



# PREDICTION ON SEVERITY OF ACCIDENT

Cousera Capstone Project

By Chenyang Liu

# INTRODUCTION

## Background

Driving is one of the most common part of life for many people. An accurate prediction on the severity of an accident would help drivers to make a decision. For a GPS service providers, such as Google Map, they can inform the driver and to choose a suitable route for the driver based on the estimation of the severity.

## Business Problem

Based on the information we have, whether the accident is an injury or a property damage?



The background features abstract blue geometric shapes, including circles, triangles, and irregular polygons, scattered across the white canvas. A large, light blue parallelogram is positioned behind the main text.

PART 1

# Methodology

# Methodology

## Classification Models

### - Logistic Regression

A statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist.

### - K-Nearest Neighbors

A type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until function evaluation.

### - Decision Tree

A decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements.

## Sampling Methods

### - Upsampling

To keep as much information as we can, we will use upsampling so that we won't lose any information.

### - Downsampling

Considering the computability of the computer for some machine learning models, we will also use downsampling to reduce the dimensions and the number of records.

## Cross Validation

### - 3-fold Cross Validation

To evaluate the result, 3-fold cross validation will be used to give a relatively fair result.

## Key Metric

### - F1-score

Since predicting an accurate result for both categories is important for us, F1-score will be chosen as the key metric to evaluate the performance of each model.

## Principal Component Analysis

Components of a collection of points in a real  $p$ -space are a sequence of  $p$  direction vectors where the  $i$ th vector is the direction of a line that best fits the data.

## Grid Search

Exhaustively search through a manually specified subset of the hyperparameter space.

The background features abstract blue geometric shapes, including circles, triangles, and irregular polygons, scattered across the white canvas. A large, light blue trapezoidal shape is positioned behind the main text.

PART 2

# Data Exploration and Data Cleaning



# Data Exploration and Data Cleaning

Originally, this data set includes **194673** records and 38 variables, which contains categorical variables, such as ADDRTYOE, and numerical variables, such as ROADCOND. There are some problems in this data set which need to be fixed.

These problems are: missing values, duplicate variables in different formats, conflict formats in the same variable and imbalanced data. Besides, to apply the data to the models, other steps like data transformation and resampling methods are also necessary.

## Missing Values

INTKEY, EXCEPTRSNCODE, EXCEPTRSNDESC, INATTENTIONIND, PEDROWNOTGRNT, PEEDING, and SDOTCOLNUM have a large number of missing values.

## Duplicate Variables in Different Formats

SEVERITYCODE & SEVERITYDESC, INCDATE & INCDTTM, SDOT\_COLCODE & SDOT\_COLDESC, and ST\_COLCODE & ST\_COLDESC show the same values but in different formats.

## Conflict Formats in the Same Variable

The values of UNDERINFL used two formats: 1/0 and Y/N. According to the understanding of data, these two formats are indicating the same values.

## Imbalanced Data

There are 132630 records with severity code as 1 while there are only 57159 records with severity code as 2, about 30% of the total number.

# Removing Missing Values and Variables with a Large Number of Missing Value

```
df.isnull().sum()
```

```
] SEVERITYCODE      0
  X                5334
  Y                5334
  OBJECTID          0
  INCKEY            0
  COLDETKEY         0
  REPORTNO          0
  STATUS            0
  ADDRTYPE         1926
  INTKEY           129603
  LOCATION          2677
  EXCEPTRSNCODE     109862
  EXCEPTRSNDESC     189035
  SEVERITYCODE.1     0
  SEVERITYDESC       0
  COLLISIONTYPE     4904
  PERSONCOUNT      0
```

There are 19 variables (out of 38) with missing values. There are 7 variables have a large number of missing values (at least 40%) out of the total 194673 records.

After removing those variables with a large number of missing values, all the remaining variables have missing values less than 3.5%.

# Excluding Variables with Same Values in Different Formats

SEVERITYCODE	SEVERITYDESC
2	Injury Collision
1	Property Damage
1	Property Damage
1	Property Damage
2	Injury Collision
1	Property Damage
1	Property Damage
2	Injury Collision
1	Property Damage
2	Injury Collision
1	Property Damage
1	Property Damage
1	Property Damage
1	Property Damage
2	Injury Collision
1	Property Damage
2	Injury Collision
1	Property Damage
2	Injury Collision
1	Property Damage

5 pairs: SEVERITYCODE & SEVERITYDESC, INCDATE & INCDTTM, SDOT\_COLCODE & SDOT\_COLDESC, and ST\_COLCODE & ST\_COLDESC.

Only one variable of each pair is kept in the data set. Besides, SEVERITYCODE has existed for twice in this data set, so one of them will be removed.





# Deleting Unnecessary Variables

Since the value of some variables are not relevant to the value of the severity that we want to predict, five variables are excluded: X, Y, LOCATION, SDOT\_COLCODE, INCDATE and ST\_COLCODE.

Other variables, such as the dummy variable created for ADDRTHOE=Alley, are deleted based on the results of t-test.

# Changing Values to the Same Format

UNDERINFL	
0	80394
1	3995
N	100274
Y	5126

The values of UNDERINFL used two formats: 1/0 and Y/N.

According to the understanding of data, these two formats are indicating the same values. These values will be changed to use the same format – 1 for Y and 0 for N.



# Data Transformation

## **Binary Variables:**

Transforming Binary Variable to 1/0 for variables – SEVERITYCODE, STATUS, and HITPARKEDCAR, is necessary since models like logistic regression can only compute numeric values.

## **Categorical Variables:**

Transforming categorical variables to dummy variables – ADDRTYPE, COLLISIONTYPE, JUNCTIONTYPE, WEATHER, ROADCOND, and LIGHTCOND,



# Dealing with Imbalanced Data with Resampling Methods

## **Upsampling:**

To keep as much information as possible, upsampling is used for logistic regression and decision tree. After upsampling, there are 265260 records used to train the models.

## **Downsampling:**

Due to computability, downsampling is used for k-nearest neighbors. After downsampling, there are 114318 records used to train the models.

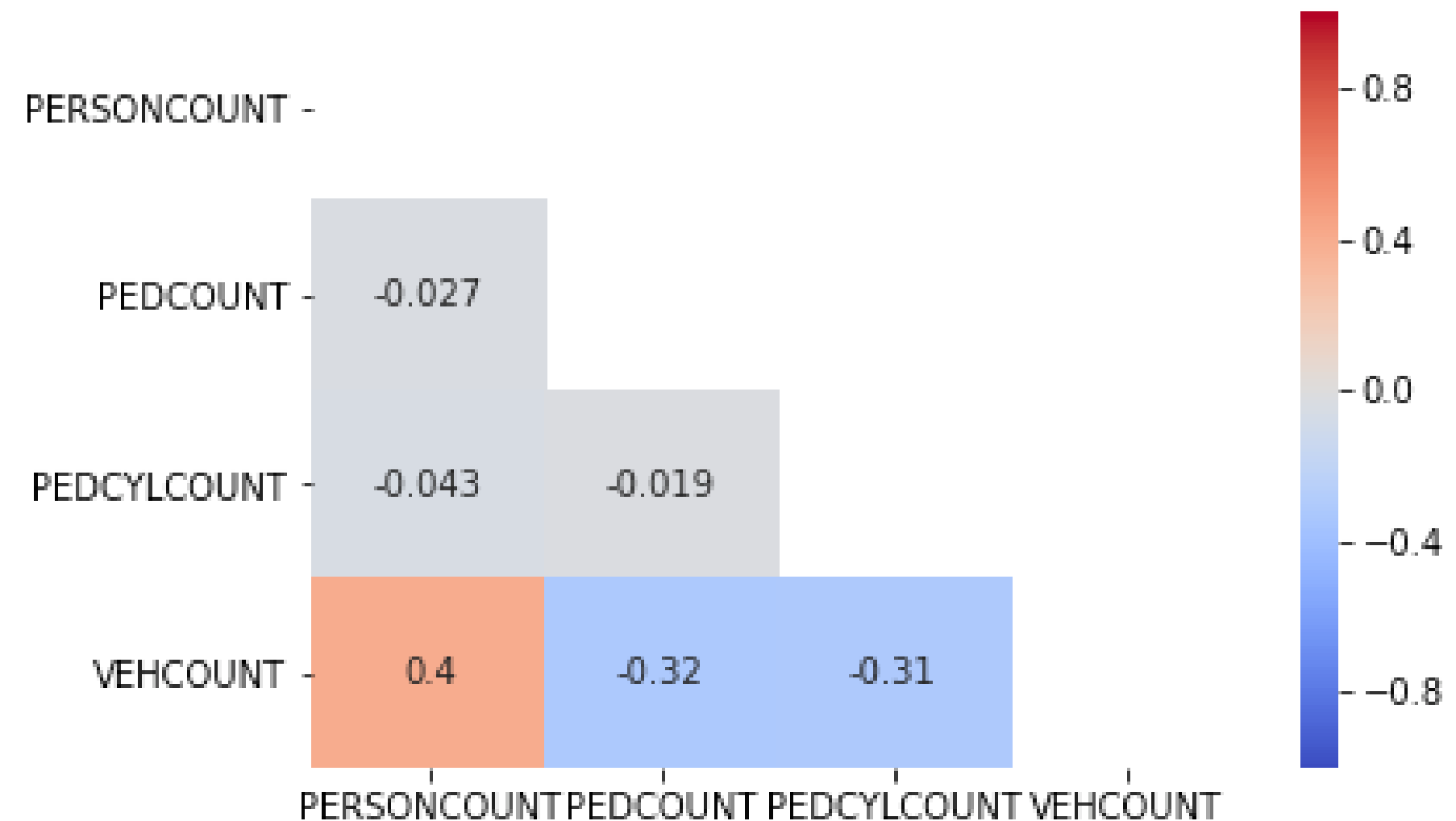
The background features abstract blue geometric shapes, including circles, triangles, and irregular blobs, scattered across the white canvas. A large, light blue parallelogram is positioned behind the main text.

PART 3

# Data Analysis

# Correlation for Numeric Data

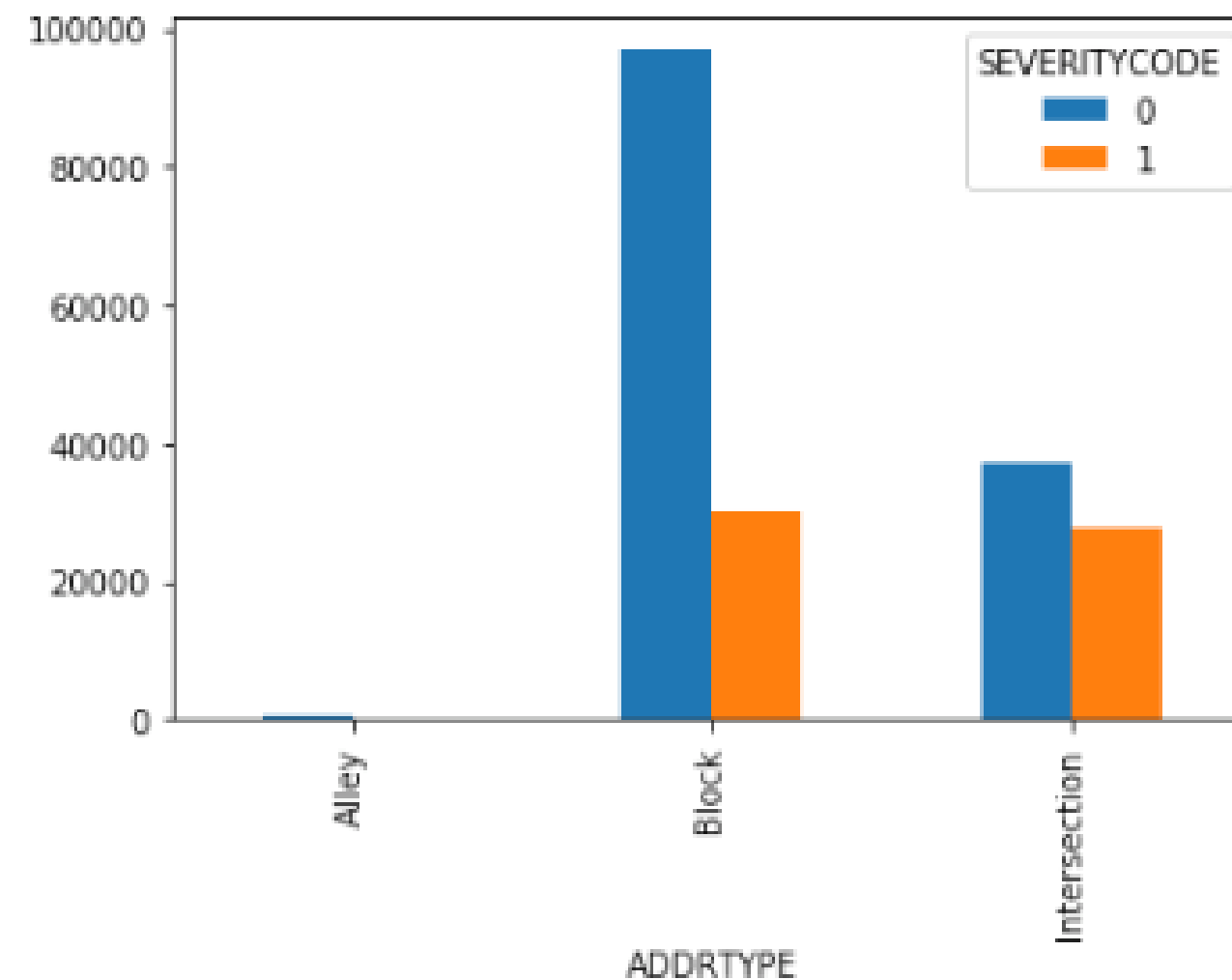
The correlations of the numeric variables are very low, which means, there is no collinearity problem here.



# Bar Charts and ANOVA Tests for Categorical Data

As shown in the bar chart and the result of the ANOVA tests, we can find that, in terms of severity, the impact of different address types has a significant difference.

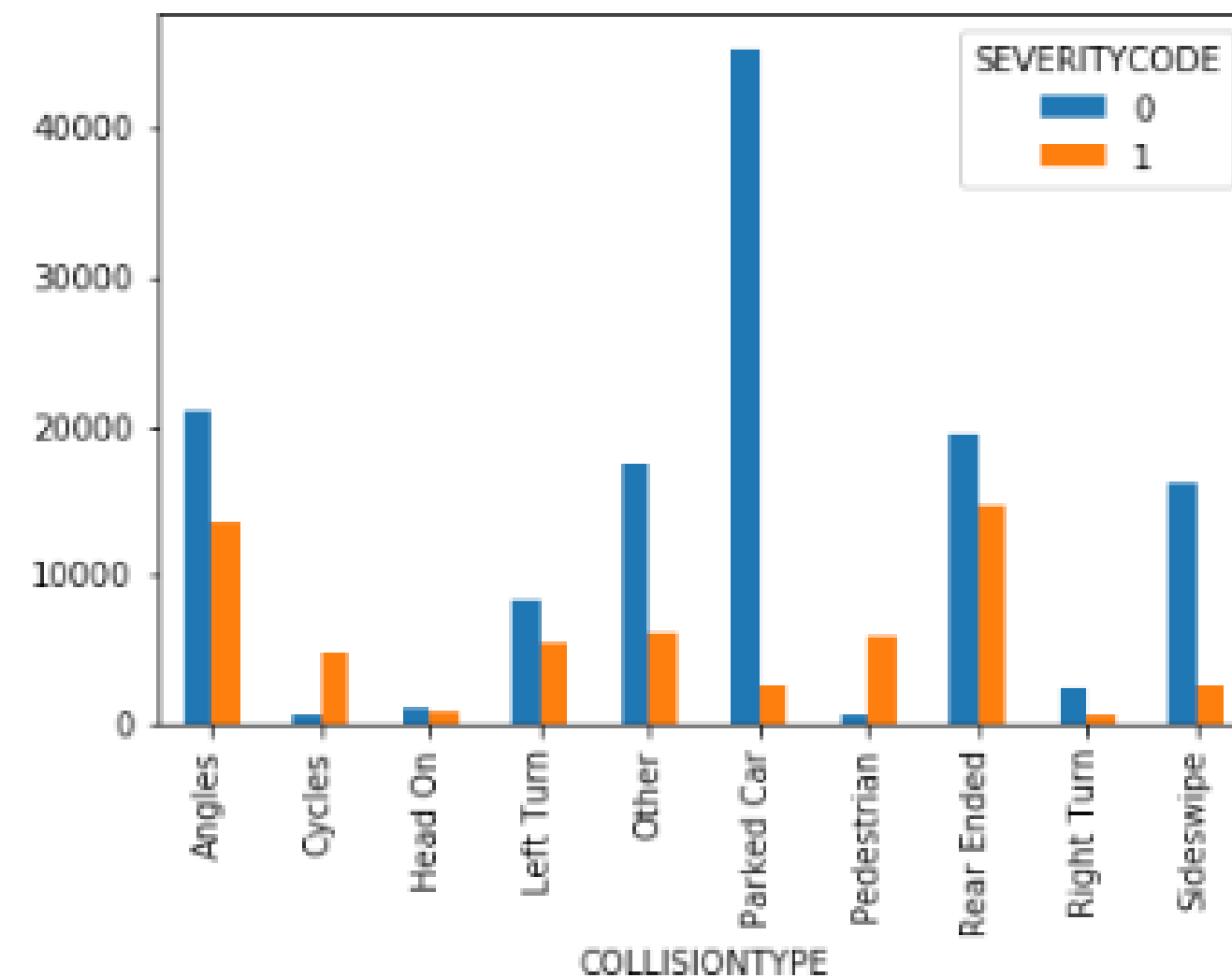
```
F_onewayResult(statistic=3530.8078377494403, pvalue=0.0)
```



# Bar Charts and ANOVA Tests for Categorical Data

As shown in the bar chart and the result of the ANOVA tests, we can find that, in terms of severity, the impact of different collision types has a significant difference.

```
F_onewayResult(statistic=5369.001306448202, pvalue=0.0)
```

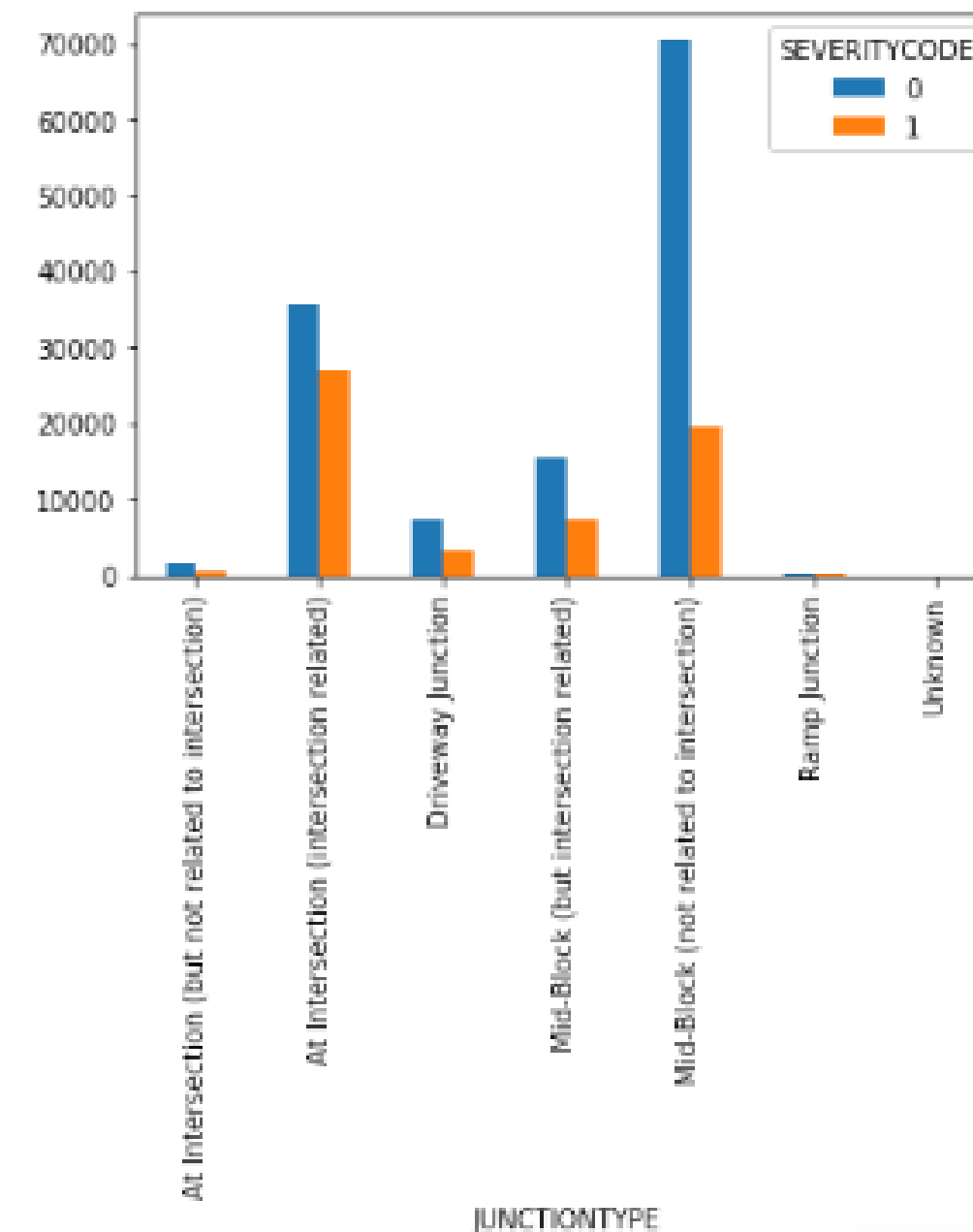




# Bar Charts and ANOVA Tests for Categorical Data

As shown in the bar chart and the result of the ANOVA tests, we can find that, the impact of different categories of junction has a significant difference.

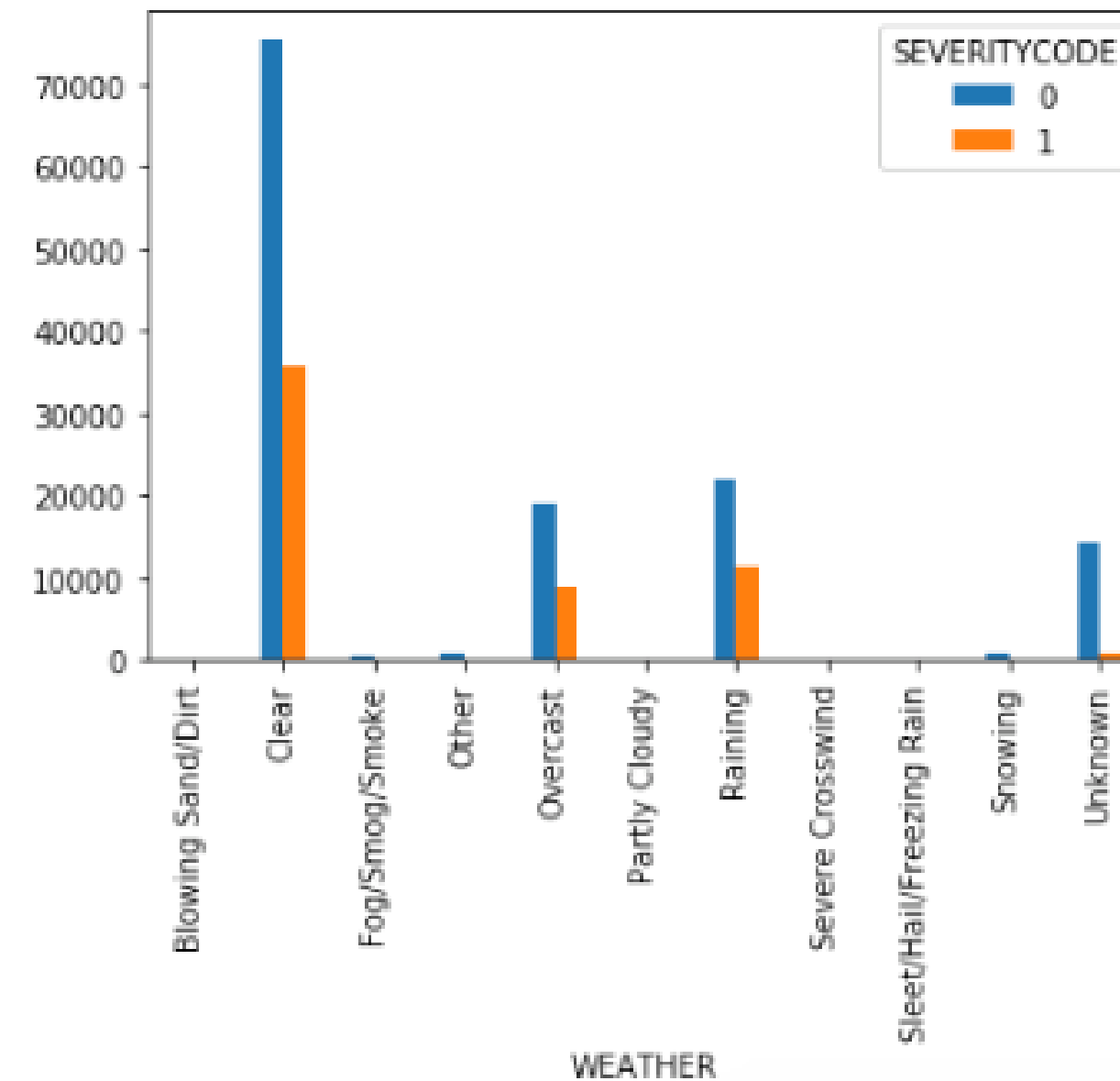
`F_onewayResult(statistic=1403.6655189455819, pvalue=0.0)`



# Bar Charts and ANOVA Tests for Categorical Data

As shown in the bar chart and the result of the ANOVA tests, we can find that, in terms of severity, the impact of different weather types has a significant difference.

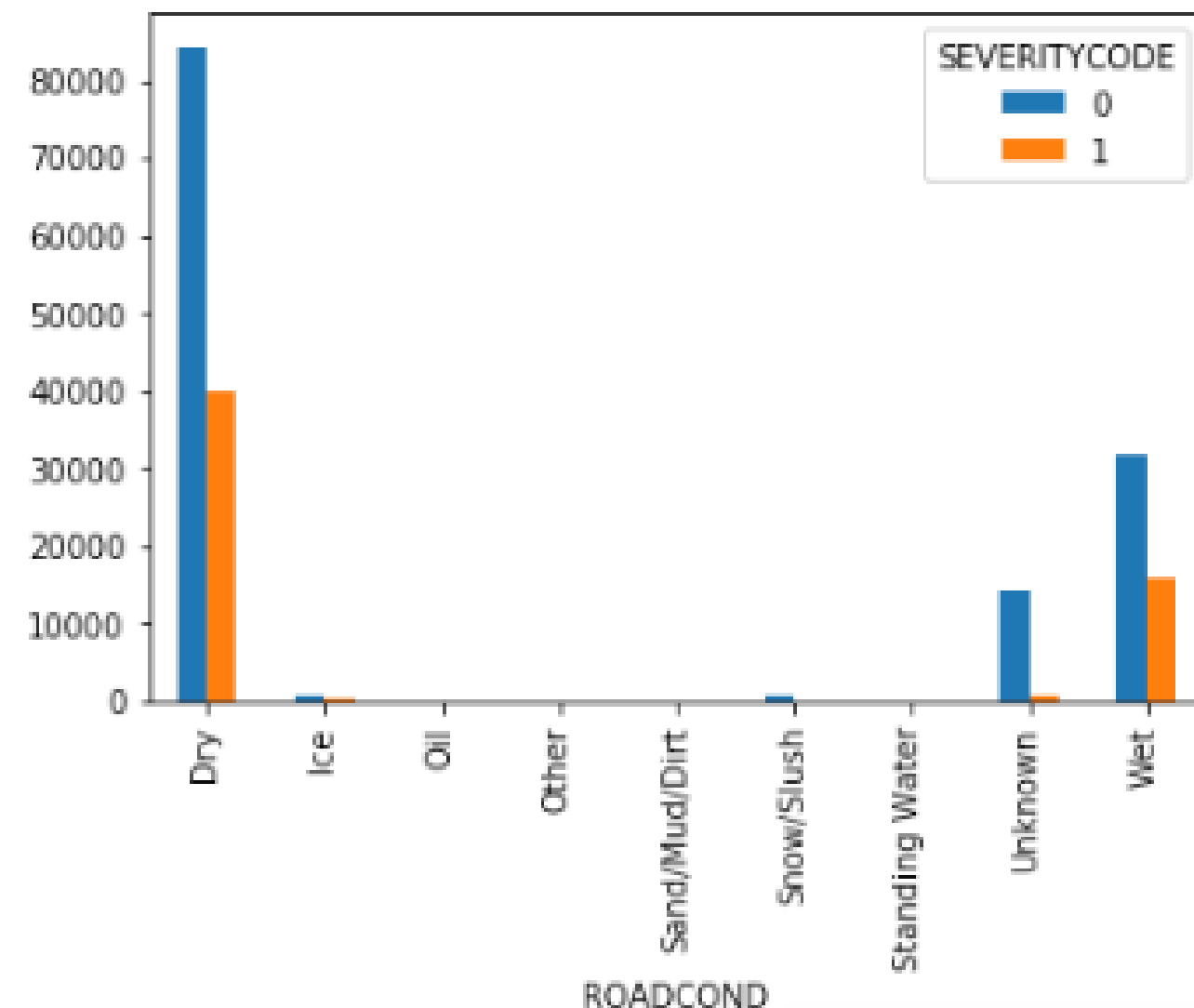
```
F_onewayResult(statistic=374.0303665512938, pvalue=0.0)
```



# Bar Charts and ANOVA Tests for Categorical Data

As shown in the bar chart and the result of the ANOVA tests, we can find that, in terms of severity, the impact of different conditions of the road during the collision has a significant difference.

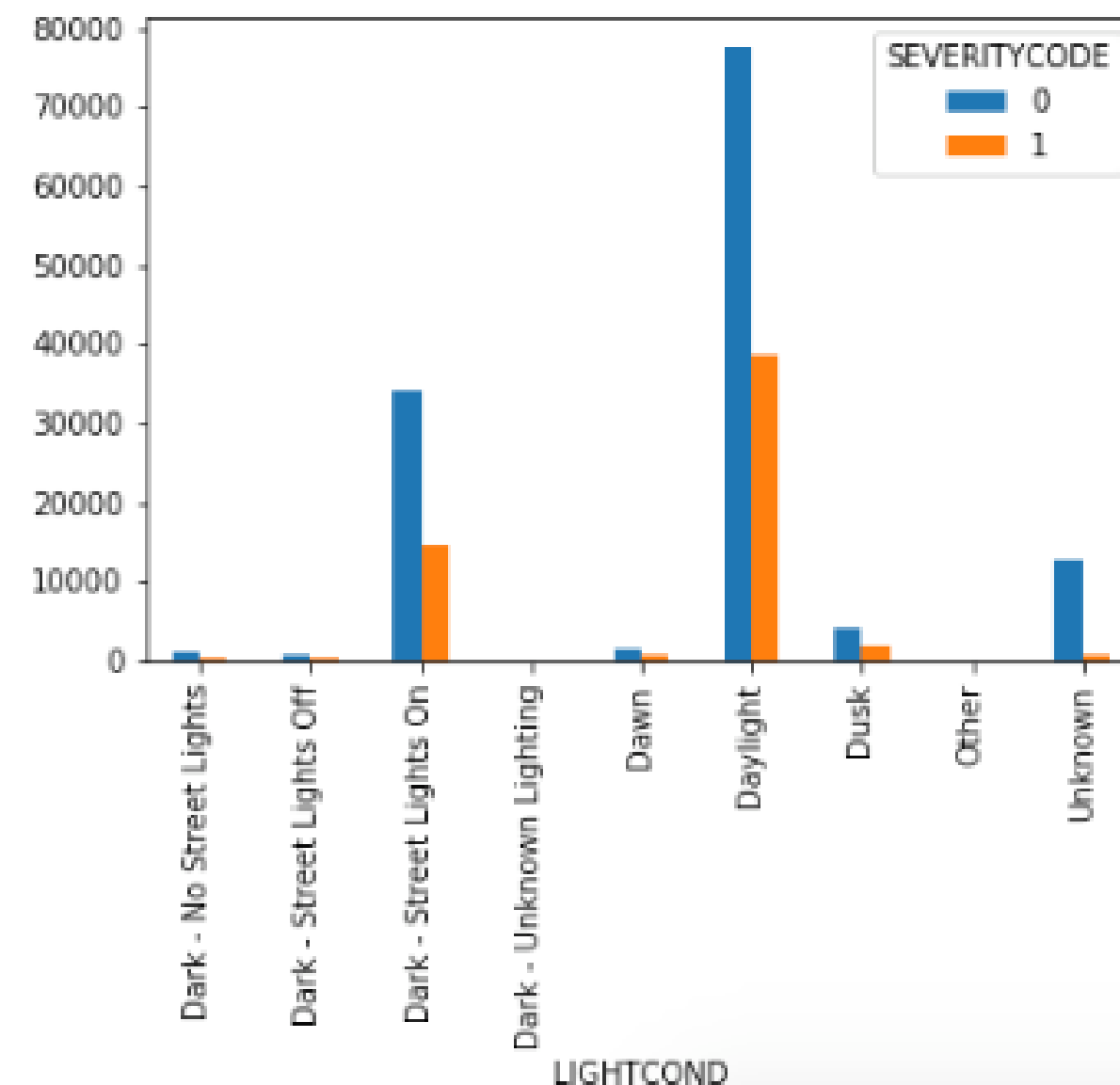
```
F_onewayResult(statistic=476.040939664256, pvalue=0.0)
```



# Bar Charts and ANOVA Tests for Categorical Data

As shown in the bar chart and the result of the ANOVA tests, we can find that, in terms of severity, the impact of different conditions of the light during the collision has a significant difference.

`F_onewayResult(statistic=463.6530730759219, pvalue=0.0)`



# T-test

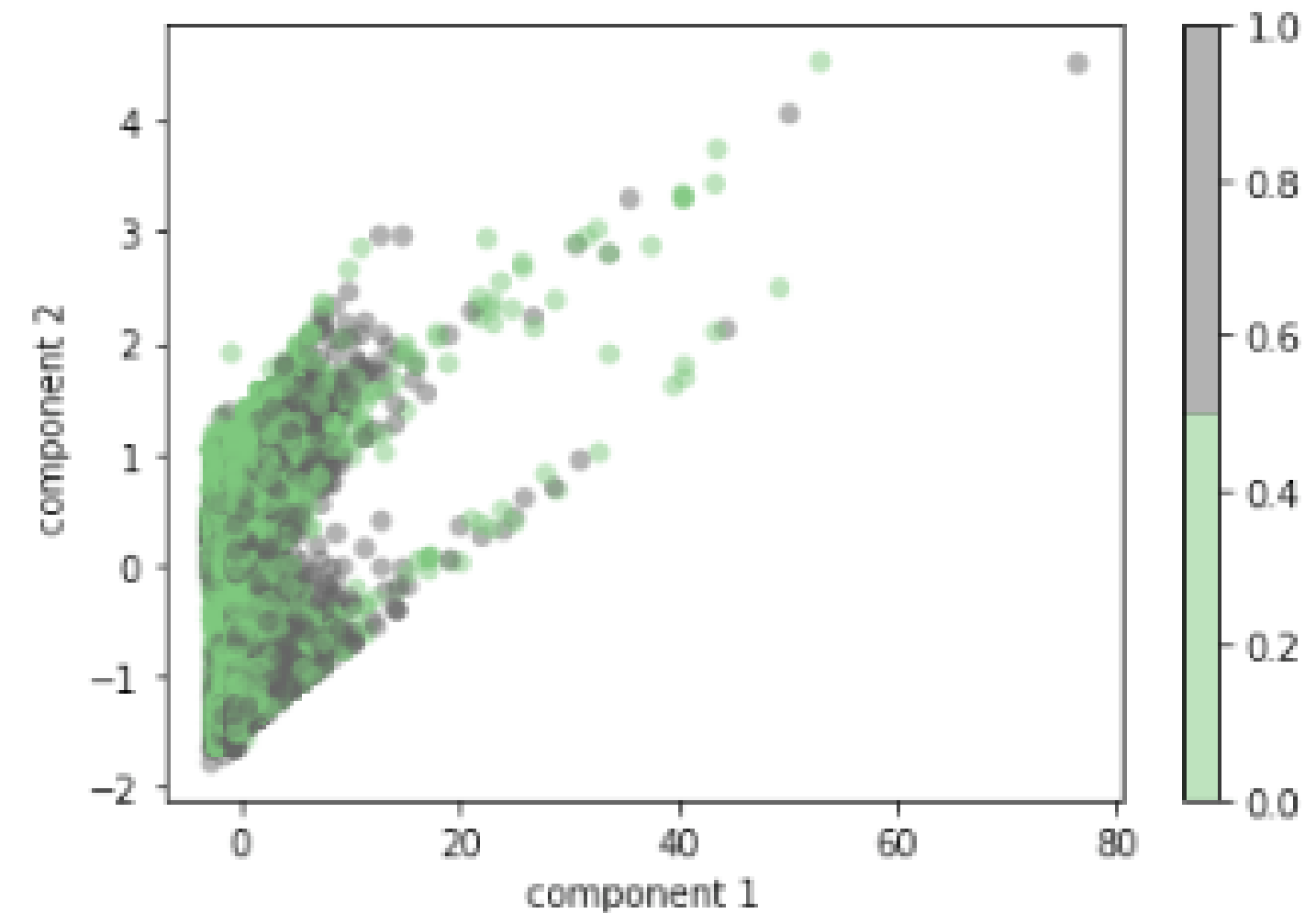
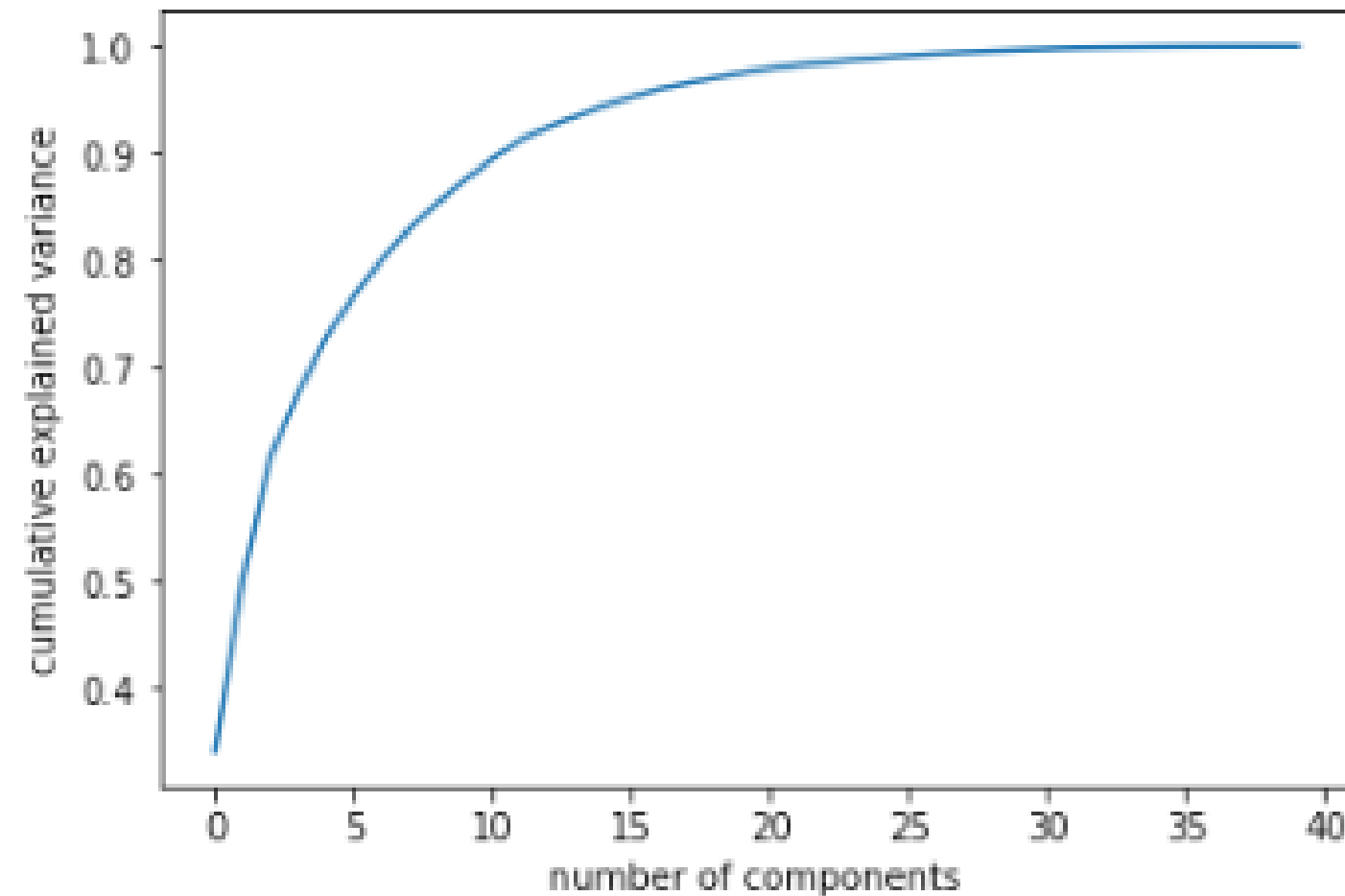
As an example of the result of t-test shown, we can notice that no matter which status it is recorded, statistically speaking, there is no significant difference on the value of severity code between the two groups of status.

```
results = rp.ttest(group1= df_trans['SEVERITYCODE'][df_trans.iloc[:,1] == 1], group1_name= "1",  
                  group2= df_trans['SEVERITYCODE'][df_trans.iloc[:,1] == 0], group2_name= "0")  
results[1].iloc[3,1]  
  
/opt/conda/envs/Python36/lib/python3.6/site-packages/scipy/stats/_distn_infrastructure.py:1920:  
multiply  
    lower_bound = self.a * scale + loc  
/opt/conda/envs/Python36/lib/python3.6/site-packages/scipy/stats/_distn_infrastructure.py:1921:  
multiply  
    upper_bound = self.b * scale + loc  
  
]: 0.2555
```

Similarity, we can notice that other variables, for example, left turn collision type, ramp junction and dawn weather.

# Principal Component Analysis

There are 3 principal components chosen. However, we can also notice that, the cumulative explained variance is relatively low, around 70%



The background features abstract blue geometric shapes, including circles, triangles, and irregular polygons, scattered across the white canvas. A large, light blue trapezoidal shape is positioned behind the text.

PART 4

# Model Evaluation




# Logistic Regression

**With upsampling data and the best parameter value for C as 0.001:**

the average F1-score of training data is 0.73.  
The three F1-scores are: 0.724, 0.724, 0.727.

The F1-score of the testing data is 0.702.

Besides, the time to complete the model building and predicting is relatively short.








## K-Nearest Neighbors

**With upsampling data and the best  
parameter value for C as 0.001:**

the average F1-score of training data is 0.68.  
The three F1-scores are: 0.682, 0.677, 0.683.

The F1-score of the testing data is 0.686

Besides, the time to complete the model  
building and predicting is the longest.






# Decision Tree

**With upsampling data and the best parameter value for C as 0.001:**

the average F1-score of training data is 0.73.  
The three F1-scores are: 0.724, 0.725, 0.727.

The F1-score of the testing data is 0.705.

Besides, the time to complete the model building and predicting is the shortest.



# CONCLUSION

Based on the results shown below, Decision Tree has a slightly better performance than Logistic Regression and a much better performance than K-Nearest Neighbor. Moreover, the time consuming of Decision Tree is also shorter than the other models.

## **F1-score with Testing Data:**

Logistic Regression – 0.702

K-Nearest Neighbors – 0.686

Decision Tree – 0.750



# RECOMMENDATION

The Ensemble Learning is recommended for future usage with the three models, because all the three models have a better performance than random guessing.

For example, if both logistic regression and decision tree predict "1" while KNN predicts "0", ensemble learning will assign "1" to the class because the majority algorithms (2 out of 3) suggest class "1".





**THANK YOU**