Bike Crash Analyses

Elliott O'Brien

2021-08-07 13:39:58

Contents

1 Background and Motivation

Assuming I work for the state of North Carolina as a data scientist, I'm working on a project that will investigate bicycle safety in the state of North Carolina in a effort to understand what factors to consider in decreasing the number serious bicycle injuries as caused by bicycle-vehicle collisions.

In this report, I will be exploring machine learning models with the caret package in R and attempt to find the best model or ensemble model that can predict a bike injury being "serious" vs. "non-serious". The resulting model will be used to find the road, bicyclist, and driver features that most affect the seriousness of an injury.

2 Analytical Dataset

2.1 Data Description

The large number of variables (61) with multiple levels for some categorical variables.

Table 1: Dataset Feature Discription

var_n	Variable	Data Type
1	X	numeric
2	Y	numeric
3	AmbulanceR	character
4	BikeAge	character
5	BikeAgeGrp	character
6	BikeAlcDrg	character
7	BikeAlcFlg	character
8	BikeDir	character
9	BikeInjury	character
10	BikePos	character
11	BikeRace	character
12	BikeSex	character
13	City	character
14	County	character

var_n	Variable	Data Type
15	CrashAlcoh	character
16	CrashDay	character
17	CrashGrp	character
18	CrashHour	numeric
19	CrashID	numeric
20	CrashLoc	character
21	CrashMonth	character
22	CrashSevr	character
23	CrashType	character
24	CrashYear	numeric
25	Developmen	character
26	DrvrAge	character
27	DrvrAgeGrp	character
28	DrvrAlcDrg	character
29	DrvrAlcFlg	character
30	DrvrInjury	character
31	DrvrRace	character
32	DrvrSex	character
33	DrvrVehTyp	character
34	HitRun	character
35	Latitude	numeric
36	LightCond	character
37	Locality	character
38	Longitude	numeric
39	NumBicsAin	character
40	NumBicsBin	character
41	NumBicsCin	character
42	NumBicsKil	character
43	NumBicsNoi	character
44	NumBicsTot	character
45	NumBicsUin	character
46	NumLanes	character
47	NumUnits	numeric
48	RdCharacte	character
49	RdClass	character
50	RdConditio	character
51	RdConfig	character
52	RdDefects	character
53	RdFeature	character
54	RdSurface	character
55	Region	character
56	RuralUrban	character
57	SpeedLimit	character
58	TraffCntrl	character
59	Weather	character
60	Workzone	character
61	OBJECTID	numeric

The bike crashes dataset is sourced from the Department of Transportation in the state North Carolina from the years April 2007 - September 2019 (North Carolina Bicycle and Pedestrian Crash Data Tool (pedbikeinfo.org)). Raw data consists of 62 columns which are a mixture of numerical and character data types. The crash severity variable, CrashSevr, will be used as a response variable in a binary classification model, “Serious” vs. “Non-serious”. Currently the variable is categorical with multiple levels of severity:

Table 2: Raw bike injury class values

BikeInjury	n	pct
A: Suspected Serious Injury	637	5.23
B: Suspected Minor Injury	5021	41.25
C: Possible Injury	4685	38.49
K: Killed	269	2.21
O: No Injury	1187	9.75
Unknown Injury	374	3.07

However, since we are only interested in reducing severe injuries, the 5 categories above will be re-binned into ‘Serious’ and ‘Non-Serious’ injuries as follows:

- serious injury = {suspected serious injury OR killed}
- non-serious injury = {no injury OR possible injury OR suspected minor injury}

and assigned to a new variable BikeInjurySerious which will be our response variable in our models. Below is a summary of the prevalence of serious injuries to bicyclists. As it can be seen the classes are unbalanced with only 7.44% of the observations resulting in serious injuries to the bicyclist. This unbalance will be talked about in more detail in the performance metrics section below.

Table 3: Prevalence of Serious Bike Accidents

BikeInjurySerious	n	pct
No	11267	92.56
Yes	906	7.44

3 Performance Metrics

Due to the imbalance of the target classes, 7.44% are serious injuries and 92.56% non-serious, the performance metric accuracy will not be useful in selecting the best model. ROC area under the curve (AUC) is a more useful metric as it attempts to balance the true positive rate (sensitivity) and the false positive rate by optimizing the area under the curve created by graphing false positive rate vs. true positive rate.

After training the models on training data, optimum probability cut-off values will be explored using model predictions. The best probability cut-off will be determined based on the assumption that the cost of a false negative, predict non-serious injury when in-fact was a serious injury, is higher than a false positive, predict serious injury when in-fact was non-serious injury. The high cost for a false negative is higher because lives could potentially be lost or a life-changing injury, while the false positive only results in potentially spending more money on road safety for bicyclists which pays off in the long run anyways. Therefore, probability cut-offs will be chosen to optimize sensitivity (true positive rate) along with accuracy, with specificity (false negative rate) being the last priority. With this in mind, let’s assume that we want to have as close to 75% for sensitivity. Specificity can be relaxed as low as 50%. The models that don’t meet these criteria will not be included in an ensemble model.

4 Data Prep and Cleaning

data cleaning is done in this section using tidyverse methods.

4.1 Data Issues

There are many issues with the data set not being tidy and clean. Some variables will need to be converted to a numerical datatype for the machine learning models to work properly. Categorical character columns will need to be converted to factors. Dummy variables will be created for every column that is not numerical. This will likely increase the number of features in the dataset to the scale of hundreds of features. Feature reduction will be carried out to reduce memory usage and computation time of fitting models. Feature reduction will be carried out by removing features that have little to no variability and will thus not contribute much information to the models being used. Additionally, features that have high correlation with other features will be removed to improve model performance.

Some observations in the dataset that are missing data and these values have been imputed manually. For numerical missing data such as age missing values were imputed based on medians of age groups if age group variable is available, otherwise age was imputed using the median age of the full dataset based on the variable in question. For categorical variables, after being converted to dummy variables (i.e. 0 or 1 in value), missing values were imputed with a 0 to help in achieving a complete dataset as possible.

4.2 Unneccessary variables

The following variables are obviously copies of other variables.

- X and Y are longitude and latitude which are already present in the data
- OBJECTID and OBJECTID_1 seem like a row number id which is not a very useful feature. Also these are duplicated columns.

Deaths and bicyclst killed descrepency

Table 4: Death count discrepency of dataset

total_deaths	total_bike_killed	total_drvr_killed	total_bike_drvr_killed
278	269	2	271

278 total deaths, 7 deaths that aren't clear since there is no indication if bicyclist or driver was killed. 271 deaths that were biking or driving, mostly bicyclist (269) deaths. There are only 2 driver deaths in the whole dataset.

These are the 7 deaths that can't be explained by bicyclist(s) and driver(s). There seems to be a third party involved that is not listed in records, i.e. other pedestrian or other cyclist that didn't cause the crash but was affected by the crash.

Since we are most concerned with serious bicycle injuries, these records will be leaved as is since no serious injury to the bicyclist was recorded. However, it the CrashSevr and DrvrInjury variables will be removed since we already have BikeInjury variable.

Duplicates

Note duplicate crash report for CrashID == 1041566349; note that only CrashLoc and CrashType are different. Based on the first record, the driver is taking a right turn, but listed as non-intersection; in second record, CrashType is "motorist overtaking - Other/Unknown" and Crash Location is listed as intersection. Therefore, the CrashLoc was likely an intersection where a motorist overtook a bicyclist. This crash record will need to be fixed to reflect this and remove duplication.

Table 5: Duplicate Record CrashID

CrashID
104156349

4.3 Data cleaning

In this section, all variables are cleaned in the following ways:

- categorical variable levels are carefully structured so that names are as consistent as possible
- "unknown", "missing" or "." values converted to explicit NA.
- ordinal categorical variables (e.g. months, days of the week, age groups, etc.) factor levels are ordered appropriately
- strings fixed to use less special characters
- derivation of response variable, BikeInjurySerious = "Serious" if "killed" or "suspected serious injury"

4.4 Missing data

The amount of data missing throughout the dataset is noticeable and should imputed if possible. NumBics columns have the most missing data with 54.9% missing. It's unlikely that NumBics will be able to be imputed since majority values are missing.

Table 6: Variables with missing values

Variable	% missing
BikeAge	0.7
BikeAgeGrp	0.4
BikeAlcDrg	27.5
BikeAlcFlg	4.1
BikeDir	3.7
BikePos	4.9
BikeRace	0.6
BikeSex	0.2
CrashGrp	0.8
CrashLoc	0.1
CrashType	0.8
DrvrAge	15.3
DrvrAgeGrp	13.7
DrvrAlcDrg	35.2
DrvrAlcFlg	14.4
DrvrRace	13.9
DrvrSex	13.6
LightCond	0.1
NumBicsAin	54.9
NumBicsBin	54.9
NumBicsCin	54.9
NumBicsKil	54.9
NumBicsNoi	54.9
NumBicsTot	54.9
NumBicsUin	54.9

Variable	% missing
NumLanes	5.8
RdCharacte	0.2
RdClass	0.7
RdConditio	0.3
RdConfig	1.2
RdDefects	0.4
RdFeature	4.4
RdSurface	0.5
SpeedLimit	4.7
TraffCntrl	1.2

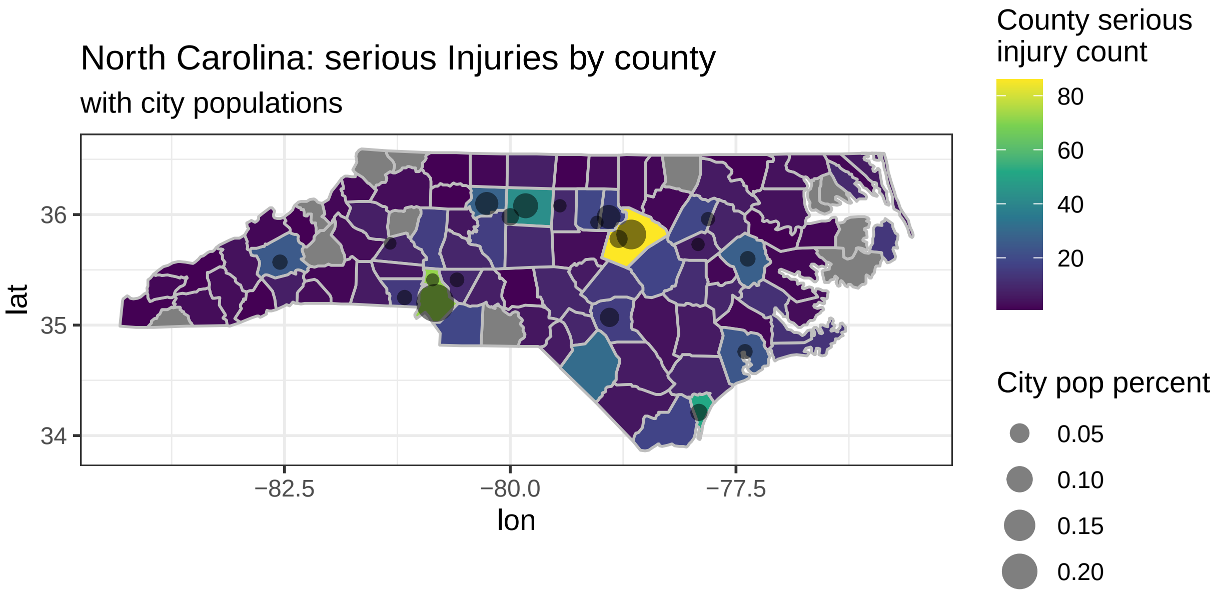
4.5 Imputations

For numerical missing data such as age, missing values were imputed based on medians of age groups if age group variable is available, otherwise age was imputed using the median age of the full dataset based on the variable in question. For two-level categorical variables (“Yes”/“No”), missing values were imputed with a “No”, indicating negative detection to help in achieving as complete a dataset as possible.

5 Exploratory Data Analysis

5.1 Response Variable: Serious Bike Injury

Let’s take a look at a geographical map of the serious crashes in the data set. The geographical data is sourced from R’s maps package. As it can be seen from the map below, a majority of the crashes seem to be occurring in counties where large cities are located.



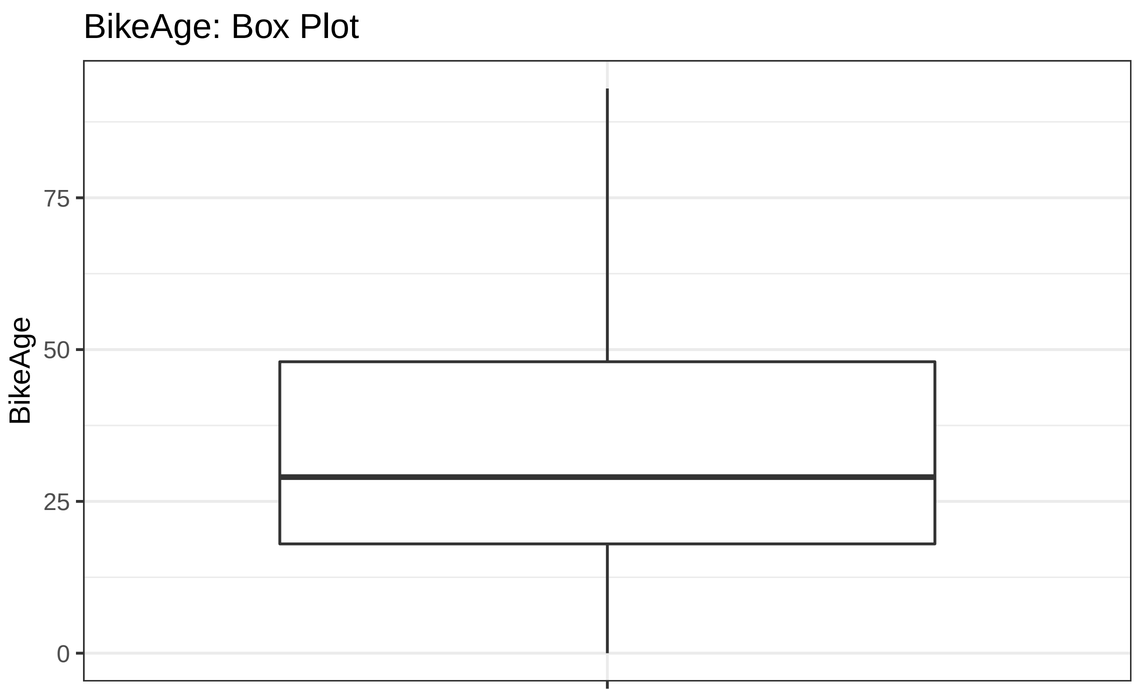
The table below confirms that around 50% of the bike accidents with serious injury occur in urban areas (>70% developed), followed by 32.5% of serious injuries being in rural areas (<30% developed). There does not seem to be a linear relationship between development-level and severity of injuries. Instead, we see more serious injuries in highly developed areas and more rural areas and not as much in between.

Table 7: Bike injury by locality

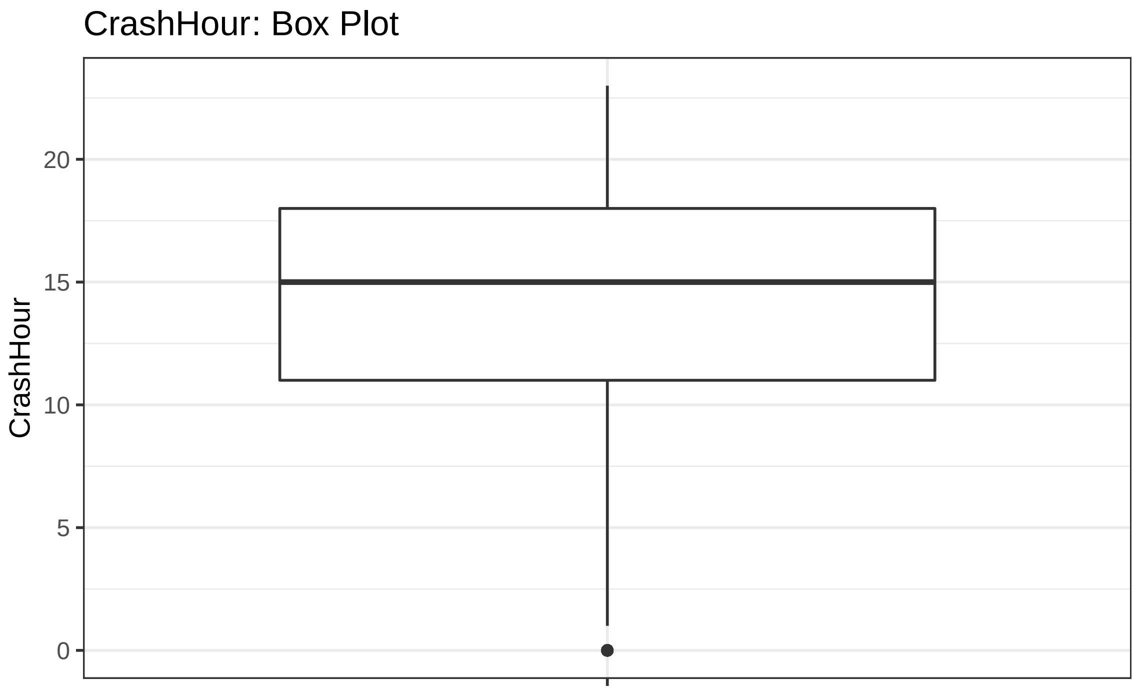
Locality	BikeInjurySerious	n	pct
Mixed (30% To 70% Developed)	No	1484	12.58
Mixed (30% To 70% Developed)	Yes	169	1.43
Rural (<30% Developed)	No	1432	12.14
Rural (<30% Developed)	Yes	294	2.49
Urban (>70% Developed)	No	7976	67.60
Urban (>70% Developed)	Yes	443	3.75

5.2 Explore numerical variables

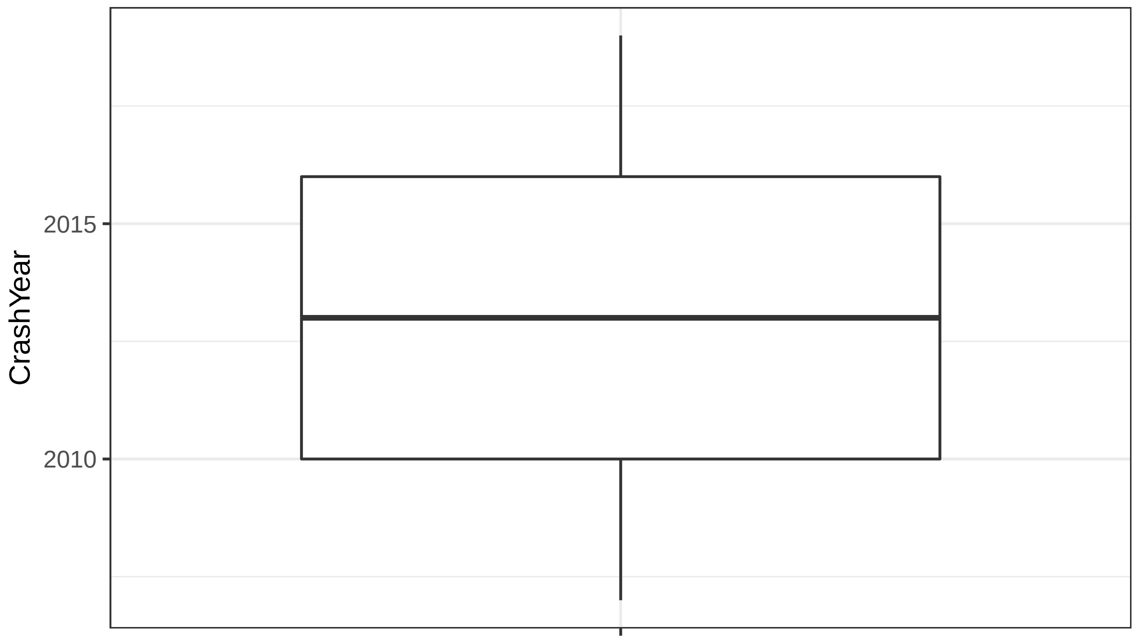
NumUnits has very little variability and will likely be removed due to near-zero variance.



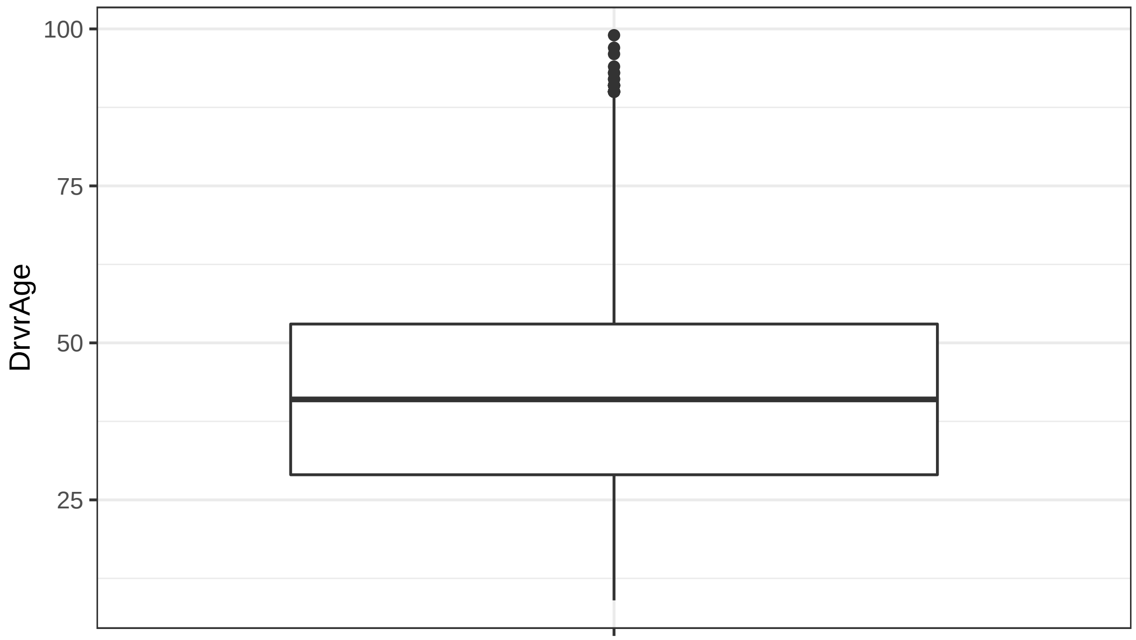
BikeAge: Box Plot



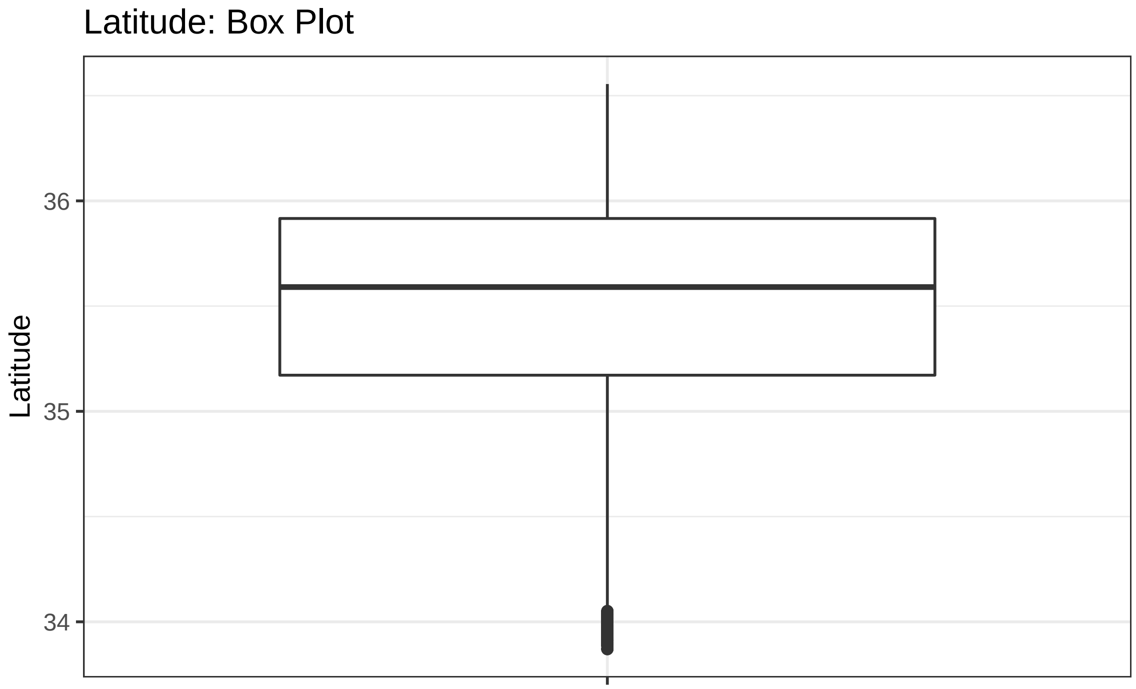
CrashHour: Box Plot

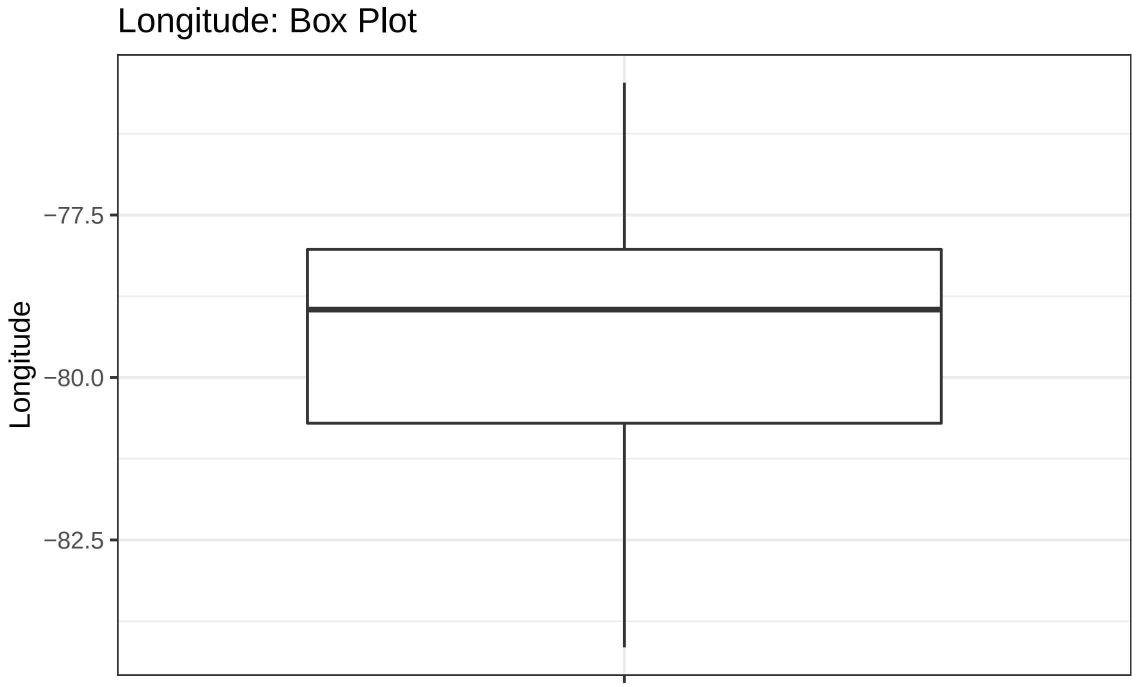


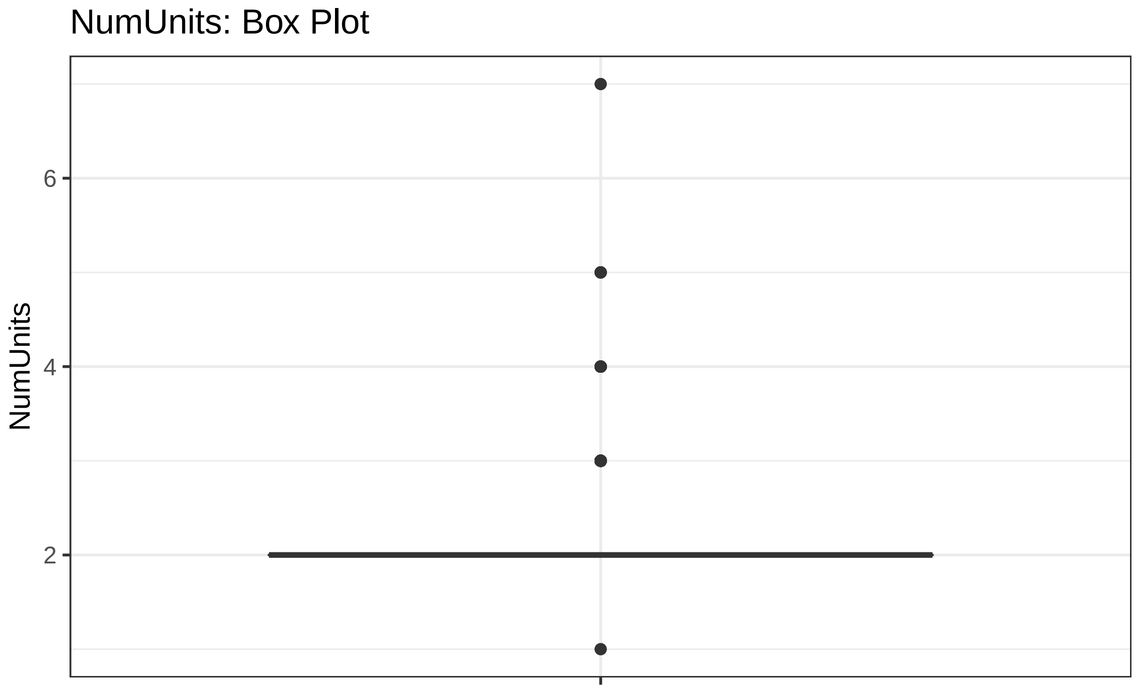
CrashYear: Box Plot



DrvrAge: Box Plot


Latitude: Box Plot
Latitude
36
35
34


Longitude: Box Plot
Longitude
−77.5
−80.0
−82.5



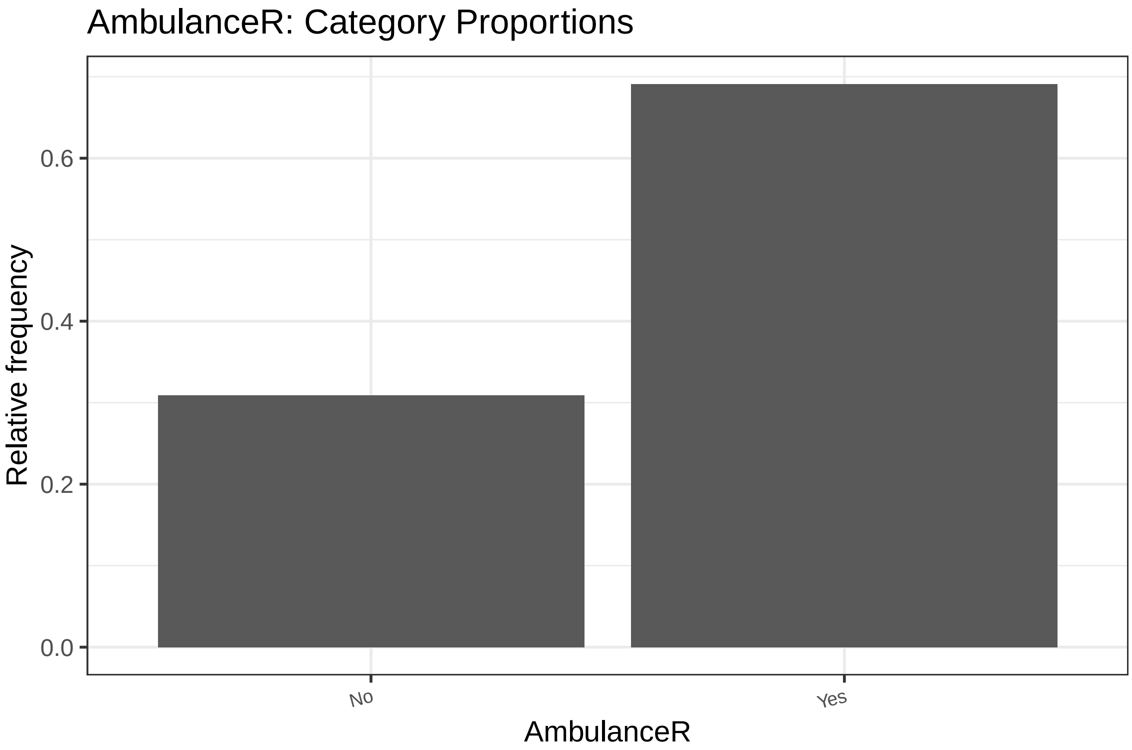
5.3 Explore categorical variables

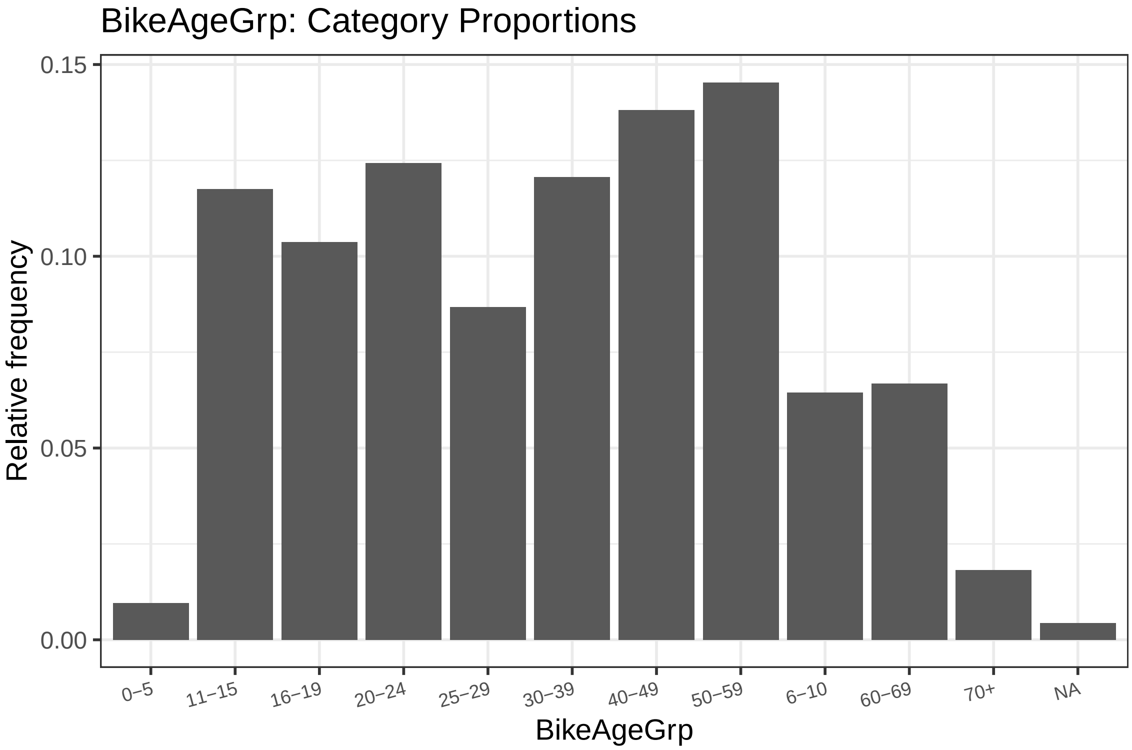
Bar plots and top-20 summaries have been produced below for all categorical features that will be attempted to be included in the modeling process.

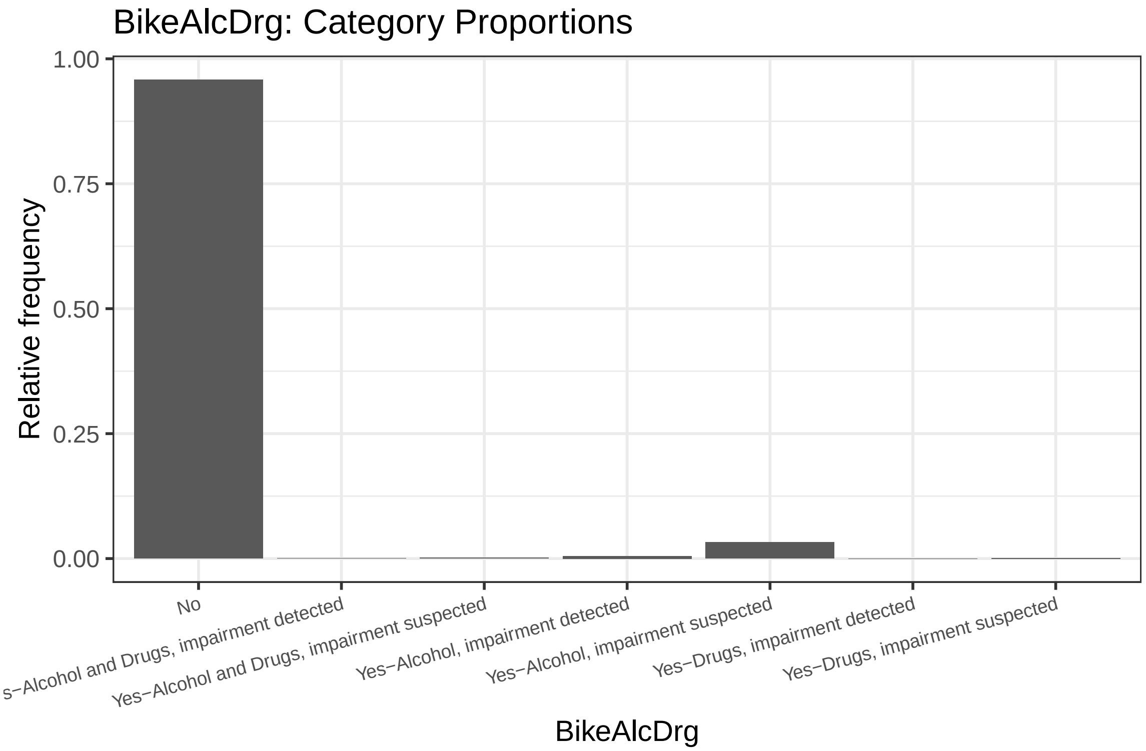
A number of categorical features have unbalanced levels. Once factors are converted to dummy variables, near-zero variance feature will be removed, thus helping by removing facors/levels that don't provide enough variability to inform the model. A percent unique values cutoff of 5% will be used to determine which feature levels are removed.

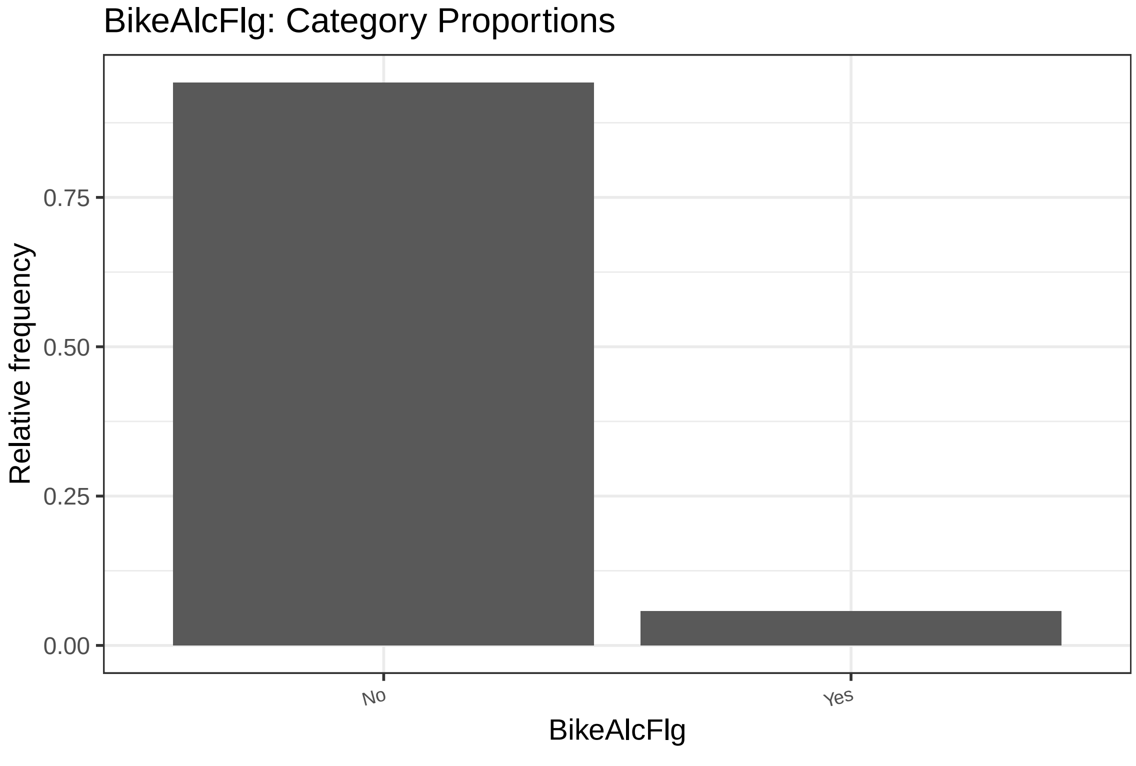
Notice that some features have too many levels to be viewed in a bar graph.
Therefore, those features' levels are displayed in a table of the top 20 levels by feature.

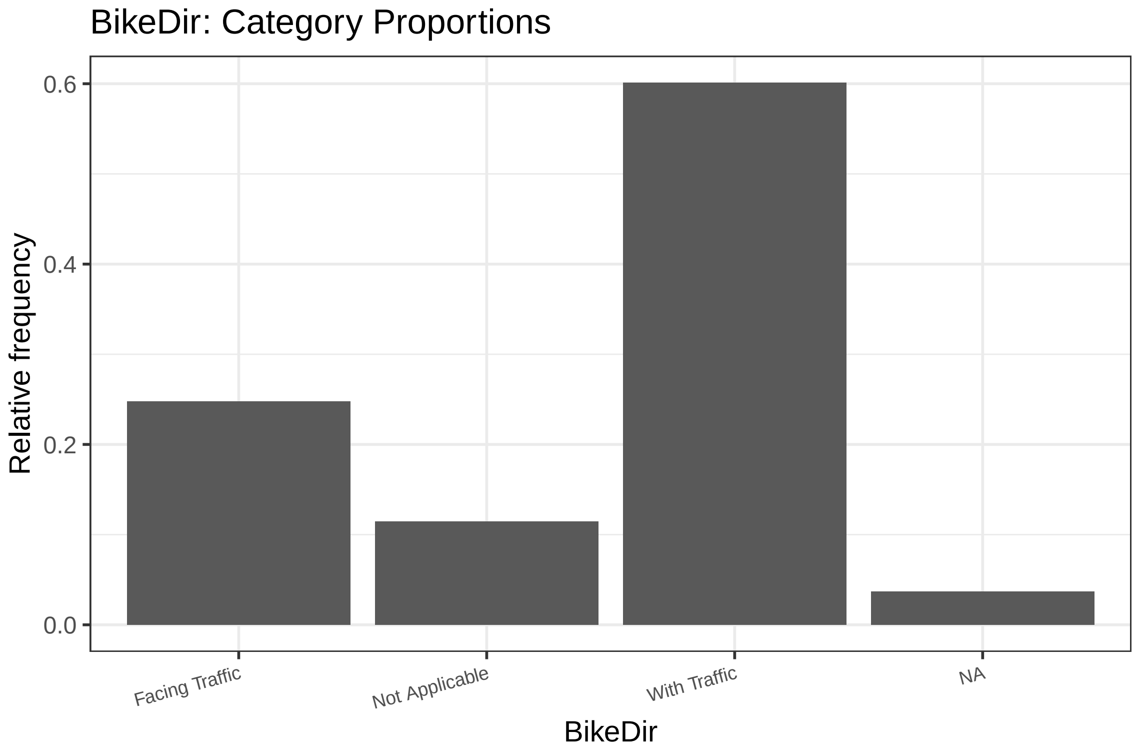
The following features CrashID, BikeInjuryCat will not be summarized as these features will not be used in the modeling process. All features of the form "NumBics"X will not be since these features are missing more than 50% of values.

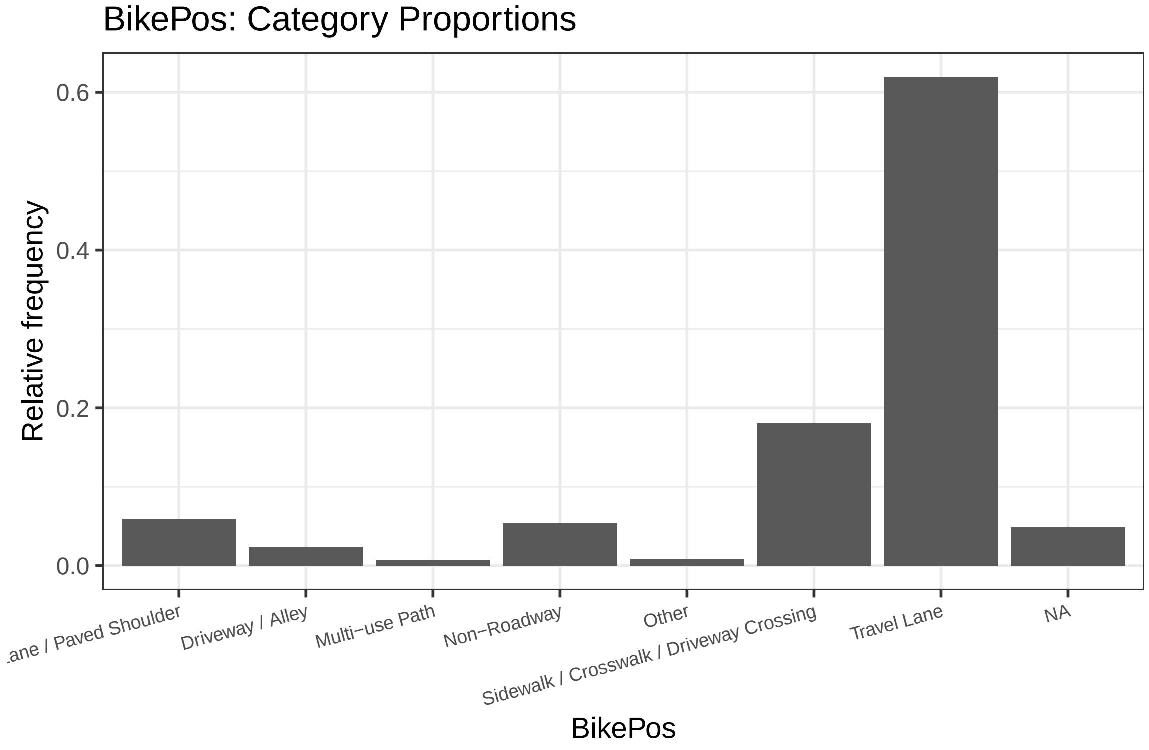

AmbulanceR: Category Proportions
Relative frequency
0.6
0.4
0.2
0.0
No
Yes
AmbulanceR

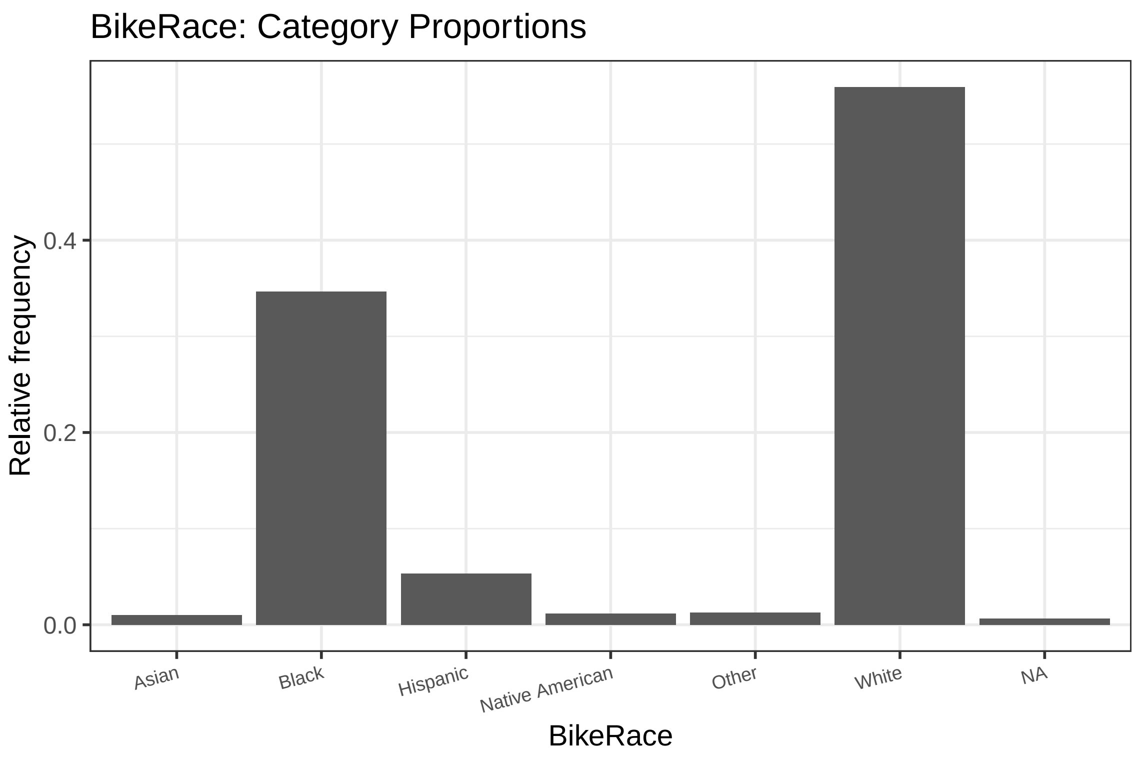

BikeAgeGrp: Category Proportions
Relative frequency
0.15
0.10
0.05
0.00
0–5
11–15
16–19
20–24
25–29
30–39
40–49
50–59
6–10
60–69
70+
NA
BikeAgeGrp

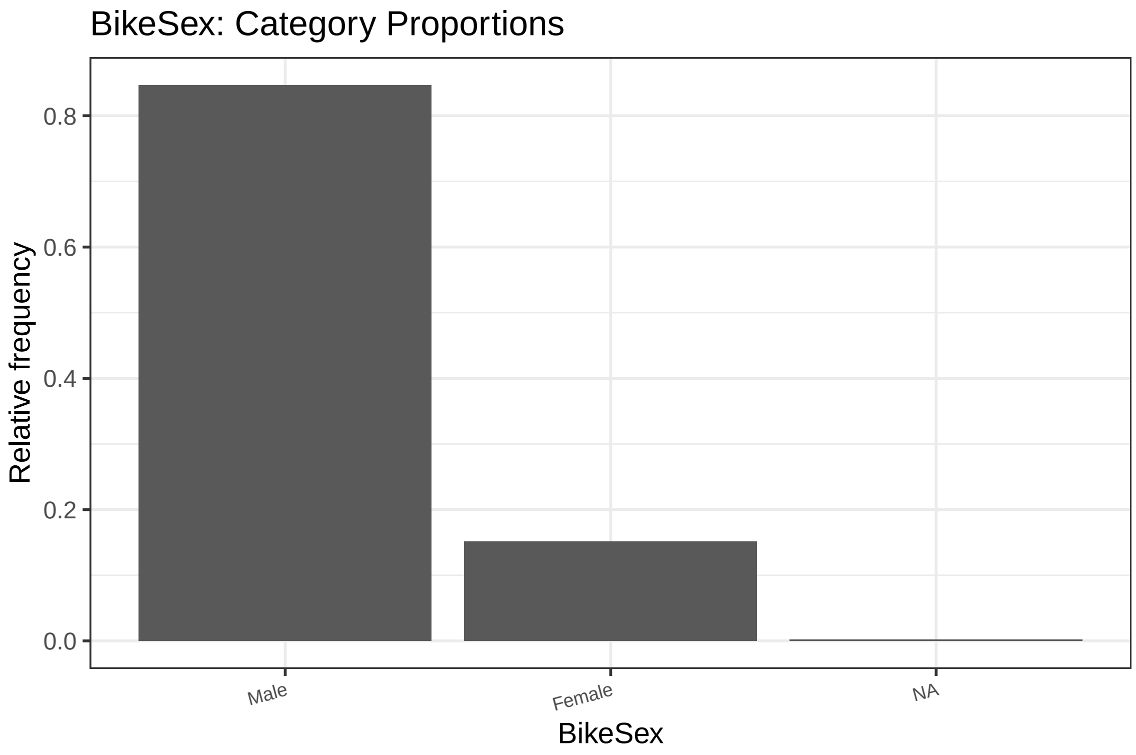

BikeAlcDrg: Category Proportions
Relative frequency
1.00
0.75
0.50
0.25
0.00
No
s−Alcohol and Drugs, impairment detected
Yes−Alcohol and Drugs, impairment suspected
Yes−Alcohol, impairment detected
Yes−Alcohol, impairment suspected
Yes−Drugs, impairment detected
Yes−Drugs, impairment suspected
BikeAlcDrg

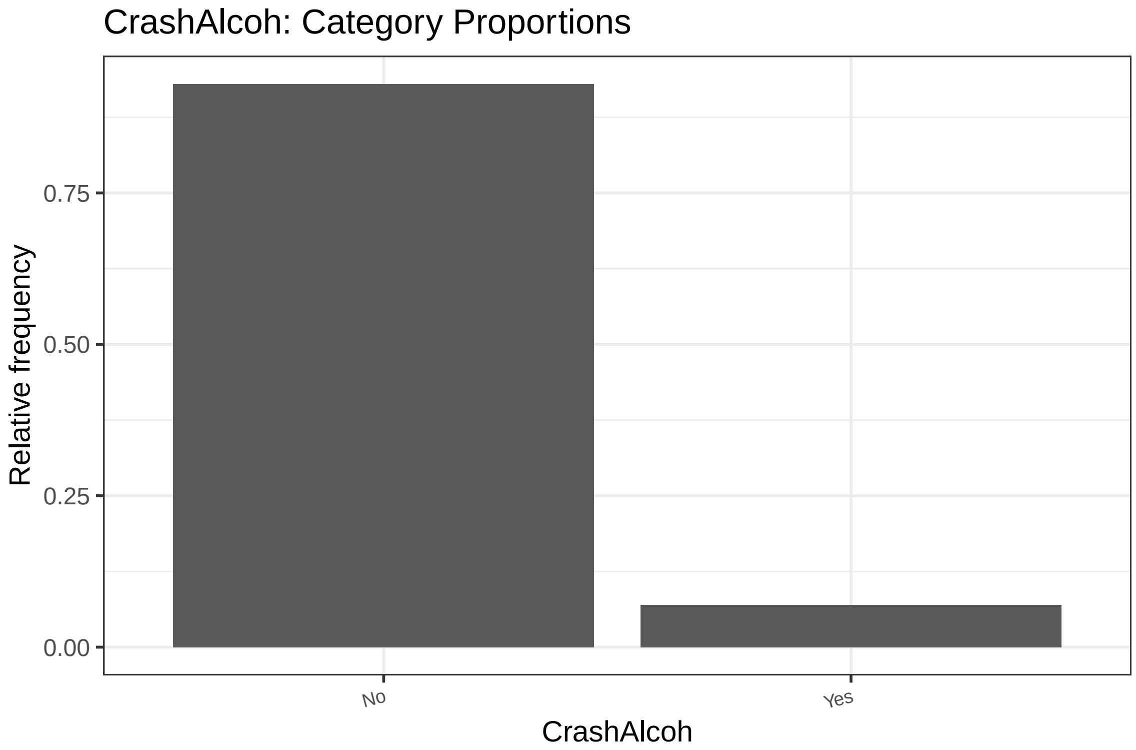

BikeAlcFlg: Category Proportions
Relative frequency
0.75
0.50
0.25
0.00
No
Yes
BikeAlcFlg

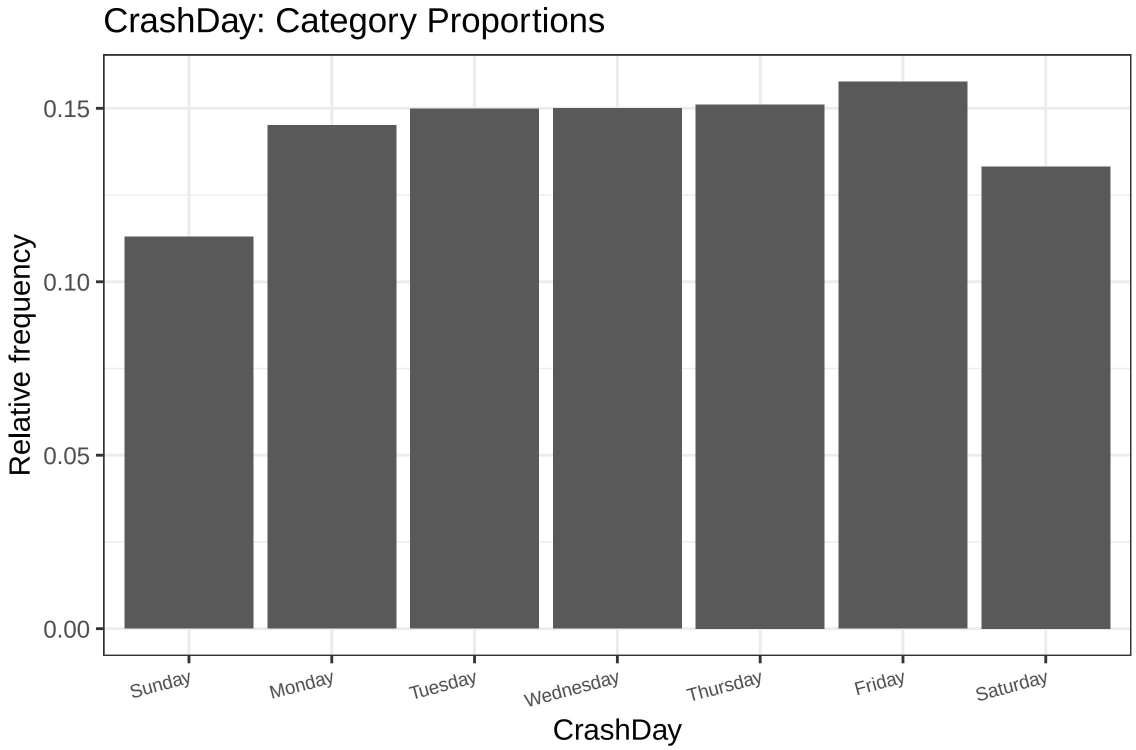

BikeDir: Category Proportions
Relative frequency
0.6
0.4
0.2
0.0
Facing Traffic
Not Applicable
With Traffic
NA
BikeDir

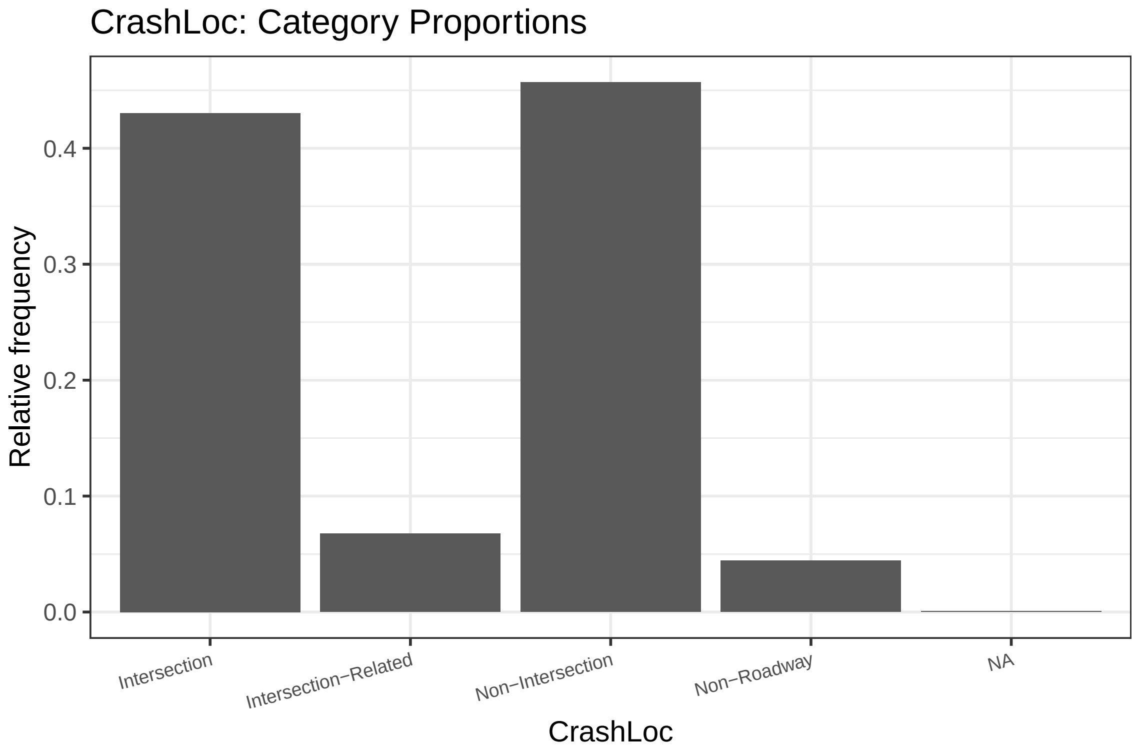

BikePos: Category Proportions
Relative frequency
0.6
0.4
0.2
0.0
.ane / Paved Shoulder
Driveway / Alley
Multi−use Path
Non−Roadway
Other
Sidewalk / Crosswalk / Driveway Crossing
Travel Lane
NA
BikePos

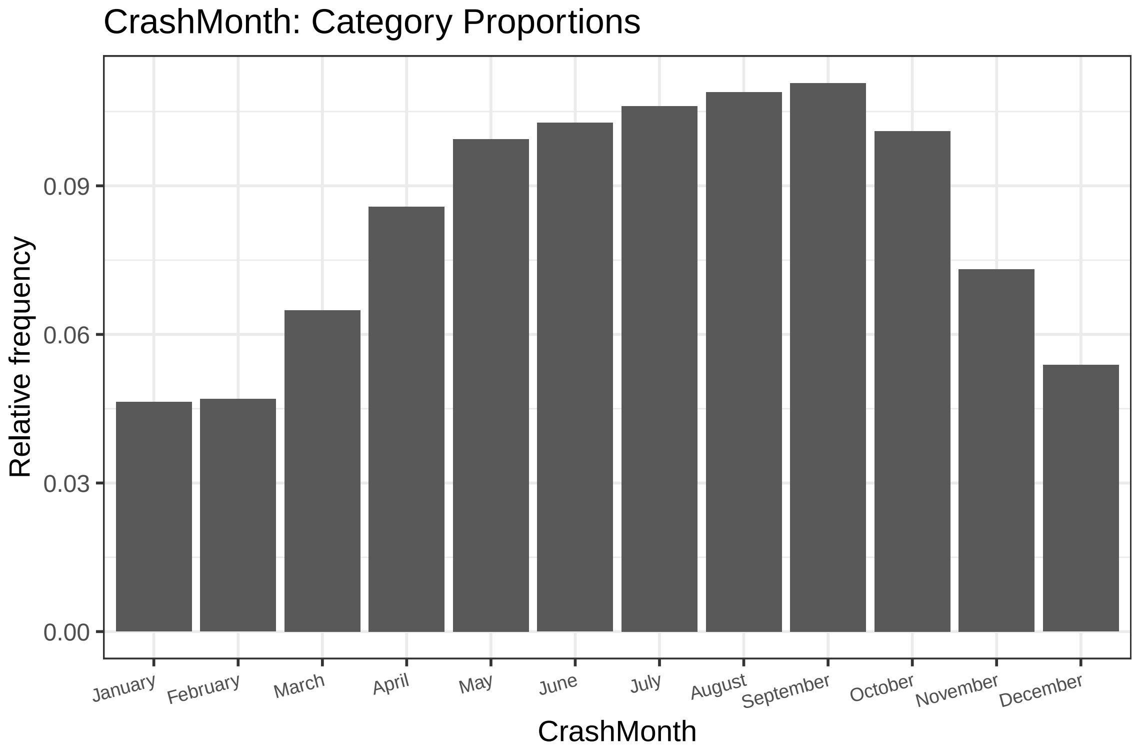

BikeRace: Category Proportions
Relative frequency
0.0
0.2
0.4
Asian
Black
Hispanic
Native American
Other
White
NA
BikeRace

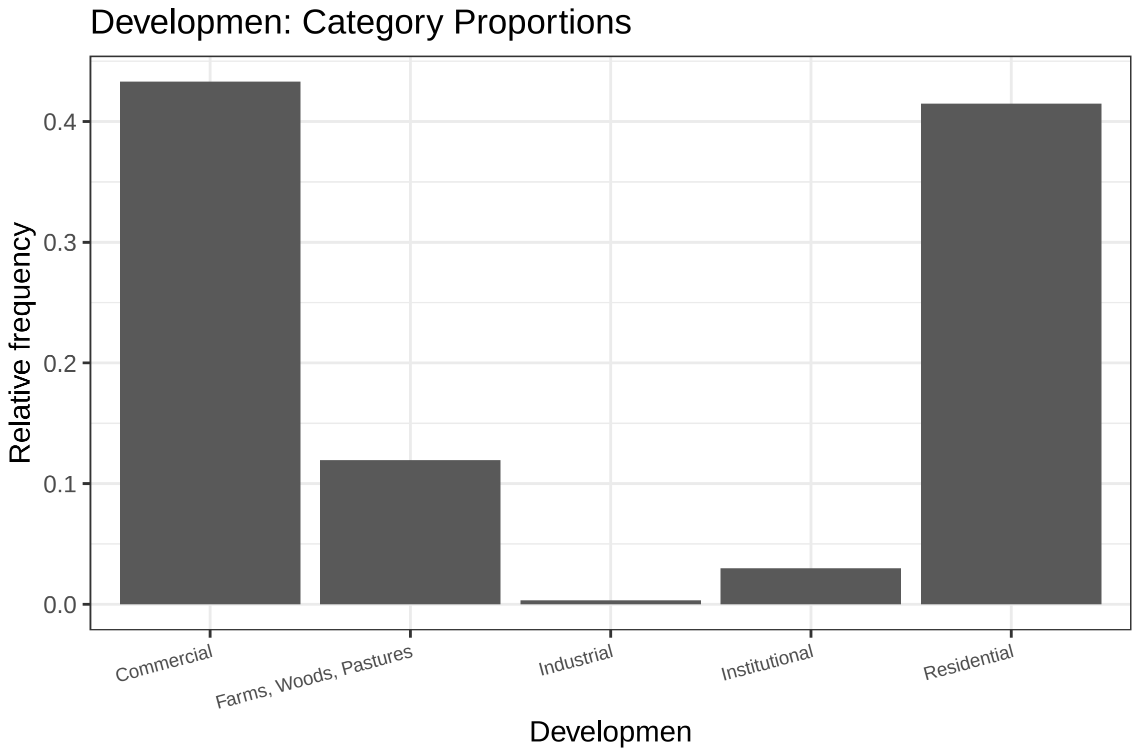

BikeSex: Category Proportions
Relative frequency
0.0
0.2
0.4
0.6
0.8
Male
Female
NA
BikeSex

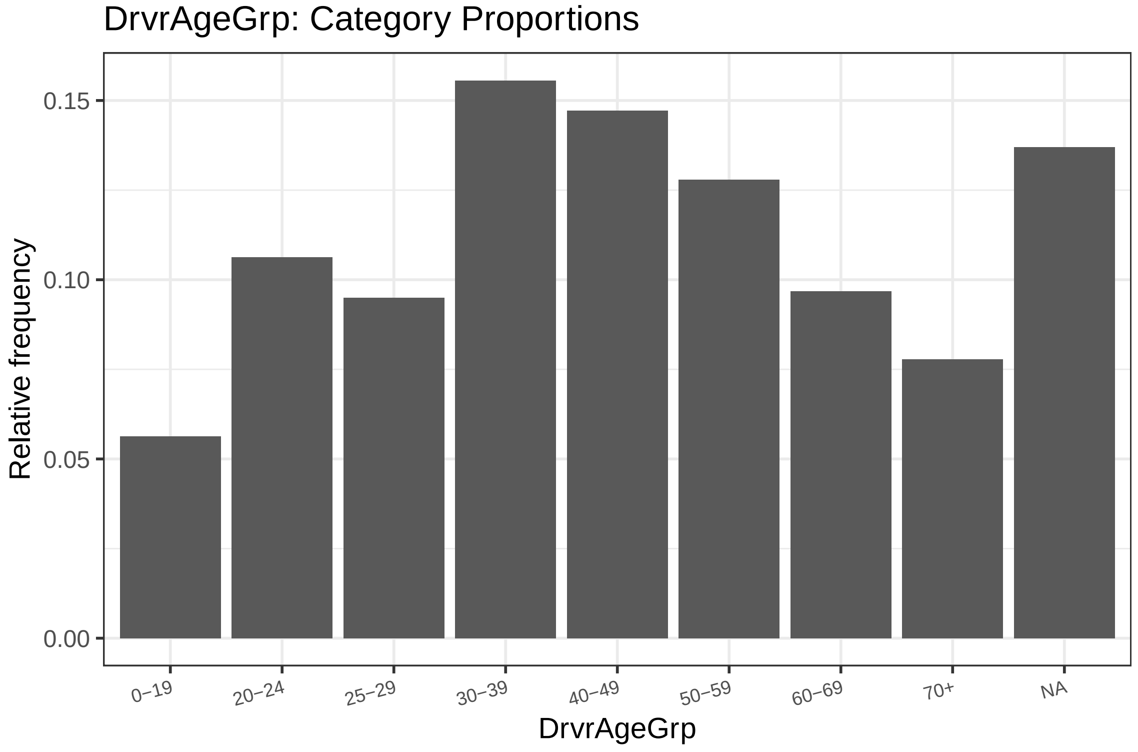

CrashAlcoh: Category Proportions

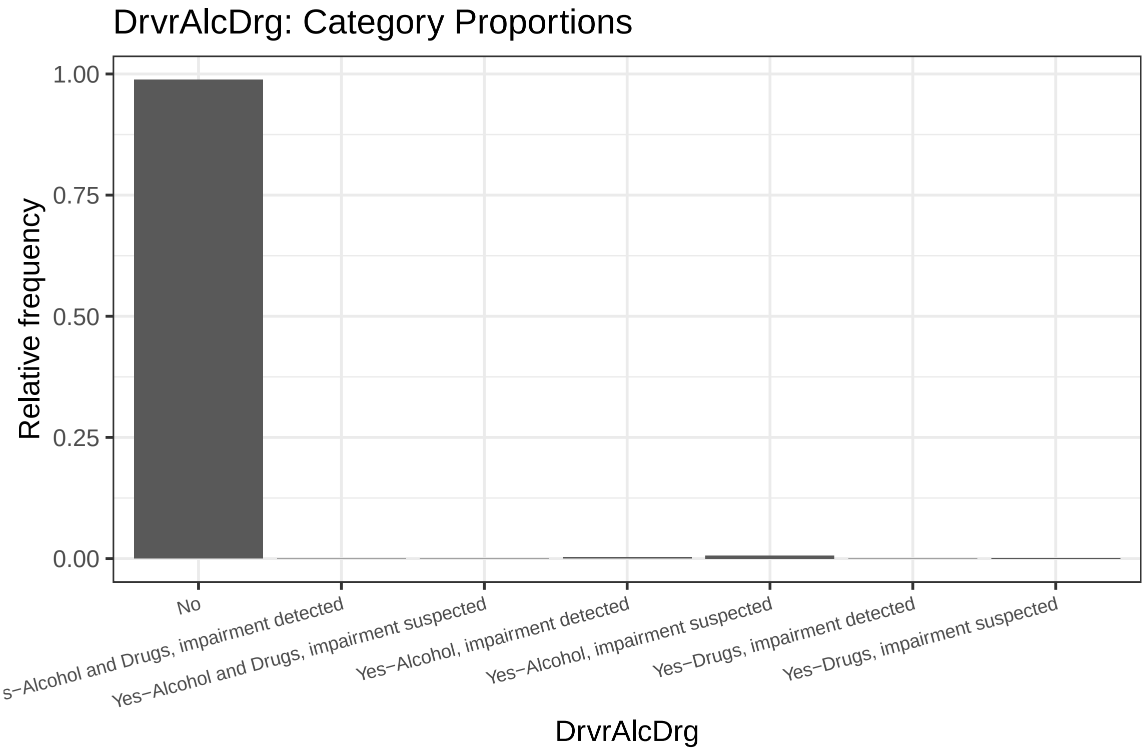

CrashDay: Category Proportions

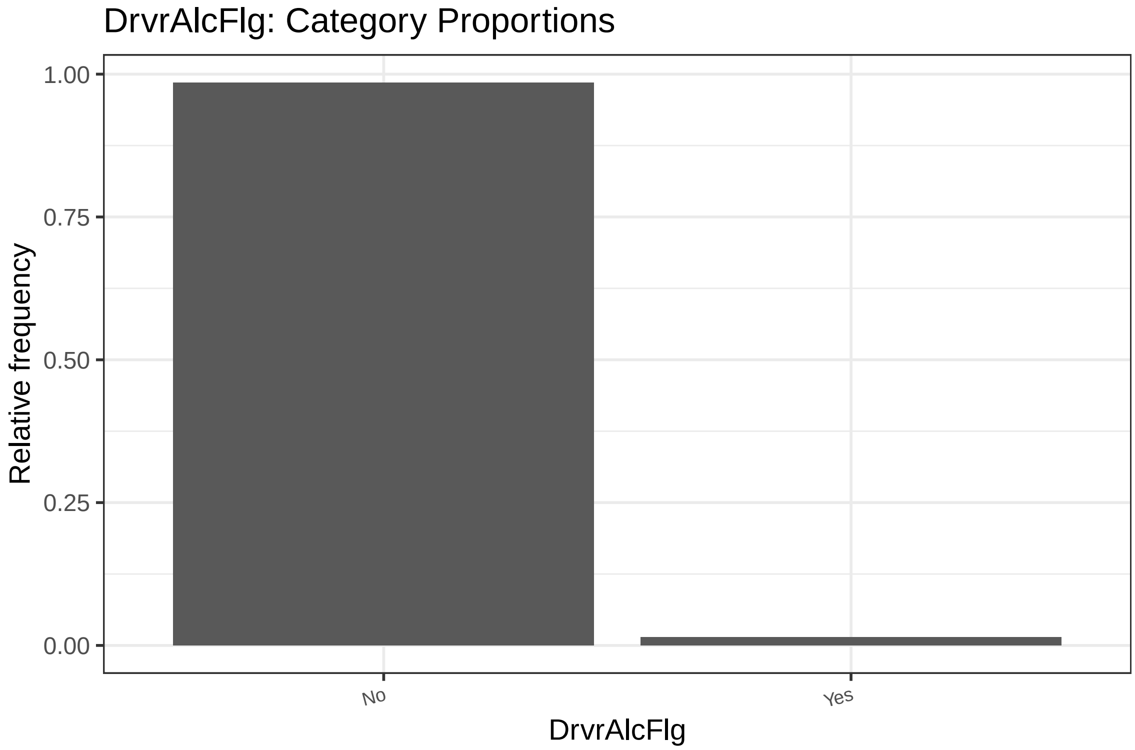

CrashLoc: Category Proportions
Relative frequency
0.4
0.3
0.2
0.1
0.0
Intersection
Intersection−Related
Non−Intersection
Non−Roadway
NA
CrashLoc

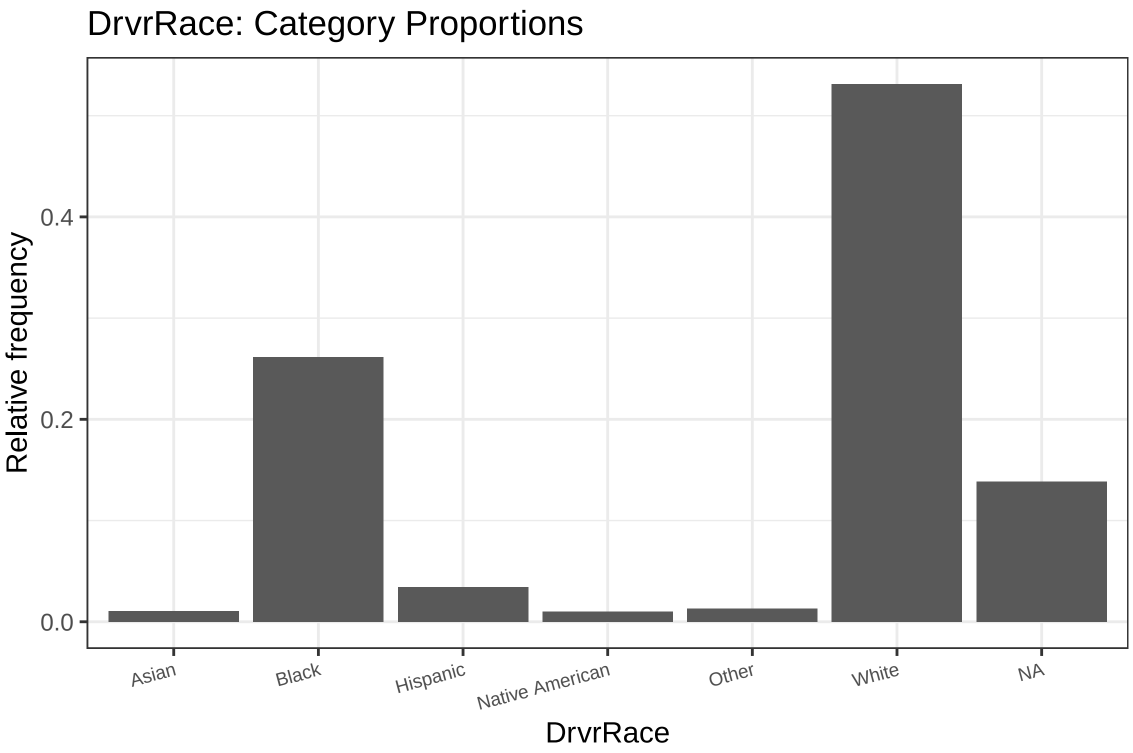

CrashMonth: Category Proportions
Relative frequency
0.09
0.06
0.03
0.00
January
February
March
April
May
June
July
August
September
October
November
December
CrashMonth

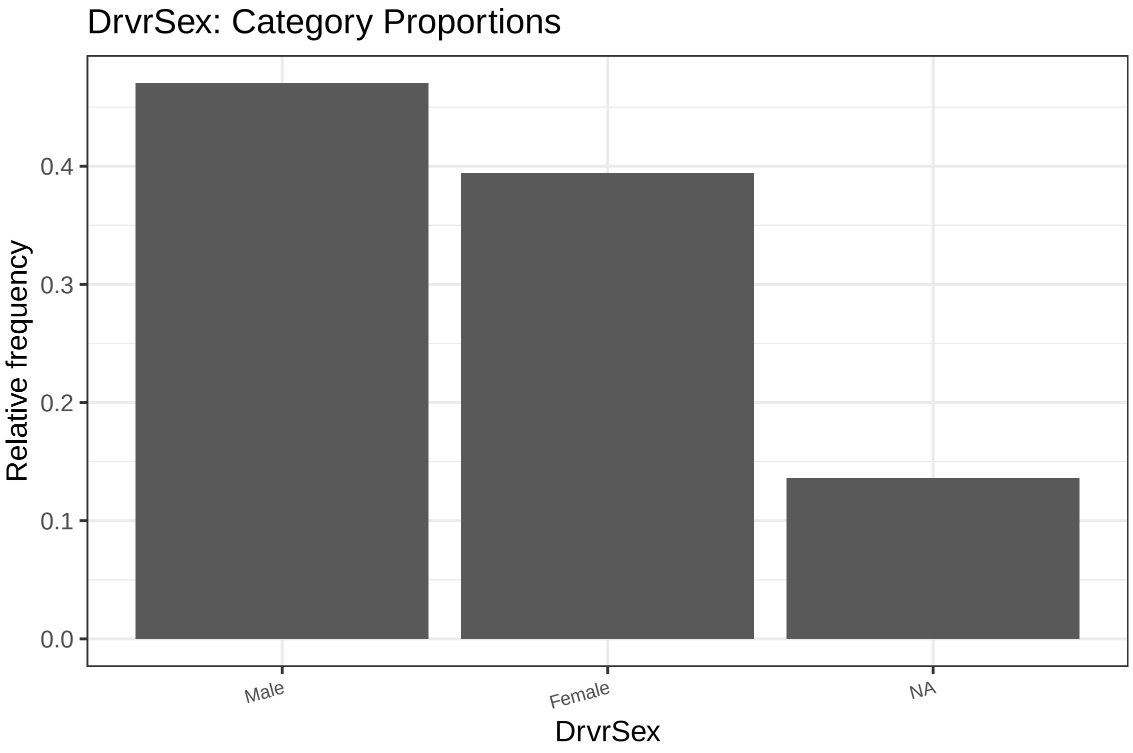

Developmen: Category Proportions
Relative frequency
0.4
0.3
0.2
0.1
0.0
Commercial
Farms, Woods, Pastures
Industrial
Institutional
Residential
Developmen

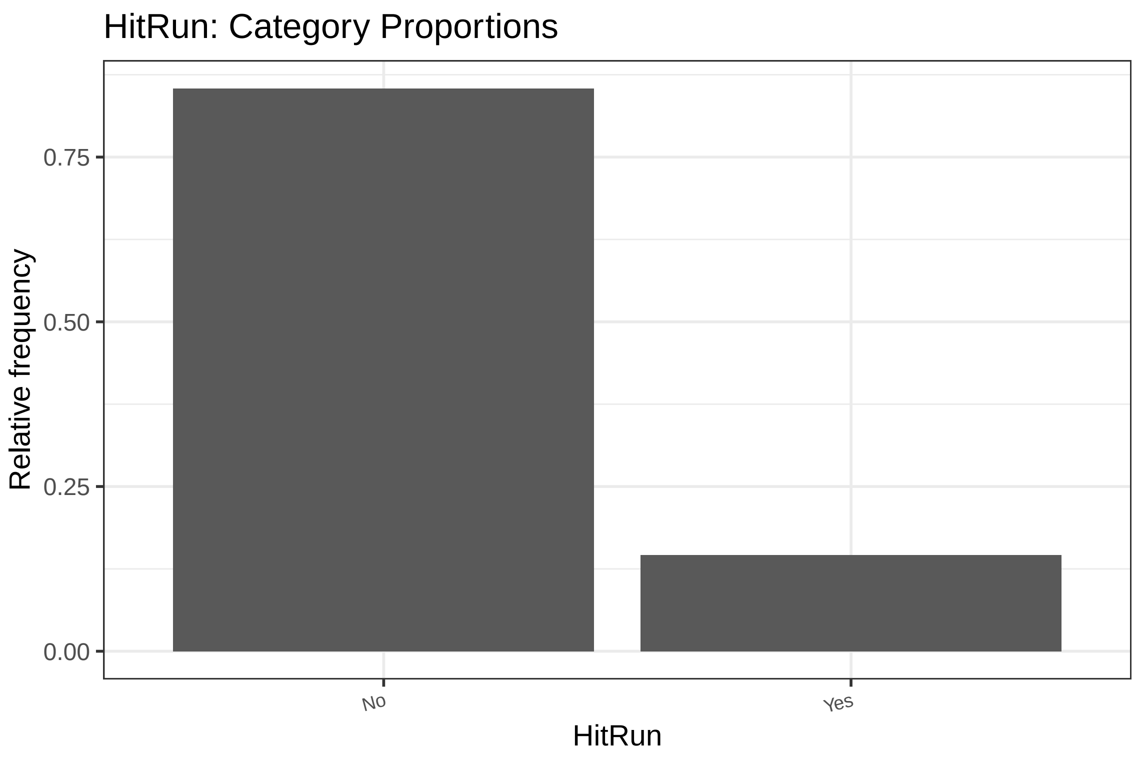

DrvrAgeGrp: Category Proportions
Relative frequency
0.15
0.10
0.05
0.00
0–19
20–24
25–29
30–39
40–49
50–59
60–69
70+
NA
DrvrAgeGrp

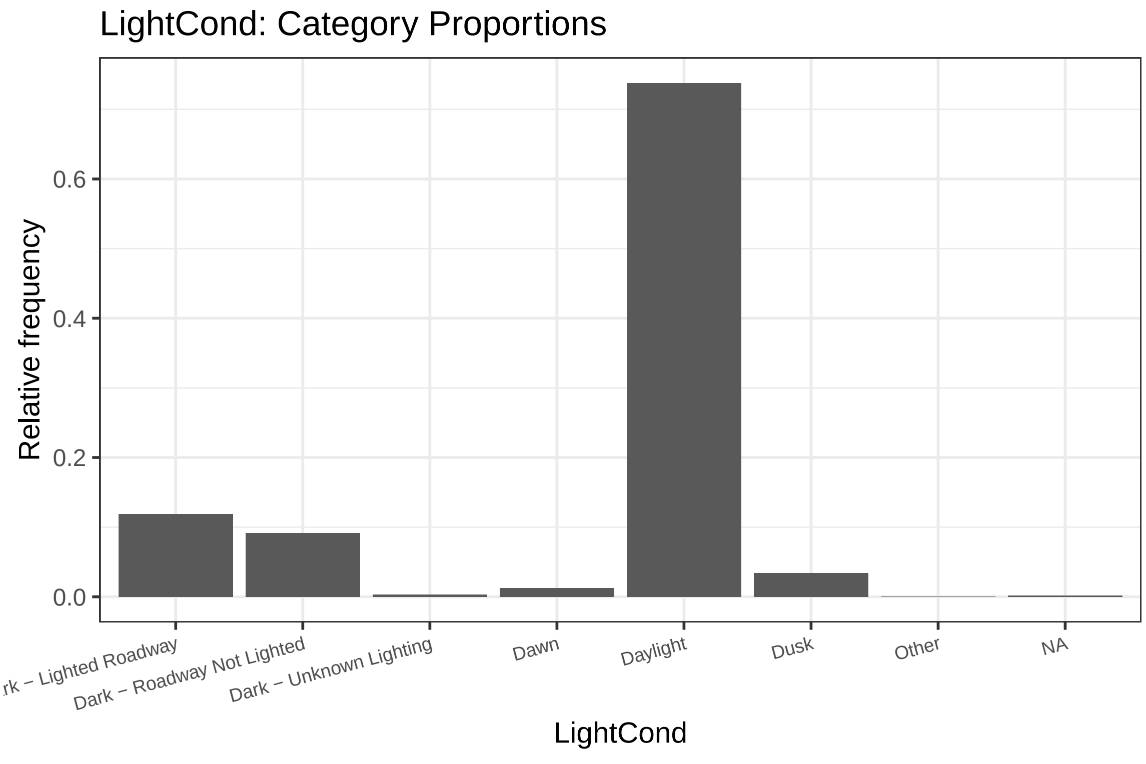

DrvrAlcDrg: Category Proportions

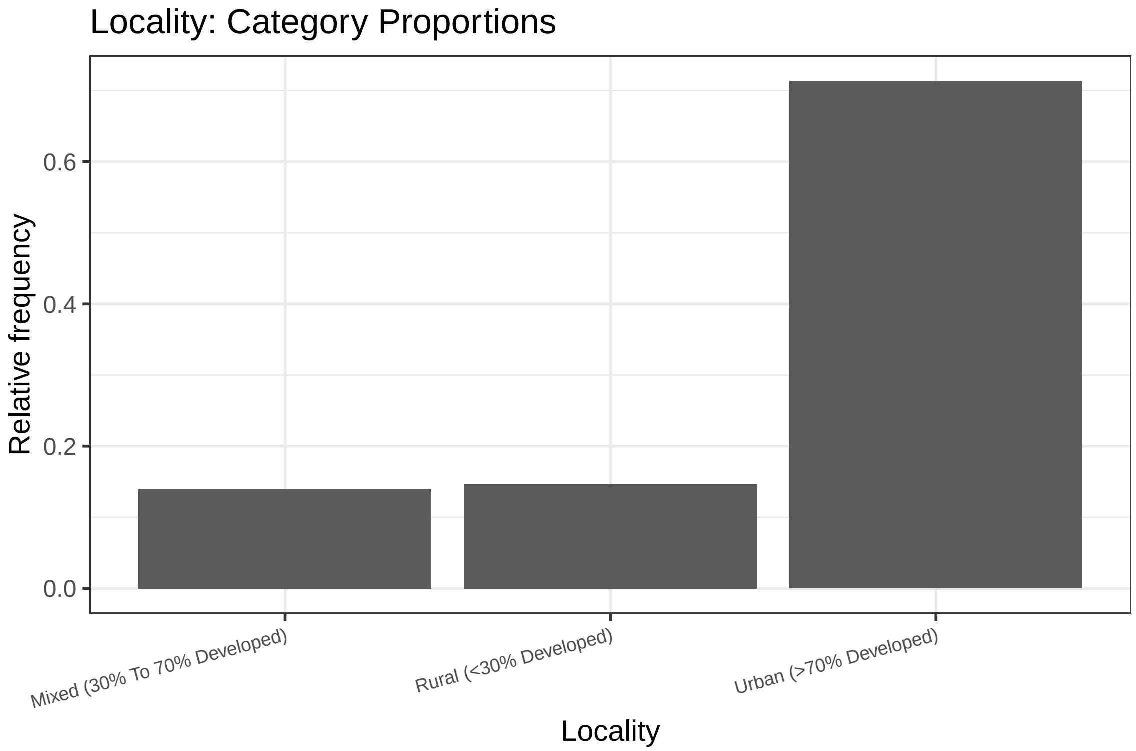

DrvrAlcFlg: Category Proportions

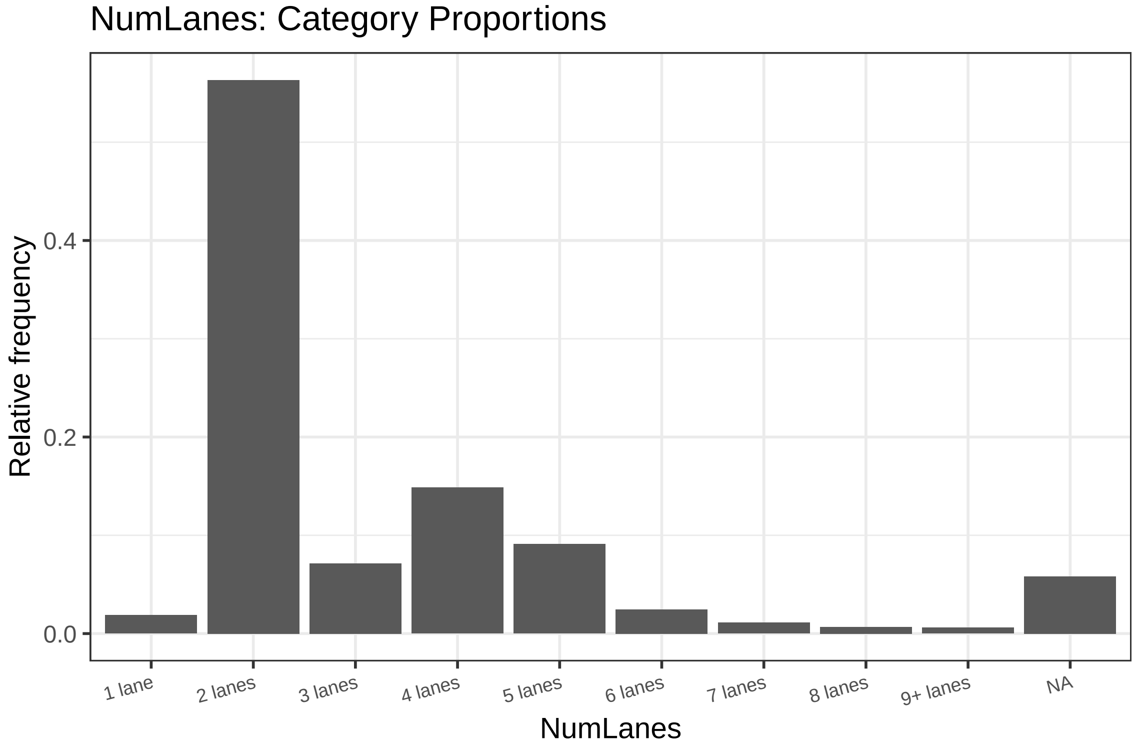

DrvrRace: Category Proportions
Relative frequency
0.0
0.2
0.4
Asian
Black
Hispanic
Native American
Other
White
NA
DrvrRace

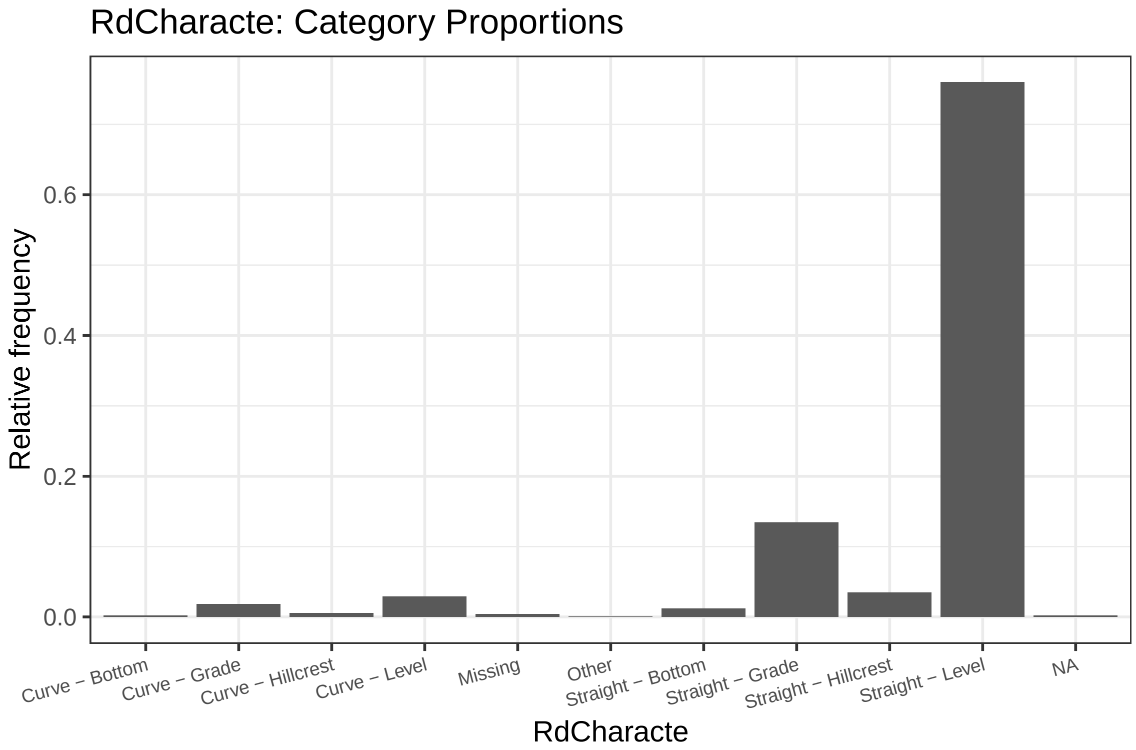

DrvrSex: Category Proportions
Relative frequency
0.0
0.1
0.2
0.3
0.4
Male
Female
NA
DrvrSex

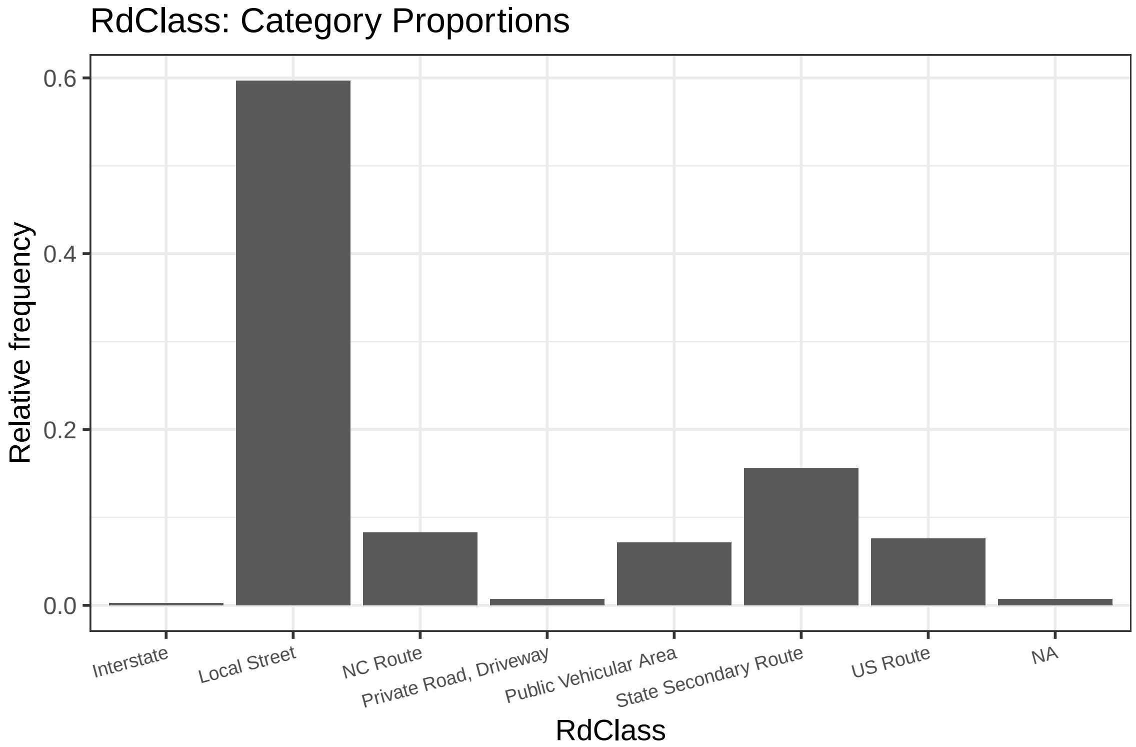

HitRun: Category Proportions
Relative frequency
0.75
0.50
0.25
0.00
No
Yes
HitRun

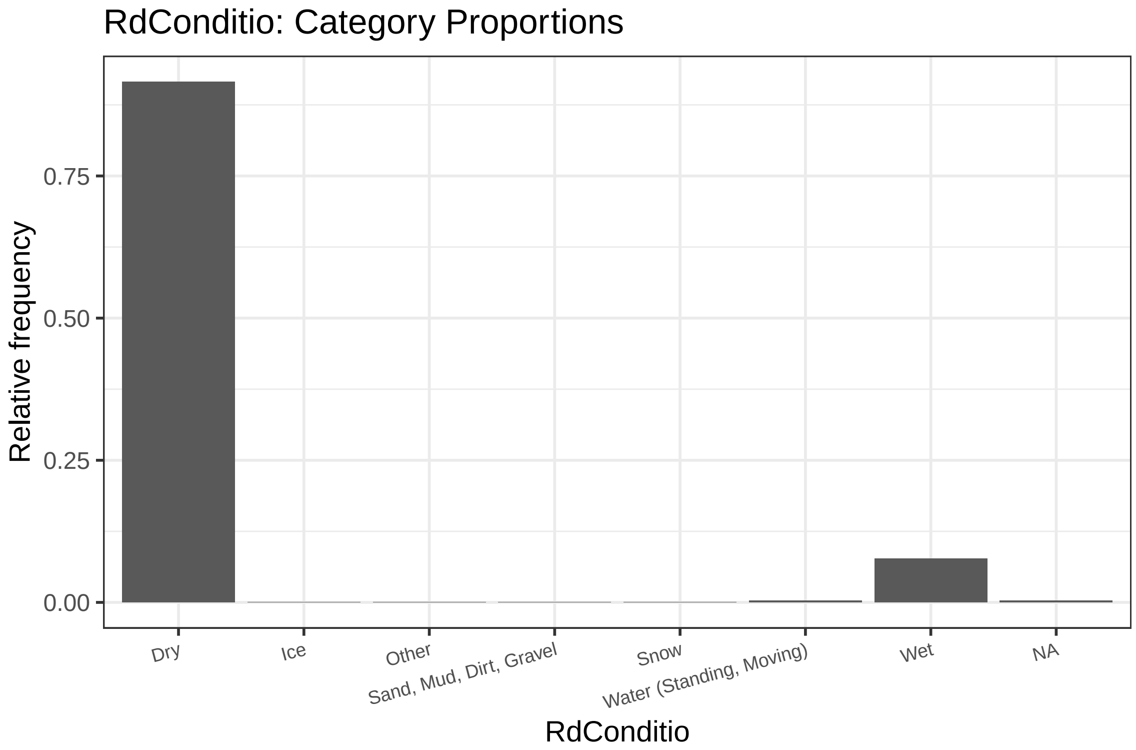

LightCond: Category Proportions
Relative frequency
0.6
0.4
0.2
0.0
rk – Lighted Roadway
Dark – Roadway Not Lighted
Dark – Unknown Lighting
Dawn
Daylight
Dusk
Other
NA
LightCond

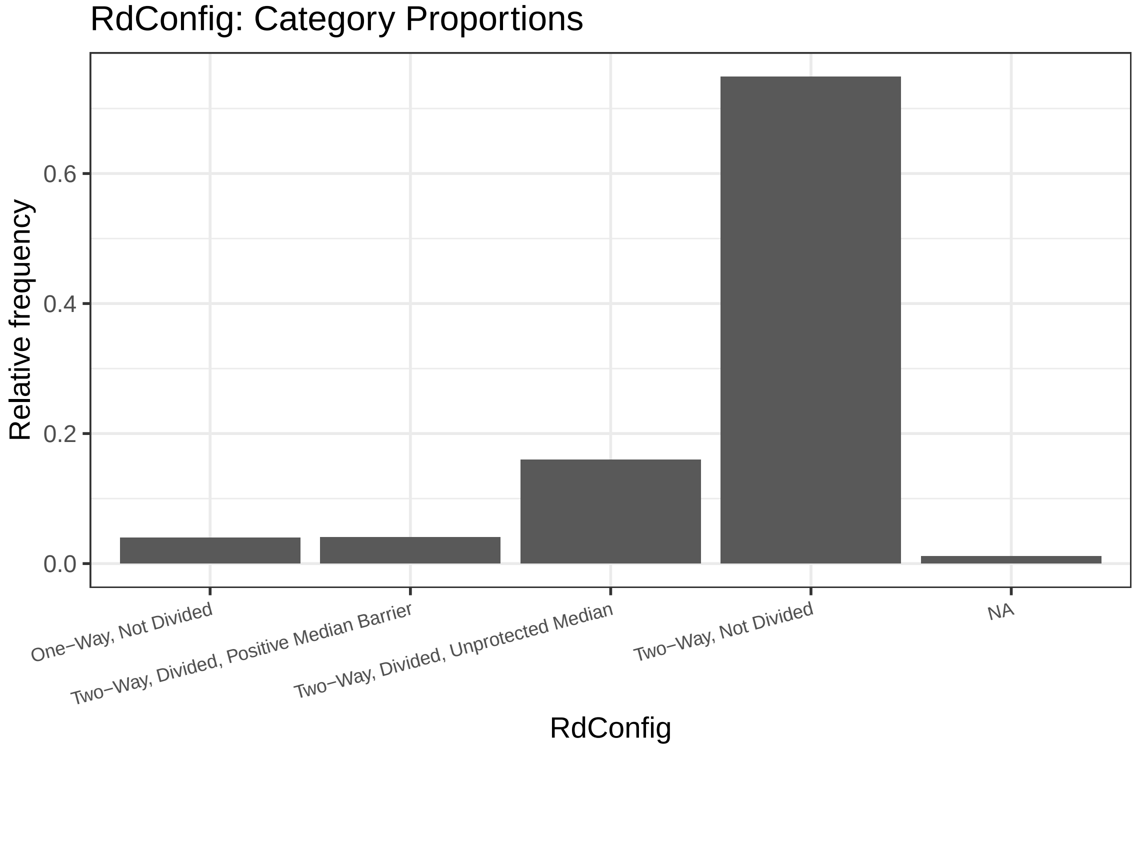

Locality: Category Proportions
Relative frequency
0.6
0.4
0.2
0.0
Mixed (30% To 70% Developed)
Rural (<30% Developed)
Urban (>70% Developed)
Locality

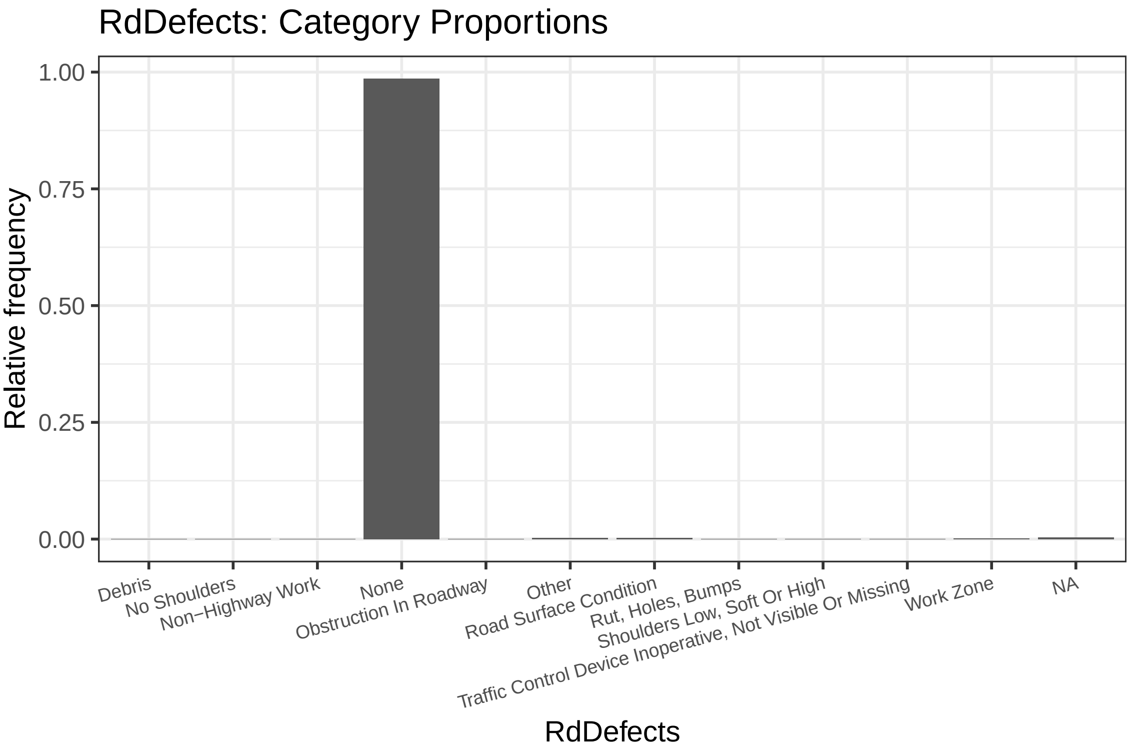

NumLanes: Category Proportions
Relative frequency
0.4
0.2
0.0
1 lane
2 lanes
3 lanes
4 lanes
5 lanes
6 lanes
7 lanes
8 lanes
9+ lanes
NA
NumLanes

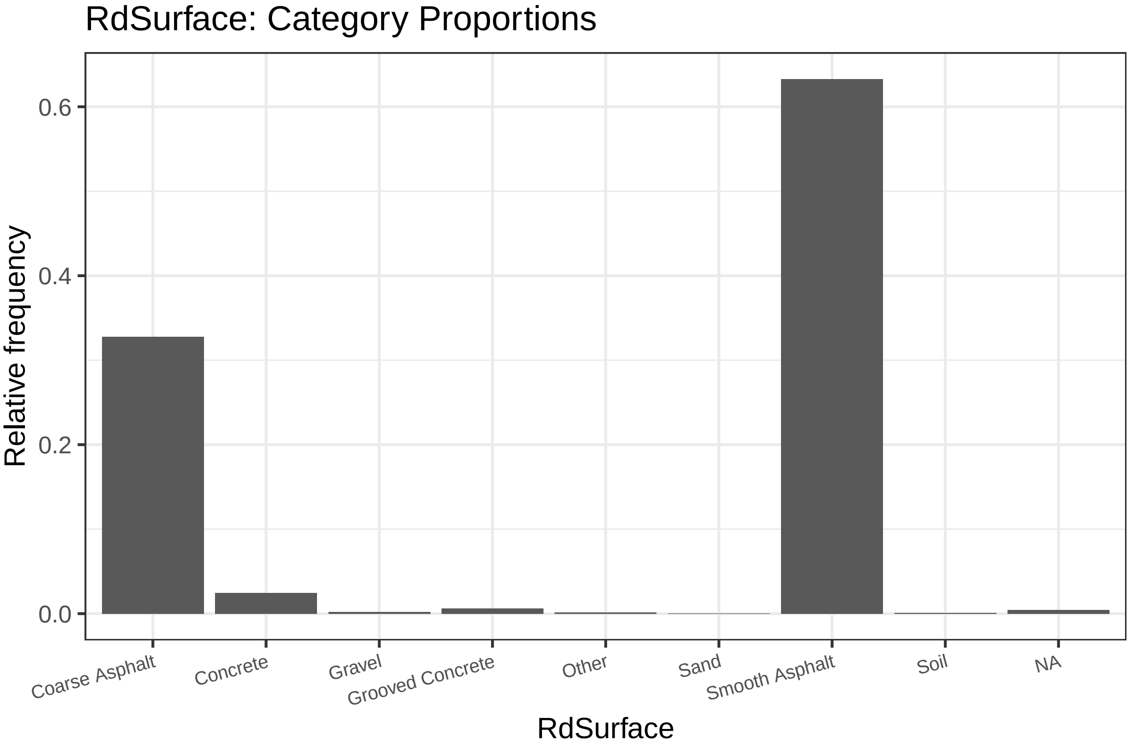

RdCharacte: Category Proportions
Relative frequency
0.0
0.2
0.4
0.6
Curve – Bottom
Curve – Grade
Curve – Hillcrest
Curve – Level
Missing
Other
Straight – Bottom
Straight – Grade
Straight – Hillcrest
Straight – Level
NA
RdCharacte

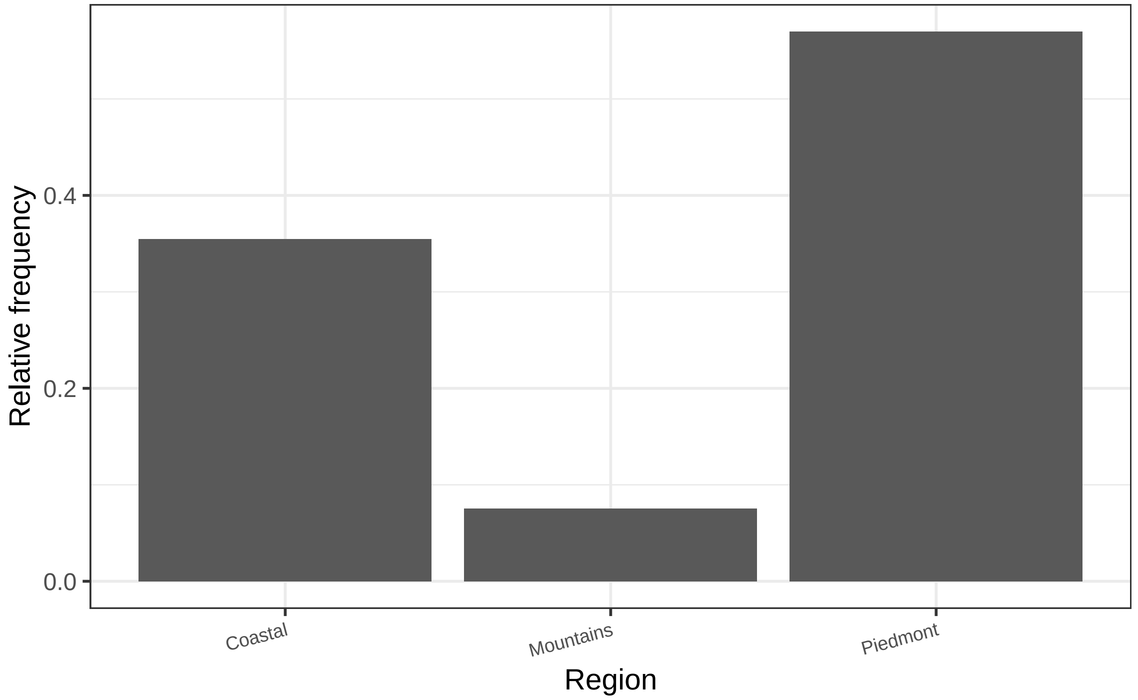

RdClass: Category Proportions
Relative frequency
0.0
0.2
0.4
0.6
Interstate
Local Street
NC Route
Private Road, Driveway
Public Vehicular Area
State Secondary Route
US Route
NA
RdClass


RdConditio: Category Proportions
Relative frequency
0.75
0.50
0.25
0.00
Dry
Ice
Other
Sand, Mud, Dirt, Gravel
Snow
Water (Standing, Moving)
Wet
NA
RdConditio

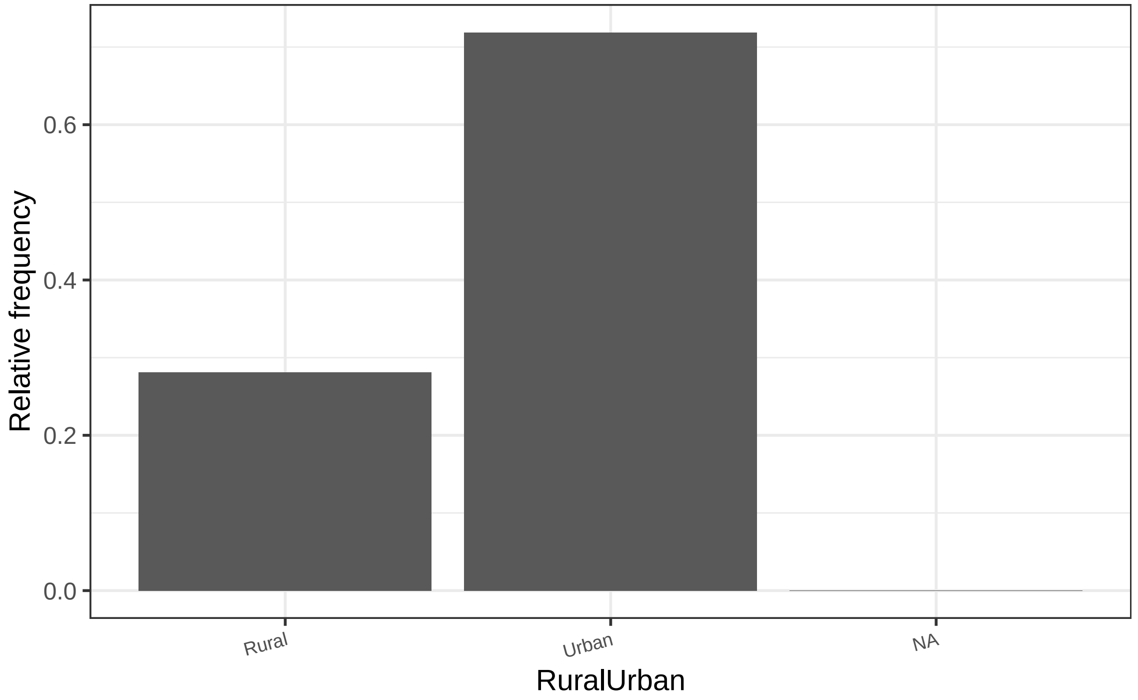

RdConfig: Category Proportions
Relative frequency
0.6
0.4
0.2
0.0
One–Way, Not Divided
Two–Way, Divided, Positive Median Barrier
Two–Way, Divided, Unprotected Median
Two–Way, Not Divided
NA
RdConfig


RdDefects: Category Proportions
Relative frequency
1.00
0.75
0.50
0.25
0.00
Debris
No Shoulders
Non−Highway Work
None
Obstruction In Roadway
Other
Road Surface Condition
Rut, Holes, Bumps
Shoulders Low, Soft Or High
Traffic Control Device Inoperative, Not Visible Or Missing
Work Zone
NA
RdDefects

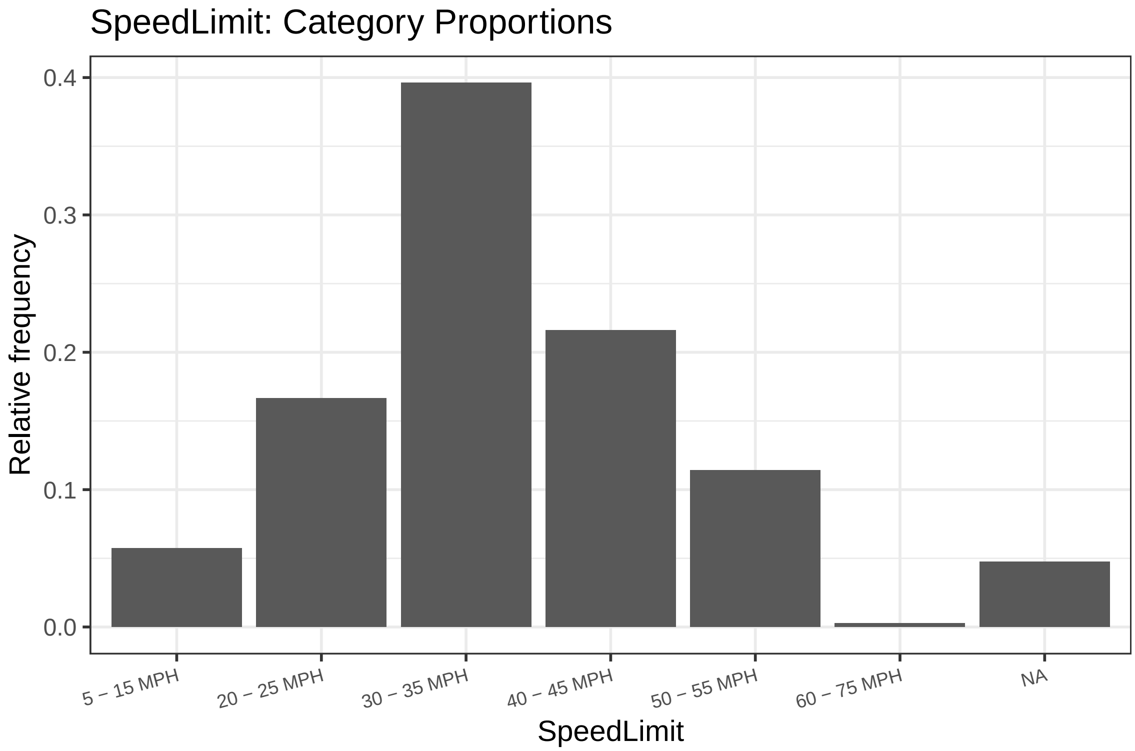

RdSurface: Category Proportions
Relative frequency
0.6
0.4
0.2
0.0
Coarse Asphalt
Concrete
Gravel
Grooved Concrete
Other
Sand
Smooth Asphalt
Soil
NA
RdSurface

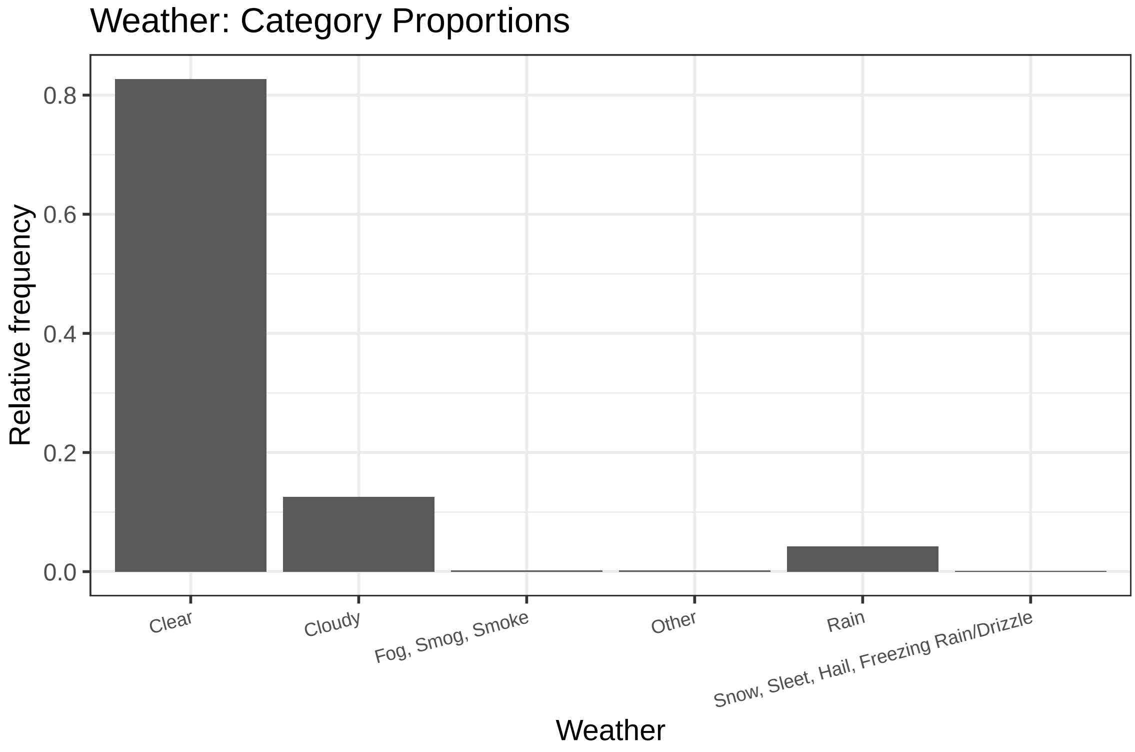


Region: Category Proportions



RuralUrban: Category Proportions


SpeedLimit: Category Proportions
Relative frequency
0.4
0.3
0.2
0.1
0.0
5 – 15 MPH
20 – 25 MPH
30 – 35 MPH
40 – 45 MPH
50 – 55 MPH
60 – 75 MPH
NA
SpeedLimit


Weather: Category Proportions
Relative frequency
0.8
0.6
0.4
0.2
0.0
Clear
Cloudy
Fog, Smog, Smoke
Other
Rain
Snow, Sleet, Hail, Freezing Rain/Drizzle
Weather

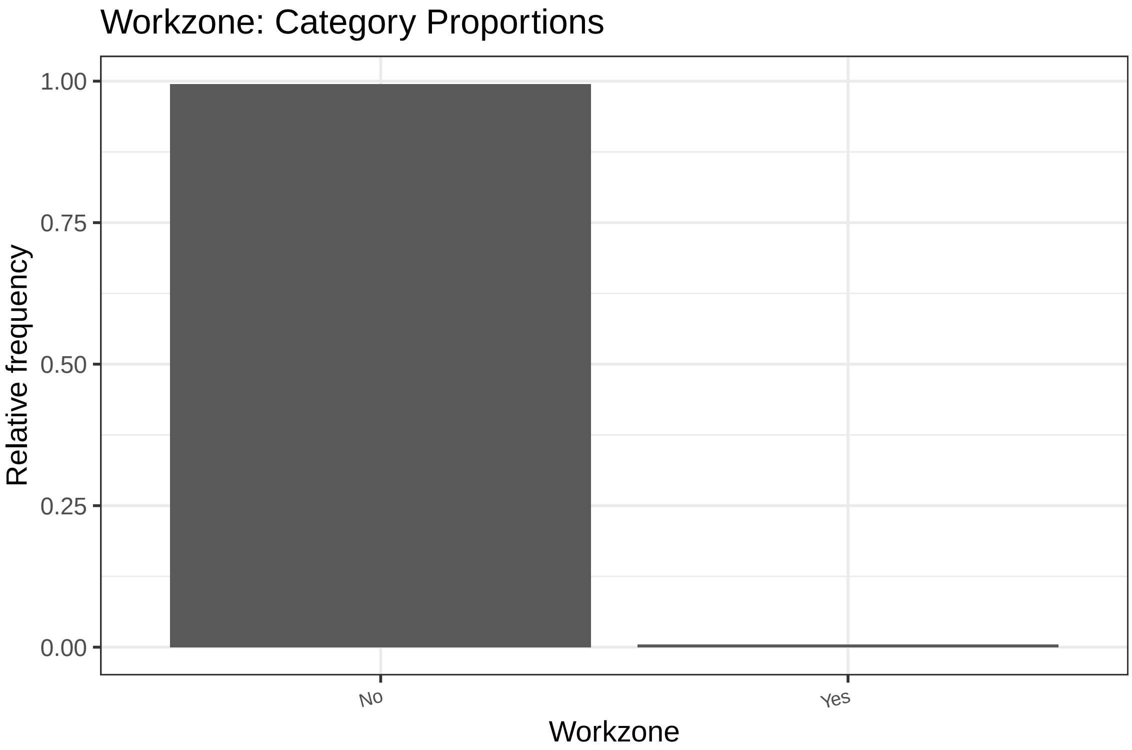


Table 8: Top 20 levels for City

City	n	pct
None - Rural Crash	2545	26.67
Charlotte	1369	14.35
Raleigh	1070	11.21
(Other)	674	7.06
Wilmington	598	6.27
Durham	534	5.60
Greensboro	483	5.06
Fayetteville	328	3.44
Asheville	236	2.47
Winston-Salem	225	2.36
Greenville	206	2.16
Cary	200	2.10
Rocky Mount	196	2.05
High Point	179	1.88
Chapel Hill	151	1.58
Gastonia	135	1.41
Jacksonville	116	1.22
Wilson	115	1.21
Kinston	95	1.00
Lumberton	88	0.92

Table 9: Top 20 levels for County

County	n	pct
Wake	1567	18.08
Mecklenburg	1511	17.44
New Hanover	795	9.17
Guilford	754	8.70
Durham	571	6.59
Cumberland	421	4.86
Buncombe	320	3.69
Pitt	302	3.48
Forsyth	300	3.46
Orange	274	3.16
Gaston	234	2.70
Dare	224	2.58
Robeson	219	2.53
Onslow	214	2.47
Iredell	180	2.08
Nash	165	1.90
Carteret	164	1.89
Craven	153	1.77
Rowan	150	1.73
Brunswick	148	1.71

Table 10: Top 20 levels for CrashGrp

CrashGrp	n	pct
Motorist Overtaking Bicyclist	2336	19.82
Motorist Failed to Yield - Sign-Controlled Intersection	1137	9.65
Motorist Left Turn / Merge	1018	8.64
Motorist Failed to Yield - Midblock	877	7.44
Bicyclist Failed to Yield - Midblock	781	6.63
Bicyclist Failed to Yield - Sign-Controlled Intersection	707	6.00
Crossing Paths - Other Circumstances	674	5.72
Motorist Right Turn / Merge	620	5.26
Bicyclist Failed to Yield - Signalized Intersection	569	4.83
Non-Roadway	523	4.44
Bicyclist Left Turn / Merge	515	4.37
Loss of Control / Turning Error	493	4.18
Motorist Failed to Yield - Signalized Intersection	391	3.32
Head-On	320	2.72
Bicyclist Overtaking Motorist	241	2.04
Parallel Paths - Other Circumstances	202	1.71
Bicyclist Right Turn / Merge	139	1.18
Backing Vehicle	93	0.79
NA's	89	0.76
Other / Unusual Circumstances	61	0.52

Table 11: Top 20 levels for CrashType

CrashType	n	pct
Motorist.Overtaking.Other.Unknown	1266	14.03
Motorist.Drive.Out.Sign.Controlled.Intersection	1046	11.60
Motorist.Left.Turn.Opposite.Direction	883	9.79
Motorist.Drive.Out.Commercial.Driveway.Alley	718	7.96
Bicyclist.Ride.Through.Sign.Controlled.Intersection	561	6.22
Non.Roadway	523	5.80
Motorist.Right.Turn.Same.Direction	514	5.70
Bicyclist.Left.Turn.Same.Direction	441	4.89
Motorist.Overtaking.Undetected.Bicyclist	430	4.77
Bicyclist.Ride.Through.Signalized.Intersection	390	4.32
Motorist.Overtaking.Misjudged.Space	329	3.65
Motorist.Overtaking.Bicyclist.Swerved	311	3.45
Head.On.Bicyclist	258	2.86
Bicyclist.Ride.Out.Other.Midblock	233	2.58
Signalized.Intersection.Other.Unknown	233	2.58
Motorist.Drive.Out.Right.Turn.on.Red	203	2.25
Bicyclist.Ride.Out.Residential.Driveway	188	2.08
Bicyclist.Ride.Out.Midblock.Unknown	169	1.87
Sign.Controlled.Intersection.Other.Unknown	165	1.83
Crossing.Paths.Intersection.Other.Unknown	160	1.77

Table 12: Top 20 levels for DrvrVehTyp

DrvrVehTyp	n	pct
Passenger Car	6153	52.21
Sport Utility	2026	17.19
Pickup	1525	12.94
Unknown	952	8.08
Van	556	4.72
Light Truck (Mini-Van, Panel)	166	1.41
Single Unit Truck (2-Axle, 6-Tire)	81	0.69
Motorcycle	60	0.51
Police	59	0.50
Commercial Bus	29	0.25
Tractor/Semi-Trailer	28	0.24
Truck/Trailer	25	0.21
Pedalcycle	24	0.20
Single Unit Truck (3 Or More Axles)	23	0.20
School Bus	22	0.19
Taxicab	18	0.15
Other Bus	16	0.14
Moped	9	0.08
Unknown Heavy Truck	8	0.07
Pedestrian	5	0.04

Table 13: Top 20 levels for RdFeature

RdFeature	n	pct
No Special Feature	6307	53.48
Four-Way Intersection	1863	15.80
T-Intersection	1569	13.30
Driveway, Public	706	5.99
NA's	515	4.37
Driveway, Private	295	2.50
Related To Intersection	198	1.68
On or Off Ramp	60	0.51
Other	57	0.48
Bridge	50	0.42
Y-Intersection	47	0.40
Traffic Circle/Roundabout	36	0.31
Shared-Use Paths Or Trails	24	0.20
Alley Intersection	21	0.18
Bridge Approach	12	0.10
Railroad Crossing	9	0.08
Non-Intersection Median Crossing	8	0.07
Underpass	7	0.06
Five-Point, Or More	6	0.05
End Or Beginning-Divided Highway	3	0.03

Table 14: Top 20 levels for TraffCntrl

TraffCntrl	n	pct
No Control Present	5360	45.43
Stop Sign	2462	20.87
Stop And Go Signal	2158	18.29
Double Yellow Line, No Passing Zone	1459	12.37
NA's	146	1.24
Yield Sign	67	0.57
Other	54	0.46
Flashing Stop And Go Signal	33	0.28
Flashing Signal With Stop Sign	25	0.21
Human Control	14	0.12
Flashing Signal Without Stop Sign	8	0.07
RR Gate And Flasher	4	0.03
School Zone Signs	4	0.03
Warning Sign	4	0.03

6 Pre-Processing and Feature Reduction

Some variables will need to be converted to a numerical datatype for the machine learning models to work properly. Categorical character columns will need to be converted to dummy variables (0, 1 vectors). This will increase the number of columns in the dataset to the scale of hundreds of features.

Feature reduction will be carried out to reduce memory usage and computation time of fitting models. Feature reduction will be carried out by removing features are highly correlated since they will affect model convergence. Features that are directly related to the response, BikeInjurySerious, will also be removed since those features would make the model results overly optimistic. Finally, dummy variables with near-zero variance (<5% unique values) will be removed since they won't contribute much information to the model.

Note that principal components analysis (PCA) will only be carried out for a linear model. Therefore, this pre-processing step (PCA) will only be carried out in the pre-processing argument of a secondary linear model to compare with a non-PCA linear model.

6.1 Create Dummy Variables

All categorical variable levels are converted to dummy variables (0, 1 vectors). Some name fixes will be required to make column names more consistent. If any dummy variables have NA's then replace with 0's as explicit negative values to preserve largest complete dataset possible.

6.2 Correlated Variables

In this section, correlated features will be removed based on complete cases. Any variables with zero variance after filtering for complete cases will be removed because they won't offer any information to a model.

Table 15: Variables removed with high pairwise correlation (rho>0.75)

Correlated Vars	Correlated Vars contin.
BikeRace.White	RdConfig.Two.Way.Not.Divided
County.Beaufort	RuralUrban.Urban
County.Buncombe	BikeInjuryCat.O
County.Chowan	BikeInjuryCat.C
County.Cumberland	BikeInjuryCat.B
County.Forsyth	City.Durham
County.Gaston	City.Rocky.Mount
County.Guilford	City.Columbia
County.Lee	CrashGrp.Non.Roadway
County.Lenoir	CrashGrp.Bicyclist.Left.Turn.Merge
County.Mecklenburg	CrashGrp.Bicyclist.Failed.to.Yield.Sign.Controlled.Intersection
County.New.Hanover	CrashGrp.Bicyclist.Failed.to.Yield.Signalized.Intersection
County.Pasquotank	CrashGrp.Bicyclist.Right.Turn.Merge
County.Pitt	CrashGrp.Head.On
County.Vance	CrashGrp.Parking.Bus.Related
County.Wake	CrashGrp.Motorist.Failed.to.Yield.Midblock
County.Watauga	CrashGrp.Motorist.Failed.to.Yield.Sign.Controlled.Intersection
County.Wayne	CrashGrp.Motorist.Left.Turn.Merge
County.Wilson	CrashGrp.Motorist.Right.Turn.Merge
CrashAlcoh.Yes	CrashLoc.Non.Roadway
Locality.Rural.LT30pct.Developed	CrashID

6.3 Variables that are directly related to response

The following variables are directly related to the response and if used in a model, the results would be overly optimistic.

- Ambulance - if ambulance used then it would already known to be serious.
- BikeInjuryDisc - directly related to response
- BikeInjuryCat - directly related to response
- NumBicsX - all variables are related to the BikeInjuryCat + 50% missing values

6.4 Create train and test dataset partitions

The training dataset accounts for 80% of the full dataset, while testing dataset will account for 20%. All models will be fit using the training data, optimum probability cut-off values will be explored from the training set and applied to the predicted probabilities from the test dataset.

Table 16: Train data: response class summary

BikeInjurySerious	n	pct
0	8724	92.425045
1	715	7.574955

Table 17: Test data: response class summary

BikeInjurySerious	n	pct
0	2168	91.903349
1	191	8.096651

6.5 Remove near-zero variance features from training dataset

Remove variables that have near-zero variance since they will likely not contribute much information to the classification models. Any variable that has less than 5% distinct values out of total number of samples will be removed from the training dataset.

```
## [1] "599 dummy variables removed for near-zero variance"
```

7 Modeling

Multiple models will be evaluated for performance based on ROC metric and by computation time. The best models will be used in an ensemble model. Models will be chosen for the ensemble based on computation time, ROC, complimentary results, i.e., one model may have higher sensitivity while another may have higher specificity, and an ensemble could balance these models.

The model predictors are selected based on caret's underlying training methodology. All final model's hyper-parameters and predictors will be chosen based on the optimum area under the ROC (i.e. AUC). AUC is a performance metric that will balance sensitivity (true positive rate) and specificity (true negative rate). AUC values range from 0 to 1 and a value of 1 considered a perfect classification model. hyper-parameter grid search is set to default which chooses hyper-parameters randomly.

Repeated 10-fold cross-validation will carried out on all models below unless otherwise noted. In some cases, the computation method is quite complex and is very slow to compute (e.g. random forests). In these cases, repeated cross-validation will be reduced to non-repeated cross-validation.

Since the prevalence of serious injury is so low (around 5%), up-sampling has been used to increase representation when 10-fold cross validation occurs.

All models will be run on the training dataset.

7.1 Linear Classifiers

Linear classifier models parameterize the probability of a response, serious injury in our case, based on a linear combination data features. These models are nice since we can intuitively see the importance of each feature in the final model with respect to how they influence the probability of a serious injury. In this section, a general linear model w/ a binary link function (aka logistic classification model) will be used which takes numerical features as inputs and will be used to predict probability of having a serious injury. Principal components analysis (PCA) will be attempted to see if the feature space can be further reduced and model improved.

glmnet package is used for logistic regression with elastic-net penalty built-in. PCA is carried out on second logistic model to see if there is any improvement. Multiple hyper-parameters, alpha (mixing parameter: 0 = ridge, 1 = lasso) and lambda (penalty size), will be explored by caret.

7.1.1 Logistic (elastic-net) Model

```
set.seed(42)
# seeds to be used reproducable resampling
seeds <- vector(mode = "list", length = 31)
# longest set of hyper-params is 33 (logistic mdl)
for(i in 1:30) seeds[[i]] <- sample.int(1000, 33)
seeds[[31]] <- sample.int(1000, 1)

# caret control
binomial_control <- trainControl(
  summaryFunction = twoClassSummary,
  classProbs = TRUE,
  savePredictions = 'final',
  sampling = 'up',
  method = 'repeatedcv',
  number = 10,
  repeats = 3,
  index = createMultiFolds(
    bike_crashes_preProcessed$BikeInjurySerious,
    k = 10,
    times = 3
  ),
  seeds = seeds
)

########################################
# ElasticNet Logistic Classification #
########################################
# timing process
a <- Sys.time()
# create clusters
cl <- makeCluster(detectCores() / 2)
registerDoParallel(cl)

log_mdl <- train(
  BikeInjurySerious ~ .,
  data = bike_crashes_preProcessed,
  method = 'glmnet',
  family = 'binomial',
  na.action = na.omit,
  trControl = binomial_control,
  tuneGrid = expand.grid(
    .alpha = seq(0, 1, .1),
    .lambda = c(0.1, 0.01, 0.001)
  ),
  seeds = seeds,
)
stopCluster(cl)
b <- Sys.time()
log_time <- b - a
print(log_time)
```

```
## Time difference of 3.419719 mins
```

7.1.2 Logistic Model (elastic-net) with PCA

```
set.seed(42)

#################################################################
# Elasticnet Logistic Classification using Principal Components #
#################################################################
# timing process
a <- Sys.time()
# create clusters
cl <- makeCluster(detectCores() / 2)
registerDoParallel(cl)
log_PCA_mdl <- train(
  BikeInjurySerious ~ .,
  data = bike_crashes_preProcessed,
  method = 'glmnet',
  family = 'binomial',
  preProcess = c('pca'),
  na.action = na.omit,
  trControl = binomial_control,
  tuneGrid = expand.grid(
    .alpha = seq(0, 1, .1),
    .lambda = c(0.1, 0.01, 0.001)
  ),
  seeds = seeds
)
stopCluster(cl)
b <- Sys.time()
log_PCA_time <- b - a
print(log_PCA_time)
```

```
## Time difference of 3.5541 mins
```

7.2 Non-linear Classification Models

In some cases, linear classifiers do not perform well when the response classes (i.e. serious, non-serious) do not have a linear relationship with the features. Put another way, linear classifiers can not produce a straight line in the feature space that separates the response classes. Below, naive Bayes classifier, neural network and random forest models are used.

7.2.1 Naive Bayes Model

```
set.seed(42)

###############
# Naive Bayes #
###############
# timing process
a <- Sys.time()
# create clusters
cl <- makeCluster(detectCores() / 2)
registerDoParallel(cl)
nb_mdl <- train(
  BikeInjurySerious ~ .,
  data = bike_crashes_preProcessed,
  method = 'naive_bayes',
  na.action = na.omit,
  trControl = binomial_control,
  tuneGrid = expand.grid(
    .laplace = 0:1,
    .adjust = 0:1,
    .usekernel = c(T, F)
  )
)
stopCluster(cl)
b <- Sys.time()
nb_time <- b - a
print(nb_time)
```

```
## Time difference of 17.8804 secs
```

7.2.2 K-nearest Neighbors Model

K-neareset neighbors is slow in computation but not as much as the random forests model. For this reason, the cross-validation has not been changed as it was for random forests (see below).

```
set.seed(42)

##################
# Neural Network #
##################
# timing process
a <- Sys.time()
# create clusters
```

```
cl <- makeCluster(detectCores() / 2)
registerDoParallel(cl)
knn_mdl <- train(
  BikeInjurySerious ~ .,
  data = bike_crashes_preProcessed,
  method = 'knn',
  na.action = na.omit,
  trControl = binomial_control,
  tuneGrid = expand.grid(
    k = 2:21
  )
)
stopCluster(cl)
b <- Sys.time()
knn_time <- b - a
print(knn_time)
```

```
## Time difference of 12.54299 mins
```

7.2.3 Random Forests Model

Random forests models are extremely computationally expensive and cross-validation on exacerbates this issue. For this reason, repeated 10-fold cross-fold validation has been reduced to single 10-fold cross-validation to speed up the process.

```
set.seed(42)

# random forests is the most computation intensive with repeatedcv
# will reduce to non-repeated cv since random forests has methods
# that are similar to cross-validation already (many trees in a forest)
rf_control <- trainControl(
  summaryFunction = twoClassSummary,
  classProbs = TRUE,
  savePredictions = 'final',
  sampling = 'up',
  method = 'cv',
  number = 10,
  index = createFolds(
    bike_crashes_preProcessed$BikeInjurySerious,
    k = 10,
    returnTrain = TRUE
  ),
  # collect the first 5 seed lists and the final model seed
  seeds = seeds[c(1:10, 31)]
)

rf_tune_grid <- expand.grid(
  .mtry = 2:11,
  .splitrule = c('gini'),
  .min.node.size = 1
)
```

```
#################
# Random Forest #
#################
# timing process
a <- Sys.time()
# create clusters
cl <- makeCluster(detectCores() / 2)
registerDoParallel(cl)
rf_mdl <- train(
  BikeInjurySerious ~ .,
  data = bike_crashes_preProcessed,
  method = 'ranger',
  num.trees = 100,
  importance = 'permutation',
  na.action = na.omit,
  trControl = rf_control,
  tuneGrid = rf_tune_grid
)
stopCluster(cl)
b <- Sys.time()
rf_time <- b - a
print(rf_time)
```

```
## Time difference of 5.518838 mins
```

8 Prediction Performance

In this section, each model will be evaluated on its' performance in accuracy, sensitivity and specificity based on the optimum probability cutoff used to classify each observation as a serious or non-serious injury. You will see plots below that show that a model with high accuracy is not necessarily the best at classifying observations. In fact, a model that predicts non-serious 100% of the time has a model accuracy of 95% since the prevalence of serious injury is 5% in the dataset.

All final predictions will be based on test dataset.

8.1 Linear Predictions

Both standard logistic model and PCA logistic model have similar outcomes. Logistic model without PCA was faster at achieving a similar result. Therefore PCA was not useful in this case and will not be used elsewhere.

8.1.1 Logistic (elastic net) Predictions

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction    Serious Non.Serious
##   Serious         496        2466
##   Non.Serious     219        6258
##
##                Accuracy : 0.7155
##                  95% CI : (0.7063, 0.7246)
##     No Information Rate : 0.9243
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1683
##
##  Mcnemar’s Test P-Value : <2e-16
##
##             Sensitivity : 0.69371
##             Specificity : 0.71733
##          Pos Pred Value : 0.16745
##          Neg Pred Value : 0.96619
##              Prevalence : 0.07575
##          Detection Rate : 0.05255
##    Detection Prevalence : 0.31380
##       Balanced Accuracy : 0.70552
##
##        ’Positive’ Class : Serious
##
```

Table 18: Top 20 most import features

	Overall
SpeedLimit.50.55.MPH	100.0000
CrashType.Motorist.Drive.Out.Sign.Controlled.Intersection	93.8318
BikeAlcFlg.Yes	70.0472
CrashGrp.Bicyclist.Failed.to.Yield.Midblock	64.0854
RdClass.Public.Vehicular.Area	63.7787
LightCond.Daylight	58.5945
BikePos.Sidewalk.Crosswalk.Driveway.Crossing	53.3659
CrashType.Motorist.Drive.Out.Commercial.Driveway.Alley	46.7850
DrvrRace.Black	43.9852
TraffCntrl.Stop.And.Go.Signal	43.5383
RdClass.Local.Street	43.3392
SpeedLimit.40.45.MPH	41.3908
RdCharacte.Straight.Level	40.0770
CrashType.Motorist.Overtaking.Other.Unknown	39.5671
RdFeature.No.Special.Feature	32.5809
DrvrRace.White	29.6834
BikePos.Travel.Lane	28.6621
BikePos.Non.Roadway	28.1256
CrashMonth.April	23.7084
NumLanes.3.lanes	21.5164

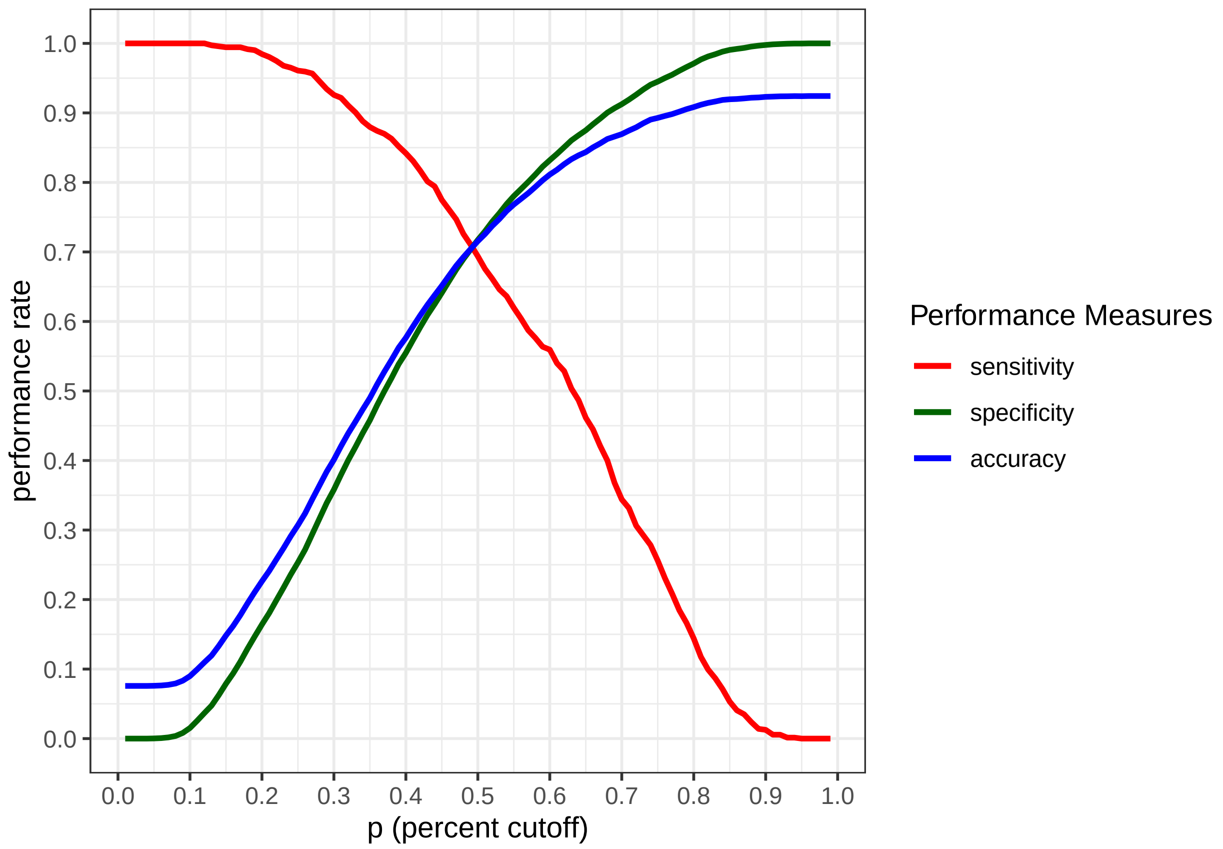


Table 19: Test Data Performance: GLM

n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	model
144	191	1355	2168	0.6354387	0.7539267	0.625	logistic

8.1.2 Logistic (elastic net) with PCA Predictions

PCA does not seem to improve the logistic model enough to be useful.

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction    Serious Non.Serious
##   Serious         501        2558
##   Non.Serious     214        6166
##
##                Accuracy : 0.7063
##                  95% CI : (0.697, 0.7155)
##     No Information Rate : 0.9243
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1627
##
##  Mcnemar's Test P-Value : <2e-16
```

```
##
##             Sensitivity : 0.70070
##             Specificity : 0.70679
##          Pos Pred Value : 0.16378
##          Neg Pred Value : 0.96646
##              Prevalence : 0.07575
##          Detection Rate : 0.05308
##    Detection Prevalence : 0.32408
##       Balanced Accuracy : 0.70374
##
##        'Positive' Class : Serious
##
```

Table 20: Top 20 most import features

	Overall
PC1	100.0000
PC74	78.4525
PC10	62.8136
PC57	60.9600
PC7	53.6930
PC6	39.5713
PC72	39.4217
PC64	38.9841
PC59	37.9597
PC70	32.9899
PC56	31.2110
PC38	30.1177
PC5	29.8255
PC66	29.7079
PC41	29.5371
PC48	27.5691
PC15	26.8022
PC8	26.4524
PC37	25.8592
PC36	24.6489

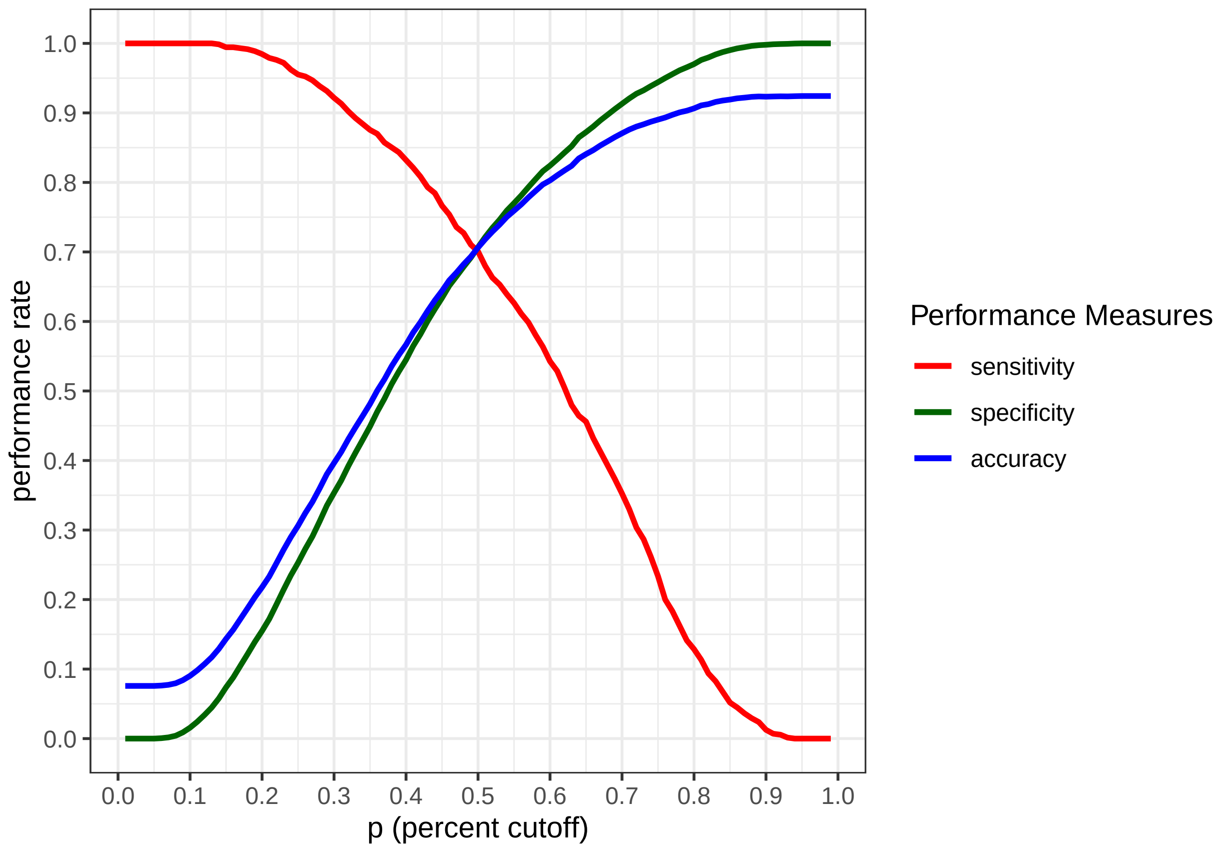


Table 21: Test Data Performance: GLM w/ PCA

n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	model
144	191	1367	2168	0.6405256	0.7539267	0.6305351	logistic w/ PCA

8.2 Non-linear Predictions Predictions

Naive bayes classifier is the fastest and has a very similar result the GLM models above. Both neural network and random forest take considerably longer to compute and don't perform much better than naive bayes. Therefore, only the naive bayes model made it into ensemble model.

Notice that for random forest model, since we dropped the repeated part of the cross-validation and dropped the k in k-fold to 5 from 10, the model is over-fitting a bit more than the other models. The high accuracy values in the random forest performance plot and the low accuracy with the test dataset confirm this. Since the random forest algorithm is slow compared to these other non-linear models it was not be used in ensemble model.

8.2.1 Naive Bayes Predictions

```
## Confusion Matrix and Statistics
##
##               Reference
## Prediction     Serious Non.Serious
##   Serious          487        2836
##   Non.Serious     228        5888
##
##                Accuracy : 0.6754
##                  95% CI : (0.6658, 0.6848)
##     No Information Rate : 0.9243
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1331
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.68112
##             Specificity : 0.67492
##          Pos Pred Value : 0.14655
##          Neg Pred Value : 0.96272
##              Prevalence : 0.07575
##          Detection Rate : 0.05159
##    Detection Prevalence : 0.35205
##       Balanced Accuracy : 0.67802
##
##        'Positive' Class : Serious
##
```

Table 22: Top 20 most import features

	Importance
City.None.Rural.Crash	100.0000
RdClass.Local.Street	94.0425
SpeedLimit.50.55.MPH	87.9994
Locality.Urban.GT70pct.Developed	87.8060
Developmen.Farms.Woods.Pastures	77.5317
BikePos.Travel.Lane	74.9192
CrashGrp.Motorist.Overtaking.Bicyclist	70.6176
CrashLoc.Non.Intersection	67.1797
LightCond.Daylight	64.4533
BikeDir.With.Traffic	62.2455
BikeAge	61.9244
BikePos.Sidewalk.Crosswalk.Driveway.Crossing	60.1323
RdClass.State.Secondary.Route	58.7295
SpeedLimit.30.35.MPH	58.5851
CrashType.Motorist.Overtaking.Other.Unknown	51.7090
LightCond.Dark.Roadway.Not.Lighted	47.8285
Region.Piedmont	40.8808
TraffCntrl.Stop.And.Go.Signal	36.7990
BikeAlcFlg.Yes	35.0899
RdCharacte.Straight.Level	33.0943

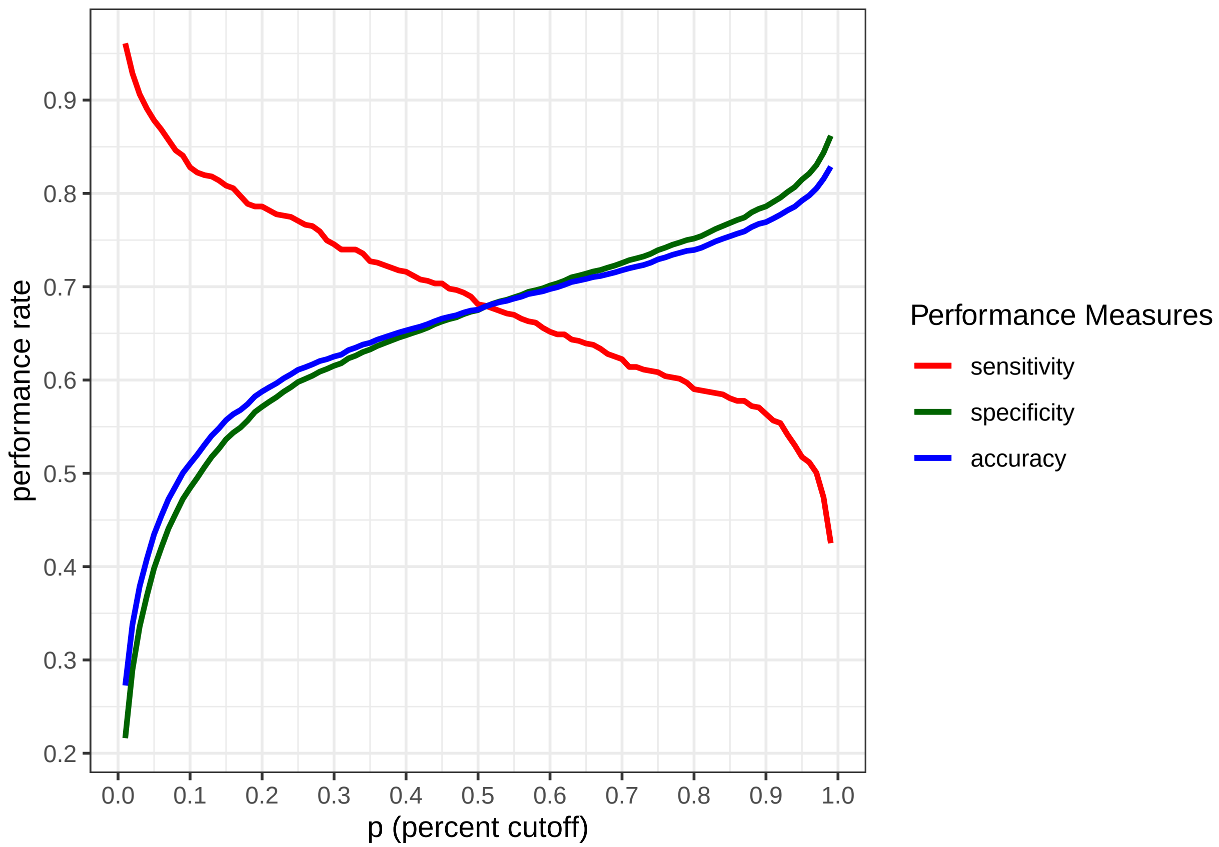


Table 23: Test Data Performance: Naive Bayes

n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	model
144	191	1269	2168	0.5989826	0.7539267	0.5853321	naive bayes

8.2.2 K-nearest Neighbors Predictions

Predictions for K-nearest neighbor take a lot longer to compute than other models.

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction    Serious Non.Serious
##   Serious         652        4066
##   Non.Serious      63        4658
##
##                Accuracy : 0.5626
##                  95% CI : (0.5525, 0.5726)
##     No Information Rate : 0.9243
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1249
##
##  Mcnemar’s Test P-Value : <2e-16
```

```
##
##             Sensitivity : 0.91189
##             Specificity : 0.53393
##          Pos Pred Value : 0.13819
##          Neg Pred Value : 0.98666
##              Prevalence : 0.07575
##          Detection Rate : 0.06908
##    Detection Prevalence : 0.49984
##       Balanced Accuracy : 0.72291
##
##        'Positive' Class : Serious
##
```

Table 24: "Top 20 most import features"

	Importance
City.None.Rural.Crash	100.0000
RdClass.Local.Street	94.0425
SpeedLimit.50.55.MPH	87.9994
Locality.Urban.GT70pct.Developed	87.8060
Developmen.Farms.Woods.Pastures	77.5317
BikePos.Travel.Lane	74.9192
CrashGrp.Motorist.Overtaking.Bicyclist	70.6176
CrashLoc.Non.Intersection	67.1797
LightCond.Daylight	64.4533
BikeDir.With.Traffic	62.2455
BikeAge	61.9244
BikePos.Sidewalk.Crosswalk.Driveway.Crossing	60.1323
RdClass.State.Secondary.Route	58.7295
SpeedLimit.30.35.MPH	58.5851
CrashType.Motorist.Overtaking.Other.Unknown	51.7090
LightCond.Dark.Roadway.Not.Lighted	47.8285
Region.Piedmont	40.8808
TraffCntrl.Stop.And.Go.Signal	36.7990
BikeAlcFlg.Yes	35.0899
RdCharacte.Straight.Level	33.0943

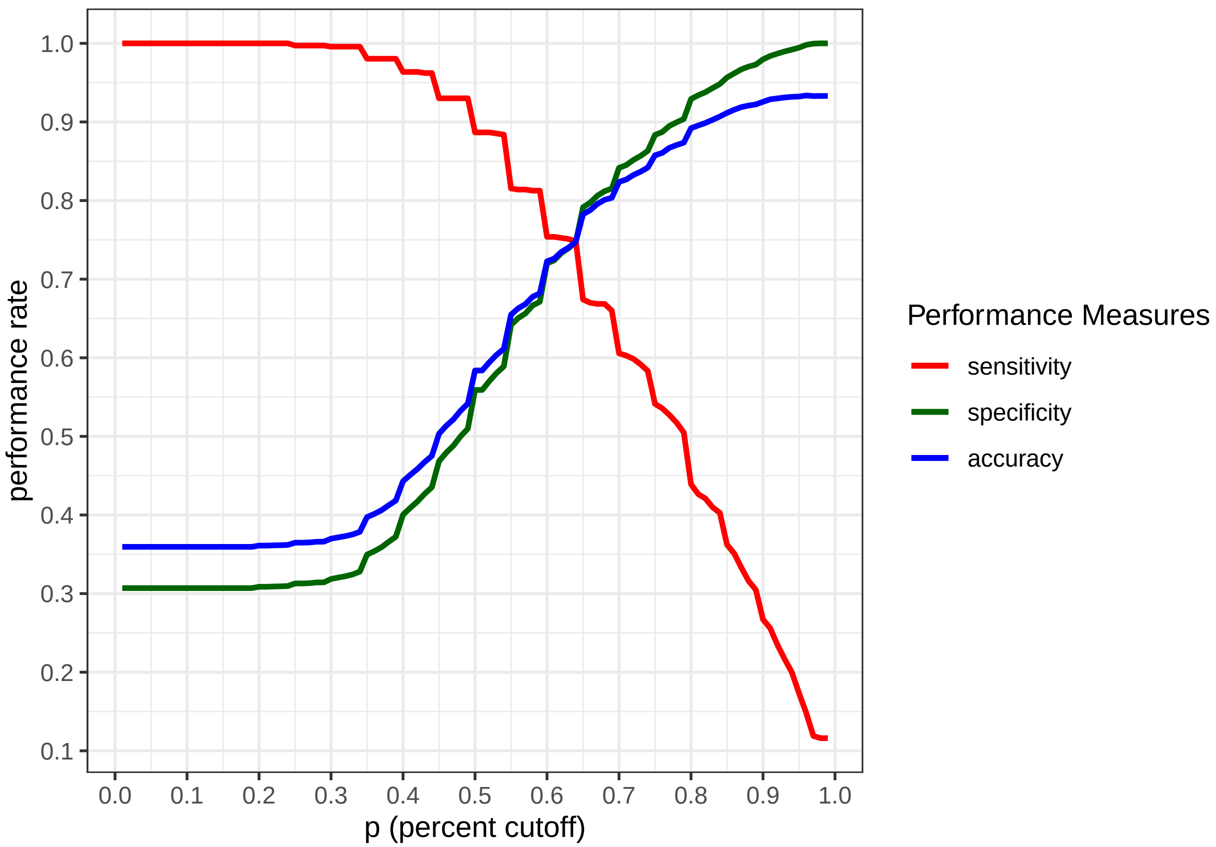


Table 25: Test Data Performance: K-nearest Neighbor

n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	model
137	191	810	2168	0.4014413	0.7172775	0.3736162	K-nearest neighbors

8.2.3 Random Forest Predictions

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction     Serious Non.Serious
##   Serious          715           0
##   Non.Serious        0        8724
##
##                Accuracy : 1
##                  95% CI : (0.9996, 1)
##     No Information Rate : 0.9243
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
##             Sensitivity : 1.00000
##            Specificity : 1.00000
##         Pos Pred Value : 1.00000
##         Neg Pred Value : 1.00000
##             Prevalence : 0.07575
##         Detection Rate : 0.07575
##   Detection Prevalence : 0.07575
##      Balanced Accuracy : 1.00000
##
##       'Positive' Class : Serious
##
```

Table 26: "Top 20 most import features"

	Overall
BikeAge	100.0000
SpeedLimit.50.55.MPH	85.5375
Longitude	81.4012
DrvrAge	78.2033
Latitude	77.0003
CrashYear	75.1454
CrashHour	72.1588
LightCond.Daylight	64.2939
RdClass.Local.Street	60.2723
BikePos.Travel.Lane	56.1102
City.None.Rural.Crash	56.0409
CrashGrp.Motorist.Overtaking.Bicyclist	54.1362
Developmen.Farms.Woods.Pastures	49.9353
Locality.Urban.GT70pct.Developed	45.6743
RdCharacte.Straight.Level	42.6546
SpeedLimit.30.35.MPH	36.4872
BikeDir.With.Traffic	36.2209
CrashType.Motorist.Overtaking.Other.Unknown	34.9676
SpeedLimit.40.45.MPH	34.1532
RdClass.State.Secondary.Route	31.6339

Table 27: Test Data Performance: Random Forest

n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	model
145	191	1172	2168	0.5582874	0.7591623	0.5405904	random forest

9 Model Performance on Test Data

The goal of this section to select the models that will go on to ensemble model.

Logistic model, naive Bayes and random forests all perform similarly. However, both logistic models, with/without PCA included in the analysis, perform better than the others as far as optimizing sensitivity while balancing accuracy and specificity. Of the two logistic models, the one that includes the PCA with the model performs slightly better. Since the time difference for each logistic model is comparable, the PCA logistic model would seem like the most useful model. However, PCA model results are difficult to understand because they are linear combinations of other features. Therefore, the non-PCA version of the model will be used in the ensemble model since the performance isn't that different and is slightly faster. Also worth of note, if PCA is used in one model and not in others, the ensemble model is harder to interpret since it has mixed PCA features and raw feature names listed in caret::varImp.

The naive Bayes model computes the fastest with the lower accuracy due to having a low specificity. Considering how fast the computation is and not having a huge drop in accuracy, this is still a reasonable model. Naive Bayes will be included in the ensemble model since it could be picking up on different features than the linear or other nonlinear models.

K-nearest neighbors was the worst performer in accuracy due to the lowest specificity and therefore, will not move on to the ensemble model.

Random forest model did not perform as well as expected give how well it performed on the training data. This seems to indicate that the model is over fit which is the opposite of what we'd expect from an ensemble decision tree count of 100 and 10-fold cross validation. It's possible that the random forest model is picking up on something that the other models aren't and will be included in the final ensemble model.

Table 28: Model Performance Comparison with Test Data

model	n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	run_time
logistic	144	191	1355	2168	0.6354387	0.7539267	0.6250000	205.1832 secs
logistic w/ PCA	144	191	1367	2168	0.6405256	0.7539267	0.6305351	213.2460 secs
naive bayes	144	191	1269	2168	0.5989826	0.7539267	0.5853321	17.8804 secs
K-nearest neighbors	137	191	810	2168	0.4014413	0.7172775	0.3736162	752.5796 secs
random forest	145	191	1172	2168	0.5582874	0.7591623	0.5405904	331.1303 secs

10 Ensemble Model

10.1 Ensemble Classifier

Logistic, naive Bayes random forests will be used in an ensemble learning model. Note that random forests took the longest to run before and is the bottleneck of the ensemble model. However, since we've already found the tuning of the models, grid search part of the fitting is not needed and will speed up the computation time.

```
## Time difference of 3.833696 mins
```

```
## A glm ensemble of 3 base models: log_mdl, nb_mdl, rf_mdl
##
## Ensemble results:
## Generalized Linear Model
##
## 47195 samples
##     3 predictor
##     2 classes: 'Serious', 'Non.Serious'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 47195, 47195, 47195, 47195, 47195, 47195, ...
## Resampling results:
##
##   ROC        Sens          Spec
##   0.7558487  6.102567e-05  0.999995
```

10.2 Ensemble Predictions

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction    Serious Non.Serious
##   Serious         295           0
##   Non.Serious     420        8724
##
##                Accuracy : 0.9555
##                  95% CI : (0.9511, 0.9596)
##     No Information Rate : 0.9243
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5649
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.41259
##             Specificity : 1.00000
##          Pos Pred Value : 1.00000
##          Neg Pred Value : 0.95407
##              Prevalence : 0.07575
##          Detection Rate : 0.03125
##    Detection Prevalence : 0.03125
##       Balanced Accuracy : 0.70629
##
##        'Positive' Class : Serious
##
```

Table 29: Top 20 most import features by each model and overall

	overall	log_mdl	nb_mdl	rf_mdl
SpeedLimit.50.55.MPH	6.8751	8.5284	4.4277	4.1796
LightCond.Daylight	4.7326	5.4613	3.2430	3.5558
BikeAlcFlg.Yes	4.2716	6.0837	1.7656	1.3123
CrashGrp.Bicyclist.Failed.to.Yield.Midblock	3.6794	5.7604	0.3885	0.2923
BikePos.Sidewalk.Crosswalk.Driveway.Crossing	3.6556	5.2039	3.0256	1.0858
RdClass.Local.Street	3.4971	3.6576	4.7318	3.1951
SpeedLimit.40.45.MPH	3.2802	4.1406	1.6586	1.8871
RdCharacte.Straight.Level	3.0316	3.5821	1.6651	2.1493
BikePos.Travel.Lane	2.9687	3.1098	3.7696	2.7110
RdClass.Public.Vehicular.Area	2.9011	4.5815	1.1511	0.1412
DrvrRace.Black	2.8405	3.8679	0.1035	1.1986
CrashType.Motorist.Overtaking.Other.Unknown	2.7372	3.4459	2.6017	1.5567
CrashType.Motorist.Drive.Out.Commercial.Driveway.Alley	2.4063	3.8514	0.8799	0.0335
DrvrRace.White	2.3824	2.8454	0.9909	1.6470
RdFeature.No.Special.Feature	2.3384	3.0096	0.5885	1.2647
BikeAge	2.2164	0.0958	3.1157	5.7352
TraffCntrl.Stop.And.Go.Signal	2.0016	2.8039	1.8515	0.6651
Developmen.Farms.Woods.Pastures	1.8233	1.5710	3.9010	2.1879
City.None.Rural.Crash	1.8164	0.7539	5.0315	3.5035

	overall	log_mdl	nb_mdl	rf_mdl
Latitude	1.8122	0.6232	0.8070	3.8267

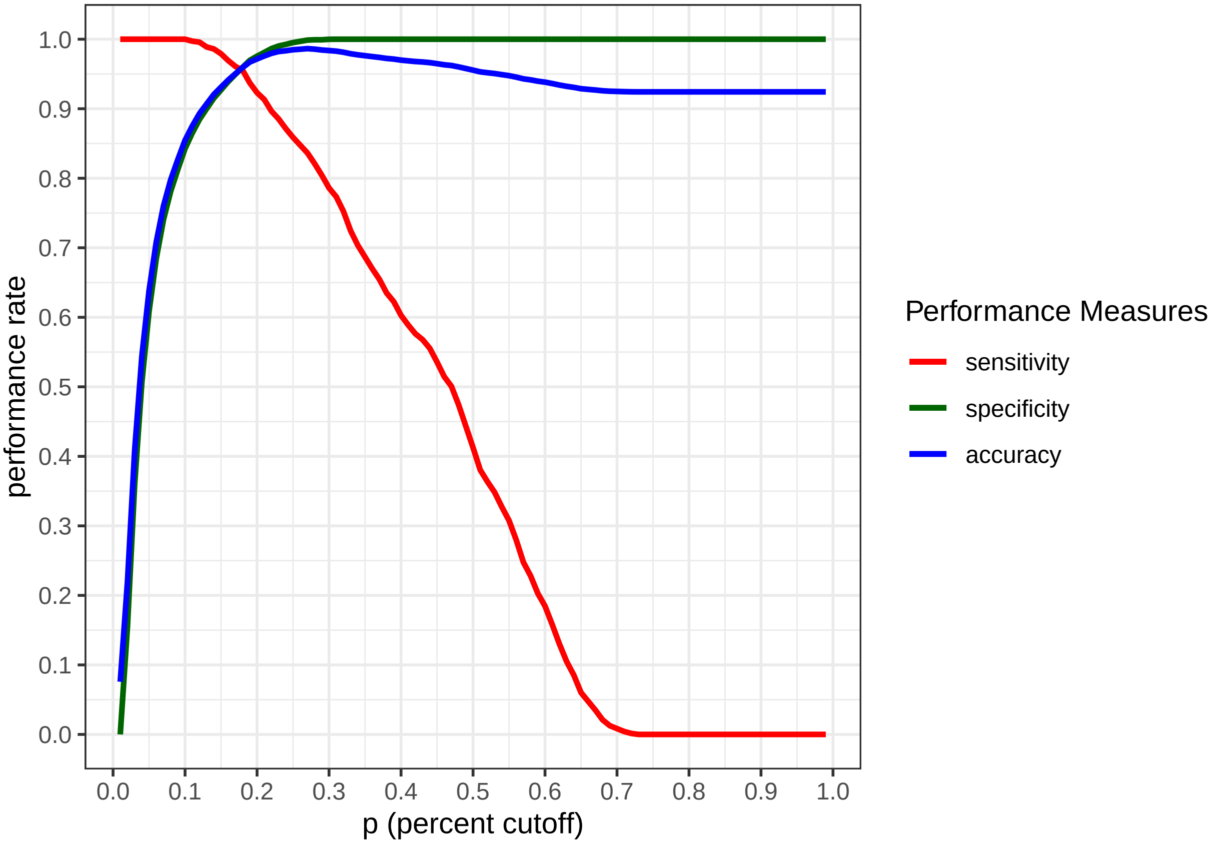


Table 30: Test Data Performance: Ensemble

n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	model
144	191	1431	2168	0.6676558	0.7539267	0.6600554	ensemble (Log & NB & RF)

11 Conclusions

All model performance metrics are listed below with the ensemble model having the best performance. The ensemble model includes an elasticnet logistic model, naive Bayes model and a random forests model. The variable importance for all models including the overall ensemble model importance weight per predictor is provided.

Table 31: Model Performances Comparison (w/ Ensemble)

model	n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	run_time
logistic	144	191	1355	2168	0.6354387	0.7539267	0.6250000	205.1832 secs
logistic w/ PCA	144	191	1367	2168	0.6405256	0.7539267	0.6305351	213.2460 secs

model	n_TP	n_obs_P	n_TN	n_obs_N	accuracy	sensitivity	specificity	run_time
naive bayes	144	191	1269	2168	0.5989826	0.7539267	0.5853321	17.8804 secs
K-nearest neighbors	137	191	810	2168	0.4014413	0.7172775	0.3736162	752.5796 secs
random forest	145	191	1172	2168	0.5582874	0.7591623	0.5405904	331.1303 secs
ensemble (Log & NB & RF)	144	191	1431	2168	0.6676558	0.7539267	0.6600554	230.0218 secs

With the ensemble model being selected as the best model, the top 20 most important variables are as follows:

Table 32: Top 20 most import features by each model and overall

	overall	log_mdl	nb_mdl	rf_mdl
SpeedLimit.50.55.MPH	6.8751	8.5284	4.4277	4.1796
LightCond.Daylight	4.7326	5.4613	3.2430	3.5558
BikeAlcFlg.Yes	4.2716	6.0837	1.7656	1.3123
CrashGrp.Bicyclist.Failed.to.Yield.Midblock	3.6794	5.7604	0.3885	0.2923
BikePos.Sidewalk.Crosswalk.Driveway.Crossing	3.6556	5.2039	3.0256	1.0858
RdClass.Local.Street	3.4971	3.6576	4.7318	3.1951
SpeedLimit.40.45.MPH	3.2802	4.1406	1.6586	1.8871
RdCharacte.Straight.Level	3.0316	3.5821	1.6651	2.1493
BikePos.Travel.Lane	2.9687	3.1098	3.7696	2.7110
RdClass.Public.Vehicular.Area	2.9011	4.5815	1.1511	0.1412
DrvrRace.Black	2.8405	3.8679	0.1035	1.1986
CrashType.Motorist.Overtaking.Other.Unknown	2.7372	3.4459	2.6017	1.5567
CrashType.Motorist.Drive.Out.Commercial.Driveway.Alley	2.4063	3.8514	0.8799	0.0335
DrvrRace.White	2.3824	2.8454	0.9909	1.6470
RdFeature.No.Special.Feature	2.3384	3.0096	0.5885	1.2647
BikeAge	2.2164	0.0958	3.1157	5.7352
TraffCntrl.Stop.And.Go.Signal	2.0016	2.8039	1.8515	0.6651
Developmen.Farms.Woods.Pastures	1.8233	1.5710	3.9010	2.1879
City.None.Rural.Crash	1.8164	0.7539	5.0315	3.5035
Latitude	1.8122	0.6232	0.8070	3.8267

11.1 Possible Interpretations of Predictors and Suggestions

The first 5 predictors by overall importance make sense as far as being predictive of serious bike crashes in that high speed limits (50 - 55 mph), bicyclist under the influence of alcohol, bicyclist failing to yield mid-block and bike position crosswalk/crossing/driveway all point to the bicyclist not be a responsible commuter.

Notice, I did not mention daylight conditions being the second highest predictor of serious bike accidents. This is because it is not really clear why this is such a good predictor. It could be due to low visibility or warning of a bicyclist and their intentions. If a bicyclist is biking at night, they would most likely have bike lights on which helps drivers to see that an unprotected traveler is in the road and gives their direction since the bike light points forward and tail light points backwards. In addition, during the day, there is more traffic on the street and the sidewalk which could distract drivers.

It may seem surprising that straight roads is a good predictor serious bike crashes, but if you think about when you drive on a straight road for a while and you get into “the zone” where you don’t remember the last 5-10 minutes of driving because you were on autopilot. This could be the case especially if the speeds are 40 - 55 mph where it would very difficult to avoid an obstacle in the road.

I don't think we have enough data to explain why a driver's race is a predictor of seriousness of a bike-vehicle crash. We would need to know more about the state of North Carolina and how roads in predominantly black neighborhoods differ in bike friendliness and traffic from roads in predominantly white neighborhoods. We may find that there is an uneven distribution of bike friendly roads in either neighborhood which could affect a driver's experience how to share the road with a bicyclist. No conclusions can be made with the data we have here. These are just possible things we could look into with more socio-ecenomic data.

Rural crashes are on the top 20 predictors list with development type: farms, woods, pastures; non-city rural. The main take away from these results is that bicyclists are not being protected on rural roads where touring bicyclists spend most of their time. A recommendation that I have is to cities that book-end the roads that are most heavily traveled by bicyclists, invest in extending the shoulder of these roads and to include bike signs. This could later be added to the bike crash data as "bicycle improved rural road" and the treatment could be tested again roads that were not treated (control).