Using Machine Learning to Predict Collision Severity

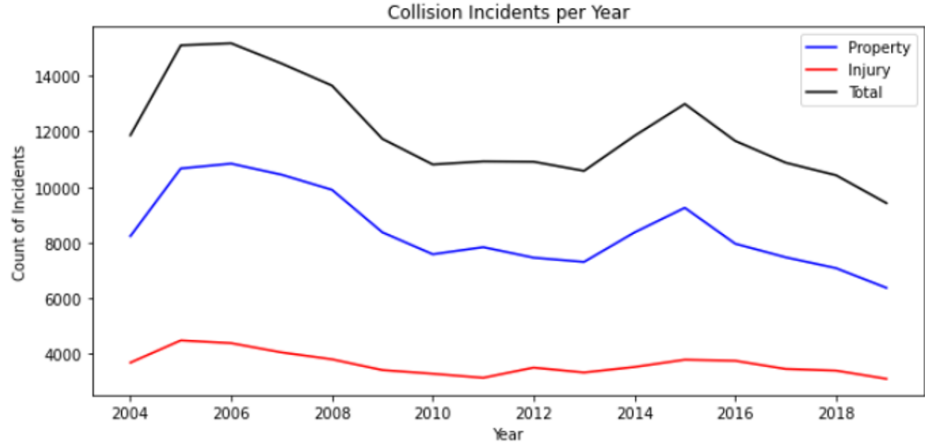
Seattle Department of Transportation Data from 2004 – 2020

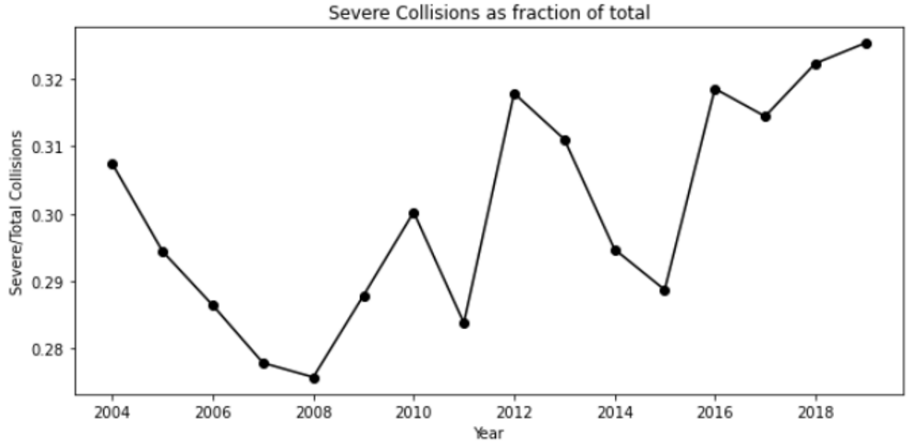
Coursera – IBM Data Science Capstone Project

Sam Steffes

Problem Statement

- Seattle Department of Transportation released a 10-year Strategic Vision in 2015, aimed at providing safe a reliable infrastructure.
- This included a Vision Zero goal of eliminating serious and fatal crashes by 2030; however, since then, serious collisions have accounted for a higher proportion of collisions year-over-year.





Problem Statement pt. II

- Can use Machine Learning to study historic data and predict the severity of collisions
- Possible benefits:
  - Better understand the factors that most contribute to severe collisions, allowing more targeted improvements
  - Provide a way to quantify and select the improvement alternatives that have the most impact (i.e. greatest reduction in severe collisions)
  - Support the development and deployment of decision-support tools that optimize emergency response resource allocation or dispatching (‘go’/‘no-go’)

Data

- Data used for modeling comes from historic collision records maintained by the SDOT Traffic Management Division
- >195,000 samples from January 2004 to May 2020
- 36 different features that describe:
  - Date, time and position of collisions
  - Codes used by the DOT and state to categorize collisions
  - Incident characteristics such as number of people, or vehicles
  - Environmental conditions at the time of the collision
  - Driver behaviors such as speeding, inattentiveness or drug/alcohol use
- Clear target variable, with cases labeled as either ‘1-property damage only’ or ‘2-injury collision’
  - This makes it ideal for supervised classification models

Methodology

- Exploratory Analysis
- Feature selection
- Data cleansing, processing and balancing
- Model training and evaluation

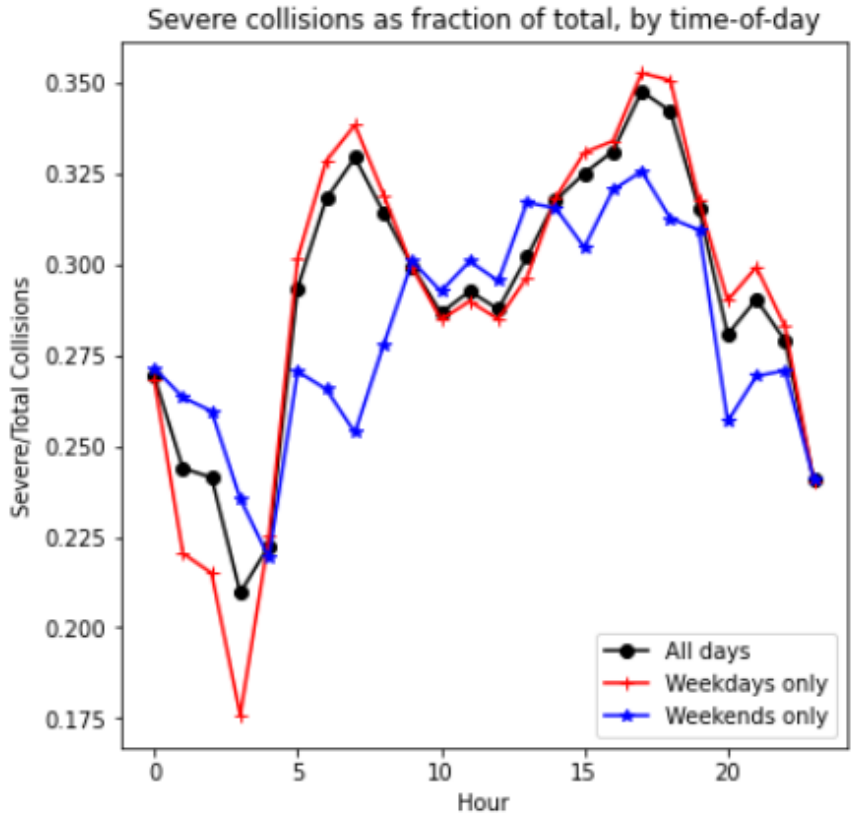
Initial Feature Selection

- 12 features were initially identified as potential modeling features, including: collision type, weather, road and lighting conditions, date and time, etc.

Column Name	Description	Possible Application/Value/Impact to model
COLLISIONTYPE	A keyword describing the collision, eg 'head-on', 'angled', 'cycles', etc.	The orientation or nature of the collision could be valuable to the likelihood of injury to people involved.
PERSONCOUNT	Total number of people involved in the collision.	More people involved in the collision means more chances for injuries... people could be seated in the vehicle in spots that were closer to the impact, etc.
PEDCOUNT	The number of pedestrians involved in the collision.	Pedestrians are at greater risk of injury when struck by vehicles because they are not protected by another vehicle frame, seatbelt, airbag, etc.
PEDCYLCOUNT	The number of bicycles involved.	As with pedestrians, bicyclists are also at greater risk of injury when struck by vehicles.
VEHCOUNT	The number of vehicles involved in the collision.	More vehicles are more chances for injury, and might suggest more extensive impact, damage or severity.
INCDATE/INCDTTM	Date and time recordings for the incident records.	This is not as clear; however, the date or time that an incident occurs may be a proxy for driving behaviours that result in more severe incidents. For example, the are higher traffic volumes during weekday rushhours, and people are in a hurry or distracted.
INATTENTIONIND	If collision was due to inattention.	Distracted driving certainly increases the probabibily of getting in an accident, and may affect severity as well.
UNDERINFL	If driver was under influence of drugs/alchohol.	Someone under the influence might be driving recklessly, causing the collision to be more severe, and may also not be wearing their seatbelt, increasing their own injury risk.
WEATHER	Weather at the time of incident.	Weather could have contributed to the nature of the collision, or the control of the vehicles before and after the collision.
ROADCOND	Condition of road at the time of incident.	This is probably closely aligned with weather, but could also capture the maintenance condition of the roadway.
LIGHTCOND	Light conditions at the time of incident.	Poor visibility may have contributed to the collision, especially where there are already blind or hidden turns, difficult or dangerous maneuvers are required, or acceptable speeds are already high.
SPEEDING	If speeding was a factor in the collision.	Speeding or reckless driving could result in more forceful collisions.

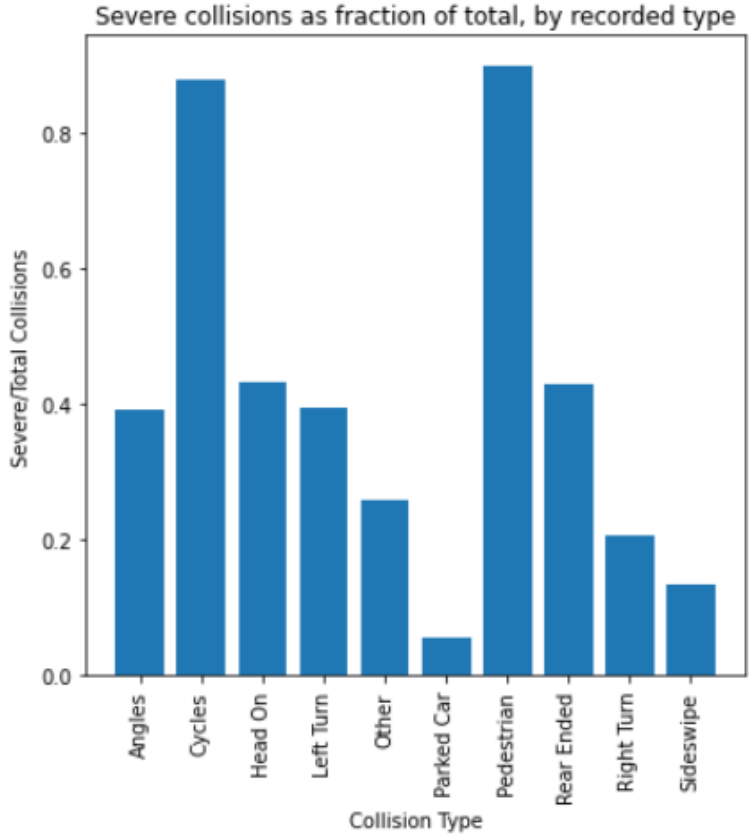
Exploratory Analysis:
Impact of Date and Time

- No disparity in rate of injury collisions for different days of the week
- However, there were clear peaks in the injury rate during common rush-hour periods (5-8 AM and 3-6 PM)



Exploratory Analysis:
Type of Collision

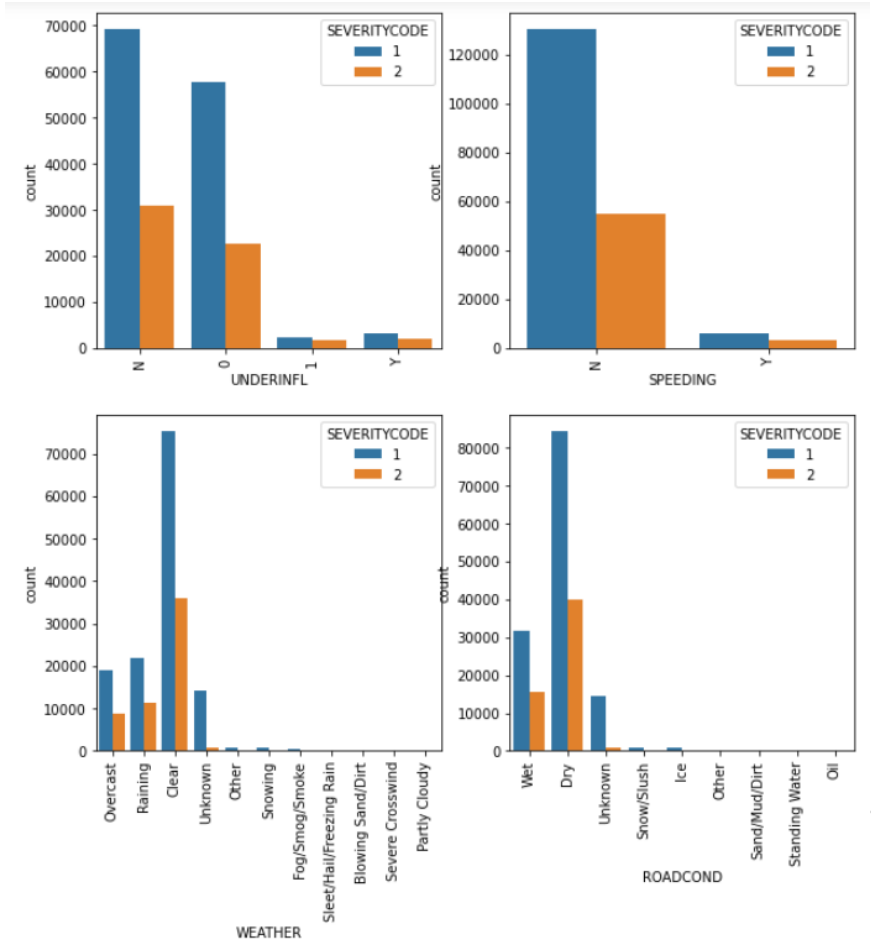
- High rate of injury type collisions for those involving pedestrians and bicycles
  - Unprotected people are more likely to be injured when struck by a heavier object
- Low rate of injury for collisions involving parked cars
  - Parked cars are likely empty and near locations with lower speed limits



Exploratory Analysis:

Environmental Conditions and Driver Behaviors

- Weather, road and light conditions did not have a significant impact as originally expect
- Driver impacts and distractions did



Final Feature Selection

- The parameters selected for model training included:
  - Collision type – converted to one-hot encoding for the 10 unique types
  - Person count – samples separated by 3 person threshold (more = 1, less = 0)
  - Vehicle count – samples separated as 2-vehicle or otherwise (yes = 1, no = 0)
  - Time – samples separated into rush-hour or non-rush-hour periods (yes = 1, no = 0)
  - Inattention – existing Yes/No field, converted to 1/0
  - Under influence – existing format mixed Y/N and 1/0, converted to 1/0
  - Speeding – existing Y/N field, converted to 1/0

Data Cleansing

- ~4900 samples with blank COLLISIONTYPE and UNDERINFL
  - >>removed
- ~5500 samples with PERSONCOUNT of 0
  - >>removed
- ~25500 samples with no recorded time (default timestamp 00:00:00)
  - >>removed

```
#Looking at the distribution for PERSONCOUNT,
df['PERSONCOUNT'].value_counts()
```

```
2    111386
3     35138
4     14445
1     11727
5      6584
0      5541
6      2702
7      1131
8       533
9       216
```

```
In [7]: #lets check how many blank values there are in each column:
        df.isnull().sum(axis=0)

Out[7]: DATETIME            0
        COLLISIONTYPE     4904
        PERSONCOUNT          0
        VEHCOUNT             0
        INATTENTIONIND       0
        UNDERINFL         4884
        SPEEDING             0
        SEVERITYCODE         0
        dtype: int64
```

Data Processing

- PERSONCOUNT, VEHCOUNT and INCDTTM manipulated to create simplified features that split the samples according to:
  - More or less than 3 people involved
  - Collisions involving 2 vehicles or other
  - Collisions occurring during or outside of AM/PM rush-hour periods
- COLLISIONTYPE transformed into one-hot encoding

```
#add columns: PCOUNT_OVER2, VCOUNT_IS2 to track whether there are 3 or more people and whether there are 2 vehicles or other
df['PCOUNT_OVER2'] = [1 if b>=3 else 0 for b in df['PERSONCOUNT']]
df['VCOUNT_IS2'] = [1 if b==2 else 0 for b in df['VEHCOUNT']]
df.head()
```

Data Balancing

- After cleansing, there was still an imbalance in the number of samples from each class
  - Majority class of 110649 property damage only (class label ‘1’)
  - Minority class of 48052 injury collisions (class label ‘2’)
- Down-sampling of the majority class was used to balance the dataset

```
#lets observe how unbalanced the dataset is...
features_df['SEVERITYCODE'].value_counts()
```

```
1    110649
2     48052
Name: SEVERITYCODE, dtype: int64
```

Final Dataset

- Final dataset of 16 one-hot encoded features
  - 48052 (x2) samples from each class
  - No normalization required

	RUSHHOUR	PCOUNT_OVER2	VCOUNT_IS2	INATTENTIONIND	UNDERINFL	SPEEDING	Angles	Cycles	Head On	Left Turn	Other	Parked Car	Pedestrian	Rea Ende
0	1	0	1	0	0	0	0	0	0	0	0	1	0	
1	0	0	1	0	0	0	0	0	0	0	0	1	0	
2	1	0	0	0	0	0	0	0	0	0	0	0	1	
3	0	0	1	1	0	0	1	0	0	0	0	0	0	
4	0	0	1	0	0	0	1	0	0	0	0	0	0	

Modeling

- 4 different classification algorithms used: KNN, Decision Tree, SVM and Logistic Regression
- Data was split into train (75%) and test (25%) sub-sets
- Parameter tuning (optimization) completed as necessary within each algorithm
  - i.e. k-value (KNN), kernel function (SVM), solver and regularization (LR)
- Best model from each method evaluated using: simple accuracy, Jaccard similarity, F1-score, log-loss and AUROC

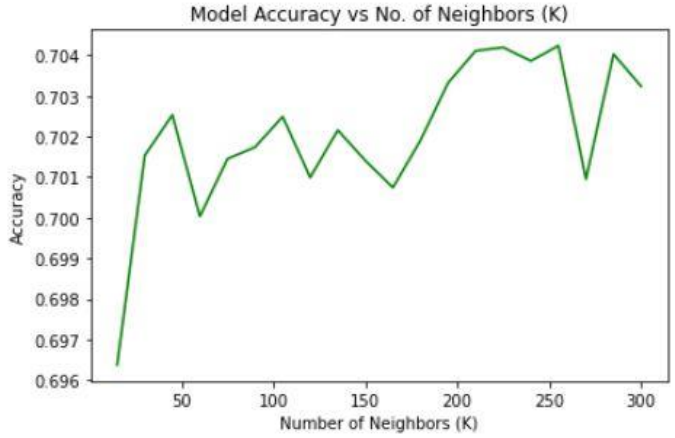
```
#using the test_train method, create test and train subsets of X and y
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=4)
print ('Train set:', X_train.shape,  y_train.shape)
print ('Test set:', X_test.shape,  y_test.shape)
```

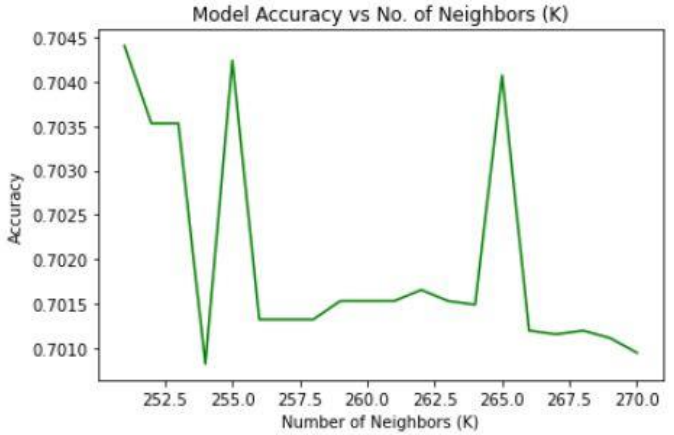
```
Train set: (72078, 16) (72078,)
Test set: (24026, 16) (24026,)
```

Modeling

K-nearest Neighbors

- k-value optimized by iterating through different values and plotting model accuracy
- Model performance was choppy on both macro & micro ranges of k
- k=250 selected





Modeling
Decision Tree

- RandomForest method used with no pre-set parameters

```
#train the model on the train subsets
RFDT = RandomForestClassifier().fit(X_train,y_train)
DT_pred = RFDT.predict(X_test)
print('Accuracy for Decision Tree: ',metrics.accuracy_score(y_test,DT_pred))
```

```
Accuracy for Decision Tree:  0.7050695080329643
```

Modeling

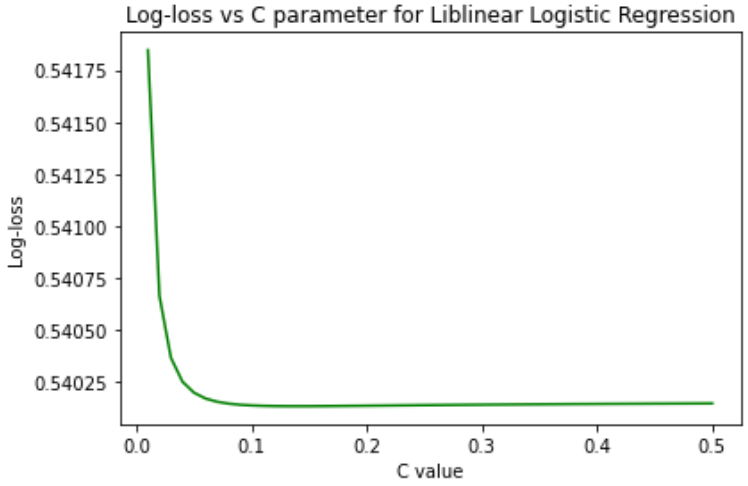
Support Vector Machines

- Models train with ‘linear’, ‘radial-based (rbf)’, ‘polynomial’ and ‘sigmoid’ kernel functions
- ‘linear’, ‘rbf’ and ‘polynomial’ models all achieved prediction accuracy between 0.702 and 0.705; sigmoid achieved 0.6299
- ‘linear’ model selected due to accuracy and slightly more balance predictions between target classes

Modeling

Logistic Regression

- Models trained for ‘liblinear’, ‘newton-cg’ and ‘sag’ solver functions
- C parameter optimized for selected ‘liblinear’ model by iterating through values and plotting log-loss
  - Optimum value of 0.14 selected
- Final C value taken as 0.01 due to higher overall prediction accuracy



Results

- Performance metrics consistent across all models – unexpected
  - Likely due to the specific features selected; some disorder still exists resulting in maximum achievable accuracy near 0.705
- Logistic Regression model performed best in 3 of 5 metrics; KNN in other 2

	Algorithm	Model feature	Accuracy	Jaccard	F1-score	Log-loss	AUROC
0	KNN	k=250	0.704612	0.507870	0.541453	0.540904	0.786865
1	Decision Tree	Random Forest	0.705070	0.503503	0.541335	0.547774	0.788573
2	SVM	linear kernel	0.702572	0.506798	0.539207	0.605143	0.750791
3	Logistic Regression	liblinear, C=0.01	0.705985	0.500636	0.541839	0.541847	0.790964

Results pt. II

- All models biased toward predicting injury type collisions
  - This may be desirable depending on the intended purpose of the model
- Precision of AUROC scores suggests well aligned models; exact amount of sensitivity results in different amounts of the more neutral samples edging across the boundary to injury classification

Confusion Matrices

KNN (w/ k=250) *Accuracy = 0.7046*

Actual Collision Type	-2297	2297**
1 - Property	7324	4697
2 - Injury	2400	9605
	1 - Property	2 - Injury

**Skew: 4578 (2>1)* — *Predicted Collision Type*

Decision Tree (w/ RandomForest) *Accuracy = 0.7051*

Actual Collision Type	-2584	2584
1 - Property	7186	4835
2 - Injury	2251	9754
	1 - Property	2 - Injury

Skew: 5152 (2>1) — *Predicted Collision Type*

SVM (w/ linear kernel) *Accuracy = 0.7026*

Actual Collision Type	-2210	2210
1 - Property	7343	4678
2 - Injury	2468	9537
	1 - Property	2 - Injury

Skew: 4404 (2>1) — *Predicted Collision Type*

Logistic Regression (w/ liblinear, C=0.01) *Accuracy = 0.7060*

Actual Collision Type	-2814	2814
1 - Property	7082	4939
2 - Injury	2125	9880
	1 - Property	2 - Injury

Skew: 5612 (2>1) — *Predicted Collision Type*

Actual sample populations: 1-Property = 12021, 2-Injury = 12005
**Skew = spread between predicted counts of each class*
***Predicted - Actual for same class*

Conclusions

- 4 models built to predict collision severity using data from Seattle DOT and popular classification algorithms
  - Prediction accuracy over 0.70 for each
  - Slight variations in sensitivity to predicting more severe collisions
- Optimum model could depend on:
  - Conservative or liberal requirements re: impact of severe collisions
  - Time consumption: KNN, SVM and Decision Tree models were slow to process
- Next steps:
  - Alternative features could be used from the same dataset
  - Apply models to data from other sources or DOTs to evaluate effectiveness
  - Explore impact of additional collision characteristics, such as vehicle make/model, posted vs actual vehicle speed, airbag release