هوش مصنوعی

تمرین شماره‌ی ۳

مدرسین: **دکتر فدایی و دکتر یعقوب‌زاده**

طراحان: جواد کاویان، محمد امانلو، مهدی نائینی

مهلت تحویل: جمعه ۲۶ اردیبهشت ۱۴۰۴، ساعت ۲۳:۵۹

مقدمه

هدف این تمرین، آشنایی با روش‌های یادگیری ماشین[1] برای پیش‌بینی نمره نهایی درس هوش مصنوعی دانشجویان بر اساس ویژگی‌های جمعیتی، اجتماعی مربوط به دانشگاه است. این پروژه شامل دوفاز اصلی است: ابتدا در فاز اول به آماده‌سازی محیط و داده‌ها می‌پردازیم تا زیرساخت‌های لازم برای تحلیل و مدل‌سازی را فراهم کنیم. در فاز دوم ، مدل‌های یادگیری ماشین توسعه داده شده، آموزش می‌بینند و با استفاده از معیارهای ارزیابی مناسب مورد سنجش قرار می‌گیرند تا دقت و عملکرد آن‌ها در پیش‌بینی نمره نهایی دانشجویان ارزیابی شود.

[1] Machine Learning

آشنایی با مجموعه داده

مجموعه داده‌ای که در اختیار شما قرار دارد شامل اطلاعات مربوط به عملکرد تحصیلی دانشجویان است. در این مجموعه، علاوه بر نمرات درس هوش مصنوعی، ویژگی‌های جمعیتی، اجتماعی و تحصیلی هر دانشجو (مانند سن، جنسیت، وضعیت خانوادگی، ساعات مطالعه و ...) گردآوری شده است.

توضیحات مربوط به ستون‌های این مجموعه داده در جدول زیر ارائه شده است:

نام ستون	توضیح
university	دانشگاه محل تحصیل دانشجو (PR – Princeton University یا CM – Carnegie Mellon University)
sex	جنسیت دانشجو (F – دختر یا M – پسر)
age	سن دانشجو (از 15 تا 22)
address	نوع محل سکونت دانشجو (U – شهری یا R – روستایی)
motherEducation	تحصیلات مادر (0 – بدون تحصیلات، 1 – ابتدایی (پایان کلاس چهارم)، 2 – کلاس پنجم تا نهم، 3 – متوسطه، 4 – عالی)
fatherEducation	تحصیلات پدر (0 – بدون تحصیلات، 1 – ابتدایی (پایان کلاس چهارم)، 2 – کلاس پنجم تا نهم، 3 – متوسطه، 4 – عالی)
motherJob	شغل مادر (teacher – معلم، health – حوزه بهداشت، services – خدمات شهری/اداری/پلیس، at_home – خانه‌دار، other – سایر)
fatherJob	شغل پدر (teacher – معلم، health – حوزه بهداشت، services – خدمات شهری/اداری/پلیس، at_home – خانه‌دار، other – سایر)
reason	دلیل انتخاب این دانشگاه (home – نزدیک به خانه، reputation – شهرت دانشگاه، course – علاقه به رشته، other – سایر)
travelTime	زمان رفت‌وآمد از خانه تا دانشگاه (1 – کمتر از 15 دقیقه، 2 – بین 15 تا 30 دقیقه، 3 – بین 30 تا 60 دقیقه، 4 – بیشتر از 1 ساعت)

studyTime	زمان مطالعه هفتگی (1 – کمتر از 2 ساعت، 2 – بین 2 تا 5 ساعت، 3 – بین 5 تا 10 ساعت، 4 – بیشتر از 10 ساعت)
failures	تعداد مردودی‌های گذشته (n اگر n بین 1 تا 3، در غیر این صورت 4)
universitySupport	دریافت حمایت آموزشی اضافی (yes – بله یا no – خیر)
paid	کلاس‌های اضافی پولی (yes – بله یا no – خیر)
higher	تمایل به ادامه تحصیلات عالی (yes – بله یا no – خیر)
internet	دسترسی به اینترنت در خانه (yes – بله یا no – خیر)
romantic	داشتن رابطه عاشقانه (yes – بله یا no – خیر)
freeTime	زمان آزاد پس از دانشگاه (1 – بسیار کم تا 5 – بسیار زیاد)
goOut	میزان تفریح با دوستان (1 – بسیار کم تا 5 – بسیار زیاد)
Dalc	مصرف الکل در روزهای کاری (1 – بسیار کم تا 5 – بسیار زیاد)
Walc	مصرف الکل در تعطیلات آخر هفته (1 – بسیار کم تا 5 – بسیار زیاد)
absences	تعداد غیبت‌های کلاس‌های درس (از 0 تا 93)
EPSGrade	نمره درس آمار و احتمالات (از 0 تا 20)
DSGrade	نمره درس علوم داده (از 0 تا 20)
finalGrade	نمره نهایی (عددی: از 0 تا 20)

پیش‌پردازش دادگان و مهندسی ویژگی‌ها

مهم‌ترین بخش هر پروژه یادگیری ماشین، بخش پیش‌پردازش داده‌ها می‌باشد. در این فاز فرمت داده‌ها را تغییر داده، اصلاح یا خلاصه می‌کنیم. چرا که در دنیای واقعی اطلاعات جمع آوری شده به راحتی کنترل نمی‌شوند و در نتیجه مقادیر خارج از محدوده، ناممکن، از دست رفته و به طور کلی گمراه کننده در دادگان وجود دارند. این فاز باعث می‌شود مدل کارا تری را توسعه دهیم.

برای انجام کامل این بخش، مراحل زیر توصیه می‌شود.

- ابتدا داده‌های از دست رفته را باید تکمیل کنیم، برای انجام این کار روش‌های مختلفی وجود دارد. با توجه به شناختی که در بخش قبلی بدست آوردید 3 روش که به نظر شما برای هر ویژگی مناسب تر است را انتخاب و اجرا کنید.
- در صورتی که امکان حذف برخی از ستون‌ها وجود دارد، آن‌ها را حذف کنید. دلیل آن را توضیح دهید.
- در صورت نیاز، داده‌های عددی را از ستون‌های متنی استخراج کنید.
- و داده‌های دسته‌بندی‌شده[2] را به مقادیر عددی تبدیل کنید تا برای مدل‌های یادگیری ماشین قابل استفاده باشند.

در گام بعدی هدف این است که دانشجویان را بر اساس نمره نهایی (finalGrade) به چهار گروه دسته‌بندی کنیم. دلیل تبدیل ستون «finalGrade» به یک ویژگی دسته‌ای این است که قرار است مدل‌های دسته‌بندی[3] آموزش داده شوند. مدل‌های دسته‌بندی برای پیش‌بینی متغیرهای دسته‌ای طراحی شده‌اند و قادر به پردازش داده‌های عددی پیوسته مانند نمره نیستند. بنابراین، برای استفاده از این مدل‌ها، باید داده‌های پیوسته را به گروه‌های مشخص تقسیم کرد. در این حالت، نمره‌های نهایی بر اساس یک آستانه مشخص به چهار دسته نمرات گروه الف (بالاتر از 17)، نمرات گروه ب (بین 14 تا 17)، نمرات گروه ج (بین 10 تا 14) و مردودی (نمرات زیر 10) تقسیم می‌شوند. به این ترتیب، داده‌ها به صورت دسته‌بندی‌شده تبدیل می‌شوند که برای مدل‌های classification مناسب است.

---

[2] Categorical

[3] Classification

توسعه، آموزش و ارزیابی مدل‌ها

در این بخش، هدف اصلی طراحی، آموزش و ارزیابی مدل‌های یادگیری ماشین، برای حل مسئله Classification است. این فرآیند به‌گونه‌ای تنظیم شده است که شما تمامی مراحل، از آماده‌سازی داده‌ها گرفته تا توسعه و ارزیابی مدل‌های پیشرفته، را به‌صورت عملی تجربه کنید. در ادامه، توضیحاتی کلی درباره این مراحل ارائه شده است:

1. **Train-Test Split**

Train-Test Split روشی برای تقسیم داده‌ها به دو مجموعه آموزش (train)، تست (test) است. این تقسیم معمولاً برای ارزیابی عملکرد مدل‌ها استفاده می‌شود و از ایجاد overfitting جلوگیری می‌کند، زیرا مدل فقط روی داده‌های آموزش آموزش می‌بیند و عملکرد آن روی داده‌های تست ارزیابی می‌شود. داده‌ها باید به سه بخش تقسیم شوند: 80٪ برای آموزش و 20٪ برای آزمون. این کار برای ارزیابی عملکرد مدل‌ها ضروری است و به جلوگیری از Overfitting کمک می‌کند.

2. **Normalization/Standardization**

Normalization/Standardization نقش مهمی در بهبود عملکرد مدل‌ها دارند. در این بخش، شما باید اهمیت این مرحله را بررسی کرده و روش‌های مختلف آن را بررسی کنید. دلیل استفاده از روش مورد نظر برای این مرحله را بیان کنید. (دقت کنید عملیات scaling با استفاده از شاخصه‌های داده آموزش ساخته شده و پس از ساخته شدن تنها روی دادگان ارزیابی و تست اعمال می‌شود.)

همچنین در صورت نیاز از هرگونه Transform عددی یا مقداری نیز می‌توانید استفاده کنید. (در این بخش استفاده از عملیات لگاریتم یا ترکیب ستون‌ها و تبدیل ویژگی‌های مستقل به ویژگی‌هایی با ارتباط موثرتر با متغیر وابسته حائز اهمیت است.)

3. **Sklearn Models**

- **Naive Bayes**

الگوریتم Naive Bayes یک الگوریتم یادگیری ماشین ساده و قدرتمند است که مبتنی بر قانون بیز است. این الگوریتم برای مسائل دسته‌بندی مورد استفاده قرار می‌گیرد که بر اساس آن فرض می‌شود ویژگی‌ها مستقل از هم هستند، حتی اگر در واقع اینگونه نباشند. به همین دلیل نام این الگوریتم Naive یا ساده است.

با استفاده از کتابخانه‌های موجود این الگوریتم را پیاده سازی اجرا و مطابق بخش ارزیابی، ارزیابی کنید.

- **Decision Tree**

درخت تصمیم یک مدل پیش‌بینی است که از ساختار درختی برای تصمیم‌گیری در مورد مقدار یک متغیر هدف استفاده می‌کند. این درخت از گره‌ها و لیستی از تقسیم‌ها تشکیل شده است که به ازای هر گره، یک متغیر و یک مقدار تقسیم‌بندی انتخاب می‌شود تا داده‌ها به گره‌های فرزند تقسیم شوند. این فرآیند ادامه پیدا می‌کند تا ویژگی‌های مهم مجموعه داده درخت تصمیم را تشکیل دهند. هدف نهایی این است که با استفاده از این درخت، می‌توان پیش‌بینی‌هایی در مورد داده‌های جدید انجام داد. درخت تصمیم به دلیل قابل فهم بودن ساختار و نتایج آن، یکی از محبوب‌ترین روش‌های یادگیری ماشین است.

با استفاده از کتابخانه‌های موجود این الگوریتم را پیاده سازی اجرا و مطابق بخش ارزیابی، ارزیابی کنید. در صورت نیاز از Prune کردن درخت استفاده کنید. سعی کنید فراپارامترهای[4] درخت را بهینه کنید و در نهایت درخت بدست آمده رسم کنید. (می‌توانید از کتابخانه Plot_tree استفاده کنید.)

مهمترین ویژگی‌ها را از نظر مدل درخت تصمیم بر اساس درخت رسم شده بیان کنید. همچنین feature_importance تمامی ویژگی‌ها را از مدل ایجاد شده دریافت کنید و آن‌ها را تحلیل کنید.

- **Random Forest**

روش‌های Ensemble در یادگیری ماشین به مجموعه‌ای از مدل‌ها اشاره دارند که به صورت همکاری برای بهبود دقت پیش‌بینی‌ها کار می‌کنند. این روش‌ها معمولاً با ترکیب چندین مدل ساده‌تر، مدل نهایی را می‌سازند که در مجموع از هر یک از مدل‌های تکی بهتر عمل می‌کند. دو روش اصلی در متدهای Ensemble وجود دارد: Bagging و Boosting به منظور کاهش واریانس مدل‌ها استفاده می‌شود و در آن چندین نمونه از داده‌ها به طور تصادفی انتخاب شده و برای هر نمونه یک مدل ساخته می‌شود. این مدل‌ها سپس ترکیب می‌شوند تا نتیجه نهایی حاصل شود.

جنگل تصادفی یکی دیگر از روش های یادگیری جمعی است که بر اساس ایده ای از تجمع از قوانین یا الگوریتم های ساده تر، به صورت تصادفی، تعدادی از مدل های یادگیری خود را اجرا می کند و سپس از ترکیب نتایج حاصل از این مدل ها برای پیش بینی مقادیر جدید استفاده می کند. در واقع، جنگل تصادفی یک مجموعه از درخت های تصمیم است که هر کدام به صورت مستقل از دیگری آموزش

[4] HyperParameters

داده می شوند و سپس نتایج آن ها ترکیب می شوند تا یک پیش بینی نهایی برای داده های ورودی انجام شود. این روش برای حل مسائل پیچیده و تعداد زیادی داده بسیار موثر و کارآمد است.
با استفاده از کتابخانه‌های موجود این الگوریتم را پیاده سازی اجرا و مطابق بخش ارزیابی، ارزیابی کنید. برای بهینه کردن فراپارامترهای درخت از RandomizedSearchCV استفاده کنید.

- **XGBoost**

XGBoost یک الگوریتم یادگیری ماشین است که بر پایه روش های گرادیان کاهشی است. این الگوریتم برای حل مسائل مختلف یادگیری ماشین از جمله طبقه بندی، پیش بینی و رتبه بندی مورد استفاده قرار می گیرد. XGBoost قابلیت اجرای سریع، کارایی بالا و افزایش دقت در پیش بینی ها را دارا می باشد.
با استفاده از کتابخانه‌های موجود این الگوریتم را پیاده سازی اجرا و مطابق بخش ارزیابی، ارزیابی کنید. برای بهینه کردن فراپارامترهای درخت از GridSearchCV استفاده کنید. برای سادگی بیشتر تنها به بهینه کردن فرا پارامترهای زیر بپردازید:

learning rate: نرخ تغییر وزن‌ها در هر گام.
n estimates: تعداد مدل‌های پایه.
min_samples_split: حداقل تعداد نمونه‌ها برای تقسیم یک گره.
min_samples_leaf: حداقل تعداد نمونه‌ها برای برگ‌ها.

max_depth: حداکثر عمق درخت‌های تصمیم.
max_features: حداکثر تعداد ویژگی‌ها برای هر تقسیم گره.

4. Decision Tree from Scratch

در این بخش، از شما انتظار می رود که یک کلاس درخت تصمیم را به صورت from scratch پیاده سازی کنید. پیاده سازی کلاس این مدل، باید بدین صورت باشد که حتما دارای سه متد زیر باشد:

1. متد init : در اینجا آرگومان هایی که فکر می کنید ممکن است مدل نیاز داشته باشد را جهت instanciate کردن مدل، تعریف کنید. یکی از آرگومان هایی که حتما نیاز است تعریف شود، آرگومان max_depth است؛ این آرگومان مشخص می کند که درخت تا چه عمقی ساخته شود؛ به عنوان مثال اگر عمق درخت 4 باشد، در یک پیمایش از ریشه درخت، با طی کردن 4 گره از درخت، به برگ می رسیم

که در آن، برای خروجی دادن، از عملیات majority vote استفاده می کنیم؛ منظور از این عملیات، این است که در آن زیر درخت خاص(که در این مثال با پیمایش 4 ویژگی به آن رسیدیم)، به ازای هر برچسب تعداد نمونه ها را می شماریم؛ به عنوان مثال اگر در این زیردرخت 5 نمونه از برچسب 0 و 7 نمونه از برچسب 1 داشته باشیم، در اینجا برچسب داده تست، 1 خواهد بود.

2. متد fit : ورودی این متد، X_train و y_train یا به عبارتی داده ها و برچسب آن ها است. این متد باید درخت تصمیم را بر اساس داده های آموزشی بسازد؛ منظور از ساخت درخت تصمیم، مشخص کردن این است که در هر سطح از ارتفاع درخت، کدام ویژگی ها وجود داشته باشند و همچنین چگونه هر ویژگی، درخت را به زیر درخت های کوچک تر تقسیم می کند. این تصمیم گیری را همانطور که در درس با آن آشنا شدید، با استفاده از معیار آنتروپی انجام دهید؛ به عبارتی، از بین تمامی کاندیداهای ویژگی، ویژگی ای را انتخاب کنید منجر به بیشترین کاهش در آنتروپی می شود.
3. متد predict : این متد بعد از اینکه درخت ساخته شد، وظیفه این را دارد که درخت را پیمایش کند تا به برگ برسد و برچسب داده را خروجی دهد.

توجه کنید که شما کاملا آزاد هستید که به هر نحو و با استفاده از هر ساختمان داده ای این درخت را پیاده سازی کنید؛ اما یک ایده، می تواند استفاده از ساختار بازگشتی باشد؛ به عبارتی درخت تصمیم، می تواند چند مشخصه(attribute) به صورت درخت داشته باشد و آن ها را با استفاده از معیار آنتروپی بسازد؛ تاکید می شود که هیچ الزامی به استفاده از این ساختار نیست.

5. Comparison with Library Implementation:

در این بخش عملکرد الگوریتم Decision Tree from Scratch با نسخه آماده آن در کتابخانه Scikit-learn مقایسه می‌کنید. این مقایسه به شما کمک می‌کند تا درک بهتری از مفاهیم و جزئیات الگوریتم‌های درختی پیدا کرده همچنین تفاوت‌ها و شباهت‌های بین پیاده‌سازی دستی و نسخه کتابخانه‌ای را بهتر درک کنید.

در نهایت به عنوان داده پیش‌بینی نهایی، اطلاعات مربوط به خودتان یا دوستانتان را به عنوان ورودی به مدلی که بیشترین دقت را روی دادگان تست گرفته است وارد کرده و نمره خود در درس هوش مصنوعی را پیش‌بینی کنید.
دقت پیش‌بینی خود را در پایان ترم متوجه خواهید شد. 😀

ارزیابی مدل‌ها

معیارهای زیادی برای سنجش و ارزیابی عملکرد مدلها وجود دارد. ارزیابی مدل های دسته بندی در یادگیری ماشینی به معنای ارزیابی عملکرد و کارایی مدل های مختلف است که برای دسته بندی داده ها استفاده می شوند. ارزیابی مدل های دسته بندی از اهمیت بسیاری برخوردار است زیرا به ما کمک می کند تا بتوانیم مدلی که می‌سازیم را با دقت بیشتری پیشرفت دهیم و اطمینان حاصل کنیم که عملکرد آن بهینه است.

با استفاده از این معیارها و ارزیابی کننده های دیگر می توان مدل های دسته بندی را مقایسه کرده و انتخاب بهترین مدل را برای مسئله خاص خود انجام داد.

برای ارزیابی مناسب از معیارهای زیر استفاده نمایید:

- ماتریس درهم‌ریختگی[5]
- Recall
- F1-Score
- Precision
- Accuracy
- میانگین‌گیری Macro و Micro و Weighted

مطالعه این دو لینک (لینک۱ و لینک۲) برای درک معیارهای فوق به شما کمک خواهد کرد.

[5] Confusion matrix

نکات پایانی

- دقت کنید که کد شما باید به نحوی زده شده باشد که نتایج قابلیت بازتولید داشته باشند.
- توضیحات مربوط به هر بخش از پروژه را بطور خلاصه و در عین حال مفید در گزارش خود ذکر کنید. از ابزارهای تحلیل داده مانند نمودارها استفاده کنید. حجم توضیحات گزارش شما هیچ گونه تاثیری در نمره نخواهد داشت و تحلیل و نمودارهای شما بیشترین ارزش را دارد.
- درباره هر بخش از مراحل پروژه می‌بایست علل استفاده یا عدم استفاده از هر الگوریتم، مزایا و معایب، عملکرد، فرا پارامترها و وضعیت خروجی‌ها را بطور دقیق مطالعه کنید. از این موضوعات در زمان تحویل پرسیده خواهد شد.
- سعی کنید از پاسخ‌های روشن در گزارش خود استفاده کنید و اگر پیش‌فرضی در حل سوال در ذهن خود دارید، حتما در گزارش خود آن را ذکر نمایید.
- پس از مطالعه کامل و دقیق صورت پروژه، در صورت وجود هرگونه ابهام یا سوال با طراحان پروژه در ارتباط باشید.
- نتایج، گزارش و کدهای خود را در قالب یک فایل فشرده با فرمت AI_CA3_[stdNumber].zip در سامانه ایلرن بارگذاری کنید. به طور مثال AI_CA3_810101999.zip
- محتویات پوشه باید شامل فایل پاسخ‌های شما به سوالات کتبی، فایل jupyter-notebook، خروجی html و فایل‌های مورد نیاز برای اجرای آن باشد. از نمایش درست خروجی‌های مورد نیاز در فایل html مطمئن شوید.
- توجه کنید این تمرین باید به صورت تک‌نفره انجام شود و پاسخ‌های ارائه شده باید نتیجه فعالیت فرد نویسنده باشد. در صورت مشاهده تقلب به همه افراد مشارکت‌کننده، نمره تمرین 100- و به استاد نیز گزارش می‌گردد. همچنین نوشته نشدن کدها توسط هوش مصنوعی نیز بررسی می‌شود!
- نیازی به نوشتن پاسخ سوالاتی که نیاز به پیاده‌سازی کد برای انجام ندارند در نوتبوک نهایی پروژه نیست. از این موارد و موارد دیگری که ممکن است در تمرین ذکر نشده باشد (مثل علل استفاده از هر روش، مزایا و معایب انجام هر مرحله از پروژه، مفاهیم موجود در هر یک از بخش‌های پروژه، محاسبات ریاضی هر بخش، نتایج از حاصل از همر عملیات) در طول تحویل حضوری پرسیده خواهد شد.

موفق باشید