

**دانشگاه تهران**

**دانشکده‌ علوم و فنون نوین**

**پردازش تصویر دیجیتال**

**تمرین شماره شش**

|  |  |
| --- | --- |
| فاطمه چیت ساز | نام و نام خانوادگی |
| 830402092 | شماره‌ دانشجویی |
| 5 آذر 1402 | تاریخ ارسال گزارش |

Contents

[سوال 1 – اضافه کردن نویز گوسی 1](#_Toc151926527)

[سوال ۲ – فیلتر fft 4](#_Toc151926528)

# سوال 1 – اضافه کردن نویز گوسی

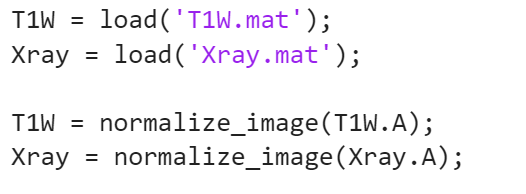
در این سوال ما ابتدا دو تصویر داریم سپس آنها را میخوانیم و میخواهیم نویز گوسی به آنها اضافه کنیم اما این نویز را میخواهیم طوری اضافه کنیم که psnr مربوط به تصویر اعداد پانزده تا سی شود

حال بعد از آنکه تصاویر نویزی خود را ساختیم زمان آن رسیده که فیلتر پایین گذر میانگین را روی تصاویر نویزی اجرا کرده و دوباره psnr را محاسبه کنیم همچنین قابل توجه است که مقادیر اندازه پنجره فیلتر نیز عددی متغیر است

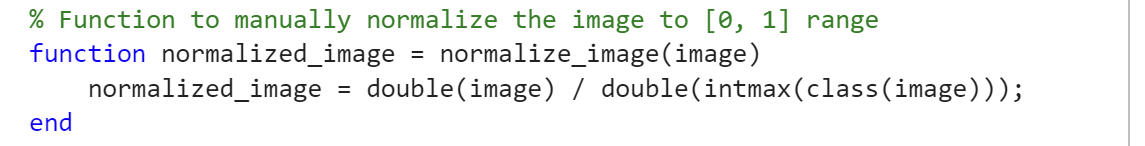
در اخر ما قصد داریم نمودار psnr اولیه به ثانویه را برای فیلتر ها با سایز های مختلف بکشیم

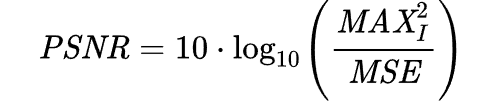
توضیح کد:

خواندن تصاویر و نرمال‌سازی آنها:

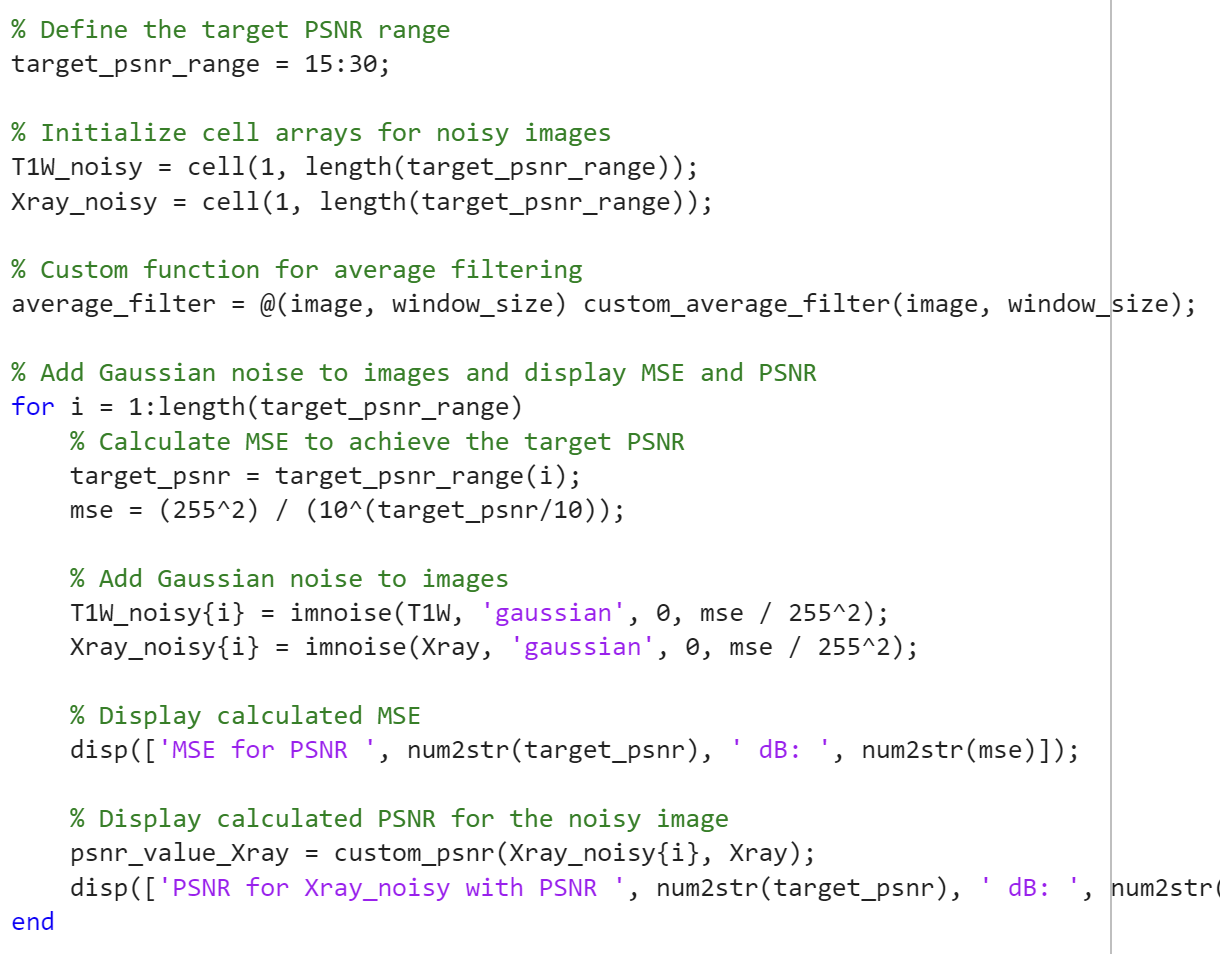


فانشن normalize image نیز توسط ما نوشته شده و اعداد موجود در تصاویر را نرمال کرده و به double هایی بین صفرو یک تبدیل میکند

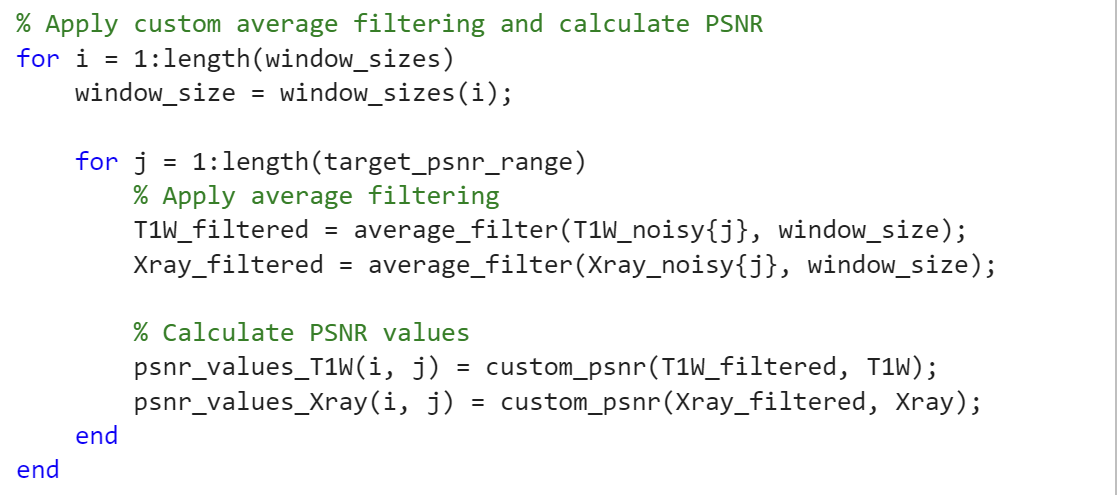


در مرحله بعدی زمان اضافه کردن نویز گوسی رسیده است اما نکته ای که وجود دارد این است که ما باید نویز را طوری اضافه کنیم که به psnr هایی بین پانزده تا سی برسیم برای اینکار به برسی فرمول psnr میپردازیم   


با توجه به اینکه تصویر ما نرمال شده است پس max ما عدد 255 است پس با توجه به مقادیر psnr که میخواهیم یعنی بازه پانزده تا سی مقادیر mse را بدست میاوریم و با توجه به آن نویز گوسی را روی تصاویر اعمال میکنیم



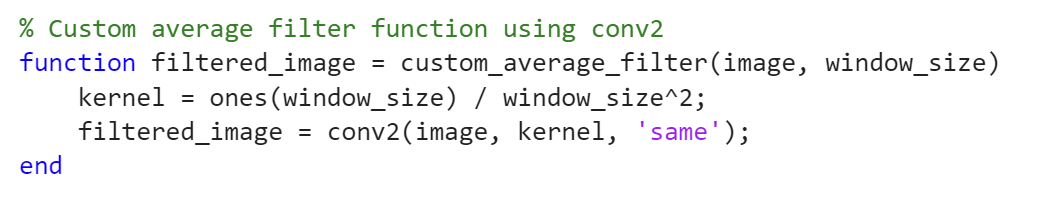
بعد از محاسبه تصاویر نویزی زمان آن رسیده تا فیلتر میانگیر پایین گذر را بر روی تصاویر اعمال کرده و تصاویر دی نویز شده را بدست آوریم



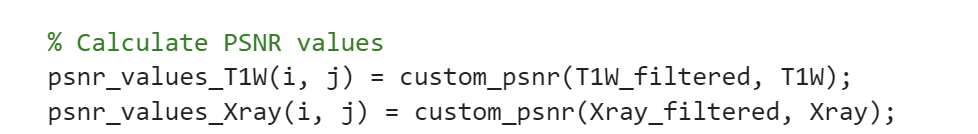
در اینجا average\_filter برای ما عملیات فیلتر پایین گذر میانگین گیر را انجام میده

برای انجام این کار دو راه حل وجود دارد یک آنکه روی تصویر حرکت کنیم و با توجه به اندازه پنجره مان میانگین را حساب کنیم ولی این کار بیسار زمانبر است ( به علت دو for تو در تو)

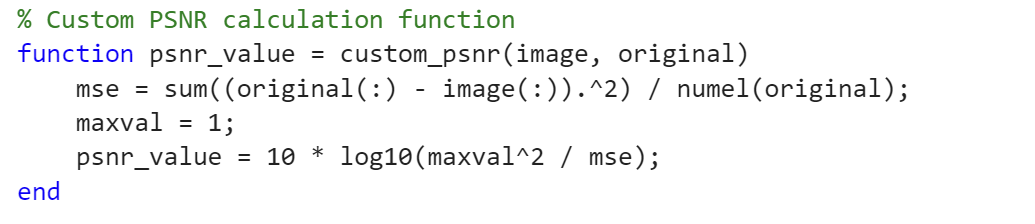
راه حل دوم استفاده از conv2 است



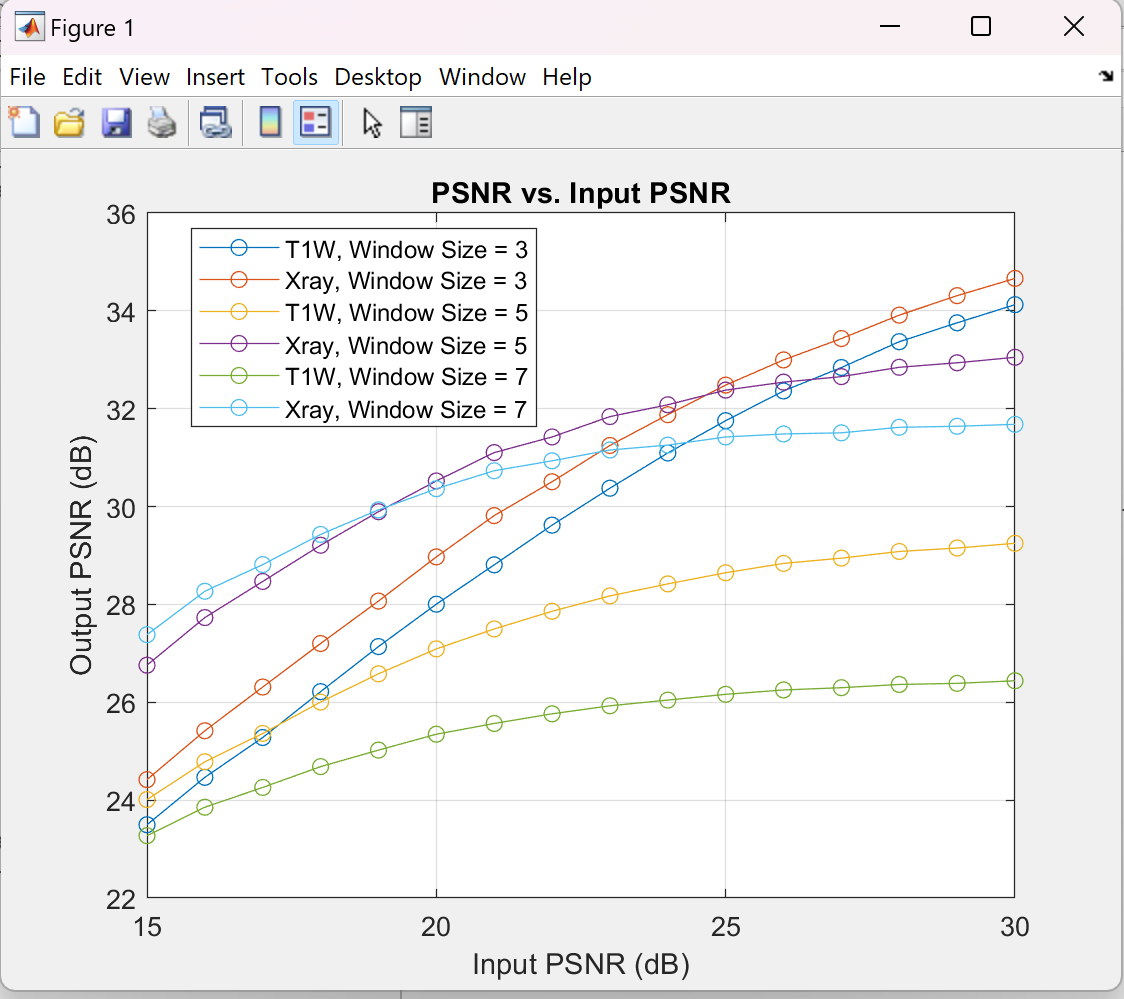
بعد از اجرای فیلتر روی تصاویر میتوان psnr را برای هر کدام محاسبه کرد



همچنین قابل توجه است که psnr از فرمول زیر بدست میاید



حال که psnr اولیه و ثانویه را داریم میتوانیم نمودار خود را رسم کنیم



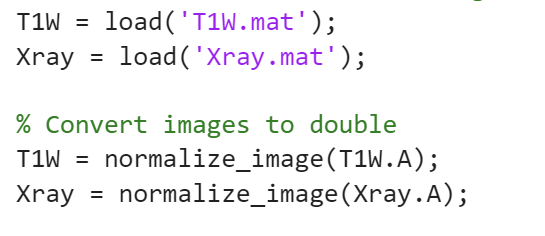
کد این سوال در q1 موجود است

# سوال ۲ – فیلتر fft

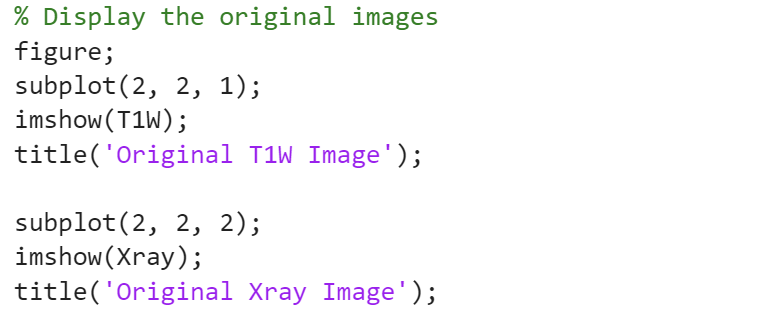
در این سوال ما قصد داریم بر روی تصویری که psnr آن 15 است فیلتر fft دو بعدی را اجرا کنیم و مقادیر درصد فرکانس حذف شده را تغیر دهیم و تاثیر آن روی Psnr بعد از فیلتر را ببینیم

توضیح کد :

در مرحله اول تصویر ها را میخوانیم و نرمال میکنیم (مثل سوال یک)



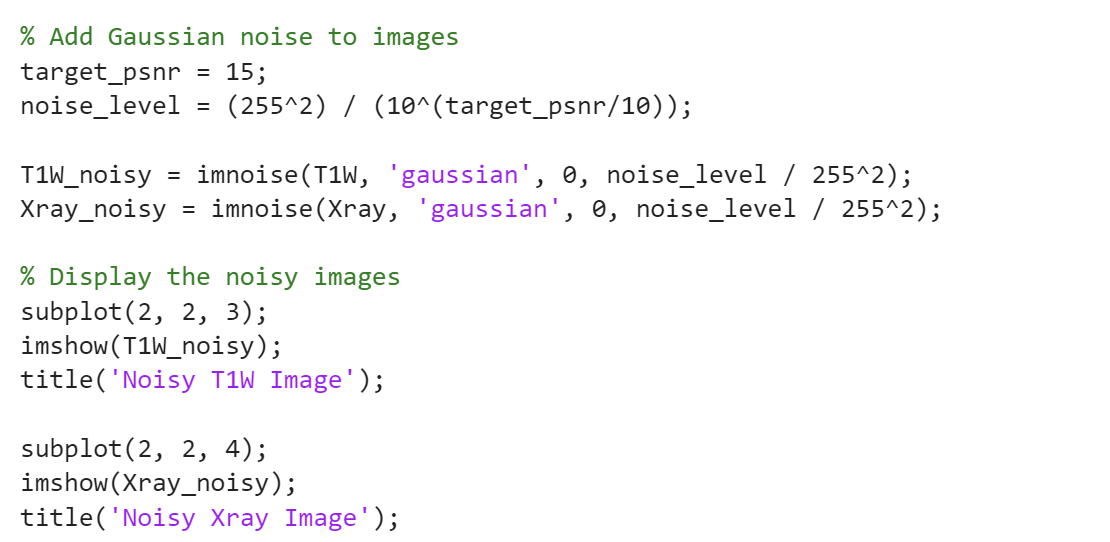
در مرحله دوم آنها را نمایش میدهیم تا دیدی نسبت به آنها داشته باشیم



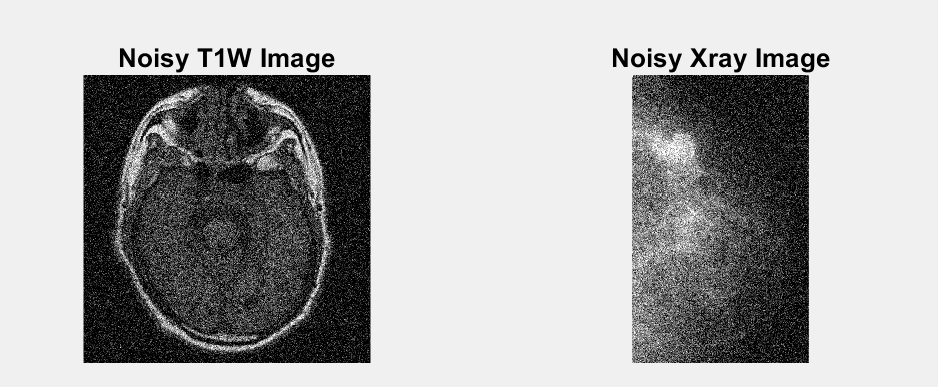
شکل:



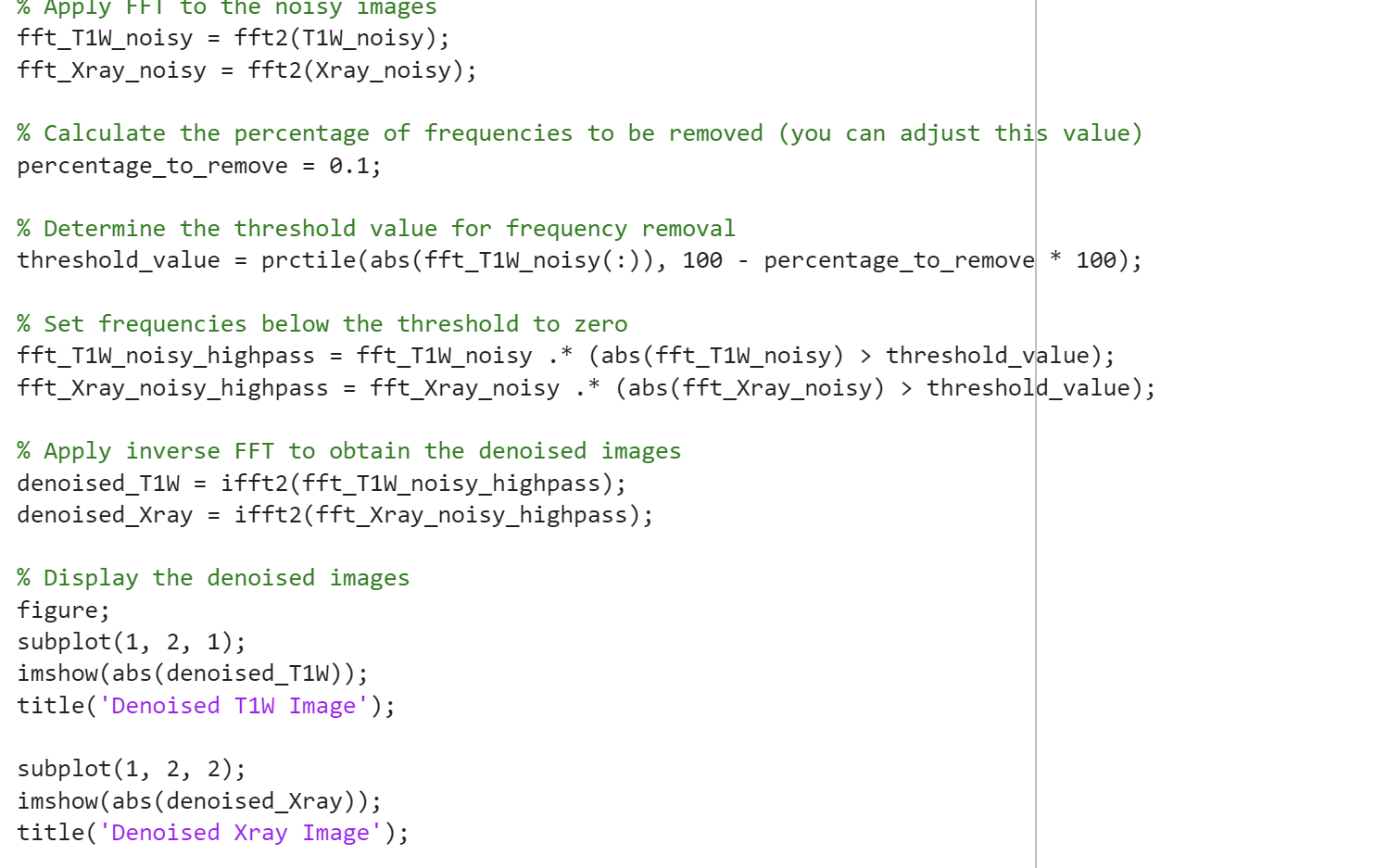
در مرحله بعدی فیلتر گوسین را روی آنها قرار میدهیم به طوری که psnr ما پانزده دسیبل شود برای این کار ابتدا noise\_level یا همان mse خود را بدست آورده و با توجه به آن فیلتر گوسی را روی تصویر اعمال کرده و تصاویر را نمایش میدهیم



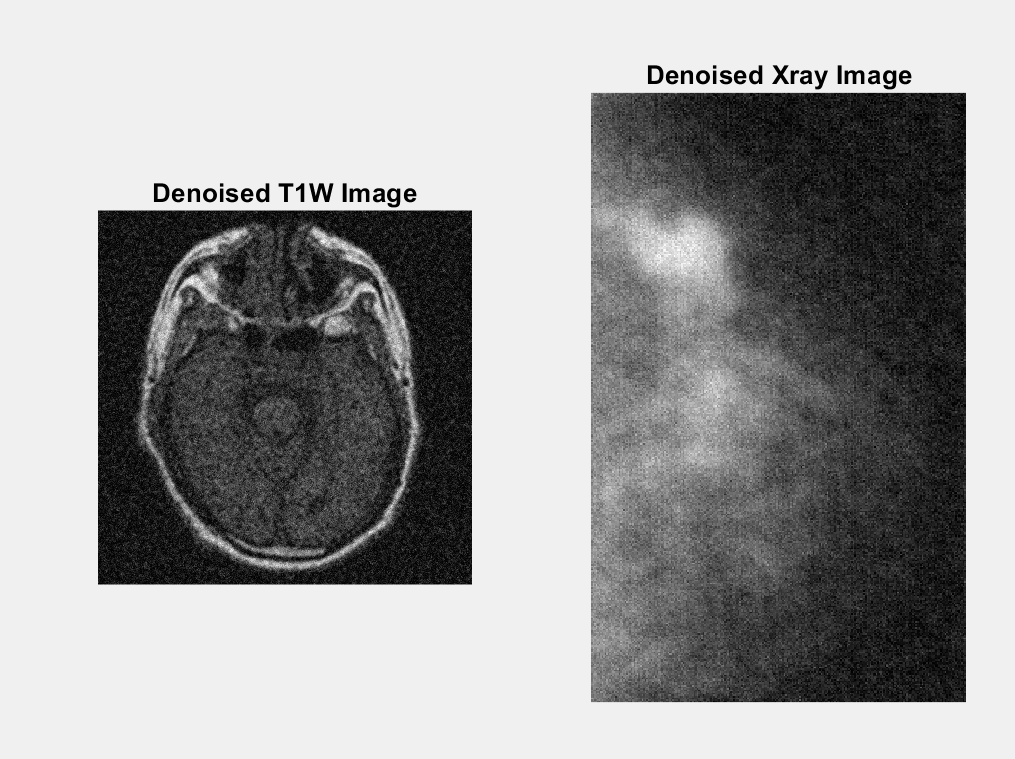
شکل:



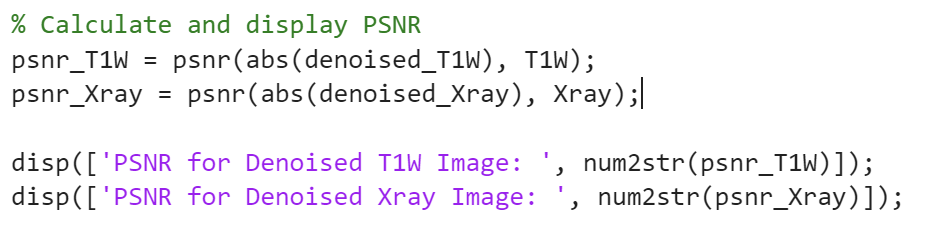
سپس در مرحله بعدی فیلتر fft را روی تصاویر اعمال کرد و عددی را به عنوان درصد حذف فرکانس برای خود در نظر گرفته و با توجه به آن threshold\_value خود را ساخته و تصویر را از آن عبور میدهیم حال ifft را روی تصاویر اعمال کرده تا تصویر دی نویز شده بدست آید و میتوان آن ها را نمایش داد



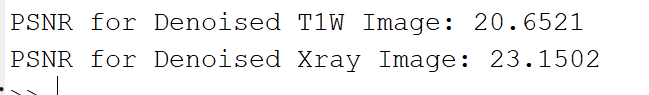
شکل :



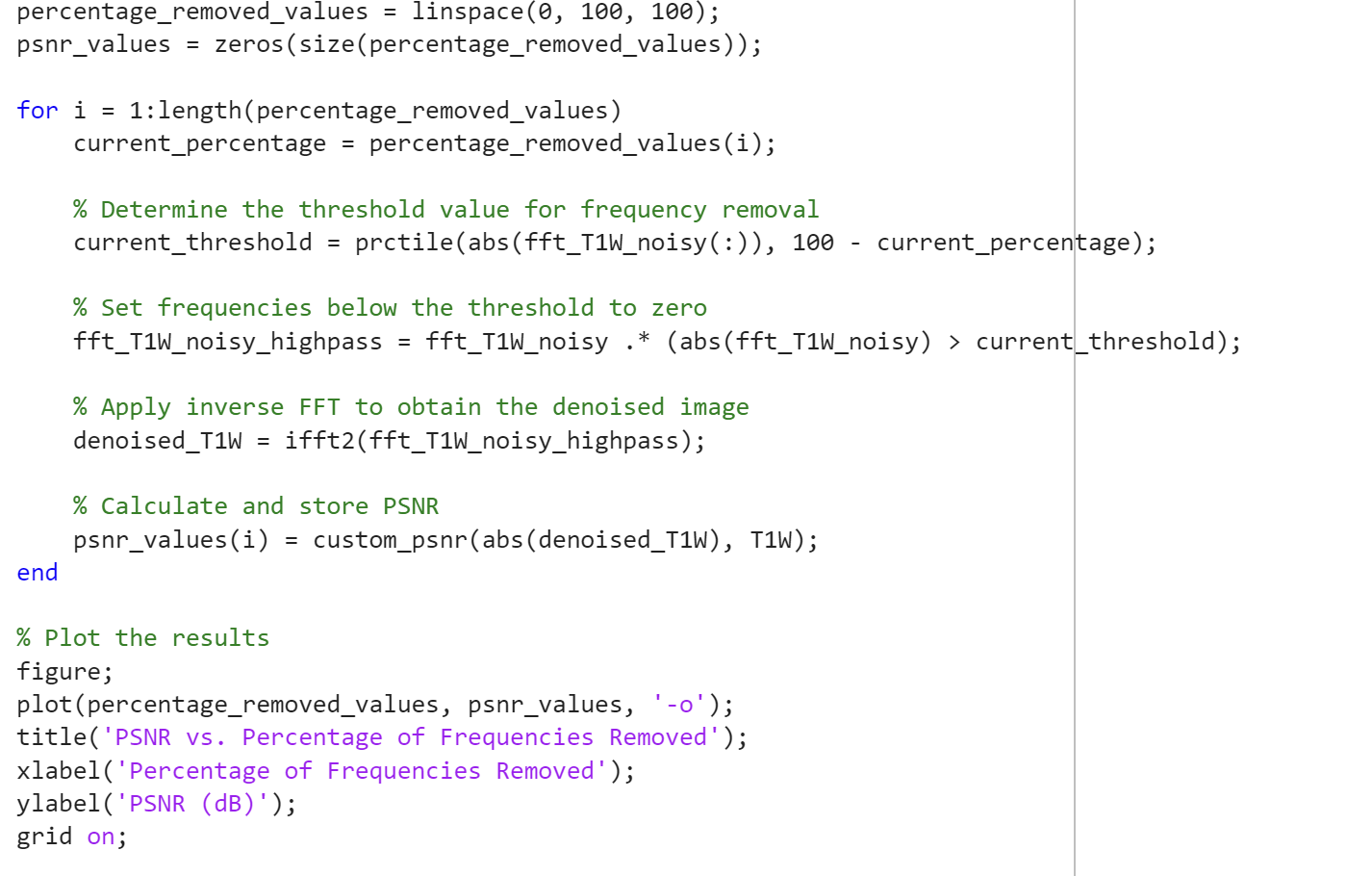
حال میتوانیم psnr را بعد از اعمال فیلتر محاسبه کنیم



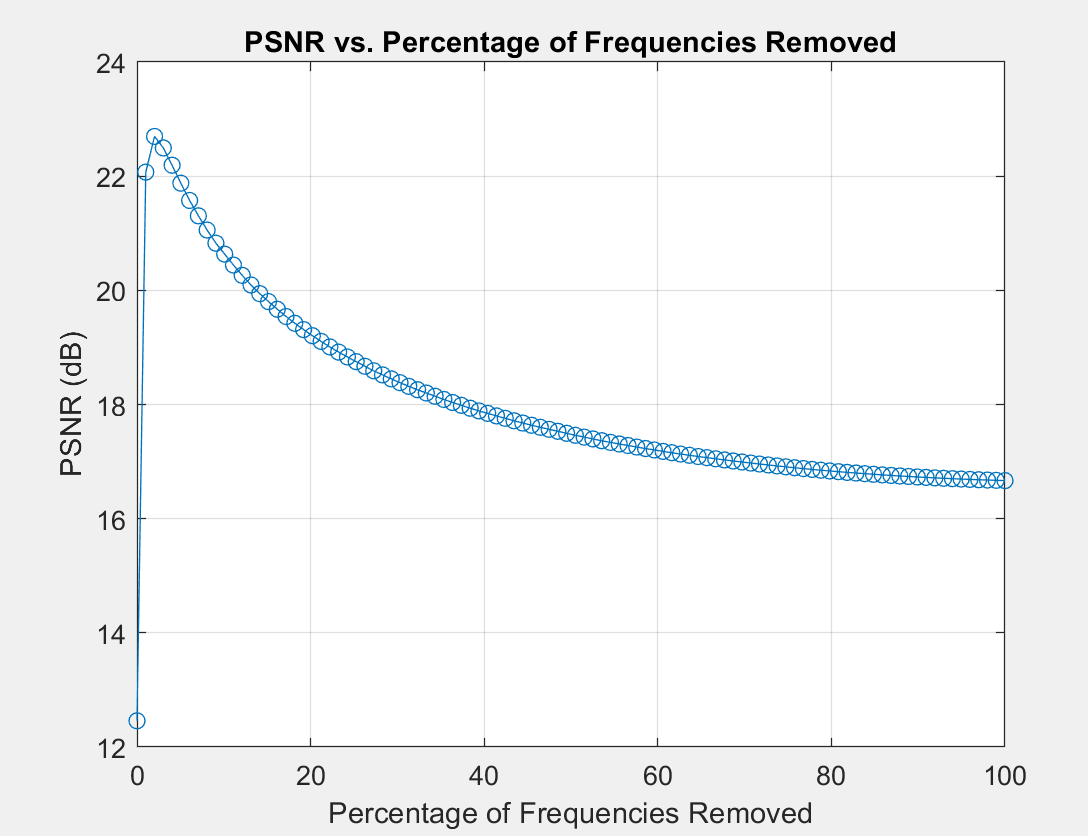
نتیجه:



و مشاهده میشود psnr بعد از اعمال تصویر بیشتر شده است یعنی به عبارتی کیفیت تصویر بیشتر شده است حال این کار را باید برای بازه ای از مقادیر مختلف درصد حذف فرکانس انجام دهیم و psnr را برای هر یک محاسبه کنیم و نموداری بر حسب آن رسم کنیم



در اینجا ما برای درصد حذف صفر تا صد این فرایند را تکرار کرده و نمودار ما به شکل زیر است



همانطور که مشاهده میشود در ابتدا با افزایش درصد psnr نیز افزایش یافته است اما بعد از مدتی psnr با افزایش درصد حذف فرکانس کاهش یافته است

فایل این کد در فایل q2 موجود است