گزارش تکلیف اول

داریوش حسن‌پور آده

۹۳۰۸۱۶۴

۱ قسمت اول تکلیف

- **داده هاي جدول ۱ را به فرمت مناسب براي ورود به نرم افزار وکا آماده نمائید.**

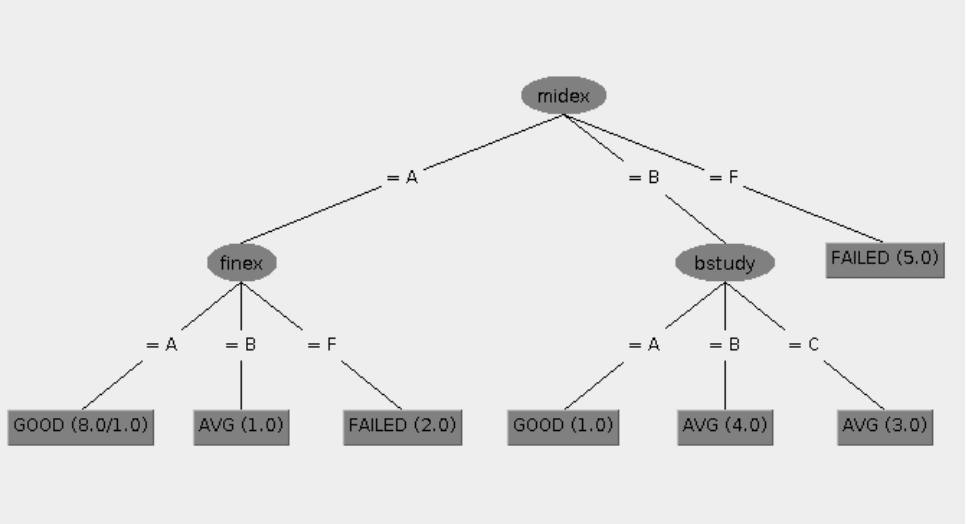
این داده‌ها را بصورت زیر به فرمت فایل قابل پردازش برای وکا بدست آورده شد – که به صورت پیوست در سامانه بارگذاری شده است. در داده‌های ایجاد شده اسامی ویژگی‌ها به‌صورت جدول ۱ آمده شده است.

نام فارسی ویژگی	نام انگلیسی ویژگی	نمایش «خوب»	نمایش «متوسط»	نمایش «ضعیف»	نمایش «زیر۵۰»
حضور فعال در کلاس	apresent	A	B	C	–
مطالعه‌ی هفتگی کتاب	bstudy	A	B	C	–
مطالعه‌ی از روی جزوه	hstudy	A	B	C	–
میان نیمسال	midex	A	B	–	F
میان نیمسال	finex	A	B	–	F
تکالیف	asgnmnt	A	B	C	–
تحقیق	resrch	A	B	C	–
پروژه	projct	A	B	C	–
نمره‌ی نهایی	final	GOOD	AVG	–	FAILED

جدول۱: نمایش ویژگی‌ها در فایل به فرمت arff و معادل نمایش مقادیر خصیه‌های آن‌ها با نمادین جدول ارائه شده.

- **بدون هرس کردن درخت تصمیم گیري مرتبط با این داده هاي آموزشي را یاد بگیرید.**

داده‌ها بعد از بارگذاری شدن در نرم‌افزار وکا با الگوریتم J.48 که همان معادل الگوریتم C4.5 می‌باشد به اجرا آورده شد، نتیجه‌ی درخت حاصله در شکل ۱ آمده است.



شکل۱: درخت تصمیم ساخته شده از داده‌های جدول ۱ – بدون اعمال هرس

- **درخت تصمیم گیري یادگرفته شده را بر روي دو نمونهٔ آزمون جدول ۲ آزمایش نمائید.**

نمره‌ی نهایی پیش‌بینی شده	project	resrch	asgnmnt	finex	midex	hstudy	bstudy	apresent
افتاده ≡ FAILED	خوب	خوب	خوب	خوب	زیر۵۰	خوب	خوب	خوب
متوسط ≡ AVG	خوب	متوسط	خوب	متوسط	خوب	خوب	متوسط	متوسط

جدول۲: نتایج آزمون داده‌های تست با درخت شکل ۱

- **درخت را با استفاده از قوانین نمایش دهید.**

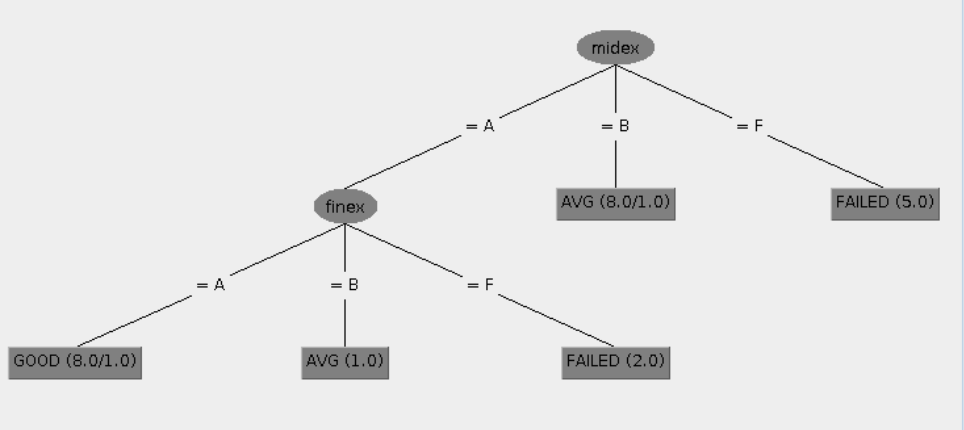
در جدول قوانین استخراج شده از درخت شکل ۱ در جدول ۳ به صورت اجزای مقدم و تالی مشخص گردیده‌اند که این قوانین را می‌توان به‌صورت مستقیم از درخت تصمیم استخراج کرد.

- **درخت تصمیم گیري را با استفاده از هرس کردن بیاموزید و نتیجهٔ آن را روي داده هاي آزمون بررسي کنید.**

درخت تصمیم حاصل با اعمال هرس به صورت شکل ۲ بدست آمد. همانطور که مشاهده می‌شود گره «مطالعه‌ی هفتگی کتاب(bstudy)» حذف گردیده است. در این درخت نیز اگر داده‌ها تست را بیازماییم به همان نتایج جدول ۲ می‌رسیم که در جدول ۴ آمده است.

مقدم	تالی
میان نیمسال = زیر-۵۰	افتاده
میان نیمسال = متوسط و **مطالعه‌ی هفتگی کتاب** = خوب	خوب
میان نیمسال = متوسط و **مطالعه‌ی هفتگی کتاب** = متوسط	متوسط
میان نیمسال = متوسط و **مطالعه‌ی هفتگی کتاب** = ضعیف	متوسط
میان نیمسال = خوب و **پایان نیمسال** = خوب	خوب
میان نیمسال = خوب و **پایان نیمسال** = متوسط	متوسط
میان نیمسال = خوب و **پایان نیمسال** = زیر-۵۰	افتاده

جدول۳: نمایش درخت تصمیم شکل ۱ به صورت قوانین



شکل۲: درخت تصمیم ساخته شده از داده‌های جدول ۱ – با اعمال هرس

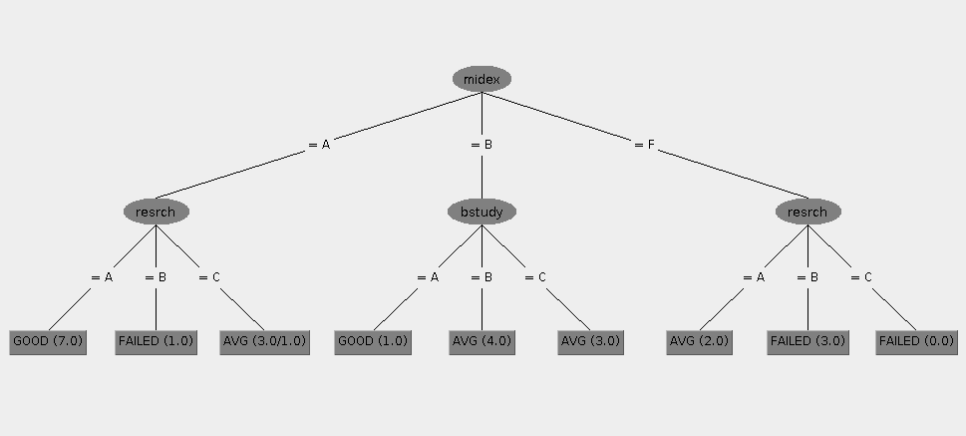
نمره‌ی نهایی پیش‌بینی شده	projct	resrch	asgnmnt	finex	midex	hstudy	bstudy	apresent
افتاده ≡ FAILED	خوب	خوب	خوب	خوب	زیر-۵۰	خوب	خوب	خوب
متوسط ≡ AVG	خوب	متوسط	خوب	متوسط	خوب	خوب	متوسط	متوسط

جدول۴: نتایج آزمون داده‌های تست با درخت شکل ۲

- **خودتان با تغييراتي كه لازم مي دانيد اثر بيش پوشش را بررسي كرده و نشان دهيد كه چرا بيش پوشش انجام گرفته است و چگونه مي توان جلوي آن را گرفت.**

در درخت تصمیم یکی از مواقعی که بیش‌پوشش رخ می‌دهد داده‌های خطادار به‌گونه‌ای باشند که در طی یادگیری متغییر هدف علاوه‌بر اینکه از فرضیه‌ی هدف منحرف شده‌ایم، عمق درخت نیز زیاد شود. به عنوان مثال اگر داده‌ها را بدین‌گونه تغییر دهیم که مقدار ۳ فرد اول که «افتاده» برچسب خورده‌اند را تغییر دهیم درخت حاصل از این تغییرات به صورت شکل ۳ بدست می‌آید. همان‌طور که در شکل ۳ مشاهده می‌شود شکل درخت فقط با دست‌کاری ۳ رکورد به‌کل تغییر پیدا کرد و چندین شاخه جدید بوجود آمد. و علت این بیش‌پوشش این است که چون داده‌ها دارای اغتشاش هستند مسیر رسیدن به فرضیه‌ی هدف گم می‌شود و الگوریتم سعی می‌کند به هر ترتیب که شده درختی با بهترین برازش برای این مجموعه داده بیابد بنابرین در نهایت به درختی می‌رسد که با داده‌های خطادار حداکثر همخوانی را دارد که باعث ایجاد بیش‌پوشش شده است. برای هرس کردن چند روش وجود دارد که یکی از آن‌ها این است که یکی از گره‌ها را با زیردرخت آن را حذف کنیم و مقدار محتمل‌ترین برچسب در

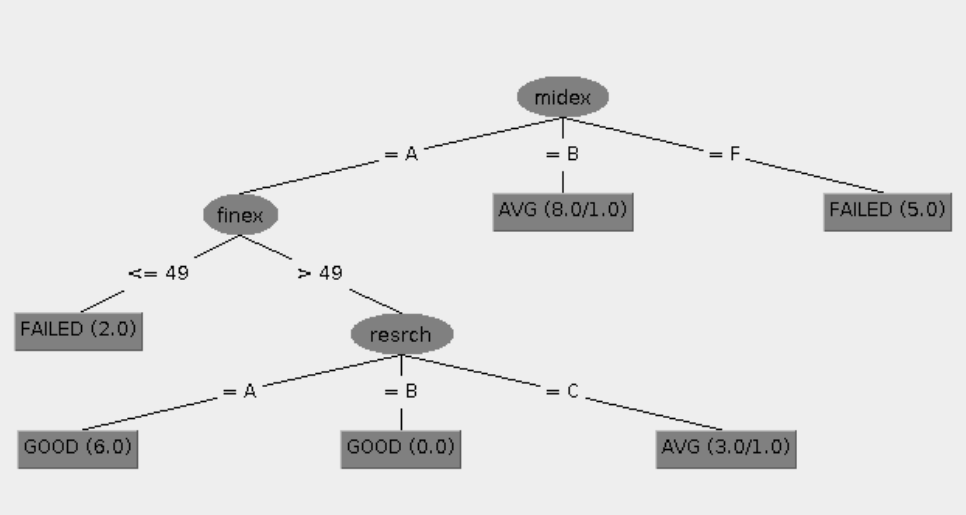
آن گره و زیردرختش را به عنوان برچسب گره در نظر بگیریم. و روش دیگر برای اجتناب از بیش‌پوشش حذف یکی از قوانین حاصله از درخت‌تصمیم که در واقع حذف یک مسیر از درخت‌تصمیم می‌باشد. در هرکدام از روش‌ها تا زمانی که میزان خطا افزایش پیدا نکرده است اقدام به حذف می‌کنیم و زمانی که خطای حاصله از آزمون درخت بعد از حذف گره یا مسیر افزایش پیدا کرد عمل هرس کردن را متوقف می‌کنیم.



شکل۳: درخت تصمیم ساخته شده از داده‌های جدول ۱ – بدون اعمال هرس و با دست‌کاری در داده‌ها

- **به دلخواه خود يكي از خصيصه ها را تبديل به خصيصة پيوسته نمائيد و دوباره درخت تصميم گيري را ياد بگيريد.**

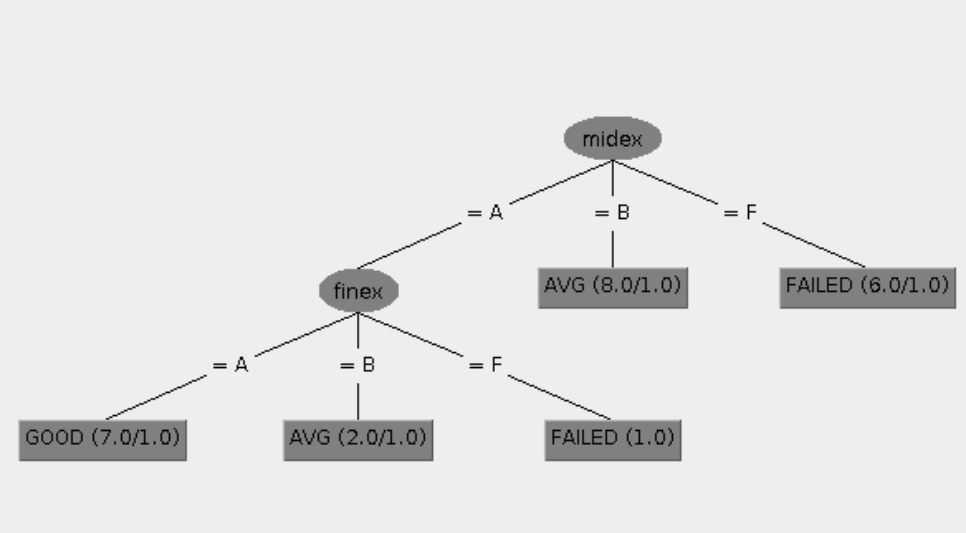
بنده ویژگی نمره‌ی پایان نمیسال رو پیوسته کردم و درخت بدست آمده بعد از پیوسته کردم این ویژگی در شکل ۴ آمده است، مقدار این ویژگی از ۰ الی ۱۰۰ براساس درجه‌ای که قبلا داشته نمره گذاری شده است. همان‌طور که می‌بینیم الگوریتم بخوبی توانسته است که بر اساس این ویژگی پیوسته درخت را دوباره تشکیل دهد، که البته تاحدودی متفاوت‌تر از درخت‌های پیشین بدست آمد.



شکل۴: درخت تصمیم ساخته شده از داده‌های جدول ۱ – با اعمال هرس و با ویژگی «پایان نمیسال» پیوسته

- **اثر داده هاي خطادار و يا خصيصه هاي بدون مقدار را بررسي و گزارش كنيد.**

بنده مقدار خصیصه‌های «میان نیمسال : داده‌ی ۶ام» و «پایان نیمسال : داده‌ی ۹ام» را به ترتیب از $A \to F$ و $F \to B$ تغییر دادم و درخت حاصل از داده‌های مغشوش شده را در شکل ۵ آورده شده است. درخت بدست آمده با داده‌های خطادار مشابه درخت با داده‌های بدون خطا شکل ۲ می‌باشد و این یعنی الگوریتم توانسته با داده‌های خطادار بخوبی کنار بیاید و مفهوم هدف را دست ندهد ولی اگر به میزان خلوص گره‌های برگ توجه کنیم می‌بینیم میزان خلوص گره‌ها در درخت داده‌های خطا متفاوت‌تر از میزان خلوص با داده‌ها دست نخورده می‌باشد.



شکل۵: درخت تصمیم ساخته شده از داده‌های جدول ۱ – با اعمال هرس و با داشتن داده‌های خطادار

۲ قسمت دوم تکلیف

- **الف) چه تغييراتي الزم است كه در داده هاي آموزشي داده شود تا بتوان فرضيه اي عطفي بوسيلة الگوريتمهاي Find-S يا حذف‌نامزد يادگرفت؟**

الگوریتم‌های Find-S یا حذف‌نامزد فقط توانایی تفکیک نمونه‌های مثبت و منفی را دارد در نتیجه چون خصیصه‌ی هدف داده‌ها دارای ۳ مقدار { خوب، متوسط و افتاده } می‌باشد این دو الگوریتم توانایی یادگیری براساس این ۳ خصیصه هدف ندارد پس ما برای اینکه بتوانیم این داده‌ها را با این دو الگوریتم آموزش دهیم باید نمونه‌ها را به صورت ۲ مقدار قبول‌شده(+) و افتاده(−) دسته‌بندی کنیم. بدین منظور در داده‌ها مقدار هر خصیصه‌ای که «افتاده» علامت‌گذاری نشده‌اند را + و مابقی را − علامت‌گذاری می‌کنیم.

- **ب) پس از انجام تغييرات داده شده، الگوريتم Find-S چه فرضيه اي را ياد مي گيرد؟**

بعد از اعمال تغییرات لازم جهت یادگیری داده‌ها توسط الگوریتم Find-S داده‌هایی که با مقدار + علامت‌گذاری شده اند(کسانی که قبول شده‌اند) را به الگوریتم Find-S می‌دهیم که روند بدست آمدن فرضیه در جدول ۵ آمده است، همان‌طور که می‌بینیم الگوریتم Find-S نتوانست فرضیه‌ی درستی به دست بدهد و درست بعد از دریافت ۴ مثال + به عام‌ترین فرضیه

ممکن رسید که در این مثال خاص به معنی شکست کامل الگوریتم می‌باشد.

projct	resrch	asgnmnt	finex	midex	hstudy	bstudy	apresent	فرد
∅	∅	∅	∅	∅	∅	∅	∅	–
خوب	خوب	خوب	خوب	خوب	خوب	خوب	خوب	۱
؟	؟	؟	خوب	خوب	خوب	خوب	خوب	۲
؟	؟	؟	؟	؟	؟	؟	خوب	۴
؟	؟	؟	؟	؟	؟	؟	؟	۶

جدول۵: روند ایجاد خاص‌ترین فرضیه به ازای هر نمونه با مقدار هدف +

- **ج) این فرضیه داده هاي آزمون را چگونه دسته بندي مي نماید؟**

باتوجه به فرضیه‌ی بدست آمده که در سطر آخر جدول ۵ نشان داده شده است، فرضیه‌ی بدست آمده هردوی داده‌های تست را به عنوان مثال + دسته‌بندی خواهد کرد – که طبق دسته‌بندی‌ای که درخت تصمیم ارائه داد درست یکی از این نتایج درست نمی‌باشد.