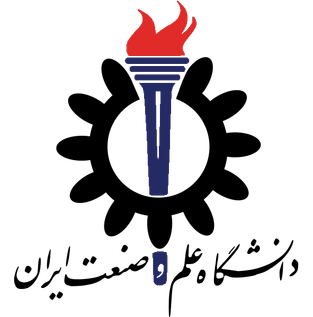
**بــه نــام خــدا**



دانشکده مهندسی کامپیوتر

**هوش مصنوعی و سیستم های خبره**

**پروژه اول – درخت تصمیم**

**دکتر آرش عبدی**

**پاییز 1403**

**طراحان :**

**محمدصادق نعمت‌پور**

**نیایش خانی**

* **در صورت وجود هر گونه ابهام در سوالات تنها به طراح آن سوال پیام دهید.**
* **با توجه به تنظیم شدن ددلاین تمارین توسط خود شما امکان تمدید وجود ندارد.**
* **زبان برنامه‌نویسی و قالب تمپلیت پایتون است ولی برای تمرین‌های اول می‌توانید از #C نیز استفاده کنید.**
* **کل محتوای ارسالی را زیپ کرده و نام آن را شماره دانشجویی خود قرار دهید.**
* **نیازی به نوشتن داک نیست.**
* **به سوالات انتهایی با دقت و به صورت کامل پاسخ دهید.**
* **انجام تمرین تک نفره است. لطفا به تنهایی انجام شود، در غیر اینصورت نمره منفی در نظر گرفته خواهد شد.**

**آیدی تلگرام طراحان :**

[@msnp1381](https://t.me/msnp1381)  
[@mainlynia](http://t.me/mainlynia/)

**پروژه اول هوش مصنوعی و سیستم های خبره: درخت تصمیم**

**هدف پروژه:** پیاده سازی درخت تصمیم با استفاده از آنتروپی و Gini index برای دیتاست‌های ذکر شده

**شرح پروژه:**

1. **تجزیه و تحلیل مجموعه داده:**

با تجزیه و تحلیل مجموعه داده‌های ارائه شده شروع کنید. Feature های موجود را گسسته سازی کنید، مقادیری که موجود نیستند را با مقداری مناسب جایگزین کنید (و یا می توانید آن سطر را حذف کنید) ، مواردی که به صورت عدد نیستند را به عدد تبدیل کنید و ... .

برای گسسته‌سازی ورودی‌های از نوع پیوسته یا ورودی‌های دارای مقادیر خیلی زیاد بازه‌های عددی در نظر بگیرید. یک ایده این است که بازه مینیمم تا ما کزیمم اعداد در مجموعه آموزشی را به تعدادی بازه مساوی تقسیم کنید و دو بازه اضافی هم برای مقادیر کمتر از مینیمم و بیشتر از ماکزیمم در نظر بگیرید. همچنین میتوانید ایده‌های دیگری را نیز برای گسسته‌سازی ورودی‌های پیوسته ارائه دهید و آنها را امتحان کنید.

**2. پیاده‌سازی الگوریتم درخت تصمیم:**

کلاسها و الگوریتم درخت تصمیم مورد نظر خود را پیاده کنید. در نوت‌بوک‌هایی که در اختیار شما قرار گرفته اند، توابع آماده پیاده‌سازی هستند و کافی است با خواندن توضیحات موجود در کد، آنها را برای هر کدام از دیتاست‌ها تکمیل کنید.(main-crona.ipynb و main-titanic.ipynb)

اگر در پیاده سازی این الگوریتم خلاقیت و نوآوری داشته باشید، نمره امتيازى خواهد داشت؛ مثلا الگوريتم شما به صورت تطبيقى (adaptive) باشد و ... .

یک بار الگوریتم خود را با آنتروپی و بار دیگر با استفاده از Gini index پیاده‌سازی كنيد.

**3. آموزش درخت تصميم (training):**

درخت تصمیم خود را با استفاده از دو ديتاست‌های زیر train كنيد. می توانيد از 2000 داده اول برای این کار استفاده کنید. (توجه داشته باشید که الگوریتم شما باید برای هر دو دیتاست به صورت جداگانه اجرا شود.)

Dataset#1: [titanic](https://www.kaggle.com/c/titanic/data) | Dataset#2: [corona test](https://www.kaggle.com/datasets/expertjha001/corona-tested-dataset)

1. **ارزيابى مدل:**

مدل درخت تصمیم خود را با استفاده از تکنیک‌های cross-validation ارزیابی کنید و نتایج را نشان دهید. از میان داده‌هایی که برای train استفاده نكرده اید به صورت تصادفی برای test انتخاب کنید.

1. **تحلیل و بررسی نتایج:**

درخت تصميم بر اساس آنتروبى و Gini index را با یكدیگر مقایسه کنید. سعی کنيد با استفاده از تكنيک‌های مختلف درخت تصمیم خود را بهبود بخشید و نتایج به دست آمده را تحلیل کنید. میزان دقت درخت خود را نمایش دهید.

اگر از ایده جدیدی در هر قسمت پروژه استفاده کردید (مثلا استفاده از روش جدیدی برای گسسته‌سازی داده‌ها)، نشان دهید که این کار چه تغییری در درخت تصمیم شما ایجاد کرده است و آیا باعث بهبود آن شده است یا خیر.

* خلاقیت شما برای افزایش دقت درخت مثل افزایش داده‌های آموزشی یا هر گونه انتخاب هوشمندانه از میان آنها، روش‌های جدیدتر و حرفه‌ای تر گسسته‌سازی و يا حتى فعاليت‌های اضافه‌تر حرفه‌اى مانند تحليل‌هاى آمارى جداگانه از فيچرها، Data cleaning يا Feature engineering و .. می تواند نمره امتیازی داشته باشد.
* در نظر داشته باشيد براى پياده سازى درخت تصميم نبايد از توابع آماده استفاده كنيد. لذا فرمول آنتروپی و Gini index تابع خود درخت تصمیم (همانند توابع بازگشتی و فرآیند درخت سازی و ...) را باید خودتان پیاده کنید.
* استفاده از توابع آماده تنها برای بخش‌های دیگر مانند خواندن اکسل، احیانا نمایش گرافیکی خروجی درخت (درصورت علاقه)، نمایش دقت خروجى و ... بلامانع است.

**ساختار پروژه و نحوه تحویل**

├── corona

│   ├── \_\_init\_\_.py

│   └── **main-corona.ipynb**

├── titanic

│   ├── \_\_init\_\_.py

│   └── **main-titanic.ipynb**

├── DecisionTree\_4031.docx

├── **decision\_tree.py**

├── **questions.docx**

├── README.md

├── requirements.txt

وظیفه شما تکمیل decision\_tree.py که نشان دهنده ساختار درخت تصمیم است.

ساختار دو نوت‌بوک main-titanic.ipynb و main-corona.ipynb کاملاً شبیه است اما در مصورسازی و پیش پردازش تفاوت‌هایی وجود دارد. که در این‌ها دیتاست های مرتبط لود و انجام می‌شود.

برای بارگذاری پروژه حتماً REAME.md را مطالعه کنید.

**آنچه تحویل داده می‌شود:**

1. کد اجرایی برنامه با توضیحات لازم برای اجرا
2. پاسخ به سوالات قرار گرفته در داک برای هر دیتاست(questions.docx)
3. هر گونه تحلیل اضافه مفید و خلاقیت😊(می تواند نمره امتیازی داشته باشد.)