مسئله

هدف: پیش‌بینی کیفیت محیط داخلی

دیتاست: FullDataset_CombinedSimple.csv

ما مدل شخصی برای هر فرد نمی‌سازیم، بلکه یک مدل عمومی داریم که روی توالی‌های زمانی همه افراد آموزش می‌بینند.

نوع مسئله:

ابتدا regression ، تولید خروجی اعشاری و محاسبه ی MAE, MSE, R^2

سپس تبدیل خروجی به عدد صحیح بین ۱ تا ۵ با رند کردن یا clip به بازه ی [1,5] و محاسبه ی Accuracy، F1، Confusion Matrix

Preprocessing & Feature Engineering

ستون هایی که برای تحلیل استفاده می شوند و وارد مدل نمی شوند:

برچسب زمانی برای مرتب سازی و محاسبه ساعت روز (Hour)، خستگی زمانی (Time Fatigue):

TimeVote,

برچسب زمانی برای محاسبه فاصله (Lag Accuracy):

Time_Troom, Time_RH, Time_CO2, Time_EA, Time_Ttrend, Time_Sound, Time_VOC, Time_Lighting,

برای ساخت فیلد location:

LocationFront, LocationBack, LocationLeft, LocationRight, LocationMiddle,

ستون هایی که به مدل منتقل می شوند:

فقط با مقدار فعلی: (مقدار ایستا ست و در طول زمان دیتاست تغییر نمی کند)

ID, Student, Gender, Age, CaseStudy

هم مقدار فعلی هم مقادیر قبلی به عنوان lag: (مقدار پویا ست و در طول زمان دیتاست تغییر می کند)

Moment, CLO_Simple, Activity, Troom, RH, CO2, Sound, Lighting, Ttrend, Trm, WindowsClosed,

فقط به عنوان lag:

Thermal_Warm, Thermal_Cold, Thermal_Draft, Thermal_Other,
IAQ_Stuffy, IAQ_Odour, IAQ_Dry, IAQ_Humid, IAQ_Other,
AC_Classroom, AC_Hallway, AC_Device, AC_Outside, AC_Other, AC_Ventilation,
Vis_Blackboard, Vis_Underexposure, Vis_Overexposure, Vis_Glare, Vis_Other,

ستون هایی که باید پیش بینی شوند: دارای مقدار 1 برای کمترین تا 5 برای پیشترین رضایت

ThermalSatisfaction, IAQSatisfaction, AcousticSatisfaction, VisualSatisfaction, **IEQSatisfaction**

ستون هایی که کنار گذاشته می شوند:

دارای correlation زیاد:

EA, CLO_Detailed,

اطلاعات تکراری از CaseStudy و Student می دهد:

Subgroup,

حجم کم داده ها:

VOC, GroupObs, AF
WindowMajority, ActivityMajority,

مرحله ۱: پاکسازی و اصلاح مقادیر (Data Cleaning)

اصلاح Age: * اگر Age خالی است، بر اساس میانگین سن در آن CaseStudy خاص (مثلاً PrimarySchool) پر شود.

(در دیتاست برای تمام سطرها با Age بدون مقدار، اگرCaseStudy برابر با PrimarySchool یا SecondarySchool باشد student برابر با TRUE می باشد)

مدیریت مقادیر مفقود (Missing Values):

برای ستون‌های عددی: به جای حذف سطر، خانه‌های خالی با مقدار ۱۰- پر شوند (مگر اینکه مقدار پیش فرض دیگری پیشنهاد شود) تا مدل بفهمد این داده در آن لحظه ثبت نشده است.

برای ستون‌های متنی: با کلمه Unknown جایگزین شوند.

عددی کردی ستون های متنی:

CLO_Simple:

Lightly: 0.5

Neutral: 0.7

Warmly: 1

برای سطرهای خالی: عدد میانگین لباس بقیه افراد در همان روز

Moment:

مقدار ۱ برای Start: (شروع کلاس؛ زمانی که فرد تازه وارد شده و هنوز با محیط تطبیق پیدا نکرده است).

مقدار ۲ برای End: (پایان کلاس؛ اوج خستگی و بیشترین زمانِ صرف شده در محیط).

مقدار ۱.۵ برای Blank: (چون نمی‌دانیم دقیقاً کی بوده، آن را حد وسط قرار می‌دهیم تا مدل فرض کند جایی بین شروع و پایان بوده است).

مرحله ۲: مهندسی ویژگی‌ها (Feature Engineering)

ایجاد مختصات مکانی (Location): یک مختصات دکارتی (X,Y) که هر مقدار آن نماینده یک جهت است، با اولویت‌بندی جهات اصلی و استفاده از Middle به عنوان جایگزین.

```
x=0
y=0
if front=true then y=1
if back=true then y=3
if Left=true then x=1
if Right=true then x=3
if x=0 then if middle=true then x=2
if y=0 then if middle=true then y=2
```

محاسبه امتیاز حساسیت (User_Sensitivity_Score): جمع جبری تیک‌های زده شده در ستون‌های Specific_Issues (مثل Thermal_Cold) برای هر سطر (نشان‌دهنده سخت‌گیری فرد).

نرخ تولید گرمای بدن (Metabolic Rate): تبدیل متون Activity به اعداد معادل بر اساس استانداردهای آسایش حرارتی (ASHRAE)

مقادیر پیشنهادی برای فعالیت (Activity) به Met:

مقدار ۱.۰ برای Test: (کمترین فعالیت فیزیکی، تمرکز ذهنی بالا در حالت نشسته).

مقدار ۱.۱ برای Passive course: (نشستن معمولی و گوش دادن در کلاس).
مقدار ۱.۲ برای Other: (مقدار متوسط فعالیت‌های پیش‌بینی نشده).
مقدار ۱.۲ برای Not applicable: (خنثی‌سازی داده‌های نامشخص).
مقدار ۱.۲ برای Blank (خالی): (جایگزینی برای جلوگیری از خطای محاسباتی).
مقدار ۱.۵ برای Presentation: (ایستادن و صحبت کردن که انرژی بیشتری می‌برد).
مقدار ۱.۷ برای Active course: (بیشترین فعالیت؛ شامل کارهای گروهی و جابه‌جایی).
استخراج ویژگی‌های زمانی:
Hour: استخراج ساعت از TimeVote. (فقط مقدار فعلی به مدل ارسال می شود)
Time_Fatigue: تفاوت زمان فعلی با اولین حضور فرد در آن روز. (فقط مقدار فعلی به مدل ارسال می شود)
Time_Delta: تفاوت زمان فعلی با سطر قبلی (بدون در نظر گرفتن ID). (فقط مقدار فعلی به مدل ارسال می شود)

یافتن تعداد lag به عنوان hyperparameter مدل

یافتن تعداد lag مناسب:
حلقه به ازای هر l از لیست lag= 3, 5, 10, 15 , 20 {
مرحله۳ مرتب‌سازی و ساخت Lagها (Vectorization)
مرحله ۴ بازگرداندن نظم و تقسیم‌بندی
اجرای پایپلاین (Train + TimeSeriesSplit + GridSearch)
Test
ارزیابی
{

شروع حلقه

مرحله ۳: مرتب‌سازی و ساخت Lagها (Vectorization)
مرتب‌سازی اول: کل دیتاست به صورت صعودی بر اساس ID و سپس TimeVote مرتب شود. (این کار باعث می‌شود سوابق هر فرد پشت سر هم قرار بگیرد).
ساخت ستون‌های Lag (از ۱ تا l):
برای تمام ستون‌های مورد توافق (سنسورها، مکان، فعالیت، پوشش، زمان، رضایت‌ها و دلایل نارضایتی)، ۱۰ ستون جدید ساخته شود.
نکته مهم: برای ستون‌های «رضایت» و «دلایل نارضایتی»، مقدار لحظه فعلی (t) حذف شود و فقط Lagها باقی بمانند تا نشت داده (Data Leakage) رخ ندهد.
مرحله ۴: بازگرداندن نظم و تقسیم‌بندی
مرتب‌سازی دوم: حالا که تمام سوابق در هر سطر تزریق شده، دیتاست را دوباره فقط بر اساس TimeVote مرتب کنید. (این کار برای حفظ واقع‌گرایی مدل در زمان آموزش حیاتی است).
تقسیم‌بندی ۸۰-۲۰ (Time-Series Split):
۸۰ درصد داده‌های قدیمی‌تر (بر اساس زمان) برای آموزش (Train).
۲۰ درصد داده‌های جدیدتر برای تست (Test).
نکته: دقت کنید که چون داده‌ها را بر اساس زمان تقسیم می‌کنید، مدل باید بتواند آینده را بر اساس گذشته پیش‌بینی کند.

TimeSeriesSplit, Grid Search و یافتن بقیه hyperparameter مدل

تعداد درخت: بهینه‌سازی: بهترین راه برای پیدا کردن عدد دقیق، استفاده از روش Grid Search و TimeSeriesSplit است
عمق درخت: بهینه‌سازی: بهترین راه برای پیدا کردن عدد دقیق، استفاده از روش Grid Search و TimeSeriesSplit است

پیدا کردن بهترین عمق (Max Depth) و تعداد درخت با استفاده از TimeSeriesSplit برای حفظ نظم زمان.
(split برحسب زمان و غیرتصادفی،
گذشته ← train
آینده ← test)
بهترین تعداد درخت: از ۱۰۰ درخت شروع کرده تعداد را زیاد می کند تا زمانی که تاثیری بر دقت نداشته باشد

Train مدل با الگوریتم Random Forest و در مرحله بعد XGBoost، سپس neural network سپس SVM

هدف: آموزش ۵ مدل مجزا برای ۵ هدف (Target)

Grid Search و Train یک فرآیند ترکیبی هستند
در نهایت شما فقط یک «شیء مدل» دارید که هم هوشمندترین عمق را انتخاب کرده و هم آموزش دیده است

تست

ارزیابی مدل

الف) میانگین مجذور خطا (Mean Squared Error - MSE): خطای جریمه‌دار (برای کاهش خطاهای بزرگ).
این معیار نشان می‌دهد که پیش‌بینی‌های مدل چقدر از واقعیت فاصله دارند. هرچه این عدد به صفر نزدیک‌تر باشد، مدل بهتر است.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

ب) دقت (Accuracy): درصدِ کلیِ تشخیص‌های درست (دسته‌بندی)
اگر بخواهیم ببینیم مدل در چند درصد موارد دقیقاً همان عددِ دانش‌آموز را درست حدس زده است:

$$Accuracy = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \times 100$$

ج) $score\ R^2$ (ضریب تعیین): درصدِ موفقیت مدل در توضیحِ رفتار محیط.
این عدد بین ۰ تا ۱ است. اگر ۱ باشد یعنی مدل شما ۱۰۰٪ رفتار دانش‌آموزان را فهمیده است

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

د) MAE (Mean Absolute Error): میانگین خطای واقعی (به واحد نمره).
برخلاف MSE که خطا را به توان ۲ می‌رساند، MAE دقیقاً به شما می‌گوید مدل به طور متوسط چند نمره اشتباه می‌کند.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

ه) F1-Score (برای ماتریس درهم‌ریختگی): تعادلِ دقت در کلاس‌های مختلف.
اگر بخواهید بگویید مدل چقدر در تشخیص افراد «ناراضی» موفق بوده، این معیار تعادل بین «دقت» و «صحت» را نشان می‌دهد.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Visualization

۱. نمودار پیش‌بینی در برابر واقعیت (Actual vs Predicted Plot)
۲. نمودار اهمیت ویژگی‌ها (Feature Importance Plot)
۳. نمودار باقی‌مانده‌ها (Residuals Plot)
۴. نمودار سری زمانی پیش‌بینی (Time-Series Forecast Plot)
۵. ماتریس درهم‌ریختگی (Confusion Matrix) - اگر داده‌ها را طبقه‌بندی کنیم

پایان حلقه