بخش نوشتاری

سوال اول

مثال‌هایی از Regression:

۱. فرض کنید می خواهیم پیش بینی آب و هوا را انجام دهیم، بنابراین برای این کار از الگوریتم رگرسیون استفاده می کنیم. در پیش بینی آب و هوا، مدل بر روی داده های قدیمی که جمع آوری کرده‌ایم، آموزش داده می شود و پس از اتمام آموزش، به راحتی می تواند هوای روزهای آینده را پیش بینی کند.
۲. محققان پزشکی اغلب از رگرسیون خطی برای درک رابطه بین دوز دارو و فشار خون بیماران استفاده می کنند. به عنوان مثال، محققان ممکن است دوزهای مختلفی از یک داروی خاص را برای بیماران تجویز کنند و نحوه پاسخ فشار خون آنها را مشاهده کنند.
۳. از رگرسیون خطی می توان برای تعیین تأثیرات نسبی سن، جنسیت و رژیم غذایی (متغیرهای پیش بینی کننده) بر قد (متغیر نتیجه) استفاده کرد.
۴. دانشمندان کشاورزی اغلب از رگرسیون خطی برای اندازه گیری اثر کود و آب بر عملکرد محصول استفاده می کنند.
۵. دانشمندان داده برای تیم‌های ورزشی حرفه‌ای اغلب از رگرسیون خطی برای اندازه‌گیری تأثیر رژیم‌های تمرینی مختلف بر عملکرد بازیکنان استفاده می‌کنند.

مثال‌هایی از Classification:

۱. می‌توان از دسته‌بندی پیش‌بینی گرم و سرد بودن آب و هوا استفاده کرد.در واقع برای گروه‌بندی دماها به دو دسته‌ی گرم و سرد(یا دسته‌های بیشتر مانند مرطوب، شرجی و ...) استفاده کرد.
۲. در ماشین‌های خودران، برای تشخیص اشیاء خارج از ماشین و دسته‌بندی آن‌ها به پیاده‌رو و موتورسیکلت و ماشین‌های دیگر.
۳. Binary classification یکی از انواع دسته‌بندی است. برای مثال میتوان spam بودن یا نبودن ایمیل را مشخص کند.
۴. پیش‌بینی ریزش(churn) به معنای شناسایی مشتریانی است که احتمالاً یک سرویس را ترک می‌کنند یا اشتراک یک سرویس را لغو می‌کنند. این یک پیش‌بینی حیاتی برای بسیاری از کسب‌وکارها است، زیرا گرفتن مشتریان جدید اغلب بیشتر از حفظ مشتریان هزینه می‌برد.
۵. ژانرهای کتاب یکی از نمونه های رایجی است که از دسته‌بندی استفاده می‌کند.

سوال دوم

بخش اول:

صحت(Accuracy): تعیین می‌کند یک مجموعه اندازه گیری معین چقدر به مقدار واقعی خود نزدیک یا دور است.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP = True positive; FP = False positive; TN = True negative; FN = False negative

دقت(Precision): برای ما مشخص می‌کند، که پاسخ‌های مثبت درست، چه درصدی از کل پاسخ های درست را تشکیل می‌دهد. در واقع می‌خواهیم مطمئن شویم که تعداد پاسخ‌های مثبت واقعی از مثبت کاذب کمتر باشد.

$$\text{Precision} = \frac{tp}{tp + fp}$$

پوشش(Recall): نشان می‌دهد در چه درصدی از داده‌ها به درستی پوشش داده شده‌اند.

$$\text{Recall} = \frac{tp}{tp + fn}$$

F1-score: میانگین وزنی دقت و پوشش است. بنابراین، این امتیاز هم مثبت کاذب و هم منفی کاذب را در نظر می گیرد. به طور شهودی درک آن به اندازه دقت آسان نیست، اما امتیاز اف معمولا مفیدتر از دقت است، به خصوص اگر توزیع کلاس ناهمواری داشته باشید.

$$F_1 = \frac{2}{\text{recall}^{-1} + \text{precision}^{-1}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\text{tp}}{\text{tp} + \frac{1}{2}(\text{fp} + \text{fn})}.$$

بخش دوم:

زمانی که مدل نتواند هر گونه خرابی را پیش بینی کند، صحت آن هنوز ۹۰٪ است. از آنجایی که ۹۰٪ داده ها درست تشخیص داده شده‌اند. بنابراین، صحت برای داده های نامتعادل خوب نیست. در سناریوهای تجاری، بیشتر داده ها متعادل نمی شوند و بنابراین صحت به معیار ضعیف ارزیابی برای مدل طبقه بندی ما تبدیل می شود.

۱. اگر ما در حال توسعه سیستمی هستیم که کلاهبرداری در تراکنش های بانکی را شناسایی می کند، مطلوب است که پوشش بسیار بالایی داشته باشیم تا اکثر تراکنش های تقلبی شناسایی می شوند که این اتفاق با کاهش دقت همراه است، زیرا بسیار مهم است که همه تقلب ها شناسایی شوند یا حداقل شبهات مطرح می شود.در مقابل، اگر منبعی از داده‌ها مانند توییتر داشته باشیم و علاقه‌مندیم بفهمیم

که توییت چه زمانی احساسات منفی را در مورد یک سیاستمدار خاص بیان می‌کند، احتمالاً می‌توانیم دقت را (برای کسب اطمینان) به قیمت از دست دادن پوشش افزایش دهیم، زیرا ما چیز زیادی را از دست نمی‌دهیم. در این مورد چیز زیادی از دست نمی دهیم و به هر حال منبع داده بسیار عظیم است. در این مدل بهتر است به جای صحت از پوشش استفاده شود.

۲. تصور کنید که روی داده های فروش یک وب سایت کار می کنید. می دانید که ۹۹ درصد از بازدیدکنندگان وب سایت خرید نمی کنند و تنها ۱ درصد از بازدیدکنندگان چیزی می خرند. شما در حال ساخت یک مدل طبقه بندی هستید تا پیش بینی کنید کدام بازدیدکنندگان وب سایت خریدار هستند و کدام یک فقط تماشاگر. حالا مدلی را تصور کنید که خیلی خوب کار نمی کند. پیش‌بینی می‌کند که ۱۰۰٪ بازدیدکنندگان شما فقط تماشاگر هستند و ۰٪ بازدیدکنندگان شما خریداران هستند. واضح است که این یک مدل بسیار اشتباه و بیهوده است. این مدل تنها ۱٪ را به اشتباه پیش‌بینی کرده است: همه خریداران به اشتباه به عنوان ناظر طبقه‌بندی شده‌اند. بنابراین درصد پیش‌بینی‌های صحیح ۹۹ درصد است. مشکل اینجاست که دقت ۹۹٪ نتیجه عالی به نظر می رسد، در حالی که مدل شما عملکرد بسیار ضعیفی دارد. در نتیجه: دقت معیار خوبی برای استفاده در هنگام عدم تعادل کلاس نیست. یکی از راه‌های حل مشکل عدم تعادل کلاس استفاده از معیارهای دقت بهتر مانند امتیاز F1 است که نه تنها تعداد خطاهای پیش‌بینی مدل را در نظر می‌گیرد، بلکه به نوع خطاهای ایجاد شده نیز توجه می‌کند.

۳. فرض کنید مجموعه داده ای از ۱۰۰۰ بیمار داریم که از این تعداد ۸۰ بیمار سرطانی و بقیه (۹۲۰ نفر) سالم هستند. این نمونه ای از یک مجموعه داده نامتعادل است، زیرا کلاس اکثریت حدود ۹ برابر بزرگتر از کلاس اقلیت است. در اینجا طبقه اکثریت سالم است و طبقه اقلیت دارای سرطان است. چنین مجموعه داده ای مجموعه داده نامتعادل است. صحت در اینجا بیشتر درصد مردم را سالم تشخیص می‌دهد و نتایج خوبی را اعلام می‌کند که اینجا خواسته‌ی مسئله نیست. بهتر است با توجه به نیاز مسئله از سایر روش‌های ارزیابی استفاده کنیم. می‌توان از امتیاز F1 استفاده کرد چون دقت و پوشش را در نظر می‌گیرد.

سوال سوم

	بیماری قلبی دارد	عروق خونی بسته	گردش خون مناسب	درد سینه
-	خیر	خیر	خیر	خیر
+	بله	بله	بله	بله
-	خیر	خیر	بله	بله
+	بله	بله	خیر	بله

$$entropy = -p(+)\log(p(+)) - p(-)\log(p(-))$$

تعداد متغیرها = 2 تعداد مثبت‌ها = 2 کل داده‌ها = 4

$$entropy = -\frac{2}{4}\log\left(\frac{2}{4}\right) - \frac{2}{4}\log\left(\frac{2}{4}\right) = 0,3010\,2$$

$gain = entropy - info$ به ازای هر feature، آن را محاسبه می‌کنیم.

درد سینه

خیر: $-\frac{0}{1}\log\left(\frac{0}{1}\right) - \frac{1}{1}\log\left(\frac{1}{1}\right) = 0$ [0+,1-]

بله: $-\frac{2}{3}\log\left(\frac{2}{3}\right) - \frac{1}{3}\log\left(\frac{1}{3}\right) = 0,11739 + 0,1590 = 0,04165$ [2+,1-]

$$info = \frac{1}{4}\times 0 + \frac{3}{4}\times 0,04165 = 0,03123$$

$$gain = 0,3010\,2 - 0,03123 = 0,26979$$

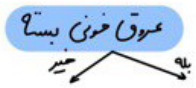
گردش خون مناسب

خیر: $-\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) = 0,3010\,2$ [1+,1-]

بله: $-\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) = 0,3010\,2$ [1+,1-]

$$info = \frac{2}{4}\times 0,3010\,2 + \frac{2}{4}\times 0,3010\,2 = 0,3010\,2$$

$$gain = 0,3010\,2 - 0,3010\,2 = 0$$



$-\frac{0}{2}\log\left(\frac{0}{2}\right)-\frac{2}{2}\log\left(\frac{2}{2}\right)=0$

$-\frac{2}{2}\log\left(\frac{2}{2}\right)-\frac{0}{2}\log\left(\frac{0}{2}\right)=0$

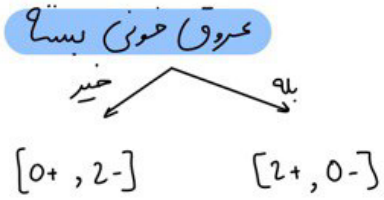
[0+, 2-] [2+, 0-]

$info = \frac{2}{4}\times 0 + \frac{2}{4} = 0$

$gain = 0,30102 - 0 = 0,30102$

با مقایسه‌ی 3 تا gain به دست آمده متوجه می‌شویم که عروق خونی بسته بیشترین gain را دارد.

با قرار گرفتن عروق خونی بسته در ریشه دیگر نیازی به محاسبه‌ی ریشه‌ها نیست چون در هر برگ آنتروپی صفر است. نمودار درختی به صورت زیر است:



سوال چهارم

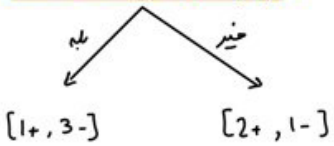
	سریال کلاه‌قرمزی را دوست دارد؟	سن	آب گازدار دوست دارد؟	پاپ‌کرن دوست دارد؟
−	خیر	۷	بله	بله
−	خیر	۱۲	خیر	بله
+	بله	۱۸	بله	خیر
+	بله	۳۵	بله	خیر
+	بله	۳۸	بله	بله
−	خیر	۵۰	خیر	بله
−	خیر	۸۳	خیر	خیر

$$entropy = -p(+)\log(p(+)) - p(-)\log(p(-))$$

$$entropy = -\frac{4}{7}\log\left(\frac{4}{7}\right) - \frac{3}{7}\log\left(\frac{3}{7}\right) = 0,13887 + 0,1577 = 0,29657$$

gain = entropy − info به ازای هر feature، آن entropy را حساب می‌کنیم.

پاپ‌کرن دوست دارد؟



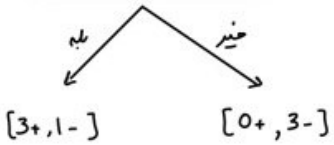
$$-\frac{1}{4}\log\left(\frac{1}{4}\right) - \frac{3}{4}\log\left(\frac{3}{4}\right) = 0,15051 + 0,0937 = 0,24421$$

$$-\frac{2}{3}\log\left(\frac{2}{3}\right) - \frac{1}{3}\log\left(\frac{1}{3}\right) = 0,11739 + 0,1590 = 0,27639$$

$$info = \frac{4}{7} \times 0,24421 + \frac{3}{7} \times 0,27639 = 0,13954 + 0,118452 = 0,258$$

$$gain = 0,29657 - 0,258 = 0,03857$$

آب گازدار دوست دارد؟



$$-\frac{3}{4}\log\left(\frac{3}{4}\right) - \frac{1}{4}\log\left(\frac{1}{4}\right) = 0,24421$$

$$-\frac{0}{3}\log\left(\frac{0}{3}\right) - \frac{3}{3}\log\left(\frac{3}{3}\right) = 0$$

$$info = \frac{4}{7} \times 0,24421 + \frac{3}{7} \times 0 = 0,13954$$

$$gain = 0,29657 - 0,13954 = 0,157021$$

ابتدا باید عددها را مرتب شده کنیم، دوم بزرگتر، سپس نقاط جداکننده را مشخص کنیم و برای هرکدام gain را محاسبه کنیم.

7	12	18	35	38	50	83
-	-	+	+	+	-	-

$\frac{12+18}{2} = 15$ $\frac{38+50}{2} = 44$

باید برای دو حالت gain را محاسبه کنیم و بیشترین gain را انتخاب کنیم:

سن

<15 >15

[0+,2-] [3+,2-]

$-\frac{0}{2}\log(\frac{0}{2}) - \frac{2}{2}\log(\frac{2}{2}) = 0$

$-\frac{3}{5}\log(\frac{3}{5}) - \frac{2}{5}\log(\frac{2}{5}) = 0,1331092 + 0,15917 = 0,29228$

$info = \frac{2}{7} \times 0 + \frac{5}{7} \times 0,29228 = 0,208771$

$gain = 0,29657 - 0,208771 = 0,08779$

سن

<44 >44

[3+,2-] [0+,2-]

$-\frac{0}{2}\log(\frac{0}{2}) - \frac{2}{2}\log(\frac{2}{2}) = 0$

$-\frac{3}{5}\log(\frac{3}{5}) - \frac{2}{5}\log(\frac{2}{5}) = 0,1331092 + 0,159176 = 0,29228$

$info = \frac{2}{7} \times 0 + \frac{5}{7} \times 0,29228 = 0,208771$

$gain = 0,29657 - 0,208771 = 0,08779$

بین gain های بدست آمده، سوال آب بازی دار دوست دارد بیشترین gain را دارد. در نتیجه در ریشه قرار می‌گیرد.

برای شاخه‌های بعدی مانند قبل gain را محاسبه می‌کنیم تا زیرشاخه ما مشخص شود.

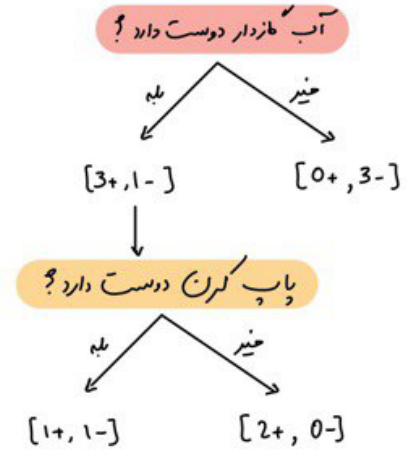
شاخه سمت راست دارای entropy صفر است بنابراین نیاز برای شاخه چپ محاسبه می‌کنیم.

آب بازدار دوست دارد؟

بله خیر

[3+,1-] [0+,3-]

ریشه‌ی درخت

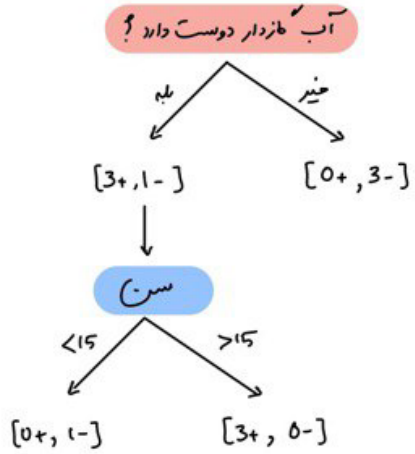


$$-\frac{1}{2}\log\left(\frac{1}{2}\right)-\frac{1}{2}\log\left(\frac{1}{2}\right)=0,30102$$

$$-\frac{2}{2}\log\left(\frac{2}{2}\right)-\frac{0}{2}\log\left(\frac{0}{2}\right)=0$$

$$info=\frac{2}{4}\times 0,30102+\frac{2}{4}\times 0=0,15051$$

$$gain=0,24421-0,15051=0,0937$$

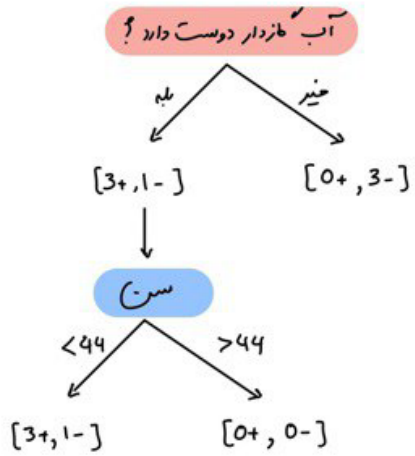


$$-\frac{0}{1}\log\left(\frac{0}{1}\right)-\frac{1}{1}\log\left(\frac{1}{1}\right)=0$$

$$-\frac{3}{3}\log\left(\frac{3}{3}\right)-\frac{0}{3}\log\left(\frac{0}{3}\right)=0$$

$$info=\frac{1}{4}\times 0+\frac{3}{4}\times 0=0$$

$$gain=0,24421-0=0,24421$$



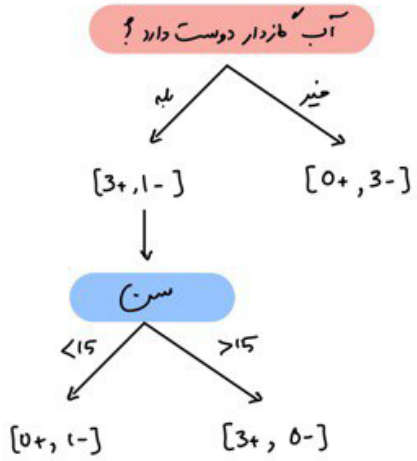
$$-\frac{3}{4}\log\left(\frac{3}{4}\right)-\frac{1}{4}\log\left(\frac{1}{4}\right)=0,24421$$

$$-\frac{0}{0}\log\left(\frac{0}{0}\right)-\frac{0}{0}\log\left(\frac{0}{0}\right)=0$$

$$info=\frac{4}{4}\times 0,24421=0,24421$$

$$gain=0,24421-0,24421=0$$

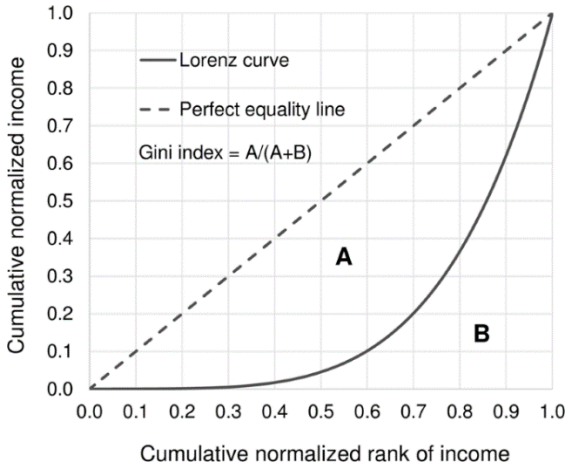
بین همه‌ی ویژگی‌ها سن که با ۱۵ نقطه شکسته شده است بیشترین gain را دارد و چون شاخه‌های آن دارای entropy صفر هستند دیگر نیازی به محاسبه‌ی موارد دیگر نیست. بنابراین درخت ما به صورت زیر خواهد بود :



سوال پنجم

شاخص جینی (یا ضریب) یک شاخص ترکیبی است که سطح نابرابری را برای یک متغیر و جمعیت معین نشان می‌دهد. این اعداد بین ۰ (برابری کامل) و ۱ (نابرابری شدید) متغیر است. بین ۰ و ۱، هر چه شاخص جینی بالاتر باشد، نابرابری بیشتر است. همانطور که گفته شد، ضریب جینی ۰ برابری کامل را بیان می کند، جایی که همه مقادیر یکسان هستند (یعنی جایی که همه درآمد یکسانی دارند).

شاخص جینی با نسبت مساحت بین خط برابری کامل(نقطه چین در تصویر زیر) و منحنی لورنز (A) تقسیم بر مساحت کل زیر خط برابری کامل (A + B) محاسبه می شود.



از شاخص دیگری که شبیه به این شاخص هستند ی‌توان به information gain و آنتروپی اشاره کرد که برای ساختن درخت از آن استفاده می‌شود.

مثال:

Past Trend	Open Interest	Trading Volume	Return
Positive	Low	High	Up
Negative	High	Low	Down
Positive	Low	High	Up
Positive	High	High	Up
Negative	Low	High	Down
Positive	Low	Low	Down
Negative	High	High	Down
Negative	Low	High	Down
Positive	Low	Low	Down
Positive	High	High	Up

P(Past Trend=Positive): ۶/۱۰

P(Past Trend=Negative): ۴/۱۰

If (Past Trend = Positive & Return = Up), probability = ۴/۶

If (Past Trend = Positive & Return = Down), probability = ۲/۶

Gini index = ۱ - ((۴/۶)^۲ + (۲/۶)^۲) = ۰.۴۵

If (Past Trend = Negative & Return = Up), probability = ۰

If (Past Trend = Negative & Return = Down), probability = ۴/۴

Gini index = ۱ - ((۰)^۲ + (۴/۴)^۲) = ۰

Gini Index for Past Trend = (۶/۱۰)*۰.۴۵ + (۴/۱۰)*۰ = ۰.۲۷

سوال ششم

بیش برازش یا overfitting: در این حالت مدل ما به خوبی به داده های آموزشی fit شده است ولی با داده های جدید و تست نمیتواند خود را تطبیق دهد. این حالت معمولا وقتی رخ میدهد که تعداد پارامترها زیاد است و مدل زیاد آموزش دیده است. در واقع زمانی اتفاق می‌افتد که مدل ما به داده‌های آموزش بسیار وابسته می‌شود.
زمانی که الگوریتم یادگیری ماشین از مجموعه داده های آموزشی بسیار بزرگتری در مقایسه با مجموعه تست استفاده می کند. این سبب می‌شود که دقت در فضای کوچکتر کاهش پیدا کند و در الگوهای بزرگ آموزش ببیند.
زمانی که الگوریتم یادگیری ماشین از پارامترهای زیادی برای مدل سازی داده های آموزشی استفاده می کند.
برای جلوگیری و رفع بیش برازش می‌توان راه‌های زیر را استفاده کرد:

- توقف زودهنگام: این روش به دنبال توقف آموزش قبل از شروع یادگیری نویز درون مدل توسط مدل است.
- آموزش با داده‌های بیشتر: گسترش مجموعه آموزشی برای گنجاندن داده‌های بیشتر می‌تواند دقت مدل را با فراهم کردن فرصت‌های بیشتر برای تجزیه و تحلیل رابطه غالب بین متغیرهای ورودی و خروجی افزایش دهد.
- افزایش داده ها(Data augmentation): در حالی که بهتر است داده های تمیز و مرتبط را به داده های آموزشی خود تزریق کنید، گاهی اوقات داده های نویز برای پایدارتر کردن مدل اضافه می شود.
- انتخاب ویژگی: وقتی یک مدل می‌سازید، تعدادی پارامتر یا ویژگی خواهیم داشت که برای پیش‌بینی یک نتیجه معین استفاده می‌شوند، اما بسیاری از اوقات، این ویژگی‌ها ممکن است برای دیگران زائد باشد. انتخاب ویژگی فرآیند شناسایی مهمترین آنها در داده های آموزشی و سپس حذف موارد نامربوط یا اضافی است.

- منظم سازی(Regularization): اگر بیش برازش اتفاق بیفتد، به این دلیل است که مدل پیچیده است، منطقی است که تعداد ویژگی ها را کاهش دهیم. اگر نمی‌دانیم کدام ویژگی‌ها را از مدل خود حذف کنیم، روش‌های منظم‌سازی می‌تواند بسیار مفید باشد. منظم‌سازی یک "جریمه" برای پارامترهای ورودی با ضرایب بزرگ‌تر اعمال می‌کند، که متعاقباً مقدار واریانس در مدل را محدود می‌کند.
- Dropout: در این روش برای همه نورون ها به غیر از نورون های آخر یک عدد تصادفی تولید میکنیم. آن نورون هایی که عدد تصادفی آنها کمتر از ۰/۵ است را علامت گذاری کرده و بعد تمام وزنهای ورودی و خروجی به آنها را حذف میکنیم. با این کار نقش نورون های بلا استفاده را حذف کرده و شبکه را سبکتر میکنیم. در نتیجه منحنی تولید شده پیچیده نیست و بیش برازش رخ نمیدهد.