به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

مدل‌های مولد عمیق

تمرین چهارم

نام و نام خانوادگی	سید نوید هاشمی
شماره دانشجویی	810101549
تاریخ ارسال گزارش	1404/11/25

فهرست

سؤال اول: مدل های دیفیوژن

بخش اول: سوالات تئوری

سوال اول

رابطه ای داده شده را به صورت زیر بازنویسی می کنیم:

$$x_t = \sqrt{\alpha_t}\, x_{t-1} + \sqrt{1-\alpha_t}\ \epsilon_t \qquad \epsilon_t \sim \mathcal{N}(0, I)$$

با توجه به این رابطه داریم:

$$\begin{aligned}
&x_0 \\
&x_1 = \sqrt{\alpha_1}\, x_0 + \sqrt{1-\alpha_1}\ \epsilon_1 \\
&x_2 = \sqrt{\alpha_2}\, x_1 + \sqrt{1-\alpha_2}\ \epsilon_2 = \sqrt{\alpha_2}\left(\sqrt{\alpha_1}\, x_0 + \sqrt{1-\alpha_1}\ \epsilon_1\right) + \sqrt{1-\alpha_2}\ \epsilon_2 \\
&\quad = \sqrt{\alpha_1\alpha_2}\, x_0 + \sqrt{\alpha_2(1-\alpha_1)}\ \epsilon_1 + \sqrt{1-\alpha_2}\ \epsilon_2
\end{aligned}$$

دو ترم آخر از هم مستقل هستند و هر دو گوسی هستند، میانگین هر دو 0 است پس میانگین توزیع جدید که از حاصل جمع اینها بدست میاید نیز 0 است، اما برای محاسبه واریانس توزیع جدید، باید واریانس این دو را جمع کنیم، خواهین داشت:

$$\sum_{new} = \alpha_2(1-\alpha_1)I + (1-\alpha_2)I = (1-\alpha_1\alpha_2)I$$

پس می توان اینگونه نوشت:

$$x_2 = \sqrt{\alpha_1\alpha_2}\, x_0 + \sqrt{1-\alpha_1\alpha_2}\ \epsilon_{new} \qquad \epsilon_{new} \sim \mathcal{N}(0, I)$$

به طرز مشابه می توان نوشت:

$$x_3 = \sqrt{\alpha_1\alpha_2\alpha_3}\, x_0 + \sqrt{1-\alpha_1\alpha_2\alpha_3}\ \epsilon \qquad \epsilon \sim \mathcal{N}(0, I)$$

و به همین صورت ثابت می شود:

$$x_t = \sqrt{\alpha_1\alpha_2 \dots \alpha_t}\, x_0 + \sqrt{1-\alpha_1\alpha_2 \dots \alpha_t}\ \epsilon \qquad \epsilon \sim \mathcal{N}(0, I)$$

$$x_t = \sqrt{\overline{\alpha_t}}\, x_0 + \sqrt{1-\overline{\alpha_t}}\ \epsilon \qquad \epsilon \sim \mathcal{N}(0, I)$$

که می دانیم می توان آن را به صورت زیر نوشت:

$$q(x_t|x_0) = \mathcal{N}(\sqrt{\overline{\alpha_t}}\, x_0\, , (1-\overline{\alpha_t})I)$$

سوال دوم

با توجه به مقاله DDPM و همچنین اسلاید های درس، این کار، مزیت های زیر را در پی دارد:

۱ .پایداری عددی و سهولت در آموزش : نویز هدف ($\in$) در تمام مراحل زمانی همواره دارای توزیع نرمال استاندارد است. این یعنی مقیاس خروجی مورد نظر شبکه همیشه ثابت است. در مقابل، میانگین یا تصویر در طول زمان تغییر مقیاس می‌دهند (مثلاً μ_θ در t های بزرگ به سمت صفر میل می‌کند). پیش‌بینی یک متغیر با توزیع ثابت و استاندارد برای شبکه عصبی بسیار ساده‌تر و پایدارتر از پیش‌بینی متغیری است که دامنه مقادیرش در طول زمان تغییر می‌کند.

۲ .بهبود کیفیت نمونه‌ها و ارتباط با Score Matching : پیش‌بینی نویز $\in$ منجر به یک تابع هزینه ساده‌شده (Simple Loss) می‌شود که طبق یافته‌های مقاله، کیفیت نمونه‌های تولیدی را نسبت به بهینه‌سازی مستقیم کران پایین تغییراتی (VLB) افزایش می‌دهد. این تابع هزینه ساده‌شده، عملاً ترم‌های مختلف Loss را دوباره وزن دهی می‌کند تا شبکه بر روی جنبه‌های بصری دشوارتر تمرکز کند. همچنین، پیش‌بینی نویز$\in$ از نظر ریاضی معادل یادگیری Score Function است که روشی قدرتمند و اثبات شده در مدل های مولد است.

سوال سوم

الف)

ترم‌های KL Divergence در کران پایین وریشنالی(VLB) ، وظیفه دارند فاصله بین توزیع معکوس مدل $(p_\theta(x_{t-1}|x_t))$ و توزیع پسین واقعی فرآیند forward $(q(x_{t-1}|x_t, x_0))$ را کمینه کنند .به بیان دقیق‌تر، از آنجا که فرآیند forward با افزودن نویز گوسی تعریف شده است، اگر x_0 معلوم باشد، توزیع بازگشت به مرحله قبل (q) به صورت فرم بسته و گوسی قابل محاسبه است .ترم‌های KL مدل را تشویق می‌کنند تا پارامتر های توزیع پیشنهادی خود (که دیدیم در اینجا تنها پارامترمان μ_θ بود) را طوری تنظیم کند که بر این توزیع واقعی به شرط x_0 منطبق شود.

ب)

طبق محاسبات مقاله DDPM ، می‌توان نشان داد که کمینه کردن ترمه ای KL معادل کمینه کردن تفاوت میانگین هاست که خود معادل کمینه کردن خطای پیش بینی نویز $(||\in - \in_\theta||^2)$ با ضرایب وزنی خاصی وابسته به زمان (t) است .در عمل، نادیده گرفتن این ضرایب وزنی و استفاده از Simple MSE، نتایج بهتری ایجاد می‌کند زیرا تابع هزینه VLB اصلی تمرکز زیادی بر مراحل زمانی اولیه (نویز کم) دارد که تأثیر کمی بر کیفیت ظاهری تصویر دارند؛ همچنین Simple MSE باعث نوعی وزن دهی دوباره می‌شود که

تمرکز آموزش را بر روی مراحل زمانی دشوارتر و تعیین‌کننده تر برای ساختار تصویر می‌گذارد و در نتیجه معیار کیفیت بصری(مانند FID) را بهبود می‌بخشد.

سوال چهارم

الف)

در فرمول نمونه‌برداریDDIM ، پارامتر η، واریانس نویز افزوده شده در فرآیند معکوس را کنترل می‌کند. رابطه کلی به‌روزرسانی x_{t-1} شامل یک ترم قطعی (بر اساس پیش‌بینی x_0 و جهت‌گیری به سمت x_t) و یک ترم تصادفی به صورت $\sigma_t\epsilon_t$ است .زمانی که $\eta = 0$ قرار داده می‌شود، مقدار σ_t برابر با صفر شده و ترم تصادفی از معادله حذف می‌گردد. در این حالت، نگاشت از x_t به x_{t-1}کاملاً قطعی می‌شود. این فرآیند معادل حل عددی یک معادله دیفرانسیل معمولی است که در آن، برای یک نویز اولیه ثابت ، مسیر تولید همواره یکسان بوده و خروجی نهایی منحصربه‌فرد خواهد بود.

ب)

مدل‌های DDPM بر اساس فرض مارکوفی بودن فرآیند معکوس بنا شده‌اند و برای اینکه توزیع معکوس بتواند تقریب خوبی از توزیع واقعی باشد، نیاز است که تغییرات در هر گام بسیار کوچک باشد (تعداد گام‌های زیاد، مثلاً ۱۰۰۰) .اما DDIM فرآیند را به صورت غیر مارکوفی فرمول‌بندی می‌کند که همان توزیع نهایی (Marginal) را دارد اما مسیر متفاوتی را طی می‌کند. با صفر کردن نویز (همانطور که دی قسمت قبل دیدیم با $\eta = 0$)، مسیر بازگشت در فضای نهان هموارتر شده و نوسانات تصادفی حذف می‌شوند. این همواری مسیر به ما اجازه می‌دهد که نویز را با گام‌های بزرگ‌تری حذف کنیم بدون اینکه خطای تقریب به شدت افزایش یابد. بنابراین می‌توان با پرش از روی گام‌های زمانی ، مثلا با ۲۰ تا ۵۰ گام، تصویری با کیفیت مشابه ۱۰۰۰ گام DDPM تولید کرد.

سوال پنجم

الف)

ایده اصلی روش Classifier-Free Guidance این است که به جای استفاده از یک شبکه عصبی Classifier جداگانه برای هدایت فرآیند تولید، از خود مدل مولد برای هدایت استفاده کنیم .در این روش، مدل دیفیوژن $\epsilon_\theta(x_t, t, c)$ به گونه‌ای آموزش داده می‌شود که بتواند هم به صورت شرطی (با برچسب کلاس یا متنc) و هم به صورت غیرشرطی (با برچسب تهی(∅)-به نوعی بدون برچسب) عمل کند (که این کار معمولاً با حذف تصادفی شرط در حین آموزش انجام می شود). در زمان نمونه‌برداری، تخمین نویز نهایی $\widetilde{\epsilon_\theta}$ ترکیبی خطی از تخمین شرطی و غیرشرطی است؛ به عبارتی داریم:

$$\widetilde{\epsilon_\theta}(x_t, c) = (1 + w)\epsilon_\theta(x_t, c) - w\epsilon_\theta(x_t, \emptyset)$$

می توان آن را بدین صورت نیز بازنویسی کرد:

$$\widetilde{\epsilon_\theta}(x_t, c) = \epsilon_\theta(x_t, \emptyset) + s.\left(\epsilon_\theta(x_t, c) - \epsilon_\theta(x_t, \emptyset)\right)$$

که در آن s (یا $w+1$ در رابطه اول) ضریب هدایت است .این روش را Classifier-Free می‌نامند زیرا برخلاف روش‌های پیشین (مانند Classifier Guidance در مقاله ذکر شده) که نیاز به آموزش یک مدل classifier جداگانه برای محاسبه گرادیان $\nabla_{x_t} \log p(y|x_t)$داشتند، در این روش هیچ شبکه classifier خارجی وجود ندارد و هدایت توسط خود مدل diffusion انجام می‌شود.

ب)

ضریب راهنمایی (w+1 در رابطه اول یا s در رابطه دوم در قسمت قبل) کنترل می‌کند که چقدر فرآیند تولید باید به سمت شرط داده شده متمایل شود.

اثر افزایش ضریب راهنمایی روی کیفیت: افزایش ضریب راهنمایی (s>1) باعث می‌شود نمونه های تولیدی تطابق بسیار بیشتری با کلاس یا متن ورودی داشته باشند و معمولاً کیفیت بصری (Visual Fidelity) افزایش یابد. در واقع مدل مجبور می‌شود به نواحی ای از توزیع احتمال برود که احتمال تعلق به آن کلاس بسیار زیاد است.

اثر افزایش ضریب راهنمایی روی تنوع: افزایش ضریب راهنمایی رابطه ای معکوس با تنوع دارد. با افزایش s، تنوع نمونه ها کاهش می‌یابد) و مدل تمایل پیدا می‌کند تا فقط نمونه های اصطلاحا استریوتایپ یا بسیار مشخص آن کلاس را تولید کند. همچنین اگر این ضریب بیش از حد زیاد شود، ممکن است باعث ایجاد تصاویر مصنوعی و غیرواقعی شود.

بخش دوم: سوالات عملی

زیربخش اول

در این زیربخش، از معماری شبکه عصبی U-Net برای یادگیری توزیع داده های مجموعه FashionMNIST استفاده شد. این شبکه شامل اتصالات میان‌بر است که اطلاعات مکانی تصویر را حفظ می‌کند. مدل برای ۱۰۰۰ گام زمانی طراحی شد و هدف آن پیش‌بینی نویز اعمال شده بر تصویر در هر مرحله بود. آموزش مدل به مدت ۲۰ ایپاک انجام شد و تابع هزینه همگرایی مناسبی را نشان داد (گرچه به علت طبیعت مدل های diffusion نمیتوان انتظار روند نزولی منظمی از آن داشت).

برای تولید تصاویر جدید، دو الگوریتم متفاوت پیاده‌سازی و مقایسه شدند:

نمونه‌برداری DDPM: این روش از الگوریتم استاندارد بازگشتی استفاده می‌کند که شامل ۱۰۰۰ مرحله است و در هر گام یک نویز تصادفی برای حفظ تنوع اضافه می‌کند.

Figure 1- نمونه‌های تولید شده با روش DDPM در ۱۰۰۰ گام زمانی. تنوع و وضوح تصاویر نشان‌دهنده آموزش موفق مدل است.

نمونه‌برداری DDIM: برای افزایش سرعت، از روش Deterministic استفاده شد که فرآیند تولید را تنها در ۲۰ گام انجام می‌دهد.

Figure 2 - نمونه‌های تولید شده با روش سریع DDIM تنها در ۲۰ گام. کیفیت بافت‌ها با وجود سرعت بسیار بالاتر حفظ شده است.

مقایسه زمان اجرا نشان‌دهنده تفاوت چشمگیر بین این دو روش است. همانطور که در نتایج ثبت شده است، تولید ۵۰ تصویر با روش DDPM حدود ۱۷.۳ ثانیه زمان برد، در حالی که روش DDIM همین تعداد تصویر را تنها در ۳۴۰ میلی‌ثانیه تولید کرد. این نشان‌دهنده افزایش سرعتی حدود ۵۰ برابر است.

```
%timeit -r 1 sample_ddim(50)
%timeit -r 1 sample_ddpm(50)

340 ms ± 0 ns per loop (mean ± std. dev. of 1 run, 1 loop each)
17.3 s ± 0 ns per loop (mean ± std. dev. of 1 run, 1 loop each)
```

Figure 3 - مقایسه زمان اجرای DDPM و DDIM

تصاویر خروجی هر دو روش کیفیت قابل قبولی دارند و دسته‌های مختلف (کیف، کفش، لباس) را به وضوح نمایش می‌دهند. همچنین مشاهده می شود که با وجود کاهش تعداد گام‌ها درDDIM ، افت کیفیت محسوسی مشاهده نمی‌شود.

در پایان این زیربخش، برای سنجش کمی کیفیت، امتیاز FID محاسبه شد که مقدار **۱۶.۲۸۴۸** بدست آمد. این عدد پایین نشان‌دهنده شباهت آماری بالای تصاویر تولید شده به تصاویر واقعی است.

زیربخش دوم

در این آزمایش، هدف آموزش مدل برای یادگیری یک سوژه خاص با استفاده از توکن sks بود. با مشاهده گرید های خروجی ، تفاوت های قابل توجهی بین سوژه اصلی و تصاویر تولید شده دیده می‌شود:

تصاویر تولید شده عمدتا چکمه‌های عمومی را نشان می‌دهند (مانند چکمه‌های پلاستیکی بارانی یا چکمه‌های خزدار زمستانی).

ویژگی‌های کلیدی سوژه، به ویژه بافت جین (Denim Texture) و رنگ آبی خاص (البته رنگ در گرید دوم کمی بهتر وجود دارد)، به درستی به فضای نهان منتقل نشده‌اند.

این پدیده ناشی از چند عامل فنی در فرآیند Fine-tuning است:

- **غلبه دانش پیشین :**مدل Stable Diffusion پیش از این آموزش دیده است که مفهوم چکمه در برف را با چکمه های سفید، خزدار یا لاستیکی ضخیم تداعی کند. از آنجایی که وزن Prior Loss در آموزش فعال بود، مدل سعی کرده دانش عمومی خود را حفظ کند و این دانش عمومی بر ویژگی‌های جدید (جین بودن چکمه) غلبه کرده است.
- **تضاد معنایی :**سوژه ما یک شیء غیرمعمول است. برای مدل، ترکیب چکمه و پارچه جین کمتر محتمل است تا چکمه و چرم/لاستیک. به همین دلیل، مدل در برابر تغییر وزن‌ها برای پذیرش این بافت جدید مقاومت کرده است.
- **محدودیت‌ها LoRA** : در اینجا از rank=4 استفاده کردیم که برای یادگیری مفاهیم کلی عالی است، اما ممکن است ظرفیت کافی برای ثبت جزئیات دقیق بافت مثل تار و پود پارچه جین را نداشته باشد.

در مجموع مدل توانسته است مفهوم کلی قرارگیری چکمه در موقعیت‌های مختلف را بازسازی کند، اما در مرحله Disentanglement(یعنی تفکیک سوژه خاص از کلاس عمومی) دچار چالش شده است. برای بهبود این نتیجه ، می‌توان Guidance Scale را کاهش داد تا خلاقیت مدل کنترل شود، یا تعداد قدم‌های آموزش و Rank شبکه LoRA را افزایش داد.

Figure 4 - تصویر نمونه از مجموعه داده آموزشی (Instance Data). سوژه یک چکمه با بافت خاص جین و بندهای بلند است که هدف مدل یادگیری این ویژگی‌های ظاهری متمایز بود.

Figure 5 - نتایج استنتاج مدل LoRA با پرامپت‌های متفاوت. همانطور که مشاهده می‌شود، مدل به جای بازتولید دقیق بافت "جین" سوژه اصلی، بیشتر تحت تأثیر دانش پیشین خود از کلاس عمومیچکمه قرار گرفته و خروجی‌ها را بر اساس کانتکست (مانند برف یا نقاشی) تطبیق داده است.

Figure 6 - نتایج دیگر با آموزش مجدد با بهبود پارامترها

سوال دوم: فلومچینگ و سری های زمانی

بخش اول: سوالات تئوری

سوال اول: میدان برداری

الف)

میدان برداری $v_\theta(x,t)$ در واقع تعیین‌کننده سرعت و جهت تغییرات ذرات داده در طول زمان است. این میدان برداری سمت راست یک ODE را به صورت زیر تعریف می‌کند :

$$\frac{dx}{dt} = v_\theta(x,t)$$

نقش اصلی این میدان، ایجاد یک نگاشت پیوسته و قطعی است. با انتگرال‌گیری از این میدان برداری در بازه زمانی $t \in [0,1]$، هر نمونه x_0 که از توزیع نویز اولیه نمونه برداری شده است، در امتداد خطوط جریان (Flow lines)حرکت کرده و به یک نمونه x_1 در توزیع داده های واقعی منتقل می‌شود .اگر بخواهیم شهود آن را توضیح دهیم ، این میدان برداری مانند جریان یک سیال عمل می‌کند که چگالی احتمال را از توزیع مبدا (نویز) به توزیع هدف (داده) هل می‌دهد.

ب)

تفاوت – تفاوت بنیادین در ماهیت فرآیند انتقال است. مدل‌های Diffusion مانند DDPM ذاتا بر پایه فرآیند های تصادفی (Stochastic) و زنجیره های مارکوف بنا شده اند که در هر گام نویز تصادفی اضافه یا کم می‌شود. در مقابل، Flow Matching (به‌ویژه در مرحله استنتاج) بر پایه دینامیک های قطعی و حل معادلات دیفرانسیل (ODE) استوار است .این تفاوت باعث می‌شود که Flow Matching انعطاف پذیری لازم

برای یادگیری مسیرهای هموارتر و مستقیم را داشته باشد که منجر به نمونه برداری سریع تر و پایدارتر می‌شود.

شباهت - این دو مدل زمانی رفتاری بسیار شبیه به هم پیدا می‌کنند که Probability Path در Flow Matchingرا منطبق بر مسیرهای Marginals مدل Diffusion انتخاب کنیم. می‌دانیم که برای هر فرآیند Diffusionتصادفی، یک ODE معادل وجود دارد که همان توزیع های احتمالی را تولید می‌کند .اگر Flow Matching را طوری تنظیم کنیم که میدان برداری آن معادل با Score Function مدل دیفیوژن شود (انتخاب مسیر گوسی)، آنگاه مدل Flow Matching دقیقا همان رفتار Probability Flow ODE مدل دیفیوژن را بازتولید می‌کند.

سوال دوم: مسیر های خطی

الف)

در روش Conditional Flow Matching ، تعریف مسیر به صورت درونیابی خطی میان نویز و داده یعنی $x_t = (1-t)x_0 + tx_1$، ساده ترین و مستقیم ترین روش برای اتصال توزیع مبدا به مقصد است .چراکه این انتخاب باعث می‌شود که میدان برداری شرطی هدف یعنی$u_t(x|x_1)$ مقدار ثابتی برابر $x_1 - x_0$ با داشته باشد. مشتق زمانی مسیر خطی نسبت به زمان t، مستقل از t است($\frac{dx}{dt} = x_1 - x_0$). این ویژگی باعث می‌شود که شبکه عصبی به جای یادگیری یک تابع پیچیده و متغیر با زمان، تنها نیاز به یادگیری یک جهت ثابت برای هر جفت داده داشته باشد که یادگیری آن بسیار ساده‌تر است و این کار بهینه است چراکه مسیر خطی در چارچوب نظریه Optimal Transport، مسیر بهینه محسوب می‌شود. چراکه در فضای اقلیدسی، خط راست کوتاه ترین فاصله بین دو نقطه است و کمترین انرژی جنبشی را برای انتقال جرم احتمالی از به مصرف می‌کند. در ادبیات Flow Matching ، این حالت خاص با نام OT-Flow Matching شناخته می‌شود که در آن ذرات داده در کوتاه‌ترین مسیر ممکن از نویز به داده حرکت می‌کنند.

ب)

در رابطه با **تابع هزینه** با انتخاب مسیر خطی، تابع هزینه پیچیده دیفرانسیلی به یک مسئله رگرسیون ساده با تابع زیان MSE تبدیل می‌شود:

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t,x_0,x_1}[||v_\theta(x,t) - (x_1 - x_0)||^2]$$

در اینجا هدف شبکه عصبی تنها تفاضل برداری $x_1 - x_0$ است که محاسبه آن بسیار ارزان و دقیق است.

در رابطه با **پایداری آموزش** در مقایسه با مدل‌های Diffusion که در آن‌ها تابع Score می‌تواند در نقاط مرزی t=0 یا t=1 به سمت بی‌نهایت میل کند و باعث ناپایداری گرادیان ها شود، میدان برداری حاصل از مسیرهای خطی همواره کراندار و هموار است. این ویژگی باعث می‌شود فرآیند آموزش بسیار پایدارتر باشد و همگرایی مدل با نرخ یادگیری‌های استاندارد سریع‌تر اتفاق بیفتد.

سوال سوم: تابع زیان

هدف درDenoising Score Matching ، آموزش مدل به گونه‌ای است که تابع امتیاز توزیع داده های نویزی شده را تخمین بزند. همانطور که می دانیم، تابع هزینه اصلی بر اساس Fisher Divergence میان امتیاز مدل و امتیاز واقعی تعریف می‌شود:

$$\mathcal{L} = \mathbb{E}_{t,x_0,x_t}[||s_\theta(x_t,t) - \nabla_{x_t} q(x_t|x_0)||^2]$$

همچنین با توجه به صورت سوال، توزیع انتقال از داده اصلی x_0 به داده نویزی x_t یک توزیع نرمال است. تابع چگالی احتمال برای این توزیع نرمال چندمتغیره به صورت زیر است:

$$q(x_t|x_0) = \frac{1}{(2\pi{\sigma_t}^2)^{d/2}} \exp\left(-\frac{||x_t - x_0||^2}{2{\sigma_t}^2}\right)$$

از طرفین لگاریتم می گیریم، داریم:

$$\log q(x_t|x_0) = -\frac{d}{2}\log(2\pi{\sigma_t}^2) - \frac{||x_t - x_0||^2}{2{\sigma_t}^2}$$

حال گرادیان عبارت فوق را نسبت به متغیر تصادفی x_t محاسبه می‌کنیم. از آنجا که جمله اول نسبت به x_t ثابت است، حذف می‌شود:

$$\nabla_{x_t} \log q(x_t|x_0) = \nabla_{x_t}\left(-\frac{(x_t - x_0)^T(x_t - x_0)}{2{\sigma_t}^2}\right)$$

$$\nabla_{x_t} \log q(x_t|x_0) = -\frac{1}{2{\sigma_t}^2} \,.\, 2(x_t - x_0)$$

$$= -\frac{x_t - x_0}{{\sigma_t}^2} = \frac{x_0 - x_t}{{\sigma_t}^2}$$

اکنون با جایگذاری مقدار محاسبه شده در معادله اصلی تابع هزینه، به فرم نهایی خواسته شده در صورت سوال می‌رسیم:

$$\mathcal{L} = \mathbb{E}_{t,x_0\sim p_{data},\epsilon\sim\mathcal{N}(0,I)}[||s_\theta(x_t,t) - \frac{x_0 - x_t}{{\sigma_t}^2}||^2]$$

بخش دوم: سوالات عملی

مرحله 1: ثبت و بررسی تابع هزینه در طول آموزش

در طول فرآیند آموزش مدلFlow Matching ، مقدار تابع هزینه بر پایه معیار میانگین مربعات خطا محاسبه شد. این مقدار در پایان هر Epoch برای داده‌های آموزش میانگین‌گیری و ثبت گردید. همچنین برای ارزیابی تعمیم پذیری مدل، در پایان هر Epoch مدل در حالت ارزیابی (Evaluation) قرار گرفت و مقدار Loss بر روی داده‌های آزمون که در فرآیند آموزش دیده نشده بودند محاسبه و ذخیره شد.

نتایج نهایی پس از ۱۰۰ Epoch به شرح زیر است:

Final Train Loss: 1.4507

Final Test Loss: 1.7657

Loss Gap: 0.3150

مرحله 2: رسم نمودار LOSS آموزش و آزمون

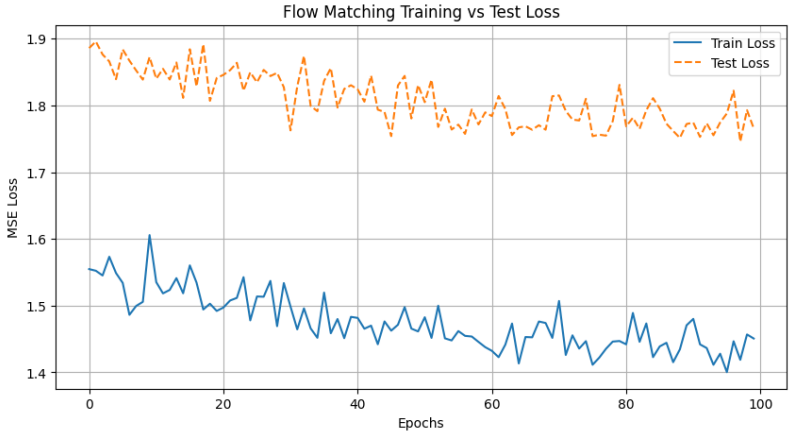

Figure 7 - نمودار روند تغییرات تابع هزینه (MSE Loss) در طول فرآیند آموزش مدل Flow Matching.

نمودار روند تغییرات تابع هزینه در طول ۱۰۰ Epoch رسم شده است. محور افقی نشان‌دهنده تعدادEpoch ها و محور عمودی نشان‌دهنده مقدار MSE Loss است.

همان‌طور که مشاهده می‌شود، نمودار Loss برای داده‌های آموزش (خط آبی) دارای یک روند نزولی پایدار است که نشان می‌دهد مدل به خوبی در حال یادگیری الگوی میدان برداری روی داده‌های آموزشی است. مقدار Loss از حدود ۱.۵۵ در ابتدای آموزش به ۱.۴۵ در انتها کاهش یافته است. خط نارنجی (Test Loss) همواره بالاتر از خط آبی قرار دارد که امری طبیعی است. این فاصله در طول

آموزش تقریبا ثابت مانده یا با شیب بسیار ملایمی نوسان داشته است (از حدود ۱.۸۸ به ۱.۷۶ رسیده است).

اگرچه فاصله‌ای (Gap) حدود ۰.۳ بین دو نمودار وجود دارد، اما نمودار Test Loss صعودی نشده است (یعنی واگرا نشده). اگر مدل دچار Overfitting شدید می‌شد، انتظار داشتیم با کاهشTrain Loss ، مقدار Test Lossشروع به افزایش کند. روند نزولی ملایم در Test Loss نشان می‌دهد مدل همچنان در حال بهبود تعمیم‌پذیری خود است. همچنین با توجه به اینکه مقادیر Loss هنوز نسبتاً بالا هستند (حدود ۱.۴۵)، این احتمال وجود دارد که ظرفیت مدل (تعداد پارامترها) یا تعدادEpoch ها برای یادگیری تمام پیچیدگی‌های سری زمانی مالی کافی نبوده باشد، یا نویز ذاتی بازارهای مالی (Stochasticity) مانع از کاهش بیشتر خطا می‌شود.

در مجموع می توان گفت مدل رفتار پایداری دارد و شکاف بین آموزش و آزمون در حد قابل قبولی است که نشان‌دهنده یادگیری الگوهای عمومی بدون حفظ کردن محض داده‌هاست.

مرحله 3: تولید داده های مصنوعی (SAMPLING)

پس از آموزش مدل، فرآیند تولید داده با حل عددی ODE زیر انجام شد:

$$\frac{dx}{dt} = v_\theta(x, t)$$

برای حل این معادله، از روش عددی اویلر استفاده شد. فرآیند تولید از نویز گوسی استاندارد در زمان t=0 آغاز شده و با گام‌های زمانی کوچک (Δt) به سمت t=1 پیش می‌رود. در هر گام، وضعیت جدید طبق رابطه زیر به‌روزرسانی می‌شود:

$$x_{t+\Delta t} = x_t + v_\theta(x_t, t).\Delta t$$

در این پیاده‌سازی از ۱۰۰ گام زمانی استفاده شد. تعداد گام‌های solver در واقع یک Trade-off بین کیفیت و سرعت است:

- **افزایش تعداد گام‌ها :** باعث کاهش خطای گسسته‌سازی می‌شود. مسیر انتگرال‌گیری دقیق‌تر طی شده و نمونه‌های تولید شده کیفیت و وفاداری بیشتری به توزیع واقعی خواهند داشت.
- **کاهش تعداد گام‌ها :**سرعت تولید را افزایش می‌دهد و هزینه محاسباتی را کم می‌کند، اما ممکن است باعث انحراف مسیر از واقعیت شده و نمونه‌های نهایی کیفیت پایین‌تری داشته باشند.

در مجموع می توان گفت انتخاب ۱۰۰ گام در اینجا تعادل مناسبی میان دقت تولید سری‌های زمانی مالی و زمان محاسبات ایجاد کرده است.

مرحله 4: نمایش نمونه های سری زمانی تولید شده

شکل زیر مقایسه‌ای بین ۵ نمونه تولید شده توسط مدل (ستون سمت چپ، سبز رنگ) و ۵ نمونه واقعی از مجموعه داده آزمون (ستون سمت راست، آبی رنگ) را پس از Denormalization نشان می‌دهد :

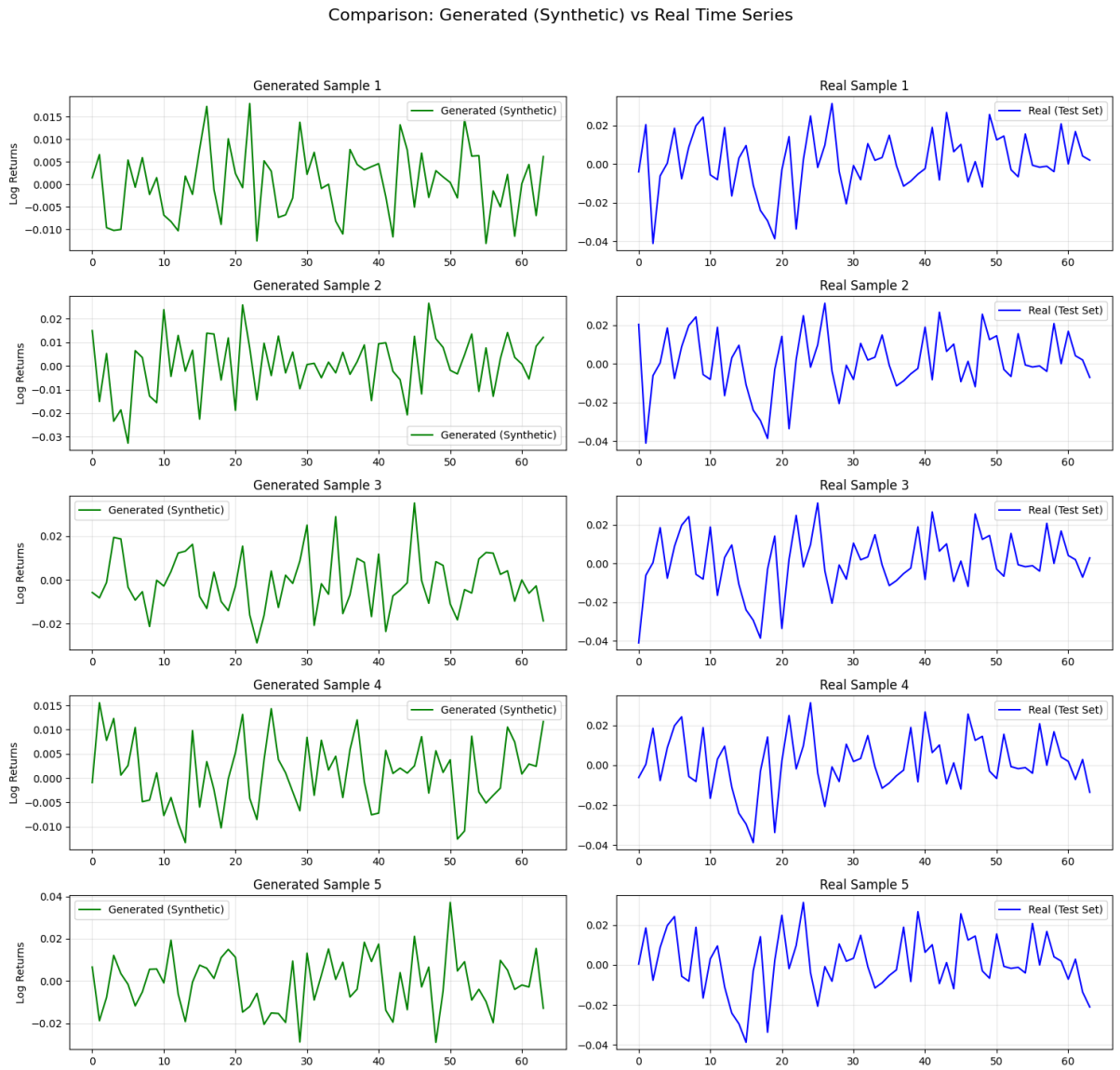

Figure 8 مقایسه کیفی سری‌های زمانی بازدهی لگاریتمی.(Log-Returns)

همان‌طور که مشاهده می‌شود، مدل توانسته است ویژگی‌های آماری کلی نظیر دامنه نوسان و آشوب‌ناکی داده‌های مالی را به خوبی بازتولید کند. همچنین بررسی‌های کیفی زیر قابل مشاهده است:

1. **سازگاری دامنه :** مقادیر بازدهی لگاریتمی در داده‌های تولید شده عمدتا در بازه [-0.03,+0.03] قرار دارند که دقیقا منطبق با دامنه تغییرات داده‌های واقعی SPY است. مدل به درستی مقیاس داده‌ها را یاد گرفته است.

2. **رفتار نوسانی :** سری‌های تولید شده همانند داده‌های واقعی، رفتاری پرنوسان و حول میانگین صفر دارند. جهش‌های ناگهانی که ویژگی ذاتی بازارهای مالی است، در نمونه‌های تولید شده نیز مشاهده می‌شود.
3. **تنوع :**نمونه‌های تولید شده کپی یکدیگر نیستند و هر کدام الگوی منحصربه‌فردی دارند، که نشان می‌دهد مدل دچار مشکل Mode Collapse نشده است.

مرحله 5: مقایسه کیفی سری های زمانی واقعی تولید شده

Figure 9 - مقایسه کیفی بافت و الگوی نوسانات. ستون سمت چپ (آبی) نمونه‌های واقعی از داده‌های آزمون SPYو ستون سمت راست (سبز) نمونه‌های مصنوعی تولید شده توسط مدل Flow Matching را نشان می‌دهد. شباهت در دامنه تغییرات و ساختار نوسانی مشهود است.

همانطور که در این نمودار مشخص است، سه نمونه از سری‌های زمانی واقعی (ستون چپ - آبی) در کنار سه نمونه تولید شده توسط مدل (ستون راست - سبز) نمایش داده شده است. با مقایسه دقیق بصری، موارد زیر قابل استنتاج است:

1. **حفظ الگوهای کلی :**مدل به خوبی توانسته است ماهیت تصادفی داده‌های مالی را یاد بگیرد. سری های تولید شده مانند نویز سفید ساده نیستند و ساختاری شبیه به سری‌های مالی دارند.
2. **بازتولید نوسانات :** خاصیت بازگشت به میانگین که در بازدهی‌های لگاریتمی واقعی دیده می‌شود (نوسان حول صفر)، در داده‌های مصنوعی نیز کاملا مشهود است.
3. **رفتار در نقاط جهش :**داده‌های واقعی دارای جهش‌های ناگهانی و تیز هستند (برای مثال در نمونه واقعی ۱). مدل نیز توانسته است جهش‌هایی با دامنه مشابه تولید کند (نمونه مصنوعی ۱ و ۲)، هرچند ممکن است فرکانس وقوع رخدادهای بسیار حدی در داده‌های مصنوعی کمی ملایم‌تر از داده‌های واقعی باشد که در مدل‌های مولد امری طبیعی است.

در مجموع از دید ناظر انسانی، تمایز قائل شدن بین سری‌های زمانی واقعی و مصنوعی دشوار است که نشان‌دهنده کیفیت بالای تولید مدل است.

مرحله 6: مقایسه توزیع داده های واقعی و تولید شده

نمودار زیر توزیع فراوانی و نمودار تخمین چگالی (KDE) بازدهی‌های لگاریتمی را برای داده‌های واقعی (آبی) و داده‌های تولید شده (سبز) مقایسه می‌کند.با بررسی دقیق موارد زیر قابل استنتاج است:

1. **انطباق مرکزی** :هر دو توزیع دقیقا حول میانگین صفر متمرکز هستند که نشان می‌دهد مدل بایاس ندارد.
2. **شکل توزیع :** منحنی سبز (داده‌های مصنوعی) با دقت بسیار بالایی بر منحنی آبی (داده‌های واقعی) منطبق است. این نشان می‌دهد مدل واریانس و انحراف معیار داده‌ها را به درستی یاد گرفته است.
3. **بررسی دنباله‌ها :**بازارهای مالی معمولا دارای توزیع با دنباله‌های پهن و کشیدگی (Kurtosis) زیاد هستند (قله تیزتر از توزیع نرمال). همان‌طور که در نمودار دیده می‌شود، قله نمودار آبی کمی تیزتر از نمودار سبز است. این یعنی مدل اندکی تمایل به نرمال‌سازی داده‌ها داشته و ممکن است رخدادهای خیلی نادر را با احتمال کمتری نسبت به واقعیت تولید کند، اما در مجموع تقریب بسیار قابل قبولی از توزیع کلی ارائه داده است.

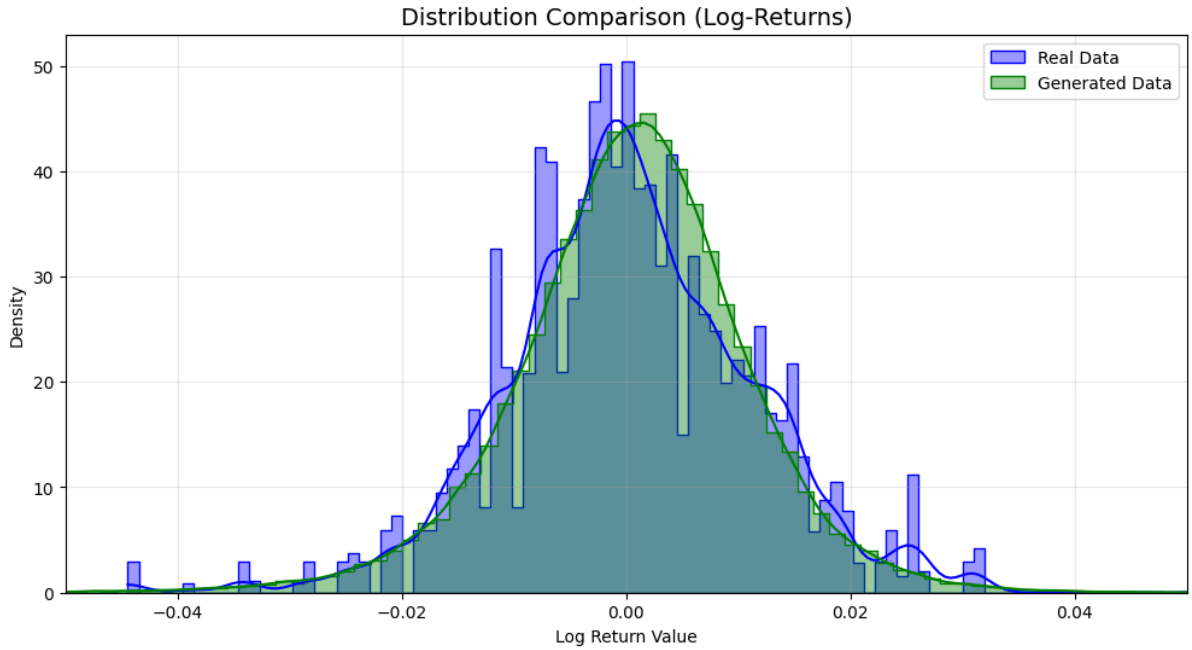

Figure 10 مقایسه توزیع آماری بازدهی لگاریتمی نمودار هیستوگرام و KDE برای داده‌های واقعی (آبی) و داده‌های تولید شده (سبز). انطباق قابل توجه دو نمودار نشان‌دهنده موفقیت مدل در یادگیری توزیع احتمالاتی حاکم بر داده‌های مالی است .

مرحله 7: ارزیابی آماری پایه و نوسانات

در جدول زیر معیارهای آماری محاسبه شده برای داده‌های واقعی و داده‌های تولید شده مقایسه شده‌اند:

معیار(Metric)	داده واقعی	داده تولید شده	وضعیت
میانگین(Mean)	0.000400	0.000438	بسیار نزدیک (Biais ناچیز)
واریانس(Variance)	0.000128	0.000108	اندکی کمتر در داده مصنوعی
نوسان(Volatility/Std)	0.011293	0.010391	خطای نسبی ~7.99%

تحلیل نتایج:

میانگین: میانگین هر دو سری بسیار نزدیک به صفر است (حدود 4×10^{-4}). این نشان می‌دهد که مدل به‌درستی ویژگی بازگشت به میانگین (Mean Reversion) یا خاصیت Martingale بازارهای مالی را یاد گرفته و دچار رانش غیرمنطقی نشده است.

نوسان: انحراف معیار داده‌های تولید شده (0.0104) بسیار نزدیک به داده‌های واقعی (0.0113) است. خطای نسبی حدود ۷.۹۹٪ نشان می‌دهد که مدل شدت نوسانات بازار را با دقت خوبی بازتولید کرده است.

واریانس و نوسان داده های مصنوعی اندکی کمتر از واقعیت است (Under-estimation). این مشاهدات با تحلیل هیستوگرام در مرحله قبل (که قله تیزتر و دنباله‌های کمی باریک‌تر را نشان می‌داد) هم‌خوانی دارد. به عبارت دیگر، مدل کمی محافظه کارانه عمل کرده و ریسک های حدی را کمتر از بازار واقعی تخمین زده است، اما بدنه اصلی توزیع نوسانات را به خوبی پوشش داده است.

مرحله 8: ارزیابی پیشرفته ساختار توزیعی و زمانی

در این مرحله، کیفیت داده‌های تولید شده با استفاده از معیارهای پیشرفته‌تری که هم شباهت توزیعی و هم وابستگی‌های زمانی را می‌سنجند، ارزیابی شد.

Sliced Wasserstein Distance – SWD: این معیار فاصله‌ی بین دو توزیع احتمال را در فضای چند بعدی اندازه‌گیری می‌کند. برخلاف مقایسه هیستوگرام که فقط توزیع مقادیر را به صورت تک‌ب عدی (بدون در نظر گرفتن زمان) بررسی می‌کرد، SWD توزیع مشترک (Joint Distribution) کل توالی ۶۴ تایی را در نظر می‌گیرد. مقدار محاسبه شده برابر با 0.000005 است؛ این مقدار بسیار نزدیک به صفر، نشان‌دهنده انطباق فوق‌العاده‌ی توزیع داده‌های تولید شده با داده‌های واقعی است. به بیان دیگر، نمونه‌های مصنوعی در فضای ویژگی ۶۴ بعدی، در همان ناحیه‌ای قرار دارند که داده‌های واقعی قرار دارند.

تحلیل خودهمبستگی(Autocorrelation): تابع خودهمبستگی (ACF) میزان وابستگی خطی بین مقادیر سری زمانی در لحظه t و لحظات قبل (t-k) را اندازه‌گیری می‌کند. این معیار برای بررسی اینکه آیا مدل توانسته الگوهای زمانی و حافظه داده‌ها را یاد بگیرد، حیاتی است. MSE بین منحنی ACF واقعی و مصنوعی برابر با 0.0051 است.

همان‌طور که در نمودار زیر مشاهده می‌شود، بازدهی های لگاریتمی واقعی (خط آبی) نوساناتی حول محور صفر دارند و همبستگی خطی قدرتمندی در لگ‌های مختلف نشان نمی‌دهند (که خاصیت ذاتی بازارهای مالی کارا است). داده‌های تولید شده (خط نارنجی) نیز رفتار مشابهی را نشان می‌دهند و نوسانات آن‌ها در همان محدوده [-0.1, 0.1] قرار دارد. اگرچه انطباق نقطه به نقطه کامل نیست (که در فرآیندهای تصادفی انتظار نمی‌رود)، اما مدل به درستی یاد گرفته است که داده‌ها نباید همبستگی طولانی‌مدت یا روند قطعی داشته باشند.

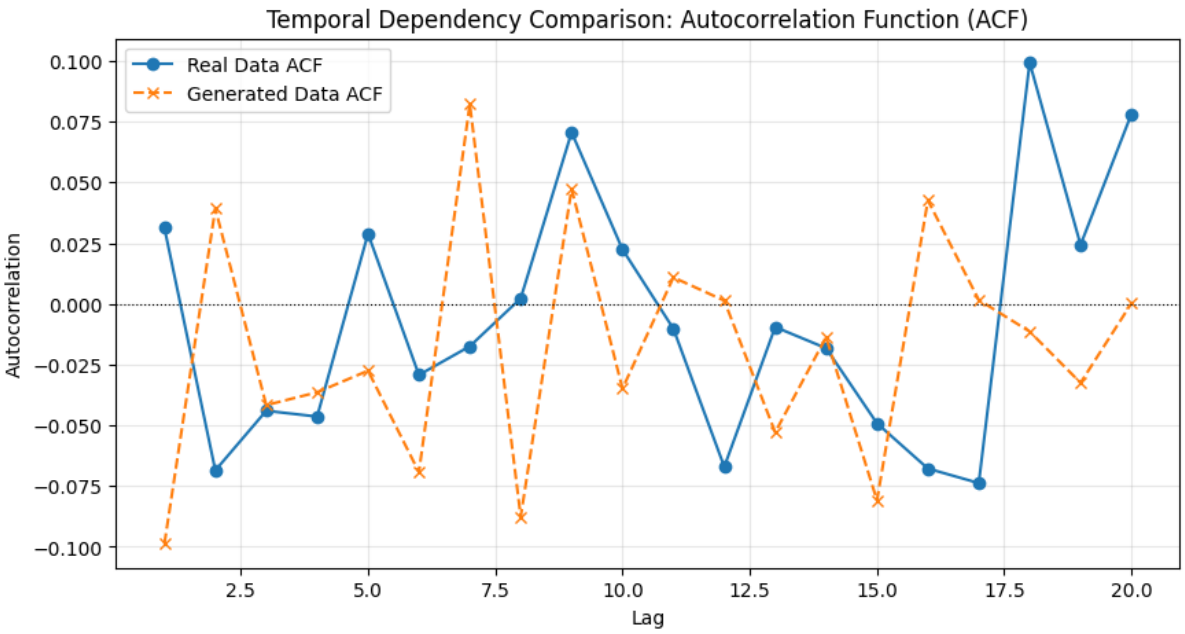

Figure 11 -مقایسه تابع خودهمبستگی (Autocorrelation Function) داده‌های واقعی و تولید شده. محور افقی لگ (تاخیر زمانی) و محور عمودی ضریب همبستگی را نشان می‌دهد. خط‌چین نارنجی (داده‌های مصنوعی) رفتار نوسانی مشابهی با خط آبی (داده‌های واقعی) دارد که نشان‌دهنده یادگیری صحیح ساختار وابستگی زمانی توسط مدل است.

در نهایت می توان گفت که مدل Flow Matching نه تنها در بازتولید توزیع آماری لحظه‌ای(هیستوگرام و SWD) موفق بوده است، بلکه توانسته دینامیک زمانی و ساختار همبستگی (ACF)پیچیده و تصادفی بازار مالی را نیز با دقت قابل قبولی شبیه‌سازی کند.