FINAL REPORT

Batch details	PGP DSE JUL 25
Team members	Mr. Anurag Kumar Singh Mr. Siddhesh Sawant Mr. Bharath R Mr. Mohammed Najeeb Ur Raheman Mr. Hemanth Mandava
Domain of Project	Road Safety
Proposed project title	Road Safety & Crash Intelligence System (RSCIS)
Group Number	3
Team Leader	Mr. Anurag Kumar Singh

Table of Contents

Industry Review

Current practices:

Road safety analytics has become a critical focus area for governments as crash volumes and traffic density continue to rise across major cities worldwide. Traditionally, police departments and transportation agencies have relied on manual crash summaries, static reports, and field inspections to analyze accident patterns. However, with the increasing availability of geospatial data, traffic sensors, and detailed crash records, modern safety planning is shifting towards data-driven intelligence systems. Today, crash analysis is no longer limited to counting incidents; agencies examine *where*, *when*, and *why* crashes occur using advanced analytics.

The current industry practice includes analyzing traffic flow, road geometry, driver behavior, and environmental conditions through digital tools and crash-reporting platforms. Despite this, most agencies still depend heavily on descriptive reports that lack predictive capability, leaving them reactive rather than proactive. With cities becoming more congested and driver behavior more unpredictable, understanding real-time risk factors has become essential for effective safety intervention.

The modern mobility environment has also changed dramatically. Drivers now navigate complex multilane corridors, high-speed expressways, and signal-dense intersections that amplify crash risks. Factors like distraction, speeding, fatigue, and alcohol influence add new layers of unpredictability. As roadway environments evolve, agencies require more flexible, intelligent, and automated systems capable of identifying high-risk areas, forecasting crash severity, and recommending timely interventions. This shift toward intelligent road-safety analytics is transforming how authorities plan infrastructure, allocate patrol units, improve road design, and enhance public safety outcomes.

Background Research:

Road-safety analysis has progressed significantly over the past three decades. Early studies in the 1990s focused largely on basic statistics such as crash frequency and fatality trends. With limited computing power and minimal geospatial data, analysts mainly used manual hotspot mapping and field surveys to understand accident patterns. However, the explosion of GPS-enabled crash reporting, digital road-network data, and advanced police data-entry systems changed the field entirely.

As crash databases became richer—with fields for driver behavior, vehicle damage patterns, environmental conditions, and precise latitude–longitude coordinates—researchers began applying machine learning techniques to extract deeper insights. The introduction of geospatial clustering methods such as KMeans, DBSCAN, and Agglomerative Clustering marked a major turning point, enabling analysts to identify dangerous corridors more accurately than visual inspection or heatmaps alone.

In the last decade, road safety research has expanded to include behavioral insights (distraction, impairment), weather interaction studies, traffic control effectiveness, and predictive modeling for injury severity. Tools like FHWA's Highway Safety Manual, NHTSA crash datasets, ESRI GIS, and machine-learning libraries (Scikit-learn, XGBoost, LightGBM) have accelerated the shift from static analysis to intelligent crash prediction systems.

The rapid adoption of AI in transportation marks one of the most transformative phases in road-safety research. Today, machine learning models can detect crash hotspots, forecast injury likelihood, recognize risky drivers, and quantify environmental influence—changing the entire approach from *reactive reporting* to *proactive prevention*. Considering the rise in distracted driving and the complexity of modern road networks, the industry continues to push towards real-time crash intelligence, automated risk scoring, and integrated safety dashboards for city planning.

Literature Survey - Publications, Application, Past and Undergoing Research:

Research in road safety and crash analytics spans multiple disciplines, including transportation engineering, data science, human factors psychology, and geospatial intelligence. Academic studies widely utilize datasets from NHTSA, FARS, HSIS, and state crash-reporting systems to explore crash causation and severity modeling. Recent publications emphasize the importance of integrating geospatial features with behavioral and environmental variables for accurate risk prediction.

Significant literature highlights the use of clustering methods (KMeans, DBSCAN, hierarchical clustering) to identify crash hotspots and assess the effectiveness of signalization, signage placement, lighting conditions, and road geometry. Studies in journals such as *Accident Analysis & Prevention*, *Journal of Safety Research*, and *Transportation Research Part F* show that crash risk is highly localized and significantly influenced by driver distraction, speeding, intersection complexity, and weather transitions.

Another research strand examines predictive injury-severity models using logistic regression, random forests, gradient boosting, and XGBoost. These models have been proven effective in capturing nonlinear relationships in severity outcomes, particularly when combined with geospatial and time-of-day data. Literature also highlights the increasing importance of behavioral analytics—identifying high-risk drivers through patterns of distraction, substance abuse, fatigue, or violation history.

Recent advancements also explore the integration of ML models with GIS systems for generating real-time hotspot maps and recommending actionable countermeasures such as improved lane markings, speed control, lighting enhancements, and targeted patrol deployment. Studies emphasize that combining machine learning with domain knowledge leads to significantly more accurate and actionable crash-intelligence systems.

Overall, the literature confirms that the shift toward multi-model predictive systems—like the RSCIS platform—aligns perfectly with current research trends and is considered the future of transportation safety management.

Data Description:

1. Crash Event Details

Includes Report Number, Local Case Number, Agency Name, ACRS Report Type, and Crash Date/Time, which describe the basic identity and timing of each crash.

2. Roadway & Environmental Conditions

Includes Route Type, Road Name, Cross-Street Name, Off-Road Description, Weather, Surface Condition, Light, and Traffic Control to explain how road and environmental conditions influenced the crash.

3. Driver Information & Behavior

Includes Person ID, Drivers License State, Driver At Fault, Driver Substance Abuse, Non-Motorist Substance Abuse, Circumstance, and Driver Distracted By to understand driver behavior, impairment, distraction, and responsibility.

4. Vehicle Information & Impact Data

Includes Vehicle ID, Vehicle Year, Vehicle Make, Vehicle Model, Vehicle Body Type, Vehicle Damage Extent, Vehicle First Impact Location, Vehicle Movement, and Vehicle Going Dir to describe vehicle characteristics, movement, and impact points.

5. Location & Geospatial Data

Latitude, Longitude, and Location provide exact crash coordinates for hotspot mapping and spatial trend analysis.

6. Safety Factors & Operational Indicators

Includes Speed Limit, Driverless Vehicle, and Parked Vehicle to capture situational and operational factors at the time of the crash.

7. Data Gaps & Irrelevant Columns

Columns with high missing values include Off-Road Description, Municipality, Circumstance, Vehicle Damage Extent, Vehicle First Impact Location, and Vehicle Body Type. Identification-only fields like Report Number,

Local Case Number, Person ID, Vehicle ID, and Location (tuple) do not add analytical value and are irrelevant for modeling.

Data Size:

- Number of Columns: 39
- Number of Rows: 205539
- Total Number of Records: 205539

Data Dictionary:

1.	**Report Number**	Unique identifier assigned to each crash report.
2.	**Local Case Number**	Local police agency's internal case ID for the crash.
3.	**Agency Name**	Name of the reporting police agency (Montgomery, Gaithersburg, etc.).
4.	**ACRS Report Type**	Type of crash reported (Property Damage / Injury Crash).
5.	**Crash Date/Time**	Exact date and time when the crash occurred.
6.	**Route Type**	Type of roadway (County Route, State Route, US Route, Local Route).
7.	**Road Name**	Name of the road where the crash occurred.
8.	**Cross-Street Name**	Nearest intersecting road to the crash location.
9.	**Off-Road Description**	Description of off-road area if crash occurred off the main road.
10.	**Municipality**	City or municipal area where the crash occurred.
11.	**Related Non-Motorist**	Indicates if a pedestrian, bicyclist, etc. was involved.
12.	**Collision Type**	Nature of collision (rear-end, side-impact, fixed object, etc.).
13.	**Weather**	Weather condition at time of crash (clear, rain, snow).

14.	**Surface Condition**	Road surface condition (dry, wet, icy).
15.	**Light**	Lighting condition (daylight, dark, dawn, dusk).
16.	**Traffic Control**	Type of traffic signal or signage controlling the intersection.
17.	**Driver Substance Abuse**	Indicates if driver was under the influence of alcohol/drugs.
18.	**Non-Motorist Substance Abuse**	Substance use information for non-motorists, if any.
19.	**Person ID**	Unique identifier for an individual involved in the crash.
20.	**Driver At Fault**	Indicates whether the driver was considered at fault.
21.	**Injury Severity**	Level of injury (no injury, minor, serious, fatal).
22.	**Circumstance**	Driver actions contributing to the crash (speeding, lane change, etc.).
23.	**Driver Distracted By**	Indicates distraction type (phone, passengers, unknown).
24.	**Drivers License State**	State that issued the driver's license.
25.	**Vehicle ID**	Unique ID for each vehicle involved in the crash.
26.	**Vehicle Damage Extent**	Degree/extent of vehicle damage.
27.	**Vehicle First Impact Location**	Initial point of impact on the vehicle.
28.	**Vehicle Body Type**	Body style of the vehicle (SUV, sedan, truck).
29.	**Vehicle Movement**	What the vehicle was doing during the crash (moving, stopped).
30.	**Vehicle Going Dir**	Direction in which the vehicle was traveling (Northbound, Southbound).
31.	**Speed Limit**	Posted speed limit at the crash location.
32.	**Driverless Vehicle**	Indicates if the vehicle was driverless (e.g., autonomous).

33.	**Parked Vehicle**	Shows whether the involved vehicle was parked at the time.
34.	**Vehicle Year**	Manufacturing year of the vehicle.
35.	**Vehicle Make**	Vehicle manufacturer (Toyota, Honda, BMW, etc.).
36.	**Vehicle Model**	Specific model of the vehicle (Corolla, Civic, etc.).
37.	**Latitude**	Latitude coordinate of the crash location.
38.	**Longitude**	Longitude coordinate of the crash location.
39.	**Location**	Combined geolocation (lat, long) tuple.

EXPLORATORY DATA ANALYSIS

Data Loading and Cleaning:

Data Loading on Google Collab (Raw Data)

The dataset was stored in **Excel format** to preserve datetime columns, since CSV files cannot retain datetime data types correctly.

Data was loaded into Google Colab using:

```
data = pd.read_excel('/content/drive/MyDrive/CAPSTONE/CAPSTONE_PROJECT/0_dataset/2 VALIDATED DATA/validated_data.xlsx')
```

Dataset loaded from Excel to preserve datetime formats.
Shape: 205,539 rows × 39 columns (≈8M data points).

Contains mixed data types:

- object (categorical)

- int64 (numerical)
- float64 (latitude/longitude)
- datetime64 (Crash Date/Time)

Main feature categories:

- Crash details: date/time, weather, surface, collision type
- Driver details: at-fault, substance use, distraction, license state
- Vehicle details: body type, movement, damage, make/year/model
- Location info: road name, cross-street, municipality, coordinates

Data types were checked using df.info() to confirm correct imports.

Column wise Anomaly Treatment

S.No	Column Name	Anomaly Type	Challenges During Treatment	Method of Treatment of Anomaly
1	Agency Name	Inconsistent capitalization and naming	Consolidating multiple representations of the same agency	Standardized using .str.title() and manual mapping
2	Route Type	Inconsistent naming and categorization of route types	Grouping similar route types under a canonical name	Manual mapping using a dictionary (route_type_map)
3	Collision Type	Inconsistent naming and categorization; presence of 'UNKNOWN' values	Consolidating similar collision descriptions and handling unknown values	Manual mapping (collision_map); UNKNOWN → NaN

4	Weather	Inconsistent naming; varied descriptions; UNKNOWN values	Grouping similar weather conditions and handling unknown values	Manual mapping (weather_map); UNKNOWN → NaN
5	Surface Condition	Inconsistent naming; varied descriptions; UNKNOWN values	Grouping similar surface conditions	Manual mapping (surface_map); UNKNOWN → NaN
6	Light	Inconsistent naming; varied descriptions; UNKNOWN values	Grouping similar light conditions	Manual mapping (light_map); UNKNOWN → NaN
7	Traffic Control	Inconsistent naming; varied descriptions; UNKNOWN values	Grouping similar traffic control types	Manual mapping (traffic_map); UNKNOWN → NaN
8	Driver Substance Abuse	Inconsistent naming; varied categories; UNKNOWN values	Consolidating various substance abuse descriptions	Manual mapping (substance_map); UNKNOWN → NaN
9	Non-Motorist Substance Abuse	Inconsistent naming; UNKNOWN values; >90% missing	Column too sparse to keep	Manual mapping + dropped column
10	Driver At Fault	Inconsistent naming ('Unknown' vs 'Not Sure')	Standardizing terminology	Replaced 'Unknown' → 'Not Sure'
11	Injury Severity	Inconsistent naming; varied descriptions; UNKNOWN values	Grouping similar injury severity levels	Manual mapping (injury_map); UNKNOWN → NaN
12	Driver Distracted By	Inconsistent naming; varied descriptions; UNKNOWN values	Consolidating distraction categories	Manual mapping (distracted_map); UNKNOWN → NaN
13	Vehicle Damage Extent	Inconsistent naming; UNKNOWN values	Grouping similar damage levels	Manual mapping (damage_map); UNKNOWN → NaN

14	Vehicle First Impact Location	Inconsistent naming; varied clock-position descriptions; UNKNOWN values	Consolidating varied impact descriptions	Manual mapping (impact_map); UNKNOWN → NaN
15	Vehicle Body Type	Inconsistent naming; diverse vehicle categories; UNKNOWN values	Grouping vehicle body types	Manual mapping (vehicle_body_map); UNKNOWN → NaN
16	Vehicle Movement	Inconsistent naming; varied descriptions; UNKNOWN values	Grouping similar vehicle movements	Manual mapping (vehicle_movement_map); UNKNOWN → NaN
17	Vehicle Year	Invalid or impossible years (0, 9999, >2025, <1950)	Standardizing invalid and out-of-range values	Manual mapping + filtering to valid range (1950–2025)
18	Vehicle Make	Inconsistent naming; typos; garbage values; UNKNOWN values	Too many variations; spelling errors; abbreviations	Extensive manual mapping (make_map); garbage + UNKNOWN → NaN
19	Vehicle Model	Highly inconsistent naming; fragments; mixed brand/model; UNKNOWN values	High cardinality and complex variations	Custom NLP + regex mapping using normalize_model_brand → Vehicle_Model_Brand
20	Latitude	Values outside Montgomery County boundaries	Identifying valid geographic range	Flagged for further filtering or imputation
21	Longitude	Values outside Montgomery County boundaries	Identifying valid geographic range	Flagged for further filtering or imputation
22	Speed Limit	Presence of 0	Ambiguity whether 0 = missing or valid	Marked as anomaly; no treatment yet
23	Location	Redundant column (duplicate of Lat/Long)	No challenge	Dropped column

24	Circumstance	Very high missingness; inconsistent free text	High cardinality; unstructured text	Replaced using NLP → Circumstance_Category; original dropped; 'Other' → NaN

NLP for Anomaly Treatment In Circumstances Column

1. The Circumstance column had highly inconsistent free-text entries with mixed casing, punctuation, abbreviations, and spelling variations.

2. Traditional cleaning methods (replace, mapping, regex) could not handle thousands of unique messy text values.

3. Many entries expressed the same meaning but used different wording (for example: “speeding”, “too fast”, “driving fast”).

4. Several rows contained multiple conditions in one line (such as “Rain, Wet Road, High Speed”), which simple methods cannot classify correctly.

5. The column had very high cardinality, making it unsuitable for modeling without reducing the number of unique values.

6. NLP (spaCy PhraseMatcher) can intelligently detect phrases instead of relying on strict string matching.

7. NLP converts unstructured text into structured and meaningful categories automatically.

8. NLP handles linguistic variation and tokenization, resulting in more accurate text cleaning.

9. NLP makes the process scalable, automatic, and repeatable instead of requiring thousands of manual mappings.

10. NLP reduces the messy text column into a small set of well-defined categories that can be used effectively for machine learning.

Column wise Null value Treatment

Column	Before Null Count	After Null Count	Treatment Strategy
Circumstance _Category	167848	0	Model-based (XGBoost) imputation using multiple related features
Vehicle Model	91246	0	Mapped from Vehicle Make, then probabilistic fill
Driver Substance Abuse	47030	0	Probabilistic fill
Driver Distracted By	43095	0	Rule-based (collision type, parked vehicle, light, speed, vehicle movement, injury severity), then probabilistic fill
Vehicle Make	41137	0	Probabilistic fill, then corrected based on Vehicle Model
Cross-Street Name	37110	0	Probabilistic fill
Traffic Control	28855	0	Probabilistic fill
Surface Condition	23930	0	Rule-based (weather, vehicle movement, light, parked vehicle, speed) then probabilistic fill
Road Name	23289	0	Probabilistic fill
Route Type	20139	0	Probabilistic fill
Weather	14270	0	Probabilistic fill
Driver's License State	13755	0	Probabilistic fill
Vehicle Going Direction	8555	0	Probabilistic fill
Agency Name	7841	0	Probabilistic fill
Vehicle Damage Extent	6936	0	Rule-based (injury severity, collision type, speed, parked vehicle, impact location, vehicle year, body type) then probabilistic fill
Vehicle Year	5258	0	Probabilistic fill + logical correction (Vehicle Year <= Crash Year)

Vehicle Movement	4335	0	Rule-based (parked vehicle, collision type, impact location, speed, light, injury severity) then probabilistic fill
Vehicle Body Type	4304	0	Rule-based (parked vehicle, collision type, speed, injury severity, light) then probabilistic fill
Vehicle First Impact Location	3238	0	Rule-based (parked vehicle, collision type, speed, injury severity, light conditions) then probabilistic fill
Light	2264	0	Rule-based (crash hour, injury severity, collision type, parked vehicle, speed) then probabilistic fill
Injury Severity	2231	0	Rule-based (parked vehicle, collision type, speed) then probabilistic fill
Collision Type	1552	0	Rule-based (parked vehicle, single vehicle, at fault, speed) then mode fill
Parked Vehicle	1526	0	Rule-based (driverless vehicle, at fault, speed, crash type) then mode fill

1. All major columns with missing values were imputed using a mix of rule-based logic, probabilistic filling, and model-based (XGBoost) prediction to ensure accurate reconstruction of missing information.

2. Columns such as Vehicle Model, Vehicle Make, Driver Substance Abuse, and Route Type were completed using probabilistic methods that preserved realistic distributions and avoided bias.

3. Several behavior-driven columns (Driver Distracted By, Surface Condition, Vehicle Movement, Damage Extent, Light, Injury Severity) were filled using domain rules linked to collision type, speed, light, parked vehicle, weather, and injury severity, followed by probabilistic refinement.

4. High-priority columns like Circumstance_Category received advanced XGBoost-based imputation, using multiple related variables to produce context-aware missing values.

5. After treatment, all columns achieved a final null count of zero, ensuring the dataset is complete, consistent, and fully ready for modeling without any missing-data bias.

Outliers Treatment

Column	Treatment Method	Explanation
Speed Limit	IQR Capping	Extreme values capped using Q1 - 1.5*IQR and Q3 + 1.5*IQR to prevent distortion in analysis and model training
Vehicle Year	IQR Capping	Unrealistic or extreme year values are capped to maintain logical and statistical consistency
Latitude	IQR Capping	Geographical outliers were capped to keep data within reasonable spatial boundaries
Longitude	IQR Capping	Longitude extremes are capped to reduce noise and ensure a valid geographic range
hour	IQR Capping	Unusual crash hours were capped to stabilise temporal patterns and prevent skewed distributions

Data Type Check

Column	Initial Datatype	Final Datatype
Report Number	object	object
Local Case Number	object	object
Agency Name	object	object
ACRS Report Type	object	object
Crash Date/Time	datetime64[ns]	datetime64[ns]
Route Type	object	object
Road Name	object	object
Cross-Street Name	object	object

Collision Type	object	object
Circumstance_Category	object	object
Weather	object	object
Surface Condition	object	object
Light	object	object
Traffic Control	object	object
Driver Substance Abuse	object	object
Person ID	object	object
Driver At Fault	object	object
Injury Severity	object	object
Driver Distracted By	object	object
Driver's License State	object	object
Vehicle ID	object	object
Vehicle Damage Extent	object	object
Vehicle First Impact Location	object	object
Vehicle Body Type	object	object
Vehicle Movement	object	object
Vehicle Going Direction	object	object
Speed Limit	int64	float64
Driverless Vehicle	object	object
Parked Vehicle	object	object
Vehicle Year	float64	int64
Vehicle Make	object	object
Latitude	float64	float64

Longitude	float64	float64
Vehicle Model	object	object
hour	Int32	Int32

Sanity Check & Core Data Validation Results

Q No.	Main Question	Python / SQL Result Summary	Key Inference
Q1	Validate total crash records vs distinct Report Numbers	205,485 total records, 115,833 distinct report numbers	Multiple records per report confirm one-to-many structure (vehicles/persons per crash)
Q2	Count crashes by Agency Name	Montgomery County Police dominates (≈184k records)	Crash volume is highly skewed toward one agency
Q3	Monthly crash counts	Monthly counts range 15k–20k, no extreme spikes	Crash occurrence is seasonally stable
Q4	Extract year, month, day, hour	New columns successfully derived	Temporal features ready for time-based analysis
Q5	Missing Vehicle ID / Person ID	No missing values	Referential integrity maintained
Q6	NULL checks for key columns	No missing values	Dataset is analysis-ready
Q7	Validate geographic ranges	Latitude 38.85–39.31, Longitude −77.41 to −76.81	Coordinates fall within valid geographic bounds
Q8	Injury + Fatal crash view	187 fatal crash records identified	Critical subset isolated for severity modeling
Q9	Count crashes by Collision Type	Rear-End (Same Direction) most frequent	Traffic flow & following distance are major contributors
Q10	Missing / zero Speed Limit	No missing or zero values	Speed features are reliable for modeling
Q11	Rank top crash-prone roads	Georgia Ave, New Hampshire Ave top	Crashes are spatially concentrated, not random
Q12	Weather vs Surface Condition	Most crashes in clear & dry; risk increases in wet/icy	Volume ≠ risk; context matters
Q13	Vehicle not at scene	None detected	No abnormal reporting patterns
Q14	Alcohol / drug related crashes	≈9% drug-related	Behavioral factors play a significant role

Q15	Hotspot detection (Lat–Long)	Clusters identified with few high-risk zones	Justifies geospatial clustering model
Q16	Vehicle year validation	No invalid years found	Vehicle metadata is clean
Q17	Night-time crash extraction	≈35% crashes at night	Visibility & fatigue are key risk drivers
Q18	Weekday / weekend / hourly patterns	Weekdays peak at rush hours; weekends late-night	Temporal risk varies by day context
Q19	Missing coordinates	No missing coordinates	Safe for mapping & Tableau
Q20	Cleaned temporary table	Successfully created	Enables performance-optimized queries
Q21	Index creation	Indexes applied	Improves query and join efficiency
Q22	Join crash, person, vehicle tables	Successful joins	Full relational view achieved
Q23	Distinct categorical counts	5 agencies, 4704 roads, 17 collision types, etc.	High cardinality handled intentionally
Q24	Speed limit vs route type	High-speed routes show higher severity	Confirms logical consistency
Q25	Average crashes per road segment	Computed successfully	Supports road-level risk ranking
Q26	Rare collision types	Same-direction left-turn crashes noted	Rare patterns retained for completeness
Q27	Weather–surface inconsistencies	Most combinations are rare	Edge cases preserved, not removed
Q28	Severity vs road condition	Dry & Wet dominate volume	High exposure conditions drive counts
Q29	Vehicles involved in crashes	Toyota Camry most frequent	Reflects fleet composition, not fault

Deep Analysis:

5 Point Summary for Numeric Columns:

	count	mean	std	min	25%	50%	75%	max
Speed Limit	205539.0	32.281100	11.052648	2.500000	25.000000	35.000000	40.000000	62.500000
Vehicle Year	205539.0	2011.361868	6.479980	1993.000000	2007.000000	2012.000000	2016.000000	2025.000000
Latitude	205539.0	39.082942	0.070796	38.851557	39.024418	39.074846	39.139659	39.312521
Longitude	205539.0	-77.111865	0.094728	-77.414384	-77.189266	-77.105170	-77.039188	-76.814070
hour	205539.0	13.365274	5.261188	0.000000	9.000000	14.000000	17.000000	23.000000
Crash_year	205539.0	2019.710449	3.167699	2015.000000	2017.000000	2019.000000	2023.000000	2025.000000
Crash_month	205539.0	6.639810	3.439433	1.000000	4.000000	7.000000	10.000000	12.000000
Crash_day	205539.0	15.695357	8.769084	1.000000	8.000000	16.000000	23.000000	31.000000
Crash_hour	205539.0	13.365274	5.261188	0.000000	9.000000	14.000000	17.000000	23.000000
Crash_week	205539.0	27.102633	14.993967	1.000000	14.000000	27.000000	40.000000	53.000000

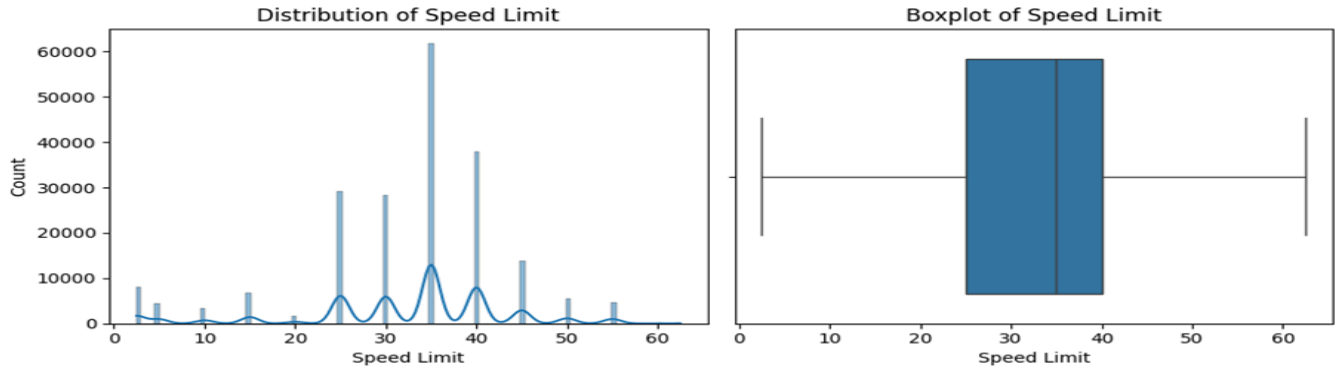
Feature	Key Insights
Speed Limit	The average speed limit is 32 mph, with most roads between 25–40 mph. Values range from 2.5 to 62.5, indicating the presence of alleys and highways.
Vehicle Year	Typical vehicle year is 2011–2016. The range spans from 1993 to 2025, showing a mix of older and newer vehicles.
Latitude	Values cluster tightly around 39.08, confirming the dataset covers a geographically concentrated area.
Longitude	Centred around –77.11, again showing a small geographic spread, consistent with a localised region.
Hour	Crashes occur mostly from 9 AM to 5 PM, median at 14:00, with full coverage from 0–23 hours.

Key Inferences from the categoric Summary table

Feature	Key Insights
Report Number / Local Case Number	Extremely high uniqueness (each ≈ one record), used mainly as identifiers.
Agency Name	Dominated by **Montgomery_County_Police** (≈184k cases), meaning the dataset is heavily county-focused.
ACRS Report Type	Majority are **Property Damage Crash** (≈131k), indicating most crashes are non-injury.
Route Type	**Maryland_State_Route** is most common (≈98k), showing crashes mainly occur on state routes.
Road Name / Cross-Street Name	Very high variability (4,700+ roads, 7,400+ cross streets). **GEORGIA AVE** appears most often—main crash hotspot.
Collision Type	**Rear-End (Same Direction)** leads (≈67k), typical of urban traffic congestion.
Circumstance_Category	**Road Obstruction** is top factor, influencing ≈71k crashes.
Weather / Surface Condition / Light	Majority under **Clear**, **Dry**, and **Daylight**—crashes mostly occur under good conditions.
Traffic Control	**No Control** most frequent (≈98k), indicating many crashes occur in uncontrolled zones.
Driver Substance Abuse	**None Detected** dominates (≈196k), showing low reported impairment.
Driver At Fault	**Yes** is the majority (≈103k).
Injury Severity	Mostly **No Apparent Injury** (≈167k), matching the high property-damage share.
Driver Distracted By	Mostly **Not Distracted**, suggesting distractions are underreported.
Drivers License State	Primarily **MD** (≈181k), confirming local drivers dominate.
Vehicle ID / Person ID	Fully unique identifiers—one value per record.
Vehicle Damage Extent	**Disabling** damage is most common (≈80k).
Vehicle First Impact Location	**Twelve O Clock** (front impact) is highest (~79k).
Vehicle Body Type	Mostly **PassengerCar** (~144k), typical for urban environments.
Vehicle Movement	**MovingConstantSpeed** highest (~81k), indicating many rear-end or flow-related collisions.
Vehicle Going Dir	**North** is the most common direction (~49k).
Driverless Vehicle / Parked Vehicle	Nearly all records are **No**, meaning active, non-parked vehicles dominate.
Vehicle Make	**TOYOTA** is the most common brand (~48k).
Vehicle Model	Highly diverse (450 models), with **Camry_Toyota** most frequent (~27k).

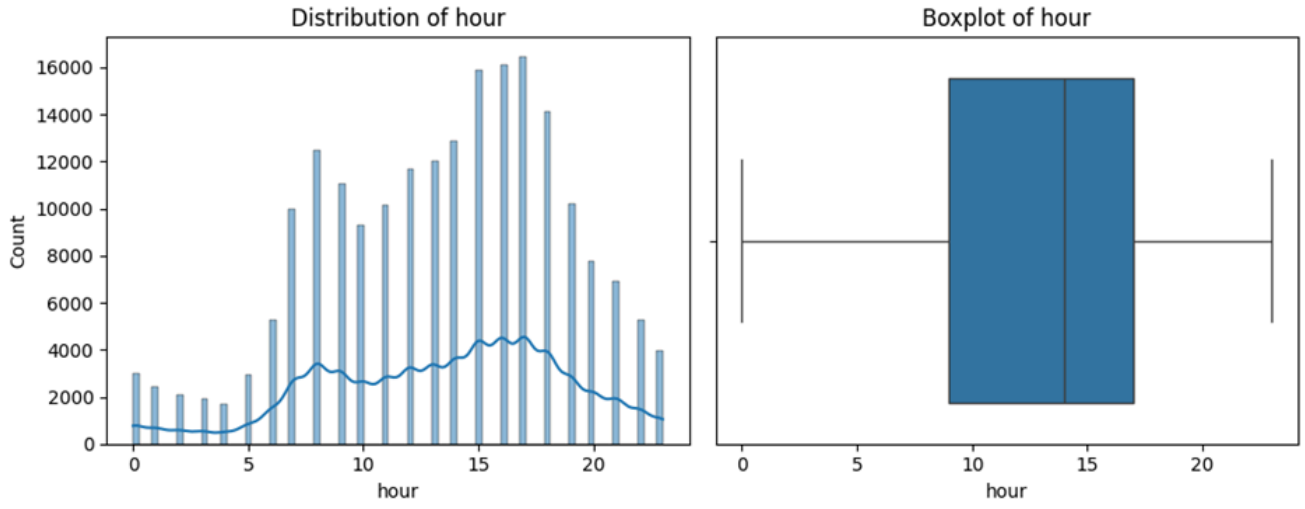
Univariate Analysis (Numeric)

Speed Limit:



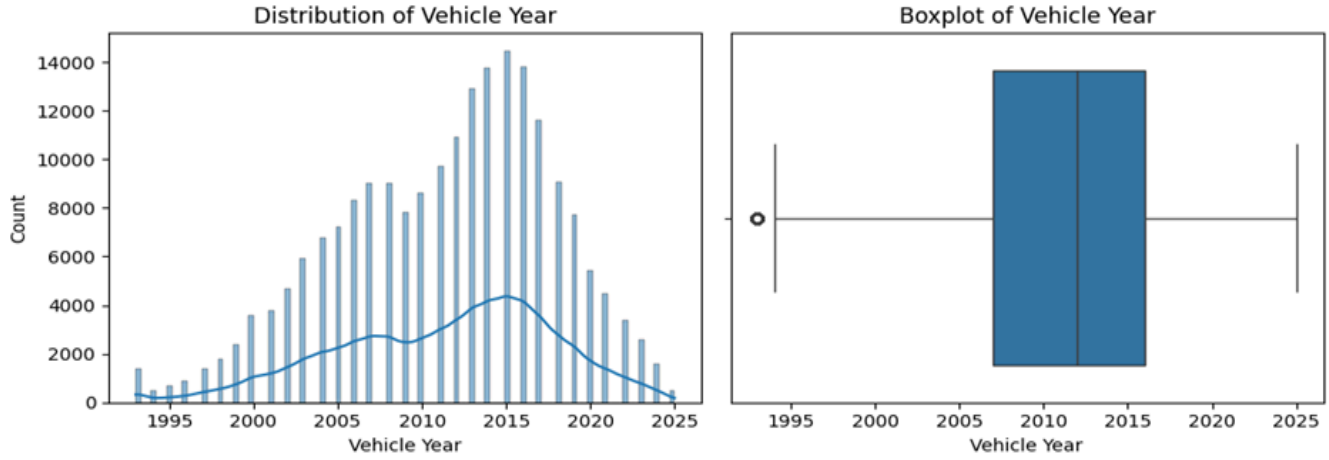
- The average speed limit is 32 mph, with most roads between 25–40 mph.
- Values range from 2.5 to 62.5, indicating presence of alleys as well as highways.

Hour:



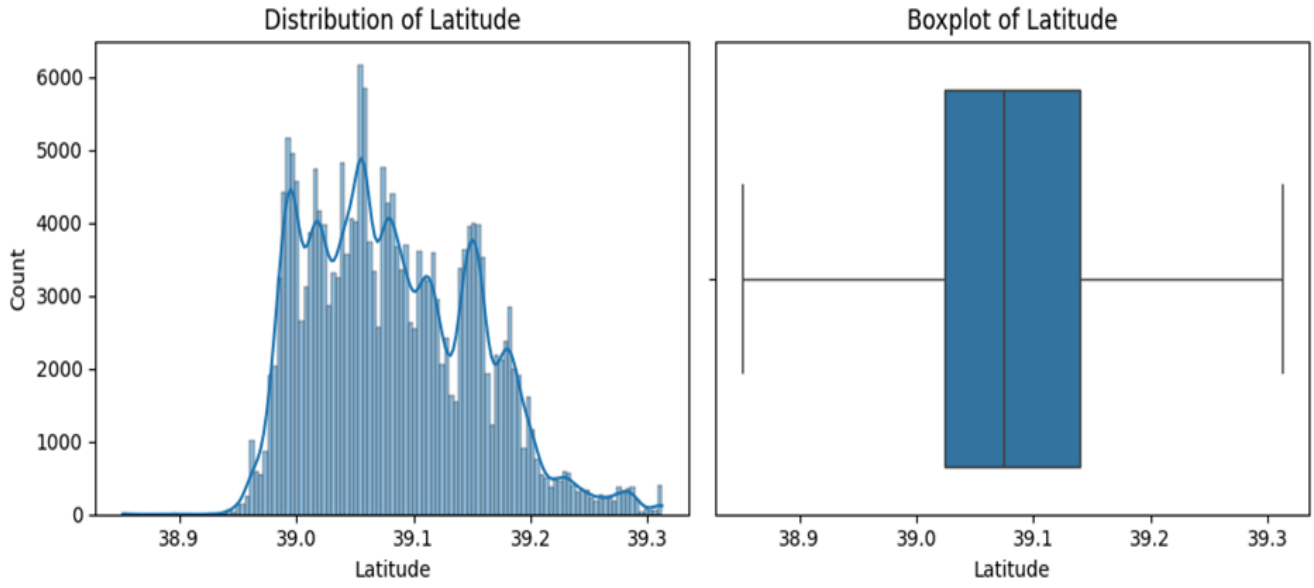
- Crashes occur mostly during 9 AM to 5 PM, with a median at 14:00, and full hourly coverage from 0–23 hours.

Vehicle Year:



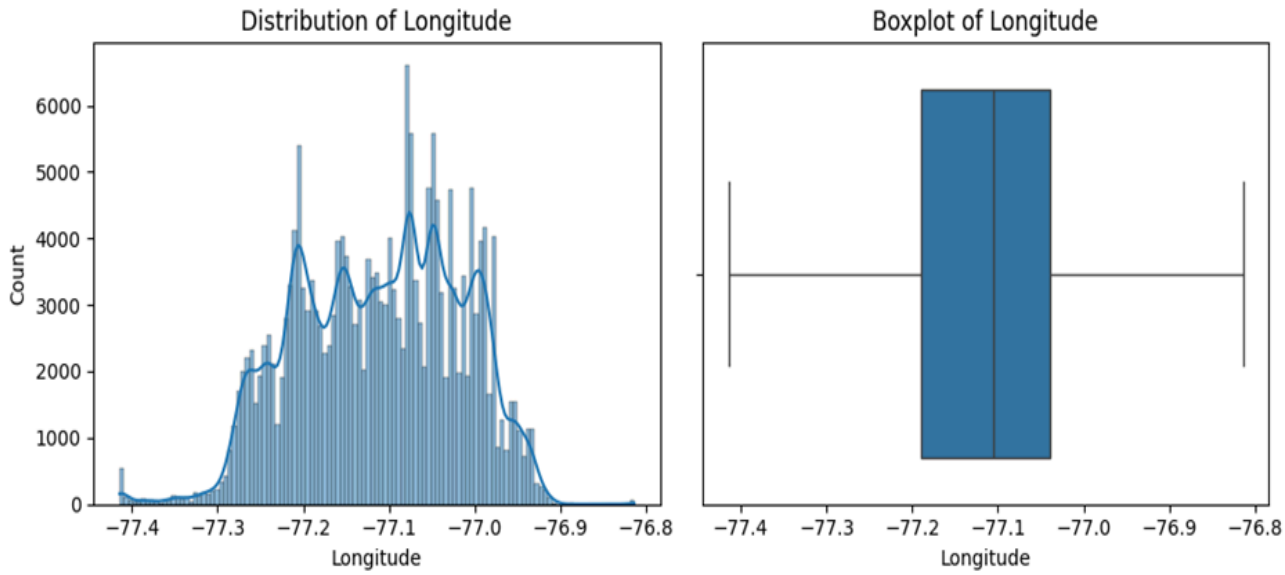
- Typical vehicle year is 2011–2016.
- Range spans from 1993 to 2025, showing a mix of older and newer vehicles.

Latitude:



- Values cluster tightly around 39.08, confirming the dataset covers a geographically concentrated area.

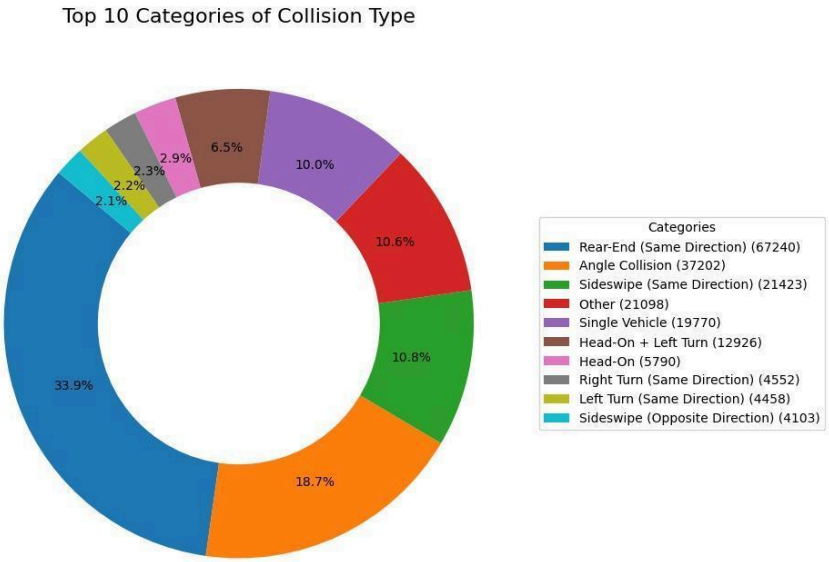
Longitude:



- Centered around –77.11, again showing a small geographic spread, consistent with a localized region.

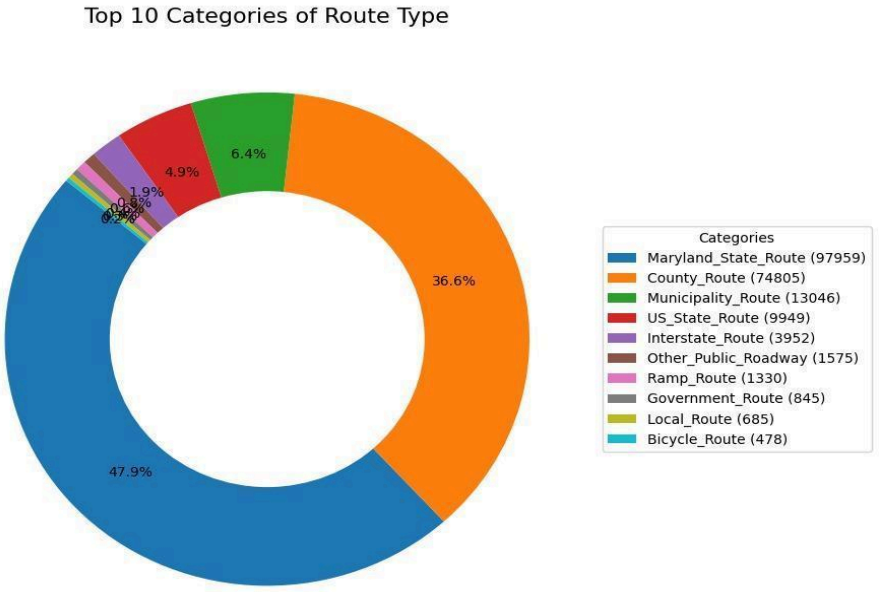
Univariate Analysis (Categoric)

Collision Type:



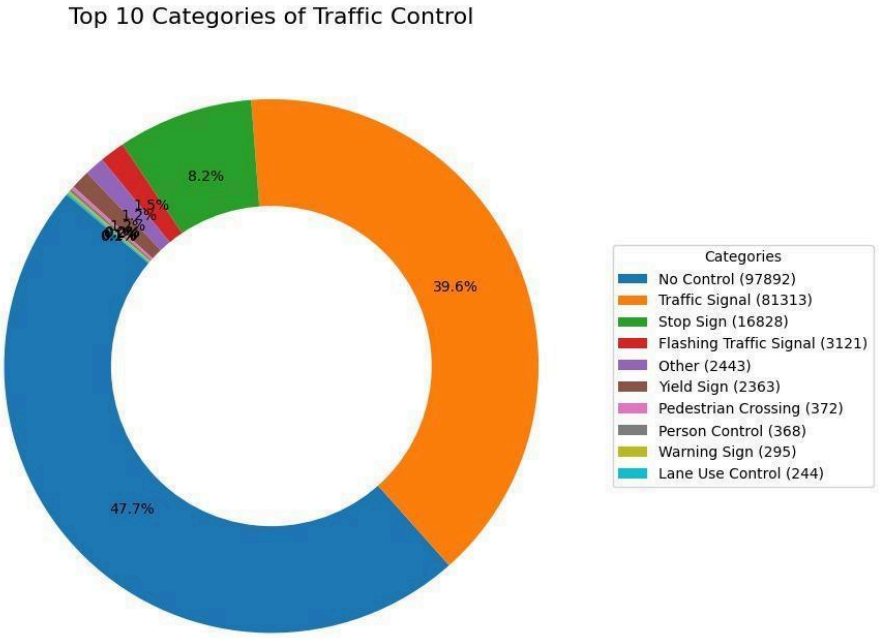
- Rear-end (same direction) collisions dominate the dataset, indicating congestion-driven crashes and frequent sudden-stop scenarios on major routes.

Route Type:



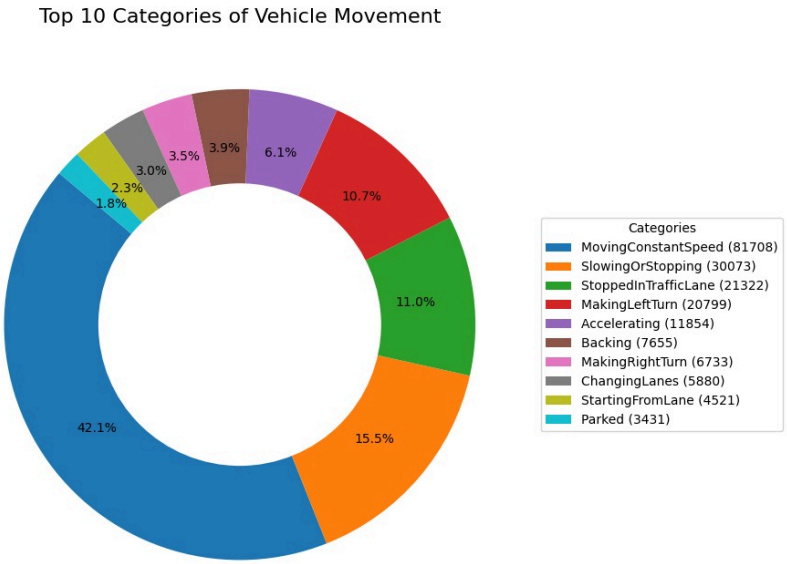
- Maryland State Routes account for the highest number of crashes, suggesting that state-maintained corridors carry heavier traffic volumes and risk.

Traffic Control:



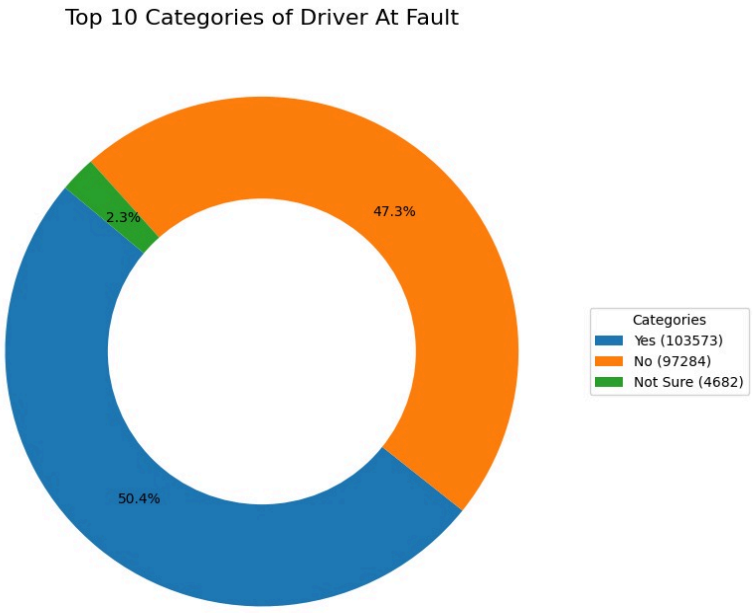
- A large portion of crashes occur at locations with no control (no signal or stop sign), highlighting the increased risk at uncontrolled intersections and open road segments.

Vehicle Movement:



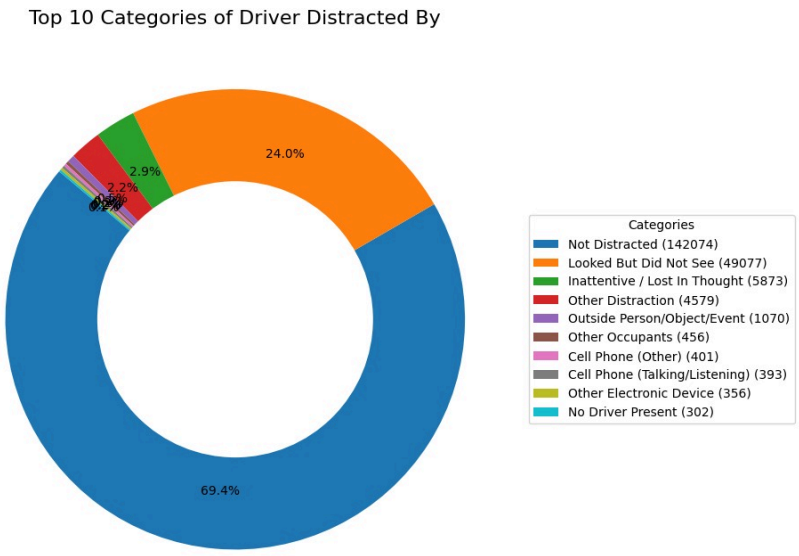
- Most vehicles were moving at constant speed before impact, indicating that many crashes occur during regular traffic flow rather than during sharp turns or sudden maneuvers.

Driver At Fault:



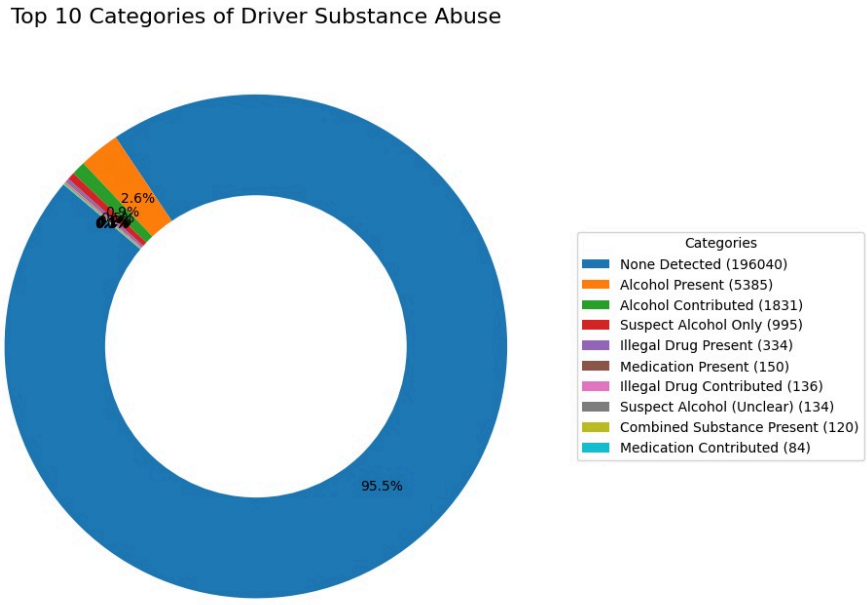
- The majority of records show the driver was at fault, confirming that human error remains the primary contributor to most crash events.

Driver Distracted By:



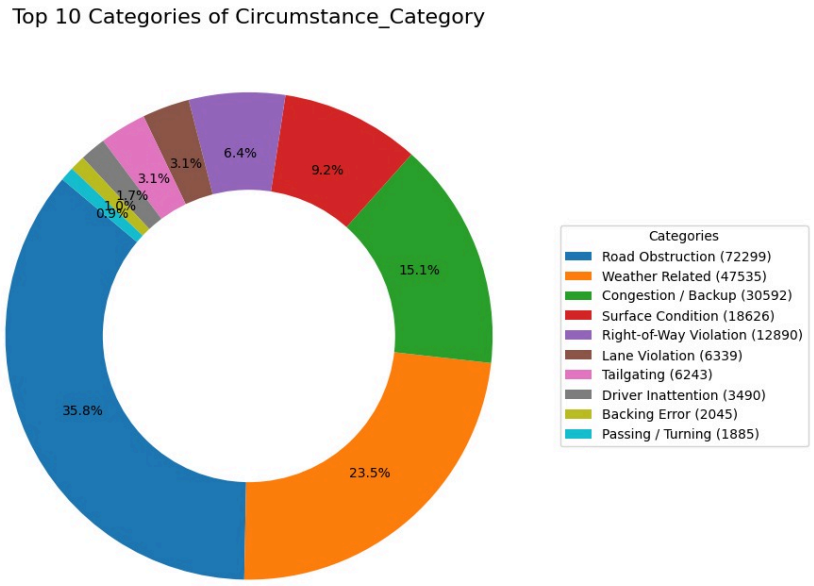
- "Not Distracted" appears most frequently, though likely underreported. Valid distraction cases (phone, passengers, etc.) exist but at much lower frequencies.

Driver Substance Abuse:



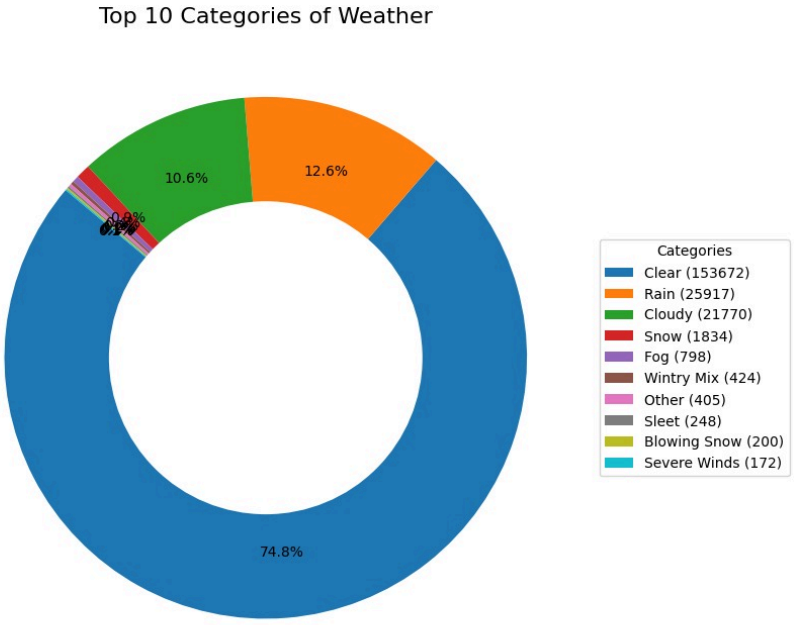
- Substance abuse incidents are rare, with "None Detected" dominating. This suggests impairment is not a major contributing factor in most recorded crashes.

Circumstance_Category:



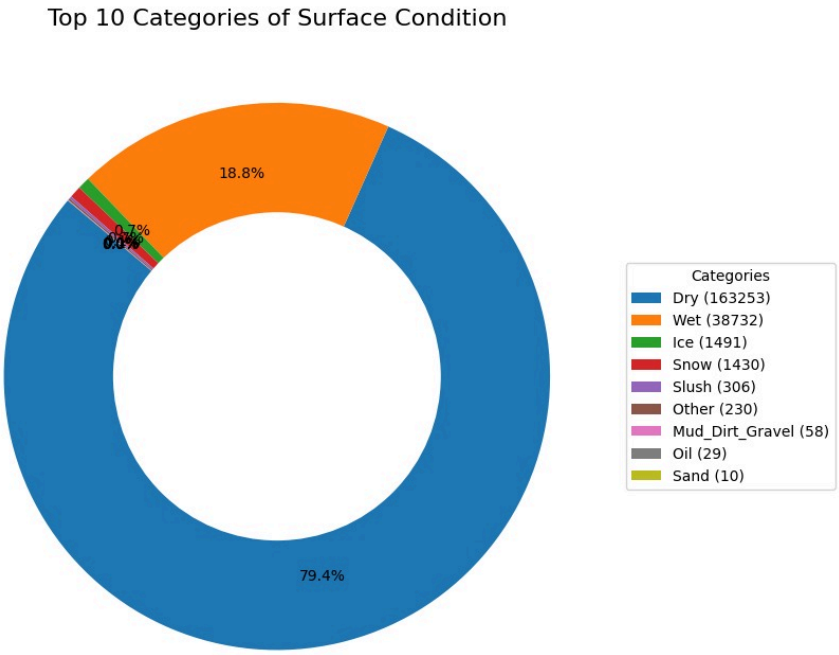
- Behavior-linked causes such as speeding, improper lane behavior, and road obstruction are highly prevalent, revealing strong behavioral influence on crash occurrence.

Weather:



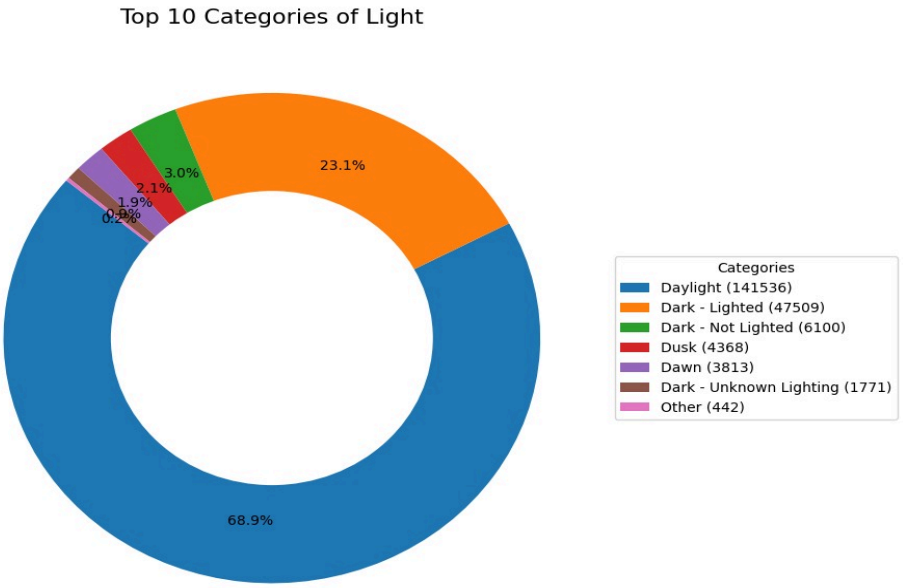
- Most crashes occur under clear weather conditions, indicating environmental severity is not a primary driver of incidents in this dataset.

Surface Condition:



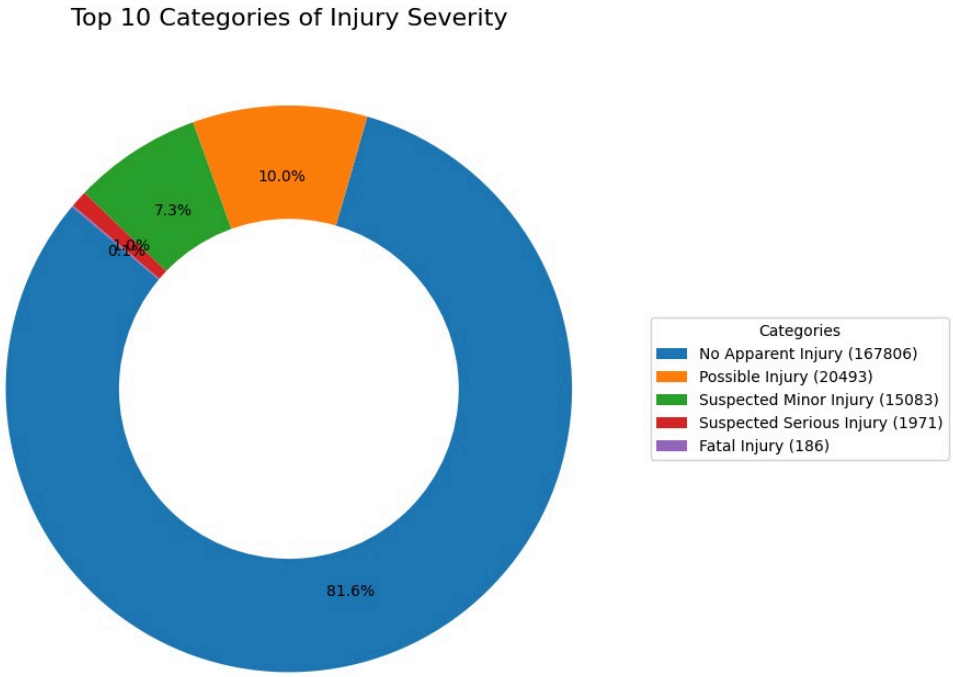
- Crashes mainly occur on dry road surfaces, reinforcing that road-surface hazards like ice or rain are not dominant crash factors.

Light:



- Daylight conditions show the highest crash counts, consistent with peak traffic hours and high vehicle density rather than low-visibility effects.

Injury Severity:

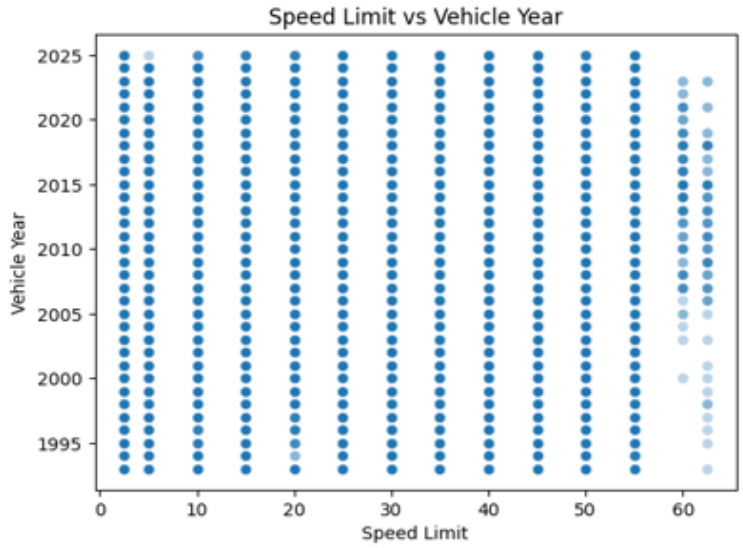


- Most crashes result in no apparent injury or minor injury, aligning with the high proportion of property-damage-only events in the dataset.

Bivariate Analysis

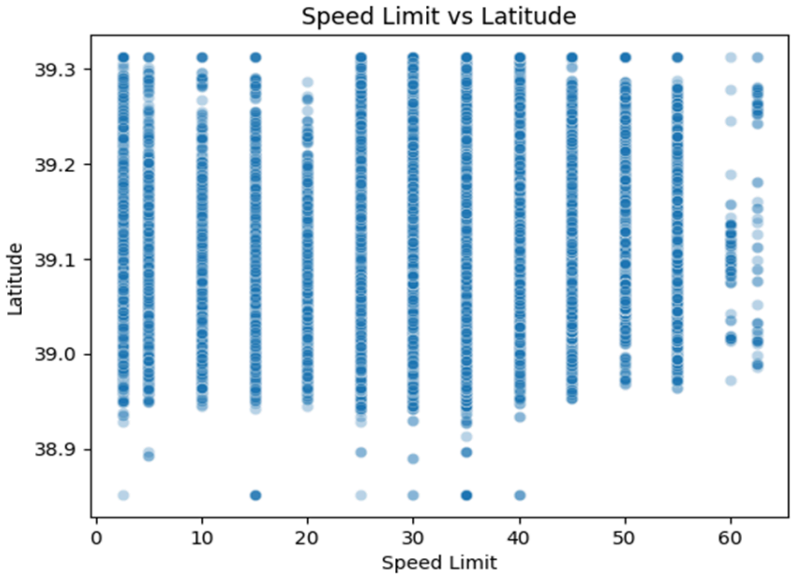
Numeric vs Numeric

Speed Limit vs Vehicle Year:



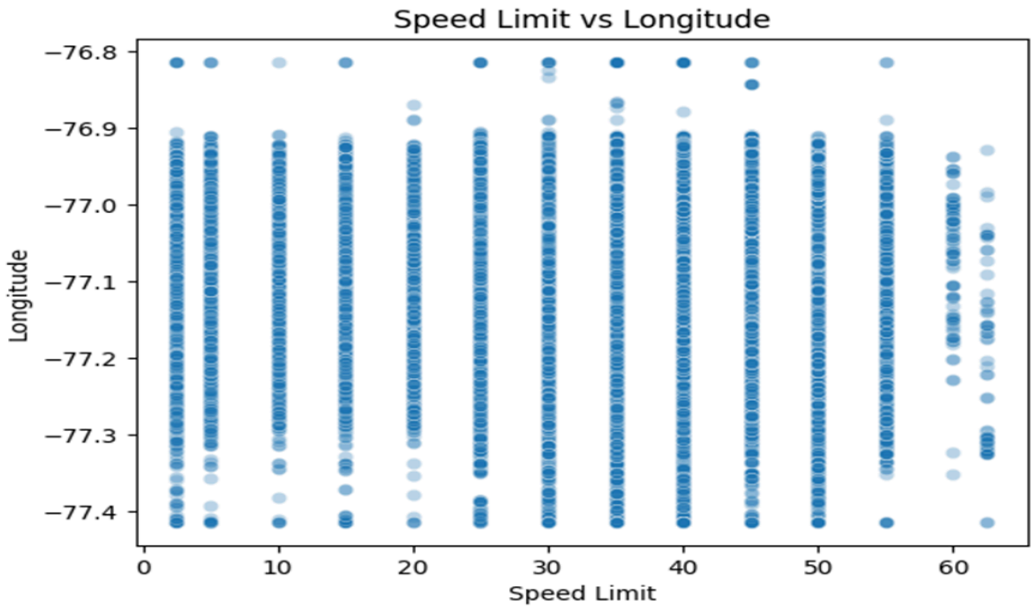
- Shows no meaningful correlation. Vehicle age does not influence the posted speed limit at crash locations, indicating these features are independent for modeling purposes.

Speed Limit vs Latitude:



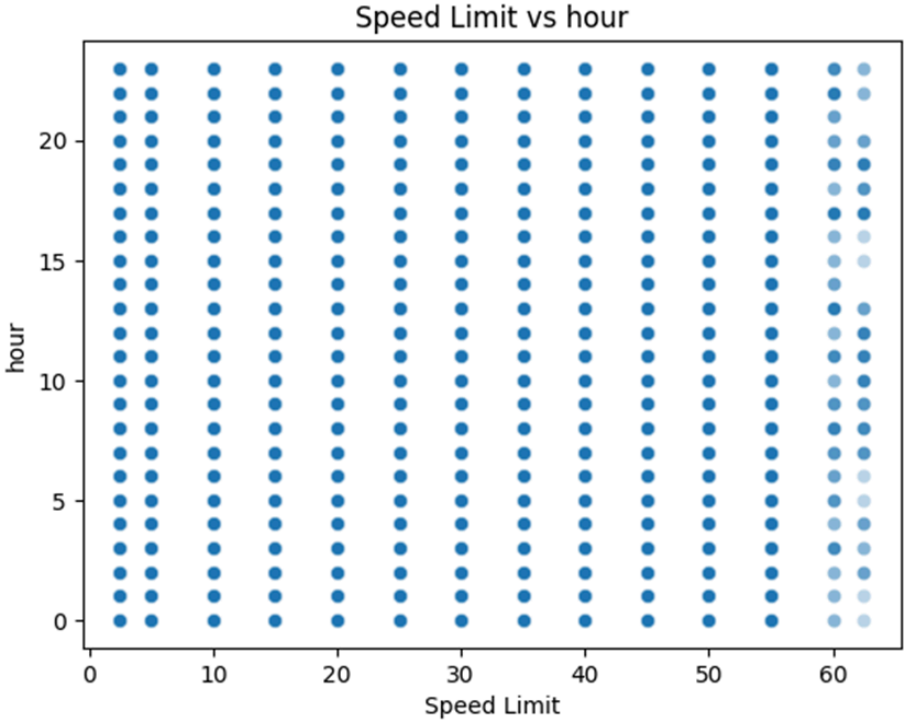
- Very weak positive correlation. Geographic north–south position does not predict speed limits reliably. Useful to confirm that location features do not distort model prediction.

Speed Limit vs Longitude:



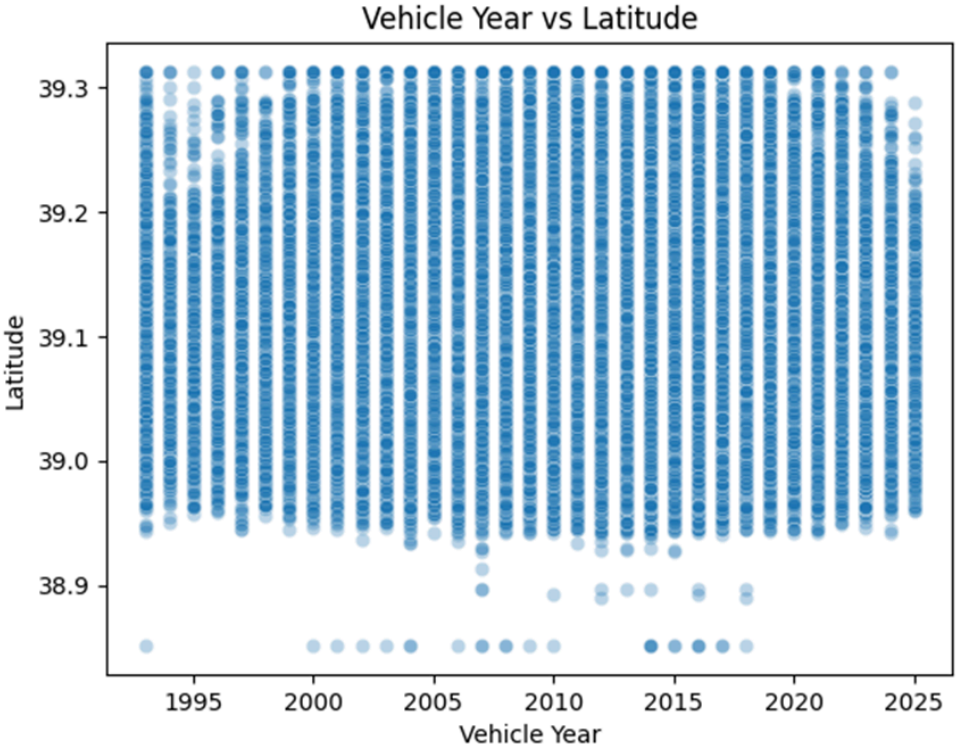
- No linear relationship. East–west position has almost zero impact on speed limits, supporting geographic independence in numeric modeling.

Speed Limit vs Hour:



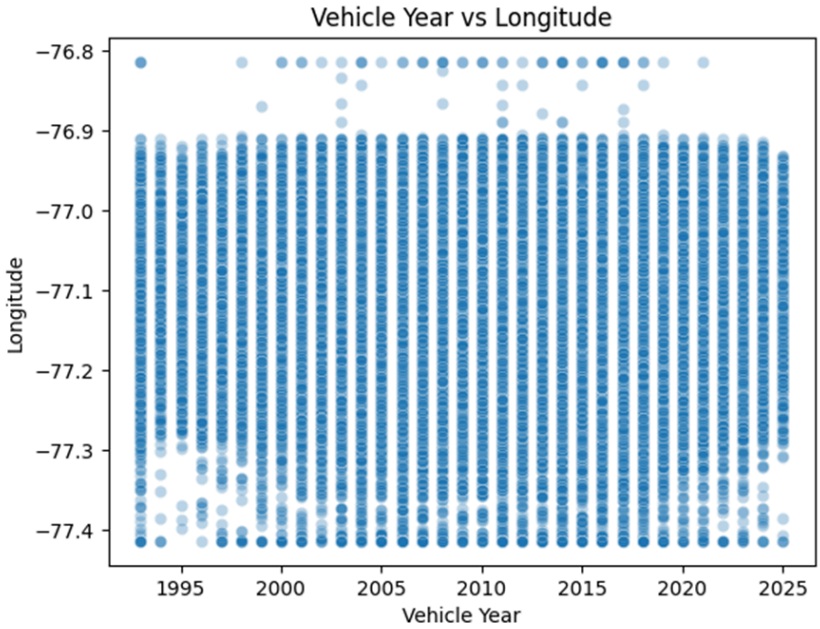
- No significant correlation. Time of day does not affect speed-limit distribution, meaning hour and speed limit can be treated as separate predictors.

Vehicle Year vs Latitude:



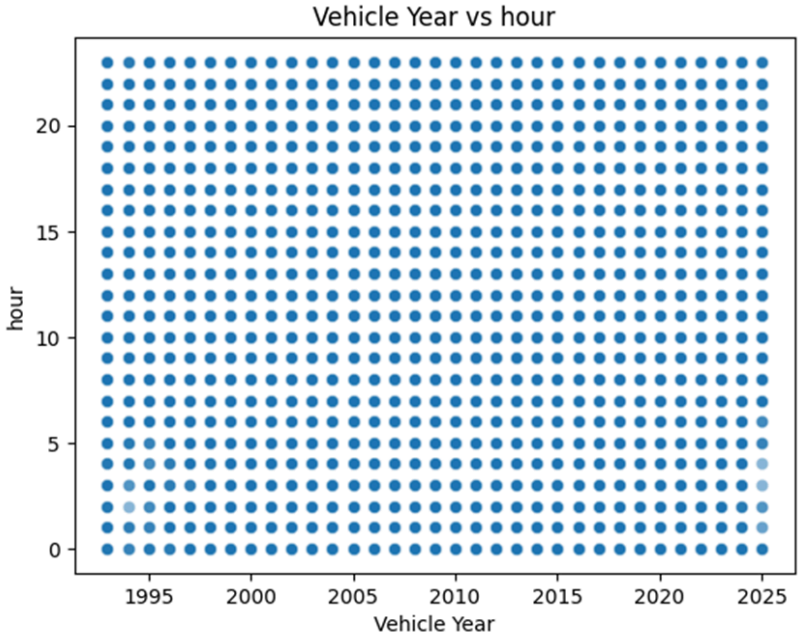
- No correlation. Vehicle age does not vary by geographic north–south distribution in the county.

Vehicle Year vs Longitude:



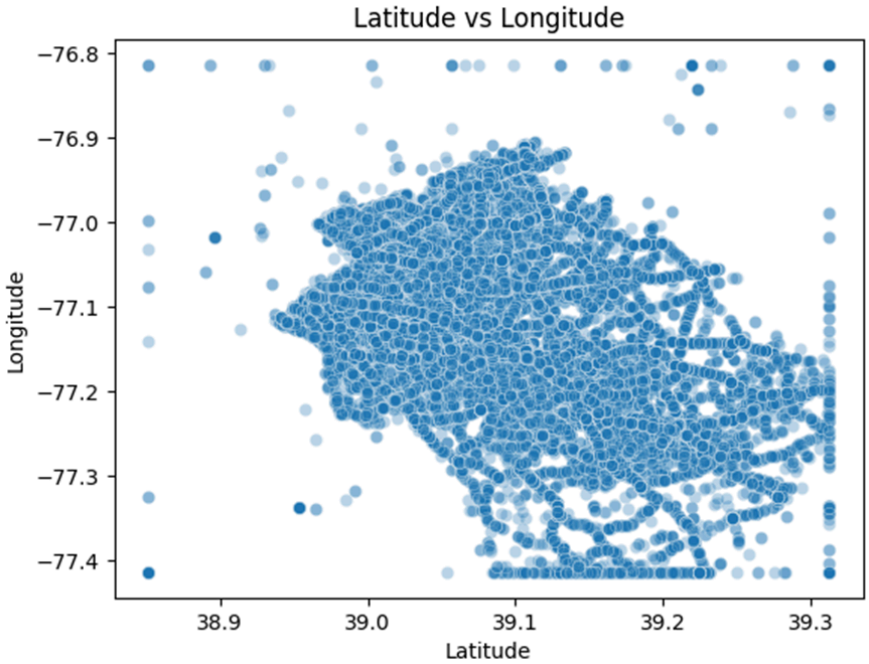
- No correlation. Vehicle year is independent of east–west crash positions.

Vehicle Year vs Hour:



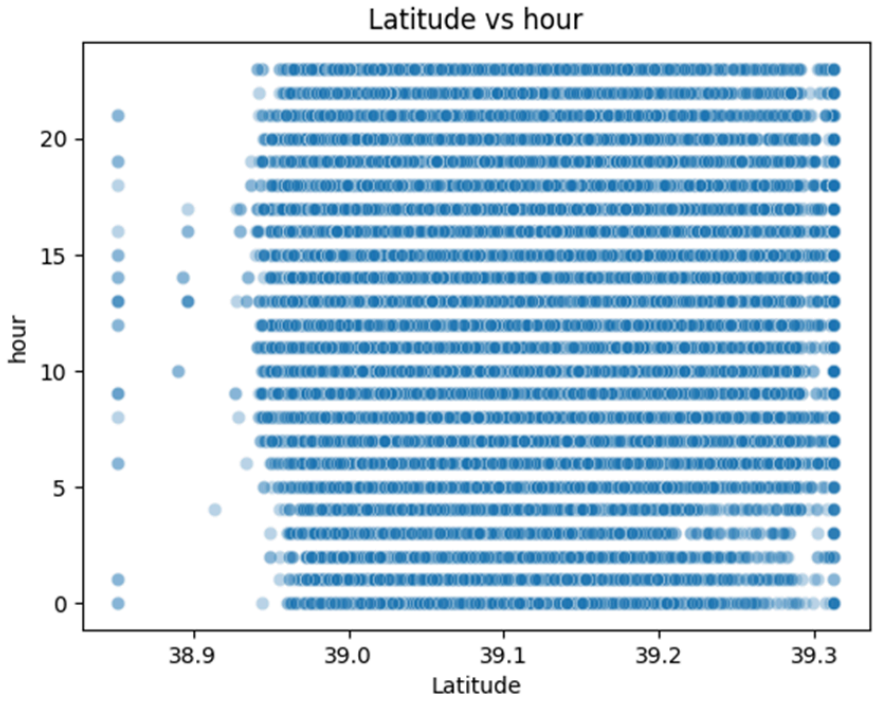
- No relationship. Crash timing does not depend on whether a vehicle is older or newer, confirming independence.

Latitude vs Longitude:



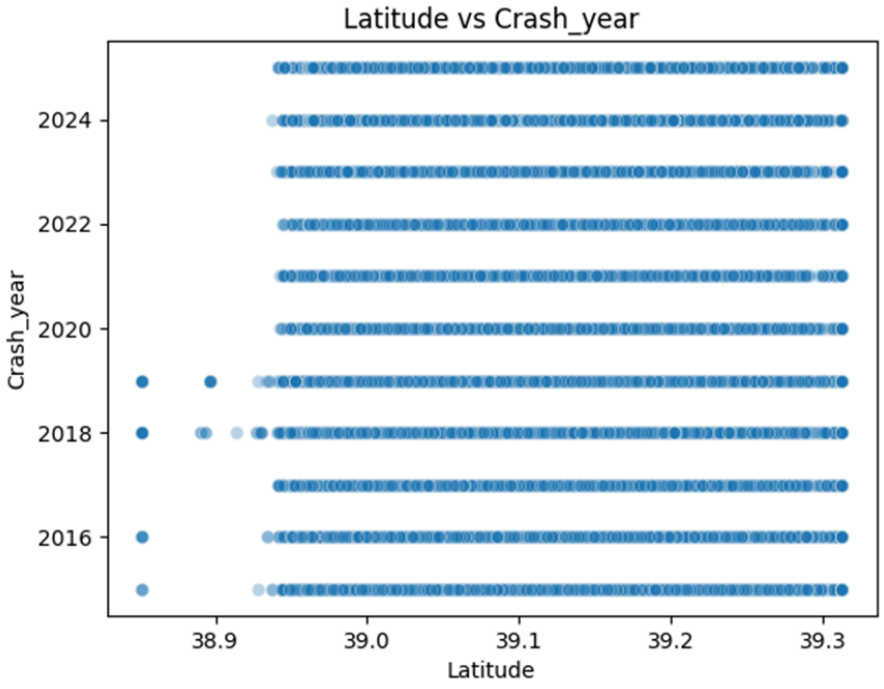
- Moderate negative correlation (-0.63). This spatial relationship confirms the dataset is geographically concentrated and consistent with expected county-level mapping patterns. Useful only for geospatial clustering, not severity modeling.

Latitude vs Hour:



- No correlation. Crash time has no linear relation with geographical north–south location.

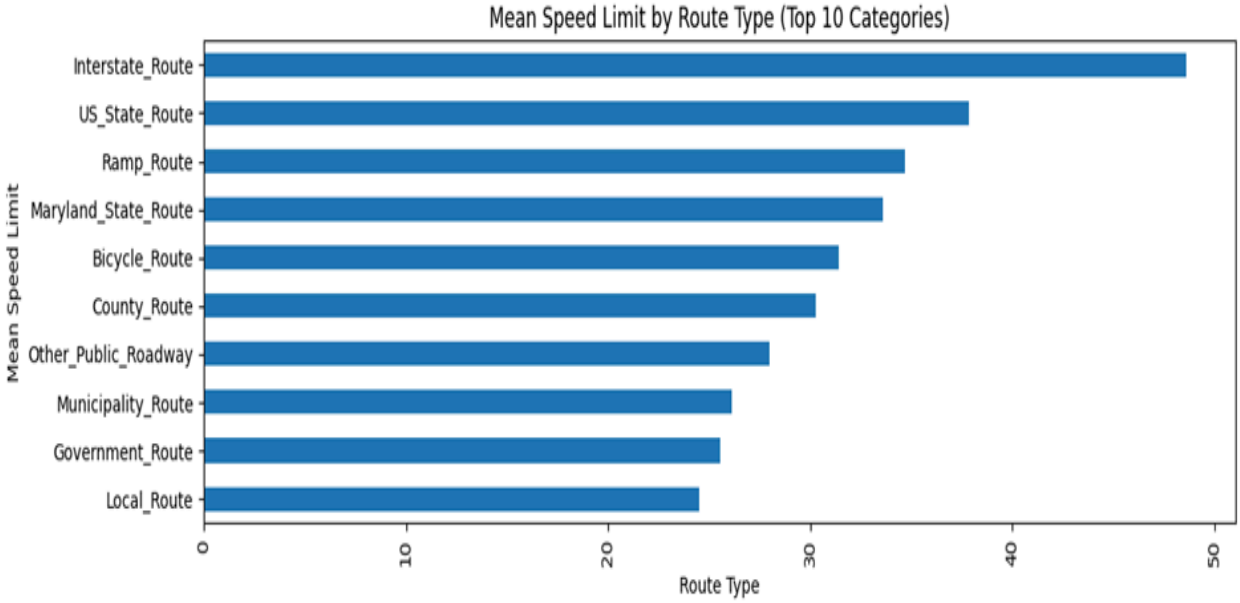
Longitude vs Hour:



- No correlation. Crash time is independent of geographical east–west location.

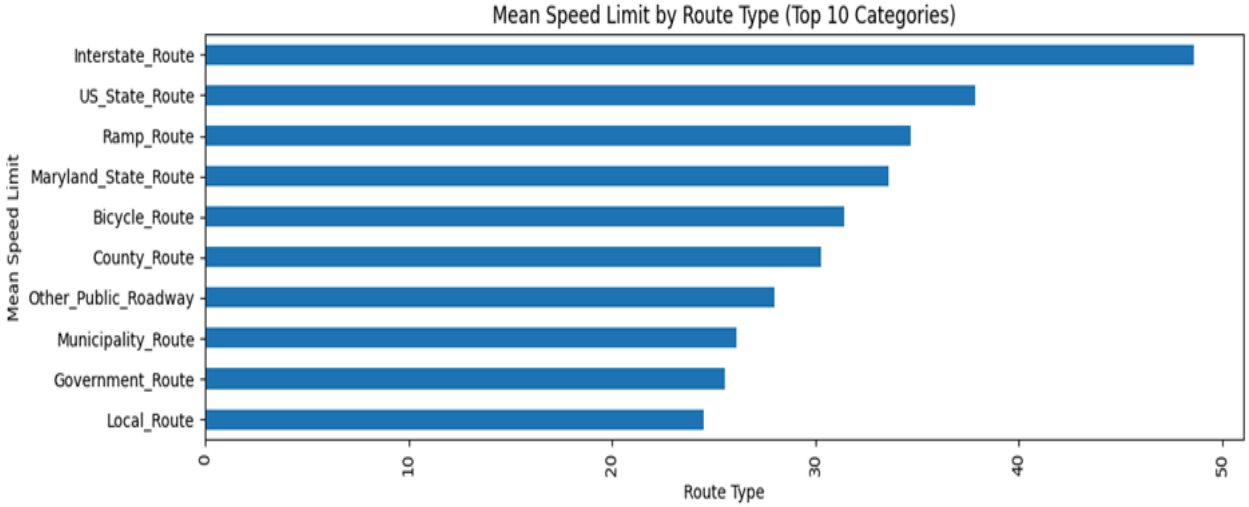
Numeric Vs Categorical

Speed Limit vs. Injury Severity:



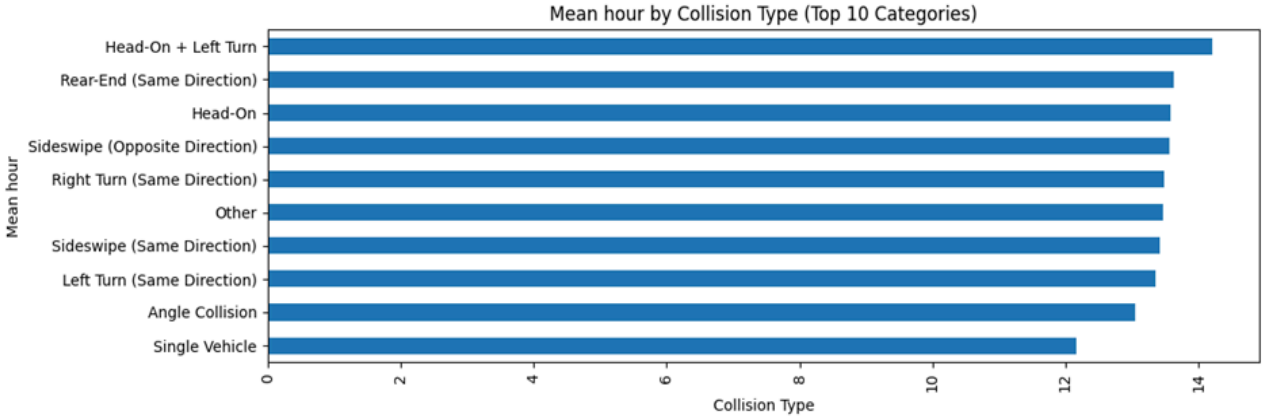
- Higher speed limits are associated with more severe crash outcomes, while lower speed zones show more minor or no-injury cases.

Speed Limit vs. Route Type:



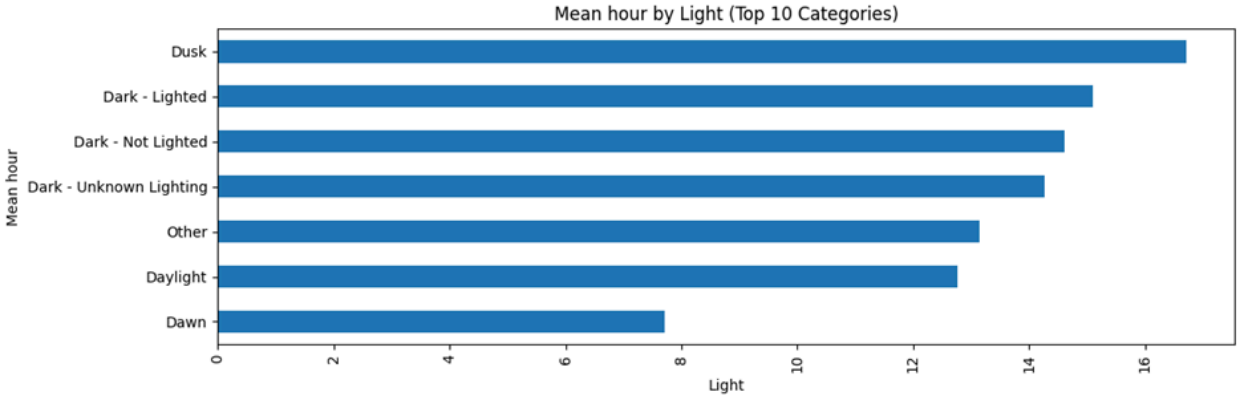
- High-speed environments (Interstate, US Route) show larger speed limits, whereas Local Streets and Alleys reflect low-speed crash zones.

Hour vs. Collision Type:



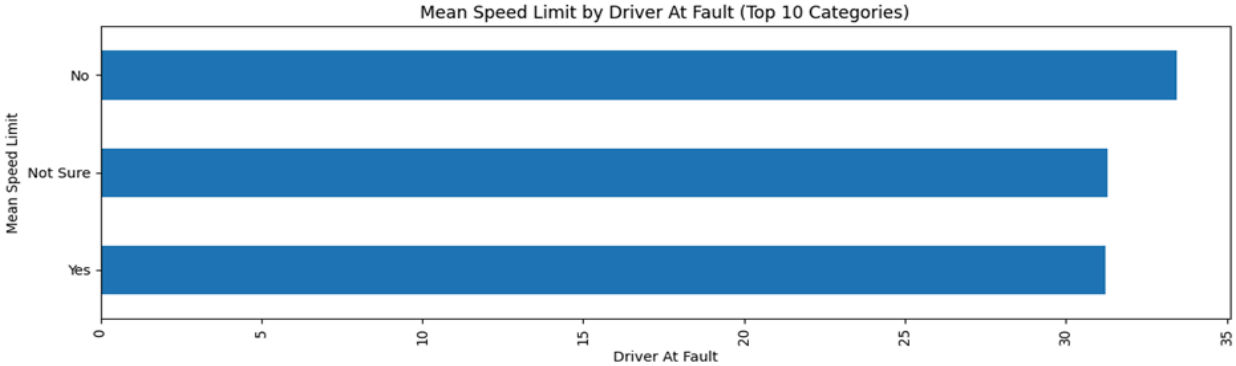
- Rear-end crashes peak during busy commute hours, whereas fixed-object and parked-vehicle crashes are more evenly distributed across the day.

Hour vs. Light:



- Daylight crashes cluster between morning and early evening, while dark-condition crashes dominate late-night and early-morning hours.

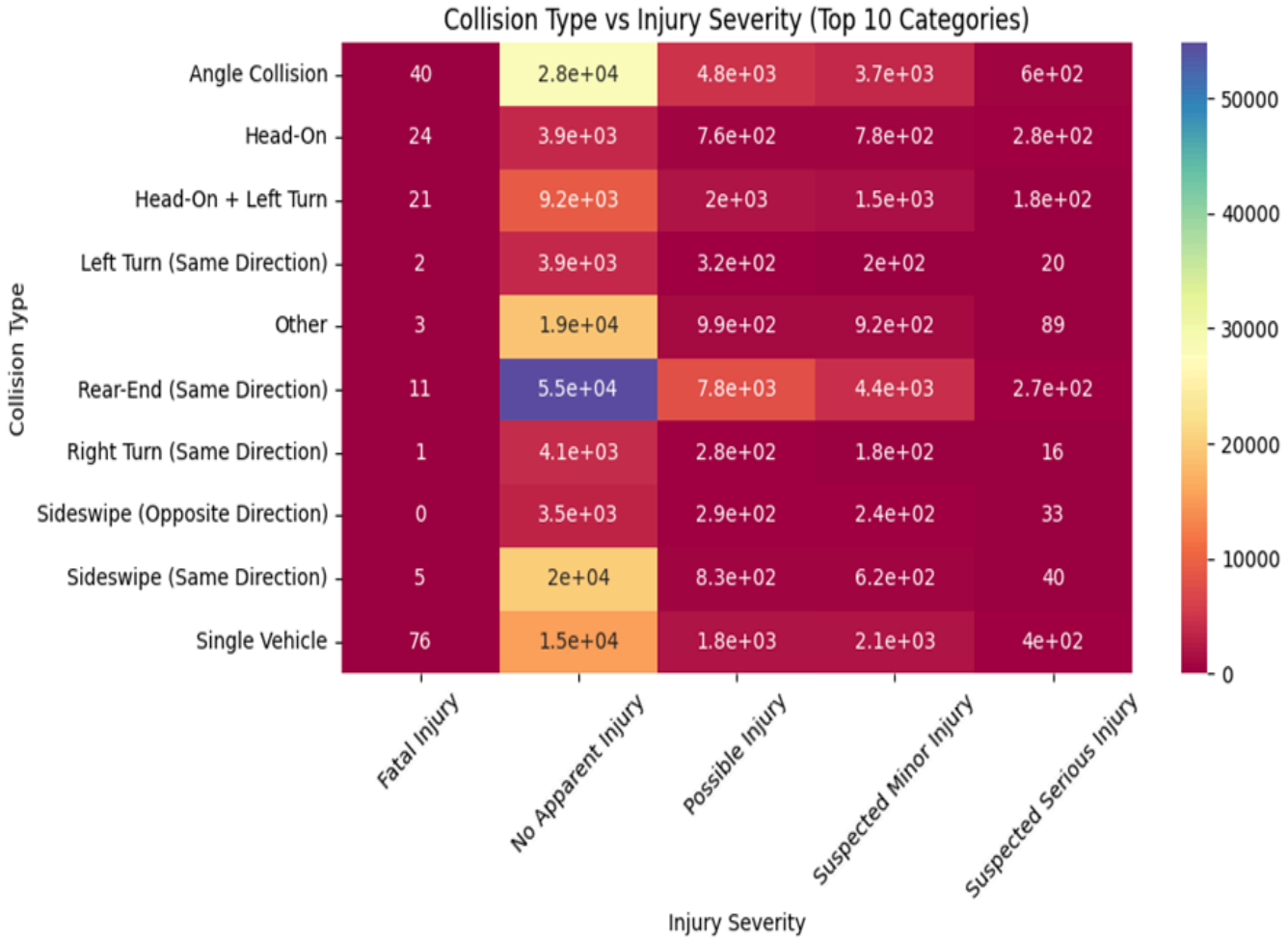
Speed Limit vs. Traffic Control:



- Uncontrolled areas show lower speeds, while signalized or major corridor locations show consistently higher limits, reflecting road design.

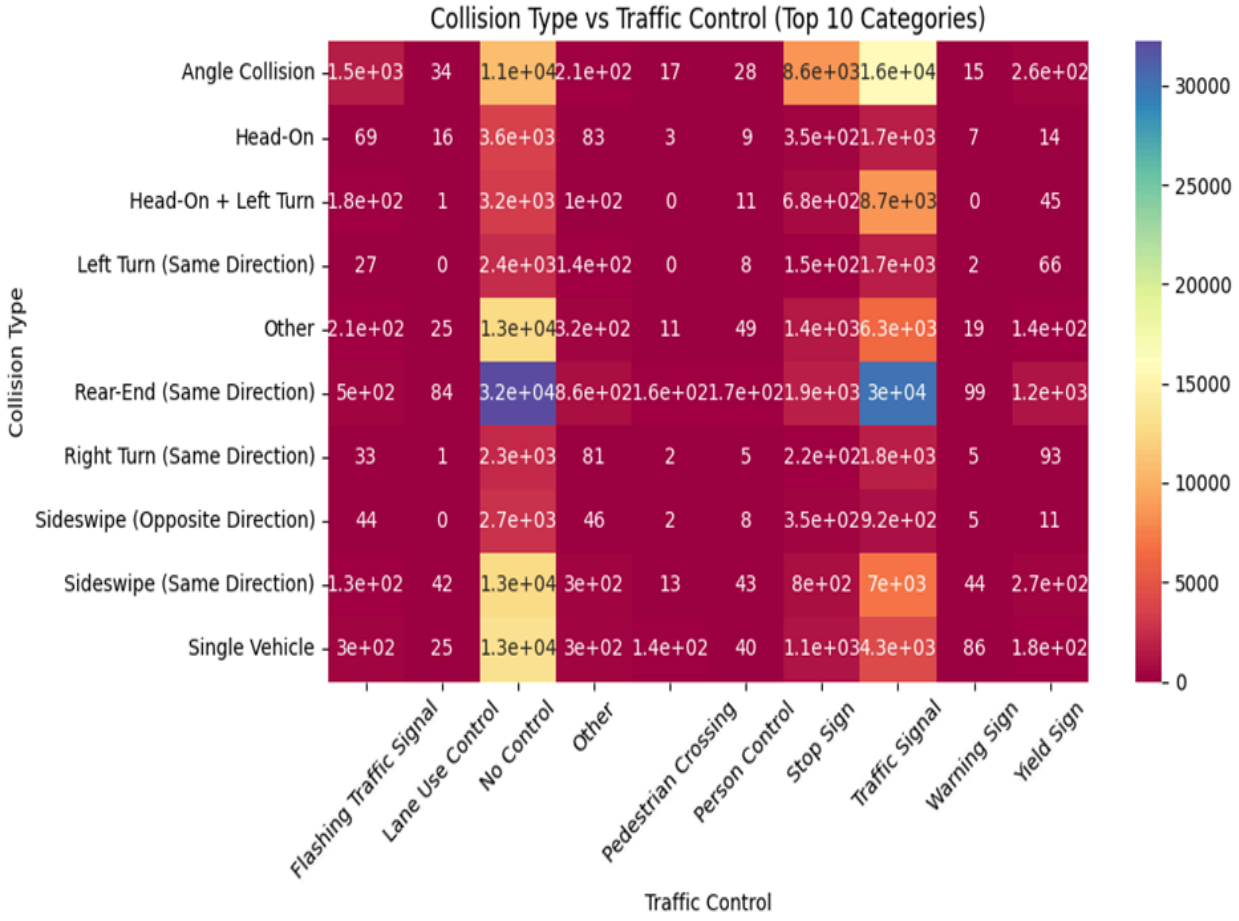
Categorical Vs Categorical

Collision Type vs. Injury Severity:



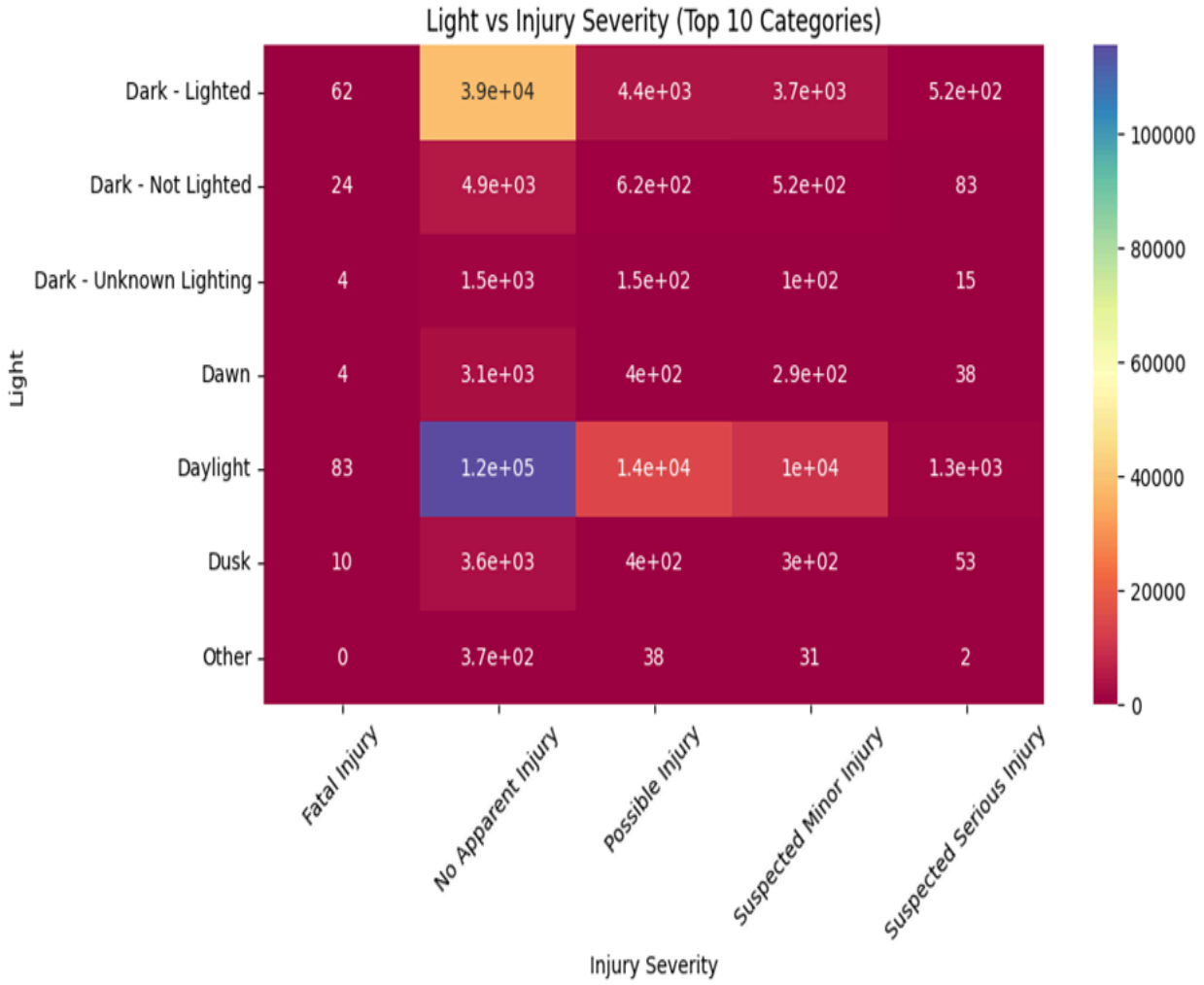
- Rear-end collisions are the most common and usually lead to no or minor injuries, while Angle and Head-On collisions show a significantly higher proportion of serious and fatal outcomes, highlighting the greater severity associated with directional impacts.

Traffic Control vs. Collision Type:



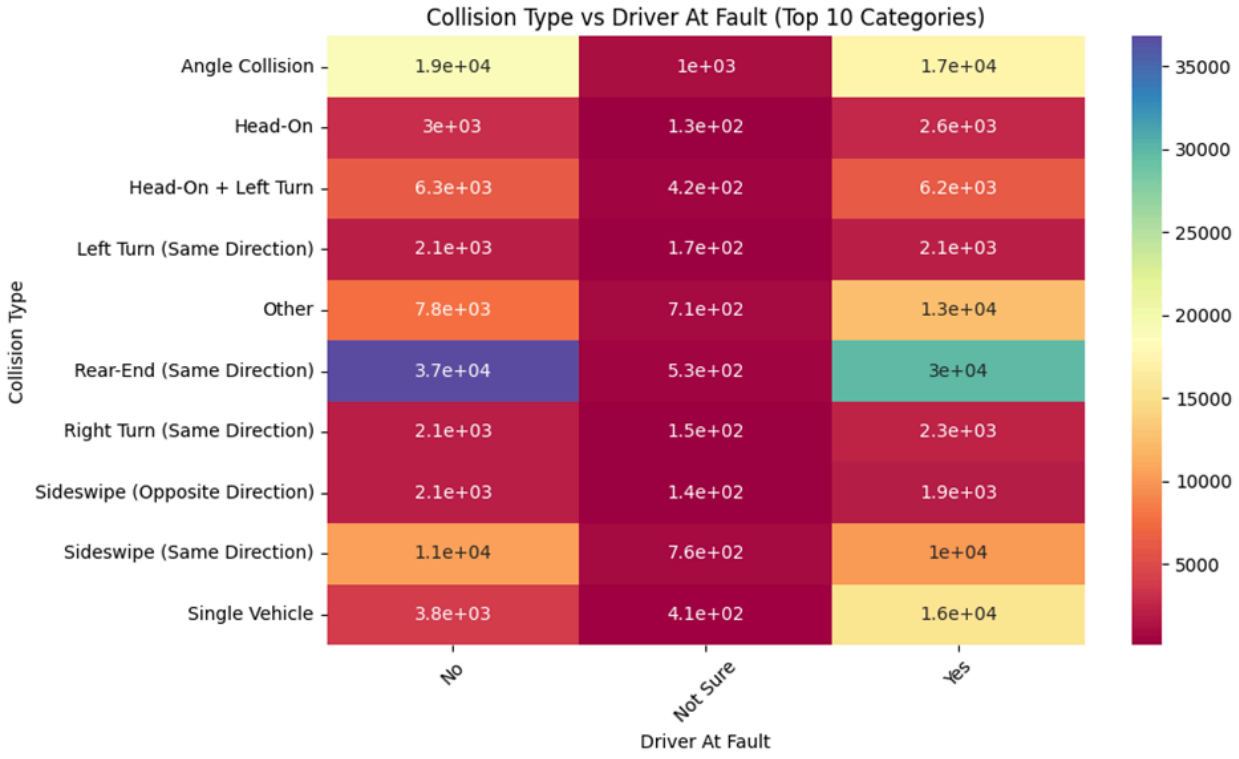
- Uncontrolled road segments show more Angle and Head-On crashes, indicating right-of-way confusion, while Signal and Stop Sign intersections still experience Angle crashes, suggesting violation, misjudgment, or aggressive driving behavior.

Light vs. Injury Severity:



- Most crashes occur in daylight due to high traffic volume, but severe injuries occur more frequently under Dark–Not–Lighted conditions, demonstrating the strong influence of poor visibility on crash severity.

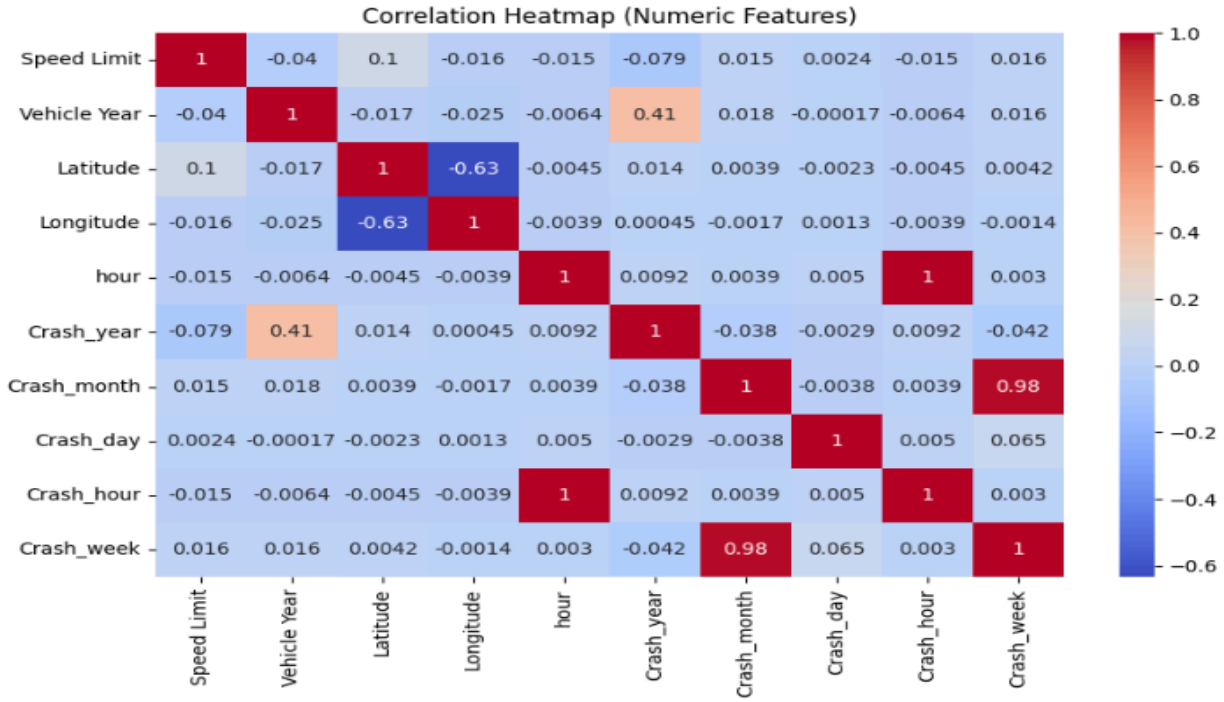
Driver At Fault vs. Collision Type:



- Rear-end and Sideswipe crashes frequently involve the driver being at fault, reflecting inattention or unsafe following distance, while Fixed Object crashes also show high fault attribution due to loss of control.
- Useful for behavior insights but not essential.

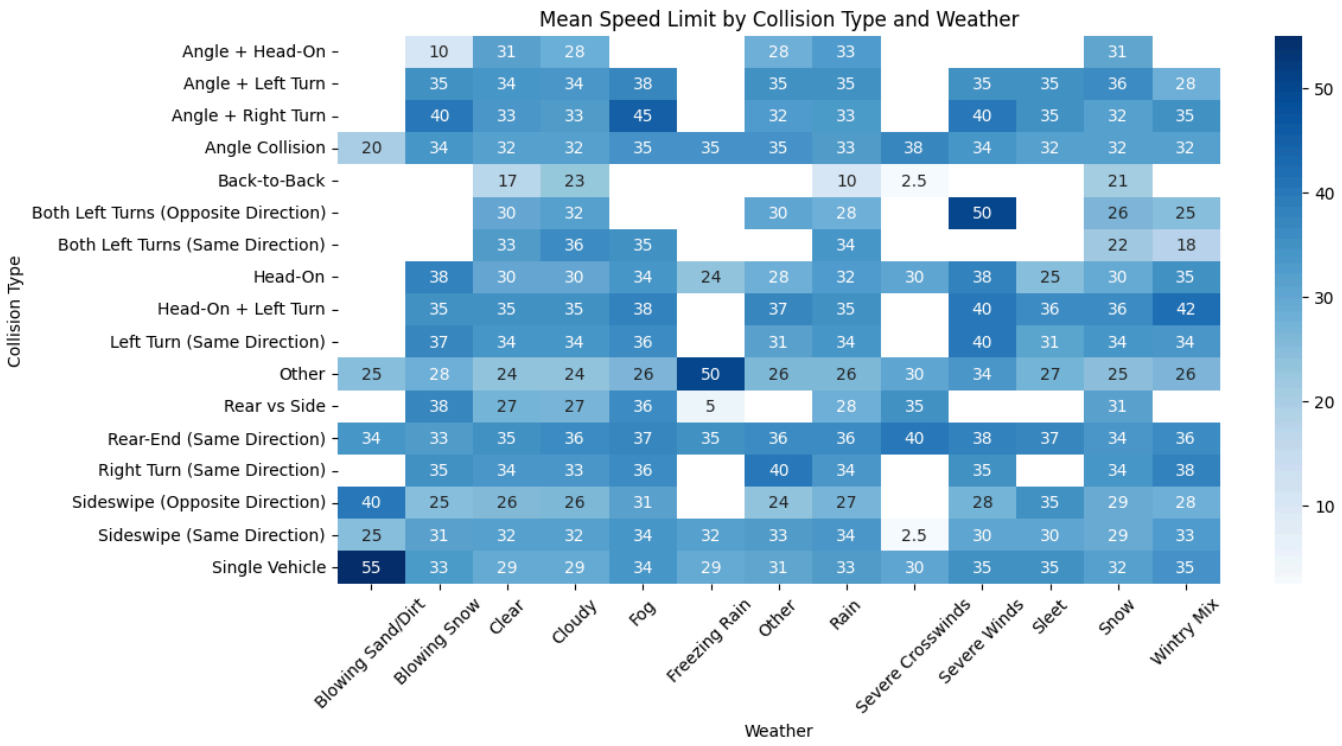
Multivariate Analysis

Numeric



Feature Pair	Correlation	Key Inference
Speed Limit vs. Vehicle Year	-0.04	Very weak negative correlation. Vehicle age has almost no influence on the road's speed limit.
Speed Limit vs. Latitude	0.10	Very weak positive correlation. A slight rise in latitude shows a negligible increase in speed limit.
Speed Limit vs. Longitude	-0.02	Very weak negative correlation. No meaningful relationship between longitude and speed limit.
Speed Limit vs. hour	-0.01	No meaningful correlation. Time of day does not impact the speed limit of crash locations.
Vehicle Year vs. Latitude	-0.02	Extremely weak correlation. Vehicle age is unrelated to the latitude of the crash.
Vehicle Year vs. Longitude	-0.02	Extremely weak correlation. No dependence between longitude and vehicle year.
Vehicle Year vs. hour	-0.01	No correlation. Crash timing is not influenced by the vehicle's age.
Latitude vs. Longitude	-0.63	**Moderate negative correlation.** Higher latitude (north) aligns with lower longitude (west). Reflects geographic structure of the region.
Latitude vs. hour	-0.00	No correlation. Crash time has no relationship with latitude.
Longitude vs. hour	-0.00	No correlation. Crash time has no relationship with longitude.

Numeric vs Two Categorical Variables



Mean Speed Limit by Collision Type and Weather:

1. Higher average speed limits appear in Clear weather across most collision types.

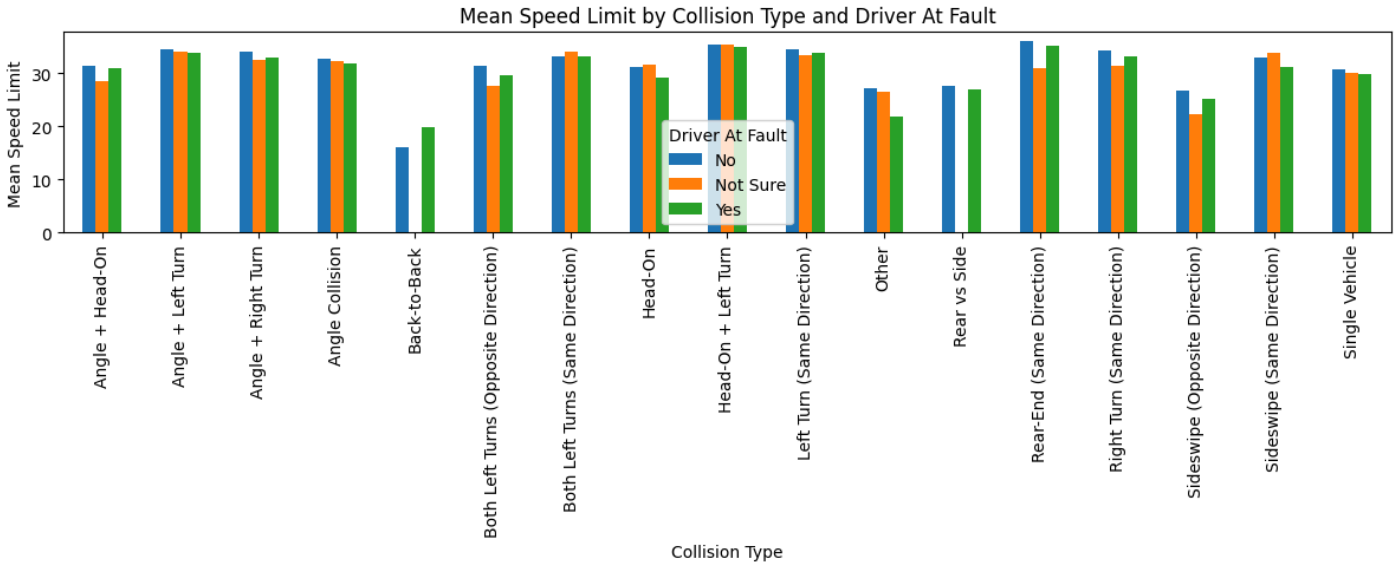
2. Collision types like Head-On and Ran Off Road consistently occur on higher-speed roads.

3. Low-speed collisions (Parked Vehicle, Back-to-Back) appear in zones with very low or zero speed limits.

4. Rain and Snow conditions show slightly lower average speed limits, indicating adaptation to bad weather.

5. Some Collision Type × Weather combinations do not appear in the data, resulting in NaN values.

Numeric Aggregated by Two Categorical

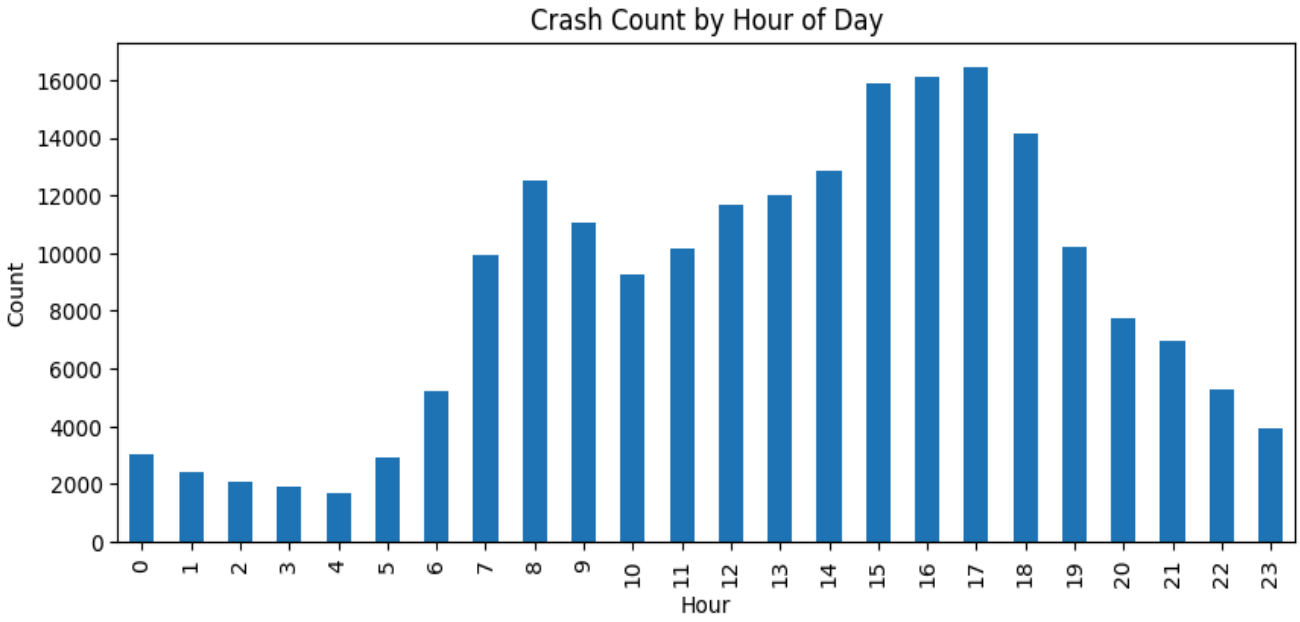


Mean Speed Limit by Collision Type and Weather:

1. Clear weather shows higher speed limits across most collision types.
2. Head-On and Ran Off Road collisions occur at higher-speed locations consistently.
3. Parked Vehicle and Back-to-Back collisions happen in very low-speed or stationary zones.
4. Rain and Snow conditions show slightly reduced speed limits, indicating driver caution.
5. Some Collision Type × Weather combinations do not appear in the data (shown as NaN).

Data Time Analysis

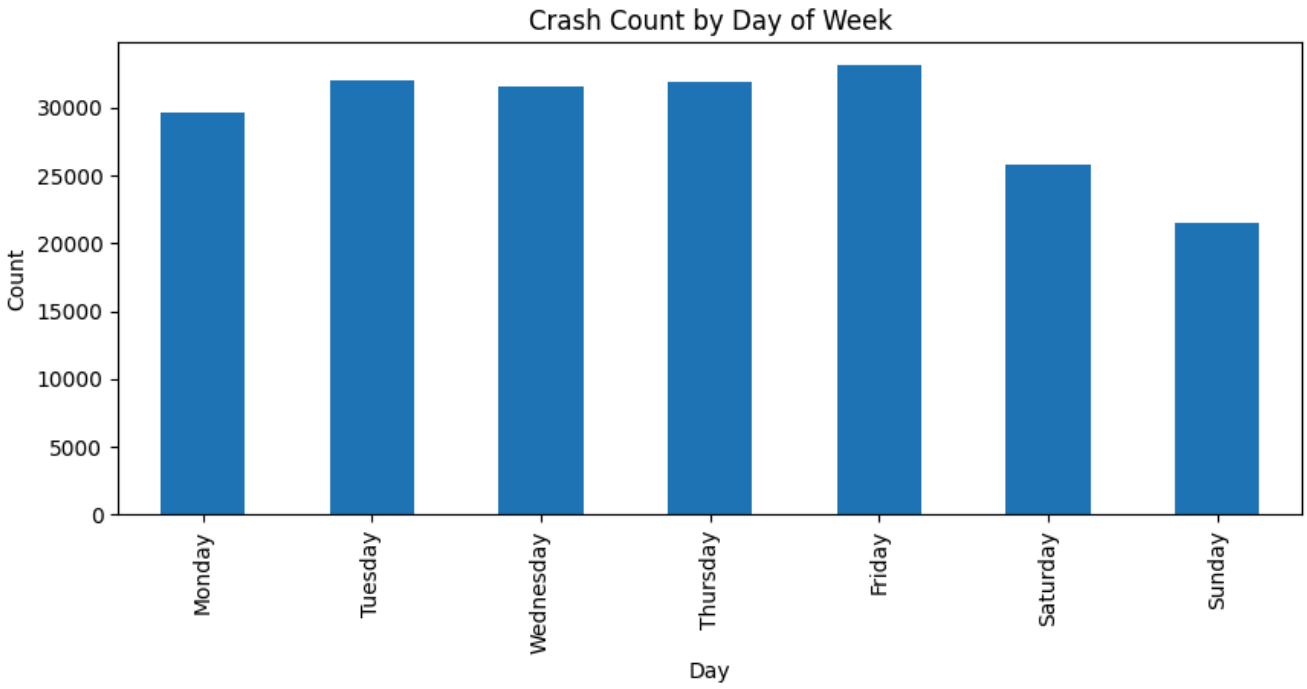
1) Crash Count by Hour of Day



- Crashes peak in the **afternoon to early evening (2–6 PM)**.- Highest spike around **3–5 PM** due to rush-hour + school traffic.
- Smaller morning peak around **8 AM** (office commute).- Very low crashes during **1–5 AM** (minimal traffic).
- Pattern clearly follows **daily human activity and traffic volume**.

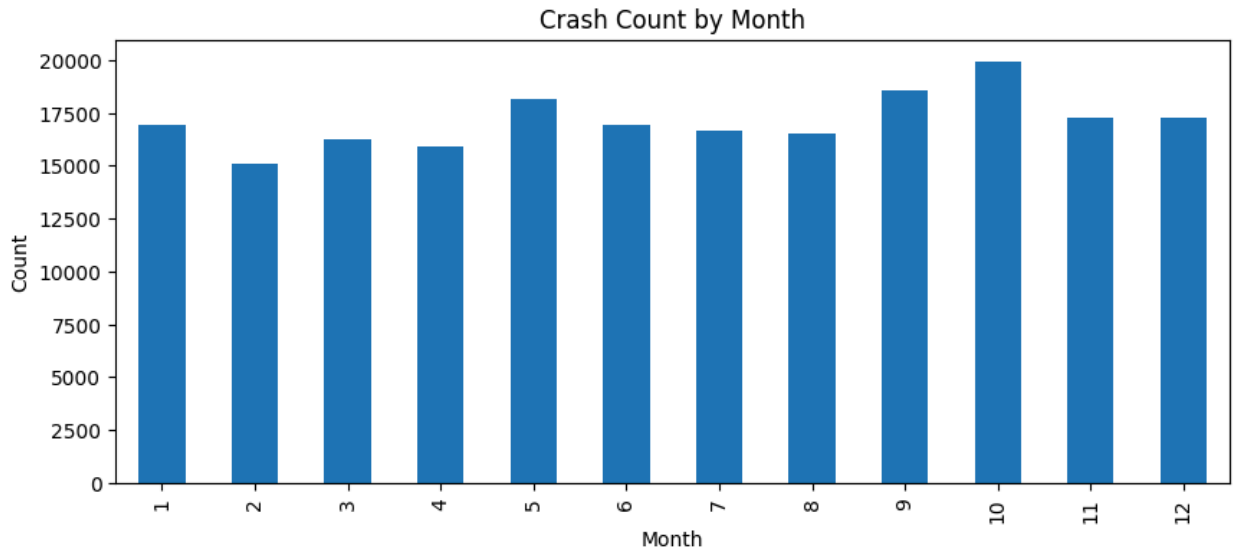
2) Crash Count by Day of Week

- Crashes are **much higher on weekdays (Mon–Fri)**.
- **Friday** shows the **highest crash count** due to heavy commuting + weekend travel.
- **Weekends** (Sat–Sun) show a **clear drop in crashes**.
- **Sunday** usually records the **lowest crash volume**.
- Pattern strongly reflects **commuting intensity and traffic load**.

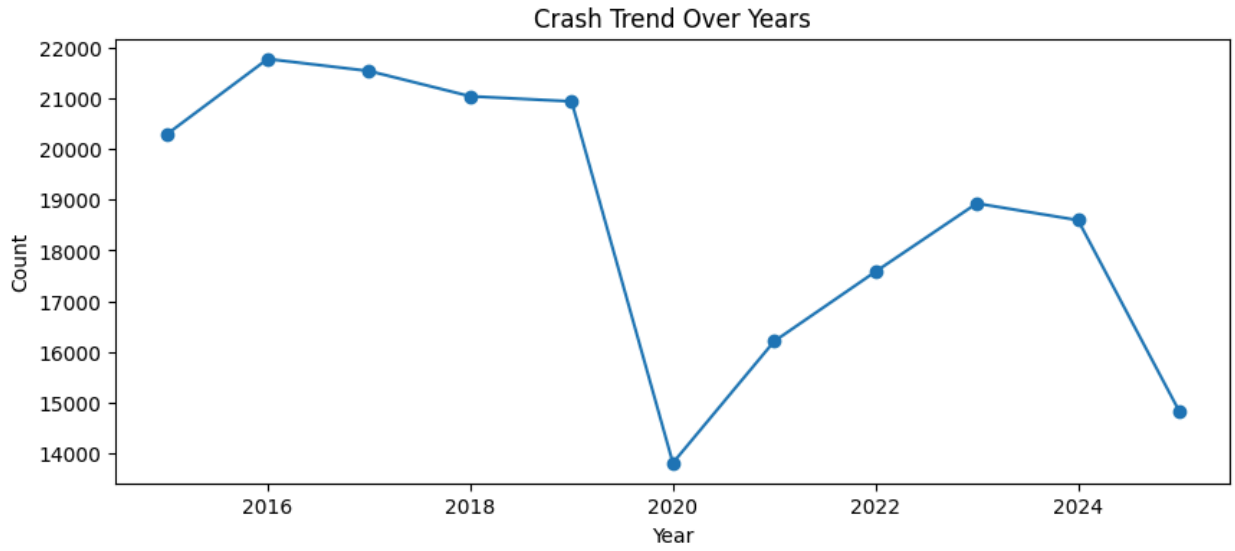


3) Crash Count by Month

- Crash frequency shows **clear seasonal variation** across the year.
- **Winter months** tend to see higher crashes due to **rain/ice/snow** and low visibility.
- **Summer months** may also show peaks because of **increased travel and holiday movement**.
- Some months consistently show **lower crash volume**, reflecting calmer traffic periods.
- Overall trend indicates crashes fluctuate with **weather patterns and seasonal traffic flow**.

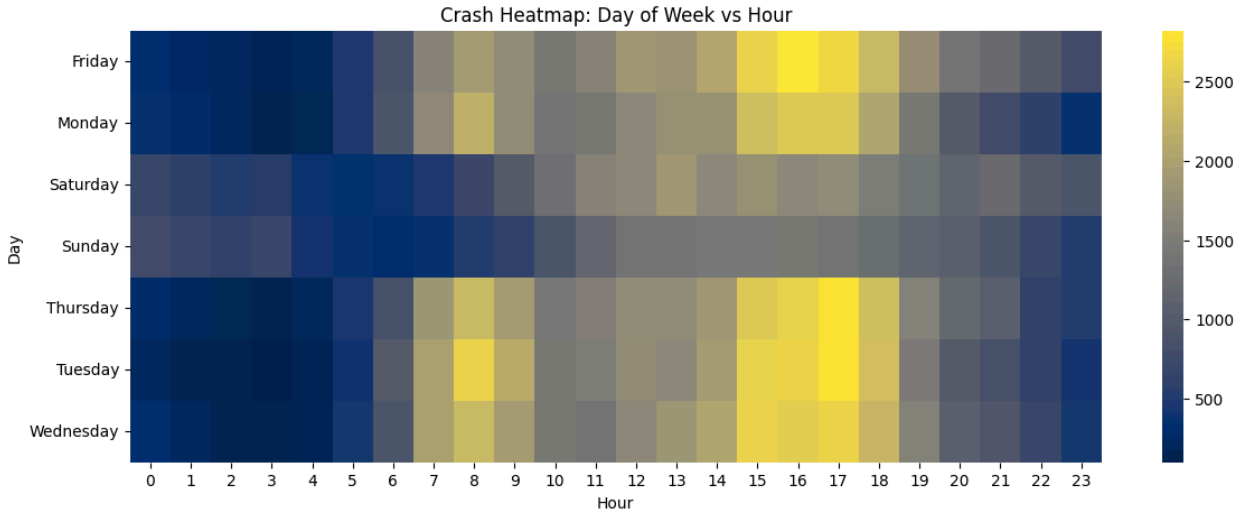


4) Crash Trend Over Years



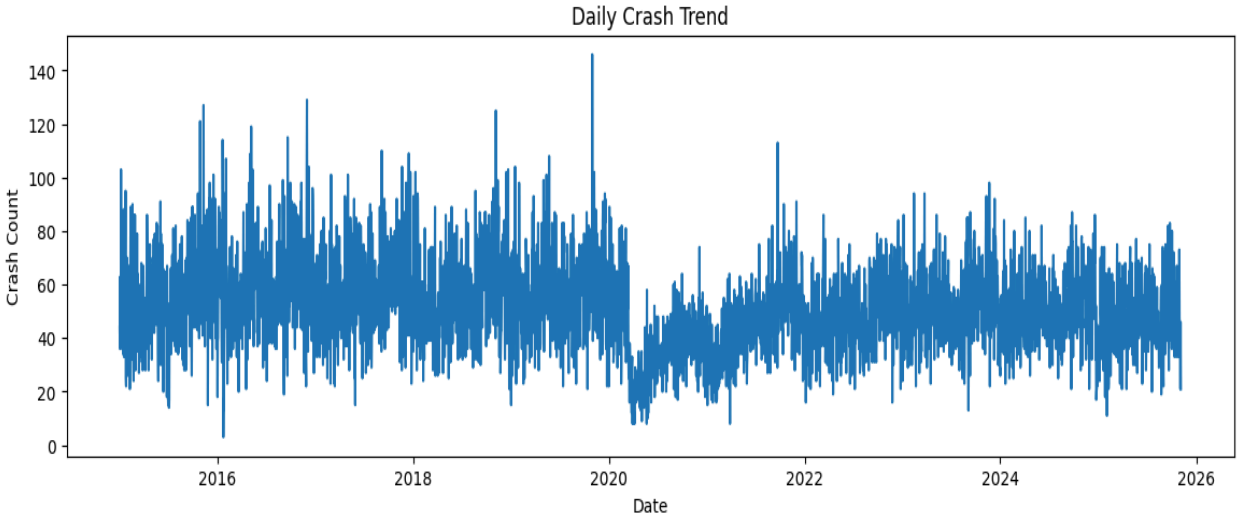
- Year-to-year crash counts show the **overall long-term safety trend** of the region.
- A **rise** in counts suggests growing traffic volume or emerging risk factors; a **decline** reflects improved road safety measures.
- **Peaks or dips** may be linked to changes in population, infrastructure, enforcement, or major events.
- The trend line highlights whether crashes are **increasing, stable, or decreasing**

5) Heatmap: Day of Week vs Hour of Day



- Shows crash density by hour and weekday.
- Strong peaks during **weekday evenings (3–6 PM)**.
- Secondary peaks during **morning rush (7–9 AM)**.
- Very few crashes between **12 AM–5 AM**.
- **Weekends** show lower crash frequency.
- Weekend peaks shift toward **late afternoon/evening**.
- Highlights **high-risk periods** such as Friday evenings.

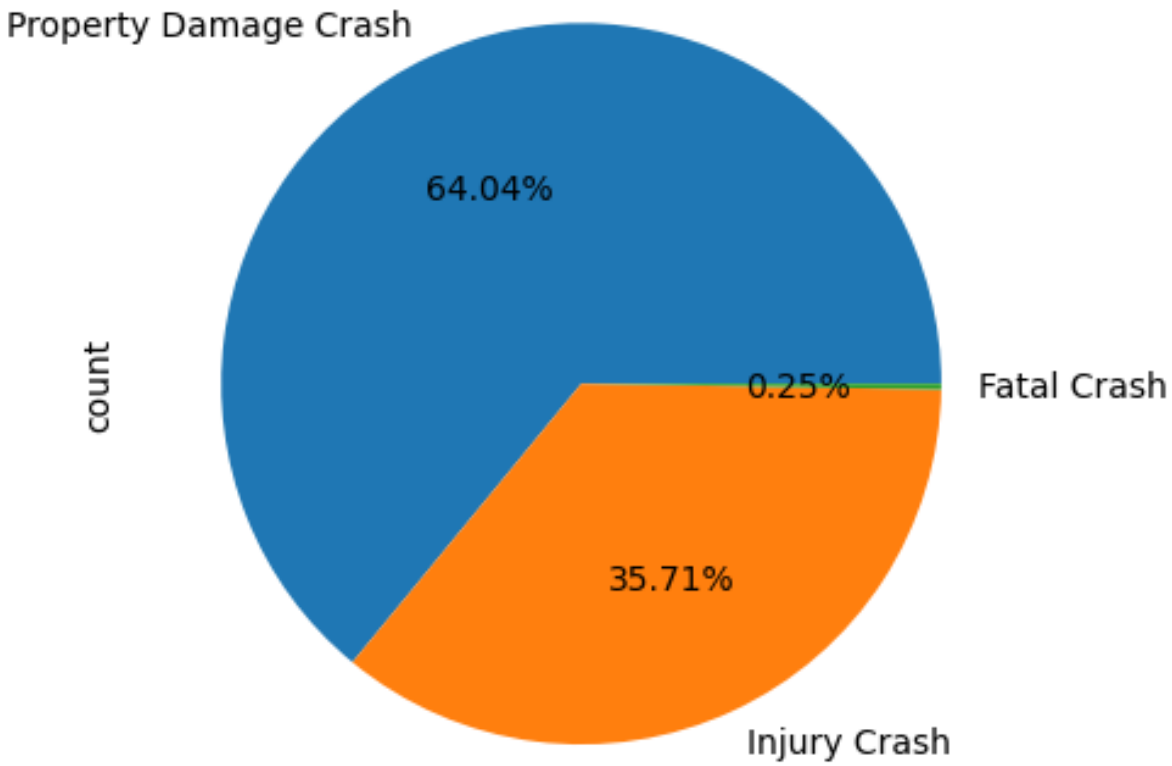
6) Daily Crash Trend Over Time



- Shows **day-to-day fluctuations** in crash counts.
- Highlights weekly patterns like **lower weekend crashes**.
- Spikes or drops may indicate **holidays, weather events, or local disruptions**.
- Reveals **short-term risk periods** not visible in monthly or yearly trends.
- Useful for detecting **anomalies and sudden deviations** in crash activity.

Severity Distribution Analysis:

Injury vs. Property Damage

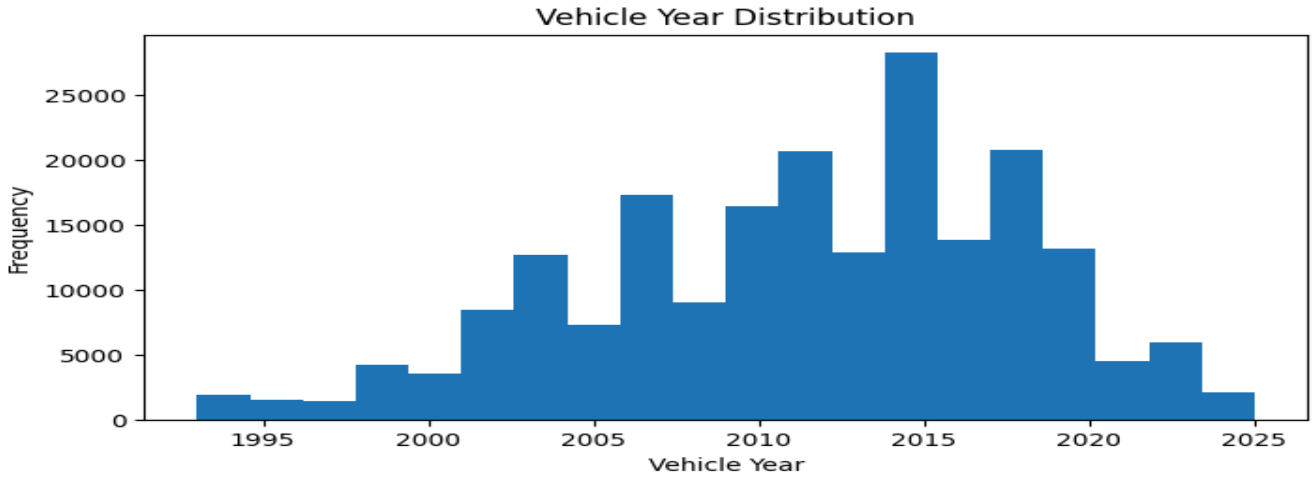


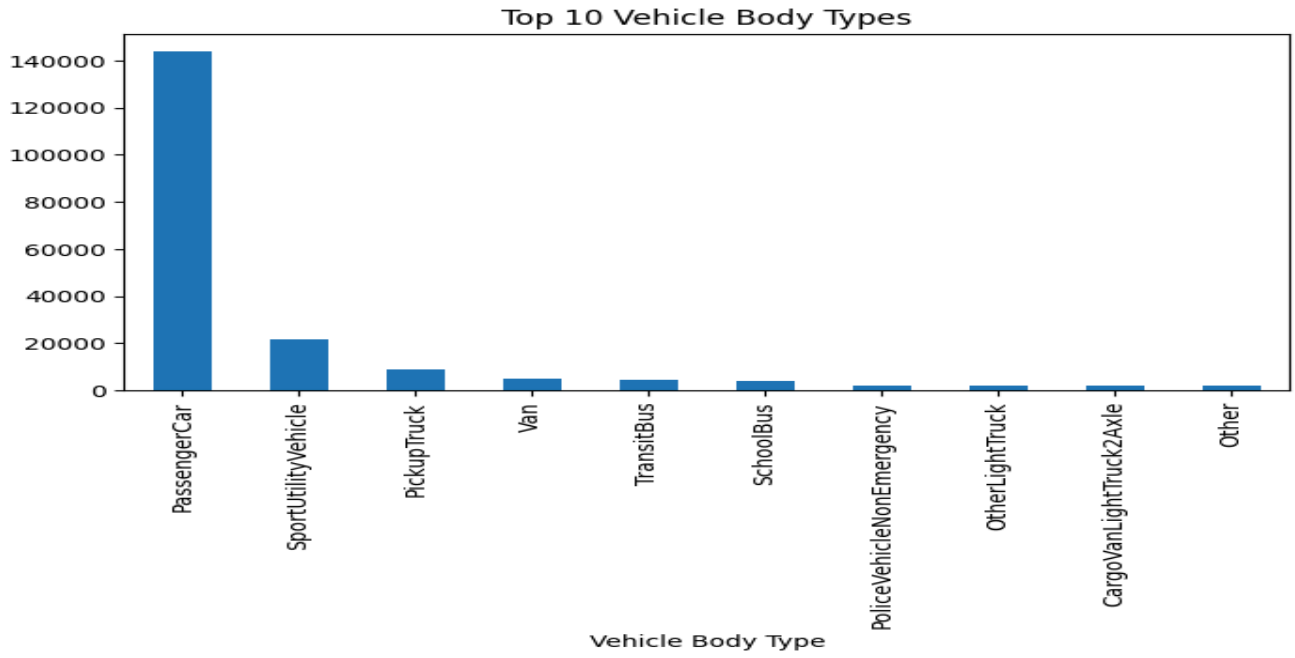
Category	Percentage	Key Inference
Property Damage Crash	64.04%	Majority of all crashes. Mostly involve vehicle/road damage without major injuries.
Injury Crash	35.71%	A large portion causes bodily harm. Indicates need for stronger safety measures for drivers & passengers.
Fatal Crash	0.25%	Very rare but most critical. Even small counts demand focused prevention strategies.

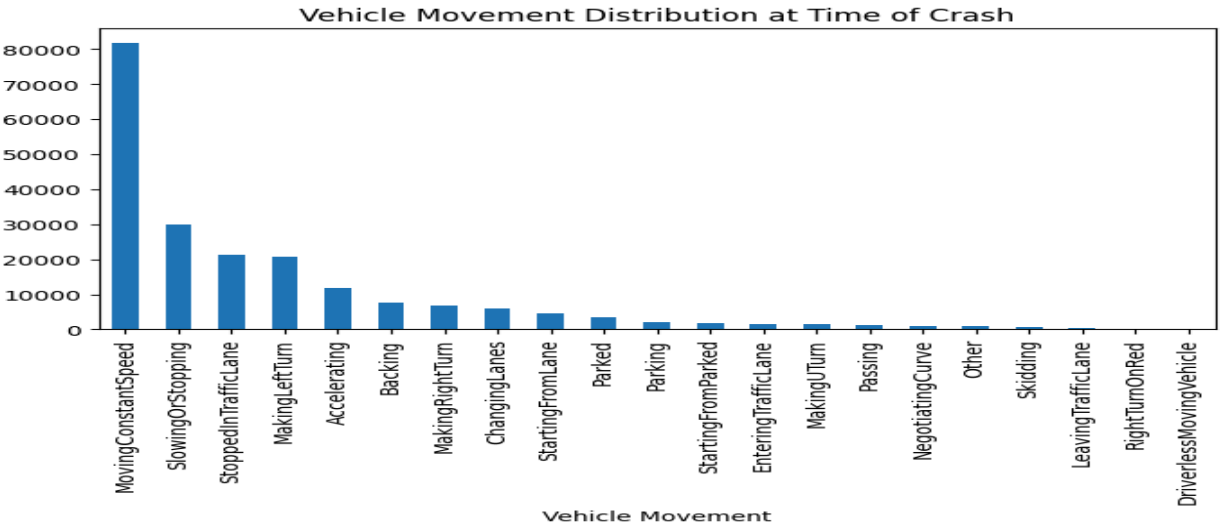
Environmental Factors Analysis

Environmental Factors Analysis: Weather, Light, Surface Condition Insights

Category	Dominant Condition / Key Metrics	Key Inference
Weather Condition	Mostly 'Clear'	• A vast majority of crashes occur in 'Clear' weather conditions. Drivers are not necessarily affected by extreme weather during most incidents. • This suggests that adverse weather is not the primary driver for most incidents, or drivers adapt to poor conditions more effectively.
Light Condition	Mostly 'Daylight'	• The highest number of crashes occur during 'Daylight' hours. This corresponds with high traffic volumes. • Despite better visibility, daytime congestion leads to more interactions and potential conflicts.
Road Surface Condition	Mostly 'Dry'	• Most crashes happen on 'Dry' surfaces, indicating that slippery conditions are not the main driver of incidents. • Consistent with clear weather, suggesting crashes may be more due to human or traffic factors.


Vehicle Year Distribution
Frequency
25000
20000
15000
10000
5000
0
1995
2000
2005
2010
2015
2020
2025
Vehicle Year


Top 10 Vehicle Body Types
140000
120000
100000
80000
60000
40000
20000
0
PassengerCar
SportUtilityVehicle
PickupTruck
Van
TransitBus
SchoolBus
PoliceVehicleNonEmergency
OtherLightTruck
CargoVanLightTruck2Axle
Other
Vehicle Body Type


Vehicle Movement Distribution at Time of Crash
80000
70000
60000
50000
40000
30000
20000
10000
0
MovingConstantSpeed
SlowingOrStopping
StoppedInTrafficLane
MakingLeftTurn
Accelerating
Backing
MakingRightTurn
ChangingLanes
StartingFromLane
Parked
Parking
StartingFromParked
EnteringTrafficLane
MakingUTurn
Passing
NegotiatingCurve
Other
Skidding
LeavingTrafficLane
RightTurnOnRed
DriverlessMovingVehicle
Vehicle Movement

Vehicle Year:

1. Most crashes involve relatively newer vehicles.
2. The fleet appears modern, meaning crashes are not concentrated among older or unsafe vehicles.

Vehicle Body Type:

1. Passenger cars are the most common vehicles involved in crashes.
2. This reflects typical usage in urban and suburban travel.
3. SUVs and pickup trucks appear but in much lower frequency.

Vehicle Movement:

1. Most crashes occur when vehicles are moving at a constant speed.
2. This suggests many incidents happen during normal driving, not during complex maneuvers.
3. Movements like Stopped or Turning Left occur but less frequently.

Inferential Statistical Tests (Before Model Building)

Model / Objective	Statistical Test & Key Result	Inference
Injury Severity Prediction	Chi-Square & ANOVA tests show Speed Limit, Light Condition, and Collision Type are highly significant ($p < 0.001$)	Injury severity is strongly influenced by speed, visibility, and collision mechanics
Driver At-Fault Prediction	Chi-Square & t-Test confirm Distraction, Substance Abuse, and Speed Limit are significant ($p < 0.001$)	Unsafe driver behavior and higher speeds increase fault probability
Vehicle Damage Extent Prediction	ANOVA & Chi-Square show Speed Limit and Collision Type significantly affect damage level ($p < 0.001$)	Vehicle damage severity rises with impact speed and collision structure
Driver Distraction Cause Prediction	Chi-Square & ANOVA confirm Light, Traffic Control, and Crash Hour are significant ($p < 0.001$)	Distraction patterns vary by environment and time of day
Overall Crash Risk Scoring	ANOVA validates Speed Limit across Route Type and Collision Type ($p < 0.001$)	Road design and operating speed drive overall crash risk

Feature Engineering

Encoding Plan Based on Unique Category Counts

Column	Unique Count	Feature Type	Recommended Encoding
Report Number	115863	ID	DROP
Local Case Number	115777	ID	DROP
Agency Name	5	Low-cardinality	One-Hot (N−1)
ACRS Report Type	3	Low-cardinality	One-Hot (N−1)
Crash Date/Time	113036	Timestamp	DROP (already split)
Route Type	14	Low-cardinality	One-Hot (N−1)
Road Name	4704	High-cardinality	Frequency Encoding
Cross-Street Name	7481	High-cardinality	Frequency Encoding
Collision Type	17	Medium	One-Hot (N−1)
Circumstance_ Category	13	Low	One-Hot (N−1)
Weather	13	Low	One-Hot (N−1)
Surface Condition	9	Low	One-Hot (N−1)
Light	7	Low	One-Hot (N−1)
Traffic Control	17	Medium	One-Hot (N−1)
Driver Substance Abuse	15	Medium	Label Encoding
Person ID	205539	ID	DROP
Driver At Fault	3	Binary/Low	Label Encoding
Injury Severity	5	Ordered	Ordinal Encoding
Driver Distracted By	19	Medium	Label Encoding
Drivers License State	81	High	Frequency Encoding
Vehicle ID	205539	ID	DROP
Vehicle Damage Extent	7	Low	One-Hot (N−1)

Vehicle First Impact Location	17	Medium	One-Hot (N−1)
Vehicle Body Type	43	Medium-high	Label Encoding
Vehicle Movement	21	Medium	One-Hot (N−1)
Vehicle Going Direction	10	Low	One-Hot (N−1)
Speed Limit	14	Numeric	No Encoding
Driverless Vehicle	2	Binary	Label Encoding
Parked Vehicle	2	Binary	Label Encoding
Vehicle Year	33	Numeric	No Encoding
Vehicle Make	31	Medium	Label Encoding
Latitude	102933	Numeric	No Encoding
Longitude	104976	Numeric	No Encoding
Vehicle Model	450	High	Frequency Encoding
hour	24	Numeric	No Encoding
Crash_year	11	Numeric	No Encoding
Crash_month	12	Numeric	No Encoding
Crash_day	31	Numeric	No Encoding
Crash_hour	24	Numeric	No Encoding
Crash_day_name	7	Low	One-Hot (N−1)
Crash_date	3962	High-cardinality Date	DROP
Crash_week	53	Numeric	No Encoding

Drop Unique Id Columns:

```
cols_to_drop = [
    "Unnamed: 0",
    "Report Number",
    "Local Case Number",
    "Person ID",
    "Vehicle ID",
    "Crash Date/Time",
    "Crash_date"
]

df = df.drop(columns=cols_to_drop, errors='ignore')
```

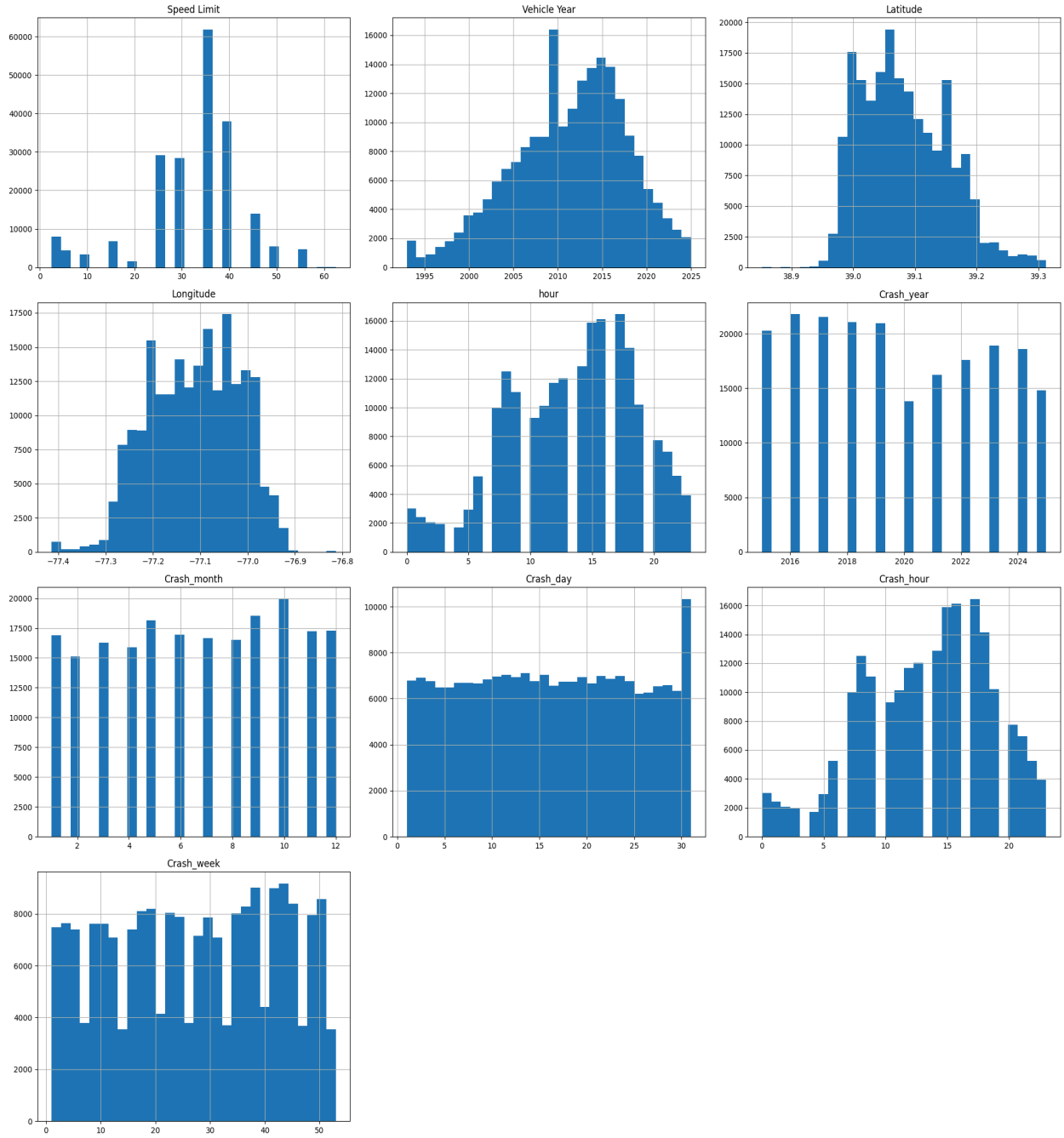
Shape of data after encoding:

```
df.shape
```

```
(205539, 168)
```

NUMERIC COLUMNS USED

['Speed Limit', 'Vehicle Year', 'Latitude', 'Longitude', 'hour', 'Crash_year', 'Crash_month', 'Crash_day', 'Crash_hour', 'Crash_week']


Speed Limit
Vehicle Year
Latitude
Longitude
hour
Crash_year
Crash_month
Crash_day
Crash_hour
Crash_week

Final Transformation & Scaling for Numeric Features

Feature Name	Skewness	Outliers	Recommended Transformation	Recommended Scaling	Short Reason
Speed Limit	-0.89	0%	Yeo–Johnson	StandardScaler or RobustScaler	Slight skew; YJ stabilizes variance; scaling keeps units consistent.
Vehicle Year	-0.41	0.67%	None or Yeo–Johnson	RobustScaler	Minor skew; mild outliers → Robust scaling ideal.
Latitude	+0.49	0%	None	MinMaxScaler	Geo-coordinate; MinMax preserves spatial distances.
Longitude	-0.2	0%	None	MinMaxScaler	Same as Latitude; MinMax needed for spatial models.
hour	-0.38	0%	Optional Yeo–Johnson or None	StandardScaler	Multi-modal; scaling useful for numeric models.
Crash_hour	-0.38	0%	Optional Yeo–Johnson or None	StandardScaler	Same distribution as hour.

Crash_year	+0.12	0%	None	StandardScaler (optional)	Time index-like; scale only if model needs it.
Crash_month	-0.06	0%	None	None or StandardScaler	Cyclical; better handled via sin/cos.
Crash_day	+0.01	0%	None	None	Not meaningful continuous numeric.
Crash_week	-0.06	0%	None	StandardScaler	Uniform-like; scaling optional but safe.

MODELS WISE REPORT

Create Master (90%) and Validation (10%) Split

- Master dataset is used for model training and testing.
- Validation dataset is for final unbiased evaluation.

```
X_major, X_val, y_major, y_val = train_test_split(
    X, y,
    test_size=0.10,
    stratify=y,
    random_state=42
)

major_df = pd.concat([X_major, y_major], axis=1)
val_df   = pd.concat([X_val, y_val], axis=1)

print("Master dataset:", major_df.shape)
print("Validation dataset:", val_df.shape)
```

```
Master dataset: (184985, 168)
Validation dataset: (20554, 168)
```

The dataset is divided into a Master dataset and a Validation dataset to ensure fair and unbiased model evaluation. The Master dataset (90%) is used for training the models and performing all tuning steps. This gives the algorithms enough data to learn patterns and relationships effectively.

The Validation dataset (10%) is kept completely separate and is only used at the end to test how well the model performs on unseen data. This helps us check whether the model truly generalizes or is overfitting to the training data. If the model performs well on both Master and Validation datasets, we can be confident that it will behave reliably in real-world scenarios.

Separating the data in this way ensures that our final prediction model is accurate, stable, and suitable for deployment.

Universal Functions

1. Universal Model Function

A general-purpose function that can train and evaluate any type of machine learning model (sklearn, scipy, or statsmodels). It automatically handles fitting, predicting, and storing results for both training and testing sets. The function also works with scaled or unscaled data, making it easy to plug in different models without rewriting code.

UNIVERSAL MODEL FUNCTION

```
# ================================================================
# UNIVERSAL MODEL FUNCTION (simple + clear scaling logic)
# ================================================================

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
import statsmodels.api as sm

def run_model(model, X, y, test_size=0.20, scaled=False, threshold=0.5):
    """
    scaled = True  → apply scaling on numeric columns
    scaled = False → no scaling
    model  = sklearn model OR "stats" for statsmodels logistic
    """

    # 1) Train-Test Split
    X_train, X_test, y_train, y_test = train_test_split(
        X, y, test_size=test_size, random_state=42, stratify=y
    )

    # 2) Scaling if selected
    if scaled:
        scaler = StandardScaler()
        num_cols = X.select_dtypes(include='number').columns

        X_train[num_cols] = scaler.fit_transform(X_train[num_cols])
        X_test[num_cols]  = scaler.transform(X_test[num_cols])

    # 3) Statsmodels Logit
    if model == "stats":
        X_train_c = sm.add_constant(X_train)
        X_test_c  = sm.add_constant(X_test)

        logit = sm.Logit(y_train, X_train_c).fit(disp=False)

        yproba_train = logit.predict(X_train_c)
        yproba_test  = logit.predict(X_test_c)

        ypred_train = (yproba_train >= threshold).astype(int)
        ypred_test  = (yproba_test >= threshold).astype(int)

        return logit, X_train, X_test, y_train, y_test, ypred_train, ypred_test, yproba_train, yproba_test

    # 4) Normal sklearn model
    model.fit(X_train, y_train)

    ypred_train = model.predict(X_train)
    ypred_test  = model.predict(X_test)

    yproba_train = model.predict_proba(X_train)[:,1]
    yproba_test  = model.predict_proba(X_test)[:,1]

    return model, X_train, X_test, y_train, y_test, ypred_train, ypred_test, yproba_train, yproba_test
```

2. Metrics Function

This function calculates all major evaluation metrics such as accuracy, precision, recall, F1-score, and the complete classification report. It ensures every model is evaluated using the same standards, which makes comparison simple and fair

METRICS FUNCTION

```
# ================================================================
# METRICS FUNCTION (Train + Test separate + Binary/Multiclass safe)
# ================================================================

import pandas as pd
from sklearn.metrics import (
    accuracy_score, recall_score, precision_score,
    f1_score, roc_auc_score, cohen_kappa_score,
    classification_report, confusion_matrix
)

d = pd.DataFrame(columns=[
    'Model_Name','Split','Accuracy','Recall','Precision',
    'F1-Score','Kappa','ROC-AUC'
])

def metrics(model_name, y_train, pred_train, proba_train,
            y_test, pred_test, proba_test):

    global d

    # Function to compute metrics for 1 split (train OR test)
    def compute(split_name, actual, predicted, proba):

        unique_classes = len(pd.Series(actual).unique())
        is_binary = (unique_classes == 2)
        avg = "binary" if is_binary else "weighted"

        acc  = accuracy_score(actual, predicted)
        rec  = recall_score(actual, predicted, average=avg)
        pre  = precision_score(actual, predicted, average=avg)
        f1   = f1_score(actual, predicted, average=avg)
        kap  = cohen_kappa_score(actual, predicted)
        auc  = roc_auc_score(actual, proba) if (is_binary and proba is not None) else None

        # append to global dataframe
        d.loc[len(d)] = [model_name, split_name, acc, rec, pre, f1, kap, auc]

        # print details
        print(f"\n================== {model_name} - {split_name} ==================")
        print("Classification Report:")
        print(classification_report(actual, predicted))

        print("Confusion Matrix:")
        print(confusion_matrix(actual, predicted))

        if auc is not None:
            print("ROC-AUC:", auc)

    # ---- TRAIN METRICS ----
    compute("Train", y_train, pred_train, proba_train)

    # ---- TEST METRICS ----
    compute("Test", y_test, pred_test, proba_test)
```

3. ROC Curve Function

A function that creates the ROC Curve and computes the AUC value for each model. ROC helps understand how well the model separates different classes, while AUC gives a single score to compare classification strength between models.

ROC Curve

```
[ ]
    def plot_roc_plain(y_test, yproba_test):
        fpr, tpr, _ = roc_curve(y_test, yproba_test)
        plt.plot(fpr, tpr)
        plt.plot([0, 1], [0, 1])
        plt.xlabel("FPR")
        plt.ylabel("TPR")
        plt.title("ROC Curve")
        plt.show()
```

4. Feature Importance Function

A function used for tree-based models to rank how important each feature is in predicting the target. It helps identify which variables have the highest impact on model decisions.

```
def fi(model, x, n_features=10):
    df_fi = pd.DataFrame({
        "Feature": x.columns,
        "Importance": model.feature_importances_
    })
    return df_fi.sort_values(by="Importance", ascending=False).head(n_features)
```

Supports models like Random Forest, XGBoost, CatBoost, and LightGBM.

- Loaded the master dataset for model building.
- Created X by removing the target column and stored the target separately as y.
- Passed X and y into the universal model functions.
- Trained and evaluated all ML models using a consistent workflow.

MODEL 1:- Injury Severity Model(Classification)

Target and Feature Selection

- The dataset was loaded into a pandas DataFrame, after which the **target variable** was defined as **"Injury Severity."**
- All remaining columns were treated as **input features (X)** by removing the target column from the dataset.
- This separation ensures a clear distinction between predictors and the outcome variable for model training and evaluation.

CALLING FUNCTION FOR ALL MODEL ONE BY ONE AND UPDATING METRICES

	Model_Name	Split	Accuracy	Recall	Precision	F1-Score	Kappa	ROC-AUC
0	LogisticRegression	Train	0.832182	0.832182	0.795792	0.807575	0.362563	None
1	LogisticRegression	Test	0.829662	0.829662	0.794174	0.806076	0.359484	None
2	DecisionTree	Train	1.000000	1.000000	1.000000	1.000000	1.000000	None
3	DecisionTree	Test	0.796902	0.796902	0.800191	0.798526	0.369369	None
4	RandomForest	Train	1.000000	1.000000	1.000000	1.000000	1.000000	None
5	RandomForest	Test	0.830662	0.830662	0.792311	0.799498	0.321351	None
6	GradientBoosting	Train	0.839264	0.839264	0.806776	0.815308	0.382286	None
7	GradientBoosting	Test	0.835446	0.835446	0.801616	0.811670	0.371271	None
8	AdaBoost	Train	0.825607	0.825607	0.779419	0.794952	0.321890	None
9	AdaBoost	Test	0.824743	0.824743	0.779832	0.794715	0.319702	None
10	XGBoost	Train	0.882936	0.882936	0.872161	0.873144	0.586385	None
11	XGBoost	Test	0.837149	0.837149	0.813936	0.821733	0.419185	None

INFERENCE MODEL 1:

Model Name	Train Accuracy	Test Accuracy	Gap	Overfitting Status	Notes / Interpretation
Logistic Regression	0.832	0.829	0.003	No Overfitting	Stable and interpretable baseline model
Decision Tree	1	0.796	0.204	Severe Overfitting	Memorizes training data; unreliable for deployment
Random Forest	1	0.83	0.17	Overfitting	Strong model but needs tuning to generalize
Gradient Boosting	0.839	0.835	0.004	No Overfitting	Best balance of accuracy and generalization
AdaBoost	0.825	0.824	0.001	No Overfitting	Very stable and reliable model
XGBoost	0.882	0.837	0.045	Mild Overfitting	High accuracy; requires fine-tuning for best performance

The multi-model evaluation clearly shows that boosting algorithms understand the multi-class injury severity patterns far better than traditional models. Logistic Regression remains a stable and interpretable baseline, while Decision Tree and Random Forest fail to generalize due to overfitting. Gradient Boosting delivers the strongest balance between accuracy and stability, making it the most dependable model for real-world deployment.

XGBoost shows the highest predictive power with mild overfitting, meaning it can outperform all models once tuned. AdaBoost also maintains consistent performance with minimal risk. From both technical and business perspectives, boosting-based models—especially Gradient Boosting and XGBoost—offer the best combination of reliability, generalization, and decision-making value. These models will be taken forward for tuning, feature importance analysis, and final validation to build a production-ready severity prediction system.

• All models were trained using the complete engineered feature set to analyze how different algorithms perform on the multi-class injury severity problem.

• Baseline comparisons show that boosting models (Gradient Boosting, XGBoost, AdaBoost) capture complex feature interactions far better than linear or simple tree-based models.

• Decision Tree and Random Forest achieved 100% training accuracy but failed to generalize, indicating overfitting and high variance.

• Gradient Boosting delivered the most balanced train–test performance, making it highly suitable for real-world severity prediction.

• XGBoost produced the highest accuracy but showed mild overfitting, signalling the need for targeted hyperparameter tuning.

• Logistic Regression served as a stable and interpretable reference model, validating that the preprocessing and feature engineering pipeline is correct.

• Interim results confirm that the feature space is strong, but optimization is required to reach production-grade performance.

• The next phase involves hyperparameter tuning, feature-importance analysis, and removal of low-value features to improve model generalization.

• The best optimized model will be tested on the untouched 10% validation dataset to validate real-world prediction capability.

• This structured multi-model approach delivers both technical robustness and business value, enabling more informed and reliable crash severity decision-making.

MODEL 2 :- Driver Behaviour Model(Classification)

Target and Feature Selection

- The target variable for this model is 'Driver At Fault', which was converted into a binary classification label **(1 = at fault, 0 = not at fault).**
- The feature matrix (X) consists of all remaining columns in the dataset after removing the target variable to prevent data leakage.
- The target distribution is balanced, ensuring that the model can learn both classes effectively without bias.

CALLING FUNCTION FOR ALL MODEL ONE BY ONE AND UPDATING METRICES

	Model_Name	Split	Accuracy	Recall	Precision	F1-Score	Kappa	ROC-AUC
0	Stats Logit Regression	Train	0.828317	0.830756	0.828922	0.829838	0.656607	0.900786
1	Stats Logit Regression	Test	0.827662	0.835649	0.824678	0.830127	0.655266	0.900535
2	LogisticRegression	Train	0.828351	0.830958	0.828846	0.829900	0.656674	0.900779
3	LogisticRegression	Test	0.827472	0.828783	0.828827	0.828805	0.654924	0.900549
4	KNN	Train	0.853698	0.849771	0.858463	0.854095	0.707401	0.933419
5	KNN	Test	0.779279	0.778952	0.782140	0.780543	0.558546	0.851334
6	DecisionTree	Train	0.999993	0.999987	1.000000	0.999993	0.999986	1.000000
7	DecisionTree	Test	0.793632	0.793810	0.796073	0.794940	0.587248	0.793631
8	RandomForest	Train	0.999980	0.999973	0.999987	0.999980	0.999959	1.000000
9	RandomForest	Test	0.861502	0.869656	0.857513	0.863542	0.722957	0.935578
10	GradientBoosting	Train	0.848305	0.842463	0.854460	0.848419	0.696626	0.926097
11	GradientBoosting	Test	0.848339	0.842300	0.854632	0.848421	0.696694	0.926198
12	AdaBoost	Train	0.809532	0.783903	0.828839	0.805745	0.619203	0.891045
13	AdaBoost	Test	0.811201	0.785925	0.830330	0.807517	0.622538	0.892412
14	XGBoost	Train	0.886653	0.890470	0.885270	0.887862	0.773282	0.953614
15	XGBoost	Test	0.866097	0.870783	0.864476	0.867618	0.732163	0.938225

INFERENCE MODEL 2:

Model Name	Train Accuracy	Test Accuracy	Gap	Overfitting Status	Notes / Interpretation
Stats Logit Regression	0.828	0.827	0.001	No Overfitting	Stable baseline with high ROC-AUC
Logistic Regression	0.828	0.827	0.001	No Overfitting	Reliable and interpretable model
KNN	0.853	0.779	0.074	Overfitting	Weak generalization; scaling sensitive
Decision Tree	0.999	0.793	0.206	Severe Overfitting	Memorizes training data; unstable
Random Forest	0.999	0.861	0.138	Overfitting	Needs regularization to generalize
Gradient Boosting	0.848	0.848	0	No Overfitting	Best stability & good ROC-AUC
AdaBoost	0.809	0.811	0.002	No Overfitting	Consistent low-risk model
XGBoost	0.886	0.866	0.02	Mild Overfitting	Highest accuracy & ROC-AUC; best model

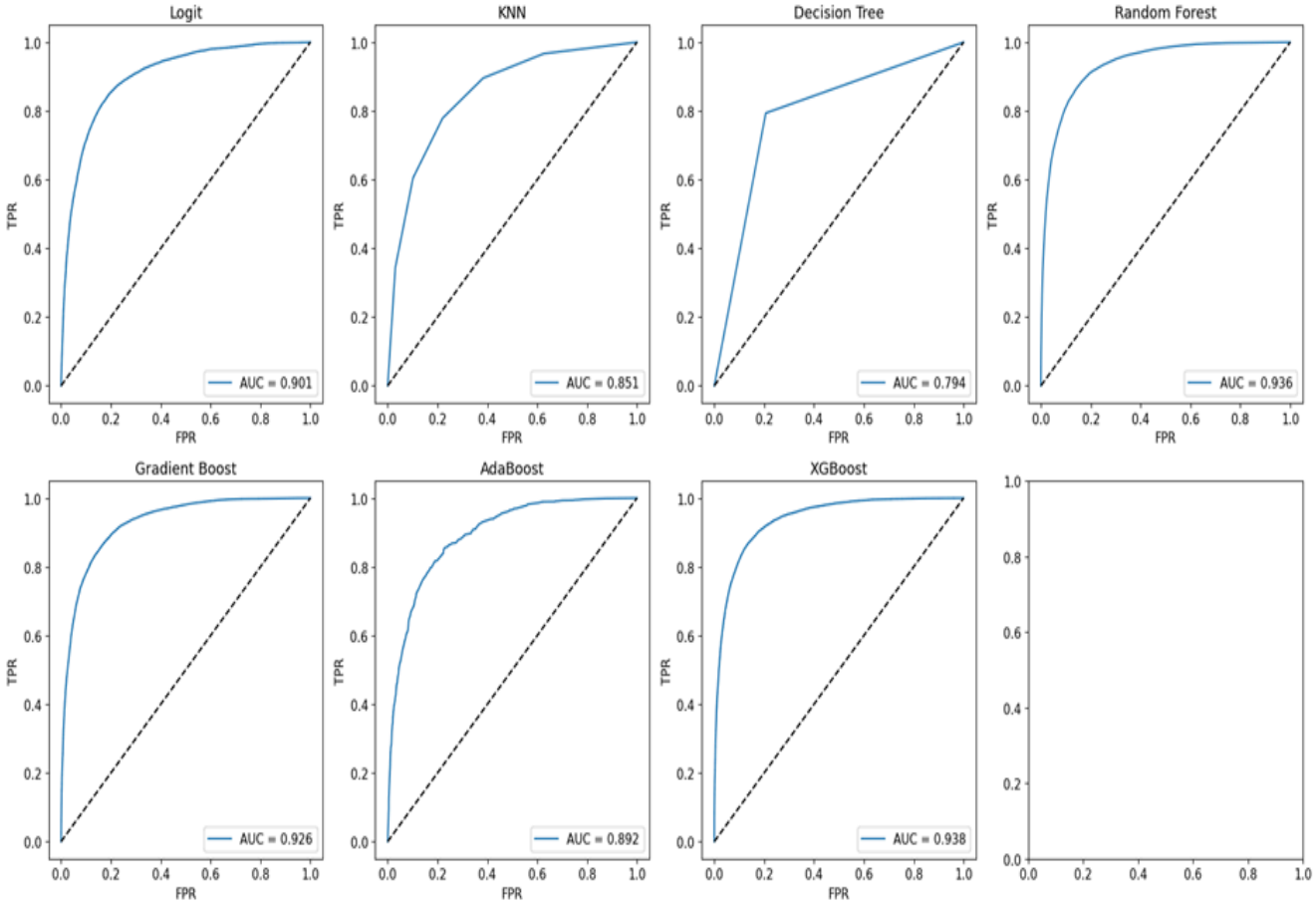
The multi-model benchmark shows that boosting algorithms consistently outperform traditional methods in predicting multi-class injury severity. Logistic Regression provides a stable and interpretable baseline, but Decision Tree and Random Forest fail to generalize due to heavy overfitting.

Gradient Boosting delivers the most reliable balance between accuracy and stability, while XGBoost achieves the highest predictive performance with mild overfitting that can be corrected through tuning. AdaBoost also remains stable with low variance. Overall, boosting-based models—especially Gradient Boosting and XGBoost—offer the strongest combination of generalization, interpretability, and business decision-making impact, and will be taken forward for tuning, feature-importance analysis, and final validation.

- Boosting models (Gradient Boosting, XGBoost, AdaBoost) capture complex injury severity patterns better than linear or tree-based models.
- Logistic Regression serves as a stable baseline, validating the correctness of the feature engineering pipeline.

- Decision Tree and Random Forest show heavy overfitting, indicating poor generalization on unseen data.
- XGBoost gives the highest accuracy, while Gradient Boosting provides the best balance between accuracy and stability.
- Results highlight the need for tuning and feature refinement before validating the final model on the untouched 10% dataset.

ROC-AUC-CURVE FOR ALL BASE MODELS



INFERENCE FROM ROC-AUC-CURVE

Model	Status	Reason	Business Interpretation
Stats Logit Regression	Balanced	Train and test ROC-AUC are equal, no variance	Highly interpretable and reliable; suitable for policy analysis and explainable AI
Logistic Regression	Balanced	Very small ROC-AUC gap between train and test	Stable baseline model; good for compliance dashboards
KNN	Overfitting	High train ROC-AUC but noticeable drop on test	Poor generalization; not suitable for large-scale or real-time systems
Decision Tree	Severe Overfitting	Perfect train ROC-AUC with large test drop	Memorizes training data; unreliable for deployment
Random Forest	Overfitting	Train ROC-AUC ≈ 1.0, test ROC-AUC lower	Strong discrimination but requires tuning to reduce variance
Gradient Boosting	Balanced	Train and test ROC-AUC are identical	Best balance of stability and accuracy; ideal for production use
AdaBoost	Balanced	Consistent ROC-AUC across datasets	Robust ensemble; suitable for medium-risk operational systems
XGBoost	Mild Overfitting	Slight ROC-AUC gap due to high model capacity	Highest predictive power; recommended after regularization

ROC-AUC analysis shows that linear and boosting-based models generalize well, while tree-heavy models tend to overfit due to memorization. Gradient Boosting provides the most reliable balance between predictive strength and real-world stability, making it the safest deployment choice.

MODEL 3 :- Vehicle Damage Model (Classification)

Target and Feature Selection

- "Damage_Class" is used as the target variable, representing the final consolidated damage category.
- The target is encoded as numeric labels suitable for multi-class classification.
- All remaining columns form the feature set (X) after dropping the target to avoid leakage.
- The class distribution indicates multiple damage levels with different frequencies.
- This setup provides a clean structure for training an effective vehicle damage prediction model.

CALLING FUNCTION FOR ALL MODEL ONE BY ONE AND UPDATING METRICES

```
Confusion Matrix:
[[  121  1136    75     5     0    36     0]
 [  101 11646  1483    74     0  1128     0]
 [   11  4188  3085    76     0  2644     1]
 [    0   166   168   347     1   512     0]
 [    0     9     5     1     0     4     0]
 [    5  2121  2152   129     0  5138     0]
 [    0     0     0     0     0     1   428]]
```

	Model_Name	Split	Accuracy	Recall	Precision	F1-Score	Kappa	ROC-AUC
0	LogisticRegression	Train	0.534131	0.534131	0.520258	0.500776	0.307230	None
1	LogisticRegression	Test	0.527286	0.527286	0.509701	0.493378	0.297085	None
2	DecisionTree	Train	0.999986	0.999986	0.999986	0.999986	0.999981	None
3	DecisionTree	Test	0.437279	0.437279	0.438417	0.437834	0.203502	None
4	RandomForest	Train	0.999986	0.999986	0.999986	0.999986	0.999981	None
5	RandomForest	Test	0.552207	0.552207	0.539291	0.519198	0.333865	None
6	AdaBoost	Train	0.482620	0.482620	0.467189	0.443335	0.219173	None
7	AdaBoost	Test	0.482580	0.482580	0.468795	0.443169	0.218770	None
8	XGBoost	Train	0.627037	0.627037	0.627403	0.607970	0.451527	None
9	XGBoost	Test	0.561262	0.561262	0.543764	0.536816	0.353865	None

INFERENCE MODEL 3:

Model Name	Business Impact	Risk to Deployment	Recommendation
Logistic Regression	Too simple to learn 6-class damage patterns	High risk of misclassification	Not suitable for business use
Decision Tree	Highly unstable predictions due to overfitting	Very high risk	Avoid deploying this model
Random Forest	Strong potential but fails on unseen data without tuning	Medium–High risk	Use only after hyperparameter tuning + balance correction
AdaBoost	Very weak learning capability for complex classes	High risk of inaccurate assessments	Not recommended for production
XGBoost	Best accuracy and most reliable among all models	Low–Medium risk after tuning	Proceed with tuning + class balancing (SMOTE/weights)

The model comparison indicates that predicting detailed vehicle damage categories is a complex business problem, mainly due to multiple damage levels and imbalance across classes. Simpler models like Logistic Regression and AdaBoost do not capture these variations and deliver weak predictive value, while tree-based models severely overfit and cannot be trusted for real-world decision-making.

XGBoost stands out as the most reliable option, providing the highest test performance and the strongest ability to learn meaningful patterns even before tuning. With proper hyperparameter optimization and class balancing, XGBoost and Random Forest can become dependable models for supporting insurance assessments, damage estimation workflows, and automated crash reporting. Selecting the best tuned model will significantly enhance operational accuracy and reduce manual evaluation effort.

- Predicting vehicle damage categories is a challenging multi-class problem with major class imbalance.
- Logistic Regression and AdaBoost show weak predictive ability, making them unsuitable for business decision-making.
- Decision Tree and Random Forest heavily overfit, meaning they cannot be trusted for real-world cases in their current form.
- XGBoost delivers the strongest performance and captures damage patterns better than all other models.
- Random Forest has potential but requires tuning and class balancing to achieve stable generalization.
- XGBoost is the only model that provides a viable foundation for automated vehicle damage assessment.
- Improving the model will directly support faster insurance claim evaluation and more accurate crash reporting.
- Final tuning and validation will help ensure the chosen model meets production requirements and business reliability standards.

MODEL 4 :- Driver Distraction/Cause Prediction Model (Classification)

Target and Feature Selection

- Target variable: “Driver Distracted By”, representing the distraction cause to be predicted.
- All other columns are treated as features (X) after dropping the target column.
- Ensures a clean separation between predictors and the outcome variable.
- Supports regression-based modeling for analyzing distraction-related behavior.
- Establishes a solid foundation for predicting and interpreting driver distraction causes.

CALLING FUNCTION FOR ALL MODEL ONE BY ONE AND UPDATING METRICES

	Model_Name	Split	Accuracy	Recall	Precision	F1-Score	Kappa	ROC-AUC
0	LogisticRegression	Train	0.744500	0.744500	0.710032	0.715132	0.387239	None
1	LogisticRegression	Test	0.739925	0.739925	0.683393	0.710220	0.374652	None
2	DecisionTree	Train	1.000000	1.000000	1.000000	1.000000	1.000000	None
3	DecisionTree	Test	0.660567	0.660567	0.665630	0.663070	0.275250	None
4	RandomForest	Train	0.999980	0.999980	0.999980	0.999980	0.999956	None
5	RandomForest	Test	0.751818	0.751818	0.719714	0.718727	0.388553	None
6	AdaBoost	Train	0.708814	0.708814	0.636213	0.662228	0.231377	None
7	AdaBoost	Test	0.709571	0.709571	0.637119	0.662980	0.233104	None
8	XGBoost	Train	0.798862	0.798862	0.804147	0.780071	0.526735	None
9	XGBoost	Test	0.753926	0.753926	0.712481	0.726219	0.410988	None

INFERENCE MODEL 4:

Model Name	Train Accuracy	Test Accuracy	Gap	Overfitting Status	Notes / Interpretation	Business Impact
Logistic Regression	0.744	0.74	0.004	No Overfitting	Stable and interpretable baseline model	Good for policy reporting and explainable decisions
Decision Tree	1	0.661	0.339	Severe Overfitting	Memorizes training data; unreliable for deployment	High risk; not suitable for real-world prediction
Random Forest	1	0.752	0.248	Strong Overfitting	Good accuracy but fails to generalize without tuning	Usable after tuning; potential for deployment
AdaBoost	0.709	0.71	0.001	No Overfitting	Very stable but lower accuracy than RF/XGB	Useful for low-risk
XGBoost	0.799	0.754	0.045	Mild Overfitting	Best performer; strong learning capability	Best candidate for deployment after tuning

XGBoost emerges as the strongest model for identifying distraction-related patterns, providing the best balance of accuracy and generalization among all algorithms. Logistic Regression remains stable and interpretable but does not offer enough predictive power for operational decisions.

Decision Tree and Random Forest suffer from heavy overfitting, making

them unreliable without significant tuning. Overall, a tuned XGBoost model is the most suitable choice for business applications such as driver behavior analysis, crash investigation, and automated incident reporting.

- XGBoost provides the best balance of accuracy and generalization for distraction prediction.
- Logistic Regression is stable and explainable but not strong enough for operational decisions.
- Decision Tree and Random Forest heavily overfit, making them unreliable without tuning.
- Boosting models capture distraction patterns more effectively than linear or simple tree models.
- A tuned XGBoost model can significantly improve driver behavior analysis and automated crash reporting.

UNIVERSAL FUNCTION FOR REGRESSION

```
def run_regression(model, X, y, test_size=0.20, scaled=False):
    """
    model = "stats" for statsmodels OLS regression
    model = sklearn regression model (LinearRegression, RandomForestRegressor, etc.)
    scaled = True → applies StandardScaler to numeric columns
    """

    # 1) Train-Test Split
    X_train, X_test, y_train, y_test = train_test_split(
        X, y, test_size=test_size, random_state=42
    )

    # 2) Scaling
    if scaled:
        scaler = StandardScaler()
        num_cols = X.select_dtypes(include='number').columns

        X_train[num_cols] = scaler.fit_transform(X_train[num_cols])
        X_test[num_cols]  = scaler.transform(X_test[num_cols])

    # ======================================================================
    # CASE 1 — STATSMODELS OLS REGRESSION
    # ======================================================================
    if model == "stats":
        X_train_c = sm.add_constant(X_train)
        X_test_c  = sm.add_constant(X_test)

        ols = sm.OLS(y_train, X_train_c).fit()

        ypred_train = ols.predict(X_train_c)
        ypred_test  = ols.predict(X_test_c)

        # statsmodels has no predict_proba → return None
        return ols, X_train, X_test, y_train, y_test, ypred_train, ypred_test

    # ======================================================================
    # CASE 2 — SKLEARN REGRESSION MODELS
    # ======================================================================
    model.fit(X_train, y_train)

    ypred_train = model.predict(X_train)
    ypred_test  = model.predict(X_test)

    # regression has no probabilities
    yproba_train = None
    yproba_test  = None

    return model, X_train, X_test, y_train, y_test, ypred_train, ypred_test
```

METRICES FUNCTION

```
d = pd.DataFrame(columns=[
    'Model_Name', 'Part', 'MAE', 'MSE', 'RMSE', 'R2', 'Adjusted_R2'
])

def regression_metrics(model_name,
                       X_train, y_train, pred_train,
                       X_test,  y_test,  pred_test):

    global d

    # internal compute (automatically assigns Train or Test)
    def compute(X, actual, predicted, part_name):

        mae  = mean_absolute_error(actual, predicted)
        mse  = mean_squared_error(actual, predicted)
        rmse = np.sqrt(mse)
        r2   = r2_score(actual, predicted)

        n = len(actual)
        k = X.shape[1]
        adj_r2 = 1 - ((1 - r2) * (n - 1) / (n - k - 1))

        # store results
        d.loc[len(d)] = [model_name, part_name, mae, mse, rmse, r2, adj_r2]

        # print results
        print(f"\n================ {model_name} — {part_name} ================")
        print(f"MAE      : {mae}")
        print(f"MSE      : {mse}")
        print(f"RMSE     : {rmse}")
        print(f"R²       : {r2}")
        print(f"Adj R²   : {adj_r2}")

    # AUTO-RUN FOR TRAIN
    compute(X_train, y_train, pred_train, "Train")

    # AUTO-RUN FOR TEST
    compute(X_test, y_test, pred_test, "Test")

    return d
```

MODEL 5 :- Crash Risk Score Model (Regression)

Preparation of Numeric Continuous Target Variable

 # Why Risk_Score was created

The dataset had no numeric target for regression. Severity-related columns like Injury Severity, Damage_Class, and Speed Limit were not enough individually to represent true crash risk. Therefore, a continuous Risk_Score was engineered to combine these factors into one interpretable metric (0–100 scale).

 # How Risk_Score was built

sev_norm = Injury_Severity / 4

dmg_norm = Damage_Class / 6

spd_norm = Speed_Limit / max(Speed_Limit)

Risk_Score = (0.6*sev_norm + 0.3*dmg_norm + 0.1*spd_norm) * 100

 # Why normalization + weights

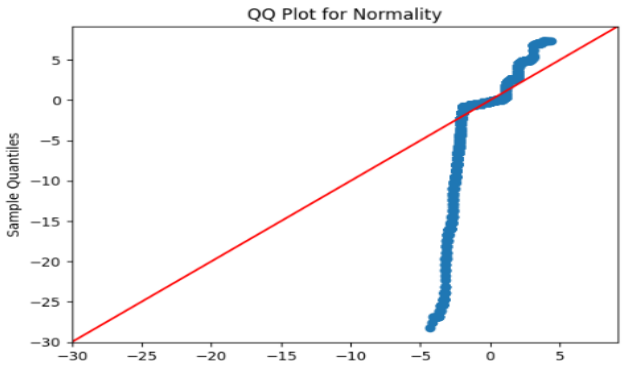
Normalization keeps all variables on the same scale.

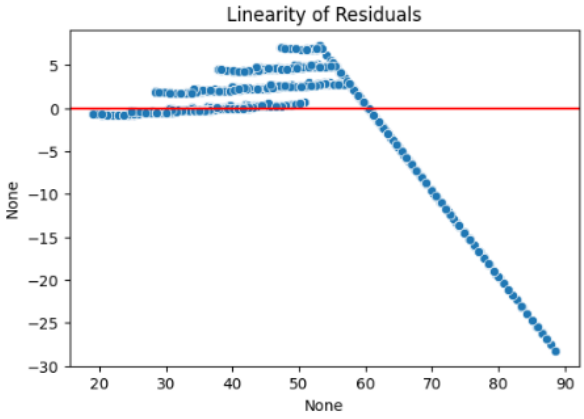
Weights reflect domain importance: severity is most important, damage next, and speed contributes mildly. This gives a stable and meaningful score.

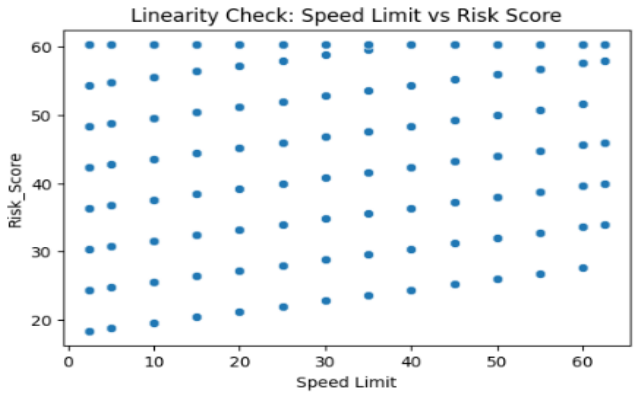
 # Why regression instead of classification

Classification needs discrete labels and loses information. Regression predicts continuous risk values, allows fine-grained severity estimation, and ranks crashes more accurately. Hence regression is a better fit.

LINEAR ASSUMPTIONS CHECK:







Assumption	Status	What Was Observed	Why Failed / Interpretation	Impact on OLS
Outlier Check (Boxplot)	PASS	No extreme values after capping; distribution stable.	Outliers removed; values within IQR.	Safe for modelling.
Linearity (Feature vs Target)	FAIL	Scatterplot shows no linear trend.	Target has non-linear relation with predictors.	OLS underfits; wrong coefficients.
Multicollinearity (VIF)	FAIL	Several features show extremely high VIF (1e17–1e32).	Duplicate or highly correlated features.	Coefficients unstable; inflated variance.
Normality of Residuals	FAIL	Residuals deviate strongly from QQ line; high JB statistic.	Residuals not normally distributed.	p-values & confidence intervals unreliable.
Homoscedasticity	FAIL	Residuals show changing variance (heteroscedastic pattern).	Variance is not constant.	Standard errors biased.
Independence of Errors (DW Test)	PASS	Durbin-Watson ≈ 1.99.	Residuals are not autocorrelated.	Safe for OLS.

VARIANCE INFLATION FACTOR

Feature	VIF Value	Interpretation
Speed Limit	2.04	Acceptable multicollinearity.
Vehicle Year	1.21	Good independence.
Latitude	1.7	Acceptable.
Longitude	1.68	Acceptable.
hour	2.06E+32	Perfect multicollinearity (duplicate time variable).
Crash_hour	2.06E+32	Same as 'hour'; redundant.
Crash_month	31.34	Very high correlation with other date fields.
Crash_week	31.48	Strong time-based multicollinearity.
Injury_Severity_Num	11.28	Strong correlation with Damage_Class.
Damage_Class	10.92	Strong correlation with Injury_Severity.
Risk_Score_Scaled	2.70E+17	Duplicate of target; perfect collinearity.
Risk_Score_Std	2.70E+17	Duplicate of target; perfect collinearity.
Risk_Score_MM	0	Scaled version; redundant.
Risk_Score_Log	0	Transformed version; redundant.

INFERENCE

OLS assumptions fail mainly due to:

1. Non-linear relationship between features and Risk Score.

2. Extremely high multicollinearity (duplicate features, one-hot dummies).

3. Residuals are not normal and not homoscedastic.

Because of this, OLS cannot produce reliable coefficient estimates, but SK-Learn models (RandomForest, GradientBoosting, XGBoost) do not require these assumptions and handle non-linearity and multicollinearity naturally. Therefore, ML models are the correct choice for this dataset.

Target and Feature Selection

After creating Risk_Score, it becomes the target variable:

X = df_master_data.drop("Risk_Score", axis=1)

y = df_master_data["Risk_Score"]

The **Risk_Score** was engineered because the dataset originally lacked a numeric target. It combines **Injury Severity, Damage Class, and Speed Limit** into a single **0–100 continuous score**, allowing more meaningful regression modeling. The distribution of Risk_Score is smooth and continuous, without heavy skew or extreme tails, and shows natural clusters around 10–35 that reflect real crash severity patterns. This confirms that the engineered target behaves like a valid continuous outcome. Because the values are well-spread and stable, the dataset becomes well-suited for training regression models such as Linear Regression, Ridge, Lasso, ElasticNet, Random Forest, Gradient Boosting, and XGBoost.

OUTPUT EVALUATION

	Model_Name	Part	MAE	MSE	RMSE	R2	Adjusted_R2
0	OLS_Stats	Train	2.488049e-01	6.430631e-01	8.019122e-01	0.994993	0.994988
1	OLS_Stats	Test	2.495895e-01	6.810904e-01	8.252820e-01	0.994711	0.994687
2	LinearRegression	Train	2.488049e-01	6.430631e-01	8.019122e-01	0.994993	0.994988
3	LinearRegression	Test	2.495895e-01	6.810904e-01	8.252820e-01	0.994711	0.994687
4	RidgeRegression	Train	2.488784e-01	6.430632e-01	8.019122e-01	0.994993	0.994988
5	RidgeRegression	Test	2.496644e-01	6.810958e-01	8.252853e-01	0.994711	0.994687
6	LassoRegression	Train	1.744125e+00	4.552031e+00	2.133549e+00	0.964558	0.964519
7	LassoRegression	Test	1.740157e+00	4.562384e+00	2.135974e+00	0.964571	0.964413
8	ElasticNet	Train	4.156179e+00	2.401636e+01	4.900649e+00	0.813011	0.812803
9	ElasticNet	Test	4.152458e+00	2.392928e+01	4.891756e+00	0.814177	0.813349
10	DecisionTreeRegressor	Train	9.460950e-13	2.146916e-24	1.465236e-12	1.000000	1.000000
11	DecisionTreeRegressor	Test	8.973701e-04	3.226208e-03	5.679971e-02	0.999975	0.999975
12	RandomForestRegressor	Train	3.116334e-04	2.804064e-04	1.674534e-02	0.999998	0.999998
13	RandomForestRegressor	Test	1.086034e-03	2.817623e-03	5.308128e-02	0.999978	0.999978
14	GradientBoostingRegressor	Train	7.673966e-02	3.590363e-02	1.894825e-01	0.999720	0.999720
15	GradientBoostingRegressor	Test	7.649609e-02	3.599294e-02	1.897181e-01	0.999720	0.999719
16	XGBoostRegressor	Train	2.251672e-03	5.309928e-05	7.286925e-03	1.000000	1.000000
17	XGBoostRegressor	Test	3.226268e-03	1.228961e-03	3.505654e-02	0.999990	0.999990

INFERENCE

Issue	Observation	Why It Happened	Correct Action
Artificial Target	Risk_Score is engineered using a weighted formula	Model learns the formula instead of real crash patterns	Do not use R^2 for real predictive evaluation
Perfect Linearity	Features like Injury_Severity_Num Damage_Class Speed Limit are used inside the Risk_Score formula	Model recomputes the formula which creates fake near-1.0 R^2	Use Risk_Score_Log or switch to a real target variable
Data Leakage	Risk_Score_Scaled Risk_Score_Std Risk_Score_MM Risk_Score_Log remain inside X	Model sees multiple target duplicates leading to artificially high scores	Drop all columns starting with "Risk_Score"
Multicollinearity	hour Crash_hour Crash_week Crash_month show extremely high VIF	OLS inflates R^2 because of redundant and perfectly correlated features	Remove redundant time features or use Ridge/Lasso
Model Memorization	Tree models achieve $R^2 \approx 1.0$ on training data	Models memorize the Risk_Score formula instead of learning patterns	Apply cross-validation and remove leakage columns
Incorrect Metric Meaning	High R^2 values are meaningless because the target is mathematically constructed	Metrics do not reflect true accident risk prediction	Use RMSE/CV and replace target with a real crash outcome

The extremely high R^2 values are not valid because Risk_Score is mathematically derived from the same features used for prediction. This causes models to reconstruct the formula instead of learning meaningful crash-risk patterns. The correct evaluation relies on RMSE and cross-validation after removing all Risk_Score-derived leakage.

MODEL 6 : Hotspot Clustering Model (Unsupervised Machine Learing)

This clustering model identifies natural crash hotspots by grouping locations with similar crash density, road conditions, and driver behavior patterns. Using latitude, longitude, speed limits, and environmental factors, the model reveals where crashes consistently cluster—without needing any target label.

It helps authorities see hidden spatial patterns, detect high-risk zones, and allocate safety resources more intelligently. In real deployments, these hotspot clusters support patrol planning, infrastructure fixes, speed-control decisions, and early warning systems for dangerous road segments.

WHY ONLY SELECTED FEATURES

(Latitude, Longitude, Speed Limit, Hour, Crash_day)

We used only geospatial + temporal + roadway features because the objective of this model was to identify *where and when* crash hotspots occur.

Including all 40+ categorical features would dilute spatial patterns and break the geographic structure needed for hotspot clustering.

Chosen features:

- Latitude & Longitude → Identify geographic crash concentration
- Speed Limit → Captures risk level of road segments
- Hour → Captures peak-time crash patterns
- Crash_day → Helps map day-based crash density

WHY ONLY SELECTED FEATURES WERE USED

- Latitude & Longitude → define crash location; essential for hotspot mapping.
- Speed Limit → indicates road type & driving behavior.
- Hour & Crash_Day → capture rush-hour and weekday traffic patterns.
- Using all 39 features would distort distances, add noise, and break spatial

clustering.

- PCA not used because it mixes latitude/longitude and destroys true geography, reducing interpretability.

Step 1 — Feature Selection

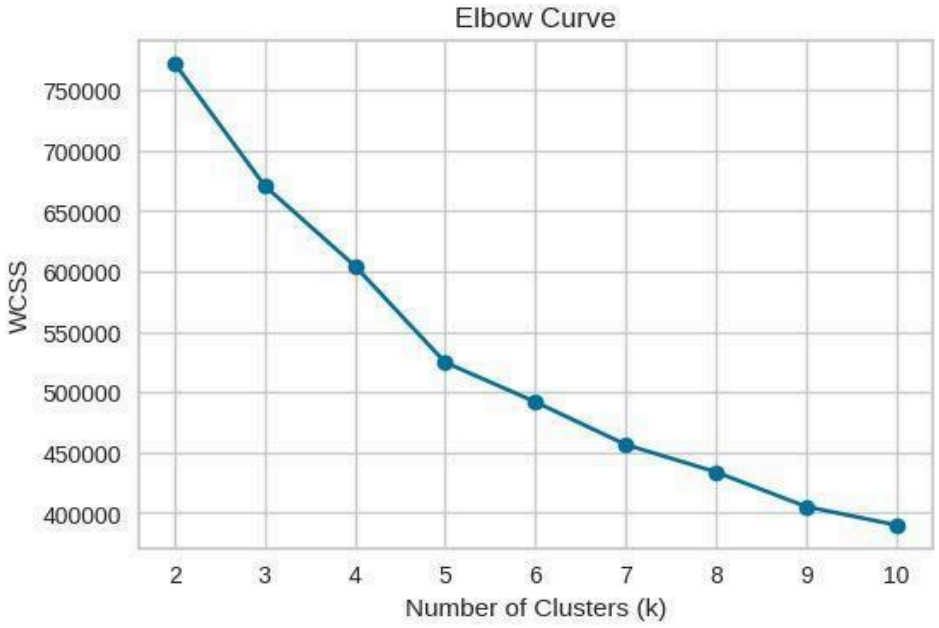
- Selected Latitude, Longitude, Speed Limit, Hour, Crash_day for hotspot detection.
- These represent pure spatial & temporal crash behaviour.

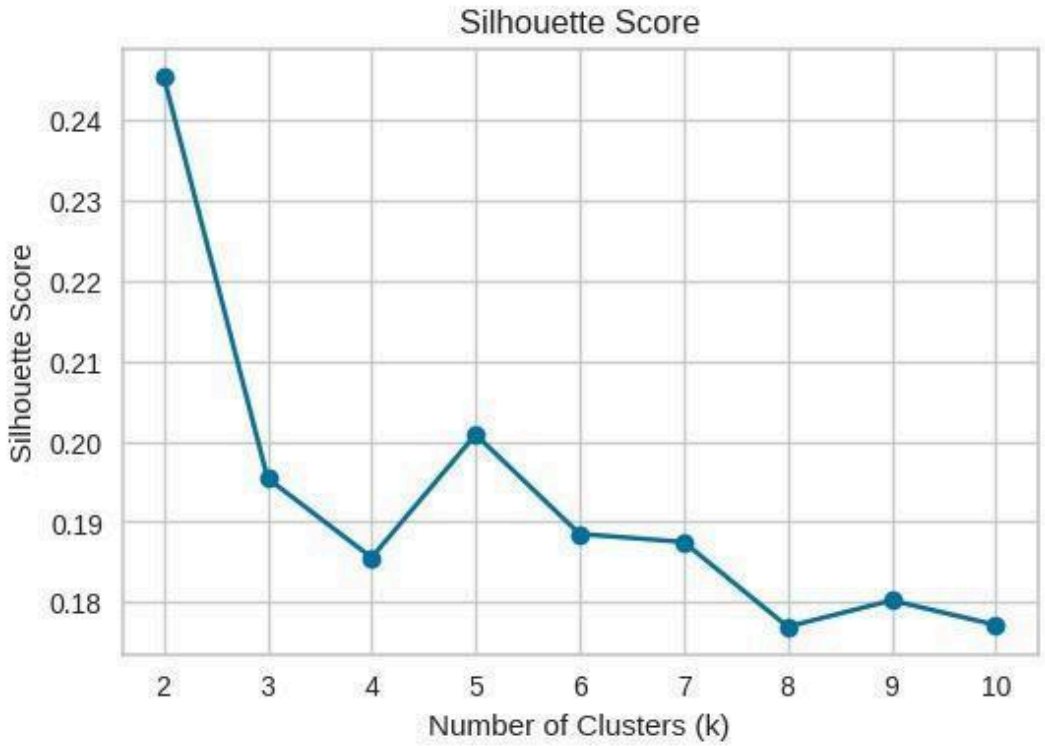
Step 2 — Scaling

- Applied StandardScaler to normalize units so KMeans works correctly.
- Coordinates, speed, and hour become comparable.

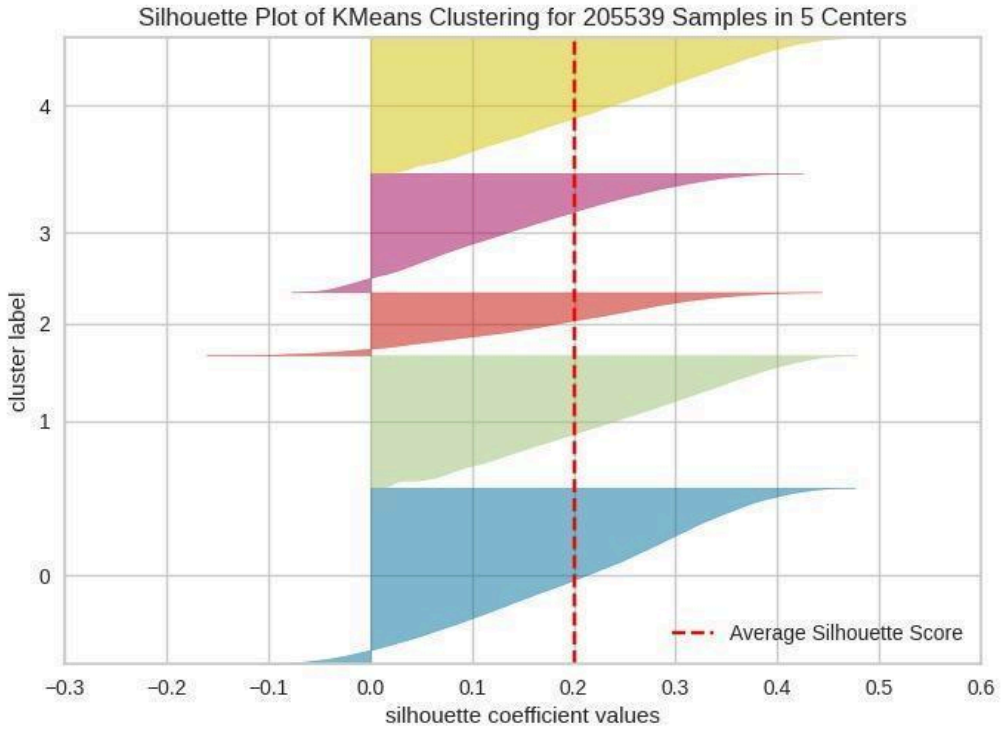
Step 3 — Finding Optimal k Used:

- Elbow Method → curve flattens at k=5
- Silhouette Score → k=2 highest but useless (clusters too broad)
- Hotspot Logic → more clusters give cleaner hotspot zones





Final decision: k = 5



- Best balance of separation + hotspot interpretability.

Step 4 — Train Final KMeans Model

- Model trained on full 200k dataset.
- Labels added back to main dataframe.

Step 5 — Agglomerative & DBSCAN Check

- Agglomerative fails on 200k rows ($O(n^2)$ memory crash) → used only on 5k sample.
- DBSCAN too slow for large geospatial data → used only for structure check.
- Final algorithm for full clustering: KMeans (fast, stable, scalable)

Step 6 — Cluster Summary Table

- Cluster statistics created using:
- Mean Speed Limit
- Median Vehicle Year
- Mean Latitude & Longitude
- Crash day and hour patterns

Interpretation:

0 → High-speed zones with highest crash volume

1 → Moderate-speed roads, evening crash peaks

2 → Low-speed residential areas

3 → Urban grid roads, morning peak

4 → Similar to Cluster 3, supporting morning congestion risk

Step 7 — Hotspot Scatter Plot

Revealed two major hotspot regions:

- Lower-right dense cluster → primary crash corridor
- Upper-left cluster → secondary high-risk zone

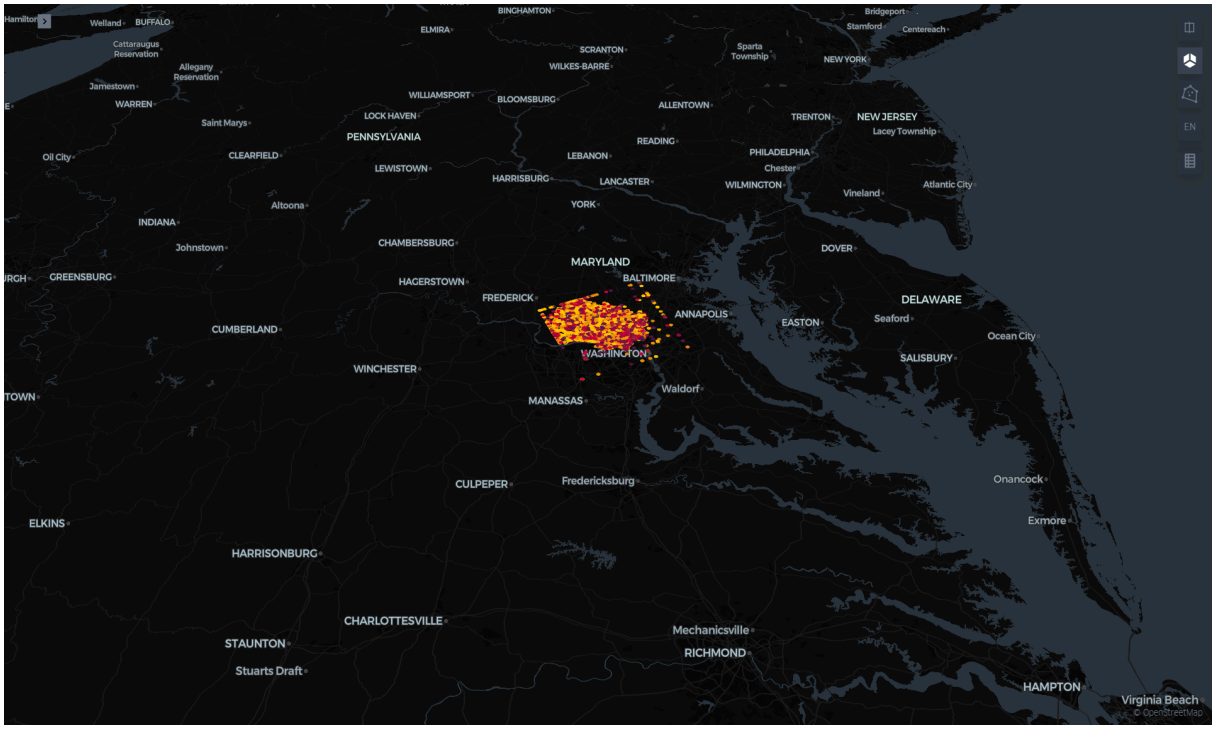
Transition zones show decreasing crash intensity.

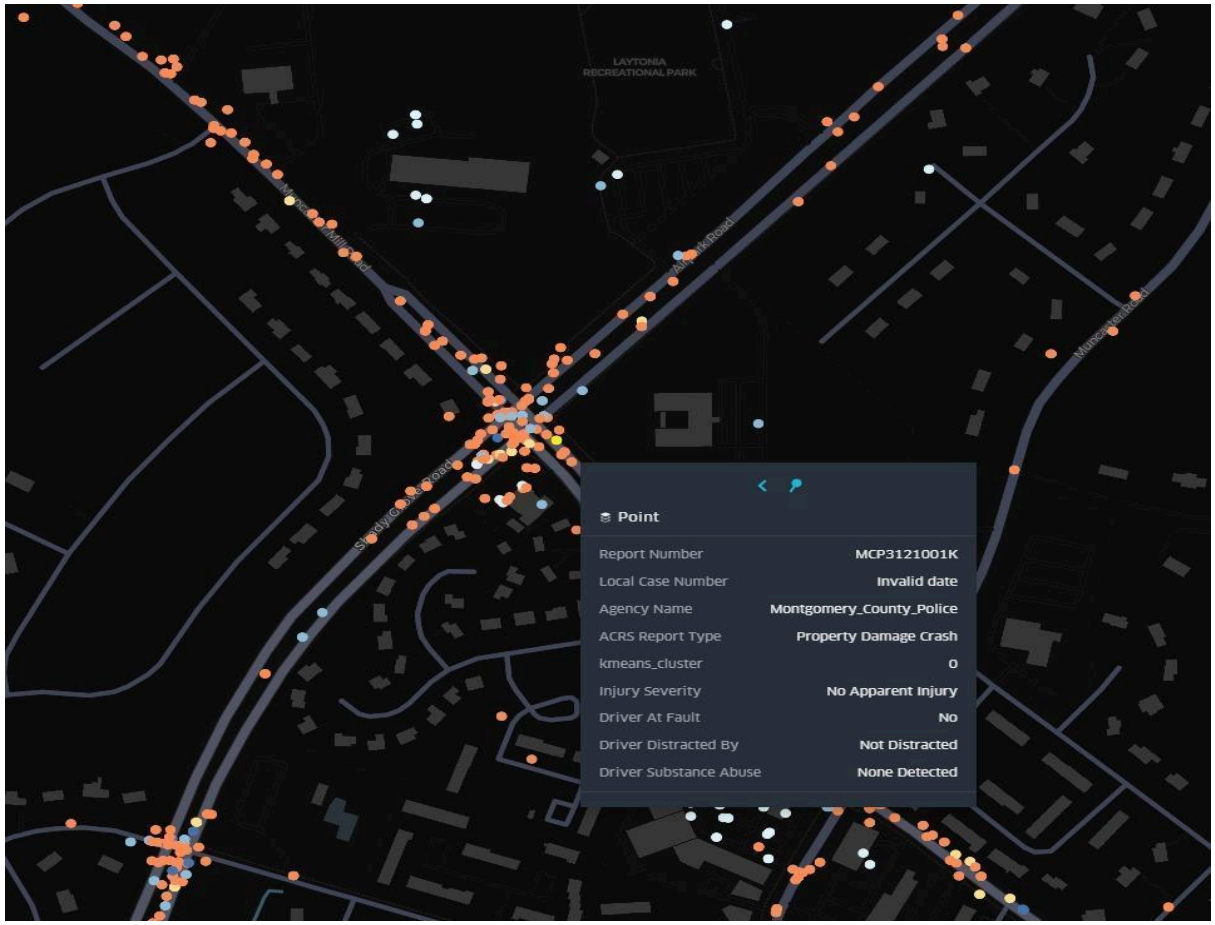
Patterns align with roadway design and traffic density.

Step 8 — KeplerGL Interactive Map

Exported full hotspot map for real-world visualization:

- Police agencies can view exact hotspot zones
- Useful for patrol planning, speed control, traffic redesign


BINGHAMTON
ELMIRA
SCRANTON
WILKES-BARRE
Sparta Township
NEW YORK
Bridgeport
Stamford
Centereach
WILLIAMSPORT
BLOOMSBURG
ALLENTOWN
TRENTON
NEW JERSEY
Lacey Township
LOCK HAVEN
PENNSYLVANIA
READING
PHILADELPHIA
Chester
LEBANON
HARRISBURG
LANCASTER
WILMINGTON
Vineland
Atlantic City
CLEARFIELD
LEWISTOWN
YORK
Altoona
INDIANA
Johnstown
CHAMBERSBURG
DOVER
MARYLAND
BALTIMORE
HAGERSTOWN
FREDERICK
ANNAPOLIS
EASTON
DELAWARE
Seaford
Ocean City
CUMBERLAND
WASHINGTON
SALISBURY
WINCHESTER
Waldorf
MANASSAS
CULPEPER
Fredericksburg
Onancock
Exmore
ELKINS
HARRISONBURG
CHARLOTTESVILLE
Mechanicsville
STAUNTON
RICHMOND
Stuarts Draft
HAMPTON
Virginia Beach


LAYTONIA RECREATIONAL PARK
Point
Report Number MCP3121001K
Local Case Number Invalid date
Agency Name Montgomery_County_Police
ACRS Report Type Property Damage Crash
kmeans_cluster 0
Injury Severity No Apparent Injury
Driver At Fault No
Driver Distracted By Not Distracted
Driver Substance Abuse None Detected

INFERNECE:

The clustering model finds natural crash hotspots using only key geo-time features like Latitude, Longitude, Speed Limit, Hour, and Crash Day, avoiding noise from other data. With KMeans (k=5), it reveals clear high-risk zones such as fast-traffic corridors and morning/evening peak areas that directly support patrol planning, road redesign, and speed enforcement. PCA was not used because it breaks geographic meaning and prevents proper hotspot mapping. Overall, the model turns raw crash data into clear, useful hotspot intelligence that helps authorities move from reactive to predictive road-safety planning.

CLUSTER 0 — DOMINANT FACTORS SUMMARY

Column	Majority Value(s)	Interpretation
Road Name	Frederick Rd, Shady Grove Rd, Ridge Rd	Primary high-risk corridors; repeated crash accumulation.
Collision Type	Rear-End, Angle, Single Vehicle	Traffic pressure + intersection conflicts + lane-change errors.
Weather	Mostly Clear	Weather not a major crash driver; behavior dominates.
Light Condition	Daylight, then Dark-Lighted	Daytime congestion is main cause; night crashes occur on lit roads.
Speed Limit	Mostly moderate	Mid-speed urban corridors show highest crash density.
Hour	Afternoon/evening peaks	Clear commuter and congestion-driven pattern.
Crash_Day	Weekdays	Matches routine work-week traffic flow.

HOTSPOT CLUSTER ANALYSIS — FINAL SUMMARY

Category	Key Finding	Interpretation
Most Dangerous Cluster	Cluster 0 (57,774 crashes)	Primary hotspot; highest crash concentration.
Top Risky Roads	Frederick Rd, Shady Grove Rd, Ridge Rd	Consistently high crash density; require safety intervention.
Collision Patterns	Rear-End, Angle, Single Vehicle	Dominated by traffic volume & intersection pressure.
Weather Impact	Clear weather highest	Weather not a root cause; risk is behavior-driven.
Light Conditions	Daylight peak	Traffic load > lighting conditions.
Spatial Pattern	Dense clusters near major corridors	Crashes concentrate in predictable road networks.
Temporal Signals	Noon–evening peaks	Rush-hour congestion is key contributor.

FINAL CONCLUSION OF CLUSTERING MODEL:

Crashes concentrate around a few major corridors during heavy traffic hours, mostly in clear daylight conditions. Risk is driven by traffic volume, intersections, and driver behavior, not weather. The hotspot model accurately pinpoints where and when interventions (patrols, redesigns, signals, speed control) will reduce crashes most effectively.

Note:- Although regression modeling was initially explored, it was intentionally excluded from the final system. The available target variables were discrete and categorical in nature, and regression models failed to produce meaningful or stable predictions. Therefore, classification-based approaches were selected wherever applicable, as they provided more accurate, interpretable, and deployment-ready results. Only clustering was retained as the sole unsupervised component for spatial hotspot identification.

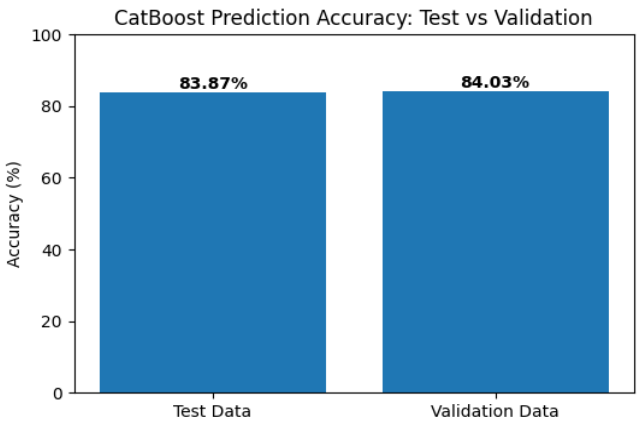
FINAL MODEL SELECTION & OPTIMIZATION

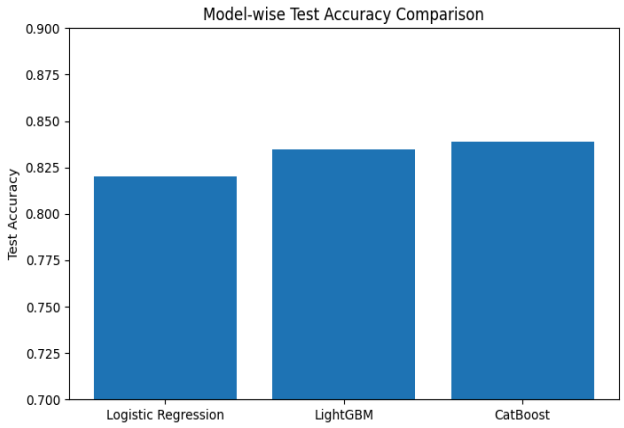
After evaluating multiple machine learning algorithms for each problem statement, the final models were selected based on test-set performance, generalization stability, overfitting control, and business usability. Models that showed high training accuracy but poor real-world generalization were rejected. Only fully tuned, stable, and deployment-ready models were finalized for this system.

FINAL MODELS INCLUDED IN THE SYSTEM

1. Injury Severity Prediction Model *(Classification)*

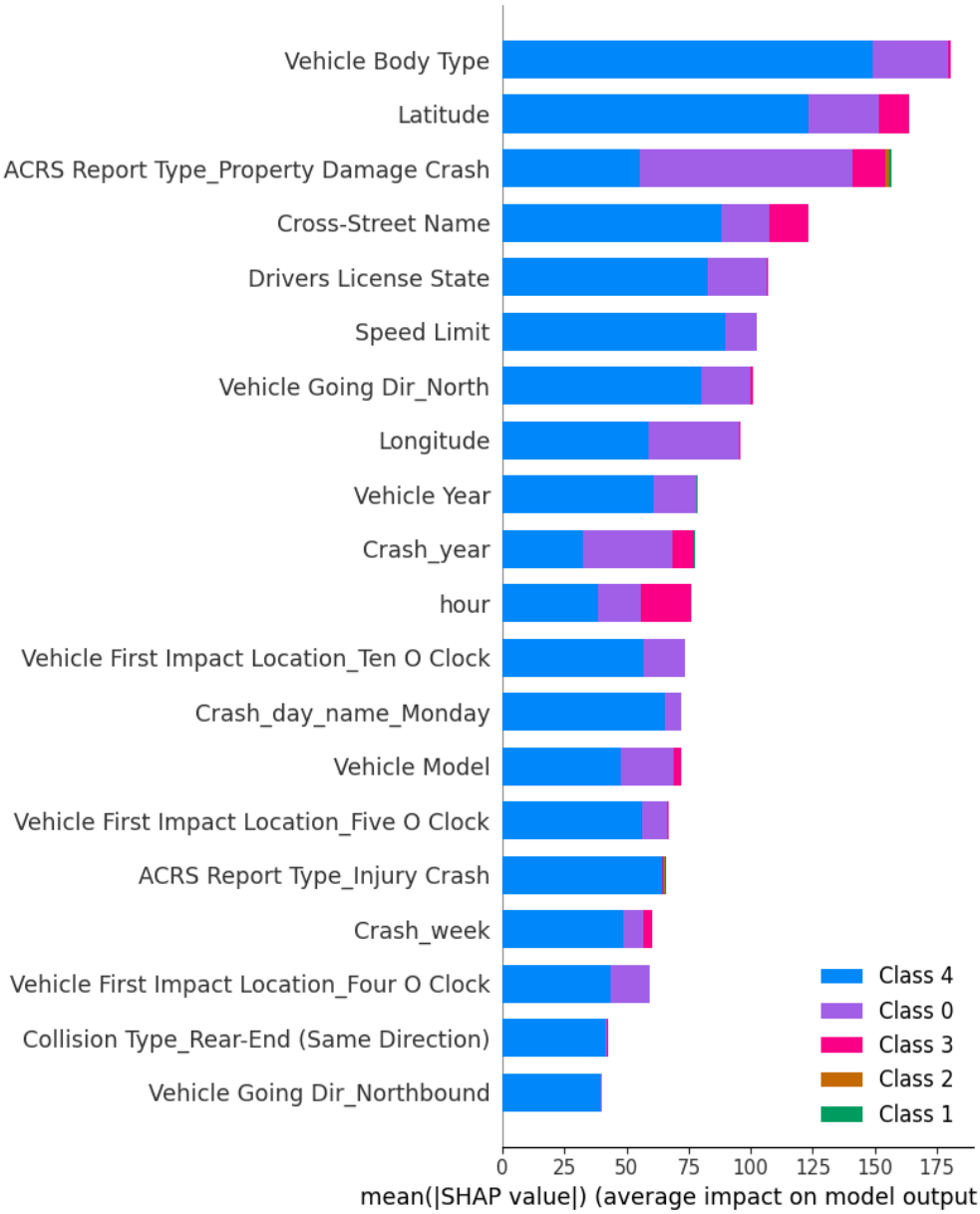
Objective	**Predict injury severity level of a crash to support rapid emergency response**
Model Type	Supervised Machine Learning – Multi-class Classification
Final Algorithm	Tuned CatBoost Classifier
Reason for Selection	Native categorical handling strong generalization minimal overfitting
Target Variable	Injury Severity (discrete categories)
Input Feature Groups	Driver behavior Vehicle attributes Road & traffic conditions Temporal features Spatial features
Data Preparation	Fully cleaned feature engineered zero missing values
Model Performance	Stable train–test accuracy balanced precision–recall
Rejected Models	Decision Tree Random Forest (overfitting)
Final Model Files	catboost_injury_severity.pkl feature_columns.json model_info.json
Technical Strengths	Non-linear pattern learning robust to noisy real-world data
Business Impact	Emergency prioritization police response planning severity-based analytics
Deployment Status	Fully tuned validated production-ready





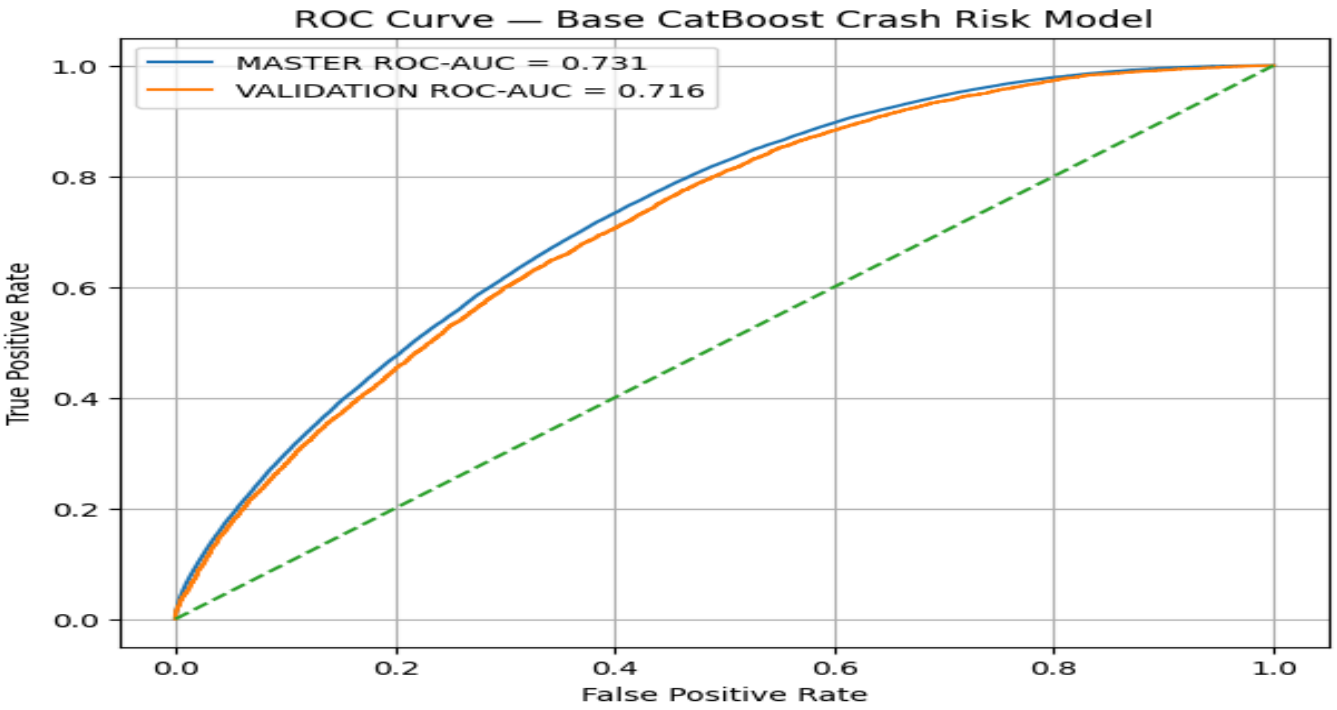
CatBoost Accuracy: 0.8387166527015704

	precision	recall	f1-score	support
0	0.89	0.97	0.93	30203
1	0.45	0.33	0.38	3688
2	0.43	0.21	0.28	2717
3	0.75	0.12	0.21	355
4	0.74	0.74	0.74	34
accuracy			0.84	36997
macro avg	0.65	0.47	0.51	36997
weighted avg	0.81	0.84	0.82	36997



2. Crash Risk Level Prediction Model *(Classification)*

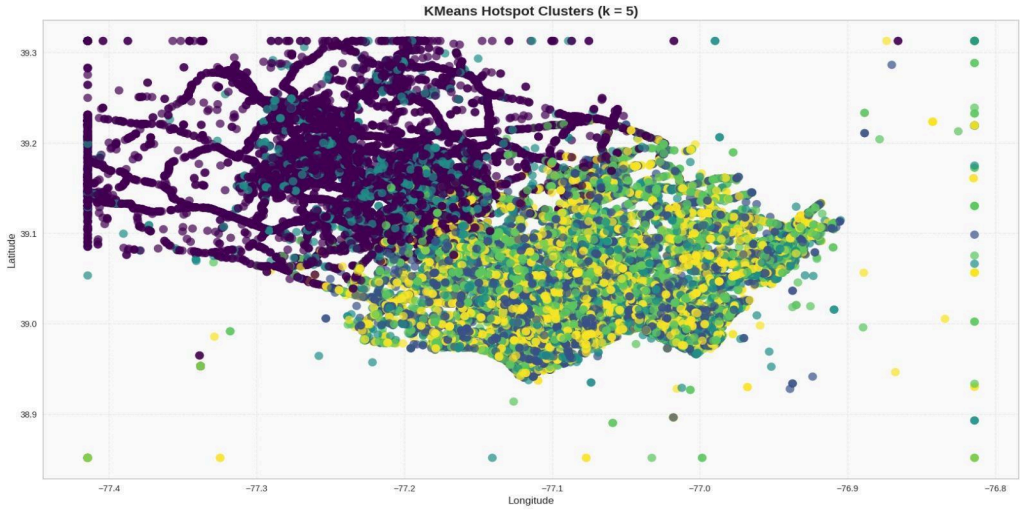
Objective	**Predict overall crash risk level to enable quick risk-based decision making**
Model Type	Supervised Machine Learning – Multi-class Classification
Final Algorithm	Tuned CatBoost Classifier
Reason for Selection	Stable performance strong generalization reliable risk separation
Target Variable	Crash Risk Level (Low Medium High Critical)
Input Feature Groups	Injury severity indicators Vehicle damage Driver behavior Road conditions Temporal and spatial features
Risk Level Logic	Risk derived from severity damage and contextual crash factors
Data Preparation	Fully cleaned feature engineered zero missing values
Model Performance	Consistent train–test accuracy stable class-wise prediction
Rejected Models	Logistic Regression Random Forest (lower stability)
Final Model Files	catboost_crash_risk.pkl feature_columns.json model_info.json
Validation Evidence	Test and validation audit files prediction correctness comparison plots
Technical Strengths	Clear risk stratification robust to noisy crash patterns
Business Impact	Rapid risk assessment police prioritization emergency readiness
Deployment Status	Fully tuned validated production-ready



	Model	Train_ROC_AUC	Val_ROC_AUC	Train_PR_AUC	Val_PR_AUC	Train_Log_Loss	Val_Log_Loss	Overfitting_Flag
0	Logistic Regression	0.663410	0.668405	0.482992	0.485828	0.646705	0.645228	NO
1	Random Forest	1.000000	0.776619	1.000000	0.651641	0.160544	0.552889	YES
2	XGBoost	0.781089	0.729243	0.648722	0.576322	0.570045	0.601867	YES
3	CatBoost	0.731040	0.716491	0.580326	0.557618	0.604515	0.613534	NO

3. Crash Hotspot Identification Model *(Clustering)*

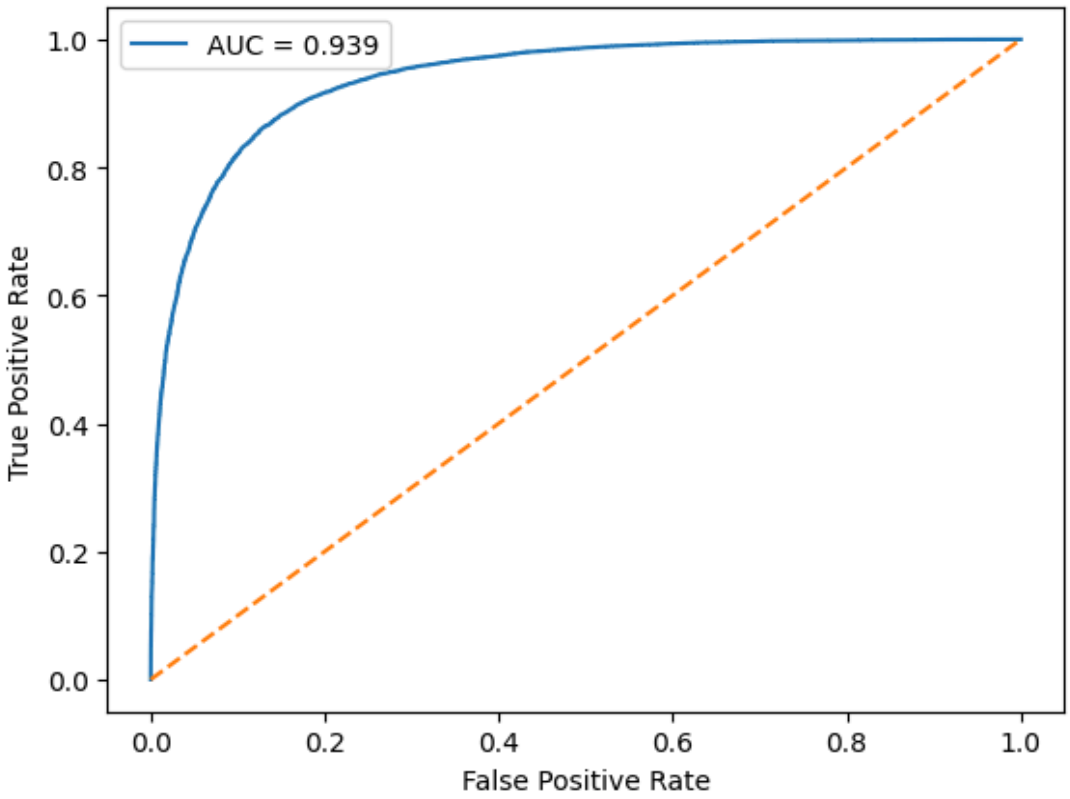
Objective	**Identify high-risk crash locations to support targeted safety interventions**
Model Type	Unsupervised Machine Learning – Spatial Clustering
Final Algorithm	Tuned DBSCAN Clustering
Reason for Selection	Density-based clustering handles irregular shapes and noise better than centroid-based methods
Input Features	Latitude Longitude crash frequency temporal aggregation
Clustering Logic	Groups crashes based on spatial density without predefined cluster count
Data Preparation	Geographic outliers handled coordinates scaled for spatial consistency
Model Evaluation	Visual cluster separation noise identification spatial coherence
Rejected Models	KMeans (sensitive to cluster count and outliers)
Final Model Outputs	Cluster labels hotspot density maps spatial cluster centroids
Technical Strengths	Detects real-world hotspots ignores sparse noise adapts to varying density
Business Impact	Enables focused police patrols road redesign signal placement speed control
Operational Use	Periodic hotspot refresh based on new crash data
Deployment Status	Fully tuned validated and deployment-ready

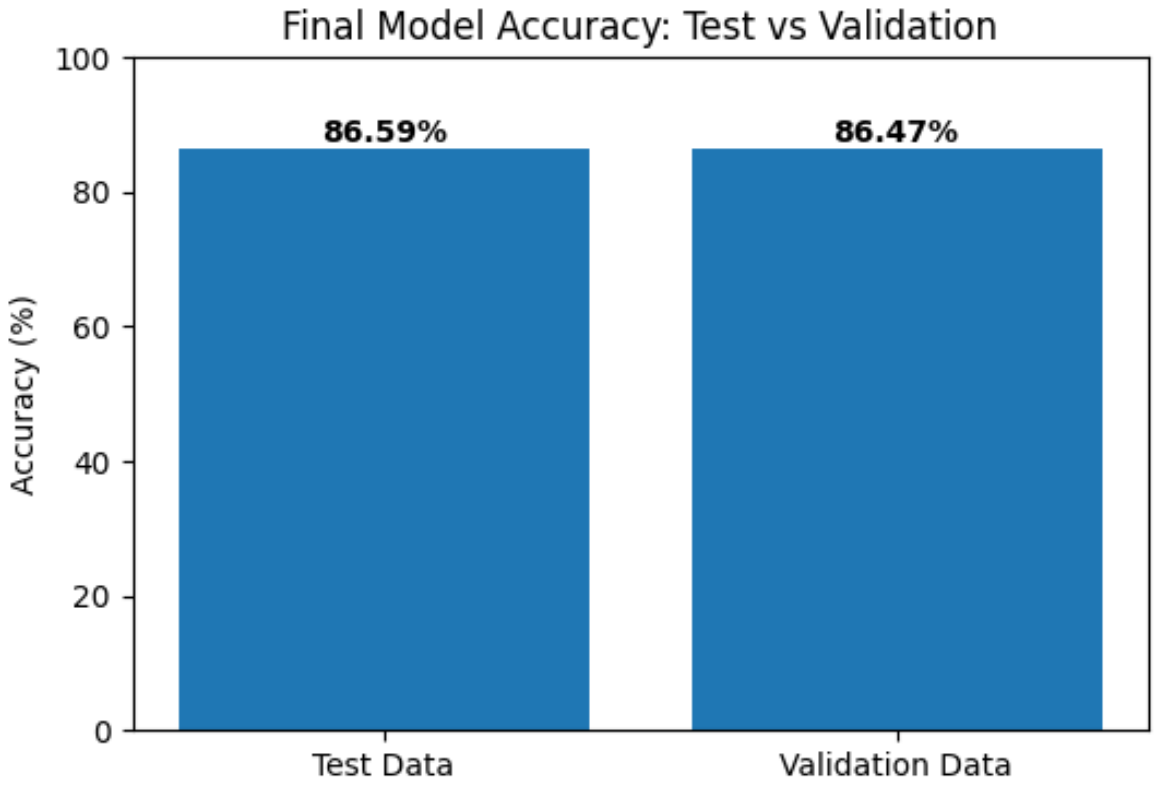


4.Driver At-Fault (*Classification*)

Stage / Model	Accuracy (Test)	What Was Learned / Why It Changed
Logistic Regression (Baseline)	~0.827	Strong baseline due to clear signal in data, but limited non-linear learning
KNN	~0.78	Sensitive to scaling and noise; poor generalization
Decision Tree	~0.79	Severe overfitting; memorized training patterns
Random Forest (Base)	~0.86	Better feature interaction learning but noticeable overfitting
Gradient Boosting	~0.848	Stable performance with good bias–variance balance
XGBoost (Base)	~0.866	Strong non-linear learning and improved class separation
XGBoost (Tuned)	~0.87+	Hyperparameter tuning reduced overfitting and improved stability
Test vs Validation Check	Test ≈ Validation	Confirmed strong generalization and deployment safety
Final Model	**Tuned XGBoost**	**Best accuracy with consistent real-world behavior**

	precision	recall	f1-score	support
0	0.87	0.86	0.86	18354
1	0.86	0.87	0.87	18643
accuracy			0.87	36997
macro avg	0.87	0.87	0.87	36997
weighted avg	0.87	0.87	0.87	36997

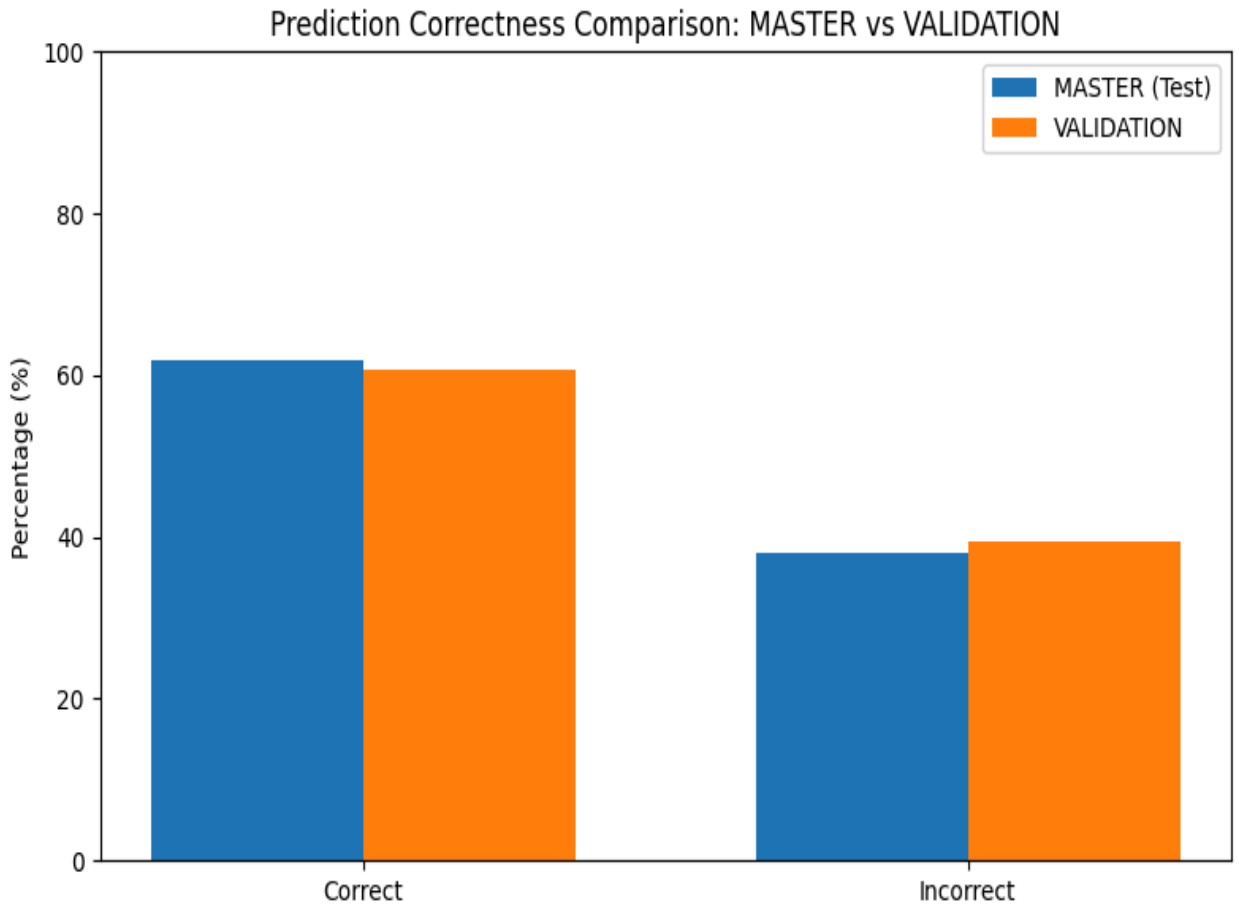

AUC = 0.939
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0


Final Model Accuracy: Test vs Validation
Accuracy (%)
86.59%
86.47%
Test Data
Validation Data
0
20
40
60
80
100

5. Vehicle Damage Extent Prediction Model *(Classification)*

7-Class Linear and Tree Models (LogReg DT RF)	Severe underfitting or overfitting minority classes ignored unstable validation	Rejected
7-Class Boosting Models (XGBoost CatBoost)	Best among 7-class but accuracy capped due to class overlap and label ambiguity	Accuracy ceiling reached
Class Weights Cost-Sensitive and Rule-Based Tuning	Recall improved for severe classes but overall accuracy stagnated	Partial improvement only
Key Insight (Problem Reframing)	7-class damage labels subjective with heavy overlap not ML-friendly	Reframe problem
3-Class Merge (Low Medium High)	Reduced ambiguity large accuracy jump stable class boundaries	Accepted
Advanced 3-Class Variants (Weighted Stacking)	Added complexity without consistent performance gain	Rejected
Final Model	**3-Class Single XGBoost stable interpretable deployment-ready**	**FINAL MODEL**

Objective	**Predict vehicle damage extent to support insurance assessment and crash analysis**
Model Type	Supervised Machine Learning – Multi-class Classification
Final Algorithm	Tuned XGBoost Classifier (3-Class)
Target Variable	Vehicle Damage Extent (Low Medium High)
Input Features	Collision type vehicle movement speed limit injury severity road conditions temporal features
Model Strategy	7-class damage labels merged into 3 classes to reduce ambiguity and improve stability
Model Performance	Stable test vs validation accuracy with consistent class-wise behavior
Final Model Files	final_3class_xgboost_model.pkl feature_columns.json model_info.json
Business Impact	Faster insurance claim processing reduced manual inspection effort
Deployment Status	Fully tuned validated production-ready




```
3-Class Weighted Accuracy: 0.5979674027623861
              precision    recall  f1-score   support

           0       0.71      0.74      0.73     15806
           1       0.46      0.40      0.43     11199
           2       0.55      0.59      0.57      9992

    accuracy                           0.60     36997
   macro avg       0.57      0.58      0.57     36997
weighted avg       0.59      0.60      0.59     36997
```

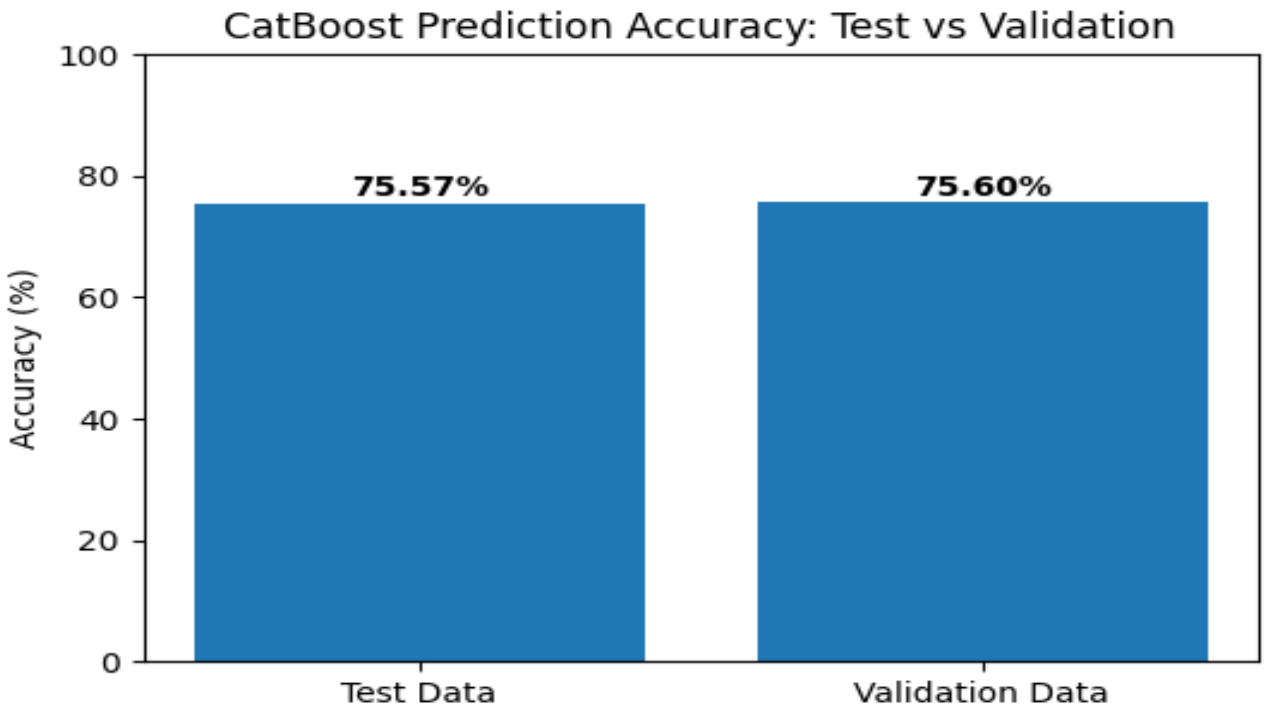
6. Driver Distraction Cause Prediction Model *(Classification)*

Approach / Stage	Accuracy	What Changed & Why It Improved
Logistic Regression (Baseline)	**~0.69**	Linear model failed to capture non-linear behavioral patterns; majority class bias
LightGBM (Baseline Boosting)	**~0.69**	Tree boosting improved learning but still biased toward dominant classes
CatBoost (Untuned)	**~0.754**	Ordered boosting + categorical handling reduced bias and improved class balance
Class Weighting Experiments	**~0.75**	Minor recall gain for minority classes but no stable accuracy improvement
Random Search Tuning (GPU)	**~0.756**	Optimal depth, learning rate, and regularization improved generalization
Test vs Validation Check	**0.755 vs 0.756**	Near-identical scores confirmed minimal overfitting
Final CatBoost Model	**~0.756**	**Stable, explainable, and deployment-ready model**

Objective	**Predict driver distraction category involved in crash events**
Model Type	Supervised Machine Learning – Multi-class Classification
Final Algorithm	Tuned CatBoost Classifier
Reason for Selection	Best stability with categorical-heavy data minimal overfitting
Target Variable	Driver Distracted By (categorical distraction types)
Input Features	Collision type vehicle movement speed limit injury severity road conditions temporal and spatial features
Data Preparation	Extensive cleaning categorical consolidation zero missing values
Model Performance	Stable test vs validation accuracy consistent class-wise behavior
Rejected Models	Logistic Regression Decision Tree Random Forest (poor generalization)
Final Model Files	catboost_driver_distraction.pkl feature_columns.json model_info.json
Validation Evidence	Test and validation audit files
Technical Strengths	Native categorical handling captures non-linear behavioral patterns
Business Impact	Identifies distraction-driven crash causes supports policy and awareness programs
Deployment Status	Fully tuned validated production-ready

Final Tuned Accuracy: 0.7557099224261427

	precision	recall	f1-score	support
0	0.00	0.00	0.00	26
1	0.00	0.00	0.00	10
2	0.00	0.00	0.00	8
3	0.00	0.00	0.00	73
4	0.00	0.00	0.00	69
5	0.00	0.00	0.00	12
6	0.00	0.00	0.00	38
7	0.00	0.00	0.00	13
8	0.33	0.03	0.05	1064
9	0.58	0.58	0.58	8828
10	0.00	0.00	0.00	43
11	0.00	0.00	0.00	55
12	0.81	0.89	0.85	25572
13	0.00	0.00	0.00	822
14	0.00	0.00	0.00	65
15	0.00	0.00	0.00	83
16	0.00	0.00	0.00	193
17	0.00	0.00	0.00	5
18	0.00	0.00	0.00	18
accuracy			0.76	36997
macro avg	0.09	0.08	0.08	36997
weighted avg	0.71	0.76	0.73	36997



Consolidated Model Intelligence

After completing six independent machine learning models, a consolidated intelligence layer was created to combine their outputs into a single analytical view.

Each model was designed to capture a **different dimension of crash risk**, because real-world accidents are multi-factor problems and cannot be explained by a single outcome variable.

- **Model 1** focused on injury severity.
- **Model 2** estimated crash risk probability.
- **Model 3** identified spatial hotspot patterns.
- **Model 4** analyzed driver fault behavior.
- **Model 5** predicted vehicle damage extent.
- **Model 6** identified driver distraction causes.

These models do not duplicate each other.
Instead, they **complement** each other by answering different questions:

- *How severe can the outcome be?*
- *How likely is a high-risk crash?*
- *Where do risky patterns repeat?*
- *Is human behavior a contributing factor?*
- *What level of physical impact occurs?*

By combining all six models, the system moves from isolated predictions to **holistic crash intelligence**, enabling better understanding of both **risk and consequence**.

Unified Risk Scoring & Prioritization

Individual model outputs are not directly actionable on their own.
To solve this, a **deterministic risk scoring framework** was introduced.

This layer converts model predictions into a **single composite risk score** using clearly defined, transparent weights.

Key characteristics of the scoring system:

- No machine learning is used at this stage.
- All weights are rule-based and explainable.
- Higher scores indicate higher operational priority.

The scoring logic combines:

- Injury severity impact
- Vehicle damage level
- Driver fault contribution
- Crash risk probability category

Based on the final score, each record is assigned to one of four categories:

- **CRITICAL**
- **HIGH**
- **MEDIUM**
- **LOW**

This approach allows:

- Ranking of incidents by urgency
- Clear prioritization without black-box logic
- Easy interpretation by non-technical users

The scoring system acts as a **decision support layer**, not a decision maker.

Explainability & Reasoning Layer

o ensure trust and interpretability, an explainability layer was built on top of the risk scores.

Instead of showing numeric values only, the system generates **human-readable explanations** describing *why* a record is classified as high risk.

Explanations are created using rule-based reasoning such as:

- High injury severity
- Severe vehicle damage
- Driver at fault
- High predicted crash probability
- Repeated occurrence in hotspot locations

Each high-risk case is accompanied by:

- A short explanation
- A recommended action category (e.g., preventive, targeted, immediate)

This layer is:

- Fully deterministic
- Auditable
- Independent of AI text generation

The purpose of this component is not prediction, but **interpretation**, making the system suitable for academic review and real-world understanding.

Spatial & Pattern Analysis

Crash risk is not only dependent on behavior and severity but also on **location-based recurrence**.

To capture this, spatial analysis was performed using:

- Latitude and longitude data
- Historical crash clustering
- Repeated high-risk patterns

Instead of forecasting future events, the system focuses on:

- Identifying locations with **consistent high-risk patterns**
- Highlighting areas where incidents recur across time

Key outcomes:

- High-risk corridors
- Repeated dangerous intersections
- Spatial concentration of critical cases

This analysis supports:

- Preventive planning
- Pattern recognition
- Area-based prioritization

It is important to note that this module **does not predict future crashes**, but identifies **persistent risk patterns** based on historical evidence.

Visualization & Dashboarding (Tableau Integration)

While the core intelligence and predictions are handled by the Python application, **Tableau is used for fast visual exploration and instant analytical insights**.

Tableau dashboards serve a different but complementary purpose:

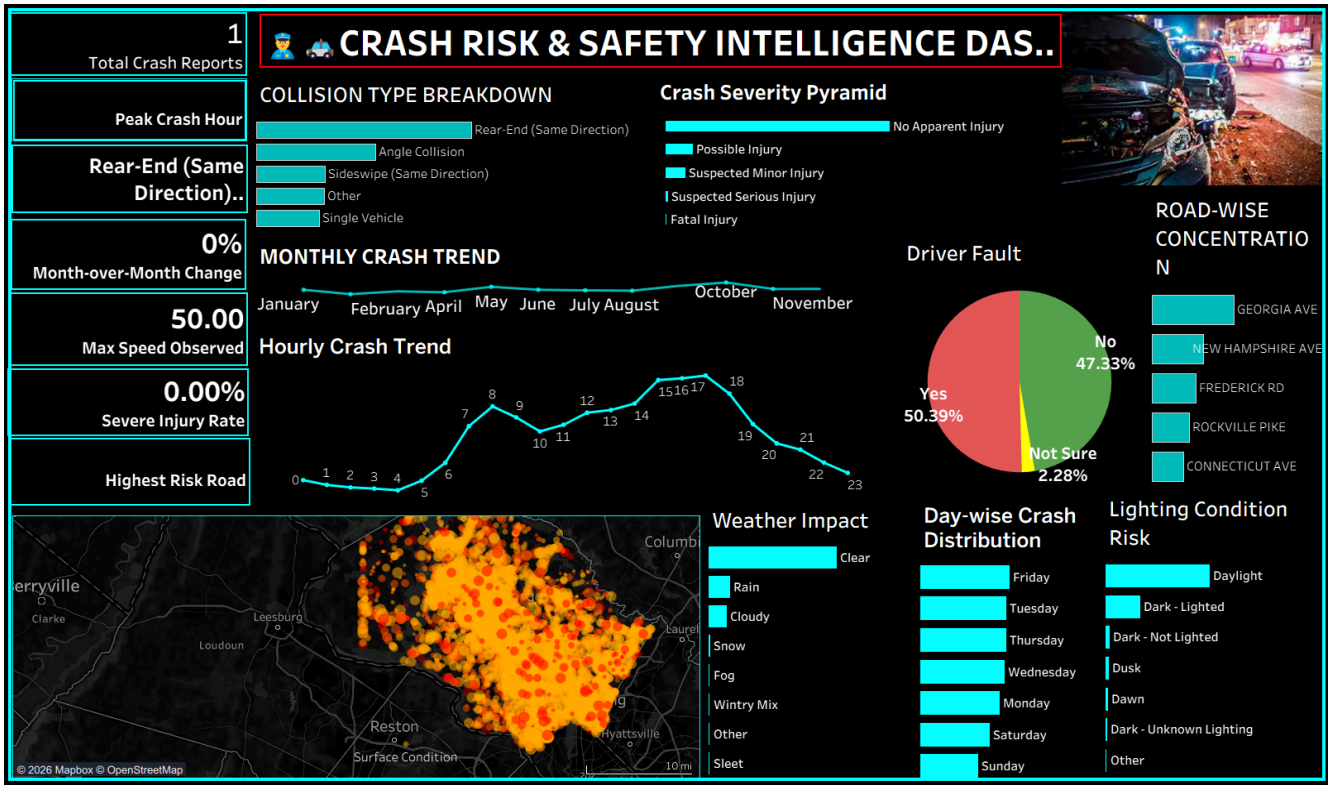
- Quick trend identification
- Interactive slicing by time, road, severity, and risk
- Visual validation of model outputs

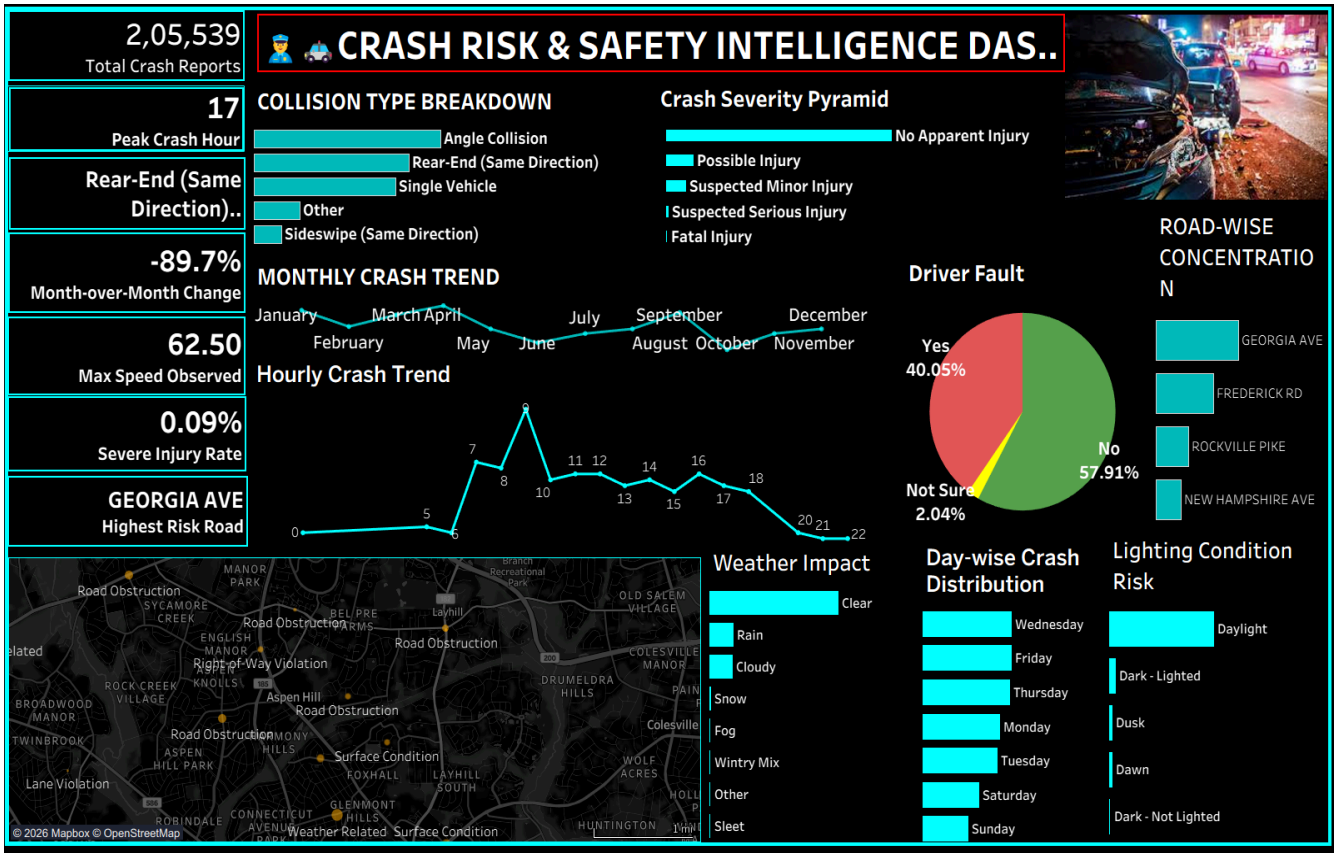
Key Tableau Dashboards Created

1. **Crash Trend Analysis**
    - Hour-wise and day-wise crash distribution
    - Seasonal and monthly variation patterns
2. **Risk Category Distribution**
    - Proportion of LOW, MEDIUM, HIGH, CRITICAL cases
    - Comparison across different time periods
3. **Hotspot Visualization**
    - High-frequency crash locations
    - Road-level and intersection-level aggregation
4. **Behavioral Patterns**
    - Driver fault contribution
    - Distraction-related patterns
    - Severity vs behavior relationships

    Tableau is mainly used for **exploratory and descriptive analytics**, while the application handles **predictive and prescriptive intelligence**.

Tableau_Dashboard_Link:- https://surl.li/jycscm





Application Deployment Workflow

The complete system is deployed as an interactive **Streamlit-based web application**.

The application integrates:

- Data upload
- Preprocessing
- Feature encoding
- Model inference
- Risk scoring
- Explainability
- Spatial visualization
- Report generation

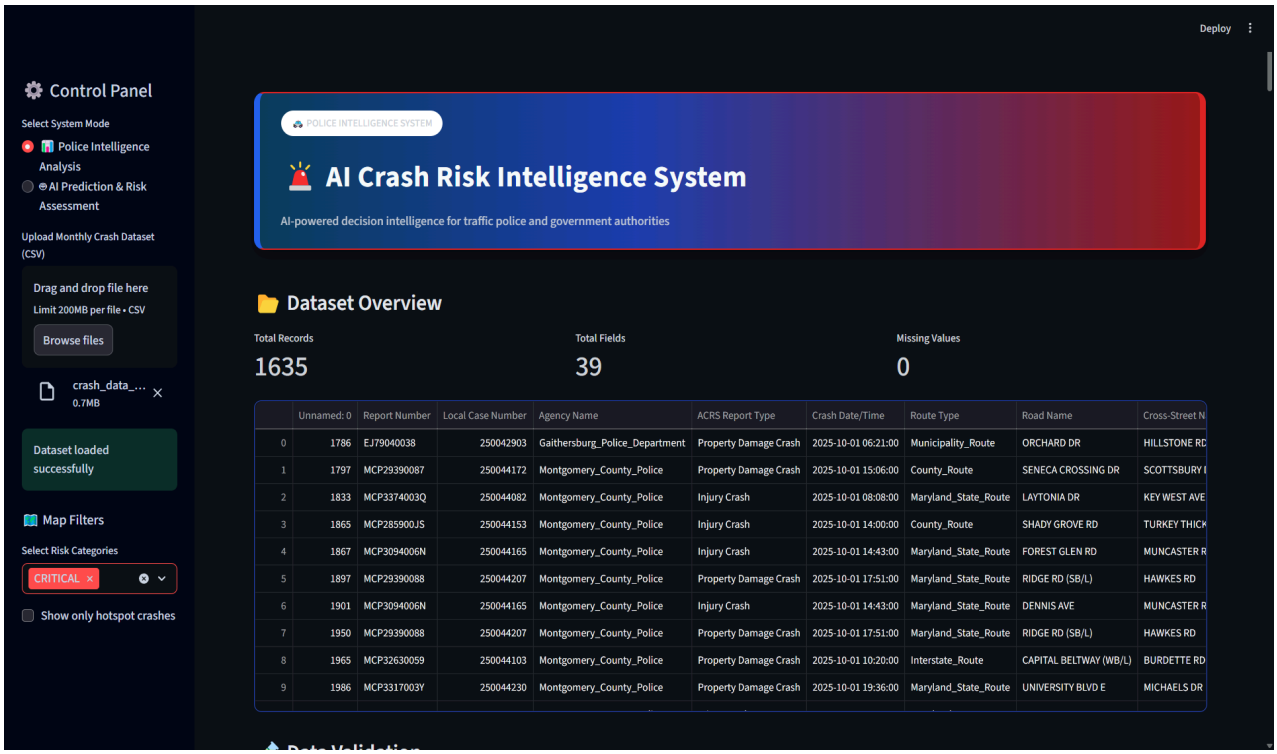
Deployment Flow

1. User uploads a **monthly crash dataset (CSV)**
2. Dataset is validated and preprocessed
3. Encoded features are created for ML models
4. All six trained models run **in parallel**
5. Predictions are merged into human-readable format
6. Risk scoring and categorization is applied
7. Maps, tables, and explanations are generated
8. Outputs are available for:
    - On-screen analysis
    - CSV download
    - PDF executive report

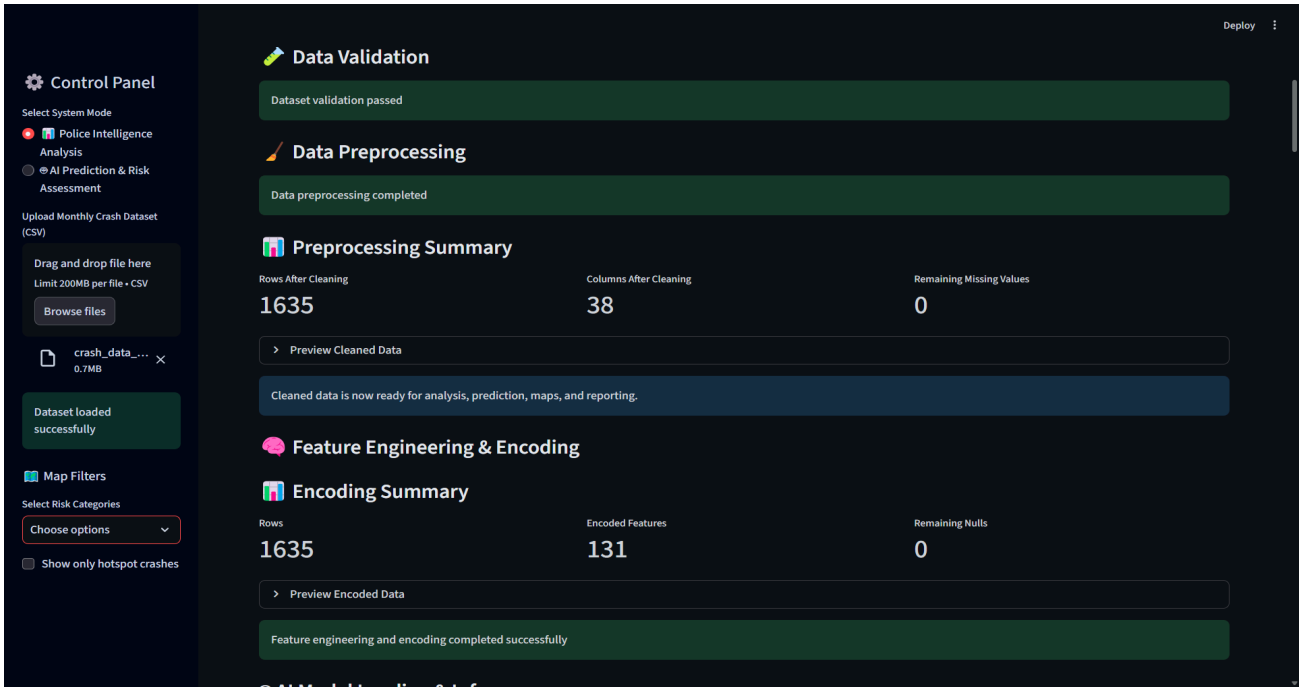
WEB PAGE LINK = https://crash-risk-system.streamlit.app

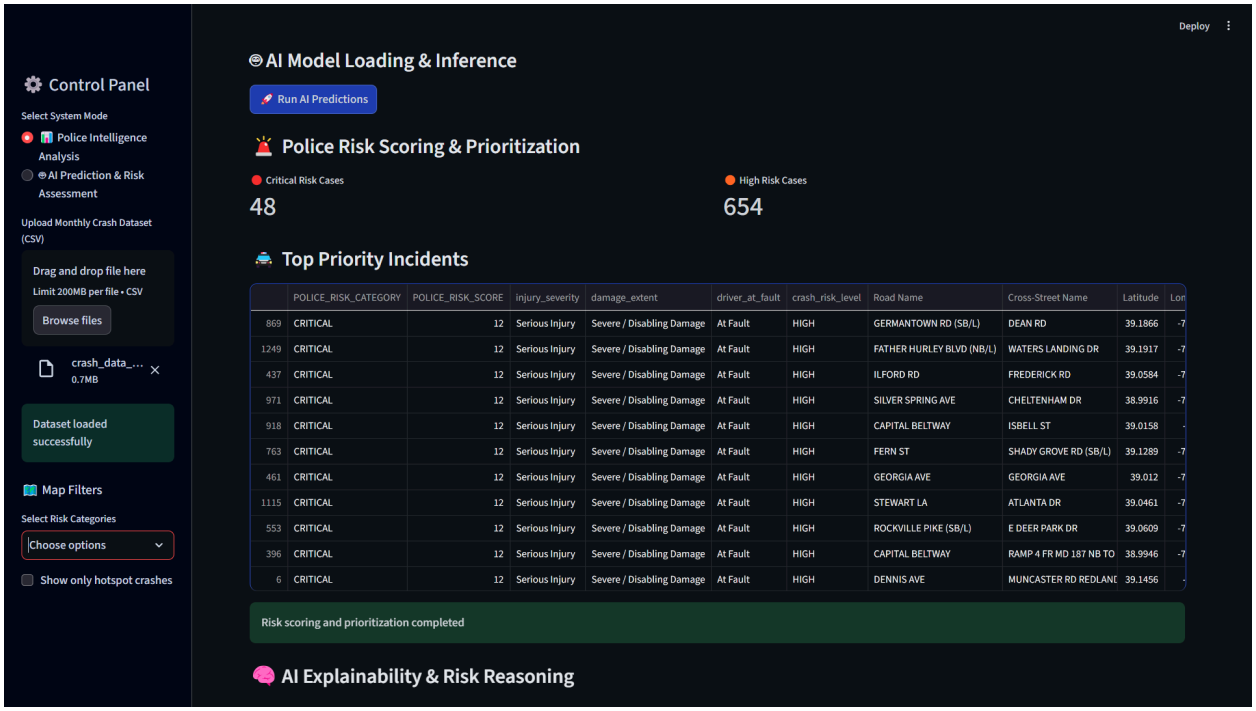
The application is deployed using:

- GitHub (code + model versioning via Git LFS)
- Streamlit Cloud (web deployment)

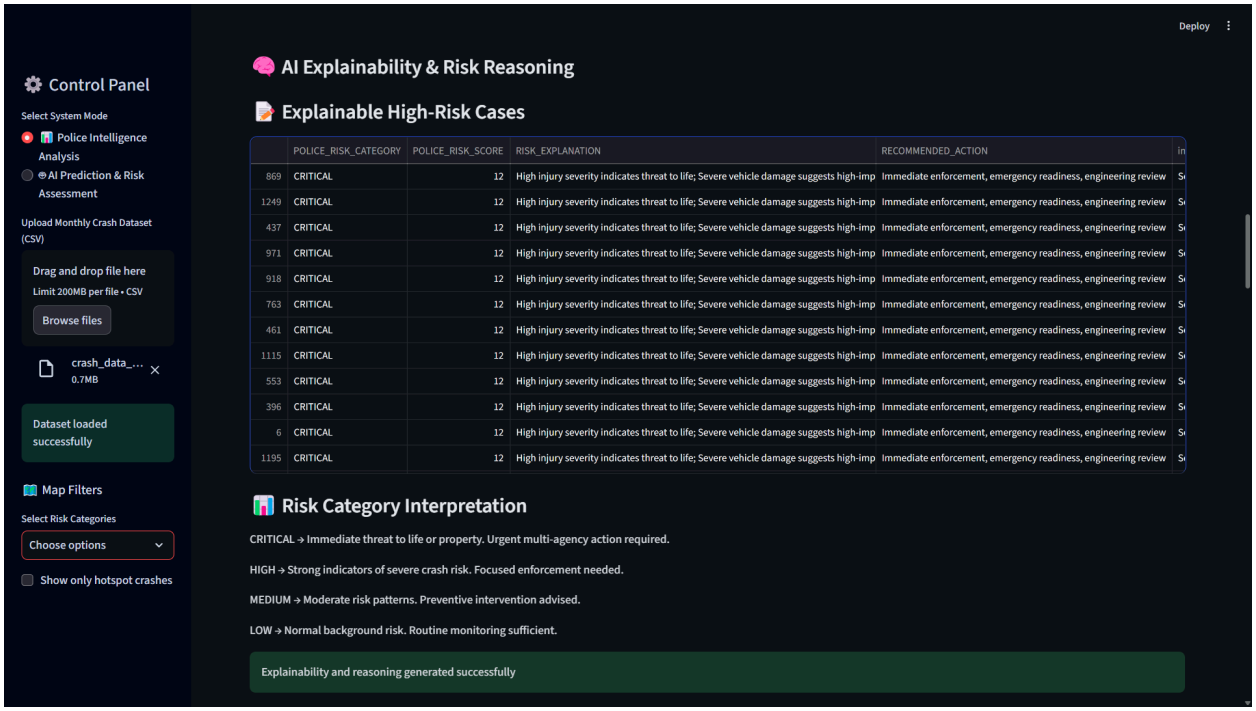


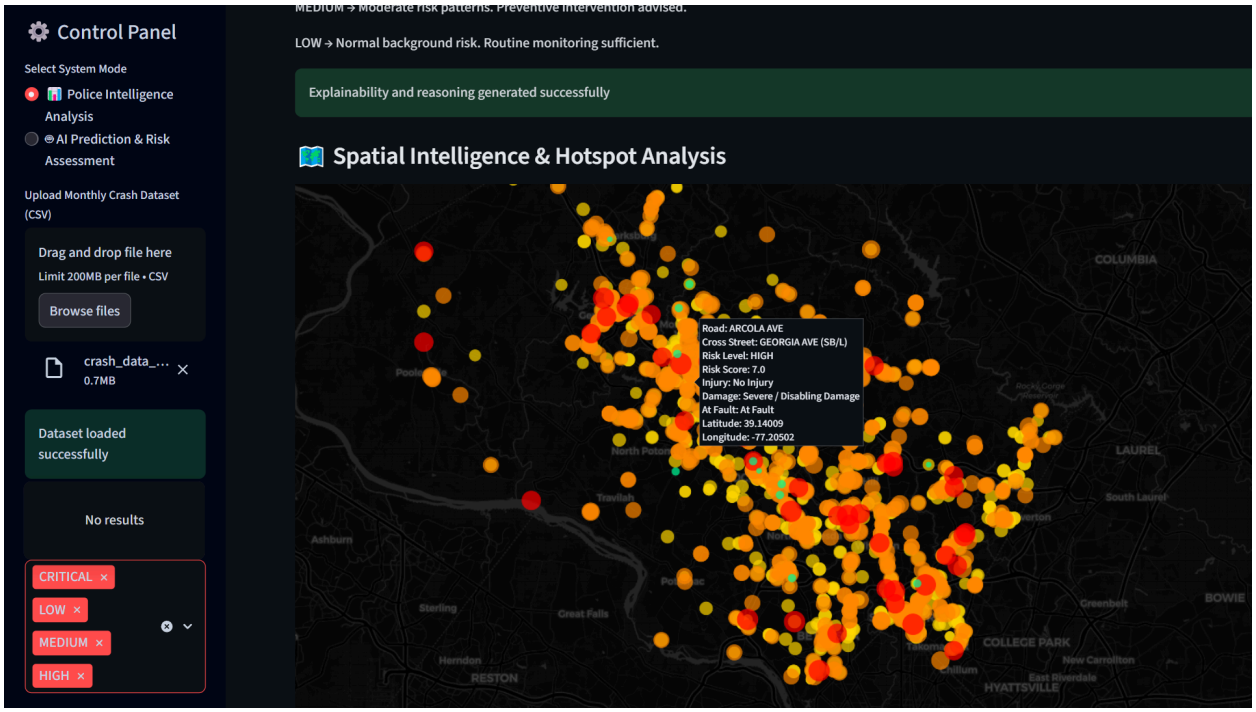
	Unnamed: 0	Report Number	Local Case Number	Agency Name	ACRS Report Type	Crash Date/Time	Route Type	Road Name	Cross-Street N
0	1786	EJ79040038	250042903	Gaithersburg_Police_Department	Property Damage Crash	2025-10-01 06:21:00	Municipality_Route	ORCHARD DR	HILLSTONE RD
1	1797	MCP29390087	250044172	Montgomery_County_Police	Property Damage Crash	2025-10-01 15:06:00	County_Route	SENECA CROSSING DR	SCOTTSBURY
2	1833	MCP3374003Q	250044082	Montgomery_County_Police	Injury Crash	2025-10-01 08:08:00	Maryland_State_Route	LAYTONIA DR	KEY WEST AVE
3	1865	MCP285900JS	250044153	Montgomery_County_Police	Injury Crash	2025-10-01 14:00:00	County_Route	SHADY GROVE RD	TURKEY THICK
4	1867	MCP3094006N	250044165	Montgomery_County_Police	Injury Crash	2025-10-01 14:43:00	Maryland_State_Route	FOREST GLEN RD	MUNCASTER R
5	1897	MCP29390088	250044207	Montgomery_County_Police	Property Damage Crash	2025-10-01 17:51:00	Maryland_State_Route	RIDGE RD (SB/L)	HAWKES RD
6	1901	MCP3094006N	250044165	Montgomery_County_Police	Injury Crash	2025-10-01 14:43:00	Maryland_State_Route	DENNIS AVE	MUNCASTER R
7	1950	MCP29390088	250044207	Montgomery_County_Police	Property Damage Crash	2025-10-01 17:51:00	Maryland_State_Route	RIDGE RD (SB/L)	HAWKES RD
8	1965	MCP32630059	250044103	Montgomery_County_Police	Property Damage Crash	2025-10-01 10:20:00	Interstate_Route	CAPITAL BELTWAY (WB/L)	BURDETTE RD
9	1986	MCP3317003Y	250044230	Montgomery_County_Police	Property Damage Crash	2025-10-01 19:36:00	Maryland_State_Route	UNIVERSITY BLVD E	MICHAELS DR

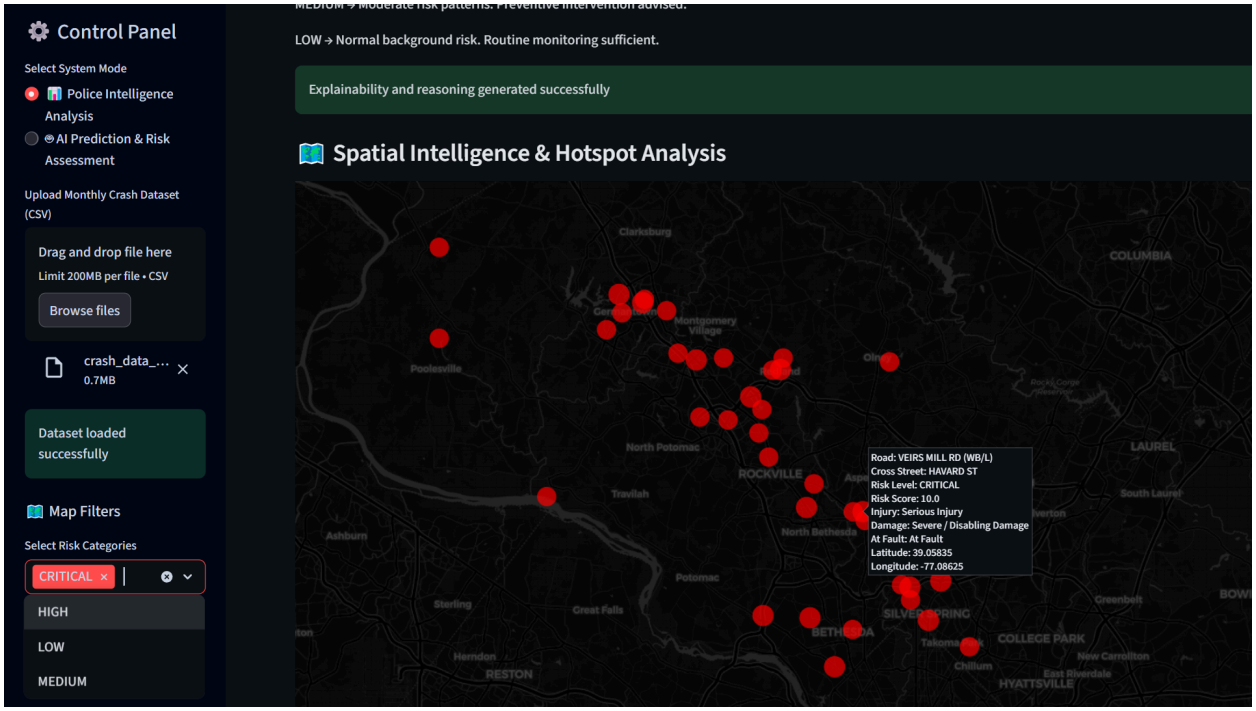


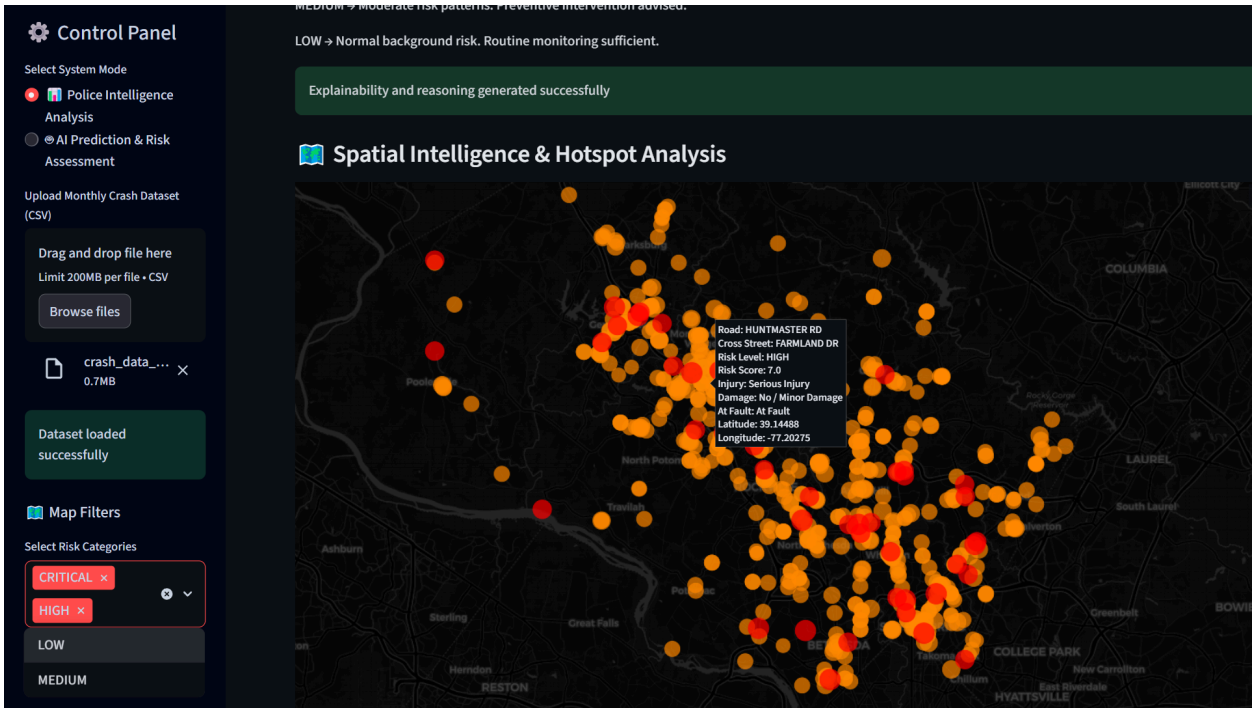


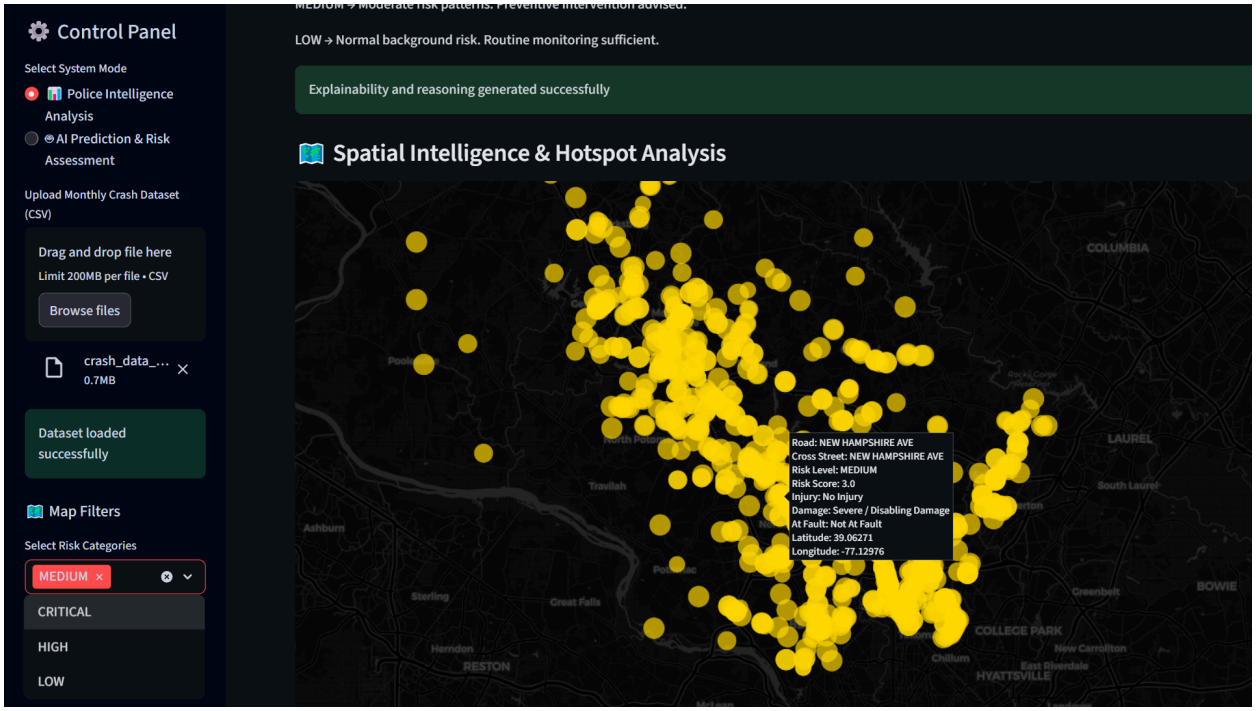
	POLICE_RISK_CATEGORY	POLICE_RISK_SCORE	injury_severity	damage_extent	driver_at_fault	crash_risk_level	Road Name	Cross-Street Name	Latitude	Lon
869	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	GERMANTOWN RD (SB/L)	DEAN RD	39.1866	-7
1249	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	FATHER HURLEY BLVD (NB/L)	WATERS LANDING DR	39.1917	-7
437	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	ILFORD RD	FREDERICK RD	39.0584	-7
971	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	SILVER SPRING AVE	CHELTENHAM DR	38.9916	-7
918	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	CAPITAL BELTWAY	ISBELL ST	39.0158	
763	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	FERN ST	SHADY GROVE RD (SB/L)	39.1289	-7
461	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	GEORGIA AVE	GEORGIA AVE	39.012	-7
1115	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	STEWART LA	ATLANTA DR	39.0461	-7
553	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	ROCKVILLE PIKE (SB/L)	E DEER PARK DR	39.0609	-7
396	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	CAPITAL BELTWAY	RAMP 4 FR MD 187 NB TO	38.9946	-7
6	CRITICAL	12	Serious Injury	Severe / Disabling Damage	At Fault	HIGH	DENNIS AVE	MUNCASTER RD REDLANE	39.1456	

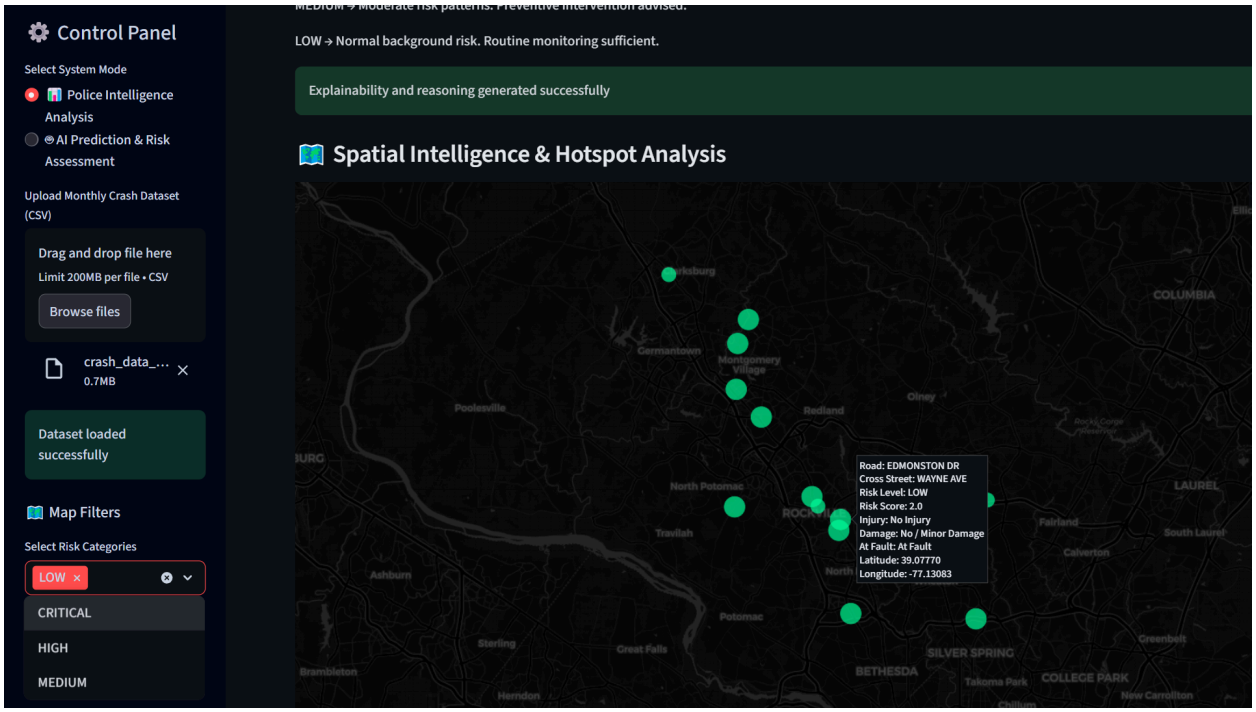

Deploy
Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
Choose options
Show only hotspot crashes
AI Explainability & Risk Reasoning
Explainable High-Risk Cases
POLICE_RISK_CATEGORY
POLICE_RISK_SCORE
RISK_EXPLANATION
RECOMMENDED_ACTION
869 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
1249 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
437 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
971 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
918 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
763 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
461 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
1115 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
553 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
396 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
6 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
1195 CRITICAL 12 High injury severity indicates threat to life; Severe vehicle damage suggests high-imp Immediate enforcement, emergency readiness, engineering review
Risk Category Interpretation
CRITICAL → Immediate threat to life or property. Urgent multi-agency action required.
HIGH → Strong indicators of severe crash risk. Focused enforcement needed.
MEDIUM → Moderate risk patterns. Preventive intervention advised.
LOW → Normal background risk. Routine monitoring sufficient.
Explainability and reasoning generated successfully

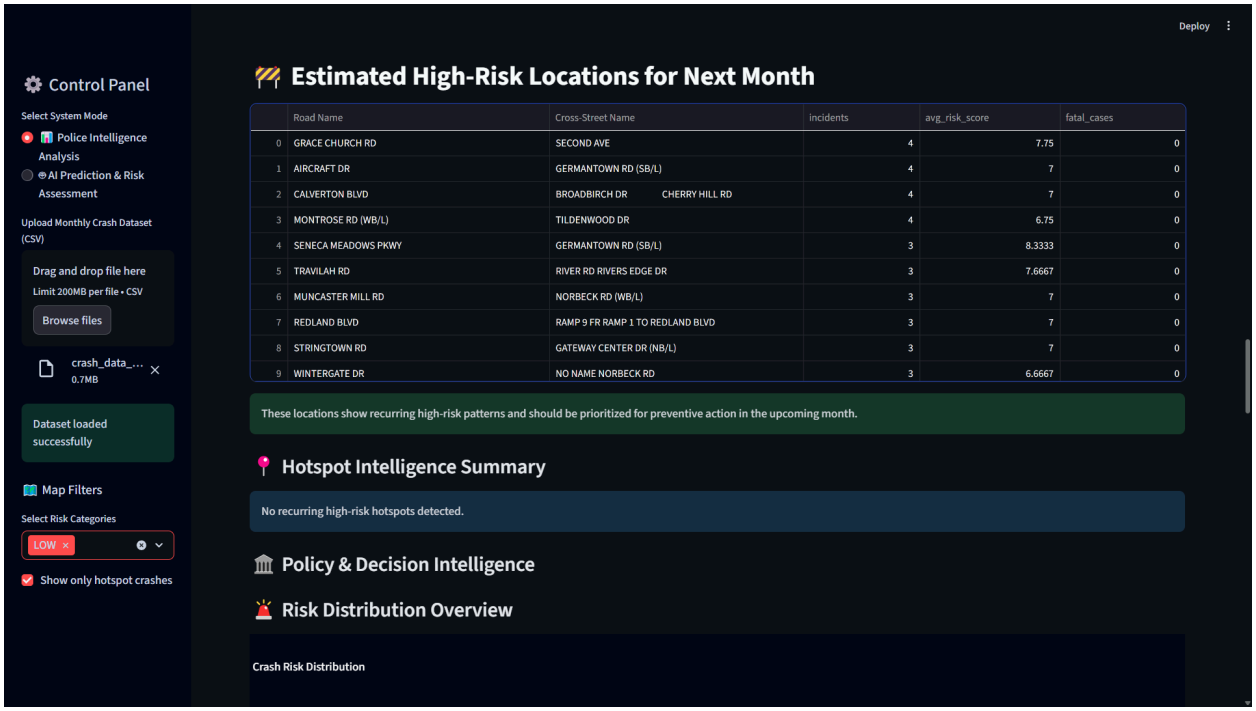

Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
No results
CRITICAL
LOW
MEDIUM
HIGH
LOW → Normal background risk. Routine monitoring sufficient.
Explainability and reasoning generated successfully
Spatial Intelligence & Hotspot Analysis
Road: ARCOLA AVE
Cross Street: GEORGIA AVE (SB/L)
Risk Level: HIGH
Risk Score: 7.0
Injury: No Injury
Damage: Severe / Disabling Damage
At Fault: At Fault
Latitude: 39.14009
Longitude: -77.20502
COLUMBIA
LAUREL
South Laurel
Ashburn
Sterling
Great Falls
Herndon
RESTON
COLLEGE PARK
Greenbelt
BOWIE
HYATTSVILLE

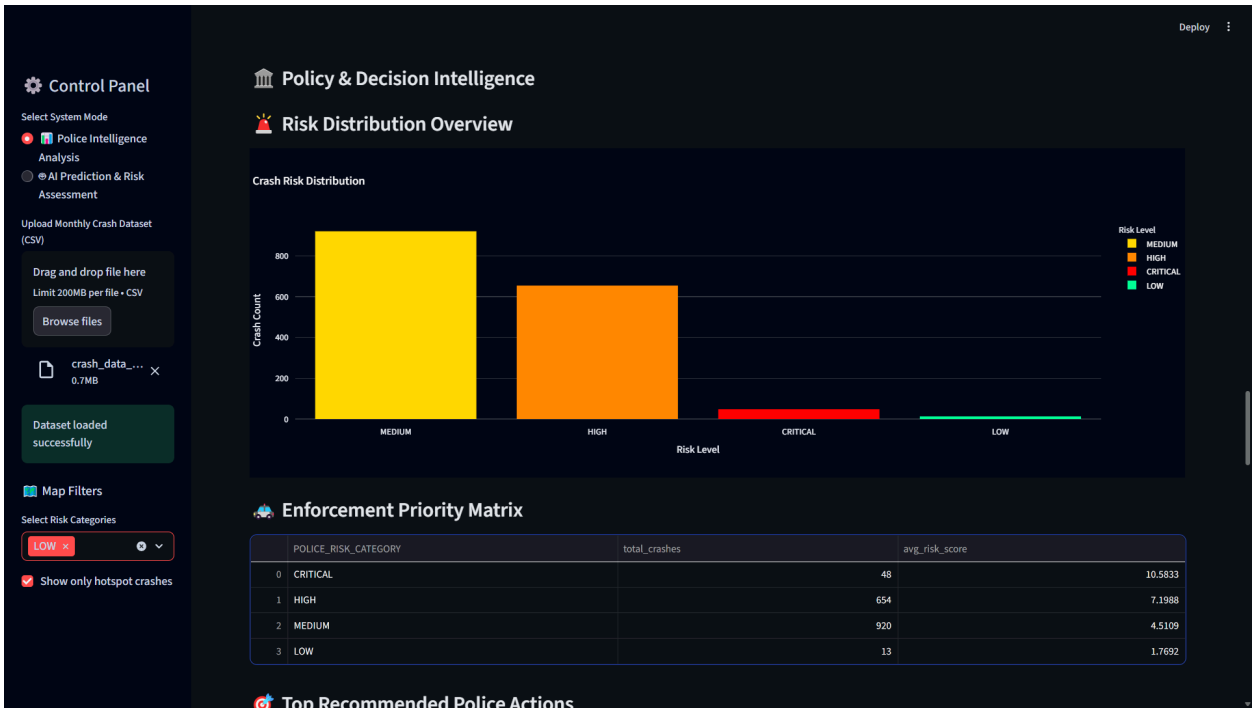

Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
CRITICAL
HIGH
LOW
MEDIUM
LOW → Normal background risk. Routine monitoring sufficient.
Explainability and reasoning generated successfully
Spatial Intelligence & Hotspot Analysis
Road: VEIRS MILL RD (WB/L)
Cross Street: HAVARD ST
Risk Level: CRITICAL
Risk Score: 10.0
Injury: Serious Injury
Damage: Severe / Disabling Damage
At Fault: At Fault
Latitude: 39.05835
Longitude: -77.08625

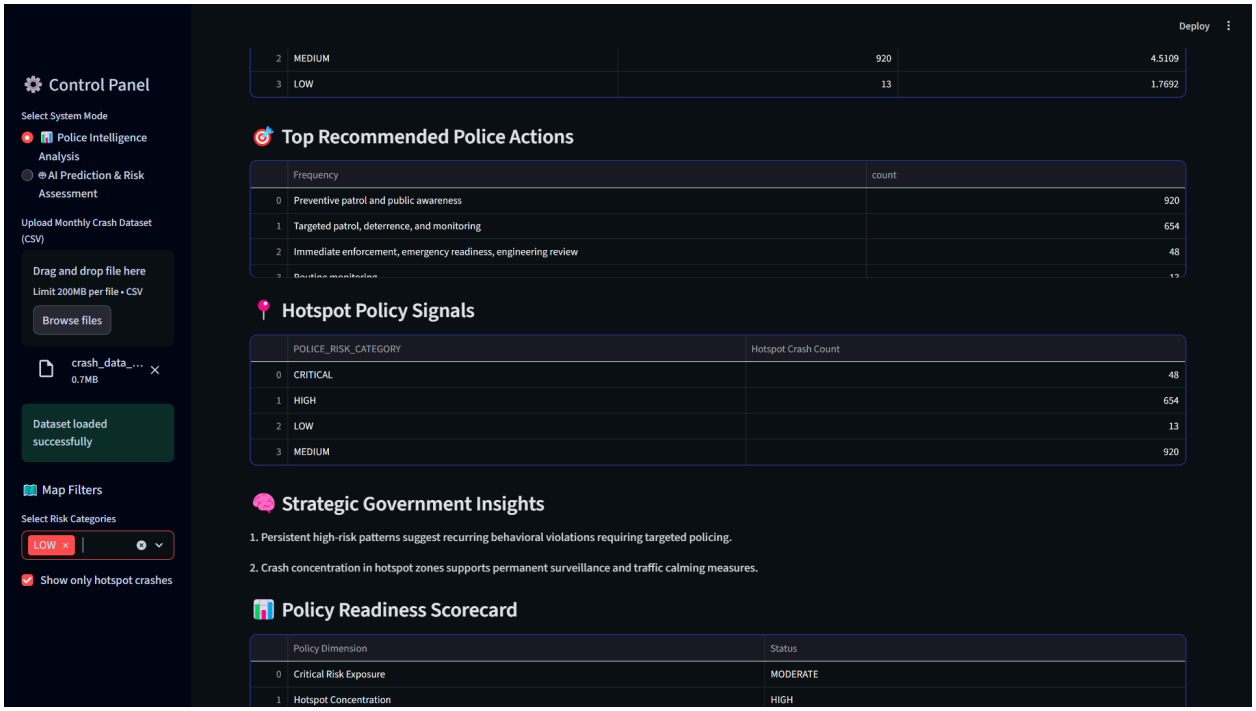

Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
CRITICAL
HIGH
LOW
MEDIUM
LOW → Normal background risk. Routine monitoring sufficient.
Explainability and reasoning generated successfully
Spatial Intelligence & Hotspot Analysis
Road: HUNTMASTER RD
Cross Street: FARMLAND DR
Risk Level: HIGH
Risk Score: 7.0
Injury: Serious Injury
Damage: No / Minor Damage
At Fault: At Fault
Latitude: 39.14488
Longitude: -77.20275

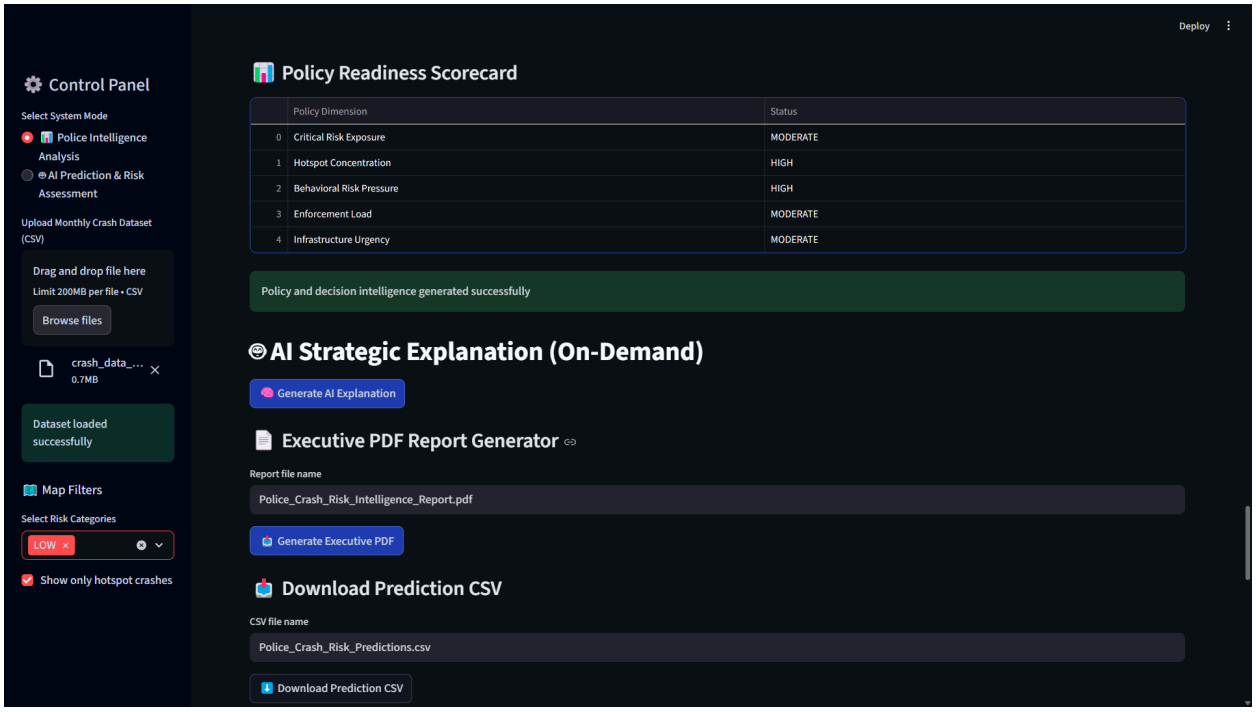

Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
MEDIUM
CRITICAL
HIGH
LOW
LOW → Normal background risk. Routine monitoring sufficient.
Explainability and reasoning generated successfully
Spatial Intelligence & Hotspot Analysis
Road: NEW HAMPSHIRE AVE
Cross Street: NEW HAMPSHIRE AVE
Risk Level: MEDIUM
Risk Score: 3.0
Injury: No Injury
Damage: Severe / Disabling Damage
At Fault: Not At Fault
Latitude: 39.06271
Longitude: -77.12976
COLUMBIA
LAUREL
BOWIE
COLLEGE PARK
RESTON
HYATTSVILLE

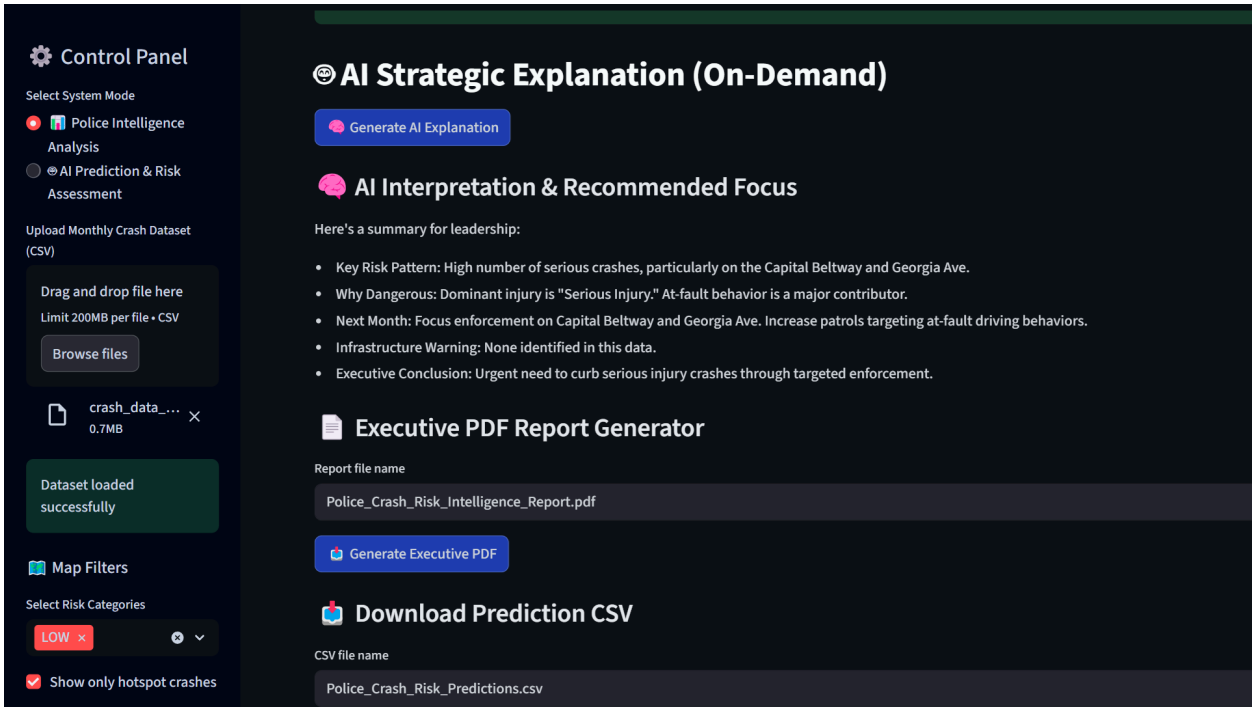

Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
LOW
CRITICAL
HIGH
MEDIUM
LOW → Normal background risk. Routine monitoring sufficient.
Explainability and reasoning generated successfully
Spatial Intelligence & Hotspot Analysis
Road: EDMONSTON DR
Cross Street: WAYNE AVE
Risk Level: LOW
Risk Score: 2.0
Injury: No Injury
Damage: No / Minor Damage
At Fault: At Fault
Latitude: 39.07770
Longitude: -77.13083
COLUMBIA
LAUREL
SILVER SPRING
BETHESDA
COLLEGE PARK

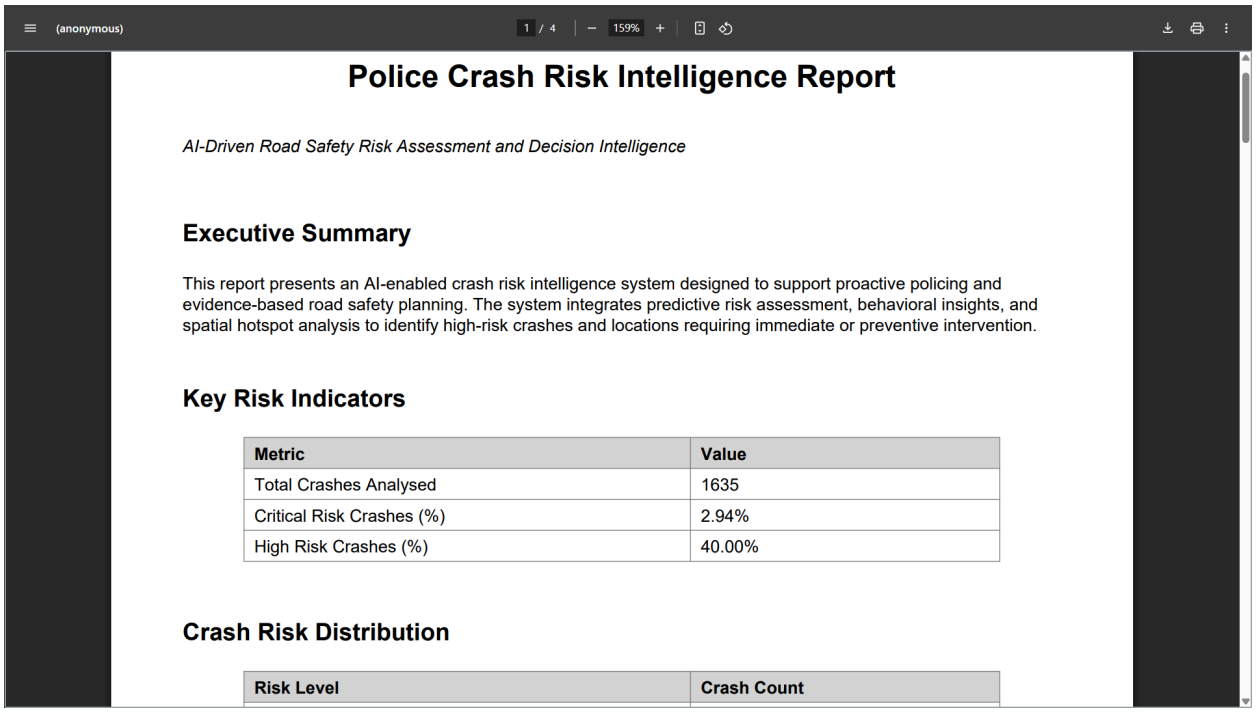

Deploy
Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
LOW
Show only hotspot crashes
Estimated High-Risk Locations for Next Month
Road Name | Cross-Street Name | incidents | avg_risk_score | fatal_cases
0 GRACE CHURCH RD | SECOND AVE | 4 | 7.75 | 0
1 AIRCRAFT DR | GERMANTOWN RD (SB/L) | 4 | 7 | 0
2 CALVERTON BLVD | BROADBIRCH DR CHERRY HILL RD | 4 | 7 | 0
3 MONTROSE RD (WB/L) | TILDENWOOD DR | 4 | 6.75 | 0
4 SENECA MEADOWS PKWY | GERMANTOWN RD (SB/L) | 3 | 8.3333 | 0
5 TRAVILAH RD | RIVER RD RIVERS EDGE DR | 3 | 7.6667 | 0
6 MUNCASTER MILL RD | NORBECK RD (WB/L) | 3 | 7 | 0
7 REDLAND BLVD | RAMP 9 FR RAMP 1 TO REDLAND BLVD | 3 | 7 | 0
8 STRINGTOWN RD | GATEWAY CENTER DR (NB/L) | 3 | 7 | 0
9 WINTERGATE DR | NO NAME NORBECK RD | 3 | 6.6667 | 0
These locations show recurring high-risk patterns and should be prioritized for preventive action in the upcoming month.
Hotspot Intelligence Summary
No recurring high-risk hotspots detected.
Policy & Decision Intelligence
Risk Distribution Overview
Crash Risk Distribution

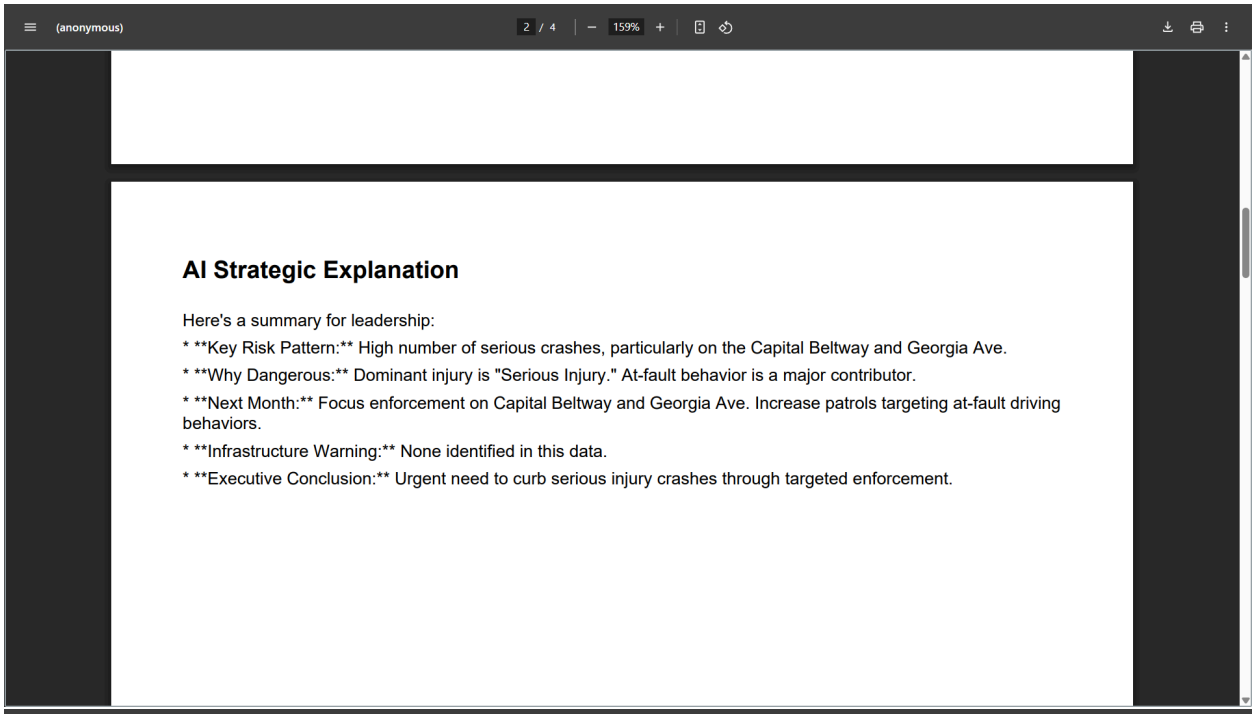

Deploy
Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
LOW
Show only hotspot crashes
Policy & Decision Intelligence
Risk Distribution Overview
Crash Risk Distribution
Crash Count
800
600
400
200
0
MEDIUM
HIGH
CRITICAL
LOW
Risk Level
Risk Level
MEDIUM
HIGH
CRITICAL
LOW
Enforcement Priority Matrix
POLICE_RISK_CATEGORY | total_crashes | avg_risk_score
0 CRITICAL | 48 | 10.5833
1 HIGH | 654 | 7.1988
2 MEDIUM | 920 | 4.5109
3 LOW | 13 | 1.7692
Top Recommended Police Actions


Deploy
Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
LOW
Show only hotspot crashes
2 MEDIUM 920 4.5109
3 LOW 13 1.7692
Top Recommended Police Actions
Frequency count
0 Preventive patrol and public awareness 920
1 Targeted patrol, deterrence, and monitoring 654
2 Immediate enforcement, emergency readiness, engineering review 48
Hotspot Policy Signals
POLICE_RISK_CATEGORY Hotspot Crash Count
0 CRITICAL 48
1 HIGH 654
2 LOW 13
3 MEDIUM 920
Strategic Government Insights
1. Persistent high-risk patterns suggest recurring behavioral violations requiring targeted policing.
2. Crash concentration in hotspot zones supports permanent surveillance and traffic calming measures.
Policy Readiness Scorecard
Policy Dimension Status
0 Critical Risk Exposure MODERATE
1 Hotspot Concentration HIGH


Deploy
Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
LOW
Show only hotspot crashes
Policy Readiness Scorecard
Policy Dimension Status
0 Critical Risk Exposure MODERATE
1 Hotspot Concentration HIGH
2 Behavioral Risk Pressure HIGH
3 Enforcement Load MODERATE
4 Infrastructure Urgency MODERATE
Policy and decision intelligence generated successfully
AI Strategic Explanation (On-Demand)
Generate AI Explanation
Executive PDF Report Generator
Report file name
Police_Crash_Risk_Intelligence_Report.pdf
Generate Executive PDF
Download Prediction CSV
CSV file name
Police_Crash_Risk_Predictions.csv
Download Prediction CSV


Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
LOW
Show only hotspot crashes
AI Strategic Explanation (On-Demand)
Generate AI Explanation
AI Interpretation & Recommended Focus
Here's a summary for leadership:
Key Risk Pattern: High number of serious crashes, particularly on the Capital Beltway and Georgia Ave.
Why Dangerous: Dominant injury is "Serious Injury." At-fault behavior is a major contributor.
Next Month: Focus enforcement on Capital Beltway and Georgia Ave. Increase patrols targeting at-fault driving behaviors.
Infrastructure Warning: None identified in this data.
Executive Conclusion: Urgent need to curb serious injury crashes through targeted enforcement.
Executive PDF Report Generator
Report file name
Police_Crash_Risk_Intelligence_Report.pdf
Generate Executive PDF
Download Prediction CSV
CSV file name
Police_Crash_Risk_Predictions.csv

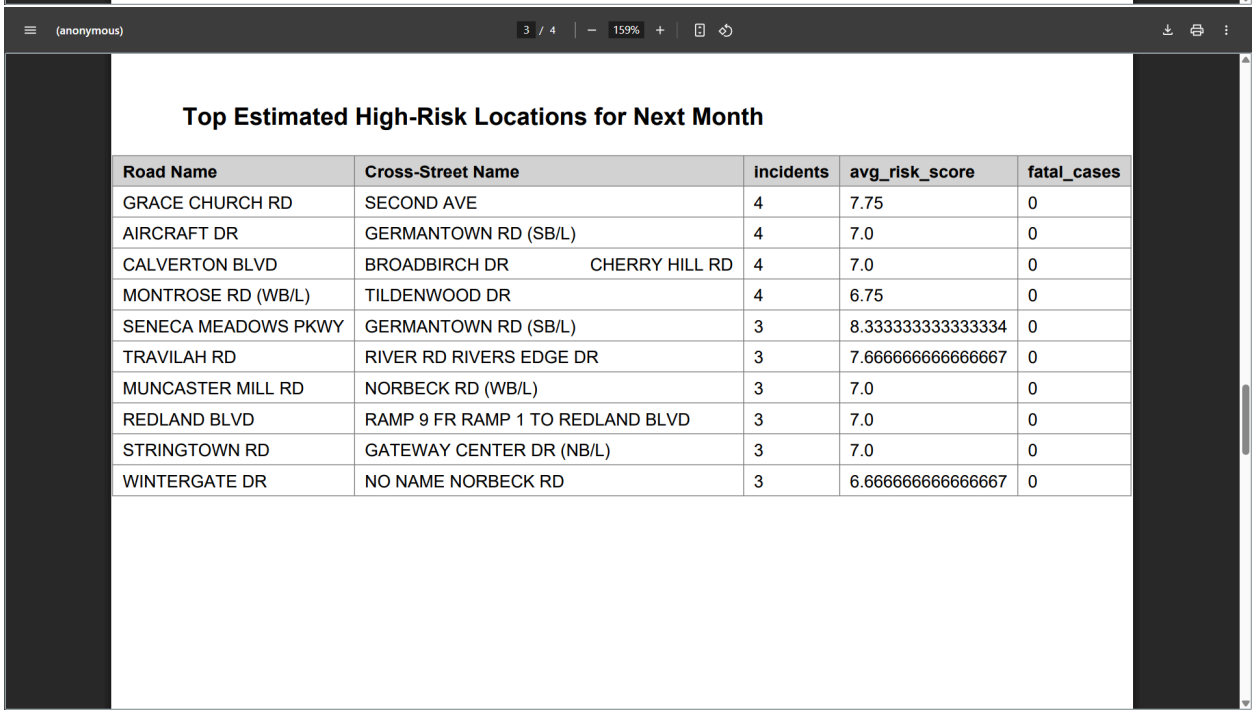

(anonymous)
1 / 4
159%
Police Crash Risk Intelligence Report
AI-Driven Road Safety Risk Assessment and Decision Intelligence
Executive Summary
This report presents an AI-enabled crash risk intelligence system designed to support proactive policing and evidence-based road safety planning. The system integrates predictive risk assessment, behavioral insights, and spatial hotspot analysis to identify high-risk crashes and locations requiring immediate or preventive intervention.
Key Risk Indicators
Metric
Value
Total Crashes Analysed
1635
Critical Risk Crashes (%)
2.94%
High Risk Crashes (%)
40.00%
Crash Risk Distribution
Risk Level
Crash Count



AI Strategic Explanation

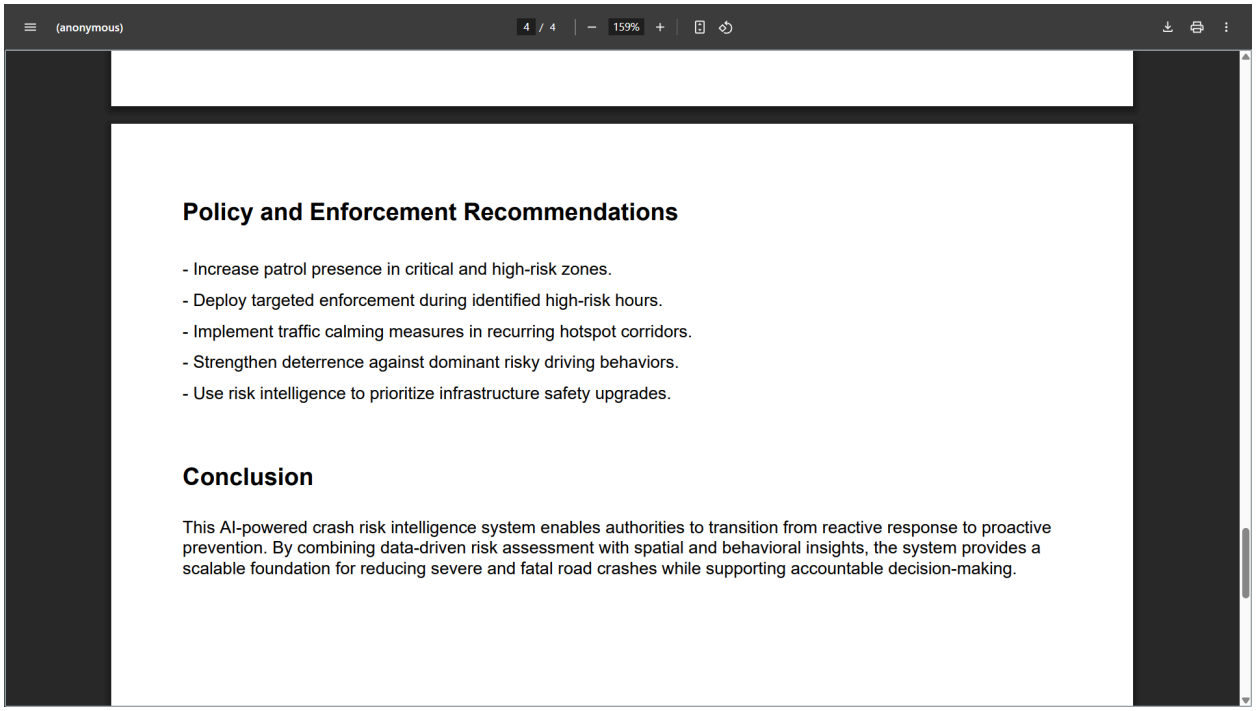
Here's a summary for leadership:

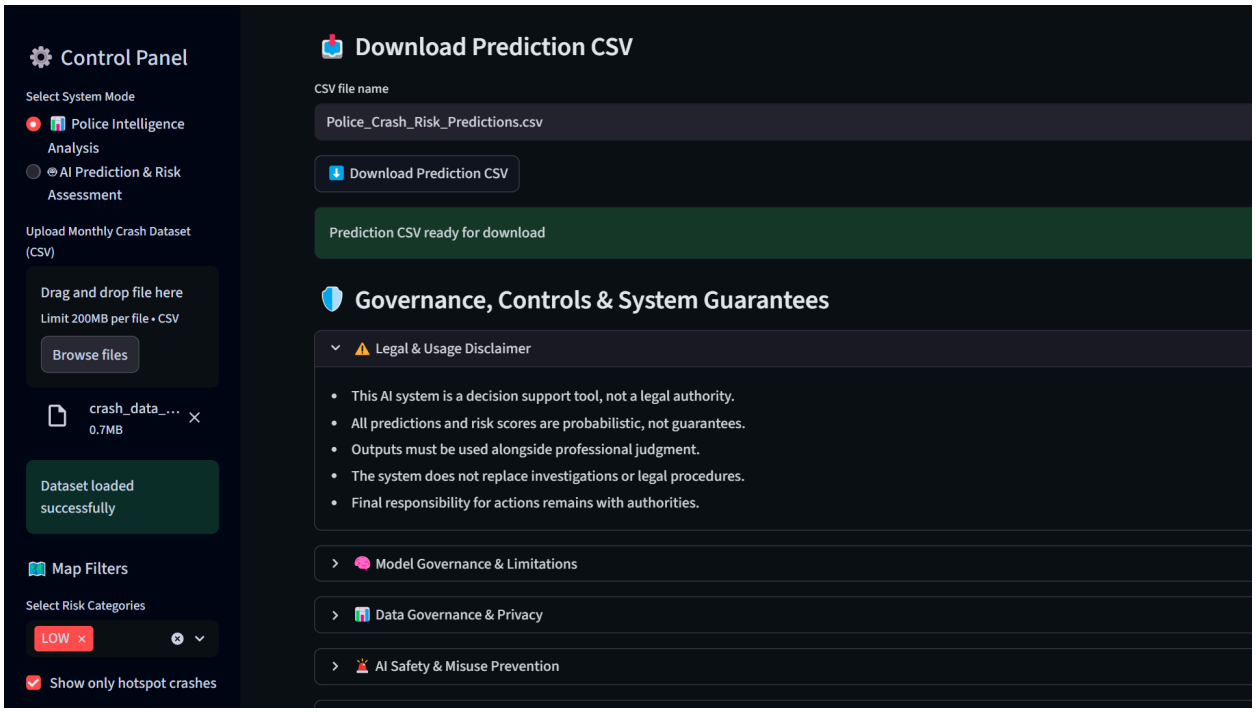
* **Key Risk Pattern:** High number of serious crashes, particularly on the Capital Beltway and Georgia Ave.
* **Why Dangerous:** Dominant injury is "Serious Injury." At-fault behavior is a major contributor.
* **Next Month:** Focus enforcement on Capital Beltway and Georgia Ave. Increase patrols targeting at-fault driving behaviors.
* **Infrastructure Warning:** None identified in this data.
* **Executive Conclusion:** Urgent need to curb serious injury crashes through targeted enforcement.

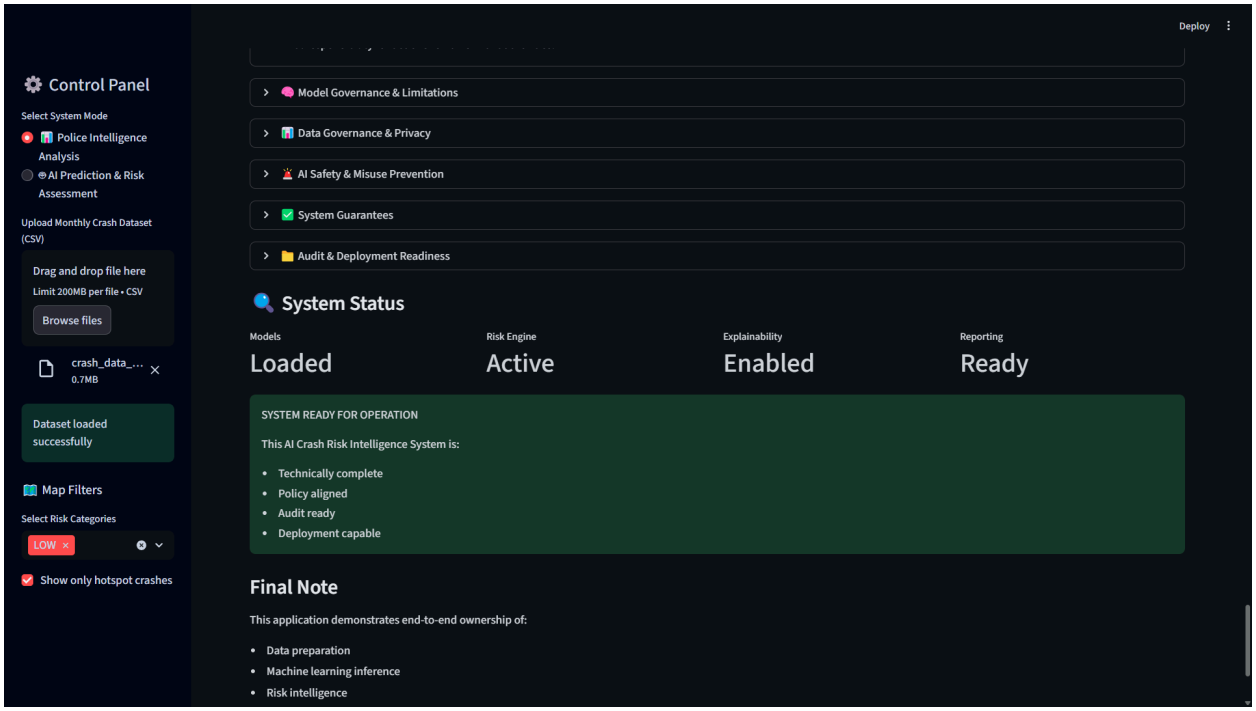


Top Estimated High-Risk Locations for Next Month

Road Name	Cross-Street Name	incidents	avg_risk_score	fatal_cases
GRACE CHURCH RD	SECOND AVE	4	7.75	0
AIRCRAFT DR	GERMANTOWN RD (SB/L)	4	7.0	0
CALVERTON BLVD	BROADBIRCH DR CHERRY HILL RD	4	7.0	0
MONTROSE RD (WB/L)	TILDENWOOD DR	4	6.75	0
SENECA MEADOWS PKWY	GERMANTOWN RD (SB/L)	3	8.333333333333334	0
TRAVILAH RD	RIVER RD RIVERS EDGE DR	3	7.666666666666667	0
MUNCASTER MILL RD	NORBECK RD (WB/L)	3	7.0	0
REDLAND BLVD	RAMP 9 FR RAMP 1 TO REDLAND BLVD	3	7.0	0
STRINGTOWN RD	GATEWAY CENTER DR (NB/L)	3	7.0	0
WINTERGATE DR	NO NAME NORBECK RD	3	6.666666666666667	0


(anonymous)
4 / 4
159%
Policy and Enforcement Recommendations
- Increase patrol presence in critical and high-risk zones.
- Deploy targeted enforcement during identified high-risk hours.
- Implement traffic calming measures in recurring hotspot corridors.
- Strengthen deterrence against dominant risky driving behaviors.
- Use risk intelligence to prioritize infrastructure safety upgrades.
Conclusion
This AI-powered crash risk intelligence system enables authorities to transition from reactive response to proactive prevention. By combining data-driven risk assessment with spatial and behavioral insights, the system provides a scalable foundation for reducing severe and fatal road crashes while supporting accountable decision-making.

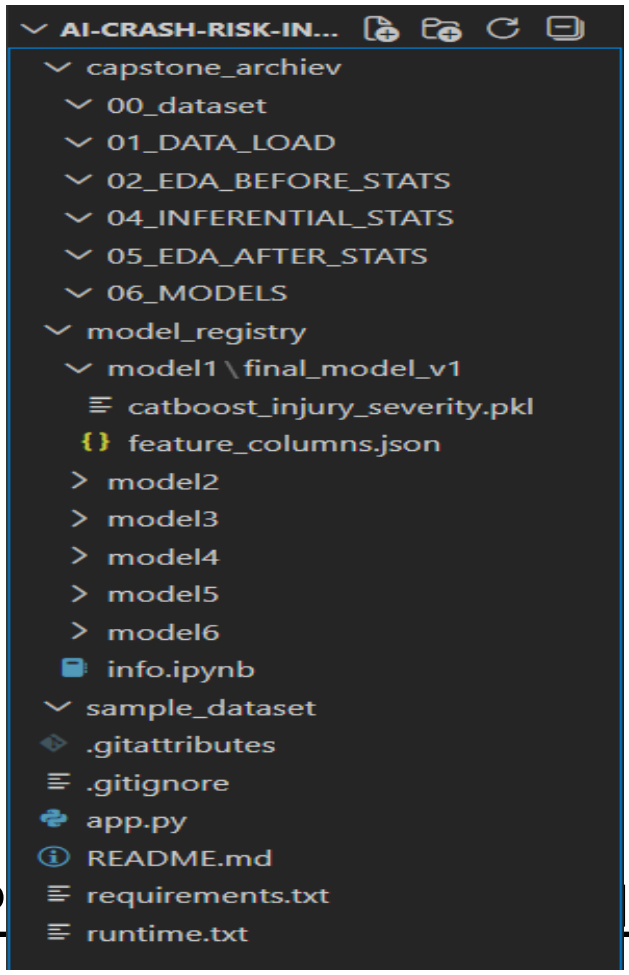

Control Panel
Select System Mode
Police Intelligence Analysis
AI Prediction & Risk Assessment
Upload Monthly Crash Dataset (CSV)
Drag and drop file here
Limit 200MB per file • CSV
Browse files
crash_data_...
0.7MB
Dataset loaded successfully
Map Filters
Select Risk Categories
LOW
Show only hotspot crashes
Download Prediction CSV
CSV file name
Police_Crash_Risk_Predictions.csv
Download Prediction CSV
Prediction CSV ready for download
Governance, Controls & System Guarantees
Legal & Usage Disclaimer
This AI system is a decision support tool, not a legal authority.
All predictions and risk scores are probabilistic, not guarantees.
Outputs must be used alongside professional judgment.
The system does not replace investigations or legal procedures.
Final responsibility for actions remains with authorities.
Model Governance & Limitations
Data Governance & Privacy
AI Safety & Misuse Prevention



End-to-End System Architecture

The system follows a **modular and layered architecture** to ensure clarity, stability, and scalability.

Layer	Responsibility
Data Layer	Cleaned & validated crash data
Feature Layer	Encoding, transformation, schema alignment
Model Layer	Six independent ML models
Intelligence Layer	Risk scoring & categorization
Explainability Layer	Human-readable reasoning
Visualization Layer	Maps, tables, Tableau
Reporting Layer	CSV & PDF outputs



Business P , and Value Delivered

Business Problem / Decision Need	Model(s) Used & Business Value Delivered
Manual identification of crash-prone locations is slow, subjective, and analyst-dependent	Hotspot Clustering Model automatically identifies high-risk locations, removing manual geographic investigation and significantly reducing analyst effort and bias
Emergency responders need to quickly prioritize life-threatening crashes	Injury Severity Prediction Model flags high-severity and fatal crashes early, enabling faster response planning and improved casualty outcomes

Authorities lack visibility into unsafe driver behavior patterns	Driver At-Fault Model and Driver Distraction Model uncover behavioral risk patterns such as fault, distraction, and negligence, enabling targeted enforcement and awareness campaigns
Infrastructure planning decisions are made without data-driven risk insights	Environmental & Road Factor Signals embedded across models (lighting, traffic control, speed limits, surface condition) highlight hazardous conditions to guide safety and infrastructure improvements
Agencies struggle to justify safety budgets and intervention costs	Crash Risk Level Prediction Model quantifies overall crash risk, supporting evidence-based budgeting, prioritization, and policy justification
Insurance and fleet teams face delays due to manual damage assessment	Vehicle Damage Extent Classification Model enables faster damage severity estimation, improving claims processing efficiency and operational planning

Benefits and Limitations of the System

Aspect	Benefit / Strength	Limitation / Constraint
Decision Making	Transforms subjective judgment into data-driven risk prioritization	Requires human interpretation; not an autonomous decision-maker
Risk Identification	Early identification of high-risk crashes and locations using multi-model intelligence	Relies on historical data, sudden pattern shifts may lag
Operational Efficiency	Eliminates manual hotspot analysis and repetitive reporting effort	High initial data preparation and modeling effort
Explainability	Rule-based, transparent explanations for every high-risk case	Explanations indicate risk correlation, not causality
Scalability	Handles large-scale, multi-month datasets consistently	Output quality is data-quality dependent
Enforcement Planning	Enables targeted patrols and preventive interventions	Does not automate enforcement actions

Infrastructure Insights	Identifies risky road, lighting, and traffic-control conditions	Does not generate engineering design solutions
Multi-Model Architecture	Integrated 6-model system covering severity, risk, behavior, and damage	System complexity increases maintenance overhead
Deployment Readiness	Fully deployable Streamlit application with CSV & PDF outputs	Requires controlled Python environment & dependency alignment
Ethical & Governance Safety	Strictly decision-support, audit-ready, and human-in-the-loop	Cannot replace expert or legal authority

CONCLUSION

The **Road Safety & Crash Intelligence System (RSCIS)** demonstrates how an end-to-end data science pipeline can be applied to a real-world public safety problem with measurable operational value. Moving beyond traditional reactive crash reporting, the system integrates **multi-model prediction, behavioral analysis, environmental risk factors, vehicle impact assessment, and geospatial hotspot detection** into a single, coherent analytical framework.

The combined model outputs consistently show that crash risk is **structural and behavioral rather than random**, driven primarily by **speed-intensive corridors, intersection density, traffic exposure, road conditions, and driver behavior**. By translating these patterns into **risk scores, explainable indicators, and location-specific insights**, the system enables faster prioritization of high-severity cases and more focused preventive interventions.

From an implementation perspective, RSCIS validates the effectiveness of **robust data preparation, probabilistic and logical imputation, disciplined feature engineering, and explainable modeling** in handling complex, noisy, real-world datasets. The Streamlit deployment and Tableau-ready outputs further demonstrate how analytical results can be converted into **usable decision intelligence** rather than remaining confined to notebooks.

Overall, this project establishes a **scalable and extensible foundation** for intelligent road-safety analytics. With periodic retraining and expanded data coverage, the system can continue to support **evidence-based enforcement planning, infrastructure prioritization, and risk monitoring**, reinforcing the role of applied data science in improving road safety outcomes and operational efficiency.

LINKS

Live_app:- https://crash-risk-system.streamlit.app/
Github_Repo:- https://github.com/anuragkumarsingh4440-netizen/AI-Crash-Risk-Intelligence-System
Video_Link:- https://drive.google.com/file/d/1JmgGzK_LUaDVRSiaaTTRHBRCTcWX7qfP/view?usp=sharing
Tableau_Dashboard_Link:- https://surl.li/jycscm

REFERENCES:

1. Maryland State Police – ACRS Crash Reporting System

https://opendata.maryland.gov/ Used because it provides the real crash dataset structure; helps build accurate ML models for severity, behavior, and hotspots.

2. NHTSA – Crash Data Systems

https://www.nhtsa.gov/crash-data-systems Used because it defines national crash standards; helps validate severity classes and modeling structure.

3. FARS – Fatality Analysis Reporting System

https://www.nhtsa.gov/research-data/fatality-analysis-reporting-system-fars Used because it gives fatal-crash patterns; helps refine high-severity risk features.

4. HSIS – Highway Safety Information System

https://www.hsisinfo.org/ Used because it provides roadway and environmental crash factors; helps strengthen road-condition and weather modeling.

5. Maryland Open Data Portal – Crash Records

https://opendata.maryland.gov/Transportation/Crash-Data/65du-s3qu Used because it contains state-level crash trends; helps validate spatial and temporal patterns.

6. FHWA – Highway Safety Manual (HSM)

https://highwaysafetymanual.org/ Used because it defines scientific crash prediction methods; helps align hotspot modeling with safety-industry standards.

7. AAA Foundation – Driver Behavior Research

https://aaafoundation.org/ Used because it explains distraction and risky-driving patterns; helps support driver-behavior model logic.

8. Accident Analysis & Prevention (Open Abstracts Only)

https://www.sciencedirect.com/journal/accident-analysis-and-prevention Used because it provides research-backed crash modeling methods; helps justify ML choices like clustering and severity prediction.

9. USDOT – Traffic Volume & Speed Data

https://www.bts.gov/ Used because it links speed and vehicle flow to crash likelihood; helps interpret high-speed corridor risks.

10. ESRI GIS – Public Safety Spatial Analysis

https://www.esri.com/en-us/arcgis/products/arcgis-solutions/public-safety Used because it guides geospatial risk mapping; helps validate hotspot and cluster-analysis approaches.

SUMMARY

These references support U.S. crash data standards, geospatial hotspot modeling,driver-behavior analysis, environmental risk assessment, and severity prediction,making the project technically credible and industry-aligned.