

به نام خدا

دانشگاه صنعتی شاهرود

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

درس یادگیری عمیق

تمرین اول

پرسش اول	نام دستیار طراح	آقای حسین پور - خانم شاه ولی
پرسش دوم	نام دستیار طراح	آقای حسین پور - خانم شاه ولی

فهرست

۱	پرسش ۱. Control VAE
۱-۱	مقدمه
۲-۱	پیاده سازی VAE
۳-۱	ارزیابی مدل VAE
۴-۱	پیاده سازی Control VAE
۲	پرسش ۲ - معرفی Generative Adversarial Networks (GANs)
۲-۱	آموزش مدل GAN بر روی دیتاست MNIST
۱-۱-۲	پیاده سازی
۱	پرسش ۱:
۲	پرسش ۲:
	ارزیابی مدل
۲-۲	مدل Wasserstein GAN
۳-۲	مدل Self-Supervised GAN
۱-۳-۲	Generator
۲-۳-۲	Discriminator

شکل‌ها

شکل ۱- معماری مدل Control VAE..... ۱

شکل ۲. نمونه خروجی تولیدی..... ۷

جدول‌ها

جدول ۱- مشخصات پیاده سازی VAE	۲
جدول ۲. معماری generator مدل GAN	۵
جدول ۳. معماری discriminator مدل GAN	۵
جدول ۴. معماری generator مدل SSGAN	۸
جدول ۵. بلاک residual در generator مدل SSGAN	۸
جدول ۶. معماری discriminator مدل SSGAN	۹
جدول ۷. بلاک residual در discriminator مدل SSGAN	۱۰

قبل از پاسخ دادن به پرسش‌ها، موارد زیر را با دقت مطالعه نمایید:

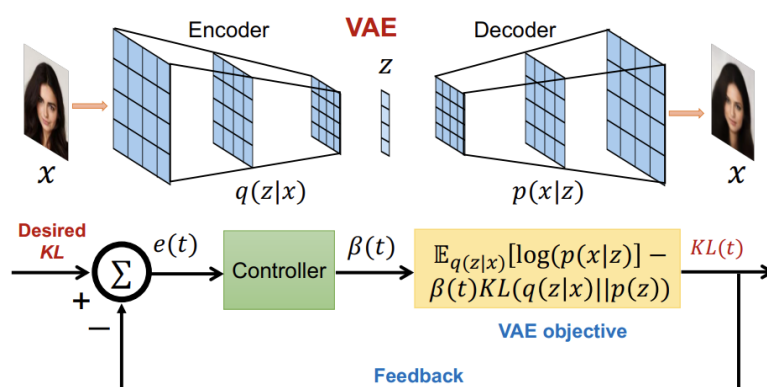
- **کیفیت گزارش شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است؛** بنابراین، لطفاً تمامی نکات و فرض‌هایی را که در پیاده‌سازی‌ها و محاسبات خود در نظر می‌گیرید در گزارش ذکر کنید.
- در گزارش خود مطابق با آنچه در قالب نمونه قرار داده شده، برای شکل‌ها زیرنویس و برای جدول‌ها بالانویس در نظر بگیرید.
- الزامی به ارائه توضیح جزئیات کد در گزارش نیست، اما باید نتایج بدست آمده از آن را گزارش و تحلیل کنید.
- **تحلیل نتایج الزامی می‌باشد، حتی اگر در صورت پرسش اشاره‌ای به آن نشده باشد.**
- **دستیاران آموزشی ملزم به اجرا کردن کدهای شما نیستند؛** بنابراین، هرگونه نتیجه و یا تحلیلی که در صورت پرسش از شما خواسته شده را به طور واضح و کامل در گزارش بیاورید.
- **کدها حتماً باید در قالب نوت‌بوک با پسوند ipynb تهیه شوند، در پایان کار، تمامی کد اجرا شود و خروجی هر سلول حتماً در این فایل ارسالی شما ذخیره شده باشد.** بنابراین برای مثال اگر خروجی سلولی یک نمودار است که در گزارش آورده‌اید، این نمودار باید هم در گزارش هم در نوت‌بوک کدها وجود داشته باشد.
- **در صورت مشاهده‌ی تقلب امتیاز تمامی افراد شرکت‌کننده در آن، 100- لحاظ می‌شود.**
- تنها زبان برنامه نویسی مجاز **Python** است.
- استفاده از کدهای آماده برای تمرین‌ها به هیچ وجه مجاز نیست.

در این تمرین با مدل Control VAE آشنا خواهید شد و به پیاده‌سازی این مدل خواهید پرداخت.

۱-۱. مقدمه

(۱۰ نمره)

در مورد ایده کلی این مقاله توضیح دهید که چگونه این مقاله می‌تواند در نهایت به مدل VAE برسد که هم کیفیت عکس‌های تولیدی و هم تنوع آن‌ها مناسب باشد؟



شکل ۱- معماری مدل Control VAE

برای آموزش مدل از بخشی از مجموعه داده [dSprites](#) (2D shape) استفاده شده است؛ به این صورت که از ویژگی orientation در این مجموعه داده مقدار صفر آن انتخاب شده است (که $\frac{1}{40}$ از کل داده‌ها است) و در نهایت ۱۸۴۳۲ داده خواهیم داشت. این مجموعه داده کوچک شده را می‌توانید از این [پیوند](#) دریافت کنید. تصاویر این مجموعه داده با اندازه ۶۴ در ۶۴ و تک کاناله هستند و مقادیر پیکسل‌های آن به صورت باینری (0 و 1) است.

¹ Shao, H., Yao, S., Sun, D., Zhang, A., Liu, S., Liu, D., ... & Abdelzaher, T. (2020, November). Controlvae: Controllable variational autoencoder. In International Conference on Machine Learning (pp. 8655-8664). PMLR.

پس از دریافت دادگان از پیوند بالا، با قطعه کد زیر می توانید آن را بارگذاری کنید.

```
file_name = "sh3_sc6_y32_x32_imgs.npz"
imgs = np.load(file_name, allow_pickle=True, encoding='latin1')
imgs = imgs['imgs']
```

۲-۱. پیاده سازی VAE

(۱۵ نمره)

یک مدل VAE را با معماری Encoder و Decoder مشخص شده بر در جدول ۶ مقاله پیاده سازی کنید. فقط توجه کنید که در سطر آخر از معماری داده شده Encoder از دو لایه fully connected با تعداد out feature با ترتیب برابر با ۲۵۶ و ۲۰ است. در گزارش خود نحوه پیاده سازی این قسمت را مختصراً توضیح دهید. از مشخصات جدول ۱ برای آموزش مدل استفاده کنید.

جدول ۱. مشخصات پیاده سازی VAE

Optimizer	Adam
Leaning rate	0.0001
Reconstruction loss	Bernoulli

توجه کنید در هنگام محاسبه هر کدام از خطاهای KL و reconstruction در هر batch از داده، آن را به صورت میانگین روی داده های batch در نظر بگیرید.

۳-۱. ارزیابی مدل VAE

(۴۰ نمره)

نمودار خطاهای KL، reconstruction و خطای کل در حین آموزش را در یک نمودار رسم کنید و آن را تحلیل کنید (مثلاً بیان کنید چرا بعضی از خطاها صعودی یا نزولی شده) (۱۰ نمره)

یک grid با اندازه 5×10 از عکسهای تولید شده که توسط مدل VAE آموزش داده شده، نشان دهید.

(۱۵ نمره)

در مورد معیار FID برای ارزیابی مدل‌های مولد تحقیق کنید و نحوه محاسبه آن را به صورت مختصر توضیح دهید. (۱۰ نمره)

با استفاده کتابخانه [pytorch-fid](#) مقدار FID را بدست بیاورید (تعداد عکس‌های تولیدی و تعداد عکس‌های حقیقی (به صورت تصادفی انتخاب می‌شود) را حداقل ۵۰۰۰ در نظر بگیرید. اگر از فریم ورکی غیر از PyTorch استفاده می‌کنید می‌توانید از پیاده‌سازی‌هایی که محاسبه شاخص FID سایر فریم ورک‌ها (مانند Keras و یا Tensorflow) بهره ببرید. (۵ نمره)

۴-۱. پیاده‌سازی Control VAE

(۳۵ نمره)

حال پس از آن که مدل VAE را پیاده‌سازی کردید باید به منظور پیاده‌سازی Control VAE باید PI Controller را طبق Algorithm 1 در مقاله به مدل VAE اضافه کنید. مقدار K_p و K_i در PI Controller را به ترتیب برابر با 0.01 و 0.001 در نظر بگیرید. در گزارش خود نحوه پیاده‌سازی این قسمت را مختصراً توضیح دهید. (۱۵ نمره)

مدل Control VAE را در دو حالت که desired KL (یا Set point برای KL) برابر با ۸ و ۱۴ است، در نظر بگیرید و برای هر حالت مدل را صورت جدا آموزش دهید و موارد زیر را گزارش کنید (۲۰ نمره):

۱- نمودارهای خطای KL و reconstruction

۲- معیار FID

(توجه) در محاسبه معیار FID در این بخش از سوال، داده‌های واقعی (دادهای مجموعه داده) باید همان‌هایی باشند که در محاسبه FID برای قسمت ۱-۳ سوال استفاده کردید.

(توجه) هم چنین در رسم نمودارها توجه داشته باشید که برای هر دو حالت Set point در یک نمودار رسم شوند مثلاً در نمودار خطای KL برای Set point ها ۸ و ۱۴ باید در یک نمودار باشند.

پرسش ۲ – معرفی Generative Adversarial Networks (GANs)

در این قسمت از تمرین قصد داریم تا با نحوه کارکرد یک مدل ساده از شبکه‌های GAN آشنا شویم. بدین منظور از شما می‌خواهیم تا یک شبکه GAN ساده‌ای را که به شما معرفی می‌کنیم را طراحی کرده و سپس آن را روی یک دیتاست نسبتاً سبک آموزش دهید و نتایج بدست آمده را ارزیابی نمایید. در نهایت از شما خواسته می‌شود تا بر روی این شبکه اصلاحاتی را اعمال کنید تا عملکرد آن بهبود پیدا کند.

۲-۱. آموزش مدل GAN بر روی دیتاست MNIST

برای برطرف شدن مشکل ناپدید شدن گرادیان^۱ در طول آموزش یک تابع خطای اشباع ناپذیر^۲ پیشنهاد می‌شود که به صورت زیر تعریف می‌گردد:

$$L_{\text{generator}}^{\text{ns}}(\theta; \phi) = -\mathbb{E}_{\mathbf{z} \sim \mathcal{N}(0, I)} [\log D_{\phi}(G_{\theta}(\mathbf{z}))]$$

برای تخمین mini-batch، از تخمین مونت کارلو از هدف یادگیری^۳ به صورت زیر بهره می‌بریم:

$$L_{\text{discriminator}}(\phi; \theta) \approx -\frac{1}{m} \sum_{i=1}^m \log D_{\phi}(\mathbf{x}^{(i)}) - \frac{1}{m} \sum_{i=1}^m \log(1 - D_{\phi}(G_{\theta}(\mathbf{z}^{(i)})))$$
$$L_{\text{generator}}^{\text{ns}}(\phi; \theta) \approx -\frac{1}{m} \sum_{i=1}^m \log D_{\phi}(G_{\theta}(\mathbf{z}^{(i)}))$$

for batch-size m , and batches of *real-data* $\mathbf{x}^{(i)} \sim p_{\text{data}}(\mathbf{x})$ and *fake-data* $\mathbf{z}^{(i)} \sim \mathcal{N}(0, I)$

^۱ vanishing gradients

^۲ non-saturating loss

^۳ learning objective

برای پیاده سازی generator و discriminator می‌توانید به ترتیب از معماری‌های نشان داده شده در جدول‌های ۲ و ۳ استفاده نمایید.

جدول ۲. معماری generator مدل GAN

	Name	In	Out	Batch Norm., Stride, Padding
1	Linear	64	512	BN
	ReLU	-	-	-
2	Linear	512	?	BN
	ReLU	-	-	-
3	PixelShuffle	-	-	-
4	Conv 3*3	16	32	BN, p=1
	ReLU	-	-	-
5	PixelShuffle	-	-	-
6	Conv 3*3	8	?	p=1

جدول ۳. معماری discriminator مدل GAN

	Name	In	Out	Stride, Padding
1	Conv 4*4	1	32	S=2, p=1
	ReLU	-	-	-
2	Conv 4*4	32	64	S=2, p=1
	ReLU	-	-	-
3	Linear	?	512	-
	ReLU	-	-	-
4	Linear	512	1	-

در جدول‌های ۲ و ۳ مقدار برخی موارد مشخص نیست و به جای آن علامت سوال قرار داده شده. یافتن مقادیر آن‌ها به عهده خودتان است. این مقادیر را بیابید، سپس طبق معماری‌های گفته شده پیاده‌سازی‌ها را انجام دهید.

۱-۱-۲. پیاده‌سازی

(۴۵ نمره)

با توجه به جزییات مطرح شده موارد زیر را انجام داده و به پرسش‌های گفته شده نیز پاسخ دهید:

پرسش ۱: یکی از عملگرهایی که در ساختار پیشنهادی Generator به شما معرفی شد، PixelShuffle بود. این مفهوم اولین بار به چه علت و برای چه کاربردی معرفی شد، در مورد نحوه عملکرد و تاثیر آن نیز توضیح دهید. (۵ نمره)

- کلاس‌های مربوط به Generator و Discriminator را با توجه به ساختارهای معرفی شده پیاده سازی کنید.

- تابع خطای مورد نیاز برای آموزش مدل را با توجه به فرمول داده شده پیاده سازی کنید.

پرسش ۲: مدل را با استفاده از دیتاست MNIST آموزش دهید و مقدار بهینه ابرپارامترها را بدست آورید و آن‌ها را گزارش کنید. (۱۰ نمره)

ارزیابی مدل

اکنون به ارزیابی مدل آموزش دیده خود بپردازید:

- پس از آموزش مدل به میزانی که کیفیت مطلوب حاصل شود، ابتدا نمودار تغییرات Loss در

طول آموزش را برای دو قسمت Discriminator و Generator ترسیم کنید. (۱۵ نمره)

- سپس خروجی مدل را به ازای عکس‌های تصادفی در سه مرحله مختلف ابتدایی، میانی و نهایی در قالب یک تصویر ۱۰ در ۱۰ مشابه شکل زیر گزارش دهید. (۱۰ نمره)

- یکی از پرکاربردترین معیارهای ارزیابی برای مدل‌های مولد، معیار FID score می باشد. عملکرد مدل خود را با محاسبه این معیار مورد بررسی قرار دهید. در نهایت، تحلیل خود را از خروجی بدست آمده با توجه به این معیار ارائه دهید. (۵ نمره)



شکل ۲. نمونه خروجی تولیدی

۲-۲. مدل Wasserstein GAN

(۲۵ نمره)

برای رفع مشکلات مطرح شده برای مدل اولیه GAN، نسخه‌های متعددی مطرح شده است. در این بخش می‌خواهیم به بررسی یکی از این نسخه‌ها، به نام Wasserstein GAN (WGAN)، بپردازیم. بنابراین به پرسش‌های زیر پاسخ دهید:

الف: این نسخه چه مشکلی از مدل GAN اولیه را برطرف کرده و به چه شکلی این کار را انجام می‌دهد؟

(۱۰ نمره)

ب: با توجه به تابع خطای معرفی شده در این روش، کد خود را اصلاح کرده، مدل را با تنظیمات جدید آموزش داده و بخش ۵ (ارزیابی مدل) سوال ۲-۱-۱ را تکرار کنید. (۱۵ نمره)

۲-۳. مدل Self-Supervised GAN

در این قسمت از تمرین قصد داریم تا با نحوه کارکرد یک مدل SSGAN آشنا شویم. بدین منظور از شما می‌خواهیم تا با استفاده از [مقاله](#) و آنچه به شما معرفی می‌کنیم، یک شبکه SSGAN طراحی کرده و سپس آن را روی دیتاست MNIST آموزش دهید و نتایج بدست آمده را مطابق بخش ۵ سوال ۲-۱-۱ ارزیابی کنید. (۳۰ نمره)

Generator ۱-۳-۲

برای پیاده سازی generator می‌توانید از معماری آورده شده در جدول ۴ استفاده کنید.

جدول ۴. معماری generator مدل SSGAN

	Name	In	Out	Batch Norm., Stride, Padding, Up-sampling
1	Linear	128	256*4*4	BN
2	Residual	256	256	UpS=T
3	Residual	256	256	UpS=T
4	Residual	256	256	UpS=T
5	ReLU	-	-	-
6	Conv 3*3	256	1	S=1, p=1
7	Tanh	-	-	-

برای پیاده سازی بلاک residual در generator می‌توانید از معماری گفته شده در جدول ۵ استفاده نمایید.

جدول ۵. بلاک residual در generator مدل SSGAN

	Name	In	Out	args
Block1	1 Batchnorm2D	in_channels	-	-
	2 ReLU	-	-	-
	3 Upsample	-	-	scale_factor=2 mode=nearest
	4 conv2D 3*3	in_channels	out_channels	strid=1 padding=1
	5 Batchnorm2D	num_features=out_channel	-	-
	6 ReLU	-	-	-
	7 conv2D 3*3	out_channels	out_channels	strid=1 padding=1
	8 Upsample	-	-	scale_factor=2 mode=nearest
output		Block1 + Layer Input		

برای هر بلاک Residual دو آرگومان ورودی in_channels و out_channels برای مشخص کردن سائز ورودی و خروجی آن داریم.

۲-۳-۲ Discriminator

برای پیاده سازی discriminator می‌توانید از معماری نشان داده شده در جدول ۶ استفاده کنید.

جدول ۶. معماری discriminator مدل SSGAN

	Name	In	Out	Batch Norm., Stride, Padding, Down-sampling
1	Residual	1	128	DnS=T
2	Residual	128	128	DnS=T
3	Residual	128	128	-
4	Residual	128	128	-
5	Linear	128	1	-
6	Linear	128	4	-

برای پیاده سازی بلاک residual در discriminator می‌توانید از معماری آورده شده در جدول ۷ استفاده نمایید.

جدول ۷. بلاک residual در discriminator مدل SSGAN

		Name	In	Out	args
Block1	1	ReLU* (not for the first Res Block)	-	-	-
	2	conv2D 3*3	in_channels	out_channels	strid=1 padding=1
	3	spectral_norm	-	-	-
	4	ReLU	-	-	-
	5	conv2D 3*3	out_channels	out_channels	strid=1 padding=1
	6	spectral_norm	-	-	-
	7	AvgPool2D 2*2	-	-	strid=2 padding=1
Block2	1	AvgPool2D 2*2	-	-	strid=2 padding=1
	2	conv2D 1*1	out_channels	out_channels	strid=1 padding=0
output			Block1 + Block2		

Good luck