تمرین شمارهٔ ۳

۱

در این سوال به ارزیابی مفاهیم ارائه شده در فصل سه کتاب مرجع [1] پرداخته می شود. (۲۰ نمره)

۱.۱ متمم فازی (**Fuzzy Complement**)

تابع $c : [0,1] \to [0,1]$ که شرط زیر را برقرار می‌کند، متمم فازی نامیده می‌شود:

$$c[\mu_A(x)] = \mu_{\bar{A}}(x)$$

با دو آکسیوم:

$$\text{c۱: Axiom} c(0) = 1 \ , \ c(1) = 0$$
$$\text{c۲: Axiom (نزولی بودن)} c(a) \geq c(b) \text{ آنگاه } a < b \text{ اگر}$$

فرض کنید در یک سامانه هوشمند تهویه مطبوع، مجموعه فازی «هوای گرم» تعریف شده و $\mu_{warm}(x)$ نشان می‌دهد هوا با دمای x چقدر گرم است. متمم این مجموعه یعنی $\mu_{\overline{warm}}(x)$ قرار است چه چیزی را نشان دهد؟ سپس توضیح دهید چرا آکسیوم c۲ برای یک تابع متمم ضروری است — اگر این شرط نقض شود (یعنی با افزایش گرمی هوا، میزان «نه‌ـگرمی» هم افزایش یابد) چه مشکل منطقی‌ای در تصمیم‌گیری سیستم پیش می‌آید؟

۲.۱ اجتماع و اشتراک فازی (**S-norms / T-norms**)

ساده‌ترین انتخاب برای اشتراک دو مجموعه فازی، عملگر $t(a,b) = \min(a,b)$ است اما در عمل خانواده‌های دیگری از عملگرها مثل ضرب جبری

$$t_{ap}(a,b) = ab$$

نیز تعریف و استفاده می‌شوند. چرا علی‌رغم سادگی min، به این عملگرهای دیگر نیاز داریم؟ حداقل دو دلیل جداگانه بیاورید: یکی از منظر کاربردی (نشان دهید با یک مثال عددی ساده، $\min(a,b)$ در برابر تغییرات مقدار بزرگ‌تر چه رفتاری دارد و چرا این رفتار همیشه مطلوب نیست) و یکی از منظر نظری.

۳.۱ اشتراک فازی (T-norms) و کران‌های آن

هر تابع $t : [0,1] \times [0,1] \to [0,1]$ که در چهار آکسیوم t۱ تا t۴ صدق کند، یک t-norm نامیده می‌شود. قضیه ۳.۲ نشان می‌دهد برای هر t-norm داریم:

$$t_{dp}(a,b) \leq t(a,b) \leq \min(a,b)$$

فرض کنید یک فیلتر اسپم دو سیگنال مستقل دارد: a میزان مشکوک‌بودن لینک‌های ایمیل، و b میزان مشکوک‌بودن آدرس فرستنده. این دو سیگنال با عملگر اشتراک AND (فازی) ترکیب می‌شوند تا میزان نهایی «مشکوک‌بودن ایمیل» به دست آید.

[1] A Course in Fuzzy Systems and Control, Li-Xin Wang

الف) چرا منطقی است که خروجی هر t-norm هرگز از $\min(a,b)$ بیشتر نشود؟ این کران بالا چه قید معقولی را روی مفهوم «AND» تحمیل می‌کند؟

ب) اگر به جای $\min(a,b)$ از ضرب جبری $t_{ap}(a,b) = ab$ استفاده کنیم، برای $a = b = 0.9$ هر دو عملگر را محاسبه و مقایسه کنید. سپس با حداقل یک دلیل توضیح دهید چرا طراح سیستم ممکن است ضرب جبری را به مینیمم ترجیح دهد، با وجود اینکه مینیمم سادگی بیشتری دارد.

۴.۱ قانون دمورگان و طبقه مرتبط (**Associated Class**)

گفته می‌شود متم c، اجتماع s و اشتراک t یک طبقه مرتبط (**Associated Class**) تشکیل می‌دهند اگر در قانون دمورگان زیر صدق کنند:

$$c[s(a,b)] = t[c(a), c(b)]$$

این شرط تضمین می‌کند که سه عملگر «هم‌خانواده» هستند، اما هیچ‌جا تضمین نمی‌کند که هر سه‌تایی دلخواه از یک متم، یک s-norm و یک t-norm که جداگانه و بدون توجه به هم انتخاب شوند، لزوماً در این رابطه صدق کنند.

با توجه به این نکته توضیح دهید: اگر در یک سامانه استنتاج فازی (Fuzzy Inference System) ، طراح به‌صورت دلخواه یک متم از خانواده Sugeno، یک اجتماع از خانواده Dombi، و یک اشتراک از خانواده Yager را — بدون بررسی اینکه طبقه مرتبط تشکیل می‌دهند یا نه — کنار هم استفاده کند، چه نوع ناسازگاری منطقی ممکن است در قواعد سیستم (مثلاً در قاعده‌ای مثل «نقیض (A،B) باید با نقیض A یا نقیض B یکی باشد») رخ دهد؟

۲ پیش بینی سری زمانی (۲۰ نمره)

در این بخش می‌خواهیم با استفاده از ابزار استنتاج فازی به شناسایی یک سیستم آشوبناک بپردازیم. سری زمانی مکی‌ـگلس (McKey-Glass) یکی از پر اهمیت ترین سری های زمانی جهت آزمایش قدرت پیش بینی مدل های ارائه شده در مقالات پژوهشی است؛ این سری زمانی به عنوان Benchmark جهت آزمایش عملکرد معماری های نوآورانه پیش بینی سری زمانی به کار می‌آید.

سری زمانی مکی‌ـگلس توسط معادله دیفرانسیل تأخیری زیر تعریف می‌شود:

$$\frac{dx(t)}{dt} = \frac{a \cdot x(t-\tau)}{1 + x^c(t-\tau)} - b \cdot x(t) \tag{۱}$$

این معادله دارای ویژگی‌های منحصربه‌فردی است که شناسایی سیستم را ضروری می‌سازد:

- غیرخطیت شدید: وجود عبارت $\frac{x^c(t-\tau)}{1+x^c(t-\tau)}$ با توان $c = 10$ باعث ایجاد غیرخطیت شدید در سیستم می‌شود. این غیرخطیت به قدری پیچیده است که روش‌های تحلیلی کلاسیک قادر به ارائه راه‌حل بسته[۲] برای آن نیستند.
- رفتار آشوبناک: برای مقادیر $\tau \geq 17$، سیستم وارد رژیم آشوب[۳] می‌شود. در این رژیم، حساسیت شدید به شرایط اولیه[۴] وجود دارد؛ به این معنا که تغییرات بسیار کوچک در شرایط اولیه منجر به تفاوت‌های بسیار بزرگ در رفتار بلندمدت سیستم می‌شود.
- بعد فرکتالی: جاذب عجیب[۵] سیستم مکی‌ـگلس دارای بعد فرکتالی غیرصحیح است که با افزایش τ افزایش می‌یابد. برای $\tau = 17$ بعد فرکتالی تقریباً 2.1 و برای $\tau = 30$ تقریباً 3.6 است. این پیچیدگی هندسی، مدل‌سازی تحلیلی را عملاً غیرممکن می‌سازد.

[2] Closed-form Solution
[3] Chaos
[4] Sensitive Dependence on Initial Conditions
[5] Strange Attractor

- تأخیر زمانی: وابستگی خروجی به مقادیر گذشته با تأخیر τ باعث می‌شود فضای فاز سیستم بی‌نهایت‌بعدی باشد. این ویژگی، مدل‌سازی مستقیم از معادلات فیزیکی را بسیار دشوار می‌کند.

ساخت مجوعه دادگان

با قرار دادن τ برابر با ۱۷ تعداد ۲۰۰۰ نمونه از سری زمانی مکی–گلس به صورت یکنواخت تولید کنید؛ سپس توزیع داده ها را به میانگین صفر و انحراف معیار ۱ نرمالیزه کنید.

مجموعه دادگان را به ۳ بخش آزمایش، آزمون و ارزیابی تقسیم کنید، به نسبت ۷۰، ۱۵ و ۱۵

برای تولید سری زمانی مکی–گلس مقادیر $a = 0.2$، $b = 0.1$ و $c = 10$ را در نظر بگیرید.

آموزش مدل **ANFIS**

تعداد delay ها را برابر ۳ در نظر بگیرید؛ همچنین در تنظیمات Genfis برابر با Grid Partitioning و به ازای هر ورودی ۲ تا تابع عضویت انتخاب کنید.

با تنظیم پارامتر های ANFIS نتایج آموزش مدل را با معیار های نظیر MAE، RMSE و r^2 ارزیابی کنید.

۳ روش Wang–Mendel مبتنی بر **Lookup Table** (۲۵ نمره)

یکی از روش‌های مهم تولید خودکار قواعد فازی از داده‌های عددی، روش Wang–Mendel است. در این روش ابتدا داده‌های ورودی و خروجی فازی‌سازی شده، سپس از روی داده‌ها قواعد فازی استخراج می‌شوند و در نهایت با حذف تعارضات، یک پایگاه قواعد (Rule Base) یا Lookup Table تشکیل می‌شود. هدف این قسمت، طراحی یک سیستم فازی دو ورودی و یک خروجی با استفاده از روش Wang–Mendel و ارزیابی عملکرد آن در تقریب یک تابع غیرخطی است.

تولید دیتاست

داده‌های آموزشی مطابق رابطه زیر تولید شوند:

$$y = \sin(\pi x_1) + 0.5x_2^2 \ , \ x_1 \in [-1, 1], x_2 \in [-1, 1], N = 500$$

نمونه آموزشی تولید گردد. برای تولید داده‌ها از کد زیر استفاده نمایید:

```
import numpy as np

np.random.seed(42)

N = 500

x1 = np.random.uniform(-1, 1, N)

x2 = np.random.uniform(-1, 1, N)

y = np.sin(np.pi*x1) + 0.5*(x2**2)

```

بخش ۱: فازی‌سازی

برای هر یک از متغیرهای موجود ۷ تابع عضویت مثلثی یکنواخت تعریف نمایید. توابع عضویت ورودی‌ها در بازه $[-1, 1]$ تعریف شوند. توابع عضویت خروجی در بازه داده‌های خروجی تولیدشده تعریف شوند. برچسب‌های فازی زیر را به کار ببرید و

سپس نمودار توابع عضویت را رسم کنید:

$$NB, NM, NS, ZE, PS, PM, PB$$

بخش ۲: استخراج قواعد Wang–Mendel

برای هر نمونه آموزشی که به صورت زیر است

$$(x_1, x_2, y)$$

ابتدا برچسب فازی غالب هر متغیر را تعیین کنید. سپس قاعده فازی متناظر را تشکیل دهید. درجه اعتبار قاعده را از رابطه زیر محاسبه کنید:

$$Degree = \mu_A(x_1) \times \mu_B(x_2) \times \mu_C(y)$$

بخش ۳: رفع تعارض قواعد

اگر چند قاعده دارای Antecedent یکسان و Consequent متفاوت باشند، فقط قاعده‌ای را نگه دارید که بیشترین درجه را دارد و بقیه را حذف کنید.

بخش ۴: تشکیل **Rule Base** نهایی

پس از حذف تعارض‌ها، Rule Base نهایی را تشکیل دهید. Lookup Table حاصل را به صورت جدولی نمایش دهید. تعداد قواعد نهایی را گزارش کنید.

بخش ۵: استنتاج فازی

برای داده‌های جدید که به صورت مقابل تعریف می‌شوند:

$$(x_1^{test}, x_2^{test})$$

مراحل پیش رو را به ترتیب انجام دهید: ابتدا فازی‌سازی ورودی‌ها را بیان کنید، سپس فعال‌سازی قواعد، پس از آن تجمیع خروجی‌ها را انجام دهید و در نهایت غیرفازی‌سازی را انجام دهید. لازم به ذکر است که برای غیرفازی‌سازی از روش Centroid استفاده نمایید.

بخش ۶: ارزیابی عملکرد

یک داده تست برای خود انتخاب کنید و آن را تولید کنید، خروجی واقعی را از رابطه اصلی محاسبه نمایید؛ سپس خروجی سیستم فازی را محاسبه کنید و معیار RMSE را به دست آورید و گزارش کنید.

بخش ۷: تحلیل نتایج

پس از بررسی نتایج خود به پرسش‌های زیر پاسخ دهید:

۱. چرا در روش Wang–Mendel نیاز به حذف تعارض قواعد وجود دارد؟

۲. افزایش تعداد توابع عضویت چه تأثیری بر تعداد قواعد و دقت سیستم دارد؟

۳. مهم‌ترین مزیت استفاده از Lookup Table در سیستم‌های فازی چیست؟

۴ مدل‌سازی مصرف سوخت خودرو و تحلیل داده با مدل ANFIS (۳۵ نمره)

توصیف مسئله

در صنایع خودروسازی و سیستم‌های مدیریت هوشمند انرژی، پیش‌بینی دقیق میزان مصرف سوخت خودرو بر اساس رفتار راننده اهمیت بالایی دارد. با این حال، استفاده از مدل‌های کاملاً جعبه‌سیاه (مانند شبکه‌های عصبی عمیق معمولی) برای سیستم‌های نظارتی یا رانندگان چندان قابل درک نیست. هدف این پروژه، طراحی یک سیستم استنتاج عصبی‌ـفازی تطبیقی (ANFIS) از صفر و بدون استفاده از کتابخانه‌های آماده (مانند skfuzzy) است تا میزان مصرف سوخت لحظه‌ای خودرو y را بر اساس دو متغیر ورودی زیر پیش‌بینی کند:

- سرعت خودرو (x_1 – Velocity) بر حسب کیلومتر بر ساعت (km/h)
- نرخ فشردن پدال گاز (x_2 – Throttle Rate) بر حسب درصد (0 تا 100)

مدل ANFIS مورد نظر از نوع تکاکی‌ـسوگنوـکانگ (TSK) مرتبه اول است و برای هر متغیر ورودی، ۲ تابع عضویت گوسی در نظر گرفته می‌شود (در مجموع ۴ قاعده فازی). علاوه بر مدل‌سازی فازی، دانشجویان باید مفاهیم تحلیل داده (Data Analysis) و یادگیری ماشین کلاسیک را که در طول ترم آموخته‌اند، روی این مسئله پیاده‌سازی و ارزیابی کنند.

کد تولید دیتابیس پروژه (**Dataset Generation**)

دانشجویان موظف هستند دیتابیس پروژه را که شامل ۵۰۰ نمونه رانندگی هم‌نهاده (Synthetic) است، با اجرای کد پایتون زیر تولید کنند. این داده‌ها بر اساس یک رابطه فیزیکی پیچیده و غیرخطیِ مصرف سوخت در موتور خودرو فرمول‌بندی شده‌اند.

```
import numpy as np
# Set random state based on last two digits of student ID(here: Random State = 42)
np.random.seed(42)
# Number of data samples
n_samples = 500
# Vehicle speed between 10 and 140 km/h
V = np.random.uniform(10, 140, n_samples)
# Throttle rate between 5 and 95 percent
T = np.random.uniform(5, 95, n_samples)
Fuel = 4.5 + 0.05 * T + 0.001 * (V - 80)**2 + 0.0002 * (T * V)

# Save database: col1=Speed, col2=Throttle, col3=Fuel Consumption
dataset_anfis = np.column_stack((V, T, Fuel))

```

دانشجویان باید گام‌های زیر را به طور کامل پیاده‌سازی کرده و گزارش تحلیلی آن را ارائه دهند:

گام ۱: طراحی معماری ۵ لایه ANFIS

ساختار لایه‌ای مدل ANFIS پیشنهادی (شامل لایه فازی‌سازی ورودی‌ها، لایه ضرب یا T-norm قواعد، لایه نرمال‌سازی وزن‌ها، لایه تابع تالی خطی TSK و لایه خروجی کل) را به صورت ریاضی فرمول‌بندی کرده و نمودار بلوکی (معماری شبکه‌ای) آن را رسم کنید.

گام ۲: پیاده‌سازی یادگیری ترکیبی (**Hybrid Learning**)

الگوریتم آموزش ترکیبی را از صفر کدنویسی کنید:

- در فاز رفت (Forward Pass): پارامترهای پیش‌فرض گوسی را ثابت نگه داشته و پارامترهای تالی قواعد فازی (p, q, r) را با روش کمترین مربعات خطا (LSE) به صورت ماتریسی در یک گام بهینه کنید.

- در فاز برگشت (Backward Pass): خطا را پس‌انتشار داده و پارامترهای پیش‌فرض (مراکز c و پهنای σ توابع عضویت گوسی) را با الگوریتم گرادیان نزولی (Gradient Descent) به روز رسانی کنید.

گام ۳: استخراج و تفسیر قواعد فازی

- پس از اتمام فرآیند آموزش، هر ۴ قاعده فازی استخراج شده را به فرم استاندارد متنی بنویسید (مثال: اگر سرعت کم و پدال زیاد باشد، آنگاه مصرف سوخت برابر است با...).
- ضرایب عددی به دست آمده در بخش تالی را با منطق مهندسی مکانیک خودرو (رفتار موتور در دورهای مختلف و شتاب‌گیری) تطبیق داده و تحلیل کنید.

گام ۴: تحلیل اکتشافی داده‌ها (EDA)

با استفاده از کتابخانه‌های متپلات‌لیب (matplotlib) یا سیبورن ،(seaborn) تحلیل‌های آماری زیر را روی داده‌های خام انجام دهید:

- نمودار توزیع فراوانی (Histogram) متغیر خروجی (Fuel) را رسم کنید.
- نمودار پاشش (Scatter Plot) متغیرهای ورودی را به صورت مجزا نسبت به خروجی رسم کنید.
- ماتریس همبستگی (Correlation Matrix Heatmap) داده‌ها را ترسیم کرده و مشخص کنید کدام متغیر همبستگی خطی قوی‌تری با مصرف سوخت دارد؟

گام ۵: مهندسی ویژگی (**Feature Engineering**)

- یک ویژگی جدیدِ مبتنی بر فیزیک خودرو به نام «شاخص اتلاف توان دینامیکی» با رابطه $P_{loss} = V^2 \times T$ تعریف و به دیتابیس اضافه کنید.
- میزان همبستگی این ویژگی مهندسی‌شده را با میزان مصرف سوخت بسنجید و تحلیل کنید چرا تزریق دانش پیش‌فرض (Domain Knowledge) به این صورت، می‌تواند به مدل‌های ساده‌تر کمک کند.

گام ۶: مدل‌سازی با یادگیری ماشین کلاسیک

داده‌ها را به دو بخش آموزش (Train 80%) و تست (Test 20%) تقسیم کرده و دو مدل زیر را روی آن‌ها برازش (Fit) کنید:

- مدل رگرسیون خطی چندگانه (Multiple Linear Regression)
- مدل درخت تصمیم برای رگرسیون (Decision Tree Regressor)

گام ۷: ارزیابی مقایسه‌ای جامع

- معیارهای ارزیابی خطای MAE (میانگین قدرمطلق خطا)، RMSE (ریشه میانگین مربعات خطا) و شاخص تبیین R^2-Score را برای هر سه مدل (رگرسیون خطی، درخت تصمیم، و مدل ANFIS طراحی‌شده) روی داده‌های تست محاسبه کنید.
- نتایج را در قالب یک جدول نهایی ارائه داده و با توجه به ماهیت فیزیکی داده‌ها، دلایل ضعف یا برتری هر مدل را به صورت تخصصی به بحث بگذارید.