به نام خدا

دانشگاه تهران

پردیس دانشکده های فنی

دانشکده مهندسی برق و کامپیوتر

تمرین شماره‌ی 4

بخش کتبی

مدرسین:

دکتر فدائی

دکتر یعقوب‌زاده

نگارش:

فاطمه محمدی 810199489

بخش کتبی

مبحث اول

سوال اول

1) وجود نداشتن یک یا چند ویژگی در داده‌های آموزش

- تخمین مقادیر گمشده: استفاده از روش‌های تخمین مانند میانگین، میانه، یا مد برای پر کردن مقادیر گمشده.
- استفاده از الگوریتم‌های پیشرفته: الگوریتم‌هایی مانند K-Nearest Neighbors (KNN) یا Multiple Imputation by Chained Equations (MICE) برای تخمین مقادیر گمشده.
- حذف ویژگی‌های ناقص: در صورتی که درصد زیادی از داده‌ها گمشده باشند، ممکن است بهتر باشد ویژگی‌های مذکور حذف شوند.
- استفاده از مدل‌های مقاوم به داده‌های گمشده: برخی مدل‌ها می‌توانند با داده‌های ناقص کار کنند، مانند درخت‌های تصمیم یا الگوریتم‌های مبتنی بر تجمع.

2) نامتعادل بودن توزیع داده‌ها در کلاس‌ها:

- روش‌های نمونه‌برداری: استفاده از روش‌های نمونه‌برداری مانند Oversampling (افزایش نمونه‌های کلاس‌های کمتر)، Undersampling (کاهش نمونه‌های کلاس‌های بیشتر)، یا روش‌های ترکیبی مانند SMOTE (Synthetic Minority Over-sampling Technique).
- استفاده از الگوریتم‌های مقاوم به عدم توازن: برخی الگوریتم‌ها مانند Random Forests یا الگوریتم‌های مبتنی بر Boosting می‌توانند با توزیع‌های نامتعادل بهتر کار کنند.
- استفاده از معیارهای ارزیابی مناسب: استفاده از معیارهایی مانند F1-Score، Precision-Recall Curve، و ROC-AUC به جای Accuracy برای ارزیابی مدل.

3) وجود نویز در داده‌ها:

- پیش‌پردازش داده‌ها: استفاده از تکنیک‌های پاکسازی داده‌ها مانند فیلترهای نویز، تکنیک‌های آماری برای شناسایی و حذف داده‌های نویزی.
- استفاده از مدل‌های مقاوم به نویز: برخی مدل‌ها مانند درخت‌های تصمیم یا الگوریتم‌های مبتنی بر تجمع می‌توانند به نویز مقاوم باشند.
- استفاده از تکنیک‌های کاهش نویز: تکنیک‌هایی مانند PCA (Principal Component Analysis) یا ICA (Independent Component Analysis) برای کاهش نویز و افزایش کیفیت داده‌ها.

4) وجود ویژگی‌های همبسته (**correlated**):

- حذف ویژگی‌های همبسته: شناسایی ویژگی‌های همبسته با استفاده از ماتریس همبستگی و حذف یکی از ویژگی‌های همبسته.
- استفاده از تکنیک‌های کاهش بعد: استفاده از تکنیک‌هایی مانند PCA برای کاهش بعد و انتخاب ترکیباتی از ویژگی‌ها که همبستگی کمتری با هم دارند.
- استفاده از الگوریتم‌های مقاوم به همبستگی: برخی الگوریتم‌ها مانند Lasso Regression می‌توانند با ویژگی‌های همبسته بهتر کار کنند.

سوال دوم

مشاور تحصیلی می‌تواند نقش آزمون دادن را با اضافه کردن یک ویژگی جدید به مدل رگرسیون خطی خود در نظر بگیرد. فرض کنیم این ویژگی جدید "تعداد آزمون‌های تمرینی" باشد. معادله رگرسیون خطی جدید به صورت زیر خواهد بود:

$$\{\text{نمره ازمون}\} = \beta_0 + \beta_1 * \{\text{ساعت مطالعاتی}\} + \beta_2 * \{\text{تعداد ازمونهای تمرینی }\}$$

در این معادله:

- β_0 ثابت یا مقدار ثابت مدل است.
- β_1ضریب مربوط به ساعت مطالعاتی است.
- β_2 ضریب مربوط به تعداد آزمون‌های تمرینی است.

(Least Squares Method)

برای پیدا کردن ضرایب مناسب β_i، مشاور می‌تواند از روش LSM استفاده کند. این روش سعی می‌کند ضرایبی را پیدا کند که مجموع مربعات اختلافات بین مقادیر پیش‌بینی شده و مقادیر واقعی را به حداقل برساند. این اختلافات به صورت زیر تعریف می‌شود:

$$SSE = \sum_{i=0}^{n} \left(y_i - \left(\beta_0 + \beta_1 x_{1_i} + \beta_2 x_{2_i}\right)\right)^2$$

که در آن y_i نمره واقعی، x_{1_i} ساعت مطالعاتی و x_{2_i} تعداد آزمون‌های تمرینی دانشجوی i ام است.

برای پیدا کردن ضرایب بهینه، باید مشتق جزئی این تابع نسبت به هر یک از ضرایب β_i را محاسبه کرده و آن‌ها را برابر صفر قرار داد. سپس این معادلات را حل کرد تا ضرایب بهینه به دست آیند.

(Gradient Descent)

روش دیگر برای پیدا کردن ضرایب بهینه، استفاده از روش (Gradient Descent) است. در این روش، ابتدا یک مجموعه از مقادیر اولیه برای ضرایب انتخاب می‌شود و سپس در هر مرحله، ضرایب به گونه‌ای به‌روزرسانی می‌شوند که مقدار تابع هزینه (در اینجا، مجموع مربعات خطا) کاهش یابد. به‌روزرسانی ضرایب به صورت زیر انجام می‌شود:

$$\beta j := \beta j - \alpha \frac{\partial}{\partial \beta_j} SSE$$

که در آن α نرخ یادگیری است و $\frac{\partial}{\partial \beta_j} SSE$ مشتق جزئی تابع هزینه نسبت به ضریبβ_jاست.

تکنیک‌های دیگر

علاوه بر روش‌های فوق، تکنیک‌های دیگری نیز برای بهینه‌سازی تابع هزینه وجود دارند، مانند:

- روش‌های بهینه‌سازی عددی: مانند BFGS یا L-BFGS.
- الگوریتم‌های تکراری: مانند الگوریتم‌های ژنتیک یا شبیه‌سازی تبرید.
- روش‌های مبتنی بر ماشین لرنینگ: مانند الگوریتم‌های مبتنی بر Boosting یا الگوریتم‌های پیشرفته‌تر مانند روش‌های Bayesian.

هر یک از این روش‌ها می‌تواند در شرایط خاصی مفید باشد و انتخاب بهترین روش بستگی به داده‌ها و پیچیدگی مدل دارد.

سوال سوم

Confusion Matrix:

	True	False
Positive	300	30
Negative	200	20

Recall*:* $\frac{\text{TP}}{\text{TP+FN}} = \frac{300}{300+20} = \frac{15}{16} = 0.94$

Precision: $\frac{TP}{TP+FP} = \frac{300}{300+30} = \frac{10}{11} = 0.91$

Accuracy: $\frac{TP+TN}{TP+FP+TN+FN} = \frac{300+200}{300+30+200+20} = \frac{500}{550} = \frac{10}{11} = 0.91$

F1: $\frac{2*Precision*Recall}{Precision+Recall} = \frac{66}{71} = 0.93$

KNN

سوال اول

1. فاصله اقلیدسی

$$d = (x2 - x1)^2 + (y2 - y1)^2$$

2. فاصله منهتن

$$d = | x2 - x1 | + | y2 - y1 |$$

Point	Class	Euclidean Distance	Manhattan Distance
(2, 0)	1	1.5	1.5
(3, 0)	1	2.5	2.5
(1, 1)	1	1.118	1.5
(2, 1)	1	1.803	2.5
(1, -1)	1	1.118	1.5
(2, -1)	1	1.803	2.5
(0, 0)	2	0.5	0.5
(0, 1)	2	1.118	1.5
(0, -1)	2	1.118	1.5
(-1, 0)	2	1.5	1.5
(1, 0)	2	0.5	0.5

نزدیک ترین نقاط:

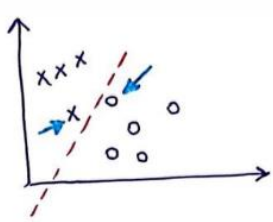
1. اقلیدسی: (0,0)و (1,0)و (1,1) -> دو تا از کلاس 2 داریم: جواب کلاس 2
2. منهتن: (2,0)و (1,0)و (1,1) -> دو تا از کلاس 2 داریم: جواب کلاس 2

Support Vector Machine

سوال اول

1) به چه نقاطی **support vector** گفته می شود و آن را روی مثالی دلخواه نمایش دهید.

به نقاطی که در تعیین مرز تصمیم‌گیری در الگوریتم‌های svm نقش اساسی دارند، support vector یا بردار پشتیبان گفته می‌شود. این نقاط نزدیک‌ترین نمونه‌ها به مرز تصمیم‌گیری هستند و تعیین‌کننده این مرز می‌باشند. برای مثال، فرض کنید دو کلاس از داده‌ها داریم که با یک خط جدا شده‌اند. نقاطی که به این خط نزدیک‌تر هستند و در تعیین موقعیت این خط نقش دارند، بردارهای پشتیبان می‌باشند.



در این مثال، نقاطی که به خط تصمیم‌گیری (مرز بین دو کلاس) نزدیک‌تر هستند (نقاط x و o نزدیک به خط)، بردارهای پشتیبان می‌باشند.

2) به نظر شما طبقه بند **SVM** برای طبقه بندی چه نوع داده هایی مناسب نیستند؟

طبقه‌بند SVM برای داده‌هایی که به شدت نویزی یا به هم ریخته هستند مناسب نیست. همچنین، برای داده‌هایی که تعداد ویژگی‌های آنها بسیار بیشتر از تعداد نمونه‌ها است (مانند داده‌های زیستی)، ممکن است SVM عملکرد مطلوبی نداشته باشد. به علاوه، برای داده‌هایی که به صورت غیرخطی قابل تفکیک هستند و نیاز به یک کرنل پیچیده دارند، استفاده از SVM می‌تواند چالش‌برانگیز باشد.

درباره **kernel** ها و نقش آن ها در طبقه بندی توضیح دهید. (توضیح دهید وظیفه **kernel** ها چیست و چجوری به طبقه بندی کمک می کنند)

کرنل‌ها توابعی هستند که داده‌ها را از فضای ویژگی اولیه به یک فضای ویژگی بالاتر منتقل می‌کنند. این کار به SVM امکان می‌دهد تا داده‌هایی را که در فضای اولیه به صورت خطی قابل تفکیک نیستند، در فضای ویژگی بالاتر به صورت خطی تفکیک‌پذیر شوند.

وظیفه کرنل‌ها:

- تبدیل داده‌ها: کرنل‌ها داده‌ها را به یک فضای ویژگی بالاتر می‌برند.
- محاسبه آسان: با استفاده از ترفند کرنل، SVM می‌تواند به جای محاسبه مستقیم در فضای ویژگی بالاتر، محاسبات را به صورت غیرمستقیم و با پیچیدگی کمتر انجام دهد.

4) تفاوت **hard svm classifier** با **soft svm classifier** بیان کنید.

hard svm classifier

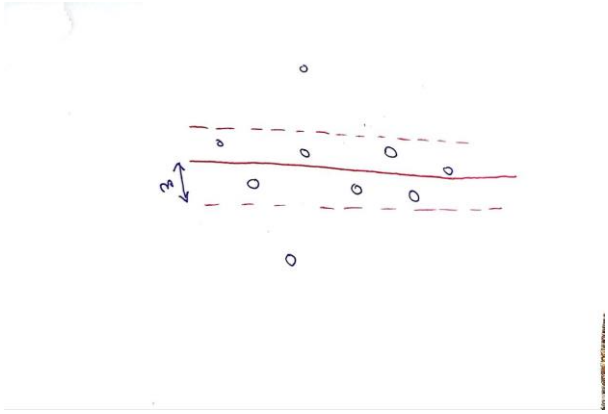
- تنها برای داده‌هایی مناسب است که به طور کامل قابل تفکیک هستند.
- به هیچ گونه خطا یا نویزی در داده‌ها اجازه نمی‌دهد.
- مدل بهینه با حداکثر حاشیه‌ای که تمام داده‌ها را به درستی طبقه‌بندی کند، به دست می‌آید.

soft svm classifier

- برای داده‌هایی مناسب است که به طور کامل قابل تفکیک نیستند یا دارای نویز هستند.
- اجازه می‌دهد برخی از داده‌ها در طرف نادرست مرز تصمیم‌گیری قرار گیرند.
- مدل بهینه با در نظر گرفتن یک متغیر خطا (C) که تعادل بین حداکثر حاشیه و خطاهای طبقه‌بندی را برقرار می‌کند، به دست می‌آید.

5) نحوه استفاده از **SVM** در مسائل رگرسیونی را با کشیدن شکل توضیح دهید.

SVM در مسائل رگرسیونی به نام SVM Regression (SVR) شناخته می‌شود. در SVR، هدف پیدا کردن یک تابع است که بیشترین تعداد نقاط داده را در یک فاصله خاص (epsilon) از تابع برآورد شده داشته باشد. این فاصله به عنوان حاشیه (margin) تعیین می‌شود.



در این مثال، خط رگرسیون به گونه‌ای تعیین شده است که اکثر نقاط داده در یک فاصله مشخص (حاشیه) از این خط قرار گیرند. این فاصله با پارامتر epsilon کنترل می‌شود. نقاطی که خارج از این حاشیه قرار می‌گیرند، به عنوان نقاط خطا در نظر گرفته می‌شوند و مدل سعی می‌کند تعداد این نقاط را به حداقل برساند.