



دانشگاه صنعتی شریف

دانشکده مهندسی برق

پایان‌نامه‌ی کارشناسی ارشد

گرایش بیوالکتریک

تشخیص پتانسیل‌های وابسته به رخداد با استفاده از تجزیه‌ی تانسوری

نگارنده

مینا جمشیدی ایدجی

استاد راهنما

دکتر محمدباقر شمس‌اللهی

۱۳۹۵ مرداد

الله
بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

این پروژه به وسیلهٔ جایزهٔ پژوهشیاری موفقیان حمایت
شده است.

بسمه تعالی

**دانشگاه صنعتی شریف
دانشکده مهندسی برق**

پایان نامه کارشناسی ارشد

عنوان: تشخیص پتانسیل های وابسته به رخداد با استفاده از تجزیه‌ی تانسوری
نگارش: «مینا جمشیدی ایدجی»

اعضا هیات داوران:

- | | |
|-----------------|---------------------------------------|
|
امضاء: | استاد راهنما: دکتر محمد باقر شمس‌الهی |
|
امضاء: | استاد داور داخلی: دکتر سپیده حاجی‌پور |
|
امضاء: | استاد داور خارجی: دکتر فرناز قاسمی |

تاریخ: مرداد ۱۳۹۵.

قدردانی

از استاد گرانقدر، جناب آقای دکتر شمس‌الهی، صمیمانه قدردانی می‌کنم. در این دو سال بسیار ازیشان آموختم و از حمایت‌هایشان بھرمند گشتم. هرچند ناتوانم از جیران زحماتشان، اما همواره تلاش خواهم کرد تا سرمایه‌ای را که از رفتار اخلاق‌مدار ایشان آموختم، همراه خویش داشته باشم. همچنین، از سرکار خانم دکتر حاجی‌پور و سرکار خانم دکتر قاسمی سپاس‌گزارم که زحمت داوری این پایان‌نامه را قبول نمودند و با راهنمایی‌هایشان باعث بهبود آن شدند.

و سپاس‌گزارم از مادر و پدرم، پشتونه‌های حقیقی زندگانی‌ام، که تمام هستی‌ام را مديون بزرگواری بی‌کرانشان هستم. هردویشان، اولین و تاثیرگذارترین آموزگاران زندگی‌ام بوده‌اند و هستند. ستودن روح بزرگوارشان در قالب کلام، بیهوده سخن راندن است و بس؛ چرا که زبان خامه ندارد توان مدح وجود نازنینشان ... تنها به قول حضرت حافظ بسنه می‌کنم که غرض ز مسجد و میخانه‌ام وصال شماست، جز این خیال ندارم خدا گواه من است.

چکیده:

استفاده از EEG در مطالعه‌ی پتانسیل‌های وابسته به رخداد (ERP) بسیار مرسوم است. ها پتانسیل‌هایی هستند که به واسطه‌ی فعالیت نورون‌های مغز در پاسخ به یک تحریک خارجی در EEG ظهر می‌کنند و دامنه‌ی بسیار کوچکی در مقایسه با سیگنال پس‌زمینه‌ی EEG دارند. به همین دلیل به طور مرسوم با استفاده از میانگین‌گیری سنکرون استخراج الگوی سیگنال ERP صورت می‌گیرد. اما در این روش اطلاعات موجود در الگوی هر تک‌ثبت از دست می‌رود.

روش‌های دو بعدی ابزارهای ارزشمندی در کاربردهای تشخیص الگو و جداسازی کور منابع (BSS) هستند. برخی از این روش‌ها نیز در استخراج اطلاعات وابسته به رخداد از سیگنال EEG استفاده شده‌اند. با این وجود این روش‌ها به دلیل ذات دو بعدی‌شان تنها می‌توانند بر دو بعد زمان و مکان اعمال شوند. این در حالی است که در بسیاری از کاربردها، سیگنال ثبت‌شده در چندین بعد دارای اطلاعات است. تانسورها و روش‌های تجزیه‌ی تانسوری می‌توانند ابزار قدرتمندی برای حفظ و استفاده از اطلاعات ابعاد مختلف سیگنال مورد مطالعه باشند. به همین دلیل این روش‌ها در پردازش داده‌های EEG بسیار مورد استقبال قرار گرفته‌اند.

مطالعات انجام شده روی کاربرد روش‌های تجزیه‌ی تانسوری در پردازش سیگنال EEG حاوی ERP را می‌توان به دو دسته‌ی کلی تقسیم کرد: گروه اول از تجزیه‌ی تانسوری برای جداسازی منابع در داده‌های حاوی ERP و استخراج الگوی ERP استفاده می‌کنند و گروه دوم تجزیه‌ی تانسوری را برای کاهش ویژگی به کار بسته‌اند.

تجزیه‌ی Tucker به دلیل انعطاف‌پذیری بالایش، به طور گسترده در روش‌های کاهش ویژگی تانسوری مورد استفاده قرار گرفته است. یکی از محبوب‌ترین این روش‌ها، که تعمیم تانسوری LDA است، با نام HODA معرفی شده است.

در این پایان‌نامه یک روش جدید کاهش ویژگی مبتنی بر تجزیه‌ی تانسوری معرفی شده است که آن را Higher Order Spectral Regression Discriminat Analysis، یا به اختصار HOSRDA، نامیده‌ایم. این روش تعمیم تانسوری روش SRDA است. برای ارزیابی الگوریتم پیشنهادی، از آن برای تشخیص کاراکتر یک P300 speller مربوط به مجموعه‌دادگان BCI Competition III-Dataset II و ۵ تکرار برای هر سطر/ستون استفاده شود، به ترتیب دقت تشخیص کاراکتر ۱۰، ۱۵، ۸۹، ۹۶/۵ و ۷۲/۹ به دست آمده است. HOSRDA کاملاً قابلیت رقابت با سایر روش‌هایی که بر این

مجموعه‌دادگان اعمال شده‌اند را دارد و تقریباً از تمامی آن‌ها پیشی گرفته است.

هم‌چنین با توجه به اهمیت طبقه‌بندی بدون سرپرست در P300 speller، فرم پیشنهادی تانسوری آن را معرفی کردیم و به بررسی دلایل عدم موفقیت تانسوری کردن این روش پرداختیم.

کلمات کلیدی:

پتانسیل وابسته به رخداد - تجزیه‌ی تانسوری - تجزیه‌ی Tucker - HODA -HOSRDA

فهرست مطالب

۱	پیش‌گفتار	
۱	۱-۱ انگیزه‌ی پژوهش	
۳	۲-۱ اهداف پژوهش	
۴	۳-۱ ساختار پایان‌نامه	
۵	۲ مفاهیم پایه	
۵	۱-۲ پیش‌گفتار	
۶	۲-۲ پتانسیل وابسته به رخداد [۱]	
۸	۳-۲ سیستم رابط مغز-رایانه	
۹	۴-۲ پارادایم P300 Speller	
۱۱	۵-۲ جمع‌بندی	
۱۳	۳ پیش‌زمینه‌ی نظری	
۱۳	۱-۳ پیش‌گفتار	
۱۳	۲-۳ نمادگذاری	
۱۴	۳-۳ روش‌های برداری کاهش ویژگی	
۱۴	۱-۳-۳ LDA	

۱۵	SRDA	۲-۳-۳
۱۸	جبر تانسوری	۴-۳
۱۸	تانسورها و عملیات میان آنها	۱-۴-۳
۱۸	تعریف اولیه	۱-۱-۴-۳
۲۲	انواع ضرب تانسوری	۲-۱-۴-۳
۲۴	ضرب‌های ماتریسی مهم در جبر تانسوری	۳-۱-۴-۳
۲۵	رتبه‌ی تانسور	۴-۱-۴-۳
۲۶	خواص مهم	۵-۱-۴-۳
۲۷	تجزیه‌ی Tucker	۲-۴-۳
۳۰	تخمین ماتریس‌های عامل و تانسور هسته	۱-۲-۴-۳
۳۲	تجزیه‌ی CP	۳-۴-۳
۳۴	تخمین ماتریس‌های عامل	۱-۳-۴-۳
۳۵	روش‌های تانسوری کاهش ویژگی	۵-۳
۳۵	فرمول‌بندی کلی روش‌های کاهش ویژگی تانسوری	۱-۵-۳
۳۷	کاهش ویژگی با استفاده از تجزیه‌ی Tucker	۱-۱-۵-۳
۳۹	کاهش ویژگی با استفاده از تجزیه‌ی CP	۲-۱-۵-۳
۳۹	نمادگذاری	۲-۵-۳
۴۰	HODA	۳-۵-۳
۴۲	TLLD	۴-۵-۳
۴۴	جعبه‌ابزارهای تانسوری	۶-۳
۴۶	جمع‌بندی	۷-۳

۴ پیشینه‌ی پژوهش

۴۷	پیش‌گفتار	۱-۴
۴۷	روش‌های برداری در تشخیص پتانسیل وابسته به رخداد در پارادایم P300 Speller	۲-۴
۴۸	SVM	۱-۲-۴
۴۸	eSVM	۲-۲-۴
۵۰	SWLDA	۳-۲-۴
۵۱	EFLD	۴-۲-۴
۵۲	CNN	۵-۲-۴
۵۲	کاربرد تجزیه‌ی تانسوری در پردازش داده‌های حاوی پتانسیل وابسته به رخداد	۳-۴
۵۲	کاربرد تجزیه‌ی CP در پردازش داده‌های حاوی پتانسیل وابسته به رخداد	۱-۳-۴

۵۳	۱-۱-۳-۴	جداسازی و مکان‌یابی منبع پتانسیل وابسته به رخداد
۶۰	۲-۱-۳-۴	کاهش ویژگی
۶۵	۲-۳-۴	کاربرد تجزیه‌ی Tucker در پردازش داده‌های حاوی پتانسیل وابسته به رخداد
۶۵	۱-۲-۳-۴	کاربرد تجزیه‌ی Tucker بدون سرپرست
۶۶	۲-۲-۳-۴	کاربرد تجزیه‌ی Tucker با سرپرست
۶۸	۴-۴	جمع‌بندی

۵ روش پیشنهادی: HOSRDA

۶۹	۱-۵	پیش‌گفتار
۷۰	۲-۵	فرمول‌بندی ریاضی و الگوریتمی روش پیشنهادی
۷۶	۳-۵	پیچیدگی محاسباتی
۷۷	۴-۵	جمع‌بندی

۶ نتایج شبیه‌سازی HOSRDA

۷۹	۱-۶	پیش‌گفتار
۷۹	۲-۶	داده‌ها
۷۹	۱-۲-۶	مجموعه دادگان استفاده شده
۸۱	۲-۲-۶	پیش‌پردازش
۸۲	۳-۶	معیارهای ارزیابی
۸۲	۱-۳-۶	آزمون آماری
۸۲	۲-۳-۶	تشخیص کاراکتر
۸۳	۴-۶	تجزیه‌ی Tucker بدون سرپرست
۸۳	۵-۶	انتخاب پارامتر و همگرایی HOSRDA
۸۳	۱-۵-۶	انتخاب تعداد ویژگی‌ها
۸۵	۲-۵-۶	همگرایی و پایداری الگوریتم
۸۸	۶-۶	ارزیابی HOSRDA
۸۸	۱-۶-۶	بررسی تمایزپذیری دو کلاس با استفاده از آزمون آماری
۸۹	۲-۶-۶	کارایی HOSRDA در تشخیص کاراکتر

۹۳ ماتریس پایه‌ی متناظر با بُعد کانال در زیرفضای HOSRDA ۷-۶

۹۴ جمع‌بندی ۸-۶

۷ طبقه‌بندِ بدون سرپرست و فُرمِ پیشنهادی تansوری آن

۹۵ پیش‌گفتار ۱-۷

۹۵ طبقه‌بند بدون سرپرست برای P300 speller ۲-۷

۹۶ BLDA ۱-۲-۷

۹۷ طبقه‌بند بدون سرپرست ۲-۲-۷

۹۹ فُرمِ پیشنهادی تansوری ۳-۷

۱۰۱ شبیه‌سازی ۴-۷

۱۰۳ جمع‌بندی ۵-۷

۸ جمع‌بندی، نتیجه‌گیری و پیشنهادها

۱۰۴ جمع‌بندی و نتیجه‌گیری ۱-۸

۱۰۵ پیشنهادها ۲-۸

مراجع

واژه نامه فارسی به انگلیسی

فهرست جداول

۷۶	۱-۵	عملیات محاسباتی HOSDA
۸۹	۱-۶	مقادیر p-value مربوط به t-test و F-test بر روی ویژگی‌های سوژه‌ی B
۲-۶	دقت تشخیص کاراکتر برای روش‌های مختلف برای دو سوژه‌ی A و B زمانی که هر تحریک R بار تکرار شده است. خطوط فاصله نشانه‌ی این است که مرجع مربوطه نتایجی را در آن قسمت گزارش نکرده است.	
۹۲

فهرست اشکال

۷	مراحل یک از آزمایش ERP [۱]	۱-۲
۹	یک نمونه از صفحه‌ای که در P300-BCI به کاربر نمایش داده می‌شود [۲]	۲-۲
	(الف) صفحه‌ی P300 speller. (ب) نمایش شماتیک شیوه‌ی ارائه‌ی تحریک‌ها به کاربر [۱]	۳-۲
۱۰		
۱۲	بلوک‌دیاگرام یک P300 speller	۴-۲
۱۹	تансور مرتبه سه [۳]	۱-۳
۱۹	فیبرهای یک تانسور مرتبه سه [۳]	۲-۳
۲۰	اسلایس‌های یک تانسور مرتبه سه [۳]	۳-۳
۲۰	تانسور یکانی [۳]	۴-۳
۲۱	تانسوری کردن مُد- n به روش [۴]	۵-۳
۲۱	تانسوری کردن مُد- n به روش [۵]	۶-۳
۲۲	تانسوری کردن مُد- n به روش [۶]	۷-۳
۲۶	تانسور مرتبه سه‌ی رتبه-یک [۳]	۸-۳
۲۸	تجزیه‌ی Tucker بر روی تانسور $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ [۷]	۹-۳
۲۹	تجزیه‌های Tucker-1,2,3 یک تانسور مرتبه سه	۱۰-۳
۳۴	تجزیه‌ی CP تانسور مرتبه سه با R مولفه [۷]	۱۱-۳
۳۸	بلوک‌دیاگرام کاهش ویژگی و طبقه‌بندی با استفاده از تجزیه‌ی Tucker	۱۲-۳
۵۳	نمایشی از یک تانسور با ابعاد $Channel \times Frequency \times Time$	۱-۴
	فاکتورهای تجزیه‌ی PARAFAC یک مولفه‌ای تانسور سه بعدی ITPC مربوط به یک سوزه. این مولفه نمایانگر فعالیت گامای پس‌سری با فرکانس حدود ۴۰ هرتز در حوالی ۱۰۰ میلی‌ثانیه پس از تحریک است [۸]	۲-۴
۵۴	فاکتورهای تجزیه‌ی PARAFAC تانسور پنج بعدی ITPC که با دو مولفه برازش شده است. همان‌طور که مشاهده می‌شود مولفه‌ی اول در هر دو شرایط آزمایش شدت تقریباً یکسانی دارد. اما مولفه‌ی دوم در تحریک دوم دارای شدت بسیار بیشتری است [۸]	۳-۴
۵۵		
۵۶	نمایش شماتیک تفاوت CP و SCP [۹]	۴-۴
۵۷	مولفه‌های تجزیه‌ی CP و SCP تانسور سه بعدی کانال×زمان×تکرار در مقاله‌ی [۹]	۵-۴

۵۷	پتانسیل برانگیخته‌ی بازسازی شده با استفاده از SCP و CP در [۹]	۶-۴
۵۹	توابع هزینه‌ی مربوط به تجزیه‌ی CP و دو تجزیه‌ی GCP و WCP	۷-۴
۶۰	استخراج ERP با استفاده از تجزیه‌ی WCP و WCP+EKS	۸-۴
۶۱	ویژگی‌های چند حوزه همراه با پروفایل‌های زمانی-فرکانسی و مکانی آن‌ها [۱۰]	۹-۴
۶۳	تحریک‌های آزمایش مربوط به [۱۱]	۱۰-۴
۶۴		۱۱-۴
۸۰	صفحه‌ی P300 speller در پروتکل Donchin [۱۲]	۱-۶
۸۱	محل قرارگیری الکترودها در سیستم [۱۲] ۱۰-۱۰	۲-۶
۸۴	عملکرد تجزیه‌ی Tucker بدون سرپرست در تشخیص کاراکتر	۳-۶
۸۴	انتخاب پارامتر در HOSRDA	۴-۶
۸۶	بررسی پایداری الگوی همگرایی الگوریتم HOSRDA با شرط توقف پیشنهادی	۵-۶
۸۷	بررسی پایداری الگوی همگرایی الگوریتم HOSRDA با شرط توقف پیشنهادی	۶-۶
۸۹	توزیع آماری دو کلاس ویژگی شماره دو از داده‌های آموزش سوژه‌ی B	۷-۶
۹۱	نمودار «دقت تشخیص کاراکتر بر حسب تعداد کاراکتر آموزش» برای HOSRDA و SRDA	۸-۶
۹۳	ستون‌های ماتریس عامل اول (منتظر با بعد کanal) از زیرفضای HOSRDA و نتایج انتخاب کanal در [۱۳]	۹-۶
۹۶	دو کلاس P300 و فاقد P300 پس از تصویر شدن بر یک بعد در BLDA	۱-۷

فهرست کلمات اختصاری

BCI	Brain Computer Interface
ERP	Event Related Potential
HO DA	Higher Order Discriminant Analysis
HOSRDA	Higher Order Spectral Regression Discriminant Analysis
LDA	Linear Discriminant Analysis
SRDA	Spectral Regression Discriminant Analysis
STDA	Spatial-Temporal Discriminant Analysis

فصل ۱

پیش‌گفتار

۱-۱ انجیزه‌ی پژوهش

مغز انسان دارای پیچیدگی مثال‌زدنی است و از سال‌ها پیش انسان مشتاق کشف رمز و رازهای عملکرد مغز بوده است. EEG^۱ یکی از روش‌هایی است که به صورت گستردۀ برای ثبت فعالیت مغز مورد استفاده قرار می‌گیرد. روش‌های دیگری همچون PET، fMRI و MEG نیز می‌توانند فعالیت مغز را برای انسان روشن‌تر سازند. با این حال EEG به دلیل کم‌هزینه بودن، عدم نیاز به دستگاه‌های پیچیده و پرحجم و نیز ثبت غیرتھاجمی^۲ مورد استقبال آزمایشگاهی و کلینیکی قرار گرفته است.

استفاده از EEG در مطالعه‌ی پتانسیل‌های وابسته به رخداد^۳ (ERP) بسیار مرسوم است. ERP‌ها پتانسیل‌هایی هستند که به واسطه‌ی فعالیت نورون‌های مغز در پاسخ به یک تحریک خارجی در EEG ظهرور می‌کنند. کاربرد EEG در دستیابی به اطلاعاتی که ERP‌ها می‌توانند از چگونگی پاسخ مغز به رخدادهای اطرافش داشته باشند، جایگاه خود را در کاربردهای کلینیکی، توانبخشی بیماران و سیستم‌های رابط مغز-رایانه^۴ (BCI) یافته است.

سیستم‌های رابط مغز-رایانه به انسان کمک می‌کنند تا بدون استفاده از اندام‌های حرکتی خود کارهای مورد نظرش را انجام دهد. از همین روی تشخیص و تحلیل پتانسیل‌های وابسته به رخداد در این سیستم‌ها می‌تواند برای کمک و توانبخشی به افرادی که دچار مشکلات حرکتی هستند کاربرد داشته

^۱Electroencephalogram

^۲Non-invasive

^۳Event Related Potential

^۴Brain-Computer Interface

باشد. همچنین با استفاده از پتانسیل‌های وابسته به رخداد می‌توان به تشخیص و طبقه‌بندی بیماران دارای مشکلات ذهنی (نظیر مشکل تمرکز، مشکلات خواندن و ...) پرداخت.

اما آن‌چه مطالعه‌ی ERP‌ها را چالش برانگیز می‌کند دامنه‌ی بسیار کوچک آن‌ها در مقایسه با سیگنال پس‌زمینه‌ی EEG است. به همین دلیل به طور مرسوم استخراج الگوی ERP با فرض صفر بودن میانگین آماری سیگنال پس‌زمینه، با استفاده از میانگین‌گیری سنکرون صورت می‌گیرد. این روش می‌تواند شباهت‌های پتانسیل‌های برانگیخته شده در تحریک‌های مختلف را به خوبی آشکار کند؛ اما در مقابل اطلاعات موجود در الگوی هر تک‌ثبت^۵ از دست می‌رود. نکته‌ی قابل توجه درباره‌ی روش میانگین‌گیری این است که با این فرض انجام می‌شود که مغز در طول زمان آزمایش پاسخ یکسانی به اتفاقات بیرونی می‌دهد. این در حالی است که عواملی که باعث از بین رفتن تمرکز کاربر می‌شوند و همچنین خستگی و عادت کردن، تا حدی کیفیت و الگوی بروز ERP را تحت تأثیر قرار می‌دهند. از طرفی استخراج اطلاعات هر تک‌ثبت در کاربردهای سیستم‌های BCI دارای اهمیت خاصی است.

روش‌های دو بُعدی (نظیر SVD و PCA و NMF) ابزارهای ارزشمندی در کاربردهای تشخیص الگو و جداسازی کور منابع^۶ (BSS) هستند. برخی از این روش‌ها نیز در استخراج اطلاعات وابسته به رخداد از سیگنال EEG استفاده شده‌اند. با این وجود این روش‌ها به دلیل ذات دو بُعدی‌شان تنها می‌توانند بر دو بُعد زمان و مکان اعمال شوند. این در حالی است که در بسیاری از کاربردها، سیگنال ثبت‌شده در چندین بُعد دارای اطلاعات است. به طور خاص، در آزمایش‌های ERP سیگنال ثبت‌شده می‌تواند (بسته به نوع آزمایش) اطلاعاتی از تک‌ثبت‌ها^۷، شرایط آزمایش^۸، سوزه‌ها^۹، زمان، مکان و فرکانس داشته باشد. در صورت تمایل به استفاده‌ی همزمان از این اطلاعات در روش‌های ماتریسی، ناچار به قرار دادن اطلاعات ابعاد مختلف در یک بعد از ماتریس خواهیم بود و این کار تحلیل نتایج را پیچیده و در مواردی غیرممکن می‌سازد. همچنین در صورت تمایل به پردازش جداگانه‌ی این ابعاد (برای مثال پردازش جداگانه‌ی اطلاعات سوزه‌ها یا تکرارها) اطلاعات مشترکی که احتمالاً بین آنها وجود دارد از دست می‌رود. تانسورها^{۱۰} که نمایشی از داده‌های چندبعدی هستند، می‌توانند ابزار قدرتمندی برای حفظ و استفاده از اطلاعات ابعاد

^۵Single Trial

^۶Blind Source Separation

^۷Trial

^۸Task Condition

^۹Subjects

^{۱۰}Tensor

مختلف سیگنال مورد مطالعه باشند. روش‌های تجزیه‌ی تانسوری^{۱۱} می‌توانند اطلاعات داده‌های تانسور را به گونه‌ای در ابعادش تجزیه نمایند که ارتباط بین اطلاعات ابعاد مختلف مورد استفاده قرار گیرد [۶]. به همین دلیل این روش‌ها در پردازش داده‌های EEG (تشخیص الگو، BSS، استفاده در سیستم‌های رابط مغز-رایانه، تشخیص مکان وقوع یک فعالیت در مغز و پاسخ به سوالات تحقیقاتی و کلینیکی علوم اعصاب) بسیار مورد استقبال قرار گرفته‌اند.

۲-۱ اهداف پژوهش

استفاده از روش‌های تجزیه‌ی تانسوری در کاربردهای مختلفی از پردازش داده‌های EEG مشاهده می‌شود؛ برای مثال تشخیص فعالیت در یک باند فرکانسی خاص، مکان‌یابی وقوع صرع و مطالعه‌ی ERP‌ها. در این پژوهش بر کاربرد روش‌های تجزیه‌ی تانسوری در پردازش داده‌های EEG حاوی ERP تمرکز می‌کنیم.

پژوهش‌هایی که از روش‌های تجزیه‌ی تانسوری در پردازش و تحلیل داده‌های حاوی ERP استفاده کرده‌اند به دو گروه تقسیم می‌شوند: هدف گروه اول جداسازی منابع داده‌ی EEG حاوی ERP و مکان‌یابی محل وقوع آن‌ها در مغز است که کاربرد آن در مطالعات علوم شناختی و علوم اعصاب است. گروه دوم، از تجزیه‌های تانسوری برای کاهش ویژگی با هدف بهبود فرآیند طبقه‌بندی استفاده کرده‌اند که برای آن‌ها کاربردهایی در سیستم‌های BCI و مطالعات علوم شناختی گزارش شده است. در این پژوهش علاوه بر آن که ادبیات موجود برای گروه اول مرور و کارایی برخی بررسی شده است، تمرکز اصلی بر گروه دوم مقالات است.

استفاده از روش‌های کاهش ویژگی تانسوری از این جهت دارای جذابیت است که در بسیاری موارد بهتر از روش‌هایی که داده‌ها را به عنوان بردار دریافت می‌کنند، عمل می‌کنند. برای مثال، در شبیه‌سازی‌های انجام شده به این نتیجه رسیدیم که تعمیم تانسوری روش LDA^{۱۲}، با نام HODA^{۱۳}، نه تنها در یافتن فضایی که تفکیک‌پذیری بهتری دارد از LDA موفق‌تر است، بلکه در مواجهه با مسائلی که تعداد داده‌های آموزش کم است (مسئله‌ی SSS) بسیار توانانter است^{۱۴}. در این راستا الگوریتمی جدید

^{۱۱}Tensor Decomposition

^{۱۲}Linear Discriminant Analysis

^{۱۳}Higher Order Discriminant Analysis

^{۱۴}در این‌باره در فصول آینده مفصل‌تر صحبت خواهیم کرد.

مبتنی بر تجزیه‌ی تانسوری برای کاهش ویژگی، با نام HOSRDA، پیشنهاد شده است که کارایی آن در طبقه‌بندی دادگان یک P300 speller مورد ارزیابی قرار گرفته است.

همچنین با توجه به اهمیت طبقه‌بندی بدون سرپرست در P300 speller امکان بهبود این روش با بهره بردن از تجزیه‌ی تانسوری بررسی شده است.

۳-۱ ساختار پایان‌نامه

در فصل دوم این نوشتار مروری سریع بر برخی مفاهیم پایه خواهیم داشت. در فصل سوم به مرور پیش‌زمینه‌ی نظری پژوهش انجام شده می‌پردازیم. پیش‌زمینه‌ی نظری شامل تمامی ابزارها و روش‌های ریاضی است که در ادامه‌ی پایان‌نامه به آن‌ها رجوع خواهیم کرد. سعی بر آن بوده است که محتوای این فصل طوری انتخاب شود که نیاز به منبع دیگری برای آشنایی با مفاهیم و روش‌های پیش‌زمینه نباشد. فصل چهار به مرور پیشینه‌ی پژوهش اختصاص دارد و در آن به مرور روش‌های برداری استفاده شده برای طبقه‌بندی در P300 speller و کاربرد روش‌های تانسوری در مطالعه‌ی ERP‌ها پرداخته شده است. در فصل پنج و شش یک الگوریتم کاهش ویژگی جدید مبتنی بر تجزیه‌ی تانسوری پیشنهاد و کارایی آن بر روی دادگان یک P300 speller ارزیابی شده است. در فصل هفتم طبقه‌بند بدون سرپرست برای P300 speller توضیح داده شده و تعمیم تانسوری آن پیشنهاد و نتایج شبیه‌سازی دو روش مقایسه و تحلیل شده است. در نهایت در فصل آخر به نتیجه‌گیری و ارائه‌ی پیشنهادات برای پژوهش‌های آینده پرداخته‌ایم. همچنین در صفحات آخر پایان‌نامه یک واژه‌نامه‌ی فارسی به انگلیسی قرار داده شده است که در آن معادل انگلیسی کلمات فارسی‌ای که به دفعات در متن استفاده شده‌اند، آورده شده است.

فصل ۲

مفاهیم پایه

۱-۲ پیش‌گفتار

در سال ۱۹۲۹ برگر^۱ برای اولین بار فعالیت نورون‌های مغز انسان را به وسیله‌ی الکترودهایی که بر سطح جمجمه قرار داده بود ثبت کرد. این فعالیت الکتریکی EEG نام دارد. حدود یک دهه پس از آن، EEG مورد استقبال پژوهشگران علوم اعصاب قرار گرفت و در کاربردهای کلینیکی و پژوهشی جایگاه خود را یافت.

EEG اندازه‌گیری بسیار کلی از فعالیت‌های مغز است و این نقطه ضعف بزرگ آن محسوب می‌شود. در واقع EEG نمایشی از ترکیب صدھا منبع فعالیت عصبی در مغز است. به همین دلیل جداسازی و تشخیص یک فعالیت خاص در آن مشکل است. اما در EEG پاسخ‌های الکتریکی‌ای نهفته است که با یک میانگین‌گیری ساده قابل شناسایی هستند. این فعالیت‌های الکتریکی پتانسیل وابسته به رخداد یا به اختصار ERP نام دارند. ریشه‌ی این نام در این است که حضور این پتانسیل‌ها در EEG وابسته به یک رخداد خاص است.

در اواسط دهه‌ی ۱۹۸۰ مطالعه‌ی ERP‌ها در میان پژوهشگران مورد استقبال قرار گرفته بود که با ظهور PET و fMRI بیم آن می‌رفت که این دو جایگزینی برای EEG و ERP برای شناخت کارکرد مغز شوند. اما رزولوشن زمانی و قابل حمل بودن دستگاه ثبت EEG باعث شد که استفاده از آن در جایگاه خود باقی بماند [۱۴].

امروزه نیز استفاده از ERP‌ها در حوزه‌های گسترده‌ی پژوهشی و کلینیکی کاربرد دارد. سیستم‌های

^۱Berger

رابط مغز-رایانه^۱، یا به اختصار BCI، از جمله مواردی هستند که ERP‌ها در آن‌ها کاربرد مهمی دارند. در ادامه‌ی این فصل ابتدا به بررسی اجمالی پتانسیل‌های وابسته به رخداد (به خصوص P300) و سیستم‌های می‌پردازیم. سپس یکی از پارادایم‌های پرکاربرد آن (P300 speller) را معرفی می‌کنیم.

۲-۲ پتانسیل وابسته به رخداد [۱]

پتانسیل‌های وابسته به رخداد فعالیت‌های الکتریکی با دامنه‌ی بسیار کم هستند که حدود چندصد میلی‌ثانیه بعد از یک تحريك در EEG ظاهر می‌شوند. برای بررسی ERP‌ها در EEG، اصولاً سیگنال زمانی قطعه‌بندی^۲ می‌شود. این قطعه‌بندی بسته به کاربرد از چندصد میلی‌ثانیه قبل از تحريك تا حدود یک ثانیه بعد از آن انجام می‌شود. به این قطعه‌ها epoch یا trial گفته می‌شود. در این نوشتار به جای این دو کلمه از واژه‌ی «تکثیت» استفاده می‌کنیم.

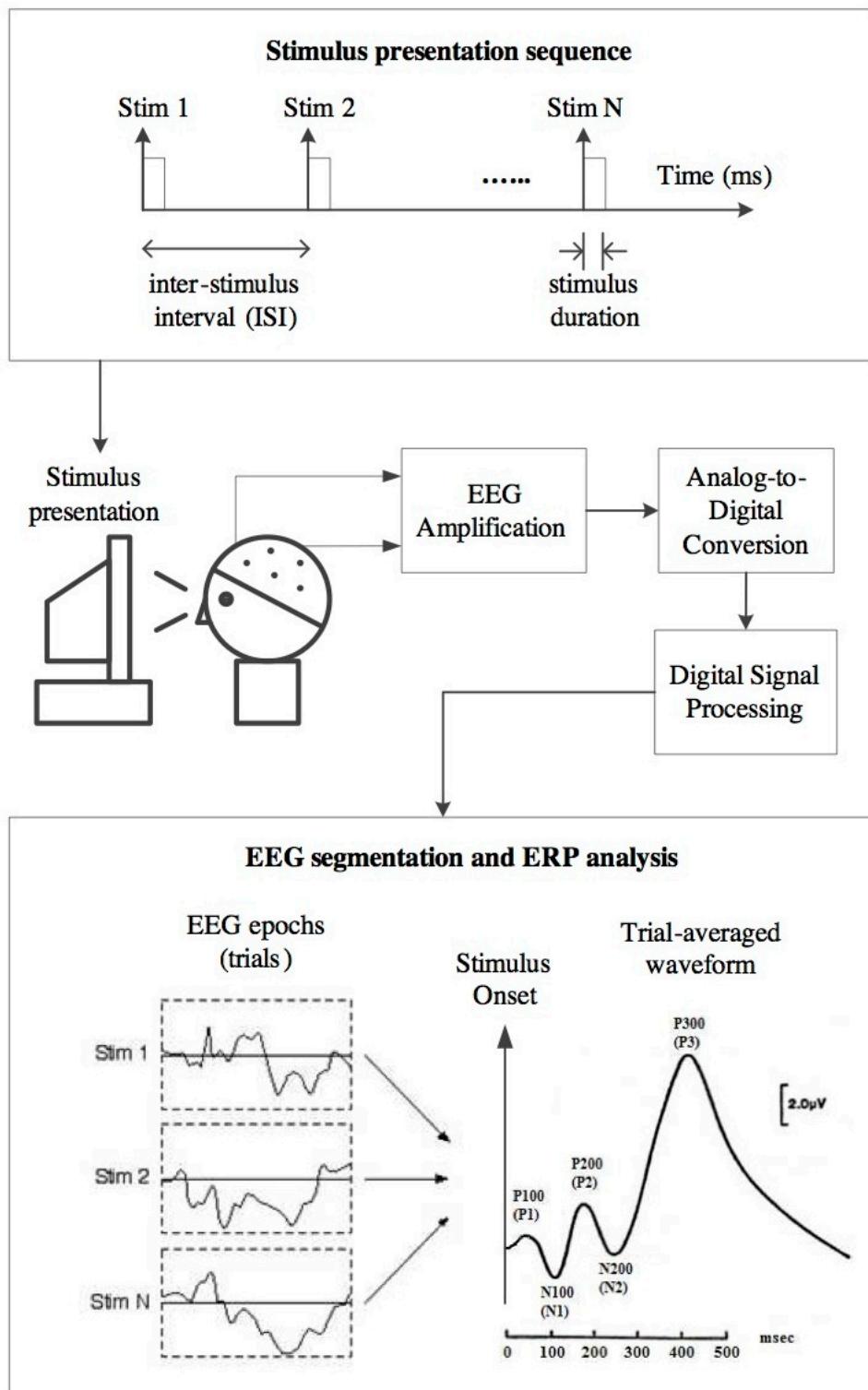
تحريك‌هایی که منجر به بروز ERP در EEG می‌شوند معمولاً به صورت زنجیروار^۳ انجام می‌شود. هم‌چنان فاصله‌ی بین تحريك‌ها به اندازه‌ای زیاد در نظر گرفته می‌شود که ERP فرصت وقوع کامل و ظهور در EEG را داشته باشد. شکل ۱-۲ نشان‌دهنده‌ی مراحل یک آزمایش ERP است. از آن جایی که دامنه‌ی ERP‌ها در مقایسه با نویز پس‌زمینه EEG بسیار کم است، معمولاً با میانگین‌گیری بین تکثیت‌های حاوی ERP و با فرض صفر بودن میانگین EEG پس‌زمینه، پتانسیل وابسته به رخداد استخراج می‌شود. استاندارد نام‌گذاری ERP‌ها به صورت ترکیب یکی از حروف N یا P با یک عدد است. حروف نشان‌دهنده‌ی قطبیدگی^۴ پتانسیل است و N و P به ترتیب به قطبیدگی منفی و مثبت در ERP‌ها اطلاق می‌شوند. عدد استفاده شده در نام‌گذاری ERP‌ها به تأخیر وقوع پتانسیل وابسته به رخداد نسبت به لحظه‌ی تحريك مرتبط است. برای مثال P300 (یا P3) نام یک پتانسیل مثبت است که حدود ۳۰۰ میلی‌ثانیه پس از تحريك به وجود می‌آید. به همین ترتیب، N200 پتانسیل منفی در حدود ۲۰۰ میلی‌ثانیه بعد از تحريك است. لازم به ذکر است که لحظه‌ی وقوع یک ERP را نمی‌توان به طور دقیق بیان کرد، چرا که نسبت به شرایط آزمایش تغییر می‌کند. تنها می‌توان از آن به صورت حدودی صحبت کرد. در این پایان‌نامه بارها از P300 نام خواهیم برد؛ به همین دلیل در اینجا کمی بیشتر به آن خواهیم پرداخت.

^۱Brain Computer Interface

^۲Segmentation

^۳Sequential

^۴Polarity



شکل ۲-۱: مراحل یک از آزمایش ERP همراه با پردازش مرسوم به وسیلهٔ میانگین‌گیری تکثبتهای [۱].

P300 به صورت یک پیک بزرگ مثبت در بازه‌ی ۵۰۰ - ۲۵۰ میلی‌ثانیه بعد از تحریک ظاهر می‌شود. از لحاظ مکانی، P300 در قسمت‌های مرکزی^۶ و آهیانه‌ای^۷ رخ می‌دهد. معمولاً آزمایشی که برای ایجاد P300 انجام می‌شود از پارادایم oddball استفاده می‌کند. در پارادایم oddball دو نوع تحریک وجود دارد که یکی از آن‌ها به صورت نادر رخ می‌دهد. تحریک‌ها به صورت زنجیروار و تصادفی به فرد نشان داده می‌شوند و سوزه باید به تحریک‌های نادر پاسخ دهد. این پاسخ می‌تواند به این صورت باشد که فرد تحریک‌های نادر را بشمارد یا هنگام وقوع آن‌ها کلیدی را فشار دهد. در این پارادایم، P300 هنگام وقوع تحریک‌های نادر در EEG ایجاد می‌شود.

۳-۳ سیستم رابط مغز-رایانه

سیستم‌های رابط مغز-رایانه، سیستم‌هایی هستند که به کاربر اجازه می‌دهند تا بدون حرکت فیزیکی، صرفاً از طریق داده‌های نهفته در سیگنال مغزی خود، وسایل و یا دستگاه‌های متصل به یک سیستم هوشمند را کنترل کند. کاربردهای وسیعی نظیر کمک به افراد ناتوان جسمی و ذهنی، توانبخشی و واقعیت مجازی^۸ را می‌توان برای BCI‌ها در نظر گرفت [۱۵].

روش‌های متعددی برای جمع‌آوری داده از مغز وجود دارد: EEG، fMRI، MEG، NIRS و fNIRS. در میان آن‌ها EEG ارزان‌قیمت‌ترین و قابل حمل^۹ است. همچنین fMRI و NIRS برای اندازه‌گیری کارکرد بلندمدت مغز مناسب هستند و رزولوشن زمانی خوبی ندارند. به همین دلیل اکثر BCI‌ها از EEG برای ثبت استفاده می‌کنند [۱۵].

همان‌طور که گفته شد، سیگنال EEG که در حالت عادی ثبت می‌شود، برآیندی از فعالیت‌های الکتریکی مغز است. به همین دلیل برای استفاده در BCI قابلیت کمی دارند. بنابراین لازم است ابتدا به وسیله‌ی تحریک‌ها و یا عملیات ذهنی^{۱۰} مغز را در حالتی قرار دهیم که سیگنال ثبت شده از آن برای استفاده در BCI مناسب باشد و بتوان آن را طبقه‌بندی کرد [۱۵]. در همین راستا آزمایش‌هایی که منجر به حضور ERP در EEG می‌شوند یکی از سناریوهایی است که می‌توان برای ثبت سیگنال در BCI‌ها متصور شد. در میان ERP‌ها، P300 یکی از مواردی است که برای کاربرد در سیستم‌های رابط

^۶Central

^۷Parietal

^۸Virtual reality

^۹Portable

^{۱۰}Mental task



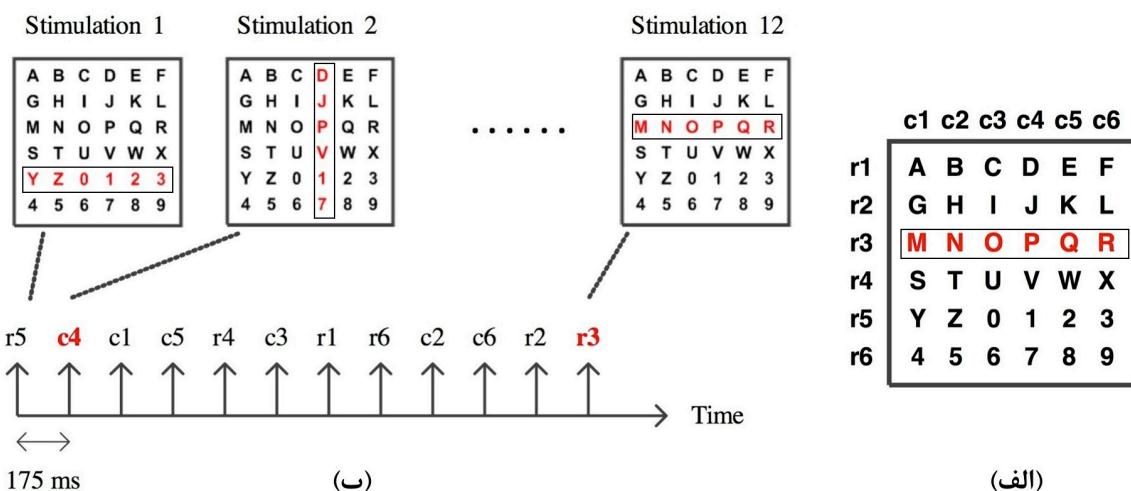
شکل ۲-۲: یک نمونه از صفحه‌ای که در P300-BCI به کاربر نمایش داده می‌شود. در این آزمایش تصاویر به صورت پی‌درپی و تصادفی روشن و خاموش می‌شوند. کاربر باید به یک تصویر خاص که مدنظرش است توجه کند. روشن شدن آن تصویر خاص باعث برانگیختن P300 در EEG می‌شود [۲].

مغز-رایانه بسیار مورد توجه قرار گرفته است^{۱۱} که کاربرد آن در سیستم‌های پردازش واژه و کنترلی است. اکثر BCI‌هایی که از P300 استفاده می‌کنند به طریقی از پارادایم oddball پیروی می‌کنند. هدف در P300-BCI‌ها این است که تشخیص داده شود کدام یک از تحریک‌ها در EEG منجر به برانگیختن P300 می‌شوند و از این طریق تحریک مورد نظر کاربر دریافت شود [۱]. برای مثال در شکل ۲-۲ یک نمونه از صفحه‌ای که در P300-BCI به کاربر نمایش داده می‌شود آورده شده است. P300 speller یکی از P300-BCI‌هایی است که در ادبیات بسیار بررسی شده است و در بخش بعدی کمی بیشتر به بررسی آن می‌پردازیم.

۴-۲ پارادایم P300 Speller

به طور مرسوم یک P300 speller از یک صفحه 6×6 تشکیل شده است که کاراکترهایی در خانه‌های آن قرار دارند (شکل ۳-۲-الف). همان‌طور که در شکل ۳-۲-ب نشان داده شده، سطرها و

^{۱۱} می‌توان به پتانسیل‌های وابسته به رخداد دیگر نظیر SSVEP اشاره کرد.



شکل ۲-۳: (الف) صفحه‌ی P300 speller. نمایش شماتیک شیوه‌ی ارائه‌ی تحریک‌ها (روشن شدن سطر/ستون‌ها) به کاربر؛ سطرهای و ستون‌ها در بلوک‌های ۱۲ تایی پشت سر هم و با ترتیب تصادفی روشن می‌شوند [۱].

ستون‌ها در بلوک‌های ۱۲ تایی به صورت زنجیروار و با ترتیب تصادفی روشن می‌شوند. این بلوک‌های تصادفی برای هر کاراکتر R بار تکرار می‌شوند (به عبارتی هر سطر (یا ستون) به ازای هر کاراکتر R بار تکرار می‌شود). در طی آزمایش، کاربر باید بر روی یک حرف تمکن کند. در این صورت، هنگام روشن شدن سطر و ستونی که حاوی آن حرف هستند P300 برانگیخته می‌شود. هدف نهایی یک P300 speller تشخیص سطر و ستونی است که P300 دارند تا کاراکتر مورد نظر کاربر تشخیص داده شود. برای این منظور مراحل زیر باید انجام شود:

۱. جداسازی تکثیت‌ها: معمولاً نمونه‌های زمانی حدود ۱۰۰۰ – ۶۰۰۰ میلی‌ثانیه بعد از تحریک به عنوان یک تکثیت جداسازی می‌شوند. هرچند معمولاً تکثیت را ترجمه‌ی single trial می‌دانند و زمانی استفاده می‌شود که تحریک فقط یک بار تکرار ^{۱۲} شود (یعنی در توضیحات بالا $R = 1$). با این وجود استفاده از تکثیت به جای trial چندان هم غیرمنطقی نیست؛ در برخی از مراجع هر یک از آنها را یک single trial دانسته‌اند ^{۱۳} [۱۶]. پس از جداسازی تکثیت‌ها از EEG، لازم است از آنها استخراج ویژگی صورت بگیرد.

۲. طبقه‌بندی: در این مرحله تکثیت‌ها طبقه‌بندی دو کلاسه می‌شوند. در برخی موارد بین تکرارهای مربوط به

^{۱۲}Repetition
^{۱۳}انتخابهای دیگر برای واژه‌ی trial، «تکرار» و «ثبت» هستند که اولی را برای repetition استفاده می‌کنیم و دومی معمولاً به ثبت EEG اشاره دارد.

هر سطر (یا ستون) میانگین‌گیری شده تا EEG پس‌زمینه حذف شود و بعد از آن طبقه‌بندی انجام می‌شود.

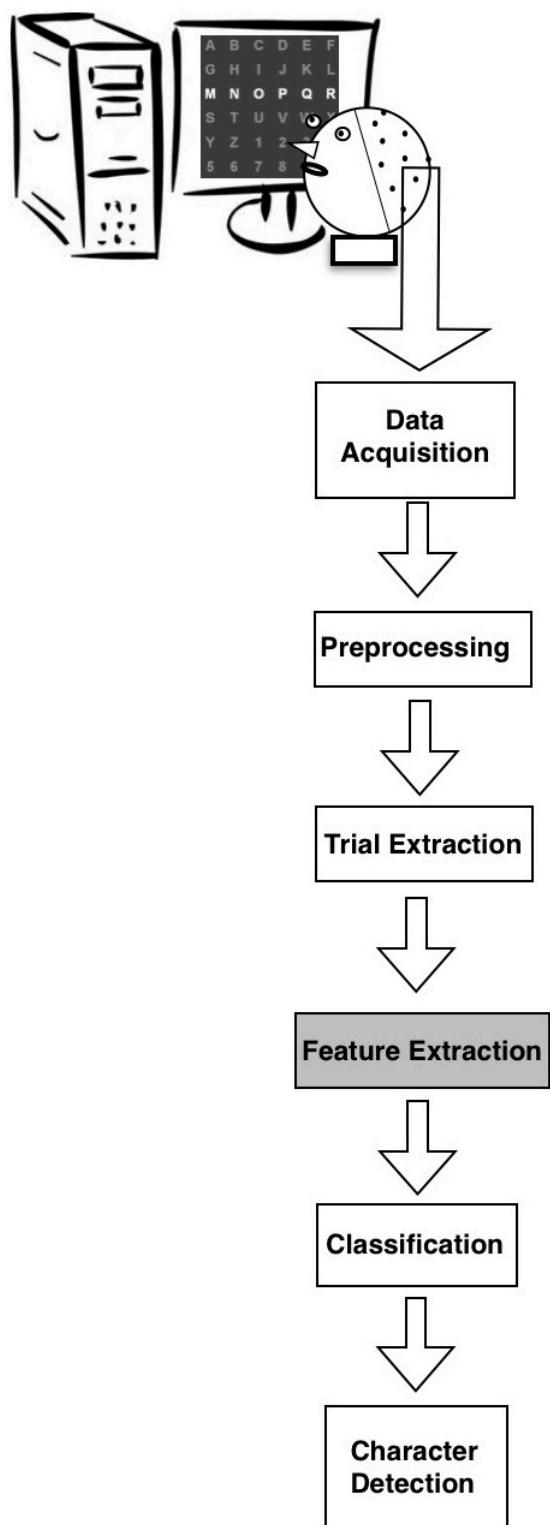
۳. تشخیص کاراکتر: از تجمعی نتایج تک‌ثبت‌ها، سطر و ستونی که حاوی کاراکتر مورد نظر کاربر است تشخیص داده شود.

شکل ۴-۲ بلوک‌دیاگرام یک سیستم P300 speller را نشان می‌دهد. در این پایان‌نامه ما عمدتاً بر مرحله‌ی استخراج ویژگی متمرکز می‌شویم. استخراج ویژگی در بسیاری از موارد با «کاهش ویژگی» مشابه تلقی می‌شود.^{۱۴}.

۵-۲ جمع‌بندی

در این فصل مفاهیم اولیه‌ی مرتبط با پایان‌نامه را مرور کردیم و مفاهیمی همچون پتانسیل وابسته به رخداد، سیستم‌های رابط مغز-رایانه و پارادایم P300 speller پوشش داده شد.

^{۱۴} به طور خاص، زمانی از «کاهش ویژگی» استفاده می‌شود که ویژگی‌های نهایی از ترکیب خطی ویژگی‌های اولیه به دست بیایند.



شکل ۲-۴: بلوک دیاگرام یک P300 speller

فصل ۳

پیش‌زمینه‌ی نظری

۱-۳ پیش‌گفتار

در این فصل به مرور مبانی نظری پایان‌نامه می‌پردازیم. این مبانی عبارت‌اند از روش‌های برداری کاهش ویژگی، جبر تانسوری و روش‌های تانسوری کاهش ویژگی. تلاش بر آن بوده تا تمامی مباحث، در حد نیاز، پوشش داده شوند به طوری که نیاز به مرجع دیگری در این زمینه نباشد.

۲-۳ نمادگذاری

هر یک از مراجع نمادگذاری مخصوص به خود را دارند، در بسیاری از موارد شباهت‌ها و تفاوت‌هایی در آن‌ها وجود دارد. در این نوشتار برای یک دست بودن توضیحات یک نمادگذاری واحد را تعریف می‌کنیم و در تمامی قسمت‌ها از آن‌ها استفاده می‌کنیم.

تانسورها، ماتریس‌ها و بردارها را به ترتیب با کاراکترهای Caligraphic، حروف بزرگ ضخیم و حروف کوچک ضخیم نشان می‌دهیم. برای مثال \mathcal{X} یک تانسور، X یک ماتریس و x یک بردار است. حروف عادی (مثل X یا x) اسکالر هستند. در ادامه‌ی این بخش نمادهایی را که در سرتاسر فصل (و پایان‌نامه) مورد استفاده قرار می‌گیرند معرفی می‌کنیم. طبیعی است که هر بخش ممکن است به فراخور نیازش نمادهای اضافه‌ای داشته باشد، در این صورت در همان بخش نمادهای مربوطه تعریف خواهند شد.

درایه‌های ماتریس A را با a_{ij} یا (i,j) a و ستون‌های آن را به صورت $[a_1, \dots, a_L]$ نشان می‌دهیم. در مسئله‌ی طبقه‌بندی تعداد داده‌ی آموزش و تعداد کلاس‌های مسئله را به ترتیب با K و C نشان می‌دهیم. اندیس‌های k ، c_i و c به ترتیب برای داده‌ها، شماره‌ی کلاس‌ها و شماره‌ی کلاس داده‌ی آن

استفاده می‌شوند. تعداد داده‌های کلاس c با K_c نشان داده می‌شود. مجموعه‌ی Ω_c مجموعه‌ی اندیس‌های داده‌های کلاس c و مجموعه‌ی Ω مجموعه‌ی اندیس‌های کل داده‌های آموزش است. ماتریس پراکندگی درون کلاسی و بین کلاسی به ترتیب با S_b و S_w نشان داده می‌شوند و $S_t \triangleq S_b + S_w$. معیارهای اسکالر پراکندگی درون کلاسی و بین کلاسی که از روی ماتریس‌های پراکندگی به دست می‌آیند را به ترتیب با s_b و s_w نمایش می‌دهیم.

۳-۳ روش‌های برداری کاهش ویژگی

بسیاری از روش‌های کاهش ویژگی تansوری، تعمیم‌هایی از روش‌های کاهش ویژگی برداری هستند. از همین روی برای درک بهتر روش‌های تansوری، که هدف این پروژه هستند، به نظر می‌رسد بررسی و معرفی روش‌های برداری مربوطه خالی از لطف نباشد. در ادامه‌ی این فصل به معرفی اجمالی روش LDA^۱ و SRDA^۲ می‌پردازیم.

LDA ۱-۳-۳

اگر $\{x_1, x_2, \dots, x_K\} \subset \mathbb{R}^m$ مجموعه‌ی K نمونه‌ی در دسترس ما، از C کلاس، باشد؛ LDA به دنبال یافتن یک تبدیل خطی $y = Ax \in \mathbb{R}^l, l < m$ است که باعث بشود نمونه‌های هم کلاس به هم نزدیک و نمونه‌های کلاس‌های مختلف از هم دور شوند. ماتریس تبدیل $A = [a_1, a_2, \dots, a_m]$ از رابطه‌ی زیر به دست می‌آید:

$$A = \underset{A}{\operatorname{argmax}} \frac{\operatorname{tr}\{A^T S_b A\}}{\operatorname{tr}\{A^T S_w A\}} \quad (1-3)$$

که در آن

$$S_b = \sum_{c=1}^C K_c (\mu^{(c)} - \mu) (\mu^{(c)} - \mu)^T \quad (2-3)$$

$$S_w = \sum_{c=1}^C \left(\sum_{i=1}^{K_c} (x_i^{(c)} - \mu^{(c)}) (x_i^{(c)} - \mu^{(c)})^T \right) \quad (3-3)$$

در روابط بالا $\mu^{(c)}$ میانگین کلاس c و μ میانگین کل داده‌ها است.

مسئله‌ی ۱-۳ را می‌توان به وسیله‌ی GEVD حل کرد: ستون‌های ماتریس A بردار ویژه‌های متناظر با

بزرگ‌ترین مقادیر ویژه‌ی مسئله‌ی زیر خواهند بود:

^۱Linear Discriminant Analysis

^۲Spectral Regression Discriminant Analysis

$$S_b \mathbf{a} = \lambda S_w \mathbf{a} \quad (4-3)$$

با تعریف S_t تابع هزینه‌ی ۳-۱ معادل با مسأله‌ی زیر خواهد بود:

$$A = \underset{\mathbf{A}}{\operatorname{argmax}} \frac{\operatorname{tr}\{\mathbf{A}^T S_b \mathbf{A}\}}{\operatorname{tr}\{\mathbf{A}^T S_t \mathbf{A}\}} \quad (5-3)$$

حل مسأله‌ی بالا معادل با یافتن \mathbf{A} بردار ویژه‌ی متناظر با بزرگ‌ترین مقادیر ویژه‌ی مسأله‌ی GEVD زیر است:

$$S_b \mathbf{a} = \lambda S_t \mathbf{a} \quad (6-3)$$

از آن‌جا که رتبه‌ی ماتریس S_b حداقل $C - 1$ است، حداقل $C - 1$ بردار ویژه‌ی متناظر با مقادیر ویژه‌ی ناصفر دارد. بنابراین LDA با حل دو مسأله‌ی ۴-۳ و ۶-۳ حداقل کاهش بُعد به $C - 1$ را انجام می‌دهد. همچنین برای این‌که مسأله‌های ۴-۳ و ۶-۳ جواب پایدار داشته باشند، لازم است S_t و S_w ماتریس‌های معکوس‌پذیر^۳ باشند. در حالی که اگر تعداد ویژگی‌ها از تعداد نمونه‌ها بیشتر باشد، برقرار خواهد بود.

برای رفع این مشکل معکوس‌نایپذیری S_t روش‌هایی پیشنهاد شده است که LDA بعد از یک مرحله کاهش ویژگی PCA یا SVD اعمال می‌شود که خود هزینه‌ی محاسباتی و زمانی به الگوریتم اضافه می‌کند. یکی از روش‌هایی که برای حل این مشکل معرفی شده، روش SRDA است که مسأله‌ی تجزیه‌ی مقدار ویژه‌بردار ویژه^۴ را به حل دستگاه معادلات خطی تبدیل می‌کند. این روش در ادامه شرح داده خواهد شد.

SRDA ۲-۳-۳

در [۱۷] روش SRDA معرفی شده است که می‌توان گفت از بازنویسی روابط LDA به دست آمده است. در این مقاله نشان داده شده است که مسأله‌ی LDA را می‌توان با حل یک دستگاه معادلات خطی حل کرد و دیگر نیازی به حل مسأله‌ی GEVD نخواهد بود که این، صرفه‌جویی در زمان و بار محاسباتی خواهد بود و مسأله‌ی معکوس‌نایپذیری ماتریس پراکندگی را نیز تا حدی مرتفع می‌سازد. در این مقاله همچنین مسأله‌ی LDA از دیدگاه آنالیز طیفی گراف^۴ بررسی شده است. دو عبارت regression و spectral در نام این روش از این دو موضوع برگرفته شده است.

^۳Non-singular

^۴Spectral graph analysis

اکنون برای آن که بتوانیم LDA را از دیدگاه گرافی بررسی کنیم، باز دیگر فرض کنیم داده‌ی K مجموعه‌ی آموزش است که هر کدام به یکی از C کلاس تعلق دارند. اگر قرار دهیم $\bar{X}^{(c)} = [\bar{x}_1^{(c)}, \bar{x}_2^{(c)}, \dots, \bar{x}_{K_c}^{(c)}]$ $\bar{x}_k = x_k - \mu$ داده‌ها (یعنی μ) از آن کم شده است. در این صورت داریم:

$$\begin{aligned} S_b &= \sum_{c=1}^C K_c (\boldsymbol{\mu}^{(c)} - \boldsymbol{\mu}) (\boldsymbol{\mu}^{(c)} - \boldsymbol{\mu})^T \\ &= \sum_{c=1}^C K_c \left(\frac{1}{K_c} \sum_{k=1}^{K_c} (\mathbf{x}_k^{(c)} - \boldsymbol{\mu}) \right) \left(\frac{1}{K_c} \sum_{i=1}^{K_c} (\mathbf{x}_k^{(c)} - \boldsymbol{\mu}) \right)^T \\ &= \sum_{c=1}^C \frac{1}{K_c} \left(\sum_{k=1}^{K_c} \bar{x}_k^{(c)} \sum_{i=1}^{K_c} (\bar{x}_k^{(c)})^T \right) \\ &= \sum_{c=1}^C \bar{X}^{(c)} \mathbf{W}^{(c)} (\bar{X}^{(c)})^T \end{aligned} \quad (7-3)$$

که در آن $\mathbf{W}^{(c)}$ ماتریس $K_c \times K_c$ است که همه‌ی درایه‌های آن ۱/ است.

$$\begin{aligned} \text{اکنون قرار دهیم } \bar{X} &= [\bar{X}^{(1)}, \dots, \bar{X}^{(C)}] \text{ و همچنین} \\ \mathbf{W} &= \begin{bmatrix} \mathbf{W}^{(1)} & & \cdots & & \cdot \\ & \ddots & & \mathbf{W}^{(2)} & \cdots & \cdot \\ & & \ddots & & \ddots & \vdots \\ \cdots & & \cdots & & \ddots & \vdots \\ & \ddots & & \cdot & & \mathbf{W}^{(C)} \end{bmatrix} \end{aligned} \quad (8-3)$$

در این صورت داریم:

$$S_b = \sum_{c=1}^C \bar{X}^{(c)} \mathbf{W}^{(c)} (\bar{X}^{(c)})^T = \bar{X} \mathbf{W} \bar{X}^T \quad (9-3)$$

و از آنجا که $S_t = \bar{X} \bar{X}^T$ داریم:

$$S_w = S_t - S_b = \bar{X} (\mathbf{I} - \mathbf{W}) \bar{X}^T = \bar{X} \mathbf{L} \bar{X}^T \quad (10-3)$$

اگر \mathbf{W} را ماتریس مجاورت گراف توصیف‌کننده‌ی روابط داده‌ها در نظر بگیریم، در این صورت \mathbf{L} ماتریس لaplاسین این گراف خواهد بود.

با قرار دادن ۹-۳ در رابطه‌های ۶-۳ خواهیم دید که ستون‌های ماتریس تبدیل A بردار ویژه‌های

مسئله‌ی زیر هستند:

$$\bar{X} \mathbf{W} \bar{X}^T \mathbf{a} = \lambda \bar{X} \bar{X}^T \mathbf{a} \quad (11-3)$$

لم ۱.۳. فرض کنید که y بردار ویژه‌ی مسأله‌ی زیر باشد:

$$W\mathbf{y} = \lambda\mathbf{y} \quad (12-3)$$

در این صورت اگر $\mathbf{y} = \bar{X}^T \mathbf{a}$, آن‌گاه \mathbf{a} بردار ویژه‌ی مسأله‌ی ۱۱-۳ متناظر مقدارویژه‌ی λ خواهد بود.

اثبات: اگر $\mathbf{y} = \bar{X}^T \mathbf{a}$ را در سمت چپ عبارت ۱۱-۳ قرار بدھیم داریم:

$$\bar{X}W\bar{X}^T \mathbf{a} = \bar{X}W\mathbf{y} = \bar{X}\lambda\mathbf{y} = \lambda\bar{X}\mathbf{y} = \lambda\bar{X}\bar{X}^T \mathbf{a} \quad \square$$

بنابراین برای یافتن ستون‌های ماتریس تبدیل A دو مرحله‌ی لازم است: ۱) بردارویژه‌های ماتریس W را بیابیم، ۲) با حل دستگاه $\mathbf{y} = \bar{X}^T \mathbf{a}$, ستون‌های A را بیابیم. اما در ادامه نشان می‌دهیم که برای به دست آوردن بردارویژه‌های ماتریس W نیازی به تجزیه‌ی بردار ویژه‌مقدار ویژه نیست و می‌توان آن‌ها را به صورت دستی از روابط ریاضی به دست آورد.

از آن‌جایی که W یک ماتریس قطری بلوکی است، رتبه‌ی آن برابر جمع رتبه‌ی بلوک‌های روی قطرش است. برای هر c ماتریس $W^{(c)}$ از رتبه‌ی یک است. مشخص است که $(1, e^{(c)} = \text{ones}(K_c, 1))$ بردار ویژه‌ی $W^{(c)}$ متناظر با مقدارویژه‌ی یک است. پس بردارویژه‌های W , متناظر با مقدارویژه‌ی یک به صورت زیر هستند:

$$\mathbf{y}_c = \begin{bmatrix} \underbrace{\cdot, \dots, \cdot}_{\sum_{i=1}^{c-1} K_i}, \underbrace{1, \dots, 1}_{K_c}, \underbrace{\cdot, \dots, \cdot}_{\sum_{i=c+1}^C K_i} \end{bmatrix}^T, \quad c = 1, \dots, C \quad (13-3)$$

از آن‌جا که یک مقدارویژه‌ی تکرارشونده‌ی W است، هر C بردار متعامدی که در فضای تولیدشده توسط \mathbf{y}_c ها قرار داشته باشند را می‌توان بردارویژه‌ی W در نظر گرفت. برای آن‌که معادله‌ی خطی $\bar{X}^T \mathbf{a} = y$ دارای جواب باشد، لازم است که y در فضای تولیدشده به وسیله‌ی سطور \bar{X} قرار داشته باشد. از آن‌جا که $\bar{X}e = 0$, بردار e بر این فضا عمود است (e برداری است که تمام درایه‌های آن یک است). پس می‌توانیم e را به عنوان اولین بردار ویژه‌ی W انتخاب کنیم و یک فرآیند عمودسازی Gram-Schmidt را به عنوان اولین بردار ویژه‌ی W اعمال کنیم و در پایان با حذف e , $C - 1$ بردارویژه باقی خواهد ماند.

۴-۳ جبر تانسوری

آرایه‌های چندبعدی را تانسور^۵ می‌خوانند. مرتبه‌ی^۶ یک تانسور برابر تعداد بُعد^۷ آن است. برای مثال اسکالرها تانسورهای مرتبه صفر، بردارها تانسورهای مرتبه یک و ماتریس‌ها تانسورهای مرتبه دو هستند (گفته می‌شود که تانسورها تعمیمی از ماتریس‌ها و بردارها هستند). به تانسورهای مرتبه سه و بالاتر تانسور مرتبه بالا^۸ گفته می‌شود. استفاده از روش‌های تجزیه‌ی تانسوری در طیف گسترده‌ای از کاربردها مورد اقبال قرار گرفته است: علوم اعصاب، پردازش سیگنال [۱۸] ، داده‌کاوی و غیره.

در این بخش به معرفی تانسورها و عملیات میان آن‌ها می‌پردازیم و دو تجزیه‌ی تانسوری Tucker و CP^۹ را، که در این نوشتار کاربرد دارند، معرفی می‌کنیم. مطالب این بخش با استفاده از [۳، ۶، ۷] تدوین شده است.

۱-۴-۳ تانسورها و عملیات میان آن‌ها

۱-۱-۴-۳ تعاریف اولیه

تعریف ۳-۱. (تانسور) فرض کنید $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$. در این صورت $I_1, I_2, \dots, I_N \in \mathbb{N}$ یک تانسور مرتبه N است که درایه‌های آن به صورت a_{i_1, i_2, \dots, i_N} یا $\mathcal{A}(i_1, i_2, \dots, i_N)$ نمایش داده می‌شوند که برای $1 \leq n \leq N$ به هریک از ابعاد یک تانسور «مُد» گفته می‌شود. از این پس از اندیس $i_n \in \{1, 2, \dots, I_n\}$ برای اشاره به مُد- n یک تانسور مرتبه N استفاده می‌کنیم. شکل ۱-۳ یک تانسور مرتبه سه را نشان می‌دهد.

تعریف ۳-۲. (فیبر) فیبرهای یک تانسور^{۱۰} بردارهایی هستند که به وسیله‌ی ثابت کردن همه‌ی اندیس‌ها به جز یکی از آن‌ها به دست می‌آیند. اگر اندیس مربوط به مُد- n -ام آزاد باشد می‌گوییم فیبرهای مُد- n به دست می‌آیند. اندیس مُد آزاد را با ”:“ نشان می‌دهیم. در تانسور تعریف ۳-۱ فیبرهای مُد- n به صورت $(i_1, i_2, \dots, i_{n-1}, :, i_{n+1}, \dots, i_N) \mathcal{A}$ نشان داده می‌شوند. شکل ۲-۳ فیبرهای یک تانسور مرتبه سه را نشان می‌دهد.

^۵Tensor

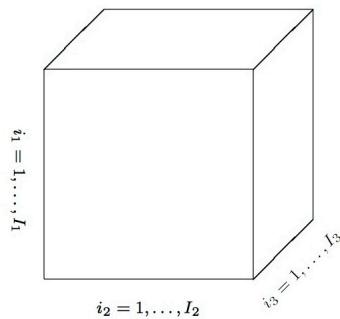
^۶Order

^۷Dimension also called Way or Mode

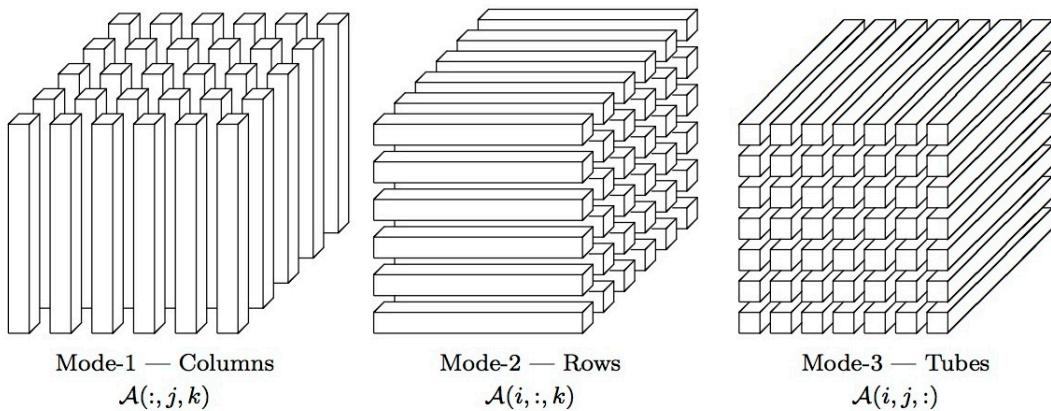
^۸Higher order Tensor

^۹Canonical Polyadic

^{۱۰}Tensor Fiber



شکل ۳-۱: تانسور مرتبه سه [۳].



شکل ۳-۲: فیرهای یک تانسور مرتبه سه [۳].

تعریف ۳-۳. (اسلایس) اسلایس‌های تانسور^{۱۱} ماتریس‌هایی هستند که به وسیله‌ی ثابت کردن همه‌ی اندیس‌ها به جز دو تا از آن‌ها به دست می‌آیند. شکل ۳-۳ اسلایس‌های یک تانسور مرتبه سه را نشان می‌دهد.

تعریف ۳-۴. (نُرم) نُرم تانسور^{۱۲} $A \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ به صورت ریشه دوم مجموع مربع درایه‌های آن

تعریف می‌شود:

$$\|A\| = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \dots \sum_{i_N=1}^{I_N} a_{i_1 i_2 \dots i_N}^2}$$

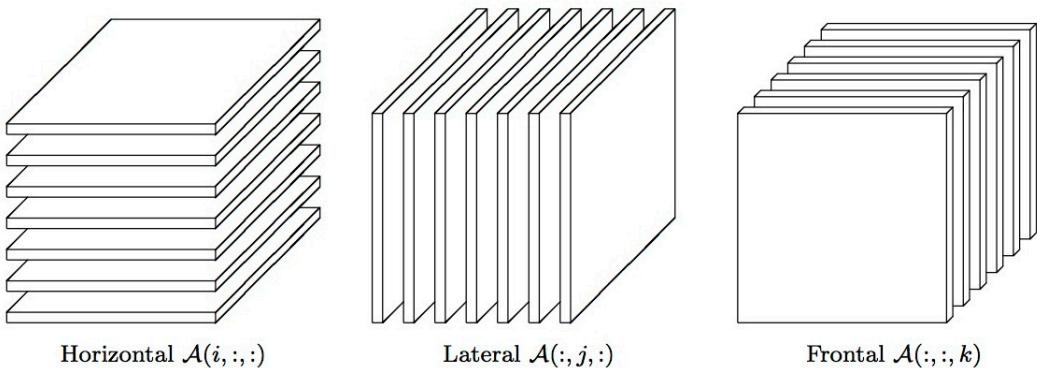
تعریف ۳-۵. (تانسور سوپرقطری) تانسور^{۱۳} $A \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ را سوپرقطری می‌گوییم هرگاه فقط برای $i_1 = i_2 = \dots = i_N$ داشته باشیم $a_{i_1 i_2 \dots i_N} \neq 0$. شکل ۳-۳ یک تانسور سوپرقطری مرتبه سه را نمایش می‌دهد.

یکی از مفاهیم مهم در جبر تانسوری، مفهوم ماتریسی کردن^{۱۴} است که به معنی قرار دادن درایه‌های

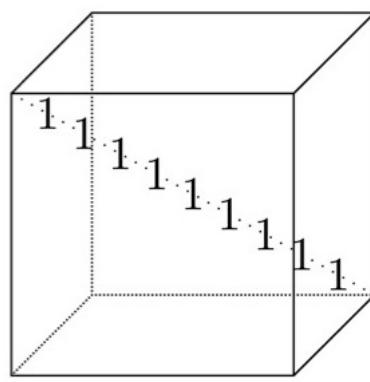
^{۱۱}Tensor Slice

^{۱۲}Super diagonal

^{۱۳}Matricization, also called Unfolding



شکل ۳-۳: اسلایس‌های یک تانسور مرتبه سه [۳].

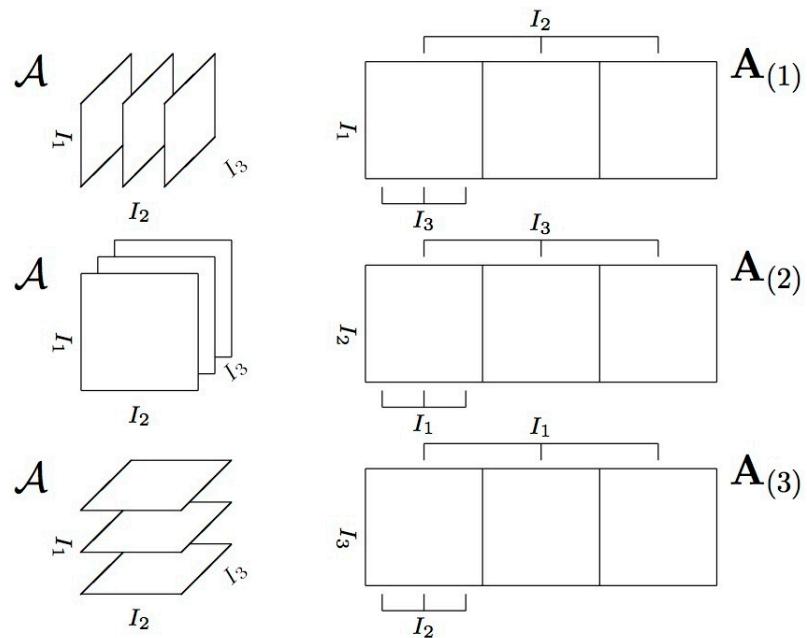


شکل ۳-۴: یک تانسور سوپرقطری که درایه‌های روی قطر آن یک است. این تانسور را تانسور یکانی گویند [۳].

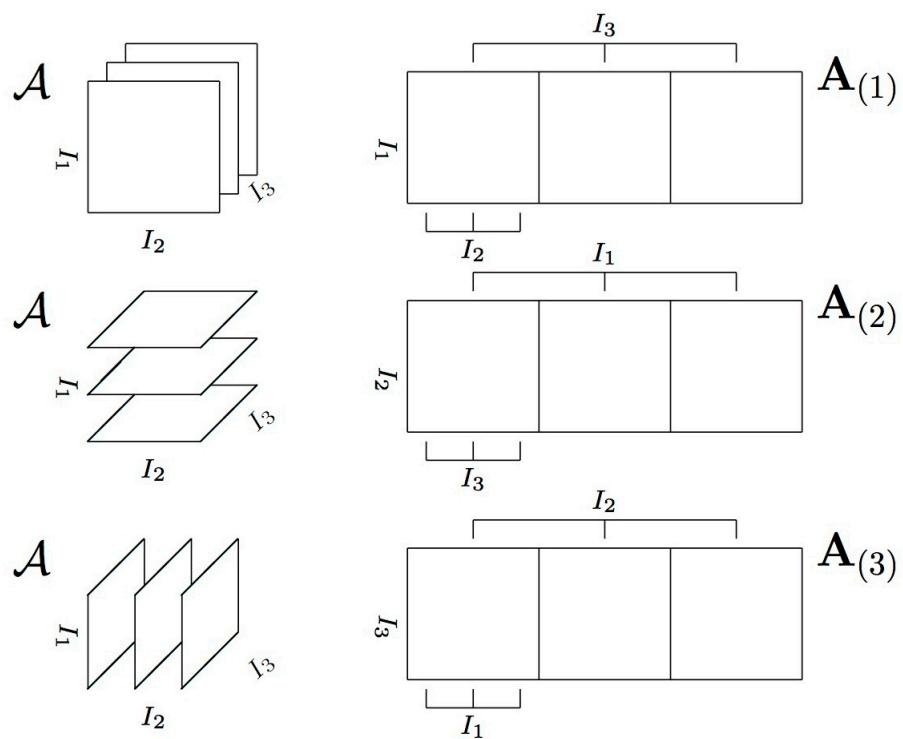
یک تانسور در کنار هم به شکل یک ماتریس است. روشن است که روش‌های متعددی برای این عمل می‌توان متصور شد. به همین دلیل ماتریسی کردن یک عمل یکتا نیست. روش مرسوم برای ماتریسی کردن تانسور «ماتریسی کردن مُد- n » است. با این وجود این روش هم یکتا نیست! سه روش ماتریسی کردن مُد- n را در اینجا معرفی می‌کنیم. انتخاب این‌که از کدام روش استفاده کنیم سلیقه‌ای است!! تنها نکته‌ی مهم این است که به خاطر داشته باشیم در تمامی عملیات از یک روش واحد برای ماتریسی‌سازی استفاده کنیم.

تعریف ۳-۶. (ماتریسی کردن مُد- n) فرم ماتریسی‌شده‌ی مُد- n ^{۱۴} تانسور $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ به وسیله‌ی $A_{(n)}$ نمایش داده می‌شود و فیبرهای مُد- n را در ستون‌های یک ماتریس می‌چینند. ترتیب قرارگیری فیبرها روش‌های مختلفی را برای ماتریسی کردن به وجود می‌آورد. شکل‌های ۳-۵ تا ۳-۷ سه روش تانسوری کردن مُد- n را نشان می‌دهند.

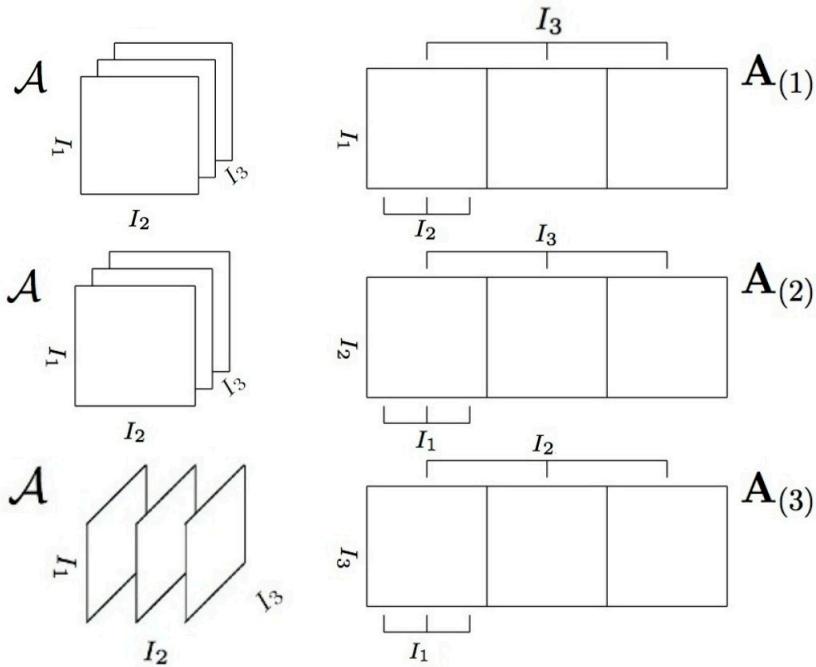
^{۱۴}Mode-n unfolding



شکل ۳-۵: تانسوری کردن مُد- n به روش [۴].



شکل ۳-۶: تانسوری کردن مُد- n به روش [۵].

شکل ۳-۷: تانسوری کردن مُد- n به روش [۶].

تعریف ۳-۷. (برداری کردن) برداری کردن^{۱۵} تانسور A به وسیله‌ی برداری کردن $A_{(1)}$ صورت می‌گیرد. برای برداری کردن یک ماتریس کافی است ستون‌های آن را در یک بردار کنار هم بچینیم.

۳-۱-۴-۲ ا نوع ضرب تانسوری

تعریف ۳-۸. (ضرب داخلی) ضرب داخلی دو تانسور $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ به صورت مجموع ضرب درایه‌های متناظر آن‌ها تعریف می‌شود:

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \dots \sum_{i_N=1}^{I_N} x_{i_1 i_2 \dots i_N} y_{i_1 i_2 \dots i_N} \quad (14-3)$$

روشن است که $\langle \mathcal{X}, \mathcal{X} \rangle = \|\mathcal{X}\|^2$

تعریف ۳-۹. (ضرب خارجی) ضرب خارجی دو تانسور $\mathcal{Y} \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_M}$ و $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ به صورت زیر تعریف می‌شود:

$$\mathcal{Z} = \mathcal{X} \circ \mathcal{Y} \in \mathbb{R}^{I_1 \times \dots \times I_N \times J_1 \times \dots \times J_M} \quad (15-3)$$

$$z_{i_1 i_2 \dots i_N j_1 j_2 \dots j_M} = x_{i_1 i_2 \dots i_N} y_{j_1 j_2 \dots j_M} \quad (16-3)$$

^{۱۵}Vectorization

که در آن

$$i_n \in \{ 1, \dots, I_n \}, 1 \leq n \leq N$$

$$j_m \in \{ 1, \dots, J_m \}, 1 \leq m \leq M$$

حالت خاصی از ضرب خارجی که در جبر تانسوری کاربرد زیاد دارد، ضرب خارجی چندین بردار است. بردارهای $a_1, \dots, a_N \in \mathbb{R}^{I_n \times 1}$ ، $1 \leq n \leq N$ را در نظر بگیرید که برای هر a . ضرب خارجی این N

بردار یک تانسور مرتبه N است که به صورت زیر تعریف می‌شود:

$$\begin{aligned} \mathcal{X} &= a_1 \circ a_2 \circ \dots \circ a_N \in \mathbb{R}^{I_1 \times \dots \times I_N} \\ (17-3) \end{aligned}$$

$$\mathcal{X}(i_1, i_2, \dots, i_N) = a_1(1) a_2(2) \dots a_N(N)$$

تعریف ۳.۱۰-۳. (ضرب منقطع^{۱۶}) ضرب منقطع دو تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_M \times J_1 \times \dots \times J_N}$ و

$\mathcal{Y} \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_N \times K_1 \times \dots \times K_P}$ در جهت مُدهای ۱ تا M ، یک تانسور با ابعاد $\mathcal{Z} = \langle \mathcal{X}, \mathcal{Y} \rangle_{1, \dots, M; 1, \dots, P} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_M \times K_1 \times \dots \times K_P}$

است و به صورت زیر تعریف می‌شود:

$$\langle \mathcal{X}, \mathcal{Y} \rangle_{1, \dots, M; 1, \dots, P}(j_1, \dots, j_N, k_1, \dots, k_P) = \sum_{i_1}^{I_1} \dots \sum_{i_M}^{I_M} x_{i_1, \dots, i_M, j_1, \dots, j_N} y_{i_1, \dots, i_M, k_1, \dots, k_P} \quad (18-3)$$

مُدهای باقی‌مانده به صورتی مرتب می‌شوند که مدهای مربوط به \mathcal{X} قبل از \mathcal{Y} بیایند. ضرب منقطع لزوماً نباید در جهت مُدهای یکسانی از دو تانسور صورت گیرد، اما باید اندازه‌ی مُدهایی که ضرب در جهت آن انجام می‌شود برابر باشند. برای مثال ضرب منقطع در جهت مُددوم تانسور $\mathcal{X} \in \mathbb{R}^{3 \times 4 \times 5}$ و مُددوم تانسور $\mathcal{Y} \in \mathbb{R}^{7 \times 8 \times 4}$ تانسور $\mathcal{Z} = \langle \mathcal{X}, \mathcal{Y} \rangle_{2, 3} \in \mathbb{R}^{3 \times 5 \times 7 \times 8}$ را برمی‌گرداند. نماد ضرب منقطع دو تانسور \mathcal{X} و \mathcal{Y} در راستای M مد یکسان به صورت زیر ساده می‌شود:

$$\langle \mathcal{X}, \mathcal{Y} \rangle_{1, \dots, M; 1, \dots, P} = \langle \mathcal{X}, \mathcal{Y} \rangle_{1, \dots, M} \quad (19-3)$$

برای دو تانسور $\mathcal{Y} \in \mathbb{R}^{J_1 \times \dots \times J_N}$ و $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ ضرب منقطع در جهت همه مُدها جز مُد n به

صورت زیر تعریف می‌شود:

$$\langle \mathcal{X}, \mathcal{Y} \rangle_{-n} = \mathbf{X}_{(n)} \mathbf{Y}_{(n)}^T, \quad (I_k = J_k, \quad \forall k \neq n) \quad (20-3)$$

در حالتی که \mathcal{X} و \mathcal{Y} دو تانسور مرتبه N با ابعاد برابر باشند ضرب منقطع آن‌ها در راستای مُدهای ۱ تا N همان ضرب داخلی دو تانسور است:

^{۱۶}Contracted product

$$\langle \mathcal{X}, \mathcal{Y} \rangle_{1, \dots, N} = \langle \mathcal{X}, \mathcal{Y} \rangle \quad (21-3)$$

همین‌طور اگر برای دو تانسور دلخواه در رابطه‌ی ۳-۱۹ داشته باشیم $M = ۰$ ، ضرب منقطع آن‌ها همان ضرب خارجی آن‌ها خواهد بود.

تعریف ۳-۱۱. (ضرب مُد n) ضرب مُد n تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ در ماتریس $U \in \mathbb{R}^{J \times I_n}$ عضو فضای

است و به صورت زیر تعریف می‌شود:

$$\begin{aligned} \mathcal{Y} &= \mathcal{X} \times_n U \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N} \\ \mathcal{Y}(i_1, i_2, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N) &= \sum_{i_n=1}^{I_n} \mathcal{X}(i_1, i_2, \dots, i_N) U(j, i_n) \end{aligned} \quad (22-3)$$

به عبارتی هر فیبر مُد- n از \mathcal{X} در U ضرب می‌شود. درک ساده‌تر از ضرب مُد n به صورت زیر است:

$$\mathcal{Y} = \mathcal{X} \times_n U \Leftrightarrow \mathbf{Y}_{(n)} = \mathbf{U} \mathbf{X}_{(n)} \quad (23-3)$$

می‌توان نشان داد:

$$\mathcal{X} \times_n A \times_m B = \mathcal{X} \times_m B \times_n A, \quad (m \neq n) \quad (24-3)$$

این خاصیت در آینده بسیار مورد کاربرد قرار خواهد گرفت.

اکنون به راحتی می‌توان برای N ماتریس $A^{(n)} \in \mathbb{R}^{I_n \times J_n}$ ، $n = 1, \dots, N$ و تانسور

تعریف کرد:

$$\mathcal{G} \times \{\mathbf{A}\} \triangleq \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 \dots \times_N \mathbf{A}^{(N)} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N} \quad (25-3)$$

همچنین نماد زیر را هم خواهیم داشت:

$$\mathcal{G} \times_{-n} \{\mathbf{A}\} \triangleq \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \dots \times_{n-1} \mathbf{A}^{(n-1)} \times_{n+1} \mathbf{A}^{(n+1)} \times_{n+2} \dots \times_N \mathbf{A}^{(N)} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times J_n \times I_{n+1} \times \dots \times I_N} \quad (26-3)$$

۳-۱-۴-۳ ضرب‌های ماتریسی مهم در جبر تانسوری

تعریف ۳-۱۲. (ضرب کرونیکر) ضرب کرونیکر^{۱۷} دو ماتریس $B \in \mathbb{R}^{K \times L}$ و $A \in \mathbb{R}^{I \times J}$ یک ماتریس است و به وسیله‌ی $A \otimes B$ نمایش داده و از رابطه‌ی زیر محاسبه می‌شود:

^{۱۷}Kronecker product

(۲۷-۳)

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \dots & a_{1J}B \\ a_{21}B & a_{22}B & \dots & a_{2J}B \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}B & a_{I2}B & \dots & a_{IJ}B \end{bmatrix} = [a_1 \otimes b_1 \quad a_1 \otimes b_2 \quad a_1 \otimes b_3 \quad \dots \quad a_J \otimes b_{L-1} \quad a_J \otimes b_L]$$

تعریف ۳-۱۳. (ضرب *Khatri-Rao*) ضرب میان دو ماتریس $B \in \mathbb{R}^{J \times K}$ و $A \in \mathbb{R}^{I \times K}$ *Khatri-Rao* ضرب میان دو ماتریس

به صورت زیر محاسبه می‌شود:

$$A \odot B = [a_1 \otimes b_1 \quad a_2 \otimes b_2 \quad \dots \quad a_K \otimes b_K] \quad (28-3)$$

اگر a و b دو بردار باشند، ضرب کرونیکر و آن‌ها برابر است.

تعریف ۳-۱۴. (ضرب هادامارد) ضرب هادامارد^{۱۸} دو ماتریس $A, B \in \mathbb{R}^{I \times J}$ ضرب

درایه-به-درایه^{۱۹} آن‌هاست:

$$A \circledast B = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} & \dots & a_{1J}b_{1J} \\ a_{21}b_{21} & a_{22}b_{22} & \dots & a_{2J}b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}b_{I1} & a_{I2}b_{I2} & \dots & a_{IJ}b_{IJ} \end{bmatrix} \quad (29-3)$$

۴-۱-۴-۳ رتبه‌ی تانسور

تعریف ۳-۱۵. (تانسور رتبه-یک) تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ را رتبه-یک^{۲۰} گویند اگر بتوان آن را

به صورت ضرب خارجی N بردار نوشت، یعنی:

$$\mathcal{X} = a^{(1)} \circ a^{(2)} \circ \dots \circ a^{(N)} \quad (30-3)$$

شکل ۳-۸ نمایش یک تانسور مرتبه سه از رتبه-یک است.

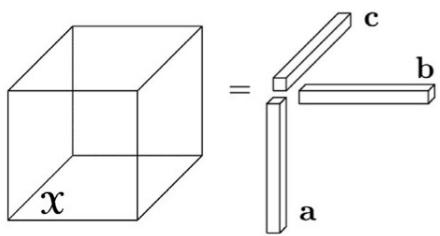
تعریف ۳-۱۶. (رتبه‌ی تانسور) رتبه‌ی^{۲۱} تانسور \mathcal{X} برابر حداقل تعداد تانسورهای رتبه یک است که از مجموع آن‌ها \mathcal{X} به دست می‌آید. اگر بتوانیم رتبه‌ی یک تانسور را دقیق محاسبه کنیم، می‌توانیم تعداد

^{۱۸}Hadamard product

^{۱۹}Elementwise

^{۲۰}Rank-1

^{۲۱}Rank



شکل ۳-۸: تانسور مرتبه سه‌ی رتبه‌یک [۳].

مولفه‌های تجزیه‌ی CP آن (بخش ۳-۴-۳) را به دست آوریم. اما روش مستقیمی برای به دست آوردن رتبه‌یک تانسور وجود ندارد چرا که یک مسأله‌ی *NP-hard* است [۳]. در عمل رتبه‌یک تانسور به صورت عددی و به وسیله‌ی برازش^{۲۲} تعداد زیادی تجزیه‌ی CP به آن به دست می‌آید (رجوع شود به بخش ۳-۴-۳). تلاش‌های متعددی برای به دست آوردن کران بالا برای رتبه‌ی تانسور در حالات خاص انجام شده است و تعاریف جنبی زیادی برای رتبه‌ی تانسور ارائه شده است که بررسی آن‌ها در حوصله‌ی این نوشتار نیست. برای جزئیات رجوع شود به [۳].

۵-۱-۴-۳ خواص مهم

در این قسمت تعدادی از خواص مهم در جبر تانسوری را بدون اثبات معرفی می‌کنیم [۷].

لم ۳-۱. برای دو تانسور $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ داریم:

$$\|\mathcal{X} - \mathcal{Y}\|^2 = \|\mathcal{X}\|^2 - 2\langle \mathcal{X}, \mathcal{Y} \rangle + \|\mathcal{Y}\|^2 \quad (31-3)$$

لم ۳-۲. فرض کنید $A \in \mathbb{R}^{J \times K}$ و $\mathcal{Y} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times K \times I_{n+1} \times \dots \times I_N}$ و $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$ در این صورت

$$\langle \mathcal{X}, \mathcal{Y} \times_n A \rangle = \left\langle \mathcal{X} \times_n A^T, \mathcal{Y} \right\rangle \quad (32-3)$$

لم ۳-۳. تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ را در نظر بگیرید و فرض کنید $Q \in \mathbb{R}^{J \times I_n}$ یک ماتریس متعامد^{۲۳} یکه است. در این صورت داریم:

$$\|\mathcal{X} \times_n Q\| = \|\mathcal{X}\| \quad (33-3)$$

لم ۳-۴. فرض کنید $A \in \mathbb{R}^{I \times L}$ و $B \in \mathbb{R}^{J \times L}$ در این صورت:

^{۲۲}Fit

^{۲۳}Orthonormal

$$(\mathbf{A} \odot \mathbf{B})^\dagger = ((\mathbf{A}^T \mathbf{A}) \circledast (\mathbf{B}^T \mathbf{B}))^\dagger (\mathbf{A} \odot \mathbf{B})^T \quad (34-3)$$

که در آن \dagger علامت شبهمعکوس^{۲۴} است.

۲-۴-۳ تجزیه‌ی Tucker

تجزیه‌ی Tucker در سال ۱۹۶۳ به وسیله‌ی معرفی شد. این تجزیه تانسور^{۲۵}

را به یک تانسور^{۲۶} هسته و یک ماتریس در هر مُد تجزیه می‌کند:

$$\mathcal{X} \approx \mathcal{G} \times \{\mathbf{A}\} = \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \dots \times_N \mathbf{A}^{(N)} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \dots \sum_{r_N=1}^{R_N} g_{r_1, r_2, \dots, r_N} \mathbf{a}_{r_1}^{(1)} \circ \mathbf{a}_{r_2}^{(2)} \circ \dots \circ \mathbf{a}_{r_N}^{(N)} \quad (35-3)$$

که در آن ماتریس‌های عامل^{۲۶} تجزیه به صورت زیر تعریف می‌شوند:

$$\mathbf{A}^{(n)} = [\mathbf{a}_1^{(n)}, \dots, \mathbf{a}_{R_n}^{(n)}] \in \mathbb{R}^{I_n \times R_n}, \quad n = 1, \dots, N$$

و تانسور^{۲۷} $\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times \dots \times R_N}$ تانسور هسته نامیده می‌شود. رابطه‌ی ۳۵-۳ را تجزیه‌ی

راتبه- (R_1, \dots, R_N) می‌گویند که در آن باید $R_n \leq I_n$. در برخی موارد اگر $I_n < R_n$ از تانسور هسته به

عنوان نسخه‌ی فشرده شده‌ی تانسور \mathcal{X} یاد می‌شود. تجزیه‌ی Tucker را به صورت اختصاری با استفاده از

عملگر^{۲۸} Tucker نمایش می‌دهند:

$$\mathcal{X} \approx [\![\mathcal{G}; \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}]\!] \quad (36-3)$$

شکل ۳-۹ نمایش تجزیه‌ی Tucker یک تانسور مرتبه سه است.

رابطه‌ی ۳۵-۳ را می‌توان به صورت درایه‌ای به شکل زیر بازنویسی کرد:

$$x_{i_1 i_2 \dots i_N} \approx \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \dots \sum_{r_N=1}^{R_N} g_{r_1 r_2 \dots r_N} a_{i_1 r_1}^{(1)} a_{i_2 r_2}^{(2)} \dots a_{i_N r_N}^{(N)} \quad (37-3)$$

همچنین رابطه‌ی زیر برای هر $n = 1, \dots, N$ برقرار است:

$$\mathbf{X}_{(n)} \approx \mathbf{A}^{(n)} \mathbf{G}_{(n)} \left(\mathbf{A}^{(N)} \otimes \dots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \dots \otimes \mathbf{A}^{(1)} \right)^T \quad (38-3)$$

بر اساس این رابطه می‌توان مشتق $\mathbf{X}_{(n)}$ را نسبت به $\mathbf{A}^{(n)}$ به صورت زیر نوشت:

$$\frac{d\mathbf{X}_{(n)}}{d\mathbf{A}^{(n)}} = \left[\left(\mathbf{A}^{(N)} \otimes \dots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \dots \otimes \mathbf{A}^{(1)} \right) \mathbf{G}_{(n)}^T \right] \otimes \mathbf{I}_{I_n} \quad (39-3)$$

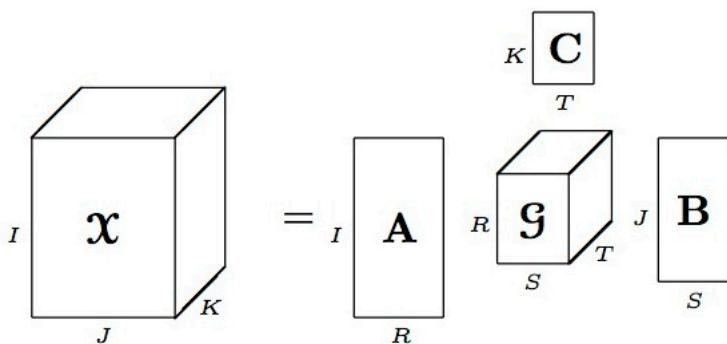
که در آن \mathbf{I}_{I_n} ماتریس یکانی $I_n \times I_n$ است.

^{۲۴}Pseudoinverse

^{۲۵}Core Tensor

^{۲۶}Factor matrix

^{۲۷}Operator



شکل ۳-۹: تجزیه‌ی Tucker بر روی تانسور $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ به صورت $\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{C}$ رتبه- (R, S, T) .

هرچند در مقاله‌ی اصلی Tucker [۱۹] شرط تعامد برای فاکتورهای تجزیه‌ی Tucker در نظر گرفته شده است، می‌توان شروطی دیگری همچون نامنفی بودن را به فاکتورها و تانسور هسته (NTD) و یا برخی از فاکتورها و (یا) برخی از درایه‌های تانسور هسته اعمال کرد (semi-NTD) [۶]. اکنون اگر بخواهیم تانسور هسته را بر حسب تانسور \mathcal{X} به دست آوریم داریم:

$$\mathcal{G} = \mathcal{X} \times \mathbf{A}^{(1)\dagger} \times_1 \cdots \times_N \mathbf{A}^{(N)\dagger} \quad (40-3)$$

روشن است که اگر ماتریس‌های عامل متعامد باشند، شبهمعکوس آن‌ها برابر ترانهاده‌ی آن‌ها خواهد بود. در این نوشتار برای سهولت بیشتر ماتریس‌های عاملی که متعامد هستند را با \mathbf{U} نشان می‌دهیم. در این صورت داریم:

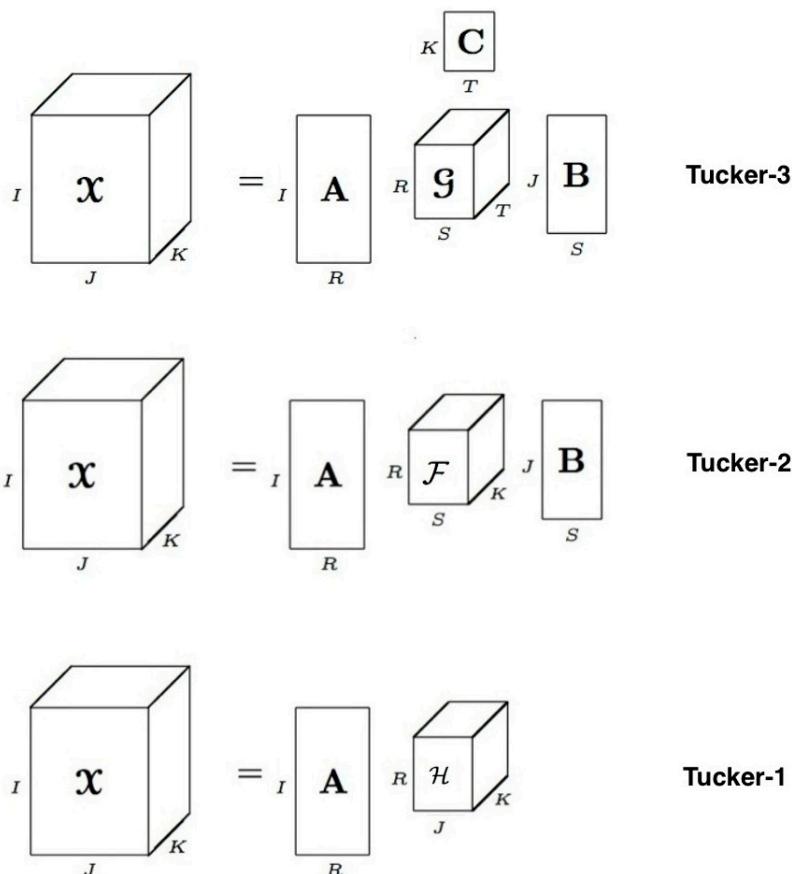
$$\mathcal{G} = \mathcal{X} \times \mathbf{U}^{(1)^T} \times_1 \cdots \times_N \mathbf{U}^{(N)^T} = [\mathcal{G}; \mathbf{U}^{(1)^T}, \dots, \mathbf{U}^{(N)^T}] \quad (41-3)$$

حالت کلی تجزیه‌ی Tucker یک تانسور مرتبه N به صورت $\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$ است (Tucker-M) یک عدد بین ۱ تا N است) که تانسور \mathcal{X} را به یک تانسور هسته و ماتریس‌های عامل در جهت M مُد آن تجزیه می‌کند. در واقع در این تجزیه M ماتریس عامل را در رابطه‌ی ۳۶-۳ برابر با I قرار می‌دهیم. برای مثال تانسور مرتبه سه $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ را در نظر بگیرید. تجزیه‌ی Tucker را برای این تانسور بر اساس رابطه‌ی ۳۵-۳ به صورت زیر می‌توان نوشت:

$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = [\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{C}] \quad (42-3)$$

این تجزیه را Tucker-3 می‌نامند [۶]. اما با ادغام یکی از ماتریس‌های عامل با تانسور هسته می‌توان آن را به Tucker-2 تبدیل کرد. برای مثال فرض کنید ماتریس C را در \mathcal{G} ادغام می‌کنیم. در این صورت با

^{۳۵}Nonnegative Tucker Decomposition



شکل ۳: تجزیه‌های Tucker-1,2,3 یک تانسور مرتبه سه.

تعریف ۲۰: داریم $\mathcal{F} = \mathcal{G} \times_3 C$

$$\mathcal{X} \approx \mathcal{F} \times_1 \mathbf{A} \times_2 \mathbf{B} = [\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{I}] \quad (43-3)$$

اگر یکی دیگر از ماتریس‌های عامل را در \mathcal{F} ادغام کنیم (مثلا \mathbf{B})، تجزیه‌ی Tucker-1 را خواهیم داشت:

$$\mathcal{X} \approx \mathcal{H} \times_1 \mathbf{A} = [\mathcal{G}; \mathbf{A}, \mathbf{I}, \mathbf{I}] \quad (44-3)$$

که در آن $\mathcal{H} = \mathcal{F} \times_2 \mathbf{B}$. شکل ۳: نمایشی از Tucker-1,2,3 روابط ۴۲-۳ تا ۴۴-۳ را نشان می‌دهد. تعمیم روابط ۴۲-۳ تا ۴۴-۳ به تانسور مرتبه N به این صورت است که می‌توان هر زیرمجموعه‌ی (N-M) تایی از ماتریس‌های عامل را در رابطه‌ی ۳۶-۳ برابر I قرار داد [۳]، به عبارتی آن‌ها را در تانسور هسته ادغام کرد.

۱-۲-۴-۳ تخمین ماتریس‌های عامل و تانسور هسته

تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ را در نظر بگیرید که می‌خواهیم آن را تجزیه‌ی Tucker رتبه- (R_1, \dots, R_N) کنیم و تانسور هسته \mathcal{G} و ماتریس‌های عامل $\mathbf{U}^{(n)}$ را به دست آوریم.^{۲۹} هدف ما انجام تجزیه با کمترین میزان خطا است. بنابراین مسئله‌ی بهینه‌سازی زیر را پیش رو خواهیم داشت:

$$\begin{aligned} & \min_{\mathcal{G}, \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}} \|\mathcal{X} - [\![\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\!] \| \\ & \text{s.t.} \quad \mathcal{G} \in \mathbb{R}^{R_1 \times \dots \times R_N} \end{aligned} \quad (45-۳)$$

$$\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n} \text{ orthonormal}, \quad n = 1, \dots, N$$

اکنون مسئله را تغییر می‌دهیم به طوری که \mathcal{G} از مسئله‌ی بهینه‌سازی حذف شود. فرض کنیم ماتریس‌های عامل را داریم در این صورت \mathcal{G} رابطه‌ی ۴۱-۳ به دست می‌آید. قضیه‌ی زیر را برای ساده‌سازی مسئله‌ی ۴۵-۳ خواهیم داشت.

قضیه ۱-۳. در صورتی که با فرض داشتن ماتریس‌های عامل، \mathcal{G} را از رابطه‌ی ۴۱-۳ محاسبه کنیم، مسئله‌ی بهینه‌سازی ۴۵-۳ به مسئله‌ی زیر تبدیل می‌شود:

$$\begin{aligned} & \max_{\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}} \|[\![\mathcal{X}; \mathbf{U}^{(1)^T}, \dots, \mathbf{U}^{(N)^T}]\!] \| \\ & \text{s.t.} \quad \mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n} \text{ orthonormal}, \quad n = 1, \dots, N \end{aligned} \quad (46-۳)$$

اثبات: با استناد به لم ۱-۳ داریم:

$$\begin{aligned} \|\mathcal{X} - [\![\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\!] \| &= \\ & \| \mathcal{X} \| ^2 - 2 \left\langle \mathcal{X}, [\![\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\!] \right\rangle + \| [\![\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\!] \| ^2 \end{aligned} \quad (47-۳)$$

همچنین از لم ۲-۳ داریم:

$$\left\langle \mathcal{X}, [\![\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\!] \right\rangle = \left\langle \mathcal{G}, [\![\mathcal{X}; \mathbf{U}^{(1)^T}, \dots, \mathbf{U}^{(N)^T}]\!] \right\rangle = \|\mathcal{G}\|^2 \quad (48-۳)$$

و در نهایت لم ۳-۳ نتیجه می‌دهد:

$$\| [\![\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\!] \| ^2 = \|\mathcal{G}\|^2 \quad (49-۳)$$

بنابراین داریم:

$$\|\mathcal{X} - [\![\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\!] \| = \|\mathcal{X}\|^2 - \|\mathcal{G}\|^2 \quad (50-۳)$$

پس مسئله‌ی ۴۵-۳ معادل بیشینه کردن $\|\mathcal{G}\|$ (مسئله‌ی ۴۶-۳) خواهد بود. \square

^{۲۹} این قسمت مشابه با منبع [۷] نوشته شده است.

با توجه قضیه‌ی بالا دو روش برای تخمین ماتریس‌های عامل وجود دارد که در ادامه توضیح داده خواهد شد.

الف - روش HOSVD

ایده‌ی اصلی پشت این روش یافتن بیشتر واریانس^{۳۰} در هر مُد مستقل از مُدهای دیگر است. این کار دقیقاً به معنی تجزیه‌ی Tucker-1 در جهت مُد n است. پس مسأله‌ی بهینه‌سازی^{۴۶-۳} به صورت زیر برای تخمین ماتریس عامل مُد n مطرح می‌شود:

$$\begin{aligned} \max_{\mathbf{U}^{(n)}} & \|[\mathcal{X}; \mathbf{I}, \dots, \mathbf{I}, \mathbf{U}^{(n)T}, \mathbf{I}, \dots, \mathbf{I}] \| \\ \text{s.t.} & \quad \mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n} \text{ orthonormal} \end{aligned} \quad (51-۳)$$

تابع هزینه در مسأله‌ی بالا معادل $\|\mathcal{X} \times_n \mathbf{U}^{(n)T}\| = \|\mathbf{U}^{(n)T} \mathbf{X}_{(n)}\|$ است. از جبر ماتریسی می‌دانیم که R_n بردار تکین چپ $\mathbf{X}_{(n)}$ جواب بهینه‌ی $\mathbf{U}^{(n)}$ است. اگر به این روش ماتریس عامل را برای هر $n = 1, \dots, N$ محاسبه کنیم در واقع در مُدهای مختلف SVD به کار بردایم؛ به همین دلیل به این روش گفته می‌شود. الگوریتم ۱ شبکه کد HOSVD (Higher Order SVD) را نمایش می‌دهد.

الگوریتم ۱ الگوریتم HOSVD برای محاسبه‌ی تجزیه‌ی Tucker رنک- (R_1, \dots, R_N) تansور $[\mathcal{X}] \in \mathbb{R}^{I_1 \times \dots \times I_N}$.

Input: $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ and R_1, \dots, R_N

Output: $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}$ and $\mathcal{G} \in \mathbb{R}^{R_1 \times \dots \times R_N}$

-
- 1: **for** $n = 1, \dots, N$ **do**
 - 2: $\mathbf{U}^{(n)} \leftarrow R_n$ leading left singular vectors of $\mathbf{X}_{(n)}$
 - 3: **end for**
 - 4: $\mathcal{G} = [\mathcal{X}, \mathbf{U}^{(1)T}, \dots, \mathbf{U}^{(N)T}]$
-

ب - روش HOOI

در صورتی که نخواهیم استقلال بین ماتریس‌های عامل مُدهای مختلف در نظر بگیریم، باید مسأله‌ی ۴۶-۳ را یک جا حل کنیم. اما این کار به دلیل زیاد بودن تعداد پارامترها ممکن نیست. به همین دلیل از روش ALS^{۳۱} استفاده می‌شود که در آن به صورت جایگزینی^{۳۲} عمل می‌شود. یعنی به نوبت برای هر مُد فرض کنیم ماتریس‌های عامل سایر مُدها را داریم و مسأله را به صورت تکراری برای مُدهای مختلف حل

^{۳۰}Variation

^{۳۱}Alternating Least Squares

^{۳۲}Alternating

کنیم. اکنون برای یک n خاص فرض کنید می‌خواهیم $\mathbf{U}^{(n)}$ را با روش فوق تخمین بزنیم. بنابراین مسأله‌ی ۴۶-۳ به صورت زیر در می‌آید:

$$\begin{aligned} & \max_{\mathbf{U}^{(n)}} \|[\mathcal{X}; \mathbf{U}^{(1)^T}, \dots, \mathbf{U}^{(N)^T}] \| \\ \text{s.t. } & \mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n} \text{ orthonormal} \end{aligned} \quad (۵۲-۳)$$

حال اگر تعریف کنیم $\mathcal{Z} = [\mathcal{X}; \mathbf{U}^{(1)^T}, \dots, \mathbf{U}^{(n-1)^T}, \mathbf{I}, \mathbf{U}^{(n+1)^T}, \dots, \mathbf{U}^{(N)^T}]$ ، مسأله به صورت زیر ساده می‌شود:

$$\begin{aligned} & \max_{\mathbf{U}^{(n)}} \|\mathcal{Z} \times_n \mathbf{U}^{(n)^T}\| = \|\mathbf{U}^{(n)^T} \mathbf{Z}_{(n)}\| \\ \text{s.t. } & \mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n} \text{ orthonormal} \end{aligned} \quad (۵۳-۳)$$

که R_n بردار تکین چپ $\mathbf{Z}_{(n)}$ جواب بهینه‌ی $\mathbf{U}^{(n)}$ خواهد بود. از آن‌جا که این الگوریتم تکراری است، آن را (HOOI) Higher Order Orthogonal Iterative را نشان می‌دهد.

الگوریتم ۲ الگوریتم HOOI. الگوریتم ALS برای محاسبه‌ی تجزیه‌ی تجزیه‌ی Tucker رنک- (R_1, \dots, R_N) تansور $[\gamma] \mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$

Input: $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ and R_1, \dots, R_N

Output: $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}$ and $\mathcal{G} \in \mathbb{R}^{R_1 \times \dots \times R_N}$

- 1: Initialize $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}$ via HOSVD (algorithm 1)
 - 2: **while** not converged **do**
 - 3: **for** $n = 1, \dots, N$ **do**
 - 4: $\mathcal{Z} = [\mathcal{X}; \mathbf{U}^{(1)^T}, \dots, \mathbf{U}^{(n-1)^T}, \mathbf{I}, \mathbf{U}^{(n+1)^T}, \dots, \mathbf{U}^{(N)^T}]$
 - 5: $\mathbf{U}^{(n)^T} \leftarrow R_n$ leading left singular vectors of $\mathbf{Z}_{(n)}$
 - 6: **end for**
 - 7: **end while**
 - 8: $\mathcal{G} = [\mathcal{X}, \mathbf{U}^{(1)^T}, \dots, \mathbf{U}^{(N)^T}]$
-

۳-۴-۳ CP تجزیه‌ی

تجزیه‌ی Caroll^{۳۳} PARAFAC در ۱۹۷۰ به وسیله‌ی Harshman معرفی شد. هم‌زمان با او Chang^{۳۴} نیز آن را با نام CANDECOMP معرفی کردند. هم‌چنین در برخی منابع این تجزیه را Canonical Polyadic Decomposition می‌گویند. به همین دلیل آن را CP که مخفف Canonical Polyadic Decomposition است نیز می‌خوانند [۳].

^{۳۳}PARAllel FACTor analysis

^{۳۴}CANonical DECOMPosition

این تجزیه تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ را به صورت مجموع R تانسور رتبه-یک تخمین می‌زند:

$$\mathcal{X} \approx \sum_{r=1}^R \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \dots \circ \mathbf{a}_r^{(N)} \quad (54-3)$$

که در آن برای هر R , $\mathbf{a}_r^{(n)} \in \mathbb{R}^{I_n}$, $r = 1, \dots, R$ و هر درایه‌ی تانسور \mathcal{X} بر اساس درایه‌های بردارهای $\mathbf{a}_r^{(n)}$ به صورت زیر خواهد بود:

$$\mathcal{X}(i_1, i_2, \dots, i_N) \approx \sum_{r=1}^R \mathbf{a}_r^{(1)}(i_1) \mathbf{a}_r^{(2)}(i_2) \dots \mathbf{a}_r^{(N)}(i_N), \quad \forall n = 1, \dots, N : i_n \in \{1, \dots, I_n\} \quad (55-3)$$

در این حالت گفته می‌شود که تجزیه‌ی CP با R مولفه انجام شده است. اکنون اگر برای هر $n = 1, \dots, N$ بردارهای $\mathbf{A}^{(n)} = [\mathbf{a}_1^{(n)} \quad \mathbf{a}_2^{(n)} \quad \dots \quad \mathbf{a}_R^{(n)}] \in \mathbb{R}^{I_n \times R}$ در ستون‌های یک ماتریس قرار دهیم، N ماتریس خواهیم داشت که آن‌ها را ماتریس‌های عامل ^{۳۵} تجزیه‌ی CP می‌نامیم. در این صورت رابطه‌ی ۵۴-۳ به صورت اختصاری به شکل زیر نوشته می‌شود [۷]:

$$\mathcal{X} \approx [\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}] \equiv \sum_{r=1}^R \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \dots \circ \mathbf{a}_r^{(N)} \quad (56-3)$$

عملگر $[\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}]$ را kruskal [۷] عملگر Tucker به صورت زیر است:

$$[\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}] = [\mathcal{I}; \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}] \quad (57-3)$$

که در آن \mathcal{I} تانسور یکانی هم مرتبه با \mathcal{X} است که اندازه‌ی مُد آن R است. شکل ۱۱-۳ تجزیه‌ی CP برای یک تانسور مرتبه سه نشان می‌دهد.

در بسیاری موارد ستون‌های ماتریس‌های عامل نرمالیزه می‌شوند تا یکه شوند و وزن‌های متناظر با اندازه‌های آن‌ها روی سوپرقطر تانسور سوپرقطری \mathcal{L} قرار داده می‌شود. \mathcal{L} هم مرتبه با \mathcal{X} است و اندازه‌ی هر مُد آن R است. در این صورت تجزیه‌ی CP به صورت زیر خواهد بود:

$$\mathcal{X} \approx [\mathcal{L}; \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}] \equiv \sum_{r=1}^R \lambda_r \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \dots \circ \mathbf{a}_r^{(N)} \quad (58-3)$$

اگر بردار λ را در قطر ماتریس Λ قرار دهیم فرم ماتریسی شده‌ی n تانسور \mathcal{X} به صورت زیر خواهد بود:

$$\mathbf{X}_{(n)} \approx \mathbf{A}^{(n)} \Lambda \left(\mathbf{A}^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \dots \mathbf{A}^{(1)} \right)^T \quad (59-3)$$

^{۳۵}Factor matrix

$$\mathcal{X} = \mathbf{a}_1 \mathbf{b}_1 \mathbf{c}_1 + \mathbf{a}_2 \mathbf{b}_2 \mathbf{c}_2 + \dots + \mathbf{a}_R \mathbf{b}_R \mathbf{c}_R$$

شکل ۱۱-۳: تجزیه‌ی CP تانسور مرتبه سه با R مولفه به صورت $\llbracket A, B, C \rrbracket$

به تخمینی بودن رابطه‌ی ۵۴-۳ دقت کنید: اگر \mathcal{X} یک تانسور با رتبه‌ی R باشد در این صورت رابطه‌ی فوق دقیق خواهد بود. اما همان‌طور که قبلاً گفته شد، محاسبه‌ی رتبه‌ی یک تانسور روش مستقیمی ندارد، به همین دلیل تجزیه‌ی CP در حالت کلی یک تخمین از \mathcal{X} است.^{۳۶} به طور کلی تجزیه‌ی CP شرطی روی ماتریس‌های عامل ندارد. اما شروطی همچون نامنفی بودن^{۳۷} NTF را قابل اعمال هستند. در صورتی که شرط نامنفی بودن به فاکتورها اعمال شود، تجزیه‌ی CP را می‌خوانند [۶].

۱-۳-۴-۳ تخمین ماتریس‌های عامل

همان‌طور که قبلاً گفتیم، نمی‌توان رتبه‌ی یک تانسور را به طور دقیق به دست آورد. بنابراین روش مستقیمی برای به دست آوردن تعداد مولفه‌های CP وجود ندارد. یکی از روش‌های مرسوم برای داده‌های با نویز کم برآش مدل‌های Rank-R CP است به این صورت که تانسور مورد نظر را به ازای $R = 1, 2, 3, \dots$ تجزیه‌ی CP با R مولفه کنیم تا بهترین برآش را به دست آوریم. اما این روش کمی خطرناک است! چرا که برخی از تانسورها دارای تخمین‌های رتبه‌ی پایین^{۳۸} هستند و ممکن است در طی تخمین رتبه‌ی تانسور با این روش، به اشتباه رتبه‌ی آن کمتر تخمین زده شود. همچنین برای داده‌های نویزی این معیار به تنها‌ی کارساز نیست، به همین دلیل روش‌های دیگری همچون CORCONDIA [۲۱] نیز برای مقایسه‌ی تعداد مولفه‌های مختلف برای CP ارائه شده‌اند که توضیح آن‌ها در حوصله‌ی این نوشتار نیست. در ادامه فرض می‌کنیم R را می‌دانیم.

مشهورترین الگوریتمی که برای محاسبه‌ی CP ارائه شده است از ALS استفاده می‌کند و با هدف

^{۳۶}Nonnegativity

^{۳۷}Nonnegative Tensor Factorization

^{۳۸}Low rank approximation

کمینه کردن نرم خطای تجزیه‌ی CP انجام می‌شود.

تansور $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ را در نظر بگیرید.^{۳۹} مسأله‌ی بهینه‌سازی زیر مطرح است:

$$\min_{\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}} \|\mathcal{X} - [\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}]\| \quad (60-3)$$

که در آن $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times R}$. مسأله‌ی فوق را باید با روش ALS حل کنیم. پس تابع هدف مسأله‌ی ۶۰-۳ را با استفاده از رابطه‌ی ۵۹-۳ بازنویسی کرده تا بتوانیم برای n خاص ماتریس عامل $\mathbf{A}^{(n)}$ را با فرض دانستن سایر ماتریس‌های عامل دیگر به دست آوریم:

$$\|\mathcal{X} - [\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}]\| = \|\mathcal{X}_{(n)} - \mathbf{A}^{(n)} \left(\mathbf{A}^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \dots \mathbf{A}^{(1)} \right)^T\| \quad (61-3)$$

پس مسأله‌ی ۶۰-۳ به صورت زیر خواهد شد:

$$\min_{\mathbf{A}^{(n)}} \|\mathbf{X}_{(n)} - \mathbf{A}^{(n)} \left(\mathbf{A}^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \dots \mathbf{A}^{(1)} \right)^T\| \quad (62-3)$$

با استفاده از لم ۴-۳ می‌توان نوشت:

$$\begin{aligned} \mathbf{A}^{(n)^T} &= \left(\mathbf{A}^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \dots \mathbf{A}^{(1)} \right)^\dagger \mathbf{X}_{(n)}^T \\ &= \mathbf{V}^\dagger \left(\mathbf{A}^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \dots \mathbf{A}^{(1)} \right)^T \mathbf{X}_{(n)}^T \end{aligned} \quad (63-3)$$

که در آن

$$\mathbf{V} = (\mathbf{A}^{(N)^T} \mathbf{A}^{(N)}) \circledast \dots \circledast (\mathbf{A}^{(n+1)^T} \mathbf{A}^{(n+1)}) \circledast (\mathbf{A}^{(n-1)^T} \mathbf{A}^{(n-1)}) \circledast \dots \circledast (\mathbf{A}^{(1)^T} \mathbf{A}^{(1)}) \quad (64-3)$$

در الگوریتم ۳ شبکه‌کد این الگوریتم ارائه شده است. این الگوریتم تعداد مولفه‌های R و مقادیر اولیه‌ی $\mathbf{A}^{(n)}$ را به عنوان ورودی دریافت می‌کند که می‌توانیم آن‌ها را به صورت تصادفی یا با HOSVD مقداردهی کرد.

۳-۵ روشهای تانسوری کاهش ویژگی

۱-۵-۳ فرمول‌بندی کلی روشهای کاهش ویژگی تانسوری

مسأله ۱.۳. (کاهش ویژگی و کاهش بُعد ماتریس‌ها با استفاده از تجزیه‌ی تانسوری [۲۰]) فرض کنید $\{X^{(k)} \in \mathbb{R}^{I_1 \times I_2}, k = 1, \dots, K\}$ مجموعه‌ی دادگانی است که می‌خواهیم برای آن‌ها یک زیرفضا که

^{۳۹} این قسمت مشابه با منبع [۷] نوشته شده است.

الگوریتم ۳ شبکه‌کد الگوریتم ALS برای محاسبه‌ی تجزیه‌ی CP با R مولفه [۷]

Input: $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ and R

Output: $A^{(1)}, \dots, A^{(N)}$ such that $\mathcal{X} \approx \llbracket A^{(1)}, \dots, A^{(N)} \rrbracket$

```

1: for  $n = 1, \dots, N$  do
2:   Initialize  $A^{(n)}$  in some way (e.g. random or HOSVD)
3:   Normalize columns of  $A^{(n)}$ 
4:    $B^{(n)} \leftarrow A^{(n)T} A^{(n)}$ 
5: end for
6: while not converged do
7:   for  $n = 1, \dots, N$  do
8:      $V \leftarrow B^{(N)} \circledast \dots \circledast B^{(n+1)} \circledast B^{(n+1)} \circledast \dots \circledast B^{(1)}$ 
9:      $A^{(n)} \leftarrow X_{(n)}(A^{(N)} \odot \dots \odot A^{(n+1)} \odot \dots \odot A^{(n-1)} \odot \dots \odot A^{(1)})^T V^\dagger$ 
10:    if  $n \neq N$  then
11:      Normalize columns of  $A^{(n)}$ 
12:    end if
13:    Set  $B^{(n)} \leftarrow A^{(n)T} A^{(n)}$ 
14:  end for
15: end while

```

به وسیله‌ی دو ماتریس $A^{(1)}$ و $A^{(2)}$ تعریف می‌شود، به دست آوریم و با تصویر $X^{(k)}$ ها بر آن کاهش ویژگی انجام دهیم. برای این کار آن‌ها را به طور همزمان تجزیه می‌کنیم:

$$X^{(k)} = A^{(1)} F^{(k)} A^{(2)T}, \quad k = 1, \dots, K \quad (65-3)$$

که در آن دو ماتریس $A^{(1)} \in \mathbb{R}^{I_1 \times J_1}$ و $A^{(2)} \in \mathbb{R}^{I_2 \times J_2}$ ماتریس‌های پایه‌ای هستند که $X^{(k)}$ ها را توصیف می‌کنند و ویژگی‌های استخراج شده در ماتریس‌های $F^{(k)} \in \mathbb{R}^{J_1 \times J_2}$ قرار می‌گیرند که $I_1 \ll J_1$ و $J_2 \ll I_2$

یک روش برای حل این مسئله کمینه‌سازی همزمانِ تابع هزینه‌ی $\|X^{(k)} - A^{(1)} F^{(k)} A^{(2)T}\|_F^2$ برای $k = 1, \dots, K$ به صورت زنجیروار است. اما یک روش جایگزین هم وجود دارد و آن استفاده از تجزیه‌های تانسوری است.

اگر $F^{(k)}$ ها را در اسلایس‌های تانسور $\mathcal{F} \in \mathbb{R}^{J_1 \times J_2 \times K}$ و $X^{(k)}$ ها را در اسلایس‌های تانسور $Tucker-2$ قرار دهیم، می‌توان نشان داد که مسئله‌ی ۱.۳ معادل تجزیه‌ی $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ است:

$$\mathcal{X} \approx \mathcal{F} \times_1 A^{(1)} \times_2 A^{(2)} \quad (66-3)$$

در حالت خاص اگر فرض کنیم $F^{(k)} = diag\{f^{(k)}\} \in \mathbb{R}^{J \times J}$ ساده می‌شود:

$$\mathcal{X} \approx \mathcal{I} \times_1 A^{(1)} \times_2 A^{(2)} \times_3 F \quad (67-3)$$

^{۴۰} این نماد به این معنی است که درایه‌های بردار $f^{(k)}$ روی قطر $F^{(k)}$ چیده شود.

که در آن $\mathbf{F} = \left[\mathbf{f}^{(1)}, \mathbf{f}^{(2)}, \dots, \mathbf{f}^{(K)} \right]^T$ و $\mathcal{I} \in \mathbb{R}^{J \times J \times J}$ تانسور یکانی است.

بر ماتریس‌های $A^{(n)}$ و $F^{(k)}$ می‌توان شروط مختلفی اعمال کرد و مسئله را به حالات خاص تبدیل کرد. برای مثال اگر $A^{(n)}$ ها متعامد باشند و $F^{(k)}$ ها مثبت معین و قطری باشند به مسئله‌ی تجزیه‌ی تانسوری بالا، *Higher Order PCA* یا *HOSVD* گفته می‌شود.

اکنون می‌توانیم مسئله‌ی ۱.۳ را به صورت زیر تعمیم دهیم:

مسئله ۲.۳. (طبقه‌بندی داده‌های N بعدی [۲۰]) فرض کنید مجموعه‌ی داده‌های آموزش همراه با برچسب‌های آن‌ها $\{\mathcal{X}^{(k)} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}, k = 1, \dots, K\}$ از C کلاس در دست است. می‌خواهیم برچسب‌های داده‌های آزمون $\{\mathcal{Y}^{(m)} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}, m = 1, \dots, M\}$ را تعیین کنیم.

۱-۱-۵-۳ کاهش ویژگی با استفاده از تجزیه‌ی Tucker

اکنون می‌خواهیم مسئله‌ی ۲.۳ را با استفاده از تجزیه‌ی Tucker حل کنیم [۲۰]. اگر K تانسور $\mathcal{X}^{(k)} \approx \mathcal{G}^{(k)} \times_1 A^{(1)} \times_2 A^{(2)} \times_3 \dots \times_N A^{(N)} = \mathcal{G}^{(k)} \times \{A\}$ باشد که $k = 1, \dots, K$ را هم‌زمان تجزیه‌ی Tucker کنیم:

$$\mathcal{X}^{(k)} \approx \mathcal{G}^{(k)} \times_1 A^{(1)} \times_2 A^{(2)} \times_3 \dots \times_N A^{(N)} = \mathcal{G}^{(k)} \times \{A\}, \quad k = 1, \dots, K \quad (68-3)$$

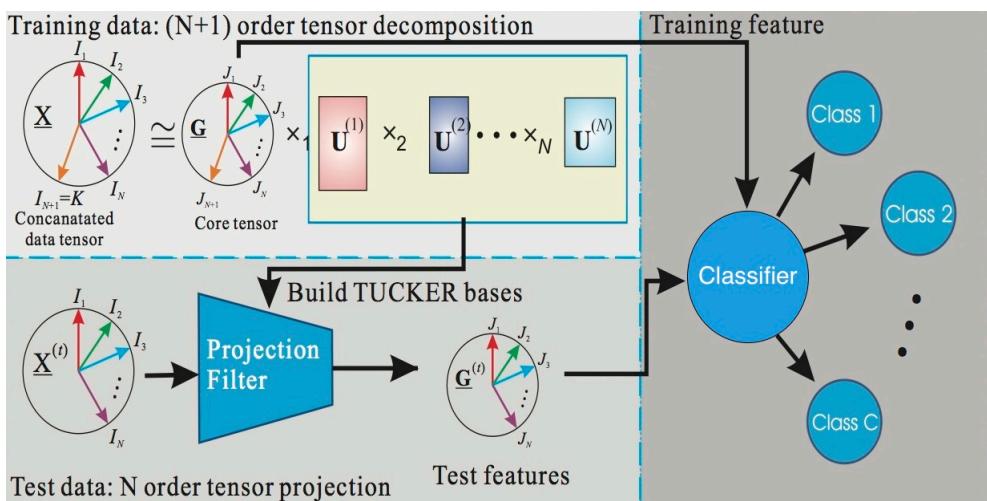
که در آن $\mathcal{G}^{(k)} \in \mathbb{R}^{J_1 \times \dots \times J_N}$ و $A^{(n)} \in \mathbb{R}^{I_n \times J_n}$ ویژگی‌های کاهش بعد یافته هستند. یعنی هر یک از $g_{j_1 \dots j_N}^{(k)}$ یک تکویژگی هستند که شدت ارتباط بین مولفه‌های پایه‌ی $a_{j_1}^{(1)}, a_{j_2}^{(2)}, \dots, a_{j_N}^{(N)}$ را مشخص می‌کند. می‌توان نشان داد که حل ۶۸-۳ معادل است با تجزیه‌ی Tucker-N تانسور \mathcal{X} که

$$\mathcal{X} \left(\underset{1}{\vdots}, \underset{2}{\vdots}, \dots, \underset{N}{\vdots}, \underset{N+1}{k} \right) = \mathcal{X}^{(k)}$$

$$\mathcal{X} \approx \mathcal{G} \times_1 A^{(1)} \times_2 A^{(2)} \times_3 \dots \times_N A^{(N)} = \mathcal{G} \times_{-(N+1)} \{A\} \quad (69-3)$$

$$\mathcal{G} \left(\underset{1}{\vdots}, \underset{2}{\vdots}, \dots, \underset{N}{\vdots}, \underset{N+1}{k} \right) = \mathcal{G}^{(k)}$$

در حالت خاص فرض کنید که ماتریس‌های پایه، یعنی $A^{(n)}$ ها، متعامدند. بر اساس قرارداد قبلی مان ماتریس‌های متعامد را با $U^{(n)}$ نشان می‌دهیم. در این صورت:



شکل ۳-۱۲: بلوك‌دیاگرام کاهش ویژگی و طبقه‌بندی با استفاده از تجزیه‌ی [۲۲] Tucker.

$$\mathcal{G} = \mathcal{X} \times_1 \mathbf{U}^{(1)^T} \times_2 \mathbf{U}^{(2)^T} \times_3 \dots \times_N \mathbf{U}^{(N)^T} = \mathcal{X} \times_{-(N+1)} \{\mathbf{U}^T\} \quad (70-3)$$

با فرض تعامد فاکتورهای پایه، می‌توانیم برای به دست آوردن فاکتورهای پایه و تansور هسته حاوی ویژگی‌ها از الگوریتم HOOI و اگر فرض نامنفی بودن را برای فاکتورها در نظر بگیریم، از NTD استفاده می‌کنیم [۲۰]. اگر در محاسبه‌ی ماتریس‌های پایه از برچسب‌های داده‌های آموزش استفاده شود، آنالیز تفکیکی^{۴۱} تansوری را خواهیم داشت که در زیربخش‌های بعدی به آن‌ها می‌پردازیم. در اینجا لازم است مفهوم از «تجزیه‌ی Tucker باسپرست و بدون سرپرست» توضیح داده شود. در اینجا وقتی از تجزیه‌ی باسپرست صحبت می‌کنیم منظور این است که از برچسب‌های داده‌های آموزش برای به دست آوردن ماتریس‌های زیرفضای Tucker استفاده شده است (یعنی قیدی بر تجزیه‌ی Tucker اعمال شده است که برچسب‌ها در آن دخیل هستند). اما در صورتی که از برچسب‌ها استفاده نکنیم می‌گوییم تجزیه‌ی بدون سرپرست انجام داده‌ایم. به عبارتی سرپرست داشتن تجزیه مربوط به این است که آیا برچسب‌ها را در تجزیه‌ی داده‌های آموزش دخالت داده‌ایم یا نه.

اکنون پس از به دست آوردن فضای پایه، کافی است داده‌های آزمون را روی فضای پایه‌ی تولید شده تصویر کنیم و ویژگی‌های آن‌ها را استخراج کنیم. فرض کنید $\mathcal{Y} = \text{cat}(N+1, \mathcal{Y}^{(1)}, \dots, \mathcal{Y}^{(M)})$ در^{۴۲} این صورت داریم:

$$\mathcal{F} = \mathcal{Y} \times_{-(N+1)} \{A^\dagger\} \quad (71-3)$$

که در آن \mathcal{F} در بعد $N+1$ خود ویژگی‌های داده‌های آزمون را دارد. یعنی برای هر $m = 1, \dots, M$ ویژگی

^{۴۱}Discriminant analysis

^{۴۲}این نماد به معنی این است که تansورهای $\mathcal{Y}^{(m)}$ را در بعد $(N+1)$ پشت سر هم قرار دهیم.

داده‌ی m ام به صورت $\mathcal{F}^{(m)} = \mathcal{F} \left(\begin{smallmatrix} ; & \cdots & ; \\ ; & N & ; \\ ; & & N+1 \end{smallmatrix} \right)$ است.

شکل ۱۲-۳ بلوک‌دیاگرام کاهش ویژگی و طبقه‌بندی با استفاده از تجزیه‌ی Tucker را نشان می‌دهد.

۲-۱-۵-۳ کاهش ویژگی با استفاده از تجزیه‌ی CP

در مراجع از تجزیه‌ی CP به صورت بدون سرپرست (بدون استفاده از زیرفضای داده‌های آموزش) استفاده شده است [۲۳] که در بخش ۳-۴ بررسی خواهد شد. در واقع تفاوت استفاده از CP و Tucker برای استخراج ویژگی با هدف طبقه‌بندی این است که در روش مبتنی بر Tucker تansور داده‌های آموزش تجزیه شده و زیرفضایی را به عنوان زیرفضای ویژگی به دست می‌دهد؛ سپس تansورهای داده‌های آزمون بر روی ماتریس‌های پایه‌ی این زیرفضا تصویر می‌شوند. در حالی که برای استفاده از CP هم داده‌های آموزش هم آزمون تجزیه می‌شوند؛ سپس با ویژگی‌های استخراج شده از داده‌های آموزش طبقه‌بند آموزش داده می‌شود.

پس در این قسمت این تغییر را در مسئله‌ی ۲.۳ می‌دهیم که فقط می‌خواهیم داده‌های $\{\mathcal{X}^{(k)} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}, k = 1, \dots, K\}$ را کاهش ویژگی دهیم. این داده‌ها می‌توانند داده‌های آموزش یا آزمون باشند. برای این کار ابتدا داده‌ها را در $M^N + 1$ ام تansور $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N \times K}$ می‌چینیم (یعنی مولفه تجزیه می‌کنیم):

$$\mathcal{X} \approx \mathcal{I} \times_1 A^{(1)} \times_2 A^{(2)} \times_3 \dots \times_N A^{(N)} \times_{N+1} F \quad (72-3)$$

در این حالت ماتریس $F \in \mathbb{R}^{K \times R}$ حاوی ویژگی‌های کاهش بعد یافته از داده‌های آموزش است^{۴۳} (فضای ویژگی R بُعد دارد).

۲-۵-۳ نمادگذاری

در ادامه‌ی این بخش به مرور برخی از روش‌های آنالیز تفکیکی که بر روی داده‌های تansوری اعمال می‌شوند می‌پردازیم. تمامی این روش‌ها از تجزیه‌ی Tucker با ماتریس‌های عامل متعامد برای حل مسئله‌ی ۲.۳ بهره برده‌اند. برای توضیح هریک از روش‌ها از نمادگذاری مشابهی استفاده می‌کنیم (حتی اگر در مقالات مربوطه نمادگذاری متفاوتی وجود داشته باشد). بدین منظور مرتبه‌ی تansورهای داده‌های آموزش را با N نشان می‌دهیم و از اندیس n برای اشاره به M^n ام تansور استفاده می‌کنیم.

^{۴۳} مراجعه شود به رابطه‌ی ۶۷-۳

$\bar{\mathcal{G}}^c = \frac{1}{K_c} \sum_{\mathcal{X}^{(k)} \in \Omega_c} \mathcal{G}^{(k)}$ میانگین تصویر $\mathcal{G}^{(k)} = \mathcal{X}^{(k)} \times \{U\}$

داده‌های کلاس c و $\bar{\mathcal{G}} = \frac{1}{K} \sum_{k=1}^K \mathcal{G}^{(k)}$ میانگین تصویرهای همهٔ داده‌های آموزش هستند.

HODA ۳-۵-۳

این روش در ادبیات به دو نام متفاوت معرفی شده است: در سال ۲۰۰۵ در [۲۴] به نام DATER^{۴۴} معرفی شد. در سال ۲۰۱۰ نیز در [۲۰] با نام HODA^{۴۵} بررسی شده است. می‌توان گفت HODA مبنای روش‌های دیگری در حوزه‌ی آنالیز تفکیکی تansوری است.

در این روش ماتریس‌های پایه‌ی متعامد به وسیله‌ی بیشینه کردن نسبت فیشر برای تansورهای

هسته $\mathcal{G}^{(i)}$ به دست می‌آیند:

$$\varphi = \underset{\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}}{\operatorname{argmax}} \frac{\sum_{c=1}^C K_c \|\bar{\mathcal{G}}^{(c)} - \bar{\mathcal{G}}\|_F^2}{\sum_{k=1}^K \|\mathcal{G}^{(k)} - \bar{\mathcal{G}}\|_F^2} \quad (73-3)$$

به همین دلیل می‌توان این روش را تعمیمی از روش LDA برداری به داده‌های تansوری دانست. مسئله‌ی ۷۳-۳ حل فرم بسته ندارد. برای حل آن از روش جایگزینی استفاده می‌شود. بدین صورت که برای به روز رسانی هر یک از ماتریس‌های پایه فرض می‌شود سایر ماتریس‌های پایه را داریم. پس یک n خاص را فرض کنید.

اکنون صورت رابطه‌ی ۷۳-۳ را (که در واقع معیار پراکندگی بین کلاسی است) به صورت trace ماتریس پراکندگی بین کلاسی بازنویسی می‌کنیم:

$$\begin{aligned} s_b &= \sum_{c=1}^C K_c \|\bar{\mathcal{G}}^{(c)} - \bar{\mathcal{G}}\|_F^2 = \sum_{c=1}^C K_c \left\| \left(\bar{\mathcal{X}}^{(c)} \times_{-n} \{\mathbf{U}^T\} \right) \times_n \mathbf{U}^{(n)T} - \left(\bar{\mathcal{X}} \times_{-n} \{\mathbf{U}^T\} \right) \times_n \mathbf{U}^{(n)T} \right\|_F^2 \\ &= \sum_{c=1}^C K_c \left\| \left(\bar{\mathcal{X}}^{(c)} - \bar{\mathcal{X}} \right) \times_{-n} \{\mathbf{U}^T\} \times_n \mathbf{U}^{(n)T} \right\|_F^2 = \sum_{c=1}^C \left\| \check{\mathcal{Z}}^{-n} \times_n \mathbf{U}^{(n)T} \right\|_F^2 \\ &= \sum_{c=1}^C \operatorname{tr} \{ \mathbf{U}^{(n)T} \left\langle \check{\mathcal{Z}}^{-n}, \check{\mathcal{Z}}^{-n} \right\rangle_{-n} \mathbf{U}^{(n)} \} = \operatorname{tr} \{ \mathbf{U}^{(n)T} \left(\sum_{c=1}^C \left\langle \check{\mathcal{Z}}^{-n}, \check{\mathcal{Z}}^{-n} \right\rangle_{-n} \right) \mathbf{U}^{(n)} \} \\ &= \operatorname{tr} \{ \mathbf{U}^{(n)T} \mathbf{S}_b^{-n} \mathbf{U}^{(n)} \} \end{aligned} \quad (74-3)$$

^{۴۴}Discriminant Analysis with Tensor Representation

^{۴۵}Higher Order Discriminant Analysis

که در آن $\bar{\mathcal{X}}$ و $\bar{\mathcal{X}}^{(c)}$ به ترتیب میانگین کل داده‌های آموزش و میانگین داده‌های کلاس c هستند و

$$\check{\mathcal{Z}}^{(c)} = \check{\mathcal{X}}^{(c)} \times_{-n} \{\mathbf{U}^T\} \quad (75-3)$$

$$\check{\mathcal{Z}}^{-n} = cat \left(N + 1, \check{\mathcal{Z}}^{(1)}, \dots, \check{\mathcal{Z}}^{(C)} \right) \quad (76-3)$$

$$\check{\mathcal{Z}}^{-n} = \check{\mathcal{X}} \times_{-(n, N+1)} \{\mathbf{U}^T\} \quad (77-3)$$

که

$$\check{\mathcal{X}}^{(c)} = \sqrt{K_c} \left(\bar{\mathcal{X}}^{(c)} - \bar{\mathcal{X}} \right) \quad (78-3)$$

$$\check{\mathcal{X}} = cat \left(N + 1, \check{\mathcal{X}}^{(1)}, \dots, \check{\mathcal{X}}^{(C)} \right) \quad (79-3)$$

در رابطه‌ی ۷۴-۳، S_b ماتریس پراکندگی بین کلاسی متناظر با مُد n است که با توجه به روابط بالا

به صورت زیر نوشته می‌شود:

$$S_b^{-n} = \sum_{c=1}^C \left\langle \check{\mathcal{Z}}^{(c)}, \check{\mathcal{Z}}^{(c)} \right\rangle_{-n} = \langle \check{\mathcal{Z}}^{-n}, \check{\mathcal{Z}}^{-n} \rangle_{-n} \quad (80-3)$$

به روش مشابه می‌توان مخرج رابطه‌ی ۷۳-۳ را (که در واقع معیار پراکندگی درون کلاسی است) به وسیله‌ی ماتریس پراکندگی درون کلاسی توصیف کرد:

$$\begin{aligned} s_w &= \sum_{k=1}^K \|\mathcal{G}^{(k)} - \bar{\mathcal{G}}^{(c_k)}\|_F^2 = \sum_{k=1}^K \|(\mathcal{X}^{(k)} \times_{-n} \{\mathbf{U}^T\}) \times_n \mathbf{U}^{(n)} - (\bar{\mathcal{X}}^{(c_k)} \times_{-n} \{\mathbf{U}^T\}) \times_n \mathbf{U}^{(n)}\|_F^2 \\ &= \sum_{k=1}^K \|(\mathcal{X}^{(k)} - \bar{\mathcal{X}}^{(c_k)}) \times_{-n} \{\mathbf{U}^T\} \times_n \mathbf{U}^{(n)T}\|_F^2 = \sum_{k=1}^K \|\tilde{\mathcal{Z}}^{(k)} \times_n \mathbf{U}^{(n)T}\|_F^2 \\ &= \sum_{k=1}^K tr\{\mathbf{U}^{(n)T} \left\langle \tilde{\mathcal{Z}}^{(k)}, \tilde{\mathcal{Z}}^{(k)} \right\rangle_{-n} \mathbf{U}^{(n)}\} = tr\{\mathbf{U}^{(n)T} \left(\sum_{k=1}^K \left\langle \tilde{\mathcal{Z}}^{(k)}, \tilde{\mathcal{Z}}^{(k)} \right\rangle_{-n} \right) \mathbf{U}^{(n)}\} \\ &= tr\{\mathbf{U}^{(n)T} S_w^{-n} \mathbf{U}^{(n)}\} \end{aligned} \quad (81-3)$$

که در آن

$$S_w^{-n} = \sum_{k=1}^K \left\langle \tilde{\mathcal{Z}}^{(k)}, \tilde{\mathcal{Z}}^{(k)} \right\rangle_{-n} \quad (82-3)$$

و

$$\tilde{\mathcal{Z}}^{(k)} = \tilde{\mathcal{X}}^{(k)} \times_{-n} \{\mathbf{U}^T\} \quad (83-3)$$

$$\tilde{\mathcal{X}}^{(k)} = \mathcal{X}^{(k)} - \bar{\mathcal{X}}^{(c_k)} \quad (84-3)$$

الگوریتم ۴ شبکه کد الگوریتم کاهش ویژگی HODA [۲۰].

Input: $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N \times K}$ concatenated tensor of K training samples.

Output: $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times J_n}, n = 1, \dots, N$ and $\mathcal{G} \in \mathbb{R}^{J_1 \times \dots \times J_N \times K}$ concatenated training feature tensors

```

1: Initialize  $\mathbf{U}^{(n)}, n = 1, \dots, N$ 
2: Calculate  $\check{\mathcal{X}}$  and  $\check{\mathcal{Z}}$  according to equations (3-80) and (3-85)
3: while not converged do
4:   for  $n = 1, \dots, N$  do
5:      $\tilde{\mathcal{Z}} = \check{\mathcal{X}} \times_{-(n, N+1)} \{\mathbf{U}^T\}$ 
6:      $S_w^{-n} = \langle \tilde{\mathcal{Z}}, \tilde{\mathcal{Z}} \rangle_{-n}$ 
7:      $\check{\mathcal{Z}} = \check{\mathcal{X}} \times_{-(n, N+1)} \{\mathbf{U}^T\}$ 
8:      $S_b^{-n} = \langle \check{\mathcal{Z}}, \check{\mathcal{Z}} \rangle_{-n}$ 
9:      $\phi = \frac{\text{trace}(\mathbf{U}^{(n)T} S_b^{-n} \mathbf{U}^{(n)})}{\text{trace}(\mathbf{U}^{(n)T} S_w^{-n} \mathbf{U}^{(n)})}$ 
10:     $\begin{bmatrix} \mathbf{U}^{(n)} \\ \Lambda \end{bmatrix} = \text{eigs}(S_b^{-n} - \phi S_w^{-n}, J_n, 'LM')$             $\triangleright \text{eigs}(S_b^{-n}, S_w^{-n}, J_n, 'LM')$ 
11:     $\begin{bmatrix} \mathbf{U}^{(n)} \\ \Lambda \end{bmatrix} = \text{eigs}(\mathbf{U}^{(n)} \mathbf{U}^{(n)T} \langle \mathcal{X}, \mathcal{X} \rangle_{-n} \mathbf{U}^{(n)} \mathbf{U}^{(n)T})$ 
12:   end for
13: end while
14:  $\mathcal{G} = \mathcal{X} \times_{-(N+1)} \{\mathbf{U}^T\}$ 

```

بنابراین برای هر $\mathbf{U}^{(n)}, n = 1, \dots, N$ از مسئله‌ی بهینه‌سازی زیر به دست می‌آید:

$$\mathbf{U}^{(n)} = \underset{\mathbf{U}^{(n)}}{\operatorname{argmax}} \frac{\text{tr}\{\mathbf{U}^{(n)T} S_b^{-n} \mathbf{U}^{(n)}\}}{\text{tr}\{\mathbf{U}^{(n)T} S_w^{-n} \mathbf{U}^{(n)}\}} \quad (85-3)$$

که حل آن به وسیله‌ی GEVD ممکن است: ستون‌های $\mathbf{U}^{(n)}$ را به وسیله‌ی J_n بردار ویژه‌ی اول مسئله‌ی مقداردهی می‌کنیم. شبکه کد این الگوریتم در الگوریتم ۴ آورده شده است.

در [۱۶] حالت خاصی از HODA (که در آن $N = 2$) با نام Spatial-Temporal Discriminant Analysis (STDA) معرفی شده است و شرط هم‌گرایی در آن به صورت زیر تعریف شده است:

$$\text{Error} = \|\mathbf{U}^{(n)}(i) - \mathbf{U}^{(n)}(i-1)\| < \varepsilon, \quad n = 1, 2 \quad (86-3)$$

که در آن (i) $\mathbf{U}^{(n)}$ ماتریس پایه‌ی n -م‌د تکرار ام الگوریتم است.

TLLD ۴-۵-۳

روش TLLD^{۴۶} در سال ۲۰۰۹ در [۲۵] معرفی شد. این روش به نوعی تعمیم تانسوری روش

LLD^{۴۷} است.

^{۴۶}Tensor Linear Laplacian Discrimination

^{۴۷}Linear Laplacian Discrimination

در LDA همچون ماتریس‌های پراکندگی بین‌کلاسی و درون‌کلاسی محاسبه می‌شوند، با این تفاوت که محاسبه‌ی آن‌ها وزن دار است. همین تفاوت میان فرمول‌بندی TLLD و HODA وجود دارد.

در TLLD معیار پراکندگی درون‌کلاسی به صورت زیر محاسبه می‌شود:

$$s_w = \sum_{c=1}^C \sum_{\mathcal{X}_k \in \Omega_c} w_k \|\mathcal{G}_k - \bar{\mathcal{G}}^c\|_F^2 \quad (87-3)$$

که در آن w_k وزن متناظر با داده‌ی k است. به طور مشابه معیار پراکندگی بین‌کلاسی از رابطه‌ی زیر محاسبه می‌شود:

$$s_b = \sum_{c=1}^C w^{(c)} K_c \|\bar{\mathcal{G}}^c - \bar{\mathcal{G}}\|_F^2 \quad (88-3)$$

که در آن w^c وزن متناظر با کلاس c است.

هدف الگوریتم بیشینه کردن s_b و کمینه‌سازی s_w است. بنابراین TLLD علاقه‌مند به حل مسئله‌ی بهینه‌سازی به صورت زیر است:

$$\underset{\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}}{\operatorname{argmax}} \frac{s_b}{s_w} \quad (89-3)$$

مسئله‌ی بالا را به روش جایگزینی حل می‌کنیم. فرض کنید می‌خواهیم ماتریس پایه‌ی n ام را بیابیم.

با بازنویسی رابطه‌ی ۸۷-۳ می‌توان دید:

$$\begin{aligned} s_w &= \operatorname{tr}\{\mathbf{U}^{(n)T} \left[\sum_{k=1}^K w_k \left(\mathbf{Z}_k - \bar{\mathbf{Z}}^{(c_k)} \right)_{(n)} \left(\mathbf{Z}_k - \bar{\mathbf{Z}}^{(c_k)} \right)_{(n)}^T \right] \mathbf{U}^{(n)}\} \\ &= \operatorname{tr}\{\mathbf{U}^{(n)T} \mathbf{S}_w^{(n)} \mathbf{U}^{(n)}\} \end{aligned} \quad (90-3)$$

که در آن $\{\mathbf{U}\}$ ماتریس پراکندگی $S_w^{(n)} = \sum_{k=1}^K w_k \left(\mathbf{Z}_k - \bar{\mathbf{Z}}^{(c_k)} \right)_{(n)} \left(\mathbf{Z}_k - \bar{\mathbf{Z}}^{(c_k)} \right)_{(n)}^T$ و $\mathcal{Z}_k = \mathcal{X}_k \times_{-n} \{\mathbf{U}\}$ درون‌کلاسی مُد n است.

به طور مشابه می‌توان معیار s_b را بازنویسی کرد:

$$\begin{aligned} s_b &= \operatorname{tr}\{\mathbf{U}^{(n)T} \left[\sum_{c=1}^C w^{(c)} K_c \left(\bar{\mathbf{Z}}^{(c)} - \bar{\bar{\mathbf{Z}}} \right)_{(n)} \left(\bar{\mathbf{Z}}^c - \bar{\bar{\mathbf{Z}}} \right)_{(n)}^T \right] \mathbf{U}^{(n)}\} \\ &= \operatorname{tr}\{\mathbf{U}^{(n)T} \mathbf{S}_b^{(n)} \mathbf{U}^{(n)}\} \end{aligned} \quad (91-3)$$

که در آن $\mathbf{S}_b^{(n)} = \sum_{c=1}^C w^{(c)} K_c \left(\bar{\mathbf{Z}}^{(c)} - \bar{\bar{\mathbf{Z}}} \right)_{(n)} \left(\bar{\mathbf{Z}}^c - \bar{\bar{\mathbf{Z}}} \right)_{(n)}^T$ میانگین همه‌ی $Z_{k(n)}$ و $\bar{\bar{\mathbf{Z}}}_{(n)}$ میانگین همه‌ی $Z_{k(n)}$ است.

بنابراین مسئله‌ی ۸۹-۳ برای به دست آوردن ماتریس پایه‌ی n ام به مسئله‌ی زیر تبدیل می‌شود:

$$\begin{aligned} \mathbf{U}^{(n)} &= \underset{\mathbf{U}^{(n)}}{\operatorname{argmax}} \frac{\operatorname{tr}\{\mathbf{U}^{(n)T} \mathbf{S}_b^{(n)} \mathbf{U}^{(n)}\}}{\operatorname{tr}\{\mathbf{U}^{(n)T} \mathbf{S}_w^{(n)} \mathbf{U}^{(n)}\}} \\ \text{s.t. } \quad \mathbf{U}^{(n)T} \mathbf{U}^{(n)} &= \mathbf{I} \end{aligned} \quad (92-3)$$

مسئله‌ی ۹۲-۳ یک مسئله‌ی trace ratio است و به وسیله‌ی GEVD حل می‌شود. شبکه کد مربوط به این روش در الگوریتم ۵ آورده شده است. برای تعیین وزن‌ها، رابطه‌ی زیر استفاده شده است.

$$\begin{aligned} w_i &= \exp\left(-\frac{d^*(\mathcal{X}_i, \Omega_{c_i})}{t}\right), \quad i = 1, \dots, K \\ w^c &= \exp\left(-\frac{d^*(\Omega_c, \Omega)}{t}\right), \quad c = 1, \dots, C \end{aligned} \quad (93-3)$$

که در آن $(..,.)^d$ نماد فاصله است. اکثر روش‌های موجود برای استخراج ویژگی، فاصله‌ی اقلیدسی را به عنوان متریک فضا در نظر می‌گیرند (همچنین روش LLD که TLLD به نوعی تعمیم آن است). این در حالی است که در بسیاری از موارد اطلاعاتی از متریک فضای داده‌ها نداریم. همچنین اصولاً روش‌ها فاصله‌ی دو مجموعه را به فاصله‌ی مراکز^{۴۸} (میانگین) آن‌ها تعریف می‌کنند. مشکل این موضوع این است که ممکن است که مراکز مجموعه‌ها روی منیفلد داده‌ها قرار نگیرد و به همین دلیل متریک فضای نمونه را نمی‌توان به آن‌ها اعمال کرد. بنابراین تعریف بهتر از فاصله تعریفی است که بر اساس ساختار داده‌ها باشد، نه یک متریک از قبل تعیین شده. روش TLLD به همین دلایل فاصله‌ی contextual را به خدمت می‌گیرد که بر اساس میزان مشارکت در ساختار داده‌ها در فضای نمونه محاسبه می‌شود. توضیح فاصله‌ی contextual در حوزه‌ی تمرکز این نوشتار نیست.

۶-۳ جعبه‌ابزارهای تانسوری

جعبه‌ابزارهای متعددی برای سهولت در انجام عملیات تانسوری وجود دارد. برخی از آن‌ها از ارائه‌ی عملیات پایه‌ای فراتر رفته و پیاده‌سازی کاربردهایی را هم در اختیار کاربرانشان می‌گذارند. در ادامه به معرفی اجمالی معروف‌ترین آن‌ها می‌پردازیم:

۱. tensor toolbox- Bader and Kolda [۲۷، ۲۸]: این جعبه‌ابزار متن-باز^{۴۹} از پرکاربردترین

جعبه‌ابزارها است و برخی جعبه‌ابزارهای دیگر هم از آن استفاده کرده‌اند. کلیه‌ی عملیات پایه‌ای

^{۴۸}Centroid

^{۴۹}Open source

الگوریتم ۵ روش کاهش ویژگی TLLD [۲۵]

Input: K training data $\{\mathcal{X}_k \in \mathbb{R}^{I_1 \times \dots \times I_N}\}_{k=1}^K$ along with their labels $\{c_k\}_{k=1}^K$ and $J_n, n = 1, \dots, N$

Output: $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times J_n}, n = 1, \dots, N$: N orthogonal basis factors and $\mathcal{G}_k \in \mathbb{R}^{J_1 \times \dots \times J_N}, k = 1, \dots, K$: Training feature tensors

- 1: Compute the weights $\{w_k\}_{k=1}^K$ and $\{w^c\}_{c=1}^C$
- 2: Initialize $\mathbf{U}^{(n)}, n = 2, \dots, N$
- 3: **repeat**
- 4: **for** $n = 1, \dots, N$ **do**
- 5: **for** $k = 1, \dots, K$ **do**
- 6: $\mathcal{Z}_k^{(n)} = \mathcal{X}_k \times_{-n} \{\mathbf{U}^T\}$
- 7: **end for**
- 8: $S_b^{(n)} = \sum_{c=1}^C w^{(c)} K_c (\bar{\mathbf{Z}}^{(c)} - \bar{\bar{\mathbf{Z}}})_{(n)} (\bar{\mathbf{Z}}^{(c)} - \bar{\bar{\mathbf{Z}}})_{(n)}^T$
- 9: $S_w^{(n)} = \sum_{k=1}^K w_k (\mathbf{Z}_k - \bar{\mathbf{Z}}^{(c_k)})_{(n)} (\mathbf{Z}_k - \bar{\mathbf{Z}}^{(c_k)})_{(n)}^T$
- 10: $[\mathbf{U}^{(n)}, \Lambda] = \text{eigs}(S_b^{(n)}, S_w^{(n)}, J_n, 'LM')$
- 11: **end for**
- 12:
- 13: **until** a criterion is met

タンسوری از قبیل الگوریتم‌های ALS تخمین تجزیه‌های تانسوری، انواع ضرب ماتریسی و ضرب مُد-n، ماتریسی‌سازی و غیره در این جعبه‌ابزار به سهولت قابل دسترس است. در این پروژه از این جعبه‌ابزار استفاده شده است.

.۲. **NFEA-Phan**:^{۵۰} این جعبه‌ابزار، از tensor toolbox استفاده می‌کند و پیاده‌سازی سه الگوریتم کاهش ویژگی تانسوری (TLLD, HODA و LTDA) به صورت کاربردی است. کدهای اصلی این جعبه‌ابزار با پسوند p. هستند و دسترسی به محتویات آن‌ها ممکن نیست.

.۳. **TDALAB-Cichocki et al.**:^{۵۱} این جعبه‌ابزار هم از tensor toolbox استفاده می‌کند. ویژگی اصلی این جعبه‌ابزار این است که پیاده‌سازی روش‌های MBSS^{۵۲} و یک روش کاهش ویژگی بر مشابه با HODA را در دسترس قرار می‌دهد.

.۴. **Nway toolbox-Bro et al.**:^{۵۳} این جعبه‌ابزار از اولین و همچنین پرکاربردترین‌ها است. علاوه بر

عملیاتی که tensor toolbox دارد، پیاده‌سازی روش‌های دیگری همچون NPLS^{۵۴} را هم دارد.

^{۵۰} <http://www.bsp.brain.riken.jp/phан/nfea/nfea.html>

^{۵۱} <http://www.bsp.brain.riken.jp/TDALAB/>

^{۵۲} Multilinear Blind Source Separation

^{۵۳} <http://www.models.life.ku.dk/nwaytoolbox>

^{۵۴} Nway Partial Least Squares

۷-۳ جمع‌بندی

در این فصل پیش‌زمینه‌ی نظری لازم برای فصول پیش‌رو را مرور کردیم. ابتدا دو روش برداری کاهش ویژگی را توضیح دادیم (LDA و SRDA). پس از آن به بیان مفاهیم اولیه و برخی جزئیات جبر تانسوری پرداختیم، فرمول‌بندی چند روش کاهش ویژگی تانسوری را توضیح دادیم و در پایان جعبه‌ابزارهای تانسوری را معرفی کردیم.

فصل ۴

پیشینه‌ی پژوهش

۱-۴ پیش‌گفتار

این فصل به دو بخش اصلی تقسیم می‌شود: ابتدا روش‌های برداری که برای طبقه‌بندی در P300 استفاده شده‌اند را مروز خواهیم کرد و پس از آن کاربرد روش‌های مبتنی بر تجزیه‌ی تانسوری در speller مطالعه‌ی ERP‌ها را بررسی می‌کنیم. لازم به توضیح است که منظور از روش‌های برداری روش‌هایی هستند که داده‌ها را به عنوان بردار دریافت می‌کنند، در حالی که روش‌های تانسوری، داده‌ها را به عنوان یک تانسور در فضا در نظر می‌گیرند. هرچند ممکن است در برخی روش‌ها پس از اعمال روش‌های کاهش ویژگی تانسوری، از طبقه‌بندهای برداری برای طبقه‌بندی نهایی استفاده شود، اما در این فصل برای تأکید بر تفاوت موجود در نگرش به داده‌ها (برداری و تانسوری) این تقسیم‌بندی صورت گرفته است.

۲-۴ روش‌های برداری در تشخیص پتانسیل وابسته به رخداد در پارادایم P300 Speller

در این بخش به معرفی اجمالی روش‌های برداری که در تشخیص پتانسیل وابسته به رخداد در P300 مورد استفاده قرار گرفته‌اند می‌پردازیم (مراد از «روش‌های برداری» روش‌هایی است که هر داده را به عنوان یک بردار در فضا در نظر می‌گیرند). معرفی این روش‌ها در این فصل به این دلیل انجام شده است که در فصل نتایج، روش پیشنهادی را با آن‌ها مقایسه کنیم. از همین جهت از آن‌جا که بررسی این روش‌ها هدف اصلی این نوشتار نیست، تنها به معرفی اجمالی آن‌هایی می‌پردازیم که بر روی دادگان مورد

استفاده‌ی ما (مجموعه‌داده‌ی دوم از مسابقه‌ی سوم BCI [۲۹] که داده‌های ثبت شده از دو سوزه است)^۱ انجام شده‌اند؛ هرچند که به صورت کلی این روش‌ها از مهم‌ترین پژوهش‌های انجام شده برای تشخیص P300 در ERP در splleller هستند. لازم به ذکر است که در اینجا به معرفی فرآیند هر روش می‌پردازیم و برای جلوگیری از تکرار مطالب، تمامی نتایج گزارش شده را در جدول ۲-۶ برای مقایسه با روش پیشنهادی آورده‌ایم.

در همه‌ی این روش‌ها پس انجام پیش‌پردازش (و انتخاب کanal) و جداسازی تک‌ثبت‌ها، برای هر تحریک (منتظر با یک تک‌ثبت) نمونه‌های زمانی کanal‌ها پشت سر هم قرار داده می‌شوند و بردار ویژگی را برای آن تک‌ثبت به وجود می‌آورند. برای مثال اگر هر کanal n نمونه‌ی زمانی داشته باشد و c کanal داشته باشیم، هر تک‌ثبت $c \times n$ نمونه‌ی زمانی (یا ویژگی) خواهد داشت.

SVM ۱-۲-۴

روش SVM یکی از بهترین روش‌های شناخته شده برای تشخیص P300 است. گروه دوم مسابقه [۲۹] از SVM استفاده کرده است. پیش‌پردازش انجام شده علاوه بر استفاده از ICA برای حذف آرتیفیکت‌های پلک زدن، فیلتر میان‌گذر و کاهش نرخ نمونه‌برداری است. نمونه‌های ۱۰۰ تا ۸۵۰ میلی‌ثانیه بعد از تحریک به عنوان تک‌ثبت‌ها جدا شده و پس از آن کاهش نرخ نمونه‌برداری با ضریب ۵ انجام شده است. برای هر سوزه ۱۰ کanal انتخاب و پس از ساخت بردارهای ویژگی SVM خطی آموزش داده شده است.

eSVM ۲-۲-۴

گروه بونده‌ی مسابقه [۱۳] از eSVM^۲ برای تشخیص P300 استفاده کرده‌اند. روش آن‌ها کماکان روش پیشرو در تشخیص کاراکترهای این مجموعه‌دادگان است. این گروه داده‌های هر ۵ کاراکتر را در یک دسته قرار داده‌اند. فرض کنیم برای هر سوزه M دسته داریم. در این صورت بر روی هر یک از این M دسته یک طبقه‌بند SVM دو کلاسه (کلاس P300 و (non-P300) آموزش داده می‌شود.تابع تصمیم‌گیری برای طبقه‌بند m به صورت زیر است:

$$f_m(\mathbf{x}) = \sum_{i \in P_m} y_i \alpha_i^{(m)} \langle \mathbf{x}, \mathbf{x}_i \rangle + b^{(m)} \quad (1-4)$$

^۱جزئیات این مجموعه دادگان در فصل ۶ بیان خواهد شد.

^۲Ensemble SVM

که در آن $\{x_{r|c}^j\}_{j=1}^J$ و $b^{(m)}$ پارامترهای SVM هستند. x داده‌ی جدید است که باید روی آن تصمیم‌گیری انجام شود و P_m مجموعه‌ی داده‌های دسته‌ی m از سوژه‌ی P است. y_i برچسب داده‌ی آن از مجموعه‌ی آموزش است.

بنابراین اگر $x_{r|c}$ داده‌ی جدیدی مربوط به یک سطر یا ستون خاص باشد هر یک از M طبقه‌بند یک امتیاز^۳ ($f_m(x_{r|c})$) به این داده نسبت می‌دهند. پس از تعیین امتیاز تمامی تکثیت‌های مربوط به یک کاراکتر، باید از تجمعی این امتیازها (که در واقع مربوط به طبقه‌بندی دو کلاسه است) تعیین کنیم کدام یک از سطرهای و ستون‌ها حاوی کاراکتر مورد نظر است. برای این کار یک سطر (یا ستون) خاص را در نظر بگیرید. میانگین امتیازهای همه‌ی تکرارهای مربوط به آن را در M طبقه‌بند میانگین‌گیری می‌کنیم. به عبارتی برای آن سطر یا ستون یک اندیس به صورت زیر دست می‌آوریم:

$$S_{r|c} = \frac{1}{M} \sum_{m=1}^M \sum_{j=1}^J f_m(x_{r|c}^j) \quad (2-4)$$

که در آن J تعداد تکرارهای روشن شدن هر سطر یا ستون است و $x_{r|c}^j$ داده‌ی مربوط به تکرار j از آن سطر یا ستون است. با جایگذاری رابطه‌ی ۱-۴ در ۲-۴ خواهیم داشت:

$$S_{r|c} = \frac{1}{M} \sum_{m=1}^M \sum_{i \in P_m} y_i \alpha_i^{(m)} \left\langle \frac{1}{J} \sum_{j=1}^J x_{r|c}^j, x_i \right\rangle + b^{(m)} \quad (3-4)$$

که در واقع معادل این است که میان تکرارهای یک سطر یا ستون میانگین‌گیری انجام شود و به این میانگین به وسیله‌ی M طبقه‌بند امتیازی نسبت داده شود و سپس میان این M امتیاز میانگین‌گیری شود. می‌دانیم که میانگین‌گیری تکرارها در P300 speller به صورت قابل توجهی به افزایش SNR کمک می‌کند. پس از محاسبه‌ی اندیس $S_{r|c}$ برای ۶ سطر (یا ستون)، آن سطر (یا ستونی) که بیشترین اندیس را دارد به عنوان سطر (یا ستون) حاوی کاراکتر انتخاب می‌شود. از برخورد این سطر و ستون کاراکتر مورد نظر به دست می‌آید.

این گروه [۱۳] یک فرآیند تکرارشونده^۴ برای انتخاب کanal برای هر سوژه انجام داده‌اند که توضیح آن از حوصله‌ی این نوشتار دور است.

^۳Score^۴Recursive

SWLDA ۳-۲-۴

طبقه‌بندی SWLDA^۵ در واقع همان رگرسیون گام به گام^۶ باینتری است که در اینجا به توضیح آن می‌پردازیم [۳۰].

فرض کنید که \mathbf{X} ماتریس $N \times M$ مشاهدات است که از N مشاهده‌ی M بعدی تشکیل شده است و \mathbf{y} بردار $1 \times N$ برچسب‌های مسئله‌ی رگرسیون خطی زیر را انجام دهیم:

$$\mathbf{y} = \mathbf{w}^T \cdot \mathbf{X} + \mathbf{b} = f(\mathbf{X}) \quad (4-4)$$

که در آن $\mathbf{w} = [w_1, w_2, \dots, w_M]^T$ بردار وزن‌های رگرسیون است. طبیعتاً انتظار داریم که هر چه تعداد ویژگی‌ها بیشتر باشد، مدل رگرسیون بهتر باشد (یا دست کم بدتر نباشد). با این حال لزوماً این «بهتر شدن» مدل رگرسیون از لحاظ آماری معنادار^۷ نیست. در این موارد برای مقایسه‌ی دو مدل رگرسیون که یکی با M_1 ویژگی و دیگری با M_2 ویژگی به دست آمده‌اند از آزمون F-test استفاده می‌کنند. اما چگونه از میان M ویژگی، زیرمجموعه‌ی ویژگی‌هایی را انتخاب کنیم که از لحاظ آماری به صورت معناداری مدل رگرسیون را بهبود بخشدند؟ برای این کار از رگرسیون گام به گام استفاده می‌شود که در زیر فرآیند آن آورده شده است. فرض کنید ω مجموعه‌ی اندیس ویژگی‌های انتخاب شده است و $\omega = \{1, \dots, M\}$.

۱. قرار می‌دهیم $\omega = k$ و ویژگی با بیشترین همبستگی با \mathbf{y} به عنوان اولین وارد ω می‌شود و یک مدل رگرسیون به صورت $\hat{\mathbf{y}}_k = f(X(:, \omega))$ برآش می‌کنیم.

۲. یک مدل رگرسیون با ویژگی متناظر با هر $\psi \in i$ به همراه ویژگی‌های متناظر با اندیس‌های داخل ω این‌که دو مدل تفاوت معنادار ندارند) میان دو مدل رگرسیون $\hat{\mathbf{z}}$ و $\hat{\mathbf{y}}_k$ به دست می‌آوریم. اگر های وجود داشت که p-value آن از آستانه (α_1) کم‌تر باشد، آن را به ω اضافه می‌کنیم. در واقع یک ویژگی در صورتی اضافه می‌شود که فرض صفر آزمون آماری رد شود، یعنی دو مدل تفاوت معنادار داشته باشند. در صورتی که ویژگی‌ای اضافه شد قرار می‌دهیم $\omega = k + 1$ و $\hat{\mathbf{y}}_k = \hat{\mathbf{z}}_i$. در غیر این صورت به مرحله‌ی ۲ می‌رویم.

^۵Stepwise LDA

^۶Stepwise regression

^۷Significant

۳. به ازای هر $\omega \in \hat{\omega}$ یک مدل رگرسیون به صورت $(X(:, \omega - \{i\})) = f(\hat{z}_i)$ برآش می‌کنیم و برای هر ω

یک p-value حاصل از F-test (با فرض صفر این‌که دو مدل تفاوت معنادار ندارند) میان دو مدل

رگرسیون \hat{z}_i و \hat{y}_k به دست می‌آوریم. اگر i وجود داشت که $p\text{-value}$ آن از آستانه (α_2)

بیشتر باشد، آن را از ω حذف می‌کنیم. در واقع یک ویژگی در صورتی حذف می‌شود که فرض صفر

آزمون آماری قبول شود، یعنی دو مدل تفاوت معناداری نداشته باشند. در صورتی که ویژگی‌ای

حذف شد قرار می‌دهیم 1 باز $k = k + 1$ و $\hat{z}_i = \hat{y}_k$ و به مرحله‌ی 2 باز می‌گردیم. در غیر این صورت،

بدون هیچ تغییری در k به مرحله‌ی 2 می‌رویم.

فرآیند بالا در صورتی متوقف می‌شود که دیگر ω تغییر نکند. البته در برخی موارد یک حد بالایی برای

تعداد ویژگی‌ها تعیین می‌کنند. به این فرآیند رگرسیون گام به گام پیشرو-پسرو^۸ گفته می‌شود.

نسخه‌هایی که فقط پیشرو یا پسرو باشند هم موجود است.

پس از توقف الگوریتم مدل رگرسیون نهایی به عنوان طبقه‌بند باینری استفاده می‌شود. در [۳۱]

این روش برای تشخیص حضور P300 Speller در داده‌های P300 استفاده شده است. طبقه‌بند فوق به هر

تکثیت یک امتیاز می‌دهد. از تجمیع این امتیازها می‌توان تشخیص کاراکتر را در P300 speller انجام داد

(شیوه‌ی تجمیع امتیازها در فصل ۶ به تفصیل بیان شده است).

EFLD ۴-۲-۴

در [۳۲] سالواریس و همکارش از طبقه‌بند FLD در حالت گروهی استفاده کرده‌اند و آن را EFLD^۹

نامیده‌اند. پیش‌پردازش انجام شده شامل فیلتر میان‌گذر و انتخاب کانال به صورت ترتیبی پیشرو^{۱۰} است.

ویژگی‌های استخراج شده تبدیل DWT^{۱۱} است. برای استفاده از طبقه‌بند گروهی، داده‌ها به ۱۰ دسته

تقسیم شده‌اند و یک طبقه‌بند FLD بر هر یک از دسته‌ها آموزش داده شده است. برای تشخیص کلاس

P300 در داده‌های آزمون مانند الگوریتم eSVM در ۴-۲-۴ عمل می‌شود: برای هر سطر یا ستون امتیاز

۱۰ طبقه‌بند روی تکرارهای مختلف آن سطر میانگین‌گیری می‌شود و سطر یا ستون با بیشترین امتیاز به

عنوان سطر یا ستون حاوی کاراکتر مورد نظر انتخاب می‌شود.

^۹Forward-backward

^{۱۰}Ensemble Fisher Linear Discriminant Analysis

^{۱۱}Sequential forward

^{۱۱}Discrete Wavelet Transform

CNN ۵-۲-۴

در [۳۳] پس از کاهش نرخ نمونه‌برداری و فیلتر میان‌گذر، داده‌ها نرمالیزه شده‌اند و یک طبقه‌بند به وسیله‌ی شبکه‌ی عصبی کانولوشنی (CNN^{۱۲}) بر اساس آن‌ها آموزش داده شده است (توضیح جزئیات آموزش شبکه عصبی از حوصله‌ی این نوشتار دور است). بر اساس این شبکه‌ی عصبی، چندین نوع طبقه‌بند معرفی کردند که دو مورد آن‌ها که بهترین عملکرد را داشته‌اند معرفی می‌کنیم: در ۱ CNN با استفاده از کل داده‌های آموزش، یک CNN آموزش می‌بیند. در ۱ MCNN-۱^{۱۳} پنج CNN به وسیله‌ی پنج زیرمجموعه از داده‌ها آموزش می‌بینند. هر زیر مجموعه کل تکثیت‌های کلاس P300 و یک پنجم کلاس غیر P300 را دارد. بدین وسیله هر شبکه با تعداد مساوی داده از دو کلاس آموزش می‌بیند.

۳-۴ کاربرد تجزیه‌ی تانسوری در پردازش داده‌های حاوی پتانسیل وابسته به رخداد

۱-۳-۴ کاربرد تجزیه‌ی CP در پردازش داده‌های حاوی پتانسیل وابسته به رخداد

تاریخچه‌ی استفاده از تجزیه‌ی CP در مطالعه‌ی EEG حاوی ERP به اوخر دهه‌ی ۱۹۸۰ باز می‌گردد [۳۴، ۳۵، ۳۶]. در این مقالات CP با نام Topographic Component Model به عنوان روش مناسبی برای آنالیز پتانسیل وابسته به رخداد چند-سوژه معرفی و برای استخراج الگوی ERP مشترک از داده‌های ساختگی چند-سوژه معرفی شده‌است. گروهی در سال ۲۰۰۴ اطلاعات زمان، فرکانس و کanal داده‌های EEG را به وسیله‌ی CP با هدف تشخیص فعالیت مغز در یک باند فرکانسی خاص تجزیه کردند [۳۷]. داده‌های EEG آن‌ها حاوی پتانسیل وابسته به رخداد نبودند اما از این جهت که این اولین پژوهش منتشر شده است که ساخت تانسور زمان-فرکانس-کanal از EEG را معرفی کرده، ایده‌ی پژوهش‌های بعدی بوده است.

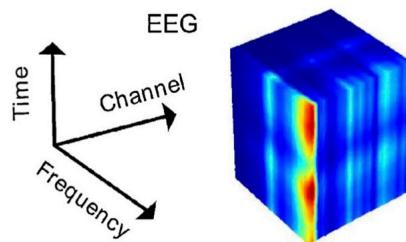
^{۱۲}Convolutional neural network

^{۱۳}Multicalssifier CNN

۱-۱-۳-۴ جداسازی و مکان‌یابی منبع پتانسیل وابسته به رخداد

الف - تجزیه‌ی CP

و همکارانش در [۸] تansورهای شامل اطلاعات حوزه‌ی زمان-فرکانس را برای اولین بار در آنالیز سیگنال EEG حاوی ERP مورد ارزیابی قرار دادند. داده‌های EEG این پژوهش مربوط به یک آزمایش با دو نوع تحریک بصری بوده است [۳۸] که هر دو نوع تحریک باعث برانگیخته شدن فعالیت گاما^{۱۴} می‌شوند اما فعالیت در تحریک نوع دوم قوی‌تر از دیگری است (توضیح آزمایش انجام شده برای ثبت EEG خارج از حوصله‌ی این نوشتار است). در این مقاله توانایی تجزیه‌ی CP در اثبات این ویژگی‌ها در دو شرایط آزمایش بررسی شده است. در ادامه از این دو نوع تحریک به عنوان «شرایط آزمایش» یاد می‌کنیم.



.[۲۳] شکل ۴: نمایشی از یک تansور با ابعاد $Channel \times Frequency \times Time$

ابتدا فرض کنید هر تکرار از آزمایش را به صورت یک تansور با ابعاد $Channel \times Frequency \times Time$ درآوریم (شکل ۴-۱). بدین معنی که از نمونه‌های زمانی هر کanal تبدیل زمان-فرکانس گرفته و ماتریس‌های به دست‌آمده را در یک تansور مرتبه سه بچینیم. فرض کنید برای تکرار شماره ^{۱۵}، تansور به دست آمده را با \mathcal{X}_e نشان دهیم. در این مقاله تجزیه‌ی PARAFAC تansورهای زیر تحلیل شده است:

۱. تansور مرتبه سه ITPC

$ITPC^{15}$ معیاری برای میزان همزمانی فعالیت در یک زمان و فرکانس خاص است که به آن phase

locking factor گفته می‌شود و به وسیله‌ی رابطه‌ی زیر محاسبه می‌شود:

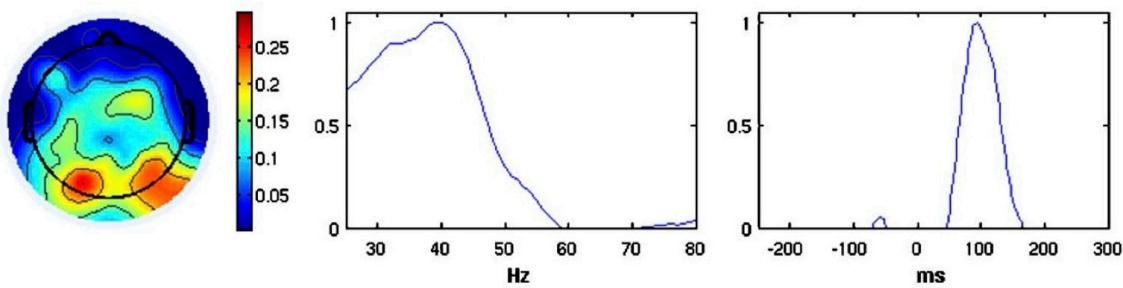
$$ITPC(c, f, t) = \left| \frac{1}{n} \sum_{e=1}^n \frac{\mathcal{X}_e(c, f, t)}{|\mathcal{X}_e(c, f, t)|} \right| \quad (5-4)$$

که در آن n تعداد تکرارهای آزمایش و $\mathcal{X}_e(c, f, t)$ ضریب تبدیل ویولت در کanal شماره‌ی c

فرکانس f و لحظه‌ی t در تکرار e ام هستند. روشن است که مقدار یک برای ITPC نشانه‌ی همزمانی

^{۱۴}Gamma activity

^{۱۵}Inter-Trial Phase Coherence



شکل ۴-۲: فاکتورهای تجزیه‌ی PARAFAC یک مولفه‌ای تانسور سه بعدی ITPC مربوط به یک سوزه. این مولفه نمایان‌گر فعالیت گامای پس‌سری با فرکانس حدود ۴۰ هرتز در حوالی ۱۰۰ میلی‌ثانیه پس از تحریک است [۸].

کامل برانگیختگی نورون‌هاست. مشخصات فعالیت برانگیخته‌ی ثبت شده در ماتریس‌های عامل تجزیه‌ی PARAFAC با یک مولفه برای یکی از سوزه‌ها در شکل ۴-۲ قابل رویت است. همان‌طور که مشاهده می‌کنید در حدود ۱۰۰ میلی‌ثانیه پس از تحریک و در حدود فرکانس ۴۰ هرتز بردارهای تجزیه CP به بیشینه‌ی خود (یعنی یک) رسیده‌اند و این به معنی همزمانی کامل برانگیختگی نورون‌ها و وقوع فعالیت وابسته به رخداد است.

۲. تانسور مرتبه پنج ITPC:

این تانسور همان ITPC است که دو بعد سوزه و شرایط آزمایش را اضافه‌تر دارد. به عبارت دیگر ITPC_{5(.,.,.,s,k)} برابر همان تانسور ITPC برای سوزه‌ی s ام در شرایط آزمایش k^{۱۶} است. پس کافی است تانسور سه‌بعدی ITPC، که در بالا به آن اشاره شد، برای همه‌ی سوزه‌ها در همه‌ی شرایط آزمایش محاسبه شود و در بُعدهای چهار و پنج تانسور جدید قرار داده شود. شکل ۳-۴ مولفه‌های تجزیه‌ی PARAFAC رنگ-۲ این تانسور را نمایش می‌دهد. در هر یک از مولفه‌های تجزیه، بردارهای متناظر با بُعد چهارم و پنجم نمایان‌گر شدت وقوع فعالیت برانگیخته^{۱۷} در سوزه و شرایط آزمایش است. برای مثال در مولفه‌ی دوم مشاهده می‌شود که شدت این فعالیت در تحریک نوع دوم شدیدتر است (بردار متناظر با بُعد پنجم).

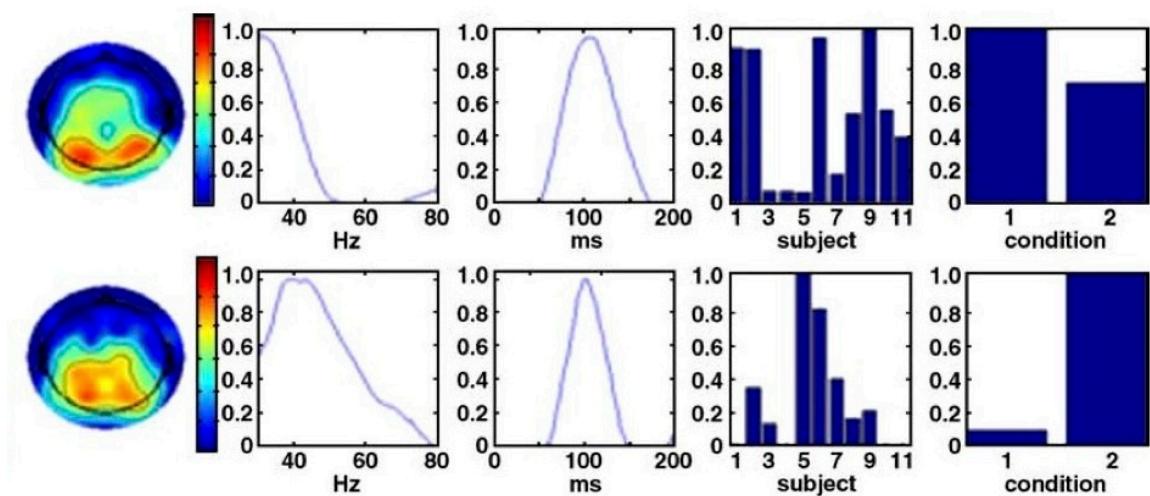
ب - تجزیه‌ی SCP

در شرایطی که خاصیت چند-خطی^{۱۸} داده‌ها از بین برود، تجزیه‌ی CP دچار انحطاط^{۱۹} می‌شود. به عبارت دیگر برخی از مولفه‌های آن با یکدیگر همبستگی زیادی پیدا می‌کنند و این نتایج را بی‌ارزش و یا

^{۱۶}Evoked Activity

^{۱۷}Multilinearity

^{۱۸}Degeneracy



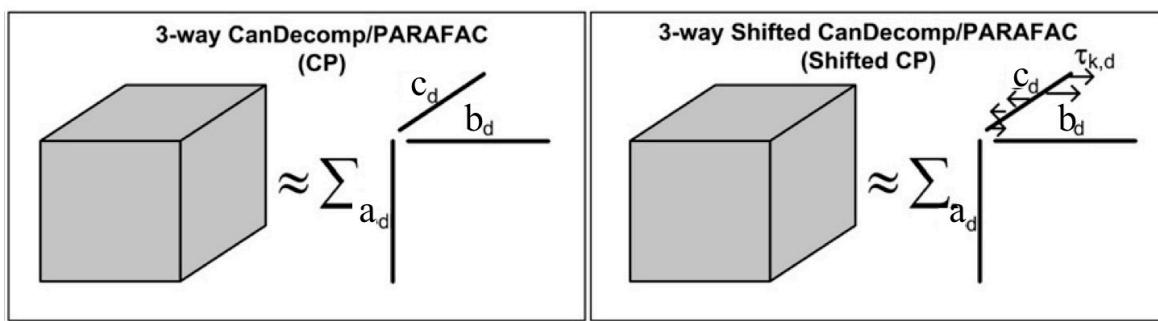
شکل ۴-۴: فاکتورهای تجزیه‌ی ITPC PARAFAC تansور پنچ بعدی که با دو مولفه برازش شده است. همان‌طور که مشاهده می‌شود مولفه‌ی اول در هر دو شرایط آزمایش شدت تقریباً یکسانی دارد. اما مولفه‌ی دوم در تحریک دوم دارای شدت بسیار بیشتری است [۸].

تحلیل آن‌ها را دشوار می‌کند. به همین جهت در این موارد شروطی همچون تعامد و یا استقلال را به فاکتورهای یک بُعد از CP اعمال می‌کنند. در صورتی که دلیل انحطاط CP شیفت‌های زمانی در تکرارهای مختلف آزمایش باشد، به جای اعمال شروط اضافه می‌توان از نسخه‌ی شیفت‌یافته‌ی CP کمک گرفت. این ادعا در سال ۲۰۰۸ برای آنالیز داده‌های EEG حاوی ERP به وسیله‌ی تجزیه‌ی یک تانسور سه بُعدی $Channel \times Time \times Trial$ آزموده شد (علاوه بر این داده‌ها، از داده‌های fMRI وابسته به رخداد هم استفاده شده که توضیح آن‌ها از موضوع این نوشتار خارج است) [۹]. تجزیه‌ی CP شیفت‌یافته (^{۱۹}SCP) به وسیله‌ی رابطه‌ی زیر تعریف می‌شود:

$$x_{ijk} = \sum_{d=1}^D a_{i,d} b_{(i-\tau_{k,d}),d} c_{k,d} + e_{ijk} \quad (4-4)$$

که در آن پروفایل زمانی b_d برای تکرار k ام به اندازه‌ی $\tau_{k,d}$ شیفت داده می‌شود. شکل ۴-۴ به صورت شماتیک تفاوت CP و SCP را نشان می‌دهد. به بیانی می‌توان این رابطه را این‌طور تعبیر کرد: اگر b_d ، c_d و τ_d به ترتیب پروفایل زمانی، شدت آن در تکرارهای مختلف و شیفت‌ها در مولفه‌ی d am تجزیه‌ی SCP باشند، این پروفایل زمانی در تکرار k ام با شیفت به اندازه‌ی $\tau_{k,d}$ و شدت $c_{k,d}$ ظاهر می‌شود. بر این اساس CP مانند SCP سه ماتریس عامل متناظر با سه مُد کانال، زمان و تکرار را تولید می‌کند که ماتریس عامل دوم، الگوهای زمانی منابع جداسازی شده در خود دارد. اما SCP یک ماتریس دیگر هم به دست می‌دهد که متناظر با تاخیر زمانی هر الگوی زمانی در هر تکرار است.

^{۱۹}Shifted CP



شکل ۴-۴: نمایش شماتیک تفاوت CP و SCP. تجزیه‌ی SCP به مُد دوم اجازه می‌دهد تا شیفت داشته باشد به طوری که پروفایل زمانی b_d برای تکرار k ام به اندازه‌ی $\tau_{k,d}$ شیفت داده شود [۹].

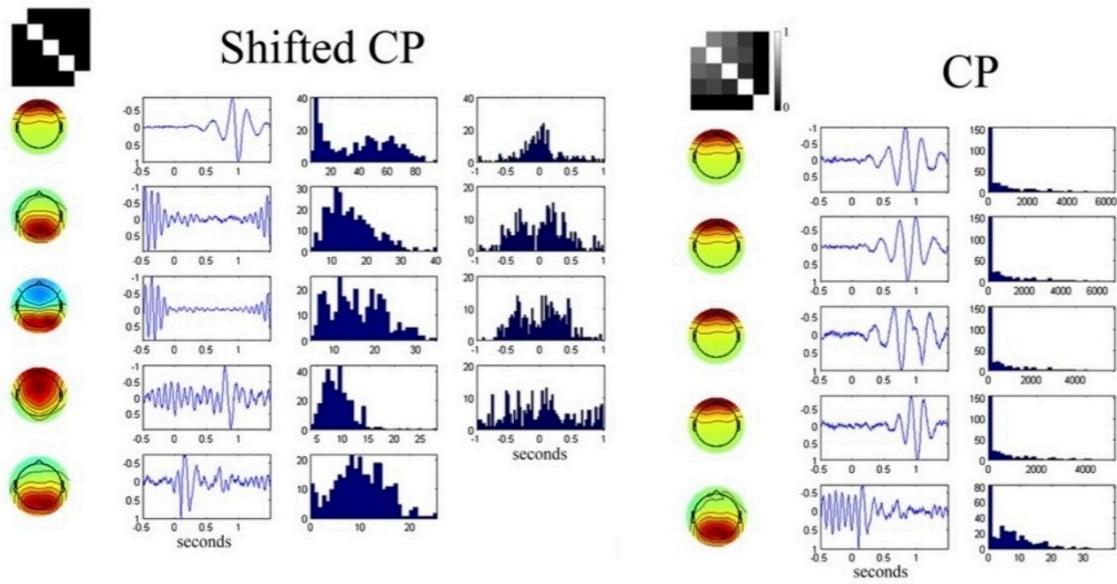
شکل ۴-۵ مولفه‌های تجزیه‌ی CP و SCP تانسور سه بُعدی کanal \times zمان \times تکرار را نمایش می‌دهد.

در این شکل مؤلفه‌ی اول SCP مربوط به پلک زدن، مؤلفه‌ی دوم و سوم مربوط به فعالیت آلفا قبل و بعد از رخداد، مؤلفه‌ی چهارم مربوط به فعالیت آلفای جلوی سر و در نهایت مؤلفه‌ی پنجم مربوط به ERP است. اما در CP چهار مؤلفه‌ی اول تحت تأثیر سیگنال ناشی از پلک زدن در تکرارهای مختلف بوده است. مؤلفه‌ی پنجم به پتانسیل وابسته به رخداد مرتبط است، هرچند تحت تأثیر فعالیت آلفای قبل از رخداد قرار گرفته است و نیز شدت آن در اکثر تکرارها کم است. همان‌طور که انتظار می‌رفت CP دچار انحطاط شده و مولفه‌هایی دارای همبستگی زیادی هستند؛ در حالی که SCP فعالیت‌های مختلف مغزی را به خوبی جداسازی نموده است. شکل ۴-۶ پتانسیل برانگیخته‌ی بازسازی شده با استفاده از مولفه‌ی پنجم تجزیه‌ی SCP را در کanal PO4 همراه با سیگنال EEG خام در همان کanal نشان می‌دهد.

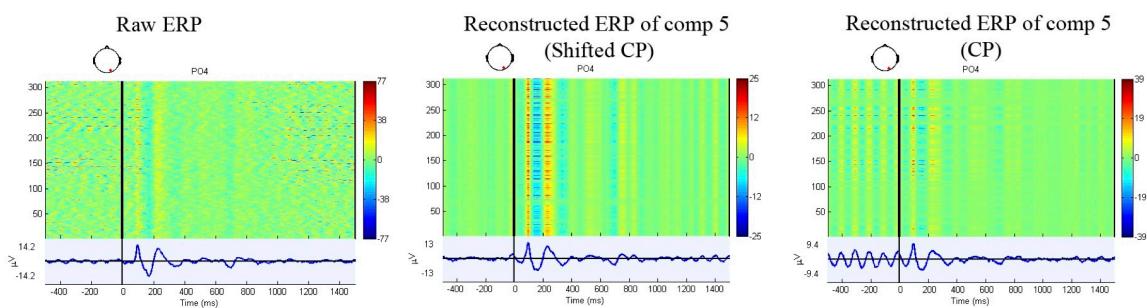
پ - تجزیه‌های GCP و WCP

نیکنظر و همکارانش در [۳۹] با تغییر تابع هزینه‌ی محاسبه‌ی تجزیه‌ی CP آن را برای جداسازی منبعی که توان بسیار کمتری از دیگر منابع دارد مقاوم‌تر ساخته‌اند. بنابراین هدف این پژوهش جداسازی یک منبع با توان کم از میان ترکیب چندین منبع و نویز است؛ که در واقع منابع دیگر برای منبع مورد نظر نقش نویز را بازی می‌کنند. کاربرد اصلی این مقاله در جداسازی سیگنال جنین از مادر معرفی شده است. اما از آن برای استخراج ERP هم استفاده شده است.

در رابطه‌ی ۳-۶۰ از بخش ۳-۴-۱ دیدیم تابع هزینه‌ی محاسبه‌ی تجزیه‌ی CP نرم تانسور خطاست. حالی از لطف نیست تا این رابطه را در اینجا برای تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times J \times K}$ بازنویسی کنیم و در آن ماتریس‌های عامل را به وسیله‌ی A ، B و C نمایش دهیم:



شکل ۴-۵: مولفه‌های تجزیه‌ی CP و SCP تانسور سه بُعدی کانال×زمان×تکرار در مقاله‌ی [۹]. در هر مولفه‌ی CP پروفایل مکانی، زمانی و شدت آن در تکرارهای آزمایش دیده می‌شود. در مولفه‌های SCP علاوه بر این موارد بردار شیفت را هم می‌توان داشت. در شکل هیستوگرام ستون‌های ماتریس عامل سوم و چهارم نمایش داده شده است.



شکل ۴-۶: تصویر سمت چپ: سیگنال EEG خام در کanal PO4 برای تکرارهای مختلف. تصویر وسط و سمت راست: پتانسیل برانگیخته‌ی بازسازی شده با استفاده از مولفه‌ی پنجم تجزیه‌ی SCP و CP در همان کanal برای تکرارهای مختلف. برای بازسازی ERPها از روی SCPها k ام، برای تکرار k ام، (پروفایل زمانی پنجم) در (شدت این پروفایل زمانی در تکرار k ام) ضرب شده و سپس به اندازه‌ی $\tau_{k,5}$ شیفت داده شده است. می‌بینیم که فعالیت برانگیخته در همه‌ی تکرارها سنکرون شده است. مشخص است که SCP توانسته فعالیت وابسته به رخداد را جداسازی کند و مولفه‌های P100 و N200 را به خوبی می‌توان در پتانسیل بازسازی شده دید. در حالی که CP تحت تاثیر فعالیت آلفای قبل تحریک است [۹].

$$\{\mathbf{A}, \mathbf{B}, \mathbf{C}\} = \operatorname{argmin}_{\{\mathbf{A}, \mathbf{B}, \mathbf{C}\}} \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K (x_{i,j,k} - \sum_{r=1}^R a_{i,r} b_{j,r} c_{k,r})^2 \quad (7-4)$$

در این رابطه اگر با استفاده از ماتریس‌های عامل تجزیه‌ی CP، تansور داده‌ها را بازسازی کنیم، به

عبارتی قرار دهیم:

$$\hat{\mathcal{X}} = [\mathbf{A}, \mathbf{B}, \mathbf{C}] \quad (8-4)$$

تابع هزینه به صورت

$$J = \|\mathcal{E}\|_F^2 = \sum_{i,j,k} (x_{i,j,k} - \hat{x}_{i,j,k})^2 \quad (9-4)$$

$$s.t. \quad \mathcal{E} = \mathcal{X} - \hat{\mathcal{X}}$$

خواهد بود و \mathcal{E} را تansور خطای نامیم و

هر یک از درایه‌های $\hat{\mathcal{X}}$ در تابع هزینه‌ی رابطه‌ی ۷-۴ به میزانی تاثیر دارند. در واقع خطاهای خیلی بزرگ مربوط به درایه‌هایی است که به میزان زیادی تحت تاثیر نویز (و یا سایر منابع) بوده‌اند و به شدت تابع هزینه‌ی محاسبه‌ی CP را تحت تاثیر قرار می‌دهند و به عبارتی به تابع هزینه اجازه نمی‌دهند روی منبع مورد نظر تمرکز کند.

با توجه به همین نکته، دو نسخه‌ی تصحیح شده از تجزیه‌ی CP که نسبت به داده‌های نویزی مقاوم‌سازی شده‌اند با نام‌های ^{۲۰}GCP و ^{۲۱}WCP معرفی شده‌اند. شکل ۴-۷ تابع هزینه‌ی CP، GCP و WCP را به ازای هر درایه از تansور خطای نشان می‌دهد.

در تجزیه‌ی GCP تابع هزینه به صورت زیر تغییر داده شده است:

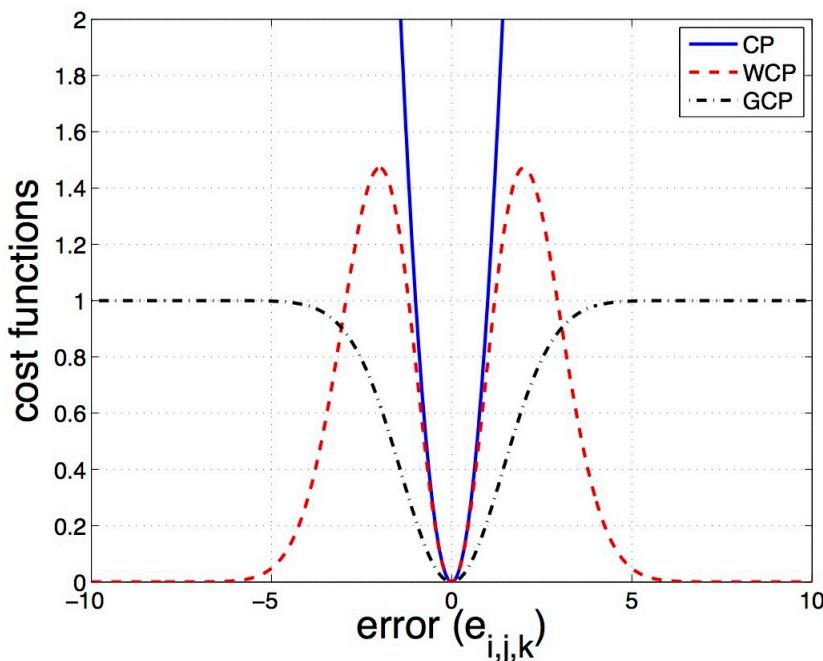
$$J = \sum_{i,j,k} \psi(x_{i,j,k} - \hat{x}_{i,j,k}) \quad (10-4)$$

که در آن $\psi(u) = 1 - \exp\left(-\frac{u^2}{2\sigma^2}\right)$. در این تابع هزینه، خطای مقدار یک محدود شده است و تاثیر نقاطی که خطای بالایی دارند کنترل شده است (بنابراین نقاطی که خطای بیشتر از ۳۵ دارند سهمی (تقریبا) برابر یک در تابع هزینه دارند). لازم به یادآوری است که این نقاطی که خطای بالایی دارند در واقع همان نقاطی هستند که به شدت تحت تاثیر نویز بوده‌اند و تمرکز تابع هزینه را بر روی جداسازی منبع تحتالشعاع قرار می‌دادند.

در تجزیه‌ی WCP تابع هزینه‌ی CP به صورت زیر تغییر داده شده است:

^{۲۰} Gaussian CP

^{۲۱} Weighted CP



شکل ۷-۴: توابع هزینه‌ی مربوط به تجزیه‌ی CP و دو تجزیه‌ی WCP و GCP معرفی شده در [۳۹].

$$J = \sum_{i,j,k} \|w_{ijk} (x_{ijk} - \hat{x}_{i,j,k})\|_F^2 \quad (11-4)$$

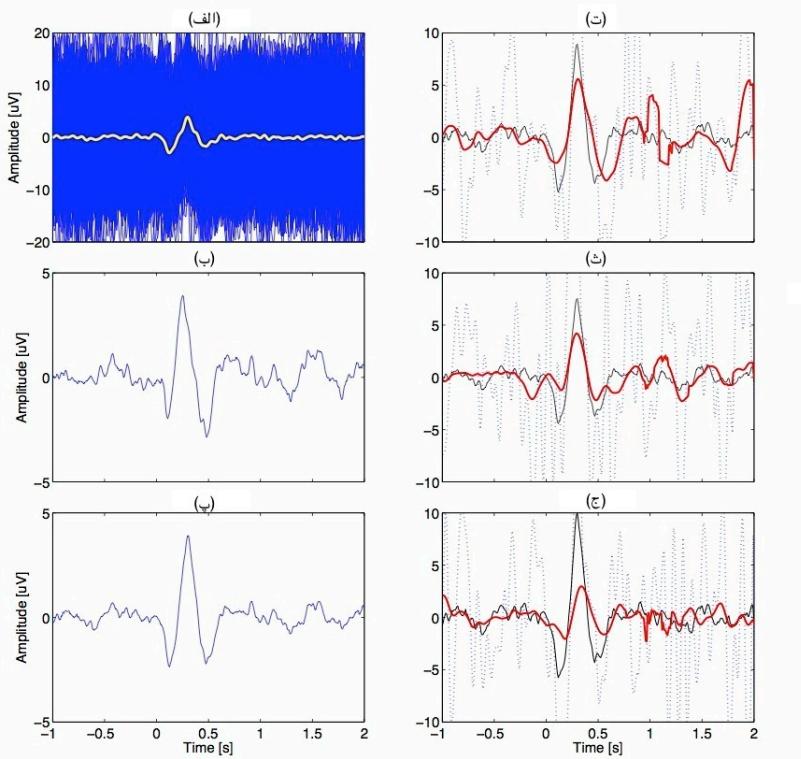
که در آن

$$w_{ijk} = \exp \left\{ -\frac{(x_{ijk} - \mu_{ij})^2}{\sigma_{ij}^2} \right\} \quad (12-4)$$

روشن است که با تعریف وزن‌ها به صورتی که در خطاهای بزرگ مقدار صفر داشته باشند، در این تابع هزینه تاثیر داده‌های پرت (خطاهای خیلی بزرگ) حذف شده است (شکل ۷-۴).

علاوه بر آن، تجزیه‌های فوق با فیلتر کالمن ترکیب شده‌اند؛ که بررسی آن در حوصله‌ی این نوشتار نیست.

از این تجزیه‌ها برای جداسازی منبع ERP در داده‌های P300 Speller استفاده شده است. برای ساخت تانسور از داده‌ها، ابتدا تکرارهای Speller جداسازی شده‌اند (هر تکرار شامل نمونه‌های زمانی یک ثانیه پیش از تحریک تا دو ثانیه پس از آن است). اکنون اگر این تکرارها در یک تانسور $channel \times time \times trial$ چیده شوند و بر آن تجزیه‌ی CP (یا WCP و GCP) انجام شود، ماتریس پایه‌ی مُد اول چگونگی توزیع مکان منابع را در کانال‌ها نشان می‌دهد و ماتریس عامل دوم الگوهای زمانی منابع استخراج شده و ماتریس سوم به نوعی شدت وقوع هریک از این الگوها را در هر تکرار نشان می‌دهند. شکل ۸-۴ نتایج اعمال تجزیه‌ی WCP با دو مولفه را بر داده‌های یک P300 Speller نشان می‌دهد.



شکل ۴-۸: استخراج ERP با استفاده از تجزیه‌ی WCP. (الف) ۵۰۰ تکرار همراه با میانگین آن‌ها. (ب) و (پ) دو ستون ماتریس عامل دوم که الگوهای زمانی استخراج شده هستند. (ت) و (ج) سه تکثیت بازسازی شده به وسیله‌ی ماتریس‌های عامل به دست آمده از WCP. نقاط نقطه‌چین نمونه‌های زمانی سیگنال اصلی، خط سیاه پتانسیل استخراج شده به وسیله‌ی WCP و خط قرمز و ضخیم پتانسیل استخراج شده به وسیله‌ی WCP همراه با فیلتر کالمون است.^[۳۹]

۲-۱-۳-۴ کاهش ویژگی

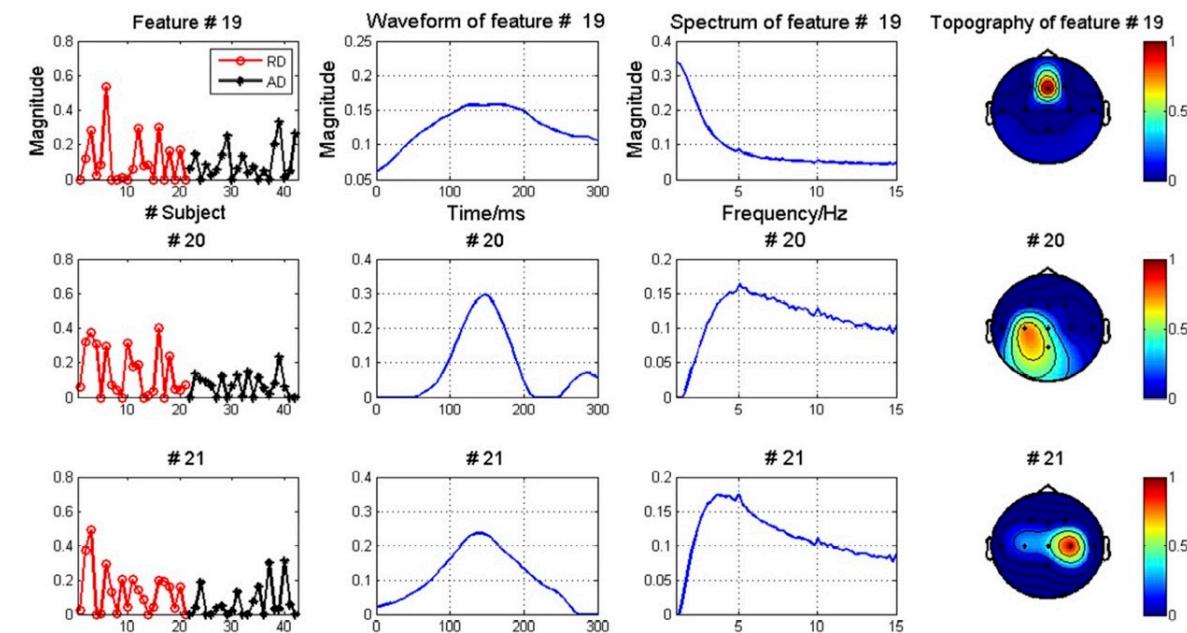
الف - تمايزپذيری

Cong و همکارانش در [۱۰] از تجزیه‌ی NTF برای استخراج ویژگی‌ای که از لحاظ آماری تفکیک‌پذیری معناداری میان دو گروه از کودکان دارای AD^{۲۲} و RD^{۲۳} ایجاد کند، استفاده کردند. در این پژوهش از صوت بدون وقفه، تحت oddball paradigm، استفاده شده تا پتانسیل وابسته به رخداد MMN^{۲۴} برانگیخته شود. آن‌ها به وسیله‌ی تجزیه‌ی تانسور مرتبه چهار وابسته به رخداد Time × Frequency × Channel × Subjects ویژگی‌های مورد نظر را از فاکتور چهارم تجزیه‌ی CP نامنفی استخراج کردند. در بخش ۳-۱-۵-۲ دیدیم که برای استخراج ویژگی با استفاده از تجزیه‌ی CP ماتریس عامل بُعدی که داده‌های کلاس‌های مختلف در آن قرار دارند، ویژگی‌های مورد نظر را در خود دارد. پس در

^{۲۲}Attention Deficit

^{۲۳}Reading Disability

^{۲۴}MisMatch Negativity



شکل ۴-۹: ویژگی‌های چند-حوزه همراه با پروفایل‌های زمانی-فرکانسی و مکانی آن‌ها مربوط به [۱۰].

اینجا تجزیه با R مولفه به شکل زیر خواهد بود:

$$\mathcal{X} = \mathcal{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \times_4 \mathbf{F} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \circ \mathbf{f}_r \quad (13-4)$$

که اگر $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times K}$ باشد، $\mathbf{A} \in \mathbb{R}^{I_1 \times R}$ ماتریس پایه متناظر با مُد زمان، $\mathbf{B} \in \mathbb{R}^{I_2 \times R}$ ماتریس پایه متناظر با مُد فرکانس، $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$ ماتریس پایه متناظر با مُد مکان و $\mathbf{F} \in \mathbb{R}^{K \times R}$ ماتریس ویژگی‌ها خواهد بود که هر یک از ستون‌های آن مربوط به یک ویژگی است. از رابطه‌ی ۱۳-۴ می‌بینیم که ستون ایم از \mathbf{F} با ستون ایم از هر یک از سه ماتریس پایه در ارتباط است. این ویژگی تجزیه‌ی CP را این‌طور عنوان می‌کنیم که ویژگی ایم دارای پروفایل زمانی a_r ، پروفایل فرکانسی b_r و پروفایل مکانی c_r است. این پروفایل‌ها در واقع خصوصیات f_r را در حوزه‌ی زمان، فرکانس و مکان نشان می‌دهند. به همین دلیل است که به ویژگی‌های استخراج شده از این طریق «ویژگی‌های چند-حوزه»^{۲۵} گفته شده است. برای مثال شکل ۹-۴ سه ویژگی را همراه با پروفایل‌های زمانی، فرکانسی و مکانی‌اش نشان می‌دهد.

با توجه به هدف مقاله که یافتن یک ویژگی است که تفکیک معناداری بین دو گروه کودکان ایجاد کند، لازم است از میان R ویژگی به دست‌آمده از تجزیه‌ی NTF انتخاب ویژگی صورت گیرد. برای این کار روش دو مرحله‌ای زیر پیشنهاد شده است:

۱. همه‌ی ویژگی‌ها مورد آزمون فرضیه‌ی ANOVA (با فرض صفر عدم وجود تفاوت معنادار میان دو

^{۲۵}Multi-domain feature

کلاس) قرار گرفته و آن‌هایی که p-value کمتر از ۰/۰۵ دارند جدا می‌شوند. در واقع ویژگی‌هایی که از این آزمون فرضیه انتخاب می‌شوند، فرض صفر را رد کرده و می‌توانند تمایز معناداری را میان دو کلاس ایجاد کنند.

۲. از میان ویژگی‌های مرحله‌ی یک، آن ویژگی‌ای که پروفایل فرکانسی و زمانی مشابه پتانسیل وابسته به رخداد مورد نظر را دارد به عنوان ویژگی نهایی انتخاب می‌شود.

با استفاده از این فرآیند انتخاب ویژگی، ویژگی شماره‌ی ۲۰ از شکل ۹-۴ انتخاب می‌شود. در این پژوهش برای تعیین تعداد مولفه‌های تجزیه‌ی CP از روش DIFFIT استفاده شده است. برای این منظور تعداد مولفه‌های تجزیه را از $m=1$ تا $m=M$ تغییر می‌دهیم و برای هر m پارامتر $fit(m)$ را به صورت زیر محاسبه می‌کنیم:

$$fit(m) = 1 - \|\mathcal{X} - \hat{\mathcal{X}}_m\|_F / \|\mathcal{X}\|_F$$

که در آن $\hat{\mathcal{X}}_m$ تخمین CP با m مولفه از \mathcal{X} است. سپس برای هر $m = 2, \dots, M$ پارامتر $diff(m)$ را به دست می‌آوریم:

$$diff(m) = fit(m) - fit(m-1)$$

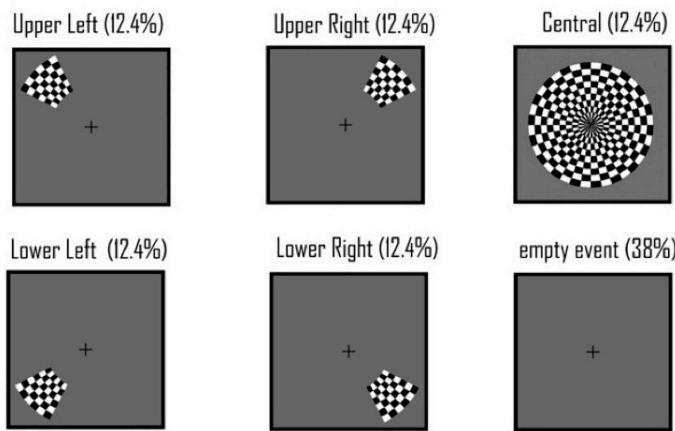
و در مرحله‌ی آخر معیار $diffit$ را محاسبه می‌کنیم:

$$diffit(m) = diff(m) / diff(m+1)$$

تعداد مولفه‌های تجزیه برابر با m ای انتخاب می‌شود که بیشترین $diffit$ را داشته باشد. پژوهش مشابهی در [۴۰] انجام شده است که در آن از NTF برای بررسی تمایزپذیری دو گروه بزرگسال سالم و دارای مشکل افسردگی استفاده شده است. از آنجا که روش آن مشابه آنچه در بالا بیان کردیم است، از توضیح آن می‌گذریم.

ب - طبقه‌بندی

در [۱۱] از تجزیه‌ی CP برای کاهش ویژگی از داده‌های EEG و fMRI یک آزمایش بصری از ۱۹ سوژه استفاده کردند. تحریک‌های این آزمایش مانند شکل ۱۰-۴ هستند. در [۴۱] نشان داده شده است که پتانسیل وابسته به هر یک از تحریک‌ها ویژگی‌های خاصی از خود نشان می‌دهد؛ به طوری که می‌توان آن‌ها را تفکیک کرد.

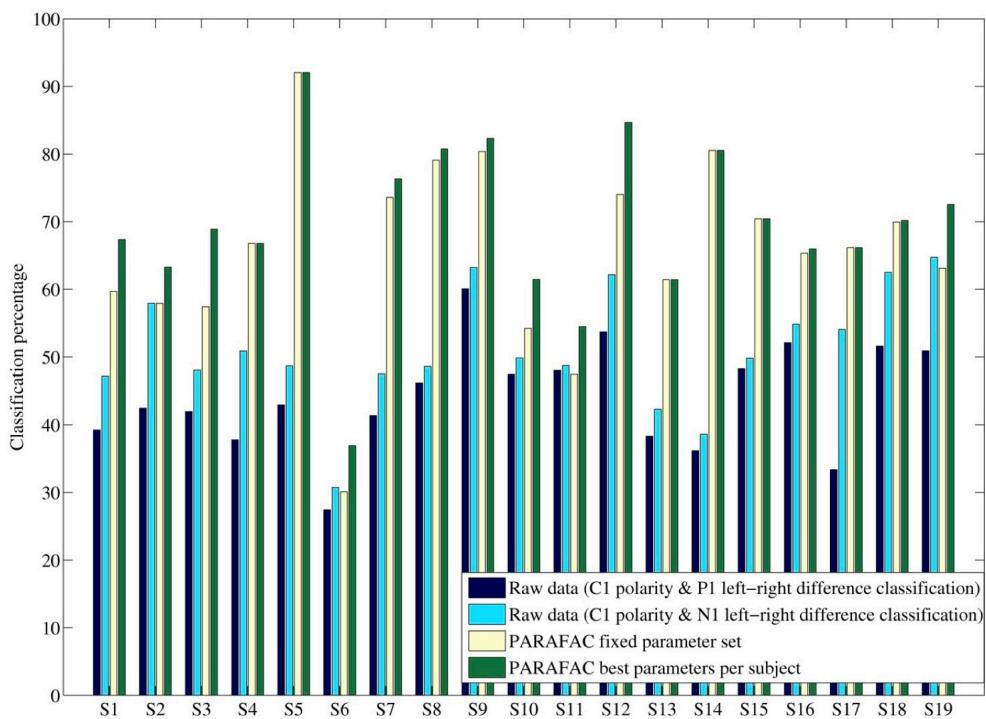


شکل ۴-۱۰: تحریک‌های آزمایش مربوط به [۱۱]. این آزمایش از [۴۱] است.

داده‌ها پس از پیش‌پردازش در یک تانسور سه بعدی چیده شده‌اند که مُدهای آن $Channel \times Time \times Trials$ است. در این پژوهش برای جلوگیری از انحطاط CP فرض تعامل بر مُد آن اعمال شده است. پس از تجزیه‌ی تانسور داده‌ها به وسیله‌ی CP، ماتریس عامل سوم آن به عنوان ویژگی به طبقه‌بندی داده می‌شود. دو نوع طبقه‌بندی انجام شده: ۱) طبقه‌بندی دوکلاسه بین تحریک‌هایی که قسمت شطرنجی در نیمه‌ی سمت راست/چپ تصویر قراردارد. ۲) طبقه‌بندی چهار کلاسه بین تحریک‌هایی که قسمت شطرنجی در سمت راست-بالا/چپ-بالا/راست-پایین/چپ-پایین تصویر قراردارد. در هر دو حالت ویژگی‌های CP از ویژگی‌های زمانی که در [۴۱] معرفی شده، تفکیک‌پذیری بسیار بیشتری میان کلاس‌ها ایجاد کرده است. از بررسی جزئی نتایج این پژوهش خودداری می‌کنیم، با این وجود برای آن‌که برتری مشهود CP بر ویژگی‌های زمانی برای خواننده مشخص شود یکی از نمودارهای این مقاله در شکل ۱۱-۴ آورده شده است.

یکی دیگر از مقاله‌هایی که از تجزیه‌ی CP برای کاهش ویژگی از داده‌های ERP استفاده کرده مقاله‌ی [۴۲] است. آزمایش این پژوهش شامل تحریک‌های صوتی‌ای بوده است که پتانسیل MMN را در EEG بر می‌انگیزد. با استفاده از تجربه‌ی NTF ویژگی‌هایی استخراج شده است که قادر به طبقه‌بندی دو گروه از کودکان سالم و دارای AD^{۲۶} باشد. روش مرسوم طبقه‌بندی این دو گروه با استفاده از پیک MMN در کودکان AD کوچک‌تر از کودکان سالم است. اما این ویژگی مقاوم نیست و تحت تاثیر جنسیت، میانگین سنی و تعداد کودکان دو گروه است. برای مثال این ویژگی نتوانسته کودکان مورد

^{۲۶}Attention Deficit



شکل ۱۱-۴: دقت طبقه‌بندی چهار کلاسه در [۱۱]. دو ستون اول مربوط به ویژگی‌های زمانی هستند. روشن است که ویژگی‌های CP تفکیک‌پذیری بسیار خوبی میان کلاس‌ها ایجاد کرده‌اند.

آزمایش این مقاله را جداسازی کند.

تansور مرتبه چهار با ابعاد $Time \times Frequency \times Channel \times Subjects$ از داده ساخته شده است. در هر کanal تبدیل زمان-فرکانس از میانگین تکرارها گرفته شده است. هم‌چنین در مُد چهارم داده‌های مربوط به سوژه‌های دو گروه سالم و AD قرار دارد. برای روشن‌تر شدن موضوع، برای هر سوژه یک داده‌ی تansوری $9 \times 130 \times 256 \times 256$ متناظر با $Time \times Frequency \times Channel$ وجود دارد. از هر گروه ۲۰ سوژه داریم. بنابراین تansوری که تجزیه می‌شود $40 \times 9 \times 256 \times 130$ است. پس از تجزیه‌ی NTF با ۹ مولفه، ماتریس عامل چهارم (که 9×40 است) به عنوان ماتریس ویژگی برای آموزش به طبقه‌بند KNN داده شده است. اکنون هر کودک جدید که برای طبقه‌بندی داشته باشیم، یک تansور ماتریس عامل چهارم آن (که 9×1 است) را به طبقه‌بند می‌دهیم تا کلاس آن را تعیین کند. با این روش دقت ۷۶ درصد در طبقه‌بندی دو گروه به دست آمده است. هرچند این دقت به نظر کم می‌آید، اما با توجه به عدم توانایی روش‌های مرسوم کاهش ویژگی برای تفکیک دو گروه، این دقت قابل توجه خواهد بود.

۲-۳-۴ کاربرد تجزیه‌ی Tucker در پردازش داده‌های حاوی پتانسیل وابسته به رخداد

در این بخش به مرور مقالاتی می‌پردازیم که از تجزیه‌ی Tucker برای پردازش داده‌های ERP استفاده کردند. در اینجا لازم است بار دیگر مراد از «تجزیه‌ی Tucker با سرپرست و بدون سرپرست» توضیح داده شود. در اینجا وقتی از تجزیه‌ی با سرپرست صحبت می‌کنیم منظور این است که از برچسب‌های داده‌های آموزش برای به دست آوردن ماتریس‌های زیرفضای Tucker استفاده شده است. اما در صورتی که از برچسب‌ها استفاده نکنیم می‌گوییم تجزیه‌ی بدون سرپرست انجام داده‌ایم. در حالت دوم ممکن است حتی داده‌ی آزمون هم داشته باشیم. ولی سرپرست داشتن تجزیه مربوط به این است که آیا برچسب‌ها را در تجزیه‌ی داده‌های آموزش دخالت داده‌ایم یا نه.^{۲۷}.

۱-۲-۳-۴ کاربرد تجزیه‌ی Tucker بدون سرپرست

در [۴۳]^{۲۸} کانگ و همکارانش از تجزیه‌ی NTD بدون سرپرست برای طبقه‌بندی دو گروه از کودکان سالم و RD استفاده کردند. تحریک استفاده شده به صورت شناوی بوده و منجر به برآنگیخته شدن EEG در MMN می‌شود. از هر گروه از کودکان ۲۱ نفر در آزمایش شرکت داشتند. در این پژوهش برای ارزیابی میزان تفکیک‌پذیری دو دسته ویژگی دو آزمایش انجام شده است. ابتدا همه‌ی داده‌ها در یک تانسور چهاربعدی $Time \times Frequency \times Channel \times Subjects$ قرار داده شده‌اند. تبدیل زمان-فرکانس از میانگین تکرارهای تحریک گرفته شده است؛ پس برای هر فرد یک تانسور سه بعدی داریم و در بُعد چهارم داده‌های همه‌ی افراد که از دو کلاس قرار دارند. این تانسور به وسیله‌ی NTD تجزیه شده است و ویژگی‌های هر فرد استخراج شده است. سپس یک اعتبارسنجی متقابل انجام شده است به این صورت که هر بار یک نفر را برای آزمون و بقیه را برای آموزش استفاده کرده‌اند.^{۲۹} برای طبقه‌بندی از LDA و SVM استفاده شده است که از Adaboost بیشترین دقت طبقه‌بندی برابر ۹۰ درصد را به دست آورده‌اند. سپس در آزمایش دوم از هر گروه ۱ نفر را کنار گذاشتند و بین چهل نفر باقی‌مانده اعتبارسنجی مشابهی انجام داده‌اند. می‌توان به این روش ۴۴۱ زیرمجموعه از سوژه‌ها داشت. میانگین نتایج برای این ۴۴۱ زیرمجموعه از داده‌ها گزارش شده است و LDA بیشترین دقت طبقه‌بندی برابر ۷۸ درصد را داده

^{۲۷} این گفتار را قبلاً در بخش ۳-۵-۱-۱ بیان کردیم. اما در اینجا به دلیل اهمیت یادآوری این مفهوم بار دیگر آورده شده است.

^{۲۸} Reading Deficit

^{۲۹} Leave-one-out

است.

۴-۲-۳-۴ کاربرد تجزیه‌ی Tucker با سرپرست

در [۴۴] از HODA برای کاهش ویژگی از داده‌های ERP یک P300 speller استفاده شده است. می‌دانیم که در P300 speller هریک از تکثیت‌های آزمایش (روشن شدن سطر یا ستون) به یکی از کلاس‌های P300 (کلاس هدف^{۳۰}) یا کلاس غیر P300 (کلاس غیرهدف^{۳۱}) تعلق دارد. به همین دلیل می‌توان تکثیت‌های آزمایش را طبقه‌بندی دو کلاسه کنیم. داده‌های این مقاله از مسابقه‌ی سوم BCI-مجموعه دادگان دو^{۳۲} گرفته شده‌اند و دقت تشخیص کاراکتر ۹۲ درصد به صورت میانگین برای دو سوژه گزارش شده است. در زیر خلاصه‌ای از مراحل این مقاله ارائه شده است:

۱. ساخت تانسور: ابتدا به هریک از تکثیت‌ها چندجمله‌ای‌های درجه ۱۱ و ۱۴ برازش شده است.

سپس در بُعد سوم یک تانسور مرتبه سه قرار داده می‌شوند. بنابراین برای هر یک از تکثیت‌ها یک تانسور مرتبه سه $X^{(k)} \in \mathbb{R}^{I_T \times I_S \times 2}$ خواهیم داشت که در آن I_T تعداد نمونه‌های زمانی، I_S تعداد کانال‌ها و در بُعد سوم هم تخمین‌های چندجمله‌ای برازش شده قرار دارد. در این مرحله تانسورهای تمامی تکثیت‌ها را در بُعد چهارم یک تانسور پشت سر هم قرار می‌دهیم.^{۳۳}

۲. یافتن فضای پایه و کاهش ویژگی: ماتریس‌های پایه و ویژگی‌های داده‌های آموزش با استفاده از روش کاهش ویژگی تانسوری HODA (که در بخش ۳-۵-۳ توضیح داده شد) به دست آمده‌اند. سپس برای به دست آوردن ویژگی‌های داده‌های آزمون مانند آن‌چه در بخش ۳-۱-۵-۱ گفتیم، تانسور داده‌های آزمون روی زیرفضای فوق تصویر می‌شوند. در این مرحله ویژگی‌های هر تکثیت یک تانسور مرتبه سه است، آن‌ها را برای ورود به طبقه‌بند برداری می‌کنیم.

۳. طبقه‌بندی دو کلاسه به وسیله‌ی LDA انجام می‌شود.

۴. بر اساس احتمال پسین دو کلاس در تکرارهای مختلف مربوط به هر سطر (یا ستون)، کاراکتر موردنظر کاربر تشخیص داده می‌شود (توضیح چگونگی تشخیص کاراکتر در بخش ۶-۳-۶).

^{۳۰}Target class

^{۳۱}Non-target class

^{۳۲}توضیح این دادگان در فصل ۶ آورده شده است.

^{۳۳}Concatenate

در [۱۶] بار دیگر از HODA برای پردازش داده‌های P300 speller استفاده شده است. ژانگ^{۳۴} و همکارانش در این پژوهش بر روی توانایی کاهش ویژگی تانسوری برای استفاده در مسائل SSS^{۳۵} تمرکز کردند. SSS زمانی رخ می‌دهد که تعداد داده‌های آموزش کم است.^{۳۶} به این موضوع قبلاً هم در اولین مقاله‌ای که HODA معرفی شد^{۳۷} [۲۴] اشاره شده بود. دلیل این‌که روش کاهش ویژگی تانسوری (و به طور خاص HODA) در مسئله‌ی SSS بر روش‌های برداری برتری دارند را با یک مثال بیان می‌کنیم: در مسئله‌ی طبقه‌بندی دو کلاسه‌ی تکثیت‌های P300 speller فرض کنید هر تکثیت I_t نمونه دارد و در کل I_S کanal داریم. در بخش ۴-۲ دیدیم که اگر بخواهیم از روش‌های برداری استفاده کنیم باید برای هر تکثیت نمونه‌های کanal‌ها را در یک بردار پشت سر هم قرار دهیم. بنابراین تعداد ویژگی‌ها برابر $I_f = I_t \times I_S$ خواهد بود. بنابراین در LDA ماتریس‌های پراکندگی $I_f \times I_f$ خواهند بود. حال اگر تعداد داده‌ها کم‌تر از I_f باشد، ماتریس پراکندگی درون کلاسی (S_w) و ماتریس پراکندگی کل (S_b) معکوس‌ناپذیر خواهند بود (این دقیقاً مسئله‌ی SSS است). برای حل این موضوع تلاش‌های مختلفی برای رگولاریزاسیون این ماتریس‌ها انجام شده است. اما اگر بخواهیم از HODA استفاده کنیم و داده‌ها را در تانسور $K \times I_t \times I_S$ قرار دهیم (کل تکثیت‌هاست); به ازای هر مُد تانسور ماتریس پراکندگی دارای ابعاد همان مُد است. برای مثال برای مُد اول ماتریس‌های پراکندگی $I_t \times I_t$ هستند. به این ترتیب اندازه‌ی بُعد ماتریس‌های پراکندگی به میزان زیادی کوچک‌تر شده و مسئله‌ی SSS به صورت چشم‌گیری بهبود می‌یابد.

در این مقاله حالت خاص HODA برای زمانی که داده‌ها ماتریس هستند ($N = 2$) با نام STDA^{۳۸} معرفی شده است. در بخش ۳-۵-۳ توضیحاتی درباره‌ی جزئیات STDA داده شد، به همین دلیل از تکرار دوباره‌ی آن‌ها در اینجا پرهیز می‌کنیم. برای بررسی کارایی STDA برای تشخیص کلاس P300 در P300-based BCI از مجموعه دادگان دوم از مسابقه‌ی BCI سوم [۲۹] استفاده شده است و چندین روش برداری با آن مقایسه شده‌اند و نشان داده شده است که روش تانسوری زمانی که تعداد داده‌های آموزش کم است، عملکرد بسیار بهتری از روش‌های برداری دارد.

^{۳۴}Zhang

^{۳۵}Small Sample Size

^{۳۶}به عبارتی تعداد داده‌های آموزش از تعداد ویژگی‌ها کم‌تر است.
^{۳۷}با نام DATER

^{۳۸}Spatial Temporal Discriminant Analysis

۴-۴ جمع‌بندی

در این فصل ابتدا روش‌های برداری‌ای که در طبقه‌بندی P300 speller مجموعه دادگان BCI مورد استفاده قرار گرفته‌اند را بدون ارائه نتایج بررسی کردیم (نتایج در Competition III-Dataset II جدول ۲-۶ آورده شده است). پس از آن به مرور کاربرد روش‌های تانسوری در مطالعه‌ی داده‌های ERP پرداختیم. دیدیم که در ادبیات این حوزه دو تجزیه‌ی CP و Tucker برای جداسازی منابع و کاهش ویژگی مورد استفاده قرار گرفته است؛ برای هر گروه از این مقالات دست کم یک نمونه به تفصیل توضیح داده شد.

فصل ۵

HOSRDA: روش پیشنهادی:

۱-۵ پیشگفتار

در بخش ۳-۵ دیدیم HODA، که یکی از روش‌های استخراج ویژگی تانسوری است، تعمیمی چندخطی از روش کاهش ویژگی LDA است. در این روش نیز همانند LDA زیرفضای تفکیک‌کننده‌ی کلاس‌ها به وسیله‌ی اعمال تجزیه‌ی مقدارویژه-بردارویژه بر ماتریس‌های پراکنده‌ی کلاس‌ها به دست می‌آید. اما HODA هم از نقاط ضعفی شبیه LDA رنج می‌برد: هرچند که در ادبیات از HODA به عنوان روشی برای رفع دغدغه‌ی معکوس‌ناپذیری ماتریس پراکنده‌ی می‌شود [۲۴] اما باز هم گفته می‌شود که در تکرارهای اولیه‌ی آن امکان معکوس‌ناپذیری ماتریس پراکنده‌ی وجود دارد [۲۰]. این موضوع در تجزیه‌ی مقدارویژه-بردارویژه مشکل‌ساز خواهد بود و به کاهش پایداری آن می‌انجامد. هم‌چنین در مواردی که بعد داده‌ها زیاد باشد، تجزیه‌ی مقدارویژه-بردارویژه پیچیدگی محاسباتی بالایی دارد.

در فصل ۳ دیدیم روش SRDA این ضعفها را در LDA مورد بررسی قرار داده و با تبدیل تجزیه‌ی مقدارویژه-بردارویژه به حل دستگاه معادلات خطی آن‌ها را بهبود داده است. ما نیز با ایده از SRDA بر آن شدیم تا آن را به داده‌های تانسوری تعمیم داده و در HODA مسئله‌ی تجزیه‌ی مقدارویژه-بردارویژه را با یک دستگاه معادلات خطی جایگزین کنیم. با این مقدمه، در ادامه‌ی این فصل تعمیم چندخطی SRDA را پیشنهاد داده و آن را (HOSRDA) Higher Order Spectral Regression Discriminant Analysis می‌نامیم. در بخش ۲-۵ فرمول‌بندی ریاضی روش پیشنهادی به تفصیل بیان و الگوریتم پیشنهادی ارائه خواهد شد. در بخش ۳-۵ پیچیدگی محاسباتی آن بررسی شده است. در بخش پایانی جمع‌بندی فصل را

خواهیم داشت.

۲-۵ فرمول‌بندی ریاضی و الگوریتمی روش پیشنهادی

در بخش ۳-۵-۱ مسئله‌ی کاهش ویژگی تانسوری را بیان کردیم. در اینجا نیز در پی حل آن مسئله هستیم.

برای تعمیم SRDA به داده‌های تانسوری، لازم است مسئله‌ی HODA را بازبینی کنیم. در فصل قبل دیدیم که در HODA برای هر $U^{(n)}, n = 1, \dots, N$ از مسئله‌ی بهینه‌سازی زیر به دست می‌آید:

$$\begin{aligned} U^{(n)} &= \underset{U^{(n)}}{\operatorname{argmax}} \frac{\operatorname{tr}\{U^{(n)T} S_b^{-n} U^{(n)}\}}{\operatorname{tr}\{U^{(n)T} S_w^{-n} U^{(n)}\}} \\ &\text{s.t.} \quad U^{(n)T} U^{(n)} = I \end{aligned} \quad (1-5)$$

که یک مسئله‌ی GEVD است. فرم ماتریس پراکندگی درون‌کلاسی به صورت زیر به دست آمد:

$$S_b^{-n} = \sum_{c=1}^C \left\langle \check{Z}^{(c)-n}, \check{Z}^{(c)-n} \right\rangle_{-n} \quad (2-5)$$

که در آن

$$\check{Z}^{(c)-n} = \check{\mathcal{X}}^{(c)} \times_{-n} \{U^T\} \quad (3-5)$$

$$\check{\mathcal{X}}^{(c)} = \sqrt{K_c} \left(\bar{\mathcal{X}}^{(c)} - \bar{\bar{\mathcal{X}}} \right) \quad (4-5)$$

همچنین ماتریس پراکندگی بین‌کلاسی به صورت زیر فرموله شد:

$$S_w^{-n} = \sum_{i=1}^K \left\langle \tilde{Z}^{(i)-n}, \tilde{Z}^{(i)-n} \right\rangle_{-n} \quad (5-5)$$

که در آن:

$$\tilde{Z}^{(i)-n} = \tilde{\mathcal{X}}^{(i)} \times_{-n} \{U^T\} \quad (6-5)$$

$$\tilde{\mathcal{X}}^{(i)} = \mathcal{X}^{(i)} - \bar{\mathcal{X}}^{(c_i)} \quad (7-5)$$

اکنون رابطه‌ی ۲-۵ را بازنویسی می‌کنیم:

$$\begin{aligned} \mathbf{S}_b^{-n} &= \sum_{c=1}^C \left\langle \check{\mathcal{Z}}^{(c)}_{-n}, \check{\mathcal{Z}}^{(c)}_{-n} \right\rangle_{-n} = \sum_{c=1}^C K_c \left\langle (\bar{\mathcal{X}}^{(c)} - \bar{\bar{\mathcal{X}}}) \times_{-n} \{\mathbf{U}^T\}, (\bar{\mathcal{X}}^{(c)} - \bar{\bar{\mathcal{X}}}) \times_{-n} \{\mathbf{U}^T\} \right\rangle_{-n} \\ &= \sum_{c=1}^C \frac{1}{K_c} \left\langle \sum_{i \in \Omega_c} ((\mathcal{X}^{(i)} - \bar{\bar{\mathcal{X}}}) \times_{-n} \{\mathbf{U}^T\}), \sum_{i \in \Omega_c} ((\mathcal{X}^{(i)} - \bar{\bar{\mathcal{X}}}) \times_{-n} \{\mathbf{U}^T\}) \right\rangle_{-n} \quad (8-5) \\ &= \sum_{c=1}^C \frac{1}{K_c} \left\langle \sum_{i \in \Omega_c} \mathcal{H}_i^{(c)}, \sum_{i \in \Omega_c} \mathcal{H}_i^{(c)} \right\rangle_{-n} \end{aligned}$$

که در آن $i \in \Omega_c$ که $\mathcal{H}_i^{(c)} = (\mathcal{X}_i - \bar{\bar{\mathcal{X}}}) \times_{-n} \{\mathbf{U}^T\}$ پس:

$$\mathbf{S}_b^{-n} = \sum_{c=1}^C \frac{1}{K_c} \left(\sum_{i \in \Omega_c} \mathbf{H}_{i(n)}^{(c)} \right) \left(\sum_{j \in \Omega_c} \mathbf{H}_{j(n)}^{(c)} \right) = \sum_{c=1}^C \mathbf{H}_{(n)}^{(c)} \mathbf{W}^{(c)} \mathbf{H}_{(n)}^{(c)T} \quad (9-5)$$

که در آن $\mathcal{H}^{(c)} = \text{cat} \left(N + 1, \mathcal{H}_1^{(c)}, \dots, \mathcal{H}_{K_c}^{(c)} \right)$

و $\mathbf{W}^{(c)}$ یک ماتریس بلوکی به صورت زیر است:

$$\mathbf{W}^{(c)} = \frac{1}{K_c} \begin{bmatrix} \mathbf{I}_{P_n} & \mathbf{I}_{P_n} & \cdots & \mathbf{I}_{P_n} \\ \mathbf{I}_{P_n} & \mathbf{I}_{P_n} & \cdots & \mathbf{I}_{P_n} \\ \dots & \dots & \ddots & \vdots \\ \mathbf{I}_{P_n} & \mathbf{I}_{P_n} & \cdots & \mathbf{I}_{P_n} \end{bmatrix} \quad (10-5)$$

در رابطه‌ی بالا I_{P_n} ماتریس یکانی $P_n \times P_n$ است که همچنین $\mathbf{W}^{(c)}$ یک ماتریس

است $KP_n \times KP_n$

اکنون اگر قرار دهیم و $\mathcal{H}^{(c)} = \text{cat} (N + 1, \mathcal{H}^{(1)}, \dots, \mathcal{H}^{(C)})$

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}^{(1)} & \cdot & \cdots & \cdot \\ \cdot & \mathbf{W}^{(2)} & \cdots & \cdot \\ \cdots & \cdots & \ddots & \vdots \\ \cdot & \cdot & \cdots & \mathbf{W}^{(C)} \end{bmatrix} \quad (11-5)$$

رابطه‌ی ۹-۵ را می‌توان به صورت زیر بازنویسی کرد:

$$\mathbf{S}_b^{-n} = \mathbf{H}_{(n)}^{-n} \mathbf{W} \mathbf{H}_{(n)}^{-nT} \quad (12-5)$$

اکنون نشان می‌دهیم $S_w^{-n} = \mathbf{H}_{(n)}^{-n} \mathbf{L} \mathbf{H}_{(n)}^{-nT}$ داریم:

$$\begin{aligned}
S_w^{-n} &= \sum_{i=1}^K \left\langle \tilde{\mathcal{Z}}^{(i)}_n, \tilde{\mathcal{Z}}^{(i)}_n \right\rangle_{-n} = \sum_{i=1}^K \left\langle \left(\mathcal{X}^{(i)} - \bar{\mathcal{X}}^{(c_i)} \right) \times_{-n} \{U^T\}, \left(\mathcal{X}^{(i)} - \bar{\mathcal{X}}^{(c_i)} \right) \times_{-n} \{U^T\} \right\rangle_{-n} \\
&= \sum_{i=1}^K \left\langle \left(\mathcal{X}^{(i)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\}, \left(\mathcal{X}^{(i)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\} \right\rangle_{-n} + S_b^{-n} - \dots \\
&\quad \vdash \sum_{i=1}^K \left\langle \left(\mathcal{X}^{(i)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\}, \left(\bar{\mathcal{X}}^{(c)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\} \right\rangle_{-n} \\
&\quad \quad \quad (13-5)
\end{aligned}$$

ترم‌های اول و سوم رابطه‌ی بالا را به ترتیب A و B می‌نامیم و داریم:

$$A = \sum_{c=1}^C \sum_{i \in \Omega_c} \left\langle \mathcal{H}_i^{-n(c)}, \mathcal{H}_i^{-n(c)} \right\rangle_{-n} = \sum_{c=1}^C \sum_{i \in \Omega_c} \mathbf{H}_{i(n)}^{-n(c)} \mathbf{H}_{i(n)}^{-n(c)T} = \mathbf{H}_{(n)}^{-n} \mathbf{H}_{(n)}^{-nT} \quad (14-5)$$

$$\begin{aligned}
B &= \vdash \sum_{c=1}^C \left\langle \sum_{i \in \Omega_c} \left(\mathcal{X}^{(i)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\}, \left(\bar{\mathcal{X}}^{(c)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\} \right\rangle_{-n} \\
&= \vdash \sum_{c=1}^C K_c \left\langle \left(\bar{\mathcal{X}}^{(c)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\}, \left(\bar{\mathcal{X}}^{(c)} - \bar{\mathcal{X}} \right) \times_{-n} \{U^T\} \right\rangle_{-n} \\
&= \vdash S_b^{-n} \quad (15-5)
\end{aligned}$$

بنابراین ۱۳-۵ به صورت زیر خواهد بود:

$$S_w^{-n} = \mathbf{H}_{(n)}^{-n} \mathbf{H}_{(n)}^{-nT} - S_b^{-n} = \mathbf{H}_{(n)}^{-n} (\mathbf{I}_{KP_n} - \mathbf{W}) \mathbf{H}_{(n)}^{-nT} = \mathbf{H}_{(n)}^{-n} \mathbf{L} \mathbf{H}_{(n)}^{-nT} \quad (16-5)$$

که \mathbf{L} ماتریس یکانی $I_{KP_n} \times KP_n$ است.

$S_t^{-n} = S_w^{-n} + S_b^{-n}$ و داشته باشیم $S_t^{-n} = \mathbf{H}_{(n)}^{-n} \mathbf{H}_{(n)}^{-nT}$ با توجه به نتایج بالا می‌توانیم تعریف کنیم

حال می‌توان مسئله‌ی ۱-۵ را با مسئله‌ی زیر معادل قرار دهیم:

$$\begin{aligned}
\mathbf{U}^{(n)} &= \underset{\mathbf{U}^{(n)}}{\operatorname{argmax}} \frac{\operatorname{tr}\{\mathbf{U}^{(n)T} S_b^{-n} \mathbf{U}^{(n)}\}}{\operatorname{tr}\{\mathbf{U}^{(n)T} S_t^{-n} \mathbf{U}^{(n)}\}} \\
&\text{s.t.} \quad \mathbf{U}^{(n)T} \mathbf{U}^{(n)} = \mathbf{I} \quad (17-5)
\end{aligned}$$

می‌توان نشان داد که $\operatorname{tr}\{\mathbf{U}^{(n)T} S_t^{-n} \mathbf{U}^{(n)}\} = \sum_{i=1}^K \|\mathcal{G}^{(i)} - \bar{\mathcal{G}}\|_F^2$

جواب مسئله‌ی بالا، J_n بردار ویژه‌ی متناظر با بزرگ‌ترین مقادیر ویژه‌ی مسئله‌ی GEVD زیر است:

$$S_b^{-n} \mathbf{u}^{(n)} = \mu S_t^{-n} \mathbf{u}^{(n)} \quad (18-5)$$

با توجه به لم ۱.۳ می‌توانیم بگوییم اگر y بردار ویژه‌ی W متناظر با مقدار ویژه‌ی λ باشد (یعنی

$$\mu = \lambda, \text{ آن‌گاه } \mathbf{u}^{(n)} \text{ جواب مسئله‌ی ۱۸-۵ است و } \mathbf{H}_{(n)}^T \mathbf{u}^{(n)} = y \text{ و } (W\mathbf{y} = \lambda\mathbf{y})$$

بنابراین می‌توانیم به جای حل مسئله‌ی GEVD، ابتدا J_n بردار ویژه‌ی اول W را یافته و در ستون‌های ماتریس Y قرار دهیم. سپس دستگاه معادلات خطی $Y = H_{(n)}^{-T} U^{(n)}$ را برای $U^{(n)}$ حل کنیم.

در فصل ۳ دیدیم که فرم بردار ویژه‌های ماتریس W در SRDA به صورت «بدیهی» به دست می‌آید. اتفاق جالب این است که در HOSRD نیز می‌توان بردار ویژه‌های ماتریس W را به صورت بدیهی به دست آورد.

از آنجایی که W یک ماتریس قطری بلوکی است، بردار ویژه‌های آن را می‌توان از بردار ویژه‌های بلوک‌های قطر آن به دست آورد. بنابراین ابتدا فرم بردار ویژه‌های ماتریس $W^{(c)}$ را، برای یک c دلخواه، بررسی می‌کنیم.

فرض کنید $y^{(c)}$ بردار ویژه‌ی ماتریس $W^{(c)}$ متناظر با مقدار ویژه λ باشد (یعنی $(W^{(c)} y^{(c)}) = \lambda y^{(c)}$):

$$\mathbf{y}^{(c)} = \left[\mathbf{y}_1^{(c)T}, \dots, \mathbf{y}_{K_c}^{(c)T} \right]^T$$

آن‌گاه داریم:

$$W^{(c)} \mathbf{y}^{(c)} = \frac{1}{K_c} \begin{bmatrix} I_{P_n} & I_{P_n} & \cdots & I_{P_n} \\ I_{P_n} & I_{P_n} & \cdots & I_{P_n} \\ \cdots & \cdots & \ddots & \vdots \\ I_{P_n} & I_{P_n} & \cdots & I_{P_n} \end{bmatrix} \begin{bmatrix} \mathbf{y}_1^{(c)} \\ \mathbf{y}_2^{(c)} \\ \vdots \\ \mathbf{y}_{K_c}^{(c)} \end{bmatrix} = \frac{1}{K_c} \begin{bmatrix} \sum_{k=1}^{K_c} \mathbf{y}_k^{(c)} \\ \sum_{k=1}^{K_c} \mathbf{y}_k^{(c)} \\ \vdots \\ \sum_{k=1}^{K_c} \mathbf{y}_k^{(c)} \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{y}_1^{(c)} \\ \mathbf{y}_2^{(c)} \\ \vdots \\ \mathbf{y}_{K_c}^{(c)} \end{bmatrix} \quad (19-5)$$

پس برای هر $p = 1, \dots, K_c$ داریم:

$$\mathbf{y}_p^{(c)} = \frac{1}{\lambda K_c} \sum_{k=1}^{K_c} \mathbf{y}_k^{(c)}$$

بنابراین اگر برای هر $p = 1, \dots, K_c$ قرار دهیم $a \triangleq y_p^{(c)}$ از رابطه‌ی بالا داریم:

$$a \triangleq y_p^{(c)} = \frac{1}{\lambda K_c} K_c a = \frac{a}{\lambda}$$

پس $\lambda = 1$

برای هر c دلخواه، روشن است که رتبه‌ی ماتریس $W^{(c)}$ برابر P_n است. بنابراین $W^{(c)}$ دارای P_n مقدار ویژه‌ی یک است که بردار ویژه‌های آن‌ها به صورت $y^{(c)} = \underbrace{\left[a_{P_n}^T, \dots, a_{P_n}^T \right]}_{K_c}^T$ قابل توصیف هستند که در آن a_{P_n} یک بردار $1 \times P_n$ دلخواه است.

اکنون با استفاده از بردارویژه‌های $W^{(c)}$ ها به دست آوردن بردارویژه‌های W با استفاده از دو نکته زیر از جبرخطی آسان است:

۱. می‌دانیم رتبه‌ی یک ماتریس قطری بلوکی برابر مجموع رتبه‌های بلوک‌های روی قطر آن است. بنابراین رتبه‌ی ماتریس W برابر CP_n است.

۲. مقادیرویژه‌ی ماتریس قطری بلوکی اجتماع مقادیرویژه‌ی بلوک‌های روی قطر است و بردارهای ویژه از بردارویژه‌های بلوک‌های قطر به دست خواهد آمد. از آن‌جا که تمامی مقادیر ویژه‌ی بلوک‌های قطر W برابر یک است، می‌توانیم به ازای هر دسته بردار ویژه‌ی $\{y^{(1)}, y^{(2)}, \dots, y^{(C)}\}$ برای بلوک‌های قطر W یک بردار ویژه‌ی ماتریس W را به صورت

یک بردار ویژه‌ی ماتریس W را به صورت $y = [y^{(1)^T}, \dots, y^{(C)^T}]^T$ تعریف کنیم:

$$Wy = \begin{bmatrix} W^{(1)} & \cdot & \dots & \cdot \\ \cdot & W^{(2)} & \dots & \cdot \\ \dots & \dots & \ddots & \vdots \\ \cdot & \cdot & \dots & W^{(C)} \end{bmatrix} \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(C)} \end{bmatrix} = \begin{bmatrix} W^{(1)}y^{(1)} \\ W^{(2)}y^{(2)} \\ \vdots \\ W^{(C)}y^{(C)} \end{bmatrix} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(C)} \end{bmatrix} = y$$

پس دیدیم که W دارای CP_n بردار ویژه‌ای است که می‌توان آن‌ها را به راحتی با استفاده از بردارهایی که به صورت تصادفی تولید می‌کنیم، به دست آوریم و سپس با حل یک دستگاه معادلات خطی ماتریس‌های پایه را به دست آوریم.

الگوریتم ۶ شبکه‌گذاری الگوریتم پیشنهادی برای HOSRDA را نشان می‌دهد. در این شبکه‌گذاری تابع *Gram – Schmidt* فرآیند عمودسازی *Gram – Schmidt* را بر روی ستون‌های ماتریس ورودی اش اعمال می‌کند. دو تابع *rand* و *repmat* همانند توابع همنامشان در MATLAB عمل می‌کنند: $rand(P_n, J_n)$ و $repmat(A, 1, m) = \begin{bmatrix} A_1, A_2, \dots, A_m \end{bmatrix}$.

$$\begin{aligned} \hat{\mathcal{X}} &= cat \left(N + 1, \hat{\mathcal{X}}^{(1)}, \dots, \hat{\mathcal{X}}^{(C)} \right) \\ \hat{\mathcal{X}}^{(c)} &= cat \left(N + 1, \hat{\mathcal{X}}^{(c)}_{(1)}, \dots, \hat{\mathcal{X}}^{(c)}_{(K_c)} \right) \\ \hat{\mathcal{X}}^{(c)}_{(i)} &= \mathcal{X}^{(i)} - \bar{\mathcal{X}}, \quad i \in \Omega_c \end{aligned} \tag{۳۰-۵}$$

لازم به ذکر است که مقداردهی اولیه‌ی ماتریس‌های پایه در الگوریتم پیشنهادی به وسیله‌ی الگوریتم

HOSVD که در فصل ۳ توضیح داده شد، انجام می‌شود. این مقداردهی اولیه برای ماتریس‌های عامل الگوریتم‌های تانسوری مرسوم است [۲۰، ۳].

الگوریتم ۶ Higher Order Spectral Regression Discriminant Analysis (HOSRDA)

Input: Set of K training data $\{\mathcal{X}_k \in \mathbb{R}^{I_1 \times \dots \times I_N}, k = 1, \dots, K\}$ along with their labels $\{c_k, k = 1, \dots, K\}$ and J_1, \dots, J_N

Output: $\{U^{(n)} \in \mathbb{R}^{I_n \times J_n}, n = 1, \dots, N\}$ are the N orthogonal basis factors of HOSRDA subspace

```

1: Initialize  $\mathbf{U}^{(n)}, n = 1, \dots, N$ 
2:  $\bar{\mathcal{X}} = \frac{1}{K} \sum_{i=1}^K \mathcal{X}_i$ 
3: Calculate  $\hat{\mathcal{X}}$  according to equations (20-5)
4: repeat
5:   for  $n = 1, \dots, N$  do
6:      $\mathcal{H}^{-n} = \hat{\mathcal{X}} \times_{-(n, N+1)} \{\mathbf{U}^T\}$ 
7:      $P_n = \prod_{\substack{m=1 \\ m \neq n}}^N J_m$ 
8:     for  $c = 1, \dots, C$  do
9:        $\mathbf{Y}^{(c)} = \text{repmat}(\text{rand}(P_n, J_n), K_c, 1)$ 
10:      end for
11:       $\mathbf{Y} = [\mathbf{Y}^{(1)}; \dots; \mathbf{Y}^{(C)}]$ 
12:      Solve  $\mathbf{H}_{(n)}^{-T} \mathbf{U}^{(n)} = \mathbf{Y}$  for  $\mathbf{U}^{(n)}$ 
13:       $\mathbf{U}^{(n)} = \text{Gram-Schmidt}(\mathbf{U}^{(n)})$ 
14:    end for
15:  until a stop criterion is met

```

شبکه‌کد الگوریتم کلی آموزش-آزمون HOSRDA برای حالت دو کلاسه در الگوریتم ۷ آورده شده است. این الگوریتم برداری حاوی احتمال تعلق هر داده‌ی آزمون به کلاس هدف را در خروجی می‌دهد. در این شبکه‌کد تابع $\text{tenmat}(\mathcal{G}, n)$ مشابه دستور همنامش در جعبه‌ابزار تانسوری [۲۸] عمل می‌کند: فرم ماتریسی شده‌ی مُد- n -تansor \mathcal{G} را در خروجی‌اش قرار می‌دهد.

الگوریتم ۷ فرآیند آموزش-آزمون با استفاده از HOSRDA

Input: Set of K training data $\{\mathcal{X}_k \in \mathbb{R}^{I_1 \times \dots \times I_N}, k = 1, \dots, K\}$ along with their labels $\{c_k, k = 1, \dots, K\}$ and J_1, \dots, J_N . Also, the set of M test data $\{\mathcal{Y}_m \in \mathbb{R}^{I_1 \times \dots \times I_N}, m = 1, \dots, M\}$ suppose that the training tensor data are concatenated in the $(N+1)$ -th mode of \mathcal{X} , \mathbf{L} is the vector of train data labels and $\mathcal{Y} = \text{cat}(N+1, \mathcal{Y}_1, \dots, \mathcal{Y}_M)$.

Output: s : scores of single trials of test data.

```

1: Get HOSRDA subspace basis factors  $(\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)})$  via HOSRDA algorithm (algorithm 6)
2:  $\mathcal{G}_{train} = \mathcal{X} \times_1 \mathbf{U}^{(1)} \times_2 \dots \times_N \mathbf{U}^{(N)}$ 
3:  $\mathcal{G}_{test} = \mathcal{Y} \times_1 \mathbf{U}^{(1)} \times_2 \dots \times_N \mathbf{U}^{(N)}$ 
4:  $\mathbf{f}_{train} = \text{tenmat}(\mathcal{G}_{train}, 3)$ 
5:  $\mathbf{f}_{test} = \text{tenmat}(\mathcal{G}_{test}, 3)$ 
6:  $\mathbf{w} = \text{LDA}(\mathbf{f}_{train}, \mathbf{L})$ 
7:  $s = \mathbf{w}^T \mathbf{f}_{test}$ 

```

جدول ۱-۵: عملیات محاسباتی HOSRDA

ردیف	عملیات محاسباتی
۱	$\bar{\mathcal{X}}$
۲	$\hat{\mathcal{X}}$
۳	$\mathcal{H}^{-n} = \hat{\mathcal{X}} \times_{-(n, N+1)} \{\mathbf{U}^T\}$
۴	$\mathbf{U}^{(n)} = \mathbf{H}_{(n)}^{-nT} \setminus \mathbf{Y}$
۵	$\mathbf{U}^{(n)} = \text{Gram-Schmidt}(\mathbf{U}^{(n)})$

۳-۵ پیچیدگی محاسباتی

برای آن که بتوانیم پیچیدگی محاسباتی روش پیشنهادی، HOSRDA، را محاسبه کنیم فرض می‌کنیم $N = 2$. یعنی داده‌ها به صورت ماتریسی هستند و داده‌های آموزش در بعد سوم تansور \mathcal{X} چجده شده‌اند. برای داده‌های EEG چند کاناله این فرض معقول و متداول است؛ با این فرض تانسور داده‌های آموزش به صورت $Channel \times Time \times Trial$ خواهد بود.

برای سنجش پیچیدگی محاسباتی از معیاری به نام فلاپ^۱ استفاده می‌کنیم. یک فلاپ معادل است با یکی از اعمال جمع، تفریق، ضرب، تقسیم و یا ریشه‌ی دوم. فلاپ معیاری ساده برای مقایسه‌ی پیچیدگی محاسباتی الگوریتم‌ها به دست می‌دهد. در این بخش از مراجع [۴۵] و [۴۶] بهره برده شده است.

در ادامه‌ی این زیر بخش خواهیم دید که HOSRDA در تکرار آم الگوریتم از $O(KI^3)$ فلاپ نیاز

$$\text{دارد که در آن } I = \max(I_1, I_2)$$

لازم به ذکر است که در هر تکرار از الگوریتم، یک حلقه‌ی for دوتایی برای به روزرسانی ماتریس‌های عامل وجود دارد. در تکرار n از این for از $O(KI_n I_m + KI_n^3)$ فلاپ نیاز است). در ادامه محاسبات تعداد فلاپ را برای تکرار n از این حلقه‌ی for آورده‌ایم. فرض کنیم $n = m - 3$. ابتدا به صورت گذرا در جدول ۱-۵ بررسی می‌کنیم که HOSRDA چه مراحل محاسباتی دارد.

محاسبه‌ی دو سطر اول: محاسبه‌ی دو سطر اول جدول ۱-۵ به تعداد $(K - 1)I_n I_m + KI_n I_m$

فلاپ نیاز دارد.

ضرب مُدِّ: می‌توان دید که فرم ماتریسی شده‌ی تانسور محاسبه‌شده در سطر سوم جدول ۱-۵ برابر با

$$U^{(m)^T} \hat{X}_{(m)} \text{ است. این ضرب به تعداد } (I_m - 1) I_n K J_m \text{ فلاپ نیاز دارد.}$$

^۱Flop

حل دستگاه معادلات: برای حل دستگاه معادلات از از دستور `mldevide` در *MATLAB* استفاده شده است. این دستور معادله‌ی $A_{m \times n}x_{n \times 1} = b_{m \times 1}$ را برای بردار x حل می‌کند و در حالتی که A یک ماتریس مستطیلی باشد از تجزیه‌ی QR استفاده می‌کند. در این روش ماتریس $A_{m \times n}$ به صورت $A = QR$ تجزیه می‌شود که در آن $Q_{m \times n}$ یک ماتریس متعامد و $R_{n \times n}$ یک ماتریس بالامثلثی است. برای حل دستگاه، معادله با $Rx = Q^T b$ جایگزین می‌شود. بنابراین تعداد فlap مورد نیاز برابر است با فlap‌های مورد نیاز برای تجزیه‌ی QR، ضرب $Q^T b$ و حل دستگاه $Rx = c$ ؛ که به ترتیب برابر $2mn^3$ ، $(2m - 1)2mn^2$ و n^3 است. پس در مجموع $2mn^3 + (2m - 1)2mn^2 + n(2m - 1)$ فlap لازم است. با فرض $1 \gg m$ به تعداد $(1 + 2mn)(n + 1)$ ساده می‌شود. اگر $1 \gg n$ ، تعداد فlap‌ها را می‌توان به $2mn^3$ ساده‌تر کرد.

در \mathcal{HOSRDA} برای دستگاه معادلات $m = J_m K I_n$ و $n = I_n$ پس حل معادله نیاز به $(1 + 2J_m K I_n)$ فlap دارد.

فرآیند عمودسازی Gram-Schmidt: برای یک ماتریس $A_{m \times n}$ فرآیند عمودسازی Gram-Schmidt نیاز به $2mn^2$ فlap دارد. پس در \mathcal{HOSRDA} برای ماتریس $U^{(n)}$ ، که یک ماتریس $J_n \times I_n$ است، فرآیند عمودسازی Gram-Schmidt نیاز به $2I_n J_n^2$ فlap دارد.

۴-۵ جمع‌بندی

در این فصل \mathcal{HOSRDA} ، به عنوان یک روش جدید برای تصویر کردن داده‌های تانسوری به فضایی که تفکیک‌پذیری کلاس‌ها بیشتر باشد، معرفی شد. این روش تعمیم چندخطی از روش SRDA است و از بازنویسی روابط HODA به دست می‌آید. دیدیم که HODA مسئله‌ی تجزیه‌ی HOSRDA مقدارویژه-بردارویژه HODA را به حل یک دستگاه معادلات خطی تبدیل می‌کند. برتری \mathcal{HOSRDA} نسبت به HODA را می‌توان در سه مورد زیر خلاصه کرد:

۱. جایگزینی تجزیه‌ی مقدارویژه-بردارویژه با دستگاه معادلات خطی امکان استفاده از رگولاریزاسیون ماتریس‌های پایه‌ی زیرفضایی به دست آمده را فراهم می‌سازد. در حالی که اعمال قیود رگولاریزاسیون بر تجزیه‌ی مقدارویژه-بردارویژه ممکن نیست. به همین دلیل \mathcal{HOSRDA} می‌تواند یک ساختار جدید برای همراه کردن قیود رگولاریزاسیون تلقی شود.
۲. در تکرارهای ابتدایی HODA امکان معکوس‌نایپذیری ماتریس پراکندگی وجود دارد. این موضوع در

تجزیه‌ی مقدارویژه-بردارویژه دغدغه‌برانگیز است. با حذف تجزیه‌ی مقدارویژه-بردارویژه در HOSRDA این دغدغه مرتفع می‌شود.

۳. در شرایطی که بُعد دادگان بسیار زیاد است، پیچیدگی HODA بیشتر از HOSRDA خواهد بود؛ چرا که HOSRDA می‌تواند از الگوریتم‌های تکراری با پیچیدگی کم برای حل دستگاه معادلات خطی اش بهره‌مند شود.

در فصل بعد به ارزیابی HOSRDA در طبقه‌بندی در P300 speller می‌پردازیم.

فصل ۶

نتایج شبیه‌سازی HOSRDA

۱-۶ پیش‌گفتار

در این فصل به بررسی نتایج شبیه‌سازی الگوریتم پیشنهادی می‌پردازیم. در بخش دوم و سوم داده‌های مورد استفاده و معیارهای ارزیابی را معرفی می‌کنیم. نتایج تشخیص کاراکتر را برای تجزیه‌ی Tucker بدون سرپرست در بخش چهارم بررسی شده است. بخش پنجم و ششم به ترتیب به بررسی انتخاب پارامتر و پایداری HOSRDA و کارایی HOSRDA در طبقه‌بندی P300 speller اختصاص دارد. ماتریس پایه‌ی متناظر با بعد کانال در زیرفضای HOSRDA را در بخش هفتم بررسی می‌کنیم.

۲-۶ داده‌ها

در این بخش ابتدا مجموعه دادگان مورد استفاده را معرفی کرده و پس از آن پیش‌پردازش اعمال شده به آن را توضیح خواهیم داد.

۲-۶-۱ مجموعه دادگان استفاده شده

مجموعه داده‌ی^۱ دوم از از پایگاه داده‌ی مسابقه‌ی سوم^۲ BCI [۲۹] مورد استفاده قرار گرفته است که داده‌های EEG ثبت شده از یک P300 speller است. پارادایم P300 speller را در بخش ۲-۴ توضیح دادیم. پروتکل P300 speller مجموعه‌دادگان فوق مشابه با پروتکل Donchin و همکارانش است که در [۴۷] و [۴۸] معرفی شده است. در این پروتکل صفحه‌ی شطرنجی به مدت ۲/۵ ثانیه به سوزه نمایش داده

^۱Dataset

^۲Berlin BCI Competition III-Dataset II

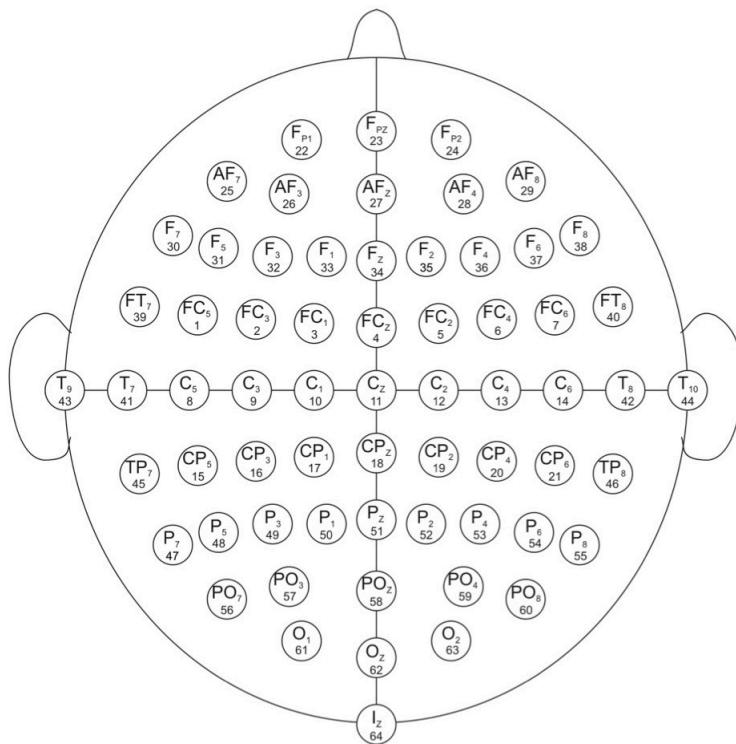
SEND						
S						
A	B	C	D	E	F	
G	H	I	J	K	L	
M	N	O	P	Q	R	
S	T	U	V	W	X	
Y	Z	1	2	3	4	
5	6	7	8	9	-	

شکل ۶-۱: صفحه‌ی P300 speller در پروتکل Donchin. هر بار یک کاراکتر به سوژه دیکته می‌شود (در شکل S باید هجی شود) [۱۲].

می‌شود (در این مدت زمان همه‌ی کاراکترها دارای شدت یکسان هستند یا اصطلاحاً صفحه خاموش است). سپس یک حرف به کاربر دیکته می‌شود و سطراها و ستون‌های ماتریس به صورت تصادفی و پشت سر هم با نرخ ۵/۷ هرتز روشن می‌شوند (شکل ۶-۱ نمایشی از صفحه‌ی P300 speller است زمانی که یک سطر آن روشن شده است). هر سطر (یا ستون) به مدت ۱۰۰ میلی‌ثانیه روشن می‌ماند و پس از آن کل صفحه ۷۵ میلی‌ثانیه خاموش می‌ماند. سطراها و ستون‌ها در بلوک‌های ۱۲ تایی روشن می‌شوند و در مجموع ۱۵ بلوک ۱۲ تایی تحریک خواهیم داشت (به عبارتی ۱۲ سطر و ستون به صورت تصادفی پشت سر هم روشن شده و این بلوک‌ها ۱۵ بار تکرار می‌شود). بنابراین به ازای هر کاراکتر ۱۸۰ تکثیت وجود دارد (۱۲ سطر و ستون، هر کدام ۱۵ بار روشن می‌شوند) و از هر ۱۸۰ تحریک مربوط به یک کاراکتر $30 = 15 \times 2$ تحریک مربوط به کلاس P300 هستند.

در این مجموعه داده سیگنال EEG با نرخ ۶۴ کانال ثبت و فیلترمیان‌گذار در باند ۰/۱ و ۶ هرتز اعمال شده است. محل قرارگیری الکترودها بر اساس سیستم ۱۰-۱۰ است که در شکل ۶-۲ نشان داده شده است.

در این مجموعه داده‌ها برچسب تحریک‌های چندین کاراکتر به عنوان داده‌های آموزش در دسترس قرار گرفته است. هدف تعیین برچسب تکثیت‌های آزمون و در نهایت تشخیص کاراکترهاست. این داده‌ها در طی پنج جلسه از دو سوژه (سوژه‌ی A و B) ثبت شده‌اند و ۸۵ کاراکتر آموزش و ۱۰۰ کاراکتر آزمون برای هر سوژه داده شده است. کاراکترهای داده‌های آزمون همزمان با اعلام نتایج مسابقه در سایت آن



شکل ۶-۲: محل قرارگیری الکترودها در سیستم ۱۰-۱۰ [۱۲].

منتشر شده است.^۳

۲-۲-۶ پیش‌پردازش

برای پیش‌پردازش داده‌ها مشابه با گروه برنده‌ی مسابقه [۱۳] مراحل زیر بر روی داده‌ها انجام شده

است:

۱. تکثیت‌ها جداسازی شده‌اند. هر تکثیت شامل نمونه‌های ۶۶۷ میلی‌ثانیه بعد از تحریک است.

بنابراین با توجه به فرکانس نمونه‌برداری، هر تکثیت دارای ۱۶۰ نمونه خواهد بود.

۲. تکثیت‌های جدا شده در مرحله‌ی اول در باند ۱/۰ تا ۱۰ هرتز فیلتر شده‌اند. از فیلتر Chebyshev

نوع یک مرتبه ۸ استفاده شده است.

۳. نرخ نمونه‌برداری به فرکانس ۲۰ هرتز کاهش داده می‌شود. برای این منظور از تابع decimate در

استفاده شده است. در این مرحله هر تکثیت دارای ۱۴ نمونه خواهد بود.

۴. داده‌های هر تکثیت در یک ماتریس 14×64 چیده می‌شوند که ابعاد آن $Channel \times Time$ است.

^۳ <http://bbc1.de/competition/iii/results/index.html>

اکنون لازم است داده‌ها تانسوری شوند. کافی است ماتریس‌های متناظر با تکثیت‌ها را در بعد سوم پشت هم قرار دهیم. در این صورت اگر K تکثیت داشته باشیم، تانسور ما $K \times 14 \times 14 \times 64$ خواهد بود. برای داده‌های آموزش $K = 100 \times 180 = 18000$ و برای داده‌های آزمون $K = 100 \times 180 = 18000$ (به ازای هر کاراکتر 180 تکثیت داریم).

۳-۶ معیارهای ارزیابی

کارایی الگوریتم پیشنهادی را می‌توان از سه منظر مورد ارزیابی قرار داد که در این قسمت به آن‌ها پرداخته می‌شود.

۱-۳-۶ آزمون آماری

از آنجا که الگوریتم HOSRDA در واقع یک الگوریتم کاهش ویژگی است؛ بنابراین لازم است که توانایی ویژگی‌های نهایی در تفکیک دو کلاس بررسی شود. یکی از راههایی که معیاری کمی در این زمینه به دست می‌دهد، آزمون آماری است. به این صورت که با به کارگیری یک آزمون مناسب، فرض صفر را تشابه دو کلاس قرار داده و با توجه به p -value به دست آمده، آن را قبول یا رد کنیم. در صورتی که فرض صفر رد شود ویژگی مربوطه در تفکیک دو کلاس موفق است. در بخش ۶-۶ از این منظر به بررسی تمایزپذیری دو کلاس می‌پردازیم.

۲-۳-۶ تشخیص کاراکتر

مهمترین هدف در پردازش داده‌های یک P300 Speller تشخیص کاراکترهای داده‌های آزمون است. برای این منظور لازم است بر اساس احتمال حضور P300 در تکرارهای مربوط به روشن شدن هر سطر (ستون)، تعیین کنیم کدامین سطر (ستون) حاوی کاراکتر مورد نظر کاربر است.

برای تشخیص این‌که هر سطر (ستون) با چه احتمالی حاوی کاراکتر مورد نظر کاربر است، مجموع احتمال حضور P300 در همه‌ی تکثیت‌های متناظر با روشن شدن آن سطر (ستون) را به عنوان معیار تصمیم‌گیری قرار می‌دهیم (بسته به این‌که تعداد تکرارهای هر تحریک چقدر باشد تعداد این تکثیت‌ها می‌تواند متفاوت باشد). سطر (ستون)ی که این معیار برای آن بیشینه باشد به عنوان سطر (ستون) حاوی کاراکتر انتخاب و تلاقی سطر و ستون انتخاب شده به عنوان کاراکتر تشخیص داده می‌شود. شبه‌گُد ۸

الگوریتم تشخیص سطر و ستون حاوی کاراکتر را نشان می‌دهد. در این شبکه‌گذار \max و zeros مشابه با تابع همنامشان در MATLAB عمل می‌کنند. (۱) یک بردار $1 \times m$ تولید می‌کند که همهٔ درایه‌های آن صفر است و $(v)_{\sim, ind} = \max(v)$ را که مقدار بیشینه دارد در ind بر می‌گرداند. در بخش ۶-۶-۲ از این منظر به بررسی عملکرد الگوریتم پیشنهادی خواهیم پرداخت.

الگوریتم ۸ تشخیص کاراکتر

Input: s : the vector of scores of single trials and c : the vector of codes of each single trial.
%suppose that the length of s and c is M .

Output: row and col : the numbers of the row and column that include the desired character

```

1:  $v = \text{zeros}(12, 1)$ 
2: for  $k = 1, \dots, M$  do
3:    $v(c(k)) = v(c(k)) + s(k)$ 
4: end for
5:  $v_{row} = v(1 : 6)$ 
6:  $v_{col} = v(7 : 12)$ 
7:  $[\sim, row] = \max(v_{row})$ 
8:  $[\sim, col] = \max(v_{col})$ 
```

۴-۶ تجزیهٔ Tucker بدون سرپرست

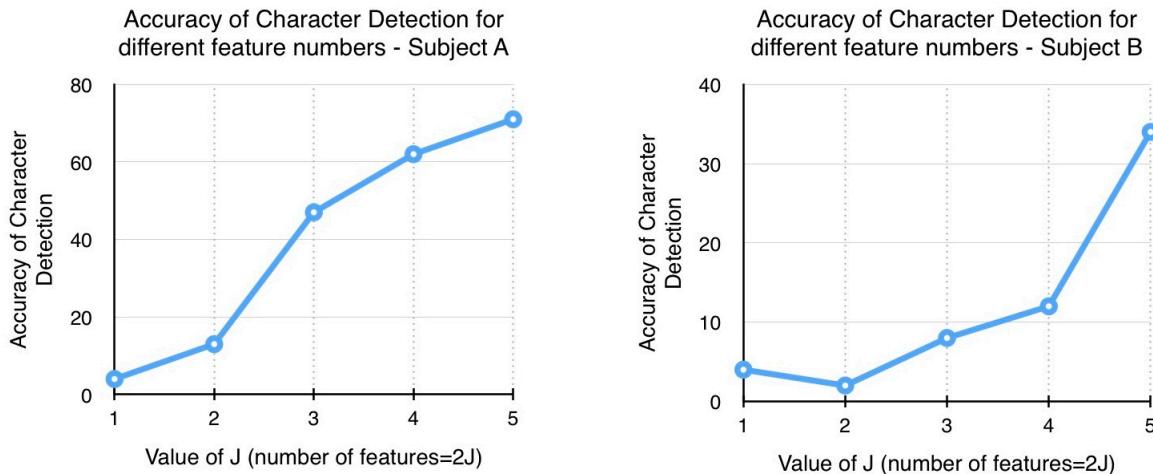
برای آن که لزوم استفاده از تجزیهٔ تانسوری با سرپرست را نشان دهیم، در یک آزمایش ساده از تجزیهٔ Tucker بدون هیچ قیدی برای استخراج ویژگی از داده‌های تانسوری EEG استفاده کردیم (در بخش ۳-۲-۱ کاربرد آن را در ادبیات دیدیم). در واقع اگر \mathcal{A} داده‌های $Channel \times Time \times Trial$ باشد، با استفاده از الگوریتم HOOI تجزیهٔ Tucker با رتبه‌ی (J, J) به آن اعمال می‌کنیم و تانسور هسته ویژگی‌ها را به ما خواهد داد. پس از آن با برداری کردن ویژگی هر تکثیت آن‌ها را به طبقه‌بند LDA می‌دهیم.

شکل ۳-۶ دقت تشخیص کاراکتر را به ازای J از یک تا پنج نشان می‌دهد. روشن است که عملکرد تجزیهٔ بدون سرپرست قابل قبول نیست.

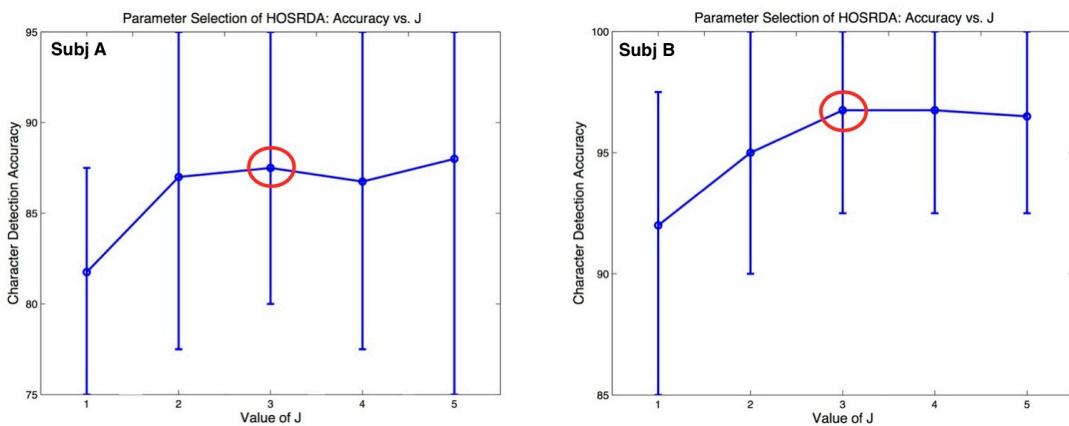
۶-۵ انتخاب پارامتر و همگرایی HOSRDA

۶-۵-۱ انتخاب تعداد ویژگی‌ها

انتخاب تعداد ویژگی‌ها در الگوریتم‌های کاهش ویژگی دارای اهمیت خاصی است و تاثیر به‌سزایی بر نتایج خواهد داشت. از طرفی رتبه‌ی تجزیهٔ Tucker، بر نتیجه‌ی آن تاثیر قابل توجهی دارد و می‌دانیم



شکل ۶-۳: عملکرد تجزیه‌ی Tucker بدون سرپرست در تشخیص کاراکتر: نمودار دقت تشخیص کاراکتر بر حسب رتبه‌ی تجزیه برای دو سوژه‌ی دادگان (سمت چپ سوژه‌ی A، سمت راست سوژه‌ی B). داده‌ها با استفاده از الگوریتم HOOI تجزیه‌ی Tucker با رتبه‌ی (J, J) به آن اعمال شده است. روشن است که عملکرد تجزیه‌ی بدون سرپرست قابل قبول نیست.



شکل ۶-۴: انتخاب پارامتر در HOSRDA: نمودار «دقت طبقه‌بندی کاراکتر بر حسب J» برای دو سوژه‌ی A و B. این نمودار با استفاده از روش اعتبارسنجی متقابل Hold Out به دست آمده است.

در الگوریتم‌های کاهش ویژگی تانسوری مبتنی بر تجزیه‌ی Tucker رتبه‌ی تجزیه در واقع بُعد فضای ویژگی است و تعداد ویژگی‌های کاهش بُعد یافته را به دست می‌دهد. در [۱۰] برای تجزیه‌ی CP به این موضوع پرداخته شده است و در [۱۶] نیز اهمیت این موضوع برای STDA خاطرنشان شده است.

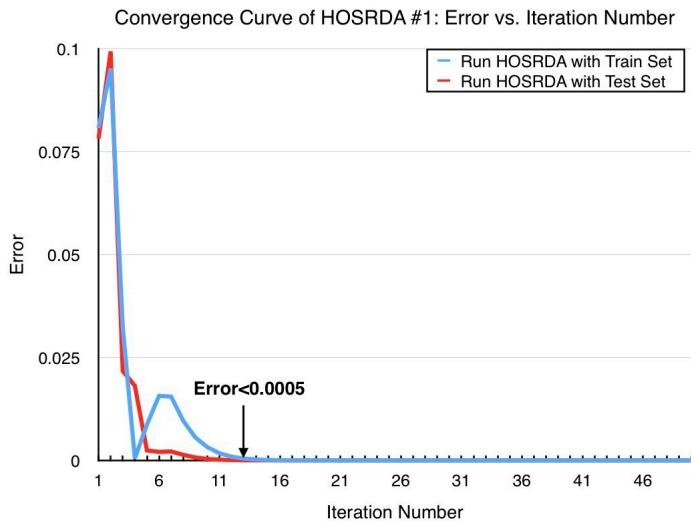
ما علاقه‌مندیم که بُعد کمینه را برای فضای ویژگی بیابیم و فرض کرده‌ایم $J_2 \triangleq J_1$ و کوچک‌ترین J را بین ۱ تا ۵ انتخاب می‌کنیم. برای این منظور از روش اعتبارسنجی متقابل Hold Out [۴۹] استفاده کرده‌ایم و داده‌های آموزش را به دو قسمت تقسیم کرده: یک J خاص را در نظر بگیرید. ۴۵ کاراکتر را به صورت تصادفی برای آموزش زیرفضای HOSRDA (به داده‌های این ۴۵ کاراکتر «داده‌های آموزش اعتبارسنجی» خواهیم گفت) و ۴۰ کاراکتر دیگر (داده‌های آزمون اعتبارسنجی) برای آزمون دقت طبقه‌بندی کاراکترها با استفاده از داده‌های تصویر شده بر زیرفضای HOSRDA، در نظر می‌گیریم. زیرفضای HOSRDA را با استفاده از داده‌های آموزش اعتبارسنجی به دست آورده و سپس داده‌های آزمون اعتبارسنجی را بر آن تصویر کرده و دقت طبقه‌بندی کاراکتر را برای آن‌ها محاسبه می‌کنیم. این عمل را (انتخاب داده‌های آموزش و آزمون اعتبارسنجی و تشخیص کاراکتر) به ازای هر J ، ده بار تکرار می‌کنیم. بدین ترتیب برای هر $J = 1, \dots, 5$ یک میانگین و یک واریانس دقت تشخیص کاراکتر خواهیم داشت، که این یک نمودار «دقت طبقه‌بندی کاراکتر بر حسب J » تولید می‌کند. با توجه به این که مقدار کمینه‌ی J مورد علاقه‌ی ماست، در میان نقاطی که بیشترین میانگین دقت را دارند، کوچک‌ترین نقطه با واریانس کمتر انتخاب می‌شود. شکل ۶-۴ نمودار فوق و J ‌های انتخاب شده را برای دو سوژه‌ی مجموعه دادگان نشان می‌دهد (مقدار J برای هر دو سوژه‌ی A و B برابر ۳ انتخاب شده است).

۲-۵-۶ همگرایی و پایداری الگوریتم

همه‌ی الگوریتم‌های تکراری نیازمند یک شرط توقف پایدار هستند که همگرایی آن‌ها را تضمین کند. همان‌طور که در بخش ۳-۵ معرفی شد، STDA نرم تغییرات ماتریس‌های پایه را به عنوان خطای نظر گرفته و در صورتی که این خطای حد آستانه‌ای کوچک‌تر شود، الگوریتم متوقف می‌شود. از آن‌جا که در HOSRDA هدف یافتن زیرفضایی است که در آن تفکیک‌پذیری دو کلاس (نسبت فیشر) بیشینه باشد، شرط توقف آن را به صورت زیر تعریف می‌کنیم:

$$\text{Error} = |\text{FisherRatio}(i) - \text{FisherRatio}(i-1)| < \varepsilon \quad (1-6)$$

که در آن $\text{FisherRatio}(i)$ نسبت فیشر در تکرار i ام الگوریتم است که برای محاسبه‌ی آن داده‌های



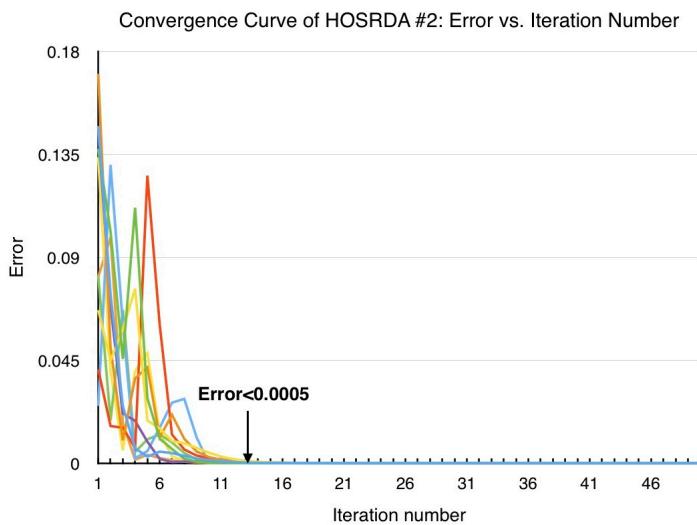
شکل ۶-۵: نمودار «خطا بر حسب شماره‌ی تکرار» برای بررسی پایداری الگوی همگرایی الگوریتم HOSRDA با شرط توقف پیشنهادی. آزمایش اول: الگوریتم HOSRDA را یک بار برای داده‌های آموزش و یک بار برای داده‌های آزمون اجرا می‌کنیم و نمودار «خطا بر حسب شماره‌ی تکرار» را رسم می‌کنیم. به روشنی می‌توان پایداری الگوی همگرایی را برای دو دسته داده مشاهده کرد.

آموزش را بر ماتریس‌های پایه‌ی زیرفضای به دست آمده در تکرار نام تصویر کرده و نسبت فیشر را برای ویژگی‌های به دست آمده محاسبه می‌کنیم. به عبارتی:

$$FisherRatio(i) = \frac{\sum_{c=1}^C K_c \|\bar{\mathcal{G}}^{(c)} - \bar{\bar{\mathcal{G}}}^*\|_F^2}{\sum_{k=1}^K \|\mathcal{G}^{(k)} - \bar{\mathcal{G}}^{(c_k)}\|_F^2} \quad (2-6)$$

که در آن $\mathcal{G}^{(k)}$ ، $\bar{\mathcal{G}}^{(c)}$ و $\bar{\bar{\mathcal{G}}}$ به ترتیب تصویر داده‌ی آموزش k ام، میانگین تصاویر داده‌های کلاس c ، میانگین تصاویر داده‌های هم کلاس k ام و میانگین تصاویر کل داده‌ها، بر زیرفضای تکرار نام هستند. اکنون لازم است همگرایی و پایداری شرط توقف معرفی شده بررسی شود. مُراد از پایداری این است که الگوی همگرایی شرط توقف به صورت کلی به ویژگی‌های خاص داده‌های استفاده شده وابسته نباشد. برای این منظور الگوریتم را با داده‌های مختلف اجرا کرده و نمودار «خطا بر حسب شماره‌ی تکرار» را رسم می‌کنیم و الگوی همگرایی را بررسی می‌کنیم. در این نمودارها لازم است الگوی همگرایی تقریباً مشابه باشد و همچنین بعد از تعداد محدودی تکرار همگرایی تضمین شود. بدین منظور بر روی داده‌های سوژه‌ی B دو آزمایش انجام دادیم:

- در آزمایش اول الگوریتم ۶ از فصل ۵ را یک بار برای داده‌های آموزش و یک بار برای داده‌های آزمون اجرا می‌کنیم. دقت شود که در اینجا تنها هدف از اجرای الگوریتم بررسی الگوی همگرایی است؛ به همین دلیل استفاده از داده‌های آزمون مانع ندارد و می‌توانیم آن‌ها را به عنوان یک دسته داده



شکل ۶-۶: نمودار «خطا بر حسب شماره‌ی تکرار» برای بررسی پایداری الگوی همگرایی الگوریتم HOSRDA با شرط توقف پیشنهادی. آزمایش دوم: ۱۰ زیرمجموعه‌ی ۴۵ کاراکتری از ۸۵ کاراکتر آموزش به صورت تصادفی انتخاب کرده‌ایم و برای هر زیرمجموعه الگوریتم HOSRDA را اجرا کرده‌ایم. باز هم می‌توان پایداری نسبی الگوی همگرایی را برای این ۱۰ مجموعه مشاهده نمود. در شکل هر رنگ مربوط به یک مجموعه داده است.

استفاده کنیم. شکل ۶-۵ نمایش نمودار «خطا بر حسب شماره‌ی تکرار» برای این دو دسته داده است. به روشنی می‌توان پایداری الگوی همگرایی را برای دو دسته داده مشاهده کرد.

۲. آزمایش دوم را با ایده از [۱۶] انجام داده‌ایم: ۱۰ زیرمجموعه‌ی ۴۵ کاراکتری از ۸۵ کاراکتر آموزش به صورت تصادفی انتخاب کرده‌ایم و برای هر زیرمجموعه الگوریتم HOSRDA را اجرا کرده و نمودار «خطا بر حسب شماره‌ی تکرار» را رسم می‌کنیم. شکل ۶-۶ نمودار همگرایی را برای این ۱۰ مجموعه نشان می‌دهد. باز هم می‌توان پایداری نسبی الگوی همگرایی را برای این ۱۰ مجموعه مشاهده نمود.

با توجه به نمودارهای آزمایش دوم مقدار آستانه‌ی خطای برای ادامه کار $\epsilon = 0.0005$ در نظر می‌گیریم. لازم به ذکر است که انتخاب ϵ یک بده-بستان^۴ میان سرعت همگرایی و دقت آن است. انتخاب ما بر اساس نمودار ۶-۶ به این صورت بوده است که کوچک‌ترین آستانه‌ای را انتخاب کنیم که برای هر ۱۰ مجموعه الگوریتم همگرا شده باشد.

^۴Trade-off

۶-۶ ارزیابی HOSRDA

در این بخش عمکرد الگوریتم پیشنهادی را از با استفاده از سه معیاری که در بخش ۳-۶ بحث شد،

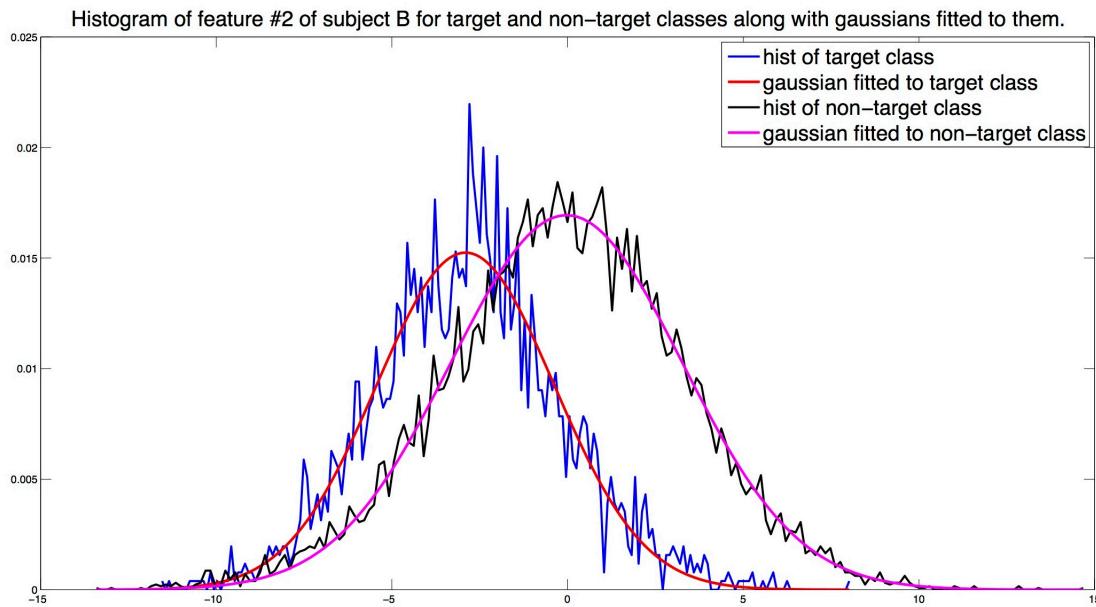
ارزیابی می‌کنیم.

۶-۶-۱ بررسی تمایزپذیری دو کلاس با استفاده از آزمون آماری

در این قسمت با استفاده از آزمون آماری به بررسی این موضوع می‌پردازیم که ویژگی‌هایی که از تصویر داده‌ها بر زیرفضای HOSRDA به دست می‌آینند چقدر در تمایز دو کلاس توانا هستند. در این جا برای جلوگیری از به درازا کشیدن مطلب نتایج را برای سوزه‌ی B ارائه می‌کنیم. برای این منظور از آزمون two sample t-test استفاده می‌کنیم و فرض صفر را تساوی میانگین دو کلاس قرار می‌دهیم. همان‌طور که در فصل ۳ توضیح داده شد در این آزمون فرض بر این است که دو کلاس دارای توزیع نرمال هستند. این فرض برای مسئله‌ی ما برقرار است؛ چرا که تصویر کردن داده‌های تانسوری بر زیرفضای HOSRDA شامل ضربهای مُد یک و دو است که در واقع ترکیب خطی داده‌های کanal-زمان است. با استناد به قضیه‌ی حد مرکزی، در صورتی که تعداد داده‌هایی که ترکیب خطی آن‌ها محاسبه می‌شود به اندازه‌ی کافی بزرگ باشد، توزیع ترکیب خطی آن‌ها نرمال است. با این حال یکی از ویژگی‌های داده‌های آموزش سوزه‌ی B همراه با توزیع نرمال برآش شده به آن در شکل ۷-۶ نمایش داده شده است.

اکنون لازم است تا بررسی کنیم که آزمون t-test را با فرض تساوی واریانس دو کلاس انجام دهیم یا عدم تساوی. هرچند با بررسی واریانس نمونه در دو کلاس برای ویژگی‌های مختلف می‌توان فهمید که واریانس دو کلاس متفاوت است، اما استفاده از آزمون two sample F-test برای این منظور به این فرض پشتونه‌ی بهتری می‌افزاید؛ در این آزمون فرض صفر برابری واریانس دو کلاس است. جدول ۶-۱ مقادیر p-value آزمون F-test را برای ۹ ویژگی سوزه‌ی B نشان می‌دهد که مقادیر بسیار کوچک رد قطعی فرض صفر را نشان می‌دهند و در نتیجه آزمون تفاوت واریانس دو کلاس را تأیید می‌کند.

اکنون یک آزمون یک آزمون two sample t-test با فرض عدم تساوی واریانس دو کلاس بر روی داده‌ها اعمال می‌کنیم. این آزمون برای ویژگی‌های سوزه‌ی B در جدول ۶-۱ آورده شده است که مقادیر بسیار کوچک آن‌ها رد فرض صفر را تایید می‌کنند و تمایز دو کلاس را نشان می‌دهند. با توجه به اطلاعات فوق می‌توانیم بگوییم ویژگی‌هایی که به وسیله‌ی HOSRDA استخراج می‌شوند تمایز معناداری بین دو کلاس



شکل ۶-۷: توزیع آماری ویژگی شماره دو از داده‌های آموزش سوزه‌ی B. شکل هیستوگرام دو کلاس همراه با منحنی نرمال برآورده شده را نمایش می‌دهد. همان‌طور که بر اساس قضیه‌ی حد مرکزی انتظار می‌رفت، مشاهده می‌شود که دو کلاس دارای توزیع نرمال هستند.

جدول ۱-۶: مقادیر p-value مربوط به F-test و t-test بر روی ویژگی‌های سوزه‌ی B. از p-value های آزمون t-test می‌توان دید که دو کلاس دارای واریانس‌های متفاوت هستند. همچنین مقادیر بسیار کوچک p-value برای نشان‌دهنده‌ی تمایز معنادار بین دو کلاس است.

t-test p-value	F-test p-value	شماره‌ی ویژگی
$2/483e-171$	$7/334e-297$	۱
$1/019e-33$	$1/366e-219$	۲
$8/473e-33$	$1/429e-84$	۳
$2/00e-323$	$3/233e-237$	۴
$1/622e-25$	$1/384e-76$	۵
$9/95e-18$	$4/407e-30$	۶
$8/493e-278$	$1/308e-92$	۷
$1/922e-66$	$2/000e-323$	۸
$0/009$	$5/459e-05$	۹

برقرار می‌کنند.

۶-۶-۲ کارایی HOSRDA در تشخیص کاراکتر

در بخش ۲-۴ روش‌هایی که در ادبیات برای تشخیص کاراکتر در دادگان مورد استفاده‌ی ما به کار برده شده است، توضیح داده شد. جدول ۶-۶ نتایج گزارش‌شده‌ی روش‌های فوق را همراه با دقت تشخیص کاراکتر روش پیشنهادی (HOSRDA) نشان می‌دهد. برای جلوگیری از تکرار مطالب از توضیح مجدد این

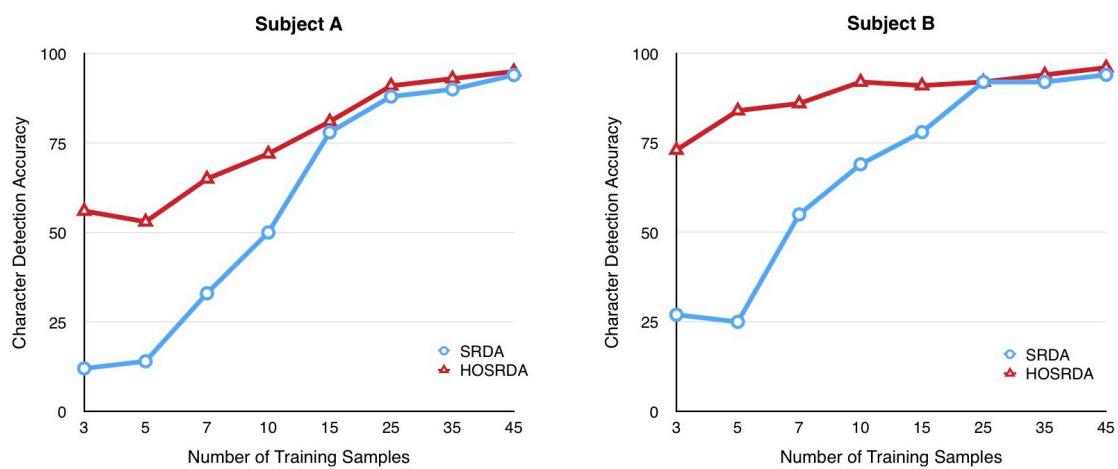
روش‌ها پرهیز می‌کنیم. در این جدول نتایج روش STDA [۱۶] نیز به عنوان روش state-of-the-art کاهش ویژگی تانسوری در تشخیص P300 آورده شده است. مقایسه‌ی HOSRDA و STDA از این جهت دارای اهمیت است که یکم، هر دو تانسوری هستند، دوم، در واقع HOSRDA از HODA مشتق شده است (و STDA حالت خاصی از HODA است). یک مسئله‌ی چالش برانگیز در P300 Speller این است که روش‌های مختلف در تعداد پایین تکرار روش شدن سطر و ستون‌ها چه عملکردی دارند، به همین دلیل در این جدول برای دقت تشخیص کاراکتر به ازای سه حالت، وقتی تعداد تکرار (R) هر تحریک در P300 Speller برابر با ۵، ۱۰ و ۱۵ در نظر گرفته شود، گزارش شده است. همان‌طور که قبلاً گفته شد، با توجه به دامنه‌ی بسیار اندک EEG در P300، هر تحریک به دفعات تکرار می‌شود تا از تجمعی نتایج طبقه‌بندی آن‌ها به دقت تشخیص کاراکتر بهتری دست یابیم.

روش پیشنهادی در تشخیص کاراکترها برای سوژه‌ی B از تمامی روش‌های دیگر برتری دارد و نقطه‌ی قوت مهم آن در تکرارهای ۵ و ۱۰ است. هرچند نتایج سوژه‌ی A را نمی‌توان بهترین نتایج خواند، اما نتایج آن از STDA بهتر بوده و با سایر روش‌ها قابل مقایسه است. به صورت میانگین بین دو سوژه نیز HOSRDA تقریباً بر تمامی روش‌ها برتری دارد و کاملاً با eSVM به عنوان روش پیشرو در تشخیص کاراکترهای P300 Speller قابل مقایسه است.

نکته‌ی دارای اهمیت این است که زمان مرحله‌ی آموزش eSVM چندین ساعت است (حدوداً ۴ ساعت به طور میانگین برای هر سوژه) در حالی که مرحله‌ی آموزش HOSRDA به طور میانگین برای هر سوژه تنها $2/5\text{s}$ زمان می‌برد. دلیل این امر را می‌توان در سادگی عملیات تانسوری جستجو کرد: هرچند در ابتدا ممکن است روش‌های تانسوری پیچیده به نظر برسند اما با دقت بیشتر در آن‌ها می‌توان دید که آن‌ها چیزی جز ضرب‌های ماتریسی و به هم پیوستن^۵ ماتریس‌ها و بردارها نیستند.

اشاره به سادگی پیش‌پردازش انجام شده در HOSRDA در مقایسه با EFLD و HODA+regression نیز خالی از لطف نیست. همان‌طور که دیدیم سالواریس و همکاران در [۳۲] از تبدیل زمان-فرکانس استفاده کرده‌اند. همچنین در [۴۴] برای ساخت تانسور از هر یک از تکثیت‌ها از تخمین چندجمله‌ای استفاده شده که با توجه به تعداد زیاد داده‌ها وقت گیر است. این در حالی است که ما از نمونه‌های زمانی (که تنها فیلتر میان‌گذر شده‌اند) استفاده کرده‌ایم.

⁵Concatenate

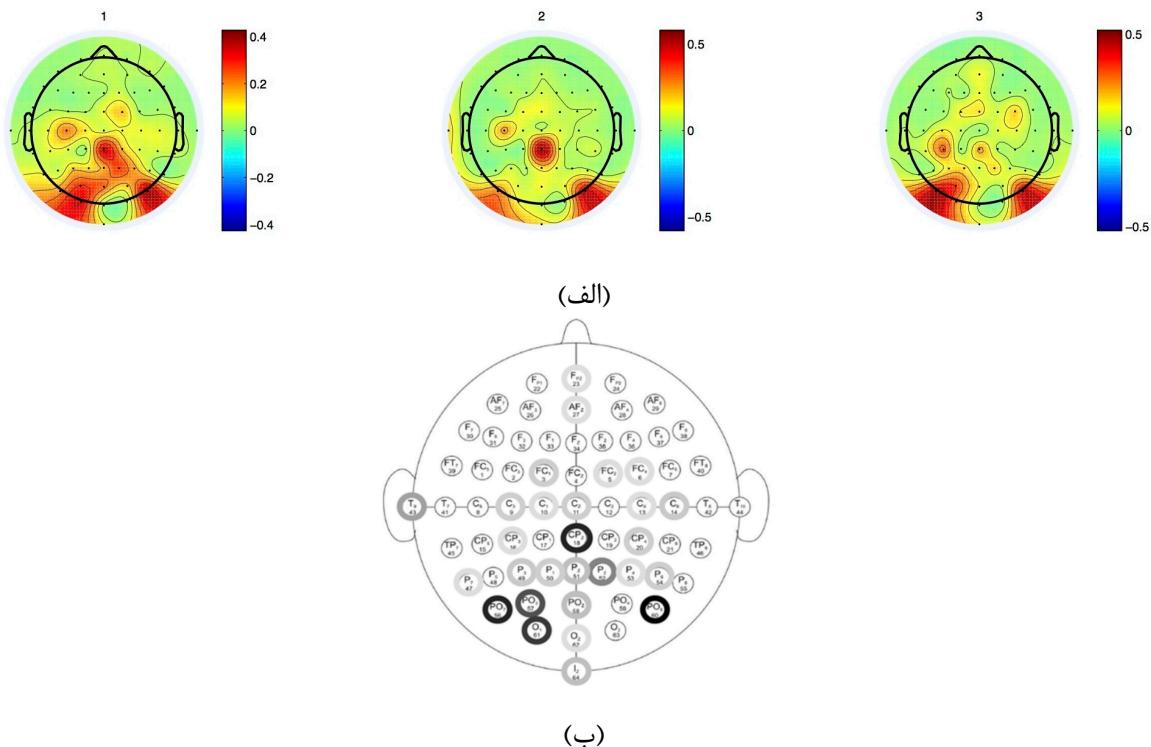


شکل ۶-۸: نمودار «دقت تشخیص کاراکتر بر حسب تعداد کاراکتر آموزش» برای دو سوژه‌ی A و B. برتری HOSRDA وقتی تعداد کاراکترهای آموزش کم است به روشنی دیده می‌شود.

اکنون با توجه به این‌که HOSRDA تعمیم تansوری SRDA است، مقایسه‌ی عملکرد این دو دید خوبی درباره‌ی تاثیر تansوری کردن روش‌های کاهش ویژگی می‌دهد. برای این مقایسه تعداد کاراکترهای آموزش را از ۳ کاراکتر تا ۴۵ کاراکتر تغییر داده و دقت تشخیص کاراکتر برای داده‌های آزمون را به دست می‌آوریم. نمودار «دقت تشخیص کاراکتر بر حسب تعداد کاراکتر آموزش» برای دو سوژه‌ی A و B در شکل ۶-۸ نمایش داده شده است. برتری HOSRDA وقتی تعداد کاراکترهای آموزش کم است به روشنی دیده می‌شود.

جدول ۶-۲: دقیق ت تشخیص کاراکتر برای روش‌های مختلف برای دو سوژه‌ی A و B زمانی که هر تحریک R بار تکرار شده است. خطوط فاصله نشانه‌ی این است که مرجع مربوطه نتایجی را در آن قسمت گزارش نکرده است.

	SubjectA			SubjectB			mean			Subject
	5	10	15	5	10	15	5	10	15	R
HOSRDA	63	84	96	82	94	97	72.5	89	96.5	
eSVM [13]	72	83	97	75	91	96	73.5	87	96.5	
CNN-1 [33]	61	86	97	79	91	92	70	88.5	94.5	
MCNN-1 [33]	61	82	97	77	92	94	69	87	95.5	
EFLD [32]	65	82	93	78	93	97	71.5	87.5	95	
STDA	56	82	95	80	93	95	68	87.5	95	
SWLDA [31]	-	-	-	-	-	-	-	-	92.5	
HODA+regression [44]	-	-	-	-	-	-	-	-	92	
Competition 2nd Rank [29]	-	-	-	-	-	-	-	-	90.5	



شکل ۶-۹: (الف) ستون‌های ماتریس عامل اول (متناظر با بُعد کanal) از زیرفضای HOSRDA داده‌های P300 Speller مربوط به سوژه‌ی B. (ب) نتایج انتخاب کanal در [۱۳]. کanal‌ها بر اساس ارزش آن‌ها با شدت رنگ مشخص شده‌اند.

۷-۶ ماتریس پایه‌ی متناظر با بُعد کanal در زیرفضای HOSRDA

همان‌طور که همواره گفته می‌شود، هر ماتریس عامل از تجزیه‌ی تansوری، ویژگی‌های مربوط به آن بُعد از تانسور را نشان می‌دهد. برای مثال اگر تانسور $Channel \times Time \times Trial$ را تجزیه می‌کنیم، ماتریس عامل اول و دوم به اصطلاح حاوی امضا زمانی و مکانی^۶ سیگنال تجزیه شونده هستند. به همین دلیل بررسی این ماتریس‌های عامل ازین جهت می‌تواند مفید باشد که در ک بهتری از زیرفضایی که داده‌ها بر آن تصویر می‌شوند به ما خواهد داد.

شکل ۶-۹الف ستون‌های ماتریس عامل اول (متناظر با بُعد کanal) از زیرفضای HOSRDA را برای سوژه‌ی B نشان می‌دهد. این توپوگرافی‌ها درواقع یک وزن‌دهی به کanal‌ها را نشان می‌دهند؛ کاری که روش‌های انتخاب کanal با صرف هزینه‌ی زمانی و محاسباتی انجام می‌دهند (مانند آن‌چه در [۱۳] آمده) و یا با درنظر گرفتن الکترودهای از پیش تعیین شده احتمال از دست رفتن اطلاعات را ایجاد می‌کنند. برای بررسی میزان اعتبار وزن‌دهی انجام شده، از نتایج انتخاب کanal در [۱۳] استفاده می‌کنیم. در شکل

^۶Time and space signatures

۶-۹ ب نتایج انتخاب کانال در مقاله‌ی فوق نشان داده شده که کانال‌ها بر اساس ارزش آن‌ها با شدت رنگ مشخص شده‌اند. مشاهده می‌شود که توپوگرافی‌های شکل ۶-۱۹ الف تطابق خوبی با کانال‌های بالرزش در شکل ۶-۹ ب دارند.

۸-۶ جمع‌بندی

در این فصل به ارزیابی الگوریتم پیشنهادی (HOSRDA) در تشخیص کاراکتر P300 speller در این فصل به ارزیابی الگوریتم پیشنهادی (HOSRDA) در تشخیص کاراکتر P300 speller پرداختیم. درباره‌ی پایداری و شیوه‌ی انتخاب پارامترهای آن صحبت کردیم و دقت تشخیص کاراکتر آن را با سایر روش‌های موجود بر روی مجموعه‌دادگان مورد استفاده بررسی کردیم. دیدیم که HOSRDA برای یکی از سوژه‌ها بیشتری دقت تشخیص کاراکتر را داشت و برای سوژه‌ی دیگر نتایج آن قابل رقابت با روش‌های دیگر بود. به طور میانگین برای دو سوژه، به ازای ۱۵، ۱۰ و ۵ تکرار برای هر تحریک، به ترتیب دقت تشخیص کاراکتر ۹۶/۵، ۸۹ و ۷۲/۹ به دست آمده است. همچنین عملکرد HOSRDA را با روش برداری آن (SRDA) مقایسه کردیم و روشن شد که تانسوری کردن باعث بهبود روشنی در عملکرد در مسأله‌ی SSS خواهد شد. در پایان نیز ماتریس پایه‌ی متناظر با بعد کانال را در زیرفضای HOSRDA بررسی کرد و دیدیم که وزن‌دهی کانال‌ها با توجه به اهمیت آن‌ها به خوبی صورت گرفته است.

فصل ۷

طبقه‌بندی بدون سرپرست و فرم پیشنهادی تانسوری آن

۱-۷ پیش‌گفتار

طبقه‌بندی با سرپرست در P300 speller محدودیت‌هایی را ایجاد می‌کند که از آن جمله می‌توان به جلسات کالیبراسیون طولانی اشاره کرد که وقت‌گیر و خسته‌کننده هستند. در [۵۰] یک طبقه‌بند بدون سرپرست برای استفاده در P300 speller معرفی شده است که از EM^۱ استفاده می‌کند. در واقع منظور از بدون سرپرست این است که برای پیش‌بینی حروف هجی‌شده به وسیله‌ی کاربر، نیازی به داده‌های آموزش برچسب‌دار نیست. این روش به خاطر حذف فرآیند آموزش و سهولت در استفاده از BCI بسیار حائز اهمیت است. در بخش ۲-۷ به توضیح این طبقه‌بند بدون سرپرست می‌پردازیم. پس از آن در بخش ۳-۷ فرمول‌بندی تانسوری شده‌ی آن را پیشنهاد داده و نتایج آن را در بخش ۷-۴ بررسی خواهیم کرد. مطالب بخش ۲-۷ با استفاده از مقاله‌ی [۵۰] نگارش شده‌اند.

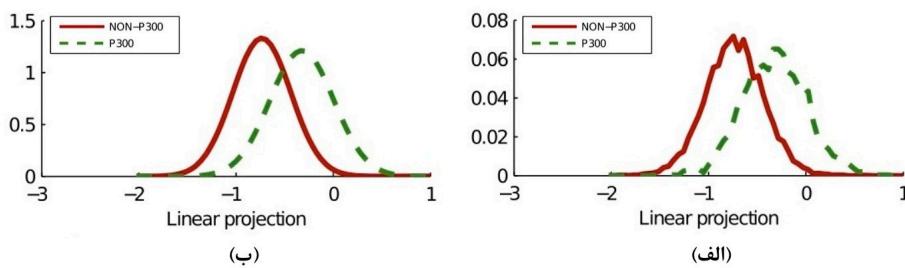
۲-۷ طبقه‌بند بدون سرپرست برای P300 speller

در [۵۰] روشی برای طبقه‌بندی بدون سرپرست در P300 speller معرفی شده است که از [۵۱] Expectation Maximization بهره برده است. طبقه‌بند پایه‌ی این روش مشابه BLDA^۲ یا BLR^۳ است.

^۱Expectation Maximization

^۲Bayesian Linear Discriminant Analysis

^۳Bayesian Linear Regression



شکل ۷-۱: دو کلاس P300 و فاقد P300 پس از تصویر شدن بر یک بعد دارای توزیع نرمال با واریانس مشابهی هستند: (الف) هیستوگرام دو کلاس. (ب) توزیع گوسی براز شده به هیستوگرامها [۵۰].

است. در ابتدا به توضیح مختصر آن می‌پردازیم.

BLDA ۱-۲-۷

مسئله‌ی BLDA به طور ساده یک مسئله‌ی رگرسیون است که $\{x_m \in \mathbb{R}^{1 \times (D+1)}, m = 1, \dots, M\}$ داده‌های آموزش و $\{t_m, m = 1, \dots, M\}$ برچسب‌های آن‌ها هستند و می‌خواهیم بردار $w \in \mathbb{R}^{(D+1) \times 1}$ را به دست آوریم به طوری که در آن $t_i = x_m w + \epsilon$ است و $\epsilon \sim N(0, \sigma^2)$. که در آن x_m و w معادل بایاس رگرسیون است. اکنون برای تعریف BLDA لازم است فرض‌های زیر را داشته باشیم:

$$p(t_m = 1) = \gamma$$

$$p(t_m = -1) = 1 - \gamma$$

$$p(w) = N(\cdot, \alpha^{-1} I)$$

$$p(x_m | t_m, w) = N(x_m | t_m, \beta^{-1})$$

که در آن $p(t_m = 1)$ احتمال حضور P300 در نمونه‌ی x_m است. همان‌طور که مشاهده می‌شود، فرض شده است که دو کلاس P300 و فاقد P300 پس از تصویر شدن بر یک بعد دارای توزیع نرمال با واریانس مشابهی هستند (شکل ۷-۱). نرمال بودن توزیع تصویر داده‌ها بر یک بعد با قضیه‌ی حد مرکزی قابل توجیه است: تصویر داده‌ها بر یک بعد در واقع ترکیب خطی تعداد زیادی متغیر تصادفی است که بر اساس قضیه‌ی حد مرکزی توزیع نرمال دارد.

اکنون احتمال پسین w را به صورت زیر خواهیم داشت:

$$p(w | \beta, \alpha, X, t) = \frac{p(w)p(X|t, w)}{\int p(w)p(X|t, w)dw} \quad (1-7)$$

که در آن $\mathbf{X} \in \mathbb{R}^{M \times D}$ همه‌ی داده‌های آموزش را در خود دارد. می‌توان نشان داد که احتمال پسین بالا دارای توزیع نرمال با میانگین زیر است:

$$\mathbf{m} = (\mathbf{X}^T \mathbf{X} + \frac{\alpha}{\beta} \mathbf{I})^{-1} \mathbf{X}^T \mathbf{t} \quad (2-7)$$

اکنون اگر x یک داده‌ی جدید باشد که می‌خواهیم برچسب آن را تخمین بزنیم، توزیع پیش‌گو به صورت زیر خواهد بود:

$$p(\hat{\mathbf{t}}|\beta, \alpha, \mathbf{x}, \mathbf{X}, \mathbf{t}) = \int p(\hat{\mathbf{t}}|\beta, \mathbf{x}, \mathbf{w}) p(\mathbf{w}|\beta, \alpha, \mathbf{X}, \mathbf{t}) d\mathbf{w} \quad (3-7)$$

که دارای توزیع نرمال با میانگین $\mathbf{x}\mathbf{m}$ است. بنابراین با روش Maximum likelihood می‌توانیم تخمین w را به صورت زیر داشته باشیم:

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X} + \frac{\alpha}{\beta} \mathbf{I})^{-1} \mathbf{X}^T \mathbf{t} \quad (4-7)$$

۲-۲-۷ طبقه‌بند بدون سرپرست

فرض کنید داده‌های یک P300 speller در دست است که مربوط به هجی کردن N کاراکتر است. ماتریس P300 speller دارای K سطر و K ستون است. برای کاراکتر n هریک از سطر و ستون‌ها R_n بار روش می‌شوند. $x_{n,r,k}^\omega$ سیگنال EEG مربوط به کاراکتر n است در زمان تکرار r ام سطر ($\omega = 0$) یا ستون ($\omega = 1$) است. به همین ترتیب $t_{n,r,k}^\omega$ برچسب $x_{n,r,k}^\omega$ است که اگر برابر ۱ باشد به معنی این است که سطر یا ستون فوق دارای P300 بوده است و اگر برابر -۱ باشد به معنی این است سطر یا ستون فوق فاقد P300 است.

در این پارادایم فرض می‌شود که برای هر کاراکتر یک سطر و یک ستون وجود دارد که حاوی P300 است؛ این سطر (یا ستون) را با پارامتر z_n^ω نشان می‌دهیم. این پارامتر مقداری بین ۱ تا K می‌گیرد (برای مثال اگر ماتریس P300 speller مشابه بخش ۱-۲-۶ باشد و اولین کاراکتر هجی شده B باشد، $z_1^\omega = 1$ و

اکنون فرض‌های زیر را خواهیم داشت:

$$p(\mathbf{w}) = \mathcal{N}(\cdot, \alpha^{-1} \mathbf{I}) \quad (5-7)$$

$$p(z_n^\omega = k) = \frac{1}{K} \quad (6-7)$$

$$p(t_{n,r,k}^\omega = 1 | z_n^\omega) = \begin{cases} 1, & z_n^\omega = k \\ \cdot, & z_n^\omega \neq k \end{cases} \quad (7-7)$$

$$p(t_{n,r,k}^\omega = -1 | z_n^\omega) = \begin{cases} \cdot, & z_n^\omega = k \\ 1, & z_n^\omega \neq k \end{cases} \quad (8-7)$$

$$p(\mathbf{x}_{n,r,k}^\omega | t_{n,r,k}^\omega, \mathbf{w}) = \mathcal{N}(\mathbf{x}_{n,r,k}^\omega \mathbf{w} | t_{n,r,k}^\omega, \beta^{-1}) \quad (9-7)$$

ازین پس از بالانویس ω صرف نظر می‌کنیم و فرض می‌کنیم که $2N$ شبه-کاراکتر داریم. شبه-کاراکترها در واقع داده‌های مربوط به سطرهای (یا ستون‌های) کاراکترها هستند. اکنون مسئله یافتن پارامترهای $\{\mathbf{w}, \beta\}$ است زمانی که داده‌هایی را مشاهده کرده‌ایم (\mathbf{X}). همچنین مسئله یک پارامتر مخفی دارد (\mathbf{z}_n). این مسئله با استفاده از EM حل می‌شود. در EM باید مسئله‌ی زیر را حل کنیم:

$$\mathbf{w}_m, \beta_m = \underset{\mathbf{w}, \beta}{\operatorname{argmax}} \sum_z p(\mathbf{z} | \mathbf{X}, \mathbf{w}_p, \beta_p) \log p(\mathbf{X}, \mathbf{z} | \mathbf{w}, \beta) + \log p(\mathbf{w} | \alpha) \quad (10-7)$$

مقادیر بهینه‌ی \mathbf{w} و β به وسیله‌ی مشتق‌گیری از رابطه‌ی بالا به دست می‌آیند. رابطه‌ی بهروزسازی \mathbf{w} به صورت زیر است:

$$\mathbf{w}_m = \sum_z p(\mathbf{z} | \mathbf{X}, \mathbf{w}_p) (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \tilde{T}(\mathbf{z}) \quad (11-7)$$

که در آن $(\tilde{T}(\mathbf{z}))$ بردار برچسب‌های متناظر با تحریک‌های \mathbf{X} است به شرطی که \mathbf{z} را بدانیم. رابطه‌ی بهروزسازی β و α به صورت خواهد بود:

$$\beta^{-1} = \left\langle \sum_z p(z_n | \mathbf{X}_n, \mathbf{w}_p, \beta_p) (\mathbf{x}_{n,r,k} \mathbf{w} - \tilde{t}_{n,r,k})^2 \right\rangle_{n,r,k} \quad (12-7)$$

$$\alpha = \frac{D}{\mathbf{w} \mathbf{w}^T} \quad (13-7)$$

بنابراین مراحل الگوریتم طبقه‌بندی بدون سرپرست، که در بالا توضیح داده شد به صورت زیر خواهد بود:

۱. مقداردهی اولیه‌ی پارامترهای \mathbf{w} , β و α

۲. محاسبه‌ی $p(z_n | \mathbf{X}_n, \mathbf{w}, \beta)$ به ازای $n = 1, \dots, 2N$ و $z_n = 1, K$

۳. به روزرسانی پارامترها از روابط ۱۱-۷ تا ۱۳-۷.

۴. اگر الگوریتم همگرا شده است، پایان. در غیر این صورت بازگشت به مرحله‌ی ۲.

از آن جا که تابع likelihood دارای اکسترمم‌های محلی بسیاری است، برای یافتن طبقه‌بند بهینه این روش پیشنهاد شده است: به ازای هر مقداردهی اولیه‌ی \mathbf{w} دو طبقه‌بند آموزش داده می‌شود (یکی با \mathbf{w} و دیگری با \mathbf{w}'). اکنون به ازای ۱۰ مقداردهی اولیه (که در واقع ۲۰ طبقه‌بند خواهد بود) فرآیند آموزش بدون سرپرست را انجام می‌دهیم. از میان این ۲۰ طبقه‌بند، طبقه‌بند با بیشتری مقدار likelihood را انتخاب می‌کنیم. فرآیند فوق را ۱۰ بار تکرار می‌کنیم. بنابراین در نهایت ۱۰ طبقه‌بند خواهیم داشت. برای گزارش عملکرد روش، میانگین و انحراف معیار دقت تشخیص کاراکتر این ۱۰ طبقه‌بند را گزارش می‌کنیم.

۷-۳ فرم پیشنهادی تانسوری

در بخش قبل برای تشخیص کاراکتر داده‌های هر شبکه‌کاراکتر را در یک ماتریس قرار دادیم (\mathbf{X}_n). اکنون اگر تکرارهای مربوط به هر شبکه‌کاراکتر را در یک تانسور با ابعاد $Time \times Channel \times Trial$ بچینیم، می‌توانیم دو بردار پایه برای زیرفضای ویژگی به دست آوریم. به عبارتی داشته باشیم:

$$p(\mathbf{w}_1) = \mathcal{N}(\mathbf{0}, \alpha_1^{-1} \mathbf{I}) \quad (14-7)$$

$$p(\mathbf{w}_2) = \mathcal{N}(\mathbf{0}, \alpha_2^{-1} \mathbf{I}) \quad (15-7)$$

$$p(\mathcal{X}_n | \mathbf{t}_n, \mathbf{w}_1, \mathbf{w}_2) = \mathcal{N}(\mathbf{y}_n | \mathbf{t}_n, \beta^{-1} \mathbf{I}) \quad (16-7)$$

که در آن \mathcal{X}_n و \mathbf{t}_n به ترتیب تانسور سه بعدی و بردار برچسب‌های متناظر با شبکه‌کاراکتر n هستند. همچنین:

$$\mathcal{X}_n \in \mathbb{R}^{I_1 \times I_2 \times KR_n}, \quad \mathbf{w}_1 \in \mathbb{R}^{1 \times I_1}, \quad \mathbf{w}_2 \in \mathbb{R}^{1 \times I_2} \quad (17-7)$$

$$\mathcal{Y}_n = \mathcal{X}_n \times_1 \mathbf{w}_1 \times_2 \mathbf{w}_2 = [\mathcal{X}_n; \mathbf{w}_1, \mathbf{w}_2] \in \mathbb{R}^{1 \times 1 \times KR_n} \quad (18-7)$$

$$\mathbf{y}_n = \mathcal{Y}_n \in \mathbb{R}^{KR_n \times 1} \quad (19-7)$$

الگوریتم ۹ فرم پیشنهادی تانسوری طبقه‌بند بدون سرپرست

Input: $\mathcal{X}_n, n = 1, \dots, 2N$ **Output:** the number of the row and columns containing P300.

- 1: Initialization for $\theta_p = \{\mathbf{w}_1, \mathbf{w}_2, \beta_p, \alpha_1, \alpha_2\}$
- 2: **while** not convergent **do** ▷ convergence of main algorithm
- 3: **while** not convergent **do** ▷ convergence of tensor algorithm
- 4: **for** $m = 1, 2$ **do**
- 5: substitute \mathbf{w}_m and α_m and in θ_p with new updated ones. ▷ from 2nd iteration
- 6: compute $p(z_n|\mathcal{X}_n, \theta_p)$, $n = 1, \dots, 2N$ and $z_n = 1, \dots, K$
- 7: update \mathbf{w}_m and α_m .
- 8: update β_m .
- 9: **end for**
- 10: **end while**
- 11: **end while**
- 12: compute $p(z_n|\mathcal{X}_n, \theta_p)$, $n = 1, \dots, 2N$ and $z_n = 1, \dots, K$
- 13: for each n, mark the row (column) with highest probability as the one containing P300.

که در آن I_2 و I_1 به ترتیب تعداد کانال‌ها و نمونه‌های زمانی هر تکثیت هستند. در این صورت:

$$p(z_n|\mathcal{X}_n, \theta_p) = \frac{\mathcal{N}(\mathbf{y}_n|\tilde{\mathbf{t}}_n(z_n), \beta^{-1}\mathbf{I})}{\sum_{z_n} \mathcal{N}(\mathbf{y}_n|\tilde{\mathbf{t}}_n(z_n), \beta^{-1}\mathbf{I})} \quad (20-7)$$

که در آن θ_p مجموعه‌ی پارامترهای $\mathbf{w}_1, \mathbf{w}_2$ و β_p است. مسئله‌ی بهینه‌سازی مشابه با رابطه‌ی ۷-۱۰ است. به صورت بیشینه‌سازی تابع زیر نوشته می‌شود:

$$J(\mathbf{w}_1, \mathbf{w}_2, \beta) = \sum_z p(z|\mathcal{X}, \theta_p) \log p(\mathcal{X}, z|\theta) + \log p(\mathbf{w}_1|\alpha_1) + \log p(\mathbf{w}_2|\alpha_2) \quad (21-7)$$

که در آن \mathcal{X} تانسور سه بعدی کل شبکه‌کاراکترهاست. می‌توان نوشت:

$$\begin{aligned} J(\mathbf{w}_1, \mathbf{w}_2, \beta) &= \sum_{n=1}^{2N} \sum_{z_n=1}^K p(z_n|\mathcal{X}_n, \theta_p) \log p(\mathcal{X}_n|\tilde{\mathbf{t}}_n(z_n), \theta) \\ &\quad + \sum_{n=1}^{2N} \sum_{z_n=1}^K p(z_n|\mathcal{X}_n, \theta_p) \log p(z_n) \\ &\quad + \log p(\mathbf{w}_1|\alpha_1) + \log p(\mathbf{w}_2|\alpha_2) \end{aligned} \quad (22-7)$$

برای به دست آوردن روابط به روزرسانی پارامترها از طریق مشتق‌گیری رابطه‌ی بالا، به دلیل تشابهی که با رابطه‌ی ۷-۱۰ وجود دارد، رابطه‌ی به روزرسانی β به صورت سرراست به دست می‌آید:

$$\beta^{-1} = \frac{1}{2NR_nK} \sum_{n=1}^{2N} \sum_{z_n=1}^K p(z_n|\mathcal{X}_n, \theta_p) \|\mathbf{y}_n - \tilde{\mathbf{t}}(z_n)\|^2 \quad (23-7)$$

برای به دست آوردن رابطه‌ی به روزرسانی \mathbf{w}_1 لازم است از ترم اول رابطه‌ی ۷-۲۲ نسبت به \mathbf{w}_1

مشتق بگیریم:

$$\begin{aligned} \frac{d}{dw_1} \log p(\mathcal{X}_n | \tilde{\mathbf{t}}(z_n), \mathbf{w}_1, \mathbf{w}_2, \beta) &= \frac{d}{dw_1} (-\frac{\beta}{2} \|\mathbf{Y}_{n(1)} - \tilde{\mathbf{t}}(z_n)^T\|^2) \\ &= -\beta (\mathbf{Y}_{n(1)} - \tilde{\mathbf{t}}(z_n)^T) \frac{d\mathbf{Y}_{n(1)}}{dw_1} \\ &= -\beta (\mathbf{w}_1 \mathbf{H}_{n(1)}^{(-1)} - \tilde{\mathbf{t}}(z_n)^T) ((\mathbf{w}_2 \otimes \mathbf{I}_K) \mathbf{X}_{n(1)}^T) \end{aligned} \quad (24-7)$$

که در آن $(\mathbf{w}_2 \otimes \mathbf{I}_K) \otimes \mathbf{X}_{n(1)}^T = \mathbf{H}_{n(1)}^{(-1)^T}$. همچنین می‌توان دید که $\mathcal{H}_n^{(-1)} = \mathcal{X}_n \times_2 \mathbf{w}_2 \in \mathbb{R}^{I_1 \times 1 \times KR_n}$.

بنابراین رابطه‌ی به روزرسانی w_1 به صورت زیر خواهد بود:

$$\mathbf{w}_1 = \sum_n \sum_{z_n} (p(z_n | \mathcal{X}_n, \theta_p) \tilde{\mathbf{t}}(z_n)^T \mathbf{H}_{n(1)}^{(-1)^T}) (\mathbf{H}_{n(1)}^{(-1)} \mathbf{H}_{n(1)}^{(-1)^T})^{-1} \quad (25-7)$$

به طریق مشابه برای w_2 داریم:

$$\mathbf{w}_2 = \sum_n \sum_{z_n} (p(z_n | \mathcal{X}_n, \theta_p) \tilde{\mathbf{t}}(z_n)^T \mathbf{H}_{n(2)}^{(-2)^T}) (\mathbf{H}_{n(2)}^{(-2)} \mathbf{H}_{n(2)}^{(-2)^T})^{-1} \quad (26-7)$$

که در آن $\mathcal{H}_n^{(-2)} = \mathcal{X}_n \times_1 \mathbf{w}_1 \in \mathbb{R}^{1 \times I_2 \times KR_n}$. الگوریتم ۹ شبکه‌کد فرم تانسوری طبقه‌بند بدون سرپرست را نشان می‌دهد. دقت کنید که در این الگوریتم به ازای هر مُد (در حلقه‌ی سطر ۴) پارامتر β نیز به روز می‌شود. در حالی که می‌دانیم این پارامتر باید برای هر دو مُد تانسور یکسان باشد. در عمل دیده شد که پس از همگرایی $\beta_2 = \beta_1$ برقرار است. هرچند می‌توان به روزرسانی β را خارج از حلقه‌ی سطر ۴ انجام داد، اما به روزرسانی آن همراه با ماتریس‌های پایه‌ی تجزیه‌ی تانسوری به دقت محاسبات کمک می‌کند. با نگاهی به این شبکه‌کد می‌توان حدس زد که نتایج شبیه‌سازی نباید از فرم برداری پیشی بگیرد؛ چرا که خود الگوریتم اصلی به صورت تکراری است (حلقه‌ی سطر ۲) و اضافه شدن یک حلقه‌ی دیگر داخل آن همگرایی به نقطه‌ی بهینه را دچار مشکل خواهد کرد.

۴-۷ شبیه‌سازی

برای مقایسه‌ی کارایی فرم تانسوری طبقه‌بند بدون سرپرست با طبقه‌بند معرفی شده در بخش ۲-۲-۲، از مجموعه‌دادگان BCI Competition II-Dataset IIb استفاده کرده‌ایم [۵۲]. این مجموعه‌داده با پروتکلی مشابه با داده‌های BCI Competition III-Dataset II (توضیح داده شده در بخش ۱-۲-۶) از یک سوژه ثبت شده است و داده‌های ۳۷ حرف برای داده‌های آموزش و ۳۱ حرف برای داده‌های آزمون در دسترس قرار دارد. دلیل انتخاب این مجموعه‌داده برای ارزیابی در اینجا این است که کار با این دادگان نسبت داده‌های مسابقه‌ی سوم ساده‌تر هستند. منظور از ساده این است نویز داده‌ها کمتر و P300 در آن‌ها

مشهودتر است و تقریباً تمامی روش‌های موجود در طبقه‌بندی P300 speller در این داده‌ها به دقت ۱۰۰ درصد دست می‌یابند (روش پیشنهادی ما، HOSRDA)، نیز برای این داده‌ها دقت ۱۰۰ درصد دارد. دلیل دیگر این است که در [۵۰] (روش بخش ۷-۲-۲) برای این مجموعه داده، در پردازش غیربرخط^۴، دقت تشخیص کاراکتر ۱۰۰ درصد با انحراف معیار صفر برای داده‌های آزمون گزارش شده است (ما هم با شبیه‌سازی که انجام دادیم نتیجه مشابهی را برای این مجموعه داده‌ها به دست آوردیم). به همین دلیل این داده‌ها برای بررسی‌های اولیه‌ی کارایی هر الگوریتمی بسیار مناسب است، چرا که اگر الگوریتمی در این داده‌ها در به دست آوردن دقت ۱۰۰ درصد ناموفق باشد، چراکی آن را باید به صورت جدی بررسی کرد. پیش‌پردازش انجام شده به صورت مراحل زیر است:

۱. اعمال فیلتر CAR.^۵

۲. فیلتر میان‌گذر در باند ۵/۰ تا ۱۵ هرتز.

۳. نرمالیزه کردن هر کanal به میانگین صفر و واریانس یک.

۴. جداسازی تکثیت‌ها و کاهش نرخ نمونه‌برداری به طوری که برای هر تکثیت ۱۰ نمونه‌ی زمانی باقی بماند.

ابتدا به نظر می‌رفت تانسوری کردن روش [۵۰] شاید بتواند از اطلاعات زمان و مکان بهتر استفاده کند. در شبیه‌سازی‌ها فرم تانسوری طبقه‌بند بدون سرپرست از ۳۱ کاراکتر آزمون، ۳۰ کاراکتر را به درستی پیش‌بینی می‌کند. هرچند این نتیجه چندان بد به نظر نمی‌رسد، اما برای این مجموعه داده دقت ۱۰۰ درصد مورد نظر است. اول این‌که از آن‌جا که در نسخه‌ی تانسوری دو بردار پایه داریم، در هرگروه به تعداد بیشتری طبقه‌بند نیاز داریم. در شبیه‌سازی‌ها ما از ۱۰ گروه ۴۰ تایی استفاده کردیم. این مسئله همراه با حلقه‌های تکرار اضافه شده زمان اجرای برنامه را به صورت زیادی افزایش می‌دهد. در روش تانسوری، حلقه‌ی تکراری با روش جایگزینی (سطر ۳ از شبه‌کد) که در هر بار تکرار حلقه‌ی اصلی (سطر ۲ از شبه‌کد) وجود دارد، نقطه ضعف بزرگ الگوریتم است. به صورت کلی طبقه‌بند بدون سرپرست (روش برداری) نسبت به مقداردهی اولیه بسیار حساس است. به همین دلیل وقتی در این حلقه‌ی سطر ۳ از شبکه کد در هر بار تکرار احتمال $p(z_n | \mathcal{X}_n, \theta_p)$ به فرض بردارهای پایه‌ای که هنوز همگرا نشده‌اند محاسبه

^۴Offline

^۵Common Average Reference

می‌شود، به شدت بر عملکرد الگوریتم تاثیر می‌گذارد.

دلیل دیگری نیز می‌توان برای کاهش کیفیت به دلیل تانسوری کردن متصور شد: الگوریتم طبقه‌بند بدون سرپرست (و همین طور BLDA) بر اساس شواهد آماری کار می‌کنند. به همین دلیل به تعداد کافی نمونه نیاز دارند. در نسخه‌ی این الگوریتم در هربار تکرار از حلقه‌ی سطر ۴ این فرض را خواهیم داشت:

$$p(\mathcal{H}_n^{(-m)} | \mathbf{t}_n, \mathbf{w}_m) = \mathcal{N}(\mathbf{h}_n | \mathbf{t}_n, \beta^{-1} \mathbf{I}) \quad (27-7)$$

که در آن $\mathcal{H}_n^{(-m)} = (\mathcal{H}_n^{(-m)} \times_m \mathbf{w}_p)_{(3)}$ درواقع KR_n داده‌ی I_m بُعدی است و تعداد «کافی» از لحاظ آماری نیست، فرض نرمال بودن بالا دچار تزلزل می‌شود. برای مثال برای بعد اول، $I_1 = 10$ که از دیدگاه آماری تعداد واقعاً کمی به شمار می‌آید.

۵-۷ جمع‌بندی

در این فصل طبقه‌بند بدون سرپرست برای P300 spletter را بررسی کردیم و فرمول‌بندی تانسوری آن را معرفی کردیم. دیدیم که عملکرد نسخه‌ی تانسوری ضعیفتر است. بر اساس تجربیات شبیه‌سازی فرم تانسوری طبقه‌بند بدون سرپرست و تجربه‌ی تعمیم تانسوری BLDA برای نگارنده، توضیحاتی برای دلایل این ضعف داده شد که دو گزاره‌ی کلی زیر برای جمع‌بندی در این زمینه مطرح می‌شود:

۱. تانسوری کردن الگوریتم‌هایی که خودشان تکراری هستند، بهبودی در عملکرد آن‌ها ایجاد نمی‌کند. چرا که حلقه‌ی تکراری مربوط به تخمین بردارهای (یا ماتریس‌های) پایه‌ی تجزیه‌ی تانسوری باعث ناپایداری الگوریتم می‌شود.

۲. در روش‌هایی که مبنای آماری دارند (همانند طبقه‌بند بدون سرپرست و BLDA) تانسوری کردن به دلیل متزلزل کردن برقراری فرض‌هایی که به واسطه‌ی قضیه‌ی حد مرکزی (و امثال آن) درنظر گرفته می‌شود احتمالاً باعث بدتر شدن عملکرد می‌شود.

فصل ۸

جمع‌بندی، نتیجه‌گیری و پیشنهادها

۱-۸ جمع‌بندی و نتیجه‌گیری

در سال‌های اخیر استفاده از تانسورها در پردازش سیگنال EEG بسیار مورد استقبال قرار گرفته است و دو تجزیه‌ی تانسوری Tucker و CP برای مطالعه‌ی دادگان حاوی ERP به صورت گستردگی مورد استفاده قرار گرفته‌اند. مطالعات انجام شده روی کاربرد روش‌های تجزیه‌ی تانسوری در پردازش سیگنال EEG حاوی ERP را می‌توان به دو دسته‌ی کلی تقسیم کرد: گروه اول از تجزیه‌ی تانسوری برای جداسازی منابع و استخراج الگوی ERP استفاده می‌کنند و گروه دوم تجزیه‌ی تانسوری را برای کاهش ویژگی، به هدف بررسی تمایزپذیری دو کلاس و یا طبقه‌بندی، به کار بسته‌اند.

هدف این پایان‌نامه بررسی ادبیات موجود در استفاده از روش‌های تجزیه‌ی تانسوری در مطالعه‌ی داده‌های ERP و ارائه‌ی یک روش جدید بر مبنای تجزیه‌ی تانسوری برای تشخیص پتانسیل وابسته به رخداد بوده است، به طوری که نتایج ارزیابی آن با روش‌های دیگر توانایی رقابت را داشته باشد.

در این نوشتار ابتدا مفاهیم اولیه و نیز پیش‌زمینه‌ی نظری مرتبط با پژوهش را معرفی کردیم. در بخش پیش‌زمینه‌ی نظری دو روش برداری کاهش ویژگی توضیح داده شده و مروری سریع بر مفاهیم اصلی جبر تانسوری و برخی روش‌های کاهش ویژگی تانسوری داشتیم. پس از آن به مرور ادبیات مرتبط BCI Competition که شامل مرور روش‌های برداری استفاده شده برای پردازش داده‌های III-Dataset II و هم‌چنین کاربرد روش‌های تجزیه‌ی تانسوری در مطالعات مرتبط با پتانسیل‌های وابسته به رخداد است.

در این پایان‌نامه فرمول‌بندی یک روش جدید کاهش ویژگی مبتنی بر تجزیه‌ی تانسوری معرفی کردیم که آن را Higher Order Spectral Regression Discriminat Analysis، یا به اختصار HOSRDA، نامیدیم. این روش تعمیم تانسوری روش SRDA است که در پیشینه‌ی نظری توضیح داده شد. همچنان پیچیدگی محاسباتی روش پیشنهادی را بر حسب تعداد فلأپ محاسبه کردیم و نشان دادیم برای یک تانسور $K \times I_1 \times I_2$ در هر تکرار از حلقه‌ی اصلی الگوریتم از مرتبه‌ی $O(KI^2)$ است که $I = \max(I_1, I_2)$. شیوه‌ی انتخاب پارامتر و پایداری همگرایی الگوریتم پیشنهادی را در شبیه‌سازی بررسی کردیم و به تجربه نشان دادیم که الگوریتم در همگرایی‌اش یک پایداری نسبی دارد و همگرایی آن به ویژگی‌های داده‌های آموزش وابسته نیست.

برای ارزیابی الگوریتم پیشنهادی از آن برای تشخیص کاراکتر یک P300 speller مربوط به مجموعه‌دادگان BCI Competition III-Dataset II استفاده کردیم. روش پیشنهادی به طور میانگین برای دو سوژه زمانی که ۱۵، ۱۰ و ۵ تکرار برای هر تحریک استفاده شود، به ترتیب به دقت تشخیص کاراکتر ۹۶/۵، ۸۹ و ۷۲/۹ دست یافت که کاملاً قابل مقایسه با سایر نتایج گزارش شده بر این مجموعه داده است. در پایان نیز وزن‌دهی کانال‌ها را در زیرفضای HOSRDA از روی ماتریس عامل متناظر با بعد کانال بررسی کردیم و با مقایسه با ادبیات دیدیم که وزن‌دهی انجام شده معتبر است.

همچنان طبقه‌بند بدون سرپرست P300 speller و فرم پیشنهادی تانسوری آن را معرفی کردیم و به بررسی دلایل عدم موفقیت تانسوری کردن این روش پرداختیم. در این‌باره دو گزاره‌ی کلی را مطرح کردیم که طبیعتاً نیاز به بررسی بیشتری دارند: ۱) تعمیم تانسوری الگوریتم‌های تکراری نمی‌تواند باعث بهبود آن‌ها شود. دلیل این اتفاق هم می‌تواند این باشد که حلقه‌ی تکراری مربوط به تخمین بردارهای (یا ماتریس‌های) پایه‌ی تجزیه‌ی تانسوری باعث ناپایداری الگوریتم می‌شود. ۲) تعمیم تانسوری روش‌هایی که مبنای آماری دارند، برقراری فرض‌هایی که به واسطه‌ی قضیه‌ی حد مرکزی (و امثال آن) در نظر گرفته می‌شوند را از بین می‌برد. به همین دلیل تانسوری کردن آن‌ها می‌تواند باعث بدتر شدن نتایج شود.

۲-۸ پیشنهادها

بر اساس تجربیاتی که در مراحل این پژوهش به دست آمد، پیشنهادات زیر برای کارهای آینده ارائه

می‌شود:

گسترش HOSRDA: همان‌طور که قبلاً توضیح داده شد، HOSRDA ساختار جدیدی را برای همراه کردن قیود رگولاریزاسیون بر روی ماتریس‌های عامل تجزیه‌ی تانسوری فراهم می‌سازد. از کارهایی که برای ادامه‌ی مسیر پیشنهاد می‌شود حل مسئله به همراه قیودی نظری^۱ تُنکی است که البته بسته به کاربرد باید اعمال شود. می‌توان HOSRDA را برای کاهش ویژگی در حوزه‌های دیگر نظری تشخیص چهره^۲ و تشخیص تصور حرکت^۳ به کار برد و بسته به کاربرد قیود مناسب را اعمال کرد.

طبقه‌بندی: در الگوریتم پیشنهادی ویژگی‌های به دست آمده از HOSRDA به بردار تبدیل و به طبقه‌بند LDA داده شدند. با توجه به این که ویژگی‌های به دست آمده از HOSRDA تانسور هستند، ممکن است طبقه‌بندهای تانسوری (نظری Support Tensor Machine) برای طبقه‌بندی آن‌ها مناسب‌تر باشند.

کرنلیزه کردن HOSRDA: یکی از حوزه‌های داغ روش‌های مبتنی بر تجزیه‌های تانسوری، کرنلیزه کردن آن‌هاست. کرنلیزه کردن HOSRDA می‌تواند برای آن در مسائل غیرخطی کاربرد ایجاد کند.

فرآیند گوسی تانسوری: با توجه به تجربه‌ی به دست آمده از تانسوری کردن BLDA و طبقه‌بند بدون سرپرست، پیشنهاد می‌شود کارایی فرآیند گوسی و فرم تانسوری آن در طبقه‌بندی P300 speller ارزیابی و مقایسه شوند. فرمول‌بندی‌هایی برای فرآیند گوسی تانسوری موجود است، هرچند ممکن است نیاز به تغییراتی داشته باشد.

^۱face recognition

^۲motor imagery

مراجع

- [1] L. W. Lih, “Multi-channel extraction methods for event-related potentials,” Ph.D. dissertation, Curtin University, Department of Electrical and Computer Engineering, 2014.
- [2] U. Hoffmann *et al.*, “An efficient p300-based brain-computer interface for disabled subjects,” *Journal of Neuroscience methods*, vol. 167, no. 1, pp. 115–125, 2008.
- [3] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.
- [4] H. A. Kiers, “Towards a standardized notation and terminology in multiway analysis,” *Journal of chemometrics*, vol. 14, no. 3, pp. 105–122, 2000.
- [5] L. De Lathauwer *et al.*, “A multilinear singular value decomposition,” *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.
- [6] A. Cichocki *et al.*, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation.* John Wiley & Sons, 2009.
- [7] T. G. Kolda, *Multilinear operators for higher-order decompositions.* United States. Department of Energy, 2006.
- [8] M. Mørup *et al.*, “Parallel factor analysis as an exploratory tool for wavelet transformed event-related eeg,” *NeuroImage*, vol. 29, no. 3, pp. 938–947, 2006.
- [9] M. Mørup *et al.*, “Shift-invariant multilinear decomposition of neuroimaging data,” *NeuroImage*, vol. 42, no. 4, pp. 1439–1450, 2008.
- [10] F. Cong *et al.*, “Benefits of multi-domain feature of mismatch negativity extracted by non-negative tensor factorization from eeg collected by low-density array,” *International journal of neural systems*, vol. 22, no. 06, p. 1250025, 2012.
- [11] K. Vanderperren *et al.*, “Single trial erp reading based on parallel factor analysis,” *Psychophysiology*, vol. 50, no. 1, pp. 97–110, 2013.
- [12] BCI Competition III, Dataset II discription. [Online]. Available: <http://www.bbci.de/competition/iii/>
- [13] A. Rakotomamonjy and V. Guigue, “Bci competition iii: dataset ii-ensemble of svms for bci p300 speller,” *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 3, pp. 1147–1154, 2008.

- [14] S. J. Luck, *An introduction to the event-related potential technique*. MIT press, 2014.
- [15] A. Cichocki *et al.*, “Noninvasive bcis: Multiway signal-processing array decompositions.” *IEEE Computer*, vol. 41, no. 10, pp. 34–42, 2008.
- [16] Y. Zhang *et al.*, “Spatial-temporal discriminant analysis for erp-based brain-computer interface,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 2, pp. 233–243, 2013.
- [17] D. Cai *et al.*, “Srda: An efficient algorithm for large-scale discriminant analysis,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 1, pp. 1–12, 2008.
- [18] A. Cichocki *et al.*, “Tensor decompositions for signal processing applications from two-way to multiway component analysis,” *arXiv preprint arXiv:1403.4462*, 2014.
- [19] L. R. Tucker, “Some mathematical notes on three-mode factor analysis,” *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.
- [20] A. Phan and A. Cichocki, “Tensor decompositions for feature extraction and classification of high dimensional datasets,” *Nonlinear theory and its applications, IEICE*, vol. 1, no. 1, pp. 37–68, 2010.
- [21] R. Bro and H. Kiers, “A new efficient method for determining the number of components in parafac models,” *Journal of chemometrics*, vol. 17, no. 5, pp. 274–286, 2003.
- [22] A. Cichocki, “Tensor decompositions: a new concept in brain data analysis?” *arXiv preprint arXiv:1305.0395*, 2013.
- [23] F. Cong *et al.*, “Tensor decomposition of eeg signals: A brief review,” *Journal of neuroscience methods*, vol. 248, pp. 59–69, 2015.
- [24] S. Yan *et al.*, “Discriminant analysis with tensor representation,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005*, vol. 1. IEEE, 2005, pp. 526–532.
- [25] W. Zhang *et al.*, “Tensor linear laplacian discrimination (tlld) for feature extraction,” *Pattern Recognition*, vol. 42, no. 9, pp. 1941–1948, 2009.
- [26] D. Zhao *et al.*, “Linear laplacian discrimination for feature extraction,” in *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR’07*. IEEE, 2007, pp. 1–7.
- [27] B. W. Bader and T. G. Kolda, “Algorithm 862: MATLAB tensor classes for fast algorithm prototyping,” *ACM Transactions on Mathematical Software*, vol. 32, no. 4, pp. 635–653, December 2006.
- [28] B. W. Bader *et al.* (2015) Matlab tensor toolbox version 2.6. [Online]. Available: <http://www.sandia.gov/~tgkolda/TensorToolbox/>
- [29] BCI Competition III, Data set II: P300 speller paradigm. [Online]. Available: <http://www.bbci.de/competition/iii/>
- [30] N. R. Draper and H. Smith, *Applied regression analysis*, 3rd ed. John Wiley & Sons, 2014.

- [31] D. J. Krusienski *et al.*, “Toward enhanced p300 speller performance,” *Journal of neuroscience methods*, vol. 167, no. 1, pp. 15–21, 2008.
- [32] M. Salvaris and F. Sepulveda, “Wavelets and ensemble of flds for p300 classification,” in *4th International IEEE/EMBS Conference on Neural Engineering, 2009. NER’09*. IEEE, 2009, pp. 339–342.
- [33] H. Cecotti and A. Gräser, “Convolutional neural networks for p300 detection with application to brain-computer interfaces,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 433–445, 2011.
- [34] J. Möcks, “Decomposing event-related potentials: a new topographic components model,” *Biological Psychology*, vol. 26, no. 1, pp. 199–215, 1988.
- [35] J. Moecks, “Topographic components model for event-related potentials and some bio-physical considerations,” *IEEE transactions on biomedical engineering*, vol. 6, no. 35, pp. 482–484, 1988.
- [36] A. S. Field and D. Graupe, “Topographic component (parallel factor) analysis of multi-channel evoked potentials: practical issues in trilinear spatiotemporal decomposition,” *Brain Topography*, vol. 3, no. 4, pp. 407–423, 1991.
- [37] F. Miwakeichi *et al.*, “Decomposing eeg data into space–time–frequency components using parallel factor analysis,” *NeuroImage*, vol. 22, no. 3, pp. 1035–1045, 2004.
- [38] C. S. Herrmann *et al.*, “Memory-matches evoke human gamma-responses,” *BMC neuroscience*, vol. 5, no. 1, p. 13, 2004.
- [39] M. Niknazar *et al.*, “Blind source separation of underdetermined mixtures of event-related sources,” *Signal Processing*, vol. 101, pp. 52–64, 2014.
- [40] F. Cong *et al.*, “Multi-domain feature of event-related potential extracted by nonnegative tensor factorization: 5 vs. 14 electrodes eeg data,” in *Latent Variable Analysis and Signal Separation*. Springer, 2012, pp. 502–510.
- [41] N. Novitskiy *et al.*, “The bold correlates of the visual p1 and n1 in single-trial analysis of simultaneous eeg-fmri recordings during a spatial detection task,” *Neuroimage*, vol. 54, no. 2, pp. 824–835, 2011.
- [42] F. Cong *et al.*, “Classifying healthy children and children with attention deficit through features derived from sparse and nonnegative tensor factorization using event-related potential,” in *Latent Variable Analysis and Signal Separation*. Springer, 2010, pp. 620–628.
- [43] F. Cong *et al.*, “Feature extraction by nonnegative tucker decomposition from eeg data including testing and training observations,” in *Neural Information Processing*. Springer, 2012, pp. 166–173.
- [44] A. Onishi *et al.*, “Tensor classification for p300-based brain computer interface,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012*. IEEE, 2012, pp. 581–584.

- [45] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.
- [46] L. N. Trefethen and D. Bau III, *Numerical linear algebra*. Siam, 1997, vol. 50.
- [47] E. Donchin *et al.*, “The mental prosthesis: assessing the speed of a p300-based brain-computer interface,” *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, pp. 174–179, 2000.
- [48] L. A. Farwell and E. Donchin, “Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials,” *Electroencephalography and clinical Neurophysiology*, vol. 70, no. 6, pp. 510–523, 1988.
- [49] S. Theodoridis and K. Koutroumbas, *Pattern recognition*. Academic press, 2003.
- [50] P. J. Kindermans *et al.*, “A bayesian model for exploiting application constraints to enable unsupervised training of a p300-based bci,” *PloS one*, vol. 7, no. 4, p. e33758, 2012.
- [51] C. M. Bishop, *Pattern recognition and Machine Learning*. Springer, 2006.
- [52] BCI Competition III, Data set II: P300 speller paradigm. [Online]. Available: <http://www.bbci.de/competition/ii/>

واژه‌نامه فارسی به انگلیسی

ا

Cross Validation اعتبارسنجی متقابل

Discriminant Analysis آنالیز تفکیکی

ب

fit برازش

پ

Event Related Potential پتانسیل وابسته به رخداد

ت

Trial تکثیت

Iterative تکراری

Repetition تکرار

ر

Brain Computer Interface رابط مغز-رایانه

Rank رتبه

Rank-1 رتبه-۱

ز

Subspace زیرفضا

چ

Multilinear چند-خطی

م

FactorMatrix ماتریس عامل

BasisMatrix ماتریس پایه

Dataset مجموعه داده

ABSTRACT

Tensors are valuable tools to represent EEG data. Tucker decomposition is the most used tensor decomposition in multidimensional discriminant analysis and tensor extension of LDA (Higher Order Discriminant Analysis-HODA) is a popular tensor discriminant method used for data of ERP-based BCIs. In this Thesis we introduce a new tensor-based feature reduction technique, named Higher Order Spectral Regression Discriminant Analysis (HOSRDA), with application in P300-based BCIs. The proposed method (HOSRDA) is a tensor extension of Spectral Regression Discriminant Analysis (SRDA) and casts the eigenproblem of HODA to a regression problem and therefore overcome the probable issue of singularity of scatter matrices. Also, the formulation of HOSRDA can provide the ability of adding regularization constraints to the factor matrices of the discriminant subspace. Additionally, when the dimension and number of data are very large, the regression problem can be solved via efficient iterative algorithms. We applied HOSRDA on data of a P300 speller from BCI competition III and reached average character detection accuracy 96.5%, 89%, and 72.5% for the two subjects, when 15, 10, and 5 repetitions are used for each intensification. HOSRDA has proved its high-accuracy performance in P300-based BCI and outperforms almost all the previous methods on the used dataset. HOSRDA, as a new multiway feature reduction technique, has a high performance for usage in P300-based BCI and can open a new framework for adding different regularization constraints in higher order feature reduction problem.

On the other side of the thesis, we have proposed the tensorized formulation of unsupervised classification in P300 speller and discussed the reasons of why it cannot perform at least the same quality of the vector-based formulation.

KEYWORDS

1. Event Related Potential
2. Tensor Decomposition
3. Tucker Decomposition
4. Higher Order Spectral Regression Discriminant Analysis (HOSRDA)
5. Higher Order Discriminant Analysis (HODA)
6. Spatial-Temporal Discriminant Analysis (STDA)



**SHARIF UNIVERSITY OF TECHNOLOGY
ELECTRICAL ENGINEERING DEPARTMENT**

M.Sc. THESIS

Title:

**Detection of Event Related Potentials
using Tensor Decomposition**

by:

Mina Jamshidi Idaji

Supervisor:

Prof. Mohammad B. Shamsollahi

August 2016