

فصل دوازدهم: ترکیب یادگیری تحلیلی و استقرایی

متدهای استقرایی محض فرضیه‌ای کلی را با پیدا کردن قاعده‌های نمونه‌های آموزشی ایجاد می‌کنند. متدهای تحلیلی محض نیز از دانش قبلی برای نتیجه‌گیری منطقی فرضیه کلی استفاده می‌کنند. این فصل متدهایی را بررسی خواهد کرد که مکانیسم‌های استقرایی و تحلیلی را ترکیب کرده و از منافع هر دو روش استفاده می‌کنند: دقت تعمیمی بیشتری با استفاده از دانش قبلی موجود دارند و در زمانی که دانش قبلی ضعف دارد به داده‌های آموزشی مشاهده شده اتکا خواهند کرد. متدهای ترکیبی حاصل از متدهای استقرایی محض و تحلیلی محض کارایی بیشتری خواهند داشت. این فصل متدهای یادگیری استقرایی تحلیلی را بر اساس نمایش‌های نمادین و شبکه‌های عصبی مصنوعی بررسی خواهد کرد.

۱۲,۱ انگیزه

در فصل‌های قبلی دو الگوی^۱ از یادگیری ماشین را دیدیم: یادگیری استقرایی و یادگیری تحلیلی. متدهای استقرایی، مثل یادگیری درختی و Backpropagation در شبکه‌های عصبی، به دنبال فرضیه‌های کلی می‌گردند که متناسب با داده‌های آموزشی باشد. متدهای تحلیلی، مثل Prolog-EBG به دنبال فرضیه‌هایی می‌گردند که با پوشش داده‌های مشاهده شده با دانش قبلی نیز متناسب باشد. این دو الگوی یادگیری بر اساس توضیحات مختلف برای یادگیری فرضیه‌ها ایجاد شده و مزیت‌ها و ضعف‌های مکمل دارند. ترکیب این متدها، متدهای یادگیری‌ای با قابلیت‌ها و قدرت بیشتری ایجاد می‌کند.

متدهای یادگیری تحلیلی محض این مزیت را دارند که تعمیم را با استفاده از میزان داده‌های کمتری انجام می‌دهند و از دانش قبلی برای کنترل یادگیری استفاده می‌کنند. با این وجود، ممکن است با دانش قبلی نادرست یا ناکامل به نتایج غلتی بیانجامد. متدهای استقرایی محض این مزیت را دارند که نیاز به دانش قبلی ندارند و نتایج را کاملاً از خود داده‌های آموزشی استخراج می‌کنند. اما زمانی که داده‌های آموزشی به اندازه‌ی کافی زیاد نیست امکان دارد به‌وسیله‌ی بایاس استقرایی‌شان، که برای تعمیم روی داده‌های مشاهده شده اعمال شده، به نتایج غلتی برسند. جدول

^۱ paradigm

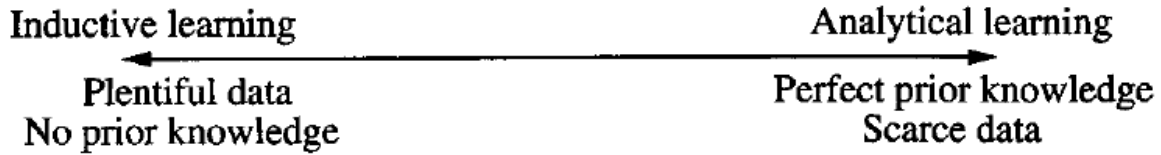
۱۲،۱ خلاصه‌ی این مزیت‌ها و ضعف‌های مکمل متدهای یادگیری استقرایی و تحلیلی را نشان می‌دهد. این فصل به این سؤال که "چگونه می‌توان این دو را در یک الگوریتم که بهترین جنبه‌های هر دو را داشته باشد گنجانده؟" می‌پردازد.

تفاوت متدهای یادگیری استقرایی و توضیحی را می‌توان در طبیعت توجیهی که برای فرضیه‌های یاد گرفته شده‌شان ارائه می‌دهند دید. فرضیه‌ی خروجی متدهای یادگیری تحلیلی محض مثل Prolog-EBG توجیهی منطقی برای فرضیه‌های یاد گرفته شده ارائه می‌دهند؛ فرضیه خروجی را می‌توان از تئوری قلمرو و نمونه‌های آموزشی نتیجه‌گیری کرد. فرضیه‌های خروجی متدهای استقرایی محض مثل Backpropagation توجیه آماری برای فرضیه‌های یاد گرفته شده ارائه می‌دهند؛ فرضیه‌ی خروجی از متغیرهای آماری با فرض اینکه نمونه‌های آموزشی به اندازه‌ی کافی زیاد هستند که نماینده‌ی توزیع نمونه‌ای حاکم بر آن باشند توجیه می‌شود. این توجیه آماری برای استقرا در نتایج PAC-learning در فصل ۷ به طور کامل بحث شده است.

با معلوم بودن اینکه متدهای تحلیلی فرضیه‌هایی با توجیه منطقی خروجی می‌دهند و روش‌های استقرایی متدهایی با توجیه آماری خروجی می‌دهند، روشن است که ترکیب این روش‌ها چگونه مفید خواهد بود: توجیه‌های منطقی با فرض‌هایشان یا همان دانش قبلی که بر پایه آن ساخته شده‌اند محدود می‌شوند و در زمانی که دانش قبلی نادرست و یا در دسترس نباشد این روش‌ها بدون بازده و غیرقابل اعتماد خواهند بود. توجیه‌های آماری فقط با داده‌ها و فرض‌های آماری که می‌کنند محدود می‌شوند و زمانی که فرض‌هایشان درباره‌ی توزیع غیرقابل اعتماد است و یا داده‌ها اندک‌اند این روش‌ها بدون بازده و غیرقابل اعتماد خواهند بود. خلاصه اینکه این دو روش برای انواع مختلفی از مسائل درست کار می‌کنند. با ترکیب آن‌ها می‌توان امید داشت که روشی کلی‌تر برای یادگیری که بر روی طیف وسیعی از مسائل یادگیری درست کار می‌کنند به دست آورد.

جدول ۱۲،۱ طیفی از مسائل یادگیری که در دانش قبلی و حجم داده‌های آموزشی متفاوت‌اند را در بر می‌گیرد. در یک طرف طیف حجم بسیار زیادی از داده‌های آموزشی موجود است و اثری از دانش قبلی نیست. در طرف دیگر طیف، دانش قبلی بسیار قوی موجود است اما داده‌های آموزشی بسیار اندک است. مسائل کاربردی اغلب جایی بین این دو سر طیف هستند. برای مثال، در بررسی پایگاه داده پزشکی برای یادگیری مفهوم "علائمی که در آن درمان X بهتر از درمان Y است" ممکن است فرض اولیه‌ای وجود داشته باشد (مثل حالتی از عمل و عکس‌العمل که مربوط به بیماری است) که می‌گوید دمای بدن بیمار بیشتر از مربوط‌تر از حالت میانی بیمار است---. به طور مشابه، در بررسی پایگاه داده‌ی یک انبار برای یادگیری مفهوم هدف "شرکت‌هایی که میزان انبارشان در طول ده ماه آینده دو برابر می‌شود" ممکن است دانش قبلی داشته باشیم که اقتصاد نوعی عمل و عکس‌العمل است، و سود عمده‌ی یک شرکت مربوط‌تر از رنگ مارک شرکت است. در هر دو تعریف مسئله، دانش قبلی غیر همه‌جانبه است، اما واضح است که در تمیز دادن ویژگی‌های مربوط از غیر مربوط کاراست.

یادگیری استقرایی	یادگیری تحلیلی
هدف: فرضیه‌ای که با داده‌ها تطابق داشته باشد	فرضیه‌ای که با تئوری قلمرو تطابق داشته باشد
توجیه: توجیه آماری	توجیه استنتاجی
مزیت: نیاز به دانش قبلی زیادی ندارد	نیاز به داده‌های آموزشی اندکی دارد
ضعف: داده‌های اندک، بایاس غلت	تئوری قلمروی غلت



شکل ۱۲،۱ طیفی از کارهای یادگیری.

در یک طرف طیف، دانش قبلی‌ای وجود ندارد و متدهای کاملاً استقرایی با پیچیدگی نمونه‌ای بالا لازم است. در طرف دیگر طیف، تئوری قلمروی کامل موجود است، که استفاده از روش‌های تحلیلی محض مثل Prolog-EBG را ممکن می‌کند. مسئله‌های کاربردی معمولاً جایی در میان این طیف قرار دارند. سؤال مطرح در این فصل این است که "چه نوع الگوریتم‌هایی می‌توانند از دانش قبلی تقریبی به همراه داده‌های موجود برای ایجاد برای فرضیه تعمیمی استفاده کنند؟". توجه دارید که حتی زمانی که از روش‌ها استقرایی محض استفاده می‌کردیم فرصت انتخاب طراحی بر پایه‌ی دانش قبلی هدف یادگیری داشتیم. برای مثال، زمانی که از Backpropagation برای مسائلی چون تشخیص گفتار^۱ استفاده می‌کردیم انتخاب‌هایی در طول طراحی ایجاد می‌شد، انتخاب‌هایی نظیر نحوه‌ی کد سازی ورودی و خروجی تابع خطایی که با شیب نزول مینیمم می‌شود، تعداد واحدهای پنهان، پیکربندی یا ساختار شبکه^۲، ضریب یادگیری و تکانه^۳ و ... بود. در این انتخاب‌ها طراح انسانی می‌تواند دانش مربوطه‌ی یادگیری را در الگوریتم یادگیری وارد می‌کند^۴. با این وجود نتیجه روشی استقرایی محض و نمونه‌ای از Backpropagation خواهد بود که توسط انتخاب‌های طراح برای تشخیص گفتار تخصصی شده است. اما در اینجا علاقه‌ی ما به چیز دیگری است. علاقه‌ی ما به سیستم‌هایی است که دانش قبلی را به عنوان ورودی صریح مثل داده‌های آموزشی را به دریافت می‌کنند، بنابراین این سیستم‌ها با وجود دریافت دانش قلمروی‌ای^۵ سیستم‌هایی کلی باقی می‌مانند. به طور خلاصه، علاقه‌ی ما در اینجا به الگوریتم‌های مستقل از قلمرو^۶ است که از ورودی دانش قلمروی‌ای استفاده می‌کنند.

از چه معیاری باید برای مقایسه‌ی روش‌های مختلف و ترکیب یادگیری استقرایی و تحلیلی استفاده کنیم؟ با دانستن اینکه در کل یادگیر کیفیت تئوری قلمرو یا داده‌های آموزشی را نمی‌داند، پس بیشتر علاقه‌ی ما به سوی متدهای کلی است که می‌توانند بدون توجه به مکان مسئله در طیف مذکور شکل ۱۲،۱ کار کنند. تعدادی از ویژگی‌های خاصی که علاقه داریم چنین روش یادگیری‌ای داشته باشد در زیر آمده است:

بدون تئوری قلمرو، چنین الگوریتمی حداقل باید کارایی در حد روش‌های استقرایی محض داشته باشد.

با داشتن تئوری قلمرو کامل، باید حداقل کارایی‌ای در حد روش‌های تحلیلی محض داشته باشد.

با داشتن تئوری قلمرو ناکامل و داده‌های آموزشی ناکامل، باید کارایی بهتر از روش‌های استقرایی محض و روش‌های تحلیلی محض داشته باشد.

باید الگوریتم با سطح نامعلومی از خطا داده‌های آموزشی تطبیق داشته باشد.

^۱ speech recognition

^۲ topology

^۳ momentum

^۴ task-specific knowledge

^۵ domain-specific knowledge

^۶ domain-independent algorithms

باید الگوریتم با سطح نامعلومی از خطا در تئوری قلمرو تطبیق داشته باشد.

توجه دارید که این لیست خواص در حالت ایده آل است. برای مثال، تطبیق خطا با داده‌های آموزشی حتی برای روش‌های آماری بدون داشتن کوچک‌ترین دانش قبلی یا فرضی مبنی بر توزیع خطا مشکل‌زاست. ترکیب یادگیری استقرایی و تحلیلی هنوز در معرض تحقیق و بررسی است. با وجود اینکه لیست بالا خواصی است که می‌خواهیم الگوریتم‌مان داشته باشد، هنوز الگوریتم‌هایی ایجاد نشده که تمامی این قیود را در حالت کلی داشته باشند.

قسمت بعد، بحثی دقیق‌تر از مسائل ترکیبی استقرایی تحلیلی را در بر می‌گیرد. زیر قسمت‌های این قسمت سه روش مختلف برای ترکیب دانش قبلی تقریبی و داده‌های آموزشی موجود را برای کنترل جستجوی یادگیر به سمت فرضیه‌ی مطلوب ارائه می‌کنند. اثبات می‌شود که هر یک از این سه کارایی‌ای بهتر از روش‌های استقرایی محض در قلمروهای مختلف دارند. برای مقایسه، از مثالی برای تصور سه روش استفاده خواهیم کرد.

۱۲,۲ روش یادگیری استقرایی تحلیلی

۱۲,۲,۱ مسئله یادگیری

مسئله یادگیری بحث شده در این فصل را می‌توان به صورت زیر خلاصه کرد:

ورودی:

مجموعه‌ی نمونه‌های آموزشی D ، که ممکن است خطا داشته باشد.

تئوری قلمرو B ، که ممکن است خطا داشته باشد.

فضای از فرضیه‌ای ممکن H

خروجی:

فرضیه‌ای که بهترین تطابق با نمونه‌های آموزشی و تئوری قلمرو دارد.

منظور دقیق از "فرضیه‌ای که بهترین تطابق با نمونه‌های آموزشی و تئوری قلمرو دارد" چیست؟ در کل، آیا فرضیه‌ای را که کمی بیشتر تطابق بر روی نمونه‌های آموزشی دارد و کمی کمتر تطابق بر روی تئوری قلمرو دارد را ترجیح می‌دهیم یا بلعکس؟ می‌توان به طور دقیق‌تر با تعریف معیارهای خطای فرضیه بر اساس تئوری قلمرو و داده‌های آموزشی این ابهام را با تعریف بر طرف کرد. با توجه به آنچه در فصل ۵ گفته شد $error_D(h)$ نسبت نمونه‌هایی که توسط h غلت دسته‌بندی می‌شوند است. بیایید $error_B(h)$ را احتمال اینکه دسته‌بندی نمونه‌ای تصادفی با h با تئوری قلمروی B تطابق نداشته باشد تعریف کنیم. حال می‌توان فرضیه‌ی خروجی مطلوب را با توجه به این خطاهای تعریف شده تعیین کرد. برای مثال، می‌توان فرضیه‌ای را ترکیب خطی‌ای از این دو معیار را مینیمم می‌کند را مطلوب قرار داد.

$$\arg \min_{h \in H} k_D error_D(h) + k_B error_B(h)$$

با وجود اینکه چنین تعبیری در اولین برخورد این عبارت به ذهن می‌رسد اما هنوز مشخص نیست که مقادیر k_B و k_D (میزان اهمیت نسبی تطابق با داده‌های آموزشی به نسبت تطابق با تئوری قلمرو) چه مقداری دارند. اگر تئوری قلمرو بسیار ضعیفی داشته باشیم و تعداد داده‌های

آموزشی زیاد و قابل اعتماد باشد بهتر است که تأثیر $error_D(h)$ بیشتر باشد. در مقابل اگر تئوری قلمرو قوی باشد و نمونه‌های اندک و پر خطا داشته باشیم بهترین نتیجه با افزایش مقدار نظیر $error_B(h)$ به دست خواهد آمد. البته اگر یادگیر در حالت کلی کیفیت تئوری قلمرو و داده‌های آموزشی را نداند، چگونگی وزن دهی این دو خطا نامشخص باقی خواهد ماند.

جواب دیگر سؤال چگونگی وزن دهی دانش قبلی و داده‌ها روش بیزی است. با توجه به آنچه در فصل ۶ گفته شد، قضیه‌ی بیز چگونگی محاسبه‌ی احتمال ثانویه‌ی $P(h|D)$ را برای فرضیه‌ی h و داده‌های آموزشی D بیان می‌کند. در کل، قضیه‌ی بیز احتمال ثانویه را بر اساس داده‌های مشاهده شده‌ی D و دانش قبلی‌ای در غالب $P(h)$ و $P(D)$ و $P(D|h)$ محاسبه می‌کند. پس می‌توان $P(h)$ و $P(D)$ و $P(D|h)$ را به صورت فرمی از دانش قبلی یا تئوری قلمرو دانست و می‌توان قضیه‌ی بیز را متدی برای وزن دهی این تئوری قلمرو و داده‌های مشاهده شده‌ی D برای تعیین احتمال ثانویه‌ی $P(h|D)$ برای h دانست. نگاه بیزی نگاهی است که باید برای انتخاب فرضیه‌ای که احتمال ثانویه‌اش بیشتر است داشت و قضیه‌ی بیز نیز متدی مناسب برای وزن دهی سهم هر یک از دو عامل دانش قبلی و داده‌های مشاهده شده است. متأسفانه، قضیه‌ی بیز به طور ضمنی فرض می‌کند که دانش قبلی درباره‌ی $P(h)$ و $P(D)$ و $P(D|h)$ کامل است. در حالی که این کمیت‌ها فقط به صورت غیر کامل^۷ در دسترس‌اند، قضیه بیز به تنهایی روشی برای ترکیب آن‌ها با داده‌های مشاهده شده ارائه نمی‌کند. (یکی از روش‌های ممکن در چنین شرایطی فرض توزیع احتمال اولیه بر روی خود مقادیر $P(h)$ و $P(D)$ و $P(D|h)$ و محاسبه‌ی مقدار امید $P(h|D)$ است. با این وجود، این روش نیاز به دانش اضافی در مورد توزیع اولیه‌ی $P(h)$ و $P(D)$ و $P(D|h)$ دارد، پس در حالت کلی این روش کارآمد نیست).

در قسمت‌های بعدی باز هم به سؤال مفهوم "متناسب‌ترین فرضیه" در طی بررسی الگوریتم‌های خاص خواهیم پرداخت. اما در حال حاضر، می‌گوییم که مسئله‌ی یادگیری، مینیمم کردن معیار ترکیبی‌ای از خطای فرضیه بر اساس داده‌ها و تئوری قلمرو است.

۱۲.۲.۲ جستجوی فضای فرضیه‌ای

چگونه می‌توان تئوری قلمرو و داده‌های آموزشی را به بهترین وجه ترکیب کرد تا بتوان جستجویی برای فرضیه‌ای قابل قبول ترتیب داد؟ این سؤال در یادگیری ماشین سؤالی بدون جواب باقی‌مانده است. این فصل، چند روش پیشنهادی را بررسی خواهیم کرد. این روش‌ها اغلب تعمیم روش‌های استقرایی‌ای بررسی شده در فصول گذشته (مثل Backpropagation و FOIL) هستند.

یکی از راه‌های درک محدوده‌ی روش‌های مختلف، بازگشت به دید یادگیری به عنوان جستجویی در میان فضای فرضیه‌ای است. ما اکثر متدهای یادگیری را به عنوان الگوریتم برای جستجو فضای فرضیه‌ای H در نظر می‌گیریم و آن‌ها را با فضای فرضیه‌ای‌ای که جستجو می‌کنند توصیف می‌کنیم، فرضیه اولیه که جستجو با آن آغاز می‌شود را h_0 در نظر بگیرید و مجموعه‌ی عملگرهای جستجو که مرحله‌های جستجو را معین می‌کند را O در نظر بگیرید و معیار هدف G که هدف جستجو را مشخص می‌کند در نظر بگیرید. در این فصل ما به سه روش متفاوت استفاده از دانش اولیه برای تغییر کارایی جستجوی متدهای استقرایی محض را بررسی خواهیم کرد.

استفاده از دانش قبلی برای ایجاد یک فرضیه اولیه که جستجو از آن شروع شود. در این روش تئوری قلمروی B برای ساخت فرضیه‌ی اولیه‌ی h_0 که با B سازگار است به کار می‌رود. سپس از روشی استقرایی با فرضیه‌ی اولیه‌ی h_0 استفاده می‌شود. برای مثال، سیستم KBANN که در زیر توضیح داده شده است سیستمی است که از شبکه‌های عصبی با همین روش استفاده می‌کند. این سیستم از دانش قبلی برای طراحی اتصال‌های واحدها و وزن‌های اولیه‌ی شبکه استفاده می‌کند تا شبکه‌ی اولیه به طور کامل با

^۷ imperfect

تئوری قلمرو سازگار باشد. سپس فرضیه‌ای شبکه‌ای به صورت استقرایی با Backpropagation و داده‌های موجود بازنگری می‌شود. با شروع جستجو از فرضیه‌ای که با تئوری قلمرو سازگار است، به نظر می‌رسد که فرضیه‌ی خروجی بیشتر شبیه تئوری قلمرو باشد.

استفاده از دانش قبلی برای تغییر هدف جستجوی فضای فرضیه‌ای. در این روش، معیار هدف G طوری تغییر می‌یابد که فرضیه‌ی خروجی علاوه بر نمونه‌های آموزشی با تئوری قلمرو نیز متناسب باشد. برای مثال، سیستم EBNN که در زیر توضیح داده خواهد شد از شبکه‌های عصبی با این روش استفاده می‌کند. از آنجایی که یادگیری استقرایی شبکه‌های عصبی از شیب نزول برای مینیم کردن خطای مربعی شبکه بر روی داده‌های آموزشی استفاده می‌کند، EBNN از شیب نزول برای مینیم کردن معیاری دیگری استفاده می‌کند. این معیار شامل جمله‌ای اضافی است که خطای شبکه‌ی یاد گرفته شده را بر اساس تئوری قلمرو بیان می‌کند.

استفاده از دانش قبلی برای تغییر مراحل موجود جستجو. در این روش، مجموعه‌ی عملگرهای O توسط تئوری قلمرو تغییر داده می‌شود. برای مثال، سیستم FOCL که در زیر توضیح داده خواهد شد دسته‌ای از horn clause ها را به همین روش یاد می‌گیرد. این روش بر پایه‌ی سیستم FOIL که جستجویی حریصانه بر روی فضای ممکن horn clause ها انجام می‌دهد طراحی شده است. در هر مرحله این سیستم فرضیه‌ی فعلی را با اضافه کردن عبارتی جدید بازنگری می‌کند. FOCL از تئوری قلمرو برای بسط مجموعه عبارات ممکن اضافه شونده به قانون در هنگام بازنگری فرضیه‌ها استفاده می‌کند، و اضافه شدن چندین عبارتی که تئوری قلمرو آن‌ها را استفاده می‌کند در یک مرحله را ممکن می‌سازد. در این روش، FOCL حرکات تک پله‌ای در فضای فرضیه‌ای که در حالت استقرایی الگوریتم به چندین مرحله احتیاج دارد را ممکن می‌سازد. این حرکات بزرگ (macro-moves) می‌توانند به شدت مسیر جستجو را تغییر دهند، بنابراین فرضیه‌ی حاصل سازگار با داده‌ها از فرضیه‌ای که از راه جستجوی استقرایی به دست می‌آید متفاوت خواهد بود.

قسمت‌های بعدی هر یک از این روش‌ها را توضیح خواهند داد.

۱۲,۳ استفاده از دانش قبلی برای مقداردهی اولیه‌ی فرضیه

یکی از روش‌ها، استفاده از دانش قبلی برای مقداردهی اولیه فرضیه به صورتی که با تئوری قلمرو مطابقت داشته باشد و بازنگری فرضیه اولیه برای تطابق با داده‌های آموزشی است. این روش توسط الگوریتم KBANN (شبکه‌های عصبی بر پایه دانش قبلی^۸) به کار گرفته می‌شود. در KBANN ابتدا شبکه‌ی اولیه‌ای ساخته می‌شود که در تمامی نمونه‌ها دسته‌بندی‌اش با دسته‌بندی تئوری قلمرو یکی است. سپس از Backpropagation برای تنظیم وزن‌ها این شبکه‌ی اولیه برای تطابق با نمونه‌های آموزشی استفاده می‌شود.

تشخیص انگیزه‌ی این تکنیک بسیار ساده است: اگر تئوری قلمرو درست باشد، فرضیه‌ی اولیه به درستی تمامی نمونه‌های آموزشی را دسته‌بندی خواهد کرد و دیگر نیازی به بازنگری در آن نخواهد بود. با این وجود اگر فرضیه‌ی اولیه همه‌ی نمونه‌های آموزشی را درست دسته‌بندی نکرده از روشی استقرایی برای بهبود تناسب با داده‌های آموزشی استفاده خواهیم کرد. توجه داشته باشید که در روش استقرایی محض Backpropagation، وزن‌ها معمولاً با مقادیر اتفاقی کوچکی مقداردهی اولیه می‌شوند. مفهوم پشت KBANN این است که اگر تئوری

^۸ Knowledge-Based Artificial Neural Network

قلمرو فقط تقریباً درست باشد، مقداردهی اولیه شبکه به صورتی که با تئوری قلمرو تطابق داشته باشد تقریب شروع بهتری نسبت به مقادیر تصادفی کوچک از تابع هدف است. و متناسباً چنین عملی ما را به سوی تعمیمی با دقت بهتر در فرضیه‌ی نهایی هدایت خواهد کرد.

این روش مقداردهی اولیه‌ی فرضیه در استفاده از تئوری قلمرو در چنین تحقیق مورد بررسی قرار گرفته است، این تحقیق‌ها شامل (Shavlik and Towell 1989) و (Towell and Shavlin 1994)، (Fu 1989,1993) و (Pratt 1993a, 1993b) می‌شود. ما از الگوریتم KBANN که در (Shavlik and Towell 1989) مطرح شده برای توضیح این روش استفاده می‌کنیم.

۱۲,۳,۱ الگوریتم KBANN

الگوریتم KBANN نمونه‌ای از روش مقداردهی اولیه‌ی فرضیه (initialize-the-hypothesis) است. این الگوریتم فرض می‌کند که تئوری قلمرویی که با دسته‌ای از گزاره‌ها بیان شده، یا همان horn clause های غیر بازگشتی بیان شده است. یک horn clause زمانی گزاره‌ای است که متغیری نداشته باشد. ورودی و خروجی KBANN به شرح زیر است:

ورودی:

مجموعه‌ای از نمونه‌های آموزشی

تئوری قلمرویی که از horn clause های گزاره‌ای غیر بازگشتی تشکیل شده است.

خروجی:

شبکه‌ی عصبی‌ای که با نمونه‌های آموزشی تناسب دارد و به سمت تئوری قلمرو بایاس شده است. دو مرحله‌ی الگوریتم KBANN ابتدا ایجاد شبکه‌ای عصبی است که کاملاً با تئوری قلمرو تناسب داشته باشد و دوم استفاده از Backpropagation برای بازنگری این شبکه‌ی اولیه برای متناسب شدن با نمونه‌های آموزشی است. این جزئیات الگوریتم شامل چگونگی ساخت شبکه‌ی اولیه در جدول ۱۲,۲ آمده و بخش ۱۲,۳,۲ توضیح داده شده است.

KBANN(Domain_Theory, Training_Examples)

Domain_Theory: مجموعه‌ای از horn clause های گزاره‌ای غیر بازگشتی.

Training_Examples: مجموعه‌ای از زوج مرتب‌های ورودی خروجی تابع هدف. (به فرم $\langle \text{input}, \text{output} \rangle$).

مرحله‌ی تحلیلی: شبکه‌ی عصبی معادل تئوری قلمرو را ایجاد کن.

برای هر ویژگی دلخواه نمونه‌ها ورودی شبکه‌ای را ایجاد کن.

برای هر horn clause از Domain_Theory، شبکه‌ای اولیه با فرایند زیر ایجاد کن:

ورودی‌های این واحد را به ویژگی‌هایی که بررسی می‌کند متصل کن.

برای هر عبارت غیر منفی حکم، وزن W را به ورودی واحد سیگموئید مربوطه نسبت بده.

برای هر عبارت منفی حکم وزن $-W$ را به ورودی واحد سیگموئید مربوطه نسبت بده.

مقدار آستانه‌ی w_0 برای این واحد را مقدار $-(n-5)W$ قرار بده، در این رابطه n تعداد عبارات غیر منفی حکم است.

ConcavityPointsUp	✓	✓	✓	✓	✓	✓	✓	✓
Expensive	✓		✓			✓		✓
Fragile	✓	✓			✓	✓	✓	✓
HandleOnTop					✓	✓		
HandleOnSide	✓			✓				✓
HasConcavity	✓	✓	✓	✓	✓	✓	✓	✓
HasHandle	✓			✓	✓	✓		✓
Light	✓	✓	✓	✓	✓	✓	✓	✓
MadeOfCeramic	✓				✓	✓	✓	
MadeOfPaper				✓				✓
MadeOfstyrofoam		✓	✓			✓		✓

جدول ۱۲,۳ کار یادگیری مفهوم فنجان.

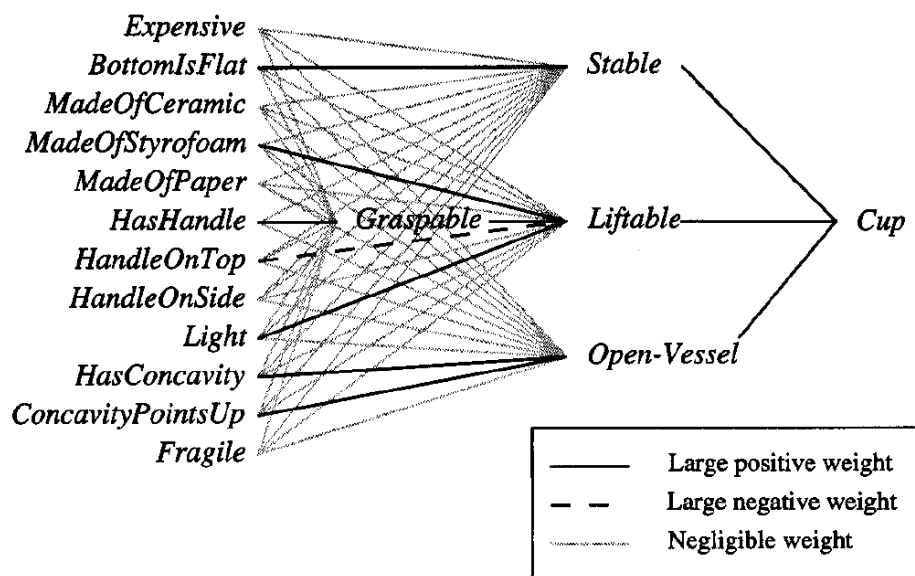
تئوری قلمروی‌ای تقریبی و مجموعه‌ای از نمونه‌های آموزشی برای مفهوم هدف "فنجان".

در مرحله‌ی اول الگوریتم KBANN (سه قدم اول) شبکه‌ای اولیه که با تئوری قلمرو سازگار باشد ساخته می‌شود. برای مثال، شبکه‌ی ساخته شده برای مفهوم "فنجان" در شکل ۱۲,۲ آمده است. در کل شبکه با ایجاد واحد سیگموئید حد آستانه‌ای برای هر $horn\ clause$ تئوری قلمرو ساخته می‌شود. مقادیر ورودی کمتر از 0.5 به منزله‌ی غلت و بزرگ‌تر از 0.5 به منزله‌ی درست در نظر گرفته می‌شود. سپس هر واحد تئوری ساخته می‌شود که مقدارش در صورت درست بودن $horn\ clause$ بزرگ‌تر از 0.5 باشد. برای هر فرض $horn\ clause$ ورودی‌ای متناسب به واحد سیگموئید داده می‌شود. سپس وزن‌های گره‌های سیگموئید تئوری تعیین می‌شود که واحد به صورت AND منطقی عمل کند. در کل، برای هر ورودی متناسب با یک شرط غیر منفی، مقدار وزن W (که مقدار مثبتی است) قرار داده می‌شود. برای هر ورودی متناسب با یک شرط منفی مقدار $-W$ نسبت داده می‌شود. و مقدار آستانه‌ی w_0 نیز $-(n-0.5)W$ نیز در نظر گرفته می‌شود، در این مقدار n تعداد ورودی‌های غیر منفی است. زمانی که مقادیر ۱ یا صفر است، مطمئن خواهیم بود که مجموع وزن‌دارشان به علاوه‌ی w_0 مثبت است (بنابراین خروجی نیز از 0.5 بیشتر خواهد بود) پس فقط و فقط اگر تمامی عبارات شرط مثبت باشند خروجی نیز مثبت خواهد بود. توجه دارید که واحدهای سیگموئید در لایه‌ی ثانویه نیازی نیست که دقیقاً ۱ یا صفر باشند و بحث بالا کاملاً درست نخواهد بود. با این وجود، اگر مقدار W به اندازه‌ی کافی بزرگ انتخاب شود، KBANN می‌تواند به درستی تئوری قلمرو را برای هر شبکه‌ای با هر عمق دلخواه را نمایش دهد. (Towell and Shavlik 1994) در گزارش‌هایشان از استفاده از این الگوریتم از $W=4$ استفاده کرده‌اند.

ورودی هر واحد سیگموئید به ورودی شبکه یا خروجی واحدهای مربوطه متصل است تا گراف وابستگی ویژگی مربوطه‌ی تعریف شده در تئوری قلمرو را نشان دهد. به عنوان آخرین قدم این مرحله تعداد قابل توجهی ورودی به واحد آستانه اضافه می‌شود که وزن‌های نظیرشان تقریباً صفر

است. نقش این ارتباط‌های مذکور، دادن توانایی لازم به شبکه برای یادگیری وابستگی احتمالی ویژگی مربوطه به دیگر ویژگی‌های شبکه است. خطوط پررنگ در شبکه‌ی ۱۲،۲ ارتباط‌هایی با وزن W و خطوط کمرنگ ارتباط‌هایی با وزن تقریباً صفر را نشان می‌دهد. مشهود است که با بزرگ بودن W به اندازه‌ی کافی این شبکه مقادیر نظیر تئوری قلمرو را نشان خواهد داد.

مرحله دوم KBANN (پله‌ی چهارم در جدول ۱۲،۲) استفاده از Backpropagation در بازبینی وزن‌های اولیه‌ی شبکه است. البته اگر تئوری قلمرو و نمونه‌های آموزشی خطایی نداشته باشند، شبکه‌ی اولیه با نمونه‌های آموزشی نیز سازگار خواهد بود. در مثال "فنجان"، چون تئوری قلمرو و داده‌های آموزشی باهم سازگار نیستند، این مرحله وزن‌های شبکه‌ی اولیه را تغییر خواهد داد. شبکه‌ی آموزش داده‌ی مثال "فنجان" در شکل ۱۲،۳ آورده شده است، خطوط توپر بالاترین مقادیر وزن‌ها و خط‌چین بالاترین مقادیر وزن‌های منفی را نشان می‌دهد و خطوط کمرنگ نماینده‌ی وزن‌های قابل صرف‌نظر است. با وجود اینکه شبکه‌ی اولیه چندین نمونه‌ی آموزشی را اشتباه دسته‌بندی می‌کند، شبکه‌ی بازبینی شده‌ی شکل ۱۲،۳ کاملاً تمامی نمونه‌های آموزشی را دسته می‌کند.



شکل ۱۲،۳ حاصل بازبینی استقرایی شبکه‌ی اولیه.

KBANN از نمونه‌های آموزشی برای تغییر وزن‌های شبکه‌ی اولیه ناشی از تئوری قلمرو استفاده می‌کند. توجه دارید که وابستگی *Liftable* به *MadeOfStyrofoam* و *HandleOnTop* در تئوری قلمرو نبود.

مقایسه‌ی وزن‌های اولیه با وزن‌های نهایی شبکه نتایج بسیار جالبی در بر خواهد داشت. همان‌طور که در شکل ۱۲،۳ نیز دیده می‌شود، وابستگی شدید در مرحله‌ی استقرایی کشف می‌شود، مثل وابستگی واحد *Liftable* به ویژگی *MadeOfStyrofoam*. توجه به اینکه گره *Liftable* توسط horn clause ی در تئوری قلمرو تعریف شده‌اند اما توسط Backpropagation به ویژگی دیگری از شبکه وابسته می‌گردد. بعد از آموزش شبکه، این واحد مفهومی متفاوت با مفهوم اولیه‌ی *Liftable* خواهد داشت.

۱۲،۳،۳ نکات

خلاصه، KBANN به صورت تحلیلی شبکه‌ای هم‌ارز با تئوری قلمرو ارائه شده ایجاد می‌کند، سپس در این فرضیه‌ی اولیه را برای تناسب بهتر با نمونه‌های آموزشی تجدیدنظر می‌کند. در این کار، این الگوریتم برای تصحیح عدم تطابق تئوری قلمرو و داده‌های مشاهده شده وزن‌های شبکه را تغییر می‌دهد.

برتری مهم در مورد KBANN بر روش‌های استقرایی محض مثل Backpropagation (که از شبکه‌ای با وزن‌های تصادفی شروع می‌شود) این است که زمانی که تئوری قلمرو نسبتاً درست است تعمیم بهتری نسبت به Backpropagation دارد، این برتری زمانی که نمونه‌های آموزشی خطای زیادی دارد بیشتر دیده خواهد شد. KBANN و دیگر روش‌هایی که از فرضیه ابتدایی کمک می‌گیرند در مسائل واقعی کارایی بهتری نسبت به سیستم‌های استقرایی محض از خود نشان داده‌اند. برای مثال، (Towell 1990) کاربرد KBANN در مسئله‌ای درباره‌ی ساختار مولکولی ژنتیکی را ارائه می‌کند. در این کاربرد هدف یادگیری تشخیص قسمت‌هایی از DNA به نام promoter region است که بر عملکرد ژن تأثیر دارد. در این تحقیق به KBANN تئوری قلمروی‌ای اولیه که از یک محقق ژنتیک^۹ دریافت شده بود به همراه مجموعه‌ای از ۵۳ نمونه‌ی مثبت و ۵۳ نمونه‌ی منفی داده شد. کارایی سیستم با استراتژی leave-one-out مورد بررسی قرار گرفت و سیستم ۱۰۶ بار آموزش داده شد. در هر بار اجرای حلقه‌ی KBANN با ۱۰۵ نمونه از ۱۰۶ نمونه آموزش داده شد و بر روی نمونه‌ی باقیمانده تست شد. نسبت خطای^{۱۰} این آزمایش را می‌توان تخمینی از خطای واقعی دانست. KBANN نسبت خطای 4/106 را پیدا کرد در حالی که نسبت خطای Backpropagation، 8/106 شد. نسخه‌های مختلف KBANN توسط (Fu 1993) به کار گرفته شده که وی به خطای 2/106 نیز بر روی همان داده‌ها رسیده است. بنابراین، اثر دانش قبلی در این آزمایش‌ها کاهش قابل توجه نسبت خطا بوده است. نمونه‌های آموزشی این آزمایش در <http://www.ics.uci.edu/~mllearn/MLRepository.html> موجود است.

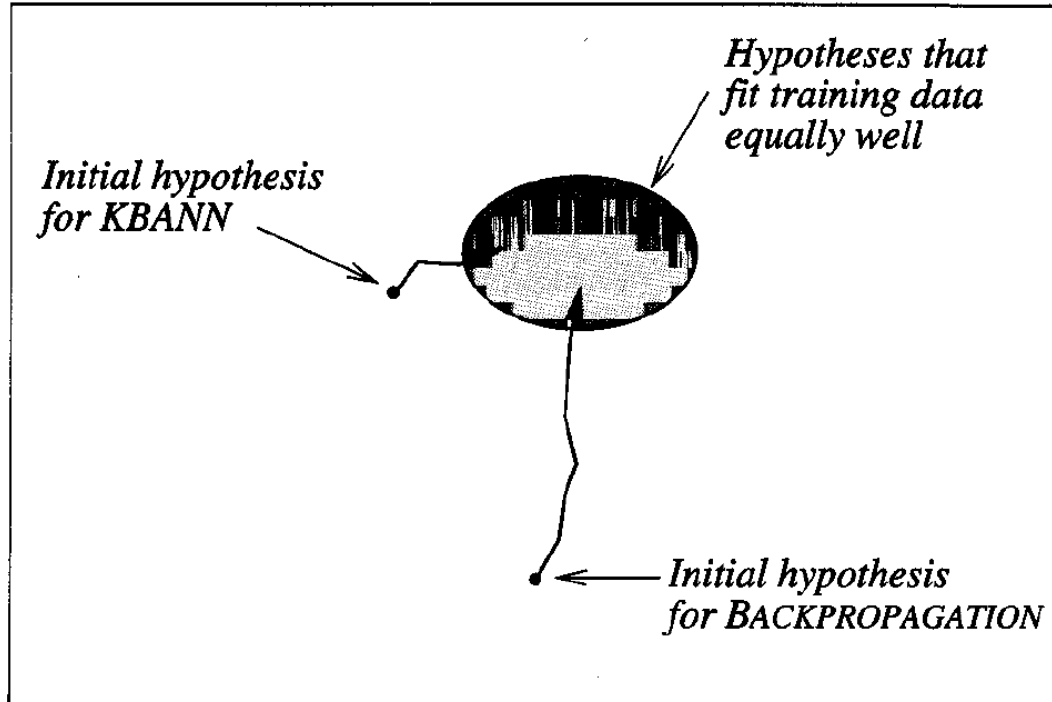
در هر دو (Fu 1993) و (Towell 1990) ذکر شده که دسته قوانین استخراجی از شبکه‌ی آموزش یافته تئوری قلمروی‌ای تجدیدنظر یافته به ما می‌دهند که سازگاری بیشتری با نمونه‌های آموزشی دارد. با وجود اینکه گاهی به دست آوردن دسته قوانین Horn clause از شبکه‌ی یاد گرفته شده ممکن است اما در حالت کلی این کار با مشکلاتی همراه است زیرا که بعضی وزن‌ها معادل horn clause صریحی ندارند. (Craven and Shavlik 1994) و (Craven 1996) متدهای متفاوتی برای استخراج قوانین سمبولیک از شبکه‌های یاد گرفته شده ارائه می‌کنند.

برای پی بردن به اهمیت KBANN بد نیست که تفاوت جستجوی فرضیه‌ای آن را با الگوریتم استقرایی محض Backpropagation مقایسه کنیم. فضای فرضیه‌ای‌ای که این دو روش جستجو می‌کنند یکی است و در شکل ۱۲،۴ نشان داده شده است. همان‌طور که مشاهده می‌کنید، تفاوت اساسی در فرضیه ابتدایی است که از آنجا جستجو آغاز می‌گردد. در شرایطی که چندین فرضیه‌ی مختلف (بردار وزن‌های مختلف) با داده‌ها سازگارند، شرطی که معمولاً زمانی که نمونه‌ها خطا دارند پیش می‌آید، KBANN بیشتر به سمت فرضیه‌ای میل می‌کند که تعمیم روی داده‌ها را مشابه تئوری قلمرو انجام می‌دهد. از طرف دیگر، Backpropagation معمولاً به فرضیه‌هایی میل می‌کند که وزن‌های کوچکی دارند، چنین فرضیه‌هایی مشابه درون‌یابی هموار بین نمونه‌های آموزشی است. به طور خلاصه، KBANN از تئوری خاص قلمرو استفاده کرده تا تعمیم روی نمونه‌ها را بایاس کند در حالی که Backpropagation از قاعده‌ای مستقل از قلمرو برای بایاس کردن شبکه به سمت وزن‌های کوچک‌تر استفاده می‌کند. توجه دارید که در این خلاصه ما از اثر مینیمم‌های موضعی چشم‌پوشی کرده‌ایم.

^۹ geneticist

^{۱۰} error rate

Hypothesis Space



شکل ۱۲,۴ جستجوی فضای فرضیه‌ای KBANN.

KBANN شبکه را متناسب با تئوری قلمرو مقداردهی اولیه می‌کند، در حالی که *backpropagation* شبکه را با مقادیر تصادفی مقداردهی اولیه می‌کند. هر دو سیستم از تغییر وزن‌های کوچک با استفاده از شیب نزول استفاده می‌کنند. زمانی که چندین فرضیه با نمونه‌های آموزشی سازگارند، ناحیه‌ی هاشور زده، KBANN و *Backpropagation* ممکن است به فرضیه‌های متفاوتی ختم شوند، زیرا که نقاط شروع متفاوتی دارند. محدودیت KBANN این حقیقت است که این روش فقط با تئوری قلمروهای گزاره‌ای سازگار است؛ به عبارت دیگر فقط *horn clause* هایی را می‌توان به این روش داد که متغیری در آن‌ها وجود نداشته باشد. البته هنگامی که تئوری قلمرو بسیار پر خطا باشد ممکن است این روش به نتیجه‌ی کاملاً اشتباهی برسد و در این حالت دقت تعمیم این روش از *Backpropagation* نیز کمتر خواهد بود. با این وجود، این روش و الگوریتم‌های مربوطه در مسائل کاربردی زیادی مفید واقع شده‌اند.

KBANN روش مقداردهی ابتدایی فرضیه‌ی را برای ترکیب یادگیری تحلیلی و یادگیری استقرایی را توصیف می‌کند. دیگر نمونه‌های این روش در (Fu 1993)، (Gallant 1988)، (Bradshaw 1989)، (Yang and Bhargava 1990) و (Lacher 1991) آمده است. این روش‌ها در تکنیک ساخت شبکه‌ی اولیه، کاربرد *Backpropagation* برای تنظیم وزن‌ها و در متد استخراج توصیف سمبولیک از شبکه‌های بازنگری شده با هم متفاوت‌اند. (Pratt 1993a, 1993b) روشی از فرضیه‌ی ابتدایی را معرفی می‌کند که در آن دانش قبلی از شبکه‌ی آموزش یافته‌ای برای کاری مشابه به دست می‌آید و از تئوری قلمرو سمبولیک استفاده نمی‌شود. متدهای یادگیری مقادیر شبکه‌ی باور بیزی‌ای که در قسمت ۶,۱۱ معرفی شده‌اند، را می‌توان دانش قبلی برای مقداردهی اولیه فرضیه قرار داد. در این حالت دانش قبلی متناسب با مجموعه‌ای از فرض‌های استقلال خواهد بود که ساختار گرافی شبکه‌ی بیزی را مشخص می‌کند، جداول احتمالات این مقادیر از داده‌های آموزشی استخراج می‌شود.

۱۲,۴ استفاده از دانش قبلی برای تغییر هدف جستجو

روش بالا با جستجوی شیب نزول و با شروع از فرضیه‌ای که با تئوری قلمرو سازگار است شروع می‌شود و در ادامه برای تطابق با داده‌های آموزشی این فرضیه تغییر داده می‌شود. روش جایگزینی برای استفاده از دانش قبلی وجود دارد، آن هم اضافه کردن اطلاعات دانش قبلی در معیار خطای شیب نزول است، پس شبکه مجبور خواهد بود تا به تابعی ترکیبی از نمونه‌های آموزشی و تئوری قلمرو میل کند. در این بخش، استفاده از تئوری قلمرو بدین صورت را بررسی خواهیم کرد. در کل، به دانش قبلی به دید مشتق معلوم تابع هدف نگاه می‌کنیم. انواع خاصی از دانش قبلی را می‌توان به صورت طبیعی با این فرم نشان داد. برای مثال، در آموزش شبکه‌ی عصبی برای تشخیص کاراکترهای دست‌نویس می‌توانیم مشتقات خاصی از تابع هدف را برای بیان دانش قبلی اینکه "کاراکتر به انتقال و دوران تصویر وابسته نیست" استفاده کنیم.

در زیر به الگوریتم TangentProp می‌پردازیم که شبکه‌ای عصبی را با توجه به مقادیر نمونه‌های آموزشی و مشتقات آموزشی آموزش می‌دهد. قسمت ۱۲,۴,۴ چگونگی به دست آوردن این مشتقات از تئوری قلمروی‌ای مشابه آنچه در مثال "فنجان" مورد استفاده قرار گرفت را توضیح خواهد داد (قسمت ۱۲,۳). در کل، این قسمت چگونگی ایجاد توضیحات برای استخراج مشتقات آموزشی از تک نمونه‌های آموزشی در الگوریتم EBNN برای استفاده در TangentProp را توضیح خواهد داد. TangentProp و EBNN نشان داده‌اند که در بسیاری از تئوری قلمروها شامل تشخیص کاراکترهای دست‌نویس و تشخیص اشیاء و درک و کنترل ربات از روش‌های استقرایی محض کارایی بهتری دارند.

۱۲,۴,۱ الگوریتم Tangentprop

الگوریتم Tangentprop (Simard 1992) اطلاعات تئوری قلمرو را با مشتقاتی از تابع هدف نسبت به تغییر ورودی‌هایش بیان می‌کند. کار یادگیری‌ای را با فضای نمونه‌ای X و تابع هدف f در نظر بگیرید. تا الآن فرض بر این بود که نمونه‌های آموزشی به صورت زوج مرتب‌های $\langle x_i, f(x_i) \rangle$ بیان می‌شود که در آن x_i یک نمونه و $f(x_i)$ مقدار یادگیری آن است. الگوریتم TangentProp فرض می‌کند که علاوه بر مقدار تابع مقدار مشتقات تابع هدف نیز در نمونه‌های آموزشی آورده شده است. برای مثال، اگر نمونه‌ی x_i با یک مقدار واقعی توصیف شود هر نمونه‌ی آموزشی ممکن است به صورت $\langle x_i, f(x_i), \frac{\partial f(x)}{\partial x} |_{x_i} \rangle$ بیان شود. که در این نمونه‌ی آموزشی $\frac{\partial f(x)}{\partial x} |_{x_i}$ مشتق تابع هدف f نسبت به x در نقطه‌ی $x = x_i$ است.

برای ایجاد شهود مزیت داشتن مقادیر مشتق علاوه بر مقادیر تابع، کار یادگیری ساده‌ی آمده در شکل ۱۲,۵ را در نظر بگیرید. در این شکل هدف یادگیری تابع هدف f نمودار سمت چپ با استفاده از سه نمونه‌ی آموزشی $\langle x_1, f(x_1) \rangle$, $\langle x_2, f(x_2) \rangle$ و $\langle x_3, f(x_3) \rangle$ است. با داشتن این سه نمونه‌ی آموزشی می‌توان انتظار داشت که الگوریتم Backpropagation تابع فرضیه‌ای هموار مثل تابع g را که در شکل وسط آورده شده‌یاد بگیرد. شکل سمت راست اثر داشتن مشتقات آموزشی یا شیب‌ها را به عنوان اطلاعات اضافی برای هر نمونه‌ی آموزشی (مثل $\langle x_1, f(x_1), \frac{\partial f(x)}{\partial x} |_{x_1} \rangle$) را نشان می‌دهد. با تناسب یادگیر هم با مقادیر آموزشی $f(x_i)$ و هم مشتقات آموزشی $\frac{\partial f(x)}{\partial x} |_{x_i}$ یادگیر شانس بیشتری در تعمیم روی داده‌های آموزشی خواهد داشت. به طور خلاصه اثر در نظر گرفتن مشتقات آموزشی کاهش بایاس زبانی^{۱۱} الگوریتم Backpropagation به سوی تعمیم هموار بین نقاط خواهد بود و در مقابل اطلاعات ورودی محض مشتقات آموزشی جایگزین خواهند شد. فرضیه‌ی حاصل h در راست‌ترین تصویر شکل تخمینی با دقت بسیاری بهتری از تابع حقیقی f دارد.

^{۱۱} syntactic

در نمونه‌ی بالا فقط انواع ساده‌ی مشتق تابع هدف در نظر گرفته شده است. در واقع، **Tangentprop** فقط مشتقات آموزشی‌ای را دریافت می‌کند که بر حسب تبدیلات ورودی X باشند. برای مثال، فرض کنید، هدف یادگیری تشخیص کاراکترهای دست‌نویس است. فرض کنید که ورودی X متناسب با تصویر یک کاراکتر باشد و هدف نیز دسته‌بندی درست این کاراکتر باشد. در این کار ممکن است علاقه داشته باشیم که یادگیر بداند که "تابع هدف به چرخش‌های کوچک کاراکتر در تصویر حساس نیست". برای بیان این دانش قبلی به یادگیر، ابتدا یک تبدیل مثل $s(\alpha, X)$ را تعریف می‌کنیم که تصویر X را α درجه دوران می‌دهد. حال می‌توانیم فرضمان را درباره‌ی دوران را با این عبارت که مشتق تابع هدف نسبت به این دوران صفر است بیان کنیم (بدین معنا که دسته‌بندی کاراکتر با چرخش عوض نمی‌شود). به عبارت دیگر، می‌توان مشتق آموزشی زیر را برای هر نمونه‌ی آموزشی x_i در نظر گرفت،

$$\frac{\partial f(s(\alpha, x_i))}{\partial \alpha} = 0$$

در این رابطه f تابع هدف و $s(\alpha, x_i)$ نیز تصویر حاصل از اعمال تبدیل S بر روی تصویر x_i است.

اما **TangentProp** چگونه از چنین مشتقات آموزشی برای تغییر مناسب وزن‌های شبکه عصبی استفاده می‌کند؟ در **TangentProp** این مشتقات آموزشی در تابع خطایی که توسط **Backpropagation** مینیمم می‌شود قرار می‌گیرد. با توجه به آنچه در فصل ۴ درباره‌ی الگوریتم **Backpropagation** گفته شد، این الگوریتم از شیب نزول برای مینیمم کردن مجموع خطاهای مربعی استفاده می‌کند،

$$E = \sum_i (f(x_i) - \hat{f}(x_i))^2$$

در این رابطه x_i نشان‌دهنده‌ی آمین نمونه‌ی آموزشی است و f نیز خود تابع هدف است و \hat{f} تابع یاد گرفته شده توسط شبکه‌ی عصبی است.

در **TangentProp** جمله‌ای اضافی به تابع خطا اضافه می‌شود تا اختلاف بین مشتقات آموزشی و مشتقات تابع یاد گرفته شده را در شبکه‌ی عصبی یا \hat{f} نیز تأثیر دهد. در کل **TangentProp** تبدیلات چندگانه را قبول می‌کند (برای مثال، ممکن است بخواهیم به طور همزمان عدم تأثیر چرخش و انتقال کاراکتر را به یادگیر نشان دهیم). هر تبدیل باید به فرم $S_j(\alpha, X)$ باشد که در آن α پارامتری پیوسته و S_j نیز مشتق‌پذیر است و داریم $S_j(0, X) = X$ (برای مثال، برای چرخش 0 درجه تبدیل همانی خواهد بود). برای هر تبدیل با فرم $S_j(\alpha, X)$ ، **Tangentprop** خطای مربعی بین مشتق آموزشی و مقدار واقعی مشتق شبکه‌ی یاد گرفته شده را در نظر می‌گیرد. خطای تغییر یافته را می‌توان به فرم زیر بیان کرد،

$$E = \sum_i \left[(f(x_i) - \hat{f}(x_i))^2 + \mu \sum_j \left(\frac{\partial f(S_j(\alpha, x_i))}{\partial \alpha} - \frac{\partial \hat{f}(S_j(\alpha, x_i))}{\partial \alpha} \right)^2 \right]_{\alpha=0} \quad (12.1)$$

در این رابطه μ ثابتی است که توسط کاربر تعیین می‌شود که اهمیت نسبی تناسب با مشتقات آموزشی را در مقابل اهمیت تناسب با مقادیر آموزشی را بیان می‌کند. توجه دارید که جمله‌ی اول این تعریف همان تعریف اصلی خطای مربعی برای مقادیر آموزشی است و جمله دوم خطای مربعی مقادیر مشتقات آموزشی است.

(Simard 1992) قانون شیب نزول را برای مینیم کردن تابع خطای تعمیم یافته‌ی E ارائه می‌کند. این تابع خطا را نیز می‌توان به همان روش فصل ۴ برای استخراج قانون ساده‌ی **backpropagation** استخراج کرد.

۱۲،۴،۲ مثالی توصیفی

(Simard 1992) نتایج به دست آمده از مقایسه‌ی تعمیم دقت **TangentProp** و متد استقرایی محض **Backpropagation** را برای مسئله‌ی تشخیص کاراکترهای دست‌نویس مشخص می‌کند. به صورت خاص‌تر، هدف در این یادگیری دسته‌بندی تصاویر حاوی تک رقم‌های ۰ تا ۹ است. در یک آزمایش هر دو الگوریتم **TangentProp** و **Backpropagation** با مجموعه‌ای از نمونه‌های آموزشی با اندازه‌های متفاوت آموزش داده شدند و سپس کارایی خروجی بر روی مجموعه‌ای از ۱۶۰ نمونه بررسی شد. دانش قبلی در نظر گرفته شده در **TangentProp** این حقیقت بود که دسته‌بندی اعداد به انتقال افقی و عمودی تصویر وابسته نیست (مثلاً مشتق تابع هدف نسبت به این تبدیلات صفر در نظر گرفته شده است). نتایج حاصل از این آزمایش در شکل ۱۲،۴ نشان داده شده است، این نتایج نشان می‌دهد که **TangentProp** با استفاده از دانش قبلی در تعمیم روی نمونه‌ها کارایی بیشتری از روش استقرایی محض **Backpropagation** دارد.

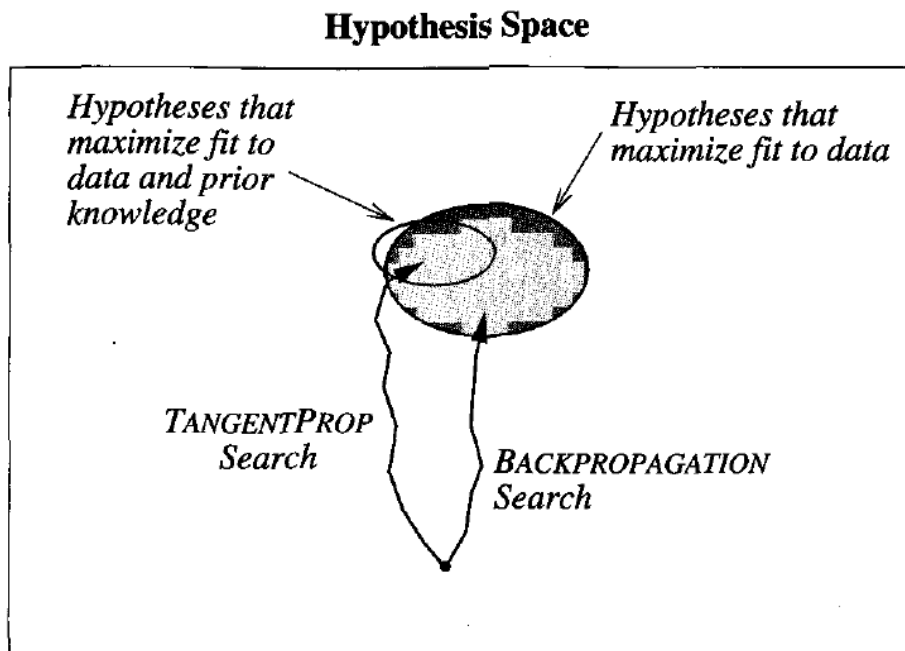
۱۲،۴،۳ نکات

خلاصه، **TangentProp** از دانش قبلی به فرم مشتقات انتظاری تابع هدف نسبت به تبدیلات استفاده می‌کند. این روش دانش قبلی را با داده‌های آموزشی مشاهده شده ترکیب می‌کند تا با مینیمم شدن این تابع هدف خطای شبکه هم برای مقادیر آموزشی و هم برای مقادیر مشتقات آموزشی مینیمم شود (شبکه هم با مقادیر آموزشی و هم دانش قبلی سازگار شود). مقدار ثابت μ درجه‌ی نسبی اهمیت این دو خطا را در خطای کل تعیین می‌کند. رفتار این الگوریتم به مقدار μ حساس است و μ باید توسط طراح تعیین شود.

با وجود اینکه **TangentProp** در ترکیب دانش قبلی و داده‌های آموزشی برای هدایت یادگیری شبکه‌ی عصبی موفق است اما نسبت به خطای دانش قبلی آسیب‌پذیر است. حالتی را در نظر بگیرید که دانش قبلی اشتباه باشد، به عبارت دیگر، مشتقات آموزشی ورودی مشتقات تابع هدف واقعی را نشان ندهند. در چنین شرایطی، الگوریتم سعی می‌کند تا به سمت مشتقات آموزشی اشتباه همگرا شود، و در نتیجه ممکن است تعمیم الگوریتم از **Backpropagation** نیز ضعیف‌تر شود. اگر در حالت پیشرفته‌تر میزان درجه‌ی خطای مشتقات آموزشی معلوم باشد می‌توان از چنین اطلاعاتی برای تعیین ثابت μ استفاده کرد و اهمیت نسبی مشتقات آموزشی را نسبت به مقادیر آموزشی مشخص کرد. اما در کل معمولاً چنین اطلاعاتی در حالت کلی در دسترس نیست. در قسمت بعد به الگوریتم **EBNN** خواهیم پرداخت که به طور خودکار مقدار μ را برای هر نمونه‌ی آموزشی به طور مجزا و بر حسب احتمال غلت بودن دانش قبلی تعیین می‌کند.

مقایسه‌ی جستجوی فضای فرضیه‌ای (فضای وزن‌ها) الگوریتم **TangentProp** و **KBANN** و **Backpropagation** نتایج جالبی در بر دارد. **TangentProp** دانش قبلی را در جستجوی فرضیه‌ای تأثیر می‌دهد و هدف جستجوی شیب نزول را با آن تغییر می‌دهد. این تغییر متناسب با تغییر هدف جستجوی فضای فرضیه‌ای است، مثالی شهودی از این تغییر در شکل ۱۲،۶ آمده است. مشابه **Backpropagation** (و نه مشابه **KBANN**)، **TangentProp** جستجوی خود را با شبکه‌ای با وزن‌های کوچک و تصادفی آغاز می‌کند. با این وجود قانون آموزش شیب نزول این الگوریتم با **Backpropagation** تفاوت دارد و این تفاوت باعث می‌شود که این روش به فرضیه‌ی انتهایی دیگری میل کند. همان‌طور که در شکل نیز نشان داده شده است، مجموعه‌ای از فرضیه‌ها که تابع هدف **TangentProp** را مینیمم می‌کند ممکن است با مجموعه فرضیه‌هایی که هدف **Backpropagation** را مینیمم می‌کند متفاوت باشد. مخصوصاً زمانی که نمونه‌های آموزشی و دانش قبلی هر دو درست باشند، و همچنین بتوان تابع هدف را با شبکه‌ی **ANN** در نظر گرفته شده نمایش داد، مجموعه‌ی بردارهای وزنی که هدف

TangentProp را راضی می‌کند زیرمجموعه‌ای از مجموعه بردارهای راضی‌کننده‌ی هدف backpropagation خواهد بود. تفاوت این دو مجموعه فرضیه انتهایی، مجموعه‌ی فرضیه‌های غلتی است که backpropagation در نظر گرفته، اما TangentProp آن‌ها را بر اساس دانش قبلی‌اش رد می‌کند.



شکل ۱۲.۶ جستجوی فضای فرضیه‌ای TangentProp.

TangentProp درست مشابه Backpropagation شبکه را مقادیر کوچک تصادفی مقداردهی اولیه می‌کند، با این وجود، این الگوریتم از تابع خطای متفاوتی برای هدایت جستجوی شیب نزول استفاده می‌کند. خطای مورد استفاده‌ی TangentProp هم خطای مقادیر آموزشی و هم خطای مشتقات آموزشی که توسط دانش قبلی به ما داده می‌شود را در نظر می‌گیرد.

توجه دارید که جایگزین دیگر برای متناسب‌سازی شبکه با مشتقات آموزشی تابع هدف اضافه کردن نمونه‌های آموزشی جدید نزدیک نمونه‌های آموزشی قبلی است، این مقادیر را می‌توان از مشتقات آموزشی و مقادیر آموزشی نمونه‌ها تخمین زد. برای مثال، در مثال تشخیص تصویر کاراکترها می‌توان تصاویر را به اندازه‌های نسبتاً کوچک انتقال داد و نمونه‌های آموزشی جدید به دست آورد و آن را با همان دسته‌بندی نمونه‌ی اصلی به شبکه داد. می‌توان انتظار داشت که این نمونه‌های تخمینی با روش Backpropagation به فرضیه‌ای مشابه فرضیه‌ی خروجی TangentProp برسد. (Simard 1992) خطای به دست آمده از این دو روش را در چنین حالتی بررسی کرده و به این نتیجه می‌رسد که با این حال کارایی TangentProp به نسبت بیشتر از روش تخمینی است. جالب است بدانید که سیستم ALVINN که برای هدایت اتومبیل طراحی شده بود (به فصل ۴ مراجعه کنید)، روشی مشابه روش تخمین مقادیر آموزشی جدید را به کار برده است. در این سیستم از دانش قبلی اینکه انتقال افقی تصویر متناسب با هدایت فرمان است استفاده کرده تا نمونه‌های جدیدی تخمین زده و نمونه‌های آموزشی مشاهده شده را افزایش دهد.

۱۲,۴,۴ الگوریتم EBNN

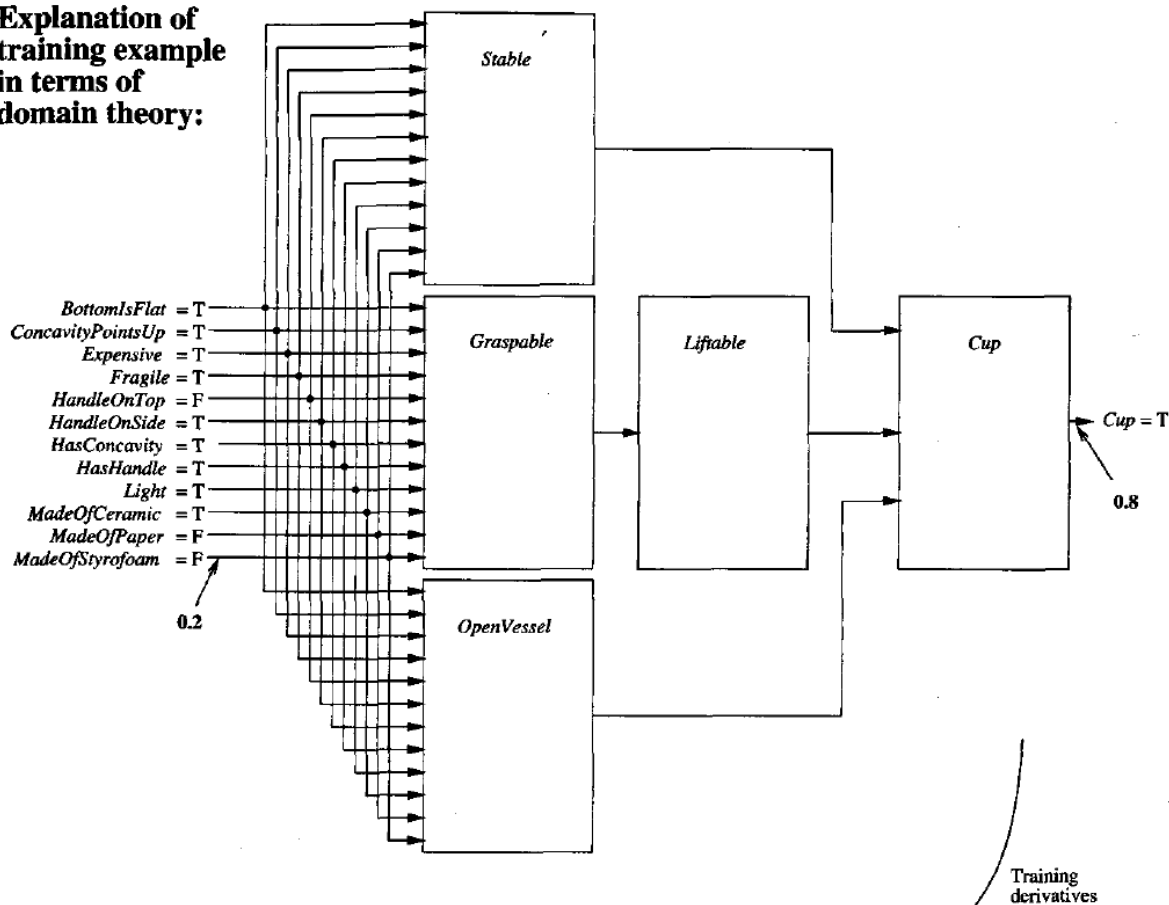
الگوریتم EBNN یا یادگیری توضیحی شبکه‌ی عصبی^{۱۲} (Mitchell and Thrun 1993a; Thrun 1996) از دو نظر بر پایه‌ی الگوریتم TangentProp ساخته شده است. ابتدا اینکه به جای اینکه به کاربر برای مشتقات آموزشی وابسته باشد، EBNN خودش مشتقات آموزشی را برای هر نمونه‌ی مشاهده شده محاسبه می‌کند. این مشتقات آموزشی بر اساس توضیح هر نمونه‌ی آموزشی بر اساس تئوری قلمرو موجود محاسبه و از توضیحات استخراج می‌شوند. دوم اینکه EBNN مشکل چگونگی وزن دهی اهمیت نسبی مؤلفه‌های استقرایی و تحلیلی را حل می‌کند (برای مثال، چگونگی انتخاب پارامتر μ در رابطه‌ی ۱۲,۱ را تعیین می‌کند). مقدار μ برای هر یک از نمونه‌های آموزشی به طور مستقل انتخاب می‌شود، این میزان بسته به این که دقت تئوری قلمرو در تخمین مقدار آموزشی چقدر است انتخاب می‌شود. بنابراین مؤلفه‌ی تحلیلی یادگیری برای نمونه‌هایی که توسط تئوری قلمرو درست دسته‌بندی می‌شوند مؤکد و در نمونه‌هایی که توضیح ضعیف است کم‌رنگ می‌شود.

ورودی‌های الگوریتم EBNN شامل (۱) مجموعه‌ای از نمونه‌های آموزشی به فرم $\langle x_i, f(x_i) \rangle$ می‌شود که هیچ مشتق آموزشی‌ای در آن ارائه نشده است، (۲) تئوری قلمروی‌ای مشابه تئوری قلمروهایی که در یادگیری توضیحی از آن استفاده کردیم (به فصل ۱۱ مراجعه کنید) با این تفاوت که در اینجا تئوری قلمرو به جای دسته‌ای *horn clause* ها با شبکه عصبی آموزش یافته‌ای نمایش داده می‌شود. خروجی EBNN شبکه‌ای عصبی است که تابع هدف f را تخمین می‌زند. این شبکه‌ی یاد گرفته شده هم طوری آموزش دیده است که هم متناسب با نمونه‌های آموزشی $\langle x_i, f(x_i) \rangle$ باشد و هم متناسب با مشتقات نتیجه‌گیری شده از تئوری قلمرو است. متناسب بود با نمونه‌های آموزشی $\langle x_i, f(x_i) \rangle$ مؤلفه‌ی استقرایی یادگیری و متناسب بودن با مشتقات آموزشی نتیجه‌گیری شده از تئوری قلمرو مؤلفه‌ی تحلیلی یادگیری است.

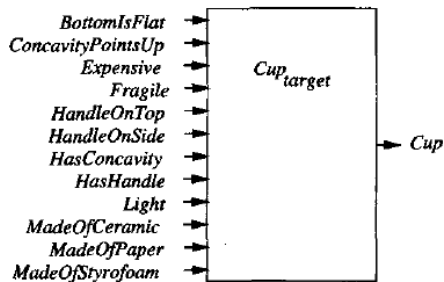
برای تصور این نوع تئوری قلمرو که EBNN از آن استفاده می‌کند شکل ۱۲,۷ را در نظر بگیرید. قسمت بالایی این شکل تئوری قلمروی‌ای برای EBNN را نشان می‌دهد که برای تابع هدف "فنجان" ایجاد شده است، هر مستطیل در این شکل یک شبکه‌ی عصبی مجزا را در تئوری قلمرو نشان می‌دهد. توجه دارید که در این مثال برای هر *horn clause* در قلمرو سمبولیک جدول ۱۲,۳ یک شبکه وجود دارد. برای مثال، شبکه‌ای به نام Graspable تمامی ویژگی‌های شبکه را به عنوان ورودی دریافت می‌کند و خروجی‌ای متناسب با ویژگی Graspable (در دست جا شدن) می‌دهد (EBNN گزاره‌ای درست را با مقدار ۰,۸ مقدار غلت گزاره‌ای را با ۰,۲ نشان می‌دهد). این شبکه مشابه Horn clause مربوط به ویژگی Graspable در جدول ۱۲,۳ عمل خواهد کرد. بعضی شبکه‌ها خروجی دیگر شبکه‌ها را به عنوان ورودی دریافت می‌کنند (برای مثال، شبکه‌ی سمت راست که Cup علامت‌گذاری شده است از خروجی شبکه‌های Lifiable, Stable و OpenVessel ورودی می‌گیرد). بنابراین، شبکه‌ای که تئوری قلمرو را می‌سازد می‌تواند مشابه Horn clause ها که گاهی ترکیبی زنجیروار از horn clause ها بودند ترکیبی زنجیروار از شبکه‌ها باشد. در کل این تئوری قلمرو ممکن است این اطلاعات توسط منبع خارجی به یادگیرنده داده شود یا ممکن است این شبکه نتیجه‌ی یادگیری قبلی همان سیستم باشد. EBNN از این شبکه‌های تئوری قلمرو برای یادگیری تابع هدف جدید استفاده می‌کند. EBNN شبکه‌های تئوری قلمرو را در طی این فرایند تغییر می‌دهد.

^{۱۲} Explanation-based Neural Network learning

Explanation of training example in terms of domain theory:



Target network:



شکل ۱۲،۷ توضیح نمونه‌ی آموزشی‌ای در EBNN.

توضیحات از پیش‌بینی تابع هدف توسط شبکه‌های تئوری قلمرو (شکل بالایی) تشکیل یافته است. مشتقات آموزشی از این توضیحات برای آموزش شبکه‌ی هدف مجزا (شکل پایینی) استفاده می‌شود. هر بلوک مستطیلی نشان‌دهنده‌ی شبکه‌ی عصبی چندلایه‌ای است.

هدف EBNN یادگیری شبکه‌ای جدید است که تابع هدف را توصیف کند. به این شبکه‌ی جدید شبکه‌ی هدف^{۱۳} می‌گوییم. در مثال شکل ۱۲،۷ شبکه‌ی هدف شبکه‌ی Cup_{target} است که در بالای شکل نشان داده شده و ویژگی‌های یک جسم را به عنوان ورودی دریافت کرده و مشخص می‌کند که جسم فنجان است یا خیر.

EBNN با استفاده از الگوریتم TangentProp که در قسمت قبل آمد شبکه‌ی هدف را یاد می‌گیرد. از قسمت‌های پیش می‌دانید که الگوریتم TangentProp شبکه‌ای را آموزشی می‌دهد که هم با مقادیر آموزشی و هم با مشتقات آموزشی سازگار باشد. الگوریتم EBNN مقادیر آموزشی‌ای $\langle x_i, f(x_i) \rangle$ را که از ورودی دریافت می‌کند و به الگوریتم TangentProp می‌دهد. علاوه بر این، EBNN مقادیر مشتقات آموزشی‌ای را که بر اساس تئوری قلمرو محاسبه می‌کند را به الگوریتم TangentProp می‌دهد. برای درک چگونگی محاسبه‌ی مشتقات آموزشی، دوباره به شکل ۱۲،۷ توجه کنید. قسمت بالایی این شکل پیش‌بینی تئوری قلمرو از مقادیر تابع هدف برای نمونه‌ی آموزشی x_i را نشان می‌دهد. EBNN مقدار مشتق این پیش‌بینی را با توجه به ویژگی نمونه‌ی ورودی محاسبه می‌کند. برای مثال در این شکل، نمونه‌ی x_i توسط ویژگی‌هایش مثل $MadeOfStyrofoam = 0.2$ (مثلاً غلت) و پیش‌بینی تئوری قلمرو با $Cup = 0.8$ (مثلاً درست) توصیف می‌شوند. EBNN مشتقات جزئی این پیش‌بینی را نسبت به ویژگی‌های نمونه را به صورت مجموعه‌ی مشتقات زیر محاسبه می‌کند.

$$\left[\frac{\partial Cup}{\partial BottomIsFlat}, \frac{\partial Cup}{\partial ConcavityPointsUp}, \dots, \frac{\partial Cup}{\partial MadeOfStyrofoam} \right]_{x=x_i}$$

این مجموعه مشتقات گرادیان تابع انتظاری تئوری قلمرو را نسبت به ورودی‌ها نشان می‌دهد. زیرنویس $x = x_i$ به این حقیقت اشاره می‌کند که این مشتقات در نقطه‌ی $x = x_i$ بررسی می‌شوند. در حالت کلی‌تر، زمانی که تابع هدف چندین خروجی دارد، گرادیان تمامی خروجی‌ها نسبت به تمامی ورودی‌ها محاسبه خواهد شد. ماتریس حاصل از این عملیات ژاکوبین (Jacobian) تابع هدف نامیده می‌شود.

برای درک اهمیت این مشتقات آموزشی در کمک به یادگیری شبکه‌ی هدف، مشتق $\frac{\partial Cup}{\partial Expensive}$ را در نظر بگیرید. اگر تئوری قلمرو بیان کند که ویژگی Expensive تأثیری بر تابع هدف Cup ندارد، مشتق $\frac{\partial Cup}{\partial Expensive}$ که از توضیحات استخراج می‌شود مقدار صفر خواهد داشت. صفر بودن مشتق بدین معناست که ویژگی Expensive تأثیری بر پیش‌بینی مقدار Cup ندارد. از طرف دیگر، مشتق بسیار بزرگ مثبت یا منفی بدین معناست که مقدار ویژگی تأثیر بسیار زیادی در تعیین مقدار هدف دارد. بنابراین، مشتقات استخراجی از توضیحات تئوری قلمرو اطلاعات مهمی در تعیین مرتبط یا نامرتب بودن ویژگی‌ها به مقدار هدف را ارائه می‌کند. زمانی که این مشتقات استخراجی به عنوان مشتقات آموزشی به TangentProp داده می‌شود تا شبکه‌ی هدف Cup_{target} را یاد بگیرد، این مشتقات بایاسی مفید برای جهت‌دهی تعمیم شبکه ارائه می‌کنند. در چنین شرایطی، بایاس نحوی معمول استقرایی شبکه‌های عصبی با این بایاس که از مشتقات آموزشی استخراجی از تئوری قلمرو تشکیل یافته جایگزین می‌شود.

در بالا نحوه‌ی استفاده از پیش‌بینی تئوری قلمرو در ایجاد مجموعه‌ای از مشتقات آموزشی آورده شده است. به عبارت دقیق‌تر الگوریتم کامل EBNN به صورت زیر است. با داشتن نمونه‌های آموزشی و تئوری قلمرو، EBNN ابتدا شبکه‌ای کامل (fully-connected) یک‌طرفه (feedforward) برای نمایش تابع هدف ایجاد می‌کند. این شبکه‌ی هدف، مشابه الگوریتم Backpropagation، با وزن‌های کوچک مقداردهی اولیه می‌شود. سپس برای هر نمونه‌ی آموزشی $\langle x_i, f(x_i) \rangle$ ، EBNN مشتق آموزشی مربوطه را در فرایندی دومرحله‌ای

^{۱۳} target network

مشخص می‌کند. ابتدا از تئوری قلمرو برای پیش‌بینی مقدار تابع هدف برای نمونه‌ی x_i استفاده می‌شود. بیاپید پیش‌بینی تابع هدف برای نمونه‌ی x_i را $A(x_i)$ در نظر بگیریم. به عبارت دیگر، $A(x_i)$ تابعی است که توسط ترکیب شبکه‌های تئوری قلمرو که برای x_i توضیحی ارائه می‌کنند تعریف می‌شود. دوم، وزن‌ها و توابع فعالیت شبکه‌های تئوری قلمرو برای استخراج $A(x_i)$ نسبت به تمامی ویژگی‌های x_i بررسی می‌شوند (ژاکوبین $A(x)$ در نقطه‌ی $x = x_i$). استخراج این مشتقات با فرایندی بسیار مشابه محاسبه‌ی عبارت δ در Backpropagation ادامه می‌یابد (تمرین ۱۲،۵). بالاخره EBNN از تفاوت جزئی الگوریتم TangentProp استفاده می‌کند و شبکه هدف را طوری آموزش می‌دهد که مقدار تابع خطای زیر مینیمم کند.

$$E = \sum_i \left[\left(f(x_i) - \hat{f}(x_i) \right)^2 + \mu_i \sum_j \left(\frac{\partial A(x)}{\partial x^j} - \frac{\partial \hat{f}(x)}{\partial x^j} \right)_{(x=x_i)}^2 \right] \quad (12.2)$$

که در این رابطه داریم

$$\mu_i \equiv \frac{|A(x_i) - f(x_i)|}{c} \quad (12.3)$$

در اینجا x_i همان x امین نمونه‌ی آموزشی است و $A(x)$ همان پیش‌بینی تئوری قلمرو برای ورودی x است. نماد x^j نیز برای نشان دادن j امین مؤلفه‌ی بردار x به کار رفته است (برای مثال، j امین گره ورودی شبکه‌ی عصبی). ثابت c نیز یک ثابت نرمالایز^{۱۴} است، این ثابت باعث می‌شود که مقدار μ_i همیشه $0 \leq \mu_i \leq 1$.

با وجود اینکه نمایش در اینجا غیر گویا به نظر می‌رسد، اما ایده‌ی این عبارت بسیار ساده است. خطای آورده شده در رابطه‌ی ۱۲،۲ همان فرم کلی تابع خطای رابطه‌ی ۱۲،۱ (رابطه‌ی مربوطه‌ی TangentProp) را دارد. جمله‌ی اول این تابع خطا، خطای مجموع مربعی بین مقادیر آموزشی $f(x_i)$ و مقدار پیش‌بینی شبکه‌ی هدف $\hat{f}(x_i)$ را تأثیر می‌دهد. جمله‌ی دوم خطای مربعی بین مقادیر مشتقات آموزشی استخراجی از تئوری قلمرو $\frac{\partial A(x)}{\partial x^j}$ و مقادیر واقعی مشتقات شبکه‌ی هدف $\frac{\partial \hat{f}(x)}{\partial x^j}$ را نشان می‌دهد. بنابراین، عبارت سمت چپ نقش قید بخش استقرایی را دارد بدین معنا که فرضیه‌ی خروجی باید با مقادیر آموزشی مشاهده شده مطابقت داشته باشد، در حالی که جمله‌ی سمت راست نقش تحلیلی را دارد بدین معنا که فرضیه‌ی خروجی باید با مشتقات آموزشی استخراجی از تئوری قلمرو مطابقت داشته باشد. توجه دارید که مشتق $\frac{\partial \hat{f}(x)}{\partial x^j}$ در رابطه‌ی ۱۲،۲ فقط حالت خاصی از رابطه‌ی $\frac{\partial \hat{f}(s_j(\alpha, x))}{\partial \alpha}$ است که در رابطه‌ی ۱۲،۱ آمده بود، در این حالت خاص $S_j(\alpha, x)$ تبدیلی است که x_i^j را به $x_i^j + \alpha$ تبدیل می‌کند، عبارت دقیق تغییر وزن در EBNN (Thrun 1996) آمده است.

اهمیت نسبی مؤلفه‌های استقرایی و تحلیلی یادگیری در EBNN با ثابت μ_i تعیین می‌شود که در رابطه‌ی ۱۲،۳ تعریف شده است. مقدار μ_i توسط تفاوت بین پیش‌بینی تئوری قلمرو $A(x_i)$ و مقدار آموزشی $f(x_i)$ تعیین می‌شود. بنابراین وزن مؤلفه‌ی تحلیلی یادگیری برای نمونه‌های آموزشی‌ای که پیش‌بینی درستی ندارند کمتر در نظر گرفته می‌شود. این وزن دهی ابتکاری فرض می‌کند که مشتقات آموزشی استخراجی از تئوری قلمرو در مواردی که تئوری قلمرو مقدار آموزشی را درست پیش‌بینی می‌کند درست‌ترند. با این وجود، می‌توان وضعیت‌هایی

^{۱۴} normalizing constant

را به وجود آورد که چنین ابتکاری موفق نباشد، در عمل این روش در چندین تئوری قلمرو موفق از Water در آمده است (Mitchell and Thrun 1993a; Thrun 1996).

۱۲,۶,۵ نکات

به طور خلاصه، الگوریتم EBNN از تئوری قلمرو برای ایجاد مجموعه‌ای از شبکه‌های عصبی آموزش دیده استفاده می‌کند و از آن به همراه نمونه‌های آموزشی برای آموزش فرضیه‌ی خروجی‌اش استفاده می‌کند. برای هر نمونه‌ی آموزشی EBNN از تئوری قلمرو برای توضیح نمونه استفاده می‌کند، سپس مشتق آموزشی را از این توضیح استخراج می‌کند. برای هر ویژگی نمونه، مشتق آموزشی‌ای که میزان تأثیر تغییر کوچک ویژگی در مقدار تابع هدف است را از تئوری قلمرو استخراج می‌کند. این مشتقات آموزشی به نسخه‌ای از TangentProp که به مشتقات آموزشی و مقادیر آموزشی شبکه‌ی هدف را تطبیق می‌دهد داده می‌شود. تطابق با مشتقات، شبکه‌ی یاد گرفته را ملزم به تطابق وابستگی‌های موجود در تئوری قلمرو می‌کند، در حالی که تطابق مقادیر آموزشی شبکه‌ی یاد گرفته را ملزم به تطابق با خود مقادیر آموزشی می‌کند. وزن μ ضرب مشتقات آموزشی، به طور جداگانه برای هر نمونه‌ی آموزشی، بر اساس دقت تئوری قلمرو برای پیش‌بینی مقادیر آموزشی این نمونه تعیین می‌شود.

نشان داده شده است که EBNN متدی کارا برای یادگیری از تئوری قلمروهای تخمینی در بسیاری از قلمروهاست. (Thrun 1996) این روش را در یادگیری نسخه‌های مختلف مسئله‌ی یادگیری Cup، که در بالا توضیح داده شد، به کار می‌برد و گزارش می‌دهد که EBNN دقت بهتری نسبت به Backpropagation، مخصوصاً زمانی که تعداد نمونه‌های آموزشی کم است، دارد. برای مثال، بعد از ۳۰ نمونه‌ی آموزشی، EBNN به خطای ریشه‌ی میانگین مربعی^{۱۵} ۵,۵ بر روی مجموعه‌ای مجزا از داده‌های تست رسید، در حالی که خطای Backpropagation ۱۲,۰ بود. (Mitchell and Thrun 1993a) استفاده از EBNN را در یادگیری کنترل ربات شبیه‌سازی شده، که تئوری قلمروش شبکه‌های عصبی‌ای که تأثیرات حرکات ربات را بر وضعیت نشان می‌دهند به کار می‌برند. به طور مشابه، EBNN با تخمینی کارایی بهتری نسبت به backpropagation کسب می‌کند. در اینجا backpropagation حداقل ۹۰ بار حلقه‌ی آموزش را برای رسیدن به سطحی از دقت تکرار می‌کند در حالی که EBNN به همان سطح از خطا را در ۲۵ حلقه می‌رسد. (O'Sullivan et al. 1997) و (Thrun 1996) چندین کاربرد دیگر EBNN را در درک و کنترل ربات‌های واقعی را با تئوری قلمرو‌ی‌ای از شبکه‌ها که تأثیر اعمال را برای ربات در محیط بسته با میکروفن^{۱۶}، دید و سنسورهای لیزری تعیین می‌شود، را توصیف می‌کنند.

EBNN رابطه‌ی جالبی با دیگر متدهای یادگیری توضیحی، مثل Prolog-EBG که در فصل ۱۱ توضیح داده شده، دارد. با توجه به آنچه در آنجا گفته شده، Prolog-EBG نیز توضیحاتی (پیش‌بینی مقادیر هدف نمونه) بر اساس تئوری قلمرو ایجاد می‌کند. در Prolog-EBG توضیحات با استفاده از تئوری قلمرو‌ی‌ای متشکل از horn clause ها و فرضیه‌ی هدف با محاسبه‌ی ضعیف‌ترین پیش‌فرضی که در آن توضیحات درست باشند بازبینی می‌شود. بنابراین وابستگی‌های نسبی در این توضیحات با فرضیه‌های horn clause های یاد گرفته شده بیان می‌شود. EBNN نیز توضیحات مشابهی ایجاد می‌کند، اما توضیحات EBNN بر پایه‌ی تئوری قلمرو‌ی‌ای از شبکه‌های عصبی، به جای horn clause ها، است. مشابه Prolog-EBG، وابستگی‌های نسبی از این توضیحات استخراج شده و برای بازبینی فرضیه‌ی هدف به کار می‌رود. در EBNN این وابستگی‌ها به فرم مشتقاتی بیان می‌شود زیرا که مشتقات نمایش طبیعی وابستگی در توابع پیوسته، مشابه شبکه‌های

^{۱۵} root-mean-squared

^{۱۶} sonar

عصبی است. در مقابل، نمایش طبیعی وابستگی‌ها در توضیحات نمادین یا اثبات‌های منطقی توصیف مجموعه‌ای از نمونه‌هایی است که این اثبات برایشان صادق است.

تفاوت‌های بسیاری بین قابلیت‌های EBNN و متدهای نمادین یادگیری فصل ۱۱ وجود دارد. تفاوت اصلی در این است که EBNN از تئوری قلمروهای ناکامل استفاده می‌کند، در حالی که Prolog-EBG از تئوری قلمروهای کامل استفاده می‌کند. این تفاوت از این حقیقت که EBNN بر پایه‌ی مکانیسم استقرایی تطابق با مقادیر آموزشی مشاهده شده و استفاده از تئوری قلمرو فقط به عنوان قید اضافی‌ای بر روی فرضیه‌ی یاد گرفته شده ساخته شده ناشی شده است. تفاوت مهم دوم از این حقیقت ناشی می‌شود که Prolog-EBG دسته‌ای افزایشی از horn clause ها را یاد می‌گیرد در حالی که EBNN از شبکه‌ی عصبی‌ای با اندازه‌ی ثابت استفاده می‌کند. همان‌طور که در فصل ۱۱ نیز گفته شد، یکی از مشکلات یادگیری دسته قوانین horn clause این است که هزینه‌ی دسته‌بندی نمونه‌های جدید با ادامه‌ی فرایند یادگیری و افزایش horn clause های جدید افزایش می‌یابد. این مشکل در EBNN وجود ندارد زیرا که شبکه‌ی هدف با اندازه‌ی ثابت زمان ثابتی برای دسته‌بندی نمونه نیاز خواهد داشت. با این وجود، شبکه‌ی عصبی با اندازه‌ی ثابت در مقابل مشکلاتی دارد، زیرا که ممکن است نتواند توابع به اندازه‌ی کافی پیچیده را نشان دهد، در حالی که دسته‌ای از horn clause ها می‌توانند با افزایش تعداد هر تابع پیچیده‌ای را نمایش دهند. (Mitchell and Thrun 1993b) بحث دقیق‌تری از رابطه‌ی EBNN و متدهای یادگیری توضیحی نمادین انجام می‌دهد.

۱۲,۵ استفاده از دانش قبلی برای تغییر عملگرهای جستجو

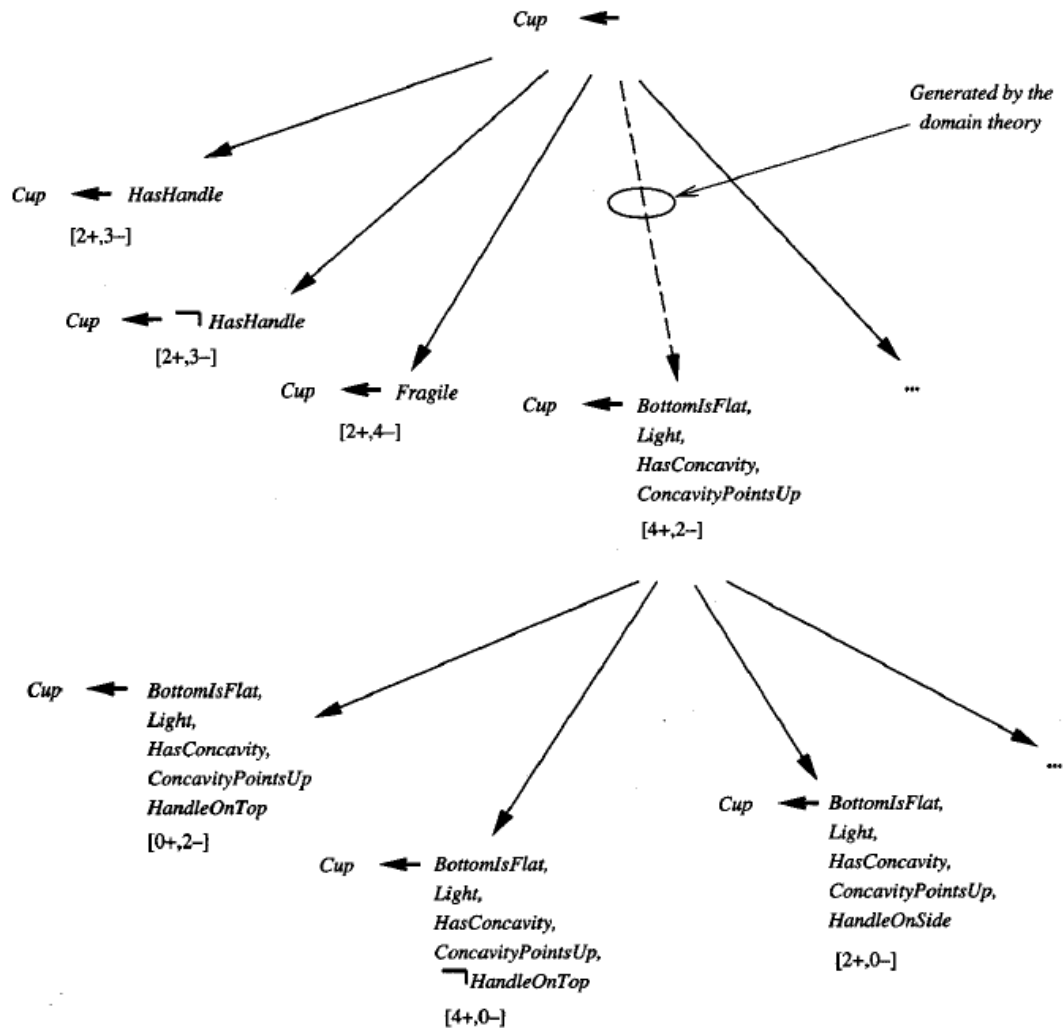
در دو قسمت گذشته دو نقش دانش قبلی در یادگیری را مطرح کردیم: مقداردهی اولیه‌ی فرضیه‌ی یادگیر و تغییر تابع هدفی که جستجوی فضای فرضیه‌ای را کنترل می‌کند. در این بخش، روش سومی برای استفاده از دانش قبلی برای تغییر جستجوی فضای فرضیه‌ای را مورد بحث قرار خواهیم داد: استفاده از دانش قبلی برای تغییر مجموعه‌ی عملگرهایی که مراحل قانونی در جستجوی فضای فرضیه‌ای را تعیین می‌کنند. این روش در الگوریتم‌های FOCL، (Pazzani 1991; Pazzani and Kibler 1992) و (Bergando and ML-SMART) (Giordana 1990) استفاده شده است. ما در اینجا از الگوریتم FOCL برای تصور روش استفاده خواهیم کرد.

۱۲,۵,۱ الگوریتم FOCL

FOCL تعمیمی از سیستم استقرایی محض FOIL است که در فصل ۱۰ به طور کامل به آن پرداختیم. هر دو سیستم FOIL و FOCL مجموعه‌ای از horn clause های درجه اول برای پوشاندن نمونه‌های مشاهده شده ایجاد می‌کند. هر دو سیستم از الگوریتمی ترتیبی که ابتدا یک قانون یاد می‌گیرد و تمامی نمونه‌های مثبت پوشانده شده با آن را حذف می‌کند و سپس فرایند را برای نمونه‌های باقی‌مانده تکرار می‌کند استفاده می‌کنند. در هر دو سیستم، هر horn clause جدید با جستجوی کلی به جزئی ساخته می‌شود، این جستجو از کلی‌ترین horn clause ممکن آغاز می‌شود (مثلاً horn clause ی که هیچ شرطی ندارد). سپس چندین جزئی سازی ممکن این horn clause ساخته می‌شود و جزئی سازی‌ای که بیشترین بهره‌ی اطلاعات را بر روی نمونه‌های آموزشی دارد انتخاب می‌شود. این فرایند تکرار می‌شود تا جزئی سازی‌های بیشتری ایجاد شود و باز هم بهترین آن‌ها انتخاب شده تا اینکه به horn clause ی برسد که کارایی رضایت‌بخش را داشته باشد.

تفاوت FOIL و FOCL در نحوه‌ی ایجاد خاص سازی‌ها در طی جستجوی کلی به جزئی برای یافتن یک horn clause است. همان‌طور که در فصل ۱۰ نیز گفته شد، FOIL هر جزئی سازی را با اضافه کردن یک عبارت جدید به شروط horn clause می‌سازد. FOCL نیز از

همین روش برای ایجاد خاص سازی‌های ممکن استفاده می‌کند اما علاوه بر آن‌ها خاص سازی‌های مبتنی بر تئوری قلمرو را نیز در نظر می‌گیرد. یال‌های پررنگ درخت جستجو در شکل ۱۲,۸ مراحل جستجوی کلی به جزئی در نظر گرفته شده در جستجوی FOIL را نشان می‌دهند. یال‌های خط‌چین در این درخت جستجو (شکل ۱۲,۸) خاص سازی‌های اضافی در نظر گرفته شده توسط FOCL و مبتنی بر تئوری قلمرو هستند.



شکل ۱۲,۸ جستجوی فضای فرضیه‌ای FOCL.

برای یادگیری یک قانون، FOCL فرضیه‌های ممکن را کلی به جزئی و به صورت افزایشی مورد بررسی قرار می‌دهد. دو نوع عملگر خاص سازی فرضیه‌ی فعلی را انجام می‌دهند. یکی از این دو نوع تنها یک عبارت جدید به قانون اضافه می‌کند (خطوط متوسط شکل). عملگر نوع دوم خاص سازی قانون فعلی را با اضافه کردن مجموعه‌ای از عبارات که به طور منطقی شرط کافی تابع هدف بر اساس تئوری قلمرو است را در نظر می‌گیرد (خطوط خط‌چین شکل). FOCL از میان این خاص سازی‌ها، با معیار کارایی‌شان بر روی داده‌ها خاص سازی‌ای انتخاب می‌کند. بنابراین تئوری قلمروهای ناکامل فقط زمانی که مدارک تأییدی داشته باشند تأثیر خواهند گذاشت. این مثال بر پایه داده‌ها و تئوری قلمروی استفاده شده در KBANN آورده شده است.

با وجود اینکه FOIL و FOCL هر دو horn clause های درجه اول را یاد می‌گیرند اما در اینجا بحث را به قوانین گزاره‌ای درجه اول یا همان horn clause های بدون متغیر محدود می‌کنیم. دوباره مفهوم هدف Cup، نمونه‌های آموزشی مربوطه و تئوری قلمرو شکل ۱۲,۳ را در نظر بگیرید. برای توصیف عملیات FOCL، ابتدا باید فرق بین عبارات آمده در تئوری قلمرو و فرضیه‌ها را مشخص کنیم. زمانی می‌گوییم

یک عبارت عملیاتی (operational) است که اجازه داشته باشیم آن را در توصیف یک فرضیه‌ی خروجی بکار ببریم. برای مثال در مثال Cup شکل ۱۲,۳ ما فقط اجازه داریم که از ۱۲ ویژگی‌ای که در نمونه‌های آموزشی آمده (مثل HasHandle, HandleOnTop) عبارات مبتنی بر این ۱۲ ویژگی بنابراین عبارات عملیاتی شمرده می‌شوند. در مقابل، عباراتی که مبتنی بر ویژگی واسطه‌ی میانی در تئوری قلمروند و مبتنی بر ویژگی اولیه‌ی نمونه‌ها نیستند عبارات غیرعملیاتی به شمار می‌آیند. نمونه‌ای از ویژگی‌های غیرعملیاتی در این مثال ویژگی Stable است.

در هر مرحله از جستجوی کلی به جزئی FOCL فرضیه‌ی فعلی خود h را با دو عملگر زیر گسترش می‌دهد:

برای هر عبارت عملیاتی که جزو h نیست، یک جزئی سازی از قانون h با اضافه کردن این تک عبارت به شروط قانون بساز. FOIL نیز از این متد برای ایجاد جزئی سازی ممکن استفاده می‌کند. فلش‌های پررنگ شکل ۱۲,۸ این نوع جزئی سازی را نشان می‌دهد. مطابق با تئوری قلمرو شرطی منطقی و عملیاتی برای تابع هدف بساز. این عبارت جدید را به شروط قانون h اضافه کن و بالاخره با حذف شروط اضافی (برای پوشش نمونه‌ها) h را هرس کن. فلش خط‌چین در شکل ۱۲,۸ چنین جزئی سازی‌ای را نشان می‌دهد. جزئیات عملگر دوم به شرح زیر است. FOCL ابتدا یکی از قوانین تئوری قلمرو که حکمش با تابع هدف یکی است را انتخاب می‌کند. اگر چندین قانون چنین حالتی داشته باشد قانونی انتخاب خواهد شد که بهره‌ی اطلاعات بیشتری بر روی تابع هدف دارد. برای مثال، با تئوری قلمرو و داده‌های جدول ۱۲,۳ فقط یک قانون چنین حالتی خواهد داشت:

Cup ← Stable, Lifiable, OpenVessel

شروط قانون انتخابی شرطی منطقی کافی برای تابع هدف را تشکیل می‌دهد. هر عبارت غیرعملیاتی در این شرط کافی با استفاده از تئوری قلمرو جایگزین می‌شود، بدین معنا که حکم قوانین با شروطشان جایگزین می‌شود. برای مثال، قانون Stable ← BottomIsFlat معادل جایگزین کردن عبارت عملیاتی BottomIsFlat به جای عبارت غیرعملیاتی Stable استفاده می‌شود. این فرایند باز کردن^{۱۷} تئوری قلمرو تا جایی که شرط کافی با عبارات عملیاتی بیان شود ادامه پیدا می‌کند. اگر چندین قانون از تئوری قلمرو برای یک عبارت توضیح آورده باشند آنکه بهره‌ی اطلاعات بیشتری دارد برای جایگزینی استفاده می‌شود. واضح است که شرط کافی با داده‌ها و تئوری قلمرو موجود در مثال Cup به صورت زیر است:

BottomIsFlat, HasHandle, Light, HasConcavity, ConcavityPointsUp

در مرحله‌ی آخر در ایجاد خاص سازی ممکن، این شرط کافی هرس می‌شود. بدین صورت که هر عبارتی که حذفش باعث کاهش دقت دسته‌بندی بر روی نمونه‌های آموزشی نمی‌شود حذف خواهد شد. این مرحله برای تشخیص جزئی سازی بیش از حد^{۱۸} ایجاد شده است زیرا که تئوری قلمروهای ناکامل ممکن است عبارات نامربوط در بر داشته باشند. در مثال حاضر، حذف عبارت HasHandle باعث افزایش کارایی می‌شود. بنابراین شرط کافی به صورت زیر خواهد بود:

BottomIsFlat, Light, HasConcavity, ConcavityPointsUp

^{۱۷} unfolding

^{۱۸} overspecialization

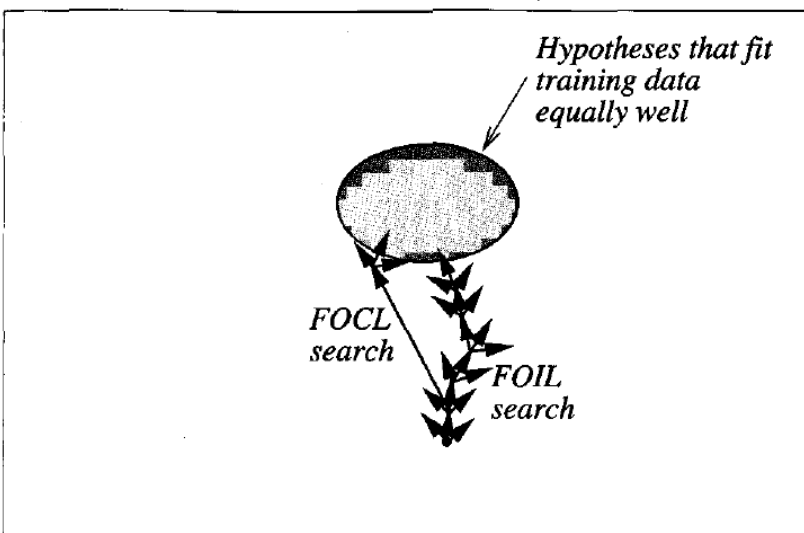
حال این مجموعه از عبارات به شروط فرضیه فعلی اضافه می‌شود. توجه داشته باشید که فرضیه حاصل با فلش خطچین در شکل ۱۲,۸ نشان داده شده است.

یکی از جزئی سازی‌های فرضیه‌ی فعلی با استفاده از دو عملگر بالا ایجاد شده است. بین این فرضیه‌ها فرضیه‌ای که بیشترین بهره‌ی اطلاعات را داشته باشد انتخاب خواهد شد. در مثال نشان داده شده در شکل ۱۲,۸ جزئی سازی انتخاب شده در مرحله‌ی اول درخت جستجو همان خاص سازی حاصل از تئوری قلمرو است. در ادامه‌ی جستجو سیستم به دنبال جزئی سازی بیشتر شروط تئوری قلمرو می‌گردد. این قسمت از جستجو قسمت استقرایی جستجو است که بازنگری در شروط اشتقاقی از تئوری قلمرو را ممکن می‌سازد. در مثال حاضر، تئوری قلمرو فقط در مرحله‌ی اول جستجو تأثیر می‌گذارد. با این وجود همیشه این اتفاق نمی‌افتد. اگر تئوری قلمرو به اندازه‌ی کافی کارا نباشد در مرحله‌ی اول از روش استقرایی استفاده می‌شود و تأثیر تئوری قلمرو به مراحل بعدی موقوف می‌شود. به طور خلاصه اینکه FOCL، horn clause ها را به فرم زیر یاد می‌گیرد،

$$c \leftarrow o_i \wedge o_b \wedge o_f$$

در این رابطه C مفهوم هدف است و o_i عطفی اولیه از عبارات عملیاتی است که در مرحله به مرحله توسط عملگر اول به قانون اضافه شده است (در مراحل قبلی)، o_b عطفی از عبارات عملیاتی است که در یک مرحله بر اساس تئوری قلمرو اضافه می‌شود و o_f نیز عطفی از عبارات عملیاتی است که در یک مرحله توسط عملگر اول اضافه می‌شود. هر یک از این سه مجموعه ممکن است تهی باشد.

Hypothesis Space



شکل ۱۲,۹ جستجوی فضای فرضیه‌ای FOCL.

FOCL مجموعه‌ای عملگرهای جستجوی FOIL را تغییر می‌دهد. با وجود اینکه FOIL فقط اضافه کردن یک عبارت را در هر مرحله در نظر می‌گیرد، اما FOIL اضافه کردن چندین عبارت استخراجی از تئوری قلمرو در یک مرحله را نیز در نظر می‌گیرد.

بحث بالا استفاده از تئوری قلمرو گزاره‌ای در ایجاد خاص سازی‌های ممکن فرضیه در طی جستجوی کلی به جزئی برای پیدا کردن horn clause مناسب را توصیف می‌کند. این الگوریتم می‌تواند به راحتی برای نمایش درجه اول تعمیم داده شود (قوانینی که متغیر نیز دارند). فصل ۱۰ به طور کامل الگوریتم FOIL و نحوه‌ی تعمیم ایجاد horn clause های درجه اول بدون متغیر به ایجاد horn clause های درجه

اول با متغیر را توضیح داده است. برای تعمیم عملگر دوم برای تطبیق با تئوری قلمروهایی با horn clause های درجه اول، در باز کردن تئوری قلمرو باید متغیرها را در نظر گرفت. این کار را می‌توان با روش فرایند regression در جدول ۱۱،۳ انجام داد.

۱۲،۵،۲ نکات

FOCL از تئوری قلمرو برای افزایش جزئی سازی‌های ممکن در نظر گرفته شده در هر مرحله از جستجوی horn clause ها استفاده می‌کند. شکل ۱۲،۹ جستجوی فضای فرضیه‌ای الگوریتم FOCL و الگوریتم استقرایی محض FOIL را مقایسه می‌کند. جزئی سازی متناسب با تئوری قلمرو در FOCL مشابه تغییرات بزرگ (macro) در جستجوی FOIL که در آن چندین عبارت همزمان و در یک گام به قانون اضافه می‌شود. این فرایند را می‌توان به دید ترفیع فرضیه‌ای که ممکن است در ادامه‌ی جستجو مورد بررسی قرار گیرد در نظر گرفت ---. اگر تئوری قلمرو درست باشد، داده‌های آموزشی نیز مطابق با آن خواهد بود، پس این خاص سازی انتخاب خواهد شد. حال اگر تئوری قلمرو درست نباشد، ارزیابی تمامی فرضیه‌ها صورت گرفته و مسیر دیگری برای ادامه‌ی جستجو انتخاب خواهد شد.

به طور خلاصه FOCL از هر دو روش ایجاد جزئی سازی ممکن با روش استقرایی و مبتنی بر تئوری قلمرو در هر مرحله از جستجو استفاده می‌کند. الگوریتم بین جزئی سازی‌های ممکن فقط بر اساس معیاری که بر اساس تجربه (نمونه‌های آموزشی) است انتخاب می‌کند. بنابراین تئوری قلمرو یادگیر را بایاس می‌کند اما انتخاب بین این بایاس و سیستم استقرایی را بر عهده‌ی تجربه (روش استقرایی) می‌گذارد. بایاسی تئوری قلمرو ایجاد می‌کند ترجیح انتخاب horn clause های مشابه horn clause های عملیاتی، منطقی و کافی استخراجی از تئوری قلمرو است. این بایاس با بایاس برنامه‌ی استقرایی محض FOIL، که ترجیح فرضیه‌های کوتاه‌تر است، ترکیب شده و بایاس این سیستم را ایجاد می‌کند.

FOCL نشان داده است که در تعدادی از کاربردها که تئوری قلمرو ناکامل در دسترس است دقت بهتری در تعمیم نسبت به الگوریتم استقرایی محض FOIL دارد. برای مثال، (Pazzani and Kibler 1992) با این روش مفهوم "چینش‌های مجاز صفحه‌ی شطرنج" را بررسی کردند. با مجموعه‌ای ۶۰ تایی از نمونه‌های آموزشی که حاوی ۳۰ نمونه‌ی مجاز و ۳۰ نمونه‌ی غیرمجاز بود، FOIL ۸۶ درصد روی مجموعه‌ی مجزایی کارایی داشت. در حالی که FOCL با همان نمونه‌های آموزشی و تئوری قلمرویی‌ای با ۷۶ کارایی، ۹۴ درصد کارایی نهایی داشت، که این خطای مقدار از نصف میزان خطای FOIL کمتر است. برای مثال، با مجموعه‌ای از ۵۰۰ نمونه‌ی آموزشی خرابی تلفن از شرکت NYNEX، FOIL کارایی ۹۰ درصد داشت در حالی که FOCL با تئوری قلمرویی‌ای با کارایی ۹۵ درصد به کارایی نهایی‌ای ۹۸ درصد روی همان مجموعه‌ی تست رسید.

۱۲،۶ آخرین پیشرفت‌ها

متدهای معرفی شده در این فصل فقط روش‌های ساده‌ی ممکن در ترکیب یادگیری تحلیلی و استقرایی است. در حالی که هر یک از این متدها نشان داده‌اند که کارایی بهتری نسبت به روش‌های استقرایی محض دارند، اما هیچ یک از این روش‌ها در روی مجموعه‌ی متنوعی از قلمروها^{۱۹} تست نشده‌یا کارایی خوبی نداشته‌اند. بحث ترکیب روش‌های استقرایی و تحلیلی هنوز یکی از بحث‌های قابل تحقیق و بررسی است.

^{۱۹} domain

۱۲,۷ خلاصه و منابع برای مطالعه بیشتر

نکات اصلی این فصل شامل موارد زیر است:

دانش قبلی تقریبی، یا تئوری قلمروها در بسیاری از مسائل یادگیری عملی در دسترس اند. روش‌های استقرایی محض مثل درخت تصمیم و شبکه‌های عصبی از چنین تئوری قلمروهایی استفاده نمی‌کنند و بنابراین فقط زمانی که مقدار داده‌های به اندازه کافی زیاد باشد درست عمل می‌کنند. روش‌های تحلیلی محض مثل Prolog-EBG از تئوری قلمروها استفاده کرده اما فرضیه‌های غلتی با داشتن دانش قبلی ناکامل ایجاد می‌کنند. متدهایی که مخلوطی از یادگیری استقرایی و تحلیلی‌اند می‌توانند از مزیت هر دو متد بهره ببرند: کاهش پیچیدگی نمونه‌ای و قدرت تصحیح دانش قبلی نادرست.

یکی از روش‌ها، الگوریتم‌های ترکیب یادگیری استقرایی و تحلیلی با توجه به تأثیر تئوری قلمرو بر جستجوی فضای فرضیه‌ای است. در این فصل به متدهایی پرداختیم که از تئوری قلمرو ناکامل برای (۱) ایجاد فرضیه اولیه‌ی جستجو استفاده می‌کند، (۲) گسترش مجموعه‌ی عملگرهای جستجو که بازنگری روی فرضیه فعلی می‌کنند استفاده می‌کنند و (۳) هدف جستجو را تغییر می‌دهند انجام دادیم.

یکی از سیستم‌هایی که از تئوری قلمرو برای مقداردهی اولیه‌ی فرضیه استفاده می‌کند الگوریتم KBANN است. این الگوریتم از تئوری قلمرو بیان شده با دسته قوانین گزاره‌ای تحلیلی برای ایجاد شبکه‌ی عصبی اولیه که معادل تئوری قلمرو باشد استفاده می‌کند. سپس این تئوری قلمرو به صورت استقرایی توسط الگوریتم Backpropagation، برای بهبود کارایی روی داده‌های آموزشی بازبینی می‌شود. نتیجه‌ی حاصل شبکه‌ای بایاس شده توسط تئوری قلمرو ابتدایی خواهد بود که وزن‌هایش با روش استقرایی روی داده‌های آموزشی بازبینی شده‌اند.

TangentProp از دانش قبلی نمایش داده شده با مشتقات تابع هدف نیز استفاده می‌کند. در بعضی زمینه‌ها، مثل پردازش تصویر، این روش، روشی عادی برای نشان دادن دانش قبلی است. TangentProp این دانش قبلی را با تغییر هدف جستجوی شیب نزول در فضای فرضیه تأثیر می‌دهد.

EBNN از تئوری قلمرو برای تغییر هدف جستجو در فضای وزن‌های ممکن یک شبکه‌ی عصبی استفاده می‌کند. این روش تئوری قلمرو به شکل شبکه‌های عصبی یاد گرفته شده‌ی پیشین معادل برای ایجاد شبکه‌ی عصبی --- ---

FOCL از تئوری قلمرو برای افزایش اعضا مجموعه‌ی فرضیه‌های ممکن در نظر گرفته شده در هر مرحله از جستجو استفاده می‌کند. این روش از تئوری قلمرو تقریبی نمایش داده شده با horn clause ها برای یادگیری مجموعه‌ای از horn clause ها که تابع هدف را تخمین می‌زنند استفاده می‌کند. FOCL از الگوریتمی پوشش ترتیبی استفاده کرده و برای یادگیری هر horn clause از جستجوی کلی به جزئی استفاده می‌کند. تئوری قلمرو برای تغییر مجموعه‌ی فرضیه‌های خاص تر ممکن در نظر گرفته شده در هر مرحله از این جستجو به کار می‌رود. فرضیه‌های ممکن سپس بر اساس کارایی‌شان روی داده‌های آموزشی سنجیده می‌شوند. با این روش، FOCL جستجوی حریمانه، کلی به جزئی و استقرایی FOIL را با زنجیر قانون (rule-chaining) و استدلال تحلیلی را ترکیب می‌کند.

اینکه چگونه به بهترین وجه ممکن دانش را با مشاهدات جدید ترکیب کنیم همچنان یکی از سؤالات اصلی یادگیری ماشین باقی مانده است.

الگوریتم‌های بسیاری وجود دارند که هدفشان ترکیب یادگیری استقرایی و یادگیری تحلیلی است. برای مثال، متدهای یادگیری شبکه‌های باور بیزی که در فصل ۶ به آن پرداختیم روش دیگری برای بحث در اینجا خواهند بود. منابع این فصل نمونه‌ها و منابع یادگیری بیشتر را دربر دارد.

تمرینات

۱۲,۱ مسئله‌ی یادگیری مفهوم GoodCreditRisk را بر روی نمونه‌های توصیفی با چهار ویژگی HasStudentLoan, HasSavingsAccount, OwnsCar و IsStudent را در نظر بگیرید. شبکه‌ی اولیه‌ی الگوریتم KBANN را شبکه‌ی حاصل از تئوری قلمروی زیر با تمامی ارتباطها در نظر بگیرید.

GoodCredit ← Employed, LowDept

Employed ← IsStudent

LowDebt ← HasStudentLoan, HasSavingsAccount

۱۲,۲ KBANN دسته‌ای از horn clause های گزاره‌ای را دریافت کرده و آن را به شبکه‌ی عصبی اولیه تبدیل می‌کند. کلاس قوانین گزاره‌ای "n از m" را در نظر بگیرید، این قوانین m عبارت شرط دارند و پارامتر مربوطه‌ی n را دربر می‌گیرند ($n \leq m$). زمانی شرط قانون گزاره‌ی "n از m" که حداقل n گزاره از m گزاره‌ی شرط آن درست باشند. برای مثال،

Student ← LivesInDorm, Young, Studies; n=2

نشان می‌دهد که زمانی مفهوم Student درست است که حداقل دو گزاره از سه گزاره‌ی شرط بالا درست باشد.

الگوریتمی مشابه الگوریتم KBANN طراحی کنید که دسته‌ای از قوانین "n از m" را دریافت کرده و شبکه‌ای عصبی سازگار با تئوری قلمرو ایجاد کند.

۱۲,۳ تعمیم الگوریتم KBANN را برای دریافت تئوری قلمروی‌ای که شامل قوانین درجه اول به جای horn clause هاست را در نظر بگیرید (horn clause هایی که متغیر نیز دارند، مشابه فصل ۱۰). اگر چنین کاری ممکن است الگوریتمی برای ساخت شبکه‌های عصبی معادل قوانین درجه اول ارائه کنید و اگر ممکن نیست مشکلات بازدارنده را بیان کنید.

۱۲,۴ این تمرین از شما می‌خواهد که شیب نزول را مشابه آنچه در TangentProp استفاده شد استخراج کنید. فضای نمونه‌ای X را شامل اعداد حقیقی در نظر بگیرید، فضای فرضیه‌ای H را نیز توابع درجه دو روی X در نظر بگیرید. یعنی،

$$h(x) = w_0 + w_1x + w_2x^2$$

(a) قانون شیب نزول را برای مینیمم کردن معیار Backpropagation به کار ببرید؛ معیار Backpropagation مینیمم کردن مجموع خطای مربعی بین فرضیه و داده‌های آموزشی است.

(b) قانون شیب نزول دیگری که همان معیار را برای TangentProp مینیمم می‌کند را استخراج کنید. فقط تبدیل $s(\alpha, x) = x + \alpha$ را در نظر بگیرید.

۱۲,۵ EBNN مشتقات آموزشی را از توضیحات حاصل از بررسی وزن‌ها و تحریک شبکه‌های عصبی ساخته شده برای توضیح به دست می‌آورد. نمونه‌ی ساده‌ای را که در آن توضیحات با یک سیگموئید با n ورودی ایجاد می‌شود را در نظر بگیرید. رابطه‌ای برای پیدا کردن مشتق آموزشی $\frac{\partial f(x)}{\partial x_j} |_{x=x_i}$ پیدا کنید، در این رابطه x_i یک نمونه‌ی آموزشی خاص ورودی به واحد، $f(x)$ خروجی واحد سیگموئید و x نشان‌دهنده‌ی j امین ورودی واحد سیگموئید است. می‌توانید از نمادگذاری x_j برای نشان دادن j امین ویژگی نمونه‌ی x استفاده کنید. راهنمایی: حل این مسئله مشابه استخراج قانون یادگیری برای backpropagation است.

۱۲,۶ دوباره مسیر یادگیری الگوریتم FOCL که در شکل ۱۲,۸ نشان داده شد را در نظر بگیرید. فرض کنید که فرضیه‌ی اولیه‌ی انتخاب شده فرضیه‌ی زیر باشد:

Cup ← -HasHandle

فرضیه‌های مرحله‌ی دوم ای که FOCL ایجاد می‌کند را پیدا کنید. فقط کافی است فرضیه‌های ایجاد شده توسط عملگر دوم جستجوی FOCL را پیدا کنید که از تئوری قلمرو استفاده می‌کند. فراموش نکنید که شروط کافی باید هرس شوند. از داده‌های جدول ۱۲,۳ برای آموزش استفاده کنید.

۱۲,۷ این فصل سه روش برای استفاده از دانش قبلی برای تأثیر بر جستجو فضای فرضیه‌های ممکن ارائه می‌کند. ایده‌های خود را برای اینکه چگونه می‌توان این سه روش را کامل کرد ارائه کنید. آیا می‌توانید الگوریتم خاصی ارائه کنید که دو روش از این سه روش را برای نمایش فرضیه‌ای خاصی به کار ببرد؟ مزیت‌ها و مضرت‌های این ترکیب چه خواهد بود؟

۱۲,۸ دوباره سؤال مطرح شده در قسمت ۱۲,۲,۱ را در نظر بگیرید. چه معیاری برای انتخاب میان فرضیه‌ها زمانی که هم داده و هم دانش قبلی در دسترس است استفاده شود؟ در این باره دیدگاه خود را بیان کنید.

فرهنگ لغات تخصصی فصل (فارسی به انگلیسی)