)\;\phi_3(B,T,M) \times \phi_4(O,S)\;\phi_5(S,B)\;\phi_6(C).$$

حال:

$$\begin{aligned} P(O,A,S,T,B,M) &= \sum_C P(C,O,A,S,T,B,M) \\ &= \frac{1}{Z}\Big(\sum_C \phi_6(C)\Big)\;\phi_1(O,A,T)\,\phi_2(A,T,M)\,\phi_3(B,T,M) \\ &\quad \times \phi_4(O,S)\,\phi_5(S,B) \\ &= \frac{1}{Z'}\phi_1(O,A,T)\,\phi_2(A,T,M)\,\phi_3(B,T,M) \\ &\quad \times \phi_4(O,S)\,\phi_5(S,B) \end{aligned}$$

اگر $\sum_C \phi_C(C) = 1$ (یا در حالت پیوسته $\int \phi_C(C)\,dC = 1$)، آنگاه Z = Z'
مگر نه $Z' = Z/\sum_C \phi_6(C)$ در هر صورت صرفا ضریب نرمال سازی ممکن است تغییر کند و ما میتوانیم C را حذف کنیم

۳.۱ شبکه مارکوف ۲

۱.۳.۱ توزیعِ توأم بر اساس کلیک‌های بیشینه

کلیک‌های بیشینهٔ گرافِ سؤال:

$$\{A,B,C\},\quad \{B,C,D\},\quad \{C,D,F\}\quad \{E,B\},\quad \{E,F\},\quad \{E,G\}.$$

بنابراین توزیعِ توأم چنین فاکتور می‌شود:

$$P(A,B,C,D,E,F,G) = \frac{1}{Z}\,\phi_1(A,B,C)\,\phi_2(B,C,D)\,\phi_3(C,D,F) \times \phi_4(E,B)\,\phi_5(E,F)\,\phi_6(E,G)$$

۲.۳.۱ بررسی استقلال

طبق الگوریتم Separation برای $X \perp Y \mid Z$ گره‌های Z و یال‌هایشان را حذف می‌کنیم؛ اگر بین X و Y مسیری نمانَد، مستقل‌اند.

آ) نادرست است. بدون شرط، مسیر $A-B-E-G$ وجود دارد؛ پس A و G وابسته‌اند.

ب) نادرست است. با حذف $\{C,D\}$ هنوز مسیر $A-B-E-F$ باز است؛ پس A و F وابسته‌اند.

ج) درست است. با حذف E، رأس G از بقیهٔ گراف جدا می‌شود و مسیری به C ندارد.

د) درست است. در ابتدا با حذف گره های B،C گراف جدید را تولید کرده، در این گراف جدید A،E مستقل میباشند چون مسیری بین آنها وجود ندارد یعنی $P(A) = P(A \mid E)$ و یعنی در گراف اصلی $P(A \mid B,C) = P(A \mid B,C,E)$

۳.۳.۱ اسکیل کردن پتانسیل در توزیعِ توأم

از بخش اول این قسمت داریم:

$$P(A,B,C,D,E,F,G) = \frac{1}{Z}\,\phi_1(A,B,C)\,\phi_2(B,C,D)\,\phi_3(C,D,F) \times \phi_4(B,E)\,\phi_5(E,F)\,\phi_6(E,G).$$

اکنون پتانسیل را ۵ برابر می‌کنیم و

$$\phi'_6(E,G) \triangleq 5\,\phi_6(E,G).$$

آنگاه

$$P'(A,B,C,D,E,F,G) = \frac{1}{Z}\,\phi_1\phi_2\phi_3\,\phi_4\phi_5\,\phi'_6 = \frac{1}{Z}\,5\,\phi_1\phi_2\phi_3\,\phi_4\phi_5\,\phi_6 = \frac{1}{Z'}\,\phi_1\phi_2\phi_3\,\phi_4\phi_5\,\phi_6, \qquad Z' = \frac{Z}{5}.$$

اثرِ «۵ برابر کردن» فقط در ثابتِ نرمال‌سازی جذب می‌شود و توزیع تغییری نمی‌کند

Variational Inference ۴.۱

رابطه زیر جواب بهینه $q(z)$ را بدست میاورد.

$$q^\star(z) = \arg\min_{q\in\mathcal{Q}} \mathrm{KL}\big(q(z)\,\|\,p(z\mid x)\big) = \arg\min_{q\in\mathcal{Q}} \mathrm{KL}\big(q(z)\,\|\,p(z,x)\big). \qquad (۱.۱)$$

درستی رابطه بالا با کمک رابطه زیر اثبات میشود (توجه کنید $p(x)$ نسبت به z ثابت است)

$$\log p(x) = -\,\mathrm{KL}\big(q(z)\,\|\,p(x,z)\big) + \mathrm{KL}\big(q(z)\,\|\,p(z\mid x)\big).$$

پس داریم:

$$\theta^\star = \arg\min_\theta \ \mathrm{KL}\big(q(z) \,\|\, p(z,x)\big) = E_q[\log \frac{q(z)}{p(z,x)}] \tag{۲.۱}$$

$$= E_q[\log \frac{q(z)}{p(x \mid z)p(z)}] \tag{۳.۱}$$

$$= E_q[\log \frac{\theta^2 \, z \, e^{-\theta z}}{z \, e^{-zx} \, e^{-z}}] \tag{۴.۱}$$

$$= E_q[\log\big(\theta^2 z e^{-\theta z}\big)] \ - \ E_q[\log\big(z e^{-zx-z}\big)] \tag{۵.۱}$$

$$= E_q[\log(\theta^2 z) - \theta z] \ - \ E_q[\log(z) - zx - z] \tag{۶.۱}$$

$$= E_q[\log(\theta^2 z)] - E_q[\theta z] \ - \ E_q[\log z] + E_q[zx + z] \tag{۷.۱}$$

$$= E_q[\log(\theta^2)] + E_q[\log z] - E_q[\theta z] \ - \ E_q[\log z] + E_q[zx + z] \tag{۸.۱}$$

$$= E_q[2\log\theta] \ - \ E_q[\theta z] \ + \ E_q[zx + z]. \tag{۹.۱}$$

توزیع احتمال روی z تعریف شده و بنابراین ترم‌های وابسته به x ثابت‌اند؛ پس داریم:

$$\theta^\star = 2\log\theta \ - \ \theta E_q[z] \ + \ (x+1)E_q[z] \tag{۱۰.۱}$$

$$= 2\log\theta \ - \ \theta\frac{2}{\theta} \ + \ (x+1)\frac{2}{\theta} \tag{۱۱.۱}$$

$$= f(\theta) \tag{۱۲.۱}$$

پس

$$f(\theta) = 2\log\theta \ - \ 2\cdot\theta \ + \ (x+1)\cdot\frac{2}{\theta}, \tag{۱۳.۱}$$

$$\frac{\partial f}{\partial\theta} = \frac{2}{\theta} \ - \ \frac{2(x+1)}{\theta^2} \ = \ 0 \implies 1 = \frac{x+1}{\theta^\star} \implies \boxed{\theta^\star = x+1}. \tag{۱۴.۱}$$

سوال ۲

VAE Training

VAE ۱.۲

تابع هزینه ۱.۱.۲

maximum log likelihood به ما میگوید که احتمال $\log p_\theta(x)$ را بیشینه بکنیم،

$$\log p_\theta(x) = \log \int p_\theta(x, z)\, dz$$

اما محاسبه انتگرال فوق غیر ممکن میباشد (حتی اگر z باینری هم باشد و تنها ۱۰۰ نمونه دیده باشیم 2^{100} جمع زدن نیاز هست که غیرممکن می باشد. بخاطر همین مسأله را تغییر میدهیم و از یک توزیع کمکی $q_\phi(z \mid x)$ استفاده میکنیم و مینویسیم:

$$\begin{aligned}
\log p_\theta(x) &= \log \int q_\phi(z \mid x)\, \frac{p_\theta(x, z)}{q_\phi(z \mid x)}\, dz \\
&\geq \mathbb{E}_{q_\phi}\big[\log p_\theta(x, z) - \log q_\phi(z \mid x)\big] \qquad \text{(نابرابریِ ینسن)} \\
&= \underbrace{\mathbb{E}_{q_\phi}\big[\log p_\theta(x \mid z)\big]}_{\text{ترم بازسازی}} - \underbrace{D_{\mathrm{KL}}\big(q_\phi(z \mid x) \,\|\, p(z)\big)}_{\text{نُرم‌ساز}} =: \mathcal{L}_{\mathrm{ELBO}}(\theta, \phi; x).
\end{aligned}$$

از سوی دیگر تجزیهٔ دقیق داریم:

$$\log p_\theta(x) = \mathcal{L}_{\text{ELBO}}(\theta, \phi; x) + D_{\text{KL}}\big(q_\phi(z \mid x) \,\|\, p_\theta(z \mid x)\big).$$

پس $\mathcal{L}_{\text{ELBO}}$ کرانِ پایینی بر $\log p_\theta(x)$ است و وقتی $q_\phi(z \mid x) = p_\theta(z \mid x)$ شود، این کران دقیقاً به $\log p_\theta(x)$ می‌رسد. برخلاف $\log p_\theta(x)$ عبارت $\mathcal{L}_{\text{ELBO}}$ کاملا قابل محاسبه و مشتق پذیر میباشد (با reparameterization trick) و میتوان آن را کمینه کرد.

نقشِ هر ترم در بهینه‌سازی.

- بازسازی $\mathbb{E}_{q_\phi}[\log p_\theta(x \mid z)]$: پارامترهای دیکودر θ را کنترل میکند تا با دریافت z نمونهٔ شبیهِ x بسازد این ترم «کیفیتِ بازسازی/likelihood» را بهبود می‌دهد.
- KL به پیشین $D_{\text{KL}}(q_\phi(z \mid x) \,\|\, p(z))$: پارامترهای انکودر ϕ را نُرم‌سازی می‌کند تا پسینِ تقریبی برابر پیشین بشود. اگر این ترم نبود، مدل ممکن بود داده ها را به بخشای مختلفی از فضای نهان مپ کند که یعنی باعث حفره هایی در این فضا بشود و اگر موقع تولید داده از روی فضای نهان از این حفره ها نمونه بگیریم، به تولید معناداری منجر نمیشود پس در اصل این عبارت باعث میشود تا مدل مولد بشود.

۲.۱.۲ دیتاست

دیتاست dSprites توسط گروه DeepMind معرفی شده است و یک مجموعه‌دادهٔ synthetic می‌باشد که برای ارزیابی مدل‌های مولد و به‌خصوص disentanglement learning استفاده می‌شود. تمامی تصاویر، شکل‌های دوبعدی ساده با رزولوشن 64×64 پیکسل و به‌صورت سیاه‌وسفید هستند و هر تصویر از ترکیب شش عاملِ پنهان[۱] به‌وجود می‌آید.

عوامل پنهان : شش عامل پنهان و مقادیر مربوط به هر کدام در جدول ۱.۲ آورده شده است:

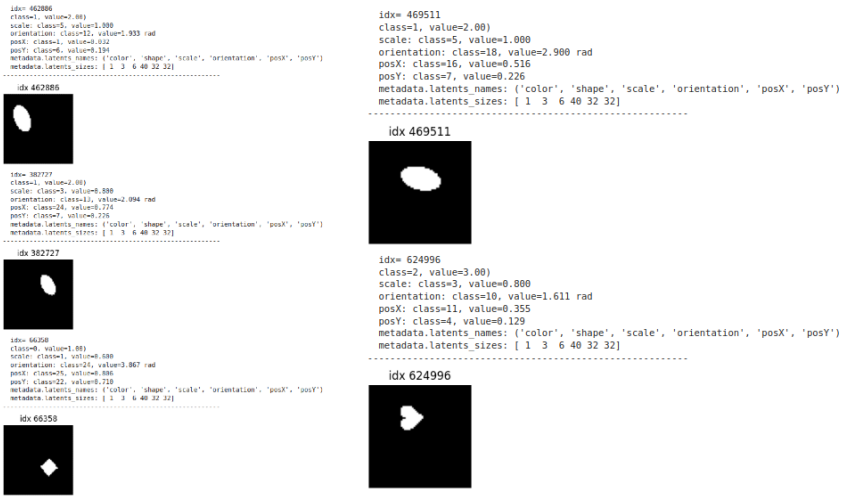
[۱] Latent Factors

جدول ۱.۲: عوامل پنهان در دیتاست dSprites

عامل	نوع مقادیر	تعداد حالت‌ها
Shape	(Square, Ellipse, Heart)	3
Scale	اندازه (پیوسته کوانتیزه‌شده)	6
Orientation	زاویه چرخش $[0, 2\pi)$	40
Color	grayscale	1
Position X	موقعیت افقی	32
Position Y	موقعیت عمودی	32

بنابراین تعداد کل تصاویر برابر است با:

$$1 \times 3 \times 6 \times 40 \times 32 \times 32 \;=\; 737{,}280$$



شکل ۱.۲: نمونه هایی از دیتاست dSprites

۳.۱.۲ حقه گرادیان

در VAE، انکودر به‌جای خروجی‌دادن یک *بردار ثابت*، پارامترهای توزیعِ نهان را می‌سازد؛ مثلاً برای گاوسی:

$$q_\phi(z \mid x) = \mathcal{N}\big(\mu_\phi(x),\, \sigma^2_\phi(x)\big).$$

سپس برای تغذیهٔ دیکودر، از این توزیع *نمونه‌گیری* می‌کنیم. نمونه‌گیری یک عملِ تصادفی و غیرقابل‌مشتق‌گیری است؛ بنابراین اگر $z \sim q_\phi(z \mid x)$ را مستقیماً به دیکودر بدهیم، مسیر گرادیان از در مسیر پارامترهای ϕ قطع می‌شود و آموزش شکست می‌خورد.

با یک ترفند بازپارامتری‌سازی میتوان این مشکل را حل کرد. به‌جای نمونه‌گیریِ مستقیم از $q_\phi(z \mid x)$، آن را به شکلِ *تابعی از نویزِ ثابت* می‌نویسیم تا تصادفیت از پارامترها *جدا* شود:

$$\varepsilon \sim \mathcal{N}(0, \mathrm{I}),$$
$$z = \mu_\phi(x) \; + \; \sigma_\phi(x) \odot \varepsilon.$$

اکنون $z = g_\phi(x, \varepsilon)$ یک نگاشتِ *قابل‌مشتق‌گیری* نسبت به ϕ است (نویزِ ε منبعِ تصادفیت است و به پارامترها وابسته نیست)؛ پس گرادیان به‌راحتی از دیکودر به انکودر عبور می‌کند.

۴.۱.۲ تحلیل مدل با بتا ۱

مدل شامل دو بخش اصلی Encoder و Decoder است که معماری هرکدام در شکل‌های ۲.۲ و ۳.۲ نمایش داده شده است. جدول ۲.۲ هایپرپارمتر های استفاده‌شده در آموزش مدل را نشان می‌دهد.

جدول ۲.۲: هایپرپارامترهای استفاده‌شده در مدل VAE.

پارامتر	مقدار
Input Shape	$(64, 64, 1)$
Filters	$[32, 64, 128]$
Dense Hidden Units	256
Latent Dimension	16
Num Downsampling	3
Batch Size	256
Epochs	50
Beta	1.0
Validation Split	0.05
Learning Rate	10^{-3}
Optimizer	Adam
Max Samples	$200,000$

Model: "encoder"

Layer (type)	Output Shape	Param #	Connected to
encoder_input (InputLayer)	(None, 64, 64, 1)	0	-
conv1 (Conv2D)	(None, 32, 32, 32)	320	encoder_input[0]…
batch_normalization (BatchNormalizatio…	(None, 32, 32, 32)	128	conv1[0][0]
leaky_re_lu (LeakyReLU)	(None, 32, 32, 32)	0	batch_normalizat…
dropout (Dropout)	(None, 32, 32, 32)	0	leaky_re_lu[0][0]
conv2 (Conv2D)	(None, 16, 16, 64)	18,496	dropout[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 16, 16, 64)	256	conv2[0][0]
leaky_re_lu_1 (LeakyReLU)	(None, 16, 16, 64)	0	batch_normalizat…
dropout_1 (Dropout)	(None, 16, 16, 64)	0	leaky_re_lu_1[0]…
conv3 (Conv2D)	(None, 8, 8, 128)	73,856	dropout_1[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 8, 8, 128)	512	conv3[0][0]
leaky_re_lu_2 (LeakyReLU)	(None, 8, 8, 128)	0	batch_normalizat…
dropout_2 (Dropout)	(None, 8, 8, 128)	0	leaky_re_lu_2[0]…
flatten (Flatten)	(None, 8192)	0	dropout_2[0][0]
dense_h (Dense)	(None, 256)	2,097,408	flatten[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 256)	1,024	dense_h[0][0]
leaky_re_lu_3 (LeakyReLU)	(None, 256)	0	batch_normalizat…
dropout_3 (Dropout)	(None, 256)	0	leaky_re_lu_3[0]…
z_mean (Dense)	(None, 16)	4,112	dropout_3[0][0]
z_log_var (Dense)	(None, 16)	4,112	dropout_3[0][0]
z (Sample)	(None, 16)	0	z_mean[0][0], z_log_var[0][0]

شکل ۲.۲: خلاصهٔ ساختار شبکهٔ انکودر.

Model: "decoder"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 16)	0
dense (Dense)	(None, 8192)	139,264
reshape (Reshape)	(None, 8, 8, 128)	0
conv2d_transpose (Conv2DTranspose)	(None, 16, 16, 128)	147,584
batch_normalization_4 (BatchNormalization)	(None, 16, 16, 128)	512
conv2d_transpose_1 (Conv2DTranspose)	(None, 32, 32, 64)	73,792
batch_normalization_5 (BatchNormalization)	(None, 32, 32, 64)	256
conv2d_transpose_2 (Conv2DTranspose)	(None, 64, 64, 32)	18,464
batch_normalization_6 (BatchNormalization)	(None, 64, 64, 32)	128
conv2d_transpose_3 (Conv2DTranspose)	(None, 64, 64, 1)	289

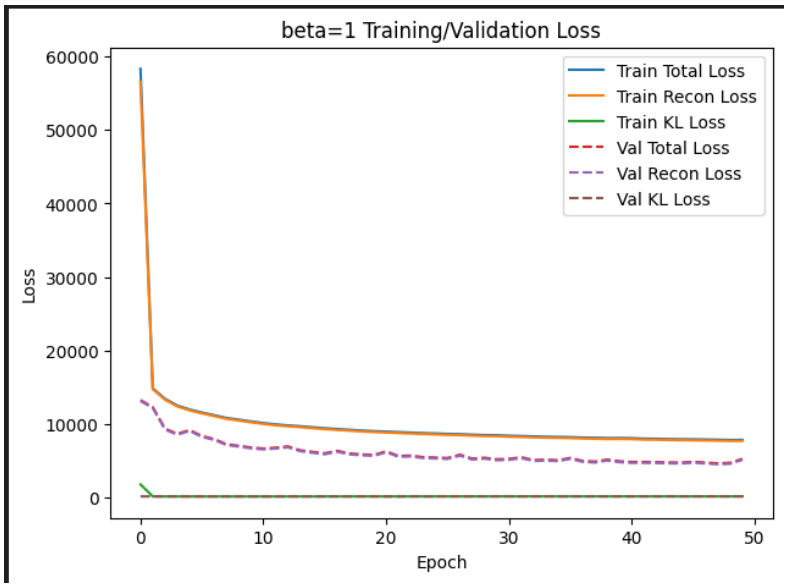
شکل ۳.۲: خلاصهٔ ساختار شبکهٔ دیکودر.

تابع هزینه در VAE شامل دو جزء اصلی است:

$$\mathcal{L} = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \beta D_{\mathrm{KL}}(q_\phi(z|x) \,\|\, p(z))$$

- ترم بازسازی $\mathbb{E}[\log p(x|z)]$: خطای بین ورودی و خروجی بازسازی‌شده را کمینه می‌کند و مدل را به تولید داده‌های مشابه ورودی وادار می‌سازد.
- ترم KL : فاصله بین توزیع پسین $q_\phi(z|x)$ و پیشین $p(z)$ را تنظیم می‌کند و فضای نهان را منظم نگه می‌دارد.
- پارامتر β نقش تنظیم‌کننده دارد: برای مدل تعادل بین دقت بازسازی و نظم فضای نهان را حفظ می‌کند.

شکل ۴.۲ تغییرات مقدار هر جزء از تابع هزینه را در طول ۵۰ دوره آموزش نشان می‌دهد.

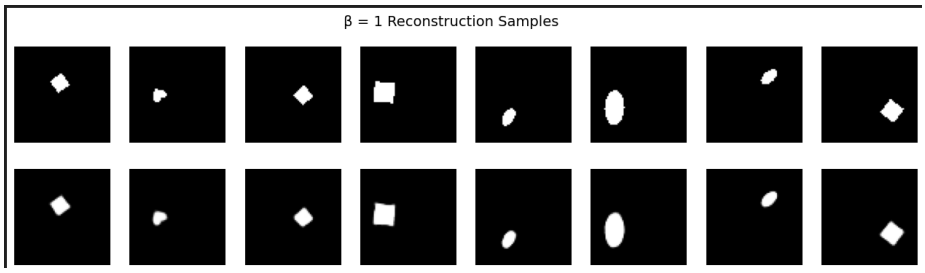


شکل ۴.۲: نمودار تغییرات تابع هزینه، بازسازی و KL برای داده‌های آموزش و اعتبارسنجی.

طبق نمودار بالا:

- ترم بازسازی (Recon loss) در اولین دوره آموزشی بسیار زیاد بوده است و به این معنی که در دوره اول تقریبا هیچ قدرت بازسازی عکس را ندارد اما بعد از چند دوره (۲ الی ۳) آموزش دیدن با سرعت تقریبا ثابت شروع به کاهش کردن میکند و یعنی مدل به قدرت بازسازی خوبی از عکس ها رسیده است.
- ترم KL بسیار کوچک مانده که نشان‌دهندهٔ هموار بودن فضای نهان و منظم شدن توزیع $q_\phi(z|x)$ است.

پس از پایان آموزش، ۸ نمونه از داده‌های آزمایش از مدل عبور داده شدند و خروجی بازسازی‌شده در شکل ۵.۲ نمایش داده شده است.



شکل ۵.۲: نمونه‌هایی از بازسازی مدل VAE برای $\beta = 1$. سطر اول ورودی اصلی و سطر دوم بازسازی مدل است.

تحلیل نتایج بازسازی:

- بازسازی‌ها شکل کلی و ویژگی‌های اصلی نمونه‌های ورودی (مانند نوع شکل، زاویه چرخش و موقعیت مکانی) را با دقت بالا حفظ کرده‌اند.
- در برخی موارد لبه‌ها کمی نرم‌تر یا تار هستند که به ماهیت توزیعی بازسازی در VAE مربوط است.

مدل آموزش‌داده‌شده با $\beta = 1$ عملکرد مناسبی در بازسازی داده‌ها داشته است. ترم بازسازی بیشترین تأثیر را در آموزش داشته، ترم KL فضای نهان را منظم نگه داشته، و نتایج تصویری کیفیت بازسازی مطلوبی را نشان می‌دهند.

۵.۱.۲ تاثیر بتا

با افزودن بتا به مدل VAE استاندارد، کنترل روی ترم KL داریم و انگار با یک ضریبی ELBO را داریم بیشینه میکنیم.

$$\mathcal{L}_{\beta\text{-VAE}} = \mathbb{E}_{q_\phi(z|x)}\big[\log p_\theta(x \mid z)\big] \;-\; \beta\, D_{\mathrm{KL}}\big(q_\phi(z \mid x) \,\|\, p(z)\big), \qquad \beta > 0.$$

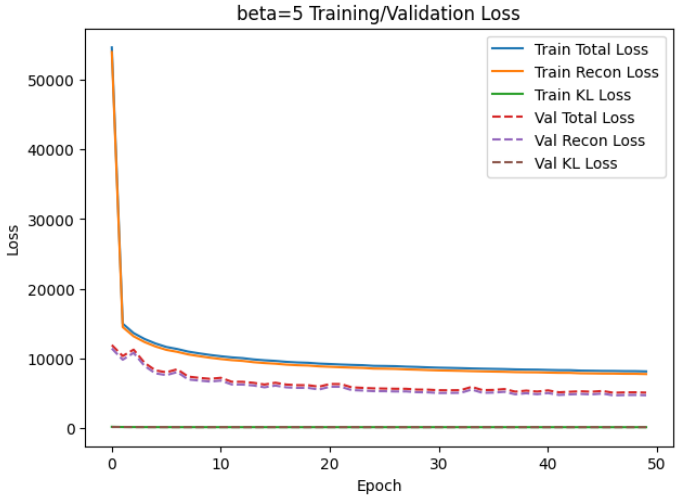
برای $\beta = 1$ به VAE استاندارد برمی‌گردیم؛ $\beta > 1$ یعنی *جریمهٔ قوی‌ترِ* واگراییِ KL، و $\beta < 1$ یعنی *جریمهٔ ضعیف‌تر*. در حالت $\beta > 1$ مدل وادار میشود اطلاعات کمتری از x را به z عبور دهد و یعنی اطلاعات مهمتر تنها کدگذاری میشوند در فضای پنهان و ابعاد فضای پنهان ویژگی های سطح بالاتری را میتوانند نشان بدهند که یعنی فضاتفسیرپذیر تر شده است. در همین راستای تفسیرپذیرتر بودن مدل عامل دیگری نیز تاثیر میگذارد با اهمیت بیشتر ترم KL باعث نزدیک تر شدن $q_\phi(z \mid x)$ به پیشین (معمولا گاوسی) میشود و یعنی ابعاد z کمتر به هم وابسته هستند و هر کدام از آنها میتوانند ویژگی خاصی را کد کنند. با افزایش اهمیت ترم KL

همینطور اعتماد ما از تصاویر تولید شده نیز بیشتر میشود چون فضای خالی در فضای پنهان مدل کمتر میشود. در کل داشتن بتا باعث میشود که ما بتوانیم اولویت بازسازی و تولید کردن مدل را با توجه به خواسته هایمون تنظیم کنیم.

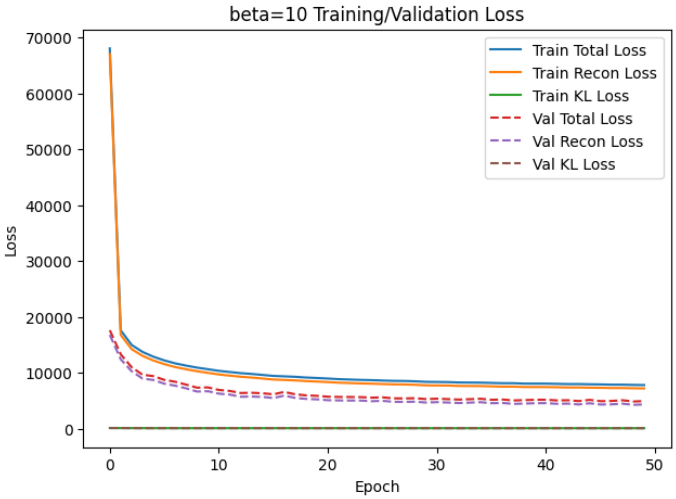
۶.۱.۲ تحلیل مدل با بتا ۵ و ۱۰

در این بخش، اثر مقادیر مختلف پارامتر β بر رفتار مدل VAE بررسی شده است. به‌منظور تحلیل دقیق‌تر، دو مقدار $\beta = 5$ و $\beta = 10$ انتخاب و مدل با همان ساختار و داده‌های بخش قبل آموزش داده شد. هدف از این آزمایش، مشاهدهٔ تأثیر ضریب منظم‌سازی KL بر همگرایی، بازسازی، و ساختار فضای نهان است.

شکل‌های ۶.۲ و ۷.۲ روند تغییرات سه مؤلفهٔ اصلی تابع هزینه (بازسازی، KL و کل هزینه) را برای هر دو مقدار β در طول ۵۰ ایپاک نمایش می‌دهند.



شکل ۶.۲: تغییرات تابع هزینه برای $\beta = 5$.



شکل ۷.۲: تغییرات تابع هزینه برای $\beta = 10$.

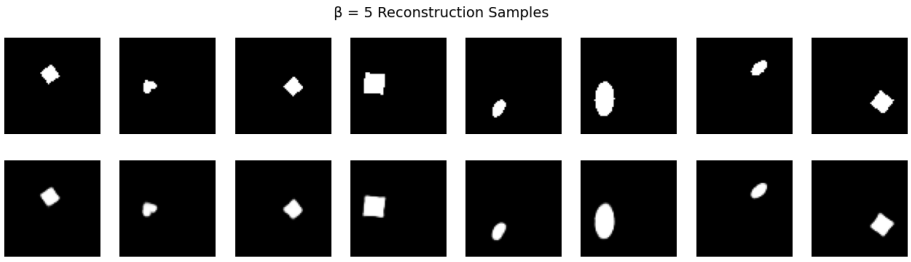
در انتهای آموزش، مقادیر نهایی تابع هزینه‌ها برای دو مدل به‌صورت زیر ثبت شد:

جدول ۳.۲: مقادیر نهایی تابع هزینه برای مقادیر مختلف β

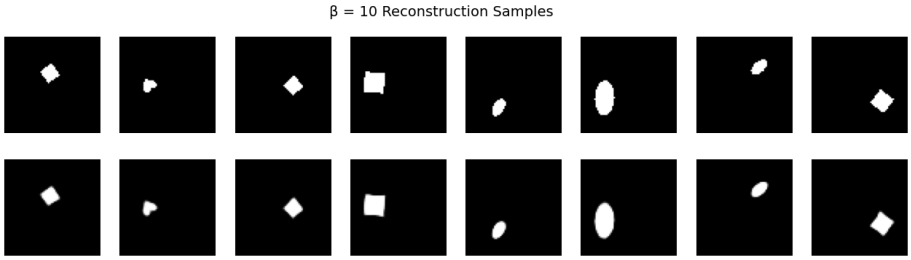
Loss Validation	KL	Reconstruction	Loss	β
5045.09	76.12	7716.18	8096.78	5
4914.43	60.62	7177.50	7783.67	10

هر دو مدل به‌خوبی همگرا شده‌اند و رفتار یکنواختی در کل فرایند آموزش دارند. در هر دو مقدار β، بخش بازسازی (Reconstruction Loss) سهم عمدهٔ تابع هزینه را دارد و به‌سرعت کاهش یافته و سپس در ناحیه‌ای پایدار قرار گرفته است. ترم KL نیز در هر دو مدل کوچک و ثابت باقی مانده که نشان‌دهندهٔ پایداری فضای نهان و نزدیکی توزیع پسین $q_\phi(z|x)$ به پیشین $p(z)$ است. در مدل با $\beta = 10$، مقدار کل هزینه اندکی کمتر است که بیانگر فشرده‌تر شدن فضای نهان و تنظیم بهتر بین بازسازی و منظم‌سازی است. در واقع، فشار KL باعث شده مدل بازنمایی تمیزتر و مؤثرتری از داده‌ها یاد بگیره.

شکل‌های ۸.۲ و ۹.۲ نمونه‌هایی از بازسازی مدل را برای دو مقدار مختلف β نمایش می‌دهند. در هر دو حالت، مدل توانسته ویژگی‌های کلیدی داده‌های ورودی (شکل، زاویهٔ چرخش، موقعیت و مقیاس) را بازسازی کند.



شکل ۸.۲: نمونه‌هایی از بازسازی مدل برای $\beta = 5$.



شکل ۹.۲: نمونه‌هایی از بازسازی مدل برای $\beta = 10$.

۷.۱.۲ معیار MIG

برای سنجش میزان استقلال و تفکیک‌پذیری فضای نهان، از معیار MIG (Mutual Information Gap) استفاده می‌شود. به‌عبارت دیگر، چقدر هر بُعد از فضای نهان می‌تواند یک عامل معنایی مستقل از داده (مانند چرخش، اندازه یا موقعیت) را بازنمایی کند.

این معیار بر پایه‌ی Mutual Information بین متغیرهای نهان z_i و فاکتورهای اصلی داده v_j تعریف می‌شود. برای هر عامل واقعی v_j (مثلاً زاویه، موقعیت، شکل)، دو بعد از فضای نهان که بیشترین اطلاعات متقابل را با آن دارند انتخاب می‌شوند و اختلاف آن‌ها اندازه‌گیری می‌شود. سپس این اختلاف، نرمال‌سازی‌شده با آنتروپی فاکتور مربوطه محاسبه می‌شود:

$$\text{MIG} = \frac{1}{K} \sum_{j=1}^{K} \frac{1}{H(v_j)} \left(I(z_{i_1}; v_j) - I(z_{i_2}; v_j) \right)$$

که در آن:

- $I(z_{i_1}; v_j)$ بیشترین Mutual Information بین بعد z_{i_1} و فاکتور v_j است؛
- $I(z_{i_2}; v_j)$ دومین مقدار بیشترین Mutual Information است؛
- $H(v_j)$ آنتروپی فاکتور v_j می‌باشد.

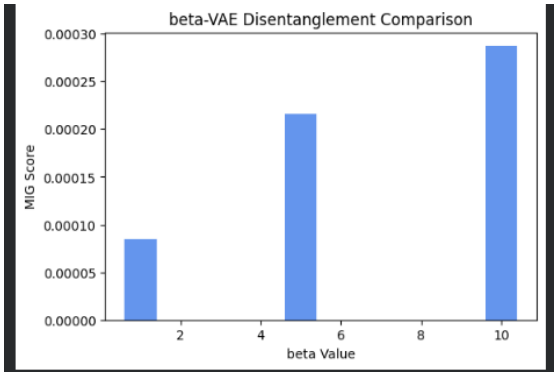
هرچه مقدار MIG بزرگ‌تر باشد، نشان‌دهندهٔ تفکیک‌پذیری بالاتر ابعاد نهان و استقلال بیشتر آن‌ها از یکدیگر است.

در این آزمایش، سه مدل با مقادیر مختلف β آموزش داده شدند ($\beta = 1, 5, 10$) و برای هر مدل، مقدار کلی MIG و مقادیر مربوط به هر فاکتور محاسبه گردید.

جدول ۴.۲: مقادیر MIG کلی و به‌ازای فاکتورهای مختلف برای مدل‌های گوناگون

posY	posX	Orientation	Scale	Shape	MIG کل	β
0.0002	0.0002	0.0001	0.0000	0.0000	0.0001	1
0.0002	0.0008	0.0001	0.0000	0.0000	0.0002	5
0.0005	0.0005	0.0005	0.0000	0.0000	0.0003	10

همان‌طور که جدول نشان می‌دهد، با افزایش مقدار β، نمرهٔ MIG به‌صورت پیوسته افزایش یافته است. مقدار کلی MIG از 0.0001 در $\beta = 1$ به 0.0003 در $\beta = 10$ رسیده است. بیشترین پیشرفت مربوط به فاکتورهای Orientation و posX/posY بوده که بیانگر استقلال بهتر ابعاد نهان در نمایش موقعیت و زاویهٔ شکل‌ها است.



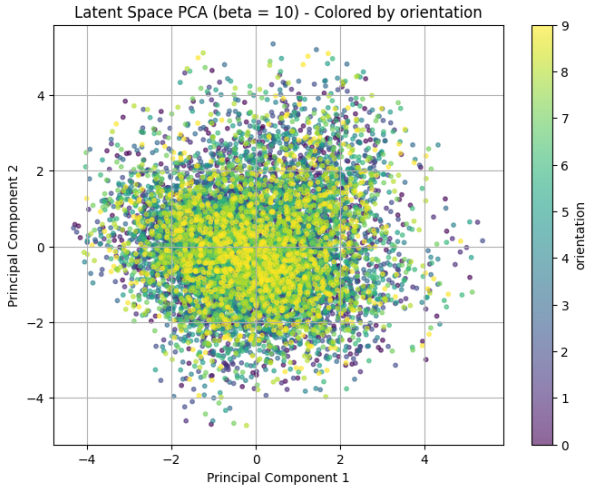
شکل ۱۰.۲: مقایسهٔ مقدار MIG برای مقادیر مختلف β.

افزایش پیوستهٔ مقدار MIG با افزایش β، به‌خوبی نشان‌دهندهٔ نقش منظم‌سازی قوی‌تر در مدل‌های β-VAE است. زمانی‌که β بزرگ‌تر می‌شود، مدل مجبور می‌شود تا اطلاعات غیرضروری را از فضای نهان حذف کند و هر بُعد از z را به بازنمایی یک عامل خاص اختصاص دهد. در نتیجه، ابعاد فضای نهان مستقل‌تر و قابل‌تفسیرتر می‌شوند.

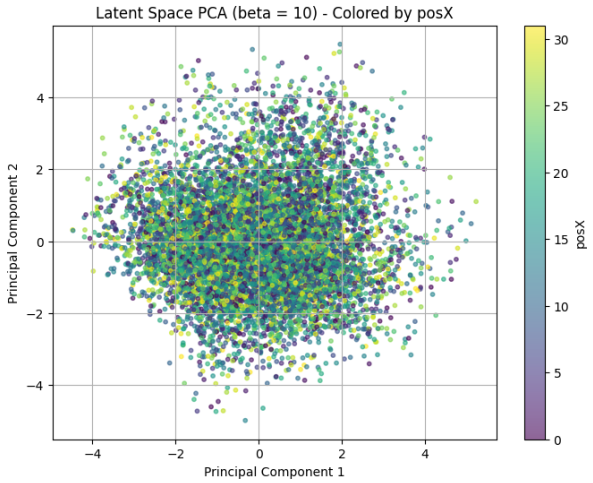
PCA ۸.۱.۲

برای بررسی تفکیک‌پذیری ابعاد فضای نهان مدل β-VAE از تحلیل مؤلفه‌های اصلی (PCA) استفاده شد. در این بخش، فضای نهان مدل با $\beta = 10$ در دو مؤلفهٔ اصلی تصویرسازی شده است. هر نمودار داده‌های نهان را در فضای دوبعدی نشان می‌دهد که رنگ نقاط نشان‌دهندهٔ مقدار یکی از عوامل اصلی داده‌ها (مانند چرخش، موقعیت یا شکل) است.

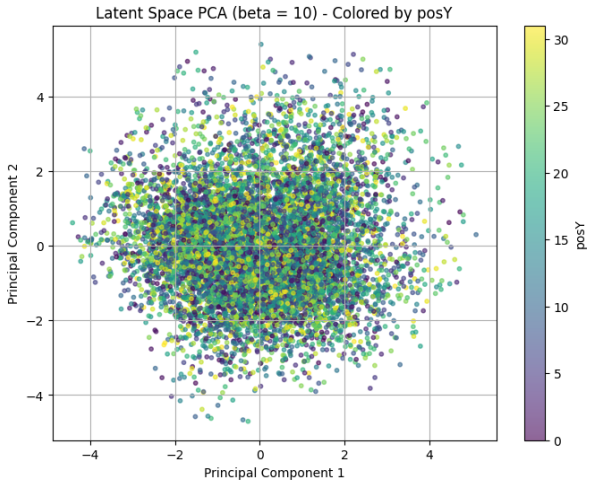
شکل‌های زیر پراکندگی داده‌ها را برای پنج فاکتور اصلی نمایش می‌دهند:



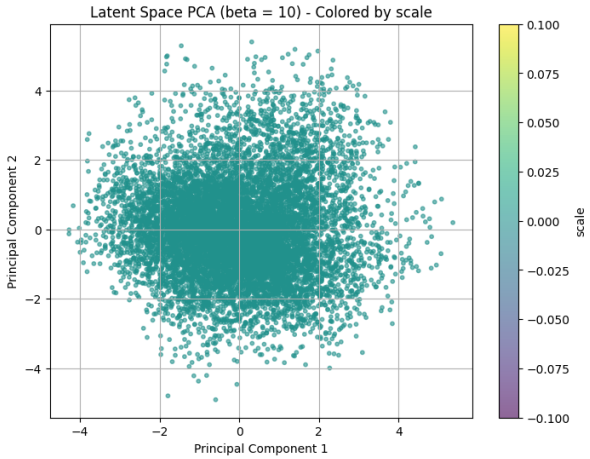
شکل ۱۱.۲: پراکندگی فضای نهان برای فاکتور orientation.



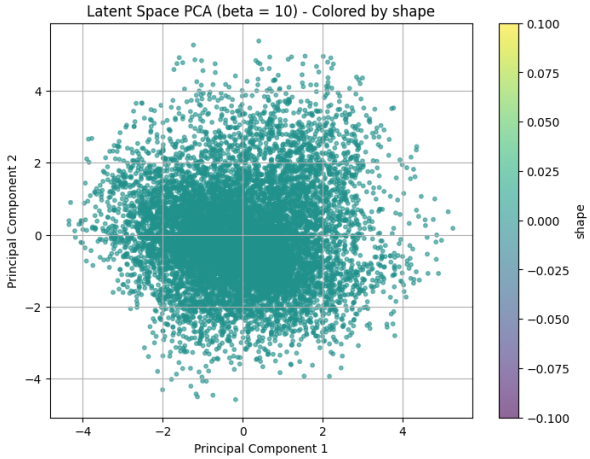
شکل ۱۲.۲: پراکندگی فضای نهان برای فاکتور posX.



شکل ۱۳.۲: پراکندگی فضای نهان برای فاکتور posY.



شکل ۱۴.۲: پراکندگی فضای نهان برای فاکتور scale.



شکل ۱۵.۲: پراکندگی فضای نهان برای فاکتور shape.

در نمودارهای بالا مشاهده می‌شود که نقاط فضای نهان در اطراف مبدأ و به‌صورت تقریباً کروی توزیع شده‌اند؛ این موضوع نشان می‌دهد که مدل موفق شده است تا حدودی توزیع نهان را مطابق با توزیع پیشین گاوسی $p(z) = \mathcal{N}(0, I)$ نگه دارد.

با بررسی رنگ‌بندی‌ها:

- در فاکتور orientation ، گرچه تغییر رنگ‌ها تدریجی نیست، ولی می‌توان الگوی ضعیفی از خوشه‌بندی مشاهده کرد. این بیانگر آن است که برخی ابعاد نهان توانسته‌اند اطلاعات مربوط به چرخش شکل را بازنمایی کنند.
- در فاکتورهای posX و posY توزیع رنگ‌ها نسبتاً یکنواخت‌تر است، اما نواحی مرکزی و پیرامونی تفاوت رنگی اندکی نشان می‌دهند. این موضوع حاکی از آن است که مدل در نمایش موقعیت اشکال در تصویر نیز به صورت ضعیف تفکیک انجام داده است.
- برای فاکتورهای scale و shape، تقریباً هیچ الگوی رنگی مشخصی دیده نمی‌شود؛ یعنی مدل در بازنمایی اندازه یا نوع شکل چندان موفق به جداسازی عوامل نشده است. این نتیجه با مقادیر پایین MIG برای همین فاکتورها سازگار است.

در مجموع می‌توان گفت که مدل $\beta = 10$ تا حدودی توانسته بین برخی عوامل (مخصوصاً موقعیت و جهت‌گیری) تمایز قائل شود، اما در سایر عوامل مانند شکل و مقیاس هنوز هم هم‌پوشانی بالایی بین ابعاد نهان وجود دارد. این نتایج با مقادیر پایین MIG و شکل توزیع گاوسی در فضای نهان هماهنگ است و نشان می‌دهد که اگرچه مدل فشرده و منظم است، اما هنوز به تفکیک کامل عوامل معنایی در ابعاد مستقل نرسیده است. برای بهبود تفکیک‌پذیری، می‌توان از مقادیر بالاتر β، استفاده کرد.

VAE Variants ۲.۲

VQ-VAE ۱.۲.۲

تفاوت ساختاری مدل VQ-VAE با VAE در مدل‌های VAE معمولی، انکودر خروجی‌هایی پیوسته تولید می‌کند که پارامترهای یک توزیع (معمولاً گاوسی) هستند:

$$q_\phi(z \mid x) = \mathcal{N}\big(\mu_\phi(x),\, \sigma^2_\phi(x) I\big).$$

نمونه‌گیری از این توزیع به‌صورت پیوسته انجام می‌شود و فضای نهان نیز پیوسته است. اما در VQ-VAE، به‌جای استفاده از توزیع پیوسته، انکودر یک بردار ویژگی $z_e(x)$ می‌سازد و سپس نزدیک‌ترین بردار از میان مجموعه‌ای محدود از بردارهای نهان (کدبوک) انتخاب می‌شود:

$$z_q(x) = e_k \rightarrow k = \arg\min_j \|z_e(x) - e_j\|_2^2.$$

مجموعه‌ی $\{e_1, e_2, \ldots, e_K\}$ کدبوک نام دارد و هر e_k نمایندهٔ «مرکز ثقل» داده‌هایی است که به آن کد نسبت

داده شده‌اند. این کدها پارامترهای قابل یادگیری شبکه هستند و در طول آموزش به‌روزرسانی می‌شوند تا به‌طور مؤثر فضای نهان را پوشش دهند. تابع هزینه در VQ-VAE تابع هزینه‌ی کلی مدل از سه بخش تشکیل شده است:

$$\mathcal{L} = \underbrace{\|x - \hat{x}\|_2^2}_{\text{بازسازی (Reconstruction)}} + \underbrace{\|\text{sg}[z_e(x)] - e_k\|_2^2}_{\text{به‌روزرسانیِ کدبوک (Codebook Loss)}} + \underbrace{\beta \|z_e(x) - \text{sg}[e_k]\|_2^2}_{\text{تعهد انکودر (Commitment Loss)}} .$$

۱. بازسازی (Reconstruction Loss). این بخش همانند VAE، مسئول بازسازی ورودی x از روی بردار گسسته‌ی انتخاب‌شده e_k است:

$$\|x - \hat{x}\|_2^2 = \|x - D(e_k)\|_2^2.$$

۲. به‌روزرسانی کدبوک (Codebook Loss). این بخش باعث می‌شود هر بردار e_k در کدبوک به سمت مرکز داده‌هایی حرکت کند که به آن اختصاص یافته‌اند:

$$\|\text{sg}[z_e(x)] - e_k\|_2^2.$$

در اینجا عملگر $\text{sg}[\cdot]$ به معنی stop-gradient است؛ یعنی گرادیان فقط برای e_k اعمال می‌شود و انکودر ثابت می‌ماند.

۳. تعهد انکودر (Commitment Loss). این بخش انکودر را وادار می‌کند تا به کدی که انتخاب کرده پایبند بماند و بین چند کد نزدیک به‌طور مداوم نوسان نکند:

$$\beta \|z_e(x) - \text{sg}[e_k]\|_2^2.$$

گسسته‌سازی در این مدل به این معناست که فضای نهان دیگر پیوسته نیست، بلکه از میان K کد مشخص انتخاب می‌شود. هر ورودی در واقع به یکی از کدهای گسسته‌ی کدبوک نگاشت می‌شود. بنابراین فضای نهان به‌جای پیوستگی، ساختار خوشه‌ای پیدا می‌کند و هر کد نماینده‌ی مجموعه‌ای از ورودی‌های مشابه است. مزایای مدل VQ-VAE

- رفع مشکل فروپاشی پسین (Posterior Collapse): اگر دیکودر بسیار قوی شده باشد، مدل می‌تواند بدون استفاده از متغیر پنهان z، ورودی x را بازسازی کند. در این حالت، برای کمینه کردن

ترم KL، انکودر تمایل دارد توزیع پسین خود را به پیشین نزدیک کند:

$$q_\phi(z \mid x) \approx p(z).$$

در نتیجه z دیگر حاوی هیچ اطلاعات خاصی از ورودی x نخواهد بود و بازنمایی نهان معنای خود را از دست می‌دهد. اما در VQ-VAE، چون دیکودر فقط از کدهای گسسته استفاده می‌کند، انکودر ناچار است اطلاعات معنی‌دار را در $z_q(x)$ رمز کند.

- فضای نهان تفسیرپذیرتر کدهای گسسته معمولاً قابل تفسیرتر از بردارهای پیوسته‌اند و می‌توانند مفاهیم سطح بالایی را نشان دهند.
- پایداری در آموزش: حذف نمونه‌گیری تصادفی از توزیع گاوسی موجب پایداری بیشتر گرادیان‌ها می‌شود.
- راحت تر مدل کردن وابستگی های طولانی فضای z گسسته و کوچکتر از فضای پیکسل ها هست و آموزش مدل مولد روی این فضا (شبیه توکن ها) سریعتر و پایدارتر میباشد.

VampPrior ۲.۲.۲

در مدل‌های VAE معمولی، پیش‌فرض را معمولاً از پیش انتخاب می‌کنیم، مانند یک پیش‌فرض گاوسی ساده. اما می‌توان پیش‌فرضی پیدا کرد که $\mathcal{L}$ را بهینه کند با حداکثر کردن تابع لاگرانژ زیر به کمک ضریب لاگرانژ β:

$$\max \; -\mathbb{E}_{z\sim q(z)}\left[-\ln p_\lambda(z)\right] + \beta \int p_\lambda(z)dz - 1.$$

جواب این عبارت:

$$p^*_\lambda(z) = \frac{1}{N}\sum_{n=1}^{N} q_\phi(z \mid x_n).$$

اما این انتخاب ممکن است منجر به overfitting بشود و همینطور پیچیدگی محاسباتی بالایی دارد از اون طرف هم پیش فرض های ساده ای مانند توزیع نرمال استاندارد باعث میشود مدل ها over-regularized .بشوند و بسیاری از ابعاد فضای نهان فعال نشود برای حل تمامی این مشکلات، میتوان از ایده بهینه posterior تجمعی استفاده کرد اما با ترکیب تعدادی داده مصنوعی pseudo-inputs که خود اینها توسط مدل یاد گرفته میشوند.

$$p_\lambda(z) = \frac{1}{K}\sum_{k=1}^{K} q_\phi(z \mid u_k),$$

۱. K: تعداد pseudo-inputs است. این‌ها داده‌های مصنوعی قابل یادگیری هستند.

۲. u_k: داده‌های مصنوعی (pseudo data points) هستند که به‌طور همزمان با پارامترهای مدل (انکودر و دیکودر) یاد گرفته می‌شوند. تابع هزینه در این مدل به‌صورت زیر تعریف می‌شود:

$$\mathcal{L} = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)] - D_{\mathrm{KL}}\left(q_\phi(z \mid x) \| p_\lambda(z)\right),$$

۳. $q_\phi(z \mid u_k)$: توزیع posterior است که انکودر مدل بر اساس داده u_k تولید می‌کند.

این ورودی های مصنوعی در اصل به عنوان نماینده هایی از منیفولد داده عمل میکنند

مقایسه پیش فرض های VAE و VampPrior

بیشینه کردن ELBO برابر است با پسین تجمعی روی کل داده ها:

$$p^*_\lambda(z) \;=\; \frac{1}{N}\sum_{n=1}^{N} q_\phi(z \mid x_n),$$

اما این انتخاب از نظر محاسباتی پرهزینه است. در مقابل، پیش‌فرضِ سادهٔ گاوسیِ استاندارد موجب *بیش‌منظم‌سازی* و فعال‌نشدنِ بسیاری از ابعادِ نهان می‌شود. روش VampPrior یک تقریب عملی از جواب بهینه هست که بدون هزینه چشمگیری این توزیع را تقریب میزند و دیگر نیازی نیست به یک توزیع همگی مپ بشوند و یک پیش فرض چند حالته و منطبق با داده به ما میدهد.. .همینطور پیش فرض و پسین به صورت جفت آموزش میبینند (در VAE پیش فرض ثابت است و پسین باید خودش را با آن تطبیق بدهد.)
VampPrior استفاده بهتری از ظرفیت نهان دارد (یعنی برخلاف نرمال استاندارد که فقط یک قله داره، این پیش‌فرض چندین مرکز (mode) در فضای نهان داره. در نتیجه هر داده‌ی واقعی می‌تونه نزدیک به یکی از این هاmode قرار بگیره و نیازی نیست همه چیز توی یه نقطه متمرکز بشه. بنابراین مدل می‌تونه از بخش‌های مختلف فضای نهان برای نمایش گروه‌های متفاوت داده استفاده کنه.)

SC-VAE ۳.۲.۲

بازنماییِ تنک (Sparse) به این معناست که بیشترِ درایه‌های بردارِ نهان صفر یا نزدیک به صفر باشند و تنها چند بعد واقعاً فعال باشند. برای بازنمایی یک داده خاص معمولاً فقط چند ویژگی کافی است، بنابراین مدل باید به‌جای پخش‌کردن اطلاعات در تمام ابعاد، تنها چند بعد مرتبط را *روشن* کند.

تنک‌سازی معمولاً با استفاده از قید یا جریمهٔ ℓ_1 در تابع هزینه بیان می‌شود:

$$\min_z \ \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1,$$

که در آن:

- D دیکشنری است،
- z بردار بازنماییِ تنک است،
- $\|z\|_1 = \sum_i |z_i|$ جریمه‌ای است که مقدارهای کوچک را به صفر نزدیک می‌کند.

در واقع ترمِ ℓ_1 باعث می‌شود بسیاری از مؤلفه‌های z صفر شوند.

الگوریتم ISTA (Iterative Shrinkage-Thresholding Algorithm) برای حل مسئلهٔ بالا استفاده می‌شود. در هر گام از ISTA دو مرحله انجام می‌شود:

۱. به‌روزرسانی گرادیانی برای کاهش خطای بازسازی،
۲. اعمال آستانه‌گذاری نرم برای صفر کردن مقادیر کوچک.

فرمول تکراریِ کلاسیک آن چنین است:

$$z^{(t+1)} = \mathcal{S}_{\alpha\lambda}\Big(z^{(t)} + \alpha D^\top (x - Dz^{(t)})\Big),$$

که $\mathcal{S}_\tau(v) = \operatorname{sign}(v)\max(|v| - \tau, 0)$ عملگرِ آستانه‌گذاری نرم (Soft Thresholding) است و α نرخ یادگیری می‌باشد.

در مدل SC–VAE، نسخهٔ آموختنی این الگوریتم با نام LISTA (Learned ISTA) استفاده می‌شود، که در آن پارامترهای ماتریس‌ها و آستانه‌ها نیز قابل یادگیری هستند:

$$z^{(t+1)} = \mathcal{S}_\theta\big(W_1 x + W_2 z^{(t)}\big),$$

چند گام از این تکرار به صورت یک زیرشبکه در دلِ VAE تعبیه می‌شود. به این ترتیب، LISTA یک نگاشتِ مستقیم و سریع از x به z یاد می‌گیرد و آموزش مدل به‌صورت end-to-end قابل انجام است.

ساختار مدل SC–VAE در مدل SC–VAE، انکودر، ماژول LISTA و دیکودر در یک زنجیره‌ی واحد قرار دارند:

$$x \xrightarrow{\text{Encoder}} h \xrightarrow{\text{LISTA}} z \xrightarrow{\text{Decoder}} \hat{x}.$$

- انکودر ویژگی اولیه را از ورودی استخراج می‌کند.
- بلوک LISTA چند گام تکرار گرادیانی + آستانه‌گذاری را برای تنک‌سازی انجام می‌دهد و z را تولید می‌کند.
- دیکودر با استفاده از z داده را بازسازی می‌کند.

تمام پارامترها (انکودر، دیکودر، آستانه‌ها و وزن‌های LISTA) به‌صورت همزمان آموزش داده می‌شوند.

تابع هزینه در SC–VAE

$$\mathcal{L} = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{\mathrm{KL}}\big(q_\phi(z|x) \,\|\, p(z)\big) + \lambda \|z\|_1.$$

مزایای بازنماییِ تنک :

- تفسیرپذیرتر: هر داده تنها چند ویژگی را فعال می‌کند و تفسیر هر ویژگی راحتر خواهد بود؛
- استفادهٔ بهتر از ظرفیت فضای نهان: برخلاف VAE کلاسیک که بسیاری از ابعاد نهان غیرفعال می‌مانند، در اینجا ابعاد خاصی برای هر نمونه واقعاً فعال‌اند و مدل از ظرفیت فضای نهان استفادهٔ کامل‌تری دارد.
- کاهش posterior collapse: به دلیل قید تنکی، مدل مجبور است ویژگی‌های متمایز را در مؤلفه‌های خاص z ذخیره کند و نمی‌تواند تمام اطلاعات را در دیکودر پنهان کند.
- فضای نهان شبیه گسسته بردار z هنوز پیوسته است، اما چون بیشتر درایه‌ها صفرند، رفتار آن شبیه انتخاب چند کُد محدود است (گویی از کدبوک گسسته انتخاب می‌شود).
- robust تر: کدهای تنک اغلب بازنمایی‌های فشرده و مقاوم به نویز تولید می‌کنند.