

Oman Traffic Analysis

1. Data Extraction

```
In [2]: """
        @author: Aamir M. Khan
        Created First: Mon Feb 25 2019
        Updated Last:

        """
        %matplotlib inline
```

Import Libraries

```
In [206]: import os
import scipy.io
import numpy as np
import pandas as pd
from math import log10, floor
from datetime import datetime

# from sklearn.model_selection import train_test_split
# from sklearn.pipeline import make_pipeline
# from sklearn.ensemble import ExtraTreesClassifier
# from sklearn.ensemble import RandomForestClassifier
# from sklearn.feature_selection import SelectFromModel
# from sklearn.feature_selection import SelectKBest
# from sklearn.model_selection import StratifiedKFold
# from sklearn.model_selection import GridSearchCV
# from sklearn.ensemble.gradient_boosting import GradientBoostingClassifier
# from sklearn.metrics import accuracy_score
# from sklearn import metrics

from IPython.display import SVG
import matplotlib.pyplot as plt

In [7]: # fix random seed for reproducibility
seed = 7
np.random.seed(seed)
```

Load Data Files

```
In [177]: accidents = pd.read_csv("accidents_monthly.csv")
accidents.head()

Out[177]:
```

	Year	January	February	March	April	May	June	July	August	September	October	November	December
0	2000	1122	984	1241	1047	1185	1087	1205	1076	1096.0	1046.0	987.0	964.0
1	2001	1114	999	1055	1010	1081	1028	1014	1043	1160.0	1283.0	1203.0	1111.0
2	2002	1251	1125	1097	696	695	635	552	662	535.0	587.0	646.0	626.0
3	2003	754	941	996	950	1026	845	838	767	711.0	827.0	758.0	784.0
4	2004	876	729	735	816	846	793	804	891	743.0	805.0	688.0	735.0

```
In [178]: injuries = pd.read_csv("injuries_month.csv")
injuries.head()

Out[178]:
```

	Year	January	February	March	April	May	June	July	August	September	October	November	December
0	2000	883	755	852	737	794	657	849	777	717.0	768.0	691.0	843.0
1	2001	868	689	856	716	767	819	827	774	761.0	857.0	869.0	822.0
2	2002	985	855	740	563	609	641	468	686	494.0	696.0	588.0	629.0
3	2003	604	642	527	546	567	509	608	654	444.0	603.0	551.0	480.0
4	2004	560	571	477	523	537	514	567	744	503.0	614.0	563.0	463.0

```
In [179]: accidents.shape, injuries.shape

Out[179]: ((19, 13), (19, 13))
```

Cleaning the Data

accidents table

```
In [180]: accidents = pd.melt(accidents,id_vars=['Year'],var_name='Month',value_name='Accidents')
accidents = accidents.reindex()
accidents.head(5)

Out[180]:
```

	Year	Month	Accidents
0	2000	January	1122.0
1	2001	January	1114.0
2	2002	January	1251.0
3	2003	January	754.0
4	2004	January	876.0

```
In [181]: accidents.dropna(inplace=True)
```

```
In [182]: months = {'January':1,'February':2,'March':3,'April':4,'May':5,'June':6,'July':7,
                  'August':8,'September':9,'October':10,'November':11,'December':12}
accidents.Month = accidents.Month.map(months)
```

```
In [183]: accidents['Time'] = pd.to_datetime(accidents.assign(Day=1).loc[:, ['Year', 'Month', 'Day']])
accidents.drop(['Year', 'Month'],axis=1,inplace=True)
accidents.set_index(['Time'], inplace=True)
accidents.head(5)
```

Out[183]:

	Accidents
Time	
2000-01-01	1122.0
2001-01-01	1114.0
2002-01-01	1251.0
2003-01-01	754.0
2004-01-01	876.0

injuries table

```
In [190]: injuries = pd.melt(injuries,id_vars=['Year'],var_name='Month',value_name='Injuries')
injuries = injuries.reindex()
injuries.head(5)
```

Out[190]:

	Year	Month	Injuries
0	2000	January	883.0
1	2001	January	868.0
2	2002	January	985.0
3	2003	January	604.0
4	2004	January	560.0

```
In [191]: injuries.dropna(inplace=True)
```

```
In [192]: months = {'January':1,'February':2,'March':3,'April':4,'May':5,'June':6,'July':7,
                  'August':8,'September':9,'October':10,'November':11,'December':12}
injuries.Month = injuries.Month.map(months)
```

```
In [193]: injuries['Time'] = pd.to_datetime(injuries.assign(Day=1).loc[:, ['Year', 'Month', 'Day']])
injuries.drop(['Year', 'Month'],axis=1,inplace=True)
injuries.set_index(['Time'], inplace=True)
injuries.head(5)
```

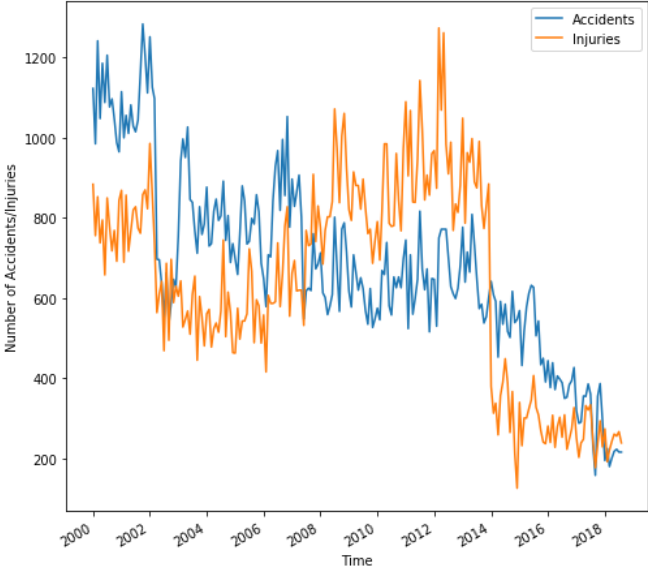
Out[193]:

	Injuries
Time	
2000-01-01	883.0
2001-01-01	868.0
2002-01-01	985.0
2003-01-01	604.0
2004-01-01	560.0

Merge two tables

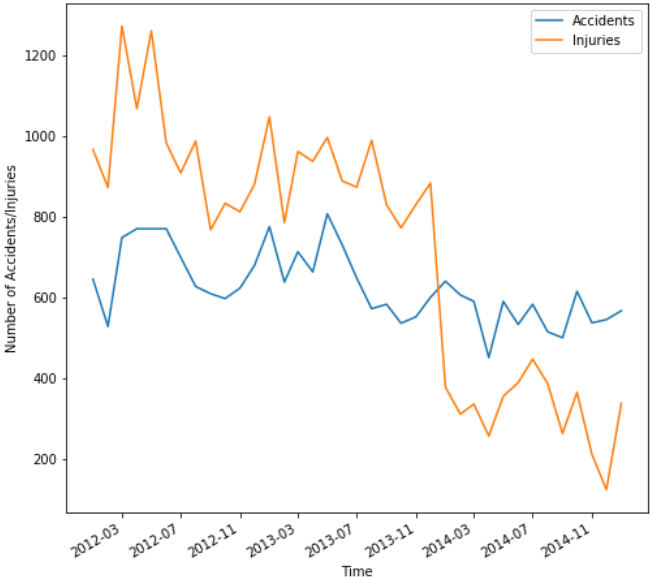
Exploratory Data Analysis

```
In [199]: plt.figure(figsize=(8,8))
accidents.Accidents.plot();
injuries.Injuries.plot();
plt.legend();
plt.ylabel('Number of Accidents/Injuries');
```



Visual for the years 2012-2014

```
In [205]: plt.figure(figsize=(8,8))
accidents.loc['2012-01-01':'2015-01-01'].Accidents.plot();
injuries.loc['2012-01-01':'2015-01-01'].Injuries.plot();
plt.legend();
plt.ylabel('Number of Accidents/Injuries');
```



Datatype integrity check

Feature engineering

Split Training/Validation/Test Sets

MACHINE LEARNING

2. Train the Model

MODEL 1

MODEL 2

MODEL 3

3. Predict New Data

Clean data

Make predictions

Rough Work