In [ ]:
```
# imports
import pandas as pd
import numpy as np
import joblib
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import RandomForestClassifier

# Dependencies: pandas, numpy, joblib(for pickle saving), scikit-learn
```

In [ ]:
```
# load in train,test,val data
test = pd.read_csv('../data/processed/test_data_processed.csv')
train = pd.read_csv('../data/processed/train_data_processed.csv')
val = pd.read_csv('../data/processed/val_data_processed.csv')

le_datetime = joblib.load('../models/datetime_encoder.pkl')

# encode datetime
test['OCCURRED_ON_DATE'] =  le_datetime.transform(test['OCCURRED_ON_DATE'])
train['OCCURRED_ON_DATE'] =  le_datetime.transform(train['OCCURRED_ON_DATE'])
val['OCCURRED_ON_DATE'] =  le_datetime.transform(val['OCCURRED_ON_DATE'])
```

In [ ]:
```
# remove _id column


val = val.drop('_id', axis=1)
test    = test.drop('_id', axis=1)
```

In [ ]:
```
# drop DISTRICT column
val = val.drop('DISTRICT', axis=1)
test    = test.drop('DISTRICT', axis=1)
train = train.drop('DISTRICT', axis=1)
```

In [ ]:
```
# define X and y
test_target = test['Severe_crimes']
test_features = test.drop('Severe_crimes', axis=1)

val_target = val['Severe_crimes']
val_features = val.drop('Severe_crimes', axis=1)

train_target = train['Severe_crimes']
train_features = train.drop('Severe_crimes', axis=1)
```

In [ ]:
```
# train model
rf = RandomForestClassifier(n_estimators=1000, max_depth=10, random_state=42)

rf.fit(train_features, train_target)
```

Out[ ]:
```
RandomForestClassifier(max_depth=10, n_estimators=1000, random_state=4
2)
```

In [ ]:
```
# save model
joblib.dump(rf, '../models/rf_model_week11.pkl')
print('model saved')
```

```
model saved
```

In [ ]:
```python
# read in other encoder
# load other encoders
le_district = joblib.load('../models/le_district.pkl')
le_description = joblib.load('../models/le_description.pkl')

# save train, test, val data with pickle
joblib.dump(train_features, '../data/processed/train_features.pkl')
joblib.dump(train_target, '../data/processed/train_target.pkl')

joblib.dump(val_features, '../data/processed/val_features.pkl')
joblib.dump(val_target, '../data/processed/val_target.pkl')

joblib.dump(test_features, '../data/processed/test_features.pkl')
joblib.dump(test_target, '../data/processed/test_target.pkl')

print('data saved')
```

```
data saved
```

In [ ]:
```python
# try everything
# load in train,test,val data from pickle
train_features = joblib.load('../data/processed/train_features.pkl')
train_target = joblib.load('../data/processed/train_target.pkl')

val_features = joblib.load('../data/processed/val_features.pkl')
val_target = joblib.load('../data/processed/val_target.pkl')

test_features = joblib.load('../data/processed/test_features.pkl')
test_target = joblib.load('../data/processed/test_target.pkl')

# load model
rf = joblib.load('../models/rf_model_week11.pkl')

# predcit on test data
test_pred = rf.predict(test_features)

# show test prediction
print(test_pred)
```

```
[1 0 1 ... 0 0 0]
```