بخش کتبی CA 4

امیرعلی شهریاری
۸۱۰۱۰۰۱۷۳

سوال اول)

۱: چنانچه حجم آوری داده ای که در نظر است، وجود نداشته باشد. می توان همانجه داده گمشده را با ویژگی های ساختاری تحت عنوان Feature engineering ویژگی های جدید را استخراج کرد و همچنین بعنوان راه حل اصلی و کلی این مشکل Imputation ها تکنونه که در بخش عملی نیز انجام دادیم، مقادیر گمشده را با راهکارهای مانند میانگین، میانه یا KNN پر می کنیم.

۲: می توان با oversampling نقاط داده از کلاس اقلیت را تکرار کرد تا با مجموعه داده، تعادل بیشتر برقرار بشود، همچنین می توان نقاط داده را از کلاس اکثریت کم کرد تا با سایز کلاس اقلیت سازگاری داشته باشد؛ همچنین می توان با دخالت دادن وزن ها، وزن بالاتری به کلاس اقلیت اختصاص داد تا توجه به نمونه های کم تکرار تر هم افزایش پیدا کند.

۳: با روش های آماری یا دانش دامنه کار می توان موارد پرت را حذف کرد، همچنین استفاده از روش های Smoothing نظیر میانگین می توان نویز را هموار سازی کرد، راه دیگر هم استفاده از روش هایی است که نویز در آنها تاثیر کمتری دارد مثل درخت های تصمیم یا Random forest

۴: می توان از تکنیک های کاهش بعد نظیر PCA استفاده کرد همچنین با Feature selection می توان کار ساز باشد که زیر مجموعه ای از مفید ترین ها که ارتباط چندانی با یکدیگر ندارند را انتخاب کند منجر به کاهش افزونگی می شود. همچنین روش های منظم سازی که به ما اجازه داشتن روابط پیچیده و متکی به داده ها این وابستگی را کاهش می دهد.

سوال ۲:

نمره آزمون = ساعت مطالعه‌ای $\times a_1$ + آزمون دادن $\times a_1$ + θ_0

می توان به صورت فوق آزمون دادن را با ضریب a_1 به آن افزود و برای یافتن a_1 از روش Least square method به این صورت عمل می کنیم که a را یافته که مجموع مجذور خطای میان نمرات امتحانی پیش بینی شده و نمرات واقعی به حداقل برسد و شیوه انجام آن، نخست محاسبه مجذور خطا برای هر نقطه سپس مجذور خطا برای همه نقاط را محاسبه کرده و مقدار a را تنظیم کرده و اینبار مجدداً مجموع مربعات خطاها را محاسبه می کنیم و این فرآیند همی تکرار می شود تا ترکیبی از ضرایب یافته تا مجذور خطاها کمینه شود.
در روش گرادیان نزولی ضریب از راه های بهینه سازی بوده. جهت گرادیان را محاسبه کرده به گونه ای که مجموع مربعات خطاها کمترین افزایش را دارد و ضرایب را در جهت مخالف تنظیم می کند این روند تا رسیدن به کمینه ادامه میابد از روش های دیگر منظم سازی را میتوان بیان کرد مانند برخی رگرسیون ها یک عبارت جریمه ای به تابع هدف اضافه کرده که با جریمه دادن به مقادیر ضرایب بزرگ از بیش از بزرگ آنها جلوگیری می کند

سوال سوم:

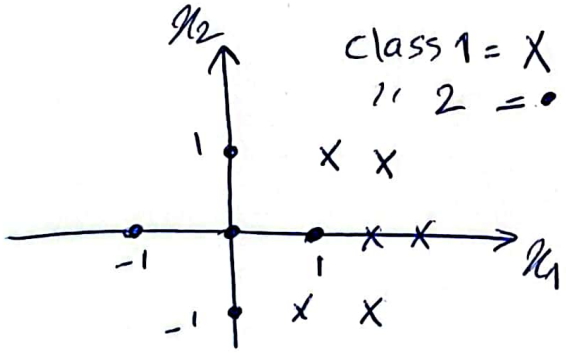
Predicted		Actual: Spam	Actual: Not Spam
	Spam	۳۰۰	۳۰
	Not Spam	۲۰	۲۰۰

$$Recall = \frac{۳۰۰}{(۳۰۰+۲۰)} = ۹۳/۷\ \%$$

$$Precision = \frac{۳۰۰}{(۳۰۰+۳۰)} = ۹۰/۹\ \%$$

$$Accuracy = \frac{۳۰۰+۲۰۰}{(۳۰۰+۲۰+۳۰+۲۰۰)} = ۹۱\ \%$$

$$F1\text{-}score = \frac{۲\times(۰/۹۰۹\times۰/۹)}{(۰/۹۰۹+۰/۹)} = ۹۲\ \%$$



KNN:

سوال اول:

داده تست (۰ ، ۰/۵)

۱) فاصله اقلیدسی:

خواص اقلیدسی نقطه تست را از هرکدام از نقاط کلاس‌های بیابیم

کلاس ۱:
- (۱،۱) = ۱/۴
- (۲،۰) = ۱/۵
- (۲،۱) = ۱/۴
- (۱،-۱) = ۱/۱ ✓
- (۲،-۱) = ۱/۴
- (۳،۰) = ۲/۲

کلاس ۲:
- (۰،۰) = ۰ ✓
- (-۱،۰) = ۱/۵
- (۰،۱) = ۱
- (۱،۰) = ۰/۵ ✓
- (۰،-۱) = ۱

با توجه به $k=۳$ نزدیک‌ترین همسایه‌ها ✓ مشخص شدند و چون دوتای آنها متعلق به کلاس ۲ بود پس نقطه کلاس ۲ طبقه‌بندی می‌شود.

۲) فاصله منهتن:

کلاس ۱:
- (۱،۱) = ۱/۵
- (۲،۰) = ۲
- (۲،۱) = ۲/۵
- (۱،-۱) = ۲ ✓
- (۲،-۱) = ۲/۵
- (۳،۰) = ۲/۵

کلاس ۲:
- (۰،۰) = ۰ ✓
- (-۱،۰) = ۱/۵
- (۰،۱) = ۱
- (۱،۰) = ۰/۵ ✓
- (۰،-۱) = ۱

نقطه ✓ خورده ۳ نقطه با نزدیک‌ترین فاصله هستند که مجددا این نیز به کلاس ۲ طبقه‌بندی می‌گردد.

SVM

سوال اول:

- supportVector ها نزدیک ترین نقطه به مرز تصمیم (hyperPlane) هستند که بین کلاس‌های مختلف قرار دارند این نقاط برای تعیین مرز تصمیم‌گیری و تعیین حاشیه های طبقه کننده ضروری هستند.

مثلاً در همان مثال ایمیل Ham و spam ایمیل های هرزنامه +1 و غیر آن -1 در SVM یک ابر صفحه می یابد که هر کلاس را با حداکثر حاشیه جدا می‌کند، این حاشیه فاصله بین ابر صفحه و نزدیک ترین نقاط داده از هر کلاس است این نزدیک ترین نقاط که حاشیه را تعریف می کند supportVector هستند.

- برای داده های بسیار بزرگ - داده های با ابعاد زیاد - داده های غیر خطی با مرزهای تصمیم‌گیری پیچیده - داده های نامتعادل که یک کلاس از کلاس دیگر بسیار پیشتر است - داده های نویزی

- آنها داده های غیر خطی قابل جداسازی را طبقه بندی می کنند. یک تابع کرنل برای نگاشت داده های ورودی از فضای ویژگی اصلی به فضایی با ابعاد بالاتر استفاده می‌شود و کلاس ها به صورت خطی قابل تفکیک می باشند و این تبدیل به آن اجازه می دهد که یک ابر صفحه بسازد و کلاس ها را با حداکثر حاشیه جدا کند.

- hard SVM ها هدف کمینه کردن نرم weightVector است و یا ضمن ابر صفحه ماکزیمم حاشیه کلاس ها را در داده های آموزشی جدا می کند بدون هرگونه margin violation

soft SVM: (قرینات بخش قبلی را با اجازه نقض برخی حاشیه و طبقه بندی.

تفاوت اصلی آنها در برخورد با حاشیه ها و طبقه بندی ها است در hard هیچگاه، نقض را نمی پذیرد و هدف یافتن یک ابر صفحه به گونه‌ای که کلاس‌ها را بدون هیچگونه طبقه بندی نادرست از هم جدا می کند ولی soft با معرفی یک پارامتر برخی موارد نقض و اشتباه را می پذیرد. در hard به نویز ها و موارد پرت بسیار حساس تر است ولی soft انعطاف پذیرتر است.