



گزارش تمرین امتیازی اصول مهندسی پزشکی

عنوان:

پردازش و طبقه‌بندی سیگنال الکترومایوگرام

آرشام لولوهری

۹۹۱۰۲۱۵۶

آرمین نوردی

99105129

استاد:

دکتر جاهد

## چکیده

در این تمرین از داده های اخذ شده در آزمایشگاه مهندسی پزشکی در جلسه آزمایشگاه این درس (که ثبت حرکات مچ و دست انجام شد) استفاده شده است. این داده در پیوست فایل ها قرار دارد (Exp1.mat) و شامل ۶ رکورد است که ۴ رکورد آخر برای حرکات مچ و دست بوده و در این تمرین از آنها استفاده شده است. در این داده، فرکانس نمونه برداری دستگاه ۲۰۰ هرتز بوده است. حرکات دست در این آزمایش شامل (به ترتیب) موارد زیر است:

۱. باز و بسته کردن کف دست

۲. بالا و پایین کردن مچ

۳. چرخش مچ

۴. بالا و پایین کردن کف دست

سیستم ثبت نیز دستگاه PowerLab بوده و از دو الکتروود EMG، یکی روی بخش داخلی و یکی بخش بیرونی ساعد و در نزدیکی مچ، استفاده شده است.

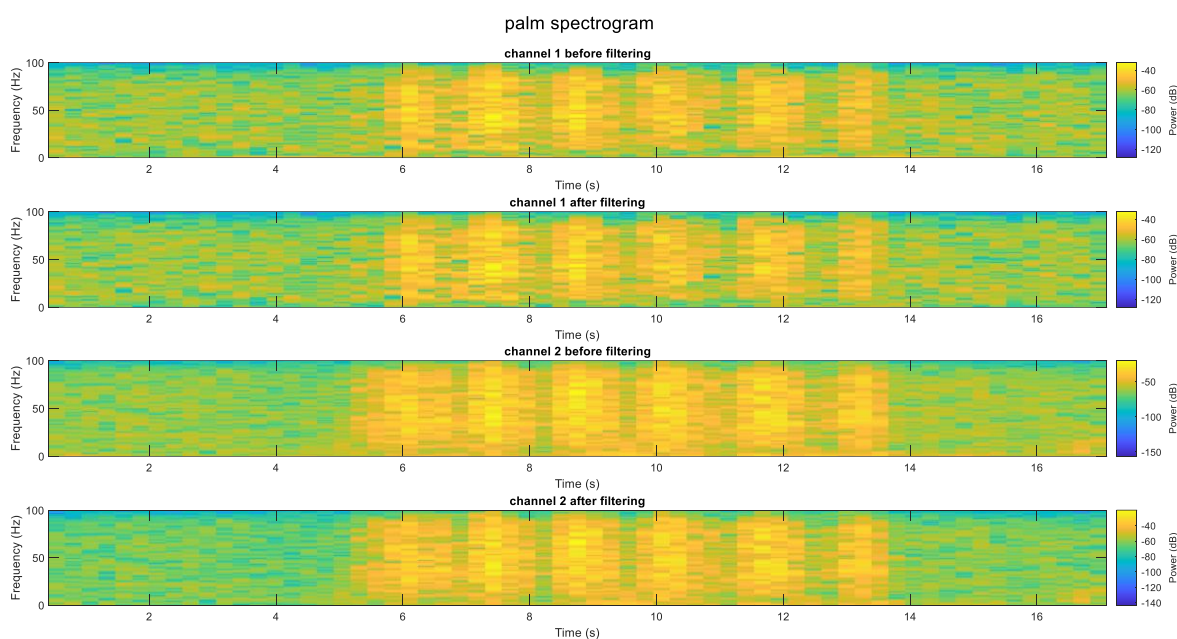
برای پیش پردازش و نیز طبقه بندی داده، از نرم افزار متلب استفاده شده است.

در این تمرین از الگوریتم های KNN و Random Forest برای طبقه بندی داده های تست استفاده شده که در بخش (ه) از گزارش، جزئیات آنها مطرح خواهد شد.

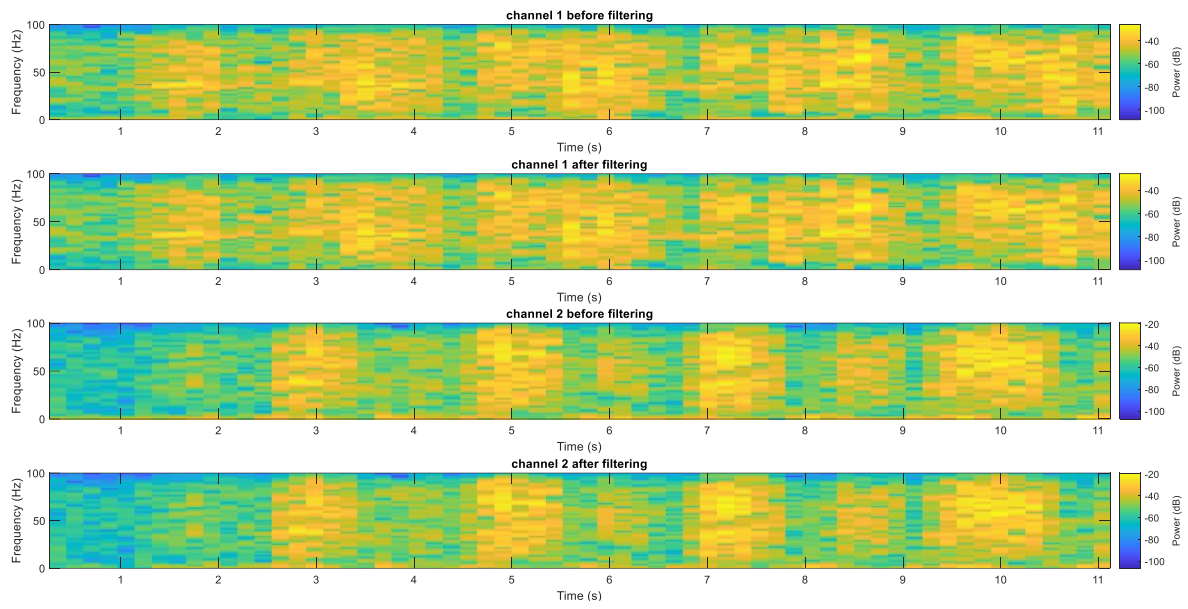
## پیش پردازش

(الف)

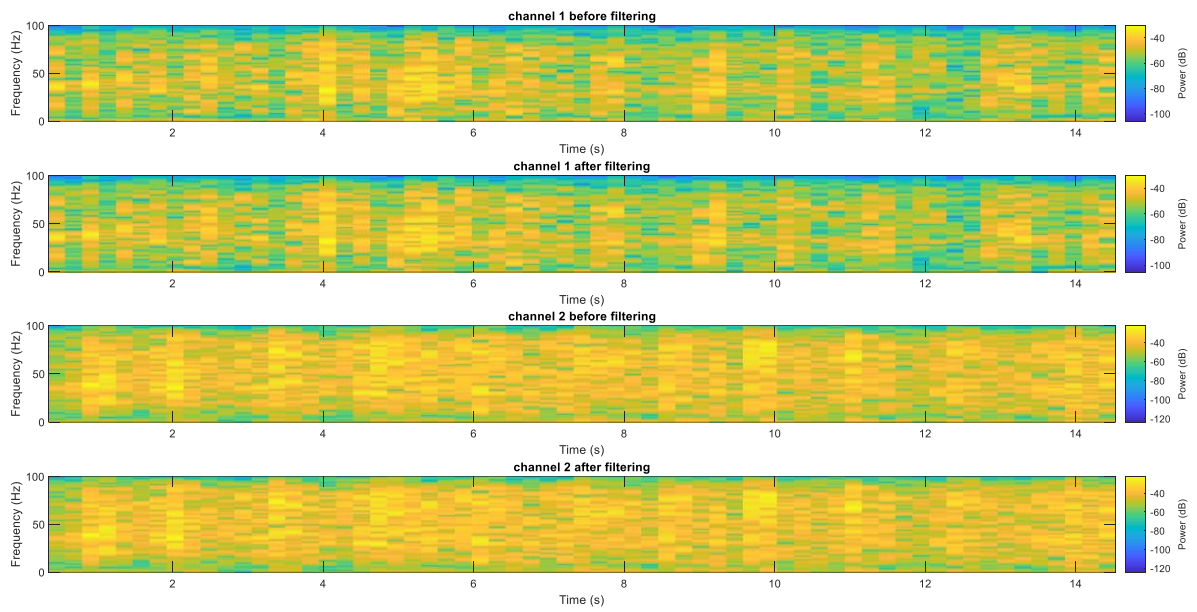
داده با استفاده از فیلتر Butterworth مرتبه یک، ابتدا از یک فیلتر بالاگذر با فرکانس قطع ۱ هرتز (برای حذف آفست سیگنال) و سپس یک فیلتر Notch ۵۰ هرتز (برای حذف نویز برق شهر) عبور میکند. سپس طیف این سیگنال ها بر حسب زمان با استفاده از تابع spectrum متلب، و برای هریک از چهار نوع حرکت دست، قبل و بعد از فیلتر رسم میشود (تبدیل زمان-فرکانسی):

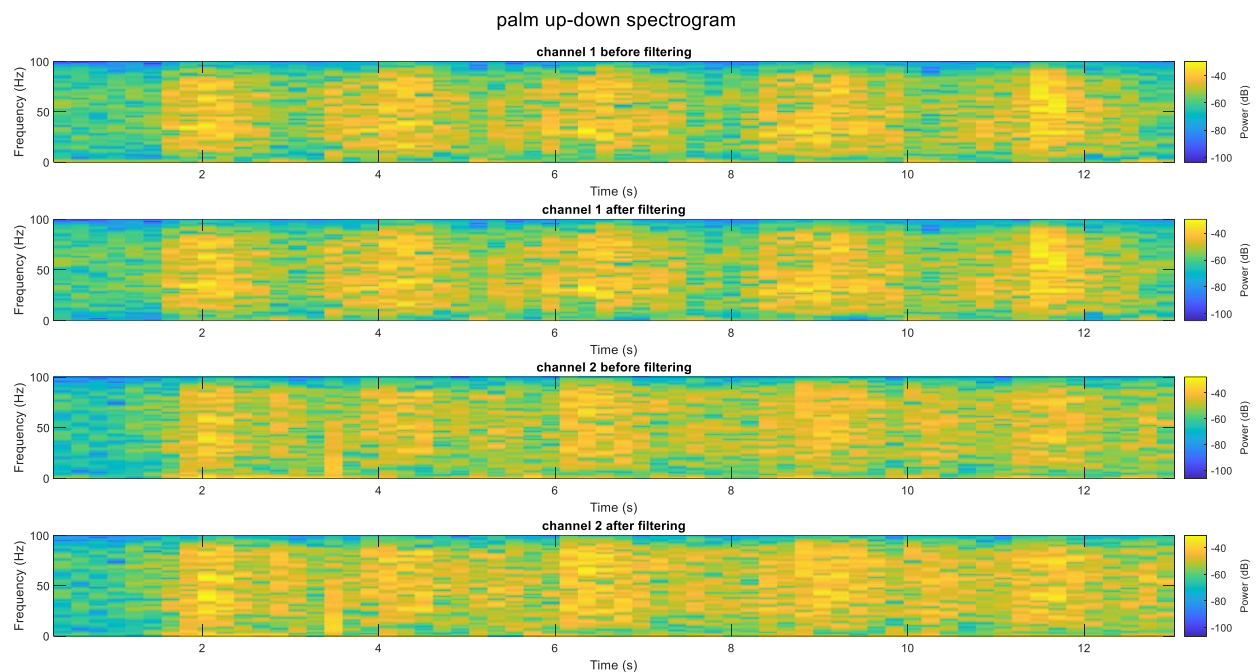


### wrist up-down spectrogram



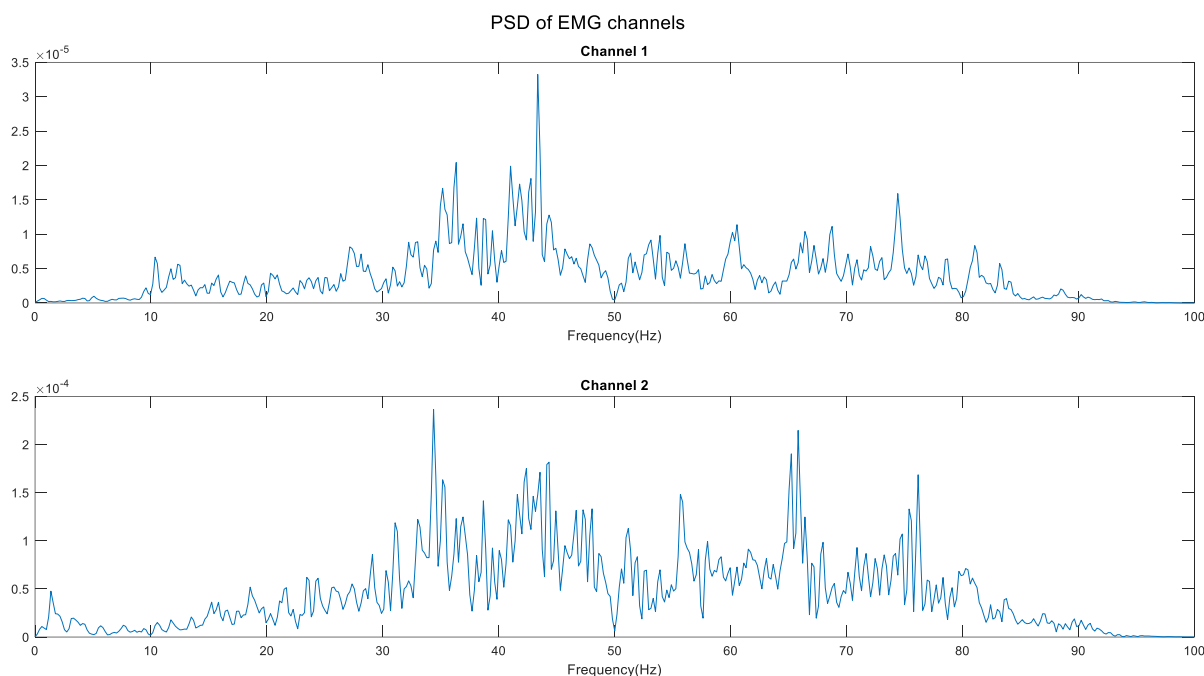
### wrist rotate spectrogram





در ماتریس **periods** در کد، زمان شروع حرکات دست در هر رکورد نوشته شده است. برای مثال، بازه انجام حرکات در رکورد اول (باز و بسته کردن کف دست) از حدود زمان 5s تا 13.5s بوده. در **spectrum** این رکورد نیز (تصویر اول) میبینیم که محتوای فرکانسی اصلی در همان حدود زمانی، و در بازه هایی قرار دارد که انقباض های عضلانی برای حرکت انگشتان انجام شده است. ضمناً میدانیم محتوای فرکانسی سیگنال های **EMG** میتواند تا ۱۰۰۰ هرتز نیز باشد و به همین دلیل، حین فعالیت، طیف در راستای محور عمودی گسترده شده است.

سپس از نمایش **Pwelch** نیز استفاده میکنیم. این تابع در پنجره های زمانی مختلف طیف را حساب کرده و میانگین طیف در تمام پنجره ها را رسم میکند که نتیجه برای دو کانال به صورت زیر است:



که اثر حذف فرکانس های پایین و نیز برق شهر در آن مشهود است.

## ب و ج)

داده به روش **z-score** نرمالیزه میشود طوری که میانگین سیگنال هر کانال، صفر و انحراف معیار آن یک شود (این گام در این مرحله انجام شده تا هردو دسته دادگان آموزشی و تست، به روش مشابهی نرمالیزه شوند و در شرایط متفاوتی وارد طبقه‌بند نشوند). سپس با استفاده از ماتریس **periods** که بالاتر توضیح داده شد، بازه هایی که انقباض و حرکت دست صورت نگرفته، حذف (**crop**) میشوند تا تنها سیگنال های مهم را داشته باشیم.

سپس پنجره بندی به نحوه ذکر شده انجام میشود. طول هر پنجره 0.2 ثانیه بوده و مقدار **overlap** آنها ۱۰ میلی ثانیه است. در این مرحله ماتریس **data** ساخته میشود که بعد اول آن شامل تمام پنجره ها (از هر ۴ رکورد)، بعد دوم آن مربوط به دو کانال ثبت کننده، و بعد سوم آن بعد زمان در هریک از این پنجره هاست. لیبل های متناظر با هر پنجره نیز (از ۱ تا ۴) در بردار **y** ذخیره میشود.

## طبقه‌بندی

(د)

در کد EMG\_classification.m، در ابتدا تمام ویژگی‌های مدنظر، برای هر پنجره زمانی از ماتریس data حساب میشود. این ویژگی‌ها توسط تابع extract\_features محاسبه میشود که شامل ویژگی‌های زیر است که برای هر دو کانال حساب میشوند:

- Mean Absolute Value
- Zero-cross rate
- Variance
- فراوانی دامنه سیگنال در ۱۰ دسته مختلف (توسط histcounts)
- ضرایب مدل AR با رتبه ۱۰. این مدل برای تخمین هر سمپل از سیگنال توسط ۱۰ سمپل قبلی آن استفاده میشود که نیاز به ۱۰ ضریب برای این کار دارد:

$$y(t) = \sum_{i=1}^P a(i) \cdot y(t-i) + \epsilon(t)$$

- فرم فاکتور، که نسبت پویایی مشتق سیگنال به پویایی سیگنال است:
- $$FF = (Var(x'')/Var(x'))/(Var(x')/Var(x))$$
- طول موج سیگنال
  - مقدار RMS
  - چگالی طیف توان یا PSD
  - فرکانس بیشینه (با بیشترین محتوای فرکانسی در PSD)
  - فرکانس میانگین (میانگین فرکانس‌های طیف توان)
  - فرکانس میانه در چگالی طیف توان
  - انرژی طیف سیگنال
  - Correlation یا ضریب همبستگی سیگنال دو کانال

تمام این ویژگی‌ها برای هر پنجره حساب شده و ماتریس ویژگی‌های X را میسازند.

به ازای هریک از ۴ لیبل موجود، ۸۰ درصد پنجره ها بعنوان داده های آموزشی، و مابقی پنجره ها بعنوان داده تست انتخاب میشوند. سپس ماتریس ویژگی های  $X$  بر این اساس به دو دسته آموزش و تست تقسیم میشود. دسته آموزش برای آموزش طبقه بند ها و دسته تست برای تست آن استفاده میشود.

ه) در این تمرین از دو الگوریتم **KNN** و **RF** برای طبقه بندی استفاده شده است:

۱. روش **KNN** بر اساس داده های آموزشی طبقه بندی شده، داده های تست را بدین صورت

طبقه بندی میکند که:

I. ابتدا پارامتر  $k$  (تعداد همسایه ها) را میگیرد، که در این کد قرار داده شده  $k=5$

II. سپس به ازای هر داده تست،  $k$  داده نزدیک به داده تست (که بطور پیش فرض با فاصله

اقلیدسی سنجیده میشود) را بررسی میکند.

III. از بین این  $k$  داده، تعداد داده های با هر لیبل (از ۱ تا ۴) شمرده میشود

IV. لیبل با بیشترین فراوانی، به داده تست جدید نیز نسبت داده میشود

در متلب با استفاده از **ficknn** میتوان این طبقه بند را طراحی کرد.

۲. روش **Random Forest** داده های آموزشی را به زیرمجموعه هایی تقسیم میکند و آموزش بر

روی هریک از این زیرمجموعه ها به صورت جداگانه انجام میشود که هریک از آنها یک

**Decision Tree** نام دارند. سپس برای داده های تست، هریک از این **Decision Tree** ها بر

اساس مدل یادگیری شده خود، تصمیمی در مورد لیبل داده تست میگیرند، و لیبل با بیشترین رای

به داده تست تخصیص داده میشود. در متلب با استفاده از **fitcensemble** میتوان این طبقه بند را

پیاده کرد.

پس از آموزش طبقه بند ها، لیبل های داده های تست را به هر دو روش استخراج میکنیم (با

استفاده از تابع **predict**). دقت یا **accuracy** روی داده های تست به صورت نسبت تعداد لیبل

های ثبت شده و صحیح، به تعداد کل لیبل هاست. بر این اساس، دقت را روی داده های آموزشی



و تست حساب میکنیم که به مقادیر 0.53 و 0.99 و 0.70 و 0.95 (به ترتیب داده تست و

آموزشی روش KNN، و داده تست و آموزشی روش Random Forest) میرسیم:

accuracy test knn: 5.337915e-01

accuracy train knn: 9.920046e-01

accuracy test random forest: 7.044674e-01

accuracy train random forest: 9.571673e-01

ماتریس Confusion نیز بر اساس لیبل های پیش بینی شده (ستون ها) و لیبل های صحیح (سطر ها) برای

داده تست، به صورت زیر است (جدول اول روش KNN و جدول دوم روش RF):

Confusion Matrix				
KNN				
True Class	1	2	3	4
	80	17	35	34
	31	140	16	8
	39	18	139	79
	55	22	53	107
Random Forest				
True Class	1	2	3	4
	124	3	22	17
	30	143	9	13
	12	24	159	80
	10	4	34	189
Predicted Class				

مشاهده میشود که عمده مقادیر روی قطر اصلی ماتریس ها قرار دارند؛ بدین معنا که بیشتر لیبل ها به درستی تشخیص داده شده اند. در روش KNN تعدادی از حرکات چرخش مچ (لیبل ۳) به اشتباه چرخش کف دست به بالا و پایین (لیبل ۴) تشخیص داده شده اند (با ۷۹ لیبل اشتباه). در روش Random Forest نیز این اشتباه رایج ترین اشتباه بوده است (با ۸۰ لیبل اشتباه).

(ی)

یکی از دلایل اشتباه های تشخیص میتواند شباهت نوع حرکات و عضلات درگیر در آن حرکات باشد (مثل لیبل ۳ و لیبل ۴ و شباهت حرکات آنها در چرخش دست) که در نتیجه آن فعالیت عضلانی مشابهی در دست ایجاد خواهد شد. در صورت وجود چنین شباهت هایی در حرکات متناظر با لیبل ها، میتوان از تعداد ویژگی های بیشتری استفاده کرد و نیز ترکیبی از ویژگی های آماری و ویژگی های فرکانسی را استفاده کرد.

ضمناً فرکانس نمونه برداری دستگاه PowerLab برابر با ۲۰۰ هرتز بود در حالی که محتوای فرکانسی سیگنال EMG میتواند تا ۱۰۰۰ هرتز نیز باشد. این باعث کاهش عملکرد ویژگی ها میشود و به خصوص بهره ویژگی های فرکانسی کاهش می یابد. برای رفع آن میتوان از ثبت هایی با نرخ نمونه برداری بیشتر استفاده کرد (معمولاً تا ماکزیمم 2kHz برای سیگنال EMG کفایت میکند)

روش ثبت سیگنال EMG سطحی، برابندی از فعالیت چندین فیبر عضلانی را ثبت میکند و این باعث میشود تفاوت فعالیت ها حین انجام حرکات مختلف، کمتر دیده شود. با استفاده از الکترودهای سوزنی تفاوت های فعالیت فیبرهای عضلانی بهتر ثبت شده و دقت طبقه بندی بالاتر خواهد رفت.

میتوان ویژگی هایی با قدرت تفکیک کمتر را حذف کرد و تنها به ویژگی های مهم و تعیین کننده اجازه ورود به فرایند آموزش را داد. میزان قدرت تفکیک و خوب یا بد بودن یک ویژگی به روشهای مختلف قابل بررسی است که یکی از آنها معیار فیشر است که در صورت وجود تنها دو لیبل، به روش زیر حساب میشود:

$$J = \frac{|S_b|}{|S_w|} = \frac{|\mu_0 - \mu_1|^2 + |\mu_0 - \mu_2|^2}{\sigma_1^2 + \sigma_2^2}$$

که در آن  $\mu_0$  میانگین مقادیر ویژگی مورد نظر روی کل داده ها،  $\mu_1$  و  $\mu_2$  همین میانگین روی داده های لیبل ۱ و ۲ هستند، و  $\sigma_1$  و  $\sigma_2$  واریانس این ویژگی ها هستند. هرچه مقدار این معیار بیشتر باشد، ویژگی مد نظر قدرت تفکیک بالاتری دارد و مناسب تر است.

البته ویژگی های دیگری نیز وجود دارند که قدرت تفکیک یک دسته ویژگی را (بجای یک تک ویژگی) حساب میکنند و میتوانند مناسبتر باشند.