

Amirkabir University of Technology
(Tehran Polytechnic)



Department of
Computer Engineering

Course : Machine Learning

Homework 3

Najmeh Mohammadbagheri

99131009

بخش تشریحی

سوال اول

در بیز ساده چون متغیرها مستقل در نظر گرفته می‌شوند و برای محاسبه‌ی احتمال توام متغیرها از ضرب احتمالات استفاده می‌شود، در صورتی که تعداد داده‌های آموزش به اندازه‌ی کافی زیاد نباشد و به ازای برخی از ویژگی‌ها به شرط یک ویژگی دیگر، نمونه‌ای وجود نداشته باشد یک صفر در دیگر احتمالات ضرب می‌شود و باعث می‌شود که نتیجه‌ی مطلوب حاصل نشود. به همین منظور از یک laplace smoothing استفاده می‌شود که در این هموارسازی به تمام مقادیر ممکن متغیرها یک مقدار ثابت اضافه می‌شود (این مقدار ثابت همان میزان شدت لاپلاس است). و سپس احتمالات محاسبه می‌شوند.

به طور مثال در یک مساله‌ی پردازش زبان اگر متغیرها حروف زبان باشند و در یک مجموعه داده کوچک نمونه‌ای وجود نداشته باشد که حرف غ را به شرط حرف گ داشته باشد. در این حالت این احتمال صفر خواهد بود. به همین منظور تمام حالت‌های ممکن را فرض می‌کنیم به تعداد مشخصی داریم و نمونه‌های موجود را با آن جمع می‌کنیم.

فرمول هموارسازی لاپلاس نیز به صورت زیر است:

$$p(X_i = x_{ik} | Y = y_j) = \frac{\text{count}(X_i = x_{ik}, Y = y_j) + l}{\text{count}(Y = y_j) + l \times k}$$

که l شدت هموارسازی و k تعداد مقادیر ممکن برای متغیرها است. در مثال ذکر شده $k = 32$ است برای زبان فارسی.

سوال دوم

این دو دسته‌بندی کننده از دو نوع مختلف دسته‌بندی کننده‌های generative و discriminative هستند. بیز ساده از نوع generative؛ یعنی مولد است. در الگوریتم‌های مولد دسته بند مقادیر $p(Y|X)$ را محاسبه می‌کند. برای محاسبه‌ی این احتمال $P(X|Y), P(Y)$ را بدست می‌آورد. با این محاسبات الگوریتم توزیع داده‌ها را فرامی‌گیرد و میتواند خودش داده‌ها را دوباره تولید کند. همانطور که میدانیم، الگوریتم بیز ساده این احتمال‌ها را بدست می‌آورد.

رگرسیون لاجستیک از نوع discriminative ؛ یعنی تمایزدهنده است. به این معنا که برای دسته‌بندی تنها یک مرز تصمیم را می‌آموزد و بر اساس آن دسته‌بندی را انجام می‌دهد.

در بیز ساده ابتدا احتمال $p(y|x)$ به شکل زیر بازنویسی می‌شود. سپس مقادیر $p(X|Y)$ از داده‌های آموزشی محاسبه می‌شود.

$$\mathbf{X} = \langle X_1, X_2, \dots, X_n \rangle, \quad X_i \perp\!\!\!\perp X_j | Y \quad (s.t: i, j = 1, 2, \dots, n \text{ \& } i \neq j)$$

$$\begin{aligned} y_{MLE} &= \operatorname{argmax}_{y \in \{+, -\}} p(\mathbf{X}, Y = y) = \operatorname{argmax}_{y \in \{+, -\}} p(\mathbf{X}|Y = y)p(Y = y) \\ &= \operatorname{argmax}_{y \in \{+, -\}} p(X_1|Y = y)p(X_2|Y = y) \dots p(X_n|Y = y)p(Y = y) \\ &= \operatorname{argmax}_{y \in \{+, -\}} p(Y = y) \prod_{i=1}^n p(X_i = x_i|Y = y) \end{aligned}$$

رگرسیون لاجستیک برای دسته‌بندی مقدار $p(Y|X)$ را مستقیماً محاسبه می‌کند و بر اساس مقدار آن دسته‌بندی را انجام می‌دهد. در ادامه توابعی که احتمال از آن‌ها محاسبه می‌شود برای حالت دو کلاسه آورده شده است. رگرسیون لاجستیک از تابع سیگموئید برای محاسبه کلاس مثبت استفاده می‌کند و ورودی این تابع یک ترکیب خطی از ویژگی‌های مختلف داده‌هاست. پس در واقع لاجستیک رگرسیون یک دسته‌بندی‌کننده خطی است. وزن‌های این ترکیب خطی به کمک داده‌های آموزشی و روش‌های بهینه‌سازی بدست می‌آیند. (البته لاجستیک غیرخطی نیز وجود دارد که ترکیب غیر خطی ویژگی‌ها به عنوان ورودی تابع سیگموئید در نظر گرفته می‌شود.)

$$P(Y = 1|X) = \frac{1}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)}$$

$$P(Y = 0|X) = \frac{\exp(w_0 + \sum_{i=1}^n w_i X_i)}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)}$$

مقایسه عملکرد مدل‌های لاجستیک و بیز ساده‌ی گوسی:

بیز ساده بایاس زیادی نسبت به داده‌های آموزش دارد. رگرسیون لاجستیک واریانس زیادی دارد که می‌توان میزان آن را با تغییر در تابع بهینه‌سازی بهبود بخشید.

اگر تعداد داده‌های آموزش به سمت بی‌نهایت برود و بیز ساده مفروضاتی مانند برابر بودن پراکندگی داده‌ها در کلاس‌های مختلف داشته باشد، نتایج دو مدل همگرا است یعنی هر دو مدل یک نتیجه را می‌دهند. اما در غیر این صورت رگرسیون لاجستیک به دلیل اینکه پارامترهای خود را براساس تمام داده‌ها محاسبه می‌کند دقت بیش‌تری نسبت به بیز ساده دارد.

سوال سوم

برچسب داده‌ی تست B است. چون در راستای x_1 انحراف معیار B از A بیشتر است و احتمال متعلق بودن داده‌ی تست به کلاس B بیشتر مساوی از احتمال متعلق بودن آن به کلاس A است. از طرف دیگر در راستای x_2 نیز احتمال متعلق بودن داده‌ی تست به کلاس B بیشتر است. چون اگر فرض کنیم میانگین هر دو کلاس در یک نقطه است و داده‌ی تست روی میانگین قرار دارد، چون انحراف معیار A بیشتر از B است پس ارتفاع قله‌ی B بیشتر از A است و در نتیجه احتمال متعلق بودن داده‌ی تست به کلاس B بیشتر است. و دلیل آخر اینکه احتمال اولیه‌ی کلاس B نیز بیشتر از کلاس A است. به همین سه دلیل زمانی که احتمال‌ها در هم ضرب می‌شوند حاصل نهایی برای کلاس B بیشتر می‌شود و داده‌ی تست برچسب کلاس B را می‌گیرد.

سوال چهارم

$$p(B|D) = \frac{p(B,D)}{p(D=T)} = \frac{p(D|B)p(B)}{p(D=T)}$$

$$p(B) = p(B|A)p(A) + p(B|\bar{A})p(\bar{A})$$

$$P(B) = (0.37 \times 0.2) + (0.21 \times 0.8) = 0.242$$

$$p(C) = p(C|A)p(A) + p(C|\bar{A})p(\bar{A})$$

$$P(C) = (0.3 \times 0.2) + (0.25 \times 0.8) = 0.26$$

$$p(D|B) = p(D|C, B)p(C) + p(D|\bar{C}, B)p(\bar{C})$$

$$P(D|B) = (0.5 \times 0.26) + (0.15 \times 0.74) = 0.241$$

$$\begin{aligned} p(D = T) &= \sum_{A,B,C} p(D = T|A, B, C)p(A, B, C) \\ &= \sum_{A,B,C} p(D = T|B, C)p(B|A)p(C|A)p(A) \end{aligned}$$

$$P(D = T) =$$

$$\begin{aligned} &(0.2 \times 0.37 \times 0.3 \times 0.5) + \\ &(0.8 \times 0.21 \times 0.25 \times 0.5) + \\ &(0.2 \times 0.63 \times 0.3 \times 0.67) + \\ &(0.8 \times 0.79 \times 0.25 \times 0.67) + \\ &(0.2 \times 0.37 \times 0.7 \times 0.15) + \\ &(0.8 \times 0.21 \times 0.75 \times 0.15) + \\ &(0.8 \times 0.79 \times 0.75 \times 0.95) + \\ &(0.2 \times 0.63 \times 0.7 \times 0.95) \\ &= 0.724 \end{aligned}$$

$$\Rightarrow P(B = T | D = T) = 0.241 \times 0.242 \div 0.724 = 0.08$$

$$p(\bar{B}|D) = \frac{p(\bar{B}, D)}{p(D = T)} = \frac{p(D|\bar{B})p(\bar{B})}{p(D = T)}$$

$$p(D|\bar{B}) = p(D|C, \bar{B})p(C) + p(D|\bar{C}, \bar{B})p(\bar{C})$$

$$P(D|\bar{B}) = (0.67 \times 0.26) + (0.95 \times 0.74) = 0.877$$

$$p(\bar{B}) = p(\bar{B}|A)p(A) + p(\bar{B}|\bar{A})p(\bar{A})$$

$$P(\bar{B}) = (0.63 \times 0.2) + (0.79 \times 0.8) = 0.758$$

$$\Rightarrow P(B = F | D = T) = 0.877 \times 0.758 \div 0.724 = 0.91$$

سوال پنجم

باتوجه به مدل مساله راهها و معیارهای متفاوتی برای انتخاب نقطه‌ی تصمیم در الگوریتم لاجستیک وجود دارد. یکی از این راهها استفاده از نمودار ROC است. نحوه‌ی ساختن این نمودار به این شکل است که نقطه‌ی تصمیم را مکان‌های مختلفی قرار می‌دهیم و ماتریس درهم ریختگی را برای هر نقطه‌ی تصمیم رسم می‌کنیم. سپس با استفاده از این ماتریس‌ها نمودار ROC را که محورهای آن TPR, FPR هستند، رسم می‌کنیم. سپس از روی این نمودار نقطه‌ای که میزان TPR آن نسبت به FPR آن بهتر است را انتخاب می‌کنیم که این انتخاب می‌تواند با درنظر گرفتن هزینه یا جریمه برای FPR باشد و یک trade off انجام شود. همچنین می‌شود بجای استفاده از نمودار ROC از نمودار precision_recall استفاده کرد. این انتخاب کاملاً بستگی به نوع مساله دارد.

سوال ششم

اگر p احتمال موفقیت و $1-p$ احتمال شکست باشد، نسبت بخت به صورت زیر است:

$$\text{Odds}(\text{success}) = p/1-p$$

$$\text{Odds}(\text{failure}) = 1-p / p$$

نسب بخت در رگرسیون لاجستیک، تاثیر ثابت یک تخمین زنده‌ی X را بر روی احتمال وقوع یک رخداد، نشان می‌دهد.

در مدل‌های رگرسیون تلاش ما براین بود که یک تاثیر ثابتی که متغیر X بر خروجی Y می‌گذارد را پیدا کنیم. در رگرسیون لاجیستیک هدف تاثیر X بر دسته‌های متفاوت Y را پیدا می‌کنیم. چون رگرسیون لاجیستیک یک دسته‌بندی کننده است نمیتوان تاثیر ثابت پیدا کرد. به ازای مقادیر متفاوت X دسته‌های مختلف پیش‌بینی میشود.

برای اینکه این تاثیر متغیر در رگرسیون لاجستیک را به یک تاثیر ثابت تبدیل کنیم، راه حل این است که بجای استفاده از احتمالات از نسبت بخت استفاده کنیم.

سوال هفتم

$$\begin{aligned}
 p(\text{age} = \text{youth}|+) &= \frac{2}{9}, p(\text{age} = \text{youth}|-) = \frac{3}{5}, p(\text{age} = \text{middle}|+) = \frac{4}{9}, \\
 p(\text{age} = \text{middle}|-) &= 0, p(\text{age} = \text{senior}|+) = \frac{3}{9}, p(\text{age} = \text{senior}|-) = \frac{2}{5}, \\
 p(\text{income} = \text{high}|+) &= \frac{2}{9}, p(\text{income} = \text{high}|-) = \frac{2}{5}, \\
 p(\text{income} = \text{medium}|+) &= \frac{4}{9}, p(\text{income} = \text{medium}|-) = \frac{2}{5}, \\
 p(\text{income} = \text{low}|+) &= \frac{3}{9}, p(\text{income} = \text{low}|-) = \frac{1}{5}, p(\text{student} = \text{no}|+) = \frac{3}{9}, \\
 p(\text{student} = \text{no}|-) &= \frac{4}{5}, p(\text{student} = \text{yes}|+) = \frac{6}{9}, p(\text{student} = \text{yes}|-) = \frac{1}{5}, \\
 p(\text{credit} = \text{fair}|+) &= \frac{6}{9}, p(\text{credit} = \text{fair}|-) = \frac{2}{5}, p(\text{credit} = \text{excellent}|+) = \\
 \frac{3}{9}, p(\text{credit} = \text{excellent}|-) &= \frac{3}{5}, p(+) = \frac{9}{14}, p(-) = \frac{5}{14}
 \end{aligned}$$

$X_1 = (\text{age} = \text{youth}, \text{income} = \text{high}, \text{student} = \text{yes}, \text{credit} = \text{fair})$

$$P(X_1|+)p(+) = \frac{2}{9} * \frac{2}{9} * \frac{6}{9} * \frac{6}{9} * \frac{9}{14} = 0.014$$

$$P(X_1|-)p(-) = \frac{3}{5} * \frac{2}{5} * \frac{1}{5} * \frac{2}{5} * \frac{5}{14} = 0.006$$

$\Rightarrow X_1$ belongs to +

$X_2 = (\text{age} = \text{senior}, \text{income} = \text{low}, \text{student} = \text{no}, \text{credit} = \text{excellent})$

$$P(X_2|+)p(+) = \frac{3}{9} * \frac{3}{9} * \frac{3}{9} * \frac{3}{9} * \frac{9}{14} = 0.007$$

$$P(X_2|-)p(-) = \frac{2}{5} * \frac{1}{5} * \frac{4}{5} * \frac{3}{5} * \frac{5}{14} = 0.013$$

$\Rightarrow X_2$ belongs to -

$X_3 = (\text{age} = \text{middle}, \text{income} = \text{medium}, \text{student} = \text{no}, \text{credit} = \text{fair})$

در این حالت با استفاده از هموارسازی لاپلاس محاسبات را انجام میدهیم چون در حالت عادی حاصلضرب صفر می شود.

$$P(X_3|+)p(+) = \frac{5}{12} * \frac{5}{12} * \frac{4}{11} * \frac{7}{11} * \frac{9}{14} = 0.025$$

$$P(X_3|-)p(-) = \frac{1}{8} * \frac{3}{8} * \frac{5}{7} * \frac{3}{7} * \frac{5}{14} = 0.006$$

$\Rightarrow X_3$ belongs to +

سوال اول

پیش پردازش انجام شده : داده‌ی ۱۰۰ برچسب آن unac خورده بود. که به صورت دستی به unacc اصلاح شد. چون داده‌ها دارای ترتیب بودند قبل از جدا کردن داده‌های تست و آموزش یک شافلینگ انجام شد. ۷۰ درصد داده‌ها برای آموزش و ۳۰ درصد برای آزمون در نظر گرفته شده است.

(الف)

معیارهای خواسته شده برای داده‌های تست:

```
sensitivity :
0.8169556840077071
specificity :
0.938985228002569
False Positive Rate :
0.06101477199743096
False Negative Rate :
0.18304431599229287
confusion matrix :
      acc  unacc  good  vgood
acc      72    42    5     0
unacc    12   333    0     0
good     15     0    9     1
vgood    18     0    2    10
```

معیارهای خواسته شده برای داده‌های آموزش:


```
sensitivity :
0.8874172185430463
specificity :
0.9624724061810155
False Positive Rate :
0.037527593818984545
False Negative Rate :
0.11258278145695365
confusion matrix :
```

| | acc | unacc | good | vgood |
|-------|-----|-------|------|-------|
| acc | 215 | 52 | 5 | 0 |
| unacc | 35 | 813 | 1 | 0 |
| good | 26 | 0 | 10 | 4 |
| vgood | 13 | 0 | 0 | 34 |

(ب)

معیارهای خواسته شده برای داده های تست:

```
sensitivity :
0.8689788053949904
specificity :
0.9563262684649968
False Positive Rate :
0.04367373153500321
False Negative Rate :
0.13102119460500963
confusion matrix :
```

| | acc | unacc | good | vgood |
|-------|-----|-------|------|-------|
| acc | 85 | 30 | 3 | 0 |
| unacc | 2 | 352 | 2 | 0 |
| good | 14 | 0 | 6 | 0 |
| vgood | 16 | 0 | 1 | 8 |

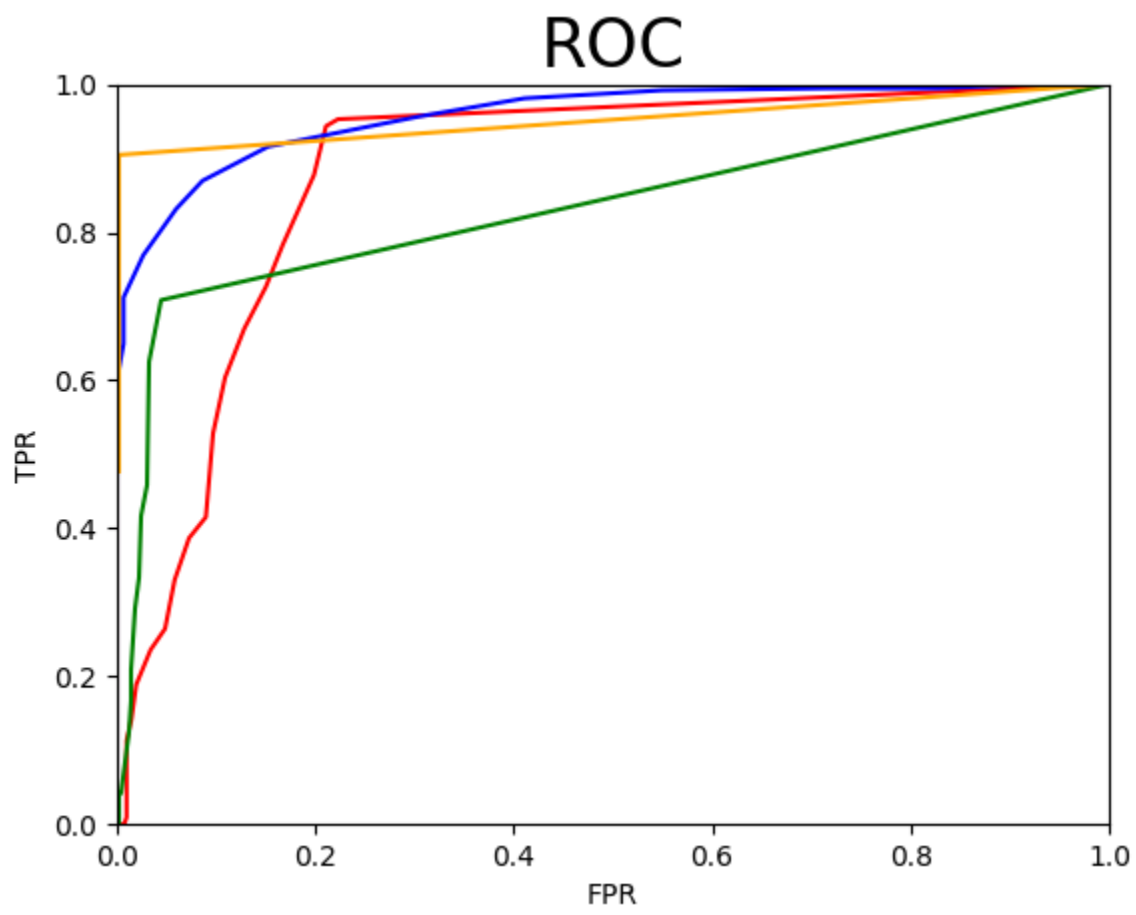
معیارهای خواسته شده برای داده های آموزش:

```

sensitivity :
0.8807947019867549
specificity :
0.9602649006622517
False Positive Rate :
0.039735099337748346
False Negative Rate :
0.11920529801324503
confusion matrix :
      acc  unacc  good  vgood
acc    203    63    5    0
unacc   32   821    2    0
good    27    0   16    0
vgood   15    0    0   24

```

(c)



Red : class acc

Blue : class unacc

Green : class good

Orange : class vgood

با روش one_vs_all برای هر کلاس نمودار رسم شده است. نمودار به ازای حد آستانه‌های ۱.۵ تا ۱۰ رسم شده.

در این حالت چون برخی داده‌ها احتمال صفر داشتند بدلیل هموار نشدن نمودارهای زرد و سبز که تعداد کمی نیز بودند دچار اشتباه در تصمیم گیری شده اند. اگر همواری سازی را انجام میدادیم و سپس رسم میکردیم نتایج بهتر می‌شد.

سوال دوم

(الف)

نحوه‌ی محاسبه‌ی خطا :

```
error = 0
predicted = np.loadtxt('train_labels.txt')
for i in range(predicted.shape[0]):
    if predicted[i] != labels[i]:
        error += 1
print(error/len(labels))
```

خطای آموزش : ۰.۷٪ ، خطای تست : ۸.۲٪

ماتریس درهم ریختگی :

برای محاسبه‌ی ماتریس درهم ریختگی هر کلاس نسبت به دیگر کلاس‌ها از کتابخانه استفاده شده است.

کد این قسمت:

```
multilabel_confusion_matrix(true, predict)
```

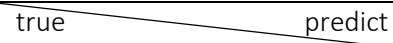

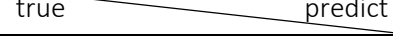
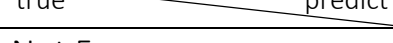
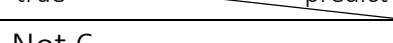
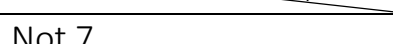
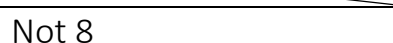

ماتریس‌های درهم ریختگی برای ۱۰ کلاس داده‌های آموزش :

| | | |
|----------------|-------|---|
| true \ predict | Not 0 | 0 |
|----------------|-------|---|

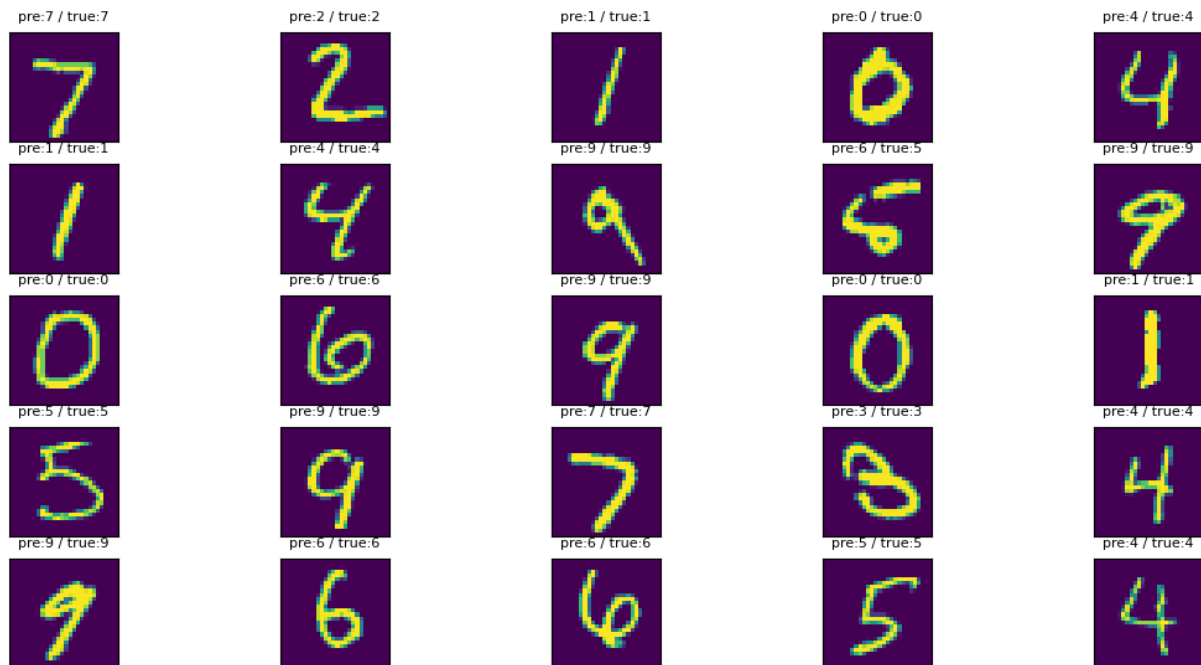
| | | |
|----------------|-------|------|
| Not 0 | 53855 | 222 |
| 0 | 127 | 5796 |
| true \ predict | Not 1 | 1 |
| Not 1 | 52997 | 261 |
| 1 | 152 | 6590 |
| true \ predict | Not 2 | 2 |
| Not 2 | 53597 | 445 |
| 2 | 534 | 5424 |
| true \ predict | Not 3 | 3 |
| Not 3 | 53318 | 551 |
| 3 | 623 | 5508 |
| true \ predict | Not 4 | 4 |
| Not 4 | 53770 | 388 |
| 4 | 373 | 5469 |
| true \ predict | Not 5 | 5 |
| Not 5 | 54097 | 482 |
| 5 | 717 | 4704 |
| true \ predict | Not 6 | 6 |
| Not 6 | 53788 | 294 |
| 6 | 213 | 5705 |
| true \ predict | Not 7 | 7 |
| Not 7 | 53390 | 345 |
| 7 | 390 | 5875 |
| true \ predict | Not 8 | 8 |
| Not 8 | 53339 | 810 |
| 8 | 698 | 5153 |
| true \ predict | Not 9 | 9 |
| Not 9 | 53418 | 633 |
| 9 | 604 | 5345 |

ماتریس درهم ریختگی برای ۱۰ کلاس داده‌های تست:

| | | |
|----------------|-------|-----|
| true \ predict | Not 0 | 0 |
| Not 0 | 8969 | 51 |
| 0 | 23 | 957 |
| true \ predict | Not 1 | 1 |

| | | |
|--|-------|------|
| Not 1 | 8820 | 42 |
| 1 | 19 | 1116 |
| true  predict | Not 2 | 2 |
| Not 2 | 8901 | 67 |
| 2 | 127 | 905 |
| true  predict | Not 3 | 3 |
| Not 3 | 8889 | 101 |
| 3 | 95 | 915 |
| true  predict | Not 4 | 4 |
| Not 4 | 8945 | 73 |
| 4 | 72 | 910 |
| true  predict | Not 5 | 5 |
| Not 5 | 9028 | 80 |
| 5 | 130 | 762 |
| true  predict | Not 6 | 6 |
| Not 6 | 8981 | 61 |
| 6 | 49 | 909 |
| true  predict | Not 7 | 7 |
| Not 7 | 8901 | 71 |
| 7 | 83 | 945 |
| true  predict | Not 8 | 8 |
| Not 8 | 8857 | 169 |
| 8 | 120 | 854 |
| true  predict | Not 9 | 9 |
| Not 9 | 8887 | 104 |
| 9 | 104 | 905 |

(ب)



(ج)

در وهله‌ی اول سرعت این روش نسبت به k نزدیک‌ترین همسایه بسیار بهتر است. در k نزدیک‌ترین همسایه به ازای هر داده باید فاصله‌ی 28×28 ویژگی را با داده‌ی تست محاسبه کرد. ۶۰۰۰۰ داده‌ی آموزشی داریم که باید هر داده‌ی تست را با این ۶۰ هزارتا مقایسه کرد. سرعت وحشتناک پایین است.

در وهله‌ی دوم در الگوریتم knn انتخاب هاپیر پارامتر k خود یک چالش اساسی است.

k نزدیک‌ترین همسایه زمانی کاربرد دارد که تعداد ویژگی‌ها کم و تعداد داده‌های آموزش نیز کم باشد.

الگوریتم لاجیستیک زمانی عملکرد بهتری دارد که داده‌های آموزش زیاد و کافی ای داشته باشد تا بتواند پارامترهایش؛ یعنی وزن‌هایش را به خوبی یاد بگیرد.

من مراحل قبل را با روش KNN امتحان نکردم که مقادیر دقیق داشته باشم برای ادعا ولی فکر میکنم که عملکرد knn نیز مانند لاجیستیک خوب است برای این مساله.

