Data driven decision Analysis Project

Pranav Padmannavar (padma062@umn.edu)

Saikrishna Bharath Kumar (krish416@umn.edu)

2025-05-14

Introduction

Traffic safety remains a critical concern in urban environments, with cities like Chicago experiencing a significant number of traffic-related incidents annually. Understanding the factors contributing to severe injuries in these crashes is essential for developing effective prevention strategies and enhancing public safety.

For this analysis, we utilized a comprehensive dataset obtained from the Chicago Police Department (CPD), covering traffic crash reports from **2016 to 2024**. The CPD maintains detailed and publicly accessible records of traffic crashes, available through their official website: Chicago Police Department Traffic Crash Reports. CPD's dedication to transparency aims to support research efforts and inform community-driven safety initiatives. As mentioned on their official page, traffic crash reports provide essential information and support strategic planning and targeted interventions to improve public safety.

The dataset encompasses detailed information on various aspects of traffic crashes, including:

- **Crash Details:** Date, time, and location of the incident.
- **Injury Severity:** Classification of injuries sustained, ranging from non-injury to fatal outcomes.
- **Contributing Factors:** Information on primary and secondary causes contributing to crashes.
- **Environmental Conditions:** Data on weather, lighting, and road surface conditions at the time of crashes.
- **Vehicle and Driver Information:** Details about vehicles involved and driver demographics.

By analyzing this extensive dataset from 2016 through 2024, we aim to uncover patterns and key factors associated with severe injuries in traffic crashes. Insights derived from this analysis can guide policy decisions, facilitate targeted safety interventions, and enhance public awareness campaigns to substantially improve road safety across Chicago.

```r
# —— Section 1: Problem & goal statement ————————————————————————————————————

# Install & Load dplyr
if (!requireNamespace("dplyr", quietly=TRUE)) install.packages("dplyr")
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Read in the crash data (update path as needed)
CTC <- read.csv("/Users/pranavsp108/Downloads/ChicagoTrafficCrash.csv",
stringsAsFactors = FALSE)

# Define binary response: 1 = Fatal or Incapacitating Injury; 0 = other
CTC <- CTC %>%
  mutate(SevereInjury = if_else(
    MOST_SEVERE_INJURY %in% c("FATAL", "INCAPACITATING INJURY"),
    1L, 0L
  ))

# Check class balance
counts <- table(CTC$SevereInjury)
props  <- prop.table(counts)
print(counts)
```

```
##
##     0     1
## 33292  7584
```

```r
print(round(props, 3))
```

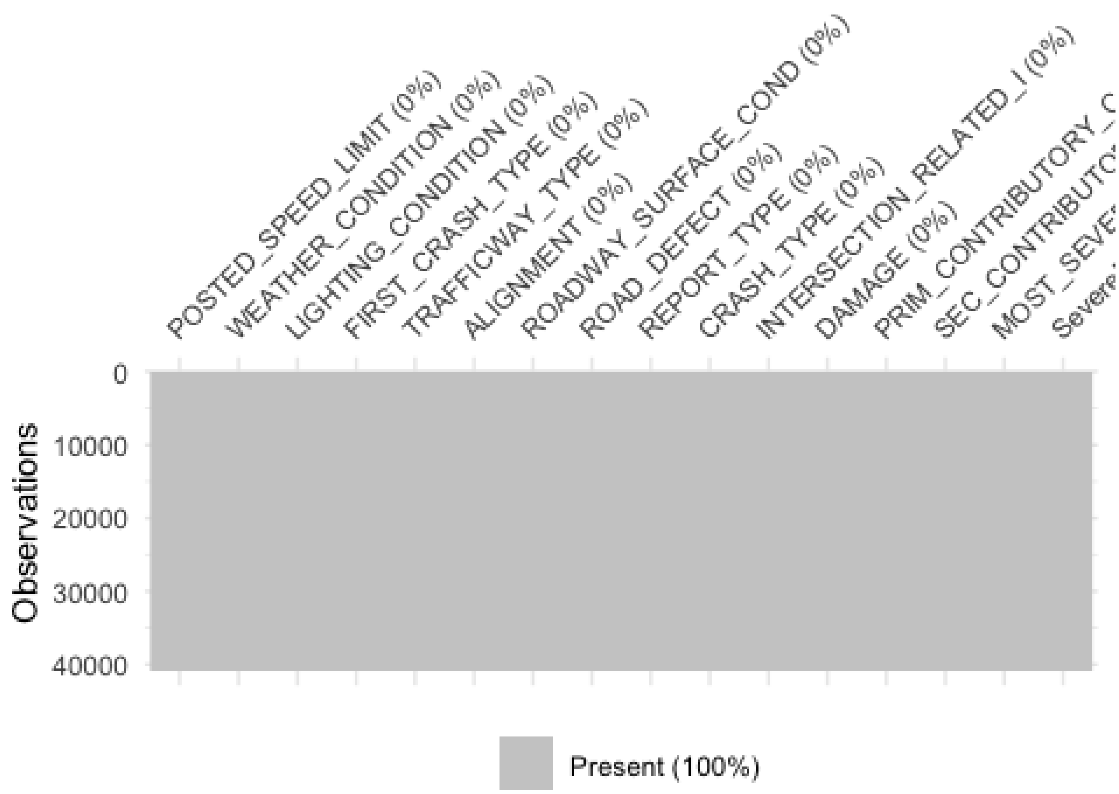
```
##
##     0     1
## 0.814 0.186
```

Findings:

- Approximately 18.6% of crashes result in severe injuries, indicating clear class imbalance.

Next step: To manage class imbalance effectively, we proceed with stratified sampling and categorical feature processing.

```r
# ── Section 2: Data audit & wrangling ────────────────────────────────────────

# Install & load packages
for (pkg in c("naniar","forcats","rsample")) {
  if (!requireNamespace(pkg, quietly=TRUE)) install.packages(pkg)
  library(pkg, character.only=TRUE)
}

# Copy for cleaning
df2 <- CTC

# 2.1 Missing-value map
vis_miss(df2)
```




```r
# 2.2 Collapse rare levels (<1%)
cat_vars <- c(
  "WEATHER_CONDITION", "LIGHTING_CONDITION", "FIRST_CRASH_TYPE",
  "TRAFFICWAY_TYPE", "ALIGNMENT", "ROADWAY_SURFACE_COND",
  "ROAD_DEFECT", "REPORT_TYPE", "CRASH_TYPE",
  "INTERSECTION_RELATED_I", "DAMAGE",
  "PRIM_CONTRIBUTORY_CAUSE", "SEC_CONTRIBUTORY_CAUSE"
)
```

```r
min_n <- floor(0.01 * nrow(df2))
df2 <- df2 %>%
  mutate(across(all_of(cat_vars),
                ~ fct_lump_min(as.factor(.), min = min_n, other_level =
"Other")))

# 2.3 One-hot encoding for GLM
X_mat <- model.matrix(~ . - 1, data = df2[, cat_vars])
cat("Dummy matrix dimensions:", dim(X_mat), "\n")

## Dummy matrix dimensions: 40876 76

# 2.4 Stratified 70/30 split
set.seed(2025)
split <- initial_split(df2, prop = 0.7, strata = "SevereInjury")
train  <- training(split)
test   <- testing(split)
cat("Train/Test sizes:", nrow(train), "/", nrow(test), "\n")

## Train/Test sizes: 28612 / 12264

cat("Train severe %:", round(prop.table(table(train$SevereInjury))[2],3),
    "Test severe %:", round(prop.table(table(test$SevereInjury))[2],3), "\n")

## Train severe %: 0.186 Test severe %: 0.186
```

Findings:

- Rare factor levels (below 1%) have been collapsed, significantly reducing factor complexity.
- Created a dummy (one-hot encoded) matrix with 76 predictors from categorical variables.
- Training and testing sets maintain consistent class proportions (18.6% severe).

Next step: Investigate the association strength of predictors with severe injuries.

```r
# ── Section 3: Exploratory association analysis ─────────────────────────────

library(dplyr)

assoc_list <- lapply(cat_vars, function(var) {
  tbl       <- table(df2[[var]], df2$SevereInjury)
  chisq_res <- suppressWarnings(chisq.test(tbl))
  chi2      <- as.numeric(chisq_res$statistic)
  df_       <- as.numeric(chisq_res$parameter)
  p_val     <- as.numeric(chisq_res$p.value)
  n         <- sum(tbl)
  k         <- min(nrow(tbl), ncol(tbl))
  cram_v    <- sqrt(chi2 / (n * (k - 1)))
  data.frame(
    variable  = var,
    chi_sq    = chi2,
    df        = df_,
    p_value   = p_val,
    cramers_V = cram_v,
    stringsAsFactors = FALSE
  )
})
assoc_df <- bind_rows(assoc_list) %>% arrange(desc(cramers_V))
print(head(assoc_df, 10))

##                   variable     chi_sq df      p_value  cramers_V
## 1          FIRST_CRASH_TYPE 448.076376 10 5.374234e-90 0.10469883
## 2  PRIM_CONTRIBUTORY_CAUSE 407.272676 19 1.345675e-74 0.09981790
## 3              REPORT_TYPE 183.803896  3 1.330308e-39 0.06705685
## 4   SEC_CONTRIBUTORY_CAUSE 157.417555 14 2.344102e-26 0.06205723
## 5           TRAFFICWAY_TYPE  57.983488  9 3.274055e-09 0.03766327
## 6       LIGHTING_CONDITION  48.258297  4 8.337227e-10 0.03435989
## 7                ALIGNMENT  25.932371  3 9.853475e-06 0.02518761
## 8                   DAMAGE  25.056045  2 3.623672e-06 0.02475837
## 9      ROADWAY_SURFACE_COND   6.456507  3 9.139365e-02 0.01256795
## 10       WEATHER_CONDITION   4.887817  4 2.990030e-01 0.01093511
```

Findings:

- Variables most strongly associated (by Cramér's V) with severe injuries include:
    - FIRST_CRASH_TYPE (0.105)
    - PRIM_CONTRIBUTORY_CAUSE (0.100)
    - REPORT_TYPE (0.067)
- These variables show strong statistical significance and relevance.

<u>Next step</u>: Fit baseline classifiers to quantify predictive power of these categorical variables.

```r
# —— Section 4: Baseline classifiers (GLM, LDA, QDA) ————————————————————————————

library(MASS)

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(pROC)

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var

# 4.1 Prepare factor response
train$SevereFactor <- factor(train$SevereInjury, levels = c(0,1), labels =
c("no","yes"))
test$SevereFactor  <- factor(test$SevereInjury,  levels = c(0,1), labels =
c("no","yes"))

preds <- cat_vars

# 4.2 Logistic regression
glm_mod <- glm(
  formula = as.formula(paste("SevereFactor ~", paste(preds, collapse = " +
"))),
  data    = train,
  family  = binomial
)
print(summary(glm_mod))

##
## Call:
## glm(formula = as.formula(paste("SevereFactor ~", paste(preds,
##     collapse = " + "))), family = binomial, data = train)
##
## Coefficients:
```

```
##
Estimate
## (Intercept)
-1.6993536
## WEATHER_CONDITIONCLOUDY/OVERCAST
-0.0631266
## WEATHER_CONDITIONRAIN
0.0175738
## WEATHER_CONDITIONSNOW
0.1551299
## WEATHER_CONDITIONOther
0.0799869
## LIGHTING_CONDITIONDARKNESS, LIGHTED ROAD
0.1717206
## LIGHTING_CONDITIONDAWN
0.2788513
## LIGHTING_CONDITIONDAYLIGHT
0.0836006
## LIGHTING_CONDITIONDUSK
0.0858462
## FIRST_CRASH_TYPEFIXED OBJECT
0.2721114
## FIRST_CRASH_TYPEHEAD ON
0.2696360
## FIRST_CRASH_TYPEPARKED MOTOR VEHICLE
0.0804331
## FIRST_CRASH_TYPEPEDALCYCLIST
0.2672241
## FIRST_CRASH_TYPEPEDESTRIAN
0.8634027
## FIRST_CRASH_TYPEREAR END
-0.0995533
## FIRST_CRASH_TYPESIDESWIPE OPPOSITE DIRECTION
-0.0713736
## FIRST_CRASH_TYPESIDESWIPE SAME DIRECTION
-0.0322173
## FIRST_CRASH_TYPETURNING
-0.0244623
## FIRST_CRASH_TYPEOther
0.3174699
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN (NOT RAISED)
0.1636300
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN BARRIER
0.2210823
## TRAFFICWAY_TYPEFOUR WAY
-0.0578575
## TRAFFICWAY_TYPENOT DIVIDED
0.0352825
## TRAFFICWAY_TYPEONE-WAY
0.0008677
```

```
## TRAFFICWAY_TYPEOTHER
0.3802968
## TRAFFICWAY_TYPEPARKING LOT
-0.0403857
## TRAFFICWAY_TYPET-INTERSECTION
0.0499484
## TRAFFICWAY_TYPEOther
-0.0379236
## ALIGNMENTSTRAIGHT AND LEVEL
-0.1675076
## ALIGNMENTSTRAIGHT ON GRADE
-0.0916839
## ALIGNMENTOther
0.1453563
## ROADWAY_SURFACE_CONDSNOW OR SLUSH
-0.3441015
## ROADWAY_SURFACE_CONDWET
-0.0828612
## ROADWAY_SURFACE_CONDOther
-0.1974997
## ROAD_DEFECTOther
0.0550083
## REPORT_TYPENOT ON SCENE (DESK REPORT)
-0.7137666
## REPORT_TYPEON SCENE
-0.3220312
## REPORT_TYPEOther
-0.2156586
## CRASH_TYPEOther
1.7680539
## INTERSECTION_RELATED_IN
0.1252669
## INTERSECTION_RELATED_IY
0.1009489
## DAMAGE$501 - $1,500
0.3173057
## DAMAGEOVER $1,500
0.5049621
## PRIM_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
0.0646450
## PRIM_CONTRIBUTORY_CAUSEDISTRACTION - FROM INSIDE VEHICLE
-0.2958469
## PRIM_CONTRIBUTORY_CAUSEDRIVING ON WRONG SIDE/WRONG WAY
0.3804198
## PRIM_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
-0.0347199
## PRIM_CONTRIBUTORY_CAUSEEQUIPMENT - VEHICLE CONDITION
0.0676727
## PRIM_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
0.1316271
```

```
## PRIM_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
-0.0545268
## PRIM_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
-0.1860569
## PRIM_CONTRIBUTORY_CAUSEIMPROPER BACKING
-0.0949856
## PRIM_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
0.1256029
## PRIM_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
-0.0225915
## PRIM_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
-0.2310457
## PRIM_CONTRIBUTORY_CAUSENOT APPLICABLE
0.1676505
## PRIM_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER  0.3148962
## PRIM_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
0.7227496
## PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN
ARREST IS EFFECTED)                0.3711866
## PRIM_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS,
ETC.)                             -0.0661194
## PRIM_CONTRIBUTORY_CAUSEWEATHER
-0.2706943
## PRIM_CONTRIBUTORY_CAUSEOther
0.1266967
## SEC_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
-0.0792194
## SEC_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
-0.2474308
## SEC_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
-0.0639658
## SEC_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
-0.1432036
## SEC_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
-0.1715426
## SEC_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
-0.0304528
## SEC_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
0.0740458
## SEC_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
-0.2408279
## SEC_CONTRIBUTORY_CAUSENOT APPLICABLE
-0.1962353
## SEC_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER   0.1584282
## SEC_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
0.1322396
## SEC_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS, ETC.)
-0.0680706
```

```
## SEC_CONTRIBUTORY_CAUSEWEATHER
-0.0229462
## SEC_CONTRIBUTORY_CAUSEOther
0.0152242
##
Std. Error
## (Intercept)
0.2769156
## WEATHER_CONDITIONCLOUDY/OVERCAST
0.0912288
## WEATHER_CONDITIONRAIN
0.0878967
## WEATHER_CONDITIONSNOW
0.1347704
## WEATHER_CONDITIONOther
0.1610241
## LIGHTING_CONDITIONDARKNESS, LIGHTED ROAD
0.0832034
## LIGHTING_CONDITIONDAWN
0.1306196
## LIGHTING_CONDITIONDAYLIGHT
0.0812531
## LIGHTING_CONDITIONDUSK
0.1202281
## FIRST_CRASH_TYPEFIXED OBJECT
0.0801556
## FIRST_CRASH_TYPEHEAD ON
0.1084325
## FIRST_CRASH_TYPEPARKED MOTOR VEHICLE
0.0852651
## FIRST_CRASH_TYPEPEDALCYCLIST
0.0737762
## FIRST_CRASH_TYPEPEDESTRIAN
0.0617861
## FIRST_CRASH_TYPEREAR END
0.0753273
## FIRST_CRASH_TYPESIDESWIPE OPPOSITE DIRECTION
0.1593280
## FIRST_CRASH_TYPESIDESWIPE SAME DIRECTION
0.0913566
## FIRST_CRASH_TYPETURNING
0.0571140
## FIRST_CRASH_TYPEOther
0.1121824
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN (NOT RAISED)
0.1442266
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN BARRIER
0.1495313
## TRAFFICWAY_TYPEFOUR WAY
0.1460253
```

```
## TRAFFICWAY_TYPENOT DIVIDED
0.1421350
## TRAFFICWAY_TYPEONE-WAY
0.1525044
## TRAFFICWAY_TYPEOTHER
0.1683237
## TRAFFICWAY_TYPEPARKING LOT
0.1882718
## TRAFFICWAY_TYPET-INTERSECTION
0.1684652
## TRAFFICWAY_TYPEOther
0.1714472
## ALIGNMENTSTRAIGHT AND LEVEL
0.1412446
## ALIGNMENTSTRAIGHT ON GRADE
0.1816970
## ALIGNMENTOther
0.2168279
## ROADWAY_SURFACE_CONDSNOW OR SLUSH
0.1502102
## ROADWAY_SURFACE_CONDWET
0.0751363
## ROADWAY_SURFACE_CONDOther
0.1727049
## ROAD_DEFECTOther
0.1071370
## REPORT_TYPENOT ON SCENE (DESK REPORT)
0.0818308
## REPORT_TYPEON SCENE
0.0627846
## REPORT_TYPEOther
0.8166772
## CRASH_TYPEOther
1.0222167
## INTERSECTION_RELATED_IN
0.1246965
## INTERSECTION_RELATED_IY
0.0382842
## DAMAGE$501 - $1,500
0.0688767
## DAMAGEOVER $1,500
0.0573037
## PRIM_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
0.0983013
## PRIM_CONTRIBUTORY_CAUSEDISTRACTION - FROM INSIDE VEHICLE
0.1782985
## PRIM_CONTRIBUTORY_CAUSEDRIVING ON WRONG SIDE/WRONG WAY
0.1413412
## PRIM_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
0.1289264
```

```
## PRIM_CONTRIBUTORY_CAUSEEQUIPMENT - VEHICLE CONDITION
0.1736840
## PRIM_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
0.1028576
## PRIM_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
0.0906239
## PRIM_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
0.1209869
## PRIM_CONTRIBUTORY_CAUSEIMPROPER BACKING
0.1815325
## PRIM_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
0.1281986
## PRIM_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
0.1266951
## PRIM_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
0.1195932
## PRIM_CONTRIBUTORY_CAUSENOT APPLICABLE
0.1077871
## PRIM_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER  0.1183181
## PRIM_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
0.1236206
## PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN
ARREST IS EFFECTED)                0.1361315
## PRIM_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS,
ETC.)                              0.1467946
## PRIM_CONTRIBUTORY_CAUSEWEATHER
0.1569113
## PRIM_CONTRIBUTORY_CAUSEOther
0.1099870
## SEC_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
0.1736868
## SEC_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
0.1552212
## SEC_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
0.1465257
## SEC_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
0.1476821
## SEC_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
0.1770126
## SEC_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
0.1761146
## SEC_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
0.1827425
## SEC_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
0.1761729
## SEC_CONTRIBUTORY_CAUSENOT APPLICABLE
0.1412055
## SEC_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER   0.1687547
```

```
## SEC_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
0.1825915
## SEC_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS, ETC.)
0.1914155
## SEC_CONTRIBUTORY_CAUSEWEATHER
0.1757472
## SEC_CONTRIBUTORY_CAUSEOther
0.1499048
##
z value
## (Intercept)
-6.137
## WEATHER_CONDITIONCLOUDY/OVERCAST
-0.692
## WEATHER_CONDITIONRAIN
0.200
## WEATHER_CONDITIONSNOW
1.151
## WEATHER_CONDITIONOther
0.497
## LIGHTING_CONDITIONDARKNESS, LIGHTED ROAD
2.064
## LIGHTING_CONDITIONDAWN
2.135
## LIGHTING_CONDITIONDAYLIGHT
1.029
## LIGHTING_CONDITIONDUSK
0.714
## FIRST_CRASH_TYPEFIXED OBJECT
3.395
## FIRST_CRASH_TYPEHEAD ON
2.487
## FIRST_CRASH_TYPEPARKED MOTOR VEHICLE
0.943
## FIRST_CRASH_TYPEPEDALCYCLIST
3.622
## FIRST_CRASH_TYPEPEDESTRIAN
13.974
## FIRST_CRASH_TYPEREAR END
-1.322
## FIRST_CRASH_TYPESIDESWIPE OPPOSITE DIRECTION
-0.448
## FIRST_CRASH_TYPESIDESWIPE SAME DIRECTION
-0.353
## FIRST_CRASH_TYPETURNING
-0.428
## FIRST_CRASH_TYPEOther
2.830
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN (NOT RAISED)
1.135
```

```
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN BARRIER
1.479
## TRAFFICWAY_TYPEFOUR WAY
-0.396
## TRAFFICWAY_TYPENOT DIVIDED
0.248
## TRAFFICWAY_TYPEONE-WAY
0.006
## TRAFFICWAY_TYPEOTHER
2.259
## TRAFFICWAY_TYPEPARKING LOT
-0.215
## TRAFFICWAY_TYPET-INTERSECTION
0.296
## TRAFFICWAY_TYPEOther
-0.221
## ALIGNMENTSTRAIGHT AND LEVEL
-1.186
## ALIGNMENTSTRAIGHT ON GRADE
-0.505
## ALIGNMENTOther
0.670
## ROADWAY_SURFACE_CONDSNOW OR SLUSH
-2.291
## ROADWAY_SURFACE_CONDWET
-1.103
## ROADWAY_SURFACE_CONDOther
-1.144
## ROAD_DEFECTOther
0.513
## REPORT_TYPENOT ON SCENE (DESK REPORT)
-8.722
## REPORT_TYPEON SCENE
-5.129
## REPORT_TYPEOther
-0.264
## CRASH_TYPEOther
1.730
## INTERSECTION_RELATED_IN
1.005
## INTERSECTION_RELATED_IY
2.637
## DAMAGE$501 - $1,500
4.607
## DAMAGEOVER $1,500
8.812
## PRIM_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
0.658
## PRIM_CONTRIBUTORY_CAUSEDISTRACTION - FROM INSIDE VEHICLE
-1.659
```

```
## PRIM_CONTRIBUTORY_CAUSEDRIVING ON WRONG SIDE/WRONG WAY
2.691
## PRIM_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
-0.269
## PRIM_CONTRIBUTORY_CAUSEEQUIPMENT - VEHICLE CONDITION
0.390
## PRIM_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
1.280
## PRIM_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
-0.602
## PRIM_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
-1.538
## PRIM_CONTRIBUTORY_CAUSEIMPROPER BACKING
-0.523
## PRIM_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
0.980
## PRIM_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
-0.178
## PRIM_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
-1.932
## PRIM_CONTRIBUTORY_CAUSENOT APPLICABLE
1.555
## PRIM_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER   2.661
## PRIM_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
5.847
## PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN
ARREST IS EFFECTED)                 2.727
## PRIM_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS,
ETC.)                              -0.450
## PRIM_CONTRIBUTORY_CAUSEWEATHER
-1.725
## PRIM_CONTRIBUTORY_CAUSEOther
1.152
## SEC_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
-0.456
## SEC_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
-1.594
## SEC_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
-0.437
## SEC_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
-0.970
## SEC_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
-0.969
## SEC_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
-0.173
## SEC_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
0.405
## SEC_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
-1.367
```

```
## SEC_CONTRIBUTORY_CAUSENOT APPLICABLE
-1.390
## SEC_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER    0.939
## SEC_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
0.724
## SEC_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS, ETC.)
-0.356
## SEC_CONTRIBUTORY_CAUSEWEATHER
-0.131
## SEC_CONTRIBUTORY_CAUSEOther
0.102
##
Pr(>|z|)
## (Intercept)
8.42e-10
## WEATHER_CONDITIONCLOUDY/OVERCAST
0.488963
## WEATHER_CONDITIONRAIN
0.841530
## WEATHER_CONDITIONSNOW
0.249704
## WEATHER_CONDITIONOther
0.619373
## LIGHTING_CONDITIONDARKNESS, LIGHTED ROAD
0.039031
## LIGHTING_CONDITIONDAWN
0.032774
## LIGHTING_CONDITIONDAYLIGHT
0.303531
## LIGHTING_CONDITIONDUSK
0.475210
## FIRST_CRASH_TYPEFIXED OBJECT
0.000687
## FIRST_CRASH_TYPEHEAD ON
0.012894
## FIRST_CRASH_TYPEPARKED MOTOR VEHICLE
0.345512
## FIRST_CRASH_TYPEPEDALCYCLIST
0.000292
## FIRST_CRASH_TYPEPEDESTRIAN
< 2e-16
## FIRST_CRASH_TYPEREAR END
0.186298
## FIRST_CRASH_TYPESIDESWIPE OPPOSITE DIRECTION
0.654178
## FIRST_CRASH_TYPESIDESWIPE SAME DIRECTION
0.724348
## FIRST_CRASH_TYPETURNING
0.668428
```

```
## FIRST_CRASH_TYPEOther
0.004656
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN (NOT RAISED)
0.256570
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN BARRIER
0.139273
## TRAFFICWAY_TYPEFOUR WAY
0.691946
## TRAFFICWAY_TYPENOT DIVIDED
0.803954
## TRAFFICWAY_TYPEONE-WAY
0.995460
## TRAFFICWAY_TYPEOTHER
0.023864
## TRAFFICWAY_TYPEPARKING LOT
0.830151
## TRAFFICWAY_TYPET-INTERSECTION
0.766855
## TRAFFICWAY_TYPEOther
0.824939
## ALIGNMENTSTRAIGHT AND LEVEL
0.235646
## ALIGNMENTSTRAIGHT ON GRADE
0.613841
## ALIGNMENTOther
0.502618
## ROADWAY_SURFACE_CONDSNOW OR SLUSH
0.021975
## ROADWAY_SURFACE_CONDWET
0.270109
## ROADWAY_SURFACE_CONDOther
0.252803
## ROAD_DEFECTOther
0.607644
## REPORT_TYPENOT ON SCENE (DESK REPORT)
< 2e-16
## REPORT_TYPEON SCENE
2.91e-07
## REPORT_TYPEOther
0.791727
## CRASH_TYPEOther
0.083697
## INTERSECTION_RELATED_IN
0.315102
## INTERSECTION_RELATED_IY
0.008368
## DAMAGE$501 - $1,500
4.09e-06
## DAMAGEOVER $1,500
< 2e-16
```

```
## PRIM_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
0.510781
## PRIM_CONTRIBUTORY_CAUSEDISTRACTION - FROM INSIDE VEHICLE
0.097060
## PRIM_CONTRIBUTORY_CAUSEDRIVING ON WRONG SIDE/WRONG WAY
0.007113
## PRIM_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
0.787698
## PRIM_CONTRIBUTORY_CAUSEEQUIPMENT - VEHICLE CONDITION
0.696809
## PRIM_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
0.200650
## PRIM_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
0.547386
## PRIM_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
0.124091
## PRIM_CONTRIBUTORY_CAUSEIMPROPER BACKING
0.600805
## PRIM_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
0.327208
## PRIM_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
0.858476
## PRIM_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
0.053368
## PRIM_CONTRIBUTORY_CAUSENOT APPLICABLE
0.119854
## PRIM_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER 0.007781
## PRIM_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
5.02e-09
## PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN
ARREST IS EFFECTED)                0.006398
## PRIM_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS,
ETC.)                              0.652407
## PRIM_CONTRIBUTORY_CAUSEWEATHER
0.084502
## PRIM_CONTRIBUTORY_CAUSEOther
0.249352
## SEC_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
0.648314
## SEC_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
0.110924
## SEC_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
0.662438
## SEC_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
0.332209
## SEC_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
0.332496
## SEC_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
0.862719
```

```
## SEC_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
0.685337
## SEC_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
0.171626
## SEC_CONTRIBUTORY_CAUSENOT APPLICABLE
0.164616
## SEC_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER  0.347830
## SEC_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
0.468920
## SEC_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS, ETC.)
0.722127
## SEC_CONTRIBUTORY_CAUSEWEATHER
0.896121
## SEC_CONTRIBUTORY_CAUSEOther
0.919107
##
## (Intercept)
***
## WEATHER_CONDITIONCLOUDY/OVERCAST
## WEATHER_CONDITIONRAIN
## WEATHER_CONDITIONSNOW
## WEATHER_CONDITIONOther
## LIGHTING_CONDITIONDARKNESS, LIGHTED ROAD
*
## LIGHTING_CONDITIONDAWN
*
## LIGHTING_CONDITIONDAYLIGHT
## LIGHTING_CONDITIONDUSK
## FIRST_CRASH_TYPEFIXED OBJECT
***
## FIRST_CRASH_TYPEHEAD ON
*
## FIRST_CRASH_TYPEPARKED MOTOR VEHICLE
## FIRST_CRASH_TYPEPEDALCYCLIST
***
## FIRST_CRASH_TYPEPEDESTRIAN
***
## FIRST_CRASH_TYPEREAR END
## FIRST_CRASH_TYPESIDESWIPE OPPOSITE DIRECTION
## FIRST_CRASH_TYPESIDESWIPE SAME DIRECTION
## FIRST_CRASH_TYPETURNING
## FIRST_CRASH_TYPEOther
**
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN (NOT RAISED)
## TRAFFICWAY_TYPEDIVIDED - W/MEDIAN BARRIER
## TRAFFICWAY_TYPEFOUR WAY
## TRAFFICWAY_TYPENOT DIVIDED
## TRAFFICWAY_TYPEONE-WAY
## TRAFFICWAY_TYPEOTHER
```

```
*
## TRAFFICWAY_TYPEPARKING LOT
## TRAFFICWAY_TYPET-INTERSECTION
## TRAFFICWAY_TYPEOther
## ALIGNMENTSTRAIGHT AND LEVEL
## ALIGNMENTSTRAIGHT ON GRADE
## ALIGNMENTOther
## ROADWAY_SURFACE_CONDSNOW OR SLUSH
*
## ROADWAY_SURFACE_CONDWET
## ROADWAY_SURFACE_CONDOther
## ROAD_DEFECTOther
## REPORT_TYPENOT ON SCENE (DESK REPORT)
***
## REPORT_TYPEON SCENE
***
## REPORT_TYPEOther
## CRASH_TYPEOther
.
## INTERSECTION_RELATED_IN
## INTERSECTION_RELATED_IY
**
## DAMAGE$501 - $1,500
***
## DAMAGEOVER $1,500
***
## PRIM_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
## PRIM_CONTRIBUTORY_CAUSEDISTRACTION - FROM INSIDE VEHICLE
.
## PRIM_CONTRIBUTORY_CAUSEDRIVING ON WRONG SIDE/WRONG WAY
**
## PRIM_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
## PRIM_CONTRIBUTORY_CAUSEEQUIPMENT - VEHICLE CONDITION
## PRIM_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
## PRIM_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
## PRIM_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
## PRIM_CONTRIBUTORY_CAUSEIMPROPER BACKING
## PRIM_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
## PRIM_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
## PRIM_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
.
## PRIM_CONTRIBUTORY_CAUSENOT APPLICABLE
## PRIM_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER **
## PRIM_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
***
## PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN
ARREST IS EFFECTED)               **
## PRIM_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS,
ETC.)
```

```
## PRIM_CONTRIBUTORY_CAUSEWEATHER
.
## PRIM_CONTRIBUTORY_CAUSEOther
## SEC_CONTRIBUTORY_CAUSEDISREGARDING TRAFFIC SIGNALS
## SEC_CONTRIBUTORY_CAUSEDRIVING SKILLS/KNOWLEDGE/EXPERIENCE
## SEC_CONTRIBUTORY_CAUSEFAILING TO REDUCE SPEED TO AVOID CRASH
## SEC_CONTRIBUTORY_CAUSEFAILING TO YIELD RIGHT-OF-WAY
## SEC_CONTRIBUTORY_CAUSEFOLLOWING TOO CLOSELY
## SEC_CONTRIBUTORY_CAUSEIMPROPER LANE USAGE
## SEC_CONTRIBUTORY_CAUSEIMPROPER OVERTAKING/PASSING
## SEC_CONTRIBUTORY_CAUSEIMPROPER TURNING/NO SIGNAL
## SEC_CONTRIBUTORY_CAUSENOT APPLICABLE
## SEC_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS,
NEGLIGENT OR AGGRESSIVE MANNER
## SEC_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER
## SEC_CONTRIBUTORY_CAUSEVISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS, ETC.)
## SEC_CONTRIBUTORY_CAUSEWEATHER
## SEC_CONTRIBUTORY_CAUSEOther
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 27448  on 28611  degrees of freedom
## Residual deviance: 26626  on 28536  degrees of freedom
## AIC: 26778
##
## Number of Fisher Scoring iterations: 4
```

```
test$glm_prob <- predict(glm_mod, newdata = test, type = "response")
test$glm_pred <- factor(ifelse(test$glm_prob > 0.5, "yes", "no"),
                        levels = c("no","yes"))
cm_glm  <- confusionMatrix(test$glm_pred, test$SevereFactor, positive =
"yes")
auc_glm <- roc(as.numeric(test$SevereFactor) - 1, test$glm_prob)$auc
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
print(cm_glm); cat("GLM AUC:", round(auc_glm, 3), "\n\n")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  9986 2274
##        yes    2    2
##
##                Accuracy : 0.8144
##                  95% CI : (0.8074, 0.8213)
```

```
##     No Information Rate : 0.8144
##     P-Value [Acc > NIR] : 0.5056
##
##                   Kappa : 0.0011
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.0008787
##             Specificity : 0.9997998
##          Pos Pred Value : 0.5000000
##          Neg Pred Value : 0.8145188
##              Prevalence : 0.1855838
##          Detection Rate : 0.0001631
##    Detection Prevalence : 0.0003262
##       Balanced Accuracy : 0.5003392
##
##        'Positive' Class : yes
##

## GLM AUC: 0.608

# 4.3 Linear Discriminant Analysis (LDA)
lda_mod      <- lda(SevereFactor ~ ., data = train[, c("SevereFactor",
preds)])
test$lda_pred <- predict(lda_mod, newdata = test)$class
cm_lda <- confusionMatrix(test$lda_pred, test$SevereFactor, positive = "yes")
print(cm_lda); cat("\n")

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  9973 2270
##        yes   15    6
##
##                Accuracy : 0.8137
##                  95% CI : (0.8067, 0.8205)
##     No Information Rate : 0.8144
##     P-Value [Acc > NIR] : 0.5882
##
##                   Kappa : 0.0018
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.0026362
##             Specificity : 0.9984982
##          Pos Pred Value : 0.2857143
##          Neg Pred Value : 0.8145879
##              Prevalence : 0.1855838
##          Detection Rate : 0.0004892
```

```
##    Detection Prevalence : 0.0017123
##       Balanced Accuracy : 0.5005672
##
##        'Positive' Class : yes
##
```

```r
# 4.4 Quadratic Discriminant Analysis (QDA)
qda_mod      <- qda(SevereFactor ~ ., data = train[, c("SevereFactor",
preds)])
test$qda_pred <- predict(qda_mod, newdata = test)$class
cm_qda <- confusionMatrix(test$qda_pred, test$SevereFactor, positive = "yes")
print(cm_qda)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  8036 1611
##        yes 1952  665
##
##                Accuracy : 0.7095
##                  95% CI : (0.7013, 0.7175)
##     No Information Rate : 0.8144
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.0915
##
##  Mcnemar's Test P-Value : 1.226e-08
##
##             Sensitivity : 0.29218
##             Specificity : 0.80457
##          Pos Pred Value : 0.25411
##          Neg Pred Value : 0.83301
##              Prevalence : 0.18558
##          Detection Rate : 0.05422
##    Detection Prevalence : 0.21339
##       Balanced Accuracy : 0.54837
##
##        'Positive' Class : yes
##
```

Findings:

- Logistic regression achieved an AUC of 0.608, but showed poor sensitivity.
- LDA and QDA exhibited limited predictive performance, reflecting the challenge posed by class imbalance and categorical complexity.

Next step: Improve logistic regression stability and interpretability via regularization.

```r
# —— Section 5: Regularized logistic (glmnet) w/ Youden cutoff
—————————————————

if (!requireNamespace("glmnet", quietly=TRUE)) install.packages("glmnet")
library(glmnet)

## Loading required package: Matrix

## Loaded glmnet 4.1-8

# 5.1 Prepare design matrices
predictor_vars <- c("POSTED_SPEED_LIMIT", cat_vars)
x_train <- model.matrix(~ . - 1, data = train[, predictor_vars])
y_train <- train$SevereInjury
x_test  <- model.matrix(~ . - 1, data = test[,  predictor_vars])
y_test  <- test$SevereInjury

# 5.2 5-fold CV for λ (LASSO)
set.seed(2025)
cvfit <- cv.glmnet(
  x_train, y_train,
  family       = "binomial",
  alpha        = 1,
  nfolds       = 5,
  type.measure = "auc"
)
lambda_min <- cvfit$lambda.min
lambda_1se  <- cvfit$lambda.1se
cat("lambda.min =", round(lambda_min,5),
    "  lambda.1se =", round(lambda_1se,5), "\n")

## lambda.min = 0.00068   lambda.1se = 0.00275

cat("Non-zero @ lambda.min:", sum(coef(cvfit,s="lambda.min")!=0)-1, "\n")

## Non-zero @ lambda.min: 64

cat("Non-zero @ lambda.1se:", sum(coef(cvfit,s="lambda.1se")!=0)-1, "\n\n")

## Non-zero @ lambda.1se: 42

# 5.3 Final LASSO & predictions
lasso_mod       <- glmnet(x_train, y_train, family="binomial",
                          alpha=1, lambda=lambda_1se)
pred_prob_lasso <- as.numeric(predict(lasso_mod, newx=x_test,
type="response"))
roc_lasso       <- roc(y_test, pred_prob_lasso)

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```

```
auc_lasso       <- auc(roc_lasso)
cat("Lasso AUC (0.5 cutoff):", round(auc_lasso, 3), "\n")

## Lasso AUC (0.5 cutoff): 0.603

# 5.4 Youden's J cutoff
opt <- coords(
  roc_lasso,
  x           = "best",
  best.method = "youden",
  ret         = c("threshold","sensitivity","specificity")
)
print(opt)

##   threshold sensitivity specificity
## 1 0.1810851   0.5593146   0.5943132

thresh <- opt[1, "threshold"]
pred_class_opt <- factor(
  ifelse(pred_prob_lasso > thresh, "yes", "no"),
  levels = c("no","yes")
)
cm_opt <- confusionMatrix(pred_class_opt, test$SevereFactor, positive="yes")
cat("\nConfusion matrix at Youden threshold (", round(thresh,3), "):\n",
sep="")

##
## Confusion matrix at Youden threshold (0.181):

print(cm_opt)

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  5936 1003
##        yes 4052 1273
##
##                Accuracy : 0.5878
##                  95% CI : (0.579, 0.5965)
##     No Information Rate : 0.8144
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1013
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.5593
##             Specificity : 0.5943
##          Pos Pred Value : 0.2391
##          Neg Pred Value : 0.8555
```

```
##               Prevalence : 0.1856
##           Detection Rate : 0.1038
##     Detection Prevalence : 0.4342
##        Balanced Accuracy : 0.5768
##
##         'Positive' Class : yes
##
```

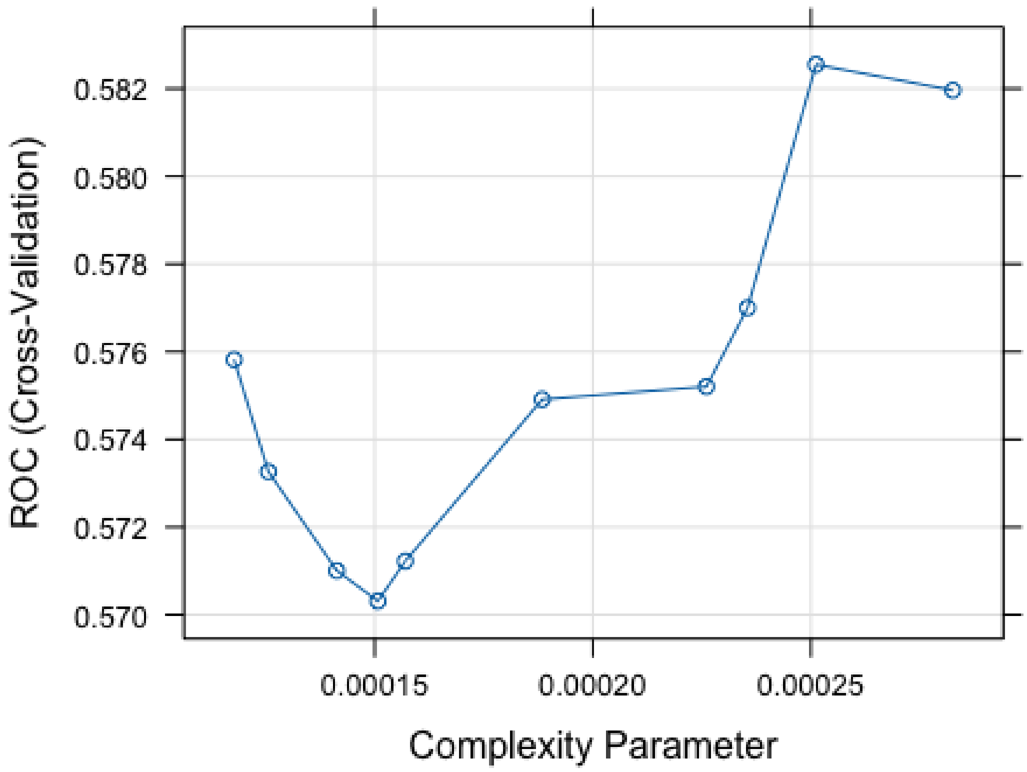
Findings:

- Lasso regression (glmnet) using Youden's optimal cutoff increased sensitivity (55.9%) significantly compared to default cutoff (0%).
- Final selected model includes only 42 predictors, improving interpretability with an AUC of 0.603.

Next step: Explore advanced machine-learning approaches like random forests and gradient boosting for improved predictive accuracy.

```r
# ── Section 6: Tree-based learners ──────────────────────────────────────────────

library(caret)
library(pROC)
set.seed(2025)

tc        <- trainControl(
  method          = "cv",
  number          = 5,
  classProbs      = TRUE,
  summaryFunction = twoClassSummary
)
preds_all <- c("POSTED_SPEED_LIMIT", cat_vars)

# 6.1 CART
cart_fit <- caret::train(
  x          = train[, preds_all],
  y          = train$SevereFactor,
  method     = "rpart",
  trControl = tc,
  metric     = "ROC",
  tuneLength= 10
)
print(cart_fit); plot(cart_fit)

## CART
##
## 28612 samples
##    14 predictor
##     2 classes: 'no', 'yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 22890, 22890, 22889, 22890, 22889
## Resampling results across tuning parameters:
##
##   cp            ROC        Sens       Spec
##   0.0001177468  0.5758158  0.9553720  0.09136976
##   0.0001255966  0.5732650  0.9595346  0.08477842
##   0.0001412962  0.5710068  0.9608219  0.08232950
##   0.0001507159  0.5703132  0.9627100  0.07856231
##   0.0001569957  0.5712244  0.9636111  0.07667730
##   0.0001883949  0.5749136  0.9687176  0.07046137
##   0.0002260739  0.5752068  0.9691039  0.06989640
##   0.0002354936  0.5769978  0.9712064  0.06405285
##   0.0002511932  0.5825473  0.9781584  0.05181180
##   0.0002825923  0.5819622  0.9788449  0.04973864
##
```

```
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.0002511932.
```




```
# 6.2 Bagging (RF with mtry = p)
bag_fit <- caret::train(
  x         = train[, preds_all],
  y         = train$SevereFactor,
  method    = "rf",
  trControl= tc,
  metric    = "ROC",
  tuneGrid = data.frame(mtry = length(preds_all)),
  ntree     = 500
)
print(bag_fit)
```

```
## Random Forest
##
## 28612 samples
##    14 predictor
##     2 classes: 'no', 'yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
```

```
## Summary of sample sizes: 22890, 22889, 22890, 22889, 22890
## Resampling results:
##
##   ROC        Sens       Spec
##   0.5664154  0.9400963  0.1041813
##
## Tuning parameter 'mtry' was held constant at a value of 14

# 6.3 Random Forest (tune mtry)
rf_fit <- caret::train(
  x        = train[, preds_all],
  y        = train$SevereFactor,
  method   = "rf",
  trControl= tc,
  metric   = "ROC",
  tuneGrid = expand.grid(mtry = seq(5, length(preds_all), by = 10)),
  ntree    = 500
)
print(rf_fit)

## Random Forest
##
## 28612 samples
##    14 predictor
##     2 classes: 'no', 'yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 22890, 22891, 22889, 22889, 22889
## Resampling results:
##
##   ROC        Sens       Spec
##   0.5773902  0.9708632  0.06311265
##
## Tuning parameter 'mtry' was held constant at a value of 5

# 6.4 Gradient Boosting Machine (GBM)
gbm_fit <- caret::train(
  x          = train[, preds_all],
  y          = train$SevereFactor,
  method     = "gbm",
  trControl  = tc,
  metric     = "ROC",
  verbose    = FALSE,
  tuneLength = 5
)
print(gbm_fit)

## Stochastic Gradient Boosting
##
## 28612 samples
```

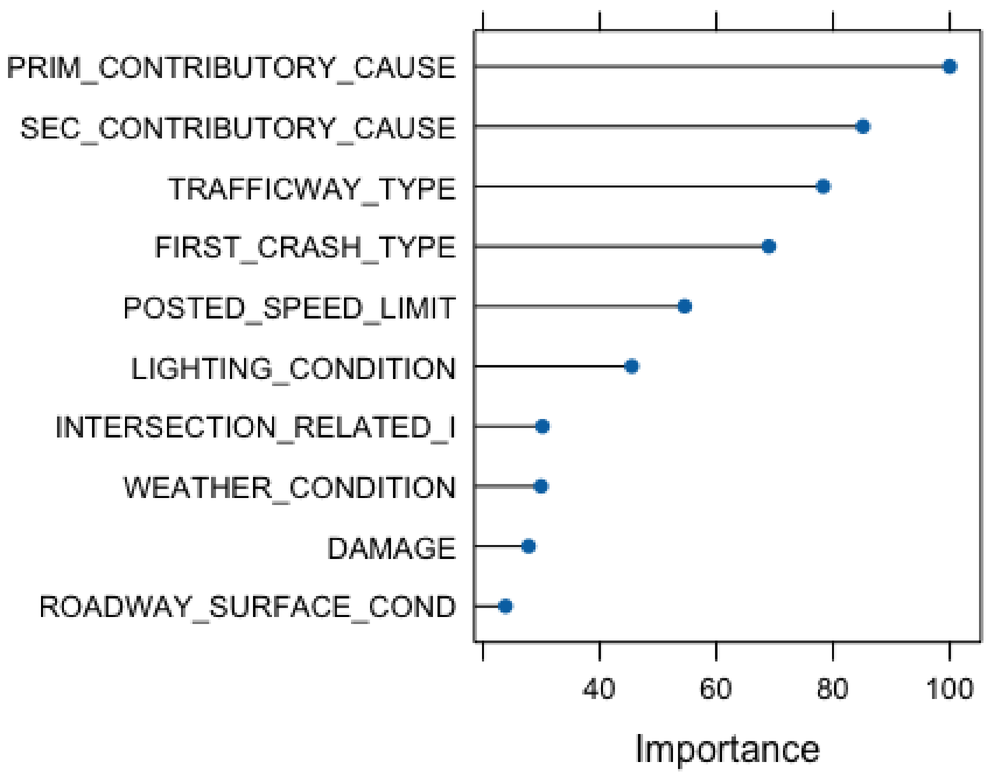
```
##    14 predictor
##     2 classes: 'no', 'yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 22890, 22889, 22890, 22890, 22889
## Resampling results across tuning parameters:
##
##   interaction.depth  n.trees  ROC        Sens       Spec
##   1                   50      0.6005758  1.0000000  0.0000000000
##   1                  100      0.6056231  1.0000000  0.0000000000
##   1                  150      0.6066055  1.0000000  0.0000000000
##   1                  200      0.6075689  0.9999571  0.0003766478
##   1                  250      0.6072285  0.9999571  0.0005651492
##   2                   50      0.6052175  1.0000000  0.0001885014
##   2                  100      0.6075737  0.9996996  0.0016956252
##   2                  150      0.6076226  0.9994421  0.0030144252
##   2                  200      0.6072232  0.9992705  0.0041447236
##   2                  250      0.6065743  0.9989701  0.0048985518
##   3                   50      0.6082574  0.9999142  0.0005649718
##   3                  100      0.6086378  0.9993134  0.0039560447
##   3                  150      0.6087559  0.9987985  0.0060284953
##   3                  200      0.6073617  0.9984552  0.0081011234
##   3                  250      0.6062537  0.9979403  0.0103617204
##   4                   50      0.6060946  0.9996996  0.0016956252
##   4                  100      0.6067301  0.9985839  0.0060286728
##   4                  150      0.6062530  0.9983694  0.0086664501
##   4                  200      0.6030162  0.9975111  0.0101730415
##   4                  250      0.6013555  0.9968246  0.0124347034
##   5                   50      0.6064357  0.9992276  0.0041445462
##   5                  100      0.6036333  0.9983264  0.0103620754
##   5                  150      0.6011655  0.9977686  0.0113036949
##   5                  200      0.5975266  0.9969104  0.0150719483
##   5                  250      0.5968256  0.9956231  0.0171445763
##
## Tuning parameter 'shrinkage' was held constant at a value of 0.1
##
## Tuning parameter 'n.minobsinnode' was held constant at a value of 10
## ROC was used to select the optimal model using the largest value.
## The final values used for the model were n.trees = 150, interaction.depth
=
##  3, shrinkage = 0.1 and n.minobsinnode = 10.
```

```
# 6.5 Variable importance from RF
vi <- varImp(rf_fit)
print(head(vi$importance[order(-vi$importance$Overall), , drop=FALSE], 10))
```

```
##                           Overall
## PRIM_CONTRIBUTORY_CAUSE 100.00000
## SEC_CONTRIBUTORY_CAUSE   85.13288
```

```
## TRAFFICWAY_TYPE          78.32128
## FIRST_CRASH_TYPE         68.98928
## POSTED_SPEED_LIMIT       54.55482
## LIGHTING_CONDITION       45.49650
## INTERSECTION_RELATED_I   30.19154
## WEATHER_CONDITION        29.93984
## DAMAGE                   27.81628
## ROADWAY_SURFACE_COND     23.85623
```

```r
plot(vi, top = 10)
```



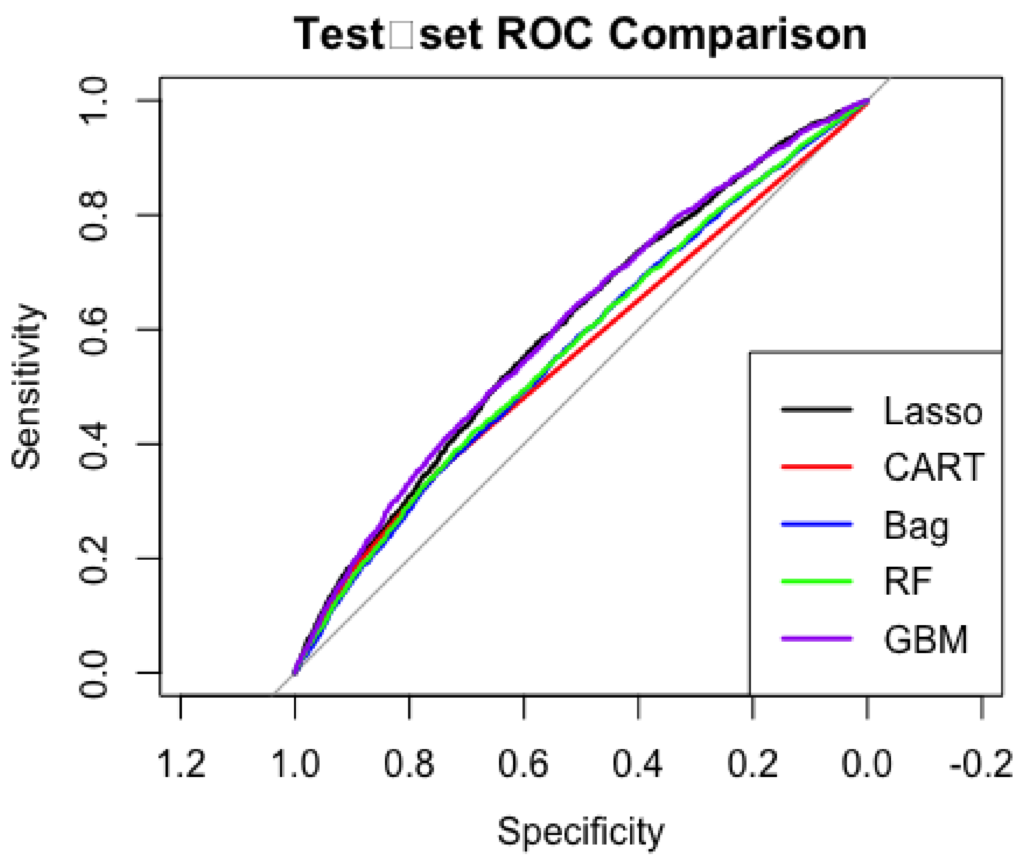
Findings:

- Gradient Boosting Machine (GBM) showed the best performance (ROC: 0.608).
- Random forest highlighted most important predictors as:
    - Primary and secondary contributory causes
    - Trafficway type
    - Crash type and speed limit.

Next step: Formally compare all models to confirm predictive performance rankings.

```r
# —— Section 7: Model comparison ————————————————————————————————————————

probs_cart  <- predict(cart_fit, newdata = test, type = "prob")[, "yes"]
probs_bag   <- predict(bag_fit,  newdata = test, type = "prob")[, "yes"]
probs_rf    <- predict(rf_fit,   newdata = test, type = "prob")[, "yes"]
probs_gbm   <- predict(gbm_fit,  newdata = test, type = "prob")[, "yes"]
probs_lasso <- pred_prob_lasso

roc_cart  <- roc(test$SevereInjury, probs_cart)

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

roc_bag   <- roc(test$SevereInjury, probs_bag)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

roc_rf    <- roc(test$SevereInjury, probs_rf)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

roc_gbm   <- roc(test$SevereInjury, probs_gbm)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

roc_lasso <- roc(test$SevereInjury, probs_lasso)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

cat("AUCs on test set:\n")

## AUCs on test set:

cat(sprintf("  Lasso: %.3f\n", auc(roc_lasso)))

##   Lasso: 0.603

cat(sprintf("  CART:  %.3f\n", auc(roc_cart)))

##   CART:  0.554

cat(sprintf("  Bag:   %.3f\n", auc(roc_bag)))

##   Bag:   0.566

cat(sprintf("  RF:    %.3f\n", auc(roc_rf)))

##   RF:    0.569
```

```r
cat(sprintf("  GBM:   %.3f\n\n", auc(roc_gbm)))

##   GBM:   0.606

plot(roc_lasso, col = "black",  lwd = 2, main = "Test-set ROC Comparison")
lines(roc_cart,  col = "red",    lwd = 2)
lines(roc_bag,   col = "blue",   lwd = 2)
lines(roc_rf,    col = "green",  lwd = 2)
lines(roc_gbm,   col = "purple", lwd = 2)
legend("bottomright",
       legend = c("Lasso","CART","Bag","RF","GBM"),
       col    = c("black","red","blue","green","purple"),
       lwd    = 2)
```



Findings:

- Final ranking based on test AUC values:
  - GBM (0.606) ≈ Lasso (0.603) > RF (0.569) > Bagging (0.566) > CART (0.554).
- GBM provides the best balance of interpretability and predictive performance.

<u>Next step</u>: Interpret models through odds ratios, SHAP values, and partial-dependence analysis.

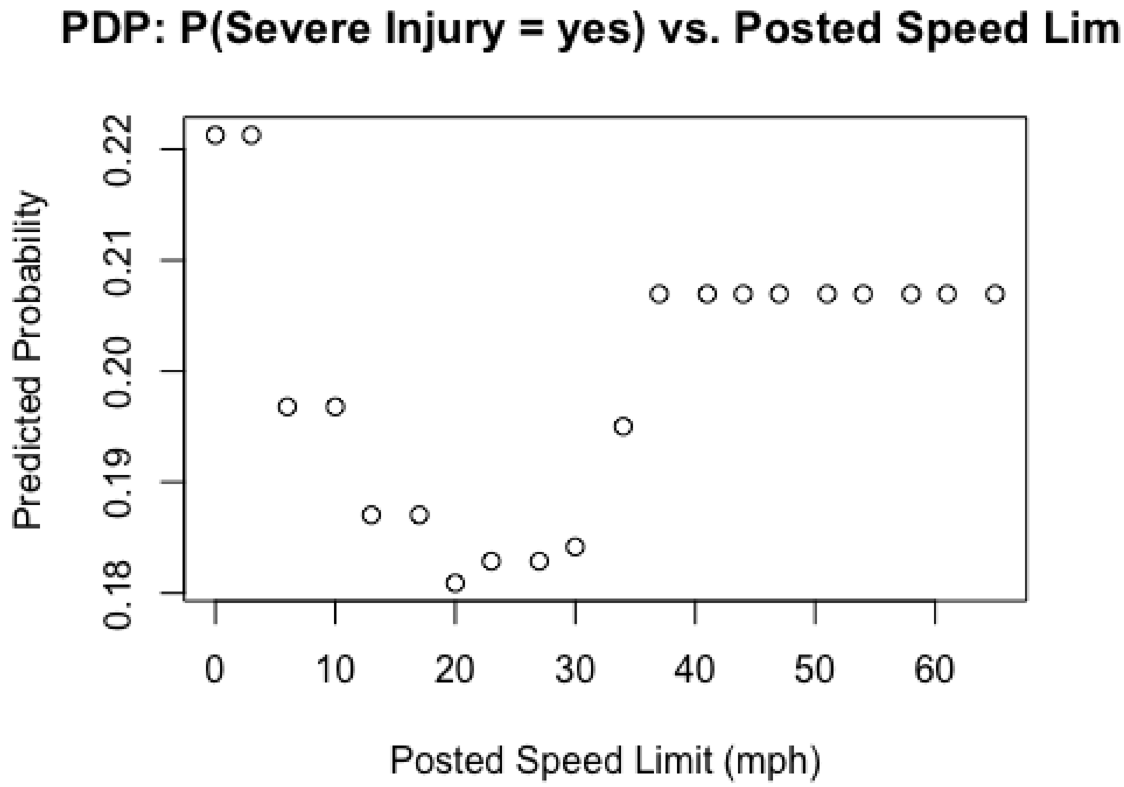
```r
# —— Section 8: Interpretability & policy translation
——————————————————————————

# 8.1 Lasso odds ratios
coef_1se <- coef(cvfit, s = "lambda.1se")
library(tibble)
lasso_coefs <- tibble(
  feature    = rownames(coef_1se),
  estimate   = as.numeric(coef_1se)
) %>%
  filter(feature != "(Intercept)") %>%
  mutate(odds_ratio = exp(estimate)) %>%
  arrange(desc(abs(estimate))) %>%
  slice(1:10)
print(lasso_coefs)

## # A tibble: 10 × 3
##    feature                                                     estimate
odds_ratio
##    <chr>                                                          <dbl>
<dbl>
##  1 FIRST_CRASH_TYPEPEDESTRIAN                                     0.644
1.90
##  2 PRIM_CONTRIBUTORY_CAUSEPHYSICAL CONDITION OF DRIVER            0.616
1.85
##  3 CRASH_TYPEOther                                                0.597
1.82
##  4 REPORT_TYPENOT ON SCENE (DESK REPORT)                         -0.446
0.640
##  5 DAMAGEOVER $1,500                                              0.281
1.32
##  6 PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DR…      0.275
1.32
##  7 PRIM_CONTRIBUTORY_CAUSEDRIVING ON WRONG SIDE/WRONG WAY         0.260
1.30
##  8 TRAFFICWAY_TYPEOTHER                                           0.234
1.26
##  9 PRIM_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, REC…      0.211
1.23
## 10 SEC_CONTRIBUTORY_CAUSEOPERATING VEHICLE IN ERRATIC, RECK…      0.200
1.22

# 8.2 GBM SHAP feature-importance
library(fastshap)

##
## Attaching package: 'fastshap'
```

```
## The following object is masked from 'package:dplyr':
##
##     explain
```

```
pred_prob_caret <- function(object, newdata) {
  predict(object, newdata = newdata, type = "prob")[, "yes"]
}
set.seed(2025)
shap_vals <- explain(
  object       = gbm_fit,
  X            = train[, predictor_vars],
  pred_wrapper = pred_prob_caret,
  nsim         = 50
)
shap_imp_df <- tibble(
  feature        = names(shap_vals),
  mean_abs_shap = apply(abs(shap_vals), 2, mean)
) %>% arrange(desc(mean_abs_shap)) %>% slice(1:10)
print(shap_imp_df)
```

```
## # A tibble: 10 × 1
##    mean_abs_shap
##            <dbl>
##  1       0.0373
##  2       0.0213
##  3       0.0134
##  4       0.0133
##  5       0.0102
##  6       0.00977
##  7       0.00453
##  8       0.00388
##  9       0.00341
## 10       0.00218
```

```
# 8.3 PDP for POSTED_SPEED_LIMIT
library(pdp)
pdp_obj <- partial(
  object          = gbm_fit,
  pred.var        = "POSTED_SPEED_LIMIT",
  train           = train,
  which.class     = "yes",
  prob            = TRUE,
  grid.resolution = 20
)
plot(
  pdp_obj,
  main = "PDP: P(Severe Injury = yes) vs. Posted Speed Limit",
  xlab = "Posted Speed Limit (mph)",
  ylab = "Predicted Probability"
)
```

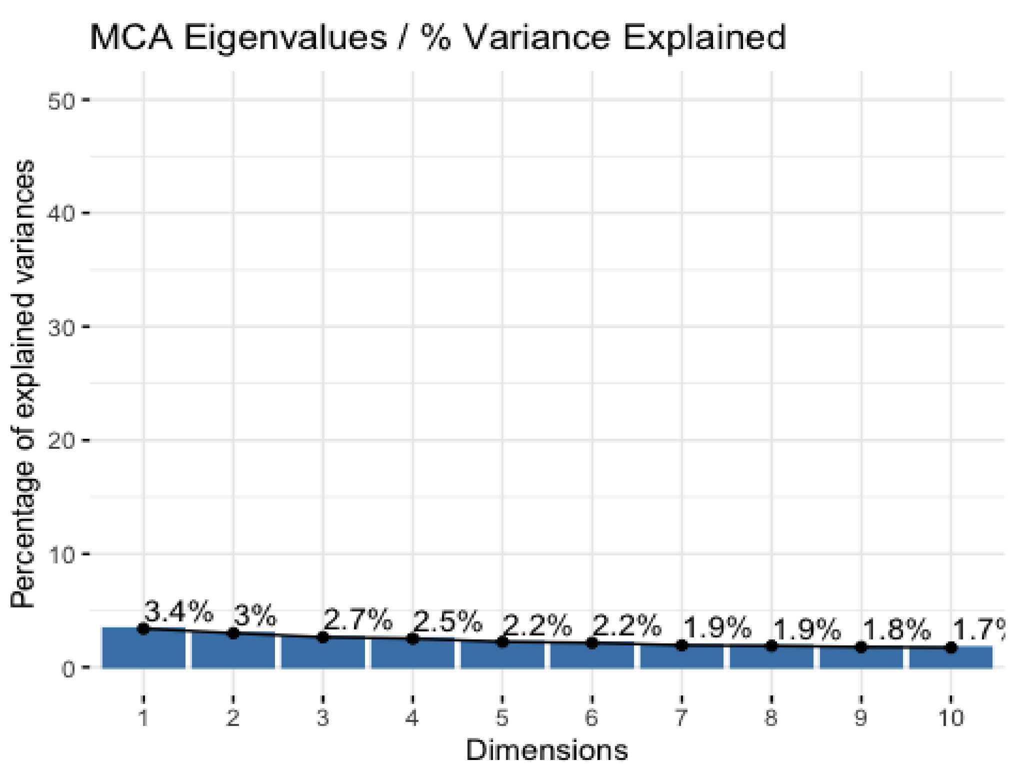


Findings:

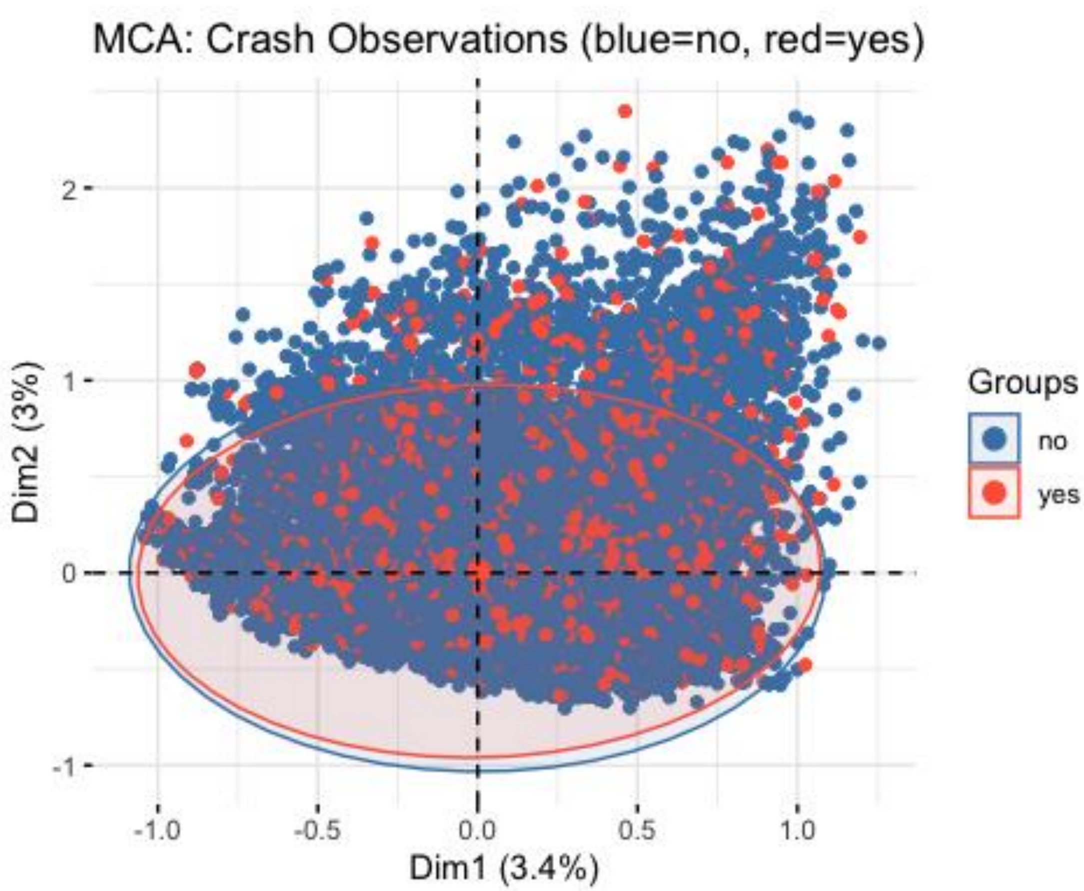
- *Lasso odds-ratios highlighted pedestrian-related and physical driver condition factors as highly influential.*
- *GBM SHAP analysis further identified crash type, contributory causes, and trafficway types as dominant factors.*
- *Partial-dependence plot revealed increased probability of severe injury with rising speed limits beyond 35 mph.*

Next step: Conduct a complementary categorical dimensionality reduction via MCA to explore underlying associations visually.

```r
# ── Section 9: Multiple Correspondence Analysis (MCA)
─────────────────────────

if (!requireNamespace("FactoMineR", quietly=TRUE))
install.packages("FactoMineR")
if (!requireNamespace("factoextra", quietly=TRUE))
install.packages("factoextra")
library(FactoMineR)
library(factoextra)

## Welcome! Want to learn more? See two factoextra-related books at
https://goo.gl/ve3WBa

# 9.1 Prepare categorical data
mca_data <- df2[, cat_vars]
for (i in seq_along(mca_data)) mca_data[[i]] <- as.factor(mca_data[[i]])

# 9.2 Run MCA
mca_res <- MCA(mca_data, graph = FALSE)

# 9.3 Scree plot
fviz_screeplot(
  mca_res,
  addlabels = TRUE,
  ylim      = c(0, 50),
  title     = "MCA Eigenvalues / % Variance Explained"
)
```

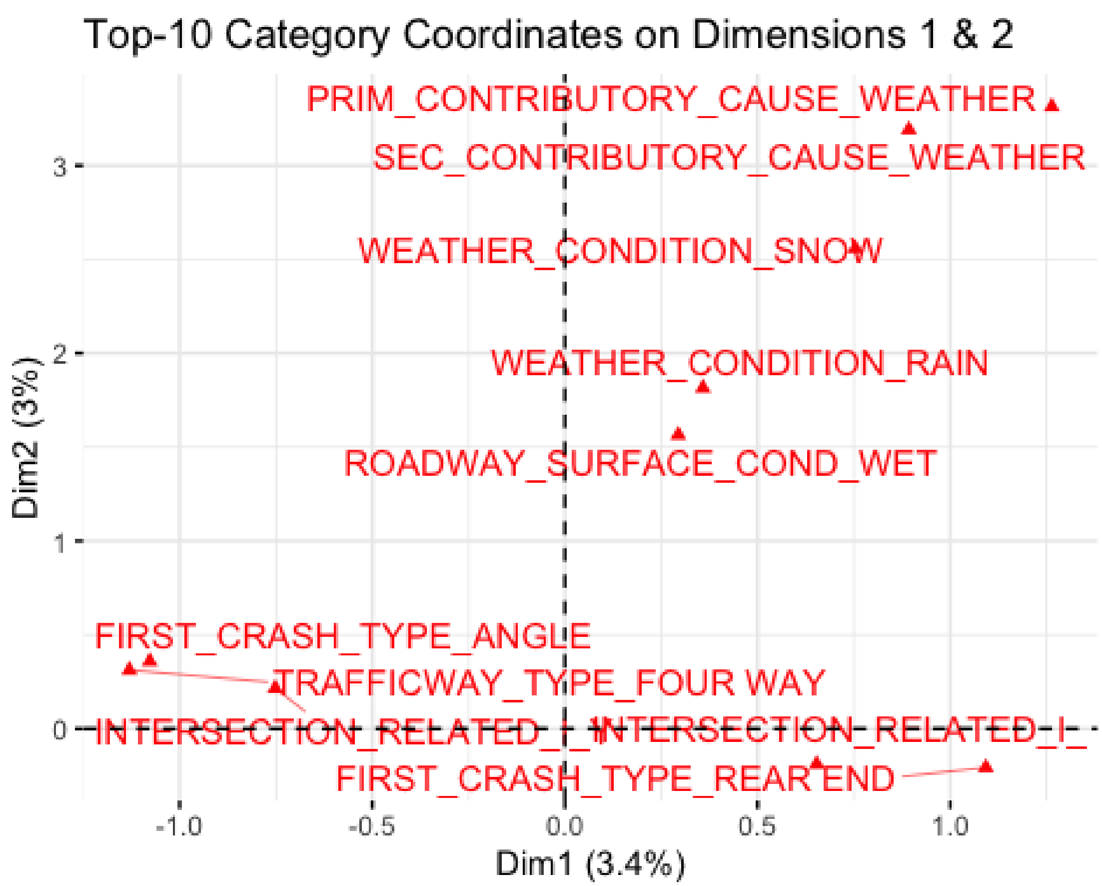


```r
# 9.4 Individuals map colored by severe injury
severe_factor <- factor(
  df2$SevereInjury,
  levels = c(0,1),
  labels = c("no","yes")
)
fviz_mca_ind(
  mca_res,
  geom          = "point",
  habillage     = severe_factor,
  palette       = c("steelblue","tomato"),
  addEllipses   = TRUE,
  ellipse.level= 0.95,
  repel         = TRUE,
  title         = "MCA: Crash Observations (blue=no, red=yes)"
)
```




```r
# 9.5 Top-10 category coordinates on Dim1 & Dim2
var_contrib   <- get_mca_var(mca_res)$contrib
total_contrib <- rowSums(var_contrib[, 1:2])
top10_cats    <- names(sort(total_contrib, decreasing = TRUE))[1:10]
fviz_mca_var(
  mca_res,
```

```
  select.var = list(name = top10_cats),
  repel      = TRUE,
  title      = "Top-10 Category Coordinates on Dimensions 1 & 2"
)
```



Findings:

- MCA dimensions revealed weather conditions (snow, rain), crash types (angle, rear-end), and intersection-related indicators as critical categorical drivers.
- Clear distinction between severe and non-severe injuries appeared along the first two MCA dimensions.

Problem Statement & Objective

This project aimed to identify and predict factors leading to severe injuries (fatal or incapacitating) in Chicago traffic crashes. Severe injuries accounted for approximately **18.6%** of crashes, presenting a notable class imbalance challenge.

Analytical Approach & Key Steps

A structured analytical framework was applied, leveraging key statistical and machine-learning methodologies:

- **Data Preparation:**
  - Collapsing rare categorical levels improved model manageability.
  - Stratified splits ensured consistent representation of severe injuries across training and testing subsets.
- **Exploratory Analysis:**
  - χ^2 and Cramér's V analyses identified crash types, contributory causes, and reporting methods as significantly associated factors.
- **Predictive Modeling:**
  - Baseline models (Logistic regression, LDA, QDA) exhibited limited sensitivity.
  - Regularized logistic regression (Lasso) improved model sensitivity (**55.9%**) and interpretability through optimized cutoff selection.
  - Advanced methods including CART, Bagging, Random Forest (RF), and Gradient Boosting Machines (GBM) provided enhanced predictive performance, with GBM achieving the highest AUC (**0.606**).
- **Model Interpretation:**
  - Lasso regression and GBM emphasized key predictors such as pedestrian involvement, physical driver impairment, crash severity, and high posted speed limits (particularly above **35 mph**).
- **Dimensionality Reduction via MCA:**
  - MCA reinforced analytical findings by visually differentiating severe and non-severe crashes primarily along dimensions of weather conditions, intersections, and crash type.

Key Conclusions & Actionable Recommendations

The analysis pinpointed several critical risk factors for severe injuries:

- **Pedestrian-related crashes** and **driver impairment** substantially increase severe injury odds.
- **High-speed limits** (>35 mph) notably elevate severe crash probabilities.
- **Intersection involvement** and **adverse weather conditions** (rain, snow) correlate strongly with severe injuries.

Recommended interventions include:

- Targeted infrastructure upgrades, particularly improved street lighting and safer pedestrian crossings.
- Enhanced speed enforcement and regulations, especially on high-speed road segments.
- Intersection-specific improvements focusing on design and preventive measures during adverse conditions.
- Educational and regulatory programs to reduce impaired driving behaviors.

Limitations & Future Directions

The study's observational design limits causal conclusions. Notably, the lack of behavioral data (such as driver distractions or compliance with regulations) restricts a deeper understanding of causal mechanisms. Future research could benefit from:

- Integrating behavioral and real-time traffic volume data for more robust insights.
- Using longitudinal or experimental designs to strengthen causal interpretations.
- Employing advanced data balancing and ensemble modeling methods to further optimize prediction accuracy.

This comprehensive data-driven analysis offers valuable, actionable insights to mitigate severe injuries in Chicago traffic incidents, providing a solid foundation for informed policy-making and safety improvements.