Using Variables of Traffic Crashes to Predict Crash Severity

STAT 480 Final Project

Farid Saud and Christopher Cebra

Introduction and Data Analysis

Overview

- Traffic crashes are one of the leading causes of fatalities and serious injuries in America, especially among people under 50[1]
- Traffic crashes are multifaceted, and may have many causes and situations
- The City of Chicago Data Portal's Traffic Crashes dataset includes information about ~900,000 crashes taking place between 2015 and 2024

★ TRAFFIC SAFETY ★

Preventing severe crashes on our City's streets. Data analysis and policy research help CDOT address traffic safety concerns centered around the following issues:

Vehicle Speed & Size

68% of Chicago traffic deaths involved drivers traveling at high speeds. Nearly half of pedestrians killed in the city are hit by an SUV or larger vehicle.

Reckless Driving

84% of traffic deaths in Chicago involve reckless behavior by people behind the wheel.

Persistent Inequities

Chicagoans who face the greatest barriers to health, income, and personal safety are also the most likely to die in traffic crashes.

Table about traffic crashes in Chicago (Source: Vision Zero)

1: https://wisqars.cdc.gov/animated-leading-causes/

Explanatory Variables

Explanatory (Factors)

TRAFFIC_CONTROL_DEVICE
DEVICE_CONDITION
WEATHER_CONDITION
LIGHTING_CONDITION
FIRST_CRASH_TYPE
TRAFFICWAY_TYPE
ALIGNMENT
ROADWAY_SURFACE_COND
ROAD_DEFECT
REPORT_TYPE
CRASH_TYPE
DAMAGE
PRIM_CONTRIBUTORY_CAUSE
SEC_CONTRIBUTORY_CAUSE
STREET_DIRECTION

Explanatory (Boolean)

CRASH_DATE_EST_I
INTERSECTION_RELATED_I
NOT_RIGHT_OF_WAY_I
HIT_AND_RUN_I
PHOTOS_TAKEN_I
STATEMENTS_TAKEN_I
DOORING_I
WORK_ZONE_I
WORKERS_PRESENT_I

Explanatory (Other)

CRASH_RECORD_ID (string)
CRASH_DATE (time)
SPEED_LIMIT (numeric)
LANE_CNT (numeric)
DATE_POLICE_NOTIFIED
STREET_NAME (string)
CRASH_HOUR
CRASH_DAY
CRASH_MONTH
LATITUDE
LONGITUDE

Data Problems

Data Setup

- Each row of the dataset corresponds to a crash.
    - The rate of crashes under various conditions cannot be studied.
    - Instead, we must use response variables as proxies for crash severity, such as injuries (INJURIES_TOTAL, INJURIES_FATAL, INJURIES_INCAPACITATING, DAMAGE)

Missing Data

- All of the Boolean variables contain “Y” if true, otherwise blank
    - Must assume that an empty cell implies No although could be no data
- Some explanatory variables, including LANE_CNT, have lots of missing values but are not Boolean

Data Size and Computation Problems

Data Size

- Overall dataset was 889,931 rows and 48 columns (*initially*), 481.7 MB
  - Too big for Github, we split it into 5 “folds”, rebuild the dataset in R/Python.
  - After data processing and design matrix, 260+ plus columns due to factors with many levels. We use LASSO for variable selection!

Data Modelling

- For model selection:
  - Cross-validation can be very good when prediction accuracy is key.
  - AICc is computationally lighter than cross-validation and may be preferred for variable selection.
  - We used both:
    - sklearn.linear_model.LogisticRegressionCV()
    - gamlr()

Research Question(s)

- Is there a set of variables which are predictive of crash severity?
- Are there variables which are causal with crash severity?

Data Cleaning

Date Variables

- CRASH_DATE, DATE_POLICE_NOTIFIED
  - to datetime format

Missing Value Handling

- Filled with 'N' (No/Negative)
  - INTERSECTION_RELATED_I
  - NOT_RIGHT_OF_WAY_I
  - HIT_AND_RUN_I
  - DOORING_I
  - WORK_ZONE_I
  - WORKERS_PRESENT_I
  - CRASH_DATE_EST_I
- REPORT_TYPE:
  - Filled with 'UNKNOWN'
- WORK_ZONE_TYPE:
  - Filled with 'UNKNOWN'

Engineered Variables

- Report_vs_Police_Notified
  - Calculated hours between crash and police notification
  - Capped at 0-48 hours
- Crash_Year
  - Extracted from CRASH_DATE
- LANE_CNT
  - Binned into: 'NARROW', 'WIDE', 'HIGHWAY', 'NOT_APPLICABLE'
- Police_district
  - Grouped beats into 25 districts

Variables dropped

- PHOTOS_TAKEN_I, STATEMENTS_TAKEN_I, CRASH_RECORD_ID,
- STREET_NO, STREET_DIRECTION, STREET_NAME, BEAT_OF_OCCURRENCE, LOCATION, LATITUDE & LONGITUDE
- INJURIES_UNKNOWN (completely empty)
- *DAMAGE**

Data Modelling

Model Building

Appropriateness of the model(s) selected for the problem

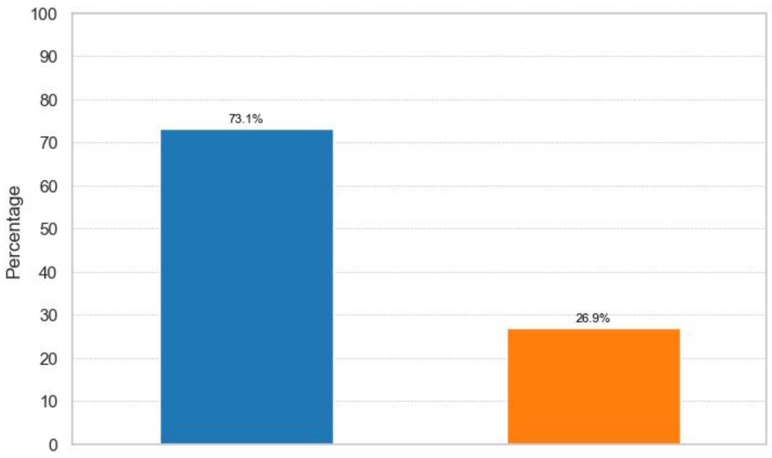
df['CRASH_TYPE']

- NO INJURY / DRIVE AWAY
- INJURY AND / OR TOW DUE TO CRASH

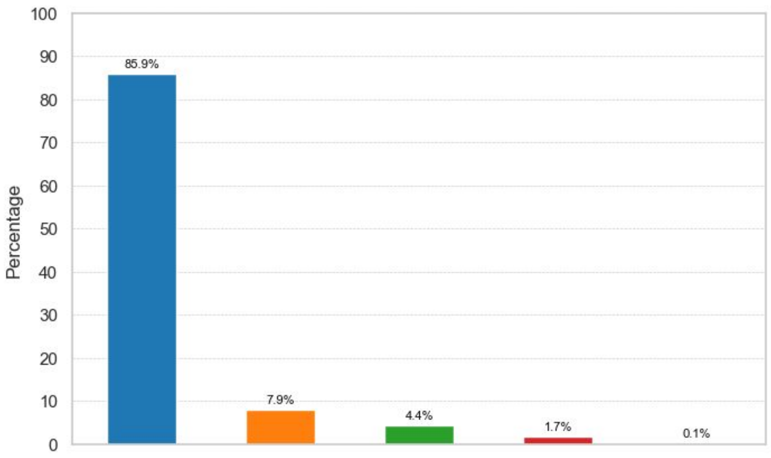
df['MOST_SEVERE_INJURY']

- NO INDICATION OF INJURY
- NONINCAPACITATING INJURY
- REPORTED, NOT EVIDENT
- INCAPACITATING INJURY
- FATAL

Lasso **Binomial** Logistic Regression



Lasso **Multinomial** Logistic Regression



Model Building

Implementation of models

```
# Seed
np.random.seed(480)
```

Binomial Logistic Regression

```
import patsy

# Binary response variable
# 2 classes:
response = 'CRASH_TYPE'

predictors = df.drop(columns=responses+not_usefull+date_vars).columns

formula = f"{response} ~ {' + '.join(predictors)}"

_, X = patsy.dmatrices(formula, data = df, return_type='dataframe')
y = df[response]
```

Multinomial Logistic Regression

```
import patsy

# Multiclass response variable
# 5 classes
response = 'MOST_SEVERE_INJURY'

predictors = df.drop(columns=responses+not_usefull+date_vars).columns

formula = f"{response} ~ {' + '.join(predictors)}"

_, X = patsy.dmatrices(formula, data = df, return_type='dataframe')
y = df[response]
```

```
import os

# CPU cores available
num_cores = os.cpu_count()
print("Total CPU cores available:", num_cores)
```

✓ 0.0s

```
Total CPU cores available: 8
```

```
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegressionCV

# Center & Scale features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X_sample)


# Lasso Logistic Regression
lasso_logistic = LogisticRegressionCV(
    penalty='l1',
    solver='saga',  # SAGA supports L1 regularization, faster for large datasets & Multi-class/Multinomial
    cv=5,
    max_iter=1000,
    tol=1e-3,
    random_state=480,
    Cs=10,  # Number of lambda values,
    n_jobs=num_cores-1,  # Use all processors -1
)

lasso_logistic.fit(X_scaled, y)
```

LogisticRegressionCV

```
LogisticRegressionCV(cv=5, max_iter=1000, n_jobs=7, penalty='l1',
                     random_state=480, solver='saga', tol=0.001)
```

Model Building

Implementation of models

```
```{r}
set.seed(480)
```
```

```
```{r}
### Design Matrix
response <- "CRASH_TYPE"


predictors <- df %>%
  select(-all_of(c(responses, not_useful, date_vars))) %>%
  colnames()


formula <- as.formula(paste(response, "~", paste(predictors, collapse = " + ")))


X <- model.matrix(formula, data = df)
y <- df[[response]]


# Treatment variable
d <- X[, "PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN ARREST IS EFFECTED)"]
XX <- X[, colnames(X) != "PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN ARREST IS EFFECTED)"]
```
```

```
#### Double Lasso

## 1st Lasso
lasso1 <- gamlr(x = XX, y = d, family = "binomial", standardize=TRUE)
B1 <- coef(lasso1)

min.AICc.lambda <- lasso1$lambda[ which.min( AICc(lasso1) ) ]
paste("Min AICc lambda: ", min.AICc.lambda)

# Predicted treatment
d_hat <- predict(lasso1, XX, type = "response")

# R squared
r2 <- cor(drop(d_hat),d)^2
paste("In-sample R^2: ", r2)



## 2nd Lasso
lasso2 <- gamlr(x = cbind(d, d_hat, XX), y = y , free=2, family = "binomial", standardize=TRUE)
B2 <- coef(lasso2)

min.AICc.lambda <- lasso2$lambda[ which.min( AICc(lasso2) ) ]
paste("Min AICc lambda: ", min.AICc.lambda)

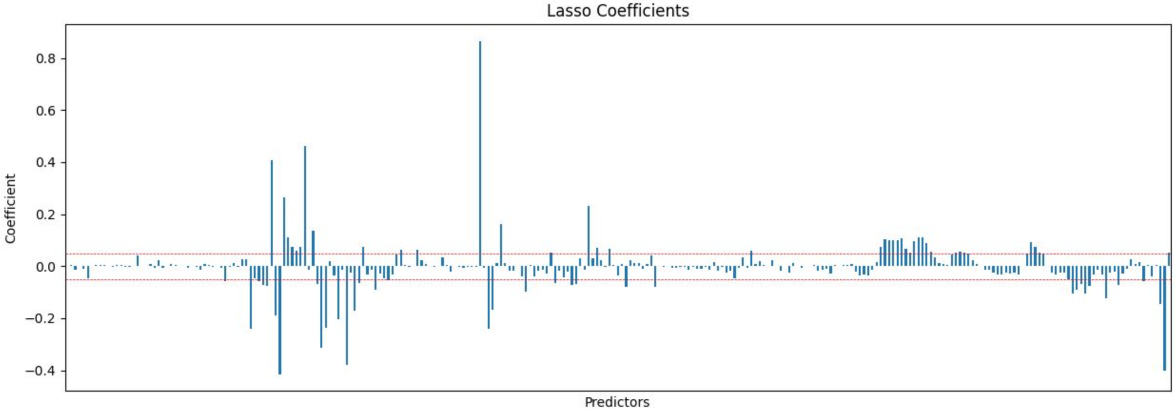
# Treatment effect after controlling for confounders
treatment_effect <- B2[2]
treatment_effect
```

Binomial Model

- Removed DAMAGE variable because output Binomial includes Tow Due to Crash
  - Otherwise DAMAGE was the most significant coefficient.
- Potential issue with LASSO (or solver) not converging–no zero coefficients

df['CRASH_TYPE']

- NO INJURY / DRIVE AWAY
- INJURY AND / OR TOW DUE TO CRASH



Most Connected with Severe Crashes →

	0
FIRST_CRASH_TYPE[T.PEDESTRIAN]	-0.4146
NUM_UNITS	-0.4006
TRAFFICWAY_TYPE[T.NOT DIVIDED]	-0.3791
TRAFFICWAY_TYPE[T.DIVIDED - W/MEDIAN (NOT RAISED)]	-0.3134
FIRST_CRASH_TYPE[T.FIXED OBJECT]	-0.2399
REPORT_TYPE[T.ON SCENE]	-0.2399
TRAFFICWAY_TYPE[T.DIVIDED - W/MEDIAN BARRIER]	-0.2380
TRAFFICWAY_TYPE[T.FOUR WAY]	-0.2047
FIRST_CRASH_TYPE[T.PEDALCYCLIST]	-0.1908
TRAFFICWAY_TYPE[T.ONE-WAY]	-0.1729

Least Connected with Severe Crashes →

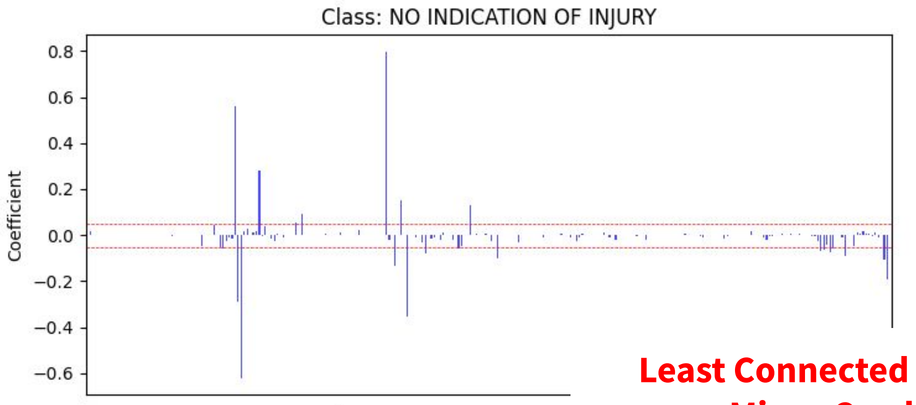
	0
REPORT_TYPE[T.NOT ON SCENE (DESK REPORT)]	0.8661
FIRST_CRASH_TYPE[T.SIDESWIPE SAME DIRECTION]	0.4609
FIRST_CRASH_TYPE[T.PARKED MOTOR VEHICLE]	0.4085
FIRST_CRASH_TYPE[T.REAR END]	0.2638
PRIM_CONTRIBUTORY_CAUSE[T.IMPROPER BACKING]	0.2318
HIT_AND_RUN_I[T.Y]	0.1620
FIRST_CRASH_TYPE[T.TURNING]	0.1348
CRASH_HOUR[T.16]	0.1115
FIRST_CRASH_TYPE[T.REAR TO FRONT]	0.1105
CRASH_HOUR[T.17]	0.1101

Multinomial Model—No Indication of Injury

- Of 267 variables and factor levels, **104 are nonzero** in Multivariate LASSO (intercept included)

df['MOST_SEVERE_INJURY']

- NO INDICATION OF INJURY
- NONINCAPACITATING INJURY
- REPORTED, NOT EVIDENT
- INCAPACITATING INJURY
- FATAL



Most Connected with Minor Crashes →

	NO INDICATION OF INJURY
REPORT_TYPE[T.NOT ON SCENE (DESK REPORT)]	0.7988
FIRST_CRASH_TYPE[T.PARKED MOTOR VEHICLE]	0.5613
FIRST_CRASH_TYPE[T.SIDESWIPE SAME DIRECTION]	0.2817
HIT_AND_RUN_I[T.Y]	0.1489
PRIM_CONTRIBUTORY_CAUSE[T.IMPROPER BACKING]	0.1296
TRAFFICWAY_TYPE[T.PARKING LOT]	0.0930
TRAFFICWAY_TYPE[T.ONE-WAY]	0.0567
LIGHTING_CONDITION[T.UNKNOWN]	0.0466
FIRST_CRASH_TYPE[T.TURNING]	0.0386
FIRST_CRASH_TYPE[T.REAR TO FRONT]	0.0292
ROADWAY_SURFACE_COND[T.SNOW OR SLUSH]	0.0244
CRASH_MONTH[T.2]	0.0188

Least Connected with Minor Crashes →

	NO INDICATION OF INJURY
FIRST_CRASH_TYPE[T.PEDESTRIAN]	-0.6203
DAMAGE[T.OVER $1,500]	-0.3527
FIRST_CRASH_TYPE[T.PEDALCYCLIST]	-0.2913
NUM_UNITS	-0.1918
INTERSECTION_RELATED_I[T.Y]	-0.1356
POSTED_SPEED_LIMIT	-0.1083
PRIM_CONTRIBUTORY_CAUSE[T.PHYSICAL CONDITION OF DRIVER]	-0.1031
Police_district[T.District 11]	-0.0882
PRIM_CONTRIBUTORY_CAUSE[T.DISREGARDING TRAFFIC SIGNALS]	-0.0776
Police_district[T.District 06]	-0.0724

Multinomial Model—Incapacitating Injury

- Of 267 variables and factor levels, **10 are nonzero** in Multivariate LASSO (intercept included)

df['MOST_SEVERE_INJURY’]

- NO INDICATION OF INJURY
- NONINCAPACITATING INJURY
- REPORTED, NOT EVIDENT
- **INCAPACITATING INJURY**
- FATAL

	INCAPACITATING INJURY
FIRST_CRASH_TYPE[T.PEDALCYCLIST]	0.0867
FIRST_CRASH_TYPE[T.PEDESTRIAN]	0.0670
PRIM_CONTRIBUTORY_CAUSE[T.PHYSICAL CONDITION OF DRIVER]	0.0324
DAMAGE[T.OVER $1,500]	0.0308
FIRST_CRASH_TYPE[T.FIXED OBJECT]	0.0221
FIRST_CRASH_TYPE[T.HEAD ON]	0.0030
PRIM_CONTRIBUTORY_CAUSE[T.UNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN ARREST IS EFFECTED)]	0.0027

Most Connected with Injurious Crashes

	INCAPACITATING INJURY
REPORT_TYPE[T.NOT ON SCENE (DESK REPORT)]	-0.1058
FIRST_CRASH_TYPE[T.REAR END]	-0.0145

Least Connected with Injurious Crashes

Causality Test - Double LASSO

Data Conclusions

- We test for causal effects on two variables in the binary response model: ‘Roadway Surface Condition/Snow or Slush’ and ‘Primary Cause/Under the Influence of Alcohol or Drugs’
    - Snow or Slush was positively associated with **NO INJURY / DRIVE AWAY**
    - Under the Influence of Alcohol or Drugs was negatively associated with **NO INJURY / DRIVE AWAY**
- The causality test will be conducted by double LASSO

Causality Test: DUI

- Performed double LASSO on Under the Influence of Alcohol or Drugs variable:

```r
# Treatment variable
d <- X[, "PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN ARREST IS EFFECTED)"]
XX <- X[, colnames(X) != "PRIM_CONTRIBUTORY_CAUSEUNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN ARREST IS EFFECTED)"]


#### Double Lasso

## 1st Lasso
lasso1 <- gamlr(x = XX, y = d, family = "binomial", standardize=TRUE)
B1 <- coef(lasso1)

min.AICc.lambda <- lasso1$lambda[ which.min( AICc(lasso1) ) ]
paste("Min AICc lambda: ", min.AICc.lambda)

# Predicted causal variable
d_hat <- predict(lasso1, XX, type = "response")

# R squared
r2 <- cor(drop(d_hat),d)^2
paste("In-sample R^2: ", r2) # 0.928505221949197


## 2nd Lasso
lasso2 <- gamlr(x = cbind(d, d_hat, XX), y = y , free=2, family = "binomial", standardize=TRUE)
B2 <- coef(lasso2)

min.AICc.lambda <- lasso2$lambda[ which.min( AICc(lasso2) ) ]
paste("Min AICc lambda: ", min.AICc.lambda)

# Treatment effect after controlling for confounders
treatment_effect <- B2[2] # 0
treatment_effect
```

```
[1] "In-sample R^2:  0.928505221949197"
[1] 0
```

← **After controlling for confounders**

PRIM_CONTRIBUTORY_CAUSE[T.UNDER THE INFLUENCE OF ALCOHOL/DRUGS

-0.0803047680626725

↑ **Initially**

Causality Test: Snow or Slush

- Performed double LASSO on Snow and Slush road condition variable:

```r
```{r}
# Treatment variable
d <- X[, "ROADWAY_SURFACE_CONDSNOW OR SLUSH"]
XX <- X[, colnames(X) != "ROADWAY_SURFACE_CONDSNOW OR SLUSH"]


#### Double Lasso

## 1st Lasso
lasso3 <- gamlr(x = XX, y = d, family = "binomial", standardize=TRUE)
B3 <- coef(lasso3)

# Predicted causal variable
d_hat <- predict(lasso3, XX, type = "response")

# R squared
r2 <- cor(drop(d_hat),d)^2
paste("In-sample R^2: ", r2) # 0.700471566598306


## 2nd Lasso
lasso4 <- gamlr(x = cbind(d, d_hat, XX), y = y , free=2, family = "binomial", standardize=TRUE)
B4 <- coef(lasso4)

# Treatment effect after controlling for confounders
treatment_effect <- B4[2]
treatment_effect # 0.
```
```

```
[1] "In-sample R^2:  0.700471566598306"
[1] 0
```

← **After controlling for confounders**

ROADWAY_SURFACE_COND[T.SNOW OR SLUSH

0.0332039599271212

↑

Initially

Model Building

Implementation of models

Hydro Partitions (Queues)

Hydro Partitions/Queues

Partition (Queue)	Node/Job Type	Max Nodes per Job	Max Duration	Max Running in Queue/user	Charge Factor
sandybridge	CPU (Intel)	TBD	7 days	TBD	1.0
sandybridge2.9	CPU (Intel)	TBD	7 days	TBD	1.0
sandybridge2.0	CPU (Intel)	TBD	7 days	TBD	1.0
interlagos	CPU (AMD)	TBD	7 days	TBD	1.0
milan	CPU (AMD)	TBD	7 days	TBD	6.0
rome	CPU (AMD)	TBD	7 days	TBD	6.0
a100	dual A100 GPU w/ any CPU	TBD	7 days	TBD	20.0
a100milan	dual A100 GPU w/ Milan CPU	TBD	7 days	TBD	20.0
a100rome	dual A100 GPU w/ Rome CPU	TBD	7 days	TBD	20.0

- Sandybridge nodes have 16 cores per node, dual-socket, 384MB (2.9 and 2.0 GHz).
- Interlagos nodes have 64 cores per node, quad-socket, 512MB.
- Milan nodes have 56 cores per node, dual socket, 256MB.
- Rome nodes have 64 cores per node, dual socket, 256MB.

Dell PowerEdge R815 Compute Node Specifications

- Number of nodes: 4
- Quad Socket (4) (16 core, AMD Interlagos) @ 2.30GHz (64 cores per node)
- 512 GB of memory
- Cache L1/L2/L3: .768/16/16 MB; L3 Total: 32 MB
- NUMA domains: 2 per socket, 8 per node
- CPUs per NUMA: domain0={0-7} domain1={8-15} domain2={32-39} domain3={40-47} domain4={48-55} domain5={56-63} domain6={16-23} domain7={24-31}
- 40 Gb/s Ethernet
- QDR 40 Gb/s InfiniBand

Conclusions and Next Steps

Conclusions

Data Conclusions

- More serious crashes originate from higher relative speed and momentum.
  - Factors like Sideswipe and Rear-End collisions have negative relations with Incapacitating Injury, and with No Injury/Tow Away in our binomial model
  - Factors like Posted Speed Limit, and collisions with Cyclists and Pedestrians have positive relations with Incapacitating Injury and Injury/Tow Away
- Some variables are likely highly correlated with response variables.
  - Consistently, the most negatively correlated with severe crashes is Report Type–Not on Scene (Desk Report), Desk Reports are likely only filed for non-severe crashes
  - Damage variable highly correlated with severe crashes as well.

Other Conclusions

- The threshold for a variable being causal is clearly higher than the threshold for it being significant in the single LASSO model. Both variables we tested were significant but not causal.

Next Steps

Data Models

- We can extend the causality test to all variables, or all variables under a certain threshold
- For original LASSO models, we can calculate significance and standard errors through another method (e.g. bootstrap), especially for binomial regression where all fitted values are nonzero
- Two methods not done in this presentation for runtime reasons

Other Approaches

- We could consider other modeling techniques entirely, or use geographic location data and time span of dataset to create things that “look like experiments” at a smaller scale in the city.

Questions?