۶۱۰۷۹۹۰۰۱ سید امیررضا موسوی مجد حل سوال سوم از پروژه یادگیری ماشین

در این سوال، (قسمت سوم پروژه) از یک سری کد های پیاده شده در قسمت اول بهره میبریم. به این صورت که مطابق سوال اول، metadata و Normal.counts.voom را در سیستم لود میکنیم و در نهایت merge میکنیم و به دیتا فریمی تحت عنوان feature_vec میرسیم.

از آنجایی که در سوال اول دیدیم که svm radial و random forest با مجموعه دیتاستی که از روش Recursive Feature Elimination بدست آمده اند، نتیجه بسیار خوبی دارند، در اینجا هم از این دو متد و این روش کاهش ویژگی بهره میبریم.

مطابق سوال اول ابتدا highly correlated genes را حذف میکنیم.

Feature Selection by filtering out highly correlated genes

نکته مهم در سوال سوم این هست که ما دو سری ماشین را train میکنیم. در لایه اول یک مدل بهینه شده تشخیص میدهد که آیا فرد بیمار هست یا خیر، و در لایه دوم فرد بیمار را براساس شدت بیماری به Advanced و Non-advanced طبقه بندی میکند. فلذا برای لایه اول باید رکورد simplified class را به normal و fibrosis ، موقتا تغییر دهیم. تابع decision_layer_one این تبدیل را انجام میدهد. یعنی رکورد Simplified_class که سه فاکتور داشت را به رکورد جدیدی تبدیل میکند که ۲ فاکتور دارد.

سپس برای لایه اول ، با کمک گرفتن از داده های filter شده، نوعی RFE بوسیله ماشین random forest انجام میدهیم. هدف ما برای اینکار یافتن ۱۰ ویژگی مهم برای طبقه بندی صحیح بیمار و سالم میباشد. پس از train کردن این مدل، ۱۰ ویژگی برتر را SD_rf_rfe_C_layer1 مینامیم و مدل آموزش دیده را از بین میبریم.

سپس از داده SD_rf_rfe_C_layer1 کمک میگیریم و دو تا ماشین svm_radial و random_forest میسازیم. (مطابق با random seed) هر کدام از این دو ماشین که طبقه بندی بهتری انجام بدهند(بر اساس accuracy)، بوسیله یک متغیر Boolean تحت عنوان choose_svm_layer1 انتخاب میشوند. در هرصورت برای لایه اول، مدل های ما train شده اند.

به لایه دوم طبقه بندی میرویم.

در این لایه ما به دیتاهای حاوی رکورد Simplified_class که Normal هستند نیاز نداریم. پس دیتا فریم feature_vec_filtered را دستکاری میکنیم تا برای آموزش ماشین دوم، دیتافریم مناسبی تهیه بکنیم.

پس از حذف کردن افراد سالم، داده ۱۱۸ فرد بیمار باقی میماند که باید طبقه بندی شوند.

در این لایه هم مجددا یک ماشین random forest اولیه میسازیم که RFE انجام دهد و ۱۰ ویژگی برتر برای طبقه بندی در لایه دوم را SD_rf_rfe_C_layer2 مینامیم. ماشین فوق را حذف کرده و مجددا دو تا ماشین svm_radial و random forest میسازیم. و هر دو را train میکنیم. هرکدام بهتر بود مطابق با متغیر بولین choose_svm_layer2 انتخاب میشوند.

در اینجا، کار آموزش ماشین ها به اتمام رسیده است. و آماده هستیم که این الگوریتم که Nested_Machine_Learning_Classifier مینامیم را تست بکنیم.

قبل از آن، به یک گزارش typical در مورد عملکرد ماشین های آموزش دیده میپردازیم

- ***choose_svm_layer1 = FALSE***
- ***choose_svm_layer2 = TRUE***
- ***Accuracy of svm_radial_SD_rf_rfe_C_layer1 = 0.853 & Kappa = 0.663***
- ***Accuracy of rf_SD_rf_rfe_C_layer1 = 0.859 & Kappa = 0.680***
- ***Accuracy of svm_radial_SD_rf_rfe_C_layer2 = 0.937 & Kappa = 0.876***
- ***Accuracy of rf_SD_rf_rfe_C_layer2 = 0.903 & Kappa = 0.805***

حال، این مدل هیبرید را تست میکنیم. از دیتا فریم اولیه ۳۹ رکورد را رندوم انتخاب میکنیم. به ماشین اول و سپس ماشین دوم میدهیم. نتیجه به صورت زیر است:

pred_lay1	*Fibrosis*	*Normal*
Fibrosis	*22*	*0*
Normal	*0*	*17*

Accuracy is 1 in the first layer of classification.

pred_lay2	*Advanced_fibrosis*	*Non_advanced_Fibrosis*
Advanced_fibrosis	*9*	*0*
Non_advanced_Fibrosis	*3*	*10*

Accuracy is 0.863636363636364 in the second layer of classification

به نظر میرسد این روش (دو سری تفکیک دوتایی) بهتر از یکسری تفکیک ۳ تایی باشد که در سوال اول دیدیم

در ضمن داده های انتخاب شده، ژن های زیر هستند و هیچ یک از ۴ متادیتا توسط الگوریتم RFE انتخاب نشد.

SD_rf_rfe_C_layer1

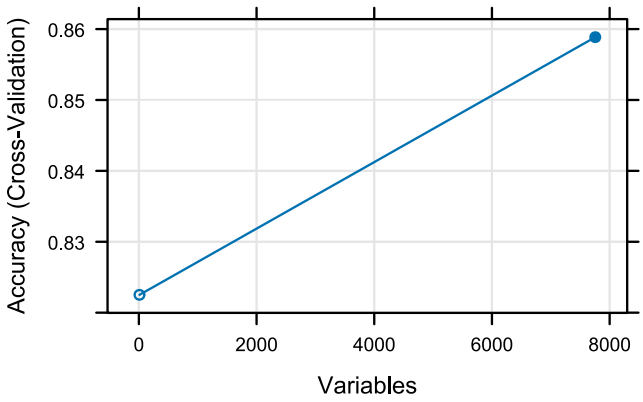
ENSG00000006704 ENSG00000164588 ENSG00000112139 ENSG00000197301 ENSG00000140986
ENSG00000078549 ENSG00000175868 ENSG00000189332 ENSG00000217442 ENSG00000179873

SD_rf_rfe_C_layer2

ENSG00000173917 ENSG00000240583 ENSG00000188517 ENSG00000170476 ENSG00000164114
ENSG00000189292 ENSG00000129538 ENSG00000148180 ENSG00000241644 ENSG00000169442

در ضمن در مدل های rf_rfe که در ابتدا آموزش دادیم، روند تغییر دقت و تعداد ویژگی های انتخاب شده از نمودار زیر پیروی میکند. (فیچر هایی که از فیلتر رد شده اند، وارد الگوریتم شده بودند)

Layer1



Layer2