گزارش پروژه

درس داده‌کاوی ۲۰۲۶

محمدامین شیرمحمدی ۶۱۰۳۰۱۱۴۸
امیررضا نعیمی ۶۱۰۳۰۰۱۰۷

چکیده

هدف :این گزارش جزئیات توسعه و بهینه‌سازی یک pipeline یادگیری ماشین را شرح می‌دهد که برای پیش‌بینی یک متغیر هدف (y) با استفاده از یک dataset اولیه شامل ۱۰۰,۰۰۰ رکورد طراحی شده است.

روش‌ها :پیش‌پردازش جامع داده‌ها انجام شد که شامل استخراج ویژگی‌های زمانی (مانند duration_min، hour، day_of_weekو (season، حذف متغیرهای هم‌خط (collinear) و پاکسازی سطرهای تکراری بود Dataset .به سه بخش تقسیم شد: ۷۰٪ برای آموزش(training) ، ۱۵٪ برای ارزیابی (validation) و ۱۵٪ برای تست .(testing) مقادیر گمشده با تولید شاخص‌های گمشده‌بودن (که از طریق آزمون‌های تقاطعی کای‌اسکوئر ارزیابی شدند) و اعمال ایمپیوتیشن میانگین و میانه برای مقادیر عددی مدیریت شدند، در حالی که مقادیر گمشده categoricalبا برچسب «Unknown» پر شدند. مهندسی ویژگی با انکودینگ فراوانی (frequency encoding) برای متن‌های با کاردینالیته بالا، وان-هات انکودینگ (one-hot encoding) برای داده‌های categorical و استانداردسازی (standard scaling)به پایان رسید. برای بهینه‌سازیpipeline ، از یک انتخاب‌کننده ویژگی مبتنی بر درخت Random Forestاستفاده شد تا فضای ویژگی‌ها از ۳۱ ویژگی به ۲۵ ویژگی ضروری کاهش یابد.

نتایج و نتیجه‌گیری :سه مدل کاندید — رگرسیون لجستیک چندجمله‌ای(Multinomial Logistic Regression) ، مدل XGBoost و پرسپترون چندلایه — (MLP) با استفاده از جستجوی تصادفی (randomized search) و با کمک بخش validation از پیش تعیین‌شده، تنظیم پارامتر (hyperparameter-tuned) شدند. مدل‌های تنظیم‌شده در قالب یک چارچوب تحلیل تصمیم‌گیری چندمعیاره (MCDA) ارزیابی شدند که در آن عملکرد (۳۵٪)، تفسیرپذیری (۲۰٪)، پیچیدگی/پایداری (۱۵٪)، کارایی محاسباتی (۱۵٪) و سازگاری با داده (۱۵٪) وزن‌دهی شدند. در نهایت، مدل XGBoostبه عنوان مدل بهینه برای پروژه انتخاب شد که توانست weighted F1-score برابر با ۰.۸۳۷۸ در بخش validationو بالاترین امتیاز تصمیم‌گیری وزنی نهایی یعنی ۷.۵۷۲ را کسب کند.

مقدمه

توسعه مدل‌های پیش‌بینی‌کننده مقاوم و کارآمد فراتر از صرفاً آموزش دادن الگوریتم‌هاست؛ این امر نیازمند یک pipelineجامع و سرتاسری (end-to-end) است که با دقت کیفیت داده‌ها، اهمیت ویژگی‌ها و ارزیابی همه‌جانبه مدل را مدیریت کند. این پروژه جزئیات طراحی، بهینه‌سازی و ارزیابی یک pipeline پیشرفته یادگیری ماشین را با هدف پیش‌بینی دقیق یک متغیر هدف categorical یعنی yاز یک dataset مقیاس‌بزرگ شامل ۱۰۰,۰۰۰ رکورد اولیه شرح می‌دهد.

برای دستیابی به عملکرد پیش‌بینی بالا در عین حفظ تفسیرپذیری و کارایی مدل، این پروژه از یک روش‌شناسی بسیار ساختاریافته پیروی می‌کند که به سه فاز اصلی تقسیم می‌شود:

- **پیش‌پردازش دقیق داده‌ها :**داده‌های دنیای واقعی ذاتاً نویزدار و ناقص هستند. فاز اولیه این پروژه بر پاکسازی و تبدیل داده‌های خام به قالبی آماده برای مدل تمرکز دارد. این شامل مهندسی ویژگی‌های زمانی (مانند duration_min، hour، day_of_weekو (season، حذف متغیرهای بسیار هم‌خط و حذف داده‌های تکراری برای تضمین یکپارچگی داده‌ها است. مقادیر گمشده به صورت سیستماتیک با استفاده از آزمون‌های کای‌اسکوئر برای تعیین روابط گمشده‌بودن بررسی شده و سپس با استراتژی‌های هدفمند میانگین، میانه و ایمپیوتیشن categorical پر می‌شوند. در نهایت، داده‌ها با استفاده از انکودینگ فراوانی، وان-هات انکودینگ و استانداردسازی تبدیل می‌شوند.
- **تقسیم‌بندی استراتژیک و انتخاب ویژگی :**برای تضمین اینکه مدل‌ها بر اساس توانایی‌شان در) Generalizationتعمیم‌پذیری) روی داده‌های دیده‌نشده ارزیابی شوند، datasetبه سه بخش تقسیم می‌شود: ۷۰٪ برای آموزش مدل، ۱۵٪ برای validation و تنظیم پارامترها، و ۱۵٪ برای تست نهایی. برای بهینه‌سازی بیشترpipeline ، از تکنیک انتخاب ویژگی مبتنی بر درخت Random Forest استفاده می‌شود. این روش با موفقیت ابعاد داده‌های پردازش‌شده را کاهش می‌دهد و ۲۵ ویژگی حیاتی را از میان ۳۱ ویژگی موجود استخراج می‌کند؛ در نتیجه نویز را به حداقل رسانده و کارایی محاسباتی را بهبود می‌بخشد.
- **ارزیابی چندمعیاره مدل :**فاز نهایی فراتر از تکیه صرف بر معیارهای دقت استاندارد (accuracy) می‌رود. سه الگوریتم دسته‌بندی مختلف — رگرسیون لجستیک چندجمله‌ای، XGBoostو پرسپترون چندلایه — (MLP) از طریق یک فرآیند جستجوی تصادفی تنظیم پارامتر می‌شوند. سپس مدل‌های تنظیم‌شده با استفاده از یک چارچوب سفارشی تحلیل تصمیم‌گیری چندمعیاره (MCDA) امتیازدهی می‌شوند. این چارچوب یک امتیاز تصمیم‌گیری نهایی را با وزن‌دهی به پنج معیار مجزا محاسبه می‌کند: عملکرد پیش‌بینی (۳۵٪)، تفسیرپذیری (۲۰٪)، پیچیدگی/پایداری (۱۵٪)، کارایی محاسباتی (۱۵٪) و سازگاری با داده (۱۵٪.(

با تلفیق مهندسی دقیق داده‌ها با یک چارچوب ارزیابی چندبعدی، این گزارش یک رویکرد مقیاس‌پذیر و عملی را برای انتخاب مناسب‌ترین معماری یادگیری ماشین برای مسائل دسته‌بندی پیچیده به نمایش می‌گذارد.

روش‌ها

3.1. Data Cleaning and Initial Feature Pruning

در اولین مرحله از pipeline پیش‌پردازش، تمرکز ما روی ساده‌سازی dataset و حذف متغیرهای بدون اطلاعات بود. بررسی دستی datasetاولیه باعث شد قبل از شروع training مدل، چند ستون حذف شوند:

```
data['f11'].unique()
[35] ✓ 0.0s
... array(['US'], dtype=object)
```

Figure 1

- ستون f11 به‌طور کامل حذف شد. بررسی‌ها نشان داد که این ستون در تمام رکوردها مقدار ثابت و یکسانی ('US') دارد (شکل ۱)؛ در نتیجه، هیچ‌گونه variance ریاضی یا قدرت پیش‌بینی برای الگوریتم‌ها فراهم نمی‌کند.
- ستون‌های f7 و f8 از فضای ویژگی‌ها کنار گذاشته شدند. استخراج معانی عددی معنادار از این فیلدهای متنی آزاد، نیازمند تکنیک‌های پیشرفته Natural Language Processing (NLP)یا استفاده از Large Language Models (LLMs) است که خارج از محدوده مباحث این ترم از درس هستند (شکل ۲)

f7	f8
Lane blocked due to accident on I-95 Northbound at Exits 9 10A 6th Ave.	I-95 S
Accident on Bluefield Ave at 16th Ave.	N 16th Ave
Incident on I-495 EB near GRAND AVE Expect long delays.	Long Island Expy E
At I-95 - Accident.	Palm Beach Lakes Blvd
Accident on I-85 Southbound after Exit 92 SC-11.	I-85 S
Accident on Kenner Ave at US-70S Harding Pike.	Harding Pike
Left hand shoulder blocked due to accident on I-675 Southbound at Anvil Block Rd.	Anvil Block Rd
Accident now on the shoulder on 128 Southbound at 93.	I-93 N
Accident on US-30 Loucks Rd both ways from Lancer Ln to Fairlane Dr.	Fairlane Dr
Incident on I-65 SB near MM 16 Drive with caution.	I-65 S
At Kellogg Dr/Exit 41 - Accident.	I-10 E
Accident on 8th Ave Eastbound at Bush Blvd.	8th Ave W
Slow traffic on I-195 N from US-250/US-33/Broad St (VA-195/Downtown Expy/I-195) to Labu	I-195 N

(Figure 2) some values of f7 and f8 columns

3.2. Feature Engineering and Temporal Analysis

برای بهبود قدرت پیش‌بینی مدل، مهندسی ویژگی‌های زمانی (temporal feature engineering) انجام شد. یک ویژگی پیوسته جدید به نام duration_min ساخته شد تا مدت زمان سپری‌شده بین f13و f2 که هر دو متغیر timestamp هستند را اندازه‌گیری کند. این مقدار از تفاضل این دو timestamp به دست آمده و به دقیقه تبدیل شده است.

یک EDA انجام شد تا میزان کارایی این ویژگی مهندسی‌شده در پیش‌بینی متغیر هدف (y) بررسی شود. همان‌طور که توزیع‌های boxplot نشان می‌دهند (شکل ۳)

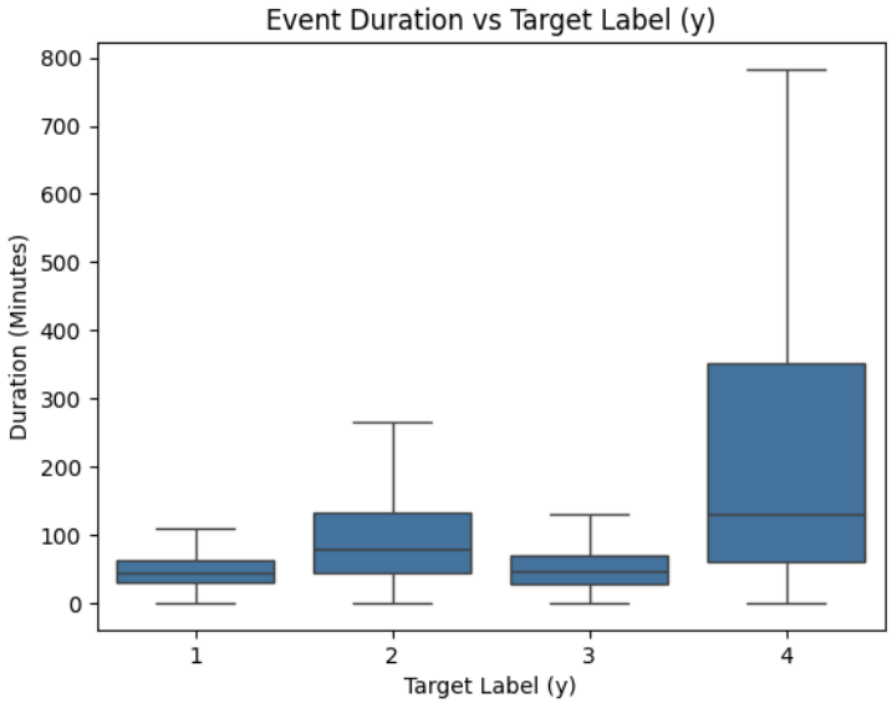


Figure 3

تخیر زمانی (temporal delay) تغییرات معناداری را در کلاس‌های مختلف نشان می‌دهد:

- **برچسب‌های ۱، ۲ و ۳ :**این کلاس‌ها معمولاً duration میانه کوتاه‌تری دارند و variance آن‌ها نسبتاً محدود است؛ به طوری که برچسب‌های ۱ و ۳ توزیع‌های بسیار مشابهی دارند.
- **برچسب ۴ :**این کلاس دارای یک interquartile range به‌طور چشمگیر وسیع‌تر و duration میانه بسیار بالاتری است

از آنجا که این متغیر ساخته‌شده تغییرات توزیعی مشخصی متناسب با متغیر هدف نشان داد، به عنوان یک ویژگی معنادار حفظ شد.

3.3. Temporal Feature Extraction

همان‌طور که می‌بینیم، ستون f13 حاوی metadata مفیدی است که می‌توانیم بررسی کنیم آیا اطلاعاتی برای پیش‌بینی متغیر هدف (y) دارد یا خیر.
در نتیجه، متغیر datetime به سه ویژگی متمایز و قابل‌تفسیر به لحاظ ریاضی تجزیه شد:

- **ساعت روز :(hour)** به عنوان یک عدد صحیح استخراج شد تا الگوریتم‌ها بتوانند الگوهای روزانه مثل ساعات اوج ترافیک صبحگاهی و عصرگاهی را تشخیص دهند.
- **روز هفته :(day_of_week)** به صورت یک عدد صحیح categorical استخراج شد تا رفتار رفت‌وآمدهای روزهای کاری از آخر هفته تفکیک شود.
- **فصل :(season)** همان‌طور که قابل حدس است، این ویژگی دست‌ساز برای این ضروری بود که تصادفات رانندگی به دلیل شرایط محیطی در فصل‌های مختلف تغییر می‌کنند (مثلاً هوای بارانی در پاییز، برف در زمستان و هوای آفتابی در تابستان). این نگاشت به صورت عددی تعریف شد: بهار (ماه‌های ۳-۵) به عنوان ۱، تابستان (ماه‌های ۶-۸) به عنوان ۲، پاییز (ماه‌های ۹-۱۱) به عنوان ۳ و زمستان (ماه‌های ۱۲، ۱ و ۲) به عنوان ۴ (این ادعا در بخش ۳.۴ ثابت می‌شود...)

اکنون pipeline ما dataset را با شاخص‌های چرخه‌ای و کاربردی غنی می‌کند. اگرچه این کار موقتاً تعداد کل ستون‌ها را افزایش می‌دهد، اما یک feature set بسیار توصیف‌گر فراهم می‌کند (این ادعا در مورد hour و day_of_week را می‌توان در بخش feature selection یعنی ۳.۱۴ ثابت کرد، اما برای هم‌راستایی با مشاهدات شهودی حفظ شده است.

ستون‌های timestamp خام یعنی f2 و f13 حذف شدند؛ زیرا اطلاعات زمانی حیاتی آن‌ها قبلاً به طور کامل استخراج شده و به ویژگی‌های duration_min، hour، day_of_week و season تبدیل شده بود.

3.4. Bivariate Analysis of Seasonal Trends

یک تحلیل دومتغیره (bivariate analysis) برای ارزیابی رابطه بین ویژگی جدید seasonو متغیر هدف (y) انجام شد. یک contingency table نرمال‌شده سطری (شکل ۴) محاسبه شد تا توزیع تناسبیseverity های تصادف در هر یک از چهار بخش فصلی بررسی شود.

y	1	2	3	4
season				
1	0.019418	0.766064	0.186306	0.028212
2	0.020251	0.746648	0.205400	0.027700
3	0.001172	0.815412	0.159126	0.024290
4	0.000280	0.836705	0.139527	0.023488

Figure 4

مشاهدات نشان می‌دهند که:

- داده‌ها دارای چولگی شدیدی هستند؛ به طوری که سطح severity شماره ۲ فارغ از فصل سال، اکثریت قاطع حوادث را تشکیل می‌دهد (بین ٪۷۴.۶ در فصل ۲ تا ٪۸۳.۶ در فصل ۴)
- تغییرات تناسبی متمایزی در طول فصل‌ها رخ می‌دهد. از همه مهم‌تر، برچسب ۱ در نیمه دوم چرخه زمانی به شدت کاهش می‌یابد. آن‌ها حدود ٪۱.۹ تا ٪۲.۰ از حوادث را در فصل‌های ۱ و ۲ تشکیل می‌دهند، اما به ترتیب به ٪۰.۱ و ٪۰.۰۲ در فصل‌های ۳ و ۴ کاهش می‌یابند.
- با کاهش فراوانی نسبی حوادث برچسب ۱ و ۳ در فصل‌های ۳ و ۴، تمرکز حوادث برچسب ۲ بیشتر شده و در ماه‌های زمستان (فصل ۴) به اوج خود می‌رسد.

این تفاوت‌های قابل‌مشاهده تأیید می‌کنند که ویژگی مهندسی‌شده season، تغییرات ظریف و غیرتصادفی توزیع severityهای تصادف را ثبت می‌کند. نشان دادن اینvariance ، ارتباط آماری این ویژگی را ثابت می‌کند.

3.5. Diurnal Feature Redundancy Assessment

برای بررسی بیشتر هم‌خطی (multicollinearity) احتمالی درdataset ، یک تحلیل بصری مقایسه‌ای روی ویژگی استخراج‌شده جدید hourدر مقابل چهار ویژگی categorical موجود(f31 ، f32، f33و f34که همگی وضعیت‌های «روز» در مقابل «شب» را کدگذاری می‌کنند، انجام شد.
ما فراوانی تصادفات را در چرخه ۲۴ ساعته، تفکیک‌شده بر اساس مقادیر categorical هر ویژگی نمایش می‌دهیم (شکل ۵)

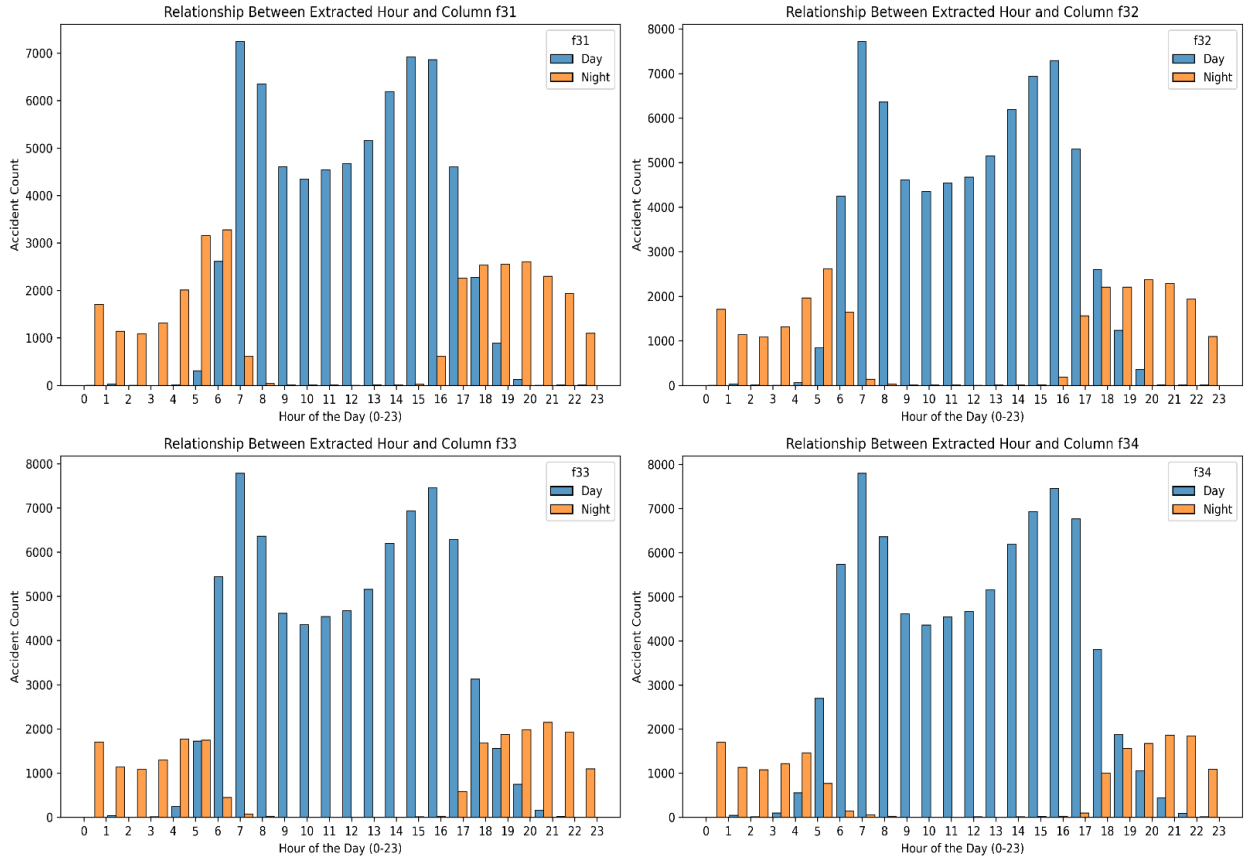


Figure 5

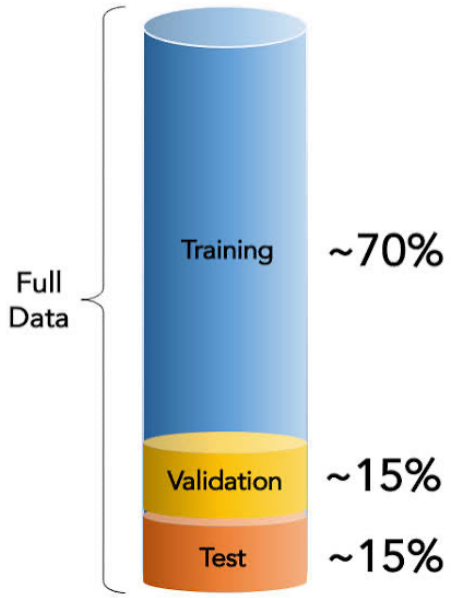
چهار نمودار فرعی (subplots) یکنواختی بالایی را نشان می‌دهند. نقاط انتقال بین دسته‌بندی‌های «روز» و «شب» تقریباً به طور کامل در سراسر f31، f32، f33و f34با هم مطابقت دارند (و دسته‌بندی‌های روشنایی روز حجم حوادث را بین ساعت ۰۷:۰۰ تا ۱۷:۰۰ تحت‌الشعاع قرار می‌دهند). بنابراین می‌توانیم آن‌ها را حذف کنیم.

3.6. deduplication

به عنوان آخرین اقدام در فرآیند پاکسازی اولیه داده‌ها، حذف داده‌های تکراری روی dataset اعمال شد.

3.7. Three-Way Dataset Partitioning

برای تضمین training مقاوم مدل و جلوگیری از data leakage در فازfeature selection ، یک پروتکل rigorous سه بخشی برای تقسیم dataset پیاده‌سازی شد. متغیرهای مستقل (X) از برچسب هدف (y) جدا شدند و داده‌ها به سه زیرمجموعه متمایز تقسیم شدند: یکtrain set ٪۷۰، یک validation set ٪۱۵ و یک test set ٪۱۵.
یک random seed ثابت (random_state=123) برای تضمین reproducibility کامل pipeline استفاده شد.



3.8. Statistical Evaluation of Missing Data and Imputation Strategy

برای مدیریت مقادیر گمشده (Missing Values) درdataset ، ابتدا یک بررسی آماری دقیق روی ستون‌های دارای مقادیر گمشده انجام شد. اگر ستونی دارای میزان زیادی از مقادیر گمشده بود، این فرضیه مطرح شد که شاید خود «گمشده بودن» اطلاعات معناداری درباره متغیر هدف (y) دارد (به این پدیده اصطلاحاً Missing Not At Random یا MNARمی‌گویند.(

برای اثبات این موضوع، از آزمون آماری **کای‌اسکوئر (Chi-Square Test)** استفاده شد:

- برای ستون‌هایی که بیش از ۲۵٪ مقادیر گمشده داشتند، ابتدا یک ستون شاخص دودویی جدید (مثل col_is_missing)ساخته شد که نشان می‌داد آیا مقداری در آن سطر گم شده است یا خیر.
- سپس جدول تقاطعی (Contingency Table) بین این وضعیت گمشده بودن و برچسب هدف (y) تشکیل شد.
- آزمون Chi-Square روی این جدول اجرا شد تا مقدار p-value محاسبه شود. آستانه معنی‌داری α=0.05 در نظر گرفته شد.

نتایج این ارزیابی:

- اگر مقدار p-value کمتر از ۰.۰۵ بود، یعنی گمشده بودنِ آن داده با متغیر هدف رابطه معناداری داشت؛ در نتیجه، ستون اصلی حفظ شد، مقادیر گمشده آن با مقدار میانه (median) یا یک مقدار جایگزین پر شدند و ستون شاخص (_is_missing) برای کمک به مدل نگ داشته شد.
- اگر مقدار p-valueبزرگ‌تر یا مساوی ۰.۰۵ بود، یعنی گمشده بودن داده ارتباطی با هدف نداشت و فقط باعث نویز می‌شود، بنابراین آن ستون به طور کامل از dataset حذف گردید.

این رویکرد آماری تضمین می‌کند که هیچ داده مفیدی بدون دلیل دور ریخته نشده و اطلاعات پنهان در پشت مقادیر گمشده به نفع مدل استخراج شده است.

3.9. Generalized Imputation for Low-Sparsity Features

ویژگی‌های باقی‌مانده که نیاز به imputation داشتند، به صورت برنامه‌نویسی‌شده به دو دسته numerical و categoricalتفکیک شدند تا مناسب‌ترین استراتژی‌های پر کردن اعمال شود:

- **Numerical Imputation:** ویژگی‌های numerical با استفاده از استراتژی میانگین (mean-fill) ایمپیت شدند. برای جلوگیری ازdata leakage ، میانگین آماری برای هر ستون منحصراً روی X_trainمحاسبه شد و سپس به طور یکنواخت برای transform کردنtrain ، validationو test sets اعمال گردید.
- **Categorical Imputation:** برای ویژگی‌های مبتنی بر string یاobject ، مقادیر گمشده با یک placeholder ثابت به نام «Unknown» جایگزین شدند. برای حفظ یکپارچگی داده‌ها، یک مکانیزم پویای جلوگیری از تداخل (collision-avoidance) پیاده‌سازی شد. الگوریتم مقادیر یکتای موجود در هر ستون categorical را اسکن می‌کند؛ اگر رشته متنی «Unknown» از قبل یک دسته معتبر در داده‌ها باشد، به طور تکرارپذیر یک پسوند عددی مثل «Unknown_1» اضافه می‌کند تا یک placeholder یکتا ساخته شود.

3.10. Comprehensive Numerical Imputation

برای تضمین پایداری ریاضی در فضای ویژگی‌ها قبل از training مدل، یک استراتژی global mean imputation روی تمام ویژگی‌های numerical باقی‌مانده اعمال شد. در حالی که رویه‌های آماری هدفمند قبلی متغیرهای دارای sparsityبالا را مدیریت کردند، این مرحله جامع به عنوان یک پشتیبان دقیق عمل می‌کند تا هرگونه missingness باقیمانده را در کل ماتریس عددی continuous و discrete خنثی کند.

3.11. One-Hot Encoding of Categorical Variables

برای امکان پردازش ریاضی متغیرهای متنی nominal توسط الگوریتم‌های پایین‌دستی، categorical encoding روی ویژگی‌های f1 و f12 اعمال شد. از استراتژی One-Hot Encoding (OHE) برای گسترش این متغیرها استفاده شد و هر حالت categorical یکتا به یک ستون شاخص دودویی متمایز تبدیل گردید که در آن ۱ نشان‌دهنده حضور دسته و ۰ نشان‌دهنده عدم حضور آن است.

3.12. Boolean Type Conversion

برای نهایی کردن یکپارچه‌سازی نوع داده‌هایdataset ، تمام شاخص‌های Boolean به صراحت به فرمت‌های integer استاندارد کست (cast) شدند. در حالی که برخی از فریم‌ورک‌های machine learning می‌توانند آرایه‌های True و Falseرا به طور ضمنی مدیریت کنند، نگاشت صریح این مقادیر به اعداد صحیح دودویی (۱ و ۰) سازگاری جهانی را در تمام مدل‌های ریاضی پایین‌دستی تضمین می‌کند.

3.13. Feature Scaling and Standardization

برای اطمینان از اینکه variance ویژگی‌هایی با مقیاس‌های ذاتا بزرگ‌تر، مدل‌های machine learning پایین‌دستی را به طور نامتناسبی تحت تأثیر قرار نمی‌دهد، standard scaling نرمال‌سازی Z-score روی تمام ستون‌های Inumerical عمال شد.

3.14. Dimensionality Reduction via Tree-Based Feature Importance

ابتدا با نحوه کارکرد یک Tree-based feature selection در کلاسیفایر Random Forest آشنا می‌شویم:
انتخاب ویژگی مبتنی بر درخت در یک کلاسیفایر Random Forest اندازه‌گیری می‌کند که هر ویژگی چقدر در کاهش عدم قطعیت یا «ناخالصی» یا impurity در تمام درخت‌های تصمیم‌گیری جنگل نقش دارد.

۱ .ارزیابی یک گره تک:(Gini Impurity)

در قلبRandom Forest ، درخت تصمیم قرار دارد. هر بار که یک درخت نیاز به انشعاب ایجاد یک split دارد، زیرمجموعه تصادفی از ویژگی‌ها را بررسی می‌کند و ارزیابی می‌کند کدام‌یک کلاس‌ها را بهتر از هم جدا می‌کند.

Gini Impurityاحتمال طبقه‌بندی نادرست یک داده جدید و تصادفی را در صورتی که برچسب کلاس تصادفی از آن گره به آن داده شود، اندازه‌گیری می‌کند. فرمول Gini Impurity به صورت زیر است:

$$Gini = 1 - \sum_{i=1}^{c} p_i^2$$

(where p_i is the probability of an item belonging to class i)

الگوریتم، ناخالصی گره والد را محاسبه می‌کند و سپس ناخالصی وزنی گره‌های فرزند را پس از یک split حساب می‌کند. تفاوت بین ناخالصی والد و ناخالصی فرزند، Information Gainاست مثلاً شکل ۶

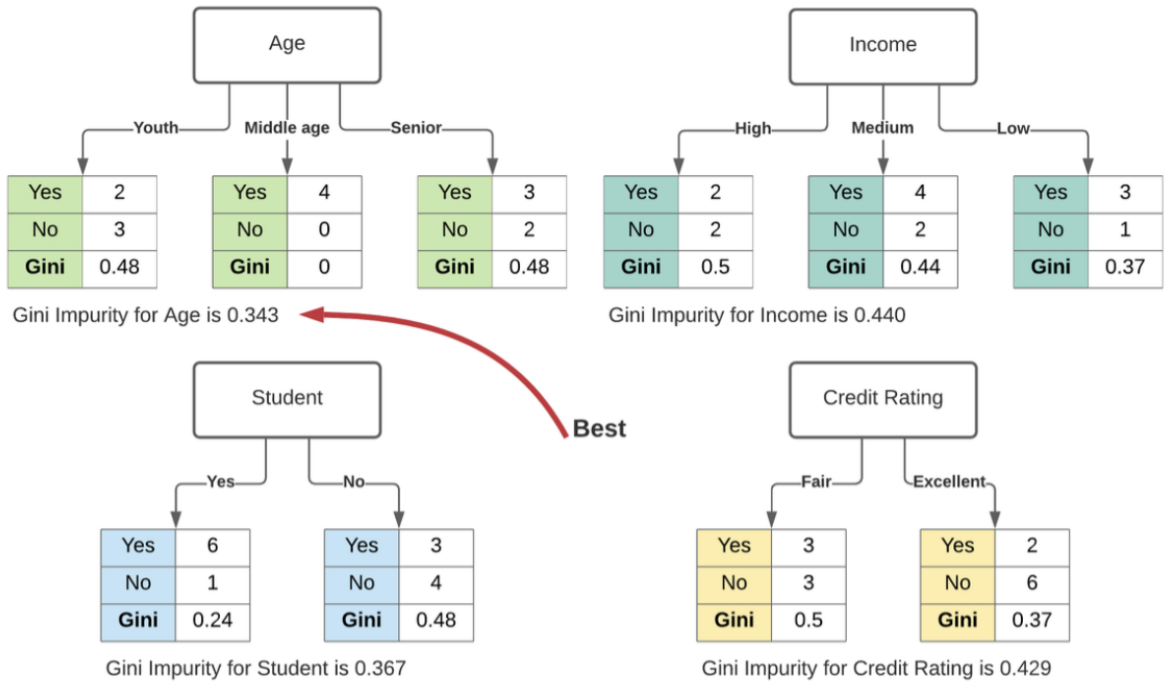


Figure 6

۲ .مقیاس‌دهی به یک درخت تک:

یک درخت تصمیم تک ده‌ها یا صدها split انجام می‌دهد. در حین ساخت، الگوریتم یک حساب جاری برای هر ویژگی نگه می‌دارد.

هر بار که «Feature A» برای ایجاد یک split استفاده می‌شود، الگوریتم Impurity Decrease را برای آن split خاص محاسبه می‌کند، آن را بر اساس تعداد نمونه‌هایی که از آن گره عبور کرده‌اند وزن‌دهی می‌کند و به امتیاز کل Feature Aبرای آن درخت اضافه می‌کند.

۳ .تجمیع در کل جنگل:(Mean Decrease in Impurity)

یک Random Forest شامل صدها درخت تصمیم مجزا است. از آنجا که هر درخت با استفاده از یک نمونه bootstrapتصادفی از داده‌ها و یک زیرمجموعه تصادفی از ویژگی‌ها در هر split ساخته می‌شود، مدل از تسلط یافتن هر ویژگی خاص بر کل فرآیند جلوگیری می‌کند.

برای به دست آوردن امتیاز نهاییfeature importance ، الگوریتم Mean Decrease in Impurity (MDI) را محاسبه می‌کند:

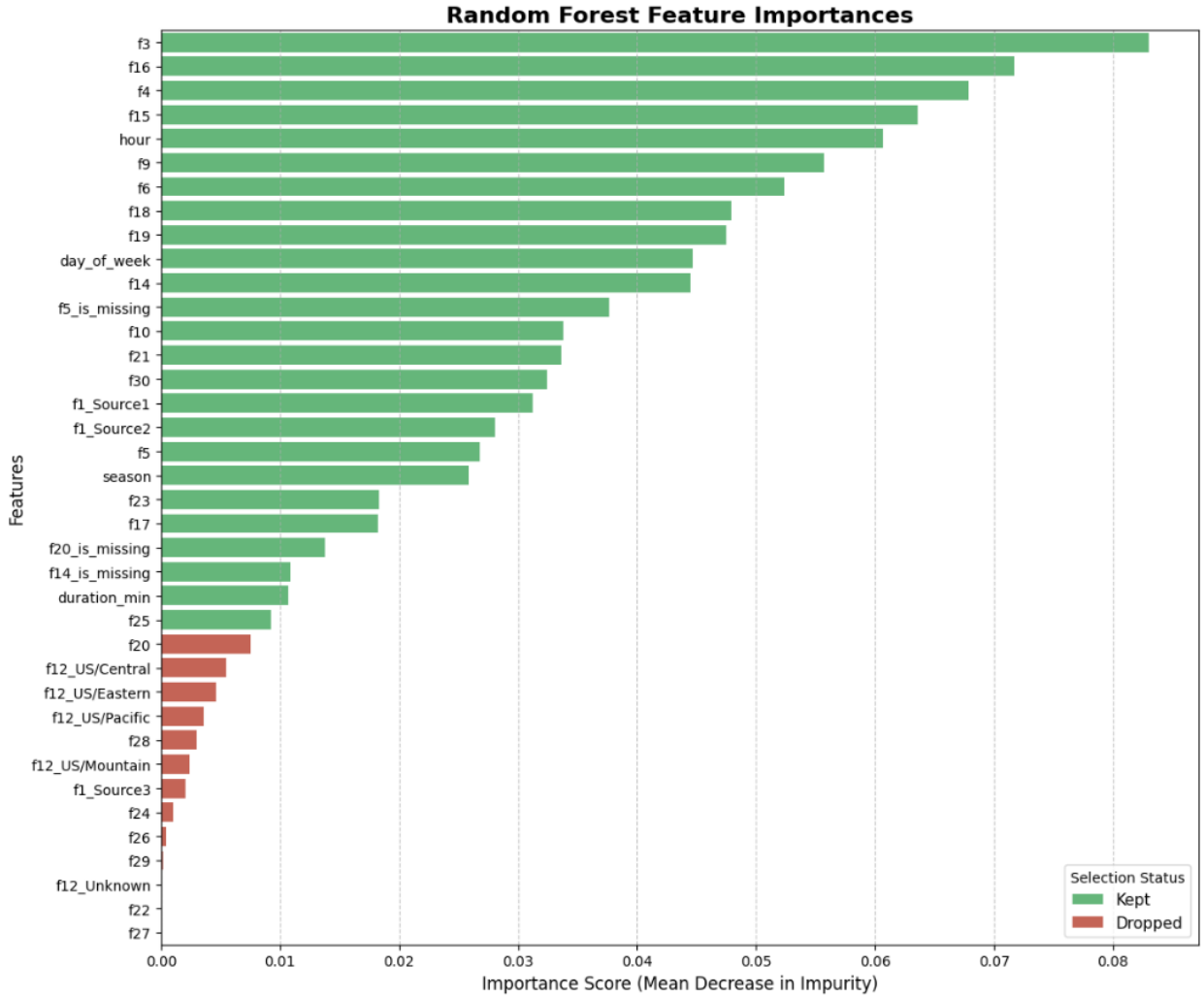
- کاهش ناخالصی کل منتسوب به «Feature A» را در هر درخت می‌گیرد، سپس آن امتیازها را در کل جنگل میانگین می‌گیرد و در نهایت امتیازها را نرمال می‌کند تا مجموع تمام اهمیت ویژگی‌ها برابر با ۱.۰ (یا ۱۰۰٪) شود.

۴ .انجام انتخاب:(Making the Selection)

پس از اتمام training مدل، آرایه‌ای از مقادیر بین 0 و 1 برای هر ویژگی در pipeline باقی می‌ماند.
برای انجام feature selection واقعی، ویژگی‌ها را به ترتیب نزولی مرتب می‌کنید. سپس می‌توانید ویژگی‌های با ضعیف‌ترین عملکرد را حذف کرده و فقط N ویژگی برتر را نگه دارید.

الگوریتم، متغیرهایی را که قدرت پیش‌بینی کمی درباره severity تصادف ارائه می‌دادند دور ریخت (شکل ۷:)

- **شاخص‌های زیرساختی :**چندین پرچم Boolean مانند f20، f22، f24، f26، f27، f28، f29
- **توسعه‌های :Categorical** متغیرهای One-Hot Encoded مشخصی که سیگنال کمی داشتند، مانند پرچم data source (f1_Source3)و تمام شاخص‌های time zone مجزا که از ویژگی اصلی f12مشتق شده بودند(US/Central) ، US/Eastern، US/Mountain، US/Pacificو حالت پیش‌فرض Unknown



با حذف این متغیرهای کم‌تأثیر، pipelineیک فضای ویژگی ناب، با سیگنال بالا و کاملاً عددی را نهایی می‌کند که کاملاً برای training و evaluation نهایی مدل آماده است.

نتایج و تحلیل

۱ .ارزیابی اولیه و معیارهای ارزیابی

از آنجا که دسته‌بندی شدت تصادفات (y) دارای نامتوازنی (Class Imbalance) است، تکیه صرف بر معیار Accuracy کافی نبود؛ لذا از **Weighted F1-Score**به عنوان معیار اصلی کیفیت نتایج استفاده شد. همچنین برای جلوگیری از بیش‌برازش (Overfitting) و ارزیابی منصفانه، مدل‌ها ابتدا روی داده‌های Train آموزش دیده‌اند، سپس بهترین تنظیمات با استفاده از Validation Set پیدا شده و در نهایت ارزیابی نهایی صورت گرفته است.

۲ .مقایسه جامع مدل‌ها با چارچوب تصمیم‌گیری چندمعیاره(MCDA)

برای انتخاب مدل نهایی، صرفاً به دقت تکیه نکردیم، بلکه ۵ معیار کلیدی شامل **عملکرد فنی(F1-Score)**، **تفسیرپذیری(Interpretability)** ، **سادگی و پایداری(Complexity & Robustness)** ، **کارایی محاسباتی (Execution Time)**و **سازگاری با ساختار داده (Data Alignment)** در قالب یک سیستم امتیازدهی وزنی (MCDA) ترکیب شدند.

- **Multinomial Logistic Regression**کمترین پیچیدگی و بیشترین تفسیرپذیری را داشت (امتیاز کامل در تفسیرپذیری)، اما به دلیل فرض خطی بودن، تطابق کمتری با روابط پیچیده و غیرخطی داده‌های جدول تصادفات داشت و امتیاز کمتری در Data Alignment کسب کرد.
- **Multi-Layer Perceptron (MLP)**توانایی یادگیری الگوهای پیچیده را داشت، اما به دلیل زمان آموزش طولانی‌تر، حساسیت بالا به مقیاس داده‌ها و ساختار جعبه‌سیاه(Black-box) ، امتیاز کمتری در بخش تفسیرپذیری و کارایی به دست آورد.
- **XGBoost** به عنوان بهترین گزینه انتخاب شد. این مدل با بهره‌گیری از ساختار درخت‌های تصمیم، تطابق فوق‌العاده‌ای با داده‌های جدولی (Tabular Data) دارد.

۳ .اثر Feature Selection بر روی عملکرد مدل‌ها

اجرای الگوریتم Feature Selection مبتنی برRandom Forest با محدود کردن ویژگی‌ها به حداکثر ۲۵ ویژگی مجاز تأثیر مثبتی بر روند کار داشت:

- حذف ویژگی‌های ضعیف (مانند پرچم‌های زیرساختی کم‌کاربرد و مناطق زمانی اضافی) باعث کاهش ابعاد داده و در نتیجه کاهش زمان آموزش مدل‌ها شد.
- حذف نویزهای غیرضروری به مدل‌ها کمک کرد تا روی ویژگی‌های کلیدی‌تر (مانند زمان، مشخصات مکانی و شرایط محیطی) تمرکز کنند که این موضوع بهبود پایداری و افت نکردن F1-Score را در پی داشت.

نتیجه‌گیری

در این پروژه، یک چرخه کامل و ساختاریافته از فرآیند داده‌کاوی روی مجموعه داده واقعی تصادفات رانندگی پیاده‌سازی شد. کارهای انجام‌شده را می‌توان به چند دسته کلیدی خلاصه کرد:

1. **شناخت و پاکسازی داده :**داده‌های اولیه از نظر مقادیر گمشده، ستون‌های ثابت یا غیرقابل پردازش بررسی و پاکسازی شدند. ویژگی‌های زمانی با استخراج ساعت، روز هفته و فصل غنی‌سازی شدند و duration تصادفات به عنوان یک ویژگی جدید اضافه شد.
2. **استخراج و کاهش ویژگی :**با استفاده از آزمون‌های آماری (مثل Chi-Square برای مقادیر گمشده) و الگوریتم‌های درخت تصمیم(Random Forest Feature Selection) ، ابعاد داده به ۲۵ ویژگی بهینه و باسیگنال بالا کاهش یافت تا از نفرین ابعاد (Curse of Dimensionality) جلوگیری شود.
3. **مدل‌سازی و تنظیم پارامترها :**سه مدل مختلفLogistic Regression) ، XGBoostو (MLP پیاده‌سازی شدند و پارامترهای آن‌ها با استفاده از روش Randomized Search روی مجموعه اعتبارسنجی (Validation)تنظیم شدند.
4. **انتخاب مدل بر اساس معیارهای چندگانه :(MCDA)** به جای تصمیم‌گیری صرفاً بر اساس یک عدد (مثل دقت)، مدل‌ها از جنبه‌های مختلفی نظیر تفسیرپذیری، زمان اجرا، پیچیدگی و سازگاری با داده ارزیابی شدند و در نهایت مدل **XGBoost** به عنوان مدل برتر و مناسب برای این مسئله انتخاب شد.

این پروژه مهارت‌های عملی ما را در زمینه تحلیل اکتشافی داده‌ها(EDA) ، پیش‌پردازش استاندارد، ارزیابی علمی و مستندسازی پروژه‌های یادگیری ماشین تقویت کرد.

مشارکت

محمد امین شیرمحمدی: بررسی و تحقیق مجموعه داده ، پیش پردازش و تمیز کردن داده، فرمت اولیه EDA، تشخیص data leak، پیاده سازی اولیه مدل ها، فرمت اولیه و تغییرات گزارش نهایی پروژه

امیررضا نعیمی: راه اندازی گیتهاب برای مشارکت در پروژه، اضافه کردن جزئیات و تغییرات کلی EDA، تغییرات و اضافه کردن جزِئیات مدل ها، تغییر missing value handling، بخش نتیجه گیری و تغییرات نهایی پروژه