What is the curse of dimensionality and how does it affect clustering?

curse of dimensionality به چالش‌ها و محدودیت‌هایی اطلاق می‌شود که در هنگام کار با داده‌های high-dimensional به وجود می‌آیند. با افزایش تعداد ویژگی‌ها یا ابعاد در مجموعه داده، چندین مشکل به وجود می‌آید. یکی از مشکلات اصلی آن این است که مقدار داده مورد نیاز برای حفظ چگالی مشخصی از داده‌ها و پوشش در ان حجم از دیتا، با افزایش تعداد ابعاد به صورت نمایی افزایش می‌یابد. بنابراین، داده‌های high-dimensionalتمایل دارند که پراکنده شوند و این امر باعث می‌شود پیدا کردن الگوها یا روابط معنادار دشوارتر شود. علاوه بر این، curse of dimensionality می‌تواند منجر به افزایش پیچیدگی محاسباتی و overfitting شود. در فضاهای high-dimensional، فواصل بین نقاط داده کمتر اطلاعاتی را منتقل می‌کنند که این باعث می‌شود سنجش تشابه یا محاسبه آمارهای معنادار به صورت دقیق مشکل‌تر شود. این موضوع می‌تواند تأثیر منفی بر الگوریتم‌های یادگیری ماشین داشته باشد که بر معیارهای فاصله یا فرضیات آماری تکیه می‌کنند. برای کاهش curse of dimensionality، تکنیک‌های کاهش ابعاد مانند principal component analysis (PCA) یا t-distributed stochastic neighbor embedding(t-SNE) می‌توانند استفاده شوند تا تعداد ابعاد را کاهش داده و در عین حفظ الگوهای مهم، در داده موثر باشند.

curse of dimensionality از روش های مختلفی میتواند تاثیر زیادی روی clustering به دلیل افزایش پیچیدگی محاسباتی و پراکندگی داده

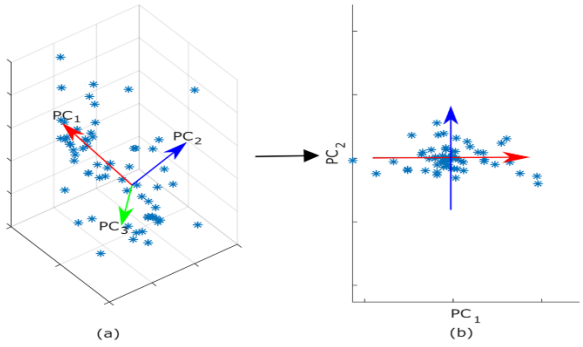
های high-dimensional داشته باشد. در فضاهای high-dimensional، مفهوم چگالی و جداسازی به شدت از دست می‌رود. نقاط داده تمایل دارند بیشتر پخش شوند که تعریف کردن clusterهای معنادار بر اساس فاصله یا چگالی را دشوار می‌کند. افزایش پراکندگی می‌تواند منجر به کم‌واضح‌تر شدن clusterها و overlapping بیشتر آنها شود که سبب می‌شود نقاط داده‌ها را سخت‌تر clusterبندی کرد.
همچنین الگوریتم‌های clustering مبتنی بر فاصله برای سنجش فاصله بین نقاط داده و تشکیل clusterها از معیار های فاصله استفاده می‌کنند. با این حال، در فضاهای high-dimensional ، فواصل بین نقاط به دلیل پدیده "distance concentration " کمتر اطلاعاتی انتقال می‌دهند. به عبارت دیگر، فواصل بین بیشتر نقاط داده به شدت مشابه یا تقریباً برابر می‌شوند که معیارهای فاصله سنتی را کمتر مؤثر می‌کند. هزینه محاسباتی الگوریتم‌های clustring نیز با افزایش تعداد ابعاد به صورت نمایی افزایش می‌یابد. با افزایش تعداد ابعاد، تعداد ترکیب‌ها و محاسبات مورد نیاز برای تعیین clusterها نیز افزایش می‌یابد. این می‌تواند منجر به پردازش‌های clustering کندتر و less efficient شود، به خصوص برای الگوریتم‌هایی که برای داده‌های high-dimensional بهینه‌سازی نشده‌اند.

In what cases would you use regular PCA, incremental PCA, randomized PCA, or random projection?

Regular PCA: معمولاً در مواردی استفاده می‌شود که دارای یک مجموعه داده با تعداد متوسط نمونه‌ها و ابعاد هستیم. این روش principal components را محاسبه می‌کند که متغیرهای مهم‌ترین در داده را به خوبی بیان می‌کنند و داده را بر روی یک فضای کم‌ابعاد پروژه می‌دهد در حالی که الگوهای مهم را حفظ می‌کند. PCA نرمال زمانی مناسب است که می‌توان کل مجموعه داده را به حافظه load کرد و محاسبات را بر روی آن انجام داد. پس به طور کلی از این نوع PCA برای تحلیل داده‌ها، استخراج ویژگی‌ها، تصویرسازی داده و کاهش بعد استفاده می‌شود. این روش مناسب است زمانی که کل مجموعه داده به طور کامل در حافظه قرار داده شده و به طور کارآمد به عنوان یک مجموعه کلی پردازش می‌شود.

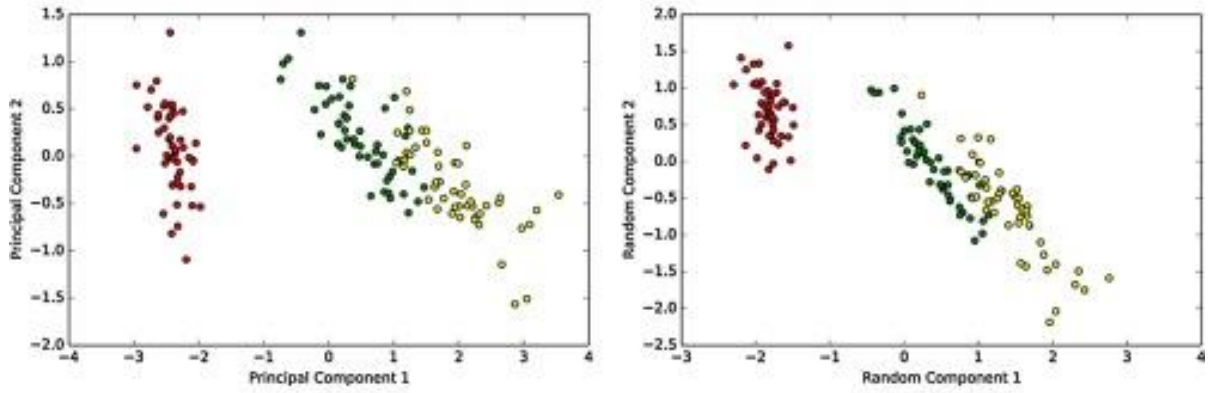
Incremental PCA (IPCA): در مواردی مفید است که دارای مجموعه داده بزرگی هستیم نمی‌توان ان را به یکباره در حافظه جای داد. IPCA داده را به صورت مینی-بچ یا بخش‌های کوچک پردازش می‌کند و principal component را به ترتیب برایشان به روزرسانی می‌کند. این رویکرد کارآمدی برای حالت‌های آنلاین یا استریمینگ است که داده به صورت تدریجی وارد می‌شود. IPCA به ما اجازه می‌دهد که در هر بار، ان را روی

یک زیرمجموعه از داده انجام دهیم و نیاز به حافظه را کاهش دهیم. IPCA امکان یادگیری آنلاین را تسهیل می‌کند و امکان یکپارچه‌سازی آسان داده‌های جدید را بدون پردازش کامل مجموعه داده میسر می‌کند. البته باید به این نکات توجه کرد که IPCA تقریبی از اجزای اصلی را ارائه می‌دهد زیرا بر اساس داده‌های جزئی، برآورد ماتریس همبستگی را به‌روزرسانی می‌کند. هم چنین انتخاب مناسب اندازه دسته امری حیاتی است.
دسته‌های کوچک می‌توانند تخمین‌های نویزی را تولید کنند، در حالی که دسته‌های بسیار بزرگ می‌توانند باعث کاهش مزیت IPCA در زمینه حافظه و محاسبات شوند. نکته جالبت توجه این است که ترتیبی که داده‌ها در آن وارد می‌شوند می‌تواند بر نتایج تأثیرگذار باشد. ترتیب غیرتصادفی ممکن است تعارضاتی را در فرآیند IPCA پدید آورد.



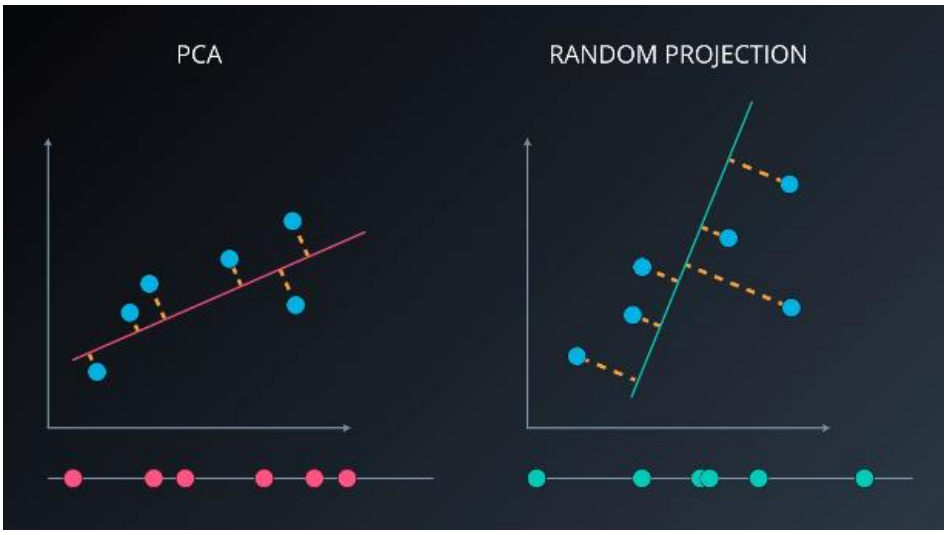
Randomized PCA: PCA تصادفی یک تخمین سریع‌تر از PCA نرمال است. این از تکنیک‌های نمونه‌برداری تصادفی استفاده می‌کند تا principal component یک مجموعه داده را با پیچیدگی محاسباتی کاهش یافته تخمین بزند. PCA تصادفی زمانی مفید است که مجموعه

داده بسیار بزرگ است و نیاز به افزایش سرعت محاسباتی بدون زیاد کردن خطای دقت است. باید توجه داشت که PCA تصادفی یک راه‌حل تقریبی را ارائه می‌دهد و ممکن است principal component دقیقی را مانند PCA سنتی تولید نکند. درجه تقریب وابسته به پارامترهای الگوریتم انتخابی و دقت مورد نیاز برای برنامه خاص است. اگر دقت اجزای اصلی از اهمیت بالایی برخوردار است و منابع محاسباتی کافی هستند، در این صورت به جای PCA معمولی باید در نظر گرفته شود.



Random Projection: هدف اصلی آن کاهش ابعاد است، در حالی که برخی از فواصل یا روابط شباهت بین نقاط داده را حفظ می‌کند. این تکنیک از ماتریس‌های تصادفی استفاده می‌کند تا داده اصلی را بر روی یک فضای کم‌بعد تصویربرداری کند. Random Projection معمولاً در مواردی استفاده می‌شود که حفظ دقیق فواصل یا بازسازی داده برای محاسبات اهمیت زیادی ندارد، اما کاهش ابعاد برای کارایی محاسباتی ضروری است.

مهم است توجه داشت که random projection، تقریبی در نمایش داده معرفی می‌کند و در فرآیند projection احتمالاً از دست رفتن اطلاعات را خواهیم داشت. میزان از دست رفتن به الگوریتم خاص random projection و ابعاد انتخابی برای projection وابسته است. بنابراین، ارزیابی توازن بین کاهش بعد و حفظ ویژگی‌های مهم داده بر اساس نیازهای خاص برنامه، بسیار مهم است.



Does it make sense to chain two different dimensionality reduction algorithms?

وصل کردن دو الگوریتم متفاوت dimensionality reduction به صورت پیاپی میتواند منطقی باشد. به این کار "stacking" or "cascade" of dimensionality reduction می‌گویند. در شرایط متفاوتی میتوان این تکنیک را استفاده کرد که عبارتند از:

Complementary strengths: الگوریتم‌های مختلف ممکن است strengthهای مختلفی در درک جنبه‌های مختلف داده داشته باشند. با ترکیب آن‌ها می‌توان از ویژگی‌های تکمیلی آن‌ها بهره‌برده و نمایش جامع‌تری از داده را استخراج کرد.

Hierarchical reduction: می‌توانید از یک الگوریتم برای انجام کاهش اولیه استفاده کرد و سپس الگوریتم دیگر را روی مجموعه داده کاهش یافته اعمال کرد. این رویکرد سلسله مراتبی می‌تواند به درک ساختارهای سراسری و محلی داده کمک کند و به نمایش دقیق‌تری از داده برسد.

پیش‌پردازش برای الگوریتم‌های خاص: برخی از تکنیک‌های کاهش ابعاد با نوع خاصی از داده یا فرضیات بهتر کار می‌کنند. به وصل کردن الگوریتم‌ها می‌تواند به پیش‌پردازش این داده‌ها کمک کند به نحوی که برای الگوریتم‌های بعدی مناسب‌تر باشد و عملکرد آن‌ها را بهبود بخشد.

البته که مهم است که در هنگام وصل کردن الگوریتم‌های کاهش بعدی به مشکلات احتمالی توجه کرد. این مشکلات شامل افزایش پیچیدگی محاسباتی، احتمال از دست دادن تفسیرپذیری و ریسک overfitting

هستند. بررسی تأثیر هر گام و چک کردن اثربخشی رویکرد وصل کردن الگوریتم ها بر مجموعه داده و مسئله مورد نظر، امری حیاتی است.

What are the main assumptions and limitations of PCA?

PCA یک تکنیک پرکاربرد برای کاهش بعد و تحلیل داده است. هدف آن تبدیل یک مجموعه داده با ابعاد بالا به یک نمایش با بُعد کمتر است، در حالی که الگوها و تغییرات مهم در داده را حفظ می‌کند. ایده اصلی پشت PCA این است که جهت‌هایی، که به عنوان principal component شناخته می‌شوند، که بیشترین تغییر را در داده نشان می‌دهند، را پیدا کند. این principal componentها به یکدیگر عمود هستند، به این معنی که uncorrelated هستند، و مقدار واریانس داده را به ترتیب کمتر می‌کنند. principal component اول بیشترین واریانس را به خود اختصاص می‌دهد، principal component دوم، دومین بیشترین واریانس را، و به همین ترتیب. پیاده سازی PCA مفروضاتی دارد:

خطی بودن: PCA فرض می‌کند که رابطه بین متغیرها در مجموعه داده خطی است. اگر روابط زیرمجموعه‌ای خطی نباشند، PCA نمایش بهینه‌ای از داده ارائه نمی‌دهد.

نرمال بودن: PCA فرض می‌کند که متغیرها در مجموعه داده توزیع نرمال دارند. از نرمال نبودن متغیرها می‌تواند عملکرد PCA را تحت تأثیر قرار دهد، به ویژه اگر این انحرافات قابل توجه باشند.

استقلال: PCA فرض می‌کند که متغیرها در مجموعه داده به یکدیگر وابسته نیستند. اگر وابستگی یا همبستگی قوی بین متغیرها وجود داشته باشد، PCA قدرت توصیف ساختار زیرین را به طور صحیح نمی‌تواند داشته باشد.

همچنین PCA می‌تواند محدودیت هایی برای ما داشته باشد:

حساسیت به داده‌های پرت: PCA به حضور داده‌های پرت در مجموعه داده حساس است. داده‌های پرت می‌توانند تأثیر قابل توجهی روی مؤلفه‌های اصلی داشته باشند و نتایج را تحریف کنند.

تفسیرپذیری: PCA راهی برای کاهش بعد داده فراهم می‌کند، اما ممکن است principal component حاصل، تفسیر مستقیمی درباره متغیرهای اصلی نداشته باشند. تفسیر معنای هر principal component ممکن است چالش برانگیز باشد.

از دست دادن اطلاعات: PCA به دنبال ضبط تغییرات مهم در داده است، اما این فرآیند باعث از دست رفتن اطلاعات کم‌اهمیت‌تر می‌شود. میزان از دست رفتن اطلاعات بستگی به تعداد مؤلفه‌های باقیمانده دارد.

روابط غیرخطی: PCA فرض می‌کند که روابط متغیرها خطی هستند. اگر روابط غیرخطی باشند، PCA نمی‌تواند ساختار زیرین را به طور کامل توصیف کند.

How can clustering be used to improve the accuracy of the linear regression model?

روش های مختلفی برای پاسخ به این سوال وجود دارد:

Feature Engineering: clustering می‌تواند به شناسایی گروه‌ها یا clusterهای معنادار در داده‌ها کمک کند. با اختصاص لیبل cluster به نقاط داده، می‌توان ویژگی‌های جدیدی بر اساس عضویت در cluster ایجاد کرد. این ویژگی‌های مبتنی بر cluster می‌توانند الگوها و روابطی را در داده‌ها که ممکن است تنها با ویژگی‌های اصلی شناسایی نشوند، به خوبی درک کنند. یکی از روش‌های بهبود قدرت پیش‌بینی مدل رگرسیون خطی استفاده از این ویژگی‌های مبتنی بر cluster در مدل رگرسیون خطی است.

شناسایی و حذف داده‌های پرت: الگوریتم‌های clustering می‌توانند داده‌های outlier یا نقاطی را که به طور قابل توجهی از الگوهای کلی در داده‌ها انحراف دارند شناسایی کنند. داده‌های پرت ممکن است بر دقت مدل رگرسیون خطی تأثیر منفی داشته باشند زیرا می‌توانند تأثیر زیادی روی ضرایب تخمین زده شده داشته باشند. با شناسایی و حذف داده‌های پرت با استفاده از تکنیک‌های clustering می‌توان مدل رگرسیون خطی را روی یک زیرمجموعه تمیزتر و نماینده‌تر از داده‌ها آموزش داد که به بهبود دقت منجر می‌شود.

تقسیم بندی داده: clustering می‌تواند به تقسیم داده ها به گروه‌های متمایز بر اساس شباهت‌ها و الگوها کمک کند. این امر مفید است زمانی که بخش‌های مختلف داده روابط خطی متفاوتی داشته باشند یا

ویژگی‌های متفاوتی داشته باشند. با آموزش مدل رگرسیون خطی جداگانه روی هر cluster، می‌توان الگوها و روابط منحصر به فرد در هر بخش را بهتر پیدا کرد که این باعث بهبود دقت نسبت به یک مدل واحد استفاده شده بر روی کل مجموعه داده می‌شود.

Feature selection: clustering می‌تواند در شناسایی ویژگی‌های مرتبط تر برای پیش‌بینی کمک کند. با انجام clustering، می‌توان اهمیت و یا ارتباط ویژگی‌ها در هر cluster را مشاهده کرد. این اطلاعات می‌تواند در انتخاب ویژگی‌ها کمک کند و اجازه دهد تا روی ویژگی‌های مهمتر برای هر cluster تمرکز کرد. با استفاده از تنها ویژگی‌های مهم در مدل رگرسیون خطی، می‌توان نویز را کاهش داده و دقت آن را بهبود بخشید.

How is entropy used as a clustering validation measure?

آنتروپی به عنوان یک معیار اعتبارسنجی clustering می‌تواند برای ارزیابی کیفیت و کارآمدی الگوریتم خوشه بندی مورد استفاده قرار بگیرد. در زمینه clustering، آنتروپی به طور معمول برای ارزیابی همگن بودن یا خلوص clusterها استفاده می‌شود. ایده اصلی در محاسبه آنتروپی، محاسبه آنتروپی نقاط داده برای هر cluster است. اگر بخواهیم به مراحل آن اشاره کنیم میگوییم:

ابتدا clusterها را به دیتا اساین میکنیم. با توجه به الگوریتم clustering، مرحله اول اختصاص نقاط داده به خوشه‌هاست. هر نقطه داده با یک cluster label مرتبط است.سپس برای هر cluster احتمالات تعلق نقاط داده به دسته‌ها یا کلاس‌های مختلف محاسبه می‌شوند. این معمولاً با در نظر گرفتن labelهای کلاس یا دسته‌های موجود در cluster و محاسبه فراوانی‌های نسبی آن‌ها انجام می‌شود. در مرحله بعد آنتروپی یک cluster با استفاده از احتمالاتی که در مرحله قبل به دست آمد، محاسبه می‌شود. آنتروپی میزان عدم قطعیت یا نظم درون cluster را اندازه‌گیری می‌کند. یک cluster با آنتروپی پایین به معنی خلوص بالا است که نقاط داده در آن به طور عمده به یک کلاس یا دسته تعلق دارند. از سوی دیگر، cluster با آنتروپی بالا بیانگر میزان کمتری از خلوص است و نقاط داده به طور متساوی در دسته‌ها یا کلاس‌های مختلف توزیع شده‌اند. برای ارزیابی عملکرد کلی الگوریتم clustering معمولاً مقادیر آنتروپی تمام clusterها تجمیع می‌شوند تا یک معیار کلی بدست آید. این می‌تواند با میانگین وزن‌دار آنتروپی clusterها انجام شود، که وزن‌ها توسط اندازه clusterها

یا تعداد نقاط داده در هر cluster تعیین می‌شوند. در نهایت نیز گام مقایسه و تفسیر انجام میشود. مقدار آنتروپی حاصل، به عنوان یک معیار اعتبارسنجی clustering عمل می‌کند. مقادیر کمتر آنتروپی نتایج بهتری در clustering با خلوص بالاتر را نشان می‌دهند، در حالی که مقادیر بیشتر آنتروپی نشان دهنده عملکرد ضعیف‌تر clustering با خلوص پایینتر هستند.

نکته مهم این است که آنتروپی فقط یکی از معیارهای اعتبارسنجی clustering است که در دسترس است. معیارهای معتبر دیگر شامل ضریب سیلوئت، شاخص دان و شاخص رند میشوند. انتخاب معیار مناسبتر به عهده نیازها و ویژگی‌های خاص داده‌های قرار گرفته در clustering است.

What is label propagation? Why would you implement it, and how? (Extra Point)

یک الگوریتم یادگیری نیمه نظارتی است که برای مسائل classification استفاده می‌شود. این الگوریتم به خصوص زمانی مفید است که داده‌های لیبل‌دار موجود کمتر از داده‌های بدون‌لیبل باشد. الگوریتم لیبل‌ها را از نمونه‌های لیبل‌دار به نمونه‌های بدون‌لیبل با توجه به شباهت آن‌ها در فضای ویژگی منتقل می‌کند. اگر به عملکرد این الگوریتم بپردازیم خواهیم داشت:

ابتدا نمونه‌های داده به عنوان بردارهای ویژگی در فضای بعد بالا نمایش داده می‌شوند که هر ویژگی یا صفت نمونه را نشان می‌دهد. سپس زیرمجموعه‌ای کوچک از نمونه‌های داده با لیبل‌های کلاس یا دسته مربوطه لیبل‌گذاری می‌شوند. این نمونه‌های برچسب‌دار به عنوان نقطه شروع برای پخش لیبل عمل می‌کنند.بعد از آن، الگوریتم شباهت بین هر زوج نمونه بر اساس بردارهای ویژگی آن‌ها محاسبه می‌کند. شباهت‌سنجی‌های رایج شامل فاصله اقلیدسی، شباهت کسینوسی و یا شباهت مبتنی بر کرنل هستند. با استفاده از شباهت بین نمونه‌ها، الگوریتم شروع به پخش لیبل از نمونه‌های لیبل‌دار به نمونه‌های بدون لیبل می‌کند. ایده این است که نمونه‌هایی که به نمونه‌های لیبل‌دار شبیه هستند، احتمالاً در یک کلاس مشترک قرار دارند. فرآیند پخش لیبل به صورت تکراری انجام می‌شود. در هر تکرار، الگوریتم لیبل‌های نمونه‌های بدون لیبل را بر اساس لیبل‌های نمونه‌های همسایه خود به روز می‌کند. به‌طوری‌که به میزان مساهمت وزن‌دار لیبل‌های نمونه‌های همسایه توسط شباهت بین نمونه‌ها توجه شود. فرآیند تکراری پخش لیبل تا رسیدن به یک معیار همگرایی ادامه

می‌یابد. این معیار می‌تواند تعداد تعیین شده‌ای از تکرارها، آستانه‌ای برای تغییر لیبل یا یک شرط خاص مبتنی بر دینامیک پخش باشد. هنگامی که فرآیند پخش لیبل به همگرایی می‌رسد، لیبل‌های همه نمونه‌ها، هم لیبل‌دار و هم بدون لیبل تعیین می‌شود. الگوریتم لیبل کلاس‌ها را به نمونه‌های بدون لیبل بر اساس لیبل‌های پخش شده اختصاص می‌دهد. از الگوریتم از جهتی مفید است که با بهره‌گیری از شباهت بین نمونه‌ها، لیبل‌ها را از نمونه‌های لیبل‌دار به نمونه‌های بدون لیبل منتقل می‌کند و اجازه می‌دهد که نمونه‌های بدون لیبل از داده‌های لیبل‌دار محدود بهره‌برداری کنند.

دلایل زیادی برای انجام این الگوریتم میتوانیم داشته باشیم:

1. این الگوریتم زمانی مفید است که تعداد داده‌های لیبل‌دار موجود برای آموزش محدود یا کافی نیست. با بهره‌گیری از شباهت بین نمونه‌های لیبل‌دار و نمونه‌های بدون لیبل الگوریتم می‌تواند از داده‌های لیبل‌دار موجود به طور موثر استفاده کند و پیش‌بینی‌هایی برای نمونه‌های بدون لیبل ارائه دهد.

2. این الگوریتم یک تکنیک محبوب در بسترهای یادگیری نیمه‌نظارتی است که ترکیبی از داده‌های لیبل‌دار و بدون لیبل را در اختیار دارد. این روش به داده‌های بدون لیبل اجازه می‌دهد در فرآیند یادگیری مشارکت کنند و با استفاده از لیبل‌های پخش شده، عملکرد مدل را بهبود بخشند.

3. در داده‌های با ابعاد بالا، لیبل‌گذاری یک تعداد زیادی نمونه با لیبل ممکن است مشکل باشد. label propagation راهی برای بهره‌برداری موثرتر از داده‌های لیبل‌دار موجود است، با اینکه لیبل‌ها را به نمونه‌های بدون لیبل به صورت یکپارچه و سازگاری منتقل می‌کند.

4. این الگوریتم می‌تواند در وظایفی مانند تشخیص اجتماع یا clustering مورد استفاده قرار گیرد، جایی که هدف گروه‌بندی نمونه‌های مشابه است. با label propagation براساس شباهت، امکان تشخیص clusterها یا اجتماعات در داده‌ها فراهم می‌شود.

5. این الگوریتم می‌تواند به عنوان بخشی از یک چارچوب یادگیری فعال استفاده شود، جایی که الگوریتم به صورت فعال نمونه‌ها را براساس امتیازهای uncertainty یا confidence برای لیبل‌گذاری انتخاب می‌کند. پخش برچسب می‌تواند برای نمونه‌های بدون لیبل، لیبل‌های اولیه ارائه کند و به فرآیند یادگیری فعال کمک کند که با یک مجموعه اولیه از داده‌های لیبل‌دار شروع می‌شود.

اگر بخواهیم یک توضیح مختصری از پیاده سازی این الگوریتم دهیم خواهیم داشت:

ابتدا داده را به صورت یک گراف نمایش میدهیم که هر نقطه داده نود باشد و یال‌ها روابط یا شباهت‌های بین نقاط داده را نشان دهند. گراف‌های معمول شامل گراف k-nearest neighbor یا گراف‌های similarity هستند. سپس با یک زیرمجموعه از نقاط داده که برچسب‌های معتبری دارند شروع میکنیم. این نقاط داده با لیبل‌های کلاس مربوطه خود مقداردهی شوند. حال لیبل‌ها را از نقاط داده لیبل‌دار به نقاط داده بدون لیبل به صورت تکراری پخش میکنیم. لیبل‌ها بر اساس شباهت یا نزدیکی بین نقاط داده در گراف پخش می‌شوند. نقاط داده همسایه لیبل یکدیگر را تحت تأثیر قرار می‌دهند و این فرآیند تا همگرایی

تکرار می‌شود. لیبل‌های نقاط داده بدون لیبل را بر اساس لیبل‌های پخش شده از نقاط داده همسایه به‌روزرسانی کنید. این به‌روزرسانی می‌تواند به روش‌های مختلفی انجام شود، مانند میانگین وزن‌دار یا با در نظر گرفتن روش انتخاب بر اساس لیبل‌های نقاط داده همسایه. پروسه پخش لیبل و به‌روزرسانی را تا زمانی که لیبل‌های نقاط داده بین تکرارها به طور قابل توجهی تغییر نکنند یا تا زمانی که شرط توقف تعیین شده برآورده شود، تکرار کنید. معمولاً الگوریتم وقتی همگرا می‌شود که لیبل‌های نقاط داده دیگر به طور قابل توجهی بین تکرارها تغییر نکند. پس از همگرایی الگوریتم، لیبل‌های پخش شده نقاط داده بدون لیبل را می‌توان به عنوان لیبل‌های پیش‌بینی شده در نظر گرفت. این لیبل‌های استنتاج شده می‌توانند برای تحلیل‌های بیشتر مانند classification، clustering و غیره استفاده شوند. Label propagation نیاز به تنظیم دقیق پارامترها دارد، مانند تعداد همسایگان برای محاسبه، روش وزن‌دهی و معیار توقف.