گزارش پروژه سوم درس داده‌کاوی

پرهام کاظمی　　علیرضا پرچمی
۹۴۳۶۱۱۰۴۳۰۱۸　　؟؟۹۴۳۶

۲۸ دی ۱۳۹۷

مقدمه

هدف از این پروژه، استفاده از الگوریتم‌های دسته‌بندی [1] برای داده‌کاوی و کشف حقایق در مجموعه‌داده‌ی Mushroom می‌باشد.

در ابتدا، با استفاده از الگوریتم K-Folds Cross-Validation، مجموعه‌داده، ۱۰ بار به مجموعه‌داده‌های آموزش و آزمایش قطعه‌بندی شده است. سپس، با تشکیل درخت‌های ID3 و CART، قوانین مورد استفاده برای دسته‌بندی استخراج و دقت و صحت درخت‌ها به کمک معیارهای F-Measure و Presicion و Recall محاسبه شده‌اند.

در نهایت، الگوریتم دسته‌بندی K نزدیک‌ترین همسایه [2] بر روی مجموعه‌داده‌های آموزش و آزمایش – که با روش hold out تقسیم‌بندی شده‌اند – اجرا شده و دقت دسته‌بند با توجه به معیارهای F-Measure و Presicion و Recall محاسبه شده است.

۱ ابزارهای استفاده شده

در پیاده‌سازی این پروژه، از کتاب‌خانه‌های زیر در زبان پایتون استفاده شده است:

jupyter برای پیاده‌سازی و استفاده از الگوریتم‌های موجود در کتاب‌خانه‌ها در محیطی مناسب.

[1] classification
[2] K-Nearest Neigbours

scikit-learn شامل پیاده‌سازی الگوریتم‌های تولید درخت‌های تصمیم‌گیری و KNN و همین‌طور محاسبه‌ی معیارهای اندازه‌گیری دقت دسته‌بندهای به‌دست‌آمده.

pandas جهت خواندن داده‌ها از فایل و آماده‌سازی و پیش‌پردازش آن‌ها.

graphviz برای نمایش گراف‌ها و درخت‌های تولید شده و ذخیره‌ی خروجی در فایل pdf.

۲ مجموعه‌داده

توضیحات دیتاست

۳ درخت تصمیم

درخت‌های تصمیم، نوعی از دسته‌بندها می‌باشند که با تقسیم‌بندی‌های متوالی مجموعه‌داده در هر گره و تصمیم در یال‌های درخت، کلاس هر نمونه‌ی ورودی را تعیین می‌کنند. در این پروژه، از دو روش استفاده از آنتروپی (درخت‌های ID3) و معیار GINI (در درخت CART)، دو دسته‌بند به دست آمده و از نظر دقت با هم مقایسه شده‌اند.

۱.۳ پیش‌پردازش داده‌ها

در این مجموعه‌داده، ستون ۱۱اُم که بیانگر ویژگی stalk-root است، دارای مقادیر گم‌شده می‌باشد. برای رفع این مشکل، در نمونه‌هایی که دارای مقدار ناقص برای این ویژگی می‌باشند، مقدار «؟» با مُد داده‌های موجود در این ستون جایگزین شده‌اند. دلیل این کار، اسمی بودن ویژگی‌های موجود می‌باشد. بدین منظور، قطعه‌کد زیر با استفاده از کتاب‌خانه pandas اجرا شده است:

```
 import pandas as pd
m11 = data.mode()['stalk-root'][0]
data.loc[data['stalk-root'] == '?', 'stalk-root'] = m11
```

در این قطعه‌کد، ابتدا فراوان‌ترین مقدار ویژگی مورد نظر در متغیر m11 ذخیره شده و سپس مقادیر مشخص شده با «؟»، توسط مُد به‌دست‌آمده جایگزین شده‌اند.

۲.۳ تقسیم‌بندی مجموعه‌داده

برای تقسیم‌بندی مجموعه‌داده به دو دسته‌ی آموزش و آزمایش، از روش K-Folds Cross-Validation استفاده شده است. در این روش، مجموعه‌داده در ابتدا به K مجموعه‌ی کوچک‌تر تقسیم شده و برای ایجاد هر مدل، یکی از زیرمجموعه‌ها به عنوان داده‌ی آزمایش و سایر داده‌های موجود، برای آموزش مدل مورد استفاده قرار می‌گیرند. برای تولید درخت‌های تصمیم‌گیری، پارامتر K برابر ۱۰ فرض شده است. الگوریتم K-Folds در کتاب‌خانه‌ی scikit-learn به صورت زیر استفاده می‌شود:

```
 from sklearn.model_selection import KFold
sets = KFold(n_splits=10)
```

۳.۳ ایجاد درخت ID3

در کتاب‌خانه‌ی scikit-learn، درخت‌های تصمیم‌گیری با استفاده از کلاس DecisionTreeClassifier تولید می‌شوند. در ورودی تابع سازنده این کلاس، پارامتر criterion نوع درخت مورد نظر را تعیین می‌کند. برای ایجاد درخت ID3، مقدار 'entropy' به این پارامتر تخصیص داده می‌شود. سپس با فراخوانی متدهای fit و predict، به ترتیب داده‌های آموزش و آزمایش را در اختیار الگوریتم قرار می‌دهیم:

```
 dt = DecisionTreeClassifier(criterion='entropy')
dt.fit(X_train, y_train)
y_pred = dt.predict(X_test)
```

سپس با استفاده از کتاب‌خانه‌ی scikit-learn، دقت کلاس‌های به‌دست‌آمده برای مجموعه‌داده‌ی آزمایش (y_pred) محاسبه می‌شود:

```
 from sklearn.metrics import precision_recall_fscore_support
presicion, recall, f_measure, _ =
precision_recall_fscore_support(y_test, y_pred,
average='micro')
```

درخت حاصل و همین‌طور دقت‌های به‌دست‌آمده در فایل خروجی (decision_trees.html) قابل مشاهده می‌باشند.