ابتدا داده را خوانده و در دیتافریم ذخیره می کنیم سپس برخی بررسی های اولیه انجام می دهیم تا با انواع فیچر ها و کلیات داده ها آشنا شویم و نمونه های کاملا مشابه (duplicate) را حذف می کنیم سپس تمام نمونه داده هایی که دارای (null) هستند را حذف می کنیم تا به بررسی تاثیر داده ها روی هم بپردازیم.  
ابتدا جدول کاهش یافته را مجدد بررسی می کنیم و فیچر هایی که از نوع عدد نیستند را شناسایی می کنیم بدای بررسی های آینده نیاز داریم که این داده ها را به داده های عددی تبدیل کنیم ابتدا داده تاریخ را به داده های سال ماه و روز می شکنیم که عددی هستند سپس به بررسی باقی فیچر های غیر عددی می پردازیم به دو روش میتوان انها را هندل کرد اگر داده های غیر عدد ترتیبی باشند مثل اول دوم سوم ... با (label encode) انهارا عددی می کنیم و به هر کدام عددی اختصاص می دهیم ولی اگر ترتیبی نداشته باشند با (one hot encode) تبدیل را انجام می دهیم که برای مقادیر فیچر تک تک فیچر جداگانه می سازد و اگر ان نمونه از ان نوع بود به ان ۱ اختصاص می دهد در غیر این صورت صفر اختصاص می دهد با اینکه اره و نه ترتیبی ندارند ولی اکثرا این داده را به صفر و یک تبدیل می کندد بعد از انجام این تبدیلات به سراغ بررسی ارتباطات می رویم کورلیشن این کار را برای ما انجام می دهد و به نوعی به ما نشان می دهد هر کدام از فیچر ها چقدر روی فیچر دیگر تاثیر میگذارد.   
کورلیشن بقیه فیچر ها با تارگت را حساب می کنیم و از میان انها فیچر هایی با بیشترین تاثیر را انخاب می کنیم .  
در اینجا من فیچر هایی را انتخاب کردم که قدر مطلق کورلیشنشان بزرگ تر از ۰.۳ باشد.   
سپس به ۵ فیچر به جز تارگت می رسیم حال این ۵ فیچر را کورلیشنشان باهم را حساب می کنیم تا متوجه شویم فیچره هست که خیلی کورلیشن بالایی نسبت به فیچر دیگر داشته باشد یا نه با این کار می خواهیم متوجه شویم فیچر های به دردنخور را جمع نمی کنیم چون اگر کورلیشن خیلی بالا باشد یکی از انها اطلاعات دیگری را به ما می دهد و لزومی به حضور هر دو فیچر نیست از دیتای اولی این ۵ فیچر به علاوه‌ی تارگت را جدا می کنیم و به بررسی (null) ها می پردازیم.

حال تعداد نال های هر فیچر را بررسی می کنیم از انجایی که تعداد نال های فیچر تارگت وفیچدپر بارش در ان روز کم هستند و خیلی راه مناسبی برای پر کردنشان نداریم نمونه های نال انها را حذف می کنیم که اکثرا هم با هم منطبق هستند سپس همان تبدیل به فیچر عددی که‌ در مرحله قبل اپبرای این دو فیچر زدیم باز هم اجرا می کنیم.

در اینجا نال ها را به دوصورت پر کردیم یکی اینکه دو فیچر ابری را از روی هم دیگر و فیچر(sunshine ) را از روی فیچر (moisture) پر می کنیم این پر کردن از روی این فیچر ها به خاطر کورلیشن بالای میان این فیچر هاست (البته اصل این پرکردن با کاری که من انجام دادن متفاوت است و به نظرم باید با ایجاد مدل های پیشبینی مثل کلاسیفیگیشن یا غیره این کار را انجام دهیم یعنی مدل را روی فیچر هایی که با فیچر هدف (نال دارد) ترین کنیم و از مدل برای پیشبینی مقادیر از دست رفته استفاده کنیم ) سپس بقیه داده های نال را حذف می کنیم و دیتا فریم به دست امده را با استفاده از (pair plot)رسم کنیم انگاه متوجه شدیم که یک شکل عجیبی در داده هایمان به وجود امده و به نوعی خیلی داده ها همخوانی ندارند.

سپس راه دیگری برای پر کردن نال ها استفاده می کنیم برای (sunshine) از روی میانگین خود فیچر نال هارا پر می کنیم و برای فیچر های ابری هم کاری انجام نمی دهیم و انهارا حذف می کنیم دیتا فریم به دست امده متشکل از نزدیک ۹۰۰۰۰ سمپل است که به نظر مناسب می آید.

سپس به بررسی داده های پرت می پردازیم این کار را با زی اسکور انجام می دهیم این اسکور درواقع کاری که انجام می دهد میانگین هر فیچر و انحراف معیار ان را حساب می کند و نشان می دهد هر داده اختلافش با میانگین به اندازه چه ضریبی از انحراف معیار است داده هایی که اختلافشان سه برابر انحراف معیار باشد را داده پرت محسوب می کنیم و آن را حذف می کنیم.

سپس توزیع فیچر هارا بررسی می کنیم و مقدار (skew) انهارا هم حساب می کنیم در واقع این مقدار نشان می دهد که داده ما چه مقدار با توزیع نرمال فاصله دارد و تمرکز داده ها به سمت چپ است یا راست هر چه این مقدار به صفر نزدیک تر یعنی داده به توزیع نرمال نزدیک تر است .  
تصمیم گرفتیم که داده هایمان را نرمال سازی نکنیم چون انقدری هیچ کدام دور از نرمال نیستند ولی کد نرمال سازی در پروژه موجود است.

بعد از پیش پردازش ها به ترین مدل های مختلف می پردازیم که ابتدا درخت تصمیم گیری را ترین می کنیم   
داده هارا به سه بخش تقسیم می کنیم یکی تست دیگری ترین و آخر هم ولیدیشن ، این بخش ولیدیشن برای سنجش اورفیت و آندرفیت است   
بعد از ترین مدل میزان پیشبینی صحیح روی ترین و ولیدیشن را می سنجیم اگر اختلاف زیادی بین این دقت ها باشد نشان دهنده اور فیت و اگر نزدیک به هم باشند و هر دو نسبتا مقدار مناسبی داشته باشند درست ترین کرده ایم   
هایپر پارامتر موثر را بیشترین عمق درخت قرار می دهیم و برای عمق های ۱۰ تا ۱۰۰ میزان درستی پیشبینی روی ولیدیشن و ترین را حساب می کنیم و با مقیاسه این اعداد هایپر پارامتر بهینه را پیدا می کنیم   
بهترین عمق را من ۵۰ در نظر گرفتم و سپس ترین نهایی را انجام دادم   
این ترین مدل نهایی روی مجموع داده های ترین و ولیدیشن انجام می دهیم با استفاده از مدل مقادیر تارگت برای داده های تست را پیشبینی می کنیم و سپس پیشبینی خود را با مقادیر هدف واقعی مقایسه می کنیم.

و از این مقایسه ماتریس اشفتگی را تشکیل می دهیم این ماتریس در واقع نشان می دهد   
چه تعداد از داده هایی که مقدار تارگت واقعیشان ۰ بوده و ما درست و غلط پیشبینی کردیم و همین تعداد ها برای داده هایی که تارگت واقعیشان ۱ و ما درست و غلط پیشبینی کردیم مقادیر درستی پرسیژن و ریکال و اف۱ هر کدام نسبت این اعداد هستند که در اخر بیان خواهیم کرد.

سپس به ترین داده های برای مدل (knn) می پردازیم این مدل موثر ترین هایپر پارامترش مقدار کا است که این را هم مثل قبل با یک حلقه درستی پیشبینی اش را روی داده های ولیدیشن و ترین بررسی می کنیم و کا برابر با ۳ را انتخاب می کنیم حال می دانیم در این مدل کاهش بعد و اسکیل کردن فیچر ها معمولا تاثیرات مثبتی روی مدل می گذارند مثل وزن دهی مشابه ، حذف اوتلایر ها ، محاسبه سریع تر فاصله و ... حال کاری که اینجا کردیم داده هارا به ۳ صورت اماده کردیم برای ترین یکی با کاهش بعد یکی با اسکیل و دیگری با هر دوی انها بر خلاف انتظار در مثال ما بهبودی در نتایج پیشبینی صورت نگرفت و په بسی عملکرد ضعیف تر هم شد همچنین تشخیص نیاز به اسکیل داده را با استفادهواز باکس پلات انجام دادیم که رنج داده هر فیچر را به خوبی نشان می دهد   
سپس در اخر هم مدل را با کا برابر با ۳ و داده اموزشی اسکیل شده ترین کردیم و ماتریس اشفتگی و دیگر پارامتر ها را خروجی دادیم.

سپس به مدل (svm) می رسیم در این مدل هم اسکیل و کاهش بعد تاثیر گذارند ولی به علت کم بود وقت فرصت نکردم اعمال کنم روی داده (مشابه مدل knn) ابرپارامتر های این مدل زیاد هستند و ابتدا برای پیدا کردنشان از تابع گریدسرچ حواستم استفاده کنم که ابر پارامتر های با بهترین دقت روی داده را خروجی می دهد استفاده کنم ولی خیلی زمان بر بود(عکس قرار داده شد) به همین خاطر با ابر پارامتر های پیشفرض ان را ترین کردیم و سپس با مدل حاصل ماتریس اشفتگی را برای داده های تست محاسبه کردیم.

Accuracy= Number of Correct Predictions / Total Number of Predictions

Precision=True Positives / True Positives+False Positives

Recall= True Positives/True Positives+False Negatives

F1 Score=2×( Precision×Recall/Precision+Recall)