

به نام خدا



دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

تمرین کامپیوتری ۴

اصول سیستم های مخابراتی

دکتر صباغیان

عرفان پناهی ۸۱۰۱۹۸۳۶۹

نیمسال اول ۱۴۰۰-۰۱

فهرست:

*** فایل مربوط به این تمرین کامپیوتری با نام CA4_Matlab_810198369.mlx پیوست شده است.

چکیده صفحه ۲ ([لینک](#))

بخش اول صفحه ۳ ([لینک](#))

بخش دوم صفحه ۸ ([لینک](#))

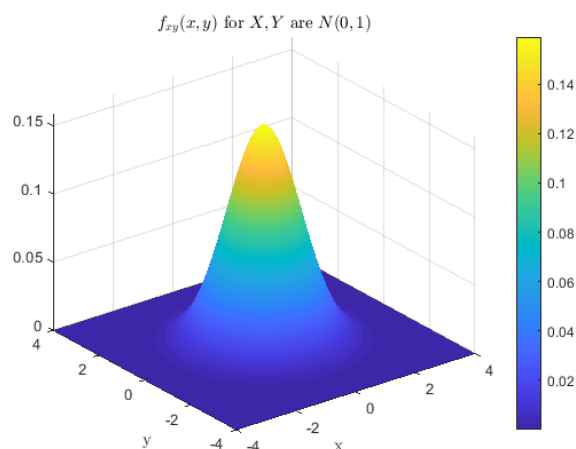
بخش سوم صفحه ۱۴ ([لینک](#))

چکیده: هدف از تمرین کامپیوتری ۴

هدف از انجام این تمرین مروری بر آمار و احتمالات مهندسی، بررسی فرایندهای تصادفی ایستا (WSS) و مدولاسیون پالس می باشد. در ابتدای کار و در بخش اول مروری کوتاه بر کلیات متغیرهای تصادفی نرمال و احتمالات شرطی و همبستگی خواهیم داشت. در بخش دوم فرایندهای تصادفی ایستا را بررسی کرده و تابع خودهمبستگی آنرا در دو بعد رسم میکنیم تا نشان دهیم به زمان بستگی ندارد. در نهایت و در بخش پایانی نیز با مدولاسیون پالس آشنا می شویم و سعی میکنیم یک فرستنده و گیرنده دیجیتال را در حضور نویز دیکودینگ پیاده سازی کنیم.

بخش اول : آمار و احتمالات

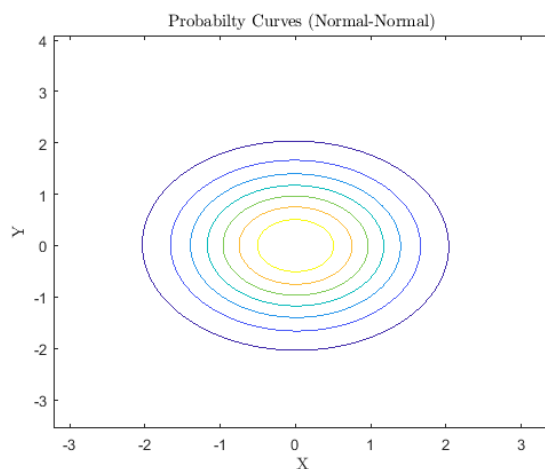
گام ۱. ابتدا براساس خواسته مسئله دو بردار تصادفی نرمال با طول بردار ۳۰۰۰ نمونه و توزیع گوسی با $\mu = 0$ و $\sigma = 1$ ایجاد کرده و سپس مطابق تصویر ۱-۱ تابع همبستگی این دو متغیر تصادفی را در ۳ بعد رسم می کنیم. با توجه به این موضوع که هر دو متغیر تصادفی نرمال هستند ضریب همبستگی آنها صفر است و در نتیجه ناهمبسته هستند. به عبارت دیگر انتظار داریم منحنی های هم احتمال به شکل دایره ای باشند.



تصویر ۱-۱ - تابع همبستگی دو متغیر تصادفی نرمال در ۳ بعد

همانطور که در تصویر بالا دیده می شود در دایره های دور رویه (که مثل کمر بند به دور رویه پیچیده) منحنی های هم احتمال دو توزیع رو نشان میدهند. برای تشخیص و رسم مجزای این منحنی ها با استفاده از دستور `contour()` تابعی به نام `Part1_1` نوشته شده است که دو متغیر تصادفی و میانگین های آنها را بعنوان ورودی گرفته و با محاسبه چگالی مشترک گوسی (با دستور `mvnpdf()`), منحنی های هم احتمال را رسم میکند.

تصویر ۱-۲ منحنی های هم احتمال دو متغیر تصادفی نرمال را نشان میدهد. همانطور که در تصویر ۱-۱ نیز مشاهده میشود، هرچه از شعاع صفر دور شویم، در حقیقت بخاطر گوسی بودن متغیر تصادفی مشترک، چگالی احتمال کمتر میشود. این موضوع در تصویر ۱-۲ به این نحو بیان میشود که فاصله بین منحنی های هم احتمال با دور شدن از مرکز بیشتر می شود.

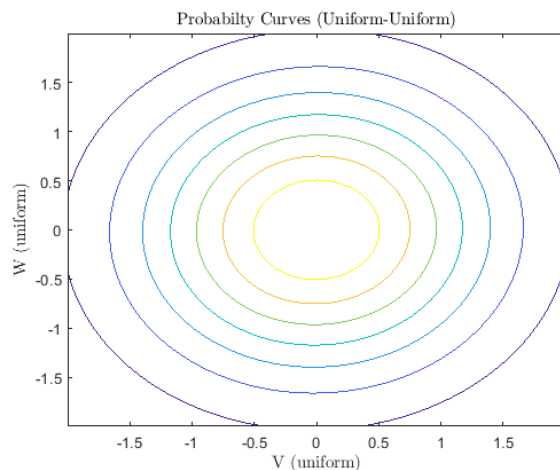


تصویر ۱-۲ - منحنی های هم احتمال دو متغیر تصادفی نرمال

گام ۲. حال دو متغیر تصادفی یکنواخت دیگر به صورت زیر تعریف میکنیم:

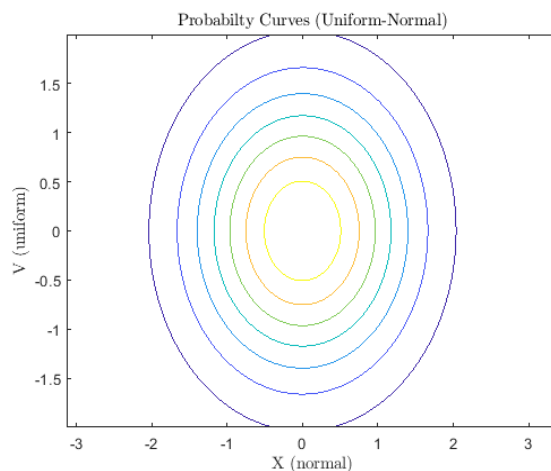
$$V \sim \text{uniform}[-2, 2] \quad , \quad W \sim \text{uniform}[-2, 2]$$

منحنی های هم احتمال متغیر های تصادفی V و W (یکنواخت-یکنواخت) در تصویر ۱-۳ نشان داده شده است. مطابق این تصویر میتوان گفت با توجه به اینکه در توزیع یکنواخت دیگر مثل توزیع گوسی با دور شدن از مرکز احتمال کاهش نمیابد و در تمام بازه احتمال یکسان است، بنابراین فاصله دو دایره متوالی با دور شدن از مرکز تغییر زیادی نمیکند.



تصویر ۱-۳ - منحنی های هم احتمال دو متغیر تصادفی یکنواخت

منحنی های هم احتمال متغیر های تصادفی V و X (یکنواخت-نرمال) در تصویر ۱-۴ نشان داده شده است.

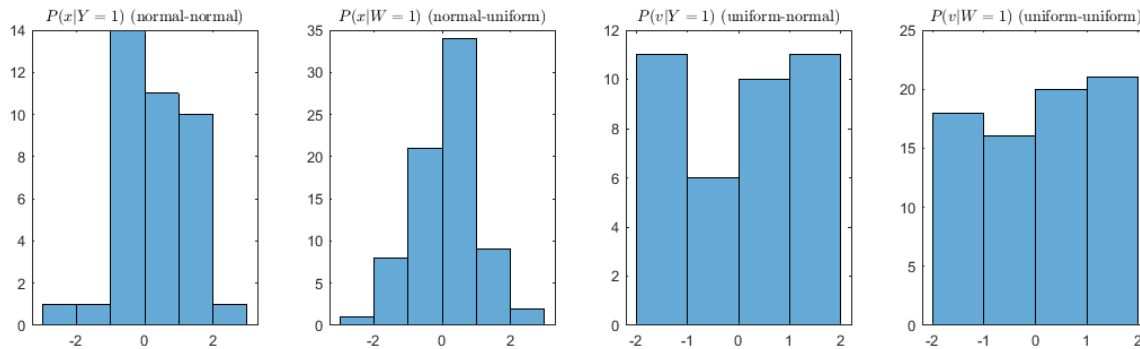


تصویر ۱-۴ - منحنی های هم احتمال دو متغیر تصادفی یکنواخت و نرمال

گام ۳. تابع مربوط به این بخش با نام $\text{Part1_3}()$ نوشته شده است. در این تابع $\delta = 0.1$ در نظر گرفته شده است.

علت استفاده از δ : در حقیقت با توجه به اینکه با متغیر های تصادفی گسسته سر و کار داریم، ممکن است دقیقاً در متغیر Y هیچگاه واقعه \mathcal{Y}_k رخ ندهد و این موضوع با محاسبات تئوری (چون با فرض پیوسته بودن متغیر تصادفی پیش میرویم) در تعارض است. پس یک بازه کوچک در نظر میگیریم که این مشکل به وجود نیاید.

نمودار هیستوگرام بردارهای تصادفی ۴ حالت (نرمال به شرط نرمال، نرمال به شرط یکنواخت، یکنواخت به شرط نرمال و یکنواخت به شرط یکنواخت) و با فرض $y_k = 1$ در تصویر ۵-۱ نشان داده شده است.



تصویر ۵-۱ - هیستوگرام برداری های تصادفی شرطی

گام ۴. ابتدا با تحلیل دستی میانگین و واریانس Z و کرلیشن Z و X را بدست می آوریم:

$$Z = aX + \sqrt{1 - a^2}Y$$

$$E[Z] = E[aX + \sqrt{1 - a^2}Y] = E[aX] + E[\sqrt{1 - a^2}Y] = aE[X] + \sqrt{1 - a^2}E[Y] = 0$$

$$\rightarrow E[Z] = 0$$

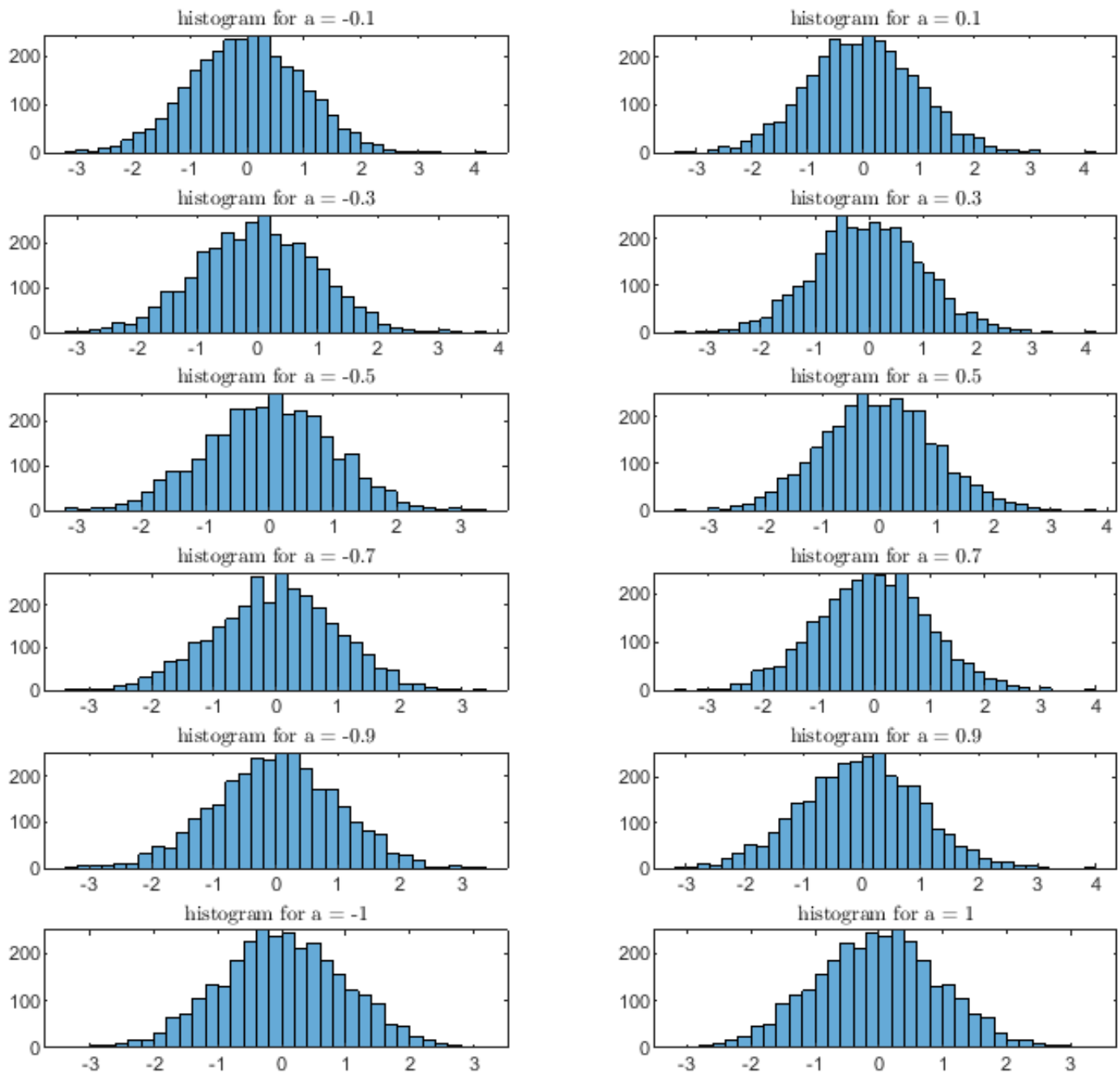
$$\begin{aligned}\sigma_Z^2 &= E[Z^2] - E^2[Z] = E[Z^2] = E[(aX + \sqrt{1 - a^2}Y)^2] \\ &= E[a^2X^2 + (1 - a^2)Y^2 + 2a\sqrt{1 - a^2}XY] \\ &= a^2E[X^2] + (1 - a^2)E[Y^2] + 2a\sqrt{1 - a^2}E[XY] \\ &= a^2\sigma_X^2 + (1 - a^2)\sigma_Y^2 + 2a\sqrt{1 - a^2}E[XY] = 1 + 2a\sqrt{1 - a^2}E[XY] = 1 \\ &\rightarrow \sigma_Z^2 = 1\end{aligned}$$

$$\begin{aligned}cov(x, z) &= E[XZ] - E[X]E[Z] = E[XZ] = E[X(aX + \sqrt{1 - a^2}Y)] \\ &= E[aX^2 + \sqrt{1 - a^2}XY] = aE[X^2] + \sqrt{1 - a^2}E[XY] = a\sigma_X^2 + \sqrt{1 - a^2}E[XY] \\ &= a + \sqrt{1 - a^2}E[XY] = a \\ &\rightarrow cov(x, z) = 1\end{aligned}$$

$$\rho_{ZX} = \frac{cov(x, z)}{\sigma_X\sigma_Z} = \frac{a}{1} = a \Rightarrow \rho_{ZX} = a$$

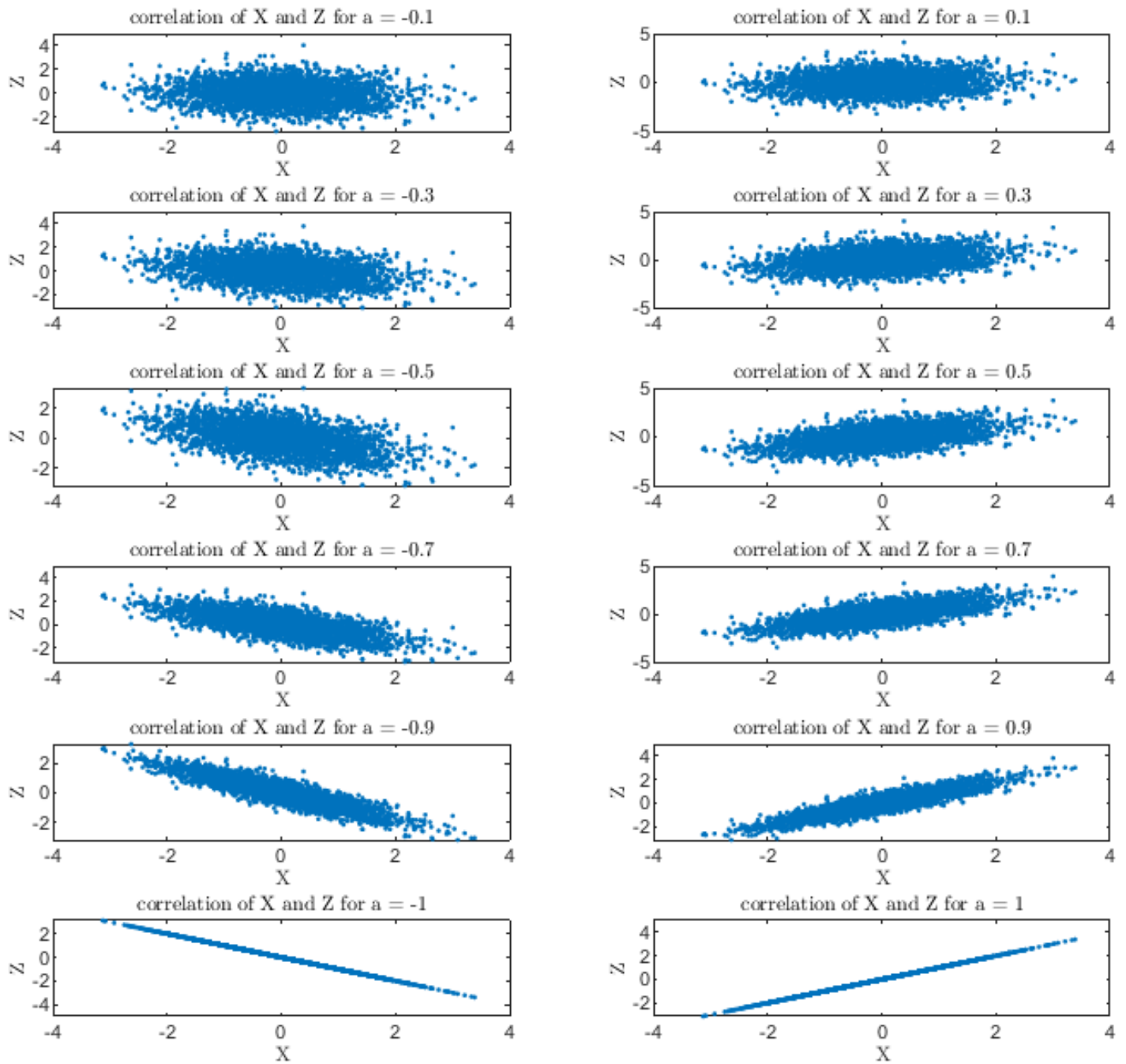
حال با استفاده از بردارهای تصادفی گام اول، بردار تصادفی Z را برای مقادیر خواسته شده a رسم میکنیم.

انتظار می‌رود که شکل گوسی هیستوگرام حفظ شود اما شباهت آن به هیستوگرام X با کم شدن ضریب a کمتر شود. تصویر ۶-۱ نمودار هیستوگرام این ۱۲ حالت را نشان می‌دهد.



تصویر ۶-۱ - هیستوگرام برداری های تصادفی Z

حال منحنی توزیع مشترک بین X و Z را بدست می‌آوریم و آنرا بصورت نقطه ای رسم می‌کنیم. همانطور که مشاهده می‌شود با بزرگ تر شدن ضریب پراکندگی نقاط کمتر شده و به عبارت دیگر همبستگی X و Z افزایش می‌یابد تا در حالتی که $a = \pm 1$ شود. در این صورت شاهد بیشترین همبستگی خواهیم بود. در حالتی که $a = 1$ باشد هر دو متغیر دقیقاً یکی هستند و در حالت $a = -1$ متغیر های تصادفی قرینه هم هستند.



تصویر ۷-۱ - منحنی نقطه ای توزیع مشترک های X و Z

بخش ۲: فرایندهای تصادفی

گام ۱. ابتدا با استفاده از روابط مربوط به امید ریاضی و خود همبستگی، میانگین و خود همبستگی فرایندهای داده شده را بدست می آوریم:

$$x_1(t, \phi_1) = \cos(2\pi t + \phi_1) \quad ; \quad \Phi_1 \sim U[-\pi, \pi]$$

$$E[X_{1t}] = E[\cos(2\pi t + \phi_1)] = \int_{-\infty}^{+\infty} \cos(2\pi t + \phi_1) \cdot f_{\Phi_1}(\phi_1) d\phi_1$$

$$= \frac{1}{2\pi} \int_{-\pi}^{+\pi} \cos(2\pi t + \phi_1) d\phi_1 = 0$$

$$R_{X_1}(t, t + \tau) = E[\cos(2\pi t + \phi_1) \cos(2\pi(t + \tau) + \phi_1)]$$

$$= \frac{1}{2} E[\cos(2\pi(2t + \tau) + \phi_1)] + \frac{1}{2} E[\cos(2\pi\tau)] = \frac{1}{2} \cos(2\pi\tau)$$

با توجه به اینکه میانگین و خودهمبستگی های فرایند تصادفی X_1 تابعی از زمان نیستند میتوان گفت این فرایند تصادفی ایستا است.

$$x_2(t, \phi_2) = \cos(2\pi t + \phi_2) \quad ; \quad \Phi_2 \sim U\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$$

$$E[X_{2t}] = E[\cos(2\pi t + \phi_2)] = \int_{-\infty}^{+\infty} \cos(2\pi t + \phi_2) \cdot f_{\Phi_2}(\phi_2) d\phi_1$$

$$= \frac{2}{\pi} \int_{-\frac{\pi}{4}}^{+\frac{\pi}{4}} \cos(2\pi t + \phi_2) d\phi_2 = \frac{2}{\pi} \left(\sin\left(2\pi t + \frac{\pi}{4}\right) - \sin\left(2\pi t - \frac{\pi}{4}\right) \right)$$

$$= \frac{2\sqrt{2}}{\pi} \cos(2\pi t)$$

$$R_{X_2}(t, t + \tau) = E[\cos(2\pi t + \phi_2) \cos(2\pi(t + \tau) + \phi_2)]$$

$$= \frac{1}{2} E[\cos(2\pi(2t + \tau) + \phi_2)] + \frac{1}{2} E[\cos(2\pi\tau)]$$

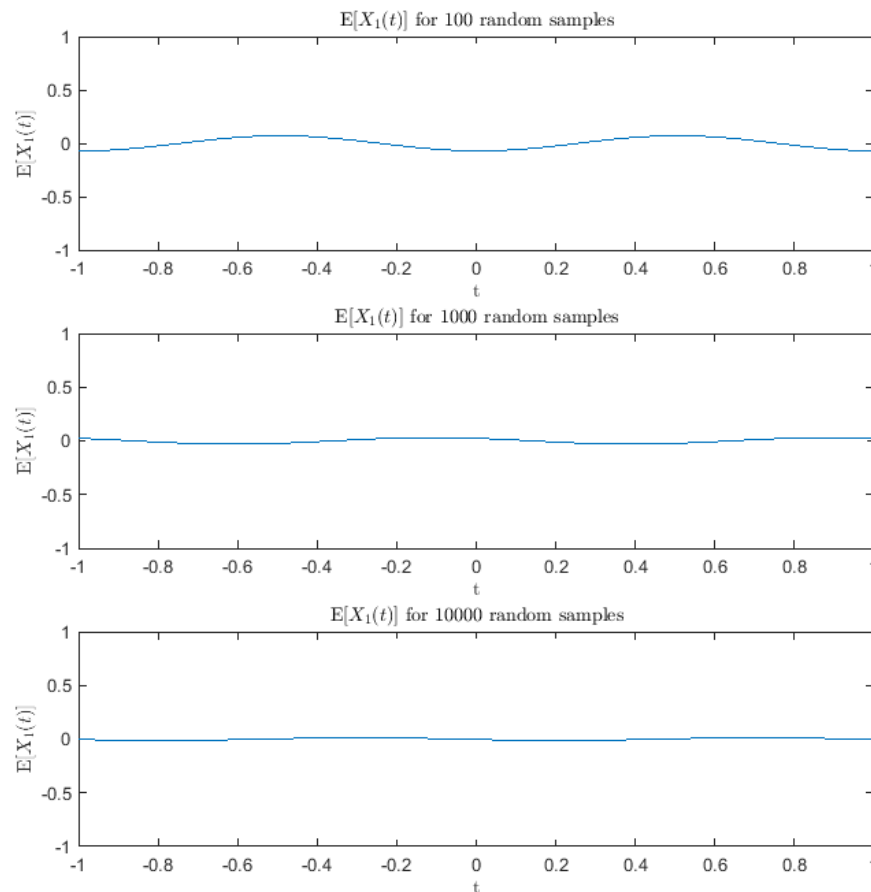
$$= \frac{2}{\pi} \int_{-\frac{\pi}{4}}^{+\frac{\pi}{4}} \cos(2\pi(2t + \tau) + \phi_2) d\phi_2 + \frac{1}{2} \cos(2\pi\tau)$$

$$= \frac{1}{\pi} \cos(2\pi(2t + \tau)) + \frac{1}{2} \cos(2\pi\tau)$$

با توجه به اینکه میانگین و خودهمبستگی های فرایند تصادفی X_2 هردو تابعی از زمان هستند میتوان گفت این فرایند تصادفی ایستا نیست.

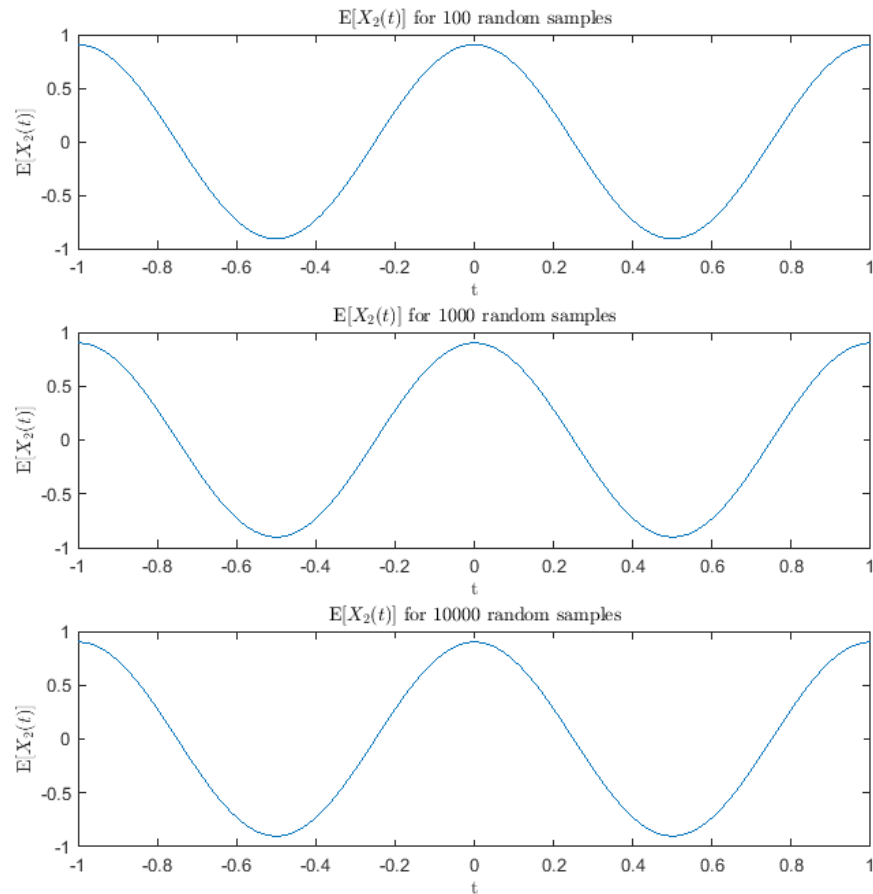
گام های ۲ و ۳. در این بخش پس از تعریف بردار زمان با فرکانس نمونه برداری ۱۰۰ هرتز، با استفاده از دستور `unifrnd()` ۶ بردار تصادفی تولید می کنیم. (سه تا برای Φ_1 و به طول های ۱۰۰، ۱۰۰۰ و ۱۰۰۰۰ و سه تا دیگر به همین ترتیب برای Φ_2). در نهایت با استفاده از ضرب ماتریسی (جهت افزایش سرعت پردازش در متلب از `for` استفاده نمیکنیم) میانگین هر فرایند را سه بار بدست آورده و مطابق با تصاویر ۲-۱ و ۲-۲ رسم میکنیم.

همانطور که در تصویر ۲-۱ مشاهده میشود برای فرایند تصادفی X_1 شاهد تغییرات ناچیز در حوزه زمان هستیم و میتوانیم بگوییم میانگین صفر است. (علت نوسان این است که ما در حالت واقعی ممکن است میانگین دقیقاً صفر برای توزیع هایمان نداشته باشیم که این موضوع همانطور که در تصویر نیز مشاهده میشود با افزایش طول بردار ها دقت میانگین افزایش می یابد).



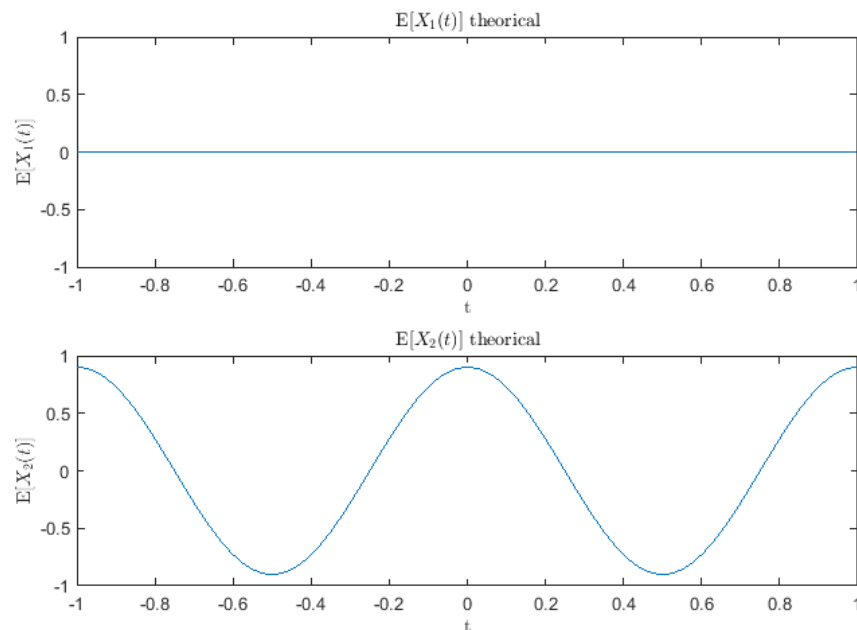
تصویر ۲-۱ - میانگین فرایند تصادفی X_1 با استفاده از متلب

برای فرایند تصادفی X_2 مطابق انتظار میبینیم که میانگین نسبت به زمان متغیر است و عبارت دیگر میانگین تابعی از زمان است و این موضوع ثابت میکند که این فرایند تصادفی ایستا نیست. (تصویر ۲-۲)



تصویر ۲-۲ - میانگین فرایند تصادفی X_2 با استفاده از متلب

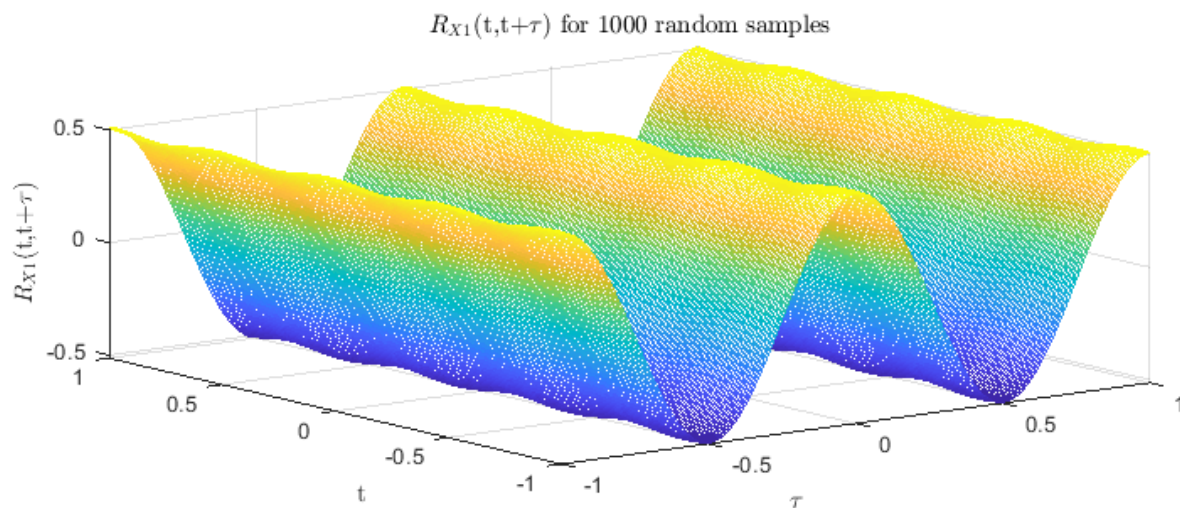
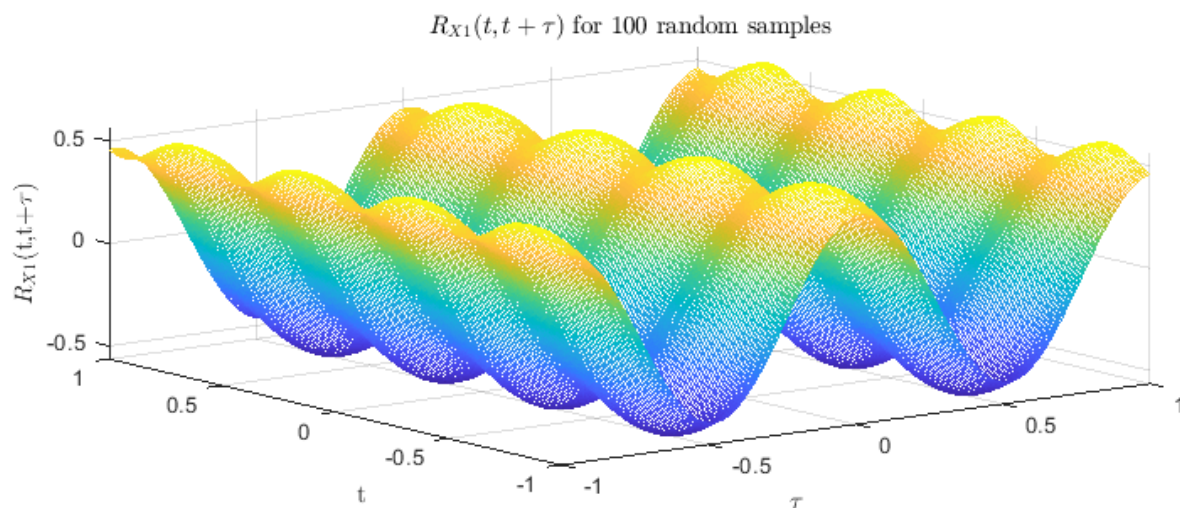
گام ۴. حال با استفاده از رابطه بدست آمده از گام ۱ نیز میانگین ها را مطابق تصویر ۲-۲ رسم میکنیم و مشاهده میکنیم که تقریباً با میانگین های بدست آمده از گام ۳ یکسان است.



تصویر ۲-۳ - میانگین فرایند های تصادفی با استفاده از محاسبات تئوری

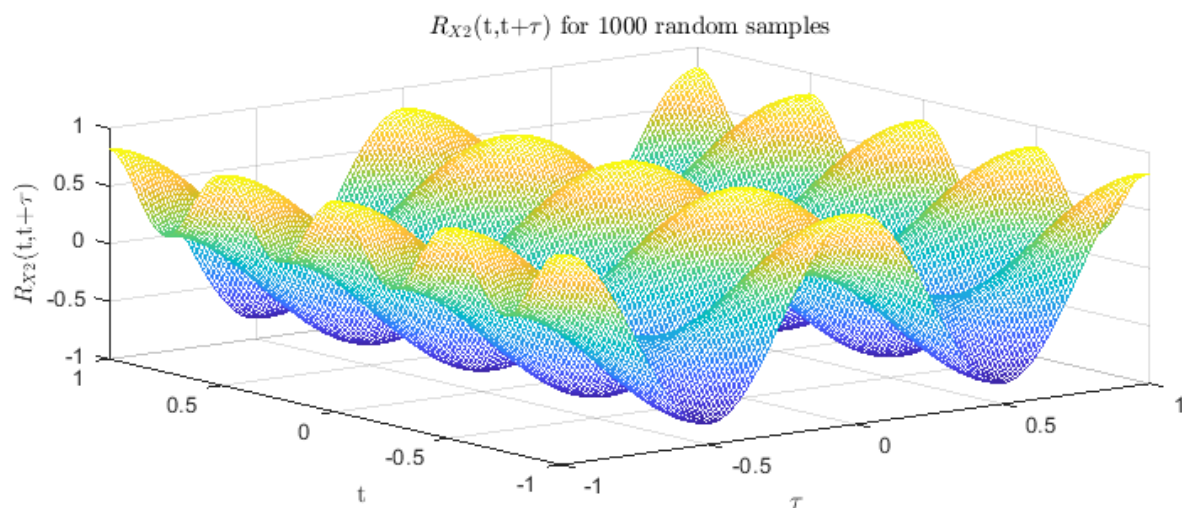
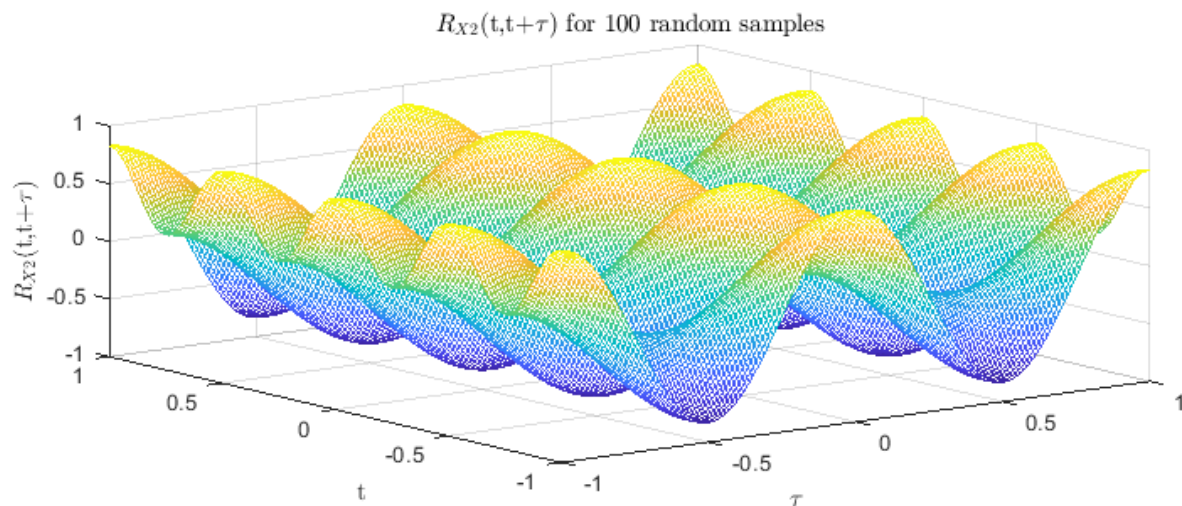
گام ۵. از تعریف تابع خودهمبستگی با استفاده از امید ریاضی در متلب خودهمبستگی هردو فرایند را برای هر دو طول بدست می آوریم. تصاویر ۲-۴ و ۲-۵ رویه های این ۴ حالت را نمایش میدهند.

همانطور که در تصویر ۲-۴ مشاهده میشود برای فرایند تصادفی X_1 شاهد تغییرات ناچیز در حوزه t هستیم و میتوانیم بگوییم خودهمبستگی فقط تابعی از τ است. (علت نوسان این است که ما در حالت واقعی ممکن است میانگین دقیقا صفر برای توزیع هایمان نداشته باشیم که این موضوع همانطور که در تصویر نیز مشاهده میشود با افزایش طول بردار ها دقت میانگین افزایش می یابد).



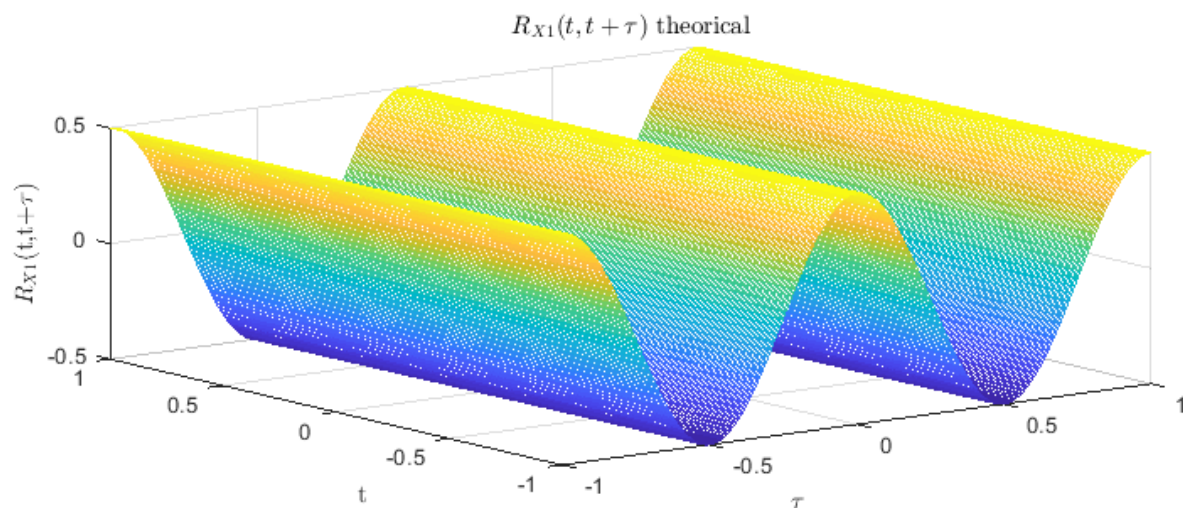
تصویر ۲-۴ - خودهمبستگی فرایند تصادفی X_1 با استفاده از متلب

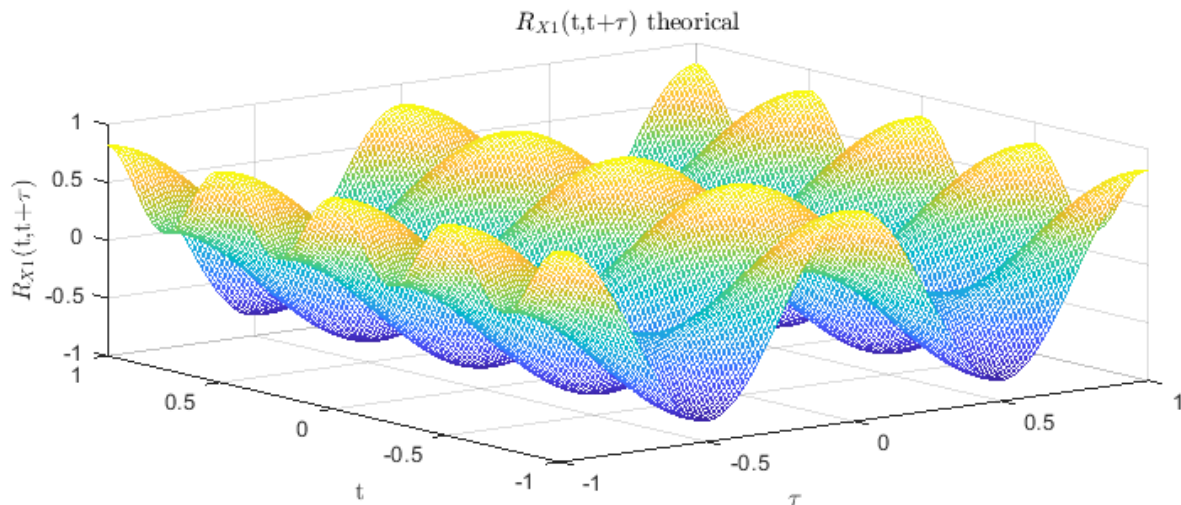
برای فرایند تصادفی X_2 مطابق انتظار می بینیم که خودهمبستگی نسبت به t متغیر است و عبارت دیگر میانگین تابعی از t است و این موضوع نیز ثابت میکند که این فرایند تصادفی ایستا نیست. (تصویر ۲-۵)



تصویر ۵-۲ - خودهمبستگی فرایند تصادفی X_2 با استفاده از متلب

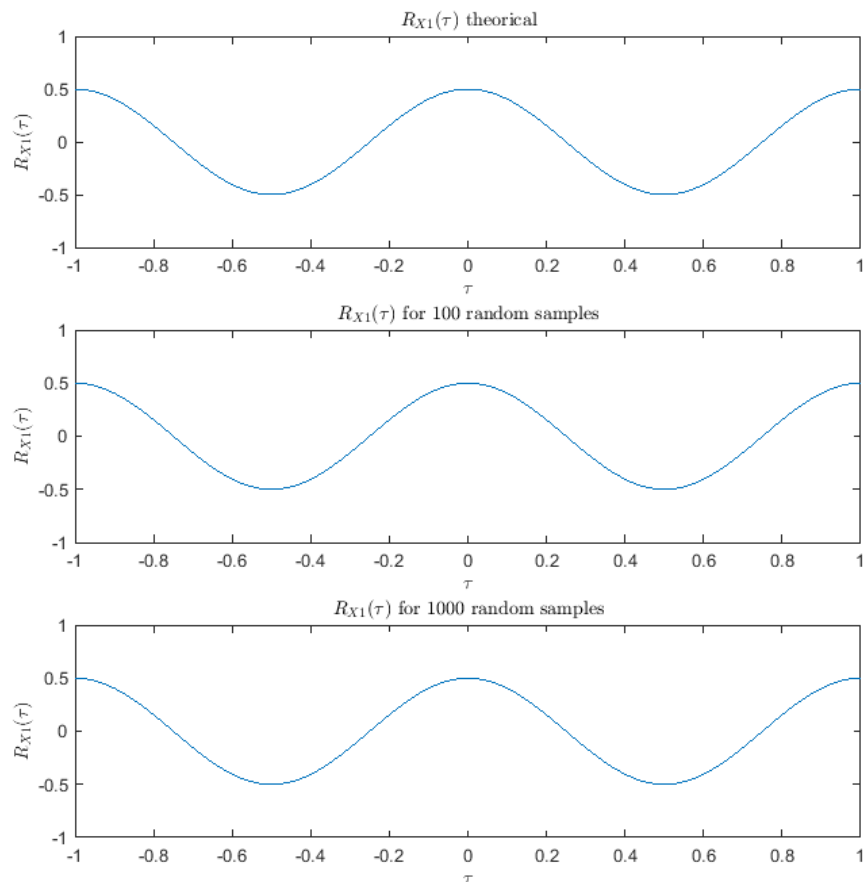
گام ۶. حال با استفاده از رابطه بدست آمده از گام ۱ نیز خودهمبستگی ها را مطابق تصویر ۶-۲ رسم میکنیم و مشاهده میکنیم که تقریباً با خودهمبستگی های بدست آمده از گام ۶ یکسان است.





تصویر ۲-۶ - خودهمبستگی فرایند های تصادفی با استفاده از محاسبات تئوری

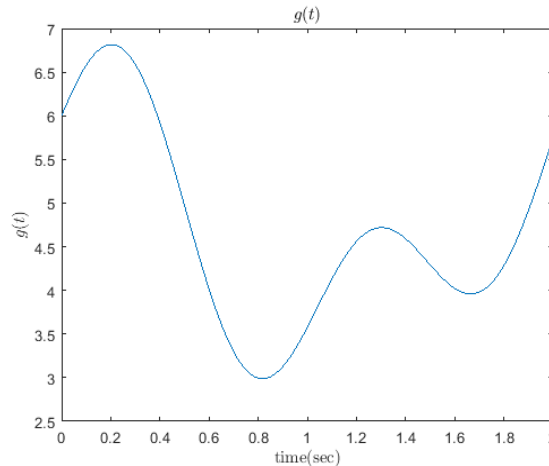
گام ۷. در نهایت با توجه به اینکه خودهمبستگی فرایند تصادفی X_1 تابعی از t نیست روی آن میانگین میگیریم و در دو بعد این همبستگی را رسم میکنیم. همانطور که در تصویر ۲-۷ مشاهده میشود هر سه منحنی با دقت خوبی با هم یکسان هستند.



تصویر ۲-۷ - مقایسه خودهمبستگی فرایند تصادفی ایستا با استفاده از محاسبات تئوری و مطلب با رسم در ۲ بعد

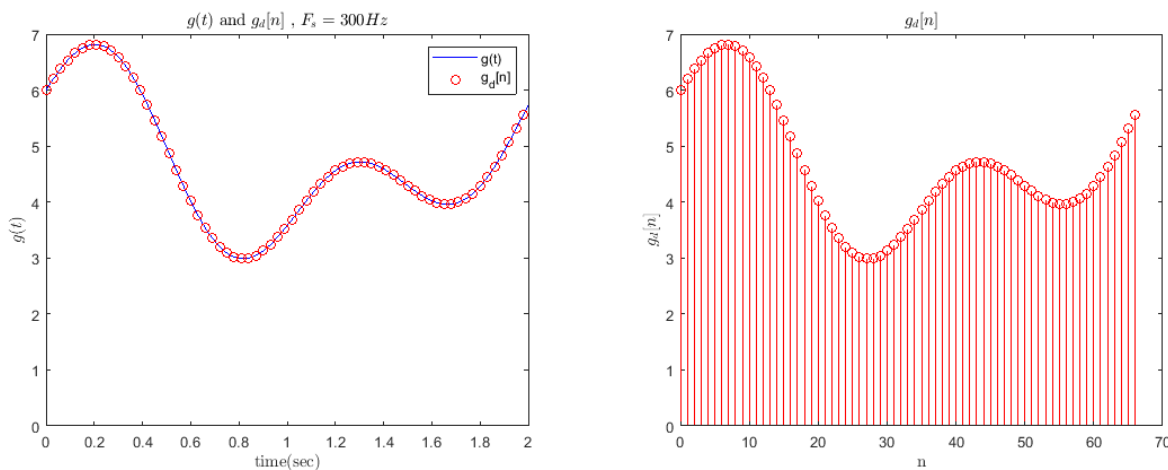
بخش ۳: کوانتیزاسیون**الف. تعریف سیگنال پیوسته و نمونه برداری و ساخت سیگنال گسسته:**

ابتدا مطابق خواسته مسئله سیگنال پیوسته را با فرکانس نمونه برداری بزرگ $10000Hz$ تعریف می کنیم. سیگنال پیوسته در تصویر ۳-۱ نشان داده شده است.



تصویر ۳-۱ - سیگنال پیوسته

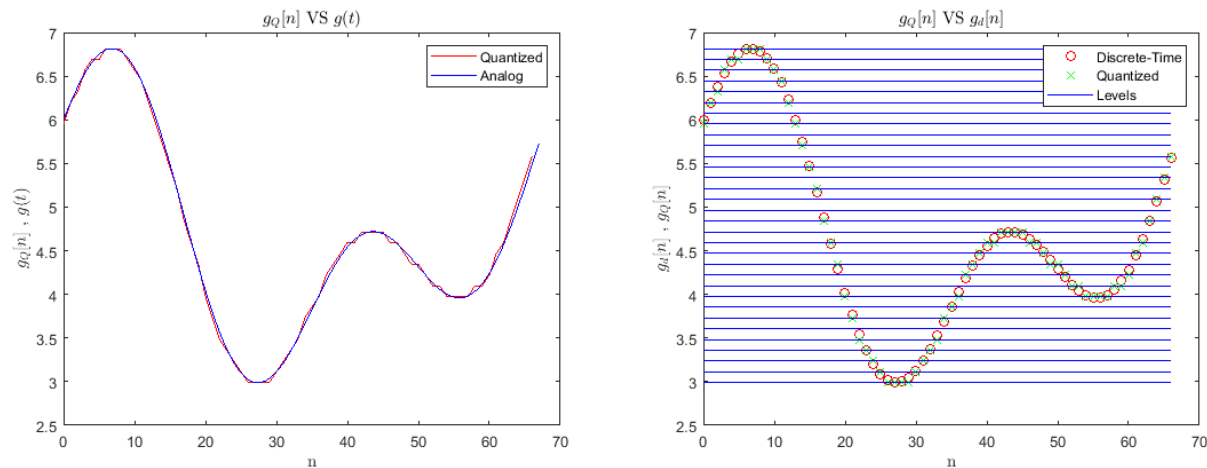
سپس با فرکانس نمونه برداری $300Hz$ از سیگنال پیوسته نمونه برداری کرده و سیگنال گسسته را میسازیم. تصویر ۳-۲ نمونه های برداشته شده از سیگنال پیوسته و بعد سیگنال گسسته را نشان میدهد.



تصویر ۳-۲ - سیگنال گسسته و نمونه های برداشته از سیگنال پیوسته

ب. اعمال سطح کوانتیزاسیون:

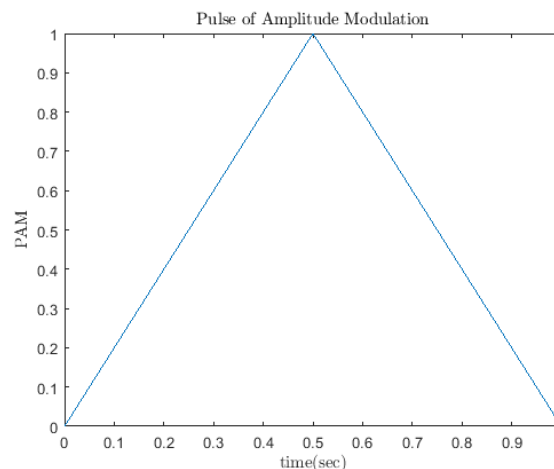
با توجه به خواسته سوال سطوح کوانتیزاسیون را در بازه دامنه سیگنال (بین حداقل و حداکثر) تعریف می کنیم. سپس در یک حلقه، هر مقدار از سیگنال گسسته را به این سطح ها تصویر میکنیم. برای این کار هر نقطه از سیگنال گسسته را به نزدیک ترین سطح کوانتیز می کنیم. تصویر ۳-۲ سیگنال کوانتیز شده را در کنار سَمپل های گسسته و سیگنال پیوسته نشان میدهد.



تصویر ۳-۳ - سیگنال کوانتایز شده را در کنار سمپل های گسسته و سیگنال

ج. دیجیتال سازی سیگنال کوانتایز شده:

ابتدا یک پالس مشابه با فایل (pulse.mat) با فرکانس 1000Hz و در ۱ ثانیه مطابق تصویر ۳-۴ ایجاد میکنیم.



تصویر ۳-۴ - پالس پایه

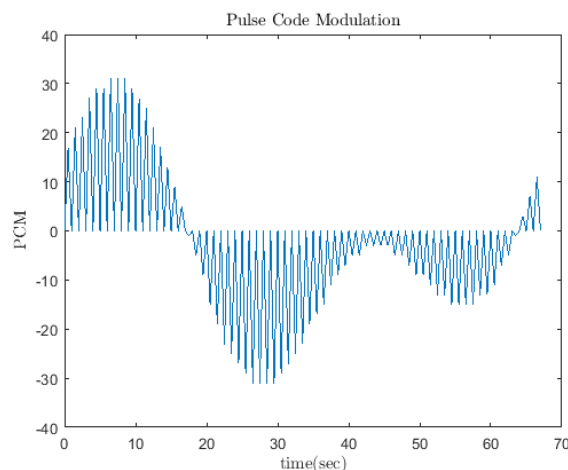
سپس انرژی سیگنال را مطابق با تصویر ۳-۵ بدست می آوریم.

```
PAM_Energy = sum(PAM.^2);
fprintf('Energy of DT-PAM = %.3f',PAM_Energy);
```

Energy of DT-PAM = 333.334

تصویر ۳-۵ - انرژی پالس پایه

حال به هریک از سطح ها یک دامنه دیجیتال از 31- (متناظر با سطح اول) با گام ۲ و تا 31 (متناظر با سطح آخر) اختصاص میدهم. سپس هر سمپل کوانتایز شده را با دامنه دیجیتال خود متناظر میکنیم و پالس ایجاد شده در قسمت قبل را در این دامنه دیجیتال ضرب میکنیم تا شکل سیگنال دیجیتال شده مطابق با تصویر ۳-۶ بدست آید.



تصویر ۶-۳ - سیگنال دیجیتال ارسال شده برای گیرنده

سپس هر یک از سیگنال ها را به روش گری کد برچسب گذاری کرده و به صورت زیر برای گیرنده ارسال میکنیم:

```
10100 10111 10110 10011 10001 10001 10000 10000 10000 10001 10011 10010 10111 10100 11101 11110 11011 01000
01011 01110 01100 00101 00110 00010 00011 00001 00000 00000 00000 00000 ... (matlab file)
```

د. دریافت سیگنال دیجیتال در گیرنده:

با در دست داشتن نسبت سیگنال به نویز در ورودی گیرنده (فرض اینکه فقط نویز دیکودینگ داریم) و مشخص بودن تعداد بیت ها میتوانیم تشخیص دهیم که توان نویز چقدر است. سپس با استفاده از آن میتوانیم نویز گوسی ایجاد شده را تشخیص دهیم و به سیگنال اعمال کنیم. با توجه به اینکه ۵ بیت ورودی و نتیجتاً ۳۲ دامنه مختلف داریم، بررسی نویز با استفاده از روابط ریاضی مربوط به نویز دیکودینگ کمی دشوار و طولانی خواهد بود. راه ساده تر این است که با استفاده از توان سیگنال دیجیتال بدست آمده از قسمت قبل و نسبت سیگنال به نویز، توان نویز را بدست آوریم. پس داریم:

$$(SNR)_D = \frac{S_D}{N_D} \rightarrow N_D = \frac{(SNR)_D}{S_D}$$

S_D توان سیگنال دیجیتال شده در قسمت قبل است. با توجه به اینکه در حقیقت و در پردازش متلب این سیگنال گسسته است پس با داشتن مربع دامنه در هر نقطه و همچنین تعداد کل نقاط میتوانیم این توان را بدست آوریم.

$$S_D = \frac{\sigma_s^2}{N(\text{number of samples})}$$

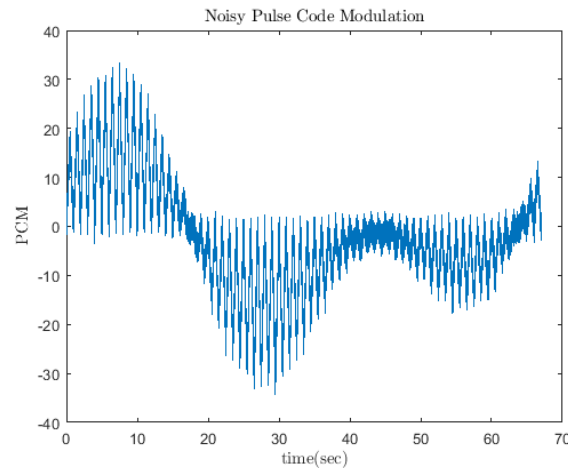
در نهایت زمانی که توان نویز را بدست آوردیم میتوانیم یک متغیر تصادفی گوسی با انحراف معیار بدست آمده از توان آن و میانگین صفر تعریف کنیم. تصویر ۷-۳ واریانس نویز را نشان میدهد.

```
fprintf('Variance = %.3f',var);
```

```
Variance = 1.158
```

تصویر ۷-۳ - واریانس نویز دیکودینگ

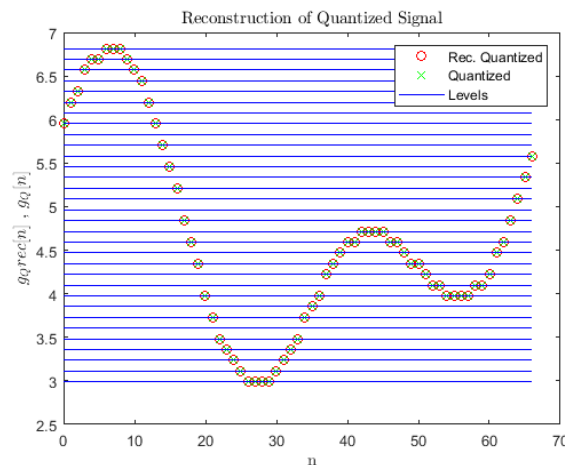
سپس این متغیر تصادفی که بیانگر نویز دیکودینگ است را به سیگنال دیجیتال شده جمع میکنیم تا سیگنال ورودی گیرنده شبیه سازی شود. تصویر ۸-۳ سیگنال ورودی گیرنده را که شامل نویز و سیگنال دیجیتال است نشان میدهد.



تصویر ۸-۳ - سیگنال ورودی گیرنده در حضور نویز

۵. دیکود کردن سیگنال دیجیتال در گیرنده:

حال می‌خواهیم با محاسبه انرژی متقابل هر ثانبه از سیگنال دیجیتال و پالس پایه، و همچنین انرژی پالس پایه به طریقی دامنه هر پالس را بدست آوریم. سپس از این طریق می‌توانیم به دیجیت هر سطح کوانتیزاسیون دست یابیم. (دقیقاً برعکس کاری که در دیجیتال سازی کردیم). سپس با داشتن دیجیت هر سطح می‌توانیم به مقدار واقعی آن سطح برسیم و سیگنال کوانتیز شده را بازیابی کنیم. تصویر ۹-۳ سیگنال کوانتیزه بازیابی شده در گیرنده را نشان می‌دهد.



تصویر ۹-۳ - سیگنال کوانتیزه بازیابی شده در گیرنده

۶. (امتیازی) تبدیل سیگنال کوانتایی شده به آنالوگ:

در این قسمت ابتدا سیگنال گسسته را طوری کوانتیزه می‌کنیم که هر سمپل به سطح پایینی خود تصویر شود. تصویر ۱۰-۳ شکل سیگنال کوانتیزه شده به این صورت را نشان می‌دهد.

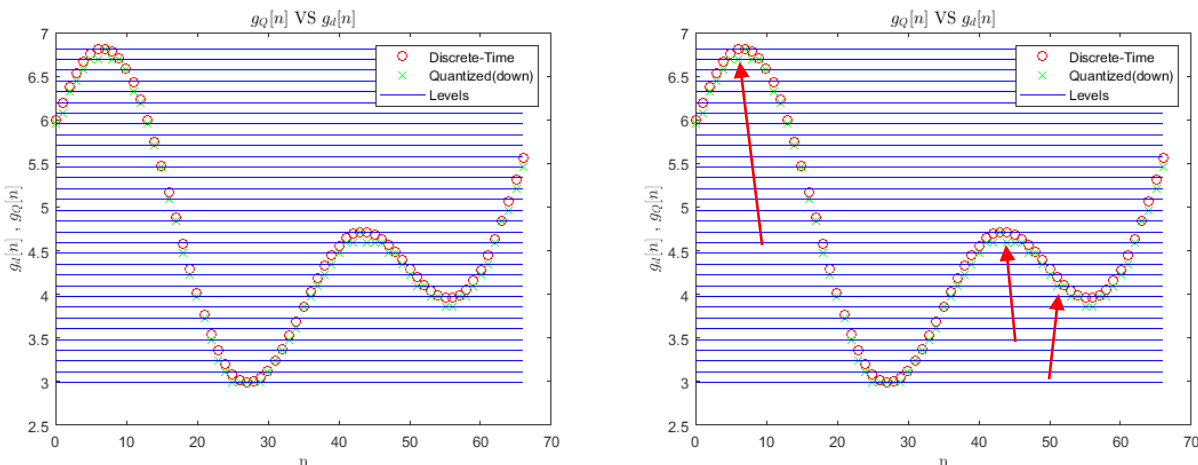
الگوریتم: با توجه به مطالب گفته شده در حقیقت در حوزه دیجیتال باید از سمپل هایی صرف نظر کنیم که وجود آنها در عملیات درونیابی باعث ایجاد خطا میشود. در حقیقت زمانی که از یک سطح به سطح دیگری پرش داریم، سیگنال پیوسته قطعاً سطوح

کوانتیزاسیون را قطع میکند و میتوانیم بگوییم سمپل مورد نظر به این نقطه تقاطع نزدیک است. پس آنرا به عنوان نقطه مشترک در نظر میگیریم. در این صورت سه حالت پرش را بررسی میکنیم:

۱. صعودی: زمانی که سیگنال از یک سطح به سطح بالاتر بپرد، نزدیک ترین سمپل به محل تقاطع اولین سمپل در سطح پرش است.

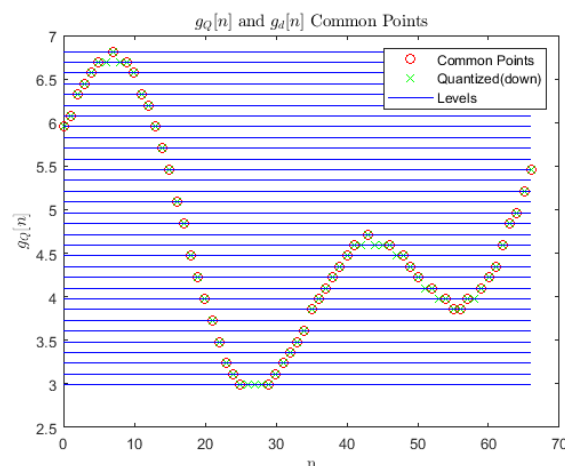
۲. نزولی: زمانی که سیگنال از یک سطح به سطح پایینتر بپرد، نزدیک ترین سمپل به محل تقاطع آخرین سمپل در سطح پرش است.

۲. اکسترمم ها: زمانی که سیگنال از یک سطح به سطح بالاتر بپرد، و سپس بعد از چند سمپل (با حتی بالافاصله) و بدون صعود مجدد، به سطح پایین تر بپرد، اولین و آخرین سمپل هم سطح را در نظر میگیریم. تصویر ۳-۱۰ سمپل هایی که قرار است از آن صرف نظر کنیم را نشان میدهد.



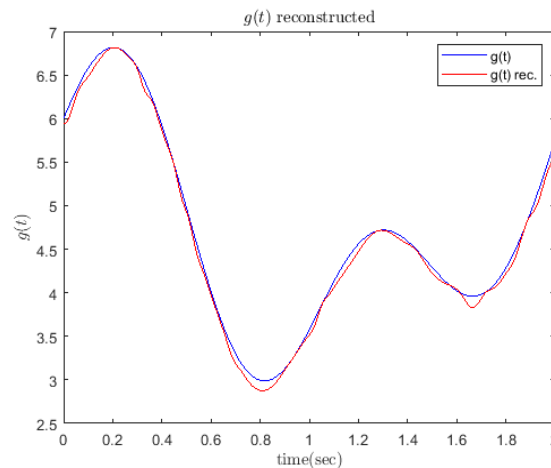
تصویر ۳-۱۰ - سیگنال کوانتیزه شده (تصویر به سطح پایینی) و نقاطی که قرار است از آنها صرف نظر کنیم.

با استفاده از الگوریتم بالا برنامه ای پیاده سازی میکنیم تا نقاط مشترک را ذخیره کند. این نقاط در کنار نقاط کوانتیزه شده در تصویر ۳-۱۱ نشان داده شده است.



تصویر ۳-۱۱ - نقاط مشترک برای درونیابی

در نهایت با استفاده از دستور `spline()` سیگنال پیوسته را با استفاده از نقاط مشترک درونیابی میکنیم. تصویر ۳-۱۲ سیگنال درونیابی شده را در کنار سیگنال آنالوگ اولیه نشان میدهد.



تصویر ۳-۱۲ - سیگنال درونیابی شده

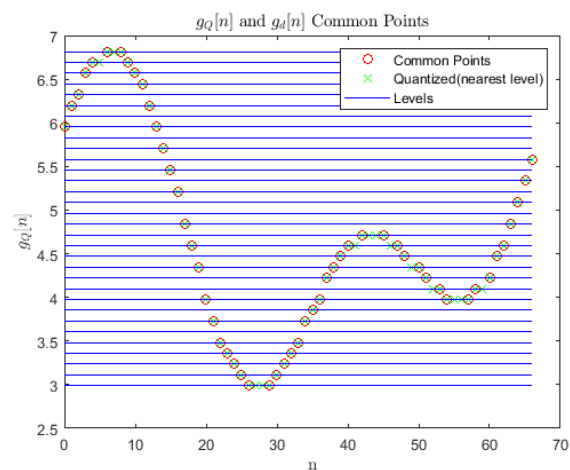
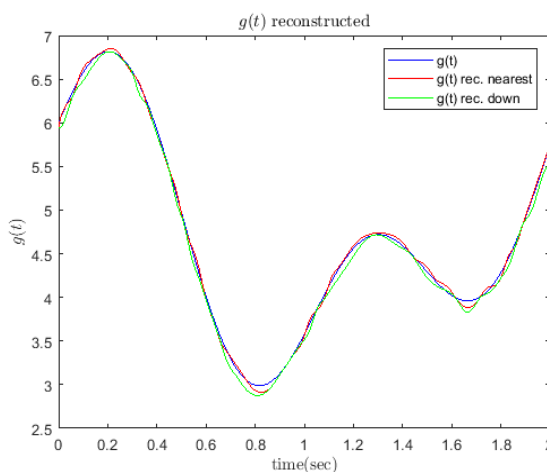
تصویر ۳-۱۳ نیز میانگین مربعات خطای درونیابی را نشان میدهد.

```
immse(g,g_rec)
```

ans = 0.0046

تصویر ۳-۱۳ - خطای سیگنال درونیابی شده

حال اگر از سیگنال دیجیتال کوانتیزه شده اولیه (نزدیکترین سطح به هر سمپل) استفاده میکردیم. نتایج مطابق تصویر ۱۴ میشد و خطای درونیابی نیز کاهش می یافت.



```
immse(g,g_rec2)
```

ans = 0.0012

تصویر ۳-۱۴ - سیگنال درونیابی شده با استفاده از کوانتیزاسیون به نزدیکترین سطح