به نام خدا


# گزارش پروژه درس داده کاوی

**نویسنده**

سید حسین فانی یزدی

**استاد**

دکتر جلالی

**تاریخ**

1401/04/19

ما دراین پروژه سعی کردیم از دیتاست پزشکی استفاده کنیم با تعداد دیتا 14980 عدد که توضیحات مربوط به دیتاست در پایین نوشته شده است :

در این دیتاست میاد حالت باز و بسته بودن چشم رو از داده های عصبی EEG پیش بینی میکنه. مدت زمان اندازه گیری 117 ثانیه بوده. حالت چشم را از طریق یک دوربین فیلم برداری میکنند و سپس هر فریم از ویدیو را تجزیه و تحلیل انجام میدهند. در این دیتاست مقدار ویژگی eyeDetection را هدف خود در نظر میگیریم که مقدار 1 برابر با بسته بودن چشم و مقدار 0 برابر با باز بودن چشم است. مقادیر دیگر ستون ها هم مربوط به پارامتر های اندازه گیری EEG است. پراکندگی پارامتر eyeDetection میزان 8257 عدد مربوط به پارامتر صفر و مقدار 6723 عدد مربوط به پارامتر یک می باشد که پراکندگی خوبی هست.

تصویری از دیتاست :

| | AF3 | F7 | F3 | FC5 | T7 | P7 | O1 | O2 | P8 | T8 | FC6 | F4 | F8 | AF4 | eyeDetection |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 4329.23 | 4009.23 | 4289.23 | 4148.21 | 4350.26 | 4586.15 | 4096.92 | 4641.03 | 4222.05 | 4238.46 | 4211.28 | 4280.51 | 4635.90 | 4393.85 | 0 |
| 1 | 4324.62 | 4004.62 | 4293.85 | 4148.72 | 4342.05 | 4586.67 | 4097.44 | 4638.97 | 4210.77 | 4226.67 | 4207.69 | 4279.49 | 4632.82 | 4384.10 | 0 |
| 2 | 4327.69 | 4006.67 | 4295.38 | 4156.41 | 4336.92 | 4583.59 | 4096.92 | 4630.26 | 4207.69 | 4222.05 | 4206.67 | 4282.05 | 4628.72 | 4389.23 | 0 |
| 3 | 4328.72 | 4011.79 | 4296.41 | 4155.90 | 4343.59 | 4582.56 | 4097.44 | 4630.77 | 4217.44 | 4235.38 | 4210.77 | 4287.69 | 4632.31 | 4396.41 | 0 |
| 4 | 4326.15 | 4011.79 | 4292.31 | 4151.28 | 4347.69 | 4586.67 | 4095.90 | 4627.69 | 4210.77 | 4244.10 | 4212.82 | 4288.21 | 4632.82 | 4398.46 | 0 |

*Figure 1 - Dataset*

لینک دیتاست :

https://www.kaggle.com/datasets/robikscube/eye-state-classification-eeg-dataset?select=EEG_Eye_State_Classification.csv

اولین الگوریتم استفاده شده الگوریتم KNN است :

در این الگوریتم از مقدار K=1 استفاده میکنیم که نتایج را در زیر مشاهده میکنید :

```
[[1984   43]
 [  41 1677]]
              precision    recall  f1-score   support

           0       0.98      0.98      0.98      2027
           1       0.97      0.98      0.98      1718

    accuracy                           0.98      3745
   macro avg       0.98      0.98      0.98      3745
weighted avg       0.98      0.98      0.98      3745

Accuracy: 0.9775700934579439
```

*Figure2 - Report Model For K=5*

بعد از آن میایم و مقدار Error را رسم میکنیم :

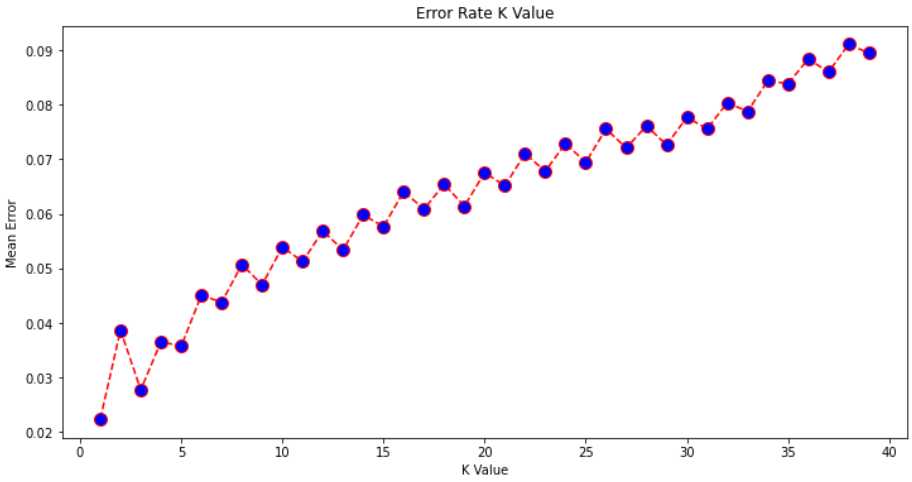


*Figure3 - Error For K From 1 to 40*

همانطور که مشاهده میکنید از مقدار k=1 کمترین مقدار error است.

همچنین نمودار مقدار صحت و تعداد k را برای داده های تمرین و تست رسم کرده ایم :

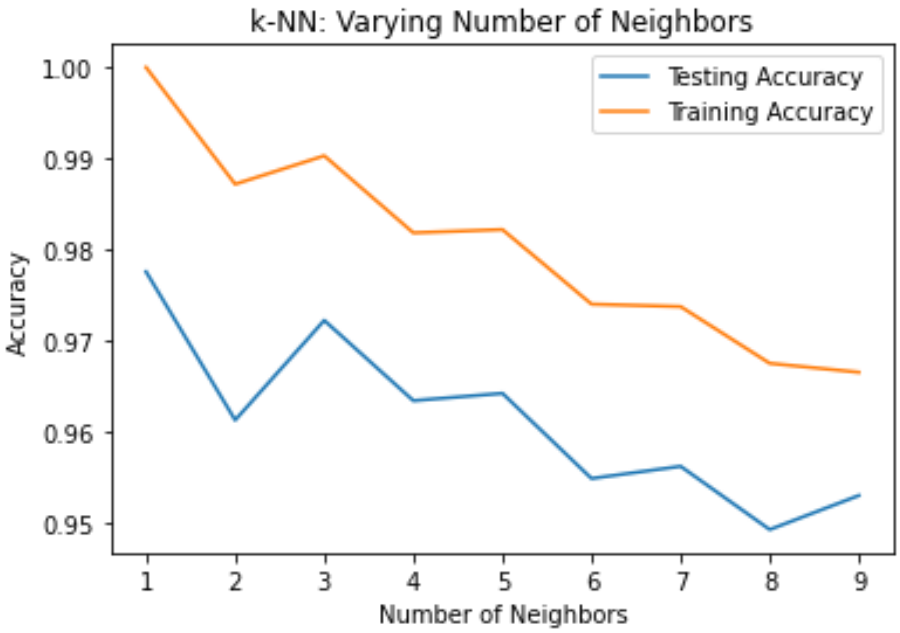


*Figure4 - ACC And K*

## الگوریتم Naïve Bayes

در این الگوریتم مقدار Accuracy ما حدودا 0.54 هست که نسبت به عملکرد الگوریتم KNN عملکرد ضعیف تری می باشدکه در زیر مشاهده میکنید:

```
[[2865    1]
 [2377    0]]
Accuracy: 0.5464428762159069
Precision: 0.27327355971003436
Recall: 0.4998255408234473
```

سپس ماتریس درهم ریختگی را برای این الگوریتم رسم میکنیم :

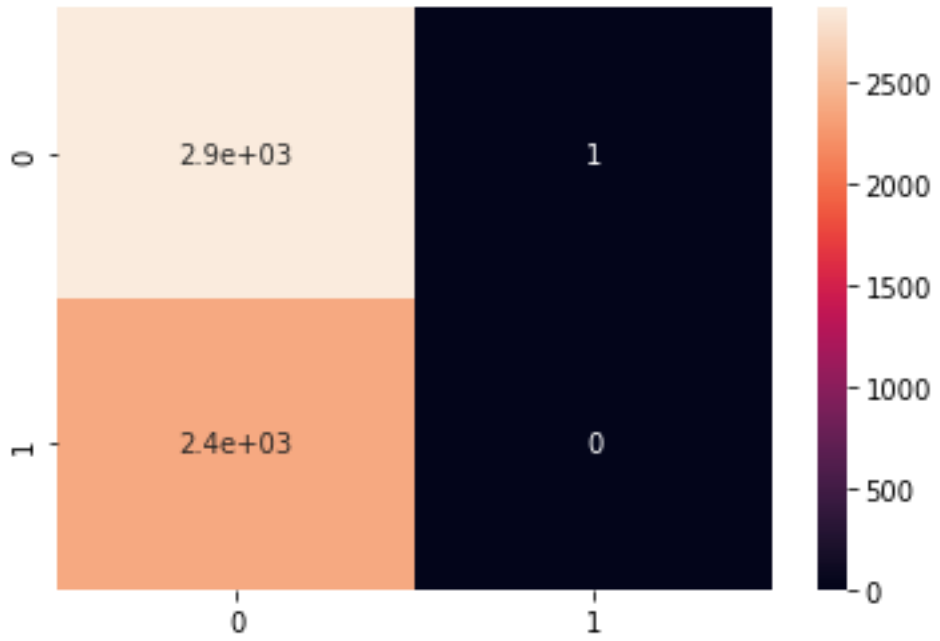


*Figure5 - TF Table*

در قسمت روش دسته بندی با الگوریتم درخت تصمیم در ابتدا از ID3 استفاده میکنیم و مقدار صحت داده شده تقریبا نزدیک به الگوریتم بیز هستش که شاهد مقدار ACC زیر هستیم :

```
0.5137540453074434
```

*Figure6 - ACC ID3*

بعد از آن به سراغ الگوریتم CART میرویم که نتیجه خوبی به ما میدهد که بهتر از الگوریتم های ID3 و Bayes است ولی نسبت به KNN ضعیف تر عمل کرده است :

```
[[2080  341]
 [ 386 1687]]
              precision    recall  f1-score   support

           0       0.84      0.86      0.85      2421
           1       0.83      0.81      0.82      2073

    accuracy                           0.84      4494
   macro avg       0.84      0.84      0.84      4494
weighted avg       0.84      0.84      0.84      4494

Accuracy: 0.8382287494437027
Precision: 0.8376626259136085
Recall: 0.8364727711157378
```

*Figure 7 - ACC CART*

الگوریتم C4.5 :

دراین الگوریتم به دلیل اینکه هدفمون object نیست از Regression استفاده میکنیم که نتایج زیر را به ما میدهد :

```
finished in  185.04679679870605  seconds
-------------------------
Evaluate  train set
-------------------------
MAE:  0.16695594125500668
MSE:  0.16565420560747662
RMSE:  0.40700639504493863
RAE:  0.6075409894361118
RRSE:  0.8183146919350317
Mean:  0.4487983978638184
MAE / Mean:  37.20065446973078 %
RMSE / Mean:  90.68802317080441 %
```

*Figure 8 - ACC C4.5*

در خوشه بندی با استفاده از داده ها مشخص شد که الگوریتم KNN میتواند روی این دیتاست عملکرد خیلی خوبی نشان دهند.

بخش 2 : خوشه بندی

در خوشه بندی اولین الگوریتم مربوط به K-means می باشد که ابتدا میایم و با دوتا خوشه شروع میکنیم. اینم بگم که برای این دیتاست ما ستون eyeDetection را برای کلاس و خوشه بندی در جزئیات به شرح زیر است :

```
[[1580    6]
 [ 102 1308]]
Accuracy: 0.9639519359145527
Precision: 0.9673958486038039
Recall: 0.9619382361621636
```

*Figure 9 - Kmeans for k=1*

در نمودار زیر مشاهده میکنیم که چرا مقدار k=1 بهتر است :

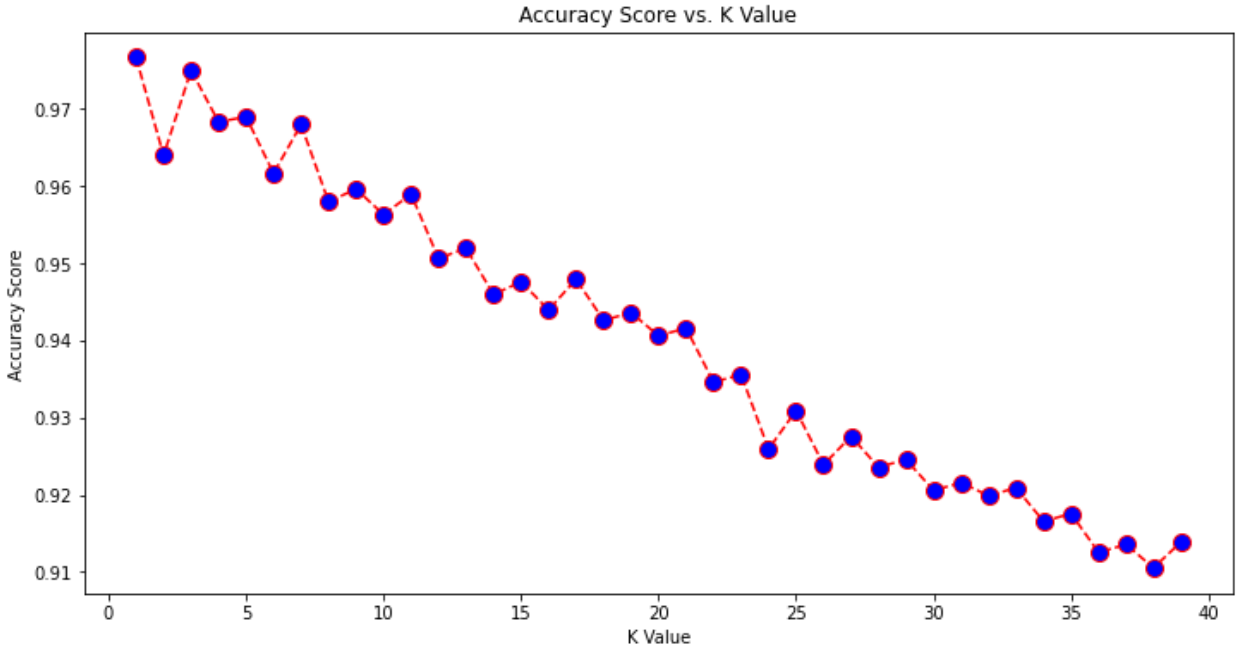


*Figure 10 - ACC and K value*

الگوریتم OPTC : برای این الگوریتم اومدم و سه تا ستون رو در نظر گرفتم به نام های F4 و F8 و eyeDetection سپس داده هارو نرمال میکنیم و نتایج به صورت زیر می باشد :

```
Result: 5446 out of 14980 samples were correctly labeled.
Accuracy score: 0.36
```

*Figure 11 - ACC for OPTICS*

نتیجه متاسفانه بسیار کم است که ممکن است از الگوریتم نوشته شده باشد.

بخش سوم الگوریتم جدید برای کلاس بندی :

از الگوریتم SVM استفاده کردیم و در دوحالت مقدار صحت را بررسی میکنیم. اولینش بدون پیش پردازش است البته که داده ها هیچ گونه داده NaN ندارد و تقریبا پیش پردازش شده است مقدار صحت را مشاهده میکنید :

```
[[1955  522]
 [1062  955]]
              precision    recall  f1-score   support

           0       0.65      0.79      0.71      2477
           1       0.65      0.47      0.55      2017

    accuracy                           0.65      4494
   macro avg       0.65      0.63      0.63      4494
weighted avg       0.65      0.65      0.64      4494

Accuracy: 0.6475300400534045
```

*Figure 12 - ACC SVM*

در حالت دوم مربوط به نرمال سازی داده ها است که نمیدونم چرا ولی میزان صحت کمتر شده است :

```
[[1912  565]
 [1076  941]]
              precision    recall  f1-score   support

           0       0.64      0.77      0.70      2477
           1       0.62      0.47      0.53      2017

    accuracy                           0.63      4494
   macro avg       0.63      0.62      0.62      4494
weighted avg       0.63      0.63      0.63      4494

Accuracy: 0.6348464619492656
```

*Figure 13 - ACC SVM with normalize*

برای خوشه بندی از الگوریتم mini batch k-means استفاده کردم که در اخر اومدم و مقادیر هدف مقایسه کردم برای صحت که به صورت زیر است :

```
Result: 8256 out of 14980 samples were correctly labeled.
Accuracy score: 0.55
```

*Figure 14 - ACC for mini batch k-means*

بخش چهارم – تاثیر پیش پردازش داده ها:

در این دیتاست استفاده شده تمام داده ها از قبل پیش پردازش شده بوده اما در الگوریتم SVM در دوحالت عادی و نرمال سازی شده استفاده کردیم که در حالت پیش پردازش شده نتیجه صحت کمتر شده بود.

برای مقایسه هم من از cross validation استفاده کردم که حاصل به صورت زیر است :

```
LR: 0.638631 (0.009929)
LDA: 0.638809 (0.008581)
KNN: 0.958967 (0.002743)
CART: 0.829907 (0.013990)
NB: 0.543741 (0.033774)
SVM: 0.562619 (0.025767)
```

*Figure 15 – Compare*

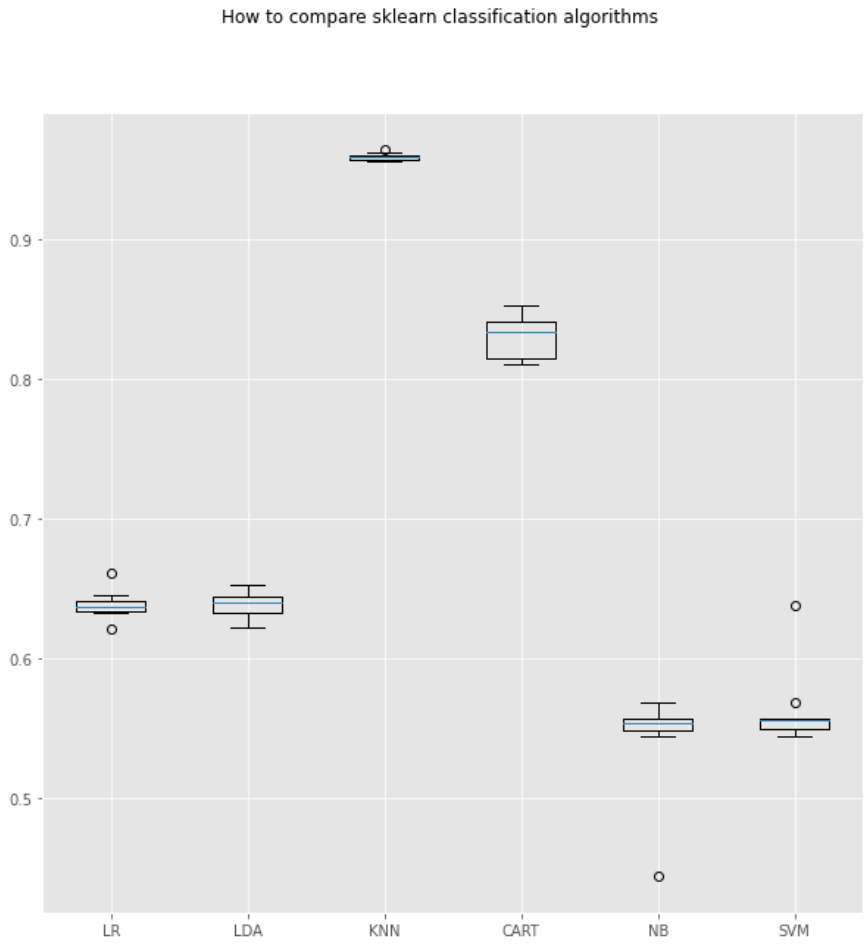


*Figure 16 - Compare chart*

کد های الگوریتم ها و دیتاست داخل لینک زیر است :

https://drive.google.com/file/d/1KoP4PCFRvte5HxjVHpz_qTzL-CadguQi/view?usp=sharing