دانشکدگان علوم
دانشکده ریاضی، آمار و علوم کامپیوتر

انتخاب ویژگی با استفاده از الگوریتم‌های فراابتکاری الهام گرفته از طبیعت

نگارنده: سپهر امیدوار

استاد راهنما: دکتر باقر باباعلی

پایان‌نامه برای دریافت درجه کارشناسی
در رشته علوم کامپیوتر

بهمن ماه ۱۴۰۰

چکیده

مسئله انتخاب ویژگی [1] یکی از مراحل مهم و اساسی در ایجاد یک سامانه مبتنی بر یادگیری ماشین [2] و بازشناسی الگو [3] است. انتخاب ویژگی، فرآیند انتخاب زیرمجموعه‌ای از ویژگی‌های مرتبط برای استفاده در ساخت مدل است. فرض اساسی هنگام استفاده از روش‌های انتخاب ویژگی این است که داده‌ها دارای ویژگی‌هایی هستند که زائد یا نامربوط هستند؛ لذا با حذف این دسته از ویژگی‌ها نه تنها باعث حذف اطلاعات مهم نخواهد شد، بلکه باعث افزایش قابلیت تعمیم [4] در ساخت مدل می‌شود.

یک الگوریتم انتخاب ویژگی را می‌توان جستجویی برای پیشنهاد یک زیرمجموعه از ویژگی‌ها به همراه یک معیار ارزیابی دانست که به زیرمجموعه‌های مختلف ویژگی امتیازی نظیر می‌کند. ساده‌ترین راه حل در نظر گرفتن تمام زیرمجموعه‌های ممکن است که میزان خطا را به حداقل می‌رساند. این یک جستجو جامع از فضا است که از نظر محاسباتی برای مجموعه ویژگی‌های زیاد غیرقابل حل است. در نتیجه، در این پروژه قصد داریم که ابتدا با مسئله انتخاب ویژگی، اهمیت، چالش‌ها، کاربردها و روش‌های کلاسیک آن آشنا شویم و به کمک آن به حل مسئله و بررسی نتایج بپردازیم.

[1] Feature Selection
[2] Machine Learning
[3] Pattern Recognition
[4] Generalization

فهرست مطالب

فصل ۱

مفاهیم مقدماتی

در یادگیری ماشینی و آمار، انتخاب ویژگی همان فرآیند انتخاب زیرمجموعه‌ای از ویژگی‌های مرتبط برای استفاده در ساخت مدل است. کاهش تعداد متغیرهای ورودی برای کاهش هزینه محاسباتی مدل سازی و در برخی موارد برای بهبود عملکرد مدل مطلوب است. هنگامی که با یک مجموعه داده سر و کار داریم، ممکن است این مجموعه داده هزاران ویژگی داشته باشد که ممکن است به علت هزینه محاسباتی سنگین در ساخت مدل، از انجام کار منصرف شویم؛ لذا نقش و اهمیت انتخاب ویژگی در این جا پررنگ می‌شود.

در کنار انتخاب ویژگی مفهومی تحت عنوان استخراج ویژگی [۱] نیز مطرح می‌شود که گام پیش از انتخاب ویژگی برای ساخت مدل است. پس ضروری است که ابتدا به کمک روش‌های استخراج ویژگی، ویژگی‌های مجموعه داده را استخراج کنیم و سپس از تکنیک‌های انتخاب ویژگی برای یافتن بهترین زیرمجموعه بهره ببریم و در نهایت با آموزش مدل به نتایج مطلوب دست خواهیم یافت. تکنیک های انتخاب ویژگی به دلایل مختلفی نظیر ساده‌سازی مدل،جلوگیری از نفرین ابعاد [۲] و ... استفاده می‌شود که در ادامه به آن‌ها خواهیم پرداخت.

۱.۱ ویژگی

تعریف ۱.۱. در یادگیری ماشین و بازشناسی الگو، یک ویژگی یک خصیصه فردی یا مشخصه قابل اندازه گیری یک پدیده است.ویژگی‌ها معمولاً به دو دسته رشته‌ای [۳] و عددی [۴] تقسیم می‌شوند.

[1] Feature Extraction
[2] Curse of Dimensionality
[3] String
[4] Numerical

مثال ۲.۱. هر فرد توسط ویژگی‌هایی نظیر قد، وزن، جنسیت، کد ملی، محل تولد و ... قابل بیان است.

تعریف ۳.۱. یک بردار ویژگی [۵] از مجموعه ویژگی‌های عددی یا رشته‌ای تشکیل می‌شود که هر مؤلفه آن بیان‌گر یک ویژگی است.

مثال ۴.۱. همان طور که در مثال ۲.۱ ویژگی‌هایی نظیر قد، وزن و ... برای هر فرد متصور شدیم، می‌توانیم هر انسان را برداری از ویژگی‌های عددی و رشته‌ای در نظر بگیریم.

۱.۱.۱ انواع ویژگی

همان طور که اشاره شد، دو دسته کلی ویژگی وجود دارد که نحوه به کار گیری آن‌ها در فرآیند آموزش مدل متفاوت است؛ لذا ضروری است که آن‌ها را به خوبی بشناسیم و نحوه استفاده از آن‌ها را در ساخت مدل بدانیم.

تعریف ۵.۱. ویژگی عددی، ویژگی است که تنها می‌تواند مقادیر عددی مانند طبیعی، صحیح و اعشاری داشته باشد.

تعریف ۶.۱. ویژگی رشته‌ای، ویژگی است که تنها می‌تواند مقادیر رشته‌ای مانند کاراکتر یا کلمه داشته باشد.

مثال ۷.۱. در مثال ۲.۱ ویژگی‌های قد، وزن و کد ملی عددی و ویژگی‌های جنسیت و محل تولد رشته‌ای هستند.

در ادامه فرض می‌کنیم که بردار ویژگی متشکل از مقادیر عددی است. به این منظور ضروری است که داده‌های رشته‌ای را با روش‌های انکودینگ [۶] که از حوصله این بحث خارج است، به مقادیر عددی تبدیل کنیم. حال بردار ویژگی آماده پردازش‌های لازم برای ساخت مدل است.

۲.۱.۱ استخراج و انتخاب ویژگی

تعریف ۸.۱. در یادگیری ماشینی، تشخیص الگو و پردازش تصویر، استخراج ویژگی از مجموعه اولیه داده‌های اندازه‌گیری شده شروع می‌شود و مقادیر مشتق شده ویژگی‌ها را ایجاد می‌کند.

به طور کلی، استخراج ویژگی شامل کاهش تعداد منابع مورد نیاز برای توصیف مجموعه بزرگی از داده‌ها است. روش‌های متنوعی برای استخراج ویژگی نظیر تحلیل مؤلفه اصلی [۷] و تحلیل تشخیصی خطی [۸] است که بررسی آن از حوصله این بحث خارج است.

[5] Feature Vector
[6] Encoding
[7] Principal Component Analysis
[8] Linear Discriminant Analysis

تعریف ۹.۱. انتخاب ویژگی فرآیند انتخاب سازگارترین، غیر زائدترین و مرتبط‌ترین ویژگی‌ها برای استفاده در ساخت مدل است که هدف اصلی آن بهبود عملکرد یک مدل و کاهش هزینه محاسباتی مدل‌سازی است.

مثال ۱۰.۱. فرض کنید داده خام اولیه مجموعه‌ای از تصاویر است و بدیهی است که این مجموعه داده به همین شکل قابل استفاده نیست. در نتیجه، ویژگی‌هایی نظیر میزان روشنایی، ساختار هندسی و ... از تصاویر استخراج می‌شود و برای مرحله انتخاب ویژگی فرستاده می‌شود. در این مرحله ویژگی‌های که در تعریف ۹.۱ صدق می‌کنند، برای ساخت مدل انتخاب می‌شوند.

۲.۱ اهمیت انتخاب ویژگی

انتخاب ویژگی نقش حیاتی در یادگیری ماشین و مدل‌سازی دارد. در زیر به چند نمونه از اهمیت‌های آن می‌پردازیم:

۱. به دست آوردن ویژگی‌ها مستلزم هزینه حافظه‌ای و زمانی است، بنابراین انتخاب ویژگی مفید است.

۲. انتخاب ویژگی، به بهبود دقت مدل کمک می‌کند.

۳. انتخاب ویژگی، زمان مورد نیاز مدل برای آموزش خود را کاهش می‌دهد.

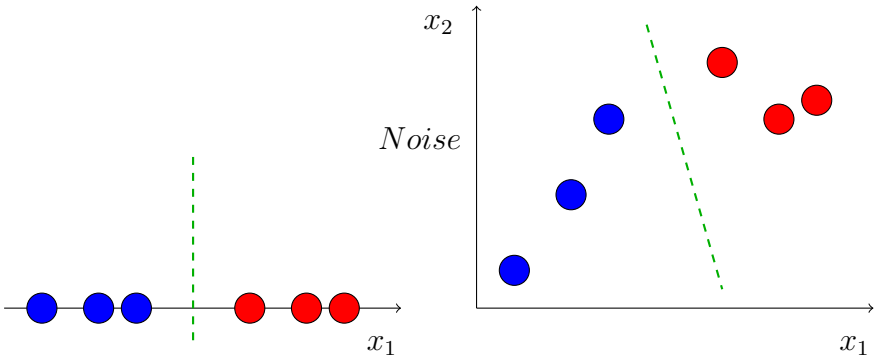
۴. انتخاب ویژگی، داده‌های نامرتبط [۹] و نویز [۱۰] را کنار می‌گذارد.

مثال ۱۱.۱. در شکل ۱.۱ دو کلاس آبی و قرمز را از هم جدا کنیم، همان‌طور که مشاهده می‌شود به علت وجود نویز در ویژگی x_2 دقت مرز تصمیم‌گیری [۱۱] دو کلاس کاهش خواهد یافت.

۳.۱ روش‌های انتخاب ویژگی

به طور کلی می‌توان از دو دیدگاه روش‌های انتخاب ویژگی را دسته‌بندی کرد. در نتیجه، در دیدگاه اول بر اساس تعداد ویژگی‌هایی که در یک لحظه بررسی می‌شوند و در دیدگاه دوم بر اساس نحوه ارزیابی ویژگی‌ها، تقسیم بندی را انجام می‌دهیم [۲].

[9] Irrelevant
[10] Noise
[11] Decision Boundary



شکل ۱.۱: تأثیر نویز بر دقت مدل [۱]

۱.۳.۱ انتخاب ویژگی بر اساس تعداد

در این نوع دسته بندی، انتخاب ویژگی بر اساس تعداد ویژگی در لحظه انجام می‌شود که مبتنی بر دو نوع است:

تعریف ۱.۱۲. روش‌های تک متغیره [۱۲] به روش‌هایی گفته می‌شود که در هر لحظه یک ویژگی را در نظر می‌گیرد و به بررسی آن می‌پردازد.

تعریف ۱.۱۳. روش‌های چند متغیره [۱۳] به روش‌هایی گفته می‌شود که در هر لحظه زیرمجموعه‌ای از ویژگی‌ها را در نظر می‌گیرد و به بررسی آن می‌پردازد.

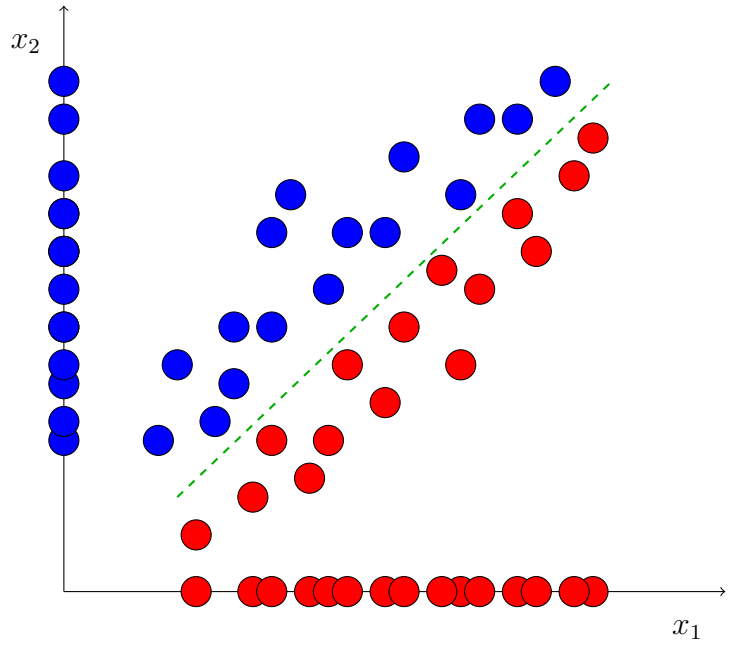
۱.۱.۳.۱ مزایا

مزیت روش‌های تک متغیره سرعت بالای آن‌ها در انتخاب ویژگی‌های مطلوب است. در نتیجه به علت سرعت بالای آن‌ها در انتخاب ویژگی به طور گسترده مورد استفاده قرار می‌گیرند.
روش‌های چند متغیره هنگامی که یک ویژگی به تنهایی تفکیک پذیری مناسبی ندارد ولی دسته‌ای از ویژگی‌ها به خوبی قابلیت تفکیک پذیری دارند، خود را نشان می‌دهند.

۲.۱.۳.۱ معایب

در مقابل تفکیک پذیری مناسب روش‌های چند متغیره، روش‌های تک متغیره در انتخاب ویژگی‌هایی که به کمک یکدیگر باعث جدایی کلاس‌ها می‌شوند، شکست می‌خورند.

[12] Uni-variate Methods
[13] Multi-variate Methods



شکل ۲.۱: تأثیر چند ویژگی بر دقت مدل [۱]

در مقابل سرعت بالای روش‌های تک متغیره، روش‌های چند متغیره به علت درنظر گرفتن مجموعه‌ای از ویژگی‌ها که به لحاظ محاسباتی هزینه سنگینی از $\mathcal{O}(۲^n)$ دارد.

مثال ۱.۱۴. در شکل ۲.۱ قصد داریم که دو کلاس آبی و قرمز را از هم جدا کنیم، اما هیچ کدام از دو ویژگی x_1 و x_2 به تنهایی تفکیک پذیری مناسبی نمی‌دهند. این در حالی است که دو ویژگی در کنار هم باعث جداسازی دو کلاس می‌شود و به صورت خطی جدا پذیر [۱۴] است.

۲.۳.۱ انتخاب ویژگی بر اساس ارزیابی

در این نوع دسته بندی، انتخاب ویژگی بر اساس نحوه ارزیابی آن انجام می‌شود که مبتنی بر سه نوع است:

تعریف ۱.۱۵. روش‌های فیلتر [۱۵] به روش‌هایی گفته می‌شود که به رتبه بندی یک ویژگی یا مجموعه‌ای از ویژگی‌ها می‌پردازد که این فرآیند مستقل از مدل است.

[14] Linearly Separable
[15] Filter Methods

تعریف ۱.۱۶. روش‌های رَپِر [16] به روش‌هایی گفته می‌شود که ارزیابی و رتبه دهی یک ویژگی یا مجموعه‌ای از ویژگی‌ها را به کمک یک مدل انجام می‌دهد.

تعریف ۱.۱۷. روش‌های اِمبِدد [17] به روش‌هایی گفته می‌شود که به انتخاب ویژگی در زمان آموزش مدل می‌پردازد و معمولاً با اضافه کردن یک جمله منظم ساز [18] به مدل قابل انجام است.

۱.۲.۳.۱ مزایا

روش‌های فیلتر به علت سادگی و مستقل بودن از دسته‌بند [19] سرعت بالایی در انتخاب ویژگی‌ها دارد. در نتیجه، قابلیت تعمیم در این دسته روش‌ها بیشتر از روش رپر بوده و ریسک بیش برازش [20] را کمتر می‌کند.
هم‌چنین روش‌های رپر از میزان دقت بالایی در انتخاب ویژگی برخوردار هستند؛ چراکه ویژگی‌هایی انتخاب می‌شوند که مدل بر روی آن‌ها دقت بالاتری دارد.

۲.۲.۳.۱ معایب

از آن‌جایی که روش‌های رپر وابسته به دسته‌بند هستند، بایستی برای هر ارزیابی، مدل آموزش داده شود که بار محاسباتی سنگینی به برنامه تحمیل می‌کند. از طرف دیگر احتمال بیش برازش در رپرها بیشتر است؛ زیرا بر اساس مطلوبیتِ مدل انتخاب می‌شوند.
در طرفِ مقابل، روش‌های فیلتر به علت آن که عموماً وابستگی بین ویژگی‌ها را نادیده گرفته و هر کدام را جداگانه ارزیابی می‌کنند، ضعف دارند که مثال ۲.۱.۳.۱ نیز نمونه‌ای از معایب آن است.

۳.۳.۱ به کار گیری روش‌های انتخاب ویژگی

در این جا قصد داریم با روش‌های مختلف و الگوریتم‌های هر کدام آشنا شویم و بررسی کنیم که در هر روش کدام ویژگی‌ها انتخاب می‌شوند و کدام کنار گذاشته می‌شوند.

تذکر ۱.۱۸. بیشتر روش‌های تک متغیره، از نوع فیلتر هستند.

تذکر ۱.۱۹. بیشتر روش‌های رپر، چند متغیره هستند.

[16] Wrapper Methods
[17] Embedded Methods
[18] Regularization
[19] Classifier
[20] Over-fitting

۱.۳.۳.۱ روش‌های فیلتر تک متغیره

الگوریتم ۲۰.۱. در این روش، تمام ویژگی‌ها را بر اساس یک معیار ارزیابی رتبه بندی می‌کنیم و به عنوان خروجی لیستی از ویژگی‌ها را به ترتیب رتبه به صورت نزولی بازمی‌گردانیم. در نتیجه، کاربر k ویژگی برتر را برمی‌گزیند.

تذکر ۲۱.۱. معیارهای ارزیابی متنوعی برای این روش وجود دارد که در ادامه به برخی از آن‌ها می‌پردازیم [۳].

تعریف ۲۲.۱. معیار همبستگی پیرسون [۲۱] به صورت زیر تعریف می‌شود:

$$R(k) = \frac{cov(X_k, Y)}{\sqrt{var(X_k)}.\sqrt{var(Y)}}$$

که X_k ویژگی kـام و Y برچسب [۲۲] کلاس مربوطه است.

نتیجه ۲۳.۱. طبق بحث‌های آماری و با توجه به رابطه همبستگی پیرسون،این معیار برای ویژگی‌هایی که رابطه آن‌ها با برچسب کلاس خطی نیستند، اصلاً مناسب نیست. در نتیجه، این معیار ویژگی‌هایی را که همبستگی خطی قوی با برچسب کلاس دارند، انتخاب می‌کنند.

تذکر ۲۴.۱. توجه به این نکته ضروری است که ممکن است یک ویژگی ارتباط غیرخطی (مانند سهمی) با برچسب کلاس داشته باشد. در این حالت معیار پیرسون همبستگی کمی را نشان خواهد داد و لذا همواره نمی‌توان به معیار پیرسون اتکا کرد.

تعریف ۲۵.۱. دو ویژگی X و Y را مستقل [۲۳] گوییم هر گاه:

$$P(X, Y) = P(X).P(Y)$$

تعریف ۲۶.۱. اطلاع متقابل [۲۴] دو متغیر به صورت زیر تعریف می‌شود:

$$MI(X, Y) = E_{X,Y}\left[log\frac{P(X, Y)}{P(X).P(Y)}\right]$$

تذکر ۲۷.۱. حال با توجه به تعریف فوق می‌توانیم ویژگی‌ها را بر اساس اطلاع مشترک مرتب کنیم و در حالتی که دو متغیر مستقل باشند، طبق رابطه فوق داریم:

$$MI(X, Y) = E_{X,Y}\left[log\frac{P(X).P(Y)}{P(X).P(Y)}\right] = E_{X,Y}[log1] = 0$$

لذا بر اساس رابطه فوق، ویژگی پایین‌ترین (بدترین) عملکرد را خواهد داشت و باید کنار گذاشته شود.

[21] Pearson Correlation Criteria
[22] Label
[23] Independent
[24] Mutual Information

۲.۳.۳.۱ روش‌های رپر چند متغیره

الگوریتم ۲۸.۱. در این روش، زیرمجموعه‌ای از ویژگی‌ها را به کمک الگوریتم‌های جستجو انتخاب کرده و سپس به آموزش دسته‌بند پرداخته و دقت آن را ذخیره می‌کنیم. با تکرار این عمل بهترین زیرمجموعه را انتخاب می‌کنیم.

تذکر ۲۹.۱. همان‌طور که در بخش ۲.۲.۳.۱ بررسی شد، روش‌های رپر چند متغیره بار محاسباتی زیادی دارند که نیاز به الگوریتم‌های فرامکاشفه‌ای را روشن‌تر خواهد کرد.

۴.۳.۱ چالش‌ها

همان‌طور که در بخش ۲.۱.۳.۱ و ۲.۲.۳.۱ معایب هر کدام از روش‌های انتخاب ویژگی را بررسی کردیم، روشن خواهد شد که هیچ روشی بر دیگری ارجح نیست و به پارامترهای دیگری نظیر جنس مجموعه داده، توزیع [۲۵] داده‌ها و نوع مدل و ... وابسته است. در نتیجه، انتخاب فیلتر یا رپر و تک متغیره یا چند متغیره به ذات مسئله وابسته است.

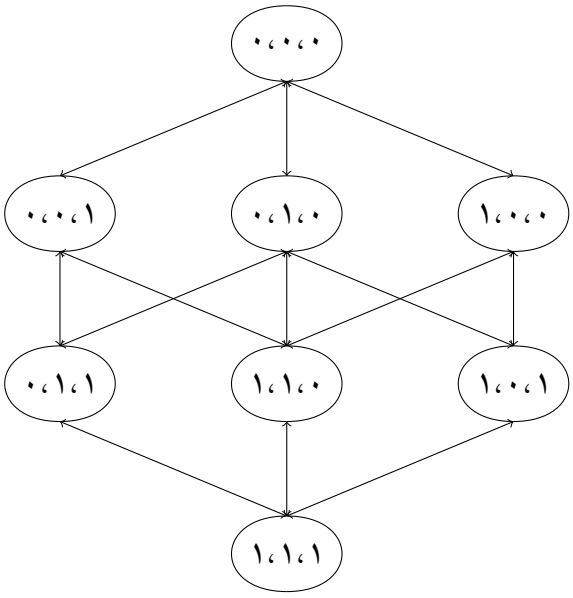
مثال ۳۰.۱. فرض کنید که مطابق شکل ۳.۱ یک مسئله به طور کلی شامل ۳ ویژگی است؛ لذا تعداد زیر مجموعه‌های ناتهی برابر ۷ است. شاید انتخاب روش چند متغیره و اعمال رپر برای این مسئله به علت کوچکی فضای ویژگی و امکان بررسی تمام حالات ممکن مناسب‌تر است. این در حالی است که جستجوی کامل فضا برای یک مجموعه داده با ۱۰۰ ویژگی ناممکن است. پس ضروری است که ابتدا با ابعاد، روابط ویژگی‌ها و ... آشنا شویم.

۴.۱ الگوریتم‌های فرامکاشفه‌ای

همان‌طور که در بخش ۴.۳.۱ اشاره شد، روش‌های رپر نیاز به الگوریتم‌های جستجو برای پیمایش فضای بزرگ پاسخ دارند؛ چرا که بررسی تمام فضا از لحاظ محاسباتی ممکن نیست. پس الگوریتم‌های کلاسیک و دقیق، یافتن راه حل بهینه را تضمین می‌کنند اما مشکلی که در این میان وجود دارد این است که این الگوریتم‌ها برای مسائل سخت، کارایی ندارند و زمان یافتن راه حل برای مسائل سخت به صورت نمایی افزایش خواهد یافت. در نتیجه برای مسائل سخت [۲۶] الگوریتم‌های کلاسیک راضی کننده نیستند. لذا وجود و بهره گیری از الگوریتم‌های فرامکاشفه‌ای احساس خواهد شد.

تعریف ۳۱.۱. الگوریتم‌های فرامکاشفه‌ای [۲۷] پارادایم‌های هوش محاسباتی هستند که به ویژه برای حل مسائل بهینه سازی پیچیده مورد استفاده قرار می گیرند [۴].

[25] Distribution
[26] NP-Hard Problems
[27] Meta-heuristic Algorithms



شکل ۱.۳: فضای جستجو برای حالتی که ۳ ویژگی داشته باشیم [۱].

۱.۴.۱ دسته بندی الگوریتم‌های فرامکاشفه‌ای

الگوریتم‌های فرامکاشفه‌ای شامل دسته‌ها و انواع مختلفی هستند. برخی از مهم‌ترین آن‌ها عبارت هستند از:

۱. الگوریتم ژنتیک [۲۸]

۲. الگوریتم بهینه‌سازی ازدحام ذرات [۲۹]

۳. الگوریتم تبرید شبیه سازی شده [۳۰]

۴. الگوریتم بهینه سازی کلونی مورچه [۳۱]

در ادامه قصد داریم که بر روی الگوریتم ژنتیک تمرکز کرده و فرآیند آن را درک کنیم. الگوریتم‌های ژنتیک معمولاً برای تولید راه‌حل‌هایی با کیفیت بالا برای مسائل بهینه‌سازی و جستجو با تکیه بر عملگرهای الهام‌گرفته از بیولوژی استفاده می‌شوند. پیش از بیان مراحل آن، به تعاریف زیر می‌پردازیم:

[28] Genetic Algorithm
[29] Particle Swarm Optimization
[30] Simulated Annealing
[31] Ant Colony Optimization

تعریف ۳۲.۱. الگوریتم ژنتیک که بر اساس نظریه انتخاب طبیعی [۳۲] داروین است بیان می‌کند که نمونه‌ها و افراد اصلح زنده می‌مانند و نمونه‌های ضعیف حذف خواهند شد [۵].

تعریف ۳۳.۱. کروموزم [۳۳] محل ذخیره سازی اطلاعات ژنی هستند و از واحدهای کوچکی به نام ژن [۳۴] تشکیل شده‌اند.

تعریف ۳۴.۱. یک جمعیت [۳۵] از تعداد مشخصی کروموزم تشکیل شده است.

تعریف ۳۵.۱. برازش [۳۶] یک موجود میزان شایستگی او در جمعیتی است که در آن حضور دارد.

تعریف ۳۶.۱. انتخاب [۳۷] کروموزم‌های برتر به معنای انتخاب زیرمجموعه‌ای از نمونه‌ها برای تولید مثل در نسل بعد است که همان قانون بقای اصلح داروین را بیان می‌کند.

تعریف ۳۷.۱. بازترکیب [۳۸] جمعیت انتخاب شده همان تولید مثل این جمعیت است که با ترکیب ژن‌های دو یا چند والد انجام پذیر است.

تعریف ۳۸.۱. جهش [۳۹] تغییر تصادفی ژن‌های یک کروموزم است.

تعریف ۳۹.۱. انتخاب برای جایگزینی [۴۰] تولید یک جمعیت به عنوان نسل جدید از والدین قبلی و فرزندان است.

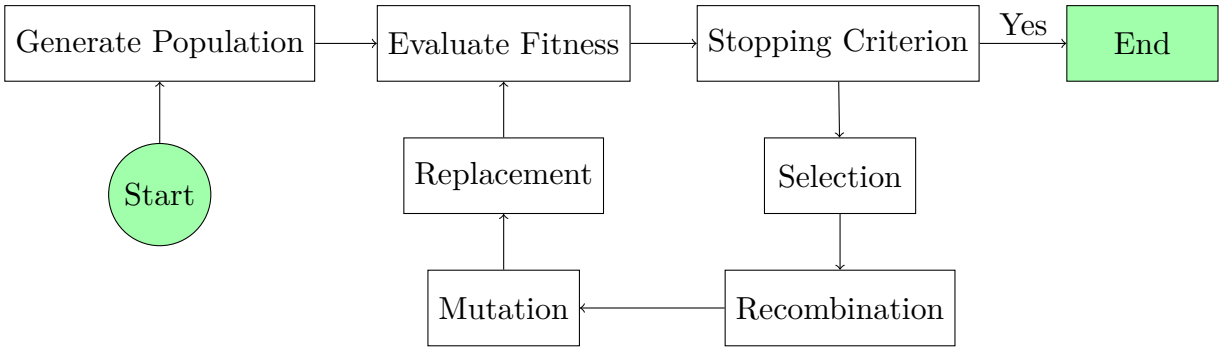
مراحل الگوریتم ژنتیک در شکل ۴.۱ رسم شده است:

۲.۴.۱ الگوریتم‌های فرامکاشفه‌ای در انتخاب ویژگی

حال که با الگوریتم ژنتیک به خوبی آشنا شدیم، به بهره‌گیری از آن در حل مسائل یادگیری ماشین به کمک انتخاب ویژگی می‌پردازیم. برای استفاده از الگوریتم ژنتیک ضروری است که با مشخص کردن کروموزم‌ها و تابع برازش، مسئله را مدل سازی کنیم. پس داریم:

- کروموزم‌ها همان داده‌های ما هستند که به تعداد m ویژگی دارند.
- تابع برازش همان دسته‌بندی است که برای ارزیابی ویژگی‌ها و به دست آوردن دقت آن به کار می‌گیریم.

[32] Natural Selection
[33] Chromosome
[34] Gene
[35] Population
[36] Fitness
[37] Selection
[38] Recombination
[39] Mutation
[40] Replacement



شکل ۴.۱: مراحل اجرای الگوریتم ژنتیک

۳.۴.۱ چالش‌ها

از آن‌جایی که این الگوریتم‌ها به نوعی تصادفی هستند، ممکن است تعداد تکرارهای زیادی لازم باشد تا بتوانیم به یک جواب بهینه دست یابیم. با این حال، چالش عمده الگوریتم‌های فرامکاشفه‌ای تنظیم تعداد بسیار زیاد پارامترهای آن‌ها است. برخی از این پارامترها عبارت هستند از:

- ❄ به منظور انتخاب کروموزم‌های برتر، روش‌های زیادی مانند انتخاب تصادفی، نسبی، رتبه‌ای و ... وجود دارند که هر کدام با تعدادی پارامتر سروکار دارند.
- ❄ به منظور بازترکیب کروموزم‌ها، روش‌های زیادی مانند بازترکیب دودویی، نقطه‌ای، یکنواخت و ... وجود دارند.
- ❄ به منظور جهش کروموزم‌ها، روش‌های زیادی مانند معکوس سازی بیتی، جهش مکمل، جهش درجی و ... وجود دارند.
- ❄ به منظور انتخاب برای جایگزینی، روش‌های مانند جایگزینی نسلی و پایدار وجود دارند.

فصل ۲

آزمایش‌ها و نتایج

در این فصل قصد داریم که به بررسی یک نمونه عملی بر اساس مطالب فصل پیشین بپردازیم. برای تحقق این هدف، روش‌های انتخاب ویژگی را برای یک مجموعه داده حقیقی (و نه ساختگی) به کار می‌گیریم. اساس و ایده کلی این فصل انتخاب یک زیرمجموعه از ویژگی‌ها و اعمال الگوریتم ژنتیک بر روی آن است. اگر چه تنظیم پارامترهای این الگوریتم کمی دشوار است و بار محاسباتی آن نیز نسبت به روش‌هایی مانند معیار پیرسون و اطلاع متقابل سنگین‌تر است، با این حال نتایج و سرعت آن به طرز چشمگیری بدون انتخاب ویژگی، بهبود پیدا خواهد کرد.
امروزه مجموعه داده‌های بسیاری توسط سازمان‌ها، ارگان‌ها و مراکز دولتی یا خصوصی در زمینه‌ها و موضوعات مختلف گردآوری شده که در فصل آینده به پاره‌ای از آن‌ها اشاره شده است. به عنوان نمونه در این جا قصد داریم که یکی از این مجموعه داده‌ها را انتخاب کرده و آموخته‌های خود را روی آن اعمال کنیم. مجموعه داده‌ای که در این فصل روی آن کار خواهیم کرد از مخزن UCI است. شما می‌توانید از طریق این پیوند به مطالعه و بررسی آن بپردازید

۱.۲ نگاهی بر مجموعه داده

این مجموعه داده به تشخیص فعالیت‌های بدن با اندازه‌گیری‌های فیزیولوژیکی پوشیدنی پرداخته است. این مجموعه داده که در سال ۲۰۱۹ جمع آوری شده است، یک مسئله دسته بندی شامل ۴ کلاس و ۴۴۸۰ نمونه با ۵۳۳ ویژگی مختلف است. در همان سال مقاله‌ای [۶] در این زمینه نوشته شد که به کمک مدل‌های یادگیری ماشین به حل این مسئله پرداختند. این مقاله تجزیه و تحلیل عمیقی از ویژگی‌های پیشنهادی برای استخراج اطلاعات از

الکتروکاردیوگرام [1] ، بیوامپدانس [2] الکتریکی قفسه سینه را انجام می‌دهد. فعالیت های مورد تجزیه و تحلیل عبارت هستند از:

- کلاس اول که شامل فعالیت خنثی است.
- کلاس دوم که عواطف عاطفی فرد برانگیخته می‌شود.
- کلاس سوم که فرد به فعالیت‌های ذهنی می‌پردازد.
- کلاس چهارم که شخص به کارهای فیزیکی و تحرکی می‌پردازد.

۲.۲ چالش‌ها

همان طور که در بخش قبل اشاره شد، این مجموعه داده شامل ۵۳۳ ویژگی یا ستون است و در نتیجه حجم بسیار سنگینی از محاسبات را می‌طلبد. همچنین به طور کلی باید توجه داشت که مسائل دنیای واقعی شامل ویژگی‌های بسیار زیادی است که این مسئله بخش کوچکی از این دنیای وسیع است. در این جا روش‌های انتخاب ویژگی می‌توانند بسیار کمک کننده باشند و با حذف ویژگی‌های زائد و نامربوط به افزایش سرعت محاسبات کمک شایانی کنند.

۳.۲ آزمایش‌ها

در این بخش به اجرای یک آزمایش و بررسی نتایج آن خواهیم پرداخت. ایده کلی برای انتخاب ویژگی در این جا، تنها به انتخاب تعداد محدودی ویژگی از میان ۵۳۳ ویژگی این مجموعه داده است. بدیهی است که انجام این کار برای بیش از ۴۰۰۰ داده به کاهش محاسبات شگرفی ختم خواهد شد.

۱.۳.۲ انتخاب ویژگی با الگوریتم‌های فراابتکاری

ابتدا به کمک الگوریتم ژنتیک، ویژگی‌های برتر که نقش اصلی را در کلاس بندی ایفا می‌کنند می‌یابیم. به این منظور و برای انتخاب تابع برازش می‌توانیم از مدل‌های مختلف یادگیری ماشین نظیر جنگل رندوم [3] ، ماشین بردار پشتیبان [4] و ... استفاده کنیم که در ادامه به بررسی دقیق آن‌ها خواهیم پرداخت.

[1] Electrocardiogram
[2] Bio-impedance
[3] Random Forest
[4] Support Vector Machine

۱.۱.۳.۲ الگوریتم جنگل رندوم

مطابق جدول زیر، در الگوریتم جنگل رندوم دو ویژگی ۲۷ و ۴۰۰ ویژگی‌های مناسبی هستند:

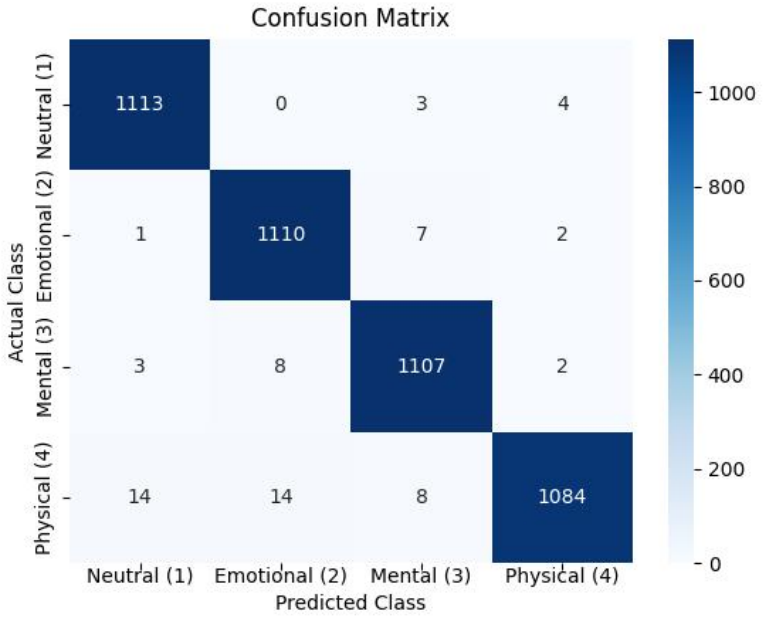
جدول ۱.۲: پارامترها و دقت الگوریتم ژنتیک در مدل جنگل رندوم

Random Forest			
Iteration	Best Features		Accuracy %
	Feature 1	Feature 2	
1	45	356	79.91
2	49	356	86.09
3	520	356	86.53
4	520	356	86.53
5	520	356	86.53
6	520	356	86.53
7	187	356	92.54
8	187	356	92.54
9	187	356	92.54
10	187	356	92.54
11	400	27	98.44
12	400	27	98.44
13	400	27	98.44
14	400	27	98.44
15	400	27	98.44
16	400	27	98.44
17	400	27	98.44
18	400	27	98.44
19	400	27	98.44
20	400	27	98.44

Parameters
Criterion = Gini
Number of Estimators = 100
Depth = None
Mutation = 70 %
Number of Features = 2
Replacement = 80 %
Number of Chromosome = 8
Recombination = 80 %
Rank = 50 %

مطابق جدول فوق سمت راست، معیار ارزیابی جنگل رندوم جینی، تعداد درخت‌ها ۱۰۰ و عمق درخت تا زمانی که به برگ نرسیده است، ادامه می‌یابد. برای پارامترهای الگوریتم ژنتیک نیز نرخ جهش برابر ۷۰ درصد، تعداد ویژگی‌ها برابر ۲، نرخ جایگزینی برابر ۸۰ درصد، اندازه جمعیت ۸، نرخ بازترکیب ۸۰ است و رتبه بندی نیز برای ۵۰ درصد برتر اعمال می‌شود.

در ماتریس درهم‌ریختگی [5] زیر خلاصه نتایج قرار گرفته است:



شکل ۱.۲: ماتریس درهم‌ریختگی الگوریتم جنگل رندوم

همان‌طور که در جدول فوق روشن است، الگوریتم به خوبی توانسته ۴ کلاس را از یک دیگر تفکیک کند که این بیان گر دقت بالای الگوریتم و انتخاب ویژگی‌های مناسب در این مجموعه داده است.

[5] Confusion Matrix

۲.۱.۳.۲ الگوریتم درخت تصمیم

مطابق جدول زیر، در الگوریتم درخت تصمیم سه ویژگی ۴۳۹ و ۲۲۸ و ۲۶۱ ویژگی‌های مناسبی هستند: مطابق

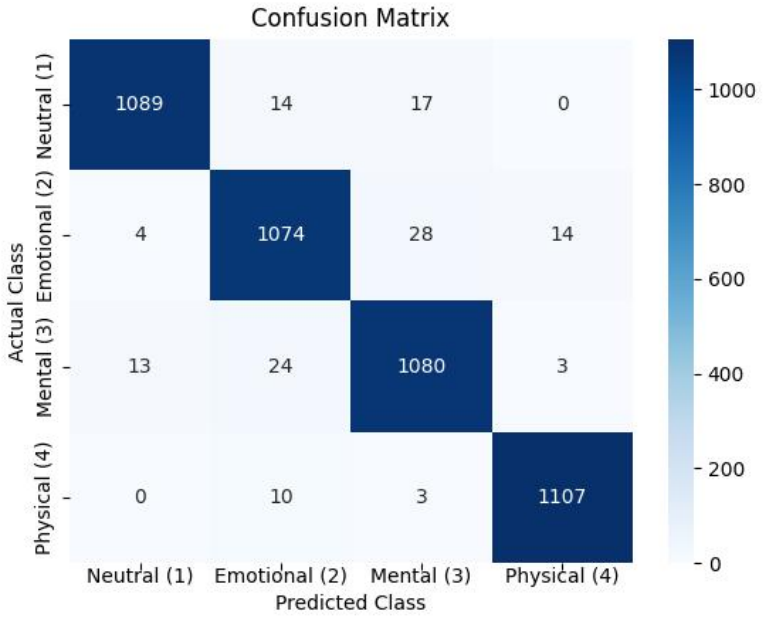
جدول ۲.۲: پارامترها و دقت الگوریتم ژنتیک در مدل درخت تصمیم

Decision Tree				
Iteration	Best Features			Accuracy %
	Feature 1	Feature 2	Feature 3	
1	309	466	374	88.71
2	309	466	374	88.71
3	309	466	374	88.71
4	309	228	261	96.54
5	309	228	261	96.54
6	309	228	261	96.74
7	309	228	261	96.74
8	16	228	261	96.88
9	16	228	261	96.92
10	16	228	261	96.92
11	16	228	261	96.92
12	439	228	261	96.92
13	439	228	261	96.99
14	439	228	261	96.99
15	411	228	261	97.1
16	411	228	261	97.1
17	411	228	261	97.1
18	187	228	261	97.1
19	439	228	261	97.1
20	439	228	261	97.1

Parameters
Criterion = Gini
Splitter = Best
Max Depth = None
Mutation = 60 %
Number of Features = 3
Replacement = 60 %
Number of Chromosome = 8
Recombination = 80 %
Rank = 50 %

جدول فوق سمت راست، معیار ارزیابی درخت تصمیم جینی، نحوه تقسیم بهترین نوع آن و عمق درخت تا زمانی که به برگ نرسیده است، ادامه می‌یابد. برای پارامترهای الگوریتم ژنتیک نیز نرخ جهش برابر ۶۰ درصد، تعداد ویژگی‌ها برابر ۳، نرخ جایگزینی برابر ۶۰ درصد، اندازه جمعیت ۸، نرخ بازترکیب ۸۰ است و رتبه بندی نیز برای ۵۰ درصد برتر اعمال می‌شود.

در ماتریس درهم‌ریختگی زیر خلاصه نتایج قرار گرفته است:



شکل ۲.۲: ماتریس درهم‌ریختگی الگوریتم درخت تصمیم

مطابق ماتریس درهم‌ریختگی فوق، الگوریتم درخت تصمیم بسیار عالی توانسته که تمام کلاس‌ها را از هم تفکیک کند.

۳.۱.۳.۲ الگوریتم XGBoost

مطابق جدول زیر، در این الگوریتم دو ویژگی ۲۱۷ و ۳۶۶ ویژگی‌های مناسبی هستند: مطابق جدول فوق سمت

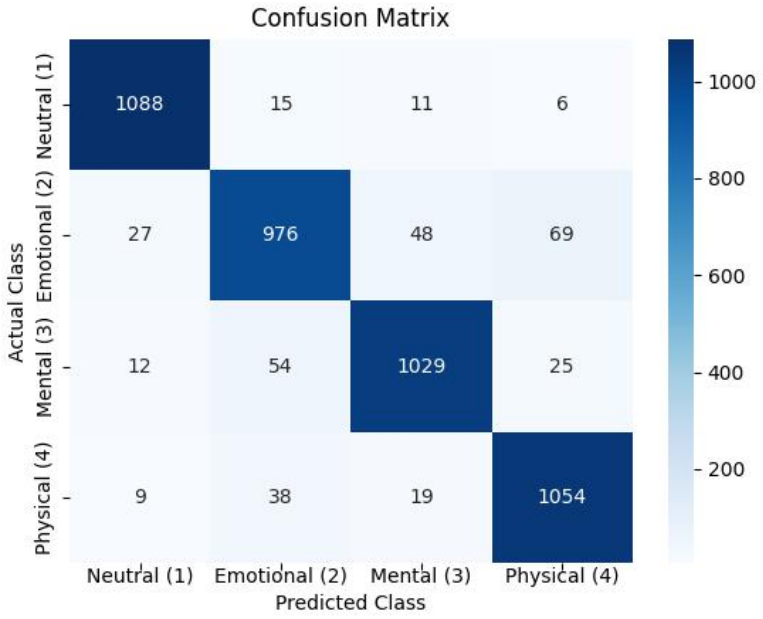
جدول ۳.۲: پارامترها و دقت الگوریتم ژنتیک در مدل XGBoost

XGBoost			
Iteration	Best Features		Accuracy %
	Feature 1	Feature 2	
1	35	227	81.45
2	35	227	81.45
3	370	436	83.97
4	370	436	83.97
5	370	436	83.97
6	370	436	83.97
7	370	436	83.97
8	370	366	83.97
9	370	366	83.97
10	264	366	87.52
11	264	366	87.52
12	264	366	87.52
13	503	366	91.5
14	503	366	91.5
15	503	366	91.5
16	217	366	92.57
17	465	366	92.57
18	217	366	92.57
19	465	366	92.57
20	217	366	92.57

Parameters
Number of Estimators = 100
Learning Rate = 0.1
Lambda = 1
Gamma = 0
Objective = multi:softmax
Eval Metric = mlogloss
Depth = None
Mutation = 90 %
Number of Features = 2
Replacement = 60 %
Number of Chromosome = 8
Recombination = 60 %
Rank = 50 %

راست، ابتدا پارامترهای این الگوریتم و سپس پارامترهای الگوریتم ژنتیک قرار دارد. نرخ جهش برابر ۹۰ درصد، تعداد ویژگی‌ها برابر ۲، نرخ جایگزینی برابر ۶۰ درصد، اندازه جمعیت ۸، نرخ بازترکیب ۶۰ است و رتبه بندی نیز برای ۵۰ درصد برتر اعمال می‌شود.

در ماتریس درهم‌ریختگی زیر خلاصه نتایج قرار گرفته است:



شکل ۳.۲: ماتریس درهم‌ریختگی الگوریتم XGBoost

مطابق ماتریس درهم‌ریختگی فوق، الگوریتم درخت تصمیم به خوبی توانسته که تمام کلاس‌ها را از هم تفکیک کند.

۴.۱.۳.۲ الگوریتم ماشین بردار پشتیبان

مطابق جدول زیر، در الگوریتم ماشین بردار پشتیبان سه ویژگی ۴۳۹ و ۲۹۴ و ۶۶ ویژگی‌های مناسبی هستند: مطابق

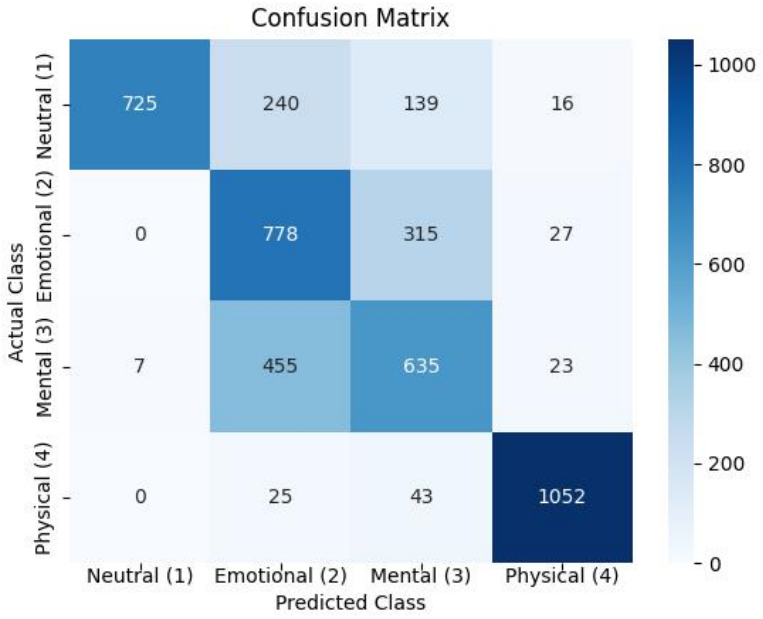
جدول ۴.۲: پارامترها و دقت الگوریتم ژنتیک در مدل ماشین بردار پشتیبان

Support Vector Machine				
Iteration	Best Features			Accuracy %
	Feature 1	Feature 2	Feature 3	
1	104	127	439	39.15
2	268	175	347	40.45
3	235	178	8	46.88
4	235	178	8	46.88
5	235	371	8	52.57
6	16	371	8	53.5
7	16	240	8	64.2
8	16	240	8	64.2
9	16	240	8	64.2
10	16	240	8	64.2
11	16	240	8	64.2
12	439	240	66	67.1
13	439	240	66	67.1
14	439	240	66	67.1
15	411	240	66	67.12
16	411	240	66	67.12
17	411	240	66	67.12
18	187	240	66	68.42
19	439	294	66	71.21
20	439	294	66	71.21

Parameters
C = 1
Kernel = RBF
Mutation = 80 %
Number of Features = 3
Replacement = 80 %
Number of Chromosome = 8
Recombination = 80 %
Rank = 50 %

جدول فوق سمت راست، پارامتر منظم ساز ۱ و هسته ماشین بردار پشتیبان RBF است. برای پارامترهای الگوریتم ژنتیک نیز نرخ جهش برابر ۸۰ درصد، تعداد ویژگی‌ها برابر ۳، نرخ جایگزینی برابر ۸۰ درصد، اندازه جمعیت ۸، نرخ بازترکیب ۸۰ است و رتبه بندی نیز برای ۵۰ درصد برتر اعمال می‌شود. مطابق این نتایج، می‌توان استدلال کرد که داده‌ها با این ۳ ویژگی به خوبی جداپذیر نیستند. با این حال، دقت ۷۱ درصد برای یک دسته بندی شامل ۴ کلاس، دقت مناسبی است.

در ماتریس درهم‌ریختگی زیر خلاصه نتایج قرار گرفته است:



شکل ۴.۲: ماتریس درهم‌ریختگی الگوریتم ماشین بردار پشتیبان

مطابق ماتریس درهم‌ریختگی فوق، الگوریتم بسیار عالی توانسته که داده‌های دو کلاس ۱ و ۴ را از هم تفکیک کند، اما متاسفانه در تشخیص و تمایز بین دو کلاس ۲ و ۳ کمی ضعیف عمل کرده است.

۵.۱.۳.۲ الگوریتم رگرسیون لجستیک

مطابق جدول زیر، در الگوریتم رگرسیون لجستیک [۶] سه ویژگی ۷۸ و ۳۱۸ و ۷۵ ویژگی‌های مناسبی هستند: مطابق

جدول ۵.۲: پارامترها و دقت الگوریتم ژنتیک در مدل رگرسیون لجستیک

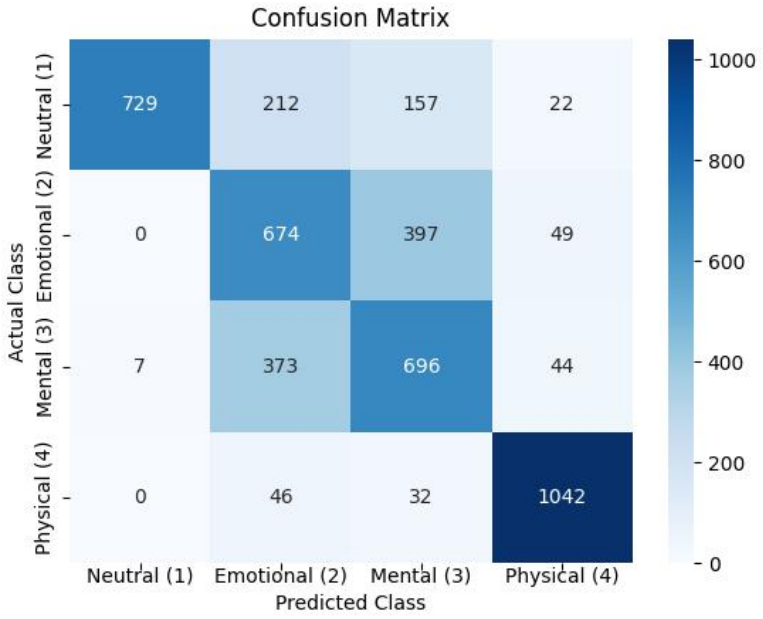
Logistic Regression				
Iteration	Best Features			Accuracy %
	Feature 1	Feature 2	Feature 3	
1	278	49	76	64.44
2	278	276	76	64.98
3	278	57	76	65.4
4	230	318	76	67.9
5	322	129	76	68.42
6	322	129	76	68.42
7	322	129	76	68.42
8	322	129	76	68.42
9	322	129	76	68.42
10	318	233	76	68.95
11	318	233	76	68.95
12	318	233	76	68.95
13	26	318	75	69.6
14	26	318	75	69.6
15	26	318	75	69.6
16	142	318	75	69.67
17	78	318	75	70.11
18	78	318	75	70.11
19	78	318	75	70.11
20	78	318	75	70.11

Parameters
Solver = lbfgs
Max Iteration = 15000
Mutation = 80 %
Number of Features = 3
Replacement = 60 %
Number of Chromosome = 32
Recombination = 60 %
Rank = 50 %

جدول فوق سمت راست، ابتدا پارامترهای این الگوریتم و سپس پارامترهای الگوریتم ژنتیک قرار دارد. نرخ جهش برابر ۸۰ درصد، تعداد ویژگی‌ها برابر ۳، نرخ جایگزینی برابر ۶۰ درصد، اندازه جمعیت ۳۲، نرخ بازترکیب ۶۰ است و رتبه بندی نیز برای ۵۰ درصد برتر اعمال می‌شود.

[6] Logistic Regression

در ماتریس درهم‌ریختگی زیر خلاصه نتایج قرار گرفته است:



شکل ۵.۲: ماتریس درهم‌ریختگی الگوریتم رگرسیون لجستیک

مطابق ماتریس درهم‌ریختگی فوق، الگوریتم رگرسیون لجستیک مشابه ماشین بردار پشتیبان در تشخیص دو کلاس ۲ و ۳ ضعف دارد.

۶.۱.۳.۲ الگوریتم بیز ساده

مطابق جدول زیر، در الگوریتم بیز ساده [۷] سه ویژگی ۶۸ و ۲۶۱ و ۲۸۱ ویژگی‌های مناسبی هستند: مطابق جدول

جدول ۶.۲: پارامترها و دقت الگوریتم ژنتیک در مدل بیز ساده

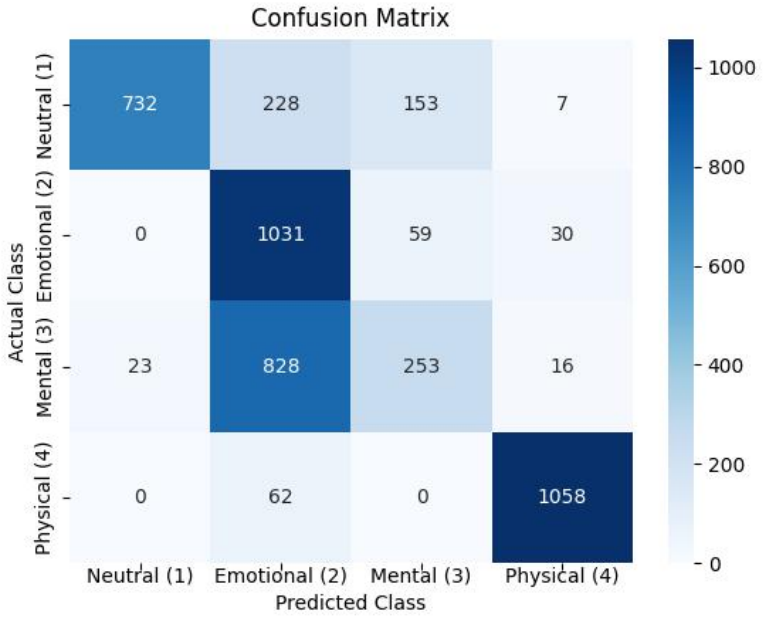
Naive Bayes				
Iteration	Best Features			Accuracy %
	Feature 1	Feature 2	Feature 3	
1	61	405	239	63.24
2	322	261	74	64.22
3	322	261	74	64.22
4	294	72	261	64.51
5	294	261	74	64.62
6	294	261	74	64.62
7	294	261	74	64.62
8	139	280	76	66.45
9	139	280	76	66.45
10	280	76	261	66.5
11	280	76	261	66.5
12	275	281	42	67.99
13	275	281	42	67.99
14	275	281	42	67.99
15	275	281	42	67.99
16	275	281	42	67.99
17	68	261	281	68.62
18	68	261	281	68.62
19	68	261	281	68.62
20	68	261	281	68.62

Parameters
Mutation = 90 %
Number of Features = 3
Replacement = 90 %
Number of Chromosome = 512
Recombination = 90 %
Rank = 50 %

فوق سمت راست، ابتدا پارامترهای این الگوریتم و سپس پارامترهای الگوریتم ژنتیک قرار دارد. نرخ جهش برابر ۹۰ درصد، تعداد ویژگی‌ها برابر ۳، نرخ جایگزینی برابر ۹۰ درصد، اندازه جمعیت ۵۱۲، نرخ بازترکیب ۹۰ است و رتبه بندی نیز برای ۵۰ درصد برتر اعمال می‌شود.

[7] Naive Bayes

در ماتریس درهم‌ریختگی زیر خلاصه نتایج قرار گرفته است:



شکل ۶.۲: ماتریس درهم‌ریختگی الگوریتم بیز ساده

مطابق ماتریس درهم‌ریختگی فوق، این الگوریتم از سایر الگوریتم‌های دیگر ضعیف‌تر عمل کرده است؛ چراکه بر روی داده‌های ۳ ویژگی انتخاب شده توزیع گوسی مفروض شدیم در حالی که ممکن است چنین نباشد.

۲.۳.۲ انتخاب ویژگی با الگوریتم‌های دیگر

در این زیربخش به بررسی معیارها و الگوریتم‌ها دیگر برای انتخاب ویژگی خواهیم پرداخت که دیدگاه فراابتکاری ندارند. بدیهی است که این روش‌ها از سرعت بالاتری نسبت به الگوریتم ژنتیک برخوردار هستند.

۱.۲.۳.۲ حذف بازگشتی ویژگی‌ها

بر اساس الگوریتم حذف بازگشتی ویژگی‌ها [۸] ، به رتبه بندی ویژگی‌ها بر اساس یک مدل خارجی خواهیم پرداخت. به بیان دقیق‌تر، هدف از حذف بازگشتی ویژگی‌ها، انتخاب ویژگی‌ها با در نظر گرفتن مجموعه‌های کوچکتر و حذف تدریجی آن‌ها تا رسیدن به یک زیرمجموعه بهینه است. لذا در ابتدا، مدل انتخابی در مورد مجموعه اولیه ویژگی‌ها آموزش می‌بیند و اهمیت هر ویژگی به دست می‌آید و سپس کم اهمیت‌ترین ویژگی‌ها از مجموعه ویژگی‌های فعلی حذف می‌شوند و این روش به صورت بازگشتی بر روی مجموعه هرس شده تکرار می‌شود تا در نهایت به تعداد مورد نظر ویژگی برای انتخاب برسیم.
با انتخاب الگوریتم درخت تصمیم به عنوان مدل خارجی و انتخاب ۲۰ ویژگی برتر، ویژگی‌های موجود در جدول ۷.۲ به عنوان مهم‌ترین ویژگی‌ها برگزیده خواهند شد. حال می‌توانیم با استفاده از یک مدل یادگیری ماشین، به آموزش و تست بپردازیم؛ چرا که از بین بیش از ۵۰۰ ویژگی، تنها ۲۰ ویژگی برتر انتخاب شده است. در نتیجه، سرعت محاسبات در این مدل به طور محسوسی کاهش خواهد یافت که این موضوع بدون استفاه از این روش انتخاب ویژگی ممکن نبود.

جدول ۷.۲: ۲۰ ویژگی برتر بر اساس الگوریتم حذف بازگشتی ویژگی‌ها و درخت تصمیم

RFE + Decision Tree				
Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
13	16	17	18	19
Feature 6	Feature 7	Feature 8	Feature 9	Feature 10
20	66	69	93	201
Feature 11	Feature 12	Feature 13	Feature 14	Feature 15
324	325	326	327	328
Feature 16	Feature 17	Feature 18	Feature 19	Feature 20
329	330	331	395	396

[8] Recursive Feature Elimination

۲.۲.۳.۲ آزمون کای دو

بر اساس آزمون کای دو [۹] وابستگی بین متغیرهای تصادفی را اندازه گیری می‌کنیم. بنابراین استفاده از این تابع، ویژگی‌هایی را که به احتمال زیاد مستقل از کلاس هستند و برای طبقه بندی نامرتبط هستند، کنار گذاشته خواهند شد. توجه شود که در این آزمون فرض می‌شود که ویژگی‌ها نامنفی هستند؛ لذا در این دادگان عمل استاندارد سازی مطابق رابطه ۱.۲ انجام می‌شود.

$$X_{new} = \frac{X - min(X)}{max(X) - min(X)} \qquad (۱.۲)$$

در این دادگان و بر اساس آزمون کای دو، ۲۰ ویژگی برتر انتخاب شده در جدول ۸.۲ قرار دارد.

جدول ۸.۲: ۲۰ ویژگی برتر بر اساس معیار کای ۲

Chi-Squared				
Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
1	6	10	21	29
Feature 6	Feature 7	Feature 8	Feature 9	Feature 10
40	42	44	45	48
Feature 11	Feature 12	Feature 13	Feature 14	Feature 15
50	51	52	53	72
Feature 16	Feature 17	Feature 18	Feature 19	Feature 20
76	232	318	321	322

۳.۲.۳.۲ معیار اطلاع متقابل

همان طور که در تعریف ۲۶.۱ و تذکر ۲۷.۱ اشاره کردیم، اطلاع متقابل بین دو متغیر تصادفی یک مقدار غیر منفی است که وابستگی بین متغیرها را اندازه گیری می‌کند. اگر دو متغیر تصادفی مستقل باشند، آنگاه این مقدار برابر با صفر است و مقادیر بالاتر به معنای وابستگی بیشتر است. حال بر این اساس ۵۰ ویژگی برتر که بیشترین وابستگی به برچسب کلاس را دارند، در جدول ۹.۲ قرار داده‌ایم.

[9] Chi-Squared Test

جدول ۹.۲: ۵۰ ویژگی برتر بر اساس معیار اطلاع متقابل

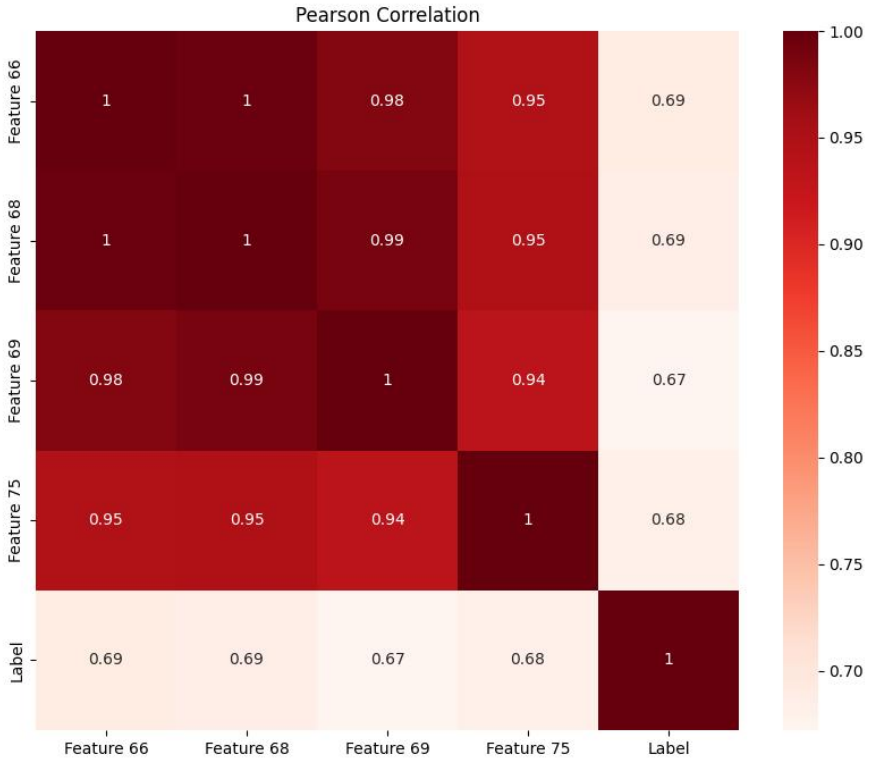
Mutual Information				
Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
6	7	27	34	35
Feature 6	Feature 7	Feature 8	Feature 9	Feature 10
41	55	180	181	195
Feature 11	Feature 12	Feature 13	Feature 14	Feature 15
196	210	211	219	233
Feature 16	Feature 17	Feature 18	Feature 19	Feature 20
327	331	332	333	338
Feature 21	Feature 22	Feature 23	Feature 24	Feature 25
345	346	355	356	359
Feature 26	Feature 27	Feature 28	Feature 29	Feature 30
360	361	366	369	373
Feature 31	Feature 32	Feature 33	Feature 34	Feature 35
374	376	380	400	414
Feature 36	Feature 37	Feature 38	Feature 39	Feature 40
428	435	436	442	449
Feature 41	Feature 42	Feature 43	Feature 44	Feature 45
450	463	464	470	477
Feature 46	Feature 47	Feature 48	Feature 49	Feature 50
478	484	504	518	532

۴.۲.۳.۲ معیار همبستگی پیرسون

حال بر اساس تعریف ۲۲.۱ به بررسی همبستگی خطی بر اساس معیار پیرسون می‌پردازیم. در این‌جا برای ویژگی‌هایی که در رابطه زیر صدق می‌کنند، ماتریس همبستگی را رسم می‌کنیم:

$$|corr(feature,\ label)| \geq 0.65 \qquad (۲.۲)$$

حال بر اساس رابطه ۲.۲ ویژگی‌های برتری را که در این شرط صدق می‌کنند، برمی‌گزینیم. بدیهی است که این ویژگی‌ها همبستگی خطی خوبی در جهت مثبت و منفی با برچسب کلاس خواهند داشت.



شکل ۷.۲: ماتریس همبستگی ویژگی‌ها و برچسب آن

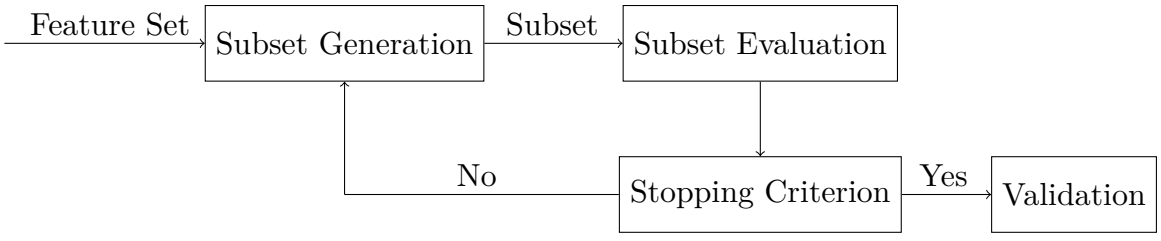
با توجه به شکل ۷.۲ در می‌یابیم که ۴ ویژگی ۶۶ و ۶۸ و ۷۵ و ۶۹ به ترتیب بهترین ویژگی‌هایی هستند که بر اساس معیار پیرسون انتخاب شده‌اند. همچنین با توجه به این شکل روشن است که همبستگی این ۴ ویژگی با یکدیگر بسیار زیاد است و به نوعی بیان می‌کند که وجود این ۴ ویژگی همزمان و در کنار هم ضروری نیست؛ لذا می‌توان ویژگی‌های زائد را کنار گذاشت.

فصل ۳

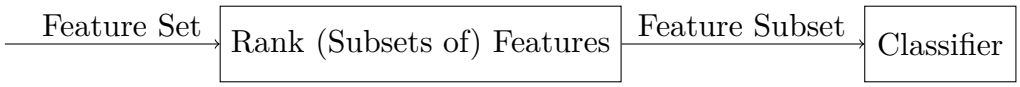
جمع بندی و پیشنهادها برای ادامه کار

۱.۳ جمع بندی

در این گردایه، ابتدا به تعریف مفاهیم اصلی در مباحث مربوط به انتخاب ویژگی پرداختیم و سپس به اهمیت‌ها و چالش‌های مربوط به آن پی بردیم. بدیهی است هر روشی که معرفی می‌شود دارای مزایا و معایب مختص به خود است؛ اما آن‌چه که اهمیت دارد این است که با بهره‌گیری از روش‌های آن بتوانیم قدمی در راستای حل مسائل و مشکلات پیرامونی برداریم. بدین منظور می‌توان روش‌های فیلتر و رپر را در این موارد به کار برد و بهترین ویژگی یا ویژگی‌های یک مجموعه داده را انتخاب کرد و در نهایت به انجام کلاس بندی یا رگرسیون بپردازیم.
اگرچه زمانی که تعداد ویژگی‌ها محدود است، انتخاب ویژگی از نظر محاسباتی چندان کمک کننده نیست، نقش این مفهوم زمانی پر رنگ می‌شود که در راستای حل مسائل دنیای واقعی که شامل صدها یا حتی هزاران ویژگی است برمی‌آییم که فصل ۲ تنها گوشه‌ای از آن را محقق کرد. در این فصل به صورت عملی به بهره‌گیری از الگوریتم‌های فرامکاشفه‌ای نظیر الگوریتم ژنتیک بر روی یک مجموعه داده واقعی از مخزن UCI پرداخته و دقت بسیار مناسبی نسبت به مقاله مربوط به آن [۶] کسب کردیم. در پایان قصد داریم، دو روش انتخاب ویژگی (رپر و فیلتر) را در دو نمودار خلاصه کنیم.
فرایند الگوریتم رپر به صورت زیر است:



فرایند الگوریتم فیلتر به صورت زیر است:



۲.۳ پیشنهادها

آنچه در این پژوهش انجام گرفت، گوشه‌ای از هزاران پژوهشی است که می‌توان در حوزه انتخاب ویژگی و به کار گیری آن به خصوص در حل مسائل سخت پیش گرفت؛ چراکه امروزه نیاز به سرعت بالای پردازش برنامه‌ها برای حل دسته وسیعی از مسائل احساس می‌شود. برخی از آن‌ها عبارت هستند از:

- مجموعه داده‌های پزشکی، رمز ارزی، اقتصادی و ... بسیاری نظیر تشخیص سرطان یا بیماری‌های قلبی، پیش بینی نرخ رمز ارزها، نتایج انتخابات و ... در سایت‌هایی نظیر مخزن UCI و $Kaggle$ در این زمینه وجود دارد.

- مسائل بهینه سازی بسیاری در دسته مسائل سخت وجود دارند که به علت بار محاسباتی نمایی آن‌ها قابل محاسبه نیستند و این مطالب می‌توانند در این حوزه ورود کنند و احتمال رسیدن به جواب بهینه را افزایش دهند.

- همان‌طور که می‌دانیم هیچ یک از این الگوریتم‌ها تضمین کننده رسیدن به جواب بهینه نیستند و در صورت غیرمحدب [1] بودن تابع هدف، به بهینه محلی همگرا خواهیم شد. در نتیجه، بهبود الگوریتم‌های فرامکاشفه‌ای و ارائه الگوریتم‌هایی جدید در این زمینه می‌توانند بهینه‌های محلی با کیفیت بهتری پیدا کنند.

[1] Non-Convex

منابع

[۱] B. BabaAli, *Feature Selection in Machine Learning*, University of Tehran, 2022

[۲] W. Liu, J. Wang, *A Brief Survey on Nature-Inspired Meta-heuristics for Feature Selection in Classification in this Decade*, IEEE 16th International Conference on Networking, Sensing and Control (ICNSC), 2019

[۳] M. Sharma and P. Kaur, *A Comprehensive Analysis of Nature-Inspired Meta Heuristic Techniques for Feature Selection Problem*, Archives of Computational Methods in Engineering, 2021

[۴] A. K. Sangaiah, M. Sheng, and Z. Zhang, *Metaheuristic Algorithms: A Comprehensive Review*, Academic Press, 2018

[۵] B. BabaAli, *Genetic Algorithm in Bio-Computing*, University of Tehran, 2020

[۶] I. Herranz, R. Pita, M. Zurera, and F. Seoane, *Activity Recognition Using Wearable Physiological Measurements: Selection of Features from a Comprehensive Literature Study*, Sensors, 2019

Abstract

Feature selection is one of the most essential and fundamental steps in creating a system based on machine learning and pattern recognition. Feature selection is the process of selecting a subset of related features for building a model. The premise of feature selection methods is that the data have redundant or irrelevant properties; thus, removing these properties will not only remove important information, but will also increase the generalization of the model.

A feature selection algorithm can be considered as a search to suggest a subset of features along with an evaluation criterion that scores different subsets of features. The simplest solution is to consider all possible subsets that minimize the error rate. It is a comprehensive search of space that is computationally unsolvable for many feature sets. As a result, in this project, we intend to first get acquainted with the problem of feature selection, importance, challenges, applications, and classical methods, and with its help to solve the problem and review the results.

College of Science
School of Mathematics, Statistics, and Computer Science

Feature Selection Using Nature-Inspired Meta-Heuristic Algorithms

Sepehr Omidvar

Supervisor: Prof. Bagher BabaAli

A thesis submitted in partial fulfillment of the requirements for
the degree of B.Sc. in Computer Science

January 2022