Data Analysis

Homework 7.

—

Ali Izadi

810199102

۱- پیش پردازش

۱- تمیزسازی داده

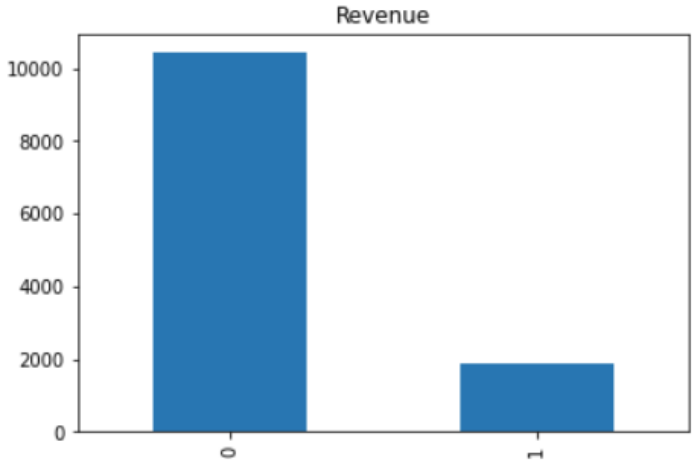
مقادیر ستون های Administrative_Duration و Informational_Duration و ProductRelated_Duration که در دیتاست صفر هستند باید با مقدار مناسب جایگزین شوند که برای پر کردن آن ها از میانه هر ستون برای پر کردن آن استفاده شده است.

ستون های categorical مختلف را میتوان به one hot encoding تبدیل کرد که به علت تعداد categoryهای مختلف دچار نحسی ابعاد در این روش میشویم و با بررسی feature importance هر کدام از این متغیرها و بررسی تعداد حالت های مختلف آن ها متوجه شدم که ترتیب هر کدام تقریبا برابر با تعداد هر یک از categoryها به صورت صعودی است و encodeکردن آن ها به صورت categorical اطلاعات بیشتری در آن ها نهفته است و یادگیری آن ها برای مدل ها آسان تر است.

۲۰ درصد داده ها به عنوان داده های تست همین ابتدا جدا شده اند تا در نهایت بهترین مدل انتخاب شده بر روی این داده ها نتایجش گزارش شود.

۲- توازن سازی کلاس ها

همان طور که در نمودار زیر مشاهده میشود تعداد کلاس ها نامتوازن هستند بنابراین دو روش استفاده از متریک مناسب و هم چنین ایجاد نمونه های جدید برای حل این مشکل در نظر گرفته شده است.



۱- استفاده از متریک مناسب

برای مقایسه مدل ها استفاده از معیار accuracy معیار مناسبی نیست زیرا داده های کلاس ها نامتوازن هستند. بنابراین برای مقایسه صحیح برتری عملکرد مدل ها از معیار f1 استفاده خواهیم کرد.

۲- ایجاد نمونه های جدید برای داده یادگیری

برای ایجاد داده های جدید از روش SMOTE استفاده خواهیم کرد.

روش smote به این صورت عمل میکند که به صورت رندوم از کلاس با تعداد کمتر نمونه انتخاب میکند سپس k نمونه نزدیک متعلق به آن کلاس را پیدا میکند و سپس یکی از این k نمونه انتخاب میشود و خطی بین نمونه اصلی و این نمونه کشیده میشود سپس نمونه های جدید بر اساس این خط و با ترکیب convex از این دو نقطه به صورت رندوم تولید میشود تا در نهایت تعداد کلاس ها برابر شوند.

۳- استاندارد سازی داده ها

برای استاندارد سازی داده ها از روش StandardScaler استفاده کرده ایم.

۲- انتخاب ویژگی

۱- نحسی ابعاد

تعداد ستون های داده ۱۷ تا است بنابراین چون تعداد ستون ها نسبتا زیاد است الگوریتم های یادگیری ممکن است از مشکل نحسی ابعاد رنج ببرند. زیرا در ابعاد بالاتر فاصله بین نقاط به درستی مشخص نیست و الگوریتم هایی که از فاصله نقاط برای یادگیری استفاده میکنند دچار مشکل میشوند و تفاوت بین داده ها به خوبی مشخص نیست.

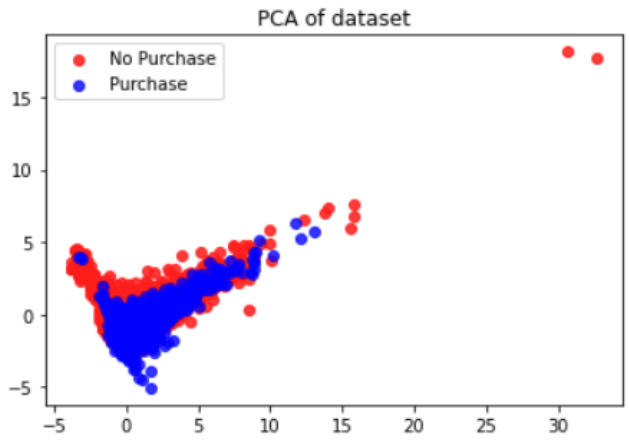
روش های کاهش بعد با تبدیلات روی ویژگی ها و کم کردن ابعاد باعث میشوند مشکل گفته شده در ابعاد بالاتر کمتر شود. گرچه این تبدیل ویژگی ها خود ممکن است اطلاعات مفیدی را از بین ببرد که در نهایت باید در عملکرد مدل ها این مورد در نظر گرفته شود که در بخش مقایسه مدل ها دو حالت کاهش بعد و بدون کاهش بعد هم مقایسه خواهد شد.

روش کاهش بعد:

برای کاهش بعد از روش PCA استفاده خواهیم کرد.

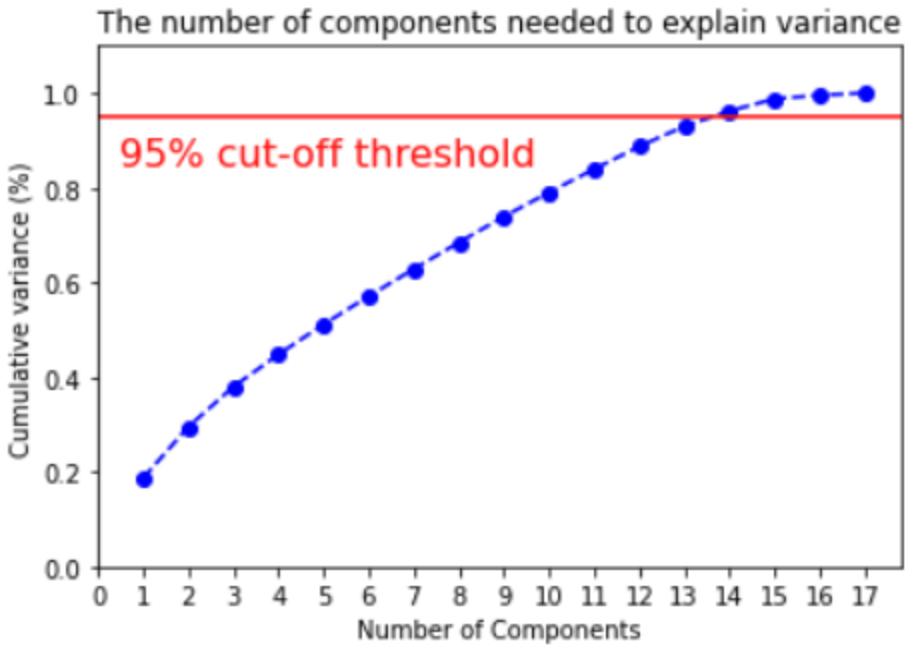
در نمودار زیر کاهش بعد با استفاده از PCA بر روی دو بعد انجام شده است.

همان طور که مشاهده میشود کاهش بعد به ۲ اطلاعات زیادی را از بین برده است و کلاس ها عملا جدا پذیر نیستند.



برای پیدا کردن بهترین بعد کاهش یافته از پارامتر خروجی pca یعنی explained variance استفاده میکنیم تا ببینیم تا چه بعدی ۹۵ درصد واریانس در داده حفظ خواهد شد.

همان طور که در شکل زیر مشاهده میشود در شکل زیر تا بعد ۱۳ نود و پنج درصد واریانس داده ها حفظ خواهد شد.

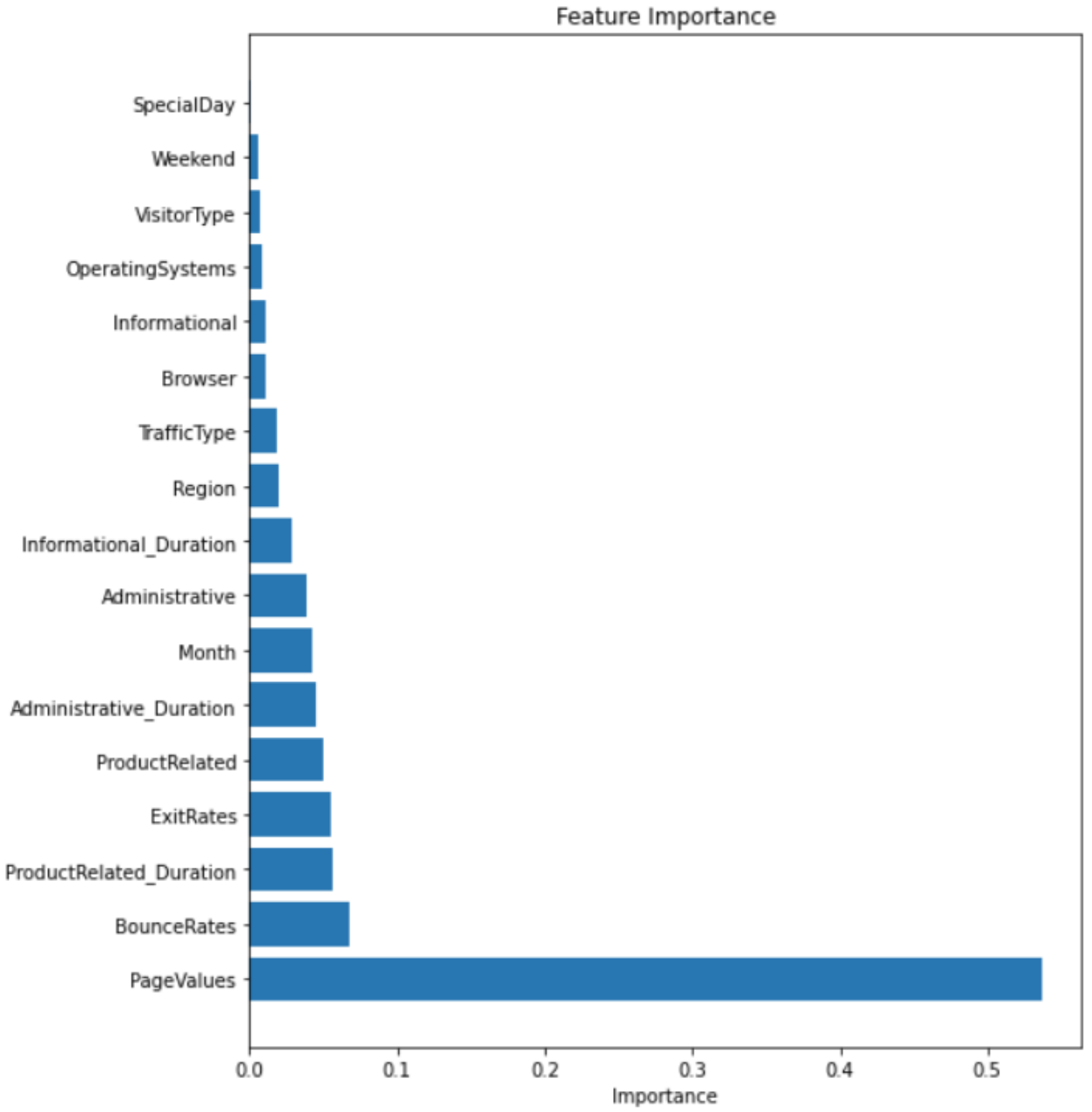


بنابراین در قسمت مقایسه مدل ها با استفاده از PCA و بدون PCA کاهش بعد را تا بعد ۱۳ انجام میدهیم.

۲- انتخاب ویژگی

روش RandomForestRegressor

در زیر ترتیب اهمین ویژگی ها بر اساس روش بالا آورده شده است. همان طور که مشاهده میشود ویژگی ای مانند Page value با اختلاف اهمیت بیشتری نسبت به سایر ویژگی ها دارد.



شش ویژگی برتر عبارتند از Page Value و BounceRates و ProductRelated_Duration و ExitRates و ProductRelated و Administrative_Duration.

با مشاهده ویژگی ها و اهمیت آن ها میبینیم که تا ویژگی نهم هم از نظر مقدار اهمیت آن ها هم از نظر اهمیت ذاتی خود در پیش بینی خرید یا نخریدن اطلاعات مفیدی در آن ها وجود دارد. بنابراین این ۹ ویژگی را انتخاب میکنیم و در ادامه در مقایسه مدل ها دو حالت تنها استفاده از این ۹ ویژگی و استفاده از همه ویژگی ها را بررسی خواهیم کرد که آیا مدل ها از نحسی ابعاد رنج میبرند و کم کردن تعداد ویژگی

ها به عملکرد مدل ها و generalization آن ها کمک خواهد کرد یا نه با همه ویژگی ها مدل ها عملکرد بهتری بر اساس متریک خواهند داشت.

۳- طبقه بندی

معیارهای انتخابی

معیارهای انتخابی برای مقایسه مدل ها علاوه بر accuracy:

۱- f1: که در مسئله با کلاس های نامتوازن کمک میکند مقایسه بهتری داشته باشیم.

۲- precision: چون برای ما مهم است که آن هایی که حتما خرید میکنند را بتوانیم همه را درست پیش بینی کنیم بنابراین این معیار را نیز در نظر میگیریم.

۳- roc auc: این معیار هم نیز جز دقیق ترین معیارها برای مدل های classification است که پیش بینی هر دو کلاس به درستی را ارزیابی میکند.

روش مقایسه مدل ها

برای مقایسه مدل ها از cross validation با تعداد ۵ fold و ۲۰ درصد داده به عنوان validation استفاده کرده ایم تا میانگین متریک ها بر روی این ۵ fold گزارش شود تا از overfit انتخاب بهترین مدل بر اساس فقط یک قسمت از داده جلوگیری کنیم.

انتخاب بهترین پارامترهای هر مدل

هر کدام از مدل های خواسته شده دارای تعدادی hyperparamter هستند که بهترین پارامتر نیز بر اساس accuracy و با روش greedy و هم چنین با استفاده از cross validation انتخاب شده است و در نهایت از بهترین پارامترهای هر مدل استفاده شده تا معیارهای مختلف انتخاب شده با cross validation برای هر مدل گزارش شود.

پارامترهای جست و جو شده برای مدل ها در زیر آورده شده است.

```
params = {'SVC': {'kernel': ['rbf'], 'C': [1]},
          'Decision Tree': {'max_depth': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]},
          'KNN': {'n_neighbors': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]},
          'MLP': {'hidden_layer_sizes': [(100,), (50,), (25,), (10,), (5,), (1,)]},
          'Logistic Regression': {'C': [0.1, 1, 10, 100, 1000]}}
```

مقایسه دقت مدل ها

در این قسمت از روش کاهش بعد یا انتخاب ویژگی ها یا افزودن داده استفاده نمیکنیم و تنها مدل ها را بر اساس بهترین پارامتر مقایسه میکنیم و در قسمت های بعدی تاثیر اضافه کردن این موارد را در عملکرد مدل ها بررسی خواهیم کرد.

در زیر نتیجه بهترین پارامترهای انتخابی بر اساس **accuracy** و هم چنین با استفاده از cross validation در زیر آورده شده است.

SVC: {'clf__C': 1, 'clf__kernel': 'rbf'}

Decision Tree: {'clf__max_depth': 4}

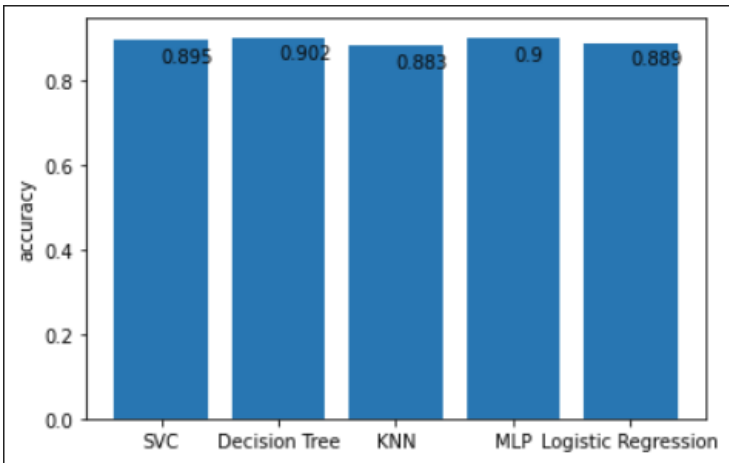
KNN: {'clf__n_neighbors': 9}

MLP: {'clf__hidden_layer_sizes': (10,)}

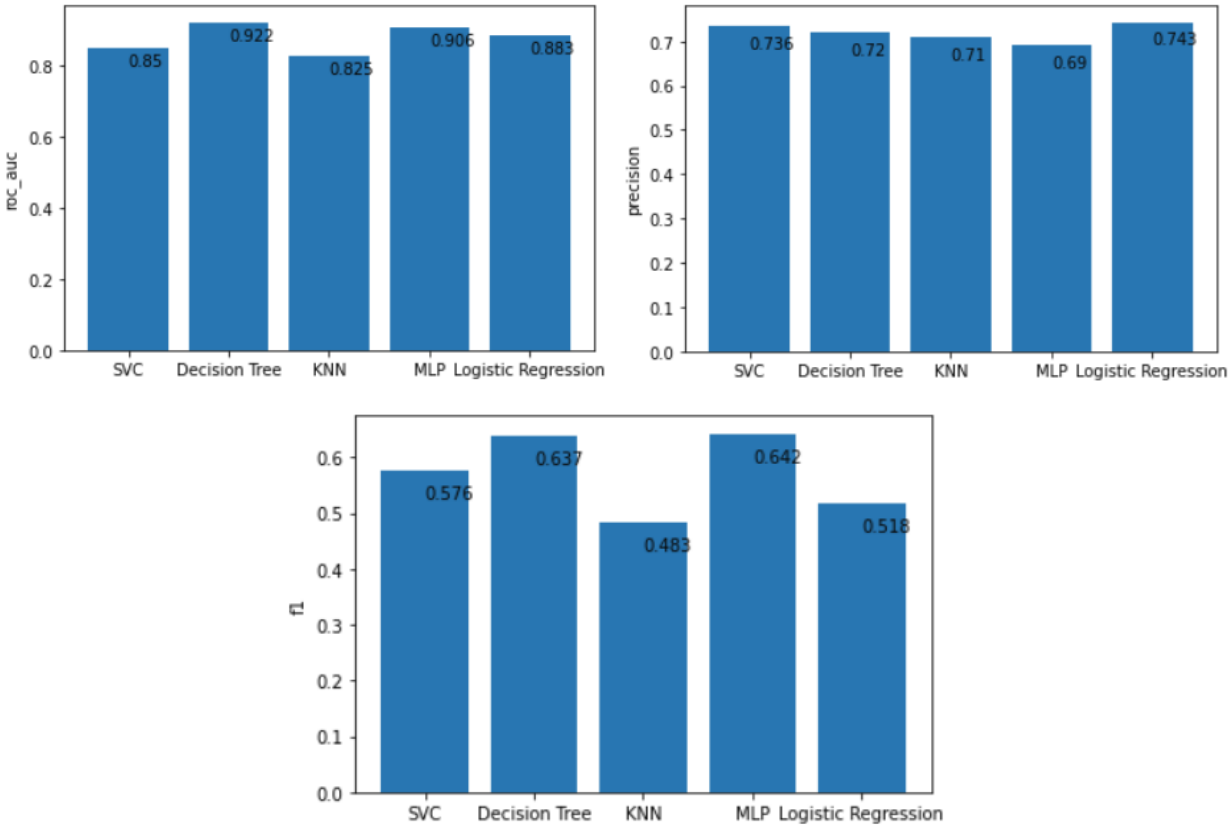
Logistic Regression: {'clf__C': 100}

بهترین پارامتر انتخابی بالا برای هر مدل استفاده شده و معیار accuracy برای این ۵ مدل با استفاده از میانگین cross validation در زیر آورده شده است.

همان طور که مشاهده میشود **accuracy** مدل ها تقریبا نزدیک به هم هستند و بهترین مدل ها MLP و Decision Tree هستند.



در زیر ۳ معیار دیگر برای مقایسه مدل ها آورده شده است. همان طور که مشاهده میشود با این که accuracy مدل ها بالاست ولی **F1** آن ها پایین است. چون کلاس ها نامتوازن هستند **بنابراین از این به بعد از معیار F1 برای مقایسه مدل ها استفاده خواهیم کرد** و هم چنین بر اساس این معیار تفاوت مدل ها بهتر مشخص شده است و MLP و Decision Tree با اختلاف قابل توجه تری از بقیه مدل ها بهتر هستند.

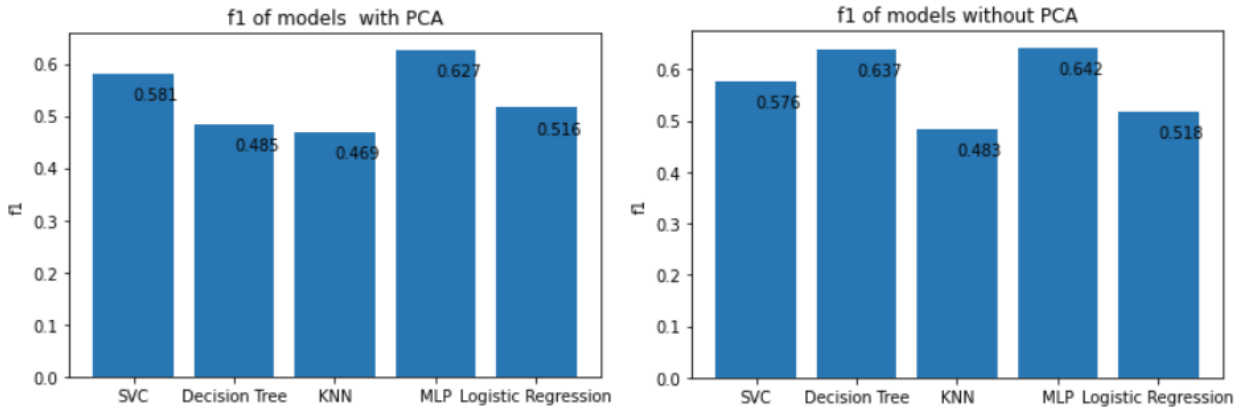


در زیر اثر اضافه کردن PCA و حذف ویژگی ها بر اساس feature importance و هم چنین توازن سازی کلاس ها با روش smote را بررسی خواهیم کرد.

توجه شد که در زیر مشابه با قبل برای ارزیابی مدل ها هم چنان سرچ بهترین پارامترها و هم چنین cross validation را با GridSearchCV انجام میدهیم تا مراحل قبل هم چنان انجام شود تا بتوان بهترین مدل ها و بررسی اضافه کردن این سه مورد را به درستی بررسی کرد و وابسته به انتخاب بهترین پارامترها در مرحله قبل نباشد.

بررسی اثر PCA

همان طور که در قسمت های قبل نتیجه گرفته شد تعداد بعدهای کاهش یافته را ۱۳ در نظر میگیریم و اثر کاهش بعد با PCA را بر روی معیار F1 را بررسی خواهیم کرد.



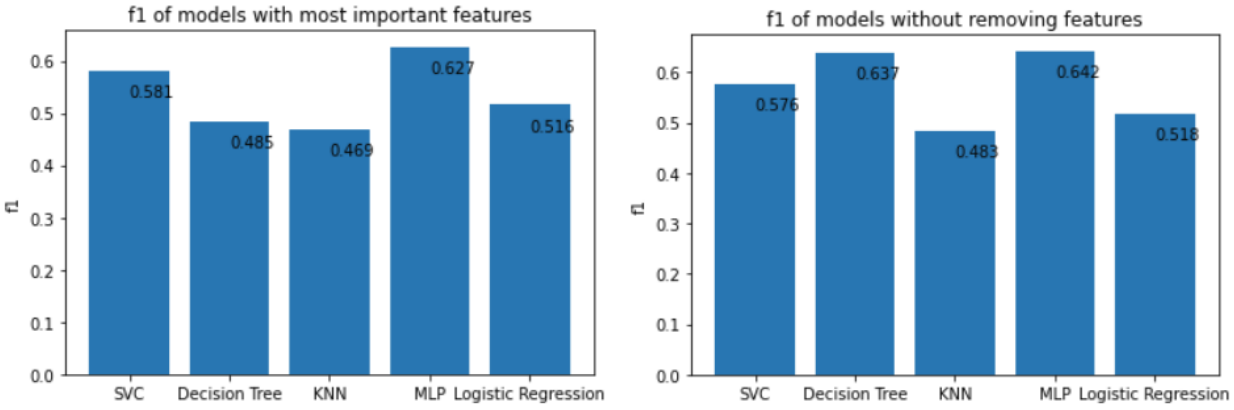
شکل بالا سمت چپ مدل ها با استفاده از PCA و شکل سمت راست بدون PCA مشابه با قبل آموزش داده شده اند.

همان طور که مشاهده میشود در مدلی مانند Decision Tree که از مشکل نحسی ابعاد رنج نمیبرد کاهش بعد باعث کاهش زیاد f1 مدل شده است ولی در روشی مثل SVM کمی f1 افزایش یافته است.

در کل ولی همان طور که مشاهده میشود کاهش بعد با PCA باعث افزایش دقت مدل ها نشده است و بهترین مدل هایی مانند MLP و Decision Tree که بدون کاهش بعد به دست آمدند دقت بهتری دارند بنابراین کاهش بعد را در نظر نمیگیرم.

بررسی اثر انتخاب ۹ ویژگی موثر با RandomForestRegressor

در زیر اثر انتخاب ۹ ویژگی برتر بر روی معیار F1 را بررسی خواهیم کرد.



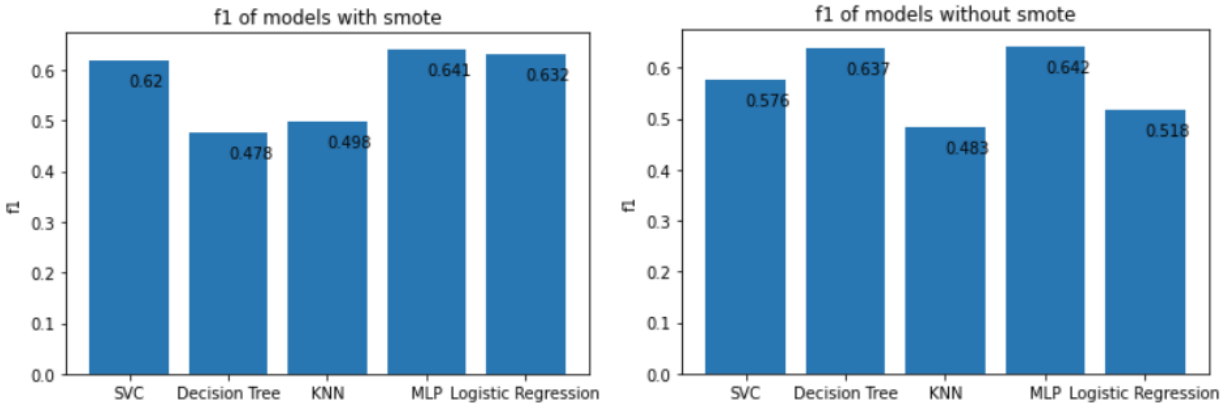
شکل بالا سمت چپ مدل ها با استفاده از انتخاب ۹ ویژگی برتر و شکل سمت راست با همه ویژگی ها مشابه با قبل آموزش داده شده اند.

همان طور که مشاهده میشود در مدلی مانند Decision Tree که از مشکل نحسی ابعاد رنج نمیبرد کاهش بعد باعث کاهش زیاد f1 مدل شده است ولی در روشی مثل SVM کمی f1 افزایش یافته است.

در کل ولی همان طور که مشاهده میشود همانند PCA حذف یک سری ویژگی ها باعث بهبود مدل نشده است و بهترین مدل هایی مانند Decision Tree که بدون حذف ویژگی ها به دست آمدند دقت بهتری دارند بنابراین حذف ویژگی ها را نیز در نظر نمیگیرم.

بررسی اثر اضافه کردن نمونه با SMOTE

روش درست کردن اضافه کردن داده ها به این گونه است که فقط باید بر روی داده ها آموزش اضافه شوند و نه بر روی داده های تست یا ارزیابی. که برای پیاده سازی این مورد از Pipeline در Sklearn استفاده شده است تا هنگام cross validation زدن با استفاده از GridSearchCV تنها داده ها بر روی داده ها آموزش تولید شوند و بر روی داده های validation یا همان foldها همان داده های قبلی را داشته باشیم.



در شکل بالا سمت چپ داده های آموزش متوازن شده اند و در شکل سمت راست همان داده ها اصلی است همان طور که مشاهده میشود روش هایی مانند SVM و Logistic Regression با این کار باعث افزایش چشمگیر F1 آنها شده است و روشی مانند Desicsion Tree به شدت F1 آن ها شده است. اما روش مانند MLP در هر دو مورد عملکر بهتری از بقیه داشته است.

بنابراین در کل اضافه کردن داده های آموزش با SMOTE را مناسب میدانیم تا مشکل عدم توازن کلاس ها را از بین ببرد.

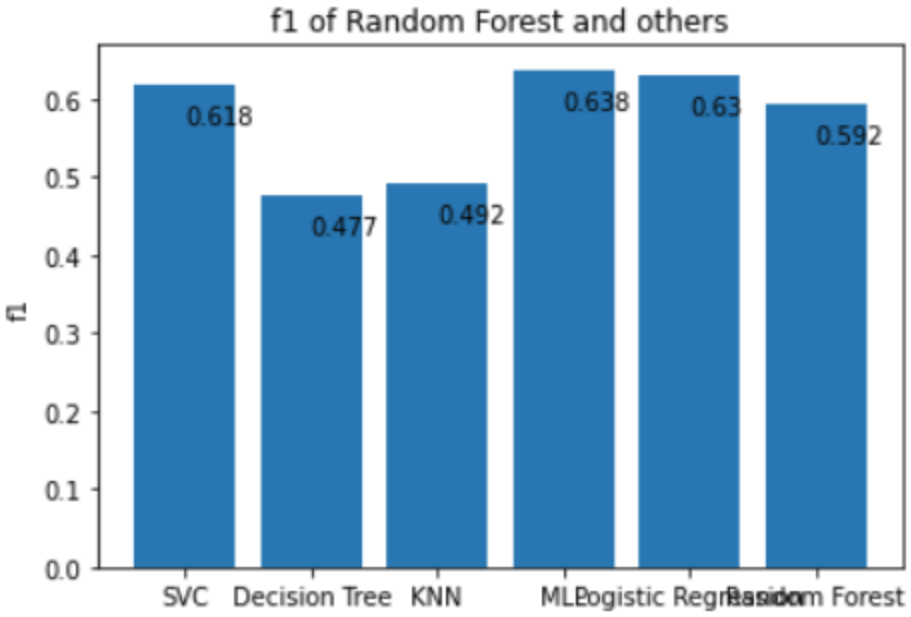
- **بنابراین مدل پیروز مدل MLP انتخاب میشود.**

مدل امتیازی

در زیر مدل Random Forest با مدل های مراحل قبل مقایسه شده است.

همان طور که مشاهده میشود f1 این مدل دقت کمتری از بهترین مدل یعنی MLP دارد.

مدل های مختلفی از جمله ADAboost و Naive bayes و Gradient Tree boosting و ... تست شدند و همه در دقت F1 کمتر از MLP بودند با این که در تعدادی دقت Accuracy کمی بالاتر میرفت. اما چون در این مسئله به علت عدم توازن کلاس ها دقت F1 درست است بنابراین معیار ما برای انتخاب بهترین مدل F1 است و بهترین مدل همان MLP انتخاب میشود.



۴- ارزیابی نهایی بر روی داده های تست

بهترین مدل مراحل قبل بر روی داده های تست ارزیابی شده اند و مقادیر مختلف متریک ها و هم چنین ماتریس confusion گزارش شده است.