

با اسمه تعالی

دانشگاه صنعتی شریف

دانشکده مهندسی برق

سیگنال‌ها و سیستم‌ها - گروه دکتر بهروزی - بهار ۱۴۰۱ - ۰۲

تمرین مطلب سری اول

موعد تحویل: ۷ فروردین - ساعت ۲۳:۵۵



تحووه نحویل:

- گزارش تمرین خود را در قالب یک فایل pdf. تحویل دهید. در گزارش لازم است تمامی خروجی‌ها و نتایج نهایی، پرسش‌های متین تمرین و توضیح مختصراً از فرآیند حل مسأله خود در هر قسمت ذکر کنید.
- کد کامل تمرین را در قالب یک فایل m. تحویل دهید. لازم است بخش‌های مختلف تمرین در section‌های مختلف تفکیک شوند و کد تحویلی منظم و دارای کامنت گذاری مناسب باشد. بدیهی است آپلود کردن کدی که به درستی اجرا نشود، به منزله‌ی فاقد اعتبار بودن نتایج گزارش شده نیز می‌باشد.
- توابعی را که (در صورت لزوم) نوشته‌اید، حتماً در انتهای کد در یک section جدا تحت عنوان functions ضمیمه کنید و از ایجاد فایل جدای m. برای آن‌ها خودداری کنید.
- مجموعه‌ی تمامی فایل‌ها (گزارش، کد اصلی و خروجی‌های دیگر در صورت لزوم) را در قالب یک فایل zip/.rar ذخیره کرده و از طریق سامانه CW تحویل دهید.
- نام گذاری فایل‌های تحویلی را به صورت CHW01_StudentNumber.pdf .m .zip .rar انجام دهید.
- ابهام یا اشکالات خود را می‌توانید با طراحان تمارین از طریق آی‌دی‌های @MMohseniZ و @ilia_ad7 مطرح نمایید.

معیار نمره‌دهی:

- ساختار مرتب و حرفة‌ای گزارش
- استفاده از توابع و الگوریتم‌های مناسب
- پاسخ به سؤالات تئوری و توضیح روش‌های مطلوب سوال
- کد و گزارش خروجی کد برای خواسته‌های مسأله

نکته مهم:

- شرافت انسانی ارزشی به مراتب والاتر از تعلقات دنیوی دارد. درصورت کشف شباهت غیر قابل توجیه بین کدها و گزارش‌های آپلود شده، نمره نهایی تمرین مذکور برای همه افرادی که مشارکت داشته‌اند قرینه نمره‌ای که کسب کرده‌اند ثبت خواهد شد.

۱ رسم نمودار سیگنال‌ها

در این سوال قصد داریم با تعریف سیگنال‌ها و ترسیم آن‌ها در مطلب آشنا شویم:

۱. نمودار سیگنال‌های گسسته—زمان زیر را به کمک دستور `stem` مطلب رسم کنید به گونه‌ای که:

- تمامی سیگنال‌ها در یک `figure` باشند.

پلات‌ها دارای `title` بوده و محورها دارای نامگذاری مناسب باشند، برای زیبایی بیشتر میتوانید `interpreter` را روی `latex` قرار دهید و از آن برای حروف‌چینی در نامگذاری‌ها استفاده کنید.

۲. سیگنال‌های $x_5[n]$ در دو ردیف اول `figure` با ابعاد مساوی رسم شوند و سیگنال $x_1[n], x_2[n], x_3[n], x_4[n]$ به تنها‌یی در ردیف سوم رسم شود.

برای نمودارها از رنگهای گوناگون استفاده کنید.

۳. تمامی سیگنال‌ها را در بازه $10 \leq n \leq 10$ رسم کنید.

$$x_1[n] = 4\delta[n+5] + 2\delta[n] - \delta[n-3]$$

$$x_2[n] = (-0.8)^n u[n]$$

$$x_3[n] = u[n-5] - u[n+5]$$

$$x_4[n] = \cos(0.5\pi n) + 3 \sin(\pi n)$$

$$x_5[n] = \sum_{i=-\infty}^n (x_1 + x_2)[i]$$

آیا $x_5[n]$ را بصورت ساده‌تری میتوان بیان کرد؟

۴. نمودار سیگنال‌های پیوسته—زمان زیر را به کمک مطلب رسم کنید به گونه‌ای که:

- تمامی سیگنال‌ها در یک `figure` باشند.

پلات‌ها دارای `title` بوده و محورها دارای نامگذاری مناسب باشند، برای زیبایی بیشتر میتوانید `interpreter` را روی `latex` قرار دهید و از آن برای حروف‌چینی در نامگذاری‌ها استفاده کنید.

۵. برای رسم سیگنال‌های $x_1(t), x_2(t), x_3(t)$ سیگنال‌ها را با فرکانس مناسب نمونه‌برداری کنید، برای اینکار میتوانید از دستور `linspace` و یا $t_{start} : t_{end} : \frac{1}{F_s}$ که در آن F_s فرکانس نمونه‌برداری است استفاده کنید.(با فرکانس نمونه‌برداری در مبحث `sampling` آشنا خواهید شد).

۶. برای رسم سیگنال‌های $x_4(t), x_5(t)$ آنها را بصورت سمبلیک تعریف کرده و سپس برای رسم از دستور `fplot` استفاده کنید. این دو سیگنال را با استفاده از دستور `hold on` در یک `subplot` و با رنگهای متفاوت رسم کنید، سپس با استفاده از `legend` از یکدیگر تمیز دهید.

- تمامی سیگنال‌ها را در بازه $-5 \leq t \leq 5$ رسم کنید.

$$x_1(t) = (1 - e^{-t})u(t)$$

$$x_2(t) = 3r(t+3) + u(t) - u(t-2)$$

$$x_3(t) = sinc(t)(u(t+3) - u(t-3))$$

$$x_4(t) = 0.4 \cos(\pi t) + 1$$

$$x_5(t) = x_4(t) \cos(30\pi t)$$

سیگنال $x_5(t)$ در حقیقت مدولاسیون AM سیگنال $\cos(\pi t)$ است، در آینده با انواع مدولاسیون آنالوگ در درس سیستم‌های مخابراتی آشنا خواهید شد که مدولاسیون AM یکی از مهم‌ترین آنها می‌باشد. برای اطلاعات بیشتر درمورد این مدولاسیون میتوانید از این لینک استفاده کنید.

۲ کانولوشن

در درس با کانولوشن بین دو بردار گسسته یک بعدی و خواص آن آشنا شده‌اید. در این سوال قصد داریم به محاسبه کانولوشن سیگنال‌های گسسته یک بعدی و دو بعدی پردازیم.

۱.۲ کانولوشن یک بعدی

در این بخش به کانولوشن یک بعدی می‌پردازیم و تابعی که خودمان برای کانولوشن یک بعدی نوشته ایم را با تابع خود مطلب مقایسه می‌کنیم:

۱. ابتدا یک تابع برای محاسبه کانولوشن دو بردار با نام `myConv` طراحی خود سعی کنید تا حد ممکن از حلقه‌های تو در تو استفاده نکنید و با استفاده از عملیات‌های ماتریسی مطلب طراحی خود را انجام دهید. دقت کنید مجاز به استفاده از توابع آماده مطلب مانند `conv` نیستید.

۲. حاصل کانولوشن دو به دو سیگنال‌های گسسته زیر را با استفاده از تابعی که نوشته‌ید بدست آورید و با تنظیم صحیح محور زمان، حاصل را رسم کنید:

$$\begin{aligned}x_1[n] &= u[n+7] - u[n-8] + 0.5\delta[n] - 4\delta[n-3] + 2\delta[n+4] \\x_2[n] &= \left(\frac{3}{5}\right)^n(u[n+12] - u[n-12]) \\x_3[n] &= n \sin(0.5n)(u[n+15] - u[n-15])\end{aligned}$$

۳. محاسبات قسمت قبل را این بار با استفاده از تابع آماده `conv` مطلب محاسبه کرده و بصورت مناسب رسم کنید و صحت عملکرد تابعی که خودتان نوشته‌اید را بررسی کنید. سپس با استفاده از دستورات `tic` و `toc` مطلب سرعت اجرای تابع خودتان و تابع آماده مطلب را برای دو بردار تصادفی با طول به اندازه کافی بزرگ با یکدیگر مقایسه کنید. (کافیست اختلاف زمان اجرا برای دو تابع به قدر کافی واضح باشد)

۲.۲ کانولوشن دو بعدی

در قسمت‌های قبل با کانولوشن یک بعدی آشنا شدید. در این قسمت قصد داریم کانولوشن بین سیگنال‌های دو بعدی (ماتریس‌ها!) را بررسی کنیم که در پردازش تصویر کاربرد زیادی دارند را بررسی کنیم. کانولوشن دو بعدی بین دو ماتریس با استفاده از رابطه زیر محاسبه می‌شود:

$$C = A * B \iff C(i, j) = \sum_p \sum_q A(p, q)B(j - p + 1, i - q + 1)$$

دو ماتریس A و B را بصورت زیر درنظر بگیرید:

$$A = \begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \\ 21 & 22 & 23 & 24 & 25 \end{bmatrix}$$

با استفاده از دستور `conv2` مطلب حاصل کانولوشن بین این دو ماتریس را حساب کرده و آن را `ConvFull` بنامید. همین عملیات را ایبار با استفاده از آپشن '`'same'`' تابع `conv2` انجام دهید و حاصل را `ConvSame` بنامید. با مقایسه ابعاد و محتوای دو ماتریس `ConvSame` و `ConvFull` عملکرد آپشن '`'same'`' را توضیح دهید.

۳ مقدمه‌ای بر پردازش تصویر

در این تمرین به بررسی مقدماتی برخی دستورات ساده برای پردازش تصویر می‌پردازیم، در ادامه اندکی با پردازش تصویر در حوزه مکان آشنا خواهیم شد.

۱.۳ برخی دستورات ساده برای پردازش تصاویر

۱. در ابتدا با استفاده از دستور `imread` یک تصویر به دلخواه خودتان را در مطلب لود کنید. این تصویر یک تانسور ۳ بعدی است که ابعاد آن به اندازه ابعاد تصویر در ۳ بوده که نشان دهنده شدت ۳ رنگ آبی، قرمز و سبز تصویر است. همانطور که مشاهده می‌کنید جنس درایه‌های این تانسور ۳ بعدی از نوع `uint8` می‌باشد، با استفاده از این `datatype` برای ذخیره سازی تصویر، هر پیکسل چه تعداد رنگ ممکن را می‌تواند نمایش دهد؟ با خواندن `document` این دستور هر ۳ رنگ را جدا کرده و در یک `subplot` به همراه تصویر اصلی نمایش دهید.

۲. حال این سه ماتریس را با هم جمع کرده و ماتریس میانگین را توسط `imshow` نشان دهید همچنین نتیجه خروجی را با نتیجه حاصل از دستور `rgb2gray` مقایسه کنید و نتایج را ذکر کنید.

۳. درایه‌های تصویر حاصل از قسمت قبل را با استفاده از دستور `im2double` از بازه ۰ تا ۱ مپ کرده و نمایش دهید.(در قسمت های بعدی فایده این کار را مشاهده خواهید کرد) آیا تفاوتی میان این تصویر و تصویر قبل مشاهده می‌کنید؟ تابع `imshow` چگونه این دو تصویر با درایه‌های متفاوت را یکسان نشان میدهد؟

۲.۳ یافتن لبه‌های تصویر (Edge Detection)

در بخش قبل با برخی از دستورات ساده برای آغاز پردازش تصویر آشنا شدیم، در این بخش به بررسی الگوریتم سوبل برای یافتن لبه‌های یک تصویر می‌پردازیم.

الگوریتم Sobel-Feldman که در سال ۱۹۶۸ توسط این دو دانشمند معرفی شد یک روش کلاسیک برای تشخیص لبه‌های یک تصویر است. برای مطالعه بیشتر درباره این الگوریتم میتوانید به این [لينک](#) مراجعه کنید

۱. در سوال قبل با کانولوشن دو سیگنال دو بعدی(ماتریس!) آشنا شدیم. در یک مدل از الگوریتم لبه‌یابی از کانولوشن دو ماتریس زیر با تصویر برای لبه‌یابی استفاده می‌شود:

$$EDK_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad EDK_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

ماتریس EK_x برای یافتن لبه‌های افقی تصویر و ماتریس EK_y برای یافتن لبه‌های عمودی ماتریس می‌باشد. یعنی اگر تصویر grayscale شده با نام `IMG` داشته باشیم:

$$IMG_{edges,x} = EDK_x * IMG \quad IMG_{edges,y} = EDK_y * IMG$$

که در آن (*) عملگر کانولوشن می‌باشد. در ادامه برای نمایش لبه‌های تصویر با استفاده از لبه‌های افقی و عمودی از رابطه زیر استفاده می‌کنیم:

$$IMG_{edges} = \sqrt{IMG_{edges,x}^2 + IMG_{edges,y}^2}$$

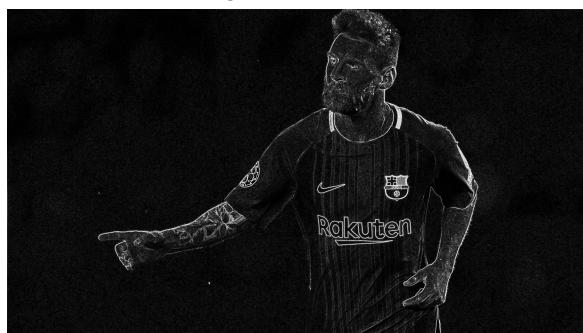
توضیح دهید چگونه این دو ماتریس میتوانند لبه‌های افقی و عمودی تصویر را تمیز دهند؟



(ب) لبه‌های افقی



(الف) تصویر اصلی



(د) لبه‌های کلی

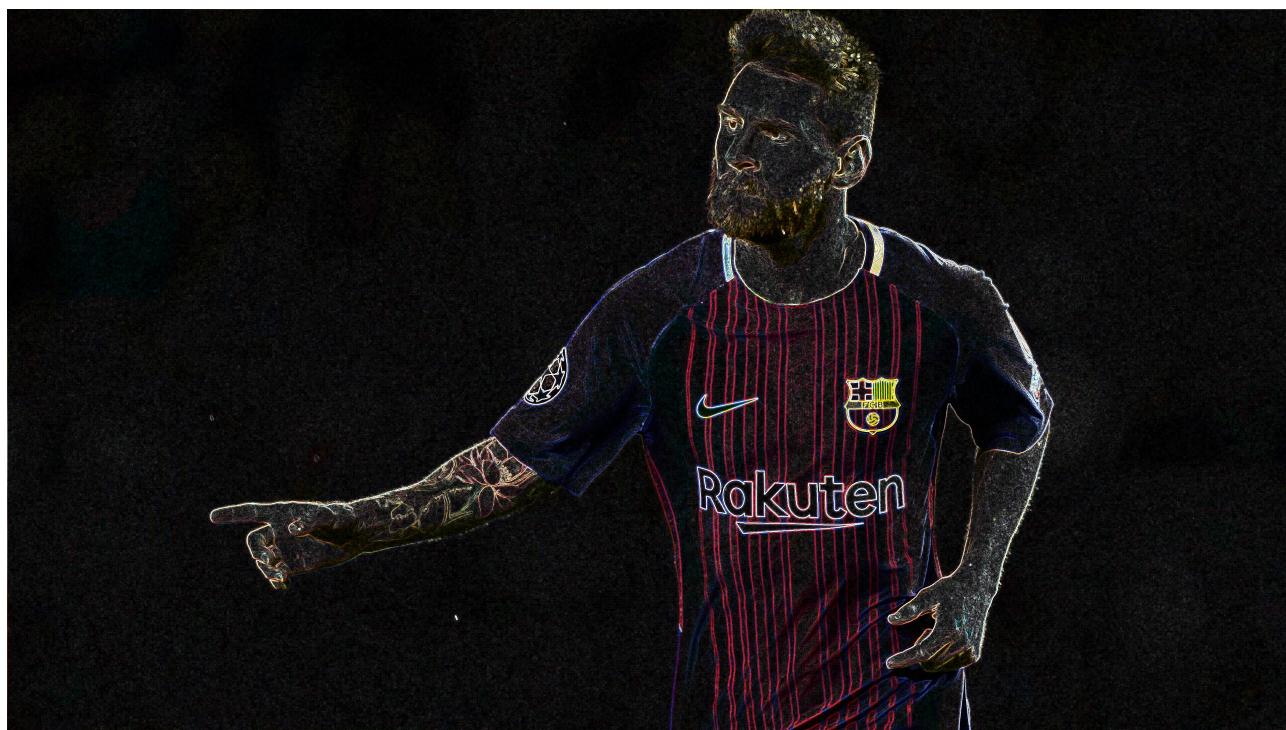


(ج) لبه‌های عمودی

شکل ۱: نمونه‌ای از تصاویر نهایی و اولیه

تصویر دلخواهی (که طبیعتاً لبه قابل تشخیص توسط چشم داشته باشد) را انتخاب کرده و پس از تبدیل آن به grayscale، عمودی و لبه‌های کلی تصویر را به همراه تصویر اصلی در گزارش بیاورید.

۲. مراحلی را که در قسمت قبل انجام دادید را اینبار بدون اینکه تصویر را grayscale کنید انجام دهید و تصویر حاصل را در گزارش بیاورید. (در حقیقت باید مراحل قسمت قبل را روی هر کانال تصویر (قرمز، سبز و آبی) انجام دهید)



شکل ۲: لبه‌های تصویر بصورت رنگی

۳.۳ آشنایی با کرنل‌ها

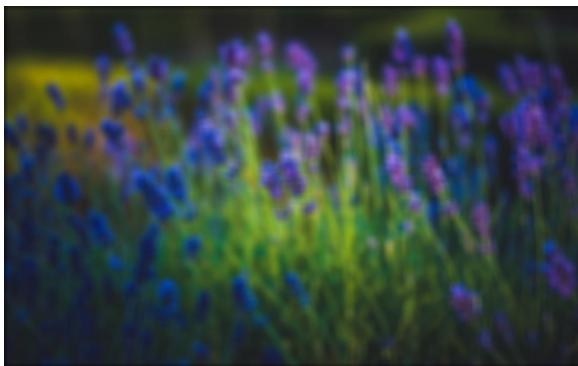
دو ماتریس EK_x, EK_y که در قسمت قبل با آنها آشنا شدیم را کرنل (Kernel) مینامند. در پردازش تصویر کرنل‌های گوناگونی وجود دارند که با کانوالو کردن آنها در تصویر، تغییرات دلخواه ما در آن ایجاد می‌شود. در این بخش با ۲ نوع دیگر از کرنل‌ها با کاربردهای مختلف آشنا می‌شویم.

Image Blurring ۱.۳.۳

همانطور که احتمالاً قبل هنگام کار با نرم‌افزارهای ویرایش تصویر متوجه شده‌اید یکی از کارهای رایجی که با تصویر می‌توان انجام داد مهآلود کردن تصویر یا به اصطلاح Blurring است. برای اینکار می‌توان از کرنل زیر استفاده کرد که به Box Blur معروف است:

$$BBK = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

توضیح دهید چگونه کانوالو کردن این کرنل با تصویر اصلی آنرا مهآلود می‌کند؟ با استفاده از این کرنل، تصویری که انتخاب کرده بودید را بصورت رنگی (در هر سه کanal تصویر) Blurry کرده و همراه تصویر اصلی در گزارش درج کنید. ممکن است رزولوشن تصویرتان بالا بوده و متوجه مهآلود شدن آن نشوید، چگونه می‌توان با تغییر کرنل میزان مهآلودگی تصویر را افزایش داد؟ اینکار را انجام دهید تا مهآلود شدن تصویر مشهود شود.



(ب) تصویر مهآلود



(الف) تصویر اصلی

شکل ۳: مهآلود کردن تصویر

Image Sharpening ۲.۳.۳

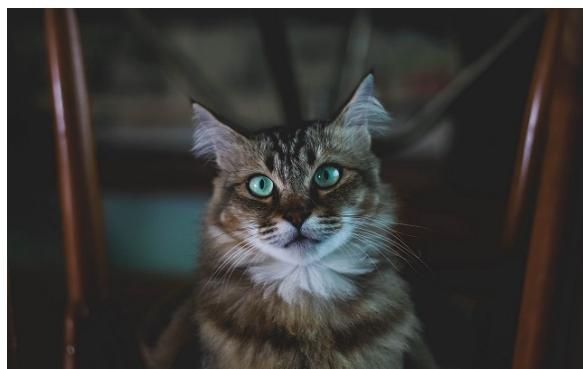
یکی دیگر از کارهایی که در ویرایش تصویر انجام می‌شود Sharp کردن تصویر است که موجب می‌شود جزئیات تصویر بیشتر دیده شوند. برای این منظور از کرنل زیر استفاده می‌شود:

$$ISK = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

با استفاده از این کرنل تصویر خود را شارپ کرده و همراه با تصویر اصلی در گزارش درج کنید.



(ب) تصویر شارپ شده



(الف) تصویر اصلی

شكل ۴: شارپ کردن تصویر

۴ مقدمه‌ای بر پردازش صوت

در این تمرین قصد داریم با مفاهیم ابتدایی مربوط با پردازش سیگنال صوت آشنا شویم.

۱. فایل Music.wav که در پوشه پروژه ضمیمه شده است را با استفاده از دستور `audioread` در محیط متلب خوانده و آن را پلات کنید. (با توجه به اینکه با مفهوم نمونه برداری هنوز آشنایی ندارید نیازی به تنظیم محور افقی زمان نیست) اگر دقت کنید دو سیگنال گسته بصورت همزمان پلات شده‌اند که یکی از آنها مربوط به کanal راست و دیگری مربوط به کanal چپ صدا می‌باشد. به این نوع فایل‌های صوتی که دارای چند کanal می‌باشد فایل صوتی استریو گفته می‌شود که با استفاده از چند میکروفون ضبط شده‌اند و تجربه واقعی تری را به شنونده منتقل می‌کنند. در این سوال قصد نداریم هر دو کanal را جداگانه مورد بررسی قرار دهیم. به این منظور میانگین کanal چپ و راست را به عنوان سیگنال‌مان در نظر می‌گیریم. در هر دو حالت با استفاده از دستور `soundsc` با فرکانس $F_s = 48KHz$ به سیگنال صوت گوش کرده و تفاوت احساس خود را شرح دهید.

۲. برای افزودن echo به سیگنال فوق میتوان سیگنال اصلی را با شیفت یافته زمانی اش جمع کرد، سیستمی که این کار را برای ما انجام میدهد به شکل زیر توصیف می‌شود:

$$y[n] = x[n] + \alpha x[n - n_d]$$

که در آن α ضریب تضعیف صوت اکو شده و $n_d = 0.2F_s$ میزان تاخیر صوت اکو شده است. در این بخش نظر گرفته می‌شود که باعث ایجاد تاخیر ۰.۲ ثانیه‌ای می‌شود. خروجی سیستم را به ازای $\alpha = 0.8$ بدست آوردن خروجی میتوانید از دستور `filter` متلب استفاده کنید. سیگنال خروجی را رسم و به آن گوش دهید.

۳. در این بخش قصد داریم مولفه اکو سیگنال را حذف کرده و سیگنال اصلی را بازیابی کنیم، برای این منظور پاسخ ضریب سیستم معکوس علی‌را باید و با استفاده از دستور `filter` پاسخ سیستم به خروجی قسمت ۲ (صوت دارای اکو) را بدست آورید. سیگنال حاصل را رسم کرده و به آن گوش دهید. آیا سیگنال اصلی بازیابی شده است؟

۴. دو قسمت قبلی را برای سیستم زیر انجام داده و نمودارهای سیگنال اصلی، سیگنال اکو و سیگنال بازیابی شده را کنار یکدیگر رسم کنید:

$$y[n] = x[n] + \alpha x[n - n_d] + \alpha^2 x[n - 2n_d] + \alpha^3 x[n - 3n_d] \quad (n_d = 0.1F_s, \alpha = 0.8)$$

۵. در این بخش به اثر نویز بر روی سیگنال می‌پردازیم. دو نویز با توزیع‌های:

$$n_1 \sim \mathcal{N}(0, 0.01)$$

$$n_2 \sim Uniform(-0.1, 0.1)$$

را جداگانه به سیگنال اصلی اضافه کنید. سیگنال‌های حاصل را رسم کرده و به آنها گوش دهید. برای تولید نویزها با توزیع گفته شده از دستورهای `rand` و `randn` استفاده کنید.

۶. بدون استفاده از دستور `chirp` یک سیگنال سینوسی با دامنه ۰.۱ که فرکانس آن بصورت خطی با زمان از $1KHz$ تا $2KHz$ افزایش می‌باید تولید کنید. این سیگنال را به سیگنال صوت اصلی اضافه کرده و به آن گوش دهید. سیگنال خروجی را با استفاده از دستور `audiowrite` با نام MusicPlusSin.wav ذخیره کرده و به فایل‌های تحویلی تان ضمیمه کنید.