دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی
مهندسی کامپیوتر

پیچیدگی محاسباتی تنظیم دقیق آستانه‌ها در شبکه‌های عصبی

نگارش

علی مهدوی فر

استاد راهنما

دکتر حمید ضرابی‌زاده

بهمن ۱۴۰۳

به نام خدا

دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی

این پایان‌نامه به عنوان تحقق بخشی از شرایط دریافت درجه کارشناسی است.

عنوان: پیچیدگی محاسباتی تنظیم دقیق آستانه‌ها در شبکه‌های عصبی

نگارش: علی مهدوی فر

کمیته ممتحنین

استاد راهنما: دکتر حمید ضرابی‌زاده امضاء:

استاد مشاور: استاد مشاور امضاء:

استاد مدعو: استاد ممتحن امضاء:

تاریخ:

سپاس

از استاد بزرگوارم، دکتر حمید ضرابی‌زاده، که با کمک‌ها و راهنمایی‌های بی‌دریغشان، مرا در به سرانجام رساندن این پایان‌نامه یاری داده‌اند، تشکر و قدردانی می‌کنم. همچنین از همکار عزیزم، آقای حمیدرضا کامکاری، که با کمک‌های خود در مراحل اولیه‌ی کار به‌ویژه در بخش ارائه‌ی الگوریتم برای مسئله در به‌ثمررسیدن این پژوهش سهیم بوده‌اند، صمیمانه سپاسگزارم.

چکیده

مطالعه‌ی پیچیدگی محاسباتی مسائل یادگیری در حوزه‌ی الگوریتم‌های هوش مصنوعی، از اهمیت نظری در آن حوزه برخوردار است. یک مسئله‌ی مورد پژوهش در این زمینه، تنظیم دقیق شبکه‌های عصبی است. مطالعات پیشین نشان می‌دهد که تنظیم دقیق یا همان بهینه‌سازی زیرمجموعه‌ای از وزن‌های شبکه‌ی عصبی روی دادگان جدید، به‌ویژه بهینه‌سازی وزن‌های آستانه یا بایاس، می‌تواند کارایی نزدیک به تنظیم دقیق تمامی وزن‌های شبکه را داشته باشد. باتوجه به سادگی این حالت خاص از مسئله، بررسی پیچیدگی زمانی و نشان‌دادن سختی محاسباتی آن می‌تواند شهودی نسبت به سختی مسئله‌ی کلی یادگیری به ما ارائه دهد.

در این پژوهش، یک شبکه‌ی عصبی برای دسته‌بندی دودویی با یک لایه‌ی نهان و تابع فعال‌ساز پله را در نظر می‌گیریم و با کاهش مسئله‌ی یافتن خوشه در گراف به مسئله‌ی تنظیم دقیق آستانه‌ها در این شبکه‌ی عصبی، نشان می‌دهیم که بنابر فرض ETH این مسئله متعلق به دسته‌ی مسائل FPT نیست به این معنا که اگر پارامتر ابعاد شبکه‌ی عصبی را ثابت فرض کنیم، برای این مسئله هیچ الگوریتم چندجمله‌ای برحسب اندازه‌ی دادگان وجود نخواهد داشت. نتایج این پژوهش نه تنها سختی محاسباتی این مسئله‌ی خاص را آشکار می‌کند بلکه با ارائه‌ی شهود نسبت به پیچیدگی ذاتی یادگیری و تنظیم دقیق شبکه‌های عصبی، راه را برای بررسی مسائل پیچیده‌تر در این حوزه هموار می‌سازد.

کلیدواژه‌ها: تنظیم دقیق، مقادیر آستانه، پیچیدگی محاسباتی، پیچیدگی پارامتری، تابع فعال‌ساز پله

فهرست مطالب

فهرست شکل‌ها

فصل ۱

مقدمه

حوزه‌ی هوش مصنوعی در دهه‌های اخیر رشد چشمگیری داشته است و شبکه‌های عصبی به عنوان یکی از ارکان اصلی این تحول شناخته می‌شوند. در میان جنبه‌های مختلف شبکه‌های عصبی، تنظیم دقیق[۱] مقادیر وزن‌های شبکه به یکی از زمینه‌های تحقیقاتی مهم تبدیل شده است. تحقیقات اخیر نشان می‌دهد که بهینه‌سازی مقادیر آستانه[۲] به تنهایی می‌تواند نتایج کارآمد و مؤثری ارائه دهد، به طوری که سربار محاسباتی کاهش یافته و عملکرد مدل همچنان در سطح بالایی باقی بماند. این پژوهش به بررسی پیچیدگی محاسباتی تنظیم مقادیر آستانه در شبکه‌های عصبی می‌پردازد، به طور خاص در شبکه‌هایی که از توابع فعال‌سازی پله استفاده می‌کنند.

این تحقیق به مسئله‌ی تنظیم دقیق شبکه‌های عصبی از دیدگاهی الگوریتمی پرداخته و آن را به عنوان یک مسئله بهینه‌سازی با قیود قطعی در نظر می‌گیرد. این رویکرد چالش‌های محاسباتی ذاتی را هدف قرار داده و شهودی درباره‌ی طبیعت مسئله ارائه می‌دهد. از طریق مدل‌سازی نظری و اثبات پیچیدگی محاسباتی، این پژوهش به درک عمیق‌تری از تعامل میان مسائل بهینه‌سازی و مسئله‌ی یادگیری شبکه‌های عصبی کمک می‌کند.

۱-۱ تعریف مسئله

تنظیم دقیق در شبکه‌های عصبی شامل تطبیق یک مدل از پیش آموزش‌دیده شده برای عملکرد بهتر روی وظایف یا داده‌های خاص است. تنظیم دقیق مقادیر آستانه به شبکه اجازه می‌دهد تا بدون نیاز به آموزش

[۱] fine-tuning
[۲] threshold terms

کامل مجدد کل شبکه، به سرعت با داده‌های جدید سازگار شود.

این پژوهش بر روی بررسی پیچیدگی محاسباتی تنظیم مقادیر آستانه در شبکه‌های عصبی با توابع فعال‌سازی آستانه و نیز با یک لایه‌ی پنهان متمرکز است. هدف این است که فرآیند تنظیم دقیق از دیدگاه الگوریتمی تحلیل شده و چالش‌های نظری مرتبط با این فرآیند شناسایی شود. به طور خاص، مسئله به عنوان یک وظیفه بهینه‌سازی فرمول‌بندی می‌شود که هدف آن یافتن مقادیر آستانه‌ی بهینه برای حداکثر کردن دقت شبکه بر روی داده‌های داده شده است. این مطالعه ضمن اثبات پیچیدگی محاسباتی این مسئله، راهکاری الگوریتمی برای آن ارائه می‌دهد که برگرفته از یافتن ارتباط بین این مسئله و دیگر مسائل الگوریتمی شناخته‌شده در حوزه‌ی نظریه‌ی گراف و نیز هندسه‌ی محاسباتی است.

۱-۲ اهمیت موضوع

شبکه‌های عصبی به عنوان یکی از ابزارهای ضروری در حوزه هوش مصنوعی شناخته شده‌اند و وظایفی از جمله تشخیص تصویر تا پردازش زبان طبیعی را متحول کرده‌اند. فرایند تنظیم دقیق در این میان اهمیت ویژه‌ای در بهبود عملکرد مدل‌های از پیش آموزش‌دیده شده و سازگاری آن‌ها با مجموعه داده‌های جدید دارد. این فرایند، نه تنها دقت مدل را در وظایف خاص افزایش می‌دهد، بلکه امکان استفاده مؤثرتر از منابع محاسباتی و کاهش هزینه‌های یادگیری شبکه را فراهم می‌سازد.

مقادیر آستانه اغلب در مقایسه با وزن‌ها ثانویه تلقی می‌شوند. با این حال مطالعات اخیر نشان داده‌اند که تنظیم دقیق مقادیر آستانه می‌تواند بهبودهای قابل توجهی ارائه دهد، در حالی که زمان یادگیری و نیاز به حافظه به طور چشمگیری کاهش می‌یابد. این جنبه به ویژه برای محیط‌های با منابع محدود، مانند دستگاه‌های لبه[۳]، که منابع محاسباتی و حافظه‌ی محدودی دارند، ارزشمند است. با تمرکز بر بهینه‌سازی آستانه‌ها، می‌توان سازگاری مدل‌های از پیش آموزش‌دیده شده را با مجموعه داده‌های جدید بهبود بخشید و به روزرسانی‌های کارآمد مدل را با حداقل هزینه محاسباتی تضمین کرد.

پیچیدگی عملی یادگیری و نیز تنظیم دقیق شبکه‌های عصبی انگیزه‌بخش یک بررسی با رویکردی الگوریتمی است که این فرآیند را به عنوان یک مسئله بهینه‌سازی قطعی در نظر بگیرد. در نظر گرفتن چالش یادگیری شبکه‌های عصبی به این صورت امکان توسعه الگوریتم‌های ساختاریافته را فراهم می‌کند.

این پژوهش نه تنها به پیامدهای نظری پیچیدگی محاسباتی تنظیم دقیق آستانه‌ها می‌پردازد، بلکه بر اهمیت راه‌حل‌های الگوریتمی در تحلیل یادگیری شبکه‌های عصبی تأکید می‌کند. با ترکیب این دیدگاه‌ها، این تحقیق تلاش دارد تا درک نظری و عملی شبکه‌های عصبی را ارتقا دهد.

[۳] edge devices

۱-۳ ادبیات موضوع

کارهای پیشین مرتبط با این پژوهش در زمینه‌ی تنظیم دقیق شبکه‌های عصبی را می‌توان از دو منظر مهم بررسی کرد: انتخاب هوشمندانه‌ی پارامترهای قابل تنظیم مانند مقادیر آستانه، و تحلیل پیچیدگی محاسباتی مسائل یادگیری شبکه‌های عصبی به عنوان یک مسئله بهینه‌سازی.

در حوزه یادگیری ماشین، تنظیم دقیق مقادیر آستانه در شبکه‌های عصبی یکی از موضوعات مورد توجه بوده است. مطالعات اخیر نشان داده‌اند که تمرکز بر تنظیم تنها مقادیر آستانه، بدون تغییر دیگر وزن‌ها، می‌تواند منجر به دقت‌های قابل‌قبول با کاهش قابل توجه سربار محاسباتی و حافظه شود. به عنوان مثال، روش BitFit [۱] یک چارچوب جدید برای تنظیم دقیق آستانه‌ها ارائه داده است که با انتخاب محدود پارامترها، امکان انطباق مدل‌ها با مجموعه داده‌های جدید را فراهم کرده و نیازهای محاسباتی را کاهش داده است. روش دیگری به نام DiffFit [۲] از مقادیر آستانه در لایه‌های خاص مدل‌های بزرگ برای افزایش سرعت یادگیری و کاهش هزینه‌های ذخیره‌سازی استفاده کرده است. این پژوهش‌ها اهمیت انتخاب محدود پارامترها را در تنظیم دقیق به وضوح نشان داده‌اند.

از سوی دیگر، تحلیل مسائل یادگیری شبکه‌های عصبی از منظر پیچیدگی محاسباتی نیز یکی از موضوعات چالش‌برانگیز است. مطالعات اولیه، مانند کار بلوم و ریوست [۳]، نشان داد که مسئله بهینه‌سازی وزن‌ها و آستانه‌ها حتی در حالت ساده‌ی شبکه‌ی عصبی با دو لایه و سه نورون به دسته‌ی مسائل ان‌پی‐کامل تعلق دارد. این نتیجه نشان‌دهنده پیچیدگی ذاتی مسائل یادگیری در شبکه‌های عصبی است. پژوهش‌های بعدی، از جمله [۴] و [۵]، با تمرکز بر شبکه‌های مبتنی بر تابع فعال‌ساز یکسوساز[۴] نشان دادند که حتی در حالت‌های خاص، مسئله‌ی یافتن وزن‌ها و آستانه‌های بهینه همچنان ان‌پی‐سخت باقی می‌ماند.

علاوه بر این، در حوزه‌ی هندسه‌ی محاسباتی، پژوهش‌هایی مانند کارهای چان [۶، ۷] نشان داده‌اند که از تکنیک‌های تقسیم‌وحل می‌توان برای کاهش پیچیدگی محاسباتی مسائلی استفاده کرد که در این پژوهش ارتباط آن‌ها را با مسئله‌ی تنظیم دقیق آستانه‌ها تبیین خواهیم کرد. با این حال، پیچیدگی زمانی همچنان به شکل نمایی باقی می‌ماند.

به طور کلی، ادبیات موجود تأکید دارد که تنظیم دقیق پارامترها می‌تواند هزینه‌های یادگیری را کاهش دهد و همچنین نشان می‌دهد که یادگیری شبکه‌های عصبی به دلیل ماهیت خود ذاتاً چالش‌برانگیز است. این پایان‌نامه با تمرکز بر هر دو جنبه، به دنبال بررسی پیچیدگی محاسباتی حالتی ساده‌تر از مسئله‌ی یادگیری است، که همان تنظیم دقیق پارامترهای برگزیده است.

---
[۴] Rectified Linear Unit (ReLU)

۱-۴ اهداف پژوهش

هدف این پژوهش بررسی مسئله‌ی تنظیم دقیق مقادیر آستانه در شبکه‌های عصبی است که پیش‌تر اهمیت آن به منظور کاهش سربار محاسباتی و بهبود عملکرد مورد توجه قرار گرفته است. همچنین، این تحقیق با تمرکز بر تحلیل پیچیدگی محاسباتی این مسئله، به دنبال ارائه‌ی درک عمیق‌تر از سختی ذاتی این مسائل از طریق یافتن کران‌های پایین محاسباتی است. به کمک استفاده از ابزارهای نظری در حوزه پیچیدگی پارامتری و یافتن الگوریتم‌های کاهش، این مطالعه به دنبال مرتبط کردن این مسئله با دیگر مسائل شناخته‌شده‌ی الگوریتمی است تا بتوان به شهود بهتری از رده‌ی سختی آن دست یافت. پیش‌بینی می‌شود که نتایج این پژوهش، راه را برای درک بهتر ساختار پیچیدگی مسائل یادگیری در حیطه‌ی شبکه‌های عصبی هموار کند.

۱-۵ ساختار پایان‌نامه

این پایان‌نامه به جهت ارائه‌ی کار پژوهشی انجام‌شده در زمینه‌ی بررسی الگوریتمی یک مسئله‌ی تنظیم دقیق در شبکه‌های عصبی و در قالب پنج فصل تنظیم شده است. ساختار کلی پایان‌نامه به صورت زیر است:

۱-۵-۱ مقدمه

این فصل ابتدا به معرفی کلی موضوع تحقیق و اهمیت آن در حوزه‌ی یادگیری ماشین الگوریتمی می‌پردازد. سپس مسئله‌ی مورد پژوهش یعنی تنظیم دقیق آستانه‌ها در شبکه‌های عصبی تعریف شده و پیچیدگی محاسباتی این مسئله مورد بحث قرار می‌گیرد. در نهایت، اهداف اصلی پژوهش و ساختار پایان‌نامه شرح داده می‌شود.

۱-۵-۲ مفاهیم اولیه

این فصل به معرفی مفاهیم پایه‌ای مرتبط با شبکه‌های عصبی و نیز مفاهیم پیچیدگی محاسباتی و به‌ویژه پیچیدگی محاسباتی پارامتری اختصاص دارد. مفاهیمی مانند ساختار شبکه‌ی پرسپترون سه‌لایه، تابع فعال‌ساز و تعاریف رسمی برای مسئله‌ی تنظیم آستانه‌ها با جزییات بیان شده‌اند. همچنین چند مسئله‌ی الگوریتمی مرتبط که در این پژوهش به وسیله‌ی زنجیره‌ای از کاهش‌ها به مسئله‌ی مورد بحث مرتبط می‌شوند تعریف شده‌اند.

۱-۵-۳ کارهای پیشین

در این فصل، مروری بر پژوهش‌های مرتبط ارائه شده است. ابتدا به مطالعاتی که تنظیم دقیق آستانه‌ها و کاربرد آن‌ها در یادگیری شبکه‌های عصبی را بررسی کرده‌اند پرداخته شده و سپس به کارهایی که پیچیدگی پارامتری مسائل مشابه را مورد تحلیل قرار داده‌اند، اشاره می‌شود.

۱-۵-۴ نتایج جدید

این فصل مهم‌ترین یافته‌های پژوهش را ارائه می‌کند. ابتدا الگوریتم پیشنهادی نمایی برای حل مسئله‌ی تنظیم آستانه‌ها توضیح داده شده و سپس کاهش‌های پارامتری از این مسئله به مسائلی در حوزه‌ی هندسه‌ی محاسباتی و نظریه‌ی گراف ارائه می‌شوند که دلالت بر سختی مسئله‌ی مورد پژوهش دارند.

۱-۵-۵ نتیجه‌گیری

در این فصل، یافته‌های اصلی پژوهش در راستای ارائه‌ی الگوریتم و اثبات کلاس سختی مسئله‌ی تنظیم دقیق جمع‌بندی شده و تأثیرات آنها در حوزه‌ی یادگیری ماشین الگوریتمی و پیچیدگی محاسباتی تحلیل می‌شود. همچنین پیشنهاداتی برای پژوهش‌های آینده، شامل بررسی مسئله‌ی تنظیم آستانه‌ها در سایر ساختارهای شبکه‌ی عصبی، ارائه می‌گردد.

فصل ۲

مفاهیم اولیه

در این فصل به تعریف مفاهیم اولیه و ذکر صورت مسائلی می‌پردازیم که در این پایان‌نامه مورد استفاده قرار گرفته‌اند.

۲-۱ مفاهیم شبکه‌ی عصبی

۲-۱-۱ تابع فعال‌ساز

در شبکه‌های عصبی، برای ایجاد خاصیت غیرخطی‌بودن از توابع فعال‌ساز غیرخطی استفاده می‌شود که رابطه‌ی بین ورودی‌های یک گره از شبکه و خروجی آن شبکه را مشخص می‌کند. در این پایان‌نامه مانند مقاله‌ی بلوم و ریوست [۳] تابع فعال‌ساز پله را در نظر می‌گیریم که تقریبی از تابع فعال‌ساز سیگماوار[۱] است و به صورت زیر تعریف می‌شود:

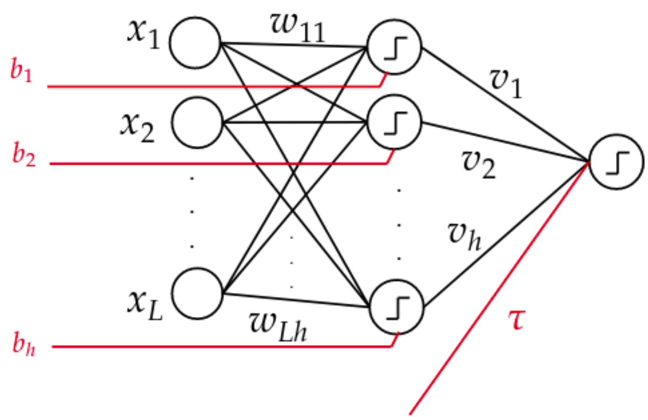
$$\sigma(x) = \begin{cases} ۰ & x \leqslant ۰ \\ ۱ & x > ۰ \end{cases}$$

۲-۱-۲ شبکه‌ی پرسپترون سه‌لایه

یک شبکه‌ی پرسپترون چندلایه[۲] شامل حداقل سه لایه گره است: یک لایه‌ی ورودی، یک لایه‌ی نهان و یک لایه‌ی خروجی. در چنین شبکه‌ای، گره‌های هر لایه تماماً به گره‌های لایه‌ی قبل و بعد از خود متصل‌اند.

[۱] sigmoid
[۲] Multi-Layer Perceptron (MLP)

شکل ۲-۱ ساده‌ترین حالت از چنین شبکه‌ای را نشان می‌دهد که در ادامه برای فرمول‌بندی مسئله‌ی تنظیم دقیق آستانه‌ها استفاده خواهد شد. تابع پله به‌عنوان تابع فعال‌ساز این شبکه انتخاب شده‌است. همان‌طور که در شکل مشاهده می‌شود، به هر نورون علاوه‌بر یال‌های ورودی از لایه‌ی پیشین، یک عدد ثابت به عنوان آستانه یا بایاس[۳] نیز اضافه شده است. مقادیر آستانه همانند وزن‌های یال‌های میان نورون‌ها قابل تنظیم و یادگیری است.



شکل ۲-۱: شمای شبکه‌ی پرسپترون سه‌لایه. شبکه دارای یک بردار ورودی با بعد L، یک خروجی تک‌بعدی دودویی، و یک لایه‌ی نهان با بعد h است.

اکنون مجموعه داده‌های برچسب‌دار به شکل زوج‌مرتب‌های ورودی-خروجی را در نظر بگیرید که شامل بردارهای ورودی $x^{(j)}$ و برچسب‌های خروجی $y^{(j)}$ است. امتیاز شبکه‌ی فوق روی این مجموعه دادگان[۴] را به صورت زیر تعریف می‌کنیم، که برابر است با تعداد داده‌هایی که خروجی شبکه با برچسب آن داده برابر است:

$$score(F, \{(x^{(j)}, y^{(j)})\}) = \sum_{j=1}^{N} I(F(x^{(j)}) = y^{(j)}) \qquad (۱-۲)$$

در این رابطه منظور از I تابع مشخصه‌ای است که مقدار آن در صورت درستی گزاره‌ی ورودی آن برابر با یک، و در غیر اینصورت برابر با صفر است. همچنین F تابعی است که خروجی شبکه‌ی عصبی شکل ۲-۱ را مشخص می‌کند و می‌توان رابطه‌ی آن را به‌صورت زیر نوشت:

$$F(x) = \sigma\left(\tau + \sum_{i=1}^{h} v_i \sigma(w_i^T x + b_i)\right)$$

هدف از مسئله‌ی یادگیری یا تنظیم دقیق، مقداردهی تمام یا بخشی از یال‌های شبکه است به‌گونه‌ای که امتیاز شبکه روی یک مجموعه دادگان به بیشترین مقدار خود برسد.

---

[۳] bias
[۴] dataset

۲-۲ مفاهیم الگوریتمی

۲-۲-۱ مسائل الگوریتمی

در این بخش به بیان دو مسئله‌ی الگوریتمی می‌پردازیم که در نهایت ارتباط نظری آن‌ها با مسئله‌ی تنظیم دقیق را اثبات خواهیم کرد.

تعریف ۲-۱ (جعبه) *یک جعبه در فضای dـ بعدی که به صورت زیر تعریف می‌شود، مکانی در فضاست که محدود است به ابرصفحه‌هایی موازی با محورهای مختصات:*

$$< (l_۱, l_۲, ..., l_d), (r_۱, ..., r_d) >:= \{(x_۱, ..., x_d) \ : \ \forall \ell \in [d] : l_\ell \leqslant x_\ell < r_\ell\}$$

همچنین با مقداردهی l_i با $-\infty$ یا r_i با $+\infty$ می‌توان جعبه‌هایی بی‌کران تعریف کرد.

مسئله‌ی ۲-۱ (عمق) *تعدادی جعبه در فضای dـ بعدی داده شده است. یک نقطه‌ی $(p_۱, ..., p_d)$ در آن فضا پیدا کنید که متعلق به بیشترین تعداد جعبه‌ها باشد.*

تعریف ۲-۲ (خوشه) *یک خوشه زیرمجموعه‌ای از رأس‌های یک گراف است که هر دو رأس مجزا از آن به یکدیگر متصل باشند.*

مسئله‌ی ۲-۲ (d‌ـ‌خوشه) *گرافی با n رأس داده شده است. مشخص کنید که آیا این گراف یک خوشه با حداقل d رأس دارد؟*

۲-۲-۲ پیچیدگی محاسباتی پارامتری

نظریه‌ی پیچیدگی محاسباتی به طور سنتی به دنبال تمایز بین مسائلی است که در زمان چندجمله‌ای حل می‌شوند (از طریق یافتن یک الگوریتم کارا) و مسائلی که ان‌پی‌ـ‌سخت‌اند (از طریق یافتن یک کاهش چندجمله‌ای). با این حال دسته‌ی مسائل دوم همچنان طیف وسیعی از مسائل را شامل می‌شوند و نظریاتی برای تمایز بین آن‌ها با ریزدانگی بیشتر پیشنهاد شده‌است. نظریه‌ی پیچیدگی پارامتری یکی از نظریات پیشنهادی است، که پیچیدگی یک مسئله را برحسب پارامتر دیگری از مسئله (افزون بر اندازه‌ی ورودی) بررسی می‌کند. یک دسته از مسائل در این نظریه، دسته‌ی FPT است که توسط داونی و فلوز [۸] پیشنهاد شده و به شکل زیر تعریف می‌شود:

تعریف ۲-۳ (FPT) *مسئله‌ی پارامتری $L \in \Sigma^* \times \mathbb{N}$ یک مسئله‌ی FPT*[۵] *خوانده می‌شود اگر و تنها اگر تابع دلخواه f وجود داشته باشد که بتوان مسئله‌ی L را برای هر ورودی x با پارامتر k در زمان $f(k) \cdot |x|^{O(1)}$ حل کرد.*

مثالی از مسئله‌ای که گمان می‌رود متعلق به دسته‌ی FPT نباشد، مسئله‌ی رنگ‌آمیزی گراف است. می‌دانیم که تشخیص ۳ــ رنگ‌پذیربودن یک گراف، ان‌پی‌ــ کامل است. الگوریتمی از مرتبه‌ی زمانی برابر با $f(k) \cdot |x|^{O(1)}$ برای $k = ۳$ در زمان چندجمله‌ای اجرا خواهد شد. پس اگر مسئله‌ی پارامتری رنگ‌آمیزی گراف در دسته‌ی FPT باشد، آنگاه $\mathrm{P} = \mathrm{NP}$ نتیجه خواهد شد.

همچنین نامحتمل است که مسئله‌ی پارامتری خوشه که بالاتر تعریف شد متعلق به دسته‌ی FPT باشد. چن و همکاران [۹] نشان داده‌اند که مسئله‌ی خوشه در زمان $f(k) \cdot n^{o(k)}$ حل نخواهد شد، مگر آنکه حدس ETH[۶] نقض شود. حدس ETH حدسی قوی‌تر از $\mathrm{P} \neq \mathrm{NP}$ است که بیان می‌کند مسئله‌ی ۳ــ صدق‌پذیری[۷] در زمان کمتر از نمایی یعنی در زمان $۲^{o(n)}$ حل نمی‌شود.

برای آنکه نشان دهیم مسئله‌ی جدیدی دارای درجه‌ی سختی پارامتری مشابه با مسئله‌ی خوشه است و در نتیجه به احتمال بالا متعلق به دسته‌ی FPT نیست، نیاز است که یک کاهش FPT از مسئله‌ی خوشه به آن مسئله ارائه دهیم. مطابق [۸] چنین کاهشی به صورت زیر تعریف می‌شود:

تعریف ۲-۴ (کاهش FPT) *یک کاهش FPT از مسئله‌ی P به مسئله‌ی Q، تابع ϕ با شرایط زیر است:*

- *$\phi(x)$ یک نمونه‌ی مثبت*[۸] *برای مسئله‌ی Q است اگر و تنها اگر x یک نمونه‌ی مثبت برای مسئله‌ی P باشد.*
- *$\phi(x)$ قابل محاسبه در زمان $f(k) \cdot |x|^{O(1)}$ باشد، که k همان پارامتر برای ورودی x است.*
- *اگر k پارامتر ورودی x و k' پارامتر ورودی $\phi(x)$ باشد، تابع g وجود داشته باشد که $k' \leqslant g(k)$ باشد.*

در فصل ۴ یک کاهش FPT از مسئله‌ی خوشه به مسئله‌ی تنظیم دقیق آستانه‌ها ارائه خواهیم کرد که نشان خواهد داد مسئله‌ی دوم نیز متعلق به دسته‌ی FPT نیست و نامحتمل است که در زمان $f(k) \cdot n^{o(k)}$ حل شود.

---

[۵] Fixed-Parameter Tractable
[۶] Exponential Time Hypothesis
[۷] 3-SAT
[۸] yes-instance

فصل ۳

کارهای پیشین

در زمینه‌ی بهینه‌سازی شبکه‌های عصبی، تنظیم دقیق مقادیر آستانه به عنوان یک رویکرد سبک و مؤثر مطرح شده است. آستانه‌ها که سطح پاسخ تابع فعال‌ساز را تنظیم می‌کنند، برای تطبیق شبکه‌های عصبی با وظایف جدید به صورت کارآمد استفاده می‌شوند. پژوهش‌های اخیر مانند چارچوب BitFit نشان داده‌اند که تنظیم دقیق مقادیر آستانه به‌تنهایی در مدل‌های مبتنی بر مبدّل[۱]، نتایجی قابل قبول در مقایسه با تنظیم دقیق تمام پارامترها ارائه می‌دهد. این روش به‌طور چشم‌گیری سربار محاسباتی و نیاز به حافظه را کاهش داده است [۱]. این رویکرد، پتانسیل بهینه‌سازی انتخابی پارامترها را در کاهش نیاز به منابع محاسباتی بدون کاهش چشم‌گیر در عملکرد برجسته می‌کند.

پیچیدگی محاسباتی مرتبط با یادگیری و تنظیم دقیق شبکه‌های عصبی نیز یکی از حوزه‌های پژوهشی مهم و چالش‌برانگیز است. مطالعات متعددی به بررسی سختی نظری این مسائل پرداخته‌اند و نشان داده‌اند که بسیاری از مسائل یادگیری، از جمله حالاتی خاص از یادگیری شبکه‌های عصبی، در دسته‌ی مسائل سخت محاسباتی مانند ان‌پی‌ـ کامل قرار دارند. به عنوان نمونه، بلوم و ریوست اثبات کرده‌اند که بهینه‌سازی یک شبکه‌ی عصبی ساده با سه نورون یک مسئله‌ی ان‌پی‌ـ کامل است [۳]. این نتایج نشان می‌دهند که بسیاری از چالش‌های محاسباتی در یادگیری ماشین، ریشه در سختی ذاتی مسائل بهینه‌سازی مرتبط دارند.

علاوه بر این، برخی از مسائل محاسباتی در حوزه‌های دیگر نیز از نظر مفهومی شباهت‌هایی با چالش‌های یادگیری و تنظیم دقیق مورد بررسی در این پژوهش دارند. برای مثال، مسئله‌ی عمق [۲] و مسئله‌ی اندازه‌گیری کلی[۳] از جمله مسائل هندسه‌ی محاسباتی هستند که پیچیدگی و سختی حل آن‌ها ارتباطاتی با مسئله‌ی

[۱] transformer
[۲] depth problem
[۳] Klee's measure problem

بهینه‌سازی آستانه‌ها در شبکه‌های عصبی دارند. همچنین، مسئله‌ی خوشه[۴] و نسخه‌ی پارامتری آن از منظر کاهش‌های محاسباتی و ارائه‌ی کران‌های پایین مرتبط، نقش مهمی ایفا می‌کنند. در این فصل، پژوهش‌های مرتبط با این مسائل مورد بررسی قرار خواهند گرفت. ارتباط دقیق بین این مسائل و مسئله‌ی تنظیم دقیق آستانه‌ها که در این پژوهش مورد بررسی قرار گرفته است، در فصل مربوط به نتایج جدید تبیین خواهد شد.

این فصل به بررسی جامع این آثار پیشین می‌پردازد تا پایه‌ای محکم برای پژوهش ارائه‌شده در این پایان‌نامه فراهم کند. با مرور اهمیت تنظیم دقیق مقادیر آستانه، بررسی‌های پیچیدگی محاسباتی در حوزه‌ی بهینه‌سازی شبکه‌های عصبی، و مسائل محاسباتی مرتبط، می‌توان جایگاه این پژوهش را در زمینه‌ی وسیع‌تر تحقیقات علمی به وضوح مشخص کرد.

۳-۱ نقش آستانه‌ها در شبکه‌های عصبی

تنظیم دقیق آستانه‌ها در شبکه‌های عصبی یکی از روش‌های کارآمد بهینه‌سازی است که تأثیر قابل توجهی در کاهش پیچیدگی محاسباتی و مصرف منابع دارد. پژوهش‌های متعددی در این زمینه نشان داده‌اند که این رویکرد نه‌تنها امکان یادگیری وظایف جدید با سربار محاسباتی کمتر را فراهم می‌کند، بلکه در بسیاری از موارد عملکرد شبکه را نیز حفظ یا حتی بهبود می‌بخشد. در این بخش، به بررسی چند اثر پژوهشی می‌پردازیم که به نقش مقادیر آستانه در بهینه‌سازی شبکه‌های عصبی و بهبود اثربخشی آن در شرایط منابع محدود پرداخته‌اند. این آثار زمینه‌ساز درکی عمیق‌تر از اهمیت و کاربرد این مقادیر در تنظیم دقیق شبکه‌های عصبی هستند.

نویسندگان [۱۰] به دنبال یافتن پاسخی برای امکان انجام وظایف یادگیری شبکه‌ی عصبی در دستگاه‌های لبه[۵] با توان پردازشی و پایین و حافظه‌ی محدود، پیشنهاد دادند که تنها مقادیر آستانه‌ی نورون‌ها بهینه شوند. آن‌ها دو اثر مثبت برای این کار ذکر کردند: نخست آنکه در این روش تعداد پارامترهای شبکه که باید بهینه شوند کاهش چشم‌گیری می‌یابد؛ در حالی که در مقایسه با بهینه‌سازی تمام پارامترهای لایه‌ی آخر، بهبود دقت هم مشاهده کردند. دیگر آنکه گلوگاه اصلی در میزان حافظه‌ی مورد نیاز برای روش انتشار معکوس[۶]، ضرورت ذخیره‌ی مقادیر میانی فعال‌سازی نورون‌های شبکه است . حال اگر تنها مقادیر آستانه باشند که متغیر هستند، این ضرورت از بین رفته و بهبود چشم‌گیر حافظه مشاهده می‌شود.

زاکن و همکاران [۱] در مقاله‌ی خود ضمن ارجاع به کار فوق که روی مدل‌های بینایی کامپیوتر آزموده شده بود، پیشنهاد کردند که تنظیم دقیق مدل زبانی BERT تنها برای پارامترهای آستانه انجام شود. آن‌ها

[۴] clique problem
[۵] edge devices
[۶] backpropagation

گزارش کردند که برای حجم داده‌ی کوچک تا متوسط، این کار به دقتی مشابه با تنظیم دقیق تمام پارامترهای شبکه رسید. همچنین برای حجم داده‌های بزرگ، روش خود را قابل قیاس با دیگر روش‌های تنظیم دقیق تُنُک دانستند که زیرمجموعه‌های دیگری از پارامترهای شبکه را برای یادگیری انتخاب می‌کردند. آن‌ها استدلال کردند که این نتیجه‌ی عملی شاهدی است که تنظیم دقیق یک شبکه‌ی زبانیِ ازپیش آموخته‌شده، نه عملی برای یاددادن وظایف جدید به شبکه، بلکه افشای دانش و توانایی‌ای است که شبکه از پیش آموخته است و این کار می‌تواند با بهینه‌سازی پارامترهای آستانه ممکن شود.

نویسندگان [۱۱] با استفاده از روش مبتنی بر پرامپت، نشان دادند که بهینه‌سازی صرفاً پارامترهای آستانه در مدل‌های زبانی ازپیش‌آموخته، می‌تواند عملکرد یادگیری صفر/کم‌نمونه[۷] را بهبود بخشد. آن‌ها با بهره‌گیری از داده‌های ورودی بی‌معنی تولیدشده توسط GPT-4 برای آشکارسازی سوگیری‌های ذاتی مدل، توانستند به طور مؤثری این سوگیری‌ها را واسنجی[۸] کرده و نقطه شروع منصفانه‌ای برای مدل ایجاد کنند. این کار، ضمن کاهش هزینه‌های محاسباتی، اثربخشی تنظیم دقیق آستانه‌ها را در بهبود قابلیت انطباق مدل با داده‌های جدید برجسته می‌کند.

یکی دیگر از پژوهش‌های برجسته در این حوزه، روش DiffFit را معرفی می‌کند که با هدف تنظیم دقیق مدل‌های انتشاری[۹] بزرگ برای کاربرد در حوزه‌های جدید طراحی شده است [۲]. DiffFit رویکردی کارآمد و ساده است که تنها با تنظیم مقادیر آستانه و افزودن ضرایب در لایه‌های خاص، سرعت آموزش را به میزان قابل توجهی افزایش می‌دهد و هزینه‌ی ذخیره‌سازی مدل را کاهش می‌دهد. این روش در مقایسه با تنظیم دقیق کامل مدل، دو برابر سرعت یادگیری را بهبود داده و تنها به ذخیره‌ی حدود ۰/۱۲ درصد از کل پارامترهای مدل نیاز داشت. این مطالعه نشان می‌دهد که حتی برای مدل‌های بزرگ، تنظیم دقیق پارامترهای آستانه می‌تواند راهکاری عملی و مؤثر برای کاهش پیچیدگی زمانی و همچنین حافظه‌ای باشد.

۲-۳ پیچیدگی محاسباتی مسائل یادگیری

یکی از جنبه‌های مهم در بررسی شبکه‌های عصبی، پیچیدگی محاسباتی مرتبط با فرآیند یادگیری آن‌ها است. پژوهش‌های متعددی به بررسی این حوزه پرداخته‌اند و نشان داده‌اند که بسیاری از مسائل یادگیری، حتی در شبکه‌های ساده، در دسته‌ی مسائل سخت محاسباتی قرار می‌گیرند. در این بخش، برخی از مطالعات برجسته در این حوزه مرور می‌شوند.

بلوم و ریوست [۳] در یکی از پژوهش‌های پیشگام، اثبات کردند که آموزش یک شبکه‌ی عصبی ساده

---

[۷] zero/few-shot learning

[۸] calibrate

[۹] diffusion models

با سه نورون در دسته‌ی مسائل ان‌پی‌ـ کامل قرار دارد. آن‌ها نشان دادند که یافتن وزن‌ها و آستانه‌هایی که خروجی شبکه را برای مجموعه‌ای از داده‌های آموزشی بهینه کند، در زمان چندجمله‌ای قابل حل نیست. این نتیجه، یکی از نخستین شواهد نظری در تأیید پیچیدگی ذاتی مسائل یادگیری شبکه‌های عصبی بود.

در ادامه‌ی این مسیر، مانورانگسی و ریچمن [۴] پیچیدگی محاسباتی آموزش شبکه‌های عصبی با توابع فعال‌ساز ReLU را بررسی کردند. آن‌ها نشان دادند که حتی برای شبکه‌ای با یک واحد ReLU، یافتن وزن‌هایی که خطای میانگین مربعات را برای یک مجموعه‌ی داده به حداقل برسانند یا دست‌کم تقریب بزنند، مسئله‌ای ان‌پی‌سخت است. همچنین در شبکه‌هایی با دو واحد ReLU، حتی در حالت‌هایی که داده‌ها به‌طور کامل قابل یادگیری هستند و یافتن وزن‌هایی برای خطای صفر ممکن است، این مسئله همچنان ان‌پی‌سخت باقی می‌ماند. این پژوهش دشواری آموزش شبکه‌های عصبی مبتنی بر ReLU را که امروزه به‌طور گسترده در یادگیری عمیق استفاده می‌شوند، برجسته می‌کند.

فروزه و همکاران [۵] به بررسی پیچیدگی آموزش شبکه‌های ReLU با تمرکز بر پارامترسازی براساس بُعد داده‌ها پرداختند. آن‌ها نشان دادند که پیچیدگی محاسباتی آموزش این شبکه‌ها با افزایش ابعاد داده‌ها بیشتر می‌شود. این مطالعه تأکید می‌کند که ویژگی‌های مجموعه‌داده می‌توانند به طور مستقیم بر کارایی و امکان‌پذیری آموزش شبکه‌های عصبی تأثیر بگذارند.

در پژوهش دیگری، فروزه و همکاران [۱۲] پیچیدگی آموزش شبکه‌های عصبی در ابعاد ثابت را مورد بررسی قرار دادند. آن‌ها اثبات کردند که حتی در صورتی که بُعد داده‌ها ثابت باشد، آموزش شبکه‌های عصبی مسئله‌ای ان‌پی‌سخت باقی می‌ماند. این نتیجه نشان می‌دهد که محدود کردن بُعد داده‌ها لزوماً مسئله‌ی آموزش شبکه‌ها را ساده‌تر نمی‌کند.

مطالعه‌ی دیگری توسط آبراهامسن و همکاران [۱۳] پیچیدگی آموزش شبکه‌های عصبی را از منظر نظریه‌ی وجودی اعداد حقیقی[۱۰] مورد بررسی قرار داده است. آن‌ها نشان دادند که آموزش شبکه‌های عصبی با توابع فعال‌ساز ReLU، مسئله‌ای $\exists\mathbb{R}$ـ کامل است. این نتیجه بدان معناست که این مسئله معادل با تصمیم‌گیری درباره‌ی وجود راه‌حلی برای سیستمی از معادلات و نامعادلات چندجمله‌ای با ضرایب صحیح و متغیرهای حقیقی است. این دسته از مسائل حتی پیچیده‌تر از مسائل ان‌پی‌ـ کامل هستند و ممکن است در رده‌ی ان‌پی قرار نگیرند. این پژوهش اهمیت پیچیدگی ذاتی آموزش شبکه‌های عصبی و چالش‌های محاسباتی مرتبط با آن را از منظر ریاضیاتی برجسته می‌کند.

این پژوهش‌ها نشان می‌دهند که مسائل یادگیری و بهینه‌سازی شبکه‌های عصبی ریشه در سختی محاسباتی ذاتی دارند. شناخت این چالش‌ها برای درک محدودیت‌های ذاتی مدل‌های شبکه عصبی ضروری است.

[۱۰] existential theory of the reals

۳-۳ مسائل الگوریتمی مرتبط

در این بخش، به بررسی مسائل الگوریتمی مرتبط با پژوهش حاضر می‌پردازیم. تمرکز اصلی بر روی مسئله‌ی عمق و مسئله‌ی خوشه است که از طریق کاهش‌های محاسباتی که در فصل بعد اثبات خواهند شد با مسئله‌ی ما در ارتباط هستند. در ادامه، چند کار پژوهشی مرتبط با معرفی الگوریتم برای این مسائل یا بررسی پیچیدگی محاسباتی آن‌ها را مرور می‌کنیم.

۳-۳-۱ مسئله‌ی عمق

مسئله‌ی عمق به دنبال تعیین حداکثر تعداد جعبه‌هایی است که در یک نقطه از فضا با هم تلاقی دارند.

چان در مقاله‌ی ابتدایی خود در [۶] الگوریتمی با پیچیدگی زمانی $n^{d/2}2^{O(log^*n)}$ برای مسئله‌ی اندازه‌گیری کلی[۱۱] ارائه داد که به‌طور مستقیم به مسئله‌ی عمق مرتبط است. در این الگوریتم، n تعداد جعبه‌ها و d بعد فضا است. این الگوریتم با استفاده از تکنیک تقسیم‌وحل و داده‌ساختارهای کارآمد، بهبود قابل‌توجهی نسبت به روش‌های قبلی ارائه می‌دهد. او همچنین یافته‌شدن الگوریتمی با مرتبه‌ی زمانی $o(n^{d/2-\varepsilon})$ را غیرمنتظره دانست.

در ادامه، چان در [۷] با ساده‌سازی روش‌های پیشین، الگوریتمی با پیچیدگی زمانی اندکی بهتر، یعنی برابر با $O(n^{d/2})$ ارائه داد. هرچند بهبود زمانی ارائه‌شده چشمگیر نیست، راه‌حل جدید چان به‌جهت سادگی در ایده و پیاده‌سازی قابل توجه است، و نیز بهترین الگوریتم یافته‌شده تا به امروز برای حل مسئله‌ی اندازه‌گیری کلی و مسئله‌ی عمق است.

۳-۳-۲ مسئله‌ی خوشه

مسئله‌ی پارامتری خوشه به دنبال یافتن زیرگرافی کامل با اندازه‌ی حداقل k در یک گراف داده‌شده است. این مسئله به دلیل کاربردهای گسترده در علوم کامپیوتر مورد توجه بسیاری قرار گرفته است و نسخه‌ی غیرپارامتری آن یکی از ۲۱ مسئله‌ی ان‌پی‌ـ کامل است که در مقاله‌ی مشهور کارپ مطرح شدند [۱۴].

مسئله‌ی خوشه به عنوان یک مسئله‌ی W[1]ـ کامل شناخته می‌شود، زیرا نویسندگان [۱۵] نشان دادند که تحت فرضیه‌ی ETH مسئله‌ی خوشه نمی‌تواند در زمان $O(f(k) \cdot n^{o(k)})$ برای هر تابع $f(k)$ حل شود. این نتیجه نشان‌دهنده‌ی سختی ذاتی این مسئله در چارچوب پیچیدگی‌های پارامتری است.

[۱۱] Klee's measure problem

علاوه بر این، نویسندگان [۱۶] نشان دادند که تقریب ثابت برای مسئله‌ی پارامتری خوشه نیز W[1]-سخت است، به این معنا که حتی یافتن تقریب‌هایی با نسبت ثابت نیز در زمان چندجمله‌ای و با ثابت‌گرفتن k امکان‌پذیر نیست، مگر آنکه اثبات شود رده‌های سختی پارامتری FPT و W[1] با یکدیگر برابرند.

فصل ۴

نتایج جدید

در این فصل، نتایج جدید حاصل از پژوهش حاضر ارائه و تحلیل می‌شوند. تمرکز اصلی این پژوهش بر روی بررسی پیچیدگی محاسباتی مسئله‌ی تنظیم دقیق آستانه‌ها در شبکه‌های عصبی با یک لایه‌ی نهان و تابع فعال‌ساز پله بوده است. با بهره‌گیری از ابزارهای نظری در حوزه‌ی پیچیدگی محاسباتی پارامتری، نشان داده شده است که این مسئله به‌طور ذاتی سخت است و نامحتمل است که بتوان آن را با الگوریتمی چندجمله‌ای حل کرد، حتی در شرایطی که ابعاد شبکه ثابت باشد.

برای رسیدن به این نتیجه، ابتدا مسئله‌ی تنظیم دقیق آستانه‌ها به‌طور دقیق فرمول‌بندی شده و سپس با استفاده از کاهش‌های محاسباتی، ارتباط این مسئله با مسائل سخت شناخته‌شده‌ای همچون یافتن خوشه در گراف نشان داده شده است. این تحلیل‌ها نه تنها سختی محاسباتی مسئله‌ی مورد بررسی را آشکار می‌کنند، بلکه به فهم عمیق‌تر چالش‌های مرتبط با یادگیری و بهینه‌سازی شبکه‌های عصبی کمک می‌کنند.

در ادامه‌ی این فصل، ابتدا یک الگوریتم تقسیم و حل برای مسئله ارائه شده و سپس کران‌های پایین محاسباتی ارائه شده‌اند. این نتایج به‌عنوان گامی به سوی درک بهتر پیچیدگی مسائل مرتبط با یادگیری شبکه‌های عصبی ارائه می‌شوند.

۴-۱ الگوریتم تقسیم و حل

در این بخش الگوریتمی را ارائه خواهیم داد که مسئله‌ی تنظیم دقیق آستانه‌ها را برای یک شبکه‌ی پرسپترون سه‌لایه که شمای آن در شکل ۲-۱ ارائه شده بود حل کند. تعریف دقیقی در مسئله در ادامه ارائه شده است.

مسئله‌ی ۴-۱ (تنظیم دقیق آستانه‌ها) *برای یک شبکه‌ی پرسپترون سه‌لایه با ابعاد* $d \times h \times 1$ *نورون با*

وزن‌های داده‌شده‌ی $w_1, \dots, w_d \in \mathbb{R}^h$ *و مجموعه‌ی* n *داده‌ی برچسب‌دار* $(x^{(j)}, y^{(j)}) \in \mathbb{R}^d \times \{0, 1\}$ *که خروجی شبکه روی هر داده‌ی ورودی توسط تابع زیر محاسبه می‌شود:*

$$F(x) = \sigma\left(\tau + \sum_{i=1}^{h} v_i \sigma(w_i^T x + b_i)\right) \tag{4-1}$$

مطلوب است یافتن مقادیر بهینه‌ی آستانه، $b = (b_1, \dots, b_d)$ *و نیز* τ*، چنان‌که تعداد داده‌های ورودی که خروجی شبکه برای آن‌ها برابر با برچسب‌شان است بیشینه شود:*

$$b_{\text{OPT}}, \tau_{\text{OPT}} = \underset{b,\tau}{\text{argmax}} \sum_{j=1}^{n} I(F(x^{(j)}) = y^{(j)})$$

برای این مسئله، یک الگوریتم تقسیم و حل ارائه خواهیم داد که بر پایه‌ی الگوریتم چان [7] برای حل مسئله‌ی عمق است. فرض کنید زیرمجموعه‌ی S از نورون‌های لایه‌ی نهان داده شده که مقدار آستانه‌ی آن‌ها قابل تنظیم است. همچنین فرض کنید که برای یک ورودی دلخواه $x^{(j)}$ مایلیم مقادیر آستانه را به‌گونه‌ای تنظیم کنیم که خروجی شبکه با برچسب خروجی آن برابر باشد. اکنون 2^d زیرمجموعه از نورون‌های لایه‌ی نهان را درنظر می‌گیریم که هرکدام حالتی را مشخص می‌کنند از فعال‌بودن تعدادی از نورون‌ها و غیرفعال‌بودن بقیه. فعال یا غیرفعال بودن نورون iـ‌ام از لایه‌ی نهان، یک محدودیت خطی روی مقدار آستانه‌ی آن را اعمال می‌کند؛ اگر آن نورون فعال باشد، باید شرط زیر برقرار باشد:

$$w_i^T x^{(j)} + b_i \geqslant 0 \Rightarrow b_i \geqslant -w_i^T x^{(j)}$$

و اگر غیرفعال باشد، شرط معکوس زیر باید برقرار باشد:

$$w_i^T x^{(j)} + b_i < 0 \Rightarrow b_i < -w_i^T x^{(j)}$$

اگر شرط‌هایی که به ازای هر زیرمجموعه از نورون‌های فعال روی بردار b اعمال می‌شود را لحاظ کنیم، 2^d می‌توانیم آن‌ها را به صورت 2^d جعبه در فضای dـ‌بعدی نشان دهیم که مجزا هستند و این فضا را افراز می‌کنند.

حال مسئله را بسته به آنکه آستانه‌ی لایه‌ی آخر (τ) قابل تنظیم یا ثابت باشد دو حالت می‌کنیم. اگر حالت ثابت را در نظر بگیریم، برای هر یک از 2^d حالت از فعال یا غیرفعال‌بودن نورون‌های لایه‌ی نهان می‌توانیم مشخص کنیم که خروجی شبکه با برچسب $y^{(j)}$ برابر شده است یا نه. برای هر ورودی، اگر بردار b درون یکی از جعبه‌های متناظر با چنین حالتی باشد، شبکه یک امتیاز در رابطه‌ی 2-1 خواهد گرفت. پس اگر برای n داده‌ی ورودی $2^d \times n$ جعبه بسازیم و الگوریتم چان را بری حل مسئله‌ی عمق در این فضا

به کار بگیریم، مقادیری برای آستانه‌های b را خواهیم یافت که بیشترین امتیاز را روی مجموعه‌ی دادگان ورودی خواهند گرفت. یادآوری می‌شود که جعبه‌های مرتبط با هر داده‌ی ورودی مجزا هستند، پس هر تقاطع نقطه‌ی b با یک جعبه، یک امتیاز برای داده‌ای متفاوت را برای شبکه نشان می‌دهد.

از سوی دیگر اگر τ نیز قابل تنظیم باشد، به ازای هرکدام از حالت‌های فعال یا غیرفعال بودن نورون‌های لایه‌ی نهان، مشابه با حالت قبل محدودیت‌هایی روی مقادیر b_1 تا b_d خواهیم داشت و این بار افزون بر آن، اگر خروجی مدنظر برای نورون‌های لایه‌ی نهان را با F_1 تا F_d نشان دهیم، یک محدودیت روی مقدار τ خواهیم داشت تا بسته به آنکه برچسب خروجی یک یا صفر باشد، خروجی شبکه (که از رابطه‌ی ۴-۱ به‌دست می‌آید) با آن برچسب برابر شود:

$$\tau \underset{y^{(j)}=0}{\overset{y^{(j)}=1}{\gtrless}} -\sum_{i=1}^{h} v_i F_i$$

پس در این حالت هم می‌توان بخش مطلوب از فضای جستجو برای مقادیر آستانه‌ها را که برای یک داده‌ی ورودی منجر به کسب امتیاز برای شبکه می‌شود توسط 2^d جعبه در فضای $\mathbb{R}^{d+1}$ نشان داد، و مجدداً کافیست مسئله‌ی عمق را در فضای جستجوی مقادیر آستانه‌ها برای $2^d \times n$ جعبه حل کنیم.

با این توصیف، در ادامه الگوریتم چان را که برای مسئله‌ی عمق به کار می‌رود، با شرایط مسئله‌ی خودمان بازنویسی کنیم. در ادامه با اغماض با پارامتر d برخورد می‌کنیم تا به سادگی نشان‌دهنده‌ی ابعاد فضای جست‌وجو در مسئله‌ی عمق باشد، مستقل از آنکه مقدار آستانه‌ی نهایی، τ، در مسئله‌ی ما قابل تنظیم باشد یا خیر.

الگوریتم ۱ الگوریتم چان برای مسئله‌ی عمق

ورودی: عدد d، جعبه‌های $B_1, ..., B_n$ و نیز Γ، و d تابع پله‌ای متفاوت $H_1, ..., H_d$.

خروجی: بیشترین امتیازی که یک نقطه می‌تواند در Γ کسب کند.

۱: اگر n کمتر از یک مقدار مشخص باشد، پاسخ را مستقیماً مصاحبه کن و برگردان.

۲: نوارها را پیدا کن و توابع پله‌ای را به‌روز کن.

۳: ناحیه‌ی Γ را به Γ_L و Γ_R تقسیم کن.

۴: به‌طور خروجی را برای Γ_L و Γ_R را محاسبه کن و مقدار بزرگتر را برگردان.

همان‌طور که در الگوریتم فوق مشخص است، چان حالتی عمومی‌تر از مسئله‌ی عمق را حل می‌کند که در آن، امتیاز هر نقطه‌ی $p = (x_1, ..., x_d)$ برابر است با تعداد جعبه‌هایی که حاوی آن نقطه هستند به‌علاوه‌ی مقادیر $H_i(x_i)$. این تعمیم به تقسیم و حل مسئله کمک می‌کند.

در ابتدا تمام توابع H_i را برابر با صفر قرار می‌دهیم و ناحیه‌ی Γ را هم کل فضای $\mathbb{R}^d$ می‌گیریم. منظور از نوار[1] در هندسه‌ی محاسباتی و به‌خصوص در این کار، ناحیه‌ای از فضاست که محدود به دو ابرصفحه‌ی عمود بر یکی از محورهای مختصات است. در این الگوریتم وقتی از نوار iـام ($Slab_i$) صحبت می‌کنیم، منظورمان مجموعه‌ای از جعبه‌هاست که وقتی به ناحیه‌ی Γ محدود شوند مانند یک نوار باشند و تنها در یکی از ابعاد ناحیه را محدود کنند. همچنین منظورمان از نوار ستاره ($Slab_*$) اجتماع چنان نوارهایی است. این الگوریتم تلاش می‌کند که در هر گام این نوارها و جعبه‌های متناظر را حذف کند تا فضای جست‌وجو ساده‌تر شود.

از آنجا که در $Slab_i$ جعبه‌ها تنها محدود به یک بعد هستند، می‌توانیم با افزودن مقدار $+۱$ به تابع پله‌ای H_i در بازه‌های متناظر، اثر این جعبه‌ها در امتیاز نقاط داخل‌شان را به آن شکل اعمال کنیم، و در ازای آن، حعبه‌های $Slab_i$ را حذف کنیم. اگر در ابتدای الگوریتم یک پیش‌پردازش انجام دهیم و جعبه‌ها را در تمام d بعد از فضا مرتب کنیم، این گام یعنی یافتن نوارها و به‌روزرسانی توابع پله‌ای تنها با یک پاک‌سازی خطی[2] در هر بعد انجام می‌شود.

پس از پایان گام دوم که ساده‌سازی بود، در گام سوم از الگوریتم، ناحیه‌ی Γ را با یک برش به دو ناحیه‌ی Γ_R و Γ_L تقسیم می‌کنیم. برای این کار، الگوریتم ابتدا به هر یک از $(d-۲)$ـوجه‌ها[3] در فضای $\mathbb{R}^d$ وزنی اختصاص می‌دهد. برای وجهی که عمود بر محورهای iـام و jـام است، وزن اختصاص‌شده برابر است با $۲^{\frac{(i+j)}{d}}$.

سپس ناحیه‌ی Γ را با استفاده از ابرصفحه $x_۱ = m$ که m میانه وزن‌دار اولین مختصات $(d-۲)$-وجه‌هایی است که به محور اول عمود هستند و Γ را قطع می‌کنند، به دو ناحیه تقسیم می‌کنیم. سپس الگوریتم محورهای مختصات را دوباره شماره‌گذاری می‌کنیم به طوری که محورهای قدیمی $۱, ۲, ..., d$ به محورهای جدید $d, ۱, ..., d-۱$ متناظر شوند.

پس از تقسیم، برای هر یک از Γ_L و Γ_R می‌توانیم مجدداً وزن‌ها را برای $(d-۲)$ـوجه‌ها محاسبه کنیم. برای وجه‌هایی که وزن $۲^{\frac{i+j}{d}}$ دارند و $i, j \neq ۱$ هستند، وزن آن‌ها با یک ضریب $۲^{\frac{۲}{d}}$ تغییر می‌کند. برای یک وجه عمود بر محور اول و محور jام، وزن آن از $۲^{\frac{۱+j}{d}}$ به $۲^{\frac{d+j-۱}{d}}$ تغییر می‌کند؛ با این حال، تعداد آن‌ها در هرکدام از Γ_L و Γ_R نصف می‌شود که به این معنی است که مجموع وزن‌های آن‌ها نیز با یک ضریب $۲^{\frac{۲}{d}}$ کاهش می‌یابد. توجه کنید که پس از هر تقسیم، ما جعبه‌هایی را که هیچ تلاقی با ناحیه‌ی محدودکننده‌ی مربوطه ندارند، به سادگی حذف می‌کنیم.

بر این اساس، از آنجا که هر وزنی که به یک وجه اختصاص داده شده عددی بین ۱ تا ۴ است، مجموع

[1] slab
[2] sweep line
[3] $(d-2)$-faces

وزن‌ها برای همه $(d-۲)$ـ وجه‌ها حداکثر $۴\binom{d}{۲}n$ خواهد بود. بنابراین، اگر $T(n,S)$ را به عنوان یک مسئله‌ی عمق با S به عنوان مجموع وزن‌های وجه‌ها و n به عنوان تعداد جعبه‌ها در نظر بگیریم، داریم:

$$T(n,S) \leqslant ۲T(n, \frac{S}{۲^{۲/d}}) + O(n) \leqslant ۲T(n, \frac{S}{۲^{۲/d}}) + O(S)$$

با حذف n و استفاده از قضیه‌ی اصلی[۴] برای تحلیل مجانبی، می‌توانیم نشان دهیم که اگر $d = ۲$ باشد، این الگوریتم مرتبه‌ی زمانی $O(n\log n)$ دارد و اگر $d > ۲$ باشد، مرتبه‌ی زمانی الگوریتم $O(S^{d/۲})$ می‌شود. از آنجا که ما $O(d^۲)$ وجه داریم که هر کدام وزن $O(۱)$ دارند، می‌توانیم استنباط کنیم که $S = d^۲n$ است که به این معنی است که برای $d > ۲$ ما الگوریتمی با مرتبه‌ی زمانی $O((d^۲n)^{d/۲}) = O(d^dn^{d/۲})$ داریم.

حال اگر به مسئله‌ی تنظیم دقیق آستانه برگردیم و d را تعداد پارامترهای آستانه‌ی قابل تنظیم بگیریم، سپس اگر تمام $۲^d \times n$ جعبه را لحاظ کنیم، برای $d \geqslant ۳$ می‌توانیم یک الگوریتم با مرتبه‌ی زمانی $O(d^d \cdot (۲^dn)^{\frac{d}{۲}}) = O(d^d \cdot \sqrt{۲}^{d^۲} n^{\frac{d}{۲}})$ را به دست آوریم. همچنین برای $d = ۲$ ما یک الگوریتم $O(n\log n)$ داریم.

توضیحی مشابه را می‌توان برای حالتی که آستانه‌ی τ قابل تنظیم است بیان کرد که پیچیدگی زیر را به دست می‌دهد:

$$O(d^d \times (۲^{d-۱}n)^{\frac{d}{۲}}) = O(d^d \cdot ۲^{d^۲} n^{\frac{d}{۲}})$$

در هر دو حالت، ما یک الگوریتم $O(f(d)n^{\frac{d}{۲}})$ برای مسئله‌ی تنظیم دقیق آستانه به دست آورده‌ایم و در ادامه به دنبال کران‌های پایین برای مسئله هستیم.

۴-۲ کاهش پارامتری

در این بخش به کمک چند لم نشان می‌دهیم که می‌توان مسئله‌ی یافتن خوشه در گراف را به مسئله‌ی تنظیم دقیق آستانه‌ها کاهش داد. برای این کار از مسئله‌ی عمق به عنوان یک مسئله‌ی میانی استفاده کرده‌ایم، و همچنین کاهش‌های ارائه‌شده ویژگی‌های یک کاهش FPT را برآورده می‌کنند. در نتیجه نشان داده می‌شود مسئله‌ی تنظیم دقیق آستانه به صورتی که در بخش قبل فرمول‌بندی شد، قابل حل در زمان $f(d)\cdot n^{o(d)}$ نخواهد بود.

لم ۴-۱ *مسئله‌ی یافتن یک خوشه با d رأس در گرافی با n رأس و m یال قابل کاهش است به مسئله‌ی عمق در فضای dـ بعدی با $۴\binom{d}{۲}(n^۲-۲m)+۱$ جعبه.*

---
[۴] master theorem

اثبات. ابتدا یک جعبه‌ی اولیه به شکل $<(۰,۰,...,۰),(n,...,n)>$ در نظر می‌گیریم. همچنین رئوس گراف را از ۰ تا $n-۱$ نام‌گذاری می‌کنیم. سپس اگر مجموعه‌ی یال‌های گراف را E بنامیم، برای هر زوج‌مرتب (u,v) که $\{u,v\} \notin E$ و برای هر $i,j \in \{۰,\dots n-۱\}$ که $i<j$ باشد، چهار جعبه به صورت زیر تعریف می‌کنیم:

$$B_{i,j,u,v,۰۰} =< (\underbrace{-\infty,...,-\infty}_{i-۱},u,-\infty,...,-\infty),(\underbrace{+\infty,...,+\infty}_{j-۱},v,+\infty,...,+\infty) >$$

$$B_{i,j,u,v,۱۱} =< (\underbrace{-\infty,...,-\infty}_{j-۱},v+۱,-\infty,...,-\infty),(\underbrace{+\infty,...,+\infty}_{i-۱},u+۱,+\infty,...,+\infty) >$$

$$B_{i,j,u,v,۰۱} =< (-\infty,...,-\infty),(\underbrace{+\infty,...,+\infty}_{i-۱},u,\underbrace{+\infty,...,+\infty}_{j-i-۱},v+۱,+\infty,...,+\infty) >$$

$$B_{i,j,u,v,۱۰} =< (\underbrace{-\infty,...,-\infty}_{i-۱},u+۱,\underbrace{-\infty,...,-\infty}_{j-i-۱},v,-\infty,...,-\infty),(+\infty,...,+\infty) >$$

به سادگی می‌توان مشاهده کرد که این چهار جعبه مجزا هستند و در نتیجه هر نقطه‌ی دلخواه p در فضای $d-$بعدی حداکثر در یکی از این چهار جعبه قرار دارد. در واقع اگر $u \leqslant p_i < u+۱$ و $v \leqslant p_j < v+۱$ باشد، نقطه‌ی p در یکی از این چهار جعبه قرار دارد؛ و در غیر این صورت، خارج از این چهار جعبه است.

حال می‌توانیم این جعبه‌ها را به عنوان ورودی مسئله‌ی عمق در نظر بگیریم و بررسی کنیم که آیا جواب آن برابر با $\binom{d}{۲}(n^۲-۲m)+۱$ است یا خیر. از آنجا که چهار جعبه‌ی متناظر با i,j,u,v اشتراکی ندارند، رسیدن به جواب فوق تنها در حالتی رخ می‌دهد که نقطه‌ی p وجود داشته باشد که هم در جعبه‌ی اولیه باشد (به بیان دیگر تمام اندیس‌های i داشته باشیم $۰ \leqslant p_i < n$) و هم برای تمام اندیس‌های i,j,u,v، نقطه‌ی p در حداقل یکی از چهار جعبه‌ی متناظر باشد. وجود چنین نقطه‌ای به این معناست که هیچ دو اندیس i,j و هیچ $\{u,v\} \notin E$ وجود نداشته باشد که $u \leqslant p_i < u+۱$ و $v \leqslant p_j < v+۱$ باشد. گزاره‌ی فوق به این معناست که برای هر i,j که $i \neq j$ داشته باشیم $\{\lfloor p_i \rfloor, \lfloor p_j \rfloor\} \in E$. پس هر دو مختصات از نقطه‌ی p توسط یک یال به هم متصل‌اند، و وجود این نقطه به معنای وجود یک خوشه‌ی d رأسی در گراف است.

$\square$

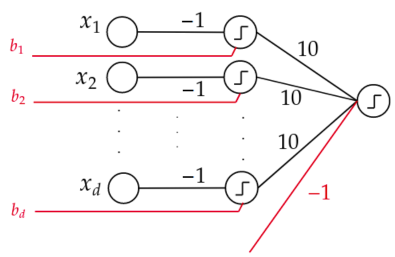
اکنون می‌توان نتیجه گرفت که مسئله‌ی عمق در زمان $O(f(d)n^{o(d)})$ قابل حل نیست، زیرا مسئله‌ی پارامتری خوشه بنا به فرض ETH در زمان $O(f(d)n^{o(d)})$ قابل حل نیست.

حال که یک کاهش از مسئله‌ی خوشه به مسئله‌ی عمق داریم، به سراغ ارائه‌ی کاهشی از مسئله‌ی عمق

به مسئله‌ی تنظیم دقیق آستانه (برای شبکه‌ی پرسپترون سه‌لایه تصویرشده در شکل ۲-۱) می‌رویم. با این حال، بررسی ما نشان داده که بسته به اینکه مقدار آستانه برای نورون لایه‌ی آخر (که آن را با τ نشان می‌دهیم) ثابت یا قابل تنظیم باشد، طبیعت مسئله متفاوت خواهد بود. به همین دلیل، در ادامه و در دو لم مجزا به هر کدام می‌پردازیم.

لم ۴-۲ *مسئله‌ی عمق با k جعبه در فضای dـ بعدی قابل کاهش است به مسئله‌ی تنظیم دقیق آستانه‌ها برای یک شبکه‌ی پرسپترون سه‌لایه با تابع فعال‌ساز پله و یک لایه‌ی نهان با d نورون، روی یک مجموعه‌ی دادگان ورودی به اندازه‌ی $k \times ۲^d$، در حالی که مقدار آستانه‌ی لایه‌ی آخر* ***ثابت*** *باشد.*

اثبات. یک ورودی برای مسئله‌ی عمق (شامل k جعبه در فضای dـ بعدی) را در نظر می‌گیریم. در ابتدا، یک عدد مثبت M به اندازه‌ی کافی بزرگ را پیدا می‌کنیم که مختصات گوشه‌های تمام جعبه‌ها بین M و $-M$ باشد. حال اگر جعبه‌هایی بی‌کران داشتیم (که طبق تعریف مسئله‌ی عمق مجاز است)، کران‌های نامحدود $-\infty$ را با $-M$ و کران‌های $+\infty$ را با $+M$ جایگزین می‌کنیم. پاسخ مسئله‌ی عمق با این جایگزینی تغییری نمی‌کند.



شکل ۴-۱: یک شبکه‌ی پرسپترون سه‌لایه که برای کاهش مسئله‌ی عمق به مسئله‌ی تنظیم دقیق آستانه استفاده می‌کنیم. لایه‌ی آخر در این شبکه نقش یک «یای منطقی» را ایفا می‌کند به‌گونه‌ای که نورون خروجی شبکه فعال خواهد شد اگر و تنها اگر حداقل یکی از نورون‌های لایه‌ی نهان فعال شوند.

سپس از شبکه‌ی عصبی که در شکل ۴-۱ به تصویر کشیده شده استفاده می‌کنیم و مجموعه‌ی دادگان برای تنظیم دقیق را در ادامه تعریف می‌کنیم. برای هر زیرمجموعه مانند S از $\{۰, \ldots d-۱\}$ و هر جعبه‌ی $B =< (l_۱, l_۲, ..., l_d), (r_۱, ..., r_d) >$ یک زوج $(x^{(S,B)}, y^{(S,B)})$ با مقادیر زیر به مجموعه‌ی دادگان ورودی شبکه‌ی عصبی اضافه می‌کنیم:

$$x^{(S,B)} = \begin{pmatrix} x_۱^{(S,B)} & x_۲^{(S,B)} & \ldots & x_d^{(S,B)} \end{pmatrix}, x_i^{(S,B)} = \begin{cases} r_i & i \in S \\ l_i & i \notin S \end{cases} \qquad (۴-۲)$$

$$y^{(S,B)} = \begin{cases} ۰ & |S| \overset{۲}{\equiv} d \\ ۱ & |S| \overset{۲}{\not\equiv} d \end{cases} \qquad (۳-۴)$$

به ازای هر جعبه، مجموعه‌ی دادگان فوق حاوی ۲^d به شکل زوج مرتب است که اعضای اول آن‌ها متناظر با ۲^d گوشه‌ی آن جعبه هستند. همچنین اعضای دوم، به شکل برچسب‌هایی دودودیی هستند که زوجیت[۵] گوشه‌ها را نشان می‌دهند. زوجیت یک گوشه از جعبه‌ی B، زوجیت تعداد محورهایی است که آن گوشه در آن محور با کران بالای جعبه (r_i) مقداردهی شده است. به عنوان مثال زوجیت گوشه‌ی $(l_۱, l_۲, ..., l_d)$ برابر با صفر است.

اکنون خود را محدود به مجموعه‌ی دادگان مرتبط با فقط یک جعبه مثل $B_۱$ می‌کنیم، که حاوی ۲^d داده است. ادعا می‌کنیم که اگر مقدار انتخابی برای مقادیر آستانه‌ی $b = (b_۱, \ldots, b_d)$ معادل با نقطه‌ای خارج جعبه‌ی $B_۱$ باشد، امتیاز شبکه‌ی عصبی یا به عبارت دیگر تعدادی از داده‌هایی که خروجی شبکه با برچسب آن‌ها یکسان است برابر با $۲^{d-۱}$ می‌شود، و اگر b متناظر با نقطه‌ای داخل جعبه باشد، این امتیاز $۲^{d-۱} + ۱$ می‌شود. اگر این ادعا ثابت شود، تنظیم دقیق آستانه‌های شبکه‌ی شکل ۴-۱ (که همان بهینه‌کردن مقدار بردار آستانه‌های b است به گونه‌ای که امتیاز شبکه روی مجموعه‌ی دادگان ساخته‌شده از روی جعبه‌ها بیشینه شود) معادل با یافتن نقطه‌ای درون بیشترین تعداد جعبه‌ها خواهد بود.

یک انتخاب برای مقادیر آستانه به صورت $b = (b_۱, \ldots, b_d)$ را برای شبکه‌ی شکل ۴-۱ در نظر بگیرید. باتوجه به ساختار این شبکه، نورون iـام لایه‌ی نهان به شرطی فعال می‌شود که $x_i \leqslant b_i$ باشد. همچنین خروجی شبکه در صورتی که حداقل یکی از نورون‌های لایه‌ی نهان فعال شود برابر با یک، و در غیر این صورت برابر با صفر است.

برای سادگی، یک انتخاب برای بردار b را با رشته‌ای از نمادهای '␣'، '+' و '−' نشان می‌دهیم. اگر $l_i \leqslant b_i < r_i$ باشد نماد iـام در رشته‌ی متناظرش را '␣' قرار می‌دهیم، اگر $b_i \geqslant r_i$ باشد آن را با '+' نشان می‌دهیم، و نهایتاً اگر $b_i < l_i$ باشد آن را با '−' نشان می‌دهیم. در ادامه نشان خواهیم داد که برای هر نقطه‌ی b که رشته‌ی متناظرش هر چیزی به‌جز "␣␣...␣␣" باشد امتیاز شبکه‌ی عصبی برابر با $۲^{d-۱}$ است و برای این رشته‌ی خاص که نمایان‌گر نقطه‌ی داخل جعبه‌ی B. است امتیاز شبکه برابر با $۲^{d-۱} + ۱$ است.

ابتدا نقاطی را در نظر بگیرید که رشته‌ی متناظرشان حداقل یک نماد '+' داشته باشد. در این حالت می‌دانیم که حداقل یک نورون فعال خواهد شد، پس شبکه‌ی عصبی فقط برای داده‌هایی با $y^{(S,B_۱)} = ۱$ امتیاز خواهد گرفت. تعداد چنین داده‌هایی برابر با $۲^{d-۱}$ است، که متناظر با تعداد زیرمجموعه‌هایی مانند S است که $|S|$ زوجیتی متفاوت از d داشته باشد. به طور مشابه نشان داده می‌شود که اگر رشته‌ی متناظر

[۵] parity

با بردار b تنها شامل نماد '$-$' باشد، هیچ نورون لایه‌ی نهان فعال نخواهد شد و شبکه فقط امتیاز داده‌هایی با برچسب $y^{(S,B_۱)} = ۰$ را خواهد گرفت که آن‌ها نیز $۲^{d-۱}$ هستند.

اکنون نقاطی را در نظر بگیرید که رشته‌ی متناظرشان به تعداد t واحد نماد '$\sqcup$' و به تعداد $d-t$ واحد نماد '$-$' دارد. برای داده‌ای با برچسب صفر، شبکه‌ی عصبی تنها در صورتی امتیازش را می‌گیرد که در اندیس‌هایی متناظر با نماد '$\sqcup$'، ورودی $x_i^{(S,B_۱)}$ برابر با l_i باشد؛ زیرا در غیر این صورت نورون iـام فعال خواهد شد و خروجی شبکه‌ی عصبی صفر نخواهد بود. برای اندیس‌های دیگر فارغ از اینکه $x_i^{(S,B_۱)}$ برابر با l_i یا r_i باشد، نورون iـام فعال نخواهد شد. با این حال زیرمجموعه‌ی S از اندیس‌ها باید به‌گونه‌ای انتخاب شود که $d \overset{۲}{\equiv} |S|$ تا برچسب خروجی صفر باشد. پس برای $۲^{d-t-۱}$ داده با برچسب صفر، شبکه‌ی عصبی امتیازشان را می‌گیرد. برای داده‌های با برچسب یک، کافیست حداقل برای یک اندیس i از t اندیسی که متناظر با نماد '$\sqcup$' هستند داشته باشیم $x_i = l_i$؛ تا نورون iـام و در نتیجه خروجی شبکه‌ی عصبی فعال شود. برای اندیس‌های متناظر با نماد '$-$' فارغ از اینکه ورودی چه باشد، نورون متناظرشان فعال نخواهد شد. پس $۲^t - ۱$ انتهاب برای اندیس‌های دسته‌ی اول و سپس $۲^{d-t-۱}$ انتخاب برای اندیس‌های باقی‌مانده داریم تا به داده‌هایی با برچسب یک برسیم که خروجی شبکه‌ی عصبی هم فعال باشد. در مجموع، تعداد داده‌هایی که شبکه‌ی عصبی امتیازشان را در این حالت می‌گیرد برابر است با

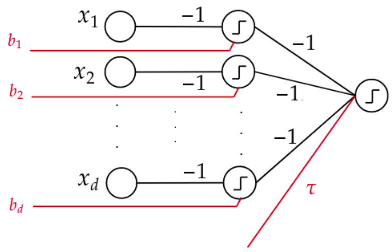
$$۲^{d-t-۱} + (۲^t - ۱) \times ۲^{d-t-۱} = ۲^{d-۱}.$$

اکنون به بررسی تنها حالت باقی‌مانده می‌پردازیم که انتخاب برای b متناظر با رشته‌ای تنها شامل نمادهای '$\sqcup$' باشد (که به معادل نقطه‌ای درون جعبه‌ی $B_۱$ است). برای داده‌های با برچسب صفر، هیچ‌کدام از نورون‌های لایه‌ی نهان نباید فعال شوند؛ پس برای تمام اندیس‌ها باید $x_i = r_i$ باشد یعنی $S = \{۱, \dots, d\}$ باشد. از آنجا که در این حالت $d \overset{۲}{\equiv} |S|$ است، پس مطابق رابطه‌ی ۴-۳ خروجی شبکه‌ی عصبی با برچسب داده مطابقت خواهد داشت. برای داده‌های با برچسب یک، کافیست حداقل یک x_i برابر با l_i باشد تا نورون iـام و در نتیجه خروجی شبکه فعال شود، و نیز $(d-۱) \overset{۲}{\equiv} |S|$ باشد تا شرط برچسب یک تضمین شود. برای برقراری شرط هم‌نهشتی $۲^{d-۱}$ انتخاب برای زیرمجموعه‌ی S داریم که به سادگی مشاهده می‌شود در تمام آن‌ها حداقل برای یک اندیس i، شرط $x_i = r_i$ برقرار است. پس در مجموع در این حالت امتیاز شبکه برابر است با $۲^{d-۱} + ۱$.

در نتیجه می‌توانیم به ازای k جعبه‌ی ورودی، یک مجموعه‌ی دادگان با اندازه‌ی $۲^d k$ بسازیم و با بیشینه‌کردن امتیاز شبکه‌ی شکل ۴-۱ چنان‌که در رابطه‌ی ۲-۱ آمده است، به امتیاز $۲^{d-۱} \times k + T$ برسیم که T تعداد جعبه‌هایی است که بردار بهینه‌ی b یا همان مقادیر آستانه‌ی نورون‌ها درون آن جعبه‌ها قرار می‌گیرد. پس با تفریق مقدار $۲^{d-۱} \times k$ از پاسخ مسئله‌ی تنظیم دقیق آستانه‌ها، می‌توانیم پاسخ مسئله‌ی

عمق را بیابیم. پس درستی کاهش نشان داده شد. □

لم ۴-۳ *مسئله‌ی عمق با k جعبه در فضای dــ بعدی قابل کاهش است به مسئله‌ی تنظیم دقیق آستانه‌ها برای یک شبکه‌ی پرسپترون سه‌لایه با تابع فعال‌ساز پله و یک لایه‌ی نهان با d نورون، روی یک مجموعه‌ی دادگان ورودی به اندازه‌ی $k \times ۲^d$، در حالی که مقدار آستانه‌ی لایه‌ی آخر* ***قابل تنظیم*** *باشد.*



شکل ۴-۲: یک شبکه‌ی پرسپترون سه‌لایه که برای کاهش مسئله‌ی عمق به مسئله‌ی تنظیم دقیق آستانه در حالتی که آستانه‌ی نورون خروجی قابل تنظیم باشد.

اثبات. برای این بخش هم مجموعه‌ی دادگان ورودی را مشابه لم قبل و به ازای ۲^d گوشه‌ی هر جعبه می‌سازیم، و مشابه لم قبل جعبه‌های بی‌کران را محدود می‌کنیم. اما این بار از یک شبکه با وزن‌های متفاوت استفاده می‌کنیم که در شکل ۴-۲ آمده است.

برای آنکه یک انتخاب برای b روی یک ورودی x با برچسب y امتیاز بگیرد ابتدا باید تعداد نورون‌های فعال‌شده در لایه‌ی نهان را بشماریم؛ یعنی تعداد اندیس‌هایی که $b_i - x_i \geqslant ۰$ باشد. اگر این تعداد را t بنامیم، آستانه‌ی نورون لایه‌ی خروجی (τ) محدود به t خواهد بود: بسته به اینکه برچسب خروجی برابر با صفر یا یک باشد، مقدار τ باید بزرگتر از یا کوچکترمساوی با t باشد.

اکنون نشان می‌دهیم مستقل از اینکه چه مقداری برای τ انتخاب کنیم، اگر بردار $b = (b_۱, \ldots, b_d)$ خارج از جعبه‌ی B باشد، شبکه همیشه امتیاز $۲^{d-۱}$ را روی مجموعه‌ی دادگان متناظر با گوشه‌های جعبه‌ی B خواهد گرفت. زیرا اگر $b \notin B$ باشد پس مجموعه‌ای از اندیس‌های i وجود دارند که b_i در بازه‌ی $[l_i, r_i)$ نباشد. اولین اندیس با این ویژگی را درنظر بگیرید و آن را j بنامید، و ابتدا فرض کنید $b_i < l_i$ باشد. هر داده‌ی ورودی شبکه که متناظر با زیرمجموعه‌ی $S \subseteq \{۱, \ldots, d\}$ است می‌تواند با داده‌ی متناظر با زیرمجموعه‌ی S' جفت شود، که با حذف یا اضافه‌کردن اندیس j به مجموعه‌ی S حاصل می‌شود: $S' = S \oplus \{j\}$. از آنجا که b_j پیش از بازه‌ی $[l_j, r_j)$ در نظر گرفته شد، نورون jــام مستقل از آنکه $x_j = l_j$ یا $x_j = r_j$ باشد، همیشه غیرفعال خواهد بود. اما داده‌های متناظر با یکی از دو مجموعه‌ی S و S' یکی دارای برچسب صفر و دیگری دارای برچسب یک است زیرا اندازه‌های این دو مجموعه زوجیتی متفاوت دارند. این بدان معناست که هرطور τ و باقی مقادیر آستانه مقداردهی شوند، شبکه از هر جفت داده دقیقاً

یک امتیاز خواهد گرفت. استدلالی مشابه را می‌توان برای حالتی آورد که $b_j \geqslant r_i$ باشد. پس برای هر جعبه‌ی B که $b \notin B$ باشد، شبکه امتیاز نیمی از دادگان مرتبط با آن، یعنی امتیاز $۲^{d-۱}$ داده را خواهد گرفت.

از طرف دیگر، حالتی را در نظر بگیرید که b درون جعبه‌ی B باشد. برای یک انتخاب τ، $R(\tau)$ را مجموعه‌ی اعداد صحیح زوج از صفر تا d بگیرید که بزرگتر از τ هستند و ، و $L(\tau)$ را مجموعه‌ی اعداد صحیح فرد از صفر تا d بگیرید که کوچکترمساوی τ هستند . در این صورت، امتیاز شبکه روی دادگان متناظر با گوشه‌های جعبه‌ی B برابر با مقدار زیر خواهد بود:

$$\sum_{t \in L(\tau) \cup R(\tau)} \binom{d}{t} \tag{۴-۴}$$

زیرا برای دادگان با برچسب صفر (که تعداد اندیس‌هایشان که $x_i = l_i$ است زوج است) باید تعداد نورون‌های فعال‌شده (یعنی تعداد اندیس‌هایی که $x_i = l_i$ است) بیشتر از τ باشد، و برای دادگان با برچسب یک (که تعداد اندیس‌هایشان که $x_i = l_i$ است فرد است) باید تعداد نورون‌های فعال‌شده (یعنی تعداد اندیس‌هایی که $x_i = l_i$ است) کمترمساوی τ باشد. با این حساب، پاسخ مسئله‌ی تنظیم دقیق باید τ را چنان بیابد که ۴-۴ را بیشینه کند و نیز b را چنان انتخاب کند که درون بیشترین تعداد جعبه‌ها قرار بگیرد. پس پاسخ مسئله‌ی تنظیم دقیق برای b متناظر می‌شود با یک پاسخ بهینه برای مسئله‌ی عمق، لذا درستی کاهش برای حالتی که آستانه‌ی لایه‌ی آخر قابل تنظیم باشد هم نشان داده شد.

□

فصل ۵

نتیجه‌گیری

در این فصل، در ابتدا نتایج جدید ارائه‌شده در پایان‌نامه را جمع‌بندی می‌کنیم و سپس مسائل باز باقی‌مانده و همچنین پیشنهادهایی برای ادامه‌ی کار ارائه خواهیم کرد.

۵-۱ نتایج

در این پایان‌نامه، به بررسی پیچیدگی محاسباتی حالتی از مسئله‌ی تنظیم دقیق شبکه‌های عصبی پرداختیم تا نشان دهیم که یافتن یک الگوریتم قطعی و چندجمله‌ای برای آن، همانند حالت‌های دیگری از مسئله‌ی یادگیری شبکه‌های عصبی، نامحتمل است. نشان دادیم که یافتن الگوریتمی FPT برای مسئله‌ی تنظیم دقیق آستانه‌ها در شبکه‌ای با یک لایه‌ی نهان یافتن نمی‌شود مگر آنکه چنین الگوریتمی برای مسئله‌ی پارامتری خوشه در نظریه‌ی گراف‌ها هم وجود داشته باشد، که این امر در تناقض با فرض مشهور ETH خواهد بود [۹]. همچنین برای مسئله‌ی خود الگوریتمی قطعی با مرتبه‌ی زمانی $O(f(d)n^{\frac{d}{2}})$ ارائه دادیم که در آن d برابر با تعداد نورون‌های لایه‌ی نهان در ساختار شبکه و n برابر با اندازه‌ی مجموعه‌ی دادگان آموزشی است. هر دو گام این پژوهش، ارائه‌ی الگوریتم و نیز اثبات پیچیدگی زمانی، به کمک یافتن کاهشی دوطرفه بین این مسئله و مسئله‌ی عمق در هندسه‌ی محاسباتی ممکن شد، که خود قابل کاهش به مسئله‌ی خوشه است.

۵-۲ کارهای آینده

در این پایان‌نامه برای اثبات پیچیدگی مسئله‌ی تنظیم دقیق، موفق به ارائه‌ی کاهشی FPT به یک مسئله‌ی ان‌پی-کامل شدیم. اما این کاهش هنوز برای نشان‌دادن ان‌پی-سخت بودن کافی نیست، زیرا اگرچه

شرایط لازم را برای یک کاهش FPT را دارد (چنان‌که در تعریف ۲-۴ بیان شد)، اما هنوز یک کاهش چندجمله‌ای نیست؛ زیرا یک ورودی از مسئله‌ی اول با پارامتر d و اندازه‌ی n را به ورودی دیگری از مسئله‌ی دوم با اندازه‌ی $O(2^d n^c)$ متناظر می‌کند. نیاز است که با کوچک‌کردن گجت‌ها در تناظر ایجادشده، ضریب نمایی 2^d به ضریبی چندجمله‌ای برحسب d تبدیل شود تا بتوان ان‌پی‌ـسخت بودن مسئله‌ی تنظیم دقیق را نیز نتیجه گرفت. در آن صورت، عدم وجود الگوریتم چندجمله‌ای برای این مسئله هم‌ارز خواهد بود با فرض $\mathcal{P} \neq \mathcal{NP}$ که فرضی محتمل‌تر از فرض ETH است. [۱۷]

همچنین در این کار فرض کردیم که تابع فعال‌ساز نورون‌ها یک تابع پله باشد. این فرض به ما کمک کرد تا تناظری با مسئله‌ی عمق بیابیم، که تنها محدود به جعبه‌ها می‌شد و نه اشکال هندسی دیگر. برای ادامه‌ی کار می‌توان حالاتی را بررسی کرد که تابع فعال‌ساز به شکل ReLU یا تابع سیگماوار باشد، که هر کدام به شکل بالقوه طبیعت مسئله را تغییر می‌دهند.

معماری‌های دیگری از شبکه‌های عصبی را نیز می‌توان مورد بررسی قرار داد. در این پژوهش به بررسی معماری متداول و تماماً متصل[۱] از یک شبکه‌ی عصبی پرداختیم، که محدود به یک لایه‌ی نهان بود. حالتی دیگر از مسئله که شایسته‌ی توجه است، معماری شبکه‌ی عصبی پیچشی[۲] است که در سال‌های اخیر کاربردهای گسترده‌تری یافته است.

---

[۱] fully connected network

[۲] Convolutional Neural Network (CNN)

مراجع

[1] E. B. Zaken, S. Ravfogel, and Y. Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–9, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[2] E. Xie, L. Yao, H. Shi, Z. Liu, D. Zhou, Z. Liu, J. Li, and Z. Li. Difffit: Unlocking transferability of large diffusion models via simple parameter-efficient fine-tuning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, page 4207–4216. IEEE, Oct. 2023.

[3] A. Blum and R. Rivest. Training a 3-node neural network is np-complete. *Advances in Neural Information Processing Systems*, 1:494–501, 1988.

[4] P. Manurangsi and D. Reichman. The computational complexity of training relu(s). *arXiv preprint arXiv:1810.04207*, 2018.

[5] V. Froese, C. Hertrich, and R. Niedermeier. The computational complexity of relu network training parameterized by data dimensionality. *Journal of Artificial Intelligence Research*, 74:1775–1790, Aug. 2022.

[6] T. M. Chan. A (slightly) faster algorithm for klee's measure problem. *Computational Geometry*, 43(3):243–250, Apr. 2010.

[7] T. M. Chan. Klee's measure problem made easy. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 410–419, 2013.

[8] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer New York, 1999.

[9] J. Chen, X. Huang, I. A. Kanj, and G. Xia. Strong computational lower bounds via parameterized complexity. *Journal of Computer and System Sciences*, 72(8):1346–1367, Dec. 2006.

[10] H. Cai, C. Gan, L. Zhu, and S. Han. Tinytl: Reduce memory, not parameters for efficient on-device learning. *Advances in Neural Information Processing Systems*, 33:11285–11297, 2020.

[11] K. He, Y. Long, and K. Roy. Prompt-based bias calibration for better zero/few-shot learning of language models. *arXiv preprint arXiv:2402.10353*, 2024.

[12] V. Froese and C. Hertrich. Training neural networks is np-hard in fixed dimension. In *Advances in Neural Information Processing Systems*, pages 44039–44049. Curran Associates, Inc., 2023.

[13] M. Abrahamsen, L. Kleist, and T. Miltzow. Training neural networks is er-complete. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, volume 34 of *NIPS '21*, pages 18293–18306, Red Hook, NY, USA, 2021. Curran Associates Inc.

[14] R. M. Karp. Reducibility among combinatorial problems. *Journal of Symbolic Logic*, 40(4):618–619, Dec. 1975.

[15] Y. Chen, K. Eickmeyer, and J. Flum. *The Exponential Time Hypothesis and the Parameterized Clique Problem*, page 13–24. Springer Berlin Heidelberg, 2012.

[16] B. Lin. Constant approximating k-clique is w[1]-hard. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, STOC '21, page 1749–1756. ACM, June 2021.

[17] M. Cygan, F. V. Fomin, . Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer International Publishing, 2015.

واژه‌نامه

۳

۳ـ صدق‌پذیری 3-SAT

الف

آستانه .... threshold
ان‌پی ـ کامل .... NP-complete
انتشار معکوس .... backpropagation

ب

بایاس، اریبی .... bias

پ

پاک‌سازی خطی .... sweep line
پرسپترون چندلایه Multi-Layer Perceptron (MLP)
پیچیدگی پارامتری .... parameterized complexity

ت

تقریب .... approximation
تنظیم دقیق .... fine-tuning

د

دستگاه‌های لبه .... edge devices
دودویی .... binary

ز

زوجیت .... parity

س

سیگماوار .... sigmoid

ش

شبکه‌ی تماماً متصل .... fully connected network
شبکه‌ی عصبی .... neural network
شبکه‌ی عصبی پیچشی .... Convolutional Neural Network (CNN)

ص

صدق‌پذیری .... satisfiability

ف

فرضیه‌ی زمان نمایی .... Exponential Time Hypothesis (ETH)

ق

قضیه‌ی اصلی .... master theorem
قطعی .... deterministic

م

مبدّل .... transformer

مجموعه دادگان .... dataset

مدل انتشاری .... diffusion model

مسئله‌ی اندازه‌گیری کلی . Klee's measure problem

مسئله‌ی خوشه .... clique problem

مسئله‌ی عمق .... depth problem

ن

نظریه‌ی وجودی اعداد حقیقی . existential theory of the reals

نمونه‌ی مثبت .... yes-instance

نوار .... slab

و

واحد یکسوساز خطی .... Rectified Linear Unit (ReLU)

واسنجی .... calibrate

وجه .... face

ی

یادگیری کم‌نمونه .... few-shot learning

Abstract

The study of the computational complexity of learning problems in the field of artificial intelligence algorithms holds significant theoretical importance. One such problem is the fine-tuning of neural networks. Previous studies indicate that fine-tuning, or optimizing a subset of neural network weights on new datasets—particularly the optimization of bias terms—can achieve performance comparable to fine-tuning all network weights. Given the relative simplicity of this specific case, analyzing its time complexity and demonstrating its computational hardness can provide insights into the inherent difficulty of the broader learning problem.

In this research, we consider a binary classification neural network with one hidden layer and a step activation function. By reducing the problem of finding a clique in a graph to the problem of fine-tuning bias terms in this neural network, we demonstrate that, under the ETH assumption, this problem does not belong to the class of FPT problems. This implies that if the neural network dimensions are fixed, there exists no polynomial-time algorithm in terms of dataset size for solving this problem. The results of this study not only reveal the computational hardness of this specific problem but also provide intuition about the inherent complexity of learning and fine-tuning in neural networks, paving the way for further investigation into more complex problems in this domain.

Keywords: Fine-Tuning, Bias Terms, Parameterized Complexity, Step Activation Function

Sharif University of Technology
Department of Computer Engineering

B.Sc. Thesis

Computational Complexity of Fine-Tuning Thresholds in Neural Networks

By:

Ali Mahdavifar

Supervisor:

Hamid Zarrabi-Zadeh

January 2025