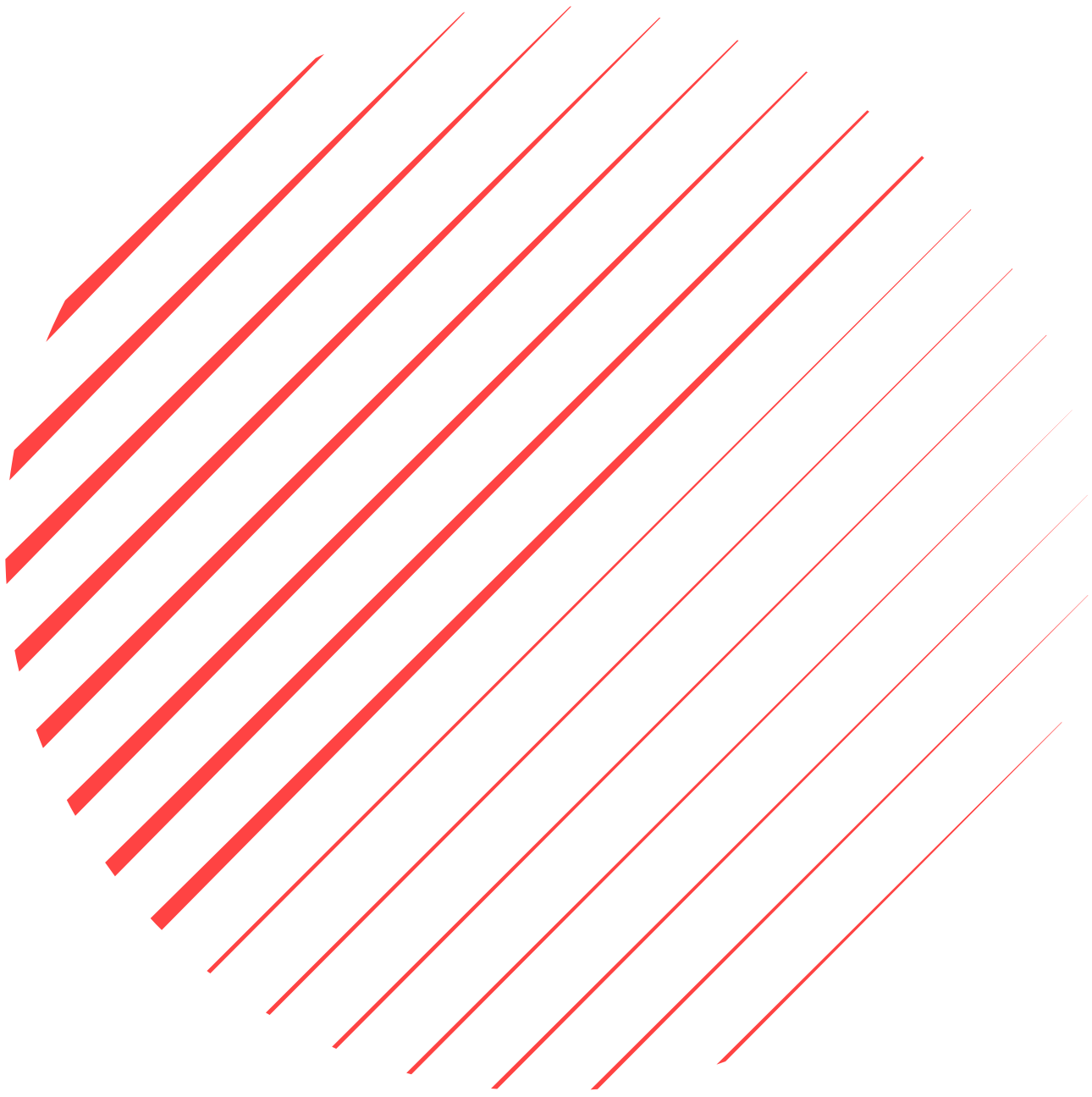
MINI PROJECT #1



FUNDAMENTAL OF INTELLIGENT SYSTEMS

Amir Mohammad Saffar
Dr. aliyarishorehdeli

Question 1

1.1

- درباره این مجموعه داده به صورت خلاصه توضیح دهید.
- ویژگی‌های [۲] موجود در این مجموعه داده را نام ببرید.
- چه تعداد نمونه [۳] در این مجموعه داده موجود است؟

- این مجموعه داده برای یکی از شرکت های اعتباری است که با از درست دادن مشتری هایش احساس نارضایتی به آن ها دست داده و میخواهند با استفاده از اطلاعاتی که از مشتری های خود در اختیار دارند، پیش بینی کنند که کدام یکی از آنها مایل به عدم ادامه همکاری خواهد بود تا با خدمات بهتر به سراغ آنها بروند. در این دیتاست، **23** ویژگی وجود دارد که از به صورت زیر هستند،

Feature	Explain
CLIENTNUM	شماره مشتری
Attrition_Flag	میزان رضایت
Customer_Age	سن مشتری
Gender	جنسیت
Dependent_count	افراد تحت تکفل فرد
Education_Level	میزان تحصیلات
Marital_Status	وضعیت تاهل
Income_Category	درامد سالیانه
Card_Category	درجه کارت
Months_on_book	مدت زمان گذشته از افتتاح حساب
Total_Relationship_Count	تعداد کل وابستگی های مشتری به موسسه
Months_Inactive_12_mon	ماه های غیر فعال در12 ماه گذشته
Contacts_Count_12_mon	تعداد مراجعات (آنلاین_ تلفنی_حضوری)
Credit_Limit	سقف اعتبار
Total_Revolving_Bal	مبلغ کل بدهی
Avg_Open_To_Buy	میانگین اعتبار موجود برای خرید
Total_Amt_Chng_Q4_Q1	تغییر کل مبلغ تراکنش‌ها از فصل چهارم سال به فصل اول
Total_Trans_Amt	مجموع مبلغ تراکنش‌ها
Total_Trans_Ct	تعداد کل تراکنش‌ها
Total_Ct_Chng_Q4_Q1	تغییر تعداد کل تراکنش‌ها از فصل چهارم سال به فصل اول
Avg_Utilization_Ratio	نسبت میانگین استفاده از اعتبار
Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_1	یک مدل یادگیری ماشین ساده و مؤثر برای طبقه‌بندی است که بر اساس قضیه بیز عمل می‌کند.

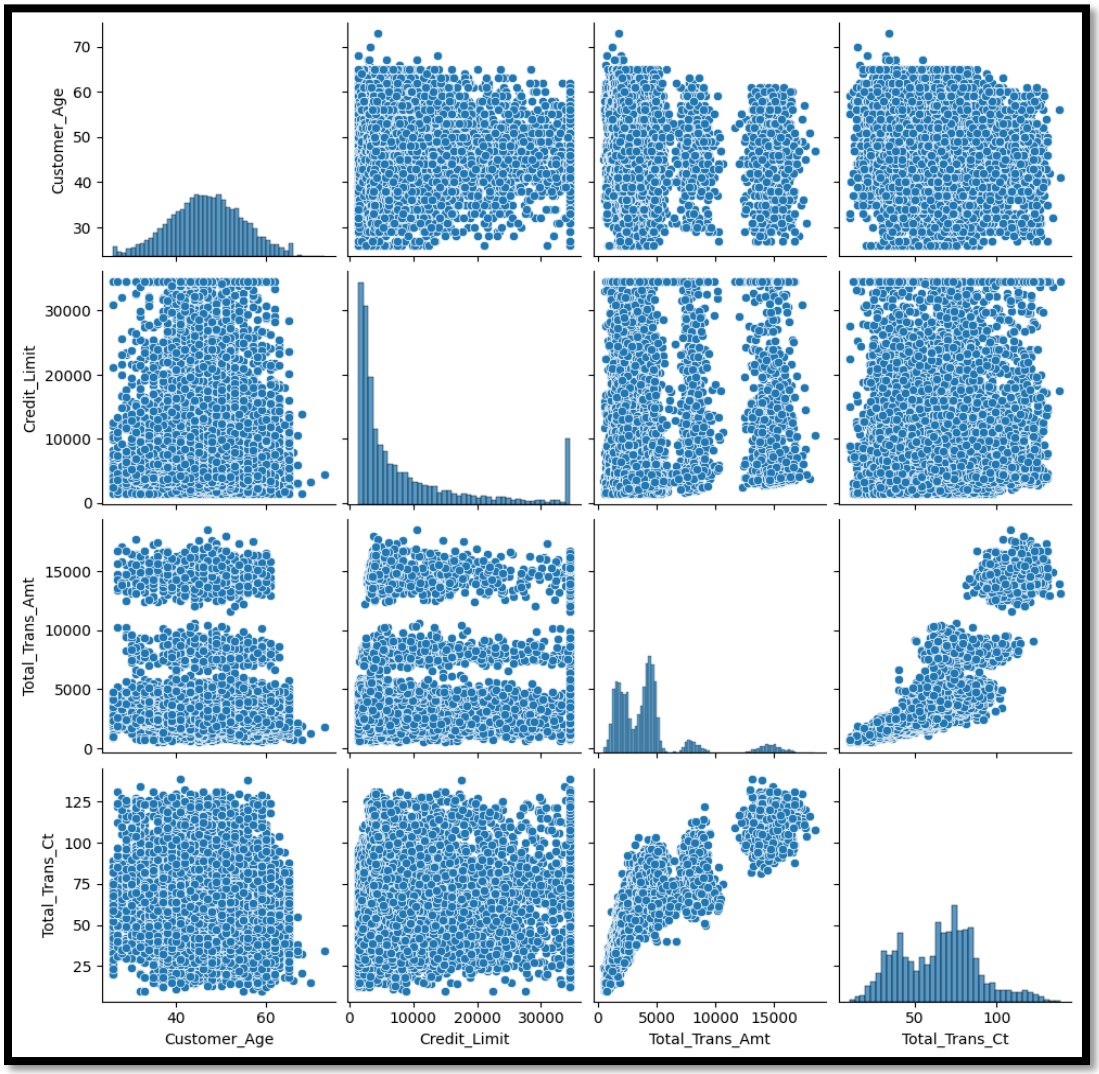
این مدل فرض می‌کند که هر ویژگی مستقل از دیگری است و با ترکیب این احتمالات، نتیجه نهایی را برای هر مشتری مشخص می‌کند.	Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_2

در این دیتاست ما 10127 تا نمونه از ویژگی ها داریم.

2.1

با استفاده از تابع `sns.pairplot` پخش [۴] داده را نمایش دهید. (در صورت زیاد بودن تعداد ویژگی‌ها، به دلخواه چهار یا پنج ویژگی را انتخاب کرده و پخش آن‌ها را نمایش دهید)

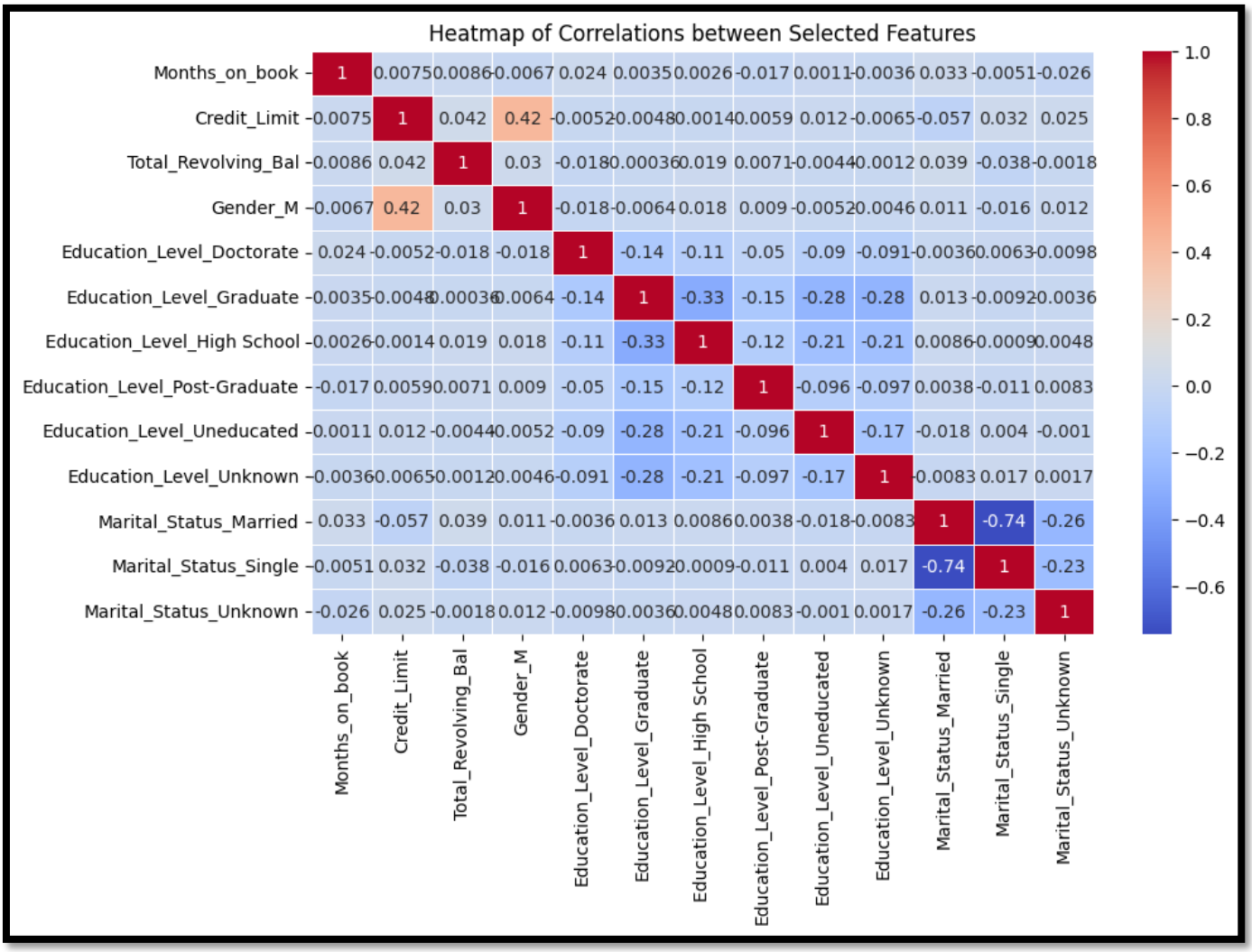
- همانطور که مشهاده می کنید 4 ویژگی انتخاب شده اند و با استفاده از دستور فوق، نمودار های زیر قابل نمایش هستند که می توان وابستگی بین ویژگی ها را از آنها استخراج کرد!



3.1

همبستگی [5] موجود میان ویژگی‌های مختلف را به صورت نقشه حرارتی [6] نشان دهید. (برای حداقل دو ویژگی طبقه‌بندی‌شده [7] و دو ویژگی پیوسته [8])

برای رسم نقشه حرارتی همبستگی بین این ویژگی‌ها، باید برخی از ویژگی‌های غیر عددی Gender و Education Level و Marital Status به اعداد تبدیل شوند. این کار به کمک روشی مانند One-Hot Encoding انجام می‌شود.



4.1

آیا در میان داده‌های موجود، داده Nan وجود دارد؟ در صورت وجود Nan در هر یک از نمونه‌ها، آن را حذف کنید.

```
Checking for 'unknown' values in the dataset:

Number of NaN values in each column after replacing 'Unknown':
CLIENTNUM                                                                                                 0
Attrition_Flag                                                                                            0
Customer_Age                                                                                              0
Gender                                                                                                    0
Dependent_count                                                                                           0
Education_Level                                                                                        1519
Marital_Status                                                                                          749
Income_Category                                                                                        1112
Card_Category                                                                                             0
Months_on_book                                                                                            0
Total_Relationship_Count                                                                                  0
Months_Inactive_12_mon                                                                                    0
Contacts_Count_12_mon                                                                                     0
Credit_Limit                                                                                              0
Total_Revolving_Bal                                                                                       0
Avg_Open_To_Buy                                                                                           0
Total_Amt_Chng_Q4_Q1                                                                                      0
Total_Trans_Amt                                                                                           0
Total_Trans_Ct                                                                                            0
Total_Ct_Chng_Q4_Q1                                                                                       0
Avg_Utilization_Ratio                                                                                     0
Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_1    0
Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_2    0
dtype: int64

Dropping rows with NaN values...
```

- در دیتاست داده در بعضی از درایه ها unknown وجود دارد که ردیف مربوط به آن ها را حذف می کنیم.

5.1

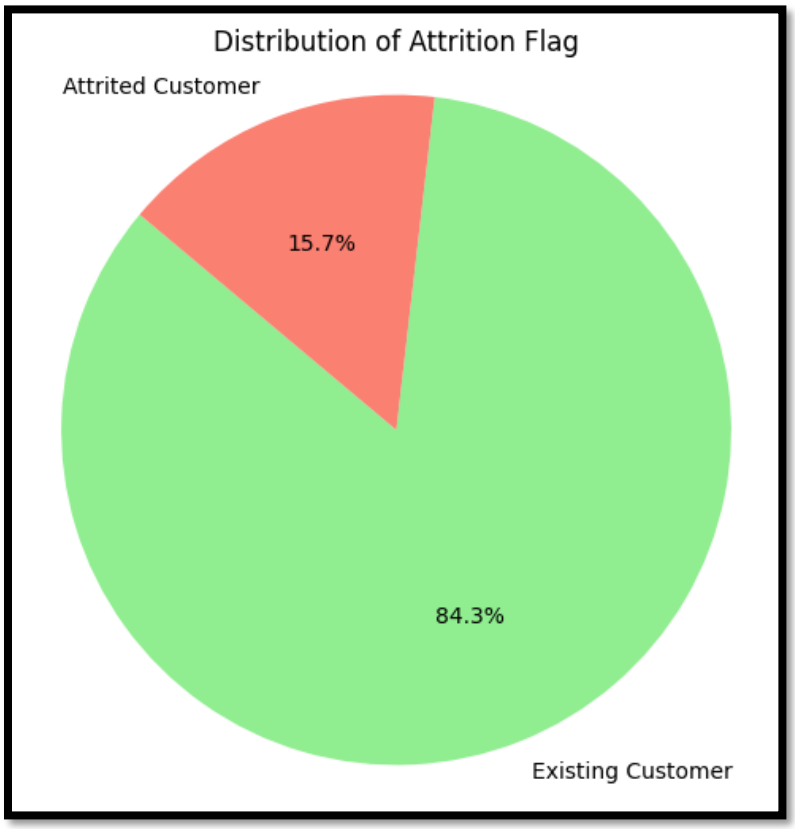
ویژگی Atrrition Flag دارای چند کلاس است.

- نام کلاس‌های موجود در این ویژگی چیست؟
- پخش داده موجود در این ویژگی را به صورت یک pie plot نمایش دهید.
- ویژگی Atrrition Flag که می‌خواهیم مدلی برای پیش‌بینی آن بسازیم، دارای عدم تعادل [۹] است. تحقیق کنید که آیا این عدم تعادل در عملکرد مدل نهایی تأثیر دارد یا نه. توضیح دهید.
- چه راهکارهایی برای اصلاح این مشکل وجود دارد؟ تحقیق کنید.
- اگر بخواهیم از یک الگوریتم برای متعادل کردن مجموعه داده‌استفاده کنیم، باید این کار را قبل از تقسیم‌بندی داده به بخش‌های آموزش و آزمون انجام دهیم یا پس از آن؟ توضیح دهید.

- دارای دو کلاس می باشد:

Existing Customer	علاقه مند به ادامه همکاری
Attrited Customer	عدم علاقه به ادامه همکاری

- خروجی کد رسم نمودار دایره ای به فرم زیر است:



- عدم تعادل داده‌ها تأثیر چشمگیری بر عملکرد مدل‌های یادگیری ماشین دارد. این عدم تعادل باعث می شود که مدل تمایل زیادی به پیش‌بینی کلاس غالب داشته باشد و کلاس‌های کم‌تعداد (مانند مشتریان از دست رفته) نادیده گرفته شوند. در نتیجه، معیارهایی مثل دقت ممکن است بالا به نظر برسند، اما عملکرد مدل در واقع برای تشخیص دقیق همه کلاس‌ها مناسب نخواهد بود. معیارهای جایگزین مانند F1-Score، ROC-AUC به ارزیابی دقیق‌تر عملکرد مدل کمک می کنند، مخصوصاً در زمانی که داده‌ها نامتعادل هستند.

- اگر میخواهیم برای متعادل‌سازی مجموعه داده‌های نامتعادل کاری انجام دهیم، معمولاً این امر را باید قبل از تقسیم داده به بخش‌های آموزش و آزمون انجام دهیم. اگر مجموعه داده‌های اصلی نامتعادل باشد و سپس به بخش‌های آموزش و آزمون تقسیم شود، احتمالاً داده‌های آزمون نیز نامتعادل خواهند بود. این به این معناست که مدل در معرض داده‌های آموزش متعادل قرار می گیرد اما با داده‌های آزمون نامتعادل مورد ارزیابی قرار می گیرد که می‌تواند به نتایج غیرقابل اعتماد و ضعیف منجر شود!

1.6.1

۱. بدون متعادل کردن داده‌ها، مدل خود را آموزش دهید.

کتابخانه‌های موردنیاز:

1. Pandas
2. sklearn
3. matplotlib

انتخاب ویژگی‌ها و هدف:

- ویژگی‌های X شامل تمامی ستون‌ها به جز Attrition Flag هستند.
- Y نیز خود ویژگی Attrition Flag است که هدف مدل است.

تبدیل ویژگی‌های دسته‌ای به عددی:

- از One-Hot Encoding برای تبدیل متغیرهای دسته‌ای استفاده شده است.

تقسیم داده‌ها:

- داده‌ها به سه بخش تقسیم شده‌اند: 60٪ برای آموزش(Train)، 20٪ برای اعتبارسنجی(Validation)، و 20٪ برای آزمون(Test).

آموزش مدل:

- از مدل **Logistic Regression** به‌عنوان یک مدل طبقه‌بندی استفاده شده است.

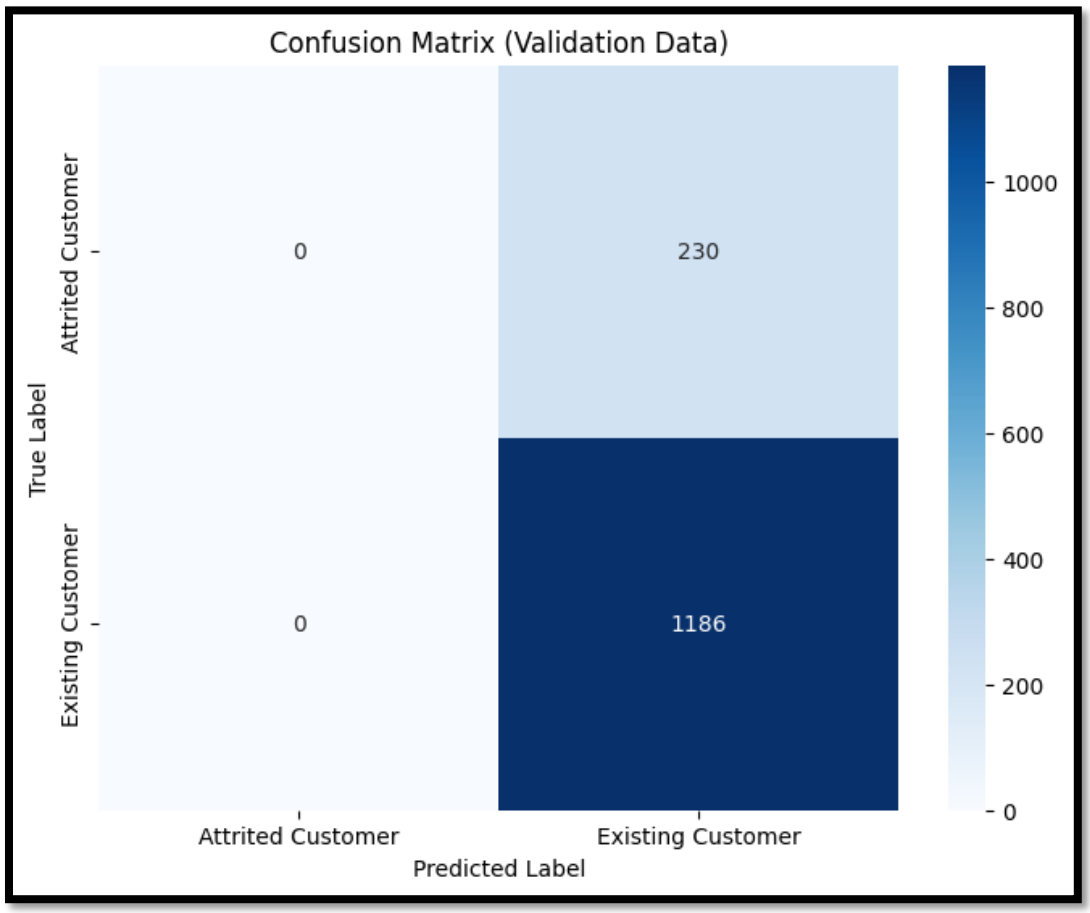
ارزیابی مدل روی داده‌های اعتبارسنجی:

- مدل روی داده‌های اعتبارسنجی ارزیابی می‌شود و گزارش طبقه‌بندی شامل معیارهای Precision ، Recall، و F1-Score است و همچنین ماتریس درهم‌ریختگی نمایش داده می‌شود.

ارزیابی نهایی روی داده‌های آزمون:

- مدل روی داده‌های آزمون نیز ارزیابی می‌شود و گزارش طبقه‌بندی و ماتریس درهم‌ریختگی برای داده‌های آزمون نیز نمایش داده می‌شود.

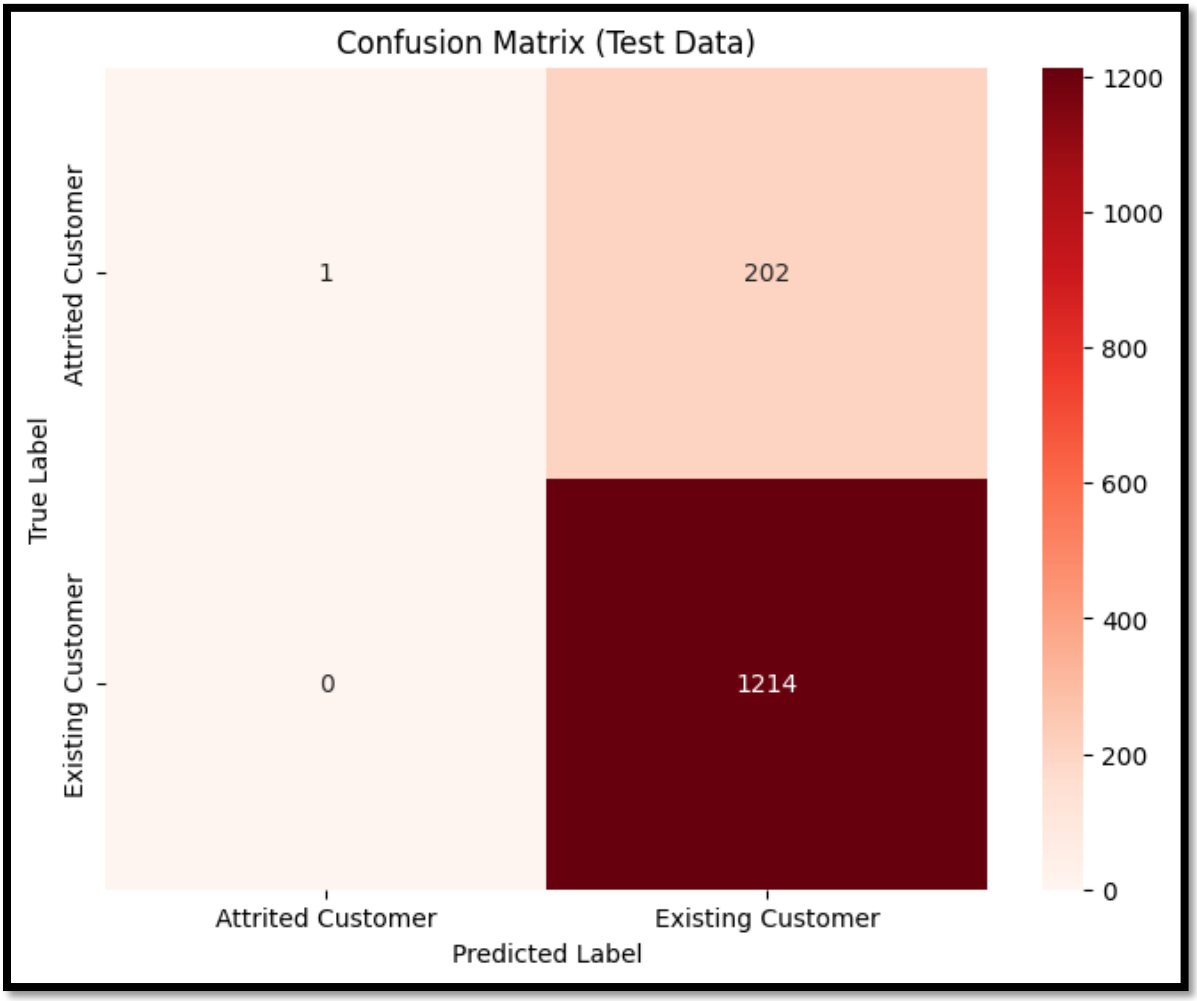
```
Classification Report (Validation Data):
                   precision    recall  f1-score   support

Attrited Customer       0.00      0.00      0.00       230
Existing Customer       0.84      1.00      0.91      1186

         accuracy                           0.84      1416
        macro avg       0.42      0.50      0.46      1416
     weighted avg       0.70      0.84      0.76      1416
```



- همین طور که مشاهده می کنید در بخش اعتبار سنجی هیچ یک از مشتری های ناراضی درست پیش بینی نشده ولی به دلیل کم بودن آنها نسبت به افراد رضایت مند مشاهده می شود که دقت 0.84 گزارش شده است.

Classification Report (Test Data):

	precision	recall	f1-score	support
Attrited Customer	1.00	0.00	0.01	203
Existing Customer	0.86	1.00	0.92	1214
accuracy			0.86	1417
macro avg	0.93	0.50	0.47	1417
weighted avg	0.88	0.86	0.79	1417



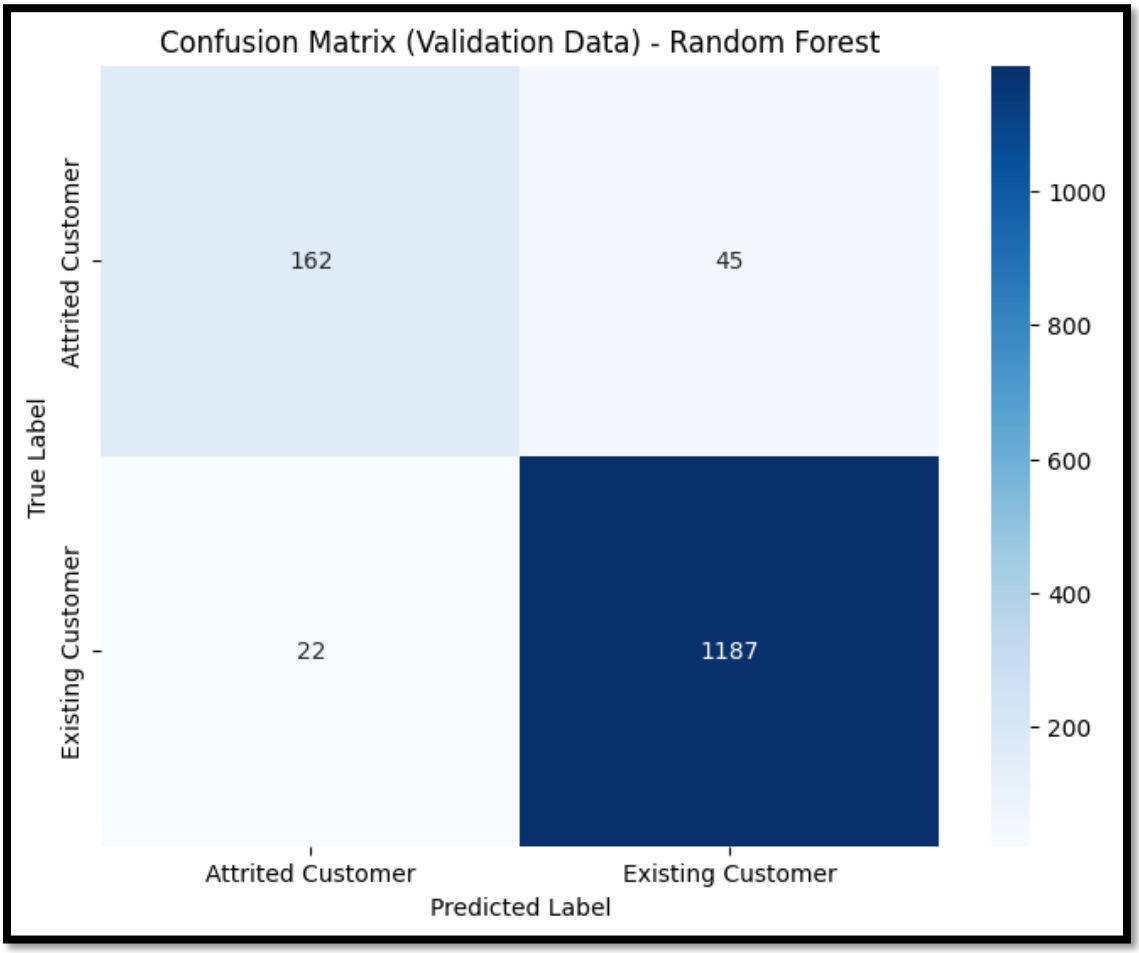
- همین طور که مشاهده می کنید در بخش آزمون، هیچ یک از مشتری های ناراضی درست پیش بینی نشده اند به غیر از یکی! ولی به دلیل کم بودن آنها نسبت به افراد رضایت مند مشاهده می شود که دقت 0.86 گزارش شده است.

✓ دلیل اصلی این اتفاق عدم متعادل بودن دسته بندی ها می باشد که باید با استفاده از الگوریتم های متلوب این کار را انجام دهیم.

حال با همین دیتاست و قبل از متعادل سازی یک بار دیگر با استفاده از Random Forest بار دیگر آموزش می دهیم!

Classification Report (Validation Data):

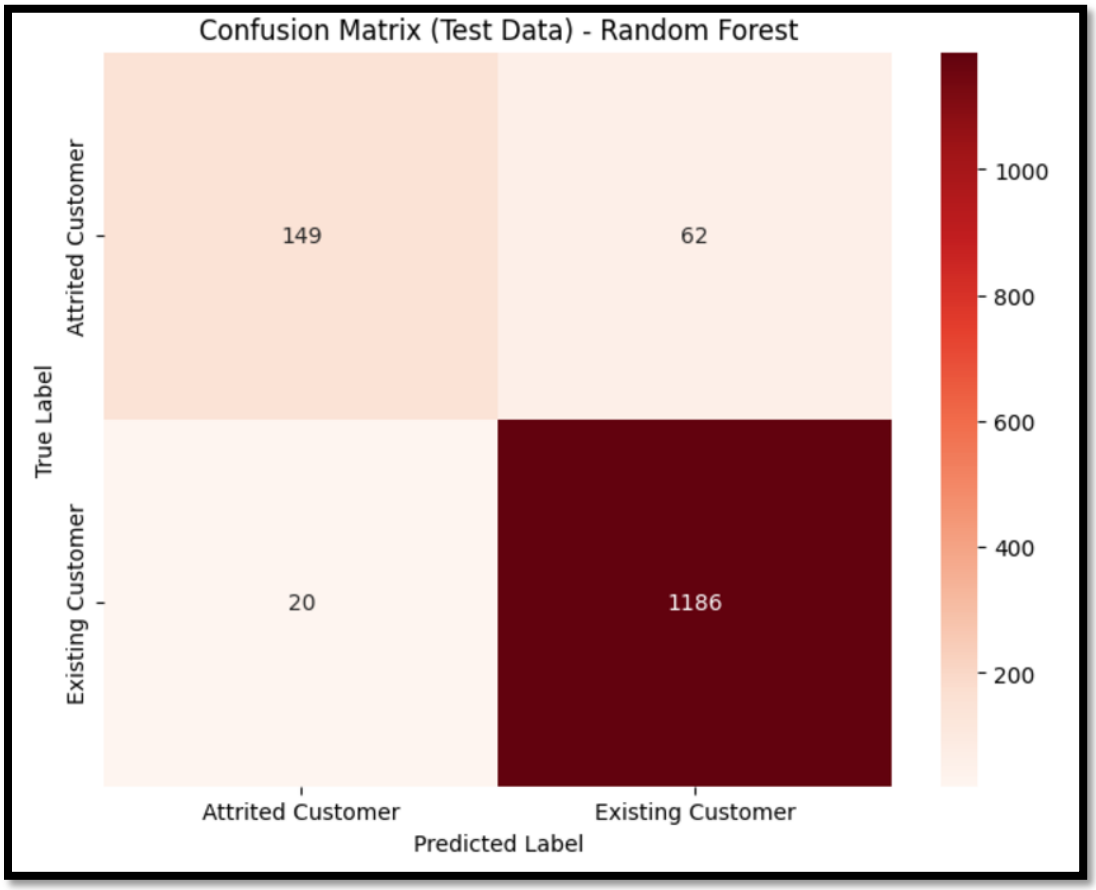
	precision	recall	f1-score	support
Attrited Customer	0.88	0.78	0.83	207
Existing Customer	0.96	0.98	0.97	1209
accuracy			0.95	1416
macro avg	0.92	0.88	0.90	1416
weighted avg	0.95	0.95	0.95	1416



- همین طور که مشاهده می کنید در بخش اعتبار سنجی 45 تا از مشتری های ناراضی درست پیش بینی نشده و فقط 22 تا از مشتری های رضایت مند به اشتباه پیش بینی شده اند. و دقت در این بخش به 0.95 رسیده است.

Classification Report (Test Data):

	precision	recall	f1-score	support
Attrited Customer	0.88	0.71	0.78	211
Existing Customer	0.95	0.98	0.97	1206
accuracy			0.94	1417
macro avg	0.92	0.84	0.88	1417
weighted avg	0.94	0.94	0.94	1417



- همین طور که مشاهده می کنید در بخش آزمون، 62 تا از مشتری های ناراضی درست پیش بینی نشده اند و فقط 20 تا از افراد رضایت مند به اشتباه ناراضی پیش بینی شده اند که بسایر عدد مناسبی است و همچنین مشاهده می شود که دقت 0.94 گزارش شده است. که عدد مناسبی می باشد.

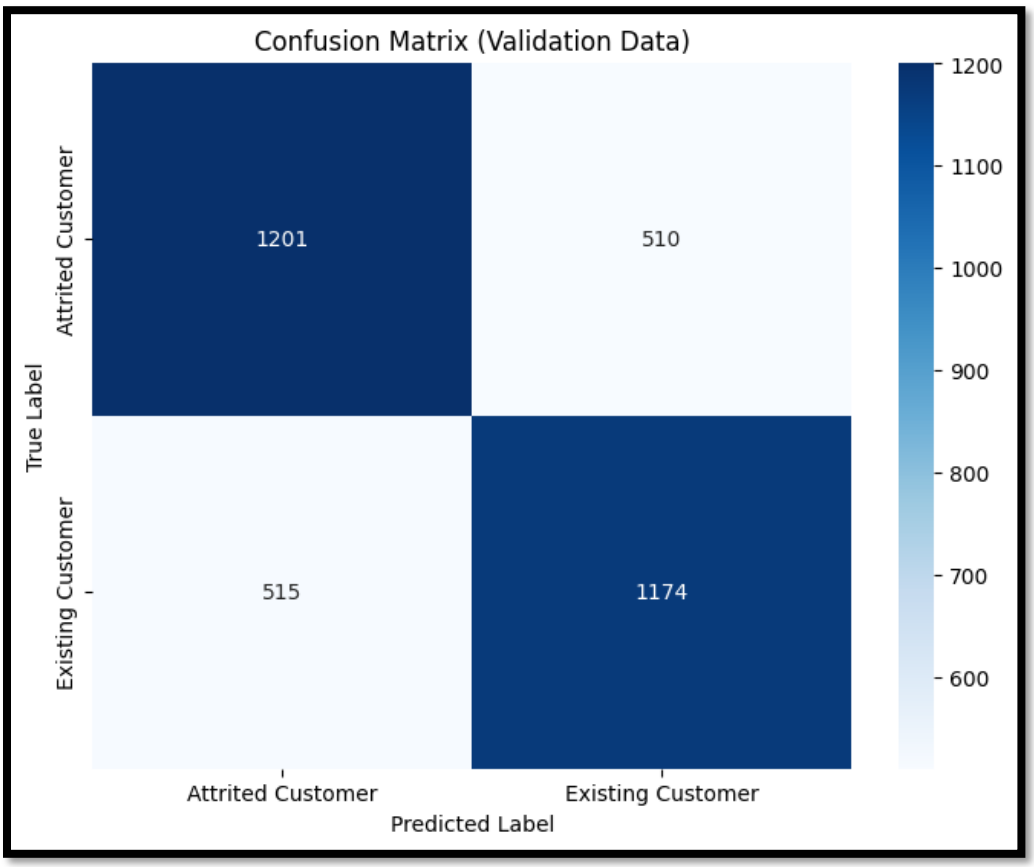
2.6.1

۲. یک الگوریتم متعادل سازی مجموعه داده را معرفی کرده و پس از متعادل کردن داده، مدل خود را آموزش دهید. (راهنمایی: اگر داده خود را متعادل کردید و مدل تنها یک کلاس را پیش‌بینی می‌کرد، بعد از متعادل کردن داده، آن را بُر بزنید [۱۳] .)

- در این بخش با استفاده از دستور (*SMOTE*) شروع به متعادل سازی سیستم میکنیم و سپس با دو مدل (logistic regression & random forest) مدل خود را آموزش می دهیم.
- ابتدا با استفاده از الگوریتم logistic regression :

```
# ستون هدف و ستون ویژگی‌ها بررسی (ستون هدف است 'Attrition_Flag' فرض می‌کنیم)
print(df['Attrition_Flag'].value_counts())

X = df.drop(columns=['Attrition_Flag'])
y = df['Attrition_Flag']

X = pd.get_dummies(X, drop_first=True)

smote = SMOTE(random_state=83)
X_resampled, y_resampled = smote.fit_resample(X, y)

X_train, X_temp, y_train, y_temp = train_test_split(X_resampled, y_resampled, test_size=0.4, random_state=83)
X_val, X_test, y_val, y_test = train_test_split(X_temp, y_temp, test_size=0.5, random_state=83)

model = LogisticRegression(max_iter=1000, random_state=83)
model.fit(X_train, y_train)

y_val_pred = model.predict(X_val)

print("Classification Report (Validation Data):")
print(classification_report(y_val, y_val_pred))
```

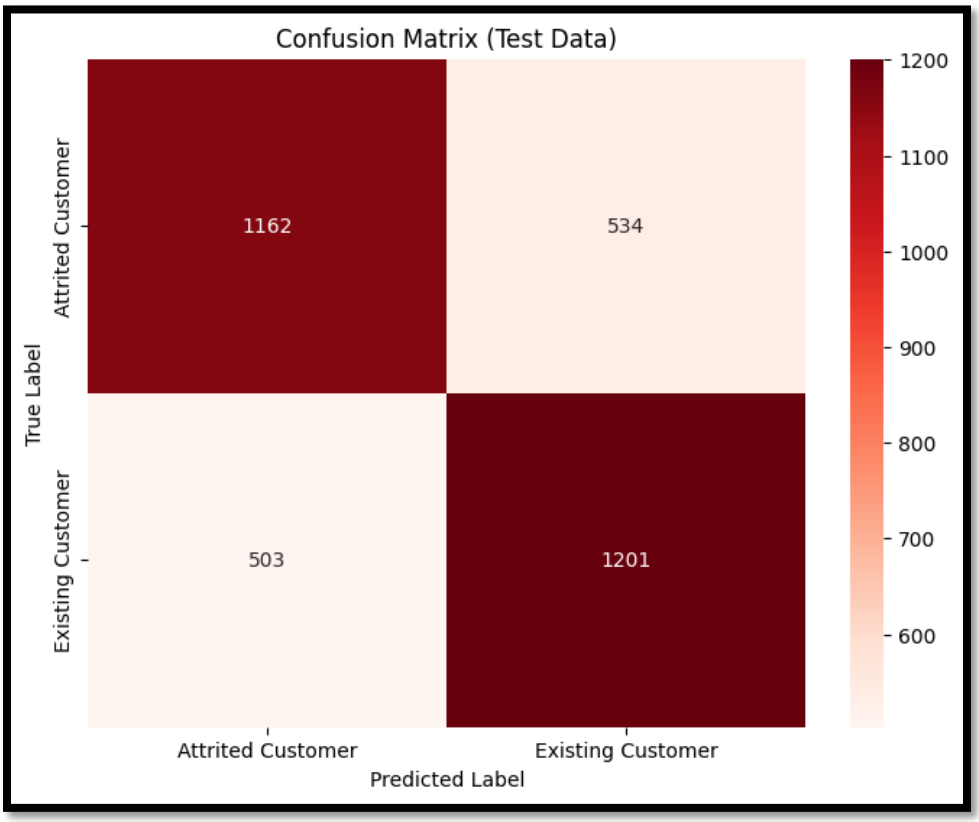
```
Classification Report (Validation Data):
                    precision    recall  f1-score   support

Attrited Customer       0.70      0.70      0.70      1711
Existing Customer       0.70      0.70      0.70      1689

         accuracy                           0.70      3400
        macro avg       0.70      0.70      0.70      3400
     weighted avg       0.70      0.70      0.70      3400
```



- همین طور که مشاهده می کنید در بخش اعتبار سنجی 510 تا از مشتری های ناراضی درست پیش بینی نشده و 515 تا از مشتری های رضایت مند به اشتباه پیش بینی شده اند. و دقت در این بخش به 0.7 رسیده است.

Classification Report (Test Data):

	precision	recall	f1-score	support
Attrited Customer	0.70	0.69	0.69	1696
Existing Customer	0.69	0.70	0.70	1704
accuracy			0.69	3400
macro avg	0.70	0.69	0.69	3400
weighted avg	0.70	0.69	0.69	3400

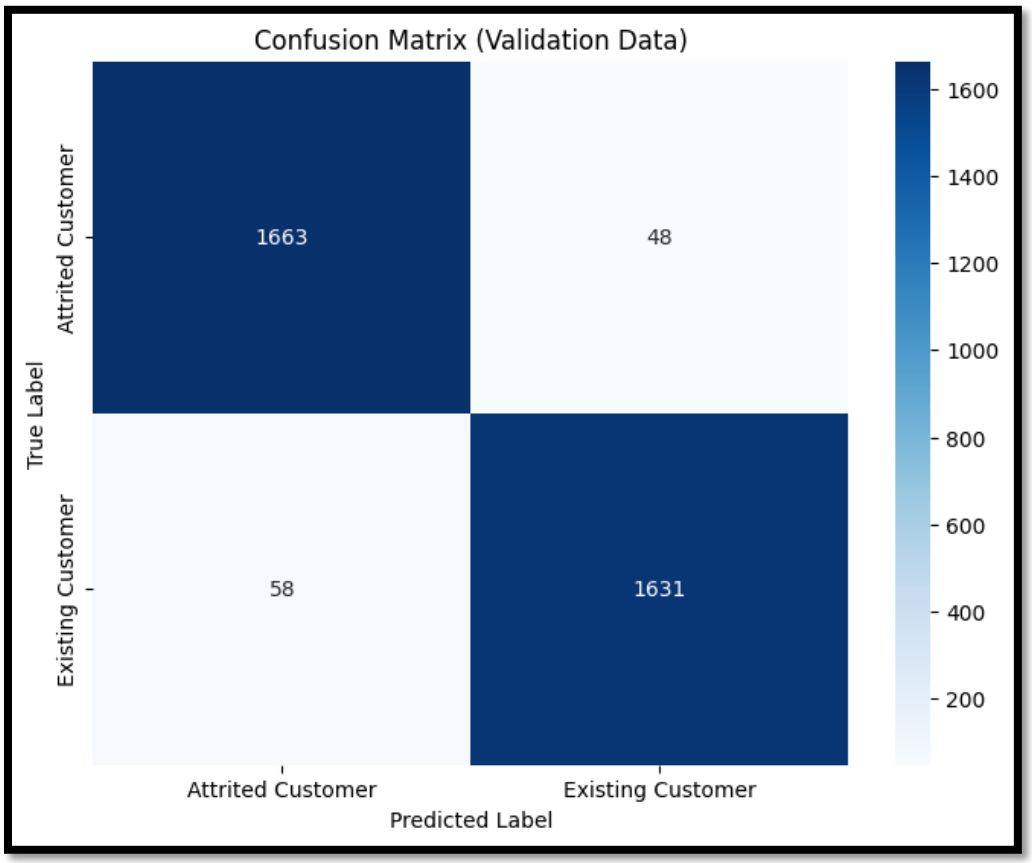


- همین طور که مشاهده می کنید در بخش آزمون، **534** تا از مشتری های ناراضی درست پیش بینی نشده اند و فقط **503** تا از افراد رضایت مند به اشتباه ناراضی پیش بینی شده اند که بسایر عدد مناسبی است و همچنین مشاهده می شود که دقت **0.69** گزارش شده است.
- اگر به نتایج همین بخش در قبل از متعادل سازی نگاهی بیاندازیم، متوجه می شویم که مدل با صحت بیشتری داده های افراد ناراضی را پیش بینی کرده است. در روش قبلی میزان دقت برای افراد ناراضی بسیار ناچیز بوده ولی در روشی که در این بخش استفاده شده، درست است که مقداری از دقت پیش بینی افراد رضایت مند کم شده، ولی باعث شده که افراد گروه دیگر نیز بتوانند با دقت مطلوبی پیش بینی شوند.

- حال با استفاده از الگوریتم Random forest :

Classification Report (Validation Data):

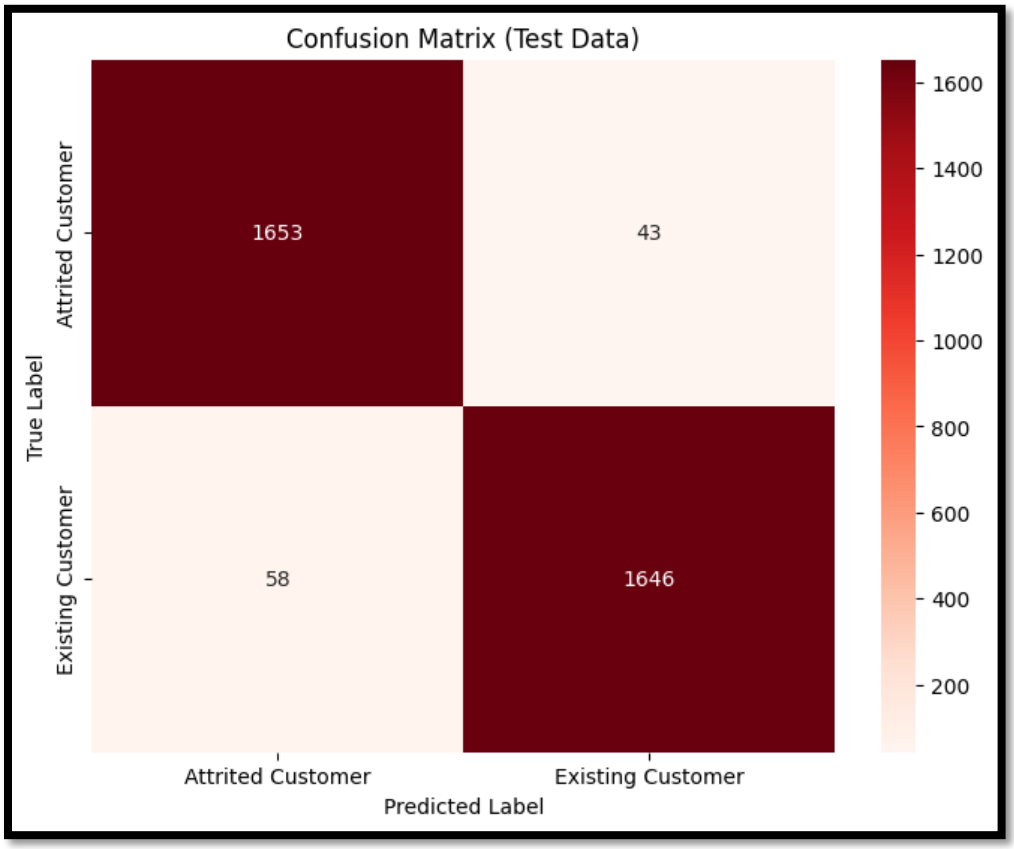
	precision	recall	f1-score	support
Attrited Customer	0.97	0.97	0.97	1711
Existing Customer	0.97	0.97	0.97	1689
accuracy			0.97	3400
macro avg	0.97	0.97	0.97	3400
weighted avg	0.97	0.97	0.97	3400



- همین طور که مشاهده می کنید در بخش اعتبار سنجی 48 تا از مشتری های ناراضی درست پیش بینی نشده و 58 تا از مشتری های رضایت مند به اشتباه پیش بینی شده اند. و دقت در این بخش به 0.97 رسیده است.

Classification Report (Test Data):

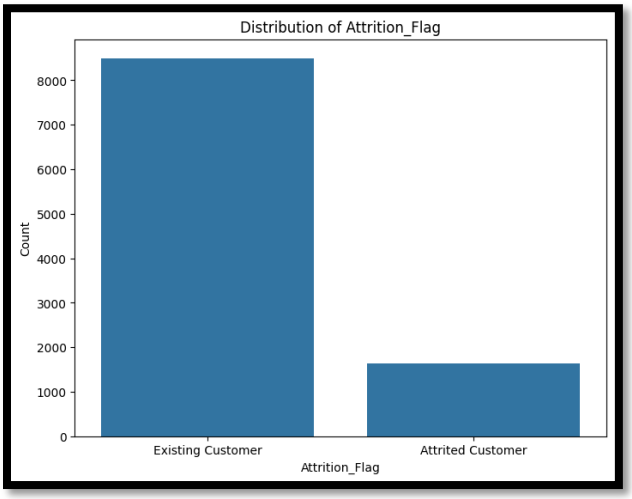
	precision	recall	f1-score	support
Attrited Customer	0.97	0.97	0.97	1696
Existing Customer	0.97	0.97	0.97	1704
accuracy			0.97	3400
macro avg	0.97	0.97	0.97	3400
weighted avg	0.97	0.97	0.97	3400

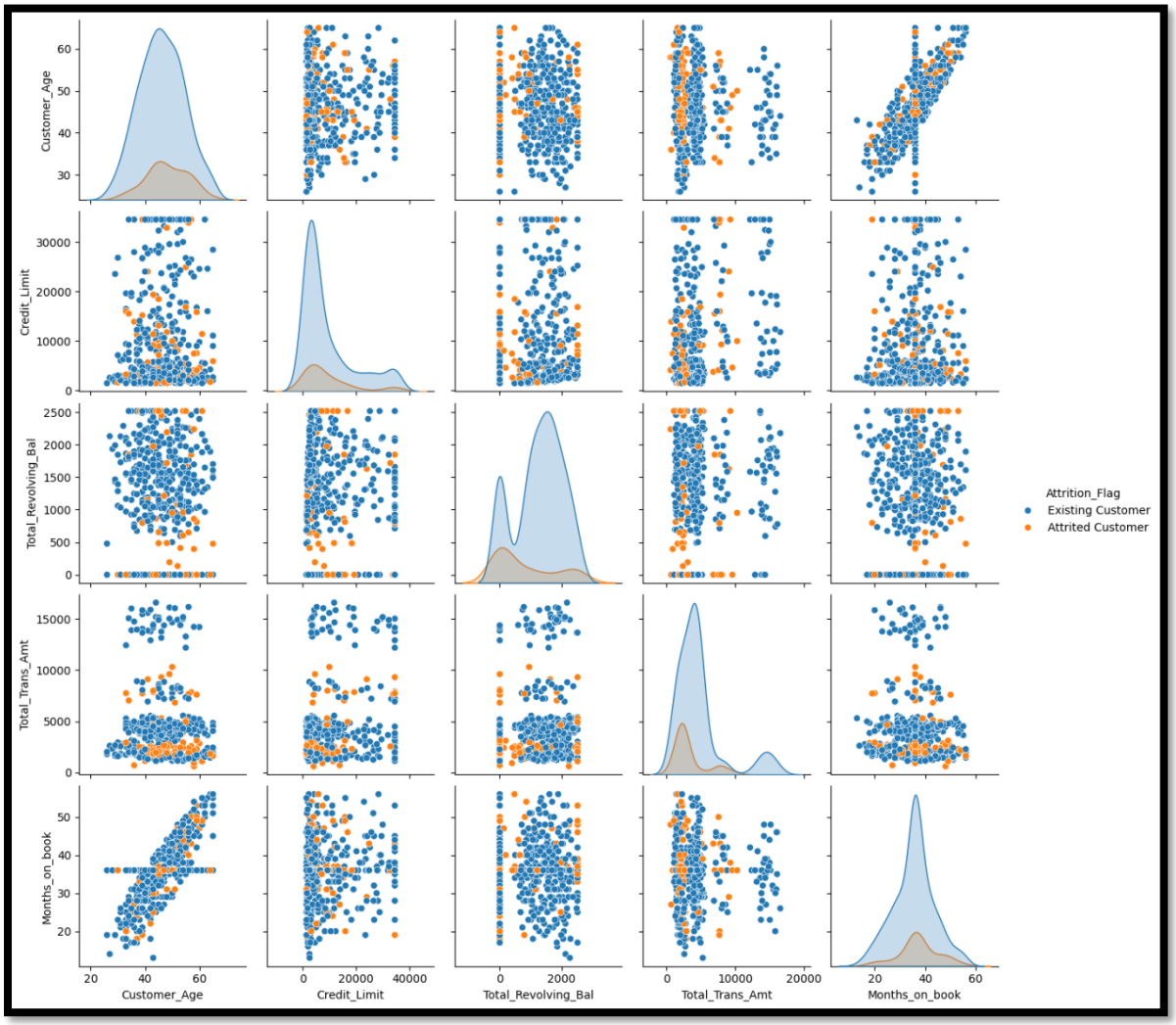


- همین طور که مشاهده می کنید در بخش آزمون، 43 تا از مشتری های ناراضی درست پیش بینی نشده اند و فقط 58 تا از افراد رضایت مند به اشتباه ناراضی پیش بینی شده اند که بسایر عدد مناسبی است و همچنین مشاهده می شود که دقت 0.97 گزارش شده است. که عدد مناسبی می باشد.
- اگر به نتایج همین بخش در قبل از متعادل سازی نگاهی بیاندازیم، متوجه می شویم که مدل با صحت بیشتری داده های افراد ناراضی را پیش بینی کرده است.

روش	دقت قبل از متعادل سازی	دقت بعد از متعادل سازی
Logistic Regression	0.86	0.69
Random forest	0.94	0.97

بخش امتیازی سوال یک





Question 2

1.2

مجموعه داده را به بخش‌های آموزش و آزمون تقسیم کنید و داده مربوط به هر یک از مجموعه داده ها را بر روی یک نمودار نمایش دهید. مشخص کنید که کدام داده برای چه مجموعه داده‌ای است.

2.2

1. Mean Squared Error- MSE:

- میانگین قدر مطلق اختلافات بین مقادیر پیش‌بینی شده و مقادیر واقعی است که حساسیت کمتری به خطاهای بزرگ نسبت به MSE دارد و میزان خطا را به صورت خطی اندازه‌گیری می‌کند.

2. Mean Absolute Error – MAE:

- این معیار به ما کمک می‌کند تا متوجه شویم مدل به طور متوسط چقدر از پیش‌بینی‌های دقیق فاصله دارد. به عبارتی دیگر، MAE میانگین قدر مطلق تفاوت بین مقادیر واقعی و مقادیر پیش‌بینی شده را به ما نشان می‌دهد.

3. Root Mean Squared Error – RMSE:

- این معیار همان MSE است با این تفاوت که از جذر آن استفاده می‌شود تا واحد آن با مقدار اصلی مقادیر داده همسان شود.

3.2

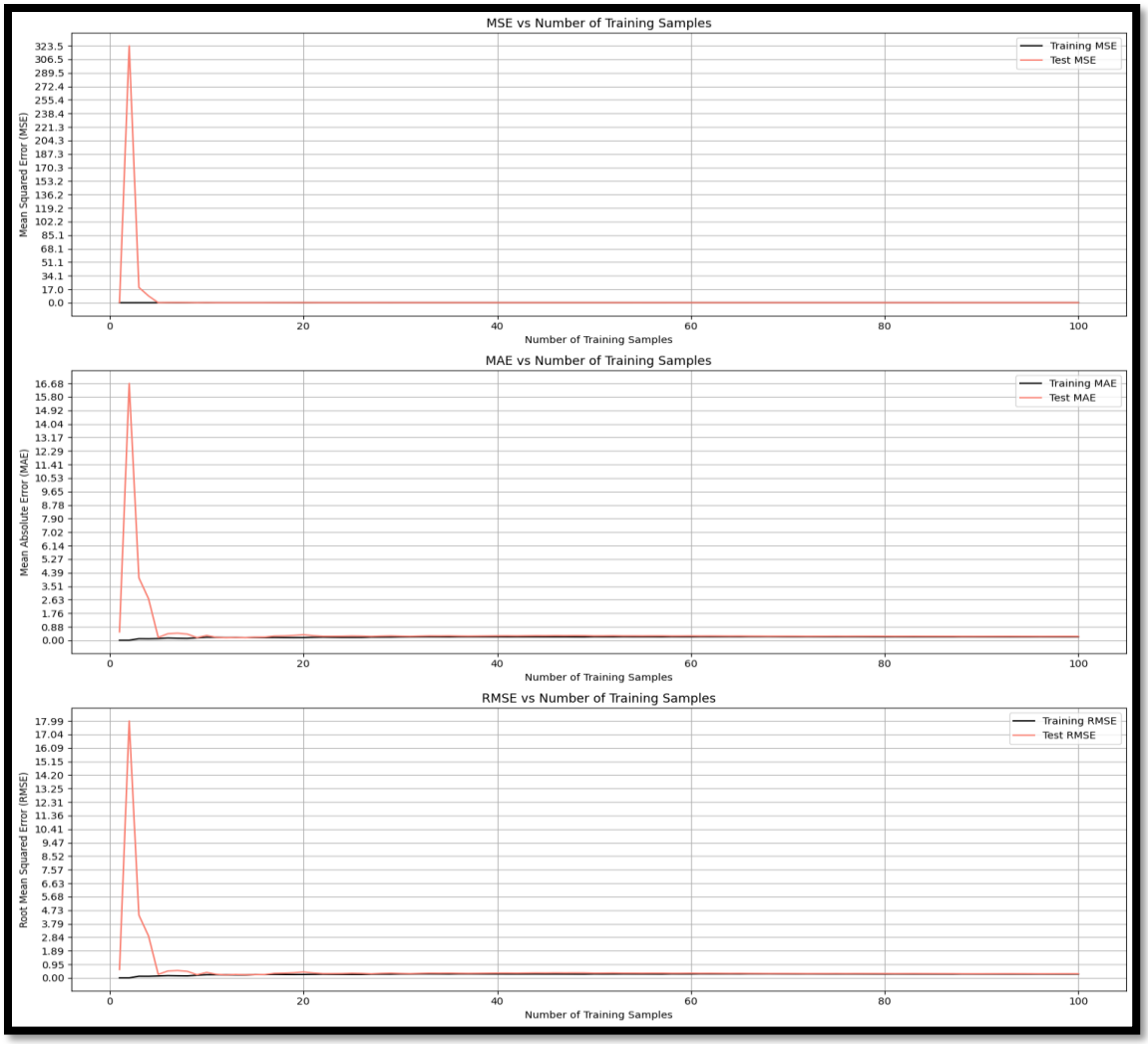
یک مدل رگرسیون خطی [۱۷] درجه اول (بدون استفاده از توابع آماده) روی داده مورد نظر آموزش دهید. به نظر شما آیا یک مدل خطی درجه اول می‌تواند به خوبی داده مورد نظر را تخمین بزند؟ توضیح دهید.

- همانطور که در نتایج مشاهده می شود مدل به خوبی نمیتواند با معادله درجه اول آموزش داده شود!

```
Mean Squared Error (MSE): 7.706647191064411e-30
Mean Absolute Error (MAE): 2.1288526497187377e-15
Root Mean Squared Error (RMSE): 2.7760848674102906e-15
```

4.2

در این بخش، تعداد دور حلقه آموزش (Iteration) را ثابت در نظر بگیرید. در ابتدا برای آموزش مدل از تنها یک داده آموزش استفاده کرده، مدل را آموزش داده و سپس مقادیر خطا برای داده آموزش و آزمون را ذخیره نمایید. در مرحله بعد یک داده به داده آموزش اضافه کرده و روند قبلی را تکرار کنید تا این که در مرحله آخر با استفاده از تمامی داده‌های آموزش مدل را آموزش دهید. نمودار خطا برای داده آزمون و آموزش را بر حسب تعداد داده آموزش رسم کنید. توضیح دهید با افزایش داده آموزش چه اتفاقی برای خطاهای آزمون و آموزش می‌افتد.



- در ابتدا، خطای آموزش نسبتاً بالا است، زیرا مدل داده‌های کافی برای یادگیری الگوهای داده را در اختیار ندارد. با افزایش تعداد داده‌های آموزشی، مدل قادر به یادگیری بهتر و شناسایی بهتر الگوهای موجود می‌شود، و در نتیجه خطای آموزش کاهش می‌یابد. به تدریج، پس از رسیدن به یک تعداد مشخص از داده‌ها، خطای آموزش به یک مقدار ثابت و حداقلی نزدیک می‌شود و دیگر به میزان قابل توجهی کاهش نمی‌یابد.

- خطای آزمون در ابتدا ممکن است بالا باشد، زیرا مدل به اندازه کافی داده برای تعمیم‌دهی به داده‌های جدید ندارد و دچار اورفیت یا آندرفیت می‌شود. با افزایش تعداد داده‌های آموزشی، مدل بهتر قادر به تعمیم‌دهی به داده‌های نادیده می‌شود، که سبب کاهش خطای آزمون می‌شود. با رسیدن به حد کافی از داده‌ها، خطای آزمون نیز به یک مقدار ثابت می‌رسد. این خطا را به دلیل وجود نویز در داده یا پیچیدگی مدل نمی‌توان کاملاً از بین برد.

5.2

با توجه به نتایج بخش قبل به سوال زیر پاسخ دهید.
برای انجام فعالیتی، خطای انسان برابر ۱ است. یک مدل یادگیری ماشین برای انجام همین فعالیت آموزش داده شده است که خطا آموزش آن برابر ۱۰ است. اگر برای آموزش این مدل از داده بیشتری استفاده کنیم، آیا می‌توانیم خطا مدل را به اندازه خطا انسان کاهش دهیم؟ توضیح دهید.

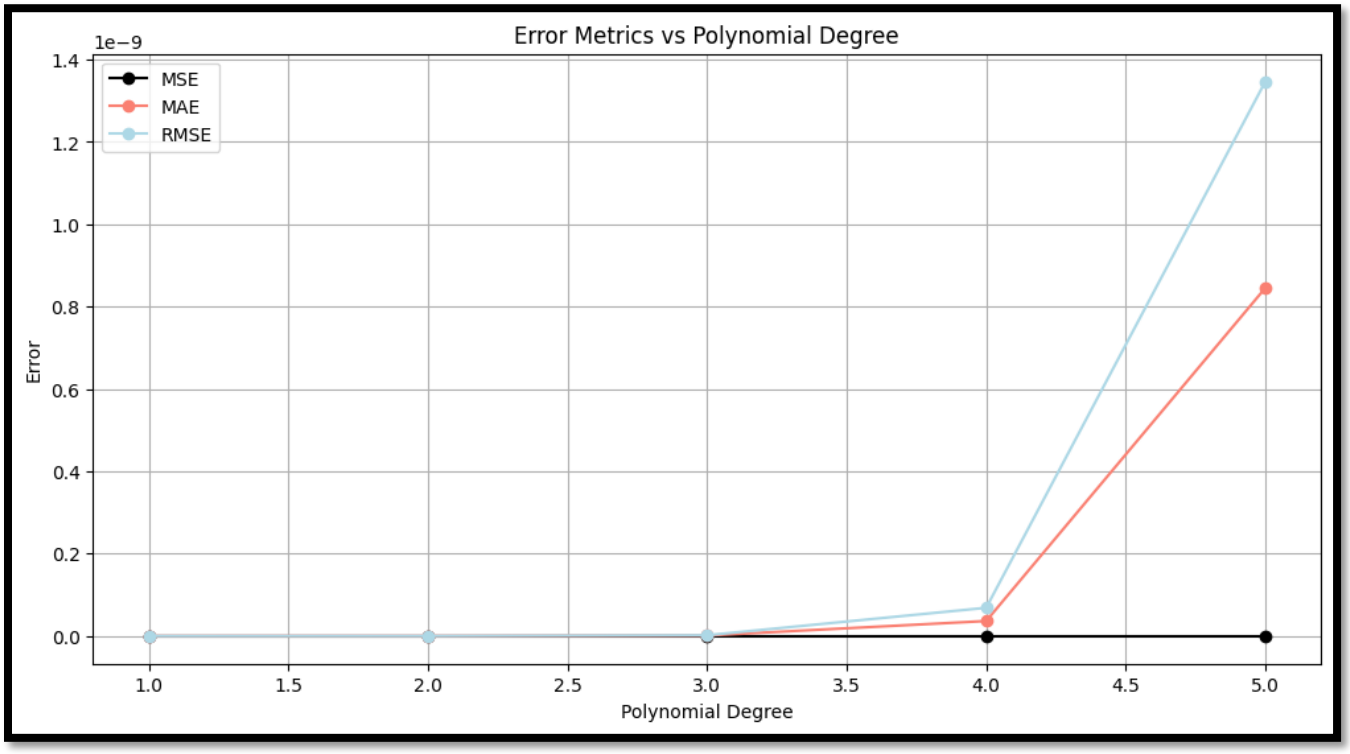
- افزایش تعداد داده‌های آموزشی به طور کلی به بهبود عملکرد مدل و کاهش خطا کمک می‌کند. با داشتن داده‌های بیشتر، مدل می‌تواند الگوها و پیچیدگی‌های بیشتری را در داده‌ها یاد بگیرد و در نتیجه دقت و تعمیم‌دهی آن افزایش یابد. بنابراین، به لحاظ تئوری می‌توان انتظار داشت که با افزایش داده‌های آموزشی، خطای مدل کاهش یابد. در کل،نکات مهمی وجود دارند که باید در نظر گرفته شوند:

I. **پتانسیل مدل:** اگر مدل مدنظر ما از نظر ساختاری به اندازه کافی پیچیده نباشد، ممکن است حتی با افزایش داده‌های آموزشی نتواند به اندازه انسان خوب عمل کند. مدل باید توانایی یادگیری الگوها و ویژگی‌های موجود در داده را داشته باشد. اگر مدل ساده ابتدایی مانند درجه اول باشد، ممکن است حتی با داده‌های زیاد، همچنان محدودیت داشته باشد.
II. **کیفیت داده‌ها:** افزایش صرفاً تعداد داده‌ها همیشه کافی نیست؛ کیفیت داده‌ها نیز اهمیت دارد. داده‌های نویزی یا غیرمرتبط می‌توانند عملکرد مدل را تحت تاثیر قرار دهند. اگر داده‌های بیشتری که اضافه می‌شوند از کیفیت بالایی برخوردار باشند، مدل می‌تواند عملکرد بهتری داشته باشد.
III. **تفاوت ماهیت مسئله برای انسان و ماشین:** گاهی اوقات، عملکرد ما انسان ها در برخی فعالیت‌ها (مثل شناسایی الگوهای پیچیده یا احساسات) ممکن است برتری خاصی نسبت به مدل‌های یادگیری ماشین داشته باشد. این موضوع به ماهیت مسئله بستگی دارد و ممکن است محدودیت‌هایی برای مدل وجود داشته باشد که حتی با داده‌های بیشتر هم نتواند کاملاً به سطح دقت انسانی برسد.

6.2

به مدل رگرسیون خطی که در بخش قبل آموزش دادید، مرحله به مرحله یک جمله با درجه دلخواه اضافه کنید. (مثلا در مرحله اول x^2 را به مدل اضافه کنید). این کار را حداقل برای ۵ جمله تکرار کنید.

- نمودار خطا بر حسب تعداد جملات چندجمله‌ای را نمایش دهید.
- آیا با افزایش تعداد جمله‌های مدل، خطای آزمون همواره کاهش میابد؟ توضیح دهید.



ابتدا با افزایش تعداد جملات، مدل بهتر الگوهای دیتا را یاد می‌گیرد و خطای آزمون کاهش می‌یابد. در ادامه اگر درجه چندجمله‌ای بیش از حد بالا برود، مدل پیچیده می‌شود و به احتمال زیاد دچار (اور فیت) می‌شود که باعث افزایش خطای آزمون می‌شود. بنابراین، با افزایش تعداد جملات چندجمله‌ای، خطای آزمون همواره کاهش نمی‌یابد؛ بلکه بعد از یک نقطه بهینه، خطای آزمون ممکن است دوباره افزایش یابد. هدف اصلی این است که مدل به اندازه کافی پیچیده باشد تا الگوهای مهم را یاد بگیرد ولی از پیچیدگی زیاد که منجر به یادگیری نویز می‌شود جلوگیری بشه!

7.2

از میان الگوریتم‌های رگرسیون موجود در کتابخانه `scikit learn`، به دلخواه ۳ الگوریتم را انتخاب کرده و به صورت خلاصه آن‌ها را توضیح دهید. سپس از این سه الگوریتم برای آموزش مدل استفاده کرده و نتایج آن‌ها را با هم مقایسه کنید.

I. Linear Regression

- رگرسیون خطی ساده‌ترین و متداول‌ترین الگوریتم رگرسیون است که تلاش می‌کند یک خط مستقیم را به داده‌ها فیت کند تا رابطه بین ویژگی‌ها و هدف را مدل‌سازی کند. این الگوریتم از روش "کمترین مربعات" برای بهینه‌سازی پارامترها استفاده می‌کند.

II. Ridge Regression

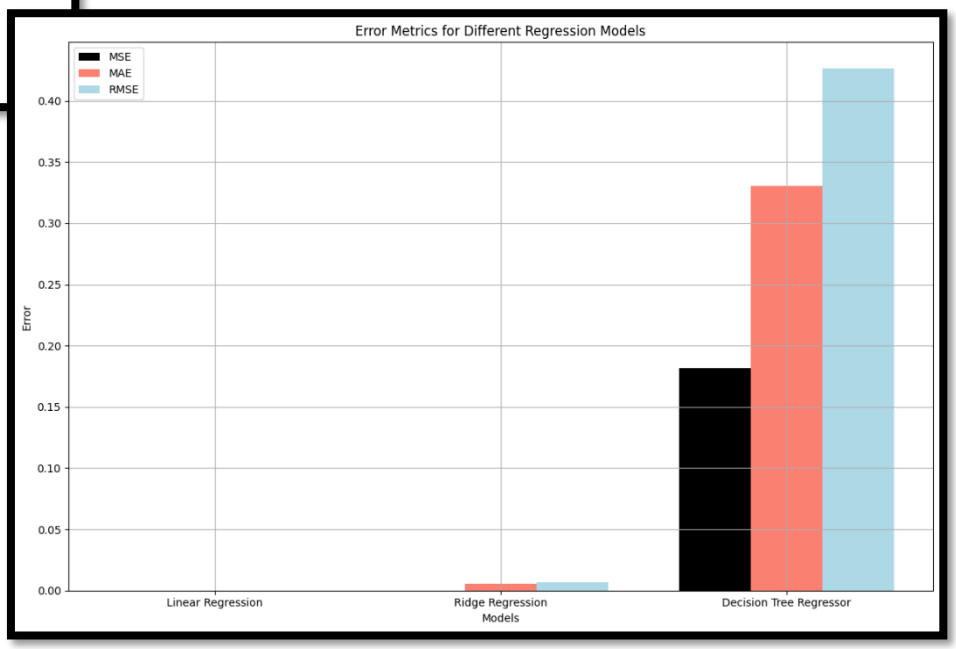
- یک نوع رگرسیون خطی است که به مدل منظم‌سازی (Regularization) اضافه می‌کند تا از اورفیت جلوگیری کند. این کار را با اضافه کردن یک جمله به تابع هزینه انجام می‌دهد که اندازه ضرایب مدل را کاهش می‌دهد. این روش برای داده‌های با هم‌خطی بالا مفید است.

III. Decision Tree Regression

- از یک ساختار درختی برای تقسیم‌بندی داده‌ها به بخش‌های مختلف و پیش‌بینی مقادیر هدف استفاده می‌کند. این الگوریتم می‌تواند به خوبی داده‌های غیرخطی و پیچیده را مدل‌سازی کند. با این حال، درخت‌های تصمیم می‌توانند مستعد اورفیت باشند مگر این که پارامترهای آن‌ها به درستی تنظیم شود.

```
Model: Linear Regression
MSE: 7.706647191064411e-30
MAE: 2.1288526497187377e-15
RMSE: 2.7760848674102906e-15

Model: Ridge Regression
MSE: 4.369701676754234e-05
MAE: 0.005316014598631097
RMSE: 0.0066103719084135

Model: Decision Tree Regressor
MSE: 0.1819249022455641
MAE: 0.3304936733866032
RMSE: 0.4265265551469968
```

- همانطور که مشاهده می شود نتایج خطاهای معین شده، در جدول و نمودار قابل بررسی می باشد.



بخش امتیازی

درباره reguralization تحقیق کنید و مدل چند جمله‌ای خود را با استفاده از reguralization دوباره آموزش دهید. (بدون استفاده از توابع آماده)

There are two main types of regularization:

1. **L1 Regularization (Lasso)**: This adds a penalty proportional to the sum of the absolute values of the coefficients, effectively shrinking some coefficients to zero, resulting in feature selection.
2. **L2 Regularization (Ridge)**: This adds a penalty proportional to the sum of the squared coefficients, limiting large coefficient values to prevent overfitting.

Also, some built-in functions like ***Ridge*** from scikit-learn could be found helpful in our projects.

- همانطور که در بخش قبلی ذکر شد، مدل دومی که استفاده شده از regularization در آن استفاده به عمل آمده است.