Derek D’Arcy
10/8/24
Intro to Data Science

Test 1 Analysis - Conclusion

Example calculations:

Exponential smoothing

$$\hat{Y}_{t+1} = \hat{Y}_t + a(Y_t - \hat{Y}_t)$$

Yhat t+1 = 1000 + (0.5)*(2200-1000) = 1600.

Trend:

$$T_{t+1} = \beta(S_{t+1} - S_t) + (1 - \beta)T_t$$

(0.5) * (1600-1000) + (1-0.5)*0 = 300

AES:

$$\hat{Y}_{t+1} = S_{t+1} + \left(\frac{1-\beta}{\beta}\right)T_{t+1}$$

(1000) + (1-0.5/0.5) * 0 =1000

Increment

$$\text{Increment} = \frac{\text{Last } \hat{\text{Y}}_\text{t} - \text{First } \hat{\text{Y}}_\text{t}}{\text{n} - 1}$$

(76213.81348 - 1000) / (18-1) = 4424.34

Forecast = last Yhat + increment
76213.81 + 4424.34 = 80638.15

List Accuracy from best to worst presenting alphas, betas, and the accuracy:

Derek D'Arcy
10/8/24
Intro to Data Science

Most to least accurate (alpha, beta)		
	0.2, 0.7	1871.846304
	0.3, 0.7	2717.671372
	0.4, 0.7	3542.578787
	0.2, 0.5	3798.517292
	0.5, 0.7	4351.286947
	0.3, 0.5	4825.35219
	0.6, 0.7	5143.306163
	0.4, 0.5	6037.65467
	0.5, 0.5	7412.522434
	0.2, 0.3	7413.477857
	0.6, 0.5	8761.563441
	0.3, 0.3	8848.459963
	0.4, 0.3	10071.53022
	0.5, 0.3	11185.08104
	0.6, 0.3	12667.32794

The accuracy measurement I chose to use was MAD, or Mean absolute deviation. I chose this as I believed that it would be easiest in the time constraints given to compare the MAD values quickly and efficiently.

The formula for this is: $MAD = \frac{1}{n}\sum_{i=1}^{n} | Y_i - \hat{Y}_i |$. A sample calculation would be the absolute value of subtracting all the AES forecast from the actual observations for each data point (1000-1000, 1600, 1900, etc.) adding that all together (1000 + 0 + 300 + 50, etc) to get the sum, and then dividing that by the number of observations (133435/18) = 7412.52 for alpha = 0.5, beta = 0.5 for example.

The strategy I chose for selecting alpha, beta, and the forecasting process involved first selecting a very neutral alpha, 0.5. I then combined this with a neutral beta of 0.5, and then chose 0.3 and 0.7 as my second and third betas, as this would give an even view on either side, with a reasonably low beta and reasonably high beta to compare with.

MAD Table	Beta		
Alpha	0.5	0.3	0.7
0.5	7412.522434	11185.08104	4351.286947

After calculating these three, I chose in a random direction, and decided to step up the alpha by 0.1, keeping the betas the same.

Derek D'Arcy
10/8/24
Intro to Data Science

MAD Table	Beta		
Alpha	0.5	0.3	0.7
0.5	7412.522434	11185.08104	4351.286947
0.6	8761.563441	12667.32794	5143.306163

From here, I saw that when stepping up the alpha by 0.1, the accuracy actually got worse, so I instead turned around and went down from my initial starting point to go to 0.4.

MAD Table	Beta		
Alpha	0.5	0.3	0.7
0.5	7412.522434	11185.08104	4351.286947
0.6	8761.563441	12667.32794	5143.306163
0.4	6037.65467	10071.53022	3542.578787

I saw that going this direction helped the accuracy, being more accurate than both the 0.6 alpha, and the 0.5 alpha. Encouraged by the results, I decided to continue stepping downwards by an equal amount, keeping the betas the same to have uniform results to compare to. This led to me choosing 0.3, and 0.2 as the fourth and fifth alpha value, seeing that the values got more accurate the lower the alpha got.

MAD Table	Beta		
Alpha	0.5	0.3	0.7
0.5	7412.522434	11185.08104	4351.286947
0.6	8761.563441	12667.32794	5143.306163
0.4	6037.65467	10071.53022	3542.578787
0.3	4825.35219	8848.459963	2717.671372
0.2	3798.517292	7413.477857	1871.846304

I also found it curious that the lower the alpha got, the more the beta value difference mattered.

By noticing that the accuracy is higher the lower the alpha value, we can infer that the forecast is more accurate when smoothing out short term fluctuations, and looking at the entire data set as a whole, rather than weighting more recent data higher.