داده کاوی

تمرین دوم – بخش پیاده سازی

آرش حریرپوش
9731505

آرش حریرپوش
9731505

.6

1) آماده سازی داده ها: در این مرحله ابتدا تعداد Null Value در هر ویژگی محاسبه می گردد تا به کمک آن ویژگی هایی که دارای missing value مشخص شود؛ شکل زیر تعداد Null Value در هر ویژگی را نمایش می دهد:

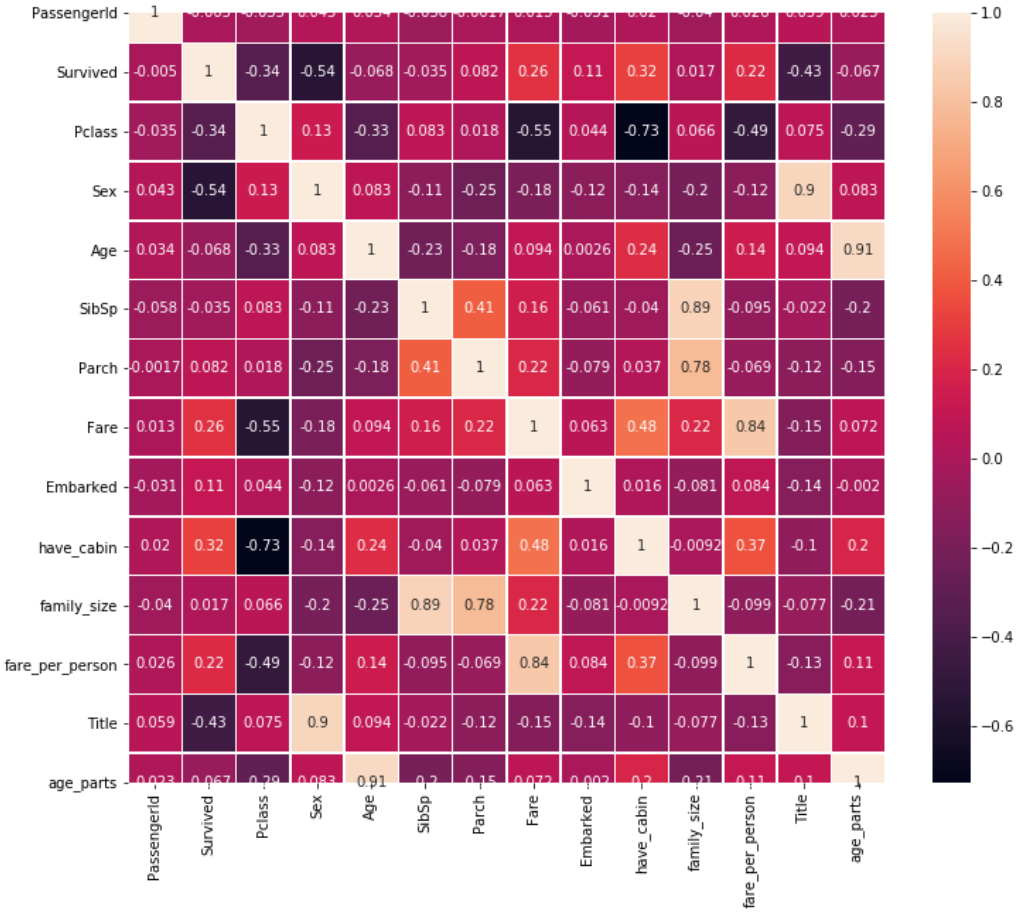
```
passengers.isnull().sum()

PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

همانطور که مشاهده می شود ویژگی های Age، Cabin و Embarked دارای Null Value می باشند، missing value ها در Age برابر میانگین ویژگی های Age قرار داده می شود و مقدار missing value ها در Cabin با مقداری جایگزین نمی شود بلکه به کمک missing value ها ویژگی Categorical جدیدی به نام have_cabin ایجاد می شود که نشان می دهد که آیا مسافر دارای کابین بوده است یا خیر؛ و مقدار missing value ها در ویژگی Embarked برابر پر تکرار ترین مقدار این ویژگی (که برابر S می باشد) قرار می گیرد. سپس به کمک ویژگی های SibSp و Parch ویژگی جدید ایجاد می شود که حاصل جمع مقدار های این دو ویژگی با یک دیگر می باشد که اندازه خانواده همراه فرد خریدار بلیط را نشان می دهد این ویژگی جدید family_size نام دارد. پس از آن به کمک این ویژگی یک ویژگی جدید به نام fare_per_person ایجاد می شود که برابر تقسیم ویژگی Fare بر family_size+1 می باشد که نشان می دهد هزینه هر بلیط به طور میانگین برای اعضای هر خانواده چقدر بوده است. حال به کمک ویژگی Name یک ویژگی جدید به نام Title ایجاد می شود که شامل عنوان افراد می باشد؛ با توجه به اینکه ویژگی Title یک ویژگی Categorical می باشد و برخی از مقدار های آن مشابه یک دیگر می باشد مقدار های بدست آمده را به صورت زیر برابر معادلشان قرار می دهیم:

عنوان های Miss، Mme، Ms، Mlle با عنوان Miss، عنوان های Mrs، Countess، Lady با عنوان Mrs، عنوان های Mr ، Dona، Major، Sir، Capt ، Don با عنوان Mr و بقیه عنوان ها که عبارتند از Rev ، Col، Jonkheer ، Dr، Master با عنوان Others جایگزین می شود. سپس به کمک ویژگی

آرش حریرپوش
9731505

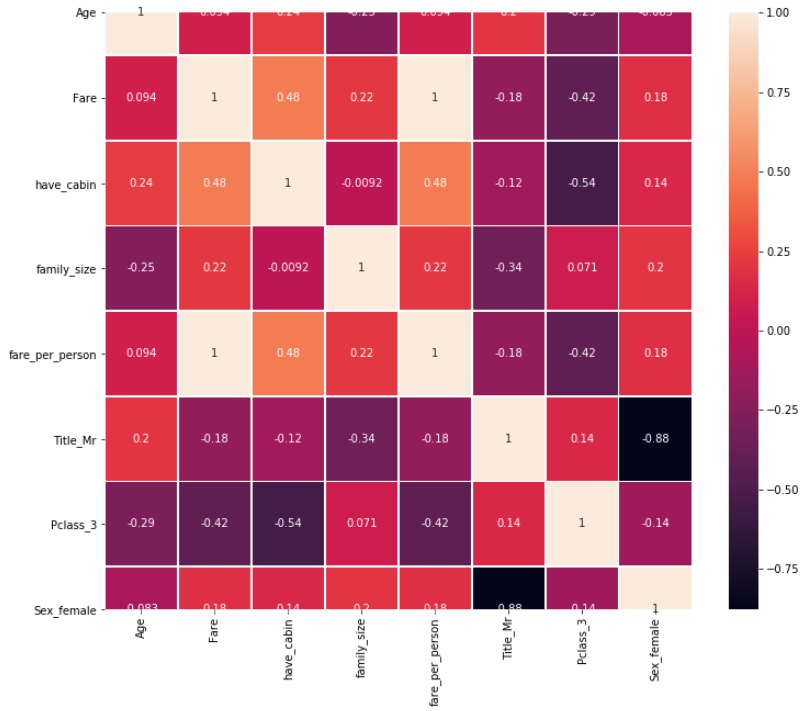
Age یک ویژگی جدید به نام age_parts می باشد که گسسته شده ویژگی Age می باشد و بدین صورت می باشد که داده های این ویژگی را بین 4 گروه دسته می کنیم. در نهایت برای مشاهده ارتباط ویژگی ها با یک دیگر ابتدا یک کپی از ویژگی ها ایجاد می گردد و سپس مقدار ویژگی های Sex و Embarked در آن به مقدار های عددی تبدیل می شود تا بتوان شباهت بین این ویژگی ها با بقیه ویژگی ها را در ماتریس Correlation مشاهده کرد:



حال از بین ویژگی های بالا ویژگی های غیر عددی که برابر با Name، Ticket و Cabin و همچنین ویژگی PassengerId که تنها یک id می باشد حذف می شوند و سپس بر روی ویژگی های Title، Pclass، Embarked، age_parts و Sex به دلیل اینکه categorical هستند One hot encoding انجام می شود. سپس به کمک نرم 1 در روش L1-based عملیات Feature Selection انجام می شود تا از بین ویژگی های موجود ویژگی های مناسب برای آموزش انتخاب شوند که پس از انجام این عملیات ویژگی های Age، Fare، have_cabin، family_size، fare_per_person، Title_Mr، Pclass_3 و

آرش حریرپوش
9731505

Sex_female به عنوان ویژگی های مطلوب برای آموزش مدل در نظر گرفته می شود، ماتریس Correlation این ویژگی ها برابر است با:



حال پس از انتخاب ویژگی ها داده ها را به نسبت 0.8 و 0.2 به داده های آموزش و تست تقسیم می کنیم.

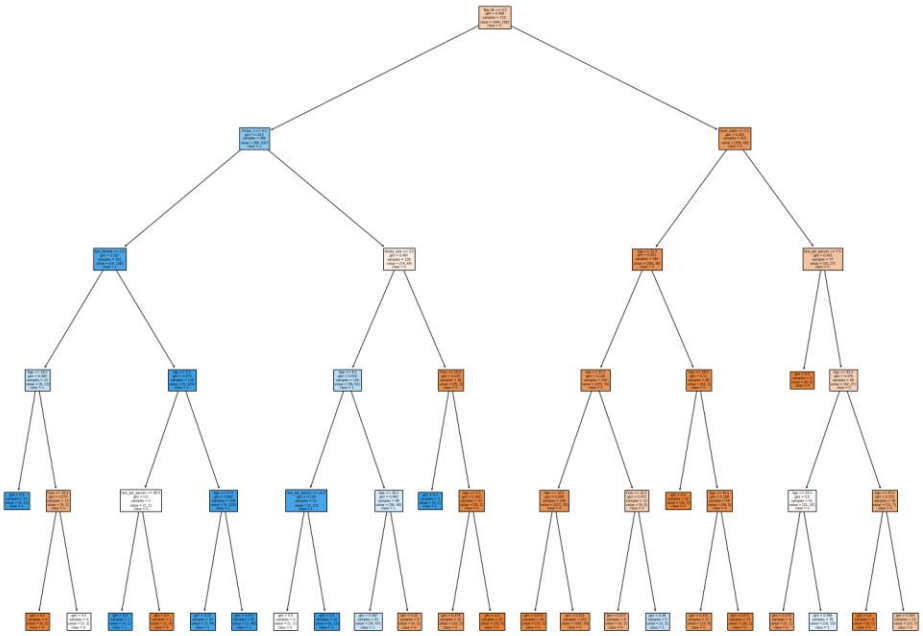
2) آموزش (training) و دسته بندی (classification): در اینجا چهار نوع درخت تصمیم با تابع تقسیم و عمق های مختلف آموزش داده می شود، که عبارتند از:

- تابع تقسیم Gini و حداکثر عمق برابر 5: در این حالت دقت مدل برابر است با:

```
Training Accuracy Is: 0.8497191011235955
Test Accuracy Is: 0.8547486033519553
```

و درخت تصمیم آموزش داد شده برابر است با:

آرش حریرپوش
9731505

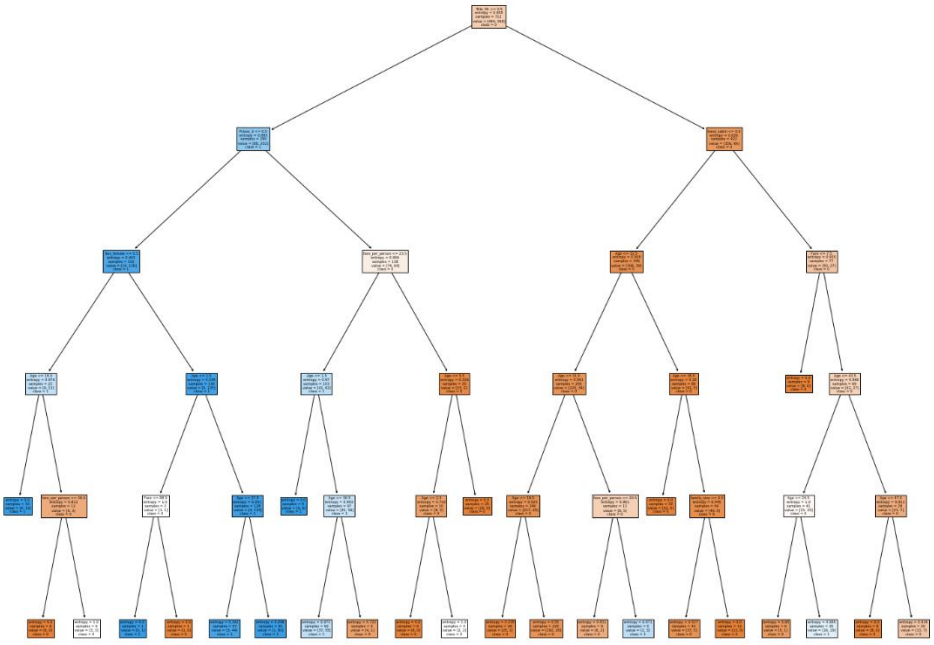


- تابع تقسیم Entropy و حداکثر عمق برابر 5: در این حالت دقت مدل برابر است با:

```
Training Accuracy Is: 0.8469101123595506
Test Accuracy Is: 0.8491620111731844
```

همانطور که مشاهد می شود با تغییر تابع تقسیم به Entropy مقدار دقت مدل بر روی داده های آموزش و تست اندکی کاهش کاهش یافته است، اما به صورت کلی تغییر زیادی بر روی دقت ها صورت نپذیرفته است؛ درخت تصمیم آموزش داده شده برابر است با:

آرش حریرپوش
9731505

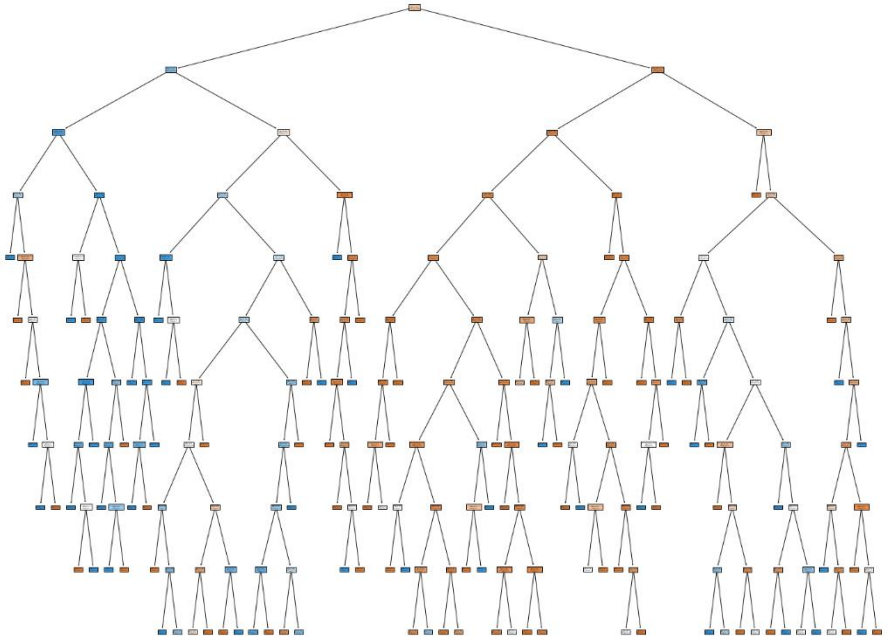


- تابع تقسیم Gini و حداکثر عمق برابر 10: در این حالت دقت مدل برابر است با:

```
Training Accuracy Is: 0.9283707865168539
Test Accuracy Is: 0.8100558659217877
```

با افزایش حداکثر عمق درخت فاصله بین دقت مدل بر روی داده های آموزشی افزایش و داده های تست کاهش یافته است و باعث می شود اندکی over fitting رخ دهد. درخت تصمیم آموزش داده شده برابر است با:

آرش حریرپوش
9731505

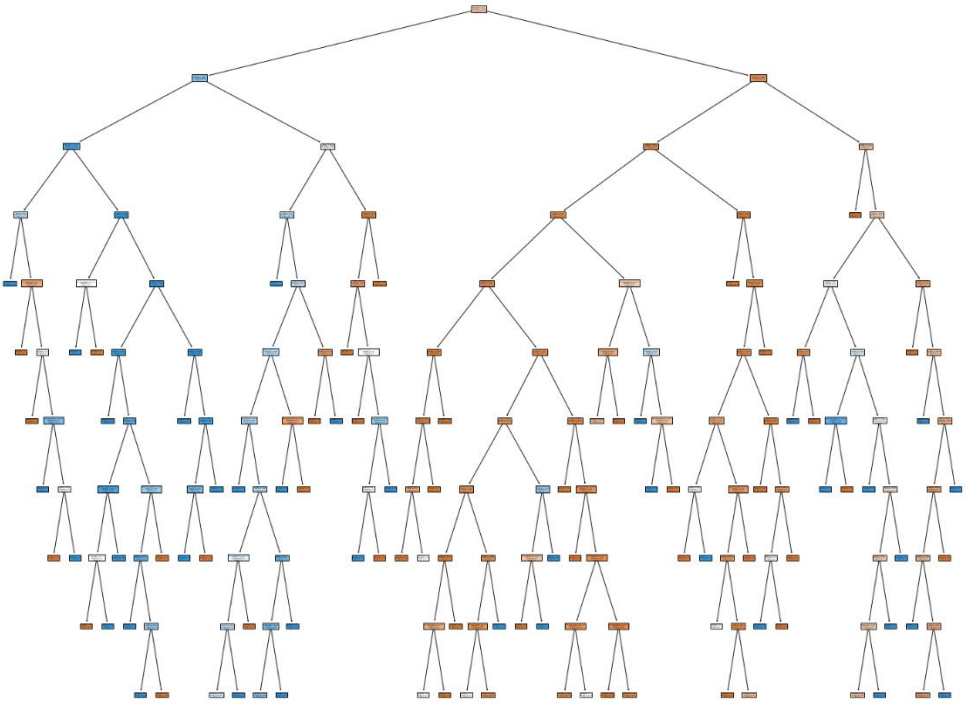


- تابع تقسیم Entropy و حداکثر عمق برابر 10: در این حالت دقت مدل برابر است با:

```
Training Accuracy Is: 0.90730337078651690
Test Accuracy Is: 0.8212290502793296
```

همانطور که مشاهد می شود دقت مدل بر روی داده های آموزشی نسبت به حالت قبل کاهش و بر روی داده های تست در این حالت نسبت به حالت قبل افزایش یافته است ؛ بنابراین نتیجه گرفته می شود با افزایش حداکثر عمق درخت امکان رخ دادن overfitting به دلیل بیشتر شدن فاصله بین دقت در داده های آموزش و تست افزایش می یابد و با تغییر تابع تقسیم از Gini به Entropy تغییر زیادی در نتایج ایجاد نمی شود اما در این مسئله در حالت Entropy فاصله بین دقت بر روی داده های آموزشی و تست کاهش می یابد؛ درخت تصمیم آموزش داده شده برابر است با:

آرش حریرپوش
9731505



(فایل ها و Notebook مربوط به این بخش در فولدر Q6 قرار دارد)

.7

1) آماده سازی داده ها: در این مرحله ابتدا تعداد Null Value در هر ویژگی محاسبه می گردد تا به کمک آن ویژگی هایی که دارای missing value مشخص شود؛ شکل زیر تعداد Null Value در هر ویژگی را نمایش می دهد:

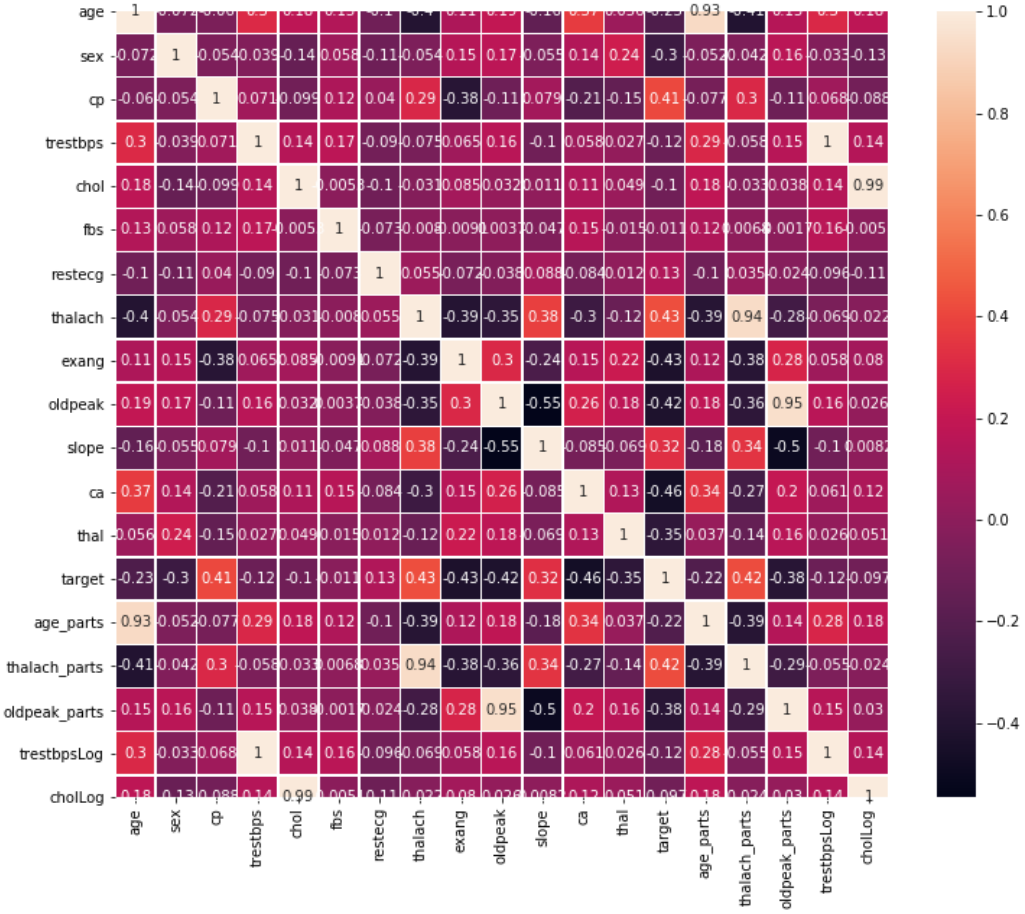
آرش حریرپوش
9731505

```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

همانطور که مشاهده می شود هیچ کدام از ویژگی دارای Null value نمی باشد، بنابراین با توجه به اینکه الگوریتم KNN به شدت به Outlier ها حساس می باشد در گام بعدی به کمک z-score داده های Outlier را حذف می کنیم، بدین صورت که مقدار z-score را برای همه ویژگی های عددی داده ها محاسبه می کنیم و سپس داده هایی که مقدار z-score یکی از ویژگی های آن ها بزرگ تر یا مساوی 3 باشد حذف می شود. پس از حذف Outlier ها ماتریس Correlation به صورت زیر ترسیم می شود:

آرش حریرپوش
9731505

حال با توجه به اینکه مقدار Correlation بین هیچ یک از ویژگی ها به اندازه کافی زیاد نیست تا بتوان آن ها را ترکیب کرد، بنابراین هیچ ویژگی جدیدی از ترکیب ویژگی های قدیمی تولید نمی شود؛ سپس با گسسته سازی ویژگی های age، thalach و oldpeak ویژگی های جدید age_parts، thalach_parts و oldpeak_parts ایجاد می شود که بدین صورت می باشد که هر یک از این ویژگی های پیوسته به 4 دسته تقسیم شده است؛ سپس این ویژگی را با کمک LabelEncoder به ویژگی های عددی تبدیل می کنیم تا بتوانیم در الگوریتم های مختلف از آن استفاده کنیم؛ سپس با Log گرفتن از ویژگی های trestbps و chol ویژگی های جدید trestbpsLog و cholLog ایجاد می شود. حال پس از ایجاد ویژگی های جدید دوباره ماتریس Correlation را رسم می کنیم تا به کمک آن ویژگی های مشابه را حذف کنیم:

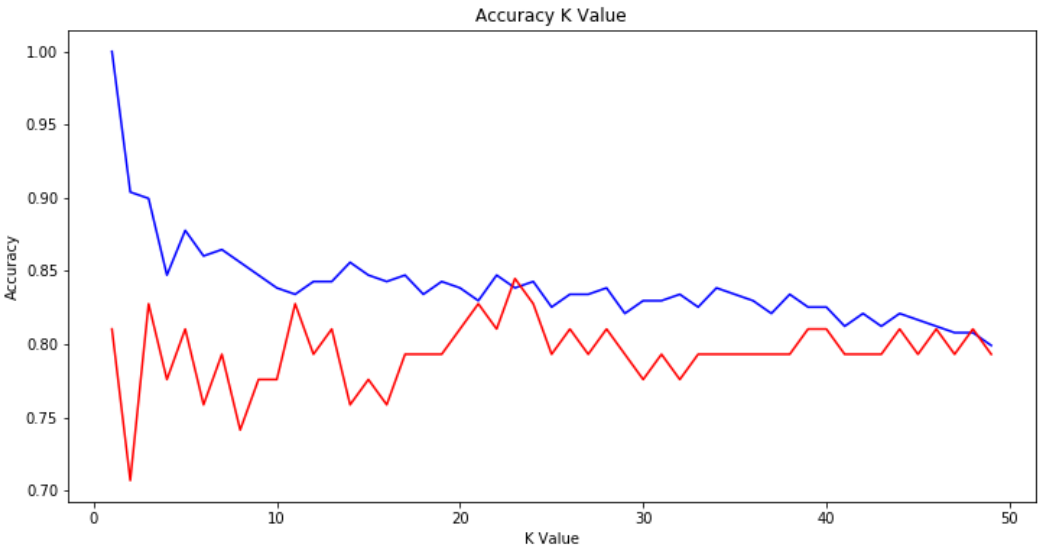


با توجه به ماتریس Correlation بالا ویژگی های ستونی که مقدار Correlation آن ها با یکی از ویژگی های سطری بزرگ تر از 0.9 باشد حذف می شود، بنابراین ویژگی های age، trestbps، chol،

oldpeak و thalach حذف می شوند. در نهایت داده ها را به نسبت 0.8 و 0.2 به داده های آموزشی و تست تقسیم می کنیم.

2) آموزش (training) و دسته بندی (classification):

ابتدا برای ساختن یک مدل KNN باید مقدار K مناسب برای مسئله پیدا شود، برای این کار ابتدا میزان دقت مدل برای داده های آموزش و تست برای K های 1 تا 50 محاسبه می شود و سپس مقدار K که تفاوت دقت آموزش و دقت تست در آن از همه کمتر است به عنوان بهترین K انتخاب می شود. در شکل زیر خط آبی رنگ دقت مدل بر روی داده های آموزشی و خط قرمز رنگ دقت مدل بر روی داده های تست را نمایش می دهد:



پس از محاسبه تفاوت دقت مدل بر روی داده های تست و آموزش در نهایت مقدار K برابر 46 در نظر گرفته می شود؛ و مدل KNN با مقدار K برابر 46 آموزش داده می شود که دقت این مدل برابر است با:

```
Training Accuracy Is: 0.8122270742358079
Test Accuracy Is: 0.8103448275862069
```

سپس مدل Gaussian Naive Bayes آموزش داده می شود که دقت این مدل بر روی داده های آموزشی و تست برابر است با:

```
Training Accuracy Is: 0.8602620087336245
Test Accuracy Is: 0.8448275862068966
```

همانطور که مشاهده می شود در مدل های آموزش داده شده دقت مدل Gaussian Naive Bayes بر روی داده های آموزش و تست بیشتر از مدل KNN می باشد، در اینجا فاصله بین دقت تست و آموزش در مدل KNN بسیار کمتر است که یکی از دلایل آن فرضی است که برای پیدا کردن K مناسب در مراحل

آرش حریرپوش
9731505

قبل در نظر گرفته شده است که K انتخاب شود که کمترین فاصله بین دقت داده های آموزشی و تست وجود داشته باشد.

(فایل ها و Notebook مربوط به این بخش در فولدر Q7 قرار دارد)