K نزدیک ترین همسایه

استاد راهنما : دکتر وحید رضایی تبار

ارائه دهندگان : حسام افشار – مهراب کیانی

فهرست مطالب

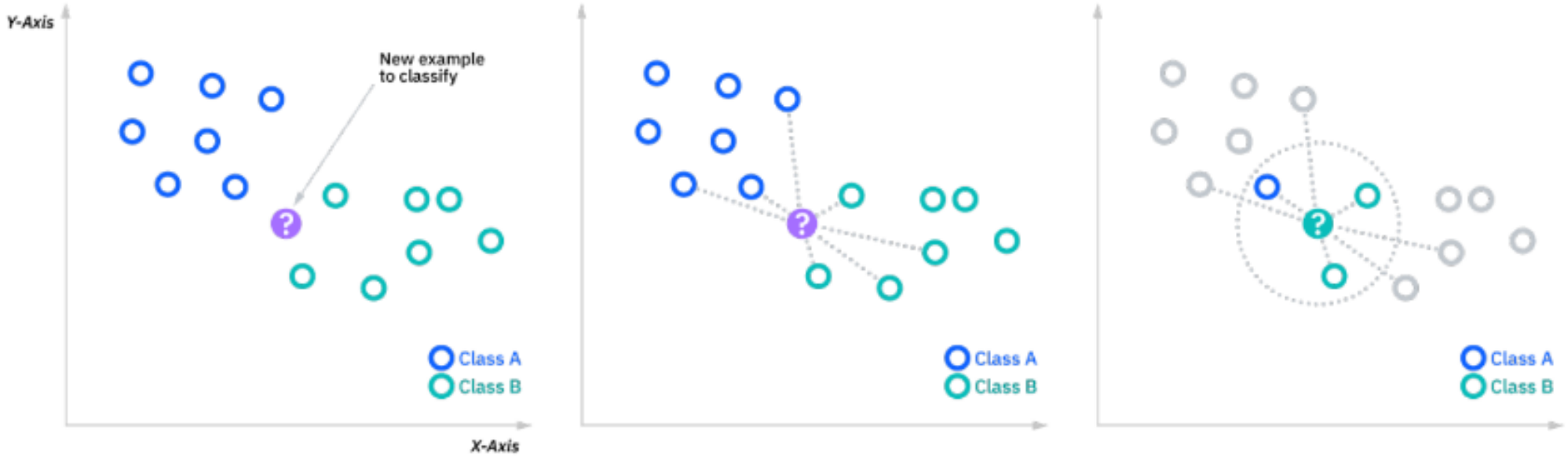
مقدمه

الگوریتم **k نزدیکترین همسایه ها**[1]، همچنین به عنوان KNN یا k-NN شناخته می شود، یک طبقه بندی کننده یادگیری ناپارامتریک و نظارت شده است که از تشابه برای انجام طبقه بندی یا پیش بینی در مورد گروه بندی یک نقطه داده فردی استفاده می کند. در حالی که می‌توان از آن برای مسائل رگرسیون یا طبقه‌بندی استفاده کرد، معمولاً به عنوان یک الگوریتم طبقه‌بندی استفاده می‌شود، و از این فرض استفاده می‌کند که نقاط مشابهی را می‌توان در نزدیکی یکدیگر یافت.

1- K Nearest Neighbors

نحوه کار الگوریتم

برای مشکلات طبقه بندی، یک برچسب کلاس بر اساس تعداد اکثریت اختصاص داده می شود. یعنی برچسبی که بیشتر در اطراف یک نقطه داده معین نشان داده می شود انتخاب می شود.



نحوه کار الگوریتم

مسائل رگرسیون از مفهومی مشابه به عنوان مسئله طبقه‌بندی استفاده می‌کنند، اما در این مورد، میانگین k نزدیک‌ترین همسایه‌ها برای پیش‌بینی گرفته می‌شود. تمایز اصلی در اینجا این است که طبقه بندی برای مقادیر گسسته استفاده می شود، در حالی که رگرسیون برای مقادیر پیوسته استفاده می شود. با این حال، قبل از انجام یک طبقه بندی، فاصله باید تعریف شود. فاصله اقلیدسی بیشتر مورد استفاده قرار می گیرد که در ادامه بیشتر به آن می پردازیم.

نحوه کار الگوریتم

برای تعیین اینکه کدام نقاط داده به یک نقطه مورد بررسی برای پیشبینی نزدیکتر هستند، فاصله بین نقطه مورد بررسی و سایر نقاط داده باید محاسبه شود. این معیارهای فاصله به شکل گیری مرزهای تصمیم کمک می کند.

فاصله اقلیدسی[1] متداول ترین اندازه گیری فاصله است و به بردارهای با ارزش واقعی محدود می شود. با استفاده از فرمول زیر، یک خط مستقیم بین نقطه مورد بررسی و نقطه دیگر اندازه گیری می شود.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

1-Euclidean distance

نحوه کار الگوریتم

فاصله منهتن[1] نیز یکی دیگر از معیارهای محبوب فاصله است که قدر مطلق بین دو نقطه را اندازه گیری می کند و به صورت زیر محاسبه میشود.

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i|$$

فاصله مینکوفسکی[2] شکل تعمیم یافته معیارهای فاصله اقلیدسی و منهتن است. پارامتر p در فرمول زیر امکان ایجاد سایر معیارهای فاصله را فراهم می کند. فاصله اقلیدسی با این فرمول نشان داده می شود که p برابر با دو باشد و فاصله منهتن با p برابر با یک نشان داده شود.

$$d(x,y) = (\sum_{i=1}^{n} |x_i - y_i|)^{1/p}$$

1- Manhattan distance
2- Minkowski distance

مدل وزن دار

پس از محاسبه تمام فواصل و یافتن K نزدیک ترین فاصله ها همانطور که قبلاً نیز گفته شد، باید از یک الگوریتم رأی گیری برای تعیین کلاس پیش بینی شده استفاده کرد. یک حالت ساده این بود که داده را به طبقه ای که بیشترین تعداد را در k نزدیک ترین همسایه دارد نسبت دهیم. روش دیگر استفاده از الگوریتم وزنی است. به این صورت که اگر فاصله هر یک از اعضای k همسایگی $d(x_i,y)$ باشد آنگاه وزن هر یک از اعضا به صورت زیر تعریف میشود:

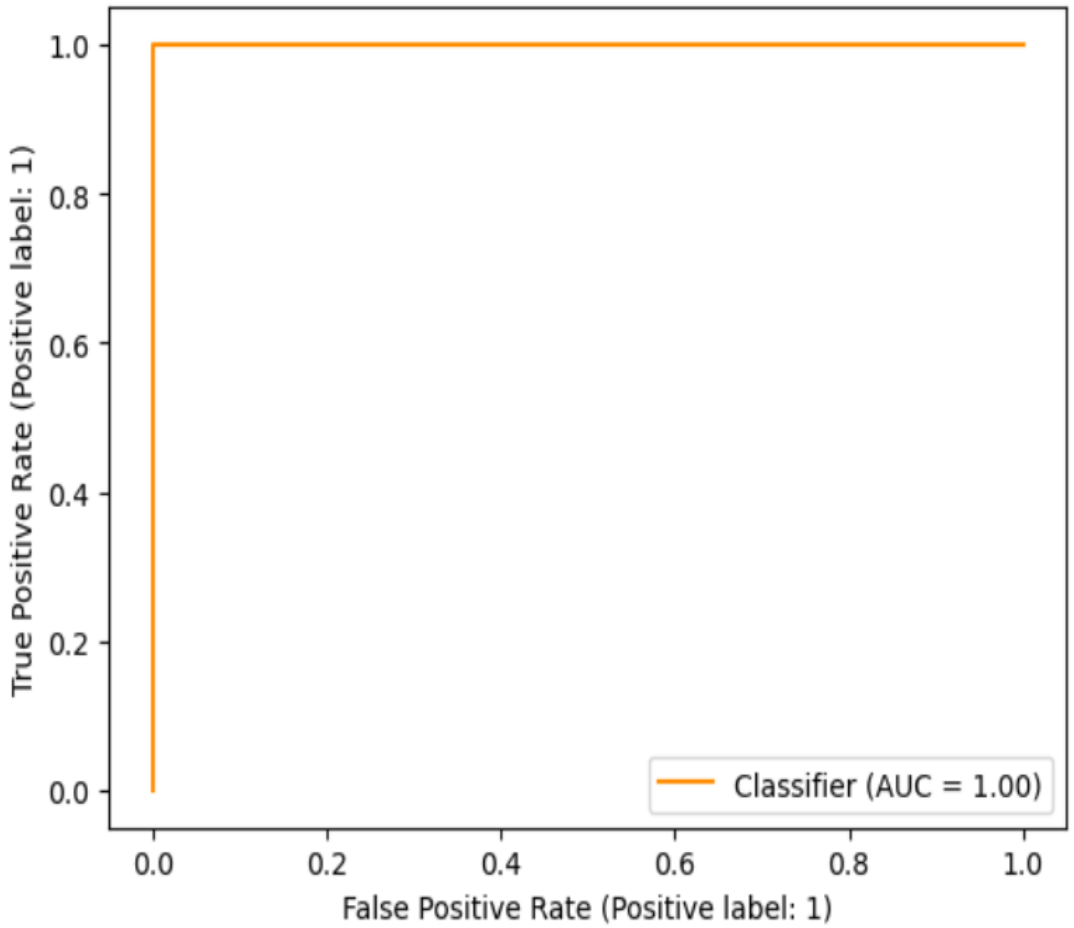
$$w_i = \frac{\frac{1}{d(x_i,y)}}{\sum_{i=1}^{k}(\frac{1}{d(x_i,y)})}$$

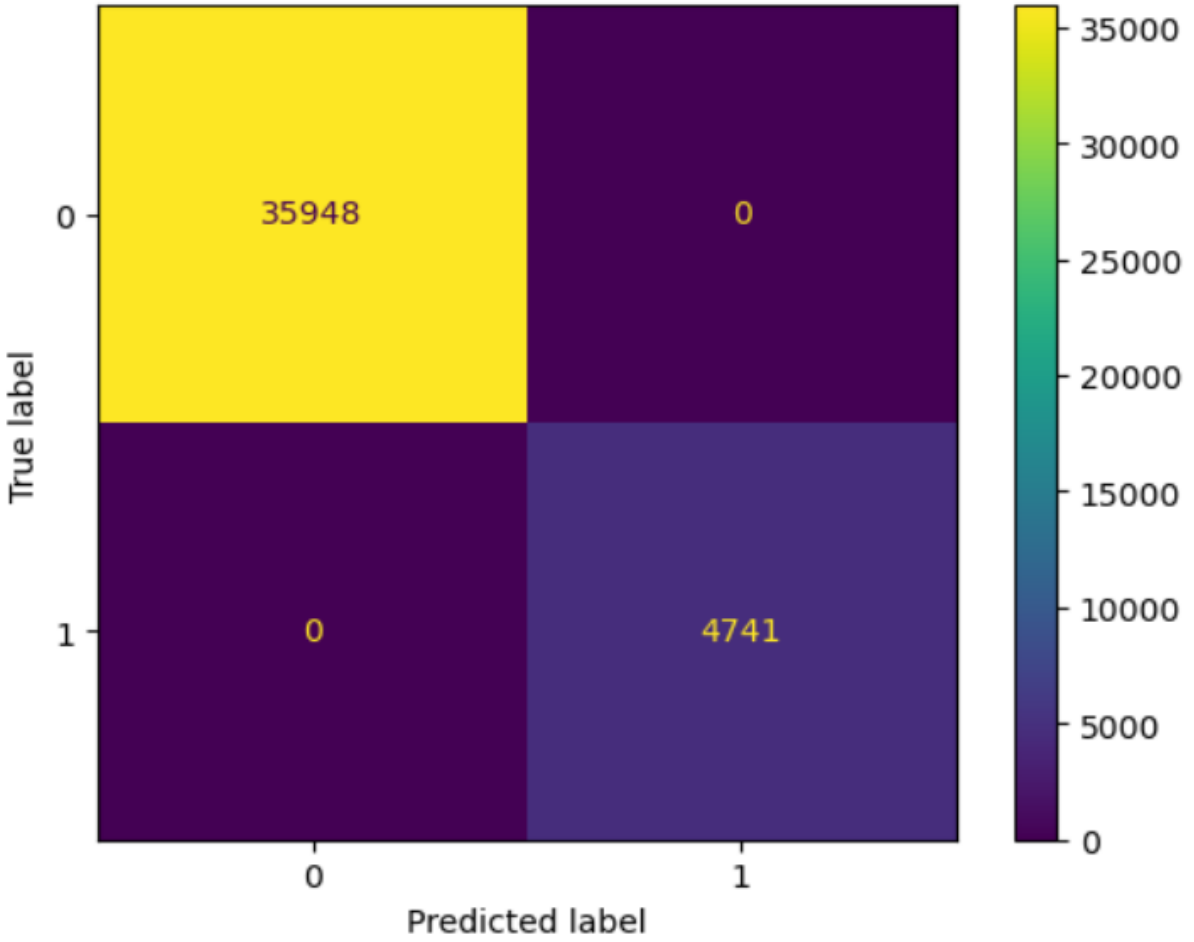
سپس مجموع وزن ها برای هر طبقه موجود در k همسایگی را حساب میکنیم. طبقه ای که بیشترین مقدار را بگیرد به عنوان برچسب داده جدید انتخاب میکنیم.

حال مدل درخت تصمیم را بر روی داده های بانک که داشتیم پیاده سازی میکنیم تا نتایج آن را بدست بیاوریم و با مدل های دیگر مقایسه کنیم. ابتدا با استفاده از داده های آموزش که در بخش رگرسیون لوژستیک تقسیم بندی کردیم بهترین پارامتر های مدل که همان K است را با استفاده از اعتبار سنجی متقابل میابیم.

بهترین مدل بدست آمده مدلی با 74 نزدیک ترین همسایه ها است. مدل را بر روی داده های آموزش ارزیابی کرده و نمودار Roc و ماتریس درهم ریختگی آن به صورت زیر بدست آمده است.

پروژه عملی


True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)
Classifier (AUC = 1.00)
0.0
0.2
0.4
0.6
0.8
1.0


True label
Predicted label
35948
0
0
4741
0
1
35000
30000
25000
20000
15000
10000
5000
0

پروژه عملی

همانطور که میبینیم دقت بدست آمده برای داده های آموزش 100 درصد است که از همه مدل ها تا به الان بهتر است اما دقت مدل برای داده های آزمون 88 درصد است که به نسبت از مدلهای دیگر کمتر است.

فایل داده ها و کد تحلیل های انجام شده با پایتون در لینک زیر قرار داده شده:

https://github.com/hesamafshar/Classification/tree/main/Banking%20Dataset%20-%20Marketing%20Targets