

تعريف هوش: از دیدگاه فلسفه \rightarrow تطبيق با محیط. که به دو شکل خودش را نشان میدهد.

- (1) از طریق تکامل: برفرض یک درخت که براساس تکامل میتواند طوری تغییر کند که درخت های بعدی با یک محیط سرد تطابق داشته باشند. در الگوریتم های تکاملی در این مورد صحبت خواهد شد.
- (2) در لحظه: تنها انسان قابل این کار است. بر فرض با سرد شدن هوا برای خود لباس درست میکند.

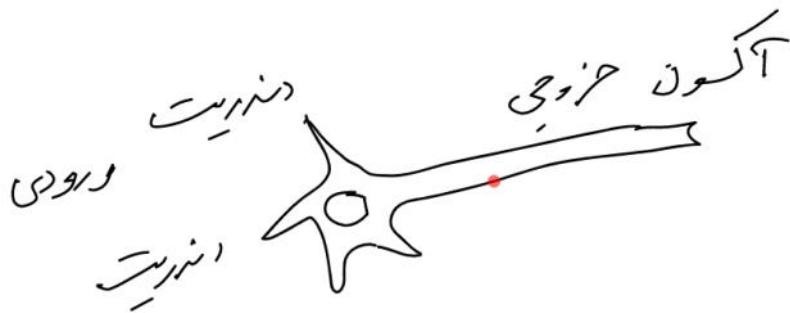
هرچند درخت هوشمند است، اما این هوش بر اساس تطبیق است که در تکامل به وجود آمده است و نسل فعلی نمیتواند از آن استفاده کند بلکه نسل های بعدی میتوانند خود را با محیط تطبیق دهند.

تفاوت هوش محاسباتی با هوش: در هوش محاسباتی بیشتر از ریاضیات استفاده می شود. ریاضیاتی اعم از: جبر، آمار و احتمال، منطق و ...

| شبکه عصبی

الهام گرفته از مغز انسان است.

در مغز انسان به سلول های عصبی نورون (Neuron) گفته میشود. در یک نورون ورودی های تحت عنوان دندریت دارد که معمولاً 3 الی 4 تا می باشند. خروجی های آن نیز آکسون می گویند.



ارتباط بین ورودی و خروجی در یک نورون: فرض کنیم هر ورودی پک ولتاژ داشته باشد. اگر مجموع این ولتاژ ها از یک حد آستانه بیشتر شود باعث میشود آکسون (خروجی) یک شلیک انجام دهد. این خروجی به صورت پالس فرکانس می باشد.



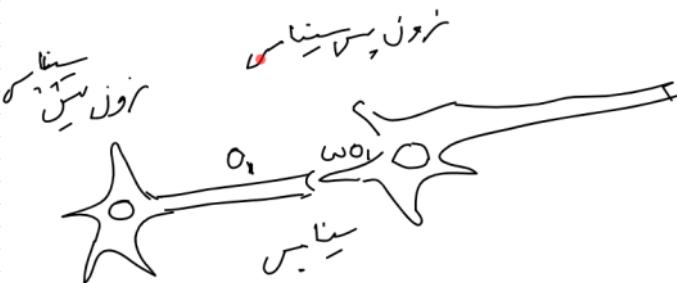
آکسون ها و دندریت ها از طریق یک اتصال شیمیایی به هم وصل می شوند. (اتصال فیزیکی نیست!) به این اتصال شیمیایی سیناپس میگویند.

و ظایف متعدد سیناپس:

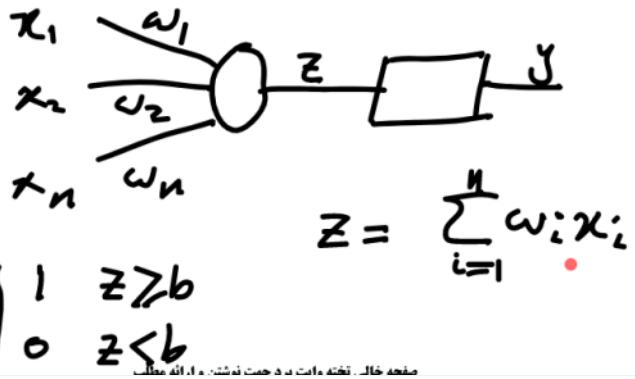
- تبدیل فرکانس به ولتاژ

- تقویت و تضعیف خروجی آکسون. **اگر** خروجی یک نورون O_1 باشد، ورودی نورون بعدی W_01 می باشد.

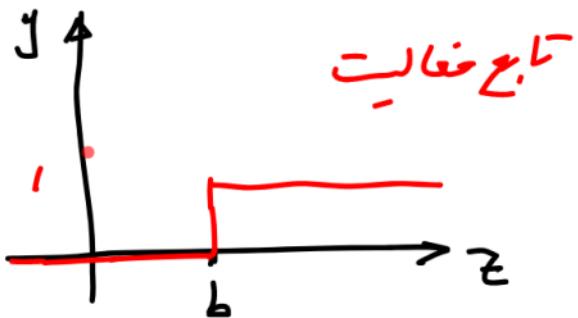
به نورون قبلی نورون پیش سیناپسی و به نورون بعدی پس سیناپسی می گویند.



پرسپترون (Perceptron): یک مدل ریاضی ساده از یک نورون است. معمولاً به شکل یک دایره کشیده می شود که دندریت x_1 تا x_n به آن وصل می شود. سیناپس ها به صورت w_1 تا w_n تعریف می شوند. خروجی را Z می نامیم. خروجی طبق تعریف برابر است با مجموع وزن سیناپس در ورودی دندریت. خروجی نهایی Z نامیده می شود که زمانی فعال میشود که از یک حد آستانه Threshold ای بیشتر شود.



در واقع تابع بین z و y یک تابع پله می باشد. به طوری که تا زمان که از b کوچکتر باشد خروجی ۰ و اگر بزرگر باشد ۱ می باشد. به این تابع پله، **تابع فعالیت (Activation Function)** گفته می شود.



پس به طور کلی داریم:

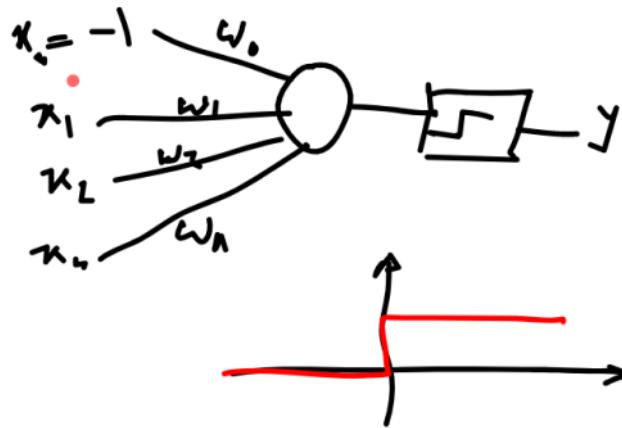
$$y = \begin{cases} 1 & \sum_{i=1}^n w_i x_i + w_0 x_0 \geq b \\ 0 & \text{else} \end{cases}$$

برای بهتر شدن شکل ریاضی، تلاش بر ساده تر کردن فرمول داریم. به طوری که رابطه y را تغییر میدهیم. ابتدا b را ابه سمت چپ نامساوی میبریم که $-b$ میشود. b را به w_0 تغییر میدهیم و سپس یک $x_0 = -1$ به آن اضافه میکنیم. در نهایت رابطه به شکل زیر تغییر میکند:

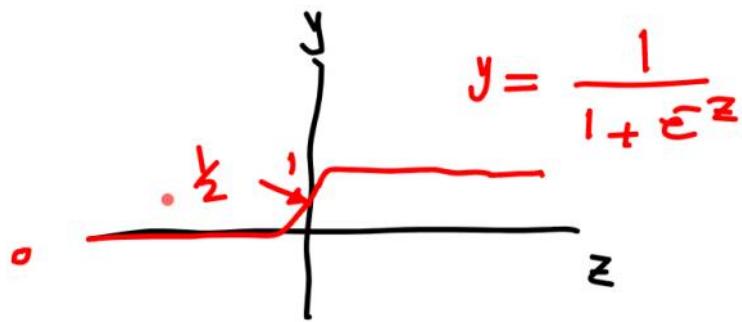
$$y = \begin{cases} 1 & \sum_{i=1}^n w_i x_i - b \geq 0 \\ 0 & < 0 \end{cases}$$

$$y = \begin{cases} 1 & \sum_{i=0}^n w_i x_i \geq 0 \\ 0 & < 0 \end{cases}$$

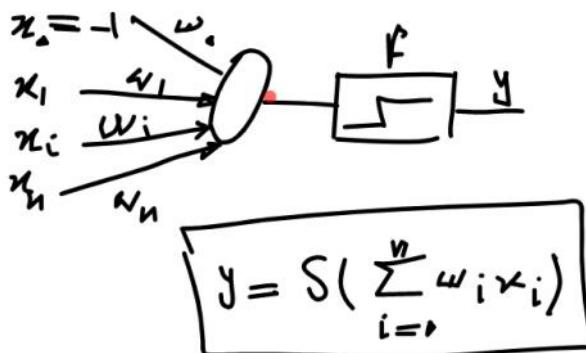
با تغییر رابطه فوق گویی داریم:



مشکل: تابع فعالیت اگر به صورت پله باشد مشتق پذیر نیست. پس برای مشتق پذیر بودن به جای استفاده از تابع پله از تابع سیگموئید (Sigmoid) استفاده می شود.



در جلسه قبل دیدیم که: پرسپترون \leftarrow مدل ریاضی ساده از نرون
مشاهده کردیم که مدل ساده پرسپترون به شکل زیر قابل نمایش است:



که منظور از S تابع Sigmoid می باشد.

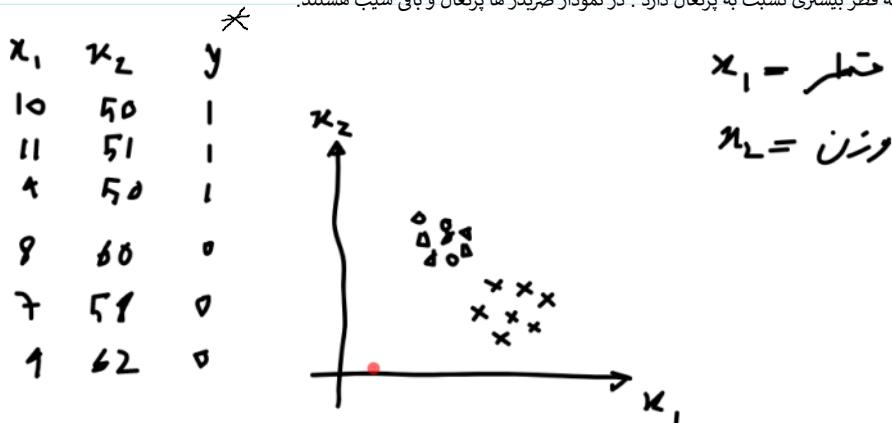
$$y = f\left(\sum_{i=1}^n w_i x_i\right)$$

$$y = S\left(\sum_{i=0}^n w_i x_i\right)$$

تابع فعالیت لزوماً سیگموئید نیست و بعداً خواهیم دید که برفرض در شبکه های عمیق از تابع فعالیت رنو استفاده خواهد شد.

حال میخواهیم با یک مثال بررسی کنیم که پرسپترون چگونه در حل مسائل به ما کمک میکند. برای مثال فرض میکنیم سبب و پرتغال داریم که قطر و وزن آن ها اندازه گیری شده اند.

اگر $y=1$ را پرتغال و $y=0$ را سبب در نظر بگیریم، میدانیم که سبب نسبت وزن به قطر بیشتری نسبت به پرتغال دارد. در نمودار ضرایرها پرتغال و باقی سبب هستند.



حال اگر بخواهیم خطی بکشیم که سبب و پرتغال را جدا کند، خط به صورت زیر می باشد:



حال اگر بخواهیم معادل این خط را بدست بیاوریم داریم:

$$-w_0 + w_1x_1 + w_2x_2 = 0$$

حال اگر بخواهیم وزن های این معادله خط را بدست بیاوریم. میدانیم در جایی که $x_1=1$ می باشد $x_2=0$ می باشد همچنین فرض میکنیم نقطه دیگر $x_1=10$ و $x_2=60$ می باشد. در این صورت داریم:

$$\begin{aligned} -w_0 + w_1x_1 + w_2x_2 &= 0 \rightarrow -4 + w_1 = 1 \rightarrow w_1 = 5, \\ -w_0 + 10w_1 + 60w_2 &= 0 \quad 10w_1 = -60w_2 \end{aligned}$$

در این حالت $w_2 = -3/20w_0$ خواهد شد و داریم:

$$\begin{aligned} -1 + x_1 - \frac{3}{20}w_0 &= 0 \\ -w_0 + 10w_1 - \frac{3}{20}w_0 &= 0 \end{aligned}$$

با توجه به رابطه های فوق در میابیم که سمت چپ خط رسم شده $0 < Z$ و سمت راست آن $0 > Z$ می باشد. (برای صحبت این موضوع میتوانیم نقطه $(0, 0)$ را در معادله تست کنیم)

اگر بخواهیم این خط را به صورت ریاضی بدست آوریم، بایستی چه کار کنیم؟ میتوانیم از داده های آموزشی کمک بگیریم. تمامی x_1, x_2 و y^* های داده شده داده آموزشی می باشند.

$$-w_0 + w_1x_1 + w_2x_2 = y^*$$

$$-w_0 + w_1 \times 10 + w_2 \times 60 = 1$$

⋮

$$-w_0 + w_1x_1 + w_2x_2 = 0$$

سپس چند معادله فوق را میتوانیم به صورت ماتریس بنویسیم. به طوری که:

$$\begin{matrix} X & W & \hat{Y} \\ \begin{pmatrix} -1 & 10 & 5 \\ -1 & 11 & 6 \\ -1 & 9 & 5 \\ -1 & 8 & 4 \\ -1 & 7 & 3 \\ -1 & 1 & 2 \end{pmatrix} & \begin{pmatrix} w_0 \\ w_1 \\ w_2 \end{pmatrix} & \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \end{matrix}$$

مجموع مربعات خطای برابر است با:

$$\sum_{i=1}^n (\sum_{j=1}^m w_j x_{ji} - y_i^*)^2 = \text{مجموع مربعات خطای خالی نخست و ایت برد جهت نوشتن و ارائه مطلب}$$

$$(XW - \hat{Y})^T (XW - \hat{Y})$$

استاد توان 2 نگذاشت ولی بایستی در رابطه باشد. هم با باز کردن ضرب داخلی 2 ماتریس و هم با این منطق، که ممکن است سیگما $\sum w_j x_{ji}$ از y^* کمتر شود و جاهای بیشتر و منق مثبت ها همیگر را خنثی کنند و نتیجه صفر شود به این موضوع میتوان لی برد.

$$E = (XW - \hat{Y})^T (XW - \hat{Y})$$

$$\frac{dE}{dw} = 2x^T(xw - y) = 0$$

| w₂ |

$$x^T x w - x^T y = 0$$

$$x^T x w = x^T y \Rightarrow w = (x^T x)^{-1} x^T y$$

صفحه خالی، تغفه، امت و د حمت نوشته، از الله مطلب

نحوه مشتق گیری ماتریسی:

$$y = x^T x$$

$$\frac{dy}{dx} = 2x$$

$$y = (Ax - B)^T (Ax - B)$$

$$\frac{dy}{dx} = 2A^T (Ax - B)$$

برای اثبات پایستی به فرم غیر برداری بنویسیم. سپس درایه به درایه مشتق گیری کنیم و سپس به حالت بردار در بیارویم و مشاهده میکنیم که چنین روابطی حاصل میگردند.

برای این که مطمئن شویم کجا پایستی تر نهاده گذاشت، پایستی اندازه ها با هم بخواند. برفرض مثال داریم:

$$y = (Ax - B)^T (Ax - B)$$

$1 \times L$ $L \times n$ $n \times 1$

$$\frac{dy}{dx} = 2A^T (Ax - B)$$

$\frac{dy}{dx}$ $m \times L$ $L \times 1$ *

پس در نهایت w برابر است با :

$$E = (\mathbf{X} \mathbf{w} - \mathbf{y}^*)^\top (\mathbf{X} \mathbf{w} - \mathbf{y}^*)$$

$$\mathbf{w} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}^*$$

راه دیگر محاسبه نیز به روش زیر میباشد که در درس جبر خطی نیز مشاهده کرده اید:

$$\mathbf{X} \mathbf{w} = \mathbf{y}^*$$

$$(2) \quad \mathbf{X}^\top \mathbf{X} \mathbf{w} = \mathbf{X}^\top \mathbf{y}^*$$

$$(3) \quad \mathbf{w} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}^*$$

طبق تئوری 14 فصل 6 کتاب جبرخطی و کاربرد های آن (David C Lay)، زمان میتوان از (2) به (3) رسید که ستون های ماتریس \mathbf{X} مستقل خطی باشند. یعنی اگر بر فرض مثالی داشته باشیم که در سیب ها و پرتقال ها $x_2 = 2x_1$ باشند، آنگاه دیگر ستون های ماتریس \mathbf{X} مستقل خطی نخواهد بود و $\mathbf{X}^\top \mathbf{X}$ معکوس پذیر نیست. پس برای حل (2) در این موارد خاص بایستی از روش Row Reduction Algorithm کمک گرفت.

- توجه شود که روش توضیح شده برای حل مسائل Regression می باشد و خط مورد نظر ما را در مسئله Clustering تولید نخواهد کرد! در این مورد در جلسه بعد بیشتر صحبت خواهد شد

چه ویژگی هایی برای حل یک موضوع مناسب تر هستند؟ 1) مستقل خطی باشند. 2) مرتبط به موضوع باشند.

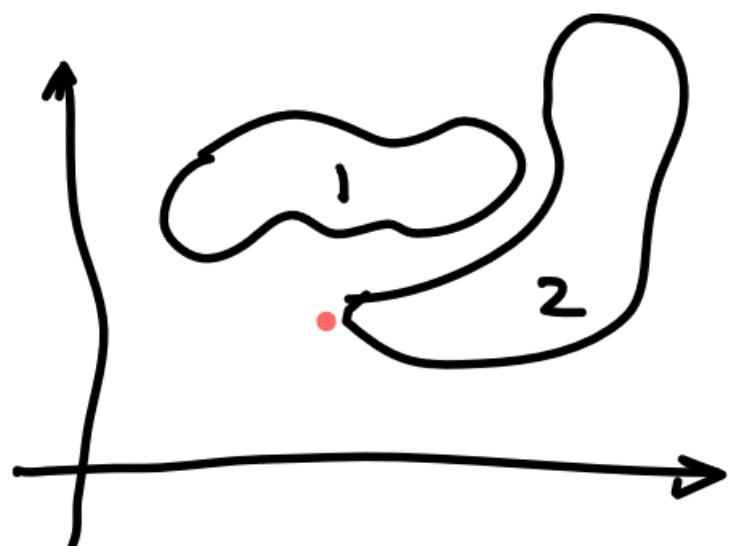
در مساله که حل کردیم 2 بعدی بود و خط جدا کننده است (با یک پرسپترون منحنی نمیتوان ساخت). در 3 بعدی صفحه و از 4 بعد به بالا ابر صفحه داریم.

انواع یادگیری:

- یادگیری با نظارت (Supervised): یادگیری از روی نمونه هایی که label دارند
- یادگیری بدون نظارت (Unsupervised): بر اساس شباهت. مثل خوش بندی (Clustering) که داده های شبیه هم در یک گروه قرار میگیرند.
- یادگیری تقویتی (Reinforcement) : یادگیری بر اساس تجربه. مثلا از خانه به دانشگاه میرویم یک مسیری انتخاب میکنیم میبینیم طولانی شد. سپس یک مسیر دیگر میرویم و در آخر بر اساس تجربه مسیری با کوتاه ترین زمان به دانشگاه را انتخاب میکنیم (صحیح و خطأ) البته کمی با صحیح و خطأ از لحاظ رسیدن به جواب متفاوت است. از این جهت که بر فرض تا زمانی که به دانشگاه رسیده ایم نمیدانیم که چهار خطای شده ایم یا نه ولی در در آزمون و خطای بالا فاصله متوجه خطای شدن خواهیم شد.

شبکه های عصبی تمامی 3 یادگیری فوق را میتوانند انجام دهند. در این درس تمرکز بر شبکه های عصبی برای یادگیری با نظارت را داریم و روی باقی تمرکز چندانی نداریم.

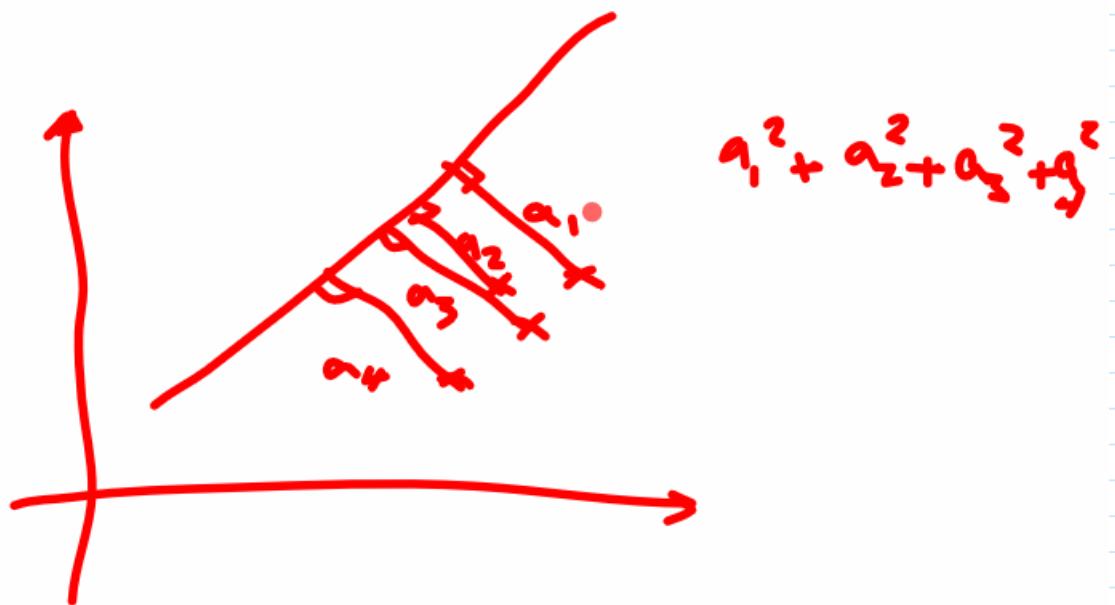
فرض کنید کلاسی داریم که به شکل خطی جدا نمی شوند:



برای جداسازی این کلاس ها با یک پرسپترون نمیتوان جداسازی کرد.

تفاوت رگرسیون و دسته بندی: در رگرسیون خروجی یک منحنی پیوسته است. ولی در دسته بندی خروجی گسته است.
برای مثال از رگرسیون میتوانیم زمان با سرعت ماشین در زمان را در نظر بگیریم و میخواهیم سرعت ماشین در آینده را پیشبینی کنیم.
برای دسته بندی نیز بر فرض میتوانیم یک سری داده از آدم ها در نظر بگیریم که یا بیماری قلبی دارند یا خیر. سپس میخواهیم پیشbینی کنیم با وجود feature های یک آدم جدید آیا این فرد دارای بیماری قلبی می باشد یا خیر.
مثال جلسه قبل یک مساله دسته بندی بود که با least squares error خواستیم دقیق بودن آن را مورد نظر قرار دهیم.
با پرسپترون هم میتوانیم هم regression هم classification را انجام دهیم.

:regression در least squares error



تلاش بر این است که حاصل جمع نوشته شده را مینیمم کند.

کراس آنتروپی: اگر γ را خروجی شبکه و خروجی خواسته شده γ^* باشد، هدف مینیمم کردن E می باشد که داریم:

$$\min E = \sum_{l=1}^L (\gamma^l - \hat{\gamma}^l)^2 = (\gamma - \hat{\gamma})^\top (\gamma - \hat{\gamma})$$

γ خروجی
 $\hat{\gamma}$ پرستپترون

اگر خروجی پرستپترون را احتمال کلاس بگیریم. $L^* p$ یعنی کلاس داده L ام فرض پرتعال باشد (پرتعال 1 و غیر آن 0) و داریم:

• O^P

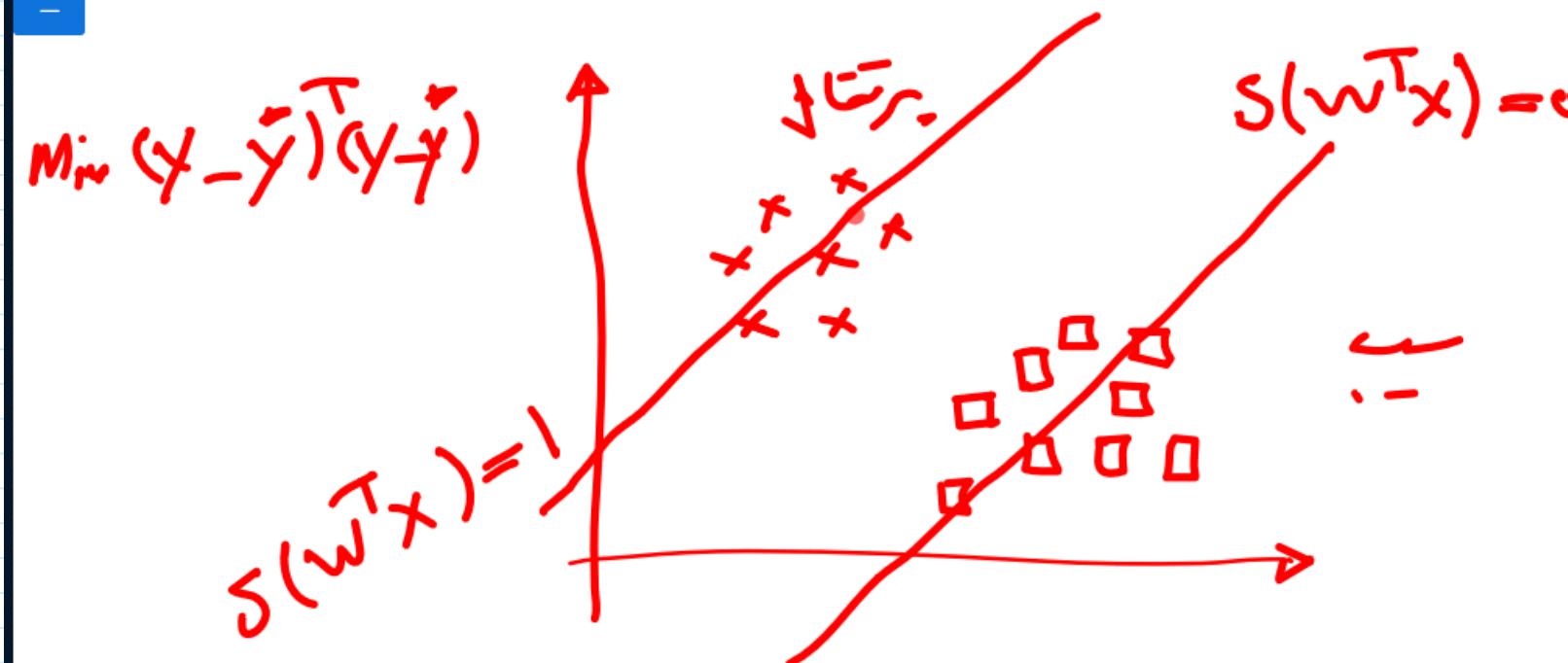
$$-\sum_{l=1}^L (\hat{P}_l \log P_l + (1 - \hat{P}_l) \log (1 - P_l))$$

میتوان اثبات کرد وقتی کراس آنتروپی مینیمم میشود که در مثال ما کلاس های پرتغال هستند احتمال 1 و برای کلاس هایی که پرتغال نیستند احتمال صفر شود

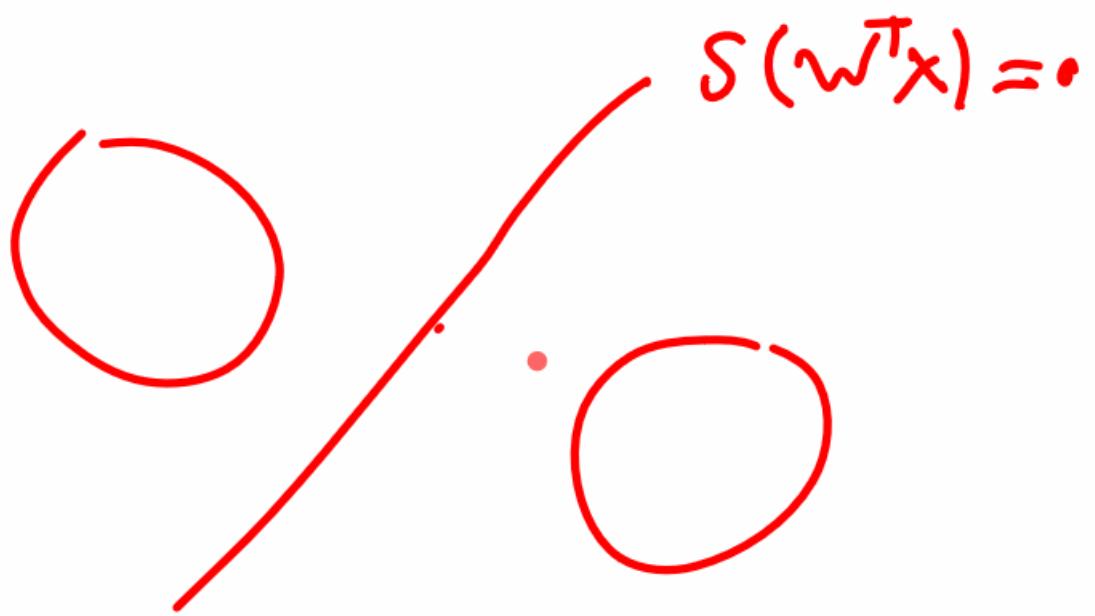
$$E = - \sum_{l=1}^L (\hat{P}_l \log P_l + (1 - \hat{P}_l) \log (1 - P_l))$$

$$\hat{P}_l = \sigma \left(\sum_{i=0}^n \omega_i x_i \right)$$

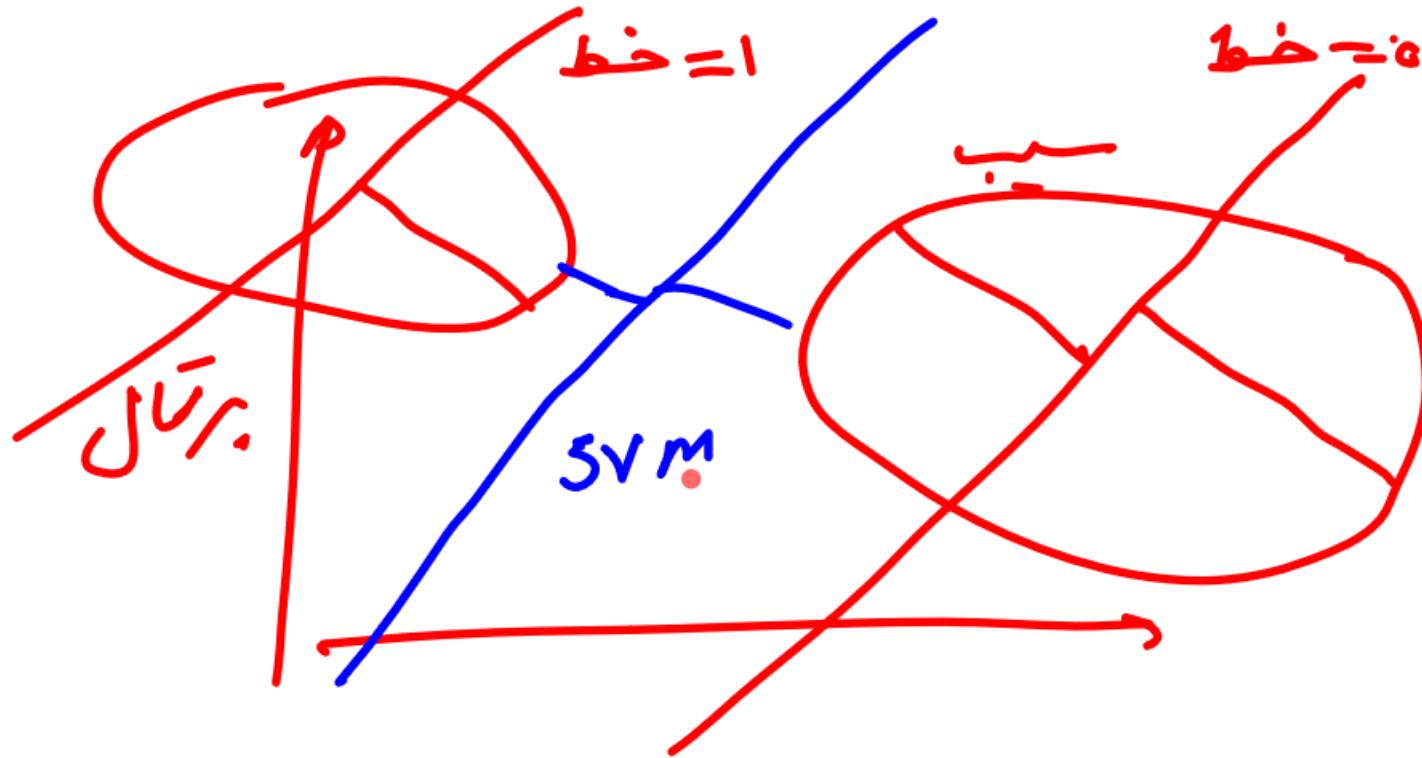
در حل به کمک (Mean Squared Error) MSE صحیح میکنیم مربع خطا را کم کنیم (همون که ریاضیش جلسه پیش بود، اون روش برای classification جواب نیست):



اما در کراس آنتروپی میخواهیم خطی بدست آوریم که فاصله از داده های هر دو کلاس به حداقل برسد (برای مساله دسته بندی):

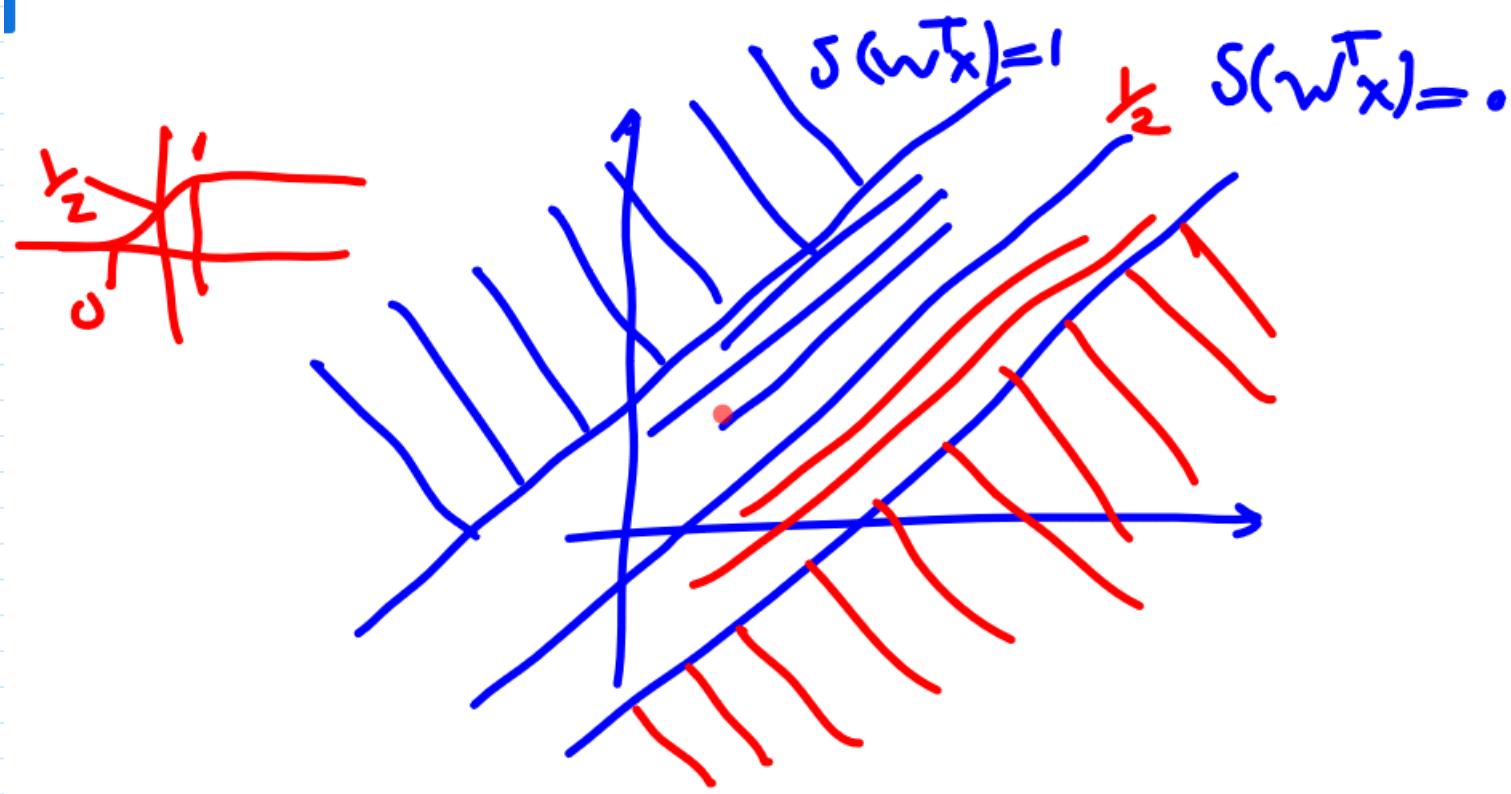


روش دیگر نیز SVM می باشد که خطی بدهست می آید که فاصله تا هر دو داده مаксیمم شود. یعنی اگر خط قرمز MSE و آبی SVM باشد داریم:



کراس آنتروپی نه به خوبی SVM می باشد و نه به بدی MSE (دقیق تر است و در یادگیری ماشین تدریس می شود)
پس برای Regression از MSE و از SVM و کراس آنتروپی برای classification استفاده میکنیم.

سیگموید فضا را به 4 قسمت تقسیم میکند:

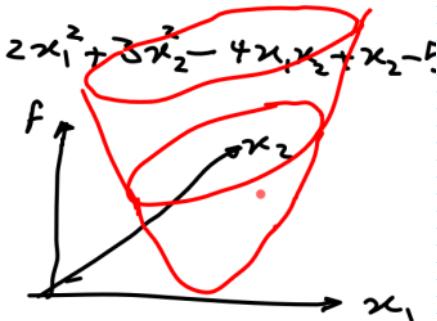


- (1) بخش بزرگتر از 1
- (2) بخش کوچکتر از 0
- (3) بخش بین 1/2 و 1
- (4) بخش بین 0 و 1/2

میخواهیم به کمک گرادیان تابع هدف کراس آنتروپی را مینیمم کنیم. لذا یک مختصه توپیخی در مورد گرادیان خواهیم داشت تا بتوانیم وزن های پرسپترون را برای تابع هدف کراس آنتروپی بدست آوریم.

$$\text{Hypothesis: } F(x_1, x_2) = 2x_1^2 + 3x_2^2 - 4x_1x_2 + x_2 - 5$$

اگر بخواهیم روی مختصات رسم کنیم به صورت فرضی یک شکل همانند شکل زیر خواهد داشت:



میخواهیم بردار گرادیان را بدست آوریم. علامت گرادیان را با یک مثلث بر عکس نشان میدهیم و داریم:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix}$$

در نقطه $x_1 = 2$ و $x_2 = 3$ میخواهیم بردار گرادیان را مشخص کنیم. داریم:

$$\frac{\partial f}{\partial x_1} = 4x_1 - 4x_2 \quad (= -4)$$

نحوه وابستگی کوئن و از این طبق

لذا درایه اول بردار گرادیان در نقطه مورد نظر 4- می باشد. برای محاسبه درایه دوم نیز داریم:

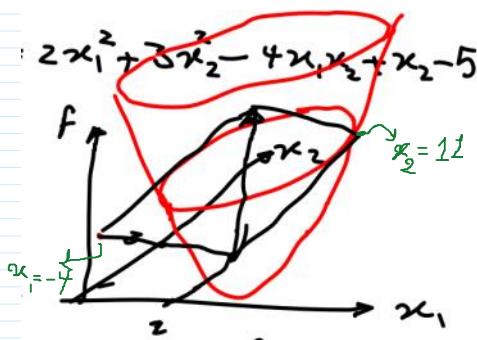
$$\frac{\partial f}{\partial x_2} = 6x_2 - 4x_1 + 1 \quad (= 11)$$

پس بردار گرادیان در نقطه خواسته شده برابر است با:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} = \begin{bmatrix} -4 \\ 11 \end{bmatrix}$$

$x_1 = 2, x_2 = 3$

اگر بخواهیم بردار بدست آورده را در نقطه مورد نظر رسم کنیم، در چهت x_1 منهای 4 تا باید جلو برویم و در چهت x_2 نیز 11 تا جلو برویم و داریم:



این بردار در صفحه x_1-x_2 است و در جهی می باشد که بیشترین مقدار افزایش تابع را در آن جهت داشته باشیم. یعنی اگر بخواهیم ماسکیم تابع را در کمترین مسیر بدست بیاوریم در جهت گرادیان حرکت میکنیم و اگر بخواهیم در کمترین مسیر به مینیمم تابع برسیم عکس جهت گرادیان حرکت میکنیم.

پس \mathbf{x}^{new} برابر است با: (گاهی اوقات old را با t و new را با $t+1$ نمایش می‌دهیم)

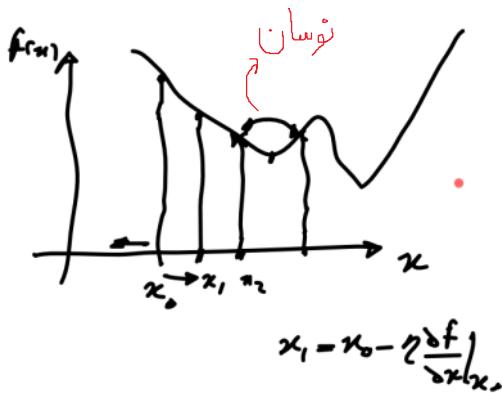
$$X^{new} = \begin{bmatrix} x_1^{new} \\ x_2^{new} \end{bmatrix} = \begin{bmatrix} x_1^{old} \\ x_2^{old} \end{bmatrix} - \gamma \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix}$$

$$X^{t+1} = X^t - \gamma \nabla f(X^t)$$

یک ضریب اتا نیز قبل گرادیان فی آید و معنی اندازه قدم را دارد (البته در صورتی که بردار گرادیان را نرماب کنیم). یعنی مشخص میکند چه میزان خلاف جهت گرادیان باید حرکت کنیم تا به مینیمم برسیم.

مثال فرض کنیمتابع زیر را داریم و در نقطه x_0 گرادیان را بدست آورده ایم. سپس به سمت مینیمم حرکت میکنیم و به x_1 میرسیم. این روند ادامه می باشد و مدت حرکتش وابسته به این است که گرادیان چه زمانی صفری شود. اندازه اتا اگر بزرگ باشد ممکن است بین 2 نقطه نوسان داشته باشیم (همون منحنی نیم دایره که کشیده) و هیچ موقع به مینیمم نرسیم.

همجین یک مشکل گرادیان هم این است که ممکن است در مینیمم محلی باشد.



حال اتا راجه اندازه بگیریم؟! ابتدا اتا را به اندازه متوسطی در نظر میگیریم. اگر $\Delta f = f_{new} - f_{old}$ منفی بود. آنگاه یعنی خوب داریم پیش میریم پس اتا را بزرگ میکنیم. مثلا در 1.1 ضرب میکنیم. ولی اگر این Δf مثبت شود. یعنی به مینیمم نزدیک نمیشیم و بلکه داریم دورز میشیم. پس باید اتا را کوچکتر میکنیم یعنی مثلا در 0.9 ضرب میکنیم.

برای این که اتا اندازه قدم باشد پایستی گرادیان در اندازه اش تقسیم شود تا بردار اندازه 1 داشته باشد:

$$X^{t+1} = X^t - \frac{\gamma}{\|\nabla f(x^t)\|} \nabla f(x^t)$$

یک روش دیگر برای بدست آوردن مینیمم در توابع ریاضی این است که مشتق بگیریم و برابر با صفر قرار دهیم. با توجه به این که گرادیان ممکن است در مینیمم محلی گیر کنند پس چرا از مشتق مساوی صفر استفاده نمیکنیم؟!

علم: باید زند تک x_i ها را برابر با صفر قرار بدهیم. یک دستگاه را حل کنیم و سپس تمامی x_i های مختلف را امتحان کنیم بینینم که این است که حل دستگاه با ابعاد زیاد ممکن است راه حل بسته ای برای آن موجود نباشد.

آنتروپی: معنی پی نظری است. مثلا وقتی بگوییم با داده های با احتمال 1 سیب و با احتمال 0 برتغال و داده دیگر پالعکس \rightarrow قطعیت کامل داریم و پی نظری = 0 است. اگر اکر با احتمال 0.5 سیب و احتمال 0.5 برتغال باشد پی نظری بیشتر است.

کراس آنتروپی تا حد امکان میخواهد این آنتروپی را کم کند. یعنی تا حد امکان یک خط جدا کننده پیدا کنند که برفرض بگویید یک داده ای با احتمال 1 سیب و 0 برتغال است.

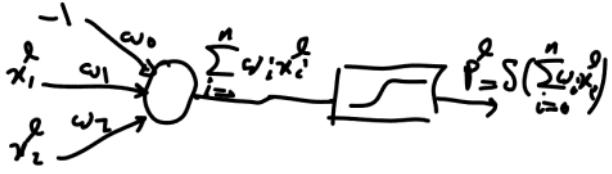
$$E = - \sum_{l=1}^L (P_l \log P_l + (1-P_l) \log (1-P_l))$$

$$E = - \sum_{l=1}^L (P_l \log P_l + (1-P_l) \log (1-P_l))$$

تفاوت کراس آنتروپی با آنتری در * است. در واقع * مقدار های واقعی است.

$$P_l = S(\sum_{i=0}^n w_i x_i)$$

پادآوری:



$$\omega_i^{t+1} = \omega_i^t - \eta \frac{\partial E}{\partial \omega_i} \quad \text{Cross entropy}$$

برای محاسبه رند کراس آنتروپی نیز داریم:

$$\frac{\partial E}{\partial \omega_i} = - \sum_{l=1}^L \left(\hat{P}_l \frac{\partial P_l}{\partial \omega_i} / \bar{P}_l \right) \left(\frac{1 - \hat{P}_l}{1 - P_l} \right) \frac{\partial P_l}{\partial \omega_i}$$

$$= \sum_{l=1}^L \left(\frac{1 - \hat{P}_l}{1 - P_l} - \frac{\hat{P}_l}{P_l} \right) \frac{\partial P_l}{\partial \omega_i}.$$

$$P_l = S \left(\sum_{i=0}^n \omega_i x_i \right)$$

$$\hat{P}_l = S(z) = \frac{1}{1 + e^{-z}}$$

$$S'(z) = S(z)(1 - S(z))$$

$$\frac{\partial P_l}{\partial \omega_i} = \frac{\partial P_l}{\partial z} \times \frac{\partial z}{\partial \omega_i} =$$

علت مشتق :

$$\frac{1}{1 + e^{-z}} \left(1 - \frac{1}{1 + e^{-z}} \right) = \frac{e^{-z}}{(1 + e^{-z})^2}$$

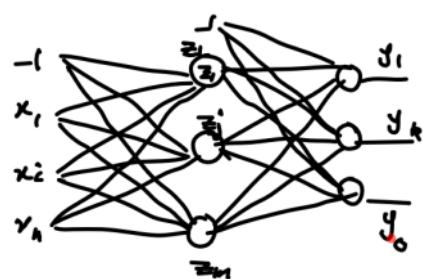
$$S'(z) \quad \frac{1 + e^{-z}}{1 + e^{-z}}$$

$$\omega_i^{t+1} = \omega_i^t - \eta \sum_{l=1}^L \left(\frac{1 - \hat{P}_l}{1 - P_l} - \frac{\hat{P}_l}{P_l} \right) \left(\sum_{i=0}^n \omega_i x_i \right)^2$$

پس ما میتوانیم به این شکل وزن ها را از یک w تصادف انتخاب کنیم و بعد با این رابطه w_l بعدی را بدست بیاوریم تا وقتی مینیمم نشده ادامه دهیم تا به جواب برسیم.

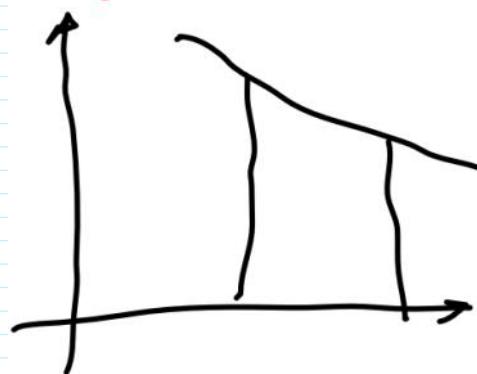
اگر نیاز به تشکیل منحنی برای جدا سازی داشته باشیم باید به سراغ MLP برویم. در شبکه ای از چند لایه پرسپترون که اگر x_1 تا x_n ورودی هایمان باشد، طبق ساده سازی قبلی $-1 = x_0$ نیز اضافه میکنیم. به لایه های وسطی لایه های بنهان میگویند.

MLP نایسی حداقل 3 برسپرون داشته باشد، این که مقدار دقیق آن چه میزان می باشد میتوان با روش های نظری صحیح و خطای حاصل کرد. به روش ریاضی اثبات شده است که یک شبکه عصبی برسپرون که تابع فعالیت غیرخطی داشته باشد، اگر حداقل 3 لایه داشته باشد هر ابرمنحنی غیرخطی را میتواند تولید کند. به لایه آخر لایه خروجی می گویند. با پاسی حداقل 3 لایه داشته باشد.

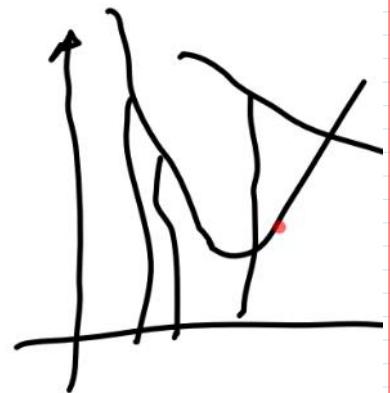


MLP

در مورد MLP جلسه بعدی بیشتر صحبت خواهیم کرد.

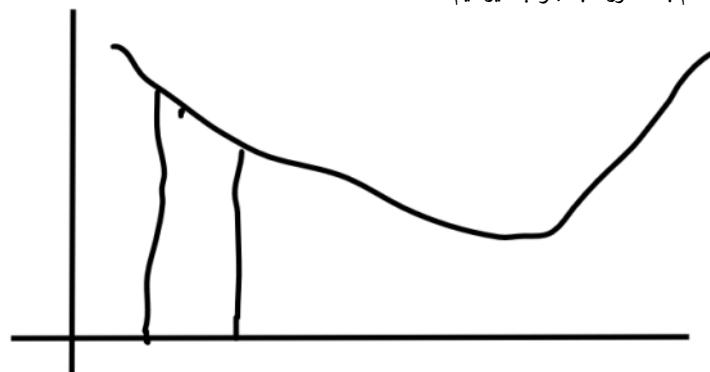


در گرادیان وقتی شبکه باید قدم ها بزرگ باشه تا سریع تر به جواب برسیم



وقتی هم شبکه اینطوری زیاده باید اندازه قدم کم باشه که نوسان نشه.

آیا شبکه لزوماً به جواب میرسیم؟ نه



در این مثال در 2 نقطه مشخص شده شبکه کم است ولی با جواب فاصله داریم.

در جلسه قبل دیدیم یک راه به روش زیر است:

$$\text{(1)} \quad w^{t+1} = w^t - \gamma \frac{\nabla f(w^t)}{\|\nabla f(w^t)\|}$$

یک راه دیگر این است که علاوه بر نرمال کردن، از نرم 2 استفاده کنیم یعنی:

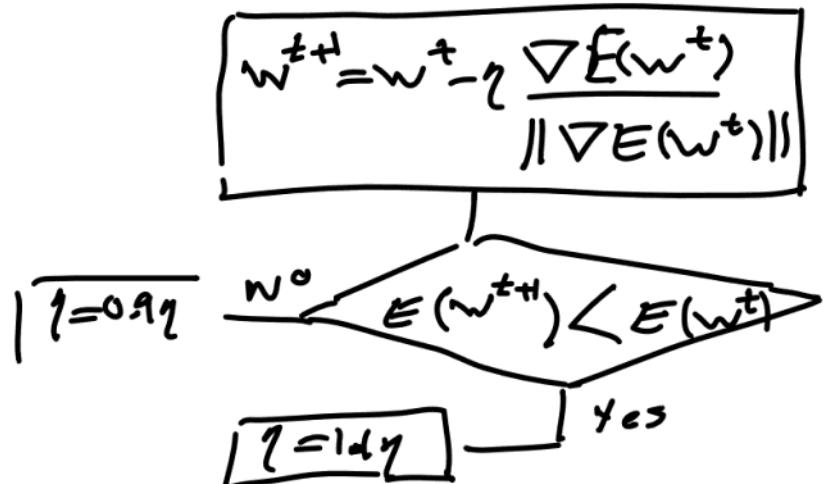
$$(2) \quad w^{t+1} = w^t - \gamma \frac{\nabla f(w^t)}{\|\nabla f(w^t)\|^2}$$

اینطوری باعث میشه جایی که شیب زیاده اندازه قدم کم بشه جایی که شیب کمه اندازه قدم زیاد بشه.

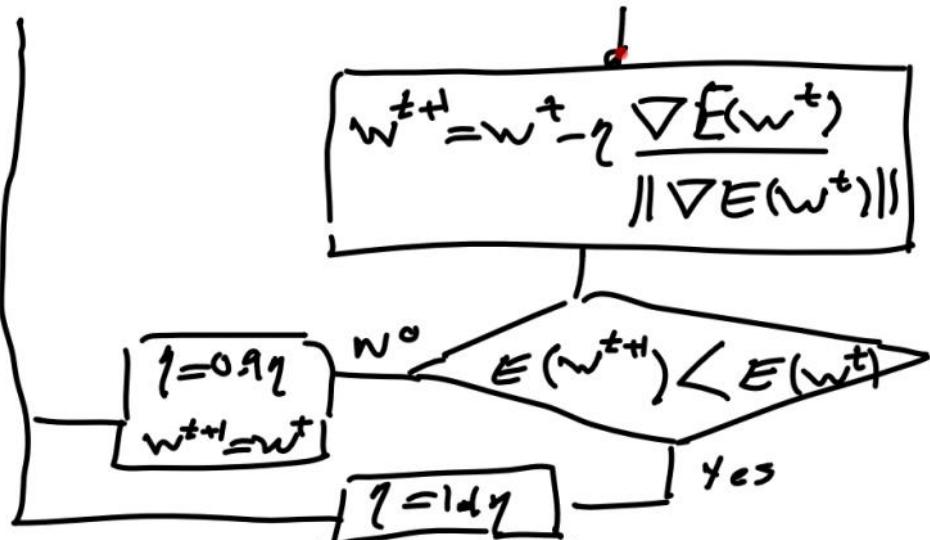
* وقتی به جواب رسیدیم که شیب خیلی کمه اون موقع باعث نمیشه که پریم اونور از جواب خارج شیم؟ نه! اون موقع خودگردایان خیلی کم میشه و اون کسر به 0 میل میکنه

* استاد پیشنهاد کرد که از فرمول (1) استفاده بشه که اتا 0.1 یا 0.01 در ابتدا قرار بگیر.

* همانطور که در جلسه پیش دیدیم، میتوانیم با فلوچارت زیر نیز ببینیم که اتا چگونه تغییر کند:



میشه وقتی اوضاع خراب شد w جدید را برابر w قدیم گذاشت، ولی خب میشه هم نداشت چون با کم کردن اتا دوباره برمیگردد به وضعیت قبلی. در نهایت داریم:

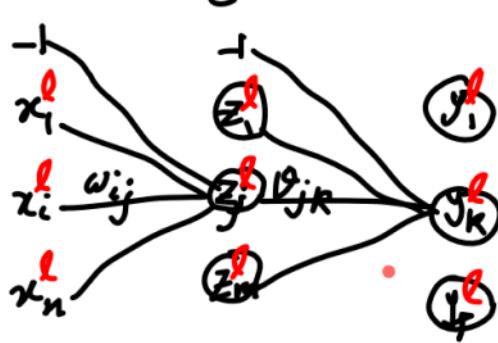


| شبکه چند لایه ای پرسپترون (MLP)

برای این که شلوغ نشه استاد فقط از هر لایه فقط یک نرون رو رسم کرد شبکه اش رو (روابط برای داده ای باشد):

شبکه چند لایه پرسترن

MLP در درجه مختنی



$$z_j^l = \sum_{i=0}^n w_{j,i}^l x_i + b_j$$

$$y_k^l = \sum_{j=0}^m v_{j,k}^l z_j^l + b_k$$

- رگرسیون: از SSE یا MSE استفاده می‌شود.

$$E_{SSE} = \frac{1}{2} \sum_{l=1}^L \sum_{k=1}^p (y_k^l - \hat{y}_k^l)^2$$

- دسته بندی: از کراس آنتروپی و SVM. مثال از کراس آنتروپی:

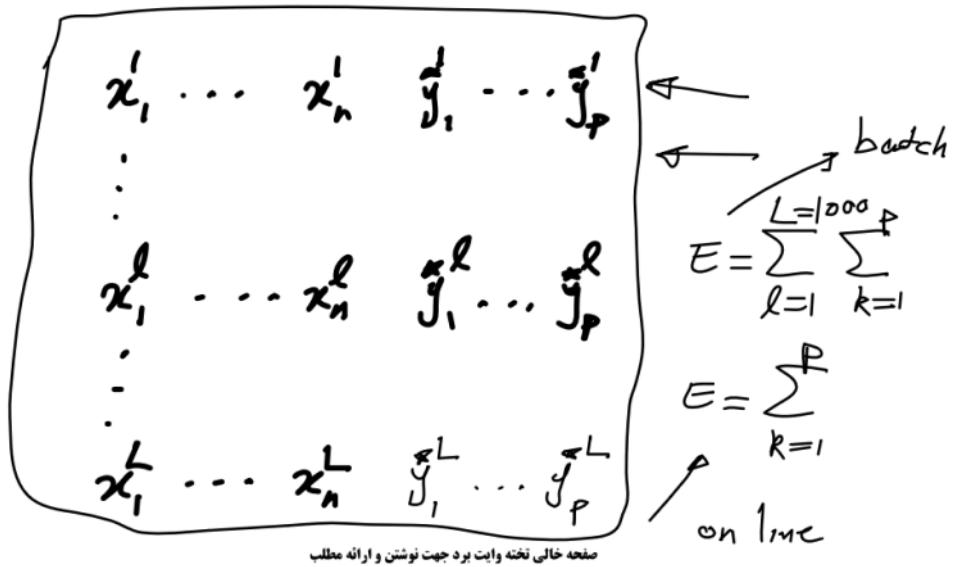
$$E_{CE} = - \sum_{l=1}^L \sum_{k=1}^p (\hat{y}_k^l \log y_k^l + (1 - \hat{y}_k^l) \log (1 - \hat{y}_k^l))$$

- * اگر داده‌ها را یک دفعه با هم یادگیری انجام دهیم و اگر هر داده جدید که می‌داد همون لحظه جداگونه یادگیری بشه Online learning می‌گویند. در batch learning وققی هر batch که یاد می‌دهیم می‌شود یک epoch. این epoch در کل میتواند یک mini batch باشد یا با کل داده‌ها.
- * اگر در فرمول E_{SSE} ما سیگنال اول رو حذف کنیم می‌شود Online learning. چون هر داده جدا خطای گیری انجام می‌شود. ولی وقتی E_{CE} هست همگی با هم هستند.
- * تفاوت **batch** و **online learning**: در **batch** هر داده جهت رو عرض می‌کند و نوسان میتواند بکند. ولی در **online learning** متوسط جهت ها گرفته می‌شود و لذا نتیجه بهتری دارد.
- * امروزه از mini batch learning استفاده می‌کنیم زیرا باعث می‌شود که از مینیموم های محلی فرار کنیم.
- * در این درس فرض می‌کنیم کلا همه داده‌های ما هم آموزش میدیم و کاری با **batch** نداریم.

فرض شود 1000 تا یک میلیون batch داده داریم (بسته به اندازه mini batch یا micro batch می‌گویند). همانطور که از جلسه قبل دیدیم خروجی هایی که میدونستیم چنین * داشت. (در واقع اگر داشته باشد desired value و اگر نداشته باشد observed value یا predicted value می‌گویند). داریم:

$$\begin{array}{ccccccc} x_1^1 & \dots & x_n^1 & y_1^1 & \dots & y_p^1 \\ \vdots & & & & & \\ x_1^l & \dots & x_n^l & y_1^l & \dots & y_p^l \\ \vdots & & & & & \\ x_1^L & \dots & x_n^L & y_1^L & \dots & y_p^L \end{array}$$

با همین داده‌ها بخوایم online و batch را بررسی کنیم داریم:



وقتی میخواهیم مشتق بگیریم در رگرسیون دیگه سیگما دوم رو نداریم چون k رو فیکس کردیم و داریم:

$$v_{jk}^{new} = v_{jk}^{old} - \eta \frac{\partial E}{\partial v_{jk}} \quad j=0, 1, \dots, m \\ R=1, \dots, P .$$

$$\textcircled{1} \quad \frac{\partial E_{SE}}{\partial v_{jk}} = \sum_{l=1}^L (y_k^l - \hat{y}_k^l) \frac{\partial \hat{y}_k^l}{\partial v_{jk}}$$

$$\textcircled{2} \quad \frac{\partial E_{CE}}{\partial v_{jk}} = + \sum_{l=1}^L \left(\frac{1 - \hat{y}_k^l}{1 - y_k^l} - \frac{y_k^l}{\hat{y}_k^l} \right) \frac{\partial \hat{y}_k^l}{\partial v_{jk}}$$

صفحه خالی تغذیه وایت برد جهت نوشتن و ارائه مطلب

در مقایسه بین 2 حالت، فقط قسمتی که زیرش خط کشیده ایم متفاوت است. اما مشتق جزئی ای که با رنگ سبز دورش خط کشیده ایم چطور محاسبه میشود؟ میتوان از chain rule کمک گرفت (جلسه پیش انجام شد) و مشتق سیگموید نسبت به u را نوشت. دیگر مشتق سیگموید نسبت به u را نمینویسیم (چون جلسه قبل نوشتم) و در کل داریم:

$$\frac{\partial \hat{y}_k^l}{\partial v_{jk}} = S' \left(\sum_{j=0}^m v_{jk} z_j^l \right) z_j^l \quad \hat{y}_k^l = S \left(\sum_{j=0}^m v_{jk} z_j^l \right)$$

با توجه به مشتق های جزئی گرفته شده، نتیجه میگیریم وزن v_{jk} در شبکه عصبی ابتدا بایستی یک مقدار رندم بگیرد و هریار به کمک روش گرادیان مطابق روابط زیر بروز شود تا مقدار بهتری را آخذ کند:

$$\begin{array}{l} \text{SSE} \\ \text{MSE} \end{array} \quad v_{jk}^{t+1} = v_{jk}^t - \gamma \sum_{\ell=1}^L (y_k^\ell - \hat{y}_k^\ell) S(\sum_{j=0}^m v_{jkj}^\ell) z_j^\ell$$

$$CE \quad v_{jk}^{t+1} = v_{jk}^t - \gamma \sum_{\ell=1}^L \left(\frac{1-y_k^\ell}{1-y_k^\ell} - \frac{\hat{y}_k^\ell}{y_k^\ell} \right) S(\sum_{j=0}^m v_{jkj}^\ell) z_j^\ell$$

$$j = 0, \dots, m \quad k = 1, \dots, P \quad (m+1)P \quad \text{فرز}$$

فرمول های فوق را میتوان به صورت ماتریسی نوشت تا در کامپیوتر راحت تر انجام گردد. مثال:

$$V^{t+1} = V^t - \gamma M$$

حال اگر بخواهیم وزن w_{ij} که قبل تراست را به کمک گرادیان بهبود دهیم داریم:

$$\begin{aligned} \omega_{ij}^{t+1} &= \omega_{ij}^t - \gamma \frac{\partial E}{\partial \omega_{ij}} \\ SSE \quad \frac{\partial E}{\partial \omega_{ij}} &= \sum_{\ell=1}^L \sum_{k=1}^P (y_k^\ell - \hat{y}_k^\ell) \frac{\partial \hat{y}_k^\ell}{\partial \omega_{ij}} \\ CE \quad \frac{\partial E}{\partial \omega_{ij}} &= \sum_{\ell=1}^L \sum_{k=1}^P \left(\frac{1-y_k^\ell}{1-y_k^\ell} - \frac{\hat{y}_k^\ell}{y_k^\ell} \right) \frac{\partial \hat{y}_k^\ell}{\partial \omega_{ij}} \end{aligned}$$

توجه شود که هنگام مشتق جزئی گرفتن دیگر سیگمای دوم حذف نشده است. این علت را به دو روش میتوان نشان داد:

- روش شهودی: در این روش میتوان گفت w_{ij} در شکل شبکه عصبی رسم شده، روی تمامی y_k ها میتواند اثر بگارد و لذا سیگمای آن حذف نمیشود. در حالی که v_{jk} تنها روی y_k اثر میگارد و دیگر سیگما نداری.

- روش ریاضی: هنگامی که نسبت به یک متغیری رند میگیریم، باقی متغیرها دیگر به چشم متغیر حساب نمی آیند و همگی 0 میشوند و لذا از سیگمای دوم در هنگامی که v_{jk} داریم فقط به شکل متغیری باشد و بیرون می آید و باقی 0 میشوند. در حالی که در w_{ij} همگی به چشم یک ضربی می باشند و حذف نمی شوند.

حال بایستی قسمتی که با دایره آبی مشخص شده اند را معلوم کنیم. با توجه به کمبود وقت این جلسه وقت نشد محاسبه شود و در جلسه بعد مشخص میکنیم.

یادآوری و انجام بخش آخر باقی مانده: در جلسه پیش راجع به مشتق گیری صحبت میکردیم. مشتق وزن های لایه خروجی را بدست آوردیم. ولی وزن های ورودی لایه مخفی فرصت به اتمام نرسید. در محاسبه خطای SSE. با توجه به این که $1/2$ میتواند در ضریب اتا بروز خیلی اهمیت ندارد. ولی به هر حال داریم:

$$E_{SSE} = \frac{1}{2} \sum_{l=1}^L \sum_{k=1}^P (\hat{y}_k^l - \tilde{y}_k^l)^2$$

$$E_{CE} = - \sum_{l=1}^L \sum_{k=1}^P \left(\hat{y}_k^l \log \hat{y}_k^l + (1 - \hat{y}_k^l) \log (1 - \hat{y}_k^l) \right)$$

نسبت به v مشتق گرفتیم، با توجه به این که داریم:

$$\hat{y}_k^l = S \left(\sum_{j=0}^m v_{jk} z_j^l \right)$$

میتوان گفت:

$$\begin{aligned} \frac{\partial E_{SSE}}{\partial v_{jk}} &= \sum_{l=1}^L \frac{(y_k^l - \tilde{y}_k^l)}{\frac{\partial y_k^l}{\partial v_{jk}}} e_k^l \\ \frac{\partial E_{CE}}{\partial v_{jk}} &= \sum_{l=1}^L \frac{\left(\frac{1 - \hat{y}_k^l}{1 - \hat{y}_k^l} - \frac{\hat{y}_k^l}{\hat{y}_k^l} \right)}{\frac{\partial \hat{y}_k^l}{\partial v_{jk}}} e_k^l \end{aligned}$$

با تغییر فوق 2 تاش یکی میشه و متوجه میشویم فرق کراس آنتروپی و SSE تنها در e^l می باشد.

$$\begin{aligned} \frac{\partial E_{SSE}}{\partial v_{jk}} &= \sum_{l=1}^L \frac{(y_k^l - \tilde{y}_k^l)}{\frac{\partial y_k^l}{\partial v_{jk}}} e_k^l \\ \frac{\partial E_{CE}}{\partial v_{jk}} &= \sum_{l=1}^L \frac{\left(\frac{1 - \hat{y}_k^l}{1 - \hat{y}_k^l} - \frac{\hat{y}_k^l}{\hat{y}_k^l} \right)}{\frac{\partial \hat{y}_k^l}{\partial v_{jk}}} \circ \delta'(z_j^l) \end{aligned}$$

در نتیجه می توانیم بنویسیم:

$$v_{jk}^{t+1} = v_{jk}^t - \gamma \sum_{l=1}^L e_k^l S' \left(\sum_{j=0}^m v_{jk}^j z_j^l \right) z_j^l$$

میتوانیم فرمول فوق رو یکم قشنگ ترش کیم. به طوری که خطای ضریدر مشتق را دلتای ال و کا میداریم:

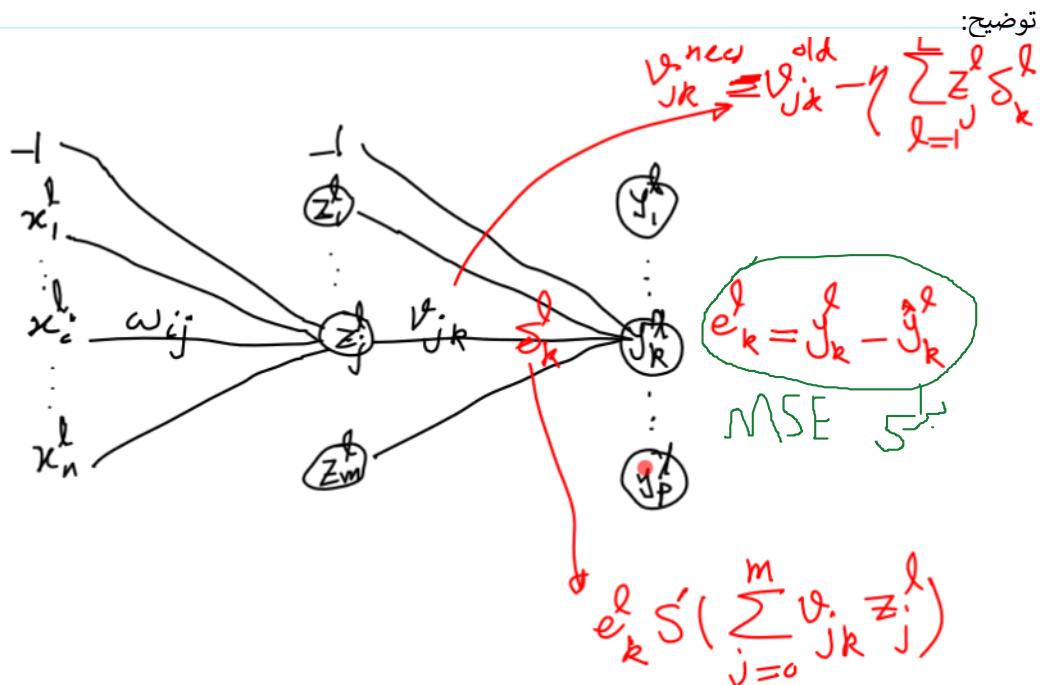
$$v_{jk}^{t+1} = v_{jk}^t - \gamma \sum_{l=1}^L e_k^l S' \left(\sum_{j=0}^m v_{jk}^j z_j^l \right) z_j^l$$

δ_k^l

$$S'(u) = S(u)(1-S(u))$$

و داریم:

$$v_{jk}^{t+1} = v_{jk}^t - \gamma \sum_{l=1}^L \delta_k^l z_j^l$$



یعنی چی؟ یعنی الان ما e_k^l رو داریم که خطای تابع سیگموید در خروجیه. این خطای خطا بعد از اعمال تابع سیگموید هستش. اگه بخوایم خطای قبلش رو بدست بیاریم باید چیکار کنیم؟ باید اون خطای رو ضرب کنیم توی مشتق تابع سیگموید تا δ_k^l حاصل بشه.

حالا برگردیم به ادامه جلسه قبل که ناقص موند. خواستیم مشتق γ نسبت به w_{ij} رو حساب کنیم. به کمک chain rule داریم

$$\frac{\partial E}{\partial w_{ij}} = \sum_{l=1}^L \sum_{k=1}^P v_k^l \left(\frac{\partial y_k^l}{\partial w_{ij}} \right) \rightarrow S' \left(\sum_{j=0}^m v_{jk} z_j^l \right) v_{jk} \frac{\partial z_j^l}{\partial w_{ij}}$$

$y_k^l = S \left(\sum_{j=0}^m v_{jk} z_j^l \right)$

$z_j^l = S \left(\underbrace{\sum_{i=0}^n w_{ij} x_i^l}_u \right)$

$\frac{\partial z_j^l}{\partial w_{ij}} = \frac{\partial z_j^l}{\partial u} \times \frac{\partial u}{\partial w_{ij}}$

$S' \left(\sum_{i=0}^n w_{ij} x_i^l \right) \xrightarrow{\quad}$

x_i^l

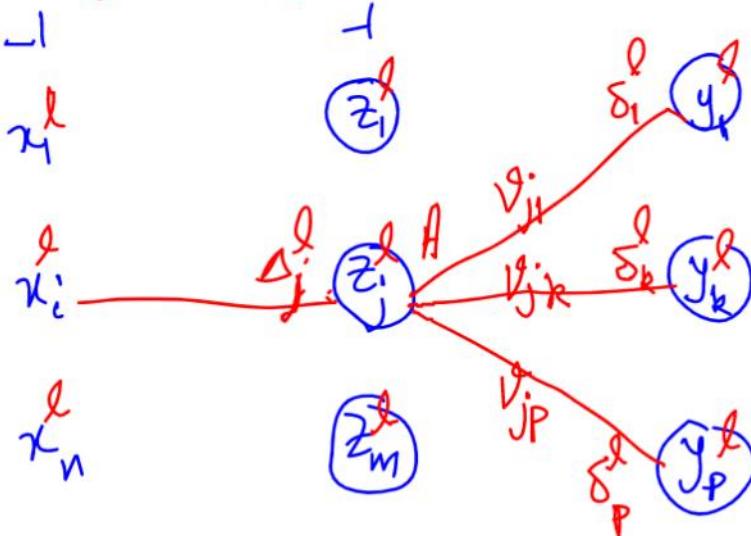
پس بخوایم w_{ij} جدید رو حساب کنیم. داریم:

$$w_{ij}^{t+1} = w_{ij}^t - \eta \sum_{l=1}^L \sum_{k=1}^P \delta_k^l v_{jk} S' \left(\sum_{i=0}^n w_{ij} x_i^l \right) x_i^l$$

$w_{ij}^{t+1} = w_{ij}^t - \eta \sum_{l=1}^L \Delta_j^l x_i^l$

در بیان شهودی داریم:

$$w_{ij}^{new} = w_{ij}^{old} - \eta \sum_{l=1}^L x_i^l \Delta_j^l$$



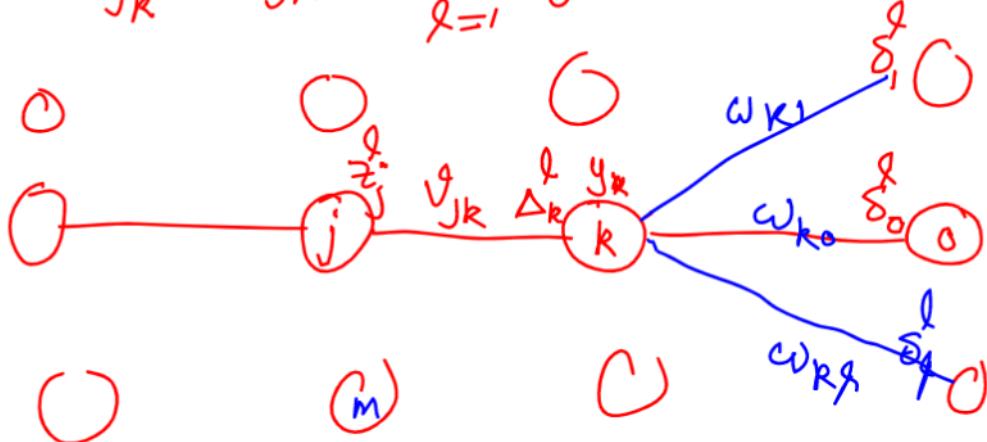
$$A = \sum_{k=1}^P v_{jk}^2 \delta_k^l$$

$$\Delta_j^l = A S' \left(\sum_{i=0}^n w_{ij} x_i^l \right)$$

با توجه به این که خطای لایه آخر به عقب میریم، به این روش Error Backpropagation می‌گویند.

اگر همین مفهوم را بسط بدھیم، در صورتی که حتی 1000 تا لایه هم داشته باشیم، میتوانیم خطای لایه قبل با وجود لایه بعدی را به روش زیر بدست آوریم:

$$v_{jk}^{new} = v_{jk}^{old} - \eta \sum_{l=1}^L z_j^l \Delta_k^l$$



$$\Delta_k^l = S' \left(\sum_{j=0}^m v_{jk} z_j^l \right) \sum_{o=1}^R w_{ko} \delta_o^l$$

پس ما همواره خطای لایه آخر را بدست می‌آوریم. همینطوری میریم عقب میرسیم به اولی که میشه یک Epoch بعد این کار را به تعداد زیادی دفعه انجام میدیم تا اوکی شه.

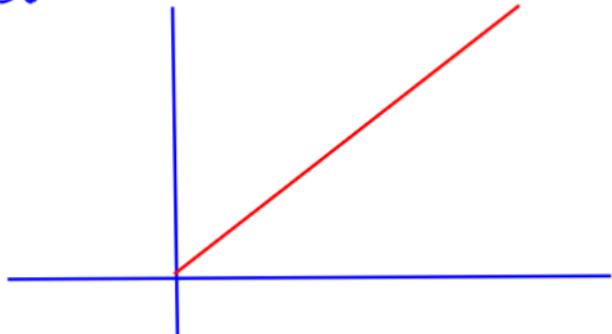
* هرچند که از لحاظ ریاضی اثبات شده است که با 3 لایه میتوان هر ابرمنحنی مورد نظر را تولید کرد. اما این کار نیاز است به صورت عمودی تعداد زیادی پرسپترون داشته باشیم. امروزه به این نتیجه رسیده اند هرچه لایه ها بیشتر باشند (به صورت افقی بیشتر داشته باشیم) میتوانیمتابع های پیچیده ای را به راحتی مدل کنیم چرا که وزن های زیاد

تری داریم. البته یک مشکلی هم که داریم اینه که وقتی تعداد لایه ها زیاد میشه برای overfit نشدن نیاز داریم داده های زیادی (حدود میلیون تا) را داشته باشیم.

* برای شبکه های عمیق وقتی لایه زیاد داریمتابع فعالیت sigmoid مناسب نیست. زیرا خطای لایه قبل که بر حسب بعدی میخواهد بدست بیاید همینطوری روی مشتق سیگموئید ضرب میشود. از طرفی مشتق سیگموئید یک عددی بین ۰ و ۱ میشود و وقتی روی هم ضرب میشن به ۰ میل میکند و عملاً گویی یادگیری نخواهیم داشت.

راه حل برای شبکه های عمیق: استفاده از تابع (ReLU Rectified linear unit): تابعی که در $x > 0$ به صورت $y = x$ و در $x < 0$ به صورت $y = 0$ می باشد.

ReLU

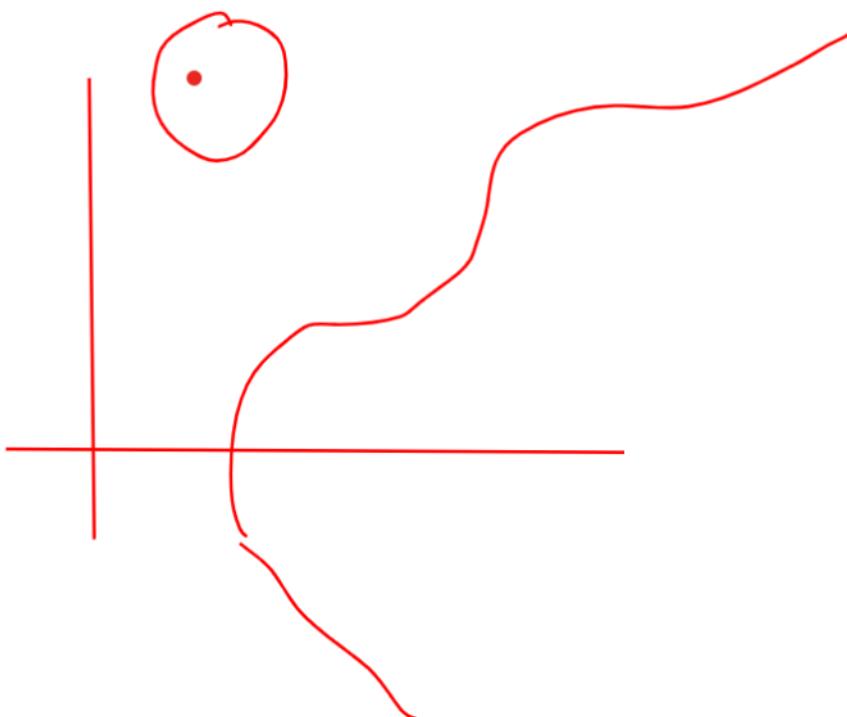


با این کار مشتق در $x > 0$ می باشد و مشکل سیگموئید را نخواهد داشت. البته با توجه به این که قسمت منفی $y = 0$ می باشد تابع ReLU نیز بعضی جاها داستان میشه و یه طوری تعییر میدن که در قسمت منفی هم یک مقداری داشته باشد.

چرا سراغ تابع فعالیت رفتیم؟ مهم ترین وظیفش این بود که غیر خطی میکنه. اگر تابع فعالیت خطی باشه ۱۰۰ تا شبکه هم داشته باشیم معادل با ۱ پرسپترون میشه. چون خطی هستند و گویی انگار شبکه نداریم.

برای توجیه دیگر مثلاً میتوان گفت تابع ReLU یک قسمت را کلا ۰ میکند و برای classification خوبه که تقسیم بندی صورت بگیره.

* شبکه عمیق بسته به نوع تابع فعالیت میتواند هر شکلی را تولید کند:



سوال: Overfitting چه می باشد؟
ما در شبکه عصبی ۲ مساله داریم.:

- یک این که آیا شبکه عصبی ما قابلیت و پیچیدگی لازم برای تولید تابع غیر خطی ما رو در حالت رگرسیون داره یا

نه. که بایس. شبکه هرچه لایه های بیشتری داشته باشد و وزن های بیشتری داشته باشد میتواند شکل های پیچیده تری را تولید کند. پس بایستی تصمیم بگیریم تابع مورد نظر ما برای مساله ما تا چه حد پیچیده باشد

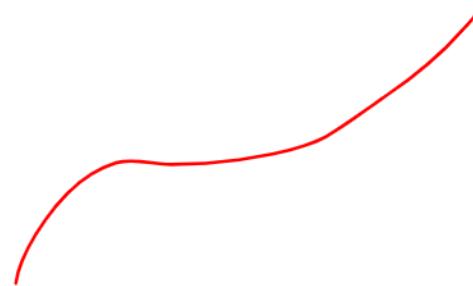
- مشکل دیگه اینه که اگر خیلی پیچیده باشه، اگر داده های آموزشی کم باشه فقط برای اونا خوب جواب میده.

(Overfitting) یعنی چون به چیز پیچیده داریم و پارامترات زیاده برای اون داده های آموزشی کم یه طوری

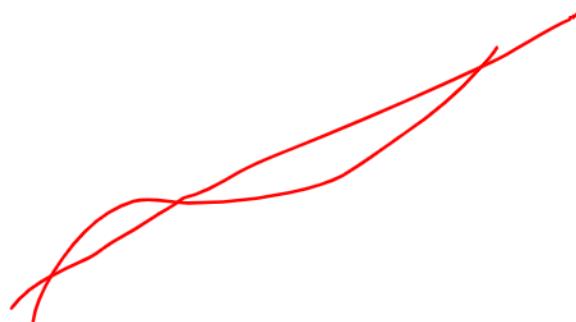
تنظیم میشه که خطای اونارو صفر میکنه و برای داده هایی که نداریم خطای خیلی زیاد میشه. ← واریانس.

در نتیجه ما اگر تعداد داده هامون زیاد باشه میتوانیم از شبکه های عمیق استفاده کنیم و مشکل overfit نداشته باشیم. ولی اگر تعداد داده ها به اندازه کافی زیاد نباشه اون موقع بایستی سطح پیچیدگی به میزانی کم بشه که overfit نشده.

مثال: فرض شود میخواهیم همچین شکلی را مدل کنیم:

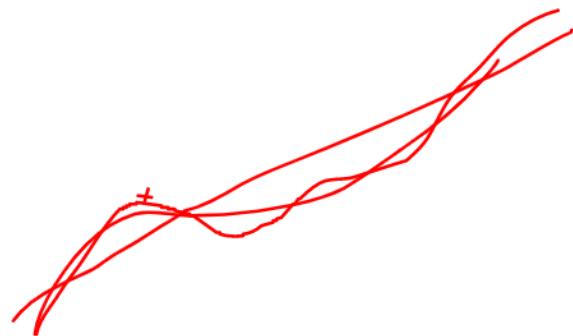


اگر با یک پرسپترون بخواهیم مدل کنیم یک خط میشود:



در این صورت این خط برای داده هایی که در داده های آموزشی مشاهده نشده خطای زیادی نخواهد داشت.

ولی اگر از یک شبکه پیچیده تری استفاده کنیم که داده های کم و حتی noise دارد، آنگاه به طوری روی آنها fit میشود که با داده های مورد نظر مشاهده نشده فاصله میگیرد و خطای زیادی شود:

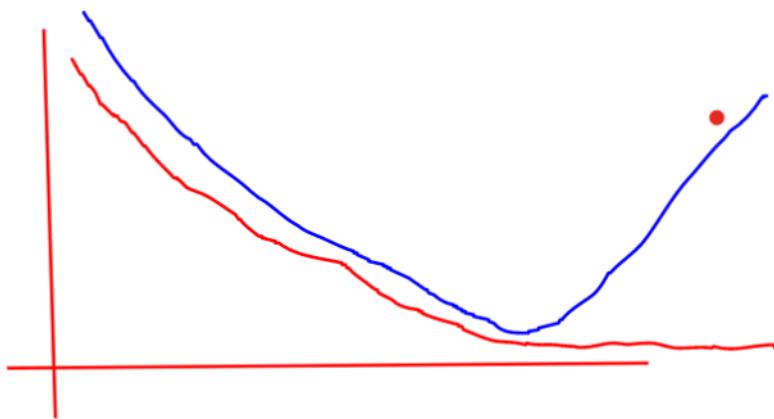


پس اگر داده کافی نداریم نمیتوانیم از شبکه های پیچیده استفاده کنیم.

اگر داده ها به اندازه کافی نباشد و بخوایم شبکه عمیق استفاده کنیم چیکار کنیم؟ داده ها رو به 3 دسته تقسیم میکنیم. دسته اول برای آموزش شبکه. دسته دوم برای تعیین کردن این که چند Epoch انجام بگیره خوبه (Validation). هدفون در دسته دوم اینه که خیلی به سمت مینیمم نزدیک نشیم تا overfit صورت بگیره. در واقع اگر ما یه منحنی داشته باشیم که محور epoch تعداد x محور عمودی خطای باشد. داریم:



مشاهده میکنیم هرچه epoch بیشتر میشود خطای رفته به رفتہ کم می شود. ولی وقتی داده های validation رو میاریم تو کار، میبینم وقتی epoch رواز یه جا به بعد بیشتر کنیم خطای نه تنها کمتر نمیشه بلکه بیشترم میشه:



از نقطه ای که دیگر validation کم نمیشه و بیشتر میشه نقطه ایست که داره overfitting صورت میگیره و باید تعداد epoch هارا دست نگه داریم

و در نهایت از داده های سوم تحت عنوان تست استفاده میشود تا خطای کار اندازه گیری شود.

* ما از روی داده های validation میتوانیم مقادیر دیگری نظری تعداد نورون ها، لایه ها و ... (به طور کلی Hyperparameters) را نیز تنظیم کنیم. به نحوی که overfitting صورت نگیرد.

تابع پایه

تابعی که بشه بقیه توابع رو بر اساس آنها نوشت.

$$f(x) = \sum_{i=1}^{\infty} w_i b_i(x)$$

تابع پایه

به عنوان مثال تبدیل فوریه، لاپلاس، تبدیل سینوس، کسینوس و .. را میتوان تابع پایه تلقی کرد.
به عنوان مثال در تبدیل کسینوس داریم:

$$f(u) = \sum_{i=1}^{\infty} w_i \cos(\omega_i u)$$



معنیش میشه این که داریم یک تابع رو به صورت ترکیب کسینوس های با فرکانس های مختلف در میاریم. (که حالا استاد مثلا 2 تاشو کشید)

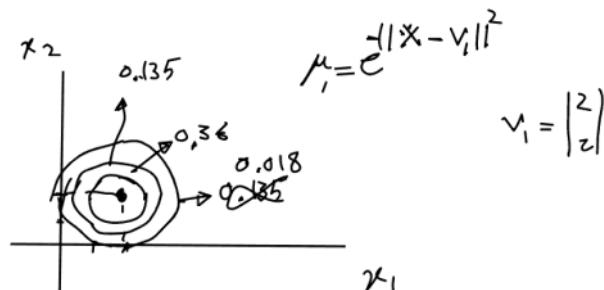
در حالت کلی مثلا در تبدیل لاپلاس تابع نمای مختلط می باشد. در تبدیل فوریه هم مشاهده کردیم نمای ساده است.
چرا به اینا تبدیل فوریه یا لاپلاس میگن؟ فضای فرکانس میشه اون W ها. یعنی با توجه به این W ها اگر W یک کسینوس با فرکانس بالا بزرگ باشه میگیم تابع در این ضریب دارای فرکانس های بالاست. وقتی میریم به فضای فرکانس فقط به W ها نگاه میکنیم. اگه تابع تغییرات زیادی داشته باشه فرکانس های بیشتری داره اگر تغییرات کمی داشته باشه فرکانس کم بیشتری داره و از طریق W ها متوجه میشویم تابع تغییراتش چقدر است.
با توجه به این که تابع تبدیل معکوس پذیر هم هستش بعد از W ها بر میگردیم به x ها. اگر بخواهیم مثلا فرکانس های بالا را حذف کنیم W هایی که برای فرکانس های بالا هست رو صفر میکنیم و بعد ($f(x)$ رو دوباره بازسازی میکنیم و تغییرات ناگهانی تابع در فرکانس های بالا حذف خواهد شد. یا مثلا میشه تابع رو فقط با W ها توصیف کرد که جنبه فشرده سازی دارد.

از نظر ریاضی اثبات می شود که هر تابعی را میتوان به صورت زیر نوشت (X به جای نوشتیم چون ممکنه ورودی ابعاد زیادی داشته باشه):

$$f(X) = \sum_{i=1}^{\infty} w_i e^{-\lambda_i X}$$

تابع پایه شعاعی

چرا بهش میگیم تابع پایه شعاعی؟ برای مثال فرض شود که $(2, 2) = v_1$ می باشد. اگر کانتور هاش رو رسم کنیم:

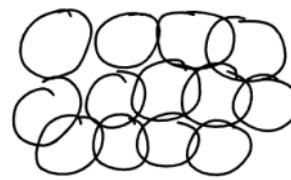


$$r_i = \sqrt{|X - v_i|^2}$$

در واقع کانتور ها در تابع شعاعی میشوند دایره هایی که مقدار تابع پایه ای در محیط این دایره ها با هم برابر هستند.

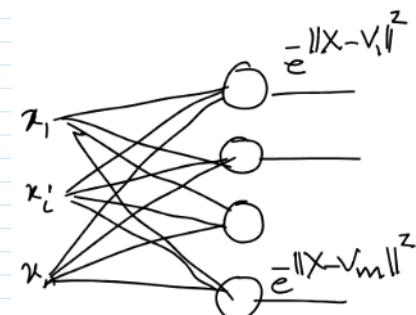
در نقطه v_1 مقدار تابع μ برابر با 1 می باشد. سپس اگر توان e را 1 کنیم. (دایره به شعاع 1) مقدار μ برابر با 0.36 خواهد شد. یعنی داخل دایره به شعاع یک مقدار μ از 1 تا 0.36 تغییر

خواهد کرد. در صورتی که شعاع دایره 2 باشد تا 0.018 کم میشه .
 میتوان گفت توابع شعاعی تا شعاع رادیکال 2 که به مقدار 0.135 میرسد را پوشش میدهد و پس از آن مقدار ناچیزی خواهد داشت.
 توابع شعاعی فضای مارا به صورت دایره های تقسیم بندی میکنند، بسته به این که مراکز به چه صورت می باشند ممکن است این توابع Overlap داشته باشند یا نداشته باشند

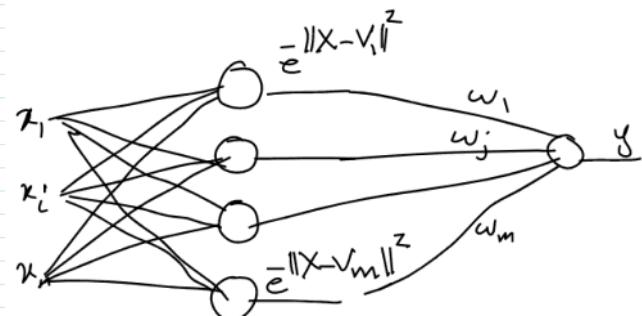


شبکه عصبی RBF - توابع پایه شعاعی (Radial Basis Function)

برای ورودی ها وزن گرفته نمیشه و خروجی به صورت یک تابع پایه شعاعی خواهد شد:



سپس در یک وزن ضرب میشوند و خروجی را تولید میکنند:



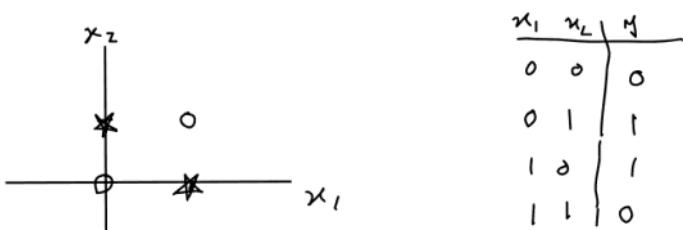
در نتیجه اگر بخواهیم فرمول y را بنویسیم:

$$y = \sum_{j=1}^m w_j e^{-\|x - v_j\|^2}$$

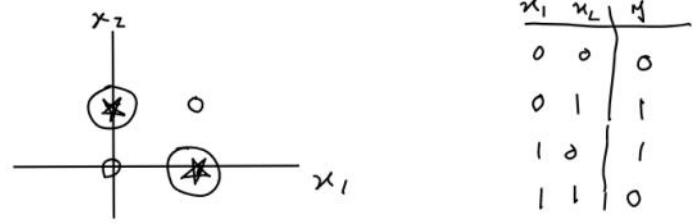
فرقش با توابع پایه اینه که سیگما تا بینهایت نیست و تا یه ثابت m هستش.

* در شبکه RBF یک لایه بیشتر نداریم. هرچند بعد ها روش های اومنده به صورت سلسه مراتبی میشه چندتا لایه درست کرد.

برای توجیه بهتر توابع شعاعی، میتوان مساله XOR را در نظر گرفت. اگر در نمودار صفر هارو با دایره 0 1 هارو با ستاره نمایش دهیم داریم:



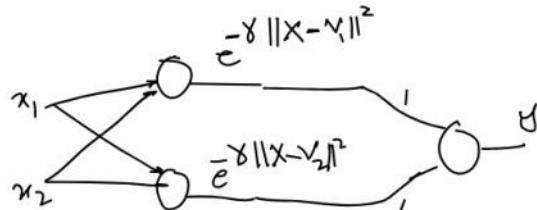
با یک پرسپترون نمیتوان این مساله را مدل کرد. در RBF با حداقل 2 تا Base function میتوان مدل کرد:



x_1	x_L	y
0	0	0
0	1	1
1	0	1
1	1	0

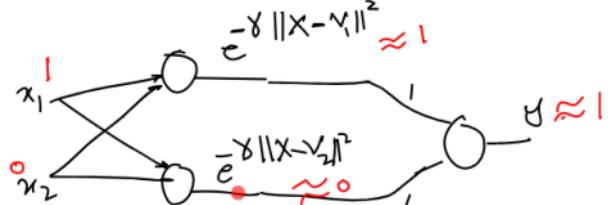
$$v_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$v_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

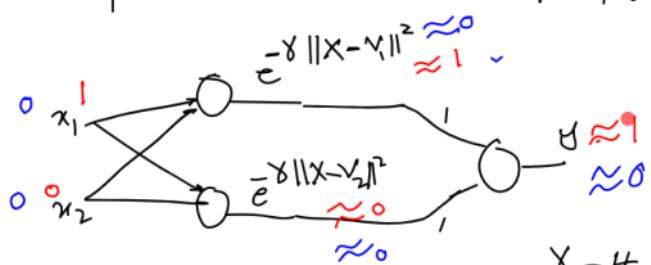


$$\gamma = 4$$

با این مدل در نواحی که ستاره داریم خروجی ما 1 و غیر آن صفر است:



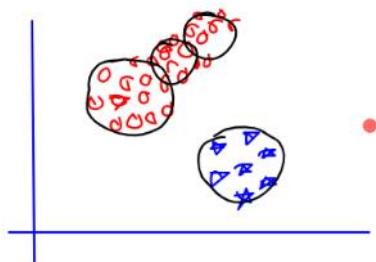
ولی مثلاً در نقطه 0 و 0 داریم:



$$\gamma = 4$$

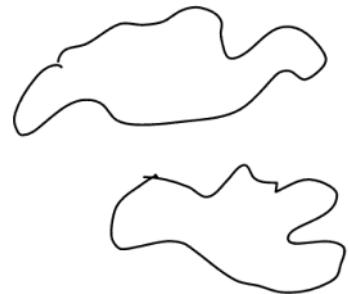
* گاما رو هرجی بیشتر بگیریم فضایی که دایره اشغال میکند کوچکتر می شود و بالعکس با کاهش گاما شعاع دایره بزرگتر میشه

در مثال پرتفا و سیب نیز اگر بخواهیم از RBF کمک بگیریم، base function، دایره های مشکی باشند داریم:

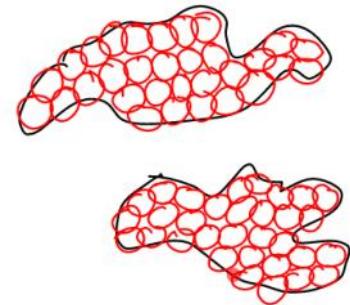


هرچه این دایره های مشکی کوچکتر باشند شکل های دقیق تری را میتوان توصیف کرد.

برای مثال اگر داشته باشیم:



برای این که بتوانیم فضای پوشش دهیم تعداد زیادی base function نیاز داریم با شعاع ثابت:



هرچه این دایره ها شعاع بیشتری داشته باشند دقت کمتری داریم و کمتر باشند دقت بیشتری داریم.

حالا باید تصمیم بگیریم چطوری این مراکز رو یاد بگیریم.

خطای SSE به روش زیر محاسبه می شود:

$$E_{SSE} = \frac{1}{2} (\mathbf{y} - \mathbf{\hat{y}})^T (\mathbf{y} - \mathbf{\hat{y}})$$

$$\mathbf{x}_1 \dots \mathbf{x}_n \begin{pmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_l \\ \vdots \\ \mathbf{y}_L \end{pmatrix} \rightarrow \mathbf{\hat{y}}$$

$$\mathbf{y} = \begin{cases} \mathbf{y}_1 = \sum_{j=1}^m w_j \mathbf{e}^{||\mathbf{x}_1 - \mathbf{v}_j||^2} \\ \vdots \\ \mathbf{y}_L = \sum_{j=1}^m w_j \mathbf{e}^{||\mathbf{x}_L - \mathbf{v}_j||^2} \end{cases}$$

$$\mathbf{x}^l = \begin{pmatrix} \mathbf{x}_1^l \\ \vdots \\ \mathbf{x}_n^l \end{pmatrix}$$

صفحه خالی نخنے وایت برد چهت نوشن و ارائه مطلب

$$y^l = \sum_{j=1}^m \omega_j \mu_j^l \quad \mu_j^l = e^{-\gamma \|x - v_j\|^2}$$

$$y^l =$$

• $y = Aw$

صفحه خالی تغییر وایت برده جهت نوشتن و ارائه مطلب

$$w = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_m \end{pmatrix}$$

$$A = \begin{pmatrix} \mu_1^1 & \mu_m^1 \\ \mu_1^2 & \mu_m^2 \end{pmatrix}$$

$$y = Aw$$

$$\hat{y} = \begin{pmatrix} \hat{y}^1 \\ \hat{y}^2 \\ \vdots \\ \hat{y}^L \end{pmatrix}$$

$$\begin{pmatrix} y^1 \\ y^2 \\ \vdots \\ y^L \end{pmatrix} = \begin{pmatrix} \mu_1^1 & \mu_1^2 & \dots & \mu_1^L \\ \mu_2^1 & \mu_2^2 & \dots & \mu_2^L \\ \vdots & \vdots & \ddots & \vdots \\ \mu_m^1 & \mu_m^2 & \dots & \mu_m^L \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_m \end{pmatrix}$$

$$\mu_j^l = e^{-\gamma \|x - v_j\|^2}$$

$$\frac{d E_{SSR}}{dw} = \frac{d}{dw} [(Aw - \hat{y})^T (aw - \hat{y})]$$

$$A^T (Aw - \hat{y}) = 0 \rightarrow A^T A w = A^T \hat{y}$$

$$w = (A^T A)^{-1} A^T \hat{y}$$

برای استفاده از کراس آنتروپی، روی خروجی بایستیتابع سیگموید بداریم که بیاد روی 0 و 1. چون توی حالت کلی خروجی ۰ میتواند هر عددی باشد.

$$E_{CE} = - \sum_{l=1}^L (\tilde{P}^l \log \tilde{P}^l + (1 - \tilde{P}^l) \log (1 - \tilde{P}^l))$$

$$\tilde{P}^l = S(y^l) = S\left(\sum_{j=1}^m \omega_j \mu_j^l\right)$$

به عنوان تمرین سی کنید به صورت برداری بنویسید و مشتق برداری بگیرید

اما در کلاس با همون صورت ک نوشتم پیش برمی داریم:

$$E_{ce} = - \sum_{l=1}^L (\hat{P}^l \log \hat{P}^l + (1-\hat{P}^l) \log (1-\hat{P}^l))$$

$$\hat{P}^l = S(y^l) = S\left(\sum_{j=1}^m w_j \mu_j^l\right)$$

$$\frac{\partial E_{ce}}{\partial w_j} = \sum_{l=1}^L \left(\frac{1-\hat{P}^l}{1-\hat{P}^l} - \frac{\hat{P}^l}{\hat{P}^l} \right) \frac{\partial \hat{P}^l}{\partial w_j} \rightarrow S'(y^l) \frac{\partial y^l}{\partial w_j}$$

$$w_j^{new} = w_j^{old} - \eta \sum_{l=1}^L \left(\frac{1-\hat{P}^l}{1-\hat{P}^l} - \frac{\hat{P}^l}{\hat{P}^l} \right) S'(y^l) \mu_j^l$$

$$w_j^{t+1} = w_j^t - \eta \sum_{l=1}^L \left(\frac{1-\hat{P}^l}{1-\hat{P}^l} - \frac{\hat{P}^l}{\hat{P}^l} \right) S'(y^l) \mu_j^l$$

* دقت شود با استفاده از سیگموید RBF بودن از بین نمیرود. بلکه انگار آخر اون شکل RBF که γ داریم، اون γ بره توی سیگموید و P رو درست کنه.

از روی همین فرمول هم میشه رابطه ماتریسی اش را نوشته. به طوری که :

$$w^{t+1} = w^t - \eta \left\{ \left[(1-\tilde{Q}) / (1-Q) - \tilde{Q} / Q \right] \star S'(y) \right\} A$$

$$\tilde{Q} = \begin{bmatrix} \hat{P} \\ \hat{P}^2 \\ \vdots \\ \hat{P}^L \end{bmatrix} \quad Q = \begin{bmatrix} P \\ P^2 \\ \vdots \\ P^L \end{bmatrix}$$

$$\tilde{Q} / Q = \begin{bmatrix} \hat{P} / P \\ \hat{P}^2 / P^2 \\ \vdots \\ \hat{P}^L / P^L \end{bmatrix}$$

صفحه خالی تغیه وایت بود جهت نوشت و ارائه مطلب

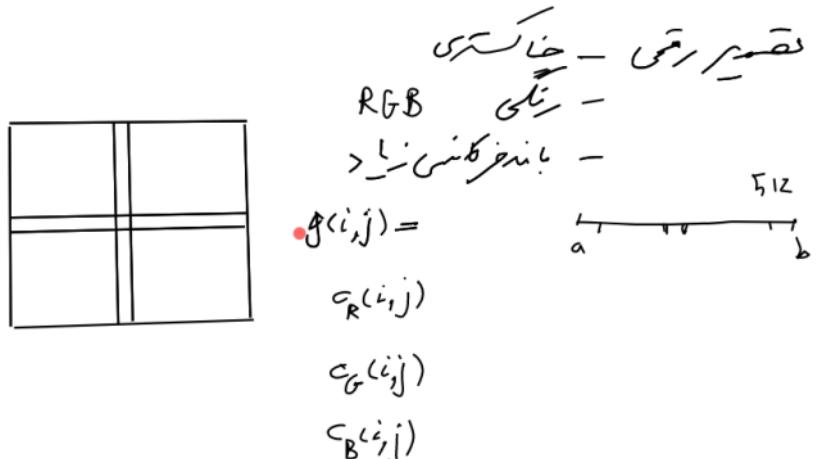
* برای یادگیری مراکز توابع شعاعی یکی از روش ها خوش بندی هستش که بعدا در این مورد صحبت خواهد شد. یا این که میتوان به صورت تصادفی انتخاب میشوند که دقت کمتری نسبت به خوش بندی دارد. گردیان به خوبی برای یادگیری مراکز جواب نمیدهد.

هر تصویر یک ماتریس 2 بعدی است که تعداد سطر و ستون ها را زولوشن را تعیین میکند. به هر نقطه از ماتریس را پیکسل میگویند.

تصاویر به 4 نوع تقسیم می شوند:

- **خاکستری (Grayscale):** هر پیکسل یک عدد بین 0 تا 255. به طوری که هرجه به 0 نزدیک باشد مشکی تر و هرجه به 255 نزدیک تر باشد سفید تر است. با $(z(i,j))$ میتوانیم میزان خاکستری بودن پیکسل در سطر i و ستون j را مشخص کنیم.
- **باینری:** تنها با صفر و یک. به طوری که صفر مشکی و 1 سفید
- **رنگی:** یک نوعش RGB می باشد. که از سه کanal قرمز، سبز و آبی تشکیل شده است و هرکدام در یک ماتریس نمایان میشوند (سه ماتریس داریم). که با $C_R(i,j)$, $C_G(i,j)$ و $C_B(i,j)$ نمایش میدهیم.
- **باند فرکانسی زیاد:** تصاویر هم با دورین های مخصوص گرفته میشود (مثل تصاویر ماهواره ای) که علاوه بر RGB، فرکانس های قابل دید را بسته به این که تصویر ماهواره ای چه میزان باند دارد، که مثلا 64 باند، 256 باند دارند یا آیند مثلا اگر از $a \times b$ را اشغال کند و به 512 تقسیم می کنند، 512 تا ماتریس حاصل می شود که هریک شدت نور متفاوتی دارند. به کمک پردازش های تصویر ماهواره ای مثلا مشخص میشود حاک از چه نوعی است، سنگ از چه نوعی است و با سطح زیر کشت چه چیزی است و ...

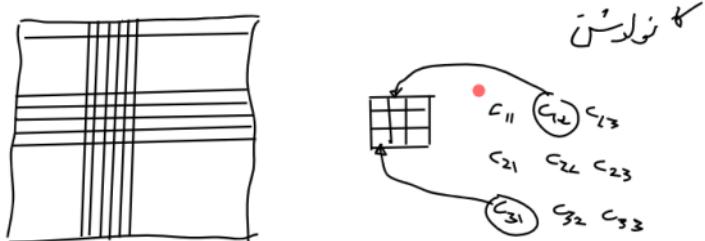
CNN - convolutional Neural Network



معمولًا در Remote Sensing از تصاویر ماهواره ای استفاده میشود. در کاربرد های معمولی ما با RGB ای هستیم یا grayscale. در RGB با توجه به این که به صورت (0-255, 0-255, 0-255) می باشد ما 256 رنگ را میتوانیم توصیف کنیم.

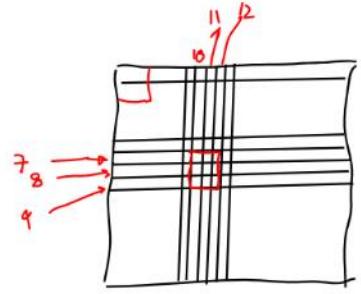
| کانولوشن

یک عملگر که بسته به این که عمگراش چی باشه میتوانه عکس روتار کنه یا لبه هارو نمایان تر کنه و ... فرض کنید که یک تصویر 32×32 پیکسل داریم (مربع سمت چپ در تصویر باین) در کانولوشن یه ماتریس کوچکتر (به اندازه سایز پنجره کانولوشن که معمولا 3×3 یا 5×5 یا 7×7 در نظر میگیرند. معمولا فرد است) در یک ماتریس بزرگتر حرکت میکند. فرض شود که یک ماتریس 3×3 داریم که مقادیر عملگر ما (ضرایب کانولوشن) را مشخص میکند.

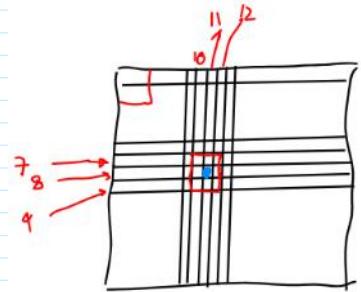


به این ماتریس کوچکتره پنجره کانولوشنی هم میگیرم.

در این پنجره کانولوشنی ما یک سری وزن وجود دارد که مشخص میکند ضرایب کانولوشن ما چه چیزی می باشد. این ماتریس کوچک در تصویر اصلی ما در پیکسل های 1 تا 3 در سطر و ستون قرار میگیرد و بدونه راست حرکت میکند تا به انتها برسد. سپس میاد پایینتر و یکدونه یکدونه میره چپ تا برسه آخر. (یعنی از راست به چپ جارو میکند میره پایین بعد به چپ چارو میکند). وقتی این ماتریس کوچیک رو روی تصویر اصلی حرکت میدیم یه تصویر جدید حاصل میشه. به صورتی که ضرایب کانولوشنی با پیکسل های تصویری ضرب میشه بعد با هم جمع میشن و پیکسل جدید وسطی رو میدن. برای درک بهتر مثلا فرض شود پنجره کانولوشنی برسه به وسیله:



اگر تصویر اولی رو با g نشون بدم. و تصویر که با اثر کانولوشن بدست میاد g' نشون بدم. پیکسل وسطی یعنی $(8, 11)$ تغییری که میکنه برابر است با:



$$g'(8, 11) = c_{11} g(7, 10) + c_{12} g(7, 11) + c_{13} g(7, 12) \\ + c_{21} g(8, 10) + c_{22} g(8, 11) + c_{23} g(8, 12) \\ + c_{31} g(9, 10) + c_{32} g(9, 11) + c_{33} g(9, 12)$$

سوال: اینطوری اکه که پیکسل های سطر و ستون اول و آخر که بدون تغییر باقی میمانه \rightarrow برای راه حل یک سطر اول و آخر، یک ستون اول و آخر به تصویر اضافه میشود. از لحاظ مقادیر هم سطر اول جدید مقدار سطر اول قیمت را میگیرد. سطر آخر جدید مقدار سطر آخر قدیم را میگیرد و ...

در شبکه عصبی یکتابع فعالیت هم روش اعمال میشود (مثل ReLU) تا اگر از یه حدی کمتر باشد صفر بگیریم:

$$g'(8, 11) = \begin{cases} c_{11} g(7, 10) + c_{12} g(7, 11) + c_{13} g(7, 12) \\ + c_{21} g(8, 10) + c_{22} g(8, 11) + c_{23} g(8, 12) \\ + c_{31} g(9, 10) + c_{32} g(9, 11) + c_{33} g(9, 12) \end{cases}$$

حال این کانولوشن چه تغییری میتوانه بدده؟ برای مثال فرض شود که پنجره کانولوشنی برابر است با:

0	0	0
-1	0	1
0	0	0

در این حالت مشتق افقی میگیرد. به صورتی که پیکسل راست هرپیکسل را منهای چی میکند و در خود میگذارد. با این کار مثلا شکل چی به راستی تبدیل میشه:



یا مثلا اگر پنجره کانولوشنی برابر با زیر باشد: (هرجند به نظرم آخرین درایه هم باید صفر باشه)

0	-1	0
0	0	0
0	1	1

اون موقع خط های عمودی حذف میشه و شکل اینطوری میشه:

~	~	~
~	~	~

یا مثلا کانولوشن زیر:

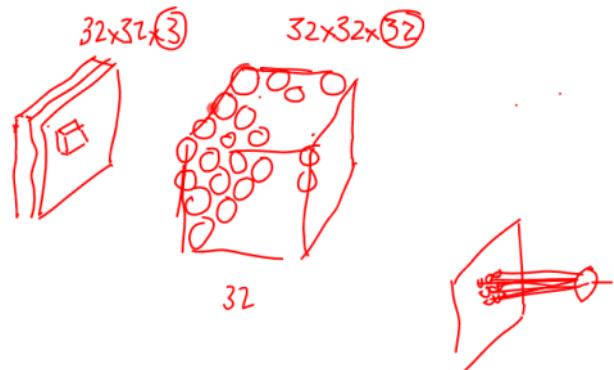
0	0	10
0	10	0
10	0	0

خط های مورب رو پر رنگ تر میکند.

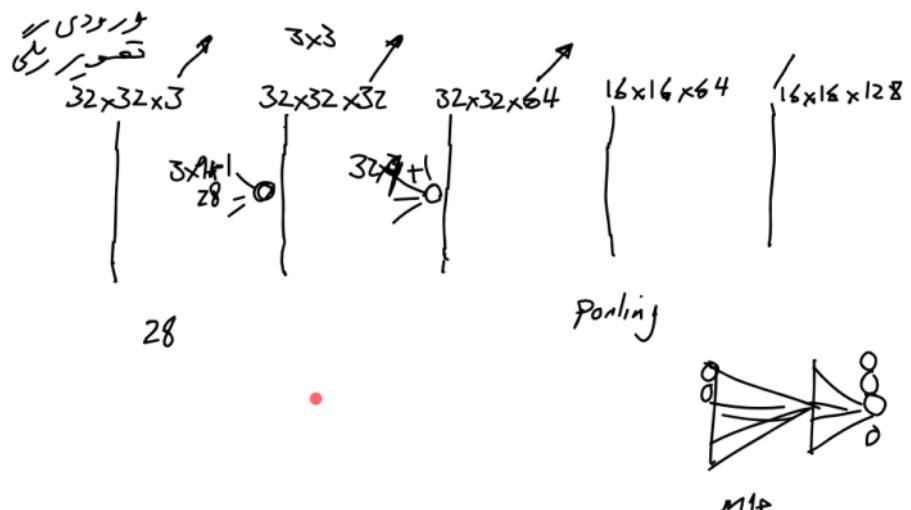
به پنجره کانولوشنی Filter هم میگر.

:CNN |

ورودی یک تصویر است. اگر تصویر را رنگ در نظر بگیریم که 32×32 می باشد. با توجه به این که 3 کanal داریم به صورت $32 \times 32 \times 3$ می شود.



در تصویر بالا پایین سمت راست یک فیلتر را نورون را مبینیم که ورودی اش 9 تا پیکسل می باشد. این حالت grayscale می باشد. در صورت که RGB باشد ورودی یک مکعب می باشد. 9 تا برای Red و 9 تا برای Green و 9 تا برای Blue و لذا 27 تا ورودی میگیرد. هر نورون در CNN معادل یک فیلتر می باشد. در مثال فوق اگر ما 32 تا فیلتر 32 تا پنجره کانولوشنی با ضرایب کانولوشن متفاوت (داشته باشیم. آنگاه هر کدام از این فیلترها باستی یک عکس باشیست) 32x32x3 را اختیار کند. پس طبق توضیحات بارگرفت فوک باستی هر فیلتر 27 تا ورودی بگیرد و یک عکس کاملاً جدید با اعمال کانولوشن تولید کند. از طرف با توجه به این که 32 تا فیلتر داریم پس باستی 32 تا عکس تولید کند و لذا $32 \times 32 \times 32$ بعد از پنهان می شود. پس متوجه شدیم هر فیلتر روی یک پیکسل اعمال میشه و یک تغییرات رو اعمال میکنه. $32 \times 32 \times 32$ تا پیکسل داشتیم و 32 تا فیلتر که نتیجش یک 32×32 شد. که هر فیلتر بسته به ضرایبی که داخلش دارد یک فیلتر به خط های عمودی توجه دارد. یک افقی. یک اورب و ...



صفحه خالی نخته و ایت برد جهت نوشن و ارانه مطلب

چرا ورودی لایه اول 28 تا شد؟ چون پنجره کانولوشنی 3×3 بود و 9 تا شد. 3 تا کanal RGB هم داشتیم که شد 27 تا. بدونه هم bias شد 28 تا

* معمولاً در CNN های استاندارد اینطوریه که لایه اول 32 باشه بعد 64 بعد 128 و ...

* در لایه سوم هم ورودی ها همونطور که تو شکل مشخص شده $9^*32 = 32$ برای پنجره کانولوشنی است و 1 برای بایاس

همینطور که جلو میریم عمق زیاد میشه. برای این که از لحاظ خروجی به مشکلی برخوریم کاری که میکن اینه که بعد یک 2 بار انجام فیلتر میان طول و عرض رو کاهش میدهند. به این کار Pooling میشود.

ما 2 نوع Pooling مشهور داریم: متوسط گیری و ماکسیمم گیری:

- متوسط گیری: 4 تا پیکسل رو متوسط میگیریم میکنیم یک پیکسل
- ماکسیمم گیری: ماکسیمم 4 تا پیکسل رو تحت یک پیکسل در نظر میگیریم.

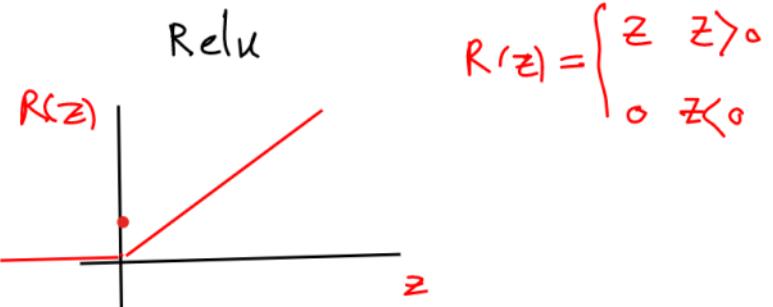
پس از Pooling در شکل مبینیم یک فیلتر دیگر اعمال شده است. پس از انجام تعدادی فیلتر، در انتها 2 لایه Fully Connected می گذارند. یعنی همه خروجی ها به همه نورون ها وصل می شوند. یعنی 2 لایه آخر مشابه MLP می شود. در لایه آخر مثلاً اگر مساله Classification باشد به تعداد کلاس ها خروجی خواهیم داشت.

حالا ما وزن های فیلتر ها و وزن های 2 لایه آخر رو چطوری بدست بیاریم؟ ← گرادیان تصادفی

$$\omega^{new} = \omega^{old} - \gamma \sum_{\delta=1}^L x \delta$$

مشابه قبیل باید به کمک backpropagation خط را انتقال بدیم به عقب و محاسبه صورت گیرد تا وزن ها اصلاح گرددند.

همانطور که از قبیل توضیح داده شد، در صورت استفاده از سیگموید برای تابع فعالیت چون تعداد لایه ها زیاد است سیگموید گرادیان رو صفر میکند (چون همین بین ۰ و ۱ عه و تو هم ضرب میشون به صفر میل میکنه) و یادگیری نخواهیم داشت. بنابر این معمولاً از ReLU استفاده میشود:



تابع ReLU از لحاظ فیلتری هم یک معنی شهودی میدهد. این که اگر فیلتری که روی تصویر اعمال میکنیم مقدارش مثبت بود ارزشمند و در نتیجه خودشو نشون میده ولی اگر کوچیک بود دیگر به درد نمیخورد و صفرش میکنیم.

گرادیان تصادفی

به جای استفاده از کل داده از mini batch استفاده میشود. فرمول یادگیری همانطور که دیدیم برابر است با:

$$\omega^{new} = \omega^{old} - \gamma \cdot \sum_{\delta=1}^L$$

این که اینقدر باشد در گرادیان تصادفی بدین شکل است که اگر مثلاً 10 هزار تا داده داشته باشیم میایم به صورت تصادفی و یکنواخت هزار تا رو انتخاب میکنیم میشه سایز mini batch. سپس در هر Epoch از این هزار تا برای آبدیت کردن وزن ها استفاده میکنیم.

مزیت: فوارکردن از مینیمم محلی. چون وقتی شبکه عمیق میشه تعداد مینیمم محلی هم زیاد میشه و وقتی از گرادیان معمولی استفاده کنیم در مینیمم محلی گیر میکنیم ولی گرادیان تصادفی شناس فوارکردن بیشتری دارد. علتش اینه که چون از همه داده ها استفاده نکردیم جهتش خیلی دقیق نمیشه که به سمت مینیمم بره و تا حدی به سمتی میرد که خطأ کمتر شود. برای درک این موضوع با شکل فرض کنیم اگر از همه داده ها استفاده کنیم اینطوری میریم به مینیمم محلی:

حالا که از بخشی از داده ها استفاده میکنیم جهتش به چپ یا راست منحرف میشه:



حالا ممکنه دیگه مینیمم محلی رو رد کنیم و به مینیمم اصلی منحرف شویم.



* در شبکه های عمیق بایستی تعداد داده ها نیز زیاد باشد. مثلاً برای CNN با هزار تا تصویر نمیشه آموزش داد و به یک میلیون تا 10 میلیون تصویر نیاز داریم.

* برای افزایش سرعت از "گرادیان تصادفی موازی" نیز استفاده میشود که در این درس به توضیح آن پرداخته نمیشود.

فرق بین متغیر قطعی و تصادفی: ما در متغیر تصادفی با خود متغیر سر و کار نداریم بلکه با احتمال سر و کار داریم. که در احتمال متغیر رو به فضای فرکانس یا فراوانی میبینیم که تعداد رخداد رو نشون میده. در متغیر قطعی فضای عوض نمیشه و با مقدار خود متغیر سر و کار نداریم.

- مثال از متغیر قطعی: قد علی 180 سانتی متره. 180 سانتی متر مقدار دقیق قد علی رو نشون میده.

- مثال از متغیر تصادفی: احتمال این که فردا بارون بیاد 60 درصده. این که احتمالش 60 درصده یعنی این که اگر ما شرایط آب و هوای مشابه فردا رو داشته باشیم. تجربه و آمار نشان میدهد از تعداد دفعات که آب و هوای مشابه فردا بوده، اگر 1000 روز بوده 600 روزش بارون اتفاق افتاده.

* زمانی میتوان از روش های آماری و احتمال استفاده کرد که **تعداد نمونه ها زیاد** باشد. مثلا من 1 میلیون بار تاس بندازم این که نشون بدم احتمال این که 2 بیاد 1/6 هستش بهتر میتونم نشون بدم تا این که 10 بار تاس بندازم. در یادگیری میتوان از روش آماری استفاده کرد چون تعداد های زیادی میتوانیم داده داشته باشیم.

متغیر نادقيق (فازی): متغیری که مقدار دقیقش رو نمیدونیم. مثلا بگیم "قد کوتاه". ما از یک بازه قد برای قد کوتاه استفاده میکنیم مثل 100 تا 130 سانتی متر. دقیق نمیشه گفت مقدار "قد کوتاه" چند سانتی متر است.

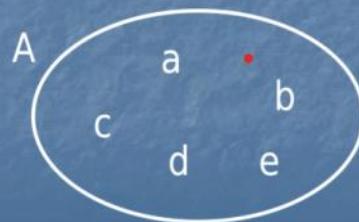
دقیق شود این مفهوم با متغیر تصادفی متفاوت است. در متغیر تصادفی ما همواره مقدار رو میبریم تو فضای فرکانس و در مرور فراوانیش صحبت میکنیم ولی در متغیر نادقيق مقدار به صورت دقیق قابل تشخیص نیست و لذا یک بازه تعریف میشود.

برای این که داخل بازه بتوانیم بین 100 تا 130 تفاوت قائل بشیم، اگر اهمیت بین قد 130 و 100 متفاوت باشد، میایم یک مقدار تعلق تعريف میکنیم. میگیم 130 سانتی متر مثلا 0.4 تعلق به قد کوتاه تعلق داره و 100 سانتی متر 1 تعلق داره به قد کوتاه. (مقدار تعلق همیشه بین 0 و 1 هستش و مجموعشون لزوما 1 نیست).

در بعضی موارد که برای معمون تفاوتشون اهمیت نداره تعلق همه رو 1 در نظر میگیریم. ← متغیر فازی بازه ای اگر تفاوت داشته باشند → متغیر فازی استاندارد

مجموعه

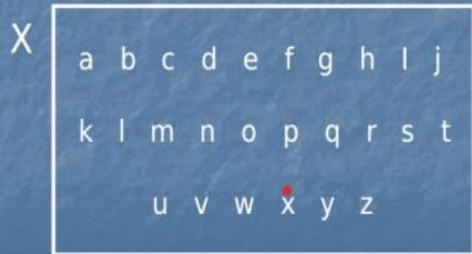
کنار هم قرار گرفتن تعدادی از اشیاء (اعضاء) که دارای ویژگیهای مشترکی هستند
مائند مجموعه شهرهای دنیا



مجموعه مرجع (X): اگر مثلا اعداد ما با 1 تا 10 سر و کار داشته باشند مجموعه مرجع ما میشه اعداد 1 تا 10. بنابراین این مسئله است که مشخص میکند مجموعه مرجع ما چه باشد.

مجموعه مرجع

مجموعه ای که شامل تمامی اشیاء ممکن در مسئله مورد نظر ما باشد



برای مثال مجموعه مرجع فوق تمام حروف انگلیسی کوچک می باشد.

زیرمجموعه: همواره داخل مجموعه مرجع

زیرمجموعه

مجموعه ای که شامل تعدادی از اشیاء یک مجموعه باشد
مانند مجموعه شهرهای ایران که زیر مجموعه شهرهای دنیا است

$A \subset X$ if $x \in A$ then $x \in X$



نمایش مجموعه:

• اگر تعداد اعضای محدود \rightarrow توسط اعضای

• اگر تعداد اعضای زیاد، مثل اعداد حقیقی بین 0 تا 10 \rightarrow توسط قائد

• توسط تابع تعلق: مشابه با روش احتمال که فضای متغیر رو مبیند توی فضای فرکانس در تابع تعلق فضای متغیر به فضای تعلق برده می شود. تابع تعلق را با $\mu_A(x)$ نمایش میدهند. در $\mu_A(x)$, A اسم مجموعه فازی و x متغیر ما می باشد. در حالت کلاسیک میتوان تابع تعلق را برابر با زیر تعریف کرد:

$$\mu_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases}$$

این که متغیر ما تا چه حد به بازه تعلق دارد فضای تعلق را مشخص میکند.

* جمع فراوانی ها میشه فراوانی کل. یعنی مثلا 10 بار تاس رو بندانزیم تعداد دفعه های تکرار 1 تا 6 رو جمع بزنیم باید 10 بشه. ولی در فضای تعلق چون فراوانی نداریم جمع تعلق ها حتما باید 1 بشه.

تئوری مجموعه ها

نمایش مجموعه

توضیح اعضاء

$$\rightarrow A = \{a, b, c\}$$

توضیح قائد

$$A = \{x \mid x \in N, x \leq 10\}$$

توضیح تابع تعلق

$$\sum \mu = 4$$

$$A = \{1, 2, 3, 4, 5, 6\} \quad \mu_A(x) = 1 \quad x \leq 10 \quad \sum \mu = 4$$

$$\mu_A(1) = 1 \quad \mu_A(2) = 1 \quad \mu_A(3) = 1 \quad \mu_A(4) = 1 \quad \mu_A(5) = 1 \quad \mu_A(6) = 1$$

در مثال زده شده یه تاس رو برداشته همه تعلق هارو 1 گرفته جمعش شده 6 در صورتی که احتمال بود حتما باید جمعش 1 میشد.
با تعمیم تعریف فوق، این که تابع تعلق علاوه بر 0 و 1 مقداری بین 0 و 1 نیز اختیار میتواند کند میتوانیم نظریه مجموعه های کلاسیک را بسط بدیم.

اگر بخواهیم مجموعه ها را با تابع تعلق نمایش بدهیم، یک راهش نمایش برداریه. یعنی مثلا اگر مجموعه ما $\{a, b, c\}$ باشه. اون قسمی که استاد دورش خط کشیده میشه روش برداریه. جاهاي که 1 شده یعنی تعلق داره جاهاي که 0 شده تعلق نداره:

تئوری مجموعه ها

$X = \{a, b, c\}$	$A = \{\sum \mu_A(x)/x\}$	مجموعه مرجع
$= N$		تعداد اعضاء
		$= 2^N = 8$ تعداد زیرمجموعه ها
{}	{0,0,0}	$\{0/a+0/b+0/c\} = \{\}$
{a}	{1,0,0}	$\{1/a+0/b+0/c\} = \{1/a\}$
{b}	{0,1,0}	$\{0/a+1/b+0/c\} = \{1/b\}$
{c}	{0,0,1}	$\{0/a+0/b+1/c\} = \{1/c\}$
{a,b}	{1,1,0}	$\{1/a+1/b+0/c\} = \{1/a+1/b\}$
{a,c}	{1,0,1}	$\{1/a+0/b+1/c\} = \{1/a+1/c\}$
{b,c}	{0,1,1}	$\{0/a+1/b+1/c\} = \{1/b+1/c\}$
{a,b,c}	{1,1,1}	$\{1/a+1/b+1/c\} = \{1/a+1/b+1/c\}$

* ترتیب اعضاء بر اساس ترتیب اعضای مجموعه است.

* اینجا پاورپوینت هم یکم به هم ریخته داره میگه اکه تعداد اعضاء N = باشه تعداد زیرمجموعه میشه 2^N

در روش غیربرداری تعلق های صفر رو نمایش نمیدهیم. اونایی که تعلق دارن رو 1 میذاریم منتهی به خط کسری داریم و مخرج خود عضو رو مشخص میکنیم:

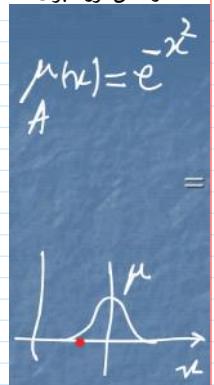
$$A = \left(\frac{1}{a} + \frac{1}{b} \right)$$

که حالا به نحو کلی ترش داریم:

$$A = \left(\frac{M(a)}{a} + \frac{M(b)}{b} \right)$$

* جدا کردن اعضاء در روش غیربرداری با "+" یا "-" انجام میشه.

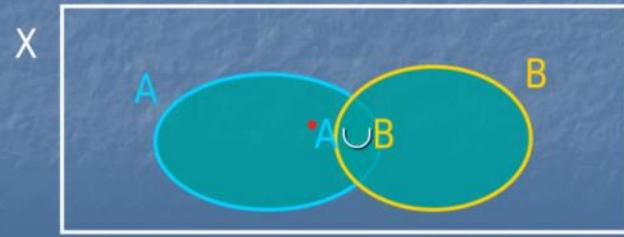
مقدار تعلق لزوما برای اعضای گستته نیست. بر فرض میتوان برای مجموعه پیوسته نوشت:



در حالت کلاسیک یک سری عملیات روی مجموعه داریم که حالا میخوایم با تابع تعلق تعریف شون کنیم.
- اجتماع:

$$A \cup B = \{x \mid x \in A \vee x \in B\}$$

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) = \text{Max}\{\mu_A(x), \mu_B(x)\}$$

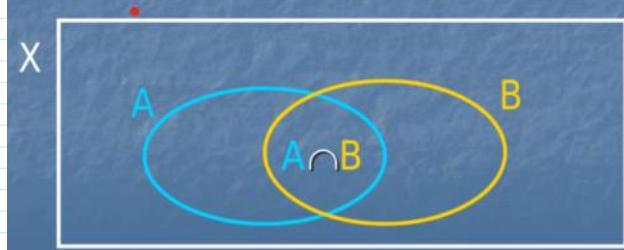


عملگر OR را میتوان با ماکس تعریف کرد. چون اگر یکیشون 1 باشه اجتماع هم 1 میشه.

- اشتراک:

$$A \cap B = \{x \mid x \in A \wedge x \in B\}$$

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) = \text{Min}\{\mu_A(x), \mu_B(x)\}$$



چرا AND ؟ چون وقتی اشتراک 1 میشه باید وسط نمودار 1 بشه که تنها زمانی که هردو 1 باشنند. از طرف MIN میشه چون خروجی MIN تنها زمانی 1 میشه که هردوشون 1 باشن.

- متمم

$$A' = \{x \mid x \notin A \wedge x \in X\}$$

$$\mu_{A'}(x) = 1 - \mu_A(x)$$



جهایی که مجموعه تعلق داره باید 0 بشه و جهایی که نیست باید 1 بشه.

خواص مجموعه ها:

عملیات روی مجموعه ها – خواص

جایجاوی

$$A \cup B = B \cup A$$

$$A \cap B = B \cap A$$

شرکت پذیری

$$A \cup (B \cup C) = (A \cup B) \cup C$$

$$A \cap (B \cap C) = (A \cap B) \cap C$$

توزیع پذیری

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

دمرگان $(A \cup B)' = A' \cap B'$

$$(A \cap B)' = A' \cup B'$$

عملیات روی مجموعه ها – خواص

$$A \cap A = A \cup \emptyset = A \cap X = A \quad A \cup A =$$

$$A \cap \emptyset = \emptyset$$

$$A \cup X = X$$

$$(A')' = A$$

$$A \cap A' = \emptyset$$

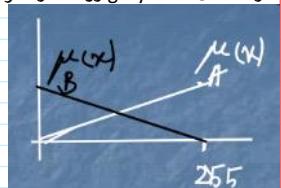
$$A \cup A' = X$$

If $A \subseteq B \subseteq C$ then $A \subseteq C$

از لحاظ زبانی وقتی در مورد اعضا صحبت میکنیم (به صورت جمله ای) از اجتماع و اشتراک استفاده میکنیم. وقتی که در راجع به تعلق صحبت میکنیم، لغت به کار آمده AND و OR می باشد.

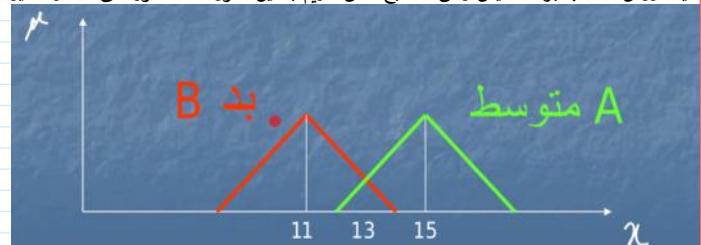
حالا میخوایم بین 0 و 1 باشے اشکالی برای نظریه مجموعه ها پیش می آید یا نه. اگر نیاد میتوانیم نظریه مجموعه ها را تعمیم بدیم.

رنگ های grayscale را اگه در نظر بگیریم. در باز 0 تا 255 هرچه به 255 نزدیک میشیم تعلق به سفید بیشتر میشه هرچه به 0 نزدیک میشیم تعلق به سیاه بیشتر میشه:



علاوه بر مورد فوق، در مجموعه کلاسیک نمیتوان سرعت زیاد، قد کوتاه یا زیبایی را مدل کرد. با توجه به این که انسان نیز از این موارد استفاده میکند برای مدل کردن انسان باستی از مجموعه فازی کمک گرفت.

یک روش مناسب برای نمایش وقتی که تابع تعلق داریم به این صورته که محور افقی مقدار متغیرها و محور عمودی مقدار تعلق باشد:

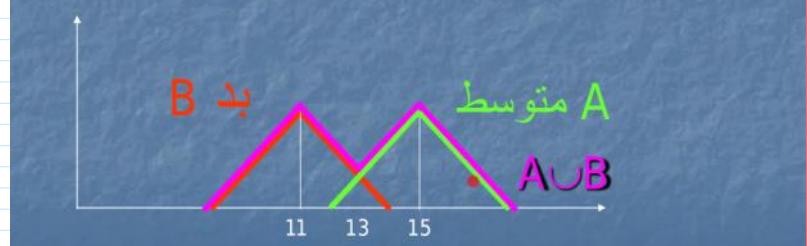


ما اگه فرض کنیم نمره 14 به اندازه 0.8 به نمرات متوسط تعلق دارد، معنیش این نیست که 10 بار امتحان بدیم 8 بارش 14 میشه (معنی احتمال نمیده). بلکه ارزش اون نمره رو نشون میده. جمع مقدار تعلق به موزان Overlap زیاد باشه از 1 بیشتر میشه جمیعش. اگر کیپ بشن جمیعش 1 میشه و اگر فاصله داشته باشن از 1 کمتر میشه.

تعلق اجتماع:

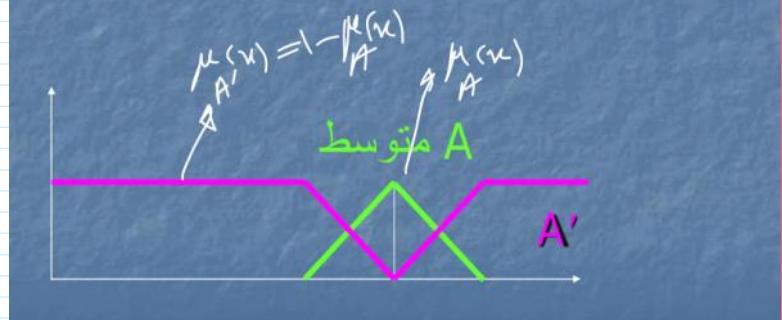
عملیات روی مجموعه های فازی - اجتماع

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) = \text{Max}\{\mu_A(x), \mu_B(x)\}$$



عملیات روی مجموعه های فازی - متمم

$$\mu_{A'}(x) = 1 - \mu_A(x)$$



عملیات روی مجموعه های فازی - خواص

جای جایی

$$A \cup B = B \cup A$$

$$A \cap B = B \cap A$$

شرکت پذیری

$$A \cup (B \cup C) = (A \cup B) \cup C$$

$$A \cap (B \cap C) = (A \cap B) \cap C$$

توزیع پذیری

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

$$A \cap (B \cup C) \Rightarrow A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

دمرگان

$$(A \cup B)' = B' \cap A'$$

$$(A \cap B)' = B' \cup A'$$

15

عملیات روی مجموعه های فازی - خواص

$$A \cup A = A \cap A = A \cup \emptyset = A \cap X = A$$

$$A \cap \emptyset = \emptyset$$

$$A \cup X = X$$

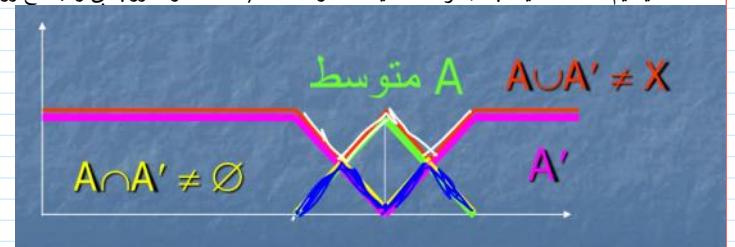
$$(A')' = A$$



If $A \subseteq B \subseteq C$ then $A \subseteq C$

$$A \subseteq C \Rightarrow \mu_A(x) \leq \mu_C(x)$$

مشاهده میکنیم که 2 خاصیت با مجموعه کلاسیک متفاوت است (استاد اشتراک رو با آبی و اجتماع رو با سفید کشید):



اگر قرار بود مجموعه مرجع رو نشون بدیم یک خط 1 میشه:



برای مجموعه تهی هم یک خط صفر میشه

با توجه به این که این 2 خاصیت، از خواص های اصلی مجموعه نمی باشد (خاصیت های اصلی دمرگان و ... می باشند) میتوانیم بگوییم مجموعه فازی تعمیم مجموعه های کلاسیک می باشد.

چرا اجتماع رو MAX و اشتراک رو MIN و متمم رو 1 - مجموعه تعريف میکنیم؟ نمیشه جور دیگه تعريف کرد؟ درصورتی میتوان داشت که 2 خاصیت زیر را ارضاء کند:

- درصورتی که 4 خاصیت اصلی رو داشته باشه (جای جایی، شرکت پذیری، توزیع پذیری و دمرگان)
- شرط مزدی: اگر مقادیر تعلق 0 و 1 شد، اجتماع و اشتراک و .. مثل حالت استاندارد باشد. (جون میخواهیم تعمیم باشد) یعنی مثلا در شکل زیر رو در نظر بگیریم. اگر 2 ستون اول ورودی هامون باشه و ستون سوم اشتراک و ستون آخر اجتماع باید داشته باشیم:

0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	1

همچنین برای متمم هم باید داشته باشیم:



ما روی اجتماع و اشتراک ها میتوانیم یه سری شرایط بداریم که اگر این شرایط برقرار باشه اون موقع میتوانه این خاصیت هارو خراب نکنه و شرایط مرزی هم داشته باشه در نتیجه نظریه مجموعه رو خراب نکنه.
این اجتماع های جدید رو اس نرم و اشتراک هارو فی نرم میگیم.

اجتماع فازی - اس نرمها

$\begin{matrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$	\rightarrow	$s(0,a) = s(a,0) = a$ $s(1,1) = 1$
		شرایط مرزی $s(a,b) = s(b,a)$
		شرط جابجایی $s(a_1, b_1) = a_1 \sqrt{b_1}$
		شرط صعودی $s(s(a,b),c) = s(a,s(b,c))$

If $a_1 \leq a_2$, $b_1 \leq b_2$ then $s(a_1, b_1) \leq s(b_2, a_2)$

شرط شرکت پذیری

در بین شرایط مرزی این که اجتماع ۱ با ۱ میشه ۱ همونظور که استاد مشخص کرده شرط آخر رو پوشش میده و اشتراک a با ۰ میشه a اون ۳ شرط باقی رو پوشش میده چون تو هرگذوم از اون ۳ تا حداقل بدونه صفر رو داره و اون یکی a تلقی میشه.

* دقت شود $s(1,a) = a$ لزوماً برابر با ۱ نیست. با استفاده از اون دو شرط مرزی موجود در اسلاید فوق میتوان $0, 0, 1, 0, 1, 1$ رو مدل کرد ولی میشه یک نوع فرمول اجتماع غیر از ماکسیمم گیری (که حالا در اسلاید های جلسه بعد هست) تعریف کرد که اجتماع a و 1 لزوماً 1 نشه

این که مثلاً میایم اجتماع رو میگیریم MAX و اشتراک رو MIN به این میگن یک کلاس نرمال. دانشمندان دیگری اومدن کلاس های نرمال دیگری هم ساختن که توی این جلسه بپوشون می بردایم.

* درق نرم عضو خنثی 1 هستش (جلسه پیش دیدیم در اس نرم عضو خنثی 0 بود) و میتوان گفت:

اشتراک فازی - تی نرمها

شرایط مرزی $t(0,0) = 0$

شرط جاپجایی $t(a,b) = t(b,a)$

شرط صعودی $t(1,a) = t(a,1) = a$

If $a_1 \leq a_2, b_1 \leq b_2$ then $t(a_1,b_1) \leq t(b_2,a_2)$

شرط شرکت پذیری $t(t(a,b),c) = t(a,t(b,c))$

برای متمم هم داریم:

متuum فازی

شرط نزولی $c(0) = 1$

شرط نزولی $c(1) = 0$

If $a_1 \leq a_2$ then $c(a_2) \leq c(a_1)$

شرط نزولی هم داریم چون متuum عکس میکنه.
* برای کلاس نرمال متuum باید خاصیت دموگان رو از بین نبرد.

حالا بایم کلاس های نرمال رو بپردازیم (نیازی به حفظ نیست صرفاً میخوایم به گوشمنون بخوره):

کلاس دومبی

$$s_\lambda(a,b) = 1 / (1 + ((1/a - 1)^{-\lambda} + (1/b - 1)^{-\lambda})^{-(1/\lambda)})$$

$$t_\lambda(a,b) = 1 / (1 + ((1/a - 1)^{+\lambda} + (1/b - 1)^{+\lambda})^{+(1/\lambda)})$$

$$\lambda \in (0, \infty)$$

یک پارامتر لمباداریم بین 0 و بینهایت که با تغییر این پارامتر اشتراک و اجتماع خواص اجتماع و اشتراک عوض میشود. راجع به این که با تغییر پارامتر اشتراک و اجتماع چطوری عوض میشن در جلوتر توضیح داده میشه.

برای کلاس هایی که متuum تعریف نمیشه متuum همون متuum استاندارد

کلاس یاگر

$$s_w(a,b) = \text{Min}(1, (a^w + b^w)^{1/w})$$

$$t_w(a,b) = 1 - \text{Min}(1, ((1-a)^w + (1-b)^w)^{1/w})$$

$$c_w(a) = (1-a^w)^{1/w}$$

$$w \in (0, \infty)$$

جمع و ضرب دراستیک

$$s_{ds}(a,b) = \{ a \text{ if } b=0, b \text{ if } a=0, 1 \text{ else} \}$$

$$t_{dp}(a,b) = \{ a \text{ if } b=1, b \text{ if } a=1, 0 \text{ else} \}$$

جمع و ضرب دراستیک اجتماع و اشتراک خوبی نیست. برای این تعریف شده که دو خاصیتی که برقرار نبود برقرار بشن. یعنی میتوان گفت:

$$A \cup A' = X$$

$$A \cap A' = \emptyset$$

جمع و ضرب دراستیک

$$s_{ds}(a,b) = \{ a \text{ if } b=0, b \text{ if } a=0, 1 \text{ else} \}$$

$$t_{dp}(a,b) = \{ a \text{ if } b=1, b \text{ if } a=1, 0 \text{ else} \}$$

وقتی که این خاصیتا بخوان اتفاق بیافتن خود اجتماع و اشتراک ها جالب نمیشن. که در اسلاید های بعدی مشاهده میکنیم.

جمع و ضرب اینشتین

$$s_{es}(a,b) = (a+b)/(1+ab)$$

$$t_{ep}(a,b) = ab/(2-(a+b-ab))$$

جمع و ضرب جبری

$$s_{as}(a,b) = a+b-ab$$

$$t_{ap}(a,b) = ab$$

کلاس سوگنو

$$c_\lambda(a) = (1-a)/(1+\lambda a)$$

جمع و ضرب جبری معادل اجتماع و اشتراک در احتمال است وقتی که مستقل باشند:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

$$P(A \cap B) = \underbrace{P(A)}_{P(B)} P(B|A)$$

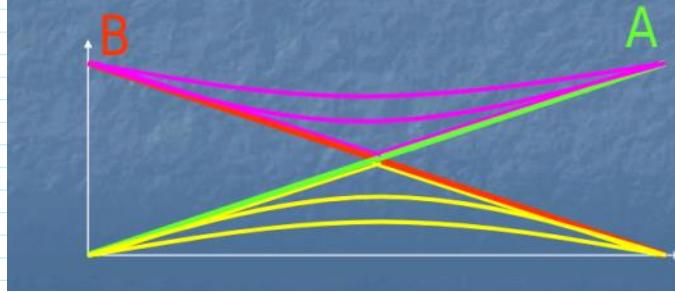
حالا برایم ببینیم چه فرقی دارن:

تئوری مجموعه ها

$$t_{dp} < t_{ep} < t_{ap} < \text{Min} < \text{Max} < s_{as} < s_{es} < s_{ds}$$

$$t_{dp} \leq t_{\lambda}, t_w \leq \text{Min} < \text{Max} \leq s_w, s_{\lambda} \leq s_{ds}$$

- $\text{Min} < \text{Averaging operators} < \text{Max}$



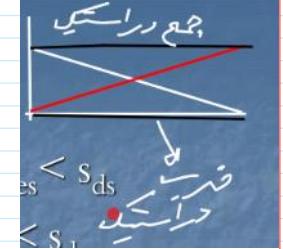
اگر یک مجموعه فازی را اینطوری تعریف کنیم:



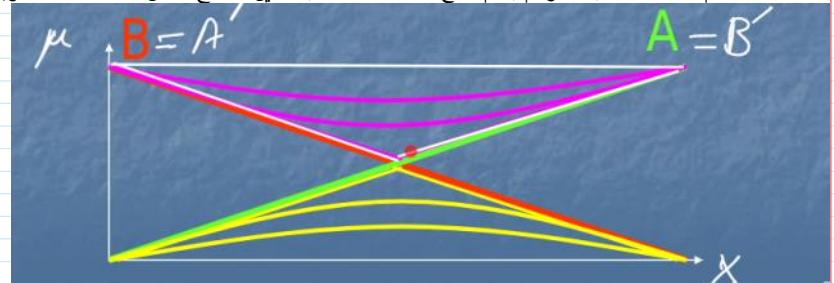
متمن اش بر عکس از صفر به بالا میره.



حال سوال میشه اجتماع و اشتراک چطوری عمل میکنه؟
در جمع دراستیک و ضرب دراستیک اجتماعیش یک خط $y=1$ و اشتراکش $y=0$ میشه:

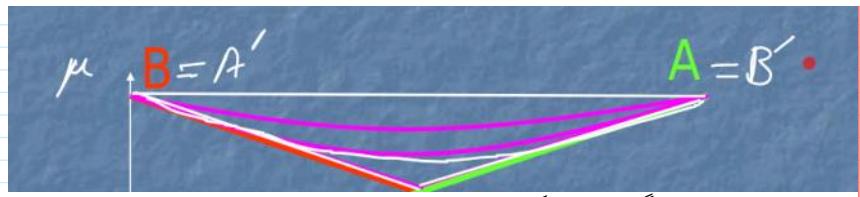


در شکل رسم شده در اسلاید میتونیم بگیم جمع دراستیک میشه بیشترین اجتماع ممکن و MAX میشه کوچکترین اشتراک ممکن (استاد با زنگ سفید کشید)

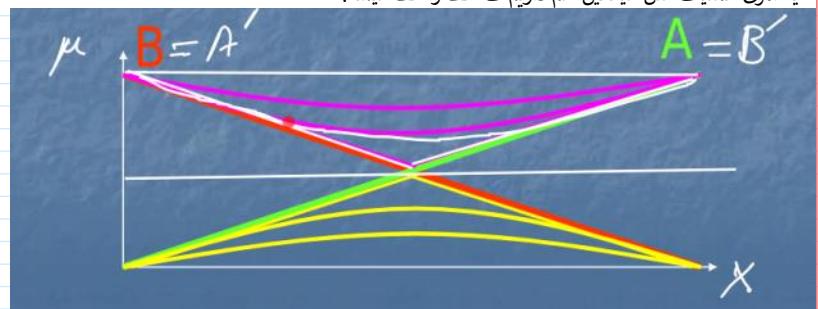


همینطور میشه گفت MIN میشه بزرگترین اشتراک ممکن و ضرب دراستیک کوچکترین اشتراک ممکن میشه.

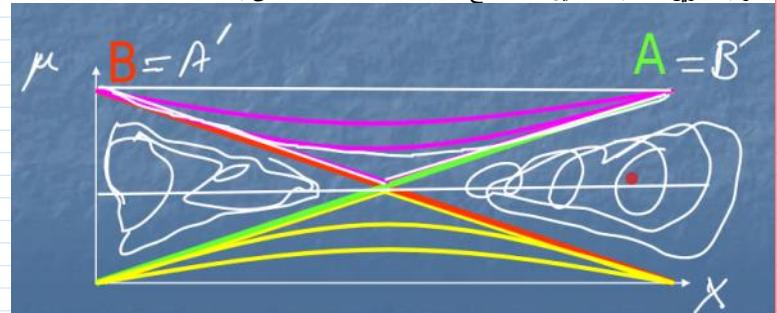
جمع جبری یک منحی ما بین میشه:



به سری عملیات مثل میانگین هم داریم ک خط واسط میشه:



نردیک ترین خط به میانگین در اجتماع MAX و در اشتراک MIN می باشد.



اجماع و اشتراک هیچکدومش این ناحیه رو cover نمیکن.

---اتمام اسلامید
تمرین:

$$\text{حجم کم} = A = \left\{ \frac{0.2}{1} + \frac{0.9}{2} + \frac{0.5}{3} \right\}$$

$$\text{حجم کم} = B = \left\{ \frac{0.4}{1} + \frac{0.8}{2} + \frac{0.3}{3} \right\}$$

$$A \cap B = \left\{ \frac{0.2}{1} + \frac{0.8}{2} + \frac{0.3}{3} \right\}$$

تصویر فوق غلطه! اینا با هم اشتراک ندارن. چرا؟ چون اینا واحداًشون فرق داره:

$$\text{حجم کم} = A = \left\{ \frac{0.2}{cm^3} + \frac{0.9}{m^3} + \frac{0.5}{n^3} \right\}$$

$$\text{حجم کم} = B = \left\{ \frac{0.4}{1} + \frac{0.8}{2} + \frac{0.3}{3} \right\}$$

یکیل یکیل

~~$$A \cap B = \left\{ \frac{0.2}{1} + \frac{0.8}{2} + \frac{0.3}{3} \right\}$$~~

به شرطی میتوان اشتراک گرفت که اینا هم مرجع بشن. مثلاً فرض کنیم يك C داریم که حجم متوسطه، چون واحداً یکیه اگه اشتراک بگیریم داریم:

$$\mu_A = A = \left\{ \frac{0.2}{1} + \frac{0.4}{2} + \frac{0.5}{3} \right\}_{cm^3}$$

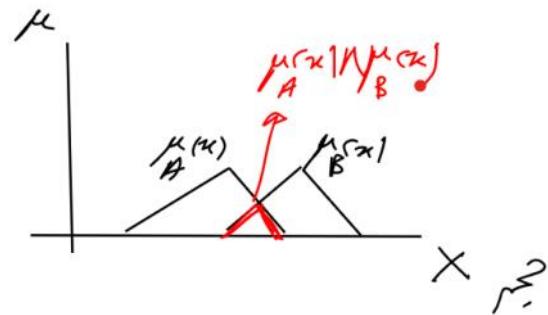
$$\mu_B = B = \left\{ \frac{0.4}{1} + \frac{0.8}{2} + \frac{0.3}{3} \right\}_{cm^3}$$

$$A \cap B = \left\{ \cancel{\frac{0.2}{1}} + \cancel{\frac{0.8}{2}} + \cancel{\frac{0.3}{3}} \right\}$$

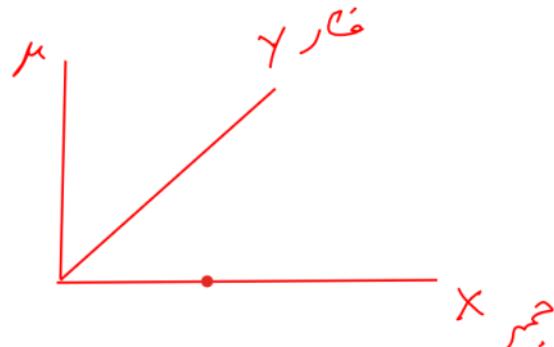
$$A \cap C = \left\{ \frac{0.4}{3} \right\} \bullet$$

$$A \cup B \cup C = \left\{ \frac{0.4}{3} + \frac{0.8}{4} + \frac{0.2}{5} \right\}$$

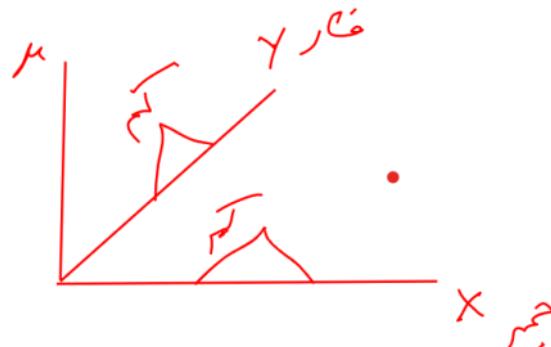
ولی اگر پیوسته باشند میتوانیم توی شکل نشون بدیم:



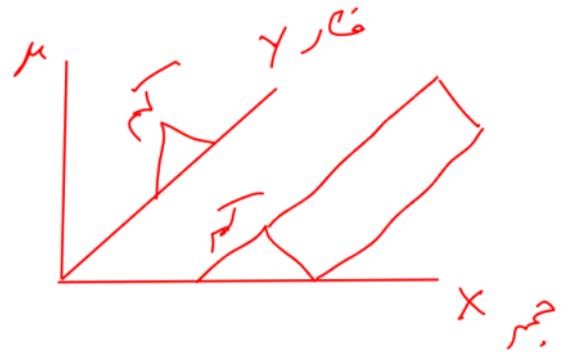
فرض کنیم که محور هامون 3 تا باشد.



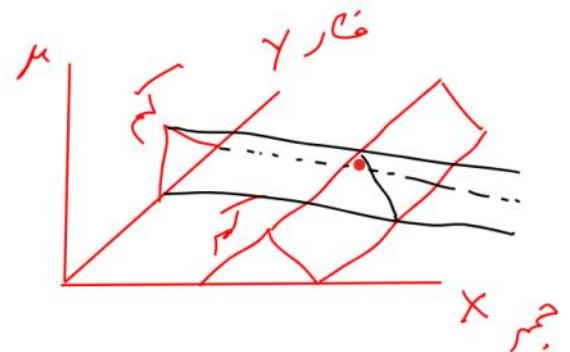
اگر فشار کم و حجم کم داشته باشیم:



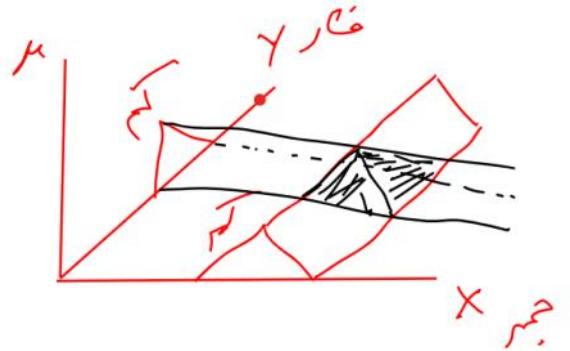
اینا هیچ اشتراک ندارند. ولی برای این که اشتراک داشته باشند، میتوانیم از عملگر توسعه استوانه ای استفاده کنیم. به این صورت که به مجموعه حجم کم مجموعه فشار را اضافه میکنیم و مجموعه رو دو بعدی میکنیم



اگر به فشار هم مجموعه مرجع حجم را اضافه کنیم با توسعه استوانه ای:



در این وسط یک هرم به وجود میاد که میشه اشتراک فشار کم و حجم کم.



--- شروع اسلاید ---

ضرب کارتزین رو میشه بر اساس توسعه استوانه ای تعریف کرد یا این که میشه مستقل تعریف کرد و توسعه استوانه ای رو بر حسب ضرب کارتزین تعریف کنیم. مثل داستان مرغ و تخم مرغه چون هیچکدام رو نمیشه اول گفت هر کدام یک دور دارند.

دور این که اول توسعه استوانه ای رو تعریف میکنیم بعد توسعه استوانه ای رو و بار بعدی برعکس

ضرب کارتزین در حالت کلاسیک به چه درد میخوره؟ اگر اعضای کلاس هوش محاسباتی رو با ویژگی اسم بیان کنیم و بگیم که حسن، علی، نقی، تقی، این چه مشکلی داره؟ ما ممکنه نتونیم همه افراد روز هم جدا کنیم. ممکنه 2 تا نقی و 2 تا تقی داشته باشیم. اگه با اسم و فامیل بایم شناسایی راحت تر میشه و در حالت کلی هرجی ویژگی بیشتر بشه شناسایی راحت تر.

مجموعه از تعدادی اشیا هستش. توی جلسه های اول سیب و پرقال رو با قطر و وزن نمایش دادیم. اگر قطر و وزن نتونن اینها رو خوب توصیف کنیم میشه یک ویژگی دیگه اضافه کرد که بهتر جدا کنیم.

اشیا در مجموعه ها میتونن بیشتر از یک ویژگی داشته باشند. ضرب کارتزین مجموعه مرجع چند بعدی رو ایجاد میکنه. اگر مجموعه مرجع اسامی رو داشته باشیم، مجموع مرجع فامیلا هم داشته باشیم. حاصل ضرب کارتزین اشون مجموع مرجع افراد رو با 2 ویژگی اسم و فامیل تولید میکنه.

البته در ضرب کارتزین ترتیب مهمه. $L \times W \times H$ دو مجموعه مرجع متفاوت میسازند

ضرب کارتزین

$$U = \{u_1, u_2\}$$

$$V = \{v_1, v_2, v_3\}$$

$$U \times V = \{(u, v) \mid u \in U, v \in V\}$$

$$U \times V = \{(u_1, v_1), (u_1, v_2), (u_1, v_3), (u_2, v_1), (u_2, v_2), (u_2, v_3)\}$$

$$\text{تعداد اعضاء} = N_U \cdot N_V$$

$$U_1 \times U_2 \times \dots \times U_n = \{(u_1, u_2, \dots, u_n) \mid u_1 \in U_1, \dots, u_n \in U_n\}$$

اگر مجموعه‌ها، n تا عضو و m تا عضو داشته باشند مجموعه حاصل از ضرب کارتزین $m \times n$ تا عضو دارد.

رابطه: یک تعریف شهودی و یک تعریف ریاضی دارد. در تعریف ریاضی می‌شود زیر مجموعه از یک حاصل ضرب کارتزین.

خود ضرب کارتزین یک مجموعه است ولی مجموعه چند بعدی.

در تعریف شهودی رابطه، فرض کیم مجموعه A, B زیر رو داریم:

$$A = \{\text{حسن}, \text{تحت}, \text{زن}\}$$

$$B = \{17, 20, 25\}$$

رابطه بین مجموعه اول و دوم خودش یک رابطه است. این که حسن 17 گرفته پس میتوانیم یک مجموعه دو بعدی در نظر بگیریم که حسن و 17 یک عضوش بشوند. وهمینطور داریم:

$$\{(25, \text{حسن}), (20, \text{تحت}), (17, \text{زن})\}$$

حاصل ضرب کارتزین همه حالت رو میدهد. ولی رابطه فقط اونایی که ما میخوایم رو میدهد.

چرا این میشه زیر مجموعه از ضرب کارتزین؟ چون ضرب کارتزین همه حالت رو تولید میکنه و رابطه ما نمیتوانه خارجش باشد و هر رابطه که تعریف کنیم داخلش هستش.

رابطه و ضرب کارتزین در ساده‌ترین حالت 2 تایی هستند. پس اگه بخوایم با تابع تعلق نشون بدیم باید تابع تعلق 2 بعدی باشد. یعنی مثلا:

$$\mu(u, y) = \begin{cases} 1 & (u, y) \in A \\ 0 & (u, y) \notin A \end{cases}$$

میشه اینطوری هم گفت:

$$xRy$$

$$xRy$$

وقتی میگیم $(x, y) \in A$ یعنی x با y رابطه داره پس xRy . بر عکس شدم که مشخصه.

در نمایش ماتریسی رابطه هم جاهایی که رابطه داریم 1 جاهایی که نداریم صفر مینداشیم:

نمایش ماتریسی رابطه

			V			
			v ₁	v ₂	v ₃	
			u ₁	1	1	0
U	u ₂			0	0	0

$$R_1(u, v) = \{(u_1, v_1), (u_1, v_2)\}$$

رابطه کامل و رابطه تھی

V			V				
	v ₁	v ₂	v ₃		v ₁	v ₂	v ₃
u ₁	1	1	1	u ₁	0	0	0
U	u ₂	1	1	U	u ₂	0	0
u ₃	1	1	1	u ₃	0	0	0

* رابطہ یک مجموعہ دو بعدیہ

رابطہ همسایگی

U={ایران, فرانسہ}

V={پاکستان, آلمان}

$$R_1(u, v) = \{(u_1, v_1), (u_2, v_2)\}$$

V		
	v ₁	v ₂
u ₁	1	0
U	u ₂	0

$$R_1(u, v) = \{(پاکستان, ایران), (آلمان, فرانسہ)\}$$

اگر بخوایم با notation جدید بنویسیم ہم:

$$R_1(u, v) = \{(\overline{u_1}, \overline{v_1}), (\overline{u_2}, \overline{v_2})\}$$

.

$$R_1(u, v) = \{(\overline{u_1}, \overline{v_1}), (\overline{u_2}, \overline{v_2}), (\overline{u_3}, \overline{v_3})\}$$

عملیات روی رابطہ ہا - اجتماع

$$R \cup S = \{U \mid U \in R \vee U \in S\}$$

$$\mu_{R \cup S}(U) = \mu_R(U) \vee \mu_S(U) = \text{Max}\{\mu_R(U), \mu_S(U)\}$$

عملیات روی رابطہ ہا - اشتراک

$$R \cap S = \{U \mid U \in R \wedge U \in S\}$$

$$\mu_{R \cap S}(U) = \mu_R(U) \wedge \mu_S(U) = \text{Min}\{\mu_R(U), \mu_S(U)\}$$

عملیات روی رابطه ها - متمم

$$\mu_{R'}(U) = 1 - \mu_R(U)$$

عملیات روی رابطه ها - زیرمجموعه

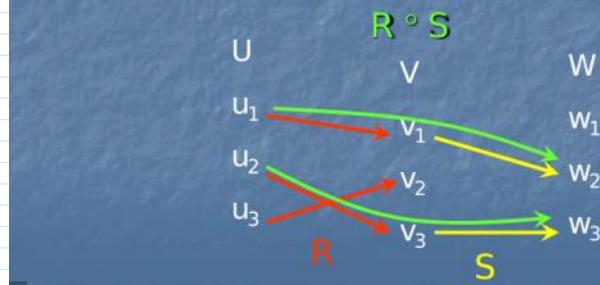
$$R \subseteq S \Rightarrow \mu_R(U) \leq \mu_S(U)$$

یک عملکرد دیگه هم داریم که ترکیبه که جلسه بعد در موردش حرف میزنیم.

عملیات روی رابطه ها - ترکیب

$$R \circ S = \cup (R \cap S) = \{(u_1, w_2), (u_2, w_3)\}$$

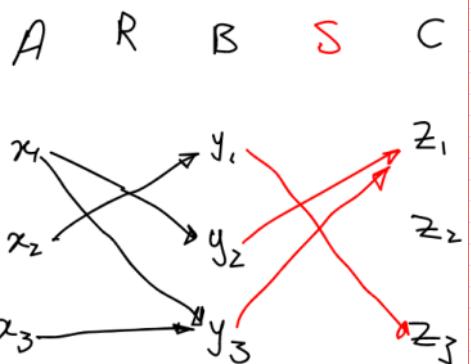
$$\mu_{R \circ S} = \vee (\mu_R \wedge \mu_S)$$



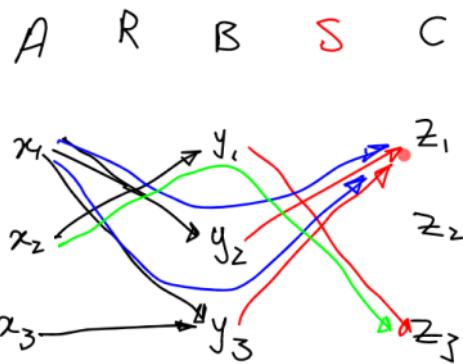
ترکیب: فرض کنیم مجموعه A و B و C داریم. به طوری که:

A	B	C
x_1	y_1	z_1
x_2	y_2	z_2
x_3	y_3	z_3

حالا فرض کنیم A و B با رابطه R و B و C هم با رابطه S در ارتباط هستند:



ترکیب به ما میگره رابطه بین A و C چیه. یعنی ما اگه رابطه A و B و همچنین رابطه B و C داشته باشیم ترکیب به ما میگره رابطه A و C چیه.



پس میتوان گفت:

$$(x_1, z_1) \in R^o S$$

$$(x_2, z_3) \in R^d S$$

باید حداقل یک مسیر وجود داشته باشد تا بتوئیم رابطه رو برقرار کنیم. اگر بخوبیم بر حسب تابع تعلق بیان کنیم میتوئیم بگیم:

$$\mu_{R^o S}(x, z_1) = (\mu_R(x_1, y_1) \wedge \mu_S(y_1, z_1)) \vee (\mu_R(x_1, y_2) \wedge \mu_S(y_2, z_1)) \vee (\mu_R(x_1, y_3) \wedge \mu_S(y_3, z_1))$$

چون y_1 در R رابطه ای نیست مقدارش 0 میشه ولی برای 2 تای دیگه برقراره 1 میشه.

میشه به نحو زیر هم نوشت:

$$\mu_{R^o S}(x, z) = \bigvee_{y \in Y} (\mu_R(x, y) \wedge \mu_S(y, z))$$

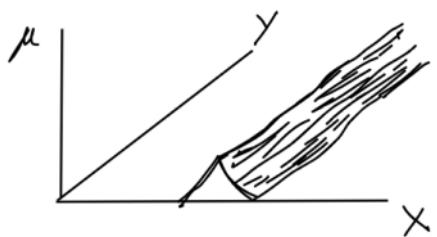
* در توسعه استوانه ای هدفمنون اضافه کردن یک بعد به یک مجموعه است و باعث میشه یک مجموعه یک بعدی رو به 2 بعدی تبدیل کنیم. کارش هم این بود که تعلق های یک مجموعه رو برای همه اعضای مجموعه دوم تکرار میکنه:

استوانه ای

$$X = \left\{ \frac{1}{a} + \frac{1}{b} + \frac{1}{c} \right\} \quad A = \left\{ \frac{0.4}{a} + \frac{0.3}{b} \right\}$$

$$Y = \left\{ \frac{1}{1} + \frac{1}{2} + \frac{1}{3} \right\}$$

$$A' = \left\{ \frac{0.4}{a,1} + \frac{0.4}{a,2} + \frac{0.4}{a,3} + \frac{0.3}{b,1} + \frac{0.3}{b,2} + \frac{0.3}{b,3} \right\}$$



حالا ما A' رو میتوئیم اینطوری تعریف کنیم:

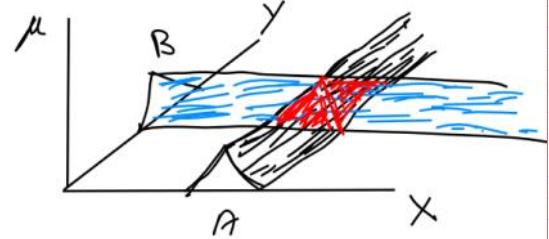
$$A' = A \times Y$$

و همون داستان میغ و تخم مرغ بین توسعه استوانه ای و حاصل ضرب کارتزین به وجود میاد.

$$\mu_{A'}(x, y) = \mu_A(x) \wedge \mu_B(y)$$

$$\mu_{A'}(x, y) = \mu_A(x)$$

از اشتراک این ۲ تا یک هرم به وجود می‌آید:



این هرم حاصل ضرب کارتزین A و B می‌شود.

حال اگر مجموعه A با مجموعه مرجع X و مجموعه B با مجموعه مرجع Y داشته باشیم:

$$\begin{array}{ll} A & X \\ B & Y \end{array}$$

$$\begin{array}{l} A' = A \times Y \rightarrow X \times Y \\ B' = X \times B \rightarrow X \times Y \end{array}$$

$$A' \cap B' = (A \times Y) \cap (X \times B) = A \times B$$

$$A \cap B = A \times B = \frac{B}{\text{تسویه اسوانه}} \text{ با توسعه اسوانه}$$

↓
لیسان بنانه

حال اگر مجموعه B و A داشته باشیم که هردو مرجع X رو داشته باشند. مجموعه C هم مرجع X و D هم مرجع Y داشته باشد داریم:

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x)$$

$$\begin{array}{ll} A & X \\ B & X \end{array}$$

$$\mu_{C \cap D}(x, y) = \mu_C(x) \wedge \mu_D(y)$$

$$\begin{array}{ll} C & X \\ D & Y \end{array}$$

میبینیم روابط بیکسان شد. چون این فرمول اگر ما یک مرجع بیکسان نیست متغیر هاش رو متفاوت فرض کنیم. خودش به طور اتوماتیک اون توسعه استوانه ای رو برای ما انجام میده و لازم نیست ما توسعه استوانه ای رو انجام بدیم.

مثال

$$\text{برهان} / \mu_{A \times B}(x, y) = \mu_A(x) \wedge \mu_B(y) \quad A = \left\{ \frac{0.2}{a_1}, \frac{0.2}{a_2}, \frac{0.3}{a_3} \right\} \quad X = \left\{ \frac{0.8}{b_1}, \frac{0.8}{b_2}, \frac{0.8}{b_3} \right\} \quad = 0$$

$$B = \left\{ \frac{0.7}{2}, \frac{0.3}{3} \right\} \quad Y = \left\{ \frac{1}{1}, \frac{1}{2}, \frac{1}{3} \right\}$$

ردیل اول

$$A \cap B = A \times B = \left\{ \frac{0.2}{a_1, 1}, \frac{0.2}{a_1, 2}, \frac{0.7}{a_2, 1}, \frac{0.7}{a_2, 2}, \frac{0.3}{a_3, 1}, \frac{0.3}{a_3, 2}, \frac{0.3}{a_3, 3} \right\}$$

ردیل دوم

$$A \times Y = \left(\frac{0.2}{a_1, 1} + \frac{0.2}{a_1, 2} + \frac{0.2}{a_1, 3} + \frac{0.8}{b_1, 1} + \frac{0.8}{b_1, 2} + \frac{0.8}{b_1, 3} \right) = \\ X \times B = \left(\frac{0.7}{a_2, 1} + \frac{0.7}{a_2, 2} + \frac{0.7}{a_2, 3} + \frac{0.3}{a_3, 1} + \frac{0.3}{a_3, 2} + \frac{0.3}{a_3, 3} \right) = \\ A \times B = (A \times Y) \cap (X \times B) = \left\{ \frac{0.2}{a_1, 1}, \frac{0.2}{a_1, 2}, \frac{0.7}{a_2, 1}, \frac{0.7}{a_2, 2}, \frac{0.3}{a_3, 1}, \frac{0.3}{a_3, 2}, \frac{0.3}{a_3, 3} \right\}$$

-- برگشت به اسلامی

ترکیب هارو به صورت ضرب ماتریس هم میشه در نظر گرفت:

روابط					Max	Min		
	F	v ₁	v ₂	v ₃	P	w ₁	w ₂	w ₃
D	R	C	T	E	S	.1	1	10
u ₁	.1	0	0	1	v ₁	1	0	0
u ₂	1	0	0	1	v ₂	0	1	1
u ₃	10	1	1	0	v ₃	0	0	1

	P	w ₁	w ₂	w ₃
D	R * S	.1	1	10
u ₁	.1	0	0	1
u ₂	1	0	0	1
u ₃	10	1	1	1

نکته که ازش موجوده اینه که بین 2 روابط باید یک مجموعه مشترک که این جا ها هستن موجود باشه و گرنه نمیشه ترکیب کرد.

روابط

عملیات روی روابط - خواص

جابجایی

$$R \cup S = S \cup R$$

$$R \cap S = S \cap R$$

$$R \cup (S \cup T) = (R \cup S) \cup T \quad \text{شرکت پذیری}$$

$$R \cap (S \cap T) = (R \cap S) \cap T$$

$$R \cup (S \cap T) = (R \cup S) \cap (R \cup T) \quad \text{توزیع پذیری}$$

$$R \cup (S \cap T) = (R \cap S) \cup (R \cap T)$$

$$(R \cup S)' = R' \cap S' \quad \text{دمرگان}$$

$$(R \cap S)' = R' \cup S'$$

روابط

ضرب کارتزین فازی

$$U = \{\mu_{u1}/u_1, \mu_{u2}/u_2\}$$

$$V = \{\mu_{v1}/v_1, \mu_{v2}/v_2\} \quad \mu_{U \times V} = \mu_U \wedge \mu_V$$

$$U \times V = \{\mu_{uv}/(u, v) \mid \mu_{uv} = \text{Min}(\mu_u, \mu_v), u \in U, v \in V\}$$

$$U \times V = \{\mu_{11}/(u_1, v_1), \mu_{12}/(u_1, v_2), \mu_{21}/(u_2, v_1), \mu_{22}/(u_2, v_2)\}$$

$$\text{تعداد اعضاء} = N_U \cdot N_V$$

$$U_1 \times U_2 \times \dots \times U_n = \{\mu_{u1, \dots, un}/(u_1, u_2, \dots, u_n) \mid$$

$$\mu_{u1, \dots, un} = \text{Min}(\mu_{u1}, \dots, \mu_{un}), u_1 \in U_1, \dots, u_n \in U_n\}$$

ضرب کارتزین فازی میشه اشتراک توسعه استوانه ای ها. این اشتراک توسعه استوانه ای ها معادل با این هست که بگیم زوج های که زوج اول برای مجموعه اول و زوج دوم برای مجموعه دوم و تعلقش اشتراک بین تعاقب ها باشه.

روابط

رابطه بین مجموعه های فازی

رابطه بین مجموعه های فازی زیرمجموعه ای از ضرب کارتزین فازی آن مجموعه ها است

$$R(u_1, u_2, \dots, u_n) \subset U_1 \times U_2 \times \dots \times U_n$$

$$\mu_R \leq \mu_{U_1 \times U_2 \times \dots \times U_n}$$

$$U \times V = \{\mu_{11}/(u_1, v_1), \mu_{12}/(u_1, v_2), \mu_{21}/(u_2, v_1), \mu_{22}/(u_2, v_2)\}$$

$$R = \{\mu_{R11}/(u_1, v_1), \mu_{R12}/(u_1, v_2), \mu_{R21}/(u_2, v_1), \mu_{R22}/(u_2, v_2)\}$$

$$\mu_{Rij} \leq \mu_{ij}$$

روابط

رابطه فازی دوری

$$U = \{\text{ایران}, \text{فرانسه}\}$$

$$V = \{\text{پاکستان}, \text{آلمان}\}$$

روابط فازی را می توان هم برای متغیرهای زبانی (**دوری**) و هم برای متغیرهای

دقیق (**فاصله**) بکار برد

	U	u₁	u₂
V	دوری	فرانسه	ایران
v₁	آلمان	.1	.8
v₂	پاکستان	.9	.2
	ان		

اگه بیشترین فاصله مرجع رو در نظر بگیریم و فاصله رو تقسیم بر اون بکنیم دوری بدست میاد.

روابط

عملیات روی رابطه های فازی - اجتماع

$$R \cup S = \{U \mid U \in R \vee U \in S\}$$

$$\mu_{R \cup S}(U) = \mu_R(U) \vee \mu_S(U) = \text{Max}\{\mu_R(U), \mu_S(U)\}$$

عملیات روی رابطه های فازی - اشتراک

$$R \cap S = \{U \mid U \in R \wedge U \in S\}$$

$$\mu_{R \cap S}(U) = \mu_R(U) \wedge \mu_S(U) = \text{Min}\{\mu_R(U), \mu_S(U)\}$$

روابط

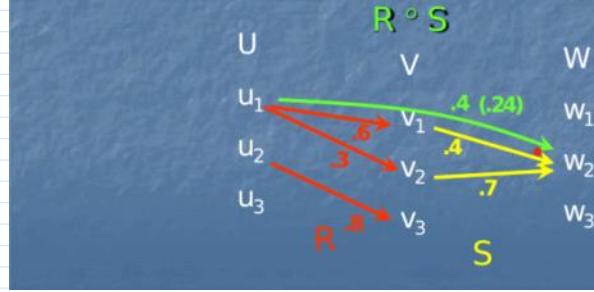
عملیات روی رابطه های فازی - ترکیب

$$R \circ S = \cup (R \cap S)$$

$$\mu_{R \circ S}(u, w) = \vee_{v \in V} (\mu_R(u, v) \wedge \mu_S(v, w))$$

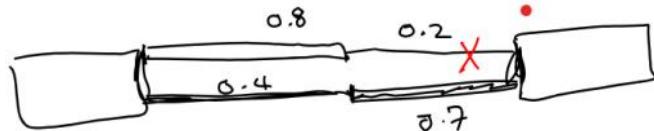
$$\mu_{R \circ S} = \text{Max}(\text{Min}(\mu_R, \mu_S))$$

$$\mu_{R \circ S} = \text{Max}(\mu_R * \mu_S)$$



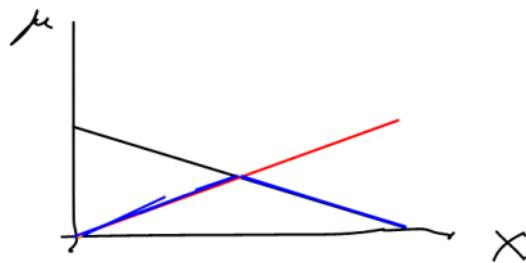
در حالت فازی خیلی وقتاً به جای Max dot Max Min از استفاده میشه. که مزیتش اینه که dot مشتق پذیره ولی min مشتق پذیر نیست.
اگر از MaxMin برمی بگیریم میشه 0.3 و 0.7 مینیمم بگیریم میشه 0.3 . بین 0.4 و 0.6 هم میشه 0.4 . در آخر 0.4 = max(0.4, 0.3) = 0.4
اگر از Max dot مسیر بالای میشه 0.24 مسیر پایینی میشه 0.21 که اشون 0.24

فرض شود ماشین خراب شده و میخوایم بوکسور کنیم. یک سری طناب کوتاه با قطر های متفاوت داریم. اگر قطر هارو نرمال کنیم فرض کنیم یکی 0.8 باشه یکی 0.4 و یکی دیگه 0.7 . فرض کنیم 2 ماشین رو مطابق به شکل زیر وصل میکنیم:

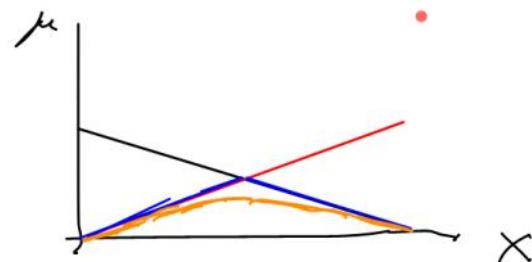


اولین طنابی که پاره میشه مسلماً همین 0.2 عه . بعدش 0.4 که پاره شد ارتباط از بین میره. پس تا وقتی که بین 2 ماشین برقرار است. این میشه توجیه فیزیکی Max Min . یعنی تا وقتی که Max Min این طناب ها یعنی 0.4 پاره نشده ارتباط برقراره.

وقتی از مینیمم برای اشتراک استفاده کنیم شبیه شکل آبیه به دست میاد که مشتق پذیر نیست:



ولی وقتی از ضرب استفاده میکنیم همچنان منحی حاصل میشے که هموار تر و مشتق پذیر میشے:



* نزدیک ترین اشتراک به مینیمم همین ضرب هست لذا تقریب خوبی برای مینیمم هستش با توجه به این که مشتق پذیرم هست.

وقتی از مینیمم برای اشتراک استفاده میکنیم هیچ تفاوتی قابل نمیشه بزرگه چیه:

$$0.2 \wedge 0.8 = 0.2$$

$$0.2 \wedge 0.5 = 0.2$$

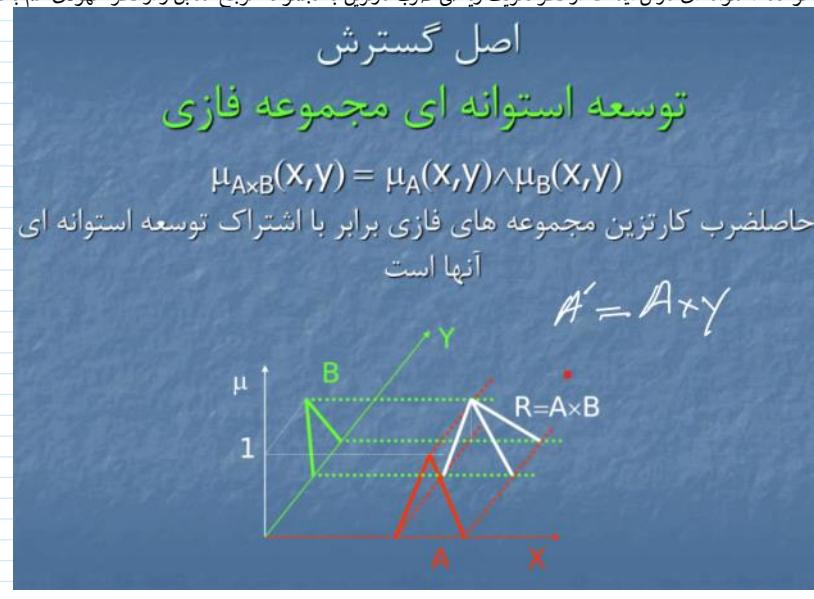
ولی وقتی از ضرب استفاده کنیم تفاوتش قابل توجه است:

$$0.2 \wedge 0.8 = 0.16$$

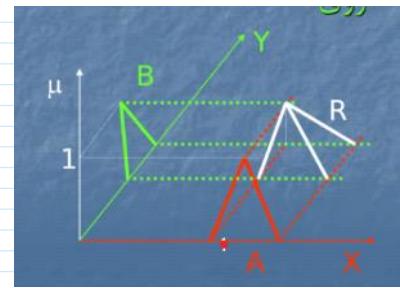
$$0.2 \wedge 0.5 = 0.1$$

--اسلاید:

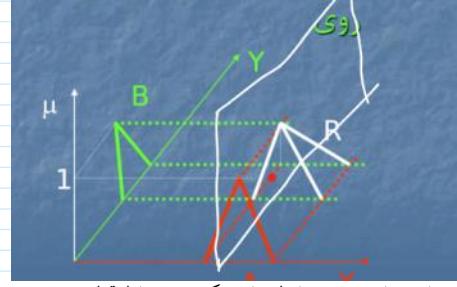
توسعه استوانه ای کارش اینه که از نظر تعریف ریاضی ضرب کارتزین با مجموعه مرجع مقابله و از نظر شهودی هم با تکرار مقدار تعلق در بعد جدید یک مقدار اضافه میکرد:



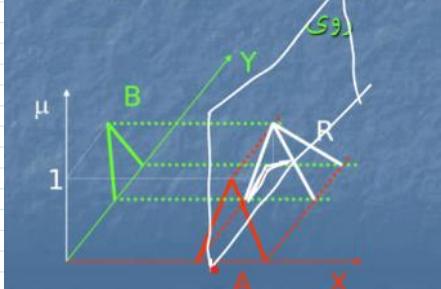
حالا چطوری بعد روکم کنیم؟ میتونیم تصویر کنیم. مثل این که سایه خود رو روی دیوار بینیم بسته به این که در کدام جهته تصویر میشے روی اون بعد. از لحاظ ریاضی فرض شود بخواهیم در یک نقطه مثل نقطه زیر در محور X تصویر رو بدست بیاریم:



میایم یک صفحه موازی با محور μ و Y رسم میکنیم:



با رسم این صفحه رابطه ما در یک سری نقاط قطع میشود:



ما برای این نقاط یک سری مقادیر تعلق داریم. چون مقدارشون بیش از پدونس ما مقدار تعلق رو در این نقطه برابر با اجتماع میگیریم و اگر اجتماع رو ماکس بگیریم میشه Max این مقدار های تعلق.

اگر رابطه R داشته باشیم تصویرش روی X میشه:

$$R_x = \bigcup_{y \in Y} R$$

$$\mu(x) = \sqrt{\mu(x,y)}$$

$$R_x \quad y \in Y$$

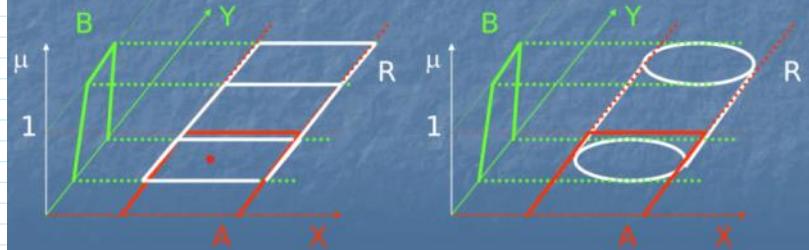
* همونطور که در گرافیک اگر مثلاً یک مکعب داشته باشیم که از وسطش یک استوانه حذف شده باشد تصویر کنیم روی محور x و y میشه دو تا مربع و نمیشه از تصویرشون شکل اصلی رو ساخت (چون دیگه حفره معلوم نیست)، در رابطه ها هم زمان هایی که بین X و Y همبستگی موجود باشد نمیتوان از تصویر رابطه را بدست آورد و تنها زمانی میشه روابط رو بدست آورد که همبستگی داشته باشند.

اصل گسترش

مجموعه فازی جدا پذیر یا نفوذ ناپذیر

$$R = \Pr_X(R) \times \Pr_Y(R) = A \times B$$

اگر مولفه ها (x, y) نسبت به هم وابستگی نداشته باشند R می تواند منحصراً توسط تصاویرش بازارسازی شود



در شکل بالا اگر استوانه عمودی رو روی X و Y محور کنیم هردو مستطیل میشود و با توسعه استوانه ای به مکعب میرسیم که با شکل اصلی متفاوت است. در این گونه موارد میگیم رابطه اولیه **جداپذیر نیست** ولی اگر بتوانیم رابطه اولیه رو بسازیم میگیم **جداپذیر است**. طبق تعریف استاد داریم:

در صورتیکه هر آن که رابطه را از جمله اشتراک توسعه استوانه ای داشته باز سازی میکند آن رابطه حبایزی نیست

که به جای اشتراک توسعه استوانه ای ضرب کارتزین هم میشه گفت چون بکین.

$$R \quad R_x \quad R_y$$

$$R = R_x \times R_y \quad \text{حبایزی}$$

پس الان استوانه مثلا میشه جدانپذیر ولی مثلا مکعب یا هرم میشه جداپذیر.

* دقیق شود ما الان در مورد مجموعه فازی مرتبه اول داریم صحبت میکنیم و حق نداریم یک مکعب یا استوانه مایل داشته باشیم چون اونموقع برای یک متغیر 2 تا مقدار تعلق داره که میشه مجموعه فازی مرتبه دوم.

$$\begin{array}{ccccc} & & & & \text{خاصیت} \\ A & B & & & x, y \\ \mu_A^{(x)} & \mu_B^{(y)} & & & x^2 + y^2 = R \\ & & & & x^2 + y^2 = R \\ & & \mu_R^{(x,y)} = \mu_A^{(x)} \wedge \mu_B^{(y)} & & \end{array}$$

ولی اگر وابسته نباشد:

$$\begin{array}{ccc} & \mu_R^{(x,y)} = \mu_A^{(x)} \wedge \mu_B^{(y)} & R_x = A \\ & \text{حبایزی} & R_y = B \end{array}$$

ولی در مواقعی که x, y وابسته اس دیگه $R_x = A$ یا $R_y = B$ یا هردو.

مشابه اش رو توی احتمالات هم داشتیم:

$$\begin{aligned}
 p(x, y) &= p(x) p(y|x) = p(x) p(y) \quad \text{متصل} \\
 \mu_R(x) &= \bigvee_{y \in R} \mu_{(x,y)} \quad \left\{ \begin{array}{l} p(x) = \int p(x, y) dy \\ p(y) = \int p(x, y) dx \end{array} \right. \equiv R_x = \bigcup_{y \in R} R \\
 \mu_R(y) &= \bigvee_{x \in R} \mu_{(x,y)} \quad \equiv R_y = \bigcup_{x \in R} R
 \end{aligned}$$

$R = R_x \times R_y$ جدول

$$\mu_R(x, y) = \mu_R(x) \wedge \mu_R(y)$$

در احتمالات هم تنها اگر مستقل باشن میشه از ضربشون به اصلیه رسید.

اصل گسترش

تصاویر مجموعه فازی

تصویر R را روی U رسم کنید

$$\mu_U(U) = \vee_{v \in V} (\mu_R(U, v)) = \text{Max}_{v \in V} (\mu_R(U, v))$$

$$\mu_V(V) = \vee_{u \in U} (\mu_R(u, V)) = \text{Max}_{u \in U} (\mu_R(u, V))$$

		v_1	v_2	v_3
	R	.4	.8	1
u_1	.7	.2	.7	.5
u_2	.9	.3	.8	.9
u_3	1	.4	.1	1

همونطور که در جدول میبینیم وقی خواستیم تصویر بگیریم برای هرکدام ماکس گرفتیم در جهت اون فلش ها

اصل گسترش

توسعه استوانه ای مجموعه فازی

حاصلضرب کارتزین برایر با اشتراک توسعه استوانه‌ای است



	V_1	V_2	V_3				
	.4	.8	1				
U_1	.7	.7	.4	.7	.8	.7	1
U_2	.9	.9	.4	.9	.8	.9	1
U_3	1	1	.4	1	.8	1	1

	v_1	v_2	v_3	
R	.4	.8	1	
u_1	.7	.4	.7	.7
u_2	.9	.4	.8	.9
u_3	1	.4	.8	1

با توجه به این که رایطه به دست او مده (چالو راستی) یا رایطه اصلی (اسلادید قلی) یکی نیست \leftrightarrow جدایزیر نیست.

اصل گسترش

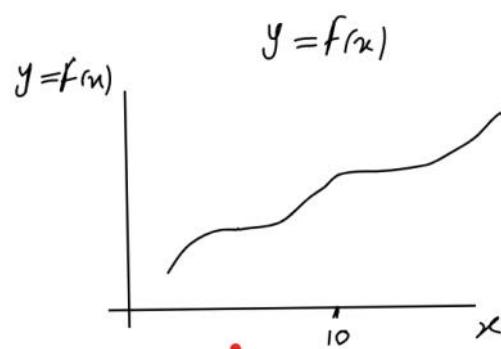
توسعه استوانه‌ای مجموعه فازی

اگر یک مجموعه فازی توسط توسعه استوانه‌ای تصاویرش قابل بازسازی باشد مجموعه جدایزیر می‌باشد

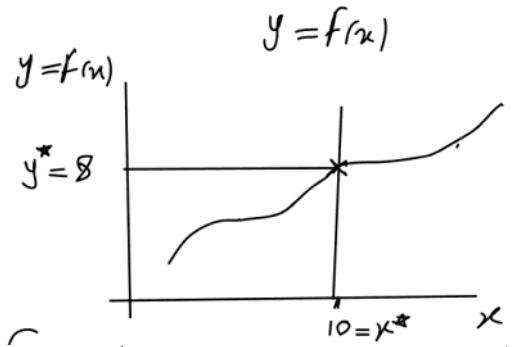
		جدا پذیر					جدا ناپذیر		
		v ₁	v ₂	v ₃			v ₁	v ₂	v ₃
	R	.4	.8	1		R	.4	.8	1
u ₁	.7	.4	.7	.7	u ₁	.7	.2	.7	.5
u ₂	.9	.4	.8	.9	u ₂	.9	.3	.8	.9
u ₃	1	.4	.8	1	u ₃	1	.4	.1	1

در جدول راسته، در ستون v_1 ماکسیمم اش، شد 0.4 و در سطر u_1 ماکسیمم اش، مبیشه 0.7 و اشتراک اینها مبیشه 0.4 که با 0.2 برابر نیست بهم، جداگذیر نیست.

فرض، شود $(x=f(y))$ دارایم. حالا ما به صورت جیری (واطشعو نداریم و) به صورت هندسه، همچنین منحجه، داره:

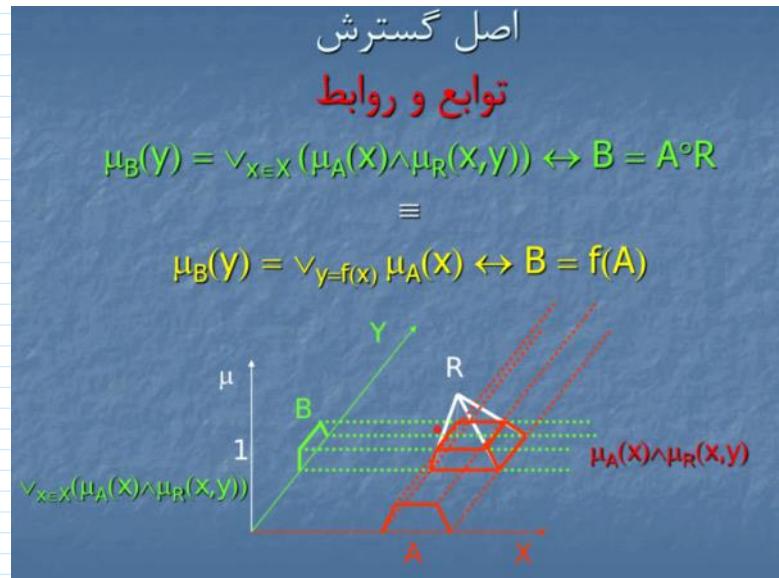


اگه بخوايم بيسيم در $x=10$ بسيئم مقدارش چي ميشه يايد از $10=x$ يك خط عمود رسم کنيم تلاقیش رو بيسيم چي ميشه:

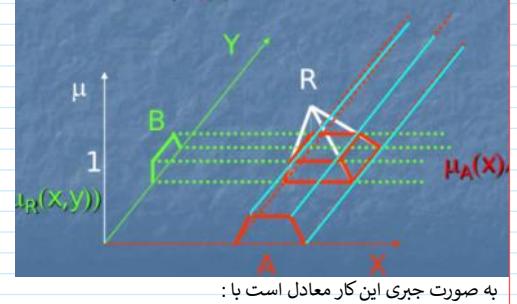


حالا اگر بخوایم با توسعه استوانه ای و تصویر کردن بخوابیم بگیم. مثل این میمونه که بگیم:

1. توسعه استوانه ای x^* را روی محور y بدهست می آوریم.
2. اشتراک توسعه استوانه ای بدهست آمده در مرحله قبل با منحنه (رابطه) بدهست می آوریم.
3. تصویر اشتراک بدهست آمده در مرحله قبل را روی محور y بدهست می آوریم.



در شکل فوق، هم رابطه R سفیده رو داریم. یک مجموعه فازی که ذوزنقه ای طور هم هست رو داریم.
اول A رو توسعه استوانه ای میدیم:



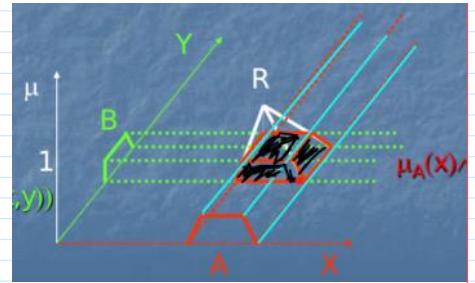
به صورت جری این کار معادل است با :

$A \times Y$

$$\mu_A(x, y)$$

$$y \in Y$$

در مرحله بعد اشتراک باید حساب بشه. اشتراک این توسعه استوانه ای با رابطه میشه این تیکه سیاه رنگ که مشخص شده:



که بر اساس جبری داریم:

$$(A \times Y) \cap R$$

$$\bigcup_{y \in Y} \mu_A^{(x_1, y)} \wedge \mu_R^{(x_1, y)}$$

حالا تصویر این اشتراک رو روی محور Y بدهست مباریم. یعنی:

$$B = \bigcup_{x \in X} (A \times Y) \cap R$$

$$\mu_B(y) = \bigvee_{x \in X} \mu_A^{(x_1, y)} \wedge \mu_R^{(x_1, y)}$$

- آنر را به R و مجموعه فائز A را داشته باشیم و خروجی B را پس از اینها حساب کنیم
- توسعه استرانه A x Y
- تعریف ترسته استرانه ای را بگیری
- نتیجه اشتراک روی مجموعه R

که این کار معادل ترکیب A و R می باشد

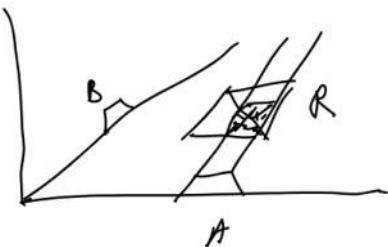
$$B = \bigcup_{x \in X} (A \times Y) \cap R = A^o R \bullet \text{ ترکیب}$$

A B R
ویرایش خروجی رابطه بین مجموعه ها

A R	$B = A^o R$
B R	$A = R^o B$
A B	$R = A \times B$

دقت شود R در آخرین خط نوشته شده تنها یکی از روابط می باشد.

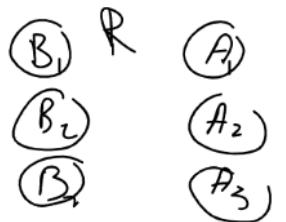
در جلسه پیش دیدیم که:



$$\bullet \quad A \circ R = B$$

$$R \circ B = A$$

اگر یک سری مجموعه فازی داشته باشیم مثلا فرض شود اگر هوا سرد باشد (A_1) آنگاه باید باید لباس گرم پوشید (B_1). بعد یک سری دیگه قاعده هم داشته باشیم در جلوتر یاد میگیریم چطوری ازشون رابطه رو بسازیم:



اگر R بحسب بیاریم انگار یک شبکه عصبی به دست آوردم که با هر ورودی میتوانیم خروجی مورد نظر را بدست بیاریم.

اصل گسترش قضیه استدلال تقریبی

فرض:

$$B = A \circ R$$

انتظار داریم که :

$$\text{if } A = A_1 \text{ & } B_1 = A_1 \circ R \text{ then } B_1 = B$$

واقعیت: فقط در صورتی که R جدایذیر باشد B بحسب می آید

یک داستان داریم که یک نگران کننده. مثلا برای هوای سرد (A) باید لباس گرم پوشید (B) فرض کنیم یک رابطه R بحسب بیاریم. برای برخی قانون ها R میشه $A \times B$. انتظار ما اینه وقتی A رو با R ترکیب کنیم B بشه ولی همیشه اینطوری نیست. تنها در صورت ممکنه که R جدایذیر باشه. در غیر این صورت یک B_1 ای میشه مثلا. اما این B_1 یک تقریب خیلی خوبی از B است.

A B

(R)

$$R = A \times B$$

$$A^o R \neq B$$

$A^o R = B$ حبیب این را

$$A^o R = B_1$$

میتوان اثبات کرد اگر A_1 را با R ترکیب کنیم و B_1 بدست بیاد. هرجی A_1 به A نزدیک تر بشه B_1 هم به B نزدیک میشه. اگر A_1 برابر با A باشه B_1 تقریباً برابر با B میشه و با هیچ روش دیگری تقریب به این خوبی نمیشه بدست آورد.

$$A_1^o R = B_1$$

$$A_1^o R = B_1$$

$$A_1 \rightarrow A$$

$$B_1 \rightarrow B$$

$$A_1 = A \quad B_1 \approx B$$

همین حرفای استاد رو اگه در اسلاید ببینیم:

اصل گسترش قضیه استدلال تقریبی

می توان اثبات کرد: اگر R جدایزیر هم نباشد، هرچه ورودی به A نزدیکتر شود، خروجی هم به B نزدیکتر خواهد شد

If $A_1 \rightarrow A$ then $B_1 \rightarrow B$

نتیجه: ترکیب ورودی و R در هر حالت بهترین تقریب ممکن را ارائه خواهد کرد

اصل گسترش

تعمیم برای توابع چند متغیره

$$\mu_B(y) = \vee (\mu_{A_1 \times \dots \times A_n}(x) \wedge \mu_R(x_1, \dots, x_n, y)) \leftrightarrow B = (A_1 \times \dots \times A_n)^o R$$

$$\mu_B(y) = \vee_{y=f(x_1, \dots, x_n)} \mu_{A_1 \times \dots \times A_n}(x) \leftrightarrow B = f(A_1 \times \dots \times A_n)$$

$$A_1 = \{0.2/1, 1/2, 0.1/3\}$$

$$A_2 = \{0.1/4, 1/5, 2/6\}$$

$$y = x^2 \quad B = ?$$

$$A_1 \times A_2 = \{0.1/(1,4), 0.2/(1,5), 0.2/(1,6), 0.1/(2,4), 0.1/(2,5), 0.2/(2,6), 0.1/(3,4), 0.1/(3,5), 0.1/(3,6)\}$$

$$y = x_1 + x_2 \quad B = ?$$

حالا بخواهیم یک مثال بزنیم ببینیم چطور میشه استفاده کرد. فرض شود ما یک $y = f(x) = x^2 - 2x + 2$ داریم. اگر x برابر با مجموعه فازی A باشد که :

$$y = f(x) = x^2 - 2x + 2$$

$$x = A = \left\{ \frac{0.2}{1} + \frac{1}{z} + \frac{0.7}{3} \right\} \bullet$$

$$B = f(A)$$

$$A^o R = B$$

حالا R رو چطوری بدست بیاریم؟ میتوانیم تابع رو به صورت یک رابطه بنویسیم. با توجه به این که A در 1 و 2 و 3 هستش میریم جایگذاری میکنیم در $f(x) = x^2 - 2x + 2$ بدست میاد و داریم:

$$y = f(x) = x^2 - 2x + 2$$

$$x = A = \left\{ \frac{0.2}{1} + \frac{1}{z} + \frac{0.7}{3} \right\}$$

$$B = f(A)$$

$$A^o R = B$$

$$z \begin{bmatrix} 1 & z & 5 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

و B برابر است با:

$$\begin{bmatrix} 0.2 & 1 & 0.7 \end{bmatrix} \begin{bmatrix} 0.2 & 0 & 0 \\ 0.7 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} =$$

$$B = \left\{ \frac{0.2}{1} + \frac{1}{z} + \frac{0.7}{5} \right\}$$

حالا اگر بخواهیم مساله رو یکم پیچیده تر بشکنیم، میتوانیم بگیم:

$$f(x_1, x_2) = x_1^2 x_2 - x_1 x_2^3 + x_1 - x_2$$

$$x_1 = A_1 = \left\{ \begin{array}{l} \frac{0.2}{-1} + \frac{0.4}{0} + \frac{0.5}{1} \end{array} \right\} \quad x_2 = A_2 = \left\{ \begin{array}{l} \frac{0.8}{0} + \frac{0.7}{1} \end{array} \right\}$$

$$y = f(A_1, A_2)$$

برای راه حل اول باید A رو بحسبت بیاریم:

$$f(x_1, x_2) = x_1^2 x_2 - x_1 x_2^3 + x_1 - x_2$$

$$x_1 = A_1 = \left\{ \begin{array}{l} \frac{0.2}{-1} + \frac{0.4}{0} + \frac{0.5}{1} \end{array} \right\} \quad x_2 = A_2 = \left\{ \begin{array}{l} \frac{0.8}{0} + \frac{0.7}{1} \end{array} \right\}$$

$$B = y = f(A_1, A_2) \quad B = A^0 R \quad A = A_1 \times A_2$$

$$A = A_1 \times A_2 = \left\{ \begin{array}{l} \frac{0.2}{-1,0} + \frac{0.2}{-1,1} + \frac{0.8}{0,0} + \frac{0.7}{0,1} + \frac{0.5}{1,0} + \frac{0.5}{1,1} \end{array} \right\}$$

چون میخوایم به روش ماتریسی حل کنیم، رابطه ای باید بسازیم که ستون های A ما با سطر های R ما هم جنس باشند.

$$f(x_1, x_2) = x_1^2 x_2 - x_1 x_2^3 + x_1 - x_2$$

$$\left\{ \begin{array}{l} \frac{0.2}{-1,0} + \frac{0.7}{-1,1} + \frac{0.8}{0,0} + \frac{0.3}{0,1} + \frac{0.5}{1,0} + \frac{0.5}{1,1} \end{array} \right\}$$

$$A = A_1 \times A_2$$

	-1	0	1
-1,0	1	0	0
-1,1	0	1	0
0,0	0	1	C
0,1	1	0	0
1,0	0	0	1
1,1	0	1	0

$$B = A^0 R$$

$$R$$

$$f(x_1, x_2) = x_1^2 x_2 - x_1 x_2^3 + x_1 - x_2$$

$$\left\{ \begin{array}{l} \frac{0.2}{-1,0} + \frac{0.7}{-1,1} + \frac{0.8}{0,0} + \frac{0.3}{0,1} + \frac{0.5}{1,0} + \frac{0.5}{1,1} \end{array} \right\}$$

$$A = A_1 \times A_2$$

-1	0	1
0,0	0,0	0,0
0,1	1,0	0,0
1,0	0,0	0,0
1,1	0,0	1,0

$$B = \left\{ \frac{0.7}{-1}, \frac{0.8}{0}, \frac{0.5}{1} \right\}$$

$$B = A^\circ R$$

=====
در حالت کلی داریم:

$$B = A^\circ R \rightarrow R \quad A_1 \times A_2 \times \dots \times A_n \quad R$$

$$B_1, B_2, \dots, B_m$$

ولی مشکل اینجا س اگر ابعاد خیلی بزرگ باشه R زیاد میشه. مثلا اگر هر کدوم از A_i ها و B_i ها 10 تا داشته باشد رابطه $R = 10^n \times 10^m$ تا عضو دارد.

اگر پیوسته باشه دیگه از این روش نمیشه حل کرد. در حالت پیوسته 2 حالت داریم:

- ورودی خروجی گسسته بشه: میشه همین که تا الان بررسی کردیم. ولی این هزینه محاسباتی بسیار زیادی دارد
- مقدار تعلق گسسته بشه: هزینه محاسباتی خیلی میاد پایین. بهش میگن برش lambda که بعده تو این درس در موردش صحبت بشه

منطق:

موضوع صحبت منطق، راجع به درستی و نادرستی گزاره ها می باشد. اگه درست باشد 1 و غیر این صورت 0 است.
مثال در یک گزاره ساده میگویی "حسن برادر علی است".

$$T(P) = \begin{cases} 1 & \text{درست} \\ 0 & \text{نادرست} \end{cases}$$

P : حسن برادر علی است

برای هر گزاره میتوان یک مجموعه و یک عضو تعريف کرد و گزاره رو تحت عنوان تعلق داشتن یک عضو به یک مجموعه بیان کرد.

P : حسن برادر علی است

P : $a \in A$

A = برادران علی

a = حسن

$$\tau(p) = \mu_A(a) = \begin{cases} 1 & a \in A \\ 0 & a \notin A \end{cases}$$

پس منطق حرف جدیدی نمیزند و منطق خودش همون مجموعه است.

گزاره های منطقی منطق کلاسیک

$P: x \text{ is in } A$ $Q: y \text{ is in } B$

If $x \in A$ then $T(P)=1$ else $T(P)=0$

If $y \in B$ then $T(Q)=1$ else $T(Q)=0$

$P \vee Q : x \in A \text{ or } y \in B \equiv T(P \vee Q) = \text{Max}(T(P), T(Q))$

$P \wedge Q : x \in A \text{ and } y \in B \equiv T(P \wedge Q) = \text{Min}(T(P), T(Q))$

$P' : x \notin A \equiv T(P') = 0$

$P \rightarrow Q : x \notin A \text{ or } y \in B \equiv T(P \rightarrow Q) = T(P' \vee Q)$

$P \leftrightarrow Q : \equiv T(P \leftrightarrow Q) = 1 \text{ if } T(P)=T(Q) \text{ else } T(P \leftrightarrow Q) = 0$

دیدم یک گزاره به صورت تعلق یک عضو به یک مجموعه است که درست بودنش وقتیه که عضو هست

قانون های منطق با شرطی هستن یا دو شرطی. شرطی رو بخوایم تو مجموعه بگیم:

$$\begin{array}{l} P \Rightarrow Q \\ A \Rightarrow B \end{array}$$

که همین با اجتماع بخوایم بگیم میشه:

$$A' \cup B$$

و دو شرطی هم بخوایم بگیم:

$$P \Leftrightarrow Q \quad (A' \cup B) \wedge (A \cup B')$$

گزاره های منطقی

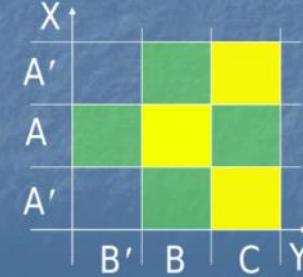
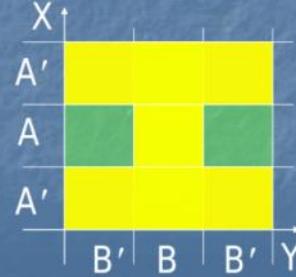
رابطه قانون شرطی کلاسیک

$P \rightarrow Q \equiv \text{If } P: x \in A \text{ Then } Q: y \in B \equiv A' \cup B$

$R = (A \times B) \cup (A' \times Y) \equiv \text{If } A \text{ Then } B$

$R = (A \times B) \cup (A' \times C) \equiv \text{If } A \text{ Then } B \text{ Else } C \equiv$

If A Then B And If A' Then C



در واقع P آنگاه Q یک رابطه است. (چرا؟)

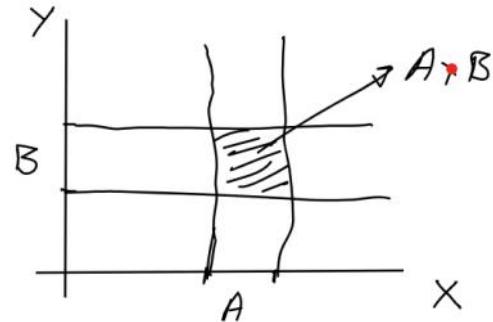
به عبارت دیگه وقتی میگیم A آنگاه B همان:

$$A \Rightarrow B \equiv A' \cup B$$

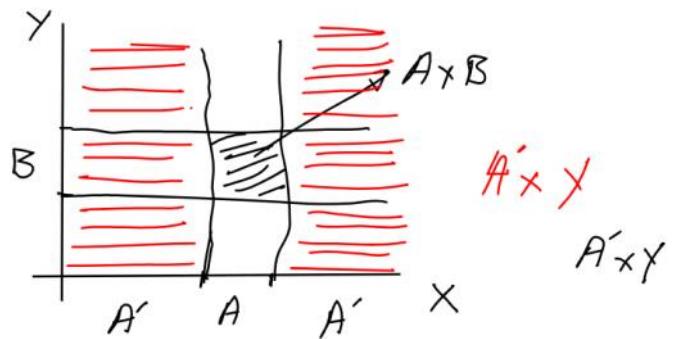
هستش. پس چرا یک رابطه درست میشه؟ چون مجموعه مرجع A و B متفاوت است. اگر 2 مجموعه داشته باشیم با هم اشتراک بگیریم اگه مجموعه مرجع یک باشه بعد تغییر نمیکنه ول اگه مجموعه مرجع اشون متفاوت باشه چون هم مرجع میکنیم ابعاد اضافه میکنیم و یک رابطه میشه و رابطه هم زیر مجموعه از حاصل ضرب کارتزین بود.

$$A' \cup B = (A' \times Y) \cup (X \times B)$$

یک از روش های بدست آوردن رابطه بر اساس توجیهات است که از یک سری قانون میکنیم، مثلاً اگر قانون "اگر هوا سرد باشد آنگاه لباس گرم بپوشیم"، اگر بهش تحت عنوان منطق ارسسطوی نگاه کنیم معنیش میشه که با هوا سرد است یا نیست. اگر باشه توجیه قانون اینه که لباس گرم بپوشیم. یعنی اگر A برقار باشه توسعه استوانه ای میدیم، B هم که لباس گرم مپوشیم رو توسعه استوانه ای میدیم و اشتراکشون میشه:



حالا اگر هوا سرد نبود چیکار کنیم؟ اون موقع نمیتونیم بگیم چیکار میشه کرد، بلکه هر کار بخوایم میتوانیم بکنیم. یعنی میشه این قسمتات قرمز میشه کرد:



بنابراین رابطه برابر میشه با:

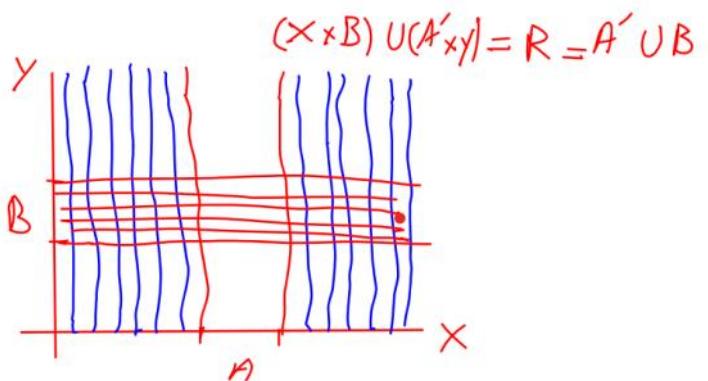
$$R = (A \times B) \cup (A' \times Y)$$

پس این رابطه ای است که بر اساس A آنگاه B در منطق کلاسیک بهش رسیدیم. یک رابطه دیگه هم از تعریف A آنگاه B بود:

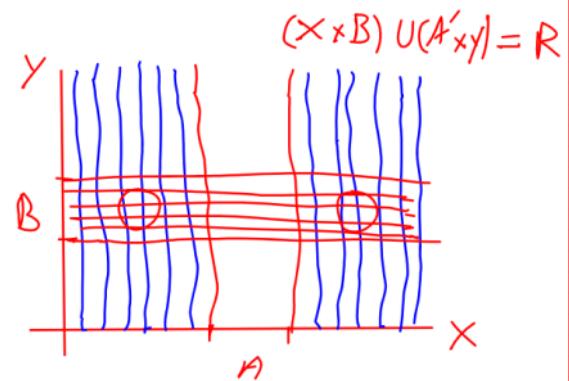
$$R_1 = (A \times B) \cup (A' \times Y)$$

$$R_2 = A' \cup B = (A' \times Y) \cup (X \times B)$$

حال آیا به نظرتون رابطه اول و دوم با هم تفاوت دارن یا ندارن؟ برای این که بهتر بفهمیم این موضوع رو میریم R_2 هم نمودارشو میکشیم:



تفاوت این ۲ تا شکل چیه؟ یک فرق کوچولو فقط دارن. این که نگاه کین این ۲ تا جا که دایره کشیدیم هم قرمز داریم هم آبی:



ولی توی قبیل همه جا یدونه بیشتر نیست. یعنی در اجتماع بعضی جاها شده $aVa = a$ و چون در حالت کلاسیک $aVa = a$ این ۲ تا شکل فرق ندارن. پس توی حالت کلاسیک برابر، توی بعضی حالت های نرمال هم برابر مثل Max که بگیریم بازه میشه. ولی توی بعضی جاها مثل جمع و ضرب جبری داریم:

$$\begin{aligned} aVb &= \\ a + b - ab & \\ \bullet & \\ aVa &= aa - a^2 \end{aligned}$$

پس اینجاها این ۲ رابطه ها با هم یک نیستن. در عمل هیچکدام مناسب نیستن. چرا؟ چون مثلا اگر بخوایم یک ریات بسازیم که براساس این منطق استفاده بکنه داریم میگیم وقتی برقرار نبود هرکاری میخوابی بکن ولی این آیا قابل قبول هست؟!

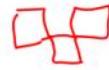
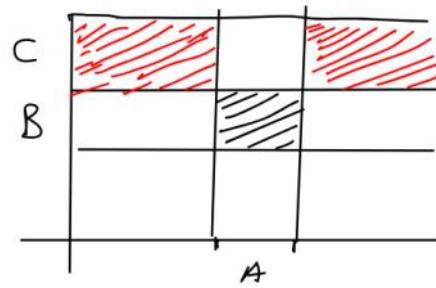
بنابراین ۲ قانون دیگه تعریف میکنیم.
اول این که وقتی شرط برقرار نبود باید بگیم چیکار کنه

$$A \hookrightarrow B \text{ else } C$$

حالا اگر بخوایم به صورت شکلی اینو در بیاریم و رابطه جبریشم بنویسیم اینطوری میشه که:

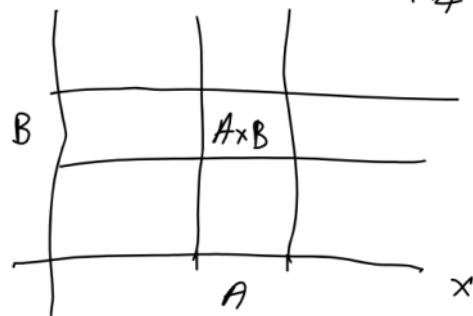
$$A \Rightarrow B \text{ else } C$$

$$R_3 = (A \times B) \cup (A' \times C)$$



اگر قانون رو به صورت درکی که روزمره ازش داریم استفاده داریم، اینطوریه که اگر شرط برقرار بود انجام بد و لی اگر نبود هیچ کار نکن، این که هیچ کار نکن میشه اجتماع با تهی که میشه:

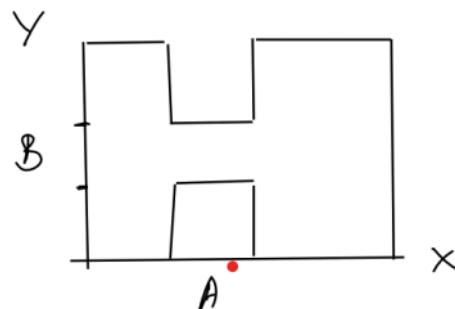
$$R_4 = (A \times B) \cup (A' \times C)$$



$$R_4 = A \times B$$

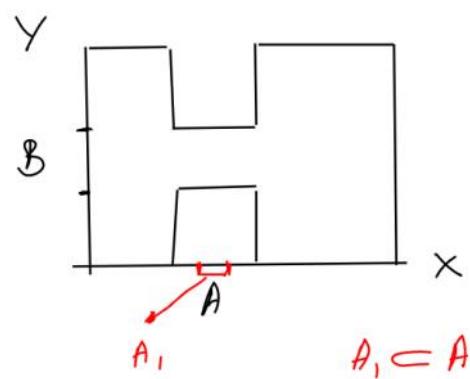
* این حرف همشون به حالت انیمیشنی در اسلاید ها موجوده

تا الان دیدیم که:



$$R_1 = R_2$$

حالا میخوایم ببینیم بسته به ورودی خروجی چی میشه. فرض شود که یک A_1 داشته باشیم:

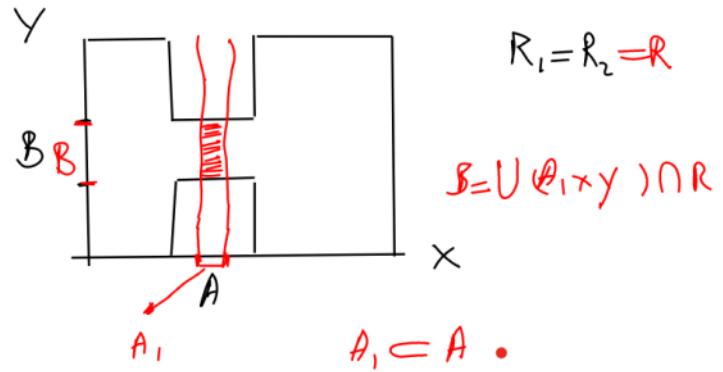


R

اگر A_1 ورودی باشه خروجی میشه به 2 روش محاسبه کرد:

$$B_1 = A_1^o R$$

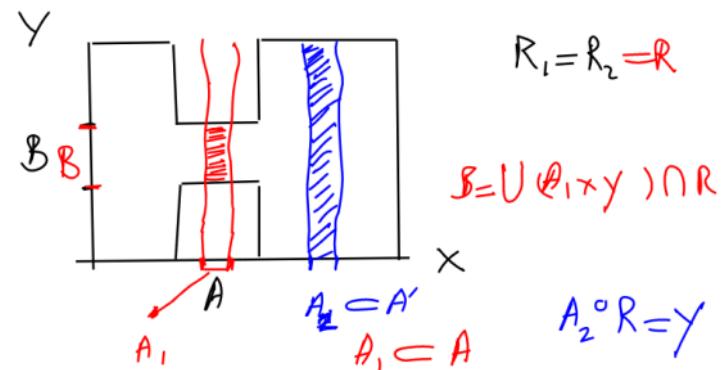
پس اول A_1 رو توسعه استوانه ای میدیم که با این کار انگار داریم در γ ضرب میکنیم. بعد اشتراک با B رو میگیریم و در آخر تصویر میکنیم رو B



$$B_1 = A_1^o R = B$$

پس B_1 برابر شد با B . نتیجه اخلاقی این چیه؟ در حالت کلاسیک حق ما زیر مجموعه از A رو داشته باشیم باز میتوانیم B رو بدست بیاریم و داستان جدایپذیر و جدانایپذیر بودن رو نداریم.

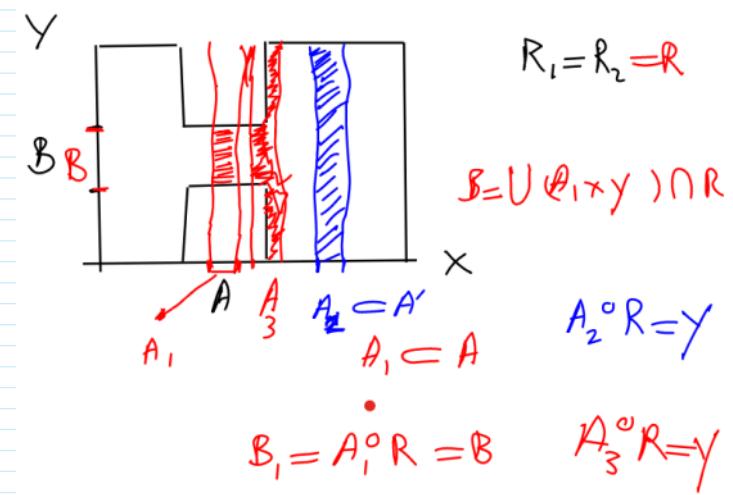
الا اگر A_2 ما زیرمجموعه از A' باشه خروجی چی میشه؟



$$B_1 = A_1^o R = B$$

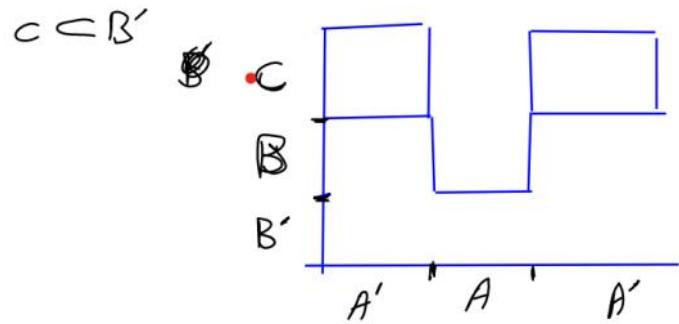
میشه مجموعه مرجع γ , یعنی وقتی شرط برقرار نبود هرکاری میخوای بکن.

اگر A_3 هم داشته باشیم مرزش بیافته، اون موقع میشه:

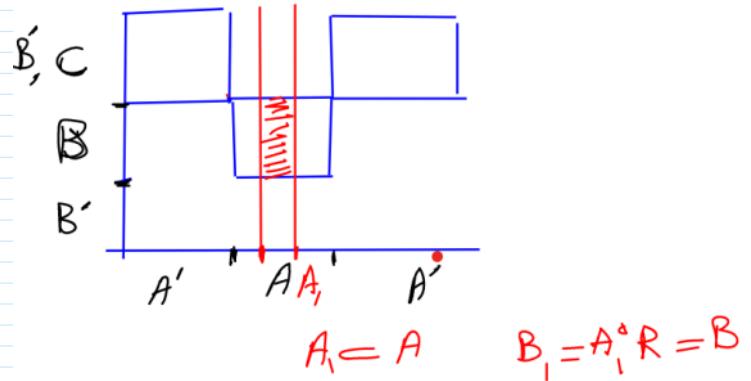


در منطق فازی سه بعدی میشے. اگه حالا بر اساس جدایزیر بودن یا نبودن داستان فرق میکرد که جلسات قبل دیدیم.

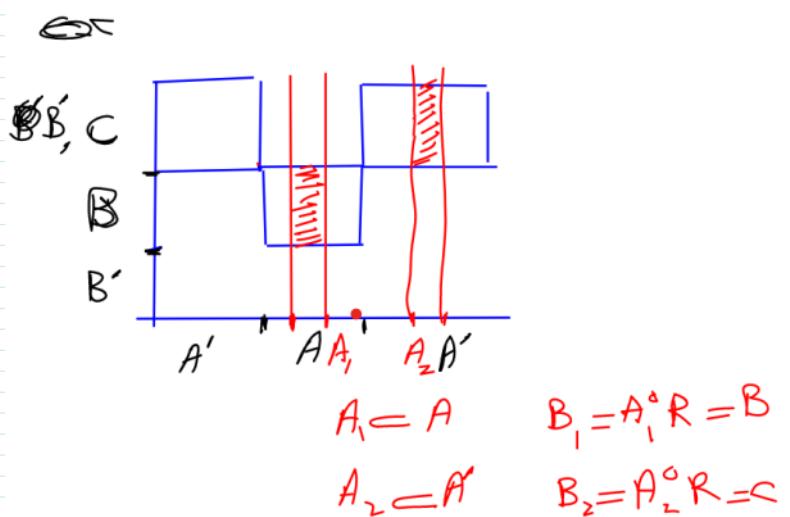
حالا برگردیم سراغ رابطه دوم که گفتیم اگر A آنگاه B در غیر این صورت:



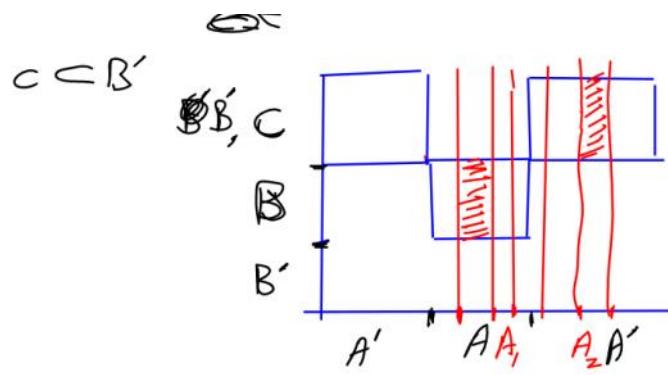
حالا اینجا بایام چند حالت چک کنیم:
حالت اول این که A_1 زیر مجموعه A باشه. در این صورت B_1 چی میشه؟



حالا اگر A_2 زیر مجموعه A' باشه، B_2 میشه:



حالا فرض کنیم یک A_3 هم تعریف کنیم که توی مرز بیافته، اون موقع داریم:



$$A_3 \cap A \neq \emptyset$$

$$A_3 \cap A' \neq \emptyset$$

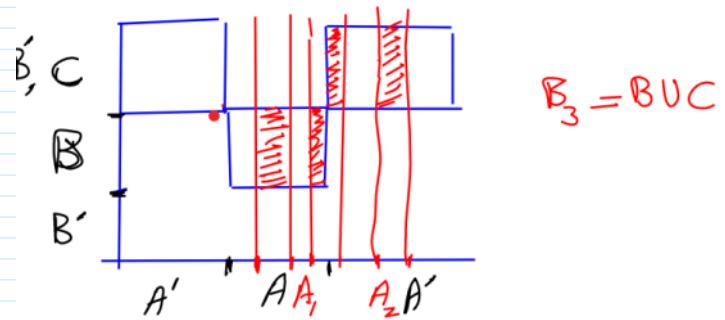
$$A = A$$

$$A_2 = A'$$

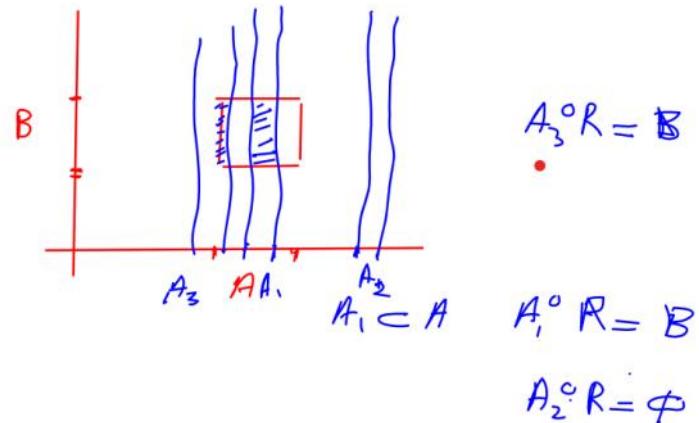
$$B_1 = A_1^o R = B$$

$$B_2 = A_2^o R = C$$

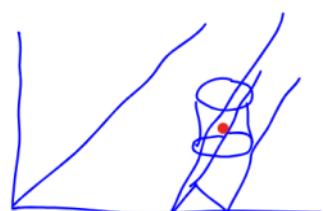
در این اصورت توسعه استوانه ای A_3 رو انجام بدیم و بعد تصویر کنیم B_3 میشه:



و در آخر حالت رو در نظر بگیریم که حالت پیش فرضیون بود، این که وقی شرط برقرار نبود هیچکار نکنیم. در این حالت داریم:



در حالت فازی هم روش همینه فقط شکلا سه بعدیه. مثلا رابطمون استوانه باشه و ورودیمون یک مثلث که توسعه استوانه ای بدیم نبیش بشه:



اشتراکشو بگیریم میشه مخروط ولی اگه همینو تصویر کنیم اونور میشه مستطیل. این رابطه الان جدایندر نیست چون تصویرارو توسعه استوانه ای بدیم استوانه نمیشه.

گزاره های منطقی منطق فازی

$P: x \text{ is } A$ $Q: y \text{ is } B$

$$T(P) = \mu_A(x) \quad T(Q) = \mu_B(y)$$

$P \vee Q : x \text{ is } A \text{ or } y \text{ is } B \equiv T(P \vee Q) = \text{Max}(T(P), T(Q))$

$P \wedge Q : x \text{ is } A \text{ and } y \text{ is } B \equiv T(P \wedge Q) = \text{Min}(T(P), T(Q))$

$$P' : T(P') = 1 - T(P)$$

$P \rightarrow Q : x \text{ is } A \text{ then } y \text{ is } B \equiv T(P \rightarrow Q) = T(P' \vee Q)$

در حالت فازی هم مثل قبلیه فقط مقدار بین 0 و 1 دارد

در جلسه قبل دیدیم:

$$A \Rightarrow B$$

$$R_1 = A' \cup B$$

$$R_2 = (A \times B) \cup (A' \times y)$$

$$R_3 = A \times B$$

$$A \Rightarrow B \text{ در سه ای}$$

$$R_4 = (A \times B) \cup (A' \times C)$$

اگر بخوایم با مقدار تعلق بگیم:

رابطه اول:

$$\mu_{R_1}(x, y) = \mu_{A'}(x) \vee \mu_B(y)$$

که آگه اجتماع را با \max تعريف کنیم میشه

$$\max\{\mu_{A'}(x), \mu_B(y)\}$$

در رابطه دوم هم:

$$\mu_{R_2}(x, y) = (\mu_A(x) \wedge \mu_B(y)) \vee \mu_{A'}(x)$$

چون y میشه 1 وقی ضرب بشه دیگه میره
که اینم برابر با:

$$\mu_{R_2} = \max \left\{ \min \left\{ \mu_A(x), \mu_B(y) \right\}, 1 - \mu_A(x) \right\}$$

برای رابطه سوم:

$$\mu_{R_3}(x, y) = \mu_A(x) \wedge \mu_B(y) = \min\{\mu_A(x), \mu_B(y)\}$$

برای رابطه چهارم:

$$\mu_{R_4}(x, y) = (\mu_A(x) \wedge \mu_B(y)) \vee (\mu_A(x) \wedge \mu_C(y))$$

$$\mu_{R_4} = \max\{\min(\mu_A(x), \mu_B(y)), \min(1 - \mu_A(x), \mu_C(y))\}$$

گزاره های منطقی
فرمehا دیگر "اگر - آنگاه"

R = A → B

$\mu_R(x, y) = \max(\min(\mu_A(x), \mu_B(y)), 1 - \mu_A(x))$
 $\mu_R(x, y) = \max(\mu_B(y), 1 - \mu_A(x))$
 $\mu_R(x, y) = \mu_A(x) \cdot \mu_B(y)$
 $\mu_R(x, y) = \min(\mu_A(x), \mu_B(y))$
 $\mu_R(x, y) = \min(1, 1 - \mu_A(x) + \mu_B(y))$
 $\mu_R(x, y) = \min(1, \mu_A(x) + \mu_B(y))$
 $\mu_R(x, y) = \min(1, \mu_B(x) / \mu_A(y))$
 $\mu_R(x, y) = \max(\mu_A(x) \cdot \mu_B(y), 1 - \mu_A(x))$

مثال:

اگر ب A_1 دو کیفیت A_2 نداشته باشد
 B A_2 A_1

$$A_1 \times A_2 \Rightarrow B$$

برای نوشتن رابطه این قانون باید از R3 استفاده بشه. چرا؟ چون نمیتوانیم بگیم وقتی شرط برقرار نبود هر کاری میخواهی بکن. پس میفهمیم بسته به کاربرد باید تصمیم بگیریم از چه فرمولی استفاده کرد.

$$A_1 \times A_2 \Rightarrow B \quad R = A_1 \times A_2 \times B$$

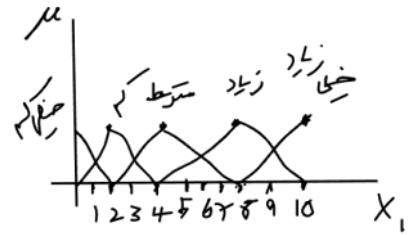
$$\mu_R(x_1, x_2, y) = \min\{\mu_{A_1}(x_1), \mu_{A_2}(x_2), \mu_B(y)\}$$

اگر به صورت مینیمم فرض کنیم مشابه فوق میشه. اگر به صورت ضرب فرض کنیم:

$$\begin{aligned} \mu_R(x_1, x_2, y) &= \min\{\mu_{A_1}(x_1), \mu_{A_2}(x_2), \mu_B(y)\} \\ &= \mu_{A_1}(x_1) \cdot \mu_{A_2}(x_2) \cdot \mu_B(y) \end{aligned}$$

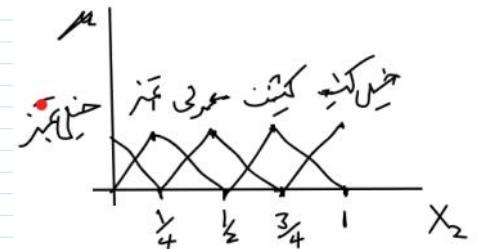
$$A_1 \text{ دارد} ; \quad A_2 \text{ کنیت} \quad B \text{ داری} \quad A_1 \times A_2 \Rightarrow B$$

اگر برای وزن لباس ها، خیلی زیاد، زیاد، معمولی و کم داشته باشیم، میشه گفت مثلا از 8 تا 10 کیلو خیلی زیاده، از 4 تا 10 بشه زیاد، از 2 تا 8 متوسط، 0 تا 4 کم و 0 تا 2 خیلی کم باشه.



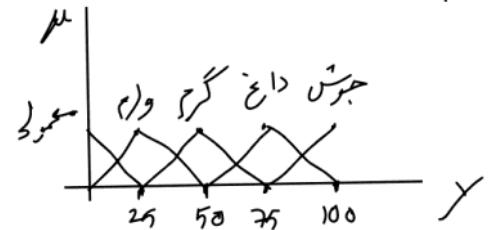
این نقاط توسط افراد خبره با آمارگیری باید تعیین بشه.

به همین ترتیب برای کشیف میتوان گفت:



نقاط کلیدی که اون نقطه ها اون بالای نمودار هستن با آنتروپی میشه مشخص کرد.

برای داغ کردن هم میشه گفت:



حال میایم میگیم که:

$$\text{اگر بسیار نسبتاً زیاد رہنمایی داشته باشند آب بخیکوبی باید باش} \\ B^* \quad A_2^* \quad A_1^*$$

برای بدست آوردن آب که B^* هست داریم:

$$B^* = A_2^* \circ R$$

$$B^* = (A_1^* \times A_2^*) \circ (A_1 \times A_2 \times B)$$

اگر برای نسبتاً زیاد بگیم 6 تا 8 باشه. با توجه به نمودار های رسم شده و مقدارایی که دارن میشه گفت:

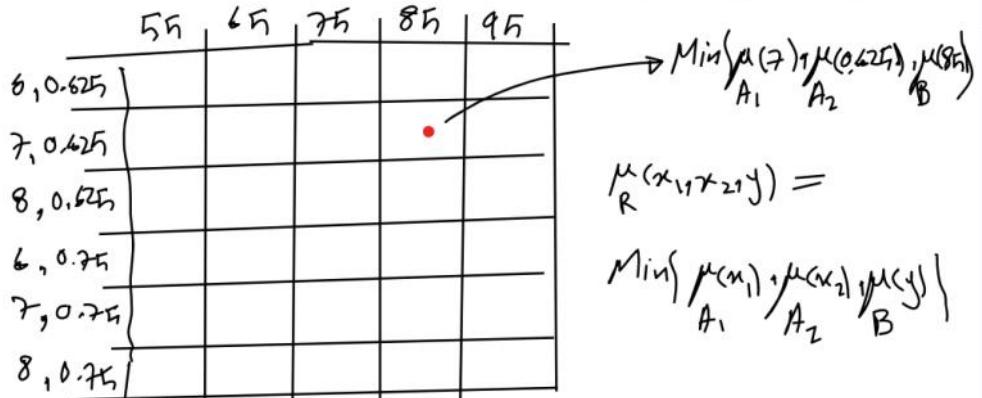
$$A_1^* = \left\{ \frac{0.3}{6} + \frac{1}{7} + \frac{0.4}{8} \right\}$$

$$A_2^* = \left\{ \frac{0.1}{0.625} + \frac{0.8}{0.75} \right\}$$

حالا ابتدا بایستی حاصل ضرب کارتزین A_1^* و A_2^* را بدست بیاریم. (وقت بشه توی قانون استار نداره ولی برای ورودی استار داره و تو امتحان حواسمند باشه اشتباه نکنیم) سپس با مینیمم گرفتن داریم:

$$A_1^* \times A_2^* = \left\{ \frac{0.1}{6,0625} + \frac{0.1}{7,0625} + \frac{0.1}{8,0625} + \frac{0.3}{6,075} + \frac{0.8}{7,075} + \frac{0.4}{8,075} \right\}$$

حالا میریم مقدار خروجی رو گرسنته سازی میکنیم. بعد رابطه رو بر حسب مقدار تعلق مینویسیم. چون در رابطه دیگه بر حسب قانونه دیگه استار نباید بذاریم! داریم:



حالا مقدار تعلق چطوری بدست بیاد؟ مثلا در خونه اول جدول رو نگاه میکنیم. $A_1 = 0$ و در 4 هم $A_2 = 0$. میریم تو اون نمودارا که کشیدیم مبینیم در نقطه 8 داریم = 6 هستش. چون خطی داره بالا اون نقطه وسطش که در 6 هستش میشه 0.5. به همین ترتیب برای بقیه هم بدست میاریم و جدول رو پر میکنیم.

*برای بدست آوردن 55 در B اینطوری عمل کرد که گفت از بین 50 تا 75 ما 25 تا مقدار داریم. 55 میشه 5 از اون 25 که 5/25 میشه 25 از 100 که حالا 0.2 در نظر گرفتن.

	55	65	75	85	95
55	0.2	0.6	1	0.6	0.2
65	0.7	0.5	0.5	0.5	0.2
75	0.2	0.5	0.5	0.5	0.2
85	0.7	0.5	0.5	0.5	0.2
95	0.2	0.6	1	0.6	0.2

7.

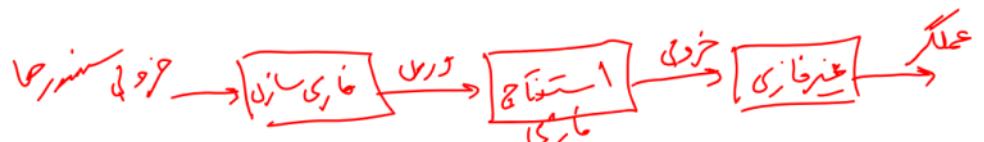
	$\frac{0.2}{55}$	$\frac{0.6}{65}$	$\frac{1}{75}$	$\frac{0.6}{85}$	$\frac{0.2}{95}$
$\frac{0.5}{55} \times 6, 0.625$	0.7 0.1	0.5 0.1	0.5 0.1	0.5 0.1	0.2
$\frac{0.5}{65} \times 7, 0.425$	0.7 0.1	0.5 0.1	0.5 0.1	0.5 0.1	0.2
$\frac{1}{75} \times 8, 0.625$	0.2 0.1	0.5 0.1	0.5 0.1	0.5 0.1	0.2
$\frac{0.5}{85} \times 1, 0.75$	0.1 0.3	0.5 0.1	0.5 0.1	0.5 0.1	0.2
$\frac{0.2}{95} \times 1, 0.75$	0.2 0.8	0.6 0.2	0.6 0.2	0.6 0.2	0.2
$ 8, 0.75$	0.2 0.4	0.6 0.4	0.4 0.4	0.4 0.4	0.2

0.2 0.6 0.75 0.6 0.2

در نتیجه:

$$\tilde{B} = \left\{ \frac{0.2}{55}, \frac{0.6}{65}, \frac{1}{75}, \frac{0.6}{85}, \frac{0.2}{95} \right\}$$

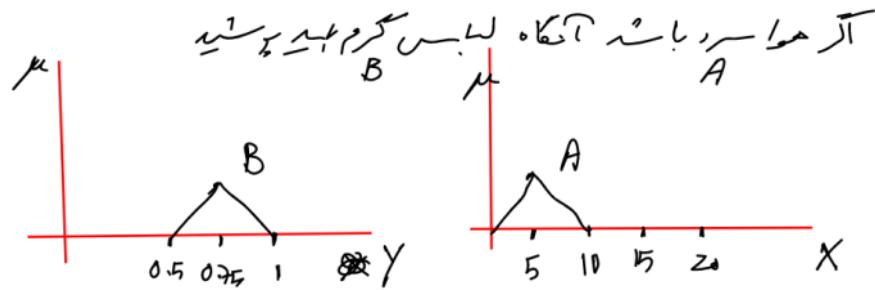
(در جواب به پرسش بچه ها، به طور کلی اتفاق زیر قراره بیافته:



جلسه هفدهم

فرض کنیم قانون داریم "اگر هوا سرد باشد، آنگاه لباس گرم باید پوشید"

می‌ایم درجه هوا رو تا 20 درجه در نظر میگیریم و برای لباس هم ضخامت رودر نظر میگیریم و نرمال میکنیم که ضخیم ترینش 1 باشه. حالا داریم:



حالا برای این قانون، رابطه اش را با حاصل ضرب کارتزین می‌بریم چون وقتی هوا سرد نبود نمیخوایم کاری کنیم. پس داریم:

$$A \Rightarrow B$$

$$R = A \times B$$

$$\mu_{R(x,y)} = \min_{(x,y)} \mu_A(x), \mu_B(y)$$

حالا فرض شود بهمنون بگن هوا 6 درجه است. با این قانون میخوایم ببینیم چه لباسی باید پوشید. چیکار کنیم؟ اگر ورودی A^* بگیریم و خروجی B^* را بخوایم بدست بپاریم داریم:

$$B^* = A^* \circ R$$

ولی ما الان A^* رونداریم. چطوری بدست بپاریم؟ می‌ایم با برآش تعریف میکنیم. به شکلی که:

$$\mu_{A^*}(x^*) = \begin{cases} 1 & x^* = 6 \\ 0 & x^* \neq 6 \end{cases}$$

سپس داریم:

$$B^* = A^* \circ R$$

$$\mu_{B^*}(y) = \bigvee_{x \in X} (\mu_{A^*}(x) \wedge \mu_R(x, y)) \quad \mu_{A^*}(x) = \begin{cases} 1 & x = 6 \\ 0 & x \neq 6 \end{cases}$$

$$\mu_{B^*}(y) = \max_{x \in X} \left\{ \min_{A^*} \left(\mu_{A^*}(x), \mu_R(x, y) \right) \right\}$$

حال 2 حالت داریم. یا این که $(A^*)_6$ امون 1 هست یا 0. تو حالتی که صفر داریم مینیمم میشه 0 و تو حالتی که 1 داریم مینیمم میشه خود اون میباشد. پس داریم:

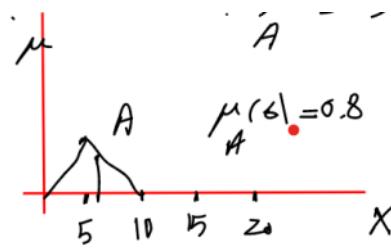
$$\mu_{B^*}(y) = \max_{x \in X} \left\{ \min_{A^*} \left(\mu_{A^*}(x), \mu_R(x, y) \right) \right\} = \max \left\{ 0, \mu_R(6, y) \right\}$$

$$\mu_{B^*}(y) = \mu_R(6, y)$$

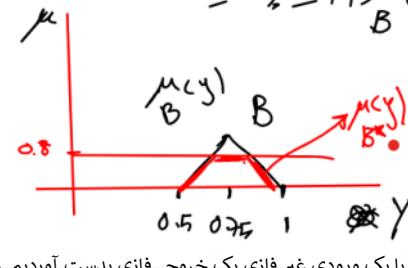
خود μ_R هم برابره با:

$$\mu_{B^*}(y) = \mu_R(6, y) = \min \left\{ \mu_A(6), \mu_B(y) \right\}$$

برای بدست آوردن مقادیر برمیگردیم به شکلا:



ولی باشد مینیمم بگیریم با $\mu_B(y)$ که میشه:



پس ما با یک ورودی غیر فازی یک خروجی فازی بدست آوردم. حالا بخوایم این خروجی فازی را غیر فازی کنیم مثلاً میشه مرکز ثقل را حساب کرد که چون ذوزنقه و متقارن هست میشه میشه 0.75 در نظر گرفت پس:

$$\mu_{A \cap B}(y) = \min(\mu_A(y), \mu_B(y))$$

$$y = 0.75$$

یک نکته جالب در این مثال اینه که اگر ورودی به جای مجموعه فازی یک مقدار دقیق باشه میشه ماکسیمم رو حذف کرد. چون بین 0 و یک مقدار باید ماکسیمم گیری بشه و دیگه ماکسیمم حذف میشه. دلیلش چیه؟ چون برای اشتراک از مینیمم استفاده کردیم و آله ضرب استفاده بشه داستان متفاوت میشه.

$$\text{Mean}\{0.2, 0.8\} = 0.8$$

$$0.2 \cdot 0.8 = 0.8$$

$$xy \quad \begin{cases} a \cdot b \\ a+b-ab \end{cases}$$

وقتی از مینیمم و ماکسیمم برای اجتماع و اشتراک استفاده کنیم دیگه تفاوقي بین 0.2 و 0.3 ایجاد نمیشه و نتیجه نهایی هم یکی میشه و برای این که یک تفاوقي قائل بشیم مثلاً باید از ضرب a+b-ab استفاده کرد.

حالا اگر به جای یک قانون تعداد قانونامون زیاد بود چیکار کنیم؟ مثلاً اگر اینا قانون هامون باشن (با فرض این که A_1, A_n مجموعه مرجع یکسانی داشته باشند):

$$A_1 \Rightarrow B_1$$

$$A_i \Rightarrow B_i$$

$$A_N \Rightarrow B_N$$

حالا برای این که اشتباه برداشت نشه اینتا رو اون بالا بناریم که با ابعاد اشتباه نگیریم:

$$A' \Rightarrow B'$$

$$A'' \Rightarrow B''$$

$$A^N \Rightarrow B^N$$

و بهمون A^* بدن پن B^* جی میشه، چیکار کنیم؟ ما برای هر قانون یک رابطه ای داریم که B^* داخل هر کدام به این روش بدست میاد:

$$A' \Rightarrow B' \quad R' \quad B'' = A^* \circ R'$$

$$A'' \Rightarrow B'' \quad R^i \quad B'' = A^* \circ R^i$$

$$A^N \Rightarrow B^N \quad R^N \quad B^N = A^* \circ R^N$$

که هر یک از A^* هایی که بدست میاد یک خروجی جزئی هستش.
برای بدست آوردن خروجی کل چیکار کنیم؟ بستگی دارد بین قوانین جی باشه. اگر بین قوانین اشتراکه میایم اشتراک میگیریم و اگر بینش اجتماعه میایم اجتماع میگیریم.

مثلا اگر بگیم: "اگر هوا سرد باشد لباس گرم بپوشیم، اگر گرم باشه لباس خنک." این قانون بینش اشتراکه یا اجتماعی؟ بینش اجتماع هست چون هر کدام حالت خاصی رو پوشش میدن.

ولی وقتی بگیم برای این که لباس گرم بپوشیم باید هوا سرد باشه و تو خونه نباشیم. اون موقع دیگه بینش اشتراک میشه

حالا اگر فرض کنیم بین قوانین اجتماع باشه. اون موقع خروجی کل میشه:

$$\tilde{B} = \tilde{B}' \cup \dots \cup \tilde{B}^N$$

اگر بین بعضی ها اجتماع و برخی اشتراک باشه هم:

$$\boxed{\tilde{B} = (\tilde{B}' \cup \tilde{B}^2) \cap \tilde{B}^3}$$

قانون دوشرطی زیر رو در نظر بگیریم:

$$A \leftrightarrow B$$

تفسیر این قانون یعنی این که:

$$A \leftrightarrow B \equiv (A' \cup B) \cap (B' \cup A)$$

$$\equiv (A \Rightarrow B) \cap (B \Rightarrow A)$$

پس دو قانون داریم بینش اشتراکه. و میتوان گفت:

$$\equiv (A \Rightarrow B) \cap (B \Rightarrow A)$$

R'

R^2

$$\tilde{B}^1 = A^* \circ R^1$$

$$\tilde{B}^2 = A^* \circ R^2$$

$$B^* = \tilde{B}^1 \cap \tilde{B}^2$$

سوال: آیا میشه بر اساس رابطه هر قانون یک رابطه نهایی تولید کرد که ورودی فقط یک بار با رابطه نهایی ترکیب بشه و هزینه محاسباتی کمتر بشه؟! جواب این سوال بسته به اینه که بینشون اشتراکه یا اجتماع، اگر بینشون اجتماع باشه میشه ولی اشتراک باشه نمیشه (چرا؟!) علت:

$$R' \quad \tilde{B}^1 = A^* \circ R'$$

$$\vdots \\ R^i \quad \tilde{B}^i = A^* \circ R^i$$

$$R^N \quad \tilde{B}^N = A^* \circ R^N$$

$$\begin{aligned} \tilde{B} &= \bigcup_i (A^* \circ R^i) \\ &= \bigcup_i (\bigcup_{x \in X} A^* \cap R^i) \end{aligned}$$

سپس اجتماع پرونی رو میشه داخل و میشه نوشت:

$$= \bigcup_{n \in X} A^* \cap \left(\bigcup_i R^i \right)$$

$$= \bigcup_{x \in X} A^* \circ \left(\bigcup_i R^i \right)$$

و بنابراین رابطه ای میشه تعریف کرد و داریم:

$$R = \bigcup_i R^i$$

$$\tilde{B} = A^* \circ R$$

ولی اگه اشتراک باشه نمیشه این کار رو کرد که حالا میشه تو خونه تمرين کرد بررسی شه که چرا نمیشه.

$$\tilde{B} = \bigcap_i (A^* \circ R^i)$$

حالا بريم سراغ روش های فازی سازی:

یکیش اینه که با تعلق بگیم. مثلا اگر یک x داشته باشیم که برابر با x^* باشه (مثل مثال امروز که برابر با 6 بود) ببایم بگیم:

$$\chi = \begin{cases} 2 & \mu_{\text{آ}}(\chi) = 1 \\ 0 & \mu_{\text{آ}}(\chi) < 1 \end{cases}$$



ولی نمیشه گفت فازی سازی شده فقط نمایش رو با تابع تعلق بیان کردیم.

روش دیگه اینه به صورت شهودی نقاط کلیدی رو مشخص کنیم که یک شخص خبره یا آمارگیری تعیین میکرد بعد اینارو طوری به صورت خطی وصل میکنیم که در هر نقطه جمعشون بشه ۱



استنتاج

مجموعه مثلث ها $U = \{(A, B, C) | A \geq B \geq C \geq 0, A + B + C = 180^\circ\}$

مثلث متساوی الاضلاع تقریبی $\mu_I = 1 - \min(A - B, B - C) / 60^\circ$

مثلث قائم الا زاویه تقریبی $\mu_R = 1 - |A - 90^\circ| / 90^\circ$

مثلث متساوی الاضلاع تقریبی $\mu_E = 1 - |A - C| / 180^\circ$

مثلث متساوی الاضلاع و قائم الا زاویه تقریبی $\mu_{IR} = \mu_I \wedge \mu_R$

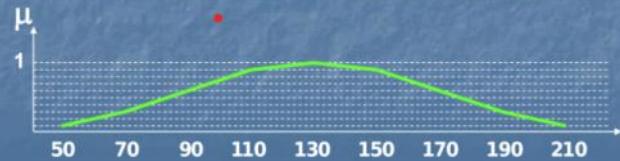
دیگر مثلث ها $\mu_T = (\mu_I \vee \mu_R \vee \mu_E)' = \mu_I' \wedge \mu_R' \wedge \mu_E'$

از فرمول های ریاضی میشه کمک گرفت. مثلا برای مثلث قائم الزاویه تقریبی بخوایم مقدار تعلق تعريف کنیم. اگر A رو بزرگترین زیاویه بعد B بعد C در نظر بگیریم. بزرگترينش باید 90 بشه پس اون رابطه زرد رنگه میگه هرچی A به 90 نزدیک تر بشه تعلق به ۱ نزدیکتر میشه و بالعکس هرچی از 90 دور کنیم به ۰ نزدیکتر میشه

از منطق هم میشه تعريف کرد مثلث برای ۳ تا مثلث تعريف کردیم برای بقیه مثلثا میشه متمم اونا.

مرتب کردن آماری

سطح	۵۰	۷۰	۹۰	۱۱۰	۱۳۰	۱۵۰	۱۷۰	۱۹۰	۲۱۰
رای	۸۰	۲۲۰	۵۸۰	۸۹۰	۱۰۰۰	۹۲۰	۶۳۰	۲۵۰	۱۱۰
ترتیب	۹	۷	۵	۳	۱	۲	۴	۶	۸
درصد	.08	.33	.58	.89	۱	.92	.63	.35	.11



یک روش هم آمارگیریه که از افراد پرسیم هر کدام چه رای دادن و منحنی بدست بیاریم

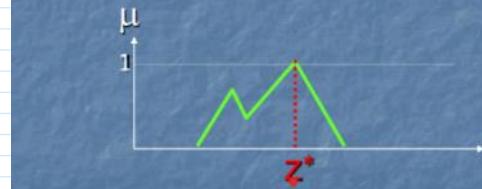
از آنتربی و شبکه عصبی و الگوریتم های تکاملی هم میشه که استاد گذر کرد چون وقت نمیشه.

حالا بیم سراغ غیر فازی سازی:

ماکریزم

جایی که تابع تعلق بیشترین مقدار را دارد

$$\mu_C(z^*) \geq \mu_C(z) \quad z \in Z$$



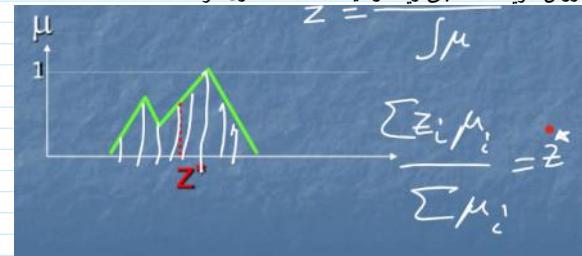
یک که امروزم دیدیم مرکز ثقل هستش:

مرکز ثقل

$$z^* = (\int z \cdot \mu_C(z) \cdot dz) / (\int \mu_C(z) \cdot dz)$$



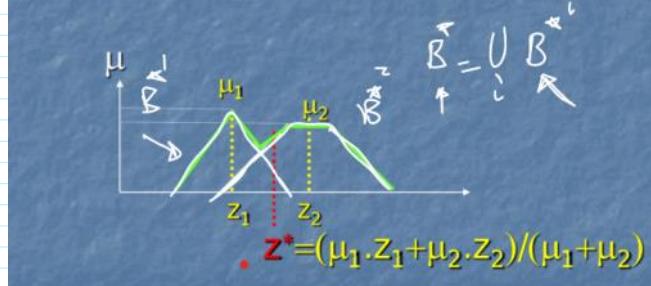
این روش هزینه محاسباتی زیاده و میشه گرسنه سازی کرد :



یک راه حل اینه که متوسط وزنی مراکز رو بگیریم:

متواتسط وزنی مراکز

$$z^* = (\sum \bar{z} \cdot \mu_C(\bar{z})) / (\sum \mu_C(\bar{z}))$$

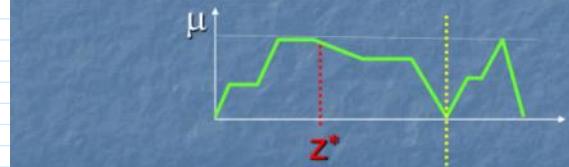


یا مثلا:

مرکز ثقل بزرگترین قسمت

اگر مجموعه فازی حداقل به دو قسمت محدب تقسیم شود

$$z^* = (\int z \mu_{Cm}(z) dz) / (\int \mu_{Cm}(z) dz)$$



تو این روش اون قسمت های کوچیک رو میگیریم نویزه و حذف میکنیم و برای بقیه مرکز ثقل رو میگیریم.

سیستم خبره فازی

برای تشکیل یک حالت استاندارد برای سیستم خبره فازی یک سری اصطلاحات و استاندارد پیش میاد که اول میایم اینارو تعریف کنیم:

- ترم های اتمیک: ترم هایی که نمیتوان به قسمت های کوچکتری تقسیم کرد مثل جوان، پیر، آهسته و برای اینا میشه مجموعه فازی هم تشکیل داد. معمولاً مقدارشون در یک بازه هستش و مقدار دقیق هم ندارن.
- صفت ها و قید های تصحیح کنده: قبل و بعد ترم های اتیک میاد. مثلاً خیلی جوان یا کم جوان یا ... که با علامت ریاضی میشه بدستش آورد.

پیاده سازی قوانین زبانی

ترمهای اتمیک

جوان، پیر، آهسته، زیبا، متوسط

ترمهای اتمیک را میتوان با مجموعه های فازی مدل کرد

صفتها و قیدهای تصحیح کننده

α^2	خیلی
$\alpha^{0.5}$	کمی
$\alpha^{1.25}$	کمی بیشتر
$\alpha^{0.75}$	کمی کمتر

- قانون استاندارد: یک قانونی رو استاندارد میگن اگه همه متغیر های ورودی در شرط باشن و قسمت نتیجه هم فقط یک باشه. به عبارت دیگه میشه اینطوری نوشتش:

$$A_1^i \times A_2^i \times \dots \times A_n^i \Rightarrow B^i$$

صفحة اسلاید:

پایگاه قوانین استاندارد

$R^1: \text{If } x_1 = A_1^1 \cap x_2 = A_2^1 \cap \dots \cap x_n = A_n^1 \text{ then } y = B^1$

$R^2: \text{If } x_1 = A_1^2 \cap x_2 = A_2^2 \cap \dots \cap x_n = A_n^2 \text{ then } y = B^2$

...

$R^m: \text{If } x_1 = A_1^m \cap x_2 = A_2^m \cap \dots \cap x_n = A_n^m \text{ then } y = B^m$

$A_1^i \times A_2^i \times \dots \times A_n^i \Rightarrow B^i$

تمام قوانین زبانی را میتوان بصورت قوانین استاندارد بازنویسی کرد

فرمتهای دیگر اگر آنگاه را میتوان بصورت استاندارد بازنویسی کرد

?

اگر یک قانونی داشته باشیم که استاندارد نباشه میشه تبدیل کرد به چند قانون استاندارد. مثال:

$$\begin{array}{ccccccc} & & & & & & \\ & x_1 & & x_2 & & x_1 & \\ & & & & & & \\ & A_1 \cup A_2 & \Rightarrow & B_1 & \subset & \subset & B_2 \end{array}$$

در این موقع باید به یکی از دو شکل زیر در بیاریم:

$$\begin{cases} X_1 \times X_2 \Rightarrow Y_1 \\ X_1 \times X_2 \Rightarrow Y_2 \end{cases}$$

برای این کار:

$$(A_1 \cup A_2 \Rightarrow B_1)$$

$$\cap ((A_1 \cup A_2) \Rightarrow B_2)$$

علت اشتراک:

$$A \Rightarrow B < /> C$$

$$(A' \cup B) \cap (A' \cup C)$$

اگر اجتماع باشه A' اجتماع با A میشه مرجع و مرجع هم وقتی داشته باشیم با هرجی اجتماع بشه میشه همواره True

حالا برگردیم به مساله امون:

$$\begin{aligned} & (A_1 \cup A_2 \Rightarrow B_1) \quad ① \\ & \cap ((A_1 \cup A_2) \Rightarrow B_2) \quad ② \end{aligned}$$

$$② \quad A'_1 \cap A'_2 \Rightarrow B_2$$

$$\begin{cases} X_1 \times X_2 \Rightarrow Y_1 \\ X_1 \times X_2 \Rightarrow Y_2 \end{cases}$$

قانون دوم استاندارد هستش چون نمیشه حالت دومی که داشتیم ولی قانون اول استاندارد نیست چون بینشون اجتماعه. پس تبدیلش میکنیم به 2 قانون:

$$\begin{aligned} & ① \quad (A \Rightarrow B_1) \quad @ \\ & \cup (A_2 \Rightarrow B_1) \quad ⑤ \end{aligned}$$

حالا a و b آیا استانداردند؟ نه! چون در a فقط مرجع X_1 رو داریم و برای b هم فقط مرجع X_2 داریم و حاصل ضرب کارتزین نداریم. پس باید هم مرجع کنیم \leftarrow توسعه استوانه ای

$$H_1 \times X_2 \Rightarrow B_1 \quad X_1 \times A_2 \xrightarrow{\sim} B_1$$

پس قوانین استخراج شده استاندارد در آخر برابرند با:

$$(A'_1 \times A'_2 \Rightarrow B_2) \cap \left((X_1 \times A_2 \Rightarrow B_1) \cup (A_1 \times X_2 \Rightarrow B_1) \right)$$

$R^1 \qquad R^2 \qquad R^3$

$$R = R^2 \cup R^3$$

و با توجه به نکته جلسه قبل که اگر اجتماع داشتیم میتوانستیم بین رابطه ها اجتماع بگیریم ولی اگر اشتراک داشتیم نمیتوانستیم اگر بهمون ورودی بدن داریم:

$$A_1^* \times A_2^* = A^*$$

$$B^* = A^* \circ R^*$$

$$B^{**} = A^* \circ R$$

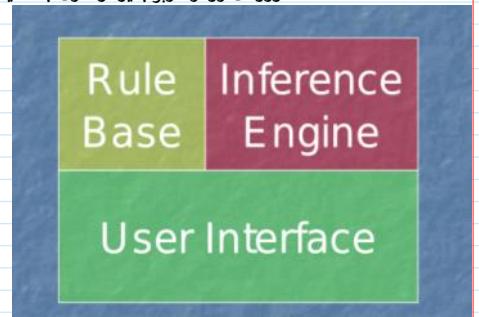
$$B^* = B^* \cap B^{**}$$

حالا برگردیم به سیستم خبره فازی. سیستم خبره فازی از سه بخش تقسیم شده:

- پایگاه قوانین

- موتور استنتاج: ورودی رو با رابطه قوانین ترکیب میکنند و خروجی جزئی بدست میارهند.

- موتور استنتاج: ورودی رو از کاربر بگیره و فازی بکنه یا به صورت تابع تعامل در بیاره و بدنه به موتور استنتاج و خروجی نهایی رو بگیره و غیر فازی کنه و بدنه به کاربر User Interface: ورودی رو از کاربر بگیره و فازی بکنه یا به صورت تابع تعامل در بیاره و بدنه به موتور استنتاج و خروجی نهایی رو بگیره و غیر فازی کنه و بدنه به کاربر



واسطه کاربر سه قسمت دارد

وارد کردن قوانین

گرفتن ورودیها و فازی کردن آنها در صورت غیر فازی بودن

غیر فازی کردن نتیجه و نمایش خروجی



انتخاب یک فرمول برای رابطه اگر آنگاه

$$R^i = A^i \rightarrow B^i \equiv \mu_{R^i}(x, y) = \text{Max}(\text{Min}(\mu_{A^i}(x), \mu_{B^i}(y)), 1 - \mu_{A^i}(x))$$

انتخاب یک فرمول برای ترکیب

$$B_j = A_j \circ R^i \equiv \mu_{B^j}(y) = \text{Max}_{x \in X} (\text{Min}(\mu_{A^j}(x), \mu_{R^i}(x, y)))$$

بدست آوردن خروجی برای تمام قوانین سپس بدست آوردن اجتماع یا

اشتراک خروجیها بسته به نوع قوانین



حالا اگر یک سیستم فازی داشته باشیم چند تا خروجی داشته باشه یا خروجی یکی بره تو ورودی یکی دیگه میشه از چندتا موتور استنتاج استفاده کرد:

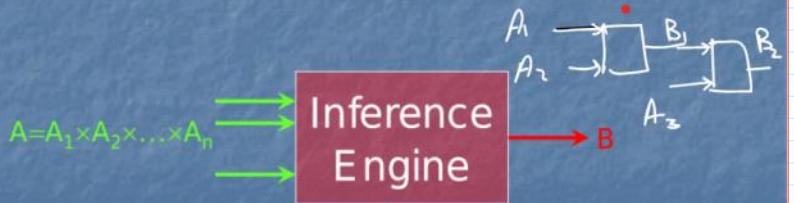
موتور استنتاج

اجزای سیستم خبره فازی

اگر تمام قوانین باید صادق باشند از اشتراک استفاده میگردد

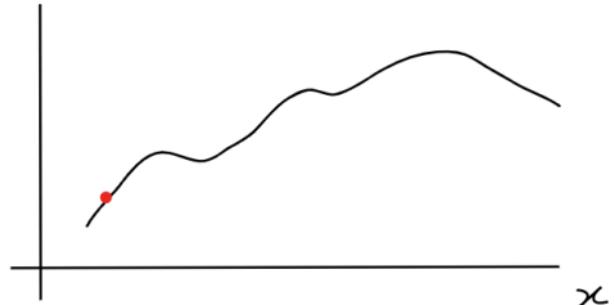
اگر برای داشتن خروجی حداقل یک قانون باید معتبر باشد از اجتماع استفاده میگردد

در صورت داشتن چند خروجی از چند موتور استنتاج استفاده میشود

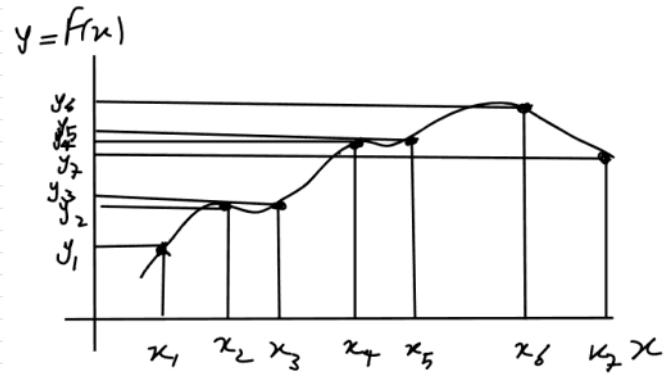


فرض کنید که شما یک سری داده دارید. میخوایم یه چیزی شبکه سیستم خبره فازی درست کنیم که مشابه شبکه عصبی کار کنه. فرض کنیم یک منحنی فرضی موجوده:

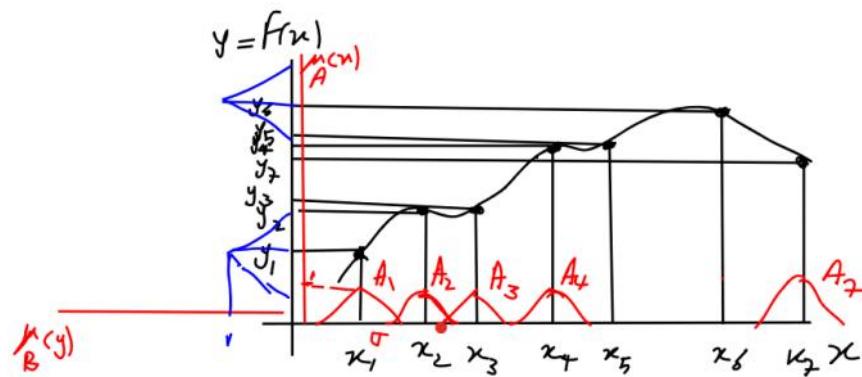
$$y = f(x)$$



ما فقط یک سری نقاط این از این منحنی رو تحت عنوان train data داریم:



حالا میخوایم یک سری قوانین درست کنیم بر اساس این داده ها که داریم.



$$A_1 \Rightarrow B_1$$

پس ما اولین کاری که کردیم اینه که قانون تعریف کردیم. چطوری؟ به ازای داده های که داریم اون داده های فازی ورودی و خروجی و اطراف هم یک مقدار پوشش بدن. حالا از بین رابطه های زیر کدامو انتخاب کنیم؟

$$R_1 = A' \cup B$$

$$R_2 = (A \times B) \cup (A' \times y)$$

$$R_3 = A \times B$$

$$R_4$$

مسلمانه R_3 چون وقتی رابطه برقراره که میدونیم چیکار بکنیم ولی برقرار نباشه هیچ کاری نباید بکنیم.

$$A' \Rightarrow B' \quad R' = A' \times B'$$

$$A^i \Rightarrow B^i \quad R^i = A^i \times B^i \quad \mu_{R^i}^{(x,y)} = \mu_{A^i}^{(x)} \mu_{B^i}^{(y)}$$

$$A^N \Rightarrow B^N \quad R^N = A^N \times B^N$$

برای ترکیب هم فرض ممکنیم از ماکس دات استفاده میکنیم

حالا بهمون یه ورودی میدن میگن خروجیش چی میشه. چیکار کنیم؟ باید فازیش کنیم. جلسه پیش دیدیم یه راه سادش اینه که:

$$x = x^* \quad A^* \quad \mu_{A^*}^{(x)} = \begin{cases} 1 & x = x^* \\ 0 & x \neq x^* \end{cases}$$

$$y = ?$$

$$B^i = A^* \circ R^i$$

$$\mu_{B^i}^{(y)} = \max_{A^*} (\mu_{A^*}^{(x)} \cdot \mu_{R^i}^{(x, y)})$$

$$= \max_{x^*} \left(\begin{array}{cc} x = x^* & \mu_{R^i}^{(x^*, y)} \\ x \neq x^* & 0 \end{array} \right) = \mu_{R^i}^{(x^*, y)}$$

$$B^i = A^* \circ R^i$$

$$\mu_{B^i}^{(y)} = \max_{A^*} (\mu_{A^*}^{(x)} \cdot \mu_{R^i}^{(x, y)})$$

$$\mu_{B^i}^{(y)} = \mu_{A^i}^{(x^*)} \mu_{R^i}^{(x)}$$

که داریم:

$$\mu_{B^i}^*(y) = \frac{\mu(x^*)}{A^i} \mu_i^*(y)$$

$$\mu_{B^*}^*(y) = \mu(y) \vee \dots \vee \mu(y)$$

اگر از متوسط وزن مراکز استفاده کنیم:

مکان
مرکز

$$\mu_{B^*}^*(y) = \frac{\sum_i z_i \mu_i^*(y)}{\sum_i \mu_i^*(x^*)}$$

مکان
مرکز

$$\mu_{B^*}^*(y) = \mu(y) \vee \dots \vee \mu(y)$$

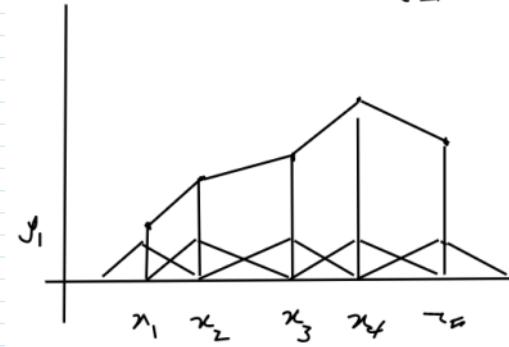
$$z^* = \frac{\sum_i z_i \mu_i}{\sum_i \mu_i} \quad y^* = \frac{\sum_i y_i \mu_i^*(x^*)}{\sum_i \mu_i^*(x^*)}$$

پس سیستم خبری فازی برابر میشه با :

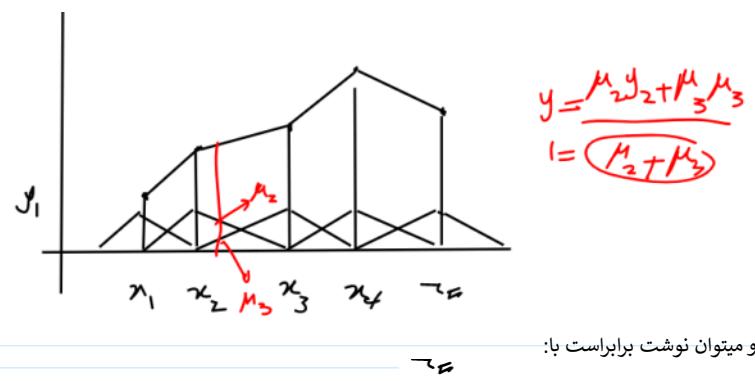
$$f(x^*) = y^* = \frac{\sum_{i=1}^N y_i \mu_i^*(x^*)}{\sum_{i=1}^N \mu_i^*(x^*)}$$

اگر بخواهیم روی شکل متوجه بشیم معنی این فرمول چه داریم:
 ما یک سری نقاط داریم که روشنون مجموعه فازی تعریف کردیم. اگر مجموعه فازی رو طوری تعریف کنیم که جمع تعلق ها در یک نقطه 1 بشه فقط صورت کسر میمونه و مخرج 1 میشه پس ساده

$$f(x) = \sum_{i=1}^N y_i \mu_{A_i}(x)$$



در مراکز مقدار تعلق برای اون مرکز 1 هستش برای بقیه 0 هستش پس تو مراکز خروجی میشه ولی در بقیه جاها میشه به همچین چیزی:



و میتوان نوشت برابر است با:

$$y = \mu_2 y_2 + (1 - \mu_2) y_3$$

ولی لزومی نداره که جمع مقدار تعلقا 1 بشه. اگه جمушون 1 بشه اون شکلی که بدست میاد قطعه قطعه خطی هستش ولی وقی جمушون 1 نباشه خطی نمیشه. وقی تعداد داده های آموزشی بیشتر باشه شکلمون دقیق تر بدست میاد حالا چه قطعه قطعه خطی باشه چه قطعه قطعه غیر خطی.



سترم حمزه ماتریس برای تقریب تابع (گرینون) رعن که عداین از نتایج تابع را داشته باشیم

$$\text{Mass Dot} \quad \text{ترکیب} \quad \mu_{R^i}(x, y) = \frac{\mu(x)}{\mu_{A^i}} \mu_{B^i}(y)$$

$$\mu_{A^i}(x) = \begin{cases} 1 & x = x^* \\ 0 & x \neq x^* \end{cases}$$

- $y^* = \frac{\sum_{j=1}^n \text{مکان} \times \text{تعلیم} \text{کل} \text{کل}}{\sum_{j=1}^n \text{مکان}}$ عنصر ماتریس سوپر گروزه مرکز

* در رابطه آخر نوشته شده سیگما تعلق در مرکز ضربیدر مرکز تقسیم بر سیگما تعلق مرکز

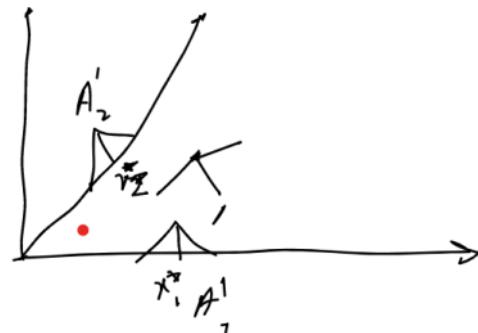
$$f(x) = \frac{\sum_{i=1}^N y_i \frac{\mu(x)}{A^i}}{\sum_{i=1}^N \mu(A^i)}$$

حالا فرض شود برای چند بعد بخوایم فرمول رو بنویسیم. اگر حاصل ضرب کارتزین رو با ضرب مدل کنیم:

- $f(x_1, \dots, x_n) = \frac{\sum_{i=1}^N y_i \prod_{j=1}^n \frac{\mu(x_j)}{A_j^i}}{\sum_{i=1}^N \prod_{j=1}^n \frac{\mu(x_j)}{A_j^i}}$

مکان ماتریس
دری عیل ام و ماتن نام

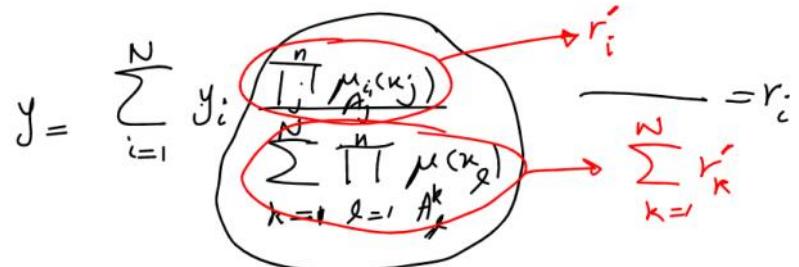
معنیش رو اگر بخوایم با شکل نشون بدیم:



حالا میتوان ایده های شبکه عصبی و سیستم فازی رو با هم قاطع کنیم یک شبکه عصبی فازی درست کنیم. به این صورت که:

$$y = \sum_{i=1}^N y_i \frac{\prod_{j=1}^n \mu_{A_j}^{(x_j)}}{\sum_{k=1}^N \prod_{l=1}^n \mu_{A_k}^{(x_l)}}$$

سپس داریم:

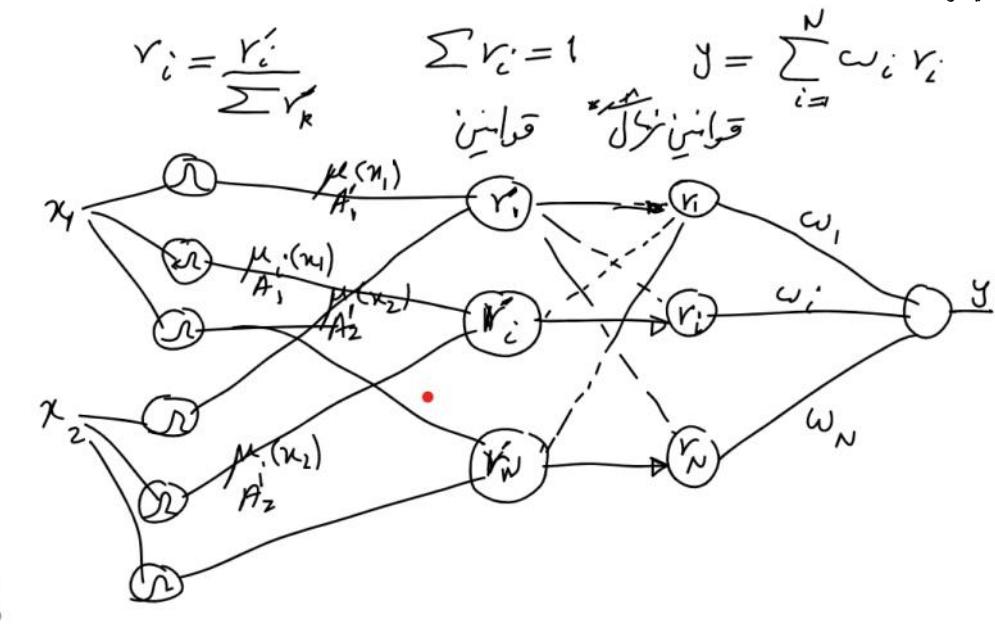


$$y = \sum_{i=1}^N y_i r'_i .$$

حالا فرض کنیم y_i ها هم در حالت شبکه عصبی بلد نیستیم و اسمشو میناریم: w_i

$$y = \sum_{i=1}^N y_i r'_i = \sum_{i=1}^N w_i r'_i$$

پس میتوانیم به این شکل درست کنیم که یک سری مجموعه فازی داریم برای هر بعد، اینا با هم حاصل ضرب کارتزین میشن قوانین رو درست میکنن. این قوانین نزمال میشن و در یک سری وزن ضرب میشن و خروجی رو میسازن:



فرق شبکه عصبی فازی که اینجا گفته شده با سیستم خبره فازی چی هست؟ فرقش اینکه:

- در سیستم خبره فازی ما w_i هارو بر اساس خروجی داده هایی که داشتیم بدست می آوردیم. ولی در شبکه عصبی فازی وزن ها یادگیری می شوند.
- مجموعه های فازی که در سیستم خبره فازی برای قوانین تعریف میکردیم بر اساس هر داده بود ولی در شبکه عصبی فازی مراکز قوانین رو هر طور که دوست داشتیم انتخاب میکنیم.

شبکه عصبی فازی 2 نوع یادگیری میتوانه داشته باشد:

1. مراکز رو باد بگیریم و وزن های خروجی رو
2. مراکز رو ثابت فرض کنیم و فقط وزن های خروجی رو بدست بیاریم.

به این نوع شبکه عصبی فازی میگیم TSK0 که خیلی شبیه به RBF هستش فقط این که در RBF ما نرمال سازی نداریم اینجا داریم.

$$y = \sum w_i r_i(x) = w^T R(x) \quad TSK_0$$

$$y = w^T R(x)$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}$$

$$y^l = w^T R(x^l)$$

$$R(x) = \begin{bmatrix} r_1(x) \\ r_2(x) \\ \vdots \\ r_N(x) \end{bmatrix}$$

در تصویر بالا مشاهده میکنیم که چطوری میشه رابطه رو به شکل برداری نوشت. اعمد داره رابطه رو برای داده L ام میگه. یعنی اگر M تا داده داشته باشیم:

$$\left\{ \begin{array}{l} y^1 = R^T(x^1)w \\ y^2 = R^T(x^2)w \\ \vdots \\ y^M = R^T(x^M)w \end{array} \right. \quad Y = \bar{R}w$$

$$\bar{R} = \begin{bmatrix} R^T(x^1) \\ R^T(x^2) \\ \vdots \\ R^T(x^M) \end{bmatrix}$$

اگر y واقعی رو \hat{y} در نظر بگیریم مجموع مرباعات خطأ رو داریم:

$$E = \frac{1}{2} (y - \hat{y})^T (y - \hat{y})$$

$$\downarrow \bar{R}w$$

$$\frac{\partial E}{\partial w} = \bar{R}^T (y - \hat{y}) = \bar{R}^T \bar{R}w - \bar{R}^T \hat{y} = 0$$

$$w = (\bar{R}^T \bar{R})^{-1} \bar{R}^T \hat{y}$$

اما بعضی موقع ها ممکنه اون ترمی که معکوسش کردیم full rank و معکوس پذیر نباشه پس میایم یک ترمی بهش اضافه میکنیم:

$$E = \frac{1}{2} (\mathbf{y} - \hat{\mathbf{y}})^T (\mathbf{y} - \hat{\mathbf{y}}) \rightarrow \mathbf{w}^T \mathbf{w}$$

$$\frac{dE}{dw} = \bar{\mathbf{R}}^T (\mathbf{y} - \hat{\mathbf{y}}) = \bar{\mathbf{R}}^T \bar{\mathbf{R}} \mathbf{w} - \bar{\mathbf{R}}^T \hat{\mathbf{y}} = 0$$

$$\mathbf{w} = (\bar{\mathbf{R}}^T \bar{\mathbf{R}})^{-1} \bar{\mathbf{R}}^T \hat{\mathbf{y}}$$

$$\mathbf{w} = (\bar{\mathbf{R}}^T \bar{\mathbf{R}} + \lambda I)^{-1} \bar{\mathbf{R}}^T \hat{\mathbf{y}}$$

یک راه دیگش هم گرادیان گرفتن هستش:

$$\mathbf{y}^l = \sum_{i=1}^n w_i r_i^l$$

$$E = \frac{1}{2} \sum_{l=1}^M (\mathbf{y}^l - \hat{\mathbf{y}}^l)^2$$

$$\boxed{\Delta \mathbf{w} = -\eta \sum_{l=1}^M (\mathbf{y}^l - \hat{\mathbf{y}}^l) r_i^l}.$$

یک از روش هایی که میشه مراکز رو تعیین کرد روش خوشه بندی هستش.

K-Means & C-means

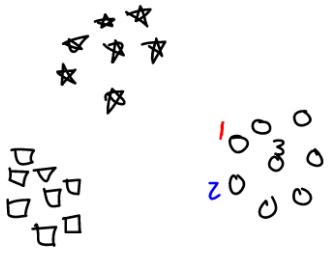
فرض کیم یک سری داده داریم:



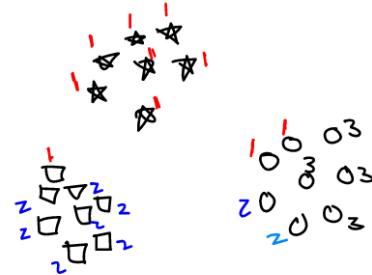
فرض میکنیم $k=c=3$

* یکی از مشکلات K-means و C-means اینه که تعداد خوشه هارو باید خودمون انتخاب کنیم. اگر این تعداد رو خوب تعیین نکیم ممکنه خوب جواب نده (مخصوصا اگر داده ها تو هم باشن) هرجند که اگر تو هم نباشن جوابش معمولا خوب نیشه.

فرض میکنیم ابتدا به صورت تصادفی 3 تا مرکز رو انتخاب میکنیم. برای این که نشان دهیم وقتی این 3 تا داده از هم جدا باشن به راحتی جواب میده در این مثال هر 3 تا مرکز رو بد انتخاب میکنیم:



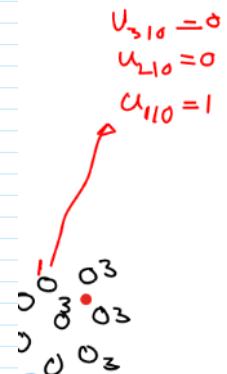
بعد نگاه میکنیم هر داده به هر مرکز که هستش لیبل داده رو با لیبل مرکز یک در نظر میگیرم:



حالا که لیبل زدیم برای داده ها، میایم مقدار تعلق هم اینطوری در نظر میگیریم:

$$u_{ik} = \begin{cases} 1 & X_k \in C_i \\ 0 & X_k \notin C_i \end{cases}$$

برای مثال اگر یکی از داده ها رو داده دهم در نظر بگیریم داریم:



حالا اگر مرکز داده های کلاس 1 رو بخوایم محاسبه کنیم :

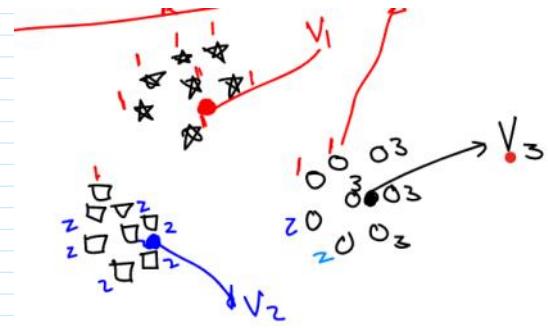
$$V_1 = \frac{\sum_k u_{1k} X_k}{\sum_k u_{1k}}$$

مخرج کسر که میشه تعداد داده هایی که لیبل 1 دارن صورتش هم میشه جمع مقدار اونایی که لیبل 1 دارن.

در حالت کلی هم داریم:

$$V_i = \frac{\sum_k u_{ik} X_k}{\sum_k u_{ik}}$$

حالا اگر به صورت شهودی مرکز هارو نشون بدیم یه همچین چیزی میشن:



حالا که مرکز جدید تعیین شد بر اساس مرکز جدید لیبل های جدید درست میکنیم و اما هارو عوض میکنیم:



یک 2 بار دیگه تکرار کنیم دیگه لیبل های داده ها دقیق معلوم میشه. به این روش میگیم C-means و K-means (تو کامپیوتر تا وقتی تکرار میشه که یاها ها تغییراتشون از یک حد آستانه کمتر بشه یا مرکز تغییراتشون از یک حد آستانه کمتر بشه)

اگر بخواهیم مساله خوش بندی رو به شکل یک مساله بهینه سازی بنویسیم:
تابع هدفش چه میشه (چه خوش بندی ای خوش بندی خوبی هست)? داده های داخل یک خوشة تا حد امکان به مرکز خوشه نزدیک باشند. به زبان ریاضی یعنی:

$$E = \sum_k \sum_c u_{ik} \|x_k - v_i\|^2$$

ما میخواهیم این را مینیمم کنیم. آیا میشه مشتق بگیریم برابر با صفر بداریم؟ نه. چون اما هامون 0 و 1 هستن و گرسنه اند نمیشه مشتق گرفت.

پس میایم اما هارو فازی فرض میکنیم. یعنی مقداری بین 0 و 1 در نظر بگیریم. این از نظر شهودی اشکالی ندارد. چرا؟ بر فرض مثال زیر رو در نظر بگیرید:



داده که دورش رو قرمز کشیدیم چرا باید تعلق 1 به قرمز داشته باشه تعلق 0 به سیاه یا بالعکس؟ فاصله اش از 2 مرکز تقریبا مساویه و میشه گفت 0.5 تعلق داره به سیاه 0.5 به قرمز
با این کار اما ها پیوسته و مشتق پذیر میشن.

اگر نسبت به اما مشتق بگیریم:

$$\frac{\partial E}{\partial u_{ik}} = \|x_k - v_i\|^2 = 0$$

$$v_i = x_k$$

که اینم امکان پذیر نیست چون نمیتوانیم برای هر داده یک مرکز داشته باشیم. پس اینم مشکل داره و باید تابع هدف تغییر پیدا کنه:

$$E = \sum_i \sum_k u_{ik}^m \|x_k - v_i\|^2$$

که مشتق بگیریم:

$$E = \sum_i \sum_k u_{ik}^m \|x_k - v_i\|^2$$

$$\frac{\partial E}{\partial u_{ik}} = m u_{in}^{m-1} \|x_k - v_i\|^2 = 0$$

• $u_{ik} = 0 \quad x_k = v_i$

بازن این جواب به درد نمیخوره. هرچند که u_{ik} رو بذاریم جواب درستی هست و مینیمم میکنه ولی به درد بخور نیست. اگر اون 2 تا تم هم برابر بذاریم باز مینیمم میشه ولی باز به درد ما نمیاد.
برای این که یه کاری بکنیم که u_{ik} ها صفر نشن باید یک محدودیت بذاریم که نتوانه صفر بکنه.

$$\sum_{i=1}^c u_{ik} = 1$$

$k=1, \dots, N$

با این محدودیت اگر یک x_1 داشته باشیم با 2 تا خوشه. اگر برای یک خوشه بشه 0.4 برای اون یکی باید بشه 0.6:

$$x_1 \quad ①$$

$$u_{11} = 0.4 \quad ②$$

$$u_{21} = 0.6$$

حالا جلوتر میبینیم این محدودیت چه مشکلی ایجاد میکنه ولی فعلا با این محدودیت بریم جلو بینیم چی میشه.

روش لگرانژ

$$E = \sum_{k=1}^N \sum_{i=1}^c u_{ik}^m \|x_k - v_i\|^2 - \sum_{k=1}^N \lambda_k \left(\sum_{i=1}^c u_{ik} - 1 \right)$$

توی روش لگرانژ میگه تابع هدف رو منهای یک سری ضرایب لگرانژ ضریر محدودیت ها بکنیم. محدودیت برای درست کردن محدودیت ها هم هر رابطه که داریم همو رو بیریم یک سمت مساوی صفر بکنیم و اون تری که سمت چپ تساوی هست رو میاریم. برای محدودیت ما این شکلی میشد:

$$\sum u_{ik} - 1 = 0$$

در روش لگرانژ یک بار نسبت به متغیرها و یکبار نسبت به ضرایب لگرانژ مشتق میگیریم. سپس 3 تا معادله بدست میاد که باید تو شون بایام ضرایب لگرانژ رو حذف کنیم و یک سری رابطه بدست بیاریم. که جلسه بعد این کار رو انجام میدیم

$$\frac{\partial E}{\partial u_{ik}} = 0 \quad \frac{\partial E}{\partial v_i} = 0 \quad \frac{\partial E}{\partial \lambda_k} = 0$$

طبق جلسه پیش داریم:

$$E = \sum_{k=1}^N \sum_{i=1}^C u_{ik}^m \|x_k - v_i\|^2 - \sum_{k=1}^N \lambda_k \left(\sum_{i=1}^C u_{ik} - 1 \right)$$

حالا اگر نسبت به u_{ik} مشتق بگیریم چون هم i مشخصه هم k مشخصه دیگه برای بقیه ترم ها صفر میشه پس سیگما حذف میشه و داریم:

$$\frac{\partial E}{\partial u_{ik}} = m u_{ik}^{m-1} D_{ik} - \lambda_k = 0$$

با مساوی صفر قرار دادن داریم:

$$u_{ik}^{m-1} = \frac{\lambda_k}{m} \times \frac{1}{D_{ik}}$$

که با رادیکال به قوه $m-1$ نداریم:

$$u_{ik} = \left(\frac{\lambda_k}{m} \right)^{\frac{1}{m-1}} \frac{1}{D_{ik}^{\frac{1}{m-1}}}$$

اگر رابطه اصلی را نسبت به λ_k هم مشتق بگیریم داریم:

$$\frac{\partial E}{\partial \lambda_k} = \sum_{i=1}^C u_{ik} - 1 = 0 \rightarrow \sum_{j=1}^C u_{jk} = 1$$

با جایگذاری مقدار u_{ik} بدست آمده در مشتق اول در رابطه دوم داریم:

$$\left[\sum_{j=1}^C \left(\frac{\lambda_k}{m} \right)^{\frac{1}{m-1}} \frac{1}{D_{jk}^{\frac{1}{m-1}}} = 1 \right] \quad (2)$$

و اسم رابطه بدست آمده در مشتق اول هم مینذاریم: ۱

$$u_{ik} = \left(\frac{\lambda_k}{m} \right)^{\frac{1}{m-1}} \frac{1}{D_{ik}^{\frac{1}{m-1}}} \quad (1)$$

سپس داریم:

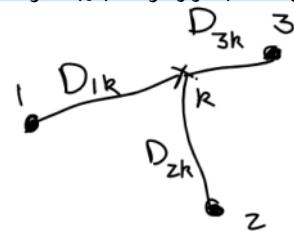
$$\left(\frac{\lambda_k}{m}\right)^{\frac{1}{m-1}} \times \frac{1}{D_{ik}^{\frac{1}{m-1}}} = u_{ik} \quad (1)$$

$$\left(\frac{\lambda_k}{m}\right)^{\frac{1}{m-1}} \sum_{j=1}^c \frac{1}{D_{jk}^{\frac{1}{m-1}}} = 1 \quad (2)$$

$\frac{(1)}{(2)}$

$$\frac{\frac{1}{D_{ik}^{\frac{1}{m-1}}}}{\sum_{j=1}^c \frac{1}{D_{jk}^{\frac{1}{m-1}}}} = u_{ik}$$

پس مقدار تعلق داده k به مرکز خوشة i با رابطه فوق حاصل می شود. معنیش چی میشه؟



حالا اگر بخوایم u_{3k} را بدست بیاریم:

$$u_{3k} = \frac{\frac{1}{D_{3k}}}{\frac{1}{D_{1k}} + \frac{1}{D_{zh}} + \frac{1}{D_{2k}}}$$

در رابطه فوق $m-1$ روندیده گرفتیم چون موقع تابع هدف به توان m رسونده بودیم. حالا نکته ای که میخوایم بهش برسیم اینه که u_{ik} میشه عکس فاصله تقسیم بر مجموع عکس فاصله ها، با این کار وقی u_{ik} ها با هم جمع بشن صورت و مخرج یکی میشه و برابر با 1 میشن و نرمال میشن (همون محدودیتی که اعمال کردیم).

حالا باید فرمول نسبت به مرکز چی میشه:

$$E = \sum_{i,k} u_{ik}^m (x_k - v_i)^\top (x_k - v_i) \dots$$

$$\frac{\partial E}{\partial v_i} = -2 \sum_k u_{ik}^m (x_k - v_i) = 0$$

$$\sum_k u_{ik}^m x_k = v_i \sum_k u_{ik}^m$$

$$v_i = \frac{\sum_{k=1}^N u_{ik}^m x_k}{\sum_{k=1}^N u_{ik}^m}$$

که مخالف ۱ هستش چه نقشی دارد؟ معمولاً یک عددی بزرگتر از ۱ تا ۲ در نظر میگیریم. هرچی توان از ۱ بیشتر میشه عدد تعلق رو به ۰ نزدیک تر میکنه و حالت فازی رو کمتر میکنه و هرچی به ۱ نزدیک تر میشه **میزان فازی بودن** بیشتری خواهیم داشت.

$$① u_{ik} = \frac{\left(\frac{1}{\|x_k - v_i\|} \right)^{\frac{2}{m-1}}}{\sum_{j=1}^c \left(\frac{1}{\|x_k - v_j\|} \right)^{\frac{2}{m-1}}}$$

$$② v_i = \frac{\sum_{k=1}^N u_{ik}^m x_k}{\sum_{k=1}^N u_{ik}^m}$$

ولی مشکل اینجوری شد که داستان مرغ و تخم و مرغ میشه. برای بدست آوردن u_{ik} به v_i نیاز داریم و بالعکس. پس مثل مراکز اولیه روتصادف بدست میاریم بعد تا هارو بدست میاریم بعد ۷ هارو بروز میکنیم و انقدر تکرار میکنیم که از یه حد آستانه کمتر بشه. مراحل به گفته‌ی استاد:

FCM

- ۱- مراکز اولیه عبارت هستند از میزان راه‌ها استجاب کردن
- ۲- با استفاده از فرمول ① u_{ik} مقدار تعلق محاسبه می‌شوند
- ۳- با استفاده از متریک ② مراکز حبیب می‌شوند
- ۴- در مرحله ③ مراکز گردید

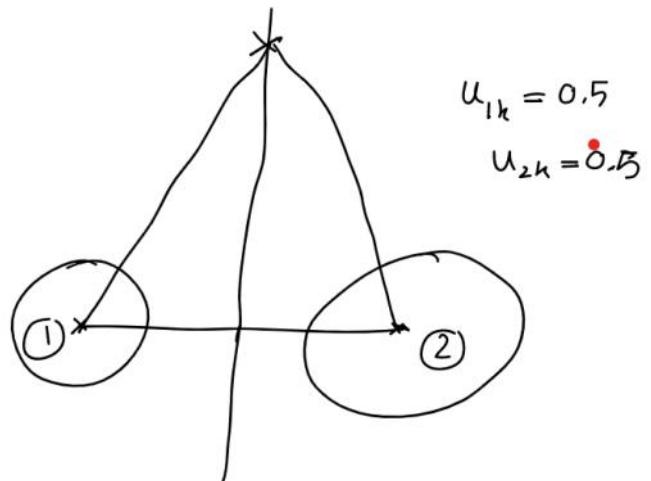
حالا بایم به معایبیش بپریم:

۱. هزینه محاسباتی
۲. مراکز اولیه حساس \leftarrow همگرای به مینیمم محلی
۳. انتخاب تعداد خوششها
۴. حساسیت به داده‌های پرت (Outlier)

مزایا:

- همگرای

توضیح مورد ۴ ام:



فاصله این داده به مرکز 1 و 2 یکسانه پس باعث میشے میزان تعلق 1 بشه در صورتی که اصن تعلق نباید داشته باشه و این شرط که ما گفتیم مجموعش باید 1 باشه باعث شد که این مشکل به وجود بپار.

برای حل این مشکل یک سری نسخه FCM اومدن محدودیت رو عوض کردن یا یک سری محدودیت جدید اضافه کردن ولی این کار هزینه محاسباتی رو زیاد میکنه. بهترین راه حل اینه که بایام داده های پرت رو حذف کنیم و از FCM استفاده کنیم.

چجوری داده های پرت رو تشخیص بدیم؟ 2 تا کار میشه کرد:

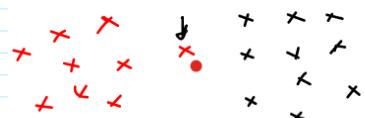
1. اگر یک داده فاصله اش از همه مرکز از یک حد آستانه بیشتر بود حذف کنیم

2. واریانس 2 داده رو بدلست بیاریم. میتوانیم یک داده رو فاصله اش با نزدیک ترین داده اش حساب کنیم و اگر از یک حد آستانه بیشتر بود حذف کنیم. روش اول بهتره چون هزینه محاسباتی کمتری دارد.

چرا داده های پرت بد هستن؟ مرکز رو میکشن به سمت خودشون و از مرکز واقعی دور میکن.

ما خوش بندی FCM رو انجام دادیم. چطوری بفهمیم هر داده متعلق به کدام خوشه است؟ هر کدام تعلق بیشتری داشت میگیم به اون خوشه داده تعلق داره

مزیت FCM نسبت به c-means چیه؟ روش FCM مرکز بهتری رو تعیین میکنه. در هر iteration ما یک سری اطلاعات رو حذف میکنیم



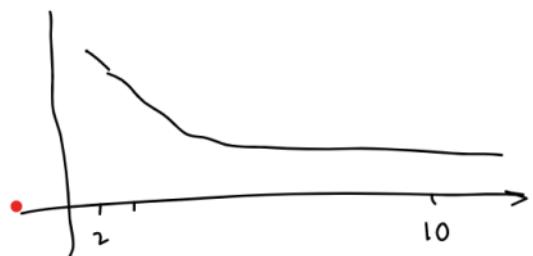
ما وقتی داده مرکزی رو به یک میدیم به اون یک نمیدیم تاثیرش بر هردو مرکز تاثیر میداره و مرکز قرمز رو به سمت خودش میکشه و سیاه هم هیچ کششی به خودش نمیداره و باعث میشه هیچ کدام دقیق تعیین نشن. ولی وقتی مقدار فازی در نظر بگیریم برای هردو دقیق تر معلوم میشه.

حالا چجوری تعداد خوشه هارو تعیین کنیم؟ یک سری معیار های کارایی داریم که بستگی به کاربرد یکیش انتخاب میشه:

- آنتروپی: هرجه u_{ik} ها به 0 و 1 نزدیک تر بشن خوش بندی بهتر و هرجی به $1/c$ نزدیک تر بشن بدتره.

$$-\sum_{k=1}^N \sum_{i=1}^c u_{ik} \log u_{ik}$$

این معیار به ازای c های مختلف رسم میشه و منحنی بدست میاد:



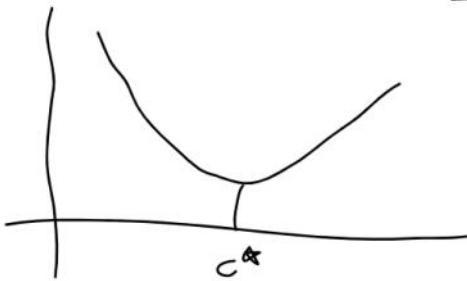
ما از طرف نمیخوایم تعداد خوشه ها زیاد بشه چون هرجی بیشتر بشه و ما بیشتر تمایل به این پیدا کنیم که به ازای هر داده یک خوشه بگیریم و آنتروپی صفر میشه. پس میایم از جایی که شروع میشه شب نمودار کم میشه رو به اندازه تعداد خوشه در نظر میگیریم.

$$\frac{1}{C} \text{trac}(U^T U)$$

یه جواری شبیه همون آنژویی میشه فقط لگاریتم نداره.
• معیار خطای درون گروهی و برون گروهی:

$$E = \sum_{k=1}^N \sum_{i=1}^c u_{ik}^m \|x_k - v_i\|^2 - \alpha \sum_{i=1}^c \|v_i - \bar{x}\|^2$$

$$- \sum_{i=1}^c \sum_{j \neq i} \|v_i - v_j\|^2$$



در اینجا 2 به 2 فاصله مرکزaro مقایسه میکنیم و میخوایم فاصله شون ماکسیمم شه. نمودار رو استاد مطمئن نیست ولی گفت احتمالا همچنین شکلی بدست مید که مینیمم ش میشه اون تعداد خوشه ها.

جلسه بیست و یکم (انتها فازی و ابتدای تکامل)

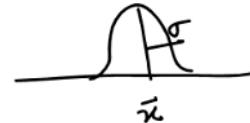
اگر ما با FCM خوش بندی رو بکنیم یک سری مرکز v_i داریم و یک سری u_{ik} که همه آنها را میشه به صورت یک ماتریس نوشت که سطر i و ستون k اش بشه u_{ik} .



$$v_i \quad u_{ik}$$

$$U = \begin{bmatrix} & & \\ & u_{ik} & \\ & & \end{bmatrix}$$

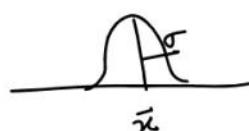
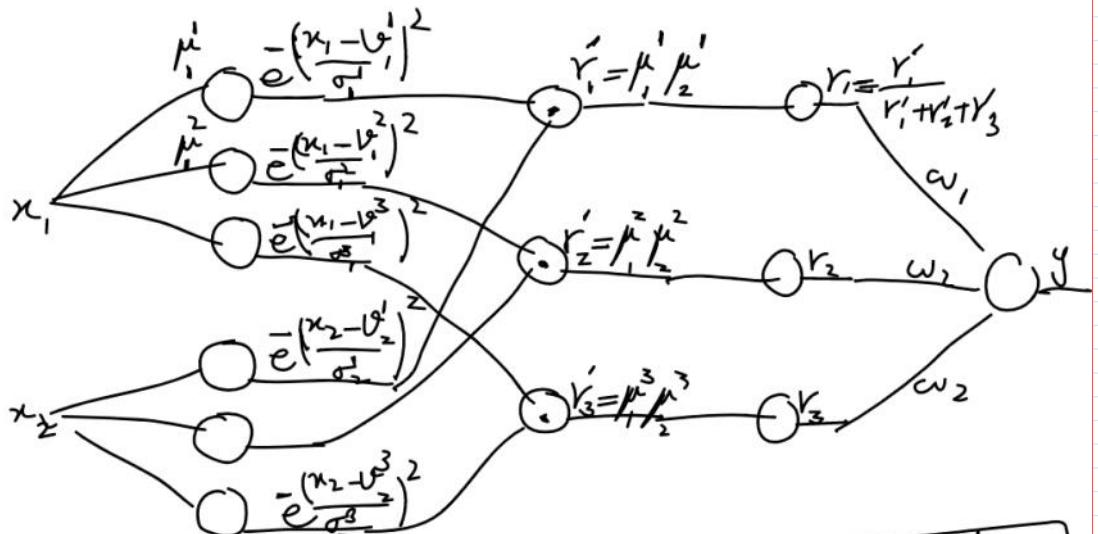
ابتدا برای شبکه عصبی فازی TSK: در شبکه عصبی فازی ما قانون هامون به این شکل هستش: مجموعه های فازی را میخواهیم به صورت قوسی داشته باشیم. در نتیجه یک مرکز نیاز داریم و یک سیگما:



مرکز هارو که مرکز خوش تعیین میکنه، در نتیجه خروجی پرسپترون ها رو در صورتی که مراکز برابر با

$$v_1 = \begin{bmatrix} v_1^1 \\ v_1^2 \\ v_1^3 \end{bmatrix} \quad v_2 = \begin{bmatrix} v_2^1 \\ v_2^2 \\ v_2^3 \end{bmatrix} \quad v_3 = \begin{bmatrix} v_3^1 \\ v_3^2 \\ v_3^3 \end{bmatrix}$$

باشند داریم:



$$y = R w \quad W \in \mathbb{R}^{T \times T^*}$$

$$E = \frac{1}{2} (y - y^*)^T (y - y^*)$$

حالا چیزی که میمونه اینه که سیگماها چطوری بدست میان.

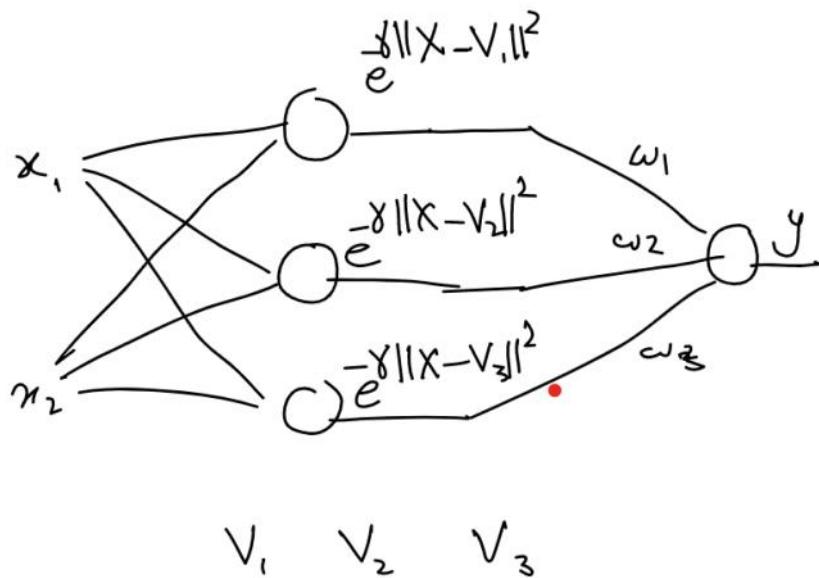
طبق فرمول داریم:

$$\sigma_j^2 = \frac{1}{N-1} \sum_{k=1}^N (x_{kj} - \bar{x}_j)^2$$

ولی ما برای یک خوش بینیم چی میشه. برای یک خوش داریم:

$$(\sigma_j^i)^2 = \frac{\sum_{k=1}^N u_{ik} (x_{kj} - v_j^i)^2}{\sum_{k=1}^N u_{ik}}$$

حالا اگر بخوایم از همین روش خوش بندی برای RBF بخوایم استفاده کنیم:



در حالت ساده اش ما مراکز که V_1 تا V_3 است رو بدهست میاریم تموم میشه

تو حالت پیچیده تریش ما میتونیم Base function هارو اینطوری بنویسیم:

$$e^{-\gamma (x - V_i)^T \sum_{j=1}^{n-1} (x - V_j)}$$

یا به طور کل تر برای i ام میتوانیم اینطوری بنویسیم:

$$e^{-\gamma (x - V_i)^T \sum_{j=1}^{n-1} (x - V_j)}$$

در حالت ساده ما یک سری دایره، کره یا ابر کره بسته به ابعاد داریم. در حالت بهبود یافته ما ناحیه بیضی چرخیده و بیضی های مورب و در حالت n بعدی ابر کره های هم بسته و مورب داریم.

V_i که شد مرکز خوش ها. سیگما چطوری بدست میاد؟ :

$$\sum_i = \frac{\sum_{k=1}^N u_{ik} (x_k - v_i) (x_k - v_i)^T}{\sum_{k=1}^N u_{ik}}$$

دترمینانس کوواریانس نام

ماتریس کوواریانس ماتریسی هستش که قطرهای اندیکاتور معیاره از بعد اول تا بعد n ام.

$$\Sigma = \begin{pmatrix} \sigma_i^2 & c_{ij} & c_{in} \\ c_{ji} & \sigma_j^2 & c_{jn} \\ c_{ni} & c_{nj} & \sigma_n^2 \end{pmatrix}$$

همستگی بعد عده زام

$$c_{ij} = p_{ij} \sigma_i \sigma_j$$

معنی همبستگی: وقتی بعد ام افزایش پیدا میکنه بعد ام چقدر افزایش پیدا کنه. اگر به همون نسبت افزایش پیدا کنه هم بستگیش 100 درصد میشه و اون زم میشه 1. اگر در جهت عکس به همون نسبت کم بشه میشه -1. بنابراین یک مقداری بین -1 و 1 میگیرد:

$$-1 \leq p_{ij} \leq 1$$

در برخی از نسخه های FCM هم به جای فاصله اقلیدسی به نحو زیر از فاصله ماهانوبیس استفاده میشه:

$$(x - v_i)^T (x - v_i) = \|x - v_i\|^2$$

فاصله اقلیدس

$$\|x - v_i\|_S^2 = (x - v_i)^T \sum_i^{-1} (x - v_i)$$

فاصله ماهانوبیس

تمام

تمام: تطابق با محیط
هدف نهایی تمام: بقای زن برتر
جزئی تمام: محدودیت منابع

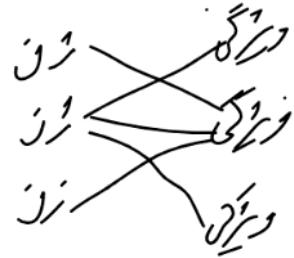
برای این که موجودات زنده بمومن و تولید مثل بکن به یک سری منابع نیاز دارند که محدود است. در نتیجه یک رقابت بین موجودات برای بدست آوردن منابع به وجود می‌آید و در این رقابت موجوداتی که بتوانند منابع را بدست بیاورند پیروز و باقی از بین میروند. در نتیجه موجودات برای پیروزی در این رقابت نیاز دارند خود را با محیط تطبیق بدهند. تطبیق با محیط خودش به 2 شکل صورت می‌گیرد:

- به صورت جنگیدن با بقیه
- منبعی انتخاب شود که تقاضا برای آن کم باشد و به راحتی بدست آید. مثل داستان کرگدن و زرافه. زرافه با بلند کردن قد خودش با تکامل از برگ‌های بالاتر تغذیه می‌کند و کرگدن با برگ‌های پایین و دیگر دعواپی سر منبع نداریم.

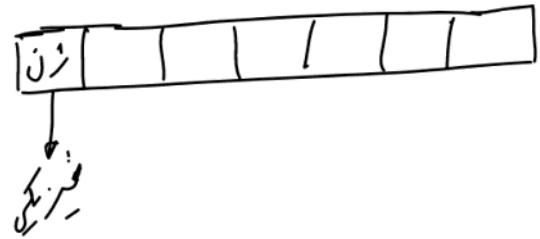
چگونه تکامل اتفاق میافته: یک بحث میکروسکوپی میشه کرد یک بحث میکروسکوپی اش به این صورته که یک موجود خود رو که نمیتونه تغییر بد. فرزندانش رو میتونه عوض کنه پس در تکامل تولید مثل خیلی مهمه. چون تولید مثل میتواند ویژگی هارو عوض کنه. اگر تولید مثل وجود نداشته باشه ویژگی ها عوض نمیشه.

چجوری این اتفاق میافته که فرزندان از والدین میتوونن بهتر بشن؟ اینو باید از نظر میکروسکوپی بررسی کرد. همه موجودات ویژگی‌های فیزیکشون از ژن هاشون میاد. در نتیجه هر موجود زنده رو میشه با یک نقشه ریتمیک معرفی کرد. ما در موجود زنده تعداد زیادی کروموزوم داریم که داخلشون ژن‌ها قرار میگیرن. ولی ما فرض میکنیم برای سادگی یک کروموزوم داریم که تمام ژن‌ها در آن باشند:

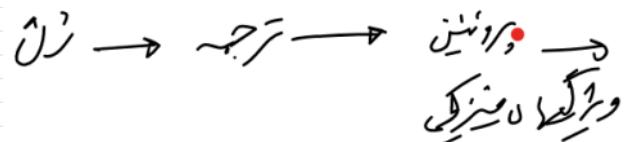
هر ژن یک یا چند فیزیکی رو تحت تاثیر قرار میده. در نتیجه رابطه ژن و رابطه فیزیکی چند به چنده:



ولی در حالت ساده و در الگوریتم تکاملی فرض میکنیم این رابطه یک به یکه. یعنی هر ژن یک ویژگی فیزیکی مثل قد و رنگ چشم رو در بر میگیرن.

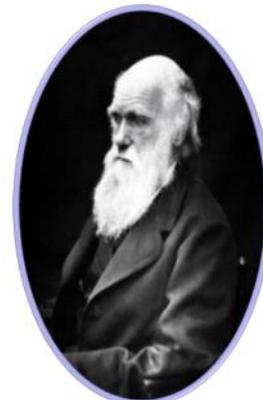


ژن‌ها کد هستند. این ژن‌ها در سلول ترجمه و **Decode** میشوند و یک پروتئین میشوند که این پروتئین‌ها ویژگی فیزیکی رو تعیین میکنند.



تئوری تکامل

تولد: 1809
وفات: 1882
منشا موجووات: 1859



تکامل توسط داروین معرفی شد.

تئوری تکامل

تکامل

شایستگی بالاتر فرزندان نسبت به والدین

درزگاهای فیزیکی مناسبتر تطابق پیشرت با محیط

تولید فرزندان شایسته پیشرت

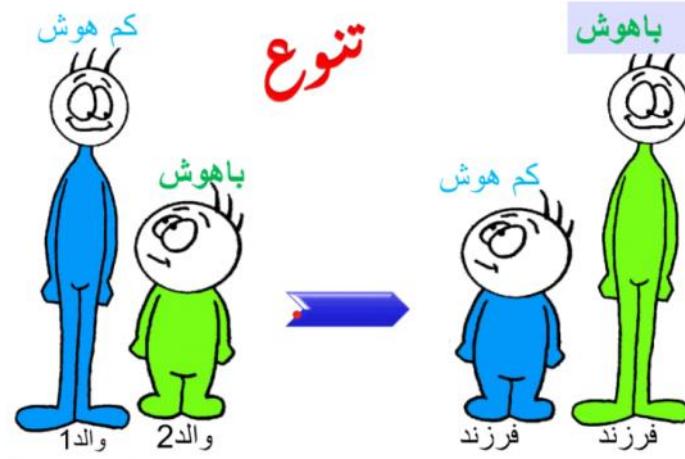
تکامل معانی مختلفی دارد. که در اسلاید مشخص شده.

انتخاب طبیعت

قانون بقای شایسته ترتیبها

منظور از شایسته ترتیب ها کسایی هست که میتوانن تولید مثل کنن و از منابع به خوبی استفاده کنن.

تئوری تکامل



با وجود 2 والد که یک کم هوش و یک با هوش و قد کوتاه هستند فرزندان 4 حالت میتوانند داشته باشند. انتخاب است که فرزند با هوش و قابلند طوفدار بیشتری دارد. پس میتوانه زوج ها و فرزندان بیشتری داشته باشند.

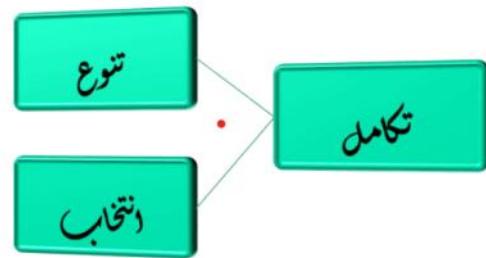
تئوری تکامل



ولی فرزند کم هوش و قد کوتاه کسی سمتش نمیره اگر بره هم خواهان و فرزندان کمتری به لحاظ آماری دارد.

تئوری تکامل

تصادف



برای تنوع تصادف هم کمک میکنے تنوع داشته باشیم
انتخاب باعث میشه موجودات با شایسته انتخاب بین و متوسط شایستگ رو میره بالا. ولی خود تنوع متوسط شایستگ رو میره پایین. با این که این 2 ویژگ در تضاد هستن باید جفت‌شون باشن تا تکامل به وجود بیاد.
به دلیل تضاد باید این 2 یک تعادل باشه و به همین دلیل تکامل گند هستش و باید در رنج میلیارد سال نگاه کرد بهمیش.

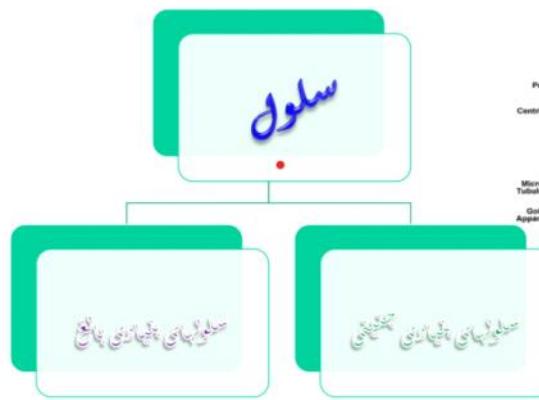


حالا میخوایم ببینیم

وزیری کیمی و لین چگونه به فرشان منقاد می شوند؟

از دید میکروسکوپی هر موجود زنده یک سری تک سلول اولی زیگوت میگن. زیگوت با تقسیم سلولی خود بسته به این که موجود تک سلولیه یا پر سلولی سلول های دیگر رو به وجود میره. خود زیگوت از لقاح اسپرم و تخمک به وجود میاد که اسپرم از موجود تر و تخمک از موجود ماده میاد. این اسپرم و تخمک با لقاح خود یک تک سلولی زیگوت رو درست میکنند و زیگوت با تقسیم سلولی موجودات پر سلولی مثل گیاهان و انسان ها و حیوانات رو به وجود میره.

تئوری تکامل



سلول خودش به دو دسته که بنیادی و معمولی تقسیم میشند. سلول بنیادی هم 2 تا داره جنبی و بالغ.

زیگوت اولین سلول بنیادی جنیف هستش که با تقسیم سلول تبدیل میشند به یک سری سلول های بنیادی جنیف بعد سلول های بنیادی جنیف به یک تعدادی که رسیدن باز با تقسیم سلولی تبدیل میشند به سلول های بنیادی بالغ مثل مغز و استخوان با تخصصی مثل پشم و گوشت و بوست و ...

فرق بین سلول های بنیادی جنیف و بالغ: سلول های بنیادی جنیف به هر نوع سلولی مثل سلول های بنیادی بالغ و تخصصی میتوانند تبدیل بشون ولی سلول های بنیادی بالغ فقط به یک دسته خاصی از سلول های بنیادی میتوانند تقسیم بشن.

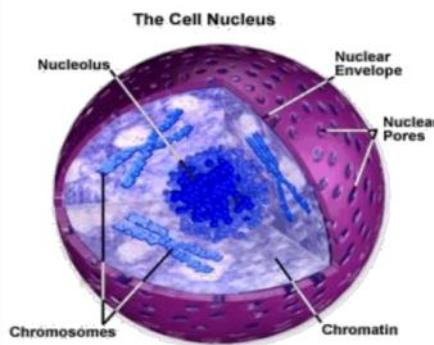
بنابرین وقتی دستمون رخم میشند، به دلیل سلول های بنیادی بالغ که در مغز و استخوان هستن میکنند و سلول های تخصصی پوست و گوشت رو درست میکنند. ولی اگر نخاع قطع بشه چون دیگه سلول های بنیادی جنیف وجود ندارند (بعد یک مدقن که نوزاد به دنیا میاد سلول های بنیادی جنیف تومی میشون) و فقط سلول های بنیادی بالغ و تخصصی دارند. سلول های بنیادی بالغ هم به هر نوع سلولی نمیتوانند تبدیل بشون و سلول بنیادی بالغ ندارند که بتوونند نخاع تولید کنند و اگر قطع بشه دیگه ترمیم بذری نیست.

به همین دلیل میان جدیدا سلول های بنیادی جنیف رو در یک باکتری های نگهداری میکنند که اگر یک زمانی یک فردی مشکل قطع نخاع بیدا کرد با همین سلول های بنیادی جنیف مشکل قطع نخاع رو حل کنن (فقط یک فرد برآخودش رو میتوانه).

البته برای حیوانات تونستن یک سری سلول های بنیادی جنیف تولید بکن و در آینده برای انسان ها هم بتوون تولید کنن (ممکنه مخفیانه الان وجود داشته باشد). چون باید زیگوت هم اسیر و تخمک مشابه نیاز هست که میشه همانند سازی. برای انسان که همانند سازی کردن زیگوت بعد چند هفته میمیره ولی برای حیوانات تونستن انجام بدن و گوسفند درست کنن. البته میتوان یک سری تغییرات بدن در DNA که مثلا گوسفند شیرش انسولین درست کنند و یک کارخونه انسولین سازی درست میکنند.

تئوری تکامل

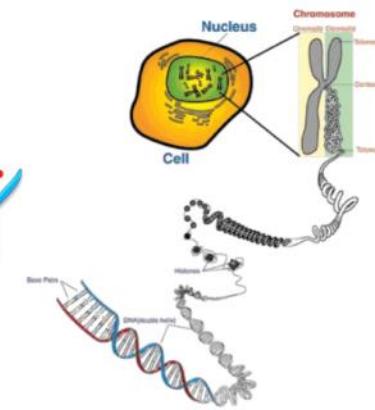
هسته



اون چیزی که ما در هسته کار داریم کروموزوم هستش که در خود کد های (زن) ذخیره میکنند که ویژگی موجود زنده رو تعیین میکنند.

تئوری تکامل

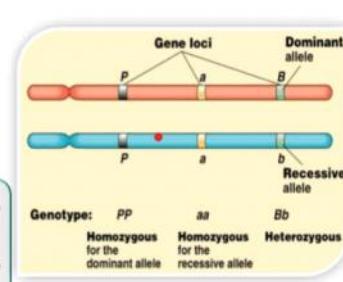
ژنوم
کروموزوم



به مجموعه ژن ها ژنوم میگن.

تئوری تکامل

• خالص
• ناخالص
• ژن



- غالب
- مغلوب

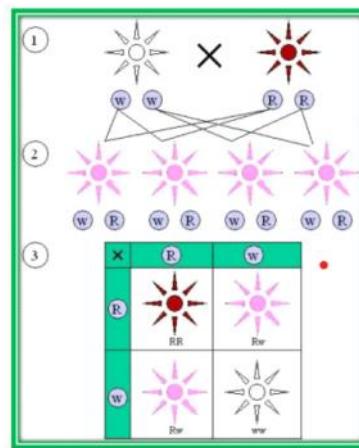


کروموزوم ها یک جفت هستند. اگر در یک جا از کروزوم مثلا رنگ چشم رو مشخص کنند در همون جا در کروزوم دیگه از اون جفت هم رنگ چشم رو مشخص میکنند. اگر هردو مثلا رنگ سیاه رو مشخص کنند بهش میگیم ژن خالص. اگر یک سیاه باشه اون یکی مثلا قهوه ای باشه میگیم ژن ناخالص.

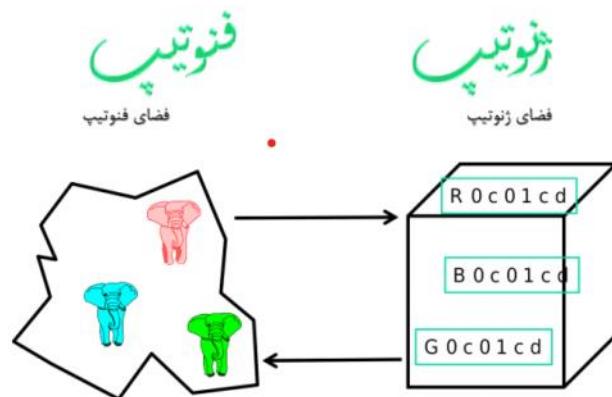
در الگوریتم های کروموزوم ها رو جفتی در نظر نمیگیریم و داستان خالص و ناخالص و غالب و مغلوب رو نداریم.

وقتی ناخالصی داریم مثل رنگ سیاه و قهوه ای اونی که تو موجود ظاهر میشه غالب اونی که نمیشه میگیم مغلوب. و این باعث میشه یک ژن مغلوب در نسل ها باقی بمانه بعد یهו ظاهر بشه و از انقرض نجات بدله.

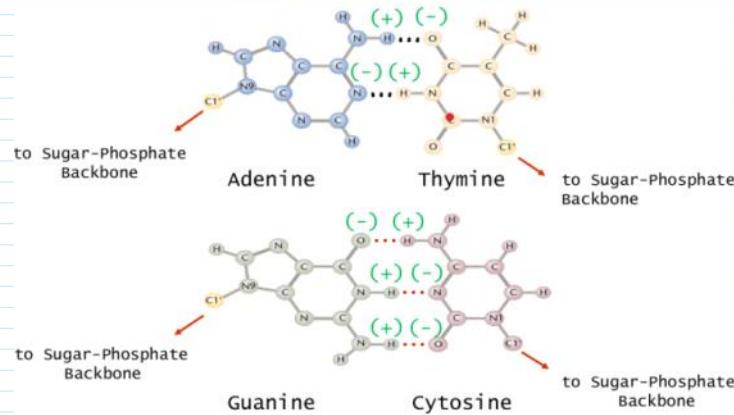
تئوری تکامل



تئوری تکامل

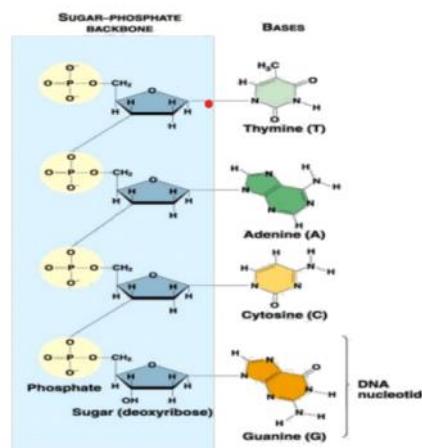


تئوری تکامل



هر مثل یک نرده بان هستش که پله هاش کد های ژنتیکی را ذخیره میکنن و این کد های ژنتیکی یک سری باز های شیمیایی هستن که در اسلايد میبینیم. این ها 2 به 2 مکمل هستن و پیوند شیمیایی میتوونن بقرار کنن و پله نرده بان ساخته بشه.
در نتیجه 4 تا کد ژنتیک داریم که میگیم A و G و C و T و از اسم اون باز های اوMD که پله های نرده بان را میسازه.

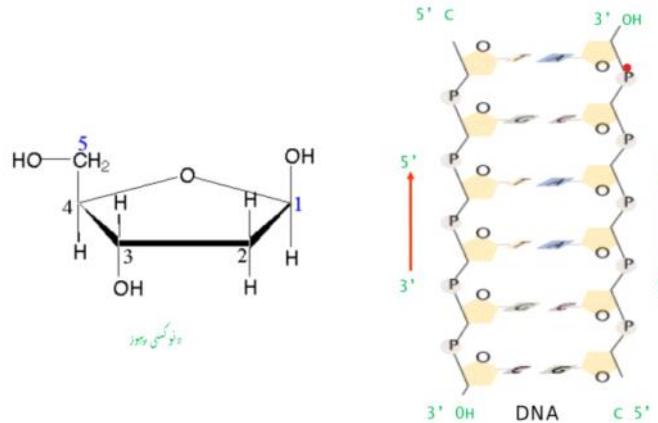
تئوری تکامل



- | | |
|------------------|--|
| ادنین: | (A) <input type="checkbox"/> |
| گوئین: | (G) <input type="checkbox"/> |
| سیتیوز: | (C) <input type="checkbox"/> |
| تیمین: | (T) <input type="checkbox"/> |
| بوراسیل: | (U) <input type="checkbox"/> |
| بروتون: | A-T <input type="checkbox"/>
G-C <input type="checkbox"/> |
| پلی امین: | A, G, T, C <input type="checkbox"/> |

قسمت عمودی نرده بان هم یک گروه کربن و یک گروه فسفات هستش که باهش کاري نداریم.

تئوری تکامل



چیزی که در این نرده بان اهمیت دارد برا ما این کد هایی که در هر پله هستن هستش چون کد ها زن هستن و ویژگی فیزیکی رو مشخص میکنند.

تئوری تکامل

	Second letter				
First letter	U	C	A	G	Third letter
U	UUU } Phe UUC } UUA } Leu UUG }	UCU } Ser UCC } UCA } UCG }	UAU } Tyr UAC } UAA Stop UAG Stop	UGC } Cys UGA Stop UGG Trp	U C A G
	CUU } CUC } Leu CUA } CUG }	CCU } CCC } Pro CCA } CCG }	CAU } His CAC } CAA } Gln CAG }	CGU } CGC } Arg CGA } CGG }	U C A G
	AUU } AUC } Ile AUU } Met	ACU } ACC } ACA } ACG }	AAU } Asn AAC } AAA } Lys AAG }	AGU } Ser AGC } AGA } Arg AGG }	U C A G
	GUU } Val GUC } GUA } GUG }	GCU } GCC } Ala GCA } GCG }	GAU } Asp GAC } GAA } Glu GAG }	GGU } GGC } Gly GGA } GGG }	U C A G

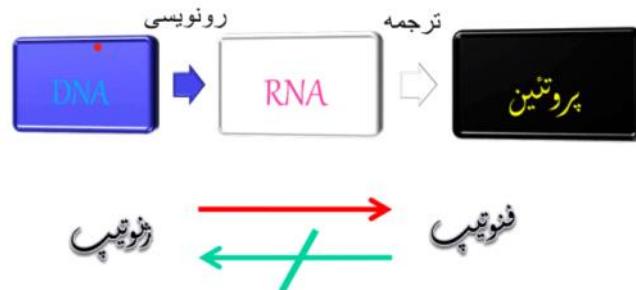
ویژگی های فیزیکی از پروتئین به وجود میان و پروتئین ها از نظر شیمیایی یک زنجیره ای از آمینو اسید ها هستن. و در نتیجه برای این که بفهمیم چه پروتئینی تولید میشه باید بفهمیم چه آمینو اسید هایی باید دنبال هم قرار بگیرن تا یک پروتئین خاص به وجود بیاد. و این که چه آمینواسید هایی کنار هم قرار بگیرن این کد های ژنتیکی مشخص میکنند. هر 3 تاکد که بهش میگیم گذن. یک کدی هستش که مشخص میکنه چه آمینو اسیدی داشته باشیم.

در DNA T A G C ما T A G C هم RNA (فرق بین T و U را کاری نداشته باشیم). در نتیجه هر 3 تا حرف میشه یک آمینو اسید. چند تاکد هم داریم که نشون میده زن ها از کجا شروع میشن و چندتا داریم مثل UAA و UAG که نشون میده کجا تمام میشن.

یک پروتئین تعداد آمینو اسید هاش چندتاست؟ همه پروتئین ها از با یک کد یکسان شروع میشه و بسته به این که چه تعداد بعدش آمینو اسید بعدش هست و چه آمینو اسید هایی باشن پروتئین های مختلفی میشه ازش درست کرد خیلی میشه.

با توجه به تکراری بودن بعضی کد ها تعداد آمینو اسید هامون 21 تا هستش. حالا اگر به متوسط هر پروتئین 100 تا آمینو اسید داشته باشه. تعداد پروتئین ها میشه 21^{100} که باهاش میشه یک جهان هستی رو درست کرد.

تئوری تکامل



در سلول از روی DNA یک کپی برداشته میشے بهش میگیره RNA. بعد از روی RNA ترجمه صورت میگیره و پروتئین تولید میشه. و همه سلول ها یک کارخونه پروتئین سازی هستن. ما مسیر از زنوتیپ به فنوتیپ داریم ولی بالعکس رو نداریم. معنیش اینه وقتی یکی ورزش میکنه و ماهیجه قوی داره این ها به زن و فرزندان نمیتونه انتقال پیدا کنه.

تئوری تکامل



لومارک

1744-1829

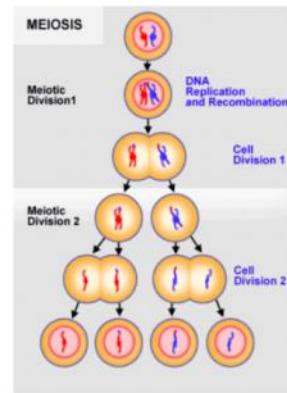


مسیر بالعکس رو لامارک اعتقاد داشت که بعدا این نظریه اش اثبات شد که اتفاق نمیافته.

ما 2 نوع تقسیم سلولی داریم:

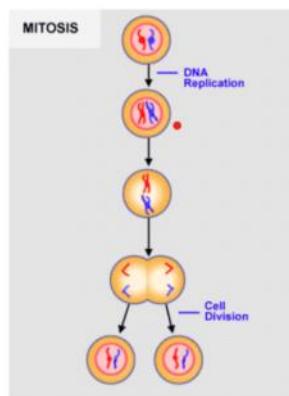
تئوری تکامل

تقسیم سلولی
میوز



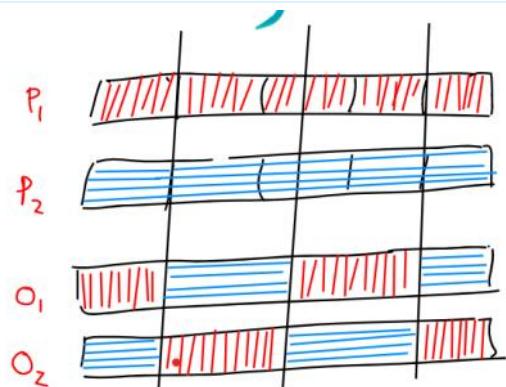
تئوری تکامل

میتوز



در میتوز وقایتی زیگوت در میاد تقسیم میشے به 2 تا که این 2 تا بنا اولی بود بالغ یا تخصصی باشن یا اگر اولی جنیفی بود جنیفی و بالغ بشه. در میوز ویژگی والدین به فرزندان منتقل میشه. یک سلول جنسی با تقسیم سلولی تبدیل میشے به 4 تا اسپرم(ماده) و اسپرم و تخمک لقاح پیدامیکنه و یک سلول فرزند رو تولید میکنه. که البته اینا دیپلولوژیدی هستش که کاری باهاش نداریم.

در حالت هابلوژیدی که باهش سر و کار داریم فقط در تک سلولی ها هستش. در این حالت کروموزوم نر یک سری ژن ها دارد. وقایتی که فرزند میخواهد به وجود بیاد به صورت تصادفی از یک سری نقاط که جا ش یا وسط ژن هستش یا جایی که ژن ها از هم جدا میشن، جایه جا میشن. اگر ژن های تر را عمودی و قرمز و ماده رو آبی و افقی در نظر بگیریم داریم:



که به این مکانیزم میگیم Crossover یا بازنگری. اگر crossover انجام نشه همون P1 و P2 رو خواهیم داشت ولی اگر بشه O1 و O2 رو داریم.

به صورت هاپلوبیدی آبای میشه و پریگ جدید داشت یا نه؟ اگر نقاط Crossover بین زن ها پاشن نمیشه ولی اگر داخل زن باشه نصف کد زن از یک والد میاد نصف دیگش از یک والد دیگر که اون موقع میشه و پریگ جدید هم داشته باشیم.
در تکامل واقعی جهش یک نوعی خطأ هستش که سعی میشه ازش جلوگیری هم بشه. ولی در الگوریتم های نکاملی مصنوعی نمیتوانیم انقدر کند عمل کنیم. میایم جهش هم بهش اضافه میکنیم و اینطوریه که به صورت تصادفی که هارو عوض کنیم.

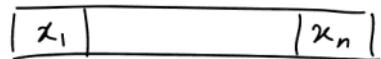
در جلسه قبل دیدیم اساس تکامل بر دو ویژگی هست. و این دو ویژگی:

- تنویر

- انتخاب

هستند. اگر از این ایده برای پیدا کردن مقدار بهینه یکتابع بخوایم استفاده کنیم چکار باید کرد؟
ابتدا باید یک تناظر بین تکامل و مساله خود به وجود بیاوریم. یعنی مساله ما به شکل ویژگی های تکامل در بیاد.

مثال میخوایم یکتابع $f(x_1, x_2, \dots, x_n)$ را ماسکیم کنیم، با توجه به این که ما در تکامل موجودات را با کروموزوم نشان میدهیم، ابتدا بایسی تابع تعیین کیم کروموزوم در مساله ما چه می باشد.
سپس پاسیتی شایستگی تعیین بشه، شایستگی میشه تابع هدف. که در مساله ما میشه همون (f) ما.
زن هامون میشه x_1 تا x_n .



زن ها میشن متغیر هایی از مساله ما که تغییر آن متغیر ها باعث تغییر شایستگی بشه.

به گامی که انجام دادیم میگیم بازنمایی (کد کردن مساله). پس از بازنمایی ما نیاز به یک جمعیت اولیه داریم. سپس میتوان گفت:



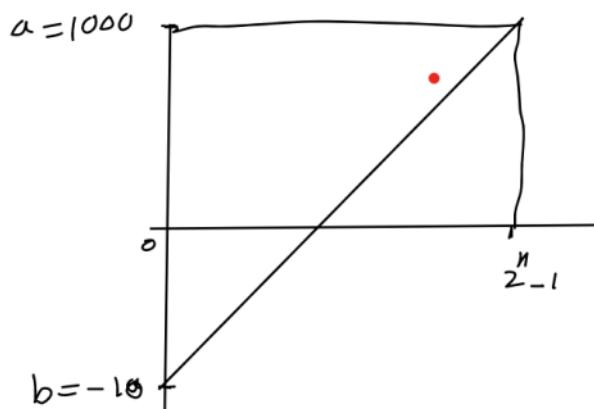
حالا جمعیت اولیه روح طوری درست کنیم؟ جمعیت اولیه بایسی **تصادف یکنواخت** باشه، چرا؟ چون میخوایم تنوع زیاد بشه و با یکنواخت تولید کردن ما از تمام نقاط به صورت یکنواخت باعث میشه بیشترین تنوع رو در ابتدا داشته باشیم.

فرض شود بخوایم ماسکیم تابع زیر را بدست آوریم:

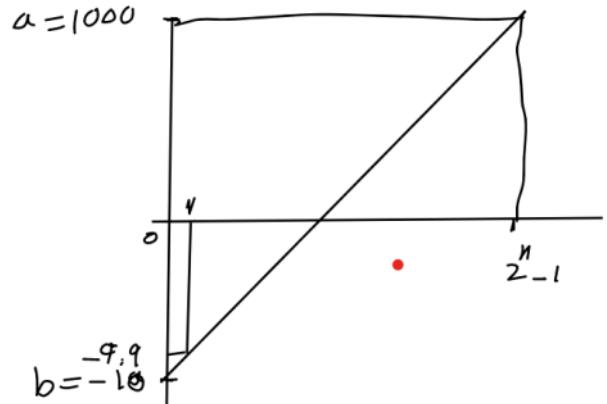
$$f(x) = x^{10} - 2x^5 - x^{100}$$

در ابتدا بایسی بازنمایی کنیم و کروموزوم را بدست آوریم. برای این کار بستگی به دقی که داریم تعداد زن های داخل کروموزوم متغیره. مثلا اگر $1000 < x < 10^{-1}$ و دقت هم 0.1 باشه، چی میشه؟

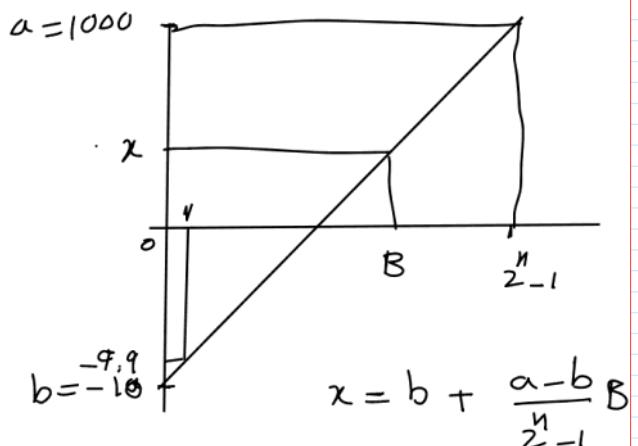
من با n بیت میتونم از 0 تا $2^n - 1$ رو نشون بدم. حالا ما از $-10 = b$ تا $1000 = a$ میخوایم عدد اعشاری داشته باشیم و با n بیت کمترین مقدار (-10) رو میخوایم با 0 نشون بدم و بیشترین مقدار رو میخوایم با $2^n - 1$ نشون بدم. در نتیجه یک رابطه خطی اگر بین اعداد اعشاری و باینری بدست بیاریم میشه این خط:



با دقی که ما داریم عدد 1 باینری میشه -9.9-



حالا عدد اعشاری اگر x باشه عدد باینری چی میشه؟



چطوری بدستش آورد؟ گفت من B رو صفر بذارم x میشه b اگر B رو $2^n - 1$ بذارم میشه a و اینطوری بدست میاد.

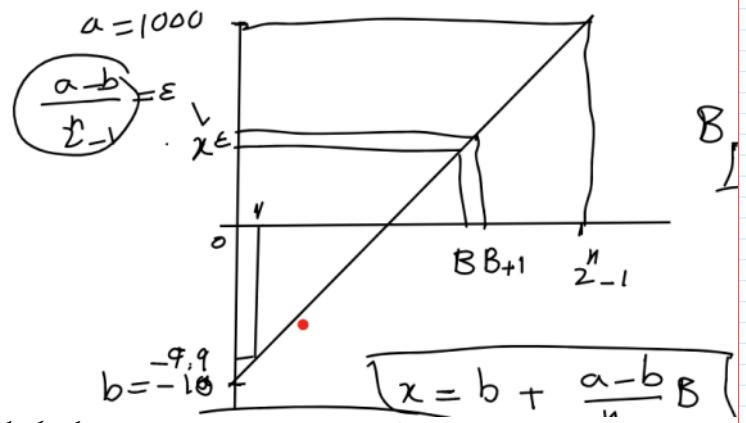
حالا خود اون B هم یک عدد حقیقیه که از تبدیل باینری به دسیمال بدست اومده:

$$\frac{b}{2^n}$$

1	0	1	1	1	0	1	0	1
---	---	---	---	---	---	---	---	---

$$B = \sum_{i=0}^{n-1} b_i \cdot 2^i$$

پس ما یک سری 0 و 1 داریم اول اونارو تبدیل میکنیم به دسیمال که دسیمال میشه بین 0 تا $2^n - 1$ بعد اونو مپ میکنیم بین -10 و 1000. حالا اپسیلون که دقتمنون هست چطوری بدست میاد؟



چون ما بازه اعدادمون از b تا a هست این تفاضلشون بازه رو بهمون میده و تقسیم بر 2^{n-1} میکنیم کوچکترین واحدمنون بدست میاد که میشه اپسیلون

$$\epsilon = 0.1 = \frac{a - b}{z^n - 1} \Rightarrow z^n - 1 = \frac{a - b}{\epsilon}$$

$$z^n = 1 + \frac{a - b}{\epsilon}$$

$$n \ln 2 = \ln(1 + \frac{a - b}{\epsilon})$$

$$n = \frac{\ln(1 + \frac{a - b}{\epsilon})}{\ln 2}$$

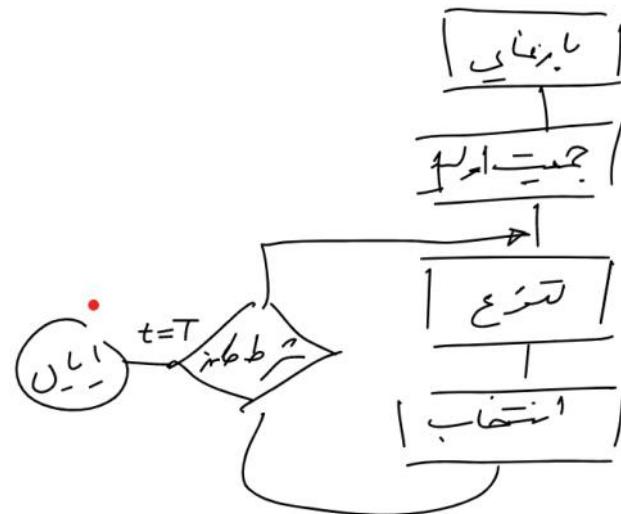
در مثال ما میشه:

$$n = 14$$

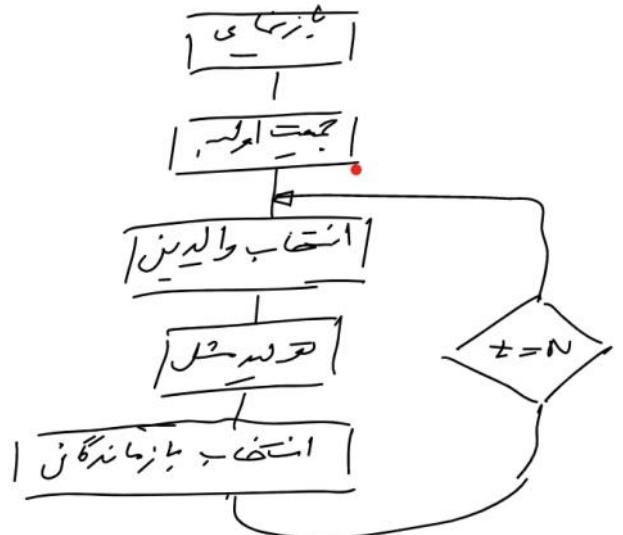
$$\frac{\ln(1 + \frac{10 \cdot 10}{0.1})}{\ln 2}$$

در واقع میشه 13.3 که چون تعداد بیت ها بایستی گستته باشه سقفشو میگیریم که کم نیاریم.

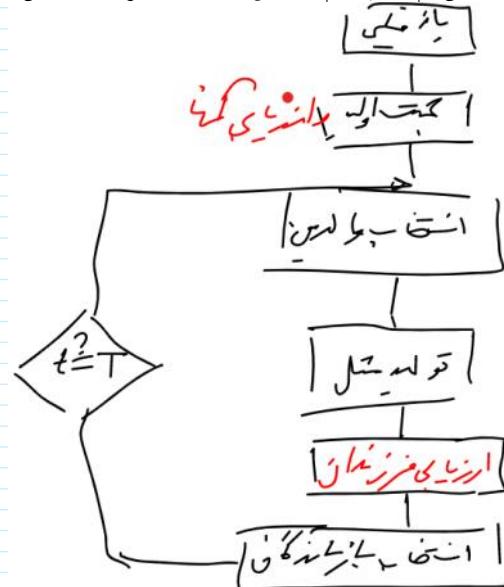
حالا اگر از $a = 2.7$ تا $b = 274.5$ داشته باشیم با دقت 0.05 اون موقع n چند میشه؟ میشه 12.4 یا به عبارتی 13 تا. فرض کنیم میخوایم تعداد کروموزوم هامون 10 تا باشه. برای هر 13 ژن داخل این 10 کروموزوم میایم یک عدد بین 0 و 1 تولید میکنیم کمتر از 0.5 بود میگیم 0 بیشتر بود میگیم 1 (تصادفی یکنواخت)



فلوچارت همچنان یه چی کم داره به طبیعت نزدیک بشه اونم انتخاب والدین و بازماندگانه، تنوع رو اگر تولید مثل در نظر بگیریم داریم:

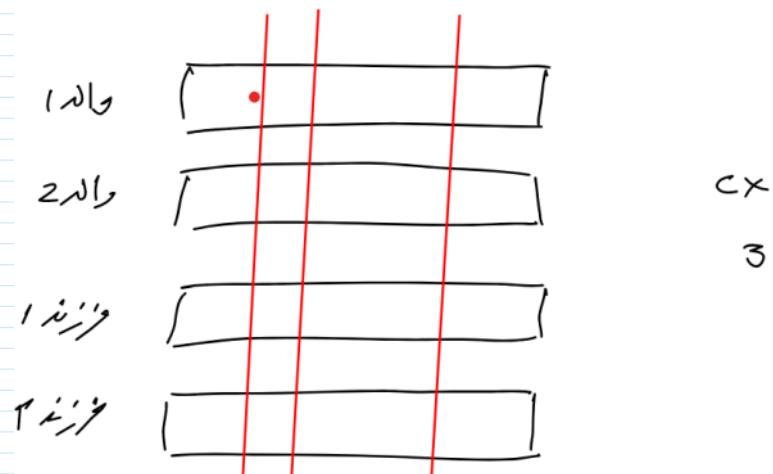


منتهی پایستگی شایستگی هم حساب کنیم اون جاش کجاست؟ بعد تولید مثل پایستگی حساب بشه



غیر از ارزیابی اولیه که برای والدین گذاشتیم بقیه از فرزندان میان و دیگه ارزیابی نیاز نیست. فقط مونده این که تولید مثل چوری انجام میشه.

یکی از راههای cross over هستش. یکی از پارامتر هاش اینه که از چند تا نقطه تقسیم بشه که حالا زیاد یا کمتر چه مزیت و معایبی دارن جلوتر میگیم. فرض کنیم تعداد نقاط تقاطع 3 تا باشه. به صورت تصادفی یکنواخت سه نقطه روی کروموزوم انتخاب میشه:



پس از crossover داریم:

والد ۱	<table border="1"><tr><td>101</td><td>11</td><td>101100</td><td>1001</td></tr></table>	101	11	101100	1001	CX
101	11	101100	1001			
والد ۲	<table border="1"><tr><td>011</td><td>01</td><td>0001100</td><td>1100</td></tr></table>	011	01	0001100	1100	3
011	01	0001100	1100			
فرزند ۱	<table border="1"><tr><td>101</td><td>01</td><td>1011001</td><td>1100</td></tr></table>	101	01	1011001	1100	.
101	01	1011001	1100			
فرزند ۲	<table border="1"><tr><td>011</td><td>11</td><td>0001100</td><td>1001</td></tr></table>	011	11	0001100	1001	.
011	11	0001100	1001			

حالا میمونه جهش که

$$\frac{1}{n} < P_m < \frac{1}{\mu}$$

$$P_m = \frac{1}{10}$$

یک مقداریه بین تعداد بیت و تعداد جمعیت. یعنی برای هر کدام یک عدد تصادفی بین ۰ و ۱ تولید میکنیم اگر از P_m کوچیکتر بود اون بیت رو عوض میکنیم.

فرزند ۱	<table border="1"><tr><td>101</td><td>0X</td><td>1011001</td><td>1100</td></tr></table>	101	0X	1011001	1100
101	0X	1011001	1100		
فرزند ۲	<table border="1"><tr><td>011</td><td>11</td><td>000110X</td><td>1001</td></tr></table>	011	11	000110X	1001
011	11	000110X	1001		

پس الگوریتم:

$$b \rightarrow B \rightarrow x \rightarrow f(x)$$

- ۱- بینهای
- ۲- جمیت اولیه و ارزیابی آن
- ۳- انتخاب والدین
- ۴- تقاطع / تنوع با ترکیب مسل
- ۵- جهش
- ۶- ارزیابی فرزندان
- ۷- انتخاب بهترین دادن
- ۸- مرط خاتمه شد یا نه

تو جلسه قبل دیدیم این مراحل رو داشتیم:

۱- بازنگاری کدکدن کروموزوم - تیسکی - مین روس لایر ری

$$b \rightarrow B \rightarrow x \rightarrow f(x)$$

۲- جست اولیه رازیابی

۳- انتخاب والدین

۴- تماطع

۵- جنس

۶- ارزیابی فرزندان

۷- انتخاب بازمانده گان

۸- سرت حاکمه گشت

مساله n - وزیر رو با $n=4$ رو میخوایم با الگوریتم فوق حل کنیم.
در مرحله اول (بازنمایی) صفحه شطرنج رو تحت عنوان ژن ها در نظر میگیریم و هر ستون یک ژن باشه. هر ژن یدونه 1 داره بقیش 0 باشه.

۱- بازنگاری

0	0	1	0
1	0	0	1
0	1	0	0
0	0	0	1

جایگشت همیزی

حایل-صحر

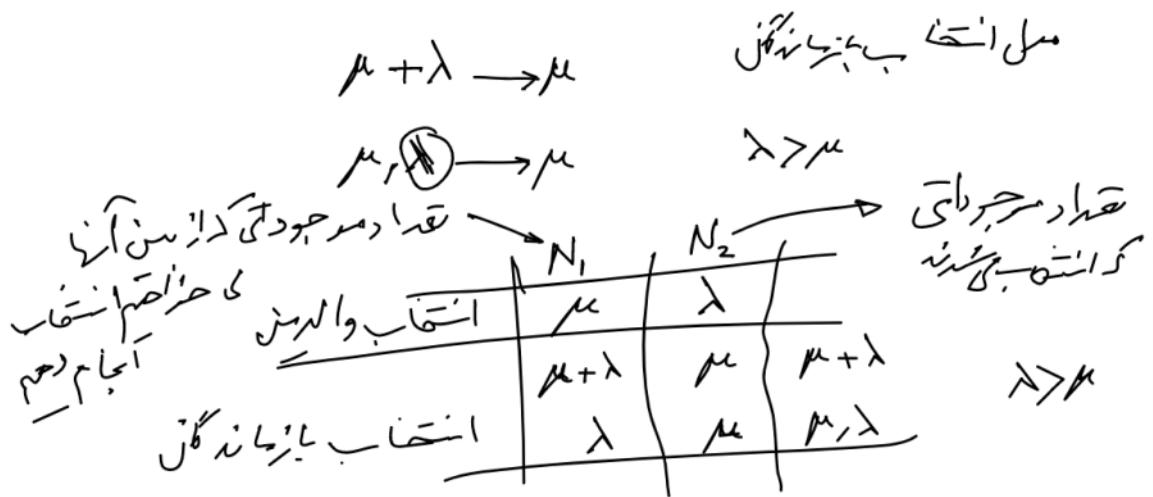
0	0	1	0
1	0	0	1
0	1	0	0
0	0	0	1

اگر تعداد موجودات رو 4 فرض کنیم و تعداد فرزندان هم 4 باشه. میشه تعداد موجوداتی که به نسل بعد منتقل میشن λ هم میشه تعداد فرزندان.

--- نکات پرسیده شده حین حل سوال توسط بچه ها:

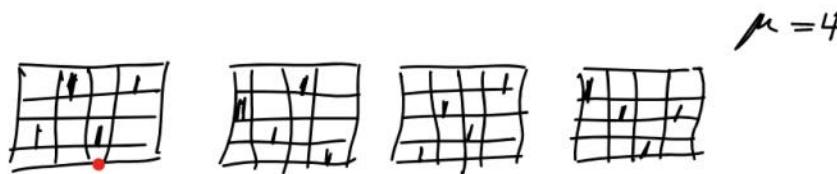
توی طبیعت والد از بین میهن فرزندان میمون وی در الگوریتم های تکاملی لزوماً اینطوری نیست. برخلاف طبیعت که جمعیت هی زیاد میشه در الگوریتم های تکاملی همیشه ثابت μ تا برای نسل بعدی انتخاب میشه.

- $\lambda + \mu$: از بین موجوداتی که داریم و فرزندانی که تولید کردیم با یک روش انتخابی مثل مناسب با شایستگی میایم μ تا رو انتخاب میکنیم.
- λ , μ : فقط از بین فرزندان μ تا برای بعدی انتخابی میشه. پس باید λ که تعداد فرزندان هست از μ بیشتر باشه که بتونیم μ تا شو انتخاب کنیم.

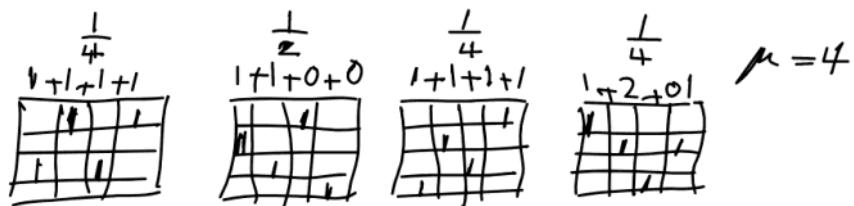


چرا N در انتخاب والدین میشه؟ چون تعداد فرزندان به اندازه تعداد والدین هستش. همیشه تعداد فرزندان برابر با تعداد والدینه. هر تعداد والدین انتخاب کنیم به همون تعداد فرزند تولید میشه.

---- بازگشت به مساله تعریف شده



با توجه به این که $\mu = 4$ ما در ابتدا 4 تا جمعیت اولیه داریم. برای ستون هر کدام میایم یک عدد تصادفی بین 1 تا 4 تولید میکنیم و سطر متناسب با اون عدد رو 1 میکنیم. حالا باستقی ارزیابی کنیم. ارزیابی رو چطوری انجام میدیم؟ میبینیم هر وزیر چندتا وزیر رو گارد میکنه. اون میشه هزینه و عکسش کیم میشه شایستگی



این روش ارزیابی روش خوبی نیست چرا؟ چون حالت های مختلف رو نتوانست شایستگی های متفاوتی داشته باشیم.

سپس میرسیم مرحله سوم: انتخاب والدین
یک راهش اینه که برای هر موجود یک احتمال میگیریم که متناسب با شایستگی در نظر میگیریم.
یه راه دیگش شایسته سالاریه. موجودات رو بر اساس شایستگی مرتب میکنیم از ابتدای لیست تا اونجا میخوابیم رو بر میداریم.

در این مثال چون $\mu = \lambda = 4$ هستش از روش اولی استفاده میکنیم. چون روش شایسته سالاری میشه همه رو برداریم و به دردمون نمیخورده. اگر متناسب با شایستگی بخوابیم بريم احتمال هاش میشن:

$$P_i \propto f_i$$

$$\lambda = \mu = 4$$

$$N_1 = \mu = 4$$

$$P_i = \frac{f_i}{\sum_{j=1}^N f_j} \rightarrow P_1 = \frac{\frac{1}{4}}{\frac{1}{4} + \frac{1}{2} + \frac{1}{4} + \frac{1}{4}} = \frac{1}{5}$$

$$P_3 = P_4 = P_1 = 0.2 \quad P_2 = 0.4$$

حالا ما میخوایم 4 تا موجود انتخاب کنیم.

-- در جواب سوال بجهه ها که گفتن شایسته سالاری چطوری کار میکنه: ---



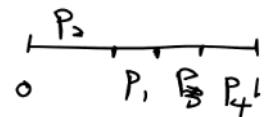
میایم اینطوری بر اساس شایستگی مرتب میکنیم و مثلا 4 تای اول را برو میداریم
باید حتما $N_1 < N_2$ باشد.

----- بازگشت به مساله:
احتمالات اینطوری شد:

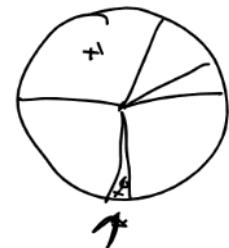
$$P_1 \quad P_3 \quad P_4 \quad P_2$$

$$0.2 \quad 0.2 \quad 0.2 \quad 0.4$$

حالا میایم توی بازه بین 0 و 1 اینارو بخش میکنیم (ترتیبیش مهم نیست):

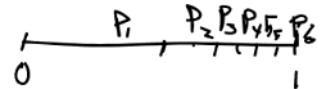


سپس یک عدد تصادفی یکنواخت بین 0 تا 1 تولید میکنیم. اگر بین 0 0.4 بود P_2 انتخاب میشه. اگر بین 0.4 0.6 تا 0.8 بود P_3 و در آخر بین 0.8 تا 1 بود هم P_4 .
به این روش میگن Roulette wheel. چرا؟ چون مثل همون دایره چرخون در کازینو هاست:

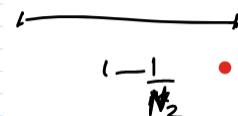


با توجه به این که N_2 حدود 10 الی 20 تا است و نمیشه اعداد کاملاً تصادفی یکنواخت تولید کرد (تعدادشون چون کمده). یک روش تحت عنوان (Stochastic Universal Sampling) که از نظر ریاضی مثل RW عه با این تفاوت که با تعداد کم هم Force میکنه یکنواختی رعایت بشه.

ما اگر بخوایم چهار تا موجود به صورت یک نواخت تولید کنیم یعنی این که اگر بین 0 تا 1 رو به 4 قسمت تقسیم کنیم باید از هر قسمت یکونه داشته باشیم. حالا میخوایم یه کاری کنیم این اتفاق بیافته. برای این کار اون بازه بین 0 تا 1 رو در نظر بگیریم با این فرض که این شکلی باشه:



فرض کنیم میخوایم N_2 تا را انتخاب کنیم. یک بازه به اندازه زیر تولید میکنیم:



الان در این مثالی که زدیم 6 = N_1 هستش ولی 4 تارو میخوایم انتخاب کنیم مثلا یعنی $4 = N_2$ هستش. بعد میایم روی این بازه $3/4$ تایی میایم $1/N_2$ تایی جدا میکنیم:



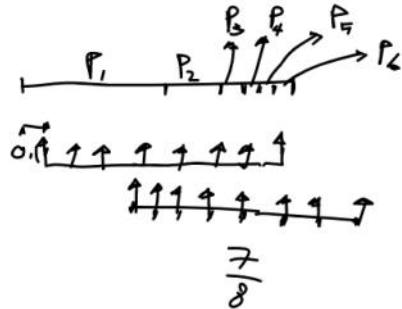
تو این مثال ما 3 قسمت میشه. حالا یک عدد تصادفی یکنواخت بین 0 تا $1/N_2$ انتخاب میکنیم. اون عدد میشه نقطه شروع بازه دوم که طولش $1/N_2 - 1$ بود:



با این کار انتخاب هامون با توجه به این فلش‌ها می‌شود:

P_1, P_1, P_2, P_5

یک مثال دیگه بزنیم بهتر جا بیافته. فرض کنیم 6 تا موجود داریم و 8 تارو می‌خواهیم انتخاب کنیم. اینطوری می‌شود:



$$N_1 = 6$$

$$N_2 = 8$$

$$1 - \frac{1}{N_2} = 1 - \frac{1}{8} = \frac{7}{8}$$

$$\text{عدد نسادی کنترل: } 0 : \frac{1}{N_2} \quad 0 : \frac{1}{8} \quad 0.1$$

$$P_1, P_1, P_1, P_2, P_2, P_3, P_4, P_6$$

اگر n_i رو بگیم تعداد دفعاتی که موجود i ام انتخاب می‌شود. امید ریاضی پیش می‌شود انتگرال n_i در P_{n_i} : که می‌شود:

$$E[n_i] = N_2 P_i$$

$$P_i = \frac{1}{3}$$

$$N_2 = 10$$

$$E[n_i] = 10 \times \frac{1}{3}$$

یعنی حدوداً 3 و خورده‌ای و بین 3 تا 4 تا از n_i انتخاب می‌شود. (در مثالی که زدیم هم P_1 حدود 0.5 بود 8 تا هم خواستیم انتخاب کنیم $8 * 0.5 = 4$ تا انتخاب شد)

یک تعریف فشار انتخاب هم داریم که می‌شود امید ریاضی انتخاب بهترین موجود که برابر با:

$$\text{عدم رعایت دباره انتخاب شده} = S_P = E[n_b] = N_2 P_b = \text{ضرایب اصل انتخاب بهترین موجود}$$

فشار انتخاب هرجی بیشتر باشه بدتره چون تکامل رو متوقف می‌کنه چون هرجی بیشتر باشه تنوع کمتر می‌شود.

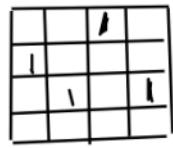
جلوتو روش انتخاب با شایستگی 2 تا مشکل دارد:

- ساکن شدن
- همگرایی زودرس

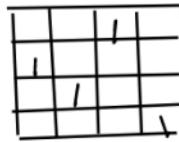


$$f_1 = \frac{1}{4}$$

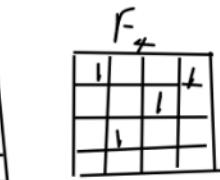
$$1+2+\sigma+1$$



$$f_2$$



$$\mu = \lambda = 4$$



$$P_i = \frac{f_i}{\sum_{j=1}^N f_j}$$

$$SP = N_2 P_2$$

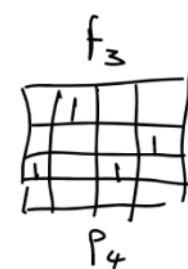
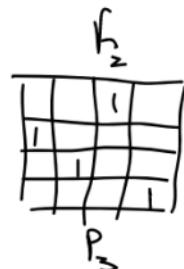
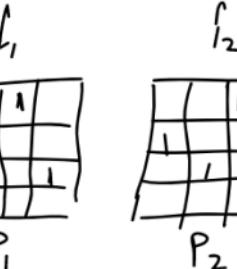
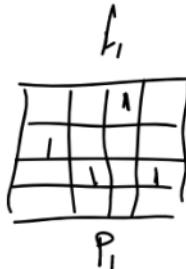
$$E[n_i] = N_2 P_2$$

$$f_1, f_2, f_3, f_4$$

$$N_1 = N_2 = 4$$

فرض کنیم انتخاب اینطوری بوده:

والدین

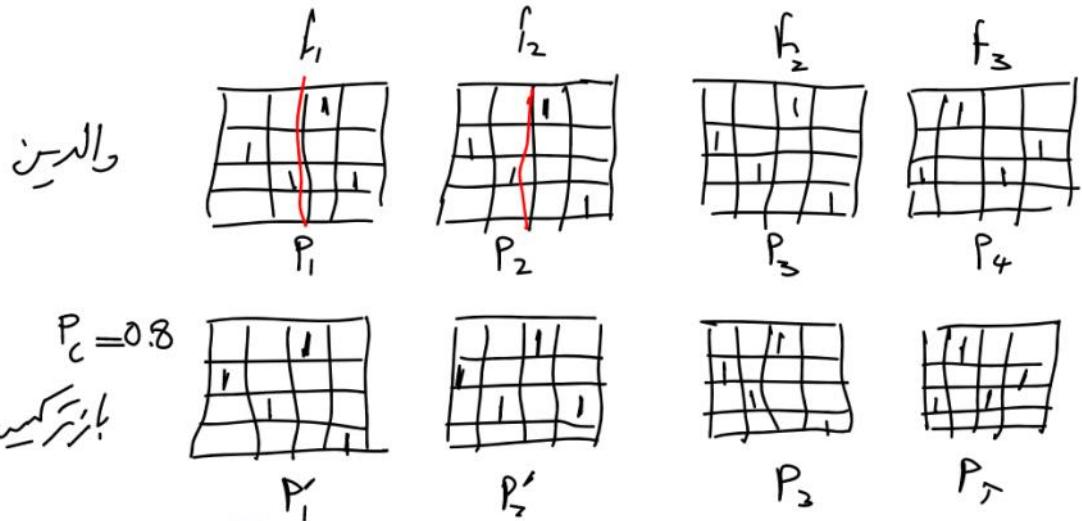


$$P_c = 0.8$$

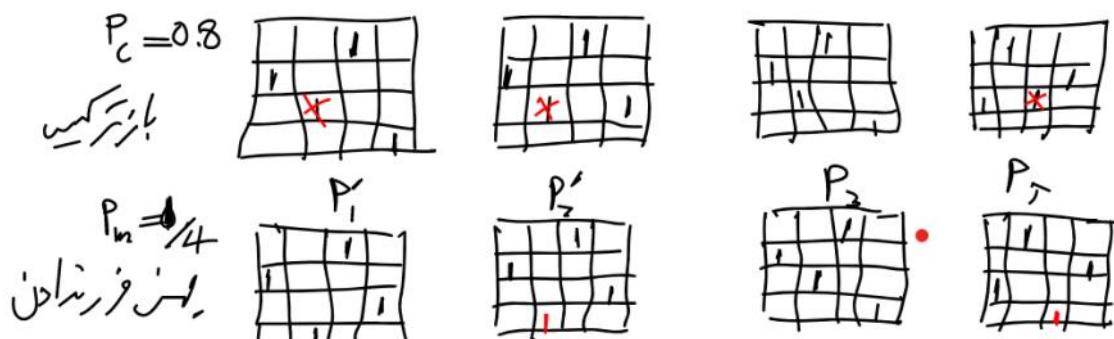
•

ما میایم $P_c = 0.8$ میزنیم. یعنی میایم بین هر جفت (جفت هامون P_1, P_2 , P_3, P_4) یک عدد بین 0 و 1 تولید میکنیم. اگر کوچیکتر از 0.8 شد Crossover میکنیم بیشتر نمیکنیم.

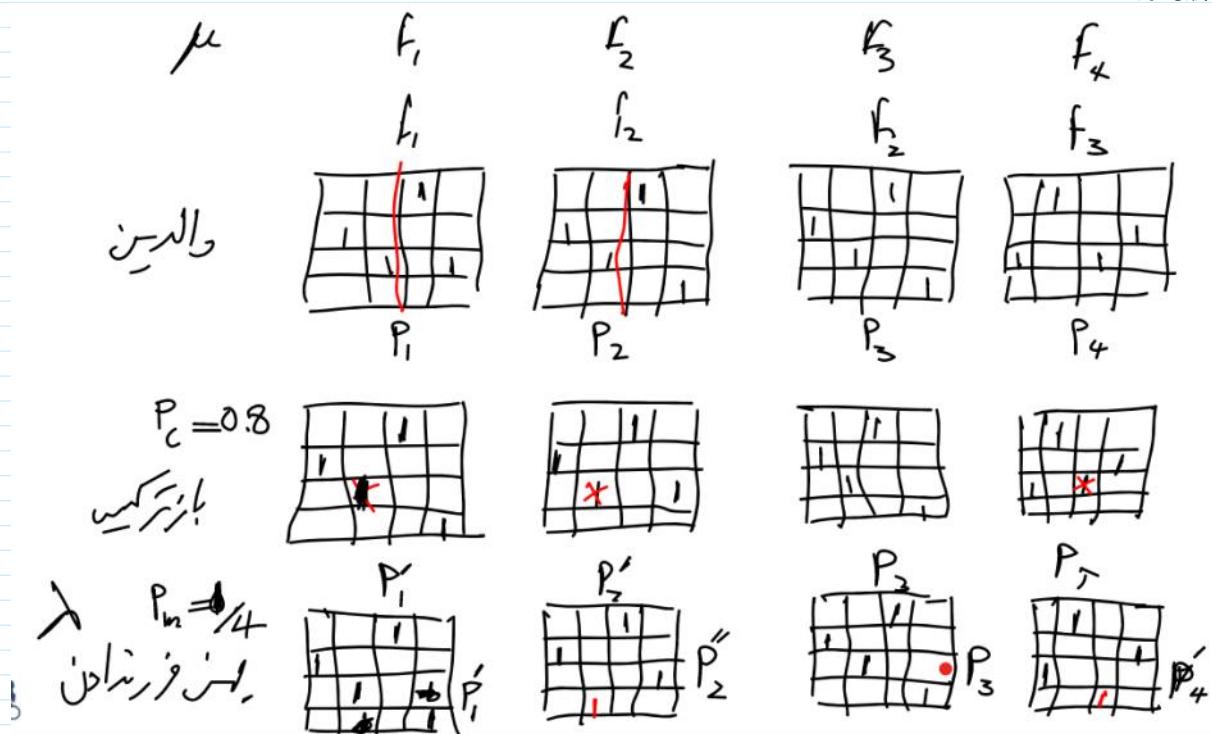
فرض کنیم برای P_1 و P_2 0.8 زیر 0.8 شه برای P_3 و P_4 بالای 0.8 بشه. فرزندان این شکلی میشه:



بعد میایم با احتمال $P_m = 0.25$ جهش رو انجام میدیم. (جهش رو عددش رو کم میگیریم احتمالشو اینجا چون کم داریم نمونه هارو 0.25 گرفت)



بعد جهش مبینیم فرزندان 2 و چهارم به جواب منتهی شدن (دیگه هیچکدام هیچکدام رو گارد نمیدن) [تو فرزند اول ستون آخر رو اشتباھی یه خونه بالاتر گذاشت و گزنه یه خونه پایین ترها].



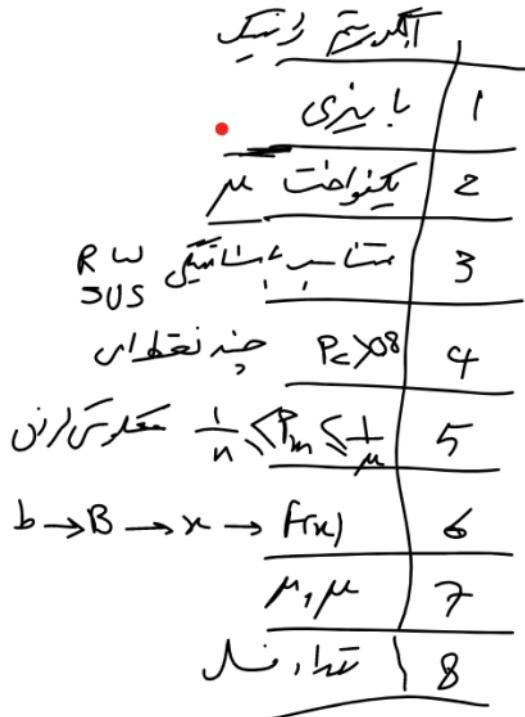
مبینیم که در بعضی فرزندان مثل P_3 نه بازترکیی داریم نه جهش و مستقیم از والد میاین.

بعد انجام crossover و جهش، مرحله والدین و بازترکیی رو حذف میکیم. فقط μ و λ میمونه (فرق μ و والدین اینه که در μ از f_1 تا f_4 رو داریم ولی در والدین اول به انتخابی بین اونا شده و f_4 نیومده و f_2 هم 2 بار اومده)

حالا میایم ارزیابی انجام میدیم بین p^1 , p^2 , p^3 و p^4 . که حالا ارزیابی با μ, λ یا $\mu + \lambda$ هست و چون $\lambda = \mu$ میگیم μ, μ . یعنی فرزندان همشون میرن به نسل بعدی در اینجا μ, λ هست.

به این روشنی که استاد توضیح میده میگیم الگوریتم ژنتیک.

- بازنمایی به صورت بازبینی هستش.
- جمعیت اولیه تصادفی یکنواخت
- ارزیابی هم که bit باید تبدیل بشه int بعد تبدیل میکنیم به حقیقی و $f(x)$ اشو حساب میکنیم
- انتخاب والدین مناسب با شایستگی SUS - RW
- در هنگام بازتکری یک $P_c >= 0.8$ داریم. به صورت چند نقطه ای میایم به صورت تصادفی یک نواخت انتخاب میکنیم.
- جهش بین $\mu < 1/n < P_m$ (البته اگه μ بزرگتر باشه جا شون عوض میشه). جهش هم به صورت bitwise میاد معکوس میکنه بیت رو بعد ارزیابی هم بیت رو میکنیم int رو میکنیم حقیقی و $f(x)$ اشو حساب میکنیم.
- انتخاب بازماندگان به صورت $\mu = \lambda$ (چون $\mu = \lambda$) و همه فرزندان به نسل بعد منتقل میشن.
- شرط خاتمه تعداد نسل



* اگر ما جهش نداشته باشیم هیچ موقعیت جواب ممکنه نرسیم. چون مثلا تو جواب ما مثلا باید بیت دهم امون 1 باشه ولی تو جمعیت اولیه هیچ کدوم بیت دهم اش یک نباشه.

عیوب های الگوریتم ژنتیک:
• مناسب با شایستگی بودن:

$$P_i \propto f_i \quad P_i = \frac{f_i}{\sum_{j=1}^n f_j}$$

○ همگرایی زودرس

$$f_a > f_b > \bar{f}$$

فرض کنیم یک موجود داریم با شایستگی f_b که از شایستگی f_a کمتره و از شایستگی optimum هم کمتره. مثلا میخوایم به شایستگی 200 برسیم. موجودی داریم با شایستگی 90 و بقیه شایستگی اشون مثلا 1 و 1.1 و 0.9 و 0.8 و روانش و روی هم شایستگی شون بشه 10. چه با RW ببریم چه با SUS اینطوری میشه که در اون خط کش ما 90 درصدش رو f_b میگیره 10 درصدش رو بقیه:

$$f_b > f_b$$

فرض کنیم بخوایم 10 تا موجود رو بخوایم انتخاب کنیم. فشار انتخاب اینطوری میشه که :

$$SP = 10 \times 0.9 = 9$$

بعنی ۹ تا والد میشن f_b و دفعه بعد همگی میشن f_b و میشه که f_b جواب بهینه ما نیست. به این میگیم همگرایی زودرس

پس همگرایی زودرس چندتا شرط دارد:

- i. انتخاب متناسب با شایستگی
- ii. فشار انتخاب بالا باشد. یعنی f_b ما از شایستگی متوسط خیلی بیشتر باشد
- iii. f_b ما فاصله داشته باشد.

○ ساکن شدن

شرط هایی که موجب به ساکن شدن میشن:

- i. انتخاب متناسب با شایستگی

ii. فشار انتخاب نداشته باشیم. یعنی f_b تقریباً با f متوسط یکی باشد و همه موجودات شانس انتخاب برابر داشته باشند

$$f_b \approx f$$

$$\left\lceil \pi \right\rceil = 1$$

در این حالت اگر مثلاً ۸ تا موجود داشته باشیم احتمال همیشون میشه $1/8$ و شانس همه برابر. در انتخاب والدین هموانا که قبل ابودن دوباره میان و چون جمعیت تنوع نداری نمیتوانه انجام بده. تو اینجا مثل همگرایی زودرس نیست که همیشون نزدیک به هم میشن و همچ تکرار میشن.

• بازنمایی بازیزی: هم مزايا داره هم معایب

منظور از بازنمایی نمایی اینه که مثلاً میومدیم تابع $x^{100} - 2x^{50} + 2x^{10} = f(x)$ رو ماکسیمم کنیم میومدیم یه تعداد بیت برash در نظر میگرفتیم.
○ معایب:

▪ وقتی مثلاً میایم یک تابع داریم ۱۰ تا متغیر داشته باشد، برا هر کدوم ۱۰ تا بیت بگیریم میشه ۱۰۰ تا بیت و تعداد بیت زیادی رو باید معرف کنیم.

▪ فرض کنیم یک شرکتی میخواهد طلا استخراج کنه. باید اول معدن رو پیدا کنه و کشف کنه. پیدا کردن این معدن هارو میگیم اکتشاف یا جستجو عمومی. بعد این که معدن رو کشف کردن باید طلا رو استخراج کنه که بهش میگیم جستجو محلی. این که چقدر دنبال اکتشاف باشیم چقدر دنبال استخراج یکی از مشکلات جستجو هستش.

اول باید محدوده جواب رو پیدا کنیم (جستجو عمومی) بعد بریم محدوده رو دقیق تر استخراج کنیم.

□ اگر اون شرکت اولین معدن رو که پیدا کرد بره سمت استخراج جه اشکالی داری؟ ممکنه بدتر ضرر کنه و اون عیار داخل معدن هزینه استخراجش حتی در نیاره.

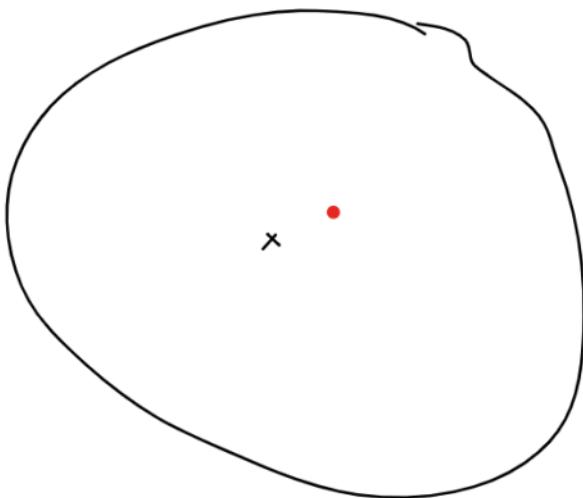
□ اگر اون شرکت وسوس داشته باشد همش دنبال معدن بگردد چی؟ همش هزینه داره میکنه و هیچ موقع منفعت بدست نمیاره. استخراج رو بعضیا انتفاع هم میگن چون نفع میرن. ولی اکتشاف رو فقط انجام بدیم هیچ موقع نفعی نداریم.

ما باید هزینه جستجو عمومی و نفع رو متعادل سازی کنیم.

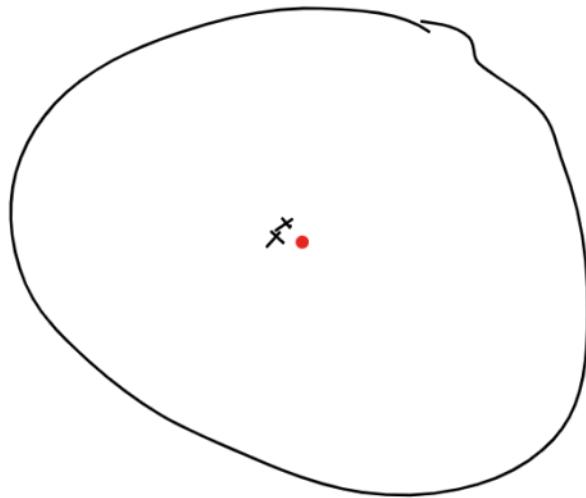
در الگوریتم های جستجو عمومی اول اکتشاف انجام میدن بعد میرن سراغ استخراج.

در الگوریتم های تکاملی میشه اینارو همزمان انجام داد. ما یکی از عملگر های crossover و جهش رو میداریم اکتشاف انجام بده اون یکیش رو میناریم جستجو محلی انجام بده.

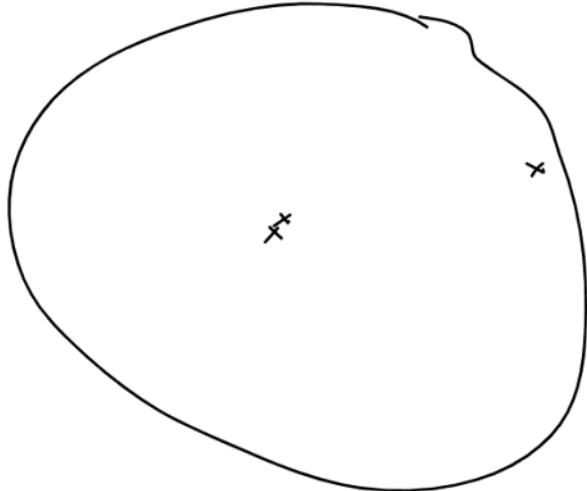
اگر ما بازنمایی بازیزی باشه جهش یکی میشه؟ بسته به این داره بیت با ارزش داره تغییر میکنه یا بیت کم ارزش



فرض کنیم این فضای جستجو ما باشه و یک موجودیتی اون وسط داریم. اگر بیت کم ارزش تغییر کنه میشه جستجو محلی چون تو همون حوالی فرق میکنه:



ولی وقتی بیت پر ارزش تغییر کنه میشه جستجو عمومی:



پس با بازنمایی نمایی ما نمیتونیم کنترل کنیم که جهش جستجو عمومی داره انجام میده یا محلی Crossover چی؟ اونم باز بستگی داره کدوم قسمت داره جایه جا میشه و نمیشه تشخیص داد.

پس در بازنمایی باینتری هیچ کنترلی نداریم جستجو عمومی داریم انجام میدیم یا محلی.

برای حل این مشکل حالا چیکار میشه کرد؟ یک راه حل سادش اینه که از gray code استفاده بشه. وقتی یک بیت رو در gray code تغییر بدیم مقدار عددیش تو همسایگی تغییر میکنه. مثلا اگر 4 بیت در نظر بگیریم:

GROW	decimal	binary	Gray code	
0	0	0000	0000	
1	1	0001	0001	
2	2	0010	0011	
3	3	0011	0010	
4	4	0100	0110	
5	5	0101	0111	
6	6	0110	0101	
7	7	0111	0100	
8	8	1000	1100	
9	9	1001	1101	
11				Two values differ in only one bit

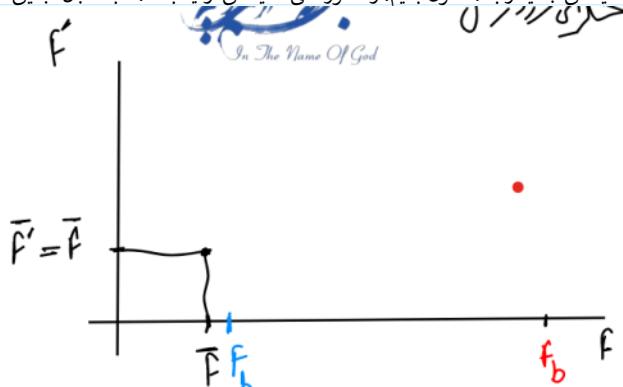
LEARN

در این صورت وقتی یک بیت رو تغییر بدیم تغییر زیادی نداریم و وقتی gray code استفاده میکنیم جهش جستجو محلی انجام میده و crossover جستجو عمومی انجام میده.

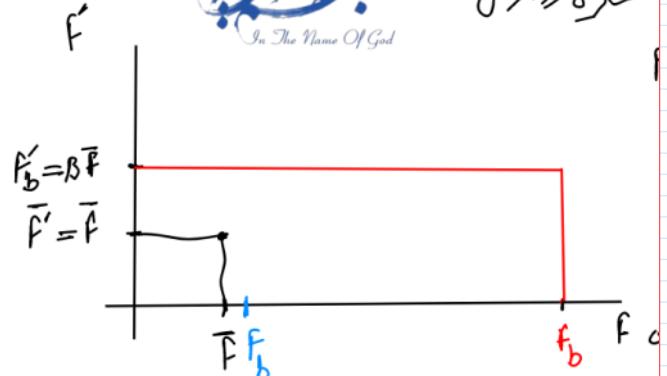
○ مزیت: یک مساله بهینه سازی ترکیبی که یک سری متغیر ها int باشن یه سری float یه سری char همرو میشه با باینتری کد کرد.

در این جلسه میخوایم بینیم برای مشکلات مثل همگرایی زودرس چه راه حل های داریم.

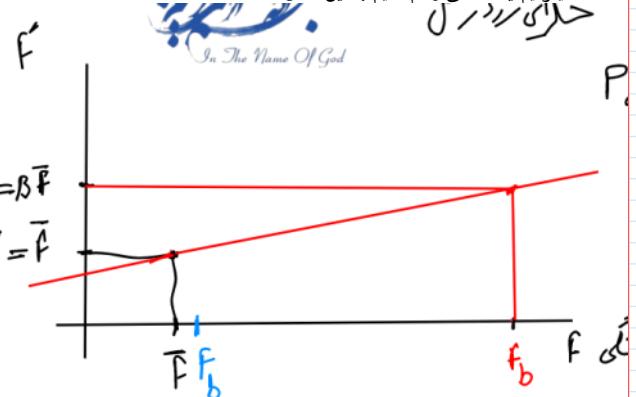
یک راه ساده که به ذهن میرسه اینه که روی شایستگی ها یک تبدیل قرار بدم و این تبدیل باعث بشه که اختلاف شایستگی خیلی زیاد یا خیلی کم نداشته باشیم. به این صورت که محور عمودی رو بگیریم شایستگی جدید و با f' نشون بدم. و محور افقی شایستگی اولیه باشه. ما به دنبال تبدیل هستیم که متوسط شایستگی رو تغییر نده:



از طرفی میخوایم شایستگی best امون دیگه خیلی فاصله نداشته باشه. یعنی یک \bar{f} بدهست بباد که بتا برابر متوسط باشه:



حالا ما میتوانیم یک خطی رسم کنیم به این شکل:



فرمول این خط چیه؟

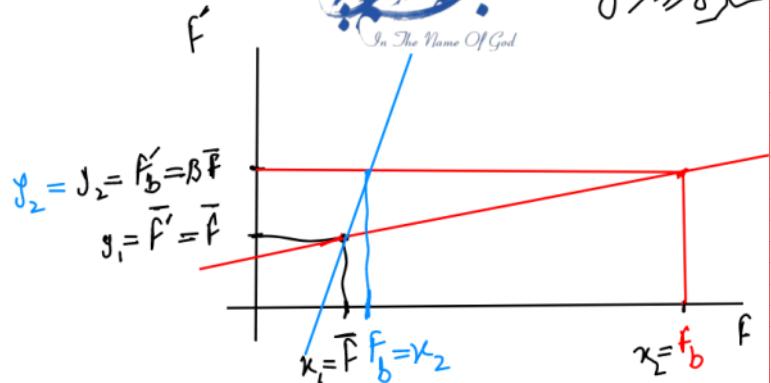
(x_1, y_1)

$$y = \frac{y_2 - y_1}{x_2 - x_1} (x - x_1) + y_1$$

(x_2, y_2)

$$\boxed{f'_i = \frac{\beta \bar{f} - \bar{f}}{f_b - \bar{f}} (f_i - \bar{f}) + \bar{f}}$$

حالا اگر بخوایم برای ساکن شدن در نظر بگیریم:



میبینیم این رابطه که بدست آوردهیم هم برای ساکن شدن هم همگرایی زودرس یکیه و جفتشو حل میکنه ولی این که بنا رو چطوری بگیریم شاید حسی نداشته باشیم، میشه این فرمول رو به جای بنا در \bar{f} متوسط بیاریم بر حسب فشار انتخاب بگیریم و خودمون فشار انتخاب رو کنترل کنیم. اگر از فشار انتخاب استفاده کنیم داریم:

فشار انتخاب خودش خصوصیات جالی داره. ما هرجی فشار انتخاب بالاتر باشه سرعت همگرایی بالاتره و هرجی کمتر باشه تنوع بیشتره. در نتیجه خودش یک پارامتر کنترلی برای تعادل بین تنوع و سرعت همگرایی به وجود میاره و اگر به جای بنا فشار انتخاب رو کنترل کنیم خیلی بهتر میتونیم الگوریتم تکاملی رو براساس این که ابتدا هستیم و تنوع نیاز داریم یا انتهای هستیم و میخوایم به جواب برسیم کنترل کنیم. هرچه فشار انتخاب بیشتر باشه جستجو محلی تر و هرجی کمتر باشه عمومی تر میشه.

ما میدونیم جمع احتمالا باید برابر با 1 باشه

$$\sum P = N_2 P_b$$

$$\sum_{i=1}^{N_1} P_i = 1$$

توجه کنیم در احتمال N_1 عه ولی در فشار انتخاب N_2 .

ما برای انتخاب لزومی نداره شایستگی رو حساب کنیم. در روش SUS (و اون یکیش) از روی شایستگی احتمال رو حساب میکردیم. توی اینجا P هارو باید با f های جدید بدست بیاریم:

$$P_i = \frac{f'_i}{\sum_{i=1}^{N_1} f'_i}$$

پس لزومی نداره ما شایستگی بدست بیاریم برای الگوریتم های تکاملی. میتونیم مستقیما احتمالات جدید بدست بیاریم. یعنی شایستگی رو همون شایستگی اولی داشته باشیم و احتمالاتی جدید بدست بیاریم. پس 2 تا معادله داریم و میخوایم P هارو خطی بر اساس شایستگی بیان کنیم و کنترل کنیم، پس P برابر میشه با:

$$P_i = \alpha f_i + b$$

اگر بخوایم a و b در بیاریم میشه:

$$\sum P = N_2 P_b$$

$$\sum_{i=1}^{N_1} P_i = 1$$

$$P_i = \alpha f_i + b$$

$$\sum P = N_2 P_b = N_2 (\alpha f_b + b) \rightarrow \alpha f_b + b = \frac{\sum P}{N_2} \quad (1)$$

$$\sum_{i=1}^{N_1} P_i = \sum_{i=1}^{N_1} (\alpha f_i + b) = \alpha \sum_{i=1}^{N_1} f_i + b N_1 = 1$$

$$\alpha \left(\sum_{i=1}^{N_1} f_i \right) + b = \frac{1}{N_1} \rightarrow \alpha \bar{f} + b = \frac{1}{N_1} \quad (2)$$

اینطوری a بدست میاد:

$$\textcircled{1} - \textcircled{2} = \alpha(f_b - \bar{f}) = \frac{s_p}{N_2} - \frac{1}{N_1} \rightarrow \alpha = \frac{\frac{s_p}{N_2} - \frac{1}{N_1}}{f_b - \bar{f}}$$

حالا میخواهیم b را بدست بیاریم. اول میایم اون رابطه اصلی رو مینذاریم:

$$P_i = \alpha f_i + b \quad \textcircled{3}$$

و داریم:

$$P_i = \alpha f_i + b = \alpha f_i - \alpha \bar{f} + \frac{1}{N_1} = \alpha(f_i - \bar{f}) + \frac{1}{N_1}$$

$\textcircled{2}$

$$P_i = \frac{\frac{s_p}{N_2} - \frac{1}{N_1}}{f_b - \bar{f}} (f_i - \bar{f}) + \frac{1}{N_1}$$

حالا توانستیم احتمال رو بر حسب فشار انتخاب بدست بیاریم.

حالا میتوانیم فشار انتخاب رو طوری تعیین کنیم که همگرایی زودرس و ساکن شدن نداشته باشیم و همچنان میتوانیم اونجا ها که تنوع میخواهیم کوچیک فرض کنیم اونجاها که میخواهیم سرعت همگرایی بره بالا بپریم بالا.

فقط نکته ای که باید در نظر بگیریم اینه که فشار انتخاب در بازه $[1, 2]$ هستش. خود 1 رو نمیشه $(1 < s_p <= 2)$

با یک روش ساده ترم میشه این کارو کرد. با این هدف موجوداتی که شایستگی هاشون خیلی فاصله داره رو کم کنیم و بالعکس اونا که کمه رو زیاد کنیم. برای این کار به جای شایستگی از رتبه استفاده میکنیم. فرض کنید یک سری موجود داریم با یک سری شایستگی:

70 80 50 5 10 1 15

میایم اونی که کمترین شایستگی داره کمترین رتبه بدیم و بر اساس شایستگی به همه رتبه بدیم:

70 80 50 5 10 1 15
6 7 5* 2 3 1 4

اگر sort کنیمشون داریم:

1 5 10 15 50 70 80
1 2 3 4 5 6 7

حالا اگر بخواهیم احتمال حساب کنیم، داریم:

$$f_i = i \quad P_i = \frac{f_i}{\sum_{j=1}^i f_j} = \frac{i}{\sum_{j=1}^i j} = \frac{2i}{N_1(N_1+1)}$$

این دفعه این شکلی میشه:

$$P_i = ai + b \quad (3) \quad SP = N_2 P_b \quad \sum_{i=1}^{N_1} P_i = 1$$

$$SP = N_2 P_b = N_2 (aN_1 + b) \rightarrow \left\{ \begin{array}{l} aN_1 + b = \frac{SP}{N_2} \quad (1) \end{array} \right.$$

$$\sum_{i=1}^{N_1} P_i = \sum_{i=1}^{N_1} (ai + b) = a \sum_{i=1}^{N_1} i + N_1 b = 1$$

$$a \frac{N_1(N_1+1)}{2} + N_1 b = 1 \rightarrow \left\{ \begin{array}{l} a \frac{N_1+1}{2} + b = \frac{1}{N_1} \quad (2) \end{array} \right.$$

$$(1) - (2) = a(N_1 - \frac{N_1+1}{2}) = \frac{SP}{N_2} - \frac{1}{N_1} \quad \left| \begin{array}{l} a = \frac{\frac{SP}{N_2} - \frac{1}{N_1}}{\frac{N_1-1}{2}} \end{array} \right.$$

$$P_i = ai + b = ai - a \frac{N_1+1}{2} + \frac{1}{N_1}$$

↑
②

$$P_i = a(i - \frac{N_1+1}{2}) + \frac{1}{N_1}$$

$$P_i = \left(\frac{SP}{N_2} - \frac{1}{N_1} \right) \frac{i - \frac{N_1+1}{2}}{\frac{N_1-1}{2}} + \frac{1}{N_1}$$

جمع بندی: روش های انتخاب

روزگاری انتخاب

۱- حفظ ادعی کلمه امت

۲- به عنوان انتخاب

۳- تیسته سالاری

۴- متناسب با استایل

$$\frac{SUS}{RW} \quad P_i = \frac{f_i}{\sum_{j=1}^n f_j}$$

۵- متناسب با سلسله مقیاس های خود

$$\frac{SUS}{RW} \quad P'_i = \frac{f'_i}{\sum_{j=1}^n f'_j}$$

۶- متناسب با سلسله مقیاس های خود

$$\frac{SUS}{RW}$$

$$P_i = af_i + b \quad \sum P = 1$$

۷- متناسب با ترتیب

$$\frac{RW}{SUS} \quad P_i = \frac{2i}{N_i(N_i+1)} \quad f_i = i$$

۸- متناسب با ترتیب

$$\frac{RW}{SUS} \quad P_i = ai + b \quad \sum P = 1$$

$$SP = N_2 P_2$$

مسئله ۷ و ۸ نیست به ۵ و ۶ چیه؟ توی متناسب با رتبه مشکل همگایی زور درس و ساکن شدن حل میشه. چون شایستگی ها یکی داره اضافه میشه و با قابلیت یدونه بیشتر نیست و یک موجود نمیتونه فاصلش با قله زیاد بشه چون یکی دارن زیاد میشن. شایستگی ها هم با هم برایر نمیتونن بشن چون یکی کی فاصله دارن.

مسئله ۷ خیلی جالب دیگش اینه که شایستگی میتونه حساب نشه. خیلی وقتا محاسبه شایستگی خیلی سخته. مثلا بخواهیم به یک صفحه شطرنج شایستگی بدیم ممکنه خیلی سخت باشه ولی ۲ تا صفحه شطرنج بدین بگن کدوم شایستگی بهتری داره راحت تر میشه جواب داد. یعنی باید به صورت کیفی مقایسه کنیم نه کمی. با مثلا ۲ تا تابلو نقاشی بدن بگن نمره بدیم سخته ولی بگن کدوم بهتره راحت تر میتونیم بگم. در نتیجه مقدار دقیق شایستگی نیازی نیست و باید یک تفاوتی ایجاد کنیم و sort کنیم و تابع دقیق شایستگی نیاز نیست. کافیه با تابع غیر دقیق داشته باشیم یا انسان و چیز دیگه ای باید sort رو انجام بد.

دو مرحله انتخاب داريم:

• والدين

• بازماندانگان

گفتيم تمام در دو صورت رخ ميده:

• تنوع

• انتخاب درگير با شايستگي

يك روش انتخاب در اين جلسه ياد ميگيريم تحت عنوان Q تورنومنت که مزيت همه روش های قبلی رو دارد. Q به جوري شبیه فشار انتخاب است. Q تا موجود رو به صورت تصادفي يک نواخت از جمیعیت بر میداريم (اگر $Q=2$ باشه بهش میگیم باينزی تورنومنت). در باينزی تورنومنت 2 تا موجود به صورت تصادفي يک نواخت برداشته ميشه، در بين اين 2 موجود بهترین رو برمیداريم. اين میتوانه با جايگذاري باشه يا بدون جايگذاري

در جايگذاري اون 2 موجود رو که برداشتم میتوانيم به جمیعیت اضافه کنيم.

در بدون جايگذاري ديگه به جمیعیت اضافه نمیکنيم.

تورنومنت مزاياي رتبه بندی رو دارد يعني مشکل همگرایي زودرس و ساكن شدن رو نداره. احتياجي نیست مقدار شايستگي رو دقیق حساب کنیم.

مزیت که Q تورنومنت نسبت به رتبه بندی دارد: در رتبه بندی باید بر حسب شایستگی sort کنیم ولی در Q تورنومنت در بین Q تا ماکسیمم رو برمیداریم. در نتیجه هزینه محاسبه Sort کردن رو نداریم و در نتیجه از این جهت نسبت به رتبه بندی مزیت دارد.

در رتبه بندی فشار انتخاب همیشه بین 1 و 2 بود، اگر بخواهیم بیشتر از 2 باشه باید از Q تورنومنت ببریم. وقتی $Q=2$ میداریم انگار فشار انتخاب به جوري 2 هست (در صورتی که با جايگذاري باشه، بدون جايگذاري باشه يه موجود بيشتر از يه دفعه نمیتوانه انتخاب بشه و در نتیجه فشار انتخاب حقيقي ميشه 1. هرچند مشکل همگرایي زودرس و ساكن شدن رو نخواهد داشت)

* تو اسلامپادا هست وقت نمیشه سرکلاس گفت ولی میشه نشون داد Q تورنومنت ادامه رتبه بندی خطی هست و تمام مزايا رو دارد و sort کردن تو ش نداره.

جمع بندی روش های انتخاب:

۱- دھادھی تکنیکات

۲- انتخاب همگی μ/μ

۳- تئیسواری

$$\text{SUS} \quad P_i = \frac{f_i}{\sum_{j=1}^n f_j}$$

$$\begin{aligned} \text{RW} \quad P_i &= f'_i \\ \text{SUS} \quad P_i &= \frac{f'_i}{\sum_{j=1}^n f'_j} \quad f'_i = \bar{f} \quad f'_b = \beta \bar{f} \quad f'_i = \alpha f_i + b \quad \text{متارسنه خلی} \end{aligned}$$

$$\begin{aligned} \text{RW} \quad SP = N_2 P_b \quad P_i = \alpha f_i + b \quad \text{باکتریل من را انتخاب} \\ \text{SUS} \quad \sum P_i = 1 \end{aligned}$$

$$\begin{aligned} \text{RW} \quad P_i &= \frac{2^i}{N_i(N_i+1)} \quad f_i = i \quad \text{نتاب برتریه} \\ \text{SUS} \end{aligned}$$

$$\begin{aligned} \text{RW} \quad P_i &= \alpha i + b \quad \text{متارسنه خلی} \\ \text{SUS} \quad SP = N_2 P_b \quad \sum P_i = 1 \end{aligned}$$

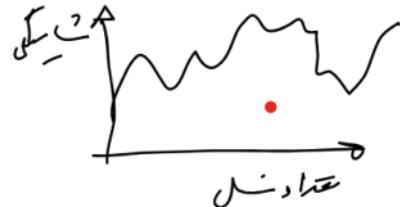
$$Q \text{ تورنومنت}$$

برای انتخاب بازماندگان 2 مدل داشتیم.

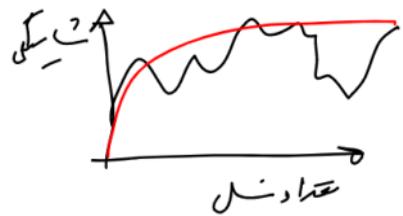
- $\mu + \lambda$: از بین μ و λ میایم μ تا بر میداریم.
- λ : فقط از فرزندان μ تا بر میداریم.

هر کدام از اینا یه ویژگی خاصی دارد:

- λ : فراموش کاره (والدارو فراموش میکن) ولی $\mu + \lambda$ حافظه داره. در نتیجه چون λ, μ والدارو فراموش میکنه فرزندان جدید و **تنوع بیشتری** داره. ولی μ سرعت همگرایی بیشتری داره.



با توجه به این که فراموش کاره شایستگیش نوسان داره و هی زیاد و کم میشه. ولی $\mu + \lambda$ وقت رو رسم میکنیم:



دیگه نوسان نداره و همگرا میشه.

- λ, μ چون فراموش کاره فرار میتونه بکنه از مینیمم محلی ولی $\mu + \lambda$ تو شگیر میکنه
- λ, μ از پارامتر های بد میتوونه فرار کنه (اگه بعدا وقت شه توضیح میده استاد که چیه)

ما قبلا هم دیدیم یه تعادلی بین جستجو عمومی و جستجو محلی باید برقرار کنیم. یا به طور دقیق تر یک تعادلی بین تنوع و همگرایی اینجاد کنیم (چون برای تکامل جفتشو نیاز داریم). اگر همچ بريم سراغ تنوع خوب نیست، همش بريم سراغ شایستگی و همگرایی خوب نیست. در نتیجه گفتم اگر جهش رو جستجو محلی انتخاب کردیم (بازترکیب) رو جستجو عمومی تعادل برقرار میشه.

- اگر انتخاب بازماندگان رواز λ, μ استفاده کنیم که تنوع زیادی داره. روش انتخاب رو نمایم از Q توزونمنت بريم که تنوع بالای داره. بلکه باید بایم از روش مناسب با شایستگی بريم. که تعادل بین تنوع و همگرایی برقرار بشه.
- اگر م از $\mu + \lambda$ استفاده کردیم چون تنوع نداره بهتره کنارش از Q توزونمنت استفاده کنیم.

به همین دلیل نمیشه گفت روش مناسب با شایستگی خوبه یا بد. به نکته دیگه داریم اینه که معمولا میان بهترین موجود رو آرشیو میکنن. بهترین موجود حالا میتونه تو تکامل شرکت کنه و فرزند تولید کنه یا فقط همینطوری نگه داریم فراموش نکنیم و بهترین جواب رو از دست ندیم.

حالات خاص:

فرض کنیم مدل های الگوریتم تکاملی هامون رو اینطوری نشون میدیم:

$$\begin{aligned} &= H(\mu + \lambda) \\ &= H(\mu, \lambda) \end{aligned}$$

حالا در یک حالت خاص داریم $(1 + 1)EA$. یعنی یه موجود داریم، یدونه فرزند تولید کیم و از بینشون یدونه رو انتخاب کیم برای نسل بعدی. در این حالت ما بازترکیبی نمیتونیم داشته باشیم چون بازترکیبی 2 تا میخواهد و ما فقط میتونیم جهش داشته باشیم. این الگوریتم میشه معادل با **تپه نوردی**.

ولی $(1, 1)EA$ معادل با **Random walk** میشه. چون یه موجود داریم یه موجود رو انتخاب میکنیم همون رو میبریم نسل بعد. دیگه شایستگی مطرح نیست.

به حالی هم $(1 + \lambda)EA$ یا $(\lambda, 1)EA$ باشه. یه موجود داریم لاندا تا فرزند با جهش تولید میکنیم. سپس از بین این لاندا تا فرزند یکی رو انتخاب میکنیم یا از بین $\lambda + 1$ تا موجود یکی رو انتخاب میکنیم. در این روش فقط جهش داره.

به حالی هم $(1 + \mu)EA$ هستش. به این روش میگن **steady state**. به خاطر این که تغییراتش آرمه. و هر دفعه یک فرزند تولید میکنه و بدترین موجود حذف میشه. (1, 1)EA که **نمیشه داشت** چون از بین 1 فرزند باید μ موجود انتخاب کرد که نشدنیه.

حالات دیگه $(\mu, \mu)EA$ که همیشه از بین فرزند انتخاب میشن و والدین از بین میهن پس بهش میگن **Generational**. $(\mu + \mu)EA$ هم که فرزندان با تعداد والدین برابره ولی ما بازماندگان رو از بین فرزندان و والدین هردو انتخاب میکنیم.

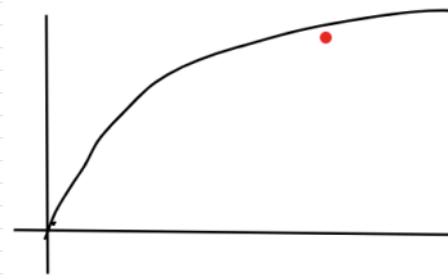
حالات کلی:

$$\begin{aligned} &EA(\mu + \lambda) \\ &EA(\mu, \lambda) \end{aligned}$$

که در دوی توسعه شده $\lambda = \mu$ (هم توسط تجربی هم به کمک روابط ریاضی میشه اثبات کرد بهترین نتیجه رو میده) چرا بیشتر از λ کیم بد میشه؟ هزینه در الگوریتم تکاملی میشه تعادل دفعاتی که ما تابع شایستگی رو ارزیابی کردیم. الگوریتم تکاملی که N تا نسل داره، μ تا موجود و λ تا فرزند تعداد دفعاتی که تابع شایستگی رو محاسبه میکنند λN . (چون نسل اول هم پایستی شایستگی رو حساب کنیم). چون تعادل نسل معمولا زیاده اون μ اول رو صرف نظر میکن λN .

حالا اگر ما $N=100$ داشته باشیم و در یک حالت $\lambda=10$ باشیم، در حالت اول $1000 \lambda=20$ باشیم.

به همین دلیل برای مقایسه الگوریتم های تکاملی محور x هارو تعداد نسل در نظر نمیگیریم و تعداد ارزیابی در نظر نمیگیریم. محور y هم متوسط شایستگی یا بهترین موجود در نسل میگیریم.



تعداد ارزیابی

یک منحنی دیگه هم اینطوری که محور x میشه ارزیابی محور y هم تنوع. تنوع هم 2 نوع داریم:

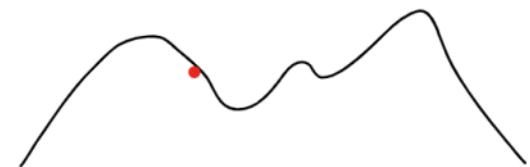
- ژنتیکی: در یک نسل واریانس زن ها حساب بشه. میایم 2 تا سیگما میداریم یکی رو موجودات یکی در زن ها و میشه:

$$\frac{1}{n\mu} \sum_k \sum_i (x_{i,k} - \bar{x}_k)^2$$

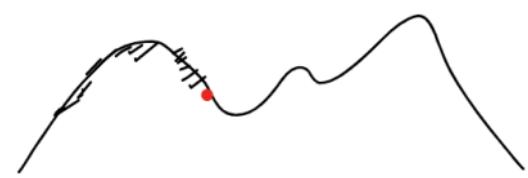
• فنوتیکی

$$\frac{1}{\mu} \sum_k (f_{i,k} - \bar{f})^2$$

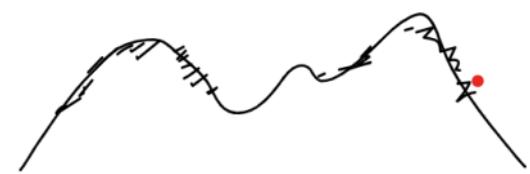
حالا کدوم رو باید بذاریم رو محور؟ تنوع ژنتیکی درسته ولی خیلی جاها این 2 با هم یکسان نیست. اگر یه همچین شکلی داشته باشیم:



اگر موجودات اینطوری پخش بشن:



تنوع ژنتیکی و فنوتیکی تقریباً برابر میشه.
ولی اگر همه جا پخش بشن:



این موقع دیگه تنوع ژنتیکی و فنوتیکی برابر نیست.

ولی چون در الگوریتم های تکاملی موجوداً بعد چند نسل تعایل دارن بزن رو یه قله در نتیجه خیلی جاها برای ساده سازی از تنوع فنوتیکی میریم بجای ژنتیکی.

ولی همیشه اینطوری نیست. یک سری الگوریتم تکاملی داریم که چند جوابی هستن، یعنی میخوایم یه کاری کنیم تمام مینیمم ها و تمام ماکسیمم های محلی رو بددست بیارن اونجا دیگه ژنتیک و فنوتیک یکی نمیشه تنوعش.

شرط خاتمه چیا میتوشه باشه:

1. تعداد ارزیابی
2. حد آستانه
3. همگرایی
4. عدم تنوع

حالا ما همگرایی رو چطوری حساب کنیم؟ اگر در چند نسل متوالی تغییرات متوسط شایستگی از یه حدی کمتر بشه، میگیم همگرا شدیم:



حالا این میتوشه همگرایی زودرس باشه یا همگرایی اصلی باشه. این که کدومشه نمیشه متوجه شد.

حالا مزیت و عیب هرکدام:

- تعداد ارزیابی:

- مزیت: در Loop بینهایت نمیافتد. میتوئیم بر اساس تعداد ارزیابی و زمان هر ارزیابی بایم زمان الگوریتم رو محاسبه کنیم. پس تضمین خروج از الگوریتم داریم که فقط در این شرط خاتمه هست.

- عیب: تعیین تعداد ارزیابی مناسب. اگر زیاد بگیریم ممکنه زود به جواب برسیم و زمان رو هی تلف کنیم تا به جواب برسیم. ممکنه کم هم بگیریم و بعد به جواب میرسیم در واقع ولی چون کم گرفتیم به نتیجه نمیرسیم. تضمین جواب رو نداریم

- حد آستانه:

- مزیت: مزیت اینه اگه از الگوریتم اومدیم بیرون حتما مطمئن میشیم که جواب رو پیدا کردیم. تنها شرط خاتمه ایه که وقتی میایم بیرون مطئنیم جواب پیدا شده برای بقیه نمیشه اینظری گفت. ممکنه همگرایی زودرس بخورن

- عیب: Loop بینهایت. برای بعضی مسائل نمیشه فهمید حد آستانه کجاست (مثل پیدا کردن مینیمم یکتابع که اصن نمیدونیم چند هست که بعد بخوایم وایستیم. ولی در مساله 8 وزیر میدونیم شایستگی 0 شد یعنی هیشکر هیشکر رو گارد نمیده و حد آستانه رو داریم)

- همگرایی:

- مزیت: اگر ما همگرا شدیم دیگه ادامه دادنش فایده نداره (چه زودرس باشه چه اصلی).

- عیب: Loop بینهایت. تضمین جواب رو نداریم.

- عدم تنوع:

- مزیت: اگر تنوع از بین رفته میایم بیرون

- عیب: Loop بی نهایت. تضمین جواب رو نداریم.

مسئله 100 وزیر را میخواهیم حل کنیم.

- اولین مرحله رو گفته بیانی هستش. بازنمایی برای 100 وزیر جه شکلی باشه خوبه؟ روش های مختلفی هستش ولی برای کد کردن یک سری شرایط داریم.
- مهمنی شرطی اینه که تمام حالت جواب (فضا) رو تونه پوشش بد. مثلا حالت باقی از نظر بگیریم و تعداد بیت هاکم باشه یک قسمت از فضای جواب رو پوشش نمیدیم.
- تا اونجا که میتوانه فضا رو کوچیک کنه:

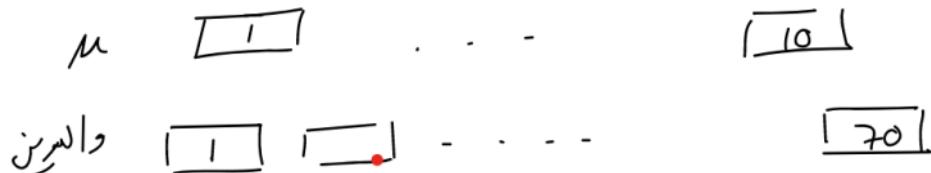
- مساله 100 وزیر رو متلا آگر یک صفحه 100 در 100 بگیریم (10 هزار تا خونه).
- یا میتوانیم 100 تا زن در نظر بگیریم هر کدام یک عددی بین یک تا ده هزار باشه (بگه تو کدوم یکی از اون 10 هزار تا خونس).
- چون هر خونه 1 تا ده هزار میتوانه باشه و ما هم 100 تا خونه داریم تعداد حالت‌امون میشه¹⁰⁰.
- در الگوریتم های تکاملی هرجی فضای بزرگ تر باشه کاری کمتر میشه و بالعکس.
- اگر حالت اولی رو بگیریم (100 در 100) اون موقع تعداد حالت ها میشه¹⁰⁰.
- ولی بازم میشه کوچیکتر کرد. در مساله 100 وزیر میدونیم اگر در سطری یک وزیر قرار بگیره در ستون های دیگه در اون سطر وزیری نمیتوانه باشه.

پس میتوانیم جایگشت ببریم که بشه! حالا چون 100 وزیر کشیدن سخته برامون استاد 8 تا رو کشید و با جایگشت مشخص کرد:



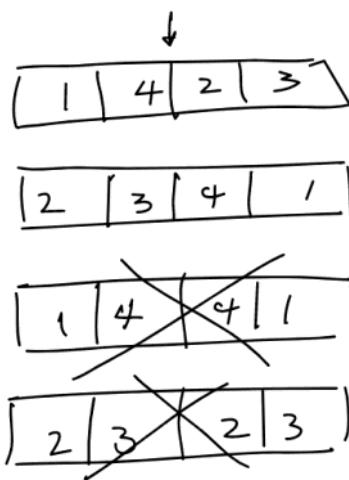
حالا آگر $m=10$ داشته باشیم، در ابتدا 10 موجود به صورت تصادفی یکنواخت تعیین میکنیم. (مرحله دوم)

مرحله سوم - انتخاب والدین: اگر $m=7$ در نظر بگیریم، تعداد فرزندان میشه 70 تا. معمولا وقتی تعداد فرزندان را از جمعیت خیلی بیشتر در نظر میگیریم، انتخاب والدین به صورت تصادفی یکنواخت میشه انتخاب کرد. (بدون شایستگی) در اینجا 10 تا موجود داریم، اعداد تصادفی یکنواخت بین 1 تا 10 تولید میکنیم و هر کدام رو اومد بر میداریم. 70 بار این کارو تکرار میکنیم. چون تصادفی یکنواخت هستش معنیش این میشه از هر موجود 7 تاکی میاد پایین.



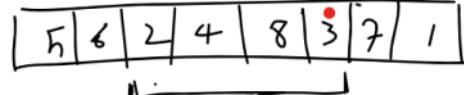
سپس میایم اینا رو جفت جفت در نظر میگیریم. (چون تصادفی یکنواخته دیگه shuffle نمیکنیم).

حالا میایم سراغ بازترکیب. اگر از بازترکیب تک نقطه‌ای استفاده کنیم:



2 فرزند تولید شده هیچ کدام جایگشت نیست و شرط مساله مارو خراب میکن. پس باید crossover ای تعریف کنیم که جایگشت رو خراب نکنه. تعداد این بازترکیبا خیلی زیاده 2 تا شوالان میگیم بقیش تو اسلامیا هست:

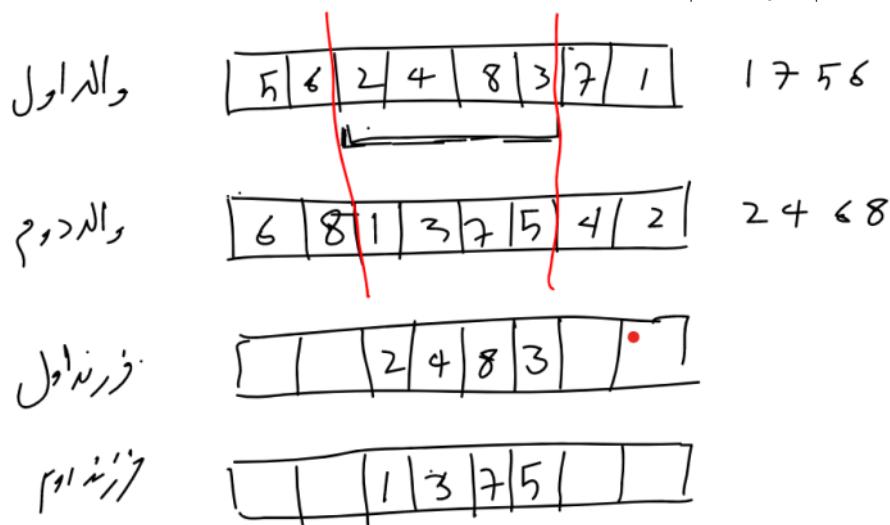
- امید crossover اینه که به صورت کاملاً تصادفی بیاد و بینی های خوب والد رو کنار هم بذاره و یک فرزند با شایستگی بیشتر رو تولید کنه.
- و بینی خوب در جایگشت: یک ترتیب خاص باعث میشه یک موجود خوب باشه یا نه.



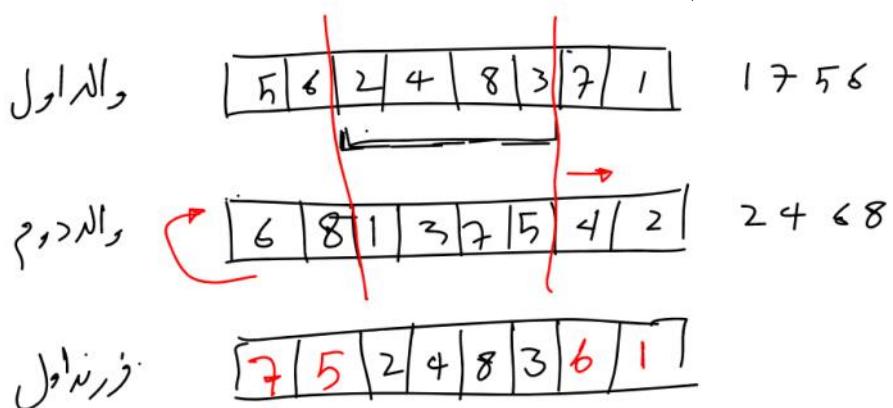
فرض کنیم این 3 4 8 2 یک ترتیب خوب باشه و بخواهیم حفظ شه و بقیه عوض بشن. این ترتیب اگر بمونه و بقیه عوض بشن جستجوی عمومیه یا محلی؟ بستگ داره که این ترتیب چقدر بزرگ

باشه. در اینجا چون 4 تا ثابت میمونن 4 تا عوض میشن به نوعی جستجو عمومیه. جستجوی محلی رو میشه همسایگی در نظر گرفت. این که 7 کنار 1 باشه میشه جستجوی محلی.

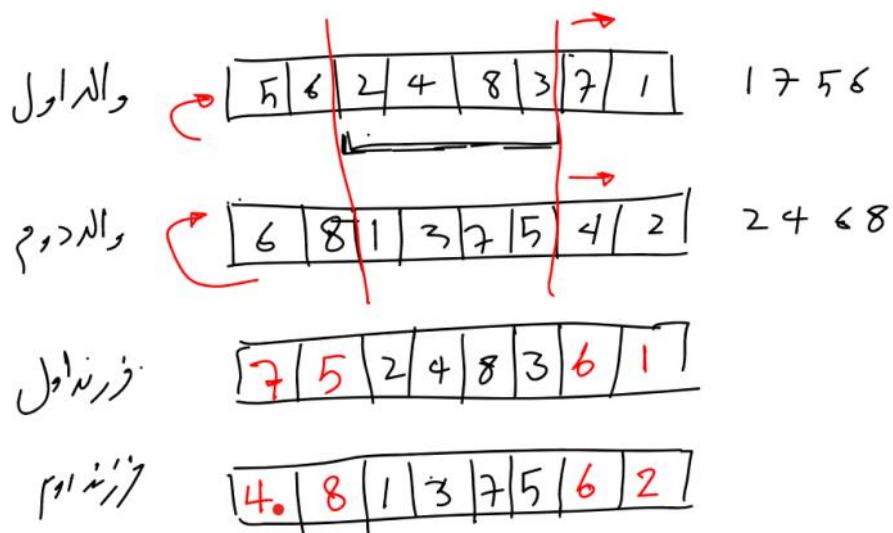
تو یه روش میایم اینطوری میکنیم:



حالا بقیه فرزند رو چطوری بچینیم؟ (اعدادی که توی 2 تا والد سمت راستش نوشته شده اعداد باقی موندش)



نحوه انتخاب اینطوریه که در فرزند اول او میدیم از نقطه بعد از crossover شروع میکنیم که عدد قرار بدیم (از سمت راست دومین خونه). چون فرزند اول هستش سعی میکنیم از اعداد والد دوم انتخاب کنیم. چطوری؟ میایم میبینیم در والد دوم اول 4 رو داریم. آیا 4 درین 1756 هست؟ نه. پس میریم سراغ 2. آیا 2 هست؟ نه. پس میریم سراغ 6. آیا 6 هست؟ اوه پس میداریم. بعد میریم میبینیم 8 نیست ولی 1 هست میداریم. بعد به ترتیب 7 و 5 رو میبینیم و میداریم.



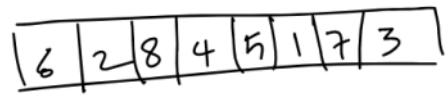
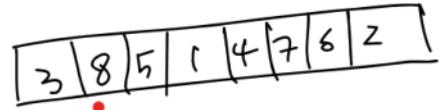
با این روش جایگشت خراب نمیشه و فرزندان تولید میشن.

** توی crossover ما همیشه تصادفی عمل میکنیم و شایستگی کمتر یا بیشتر میشه و لی امیدمون اینه بیشتر بشه.

روش دیگه بازترکییه:

واله اول

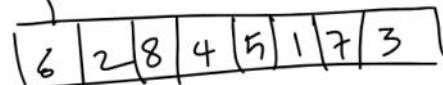
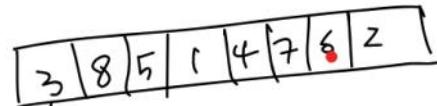
الله (ع)



نحوه عملکرد اینضوریه که اول از 3 شروع میکنیم میبینیم روش 6 عه:

واله اول

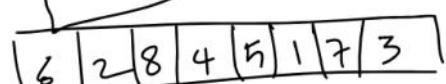
الله (ع)



بعد میریم تو والد اول 6 رو پیدا میکنیم و وصل میکنیم:

واله اول

الله (ع)



بعد روش 7 عه:

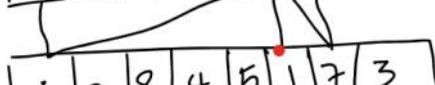
واله اول

الله (ع)



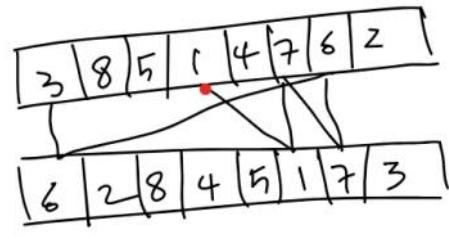
واله اول

الله (ع)



واله اول

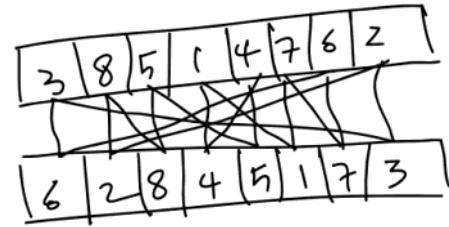
واله دوم



و همینطوری ادامه میدیم...

واله اول

واله دوم

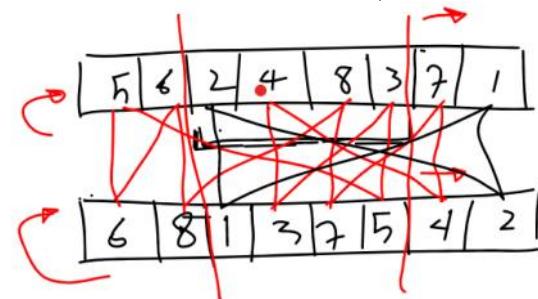


در این مثال یه حلقه بیشتر نداره. البته چون 8 تا بود اینطوریه 100 تا بکنیم چندتا حلقه به وجود میاد.

در یک مثال دیگه بررسی کنیم مثلا 2 تا حلقه میشه:

واله اول

واله دوم



حلقه اول رو ثابت میداریم و دوم رو جایه جا میکنیم.

یعنی مثلا 6 رواز واله اول مینویسم بعد 2 رو که میبینیم چون حلقه دومیه جاش 1 میداریم میشه:

5 6 1 4 8 3 7

بعد حلقه دوم رو میبینیم جای 1 میاد 2 و داریم:

5 6 1 4 8 3 7 2

به همین ترتیب از والد دوم شروع کنیم داریم:

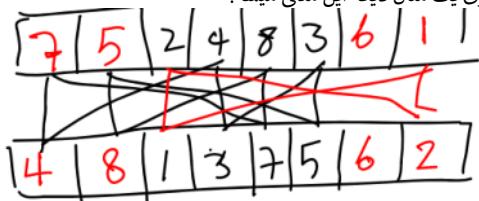
6 8 2 3 7 5 4 1

حالا این تعداد حلقه ها ممکنه بیشتر باشن یکی در میون جایه جا میکنیم.

فرزند اول

فرزند دوم

توی یک مثال دیگه این مدلی میشه:



فرزند:

7 5 1 4 8 3 6 2

4 8 2 3 7 5 6 1

سپس میرسیم به جهش. یکی از جهش هایی که جستجو محلی هستش اینه که به طور تصادف یکنواخت بیایم 2 تا بیت رو انتخاب کنیم تعویض کنیم:

1	3	6	8	7	5	2	4
1	3	2	8	7	5	6	4

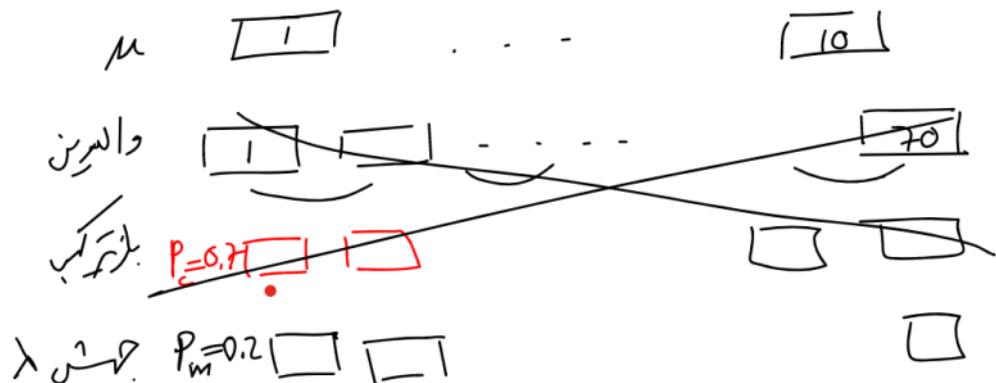
چون 2 تا ترتیب زیادی رو عوض نمیکنند (مخصوصا وقتی ژن ها 100 تاس) میشه جستجو محلی.
یک راه دیگش درج کردنه، یعنی 2 رو بیاریم بعد 6:

تغییر

1	3	6	8	7	5	2	4
1	3	2	8	7	5	6	4

13 6 2 8 7 5 4

در مساله ما با احتمال 20 درصد روی هر ژن جهش انجام میشود و قبلی حذف میشن:



بعد میرسیم به انتخاب بازماندگان
• μ, λ با روش مناسب با شایستگی
• $\mu + \lambda$ با تورنومنت

۱- جاگه
 $\mu = 0.5$ دعا هنوز نیست

۲- $\lambda = 70$ نصادری نکنیست

۳- $P_c = 0.7$ حلقه های جستجوی عمده

۴- $P_m = 0.2$ عویض نیز درج

۵- $c \rightarrow f = \frac{1}{c}$

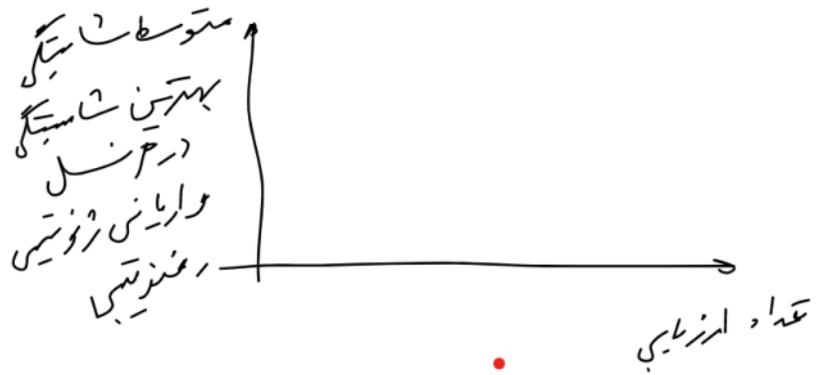
۶- $\mu, \lambda, P_i \propto f_i$

۷- $\mu + \lambda = \text{تقریباً} \lambda$

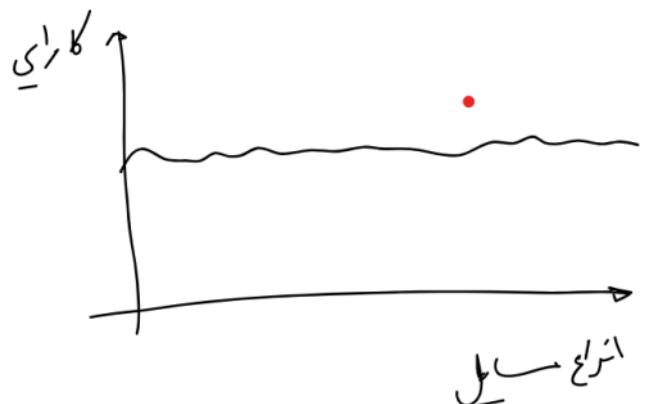
۸- سرتاهاي خاتمه

** بهترین موجود رونگه میداریم

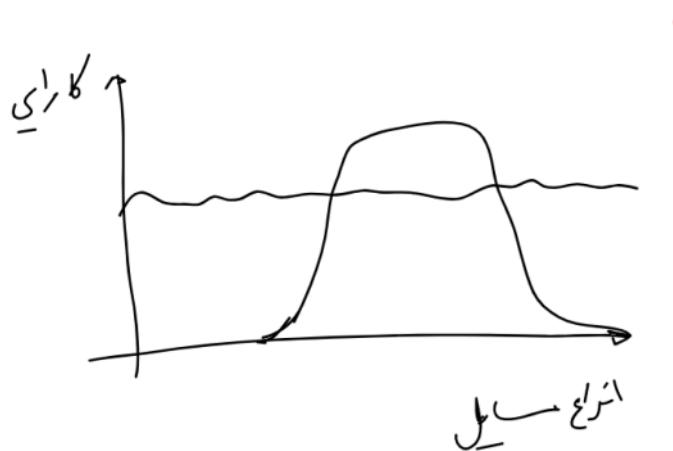
البته برای کارای تحقیقان معمولاً شرط خاتمه رو تعداد نسل های زیاد در نظر میگیرن. منحنی که باید رسم کنیم برا این که بینیم چطوری داره الگوریتم کار میکنه:



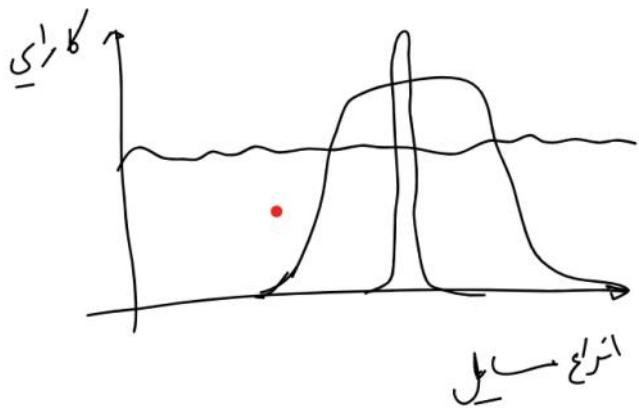
الگوریتم های تکاملی میتوانن عمومی باشن:



برای همه مسائل په کارایی قابل قبولی دارن.
اگر بازنمایی رو جایگشت کنیم همه رو که نمیشه حل کرد بلکه محدوده ای رو میشه با کارایی بهتر حل کرد:



میشه یه الگوریتمی رو خصوصی تر کرد به طوری که بازنگری و جهش رو از دانش اون مساله کمک بگیره. در این صورت یک الگوریتم داریم که برای یک مساله خاص یک جواب خوبی رو فقط داره:



استراتژی های تکامل - Evolutionary Strategies (ES)

الگوریتم های تکاملی به طور کلی کند هستند. یک سری استراتژی میناریم که این سرعت افزایش پیدا کنند. در ES استراتژی هایی در نظر گرفته میشند که سرعت رسیدن به جواب رو افزایش میدهند.

ساده ترین حالتش (1+1) ES است. که یک موجود داریم و یک فرزند تولید میشند. Crossover نداریم و فقط جهش اتفاق میافته.

1. بازنمایی اعداد حقیقی

فرض شود که میخواهیم در یک کشور یک تعداد فروندگاه بسازیم. N شهر و M فروندگاه داریم به طوری که $M < N$ ولی میخواهیم اینارو طوری بسازیم که فاصله هر شهر تا نزدیک ترین فروندگاه مینیمیم بشود.

اگر موقعیت شهر هارو با x_i و y_i نشان دهیم و موقعیت فروندگاه هارا با x_{aj} و y_{aj} نشان دهیم، تعداد Ω ها $2M$ تا باشد به طوری که:

$$\boxed{x_{aj} \quad y_{aj}}$$

2. اگر مختصات شهر هارا از a تا b یعنی $[a, b]$ برای x ها و $[c, d]$ برای y ها در نظر بگیریم، مقداردهی تصادفی اولیه چطوری باید باشه؟

اعداد تصادفی

$$x_{aj} = a + (b - a) U(0, 1)$$

$$y_{aj} = c + (d - c) U(0, 1)$$

با این فرمول اعداد تصادفی یک نوخت بین بازه مورد نظر را تولید میکنیم

3. تابع ارزیابی:

تابع هزینه:

$$\boxed{C_B = \sum_{i=1}^N \sum_{j=1}^M \sqrt{(x_i - x_{aj})^2 + (y_i - y_{aj})^2}}$$

$$f = \frac{1}{C_B}$$

4. جهش:

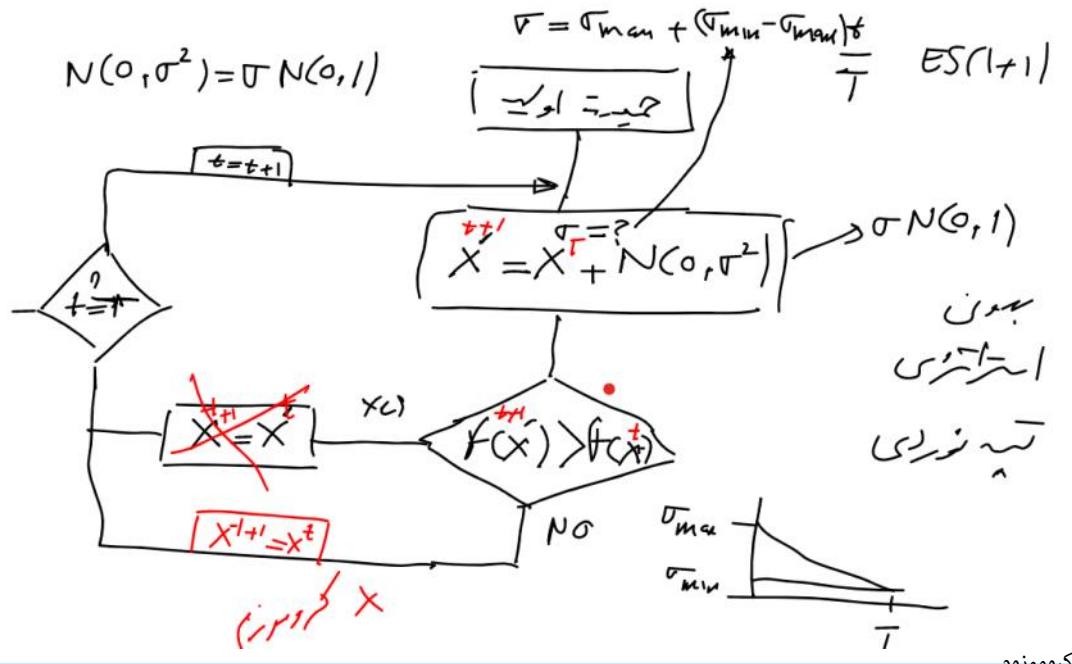
$$\bar{x} = x + N(0, \sigma^2)$$

یعنی \bar{x} قبلی را با یک نویز نرمال با انحراف معیار σ اضافه میکنیم.
از آمار احتمال باید بلد باشیم که:

$$N(0, \sigma^2) = \sigma N(0, 1)$$

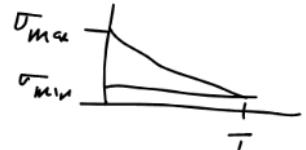
یعنی یک عدد تصادفی نرمال با میانگین 0 و واریانس 1 پیدا کنیم و ضرب کنیم در σ

فلوچارت (ES(1+1) بدون استراتژی تکامل):

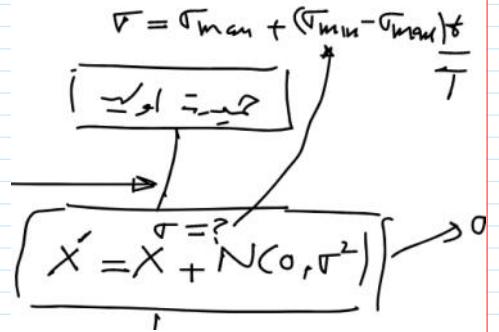


X کروموزوم
* مشابه تپه نورده شد.

برای افزایش سرعت، استراتژی پیشنهادی به این صورت که وقتی نزدیک جواب شدیم قدم رو کوچیک و وقتی دور شدیم اندازه قدم بزرگ بشه. یعنی ۵ یکسان نیست و هرجی به ۷ نزدیک میشیم کمتر میشه:



نحوه محاسبش و جایگاهش:



با این کار با تغییر t از 0 تا T به صورت خطی کم میشه.

اما میشه بهترش کرد؟ یک تر دکترا در آلمان اثبات کرده که در ES(1+1) قانون یک پنجم موفقیت داریم. در این قانون میگه "سرعت وقتی ماکزیمم میشه که تعداد موفقیت ها برابر با 1/5 باشد".

از هر پنج جهش ما یکیش بهتر از والد باشه.

اگر ماتعداد موفقیتمون بیشتر از 1/5 باشه دور از جوابیم یا نزدیکشیم؟ وقتی بیشتر از 1/5 باشه ما دور از جوابیم و هر قدمی که بر میداریم مارو به جواب نزدیک میکنه. در یک فضای سه بعدی یا بعدی اگر ما بخواهیم به یک نقطه برسیم، وقتی فاصلهای زیاد باشه تعداد مسیر هایی که مارو به اون نقطه نزدیک کنن زیاده در نتیجه احتمال این که مسیری که انتخاب میکنیم مارو به جواب برسونه بالا میره < مثلاً جنوب هرآنین میخوایم برم تجربیش. از این مسیر 360 درجه ای 180 درجه شمایلشو انتخاب کنیم حرکت کنیم به تجربیش نزدیک میشیم (چون فاصله زیاده) ولی هرجی به تجربیش نزدیک تر میشیم اون زاویه که ارو به تجربیش هدایت میکنه کوچیکتر و کوچیکتر میشه.

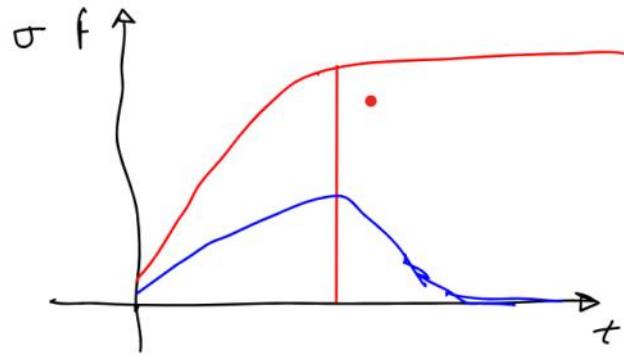
یک دلیل دیگه که وقتی نزدیک جواب میشیم تعداد موفقیت کم میشه: اگر قدم ها زیاد باشه ما جواب رو رد میکنیم. هنگامی که نزدیک جوابیم اندازه قدم خیلی اثر داره ولی وقتی دوریم اهمیت نداره

اطلاعات فوق رو چطوری به سرعت ربطش بدیم؟ اگر احتمال موفقیت P باشه:

$$0 < c < 1 \quad c = 0.8$$

$P_S > \frac{1}{5}$	$\sigma = \frac{\sigma}{c}$	P_S
$P_S = \frac{1}{5}$	$\sigma = \sigma$	
$P_S < \frac{1}{5}$	$\sigma = c\sigma$	

اگر قرمز رو با شایستگی رسم کنیم و آبی رو:



سیگما رو معمولاً از یه حد آستانه هم میذارن کوچکتر نشه، یعنی اگر $\sigma < \sigma_0$ بشود آنگاه $\sigma = \sigma_0$ میگذاریم.

حالا چطوری P_S رو میشه حساب کرد؟ یک راهش متوسط گیری پنجه ای هستش:
فرض کنید از x_1 تا x_k داریم (x متغیر دلخواه) و میخواهیم متوسط گیریم:

$$x_1, x_2, \dots, x_k$$

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_k}{k}$$

حالا اگر x_{k+1} بیاد تو کار و بخوایم از x_2 تا x_{k+1} متوسط گیریم. به طوری که از x متوسط قبلی استفاده کنیم.

$$\bar{x}_{new} = \frac{K \bar{x}_{old} - x_1 + x_{k+1}}{K}$$

الان مشکل اینجاست که الان x_1 دفعه بعد x_2 و ... رو حافظه نگه داریم. یعنی اگر k تا اضافه بشه باید k تایی قبل رو نگه داریم.
پس میایم یک تقریب میزنیم. به این صورت که:

$$\bar{x}_{new} = \frac{K \bar{x}_{old} - x_1 + x_{k+1}}{K} \approx \frac{K-1}{K} \bar{x}_{old} + \frac{1}{K} x_{new}$$

$$\frac{x_{new}}{K} = (1-\beta) \frac{x_k}{K} + \beta x_{knew}$$

حالا اگر $x_{new} = S$ بشه. به طوری که S بشه موفق بودن. اگر موفق باشیم 1 و نباشیم 0. اون موقع x متوسط چی میشه؟ میشه احتمال موفقیت. چرا؟

$$\begin{matrix} x_1 & x_2 & x_3 & x_4 \\ 1 & 0 & 1 & 0 \end{matrix}$$

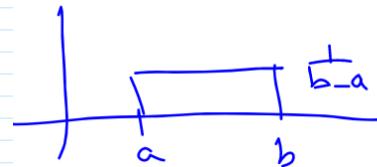
$$\frac{2}{K} = \frac{2}{4} = \frac{1}{2}$$

مثلثا تو ای مثلا اگر موفق باشه 1 و نباشه 0 اون موقع $1/2$ میشه احتمال موفقیت پس میتوان گفت:

$$P_S = (1-\beta)P_S + \beta S$$

S هم برابره 1 عه اگ فرزند بهتره و برابر با 0 اگر فرزند بدتره B هم میشه $1/k$ هم تعداد نسل هایی که احتمال در آن تعداد حساب میشود. اگر $1/100$ روى 100 تا نمونه ولی اگر $1/10$ تا باشه روی 10 تا نمونه.

---- فرق بین تصادفی نرمال و یکنواخت
احتمال انتخاب شدن یک عدد در بازه a تا b برای یکنواخت:



برای نرمال:

بین [-1, 1] باشه میشه 60 و خورده ای درصد. بین [2, -2] میشه نزدیک 100 درصد و برای بقیه نزدیک 0 درصد

$$[-1, 1] \quad 63$$

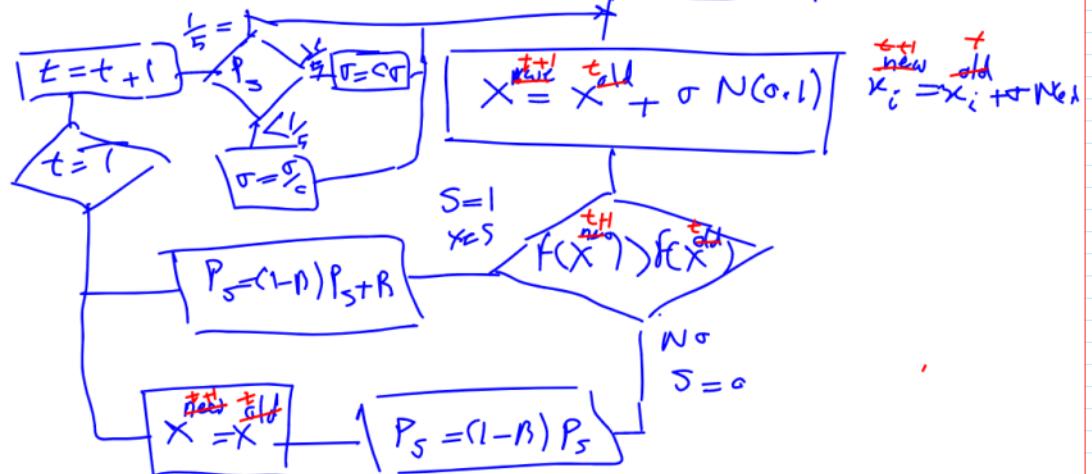


میانگین اعدادی که انتخاب میشه قدر مطلقاتون نزدیک به 1 میشه. به همین دلیل به 5 میگیم اندازه قدم. چون توی اندازه قدم که جهت مهم نیست و با یک توزیع نرمال با اندازه قدم سیگما قدم بر میداریم چون متوسط قدر مطلق تقریبا 1 هست در نتیجه ضریدر 5 بشه میشه 5 و به همین دلیل به 5 میگیم اندازه قدم.

حالا میریم فلوچارت ES(1+1) رو میکشیم:

$$\begin{cases} X_{aj} = a + (b-a) U(0,1) \\ Y_{ej} = c + (d-c) U(0,1) \end{cases}$$

ES(1+1)



X کروموزوم و x زن تقوی وقی که شایستگی رو جک کردیم اگر بدتر بود $x^{t+1} = x^t$ میزاریم چون والد که شایستگی ببیشتری داره باید بره نسل بعد.

ES(1+1) مثل تبه نوردی ممکنه توی مینیمم محلی گیر بکنه. در حالت کالی ولی $(\mu+\lambda)$ ES میتونه فرار کنه.

یک ورزن تصحیح شده از ES(1+1) با ۱/۵ موفقیت وجود داره. که میتونه از مینیمم محلی فرار کنه. به این صورت که یک احتمال ۲۰/۱ هم میگیره. از ۵/۱ کمتر میگه اگر بین ۵/۱ و ۲۰/۱ باشه عین قبل عمل

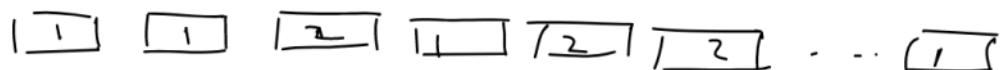
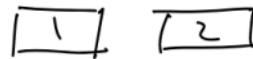
در ES($\mu+\lambda$) در حالت کلی:
بازنمایی اعداد حقیقی است.

جمعیت اولیه تصادفی یکنواخت است. (μ تا جمعیت)
انتخاب والدین تصادفی یکنواخت. $\lambda \approx 7\mu$

در اول جهش انجام میشه بعد بازنگری.

- جهش: نویز نرمال با اندازه قدم ۵ ($x'_i = x_i + \sigma N(0,1)$)
چرا جهش قبل بازنگری؟

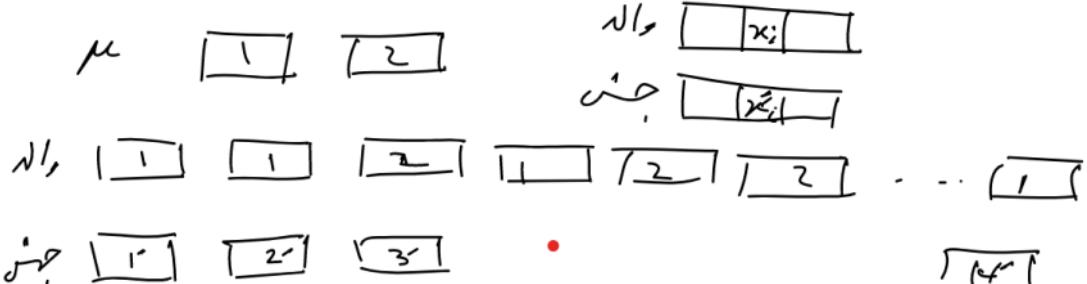
$$\mu = 2 \quad \lambda = 14$$



الان اگر اول بازنگری انجام بدیم چون خیلی شبیه والدین در میاد خوب نیست. اول جهش میدیم که با نویز نرمال اضافه شده با هم متفاوت شن بعد بازنگری انجام میشه.
احتمال جهش $P_n = 1$ در صورتی که $P_c < 0.4$ ***

$$\mu = 2 \quad \lambda = 14$$

$$x'_i = x_i + \sigma N(0,1)$$



سپس موجوداتی که جهش داده شدن ۲ تا ۲ تا کنار هم قرار میگیرن و برای هر چهار گروه میگیرند اگر از ۰.۴ تولید میکنیم اگر از ۰.۴ کمتر شد بازنگری انجام میشه بیشتر شد نمیشه.
چرا بازنگری نقطه ای برای بازنمایی اعداد حقیقی مناسب نیست؟ چون از وسط رُن نمیشه جدا کرد و اعداد حقیقی وسط نداره و فقط میتونه رُن هارو جایه جا کنه و رُن جدیدی تولید نمیشه. بنابراین از بازنگری حسابی انجام میشه.

بازنگری حسابی:

سه نوع داره:

- کامل: برای هر دو رُن

- تکی: برای یک رُن

- ساده: از یک رُن به بعد

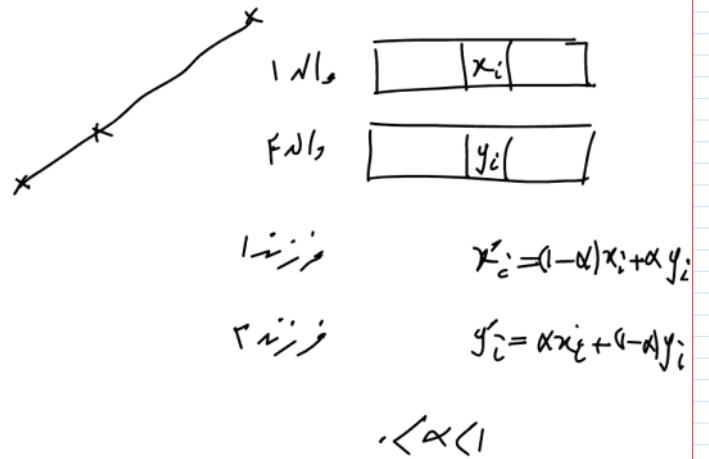
بازنگری حسابی کامل:

X

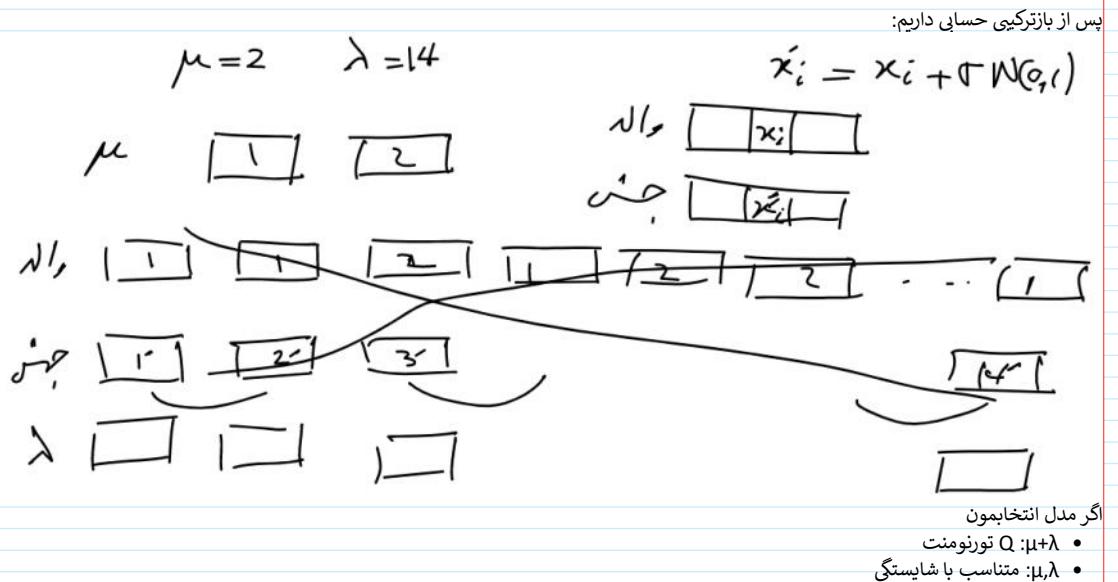
X

این ۲ والد رو فرض کنیم داشته باشیم.

در بازنگری حسابی میگه برای این ک فرزند ویژگی اینارو داشته باشه باید در یک خطی که اینارو به هم وصل میکنه قرار بگیره. که فرمولش میشه:



اگر $\alpha=0.5$ باشه دقیقا اون وسط میافتن جفت‌شون



پس مراحل:

۱- اعداد حقیقی

۲- μ نماد میانگین است

۳- $\lambda = \sigma^2$ دصدای میانگین است

۴- هر سه $\mu = \sigma = \lambda$ نویز حاصل با عزم

۵- $P_{\text{error}} \leq 0.4$ حسای

۶- $\lambda + \mu$ توزیع نمایی است

۷- λ, μ متناسب باشند

۸- λ تراکمی است

استراتژی تعیین اندازه قدم بین $\mu + \lambda$ و $\mu - \lambda$:

۱. پارامتر ثابت: روش خوبی نیست

۲. تابع مشخصه از زمان: مثل تابع خطی یا تابع نمایی

۳. روش تطبیقی: قانون $1/5$ موقوفت

اول این که باید یک فیدبک مناسب پیدا کرد (این که مثلاً توی $ES(1+1)$ موقوفت بود برای بقیه که نیست) این که $1/5$ از کجا بدست میاد هم مورد نیاز است (توی $ES(1+1)$ رو میدونیم $1/5$ عه برای بقیش که نمیدونیم)

اگر ۲ مورد فوق رو بشه پیدا کرد بهترین روش

۴. خود تطبیقی: اگر از ۳ نمونی استفاده کنیم میایم سراغ این روش که خودش خودشو تطبیق میده

در روش خود تطبیقی میایم یک زن هم برای اندازه قدم مینذاریم

$$\frac{1}{n} \sum_{i=1}^n x_i$$

و تکامل خودش باید تعیینش کنه.

در این زن جدید فقط جهش انجام میشه و بازنگری رخ نمیده. جهشش هم با زن های اصلی متفاوته، اگر مقدار جهش یافته سیگما رو σ بنامیم، داریم:

$$x' = \bar{x} - \sigma N(0, 1)$$

$$x' = x_i + \sigma N(0, 1)$$

به دو دلیل جهش فرق میکنه و نماییه به جای خطی:

• یک دلیل این که سیگما نمیخوایم منفی بشه و میخوایم یا کوچیکتر بشه یا بزرگتر.

• این کار تاریخی رخ بده و اگر فرآنس تغییراتش خیلی زیاد بشه تکامل نمیتوونه بهمه خوبه یا بده به ضریب تاو که در نمایی میاد میگیم ضریب یا دگری و عدد کوچیکه.

سوال: اول سیگما رو جهش بدیم بعد x یا بالعکس؟

$$x' = x + \sigma' N(0, 1)$$

$$\sigma' = \sigma e^{-\lambda N(0, 1)}$$

حالت اول

$$\sigma' = \sigma e^{-\lambda N(0, 1)}$$

حالت دوم

$$x' = x + \sigma' N(0, 1)$$

ما هیچ معیاری برای این که بفهمیم اندازه قدم خوب هست یا نه نداریم. در نتیجه باقیستی بر اساس شایستگی که از x بدست میاریم بگیم خوب بوده. چون x خوب بوده پس اندازه قدمش هم خوب بوده (ولی 100 درصد هم اینطوری نیست) چون اندازه قدم در یک نویزی داره ضرب میشه).

بنابراین برای این که بفهمیم خوب هست یا نه اول x رو جهش میدیم بعد شایستگی رو حساب میکنیم و احتمال میدیم که اندازه قدم خوب باشه. ← حالت دوم رخ میده و اول سیگما جهش یافته میشه و با سیگما جهش یافته شده x رو میریم میبینیم بهتر شد یا نه.

پس با خود تطبیقی:

x_1	x_n	σ
-------	-------	----------

۱- اعداد حقیقی

۲- μ نمادی کلیز است

۳- $\lambda = \gamma \mu$ حصادی کلیز است

۴- جریان $P_m = 1 - \text{نزیرزیال}(\mu)$ با عدم σ

۵- بازرسی $P_c < 0.4$ حایی

۶- $\lambda + \mu$ و λ توزیع
۷- μ, λ متناسب بستگی

۸- جریان تراکمی



یک حالت دیگه هم خود تطبیقی داریم که صرفًا بیان میشه سر کلاس:
در این روش برای هر زن یک سیگما هم در نظر میگیریم:

x_1	x_n	σ_1	σ_n
-------	-------	------------	------------

فرض کنیم یک ولد داریم:

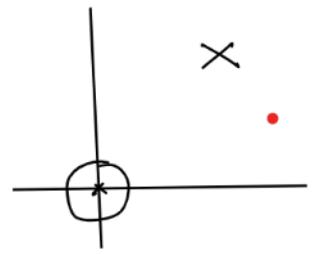
•

x

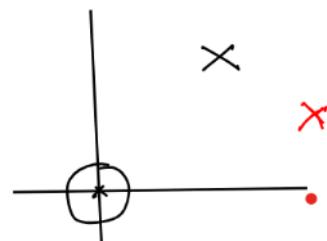
جواب هم اینجاست:

X

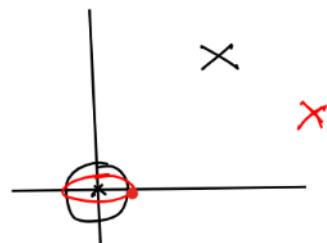
با جهش احتمال این که جواب در این دایره بیافته وجود دارد:



ولی اگر یک سیگما 1 و سیگما 2 داشته باشیم. اگر الگوریتم بتونه سیگما هارو درست تشخیص بده و جواب اینجا باشه:



در جهت x_1 باید بیشتر از x_2 حرکت کنه. اون موقع با چند سیگما به شکل بیضی در میشه آورد:



ابنطوري شانس این که به جواب برسیم بیشتره. یه حالت دیگه هم هست بیضی رو بخرخونیم به سمت جواب که خبلی پیچیده میشه و در موردش حرف نمیزنیم.