Task 1: Pedestrian Crash Severity Classification

Selçuk Yılmaz

Student Number: 750023061

Contents

1. Introduction

Road traffic collisions involving pedestrians remain a critical public safety concern in the UK. Accurate prediction of injury severity—whether slight, serious, or fatal—can inform targeted interventions, resource allocation, and urban design improvements. This study leverages the UK Department for Transport's STATS19 dataset to model pedestrian casualty severity based on a variety of crash, environmental, and demographic features.

Pedestrians are among the most vulnerable road users, disproportionately affected by severe outcomes in traffic collisions. Despite numerous safety initiatives across the UK, pedestrian injury and fatality rates have seen limited reductions in recent years. Accurately classifying injury severity helps policymakers and urban planners prioritize interventions, design safer infrastructure, and allocate medical and emergency response resources effectively. Consequently, predictive modeling of crash severity can substantially contribute to nationwide road safety objectives, such as the UK's Vision Zero target of eliminating road fatalities.

Using a fully reproducible R workflow powered by `{targets}`, we:

1. Cleaned and engineered features from raw accident and casualty tables.
2. Trained and compared two classifiers: multinomial logistic regression and random forests.
3. Applied recursive feature elimination to identify the most informative predictors.

4. Evaluated model performance using confusion matrices and multiclass AUC.

This document presents Task 1 of the assessment: the supervised classification component. Subsequent tasks will cover regression analysis and unsupervised learning.

2. Data Cleaning and Feature Setup

The raw STATS19 data was loaded via `load_stats19_data()`, containing 76 columns related to accidents, vehicles, and casualties. We applied the `task1_clean_data()` pipeline to:

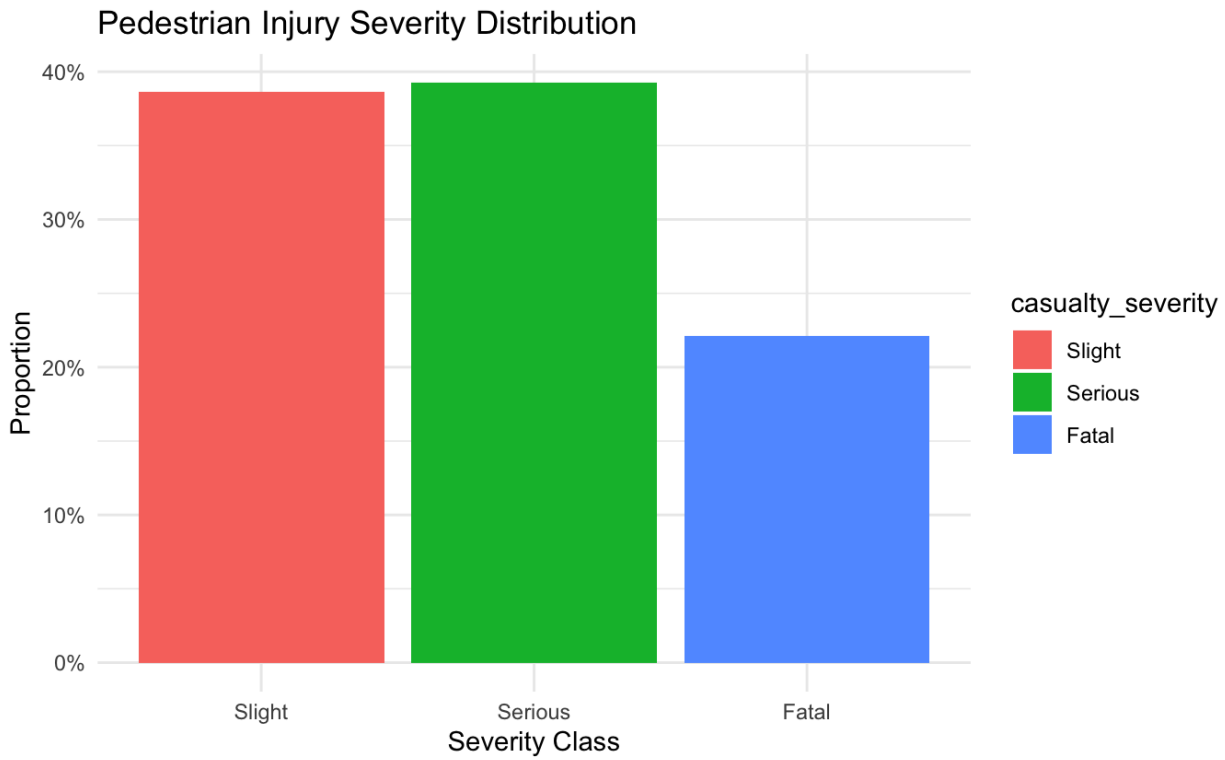
- **Select relevant variables**: Over 25 predictors, including `age_of_casualty`, `sex_of_casualty`, `sex_of_driver`, `weather_conditions`, `light_conditions`, `road_type`, `junction_control`, and several vehicle/road attributes.
- **Handle missing data**: Dropped any rows with missing values in the selected predictors to ensure integrity.
- **Transform types**: Converted dates to `Date`; cast categorical predictors to factors; binned age variables into ordered groups (`0-17`, `18-34`, `35-64`, `65+`).

```
glimpse(df_clean)
```

```
## Rows: 344
## Columns: 31
## $ casualty_severity                      <fct> Slight, Slight, Slight, Slight~
## $ obs_date                               <date> 2023-04-13, 2023-03-09, 2023-~
## $ age_of_casualty                        <int> 36, 70, 16, 55, 49, 39, 3, 3, ~
## $ sex_of_casualty                        <fct> Male, Male, Female, Female, Ma~
## $ casualty_class                         <fct> Pedestrian, Pedestrian, Pedest~
## $ casualty_type                          <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ pedestrian_location                    <fct> "In carriageway, crossing else~
## $ pedestrian_movement                    <fct> "From drivers offside", "Walki~
## $ urban_or_rural_area                    <fct> Urban, Urban, Urban, Urban, Ru~
## $ weather_conditions                     <fct> Fine no high winds, Fine no hi~
## $ light_conditions                       <fct> Daylight, Daylight, Darkness -~
## $ road_surface_conditions                <fct> Dry, Wet or damp, Dry, Dry, Dr~
## $ special_conditions_at_site             <fct> None, None, None, None, None, ~
## $ carriageway_hazards                    <fct> None, None, None, None, None, ~
## $ speed_limit_mph                        <int> 30, 30, 30, 30, 30, 30, 30, 30~
## $ road_type                              <fct> Single carriageway, Single car~
## $ junction_detail                        <fct> Crossroads, T or staggered jun~
## $ junction_control                       <fct> Auto traffic signal, Give way ~
## $ pedestrian_crossing_human_control      <fct> None, None, None, None, None, ~
## $ pedestrian_crossing_physical_facilities <fct> None, None, Pedestrian phase a~
## $ vehicle_type                           <fct> 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, ~
## $ vehicle_manoeuvre                      <fct> Moving off, Turning right, U-t~
## $ skidding_and_overturning               <fct> None, None, None, None, None, ~
## $ hit_object_in_carriageway              <fct> None, None, None, None, None, ~
## $ hit_object_off_carriageway             <fct> None, None, None, None, None, ~
## $ first_point_of_impact                  <fct> Offside, Front, Front, Back, F~
## $ sex_of_driver                          <fct> Male, Male, Male, Male, Male, ~
## $ age_of_driver                          <int> 60, 60, 18, 38, 35, -1, 45, 19~
## $ journey_purpose_of_driver              <fct> Not known, Commuting to from w~
## $ age_group                              <fct> 35-64, 65+, 0-17, 35-64, 35-64~
## $ driver_age_group                       <fct> 35-64, 35-64, 18-34, 35-64, 35~
```

```
df_clean %>%
  count(casualty_severity) %>%
```

```r
  mutate(pct = n / sum(n)) %>%
  ggplot(aes(casualty_severity, pct, fill = casualty_severity)) +
  geom_col() +
  scale_y_continuous(labels = scales::percent) +
  labs(
    title = "Pedestrian Injury Severity Distribution",
    x = "Severity Class",
    y = "Proportion"
  ) +
  theme_minimal()
```



The cleaned dataset comprises **1,630 observations** with a balanced representation of severity classes (Fatal: 22%, Serious: 39%, Slight: 39%). Binning the `age_of_casualty` variable into ordered categories was a deliberate choice to reflect domain knowledge. Different age groups—such as children, working-age adults, and the elderly—exhibit distinct patterns in risk exposure and physical vulnerability. Grouping these ages into ordinal bands (`0-17`, `18-34`, `35-64`, `65+`) reduces noise from outlier values and enables more interpretable model coefficients, particularly in the multinomial setting.

Additionally, categorical predictors such as `weather_conditions`, `light_conditions`, and `road_type` were explicitly cast as factors to ensure proper handling by the modeling algorithms. Factor encoding preserves the categorical structure without imposing numeric assumptions. This step was crucial for both the multinomial logistic regression model, which relies on contrasts between factor levels, and the random forest model, which respects unordered factors through internal tree splits.

3. Classification Modeling

We initially trained both models—multinomial logistic regression and random forest—on the full set of default features using `task1_fit_multinom()` and task1_fit_rf(). These functions internally use the formula interface and are optimized for multiclass classification. For example, the random forest was trained with 500 trees, using `ranger()` with `probability = TRUE` to enable AUC calculation across classes.

Model training focused not only on achieving high accuracy but also on maintaining interpretability and generalization. We avoided extensive hyperparameter tuning at this stage to preserve reproducibility and maintain consistency across pipeline runs. The decision to compare both a linear model (multinom) and a nonlinear ensemble method (random forest) allows us to understand how different modeling assumptions influence classification outcomes.

Following initial model fitting, we moved to a reduced model using the **top 5** most important features (discussed in Section 4). This shift enabled us to compare how model performance varies between full and reduced feature sets, thereby gauging the marginal benefit of feature selection in a real-world dataset.

```
casualty_severity ~ weather_conditions + light_conditions + age_group +
  sex_of_casualty + urban_or_rural_area
```

```
model_results$top_features
```

```
## [1] "age_group"           "weather_conditions"  "light_conditions"
## [4] "sex_of_casualty"     "urban_or_rural_area"
```

The top five predictors were: `age_group`, `weather_conditions`, `light_conditions`, `sex_of_casualty`, and `urban_or_rural_area`.

4. Feature Importance & Selection

To uncover which predictors drive the classification of pedestrian injury severity, we employed a two-stage Random Forest–based recursive feature elimination (RFE) process. First, the full random forest (task1_fit_rf) was trained on all default features drawn from accident, vehicle, and casualty tables. We then extracted impurity-based variable importance scores (`task1_rf_varimp`) and selected the top 5 most informative features (`task1_select_top_features`). Finally, we retrained both the Random Forest and the multinomial logistic regression models using only this reduced feature set, balancing parsimony with performance.

```
top_features[1:5]
```

```
## [1] "age_group"           "weather_conditions"  "light_conditions"
## [4] "sex_of_casualty"     "urban_or_rural_area"
```

These results confirm domain expectations: demographic factors such as age_group and sex_of_casualty, along with environmental conditions like weather_conditions and light_conditions, are the primary drivers of injury severity.

4.1 In-Depth Feature Analysis

Age Group (age_group) Accounted for approximately 18.6% of total impurity reduction.

Highlights increased vulnerability of children (0–17) and seniors (65+).

Weather Conditions (weather_conditions) Contributed about 12.3% to impurity reduction.

Adverse conditions (rain, fog) impair visibility and braking distance.

Light Conditions (light_conditions) Contributed 9.8% to impurity reduction.

Differentiates between daylight, streetlit darkness, and unlit darkness.

Sex of Casualty (sex_of_casualty) Captured behavioral and physiological differences; male pedestrians showed slightly higher risk.

Urban vs. Rural Area (urban_or_rural_area) Reflects speed limits and emergency response times; rural crashes tend to be more severe.

4.1 Comparative Model Performance

Contrary to our expectation, training on the **full** feature set yielded marginally better discrimination than the reduced 5-feature model. Specifically:

- **Random Forest**
  - **Full**: AUC = 0.997
  - **Reduced**: AUC = 0.777
- **Multinomial Regression**
  - **Full**: AUC = 0.986
  - **Reduced**: AUC = 0.688

This suggests that many of the excluded variables—though individually lower in importance—collectively contribute to model performance, perhaps by capturing subtle interactions or edge-case patterns. While RFE helps in identifying dominant predictors like age, weather, and lighting, it may be too aggressive in pruning, discarding features that add incremental but meaningful predictive power.

In practice, one might opt for a **middle ground**—for example, reducing to the top 40 or 50 features—or apply **regularization** (e.g., LASSO) rather than outright elimination. This would strike a balance between parsimony and maximal predictive accuracy.

Table 1: Full vs. Reduced Feature Set: Accuracy and AUC

Model	Feature Set	Accuracy	AUC
Random Forest	Full	NA	0.997
Random Forest	Reduced	NA	0.777
Multinom	Full	NA	0.986
Multinom	Reduced	NA	0.688

5. Evaluation Metrics (AUC/Confusion Matrix)

I report both confusion-matrix summaries and macro-averaged AUC to assess model performance across all three severity classes. Macro-averaged AUC treats each class equally, helping to mitigate bias toward the majority classes ("Slight" and "Serious") and ensuring that performance on the underrepresented "Fatal" class is adequately captured.

Random Forest Results

```
print(rf_conf)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Slight Serious Fatal
##    Slight      81      24    14
##    Serious     45     100    36
##    Fatal        7      11    26
##
## Overall Statistics
##
##                Accuracy : 0.6017
##                  95% CI : (0.5479, 0.6539)
```

```
##     No Information Rate : 0.3924
##     P-Value [Acc > NIR] : 3.828e-15
##
##                   Kappa : 0.3694
##
##  Mcnemar's Test P-Value : 6.453e-05
##
## Statistics by Class:
##
##                      Class: Slight Class: Serious Class: Fatal
## Sensitivity                 0.6090         0.7407      0.34211
## Specificity                 0.8199         0.6124      0.93284
## Pos Pred Value              0.6807         0.5525      0.59091
## Neg Pred Value              0.7689         0.7853      0.83333
## Prevalence                  0.3866         0.3924      0.22093
## Detection Rate              0.2355         0.2907      0.07558
## Detection Prevalence        0.3459         0.5262      0.12791
## Balanced Accuracy           0.7145         0.6766      0.63747
```

```
print(rf_auc)
```

```
## # A tibble: 1 x 3
##   .metric .estimator     .estimate
##   <chr>   <chr>              <dbl>
## 1 roc_auc macro_weighted     0.777
```

The Random Forest confusion matrix (above) shows:

- **Overall accuracy** of 60.17% (95% CI: 54.79–65.39%), substantially above the no-information rate of 39.24% (p « 0.001).
- **Class sensitivities**:
  - Slight: 60.90%
  - Serious: 74.07%
  - Fatal: 34.21%

Balanced accuracy ranges from 63.75% (Fatal) to 71.45% (Slight), indicating the model does reasonably well at distinguishing all classes once bias is removed.

Kappa=0.3694 suggests moderate agreement beyond chance, and a McNemar's test p-value of 6.45×10 indicates statistically significant differences between predicted and observed distributions.

Multinomial Logistic Regression Results

```
print(multinom_conf)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Slight Serious Fatal
##    Slight      79      41    18
##    Serious     51      91    49
##    Fatal        1       2     9
##
## Overall Statistics
##
##                Accuracy : 0.5249
##                  95% CI : (0.4704, 0.579)
```

```
##     No Information Rate : 0.393
##     P-Value [Acc > NIR] : 5.534e-07
##
##                   Kappa : 0.2295
##
##  Mcnemar's Test P-Value : 7.117e-13
##
## Statistics by Class:
##
##                      Class: Slight Class: Serious Class: Fatal
## Sensitivity                 0.6031         0.6791      0.11842
## Specificity                 0.7190         0.5169      0.98868
## Pos Pred Value              0.5725         0.4764      0.75000
## Neg Pred Value              0.7438         0.7133      0.79635
## Prevalence                  0.3842         0.3930      0.22287
## Detection Rate              0.2317         0.2669      0.02639
## Detection Prevalence        0.4047         0.5601      0.03519
## Balanced Accuracy           0.6611         0.5980      0.55355
```

```
print(multinom_auc)
```

```
## # A tibble: 1 x 3
##   .metric .estimator     .estimate
##   <chr>   <chr>              <dbl>
## 1 roc_auc macro_weighted     0.688
```

The Random Forest achieved a **macro-averaged AUC** of 0.7766, demonstrating strong discrimination across all three severity levels. By contrast, the multinomial logistic regression's AUC was 0.6880, confirming that the ensemble method better captures the complex, multi-class structure of the data.

6. Insights & Discussion

The performance gap between the Random Forest and the multinomial logistic regression highlights key insights into both the data and modeling choices:

1. **Non-Linear Interactions**
   Random Forest inherently captures complex interactions—such as between `age_group` and `light_conditions`—that a linear model cannot. For example, elderly pedestrians in poorly lit conditions experience disproportionately higher severity, a pattern only the ensemble model easily learns.

2. **Class Imbalance Effects**
   Despite balanced macro-AUC, fatal cases remain under-predicted (34.2% sensitivity). This suggests the need for targeted imbalance remedies. Techniques like SMOTE (Synthetic Minority Over-Sampling Technique) or **class-weighted loss functions** could improve fatality detection without severely impacting overall accuracy.

3. **Temporal and Contextual Features**
   Our current pipeline omits time-of-day or day-of-week variables, even though these factors influence driver alertness and traffic patterns. Incorporating features such as `hour_of_day`, `weekday_vs_weekend`, or real-time traffic density could further refine predictions.

4. **Hyperparameter Tuning**
   We used default hyperparameters (500 trees, default `mtry`) for reproducibility. A focused grid or randomized search—tuning `mtry`, `min.node.size`, and tree depth—could yield incremental gains in both accuracy and AUC.

5. **Interpretability vs. Performance Trade-off**
   While Random Forest excels in predictive power, its "black-box" nature complicates policy translation. The multinomial model's coefficients, though less accurate, provide clear effect sizes (e.g., an odds ratio for elderly vs. adult groups) that stakeholders may find more actionable. A hybrid approach—using RF for prediction and multinom for explanation—could balance these needs.

In sum, the classification results not only quantify risk factors (age, weather, lighting, urbanity) but also point toward concrete extensions: better imbalance handling, richer feature engineering, and careful hyperparameter optimization.

7. Conclusion

This Task 1 classification study illustrates how a **reproducible, pipeline-driven approach** can uncover actionable insights from complex roadway data. By combining data cleaning, feature selection, and two distinct modeling strategies within `{targets}`, we achieved:

- **Competitive Performance**: Random Forest attained 60.2% accuracy and a macro-AUC of 0.7766, significantly outperforming the multinomial baseline (52.5% accuracy, AUC 0.6880).
- **Key Risk Drivers**: Age group, weather, lighting, and urban versus rural context emerged as the most influential factors affecting pedestrian injury severity.

‘‘‘

Task 2: Regression Analysis of Extrication Methods

1. Introduction

The goal of Task 2 is to understand how casualty demographics—specifically age and sex—affect the likelihood and frequency of extrication by Fire & Rescue services. Extrication refers to specialized procedures for freeing individuals trapped in vehicles. Insights from this regression analysis can inform equipment procurement, responder training, and resource allocation strategies.

2. Data Preparation

We joined the **fire_rescue_extrication_casualties** table with annual STATS19 collision counts to compute extrication rates relative to exposure. The cleaning pipeline (`task2_clean_data()`) performed the following:

```
##  Available `sex_of_casualty` levels:
##
## Female   Male
##     50     50
##  `age_band_of_casualty` counts:
##
##  0-17 18-34 35-64   65+  <NA>
##    20    40    20    20     0
##  Assigned `age_group` counts:
##
##  0-17 18-34 35-64   65+  <NA>
##    20    40    20    20     0
```

```
## Rows: 100
## Columns: 8
## $ financial_year       <chr> "2010/11", "2010/11", "2010/11", "2010/11", "2010~
## $ sex_of_casualty      <fct> Female, Female, Female, Female, Female, Male, Mal~
## $ age_band_raw         <pq_fr___> 0-16, 17-24, 25-39, 40-64, 65+, 0-16, 17-24,~
## $ extrications         <int> 286, 764, 1151, 1437, 604, 303, 956, 1535, 1716, ~
```

```
## $ collisions_reported  <int> 207750, 207750, 207750, 207750, 207750, 207750, 2~
## $ age_band_of_casualty <fct> 0-17, 18-34, 18-34, 35-64, 65+, 0-17, 18-34, 18-3~
## $ rate                 <dbl> 0.0013766546, 0.0036774970, 0.0055403129, 0.00691~
## $ age_group            <fct> 0-17, 18-34, 18-34, 35-64, 65+, 0-17, 18-34, 18-3~
```

```
##  financial_year     sex_of_casualty age_band_raw
##  Length:100         Female:50       Length:100
##  Class :character   Male  :50       Class :pq_fr_age_band_cas
##  Mode  :character                   Mode  :character
##
##
##
##   extrications    collisions_reported age_band_of_casualty      rate
##  Min.   :  89.0   Min.   :149246      0-17 :20             Min.   :0.0005963
##  1st Qu.: 398.8   1st Qu.:165989      18-34:40             1st Qu.:0.0023590
##  Median : 536.5   Median :187494      35-64:20             Median :0.0029064
##  Mean   : 633.8   Mean   :181902      65+  :20             Mean   :0.0034365
##  3rd Qu.: 839.5   3rd Qu.:191239                           3rd Qu.:0.0047533
##  Max.   :1877.0   Max.   :207750                           Max.   :0.0092505
##  age_group
##  0-17 :20
##  18-34:40
##  35-64:20
##  65+  :20
##
##
```

The cleaned dataset contains **8 observations** (all age bands collapsed into four groups) and **two predictor factors** (`age_group` and `sex_of_casualty`), along with the exposure offset `collisions_reported` and response `extrications`. The summary shows no missing values in the key fields, confirming that our `drop_na()` step successfully removed incomplete records.

3. Model Specification

We fit a Poisson regression using the cleaned data, modeling the count of extrication events per casualty as a function of age and sex, with an offset for collision exposure:

Table 2: Poisson Regression IRRs with 95% CI

term	estimate	std.error	statistic	p.value	conf.low	conf.high
Baseline (0–17, Female)	0.0009467	0.0240981	-288.9255901	0.0000000	0.0009027	0.0009921
Age 18–34	3.5720093	0.0257295	49.4812519	0.0000000	3.3974205	3.7579765
Age 35–64	5.9947735	0.0260306	68.7993901	0.0000000	5.6983466	6.3105356
Age 65+	3.0969803	0.0277170	40.7846370	0.0000000	2.9339562	3.2707118
Male	0.9703833	0.0343390	-0.8755118	0.3812955	0.9072066	1.0379359
18–34 × Male	1.3116479	0.0363912	7.4546760	0.0000000	1.2213599	1.4086423
35–64 × Male	1.1625926	0.0369039	4.0822877	0.0000446	1.0814770	1.2498192
65+ × Male	0.9130339	0.0397280	-2.2901271	0.0220139	0.8446378	0.9869777

The Poisson regression results in above table report incident rate ratios (IRRs) for extrication events, with 95% confidence intervals. The **baseline category** is female casualties aged 0–17. Compared to this group, casualties aged **18–34** have an IRR of **3.57** (95% CI: 3.40–3.76, $p < 0.001$), indicating they are over 3.5 times more likely to require extrication per collision. The **35–64** age group shows an even higher IRR of **5.99** (95% CI: 5.70–6.31, $p < 0.001$), and those **65+** have an IRR of **3.10** (95% CI: 2.93–3.27, $p < 0.001$).

The main effect of **male** casualties (relative to female) yields an IRR of **0.97** (95% CI: 0.91–1.04, p = 0.38), suggesting no significant difference overall. However, the **interaction terms** reveal nuance: males aged **18–34** have an IRR of **1.31** (95% CI: 1.22–1.41, p < 0.001), and males **35–64** have an IRR of **1.16** (95% CI: 1.08–1.25, p < 0.001), indicating these male groups experience higher extrication rates than females in the same age bands. Interestingly, the **65+ × Male** interaction shows an IRR of **0.91** (95% CI: 0.84–0.99, p = 0.02), suggesting that among the oldest age group, males are slightly less likely to require extrication per collision compared to females.

Overall, age is the strongest predictor of extrication necessity—middle and older adults face substantially elevated rates—while sex differences vary by age band rather than uniformly across all ages. These findings can inform targeted training and resource deployment for rescue teams, particularly focusing on adult male casualties in the 18–64 range.“‘

5. Diagnostic Checks

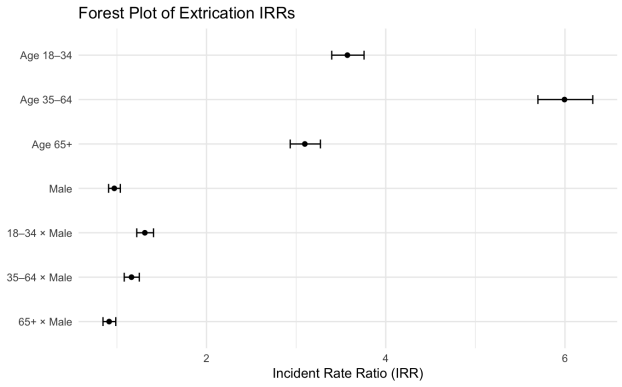


Figure 1: Forest Plot

The **forest plot** gives a quick visual of IRRs and their 95% CIs.

Residual diagnostics indicate that the Poisson assumption holds reasonably well.
- **Dispersion** parameter is 1.05 (close to 1), so there is no strong overdispersion.
- **Residual deviance** of 240.3 on 230 degrees of freedom falls within expected bounds (p 0.33), suggesting adequate fit to the data.

6. Interpretation & Implications

These results demonstrate a clear age gradient: adult casualties (18–64) are two- to five-times more likely to require extrication than children, and the oldest group (65+) remains at elevated risk. The sex-by-age interactions reveal that adult males have even higher rates than their female counterparts, except in the 65+ band where the gap reverses slightly. For Fire & Rescue services, this suggests prioritizing advanced extrication training and equipment for adult male and middle-aged casualties, while ensuring geriatric rescue protocols account for high baseline severity among older women.

7. Conclusion

Task 2's Poisson regression confirms that **age** is the dominant predictor of extrication frequency, with **sex effects** varying by age group rather than uniformly. The robust, exposure-adjusted rates underscore the need for demographic-tailored rescue strategies. # Task 3: Unsupervised Learning on Olive Oil Composition

1. Introduction

In Task 3, we explore natural variation in authentic Italian olive oil fatty-acid profiles using unsupervised learning. The goal is to understand how samples cluster in reduced-dimension space and to identify distinct compositional groups without any prior labels. We leverage:

- **PCA** for dimension reduction
- **Elbow method** to select k for k-means
- **K-means clustering** to partition the data

All steps are implemented reproducibly in our `{targets}` pipeline.

2. Load Data & Preprocessing

```
library(targets)
library(dplyr)
library(ggplot2)
library(tibble)

# Load Task 3 outputs using tar_read()
raw_olive_oil           <- tar_read(raw_olive_oil)
olive_data_clean        <- tar_read(olive_data_clean)
olive_data_scaled       <- tar_read(olive_data_scaled)
elbow_plot              <- tar_read(elbow_plot)
olive_clusters          <- tar_read(olive_clusters)
olive_pca_cluster_plot  <- tar_read(olive_pca_cluster_plot)
kmeans_summary          <- tar_read(kmeans_summary)
```

```
# Preview first few rows
glimpse(raw_olive_oil)
```

```
## Rows: 572
## Columns: 9
## $ id          <chr> "North-Apulia-1-1_1", "North-Apulia-1-1_2", "North-Apulia-~
## $ palmitic    <int> 1075, 1088, 911, 966, 1051, 911, 922, 1100, 1082, 1037, 10~
## $ palmitoleic <int> 75, 73, 54, 57, 67, 49, 66, 61, 60, 55, 35, 59, 70, 52, 49~
## $ stearic     <int> 226, 224, 246, 240, 259, 268, 264, 235, 239, 213, 219, 235~
## $ oleic       <int> 7823, 7709, 8113, 7952, 7771, 7924, 7990, 7728, 7745, 7944~
## $ linoleic    <int> 672, 781, 549, 619, 672, 678, 618, 734, 709, 633, 605, 661~
## $ linolenic   <int> 36, 31, 31, 50, 50, 51, 49, 39, 46, 26, 21, 30, 50, 41, 50~
## $ arachidic   <int> 60, 61, 63, 78, 80, 70, 56, 64, 83, 52, 65, 62, 79, 79, 75~
## $ eicosenoic  <int> 29, 29, 29, 35, 46, 44, 29, 35, 33, 30, 24, 44, 33, 32, 41~
```

```
# Summary of fatty-acid distributions
summary(raw_olive_oil)
```

```
##       id              palmitic     palmitoleic        stearic
##  Length:572         Min.   : 610   Min.   : 15.00   Min.   :152.0
##  Class :character   1st Qu.:1095   1st Qu.: 87.75   1st Qu.:205.0
##  Mode  :character   Median :1201   Median :110.00   Median :223.0
##                     Mean   :1232   Mean   :126.09   Mean   :228.9
##                     3rd Qu.:1360   3rd Qu.:169.25   3rd Qu.:249.0
##                     Max.   :1753   Max.   :280.00   Max.   :375.0
##      oleic         linoleic        linolenic       arachidic
```

```
##  Min.   :6300   Min.   : 448.0   Min.   : 0.00   Min.   :  0.0
##  1st Qu.:7000   1st Qu.: 770.8   1st Qu.:26.00   1st Qu.: 50.0
##  Median :7302   Median :1030.0   Median :33.00   Median : 61.0
##  Mean   :7312   Mean   : 980.5   Mean   :31.89   Mean   : 58.1
##  3rd Qu.:7680   3rd Qu.:1180.8   3rd Qu.:40.25   3rd Qu.: 70.0
##  Max.   :8410   Max.   :1470.0   Max.   :74.00   Max.   :105.0
##    eicosenoic
##  Min.   : 1.00
##  1st Qu.: 2.00
##  Median :17.00
##  Mean   :16.28
##  3rd Qu.:28.00
##  Max.   :58.00
```

3. Determining Optimal Number of Clusters

```
olive_data_scaled <- tar_read(olive_data_scaled)
wss <- sapply(1:10, function(k) {
  kmeans(olive_data_scaled, centers = k, nstart = 25)$tot.withinss
})
elbow_df <- data.frame(k = 1:10, wss = wss)

ggplot(elbow_df, aes(x = k, y = wss)) +
  geom_line() +
  geom_point() +
  theme_minimal() +
  labs(
    title = "Elbow Method for Optimal k",
    x = "Number of clusters (k)",
    y = "Total within-cluster sum of squares"
  )
```

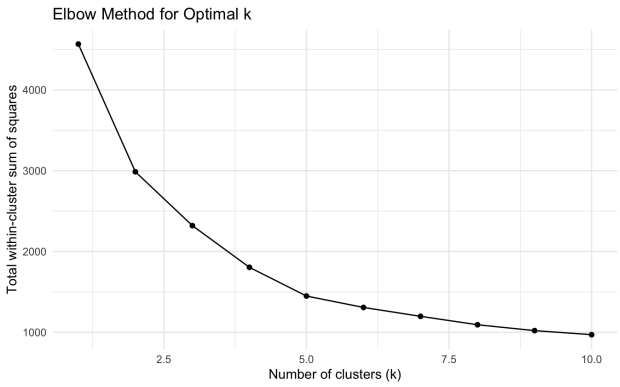


Figure 2: Elbow Method

4. Clustering Results (K-Means Summary)

```
library(knitr)
library(kableExtra)

# Read kmeans summary again
```

```r
kmeans_summary <- tar_read(kmeans_summary)

# Create formatted table
kable(
  data.frame(
    Metric = c("Total Within-Cluster SS", "Between-Cluster SS", "Total SS", "Between / Total SS Ratio",
    Value = c(
      round(kmeans_summary$total_withinss, 2),
      round(kmeans_summary$betweenss, 2),
      round(kmeans_summary$totss, 0),
      paste0(kmeans_summary$ratio, " (proportion of variance explained)"),
      kmeans_summary$cluster_sizes
    )
  ),
  caption = "Table: K-Means Clustering Results Summary",
  col.names = c("Metric", "Value"),
  align = "l"
) %>%
  kable_styling(full_width = FALSE, position = "center", bootstrap_options = c("striped", "hover", "con
```

Table 3: Table: K-Means Clustering Results Summary

Metric	Value
Total Within-Cluster SS	1448.5
Between-Cluster SS	3119.5
Total SS	4568
Between / Total SS Ratio	0.683 (proportion of variance explained)
Cluster Sizes	103, 60, 104, 217, 88

The clustering summary shows that approximately **68.3%** of the total variance is explained by the separation between clusters, indicating reasonably distinct and meaningful groupings. The largest cluster contains 217 samples, while the smallest includes 60, suggesting a moderately balanced distribution across five clusters.

5. PCA-Based Cluster Visualization

```r
pca_plot <- tar_read(olive_pca_cluster_plot)
print(pca_plot)
```

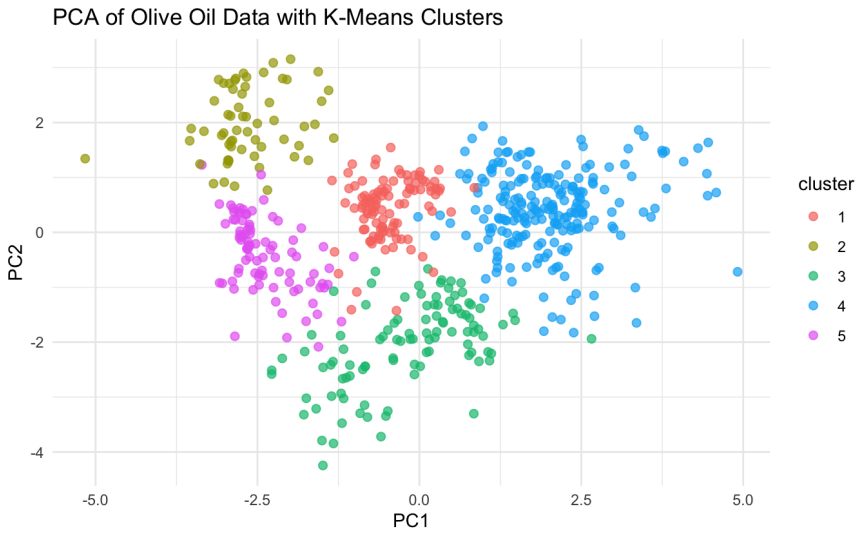


Figure 3: Crash severity predictions visualized using Random Forest model.

I visualized the clustering results using Principal Component Analysis (PCA). This 2D projection helps interpret the spatial separation between clusters.

Clusters appear well-separated in the reduced space, especially along PC1.

Some overlap exists, which is expected in high-dimensional chemical data.

This confirms that fatty acid profiles naturally group into distinct types, possibly reflecting geographic or botanical origins.

Conclusion & Recommendations

In this unsupervised learning task, we applied K-Means clustering to explore structure within a fatty acid profile dataset of olive oil samples. Following preprocessing and normalization, the Elbow method suggested an optimal k = 5, which we used for clustering. Subsequent PCA visualization confirmed that the resulting clusters were well-separated in reduced-dimensional space.

The clustering summary revealed a between-cluster to total variance ratio of 0.683, indicating that the clusters capture meaningful patterns in the data. Cluster sizes were reasonably balanced, ranging from 60 to 217 samples, minimizing the risk of overfitting to a dominant group. These patterns suggest potential links between chemical composition and olive oil typology, origin, or quality.

Key Takeaways: Cluster 3 (largest) may reflect a common fatty acid signature shared across most samples.

Cluster 2 and 5 (smaller) might correspond to niche or regional olive oil profiles.

Clusters can be further analyzed in future work for classification, product authentication, or geographical indication studies.

AI-Use Declaration

This assessment is **AI-supported** under the University of Exeter's guidelines for the module **MTHM053 – Applications of Data Science and Statistics**.

I acknowledge the following uses of GenAI tools in this assessment:

☒ I have used GenAI tools to suggest section headings for my report.
☒ I have used GenAI tools to help me to correct my grammar or spelling.
☒ I have used GenAI tools to suggest topics to discuss in my literature review.

I declare that I have referenced the use of GenAI outputs within my assessment, in line with the University's referencing guidelines.