دانشگاه
خواجه نصیرالدین طوسی
K. N. Toosi University
of Technology

**دانشکده مهندسی برق و کامپیوتر – گرایش مهندسی کنترل**

# امتحان میانترم درس یادگیری ماشین

دانشجو:

**محمد قاسم امامی مقدم**

شماره دانشجویی:

**40202314**

استاد:

**دکتر علیاری**

لینک گوگل کولب **و** لینک گیت هاب

بهار 1403


بسم الله الرحمن الرحیم

# فهرست مطالب

# سوال ۱)

**(1–1**

روش زیادی برای طبقه بندی دو کلاسه وجود دارد که بیز یکی از بهترین متد های موجود میباشد. در طبقه بندی به روش بیز به طور مستقیم احتمال تعاق هر نمونه به کلاس های مختلف را بیان میکند که همین امر برای تصمیمی گیری کار ما را راحت تر میکند.

از طرفی نیز روش بیز در مسایل دو کلاسه عملکرد خوبی دارد و نیز توانایی کار کردن با انواع مختاف توزیع را نیز دارد.به طور کلی به دلیل عملکرد مبتنی بر احتمال وجود داده در هر کلاس تصمیم گیر یرا انجام میدهد. از طرفی نسبت به متد های رگرسیون لجستیک یا درخت تصمیم عملکرد بهتری دارد.

ولی این متد به طور قطع بهترین روس برای طبقه بندی دو کلاسه نمی باشد زیرا هیچ روشی وجود ندارد که به طور قطع در تمامی مسایل دو کلاسه بهترین باشد و نیز در برخی موارد متد هایی نظیر svmیا شبکه های عصبی عملکرد بهتری دارند.

پس به طور قطع این گذاره درست نمیباشد و تا حدودی صحت دارد.

**(1–2**

در واقع روش بیز برای تخمین پارامتر ها اطلاعات قبلی را درمورد توزیع پارامتر ها در نظر میگیرد در واقع به طور کای همین در نظر گرفتن اطلات قبلی تا حد خوبی به باعث جلوگیری از بیش حد پیچیده شدن مدل و OVERFIT میشود .

از انجایی که در صورت سوال از فعل میتواند استفاده شده این گذاره درست می باشد .

**(1–3**

در واقع GAIN INFORMATION بر اساس انتروپی و میزان قطعیت عمل میکند.

حال اگر میزان حالات یک ویژگی زیاد باشد میزان GAIN INFORMATIONمربوط به ان ویژگی به طور مصنوعی بالا می باشد حتی در صورتی که ان ویژگی برای طبقه بندی مناسب نباشد. پس این امر ممکن است که منجر به انتخاب ویژگی های نا درست و ناکارامدی درخت بشوند. درواقع این امر موجب OVERFITو افزایش هزینه محاسبات میشود زیرا باسد تمامی حالات را بررسی کند. البته در حالاتی که این حالات مرتبط باشند مبتواند عملکرد درخت را بهبود ببخشد.

پس این گذاره تا حد خوبی درست است.

**(1–4**

یک شبکه عصبی چند لایه با توابع فعال ساز خطی ئر لایه های پنهان، همانند یک شبکه عصبی تک لایه عمل میکند زیرا نوابع فعال ساز خطی هیچ غیر خطی گری را به سیتم وارد نمی کنند. پس در واقع خروجی ترکیب خطی از ورودی هاست که هیچ تفاوتی با خروجی یک شبکه عصبی تکلایه ندارد. پس این گذاره قطعا درست است.

# سوال ۲)

## 2-1)

ابتدا با توجه به داده نمونه داده شده که به ضورت :

$$Feature = [A, B, C, D, E, F] \rightarrow X = [1,1,0,1,0,1] \text{ و}$$

$$A = 1, B = 1, C = 0\ , D = 1\ , E = 0, F = 1$$

ابتدا در نود اول به بررس مقدار Aمیپردازیم از انجایی که مقدار این ویژگی برای داده ی داده شده برابر با یک نیباشد به سمت شاخه سمت راست میرویم . سپس در این نود به بررس یمقدار ویژگی Dمیپردازیم که مقدار ان برابر با 1 میباشد و مجددا یا شاخه سمت راست میرویم . حال به پرسپترون تصمیم گیری رسیدیم معادله مربوط به این پرسپترون جهت تصمیم گیری برابر با :

$$out = sign[(1 * B\ ) + (0 * C) + (E * -1) + (F * 1) + 1(this\ is\ for\ bias)\ ]$$

با جاگذاری مقادیر رسیده به این پرسپترون داریم:

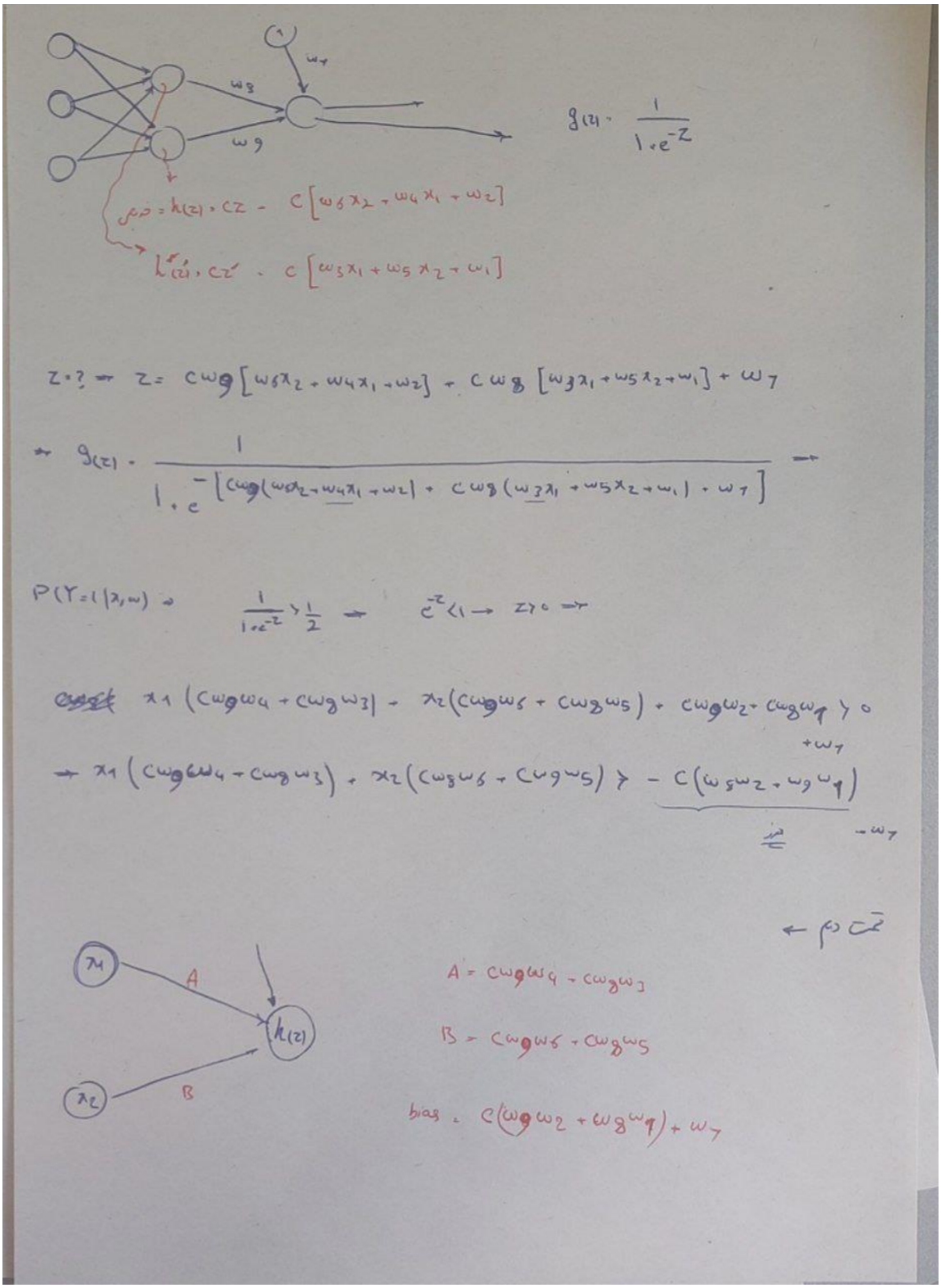
$$out = sign(1 + 0 + 0 + 1 + 1) = sign(3) = 1$$

## 2-2)

قسمت اول این گذاره نادرست است لزومی ندارد که مرزهای درخت تصمیم خطی باشند.

برای قسمت دوم با کاهش عمق درخت در برگ ها تصمیم بر اساس اکثریت نمونه ها صورت میگیرد و ممکن است با توجه به زیاد بودن ویژگی ها و حالات برخی حالات و ویژگی ها مورد بررسی و طقه بندی قرار نمیگیرند در واقع برخی ویژگی ها استفاده نشده می میانند. در درخت پرسپترون این اتفاق رخ نمیدهد زیرا در برگ های نهایی میتوان ویژگی های دیده نشده را نیز برای پیش بینی کلاس مورد بررسی قرار داد.

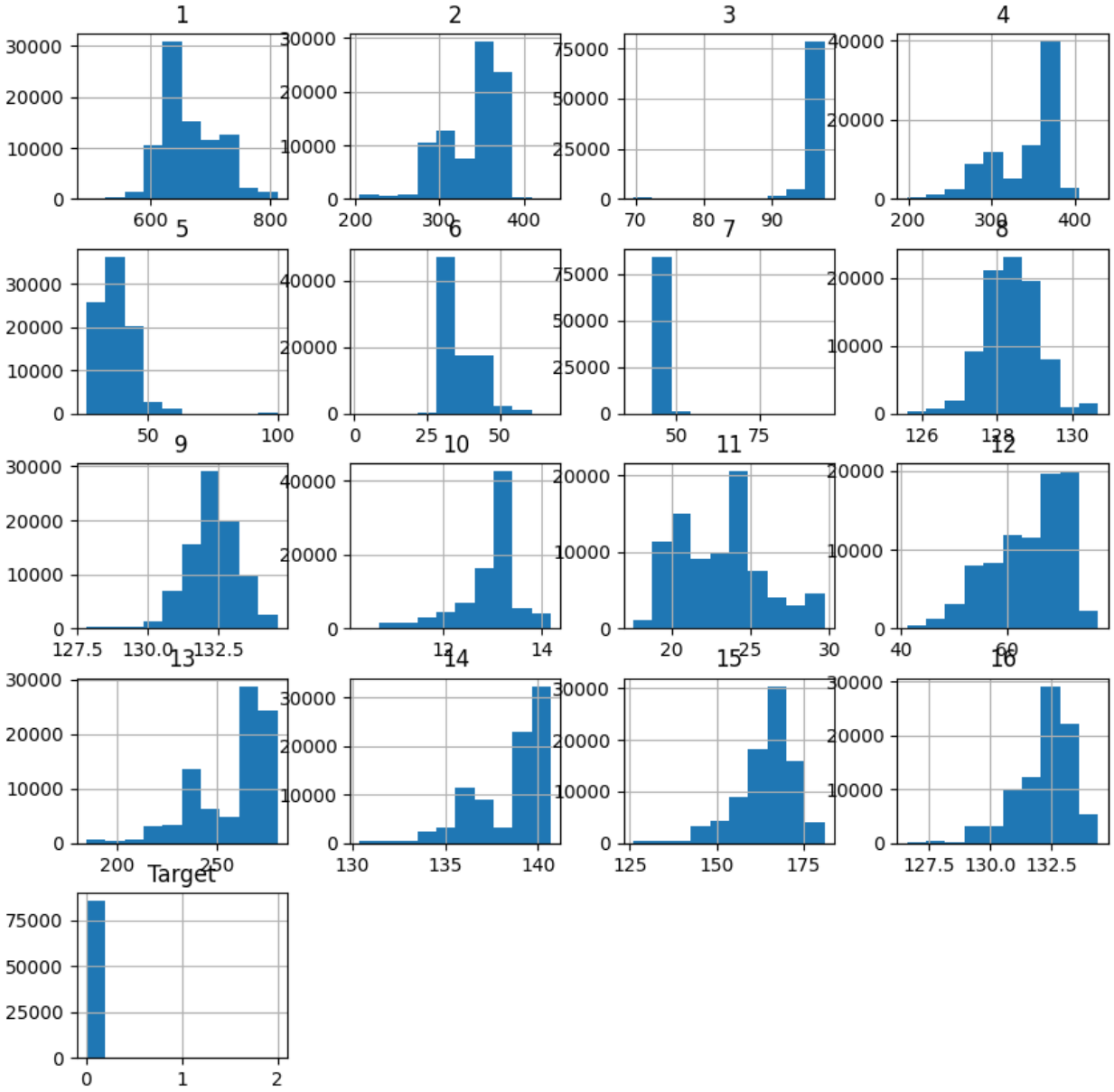
# سوال ۳)


w7
w8
w9
g(z) = 1 / (1 + e^-z)
h(z) = cz = c[w8x2 + w4x1 + w2]
h'(z') = cz' = c[w3x1 + w5x2 + w1]
z = ? → z = cw9[w8x2 + w4x1 + w2] + cw9[w3x1 + w5x2 + w1] + w7
g(z) = 1 / (1 + e^-[cw9(w8x2 + w4x1 + w2) + cw9(w3x1 + w5x2 + w1) + w7])
P(Y=1|x,w) → 1/(1+e^-z) > 1/2 → e^-z < 1 → z > 0
x1(cw9w4 + cw9w3) + x2(cw9w8 + cw9w5) + cw9w2 + cw9w1 + w7 > 0
x1(cw9w4 + cw9w3) + x2(cw9w8 + cw9w5) > -c(w8w2 + w9w1) - w7
x1
x2
A
B
h(z)
A = cw9w4 + cw9w3
B = cw9w8 + cw9w5
bias = c(w9w2 + w9w1) + w7

# سوال ۴)

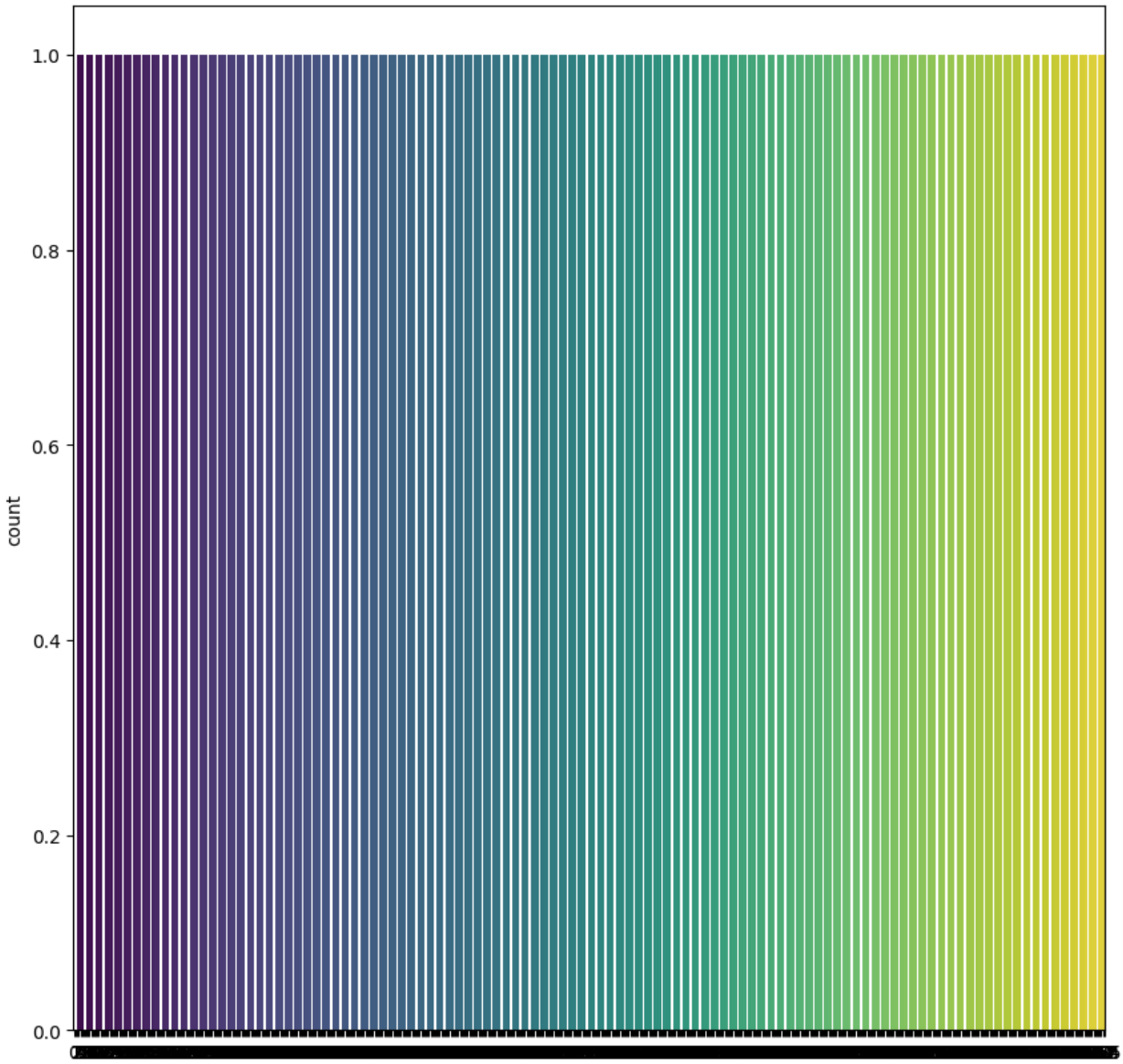
در این قسمت ابتدا داده ها را فراخوانی میکنیم .سپس در ابتدای مرحلهخ ماتریس هم بستگی را رسم میکنیم تا متوجه شویم که کدام ویژگی با ستون هدف که خودمان به ان اضافه کردیم بیشترین وابستگی را دارد.

در ابتدا نمودار تمامی ستون ها را رسم میکنیم تا شکا توزیع و پرلکندگی انها را ببینیم



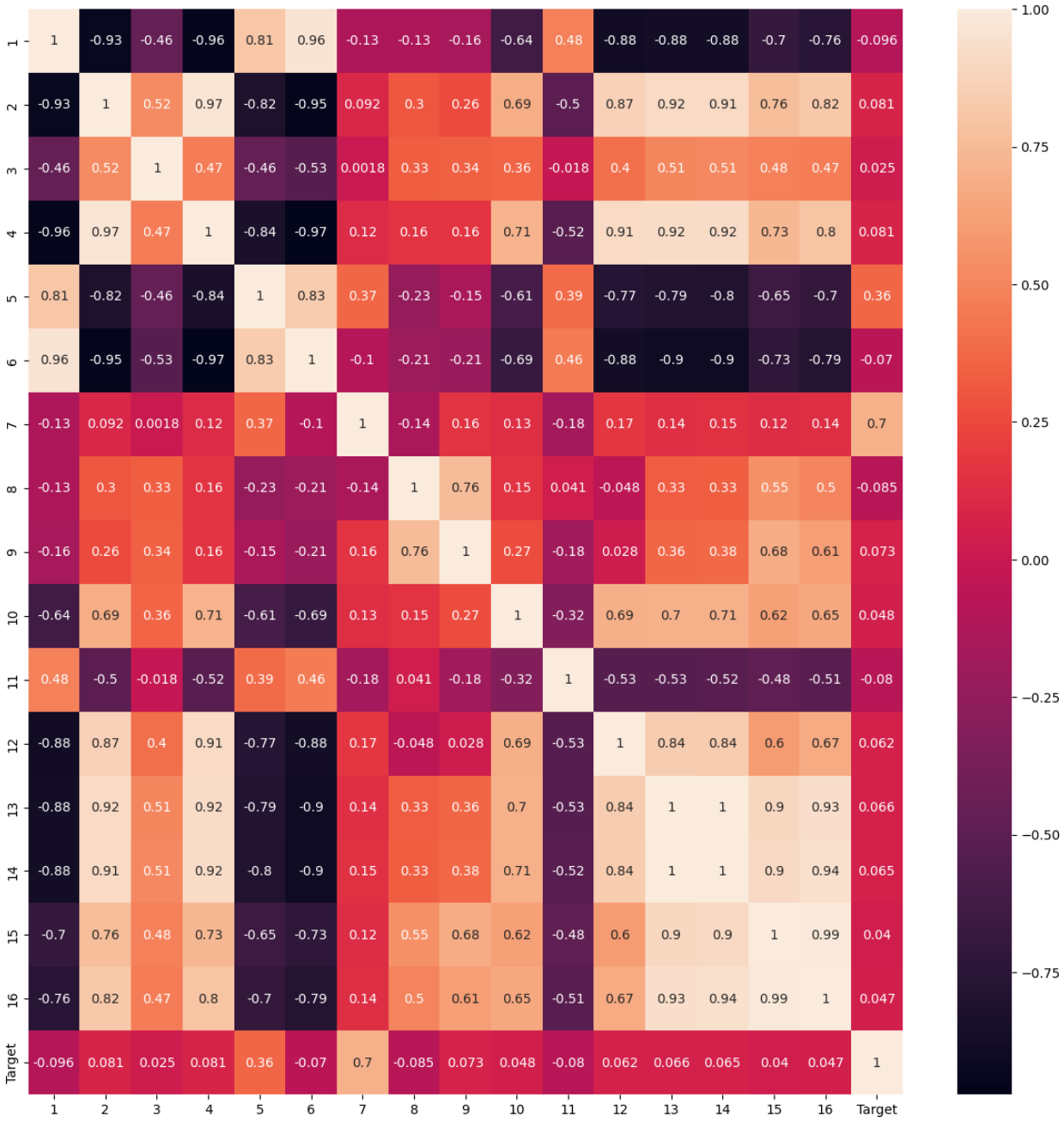

در ابتدای امر دیده میشود که تعداد داده هایی که نرمال هستند بسیار بیشتر از داده های خطا است که این به معنای این است که داده

ها بالانس نیستند و می بایست under samplingانجام داد .




ولی در این قسمت بدون در نظر گرفتن این متد ادامه مدهیم تا نتیجه را بررسی کنیم.

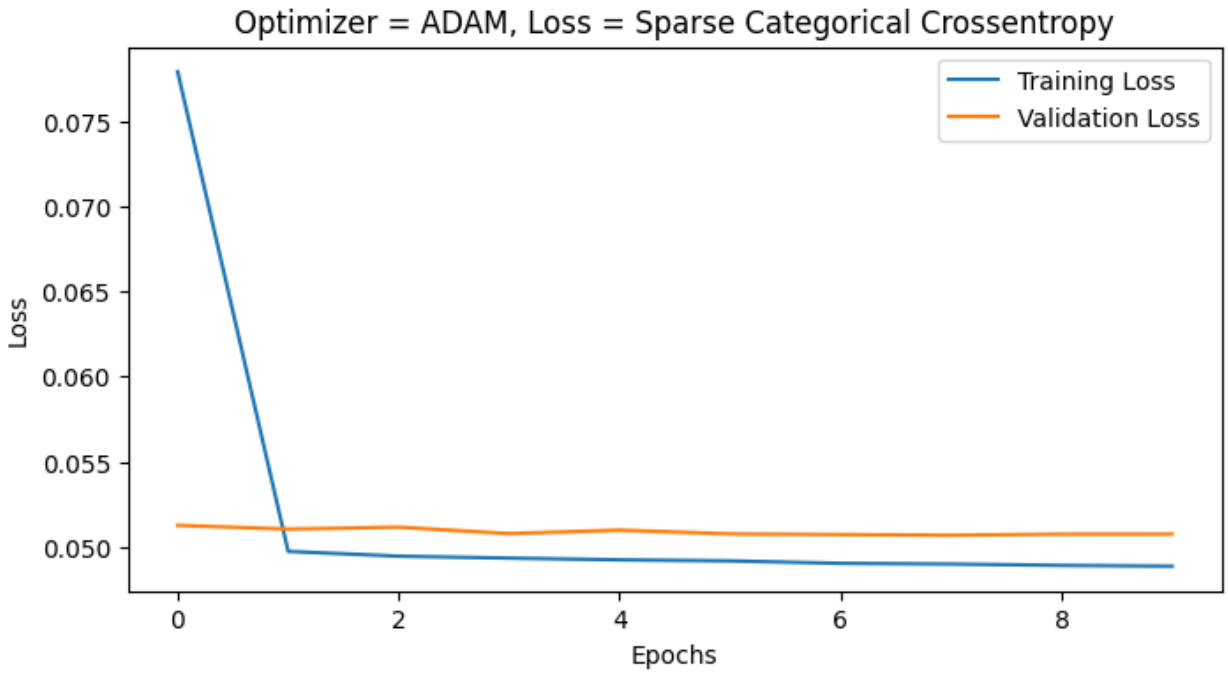
در ابتدا با رسم ماتریس هم بستگی دبیده میشود که ستون هدف که خودمان در ست کردیم با ستون هفت ام بیشترین وابستگی را دارند. پس تنها همان ستون را جدا کرده و به تنهایی مورد بررسی قرار میدهیم

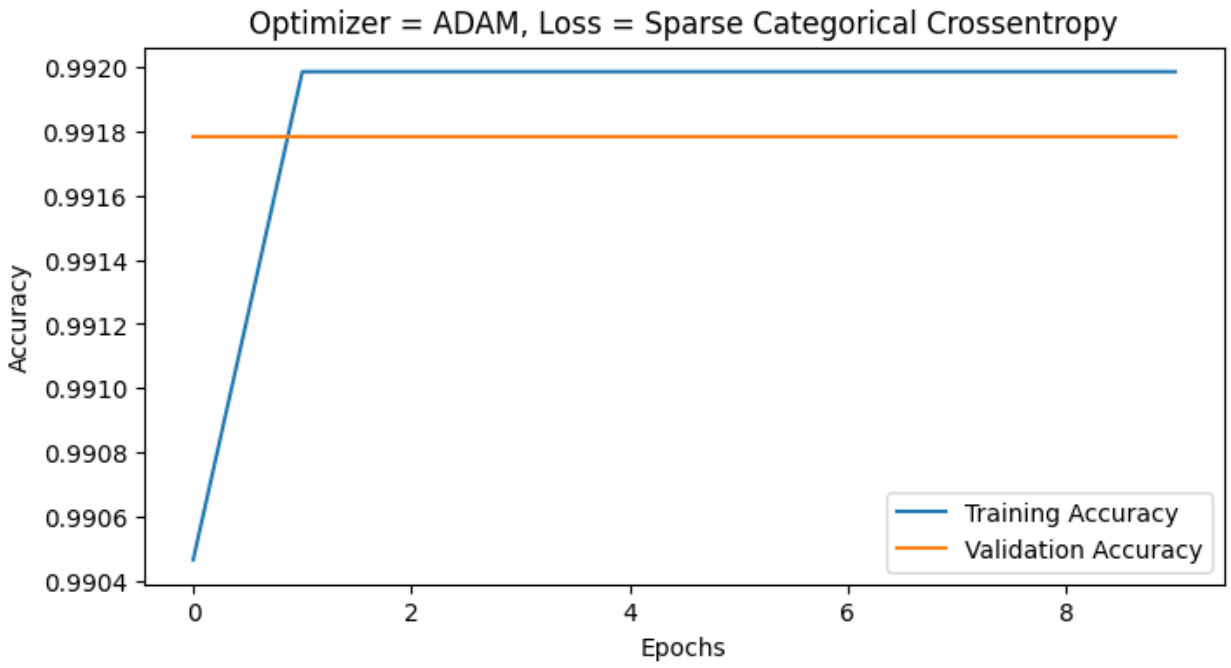



حال پس از جدا سازی این ستون به سراغ جدا سازی داده های اموزش و تست و ارزیابی میپردازیم تا باقی لامور را با استفاده از انها انجام دهیم.

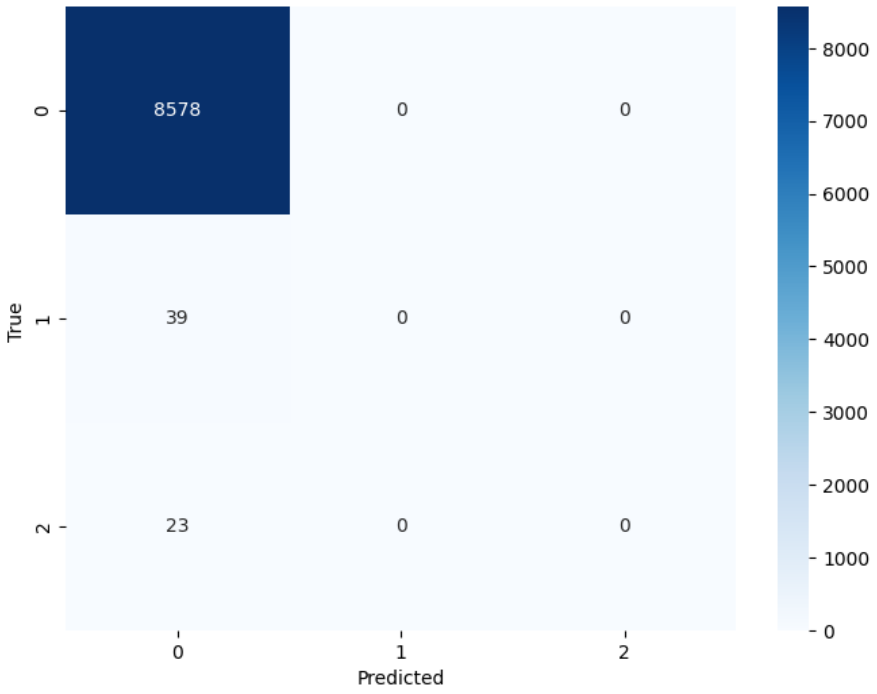
در  نهایت هم پس از جدا سازی داده ها به سراغ رسم نمودار هزینه و دقت میرویم: دقت مربوط به سیستم پیاده سازی شده.

```
Epoch 1/10
6912/6912 - 18s - loss: 0.0779 - accuracy: 0.9905 - val_loss: 0.0513 - val_accuracy: 0.9918 - 18s/epoch - 3ms/step
Epoch 2/10
6912/6912 - 13s - loss: 0.0497 - accuracy: 0.9920 - val_loss: 0.0510 - val_accuracy: 0.9918 - 13s/epoch - 2ms/step
Epoch 3/10
6912/6912 - 10s - loss: 0.0495 - accuracy: 0.9920 - val_loss: 0.0512 - val_accuracy: 0.9918 - 10s/epoch - 1ms/step
Epoch 4/10
6912/6912 - 15s - loss: 0.0493 - accuracy: 0.9920 - val_loss: 0.0508 - val_accuracy: 0.9918 - 15s/epoch - 2ms/step
Epoch 5/10
6912/6912 - 17s - loss: 0.0492 - accuracy: 0.9920 - val_loss: 0.0510 - val_accuracy: 0.9918 - 17s/epoch - 2ms/step
Epoch 6/10
6912/6912 - 20s - loss: 0.0492 - accuracy: 0.9920 - val_loss: 0.0508 - val_accuracy: 0.9918 - 20s/epoch - 3ms/step
Epoch 7/10
6912/6912 - 20s - loss: 0.0490 - accuracy: 0.9920 - val_loss: 0.0507 - val_accuracy: 0.9918 - 20s/epoch - 3ms/step
Epoch 8/10
6912/6912 - 14s - loss: 0.0490 - accuracy: 0.9920 - val_loss: 0.0507 - val_accuracy: 0.9918 - 14s/epoch - 2ms/step
Epoch 9/10
6912/6912 - 13s - loss: 0.0489 - accuracy: 0.9920 - val_loss: 0.0508 - val_accuracy: 0.9918 - 13s/epoch - 2ms/step
Epoch 10/10
6912/6912 - 11s - loss: 0.0489 - accuracy: 0.9920 - val_loss: 0.0508 - val_accuracy: 0.9918 - 11s/epoch - 2ms/step
```






در نهایت هم کانفیوژن ماتریس را رسم میکنیم تا عملکرد سیستم را در یابیم

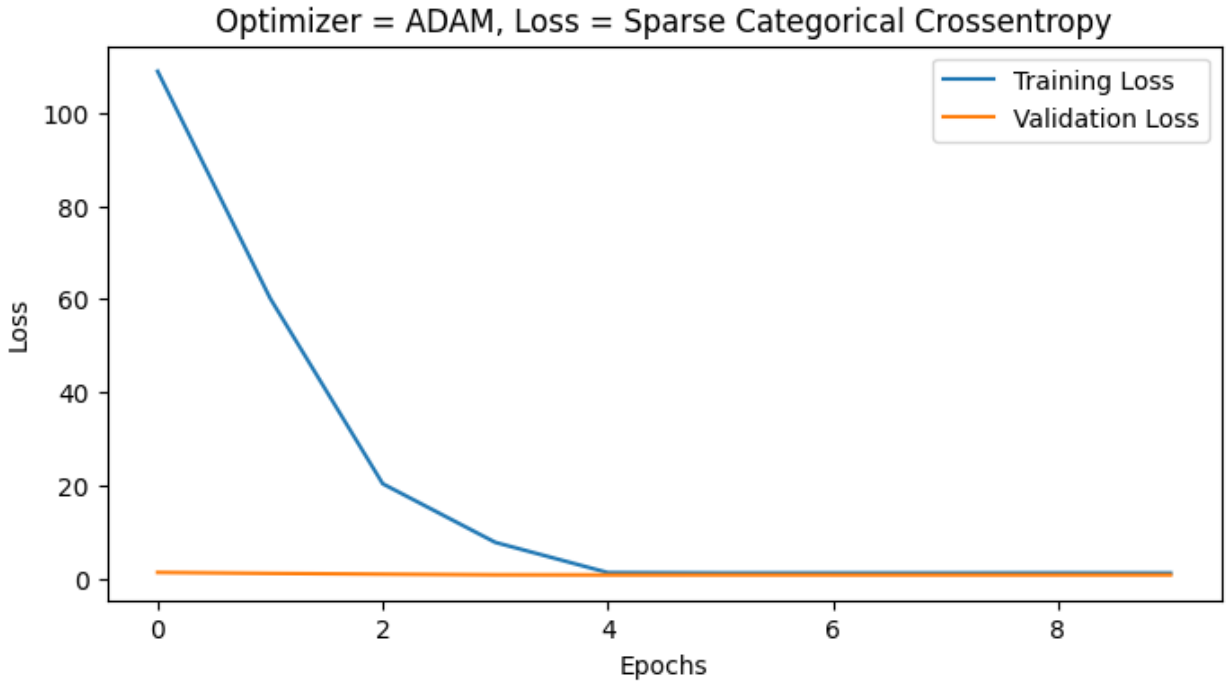


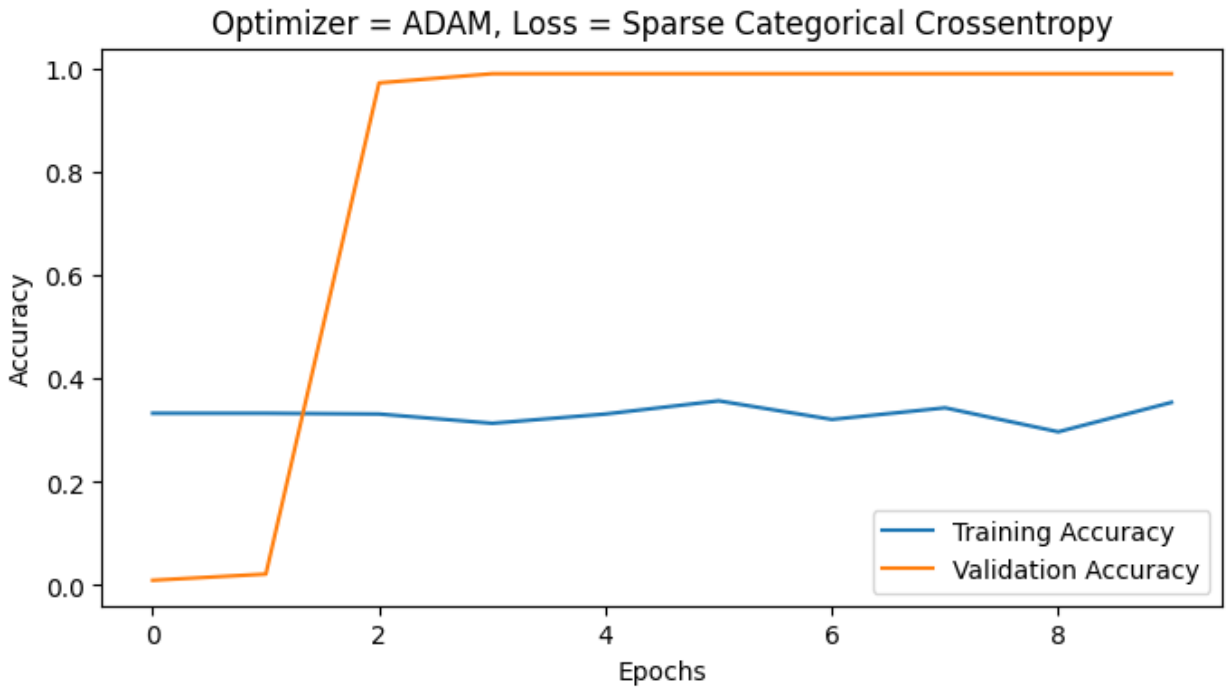

همان طور که دیده میشود تعداد داده هایی که نرمال هستند بسیار زیاد می باشد و سیسیتم تصمیم بر این گرفته تا تمامی موارد را نرمال لیبل بزند .

دقت شود لیبل صفر برای داده های نرمال و لیبل 1و2 به ترتیب برای داده های خطای 18 و 16 می باشند. حال به سراغ under

samplingمیرویم

```
Epoch 1/10
67/67 - 6s - loss: 108.9057 - accuracy: 0.3333 - val_loss: 1.2414 - val_accuracy: 0.0094 - 6s/epoch - 83ms/step
Epoch 2/10
67/67 - 1s - loss: 60.1252 - accuracy: 0.3333 - val_loss: 1.0335 - val_accuracy: 0.0214 - 1s/epoch - 15ms/step
Epoch 3/10
67/67 - 1s - loss: 20.2900 - accuracy: 0.3318 - val_loss: 0.8762 - val_accuracy: 0.9740 - 1s/epoch - 21ms/step
Epoch 4/10
67/67 - 1s - loss: 7.6972 - accuracy: 0.3138 - val_loss: 0.7216 - val_accuracy: 0.9918 - 1s/epoch - 16ms/step
Epoch 5/10
67/67 - 1s - loss: 1.2030 - accuracy: 0.3318 - val_loss: 0.6952 - val_accuracy: 0.9918 - 1s/epoch - 21ms/step
Epoch 6/10
67/67 - 1s - loss: 1.1209 - accuracy: 0.3574 - val_loss: 0.6989 - val_accuracy: 0.9918 - 989ms/epoch - 15ms/step
Epoch 7/10
67/67 - 1s - loss: 1.1198 - accuracy: 0.3213 - val_loss: 0.6974 - val_accuracy: 0.9918 - 1s/epoch - 16ms/step
Epoch 8/10
67/67 - 3s - loss: 1.1165 - accuracy: 0.3438 - val_loss: 0.6958 - val_accuracy: 0.9918 - 3s/epoch - 40ms/step
Epoch 9/10
67/67 - 1s - loss: 1.1310 - accuracy: 0.2973 - val_loss: 0.6973 - val_accuracy: 0.9918 - 1s/epoch - 17ms/step
Epoch 10/10
67/67 - 1s - loss: 1.1056 - accuracy: 0.3544 - val_loss: 0.6990 - val_accuracy: 0.9918 - 1s/epoch - 16ms/step
```




Optimizer = ADAM, Loss = Sparse Categorical Crossentropy
Accuracy
Epochs
Training Accuracy
Validation Accuracy
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8