



تشخیص زودهنگام نیاز بیمار به مراقبت‌های ویژه

کیان باختری

دانشجوی کارشناسی مهندسی کامپیوتر دانشگاه صنعتی شریف
bakhtari.kian@gmail.com

۱۴۰۰ تیر

چکیده: هنگام اوج گیری نرخ ابتلا به بیماری‌های مانند کووید ۱۹، بخش مراقبت‌های ویژه (ICU) سرریز از بیماران بدخل شده و شیوه‌ی درست یا نادرست مدیریت منابع در یک بیمارستان می‌تواند در هر روز به چندین بیمار زندگی دوباره ببخشد یا جان آن‌ها را بگیرد. مستله‌ی پیش‌بینی نیاز بیماران به مراقبت‌های ویژه، جهت بهینه‌سازی مدیریت منابع مطرح می‌شود. در این گزارش به شرح چگونگی حل این مستله با استفاده از یادگیری ماشین و **مجموعه دادگان منتشر شده از بیمارستان Sirio libanes** در شهر سانوپائولوی برزیل پرداخته شده و فرایند طی شده از تحلیل اکتشافی داده‌ها تا انتخاب و نتایج مدل نهایی بررسی می‌شود. این مجموعه‌ی داده شامل اطلاعات و نتایج آزمایش‌های پزشکی ۳۸۵ بیمار مبتلا به کووید ۱۹ است که با نیاز یا عدم نیاز بیمار به مراقبت‌های ویژه برچسب‌گذاری شده است. نتایج نهایی برای حالتی که داده‌های یادگیری و آزمون بر اساس بیمار جدا شده باشند، F1 score ۸۰ و برای حالتی که بر اساس بیمار جدا نشده باشند ۹۸ به دست آمده است. برای مطالعه‌ی توضیحات در مورد این دوروش جداسازی، به بخش ۵ مراجعه کنید. در کنار این گزارش، یک نوت‌بوک ژوپیتر به زبان پایتون قرار دارد که برای هر بخش، هر عمل و علت آن، توضیحات لازم را در بر دارد. توصیه می‌شود برای دیدن کدها، جزئیات، و ترتیب دقیق پروشه‌های طی شده، این نوت‌بوک را مطالعه کنید.

۱ مقدمه
اکتشافی داده‌ها و مهندسی ویژگی‌ها را بررسی کرده و سپس به انتخاب و تست مدل‌های یادگیری ماشین خواهیم پرداخت.

۲ تحلیل اکتشافی داده

مجموعه‌ی دادگان خام که در اختیار داریم، بیش از ۲۰۰ ستون (ویژگی) و حدود ۲۰۰۰ سطر (سپل) دارد که اطلاعات جمعتی، علائم حیاتی و آزمایش خون ۳۸۵ بیمار مبتلا به کووید را در خود جای داده است. در بخش‌های پیش‌رو به شرح اعمال صورت گرفته برای پاکسازی و آماده‌سازی دادگان می‌پردازیم. در این پروژه از امکانات کتابخانه‌های Pandas، Numpy و Sklearn استفاده شده است.

۱-۲ آشنایی با ساختار مجموعه داده‌ها

منظور از ساختار مجموعه داده‌ها، مواردی از قبیل تعداد سطرها و ستون‌ها، چگونگی برچسب‌گذاری، شناخت ویژگی‌های محوری (مانند ستون‌های window و identifier)، تعداد سطرها به ازای هر بیمار و... می‌باشد.

بدون تردید، موفقیت دانشمندان علوم داده در حل مستله‌ای که در چکیده مطرح شد برای نظام سلامت و کادر درمان بسیار کارگشا است. اگر هرچه سریع‌تر بتوانند پیش‌بینی واقع‌گرایانه‌ای از وضعیت آینده‌ی بیمار به دست آورند، منابع محدود را بهتر مدیریت کرده و به انسان‌های بیشتری زندگی دوباره می‌بخشند. در این مجموعه‌ی داده به ازای هر بیمار، پنج نمونه (سطر) از اطلاعات پزشکی وجود دارد که در پنجره‌های زمانی متفاوت از بیمار گرفته شده‌اند. پنجره‌ی اول مربوط به دو ساعت ابتدایی پذیرش بیمار است. پنجره‌ی دوم مربوط به ساعات دوم تا چهارم، پنجره‌ی سوم ساعات چهارم تا ششم، پنجره‌ی چهارم ساعات ششم تا دوازدهم و پنجره‌ی پنجم مربوط به زمان‌های پس از ساعت دوازدهم می‌باشد. در هر پنجره توسط ستون ICU مشخص شده که بیمار در مراقبت‌های ویژه حضور دارد یا خیر. هرچه بتوانیم با استفاده از پنجره‌های زمانی ابتدایی تر سرنوشت بیمار را پیش‌بینی کنیم، کار پر ارزش‌تری صورت گرفته است؛ چرا که کادر درمان فرصت بیشتری برای مدیریت منابع در اختیار خواهد داشت. ابتدادرآیند تحلیل

در مراقبت‌های ویژه ندارند. اکنون می‌خواهیم هر بیمار را به صورت یک موجودیت واحد بینیمیم. یعنی یک بیمار مشخص در نهایت یا به مراقبت‌های ویژه نیاز پیدا کرده است یا نکرده است. این که در کدام پنجره‌ی زمانی از بخش به ICU منتقل شده است اهمیتی ندارد (حداقل برای هدف این بخش). به همین منظور، یک برچسب‌گذاری جدید به نام extended ICU انجام شد. بدین صورت که برای هر بیماری که در نهایت به مراقبت‌های ویژه منتقل شده بود، در هر پنج سطر آن بیمار، برچسب نیاز به مراقبت‌های ویژه زده شد. نمودار تعداد بیماران نیازمند مراقبت‌های ویژه بر اساس این که سن بیمار بالای ۶۵ سال است یا پایین ۶۵ سال را در شکل ۳ مشاهده می‌کنید. همین طور تعداد بیماران نیازمند به مراقبت‌های ویژه بر حسب دهک سنی و جنسیت به ترتیب در شکل‌های ۴ و ۵ قابل مشاهده هستند. این دانسته‌ی پیشین که کووید ۱۹

افرادی با سن بالاتر را بیشتر گرفتار می‌کند تایید می‌شود.

در ادامه، قصد لمس کردن داده‌هایی را داریم که کمی بیشتر به دنیای پزشکی مربوط هستند. انتظار داریم ضعیف بودن سیستم ایمنی بدن در چگونگی مقابله‌ی بیمار با ویروس کرونا تاثیرگذار باشد. برای بررسی این موضوع، نمودار تعداد افراد نیازمند مراقبت‌های ویژه بر حسب ویژگی Immunocompromised شکل ۶ رسم شده است. به نظر نمی‌رسد همبستگی معنی‌داری وجود داشته باشد که کمی تعجب برانگیز است اما حتماً دلیلی معتبر برای این موضوع در دنیای پزشکی وجود دارد.

در شکل ۷ نمودار تعداد بیماران نیازمند مراقبت‌های ویژه بر حسب ویژگی HTN یا همان فشار خون بالا نمایش داده شده است که نشان‌دهنده‌ی تاثیر این فاکتور بر وحامت اوضاع بیمار می‌باشد. همچنین نمودار توزیع میانگین نرخ تنفس در دقیقه برای دو گروه بیماران نیازمند به مراقبت‌های ویژه و سایرین در شکل ۸ ترسیم شده و نشان دهنده‌ی تاثیر این ویژگی بر نیاز نهایی بیمار است. در شکل ۹، توزیع بیشینه‌ی دمای بدن بیمار برای دو گروه مورد بررسی نمایش داده شده و نشان‌دهنده‌ی ارتباط نسی تب بیمار با نیاز آینده‌ی او به مراقبت‌های ویژه است. با توجه به شیوه‌ی عملکرد ویروس کرونا و تاثیری که بر سیستم تنفسی می‌گذارد، می‌توان انتظار داشت که میزان کسیزن خون بیمار از فاکتورهای تعیین‌کننده باشد. توزیع اکسیزن خون بیماران برای دو گروه مورد بررسی در شکل ۱۰ قابل مشاهده است. به نظر توزیع این فاکتور برای دو گروه تقریباً یکسان است. این موضوع به ما یادآوری می‌کند ویژگی‌های متفاوت حاضر در مجموعه‌ی دادگان به راحتی و بدون داشتن پزشکی قابل تفسیر نیستند و باید با بررسی‌های آماری بیشتر، مجموعه دادگان را برای استفاده در یادگیری ماشین آماده کرد.

جست‌وجوی خوش‌های احتمالی در مجموعه‌ی داده‌ها نیز باید صورت گیرد. ممکن است در فضای ویژگی‌ها، داده‌هایی که برچسب مراقبت ویژه دارند و آن‌هایی که ندارند به نحوی از هم جدا شده باشند. برای بررسی این موضوع از الگوریتم مصورسازی t-SNE استفاده شد که تلاش دارد تا همسایگی نقاط را حفظ کند. نتیجه‌ی اجرای این الگوریتم را برای پنج perplexity متفاوت در شکل ۱۱ مشاهده

مقادیر اولین ستون با نام index مطابق با Patient visit identifier بود پس حذف شده و برای راحتی، نام ستون PID تغییر پیدا کرد. از ترتیب صحیح و یکتاپی مقادیر این ستون اطمینان حاصل شد و تعداد پنجره‌های زمانی به ازای هر بیمار نیز بررسی شد که کم یا زیاد نباشد. همچنین ترتیب صحیح مقادیر در ستون dow بررسی شده و اطمینان حاصل شد که ستون‌هایی که معرف هویت بیمار، زمان داده‌گیری، و حضور در مراقبت‌های ویژه هستند، مقادیر گم شده (Null value) نداشته باشند.

۲-۲ شناسایی مقادیر گم شده، دیتاتایپ‌ها و پارامترهای آماری ویژگی‌ها

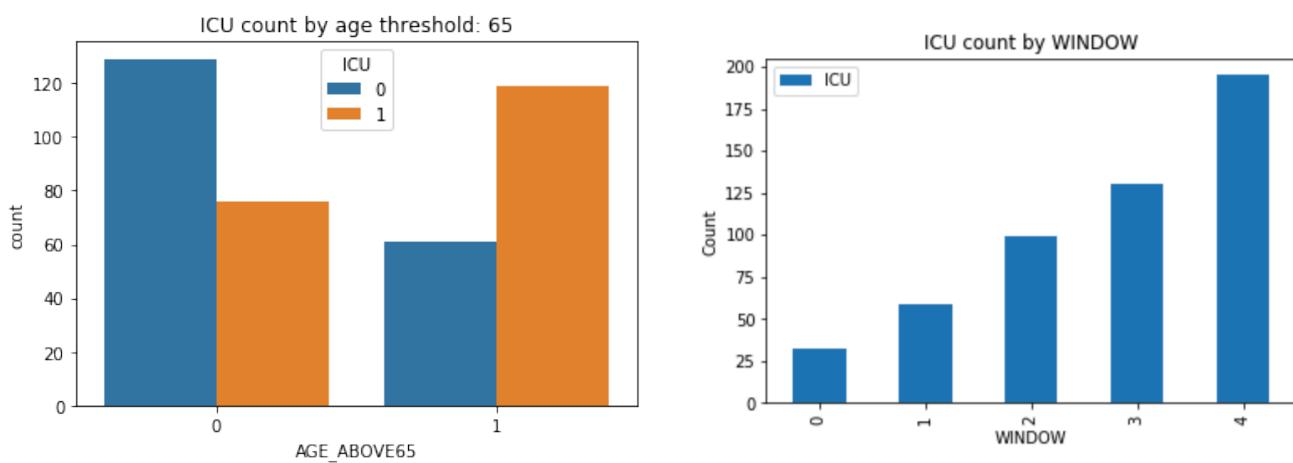
هدف اصلی این بخش، شناخت بهتر ویژگی‌هاست. برای به دست آورد یک دید کلی و مناسب، تابع get df columns info و استفاده شده و اطلاعاتی برای هر ویژگی نمایش داده شده است. اطلاعاتی مانند عددی یا غیر عددی بودن، مقادیر یکتا، درصد مقادیر ناموجود، کمینه، بیشینه، میانگین و واریانس. بخشی از خروجی این تابع را برای ستون اول در شکل ۱ مشاهده می‌کنید. برخی ستون‌ها واریانس صفر دارند و این یعنی مقادیرشان در سطرهای مختلف یکسان است. این ستون‌ها حذف شدند. ستون‌هایی که مقادیرشان سطربه سطربه یکسان بودند نیز شناسایی شده و فقط یکی‌شان در مجموعه نگه‌داری شد. در شکل ۱ مشخص است که بسیاری از ستون‌ها، مقادیر قابل توجهی داده‌ی ناموجود دارند. در واقع، بیش از ۷۰ درصد ویژگی‌ها، تعداد داده‌های ناموجودشان بیشتر از داده‌های موجود می‌باشد. در این بخش قصد جایگذاری این مقادیر را نداریم و همچنان در مرحله‌ی شناخت دادگان هستیم. ستون‌هایی که مقادیر غیر عددی دارند به عددی تبدیل شدند تا در بخش‌های بعدی بتوانیم اطلاعات آماری دادگان را بررسی کنیم؛ به عنوان مثال بتوانیم همبستگی (correlation) بین هر دو ستون دلخواه را بررسی کنیم (اگر داده‌های غیر عددی باشند این کار ممکن نیست). منتشرکنندگان این مجموعه داده، اظهار داشتند که داده‌ها بین ۱ و ۰-۱ مقیاس شده است (منبع). برای کسب اطمینان این موضوع نیز چک شده و اطمینان حاصل شد.

۳-۲ بررسی مقادیر، همبستگی‌ها و مصوروسازی

اکنون بناست بیشتر به معانی داده‌ها توجه شود. چگونگی تاثیر ویژگی‌های ملموس‌تر مانند داده‌های جمعیتی با نیاز به مراقبت‌های ویژه مطابقت داده شده و شهود ابتدایی از حال و هوای پزشکی و کاری که پیش رو است به دست آید. نیازی نیست پزشک باشیم تا حدس بزنیم احتمالاً هرچه در پنجره‌ی زمانی به جلو حرکت می‌کنیم، تعداد بیمارانی که به بخش مراقبت‌های ویژه منتقل می‌شوند نیز افزایش می‌یابد. نمودار تعداد بیماران نیازمند به مراقبت‌های ویژه بر اساس پنجره‌ی زمانی در شکل ۲ نمایش داده شده است. برای ادامه‌ی فرایند تحلیل داده و مصوروسازی، باید توجه شود بیمارانی که در نهایت به مراقبت‌های ویژه منتقل شده‌اند، در پنجره‌های زمانی ابتدایی تر برچسب حضور

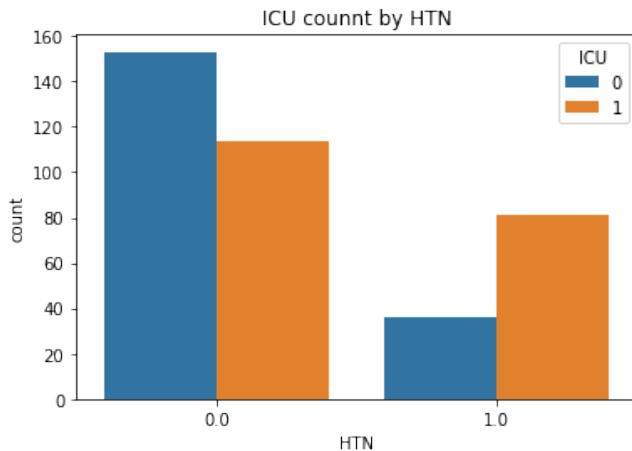
Column name	Column Pandas-type	Column Python-type	Numeric/Non-numeric	% of null values	Unique values count	Min	Max	Mean	Variance	
0	PID	int64	int	Numeric	0.00	---	0	384	192.0	12358.42
1	AGE_ABOVE65	int64	int	Numeric	0.00	---	0	1	0.47	0.25
2	AGE_PERCENTIL	object	str or mixed	Non-numeric	0.00	10	---	---	---	---
3	GENDER	int64	int	Numeric	0.00	---	0	1	0.37	0.23
4	HTN	float64	float	Numeric	0.26	---	0.0	1.0	0.21	0.17
5	IMMUNOCOMPROMISED	float64	float	Numeric	0.26	---	0.0	1.0	0.16	0.13
6	OTHER	float64	float	Numeric	0.26	---	0.0	1.0	0.81	0.15
7	ALBUMIN_MEDIAN	float64	float	Numeric	57.35	---	-1.0	1.0	0.53	0.05
8	ALBUMIN_MEAN	float64	float	Numeric	57.35	---	-1.0	1.0	0.53	0.05
9	ALBUMIN_MIN	float64	float	Numeric	57.35	---	-1.0	1.0	0.53	0.05
10	ALBUMIN_MAX	float64	float	Numeric	57.35	---	-1.0	1.0	0.53	0.05
11	ALBUMIN_DIFF	float64	float	Numeric	57.35	---	-1.0	-1.0	-1.0	0.0
12	BE_ARTERIAL_MEDIAN	float64	float	Numeric	57.35	---	-1.0	1.0	-0.96	0.03
13	BE_ARTERIAL_MEAN	float64	float	Numeric	57.35	---	-1.0	1.0	-0.96	0.03
14	BE_ARTERIAL_MIN	float64	float	Numeric	57.35	---	-1.0	1.0	-0.96	0.03
15	BE_ARTERIAL_MAX	float64	float	Numeric	57.35	---	-1.0	1.0	-0.96	0.03
16	BE_ARTERIAL_DIFF	float64	float	Numeric	57.35	---	-1.0	-1.0	-1.0	0.0
17	BE_VENOUS_MEDIAN	float64	float	Numeric	57.35	---	-1.0	1.0	-0.93	0.03
18	BE_VENOUS_MEAN	float64	float	Numeric	57.35	---	-1.0	1.0	-0.93	0.03
19	BE_VENOUS_MIN	float64	float	Numeric	57.35	---	-1.0	1.0	-0.93	0.03
20	BE_VENOUS_MAX	float64	float	Numeric	57.35	---	-1.0	1.0	-0.93	0.03
21	BE_VENOUS_DIFF	float64	float	Numeric	57.35	---	-1.0	-1.0	-1.0	0.0
22	BIC_ARTERIAL_MEDIAN	float64	float	Numeric	57.35	---	-1.0	1.0	-0.31	0.01

شکل ۱: اطلاعات مقدماتی آماری از ۲۲ ستون ابتدایی مجموعه دادگان

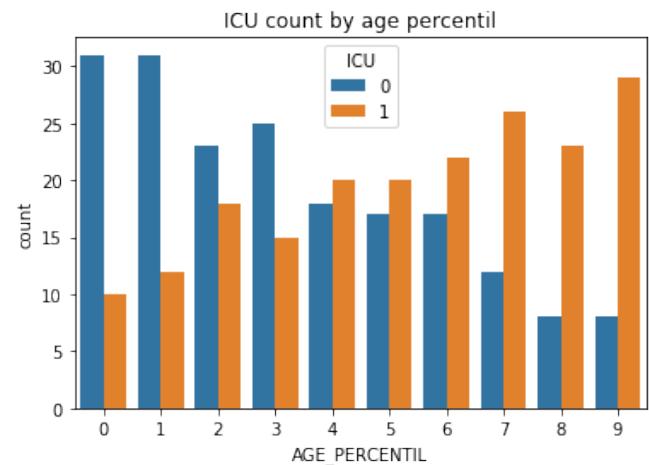


شکل ۳: تعداد بیماران نیازمند مراقبت‌های ویژه بر حسب پنجره‌ی زمانی پایین‌تر بودن سن‌شان از ۶۵ سال

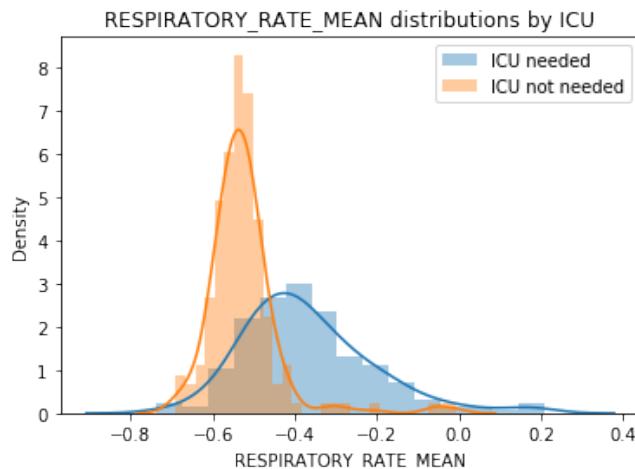
شکل ۲: تعداد بیماران نیازمند به مراقبت‌های ویژه بر حسب پنجره‌ی زمانی



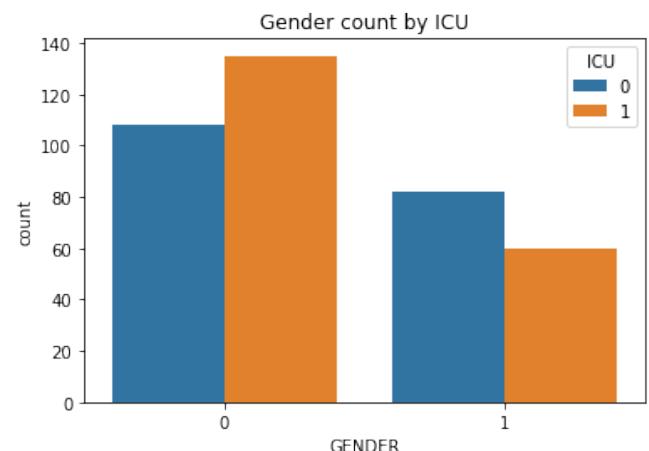
شکل ۷: تعداد بیماران نیازمند مراقبت‌های ویژه بر حسب بالا یا نرمال بودن فشار خون



شکل ۴: تعداد بیماران نیازمند مراقبت‌های ویژه بر حسب دهک سنی



شکل ۸: توزیع میانگین نرخ تنفس در دقیقه بر حسب نیاز یا عدم نیاز به مراقبت‌های ویژه

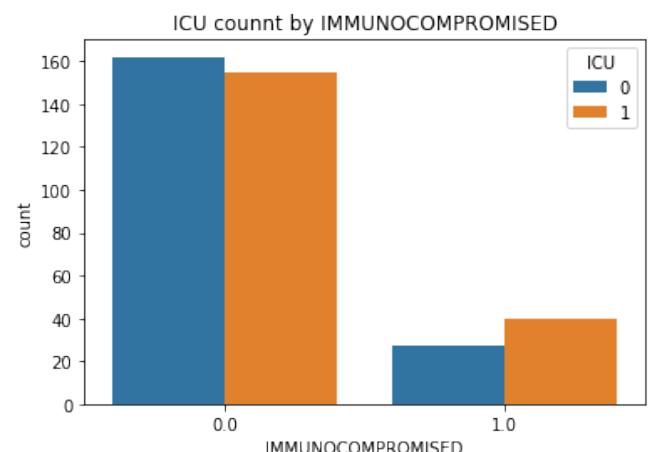


شکل ۵: تعداد بیماران نیازمند مراقبت‌های ویژه بر حسب جنسیت

می‌کنید. این الگوریتم پس از جایگذاری مقادیر ناموجود اجرا شده است. نتایج به دست آمده نشان‌دهنده‌ی هیچ تجمعی بر حسب برچسب هدف نیست و اطلاعات جدید و ارزشمندی به دست ما نداده است.

۴-۲ آماده‌سازی برای مهندسی ویژگی‌ها، جایگذاری مقادیر ناموجود، پالایش داده‌های پرت

پیش‌تر ذکر شد که بیش از ۷۰ درصد ستون‌های مجموعه بیش از ۵۰ درصد مقادیرشان ناموجود است. کسر بزرگی به نظر می‌رسد و اگر با مجموعه‌ی دادگان دیگری سروکار داشتیم که همین مقدار داده‌های ناموجود دارد، باید سیاست‌های متفاوتی را برای جایگذاری این مقادیر در نظر گرفته و بر اساس بررسی‌ها یک یا چند مورد از این سیاست‌ها را اعمال می‌کردیم. علاوه‌بر سیاست‌های جایگذاری مقادیر، سیاست‌هایی چون حذف ستون‌هایی که کسر زیادی شان مقادیر ناموجود است نیز در نظر گرفته می‌شوند. اما در این مجموعه داده که

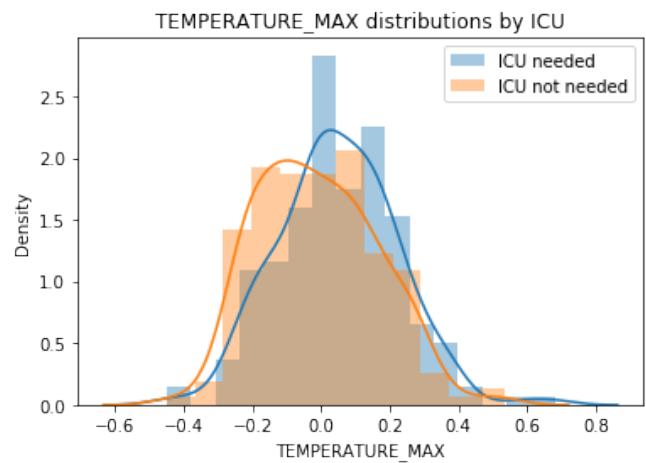


شکل ۶: تعداد بیماران نیازمند مراقبت‌های ویژه بر حسب نرمال یا ضعیف بودن سیستم ایمنی بدن

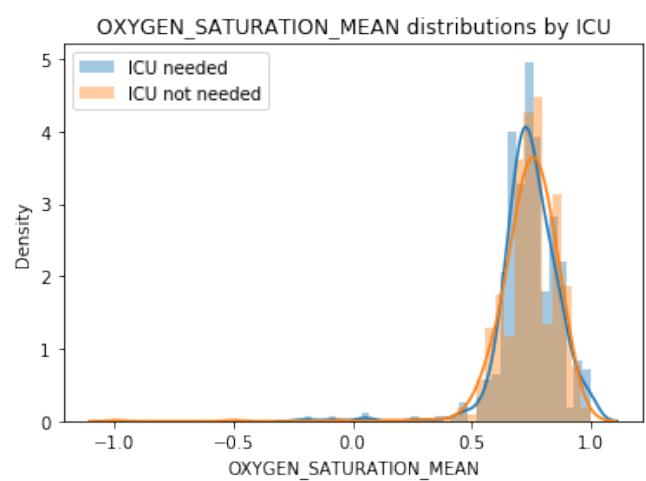
در اختیار داریم، سیاست جایگذاری مقادیر ناموجود از پیش توسط منتشرکنندگان مجموعه داده **توصیه شده است**. آن‌ها اظهار داشته‌اند که وقتی مقادیر یک ویژگی برای یک بیمار در چند پنجره‌ی زمانی ناموجود است، به آن معناست که بیمار در آن پنجره از نظر حیاتی با ثبات بوده و آزمایش مجدد از او گرفته نشده است. همچنین اظهار داشته‌اند که از نظر پژوهشکی منطقی است که تصویر کنیم مقادیر این ویژگی‌ها در این پنجره‌های زمانی، با پنجره‌های قبلی همان بیمار برابر است. پس، داده‌های ناموجود در هر ویژگی و بیمار، با داده‌ی پنجره‌ی قبلی همان بیمار در همان ویژگی جایگزین می‌شود. اگر پس از صورت گرفتن این جایگذاری، مشاهده شد که در اولین پنجره مقادیر ناموجودی وجود دارد، از متود `bfill` استفاده شده و آن مقدار را با مقادیر بعدی همان ویژگی و همان بیمار جایگزین می‌کنیم. بدین ترتیب، داده‌های ناموجود جایگذاری می‌شوند (البته هنگام مهندسی ویژگی‌ها، یک سیاست دیگر نیز آزموده خواهد شد).

پس از مقداردهی داده‌های ناموجود، مشاهده می‌شود که همچنان تعدادی خانه با مقادیر گم شده در مجموعه دادگان وجود دارند. با بررسی‌های بیشتر، متوجه می‌شویم برای بیمار شماره‌ی ۱۹۹ تمامی ویژگی‌ها ناموجود است و به همین دلیل با متود شرح داده شده در پاراگراف بالا، مقادیرش جایگزین نشده‌اند. این بیمار را از مجموعه `tags` دادگان حذف کرده و مجموعه مجدداً `index` می‌شود. ویژگی `tags` بیشترین میزان داده‌های ناموجود را داشته و حتی پس از انجام جایگزینی با متودهای `ffill` و `bfill` همچنان `ffill` درصد بیماران مقداری در این ستون ندارند. حذف این ستون منطقی به نظر می‌رسد، اما پیش از این کار موظف هستیم همبستگی این ویژگی با ستون هدف (ICU) را بررسی کنیم. چرا که علی‌رغم درصد بالای داده‌های ناموجود، شاید اطلاعات و روابط مهمی بین این ویژگی و ستون هدف وجود داشته باشد. به همین منظور، نمودار تعداد بیماران با `tag`‌های متفاوت بر حسب حضور یا عدم حضور در مراقبت‌های ویژه ترسیم شده و در شکل ۱۲ نمایش داده شده است. در ظاهر از این نمودار می‌توان برداشت کرد که همبستگی آنچنان معنی‌داری وجود ندارد و با توجه به این که بیش از ۶۰ درصد داده‌های این ستون ناموجود است، می‌توان برای حذف این ستون اقدام کرد. اما برای کسب اطمینان بیشتر، همبستگی مورد بحث به صورت عددی نیز محاسبه شده و برابر با ۰.۰۶ به دست آمد که تایید کننده نتیجه‌گیری پیشین است. این ستون کنارگذاشته شد. در ادامه‌ی بررسی‌ها متوجه شدیم که بیمار شماره‌ی ۲۸۶ هم در بیش از ۱۴۰ ویژگی، فقط مقادیر ناموجود دارد. این بیمار را نیز از داده‌ها حذف کردیم.

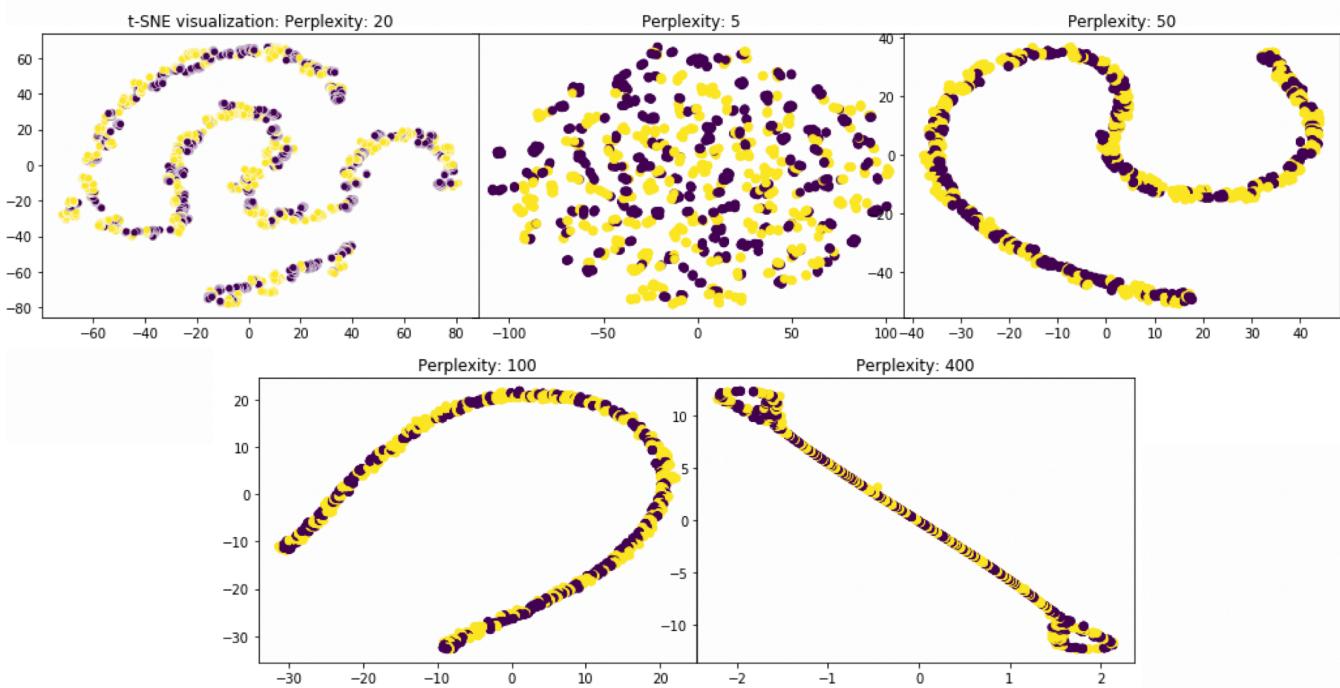
در این مرحله به پالایش داده‌های پرت پرداخته شد. چالشی که با آن روبرو هستیم آن است که بدون حضور و مشورت متخصصان (پژوهشکان) تعیین کردن چیستی داده‌ی پرت کار مشکلی است. به لحاظ آماری معیارها و تعاریف متفاوتی برای داده‌ی پرت وجود دارد اما چیزی که ما اینجا به عنوان داده‌ی پرت تلقی می‌کنیم ممکن است واقعاً از نظر پژوهشکی داده‌ی پرت نبوده و اتفاقاً نشان‌دهنده‌ی یک ویژگی مهم برای



شکل ۹: توزیع بیشینه‌ی دمای بدن بیمار بر حسب نیاز بیمار به مراقبت‌های ویژه



شکل ۱۰: توزیع میزان اکسیژن خون بیمار بر حسب نیاز یا عدم نیاز بیمار به مراقبت‌های ویژه



شکل ۱۱: نتیجه‌های مصورسازی با الگوریتم t-SNE برای پنج پارامتر perplexity متفاوت

۱-۳ همبستگی‌ها

ابتدا برای به دست آوردن شهود کلی، نقشه‌ی حرارتی همبستگی (correlation heatmap) برای تمامی ویژگی‌ها رسم شده شد و در شکل ۱۶ نمایش داده شده است. در شکل ۱۷ نیز، همبستگی ویژگی‌های مجموعه با ستون هدف به شکل نقشه‌ی حرارتی و مرتب شده از بیشترین به کمترین رسم شده است. مشاهده می‌شود که بیشترین همبستگی نزدیک به ۰.۴ است، در همبستگی‌های منفی نیز به -۰.۴ نزدیک می‌شویم و سایر ویژگی‌ها در این میان قرار گرفته‌اند. در نوت‌بوک پروره، حالت دیگری از همین نقشه رسم شده است که مجموعه دادگان آن شامل داده‌های پرت هم می‌شود و نتیجه‌ی آن تفاوت معنی‌داری با نقشه‌ی صفحه‌ی قبل ندارد. جهت جلوگیری از شلوغی بیش از حد و از دست رفتن انسجام، از نمایش این نقشه در این گزارش پرهیز شده است. جهت تمایل می‌توانید به نوت‌بوک مراجعه کنید.

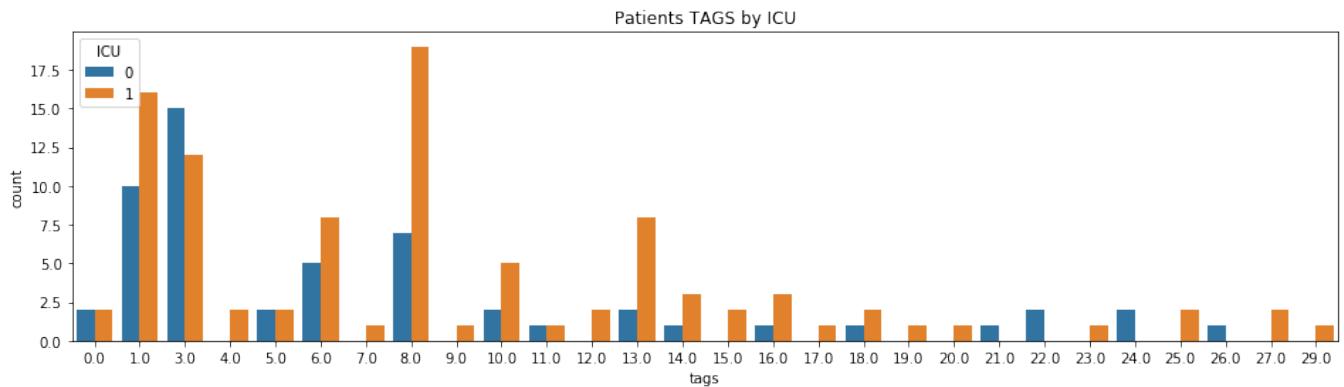
۲-۳ کاهش بعد

در مجموعه دادگانی که به عنوان ورودی به الگوریتم‌های یادگیری ماشین تحویل می‌شوند، حضور ستون‌هایی که واقعاً همبستگی معنی‌داری با ستون هدف ندارند یا ترکیب‌های خطی از ویژگی‌های دیگر هستند، به احتمالی موجب کاهش دقت و کیفیت یادگیری می‌شود. به همین منظور، باید بررسی شود که آیا حذف این گونه ویژگی‌ها از مجموعه دادگان باعث افزایش کیفیت یادگیری می‌شود یا خیر. به طور معمول ممکن است در حین پاکسازی داده‌ها، چند ویژگی به علت وجود بیش از حد مقادیر ناموجود حذف شوند، مانند ویژگی tags که پیش‌تر با آن روبرو شدیم و پس از بررسی بیشتر آن را کنار گذاشتیم. اما تا به این

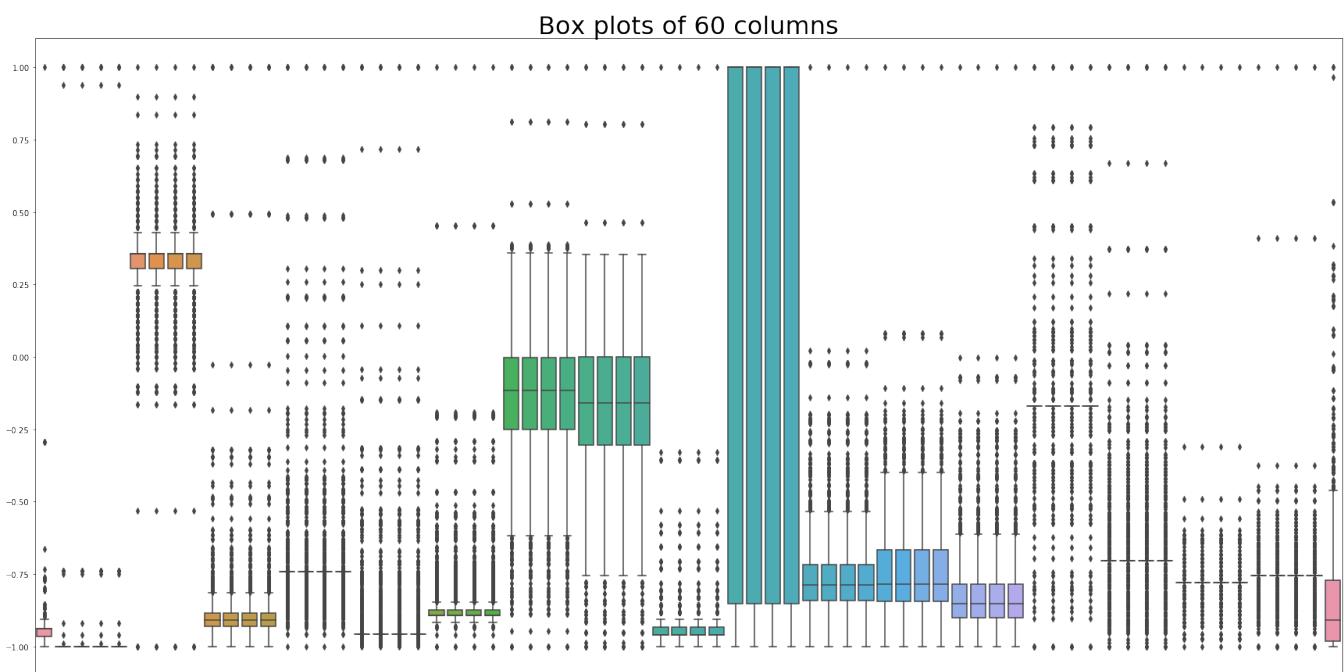
بیمار باشد. با این تفاصیل، سادگی پارامتر انحراف از میانگین باعث شد تا به عنوان معیار برای شناسایی داده‌ی پرت انتخاب شود؛ یعنی اگر فاصله‌ی داده‌ای از میانگین آن ویژگی بیش از ضریب مشخصی از انحراف از میانگین بود، آن داده پرت تلقی شود. پیش از انتخاب این ضریب، نگاهی به نمودار جعبه‌ای ۶۰ ستون (از ستون‌های میانی) مجموعه بیاندازیم (شکل ۱۳). محزز است که داده‌های پرت وجود دارند. این نمودار را چندین بار با استفاده از ضرایب مختلف برای فیلتر کردن داده‌های پرت پالایش کردیم تا در نهایت ضریب ۲.۵ انتخاب شد. البته که تفاوت آنچنان معنی‌داری میان این ضریب و برای مثال ۲ یا ۳ وجود ندارد. در شکل ۱۴ همین نمودار را پس از پالایش داده‌های پرت مشاهده می‌کنید و در شکل ۱۵ همان نمودار پالایش شده، منتها این بار مقیاس شده قابل مشاهده است. تفاوت شکل ۱۳ با ۱۵ نشان می‌دهد که پالایش داده‌های پرت باعث گستردگی بیشتر توزیع ویژگی‌ها شده است. لازم به ذکر است که با توجه به مواردی که در مورد چیزی داده‌ی پرت ذکر شد، مجموعه دادگان پس از پالایش داده‌های پرت در یک مجموعه‌ی جدید نگهداری شده و از حالا به بعد، تمامی بررسی‌ها بر روی هر دو مجموعه داده انجام خواهند شد.

۳ مهندسی ویژگی‌ها

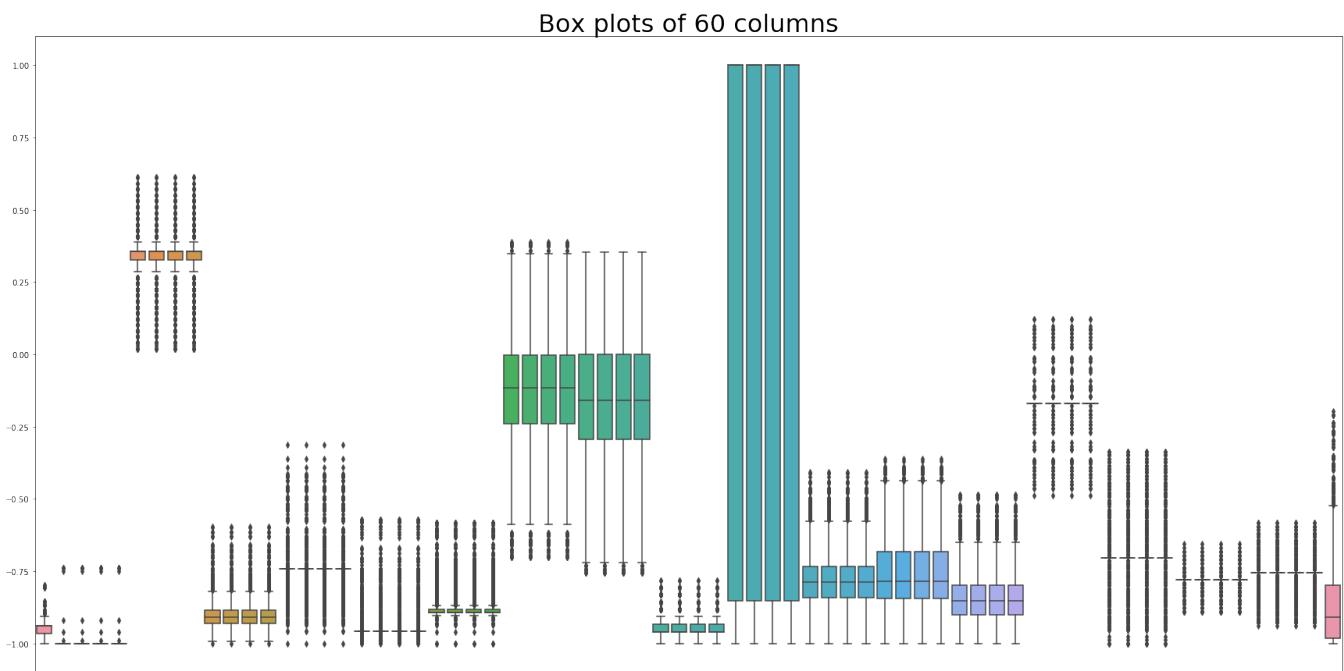
این بخش با بررسی همبستگی ویژگی‌ها به شکل عمومی تر و مفصل‌تر از قبل آغاز می‌شود.



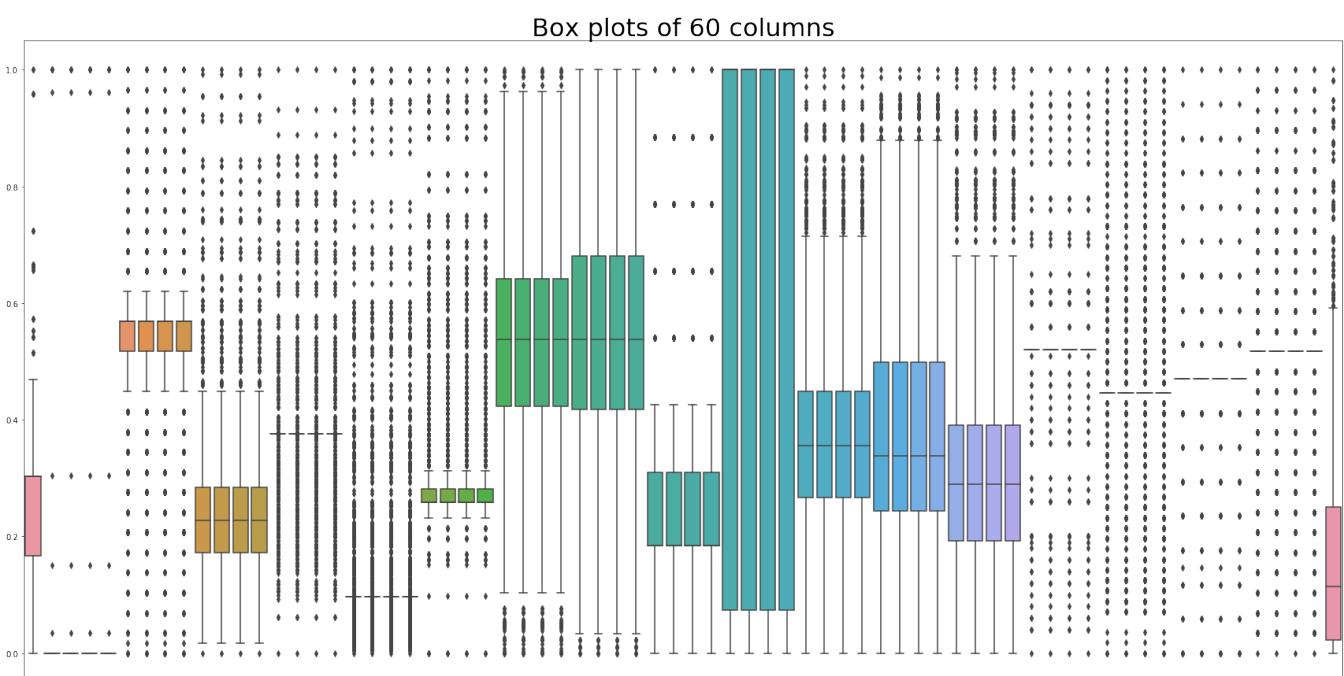
شکل ۱۲: تعداد بیماران نیازمند مراقبت‌های ویژه بر حسب ویژگی TAGS



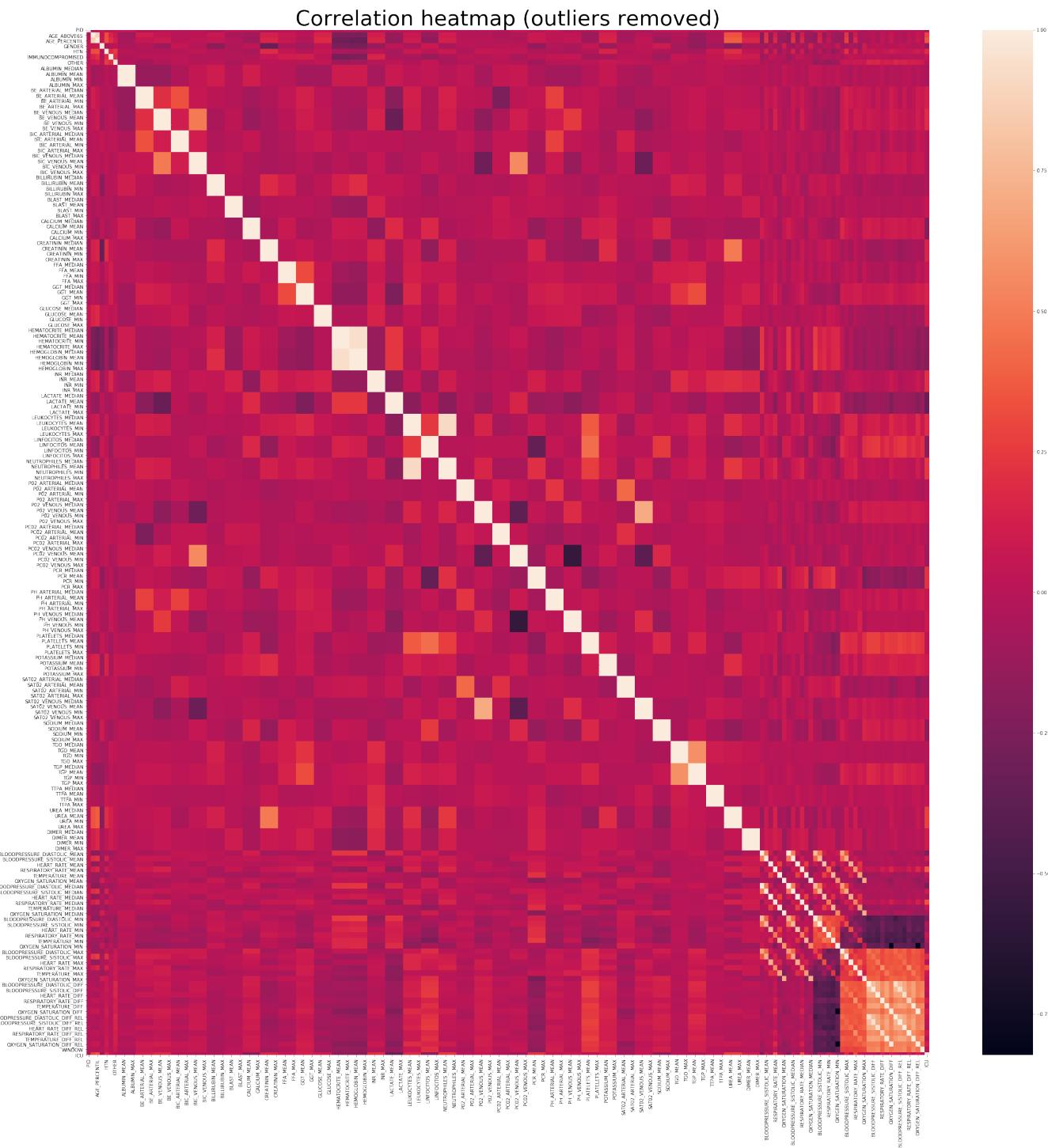
شکل ۱۳: نمودار جعبه‌ای ۶۰ ستون میانی مجموعه داده‌ها به منظور نمایش حضور داده‌های پرت



شکل ۱۴: نمودار جعبه‌ای ۶۰ ستون میانی مجموعه داده‌ها پس از پالایش داده‌های پرت

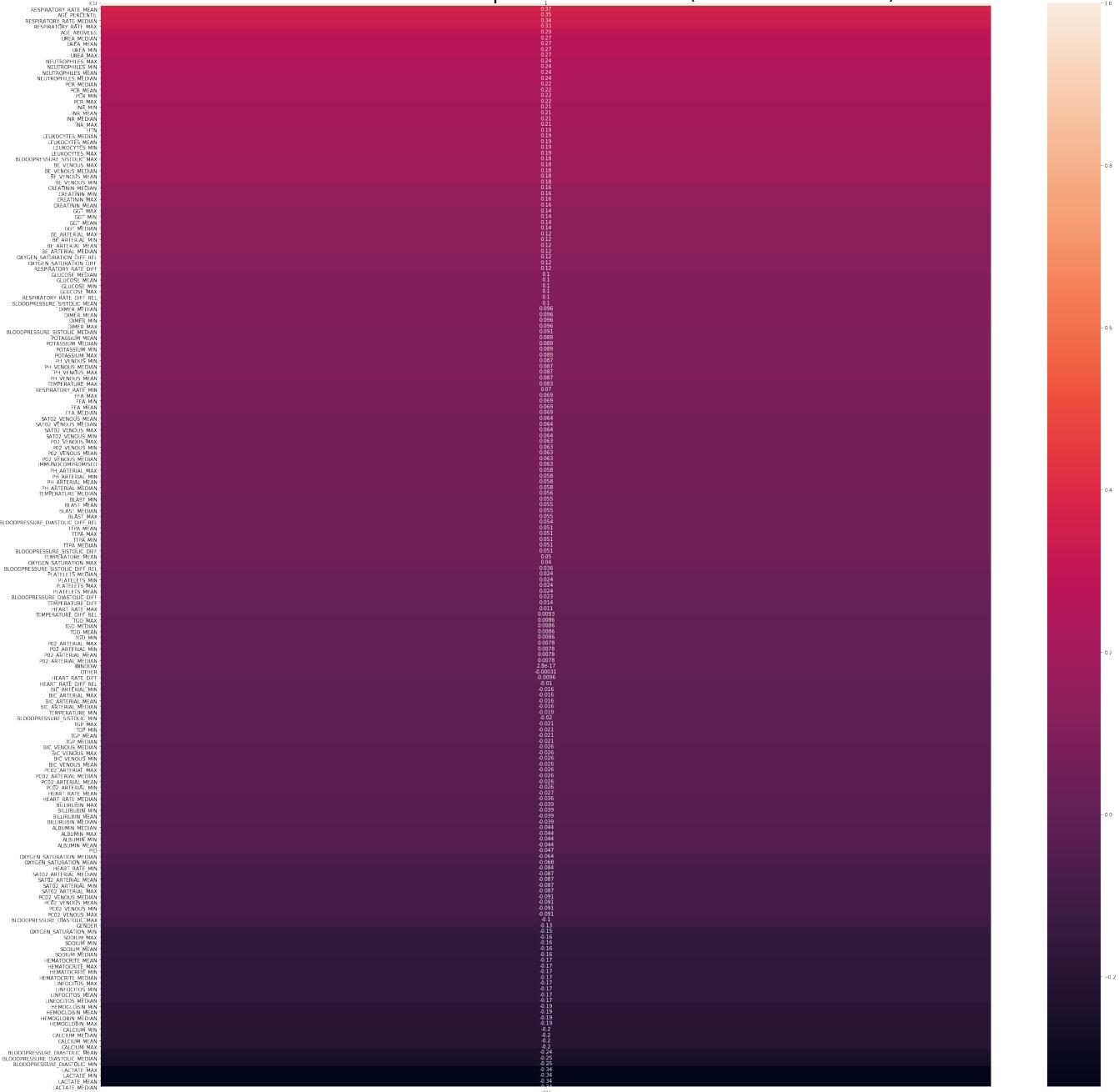


شکل ۱۵: نمودار جعبه‌ای ۶۰ ستون میانی مجموعه داده‌ها پس از پالایش داده‌های پرت و مقیاس شدن مجدد بین ۰ و ۱



شکل ۱۶: نقشه‌ی حرارتی همبستگی (correlation heatmap) برای تمامی ویژگی‌ها

Sorted correlation heatmap for ICU column (outliers removed)



شکل ۱۷: همبستگی (correlation) ویژگی‌های مجموعه با ستون هدف به شکل نقشه‌ی حرارتی و مرتب شده از بیشترین به کمترین

۴ مهندسی ویژگی‌ها: تست و بررسی مقدماتی با چند مدل

تعداد پنج مجموعه‌ی داده در دست داریم و هدف این است که بررسی شود کدام این‌ها برای فرایند اصلی مناسب‌تر است و مدل‌ها بیشتر از کدام ویژگی‌ها استفاده می‌کنند. در این بخش، در تمامی مدل‌هایی که train و test train می‌شوند، جدایی داده‌های train و test بر اساس بیمار یا پنجه‌ی زمانی نبوده و کل مجموعه‌ی داده‌ی مورد استفاده توسطتابع train split در کتابخانه‌ی sklearn جذا شده‌اند. لطفاً برای توضیحات test در مراحل بعدی مجموعه‌ی جداسازی داده‌های یادگیری و آزمون به ابتدای بیشتر در مورد شیوه‌ی جداسازی داده‌های یادگیری و آزمون به ابتدای ۵ مراجعه کنید. مقصود این است که با چند مدل معروف این پنج مجموعه‌ی داده‌ای که در اختیار داریم را بی‌آزماییم و پیدا کنیم که آیا عملکرد مدل‌ها با مجموعه‌های کاهش بعد داده شده بهتر است؟ آیا حضور یا عدم حضور داده‌های پرت در کیفیت یادگیری مدل‌ها تاثیر دارد؟ آیا حذف ویژگی‌هایی که خود مدل آن‌ها را مهم ندانسته، به بهبود نتایج کمک می‌کند؟ تمامی آزمایش‌ها ده بار تکرار شده و نتایج شان قابل مشاهده است.

۱-۴ درخت تصمیم (Decision tree)

مجموعه‌ی داده‌ای که کاهش بعد داده نشده و پالایش داده‌های پرت روی آن صورت گرفته است، در نوت‌بوک پروژه ۲ نام دارد. در این گزارش نیز هنگام ارجاع دادن به این مجموعه، از همین نام استفاده می‌شود. یک درخت تصمیم با این مجموعه داده ۱۰ مرتبه train شده و نتایج آن در شکل ۱۹ به نمایش درآمده است. نمودار میزان اهمیت ویژگی‌های مختلف (Feature importance) را برای این مدل در شکل ۲۰ مشاهده می‌کنید. همبستگی تعدادی از ویژگی‌هایی که پر اهمیت شمرده شده‌اند با ستون هدف بررسی شده و از طریق خروجی متود featureImportance متوسط شدیم که مدل حدود ۱۰۰ ویژگی را استفاده می‌کند. آزمودیم که آیا حذف ویژگی‌هایی که مدل آن‌ها را بی‌اهمیت انگاشته به بهبود نتایج کمک می‌کند یا خیر. همان‌طور که نتایج این آزمایش در شکل ۲۱ قابل مشاهده است، پیشرفتی حاصل نشده است. سپس، مجموعه‌ی کاهش بعد داده شده (data2pca) آزموده شد که نتایج آن در شکل ۲۲ آورده شده است. این مجموعه عملکرد بهتری نداشت و نتایج کم ارزش‌تری به دست داد. در گام بعد، مجموعه‌های شامل داده‌های پرت را آزمودیم. هر دو مجموعه عملکردی بسیار نزدیک به مجموعه‌ی هم بعد خودشان داشتند (یعنی شکل‌های ۱۹ و ۲۲) که یعنی حضور یا عدم حضور داده‌های پرت تاثیر چندانی بر کیفیت یادگیری مدل نگذاشته است. نتایج دقیق این آزمون‌ها در نوت‌بوک پروژه موجود است اما با توجه به توضیحات ارائه شده در این گزارش به نمایش در نیامده‌اند.

در قدم بعد آزموده شد که اجرای PCA با حفظ میزان بیشتری از واریانس (این بار ۹۹.۹ درصد) چه تاثیری بر نتایج می‌گذارد. تعداد ستون‌ها از ۱۸۹ به ۷۰ کاهش یافته و امتیاز F1 از ۸۰ به ۸۲ افزایش یافته که پیشرفت پر اهمیتی شمرده نمی‌شود. جدول نتایج دقیق در نوت‌بوک

مرحله ویژگی دیگری را حذف نکرده‌ایم چرا که تواستیم با یک سیاست معقول و توصیه شده توسط پژوهشکان، مقادیر ناموجود را جایگذاری کرده و جلوی از دست رفتن این اطلاعات را بگیریم.

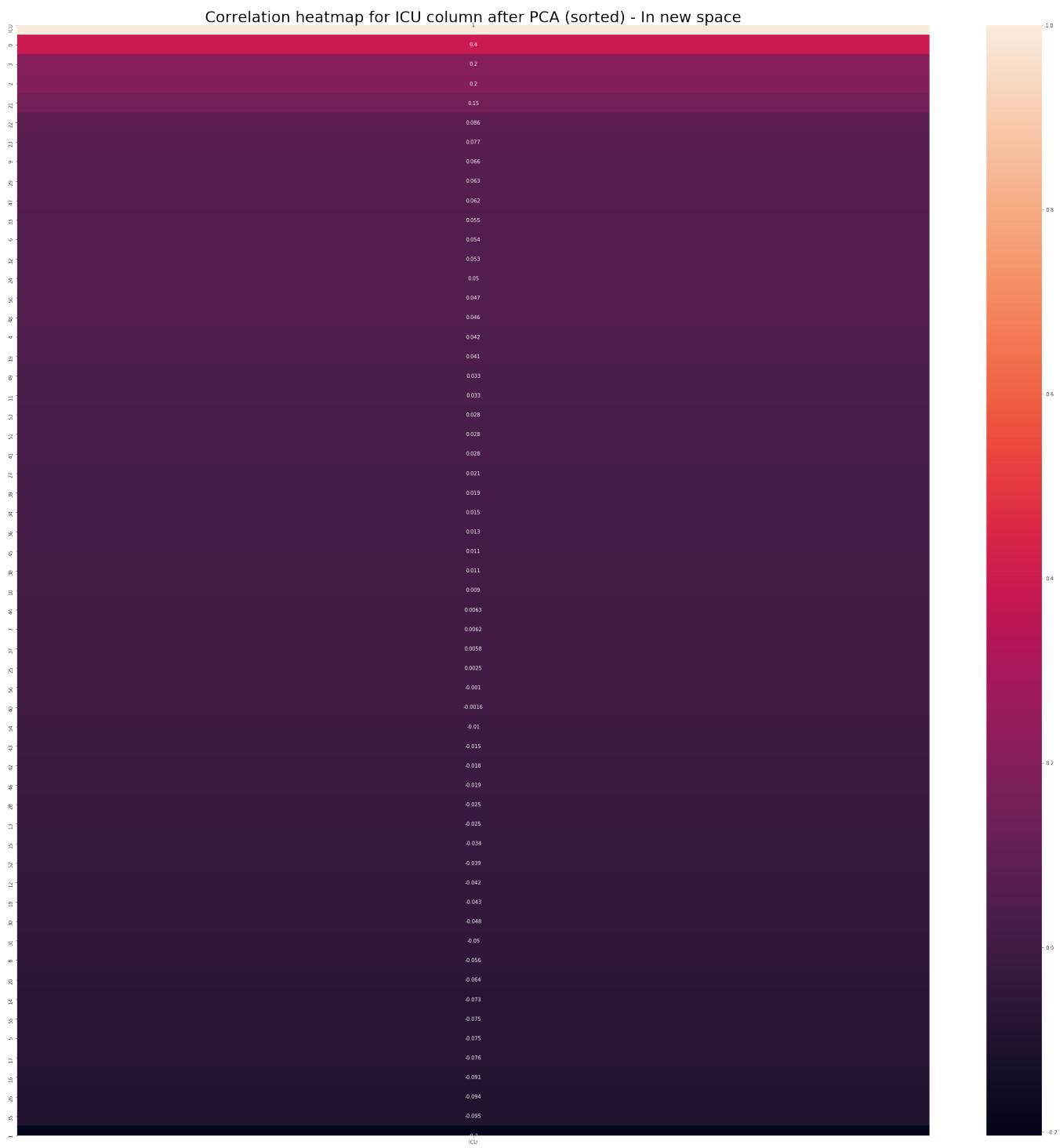
در این مرحله، برای کاهش تعداد ستون‌های مجموعه دادگان، از الگوریتم کاهش بعد PCA استفاده می‌کنیم که با حفظ حداقل واریانس، بعد نمونه‌ها را کاهش می‌دهد. در قدم اول میزان حفظ واریانس را برابر ۹۹ درصد قرار دادیم. به عنوان شروع برای آزمون و خطأ، رقم مطلوبی به نظر می‌رسد. الگوریتم ضمن حفظ ۹۹ درصد واریانس، تعداد ستون‌ها را از ۱۸۹ به ۵۷ کاهش داد که تفاوت چشم‌گیری می‌باشد. نقشه‌ی حرارتی همبستگی با ستون هدف (ICU) پس از اعمال کاهش بعد در شکل ۱۸ قابل مشاهده است. این تصویر برای فضای جدید ویژگی‌ها رسم شده و تصویر مشابهی پس از بازگرداندن ویژگی‌های جدید به فضای قبلی در نوت‌بوک پروژه موجود است که بیشتر به شکل ۱۷ شباهت دارد و برای جلوگیری از شلوغی بیش از حد در این گزارش نمایش داده نشده است.

هدف از بررسی این موارد این است که تاثیر کاهش بعد روی عملکرد نهایی شناخته شود. سیاست کلی ما این است که تا قبل از شروع فرایند اصلی یادگیری، چندین مجموعه‌ی داده داشته باشیم و بدایم مدل‌های یادگیری مانشین با کدام یک از این‌ها عملکرد بهتری دارند که بیشتر از همان مجموعه استفاده کنیم. تا این نقطه، ما چهار مجموعه‌ی داده داریم:

- با بعد اصلی و با حضور داده‌های پرت
- با بعد کاهش داده شده و با حضور داده‌های پرت
- با بعد اصلی و بدون حضور داده‌های پرت
- با بعد کاهش داده شده و بدون حضور داده‌های پرت

۳-۳ عدم استفاده از نمونه‌های حاضر در مراقبت ویژه

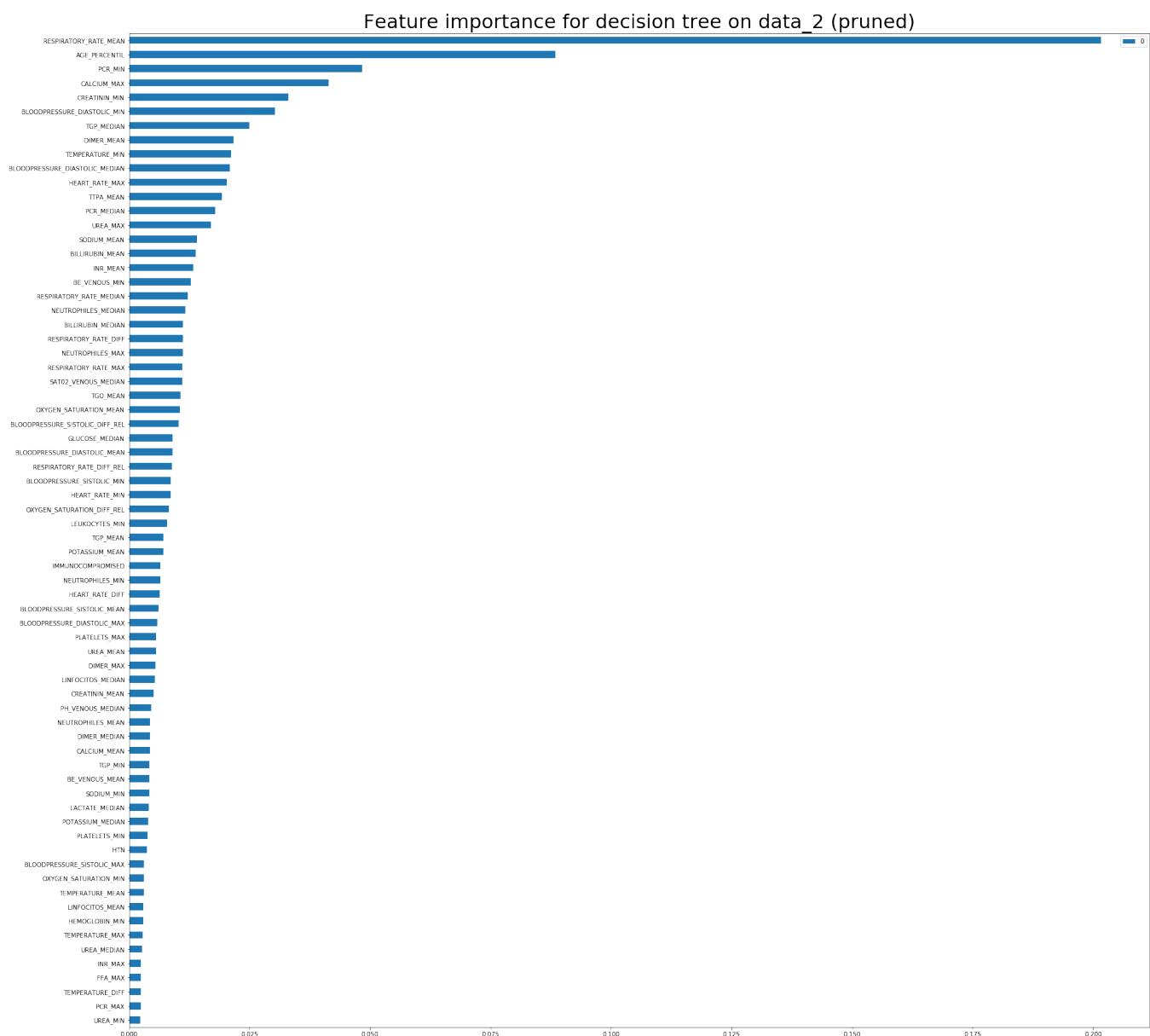
منتشرکنندگان مجموعه‌ی داده توصیه کرده‌اند که با توجه به نامشخص تر بودن زمان داده‌گیری در بخش مراقبت‌های ویژه، بهتر است که از سطرهایی که خود برچسب حضور در مراقبت‌های ویژه دارند برای یادگیری استفاده نشده و به جای آن‌ها، پنجه‌های زمانی پیشین بیمارانی که در نهایت به ICU منتقل شدند با حضور در مراقبت‌های ویژه برچسب‌گذاری شوند و از آن‌ها برای یادگیری استفاده شود. همان‌طور که در بخش مصوروسازی ذکر شد، ستون جدیدی که پنجه‌های پیشین را هم برچسب‌گذاری کرده است به داده‌ها اضافه کردیم اما در این ستون، سطرهایی که از ابتدا برچسب حضور در ICU داشتند برخلاف توصیه‌ی منتشرکنندگان حذف نشده است. لذا مجموعه‌ی جدیدی درست شد که این مورد در آن رعایت شده باشد و کنار چهار مجموعه‌ی ای که انتهای زیر بخش قبل ذکر کردیم قرار گرفت تا تست و بررسی شود.



شکل ۱۸: همبستگی (correlation) ویژگی‌های مجموعه با ستون هدف به شکل نقشه‌ی حرارتی و مرتب شده از بیشترین به کمترین، پس از اعمال الگوریتم کاهش بعد PCA با حفظ ۹۹ درصد واریانس (تعداد ستون‌ها از ۱۸۹ به ۵۷ کاهش یافت)

	0	1	2	3	4	5	6	7	8	9	Mean	Std
Accuracy	0.923611	0.934028	0.909722	0.892361	0.899306	0.909722	0.923611	0.892361	0.909722	0.885417	0.907986	0.014995
F1_score	0.928571	0.923695	0.913333	0.901587	0.901695	0.898438	0.927632	0.889680	0.910959	0.888889	0.908448	0.014012
Precision	0.922581	0.950413	0.907285	0.893082	0.910959	0.905512	0.940000	0.886525	0.910959	0.904110	0.913142	0.018693
Recall	0.934641	0.898438	0.919463	0.910256	0.892617	0.891473	0.915584	0.892857	0.910959	0.874172	0.904046	0.016526
ROC_AUC	0.922876	0.930469	0.909372	0.890734	0.899546	0.908001	0.924210	0.892375	0.909705	0.885991	0.907328	0.014447

شکل ۱۹: نتایج ده مرتبه یادگیری و آزمون درخت تصمیم با مجموعه‌ی data2 (شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطحی)



شکل ۲۰: میزان اهمیت ویژگی‌ها برای درخت تصمیم، آموزش داده شده با مجموعه‌ی data2

	Train: 0	Train: 1	Train: 2	Train: 3	Train: 4	Train: 5	Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
Accuracy	0.923611	0.940972	0.916667	0.909722	0.916667	0.916667	0.881944	0.906250	0.895833	0.923611	0.913194	0.015372
F1_score	0.924658	0.937729	0.919463	0.900763	0.911765	0.918919	0.885135	0.903915	0.881890	0.919708	0.910394	0.016672
Precision	0.924658	0.934307	0.925676	0.900763	0.911765	0.925170	0.867550	0.894366	0.896000	0.940299	0.912055	0.021185
Recall	0.924658	0.941176	0.913333	0.900763	0.911765	0.912752	0.903448	0.913669	0.868217	0.900000	0.908978	0.017848
ROC_AUC	0.923596	0.940983	0.916812	0.908980	0.916409	0.916807	0.881794	0.906499	0.893228	0.922973	0.912808	0.015679

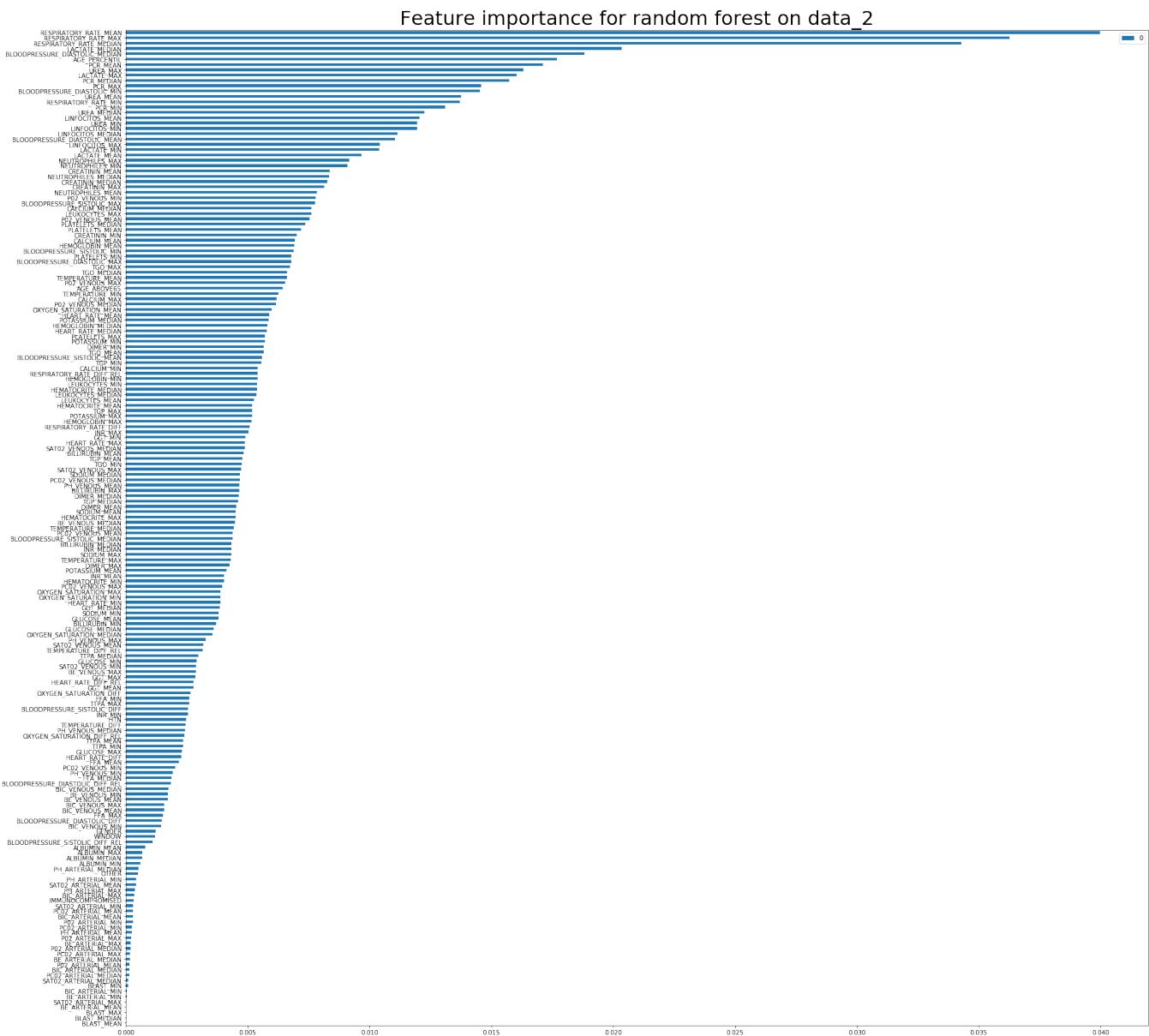
شکل ۲۱: نتایج ده مرتبه یادگیری و آزمون با درخت تصمیم و مجموعه‌ی 2 data (شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطري)

	Train: 0	Train: 1	Train: 2	Train: 3	Train: 4	Train: 5	Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
Accuracy	0.812500	0.850694	0.767361	0.815972	0.822917	0.840278	0.805556	0.833333	0.819444	0.812500	0.818056	0.021472
F1_score	0.813793	0.849123	0.763251	0.826230	0.814545	0.844595	0.822785	0.834483	0.827815	0.805755	0.820237	0.022938
Precision	0.786667	0.870504	0.812030	0.834437	0.829630	0.856164	0.860927	0.840278	0.816993	0.811594	0.831922	0.024655
Recall	0.842857	0.828767	0.720000	0.818182	0.800000	0.833333	0.787879	0.828767	0.838926	0.800000	0.809871	0.034672
ROC_AUC	0.813320	0.851003	0.769420	0.815807	0.822297	0.840580	0.808574	0.833398	0.818744	0.812162	0.818531	0.020885

شکل ۲۲: نتایج ده مرتبه یادگیری و آزمون با درخت تصمیم و مجموعه‌ی 2 data (شیوه‌ی جداسازی داده‌ای یادگیری و آزمون: تصادفی سطري)

	Train: 0	Train: 1	Train: 2	Train: 3	Train: 4	Train: 5	Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
Accuracy	0.975694	0.958333	0.972222	0.937500	0.961806	0.986111	0.975694	0.954861	0.975694	0.940972	0.963889	0.015230
F1_score	0.976744	0.958621	0.972414	0.930769	0.958801	0.986577	0.974910	0.949416	0.975610	0.943144	0.962701	0.016697
Precision	0.973510	0.965278	0.965753	0.909774	0.955224	0.993243	0.985507	0.945736	0.972222	0.921569	0.958782	0.025218
Recall	0.980000	0.952055	0.979167	0.952756	0.962406	0.980000	0.964539	0.953125	0.979021	0.965753	0.966882	0.011304
ROC_AUC	0.975507	0.958422	0.972222	0.939111	0.961848	0.986377	0.975467	0.954688	0.975717	0.940623	0.963998	0.015026

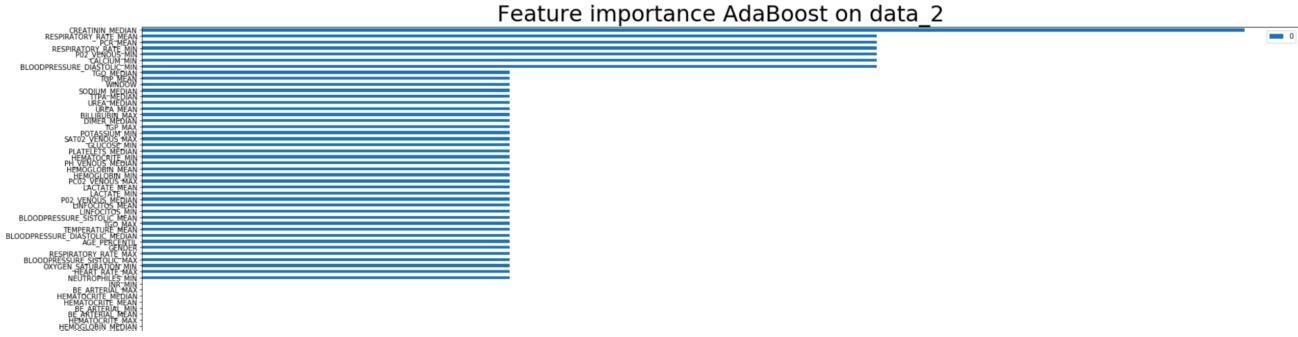
شکل ۲۳: نتایج ده مرتبه یادگیری و آزمون با جنگل تصادفی و مجموعه‌ی 2 data (شیوه‌ی جداسازی داده‌ای یادگیری و آزمون: تصادفی سطري)



شکل ۲۴: میزان اهمیت ویژگی‌ها برای جنگل تصادفی، آموزش داده شده با مجموعه‌ی data2 (نتایج در شکل ۲۳)

	Train: 0	Train: 1	Train: 2	Train: 3	Train: 4	Train: 5	Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
Accuracy	0.972222	0.965278	0.968750	0.961806	0.961806	0.972222	0.951389	0.961806	0.961806	0.947917	0.962500	0.007575
F1_score	0.972414	0.963768	0.969697	0.957854	0.962457	0.972028	0.956250	0.962199	0.962199	0.944649	0.962352	0.007885
Precision	0.979167	0.956835	0.972973	0.961538	0.965753	0.958621	0.932927	0.958904	0.965517	0.927536	0.957977	0.015364
Recall	0.965753	0.970803	0.966443	0.954198	0.959184	0.985816	0.980769	0.965517	0.958904	0.962406	0.966979	0.009348
ROC_AUC	0.972313	0.965534	0.968833	0.961176	0.961861	0.972500	0.948718	0.961780	0.961846	0.948945	0.962351	0.007888

شکل ۲۵: نتایج ده مرتبه یادگیری و آزمون جنگل تصادفی با مجموعه‌ی data2 و با حذف ویژگی‌های بی‌اهمیت (شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطحی)



شکل ۲۶: میزان اهمیت ویژگی‌ها برای AdaBoost، آموزش داده شده با مجموعه‌ی data2

مانده با همان سیاست قبلی داده‌هاشان تکمیل شده است. این مجموعه داده نیز با استفاده از هر سه مدل مورد آزمایش قرار گرفت و با هیچ یک از مدل‌ها نتیجه‌ی بهتری نسبت به آزمایش‌های پیشین به دست نداد. جداول و نتایج دقیق این آزمون‌ها نیز در نوتبوک پروژه در دسترس هستند اما با توجه به نتیجه‌گیری انجام شده لزومی نداشت که در این گزارش به نمایش درآیند.

۶-۴ نتیجه‌گیری

از تمامی آزمایش‌های صورت گرفته، می‌توان نتیجه گرفت که کاهش بعد (با هر یک از روش‌های تست شده) باعث پیشرفت عملکرد نخواهد شد و حضور یا عدم حضور داده‌های پرت هم تاثیر چندانی در نتیجه نمی‌گذارد. با این وجود، در بخش یادگیری و تست مدل‌های مختلف باز هم مجموعه دادگان متفاوت را آزمایش خواهیم کرد.

۵ پیش‌بینی زودهنگام: یادگیری و آزمون

در این بخش بناست که مدل‌های مختلف و عملکردشان برای پیش‌بینی زودهنگام بررسی شود. پیش‌بینی زودهنگام یعنی داده‌های تست، فقط شامل نمونه (سطر) هایی باشد که مربوط به پنجره‌ی زمانی اول هستند. نتایج این بخش نشان‌دهنده‌ی آن است که اگر یک بیمار به عنوان مبتلا به ویروس کرونا تشخیص داده شود، با چه دقیقی می‌توان در دو ساعت اول پذیرش اش نیاز آینده‌ی او به مراقبت‌های ویژه را پیش‌بینی کرد. حائز اهمیت است که شیوه‌ی جداسازی داده‌های یادگیری و آزمون شرح داده شود. جداسازی به دوروش صورت می‌گیرد:

- جداسازی تصادفی سطري: به آن معناست که مجموعه‌ی داده شافل شده و با استفاده ازتابع train test split مجموعه‌ی یادگیری و آزمون جدا شوند.
- جداسازی تصادفی بر اساس بیمار: در مسائل علوم داده که به حوزه‌ی زیست‌شناسی و پزشکی مربوط می‌شود، داده‌های مربوط به یک فرد در حالت سالم و ناسالم تقاضا خیلی کمتری با یکدیگر دارند تا داده‌های مربوط به دو فرد سالم یا دو فرد بیمار. یعنی در این مجموعه داده، احتمالاً سطرهایی که مربوط به یک بیمار هستند، شباهت زیادی با هم دارند، خواه آن بیمار نیازمند

موجود است.

۲-۴ جنگل تصادفی (Random forest)

مراحلی که برای درخت تصمیم طی شد در این بخش برای جنگل تصادفی اجرا شده است. نتایج اجرای این آزمون‌ها به شرح مقابل می‌باشد: یادگیری و آزمون با مجموعه‌ی data2 در شکل ۲۳ و نمودار اهمیت ویژگی‌های آن در شکل ۲۴، و همچنین نتایج یادگیری و آزمون با مجموعه‌ی data2 ضمن حذف ویژگی‌های بی‌اهمیت در شکل ۲۵ قابل مشاهده است. پیداست که امتیازات خوبی به دست آمده اما مانند بخش قبل، حذف ویژگی‌های بی‌اهمیت کمک شایانی نکرده است.

۳-۴ AdaBoost

مراحلی که برای درخت تصمیم و جنگل تصادفی انجام شده بود برای مدل AdaBoost هم تکرار شد. نتایج به صورت کلی بهتر نبود و امتیاز F1 حدود ۸۷ حاصل شد. مجدداً استفاده از مجموعه‌های کاهش بعد داده شده با PCA و یا با حذف ویژگی‌های بی‌اهمیت تاثیری در بهبود نتیجه نداشت. نمودار اهمیت ویژگی‌ها برای الگوریتم AdaBoost در شکل ۲۶ قابل مشاهده است.

۴-۴ اجتماع ویژگی‌های پر اهمیت

یادگیری و آزمون آزمایشی با سه مدل صورت گرفت و برای هر مدل متوجه شدیم کدام ویژگی‌ها پر اهمیت هستند. در آزمایش بعدی، همه‌ی ویژگی‌هایی که برای این سه مدل مهم بودند را گردآوری کرده و سایر ویژگی‌ها کنار گذاشته شدند. مجدداً برای هر سه مدل درخت تصمیم، جنگل تصادفی و الگوریتم AdaBoost فرایند یادگیری و آزمون تکرار شد و نتیجه‌های که بهتر از آزمایش‌های پیشین باشد مشاهده نشد. تا این مرحله از آزمایش‌ها، هیچ مشاهده‌ای نداشتمیم که تایید کند حذف تعدادی ویژگی (با هر روشه) می‌تواند باعث بهبود نتایج شود.

۵-۴ بازنگری در سیاست مقابله با داده‌های ناموجود

در کنار آزمایش‌هایی که تا این مرحله صورت گرفت، سیاست حذف ویژگی‌هایی که داده‌های ناموجود بسیاری داشته‌اند نیز آزموده شد. مجموعه داده‌ای به نام data3 در دست است که در آن ویژگی‌هایی که بیش از نصف مقادیرشان ناموجود بوده حذف شده‌اند و ستون‌های باقی

Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
0.756098	0.770270	0.705882	0.769231	0.768100	0.039729
0.787234	0.760563	0.705882	0.769231	0.778374	0.037966
0.804348	0.729730	0.666667	0.810811	0.767573	0.047296
0.770833	0.794118	0.750000	0.731707	0.792542	0.053650
0.753064	0.772059	0.708333	0.771259	0.767801	0.040739

شکل ۲۸: همتای شکل ۲۷، با این تفاوت که یادگیری با همهی پنجره‌های زمانی انجام شده است.

یادگیری، تابع هزینه و ضریب رگولاریزیشن آزموده شده و پس از ۴۰ مرتبه اجرای الگوریتم با ابرپارامترهای متفاوت (و ۱۰ مرتبه تکرار جستجو)، نتیجه آن شد که بهترین تابع هزینه L_1 است و بهترین ضریب رگولاریزیشن 1.5 نتایج نهایی را در شکل ۳۱ مشاهده می‌کنید که میانگین امتیاز F1 برابر با 80 به دست آمده است. مدل با مجموعه داده‌های دیگری که در اختیار داریم نیز تست شد و نتایج بهتری به دست نیامد.

۲-۵ Logistic Regression و جداسازی بر اساس بیمار

این بخش مستقیماً با جستجوی جدولی برای بهترین ابرپارامترها آغاز شده و پس از به دست آمدن آن‌ها، ۱۰ مرتبه یادگیری و آزمون انجام شده است. نتایج این بخش در شکل ۳۲ به نمایش گذاشته شده است.

۳-۵ درخت تصمیم و جداسازی تصادفی سط्रی

مشابه با فرایندی که برای مدل قبلی انجام شد برای این مدل نیز صورت گرفت. ابتدا فقط با داده‌های پنجره‌های ابتدایی آموزش دادیم که نتیجه‌اش از Logistic Regression هم نامطلوب‌تر بود. نتایج دقیق این آزمایش‌ها را می‌توانید در نوتبوک پروژه دنبال کنید. با اضافه شدن پنجره‌های بعدی به مجموعه یادگیری از چهار پنجره‌های ابتدایی حاصل شد که باز هم مثل قبل تفاوت معنی‌داری با آموزش با همهی پنجره‌ها نداشت. همان‌طور که در شکل ۳۳ قابل مشاهده است، امتیاز F1 برابر با 95 به دست آمده است. نمودار امتیازات این مدل برای بیشینه عمق‌های متفاوت در شکل ۳۴ رسم شده است. برای تنظیم بهتر ابرپارامترها از جستجوی جدولی استفاده شد و مجدداً با بهترین ابرپارامترهای حاصل شده، مدل تحت آموزش و آزمون قرار گرفت. ابرپارامترهای آزموده شده در جستجوی جدولی این مدل را می‌توانید در شکل ۳۵ مشاهده کنید (جستجو ۱۰ مرتبه انجام شده است). در ادامه، نتایج پیشرفته نداشتند و همان شکل ۳۳ نماینده‌ی بهترین نتایج این بخش است. مدل با مجموعه داده‌ای دیگری که در اختیار داریم نیز تست شد و نتایج بهتری به دست نیامد.

Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
0.746667	0.812500	0.797297	0.750000	0.779116	0.038414
0.739726	0.814815	0.782609	0.790698	0.785726	0.035934
0.771429	0.846154	0.771429	0.755556	0.779474	0.034752
0.710526	0.785714	0.794118	0.829268	0.794013	0.054399
0.747155	0.813910	0.797059	0.737215	0.778207	0.040304

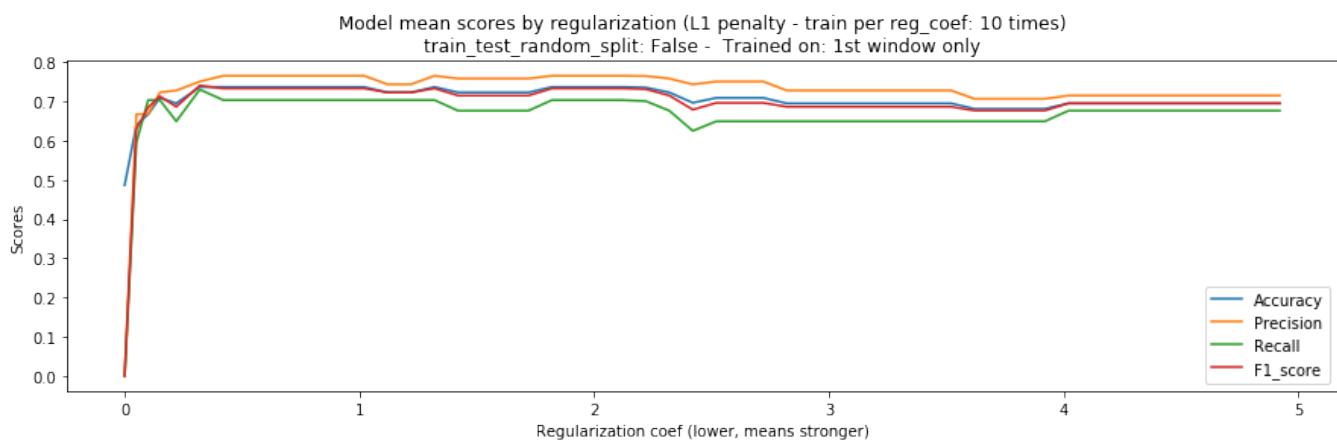
شکل ۲۷: میانگین نتایج ده بار آموزش و آزمون LogisticRegression در بخش پیش‌بینی زودهنگام (یادگیری با چهار پنجره‌ی اول و آزمون با پنجره‌ی اول) - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطري

مراقبت‌های ویژه باشد یا نباشد. حال برای مثال اگر در حالت دیگر جداسازی، یعنی جداسازی تصادفی سطري، دو سطر از یک بیمار در داده‌های آزمون باشد و سه سطر از همان بیمار در داده‌های یادگیری، ممکن است دقتی بالا اما کم دور از واقعیت نتیجه شود. چرا که در عمل قرار است این سیستم برای بیماران جدید پیش‌بینی انجام دهد نه همان قبلی‌ها.

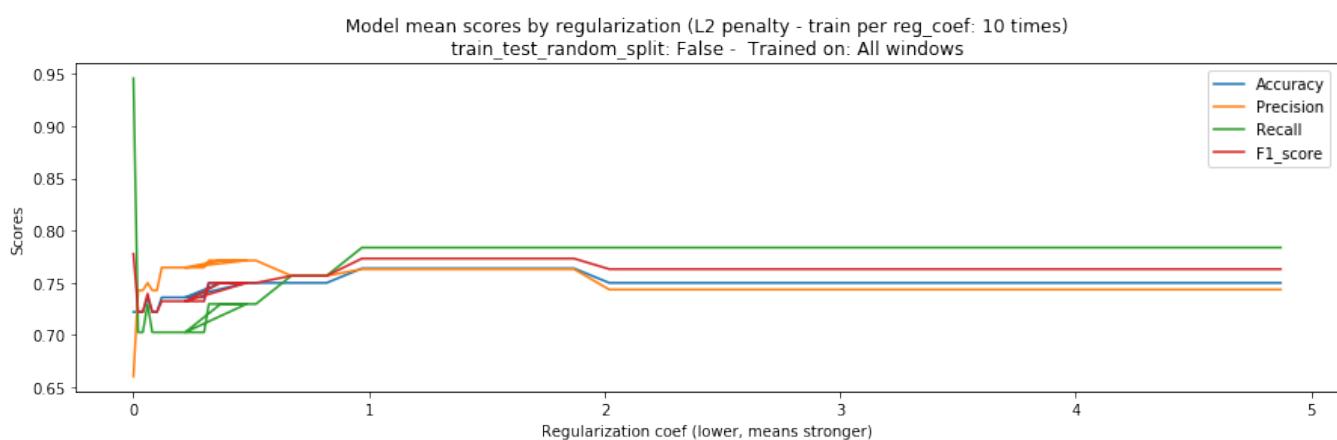
۱-۵ Logistic Regression و جداسازی تصادفی سطري

این پرسش اساسی وجود دارد که برای پیش‌بینی زودهنگام، بهترین مجموعه‌ی داده برای یادگیری بهتر است که شامل کدام پنجره‌های زمانی باشد؟ از طرفی ممکن است داده‌های پنجره‌ی اول بهتر باشند چون شرایط نزدیک‌تری به داده‌های آزمون دارند و از طرفی هم مقدارشان به نسبت کم است. به منظور بررسی این موضوع، ابتدا یک مدل Logistic Regression با مجموعه‌ی داده‌هایی که فقط شامل پنجره‌ی زمانی اول هستند آموزش داده و آزموده شد. سپس، مرحله به مرحله پنجره‌های زمانی جلوتر را به داده‌های یادگیری اضافه کردیم. میانگین امتیاز F1 از 72 برای اولین پنجره‌ی زمانی آغاز شد و با اضافه شدن پنجره‌های بعدی به داده‌های یادگیری تا 77 و 78 برای چهار پنجره‌ی اول و هر پنج پنجره پیش رفت. تفاوت بین نتایج چهار پنجره‌ی اول و همهی پنجره‌ها ناچیز و قابل صرف نظر است. میانگین امتیازات هنگامی که مدل با چهار پنجره‌ی ابتدایی آموزش داده شده در شکل ۲۷ و برای زمانی که با همهی پنجره‌ها آموزش داده شده در شکل ۲۸ قابل مشاهده است. کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است.

در ادامه، چندین نمودار امتیازات برای پارامترهای رگولاریزیشن مختلف رسم شده است که دو نمونه از آن‌ها را در شکل‌های 29 و 30 مشاهده می‌کنید. رسم این نمودارها برای پیدا کردن بهترین ضریب رگولاریزیشن به هدف جلوگیری از بیش‌برازش (over fitting) انجام شد و پس از به دست آمدن نتایج حدودی، برای تنظیم ابرپارامترهای مدل از جستجوی جدولی (grid search) استفاده کردیم. پارامترهای متفاوتی از جمله تعداد بیشینه‌ی گام، ضریب



شکل ۲۹: یک نمونه از نمودارهای میانگین امتیازات Logistic Regression بر حسب ضرایب رگولاریزیشن در بخش پیش‌بینی زودهنگام



شکل ۳۰: یک نمونه از نمودارهای میانگین امتیازات Logistic Regression بر حسب ضرایب رگولاریزیشن در بخش پیش‌بینی زودهنگام

۴-۵ درخت تصمیم و جداسازی بر اساس بیمار

مجدداً با استفاده از جستجوی جدولی ابرپارامترها تنظیم شده و امتیاز F1 برابر با ۶۷ به دست آمد که به خوبی نتیجه LogisticRegression در این بخش نیست. نتایج نهایی این مدل با جداسازی داده‌های یادگیری و آزمون بر اساس بیمار، در شکل ۳۶ نمایش داده شده است.

۵-۵ جنگل تصادفی و جداسازی تصادفی سطري

حتماً با مطالعه قسمت‌های قبل با شیوه‌ی کلی فرایندهای صورت گرفته آشنا شده‌اید. در این قسمت از تکرار همان توضیحات پرهیز شده و فقط نتایج شرح داده می‌شود. پس از تنظیم ابرپارامترها، داده‌های یادگیری را ابتدا با پنجره‌ی اول تشکیل دادیم و مرحله به مرحله تا همه‌ی پنجره‌ها پیش رفتیم. در نهایت نتیجه‌ی مطلوب ۹۸ برای امتیاز F1 به دست آمد. جدول امتیازات این مدل در شکل ۳۷ قابل مشاهده است.

۶-۵ جنگل تصادفی و جداسازی بر اساس بیمار

مانند قسمت پیشین، از تکرار جزئیات فرایند طی شده پرهیز می‌شود. پس از تنظیم ابرپارامترها، داده‌های یادگیری ابتدا با پنجره‌ی اول تشکیل شد و مرحله به مرحله تا در بر گرفتن همه‌ی پنجره‌ها پیش رفت. در نهایت نتیجه‌ی ۷۹ برای امتیاز F1 به دست آمد. جدول امتیازات این مدل در شکل ۳۸ قابل مشاهده است.

۷-۵ AdaBoost و جداسازی سطري

فرایند طی شده در قسمت‌های قبلی برای این مدل هم تکرار شد. جستجوی جدولی برای تنظیم ابرپارامترها انجام شده و نتیجه‌ی این بخش در شکل ۳۹ به نمایش درآمده است. امتیاز F1 برابر با ۹۷ به دست آمده است. تمامی نتایج، جداول و جزئیات مربوطه در نوتبوک پروژه موجود بوده و جهت جلوگیری از تکرار و شلوغی، در این گزارش ارائه نشده‌اند.

۸-۵ AdaBoost و جداسازی بر اساس بیمار

مانند قبل، جستجوی جدولی برای تنظیم ابرپارامترها انجام شده و در شکل ۴۰ مشاهده می‌کنید که امتیاز F1 برابر با ۶۸ به دست آمده است. تمامی نتایج، جداول و جزئیات مربوطه در نوتبوک پروژه موجود هستند و جهت جلوگیری از تکرار و شلوغی، در این گزارش ارائه نشده‌اند.

۶ پیش‌بینی عمومی: یادگیری و آزمون با تمامی پنجره‌های زمانی

پیش‌بینی زودهنگام در بخش پیشین مورد بررسی قرار گرفت و نتایج آن ارائه شد. اما آخرین مرحله‌ی این پروژه همچنان باقی مانده است. یعنی به دست آوردن مدلی که در پیش‌بینی نیاز بیمار به مراقبت‌های ویژه، با داده‌های آزمونی که از تمامی پنجره‌های زمانی تشکیل شده‌اند بتواند عملکرد قابل قبولی داشته باشد. مسیری که در این قسمت طی کردیم

Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
0.821918	0.703704	0.809524	0.880000	0.797301	0.048719
0.831169	0.727273	0.793103	0.896552	0.804166	0.046359
0.820513	0.680851	0.718750	0.906977	0.781846	0.078572
0.842105	0.780488	0.884615	0.886364	0.832910	0.039077
0.821053	0.702744	0.820686	0.878666	0.800143	0.048894

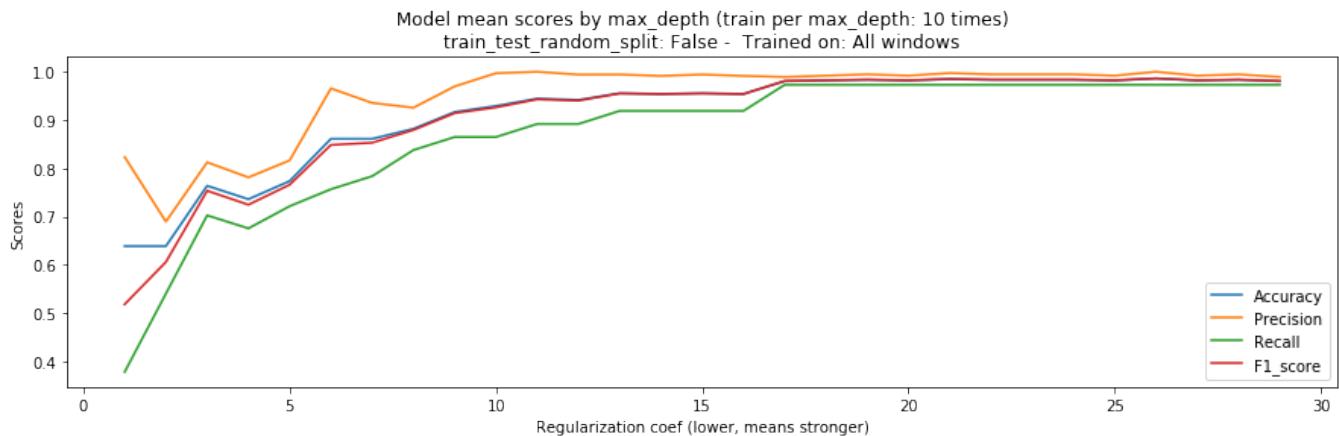
شکل ۳۱: میانگین نتایج ده بار آموزش و آزمون LogisticRegression در بخش پیش‌بینی زودهنگام - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطري

Train: 6	Train: 7	Train: 8	Train: 9	Mean	Std
0.746479	0.662162	0.645161	0.734375	0.725610	0.046611
0.756757	0.626866	0.656250	0.721311	0.722663	0.055578
0.700000	0.677419	0.617647	0.628571	0.697171	0.072282
0.823529	0.583333	0.700000	0.846154	0.757962	0.073925
0.749603	0.660088	0.646875	0.752024	0.729764	0.045815

شکل ۳۲: میانگین نتایج ده بار آموزش و آزمون LogisticRegression در بخش پیش‌بینی زودهنگام - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: بر اساس بیمار

Train 6	Train 7	Train 8	Train 9	Mean	Std
0.967742	0.951220	0.906250	0.977273	0.952130	0.019508
0.973001	0.948026	0.915004	0.966667	0.953094	0.016777

شکل ۳۳: میانگین نتایج ده بار آموزش و آزمون درخت تصمیم در بخش پیش‌بینی زودهنگام - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطري



شکل ۳۴: نمودار میانگین امتیازات درخت تصمیم بر حسب بیشینه عمق در بخش پیش‌بینی زودهنگام

مشابه بخش پیش‌بینی زودهنگام است. یعنی با استفاده از جست‌وجوی جدولی (grid search) ابرپارامترهای مدل‌ها تنظیم شده و به آموزش و آزمون آها با دو روش جداسازی (بر اساس بیمار یا تصادفی سط्रی) پرداخته شده است. با توجه به عملکرد بهتر مدل‌های تجمعی (جنگل تصادفی و AdaBoost) در بخش پیش‌بینی زودهنگام، در این قسمت فقط این دو مدل بررسی شده‌اند.

۱-۶ جنگل تصادفی و جداسازی تصادفی سطري

مطلوب جدید و ناگفته‌ای برای ارائه وجود ندارد. تنها تفاوت این بخش با مدل جنگل تصادفی در بخش پیش‌بینی زودهنگام در این مورد است که در این مدل داده‌های آزمون بر اساس پنجره‌ی زمانی فیلتر نشده‌اند. بهترین نتیجه‌ی به دست آمده شامل امتیاز F1 برابر با ۹۷ است و در شکل ۴۱ به نمایش درآمده است.

۲-۶ جنگل تصادفی و جداسازی بر اساس بیمار

در این قسمت سعی شده تا هنگامی که داده‌های یادگیری و آزمون هیچ دو سطري که مربوط به یک شخص باشند را به اشتراک نمی‌گذارند، از جنگل تصادفی استفاده شده تا پیش‌بینی نیاز بیماران به مراقبه‌های ویژه برای حالتی که داده‌های آزمون تمامی پنجره‌های زمانی را شامل می‌شود به شکل مطلوبی صورت گیرد. فرایندهای انجام شده در بخش‌های پیشین، در این بخش نیز صورت گرفته و در شکل ۴۲ مشاهده می‌کنید که امتیاز F1 برابر با ۸۰ به دست آمده است.

۳-۶ AdaBoost و جداسازی سطري

فرایند طی شده در قسمت‌های قبلی برای این مدل هم تکرار شد. جست‌وجوی جدولی برای تنظیم ابرپارامترها انجام شده و نتیجه‌ی این بخش در شکل ۴۳ به نمایش درآمده است. امتیاز F1 برابر با ۹۴ شده و تمامی نتایج، جداول و جزئیات مربوطه در نوت‌بوک پروژه موجود هستند (جهت جلوگیری از تکرار و شلوغی، در این گزارش ارائه نشده‌اند). علاوه‌بر نتایج این آزمون‌ها، مجموعه دادگان دیگری

```
param_grid = {
    'criterion': ['gini', 'entropy'],
    'max_depth': [20, 50, 150, 200],
    'splitter': ['best', 'random'],
    'min_samples_leaf': [1, 3, 5, 10],
    'max_features': ['sqrt', None, 'log2']
}
```

شکل ۳۵: ابرپارامترهای استفاده شده در جست‌وجوی جدولی (grid Search) برای مدل درخت تصمیم در بخش پیش‌بینی زودهنگام

Train 6	Train 7	Train 8	Train 9	Mean	Std
0.694737	0.720000	0.75000	0.625000	0.679091	0.04357
0.675000	0.729268	0.72381	0.640698	0.676765	0.03267

شکل ۳۶: میانگین نتایج ده بار آموزش و آزمون درخت تصمیم در بخش پیش‌بینی زودهنگام - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: بر اساس بیمار

	Train #1	Train #2	Train #3	Train #4	Train #5	Train #6	Train #7	Train #8	Train #9	Train #10	Mean	Std
Accuracy	1.0	0.986301	0.985294		1.0	0.974684	1.0	1.0	0.975000	1.0	0.975309	0.989659
F1_score	1.0	0.988506	0.984127		1.0	0.972973	1.0	1.0	0.978261	1.0	0.972222	0.989609
Precision	1.0	0.977273	1.000000		1.0	1.000000	1.0	1.0	0.978261	1.0	0.972222	0.992776
Recall	1.0	1.000000	0.968750		1.0	0.947368	1.0	1.0	0.978261	1.0	0.972222	0.986660
ROC_AUC	1.0	0.983333	0.984375		1.0	0.973684	1.0	1.0	0.974425	1.0	0.975000	0.989082

شکل ۳۷: میانگین نتایج ده بار آموزش و آزمون جنگل تصادفی در بخش پیش‌بینی زودهنگام (یادگیری با چهار پنجره‌ی اول و آزمون با پنجره‌ی اول) کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطري

	Train #1	Train #2	Train #3	Train #4	Train #5	Train #6	Train #7	Train #8	Train #9	Train #10	Mean	Std
Accuracy	0.814286	0.858974	0.773333	0.810127	0.714286	0.793478	0.769231	0.702703	0.688312	0.776471	0.770120	0.051379
F1_score	0.821918	0.870588	0.813187	0.819277	0.736842	0.786517	0.820000	0.717949	0.739130	0.765432	0.789084	0.045848
Precision	0.810811	0.840909	0.804348	0.850000	0.800000	0.714286	0.773585	0.717949	0.680000	0.775000	0.776689	0.053551
Recall	0.833333	0.902439	0.822222	0.790698	0.682927	0.875000	0.872340	0.717949	0.809524	0.756098	0.806253	0.067163
ROC_AUC	0.813725	0.856625	0.761111	0.812016	0.720774	0.802885	0.742622	0.701832	0.676190	0.775776	0.766356	0.053723

شکل ۳۸: میانگین نتایج ده بار آموزش و آزمون جنگل تصادفی در بخش پیش‌بینی زودهنگام (یادگیری با چهار پنجره‌ی اول و آزمون با پنجره‌ی اول) کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: بر اساس بیمار

Train #6	Train #7	Train #8	Train #9	Train #10	Mean	Std
0.979123	0.964509	0.979123	0.970772	0.960334	0.966180	0.008021
0.980469	0.965235	0.979424	0.969163	0.960825	0.966424	0.008776
0.980469	0.971193	0.975410	0.982143	0.954918	0.965710	0.014346
0.980469	0.959350	0.983471	0.956522	0.966805	0.967284	0.009168
0.979024	0.964653	0.979077	0.970229	0.960293	0.966181	0.007769

شکل ۴۱: میانگین نتایج ده بار آموزش و آزمون جنگل تصادفی در بخش پیش‌بینی عمومی - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطري

Train 6	Train 7	Train 8	Train 9	Mean	Std
0.964706	0.955056	0.985075	0.986486	0.972922	0.014249
0.967033	0.956522	0.983607	0.985507	0.973726	0.012553
0.956522	0.977778	1.000000	1.000000	0.984298	0.017125
0.977778	0.936170	0.967742	0.971429	0.963752	0.019649
0.963889	0.956180	0.983871	0.985714	0.972695	0.013780

شکل ۳۹: میانگین نتایج ده بار آموزش و آزمون AdaBoost در بخش پیش‌بینی زودهنگام - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطري

Train #6	Train #7	Train #8	Train #9	Train #10	Mean	Std
0.804819	0.773196	0.827027	0.789610	0.782353	0.804005	0.027717
0.813793	0.803571	0.828877	0.813793	0.767296	0.809588	0.040450
0.786667	0.789474	0.778894	0.753191	0.797386	0.796435	0.061675
0.842857	0.818182	0.885714	0.885000	0.739394	0.826865	0.044599
0.804355	0.766234	0.830037	0.785743	0.781126	0.803821	0.027901

Train 6	Train 7	Train 8	Train 9	Mean	Std
0.704225	0.670588	0.714286	0.655172	0.685416	0.039299
0.746988	0.666667	0.720930	0.625000	0.683994	0.049059
0.720930	0.666667	0.720930	0.625000	0.678477	0.044522
0.775000	0.666667	0.720930	0.625000	0.692588	0.068393
0.693952	0.670543	0.714124	0.652926	0.684928	0.038038

شکل ۴۲: میانگین نتایج ده بار آموزش و آزمون جنگل تصادفی در بخش پیش‌بینی عمومی - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: بر اساس بیمار

شکل ۴۰: میانگین نتایج ده بار آموزش و آزمون AdaBoost در بخش پیش‌بینی زودهنگام - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: بر اساس بیمار

Train 6	Train 7	Train 8	Train 9	Mean	Std
0.926893	0.929504	0.955614	0.955614	0.942820	0.014974
0.930000	0.926027	0.955844	0.957606	0.942596	0.016228
0.930000	0.928571	0.958333	0.969697	0.946437	0.020909
0.930000	0.923497	0.953368	0.945813	0.939034	0.017859
0.926749	0.929249	0.955631	0.956240	0.942863	0.015128

شکل ۴۳: میانگین نتایج ده بار آموزش و آزمون AdaBoost در بخش پیش‌بینی عمومی - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: تصادفی سطحی

Train 6	Train 7	Train 8	Train 9	Mean	Std
0.775956	0.732782	0.737819	0.830835	0.779779	0.037817
0.788333	0.714298	0.737960	0.824123	0.769699	0.045914

شکل ۴۴: میانگین نتایج ده بار آموزش و آزمون AdaBoost در بخش پیش‌بینی عمومی - کادر قرمز رنگ سطر مربوط به امتیاز F1 را مشخص کرده است. شیوه‌ی جداسازی داده‌های یادگیری و آزمون: بر اساس بیمار

که قبل توضیح داده شد نیز آزموده شدند اما هیچ‌یک نتیجه‌ی بهتری از مجموعه‌ی اصلی (یعنی data2) نداشتند.

۴-۶ AdaBoost و جداسازی بر اساس بیمار

مانند قبل، جست‌وجوی جدولی برای تنظیم ابرپازامترها انجام شده و در شکل ۴۴ مشاهده می‌کنید که امتیاز F1 برابر با ۷۸ به دست آمده است. تمامی نتایج، جداول و جزئیات مربوطه در نوتبوک پروژه موجود هستند و جهت جلوگیری از تکرار و شلوغی، در این گزارش ارائه نشده‌اند.

۷ نتایج نهایی

- پیش‌بینی زودهنگام با جداسازی تصادفی سطحی: ۹۸ امتیاز F1
- پیش‌بینی زودهنگام با جداسازی بر اساس بیمار: ۷۹ امتیاز F1
- پیش‌بینی عمومی با جداسازی سطحی: ۹۷ امتیاز F1
- پیش‌بینی عمومی با جداسازی بر اساس بیمار: ۸۰ امتیاز F1