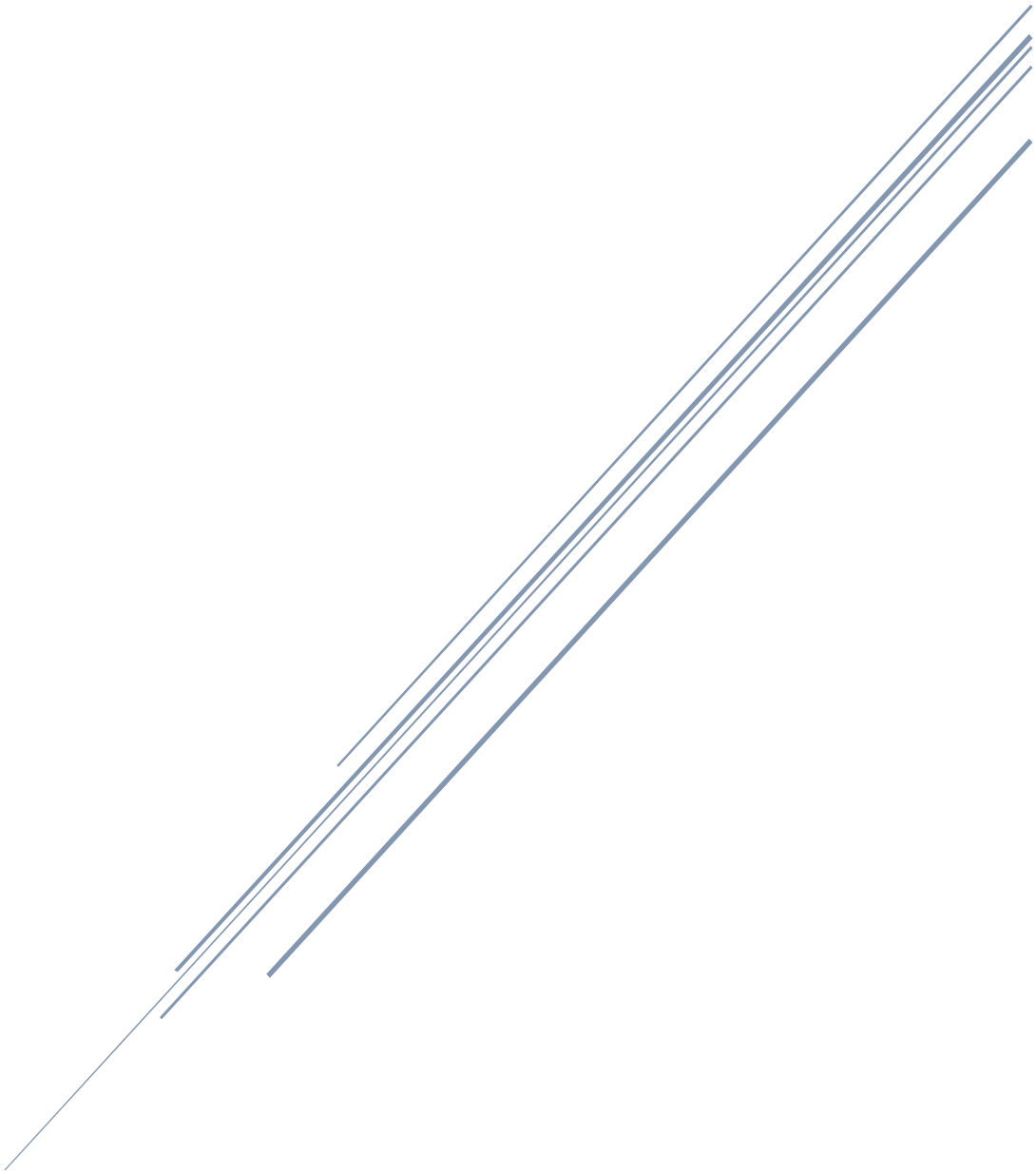
DATA MINING HOMEWORK 2

TAHSIN ILKHAS ZADEH STD.NO: 400422034

Datasets:

1. Mobile Price
2. Apartment Rental Offers in Germany



Shahid Beheshti University, Department of computer science

Data Mining Course: Dr.Farahani, Dr.Parand
Teacher Assistant: Ali Sharifi
May 18, 2022

فهرست عناوین	صفحه

سوال ۶ . بررسی استفاده از تبدیل log transform و سایر تبدیلات

چرا به تبدیل ویژگی و مقیاس بندی نیاز داریم؟

اغلب، ما مجموعه داده هایی داریم که در آن ستون های مختلف ، واحدهای متفاوتی دارند - مثلا یک ستون می تواند بر حسب کیلوگرم باشد، در حالی که ستون دیگر می تواند بر حسب سانتی متر باشد. علاوه بر این، می‌توانیم ستون‌هایی مانند درآمد داشته باشیم که می‌تواند از ۲۰۰۰۰ تا ۱۰۰۰۰۰ و حتی بیشتر باشد. در حالی که یک ستون سنی که می تواند از ۰ تا ۱۰۰ (حداکثر) متغیر باشد. بنابراین، درآمد حدود ۱۰۰۰ برابر بیشتر از سن است. اما چگونه می توان مطمئن بود که مدل با هر دو متغیر به طور یکسان رفتار می کند؟

وقتی این feature را به مدل آن‌طور که هست می‌دهیم، احتمال اینکه درآمد به دلیل ارزش بزرگ‌تر بر نتیجه تأثیر بگذارد، وجود دارد. اما این لزوماً به این معنی نیست که به عنوان یک عامل پیش بینی کننده ، فیچر درآمد ، مهم تر است. بنابراین، برای اهمیت دادن به سن و درآمد، به مقیاس‌بندی ویژگی نیاز داریم.

در بیشتر نمونه‌های مدل‌های یادگیری ماشینی، مقیاس‌کننده استاندارد یا مقیاس‌کننده MinMax را مشاهده کرده‌اید. با این حال، کتابخانه قدرتمند sklearn بسیاری از تکنیک‌های مقیاس‌بندی تبدیل ویژگی‌های دیگر را نیز ارائه می‌کند که بسته به داده‌هایی که با آن‌ها سروکار داریم، می‌توانیم از آنها استفاده کنیم.

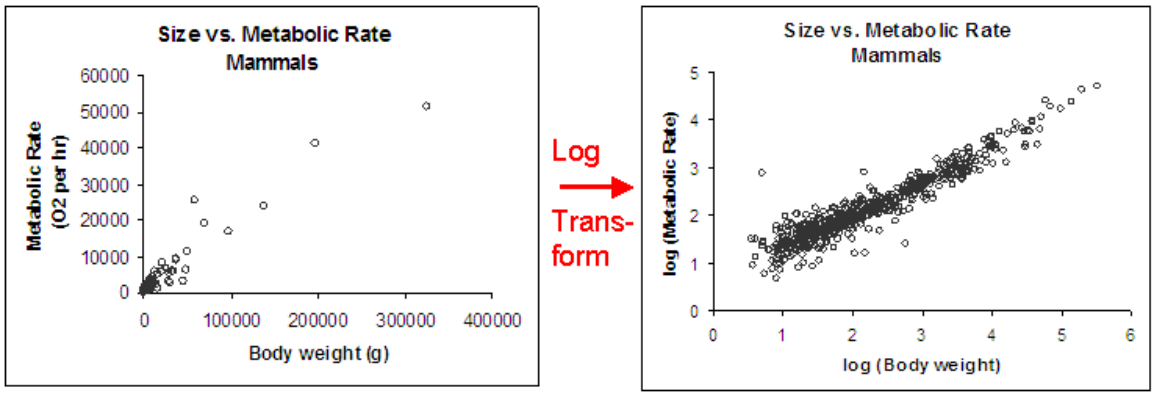
تبدیلات دیگری نیز وجود دارد که در موارد لزوم می توان از آنها استفاده کرد مثلا:

- MinMax Scaler : مقیاس‌کننده MinMax یکی از ساده‌ترین مقیاس‌کننده‌ها برای درک است که همه داده ها را بین ۰ و ۱ مقیاس می کند.
- Standard Scaler : برای هر ویژگی، مقیاس‌کننده استاندارد مقادیر را طوری مقیاس می‌دهد که میانگین ۰ و انحراف استاندارد ۱ (یا واریانس) باشد.
- MaxAbsScaler : مقیاس‌کننده MaxAbs ابتدا قدر مطلق هر مقدار در ستون را می گیرد و سپس حداکثر مقدار مطلق هر ستون را می‌گیرد و هر مقدار در ستون را بر حداکثر مقدار تقسیم می‌کند . این عملیات داده ها را بین محدوده [-۱، ۱] مقیاس می کند.
- Robust Scaler : Robust Scaler ، به موارد دورافتاده حساس نیست. این مقیاس‌کننده، میانه را از مقیاس داده‌ها حذف می‌کند به صورت :$IQR = Q3 - Q1$ و $x\_scaled = (x - Q1)/(Q3 - Q1)$
- Quantile Transformer Scaler
- Log Transformation
- Power Transformer Scaler
- Unit Vector Scaler/Normalizer

تبدیل log داده ها چیست؟

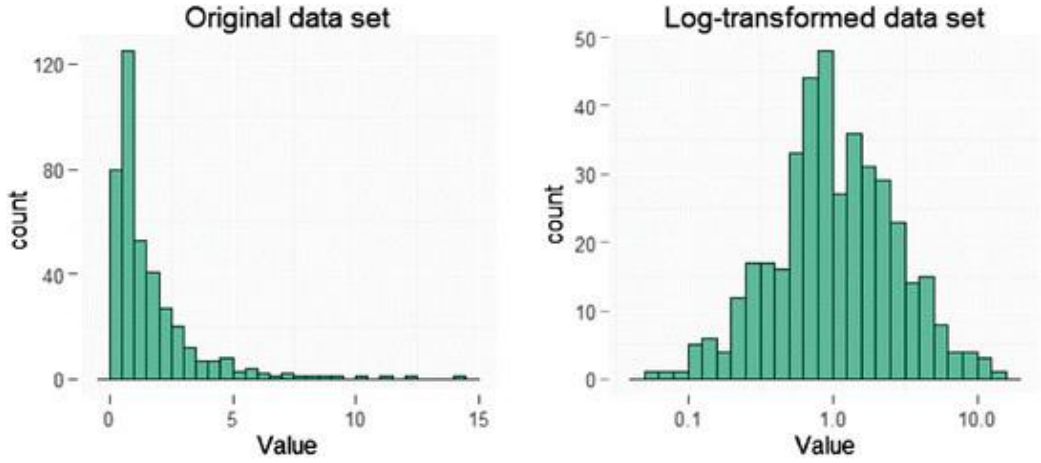
تبدیل log transform یک روش تبدیل داده است که در آن هر متغیر x را با یک log(x) جایگزین می‌کند. انتخاب پایه لگاریتمی معمولاً به تحلیلگر واگذار می‌شود و به اهداف مدل‌سازی آماری بستگی دارد. فرم کلی یک تابع لگاریتمی به این صورت است: $f(x) = k + alogb(x - h)$ که در آن a، b، k و h اعداد حقیقی هستند به طوری که b یک عدد مثبت ≠ ۱ است و x - h > 0. به عنوان مثال: $f(x) = 4 + 3log(x - 5)$

معمولا در داده کاوی ، داده ها را برای سازماندهی بهتر آنها تغییر شکل می دهند. استفاده از داده های تبدیل شده هم برای انسان و هم برای رایانه آسان تر است. داده‌های فرمت‌شده و تأیید شده به‌درستی کیفیت داده‌ها را بهبود می‌بخشد و از برنامه‌ها در برابر مشکلات بالقوه مانند مقادیر null، duplicate های غیرمنتظره، ایندکس نادرست و فرمت‌های ناسازگار محافظت می‌کند.



تبدیل log transform در مهندسی ویژگی چیست؟

Log Transform یکی از محبوب ترین تکنیک های تبدیل است. log Transform نیز تأکید بر نقاط outlier را کاهش می‌دهد و به ما امکان می‌دهد به طور بالقوه یک توزیع زنگ‌شکل به دست آوریم. ایده این است که گرفتن گزارش داده ها می تواند تقارن را به داده ها بازگرداند. تبدیل log همیشه برای تجزیه و تحلیل داده ها ضروری نیست. این می تواند به تجزیه و تحلیل آماری که ما انجام می دهیم بستگی داشته باشد.



چه زمانی باید از تبدیل log استفاده کنیم؟

تبدیل log را می توان برای تبدیل توزیع های highly skewed به less skewed استفاده کرد. این می تواند هم برای تفسیرپذیرتر کردن الگوهای موجود در داده ها و هم برای کمک به برآورده کردن مفروضات آمار استنباطی ارزشمند باشد.

در درجه اول برای تبدیل یک توزیع اریب[1] به توزیع معمولی[2]/توزیع کم انحراف[3] استفاده می شود. در این تبدیل، مقادیر یک ستون را می گیریم و به جای آن از این مقادیر به عنوان ستون استفاده می کنیم وعملیات log نقش دوگانه دارد :

- کاهش تاثیر مقادیر خیلی کم
- کاهش تأثیر مقادیر بیش از حد بالا

توجه شود که اگر داده‌های ما مقادیر منفی یا مقادیری از ۰ تا ۱ داشته باشند، نمی‌توانیم تبدیل log را مستقیماً اعمال کنیم و از آنجایی که گزارش اعداد منفی و اعداد بین ۰ و ۱ تعریف نشده است، خطا یا مقادیر NaN را در داده ها ی خود دریافت می‌کنیم. در چنین مواردی، می‌توانیم یک عدد به این مقادیر اضافه کنیم تا همه آنها بزرگتر از ۱ شوند. سپس، می‌توانیم تبدیل log را اعمال کنیم.

بطور خلاصه، هر تکنیک مقیاس‌بندی feature ، خصوصیات خاص خود را دارد که می‌توانیم از آنها برای بهبود مدل خود استفاده کنیم. با این حال، درست مانند سایر مراحل در ساخت یک مدل پیش‌بینی، انتخاب مقیاس‌کننده مناسب نیز یک فرآیند آزمون و خطا است و بهترین مقیاس‌کننده‌ای وجود ندارد که هر بار کار کند.

سوال ۸. Cross validation and bootstrapping

Cross validation و bootstrapping هردو از روش‌های نمونه‌گیری مجدد هستند.

بوت استرپینگ معیاری است که بر نمونه‌گیری تصادفی با جایگزینی تکیه دارد. بوت استرپ مجدداً با جایگزینی نمونه‌برداری می‌کند (و معمولاً مجموعه‌های داده «جایگزین» جدیدی با تعداد موارد مشابه مجموعه داده اصلی تولید می‌کند). این روشی است که در بسیاری از موقعیت‌ها مانند اعتبارسنجی عملکرد یک مدل پیش‌بینی‌کننده، روش‌های ensemble ، تخمین bias و واریانس پارامتر یک مدل و غیره کمک می‌کند. به دلیل ترسیم با جایگزینی، یک مجموعه داده بوت استرپ ممکن است حاوی چندین نمونه از همان موارد اصلی باشد و ممکن است دیگر موارد اصلی را کاملا حذف کند.

ما می توانیم این روش را چندین بار تکرار کنیم و میانگین امتیاز را به عنوان تخمین عملکرد مدل خود محاسبه کنیم. همچنین، Bootstrapping به روش‌های ensemble training مربوط می‌شود، زیرا می‌توانیم با استفاده از هر مجموعه داده بوت استرپ یک مدل بسازیم و این مدل‌ها را در یک مجموعه با استفاده از رأی اکثریت (برای طبقه‌بندی) یا محاسبه میانگین (برای پیش‌بینی‌های عددی) برای همه موارد، «bag» کنیم. این مدل ها به عنوان نتیجه نهایی ما هستند.

Cross validation بدون جایگزینی مجدد نمونه‌برداری می‌کند و بنابراین دیتاست های جایگزینی تولید می کند که کوچکتر از اصلی هستند. Cross validation روشی برای اعتبارسنجی عملکرد یک مدل است و با تقسیم داده های آموزشی به k قسمت انجام می شود. ما فرض می کنیم که قسمت های k-1 مجموعه آموزشی است و استفاده از قسمت دیگر مجموعه تست ما است. می‌توانیم هر بار با نگه داشتن بخش متفاوتی از داده‌ها، این k بار را به‌طور متفاوت تکرار کنیم. در نهایت، میانگین نمره k را به عنوان تخمین عملکرد خود در نظر می گیریم. اعتبار سنجی متقاطع می تواند از bias یا واریانس تاثیر پذیرد. با افزایش

[1] skewed distribution
[2] normal distribution
[3] less-skewed

تعداد تقسیم ها، واریانس نیز افزایش می یابد و bias کاهش می یابد. از سوی دیگر، اگر تعداد تقسیم‌ها را کاهش دهیم، bias افزایش و واریانس کاهش می‌یابد.

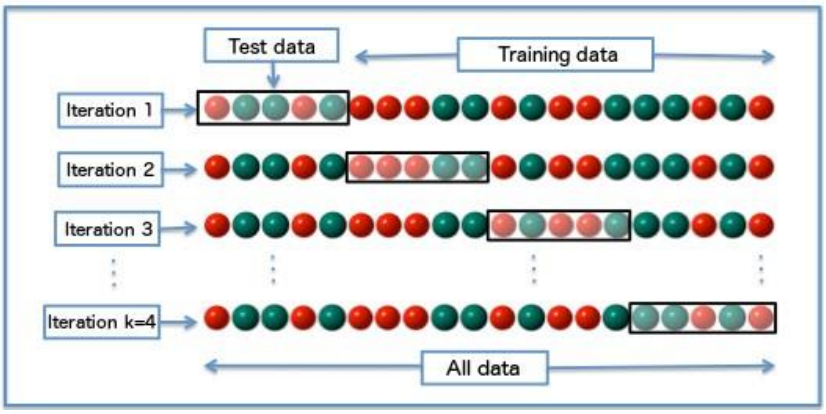
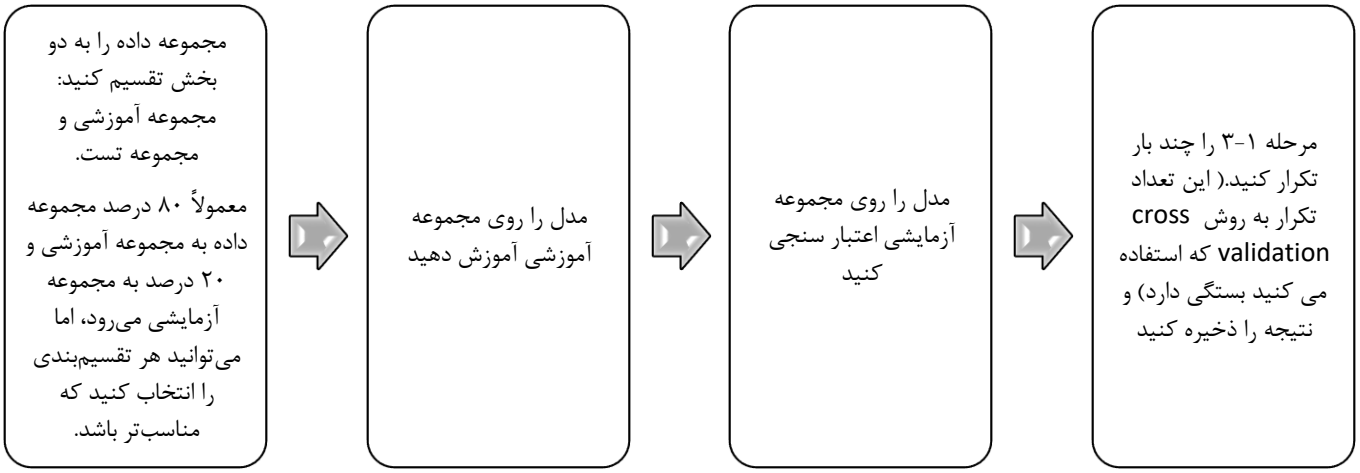


Diagram of k-fold cross-validation with k=4.

به طور خلاصه، تکنیک های مختلفی وجود دارد که ممکن است برای Cross validation یک مدل استفاده شود. با این حال، همه آنها یک الگوریتم مشابه دارند:



الگوریتم تکنیک Hold-out cross-validation

Cross validation مجموعه داده موجود را برای ایجاد مجموعه داده های متعدد تقسیم می کند و روش بوت استرپینگ از مجموعه داده اصلی برای ایجاد مجموعه داده های متعدد پس از نمونه برداری مجدد با جایگزینی استفاده می کند. هنگامی که برای اعتبارسنجی مدل استفاده می شود، بوت استرپینگ به اندازه Cross validation قوی نیست. Bootstrapping بیشتر در مورد ساخت ensemble models یا فقط تخمین پارامترها است

ما معمولاً از روش Hold-out در مجموعه داده‌های بزرگ استفاده می‌کنیم، زیرا فقط یک بار نیاز به آموزش مدل دارد.

سوال ۹ . 5x2 cross validation

5x2 cross validation عبارت است از ۵ بار تکرار 2-fold و سپس محاسبه میانگین . در سالهای گذشته، در این مقاله[۴] نوشته Dietterich، روش نمونه‌گیری مجدد را به نام اعتبارسنجی متقابل ۵×۲ توصیه می‌کند که شامل ۵ تکرار اعتبارسنجی متقاطع ۲ برابری است. استفاده از آزمون مک نمار[۵] یا اعتبارسنجی متقاطع ۵×۲ در بیشتر ۲۰ سال از زمان انتشار مقاله به توصیه اصلی تبدیل شده است.

در این روش ، ابتدا از t تست جفتی ۵ در ۲ استفاده می کنیم و این تست برای هر طبقه‌بندی کننده، ۵ بار تکرار 2-fold را اجرا می‌کند (دو تا برای اطمینان از این است که هر مشاهده فقط در مجموعه trainیا test برای یک تخمین واحد از مدل انتخاب شود.). سپس از تفاوت امتیازات برای محاسبه آمار t از فرمول زیر استفاده می شود:

$$t = \frac{p_1^{(1)}}{\sqrt{\frac{1}{5}\sum_{i=1}^{5} s_i^2}}$$

where :

- $p_1^{(1)}$ is the classifiers' scores difference for the first fold of the first iteration
- s_i^2 is the estimated variance of the score difference for i^{th} iteration. This variance computes as $\left(p_i^{(1)} - \overline{p_i}\right)^2 + \left(p_i^{(2)} - \overline{p_i}\right)^2$
- $p_i^{(j)}$ is the classifiers' scores difference for the i^{th} iteration and fold j
- $\overline{p_i} = \left(p_i^1 + p_i^2\right)/2$

از آزمون t-student برای به‌روزرسانی شده نتایج استفاده می‌شود تا درجات محدود آزادی را با توجه به وابستگی بین نمرات مهارت تخمین زده بهتر منعکس کند.

آنچه باید بدانیم این است که بر اساس فرضیه صفر (یعنی هر دو طبقه‌بندی‌کننده از نظر آماری با هم برابر هستند) فرض می‌شود که تفاوت امتیاز بین دو طبقه‌بندی کننده در هر فولد از توزیع نرمال پیروی می‌کند. با این فرض آمار t از توزیع t با ۵ درجه آزادی پیروی می کند.

برای آزمایش فرضیه صفر، مقدار t را محاسبه کرده و بررسی می کنیم که آیا توزیع t با ۵ درجه آزادی را برآورده می کند. یعنی ما بررسی می‌کنیم که آیا مقدار به نظر یک مقدار پرت است یا نه. اگر مقدار به اندازه کافی به ۰ نزدیک شود، فرضیه صفر برآورده می شود و طبقه بندی کننده ها برابر فرض می شوند.توجه شود که بیشتر آزمون‌های آماری برای طبقه‌بندی‌کننده‌ها از امتیاز دقت استفاده می‌کنند که می توان این را به Gini تعمیم داد.

مزیت مهمی که از cross validation تکراری (یا تخمین‌های خارج از بوت استرپ) به دست می‌آید این است که چندین پیش‌بینی برای یک مورد مشابه توسط مدل‌های جایگزین متفاوت دارید. این ها با آموزش در مجموعه های آموزشی کمی متفاوت ، متفاوت هستند. بنابراین، می‌توانید واریانس را به دلیل این تفاوت‌های جزئی در داده‌های train، با توجه به مبادله چند مورد train، اندازه‌گیری کنید. تکرارها به تخمین این واریانس (و کاهش عدم قطعیت واریانس مربوطه در برآورد نهایی) کمک می کنند.

[4] Approximate Statistical Tests For Comparing Supervised Classification Learning Algorithms, Thomas G. Dietterich, department of computer science, Oregon state university, Corvallis, OR 97331,dec 30 ,1997

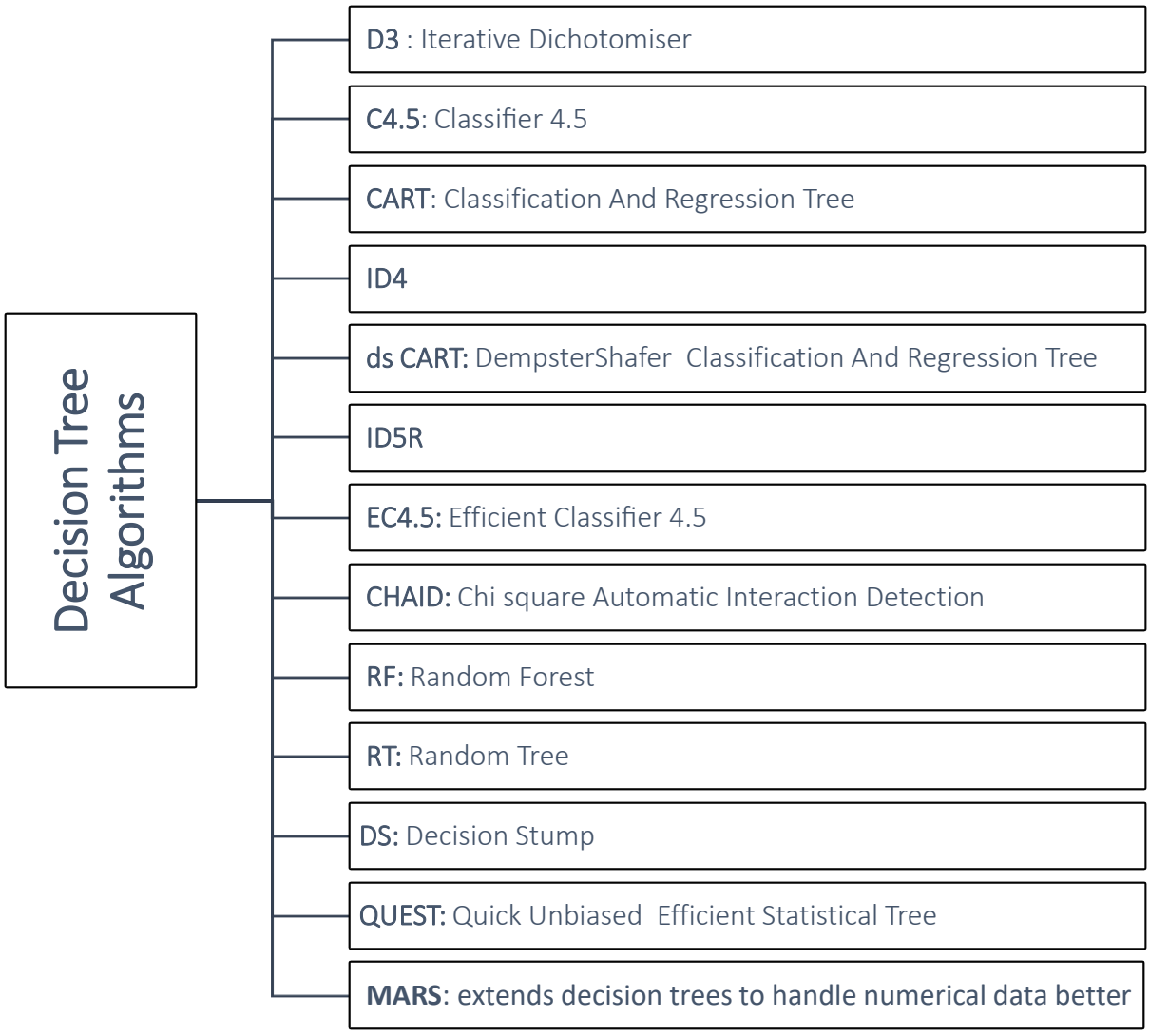
[5] McNemar’s statistical hypothesis test

علاوه بر این، شما انتخاب واقعی موارد آزمایشی را دارید که به واریانس کمک می‌کنند: تخمین‌ها بر اساس تعداد کل کمتر موارد مستقل، قطعیت کمتری دارند. با این نوع واریانس، موارد واقعی بیشتر، کمک کننده است، اما تکرارهای بیشتر هیچ تاثیری بر این منبع عدم قطعیت ندارند. توجه به این نکته ضروری است که این سهم های واریانس مستقل از یکدیگر هستند.

سوال ۱۰. الگوریتم های متداول برای ساخت درخت تصمیم Decision tree در داده کاوی

Decision Tree یکی از قدرتمندترین و محبوب ترین الگوریتم ها است. الگوریتم درخت تصمیم در دسته الگوریتم های یادگیری نظارت شده[۶] قرار می گیرد و هم برای متغیرهای خروجی پیوسته و هم برای متغیرهای categorical[۷] کار می کند.

درخت تصمیم گره ها را روی همه متغیرهای موجود تقسیم می کند و سپس تقسیمی را انتخاب می کند که منجر به اکثر گره های فرعی همگن می شود. الگوریتم ID3 درخت‌های تصمیم را با استفاده از رویکرد جستجوی حریصانه از بالا به پایین در فضای شاخه‌های ممکن و بدون پس‌گرد ایجاد می‌کند. الگوریتم های درخت تصمیم قابل توجه عبارتند از:



[6] Supervised
[7] categorical output variables.

الگوریتمID3

این الگوریتم یکی از ساده ترین الگوریتم های درخت تصمیم (decision-tree)است. در این الگوریتم درخت تصمیم از بالا به پایین ساخته می‌شود. این الگوریتم با این سوال شروع می‌شود: کدام ویژگی باید در ریشه درخت مورد آزمایش، قرار بگیرد؟ برای یافتن جواب از معیار بهره اطلاعات استفاده می‌شود.

با انتخاب این ویژگی، برای هر یک از مقادیر ممکن آن یک شاخه ایجاد شده و نمونه- های آموزشی بر اساس ویژگی هر شاخه مرتب می‌شوند. سپس عملیات فوق برای نمونه- های قرار گرفته در هر شاخه تکرار می- شوند تا بهترین ویژگی برای گره بعدی انتخاب شود.

الگوریتمC4.5

این الگوریتم یکی از تعمیم های الگوریتم ID3 است که از معیار نسبت بهره (Gain ratio)استفاده می- کند. الگوریتم هنگامی متوقف می‌شود که تعداد نمونه ها کمتر از مقدار مشخص شده‌ای باشد. این الگوریتم از تکنیک پس هرس استفاده می‌کند و همانند الگوریتم قبلی داده‌های عددی را نیز می پذیرد.

از نقاطِ ضعف الگوریتم ID3 که در C4.5 رفع شده است می‌توان به موارد زیر اشاره کرد:

الگوریتم C4.5 می‌تواند مقادیر گسسته یا پیوسته را در ویژگی‌ها درک کندو الگوریتم C4.5 قادر است با وجود مقادیر گمشده نیز درخت تصمیم (decision tree)خود را بسازد، در حالی که الگوریتمی مانند ID3 و بسیاری دیگر از الگوریتم‌های طبقه‌بندی نمی‌توانند با وجود مقادیر گمشده، مدلِ خود را بسازند.

سومین موردی که باعث بهینه شدن الگوریتم C4.5 نسبت به ID3 می‌شود، عملیاتِ هرس کردن جهت جلوگیری از بیش برازش می‌باشد. الگوریتم‌هایی مانند ID3 به خاطر اینکه سعی دارند تا حد امکان شاخه و برگ داشته باشند (تا به نتیجه مورد نظر برسند) با احتمال بالاتری دارای پیچیدگی در ساخت مدل و این پیچیدگی در بسیاری از موارد الگوریتم را دچار بیش برازش و خطای بالا می‌کند. اما باعملیات هرس کردن درخت که در الگوریتم ۵ انجام می‌شود، می‌توان مدل را به یک نقطه بهینه رساند که زیاد پیچیده نباشد (و البته زیاد هم ساده نباشد) و بیش برازش یا کم برازش (Underfitting)رخ ندهد.الگوریتم C4.5این قابلیت را دارد که وزن‌های مختلف و غیر یکسانی را به برخی از ویژگی‌ها بدهد.

الگوریتمCHAID

محققان آمار کاربردی، الگوریتم‌هایی را جهت تولید و ساخت درخت تصمیم توسعه دادند. الگوریتم CHAID در ابتدا برای متغیرهای اسمی طراحی شده بود. این الگوریتم با توجه به نوع برچسب کلاس از آزمون‌های مختلف آماری استفاده می‌کند. این الگوریتم هرگاه به حداکثر عمق تعریف شده‌ای برسد و یا تعداد نمونه‌ها در گره جاری از مقدار تعریف شده‌ای کمتر باشد، متوقف می‌شود. الگوریتم CHAID هیچگونه روش هرسی را اجرا نمی‌کند.

الگوریتم درخت تصمیم به این صورت است که یک گره ریشه در بالای آن قرار دارد و برگ های آن در پایین می باشند. یک رکورد در گره ریشه وارد می شود و در این گره یک تست صورت می گیرد تا معلوم شود که این رکورد به کدام یک از گره های فرزند (شاخه پایین‌تر) خواهد رفت

CHAID. هنگام محاسبه درختان طبقه بندی، تقسیم های چند سطحی را انجام می دهد.

MARS: درختان تصمیم را برای مدیریت بهتر داده های عددی گسترش می دهد.

الگوریتم‌های ساخت درخت‌های تصمیم معمولاً از بالا به پایین کار می‌کنند، با انتخاب متغیری در هر مرحله که به بهترین نحو مجموعه موارد را تقسیم می‌کند. الگوریتم های مختلف از معیارهای مختلفی برای اندازه گیری "بهترین" استفاده می کنند. اینها عموماً همگنی متغیر هدف را در زیر مجموعه ها اندازه گیری می کنند. چند نمونه در زیر آورده شده است. این معیارها برای هر زیر مجموعه نامزد اعمال می شود و مقادیر حاصل با هم ترکیب می شوند (مثلاً میانگین می شوند) تا معیاری از کیفیت تقسیم ارائه شود.

درختان استنتاج شرطی[8] رویکرد مبتنی بر آمار که از آزمون‌های ناپارامتریک به عنوان معیارهای تقسیم استفاده می‌کند، برای جلوگیری از برازش بیش از حد برای آزمایش‌های متعدد تصحیح شده است. این رویکرد منجر به انتخاب بی‌طرفانه پیش‌بینی‌کننده می‌شود و نیازی به هرس ندارد.

ناخالصی جینی Gini impurity

ناخالصی جینی، شاخص تنوع جینی، یا شاخص جینی سیمپسون در تحقیقات تنوع زیستی، توسط الگوریتم CART (درخت طبقه بندی و رگرسیون) برای طبقه بندی درختان استفاده می شود، ناخالصی جینی (به نام ریاضیدان ایتالیایی کورادو جینی) معیاری برای سنجش است. اگر به طور تصادفی بر اساس توزیع برچسب‌ها در زیرمجموعه برچسب‌گذاری شود، هر چند وقت یک‌بار یک عنصر به‌طور تصادفی از مجموعه به اشتباه برچسب‌گذاری می‌شود.

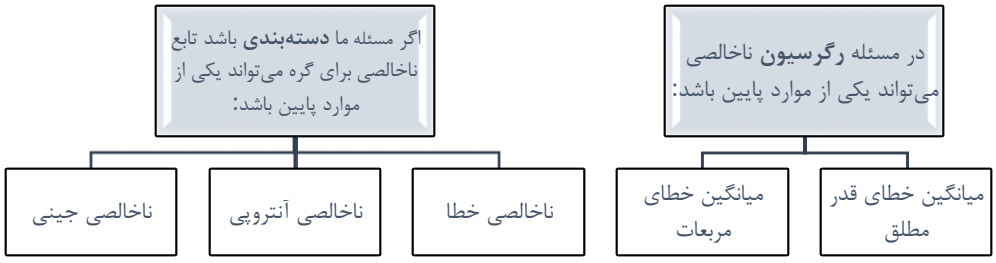
ناخالصی جینی را می توان با جمع کردن احتمال p_i یک آیتم با برچسب i اشتباه در برابر احتمال $\sum_{k \neq i} p_k = 1 - p_i$ محاسبه کرد و زمانی که همه موارد در گره در یک دسته هدف قرار می گیرند به حداقل خود (صفر) می رسد.

Information gain

Information gain در درخت تصمیم توسط الگوریتم های تولید درخت ID3، C4.5 و C5.0 استفاده شده. کسب اطلاعات مبتنی بر مفهوم " آنتروپی و محتوای اطلاعاتی" از نظریه اطلاعات[9] است. آنتروپی به صورت زیر تعریف می شود:

$$\mathrm{H}(T) = \mathrm{I}_E(p_1, p_2, \ldots, p_J) = -\sum_{i=1}^{J} p_i \log_2 p_i$$

به‌طور دقیق تر در هر گره ما سعی میکنیم یک بُعد از متغیرهایی وابسته را به همراه یک آستانه انتخاب کنیم و داده‌ها را برحسب این بُعد و آستانه به دو نیم تقسیم کنیم، به قسمی که بطور متوسط در هر دو نیم متغیرهای مستقل y خیلی به هم نزدیک و همسان شده باشند. این بعد و آستانه را می‌نامیم. حال سؤال اینجاست که کدام بُعد از متغیرهای وابسته و چه آستانه‌ای را باید انتخاب کرد. به زبان ریاضی باید آن تتایی را انتخاب کرد که ناخالصی داده را کم کند.



[8] Conditional Inference Trees
[9] Information theory

سوال ۱۳. هرس درخت تصمیم : Prepruning و Postpruning

در درخت تصمیم مشکلی با نام overfitting مطرح است . ممکن است در ایجاد درخت تعداد زیادی شاخه به وجود آید که دلیل آن وجود آنومالی در داده‌ها است . آنومالی به دلیل وجود نویز و داده های پرت به وجود آید از طرفی ممکن است درخت ایجاد شده برای داده های جدید ضعیف عمل کند

راه حل: هرس نمودن شاخه‌های زائد است و اینکار باعث کوچک شدن، ساده شدن و به الطبع فهم آسان درخت خواهد شد. ازجهتی برای داده های تست نیز عملکرد بهتری خواهد داشت . برای هرس دو روش وجود دارد:

۱. Prepruning
۲. Postpruning

Prepruning

- این روش در زمان ساخت درخت اعمال می‌شود
- درواقع در صورتی که تشخیص داده شود که تقسیم بیشتر یک شاخه بهبودی در دقت نخواهد داشت آن شاخه هرس شده و دیگر تقسیم انجام نمیگیرد
- کلاس آن شاخه براساس بیشترین تعداد اعضای آن محاسبه می‌شود
- مثلا درصورتی که بیشتر داده‌های باقی‌مانده در آن شاخه yes باشند، شاخه را دیگر تقسیم نکرده و یک برگ با مقدار برچسب yes قرار میدهیم
- تقسیم شدن یا نشدن براساس تعیین یک آستانه مناسب است
- تعیین این آستانه ساده نیست

Postpruning

- روش متداول تر است
- زمانی که درخت تصمیم ایجاد شد اعمال می‌شود
- درصورتی که زیر درخت یک گره، هرس شود آن گره تبدیل به برگ می‌شود
- کلاس این برگ بیشترین درصد کلاس‌های آن زیر درخت خواهد شد
- این روش از روش قبلی هزینه برتر است

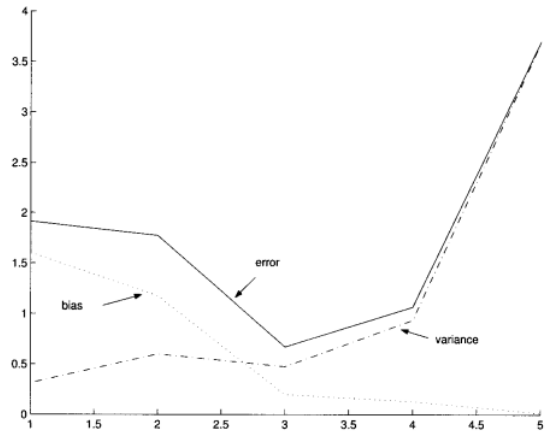
بعنوان نمونه :

- بطور مثال در درخت CART از معیار cast complexity برای پس هرس استفاده می‌شود
- این معیار توسط دو عامل تعیین می‌شود
  - تعیین برگ های درخت(number of leaves)
  - نسبت خطای درخت(error rate)
- هرس از پایین درخت شروع می‌شود و براساس این معیار هرس شدن یا نشدن یک شاخه تعیین می‌شود
- برای تعیین خطای درخت از مجموعه pruning set برای تعیین درصد خطا استفاده میشود

توجه داشته باشید که هرس از پایین به بالا و چپ به راست است.

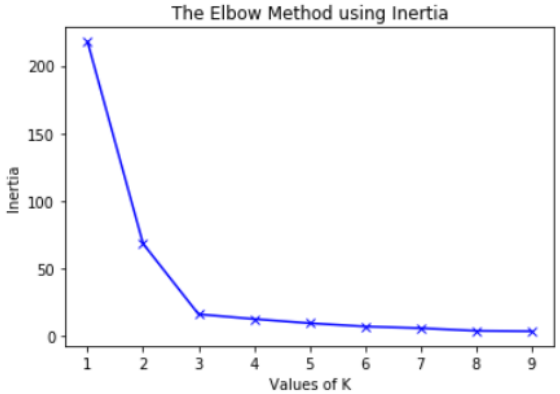
سوال ۱۴. The elbow method

CSS [10]مجموع مجذور فاصله بین هر نقطه و مرکز در یک خوشه است. هنگامی که WCSS[11] را با مقدار K رسم می کنیم، نمودار شبیه یک Elbow به نظر می رسد. با افزایش تعداد خوشه ها، مقدار WCSS شروع به کاهش می کند. برای تعیین تعداد بهینه خوشه ها، باید مقدار k را در "زانو(elbow) " انتخاب کنیم، یعنی نقطه ای که پس از آن اعوجاج/اینرسی (distortion/inertia) به صورت خطی شروع به کاهش می کند.



شکل ۱ :نمودار بایاس و واریانس بنا بر مرتبه های مختلف مدل

بنابراین مثلا برای داده های داده شده درشکل زیر ، نتیجه می گیریم که تعداد بهینه خوشه برای داده ها ۳ است.

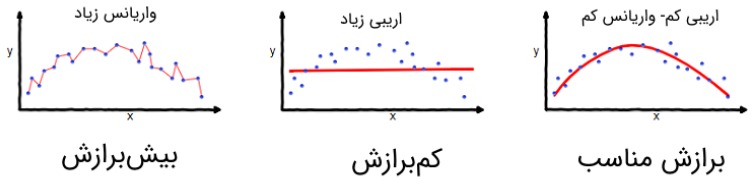


خطای بایاس: وجود فرضیه‌های مختلف روی مدل و الگوریتم یادگیری منجر به ایجاد خطای بایاس (اریبی) می‌شود. بزرگ بودن اریبی می‌تواند الگوریتم یا مدل آماری را از کشف روابط بین ویژگی‌ها (Features) و متغیر پاسخ (Target Variable) باز دارد. اغلب بزرگ بودن خطای اریبی، منجر به کم‌برازش (Underfitting) می‌شود.

خطای واریانس: حساسیت زیاد مدل با تغییرات کوچک روی داده‌های آموزشی، نشانگر وجود واریانس زیاد است. این امر نشانگر آن است که اگر مدل آموزش داده شده را روی داده‌های آزمایشی به کارگیریم، نتایج حاصل با داده‌های واقعی فاصله زیادی خواهند داشت. متاسفانه افزایش واریانس در این حالت منجر به مدل‌بندی مقادیر Noise شده و به جای پیش‌بینی صحیح، دچار پیچیدگی و مشکل «بیش‌برازش» (Overfitting) می‌شود.

[10] Cluster-Sum of Squared
[11] Within-Cluster-Sum of Squared Errors (WCSS)

مدل‌های با واریانس بزرگ (مثلا رگرسیون چند جمله‌ای هم‌مرتبه با تعداد مشاهدات)، که معمولاً پیچیده‌تر هستند، این امکان را می‌دهد تا داده‌های train به خوبی برازش شوند. با این وجود، ممکن است مشاهدات برازش شده دارای خطا یا نویز باشند که متاسفانه مدل تحت تاثیر آن‌ها، برآوردها را با دقت انجام داده است. به این ترتیب پیش‌بینی آن‌ها باعث افزودن پیچیدگی در مدل شده است. در حالیکه این امر از طرفی دقت برآوردها را هم برای داده‌های آزمایشی کمتر می‌کند. در مقابل، مدل‌هایی که دارای بایاس بزرگی هستند، نسبتاً ساده بوده (مثل مدل رگرسیون دو جمله‌ای یا حتی خطی) اما ممکن است واریانس کوچکتری را براساس مجموعه داده‌های آزمایشی ایجاد کنند.

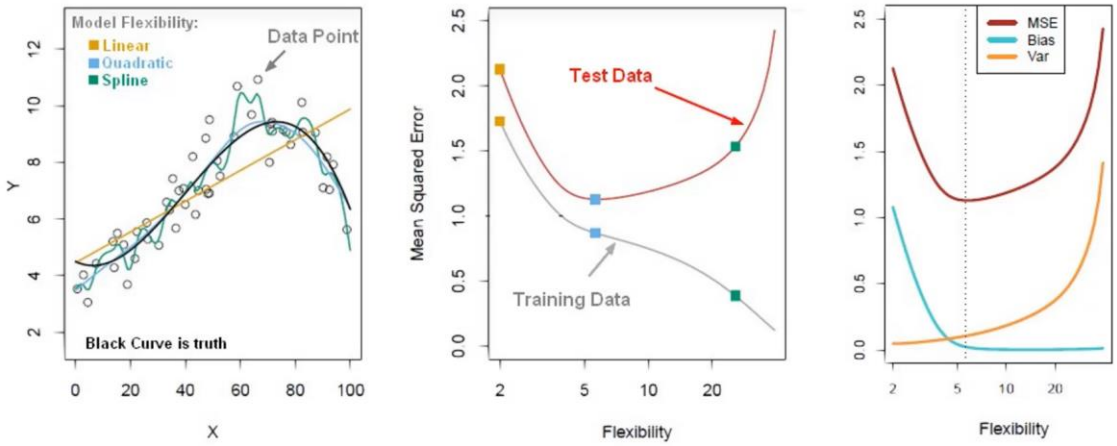


استفاده از روش‌های کاهش بُعد (Dimensionality Reduction) و انتخاب ویژگی (Feature Selection) می‌توانند واریانس را به کمک ساده‌سازی مدل انجام دهند. از طرفی افزایش داده‌های آموزشی باعث کاهش واریانس خواهد شد.
افزایش تعداد متغیرهای پیش‌گو (Predictors)، باعث کاهش بایاس (اریبی) می‌شود ولی این امر به قیمت افزایش واریانس خواهد بود. الگوریتم‌های یادگیری، معمولا از یک پارامتر تنظیم کننده برای موازنه واریانس و بایاس استفاده می‌کنند.
از آنجایی که همه منابع خطا (مربع اریبی، واریانس مدل یا واریانس عبارت خطا) مثبت یا حداقل نامنفی هستند، می‌توان یک کران پایین برای خطای مدل روی داده‌های آزمایشی در نظر گرفت.
هر چه مدل پیچیده‌تر باشد، نقاط بیشتری از داده‌های آموزشی را پوشش می‌دهد و بایاس نیز کم خواهد بود. در حالیکه یک مدل پیچیده، باعث بوجود آمدن خطای زیاد برای برازش داده‌های جدید خواهد شد در نتیجه واریانس مدل را برای چنین داده‌هایی، زیاد می‌کند.
پس برای ساختن یک مدل خوب، باید تعادل خوبی بین واریانس و بایاس پیدا کنیم به طوری که خطای کل را به حداقل برساند.به همین دلیل خطای مدل را به واریانس و بایاس (اریبی) تجزیه می‌کنیم. در نهایت، تابع زیان MSE (میانگین مربعات خطا) یا لگاریتم درستنمایی منفی به کمک محاسبه امید ریاضی و طبق رابطه زیر حاصل می‌شود(خطای کل) :

$$\text{Total Error} = \text{Bias } 2 + \text{Variance} + \text{Irreducible Error}$$

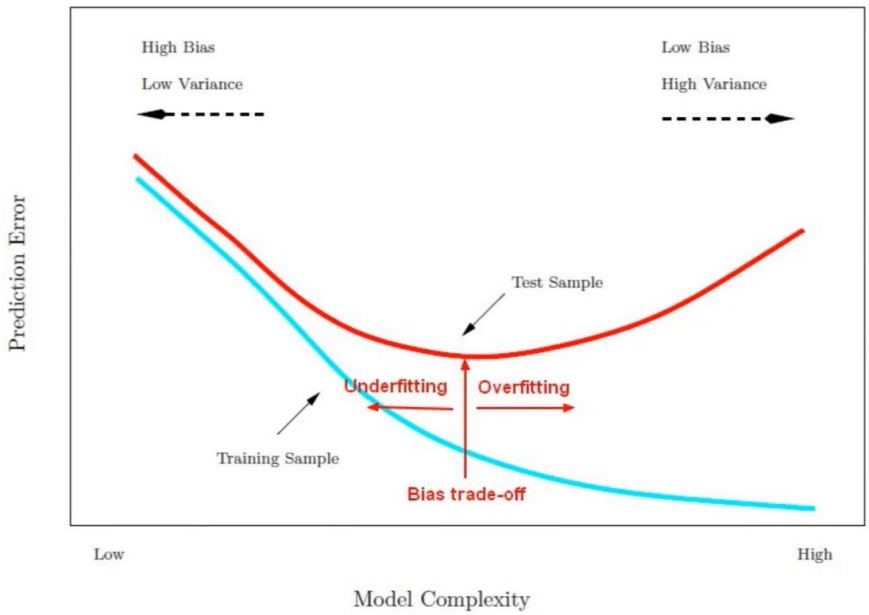
$$\Longrightarrow \quad MSE = E_x\{Bias_D[\hat{f}(x;D)]^2 + Var_D[\hat{f}(x;D)]\} + \sigma^2$$

تعادل بهینه بین بایاس و واریانس، باعث می شود مدل overfitt و underfitt نشود.



در تصویر فوق ، نمودار سمت چپ در شکل بالا یک انعطاف پذیری (flexibility) مدل را به صورت خطی، درجه دوم و برازش اسپلاین توصیف می کند. برای مقایسه مدل‌هایمان با یکدیگر، باید انعطاف‌پذیری در مقابل میانگین مربعات خطا را همانطور که در نمودار میانی بالا توضیح داده شد، ترسیم می کنیم.

نمودار داده های آموزشی را در مقابل داده های آزمون ترسیم می کند. می بینیم که برای مدل خطی (به رنگ زرد) خطای بالایی داریم. خطا برای مدل درجه دوم (به رنگ آبی) کاهش می یابد، ستاره خطا دوباره برای مدل spline (به رنگ سبز) افزایش می یابد، اما به طور اساسی فقط در مجموعه تست، در عین حال برای مجموعه آموزشی، خطا همچنان کاهش می یابد: این شرط overfitting نامیده می شود.



در شکل بالا ، خطای پیش بینی را به عنوان تابعی از پیچیدگی مدل (model complexity) نشان می دهد. همانطور که به سمت چپ حرکت می کنیم، بایاس زیاد اما واریانس کم =underfit داده ها را داریم. همانطور که به سمت راست حرکت می کنیم، جایی که پیچیدگی بالاتری وجود دارد، یک بایاس کمتر اما واریانس بالا = overfit داده ها را دریافت می کنیم.

در اغلب «مدل‌های پیش‌بین(Predictive Model) ، وجود بایاس کوچک برای پارامترها منجر به واریانس بزرگ برای مدل خواهد شد. البته برعکس این حالت نیز وجود دارد، به این معنی که با کوچک کردن واریانس مدل، با مشکل بزرگ شدن بایاس یا اریبی پارامترها مواجه خواهیم شد.
مسئله اصلی آن است که در یک مدل مناسب، هم بایاس و هم واریانس باید حداقل ممکن باشند. ولی متاسفانه، کمینه‌سازی (Minimization) هر دو این شاخص‌ها به شکل توام، امکان‌پذیر نیست. چنین وضعیتی را «تناقض واریانس-اریبی» (Bias-Variance Dilemma) می‌نامند.

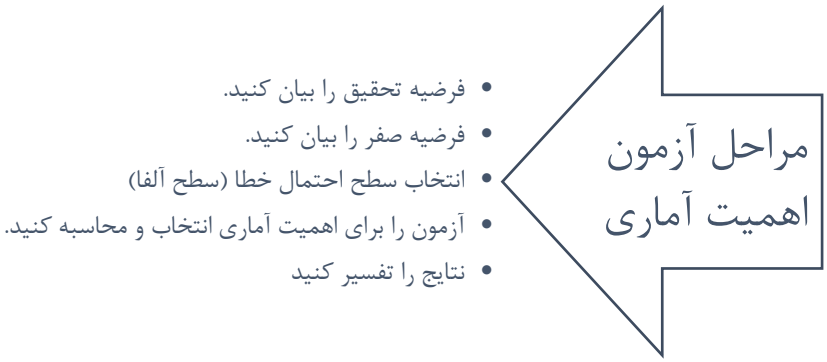
تجزیه میزان خطای کل به واریانس و بایاس، همچنین بهره‌گیری از کمینه‌سازی آن‌ها در مدل رگرسیونی، ایده اصلی در روش‌های رگرسیون با قاعده‌سازی (Regularization) نظیر «رگرسیون لاسو» (Lasso Regression) و رگرسیون Ridge Regression است. به این ترتیب با استفاده از تکنیک‌هایی مانند جریمه کردن مدل، مثلا در «رگرسیون لاسو» (Lasso Regression) یا رگرسیونRidge Regression موازنه واریانس و بایاس برقرار می‌شود و مدلی بدون underfitting یا overfitting بدست می‌آید.
روش‌های قاعده‌سازی، بایاس را وارد روش حل و برآورد پارامترهای مدل کرده و به این ترتیب واریانس مدل را نسبت به روش‌های رگرسیون عادی (Ordinary Least Squares) یا OLS کاهش می‌دهند. هر چند روش‌های مبتنی بر OLS، برآوردگرهای نااریب برای پارامترهای رگرسیونی ایجاد می‌کنند ولی وجود واریانس کوچکتر در مدل‌های رگرسیونی با قاعده، مفیدتر بوده و مقادیر برآورد شده توسط آن‌ها، دارای خطای کمتری هستند.

سوال ۲ تسک امتیازی: چگونه می توان با استفاده از statistical significance tests به مقایسه مدل ها پرداخت؟

از آزمون‌های معنی‌داری آماری برای پاسخ به این سؤال استفاده می‌شود: احتمال اینکه آنچه ما فکر می‌کنیم رابطه بین دو متغیر است واقعاً فقط یک اتفاق تصادفی باشد چقدر است؟ اگر نمونه های زیادی را از یک جامعه انتخاب کنیم، آیا باز هم رابطه یکسانی بین این دو متغیر در هر نمونه پیدا می کنیم؟ اگر بتوانیم یک سرشماری از جامعه انجام دهیم، آیا این رابطه در جمعیتی که نمونه از آن گرفته شده است نیز وجود دارد؟ یا اینکه یافته ما فقط به دلیل شانس تصادفی است؟

آزمون‌های معنی‌داری آماری به ما می‌گویند که احتمال اینکه رابطه‌ای که فکر می‌کنیم پیدا کرده‌ایم فقط به دلیل شانس تصادفی باشد چقدر است. آنها به ما می گویند که اگر فرض کنیم که رابطه ای وجود دارد، احتمال اشتباه ما چقدر است. ما هرگز نمی توانیم به طور کامل ۱۰۰٪ مطمئن باشیم که رابطه ای بین دو متغیر وجود دارد. منابع خطا بیش از حد قابل کنترل هستند، به عنوان مثال، خطای نمونه گیری، سوگیری محقق، مشکلات مربوط به قابلیت اطمینان و اعتبار، اشتباهات ساده و غیره. اما با استفاده از تئوری احتمال و منحنی نرمال، می‌توانیم احتمال اشتباه بودن را تخمین بزنیم، اگر فرض کنیم که یافتن یک رابطه درست است. اگر احتمال اشتباه اندک باشد، می گوییم که مشاهده ما از رابطه یک یافته آماری معنی دار است.

معنی‌داری آماری به این معناست که شانس خوبی وجود دارد که در یافتن رابطه‌ای بین دو متغیر درست باشیم. اما اهمیت آماری با اهمیت عملی یکسان نیست. می‌توانیم یافته‌های آماری معنی‌داری داشته باشیم، اما پیامدهای آن یافته ممکن است کاربرد عملی نداشته باشد. محقق باید همیشه هم اهمیت آماری و هم اهمیت عملی هر یافته تحقیق را بررسی کند.



- **آزمون فرضیه‌های آماری برای مقایسه عملکرد دو مدل**

رایج‌ترین آزمون فرضیه‌های آماری مورد استفاده برای مقایسه عملکرد مدل‌های ML، آزمون t زوجی (paired Student's t-test) است که از طریق نمونه‌های فرعی تصادفی از مجموعه داده آموزشی(random subsamples of the training dataset) ترکیب شده است. فرضیه صفر در این آزمون این است که بین عملکرد دو مدل کاربردی ML تفاوتی وجود ندارد. آزمون های دیگری نیز وجود دارند مثلا آزمون کای دو ، F-Test وغیره .

- **تست chi-square**

برای داده های اسمی و ترتیبی از Chi-Square به عنوان آزمون معناداری آماری استفاده می شود. برای محاسبه مربع کای، جدولی که توزیع مشترک دو متغیر را نشان می دهد مورد نیاز است.سپس در این جدول با مراحل زیر مربع کای را محاسبه می کنیم:

ما نمی توانیم ارزش آمار Chi-Square را به تنهایی تفسیر کنیم. در عوض، ما باید آن را در یک زمینه قرار دهیم. در تئوری، مقدار آمار Chi-Square به طور معمول توزیع می شود. یعنی مقدار آمار Chi-Square شبیه یک منحنی معمولی (زنگ شکل) است. بنابراین می توانیم از خواص منحنی نرمال برای تفسیر مقدار بدست آمده از محاسبه آمار Chi-Square استفاده کنیم.

اگر مقداری که برای Chi Square به دست می‌آوریم به اندازه کافی بزرگ باشد، می‌توان گفت که سطح معنی‌داری آماری را نشان می‌دهد که در آن می‌توان رابطه بین دو متغیر را فرض کرد. با این حال، اینکه آیا مقدار به اندازه کافی بزرگ باشد به دو چیز بستگی دارد: اندازه جدول احتمالی که آمار Chi-Square از آن محاسبه شده است. و سطح آلفای که انتخاب کرده ایم.

هرچه اندازه جدول بزرگتر باشد، در صورت مساوی بودن سایر موارد، باید مقدار Chi-Square بزرگتر باشد تا به اهمیت آماری برسد. به طور مشابه، هرچه سطح آلفا دقیق‌تر باشد، اگر سایر موارد برابر باشند، باید مقدار Chi-Square بزرگ‌تر باشد تا به معنی‌داری آماری برسد.

اصطلاح "درجات آزادی" برای اشاره به اندازه جدول احتمالی استفاده می شود که مقدار آماره Chi-Square بر روی آن محاسبه شده است. درجه آزادی به صورت حاصل ضرب (تعداد سطرهای جدول منهای ۱) برابر (تعداد ستون های جدول منهای) محاسبه می شود.

هر دو آزمون t و آزمون کای دو آزمون های آماری هستند که برای آزمایش و احتمالاً رد یک فرضیه صفر طراحی شده اند. فرضیه صفر معمولاً عبارتی است مبنی بر اینکه چیزی صفر است یا چیزی وجود ندارد. به عنوان مثال، می توانید این فرضیه را که تفاوت بین دو میانگین صفر است، یا می توانید این فرضیه را آزمایش کنید که بین دو متغیر رابطه وجود ندارد.
تفاوت بین آزمون کای دو و آزمون تی این است که آزمون t به شما این امکان را می دهد که بگویید یا "ما می توانیم فرضیه صفر میانگین های برابر را در سطح ۰،۰۵ رد کنیم" یا "شواهد کافی برای رد صفر میانگین های برابر در سطح ۰،۰۵ نداریم." آزمون کای اسکوئر به شما امکان می دهد بگویید یا "ما می توانیم فرضیه صفر عدم وجود رابطه را در سطح ۰،۰۵ رد کنیم" یا "شواهد کافی برای رد عدد صفر در سطح ۰،۰۵ نداریم."

- **آزمایش اهمیت یک مدل با F-TEST :**

برای محاسبه F-test اهمیت کلی(overall significance) ، نرم افزار آماری شما فقط باید عبارت های مناسب را در دو مدلی که با هم مقایسه می کند، قرار دهد. آزمون F-test کلی ، مدلی که شما مشخص کرده اید را با مدلی که هیچ متغیر مستقلی ندارد مقایسه می کند. این نوع مدل به عنوان یک مدل فقط رهگیری (intercept-only model) نیز شناخته می شود.
هنگام برازش داده‌ها با استفاده از رگرسیون غیرخطی، اغلب مواقعی پیش می‌آید که باید بین دو مدل انتخاب کرد که هر دو به خوبی با داده‌ها مطابقت دارند. پس از ترسیم باقیمانده‌های هر مدل و مشاهده مقادیر r^2 برای هر مدل، هر دو مدل ممکن است متناسب با داده‌ها به نظر برسند.

در این حالت می توان آزمون F انجام داد تا ببینیم کدام مدل از نظر آماری بهتر است. این آزمون F پاسخ قطعی می دهد و بر تفسیر دلخواه مقدار r^2 یا نمودار باقی مانده تکیه نمی کند.

یک آزمون F از یک توزیع F پیروی می کند و می تواند برای مقایسه مدل های آماری استفاده شود. آماره F با استفاده از یکی از دو معادله بسته به تعداد پارامترها در مدل ها محاسبه می شود:

- اگر هر دو مدل تعداد پارامترهای یکسانی داشته باشند، فرمول آماره F به صورت $F = \frac{SS1}{SS2}$ است که در آن SS1 مجموع مربعات باقیمانده برای مدل اول و SS2 مجموع مربعات باقیمانده برای مدل دوم است.

درجات آزادی N - V وجود دارد، که در آن N تعداد نقاط داده و V تعداد پارامترهای تخمین زده شده است (در هر پارامتر تخمین زده شده یک درجه آزادی از دست می رود). سپس آماره F حاصل را می توان با یک جدول F برای استخراج مقدار p-value مقایسه کرد.

- اگر مقدار p-value بزرگ باشد (بزرگتر از α) مدل اول از نظر آماری بهتر از مدل دوم است.
- اگر مقدار p-value کوچک باشد (کمتر از ۱-α) مدل دوم از نظر آماری بهتر از مدل اول است.

➢ اگر مدل ها تعداد پارامترهای متفاوتی داشته باشند، فرمول به صورت زیر در می آید:

$$F = \frac{(SS_1 - SS_2)/(df_1 - df_2)}{SS_2/df_2}$$

مجموع مربعات هر مدل و درجات آزادی برای هر مدل مانند قبل محاسبه می شود (توجه داشته باشید که مدل ها برای این مورد درجات آزادی متفاوتی خواهند داشت).

علاوه بر این، اولین مدل باید مدلی با پارامترهای کمتر باشد (یعنی مدل ساده تر). یک بار دیگر می توان از آماره F و درجات آزادی برای تعیین مقدار p استفاده کرد. هنگام یافتن مقدار p از درجه آزادی $df_1 - df_2$ و df_2 استفاده کنید. در این مورد، مقدار p کمتر از α نشان می‌دهد که مدل پیچیده‌تر (مخرج آماره F) به طور قابل‌توجهی بهتر از مدل ساده‌تر با داده‌ها مطابقت دارد.

توجه داشته باشید که آزمون F چیزی در مورد اهمیت فیزیکی مدل به شما نمی گوید و موردی ندارد که مدلی را انتخاب کنید که علیرغم عدم همخوانی مناسب با داده ها ، معنای فیزیکی بهتری داشته باشد.

هنگام مقایسه میانگین دو گروه از آزمون تی استفاده می شود هنگام مقایسه میانگین بیش از دو گروه از آزمون های ANOVA و MANOVA استفاده می شود.

تحلیل واریانس (ANOVA)

برای مقایسه برازش های دو مدل، می توانید از تابع anova() با اشیاء رگرسیون به عنوان دو آرگومان جداگانه استفاده کنید. تابع anova() اشیاء مدل را به عنوان آرگومان می گیرد و یک ANOVA را برمی گرداند و آزمایش می کند که آیا مدل پیچیده تر به طور قابل توجهی در گرفتن داده ها بهتر از مدل ساده تر است یا خیر. همچنین ، آزمون ANOVA امکان مقایسه بیش از دو گروه را به طور همزمان دارد برای تعیین اینکه آیا رابطه ای بین آنها وجود دارد یا خیر.

فرمول ANOVA به شکل زیر است:

$$\mathrm{F} = \frac{\mathrm{MST}}{\mathrm{MSE}}$$

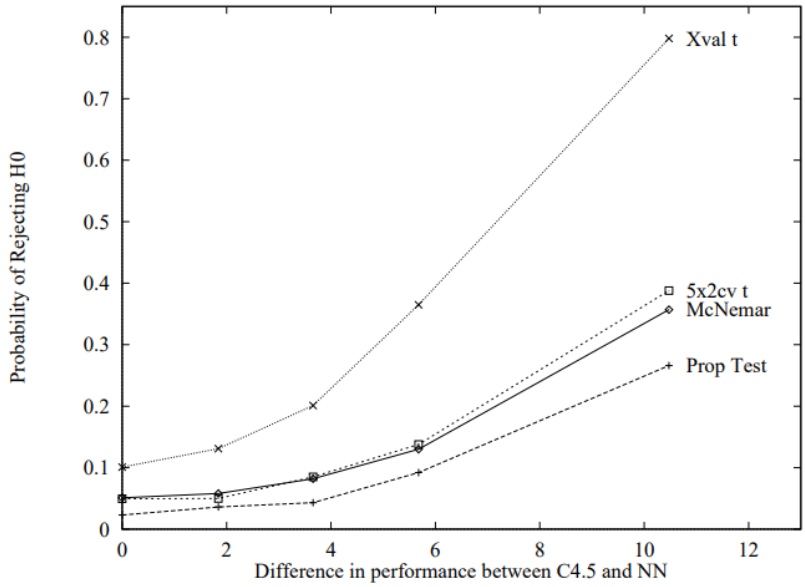
که در آن :

F=ANOVA coefficient
MST=Mean sum of squares due to treatment
MSE=Mean sum of squares due to error

در ANOVA، فرضیه صفر این است که تفاوتی بین میانگین‌های گروهی وجود ندارد. اگر هر گروهی به طور قابل توجهی با میانگین کلی گروه تفاوت داشته باشد، آنگاه ANOVA یک نتیجه آماری معنی دار را گزارش خواهد کرد.
تفاوت بین ANOVA و chi-square این است که از chi-square برای بررسی اینکه آیا توزیع طبقات و با یک مدل توزیع سازگار است (اغلب توزیع برابر، اما نه همیشه) استفاده می شود، در حالی که از ANOVA برای بررسی اینکه آیا تفاوت در میانگین بین نمونه ها معنی دار است یا خیر استفاده می شود.

آیا می توانیم از R-Squared برای مقایسه مدل ها استفاده کنیم؟

از R-Squared برای مقایسه مدل ها استفاده نکنید. دلایل متفاوتی برای این وجود دارد مثلا در بسیاری از موقعیت‌ها، R-Squared در مقایسه با مدل‌ها گمراه‌کننده است. به طور کلی، r-squared بالاتر نشان می دهد که تنوع بیشتر توسط مدل توضیح داده شده است. با این حال، همیشه اینطور نیست که r-squared بالا برای مدل رگرسیون خوب باشد.
R-squared نشان نمی دهد که آیا یک مدل رگرسیون کافی است یا خیر. شما می توانید یک مقدار R-squared پایین برای یک مدل خوب، یا یک مقدار R-squared بالا برای مدلی که با داده ها مطابقت ندارد، داشته باشید! R-squared در خروجی شما یک برآورد مغرضانه از جمعیت R-squared است.



تصویر فوق، قدرت چهار آزمون آماری در تسک تشخیص حروف. محور افقی تعداد نقاط درصدی را ترسیم می کند که دو الگوریتم C4.5 و NN در هنگام آموزش روی training set با اندازه ۳۰۰ تفاوت دارند.

به طور خلاصه ، آزمون‌های معنی‌داری آماری برای تخمین احتمال اینکه رابطه مشاهده‌شده در داده‌ها تنها به صورت تصادفی رخ داده است، استفاده می‌شود. احتمال اینکه متغیرها واقعاً در جامعه نامرتبط باشند. می توان از آنها برای فیلتر کردن فرضیه های غیرمنتظره استفاده کرد.
آزمون‌های اهمیت آماری به این دلیل استفاده می‌شوند که معیار مشترکی را تشکیل می‌دهند که برای بسیاری از افراد قابل درک است، و اطلاعات ضروری را در مورد یک پروژه تحقیقاتی که می‌تواند با یافته‌های پروژه‌های دیگر مقایسه شود، به اشتراک می‌گذارد.

با این حال، آنها اطمینان نمی دهند که این تحقیق با دقت طراحی و اجرا شده است. در واقع، آزمون‌های اهمیت آماری ممکن است گمراه‌کننده باشند، زیرا اعداد دقیقی هستند. اما هیچ ارتباطی با اهمیت عملی یافته های تحقیق ندارند. در نهایت، همیشه باید از معیارهای ارتباط[12] همراه با آزمون ها برای معناداری آماری استفاده کرد. دومی احتمال وجود رابطه را تخمین می زند. در حالی که اولی قدرت (و گاهی اوقات جهت) رابطه را تخمین می زند. هر کدام کاربرد خود را دارند و زمانی که با هم استفاده می شوند بهترین هستند.

نتیجه

- آزمون‌های فرضیه‌های آماری می‌توانند به مقایسه مدل‌های یادگیری ماشین و انتخاب مدل نهایی کمک کنند.
- استفاده ساده از آزمون های فرضیه های آماری می تواند منجر به نتایج گمراه کننده شود.
- استفاده صحیح از آزمون‌های آماری چالش برانگیز است، و برای استفاده از آزمون مک‌نمار یا اعتبارسنجی متقابل ۵×۲ با آزمون tزوجی اصلاح‌شده توافق نظر وجود دارد.

سوال ۳ تسک امتیازی : معیار Matthews Correlation Coefficient (MCC) چیست و کجا استفاده می شود؟

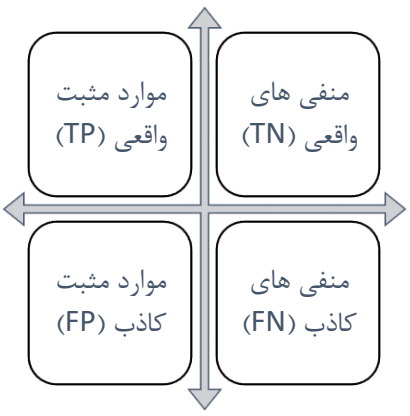
ضریب همبستگی متیوز چیست؟

ضریب همبستگی متیوز که به اختصار MCC نیز نامیده می شود توسط برایان متیوز در سال ۱۹۷۵ اختراع شد. MCC یک ابزار آماری است که برای ارزیابی مدل استفاده می شود. وظیفه آن اندازه گیری یا اندازه گیری تفاوت بین مقادیر پیش بینی شده و مقادیر واقعی است و معادل آمار مربع کای (chi-square statistics) برای یک جدول احتمالی ۲×۲ است که در آن n تعداد کل مشاهدات است.

$$|MCC| = \sqrt{\frac{\chi^2}{n}}$$

فرمول ضریب همبستگی متیوز

MCC بهترین معیار طبقه بندی تک مقداری است که به خلاصه کردن confusion matrix یا ماتریس خطا کمک می کند. یک ماتریس confusion دارای چهار موجودیت است:



[12] measures of association

و با فرمول زیر محاسبه می شود:

$$MCC = \frac{TN \times TP - FN \times FP}{\sqrt{(TP+FP)\,(TP+FN)\,(TN+FP)\,(TN+FN)}}$$

اگر پیش‌بینی نرخ‌های خوبی را برای هر چهار مورد از این نهادها به دست آورد، گفته می‌شود که معیار قابل اعتمادی است که نمرات بالایی ایجاد می‌کند. و برای مطابقت با اکثر ضرایب همبستگی، MCC نیز بین ۱+ و ۱- متغیر است:

- ضریب ۱+ یک پیش‌بینی کامل را نشان می‌دهد
- ۰ بهتر از پیش‌بینی تصادفی نیست
- ۱- نشان‌دهنده اختلاف کامل بین پیش‌بینی و مشاهده است
- اگر MCC برابر با ۱-، ۰ یا ۱+ نباشد، نشانگر قابل اعتمادی نیست که نشان دهد یک پیش‌بینی‌کننده چقدر شبیه حدس‌های تصادفی است، زیرا MCC به مجموعه داده‌ها وابسته است.

دقت و امتیاز F1 محاسبه‌شده بر روی confusion matrix ها یکی از محبوب‌ترین معیارهای پذیرفته‌شده در تسک های طبقه‌بندی باینری بوده است (و هنوز هم هستند). با این حال، این معیارهای آماری می توانند به طور خطرناکی نتایج بیش از حد خوش بینانه را به ویژه در مجموعه داده های نامتعادل نشان دهند.

در حالی که هیچ راه کاملی برای توصیف ماتریس اشتباه مثبت و منفی درست و غلط با یک عدد وجود ندارد، ضریب همبستگی متیوز به طور کلی به عنوان یکی از بهترین معیارها در نظر گرفته می شود. سایر معیارها، مانند نسبت پیش‌بینی‌های صحیح (که دقت نیز نامیده می‌شود)، زمانی که دو کلاس اندازه‌های بسیار متفاوتی دارند، مفید نیستند. به عنوان مثال، اختصاص دادن هر شی به مجموعه بزرگتر به نسبت بالایی از پیش بینی های صحیح دست می یابد، اما به طور کلی طبقه بندی مفیدی نیست.

در عوض، ضریب همبستگی متیوز (MCC)، نرخ آماری قابل اعتمادتری است که تنها در صورتی امتیاز بالایی ایجاد می‌کند که پیش‌بینی نتایج خوبی در هر چهار دسته confusion matrix (مثبت واقعی، منفی کاذب، منفی درست و نادرست) متناسب با اندازه عناصر مثبت و اندازه عناصر منفی در مجموعه داده بدست آورد.
MCC در ارزیابی طبقه‌بندی‌های باینری امتیاز آموزنده‌تر و درست‌تری نسبت به دقت و امتیاز F1 ایجاد می‌کند، و توصیه می شود ضریب همبستگی متیوز باید به دقت و امتیاز F1 در ارزیابی وظایف طبقه‌بندی باینری توسط همه جوامع علمی ترجیح داده شود.