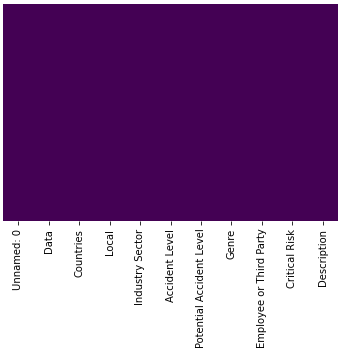
INTERIM REPORT

This interim report summarizes the 1st 3 deliverables mentioned in the project document.

1. Summary of problem statement, data and findings. Every good abstract describes briefly what was intended at the outset, and summarizes findings and implications.

1. Summary of the Approach to EDA and Pre-processing. Include any insightful visualization you have teased out of the data. If you’ve identified particularly meaningful features, interactions or summary data, share them and explain what you noticed. Visual displays are powerful when used well, so think carefully about what information the display conveys.

**Null values check:**

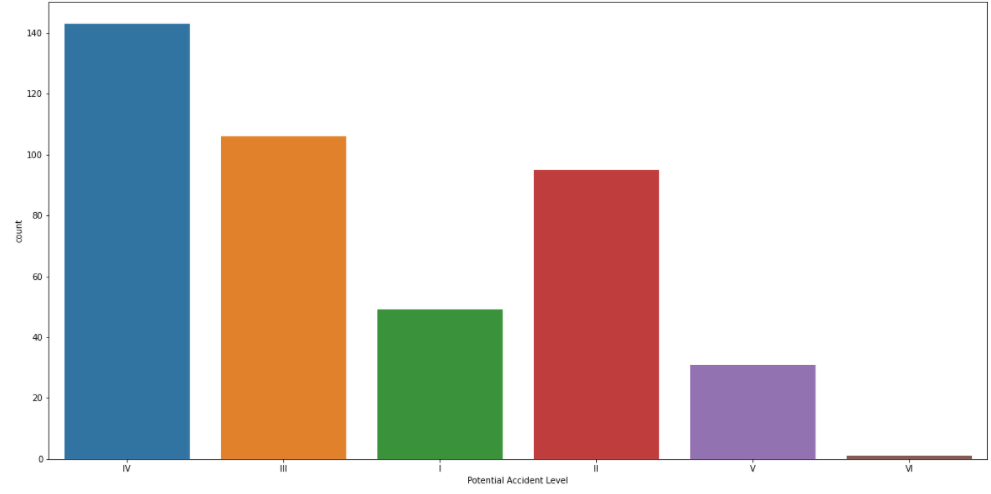


**Observation:-**

* From HeatMap we conclude that there are no null or Nan values provided in data.

**Univariate Analysis:**

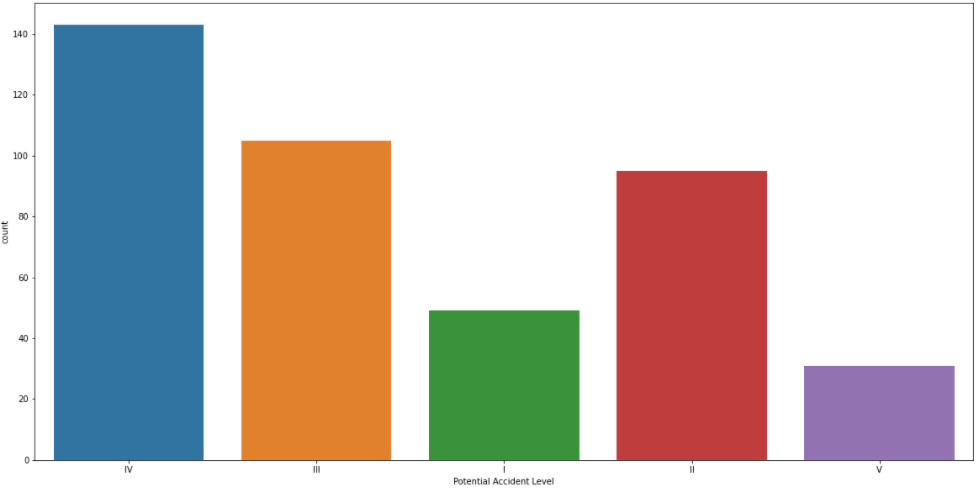
Potential Accident Level column:



**Observation:-**

* The dataset provided here is imbalanced.
* Potential Accident Level IV has highest number of records i.e. 143.
* Potential Accident Level III has second highest number of records i.e. 106.
* Potential Accident Level II has third highest number of records i.e. 95.
* Potential Accident Level VI has lowest number of records i.e. 1.

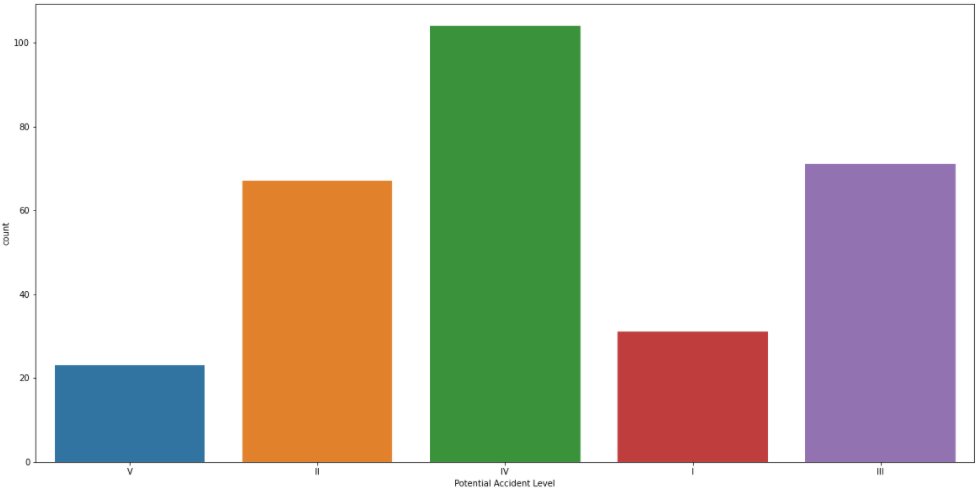
Potential Accident Level column after deleting level VI:



**Observation:-**

* Level VI has only 1 record in dataset hence deleted.
* Number of records for rest of the levels remains same.

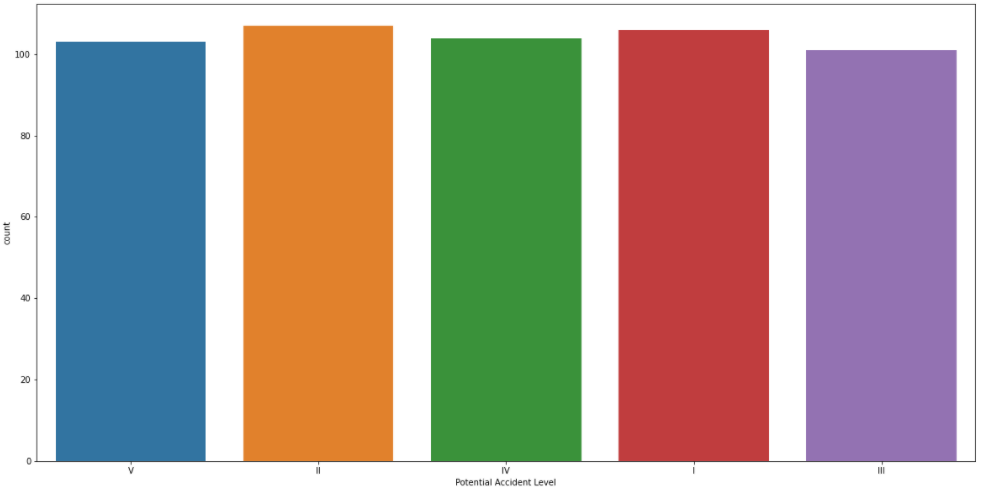
Potential Accident Level column after train-test split:



**Observation:-**

* Above bar plot is of train data after 70%-30% train-test split

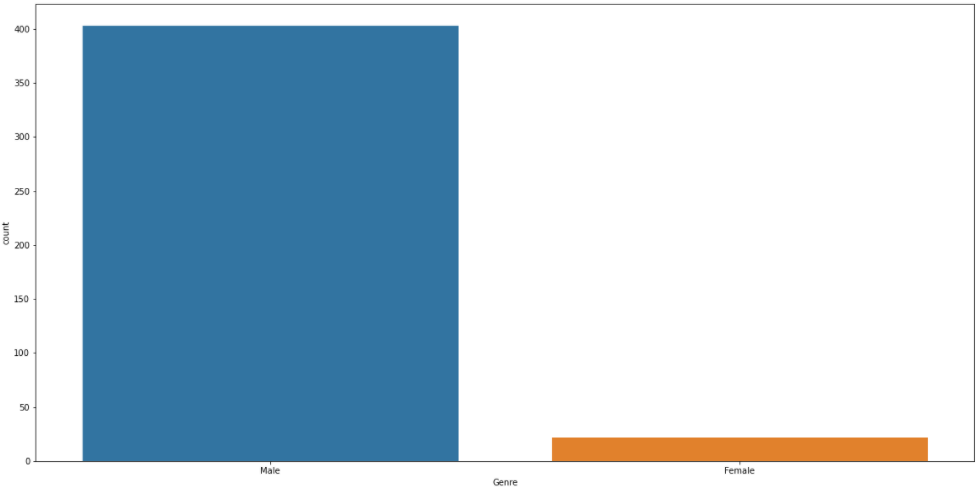
Potential Accident Level column after data augmentation:



**Observation:-**

* Before data augmentation the data was highly imbalance.
* Now the data is somewhat balanced.

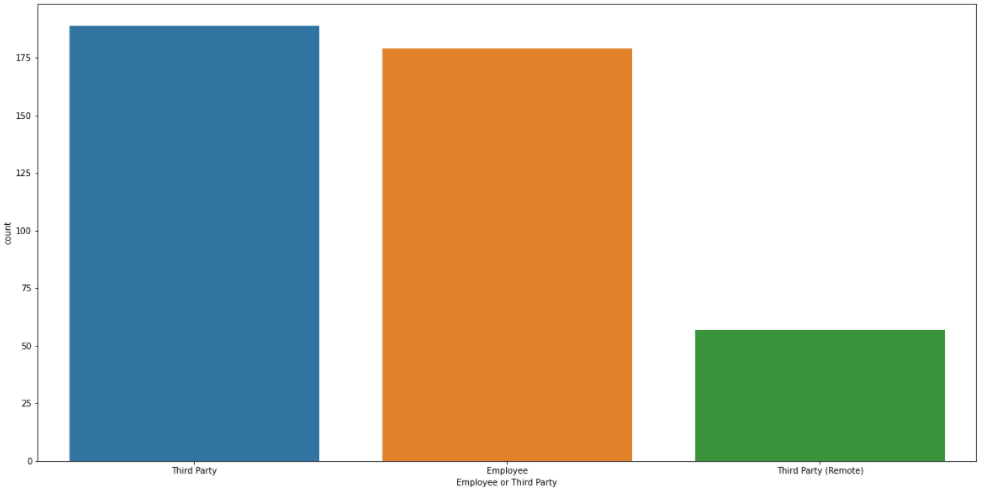
Genre column:



**Observation:-**

* Under Genre there are 2 categories - Male & Female.
* In dataset, Male population is higher than Females.
* Male population is 403 which is 94.82% in that category.
* Female population is 22 which is 5.18% in that category.

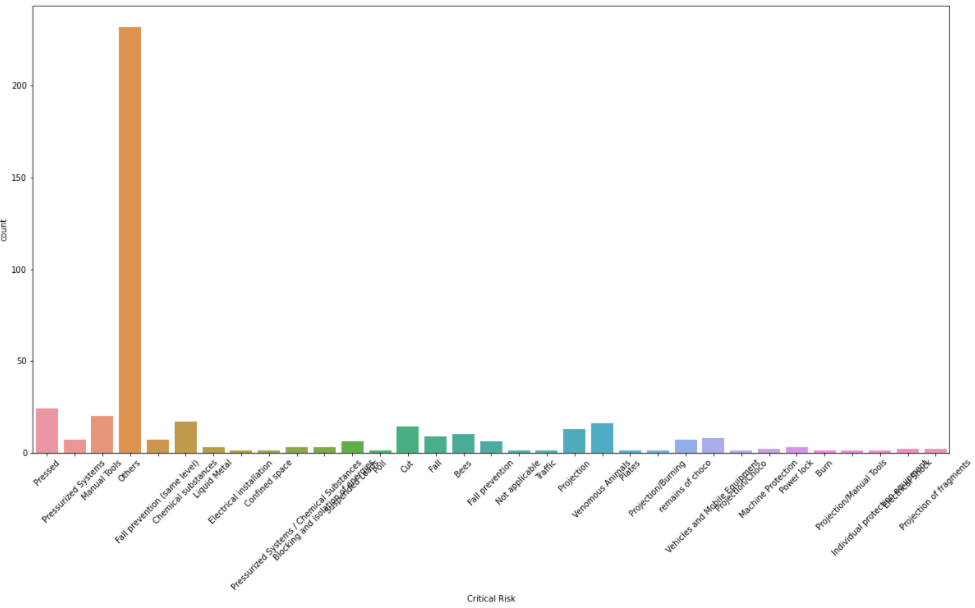
Employee or Third Party column:



**Observation:-**

* Under Employee or Third Party column there are 3 categories - Third Party, Employee & Third Party (Remote).
* In dataset, Third Party category is highest - 189 which is 44.47% in that category.
* Employee category is second highest - 179 which is 42.12% in that category.
* Third Party (Remote) category is third highest - 57 which is 13.41% in that category.

Critical Risk column:



**Observation:-**

* Under Critical Risk column there are 33 categories -
  + Pressed.
  + Manual Tools
  + Chemical substances
  + Venomous Animals
  + Cut
  + Projection
  + Bees
  + Fall
  + Vehicles and Mobile Equipment
  + Fall prevention (same level)
  + remains of choco
  + Pressurized Systems
  + Fall prevention
  + Suspended Loads
  + Blocking and isolation of energies
  + Pressurized Systems / Chemical Substances
  + Power lock
  + Liquid Metal, Projection of fragments
  + Machine Protection
  + Electrical Shock
  + Individual protection equipment
  + Projection/Manual Tools
  + Burn
  + Poll
  + Projection/Choco
  + Projection/Burning
  + Plates
  + Confined space
  + Traffic
  + Not applicable
  + Electrical installation
  + Others
* In dataset, “Others” category is highest - 232 which is 54.59% in that category.
* Pressed is second highest - 24 which is 5.65% in that category.
* Manual Tools category is third highest - 20 which is 4.71% in that category.

Statistical evaluation on Description column:

Count 425.000000

Mean 368.280000

STD 178.944426

Min 94.000000

25% 227.000000

50% 335.000000

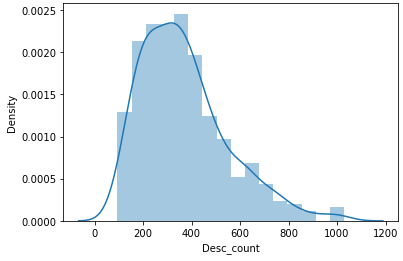
75% 457.000000

Max 1029.000000

**Observation:-**

* In dataset we have Description column, a new column called Desc\_count has been added based on Description column.
* Desc\_count contains total character count under Description for each record.
* We have used describe function on this Desc\_count column which shows various statistics.
  + There total 425 records.
  + The mean of Desc\_count column is 368.28
  + Standard deviation is 178.94 for Desc\_count column.
  + The minimum description length is 94 characters.
  + The maximum description length is 1029 characters.

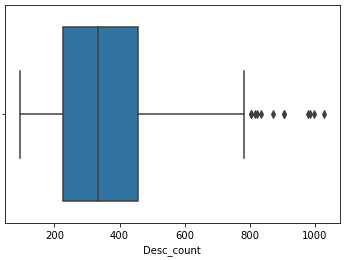
Desc\_count column:



**Observation:-**

* Above is distribution plot or Gaussian curve for Desc\_count.
* The curve is not exactly bell shaped & is skewed.
* The curve is more skewed on right side whereas on left side its steep.

Desc\_count column:

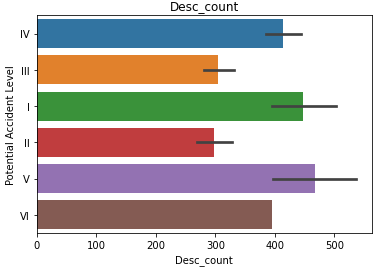


**Observation:-**

* Desc\_count column has outliers near max whisker.
* There are more number of data between 75% & Max whisker as compared to 25% & min whisker.
* Also there are more number of data between 50% & 75% points.

**Bivariate Analysis:**

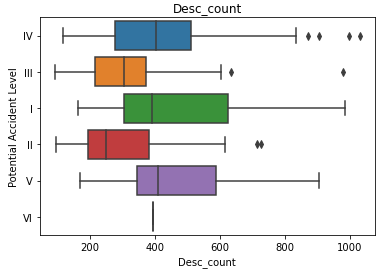
Potential Accident Level column versus Desc\_count column:



**Observation:-**

* Above is a bar plot between Potential Accident Level column & Desc\_count column.
* Potential Accident Level V has highest number of Desc\_count value.
* Potential Accident Level I has 2nd highest number of Desc\_count value.
* Potential Accident Level IV has 3rd highest number of Desc\_count value.

Potential Accident Level column versus Desc\_count column:



**Observation:-**

* Above is a box plot between Potential Accident Level column & Desc\_count column.
* Potential Accident Level IV has highest number - 4 of outliers near max whisker.
* Potential Accident Level III & II has 2 outliers near max whisker.
* None of the box plot has outliers near min whisker.
* All Potential Accident Levels has more number of data between 75% & Max whisker as compared to 25% & Min whisker.
* For Potential Accident Level IV & III there are more number of data between 50% & 75% points.
* For Potential Accident Level I, II & V there are more number of data between 25% & 50% points.
* For Potential Accident Level VI there is only 1 record hence box plot is not visible.

1. Deciding Models and Model Building. Based on the nature of the problem, decide what algorithms will be suitable and why? Experiment with different algorithms and get the performance of each algorithm.

The problem would be solved by implementing Machine Learning (ML) models & Deep Learning (DL) models. Following algorithms will be implemented under each model.

**Machine Learning (ML) models:**

Logistic Regression algorithm:

* Logistic regression is Machine Learning algorithms which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.
* Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.
* Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.
* In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
* The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
* Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.
* Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification.

Naive Bayes Classifier algorithm:

* Naive Bayes algorithm is a supervised learning algorithm which is based on Bayes theorem and used for solving classification problems.
* It is mainly used in text classification that includes a high-dimensional training dataset.
* Naive Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.
* It is a probabilistic classifier which means it predict on the basis of the probability of an object.
* Some popular examples of Naive Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

Random Forest algorithm:

* Random Forest is machine learning algorithm that comes under supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.
* Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.
* Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.
* The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Support Vector Machine (SVM) algorithm:

* Support Vector Machine or SVM is Supervised Learning algorithms, which is used for Classification as well as Regression problems. It is primarily used for Classification problems in Machine Learning.
* The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.
* SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

K-Nearest Neighbor (KNN) algorithm:

* K-Nearest Neighbor is Machine Learning algorithms based on Supervised Learning technique.
* K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
* K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K-NN algorithm.
* K-NN algorithm can be used for Regression as well as for Classification but it is mostly used for the Classification problems.
* K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
* It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
* KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

**Deep Learning (DL) models:**

Long Short-Term Memory (LSTM) algorithm:

* Long short-term memory (LSTM) is an artificial recurrent neural network (RNN) architecture used in the field of deep learning.
* Unlike standard feed forward neural networks, LSTM has feedback connections. It can process not only single data points (such as images), but also entire sequences of data (such as speech or video).
* A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.
* LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series.
* LSTMs were developed to deal with the vanishing gradient problem that can be encountered when training traditional RNNs.
* Relative insensitivity to gap length is an advantage of LSTM over RNNs.

Bidirectional Long Short-Term Memory (Bi-LSTM) algorithm:

* Bidirectional long-short term (Bi-LSTM) memory is the process of making any neural network have the sequence information in both directions backwards (future to past) or forward (past to future).
* In bidirectional, our input flows in two directions, making a Bi-LSTM different from the regular LSTM.
* With the regular LSTM, we can make input flow in one direction, either backwards or forward. However, in bi-directional, we can make the input flow in both directions to preserve the future and the past information.

Gated Recurrent Units (GRU) algorithm:

* To solve the Vanishing-Exploding gradients problem often encountered during the operation of a basic Recurrent Neural Network, many variations were developed. One of the most famous variations is the Long Short Term Memory Network (LSTM). One of the lesser-known but equally effective variations is the Gated Recurrent Unit Network (GRU).
* GRU consists of only three gates and does not maintain an Internal Cell State. The information which is stored in the Internal Cell State in an LSTM recurrent unit is incorporated into the hidden state of the Gated Recurrent Unit. This collective information is passed onto the next Gated Recurrent Unit. The different gates of a GRU are as described below:-
* Update Gate: It determines how much of the past knowledge needs to be passed along into the future. It is analogous to the Output Gate in an LSTM recurrent unit.
* Reset Gate: It determines how much of the past knowledge to forget. It is analogous to the combination of the Input Gate and the Forget Gate in an LSTM recurrent unit.
* Current Memory Gate: It is often overlooked during a typical discussion on Gated Recurrent Unit Network. It is incorporated into the Reset Gate just like the Input Modulation Gate is a sub-part of the Input Gate and is used to introduce some non-linearity into the input and to also make the input Zero-mean. Another reason to make it a sub-part of the Reset gate is to reduce the effect that previous information has on the current information that is being passed into the future.

Bidirectional Encoder Representations from Transformers (BERT) algorithm:

1. How to improve your model performance? What are the approaches you can take to improve your model? Can you do some feature selection, data manipulation and model improvements?