```
import os
import matplotlib.pylab as plt
import pandas as pd
import datetime, warnings, scipy
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn import metrics, linear_model
from sklearn.model_selection import train_test_split, cross_val_score, cross_val_predict
from sklearn.preprocessing import StandardScaler
import keras
import keras.utils
from keras import utils as np_utils
from keras.models import Sequential,Model
from keras.models import Model as KerasModel
from keras.layers import Input, Dense, Activation, Reshape
from keras.layers import Concatenate, Dropout
from keras.layers.embeddings import Embedding
from keras.callbacks import ModelCheckpoint
from keras.utils import plot_model
plt.rcParams["patch.force_edgecolor"] = True
plt.style.use('fivethirtyeight')
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "last_expr"
pd.options.display.max_columns = 50
warnings.filterwarnings("ignore")
%matplotlib inline
from scipy import stats
```

```
Using TensorFlow backend.
```

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6bn6qk8qdgf4n4g3pfee6491hc0brc4i.apps.google

Enter your authorization code:
```

Saved successfully!

```
cd /content/gdrive/My Drive/Colab Notebooks/dataset/Indiapolice/indiapolice/MorthDataSets/
```

```
/content/gdrive/My Drive/Colab Notebooks/dataset/Indiapolice/indiapolice/MorthDataSets
```

```
ls
```

```
analy2.csv    gdrive/    subset/                      top.csv
analysis.csv  model.png  tbl_tr_accident_report.xlsx
```

```
data_df = pd.read_excel('tbl_tr_accident_report.xlsx')
print('DataFrame size:', data_df.shape)
```

```
DataFrame size: (201201, 89)
```

```
data_df.describe()
```

	Unit_ID	Crime_No	Accident_DateTime	Fatal_NonFatal	Accident_Classification	Severity	Collision_Type	Accident_Spo
count	201201.000000	2.012010e+05	0.0	172590.0	201201.000000	201201.000000	201201.000000	201201.00000
mean	1186.131028	2.016045e+09	NaN	0.0	6287.527149	6128.000219	6732.037743	6286.78454
std	599.210953	1.461009e+06	NaN	0.0	60.626527	80.404165	1617.104603	285.22280
min	57.000000	2.014000e+09	NaN	0.0	0.000000	0.000000	0.000000	0.00000
25%	726.000000	2.015000e+09	NaN	0.0	6274.000000	6112.000000	6091.000000	6083.00000
50%	1203.000000	2.016000e+09	NaN	0.0	6274.000000	6112.000000	6093.000000	6083.00000
75%	1761.000000	2.017000e+09	NaN	0.0	6274.000000	6113.000000	6525.000000	6719.00000
max	2245.000000	2.019000e+09	NaN	0.0	6514.000000	11012.000000	20104.000000	6810.00000

8 rows × 62 columns

```
#data_df = data_df.sample(n=1000, random_state=2)
```

```
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 201201 entries, 0 to 201200
Data columns (total 89 columns):
Unit_ID                         201201 non-null int64
Crime_No                        201201 non-null int64
Accident_DateTime               0 non-null float64
Fatal_NonFatal                  172590 non-null float64
Accident_Classification         201201 non-null int64
Severity                        201201 non-null int64
Collision_Type                  201201 non-null int64
Accident_Spot                   201201 non-null int64
Accident_Location               201200 non-null object
Accident_Zone                   0 non-null float64
NoOfVehicle_Involved            201201 non-null int64
Junction_Control                201201 non-null int64
Road_Character                  201201 non-null int64
Road_Type                       201201 non-null int64
Road_Classification             201201 non-null int64
Road_Width                      45337 non-null object
Shoulder_Width                  201201 non-null int64
Seperation                      201201 non-null int64
Surface_Type                    201201 non-null int64
Surface_Condition               201201 non-null int64
Road_Condition                  201201 non-null int64
weather                         201201 non-null int64
Location_Type                   201201 non-null int64
Main_Cause                      201201 non-null int64
Hit_Run                         201201 non-null int64
XY_Map                          1994 non-null object
X_Value                         1852 non-null object
AccidentLocation_Map            9641 non-null object
Collision_Diagram               523 non-null object
Y_Value                         1813 non-null object
Route                           4554 non-null object
KM                              6409 non-null object
HundredM                        201201 non-null int64
Node_Map                        201201 non-null int64
Node1                           201201 non-null int64
Node2                           201201 non-null int64
Accident_City                   0 non-null float64
Accident_Road                   201099 non-null object
Reporting_Officer               201201 non-null int64
LandMark_First                  200150 non-null object
                                20706 non-null object
                                200713 non-null object
Distance_LandMark_Second        12839 non-null object
Junction_Second_Name            0 non-null float64
Distance_Second_Road            0 non-null float64
Research_Code                   0 non-null float64
Accident_Description            200785 non-null object
Action_Special_Note             11250 non-null object
UserID                          186728 non-null float64
Inspection_Officer_Name         201201 non-null int64
Inspection_Officer_Rank         201201 non-null int64
Inspection_Officer_Code         0 non-null float64
Actual_DateOf_Occurance         199982 non-null datetime64[ns]
FromDate_Of_Occurance           1216 non-null datetime64[ns]
ToDate_Of_Occurance             1216 non-null datetime64[ns]
Lane_Type                       201201 non-null int64
Road_Markings                   201201 non-null int64
Spot_Conditions                 201201 non-null int64
Date_Of_Filling                 201201 non-null datetime64[ns]
Highway_Patrol_No               201201 non-null int64
Aid_Post_No                     201201 non-null int64
Aid_Post_Distance               3360 non-null object
Trauma_Care_Centre_No           5580 non-null object
Trauma_Care_Centre_Distance     5723 non-null object
Checked                         201201 non-null bool
Signage                         1463 non-null object
Side_Walk                       201201 non-null int64
Modified_Unit_ID                201201 non-null int64
Action_Flag                     201201 non-null object
Sync_Id                         201201 non-null int64
Sync_Id_Old                     201201 non-null int64
NoOf_Animal_Injured             201201 non-null int64
NoOf_Animal_Killed              201201 non-null int64
DEDT                            201201 non-null datetime64[ns]
Accident_Location_id            166248 non-null float64
NoOfMinorInjured                498 non-null float64
NoOfNonVehicle                  472 non-null float64
RoadNumber                      240 non-null float64
Chainage                        36 non-null float64
SpeedLimit                      1819 non-null float64
Road_typeb                      1819 non-null float64
RoadJunction                    1819 non-null float64
Accident_spotb                  1819 non-null float64
PotHoles                        1819 non-null float64
SteepGradient                   1819 non-null float64
Footpath                        1819 non-null float64
FootBridgeSubway                1819 non-null float64
ZebraCrossing                   1819 non-null float64
Collision_Typeb                 1819 non-null float64
dtypes: bool(1), datetime64[ns](5), float64(24), int64(38), object(21)
```

Saved successfully!

```
memory usage: 135.3+ MB
```

```
data_df.columns
```

```
Index(['Unit_ID', 'Crime_No', 'Accident_DateTime', 'Fatal_NonFatal',
       'Accident_Classification', 'Severity', 'Collision_Type',
       'Accident_Spot', 'Accident_Location', 'Accident_Zone',
       'NoOfVehicle_Involved', 'Junction_Control', 'Road_Character',
       'Road_Type', 'Road_Classification', 'Road_Width', 'Shoulder_Width',
       'Seperation', 'Surface_Type', 'Surface_Condition', 'Road_Condition',
       'weather', 'Location_Type', 'Main_Cause', 'Hit_Run', 'XY_Map',
       'X_Value', 'AccidentLocation_Map', 'Collision_Diagram', 'Y_Value',
       'Route', 'KM', 'HundredM', 'Node_Map', 'Node1', 'Node2',
       'Accident_City', 'Accident_Road', 'Reporting_Officer', 'LandMark_First',
       'LandMark_Second', 'Distance_LandMark_First',
       'Distance_LandMark_Second', 'Junction_Second_Name',
       'Distance_Second_Road', 'Research_Code', 'Accident_Description',
       'Action_Special_Note', 'UserID', 'Inspection_Officer_Name',
       'Inspection_Officer_Rank', 'Inspection_Officer_Code',
       'Actual_DateOf_Occurance', 'FromDate_Of_Occurance',
       'ToDate_Of_Occurance', 'Lane_Type', 'Road_Markings', 'Spot_Conditions',
       'Date_Of_Filling', 'Highway_Patrol_No', 'Aid_Post_No',
       'Aid_Post_Distance', 'Trauma_Care_Centre_No',
       'Trauma_Care_Centre_Distance', 'Checked', 'Signage', 'Side_Walk',
       'Modified_Unit_ID', 'Action_Flag', 'Sync_Id', 'Sync_Id_Old',
       'NoOf_Animal_Injured', 'NoOf_Animal_Killed', 'DEDT',
       'Accident_Location_id', 'NoOfMinorInjured', 'NoOfNonVehicle',
       'RoadNumber', 'Chainage', 'SpeedLimit', 'Road_typeb', 'RoadJunction',
       'Accident_spotb', 'PotHoles', 'SteepGradient', 'Footpath',
       'FootBridgeSubway', 'ZebraCrossing', 'Collision_Typeb'],
      dtype='object')
```

```
test = pd.DataFrame(data_df.isna().sum())
```

```
print(data_df.info())
```

Saved successfully!

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 201201 entries, 0 to 201200
Data columns (total 89 columns):
Unit_ID                         201201 non-null int64
Crime_No                        201201 non-null int64
Accident_DateTime               0 non-null float64
Fatal_NonFatal                  172590 non-null float64
Accident_Classification         201201 non-null int64
Severity                        201201 non-null int64
Collision_Type                  201201 non-null int64
Accident_Spot                   201201 non-null int64
Accident_Location               201200 non-null object
Accident_Zone                   0 non-null float64
NoOfVehicle_Involved            201201 non-null int64
Junction_Control                201201 non-null int64
Road_Character                  201201 non-null int64
Road_Type                       201201 non-null int64
Road_Classification             201201 non-null int64
Road_Width                      45337 non-null object
Shoulder_Width                  201201 non-null int64
Seperation                      201201 non-null int64
Surface_Type                    201201 non-null int64
Surface_Condition               201201 non-null int64
Road_Condition                  201201 non-null int64
weather                         201201 non-null int64
Location_Type                   201201 non-null int64
Main_Cause                      201201 non-null int64
Hit_Run                         201201 non-null int64
XY_Map                          1994 non-null object
X_Value                         1852 non-null object
AccidentLocation_Map            9641 non-null object
Collision_Diagram               523 non-null object
Y_Value                         1813 non-null object
Route                           4554 non-null object
KM                              6409 non-null object
HundredM                        201201 non-null int64
Node_Map                        201201 non-null int64
Node1                           201201 non-null int64
Node2                           201201 non-null int64
Accident_City                   0 non-null float64
Accident_Road                   201099 non-null object
Reporting_Officer               201201 non-null int64
LandMark_First                  200150 non-null object
                                20706 non-null object
                                200713 non-null object
Distance_LandMark_Second        12839 non-null object
Junction_Second_Name            0 non-null float64
Distance_Second_Road            0 non-null float64
Research_Code                   0 non-null float64
Accident_Description            200785 non-null object
Action_Special_Note             11250 non-null object
UserID                          186728 non-null float64
Inspection_Officer_Name         201201 non-null int64
Inspection_Officer_Rank         201201 non-null int64
Inspection_Officer_Code         0 non-null float64
Actual_DateOf_Occurance         199982 non-null datetime64[ns]
FromDate_Of_Occurance           1216 non-null datetime64[ns]
ToDate_Of_Occurance             1216 non-null datetime64[ns]
Lane_Type                       201201 non-null int64
Road_Markings                   201201 non-null int64
Spot_Conditions                 201201 non-null int64
Date_Of_Filling                 201201 non-null datetime64[ns]
Highway_Patrol_No               201201 non-null int64
Aid_Post_No                     201201 non-null int64
Aid_Post_Distance               3360 non-null object
Trauma_Care_Centre_No           5580 non-null object
Trauma_Care_Centre_Distance     5723 non-null object
Checked                         201201 non-null bool
Signage                         1463 non-null object
Side_Walk                       201201 non-null int64
Modified_Unit_ID                201201 non-null int64
Action_Flag                     201201 non-null object
Sync_Id                         201201 non-null int64
Sync_Id_Old                     201201 non-null int64
NoOf_Animal_Injured             201201 non-null int64
NoOf_Animal_Killed              201201 non-null int64
DEDT                            201201 non-null datetime64[ns]
Accident_Location_id            166248 non-null float64
NoOfMinorInjured                498 non-null float64
NoOfNonVehicle                  472 non-null float64
RoadNumber                      240 non-null float64
Chainage                        36 non-null float64
SpeedLimit                      1819 non-null float64
Road_typeb                      1819 non-null float64
RoadJunction                    1819 non-null float64
Accident_spotb                  1819 non-null float64
PotHoles                        1819 non-null float64
SteepGradient                   1819 non-null float64
Footpath                        1819 non-null float64
FootBridgeSubway                1819 non-null float64
ZebraCrossing                   1819 non-null float64
Collision_Typeb                 1819 non-null float64
dtypes: bool(1), datetime64[ns](5), float64(24), int64(38), object(21)
```

Saved successfully!

```
memory usage: 135.3+ MB
None
```

```
data_df['Severity'].value_counts()
```

```
6112     78105
6113     60080
6111     35958
6114     18970
6503      8082
0            2
5101         1
7908         1
9787         1
11012        1
Name: Severity, dtype: int64
```

```
def missing_values_table(df):
        # Total missing values
        mis_val = df.isnull().sum()

        # Percentage of missing values
        mis_val_percent = 100 * df.isnull().sum() / len(df)

        # Make a table with the results
        mis_val_table = pd.concat([mis_val, mis_val_percent], axis=1)

        # Rename the columns
        mis_val_table_ren_columns = mis_val_table.rename(
        columns = {0 : 'Missing Values', 1 : '% of Total Values'})

        # Sort the table by percentage of missing descending
        mis_val_table_ren_columns = mis_val_table_ren_columns[
            mis_val_table_ren_columns.iloc[:,1] != 0].sort_values(
        '% of Total Values', ascending=False).round(1)

        # Print some summary information
        print ("Your selected dataframe has " + str(df.shape[1]) + " columns.\n"
                                              al_table_ren_columns.shape[0]) +
                                              ssing values.")

        # Return the dataframe with missing information
        return mis_val_table_ren_columns
```

Saved successfully!

```
missing_values_table(data_df)
```

Your selected dataframe has 89 columns.
There are 47 columns that have missing values.

	Missing Values	% of Total Values
Accident_DateTime	201201	100.0
Junction_Second_Name	201201	100.0
Research_Code	201201	100.0
Distance_Second_Road	201201	100.0
Accident_City	201201	100.0
Inspection_Officer_Code	201201	100.0
Accident_Zone	201201	100.0
Chainage	201165	100.0
RoadNumber	200961	99.9
NoOfNonVehicle	200729	99.8
NoOfMinorInjured	200703	99.8
Collision_Diagram	200678	99.7
FromDate_Of_Occurance	199985	99.4
ToDate_Of_Occurance	199985	99.4
Signage	199738	99.3
Y_Value	199388	99.1
Road_typeb	199382	99.1
SpeedLimit	199382	99.1
SteepGradient	199382	99.1
RoadJunction	199382	99.1
FootBridgeSubway	199382	99.1
	199382	99.1
	199382	99.1
Accident_spotb	199382	99.1
Footpath	199382	99.1
Collision_Typeb	199382	99.1
X_Value	199349	99.1
XY_Map	199207	99.0
Aid_Post_Distance	197841	98.3
Route	196647	97.7
Trauma_Care_Centre_No	195621	97.2
Trauma_Care_Centre_Distance	195478	97.2
KM	194792	96.8
AccidentLocation_Map	191560	95.2
Action_Special_Note	189951	94.4
Distance_LandMark_Second	188362	93.6
LandMark_Second	180495	89.7
Road_Width	155864	77.5
Accident_Location_id	34953	17.4
Fatal_NonFatal	28611	14.2
UserID	14473	7.2
Actual_DateOf_Occurance	1219	0.6
LandMark_First	1051	0.5
Distance_LandMark_First	488	0.2
Accident_Description	416	0.2
Accident_Road	102	0.1
Accident_Location	1	0.0

Saved successfully!

```
drop_cols = ['Footpath'
```

```
drop_cols = ['Footpath',
'FootBridgeSubway',
'ZebraCrossing',
'Road_typeb',
'Collision_Typeb',
'Y_Value',
'X_Value',
'XY_Map',
'Aid_Post_Distance',
'Route',
'Trauma_Care_Centre_No',
'Trauma_Care_Centre_Distance',
'KM',
'AccidentLocation_Map',
'Action_Special_Note',
'Distance_LandMark_Second',
'LandMark_Second',
'Road_Width',
'Accident_DateTime',
'Accident_Zone',
'Accident_City',
'Junction_Second_Name',
'Distance_Second_Road',
'Research_Code',
'Inspection_Officer_Code',
]
```

```
data_df.drop(drop_cols, axis=1, inplace=True)
```

```
data_df.shape
```

```
(201201, 64)
```

```
data_df.dtypes.value_counts()
```

```
int64             38
float64           12
dtype: int64
```

Saved successfully!

```
data_df.describe()
```

	Unit_ID	Crime_No	Fatal_NonFatal	Accident_Classification	Severity	Collision_Type	Accident_Spot	NoOfVehicle_Invo
count	201201.000000	2.012010e+05	172590.0	201201.000000	201201.000000	201201.000000	201201.000000	201201.00
mean	1186.131028	2.016045e+09	0.0	6287.527149	6128.000219	6732.037743	6286.784549	1.55
std	599.210953	1.461009e+06	0.0	60.626527	80.404165	1617.104603	285.222802	0.61
min	57.000000	2.014000e+09	0.0	0.000000	0.000000	0.000000	0.000000	1.00
25%	726.000000	2.015000e+09	0.0	6274.000000	6112.000000	6091.000000	6083.000000	1.00
50%	1203.000000	2.016000e+09	0.0	6274.000000	6112.000000	6093.000000	6083.000000	2.00
75%	1761.000000	2.017000e+09	0.0	6274.000000	6113.000000	6525.000000	6719.000000	2.00
max	2245.000000	2.019000e+09	0.0	6514.000000	11012.000000	20104.000000	6810.000000	64.00

```
missing_values_table(data_df)
```

```
Your selected dataframe has 64 columns.
There are 22 columns that have missing values.
```

	Missing Values	% of Total Values
Chainage	201165	100.0
RoadNumber	200961	99.9
NoOfNonVehicle	200729	99.8
NoOfMinorInjured	200703	99.8
Collision_Diagram	200678	99.7
FromDate_Of_Occurance	199985	99.4
ToDate_Of_Occurance	199985	99.4
Signage	199738	99.3
PotHoles	199382	99.1
Accident_spotb	199382	99.1
RoadJunction	199382	99.1
SpeedLimit	199382	99.1
SteepGradient	199382	99.1
Accident_Location_id	34953	17.4
Fatal_NonFatal	28611	14.2
UserID	14473	7.2
Actual_DateOf_Occurance	1219	0.6
LandMark_First	1051	0.5
Distance_LandMark_First	488	0.2
Accident_Description	416	0.2
Accident_Road	102	0.1
	1	0.0

Saved successfully!

```
drop_cols =['Chainage',
'RoadNumber',
'NoOfNonVehicle',
'NoOfMinorInjured',
'Collision_Diagram',
'FromDate_Of_Occurance',
'ToDate_Of_Occurance',
'Signage',
'PotHoles',
'Accident_spotb',
'RoadJunction',
'SpeedLimit',
'SteepGradient']
```

```
data_df.drop(drop_cols, axis=1, inplace=True)
```

```
data_df.columns
```

```
Index(['Unit_ID', 'Crime_No', 'Fatal_NonFatal', 'Accident_Classification',
       'Severity', 'Collision_Type', 'Accident_Spot', 'Accident_Location',
       'NoOfVehicle_Involved', 'Junction_Control', 'Road_Character',
       'Road_Type', 'Road_Classification', 'Shoulder_Width', 'Seperation',
       'Surface_Type', 'Surface_Condition', 'Road_Condition', 'weather',
       'Location_Type', 'Main_Cause', 'Hit_Run', 'HundredM', 'Node_Map',
       'Node1', 'Node2', 'Accident_Road', 'Reporting_Officer',
       'LandMark_First', 'Distance_LandMark_First', 'Accident_Description',
       'UserID', 'Inspection_Officer_Name', 'Inspection_Officer_Rank',
       'Actual_DateOf_Occurance', 'Lane_Type', 'Road_Markings',
       'Spot_Conditions', 'Date_Of_Filling', 'Highway_Patrol_No',
       'Aid_Post_No', 'Checked', 'Side_Walk', 'Modified_Unit_ID',
       'Action_Flag', 'Sync_Id', 'Sync_Id_Old', 'NoOf_Animal_Injured',
       'NoOf_Animal_Killed', 'DEDT', 'Accident_Location_id'],
      dtype='object')
```

```
drop_cols =[ 'Unit_ID', 'Crime_No','Accident_Location_id','Node_Map','Reporting_Officer','LandMark_First', 'Distance_LandMark_First', 'Accident
       'Actual_DateOf_Occurance','Date_Of_Filling','Modified_Unit_ID','Aid_Post_No','Action_Flag','Sync_Id', 'Sync_Id_Old']
```

```
data_df.drop(drop_cols, axis=1, inplace=True)
```

```
data_df.shape
```

```
(201201, 33)
```

```
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 201201 entries, 0 to 201200
Data columns (total 33 columns):
Fatal_NonFatal             172590 non-null float64
Accident_Classification    201201 non-null int64
Severity                   201201 non-null int64
Collision_Type             201201 non-null int64
Accident_Spot              201201 non-null int64
Accident_Location          201200 non-null object
NoOfVehicle_Involved       201201 non-null int64
Junction_Control           201201 non-null int64
Road_Character             201201 non-null int64
Road_Type                  201201 non-null int64
Road_Classification        201201 non-null int64
Shoulder_Width             201201 non-null int64
Seperation                 201201 non-null int64
Surface_Type               201201 non-null int64
Surface_Condition          201201 non-null int64
Road_Condition             201201 non-null int64
weather                    201201 non-null int64
Location_Type              201201 non-null int64
Main_Cause                 201201 non-null int64
Hit_Run                    201201 non-null int64
HundredM                   201201 non-null int64
Node1                      201201 non-null int64
Node2                      201201 non-null int64
Accident_Road              201099 non-null object
Lane_Type                  201201 non-null int64
Road_Markings              201201 non-null int64
Spot_Conditions            201201 non-null int64
Highway_Patrol_No          201201 non-null int64
Checked                    201201 non-null bool
Side_Walk                  201201 non-null int64
NoOf_Animal_Injured        201201 non-null int64
NoOf_Animal_Killed         201201 non-null int64
DEDT                       201201 non-null datetime64[ns]
dtypes: bool(1), datetime64[ns](1), float64(1), int64(28), object(2)
```

Saved successfully!

```
data_df.drop('Checked', axis=1, inplace=True)
```

```
missing_values_table(data_df)
data_df.drop('Fatal_NonFatal', axis=1, inplace=True)
data_df.drop('DEDT', axis=1, inplace=True)
```

```
Your selected dataframe has 32 columns.
There are 3 columns that have missing values.
```

```
data_df = data_df.dropna( )
```

Double-click (or enter) to edit

```
data_df.Severity[data_df.Severity == 6112] = 1
data_df.Severity[data_df.Severity == 6113] = 0
data_df.Severity[data_df.Severity == 6111] = 1
data_df.Severity[data_df.Severity == 6114] = 0
data_df.Severity[data_df.Severity == 6503] = 0
data_df.Severity[data_df.Severity == 9787] = 0
data_df.Severity[data_df.Severity == 7908] = 0
data_df.Severity[data_df.Severity == 11012] = 0
data_df.Severity[data_df.Severity == 5101] = 1
```

```
data_df['Severity'].value_counts()
```

```
1    114013
0     87086
Name: Severity, dtype: int64
```

```
numeric_data=data_df.select_dtypes(include=[np.number])
num_list = list(numeric_data.columns)
num_list.remove('Severity')
```

```
num_list
```

```
['Accident_Classification',
 'Collision_Type',
 'Accident_Spot',
 'NoOfVehicle_Involved',
 'Junction_Control',
 'Road_Character',
 'Road_Type',
 'Road_Classification',
 'Shoulder_Width',
 'Seperation',
 'Surface_Type',
 'Surface_Condition',
 'Road_Condition',
 'weather',
 'Location_Type',
 'Main_Cause',
 'Hit_Run',
 'HundredM',
 'Node1',
 'Node2',
 'Lane_Type',
 'Road_Markings',
 'Spot_Conditions',
 'Highway_Patrol_No',
 'Side_Walk',
 'NoOf_Animal_Injured',
 'NoOf_Animal_Killed']
```

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scaler.fit(data_df[num_list])
scaled = scaler.transform(data_df[num_list])
for i, col in enumerate(num_list):
  data_df[col] = scaled[:,i]
```

```
data_df.drop_duplicates(keep=False,inplace=True)
```

```
data_df = data_df.dropna(axis = 0, how ='any')
```

Saved successfully!

```
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 193784 entries, 0 to 201200
Data columns (total 30 columns):
Accident_Classification    193784 non-null float64
Severity                   193784 non-null int64
Collision_Type             193784 non-null float64
Accident_Spot              193784 non-null float64
Accident_Location          193784 non-null object
NoOfVehicle_Involved       193784 non-null float64
Junction_Control           193784 non-null float64
Road_Character             193784 non-null float64
Road_Type                  193784 non-null float64
Road_Classification        193784 non-null float64
Shoulder_Width             193784 non-null float64
Seperation                 193784 non-null float64
Surface_Type               193784 non-null float64
Surface_Condition          193784 non-null float64
Road_Condition             193784 non-null float64
weather                    193784 non-null float64
Location_Type              193784 non-null float64
Main_Cause                 193784 non-null float64
Hit_Run                    193784 non-null float64
HundredM                   193784 non-null float64
Node1                      193784 non-null float64
Node2                      193784 non-null float64
Accident_Road              193784 non-null object
Lane_Type                  193784 non-null float64
Road_Markings              193784 non-null float64
Spot_Conditions            193784 non-null float64
Highway_Patrol_No          193784 non-null float64
Side_Walk                  193784 non-null float64
NoOf_Animal_Injured        193784 non-null float64
NoOf_Animal_Killed         193784 non-null float64
dtypes: float64(27), int64(1), object(2)
memory usage: 45.8+ MB
```

```
features = ['Accident_Classification','NoOfVehicle_Involved','HundredM','Node1','Node2',  'Road_Markings','NoOf_Animal_Injured','NoOf_Animal_Ki
target = data_df.loc[:, data_df.columns == 'Severity'].columns
```

```
features
```

```
['Accident_Classification',
 'NoOfVehicle_Involved',
 'HundredM',
 'Node1',
 'Node2',
 'Road_Markings',
 'NoOf_Animal_Injured',
 'NoOf_Animal_Killed']
```

```
data_df.shape
```

```
(193784, 30)
```

```
X_train, y_train = data_df.iloc[:155027][features], data_df.iloc[:155027][target]
X_val, y_val = data_df.iloc[155027:193784][features], data_df.iloc[155027:193784][target]
X_test = data_df.iloc[193684:][features]
```

```
X_train2 = X_train.copy()
```

```
from sklearn.datasets import make_friedman1
from sklearn.feature_selection import RFE
from sklearn.svm import SVR
estimator = SVR(kernel="linear")
selector = RFE(estimator, 8, step=200)
selector = selector.fit(X_train2, y_train)
selector.support_
```

```
array([ True,  True, False, False, False, False,  True,  True,  True,
        True, False, False,  True,  True])
```

```
pd.DataFrame([selector.support_ ,X_train2.columns]).to_csv("top.csv")
```

```
X_train2.columns
```

```
Index(['Accident_Classification', 'Collision_Type', 'Accident_Spot',
       'NoOfVehicle_Involved', 'Junction_Control',
       'Type', 'Road_Classification', 'Shoulder_Width',
       'Seperation', 'Surface_Type', 'Surface_Condition', 'Road_Condition',
       'weather', 'Location_Type', 'Main_Cause', 'Hit_Run', 'HundredM',
       'Node1', 'Node2', 'Accident_Road', 'Lane_Type', 'Road_Markings',
       'Spot_Conditions', 'Highway_Patrol_No', 'Side_Walk',
       'NoOf_Animal_Injured', 'NoOf_Animal_Killed'],
      dtype='object')
```



```
%load_ext tensorboard
import datetime as datetime
logdir="gdrive/My Drive/Colab Notebooks/dataset/TataTele Services/logs/scalars8/" + datetime.datetime.now().strftime("%Y%m%d-%H%M%S")
tensorboard_callback = keras.callbacks.TensorBoard(log_dir=logdir)
```

```
The tensorboard extension is already loaded. To reload it, use:
  %reload_ext tensorboard
```

```
from numpy import loadtxt
from keras.models import Sequential
from keras.layers import Dense
# load the dataset
# define the keras model
model = Sequential()
model.add(Dense(100, input_dim=8, activation='relu'))
model.add(Dense(100, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
# compile the keras model
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
# fit the keras model on the dataset
#model.fit(X_train, y_train, epochs=150, batch_size=10, verbose=0)
history  =  model.fit(X_train,y_train,validation_data=(X_val,y_val) , epochs =  100 , batch_size = 100, verbose= 2,callbacks=[tensorboard_callb
# make class predictions with the model
predictions = model.predict_classes(X_val)
```

```
Train on 155027 samples, validate on 38757 samples
Epoch 1/100
 - 3s - loss: 0.6822 - acc: 0.5656 - val_loss: 0.6743 - val_acc: 0.5892
Epoch 2/100
 - 3s - loss: 0.6796 - acc: 0.5706 - val_loss: 0.6684 - val_acc: 0.5915
Epoch 3/100
 - 3s - loss: 0.6683 - acc: 0.5970 - val_loss: 0.6709 - val_acc: 0.5976
Epoch 4/100
 - 3s - loss: 0.6341 - acc: 0.6062 - val_loss: 0.6074 - val_acc: 0.6031
Epoch 5/100
 - 3s - loss: 0.6105 - acc: 0.6070 - val_loss: 0.5898 - val_acc: 0.6387
Epoch 6/100
 - 3s - loss: 0.6044 - acc: 0.6091 - val_loss: 0.5875 - val_acc: 0.6125
Epoch 7/100
 - 3s - loss: 0.6000 - acc: 0.6186 - val_loss: 0.5848 - val_acc: 0.6398
Epoch 8/100
 - 3s - loss: 0.5988 - acc: 0.6220 - val_loss: 0.5797 - val_acc: 0.6608
Epoch 9/100
 - 3s - loss: 0.5972 - acc: 0.6266 - val_loss: 0.5811 - val_acc: 0.6418
Epoch 10/100
 - 3s - loss: 0.5969 - acc: 0.6291 - val_loss: 0.5835 - val_acc: 0.6469
Epoch 11/100
 - 3s - loss: 0.5946 - acc: 0.6318 - val_loss: 0.5779 - val_acc: 0.6588
Epoch 12/100
 - 3s - loss: 0.5951 - acc: 0.6327 - val_loss: 0.5783 - val_acc: 0.6566
Epoch 13/100
 - 3s - loss: 0.5941 - acc: 0.6316 - val_loss: 0.5930 - val_acc: 0.6515
Epoch 14/100
 - 3s - loss: 0.5937 - acc: 0.6346 - val_loss: 0.5803 - val_acc: 0.6582
Epoch 15/100
 - 3s - loss: 0.5923 - acc: 0.6352 - val_loss: 0.6130 - val_acc: 0.6161
Epoch 16/100
 - 3s - loss: 0.5921 - acc: 0.6371 - val_loss: 0.5764 - val_acc: 0.6598
Epoch 17/100
 - 3s - loss: 0.5920 - acc: 0.6361 - val_loss: 0.5852 - val_acc: 0.6443
Epoch 18/100
 - 3s - loss: 0.5919 - acc: 0.6365 - val_loss: 0.5770 - val_acc: 0.6474
Epoch 19/100
 - 3s - loss: 0.5906 - acc: 0.6381 - val_loss: 0.5813 - val_acc: 0.6441
Epoch 20/100
 - 3s - loss: 0.5923 - acc: 0.6387 - val_loss: 0.5872 - val_acc: 0.6345
Epoch 21/100
 - 3s - loss: 0.5914 - acc: 0.6380 - val_loss: 0.5777 - val_acc: 0.6585
Epoch 22/100
 - 3s - loss: 0.5900 - acc: 0.6393 - val_loss: 0.5784 - val_acc: 0.6445
Epoch 23/100
 - 3s - loss: 0.5896 - acc: 0.6398 - val_loss: 0.5813 - val_acc: 0.6434
Epoch 24/100
 - 3s - loss: 0.5904 - acc: 0.6396 - val_loss: 0.5858 - val_acc: 0.6337
Epoch 25/100
 - 3s - loss: 0.5909 - acc: 0.6390 - val_loss: 0.5776 - val_acc: 0.6601
Epoch 26/100
 - 3s - loss: 0.5889 - acc: 0.6396 - val_loss: 0.5720 - val_acc: 0.6673
Epoch 27/100
 - 3s - loss: 0.5885 - acc: 0.6410 - val_loss: 0.5794 - val_acc: 0.6447
Epoch 28/100
 - 3s - loss: 0.5886 - acc: 0.6408 - val_loss: 0.5726 - val_acc: 0.6592
Epoch 29/100
 - 3s - loss: 0.5874 - acc: 0.6412 - val_loss: 0.5717 - val_acc: 0.6623
Epoch 30/100
 - 3s - loss: 0.5883 - acc: 0.6397 - val_loss: 0.5764 - val_acc: 0.6610
Epoch 31/100
 - 3s - loss: 0.5881 - acc: 0.6406 - val_loss: 0.5720 - val_acc: 0.6583
Epoch 32/100
 - 3s - loss: 0.5877 - acc: 0.6404 - val_loss: 0.5737 - val_acc: 0.6517
Epoch 33/100
 - 3s - loss: 0.5864 - acc: 0.6417 - val_loss: 0.5710 - val_acc: 0.6672
Epoch 34/100
 - 3s - loss: 0.5868 - acc: 0.6408 - val_loss: 0.5767 - val_acc: 0.6519
Epoch 35/100
 - 3s - loss: 0.5867 - acc: 0.6414 - val_loss: 0.5734 - val_acc: 0.6685
Epoch 36/100
 - 3s - loss: 0.5862 - acc: 0.6414 - val_loss: 0.5743 - val_acc: 0.6574
Epoch 37/100
 - 3s - loss: 0.5858 - acc: 0.6419 - val_loss: 0.5793 - val_acc: 0.6616
Epoch 38/100
 - 3s - loss: 0.5856 - acc: 0.6417 - val_loss: 0.5774 - val_acc: 0.6633
Epoch 39/100
 - 3s - loss: 0.5842 - acc: 0.6422 - val_loss: 0.5699 - val_acc: 0.6658
Epoch 40/100
 - 3s - loss: 0.5848 - acc: 0.6425 - val_loss: 0.5734 - val_acc: 0.6573
Epoch 41/100
 - 3s - loss: 0.5837 - acc: 0.6425 - val_loss: 0.5700 - val_acc: 0.6628
Epoch 42/100
 - 3s - loss: 0.5837 - acc: 0.6425 - val_loss: 0.5701 - val_acc: 0.6633
Epoch 43/100
 - 3s - loss: 0.5836 - acc: 0.6437 - val_loss: 0.5754 - val_acc: 0.6512
Epoch 44/100
 - 3s - loss: 0.5841 - acc: 0.6431 - val_loss: 0.5679 - val_acc: 0.6688
Epoch 45/100
 - 3s - loss: 0.5850 - acc: 0.6428 - val_loss: 0.6142 - val_acc: 0.6623
Epoch 46/100
 - 3s - loss: 0.5921 - acc: 0.6403 - val_loss: 0.5697 - val_acc: 0.6656
```

Saved successfully!

```
Epoch 47/100
 - 3s - loss: 0.5832 - acc: 0.6417 - val_loss: 0.5753 - val_acc: 0.6654
Epoch 48/100
 - 3s - loss: 0.5830 - acc: 0.6419 - val_loss: 0.5671 - val_acc: 0.6660
Epoch 49/100
 - 3s - loss: 0.5822 - acc: 0.6435 - val_loss: 0.5717 - val_acc: 0.6340
Epoch 50/100
 - 3s - loss: 0.5822 - acc: 0.6430 - val_loss: 0.5732 - val_acc: 0.6579
Epoch 51/100
 - 3s - loss: 0.5831 - acc: 0.6428 - val_loss: 0.5689 - val_acc: 0.6633
Epoch 52/100
 - 3s - loss: 0.5821 - acc: 0.6434 - val_loss: 0.5713 - val_acc: 0.6677
Epoch 53/100
 - 3s - loss: 0.5823 - acc: 0.6439 - val_loss: 0.5674 - val_acc: 0.6677
Epoch 54/100
 - 3s - loss: 0.5818 - acc: 0.6437 - val_loss: 0.5702 - val_acc: 0.6633
Epoch 55/100
 - 3s - loss: 0.5816 - acc: 0.6440 - val_loss: 0.5668 - val_acc: 0.6668
Epoch 56/100
 - 3s - loss: 0.5814 - acc: 0.6443 - val_loss: 0.5691 - val_acc: 0.6633
Epoch 57/100
 - 3s - loss: 0.5820 - acc: 0.6444 - val_loss: 0.5664 - val_acc: 0.6688
Epoch 58/100
 - 3s - loss: 0.5818 - acc: 0.6435 - val_loss: 0.5697 - val_acc: 0.6534
Epoch 59/100
 - 3s - loss: 0.5814 - acc: 0.6440 - val_loss: 0.5654 - val_acc: 0.6668
Epoch 60/100
 - 3s - loss: 0.5820 - acc: 0.6446 - val_loss: 0.5668 - val_acc: 0.6669
Epoch 61/100
 - 3s - loss: 0.5809 - acc: 0.6447 - val_loss: 0.5684 - val_acc: 0.6675
Epoch 62/100
 - 3s - loss: 0.5824 - acc: 0.6442 - val_loss: 0.5677 - val_acc: 0.6680
Epoch 63/100
 - 3s - loss: 0.5815 - acc: 0.6448 - val_loss: 0.5679 - val_acc: 0.6639
Epoch 64/100
 - 3s - loss: 0.5816 - acc: 0.6446 - val_loss: 0.5661 - val_acc: 0.6663
Epoch 65/100
 - 3s - loss: 0.5814 - acc: 0.6447 - val_loss: 0.5670 - val_acc: 0.6678
Epoch 66/100
 - 3s - loss: 0.5812 - acc: 0.6451 - val_loss: 0.5648 - val_acc: 0.6674
Epoch 67/100
 - 3s - loss: 0.5809 - acc: 0.6447 - val_loss: 0.5670 - val_acc: 0.6679
Epoch 68/100
```

Saved successfully!

```
448 - val_loss: 0.5670 - val_acc: 0.6674
 - 3s - loss: 0.5838 - acc: 0.6441 - val_loss: 0.5734 - val_acc: 0.6572
Epoch 70/100
 - 3s - loss: 0.5819 - acc: 0.6442 - val_loss: 0.5683 - val_acc: 0.6672
Epoch 71/100
 - 3s - loss: 0.5810 - acc: 0.6452 - val_loss: 0.5656 - val_acc: 0.6676
Epoch 72/100
 - 3s - loss: 0.5805 - acc: 0.6448 - val_loss: 0.5684 - val_acc: 0.6676
Epoch 73/100
 - 3s - loss: 0.5819 - acc: 0.6450 - val_loss: 0.5715 - val_acc: 0.6615
Epoch 74/100
 - 3s - loss: 0.5824 - acc: 0.6446 - val_loss: 0.5663 - val_acc: 0.6666
Epoch 75/100
 - 3s - loss: 0.5804 - acc: 0.6456 - val_loss: 0.5713 - val_acc: 0.6595
Epoch 76/100
 - 3s - loss: 0.5805 - acc: 0.6457 - val_loss: 0.5757 - val_acc: 0.6576
Epoch 77/100
 - 3s - loss: 0.5831 - acc: 0.6446 - val_loss: 0.5668 - val_acc: 0.6589
Epoch 78/100
 - 3s - loss: 0.5812 - acc: 0.6448 - val_loss: 0.5661 - val_acc: 0.6675
Epoch 79/100
 - 3s - loss: 0.5803 - acc: 0.6453 - val_loss: 0.5653 - val_acc: 0.6675
Epoch 80/100
 - 3s - loss: 0.5819 - acc: 0.6444 - val_loss: 0.5691 - val_acc: 0.6498
Epoch 81/100
 - 3s - loss: 0.5821 - acc: 0.6450 - val_loss: 0.5993 - val_acc: 0.6668
Epoch 82/100
 - 3s - loss: 0.5843 - acc: 0.6444 - val_loss: 0.5688 - val_acc: 0.6679
Epoch 83/100
 - 3s - loss: 0.5807 - acc: 0.6453 - val_loss: 0.5670 - val_acc: 0.6674
Epoch 84/100
 - 3s - loss: 0.5803 - acc: 0.6456 - val_loss: 0.5708 - val_acc: 0.6589
Epoch 85/100
 - 3s - loss: 0.5807 - acc: 0.6447 - val_loss: 0.5689 - val_acc: 0.6643
Epoch 86/100
 - 3s - loss: 0.5805 - acc: 0.6450 - val_loss: 0.5722 - val_acc: 0.6628
Epoch 87/100
 - 3s - loss: 0.5811 - acc: 0.6451 - val_loss: 0.5643 - val_acc: 0.6674
Epoch 88/100
 - 3s - loss: 0.5811 - acc: 0.6438 - val_loss: 0.5737 - val_acc: 0.6523
Epoch 89/100
 - 3s - loss: 0.5801 - acc: 0.6448 - val_loss: 0.5653 - val_acc: 0.6672
Epoch 90/100
 - 3s - loss: 0.5802 - acc: 0.6455 - val_loss: 0.5651 - val_acc: 0.6673
Epoch 91/100
 - 3s - loss: 0.5800 - acc: 0.6455 - val_loss: 0.5670 - val_acc: 0.6637
Epoch 92/100
 - 3s - loss: 0.5803 - acc: 0.6448 - val_loss: 0.5737 - val_acc: 0.6634
Epoch 93/100
```

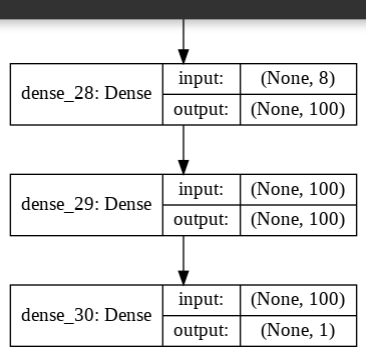
```
 - 3s - loss: 0.5812 - acc: 0.6452 - val_loss: 0.5646 - val_acc: 0.6682
Epoch 94/100
 - 3s - loss: 0.5803 - acc: 0.6458 - val_loss: 0.5827 - val_acc: 0.6611
Epoch 95/100
 - 3s - loss: 0.5808 - acc: 0.6440 - val_loss: 0.5687 - val_acc: 0.6670
Epoch 96/100
 - 3s - loss: 0.5808 - acc: 0.6442 - val_loss: 0.5648 - val_acc: 0.6675
Epoch 97/100
 - 3s - loss: 0.5810 - acc: 0.6439 - val_loss: 0.5699 - val_acc: 0.6559
Epoch 98/100
 - 3s - loss: 0.5808 - acc: 0.6442 - val_loss: 0.5656 - val_acc: 0.6681
Epoch 99/100
 - 3s - loss: 0.5800 - acc: 0.6453 - val_loss: 0.5653 - val_acc: 0.6670
Epoch 100/100
 - 3s - loss: 0.5799 - acc: 0.6455 - val_loss: 0.5664 - val_acc: 0.6551
```

```
print(model.summary())
```

```
Model: "sequential_11"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_28 (Dense)             (None, 100)               900
_________________________________________________________________
dense_29 (Dense)             (None, 100)               10100
_________________________________________________________________
dense_30 (Dense)             (None, 1)                 101
=================================================================
Total params: 11,101
Trainable params: 11,101
Non-trainable params: 0
_________________________________________________________________
None
```

```
import pydot_ng as pydot
```

```
plot_model(model, show_shapes=True, show_layer_names=True, to_file='model.png')
from IPython.display import Image
Image(retina=True, filename='model.png')
```


```
model.summary()
```

```
Model: "sequential_11"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_28 (Dense)             (None, 100)               900
_________________________________________________________________
dense_29 (Dense)             (None, 100)               10100
_________________________________________________________________
dense_30 (Dense)             (None, 1)                 101
=================================================================
Total params: 11,101
Trainable params: 11,101
Non-trainable params: 0
_________________________________________________________________
```

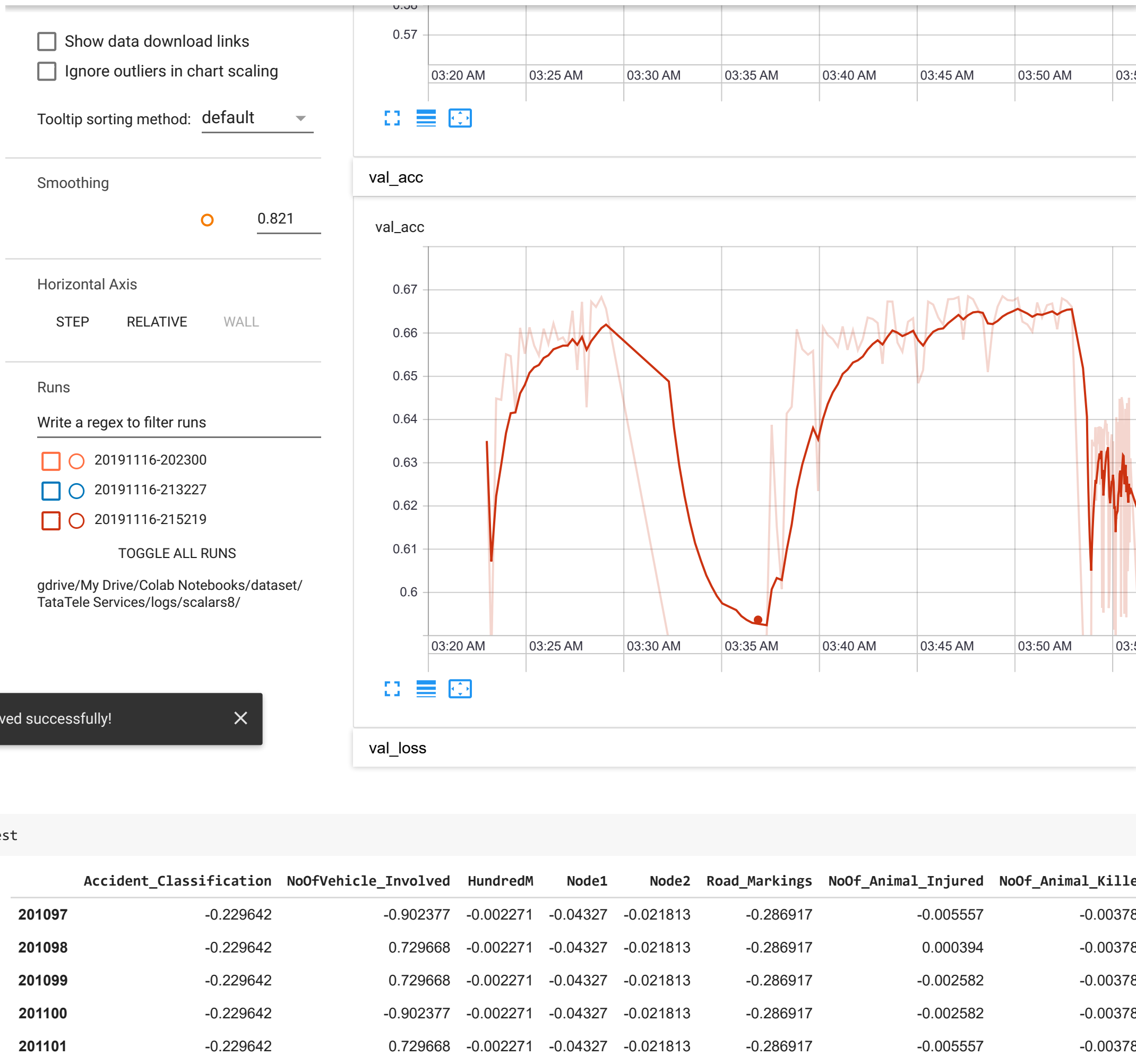
```
%load_ext tensorboard
import datetime as datetime
logdir="gdrive/My Drive/Colab Notebooks/dataset/TataTele Services/logs/scalars8/" + datetime.datetime.now().strftime("%Y%m%d-%H%M%S")
tensorboard_callback = keras.callbacks.TensorBoard(log_dir=logdir)
```

```
The tensorboard extension is already loaded. To reload it, use:
  %reload_ext tensorboard
```

```
model.save("gdrive/My Drive/Colab Notebooks/dataset/sapientmodel.h5")
```

```
%tensorboard --logdir "gdrive/My Drive/Colab Notebooks/dataset/TataTele Services/logs/scalars8/"
```

```
Reusing TensorBoard on port 6006 (pid 2250), started 0:00:13 ago. (Use '!kill 2250' to kill it.)
```




```
X_test
```

	Accident_Classification	NoOfVehicle_Involved	HundredM	Node1	Node2	Road_Markings	NoOf_Animal_Injured	NoOf_Animal_Kille
201097	-0.229642	-0.902377	-0.002271	-0.04327	-0.021813	-0.286917	-0.005557	-0.00378
201098	-0.229642	0.729668	-0.002271	-0.04327	-0.021813	-0.286917	0.000394	-0.00378
201099	-0.229642	0.729668	-0.002271	-0.04327	-0.021813	-0.286917	-0.002582	-0.00378
201100	-0.229642	-0.902377	-0.002271	-0.04327	-0.021813	-0.286917	-0.002582	-0.00378
201101	-0.229642	0.729668	-0.002271	-0.04327	-0.021813	-0.286917	-0.005557	-0.00378
...	...	...	...	...	...	...	...	
201195	-0.229642	0.729668	-0.001511	-0.04327	-0.021813	3.448659	-0.005557	0.00024
201196	-0.229642	-0.902377	-0.002271	-0.04327	-0.021813	-0.286917	-0.005557	-0.00378
201197	-0.229642	0.729668	-0.002271	-0.04327	-0.021813	-0.286917	-0.002582	-0.00378
201198	-0.229642	0.729668	-0.002271	-0.04327	-0.021813	-0.286917	-0.005557	-0.00378
201200	-0.229642	0.729668	-0.002271	-0.04327	-0.021813	-0.286917	-0.002582	-0.00378

100 rows × 8 columns

```
model.predict_classes(X_test)
```

```
[0],
[1],
[0],
[1],
[0],
[1],
[1],
[0],
[1],
[1],
[1],
[0],
[1],
[1],
[0],
[0],
[1],
[1],
[1],
[0],
[1],
[0],
[0],
[0],
[1],
[1],
[1],
[1],
[0],
[0],
[0],
[1],
[1],
[0],
[1],
[1],
[0],
[0],
[1],
[1],
[1],
[1],
[1],
[0],
[0],
[0],
[1],
[1],
[1],
[1],
[0],
[1],
[1],
[0],
[0],
[0],
[0],
[1],
[0],
[0],
[0],
[0]], dtype=int32)
```

Saved successfully!