Seattle Car Accident Severity

Abhilash

Business Understanding

- The Seattle government is going to prevent avoidable car accidents by employing methods that alert drivers, health system, and police to remind them to be more careful in critical situations.
- In most cases, not paying enough attention during driving, abusing drugs and alcohol or driving at very high speed are the main causes of occurring accidents that can be prevented by enacting harsher regulations.
- Besides the aforementioned reasons, weather, visibility, or road conditions are the major uncontrollable factors that can be prevented by revealing hidden patterns in the data and announcing warning to the local government, police and drivers on the targeted roads.

Target Audience

- The target audience of the project is local Seattle government, police, rescue groups, and last but not least, car insurance institutes. The model and its results are going to provide some advice for the target audience to make insightful decisions for reducing the number of accidents and injuries for the city.

Data

We chose the unbalanced dataset provided by the Seattle Department of Transportation Traffic Management Division with 194673 rows (accidents) and 37 columns (features) where each accident is given a severity code. It covers accidents from January 2004 to May 2020. Some of the features in this dataset include and are not limited to Severity code, Location/Address of accident, Weather condition at the incident site, Driver state (whether under influence or not), collision type. Hence we think its a good generalized dataset which will help us in creating an accurate predictive model.

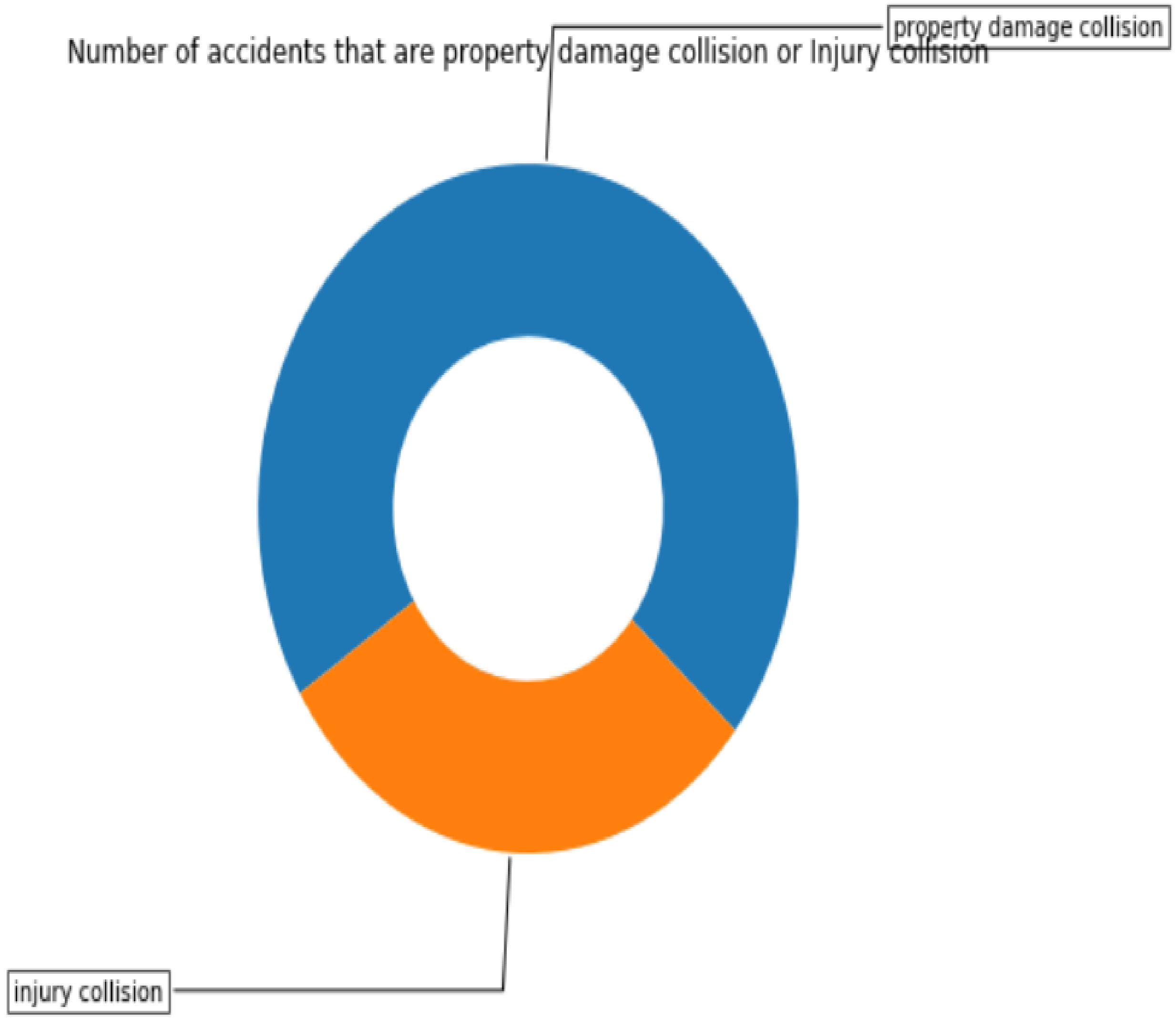
The unbalance with respect to the severity code in the dataset is as follows.

SEVERITY CODE Count

- 1 – 136485
- 2 – 58188

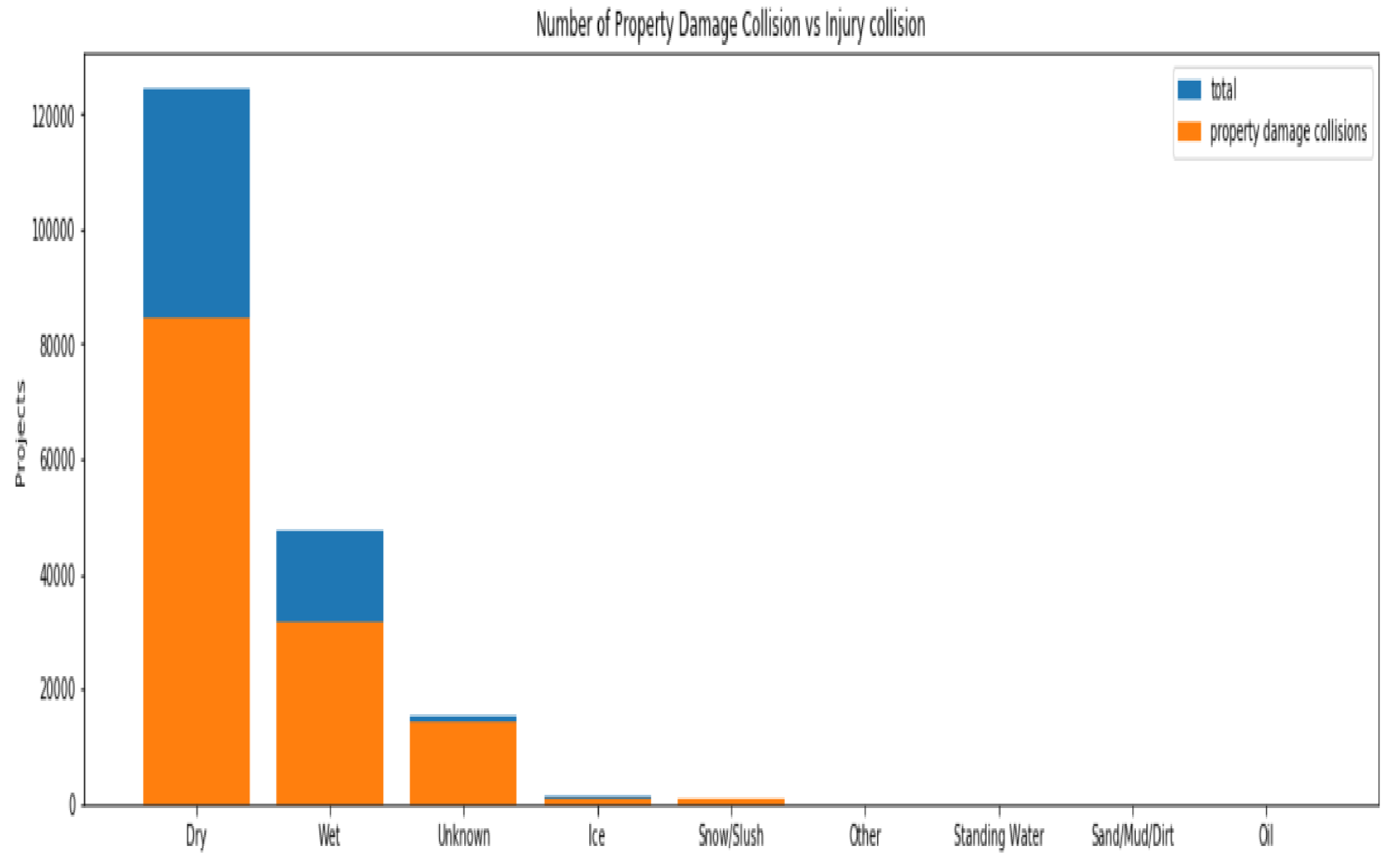
Exploratory Data Analysis

```
Number of accidents that are property damage collision 136485 , ( 70.10987656223513 %)
Number of accidents that are injury collision 58188 , ( 29.890123437764863 %)
```



Exploratory Data Analysis

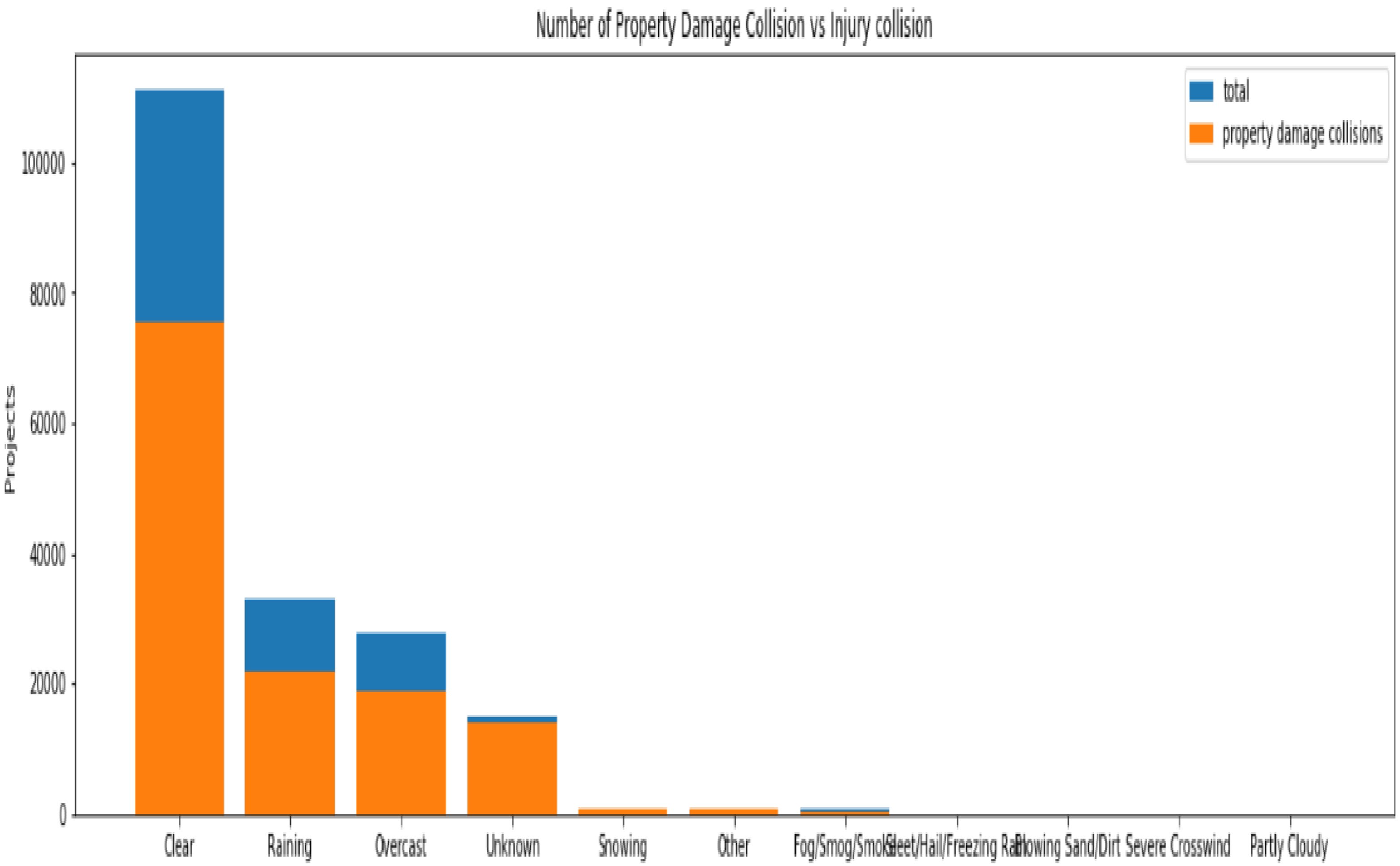
ROADCOND



	ROADCOND	SEVERITYCODE	total	Avg
0	Dry	84446	124510	1.321773
8	Wet	31719	47474	1.331866
7	Unknown	14329	15078	1.049675
1	Ice	936	1209	1.225806
5	Snow/Slush	837	1004	1.166335

Exploratory Data Analysis

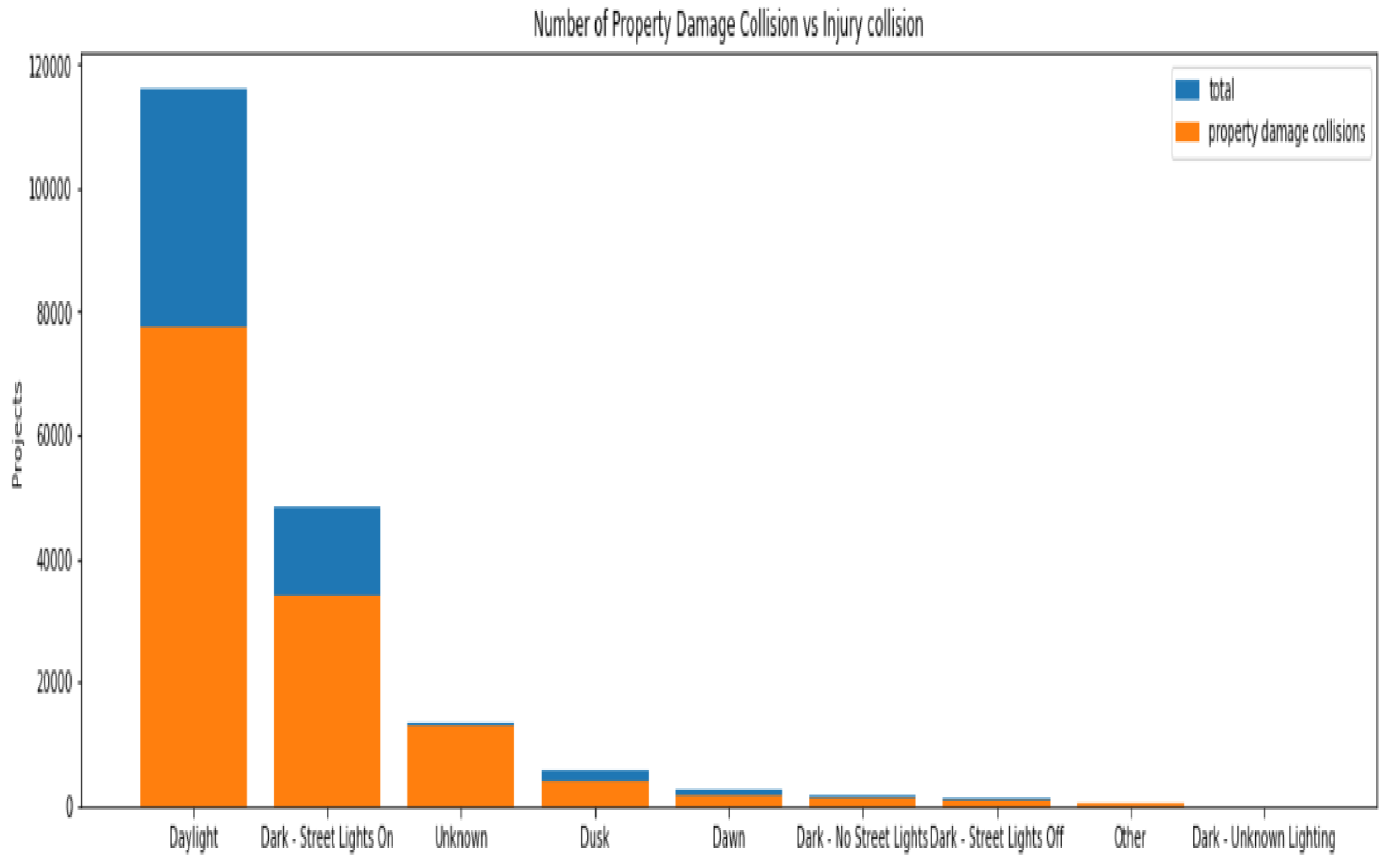
WEATHER



	WEATHER	SEVERITYCODE	total	Avg
1	Clear	75295	111135	1.322491
6	Raining	21969	33145	1.337185
4	Overcast	18969	27714	1.315544
10	Unknown	14275	15091	1.054072
9	Snowing	736	907	1.188534

Exploratory Data Analysis

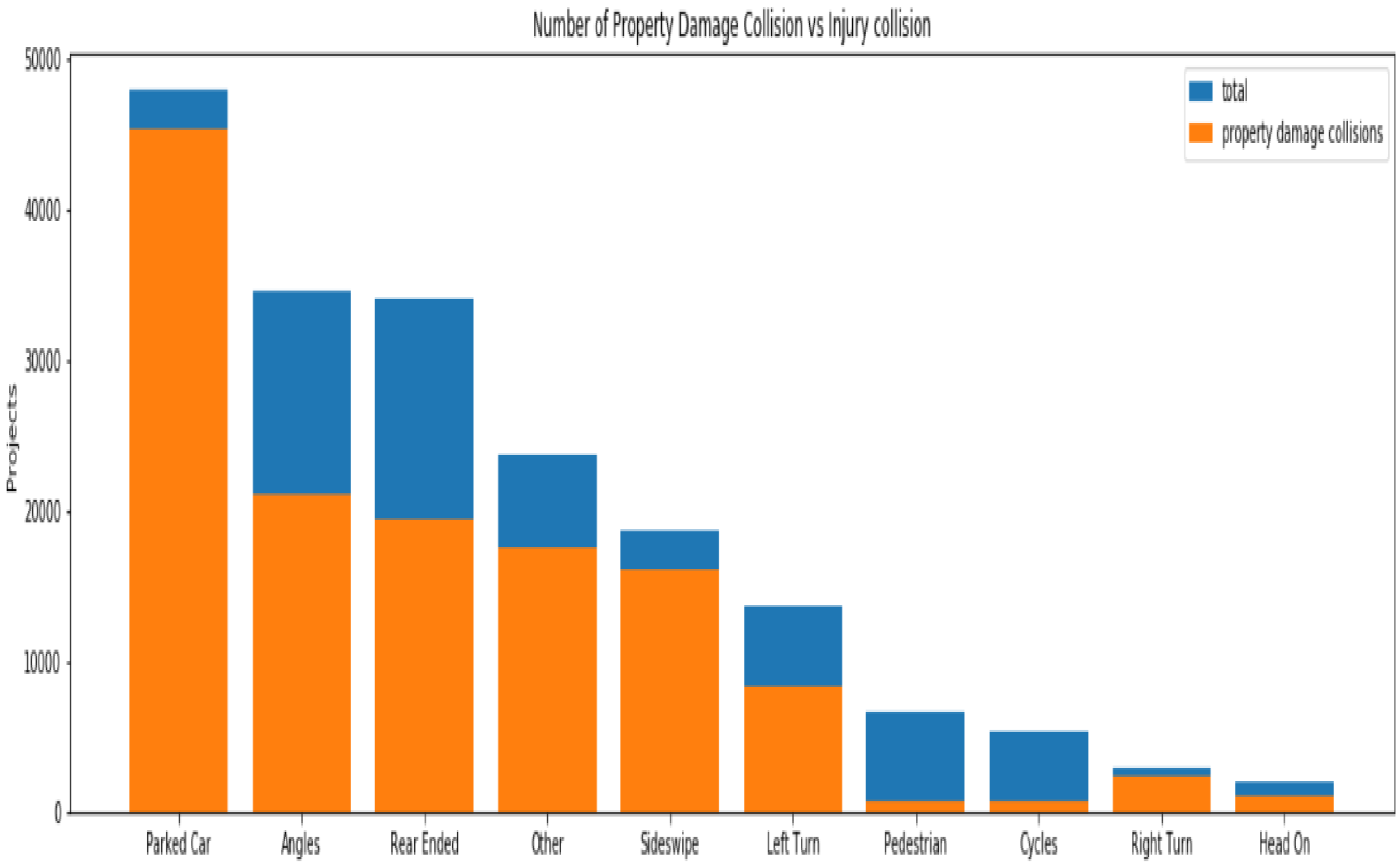
LIGHTCOND



	LIGHTCOND	SEVERITYCODE	total	Avg
5	Daylight	77593	116137	1.331884
2	Dark - Street Lights On	34032	48507	1.298411
8	Unknown	12868	13473	1.044905
6	Dusk	3958	5902	1.329380
4	Dawn	1678	2502	1.329337

Exploratory Data Analysis

COLLISIONTYPE



	COLLISIONTYPE	SEVERITYCODE	total	Avg
5	Parked Car	45325	47987	1.055473
0	Angles	21050	34674	1.392917
7	Rear Ended	19419	34090	1.430361
4	Other	17591	23703	1.257858
9	Sideswipe	16103	18609	1.134666

Machine Learning Models

KNN

K Nearst Neigbours

```
from sklearn.neighbors import KNeighborsClassifier
k = 17
knn = KNeighborsClassifier(n_neighbors = k).fit(X_train,y_train)

knn_y_pred = knn.predict(X_test)
knn_y_pred[0:5]
```

```
array([2, 2, 1, 1, 2], dtype=int64)
```

KNN Evaluation

```
jaccard_score(y_test, knn_y_pred)
```

```
0.3091637411108111
```

```
f1_score(y_test, knn_y_pred, average='macro')
```

```
0.5477714681769319
```

Machine Learning Models

Decision Tree

Decision Tree

```python
from sklearn.tree import DecisionTreeClassifier
dt = DecisionTreeClassifier(criterion="entropy", max_depth = 7)

dt.fit(X_train,y_train)
```

```
DecisionTreeClassifier(criterion='entropy', max_depth=7)
```

```python
dt_y_pred = dt.predict(X_test)
```

Decision Tree Evaluation

```python
jaccard_score(y_test, dt_y_pred)
```

```
0.2873687679487783
```

```python
f1_score(y_test, dt_y_pred, average='macro')
```

```
0.5450597937389444
```

Machine Learning Models

Linear Regression

Linear Regression

```python
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix
LR = LogisticRegression(C=6, solver='liblinear').fit(X_train,y_train)
```

```python
LR_y_pred = LR.predict(X_test)
```

```python
LR_y_prob = LR.predict_proba(X_test)
```

```python
LR_y_prob = LR.predict_proba(X_test)
log_loss(y_test, LR_y_prob)
```

```
0.6849535383198887
```

Linear Regression Evaluation

```python
jaccard_score(y_test, LR_y_pred)
```

```
0.2720073907879108
```

```python
f1_score(y_test, LR_y_pred, average='macro')
```

```
0.511602093963383
```

Result

ML Model	Jaccard Score	F1 Score	Accuracy
KNN	0.30	0.55	0.56
Decision Tree	0.28	0.54	0.57
Linear Regression	0.27	0.51	0.53

Based on the above table, KNN is the best model to predict car accident severity.

Conclusion

Based on the dataset provided for this capstone from weather, road, and light conditions pointing to certain classes, we can conclude that particular conditions have a somewhat impact on whether or not travel could result in property damage (class 1) or injury (class 2).