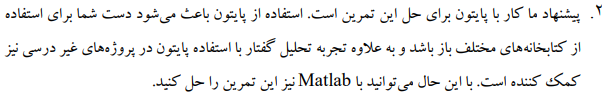
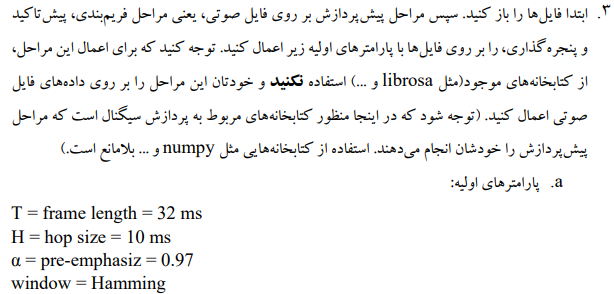
# مراحل







قطعه کد مربوط به این مرحله، در بخش 3 (part 3) واقع در کد‌ها آمده است. در این قسمت سه تابع pre\_emphasis، framing، windowing نوشته شده است که به ترتیب عملیات پیش‌تاکید، فریم‌بندی و پنجره‌گذاری بر روی سیگنال یا سیگنال‌های ورودی انجام می‌دهند. سپس تابع preprocess نوشته شده است که پارامتر‌های اولیه را دریافت می‌کند و بر روی سیگنال ورودی عملیات پیش‌پردازش (به ترتیب پیش‌تاکید، فریم‌بندی و پنجره‌گذاری) را انجام می‌دهد و فریم‌های پیش‌پردازش شده را خروجی می‌دهد.

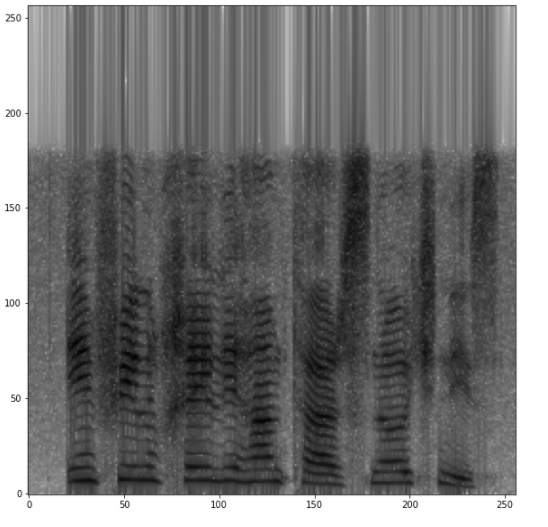
نکته) از تابع np.hamming برای پنجره‌گذاری hamming استفاده می‌شود.

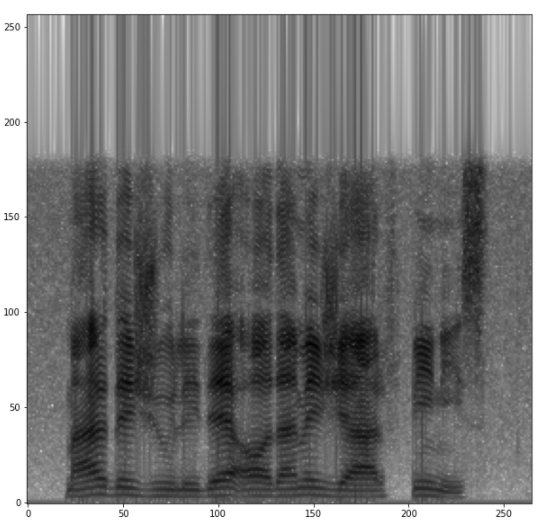


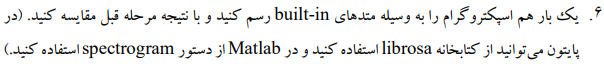
با استفاده از کلاس np.fft و تابع rfft تبدیل فوریه‌ی فریم‌ها را می‌گیریم و قدرمطلق آن‌ها را ذخیره می‌کنیم. قطعه کد این مرحله در بخش 4 فایل jupyter آمده است.



کد مربوط به این مرحله، در بخش 5 فایلjupyter آمده است. نمودار‌های اسپکتروگرام سیگنال‌های 1 و2 به ترتیب به صورت زیر می‌باشد. برای ترسیم اسپکتروگرام باید از فرمول استفاده شود و با به دست آمدن ماتریس مورد نظر و ترانهاده کردن آن، با کمک تابع imshow آن را ترسیم می‌کنیم.



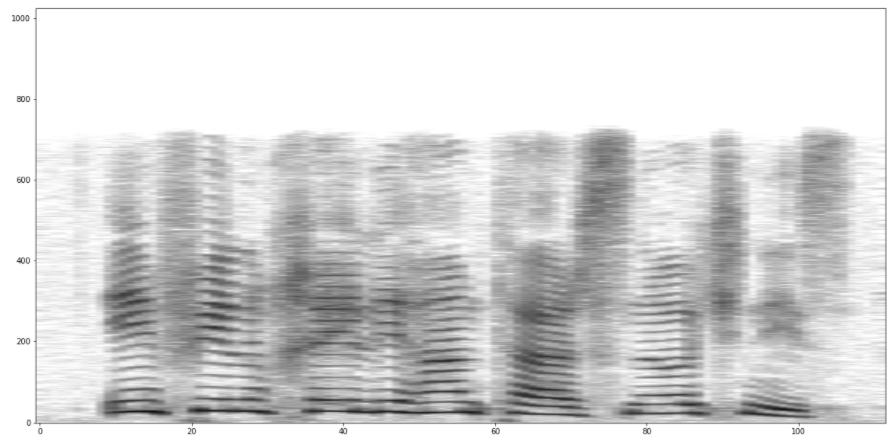


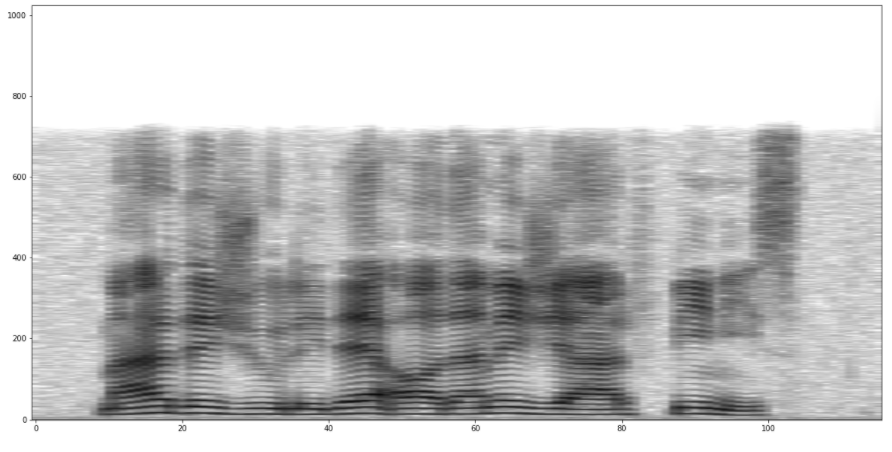


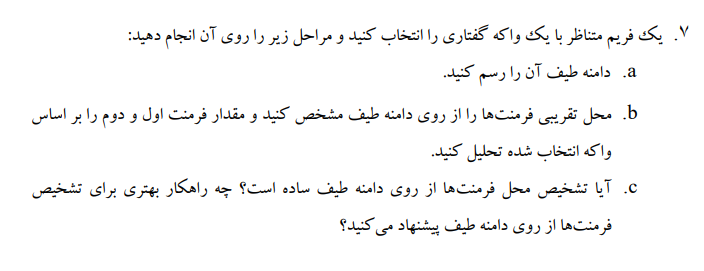
کد مربوط به این مرحله در بخش 6 آمده است. تابعی تحت عنوان spectogrm\_plt (به صورت lambda) نوشته شده است، که سیگنال خوانده شده گرفته و اسپکتروگرام آن را ترسیم می‌کند.

تصاویر به دست آمده، خیلی مشابه تصاویر قبلی می‌باشد و به مانند آن است که تمام اعداد ماتریس قبلی (آن که خودمان محاسبه کردیم.) چند برابر اعداد ماتریس فعلی است. (رنگ‌بندی قبلی پر رنگ تر است.) همچنین ابعاد ماتریس در دو بخش متفاوت است.

تصاویر جدید به ترتیب به صورت زیر است:

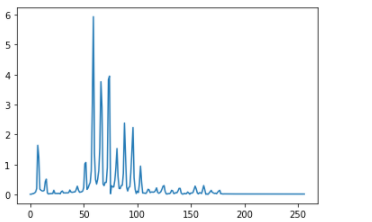




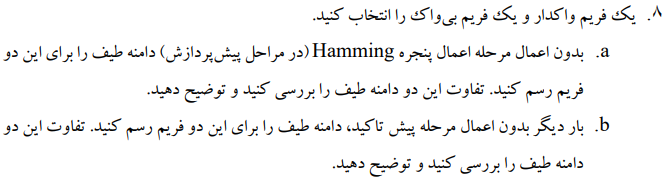


برای پیدا کردن یک واکه، فریم دارای بیشترین میانگین انرژی را دارا می‌باشد، انتخاب شده است. انرژی وابسته به مجذور magnitude است و اگر بیشترین میانگین مجذور magnitude فریم‌ها را بدست بیاوریم، یک واکه بدست آمده است.

کد مربوط به این مرحله در بخش 7 آمده است که نشان می‌دهد فریم با اندیس 52 یک واکه است. نمودار آن به صورت زیر است.



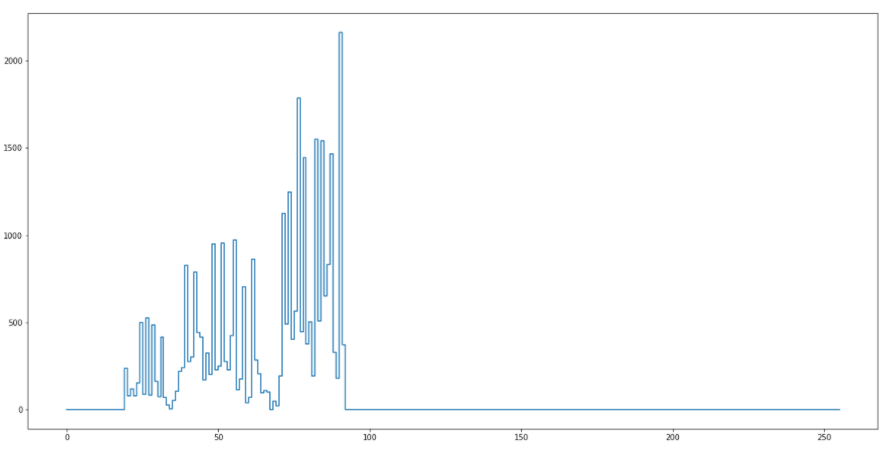
همچنین از اول تا فریم 52 ام پخش شد که نشان می‌دهد برای واکه‌ی /ای/ است.



مقادیر میانگین pitch هر فریم را بدست می‌آوریم. بی واک به معنای صفر بودن این مقدار است و واکدار عدم صفر بودن. فریم با اندیس 68 مربوط به یک بی‌واک و فریم با اندیس 29 مربوط به یک واکدار است.

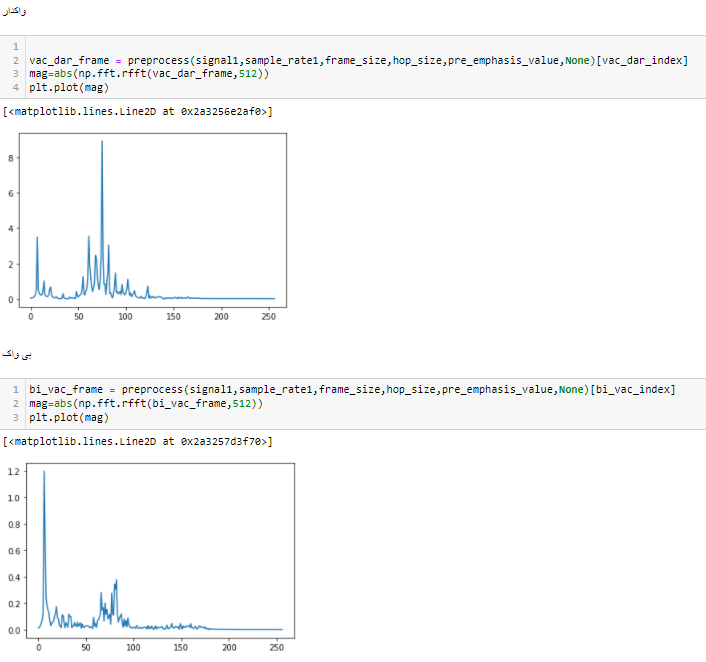
قطعه کد پیدا کردن pitch در قسمت PART 8 آمده است.

همچنین نمودار پله‌ای مقدار میانگین pitch بر حسب شماره‌ی فریم به صورت زیر است.



بخش a:

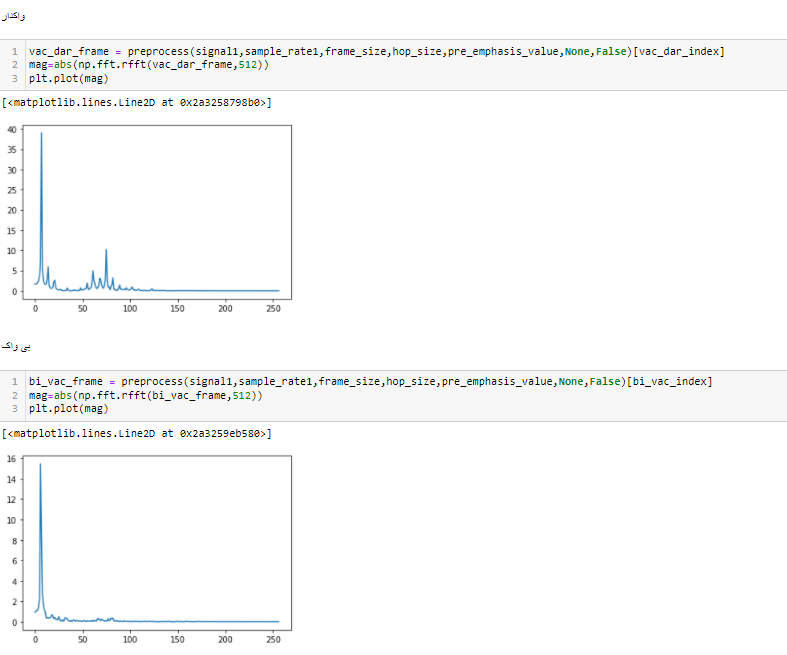
نمودار دامنه‌ی طیف فریم واکدار و بی واک بدون پنجره‌ی hamming به صورت زیر است. تفاوت آن‌ها مقدار دامنه‌ی طیف آن‌ها است.

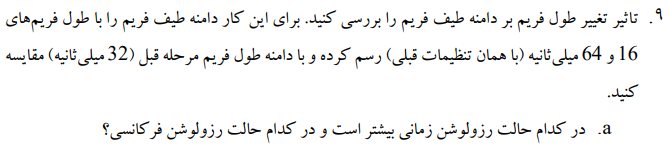


بخش b:

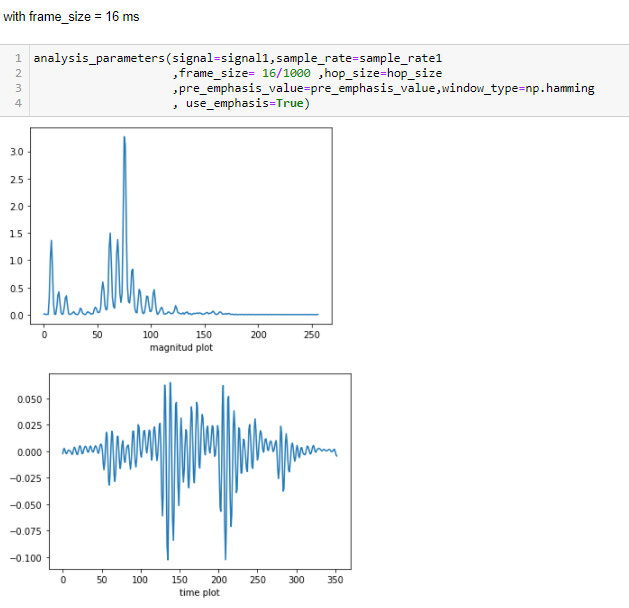
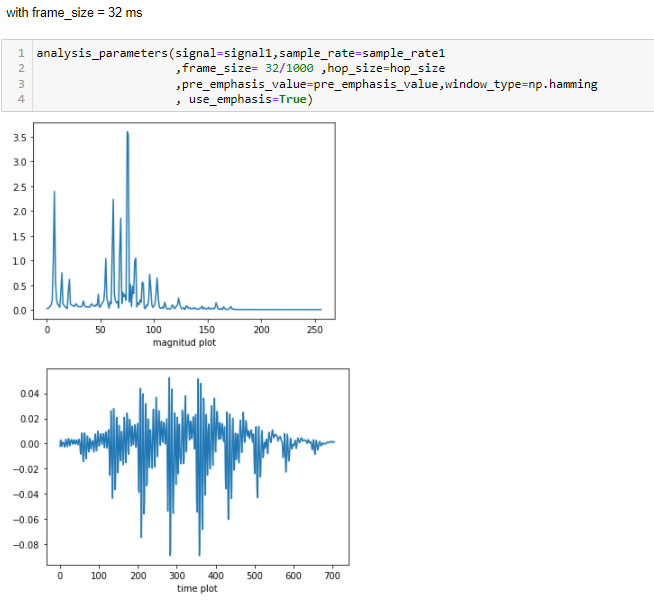
با تغییر پارامتر use\_emphasis به مقدار False عملیات‌های بخش قبل را تکرار کردیم و اشکال زیر به دست آمده است.

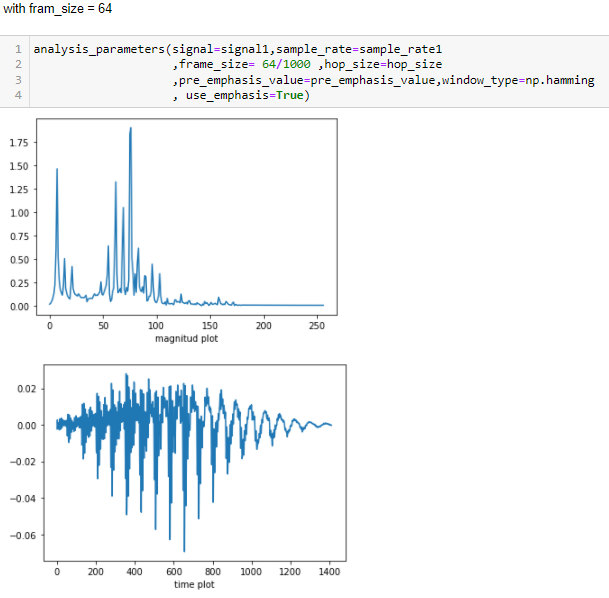
به دلیل آن که epmphasis تغییرات را در نظر می‌گیرد، در جاهایی که تغییر بیشتر داشتیم نمودار زیاد می‌شود. همچنین باز هم تفاوت این دو نمودار در مقادیر دامنه طیف است.



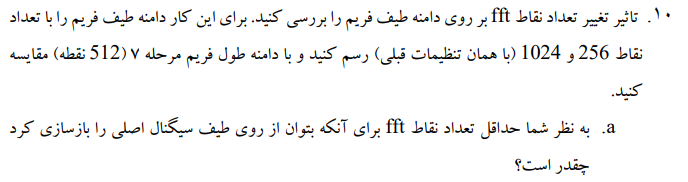


در این قسمت، یک تابع نوشته شده است که در آن پارامتر‌های preprocessing گرفته شده و برای فریم مثلا 30 ام نمودار‌های دامنه‌ی طیف و دامنه‌ی زمانی (خود فریم) کشیده می‌شود. برای سه حالت گفته شده نمودار‌ها به صورت زیر است:

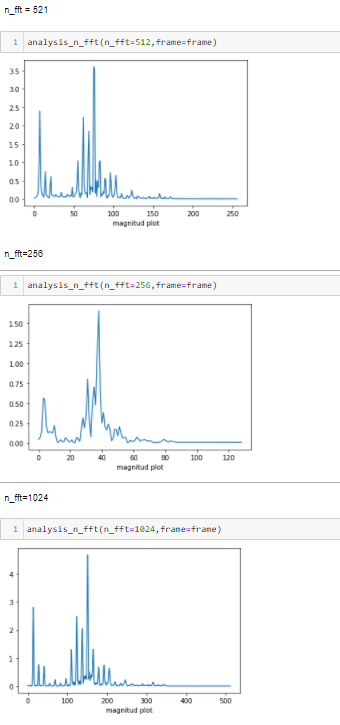




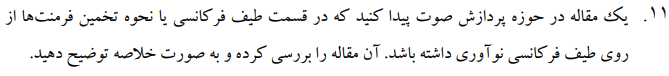
در حالت frame-size=16 رزولوشن زمانی بیشتر است و در حالت frame-size=64 رزولوشن فرکانسی



در این بخش تابعی نوشته شده است که تعداد نقاط fft و فریم را در ورودی می‌گیرد و نمودار طیف آن را ترسیم می‌کند که برای مقادیر بالا به صورت زیر است:



حداقل مقدار n\_fft باید نصف مقدار sample\_rate باشد تا مشکل aliasing رخ ندهد.



مقاله‌ی Formant extraction from linear‐prediction phase spectra را انتخاب کردم.

DOI : <https://doi.org/10.1121/1.381864>

در این مقاله گفته است که با استفاده از مشتق طیف فاز می‌توان فرمنت‌های دقیقی به دست آورد. کاربرد این روش از طریق نمونه‌هایی از طیف‌های پیش‌بینی خطی به‌دست‌آمده برای مدل‌های شبیه‌سازی‌شده و برای بخش‌های گفتار واقعی نشان داده شده است.