

بسمه تعالی

Artificial Intelligence

Assignment 4

Mohammad Mahdi Islami
810195548

توضیحات کلی

ابتدا در این بخش داده ها را از فایل **CSV** خوانده و سپس با استفاده از دستوری در کتابخانه **sklearn** درخت تصمیم را تشکیل میدهیم و دقت آن را با دادن **Target_Data** به آن میسنجیم و به صورت زیر میشود:

(۱)

Q1: 0.6557377049180327

سپس داده های موجود را به صورت تصادفی به پنج دسته‌ی ۱۵۰ تایی تقسیم میکنیم تا گام اول برای تشکیل جنگل تصادفی را برداریم.

سپس نتیجه هر یک از این درخت ها را با توجه به داده های ترین شده مینگریم و طبق رای گیری هر گزینه ای که درخت های بیشتری آن را به عنوان جواب بازگردانده بود آن را انتخاب میکنیم. و سپس دقت آن را با دادن **Target_Data** میسنجیم که به صورت زیر میشود:

(۱ و ۲)

Q2-2: 0.6885245901639344

سپس از میان ویژگی ها هر بار یکی را در نظر گرفته و حذف میکنیم و باز با استفاده از کتابخانه **sklearn** دقت حاصل شده را با دادن **Target_Data** میسنجیم.

(۳)

0.6557377049180327 : **age** - removing feature - 3-Q2
0.639344262295082 : **sex** - removing feature - 3-Q2
0.6721311475409836 : **cp** - removing feature - 3-Q2
0.7377049180327869 : **trestbps** - removing feature - 3-Q2
0.6229508196721312 : **chol** - removing feature - 3-Q2
0.6557377049180327 : **fbs** - removing feature - 3-Q2
0.6721311475409836 : **restecg** - removing feature - 3-Q2
0.6885245901639344 : **thalach** - removing feature - 3-Q2
0.639344262295082 : **exang** - removing feature - 3-Q2
0.6557377049180327 : **oldpeak** - removing feature - 3-Q2
0.7704918032786885 : **slope** - removing feature - 3-Q2
0.7377049180327869 : **ca** - removing feature - 3-Q2
0.6721311475409836 : **thal** - removing feature - 3-Q2

سپس از میان ویژگی ها ۵ ویژگی را در نظر میگیریم و مانند بخش های قبل که درخت هایی با اندازه های ۱۵۰ تایی و داده های یادگیری رندوم انتخاب کرده بودیم جنگل تصادفی را ایجاد میکنیم و که هم ویژگی ها در آن رندوم است و هم داده های ترینینگ و تعداد درخت ها نیز ۵ عدد است:

۴ و ۵)

0.5901639344262295 :4-Q2
0.6885245901639344 :5-Q2

سوالات در صورت پروژه

۱ - BootStrap:

یک روش برای انتخاب تصادفی یک مشخصه در برنامه است. مثل انتخاب تصادفی داده های یادگیری برای برنامه و درخت تصمیم این روش باعث تخفیف انحراف معیار و واریانس میشود.

۲ - Overfitting:

به این معناست که برنامه ی ما که با داده های یادگیری شده در حال کار است اگر یک داده ای که تا حالا با آن مواجه نشده است را نتواند به درستی هندل کند و در دچار خطا شود و تنها با داده هایی که در یادگیری با آن ها آشنایی دارد را بشناسد و آن ها را هندل کند.

۲ - Bagging:

برای تولید مجموعه های آموزشی متفاوت استفاده میشود.

از یک مجموعه ی آموزشی به طول N ، N داده به صورت تصادفی و همراه با جایگزینی استخراج میشود. فرآیند فوق L بار تکرار میشود. از رأیگیری (میانگینگیری) برای ترکیب نتایج استفاده میشود. پایداری الگوریتم یادگیری را افزایش میدهد و در واقع واریانس را کاهش میدهد.

اگر درختی منجر به پاسخ نشد یا داده ای اشتباه پیشنهاد داد درخت های دیگر این را پوشش دهند و پاسخ به درستی بیشتر نزدیک شود.

۳ - با استفاده از Bagging میتوان نتایج random forest را ارزیابی کرد و پاسخ درخت هارا با جمع کرد و در واقع برآیند پاسخ های درخت های تصادفی را برگرداند.

در واقع random forest داده های یادگیری را میان درخت های تصمیم پخش میکند تا خطای ماشین را کاهش دهد.

-۴

چون داده ها هر بار تصادفی انتخاب میشوند در نتیجه این مقادیر متفاوت است اما میتوان گفت که باید نزدیک به هم باشند و تفاوت زیادی با هم نداشته باشند.

