ASSGN_4_PART_2

March 3, 2022

[46]:
```
# importing modules
import numpy as np
import pandas as pd
import random
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
from scipy.io import loadmat
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import cross_validate
from sklearn.metrics import accuracy_score
from sklearn.model_selection import validation_curve
```

[27]:
```
# loading the dataset
(x_train, y_train), (x_test, y_test) = keras.datasets.mnist.load_data()
y_train = y_train.reshape((-1,1))
y_test = y_test.reshape((-1,1))
```

[28]:
```
x = np.vstack((x_train,x_test))
y = np.vstack((y_train,y_test))
```

[30]:
```
# reshaping the input data to the range 0 - 1
x = x.reshape(-1, 28 * 28).astype("float32") / 255.0
```

[36]:
```
train_size = (int)(.75*x.shape[0])
```

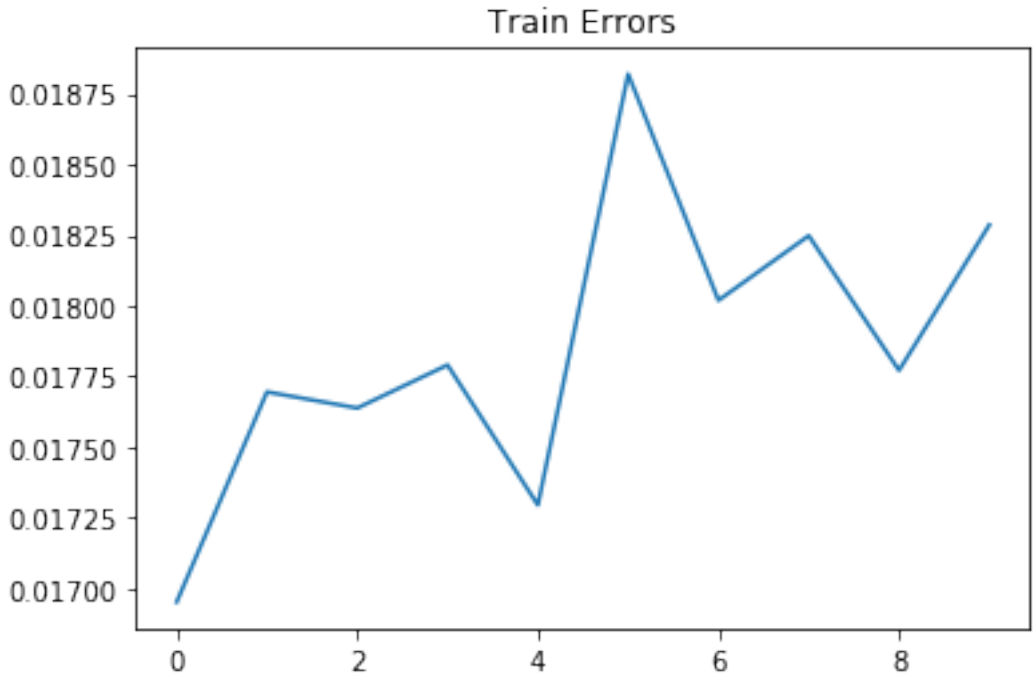
[41]:
```
x_train = x[:train_size,:]
x_test = x[train_size:,:]

y_train = y[:train_size,:]
y_test = y[train_size:,:]
```

[49]:
```
# Training the decision tree with 10-fold cross validation
clf = DecisionTreeClassifier(max_depth=20,random_state=0)
```

```
scores = cross_validate(estimator = clf, X = x_train,y= y_train,␣
 ↪cv=10,return_estimator=True)
clf = clf.fit(x_train, y_train)
```

[50]:
```
train_error = []
test_error = []
for i in range(10):
    train_outputs = scores["estimator"][i].predict(x_train)
    preds = scores["estimator"][i].predict(x_test)
    train_error.append(1-accuracy_score(y_true = y_train,y_pred =␣
 ↪train_outputs))
    test_error.append(1-accuracy_score(y_true = y_test,y_pred = preds))
```

[51]:
```
plt.plot(train_error)
plt.title("Train Errors")
plt.show()
```



[52]:
```
plt.plot(test_error)
plt.title("Test Errors")
plt.show()
```

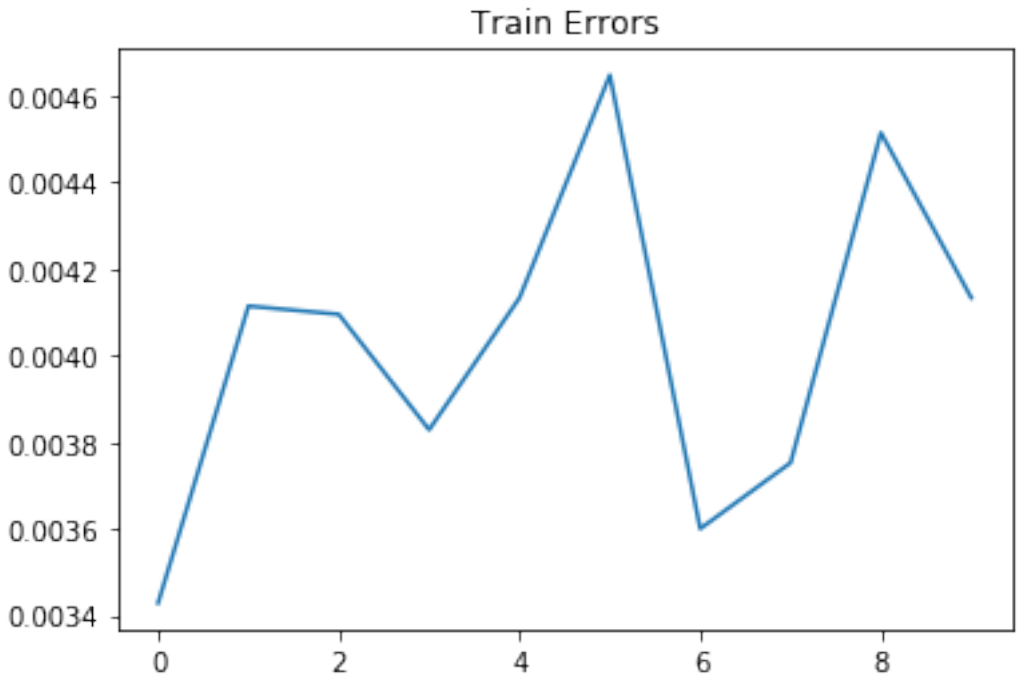
[53]:
```python
# Training the random forest with 10-fold cross validation
# Both train and test errors are seen improving
rfclf = RandomForestClassifier(max_depth=20,random_state=0)
scores = cross_validate(rfclf, x_train, y_train, cv=10,return_estimator=True)
rfclf = rfclf.fit(x_train, y_train)
```

```
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
```

```
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:5:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples,), for example using
ravel().
  """
```

[54]:
```
train_error = []
test_error = []
for i in range(10):
    train_outputs = scores["estimator"][i].predict(x_train)
    preds = scores["estimator"][i].predict(x_test)
    train_error.append(1-accuracy_score(y_true = y_train,y_pred =␣
 ↪train_outputs))
    test_error.append(1-accuracy_score(y_true = y_test,y_pred = preds))
```

[55]:
```
plt.plot(train_error)
plt.title("Train Errors")
plt.show()
```

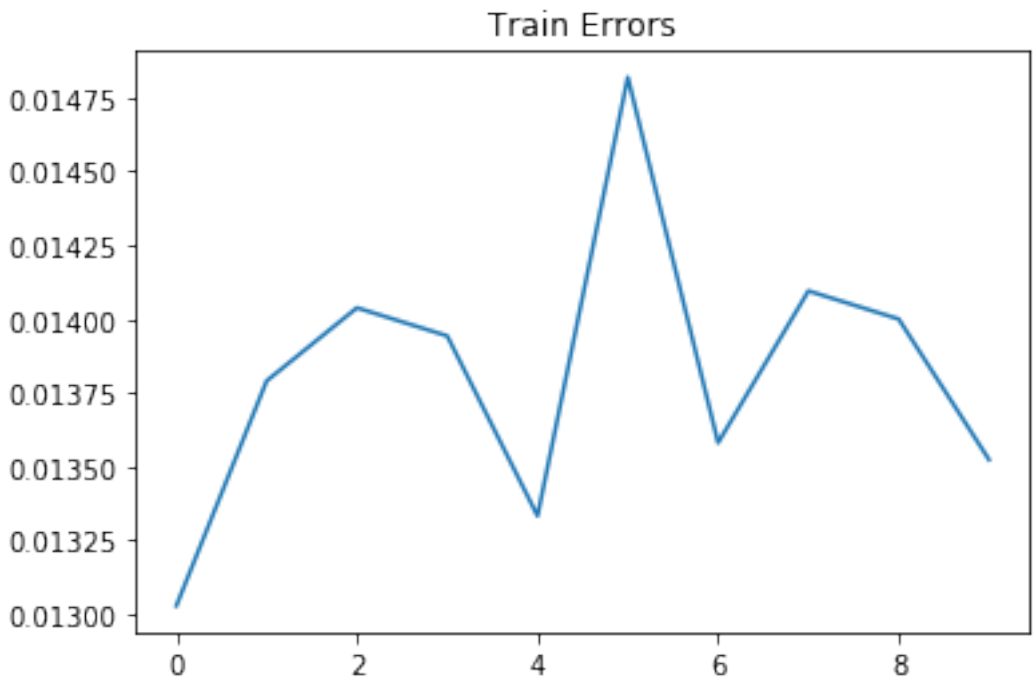


[56]:
```
plt.plot(test_error)
plt.title("Test Errors")
plt.show()
```

[57]:
```
# Training the decision tree with 10-fold cross validation
clf = DecisionTreeClassifier(max_depth=50,random_state=0)
scores = cross_validate(estimator = clf, X = x_train,y= y_train,␣
↪cv=10,return_estimator=True)
clf = clf.fit(x_train, y_train)
```

[58]:
```
train_error = []
test_error = []
for i in range(10):
    train_outputs = scores["estimator"][i].predict(x_train)
    preds = scores["estimator"][i].predict(x_test)
    train_error.append(1-accuracy_score(y_true = y_train,y_pred =␣
↪train_outputs))
    test_error.append(1-accuracy_score(y_true = y_test,y_pred = preds))
```

[59]:
```
plt.plot(train_error)
plt.title("Train Errors")
plt.show()
```



[60]:
```
plt.plot(test_error)
plt.title("Test Errors")
plt.show()
```

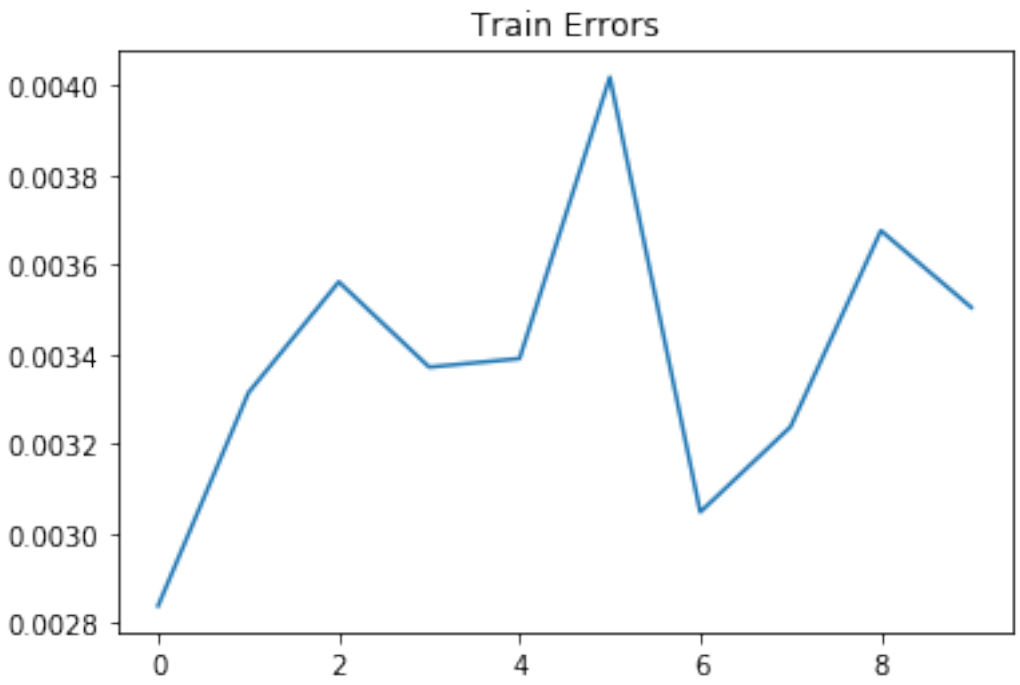
[61]:
```
# Training the random forest with 10-fold cross validation
# Both train and test errors are seen improving
rfclf = RandomForestClassifier(max_depth=50,random_state=0)
scores = cross_validate(rfclf, x_train, y_train, cv=10,return_estimator=True)
rfclf = rfclf.fit(x_train, y_train)
```

```
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
```

```
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-
packages/sklearn/model_selection/_validation.py:515: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/home/tfjuror/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:5:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples,), for example using
ravel().
  """
```

[62]:
```
train_error = []
test_error = []
for i in range(10):
    train_outputs = scores["estimator"][i].predict(x_train)
    preds = scores["estimator"][i].predict(x_test)
    train_error.append(1-accuracy_score(y_true = y_train,y_pred =␣
 ↪train_outputs))
    test_error.append(1-accuracy_score(y_true = y_test,y_pred = preds))
```

[63]:
```
plt.plot(train_error)
plt.title("Train Errors")
plt.show()
```



[64]:
```
plt.plot(test_error)
plt.title("Test Errors")
plt.show()
```

[ ]:

[ ]:

[ ]: