نام و نام خانوادگی: درسا الهوئی

شماره دانشجویی: ۹۷۲۲۲۰۰۵

شماره تمرین: ۴ **تاریخ تحویل: ۱۷ دی ۱۴۰۰**

عنوان تمرین: پیشبینی دانلود یک برنامه توسط کاربر با استفاده از الگوریتم های طبقه بندی مختلف

---

۱. مقدمه

هدف این تمرین، پیشبینی دانلود یک برنامه توسط کاربر با توجه به اطلاعات در دسترس است. به این منظور مدل های مختلفی را می آزماییم تا دقیق ترین آنها را برگزینیم.

۲. پردازش داده ها

در گام اول تعداد مقادیر تهی[۱]در هر ستون از داده اولیه را بدست می آوریم.

```
ip                     0
app                    0
device                 0
os                     0
channel                0
click_time             0
attributed_time    1843715
is_attributed          0
dtype: int64
```

بیش از $\frac{۲}{۳}$ داده ها در این ستون مقدار تهی دارند. در نتیجه این ستون را حذف می کنیم.

حدود $\frac{۳}{۴}$ داده ها در ستون is_attributed (ستونی که قصد پیشبینی کردن آن در داده های آزمایشی را داریم.) مقدار ۰ دارند (به این معنی که کاربر برنامه را دانلود نکرده است.). به منظور متعادل سازی داده ها به صورت تصادفی مجموعه داده جدیدی می سازیم که تعداد ۰ها و ۱ها در این ستون، مقداری برابر باشد.

سپس داده های آموزشی و آزمایشی را با نسبت های ۰,۸ و ۰,۲ از داده اولیه، جدا می کنیم.

در ستون click_time با توجه به بررسی ها، از آنجا که تاریخ با تلرانس چند روز متغیر است، تمرکز خود را بر روی ساعتی که در طول یک شبانه روز دانلود در آن صورت گرفته است می گذاریم. بر همین اساس ۲۴ ساعت روز را به سه بازه ۸ ساعته تقسیم کردیم. به بازه ساعت ۶ الی ۱۳ عدد ۰، به بازه ساعت ۱۴ الی ۲۱ عدد ۱ و به

[۱] Null

بازه ساعت ۲۲ الی ۵ صبح عدد ۲ را نسبت دادیم. این اعداد انتسابی را با عنوان ستون time به داده های آموزشی و آزمایشی می افزاییم و ستون click_time را حذف می کنیم.

در ادامه به منظور انکود[۱]کردن داده ها، از آنجا که تعداد مقادیر منحصر به فرد در هر ستون زیاد است و استفاده از روشهایی همچون وان_هات_انکودینگ[۲]حجم داده ها را به میزان خیلی زیادی افزایش می دهد، از روش تارگت_انکودینگ[۳]استفاده می کنیم که ستون جدیدی به داده ها نمی افزاید.

در مرحله بعد ستون is_attributed که ستون مورد بررسی ما با توجه به شواهد موجود می باشد را در هر دو مجمعه داده های آموزشی و آزمایشی جدا می کنیم.

در این گام پردازش داده ها پایان می یابد.

۳. رگرسیون لجیستیک[۴]

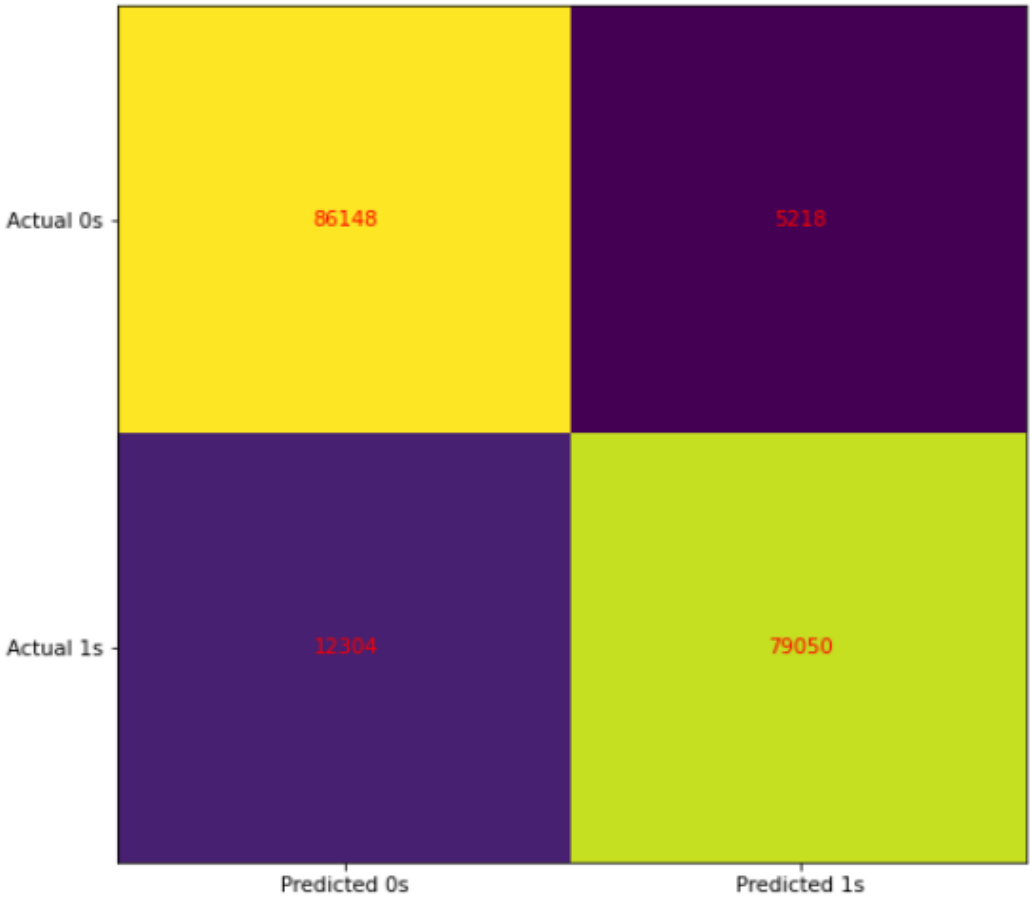
رگرسیون لجیستیک، یکی از الگوریتم های یادگیری ماشین[۵]است که از آن غالبا در مسائل طبقه بندی[۶]که متغیرهای کیفی[۷]مطرح هستند استفاده می شود. به طور کلی خروجی در رگرسیون لجستیک به شکل ۰ یا ۱ می باشد. زمانی که تعداد کلاس های خروجی برابر ۲ باشد، به آن طبقه بندی باینری[۸]گفته می شود. تفاوت رگرسیون لجستیک با رگرسیون خطی[۹]این است که در رگرسیون خطی، معادله خط می تواند هر مقداری داشته باشد در حالی که در رگرسیون لجستیک مقادیری که در خروجی داریم احتمالا متعلق به یکی از دو کلاس ۰ یا ۱ است. در رگرسیون خطی هدف پیش بینی مقادیر پیوسته در خروجی است در حالی که هدف رگرسیون لجستیک پیشبینی مقادیر گسسته است و در آن قصد داریم داده ها را به دو یا چند کلاس مشخص طبقه بندی کنیم. بنابراین مقادیر باید بین ۰ و ۱ باشند که این موضوع با معادله خطی امکان پذیر نیست. تابعی که در رگرسیون لجستیک استفاده می شود، تابع سیگموید[۱۰]یا همان لجستیک می باشد. مرز تصمیم گیری[۱۱]نیز به ما کمک می کند تا دو کلاس را از یکدیگر متمایز کنیم. این مرز تصمیم گیری را تابع لجستیک مشخص می کند.

در اینجا ما مدل رگرسیون لجیستیک را روی داده های آموزشی، اجرا می کنیم تا الگوها و شواهد موجود را بیاموزد. سپس داده های آزمایشی را به آن ورودی می دهیم تا میزان دقت مدل را بدست آوریم. در اینجا مدل ما از دقت ۹۰٪ برخوردار است.

---

۱ Encode
۲ One Hot Encoding
۳ Target Encoding
۴ Logistic Regression
۵ Machine Learning
۶ Classification
۷ Categorical
۸ Binary Classification
۹ Linear Regression
۱۰ Sigmoid
۱۱ Decision Boundary

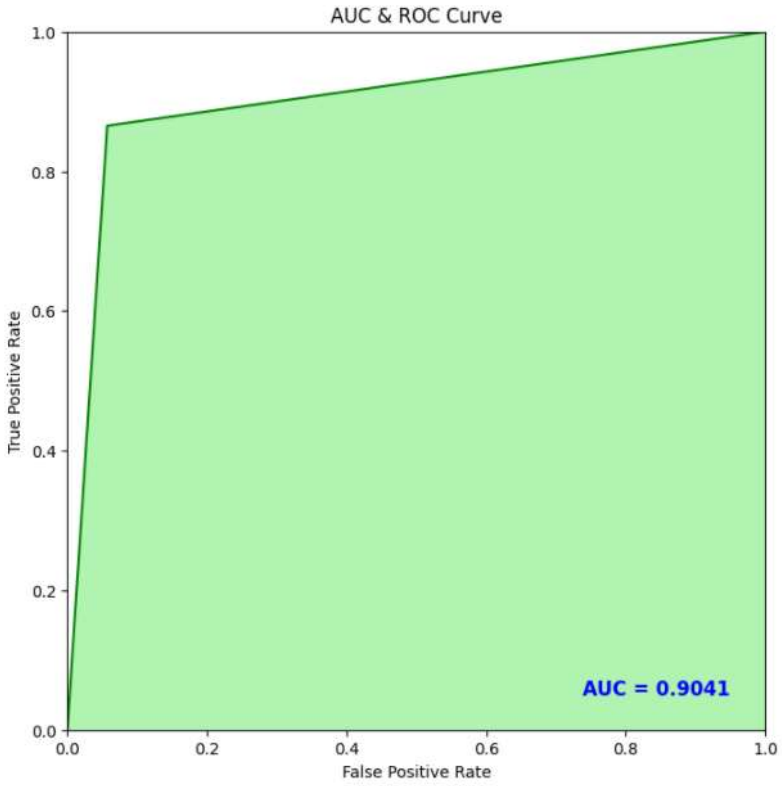
در گام بعدی ماتریس درهم ریختگی[1] را رسم می کنیم.



تفسیر این ماتریس به این صورت است که در مربع بالا سمت چپ تعداد داده هایی که مقدار واقعی آنها ۰ می باشد و مدل آنها را ۰ پیشبینی کرده است نوشته شده و در مربع بالا سمت راست تعداد داده هایی که مقدار واقعی آن ها ۰ است اما مدل آنها را ۱ پیشبینی کرده است نوشته شده است. در مربع پایین سمت چپ تعداد داده هایی که مقدار واقعی آنها ۱ است اما مدل آنها را ۰ پیشبینی کرده نوشته شده و در مربع پایین سمت راست تعداد داده هایی که مقدار ۱ ذاشته اند و مدل به درستی آنها را پیشبینی کرده، نوشته شده است.

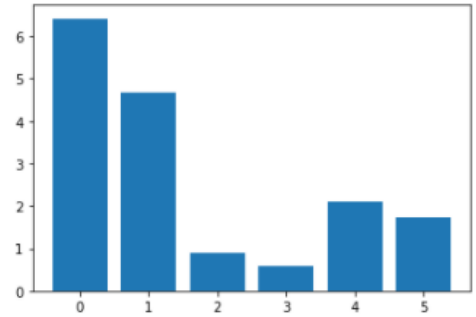
[1] Confusion Matrix

نمودار AUC و ROC نیز در تصویر زیر قابل مشاهده می باشد.

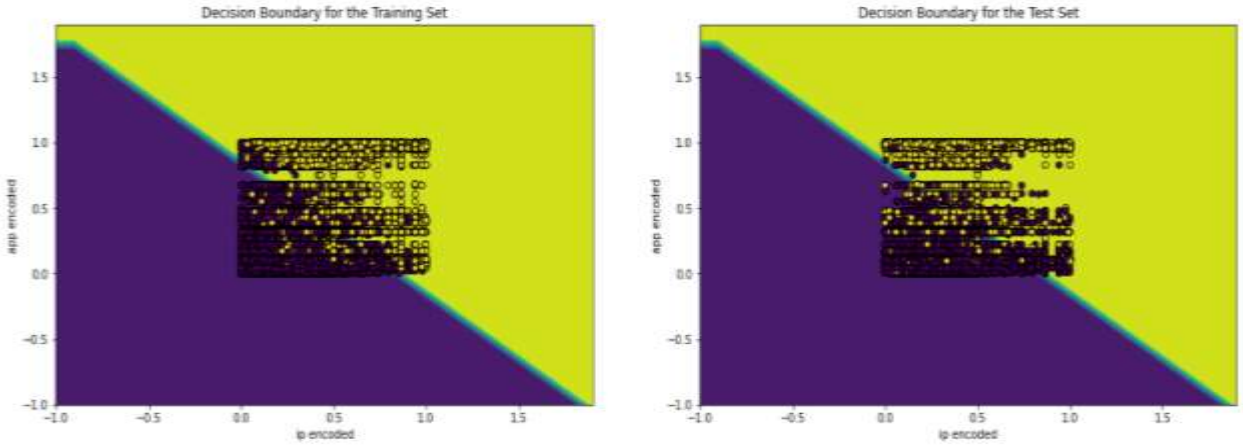


به منظور انتخاب دو متغیر برای رسم مرز تصمیم گیری، از coef. استفاده کردیم و دو متغیری که بیشترین تاثیر در پیشبینی های ما داشتند (app و ip) را انتخاب کردیم.

```
Feature: ip encoded, Score: 6.4190024647518475
Feature: app encoded, Score: 4.685700913170679
Feature: device encoded, Score: 0.8844742771327117
Feature: os encoded, Score: 0.5737112997270695
Feature: channel encoded, Score: 2.088636521086708
Feature: time encoded, Score: 1.7363266396485608
```



مرز تصمیم گیری را با استفاده از دو متغیر ذکر شده در بالا رسم می کنیم. رنگ هر یک از نقطه ها، بیانگر کلاس آنها می باشد.



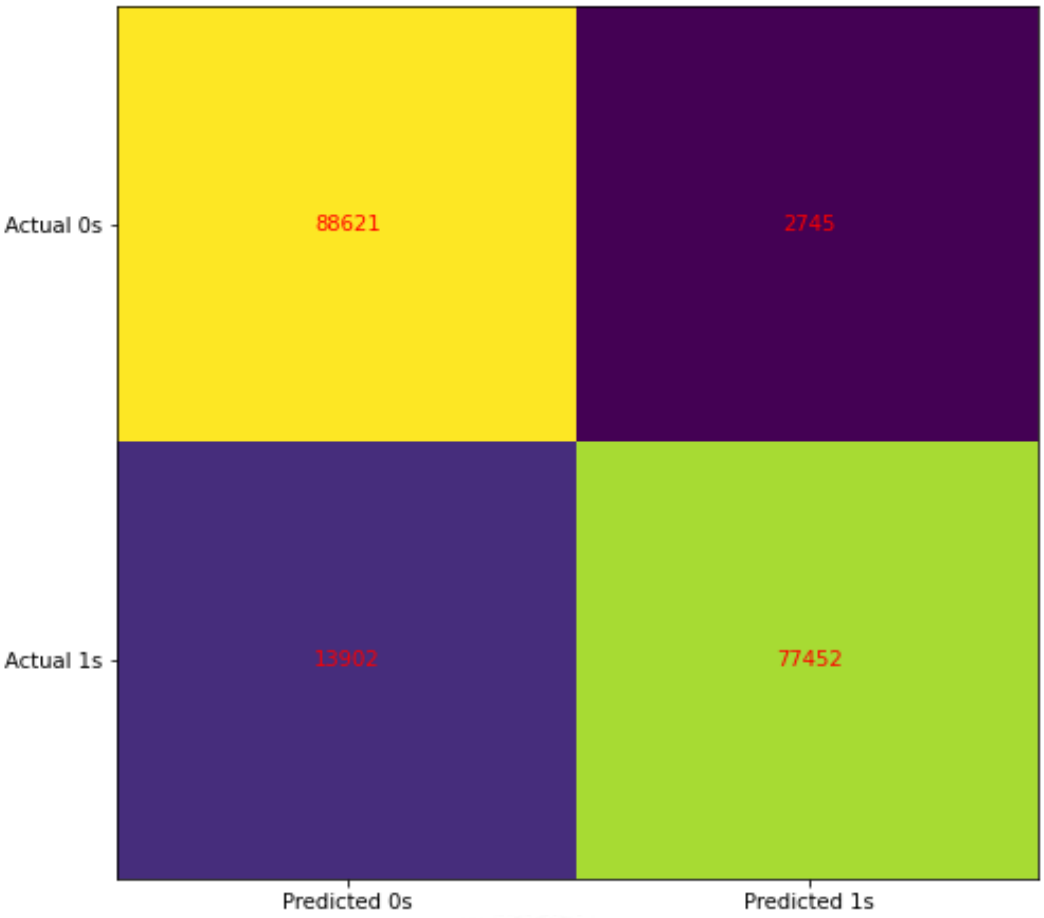
۴. ماشین بردار پشتیبان[۱]

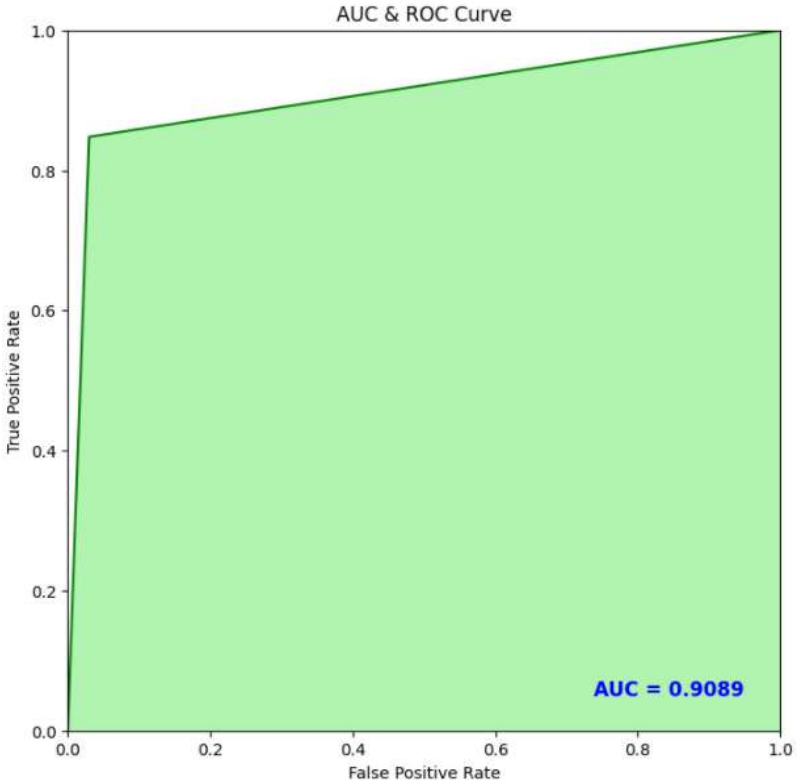
ماشین بردار پشتیبان یک الگوریتم نظارت شده یادگیری ماشین است که غالبا برای مسائل طبقه بندی استفاده می شود. در این الگوریتم، هر نمونه داده به عنوان یک نقطه در فضای n-بعدی (n تعداد ویژگی های یک نمونه داده می باشد.) روی نمودار پراکندگی داده ها ترسیم شده سپس با ترسیم یک خط راست، داده های مختلف و متمایز از یکدیگر دسته بندی می شوند. بردارهای پشتیبان در واقع مختصات یک مشاهده منفرد هستند.

به این جهت که محاسبات این بخش سنگین است، نمونه کوچکتری از داده ها را به عنوان داده آموزشی به مدل می دهیم. دقت مدل ما در اینجا، در حدود ۹۱٪ می باشد.

ماتریس در هم ریختگی و نمودار AUC و ROC مربوط به این مدل را می توانید در ادامه مشاهده کنید.

[۱] SVM


Actual 0s
Actual 1s
88621
2745
13902
77452
Predicted 0s
Predicted 1s

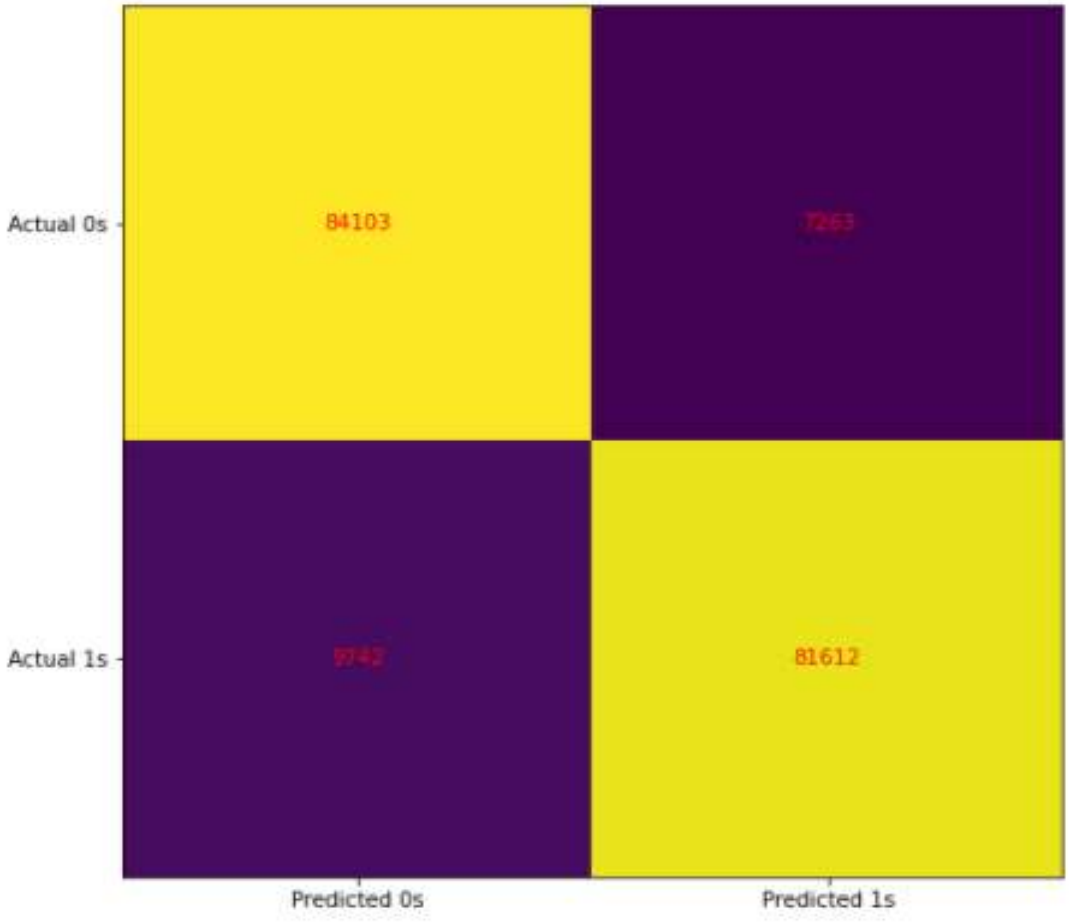

AUC & ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
AUC = 0.9089

۵. k-نزدیکترین همسایگی[۱]

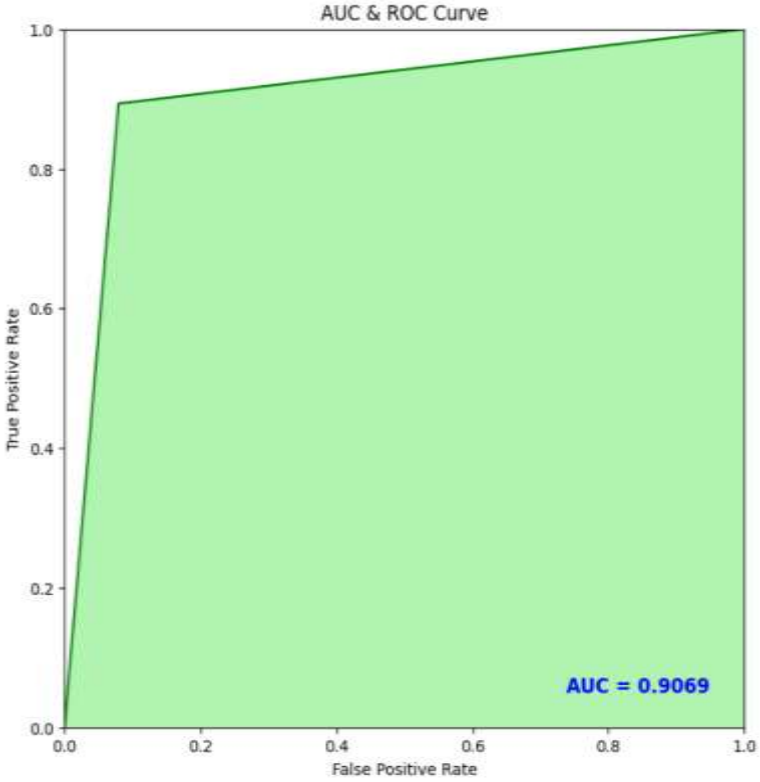
الگوریتم k-نزدیکترین همسایگی، یکی از ساده ترین الگوریتم ها در بحث طبقه بندی می باشد. این الگوریتم از تشابه ویژگی برای پیش بینی مقادیر نقاط داده جدید استفاده می کند. یعنی به نقطه جدید بر اساس میزان مطابقتی که با نقاط مجموعه آموزشی دارد، مقداری را تخصیص می دهد.

بعد از آموزش مدل روی مجموعه داده آموزشی، داده آزمایش را به مدل می دهیم. دقت مدل ما در اینجا، در حدود ۹۱٪ می باشد.

ماتریس در هم ریختگی و نمودار AUC و ROC مربوط به این مدل را می توانید در ادامه مشاهده کنید.



[۱] K Nearest Neighbor (KNN)



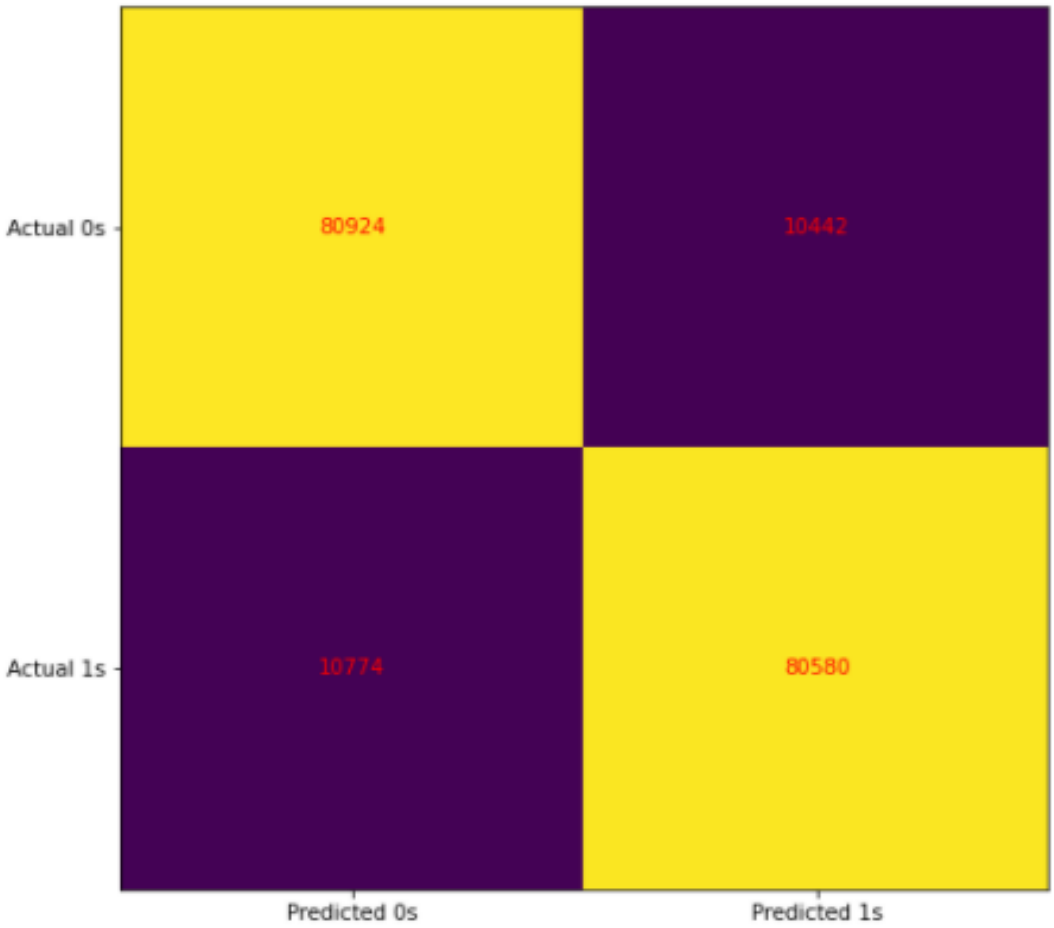
۶. درخت تصمیم[۱]

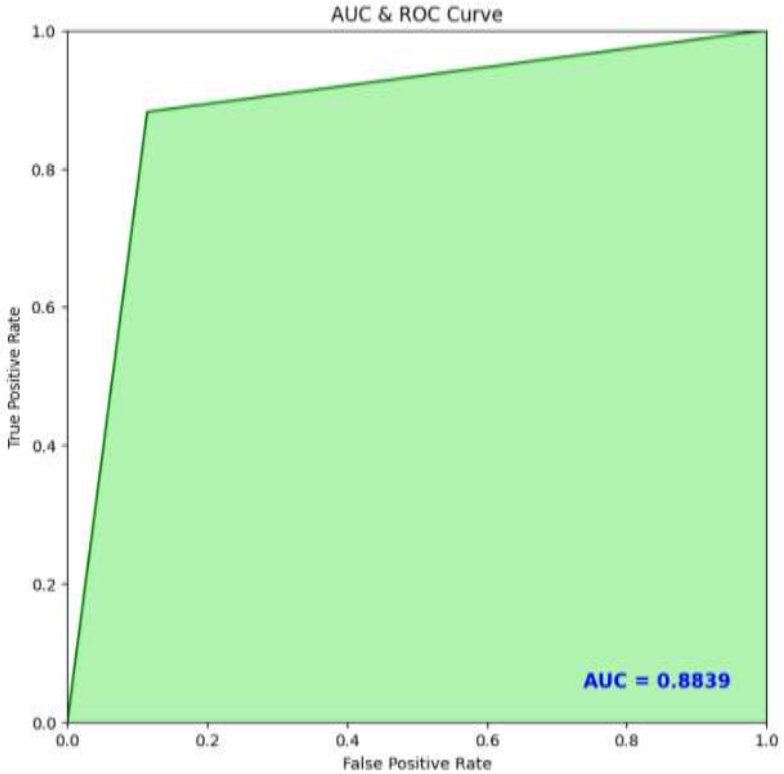
درخت تصمیم مدلی است که برای دسته بندی و رگرسیون به کار می رود. این مدل امکان تولید خروجی های گوناگون و انجام تصمیم گیری با داده ها را فراهم می کند.

بعد از آموزش مدل روی مجموعه داده آموزشی، داده آزمایش را به مدل می دهیم. دقت مدل ما در اینجا، در حدود ۸۸٪ می باشد.

ماتریس در هم ریختگی و نمودار AUC و ROC مربوط به این مدل را می توانید در ادامه مشاهده کنید.

[۱] Decision Tree


Actual 0s
Actual 1s
80924
10442
10774
80580
Predicted 0s
Predicted 1s

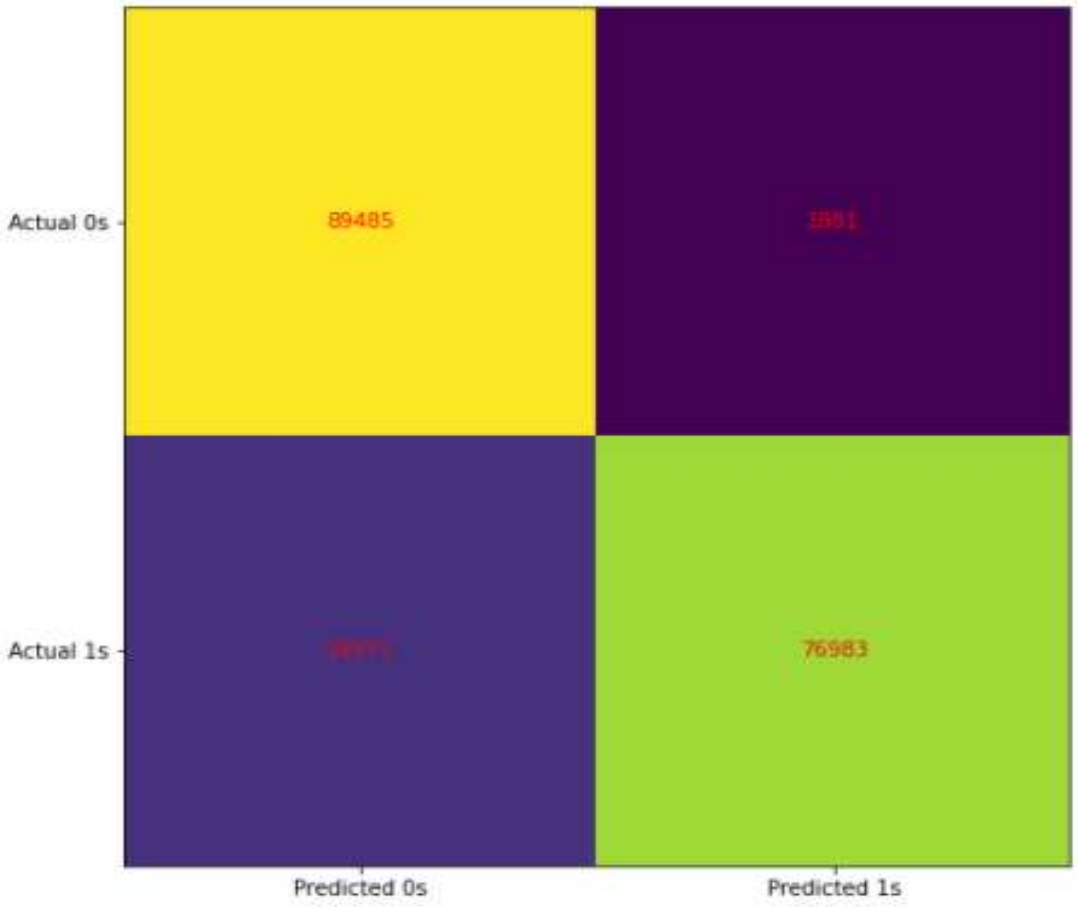

AUC & ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
AUC = 0.8839

۷. جنگل تصادفی[۱]

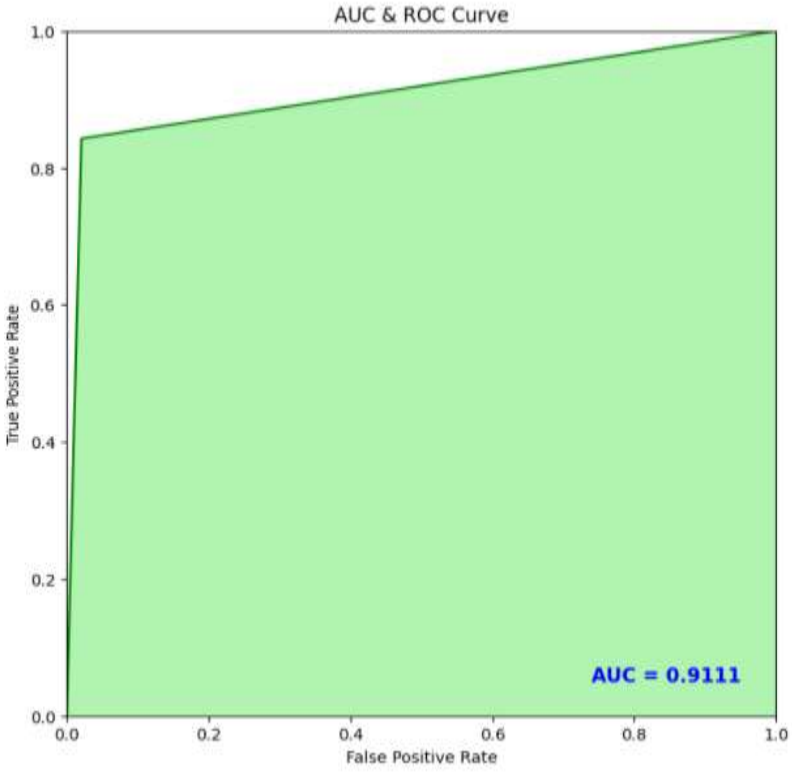
الگوریتم جنگل تصادفی، یک الگوریتم یادگیری ماشین با قابلیت استفاده آسان است که اغلب اوقات نتایج بسیار خوبی را حتی بدون تنظیم فراپارامترهای آن، فراهم می‌کند. این الگوریتم به دلیل سادگی و قابلیت استفاده، هم برای دسته‌بندی و هم رگرسیون، یکی از پر کاربردترین الگوریتم‌های یادگیری ماشین محسوب می‌شود.

بعد از آموزش مدل روی مجموعه داده آموزشی، داده آزمایش را به مدل می دهیم. دقت مدل ما در اینجا، در حدود ۹۱٪ می باشد.

ماتریس در هم ریختگی و نمودار AUC و ROC مربوط به این مدل را می توانید در ادامه مشاهده کنید.



[۱] Random Forest



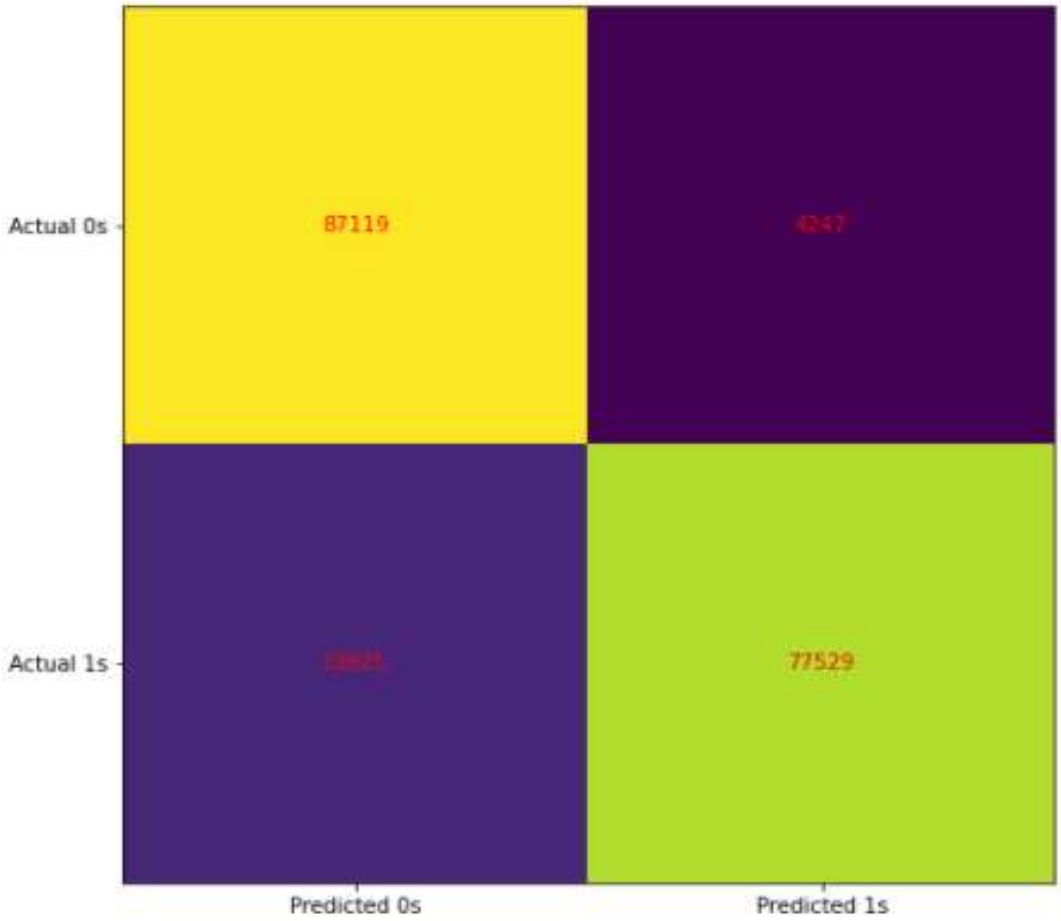
۸. نایو بیز[۱]

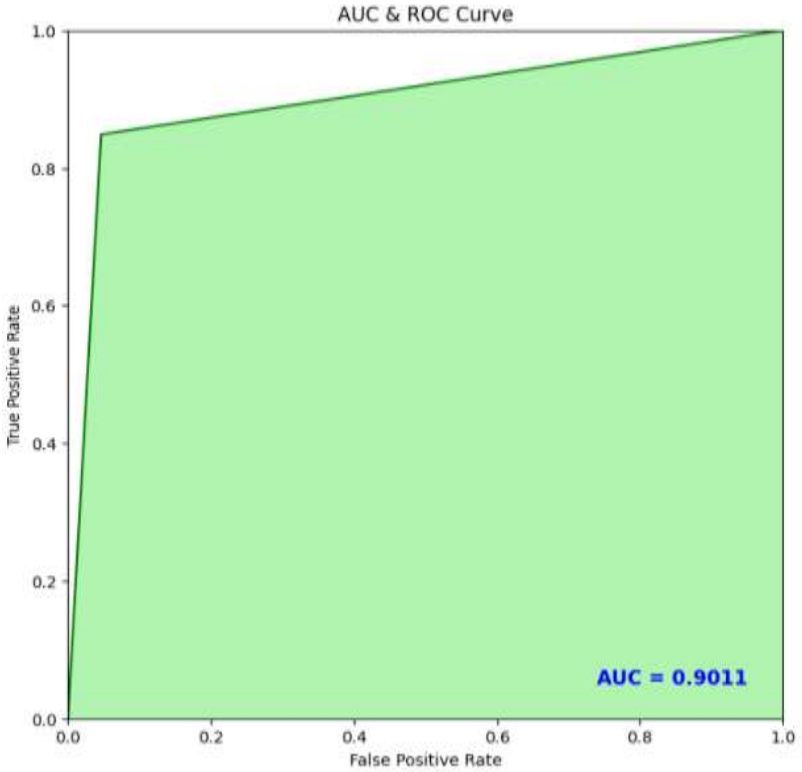
الگوریتم طبقه بندی کننده بیز در مواقعی که تعداد مشاهدات کمی در دسترس باشد عملکرد خوبی دارد. این الگوریتم، روشی برای دسته بندی پدیده ها براساس احتمال وقوع یا عدم وقوع آنها می باشد. الگوریتم طبقه بند بیز با بکارگیر قضیه بیز و فرض استقلال بین متغیرها به عنوان عضوی در خانواده دسته بندی های براساس احتمال[۲]قرار می گیرد. برای بکارگیری طبقه بند بیز ساده، الگوریتم یکتایی وجود ندارد. در عوض خانواده ای از الگوریتم ها هستند که با فرض استقلال ویژگیها یا متغیرها نسبت به یکدیگر عمل می کنند.

بعد از آموزش مدل روی مجموعه داده آموزشی، داده آزمایش را به مدل می دهیم. دقت مدل ما در اینجا، در حدود ۹۰٪ می باشد.

ماتریس در هم ریختگی و نمودار AUC و ROC مربوط به این مدل را می توانید در ادامه مشاهده کنید.

[۱]Naïve Bayes Algorithm [۲]Probabilistic Classifiers


Actual 0s
Actual 1s
87119
4247
77529
Predicted 0s
Predicted 1s


AUC & ROC Curve
True Positive Rate
False Positive Rate
AUC = 0.9011
0.0
0.2
0.4
0.6
0.8
1.0

۹. منابع

https://blog.faradars.org/

https://medium.com/analytics-vidhya/target-encoding-vs-one-hot-encoding-۶۴e۳b۷e۷a۲۷۶with-simple-examples-

https://www.kaggle.com/caesarlupum/catcomp-simple-target-encoding