**Assignment 5**

**Compare the accuracies of Moving Average, Single Exponential Smoothing, Double Exponential Smoothing and Triple Exponential Smoothing**

def MAM(X,window):

history = [X[i] for i in range(window)] #Initially observed only 3 values

test = [X[i] for i in range(window, len(X))] #WE can't predict X[2] as there is not enough data

predictions = []

for t in range(len(test)):

length = len(history)

yhat = mean([history[i] for i in range(length-window,length)])

obs = test[t]

predictions.append(yhat)

history.append(obs)

#print(predictions[t],history[t])

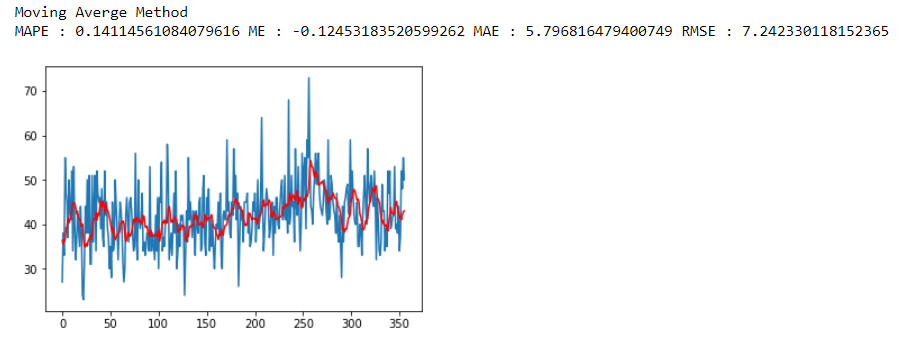
print("Moving Averge Method")

forecast\_accuracy(predictions,test)

pyplot.plot(test)

pyplot.plot(predictions, color='red')

pyplot.show()



def ES(series, alpha):

result = [series[0]] # first value is same as series

for n in range(1, len(series)):

result.append(alpha \* series[n-1] + (1 - alpha) \* result[n-1])

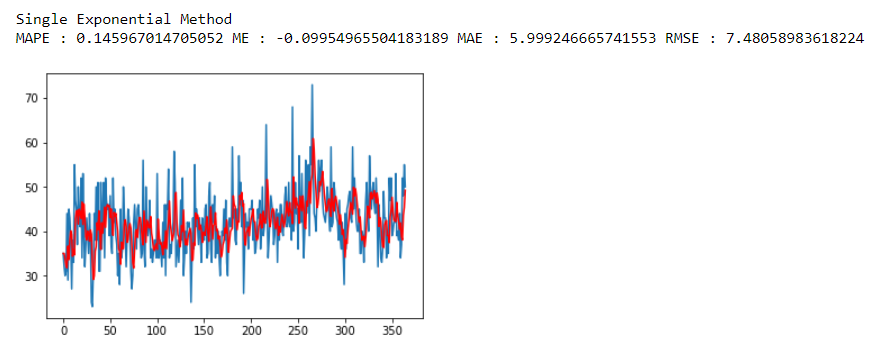
print("Single Exponential Method")

forecast\_accuracy(result,series)

pyplot.plot(series)

pyplot.plot(result, color='red')

pyplot.show()



def DES(series, alpha, beta):

result = [series[0]]

for n in range(1, len(series)+1):

if n == 1:

level, trend = series[0], series[1] - series[0]

if n >= len(series): # we are forecasting

val = result[-1]

else:

val = series[n]

last\_level, level = level, alpha\*val + (1-alpha)\*result[n-1]

trend = beta\*(level-last\_level) + (1-beta)\*trend

result.append(level+trend)

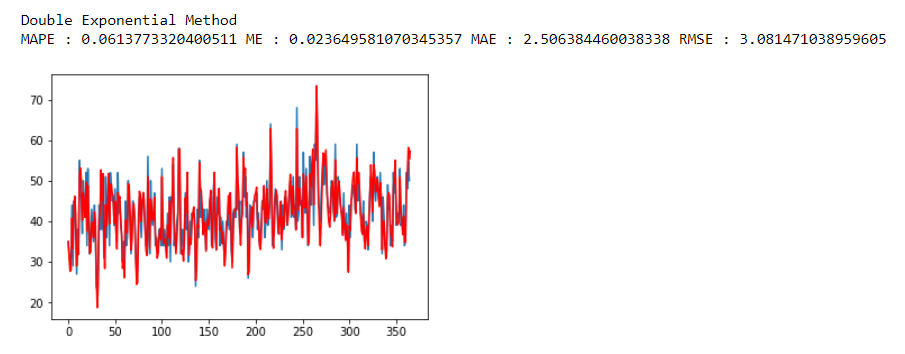
print("Double Exponential Method")

forecast\_accuracy(result,series)

pyplot.plot(series)

pyplot.plot(result, color='red')

pyplot.show()



def TES(series, slen, alpha, beta, gamma, n\_preds):

result = []

seasonals = initial\_seasonal\_components(series, slen)

for i in range(len(series)+n\_preds):

if i == 0: # initial values

smooth = series[0]

trend = initial\_trend(series, slen)

result.append(series[0])

continue

if i >= len(series): # we are forecasting

m = i - len(series) + 1

result.append((smooth + m\*trend) + seasonals[i%slen])

else:

val = series[i]

last\_smooth, smooth = smooth, alpha\*(val-seasonals[i%slen]) + (1-alpha)\*(smooth+trend)

trend = beta \* (smooth-last\_smooth) + (1-beta)\*trend

seasonals[i%slen] = gamma\*(val-smooth) + (1-gamma)\*seasonals[i%slen]

result.append(smooth+trend+seasonals[i%slen])

print("Triple Exponential Method")

forecast\_accuracy(result,series)

pyplot.plot(series)

pyplot.plot(result, color='red')

pyplot.show()

