از نظر دقت از بقیه خیلی روش بهتری است و مقادیر اولیه این که داره حتی اگر نشناسی به عنوان مقدار اولیه انتخاب کنیم، مقادیر (تصادفی هم) اولیه اش را نشانش دهیم (تصادفی)، بازدهم این روش همگرا میشه و در جواب خوش می دهد.

تمرین: تولید عدد از Mixture of Gassian ما در اینجا یک متغیره را گفتیم، شما چند متغیره را انجام دهید!!

(غایب) جلسه 8   92/08/6   Data mining   (k-means-hour.pdf)

یک کتاب داده کاوی دیگه که جدیدا روش نشدن به فارسی است میتوانید تهیه و استفاده کنید (ترجمه Tan است) = کتاب داده کاوی کاربردی دکتر محمد صنیعی زاده دانشگاه تربیت مدرس.

این جلسه میخواهیم در مورد k-means صحبت کنیم، در واقع در مورد روش های clustering که صحبت کردیم، ما جلسه ی پیش یک اشاره ای کردیم به Model Based. در Model Based، روش clustering آدم گفتیم: این روش به خودی روش سلسله مراتبی نیست ولی اگر که داده های ما فرض نرمال بودن را داشته باشند، جوابش با سلسله مراتبی تفاوت چندانی ندارد، و تقریبا به یک جواب می رسد اگر در سلسله مراتبی از norm اقلیدسی استفاده کنیم و آخرم مسئله را با چهار روش سعی کردیم حل کنیم البته این بیشتر جنبه ی آموزشی داشت نه پژوهشی، که بدانیم یک مسئله را واقعا ما می توانیم با چهار روش خیلی از جاها حل کنیم: به روش دقیق - با روش ریاضی - با EM الگوریتم و نهایتا با استفاده از شبیه سازی.

اینکه ما چطوری این کار را در Model Based clustering انجام می دهیم از حوصله ی کلاس و درس ما خارج است. ولی ما باید بتوانیم با نرم افزار این کار را انجام دهیم، یعنی استاد ما نمی خواهد بایستی قبلا یک برنامه بنویسیم که Model Based clustering انجام بدهد، ممکن است که ما در درس clustering همچین کاری انجام بدهید ولی در درس داده کاوی ما نمی خواهیم این قدر ریز بشویم و در clementine میتوانیم این کار (Model Based clustering) را انجام دهیم. ولی سلسله مراتبی را استاد میخواهد که با دست انجام دهیم. امروز می خواهیم نوع سوم clustering بپردازیم البته انواع clustering سه روش نیست، سه روش عمده اش را ما به ترتیب می گوییم: اول = سلسله مراتبی، دوم = دقیق، سوم = Model Based، چهارم = آماری

پنجم = k-means که یکی از تکنیک های قبل از آماری است ولی این روش هم اگر که داده های ما از توزیع نرمال تبعیت کنند و norm ای که در نظر میگیریم، norm اقلیدسی باشد توی این k میانگین و یا اینکه فاصله ای که در این k میانگین در نظر میگیریم هم فاصله اقلیدسی باشد، آن موقع جواب این روش هم یکی می شود ولی خوبی k-means این است که سرعتش از Model Based هم بیشتر است، به خاطر همین هم است که خیلی وقتها با الگوریتم در بحث شروع clustering همیشه ما با k-means شروع می کنیم، پس امروز ما میخواهیم روش سوم پارتیشنینگ را در مورد روش صحبت کنیم، روش سوم که پارتیشنینگ است.

و k میانگین یکی از روش های پارتیشنینگ است. چطوری ما داده ها را کلاستر میکنیم؟! با k میانگین!؟ مستقیم ترین روش کلاسترینگ k میانگین است. سریع ترین روش کلاسترینگ هم میباشد و متاسفانه کم دقت ترین روش هم می باشد که البته کم دقت ترینش بستگی دارد حالا بعدا می گوییم که بستگی به چی دارد؟! هدف ما در clustering چیست؟! ما n تا نقطه داریم در فضای R^d میگوییم آنها را تقسیم شان کنید به k تا، دقت کنید که اگر k معلوم باشد گفتیم که بخش‌بندی میکنیم؟

classification مثل بحث clustering نیست. ولی clustering برای k معلوم خیلی مسئله ای پیچیده ای است بخاطر همین ما k را معلوم میگیریم و برای k های مختلف این را انجام می دهیم، بعد تصمیم گیری میکنیم چه k ای انتخاب کنیم! گفتیم که در روش ها معمولا k را بین ۲ یا ۳ در نظر میگیرند و بعد بسته به مسئله، حالا از یک هم همه شروع نمیکنند، از یک جایی شروع میکنند، فقط تصمیم میگیرند که کدام cluster بهتر است با آن چند روشی که در سلسله مراتبی گفتیم. حالا اگر C(x) نزدیکترین کلاستر مرکز x باشد آن موقع k مرکزی می باشد $\min_{x\in X}\max\|x-C(x)\|$ ، یعنی ما دنبال یک کلاستری میگردیم که کلاسترهای C(x) (center) است که x های C(x) با حداقل فاصله تا مرکز کمترین باشد، و فاصله بین خوشه ها باید بیشترین باشد، بخاطر همین minmax قرار داده، اگر ما بتوانیم این کار را انجام بدهیم اسم این میگوییم k-Center . بلکه انتظار خواهد between و witween را نگویید، اگر بخواهیم k-median را حساب کنیم، دقت کنید Midean چی بود؟ میانه است، میانه توی ۲ بعدی تعریف نشده ولی میانگین در ۲ بعد تعریف شده، یعنی اگر ما به فرض چند تا عدد داشته باشیم توی یک بعدی، بخواهیم میانه اش را پیدا کنیم، میاییم اعداد را مرتب میکنیم، وسطی را به عنوان میانه در نظر میگیریم، این میشه میانه ولی اگر اعداد ما دو بعدی باشند، ما نمی توانیم میانه را در ۲ بعد تعریف کنیم، پس مسئله ای که ما در اینجا برخورد میکنیم، این مسئله k-median اول median ای نیست که ما در آمار خواندیم، شبیه آن است ولی آن نیست، چون اولا در حالت چند بعدی اصلا تعریف نشده. k-median را تعریف میکنیم، این یک تعریف جدید است و آن چیزی (median) تعریف قبلی نیست. k-median را تعریف میکنیم که میگیم x های متعلق به x، norm(x−C(x)):

$$k\text{-}median = \min \sum_{x\in X} \|x - C(x)\|$$

در قبلی اول میگفتیم ماکزیمم این norm هاست، یعنی برای تمام x های متعلق به X، x−C(x) را حساب میکنند بعد ماکزیممش را که حساب کردیم، مینیمم اش را در نظر میگیریم ولی در اینجا ما می آییم سام‌هایشان x−C(x) را حساب میکنیم، norm را حساب میکنیم بعد سام هایشان norm را می گیریم، بعد مینیمم آن را حساب میکنیم. هنوز الگوریتم که چطور این کار را انجام بدهیم نگفتیم. اول نقطه وسطی را چگونه پیدا میکنیم؟ به صورت Random فرض میکنیم نقطه وسطی باشد

More on www.faruk.ba

Article Top

Comments and Discussions

Add a Comment or Question

Search this forum Go

Profile popups Spacing Relaxed Noise Medium Layout Normal Per page 25 Update

-- There are no messages in this forum --

توجه کنید که در k_median ها دیگه minmax را در نظر نمی‌گیریم، به جای اینکه minmax بگیریم می‌آییم min می‌گیریم
سامیشن را می‌گیریم و هوامان باشد که این اسمش k_median است و در چند بعد می‌توان آن را تعریف کرد.
k-Median Square نرم را به توان ۲ می‌رسانند: k-median Square : $\min \sum_{x \in X} \|x-C(x)\|^2$
قبلی بهتر بود که پرزور بود یعنی اگر داده‌ی پرت داشتیم خیلی حساس نبود ولی این یکی حساس است نسبت به داده‌های
پرت. و بعضی وقتها از این استفاده می‌کنند البته معمولاً قبلی بهتر است. حالا k میانگین چطوری است؟!
ص6: ما می‌خواهیم این داده‌ها را به ۳ گروه تقسیم کنیم، به ما می‌گویند k=۳ است، میگه بیا سه تا از داده‌ها را به
تصادف انتخاب کن، این اولین نکته که نقطه‌ی سیاه در k میانگین است، یعنی چون که می‌خواهیم روش یک روشی باشد
که ماشین بتواند پیاده‌سازی بکند، نباید ذهنی باید یک طوری این سه عدد اولیه را انتخاب کنیم، نمی‌توانیم توش دست کاری کنیم،
ساده‌ترین راه این است که بیاییم به صورت تصادفی انتخابشان کنیم، این دو تا ایراد دارد توی k میانگین: اگر
این سه عدد معمولا این بار در یک خوشه انتخاب بشوند، k میانگین به بدترین جواب ممکن می‌رسد متاسفانه و ایراد
بعدی این روش هم این است که هر دفعه ما این روش را اجرا کنیم، به یک جواب می‌رسد یعنی ممکن است ما دو دفعه
این کار را انجام بدهیم و دو تا جواب متفاوت بدست آوریم.

(6)

در حالی که در سلسله مراتبی همچین چیزی امکان ندارد، توی مدل بس امکان داره، اما امکانش خیلی خیلی ضعیف است اما توی K میانگین احتمالش همچین ضعیف نیست، پس عدد تصادفی یک مشکل‌ها را حل میکند، روشش را اتوماتیک می‌کند، که نکنیم یک ماشین برامان انجام دهد، ولی دوتا ایراد قبل را دارد. ایراد اول قابل رفع است، یعنی ما میتوانیم اگر داده‌هایی را که انتخاب کردیم، فاصله‌شان از یک حدی کمتر بود، بگیم داده‌های تصادفی را دوباره انتخاب کن، این کار را معمولاً اول همه‌ی برنامه‌های K میانگین انجام می‌دهیم ولی در مورد این که اگر همه در خوشه‌های متفاوت انتخاب شدند، ولی امکان داشته باشه که جواب‌های K میانگین‌ها بهینه نباشد، باز هم وجود دارد. البته این چه فایده‌ای دارد این است که معمولاً توی روش‌های پیچیده نیاز به یک خوشه بندی اولیه داریم. K میانگین بهترین روش خوشه بندی اولیه است یعنی خروجی K میانگین میشه ورودی الگوریتم‌های دیگه ما. پس این نیست که K میانگین روش بدی باشد بلکه نه تنها بد نیست، اگر که این مقادیر اولیه را خوب انتخاب کنیم، K میانگین خوبه. روشی که K میانگین محاسباتش را انجام می‌دهد براساس EM الگوریتم است، یعنی یک روش بین ریاضی و آماری است. K میانگین هم با دست راحت نیست انجامش دهیم. پس اولین قدم در الگوریتم K میانگین انتخاب K نقطه به تصادف است. بعد چیکار میکنیم؟! بعد می‌آییم فاصله‌ی تمام نقاط را تا آن سه مرکزی که انتخاب کردیم حساب میکنیم. فاصله هر نقطه که به هر مرکز نزدیک‌تر بود (36) آن نقطه را جزء آن خوشه در نظر میگیریم، پس خیلی راحت است، تنها سوالی که پیش می‌آید اینه چرا این نقاطی که به تصادف انتخاب کردیم، مرکز شدند؟ آن مرکز یک شرطی داشت، اون شرط را اینجا اعمال کردیم؟! بله، ما فعلاً مرکز اولیه را پیدا کردیم، این بهترین مرکز نیست، باید بیاییم الان بهترین مرکز را حساب کنیم (فعلاً اقلیدسی و هرچی میخواهید میتوانید فاصله‌ها را حساب کنید) حالا می‌آییم بهترین مرکز را پیدا میکنیم، دقت کنید که لازم نیست مرکز از خود داده‌ها باشد، مثل میانگین است مثلاً در بازی تاس اگر ما تنها یکبار بخواهیم تاس را بیندازیم احتمال 3.5 تومان ما برنده میشویم، یعنی متوسطها را آمدیم حساب کردیم. 3.5 در اعداد تاس نیست ولی به طور متوسط ما 3.5 تومان برنده هستیم اگر در بازی تاس شرکت کنیم. اینجا هم همین است الان ما دوتا داده داشتیم در خوشه ولی مرکز مشخص نبود، پس ما باید مرکز اصلی‌شان را حساب کنیم، پس مرکز اصلی داده‌ها، اول اعداد تصادفی انتخابی ما بود، خود این مراکز، یک مراکز بی ربطی هستند که پس می‌آییم دوباره مراکز را حساب میکنیم، حالا دوباره می‌آییم فاصله تمام نقاط را تا مرکز حساب میکنیم، فاصله‌ی هر نقطه تا مرکز که کمینه بود، آن را جزء همان خوشه می‌گیریم، ممکن است یک داده که همان طور که می‌بینید از خوشه دوم یک داده پریده رفت تو خوشه اول و یک داده از خوشه سوم پرید رفت تو خوشه دوم، چون فاصله را با مرکز جدید حساب کردیم. (صفحه ۹ و ۱۵)

حالا دوباره می آییم مرکز جدید را مجدد حساب میکنیم، مرکز جدیدی که الان تشکیل شده، دوباره مرکز عوض میشود، حالا دوباره می آییم فاصله ی تمام نقاط را تا مرکز جدیدی که پیدا شده، محاسبه می کنیم تا ببینیم چه تغییری رخ می دهد، تغییر رخ می دهد یا رخ نمی دهد؟! این کار را تا کی ادامه می دهیم؟! تا وقتی که مراکز تغییر نکنند، این هم شد روش k means. پس مراکز اولیه را تنها به صورت تصادفی انتخاب کردیم و مراکز بعدی را به صورت تحلیلی بدست می آوریم. حالا جالب اینکه این الگوریتم را بتوانیم با کامپیوتر پیاده کنیم با توجه به فاصله ای که در اختیار داریم، برنامه نویسی مان متفاوت میشود. سوال!! آیا روش های دیگر هم داریم؟! بیشتر ما می خواهیم به صورت unsupevise کار کنیم، اگر بخواهیم supervise کار کنیم، بله، می توانیم هیستوگرام داده ها را رسم کنیم، خودمان یک Threshold (حد آستانه) تعیین می کنیم، وسط آنها را می بریم به عنوان مراکز می دهیم پس می توانیم این کار را انجام بدهیم. اسلایدهای دیگر بحث های تخصصی داره انجام می دهد که بد هم نیست ببینیم، مثلاً گفته چقدر این الگوریتم سریع است؟ گفته خیلی سریع است، یعنی بعد از iteration های خیلی کمی همگرا می شود و واقعاً هم همین جوراست و در مسئله ی image processing شاید توی ۱۰ تا iteration همگرا شود. از این به بعد بیشتر صحبتهای ریاضی در اسلایدها شده که ما واقعاً باهاش کاری نداریم، پس ما تا همان اسلاید 12 را کفایت میکند برایمان. در spss تنها خروجی این الگوریتم نشان داده میشود و مراحل نمایشش داده نمی شود.

اسلاید پاورپوینت 8clst

میخواهیم در مورد clustering کمی بیشتر صحبت کنیم، دلیل اش هم این است که ممکن است داده هایی که ما با آنها سروکار داریم، همه ی داده های پیوسته نباشد و بعد برای clustering اینجور داده ها نتوانیم از روش های معمول استفاده کنیم. اول بحث clustering را مطرح کرده، که ما باهاش آشنایی داریم، پس کاربردهایش را گفته که اینها را قبلاً گفتیم، چند تا مثال هم برایمان زده و cluster خوب را تعریف کرده. قبل بحث خوب، وقتی گفتیم کلاسترینگی خوب است که داخل کلاس ها واریانس ها کم باشد و بین کلاس ها واریانس ها زیاد باشد و کیفیت کلاسترینگ مان هم به measure ای دارد که انتخاب کردیم و روش مان یعنی ما همیشه یک measure ای توی روش مان داریم و روش های مختلف هم داریم. یک measure را همینطور در روش های مختلف به کار برد. اگر این دو تا را خوب انتخاب کرده باشیم بهترین جواب را خواهیم گرفت که البته اینجا ادامه هم داده که گفته کیفیت clustering خوب هم اون clustering ای خوبه است که بتواند الگوهای پنهان را برای ما کشف کند و واقعاً در داده کاوی هم همین است یعنی یک وقتهایی به عنوان مثال استان های کشور را وقتی که ما cluster

می کنیم، شاید که برایتان خیلی جالب باشد، تهران با ۳ استان مثلاً هم رده میشود؟! با همین شهر و علاوه بوشهر، در صورتی که
به نظر ما شاید که در ... علاوه بر بوشهر و همین شهر در اصفهان هیچ شباهتی بین این دو تا شهر و تهران نمی بینند ولی یک چیزی
که در این سه تا خیلی پررنگ است، درآمد زیاد این سه شهر است یعنی از تمام شهرهای کشور این سه شهر درآمدشان بیشتر است.
جالب هم این است که وقتی clustering انجام می دهیم اینها را هم اندازه یک خوشه براساس معیارهای مختلفی موجود دارد از جمله
معیارهای فرهنگی، سلامت، آموزش و ... و درآمد هم جزء آن است ولی درآمد آنقدر مهم است که بقیه متغیرها را
(صفحه 7)
Dominate می کند (تحت تأثیر خود قرار می دهد). ما روش clustering ای که میخواهیم الزاماً (هم) باید این ۹ تا ویژگی
را بتواند توی خودش داشته باشد، یعنی اگر scale داده ها را عوض کنیم کلاسترهایمان تأثیری نداشته باشد، تغییر رخ
ندهد، Attribute های مختلفی اگر در بر داشته باشیم، بتواند کلاستر کند، اگر کلاسترهای ما اشکال مختلفی داشته
باشند چون به عنوان مثال اگر داده های ما (عیب دیگر k میانگین) که چون روی مرکز داره صحبت میکند، اگر خوشه
های ما به صورت بیضوی و دایره نباشند، k میانگین افتضاح به بار می آورد، کافی است به شکل U باشد، این U را
دایره حساب میکند و خیلی از داده های دیگر را هم قاطی این داده ها میکند، پارامترهای ورودی اش نباید زیاد باشد، باید بتواند
nosie و outliers یعنی اغتشاشات و داده های پرت را بتواند باهاش کار کند، نسبت به داده های ورودی حساسیت زیادی
نداشته باشد، با ابعاد بالا بتواند کار کند، اگر شرطی، محدودیتی توسط کاربرمان اعمال شد بتواند آنها را در نظر بگیرد و آخرش
هم قابل تفسیر و استفاده باشد چون ممکن است یک cluster ای را انجام دهیم، قابل تفسیر نباشد، ساختار داده های ما
در clustering بصورت یک ماتریس بود، ما p تا متغیر داریم، n تا عدد داریم و Dissimilarity matrix مان را
هم همان طور که در صفحه ۹ می بینید درست میکنیم، چطوری این quality clustering مان را اندازه گیری می کنیم؟!
خوب گفتیم با D_{ij} نشان می دهیم که فاصله بین نقاط را به ما نشان می دهد، $D_{ij}=0$ شود یعنی فاصله از هم
دیگر ندارند، تنها چیزی که باید توجه کنیم این است که این D بسته به نوع داده هایمان باید تعریف مناسبی داشته باشد، یعنی
اگر داده های ما مثلاً باینری باشند یا Categoricall باشند یعنی تعدادی باشند یا ordinal باشند اینها D ای که تعریف
میکنیم متفاوت است با D ای که مثلاً در داده های پیوسته که به صورت اقلیدسی باشند. این جا در جدول داده ای که مشخص
است ما با هاشور مشخص کردیم را اینجا گفتیم (صفحه ۱۱) یعنی مقیاس نشان نسبی یا فاصله ای باشد - باینری باشند -

اسمی آخرش یا نسبتی باشند و یا مختلط باشند. برای داده های معمولی تجربی فاصله مان نرم اقلیدسی نیست، قدر مطلق از norm اقلیدسی بهتر است چرا بین ما از norm اقلیدسی بیشتر استفاده میکنیم؟! چون که محاسباتش ساده‌تر است و الا قدر مطلق با عقل سلیم بیشتر جور در می‌آید. بعضی جاها لازمه کارهای محاسباتی مان ساده‌تر شود. ما معمولاً داده هایمان را استاندارد میکنیم، داده های استاندارد یعنی منهای میانگین تقسیم بر جذر واریانس اش میکنیم، این به ما این امکان را می‌دهد که داده هایی داشته باشیم که میانگین شان صفر و واریانس شان ۱ باشد (صفحه 12) حالا به جای standard deviation یعنی وقتی می‌آییم mean absolute deviation می‌گذاریم. mean absoluted deviation همان S_f است.

$$s_f = \frac{1}{n}\left(|x_{1f}-m_f| + |x_{2f}-m_f| + \cdots + |x_{nf}-m_f|\right)$$

$$m_f = \frac{1}{n}\left(x_{1f}+x_{2f}+\cdots+x_{nf}\right) \qquad z_{if} = \frac{x_{if}-m_f}{s_f}$$

اگر داده ی پرت داشته باشیم این بهتر از استاندارد deviation عمل می کند. مترمین کوسکی را که قبلاً در درس صحبت کرده بودیم، که حالت کلی‌تر از نرم اقلیدسی است و یک D فاصله‌مان باید یک همچین خاصیتی داشته باشد، در این چهار تا شرط صدق کند (صفحه 13، 14) که اینجا داره توضیح می‌دهد: ما میتوانیم این Distance هایمان را خیلی کلی ترکیب کنیم. مثلاً به صورت Corelation که البته دیگه آن موقع به جای Dissimilarity خود similarity را به ما می‌دهد. اگر داده های ما Categorical باشند، یعنی a، b، c و d تعداد اند (صفحه 15)

	1	0	sum
1	a	b	a+b
0	c	d	c+d
sum	a+c	b+d	p

این تعداد را ما میتوانیم به دست آوریم. و حالا انتخاب D که تعریف میکنند، که simple matching coefficient که گفته، که گفته اگر symmetric binary variable ما باشد آن موقع

$$d(i,j) = \frac{b+c}{a+b+c+d}$$

اگر هم بخواهیم noninvariant هم باشد آن موقع $d(i,j) = \frac{b+c}{a+b+c}$.

یک مثال زده که این مثال برایمان جدید است (صفحه 16) جنسیت که یک صفت symetric است ولی بقیه دیگر symetric نیست، چند تا متغیر داریم؟! اینجا 7 تا متغیر داریم مثلاً d(jack, Mary) را که میخواهیم حساب کنیم برای کدام متغیر میخواهیم حساب کنیم؟! چرا d = .33 شد؟! فرض کنید ما به جای Y و P را بگذاریم 1 و N را بگذاریم صفر. آن موقع وقتی میخواهیم فاصله jack و mary را حساب کنیم، اگر binary variable ما symetric باشد، $d = \frac{b+c}{\text{کل اش}}$ ولی اگر symetric نباشد $d = \frac{b+c}{a+b+c}$. اینجا هم ما همین کار را انجام می‌دهیم.

چون فقط اولی symmetric است، جنسیت را اگر بخواهیم بگذاریم کنار، باید این مسئله را به یک جدول ۲ ابعادی تبدیلش کنیم. دقت کنید اینجا جدولمان ۲ بعدی است ولی ۲ تا مقدار دارد، اونجا هم ۲ بعدی است ولی مقادیرمان زیادتر است. ما هر متغیر را می‌توانیم به عنوان سطر بگیریم، بقیه را به عنوان ستون بگیریم، یعنی به همان جدول ۲ ابعادی درش بیاوریم. اگر بخواهیم به جدول ۲ بعدی درش بیاوریم که فراوانی‌هاشان را بنویسیم، اون موقع می‌تواند این را حساب کند. (میخواهیم این را به عنوان تمرین بگذاریم برای جلسه بعد) جلسه بعد به استاد یادآوری کنید که اسلاید ۱۶ را دوباره بگویند و خودشان حساب کنند.

اگر متغیرها اسمی باشند، اون موقع فاصله را به این صورت حساب میکنیم: تعداد اونهایی که با p برابرند تقسیم بر کل اش

m را نوشته تعداد اونهایی که با هم برابرند مثل وقتی که دارید احتمال

$$d(i,j) = \frac{p-m}{p}$$

حساب میکنید، یک همچنین حالتی دارد. پس اگر متغیرها اسمی بود، وقتی میخواهیم برای متغیر اسمی فاصله را حساب کنیم، فاصله یک تعبیری شبیه احتمال دارد. اگر متغیرها ترتیبی باشند، رتبه‌ای باشند، اگر رتبه‌ای باشند، دوباره برای محاسبه‌ی فاصله اول یک تغییر متغیر می‌دهیم، سپس از روش‌های معمول برای فاصله حساب میکنیم:

$$r_{if} \in \{1, \dots, M_f\}$$

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

پس اگر متغیرها ترتیبی بود، ما می‌آییم بر حسب اینها مرتبش می‌کنیم، ترتیبی که هست یعنی مثلاً a، b، c، d و e و f و g اینها کاراکتر هستند ولی ترتیب دارند، یعنی از نظر ما مثلاً برای sort کردن اینها به ترتیب باشند، ما نمی‌توانیم با متغیرهای a و b ما خودشان کار کنیم توی فرمول بگذاریمشان، با rank اش کار می‌کنیم، میگوییم rank a اش ۱ است در حروف انگلیسی، b رنکش ۲ در حروف انگلیسی و ... حالا می‌آییم این اعدادی را که به حروف نسبت دادیم، براساس این اعدادی که بهشان نسبت دادیم، این z را درست می‌کنیم، حالا این z ای که درست میکنیم براساس rank منهای ۱ تقسیم بر $M-1$ که $M-1$ بزرگترین rank مان است. این z_i که به دست آوردیم، با این z_i ها می‌آییم با روش معمولی فاصله‌اش را حساب میکنیم. اگر متغیرهامان نسبتی باشند، اون موقع ما می‌توانیم یک تغییر متغیر بدهیم با هاش کار کنیم. البته تا اون جایی که می‌توانیم سعی میکنیم تغییر متغیر استفاده نکنیم، ولی خوبی تغییر متغیر این است که مثلاً لگاریتم که بگیریم، برای داده‌هایی که خیلی بزرگ هستند را خیلی کوچک می‌کند، داده‌های کوچک را کمتر کوچک میکند، داده‌ها حالت نرمال پیدا می‌کنند. وقتی که داده‌ها نرمال بشوند، تکنیک‌های خوشه‌بندی خیلی بهتر عمل میکنند. بعد که خوشه‌بندی را انجام دادیم، می‌توانیم دوباره، یک تبدیل معکوس بزنیم روی داده‌ها تا

به داده های اولیه برگردیم، در حالت معمولی سعی میکنیم ما هیچ وقت این کار را انجام ندهیم بر روی داده ها تبدیل نزنیم اگر داده ی پرتی چیزی داشته باشیم این کار را انجام می دهیم، ولی مهندس ها غالباً این کار را انجام میدهند یعنی وقتی می خواهند خوشه بندی انجام دهند، یک تبدیل روی داده ها می زنند مثلاً می بارزش توی فاصله ی صفر و یک، که بتوانند خیلی راحت تر با هاش کار کنند، هم در برنامه نویسی ساده تر است و هم underflow و overflow از این چیزها خیلی کمتر می بینیم. اگر متغیری که داریم نوعش فاصله باشد یعنی چون multivariate هستند، یکی اش مثلاً عددی باشد، یکی فاصله ای باشد، ما اینها را معمولاً از این فرمول استفاده میکنیم (صفحه ۲۵)

$$d(i,j) = \frac{\sum_{f=1}^{p} \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{p} \delta_{ij}^{(f)}}$$

فعلاً ما همان قبلی ها را در نظر بگیریم، فقط توی ذهنمان باشد که اگر فاصله ای بودند هم میشه دوباره اینها را خوشه بندی کرد.

انواع روش های خوشه بندی را صفحه ۲۱ گفته که قبلاً در موردش صحبت کردیم Partitioing Methods همان k میانگین است که می خواهیم جلسه بعد بیشتر در موردش صحبت کنیم. پیشنهاد استاد = سعی کنید یک الگوریتم k میانگین با هر نرم افزاری که بلدید یا پیدا میکنید داده را خوشه بندی کنید.

TIME SERIES DATA MINING - 8dataseries.ppt 92/8/13 جلسه 9