درس داده کاوی

دکتر عمرانپور

منبع: Introduction to Datamining - Jiawei Han

نمره دهی درس:

میان ترم: ۵ نمره ← بعد از تدریس ۵-۴ فصل

پایان ترم: ۶ نمره

تمرینات: ۴ نمره

ارائه: ۳ نمره ← از میان یکسری مقالات مشخص موضوع تأیید شود و پس ارائه شود. پس یک مقاله دیگر هم باید انتخاب و ارائه شود. (مجموعاً باید دو مقاله ارائه شود و با هم مقایسه شود و در نهایت پیشنهادی در رابطه با بهبود عملکرد مقاله‌ای داده شود!)

(می‌توانیم مقالات را از وبسایت‌های دریافت کنیم که کد عملی برنامه هم موجود باشد. احتمالاً باید در Github یا codeocean موجود باشد.)

پروژه پایانی: ۳ نمره

نرم افزارها و زبان های مورد استفاده:

- MATLab
- Python

در روش بررسی پروژه پایانی، می‌توانیم روشی را که جهت بهبود مقاله‌ی ارائه انجام دهیم، به صورت پیاده سازی نرم افزاری درآورده و بررسی که به صورت عملی صورت گیرد.

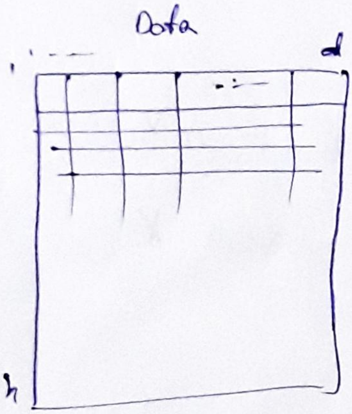
- داستان مربوط به ماهیگیری و دسته بندی ماهی ها مطرح شد. (توضیحات مربوط بسیار زیاد بود ولی مجموعاً خلاصه‌ای از تمامی مباحث مربوط به این درس مطرح شد که سرفصل های اصلی درس را مطرح می‌کرد.)

داده کاوی

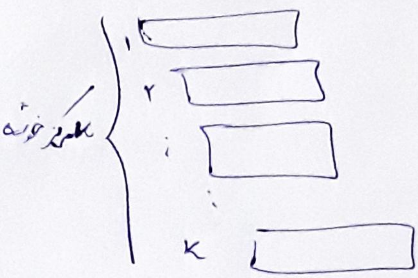
۱۴۰۱/۷/۱۰ : در جلسه قبل درباره داستان یادگیری و دسته‌بندی صحبت شد. درباره ann و feature ها صحبت شد.

۱۴۰۱/۷/۱۲ : درباره مقالات ISI کنفرانسی و ژورنالی صحبت شد اما چیزی تدریس نشد.

۱۴۰۱/۷/۱۷ :



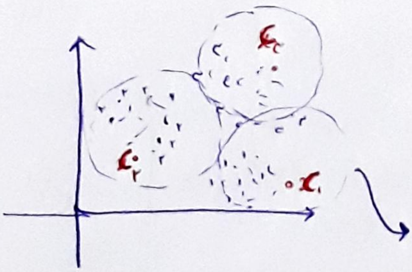
روش خوشه‌بندی K-means :

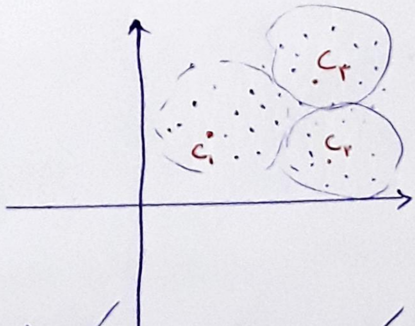


۱- مقداردهی تصادفی به K مرکز خوشه

۲- برچسب گذاری داده‌ها بر اساس نزدیک‌ترین مرکز خوشه

۳- مراکز خوشه جدید تولید می‌کنیم (میانگین داده‌های یک برچسب)



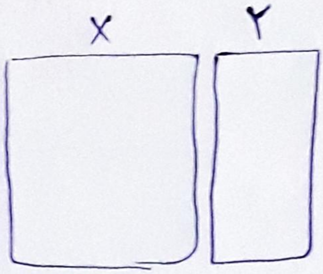


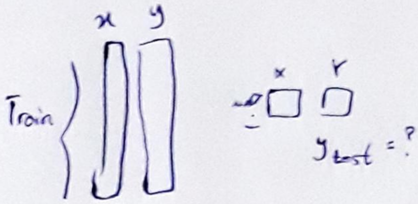
۴- مرحله ۲ و ۳ آنقدر تکرار شود تا برچسب‌ها یا مراکز خوشه‌ها در ۲ تکرار متوالی تغییر نکند.

K-medoid

روش خوشه‌بندی K-~~medio~~id : مانند روش قبل است با این تفاوت که مراکز خوشه‌ها از میان نقاط موجود در میان نقاط داده‌ها انتخاب می‌شود. (در روش اول، نقاط مرکز مجازی هستند)

⑧ رگرسیون خطی، هدف رسیدن به رابطه‌ای چون $Y = \alpha + \beta X$





$\bar{Y}$: میانگین Y های داده های Train

$\bar{X}$: میانگین X های داده های Train

$$\alpha = \bar{Y} - \beta \bar{X}$$

$$\beta = \frac{\sum_{i=1}^{n} \cancel{(x_i - \bar{x})}(Y_i \cancel{- \bar{Y}})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

۱۴۰۱/۷/۱۹

انواع داده:

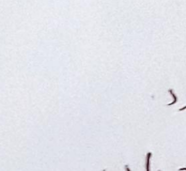
کمی: قابل شمارش و یا اندازه‌گیری هستند
با یک عدد نمایش داده می‌شوند.
[فاصله در این نوع متغیر قابل تعریف است]

داده متغیر: گسسته / پیوسته

کیفی: بر اساس یک ویژگی خاص در دسته‌های مختلف قرار می‌گیرند.
Category / ویژگی
[فاصله قابل تعریف نیست]

- ترتیبی (ordinal): رتبه دانشجویان ← عملگر مقایسه $> <$ $=$ دارد
- اسمی / اسم (nominal): دسته‌بندی حیوانات ← فقط عملگر $=$ و $\neq$ دارد

ابعاد داده‌ها: فرض کنیم ۱۰۰ داده در یک بعد داریم. اگر بخواهیم با حفظ چگالی این داده‌ها را به ۵ بعد تغییر دهیم به 100^5 داده نیاز خواهیم داشت.



برای بدست آوردن الگوی میان داده‌ها، هرچه ابعاد بالاتر باشد، نیاز به داده‌های بیشتری است.

مفاهیم آماری:

شاخص های مرکزی

۱. میانگین

- حسابی: $M_A = \frac{\sum_{i=1}^{n} x_i}{n}$
- هندسی: $M_G = \left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}}$
- هارمونیک: $M_H = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}$
- وزنی: $M_w = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$

و $M_H \leq M_G \leq M_A$

۲. میانه (داده میانی)

۳. مد (نما) → داده با بیشترین مورد

شاخص های پراکندگی

۱. دامنه تغییرات: $Max(x_1, x_2, \dots, x_n) - Min(x_1, x_2, \dots, x_n)$

۲. میانگین انحرافات: $\frac{1}{n} \sum_{i=1}^{n} |x_i - M_A|$

۳. واریانس: $\delta^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - M_A)^2$

۴. انحراف معیار: $\delta = \sqrt{\delta^2}$

تمرین: از هر مورد از میانگین گیری‌ها یک کاربرد مثال بزنید.

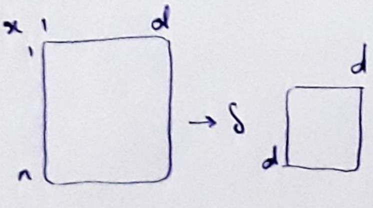
شاخص های برای دو ویژگی:

- کوواریانس: $Cov(x,y) = \frac{1}{n} \sum_{i=1}^{n} (x_i - M_x)(y_i - M_y)$

(در مفهوم می‌تواند رابطه بین دو ویژگی را نشان دهد)

- ضریب همبستگی: $Correlation(x,y) = \frac{Cov(x,y)}{\sqrt{Var(x) . Var(y)}}$

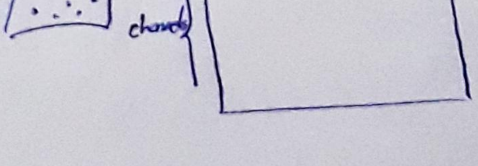
ماتریس کوواریانس ویژگی:



$$S_{i,j} = \frac{1}{n} \sum_{k=1}^{n} (x_{ki} - M_i)(x_{kj} - M_j)$$

تمرین: پیاده سازی یک سری Data مرتبط با کانال های سیگنالی (داده‌ها از جایی پیدا شود). کوواریانس برای این داده‌ها محاسبه شود و اطلاعات فیچرهای انتخاب شده به یک دسته‌بندی کننده K-NN داده شود.

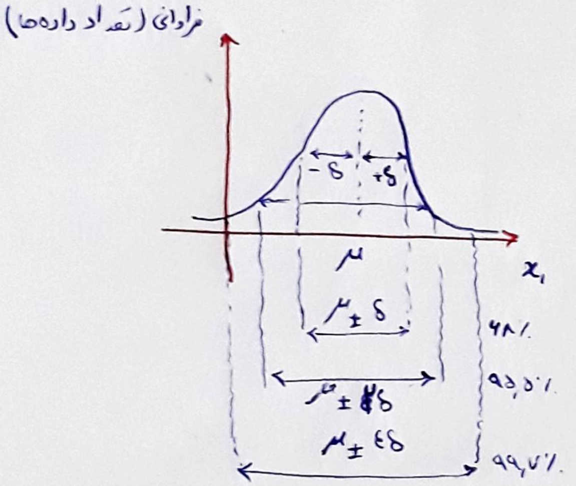
تشخیص سیگنال



نکته: سیگنال ها به صورت Mono نباشد و چند کاناله باشد.

مثل :

Correlation α = (−)
Correlation α = (+)
Correlation – = 0



(*) داده های نرمال

میانگین = μ

انحراف معیار = δ

(*) داده های نرمال استاندارد

$\mu = 0$

$\delta = 1$

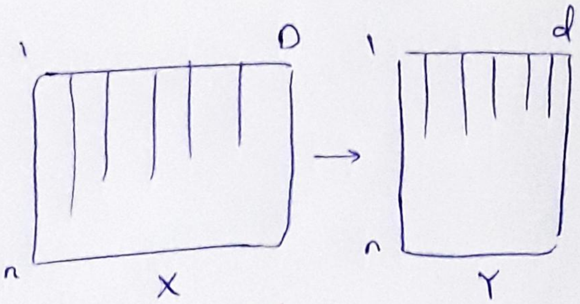
مثال: فرض کنیم داده های x_i که شامل $\{x_1, \dots, x_n\}$ و نرمال هستند را در اختیار داریم. آن‌ها را به فرم نرمال استاندارد تبدیل نماییم.

$$y_i = \frac{x_i - \mu_x}{\delta_x}$$

$$\mu_y = \frac{\sum_{i=1}^{n} y_i}{n} = \frac{\sum_{i=1}^{n} \left(\frac{x_i - \mu_x}{\delta_x}\right)}{n} = \frac{1}{\delta_x}\left(\sum_{i=1}^{n} \frac{x_i}{n} - \sum_{i=1}^{n} \frac{\mu_x}{n}\right) = \frac{0}{\delta_x} = 0$$

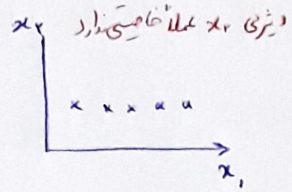
$$\delta_y = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \mu_y)^2}{n}} = \sqrt{\frac{\sum_{i=1}^{n} (y_i^2 - 2y_i\mu_y + \mu_y^2)}{n}} = \sqrt{\frac{\sum_{i=1}^{n} y_i^2}{n}} = \sqrt{\frac{\sum_{i=1}^{n} \left(\frac{x_i - \mu_x}{\delta_x}\right)^2}{n}}$$

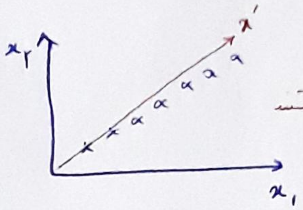
$$= \sqrt{\frac{\sum (x_i - \mu_x)^2}{\delta_x^2\, n}} = \frac{1}{\delta_x}\sqrt{\frac{\sum (x_i - \mu_x)^2}{n}} = \frac{1}{\delta_x} \times \delta_x = 1$$



کاهش ابعاد:

- خطی
- غیرخطی





x_2, x_1 هر دو مهم هستند

- انتخاب ویژگی
- استخراج ویژگی

① روش PCA برای استخراج ویژگی:

تحلیل مؤلفه‌های اصلی (Principle component Analysis)

۰. (معمولاً) ابتدا میانگین داده‌ها را صفر می‌کنیم.

۱. ماتریس کواریانس ویژگی‌ها را بدست آورید. $C_{D\times D}$

۲. بردارهای ویژه و مقادیر ویژه ماتریس C را محاسبه می‌کنیم.

$$C \times E_{vector} = \lambda \times E_{vector}$$

۳. معادله $|\lambda I - C| = 0$ را حل می‌کنیم، مقادیر ویژه را مرتب می‌کنیم. (به صورت نزولی)

$$\lambda_1 > \lambda_2 > \lambda_3 \dots > \lambda_D$$

$$\begin{vmatrix} \lambda I - c_{11} & -c_{12} & \cdots \\ -c_{21} & \lambda I - c_{22} & \\ \vdots & & \lambda - c_{33} \\ & & & \ddots \end{vmatrix}_{D\times D}$$

D معادله D مجهولی

↓

λ را در رابطه بالا قرار دهیم تا E_{vector} ها بدست آوریم.

۴. d تعداد ابعاد فضای جدید ممکن است توسط کاربر مشخص شود.

اگر توسط کاربر مشخص نشود:

$$\frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{D} \lambda_i} \geq 0.95$$

d را از طریق فرمول مقابل بدست می‌آوریم.

یعنی ۹۵ درصد از خصوصیات داده اصلی حفظ شود.

در نهایت به معادله $Y_{n\times d} = X_{n\times D} \times W_{D\times d}$ می‌رسیم.

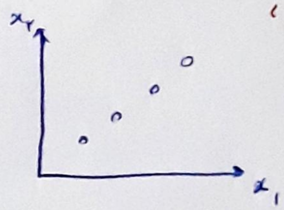
نکته، بردارهای ویژه باید برهم متعامد هستند (به عبارتی با هم وابستگی ندارد و Correlation آن‌ها صفر است)

مثال،

Data

۱	۱
۲	۲
۳	۳
۴	۴
۵	۵
۶	۶
۷	۷
۸	۸
۹	۹
۱۰	۱۰

برای این قسمت حل می‌کنیم



$$cov(x_1, x_2) = \sum_{i=1}^{5} \frac{(x_{i1}-\mu_1)(x_{i2}-\mu_2)}{5} = \frac{4+1+0+1+4}{5} = 2$$

$$C = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}, \quad C \times E = 4 \times E \longrightarrow \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}\begin{bmatrix} E_1 \\ E_2 \end{bmatrix} = 4\begin{bmatrix} E_1 \\ E_2 \end{bmatrix}$$

$$\longrightarrow \begin{matrix} 2E_1 + 2E_2 = 4E_1 \\ 2E_1 + 2E_2 = 4E_2 \end{matrix} \Rightarrow E = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \; E_1 = E_2 \longrightarrow$$

طول E باید برابر یک باشد

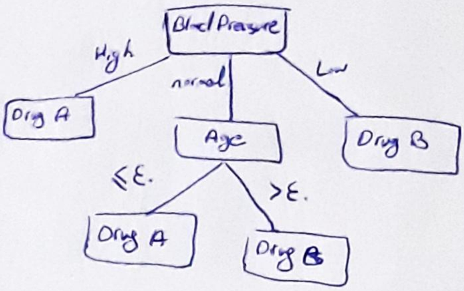
پس $E = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$

$$(\lambda - 2)^2 - 4 = 0 \longrightarrow \lambda = 4, 0 \qquad \begin{vmatrix} \lambda-2 & 2 \\ 2 & \lambda-2 \end{vmatrix} = 0$$

$$Y = X \times W, \quad W = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \longrightarrow Y = \underset{5\times 2}{\begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 3 \\ 4 & 4 \\ 5 & 5 \end{bmatrix}} \times \underset{2\times 1}{\begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}} = \begin{bmatrix} \frac{2}{\sqrt{2}} \\ \frac{4}{\sqrt{2}} \\ \cancel{\frac{6}{\sqrt{2}}} \\ \frac{8}{\sqrt{2}} \\ \frac{10}{\sqrt{2}} \end{bmatrix}$$

۱۴۰۱/۸/۱۰

روش دسته بندی درخت تصمیم:



مدل جهت نمایش:
- طبقه کننده ها DecisionTree
- رگرسیون RegressionTree

- تعدادی گره و یال
- برگ ها دارای برچسب کلاس
- گره های میانی دارای برچسب ویژگی هستند
- روی هر یال، بازه یا مقدار ویژگی گره بالای (پدر) نوشته می شود.

دو مسأله مهم در ساخت درخت:

۱- مناسب‌ترین ویژگی برای هر گره کدام است؟

۲- شرط پایان؟
- تمام نمونه های یک برگ متعلق به یک کلاس باشند
- به حداکثر عمق درخت رسیده باشیم
- تعداد نمونه های گره از یک حداقل تعداد کمتر باشد.

معیارهای انتخاب صفات:

a) information Gain: $IG(A) = Entropy(D) - Entropy_A(D)$

برای ویژگی A مقدار فوق را محاسبه کنیم:

D: داده های آموزشی است

$$Entropy(D) = -\sum_{i=1}^{c} (P_i \times \log_2 P_i)$$

$$Entropy_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Entropy(D_j)$$

C: تعداد برچسب کلاس ها

P_i: احتمال اینکه نمونه متعلق به کلاس i باشد

v: تعداد اعضای دامنه ویژگی A

D_j: قسمتی از داده های اولیه که مقدار ویژگی آن ها v_j است

$|D|$: اندازه داده های D

ID	Age	income	job	Computer
۱	old	M	St	No
۲	middle	H	T	No
۳	old	L	T	No
۴	young	M	T	yes
۵	young	L	T	yes
۶	old	M	S	yes
۷	middle	M	S	yes
۸	young	H	T	No
۹	old	H	S	No
۱۰	middle	H	S	NO

مثال :

$Entropy(D) =$

$$-\frac{۶}{۱۰} \times Log_۲ \frac{۶}{۱۰} - \frac{۴}{۱۰} \times Log_۲ \frac{۴}{۱۰} = ۰,۹۷۰$$

$$Entropy_{Age}(D) = \left(\frac{۴}{۱۰} \times \left(-\left(\frac{۳}{۴} Log \frac{۳}{۴} + \frac{۱}{۴} Log \frac{۱}{۴}\right)\right)\right) + \left(\frac{۳}{۱۰} \times \left(-\left(\frac{۲}{۳} Log \frac{۲}{۳} + \frac{۱}{۳} Log \frac{۱}{۳}\right)\right)\right) + \left(\frac{۳}{۱۰} \times \left(-\left(\frac{۲}{۳} Log \frac{۲}{۳} + \frac{۱}{۳} Log \frac{۱}{۳}\right)\right)\right)$$

$$= ۰/۸۷۵$$

$$Entropy_{income}(D) = ۰/۰۲۴$$

$$Entropy_{job}(D) = ۰/۹۷۰$$

$$\Rightarrow IG(Age) = ۰/۰۹۵$$

$$IG(income) = ۰/۹۹۶$$

$$IG(job) = ۰$$

موضوع پژوهشی : راه حل های کوتاه کردن ارتفاع درخت تصمیم

نکته : $Entropy$ (همه هم کلاس) $= -1 \, Log \, 1 - ۰ \, Log \, ۰ = ۰$

〃 (نصف کلاس ۱ ، نصف کلاس ۲) $= -\frac{1}{۲} Log \frac{1}{۲} - \frac{1}{۲} Log \frac{1}{۲} = 1$

۱۴۰۱/۰۸/۰۳

- معیارهای انتخاب ویژگی در درخت تصمیم (ادامه)

۱. Information Gain ✓

۲. Gini Index

۳. Gain Ratio

۴. Likelihood Ratio (به عنوان تمرین مطالعه شود)

① $$Gini(D) = 1 - \sum_{i=1}^{c} P_i^2$$

C: تعداد کلاس ها

P_i: احتمال تعلق داده ها به کلاس i

$$Gini_A(D) = \frac{|D_1|}{|D|} * Gini(D_1) + \frac{|D_2|}{|D|} * Gini(D_2)$$

اگر ویژگی A داده ها را به دو سطح تقسیم کند

$$Gini(A) = Gini(D) - Gini_A(D)$$

هر ویژگی که مقدار Gini بزرگ تری داشته باشد انتخاب شود

$$Gain\ Ratio_A(D) = \frac{Information\ Gain(A)}{Split\ Info(A)}$$

$$,\quad Split\ Info(A) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} Log \frac{|D_j|}{|D|}$$

در فرمول ①، اگر خالص متعلق به یک کلاس باشند ← به صفر میل می کند. اگر نباشند، یعنی بی نظمی داشته باشیم، از صفر بیشتری شود.

Support Vector Machine (SVM) - ماشین بردار پشتیبان

پایه این روش برای دسته‌بندی دو کلاسه است.

کلاس +

کلاس *

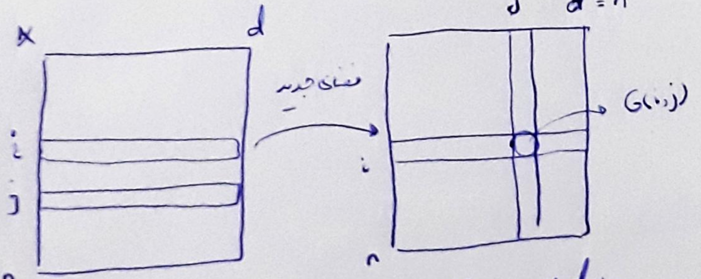
نکته: خطی را انتخاب می‌کنیم که بیشترین حاشیه (Margin) را ایجاد کند.

تمرین: یک مقاله (یادداشت) با توجه به موضوعاتی که تا امروز تدریس شد برای هفته آینده (۱۴۰۱/۸/۸) آماده شود.

موضوع پژوهش: استفاده از SVM برای تمایز Node های Decision Tree

---

نکات مربوط به توابع Kernel :



Grom Matrix

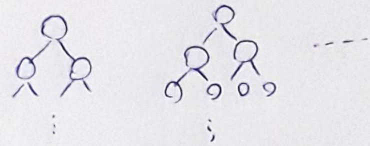
$$G(i,j) = \text{kernel}(\vec{x}_i, \vec{x}_j)$$

Kernel

- Linear $\rightarrow \vec{x}_i \cdot \vec{x}_j = |\vec{x}_i| \times |\vec{x}_j| \times \cos(\theta)$ — شباهت بین نمونه‌ها را می‌دهد
- Polynomial $\rightarrow (x_i^T \cdot x_j + C)^d$ — داده‌ها به صورت توانی نمایش داده می‌شوند
- RBF $\rightarrow e^{-\frac{\|x_i - x_j\|^2}{2\delta^2}}$ — اختلاف و شباهت بین نمونه‌ها را به صورت نمایی نشان می‌دهد

Random forest:

۱۴۰۱/۸/۱۷



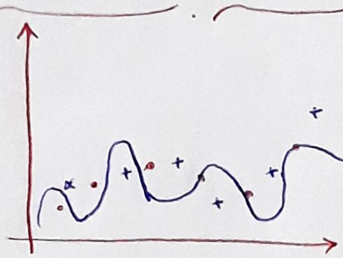
DT1 DT2 ...

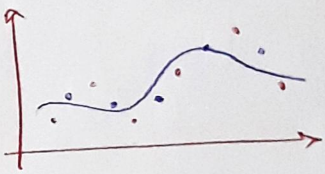
Combination of classifiers:

یکسری دسته‌بند
ابتدا آموزش داده می‌شوند
پس خروجی آن‌ها به نحوی با هم در نظر گرفته می‌شوند
تا داده تست برچسب گذاری شود.

Train 80%
Test 20%

نکته: مسئله Overfitting:

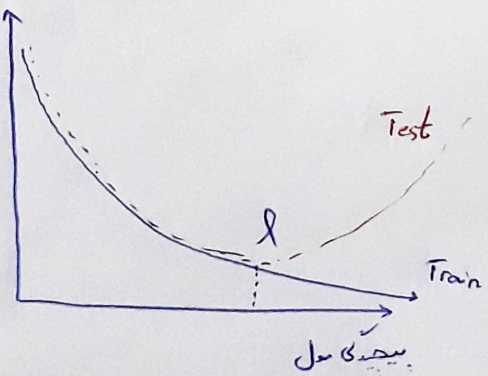




چند جمله‌ای درجه ۱۰ — چند جمله‌ای درجه ۳

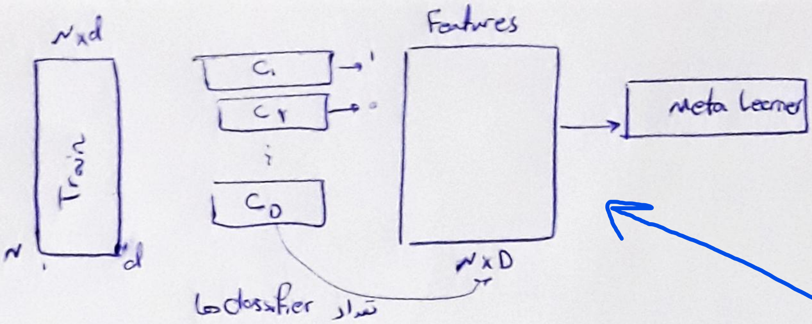
دقت Train: ۱۰۰٪ — دقت Train: 80 درصد

دقت تست < دقت تست



پیچیدگی مدل از یک نقطه به بالاتر موجب می‌شود مدل روی داده‌های Train اصطلاحاً fit شود و قابلیت تعمیم (generalization) روی داده‌های تست کم شود.

برای بالا بردن قابلیت تعمیم درختان، ۲ کار انجام می‌شود؛ اصطلاحاً داده‌ها را در یک سری Bag (کیسه) قرار می‌دهیم. به تعداد درختان این Bag ها را تولید می‌کنیم. ① به صورت تصادفی (مثلاً) ۹۰٪ داده‌های Train را به DT_i نسبت می‌دهیم و این عمل مجدد برای tree های دیگر تکرار شود. ② علاوه بر آن، (مثلاً) ۹۰٪ feature ها را در این bag ها در نظر می‌گیریم و مابقی ۱۰٪ به صورت تصادفی حذف شود.



روش های تصمیم گیری نهایی :

1- رأی گیری
2- ترکیب میانگین وزن دهی شده
3- boosting
4- bagging
5- stacking

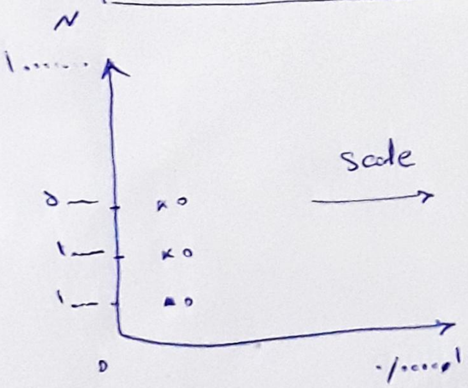
train each model on X% of data
then averages the predictions

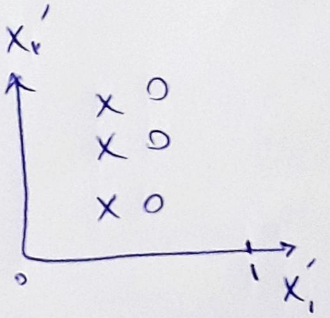
X

شمارش	قند خون	چربی خون	جرم
۱۲	۷۹	۱۰۰	۸۵
۸	۱۸۰	۲۰۰	۹۰
۹	۲۰۰	۱۰۰۰	۶۲
۱۱	۵۰۰	۸۰۰۰	۵۷
⋮	⋮	⋮	⋮

نرمالیزیشن :

با توجه به اینکه بازه عددی مقادیر feature های مختلف در بازه یکسان قرار ندارند، در مسأله دسته بندی در هنگام مدل سازی یا اجرا، کار پیچیده می‌شود و بعضاً ممکن است خطا ایجاد کند.





روش Linear :

$$x' = \frac{x - Min}{Max - Min}$$

روش گاوسی :

$$x' = \frac{x - \mu}{\delta}$$

۱۴۰۱/۰۸/۲۲ ←

۱۴۰۱/۰۸/۲۹ امتحان میان ترم

۱۴۰۱/۹/۶ سئوال انتخاب شده ارائه شود

نکته درباره مبحث SVM ، استفاده از دسته بندهای دو کلاسه برای مسائل چند کلاسه (مانند SVM)

مثلاً اگر بجای کلاس ۱ و ۲ ، کلاس های ۱،۲،...،m داشتیم چه کار کنیم؟ می توانیم به تعداد $\frac{m(m-1)}{2}$ SVM جداگانه برای لیبل های مختلف درست کنیم. نهایتا در فاز تست رأی گیری کنیم. (مثل onetoone)

$SVM_{1,2}$
$SVM_{1,3}$
⋮
$SVM_{1,m}$

مدل ۲) one to other : کلاس اول را یک کلاس و باقی کلاس ها همه یک کلاس

SVM_1 : $\boxed{۲،۳، \ldots m}$ $\boxed{1} \rightarrow ۱$

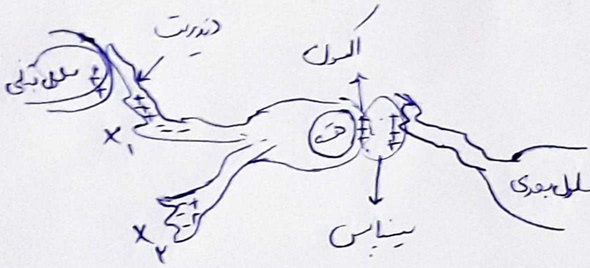
SVM_2 : $\boxed{۱،۳، \ldots m}$ $\boxed{2} \rightarrow ۱$

⋮

SVM_m : $\boxed{۱،۲، \ldots m-1}$ $\boxed{m} \rightarrow ۱$

یک راه این است که از stacking در فاز Train استفاده کنیم و در نهایت سیستم Metalearner پیاده شود. هرچند این روش احتمالاً خوب نیست، چون بایاس را در استفاده از تنها یک دسته بند بالا می بریم.

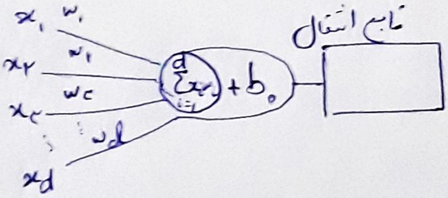
شبکه عصبی :



سلول های عصبی (نورون ها) حسی / حرکتی

مدل محاسباتی نورون :

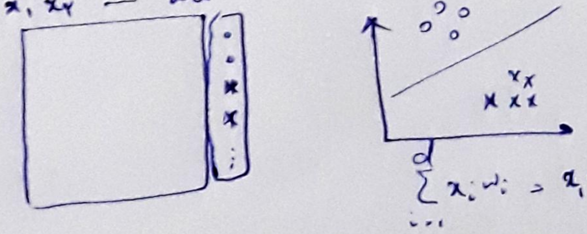
پرسپترون



$$\text{if } \sum_{i=1}^{d} x_i w_i > \underbrace{\theta}_{\text{threshold}} \longrightarrow \text{fire output} = +1 \text{ else output} = 0$$

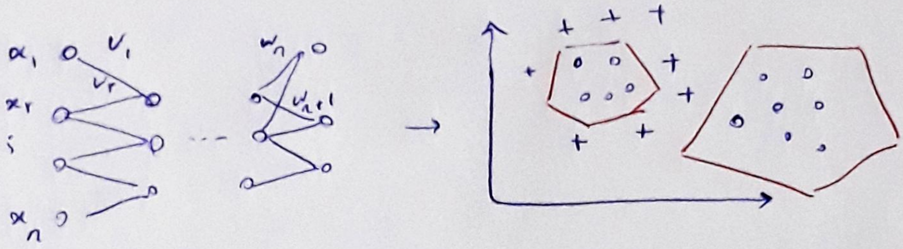
$, \; x_0 = 1$

مثال، در مسأله دسته بندی نیز اگر از پرسپترون استفاده کنیم، گویا یک معادله خط ایجاد کرده ایم :



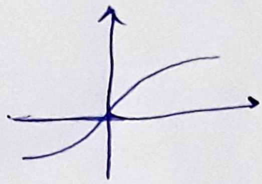
$$\sum_{i=1}^{d} x_i w_i = x_1 w_1 + x_2 w_2 + b_0 > 0$$

نکته: ممکن است نتوانیم به صورت خطی ساده را جدا کنیم در این حالت باید از MLP یا Multi Layer Perceptron استفاده کنیم.



نکته: در تابع انتقال، می‌توانیم به جای اینکه خروجی Σ فقط 0 یا 1 باشد، خود مقدار Σ را گرفته و در تابع دیگری نگاشت کنیم. مثلاً:

$\Sigma \longrightarrow Z$



or