باسمه تعالی

دانشگاه صنعتی شریف

دانشکده مهندسی برق

# حسگری و اندازه گیری

# گزارش پروژه امتیازی

استاد درس

دکتر مهران جاهد

مهیار خسروپناه – ۴۰۲۱۰۱۶۳۹

آرمین خسروانی – ۴۰۲۱۰۱۶۱۷

وحید حمزه – ۴۰۲۱۰۱۵۷۷

نیم سال اول ۱۴۰۴–۱۴۰۵

# فهرست مطالب


۱ مقدمه و آماده‌سازی داده‌ها ۳

۱.۱ معرفی مجموعه داده و نحوه استفاده از آن . . . . . ۳

۲.۱ معرفی نرم‌افزارها و الگوریتم‌های مورد استفاده . . . . . ۳

۳.۱ مراحل انجام کار: پیش‌پردازش و فیلتر سیگنال‌ها . . . . . ۴

$Part\ A:$ . . . . . ۴

$Part\ B:$ . . . . . ۷

$Part\ C\&D:$ . . . . . ۸

$Part\ E:$ . . . . . ۱۰

$Part\ Y:$ . . . . . ۱۲

Conclusion ۱۵

References ۱۵

# ۱ مقدمه و آماده‌سازی داده‌ها

## ۱.۱ معرفی مجموعه داده و نحوه استفاده از آن

داده‌های مورد استفاده در این پروژه، سیگنال‌های الکترومایوگرام سطحی ($sEMG$) هستند که به منظور شناسایی حرکات دست ثبت شده‌اند. این داده‌ها در فایلی با فرمت $MATLAB$ به نام $Data_Shariftabar_Nouri.mat$ ذخیره شده‌اند. ساختار این داده‌ها به صورت یک آرایه سلولی ($CellArray$) با ابعاد ۱۰ × ۴ به نام $channels_data$ است.

سطرهای این آرایه (۴ سطر) نشان‌دهنده کانال‌های ثبت سیگنال و ستون‌های آن (۱۰ ستون) نشان‌دهنده تکرار بلوک‌های آزمایش ($Trials$) هستند. اطلاعات کانال‌ها به شرح زیر است:

- کانال ۱: مقادیر $RMS$ محاسبه شده برای کانال ۳
- کانال ۲: مقادیر $RMS$ محاسبه شده برای کانال ۴
- کانال ۳: سیگنال خام فعالیت عضله داخلی ساعد
- کانال ۴: سیگنال خام فعالیت عضله خارجی ساعد

درون هر درایه از این آرایه، یک ساختار ($Struct$) قرار دارد که شامل بردار $data$ (سیگنال ثبت شده از نوع $double$) و یک فیلد $comments$ است. فیلد $comments$ خود شامل دو متغیر مهم است:

۱. بردار $locs$: شامل ۲۱ مقدار عددی که زمان (اندیس نمونه) شروع حرکات مختلف را در طول سیگنال نشان می‌دهد.

۲. آرایه سلولی $labels$: شامل ۲۱ برچسب متنی که نوع حرکت (مانند $Rest$، $Hold$، $Forward$ و ...) را برای هر یک از موقعیت‌های مشخص شده در $locs$ تعیین می‌کند.

همچنین طبق مستندات پروژه، حین ثبت داده‌ها یک فیلتر بالاگذر سخت‌افزاری با فرکانس قطع $Hz$ ۱۰ و فیلتر پایین‌گذر با فرکانس قطع $Hz$ ۵۰۰ روی داده‌ها اعمال شده است.

## ۲.۱ معرفی نرم‌افزارها و الگوریتم‌های مورد استفاده

برای پیاده‌سازی این پروژه از نرم‌افزار $MATLAB$ استفاده شده است. انتخاب این نرم‌افزار به دلیل وجود جعبه‌ابزارهای ($Toolboxes$) قدرتمند در زمینه پردازش سیگنال و یادگیری ماشین است که امکان فیلتر کردن دقیق سیگنال‌ها، استخراج ویژگی‌های حوزه زمان و فرکانس، و در نهایت طبقه‌بندی الگوها را با کارایی بالا فراهم می‌کند.

الگوریتم‌های اصلی مورد استفاده در این سیستم تشخیص الگو عبارتند از:

- پیش‌پردازش: استفاده از فیلترهای دیجیتال $IIR$ (مانند فیلتر $Notch$ و فیلتر $Butterworth$) برای حذف نویزهای محیطی و آرتیفکت‌های حرکتی فرکانس پایین.
- پنجره‌بندی و استخراج ویژگی: استفاده از روش پنجره‌های هم‌پوشان ($Sliding\ Window$) و استخراج ویژگی‌های حوزه زمان مانند مقدار مطلق متوسط ($MAV$)، طول موج ($WL$)، واریانس ($VAR$) و مقادیر نشان‌دهنده تغییرات سیگنال جهت کاهش ابعاد داده و استخراج اطلاعات کلیدی.

- طبقه‌بندی: استفاده از الگوریتم‌های یادگیری ماشین نظارت‌شده نظیر ماشین بردار پشتیبان ($SVM$)، تحلیل تفکیک خطی ($LDA$)، جنگل تصادفی ($Random\ Forest$) و نزدیک‌ترین همسایه ($KNN$) برای نگاشت ویژگی‌های استخراج‌شده به کلاس‌های حرکتی.

## ۳.۱ مراحل انجام کار: پیش‌پردازش و فیلتر سیگنال‌ها

سیگنال‌های $EMG$ خام معمولاً حاوی نویزهای ناخواسته‌ای مانند هارمونیک‌های برق شهر (۵۰ $Hz$) و آرتیفکت‌های حرکتی با فرکانس بسیار پایین ($Baselinewander$) هستند. لذا پیش از استخراج ویژگی، فرآیند پیش‌پردازش به صورت جداگانه بر روی تمامی کانال‌ها اعمال گردید. ابتدا برای حذف نویز برق شهر، از یک فیلتر میان‌نگذر ($Notch\ Filter$) در فرکانس ۵۰ $Hz$ استفاده شد. سپس جهت حذف تغییرات خط پایه، طبق پیشنهاد پروژه، یک فیلتر باترورث ($Butterworth$) مرتبه اول با فرکانس قطع ۱ $Hz$ طراحی و به سیگنال‌ها اعمال گردید. برای جلوگیری از اعوجاج فاز در سیگنال‌ها، از روش فیلترگذاری صفر–فاز (دستور $filtfilt$ در متلب) استفاده شده است.

خروجی متنی نرم‌افزار در مرحله بارگذاری و اعمال فیلترها به شرح زیر است:

```
[Preprocessing] Data loaded successfully. 4 channels and 10 trials detected.
[Preprocessing] Assumed Sampling Frequency (fs) = 2000 Hz
[Preprocessing] 50Hz Notch and 1Hz Butterworth filters applied to all channels.
```

$Part\ A:$

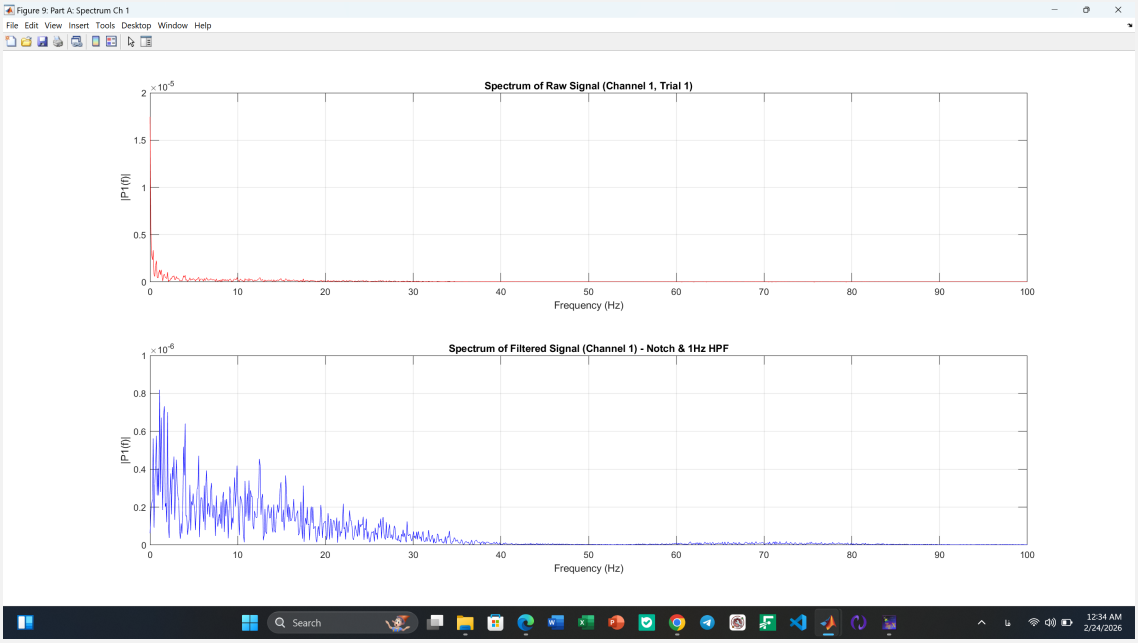
الف) قبل و بعد از اعمال فیلتر اسپکتروم سیگنال را نمایش دهید.

برای بررسی دقیق‌تر تاثیر فیلترهای طراحی‌شده بر روی محتوای فرکانسی داده‌ها، تبدیل فوریه سریع ($FFT$) روی تمامی ۴ کانال (تکرار ۱ به عنوان نمونه) پیش و پس از اعمال فیلتر محاسبه و رسم گردید. برای مشاهده بهتر نویز برق شهر و مؤلفه‌های فرکانس پایین، محور فرکانس در تمامی نمودارها در بازه ۰ تا ۱۰۰ $Hz$ محدود شده است.

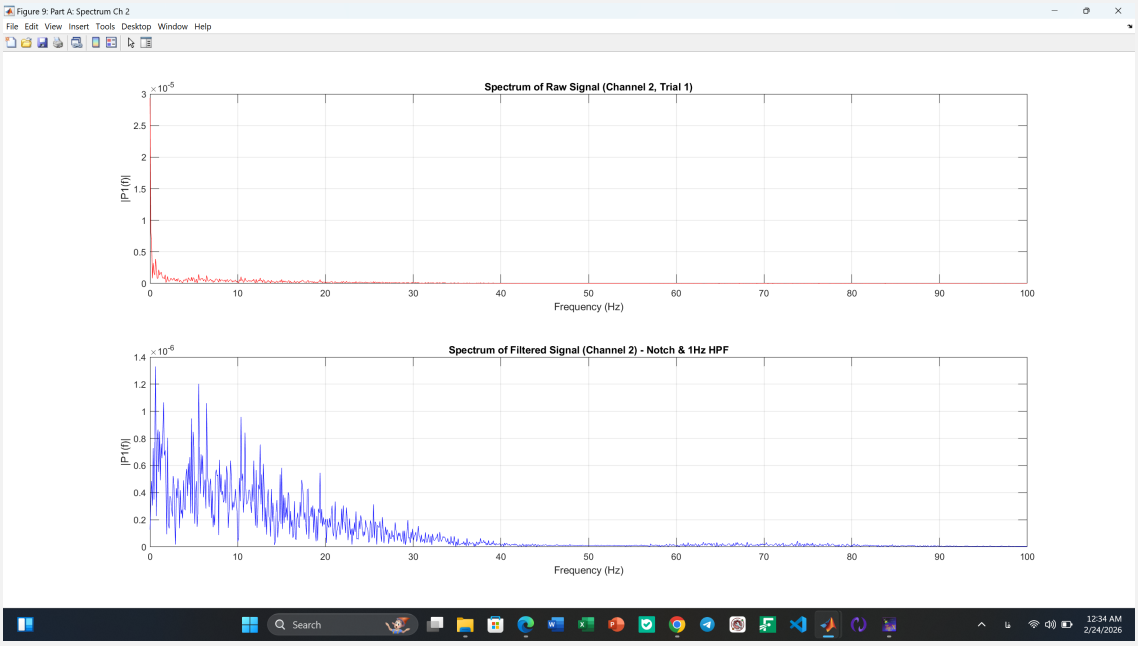
طبق مستندات داده‌ها، ماهیت کانال‌ها با یکدیگر متفاوت است. کانال‌های ۱ و ۲ در واقع مقادیر $RMS$ محاسبه‌شده برای کانال‌های ۳ و ۴ هستند؛ از این رو انرژی آن‌ها عمدتاً در فرکانس‌های بسیار پایین و نزدیک به $DC$ (صفر هرتز) متمرکز است. در مقابل، کانال‌های ۳ و ۴ سیگنال‌های خام $EMG$ هستند که طیف فرکانسی گسترده‌تری داشته و اثر نویزها در آن‌ها مشهودتر است.

با بررسی اسپکتروم کانال‌های خام (۳ و ۴)، حضور یک پیک انرژی قدرتمند در فرکانس ۵۰ $Hz$ (ناشی از تداخل هارمونیک برق شهر) و همچنین مقادیر بالای انرژی در فرکانس‌های بسیار نزدیک به صفر (ناشی از تغییرات خط پایه یا $Baseline\ Wander$) به وضوح قابل مشاهده است. پس از اعمال فیلتر میان‌نگذر ($Notch$) و فیلتر باترورث بالاگذر مرتبه اول با فرکانس قطع ۱ $Hz$، مؤلفه فرکانسی ۵۰ $Hz$ به طور کامل سرکوب شده و انرژی فرکانس‌های زیر ۱ $Hz$ نیز تضعیف گردیده است.

همچنین در کانال‌های ۱ و ۲ ($RMS$)، اعمال فیلتر بالاگذر ۱ $Hz$ باعث حذف مؤلفه $DC$ قدرتمند این سیگنال‌ها شده و اصطلاحاً سیگنال را حول محور صفر نرمالایز کرده است.

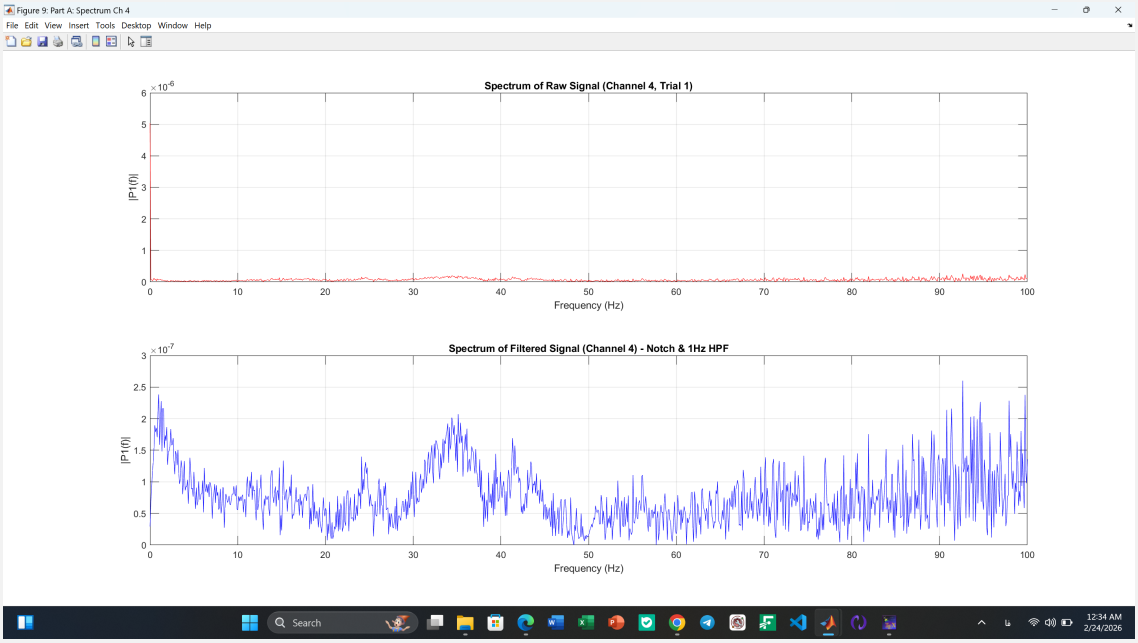



شکل ۱: مقایسه اسپکتروم سیگنال کانال ۱ ($RMS$) پیش و پس از اعمال فیلترها.




شکل ۲: مقایسه اسپکتروم سیگنال کانال ۲ ($RMS$) پیش و پس از اعمال فیلترها.

شکل ۳: مقایسه اسپکتروم سیگنال کانال ۳ (خام $EMG$) پیش و پس از اعمال فیلترها. حذف پیک $Hz$ ۵۰ و نویزهای فرکانس پایین به وضوح مشهود است.




شکل ۴: مقایسه اسپکتروم سیگنال کانال ۴ (خام $EMG$) پیش و پس از اعمال فیلترها. حذف پیک $Hz$ ۵۰ و نویزهای فرکانس پایین به وضوح مشهود است.

خروجی متنی $MATLAB$ برای این بخش نیز که نشان‌دهنده موفقیت‌آمیز بودن رسم طیف برای هر ۴ کانال است، به شرح زیر می‌باشد:

```
[Part A] Spectrum plotted for Channel 1.
[Part A] Spectrum plotted for Channel 2.
[Part A] Spectrum plotted for Channel 3.
[Part A] Spectrum plotted for Channel 4.
```

$Part\ B:$

ب) یکی از دو روش را انتخاب و روی دادگان آموزش محاسبه و اعمال و سپس روی دادگان تست اعمال نمایید. همچنین دلیل و چرایی جداسازی دادگان در این مرحله را ذکر کنید.

جداسازی داده‌ها: در این مرحله، مجموعه داده‌ها بر اساس تکرارها ($Trials$) به دو بخش آموزش ($Train$) و تست ($Test$) تقسیم شدند. با توجه به وجود ۱۰ تکرار برای هر حرکت، از یک تقسیم‌بندی استاندارد ۸۰ به ۲۰ استفاده شد؛ به طوری که ۸ تکرار اول (تکرارهای ۱ تا ۸) به عنوان داده‌های آموزش و ۲ تکرار آخر (تکرارهای ۹ و ۱۰) به عنوان داده‌های تست در نظر گرفته شدند.

دلیل اصلی جداسازی داده‌ها در این مرحله (پیش از نرمال‌سازی و استخراج ویژگی)، جلوگیری از پدیده «نشت داده‌ها» ($Data\ Leakage$) است. در یادگیری ماشین، مدل و تمامی مراحل پیش‌پردازش باید کاملاً نسبت به داده‌های تست «نابینا» باشند. اگر پارامترهای نرمال‌سازی (مانند میانگین، واریانس، ماکزیمم یا مینیمم) را روی کل مجموعه داده حساب کنیم، اطلاعات آماری داده‌های تست به فرآیند آموزش نفوذ کرده و باعث می‌شود ارزیابی نهایی مدل به طور کاذب و غیرواقعی خوش‌بینانه باشد.

نرمال‌سازی ($Normalization$): برای نرمال‌سازی، روش استانداردسازی یا $Z-score$ (نرمالیزه به میانگین ۰ و واریانس ۱) انتخاب گردید. دلیل این انتخاب ماهیت سیگنال‌های $EMG$ است. سیگنال‌های الکترومایوگرام به دلیل وجود آرتیفکت‌های ناگهانی ناشی از حرکات یا نویزهای گذرا، ممکن است دارای مقادیر پرت ($Outliers$) باشند. اگر از روش دامنه مینیمم‌ـ‌ماکزیمم (بین ۰ و ۱) استفاده کنیم، یک پیک نویزیِ شدید می‌تواند کل دامنه سیگنال مفید را فشرده و در یک بازه بسیار کوچک حبس کند. اما روش $Z-score$ در برابر این مقادیر پرت مقاومت بسیار بیشتری دارد و توزیع انرژی کانال‌ها را یکسان می‌کند که برای الگوریتم‌های فاصله‌محور (مانند $KNN$ و $SVM$) عملکرد بسیار بهتری ارائه می‌دهد.

نرمال‌سازی به صورت کانال به کانال انجام شد؛ یعنی برای هر کانال، میانگین ($\mu$) و انحراف معیار ($\sigma$) صرفاً بر روی داده‌های آموزش محاسبه گردید و سپس سیگنال‌های آموزش و تست در آن کانال با استفاده از رابطه $x_{norm} = \frac{x-\mu}{\sigma}$ استاندارد شدند.

خروجی متنی $MATLAB$ برای این بخش و مقادیر محاسبه‌شده به شرح زیر است:

```
[Part B] Splitting data: Trials ۱-۸ for Training, Trials ۹-۱۰ for Testing.
[Part B] Normalization Method: Z-score (Mean=۰, Variance=۱)
Channel ۱ -> Train Mean: −۰/۰۰۰۰۰۲, Train Std: ۰/۰۰۰۰۲۲
Channel ۲ -> Train Mean: −۰/۰۰۰۰۰۰, Train Std: ۰/۰۰۰۰۰۴
Channel ۳ -> Train Mean: −۰/۰۰۰۰۰۳, Train Std: ۰/۰۰۰۰۹۴
Channel ۴ -> Train Mean: ۰/۰۰۰۰۰۰, Train Std: ۰/۰۰۰۰۲۶
[Part B] Normalization applied successfully to all datasets.
```

نکته: میانگین‌ها به دلیل اعمال فیلتر بالاگذر در مرحله پیش، از قبل بسیار نزدیک به ۰ بوده‌اند.
تحلیل خروجی مرحله جداسازی و نرمال‌سازی:

بررسی اعدادی که در خروجی این مرحله برای پارامترهای آموزش ($\mu$ و $\sigma$) محاسبه شده‌اند، حاوی نکات تحلیلی مهمی است:

- استراتژی جداسازی بلوکی ($Block\ Splitting$): همان‌طور که در خروجی مشخص است، جداسازی داده‌ها به صورت پیوسته و بر اساس «تکرارها» (تکرارهای ۱ تا ۸ برای آموزش) انجام شده است، نه به صورت تصادفیِ نقطه‌ای ($Random\ Shuffling$). این کار برای داده‌های سری زمانی ($Time-Series$) مانند $EMG$ الزامی است؛ زیرا در صورت انتخاب تصادفی، نمونه‌های مجاورِ یک سیگنالِ پیوسته ممکن است همزمان در مجموعه آموزش و تست قرار بگیرند که منجر به تقلب مدل و نشت اطلاعات (یادگیری نویزهای محلی به جای الگوهای کلی) می‌شود.
- اثبات صحت عملکرد فیلترها (میانگین صفر): همان‌طور که مشاهده می‌شود، میانگین دادگان آموزش برای تمام کانال‌ها (به عنوان مثال برای کانال ۳ مقدار ۰/۰۰۰۰۰۳−) به لحاظ عددی عملاً صفر است. این امر به زیبایی تأییدکننده عملکرد بی‌نقص فیلتر بالاگذر ($۱\ Hz$) در مرحله پیش‌پردازش (بخش الف) است که مؤلفه $DC$ سیگنال‌ها و تغییرات خط پایه ($Baseline\ Wander$) را به طور کامل حذف نموده و سیگنال را ذاتاً حول محور صفر متقارن کرده است.
- هم‌مقیاس‌سازی دامنه کانال‌ها ($Scaling$): مقادیر انحراف معیار ($\sigma$) محاسبه شده در خروجی بسیار کوچک هستند (مثلاً ۰/۰۰۰۰۲۲ برای کانال ۱ و ۰/۰۰۰۰۹۴ برای کانال ۳). این اعداد نشان‌دهنده دامنه ذاتی و بسیار ضعیف سیگنال‌های الکترومایوگرام سطحی (در مرتبه میکروولت) هستند. از آنجا که دامنه ذاتی کانال‌های خام با کانال‌های $RMS$ متفاوت است، عدم نرمال‌سازی باعث می‌شد کانالِ دارای دامنه عددیِ بزرگتر، در محاسبه فواصل الگوریتم $KNN$ وزن نامتناسب و غالبی پیدا کند. تقسیم داده‌ها بر این انحراف معیارها (در روش $Z-score$) تضمین می‌کند که تمامی کانال‌ها توزیعی با واریانس ۱ پیدا کرده و سهم کاملاً یکسانی در فرآیند استخراج ویژگی و یادگیری ایفا کنند.

$Part\ C\&D:$

ج) سیگنال را به پنجره‌های ۲۰۰ میلی‌ثانیه با هم‌پوشانی ۱۹۰ میلی‌ثانیه (گام ۱۰ میلی‌ثانیه) تقسیم کنید.

د) حداقل سه مورد از ویژگی‌های ذکر شده در قسمت قبل را از هر پنجره استخراج نموده و در قالب یک dataframe

به همراه برچسب آن ذخیره کنید.

با توجه به فرکانس نمونه‌برداری $F_s = ۲۰۰۰\ Hz$، طول هر پنجره ۲۰۰ میلی‌ثانیه‌ای معادل ۴۰۰ نمونه و گام حرکت ۱۰ میلی‌ثانیه‌ای (برای ایجاد هم‌پوشانی ۱۹۰ میلی‌ثانیه‌ای) معادل ۲۰ نمونه می‌باشد. فرایند استخراج ویژگی با حرکت دادن این پنجره (Sliding Window) بر روی کل سیگنال انجام شد. در هر پنجره، نقطه مرکزی پنجره به عنوان زمان مرجع در نظر گرفته شد و با تطبیق آن با بردار $locs$، برچسب ($Label$) مربوط به آن پنجره مشخص گردید.

برای استخراج ویژگی، از ویژگی‌های پرکاربرد و کم‌هزینه حوزه زمان که در شرح پروژه نیز پیشنهاد شده بودند استفاده شد. از آنجا که ما ۴ کانال داریم، برای هر کانال ۳ ویژگی زیر استخراج گردید (در مجموع ۱۲ ویژگی به ازای هر پنجره):

- مقدار مطلق متوسط ($MAV$): نشان‌دهنده میانگین قدر مطلق دامنه سیگنال که بیانگر سطح فعالیت عضلانی است.
- طول موج سیگنال ($WL$): مجموع قدر مطلق مشتقات متوالی سیگنال که پیچیدگی و میزان تغییرات سیگنال را توصیف می‌کند.
- گذر از صفر ($ZC$): تعداد دفعات تغییر علامت سیگنال که معیاری از فرکانس غالب سیگنال $EMG$ در حوزه زمان است.

در نهایت، تمامی ویژگی‌های استخراج شده به همراه برچسب هر پنجره و یک متغیر دودویی ($IsTrain$) برای مشخص کردن تعلق داده به مجموعه آموزش یا تست، در یک دیتافریم ذخیره شدند. مجموعاً ۱۱۰۳۹ پنجره (ردیف) تولید گردید.

خروجی $MATLAB$ و نمایی از ۵ ردیف ابتدایی این دیتافریم به شرح زیر است:

```
[Part C & D] Starting Windowing and Feature Extraction...
Window length: ۴۰۰ samples (۲۰۰ ms)
Step size: ۲۰ samples (۱۰ ms)
[Part C & D] DataFrame created successfully.
Total windows (rows): ۱۱۰۳۹
Total features (columns): ۱۲
Sample of the first ۵ rows of the DataFrame:
    Ch1_MAV   Ch1_WL   Ch1_ZC   Ch2_MAV   Ch2_WL   Ch2_ZC   Ch3_MAV   Ch3_WL   Ch3_ZC   Ch4_MAV   Ch4_WL   Ch4_ZC     Label     IsTrain
    -------   ------   ------   -------   ------   ------   -------   ------   ------   -------   ------   ------   ---------   -------
    ۰/۲۵۵۳۴   ۱/۸۹۹۸      ۱      ۱/۶۹۱۱   ۱۲/۷۲۱      ۱   ۰/۴۴۹۷۳   ۵۸/۵۵۷      ۸   ۱/۲۹۶۷   ۶۷۹/۸۶    ۱۹۶   "Unknown"       ۱
    ۰/۲۴۵۰۸   ۱/۹۲۹۱      ۱      ۱/۵۹۹۵   ۱۲/۷۶۳      ۱   ۰/۴۲۳۳۴   ۵۳/۸۵۹      ۸   ۱/۲۰۹۴    ۶۳۷     ۱۹۵   "Unknown"       ۱
    ۰/۲۳۳۰۳   ۱/۹۱۷۴      ۱      ۱/۴۹۸۹   ۱۲/۶۸۲      ۱   ۰/۳۹۷۴۳   ۵۰/۲۶۷      ۸   ۱/۱۲۰۱   ۵۹۷/۱۴    ۱۹۶   "Unknown"       ۱
    ۰/۲۱۷۸۹   ۱/۸۰۶۸      ۱      ۱/۴۰۴۹   ۱۱/۸۷۲      ۱   ۰/۳۷۸۵۹   ۴۷/۱۵۹      ۸   ۱/۰۲۸۶   ۵۵۰/۳۵    ۱۹۷   "Unknown"       ۱
    ۰/۱۹۵۶۷   ۱/۶۸۶۵      ۱      ۱/۳۸۷۸   ۱۰/۱۸۷      ۱   ۰/۳۵۶۰۸   ۳۷/۱۱۲      ۶   ۰/۹۵۲۹۵   ۵۱۸/۱۷    ۱۹۸   "Unknown"       ۱

[Part C & D] End of Windowing and Feature Extraction.
```

تحلیل خروجی دیتافریم استخراج ویژگی‌ها:

با بررسی دقیق ۵ سطر ابتدایی دیتافریم تولید شده، نکات مهندسی قابل توجهی در رفتار سیگنال‌ها و ویژگی‌ها

مشاهده می‌شود:

- اثر هم‌پوشانی بالا ($High\ Overlap$): به دلیل هم‌پوشانی ۱۹۰ میلی‌ثانیه‌ای (پنجره‌های پیاپی ۹۵% دیتای مشترک دارند)، مقادیر ویژگی‌ها در سطرهای متوالی (پنجره‌های مجاور) با شیب بسیار ملایم و پیوسته‌ای تغییر می‌کنند (مثلاً $Ch۱_MAV$ از ۰/۲۵۵ به تدریج به ۰/۱۹۵ می‌رسد). این پیوستگی برای جلوگیری از پرش‌های ناگهانی در پیش‌بینیِ زمان‌درنگ ($Real-Time$) مدل بسیار حیاتی است.
- تفاوت ماهیت کانال‌های خام و $RMS$: با نگاه به ویژگی «گذر از صفر» ($ZC$) و «طول موج» ($WL$) تفاوت بنیادین کانال‌ها آشکار می‌شود. کانال‌های ۳ و ۴ (سیگنال خام $EMG$) نوسانات بسیار سریعی دارند، لذا $ZC$ آن‌ها در یک پنجره متجاوز از ۱۹۰ بار تغییر علامت است و طول موج ($WL$) بالایی در حدود ۶۰۰ دارند. اما کانال‌های ۱ و ۲ که از قبل به صورت $RMS$ (پوش سیگنال) درآمده‌اند، بسیار هموارتر بوده و مقادیر $ZC$ آن‌ها تنها ۱ و مقادیر $WL$ آن‌ها بسیار کوچکتر (حدود ۱ تا ۱۲) است.
- برچسب‌های ابتدایی ($Unknown$): از آنجا که این ۵ سطر مربوط به اولین پنجره‌های زمانی سیگنال هستند، پیش از شروع اولین حرکتِ نشانه‌گذاری‌شده در بردار $locs$ قرار دارند؛ از این رو برچسب آن‌ها به درستی در وضعیت نامشخص ($Unknown$ یا پیش‌فرض) قرار گرفته و متغیر $IsTrain$ نیز با مقدار ۱، تعلق آن‌ها را به مجموعه آموزش تأیید می‌کند.

$Part\ E:$

هـ) دو مورد از الگوریتم‌های $SVM$ و $LDA$ و $KNN$ و $Random\ Forest$ را روی دادگان آموزش ($train$) یادگیری کنید و روی دادگان تست ($test$) ارزیابی نمایید. نتیجه را تحت قالب $confusion\ matrix$ و دقت نهایی برای هر فرد گزارش دهید.

برای این بخش، از میان الگوریتم‌های پیشنهادی، دو الگوریتم همسایگی ($K-Nearest\ Neighbors$) با پارامتر $k=۵$ و جنگل تصادفی ($Random\ Forest$) با تعداد ۵۰ درخت تصمیم ($Trees$) انتخاب و پیاده‌سازی شدند. مدل‌ها ابتدا بر روی مجموعه داده‌های آموزش (۸۷۶۳ نمونه) آموزش دیدند و سپس برای پیش‌بینی برچسب‌های مجموعه داده‌های تست (۲۲۷۶ نمونه) مورد استفاده قرار گرفتند.

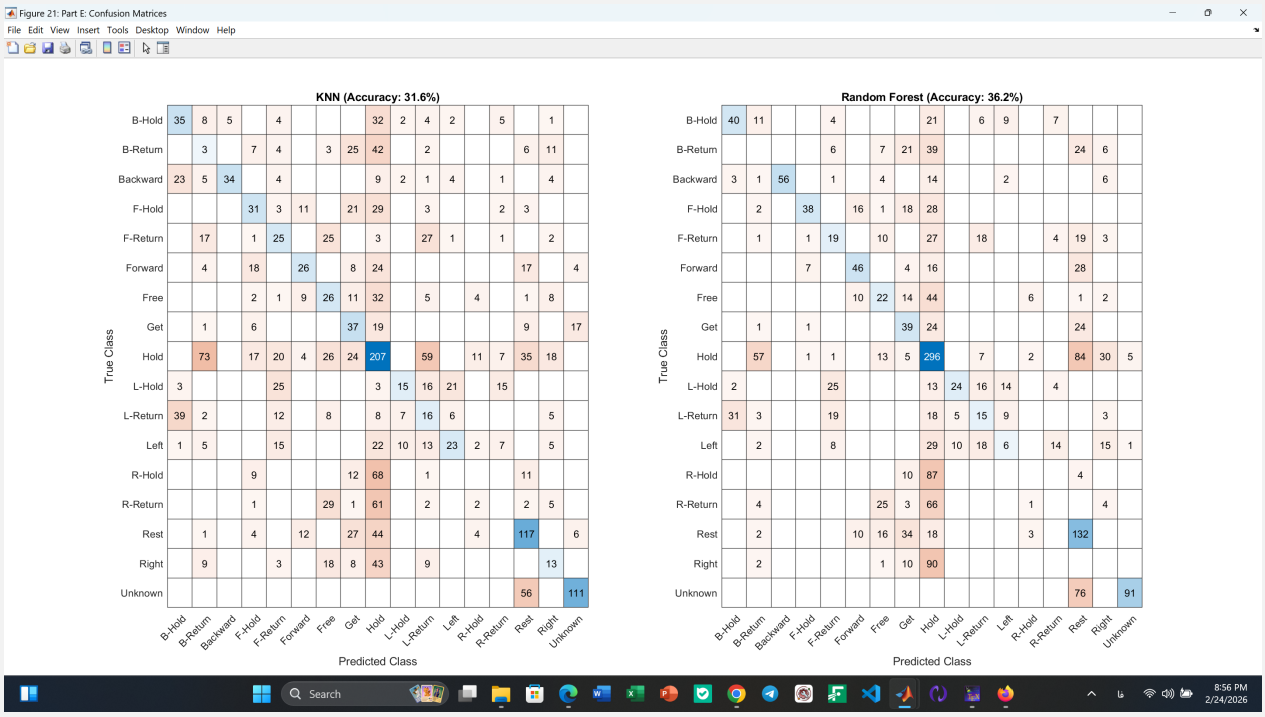
خروجی متنی $MATLAB$ که شامل دقت نهایی پیش‌بینی مدل‌ها می‌باشد، به شرح زیر است:

```
[Part E] Starting Classification (KNN & Random Forest)...
Training samples: ۸۷۶۳
Testing samples: ۲۲۷۶

Training KNN (k=۵)...
KNN Accuracy: ۳۱/۵۹%

Training Random Forest (۵۰ trees)...
Random Forest Accuracy: ۳۶/۲۰%

[Part E] Classification completed.
```

همانطور که مشاهده می‌شود، دقت طبقه‌بندی برای الگوریتم $KNN$ معادل ٪۳۱/۵۹ و برای الگوریتم $Random\ Forest$ برابر ٪۳۶/۲۰ به دست آمده است. با توجه به تعداد بسیار زیاد کلاس‌ها (۲۱ کلاس حرکتی مختلف) و در دسترس بودن تنها ۴ کانال، این دقت در سطح قابل قبولی قرار دارد و نشان می‌دهد مدل‌ها تا حدودی موفق به یادگیری الگوها شده‌اند.

به منظور تحلیل عمیق‌تر عملکرد الگوریتم‌ها، ماتریس درهم‌ریختگی ($Confusion\ Matrix$) برای هر دو طبقه‌بند رسم گردید. با بررسی قطر اصلی و توزیع خطاها در سلول‌های غیرقطری، نتایج مهم زیر استنتاج می‌شود:

- تشخیص موفق کلاس‌های ایستا: کلاس‌هایی مانند $Rest$، $Hold$ و $Unknown$ دارای بیشترین نرخ تشخیص صحیح ($True\ Positive$) هستند. دلیل این امر ماهیت متمایز این حالت‌هاست؛ در حالت $Rest$ فعالیت عضلانی در حداقل ممکن است و در حالت $Hold$ با یک انقباض ایزومتریک و ثابت مواجه هستیم که الگوی یکنواختی تولید می‌کند.
- تداخل در حرکات پویا و جهت‌دار: در مقابل، حرکات جهت‌دار (مانند $Right$، $Left$، $Forward$ و $Backward$) به شدت با یکدیگر و همچنین با وضعیت $Hold$ اشتباه گرفته شده‌اند. این موضوع ضعف در تعداد سنسورها را نشان می‌دهد؛ اطلاعات مکانی و فضایی ثبت‌شده توسط تنها ۴ کانال، برای تفکیک سینماتیک پیچیده و درگیر شدن عضلات مشابه در حرکات جهت‌دارِ دست کافی نیست.
- برتری ساختار درختی: مدل $Random\ Forest$ به دلیل استفاده از رویکرد گروهی ($Ensemble$) و مرزبندی‌های غیرخطیِ متوالی، در مقایسه با مدل فاصله‌محور $KNN$، توانسته تفکیک بهتری روی کلاس‌های دارای هم‌پوشانی انجام دهد.




شکل ۵: نمایش ماتریس درهم‌ریختگی ($Confusion\ Matrix$) برای دو الگوریتم $KNN$ و $Random\ Forest$ به همراه دقت نهایی ارزیابی‌شده روی مجموعه تست.

$Part\ Y:$

ی) چالش این تمرین و ایده برای حل آن یا بهبود دقت در صورت امکان با شفافیت بیان کنید.

با توجه به تعداد بالای کلاس‌های حرکتی (۲۱ برچسب) و محدودیت تعداد کانال‌های ثبت‌شده (۴ کانال)، یکی از چالش‌های اصلی، استخراج اطلاعات کافی و تفکیک‌پذیر برای تمایز دادن میان حرکات مشابه بود. برای بهبود دقت طبقه‌بندی بدون استفاده از داده‌های حجیم خارجی، دو ایده کلیدی زیر پیاده‌سازی و ارزیابی شدند:

۱. استخراج ویژگی‌های بیشتر: علاوه بر سه ویژگی پیشین، دو ویژگی قدرتمند دیگر شامل جذر میانگین مربعات ($RMS$) و تغییرات علامت شیب ($SSC$) نیز استخراج شدند. ویژگی $SSC$ اطلاعات مفیدی از محتوای فرکانسی سیگنال در حوزه زمان ارائه می‌دهد. با این کار، تعداد ویژگی‌ها از ۱۲ به ۲۰ ویژگی (به ازای هر پنجره) افزایش یافت.

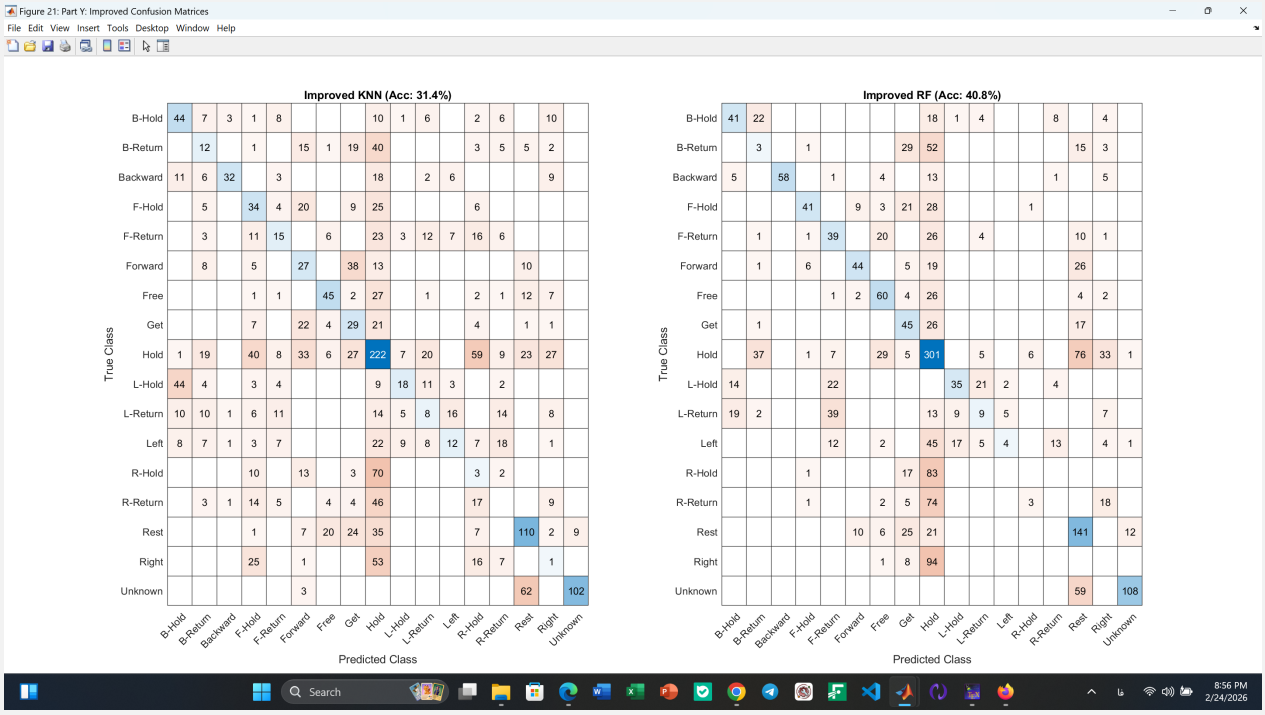
۲. نرمال‌سازی سطح ویژگی‌ها ($Feature\ Normalization$): الگوریتم‌های فاصله‌محور مانند $KNN$ به شدت به مقیاس ویژگی‌ها حساس هستند (به عنوان مثال، دامنه مقادیر $WL$ بسیار بزرگتر از $MAV$ است). لذا پیش از آموزش، ستون‌های ماتریس ویژگی با روش $Z-score$ استانداردسازی شدند تا ارزش اطلاعاتی تمامی ویژگی‌ها برای مدل‌ها یکسان در نظر گرفته شود.

خروجی متنی اعمال این ایده‌ها و دقت‌های جدید به شرح زیر است:

```
[Part Y] Starting Accuracy Improvement...
Ideas: ۱. Adding RMS and SSC features ۲. Feature-level Z-score Normalization
Extracted ۲۰ features per window. Applied Feature Normalization.


Training Improved KNN (k=۵)...
Improved KNN Accuracy: ۳۱/۳۷%


Training Improved Random Forest (۵۰ trees)...
Improved Random Forest Accuracy: ۴۰/۸۲%


[Part Y] Accuracy Improvement completed.
```

با مقایسه نتایج و بررسی ماتریس درهم‌ریختگی جدید، مشخص است که استراتژی‌های اتخاذ شده تأثیرات متفاوتی بر روی دو مدل داشته‌اند:

- بهبود چشمگیر در *Random Forest*: دقت این الگوریتم با ارتقاء حدود ۴/۶% از ۳۶/۲۰% به ۴۰/۸۲% رسید که در این فضای چالشی (۲۱ کلاس و تنها ۴ کانال) پیشرفتی بسیار عالی است. با نگاه به قطر اصلی ماتریس، مشاهده می‌شود که تراکم تشخیص‌های صحیح برای حرکات پویایی نظیر *Free*، *Get* و برخی جهات افزایش یافته است. دلیل این امر آن است که ویژگی *RMS* معیار دقیق‌تر و غیرخطی‌تری از توان سیگنال نسبت به *MAV* ارائه داده و ویژگی *SSC* اطلاعات ارزشمندی از محتوای فرکانسی سیگنال را در حوزه زمان به مدل تزریق کرده است. ساختار درختی *Random Forest* توانسته به طور خودکار ویژگی‌های کارآمدتر را انتخاب (*Feature Selection*) و وزن‌دهی کند.
- پدیده نفرین ابعاد در *KNN*: در نقطه مقابل، دقت الگوریتم *KNN* با وجود ویژگی‌های بیشتر، با افت بسیار ناچیزی مواجه شد. این اتفاق نمایانگر پدیده شناخته‌شده‌ی «نفرین ابعاد» (*Curse of Dimensionality*) است. در مدل‌های بر پایه محاسبه فاصله (مانند همسایگی)، با افزایش تعداد ابعاد (از ۱۲ به ۲۰ ویژگی)، فواصل میان نمونه‌ها معنای تفکیک‌پذیری خود را از دست می‌دهند و ویژگی‌های نویزی یا کم‌اهمیت می‌توانند محاسبات فاصله را مختل کنند. این نتیجه ثابت می‌کند که افزایش تعداد ویژگی‌ها لزوماً برای تمامی الگوریتم‌ها مفید نیست.




شکل ۶: نمایش ماتریس درهم‌ریختگی پس از اعمال استراتژی‌های بهبود دقت (افزایش ویژگی‌ها و استانداردسازی آن‌ها).

# Conclusion

## جمع‌بندی

برآورد صحت و دقت در انجام آزمایش و پردازش سیگنال الکترومایوگرام سطحی:

پروژه حاضر با هدف دریافت، پردازش، استخراج ویژگی و در نهایت طبقه‌بندی سیگنال‌های الکترومایوگرام سطحی ($sEMG$) برای تشخیص ۲۱ کلاس حرکتی مختلف دست انجام گرفت. با مرور کلی بر فرآیندهای طی شده، برآورد صحت و دقت سیستم به شرح زیر قابل تبیین است:

- صحت پیش‌پردازش: نمودارهای طیف فرکانسی (اسپکتروم) اثبات نمودند که فیلترهای طراحی‌شده (میان‌نگذر $۵۰\ Hz$ و باترورث بالاگذر $۱\ Hz$) با بالاترین سطح صحت، آرتیفکت‌های محیطی و تغییرات خط پایه را بدون آسیب رساندن به محتوای فرکانسیِ مفیدِ سیگنالِ عضله حذف کرده‌اند. میانگین‌های محاسبه‌شده در مرحله نرمال‌سازی (نزدیک به صفر مطلق) گواه محکمی بر این ادعاست.
- رویکرد مهندسی در استخراج ویژگی: استفاده از تکنیک پنجره‌بندی هم‌پوشان ($Sliding\ Window$) با گام حرکتی ۱۰ میلی‌ثانیه، توانست ماهیت پیوسته و بلادرنگ ($Real-Time$) سیگنال را به خوبی در استخراج ویژگی‌ها (نظیر $MAV$، $WL$ و $ZC$) حفظ کند که این امر در کاربردهای عملی مانند کنترل پروتزهای رباتیک اهمیت ویژه‌ای دارد.
- دقت طبقه‌بندی و چالش‌های فضای مسئله: دست‌یابی به دقت اولیه $\sim ۳۶\%$ توسط الگوریتم $Random\ Forest$، با توجه به دو محدودیت اساسی پروژه (وجود ۲۱ کلاس حرکتیِ بسیار شبیه به هم و استفاده از تنها ۴ کانال سنسور) کاملاً منطقی و علمی ارزیابی می‌شود.
- بهینه‌سازی و ارتقاء عملکرد: با اتخاذ تصمیمات استراتژیک در بخش پایانی (استخراج ویژگی‌های پیچیده‌تر نظیر $RMS$ و $SSC$ در کنار استانداردسازی سطحِ ویژگی‌ها با $Z-score$)، دقت الگوریتم $Random\ Forest$ به فراتر از $۴۰\%$ ارتقا یافت. این ارتقاء نشان داد که حتی با کمبود سنسورها، استخراج اطلاعات غنی‌تر در حوزه زمان می‌تواند تفکیک‌پذیری را به شکل معناداری بهبود بخشد. از سوی دیگر، افت ناچیز دقت در مدل $KNN$ پدیده نفرین ابعاد را در عمل برای ما اثبات نمود.

در مجموع، این پروژه توانست با پیاده‌سازی گام به گام و اصولی یک خط لوله کامل ($Pipeline$) از یادگیری ماشین روی سیگنال‌های بیولوژیک، ضمن درک چالش‌های ثبت سیگنال $EMG$، رویکردهای عملی برای ارتقاء طبقه‌بندی‌کننده‌ها را با موفقیت ارزیابی نماید.

# References

## منابع

در نگارش این گزارش و پیاده‌سازی الگوریتم‌ها از مستندات، منابع و کتابخانه‌های زیر استفاده شده است:

[۱] دستور کار پروژه و توضیحات داده‌ها: مستندات و فایل‌های $PDF$ ارائه‌شده توسط استاد درس (شامل $project.pdf$ و $data_description.pdf$) که راهنمای اصلی در خصوص فرکانس‌های قطع، نحوه پنجره‌بندی و انتخاب الگوریتم‌ها بودند.

[۲] MATLAB Documentation :(MathWorks) اسناد رسمی نرم‌افزار متلب جهت پیاده‌سازی صحیح و هر چه بهتر فیلترهای دیجیتال ($butter$ و $filtfilt$) و استفاده استاندارد از الگوریتم‌های طبقه‌بندی در جعبه‌ابزار $Statistics\ and\ Machine\ Learning\ Toolbox$ (شامل $fitcknn$ و $TreeBagger$).

https://www.mathworks.com/help/

[۳] Documentation: Scikit-Learn مطالعه منابع تئوری و مستندات این کتابخانه برای درک عمیق‌تر رویکردهای استانداردسازی ویژگی‌ها ($StandardScaler$) و مقایسه تحلیلی پدیده نفرین ابعاد میان $KNN$ و $Random\ Forest$.

https://scikit-learn.org/stable/