Car Accident Severity
Prediction

Vedant Mane

IBM Capstone Project

The Problem:

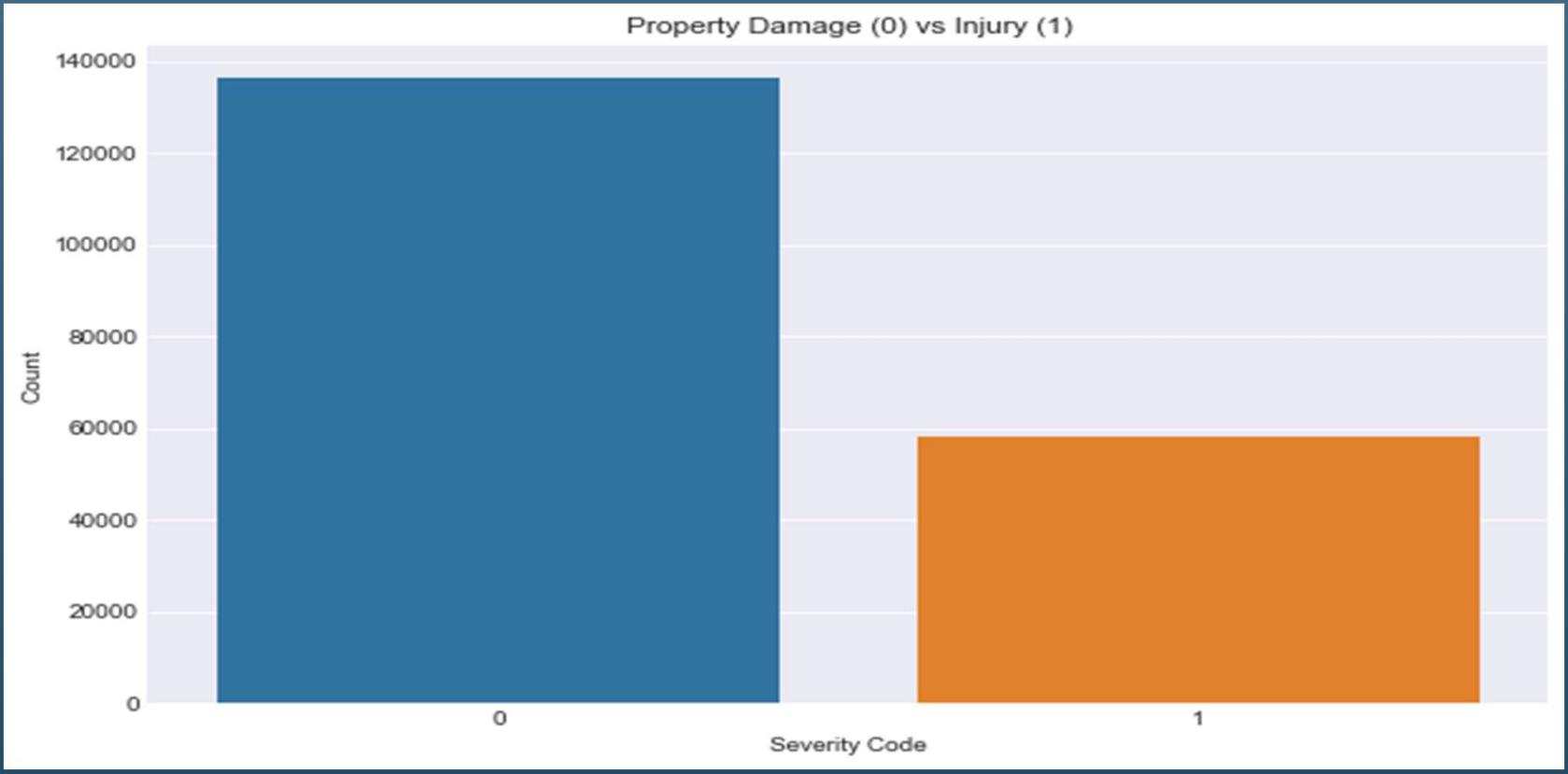
- Car Accidents → Major issue around the globe
- Personal vehicles: Increasing day by day
- Seattle, Washington: 2x cars since 2010
- Increase in Car Accidents around the city

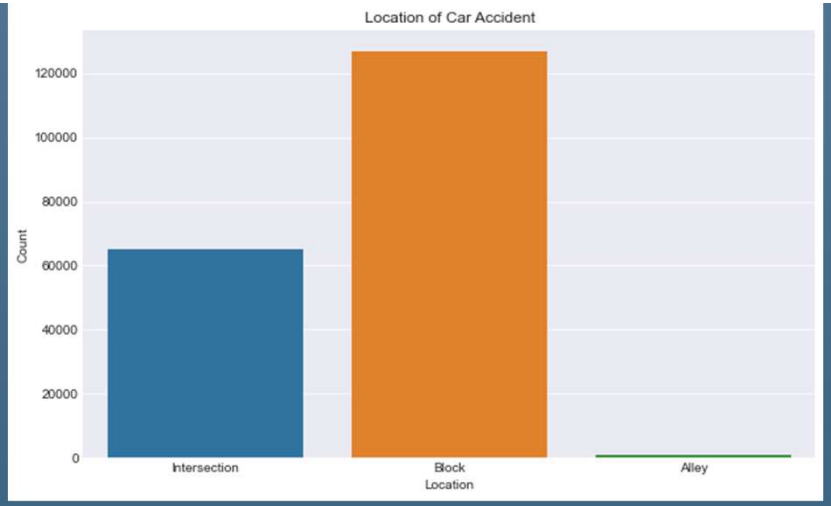
Data Source & Pre-processing:

- Data Source: Given by IBM for the project.
- Metadata: Link
- Dataset: 37 Features, 194673 Obervations.
- Pre-Processing:
  - SEVERITYCODE: 0 → Property Damage, 1 → Injury
  - Integer Encoding: INATTENTIONIND, UNDERINFL, SPEEDING , JUNCTIONTYPE
  - One-hot Encoding: LIGHTCOND, ROADCOND, WEATHER
  - Normalize

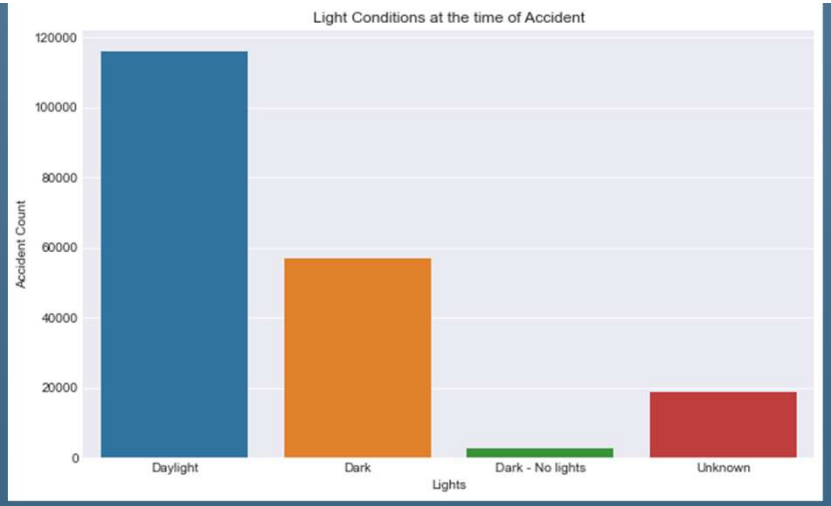
Getting Insights on Data:

Highly Skewed Target Variable

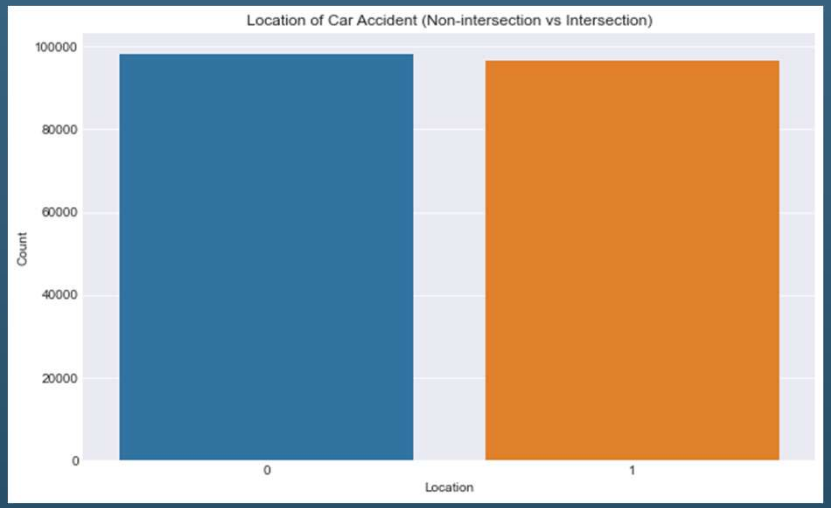




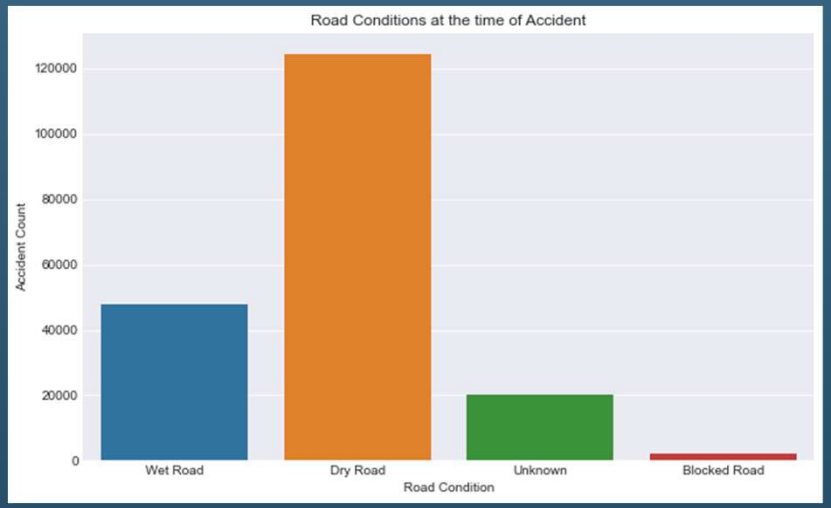
“ADDRTYPE”



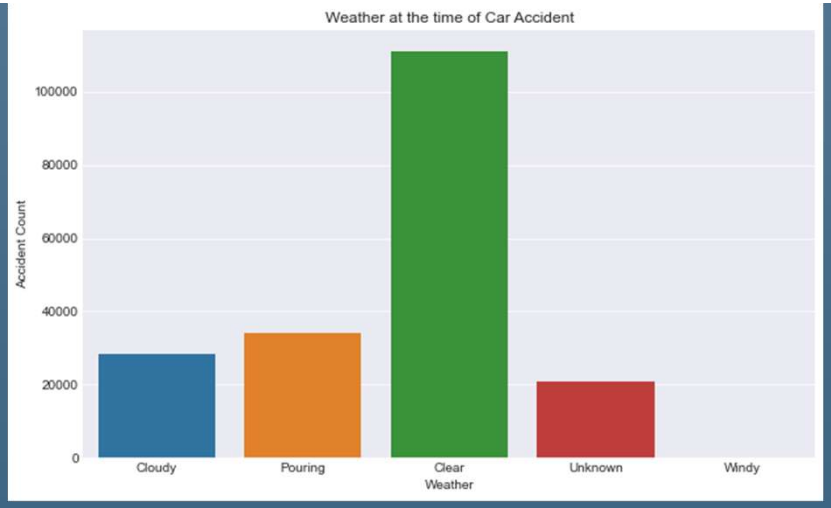
LIGHTCOND



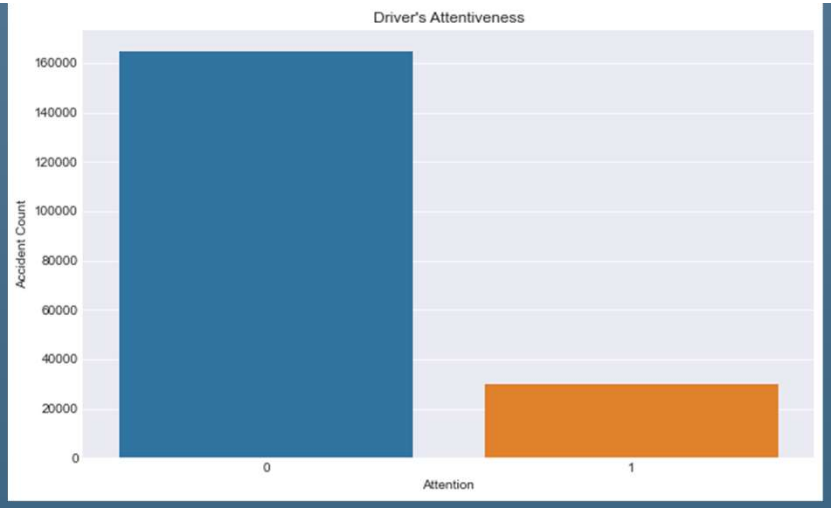
“JUNCTIONTYPE”



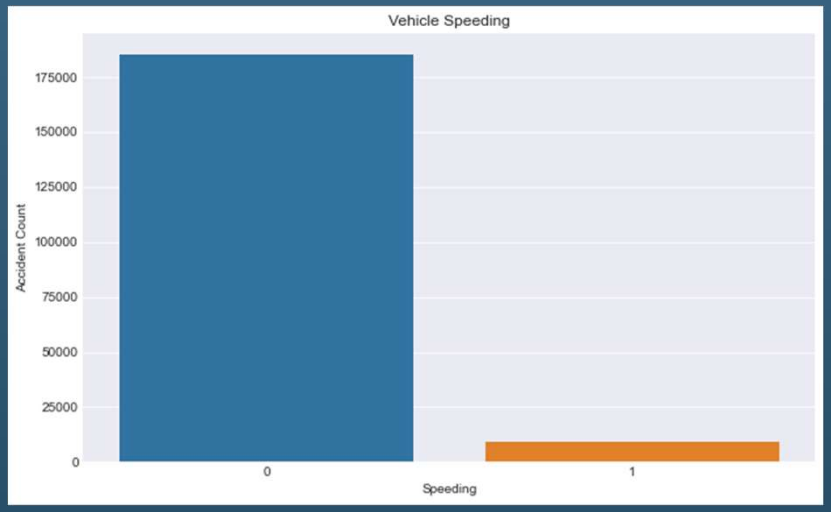
ROADCOND



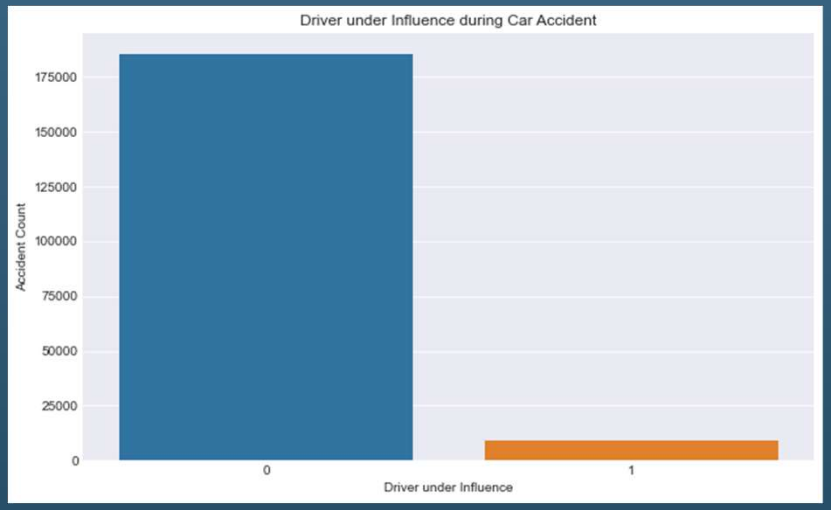
"WEATHER"



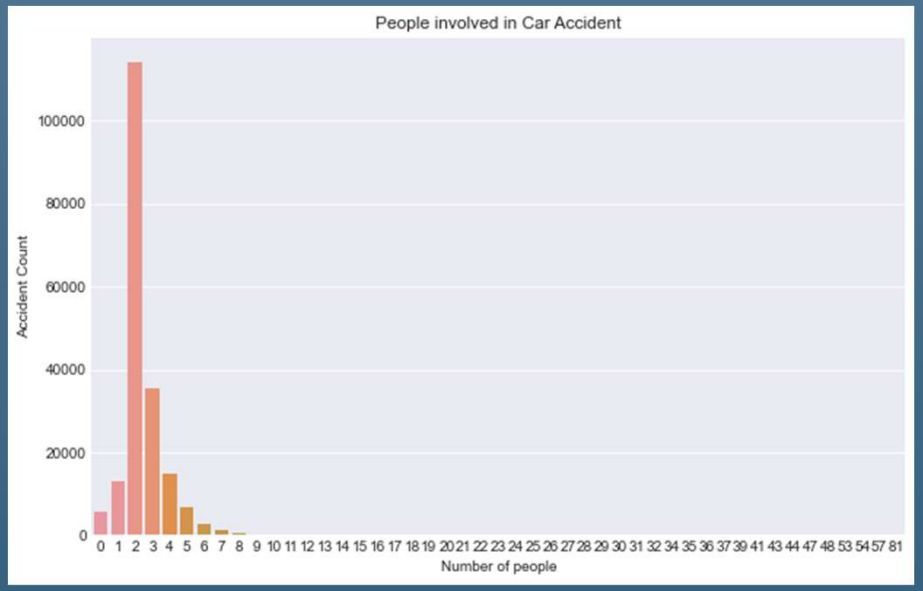
"INATTENTIONIND"



"SPEEDING"

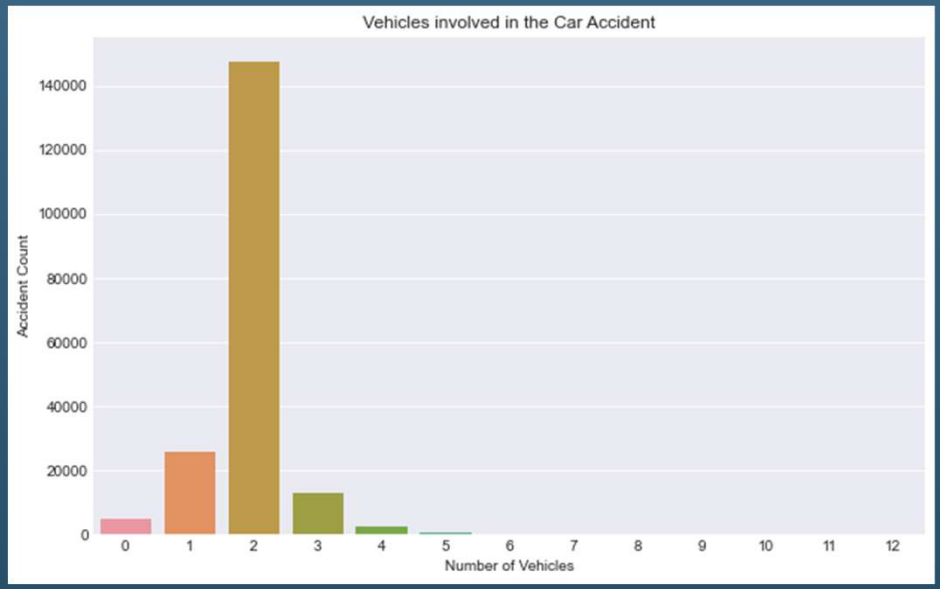


"UNDERINFL"

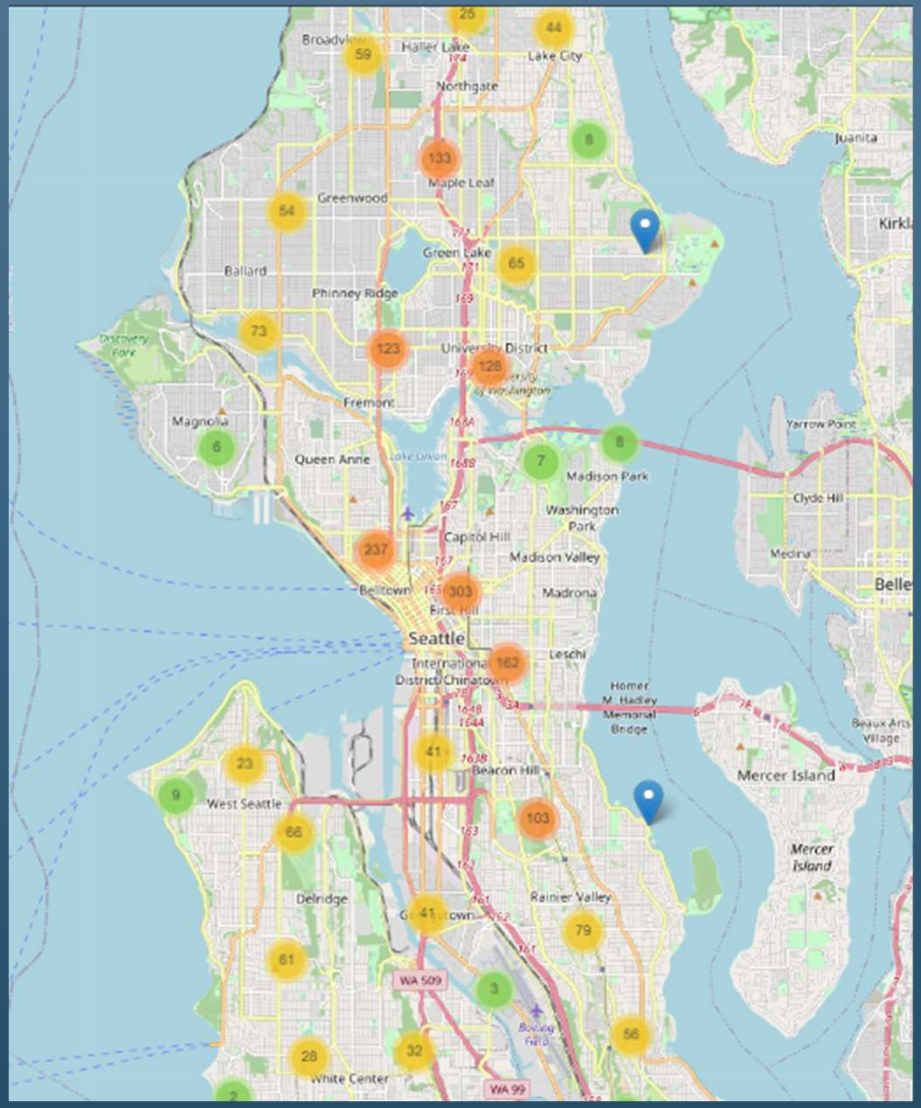


"PERSONCOUNT"

"VEHCOUNT"



Map of Seattle, Washington explaining car accidents in different locations



Model Building

- Classification Problem

- Use:
  - Logistic Regression
  - Decision Tree

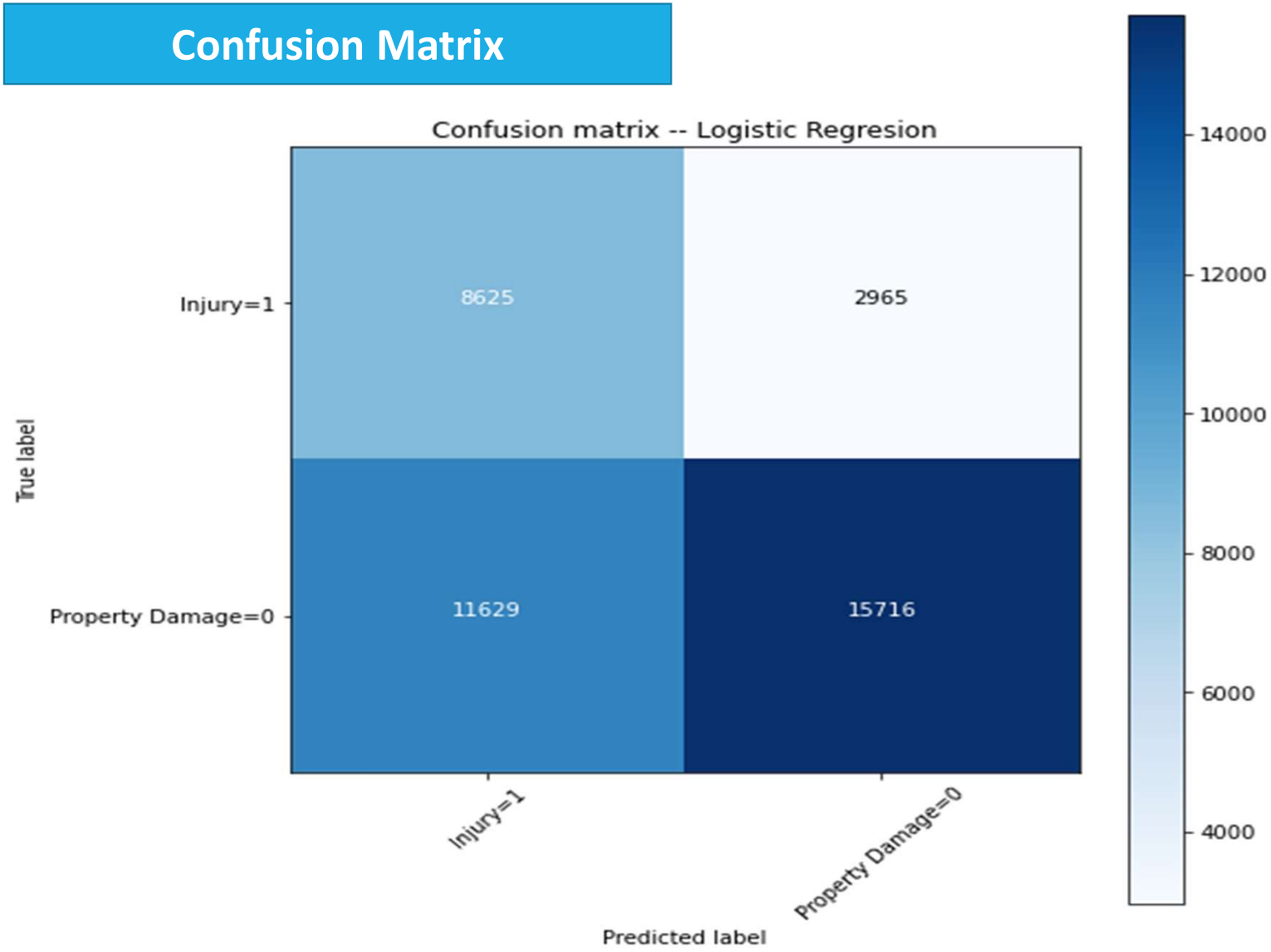
- KNN – Performs poorly for highly skewed datasets.
- SVM – Not suitable for large datasets.

Logistic Regression:

- Logistic Regression Classifier
- Scikit-Learn Library
- Parameters: C = 0.01 (Regularization Strength), Solver = “liblinear”
- Classification Report:

	Precision	Recall	F1 Score	Support
0	0.82	0.63	0.71	27345
1	0.43	0.67	0.52	11590
Accuracy			0.64	38935
Macro average	0.62	0.65	0.62	38935
Weighted average	0.70	0.64	0.65	38935
Log loss			0.63	

Confusion Matrix



Model Evaluation:

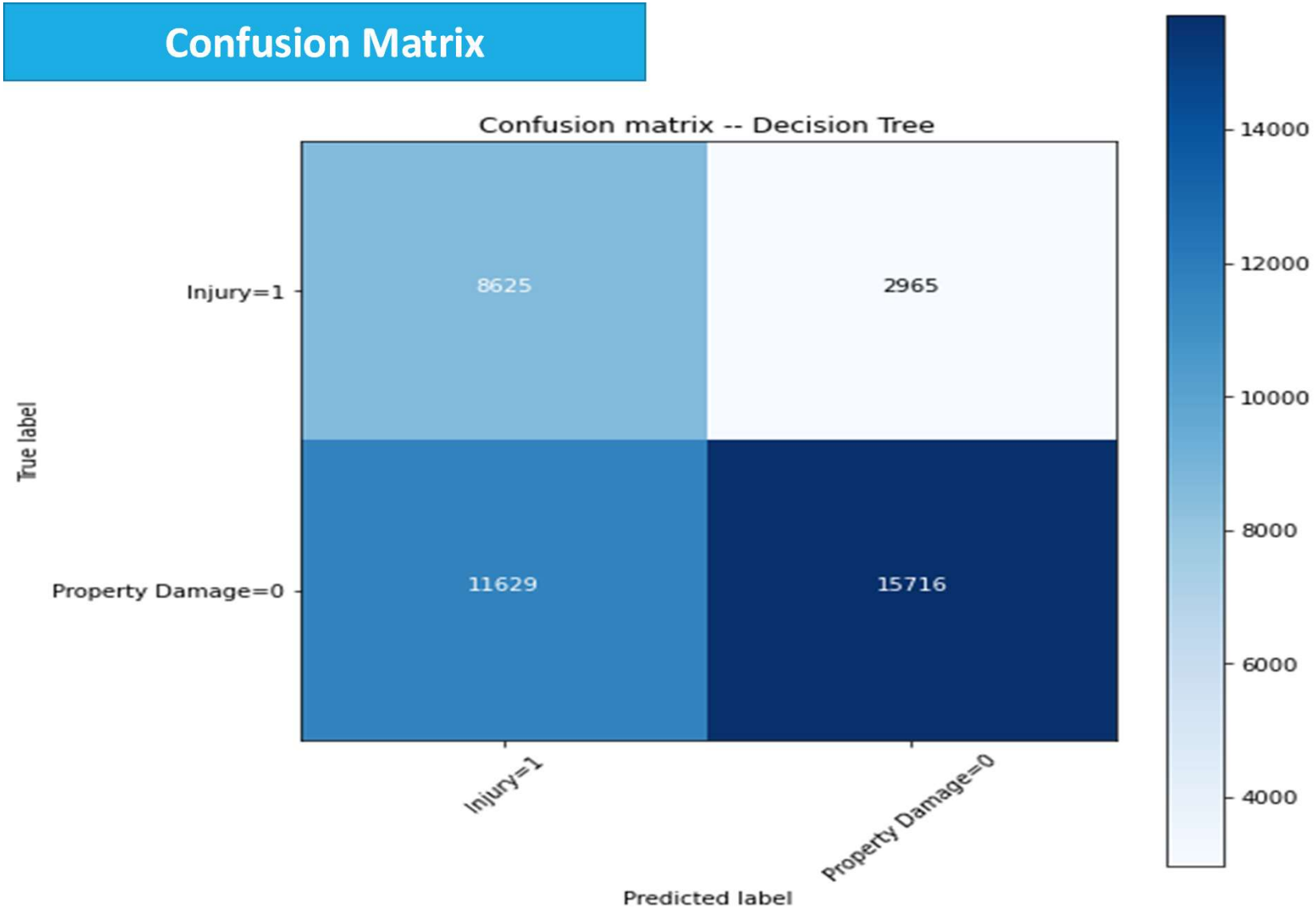
- Accuracy Score: **0.63850006420958**
- F1 – Score: **0.5240912933220624**
- Log Loss: **0.6311264004171047**

Decision Tree:

- Decision Tree Classifier
- Scikit-Learn Library
- Parameters: Criterion = “entropy”, max_depth = 4
- Classification Report:

	Precision	Recall	F1 Score	Support
0	0.84	0.57	0.68	27345
1	0.43	0.74	0.54	11590
Accuracy			0.63	38935
Macro average	0.63	0.66	0.61	38935
Weighted average	0.72	0.63	0.64	38935

Confusion Matrix



Model Evaluation:

- Accuracy Score: **0.6251701553871838**
- F1 – Score: **0.5417033036050747**

Discussion:

Algorithm	Average F1 Score	Property Damage (0) vs Injury (1)	Precision	Recall
Logistic Regression	0.65	0	0.82	0.63
		1	0.43	0.67
Decision Tree	0.64	0	0.84	0.57
		1	0.43	0.74

- Both models → Similar Accuracy
- Logistic Regression: Balanced Recall
- Decision Tree: Better Precision

Recommendations:

- Launch development projects for those areas with higher concentration of accidents and most severe accidents in order to minimize the effects of these two factors.
- Install safety signs on roads and ensure all necessary precautionary measures are taken by the people living in the area.
- Be extra careful around the highways along the city's centre as it has the highest number of accidents reported.

Conclusion:

- Comparing both models
  - Similar accuracy scores for target variables
  - Decision Tree → Higher Precision
  - Logistic Regression → Balanced Recall
  - Suggestion: Use both models based on requirements.
- The models could have performed better if a few more things were available:
  - A balanced dataset for the target variable
  - More instances of the accidents recorded that took place in Seattle, Washington.
  - Less missing values for features such as Speeding and Under Influence.
  - More factors, such as precautionary measures taken while driving, etc.

Thank you!!!