Try SVM classifier on MNIST dataset, compare the preformance of linear, polynomial and RBF kernels.

In [ ]:
```
import numpy as np
from sklearn import svm
from sklearn import metrics
import matplotlib.pyplot as plt
from mlxtend.plotting import plot_decision_regions
```

In [ ]:
```
from tensorflow.keras.datasets import mnist
(X_train, y_train), (X_test, y_test) = mnist.load_data()
X_train, X_test = np.array(X_train, np.float32), np.array(X_test,np.float32)
num_features = 784
X_train, X_test = X_train.reshape([-1, num_features]), X_test.reshape([-1, nu
X_train, X_test = X_train/255, X_test/255
```

In [ ]:
```
clf = svm.SVC(kernel="rbf")
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
print("Accuracy : ",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy :  0.9792
```

In [ ]:
```
clf = svm.SVC(kernel="linear")
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
print("Accuracy : ",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy :  0.9404
```

In [ ]:
```
clf = svm.SVC(kernel="poly")
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
print("Accuracy : ",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy :  0.9771
```

Using linear, poly, rbf ,accuracy increases respectively.