بسمه تعالی

گزارش کار پروژه اول

آیدین کاظمی 810101561

بخش اول

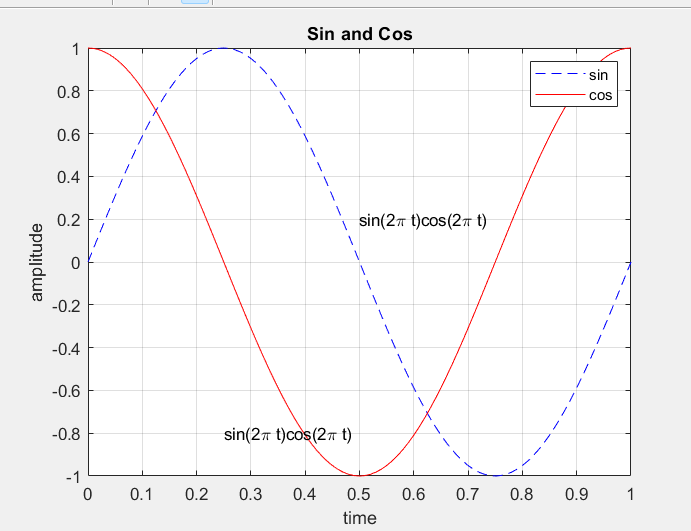
1-1)

در خط اول متغیر مستقل t به شکل یک بردار بین 0 و 1 با گام های 0.01 ساخته شده و سپس در خط های دو و سه توابعsin(2\*pi\*t), cos(2\*pi\*t) به ترتیب ساخته شده و در دو بردار z1 و z2 ذخیره میشوند.

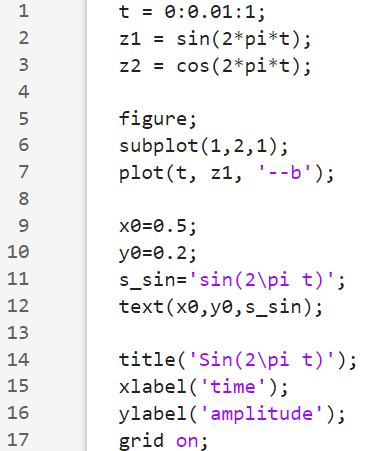
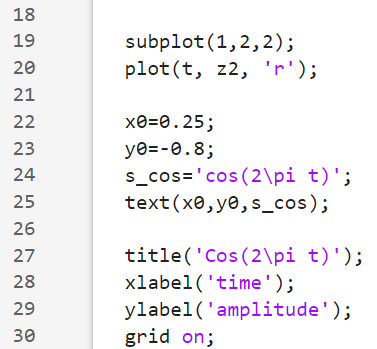
در بخش بعدی دستور figure صفحه رسم را باز کرده و با دستور های plot، دو تابع z1 و z2 بر حسب t رسم میشوند. نقش hold on در خط 7 نگه داشتن نمودار کشیده شده مربوط به z1 در صفحه رسم است و در صورتی که این خط حذف شود، فقط نمودار دوم در صفحه باقی میماند و نمودار اول پاک میشود. دو آرگومان --b و r به ترتیب به معنای خط چین آبی و رنگ قرمز نمودار رسم شده هستند.

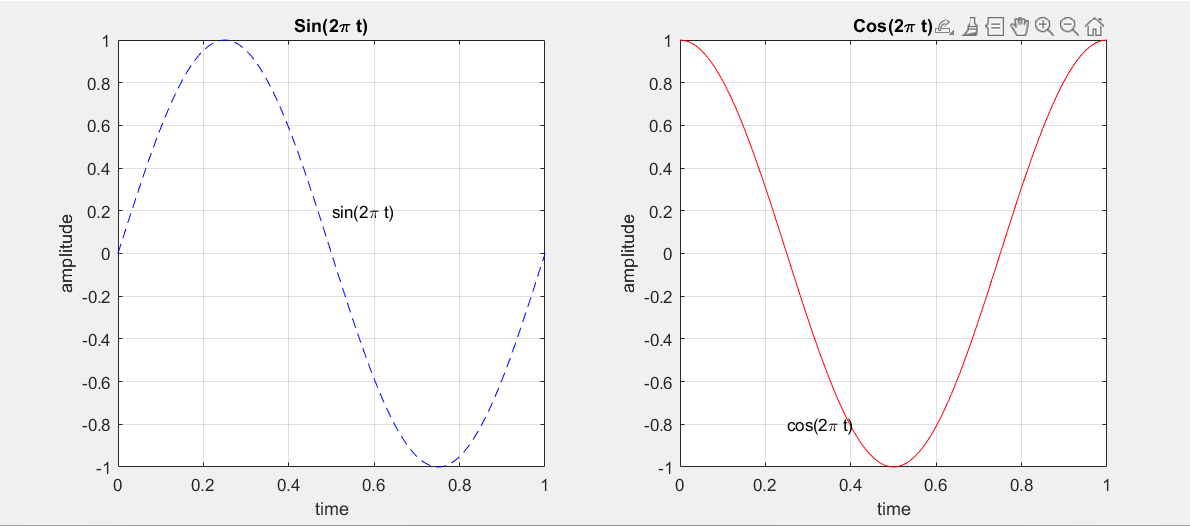
در مرحله بعد دو متغیر x0 و y0 را تعریف میکنیم که دو بردار حاوی به ترتیب طول و عرض های نقاط مد نظر برای قرار دادن متن های متناظر با نمودار های سینوس و کسینوس هستند، که این متن ها نیز در متغیر s تعریف و نگه داری شده و در نهایت با دستور text در صفحه نمودار ها در مکان تعین شده نوشته میشوند.

در بخش پایانی ابتدا title نام نمودار را مشخص کرده و سپس دستور legend نام مربوط به هر نمودار را در گوشه صفحه همراه با شکل آن نمودار رسم میکند. نهایتا xlabel و ylabel محور های متناظر خود را نام گذاری کرده و دستورgrid on صفحه را به بخش های مربعی تقسیم میکند. نمودار نهایی:

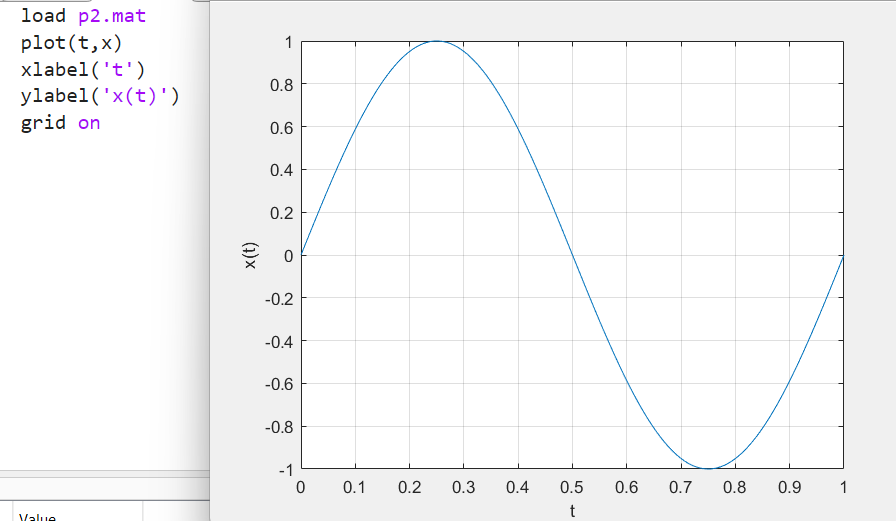


1-2)تابع subplot(m,n,x) به این صورت کار میکند که m تعداد سطر های حاوی عکس ها، n تعداد ستون های عکس و x شماره خانه مد نظر است و از 1 شروع شده و تا m \* n ادامه دارد:

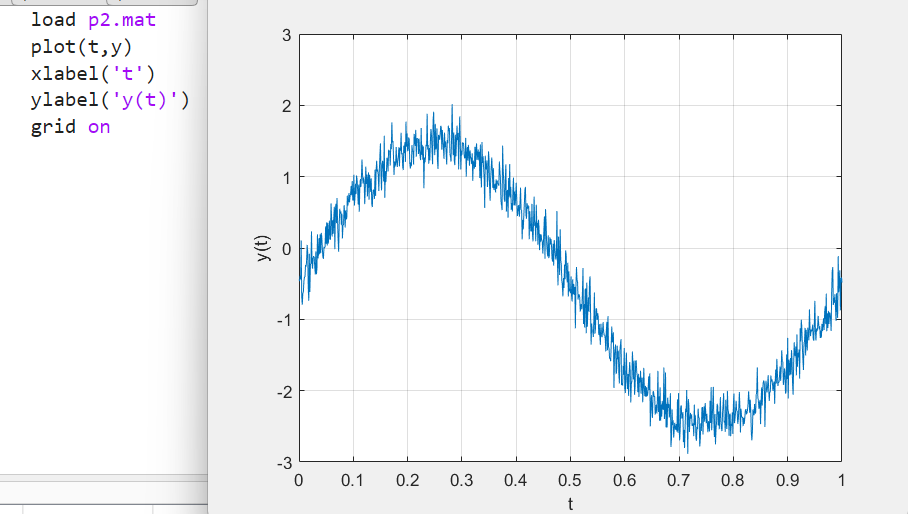


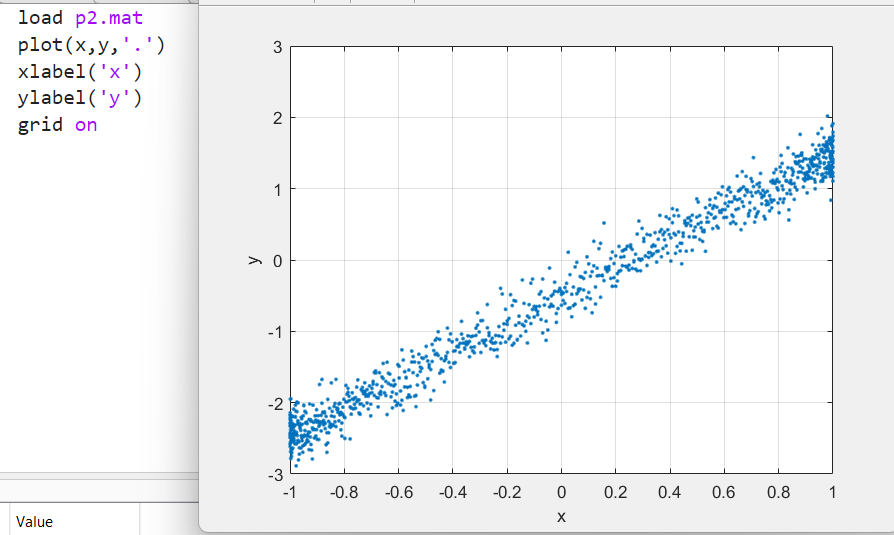
و نمودار رسم شده در نهایت:

بخش دوم

2-1) قطعه کد استفاده شده و شکل نمودار:

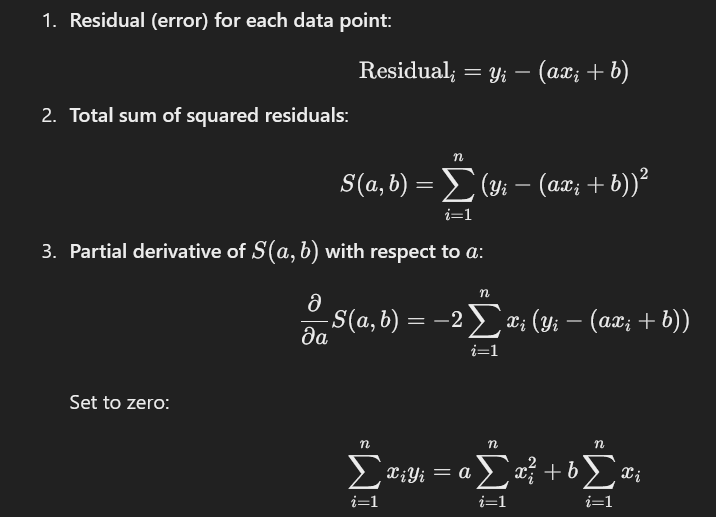
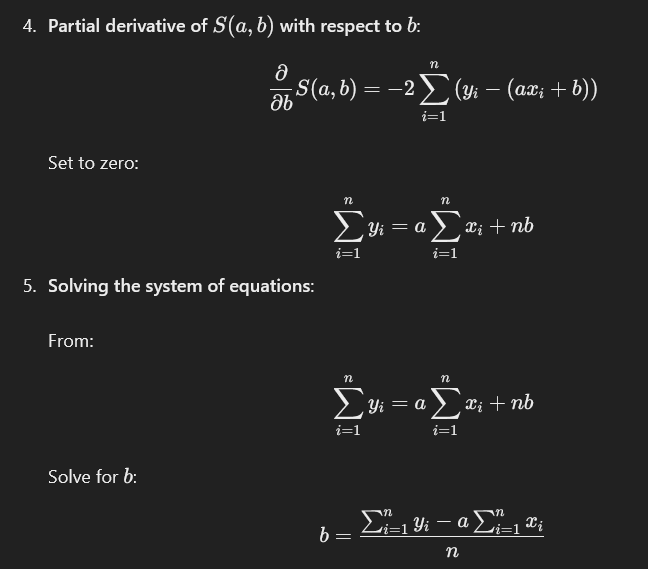
2-2) قطعه کد استفاده شده و شکل نمودار:

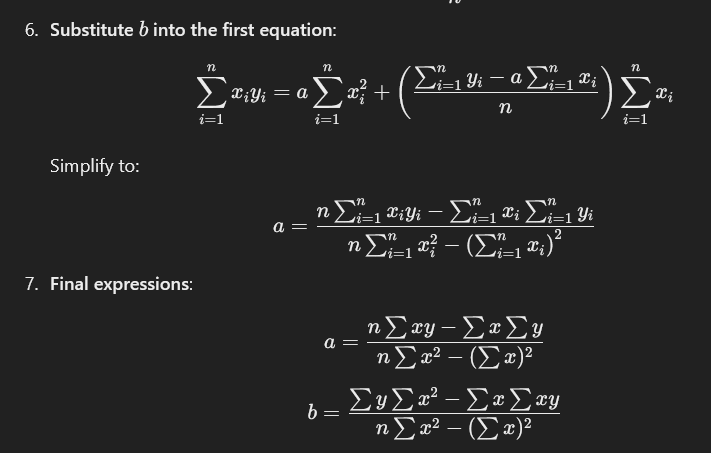


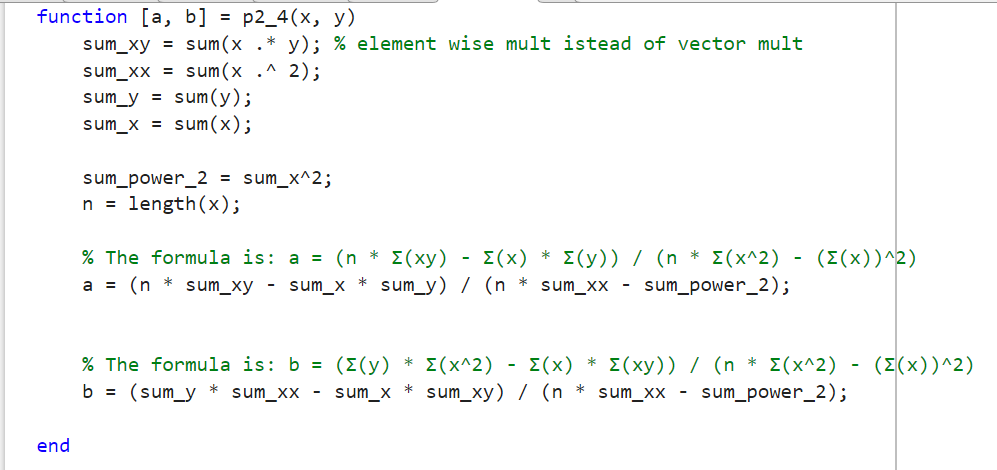
2-3) قطعه کد استفاده شده و شکل نمودار:

شیب خط برابر مقدار آلفا و عرض از مبدا برابر با مقدار بتا میباشد.

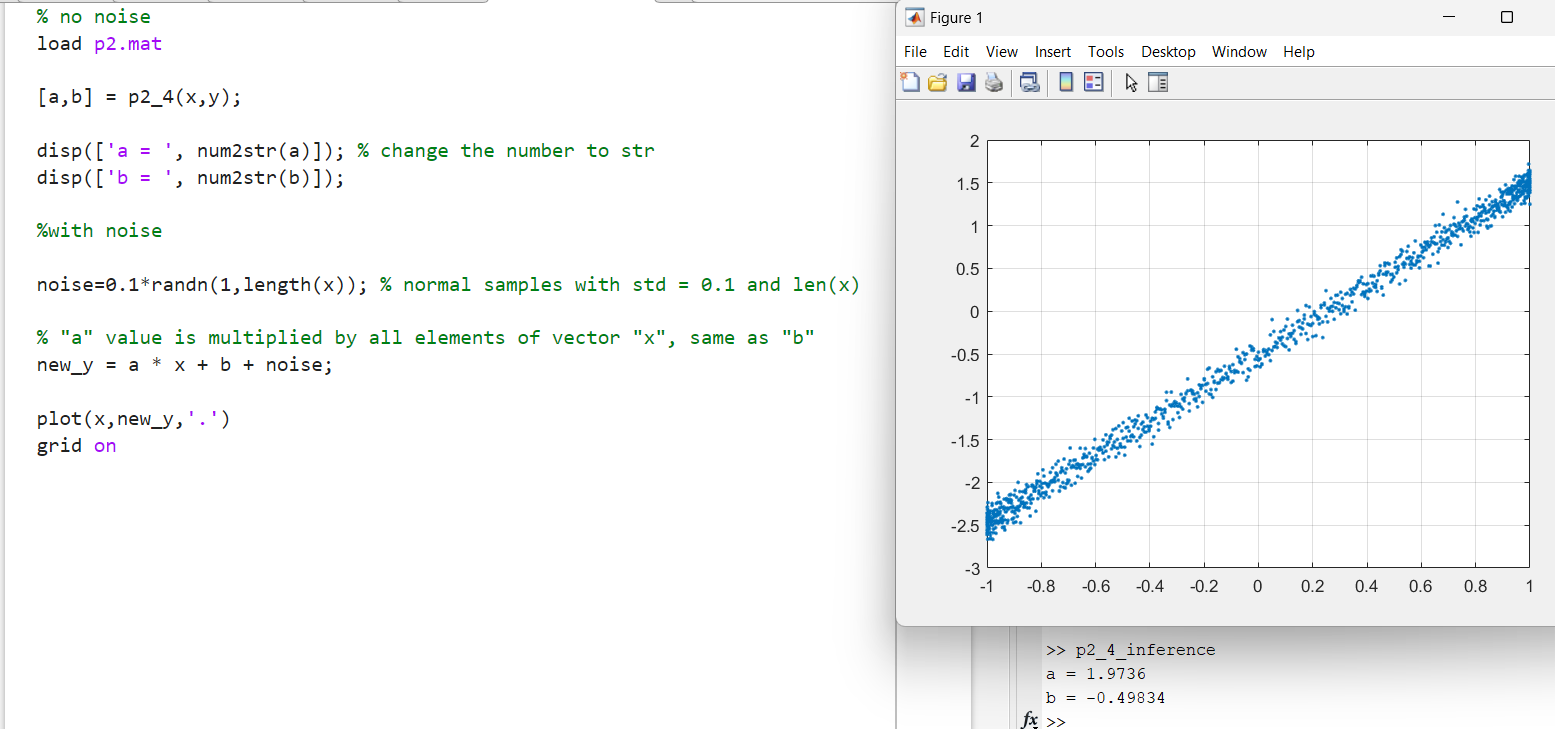
2-4) ابتدا از راهنمایی داده شده استفاده کرده و از معادله یک بار نسبت به آلفا و بار دیگر نسبت به بتا مشتق میگیریم، برابر با صفر قرار داده و مقادیر را برای آلفا و بتا حل میکنیم. توضیحات هوش مصنوعی برای جزئیات حل معادله به صورت زیر میباشد:





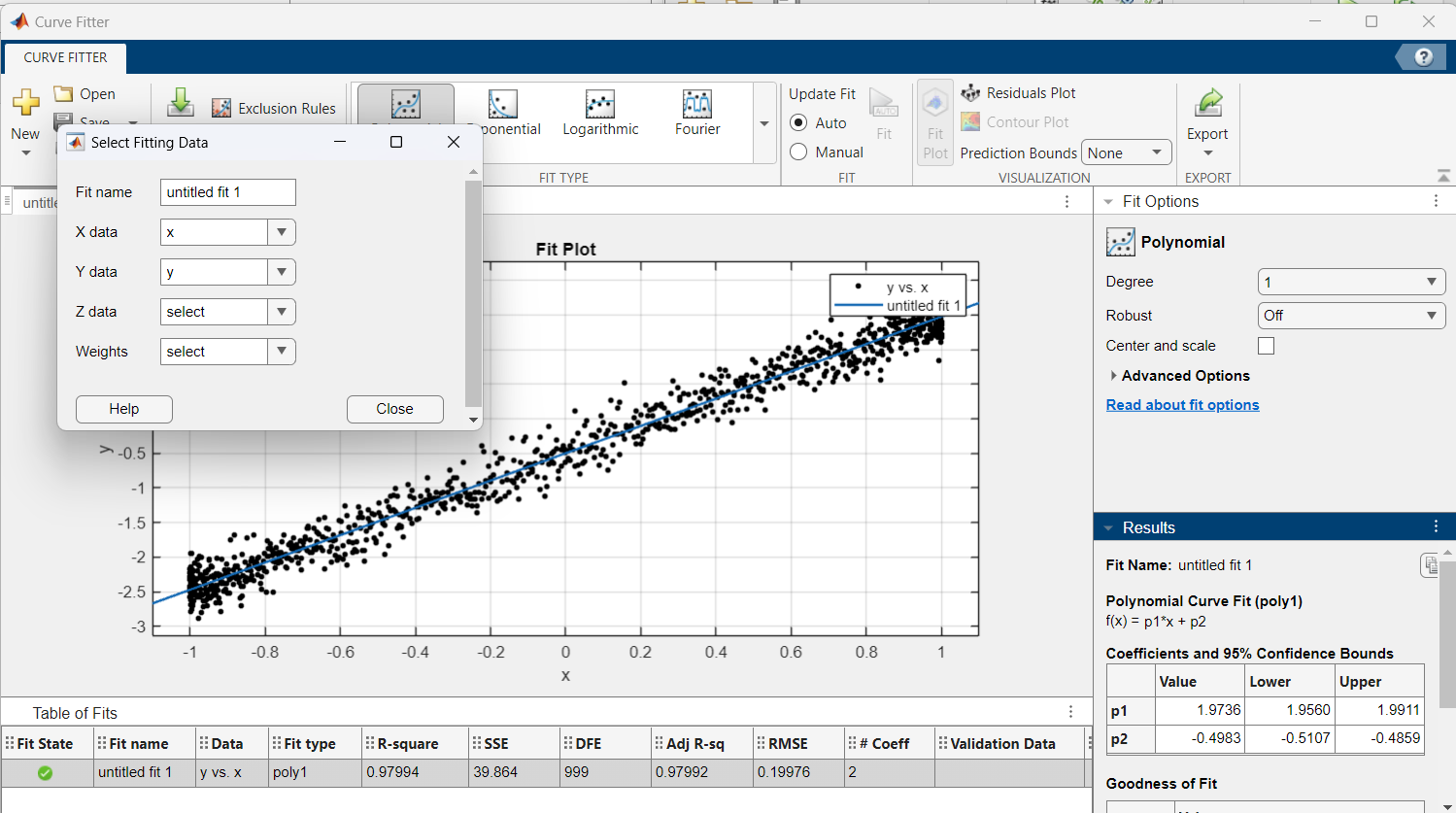
بنابراین و با توجه به توضیحات داده شده، اسکریپت زیر را مینویسیم (کامنت های مناسب برای توضیح گذاشته شده است):

حال داخل اسکریپت p2\_4\_inference هم حالت با نویز و هم حالت بدون نویز را بررسی میکنیم. کد این اسکریپت به همراه نمودار رسم شده با نویز و همچنین مقادیر a و b (در قسمت کامند ویندو) در تصویر زیر قابل مشاهده میباشند:



که شبیه به نمودار اولیه میباشد، و بنابراین صحت پاسخ ما را تایید میکند.

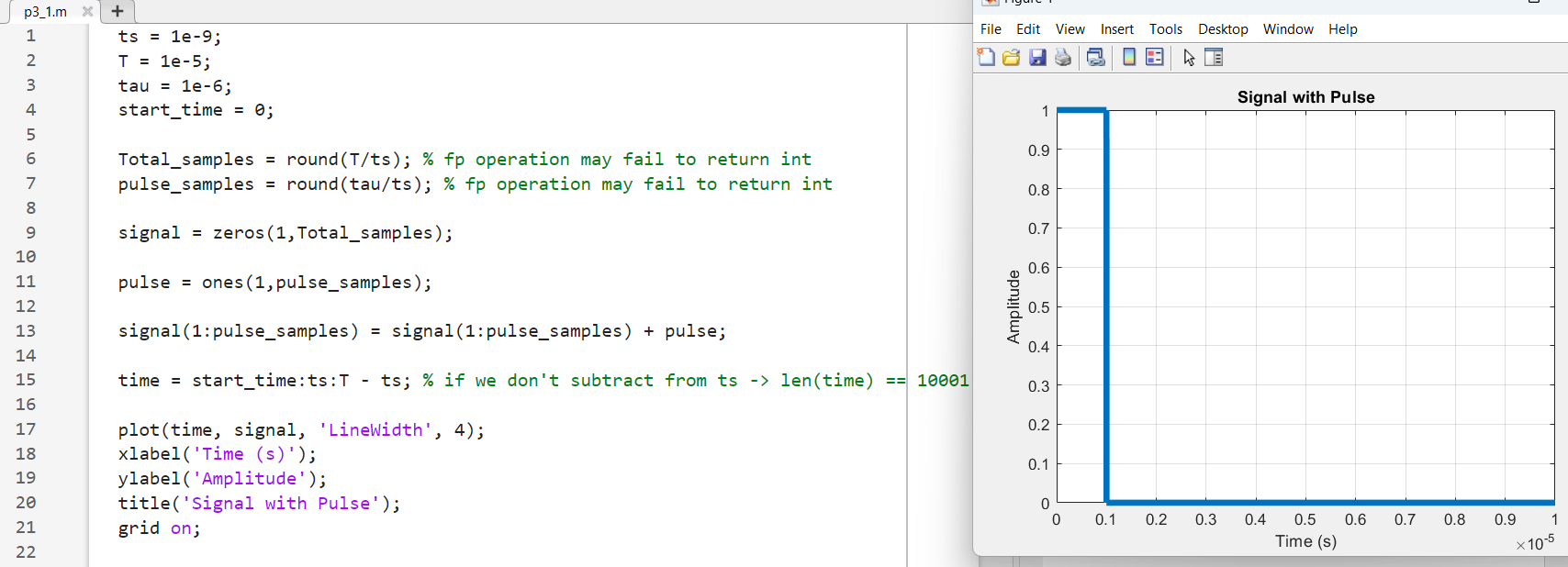
2-5) مراحل گفته شده را طی میکنیم و به نتیجه زیر میرسیم:



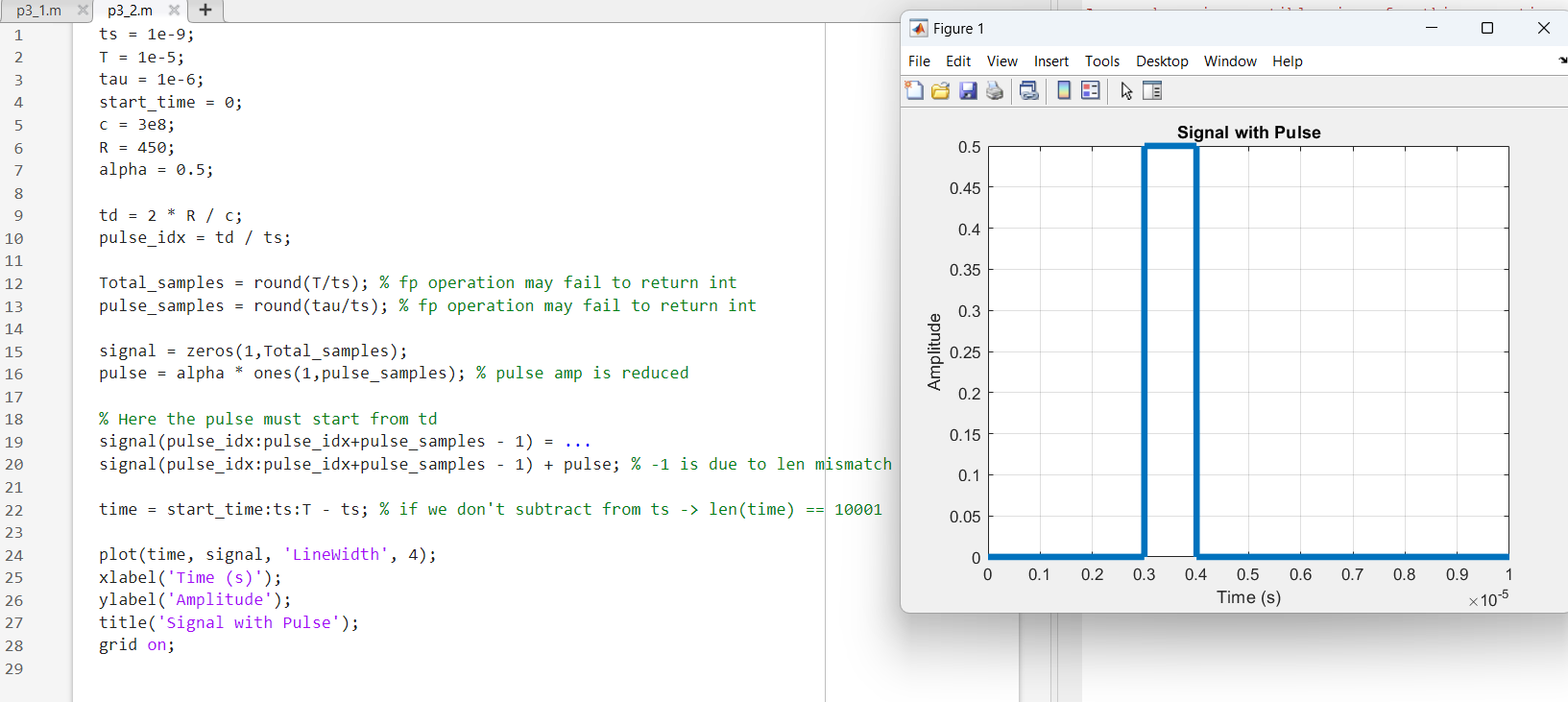
همانطور که در عکس مشاهده میشود، مقادیر به دست آمده با مقادیر کامپیوتری همخوانی دارند.

بخش سوم

3-1) قطعه کد مربوطه و شکل پالس در عکس زیر قابل مشاهده میباشد (کد با کامنت های مناسب توضیح داده شده):

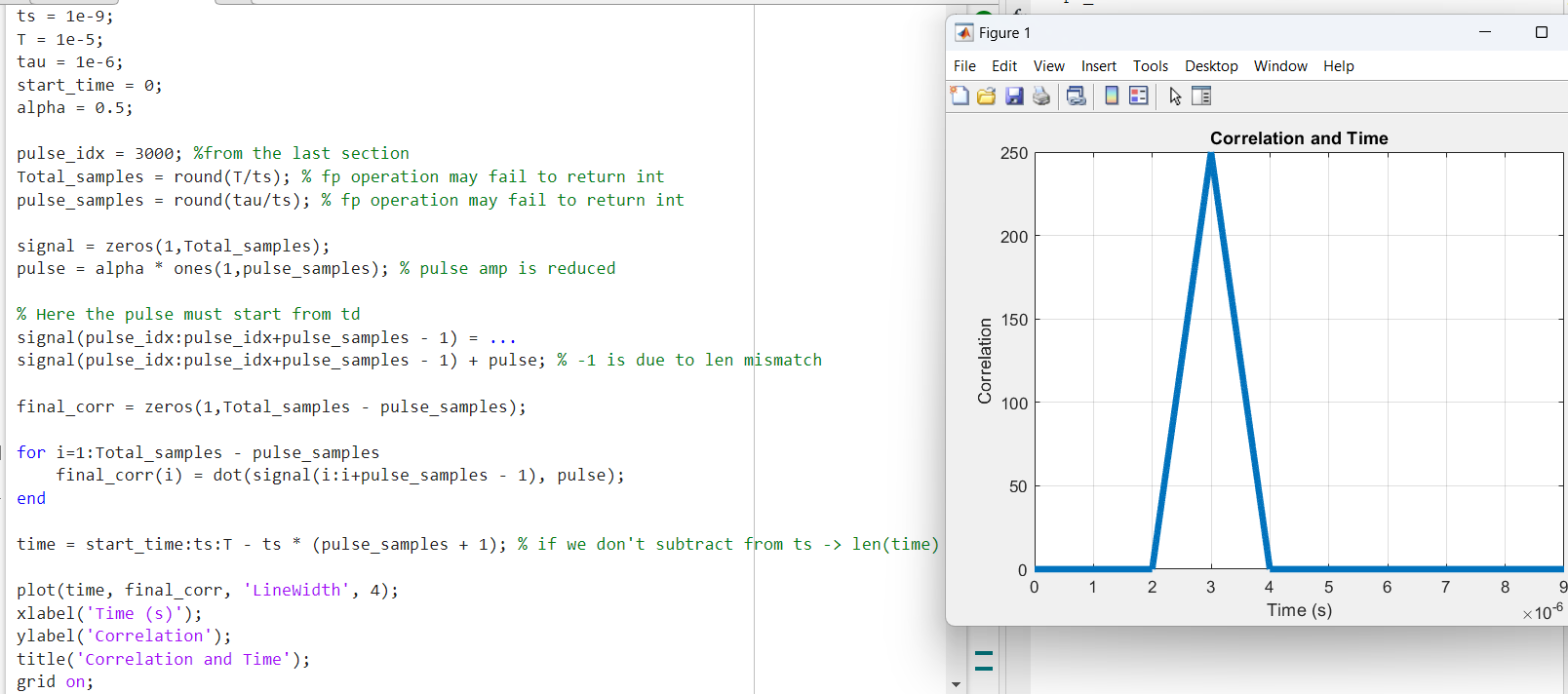


ابتدا یک سیگنال کلی به تعداد تمام سمپل های مورد نظر و به اندازه صفر درست میکنیم. در مرحله بعد، یک سیگنال به تعداد سمپل های پالس و دامنه یک درست کرده، پالس را به سیگنال کلی اضافه کرده و بردار زمان را میسازیم. نهایتا سیگنال را بر اساس بردار رسم میکنیم.

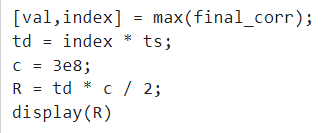
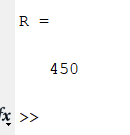
3-2) قطعه کد مربوطه و شکل پالس در عکس زیر قابل مشاهده میباشد (کد با کامنت های مناسب توضیح داده شده):

همانطور که مشاهده میکنید در ابتدا مقدار td محسابه شده، سپس ایندکس مربوطه به این زمان حساب شده و نهایتا پالس از ایندکس جدید و با دامنه نصف دوباره رسم شده است.

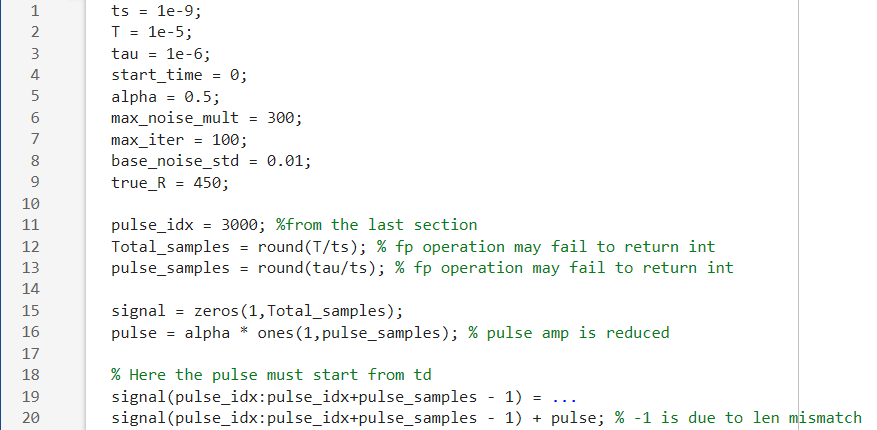
3-3) قطعه کد مربوطه و شکل کوررلیشن در عکس زیر قابل مشاهده میباشد (کد با کامنت های مناسب توضیح داده شده):

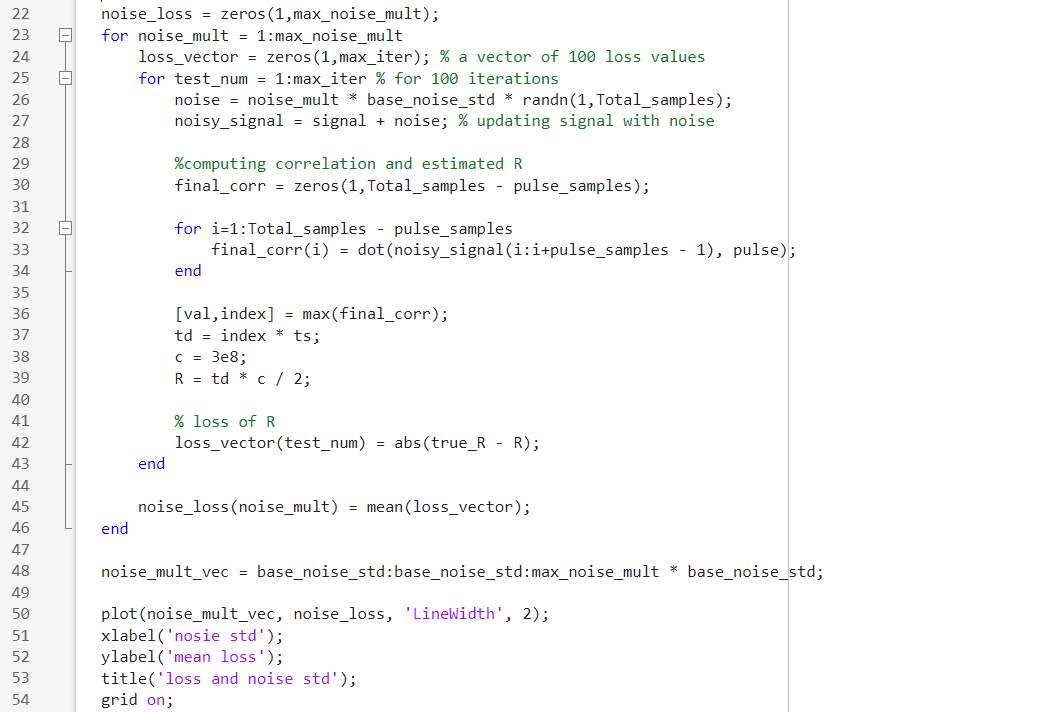


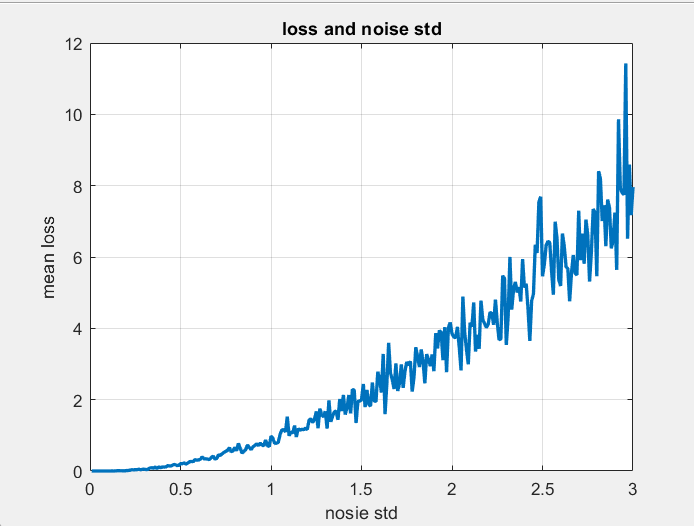
همانطور که مشاهده میشود، به فرض داشتن ایندکس سیگنال و ساختن سیگنال مربوطه، با استفاده از حلقه میتوانیم یک پالس را در طول سیگنال جا به جا کرده (دامنه پالس جا به جا شونده نیز 0.5 میباشد که در ایندکس پیک تاثیری نمیگذارد) و نهایتا مقادیر کوررلیشن را ذخیره کرده و بر اساس زمان رسم کنیم، که همانطور که مشاهده میشود به طور تقریبی در زمان 3 این نمودار پیک میزند. حال با استفاده از کد زیر میتوانیم دقیق زمان را و سپس فاصله را حساب کنیم:



که همان نتیجه ای است که در قسمت قبل به دست آوریم.

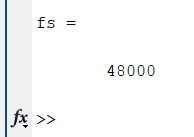
3-4) قطعه کد مربوطه در عکس زیر قابل مشاهده میباشد (کد با کامنت های مناسب توضیح داده شده):

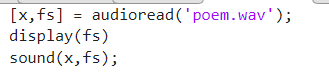


همانطور که مشاهده میشود ابتدا سیگنال طبق مراحل قبل تولید شده، سپس با 300 انحراف معیار مختلف، هر بار 100 نویز تولید شده و میانگین خطای این نویز ها به عنوان خطای مربوط به انحراف معیار متناظر گزارش شده است. نهایتا مقادیر این میانگین خطا ها رسم شده که به شکل زیر میباشد:

همانطور که مشاهده میشود، طبق انتظاری که داشتیم خطا با بالا رفتن قدرت نویز بیشتر میشود و رابطه ای مستقیم بین این دو وجود دارد.

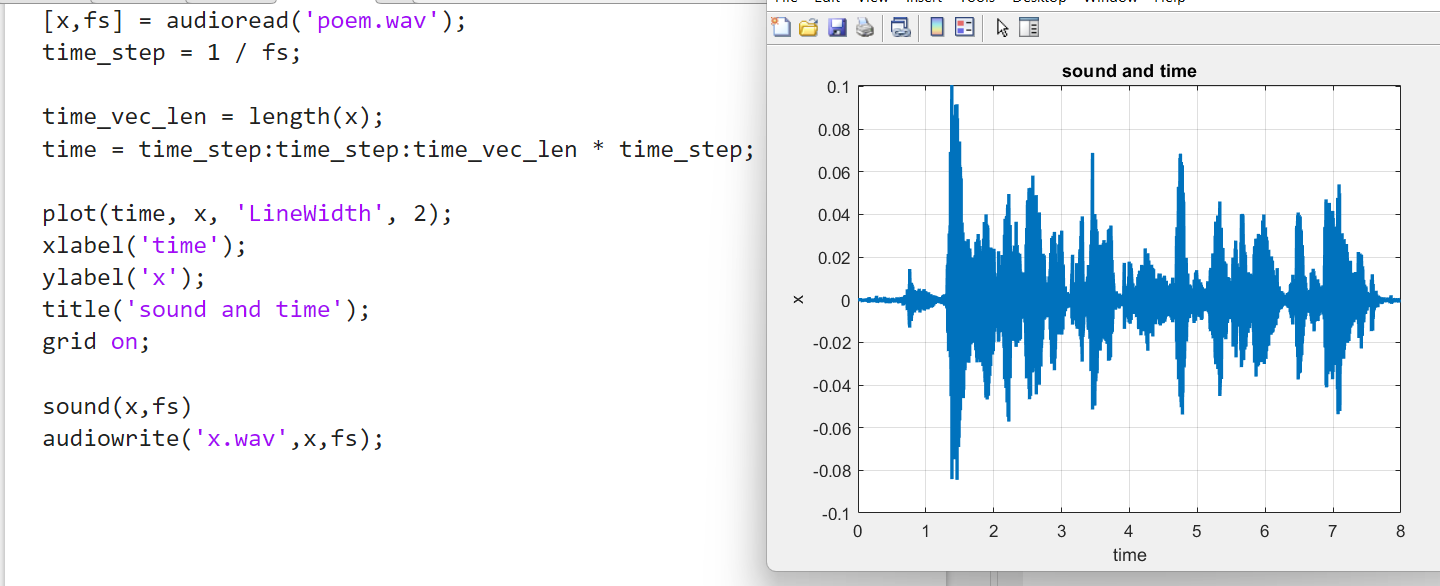
بخش چهارم

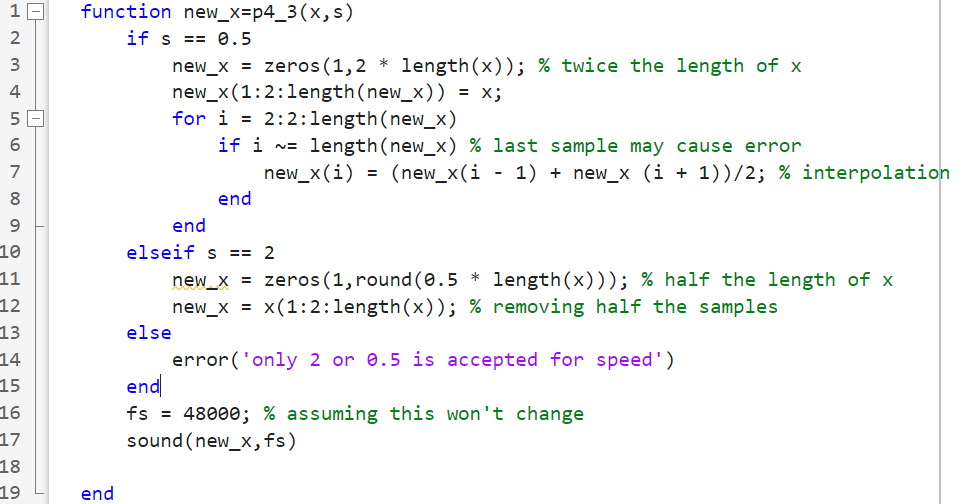
4-1) فایل ضبط شده را با کمک کد زیر لود کرده و فرکانس نمونه برداری آن را مشاهده میکنیم:



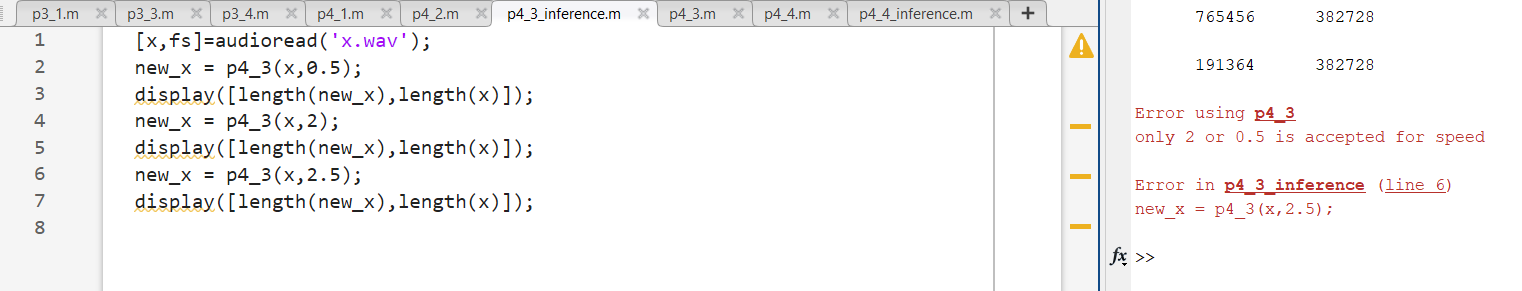
همانطور که مشاهده میکنید فرکانس نمونه برداری این فایل صوتی 48000 میباشد.

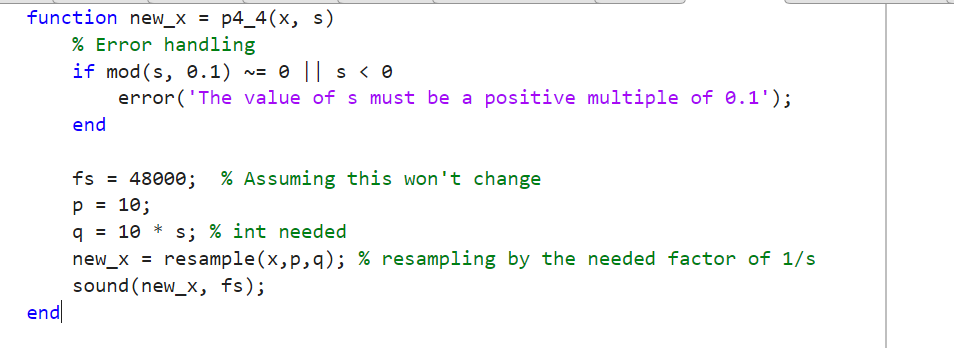
4-2) قطعه کد مربوطه و شکل موج صدا در عکس زیر قابل مشاهده میباشد:

در اینجا فاصله زمانی نمونه ها را ابتدا با تقسیم یک بر فرکانس حساب کرده و سپس با کمک طول x و مقدار این فاصله های زمانی، محور زمان خود را تشکیل داده و نهایتا مقادیر را رسم میکنیم.

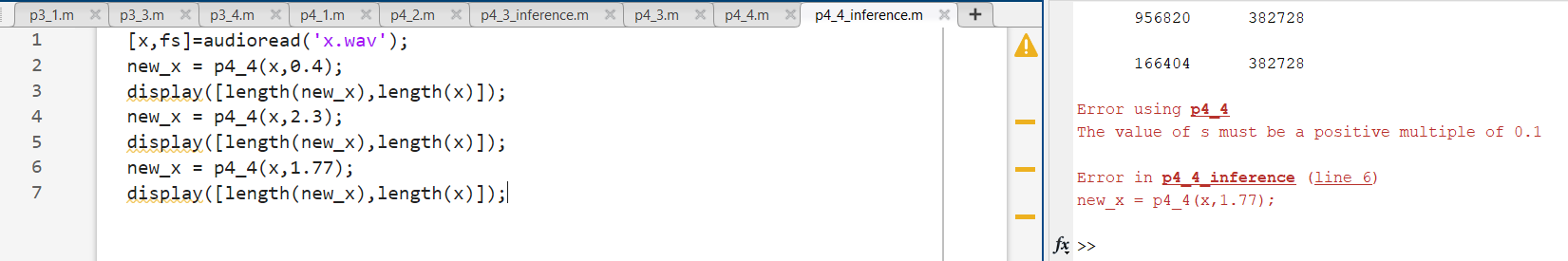
4-3) تابع نوشته شده به صورت زیر میباشد:

تابع را به صورت زیر، سه بار با مقادیر 0.5، 2 و غیر این دو تست میکنیم. همانطور که مشاهده میشود، طول وکتور برگشتی تغییر میکند و صدا نیز دچار تغییر میشود:



4-4) تابع نوشته شده به صورت زیر میباشد:

که در آن از تابع resample استفاده شده که دو مقدار صحیح p و q را گرفته و از سیگنال به نسبت p/q سمپل کم میکند (حالت تند شدن) یا به سیگنال به همین نسبت با کمک اینترپولیشن سمپل اضافه میکند. از طرفی مقادیر غیر بخش پذیر بر 0.1 یا منفی نیز در این روش مورد قبول نمیباشد. تابع در خطوط زیر تست شده است:



که همانطور که میبینید طول وکتور و صدای پخش شده با توجه به سرعت تغییر کرده و تابع نیز به درستی خطا ها را هندل میکند.