applied sciences

MDPI

Article

An Interpretable Machine Learning-Based Hurdle Model for Zero-Inflated Road Crash Frequency Data Analysis: Real-World Assessment and Validation

Moataz Bellah Ben Khedher [1,2] and Dukgeun Yun [1,2,*]

[1] Department of Civil and Environmental Engineering, KICT School, University of Science and Technology, Daejeon 34113, Republic of Korea; moataz.khedher@gmail.com
[2] Department of Highway and Transportation Research, Korea Institute of Civil Engineering and Building Technology, Goyang-Si 10223, Republic of Korea
* Correspondence: dkyun@kict.re.kr

Abstract: Road traffic crashes pose significant economic and public health burdens, necessitating an in-depth understanding of crash causation and its links to underlying factors. This study introduces a machine learning-based hurdle model framework tailored for analyzing zero-inflated crash frequency data, addressing the limitations of traditional statistical models like the Poisson and negative binomial models, which struggle with zero-inflation and overdispersion. The research employs a two-stage modeling process using CatBoost. The first stage uses binary classification to identify road segments with potential crash occurrences, applying a customized loss function to tackle data imbalance. The second stage predicts crash frequency, also utilizing a customized loss function for count data. SHapley Additive exPlanations (SHAP) analysis interprets the model outcomes, providing insights into factors affecting crash likelihood and frequency. This study validates the model's performance with real-world crash data from 2011 to 2015 in South Korea, demonstrating superior accuracy in both the classification and regression stages compared to other machine learning algorithms and traditional models. These findings have significant implications for traffic safety research and policymaking, offering stakeholders a more accurate and interpretable tool for crash data analysis to develop targeted safety interventions.

Keywords: crash frequency; machine learning; CatBoost; SHAP; accident analysis; road safety

Citation: Ben Khedher, M.B.; Yun, D. An Interpretable Machine Learning-Based Hurdle Model for Zero-Inflated Road Crash Frequency Data Analysis: Real-World Assessment and Validation. *Appl. Sci.* **2024**, *14*, 10790. https://doi.org/10.3390/app142310790

Academic Editor: Suchao Xie

Received: 20 August 2024
Revised: 12 November 2024
Accepted: 16 November 2024
Published: 21 November 2024

1. Introduction

Road traffic accidents remain a pressing global issue, claiming approximately 1.19 million lives annually, according to the World Health Organization (WHO) [1]. In the United States, for example, the year 2020 saw around 40,000 fatalities due to traffic collisions, with an additional 2.1 million people requiring emergency medical care as a result of such incidents [2]. Beyond the tragic human toll, the economic burden is immense, with annual costs in the U.S. alone reaching an estimated USD 430 billion—a figure that accounts for healthcare expenses, productivity losses, and the diminished quality of life [3]. On a global scale, the WHO reports that road crashes cost countries, on average, roughly 3% of their gross domestic product (GDP) [1].

The field of traffic safety research has evolved significantly over the decades, driven by the dual objectives of understanding the complex nature of road traffic crashes and developing effective interventions to reduce their occurrence and severity. At the heart of this research lies crash prediction modeling, a methodology used for identifying the underlying factors contributing to road crashes' occurrence and the resulting traffic injuries and fatalities [4]. These factors include, but are not limited to, traffic volume, road geometry, environmental conditions, vehicle characteristics, and driver behavior.

Crash prediction modeling is a statistical approach that is used to analyze how various predictors affect crash severity and frequency. The primary focus is on identifying relationships that help predict the occurrence of crashes and the likelihood of different injury types. Crash severity modeling aims at identifying influential factors in the severity of injuries, where outcomes are confined to categories such as those in the KABCO scale. Crash frequency modeling, on the other hand, quantitatively assesses the relationship between crash counts and their influencing factors. Historically, crash frequency analysis has relied heavily on count models, with the Poisson and negative binomial models being particularly prominent due to their suitability for handling count data [4,5]. These models have been instrumental in establishing relationships between crash occurrences and a wide array of risk factors, thereby facilitating the prediction of crashes and the assessment of their severity across different roadway environments. These models serve as the foundation for the development of Safety Performance Functions (SPFs), which are extensively used in the *Highway Safety Manual* to estimate crash frequency and conduct road safety analysis [6].

Despite these advancements, researchers and practitioners in the field have encountered significant challenges when dealing with datasets characterized by a high prevalence of zero-crash instances and excessive dispersion. Such datasets, often resulting from the low frequency of crashes at individual sites or within specific time periods, present a skewed distribution that traditional count models are not able to handle efficiently. This issue is due to the fact that crash frequency data are mostly characterized by overdispersion, with the variance of crash counts typically exceeding the mean, leading to model misspecification and biased risk factor estimates [7,8].

In response to these limitations, this study introduces an interpretable machine learning-based hurdle model framework tailored for analyzing zero-inflated crash frequency data. The research adopts a two-stage modeling process that utilizes advanced machine learning algorithms (specifically, CatBoost). The first stage employs a binary classification to identify road segments with potential crash occurrences, addressing data imbalance through a customized loss function. The second stage focuses on predicting the frequency of crashes given their occurrence, again using a customized loss function to handle count data. Additionally, SHapley Additive exPlanations (SHAP) analysis is integrated to interpret the model outcomes, offering detailed insights into the factors affecting crash likelihood and frequency. The use of CatBoost, a machine learning algorithm, enables our model to capture complex, non-linear relationships between variables—an issue that traditional models struggle with due to their linear structure. Additionally, the integration of SHAP analysis enhances the interpretability of our model, providing detailed insights into the contribution of each factor to crash occurrences and frequencies. This combination of handling zero-inflation, accommodating complex relationships, and ensuring interpretability sets our model apart from traditional approaches, making it more suited for real-world crash data analysis.

The objective of this study is to develop and validate a machine learning-based hurdle model utilizing the CatBoost algorithm, known for its superior handling of categorical data and complex, non-linear relationships. By leveraging CatBoost's classification and regression capabilities within a two-stage framework, we aim to demonstrate enhanced accuracy and interpretability compared to traditional statistical methods. Equipping stakeholders with this more robust tool for crash data analysis will facilitate the development of finely tuned safety interventions, ultimately contributing to the reduction in traffic crashes and their associated impacts.

The remainder of this paper is organized as follows: Section 2 reviews traditional statistical and modern machine learning approaches in crash data modeling, specifically addressing issues like zero-inflation and overdispersion in traffic crash frequency data. Section 3 outlines our proposed methodology, introducing a two-stage machine learning hurdle model that leverages classification and regression to handle zero-inflated crash data. Section 4 presents an empirical assessment using real-world crash data from South Korea to evaluate the model's robustness and effectiveness, focusing on validating the model's

predictive accuracy and interpretability. In Section 5, we discuss key findings, limitations, and implications for traffic safety policy and interventions. Finally, Section 6 concludes this study, summarizing the contributions and suggesting future research directions.

2. Literature Review

2.1. Traditional Models

Crash data often exhibit overdispersion and zero-inflation, which pose significant challenges for traditional statistical models like the Poisson and negative binomial models. Overdispersion occurs when the variance of crash counts exceeds the mean, leading to model misspecification and biased risk factor estimates. Zero-inflation refers to the high incidence of zero-crash data, where many observed counts are zero, skewing the distribution and complicating the modeling process. These characteristics make it difficult to accurately model crash data using conventional methods. Overdispersion and zero-inflation are common issues in traffic crash data due to the sporadic nature of crashes at specific locations and times. The Poisson model, which assumes that the mean and variance of crash counts are equal, is often inadequate for such data. This inadequacy can lead to underestimated standard errors and misleading inferences [9]. To address overdispersion, the negative binomial model introduces an extra parameter to account for the variance exceeding the mean. However, this model still struggles with zero-inflation, where the frequency of zero counts is higher than what the model predicts [10].

The hurdle model is another approach that is used to address zero-inflated data. It consists of two parts: a binary component to model the occurrence of zeros, and a count component to model the positive counts. The hurdle model has been successfully applied in various studies to analyze crash data with excess zeros [11–14]. This model is particularly useful when the data generation process for zeros is different from that for positive counts, allowing for a more flexible and accurate representation of the data. Moreover, advanced statistical techniques such as generalized estimating equations (GEEs) and hierarchical Bayesian models have been employed to handle the complexities of crash data [15]. GEEs account for the correlation between observations in longitudinal data, making them suitable for repeated measures of crash frequency over time. Hierarchical Bayesian models, on the other hand, allow for the incorporation of prior information and hierarchical structure in the data, providing robust estimates even in the presence of overdispersion and zero-inflation [16]. The Poisson–gamma model, a variant of the negative binomial model, introduces a gamma-distributed random effect to account for extra-Poisson variability [17]. This model can handle overdispersion better than the standard negative binomial model but still may not fully address zero-inflation. Another advanced approach is the use of finite mixture models, which assume that the population is composed of several subpopulations, each with its own distribution of crash counts [18].

2.2. Machine Learning Approaches

Despite the improvements offered by these advanced models, they often require complex estimation procedures and substantial computational resources. Additionally, the interpretability of these models can be limited, especially when dealing with high-dimensional data and numerous predictors. This has led researchers to explore alternative modeling approaches, such as machine learning techniques, which can offer greater flexibility and predictive power. Machine learning models, particularly those like CatBoost, offer flexibility and improved performance in dealing with non-linear relationships in crash data. Unlike traditional statistical models, machine learning approaches do not require predefined functional forms and can handle high-dimensional data, making them suitable for complex traffic datasets [19]. Machine learning models can also incorporate various data types and interactions, providing a more comprehensive analysis of the factors influencing crash frequency [20]. The efficacy of machine learning in crash frequency analysis has been underscored through various studies; for instance, Xie et al. [21] and Li et al. [22] employed methods such as Bayesian neural networks (NNs), and support vector machines

(SVMs), respectively, and found these to surpass the negative binomial model in accuracy. Following this, Abdel-Aty and Haleem [23] and Haleem et al. [24] successfully applied multivariate adaptive regression splines (MARS) to crashes at un-signalized intersections and urban freeway interchanges, showcasing MARS's capability to capture complex, non-linear relationships and interactions. Furthermore, Zeng et al. [25] utilized a neural network model to delve into the non-linear dynamics between risk factors and crash frequency, incorporating techniques to enhance model interpretability and reduce overfitting. More recently, Zhang et al. [26] explored ensemble machine learning (EML) techniques, including random forest and Light Gradient-Boosting Machine (LightGBM), among others, for their application in crash frequency analysis. LightGBM, known for its efficiency and minimal memory demands, has shown promise in severity analysis and is now being considered for frequency modeling [27,28].

2.3. Addressing Limitations in Crash Analysis Models

Traditional statistical models like the Poisson and negative binomial models have served as the foundation of crash frequency analysis due to their simplicity and capacity to model count data. However, as the literature reveals, these models struggle with data complexities that are commonly encountered in crash datasets, such as overdispersion and zero-inflation. This is particularly problematic in datasets with high proportions of zero-crash observations, where traditional count models may yield biased or inaccurate estimates. Although various advanced statistical approaches, such as the negative binomial–Lindley, zero-inflated, and finite mixture models, have attempted to address these limitations, they often require intensive parameterization and complex estimation methods that can hinder interpretability and scalability.

Machine learning models have been explored in recent studies to overcome some of these challenges by offering greater flexibility in capturing non-linear relationships and accommodating high-dimensional data. However, a key limitation of many machine learning approaches is the "black box" nature of their predictions, which can obscure insights into variables' importance and interactions, limiting their interpretability and, ultimately, their practical utility in policymaking and intervention planning.

To address these gaps, this study introduces a machine learning-based hurdle model using CatBoost, a gradient-boosting algorithm that excels in handling categorical data and managing overfitting through ordered boosting techniques. This approach not only provides the flexibility required to model the complex relationships inherent in zero-inflated crash frequency data but also leverages SHAP to improve interpretability. This combination offers a novel, interpretable machine learning framework that builds upon the existing literature by enhancing accuracy in the prediction of crash frequencies while offering stakeholders detailed insights into variable contributions, which are essential for targeted road safety interventions.

3. Proposed Methodology

3.1. Generalized Machine Learning Hurdle Model

The hurdle model is a statistical approach designed to handle count datasets characterized by an excess of zero outcomes. This is particularly relevant in the crash frequency analysis, where the number of crashes recorded for different segments of a road network can exhibit a large number of zero counts due to the absence of crashes in many segments during a specified observation period. Given a set of covariates X, the hurdle model for crash frequency can be formally specified as follows [29]:

$$\mathrm{P}(\mathrm{Y}=\mathrm{C}\mid\mathrm{X})=\begin{cases}\pi(x), & if\ C=0\\ (1-\pi(x))\cdot f(C;\lambda(X)), & if\ C>0\end{cases} \quad (1)$$

where the probability of observing a crash count C, given a set of explanatory variables X, is modeled by a piecewise function. If C equals zero, the probability is denoted by $\pi(x)$, representing the likelihood of observing no crashes. On the other hand, if C is

greater than zero, the probability is a product of two terms: $(1 - \pi(x))$, and a probability mass function $f(C; \lambda(X))$. This probability mass function characterizes the distribution of positive crash counts and is typically modeled by a distribution suitable for count data, like the Poisson or negative binomial distribution, with $\lambda(X)$ symbolizing the parameter of the distribution—generally the mean rate of crashes, which is formulated as a function of the covariates X.

The binary component $\pi(x)$ serves as a "hurdle" that the model must overcome to observe positive counts. Subsequently, the count component $f(C; \lambda(X))$ elucidates the distribution of crash counts conditional upon surpassing the hurdle, signifying that at least one crash has occurred. The expected value of the crash count, conditional on X, especially for non-zero outcomes, is then given as follows:

$$E[Y \mid X] = P(Y > 0 \mid X) \cdot E[Y \mid Y > 0, X] + P(Y = 0 \mid X) \cdot E[Y \mid Y = 0, X] \quad (2)$$

$$E[Y \mid X] = P(Y > 0 \mid X) \cdot E[Y \mid Y > 0, X] \quad (3)$$

where $E[Y \mid Y > 0, X]$ is the expected value of the count variable Y given that at least one event (crash) has occurred, derived from the positive part of the hurdle model. The overall expectation effectively adjusts the expected count to account for the hurdle at zero. The absence of the term $P(Y = 0 \mid X) \cdot E[Y \mid Y = 0, X]$ is justified by the fact that this term equals zero, since $E[Y \mid Y = 0, X]$ is inherently zero. Hence, it does not contribute to the expectation and is excluded from the equation.

The traditional model employs a binary choice model, usually as logistic regression, to estimate the probability of a non-zero count, $P(Y > 0 \mid X)$, followed by a truncated count data model, such as the Poisson or negative binomial model, to estimate $E[Y \mid Y > 0, X]$. However, this conventional methodology may fall short when faced with the high-dimensional and complex nature of transportation data, necessitating a more nuanced approach.

For improved crash frequency analysis, a two-stage machine learning-based framework is proposed. It is built upon the foundation laid by the hurdle model. This proposed framework adapts the traditional hurdle model to accommodate a broader set of predictive factors and potential non-linearity in the data, enhancing its applicability to real-world scenarios, where the crash frequency is influenced by complex and often interrelated factors. The first stage harnesses machine learning to ascertain the binary likelihood of crash occurrence, effectively segmenting road sections into those without incidents and those with potential incidents. Subsequently, the second stage applies machine learning to estimate the frequency of crashes given their occurrence, thereby addressing the conditional aspect of the distribution. In a generalized two-stage framework for crash frequency analysis, the objective is to employ distinct machine learning models for both stages, ensuring that each stage is tailored to the specific aspect of the crash frequency distribution that it is intended to model. Figure 1 illustrates the conceptual framework of the generalized machine learning-based hurdle model for crash frequency analysis.

The first stage is a binary classification model that predicts the likelihood of a road segment experiencing any crashes. The outcome of this model is binary: either a crash occurs (at least one), or it does not (zero crashes). Advanced machine learning classifiers such as logistic regression, support vector machines, boosting tree machines, or neural networks can be used in this stage, depending on the complexity of the data and the non-linear relationships between the features. The output of the binary classification model denoted $\phi(x)$ can be predicted from the training data. Before building the model, it is necessary to create a binary outcome where all positive counts are transformed into one class (coded as "1", for example) and zero counts are another class (coded as "0", for example). This transformation simplifies the target variable to a binary outcome suitable for classification algorithms.

Following the classification stage, the second stage employs a machine learning regressor to predict the frequency of crashes, specifically focusing on segments where at least one

crash has occurred. This regressor might be a gradient-boosting machine, a generalized additive model, a neural network-based regression model, or any machine learning model designed for count data, which can effectively process the intrinsic complexities of non-zero crash counts. The regression model, denoted $\mu(x)$, can be estimated from training data using positive counts only.

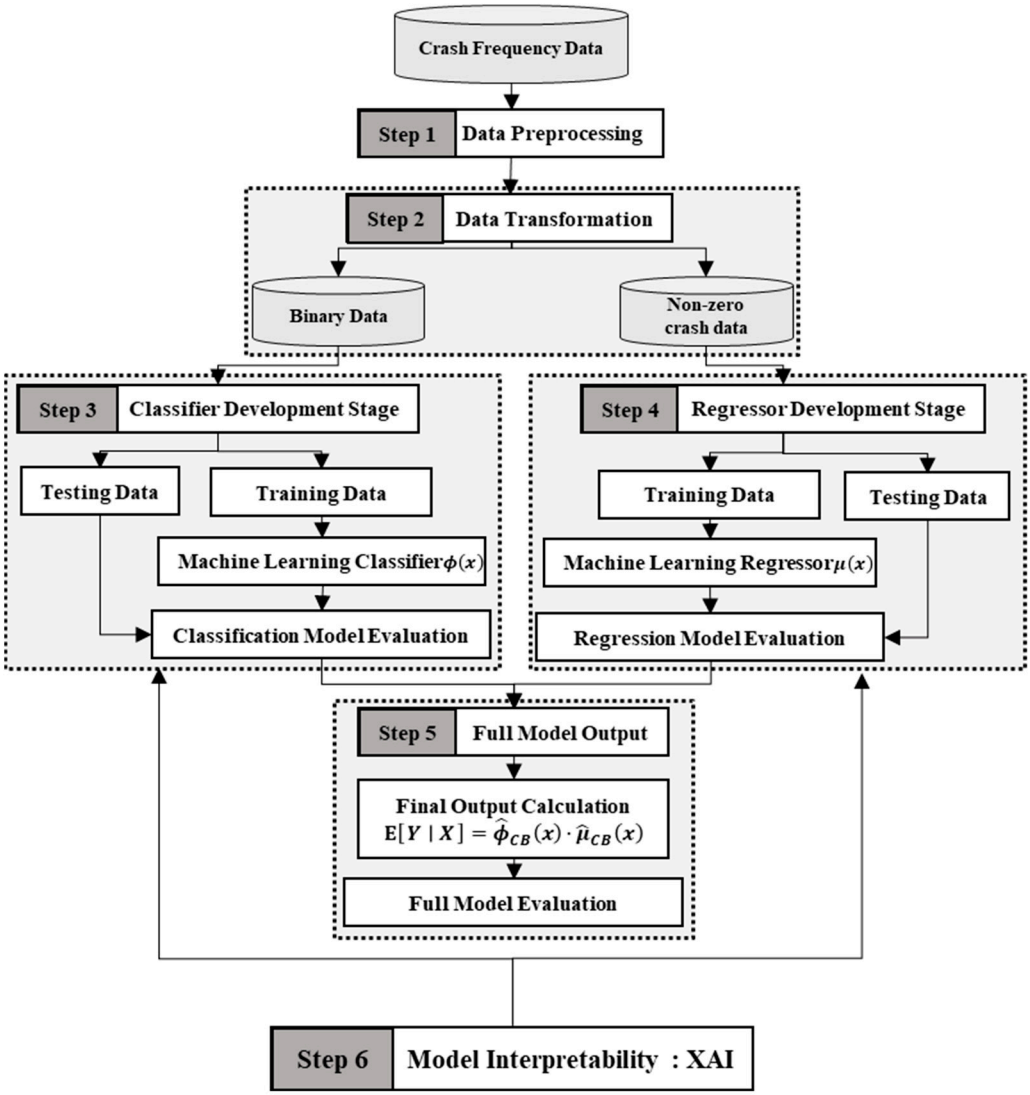

Figure 1. Conceptual framework of the generalized machine learning-based hurdle model.

Based on the above, the expected crash frequency for a given road segment, considering all segments, integrates the outcomes from both stages, expressed as follows:

$$\mathrm{E}[\mathrm{Y} \mid \mathrm{X}] = \hat{\phi}(x) \cdot \hat{\mu}(x) \tag{4}$$

where $\hat{\phi}(x)$ is the probability of the road segment experiencing any crashes, obtained from the machine learning classifier $\phi(x)$, and $\hat{\mu}(x)$ is the expected number of crashes given that the segment has already surpassed the hurdle of zero crashes obtained from the machine learning regressor $\mu(x)$.

By employing separate models, the framework accommodates different data distributions and relationships within the two stages and allows for capturing the complex, non-linear relationships in the data, thus providing a robust prediction mechanism for crash frequency. This dual-model approach may significantly enhance the understanding and prediction of crash frequencies, thereby informing more effective road safety strategies.

The following steps outline the methodological framework applied in this study to develop and validate a machine learning-based hurdle model for crash frequency analysis, as illustrated in Figure 1:

Step 1: Data Preprocessing

Data preprocessing is essential for ensuring that the dataset is clean, reliable, and suitable for machine learning analysis. This step includes the following:

- **Data Cleaning**: Removing redundant, erroneous, or inconsistent entries that could bias the analysis. For example, duplicate records or missing values are identified and handled to ensure data quality.
- **Feature Engineering**: Creating or transforming variables to better capture underlying patterns in the data. This may involve modifying road segment attributes or combining features to enhance the dataset's predictive power.
- **Variable Transformation**: Converting categorical variables into numerical representations if needed, and applying transformations to ensure compatibility with machine learning algorithms.
- **Dataset Splitting**: Dividing the dataset into training and testing sets, typically with an 80:20 split, to enable unbiased model evaluation on unseen data.

Step 2: Data Transformation

Once the data are preprocessed, they are further transformed to create two distinct datasets tailored for the binary classification and regression stages. This step prepares the data to suit the requirements of each model within the hurdle framework:

- **Binary Data Creation**: For the classification stage, a binary dataset is generated, where the target variable indicates the presence or absence of crashes. This binary outcome facilitates training the classifier to distinguish between crash-prone and non-crash segments.
- **Non-Zero Data Creation**: For the regression stage, a subset of the data containing only non-zero crash counts is created. This dataset allows the regression model to focus solely on segments where crashes are predicted, enabling accurate frequency prediction.

Step 3: Classifier Development Stage

This stage involves developing a binary classification model to identify road segments with potential crash occurrences. Key actions include the following:

- **Dataset Splitting:** Dividing the data into training and testing sets to evaluate the model's performance on unseen data.
- **Model Selection:** Choosing a classification algorithm suited for high-dimensional and categorical data.
- **Handling Class Imbalance:** Applying methods to balance the class distribution, increasing sensitivity to the minority class (crash cases).
- **Classification Model Evaluation:** Assessing the classification model's effectiveness in identifying crash-prone segments using metrics like accuracy, precision, recall, and F1 score.

Step 4: Regressor Development Stage

For segments where a crash is predicted, the model advances to the regression stage to estimate crash frequency. Steps include the following:

- **Dataset Splitting:** Similar to the classification stage, dividing data into training and testing sets for unbiased evaluation.
- **Regression Model Application:** Selecting a suitable regression model with a loss function aligned with count-based data.
- **Feature Selection:** Emphasizing key features influencing crash frequency, such as road geometry and traffic volume, to improve accuracy.
- **Regression Model Evaluation:** Using metrics like Root-Mean-Square Error (RMSE) and Mean Absolute Error (MAE) to assess prediction accuracy.

Step 5: Full Model Output

After the classification and regression stages are complete, the final model output is calculated by combining the results from both models:

- **Final Output Calculation:** The expected crash frequency is computed, integrating the predictions from the classifier and the regressor.
- **Full Model Evaluation**: The combined model's performance is evaluated to ensure that it provides a reliable analysis of crash likelihood and frequency. This overall evaluation includes both the classification and regression results to confirm the model's robustness.

Step 6: Model Interpretability

Interpretability is crucial for ensuring that model predictions can inform practical safety interventions. This step involves using Explainable AI (XAI) techniques to make the model's predictions interpretable, offering insights into the contribution of each feature to crash predictions.

- **Interpretation for Classification**: Interpretability techniques for the classification model indicate the impact of each feature on the likelihood of a crash occurring, helping to clarify which factors most strongly influence the outcome.
- **Interpretation for Regression**: In the regression model, interpretability methods show how specific features affect the crash frequency predictions. For instance, traffic volume might be associated with increased crash frequency, while other factors could reduce it.
- **Insights for Road Safety Planning**: By making the model predictions interpretable, stakeholders gain actionable insights into the factors driving crash risks. This information supports targeted interventions, such as adjustments to road design or safety measures, to enhance traffic safety.

Together, these steps ensure a comprehensive analysis of crash data, from initial data preparation to the interpretability of predictions, facilitating the development of accurate and actionable road safety strategies.

3.2. CatBoost Hurdle Model

The standard hurdle model, employed in crash frequency analysis, traditionally features a binary component that determines crash occurrence and a subsequent count data component to assess crash frequency. Conventional models that rely on traditional statistical models often struggle to capture the complex and non-linear interactions present in crash data. CatBoost is an advanced machine learning algorithm that excels in both classification and regression tasks, making it adept at not only discerning the likelihood of crash occurrences through its binary classification capabilities but also predicting the frequency of crashes more accurately with its regression prowess [30]. CatBoost overcomes the limitations of parametric models by efficiently processing categorical features and capturing non-linear relationships, without the need to rely on rigid assumptions (as often required with traditional statistical models). Its capacity for direct optimization of loss functions enables a more nuanced adaptation to the data, enhancing the model's ability to capture the underlying patterns.

CatBoost is a sophisticated variant of Gradient Boosting Trees (GBT), which is an advanced machine learning approach that is suitable for both regression and classification tasks, leveraging an ensemble of decision trees. The concept revolves around sequentially correcting the prediction errors of an aggregation of simple models, typically decision trees, using a gradient descent optimization approach. In a decision tree, which forms the base learner in GBT, each internal node represents a test on an attribute, branches denote the outcome of the test, and leaf nodes correspond to output values. In regression scenarios, these outputs are continuous values, often the average of the target values of the instances within that leaf. For classification, leaf nodes represent class labels, usually determined by the majority class among instances in that region. The GBT model is built iteratively, with each tree constructed to correct the errors made by the ensemble of previously built trees.

This process is guided by the gradient of a loss function, which quantifies the difference between the observed and predicted values. In each iteration of the GBT algorithm, the negative gradient of this loss function is computed for each instance. These gradients serve as the target values for training the subsequent tree in the ensemble.

CatBoost further refines this model, particularly excelling in handling categorical data and complex datasets. It employs an ordered boosting technique to reduce the risk of overfitting, a common challenge in standard boosting methods. In this technique, for each tree, CatBoost uses a random permutation of the dataset, training each tree on a different subset of data defined by the permutation. This method ensures that the tree's training is influenced only by the part of the data that precedes each point in the permutation, making it effective for both regression and classification. Moreover, CatBoost introduces an innovative approach to process categorical features. It converts categorical variables into numerical values during training, using a target-based statistic. This transformation is mathematically represented as follows:

$$\hat{x}_{\sigma(j),p} = \frac{\sum_{i=1}^{p-1}\left[x_{\sigma(i)} = x_{\sigma(p)}\right] y_{\sigma(j)} + qH}{\sum_{i=1}^{p-1}\left[x_{\sigma(i)} = x_{\sigma(p)}\right] + q} \mathrm{j, p} = 1, 2, 3, \ldots, \mathrm{n} \quad (5)$$

where $\hat{x}_{\sigma(j),p}$ is the numerical representation of the j^{th} feature in permutation p; $\left[x_{\sigma(j)} = x_{\sigma(p)}\right]$ is the Iverson bracket that serves as an indicator function, with output 1 in case $x_{\sigma(j)} = x_{\sigma(p)}$, and 0 otherwise; $y_{\sigma(j)}$ is the target variable; H is the a priori distribution term; and q is the weight coefficient of the a priori distribution term.

In addition, CatBoost has demonstrated better performance than other machine learning models in dealing with heterogeneous datasets [31]. Given the heterogeneous nature of road traffic accident data, CatBoost may be a more suitable choice for the current work. Reformulating the generalized hurdle model with the integration of the CatBoost algorithm led to the CatBoost hurdle model, expressed as follows:

$$\mathrm{E[Y \mid X]} = \hat{\phi}_{CB}(x) \cdot \hat{\mu}_{CB}(x) \quad (6)$$

where $\hat{\phi}_{CB}(x)$ denotes the probability outcome of the binary classification CatBoost algorithm ϕ_{CB} trained on crash versus non-crash data, and $\hat{\mu}_{CB}(x)$ denotes the expected outcome of the regression CatBoost algorithm μ_{CB} trained on the positive crash count data.

3.3. Customization of Loss Functions

3.3.1. Classification Stage

In the first stage of the proposed two-part model for crash frequency analysis, we confront the significant challenge of data imbalance, which is a common issue in various fields, particularly pronounced in traffic accident analysis. This imbalance arises due to a disproportionate representation of "non-crash" instances compared to "crash" instances, leading to a skewed dataset where the majority class overwhelmingly outnumbers the minority class. Such a scenario can severely bias a predictive model towards predicting the majority class, thereby diminishing its performance in accurately identifying crash occurrences. To address this challenge, a custom loss function that specifically targets the imbalance problem is incorporated in the binary classification CatBoost model. The mathematical foundation of the proposed approach is rooted in modifying the traditional log-loss, a widely used metric in binary classification tasks, to account for the class imbalance. The log-loss for a binary classifier is defined as follows:

$$\mathrm{L(y, p)} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \quad (7)$$

where L is the log-loss, N is the number of observations, y_i is the actual label of the i^{th} observation, and p_i is the predicted probability of the i^{th} observation belonging to the positive class.

To customize this loss function for handling data imbalance, we introduce sample weights that are inversely proportional to the class frequencies. Given a dataset with classes C_j, where C_0 represents the majority class (non-crash) and C_1 the minority class (crash), the number of instances in each class is denoted by n_{C_0} and n_{C_1}, respectively. The sample weight for an instance belonging to class C_j is computed as follows:

$$w_{C_j} = \frac{1}{n_{C_j}} \tag{8}$$

This weighting scheme ensures that the loss contribution of each sample is inversely proportional to the prevalence of its class, thereby amplifying the model's focus on the minority class. The weighted log-loss for a binary classification model can be expressed as follows:

$$L_{weighted}(\mathrm{y}, \mathrm{p}) = -\frac{1}{N}\sum_{i=1}^{N} w_{C_j}[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \tag{9}$$

This formulation allows for a more balanced treatment of classes, enabling the model to improve its sensitivity to the minority class without losing specificity to the majority class.

3.3.2. Regression Stage

In the regression stage of the proposed framework for crash frequency analysis, the focus shifts towards accurately predicting the actual number of crashes, given that a crash is predicted to occur. This necessitates a nuanced approach to modeling count data, which inherently possesses unique characteristics and challenges. CatBoost, by default, utilizes the Mean Squared Error (MSE) loss function for regression tasks. Given the specific nature of crash frequency data, which predominantly comprise count data, MSE might not be the most effective loss function. Count data, especially in the context of crash frequency, can exhibit overdispersion, where the variance exceeds the mean, or underdispersion, where the variance is less than the mean. To address this and better model the count nature of crash data, the utilization of the Poisson loss function is proposed for the regression stage of the framework. The Poisson distribution is a natural choice for modeling count data, particularly for its capacity to handle the discreteness and non-negative nature of such data. The Poisson loss function, often referred to as the log-likelihood of the Poisson distribution, is given as follows:

$$L_{Poisson} = -\sum_{i=1}^{n}(y_i \log(\hat{y}_i) - \hat{y}_i - \log(y_i!)) \tag{10}$$

where y_i represents the actual count value, $\hat{y}_i$ is the predicted count value, and n is the total number of observations. The term $\log(y_i!)$ in the Poisson loss function does not vary with the predictors and is solely dependent on the actual counts y. This independence from the model's inputs means that, during optimization, the term does not influence the model's ability to learn from the data. Consequently, for model training and optimization, this constant term can be omitted without affecting the relative evaluation of different model configurations. Therefore, the final Poisson loss function becomes

$$L_{Poisson} = -\sum_{i=1}^{n}(y_i \log(\hat{y}_i) - \hat{y}_i) \tag{11}$$

The utilization of the Poisson loss function aligns the model's optimization process with the distributional properties of count data. Unlike the MSE, which penalizes deviations from the actual values uniformly, the Poisson loss introduces a dependence on the actual counts, making the penalty for deviation more sensitive to the scale of the data. This is

particularly beneficial in crash frequency modeling, where the range of counts can vary significantly across observations.

3.4. Model Interpretability

Upon completing the training and construction of the model, the necessity for interpretability becomes paramount to understand the influence of variables on the predicted outcomes, particularly how road geometry factors affect crash frequency. Traditional statistical models offer insights into variable weights directly, enabling straightforward sensitivity analysis. However, machine learning models are often perceived as "black boxes" due to their complex internal mechanisms, making the task of interpreting these models more challenging. In recent years, efforts have been made to develop methodologies to allow for the interpretability of machine learning models to reveal their inner workings. Among these methodologies, SHAP (SHapley Additive exPlanations) stands out as a powerful tool for model interpretability. SHAP leverages the concept of Shapley values from cooperative game theory, providing a robust framework to attribute the prediction of a model to its input features. The Shapley value is a method to fairly distribute the "payout" (prediction) among the "players" (features), based on their contribution to the outcome.

Mathematically, the contribution of a feature value to a prediction is determined by comparing what a model predicts with and without the feature, averaged over all possible combinations of other features. For a given prediction, the SHAP value of a feature is the average marginal contribution of that feature across all possible coalitions. Formally, for a prediction model f and a feature set S excluding the feature i, the Shapley value ϕ_i is calculated as follows:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [f(S \cup \{i\}) - f(S)] \tag{12}$$

where N is the set of all features, S is a subset of features excluding i, $|S|$ is the cardinality of S, and $f(S)$ represents the model prediction with features in S.

The term $f(S \cup \{i\}) - f(S)$ represents the marginal contribution of feature i to the prediction. This measures how much the feature i changes the model's prediction when added to the subset S of other features. The summation involves averaging this marginal contribution over all possible subsets of features S that do not include i. This ensures that the contribution of i is fairly evaluated in the context of all possible interactions with other features. The weighting factor $\frac{|S|!(|N|-|S|-1)!}{|N|!}$ ensures that each subset combination is given an appropriate weight. This factor is derived from combinatorial principles, ensuring that contributions from features are fairly balanced regardless of their position or the number of other features. SHAP values fairly distribute the total difference between the model's actual prediction and a baseline value (e.g., the mean prediction) among all features. This ensures that each feature's contribution is assessed in the context of its interactions with all other features. In summary, the SHAP value calculation takes into account all possible scenarios in which a feature interacts with other features, making it a powerful tool for capturing both the main effects and the interaction effects of the feature on the prediction. By aggregating these values, SHAP offers a comprehensive view of how each feature contributes to the model's predictions, whether by increasing or decreasing the predicted outcome.

In the context of the proposed two-component framework (classification and regression), interpretability is applied separately to each component. For the classification stage, SHAP analysis identifies factors contributing to crash occurrence, while for the regression stage it determines factors influencing crash counts. This dual approach to interpretability mirrors the structure of traditional two-part models, such as hurdle models and zero-inflated models, which also produce separate sets of coefficients for each component of the analysis.

4. Empirical Assessment Using Real-World Data

This section presents an empirical assessment of the proposed machine learning-based hurdle model, utilizing real-world crash data to validate the framework's performance and robustness. The assessment aims to demonstrate the practical applicability of the model in handling zero-inflated crash frequency data, as well as to showcase its capacity to deliver accurate predictions of crash occurrences and frequencies. By applying the model to a dataset of road crashes from South Korea, we provide a systematic evaluation across both stages of the hurdle framework: the binary classification of crash occurrence, and the regression of crash counts for segments with non-zero crashes.

The focus here is not on conducting a detailed analysis of the specific patterns or insights within the data themselves, as this would require additional data and a broader scope of investigation. Instead, the emphasis is on testing the effectiveness and adaptability of the proposed methodology in a real-world context, offering initial evidence of its potential for practical deployment. The results highlight the model's ability to address the challenges of zero-inflation and overdispersion, providing a foundation for further research that may include more extensive case studies in future work.

4.1. Data

The dataset used in our analysis covers the years 2011 to 2015. This timeframe was chosen to gather a comprehensive set of data related to traffic accidents on general national roads, focusing on understanding the factors contributing to road safety. The primary method for data collection involved the use of the ARASEO (Automated Road Analysis and Safety Evaluation Tool) vehicle, an advanced road inspection vehicle equipped with technology for capturing detailed road geometry [32]. This vehicle collected data on the physical characteristics of the roads, including road geometry data. Measurements were taken at one-meter intervals, providing the high level of detail necessary for accurate safety analysis. The geometric raw data collected by the ARASEO vehicle, along with data from the road registry, were processed and analyzed using geographic information systems (GISs). Additionally, the obtained data were combined with traffic volume and historical accident data to create the final database, which could be used for further analysis. The final data were obtained from the Korea Institute of Civil Engineering and Building Technology (KICT) in South Korea.

Following the necessary data preprocessing, which included the removal of redundant and erroneous entries, the finalized dataset consisted of 8636 road segments and encompassed a total of 3182 crashes recorded over the specified period. As depicted in Figure 2, the frequency of crashes across these segments varied, with counts ranging from 0 to 24 incidents. Notably, the dataset displayed a pronounced skew towards zero counts. There were 7052 road segments with no recorded crashes, constituting approximately 81% of the dataset. This high prevalence of zeros poses a particular challenge for predictive modeling and underscores the necessity for a model adept at handling zero-inflated data, a key attribute of the new model proposed in this study.

The data include a mix of continuous and categorical variables reflecting the physical characteristics of the road environment and traffic patterns. For continuous factors, we have measurements such as the length of homogeneous road sections, the degree of road slope, the curvature of the road, lane widths, median widths, and the road shoulder width. Traffic volume is quantified by the annual average daily traffic (AADT), providing an indicator of road utilization intensity. A detailed description and summary statistics of continuous variables, including the dependent variable, is provided in Table 1. The categorical data introduce indicators for elements like the number of lanes, the presence of climbing lanes (which allow slower-moving vehicles to move aside), median barriers, and guardrails, all of which are safety features that can potentially influence crash occurrences. The presence of street lights, distinguishing between well-lit and poorly lit stretches of road, is also included. Additionally, the roads are categorized into rural or suburban segments, which

can inform the analysis based on varying traffic behaviors and environmental conditions. Table 2 shows summary statistics of the categorical variables.

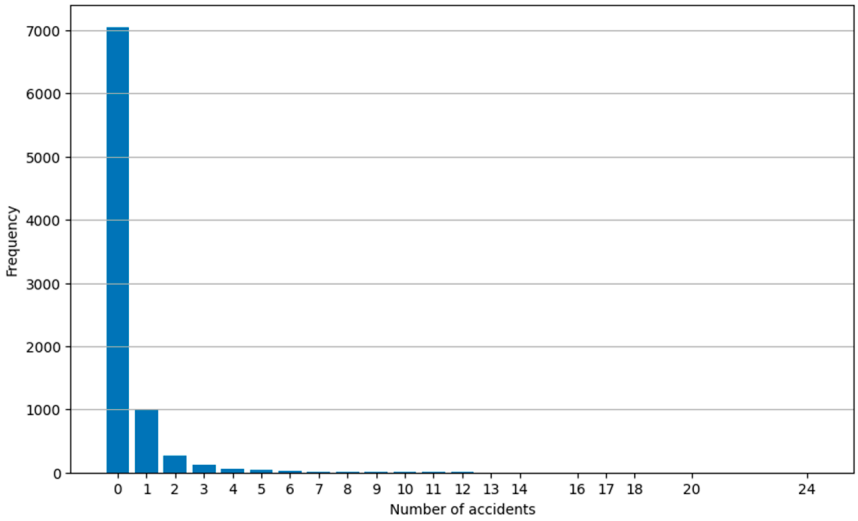

Figure 2. Histogram of road crash counts (dependent variable).

Table 1. Summary statistics of continuous variables.

Variable	Definition (Unit)	Mean	Standard Deviation	Minimum	Maximum
Crash Count	Crash frequency count	0.37	1.21	0	24
Section Length	Homogeneous section length (meter)	129.90	131.44	50	3100
SLOPE	Vertical slope absolute value (%)	1.94	1.76	0	12.05
Curve Radius	Horizontal curve radius (meter)	272.21	437.62	0	2000
Road Width	Lane width (meter)	3.79	0.68	2.5	5
Median Width	Median width (meter)	0.91	0.79	0	4
Shoulder Width	Shoulder width (meter)	0.59	0.69	0	11.3
AADT	AADT	9306.97	9253.74	321	54,307

Table 2. Summary statistics of categorical variables.

Variable	Definition	Category	Frequency	Proportions
Lane Count	Indicator variable for number of lanes	1	3393	39.29%
		2	5147	59.60%
		3	80	0.93%
		4	16	0.19%
Climb	Indicator variable for climbing lanes	1	180	2.08%
		0	8456	97.92%
Median	Indicator variable for median barriers	1	4711	54.55%
		0	3925	45.45%
Guardrail	Indicator variable for guardrails	1	6510	75.38%
		0	2126	24.62%
Street Light	Indicator variable for street lights	1	535	6.19%
		0	8101	93.81%
City Division	Indicator variable for road category	1 (Rural)	7214	83.53%
		0 (Suburban)	1422	16.47%

4.2. Experimental Design

This section presents a detailed methodology for the real-world assessment of the proposed machine learning-based hurdle model for analyzing zero-inflated crash frequency data. The evaluation focuses on three key areas: the performance of the classification stage, the performance of the regression stage, and the overall model results. Figure 3 provides a comprehensive overview of the entire process, including data preprocessing, model training, and evaluation. The real-world assessment aims to validate the model's practical applicability and robustness in predicting crash occurrences and frequencies. This assessment is conducted in a structured manner, ensuring that each stage of the model is rigorously tested and compared against several benchmark models.

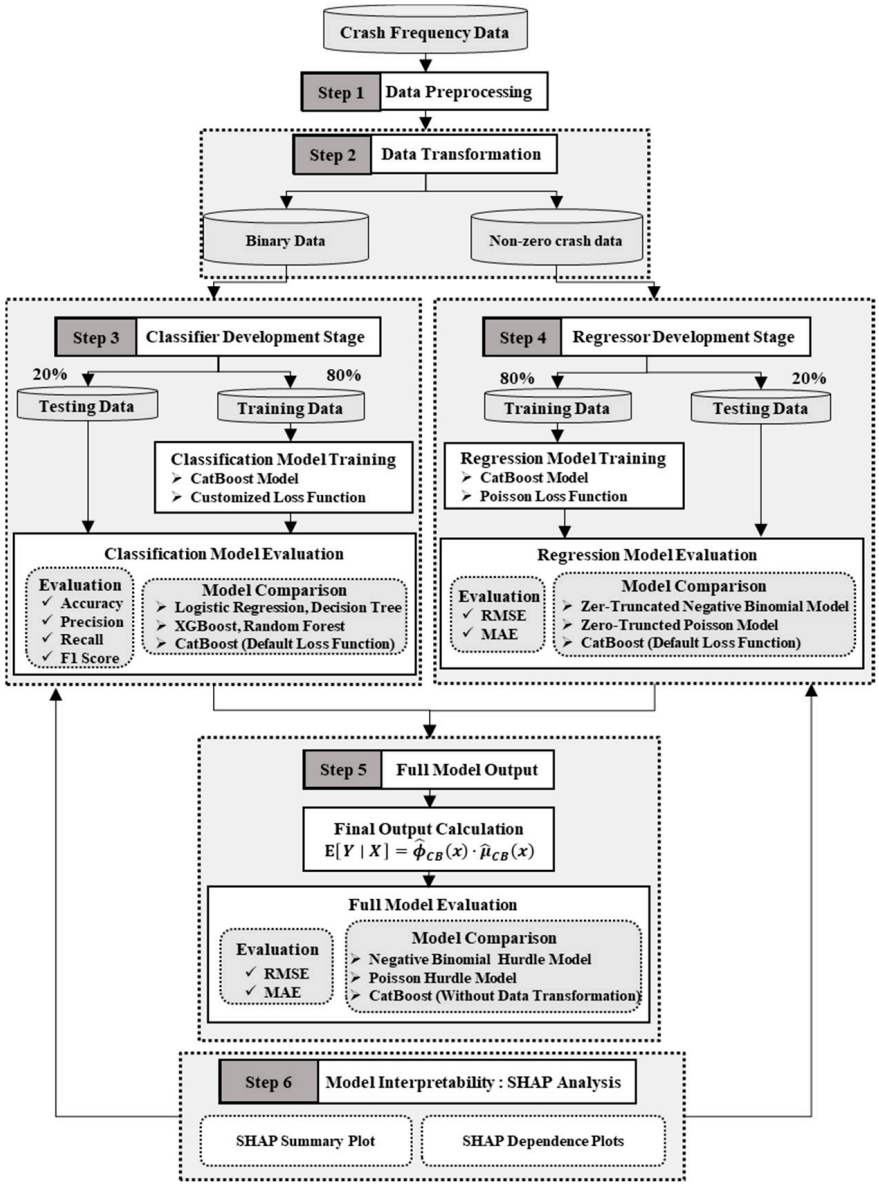

Figure 3. Flowchart of the proposed model assessment approach.

4.2.1. Classification Stage

The CatBoost model was introduced as a robust machine learning algorithm that excels in handling categorical features and preventing overfitting. For our current analysis of crash frequency data, we employed the CatBoost model, with a particular focus on its loss function capabilities. We utilized the customized loss function to tackle the class imbalance problem inherent in crash frequency data, where non-occurrences (zeroes)

significantly outnumber occurrences (crashes). This customization enables the model to pay closer attention to the minority class, thereby improving the sensitivity of crash prediction. Additionally, we evaluated the performance of the CatBoost model using its default loss functions for comparison purposes, in order to demonstrate the effectiveness of custom loss functions in improving model performance for imbalanced datasets. The performance was compared against benchmark models.

In traditional hurdle models for zero-inflated crash frequency data, logistic regression (LR) is commonly employed in the first stage to distinguish zero counts from positive counts. Unlike linear regression, which is unsuitable for binary outcomes, LR is apt for categorization tasks, since it predicts probabilities that naturally lie between zero and one. The LR model, in our context, is aimed at discerning the binary response variable: whether or not a crash has occurred within road segments. The LR model formulates the relationship between the binary response variable y, where $y \in \{0,\ 1\}$, and the predictors $\mathrm{X} = (x_1, x_2, \ldots, x_m)$. The predictors comprise various road geometry characteristics and traffic metrics, such as road curvature, lane width, and traffic volume. The probability of a crash occurrence, denoted as $\mathrm{P}(Y = 1|X)$, is modeled by the logistic function:

$$\mathrm{P}(Y = 1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m)}} \tag{13}$$

where β_0 is the intercept, and $\beta_1, \ldots, \beta_m$ are the estimated coefficients that quantify the impact of each predictor.

The logistic model equation for our current analysis is expressed as follows:

$$\log\left(\frac{\mathrm{P}(Y = 1|X)}{1 - \mathrm{P}(Y = 1|X)}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m \tag{14}$$

This log-odds or logit transformation ensures that the output probability is confined within the range $[0, 1]$.

The decision tree (DT) classifier is a machine learning model that can be employed to categorize road segments based on crash occurrences. The DT model is comparable to a flowchart with a tree structure, where each internal node represents a "question" on features of the data, each branch represents the outcome of that question, and each leaf node represents a class label (outcome).

The Gini index is commonly used as a criterion for creating the decision rules at each node and can be mathematically expressed as follows:

$$Gini(D) = 1 - \sum_{y=1}^{n} p_y^2 \tag{15}$$

where is D is the dataset, n is the number of classes, and p_y is the relative frequency of class y in D. When a dataset D is split into two subsets D_1 and D_2, the Gini index for the split is computed using the following formula:

$$Gini_{split}(D) = \frac{N_1}{N} Gini(D_1) + \frac{N_2}{N} Gini(D_2) \tag{16}$$

where N_1 and N_2 are the sizes of datasets D_1 and D_2, respectively, and N is the total number of instances in D. The goal is to choose the split that minimizes the Gini index, indicating the most homogeneous branches. DTs have parameters that can be fine-tuned—such as the minimum number of samples required to be at a leaf node, or the maximum depth of the tree—to prevent overfitting and ensure the model's generalization to new data.

Random forest (RF) serves as an advanced classification method to address the binary outcome of crash occurrences in road segments. RF operates on the principles of Classification and Regression Trees (CART), amalgamating a multitude of decision trees to increase prediction accuracy and mitigate overfitting—a common drawback of single-decision-tree models. The RF algorithm enhances generalization by constructing a "forest" of trees, each

predicated on a random subset of the data, and making decisions based on a diverse array of predictor variables. Each tree in the forest votes for a class label, and the class receiving the most votes becomes the model's prediction.

The individual trees within an RF model are grown using the following steps:

- **Bootstrap Sampling:** A bootstrap sample from the overall data is used to grow each tree, promoting variability amongst the trees.
- **Random Predictor Selection:** At each split within a tree, a random subset of predictors is considered, which ensures that the trees in the forest are uncorrelated and strengthens the ensemble's predictive power.
- **Out-Of-Bag Error Estimation:** For each tree, the out-of-bag (OOB) data not included in the bootstrap sample serve as a validation set to estimate prediction error, offering an unbiased evaluation of model performance without the need for a separate test set.

The final decision in an RF model is typically made by averaging the predictions (regression) or taking a majority vote (classification) from all of the trees. The model's efficacy is further optimized through parameters such as the number of trees, the maximum depth of each tree, and the minimum samples required at each leaf node. The Information Gain Ratio (IGR), a modification of the traditional Information Gain, is utilized as a splitting criterion in the RF algorithm, calculated as follows:

$$\text{Information Gain Ratio(IGR)} = \frac{\text{Information Gain (X)}}{\text{Split Info (X)}} \tag{17}$$

where X is a randomly chosen example from the training set, Information Gain measures the reduction in entropy or impurity before and after the split, and Split Info measures the potential information generated by the split, accounting for the number and size of branches.

XGBoost is a sophisticated ensemble technique that builds upon the principles of gradient boosting and extends it through advanced regularization techniques and system optimization. The XGBoost algorithm combines the predictions from a series of decision trees, summing the contribution of each to forecast crash occurrences:

$$\hat{y}_i(x) = \sum_{T=1}^{T} f_T(x), \quad (f_T \in F) \tag{18}$$

where $\hat{y}_i(x)$ is the prediction for instance i, T represents the number of trees, f_T is an individual tree, and F signifies the function space of all possible trees.

The objective function of XGBoost, which it seeks to minimize, encompasses a loss function that gauges the accuracy of predictions and a regularization term to mitigate overfitting:

$$Obj(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{T=1}^{T} \Omega(f_T) \tag{19}$$

The loss function is denoted by $(y_i, \hat{y}_i)$, measuring the divergence between the predicted value $\hat{y}_i$ and the actual value y_i. The regularization term, $\Omega(f_T)$, penalizes complexity, reducing the risk of overfitting.

In our analysis, the evaluation of the predictive performance of the models applied to Korean road crash frequency data relied on the foundational tool of the confusion matrix, along with key performance metrics such as accuracy, precision, and recall. This approach is commonly used to evaluate the performance of classification models. A confusion matrix is a tabular representation of actual versus predicted conditions, providing a clear visualization of the performance of a classification model. For a binary classifier, the confusion matrix, as shown in Table 3, consists of two rows and two columns that report the numbers of the following:

- True positives (TP): Correctly predicted positive observations.
- False positives (FP): Incorrectly predicted positive observations (type I error).
- True negatives (TN): Correctly predicted negative observations.
- False negatives (FN): Incorrectly predicted negative observations (type II error).

Table 3. Confusion matrix.

	Actual Values	
Predicted Value	**Positive**	**Negative**
Positive	True positives (TP)	False positives (FP)
Negative	False negatives (FN)	True negatives (TN)

The accuracy indicates the proportion of true results (both true positives and true negatives) to the total number of cases examined. It is defined as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{20}$$

Precision reflects the ratio of true positive observations to the total predicted positives, and it is given as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{21}$$

Recall, or sensitivity, measures the proportion of actual positives correctly identified by the model, emphasizing the model's ability to detect positive instances. It is formulated as follows:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{22}$$

These metrics are essential not only for understanding the overall effectiveness of the model but also for providing insight into the balance between the model's sensitivity (recall) and precision.

Additionally, the F1 score, which is the harmonic mean of precision and recall, is introduced to provide a single metric that balances both concerns. The F1 score is beneficial when the class distribution is imbalanced, as it considers both false positives and false negatives. It is defined as follows:

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{23}$$

The F1 score provides a more comprehensive measure of a model's performance, especially in cases where precision and recall are not evenly balanced. It serves as an overall indicator of the model's accuracy and robustness in classification tasks.

4.2.2. Regression Stage

In the regression stage of our analysis for crash frequency data on Korean roads, we adapted the CatBoost algorithm to function as a regressor. While CatBoost is typically used for classification tasks, it also performs well in regression, particularly when dealing with count data like crash frequencies. As explained earlier, the CatBoost Regressor with a Poisson loss function is specifically tailored for count data, which inherently follow a Poisson distribution. The Poisson loss, also known as log-likelihood for Poisson regression, is suitable for modeling the number of events occurring within a fixed interval, given that these events happen at a constant rate and independently of the time since the last event. For comparison, we also deployed the CatBoost Regressor utilizing its default loss function that employs MSE. This approach served as a benchmark to evaluate the efficacy of employing the Poisson loss function for our specific dataset. The default loss function is generally designed to handle various types of regression tasks, including those where the

target variable is continuous. By comparing the performance of the CatBoost Regressor with Poisson loss to that with the default loss, we aimed to demonstrate the advantages of customizing the loss function to fit the nature of our data, specifically addressing the crash count data. Also, similar to the classification stage, the performance was compared against benchmark models.

The zero-truncated Poisson (ZTP) model is particularly suited for count data that do not include the zero-count category, which is a common scenario in crash data analysis when considering only segments where crashes have occurred. The probability mass function (PMF) is adjusted to account for the absence of zero counts, effectively shifting the distribution to start from one instead of zero. The PMF of a standard Poisson distribution is given as follows:

$$\mathrm{P}(\mathrm{Y} = y) = \frac{e^{-\lambda}\lambda^{y}}{y!} \tag{24}$$

where λ is the rate parameter (mean number of occurrences in an interval), and y is the count of occurrences with $y = 0, 1, 2, \ldots$

For the ZTP model, the PMF becomes

$$\mathrm{P}(Y = y|Y > 0) = \frac{P(Y = y)}{1 - P(Y = 0)} = \frac{\frac{e^{-\lambda}\lambda^{y}}{y!}}{1 - e^{-\lambda}} \quad for\ y = 1, 2, 3, \ldots \tag{25}$$

This adjusted PMF ensures that the probability of zero count is excluded from the model. The denominator, $1 - e^{-\lambda}$, serves to normalize the probabilities so that they sum to 1 over the domain $y = 1, 2, 3, \ldots$, effectively removing the probability mass that would have been at zero in a non-truncated Poisson distribution.

The zero-truncated negative binomial (ZTNB) model is an extension of the negative binomial (NB) distribution that is used for count data with overdispersion, where the variance exceeds the mean and where the count of zero is not observed or is excluded. In a negative binomial (NB) distribution, the probability of observing y events is given as follows:

$$\mathrm{P}(\mathrm{Y} = y) = \frac{\Gamma(y + k)}{\Gamma(\mathrm{k})\mathrm{y}!}\left(\frac{\lambda}{\lambda + k}\right)^{y}\left(\frac{k}{\lambda + k}\right)^{k} \tag{26}$$

where Γ is the gamma function, y is the number of occurrences, λ is the mean number of occurrences, and k is the dispersion parameter.

For the ZTNB model, the probability mass function (PMF) is adapted to account for the absence of zero counts, and it is defined as follows:

$$\mathrm{P}(Y = y|Y > 0) = \frac{P(Y = y)}{1 - P(Y = 0)} \tag{27}$$

Incorporating the condition that $y > 0$ into the negative binomial PMF and substituting the expression for $\mathrm{P}(\mathrm{Y} = 0)$ from the NB distribution, we can obtain the PMF of the ZTNB model:

$$\mathrm{P}(Y = y|Y > 0) = \frac{\frac{\Gamma(y+k)}{\Gamma(\mathrm{k})\mathrm{y}!}\left(\frac{\lambda}{\lambda+k}\right)^{y}\left(\frac{k}{\lambda+k}\right)^{k}}{1 - \left(\frac{k}{\lambda+k}\right)^{k}} \quad for\ y = 1, 2, 3, \ldots \tag{28}$$

This adjusted PMF for the ZTNB model removes the probability weight of $y = 0$ and reallocates it across the positive counts, effectively accommodating the overdispersion and absence of zeroes in the crash data.

The evaluation of the predictive models in the regression stage is conducted using two statistical metrics: Root-Mean-Square Error (RMSE) and Mean Absolute Error (MAE). These metrics are widely used for comparing the accuracy of models by quantifying the differences between the values predicted by the models and the observed values from the generated data.

RMSE is a measure of the average magnitude of the errors between the predicted and actual values, giving higher weight to larger errors. It is defined mathematically as the square root of the average of squared differences between prediction and actual observation. The RMSE for n predictions is computed as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{29}$$

where y_i represents the acual crash counts, and $\hat{y}_i$ represents the predicted crash counts for the i-th observation out of n total observations.

MAE, on the other hand, measures the average of the absolute errors between the predicted and the actual values. Unlike RMSE, MAE treats all errors equally, providing a straightforward measure of prediction accuracy. It is expressed as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{30}$$

MAE is particularly insightful because it is in the same unit as the data being predicted—in this context, the number of crashes.

Both RMSE and MAE are valuable metrics for model evaluation in this study. RMSE is sensitive to outliers, as it tends to penalize larger errors more severely, while MAE provides a simple average error magnitude. Together, they offer a comprehensive assessment of the model's prediction performance, with RMSE emphasizing the model's accuracy across the entire range of data, and MAE providing a measure of central tendency in the prediction errors.

4.2.3. Full Model Assessment and Interpretability

In the full model assessment stage of our analysis, we synthesized the outcomes of the classification and regression stages from our proposed two-stage model to obtain a final prediction of crash frequencies. This integration is crucial for providing a more holistic view of crash risk on Korean roads.

To validate the efficacy and accuracy of our proposed model, we compared it against the negative binomial hurdle (NBH) model and the single-stage CatBoost model. The NBH model is a compound model, combining a binary outcome model for predicting whether crashes occur with a count model for the number of crashes, conditioned on at least one crash occurring. It handles excess zeros and overdispersion, making it well suited for crash frequency data. We also compared our proposed model's performance against a single-stage CatBoost model that directly predicts the crash frequency. This comparison helps us understand the benefit of the two-stage modeling approach in capturing the complex distribution of crash data. The assessment involved comparing the RMSE and MAE between the predicted crash frequencies from our model and those from the benchmark models. By doing this, we can discern the model that provides the closest approximation to the real-world data, and thus, the most reliable for practical use in road safety analysis and intervention planning.

Model interpretability is essential to ensure that the actionable insights derived from machine learning models are transparent and trustworthy, especially in critical fields such as road safety analysis. In this study, we implemented SHAP (SHapley Additive exPlanations) values to interpret the predictions of our model at both the classification and regression stages. The SHAP summary plot and SHAP partial dependence plots are shown and explained below. The SHAP summary plot provides a global view of the impact and

importance of each feature across all predictions. In essence, it visualizes the magnitude and direction (positive or negative impact) of each feature on the model output. For our classification stage, the SHAP summary plot illustrates how each predictor contributes to the likelihood of a crash occurring on a road segment. For the regression stage, it displays how the predictors affect the expected count of crashes, given that at least one crash has occurred. SHAP partial dependence plots take this a step further by showing the effect of a single feature on the prediction outcome while averaging out the effects of all other features. These plots help us to understand complex interactions and non-linear relationships between features and the target variable.

For both the classification and regression stages, examining SHAP values allows us to decompose the model output into the sum of effects from each feature, providing a powerful tool for explanation. This approach aligns with the analysis method used in traditional hurdle models, where the two processes are analyzed separately.

4.3. Results

4.3.1. Classification Stage

In the classification stage of our analysis, the data were strategically split into training and testing sets at an 80:20 ratio, employing stratified sampling to maintain the proportion of the minority class. This sampling strategy ensured that the model was tested against a realistic distribution of crash instances, which is essential when the positive class (crash occurrence) represents a smaller portion of the data. By doing so, we mitigated the risk of developing a biased model that overlooks the less frequent, yet critical, positive cases. The analysis was conducted within the Python 3.8 environment, employing the Scikit-learn library, a robust tool for machine learning analysis. Table 4 shows a summary of each model's classification performance.

Table 4. Confusion matrix.

Classifier	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.82	0.56	0.15	0.23
Decision Tree	0.74	0.31	0.32	0.31
Random Forest	0.81	0.5	0.25	0.33
XGBoost	0.82	0.51	0.28	0.36
CatBoost (Default Loss Function)	0.83	0.58	0.26	0.35
CatBoost (Custom Loss Function)	0.81	0.81	0.77	0.78

Logistic regression showed a high accuracy of 0.82 but had a low recall of 0.15, indicating that it was not as effective in identifying the minority class, which is critical in crash prediction. Decision trees demonstrated an accuracy of 0.74 with a recall of 0.32, suggesting a better performance than logistic regression in detecting the minority class, but at the cost of overall accuracy. Random forest improved upon decision trees' performance, with an accuracy of 0.81 and a recall of 0.25. XGBoost presented similar accuracy to logistic regression, at 0.82, but improved recall, at 0.28, minimally balancing overall accuracy with minority class detection. The CatBoost model with the default loss function showed a slight improvement over XGBoost, with a recall of 0.26 but a higher accuracy of 0.83. Notably, the CatBoost model with a custom loss function demonstrated a significant leap in performance, with a recall of 0.77 and precision of 0.81, while maintaining an accuracy of 0.81. These numbers indicate a robust ability to detect the minority class without sacrificing the accuracy of the model. The recall metric is particularly informative for imbalanced datasets like ours, as it measures the model's ability to identify the positive class (crash occurrences). A high recall is valuable in practical applications, where failing to predict a crash could have serious consequences. Despite a slight compromise on accuracy, the customized CatBoost model's exceptional recall suggests that it is highly adept at detecting true crash events, outweighing the trade-off of a marginally lower accuracy score. Additionally, the F1 score provides a comprehensive measure of the balance between precision and recall.

Despite its high accuracy, logistic regression had a low F1 score of 0.24, reflecting its poor recall. Decision trees had an F1 score of 0.31, slightly improving on logistic regression but still indicating imbalances in performance. Random forest showed an F1 score of 0.33, representing a more balanced performance compared to decision trees. XGBoost, with an F1 score of 0.36, further balanced precision and recall. The CatBoost model with the default loss function had an F1 score of 0.36, indicating a slightly better balance than XGBoost. The CatBoost model with a custom loss function stood out, with a high F1 score of 0.79, underscoring its superior ability to balance precision and recall while effectively identifying crash occurrences.

The receiver operating characteristic (ROC) curve and the corresponding plot of training and validation losses provide a deeper insight into the classification performance and the learning process of our custom CatBoost model. The ROC curve is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. It is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The ROC curve for our custom CatBoost model is shown in Figure 4, where the area under the curve (AUC) is 0.77. This AUC value is a measure of the model's ability to distinguish between the classes and is considered good, indicating a high level of separability. The closer the ROC curve follows the left-hand border and then the top border of the ROC space, the more accurate the test. The curve's significant elevation above the diagonal line of no discrimination suggests the model's strong classification capabilities. Training and validation loss curves are critical for understanding the model's learning over iterations. They provide information on whether the model is fitting well, overfitting, or underfitting. The plot of training and validation losses, depicted in Figure 5, shows a steady decline in both training and validation losses over the iterations, which is a positive indicator of a good model fit. The convergence of the two curves and the absence of a significant gap between them indicate that the model generalizes well to unseen data, avoiding overfitting. The losses decrease, suggesting that the model has learned the patterns within the data to a satisfactory degree, and that further iterations would yield diminishing improvements. These analyses collectively offer a comprehensive evaluation of the model's performance, highlighting its predictive power and learning efficiency. The ROC curve confirms the model's adeptness in classification, while the loss curves attest to the stability and reliability of the model's training process.

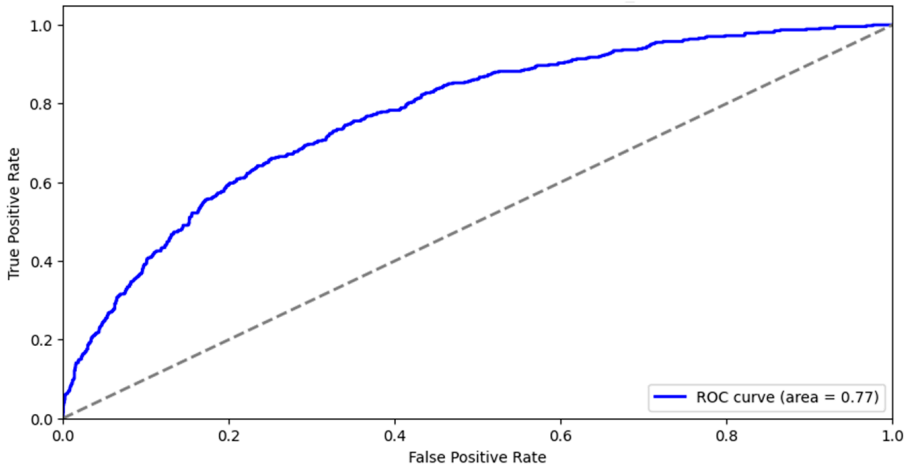

Figure 4. ROC curve for custom CatBoost classification model.

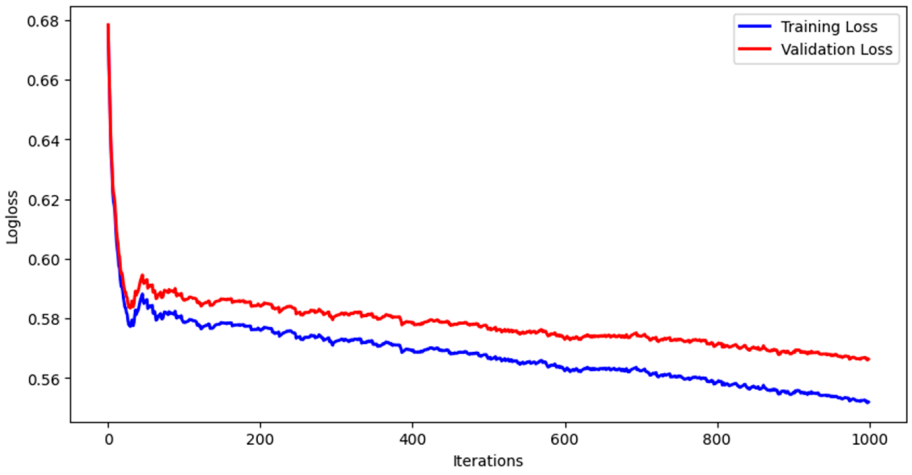

Figure 5. Training and validation losses for custom classification model.

4.3.2. Regression Stage

In the regression stage of our analysis, we specifically sought models that are adept at handling positive count data, as our dataset does not contain zero-crash frequencies at this stage. For this task, we employed two distinct types of models: zero-truncated models developed in R using the PSCL package, which are particularly designed for datasets without zero counts, and the CatBoost models developed in a Python 3.8 environment using the Scikit-learn library. As before, the data were split at an 80:20 ratio to ensure consistent training and testing datasets across the analysis. The results showcased in Table 5 provide a comprehensive look at each model's performance.

Table 5. Confusion matrix.

Model	RMSE	MAE
Zero-Truncated Poisson	4.873	1.92
Zero-Truncated Negative Binomial	2.321	1.282
CatBoost (Default Loss Function)	1.997	1.101
CatBoost (Poisson Loss Function)	1.855	0.916

The zero-truncated Poisson model typically assumes that the mean and variance of the distribution are equal, which may not be a suitable assumption for all datasets, as indicated by its higher RMSE (4.873) and MAE (1.92). These metrics suggest a less accurate fit for our crash frequency prediction. The zero-truncated negative binomial model extends the Poisson model by allowing for variance that exceeds the mean, often a more realistic assumption for real-world count data. This is reflected in its improved performance, with lower RMSE (2.321) and MAE (1.282) values. The CatBoost model with the default loss function demonstrated further improvements in predictive accuracy. Notably, the CatBoost model with the Poisson loss function significantly enhanced the model's predictive accuracy, with the lowest RMSE (1.855) and MAE (0.916). These metrics illustrate the model's precision and its effective handling of the count nature of our crash frequency data. In essence, the CatBoost model with the Poisson loss function not only demonstrates superior performance in predictive accuracy but also exemplifies stability and reliability in its learning process, as evidenced by the behavior of its loss curves over numerous iterations.

The plot of training and validation losses for the CatBoost model with the Poisson loss function, as shown in Figure 6, confirms these findings. The plot reveals a significant initial decrease in training loss, indicating a rapid capture of the underlying distribution in the data. The alignment and stabilization of training and validation losses suggest a model

that is well tuned and generalizes effectively to unseen data, as shown by the narrow gap between the two curves.

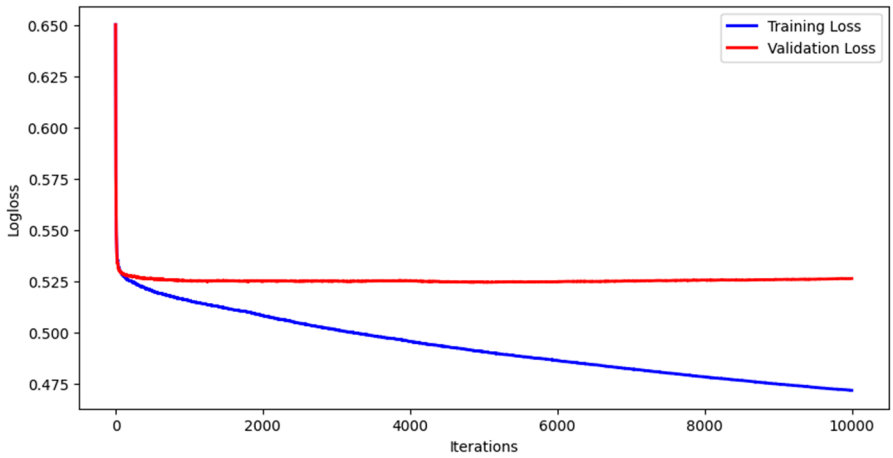

Figure 6. Training and validation losses for custom CatBoost regression model.

4.3.3. Full Model Assessment

The full model evaluation results, as depicted in the Table 6, underscore the efficacy of our proposed two-stage model in accurately predicting crash frequencies. The Poisson hurdle model and the negative binomial hurdle model serve as traditional benchmarks, with their respective RMSE and MAE indicating decent performance levels. The Poisson hurdle model reports an RMSE of 2.097 and an MAE of 0.527, while the negative binomial hurdle model shows an improvement, with an RMSE of 1.336 and an MAE of 0.511. The CatBoost model in its single-stage form further refines these metrics, achieving an RMSE of 1.197 and an MAE of 0.421, suggesting a more precise prediction capability. However, our proposed model outperforms all benchmarks, with the lowest RMSE of 0.978 and MAE of 0.346. These values indicate superior predictive accuracy, highlighting our model's robustness in dealing with the complexities of zero-inflated crash frequency data. The results suggest that the integration of the classification and regression stages enables our model to deliver highly accurate and reliable predictions.

Table 6. Comparative evaluation of full model performance.

Model	RMSE	MAE
Poisson Hurdle Model	2.097	0.527
Negative Binomial Hurdle Model	1.336	0.511
CatBoost (Single Stage)	1.197	0.421
Proposed Model	0.978	0.346

4.3.4. Model Interpretability

The interpretability of predictive models is a crucial component, particularly in road safety analysis, where understanding the underlying factors influencing crash frequency is as important as the accuracy of the predictions themselves. The SHAP summary plots, shown in Figures 7 and 8, bridge the opaque complexity of machine learning models and the transparent rationale required for practical application and policy development. In SHAP summary plots, the horizontal axis denotes the magnitude of each feature's SHAP value in influencing the model output. Features that shift the prediction higher are located towards the right, while those contributing to a lower prediction appear on the left. The color intensity represents the feature value itself; warmer colors indicate higher feature values, while cooler colors indicate lower ones. This visual encoding helps distinguish which features, in their high-value state, act to increase the predicted crash frequency, and which ones in the same state serve to decrease it.

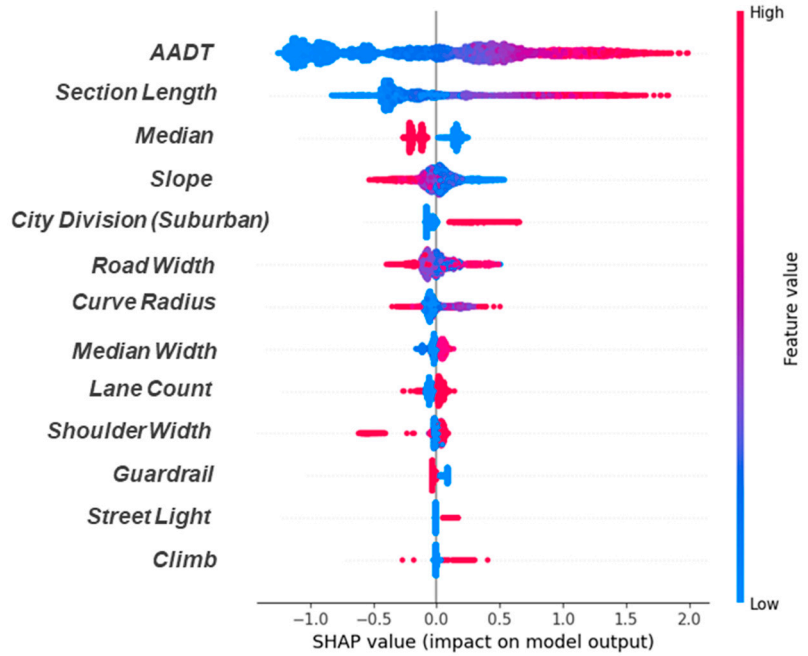

Figure 7. SHAP summary plot for classification stage.

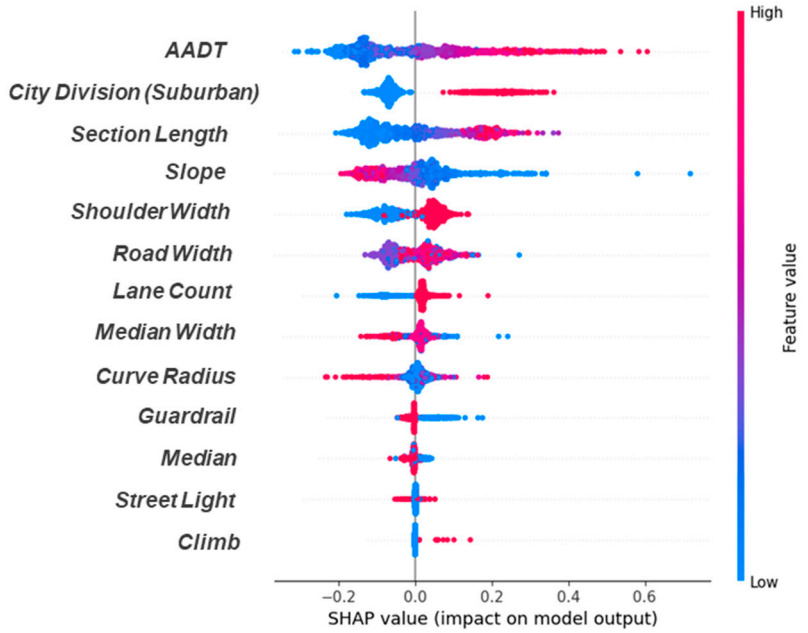

Figure 8. SHAP summary plot for regression stage.

In the binary classification stage, we examined the likelihood of a road segment having any crashes versus none. The SHAP summary plot shown in Figure 7 reveals that the AADT exerts the most significant influence, with higher traffic volumes generally increasing the probability of crashes. This aligns with established traffic safety research that correlates increased traffic density with higher crash potential. Conversely, features like median barriers tend to reduce the likelihood of crashes when present, highlighting their effectiveness as a safety feature. The road segment length also appears to be a key factor. The plot shows that longer road segments are associated with an increased probability of crash occurrence, which could reflect a greater exposure to risk over extended distances. Meanwhile, the slope has a varied impact, suggesting that while steeper slopes may contribute to higher crash risks, the relationship is not uniformly straightforward across all road segments.

Moving to the regression stage, we focused on the number of crashes for road segments where crashes are expected to occur. In the SHAP summary plot shown in Figure 8, AADT continues to be a prominent factor, with higher values pushing the predicted number of crashes up. This suggests that traffic volume affects not only the presence of crashes but also their frequency. The influence of road geometry is further explained here. The slope and shoulder width features, for example, demonstrate a mix of positive and negative effects on crash frequency. A steep slope may increase the risk, but a wide shoulder could potentially mitigate it by allowing drivers more room for error.

Finally, we report the SHAP partial dependence plots (PDPs) for the regression model. Interpreting SHAP PDP plots involves understanding the relationship between the target variable and one or two predictor variables, while keeping all other features constant. These plots help to understand the marginal effect of each feature across its entire range, irrespective of the presence or distribution of other features in the data. SHAP PDP plots of selected features are shown in Figure 9. The additional variable on the right side of each PDP plot indicates the color scale used to differentiate data points based on another feature's value. This color scale helps visualize how the interaction between the primary feature on the x-axis and this secondary feature can impact the prediction results. The gradient or color intensity represents the range or distribution of values for the secondary feature, allowing for a multi-dimensional analysis within the same plot. This study only considers the main feature of each PDP in our interpretation.

The AADT plot shows a clear positive correlation. As the volume increases, so does the impact on crash frequency, as depicted by rising SHAP values. This is expected, as more traffic typically increases the likelihood of accidents. The PDP for suburban roads indicates a distinction in crash frequency when compared to rural roads. There is a clear jump in SHAP values for suburban areas, which could be due to higher traffic volumes, intersections, and varying road-user behavior in suburban settings compared to rural areas. The PDP for road segment length indicates a non-linear relationship with the crash frequency. Initially, as the length increases, the SHAP value increases, suggesting a higher crash frequency. However, beyond a certain point, the SHAP values decrease, implying that longer road segments may not always correlate with higher crash frequencies. The presence of high SHAP values for shorter lengths may be associated with urban areas where intersections and traffic are more frequent, potentially leading to more accidents. Wider shoulders on roads show a mix of positive and negative SHAP values, which suggests that there is not a straightforward relationship between shoulder width and crash frequency. It is possible that very wide shoulders might sometimes be misinterpreted by drivers as an extra lane or may encourage stopping, leading to accidents, while in other contexts they provide a safety margin. The introduction of a climbing lane generally shows a positive SHAP value, indicating an increased crash frequency. This might reflect more complex road geometries or changing driver behaviors such as overtaking in these areas. The presence of a median barrier seems to have a mixed impact on the crash frequency. Some points indicate a decrease in crash frequency (negative SHAP values), possibly due to the barrier preventing crossover accidents, while others show an increase (positive SHAP values), which could be due to collisions with the barrier itself or other complexities that are not captured by this variable alone. The effect of street lighting on crash frequency is not clear-cut. While one might expect improved visibility to reduce crashes, the PDP suggests a mixed relationship, where the effect may vary based on other factors such as road type, traffic volume, or driver complacency in well-lit areas. The PDP plot for lane width indicates that varying lane widths have a diverse impact on crash frequency, with no clear trend. This implies that lane width alone is not a consistent predictor of crash frequency and must be considered in conjunction with other road features.

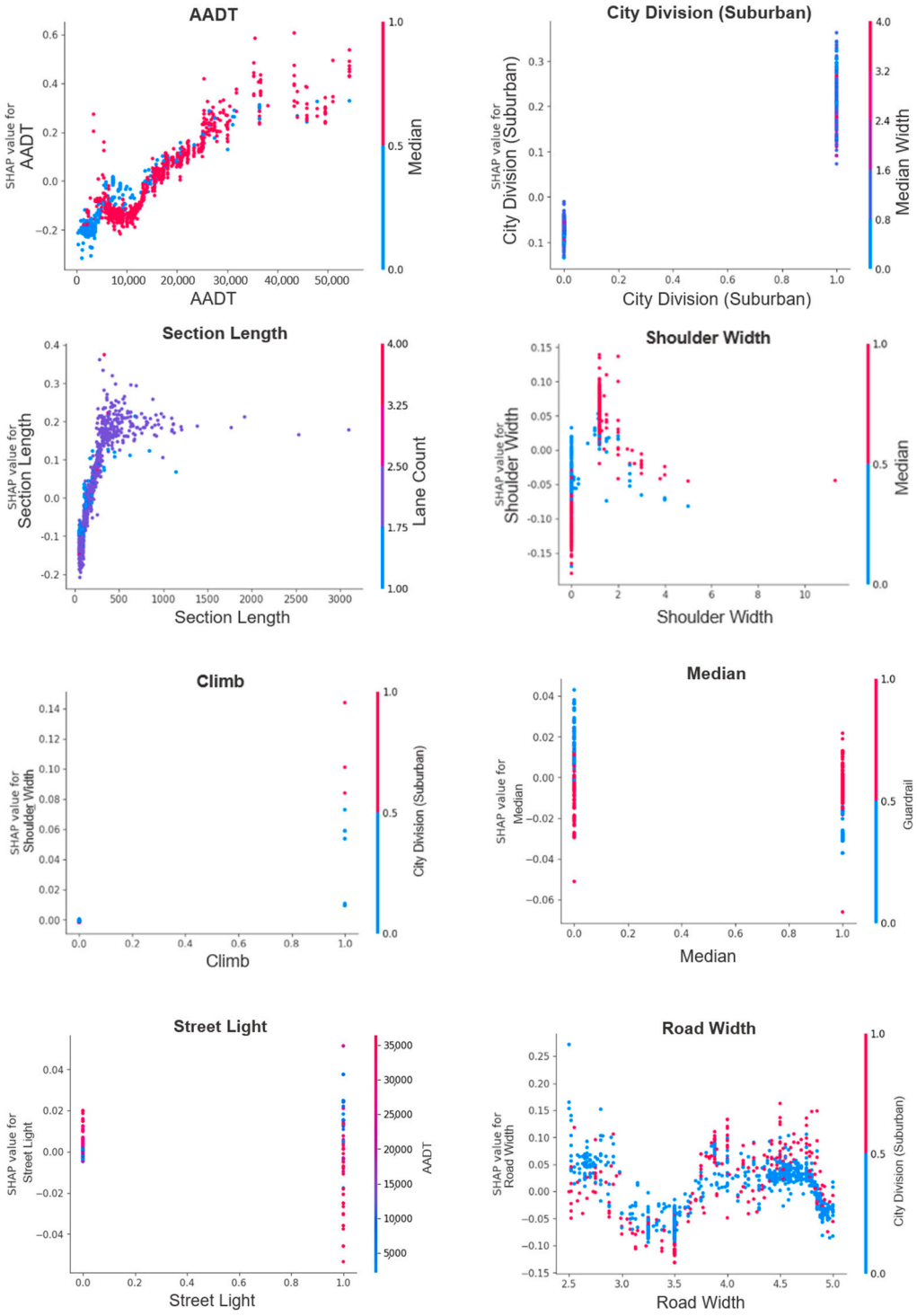

Figure 9. SHAP PDP plots for selected variables.

5. Discussion

Crash frequency modeling has long been recognized as a complex challenge, particularly when dealing with zero-inflated datasets where a significant proportion of road segments report no crashes. Historically, traditional statistical models, such as the Poisson and negative binomial models, have been the go-to methods for handling crash frequency data. However, these models often struggle with overdispersion and an excess of zeros, leading to less accurate predictions and difficulties in identifying the underlying risk factors. In response to these limitations, data-driven approaches based on machine learning

have emerged as powerful alternatives. Machine learning models can capture complex, non-linear relationships in data, offering potential improvements in prediction accuracy and interpretability.

In this study, we proposed a novel machine learning-based hurdle model to analyze crash frequency data, addressing the limitations of traditional models. The framework consists of two stages: a classification stage to predict the occurrence likelihood of crashes, and a regression stage to estimate the frequency of crashes for segments where crashes occur. The results from the classification stage indicate that the CatBoost model, particularly when enhanced with custom loss functions, significantly outperformed traditional models. The precision and recall scores were particularly noteworthy, demonstrating the model's capability to handle imbalanced data and accurately predict the likelihood of crashes. This stage demonstrated a clear advantage in handling zero-inflated data while ensuring high sensitivity to crash occurrences, an essential feature for effective road safety interventions. In the regression stage, where the task was to predict the number of crashes for segments where they occurred, the CatBoost model again outperformed traditional count-based models like the zero-truncated Poisson and negative binomial models. By utilizing a Poisson loss function, the CatBoost model more effectively captured the count nature of crash data, leading to lower RMSE and MAE values. These results underscore the flexibility of the proposed machine learning framework in adapting to the unique challenges posed by crash frequency data, particularly in handling overdispersion and non-linear relationships between predictors.

The full model evaluation further solidified the superiority of the proposed two-stage approach. When compared to benchmark models, such as the Poisson hurdle and negative binomial hurdle models, the proposed model achieved the lowest RMSE and MAE values. This demonstrates its enhanced accuracy and ability to generalize well to different road segments, providing a reliable tool for road safety analysis. To provide further insights into the model's decision-making process, SHapley Additive exPlanations (SHAP) was employed. SHAP helped to interpret the model outputs by quantifying the contribution of individual variables to the predictions. While SHAP does not provide numerical coefficients like traditional statistical models, it offers a more flexible interpretation of variables' importance. Features such as road curvature, traffic volume (AADT), and segment length were identified as critical drivers of crash risk, which is consistent with the established road safety literature. This interpretability is crucial for practitioners, as it enables them to make informed decisions about which road segments require safety interventions. Although SHAP does not generate numerical metrics comparable to traditional regression coefficients, it offers significant advantages in explaining complex machine learning models. The insights gained from SHAP can be used to develop targeted safety interventions, as it highlights the most influential risk factors across various road segments.

Despite these improvements, certain limitations must be acknowledged. Firstly, the model's performance is tied closely to the dataset derived from South Korean roads, and this may impact its generalizability to other regions with different road conditions. Furthermore, the segmentation of road sections, with lengths as short as 50 m, may contribute to a high incidence of zero crashes. Future research could explore the use of longer or more homogeneously defined segments to further validate the model's generalizability and assess its predictive performance across varied road segment lengths.

Additionally, this study did not incorporate extensive validation techniques, such as robust training/testing splits, which could enhance the model's robustness and applicability to unseen data. We suggest that future studies should apply advanced validation methodologies, including cross-validation and separate training and test datasets, to strengthen the model's reliability and reduce the risk of overfitting. Moreover, the current model does not account for temporal dynamics, such as changing traffic patterns, infrastructure modifications, or evolving traffic regulations, all of which can influence crash risks over time. Future iterations of the model could integrate temporal variables to capture these evolving risk factors, enhancing the model's long-term predictive relevance.

Expanding the data sources to include real-time traffic data, weather conditions, and socio-economic factors could provide a more comprehensive analysis of factors influencing crash occurrences. Finally, to improve the model's interpretability, advanced techniques such as Individual Conditional Expectation (ICE) and Accumulated Local Effects (ALE) plots could complement SHAP values, offering deeper insights into predictor–variable relationships. Addressing potential biases in underlying crash data, such as underreporting, is also essential to ensure fairness and accuracy in model predictions.

6. Conclusions

In addressing the critical challenges faced by traditional statistical models in traffic safety analysis, particularly their limitations with zero-inflated and overdispersed data, this study proposes a novel data-driven approach through the development of a machine learning-based hurdle model. This initiative was grounded in the need to enhance the predictive accuracy and interpretability of crash frequency analyses, critical components for the formulation of effective road safety policies and interventions. The motivation for this research originated from a comprehensive review of existing methodologies in traffic safety analysis, which highlighted significant gaps in the ability of traditional models to accurately capture the complexities of crash data. Recognizing the potential of machine learning to address these gaps, we embarked on a journey to develop, validate, and test a new model capable of offering deeper insights into crash risk factors and their interactions.

The machine learning-based hurdle model developed in this study represents an important methodological advancement in the field of traffic safety analysis. It is characterized by its dual-stage process, wherein the first stage addresses the probability of zero versus non-zero outcomes (zero inflation) and the second stage models the frequency of crashes given that they occur. This approach allows for a more nuanced analysis of crash data, overcoming the limitations inherent in traditional Poisson and negative binomial models. Despite the complexity often associated with machine learning models (black-box models), efforts were made to enhance the interpretability of the hurdle model, integrating SHAP values to elucidate the influence of various predictors on crash frequency outcomes. The proposed methodology encompassed not only the model's theoretical formulation but also its empirical application to real-world crash data from Korean roads. These steps were critical in demonstrating the model's effectiveness and robustness, affirming its potential utility in traffic safety research and policymaking.

The practical implications of the findings are significant. The enhanced predictive accuracy and interpretability of the proposed model could lead to better-informed decision-making in traffic safety. Crash frequency modeling is useful for network screening, Safety Performance Functions (SPFs), Crash Modification Factors (CMFs), and other applications to identify high-risk locations and evaluate the effectiveness of road safety interventions. Policymakers and traffic safety professionals can use these insights to design more effective interventions, potentially reducing the frequency and severity of road traffic accidents. The framework's adaptability also means that it can theoretically be applied in various geographical contexts, making it a valuable tool for global traffic safety improvements, since it is a data-driven approach. Additionally, the ability to interpret model outputs through SHAP values allows for more transparent and justifiable policy decisions, which can improve public trust and the effectiveness of safety measures.

Despite its strengths, this study has several limitations. The model's performance is tied to data quality, which may limit its generalizability beyond South Korean roads. Short road segments in the dataset may have contributed to a high incidence of zero crashes, suggesting that future research could explore longer segments to improve accuracy. Additionally, the validation process lacked extensive training/testing splits, impacting its robustness. Future work should incorporate rigorous validation techniques and consider temporal dynamics, such as evolving traffic patterns and regulations, to enhance the model's long-term relevance. Data collection biases, like underreporting, also pose challenges; thus, improving data practices is essential for fairer predictions. Further in-

terpretability methods, such as ICE and ALE plots, could deepen insights into variables' impacts on crash risks. Expanding the model's application to diverse regions and exploring advanced machine learning techniques, including ensemble methods and deep learning, could enhance its adaptability, accuracy, and stability, ultimately supporting better road safety strategies across different contexts.

Author Contributions: Conceptualization, M.B.B.K.; methodology, M.B.B.K. and D.Y.; formal analysis, M.B.B.K.; investigation, M.B.B.K. and D.Y.; supervision, D.Y.; writing—original draft preparation, M.B.B.K.; writing—review and editing, M.B.B.K. and D.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by a Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure, and Transport (Grant 21AMDP-C160881-01, Future Road Design and Testing for Connected and Autonomous Vehicles).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. World Health Organization. Road Traffic Injuries. Available online: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries (accessed on 3 August 2024).
2. Centers for Disease Control and Prevention. Road Traffic Accidents. Available online: https://www.cdc.gov/transportation-safety/global/index.html (accessed on 15 November 2024).
3. Centers for Disease Control and Prevention. Financial Impact of Road Traffic Crashes. Available online: https://www.cdc.gov/transportation-safety/global/publications.html (accessed on 15 November 2024).
4. Lord, D.; Mannering, F. The Statistical Analysis of Crash-Frequency Data: A Review and Assessment of Methodological Alternatives. *Transp. Res. Part A Policy Pract.* **2010**, *44*, 291–305. [CrossRef]
5. Mannering, F.; Bhat, C. Analytic Methods in Accident Research: Methodological Frontier and Future Directions. *Anal. Methods Accid. Res.* **2014**, *1*, 1–22.
6. American Association of State Highway and Transportation Officials (AASHTO). *Highway Safety Manual*, 1st ed.; AASHTO: Washington, DC, USA, 2010; ISBN 978-1-56051-477-0.
7. Lambert, D. Zero-Inflated Poisson Regression, with an Application to Defects in Manufacturing. *Technometrics* **1992**, *34*, 1–14. [CrossRef]
8. Geedipally, S.R.; Lord, D.; Dhavala, S.S. The Negative Binomial-Lindley Generalized Linear Model: Characteristics and Application using Crash Data. *Accid. Anal. Prev.* **2012**, *45*, 258–265. [CrossRef]
9. Shankar, V.; Mannering, F.; Barfield, W. Effect of roadway geometrics and environmental factors on rural freeway accident frequencies. *Accid. Anal. Prev.* **1995**, *27*, 371–389. [CrossRef] [PubMed]
10. Lord, D.; Washington, S.P.; Ivan, J.N. Poisson, Poisson-gamma and zero inflated regression models of motor vehicle crashes: Balancing statistical fit and theory. *Accid. Anal. Prev.* **2005**, *37*, 35–46. [CrossRef]
11. Son, J.; Sayed, T.; Chung, Y. Modeling the relationship between truck accidents and geometric design of road sections: Poisson versus negative binomial regressions. *Accid. Anal. Prev.* **2011**, *43*, 673–682.
12. Hosseinpour, M.; Yahaya, A.S.; Sadullah, A.F.; Ghadiri, S.M.R. A comparative study of count models: Application to pedestrian-vehicle crashes along Malaysia federal roads. *Traffic Inj. Prev.* **2013**, *14*, 630–638. [CrossRef]
13. Cai, Q.; Lee, J.; Eluru, N.; Abdel-Aty, M. Macro-level pedestrian and bicycle crash analysis: Incorporating spatial spillover effects in dual state count models. *Accid. Anal. Prev.* **2016**, *93*, 14–22. [CrossRef]
14. Khedher, M.B.B.; Yun, D. Generalized linear models to identify the impact of road geometric design features on crash frequency in rural roads. *KSCE J. Civ. Eng.* **2022**, *26*, 1388–1395. [CrossRef]
15. Lord, D.; Persaud, B. Accident prediction models with and without trend: Application of the generalized estimating equations procedure. *Transp. Res. Rec. J. Transp. Res. Board* **2000**, *1717*, 102–108. [CrossRef]
16. Miaou, S.-P.; Song, J.J. Bayesian ranking of sites for engineering safety improvements: Decision parameter, treatability concept, statistical criterion, and spatial dependence. *Accid. Anal. Prev.* **2005**, *37*, 699–720. [CrossRef] [PubMed]
17. Geedipally, S.R.; Lord, D. Examination of crash variances estimated by Poisson-gamma and Conway-Maxwell-Poisson models. *Transp. Res. Rec.* **2011**, *2241*, 59–67. [CrossRef]
18. Park, E.S.; Lord, D. Application of finite mixture models for vehicle crash data analysis. *Accid. Anal. Prev.* **2009**, *41*, 683–691. [CrossRef] [PubMed]

19. Wen, X.; Xie, Y.; Jiang, L.; Pu, Z.; Ge, T. Applications of machine learning methods in traffic crash severity modelling: Current status and future directions. *Transp. Rev.* **2021**, *41*, 855–879. [CrossRef]
20. Tang, J.; Liang, J.; Han, C.; Li, Z.; Huang, H. Crash injury severity analysis using a two-layer stacking framework. *Accid. Anal. Prev.* **2019**, *122*, 226–238. [CrossRef]
21. Xie, Y.; Lord, D.; Zhang, Y. Predicting motor vehicle collisions using Bayesian neural network models: An empirical analysis. *Accid. Anal. Prev.* **2007**, *39*, 922–933. [CrossRef]
22. Li, X.; Lord, D.; Zhang, Y.; Xie, Y. Predicting motor vehicle crashes using support vector machine models. *Accid. Anal. Prev.* **2008**, *40*, 1611–1618. [CrossRef]
23. Abdel-Aty, M.; Haleem, K. Analyzing angle crashes at unsignalized intersections using machine learning techniques. *Accid. Anal. Prev.* **2011**, *43*, 461–470. [CrossRef]
24. Haleem, K.; Gan, A.; Lu, J. Using multivariate adaptive regression splines (MARS) to develop crash modification factors for urban freeway interchange influence areas. *Accid. Anal. Prev.* **2013**, *55*, 12–21. [CrossRef]
25. Zeng, Q.; Huang, H.; Pei, X.; Wong, S.C.; Gao, M. Rule extraction from an optimized neural network for traffic crash frequency modeling. *Accid. Anal. Prev.* **2016**, *97*, 87–95. [CrossRef]
26. Zhang, X.; Waller, S.T.; Jiang, P. An ensemble machine learning-based modeling framework for analysis of traffic crash frequency. *Comput.-Aided Civ. Infrastruct. Eng.* **2020**, *35*, 258–276. [CrossRef]
27. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A highly efficient gradient boosting decision tree. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 3146–3154.
28. Al Mamlook, R.E.; Abdulhameed, T.Z.; Hasan, R.; Al-Shaikhli, H.I.; Mohammed, I.; Tabatabai, S. Utilizing machine learning models to predict the car crash injury severity among elderly drivers. In Proceedings of the 2020 IEEE international conference on electro information technology (EIT), Chicago, IL, USA, 31 July–1 August 2020; pp. 105–111.
29. Mullahy, J. Specification and testing of some modified count data models. *J. Econom.* **1986**, *33*, 341–365. [CrossRef]
30. Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A.V.; Gulin, A. CatBoost: Unbiased Boosting with Categorical Features. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 6639–6649.
31. Hancock, J.T.; Khoshgoftaar, T.M. CatBoost for Big Data: An Interdisciplinary Review. *J. Big Data* **2020**, *7*, 94. [CrossRef]
32. Park, J.H.; Yun, D.G.; Seong, J.G.; Lee, J.S. Introduce Advanced Road Research Vehicle-'ARASEO'. *Transp. Technol. Policy* **2012**, *9*, 47–52.