

# شبکه های عصبی

وحیدکریمی پور- دانشکده فیزیک - دانشگاه صنعتی شریف

۱۴ تیر ۱۳۹۹

---

## ۱ مقدمه

یک سوال دیرینه این است که چگونه مغز می تواند این حجم عظیم از اطلاعات گوناگون را به حافظه بسپارد و آن را به یاد بیاورد. چگونه می تواند تصاویر و اشکال را شناسایی کند، چگونه می تواند محاسبه کند و سرانجام اینکه چگونه می تواند فکر کند. تقریباً همه این قابلیت ها را کامپیوترهای امروزی نیز دارند ولی میزان این قابلیت ها در مغز و کامپیوتر بسیار متفاوت است. به عنوان مثال در حالیکه کامپیوتر می تواند در کسری از ثانیه اعداد بیست رقمی را در هم ضرب کند، یک مغز متعارف در ضرب کردن اعداد دورقمی نیز کند است. از طرف دیگر در حالیکه کامپیوتر ظرفیت محدودی برای ذخیره کردن طرح ها (تصاویر، اعداد، صداها و کلمات...) دارد، به نظر می رسد یک مغز متعارف ظرفیت نامحدودی برای ذخیره کردن طرح ها دارد. در طول یک روز مغز ما میلیون ها طرح را از طریق اعصاب بینایی، شنوایی، لامسه و بویایی دریافت می کند ولی سرانجام بسیاری از این طرح ها را طی فرآیندی که هنوز کاملاً شناخته شده نیست، دور می ریزد. ما قیافه تمام آدمهایی را که در خیابان می بینیم و صدای بوق ماشین ها و آدمها را برای همیشه به حافظه نمی سپاریم. برخلاف کامپیوتر مغز این طرح ها را بر اساس آدرس آنها ذخیره نمی کند و هنگام به یادآوردن آنها نیز آنها را بر اساس آدرس یا ترتیب دریافت و ذخیره شدن آنها به یاد نمی آورد بلکه بر اساس محتوی و بیشتر بر اساس وابستگی آنها به هم به یاد می آورد. این چیزی است که به آن تداعی معانی می گویند یعنی اینکه یک طرح و خاطره می تواند طرح ها و خاطره های دیگر را به یاد بیاورد.



شکل ۱: جان هاپفیلد (۱۹۳۳-)، فیزیکدان امریکایی که به خاطر ابداع مدلی برای حافظه وابسته در سال ۱۹۸۵ شناخته شده است.



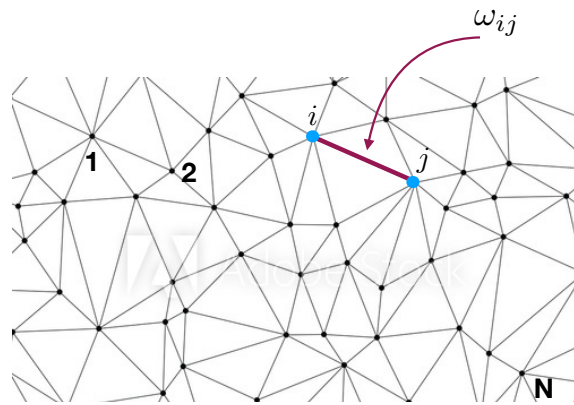
شکل ۲: مغز با دیدن چند نوع فنجان مفهوم و شکل کلی فنجان را یاد می‌گیرد و طبقه‌بندی می‌کند. بعد از آن هر فنجانی را از هر شکل و رنگ و اندازه‌ای و هر جهتی حتی فنجان‌های شکسته را نیز در همان طبقه جای می‌دهد و شناسایی می‌کند. کاری که از کامپیوتر ساخته نیست.

یکی دیگر از تفاوت‌های مهم مغز و کامپیوتر قدرت طبقه‌بندی و انتزاع<sup>۱</sup> مغز است. به این ترتیب که می‌تواند از تصاویر نشان داده شده در شکل (۲) و نظایر آن مفهوم کلی فنجان را ساخته و هم طرح‌های مشابه دیگر را حتی اگر مثل یکی از طرح‌های نشان داده شده در این شکل نباشند تحت نام فنجان طبقه‌بندی کند.

هم‌چنین در حالیکه توانایی کامپیوترها برای تشخیص طرح‌ها بسیار محدود است و این همان خاصیتی است که سایت‌های اینترنتی برای تمیزدادن کاربرهای انسانی از کاربرهای اتوماتیک از آن استفاده می‌کنند، کوچکترین و ابتدایی‌ترین موجودات زنده مثل زنبور عسل یا قورباغه توانایی حیرت‌انگیزی در شناسایی دقیق طرح‌ها دارند حتی اگر این طرح‌ها ناقص باشند یا با سرعت زیاد در حال حرکت باشند.

---

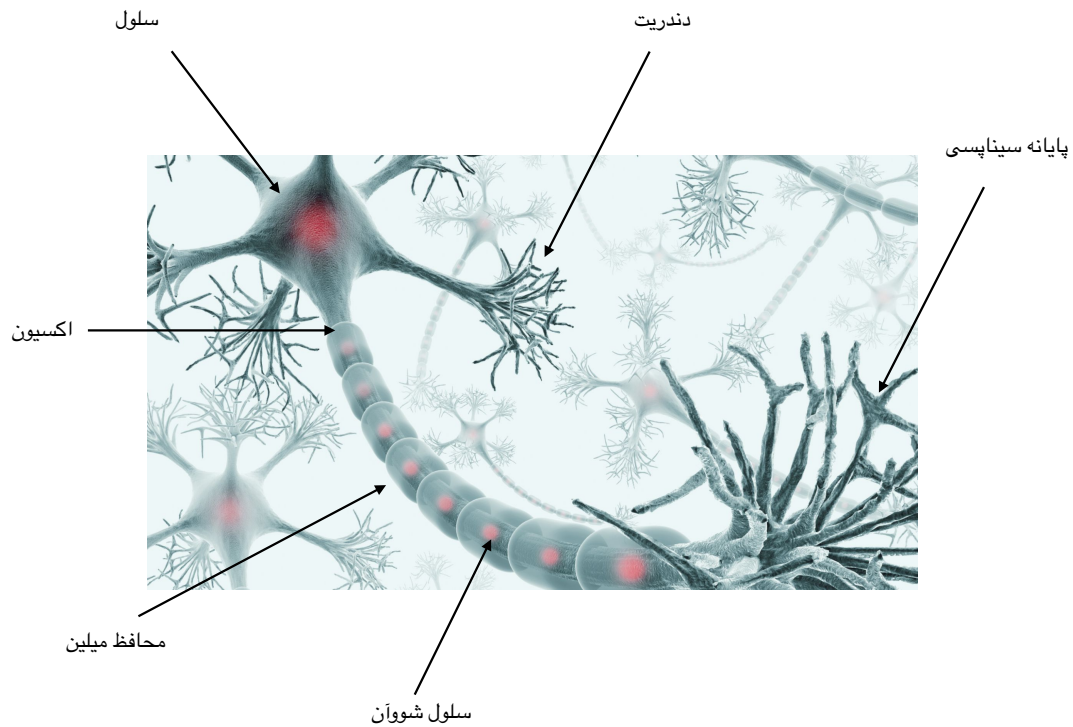
Classification and Abstraction<sup>1</sup>



شکل ۳: شمایی از یک شبکه عصبی برای حافظه.

مغز یک موجود زنده در هر لحظه هزاران کار مختلف، از دریافت و آنالیز داده های ورودی حسی گرفته تا کنترل تمام اعمال ارادی و غیر ارادی و اعمال فیزیولوژیک را انجام می دهد. به همه این ها نیز می بایست فکر کردن، و به یادآوردن و تصور کردن را نیز اضافه کرد. با وجود اینکه پیشرفت های خیلی زیادی در دهه های گذشته در مورد شناخت مغز صورت گرفته می توان گفت که به طور نسبی هیچ چیز در باره مغز و کارکرد آن نمی دانیم. دانش ما برای پاسخگویی به سوالاتی که در ابتدای این درس طرح شد، بسیار جزئی و ناقص است. با این وجود مدل شبکه های عصبی می تواند به ما کمک کند تا نخستین گام ها را در این ارتباط برداریم. بدیهی است که این مدل در مقایسه با واقعیت بسیار ابتدایی است و تنها می تواند مدل بسیار ساده ای باشد که به تدریج با مدل های پیچیده تر یا با مدل های به کلی متفاوت جایگزین شود. مشاهدات میکروسکوپی که سابقه آنها به اوایل قرن بیستم می رسد نشان می دهند که شبکه عصبی در تقریباً تمامی موجودات زنده از مجموعه عظیمی از سلول های کوچک به نام نورون تشکیل شده که با اتصالاتی موسوم به اتصالات سیناپسی با یک دیگر مرتبط هستند. شبکه عصبی انسان که مرکز آن در مغز قرار دارد و از آنجا به نخاع و دیگر اندام ها گسترده است از حدود  $10^{11}$  نورون تشکیل شده که با حدود  $10^{15}$  اتصال سیناپسی با یکدیگر مرتبط اند. بنابراین به طور متوسط یک نورون با ده هزار نورون دیگر در ارتباط است. به این ترتیب می توان شبکه عصبی را با یک گراف که شامل  $N$  نقطه است و تعداد  $L$  اتصال است نشان داد، شکل (۳).

فرض می شود که در حالت عادی یک نورون در حالت خاموش یا غیرفعال قرار دارد اما تحت شرایطی می تواند فعال شود. حالت خاموش یا غیرفعال  $^2$  را با 0 و حالت روشن یا فعال  $^3$  را با 1 مشخص می کنیم.



شکل ۴: شمای کلی از یک نورون. پالس های الکتریکی از نورون به درون سیناپس منتقل می شود و از آنجا نورون های دیگر را تحت تاثیر قرار می دهد.

به این ترتیب هر نورون  $i$  را با یک متغیر  $n_i$  نشان می دهیم که می تواند دو مقدار صفر و یک را اختیار کند. وقتی که نورون خاموش است هیچ سیگنالی به اطراف نمی فرستد ولی وقتی که روشن است یک سیگنال به نورون هایی که به آن متصل هستند ارسال می کند. نرونی که در

Inactive<sup>2</sup>  
Active<sup>3</sup>

حالت خاموش است تنها وقتی روشن می شود که پتانسیل ناشی از سیگنال هایی که از نورون های دیگر دریافت کرده از یک حد آستانه بیشتر شود. این حد آستانه را برای نورون  $i$  با  $\nu_i$  نشان می دهیم. پتانسیلی که روی این نورون جمع شده را نیز با  $h_i$  نشان می دهیم. بنابراین خواهیم داشت:

$$\begin{aligned} n_i &\longrightarrow 1 && \text{if} && h_i \geq \nu_i \\ n_i &\longrightarrow 0 && \text{if} && h_i < \nu_i. \end{aligned} \quad (1)$$

حال سوال این است که  $h_i$  چگونه به سیگنال های رسیده از نورون های دیگر مرتبط است. در ساده ترین حالتی که در نظر می گیریم،  $h_i$  را به صورت زیر می نویسیم:

$$h_i := \sum_j \omega_{ij} n_j, \quad (2)$$

$\omega_{ji}$  را شدت اتصال سیناپسی بین نورون  $i$  و  $j$  می نامیم. دقت کنید که اگر چه  $\omega_{ji}$  را شدت اتصال سیناپسی می نامیم ولی این کمیت یک عدد حقیقی است که می تواند منفی نیز باشد چرا که ممکن است روشن شدن نورون  $i$  تاثیر منفی روی روشن شدن نورون  $j$  بگذارد. نتیجه کلی این است که اگر حالت نورون های شبکه را در زمان  $t$  با  $\{n_i(t)\}$  نشان دهیم، وضعیت آنها در زمان  $t + 1$  با قاعده زیر تعیین می شود:

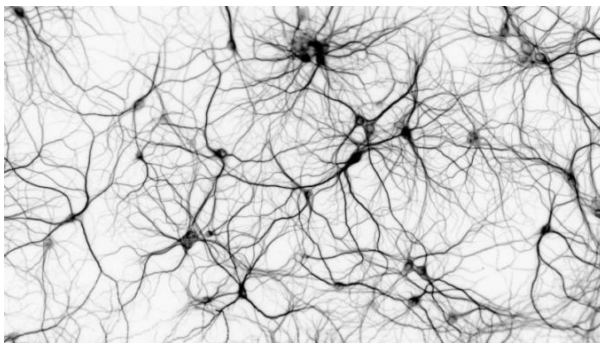
$$\begin{aligned} n_i(t+1) &= 1 && \text{if} && \sum_j \omega_{ij} n_j(t) \geq \nu_i \\ n_i(t+1) &= 0 && \text{if} && \sum_j \omega_{ij} n_j(t) \leq \nu_i. \end{aligned} \quad (3)$$

همانطور که از معادلات بالا معلوم است، دینامیک نورون ها یک دینامیک موازی است به این معنا که در هر بازه زمانی واحد (حدود یک صدم ثانیه برای مغز) وضعیت تمام نورون ها همراه با هم تغییر می کند. بنابراین اگر برای مغز یک سیکل زمانی<sup>۴</sup> در نظر بگیریم، این سیکل در مقایسه با سیکل زمانی کامپیوتر که حدود  $10^{-9}$  ثانیه است بسیار طولانی است. اما نکته این است که این تقیصه توسط دینامیک همزمان و موازی بالا جبران می شود. .... برای آنکه از روش های مکانیک آماری و آنچه که در باره سیستم های اسپینی می دانیم بتوانیم استفاده کنیم، بهتر است که بجای متغیر  $n_i$  از یک متغیر جدید  $s_i$  برای نشان دادن وضعیت یک نورون استفاده کنیم. بنابراین قرار می دهیم

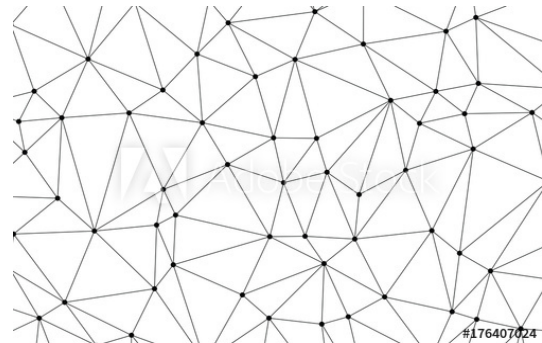
$$n_i = \frac{1 + s_i}{2} \quad \text{or} \quad s_i = 2n_i - 1. \quad (4)$$

دینامیک شبکه عصبی بر حسب متغیرهای جدید به شکل زیر خواهد بود:

Clock Time<sup>f</sup>



**a**



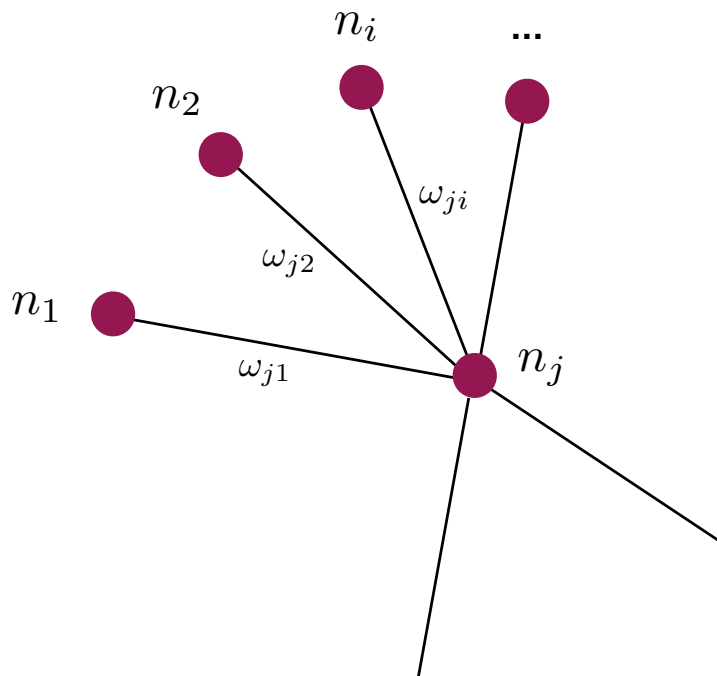
**b**

شکل ۵: .. سمت چپ، شمایی از یک شبکه عصبی که البته تصویری واقعی نیست. نقاط تاریک نورون ها را نشان می دهد که به وسیله سیناپس ها به دیگر نورون ها متصل شده اند. سمت راست مدلی از یک شبکه عصبی است.

$$\begin{aligned}
 s_i(t+1) &= 1 & \text{if } \sum_j \omega_{ij} \left( \frac{1+s_j}{2} \right) &\geq \nu_i \\
 s_i(t+1) &= -1 & \text{if } \sum_j \omega_{ij} \left( \frac{1+s_j}{2} \right) &< \nu_i.
 \end{aligned} \tag{5}$$

می توان با تعریف

$$\theta_i := \nu_i - \frac{1}{2} \sum_j \omega_{ij} \tag{6}$$



شکل ۶: وضعیت یک نورون با وضعیت تمام نورون ها و شدت سیناپس های مرتبط با آن مشخص می شود.

و هم چنین باز تعریف  $\omega_{ij} \rightarrow \frac{1}{2}\omega_{ij}$  دینامیک شبکه عصبی را به شکل زیر بازنویسی کرد:

$$\begin{aligned}
 s_i(t+1) &= 1 & \text{if } \sum_j \omega_{ij}s_j(t) &\geq \theta_i \\
 s_i(t+1) &= -1 & \text{if } \sum_j \omega_{ij}s_j(t) &< \theta_i.
 \end{aligned} \tag{۷}$$

همه این ها به صورت فشرده زیر قابل نوشتن است:

$$s_i(t+1) = \text{sign}(h_i(t) - \theta_i) = \text{sign}\left(\sum_j \omega_{ij}s_j(t) - \theta_i\right). \tag{۸}$$



در روابط بالا  $\text{sign}(x)$  تابع علامت است یعنی

$$\text{sign}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0. \end{cases} \quad (9)$$

حال که شبکه عصبی تعریف شده است می توانیم از خود بپرسیم که چگونه می توان از آن استفاده کرد. در بخش های بعدی این درس به چندتا از کاربردهای شبکه عصبی می پردازیم. نخستین آنها حافظه است.

## ۲ حافظه های وابسته

در یک رایانه طرح ها در آدرس های معین ذخیره می شوند و تنها با دادن همان آدرس ها نیز قابل بازیابی هستند. کامپیوتر هر طرحی را با دانستن آدرس دقیق آن به یاد می آورد. حتی وقتی که کل کامپیوتر را برای یافتن یک طرح خاص یا یک کلمه خاص جستجو می کنیم، این جستجو در آدرس ها صورت می گیرد. اما در حافظه ما طرح ها بر اساس محتوی به یاد سپرده می شوند و بر اساس محتوی نیز به یاد آورده می شوند. لافل ما هیچ گونه آگاهی ای نداریم که یک نام خاص، یک منظره خاص و یا یک رابطه ریاضی خاص یا دانش خاصی در کجای مغز ما ذخیره شده است. هنگام یادآوری نیز این داده ها را بر اساس آدرس آنها یا بر اساس تاریخ به یادسپاری آنها به یاد نمی آوریم، بلکه براساس وابستگی آنها به چیزهای دیگر به یاد می آوریم. به عنوان مثال نام یک شخص را با دقت به اینکه او را در کجا دیده ایم یا اینکه وی دوست مشترک ما با چه کسی بوده است به یاد می آوریم. به همین دلیل این نوع حافظه را حافظه بر اساس وابستگی<sup>۵</sup> می خوانیم. در این درس می خواهیم ببینیم چگونه یک شبکه عصبی می تواند مدلی برای یک حافظه وابسته باشد. نخست دقت می کنیم که یک شبکه عصبی با  $N$  تا نورون می تواند در  $2^N$  حالت<sup>۶</sup> یا هیئت<sup>۷</sup> قرار بگیرد. تخمین زده می شود که برای مغز انسان تعداد نورون ها از مرتبه  $10^{11}$  است. بنابراین مغز علی الاصول می تواند در  $2^{10^{11}}$  حالت مختلف قرار بگیرد. این عدد، عدد بسیار بزرگی است. برای آنکه مرتبه بزرگی این عدد را بفهمیم کافی است توجه کنیم که تعداد کل اتم های کیهان از مرتبه  $10^{80}$  یا  $2^{250}$  است. می توان تصور کرد که مغز انسان در یک واحد زمانی که از مرتبه  $10^{-2}$  ثانیه است، حالتش عوض می شود و وقتی که در یک حالت معین قرار دارد مثل این است که یک خط از برنامه کنترل بدن یا ذهن و حواس انسان در حال اجراست، یعنی در آن حالت خاص تکلیف تمام اعمال فیزیولوژیک، اندام های ارادی و غیرارادی، داده های حسی ای که می بایست پردازش شوند و محتوی ذهن

Associative Memory<sup>۵</sup>  
state<sup>۶</sup>  
configuration<sup>۷</sup>

و خاطره معلوم است. در یک واحد زمانی بعد حالت عوض می شود و به همراه آن همه دستورات، داده ها و طرح ها و محتوی ذهن نیز عوض می شود. اگر چنین باشد می توان تصور کرد که بخش کوچکی از شبکه عصبی که مربوط به  $N < N$  تا نورون است و تنها به حافظه اختصاص دارد و نه به اعمال کنترل مغز، مسئول نگاهداری طرح ها و به یاد آوردن آنهاست. تعداد کل طرح هایی که در این قسمت ذخیره می شود برابر است با  $2^N$ . چنانکه دیدیک یک بار که سیناپس های  $w_{ij}$  تعیین می شوند، دینامیک شبکه عصبی به طور کامل و دقیق تعیین می شود، در نتیجه اگر از یک طرح اولیه شروع کنیم به طور یکتا حالت شبکه عصبی در تمام زمان های آینده تعیین خواهد شد. حال می توان سوال کرد پس داده های حسی جدید اعم از بینایی، شنوایی، و نظایر آن و هم چنین آموخته های جدید به کجا می روند و چه تاثیری از خود باقی می گذارند. پاسخ آن است که داده های جدید اتصالات سیناپسی را تغییر می دهند. این خاصیت یعنی قابلیت تغییر اتصالات سیناپسی در طول زمان پلاستیسیته سیناپسی<sup>۸</sup> خوانده می شود. هرگاه دقت کنیم که تعداد اتصالات سیناپسی از مرتبه  $10^{15}$  است، و هر اتصال سیناپسی نیز با یک عدد حقیقی  $w_{ij}$  معین می شود، متوجه تعداد بسیار زیاد انواع اتصالات سیناپسی می شویم. حتی اگر یک اتصال سیناپسی تنها یک عدد صفر و یک باشد متوجه می شویم که انواع پیکربندی های سیناپسی از مرتبه  $2^{10^{15}}$  است که عدد بسیار بزرگی است. این عدد بسیار بزرگ تر از تعداد اطلاعاتی است که می تواند به صورت کد ژنتیک روی دی-ان-آ نوشته شده باشد. بنابراین قسمت اعظم پیکربندی سیناپسی می بایست بعد از تولد و در طی فرایند یادگیری و دریافت داده های حسی تنظیم شده باشد. یک آزمایش ساده می تواند این موضوع را تایید کند. در یک آزمایش بلافاصله پس از تولد چشم های بچه گربه های نوزاد بسته شده و به مدت چندین هفته این بچه گربه ها از دیدن دنیای اطراف محروم بوده اند. بعد از بازکردن چشم بندها، معلوم شده که بچه گربه ها به کلی حس بینایی خود را از دست داده اند اگر چه چشم ها، شبکه و تمام اندام های مربوط به بینایی آنها سالم بوده است. نتیجه کلی آن است که مغز یا شبکه عصبی با ذخیره کردن طرح های جدید، اتصالات سیناپسی اش تغییر می کند و به اصطلاح این طرح را یاد می گیرد تا بعداً بتواند آنها را به یاد آورد. برای سادگی فرض کنید که یک شبکه عصبی یک طرح مثلاً شکل یک فنجان را یاد گرفته باشد. این طرح را با شماره (۱) نشان می دهیم. در این طرح نورون ها (که گاهی اوقات آنها را اسپین نیز می نامیم) هیئت مشخصی به خود می گیرند. در این هیئت مشخص که شکل و رنگ و تمام اطلاعات مربوط به فنجان در آن کد شده است، وضعیت نورون  $i$  ام، یا اسپین  $i$  ام را با  $\sigma_i^{(1)}$  نشان می دهیم. حال سوال این است که پس از یادگیری این طرح سیناپس ها به چه شکلی در می آیند؟ در ساده ترین مدل که به مدل هاپفیلد<sup>۹</sup> معروف است، فرض می شود که سیناپس ها شکل زیر را اختیار می کنند:

$$w_{ij} = \frac{1}{N} \sigma_i^{(1)} \sigma_j^{(1)} \quad (10)$$

---

Synaptic Plasticity<sup>۸</sup>  
Hopfield<sup>۹</sup>

بنابراین از این پس دینامیک شبکه عصبی از معادلات زیر تعیین می شود:

$$s_i(t+1) = \text{sign}\left(\sum_{j=1}^N \omega_{ij} s_j(t)\right) = \text{sign}\left(\sum_{j=1}^N \frac{1}{N} \sigma_i^{(1)} \sigma_j^{(1)} s_j(t)\right) \quad (11)$$

حال از خود می پرسیم که معنای به یاد آوردن چیست؟ از نظر شهودی می توانیم معنای زیر را برای آن قایل شویم. اگر در لحظه  $t$  شبکه در وضعیتی قرار گیرد که شبیه به طرح  $\sigma^{(1)}$  باشد، در زمان  $t+1$  در وضعیتی قرار خواهد گرفت که شباهت بیشتری با طرح  $\sigma^{(1)}$  خواهد داشت. پس از طی چند مرحله کاملاً در طرح  $\sigma^{(1)}$  قرار خواهد گرفت و این طرح را به طور کامل به یاد خواهد آورد. این همان معنایی است که از به یاد آوردن و شناسایی طرح ها از روی یک نمونه شبیه به آنها در نظر داریم. به خصوص انتظار داریم که اگر در زمان  $t$  طرح  $\sigma^{(1)}$  بدون هیچ نقص و خطایی به شبکه عصبی داده شود، در زمان  $t+1$  بازهم شبکه در همان طرح  $\sigma^{(1)}$  باقی خواهد ماند و تغییر نخواهد کرد. این انتظارات را با مدل هایفیلد می سنجم:

$$\begin{aligned} s_i(t) &= \sigma_i^{(1)} \quad \rightarrow \quad s_i(t+1) = \text{sign}\left(\sum_{j=1}^N \frac{1}{N} \sigma_i^{(1)} \sigma_j^{(1)} \sigma_j^{(1)}\right) \\ &= \text{sign}\left(\sigma_i^{(1)} \frac{1}{N} \sum_{j=1}^N 1\right) = \text{sign}(\sigma_i^{(1)}) = \sigma_i^{(1)}. \end{aligned} \quad (12)$$

بنابراین اگر طرح  $\sigma^{(1)}$  به این شبکه نشان داده شود، دقیقاً آن را به یاد می آورد. حال فرض کنید طرحی که به شبکه نشان داده می شود در  $n$  نقطه با طرح  $\sigma^{(1)}$  برابر و در  $N-n$  نقطه با آن مخالف است. بدون نقض کلیت می توانیم فرض کنیم که:

$$s_i(t) = \begin{cases} \sigma_i^{(1)} & i = 1, 2, \dots, n, \\ -\sigma_i^{(1)} & i = n+1, n+2, \dots, N. \end{cases} \quad (13)$$

در این صورت خواهیم داشت:

$$\begin{aligned} s_i(t+1) &= \text{sign}\left(\sum_{j=1}^N \frac{1}{N} \sigma_i^{(1)} \sigma_j^{(1)} s_j(t)\right) \\ &= \text{sign}\left(\sum_{j=1}^n \frac{1}{N} \sigma_i^{(1)} \sigma_j^{(1)} s_j(t) + \sum_{j=n+1}^N \frac{1}{N} \sigma_i^{(1)} \sigma_j^{(1)} s_j(t)\right) \\ &+ \text{sign}\left(\frac{n}{N} \sigma_i^{(1)} - \sigma_i^{(1)} \frac{N-n}{N}\right) = \text{sign}\left(\frac{2n-N}{N} \sigma_i^{(1)}\right). \end{aligned} \quad (14)$$

بنابراین خواهیم داشت:

$$s_i(t+1) = \begin{cases} \sigma_i^{(1)} & n > \frac{N}{2} \\ -\sigma_i^{(1)} & n < \frac{N}{2} \end{cases} \quad (15)$$

این نتیجه با شهود ما نیز سازگار است، به این معنا که اگر طرحی که در ابتدا نشان داده می شود تفاوت خیلی زیادی با طرح اصلی نداشته باشد ما می توانیم آن طرح را شناسایی کنیم.

حال فرض کنید که تعداد  $p$  طرح

$$\sigma^{(1)}, \quad \sigma^{(2)}, \quad \dots, \sigma^{(p)} \quad (16)$$

در شبکه ذخیره شده باشند. در این صورت سیناپس ها چه خواهند بود؟ بنابر مدل هاپفیلد سیناپس ها عبارتند از:

$$\omega_{ij} = \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)}. \quad (17)$$

اگر طرح  $\sigma^{(\nu)}$  را به این شبکه نشان دهیم چه خواهد شد؟ در این صورت داریم:

$$s_i(t+1) = \text{sign} \left[ \sum_{j=1}^N \left( \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_j^{(\nu)} \right] \quad (18)$$

حال جمع داخل پرانتز را به دو قسمت  $\mu = \nu$  و  $\mu \neq \nu$  جدا می کنیم و می نویسیم:

$$\begin{aligned} s_i(t+1) &= \text{sign} \left[ \sum_{j=1}^N \left( \frac{1}{N} \sigma_i^{(\nu)} \sigma_j^{(\nu)} + \frac{1}{N} \sum_{\mu \neq \nu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_j^{(\nu)} \right] \\ &= \text{sign} \left[ \sum_{j=1}^N \frac{1}{N} \sigma_i^{(\nu)} + \frac{1}{N} \sum_{j=1}^N \sum_{\mu \neq \nu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \sigma_j^{(\nu)} \right] \\ &= \text{sign} \left[ \sigma_i^{(\nu)} + \frac{1}{N} \sum_{j=1}^N \sum_{\mu \neq \nu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \sigma_j^{(\nu)} \right] \end{aligned} \quad (19)$$

اگر طرح های ضبط شده شباهت کمی با هم داشته باشند می توان فرض کرد که عبارت  $\sigma_i^{(\mu)} \sigma_j^{(\mu)} \sigma_j^{(\nu)}$  یک متغیر تصادفی است که مقادیر

$-1$  و  $1$  را اختیار می کند و در نتیجه مجموع جملات درون  $\sum_j \sum_{\mu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \sigma_j^{(\nu)}$  برای این جملات از مرتبه  $Np$  است. به این ترتیب خواهیم داشت:

$$s_i(t+1) = \text{sign} \left( \sigma_i^{(\nu)} + O\left(\sqrt{\frac{p}{N}}\right) \right). \quad (20)$$

اگر  $p \ll N$  باشد، آنگاه

$$s_i(t+1) = \sigma_i^{(\nu)} \quad (21)$$

یعنی اینکه شبکه طرح  $\sigma^{(\nu)}$  را شناسایی می کند.

حال فرض کنید که طرح فوق را به طور ناقص به شبکه نشان دهیم یعنی

$$s_i(t) = \begin{cases} \sigma_i^{(\nu)} & i = 1, 2, \dots, n, \\ -\sigma_i^{(\nu)} & i = n+1, n+2, \dots, N. \end{cases} \quad (22)$$

در این صورت داریم:

$$s_i(t+1) = \text{sign} \left[ \sum_{j=1}^N \left( \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) s_j^{(t)} \right] \quad (23)$$

برای محاسبه این عبارت نخست جمع روی نورون ها یعنی  $\sum_{j=1}^N$  را به دو قسمت  $\sum_{j=1}^n$  و  $\sum_{j=n+1}^N$  می شکنیم و بدست می آوریم:

$$s_i(t+1) = \text{sign} \left[ \sum_{j=1}^n \left( \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_j^{(\nu)} - \sum_{j=n+1}^N \left( \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_j^{(\nu)} \right]. \quad (24)$$

حال در هرکدام از این جمع ها، جمع روی طرح ها را به دو قسمت  $\mu = \nu$  و  $\mu \neq \nu$  می شکنیم. نتیجه اش این خواهد بود:

$$s_i(t+1) = \text{sign} \left[ \sum_{j=1}^n \left( \frac{1}{N} \sigma_i^{(\nu)} \sigma_j^{(\nu)} \right) \sigma_j^{(\nu)} + \sum_{j=1}^n \left( \frac{1}{N} \sum_{\mu \neq \nu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_j^{(\nu)} \right] \\ - \sum_{j=n+1}^N \left( \frac{1}{N} \sigma_i^{(\nu)} \sigma_j^{(\nu)} \right) \sigma_j^{(\nu)} - \sum_{j=n+1}^N \left( \frac{1}{N} \sum_{\mu \neq \nu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_j^{(\nu)} \quad (25)$$

جملات اول و سوم را در یکجا و جملات دوم و چهارم را نیز در یک جا جمع می کنیم:

$$s_i(t+1) = \text{sign} \left[ \frac{2n-N}{N} \sigma_i^{(\nu)} + \dots \right] \quad (26)$$

که در آن .... نشان دهنده جملات دوم و چهارم است. بازم بنا بر استدلال ولگشت می دانیم که جملات دوم و چهارم از مرتبه  $\sqrt{\frac{p}{N}}$  هستند.

بنابراین

$$s_i(t+1) = \text{sign} \left[ \frac{2n-N}{N} \sigma_i^{(\nu)} + O\left(\sqrt{\frac{p}{N}}\right) \right] \quad (27)$$

در نتیجه اگر  $p \ll N$  باشد و  $n > \frac{N}{2} + O(\sqrt{pN})$  آنگاه

$$s_i(t+1) = \sigma_i^{(\nu)}. \quad (28)$$

به این ترتیب می بینیم که وقتی تعداد طرح ها زیاد می شود برای شناسایی دقیق یک طرح می بایست شباهت آن طرح با طرح اولیه بیشتر از موقعی باشد که تنها یک طرح در حافظه ذخیره شده است.

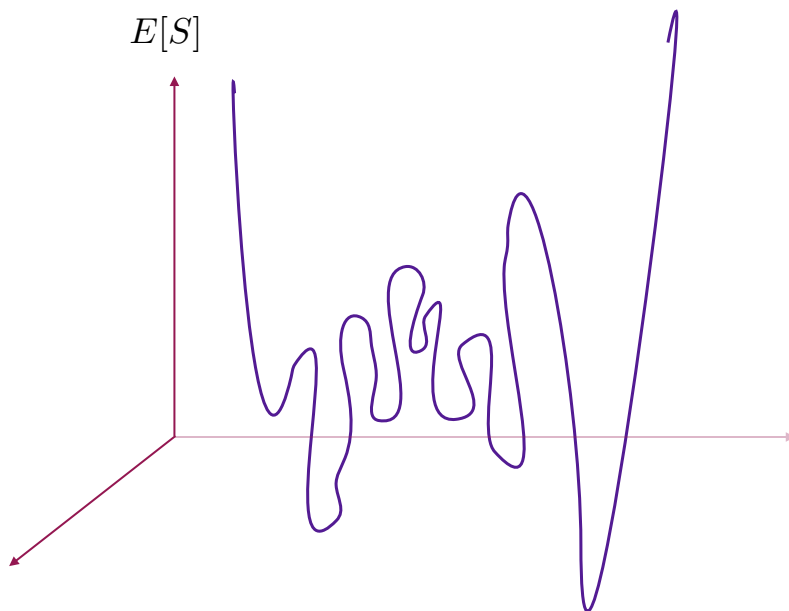
ایا برای دینامیک یک شبکه عصبی می توان کمیتی شبیه به انرژی چنان تعریف کرد که تحت دینامیک شبکه همواره کاهش یابد. اگر چنین چیزی را بتوان تعریف کرد معنایش این خواهد بود که طرح های ضبط شده منطبق بر کمینه های موضعی این تابع انرژی هستند، شکل (۷). این تصویر با آنچه که ما در باره فرایند یادآوری طرح های ضبط شده از روی طرح های ناقص دیدیم نیز مطابقت دارد، به این معنا که اگر در نزدیکی یک کمینه موضعی مثل  $\sigma^{(\nu)}$  باشیم، در اثر دینامیک شبکه به همان نقطه کمینه می رسیم، اما اگر اختلاف با طرح مورد نظر یعنی نقطه کمینه مورد نظر از یک حد معین بیشتر باشد، ممکن است به یک کمینه مجاور یعنی به یک طرح اشتباه برویم. بنابراین ممکن است در تابع انرژی شبکه تعداد زیادی نقطه کمینه وجود داشته باشد و هر نقطه کمینه نیز خود دارای تعداد زیادی نقطه های کمینه دیگر باشد و الی اخر. به این خاصیت *Ultrametricity* می گویند. شکل (۷) نشان می دهد که این خاصیت به چه معناست.

تابع انرژی ای که تحت دینامیک شبکه عصبی کاهش می یابد تابع زیر است:

$$\mathcal{E}[s] = - \sum_{i,j} \omega_{ij} s_i s_j. \quad (29)$$

بنابراین پتانسیل شبکه عصبی کاملاً شبیه به انرژی مدل آیزینگ با ضرایب جفتدگی تصادفی است که توسط سیناپس های  $\omega_{ij}$  تعیین می شود. حال خواص این تابع انرژی را مطالعه می کنیم. نخست مقدار آن را برای یک طرح ضبط شده مثل  $\sigma^{(\nu)}$  محاسبه می کنیم:

$$\begin{aligned} \mathcal{E}[\sigma^{(\nu)}] &= - \sum_{i,j} \omega_{ij} \sigma_i^{(\nu)} \sigma_j^{(\nu)} \\ &= - \sum_{i,j} \left( \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_i^{(\nu)} \sigma_j^{(\nu)} \\ &= - \sum_{i,j} \left( \frac{1}{N} \sigma_i^{(\nu)} \sigma_j^{(\nu)} + \frac{1}{N} \sum_{\mu \neq \nu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) \sigma_i^{(\nu)} \sigma_j^{(\nu)} \\ &= - \frac{1}{N} N^2 - \frac{1}{N} \sum_{\mu \neq \nu} \sum_{i,j} \sigma_i^{(\mu)} \sigma_j^{(\mu)} \sigma_i^{(\nu)} \sigma_j^{(\nu)} \\ &= -N - \frac{1}{N} \sum_{\mu \neq \nu} \left( \sum_i \sigma_i^{(\mu)} \sigma_i^{(\nu)} \right) \left( \sum_j \sigma_j^{(\mu)} \sigma_j^{(\nu)} \right) \end{aligned}$$



شکل ۷: خاصیت اولترامتریسیته. درون هر نقطه کمینه نقاط کمینه بیشتری وجود دارد.

$$= -N - N \sum_{\mu \neq \nu} \langle \sigma^{(\mu)}, \sigma^{(\nu)} \rangle^2, \quad (30)$$

که در آن از تعریف ضرب داخلی

$$\langle \sigma^{(\mu)}, \sigma^{(\nu)} \rangle = \frac{1}{N} \sum_{i=1}^N \sigma_i^{(\mu)} \sigma_i^{(\nu)} \quad (31)$$

استفاده کرده ایم. دقت کنید که

$$-1 \leq \langle \sigma^{(\mu)}, \sigma^{(\nu)} \rangle \leq 1, \quad (32)$$

بنابراین تا کنون نشان داده ایم که

$$\mathcal{E}[\sigma^{(\nu)}] = -N \left( 1 + \sum_{\mu \neq \nu} \langle \sigma^{(\mu)}, \sigma^{(\nu)} \rangle^2 \right). \quad (33)$$

این نتیجه به این معناست که اگر همپوشانی طرح ها کم باشد به این معنا که اگر  $|\langle \sigma^{(\mu)}, \sigma^{(\nu)} \rangle| \ll 1$  آنگاه  $\mathcal{E}[\sigma^{(\nu)}] \approx -N$ . برای آنکه بینیم آیا واقعا طرح  $\sigma^{(\nu)}$  یک کمینه انرژی است یا نه، انرژی طرحی را حساب می کنیم که اندکی با طرح  $\sigma^{(\nu)}$  متفاوت است و در یک اسپین مثلا

اسپین شماره یک با آن متفاوت است. به این ترتیب این طرح دارای شکل زیر است:

$$s_i = \begin{cases} -\sigma_i^{(\nu)} & i = 1 \\ \sigma_i^{(\nu)} & i \neq 1. \end{cases} \quad (۳۴)$$

در این صورت خواهیم داشت:

$$\mathcal{E}[s] - \mathcal{E}[\sigma^{(\nu)}] = - \sum_{ij} \omega_{ij} (s_i s_j - \sigma_i^{(\nu)} \sigma_j^{(\nu)}). \quad (۳۵)$$

عبارت داخل پرانتز را با  $A_{ij}$  نشان می دهیم. براحتی معلوم می شود که

$$A_{ij} = \begin{cases} 0 & \text{if } (i > 1, j > 1) \text{ or } (i = 1, j = 1) \\ -2\sigma_i^{(\nu)} \sigma_j^{(\nu)} & \text{if } (i = 1, j > 1) \text{ or } (i > 1, j = 1) \end{cases} \quad (۳۶)$$

بنابراین

$$\mathcal{E}[s] - \mathcal{E}[\sigma^{(\nu)}] = 2 \sum_{j=1}^N \omega_{1j} \sigma_1^{(\nu)} \sigma_j^{(\nu)} + 2 \sum_{i=1}^N \omega_{i1} \sigma_i^{(\nu)} \sigma_1^{(\nu)} \quad (۳۷)$$

و چون  $\omega_{ij} = \omega_{ji}$

$$\mathcal{E}[s] - \mathcal{E}[\sigma^{(\nu)}] = 4 \sum_{j=1}^N \omega_{1j} \sigma_1^{(\nu)} \sigma_j^{(\nu)} \quad (۳۸)$$

با جایگزای بجای  $\omega_{1j}$  بدست می آوریم:

$$\mathcal{E}[s] - \mathcal{E}[\sigma^{(\nu)}] = 4 \frac{1}{N} \sum_{j=1}^N \sum_{\mu=1}^p \sigma_1^{(\mu)} \sigma_j^{(\mu)} \sigma_1^{(\nu)} \sigma_j^{(\nu)} \quad (۳۹)$$

جمع روی  $\mu$  را به دو قسمت  $\mu = \nu$  و  $\mu \neq \nu$  تجزیه می کنیم:

$$\begin{aligned} \mathcal{E}[s] - \mathcal{E}[\sigma^{(\nu)}] &= 4 \frac{1}{N} \sum_{j=1}^N (1 + \sum_{\mu \neq \nu} \sigma_1^{(\mu)} \sigma_j^{(\mu)} \sigma_1^{(\nu)} \sigma_j^{(\nu)}) \\ &= 4 (1 + \frac{1}{N} \sum_{\mu \neq \nu} \sigma_1^{(\mu)} \sigma_1^{(\nu)} \langle \sigma^{(\mu)} \sigma^{(\nu)} \rangle) \end{aligned} \quad (۴۰)$$

حال اگر طرح ها همپوشان نباشند، داریم  $\langle \sigma^{(\mu)}, \sigma^{(\nu)} \rangle \approx 0$  و از آن نتیجه می گیریم

$$\mathcal{E}[s] - \mathcal{E}[\sigma^{(\nu)}] \approx 4.$$



این نتیجه به این معناست که هر گاه یک اسپین را در یک طرح ضبط شده عوض کنیم، انرژی به اندازه چهار واحد اضافه می شود. بنابراین یک طرح ضبط شده در کمینه انرژی است.

تا کنون دیده ایم که:

الف) طرح های ضبط شده در کمینه انرژی خود هستند،

ب) هرگاه طرحی به شبکه می دهیم که با یک طرح ضبط شده تفاوت کمی دارد، دینامیک طبیعی این شبکه آن طرح را به سمت طرح ضبط شده که انرژی کمینه دارد می برد.

این ها شواهد غیرمستقیمی هستند برای اینکه دینامیک شبکه عصبی واقعا شبکه را به سمت انرژی های کمتر سوق می دهد. آیا این موضوع را به طور مستقیم می توان دید؟ برای فهم این موضوع روش تغییر دادن<sup>۱۰</sup> نورون ها را به کار می بریم که ضمن اینکه با دینامیک واقعی شبکه یکسان است، برای منظوری که داریم ساده و سرراست است. در این روش یک نورون را به طور تصادفی انتخاب می کنیم و حالت آن را مطابق با قاعده

$$s_i(t+1) = \text{sign}\left(\sum_{j=1}^N \omega_{ij}s_j(t)\right) \quad (41)$$

عوض می کنیم. (می توانیم همه نورون ها را به طور منظم و یک به یک مطابق با قاعده بالا تغییر داد.) البته برای این که این روش با دینامیک واقعی شبکه که طی آن همه نورون ها در یک لحظه تغییر می کنند، یکسان باشد، می بایست در هر مرحله که یک نورون را تغییر می دهیم، برای تغییر دادن بقیه نورون ها مطابق با رابطه فوق از حالت قبلی آن نورون ها و نه حالت جدیدشان استفاده کنیم. حال بیایید تغییر انرژی شبکه را برای یک حرکت دلخواه از این نوع حساب کنیم. می دانیم که

$$\mathcal{E}[s] = -\sum_{i,j} \omega_{ij}s_i s_j = -\sum_i s_i \left(\sum_j \omega_{ij}s_j\right) \quad (42)$$

و یا

$$\mathcal{E}[s] = -\sum_i s_i h_i \quad (43)$$

---

<sup>۱۰</sup>Updating

که در آن  $h_i = \sum_j \omega_{ij} s_j$  مثل یک میدان مغناطیسی موضعی در نقطه  $i$  است که از اثر بقیه اسپین ها درست شده است. داریم

$$\mathcal{E}_t[s] = - \sum_i s_i(t) h_i(t). \quad (44)$$

اگر فقط اسپین  $i$  ام را تغییر دهیم، خواهیم داشت:

$$\mathcal{E}_{t+1}[s] = - \sum_i s_i(t+1) h_i(t) = - \sum_i \text{sign}(h_i(t)) h_i(t) = - \sum_i |h_i(t)| = - \sum_i |s_i(t) h_i(t)|. \quad (45)$$

اما می دانیم که  $x \leq |x|$  بنا بر این

$$\mathcal{E}_{t+1}[s] \leq - \sum_i s_i(t) h_i(t) = \mathcal{E}_t[s]. \quad (46)$$

در نتیجه با هر حرکت اسپین ها انرژی شبکه اسپینی کاهش می یابد.

### ۳ شبکه های عصبی کاتوره ای

تا کنون فرض کردیم که دینامیک یک شبکه عصبی کاملا متعین<sup>۱۱</sup> است. این دینامیک توسط معادله زیر بیان می شد:

$$s_i(t+1) = \text{sign}(h_i(t)) \quad s_i \pm 1 \quad (47)$$

که در آن

$$h_i(t) = \sum_{j=1}^N \omega_{ij} s_j(t) \quad (48)$$

در این بخش می خواهیم این دینامیک متعین را با یک دینامیک تصادفی یا استوکاستیک عوض کنیم. در دینامیک استوکاستیکی که معرفی می کنیم یک پارامتر وجود خواهد داشت مثل  $T$  که هرگاه آن را به سمت صفر میل می دهیم، همان دینامیک متعین احیا خواهد شد. بنابراین با این تعمیم چیزی را از دست نمی دهیم زیرا در پایان همیشه می توانیم  $T$  را مساوی صفر قرار دهیم. این کار یک فایده دارد و آن این است که می توانیم از روش هایی که در مکانیک آماری برای مطالعه این گونه سیستم ها یاد گرفته ایم استفاده کنیم. انگیزه مهم تری که برای این تعمیم وجود دارد این است که یک شبکه عصبی طبیعی یا مصنوعی واقعا ممکن است تحت تاثیر نوفه قرار بگیرد و شاید قانون مندی خیلی دقیقی برای دینامیک آن

<sup>۱۱</sup>Deterministic

وجود نداشته باشد و به همین جهت رفتار تصادفی از خود نشان دهد. به عنوان مثال دیده شده است که بعضی از نورون ها بدون هیچ دلیلی روشن می شوند یا به اصطلاح آتش می کنند یا بالعکس در حالیکه سیگنال لازم برای روشن شدن آنها از حد آستانه بیشتر شده است هم چنان خاموش می مانند و بعد به طور تصادفی روشن می شوند. می توانیم پارامتر  $T$  را که درجه تصادفی بودن یک شبکه را معین می کند، دما بخوانیم، اما باید توجه کنیم که این دما، دمای محیط نیست بلکه تنها به خاطر نبودن یک اسم بهتر و با تشابه با ساختار مکانیک آماری از این اسم استفاده شده است.

بعد از این مقدمه به تعریف دینامیک تصادفی می رسم. متغیرهای  $s_i$  همچنان مقادیر 1 و -1 را اختیار می کنند. سیگنالی که باعث آتش کردن نورون  $i$  ام می شود همچنان برابر است با  $h_i = \sum_j \omega_{ij} s_j$ . اما این سیگنال فقط احتمال آتش کردن نورون  $i$  ام را تعیین می کند. یعنی

$$\mathcal{P}[s_i(t+1) = 1] = f(h_i(t)) \quad \mathcal{P}[s_i(t+1) = -1] = 1 - f(h_i(t)). \quad (49)$$

برای تابع  $f$  شرط های مناسبی در نظر می گیریم که از نظر فیزیکی معقول باشند. نخست آنکه این تابع می بایست صعودی باشد، دوم آنکه در حد  $h_i \rightarrow \infty$  می بایست به سمت 1 و در حد  $h_i \rightarrow -\infty$  به سمت صفر میل کند. هم چنین انتظار داریم که این تابع به پارامتر  $T$  وابسته باشد و در حد  $T \rightarrow 0$  به سمت تابع پله یعنی  $\Theta(h_i)$  میل کند. بنابراین رفتار کلی چنین تابعی مطابق با شکل (8) است. دقت کنید که تابع پله  $\Theta(x)$  با تابع علامت یعنی  $\text{sign}(x)$  فرق دارد ولی این دو تابع رابطه ساده ای با هم دارند:

$$\Theta(x) = \frac{1 + \text{sign}(x)}{2}. \quad (50)$$

برای چنین تابعی انتخاب های زیادی وجود دارد که از نظر نتایج نهایی فیزیکی هم ارز هستند. ساده ترین انتخاب برای این تابع چنین است:

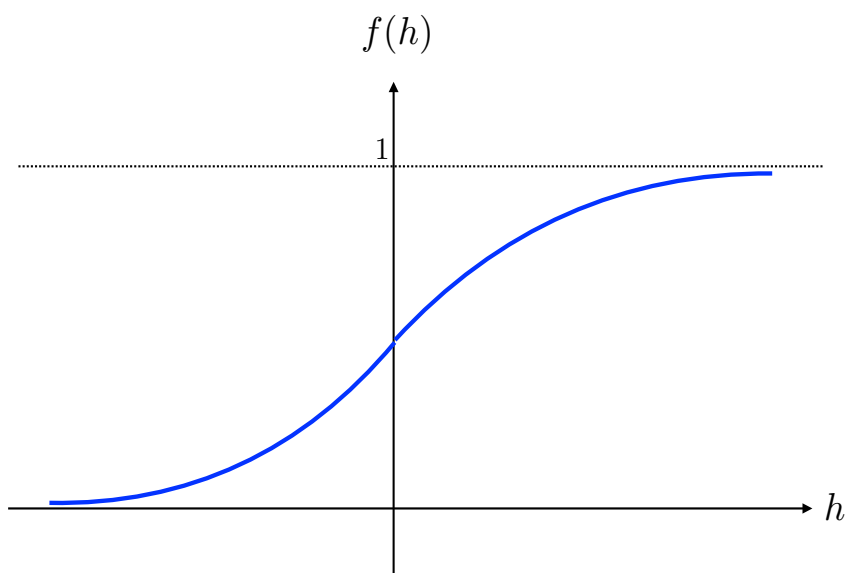
$$f(h) = \frac{1}{1 + e^{-2\beta h}} \quad \beta = \frac{1}{T}. \quad (51)$$

این انتخاب شرط تقارن زیر را نیز برآورده می کند:

$$f(h) + f(-h) = 1. \quad (52)$$

بنابراین احتمالات روشن بودن یا نبودن نورون ها در  $t$  به شکل زیر در می آید:

$$\mathcal{P}[s_i(t+1) = 1] = \frac{1}{1 + e^{-2\beta h_i(t)}} \quad \mathcal{P}[s_i(t+1) = -1] = \frac{1}{1 + e^{2\beta h_i(t)}}. \quad (53)$$



شکل ۸: شکل تابع  $f(h) = \frac{1}{1+e^{-2\beta h}}$ . هرچه که دما پایین تر باشد این تابع به تابع پله یعنی  $\theta(h)$  نزدیک تر می شود و در دمای صفر دقیقاً منطبق بر تابع پله می شود.

اما در اینجا با یک سوال روبرو می شویم و آن اینکه اگر روشن شدن یا خاموش شدن نورون ها یک فرآیند تصادفی است، در این صورت احتمال روشن بودن یا روشن نبودن یک نورون در لحظه  $t$  می بایست به شرایط اولیه بستگی یعنی وضعیت نورون در لحظه صفر بستگی داشته باشد. مثلاً ولگردی را در نظر بگیرید که قدم های خود را به چپ و راست به صورت تصادفی برمی دارد، در این صورت احتمال اینکه در یک زمان معین در کجا باشد مسلماً بستگی به موقعیت او در زمان اولیه دارد. در واقع سوال این است که چه نوع نرخ های گذاری از روشن به خاموش یا خاموش به روشن باعث می شود که احتمال روشن بودن یا نبودن نورون در یک لحظه بستگی به شرایط اولیه نداشته باشد؟ برای پاسخ به این سوال معادله تحول مارکوفی وضعیت نورون ها را می نویسیم: این معادله به شکل زیر است:

$$\begin{aligned} \mathcal{P}[s_i(t+1) = 1] &= (1 - a_t)\mathcal{P}[s_i(t) = 1] + b_t\mathcal{P}[s_i(t) = -1] \\ \mathcal{P}[s_i(t+1) = -1] &= a_t\mathcal{P}[s_i(t) = 1] + (1 - b_t)\mathcal{P}[s_i(t) = -1], \end{aligned} \quad (54)$$

که در آن  $a_t$  احتمال این است که نورونی که در لحظه  $t$  روشن بوده خاموش شود و  $b_t$  احتمال این است که نورونی که همان لحظه خاموش بوده

روشن شود. چنین معادله ای را می توان به صورت ماتریسی زیر نوشت:

$$\begin{pmatrix} \mathcal{P}[s_i(t+1) = 1] \\ \mathcal{P}[s_i(t+1) = -1] \end{pmatrix} = \begin{pmatrix} 1 - a_t & b_t \\ a_t & 1 - b_t \end{pmatrix} \begin{pmatrix} \mathcal{P}[s_i(t) = 1] \\ \mathcal{P}[s_i(t) = -1] \end{pmatrix} \quad (55)$$

در واقع اگر بخواهیم معادله (55) را حل کنیم و توابع احتمال را بدست بیاوریم باید بنویسیم:

$$\begin{pmatrix} \mathcal{P}[s_i(t+1) = 1] \\ \mathcal{P}[s_i(t+1) = -1] \end{pmatrix} = \begin{pmatrix} 1 - a_t & b_t \\ a_t & 1 - b_t \end{pmatrix} \dots \begin{pmatrix} 1 - a_0 & b_0 \\ a_0 & 1 - b_0 \end{pmatrix} \begin{pmatrix} \mathcal{P}[s_i(0) = 1] \\ \mathcal{P}[s_i(0) = -1] \end{pmatrix} \quad (56)$$

که نشان می دهد احتمال وضعیت هر نورون بستگی به شرایط اولیه آن دارد. بنابراین می خواهیم نرخ های  $a_t, b_t$  را به گونه ای تعیین کنیم که حل معادله مارکوفی (55) با معادله (53) سازگار باشد. اگر به معادله (55) نگاه کنیم، راه حل مسئله را براحتی پیدا می کنیم. کافی است تقاضا کنیم که عناصر هر سطر ماتریس با هم مساوی باشند، یعنی کافی است داشته باشیم:

$$1 - a_t = b_t \quad (57)$$

در این صورت از همین معادله خواهیم داشت:

$$\begin{aligned} \mathcal{P}[s_i(t+1) = 1] &= (1 - a_t)\mathcal{P}[s_i(t) = 1] + b_t\mathcal{P}[s_i(t) = -1] = (1 - a_t) \\ \mathcal{P}[s_i(t+1) = -1] &= a_t\mathcal{P}[s_i(t) = 1] + (1 - b_t)\mathcal{P}[s_i(t) = -1] = a_t, \end{aligned} \quad (58)$$

که در طرف راست این معادلات از شرط (57) و شرط بهنجارش احتمالات در هر لحظه استفاده کرده ایم. پس کافی است که قرار دهیم:

$$a_t \equiv \frac{1}{1 + e^{2\beta h_i(t)}}. \quad (59)$$

به این ترتیب می بینیم که نرخ روشن شدن یا خاموش شدن نورون ها با احتمال روشن بودن یا روشن نبودن آنها یکسان می شود.

یک نورون تصادفی مانند یک نورون تعیینی مقادیر 1 و -1 را دارد ولی این مقادیر را با احتمالات معین اختیار می کند. در یک لحظه نمی توان مقدار دقیق یک نورون را تعیین کرد اما می توان متوسط فعالیت آن را در زمان  $t$  مشخص کرد. از معادله (53) داریم:

$$\langle s_i(t+1) \rangle = f(h_i(t)) - (1 - f(h_i(t))) \quad (60)$$

و برای تابع  $f(h) = \frac{1}{1+e^{-2\beta h}}$

$$\langle s_i(t+1) \rangle = f(h_i(t)) - f(-h_i(t)) = \tanh \beta h_i(t). \quad (61)$$

بنابراین مقدار متوسط  $\langle s_i(t+1) \rangle$  به مقدار  $h_i(t)$  بستگی دارد که خود تابعی است از  $s_i$  ها در لحظه  $t$ . اگر روی آنزاملی از شبکه ها متوسط بگیریم، بدست می آوریم:

$$\langle s_i(t+1) \rangle = \langle \tanh \beta h_i(t) \rangle. \quad (62)$$

این معادلات هنوز کمکی به یافتن دینامیک مقادیر متوسط نمی کنند. برای آنکه بتوانیم گامی به جلو برداریم می بایست از تقریب میدان متوسط استفاده کنیم یعنی قرار دهیم

$$\langle \tanh \beta h_i(t) \rangle \approx \tanh \beta \langle h_i(t) \rangle \quad (63)$$

در نتیجه معادله (62) به صورت زیر درمی آید:

$$\langle s_i(t+1) \rangle = \tanh \left( \beta \sum_{j=1}^N \omega_{ij} \langle s_j(t) \rangle \right). \quad (64)$$

حال می توان با دانستن فرم سیناپس های  $\omega_{ij}$  تحول متوسط های  $\langle s_j \rangle$  و بخصوص حالت پایای این تحول را مطالعه کرد. دقت کنید که یک حالت پایا چیزی نیست جز یک طرح ضبط شده که به یادآورده می شود زیرا دینامیک طبیعی این سیستم، یک طرح دلخواه را که تا حدودی شبیه به طرح ضبط شده به سمت آن سوق می دهد. در زیر دو مثال مهم را بررسی می کنیم:

■ مثال اول: در این مثال شکل سیناپس ها به صورت زیر است:

$$\omega_{ij} = \frac{1}{N} \sigma_i \sigma_j. \quad (65)$$

این شکل از سیناپس ها به معنای این است که یک طرح  $\sigma$  توسط شبکه آموخته شده است. در این صورت معادله (64) برای حالت پایا به شکل زیر درمی آید:

$$\langle s_i \rangle = \tanh \beta \left( \frac{1}{N} \sum_{j=1}^N \sigma_i \sigma_j \langle s_j \rangle \right) \quad (66)$$

دقت کنید که چون در جستجوی حالت پایا هستیم، در دو طرف معادله بجای  $\langle s_j(t) \rangle$  و  $\langle s_i(t+1) \rangle$  به ترتیب  $\langle s_j \rangle$  و  $\langle s_i \rangle$  قرار داده ایم. معادلات فوق یک دستگاه معادلات جفت شده غیر خطی است. می توان برای حل آن از حدس و شهود فیزیکی استفاده کرد. در حالت متعین می دانیم که حالت پایا برابر است با  $s_i = \sigma_i$  یعنی طرح به یادآورده شده همان طرح ضبط شده یا آموخته شده است. در حالت تصادفی می توان حدس زد که تصویری که به یاد آورده می شود، همان طرح آموخته شده است ولی کمی تصویر محو شده است. بنابراین قرار می دهیم:

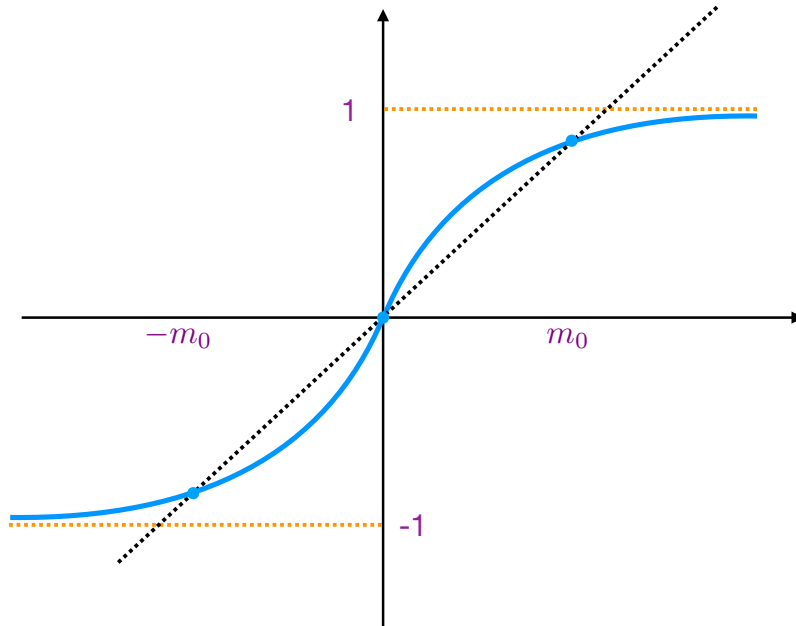
$$\langle s_i \rangle = m\sigma_i, \quad (67)$$

که در آن  $-1 \leq m \leq 1$ . می توان انتظار داشت که در دماهای کم  $m$  بسیار نزدیک به یک و در دماهای زیاد  $m$  نزدیک به صفر است یعنی طرح کاملاً محو شده است و هیچ اطلاعاتی در آن نیست. با جایگذاری (67) در (66) بدست می آوریم:

$$m\sigma_i = \tanh \beta m\sigma_i \quad (68)$$

و یا

$$m = \tanh \beta m. \quad (69)$$



شکل ۹: روش گرافیک برای حل معادله  $m = \tanh \beta m$ .  $m$  بسته به مقدار  $\beta$  این معادله می تواند یک یا سه جواب داشته باشد.

این معادله را می توان با استفاده از رسم منحنی ها حل کرد. نقطه تقاطع منحنی  $f(m) = \tanh \beta m$  با خط  $g(m) = m$  بستگی به مقدار  $\beta$  دارد. نتیجه در شکل (۹) نشان داده شده است.

بنابراین یک دمای بحرانی وجود دارد که بالاتر از آن هیچ طرحی به یادآورده نمی شود. این دمای بحرانی  $T = 1$  است. این پدیده نظیر گذار فاز در سیستم های مغناطیسی است. در پایین تر از این دما معادله (۶۹) سه حل متفاوت دارد،  $m = 0$  و  $m = \pm m_0$ . حل  $m = 0$  یک حل ناپایدار است. دلیل آن را می توان با استفاده از معادله دینامیکی (۶۴) فهمید. بیایید این معادله را با حدس زیر حل کنیم:

$$\langle s_i(t+1) \rangle = m(t+1)\sigma_i \quad \langle s_i(t) \rangle = m(t)\sigma_i. \quad (70)$$

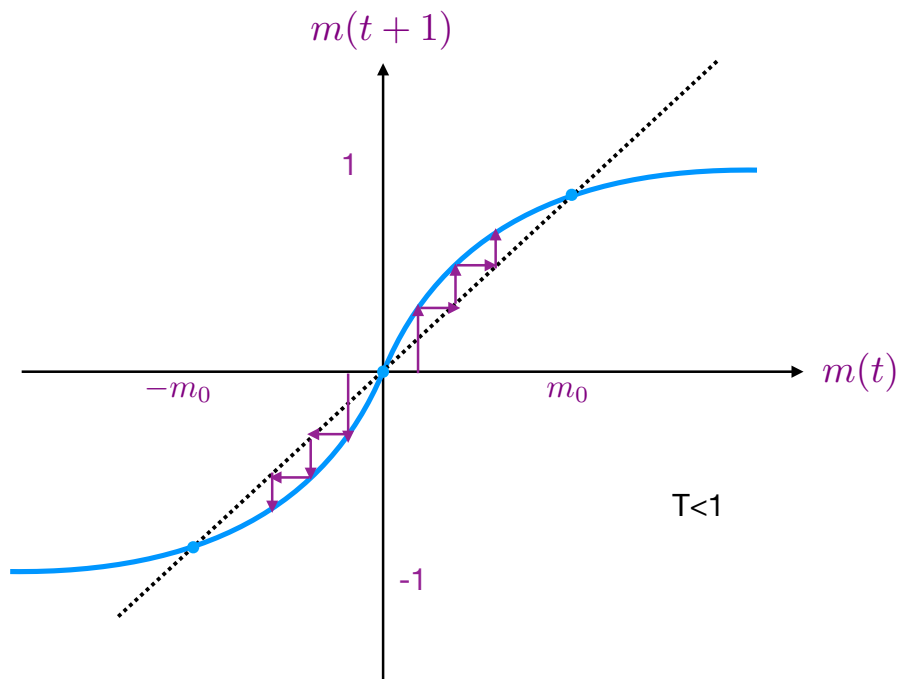
در این صورت معادله (۶۴)



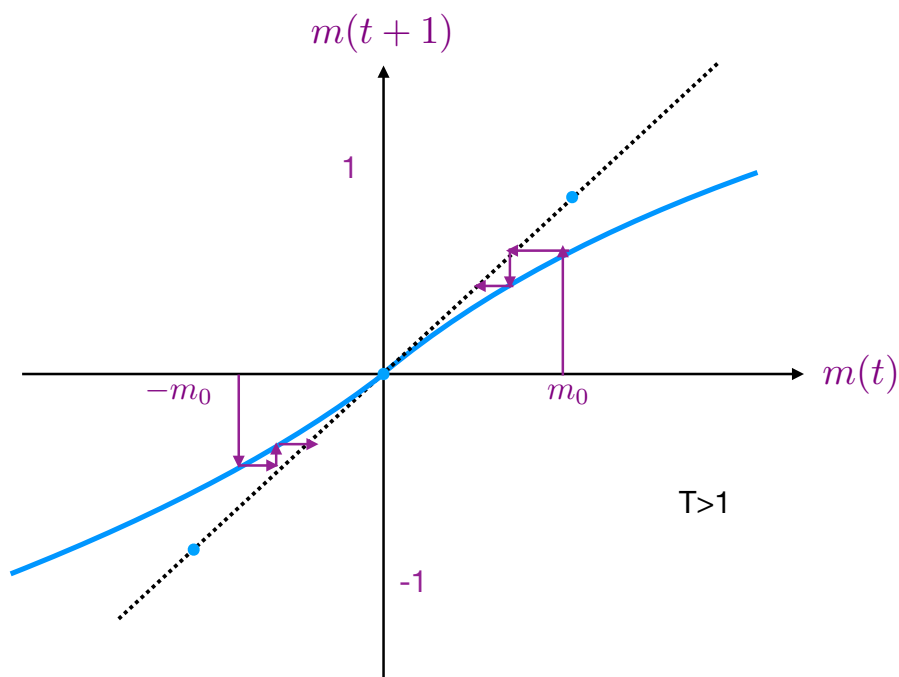
به شکل زیر در می آید:

$$m(t+1) = \tanh \beta m(t). \quad (71)$$

حال از شکل (۹) می توان برای حل این معادله دینامیکی و یافتن نقاط تعادل آن استفاده کرد. شکل های زیر جهت حرکت را در حالت های مختلف نشان می دهد. دو نقطه در نزدیکی  $m = 0$  در نظر گرفته شده است. دیده می شود که وقتی  $T < 1$  است، نقطه  $m = 0$  ناپایدار است و وقتی  $T > 1$  است، نقطه  $m = 0$  یک نقطه ناپایدار است.



شکل ۱۰: حل معادله دینامیکی  $m(t+1) = \tanh \beta m(t)$ . وقتی که  $\beta > 1$  است، حل های پایدار حل های غیر صفر هستند.



شکل ۱۱: حل معادله دینامیکی  $m(t+1) = \tanh \beta m(t)$  وقتی که  $\beta < 1$  است، حل پایدار حل صفر است.

■ مثال دوم: در این مثال داریم:

$$\omega_{ij} = \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)}. \quad (۷۲)$$

این نوع سیناپس نشانه آن است که شبکه عصبی طرح های  $\sigma^{(1)}$  تا  $\sigma^{(p)}$  را یاد گرفته است. حال می خواهیم ببینیم تحت چه شرایطی این طرح ها را به یاد می آورد. معادله دینامیکی عبارت است از:

$$s_i(t+1) = \tanh \beta \left[ \sum_{j=1}^N \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right] s_j(t). \quad (۷۳)$$

به دنبال یک حل پایدار از این سیستم معادلات هستیم. بازهم از شهود خود و حلی که برای شبکه های تعیینی به دست آورده ایم استفاده می کنیم. در حالت تعیینی می دانیم که هر کدام از طرح های  $\sigma^{(\nu)}$  یک حل پایاست. بنابراین در این جا هم مطابق با همان استدلالی که در مثال اول

دیدیم در دو طرف معادله قرار می دهیم  $s_i = m\sigma_i^{(\nu)}$  و بدست می آوریم:

$$m\sigma_i^{(\nu)} = \tanh \beta \left[ \left( \sum_{j=1}^N \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \right) m\sigma_j^{(\nu)} \right]. \quad (74)$$

بازهم جمع روی  $\mu$  را به دو قسمت  $\mu = \nu$  و دو قسمت  $\mu \neq \nu$  می شکنیم. در این صورت داریم:

$$m\sigma_i^{(\nu)} = \tanh \beta \left[ m\sigma_i^{(\nu)} + \frac{m}{N} \sum_{j=1}^N \sum_{\mu \neq \nu}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)} \sigma_j^{(\nu)} \right]. \quad (75)$$

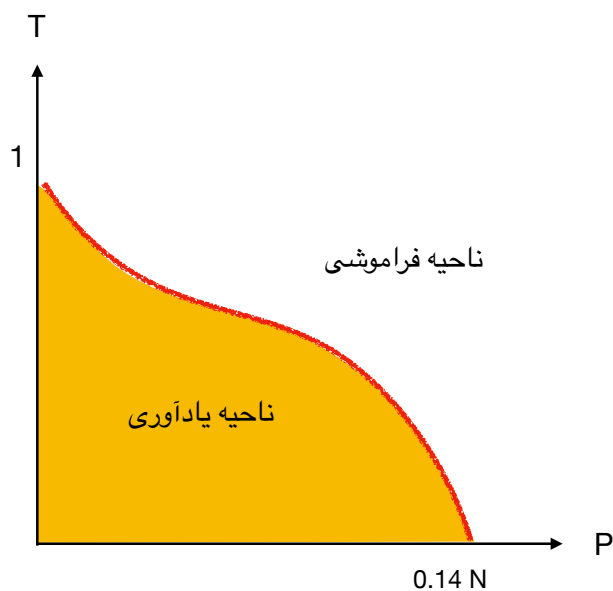
که از آن نتیجه می گیریم

$$m\sigma_i^{(\nu)} = \tanh \beta \left[ m\sigma_i^{(\nu)} + \frac{m}{N} O(\sqrt{Np}) \right]. \quad (76)$$

بنابراین اگر  $p \ll N$  باشد، خواهیم داشت:

$$m\sigma_i^{(\nu)} \approx \tanh \beta m\sigma_i^{(\nu)}. \quad (77)$$

که همان معادله ای است که برای مثال اول داشتیم. بنابراین در مورد  $p$  طرح نیز همان نوع گذار فاز را در نقطه  $T = 1$  می بینیم. واضح است که این نتیجه در حدی که  $p$  بسیار کوچکتر از  $N$  باشد معتبر است. وقتی که تعداد طرح ها زیاد شود ناگزیر بین آنها همبستگی بوجود می آید و نتایج بالا تغییر می کند. در آینده بعدی نشان خواهیم داد که دیاگرام فاز به  $T$  و  $p$  بستگی دارد که دیاگرام فاز آن در شکل (۱۲) نشان داده شده است. بنابراین یک شبکه عصبی هاپفیلد نمی تواند بیش از  $0.14N$  تصویر را در خود ذخیره کند چرا که در غیر این صورت هیچ کدام از این طرح های ضبط شده را نمی تواند به یاد بیاورد.

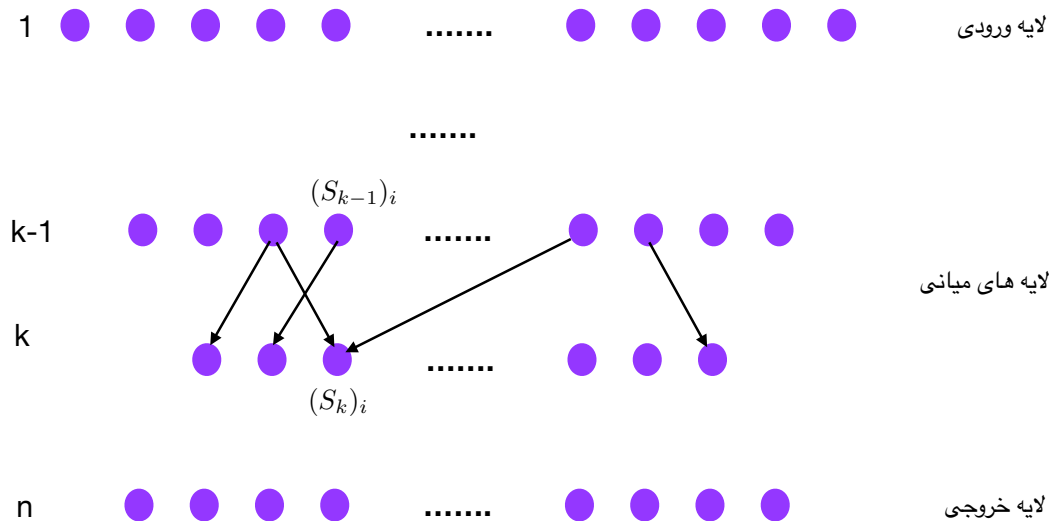


شکل ۱۲: دیاگرام فاز شبکه هاپفیلد. تنها در ناحیه رنگ شده طرح‌ها به یاد آورده می‌شوند.

#### ۴ شبکه‌های سایبرنتیک

آنچه که تا کنون گفتیم یعنی به یاد سپردن و به یاد آوردن طرح‌ها (به معنای کلی آن) تنها یکی از کارکردهای مغز است. یک کار مهم دیگر مغز و دستگاه عصبی کنترل تمام فرایندها و واکنش نشان دادن به محرک‌هاست و به احتمال زیاد طی فرآیند تکامل بیولوژیک نخستین بار مغز برای چنین کاری در ساده‌ترین موجودات زنده شکل گرفته است. شبکه عصبی‌ای که کار آن کنترل و پاسخ دادن به محرک‌های بیرونی است ساختاری کاملاً متفاوت با شبکه عصبی مربوط به حافظه دارد. چنین شبکه‌ای که به آن شبکه سایبرنتیک<sup>۱۲</sup> می‌گوییم، ساخت جهت‌داری دارد که آن را برای دریافت ورودی از یک طرف و نشان دادن خروجی در طرف دیگر مناسب می‌سازد. ساده‌ترین این شبکه‌ها Feed Forward Layered Network نامیده می‌شوند که در شکل (۱۳) یک نمونه از آنها دیده می‌شود.

<sup>۱۲</sup>Cybernetic Network



شکل ۱۳: شمای کلی یک شبکه سایبرنتیک. بسته به نوع کاری که این شبکه انجام می دهد تعداد نورون ها و تعداد لایه ها می تواند متفاوت باشد.

ورودی این شبکه بالاترین لایه و خروجی آن پایین ترین لایه یا لایه  $n$  ام است و بقیه لایه ها لایه های میانی یا به اصطلاح لایه های پنهان هستند. نماد  $(S_k)_i$  برای نشان دادن اسپین یا نورون  $i$  ام در لایه  $k$  ام به کار می رود. هم چنین یک فرق مهم در این جا این است که میزان فعالیت نورون ها در شبکه سایبرنتیک الزاما یک مقدار گسسته نیست و می تواند پیوسته باشد. علیرغم این هم چنان نماد  $(S_k)_i$  را مقدار اسپین می نامیم. برای چنین شبکه ای داریم:

$$(S_k)_i = f(h_i) \quad (78)$$

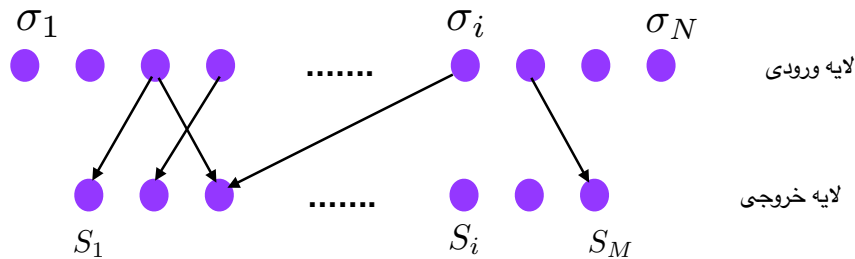
که در آن  $h_i$  مجموع سیناپس های ورودی به نورون  $i$  ام در لایه  $k$  ام است یعنی

$$h_i = \sum_j \omega_{ij} (S_{k-1})_j. \quad (79)$$

البته در اینجا فرض کرده ایم که نورون های هر لایه تنها از لایه قبلی تاثیر می گیرند ولی این حالت کلی نیست و در یک شبکه سایبرنتیک ممکن است یک لایه از لایه های قبل تر نیز تاثیر بگیرد. برای تابع  $f$  معمولاً یکی از دو شکل زیر در نظر گرفته می شود که شکل اول برای شبکه های متعین و شکل دوم برای شبکه های تصادفی است:

$$f(h_i) = \begin{cases} \text{sign}(h_i - \theta_i) & S_i = 1, -1, \\ \tanh \beta(h_i - \theta_i) & -1 < S_i < 1. \end{cases} \quad (۸۰)$$

ساده ترین شبکه عصبی سایبرنتیک که به آن پرسپترون<sup>۱۳</sup> نیز گفته می شود، شبکه ای است که هیچ لایه میانی ندارد. حالت نورون های ورودی این شبکه را با  $\sigma_i$  و حالت نورون های خروجی آن را با  $S_i$  نشان می دهیم، شکل (۱۴).



شکل ۱۴: یک پرسپترون. فقط یک لایه ورودی و یک لایه خروجی وجود دارد.

طبیعی است که در پرسپترون دیگر نماد  $k$  یعنی شماره لایه ها ضرورتی ندارد. از این شبکه در حالت ایده آل انتظار داریم که به محرک  $\sigma^{(\mu)}$  پاسخ  $\zeta^{(\mu)}$  را بدهد، ولی به دلیل این که این شبکه ایده آل نیست ممکن است پاسخ  $S^{(\mu)}$  را بدهد. در اینجا اندیس  $\mu$  نشان دهنده شماره محرک

<sup>۱۳</sup> Perceptron

و پاسخ است. در واقع انتظار ما این است که این شبکه به مجموعه ای از محرک های

$$\{\sigma^{(1)}, \sigma^{(2)}, \dots, \sigma^{(p)}\}$$

پاسخ های

$$\{\zeta^{(1)}, \zeta^{(2)}, \dots, \zeta^{(p)}\}$$

را بدهد. برای این که این شبکه چنین کاری انجام دهد کافی است که سیناپس ها را به شکل زیر تنظیم کنیم:

$$\omega_{ij} = \frac{1}{p} \sum_{\mu=1}^N \zeta_i^{(\mu)} \sigma_j^{(\mu)}. \quad (81)$$

فعلا نیز شبکه را به صورت تعیینی در نظر می گیریم. در این صورت برای ورودی  $\sigma_i^{(\mu)}$  حتما خروجی دلخواه  $\zeta_i^{(\mu)}$  را خواهیم داشت:

$$S_i = \text{sign}\left(\frac{1}{N} \sum_{j,\nu} \zeta_i^{(\nu)} \sigma_j^{(\nu)} \sigma_j^{(\mu)}\right) = \sigma_i^{(\mu)}. \quad (82)$$

■ تمرین: این نوع استدلال را قبلا دیده ایم. نشان دهید که رابطه بالا در حد  $p \ll N$  درست است.

اما نکته این است که در این صورت شبکه عصبی هیچ کار خاصی برای ما انجام نداده است. آنچه که ما واقعا می خواهیم این است که یک مجموعه کوچکتری (معمولا خیلی کوچکتر) از ورودی ها و خروجی های مورد انتظار را به شبکه بدهیم. به این مجموعه کوچکتر مجموعه یادگیری<sup>۱۴</sup> می گوئیم. انتظار ما این است که این شبکه از روی این مجموعه یادگیری سیناپس هایش را طوری تنظیم کند که بتواند پاسخ های صحیح را برای مجموعه بزرگ تری از ورودی ها نیز ارائه کند. برای این کار باید شبکه عصبی مجهز به روشی برای تغییر پله پله سیناپس هایش در جهت درست و ایده آل باشد طوری که هر چه مجموعه یادگیری اش بزرگ تر باشد، شکل سیناپس های نهایی اش به شکل صحیحی که در بالا گفته شد نزدیک تر باشد. البته عملکرد شبکه عصبی وقتی خوب و قابل تحسین است که به ازای مجموعه یادگیری کوچکی بتواند پاسخ های صحیح را برای مجموعه خیلی بزرگتری فراهم کند. حال سوال این است که چنین شیوه ای برای تنظیم سیناپس های شبکه عصبی کدام است. باید دقت کرد که طرز کار شبکه عصبی و عاملی که باعث موفقیت آن در پاسخگویی به محرک ها می شود، مبتنی بر درک رابطه منطقی بین محرک و پاسخ نیست، بلکه شبکه عصبی تنها با تنظیم سیناپس های خود و بدون درک منطق و ساختار یک مسئله، به صورت طوطی وار یاد می گیرد که در پاسخ به یک مجموعه بزرگ از محرک ها که معمولا بسیار بزرگ تر از مجموعه آموزشی آن است، چگونه رفتار کند. در فرآیند یادگیری، سیناپس های شبکه عصبی می بایست بخوبی تعیین شوند. معمولا می توان از یک انتخاب اولیه برای سیناپس شروع کرد و بسته به پاسخ هایی که می گیریم سیناپس ها را کم کم تصحیح کنیم. در زیر این روش را توضیح می دهیم.

<sup>۱۴</sup> Training Set

## ۱.۴ یادگیری با استفاده از شیب

هدف این نوع آموزش که یادگیری به روش شیب<sup>۱۵</sup> نامیده می شود، این است که فاصله میان پاسخ های مطلوب و پاسخ های واقعی به حداقل کاهش یابد. در اینجا اندیس  $\mu$  برای اعضای مجموعه یادگیری بکار می رود. برای این منظور تابع فاصله را به شکل زیر تعریف می کنیم:

$$D := \frac{1}{2} \sum_{i,\mu} (\zeta_i^\mu - S_i^\mu)^2 = \sum_{i,\mu} (1 - \zeta_i^\mu S_i^\mu) = \sum_{i,\mu} (1 - \zeta_i^\mu f(h_i^\mu)). \quad (۸۳)$$

این فاصله تابعی از سیناپس های  $\omega_{ij}$  و استانه های  $\theta_i$  است. می بایست در هر گام این پارامترها چنان تغییر داد که این فاصله کمتر شود. به عبارت بهتر می نویسیم:

$$\delta D = \sum_{i,j} \frac{\partial D}{\partial \omega_{ij}} \delta \omega_{ij} + \sum_i \frac{\partial D}{\partial \theta_i} \delta \theta_i. \quad (۸۴)$$

حال اگر در هر مرحله قرار دهیم

$$\delta \omega_{ij} \propto -\frac{\partial D}{\partial \omega_{ij}}, \quad \delta \theta_i \propto -\frac{\partial D}{\partial \theta_i} \quad (۸۵)$$

این فاصله در هر مرحله کم و کمتر خواهد شد. بنابراین کافی است که مشتق های  $D$  را نسبت به پارامترهای شبکه یعنی سیناپس ها و  $\theta_i$  ها محاسبه می کنیم.

$$\frac{\partial D}{\partial \omega_{jk}} = \sum_{i,\mu} -\zeta_i^\mu f'(h_i^\mu) \frac{\partial h_i^\mu}{\partial \omega_{jk}}. \quad (۸۶)$$

اما داریم:

$$h_i^\mu = \sum_k \omega_{ik} \sigma_k^\mu - \theta_i, \quad (۸۷)$$

و در نتیجه بدست می آوریم:

$$\frac{\partial h_i^\mu}{\partial \omega_{jk}} = \delta_{ij} \sigma_k^\mu. \quad (۸۸)$$

بنابراین:

$$\frac{\partial D}{\partial \omega_{jk}} = -\sum_{i,\mu} \zeta_i^\mu f'(h_i^\mu) \delta_{ij} \sigma_k^\mu$$

---

<sup>۱۵</sup> Learning by Gradient Method



$$= -\sum_{\mu} \zeta_j^{\mu} f'(h_j^{\mu}) \sigma_k^{\mu} = -\sum_{\mu} \Delta_j^{\mu} \sigma_k^{\mu}, \quad (89)$$

که در آن

$$\Delta_j^{\mu} = \zeta_j^{\mu} f'(h_j^{\mu}). \quad (90)$$

حال می توان سیناپس ها را در جهت عکس گرادیان تغییر داد و مطمئن بود که برای تغییرات کم، فاصله ی  $D$  کم خواهد شد. بنابراین قرار می دهیم:

$$\delta \omega_{jk} = \epsilon \sum_{\mu} \Delta_j^{\mu} \sigma_k^{\mu}. \quad (91)$$

همین کار را می بایست برای تغییر آستانه های  $\theta_i$  نیز به کار برد، یعنی:

$$\begin{aligned} \frac{\partial D}{\partial \theta_l} &= -\sum_{i,\mu} \zeta_i^{\mu} f'(h_i^{\mu}) \frac{\partial h_i^{\mu}}{\partial \theta_l} \\ &= \sum_{i,\mu} \zeta_i^{\mu} f'(h_i^{\mu}) \delta_{il} = \sum_{\mu} \zeta_l^{\mu} f'(h_l^{\mu}) \end{aligned} \quad (92)$$

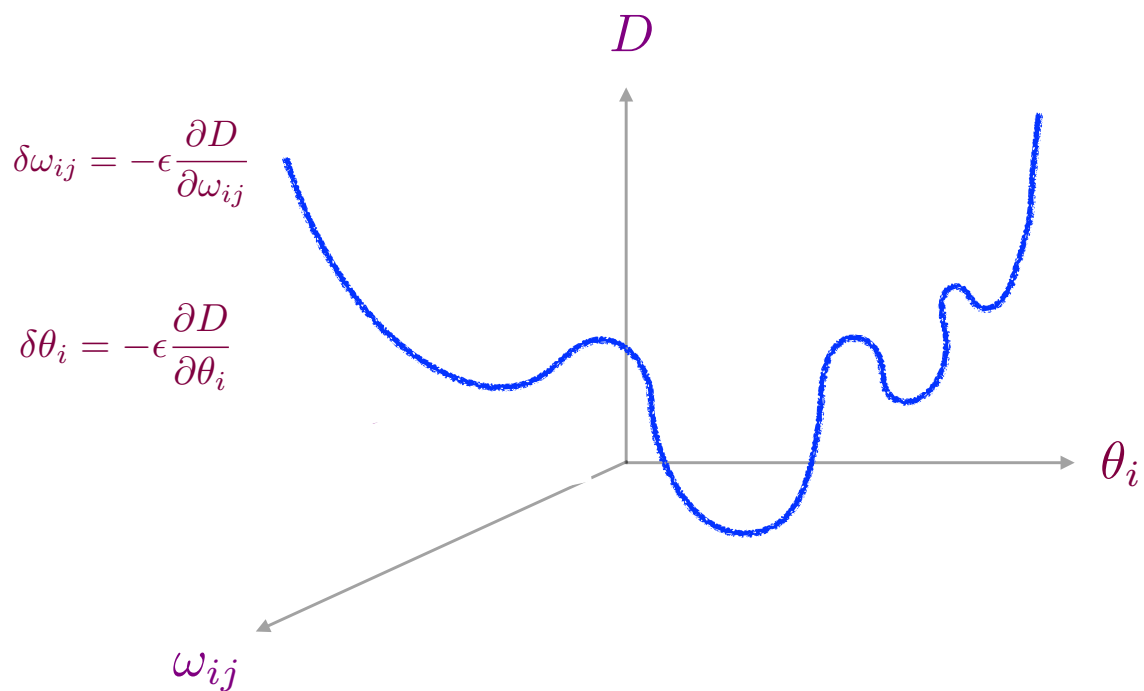
که از آن نتیجه می شود:

$$\delta \theta_l = -\epsilon \sum_{\mu} \zeta_l^{\mu} f'(h_l^{\mu}). \quad (93)$$

به ای ترتیب می توان سیناپس های  $\omega_{ij}$  و آستانه های  $\theta_i$  را تا زمانی که گرادیان به صفر می رسد یا خیلی کوچک می شود ادامه داد. روشی که در بالا برای یک پرسپترون بکار بردیم می توان برای شبکه های چندلایه نیز به کار برد. در این روش نخست سیناپس های آخرین لایه تصحیح می شوند و سپس سیناپس های لایه ماقبل آخر و همین طور تا سیناپس های لایه اول تصحیح می شوند. به همین دلیل این روش، روش تصحیح خطای رو به عقب<sup>۱۶</sup> نامیده می شود.

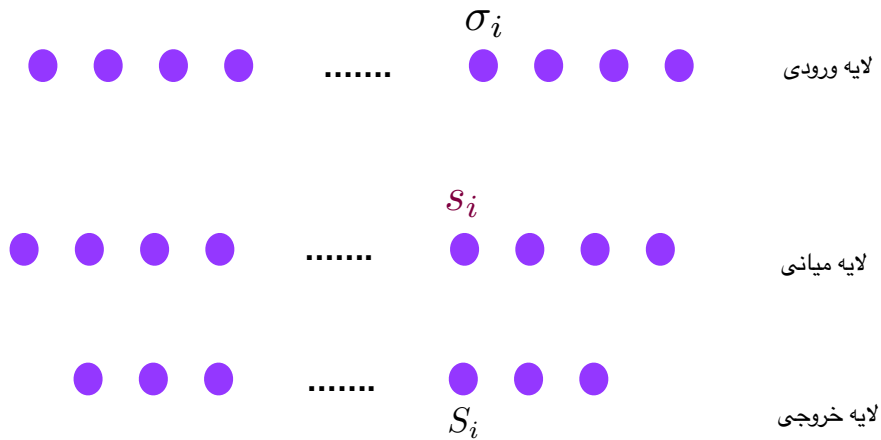
---

<sup>۱۶</sup> Learning by error-back propagation



شکل ۱۵: روش گرادیان برای تعیین شدت سیناپس ها.

برای درک روش گرادیان برای شبکه های چندلایه به شکل شکل (۱۶) توجه می کنیم. این شکل یک شبکه عصبی را با یک لایه درونی نشان می دهد.



شکل ۱۶: یک شبکه عصبی سایبرنتیک با یک لایه درونی.

متغیرهایی که حالت نورون ها را در لایه های مختلف نشان می دهند و هم چنین سنابس ها و آستانه ها با روابط زیر مشخص تعریف شده اند.

داریم:

$$\begin{aligned} s_i &= f\left(\sum_j \bar{\omega}_{ij} \sigma_j - \bar{\theta}_i\right) \\ S_i &= f\left(\sum_j \omega_{ij} s_j - \theta_i\right). \end{aligned} \quad (94)$$

داریم:

$$D = \sum_{i,\mu} (1 - \zeta_i^{(\mu)} S_i^{(\mu)}) \quad (95)$$

روابط (۹۴) نشان می دهند که این فاصله تابعی از  $\omega_{ij}$  ها  $\bar{\omega}_{ij}$  ها  $\theta_i$  ها و  $\bar{\theta}_i$  هاست. می توان مشابه با آنچه که در مورد شبکه های دولایه

ای انجام دادیم مشتقات  $D$  را نسبت به این متغیرها حساب کنیم و سپس پارامترهای شبکه را به شکل زیر تغییر دهیم:

$$\delta\omega_{ij} = -\epsilon \frac{\partial D}{\partial \omega_{ij}} \quad \delta\bar{\omega}_{ij} = -\epsilon \frac{\partial D}{\partial \bar{\omega}_{ij}}$$

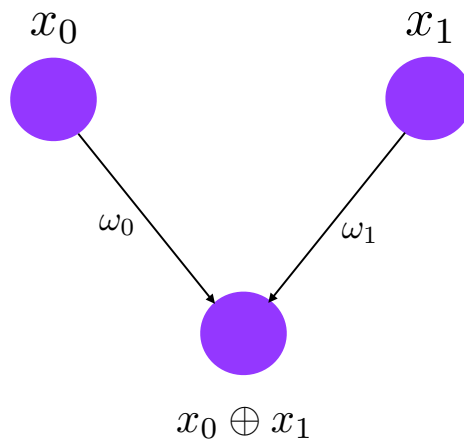
$$\delta\theta_i = -\epsilon \frac{\partial D}{\partial \theta_i} \quad \delta\bar{\theta}_i = -\epsilon \frac{\partial D}{\partial \bar{\theta}_i}. \quad (96)$$

با تغییرات پله پله سیناپس ها و آستانه ها مطابق با روابط بالا مطمئن خواهیم بود که فاصله در هر مرحله کمتر می شود.

آیا واقعا به لایه های میانی احتیاج داریم؟ نمی توان هر نوع شبکه ای را تنها با یک لایه ورودی و یک لایه خروجی و تنظیم مناسب سیناپس ها و آستانه ها ساخت؟ یک مثال ساده نشان می دهد که این کار امکان پذیر نیست. فرض کنید که می خواهیم شبکه ای بسازیم که یک تابع ساده مثل تابع منطقی  $XOR$  را پیاده سازی کند. این تابع با جدول منطقی زیر تعریف می شود:

$x_1$	$x_0$	$x_0 \oplus x_1$
0	0	0
0	1	1
1	0	1
1	1	0

یک پرسپترون که بخواهد این تابع را پیاده سازی کند در شکل (۱۷) نشان داده شده است.



شکل ۱۷: یک پرسپترون بدون لایه درونی که قرار است تابع  $xor$  را پیاده سازی کند.

در این صورت می بایست داشته باشیم:

$$x_0 \oplus x_1 = \Theta(\omega_0 x_0 + \omega_1 x_1 - \theta). \quad (97)$$

دقت کنید که در این بخش استثنائاً بجای متغیرهای اسپینی  $s$  مقادیر  $-1$  و  $1$  اختیار می کردند، از متغیرهای بول یا همان متغیرهای نورونی  $x$  استفاده می کنیم که مقادیر  $0$  و  $1$  اختیار می کنند. به همین دلیل نیز بجای تابع  $\text{sign}(x)$  از تابع  $\Theta(x) = \frac{1+\text{sign}(x)}{2}$  استفاده می کنیم. مطابق با جدول این تابع خواهیم داشت:

$$\begin{aligned} 0 &= \Theta(-\theta) \\ 1 &= \Theta(\omega_0 - \theta) \\ 1 &= \Theta(\omega_1 - \theta) \\ 0 &= \Theta(\omega_0 + \omega_1 - \theta), \end{aligned} \quad (98)$$

و یا

$$\begin{aligned} 0 &< \theta \\ 0 &< \omega_0 - \theta \\ 0 &< \omega_1 - \theta \\ 0 &< -\omega_0 - \omega_1 + \theta. \end{aligned} \quad (99)$$

و بوضوح این نامساوی ها با یک دیگر ناسازگارند.

این مثال ساده نشان می دهد که استفاده از شبکه های سایبرنتیکی که بیش از یک لایه دارند اجتناب ناپذیر است.

### ■ تمرین: (موعد تحویل: دهم تیرماه ۱۳۹۹)

نشان دهید که تابع  $xor$  را با یک پرسپترون که تنها یک لایه میانی با دو نورون دارد، می توان پیاده سازی کرد.

## ۲.۴ نمایش یک تابع دلخواه از متغیرهای دوتایی

فرض کنید یک شبکه با  $n$  تا نورون ورودی و یک نورون خروجی داریم. این شبکه ممکن است لایه های درونی نیز داشته باشد. حال از خود سوال می کنیم که آیا ممکن است سیناپس های این شبکه را چنان تعیین کرد که بتواند یک تابع دلخواه مثل  $\{0, 1\} \rightarrow \{0, 1\}^n$  را نمایش دهد، یعنی به ازای هر ورودی  $x \in \{0, 1\}^n$  در نورون خروجی مقدار  $f(x)$  پدیدار شود. دیدیم که اگر شبکه لایه میانی نداشته باشد پاسخ این سوال منفی است. مثال نقض کننده آن هم تابع  $XOR$  است. در این جا می خواهیم نشان دهیم که اگر شبکه عصبی دارای تنها یک لایه میانی باشد، آنگاه می توان هر تابع دلخواهی را با انتخاب سیناپس های مناسب نمایش داد. در واقع نشان خواهیم داد که برای نمایش چنین تابعی یک لایه میانی با  $2^n$  نورون کافی است. دقت کنید که ممکن است این شبکه بهترین شبکه ممکن نباشد. هم چنین دقت کنید که تعداد نورون های لایه میانی به صورت نمایی با  $n$  رشد می کند و این موضوع نشان می دهد که چرا نمی توان یک شبکه عصبی را به صورت سازنده <sup>۱۷</sup> برای نمایش هر تابعی ساخت و می بایست برای آموزش آن بر یک فرایند یادگیری تکیه کرد. ایده کلی اثبات را با ذکر یک مثال نشان می دهیم. خواننده با مطالعه این مثال و حل تمرین بعدی می تواند استدلال قضیه را برای حالت کلی دریابد.

■ مثال ۱: جدول (۲.۴) یک تابع  $\{0, 1\} \rightarrow \{0, 1\}^2$  را نشان می دهد.  $f_0$  تا  $f_3$  یک یا صفر هستند.

$x_1$	$x_0$	$f(x)$
0	0	$f_0$
0	1	$f_1$
1	0	$f_2$
1	1	$f_3$

جدول ۱: جدول مربوط به مثال یک

در واقع مقادیر سمت چپ این جدول نمایش اعداد 0 تا 3 در مبنای دو هستند. لایه اول طبیعتاً دو نورون دارد. در حالت کلی لایه میانی دارای  $2^n$  نورون است که هر کدام مطابق با نمایش دو-دویی یکی از اعداد  $m \in \{0, 2^n - 1\}$  هستند. در این مثال نورون های لایه وسط متناظر با اعداد 0، 1، 2، و 3 هستند. سیناپس های لایه اول طوری انتخاب شده اند که هرگاه نمایش دو-دویی یک عدد در لایه اول قرار گرفت تنها نورون

<sup>۱۷</sup>Constructive

مربوط به آن عدد در لایه میانی روشن شده و بقیه نورون ها خاموش بمانند. به این ترتیب می خواهیم چنین جدولی داشته باشیم که در آن وضعیت نورون های لایه وسط نیز نشان داده شده اند:

$x_1$	$x_0$	$s_0$	$s_1$	$s_2$	$s_3$	$f(x)$
0	0	1	0	0	0	$f_0$
0	1	0	1	0	0	$f_1$
1	0	0	0	1	0	$f_2$
1	1	0	0	0	1	$f_3$

جدول ۲: جدول مربوط به مثال یک وقتی که مقادیر نورون ها در لایه میانی نیز داده شده اند. مقادیر همه نورون ها به صورت صفر و یک نشان داده شده اند.

از این جدول نخست در می یابیم که تابع  $f$  به صورت زیر است:

$$f = f_0s_0 + f_1s_1 + f_2s_2 + f_3s_3. \quad (100)$$

حال از این اتحاد ساده استفاده می کنیم که با توجه به تعریف تابع علامت برای هر مقدار  $a \in \{0, 1\}$  رابطه زیر برقرار است:

$$a = \Theta\left(a - \frac{1}{2}\right). \quad (101)$$

بنابراین از این رابطه استفاده می کنیم و تابع  $f$  را به صورت زیر می نویسیم:

$$f = \Theta\left(f_0s_0 + f_1s_1 + f_2s_2 + f_3s_3 - \frac{1}{2}\right) = \Theta\left(\sum_i f_i s_i - \frac{1}{2}\right). \quad (102)$$

این رابطه سیناپس های لایه آخر و هم چنین مقدار آستانه آن را تعیین می کند، یعنی

$$\omega_i = f_i \quad \theta = \frac{1}{2}.$$

حال به سراغ لایه اول می رویم. همان جدول بالا نشان می دهد که رابطه زیر برقرار است:

$$s_0 = \overline{x_1} \overline{x_0}, \quad s_1 = \overline{x_1} x_0, \quad s_2 = x_1 \overline{x_0}, \quad s_3 = x_1 x_0, \quad (103)$$

که در آن  $\bar{x}_i = 1 - x_i$  اگر طرف راست این عبارت ها ترکیب های خطی از  $x_i$  ها بودند می توانستیم بلافاصله سیناپس ها را تعیین کنیم. اما این عبارت ها به صورت ضربی و از درجه دوم هستند. پس نخست این عبارت های ضربی را به صورت جمع می نویسیم. این کار البته برای متغیرهای صفر و یک امکان پذیر است. به طور کلی به ازای هر دو متغیر  $x, y \in \{0, 1\}$  می خواهیم بنویسیم:

$$xy = \Theta(\alpha x + \alpha y - \theta). \quad (104)$$

با در نظر گرفتن مقادیر مختلف برای  $x$  و  $y$  می فهمیم که این رابطه وقتی برقرار است که داشته باشیم:

$$0 = \Theta(-\theta), \quad 0 = \Theta(\alpha - \theta), \quad 1 = \Theta(2\alpha - \theta) \quad (105)$$

که با انتخاب

$$\alpha = 1, \quad \theta = 2$$

براحتی برقرار می شود. بنابراین خواهیم داشت:

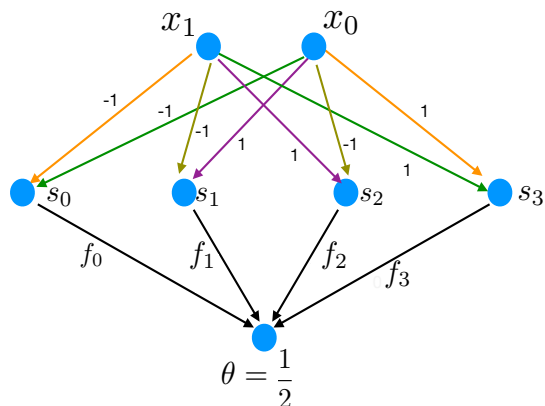
$$xy = \Theta(x + y - 2).$$

با استفاده از این رابطه و رابطه (103) و توجه به اینکه  $\bar{x}_i = 1 - x_i$  بدست می آوریم:

$$\begin{aligned} s_0 &= \Theta(-x_1 - x_0) \\ s_1 &= \Theta(-x_1 + x_0 - 1) \\ s_2 &= \Theta(x_1 - x_0 - 1) \\ s_3 &= \Theta(x_1 + x_0 - 2). \end{aligned} \quad (106)$$

به این ترتیب سیناپس ها و پتانسیل های آستانه برای نورون های لایه اول نیز بدست می آید.





شکل ۱۸: شبکه مربوط به مثال یک

■ **تمرین: (موعد تحویل: دهم تیرماه ۱۳۹۹).**

الف: برای یک تابع  $f : \{0, 1\}^3 \rightarrow \{0, 1\}$  ساختار شبکه و سیناپس هایی که این تابع را نمایش می دهد و هم چنین پتانسیل های آستانه را برای نورون های لایه ها بدست آورید.

ب: از آنچه که در باره مثال یک و قسمت اول این تمرین یاد گرفتید حالت کلی را برای تابع  $f : \{0, 1\}^n \rightarrow \{0, 1\}$  حل کنید.

## ۵ کاربردها

بعد از این مقدمات به ذکر چند مثال از کاربردهای شبکه های عصبی می پردازیم. باید به یاد داشته باشیم که شبکه عصبی با درک منطق درونی یک مسئله نیست که قادر به حل آن می شود بلکه با آموزش دیدن برای یک مجموعه کوچک از مثال ها قادر می شود که مجموعه بزرگ تری از مثال ها را نیز حل کند. این نحوه یادگیری درست مثل یادگیری کودک یا نوزاد هر موجود زنده دیگری است که با مشاهده دنیای اطراف به تدریج واکنش های خود را تصحیح می کند. امروزه با رشد سریع حجم داده ها از یک طرف و افزایش قدرت و سرعت پردازش این داده ها از طرف

دیگر، مجموعه کاربردهای شبکه های عصبی یا آنچه که امروزه به نام یادگیری ماشینی<sup>۱۸</sup> خوانده می شود، به سرعت در حال گسترش است. طبیعتاً ما فقط می توانیم به گوشه ای از این کاربردها اشاره کنیم آن هم بدون پرداختن به جزئیات. خواننده علاقمند می تواند برای فهم بهتر هرکدام از این کاربردها به کتاب های تخصصی این موضوع نگاه کند. یک منبع خوب کتاب زیر است: B. Neural Networks, An Introduction, Muller, J. Reinhardt, and M. T. Strickland; Springer 1995.

## ۱.۵ ساختار پروتئین ها

■ هر پروتئین یا ماکرومولکول رشته ای بسیار بلند است که از توالی مجموعه ای از ۲۱ آمینو اسید تشکیل شده است. این که در ساختار یک پروتئین چه نوع آمینو اسیدهایی و به چه ترتیبی به کار رفته است ساختار اولیه<sup>۱۹</sup> آن پروتئین نامیده می شود. به محض آنکه ساختار اولیه پروتئین تعیین شد، بلافاصله پروتئین به دور خود می پیچد و ساختار ثانویه ای<sup>۲۰</sup> پیدا می کند که تناظر یک به یک با آن ساختار اولیه دارد. ساختار های ثانویه به سه گروه کلی مارپیچ آلفا<sup>۲۱</sup> صفحه بتا<sup>۲۲</sup> و مارپیچ تصادفی<sup>۲۳</sup> تقسیم می شوند. در مرحله آخر ماکرومولکول پروتئین کاملاً در هم فشرده می شود و شکلی پیدا می کند که برای انجام اعمال بیولوژیک مشخص آماده می شود. در این حالت نهایی است که پروتئین مثل یک ابزار مشخص مثلاً قفل و کلید سلولی یا حمل کننده سلولی عمل می کند.

---

Machine Learning<sup>۱۸</sup>

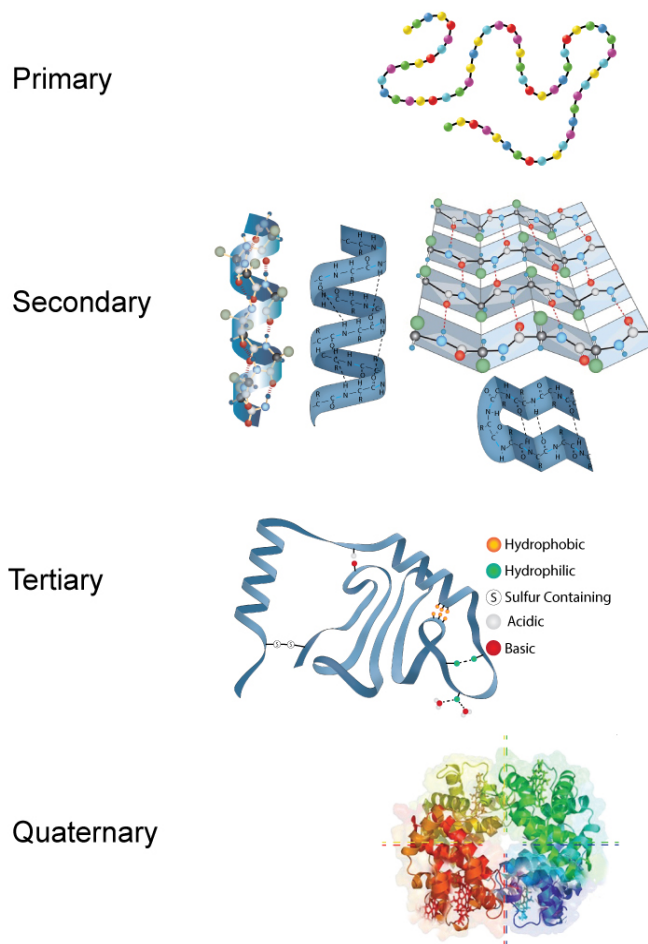
Primary Structure<sup>۱۹</sup>

Secondary Structure<sup>۲۰</sup>

$\alpha$ -helix<sup>۲۱</sup>

$\beta$ -Sheet<sup>۲۲</sup>

Random Coil<sup>۲۳</sup>



شکل ۱۹: ساختارهای چهارگانه پروتئین.

بنابراین، این که با مطالعه نیروهای میکروسکوپی بین اتم ها و آمینو اسیدها حتی برای وقتی که مدل های بسیار ساده ای برای آمینو اسیدها در نظر می گیریم، تعیین کنیم که یک پروتئین با ساختار اولیه ی معین چه نوع ساختار ثانوی ای پیدا می کند ، یعنی آن ساختار ثانویه ای که کمترین انرژی را دارد کدام ساختار است، مسئله بسیار دشوار ولی بغایت مهمی است. ( در این جا می توانیم برای سادگی دما را صفر فرض کنیم ، در نتیجه فقط حالت پایه در ساختار پروتئینی برای ما مهم است. ) به یاد بیاوریم که پیدا کردن حالت پایه در یک مدل مکانیک آماری با برهم کنش های یکسان مثل مدل آیزینگ در هر بعدی بسیار ساده است. اما در اینجا با زنجیره ای رویرو هستیم که درجات آزادی

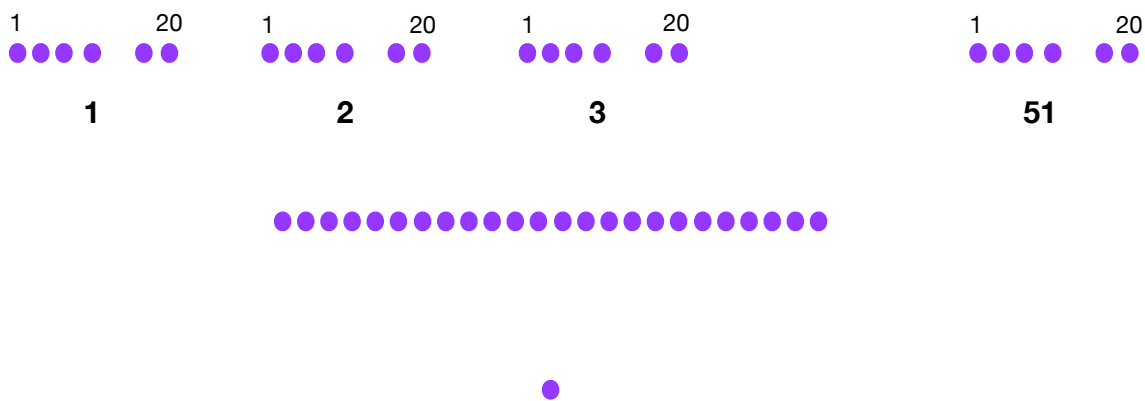
اش تنها اسپین نیست بلکه موقعیت های مکانی مونومرهاست و زنجیره همگن هم نیست. در واقع شکل های مختلفی که یک مدل ساده از پروتئین می تواند در فضای بدو بعدی یا سه بعدی (حتی اگر آن فضا را به صورت یک شبکه مربعی در نظر گرفته باشیم) اختیار کند، بقدری زیاد است که مسئله یافتن شکل های با کمینه انرژی را عملاً غیرممکن می کند. این مسئله به مسئله تا شدن پروتئین<sup>۲۴</sup> مشهور است. اما همانطور که گفتیم یک شبکه عصبی بدون اینکه قادر به درک روابط منطقی حاکم بر یک پدیده باشد می تواند با یادگیری طوطی وار از یک مجموعه مثال، راه حل های تقریبی را برای مجموعه ای بزرگ نیز پیدا کند. شبکه عصبی برای تعیین ساختار ثانویه از روی ساختار اولیه مورد استفاده قرار گرفته است. در این مورد از شبکه خواسته شده است که تعیین کند آیا ساختار ثانوی یک پروتئین مشخص از نوع مارپیچ آلفا است یا خیر. برای مدل کردن ساختار اولیه پروتئین به ترتیب زیر عمل شده است. با توجه به اینکه همه پروتئین ها از ۲۱ نوع آمینو اسید ساخته می شوند، برای کد کردن نوع آمینو اسید از ۲۰ نورون استفاده شده است. ترتیب کد کردن به شکل زیر است.

$$\begin{aligned}
 0, 0, 0, \dots, 0, 0 &\leftrightarrow A_0 \\
 0, 0, 0, \dots, 0, 1 &\leftrightarrow A_1 \\
 0, 0, 0, \dots, 1, 0 &\leftrightarrow A_2 \\
 \dots 1, 0, 0, \dots, 0, 0 &\leftrightarrow A_{20} \quad .
 \end{aligned}
 \tag{۱۰۷}$$

این روش کد کردن مسلماً برای منظوره های دیگر (مثلاً برای ارسال اطلاعات) بهترین و فشرده ترین نوع کد کردن نیست، ولی برای شبکه عصبی و تعیین سیناپس ها و آموزش آنها بهترین روش است. با توجه به آنکه پروتئین های مطالعه شده همگی دارای ۵۱ آمینو اسید بوده اند. لایه اول که ساختار اولیه پروتئین را نشان می دهد دارای  $51 \times 20 = 1020$  نورون بوده است، شکل (۲۰). لایه دوم دارای ۴۰ نورون و لایه آخر نیز دارای یک نورون بوده است که ساختار مارپیچ الفا یا صفحه بتا را تعیین می کند. تعداد سیناپس ها نیز 40,000 تا بوده است. برای آموزش و تست این شبکه عصبی از بانک پروتئین آزمایشگاه بروکهیون<sup>۲۵</sup> استفاده شده. نهایتاً معلوم شده که این شبکه عصبی در هفتاد درصد موارد می تواند ساختار ثانویه را به درستی پیش بینی کند.

---

Protein Folding Problem<sup>۲۴</sup>  
Brookhaven National Laboratory<sup>۲۵</sup>



شکل ۲۰: شبکه مربوط به مثال یک

## ۲.۵ پیش بینی سری های زمانی

سری های زمانی<sup>۲۶</sup> سیستم های دینامیکی ساده ای هستند که تغییرات یک یا چند متغیر حقیقی را معمولاً در زمان های متوالی گسسته نمایش می دهند. این سری ها در پیش بینی وضع آب و هوا، پیش بینی های اقتصادی و بازارهای بورس و خیلی از زمینه های دیگر اهمیت دارند. در ساده ترین حالت تحول این سری های زمانی ممکن است به صورت زیر باشد:

$$x(t+1) = F[x(t)], \quad (108)$$

که در آن  $F[x]$  یک تابع مشخص است. اما نکته در این است که ما تنها به سری زمانی در یک بازه مشخص دسترسی داریم و خود تابع  $F[x]$  را نمی دانیم. چنین وضعی در مثال هایی که در بالا برشمردیم مثل پیش بینی هوا، یا بازار بورس و نظایر آن کاملاً قابل تصور است. بنابراین می توان گفت که سری زمانی مطابق با یک قانون مندی پیش می رود ولی ما از آن قانون مندی یعنی همان تابع  $F[x]$  بی خبریم و کارمان این است که این قانون مندی را از مجموعه محدودی از داده ها کشف کنیم و سپس آن را برای پیش بینی سری زمانی در آینده استفاده کنیم. در اینجا می توان از

<sup>۲۶</sup>Time Series

یک ماشین برای کشف این قانون مندی استفاده کرد. در این جا هدف این است که از یک مجموعه یادگیری

$$\{(x^\mu, x'^\mu = F[x^\mu])\} \quad \mu \in \{1, 2, \dots, p\}$$

ماشین بتواند تابع  $F[x]$  را با تقریب خوبی بدست آورد و به کمک آن بتواند سری زمانی را برای زمان های آینده نیز پیش بینی کند. نکته مهمی که در اغلب سیستم های دینامیکی وجود دارد این است که حتی وقتی تابع  $F[x]$  ساده است، رفتار سری زمانی آشوبناک است به این معنا که تغییری کوچک در شرایط اولیه باعث تغییرات بزرگ و غیرقابل پیش بینی در سری زمانی می شود. به عنوان نمونه یکی از این توابع تابع موسوم به لجستیک<sup>۲۷</sup> است که دانشجو می تواند به راحتی رفتار آشوبناک آن را ببیند. این تابع به صورت زیر است:

$$F[x] = 4x(1 - x), \quad 0 \leq x \leq 1. \quad (109)$$

در همین مثال جالب است که حتی از درون داده های آشوبناک نیز یک شبکه عصبی موفق شده است تا تابع  $F[x]$  را با تقریب خوبی بدست بیاورد. شبکه عصبی ای که در این مورد به کار رفته شامل یک نورون ورودی و یک نورون خروجی و پنج نورون میانی بوده است. هم چنین یک سیناپس نیز به طور مستقیم ورودی را به خروجی وصل کرده است. تابع آستانه برای فعال کردن نورون ها نیز به صورت زیر در نظر گرفته شده:

$$f(h) = \frac{1}{1 + e^{2h}}. \quad (110)$$

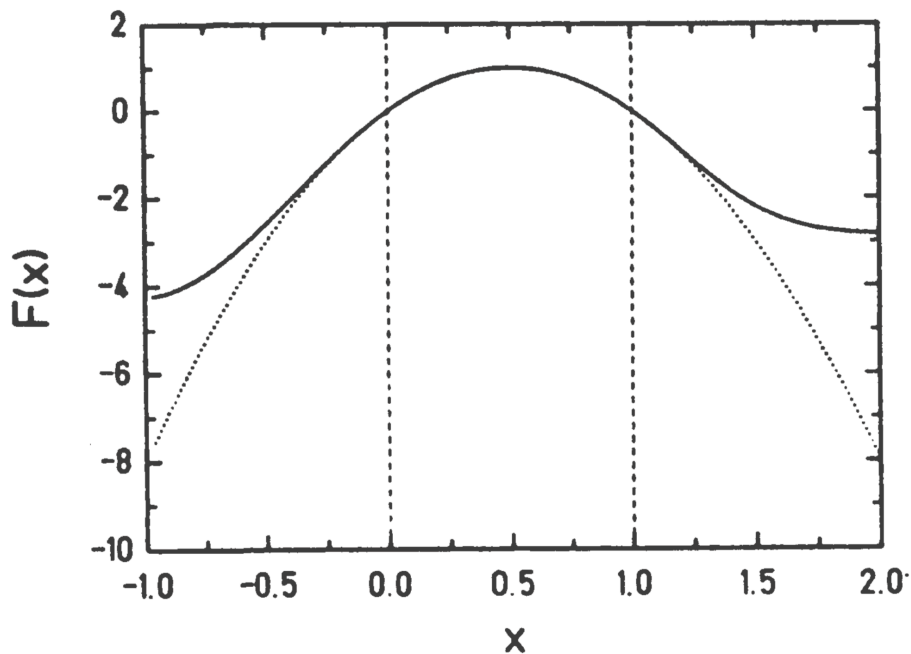
مجموعه آموزشی شامل هزار زوج ورودی و خروجی بوده و خطای ماشین در پیش بینی زوج های جدید، به صورت زیر بوده است:

$$\left\langle \frac{(x_{predicted}(t+1) - x_{correct}(t+1))^2}{x_{correct}^2(t+1)} \right\rangle = 1.4 \times 10^{-4}. \quad (111)$$

از این جالب تر این بوده که ماشین توانسته است شکل تابع  $F[x]$  را به صورت زیر حدس بزند:

$$F[x] = -0.64f(-1.11x - 0.26) - 1.3f(2.22x - 1.71) - 2.285(3.91x + 4.82) \\ - 3.905f(2.46x - 3.05) + 5.99f(1.68x + 0.60) + 0.31x - 2.04 \quad (112)$$

که در آن  $f(x)$  همان تابع آستانه است. شکل (۲۱) این تابع را با تابع واقعی مقایسه کرده است.

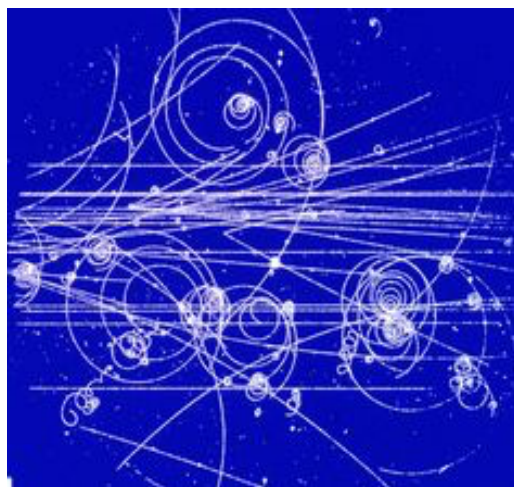


شکل ۲۱: مقایسه تابعی که شبکه عصبی از روی داده های آموزشی بدست آورده با تابع لجستیک. خط چین تابع لجستیک و خط پر تابعی است که ماشین بدست آورده و در محدوده  $0 \leq x \leq 1$  کاملاً بر هم منطبق اند.

### ۳.۵ یادگیری ماشینی در فیزیک انرژی های بالا

در شتابدهنده ها آنچه که از آشکارگرها حاصل می شود میلیونها تصویر درهم و برهم از مسیر ذرات اولیه، واپاشی آنها و تولید ذرات جدید است. این تصاویر به طرز سرگیجه آوری پیچیده هستند. می بایست با تجزیه تحلیل دقیق این تصاویر فهمید که در کجا یک ذره مثل ذره  $Z$  واپاشیده است و به چه ذراتی واپاشیده. در کجا یک جفت ذره مثل  $e^+, e^-$  یا  $\mu^+, m^-$  تولید شده و آیا اینکه جایی از این تصاویر نشاندهنده تولید ذره ای مثل ذره هیگس هست یا نه. آیا جایی در این تصاویر نشانه ای از تولید جفت کوارک سنگین دیده می شود، آیا جایی نشانه ای از تولید جت (آبشاری از ذرات تولید شده در برخورد) هست یا خیر؟ در زمان های قدیم یعنی مثلاً دهه هشتاد میلادی که تعداد این تصاویر تولید شده در شتابدهنده ها به صدها و یا هزار می رسید، این تصاویر توسط مجموعه بزرگی از کاربران آموزش دیده ولی غیرمتخصص تجزیه تحلیل می شدند. توان مغز انسانی

برای شناسایی و دسته بندی این طرح ها بیشتر از هر رایانه ای بود. اما امروزه که میلیون ها تصویر تولید می شود برای شناسایی حادثه هایی که در این عکس ها هست از شبکه های عصبی استفاده می شود.

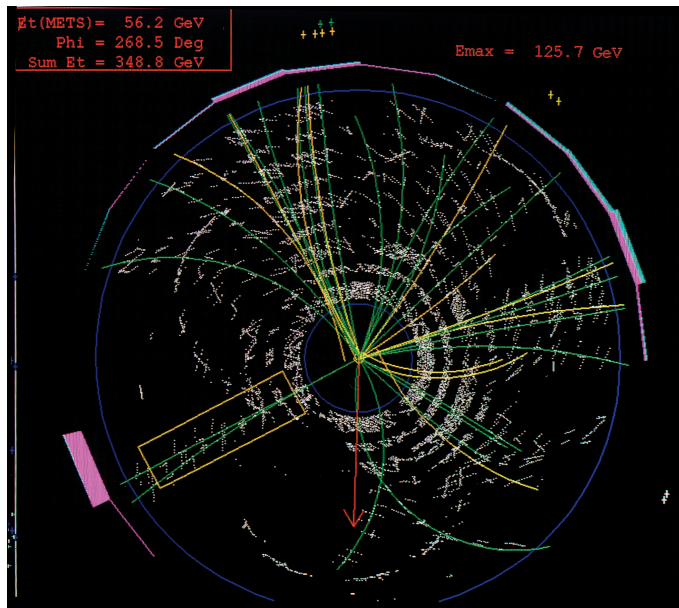


شکل ۲۲: یک تصویر نمونه از مسیرهای ذرات در یک آشکارگر ذرات بنیادی

در یک مورد مشخص، هدف آموزش ماشین برای تشخیص حادثه  $Z^0 \rightarrow b\bar{b}$  های  $Z^0$  یعنی واپاشی ذره  $Z^0$  واسطه نیروی برهم کنش ضعیف به یک جفت کوارک سنگین بوده است. مجموعه آموزشی شامل تعداد زیادی از نتایج واقعی آزمایش های شامل حادثه  $Z^0 \rightarrow b\bar{b}$  و فاقد آن بوده است. ماشین باید از این مجموعه محدود یاد بگیرد که تصویرهای آینده ای که به او داده می شود کدام یک شامل این حادثه هست و کدام یک نیست. به این کار اصطلاحاً برچسب زدن  $Z^0$  گفته می شود. شبکه عصبی ای که برای این منظور به کار رفته شامل ۴۳ تا ۴۵ نورون ورودی، و یک لایه میانی و یک نورون نهایی بوده است. یک مجموعه ده هزارتایی از حادثه برای آموزش به کار رفته که ۵۰۰۰ تای آن شامل کوارک  $b$  و ۵۰۰۰ تای دیگر شامل کوارک های از نوع دیگر بوده است. در وهله اول اطلاعات مسیرها در دستگاه مختصات دکارتی روی لایه ورودی شبکه کد شده که منجر به نتایج خوبی نشده ولی در مراحل بعد، اطلاعات مربوط به جت ها (انبوهی از ذرات که همگی در یک راستا حرکت می کنند و معمولاً در برهم کنش های قوی دیده می شوند) نشان دهنده برتری قابل توجه شبکه عصبی به روش های متداول بوده است.

Events<sup>۲۸</sup>  
Tagging<sup>۲۹</sup>





شکل ۲۳: یک نمونه از جت هایی که در برخورد های ذرات سنگین تولید می شوند.

به این مثال ها می توان انبوهی از مثال های برای کارهای دیگر نیز افزود مثل ماشین هایی برای تبدیل صدا به متن، تبدیل دست نوشته به نوشته تایپ شده، ترجمه، شناسایی طرح ها، انواعی از بازی ها و نظایر آن، اگر چه به نظر من مهم ترین و جذاب ترین استفاده از توانایی ماشین ها برای تجزیه تحلیل حجم عظیمی از داده هاست که در حوزه علوم خالص مثل کیهانشناسی، ذرات بنیادی، یا ژنتیک تولید می شوند. خواننده علاقمند می تواند این موضوعات را در کتاب ها یا مقالات مربوط به شبکه های عصبی مطالعه کند.

## ۶ ارتباط شبکه های عصبی و سیستم های اسپینی

آیا بین یک شبکه عصبی و یک سیستم اسپینی ارتباط عمیقی وجود دارد؟ آیا می توانیم از آنچه که در باره سیستم های اسپینی آموختیم برای مطالعه شبکه های عصبی استفاده کنیم؟ اگر رفتار یک شبکه عصبی کاتوره ای باشد، می توان انتظار داشت که وضعیت مجموعه نورون ها

$$(n_1, n_2, \dots, n_N) \quad n_i = \{0, 1\}$$

یا اسپین های متناظر با آنها

$$(s_1, s_2, \dots, s_N) \quad s_i = \{-1, 1\}$$

تنها به صورت احتمالاتی معلوم باشد. به عبارت بهتر در حالت تعادل باید تابعی مثل

$$P(s_1, s_2, \dots, s_N)$$

داشته باشیم که تعیین کند با چه احتمالی شبکه عصبی (یا اسپینی) در یک حالت مشخص است. در نتیجه در حالت تعادل تنها می توان گفت که وضعیت متوسط شبکه عصبی تنها با متوسط های زیر مشخص می شود:

$$(\langle s_1 \rangle, \langle s_2 \rangle, \dots, \langle s_N \rangle).$$

سوال این است که این متوسط ها چگونه حساب می شوند و نشان دهنده چه چیزی در حافظه شبکه عصبی هستند؟ از هم اکنون بدون انجام محاسبه می توانیم بگوییم که چه انتظاری داریم: فرض کنید که تنها یک طرح را ضبط کرده باشیم. در این صورت اگر حافظه شبکه عصبی خوب کار کند انتظار داریم وقتی که طرح اولیه  $(\langle \sigma_1 \rangle, \langle \sigma_2 \rangle, \dots, \langle \sigma_N \rangle)$  را به شبکه نشان می دهیم (یعنی حالت اولیه اش طرح فوق باشد) در اثر دینامیک تعینی شبکه عصبی نیز دقیقاً در این حالت قرار بگیرد. اما وقتی که شبکه به صورت کاتوره ای کار می کند در حالت تعادل اش اسپین ها دارای افت و خیز خواهند بود و تنها مقدار متوسط آنها معلوم خواهد بود. در نتیجه می توانیم بررسی کنیم که در حالت تعادل، شبکه عصبی چقدر شبیه به طرح ضبط شده است. به همین دلیل یک تابع شباهت به صورت زیر تعریف می کنیم:

$$q = \frac{1}{N} \sum_{i=1}^N \sigma_i \langle s_i \rangle \quad -1 \leq q \leq 1. \quad (113)$$

حال فرض کنید که دو طرح  $\sigma^1$  و  $\sigma^2$  به حافظه سپرده شده باشند. در این صورت می توانیم کمیت برداری  $\mathbf{q} = (q^1, q^2)$  را تعریف کنیم:

$$q^1 = \frac{1}{N} \sum_{i=1}^N \sigma_i^1 \langle s_i \rangle, \quad q^2 = \frac{1}{N} \sum_{i=1}^N \sigma_i^2 \langle s_i \rangle. \quad (114)$$

در این صورت معلوم می شود که شبکه در حالت تعادل چه شباهتی به طرح شماره یک و چه شباهتی به طرح شماره دو دارد. در حالت ایده آل انتظار داریم که شبکه دارای دو حالت تعادل متفاوت باشد که در یکی از آنها شبیه به طرح شماره یک و در دیگری شبیه به طرح شماره دو است. البته ممکن است حالت های تعادل دیگری هم داشته باشد که به هیچ کدام از این دو طرح شباهت زیادی نداشته باشد. این حرف ها را می توان به صورت کلی تعمیم داد که در آن بردار  $\mathbf{q} = (q^1, q^2, \dots, q^p)$  را تعریف می کنیم (برای وقتی که  $p$  طرح را ضبط کرده باشیم). هدف ما این است که این نقاط کمینه را پیدا کنیم و این بردارهای شباهت را مطالعه کنیم.

در بخش های گذشته دیدیم که دینامیک متعین یک شبکه عصبی یعنی روابط

$$s_i(t+1) = \text{sign}(h_i(t)) \quad (115)$$

همواره یک کمیت موسوم به انرژی یعنی

$$E[s] = -\frac{1}{2} \sum_{i,j} \omega_{ij} s_i s_j \quad (116)$$

را کاهش می دهد، یعنی هرگاه از یک طرح دلخواه شروع کنیم، به یک کمینه موضعی از تابع انرژی می رسیم. اما برای نورون های تصادفی رابطه (115) جای خود را به یک رابطه احتمالاتی می دهد به این معنا که برای چنین نورون هایی تنها می توانیم احتمالات گذار را تعیین کنیم. اگر پای احتمال در میان است، طبیعی است که بپرسیم احتمال اینکه شبکه اسپینی در یک حالت معین باشد چقدر است؟ برای پاسخ به این سوالات به عبارتی که برای احتمال گذار داریم توجه می کنیم:

$$W(s_i = -1 \rightarrow s_i = 1) = \frac{1}{1 + e^{-2\beta h_i}} \quad (117)$$

و

$$W(s_i = 1 \rightarrow s_i = -1) = \frac{1}{1 + e^{2\beta h_i}} \quad (118)$$

یا به صورت فشرده خواهیم داشت:

$$W(s_i \rightarrow -s_i) = \frac{1}{1 + e^{2\beta h_i s_i}} \quad (119)$$

حال از خود سوال می کنیم که با چنین احتمالات گذاری احتمال اینکه شبکه عصبی در یک هیئت اسپینی معین باشد چقدر است؟ برای پاسخ به چنین سوالی مثل همه فرایندهای تصادفی دیگر به معادله اصلی توجه می کنیم: هیئت های نورونی را با  $C, C', \dots$  نشان می دهیم. معادله اصلی نیز به صورت زیر است:

$$P(C, t+1) = \sum_{C' \neq C} P(C', t) W(C' \rightarrow C) + P(C, t) W(C \rightarrow C) \quad (120)$$

منظور از  $W(C \rightarrow C)$  نیز احتمال ماندن در هیئت  $C$  است، یعنی

$$W(C \rightarrow C) = 1 - \sum_{C' \neq C} W(C \rightarrow C'). \quad (121)$$

از ترکیب (۱۲۰) و (۱۲۱) بدست می آوریم:

$$P(C, t+1) - P(C, t) = \sum_{C' \neq C} [P(C', t)W(C' \rightarrow C) - P(C, t)W(C \rightarrow C')] \quad (122)$$

در حالت تعادل که شرط توازن ظریف برقرار است داریم:

$$P(C', t)W(C' \rightarrow C) = P(C, t)W(C \rightarrow C') \quad \forall C, C'. \quad (123)$$

تحت این شرایط طرف چپ معادله (۱۲۲) نیز صفر شده و تعادل برقرار می شود یعنی هیچ چیز با زمان تغییر نمی کند. با فرض اینکه در هر

بازه زمانی تنها یکی از اسپین ها تغییر می کند، از رابطه (۱۲۳) بدست می آوریم:

$$\frac{P(s_1, s_2, \dots, -s_k, \dots, s_N)}{P(s_1, s_2, \dots, s_k, \dots, s_N)} = \frac{W(s_k \rightarrow -s_k)}{W(-s_k \rightarrow s_k)} = e^{-2\beta h_k s_k}. \quad (124)$$

با تکرار رابطه (۱۲۴) بدست می آوریم:

$$P(s_1, s_2, \dots, s_N) = \frac{1}{Z} e^{\beta(h_1 s_1 + h_2 s_2 + \dots + h_N s_N)} \quad (125)$$

که در آن  $Z$  یک ضریب ثابت است. با توجه به اینکه  $h_i = \sum_j \omega_{ij} s_j$  خواهیم داشت:

$$P(s_1, s_2, \dots, s_N) = \frac{1}{Z} e^{\frac{1}{2}\beta \sum_{i,j} \omega_{ij} s_i s_j} = \frac{1}{Z} e^{-\beta E[s_1, \dots, s_N]} \quad (126)$$

که در آن عبارت

$$E[s_1, s_2, \dots, s_N] = \frac{-1}{2} \sum_{i,j} \omega_{ij} s_i s_j \quad (127)$$

را می توان به عنوان انرژی یک هیئت نرونی در نظر گرفت. به این ترتیب یک ارتباط زیبا بین شبکه عصبی و یک سیستم اسپینی برقرار می شود.

تمام مشخصات شبکه اسپینی، این که چه تعداد طرح هایی را و به چه شکلی ضبط کرده و میزان همپوشانی آنها چقدر است، به صورت ضرایب

جفتیگی  $\omega_{ij}$  در سیستم اسپینی وارد می شود. به این ترتیب احتمال اینکه شبکه عصبی در یک هیئت خاص قرار بگیرد با عبارت زیر داده می

شود:

$$P(s_1, s_2, \dots, s_N) = \frac{1}{Z} e^{\frac{1}{2}\beta \sum_{i,j} \omega_{ij} s_i s_j} \quad (128)$$

داده می شود که در آن  $\beta$  پارامتری است که میزان تصادفی بودن رفتار نوروں ها را تعیین می کند. می توان این پارامتر را به صورت  $\beta = \frac{1}{T}$  نوشت

و  $T$  را دمای شبکه عصبی خواند، و البته به یاد داشت که این دما، با دمای متعارف متفاوت است. از آن جا که شبکه اسپینی رفتار احتمالی دارد

تنها می توان در حالت تعادل مقدار متوسط اسپین ها را حساب کرد. برای محاسبه مقدار متوسط نیز می بایست تابع پارش را حساب کرد. تابع پارش برابر است با:

$$Z = \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta}{2} \sum_{i,j} \omega_{i,j} s_i s_j} \quad (129)$$

برای آنکه مقدار متوسط  $\langle s_i \rangle$

را حساب کنیم، می بایست تابع پارش در حضور میدان را حساب کنیم. این میدان لزومی ندارد یک میدان واقعی باشد کافی است که به آن به عنوان یک عامل کمکی برای محاسبه نگاه کنیم. این تابع پارش عبارت است از:

$$Z[\beta, h] = \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta}{2} \sum_{i,j} \omega_{i,j} s_i s_j + \sum_i h_i s_i} \quad (130)$$

به این ترتیب همه چیز دوباره تبدیل شده است به محاسبه تابع پارش یک مدل که شبیه مدل آیزینگ است. تنها تفاوت، که البته تفاوت فوق العاده بزرگی است، این است که این مدل آیزینگ برهم کنش های پیچیده و غیرهمگنی بین همه اسپین ها دارد که از رابطه

$$\omega_{ij} = \frac{1}{N} \sum_{\mu=1}^N \sigma_i^{(\mu)} \sigma_j^{(\mu)}. \quad (131)$$

تعیین می شوند و دیگر از برهم کنش های همگن بین نزدیک ترین همسایه ها خبری نیست. بنابراین باید خود را برای محاسبه تابع پارش چنین مدل هایی آموزش دهیم. این کاری است که از بخش بعدی شروع به انجام آن می کنیم. نخست یک مثال ساده و ایده آل را حل می کنیم. این مدل ما را برای حل مدل واقعی تر آماده می کند، مدلی که به آن مدل هاپفیلد<sup>۳۰</sup> می گوئیم.

## ۱.۶ مدل آیزینگ با برهم کنش های بلند برد و یکنواخت

قبل از آنکه تابع پارش (۱۲۹) یا (۱۳۰) را حساب کنیم، یک تابع پارش ساده تر را حساب می کنیم. می دانیم که در یک شبکه عصبی که با قانون یادگیری هب<sup>۳۱</sup>  $p$  تا طرح را ضبط کرده است،  $\omega_{ij}$  ها برابرند با:

$$\omega_{ij} = \frac{1}{N} \sum_{\mu=1}^N \sigma_i^{(\mu)} \sigma_j^{(\mu)}. \quad (132)$$

مطالعه یک مدل اسپینی با چنین ضرایب جفتیدگی بسیار سخت است. بنابراین کار خود را پله پله به پیش می بریم. نخست یک مدل خیلی ساده تر را بررسی می کنیم که ربطی به شبکه های عصبی و حافظه و نظایر آن ندارد فقط مهارت های ما را در مطالعه سیستم های اسپینی بالا می برد و

<sup>۳۰</sup>Hopfield Model  
<sup>۳۱</sup>Hebb Learning Rule

ما را برای مطالعه شبکه اسپینی آماده می کند. در این مدل ساده تر همه شدت های سیناپسی با هم برابرند و در حقیقت داریم

$$\omega_{ij} = \frac{\omega}{N}.$$

این مدل چیزی نیست جز مدل آیزینگ با برهم کنش بلند برد، چون که در آن همه اسپین ها با شدت مسای با هم برهم کنش دارند. انرژی هر هیئت از اسپین ها در این مدل برابر است با:

$$E[s_1, s_2, \dots, s_N] = -\frac{\omega}{2N} \sum_{i,j} s_i s_j. \quad (133)$$

و تابع پارش برابر است با:

$$Z(\beta, h) = \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta\omega}{2N} \sum_{i,j} s_i s_j + h\beta \sum_i s_i} \quad (134)$$

به دلیل همگنی می دانیم که متوسط همه اسپین ها هم با هم برابر است یعنی

$$\langle s_1 \rangle = \langle s_2 \rangle = \dots \langle s_N \rangle.$$

با تعریف  $S = \sum_i s_i$  خواهیم داشت:

$$Z(\beta, h) = \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta\omega}{2N} S^2 + \beta h S}.$$

اگر چه عبارت انرژی ساده شده و تنها به اسپین کل یعنی  $S$  بستگی دارد، اما محاسبه تابع پارش هم چنان دشوار است چرا که می بایست روی تمام وضعیت تک تک اسپین ها جمع بزنیم. اگر در عبارت انرژی تنها جمله درجه یک یعنی  $S$  وجود می داشت انجام این کار خیلی ساده بود. برای این که از شر جمله مربعی یعنی  $S^2$  خلاص شویم از رابطه انتگرال گاوسی استفاده می کنیم. می دانیم

$$\int dx e^{-\frac{a}{2}x^2 + bx} = \sqrt{\frac{2\pi}{a}} e^{\frac{b^2}{2a}}. \quad (135)$$

قرار می دهیم:

$$b \rightarrow S, \quad a = \frac{N}{\beta\omega}, \quad x \rightarrow m \quad (136)$$

بنابراین

$$e^{\frac{\beta\omega}{2N} S^2} = \sqrt{\frac{N}{2\pi\beta\omega}} \int dm e^{-\frac{N}{2\beta\omega} m^2 + mS}. \quad (137)$$

حال می توان جمع روی  $s_i$  ها را انجام داد:

$$\sum_i e^{(\beta h + m)s_i} = 2 \cosh(\beta h + m) \quad (138)$$

و

$$\sum_{s_1, s_2, \dots, s_N} e^{(\beta h + m)S} = [2 \cosh(\beta h + m)]^N. \quad (139)$$

در نتیجه خواهیم داشت:

$$Z(\beta, h) = \left(\frac{N}{2\pi\beta\omega}\right)^{\frac{1}{2}} \int dm e^{-\frac{N}{2\beta\omega}m^2 + N \ln[2 \cosh(\beta h + m)]}. \quad (140)$$

حال می توان انتگرال فوق را با روش تقریب نقطه زینی محاسبه کرد. یادآوری می کنیم که این روش در حد  $N \rightarrow \infty$  مقدار دقیق انتگرال را بدست می دهد. داریم:

$$I = \int dm e^{-N\beta f(m)} \approx \int dm e^{-N\beta\{f(m_0) + \frac{1}{2}f''(m_0)(m-m_0)^2\}} \quad (141)$$

که در آن

$$f(m) = \frac{m^2}{2\beta^2\omega} - \frac{1}{\beta} \ln[2 \cosh(\beta h + m)] \quad (142)$$

و  $m_0$  نقطه ای است که در آن  $f(m)$  کمینه می شود. بنابراین

$$I \approx e^{-N\beta f(m_0)} \sqrt{\frac{2\pi}{N\beta f''(m_0)}}. \quad (143)$$

به این ترتیب نتیجه نهایی عبارت خواهد شد از:

$$Z(\beta, h) \approx \sqrt{\frac{1}{\beta\omega f''(m_0)}} e^{-N\beta f(m_0)}. \quad (144)$$

سه نکته را باید روشن کنیم: اینکه تعبیر فیزیکی  $f(m_0)$  چیست؟ این که مقدار  $m_0$  چقدر است؟ و این که تعبیر فیزیکی  $m_0$  چیست؟

نخست به نکته اول می پردازیم. می دانیم که رابطه انرژی آزاد هلمهولتز با تابع پارش به شکل زیر است:

$$F - = -\frac{1}{\beta} \ln Z. \quad (145)$$

اگر تابع انرژی آزاد را بر واحد اسپین بخواهیم و آن را با  $\tilde{f} = \frac{F}{N}$  نشان دهیم، داریم:

$$\tilde{f} \equiv \frac{F}{N} = -\frac{1}{N\beta} \ln Z \quad (146)$$

بنابراین

$$\tilde{f} = f(m_0). \quad (147)$$

دقت کنید که عبارت بالا را در حد ترمودینامیک یعنی  $N \rightarrow \infty$  حساب کرده ایم. بنابراین تابع انرژی آزاد چیزی نیست جز همان  $f(m_0)$ . یعنی هرگاه تابع  $f(m, h, T)$  را پیدا کنیم و کمینه آن را به عنوان تابعی از  $m$  حساب کنیم، انرژی آزاد به عنوان تابعی از  $h$  و  $T$  تعیین می شود. حال به نکته دوم می پردازیم. مقدار  $m_0$  برابر با چیست؟ می دانیم که  $m_0$  از رابطه  $\frac{d}{dm} f(m)|_{m=m_0} = 0$  بدست می آید. بنابراین:

$$f(m) = \frac{1}{2\beta^2\omega} m^2 - \frac{1}{\beta} \ln[2 \cosh(\beta h + m)] \quad (148)$$

و

$$f'(m) = \frac{m}{\beta^2\omega} - \frac{1}{\beta} \tanh(\beta h + m). \quad (149)$$

از ترکیب این دو رابطه بدست می آوریم:

$$m_0 = \beta\omega \tanh(\beta h + m_0). \quad (150)$$

رابطه بالا را تنها می توان به روش ترسیمی حل کرد. اما قبل از حل این معادله بهتر است معنای فیزیکی و شهودی  $m_0$  را بفهمیم. با استفاده از تابع پارش می توان مقدار متوسط  $\langle s_i \rangle$  یعنی مغناطش بر واحد اسپین را فهمید. داریم:

$$\sum_i \langle s_i \rangle = \frac{1}{Z} \frac{\partial Z}{\partial h\beta} = \frac{\partial \ln Z}{\partial h\beta} \quad (151)$$

و از آنجا

$$\langle s_i \rangle = \frac{1}{N} \frac{\partial}{\partial h\beta} \ln Z. \quad (152)$$



از طرفی می دانیم که در حد  $N \rightarrow \infty$

$$\frac{1}{N} \ln Z = -g(m_0) \quad (153)$$

بنابراین

$$\langle s_i \rangle = -\frac{\partial}{\partial h} g(m_0). \quad (154)$$

البته باید دقت کنیم که  $g$  را باید به شکل دقیق تر زیر نوشت:

$$g = g(m_0(h), h) \quad (155)$$

یعنی اینکه  $g$  یک بستگی صریح به  $h$  و یک بستگی ضمنی از طریق  $m_0$  دارد. بنابراین:

$$\langle s_i \rangle = \frac{\partial g}{\partial m} \Big|_{m_0} \frac{\partial m_0}{\partial h} + \frac{\partial g(m, h)}{\partial h} \Big|_{m=m_0}. \quad (156)$$

اما  $(\frac{\partial g}{\partial m})_{m_0}$  برابر با صفر است. بنابراین

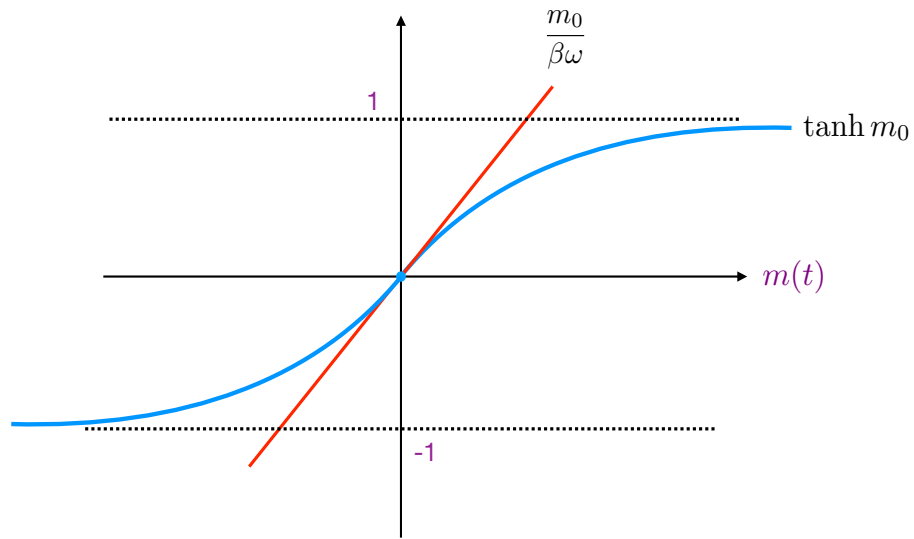
$$\langle s_i \rangle = \frac{\partial g}{\partial h}(m, h) \Big|_{m=m_0} = \tanh(\beta h + m_0) = \frac{m_0}{\beta \omega}. \quad (157)$$

پس در واقع  $m_0$  متناسب با مغناطش متوسط هر اسپین است. حال می توانیم مقدار  $m_0$  را به روش ترسیمی از رابطه (۱۷۵) پیدا کنیم. معمولاً علاقمند به یافتن میزان مغناطش خود به خود یعنی مغناطش در غیاب میدان مغناطیسی هستیم. بنابراین در رابطه (۱۷۵) می بایست قرار دهیم

$h = 0$ . پس هدف ما حل ترسیمی معادله

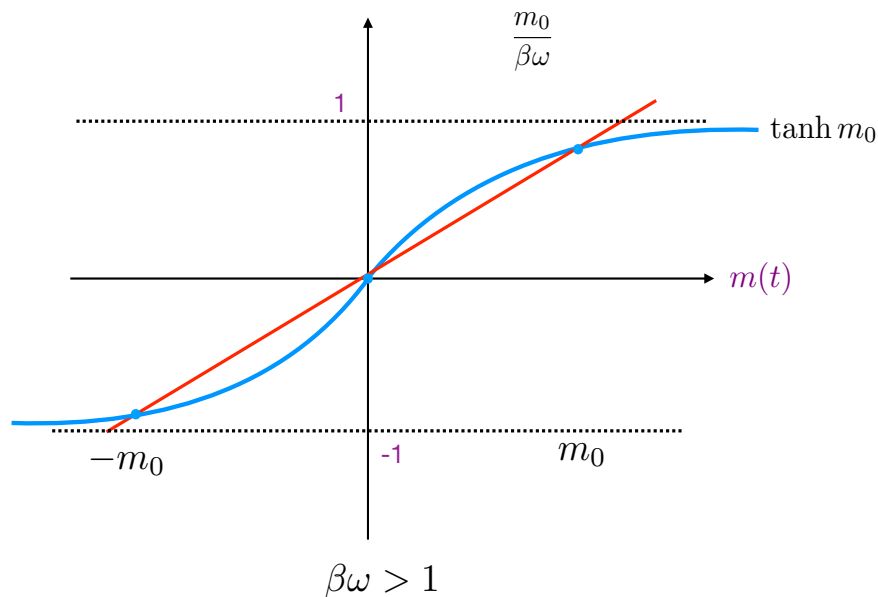
$$m_0 = \beta \omega \tanh m_0 \quad (158)$$

شکل های (۲۴) و (۲۵) در این باره گویا هستند.



$$\beta\omega < 1$$

شکل ۲۴: حل ترسیمی معادله (۱۷۵) وقتی که  $\beta\omega < 1$  است.



شکل ۲۵: حل ترسیمی معادله (۱۷۵) وقتی که  $\beta\omega \geq 1$  است.

بنابراین دمای بحرانی از رابطه  $\beta\omega = 1$  بدست می آید. یعنی دمای بحرانی عبارت است از  $T_c = \omega$ . برای  $T < T_c$  سه جواب  $m_0$ ,  $-m_0$ , و  $0$  و برای  $T > T_c$  تنها یک جواب  $0$  داریم. اما باید دقت کنیم که برای  $T < T_c$  جواب  $0$  ناپایدار است به این معنا که با کمی انحراف از مقدار صفر، جواب معادله به سمت یکی از دو مقدار  $m_0$  یا  $-m_0$  میل می کند. برای فهم بهتر این موضوع می توانیم به تابع انرژی آزاد هلمهولتز توجه کنیم. می دانیم که این تابع برابر است با  $\tilde{f}(h, T) = f(m_0, h, T)$ . در این جا به شرایطی علاقمندیم که میدان مغناطیسی حضور ندارد. یعنی شرایط  $h = 0$ . بنابراین انرژی آزاد تنها تابعی از دما خواهد بود. یعنی

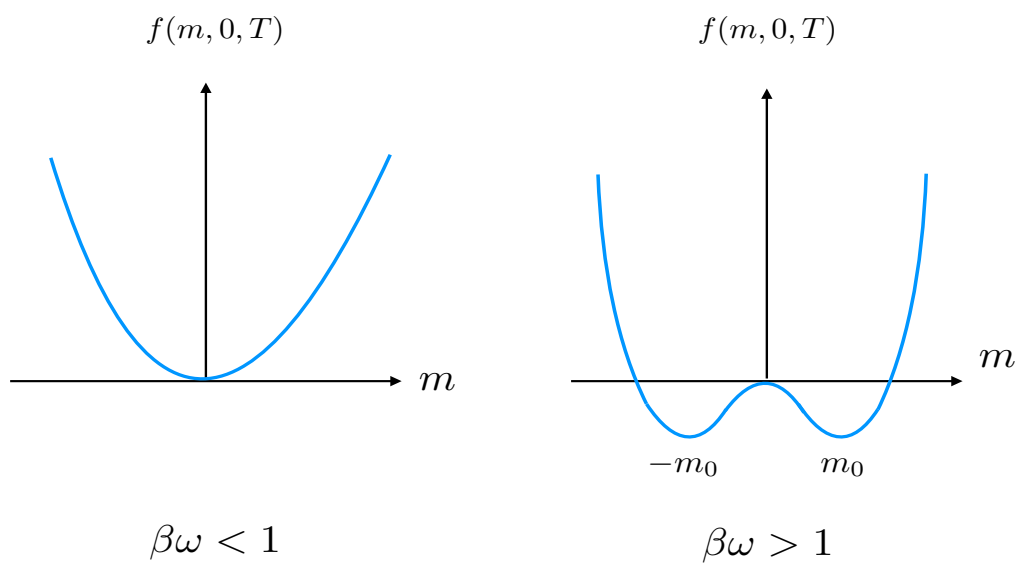
$$\tilde{f}(T) = f(m_0, 0, T). \quad (159)$$

هم چنین می دانیم که یک سیستم بسته در تعادل گرمایی در هیئت قرار خواهد گرفت که انرژی آزاد آن به کمترین مقدار خود برسد. بنابراین

تابع  $f(m, 0, T)$  را در نظر می‌گیریم. این تابع برابر است با:

$$f(m, 0, T) = \frac{m^2}{2\beta^2\omega} - \frac{1}{\beta} \ln[2 \cosh m]. \quad (160)$$

خواننده براحتی می‌تواند نشان دهد که شکل این تابع بسته به مقدار  $\beta\omega$  یکی از دو منحنی نشان داده شده در شکل (۲۶) است. به این ترتیب معلوم می‌شود که در  $T > T_c$  کمترین مقدار انرژی آزاد مربوط به  $m = 0$  و برای  $T < T_c$  کمترین مقدار انرژی آزاد مربوط به  $\pm m_0$  است.



شکل ۲۶: منحنی های تابع انرژی آزاد در بالا و پایین دمای بحرانی.

بدلیل تقارن نیز هرکدام از مقادیر  $m_0$  یا  $-m_0$  می‌توانند اختیار شوند و مغناطش خودبخودی سیستم می‌تواند هر کدام از این دو مقدار باشد. حال به اندازه کافی تجربه اندوخته ایم که به مطالعه مدل واقعی تر هاپفیلد پردازیم. طبیعی است که می‌خواهیم این مدل را در حد ترمودینامیک یعنی وقتی که تعداد نوروں ها  $N$  به سمت بی نهایت میل می‌کند حل کنیم. اما تنها این حد نیست که رفتار مدل هاپفیلد را تعیین می‌کند، بلکه در باره حد پارامتر  $p$  نیز باید تصمیم بگیریم چرا که این پارامتر نیز در حد ترمودینامیک اهمیت دارد. به همین منظور دو رژیم جداگانه را در نظر می‌گیریم.

## ۷ مدل هاپفیلد در حد ترمودینامیک وقتی که تعداد طرح ها محدود باقی می ماند.

مطالعه مدل آیزینگ در بخش گذشته ما را برای مطالعه مدل هاپفیلد در این بخش آماده کرده است. می توانیم این مدل را درست مثل یک شبکه اسپینی با ثابت های جفتدگی  $\omega_{ij} = \frac{1}{N} \sum_{\mu=1}^p \sigma_i^{(\mu)} \sigma_j^{(\mu)}$  در نظر بگیریم و بسیاری از مراحل بخش قبلی را تکرار کنیم. هدف ما آن است که کمینه تابع آزاد را برای چنین شبکه ای پیدا کنیم. در واقع تابع انرژی برای هر هیئت این شبکه از رابطه

$$E[s] = -\frac{1}{2} \sum_{i,j} \omega_{ij} s_i s_j \quad (161)$$

تعیین می شود و می دانیم که در دینامیک تعینی <sup>۳۲</sup> برای این شبکه عصبی تابع انرژی بالا کمینه می شود. این امر به این معناست که هرگاه از یک طرح اولیه شروع کنیم، دینامیک شبکه عصبی ما را به جایی می برد که در یک می نی مم موضعی تابع  $E$  قرار دارد. ولی وقتی که دینامیک شبکه تعینی نیست بلکه یک دینامیک تصادفی در دمای  $T$  است، آنگاه طرح های تعادلی طرح هایی هستند که کمینه تابع انرژی آزاد  $F = -kT \ln Z$  هستند. در دمای صفر و با توجه به رابطه ترمودینامیکی  $F = E - TS$  انرژی آزاد با انرژی  $E$  یکی می شود. ولی حتی در دمای صفر نیز پیدا کردن کمینه های تابع  $F$  با استفاده از روش های مکانیک آماری بسیار آسان تر از پیدا کردن کمینه های  $E$  است. بعد از این مقدمات به محاسبه تابع پارش می پردازیم:

$$\begin{aligned} Z &= \sum_{s_1, \dots, s_N} e^{-\beta E[s_1, s_2, \dots, s_N]} = \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta}{2} \sum_{i,j} \omega_{ij} s_i s_j} \\ &= \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta}{2N} \sum_{i,j,\mu} \sigma_i^{(\mu)} \sigma_j^{(\mu)} s_i s_j}. \end{aligned} \quad (162)$$

حال ضرب داخلی زیر را تعریف می کنیم:

$$\langle \sigma^{(\mu)}, S \rangle = \sum_{i=1}^N \sigma_i^{(\mu)} s_i. \quad (163)$$

این عبارت در واقع میزان همپوشانی طرح موجود در شبکه یعنی  $(s_1, s_2, \dots, s_N)$  را با طرح ذخیره شده  $\sigma^{(\mu)} = (\sigma_1^{(\mu)}, \sigma_2^{(\mu)}, \dots, \sigma_N^{(\mu)})$  بدست می دهد. مقدار این همپوشانی می تواند از  $-N$  تا  $N$  تغییر کند. در نتیجه تابع پارش به شکل زیر در می آید:

$$Z(\beta) = \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta}{2N} \sum_{\mu=1}^p \langle \sigma^{(\mu)}, S \rangle^2}. \quad (164)$$

اگر بخواهیم تابع پارش را در حضور میدان مغناطیسی حساب کنیم، باید به کمیت زیر توجه کنیم که در آن  $\mathbf{h} = (h_1, h_2, \dots, h_N)$  یک میدان

<sup>۳۲</sup>Deterministic

مغناطیسی ناهمگن است:

$$\begin{aligned} Z(\beta, \mathbf{h}) &= \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta}{2N} \sum_{\mu=1}^p \langle \sigma^{(\mu)}, S \rangle^2 + \beta \sum_i h_i s_i} \\ &= \sum_{s_1, s_2, \dots, s_N} e^{\frac{\beta}{2N} \sum_{\mu=1}^p \langle \sigma^{(\mu)}, S \rangle^2 + \beta \langle \mathbf{h}, S \rangle}. \end{aligned} \quad (165)$$

بازهم از رابطه انتگرال گاوسی استفاده می کنیم:

$$e^{\frac{\beta}{2N} \langle \sigma^{(\mu)}, S \rangle^2} = C \int dm^\mu e^{-\frac{N}{2\beta} (m^\mu)^2 + m^\mu \langle \sigma^{(\mu)}, S \rangle} \quad (166)$$

که در آن  $m_\mu$  متغیر انتگرال گیری کمکی است. در نتیجه

$$e^{\frac{\beta}{2N} \sum_{\mu=1}^p \langle \sigma^{(\mu)}, S \rangle^2} = C \int d\mathbf{m} e^{-\frac{N}{2\beta} \mathbf{m} \cdot \mathbf{m} + \sum_{\mu=1}^p m_\mu \langle \sigma^{(\mu)}, S \rangle} \quad (167)$$

در اینجا  $C$  یک مقدار ثابت عددی است که نقشی در نتایج نهایی ندارد و همواره موقع محاسبه متوسط ها از صورت و مخرج حذف می شود.

تابع پارش برابر می شود با:

$$Z(\beta, \mathbf{h}) = C \int d\mathbf{m} \sum_{s_1, s_2, \dots, s_N} e^{-\frac{N}{2\beta} \sum_{\mu=1}^p (m^\mu)^2 + \sum_{\mu=1}^p m^\mu \langle \sigma^{(\mu)}, S \rangle + \beta \langle \mathbf{h}, S \rangle} \quad (168)$$

حال جمع روی  $s_i$  ها را انجام می دهیم. داریم:

$$\begin{aligned} \sum_{s_1, s_2, \dots, s_N} e^{\sum_{i=1}^N \sum_{\mu=1}^p m^\mu \sigma_i^{(\mu)} s_i + \beta \sum_{i=1}^N h_i s_i} &= \prod_{i=1}^N \sum_{s_i=-1, 1} e^{[(\sum_{\mu=1}^p m^\mu \sigma_i^{(\mu)}) + \beta h_i] s_i} \\ &= \prod_{i=1}^N 2 \cos(\beta h_i + \sum_{\mu=1}^p m^\mu \sigma_i^{(\mu)}). \end{aligned} \quad (169)$$

با تعریف بردار

$$\mathbf{m} = (m^1, m^2, \dots, m^p)$$

و ضرب داخلی

$$\mathbf{m} \cdot \boldsymbol{\sigma}_i := \sum_{\mu=1}^p m^\mu \sigma_i^{(\mu)}$$

تابع پارش برابر می شود با:

$$Z(\beta, \mathbf{h}) = C \int d\mathbf{m} e^{-\frac{N}{2\beta} \mathbf{m} \cdot \mathbf{m}} \prod_{i=1}^N 2 \cos(\beta h_i + \mathbf{m} \cdot \boldsymbol{\sigma}_i)$$

$$= C \int d\mathbf{m} e^{-\frac{N}{2\beta} \mathbf{m} \cdot \mathbf{m} + \sum_{i=1}^N \ln [2 \cos (\beta h_i + \mathbf{m} \cdot \boldsymbol{\sigma}_i)]}. \quad (170)$$

می توان تابع پارش را به صورت زیر نوشت:

$$\begin{aligned} Z(\beta, \mathbf{h}) &= C \int d\mathbf{m} e^{-\frac{N}{\beta} \left[ \frac{1}{2} \mathbf{m} \cdot \mathbf{m} - \frac{\beta}{N} \sum_{i=1}^N \ln \{2 \cos (\beta h_i + \mathbf{m} \cdot \boldsymbol{\sigma}_i)\} \right]} \\ &= C \int d\mathbf{m} e^{-\frac{N}{\beta} f(\mathbf{m})} \end{aligned} \quad (171)$$

که در آن

$$f(\mathbf{m}) := \frac{1}{2} \mathbf{m} \cdot \mathbf{m} - \frac{\beta}{N} \sum_{i=1}^N \ln \{2 \cos (\beta h_i + \mathbf{m} \cdot \boldsymbol{\sigma}_i^{(\mu)})\} \quad (172)$$

یک کمیت نافزونور است و به  $N$  بستگی ندارد. بنابراین در حد  $N$  های بزرگ، و با استفاده از تقریب نقطه زینی، تابع پارش برابر می شود با

$$Z \approx C e^{-\frac{N}{\beta} f(\mathbf{m}_0)} \quad (173)$$

که در آن  $\mathbf{m}_0$  برداری است که تابع  $f(\mathbf{m}_0)$  را کمینه می کند. به عبارت دیگر:

$$\left. \frac{\partial f(\mathbf{m})}{\partial m^\mu} \right|_{\mathbf{m}_0} = 0, \quad (174)$$

و یا

$$m_0^\mu = \frac{\beta}{N} \sum_{i=1}^N \sigma_i^{(\mu)} \tanh(\beta h_i + \mathbf{m}_0 \cdot \boldsymbol{\sigma}_i). \quad (175)$$

در حد  $N$  های بزرگ و با استفاده از تقریب نقطه زینی خواهیم داشت:

$$Z(\beta, \mathbf{h}) = e^{-\frac{N}{2\beta} \mathbf{m}_0 \cdot \mathbf{m}_0 + \sum_{i=1}^N \ln [2 \cos (\beta h_i + \mathbf{m}_0 \cdot \boldsymbol{\sigma}_i)]} \quad (176)$$

که در آن  $\mathbf{m}_0$  از رابطه (175) تعیین می شود. به این ترتیب تابع پارش و در نتیجه بقیه کمیت های فیزیکی را حساب کرده ایم چرا که رابطه اخیر تابع پارش را بر حسب  $\mathbf{m}_0$  تعیین می کند و این کمیت نیز توسط رابطه (175) معلوم می شود. البته باید توجه کنیم که کمیت  $\mathbf{h} = (h_1, h_2, \dots, h_N)$  را که شبیه به میدان مغناطیسی است تا انتها نگاه می داریم و پس از محاسبه مشتق ها نسبت به آن و بدست آوردن متوسط های مختلف آن را برابر با صفر قرار می دهیم. اگر چه  $\mathbf{m}$  را به صورت یک کمیت کمکی وارد محاسبات خود کردیم اما می توانیم نشان

دهیم که این کمیت یک معنای فیزیکی روشن دارد. برای اینکه معنای آن را دریابیم.

نخست توجه می کنیم که ما به متوسط مقدار یک اسپین یا نورون بخصوص علاقه نداریم. بلکه می خواهیم ببینیم که طرحی که در یک حالت تعادل می بینیم به طور متوسط چقدر شبیه به یک طرح ذخیره شده است. یعنی علاقمند به کمیت زیر هستیم:

$$q^\mu := \frac{1}{N} \sum_{i=1}^N \sigma_i^{(\mu)} \langle s_i \rangle. \quad (177)$$

این کمیت مقدار همپوشانی طرح  $\langle s_i \rangle$  را با طرح  $\sigma^\mu$  بدست می دهد و مقدار آن بین  $-1$  و  $1$  تغییر می کند. از رابطه (۱۶۵) داریم:

$$q^\mu = \frac{1}{N} \sum_{i=1}^N \sigma_i^\mu \frac{\partial}{\partial \beta h_i} \ln Z. \quad (178)$$

با توجه به اینکه  $Z = C e^{-N\beta f(\mathbf{m}_0, \mathbf{h})}$  بدست می آوریم:

$$q^\mu = \frac{1}{N} \sum_{i=1}^N \sigma_i^\mu \langle s_i \rangle = - \sum_{i=1}^N \sigma_i^\mu \frac{\partial}{\partial h_i} f(\mathbf{m}_0, \mathbf{h}) \quad (179)$$

دقت کنید که  $\mathbf{m}_0$  از طریق معادله (۱۷۵) خود تابع  $h$  است. یعنی باید بنویسیم:

$$q^\mu = - \sum_{i=1}^N \sigma_i^\mu \left[ \frac{\partial f(\mathbf{m}_0(\mathbf{h}), \mathbf{h})}{\partial \mathbf{m}_0} \frac{\partial \mathbf{m}_0}{\partial h_i} + \frac{\partial f(\mathbf{m}_0(\mathbf{h}), \mathbf{h})}{\partial h_i} \right] \quad (180)$$

جمله اول بنا بر تعریف  $\mathbf{m}_0$  مساوی صفر است و در نتیجه:

$$q^\mu = - \sum_{i=1}^N \sigma_i^\mu \frac{\partial f(m_0(h), h)}{\partial h_i} \quad (181)$$

با توجه به رابطه

$$f(\mathbf{m}, h) = \frac{1}{2\beta^2} \mathbf{m} \cdot \mathbf{m} - \frac{1}{N\beta} \sum_{i=1}^N \ln \left[ 2 \cosh(\beta h_i + \mathbf{m}_0 \cdot \sigma_i) \right] \quad (182)$$

بدست می آوریم:

$$q^\mu = - \sum_{i=1}^N \sigma_i^\mu \left( \frac{1}{N\beta} \right) \beta \tanh(\beta h_i + \mathbf{m}_0 \cdot \sigma_i)$$



$$= \frac{1}{N} \sum_{i=1}^N \sigma_i^\mu \tanh(\mathbf{m}_0 \cdot \sigma_i) \quad (183)$$

و یا

$$q^\mu = \langle\langle \sigma^\mu \tanh(\mathbf{m}_0 \cdot \sigma) \rangle\rangle = \frac{1}{\beta} m_0^\mu. \quad (184)$$

در این جا از یک نماد جدید یعنی  $\langle\langle X \rangle\rangle$  استفاده کرده ایم. این نماد جایگزینی است برای عبارت زیر:

$$\langle\langle X \rangle\rangle := \frac{1}{N} \sum_i \langle X_i \rangle, \quad (185)$$

یعنی روی کمیت  $X_i$  هم متوسط آماری به دلیل افت و خیزهای تصادفی و به اصطلاح گرمایی ناشی از  $\beta$  گرفته ایم و هم روی همه نوروں ها متوسط گرفته ایم. بنابراین  $m^\mu$  متناسب است با همپوشانی طرح  $\langle s_i \rangle$  طرح ضبط شده ی  $\sigma^\mu$ .

حال می پردازیم به حل معادلات:

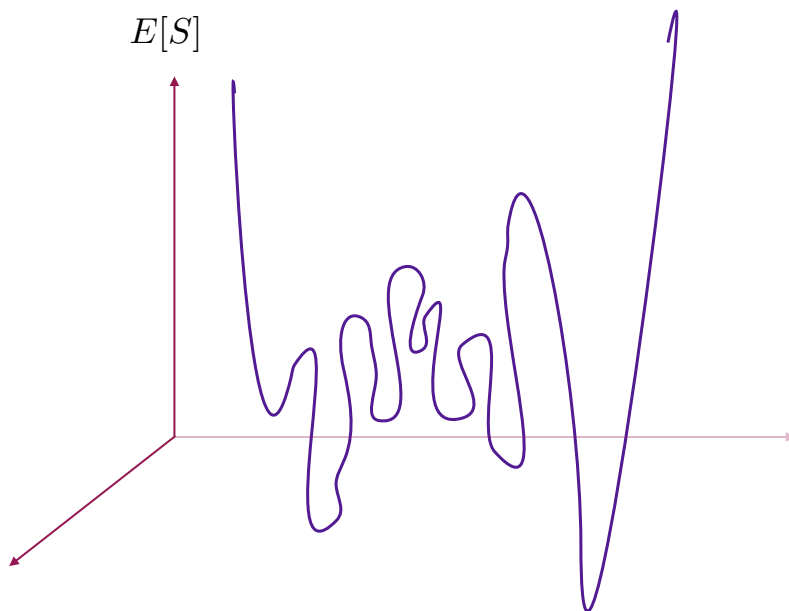
$$m_0^\mu = \beta \langle\langle \sigma^{(\mu)} \tanh(\mathbf{m}_0 \cdot \sigma) \rangle\rangle \quad (186)$$

همانطور که گفتیم، این معادلات مجموعه ای  $p$  تایی از معادلات غیرخطی کوپل شده برای متغیرهای  $m^\mu$  هستند. این معادلات نقاط کمینه تابع انرژی آزاد  $f(\mathbf{m}(T), T)$  را نشان می دهند. دقت کنید که  $\mathbf{h}$  را برابر با صفر قرار داده ایم. تعداد این کمینه ها بسیار زیاد است، شکل (۲۷). قبل از آنکه به روش های پیدا کردن این نقاط کمینه پردازیم به یک خاصیت مهم آنها بدون اثبات اشاره می کنیم. نشان داده شده است که این نقطه کمینه دارای خاصیتی به نام تودرتویی<sup>۳۳</sup> هستند که به ساختار خودشبهه و شاخه شاخه ی تابع  $f$  اشاره دارد. به عبارت دیگر اگر یکی از دره های تابع  $f$  را بزرگ کنیم، در درون آن تعداد زیادی دره دیگر پیدا می کنیم و این خاصیت درحد ترمودینامیک بارها و بارها تکرار می شود. حال می پردازیم به حل معادلات:

$$m^\mu = \beta \langle\langle \sigma^\mu \tanh(\mathbf{m} \cdot \sigma) \rangle\rangle. \quad (187)$$

همانطور که گفتیم، این معادلات مجموعه ای  $p$  تایی از معادلات غیرخطی کوپل شده برای متغیرهای  $m^\mu$  هستند. واضح است که هیچ گونه امکانی برای حل دقیق و تحلیلی این معادلات بدون فرض های ساده کننده وجود ندارد. در نخستین تقریب بیابید فرض کنیم که همه طرح ها با

<sup>۳۳</sup>Inermittency



شکل ۲۷:

وضوح یکسان به خاطر آورده می شوند. یعنی

$$\mathbf{m} = (m, m, \dots, m). \quad (188)$$

در این صورت معادله ( ) تبدیل می شود به:

$$m = \beta \langle \langle \sigma^\mu \tanh m \left( \sum_{\nu=1}^p \sigma^\nu \right) \rangle \rangle \quad (189)$$

با فرض  $\langle \langle \sigma^\mu \sigma^\nu \rangle \rangle = \delta^{\mu,\nu}$  خواهیم داشت:

$$m = \beta \tanh m. \quad (190)$$

قبلا دیدیم که اگر  $\beta < 1$  باشد، جواب  $m = 0$  است، و اگر  $\beta < 1$  باشد، دو جواب  $\pm m_0$  وجود خواهد داشت. حال فرض کنید که  $k$  تا طرح

با وضوح یکسان به خاطر آورده می شوند و  $p - k$  تا طرح اصلا به یاد آورده نمی شوند. بدون نقض کلیت می توان  $\beta f m$  را به صورت زیر نوشت:

$$\mathbf{m} = (\underbrace{m, m, \dots, m}_k, \underbrace{0, 0, \dots, 0}_{p-k}) \quad (191)$$

در این صورت داریم:

$$\begin{aligned} \mu = 1 &\rightarrow m = \beta \langle \langle \sigma^1 \tanh m(\sigma^1 + \sigma^2 + \dots + \sigma^k) \rangle \rangle \\ \mu = 2 &\rightarrow m = \beta \langle \langle \sigma^2 \tanh m(\sigma^1 + \sigma^2 + \dots + \sigma^k) \rangle \rangle \\ &\dots\dots\dots \\ \mu = k &\rightarrow m = \beta \langle \langle \sigma^k \tanh m(\sigma^1 + \sigma^2 + \dots + \sigma^k) \rangle \rangle. \end{aligned} \quad (192)$$

با جمع کردن همه این تساوی ها به رابطه زیر می رسم:

$$km = \beta \langle \langle (\sigma^1 + \sigma^2 + \dots + \sigma^k) \tanh m(\sigma^1 + \sigma^2 + \dots + \sigma^k) \rangle \rangle. \quad (193)$$

■ مثال:  $k = 2$ .

داریم:

$$2m = \beta \langle \langle (\sigma^1 + \sigma^2) \tanh m(\sigma^1 + \sigma^2) \rangle \rangle. \quad (194)$$

با توجه به این که فرض کرده ایم متغیرهای  $\sigma^1$  و  $\sigma^2$  متغیرهای مستقل با توزیع یکنواخت هستند که مقادیر یک و منهای یک را اختیار می کنند، معادله بالا تبدیل می شود به:

$$2m = \beta \tanh 2m. \quad (195)$$

که حل آن بسته به مقدار  $\beta$  دو مقدار مختلف برای  $m$  بدست می دهد. بنابراین تعداد  $\binom{n}{2}$  تا جواب برای معادلات (۱۸۷) پیدا کردیم که در آن  $\mathbf{m}$  دو مولفه غیر صفر دارد. عدد  $\binom{n}{2}$  ناشی از این است که این دو مولفه غیر صفر می توانند در مکان های مختلف باشند.

■ مثال ۳:  $k = 3$  از معادله (۱۹۳) بدست می آوریم:

$$3m = \beta \langle \langle (\sigma^1 + \sigma^2 + \sigma^3) \tanh m(\sigma^1 + \sigma^2 + \sigma^3) \rangle \rangle. \quad (196)$$

■ تمرین: ثابت کنید که این معادله منجر به معادله زیر می شود:

$$m = \frac{\beta}{4}(\tanh m + \tanh 3m). \quad (197)$$

با حل کردن این معادله (به صورت ترسیمی) می توان مقادیرهای  $m$  را بدست آورد. به این ترتیب تعداد  $\binom{n}{3}$  جواب از معادله (۱۸۷) بدست می آوریم که در آنها  $m$  دارای سه مولفه غیرصفر است. یعنی در این کمینه ها، سه تا از طرح ها با وضوح  $m$  به یاد آورده می شوند و  $p - 3$  تا از طرح ها به یاد آورده نمی شوند. به همین ترتیب می توان علی الاصول تعداد بیشتری از جواب ها را به دست آورد. ولی واضح است که این جواب هایی که از معادله (۱۹۳) بدست می آیند همه جواب های معادله (۱۸۷) نیستند. و بالاخره باید تاکید کنیم که فرض ما در باره این که

$$\langle \sigma^\mu \rangle = 0, \quad \langle \sigma^\mu \sigma^\nu \rangle = 0,$$

یک فرض بسیار ساده کننده است. در این فرض هیچ نوع اطلاعاتی در مورد هم بستگی طرح ها با یکدیگر وجود ندارد. در واقع فرض کرده ایم که هیچ نوع همبستگی بین طرح ها وجود ندارد. در یک مطالعه دقیق تر می بایست همبستگی طرح ها را با هم در نظر گرفت.

ادامه دارد.....