یادگیری ماشین

تمرین «یک»

دکتر سامان هراتی‌زاده

دانشگاه تهران - دانشکده سامانه‌های هوشمند

نیم‌سال اول ۱۴۰۳-۱۴۰۲

دستیاران آموزشی

روشنک طالشی

امیرمحمد کویش‌پور

ساعت ۲۳:۵۹ | ۲۹ مهر ۱۴۰۲

۱. [۳۰٪ - `1R`] هدف از این تمرین پیاده‌سازی الگوریتم `OneRule` است.

در این الگوریتم، قاعده‌ای ساده توسط الگوریتم ایجاد می‌شود که بر اساس آن، هر نمونه داده به یکی از دسته‌ها تعلق می‌گیرد. پس خروجی اصلی در این تکلیف استخراج یک قاعده تصمیم‌گیری است. در ادامه، به توضیح بخش‌های مختلف این سوال می‌پردازیم:

- **پیش‌پردازش داده**

قدم بعدی در حل سوال این است که داده‌ها را پیش‌پردازش کنید تا مشکلات موجود در آن‌ها شناسایی و رفع شوند. این مرحله شامل بررسی و پردازش داده‌های از دست رفته، مقادیر نامعتبر و همچنین داده‌های عددی هستند. برای پاکسازی داده‌های نامعلوم میتوانید آن‌ها را با مقادیر مناسب مانند مینیمم یا میانگین جایگزین کنید و در صورت لزوم آن سطر را حذف کنید. همچنین در صورت لزوم میتوانید با روش‌هایی مقادیر عددی را نیز به اسمی تبدیل کنید. راه حل شما برای کار با ستون‌هایی با محتوای عددی چیست؟

- **پیاده‌سازی الگوریتم `OneRule`**

در این بخش از شما انتظار داریم که یک در پایتون یک کلاس به نام `OneRule` تعریف کنید که دارای دو تابع اصلی `fit` و `predict` و دیگر توابع است.

تابع `fit`: در این بخش، قاعده‌ها بر اساس داده‌های آموزشی تولید می‌شوند. الگوریتم سعی می‌کند با تحلیل داده‌های آموزشی یک قاعده ساده تعیین کند که بر اساس آن، دسته‌بندی موثری بر روی داده‌ها انجام شود. در ورودی داده‌های اموزش وارد میشوند و خروجی قاعده تولید شده است. به ازای هر ستون `accuracy` محاسبه شده را نیز گزارش کنید.

Algorithm 1 Rule Extraction from Training Data

```
Input: Training data D
Output: Rule R
for i ∈ {1, ..., n} do                          ▷ number of attributes
    for v ∈ {1, ..., m} do                      ▷ number of values of attribute i
        c_i ← count(D_i)                        ▷ number of instances of class c in D_i
        c ← max(c_i)                            ▷ most frequent class in D_i
        R ← assign(R, c, i, v)                  ▷ assign class c to attribute i with value v
    end for
end for
e ← error(R, D)                                 ▷ error rate of rule R on data D
Return R                                        ▷ rule with smallest error rate
```

شکل ۱: الگوریتم 1R

تابع predict: در این تابع، داده‌های تست با استفاده از قاعده‌های تولید شده در تابع fit، دسته‌بندی می‌شوند. این تابع وظیفه پیش‌بینی دسته‌بندی داده‌های تست را بر عهده دارد. در ورودی، داده تست و قاعده را بدهید و در خروجی امتیازهای ذکر شده اعلام شود.

- **آشنایی با داده**

در این بخش، دادگان HR-Employee-Attrition.csv را تصادفی به دو بخش تقسیم کنید: ۸۰٪ برای آموزش و ۲۰٪ برای آزمایش. این تقسیم برای ارزیابی عملکرد الگوریتم One Rule استفاده می‌شود. داده‌های آموزش برای ساخت قاعده و داده‌های آزمایش برای ارزیابی عملکرد و دقت الگوریتم استفاده می‌شوند.

در پایان سوال توضیحاتی در مورد دادگان ارائه شده آورده شده است.

- **ارزیابی**

در ادامه ماتریس درهم‌ریختگی[1] را رسم کنید (در ادامه توضیحاتی در این باره آورده‌ایم) و نهایتا با استفاده از معیارهایی مانند precision ، recall ، F-score و ROC عملکرد الگوریتم OneRule بر روی داده‌های آزمایش بررسی شود. این معیارها به عنوان معیارهای ارزیابی استفاده می‌شوند تا دقت و صحت دسته‌بندی الگوریتم را نشان دهند. نتایج این ارزیابی، میزان عملکرد و قابلیت پیش‌بینی الگوریتم OneRule را نشان می‌دهد. مزایا و محدودیت‌های این الگوریتم چیست؟ آیا راهی برای بهبود محدودیت‌های آن سراغ دارید؟ در گزارش شرح دهید.

ماتریس درهم‌ریختگی یک ابزار اندازه‌گیری عملکرد است که در یادگیری ماشین و آمار برای ارزیابی دقت یک مدل طبقه‌بندی استفاده می‌شود. این جدولی است که پیش بینی‌های انجام شده توسط مدل را در برابر برچسب‌های کلاس واقعی داده‌ها خلاصه می‌کند.

جدول ۱: ساختار ماتریس درهم‌ریختگی

		Predicted Class	
		Positive	**Negative**
Actual Class	**Positive**	True Positive	False Negative
	Negative	False Positive	True Negative

[1] Confusion Matrix

در ماتریس درهم‌ریختگی، ردیف‌ها برچسب‌های کلاس واقعی را نشان می‌دهند، در حالی که ستون‌ها نشان دهنده برچسب‌های کلاس پیش بینی شده هستند. چهار عبارت موجود در ماتریس دارای تفاسیر زیر هستند:

- مثبت واقعی[1] : مدل کلاس مثبت را به درستی پیش بینی کرد.
- مثبت کاذب[2] : مدل به اشتباه کلاس مثبت را زمانی که کلاس واقعی منفی بود (خطای نوع I) پیش بینی کرد.
- منفی کاذب[3] : مدل به اشتباه کلاس منفی را زمانی که کلاس واقعی مثبت بود (خطای نوع II) پیش بینی کرد.
- منفی واقعی[4] : مدل کلاس منفی را به درستی پیش بینی کرد.

دقت، یادآوری و امتیاز F1 معمولاً معیارهایی هستند که از ماتریس درهم‌ریختگی برای ارزیابی عملکرد یک مدل طبقه‌بندی استفاده می‌شوند. آنها به صورت زیر محاسبه می‌شوند:

- **Precision:** دقت نسبت مثبت‌های واقعی به مجموع مثبت‌های واقعی و مثبت های کاذب است. دقت پیش‌بینی‌های مثبت مدل را اندازه‌گیری می‌کند. فرمول (۱) نحوه محاسبه این روش را نشان می‌دهد.

$$Recall = \frac{TP}{TP + FP} \qquad (1)$$

دقت بر نسبت پیش‌بینی‌های مثبتی که واقعاً درست هستند، تمرکز می‌کند و بینشی در مورد توانایی مدل برای اجتناب از مثبت‌های کاذب ارائه می‌کند.

- **Recall:** Recall که به عنوان حساسیت یا نرخ مثبت واقعی نیز شناخته می‌شود، نسبت مثبت های واقعی به مجموع مثبت‌های واقعی و منفی های کاذب است. توانایی مدل در شناسایی صحیح موارد مثبت را می‌سنجد.

$$Recall = \frac{TP}{TP + FN} \qquad (2)$$

[1] True Positive
[2] True Negative
[3] False Negative
[4] True Negative

Recall بر نسبت نمونه‌های مثبتی که به درستی شناسایی شده‌اند، تأکید می‌کند، که نشان‌دهنده توانایی مدل برای اجتناب از منفی‌های کاذب است.

- **امتیاز F1:** امتیاز F1 میانگین هارمونیک دقت و یادآوری است. اندازه گیری متعادلی را ارائه می‌دهد که هم precision و هم Recall را به طور همزمان در نظر می‌گیرد.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

امتیاز F1 ، precision و Recall را در یک متریک واحد ترکیب می‌کند، که زمانی مفید است که توزیع کلاس نامتعادل باشد.

- **منحنی مشخصه عملیاتی گیرنده (ROC) :** یک نمایش گرافیکی است که عملکرد یک مدل طبقه بندی باینری را در آستانه‌های طبقه بندی مختلف نشان می‌دهد. نرخ مثبت واقعی[1] را در برابر نرخ مثبت کاذب[2] در تنظیمات آستانه‌های مختلف ترسیم می کند.

فرمول FPR و TPR به صورت زیر است:

$$FPR = \frac{FP}{FP + TN} \quad (4)$$

$$TPR = \frac{TP}{TP + FN} \quad (5)$$

منحنی ROC با نشان دادن اینکه چگونه عملکرد مدل با مقادیر آستانه متفاوت تغییر می کند، به ارزیابی مبادله بین مثبت واقعی و مثبت کاذب کمک می‌کند. ناحیه زیر منحنی ROC (AUC-ROC) اغلب به عنوان یک متریک خلاصه برای تعیین کمیت عملکرد کلی یک طبقه‌بندی کننده استفاده می‌شود. AUC-ROC بالاتر نشان دهنده توانایی تشخیص بهتر مدل است.

ستون‌های داده‌ی کارکنان منابع انسانی به شرح زیر است

- **Age**: این ستون نشان دهنده سن کارکنان است.

[1] True Positive Rate
[2] False Positive Rate

- **Attrition**: این ستون نشان می دهد که آیا کارمندی شرکت را ترک کرده است یا هنوز مشغول به کار است. این ستون حاوی مقادیر باینری مانند «بله» یا «خیر» برای نشان دادن وضعیت فرسایشی است. در این داده این ستون را به عنوان کلاس مدنظر قرار داده‌ایم.
- **BusinessTravel**: این ستون تعداد یا نوع سفرهای کاری انجام شده توسط کارمندان را نشان می دهد. شامل دسته‌هایی مانند «غیر مسافرتی»، «به ندرت سفر کنید» یا «مکرر سفر کنید».
- **DailyRate**: این ستون به نرخ یا حقوق روزانه کارکنان اشاره دارد.
- Department: این ستون بخش یا ناحیه عملکردی را مشخص می کند که کارکنان در آن کار می کنند. شامل بخش‌هایی مانند «فروش»، «منابع انسانی» یا «تحقیق و توسعه» باشد.
- **DistanceFromHome**: این ستون فاصله بین خانه کارمند و محل کارشان را بر حسب مایل نشان می دهد.
- **Education**: این ستون نشان دهنده بالاترین سطح تحصیلات است که توسط کارکنان کسب شده است. شامل مقولاتی مانند "دبیرستان"، "مدرک لیسانس" یا "مدرک کارشناسی ارشد" است.
- **EducationField**: این ستون رشته تحصیلی یا تخصص تحصیلی کارکنان را مشخص می کند. این شامل حوزه‌هایی مانند "بازاریابی"، "مهندسی" یا "مالی" است.
- **Environment Satisfaction**: این ستون میزان رضایت کارکنان را از محیط کاری خود می‌سنجد. که در مقیاسی مانند ۱ تا ۵ نمره گذاری شده‌است که نشان دهنده سطوح متفاوتی از رضایت است.
- **Gender**: این ستون نشان دهنده جنسیت کارکنان است.
- **JobLevel**: این ستون نشان دهنده سطح سلسله مراتبی یا رتبه پست های کارکنان در داخل شرکت است. می توان آن را به صورت عددی نشان داد، مانند ۱، ۲، ۳ و ... .
- **MaritalStatus**: این ستون وضعیت تأهل کارمندان را نشان می دهد، مانند «مجرد»، «متاهل» یا «طلاق».
- **MonthlyRate**: این ستون به نرخ ماهانه یا حقوق کارکنان اشاره دارد.
- **NumCompaniesWorked**: این ستون نشان دهنده تعداد شرکت هایی است که کارمندان قبل از پیوستن به شرکت فعلی در آنها کار کرده اند.
- **OverTime**: این ستون نشان می دهد که آیا کارمندان اضافه کار می کنند یا خیر.
- **PerformanceRating**: این ستون نشان دهنده رتبه بندی یا ارزیابی عملکرد کارکنان است.
- **WorkLife Balance**: این ستون تعادل درک شده کار و زندگی کارکنان را اندازه گیری می‌کند. آن را در مقیاس ۱ تا ۴ نمره داده‌اند که سطوح مختلف تعادل کار و زندگی را نشان می‌دهد.
- **YearsAtCompany**: این ستون نشان دهنده تعداد سال‌هایی است که کارکنان در شرکت فعلی کار کرده اند.

۲. [PRISM - ۳۰٪] هدف از این تمرین پیاده‌سازی الگوریتم PRISM است.

الگوریتم PRISM یک الگوریتم است که در داده‌کاوی و تحلیل داده مورد استفاده قرار می‌گیرد. PRISM برای کشف قوانین قابل توجه و سودآور بر اساس استخراج زیرگروه‌های جالب از داده‌ها استفاده می‌کند.

الگوریتم PRISM با استفاده از معیارهایی مانند پوشش[1] و اطمینان[2] به دنبال زیرگروه‌هایی در داده‌ها می‌گردد که دارای خواص موردنظر هستند. این الگوریتم با بهره‌گیری از روش‌های تجزیه و تحلیل آماری و ارزیابی جوانب مختلف قوانین، تلاش می‌کند قوانینی را با پوشش بالا و اطمینان قابل قبول پیدا کند.

- **پیش‌پردازش داده**

قبل از اعمال الگوریتم PRISM بر روی داده‌های قلب و عروق cardio.csv، نیاز است داده‌ها را در بخش پیش‌پردازش تمیز کنید. این عمل شامل مراحل زیر می‌شود:

- **حذف داده‌های تکراری**: در صورت وجود داده‌های تکراری، آن‌ها را حذف کنید تا داده‌ها به صورت منحصر به فرد باقی بمانند.
- **پر کردن مقادیر از دست رفته**: در صورتی که برخی از ستون‌ها دارای مقادیر ازبین‌رفته[3] باشند، می‌توانید این مقادیر را با روش‌هایی مانند میانگین، مد، یا مقدار پرتکرار در آن ستون جایگزین کنید.
- تبدیل داده‌های عددی به اسمی: در صورتی که داده‌ها شامل متغیرهای عددی باشند، نیاز است آن‌ها را به صورت اسمی[4] تبدیل کنید.

در پایان سوال توضیحاتی در مورد ساختار دادگان ارائه شده آورده شده است.

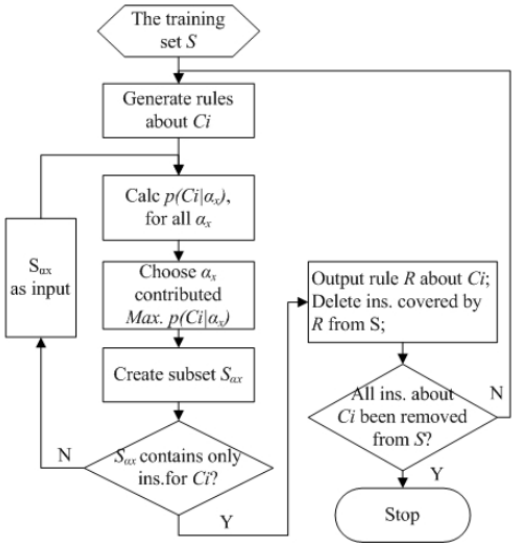
- **پیاده‌سازی الگوریتم `Prism`**

در این بخش مانند سوال قبل، خروجی قابل انتظار کلاس Prism خواهد بود که دارای حداقل دو متد fit و predict و دیگر متدهای مورد نیاز است. همانطور که پیش‌تر توضیح دادیم در بخش پیش‌پردازش، داده را به دو بخش آموزش و تست به نسبت ۸۰ به ۲۰ به صورت رندوم تقسیم کنید.

[1] Coverage
[2] Confidence
[3] Missing Value
[4] Categorical

متد `fit`: در این بخش، قاعده‌ها بر اساس داده‌های آموزشی تولید می‌شوند. الگوریتم سعی می‌کند با تحلیل داده‌های آموزشی قواعدی را تعیین کند که بر اساس آن، دسته‌بندی موثری بر روی داده‌ها انجام شود. ورودی الگوریتم ماتریس مشخصه `X_train` و بردار هدف `y_train` و خروجی الگوریتم قواعد تولید شده خواهد بود.

متد `predict`: در این متد، داده‌های تست با استفاده از قاعده‌های تولید شده در متد `fit`، دسته‌بندی می‌شوند. این تابع وظیفه پیش‌بینی دسته‌بندی داده‌های تست را بر عهده دارد. در ورودی، داده تست و قاعده را بدهید و در خروجی امتیازهای خواسته شده اعلام شود.



شکل ۲: فلوچارت مربوط به الگوریتم prism

- **ارزیابی**

در ادامه ماتریس درهم‌ریختگی را رسم کنید و نهایتا با استفاده از معیارهای `precision`، `recall`، `F-score` و ROC عملکرد الگوریتم `Prism` بر روی داده‌های آزمایش بررسی شود. الگوریتم `OneRule` و `Prism` را مقایسه کنید در چه شرایطی هر یک از این دو الگوریتم را انتخاب می‌کنید، در گزارش شرح دهید.

ستون‌های دادگان قلبی به شرح زیر است

- **Age**: این ستون نشان دهنده سن افراد در مجموعه داده است.
- **gender**: این ستون جنسیت افراد را نشان می دهد. دارای مقادیری مانند "مذکر" و "مونث" باشد که نشان دهنده جنسیت شرکت کنندگان است.

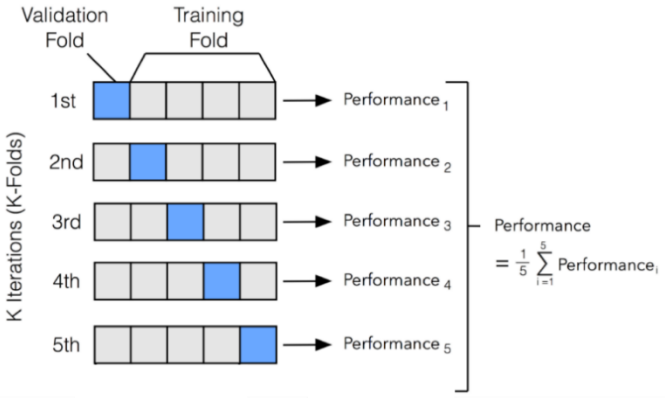
- **height**: این ستون نشان دهنده قد افراد است که معمولاً در سانتی متر اندازه گیری می شود
- **weight**: این ستون نشان دهنده وزن افراد است که معمولاً بر حسب کیلوگرم اندازه گیری می شود.
- **ap_hi**: این ستون نشان دهنده فشار خون سیستولیک افراد است. فشار خون سیستولیک مقدار بالاتری است که در طول اندازه گیری فشار خون اندازه گیری می شود و فشار در شریان ها را در هنگام انقباض قلب منعکس می کند.
- **ap_lo**: این ستون نشان دهنده فشار خون دیاستولیک افراد است. فشار خون دیاستولیک مقدار پایین‌تری است که در حین اندازه‌گیری فشار خون اندازه‌گیری می‌شود و فشار در شریان‌ها را زمانی که قلب بین ضربان‌ها استراحت می‌کند، منعکس می‌کند.
- **cholesterol**: این ستون میزان کلسترول افراد را نشان می دهد. دارای دسته‌هایی مانند «طبیعی»، «بیش از حد طبیعی» یا «بالا» باشد که سطوح مختلف کلسترول را نشان می‌دهد.
- **gluc**: این ستون نشان دهنده سطح گلوکز افراد است. این می تواند دارای دسته هایی مانند "طبیعی"، "بالاتر از نرمال" یا "بالا" باشد که سطوح مختلف گلوکز در خون را نشان می¬دهد.
- **smoke**: این ستون نشان می دهد که افراد سیگاری یا غیرسیگاری هستند.
- **alco**: این ستون نشان می دهد که آیا افراد الکل مصرف می کنند یا خیر.
- فعال: این ستون نشان می دهد که آیا افراد از نظر بدنی فعال هستند یا خیر.
- **cardio**: این ستون نشان دهنده وجود یا عدم وجود بیماری قلبی عروقی در افراد است. مقادیر دودویی مانند "۱" و "۰" دارد که "۱" نشان دهنده وجود بیماری قلبی عروقی و "۰" نشان دهنده عدم وجود آن است. که این ستون همان کلاس مد نظر ماست.

---

۳. [۱۰٪ - اعتبارسنجی متقابل] هدف از این بخش استفاده از k-بخش اعتبارسنجی متقابل در یادگیری ماشین، ارزیابی توانایی یک مدل پیش بینی در عملکرد و قابلیت تعمیم است. **این سوال را به کمک ۵۰۰ نمونه تصادفی از مجموعه‌دادگان `cardio.csv` و `adult_income_cencus.csv` و کل مجموعه `stroke_production.csv` آزمایش کنید.**

در اعتبارسنجی متقابل K-fold، مجموعه داده موجود به K بخش مساوی تقسیم می‌شود. ایده اصلی این روش این است که مدل را K بار آموزش داده و ارزیابی کنیم. در هر بار آموزش، یکی از فولدها به عنوان مجموعه

اعتبارسنجی استفاده شده و K-1 فولد دیگر به عنوان مجموعه آموزش استفاده می‌شوند. این فرآیند به ما امکان می‌دهد که از کل مجموعه داده برای آموزش و اعتبارسنجی استفاده کنیم و برآورد قوی‌تری از عملکرد مدل بدست آوریم.



شکل ۳: روش k-fold اعتبارسنجی متقابل

مراحل انجام اعتبارسنجی متقابل K-fold عبارتند از:

۱. **تقسیم داده:** مجموعه داده به K بخش یا فولد مساوی تقسیم می‌شود.
۲. **آموزش مدل و ارزیابی:** K بار تکرار می‌شود. هر بار یکی از فولدها به عنوان مجموعه اعتبارسنجی استفاده می‌شود و K-1 فولد دیگر برای آموزش مدل به‌کار می‌روند. مدل بر روی مجموعه آموزش آموزش داده می‌شود و بر روی مجموعه اعتبارسنجی ارزیابی می‌شود.
۳. **معیارهای عملکرد:** در پایان هر بار تکرار، با استفاده از یک معیار ارزیابی، مانند دقت، صحت، بازخوانی یا خطا میانگین مربعات میانگین، عملکرد مدل سنجیده می‌شود.
۴. **تجمیع نتایج:** معیارهای عملکرد به دست آمده در هر بار تکرار معمولاً میانگین گیری می‌شوند تا تخمینی کلی از عملکرد مدل به دست آید. شما می‌توانید از اعتبارسنجی متقابل K-fold برای برآورد قوی‌تر عملکرد مدل استفاده کنید. این روش به ما کمک می‌کند تا میزان تعمیم‌پذیری مدل به داده‌های ناشناخته را ارزیابی کنیم و مشکلات بیش‌برازش یا کم‌برازش را شناسایی کنیم. همچنین، از تمام داده‌های موجود بهره‌برداری کنیم زیرا هر نمونه به عنوان داده آموزش و اعتبارسنجی استفاده می‌شود.

مقادیر معمول برای K عبارتند از ۵ و ۱۰، اما انتخاب مقدار K ممکن است بسته به حجم مجموعه داده و منابع محاسباتی موجود متفاوت باشد. مقادیر بزرگتر از K، برآورد دقیق‌تری از عملکرد مدل را فراهم می‌کنند، اما به

مقدار بیشتری منابع محاسباتی نیاز دارند. با استفاده از این روش یک کلاس تعریف کنید که دیتا و مقدار k را دریافت کند و محاسبات را انجام دهد. نتایج را برای همان معیارهای نام برده در سوالات قبل برای دو الگوریتم 1R و PRISM، **برای k با مقادیر ۳ و ۵ و ۷** گزارش کنید.

---

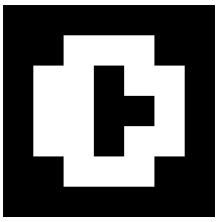
۴. [KNN - ۳۰٪] هدف از این تمرین پیاده‌سازی الگوریتم نزدیک‌ترین همسایه است.

در این بخش شما می‌بایست به طراحی و پیاده‌سازی طبقه‌بند k-نزدیک‌ترین همسایه[1] بپردازید، همچنین دادگان ارقام دست‌نویس دودویی[2] در اختیار شما قرار گرفته است. مجموعه داده ارائه شده شامل اعداد ۰ تا ۹ بوده که در قالب مجموعه‌ای از صفرها و یک‌ها در ابعاد ۲۸ در ۲۸ نمایش داده می‌شوند.

- **مجموعه دادگان**

هر نمونه در مجموعه دادگان دارای ارتفاعی با ۲۸ پیکسل و عرضی با ۲۸ پیکسل، که در مجموع ۷۸۴ پیکسل است. هر پیکسل می‌تواند دو مقدار صفر و یک را بگیرد، شکلی که پیکسل‌های دارای مقدار «۱» نشان می‌دهند، همان عددی است که توسط کاربر نوشته‌شده است و بدین صورت مجموعه دادگان آموزشی، دارای ۷۸۵ ستون است، که ستون برچسب[3] رقمی است که توسط کاربر ترسیم شده است.

```
0000000
0011100
0110110
0110010
0110110
0011100
0000000
```



شکل ۴: نمایش عدد صفر در مجموعه دادگان

- **پیاده‌سازی**
  - کلاس KNN که پیاده‌سازی می‌کنید، باید حداقل دارای متدهای زیر باشد.

- `1. __init__(self, k=3)`

طبقه‌بند KNN را با مقدار پیش‌فرض K (تعداد همسایگان) به صورت ۳ مقداردهی کنید.

[1] KNN Classifier
[2] Digit-recognizer
[3] Label

- **`2. fit(self, X_train, y_train)`**

این متد مدل `KNN` را بر ماتریس مشخصه `X_train` و بردار هدف `y_train` آموزش می‌دهد.

- **`3. predict(self, X_test)`**

این متد داده‌های آزمون `X_test` را می‌پذیرد و برچسب‌های پیش‌بینی شده را بر می‌گرداند.

هر گونه متد یا کلاس اضافی را که برای راه‌حل خود ضروری می‌دانید، اضافه کنید (هدف و استفاده از هر جزء اضافی را مستند کنید).

- **گزارش تفصیلی**
  - در قالب `PDF` رویکرد خود، تصمیمات طراحی، چالش‌های پیش‌روی و راه‌حل اجرا شده را گزارش کنید.
  - تاثیر مقادیر مختلف `k` بر عملکرد مدل را مورد بحث قرار دهید.
  - برای توضیح منطق و هدف از هر بخش در کد خود، کامنت‌های معناداری[1] ارائه دهید.

عملکرد شما بر اساس معیارهای زیر ارزیابی خواهد شد؛

- صحت و عملکرد طبقه‌بند `KNN`.
- شفافیت و سازماندهی کد.
- کیفیت و کامل بودن گزارش.
- نشان‌دادن درک تاثیر پارامترهای کلیدی، مانند `K` در مدل خود.

بخش امتیازی (+18%)

- دادگان ارائه شده را به تصاویر متناظر تبدیل کنید.
- پیاده‌سازی‌ای برای کلاس `KNN` ارائه کنید که هم از تصاویر و هم از دادگان متنی برای آموزش و پیش‌بینی استفاده کند.
- روش‌های موجود برای ادغام اطلاعات[2] از هر دو وجه[3] برای بهبود دقت را کاوش کنید.
- نه تنها جزئیات فنی، بلکه بینش‌های بکار رفته برای طراحی فرایندهای خود را گزارش کنید.
- در مورد چالش‌های موجود در ترکیب داده‌های چندوجهی و راه‌حل‌های نوآورانه خود بحث کنید.
- نتایج را با تجزیه و تحلیل کمی وکیفی ارائه کنید.

[1] Meaningful Comments
[2] Information Fusion
[3] Modality