به نام خدا

محمدرضا عرب زاده

810195429

Prj4

091DD730BC700B43D4555D3382FE99EE2622EE6CF78D7D90E208441824E501E2

برای پیاده سازی در ابتدا بعد از خواندن داده ها توسط پانداس، عملیات درهم سازی بر روی آنها انجام می شود تا اطلاعات موجود بهتر پخش شوند.

سپس با استفاده از sklearn درخت تصمیم بر روی 80 درصد داده ها یعنی تا ایندکس 240 انجام می شود و بعد با باقی مانده اطلاعات یعنی از ایندکس 240 به بعد درخت های ساخته شده تست می شوند. دقت خروجی در این حالت به طور میانگین 78 درصد و در بهترین حالت 83 درصد می باشد.

برای bagging پنج وکتور به طول 150 با داده های رندوم بین 0 تا 240 تولید شده و سپس بر اساس همین دسته داده ها 5 درخت ساخته می شود. برای ارزیابی نیز از داده های ایندکس 240 به بعد استفاده می شود و دقت دیده شده در این حالت به طور میانگین 78 درصد و در بهترین حالت 86 درصد بوده است.

برای random forest نیز همانند bagging داده ها را در پنج دسته قرار داده و 5 ویژگی را به طور رندوم برای هر دسته انتخاب کرده و درخت را برای آن می سازیم. دقت بدست آمده برای این حالت به طور میانگین 74 درصد و در بهترین حالت برابر 86 درصد شد.

Bootstraping یک روش برای مدل سازی می باشد که با استفاده از نمونه برداری تصادفی سعی در کم کردن انحراف از معیار و واریانس پیشبینی دارد. با توجه به اینکه در این روش از چندین دسته بندی به صورت تصادفی استفاده می شود، و هر دسته بندی اندازه کوچک تری نسبت به داده های اصلی دارد، انحراف از معیار کاهش می یابد. چون از چندین دسته بندی استفاده کرده و روی هر کدام جداگانه درخت تصمیم را می سازد، مقدار واریانس پیشبینی خود را کاهش می دهد.

Overfiting یک مشکل در پیشبینی داده ها می باشد به این صورت که مدل ساخته شده بر روی داده های train بسیار خوب کار می کند اما بر روی داده های تست دقت مناسبی را در پیشبینی نمی تواند داشته باشد. در خت با توجه به اینکه بر روی داده ها مستقیما ساخته می شود، انعطاف کمی در برابر داده های متفاوت خواهد داشت و نمی تواند حالت هایی را که در داده های train نیامده است را به خوبی پیشبینی کند. Bagging سعی دارد با دسته بندی داده ها و انتخاب آنها بر اساس شانس برای هر دسته، حالت های مختلف را به وجود آورد تا بتواند واریانس خود را کم کند.

هر دو بسیار شبیه به هم می باشند با این تفاوت که random forest تنها از درصد محدودی از ویژگی ها برای هر نود استفاده می کند. این روش سعی دارد زمان train کردن را کاهش دهد. باتوجه به اینکه random forest درصد خاصی از ویژگی ها را مورد بررسی قرار می دهد، در نتیجه برای داده های با تعداد زیاد ویژگی گزینه مناسبی می باشد. همچنین با توجه به ماهیت آن می توان از آن به صورت موازی بهره برداری کرد.

در حذف ویژگی ها دو ویژگی ca و cp بیشترین تاثیر را در دقت می گذاشتند.

دقت به صورت میانگین برای دو حالت درخت تصمیم و خوشه بندی یکسان است اما در صورت انتخاب بهینه خوشه بندی بهتر عمل می کند. همچنین دقت جنگل تصادفی در حالت میانگین بد تر است اما در حالت بهینه دقتی اندازه خوشه بندی ارائه می دهد و همچنین می تواند سریع از آن درخت تصمیم را پیاده سازی کند.