**Homework 4**

ISYE – 6501

02/10/2022

# Question 7.1

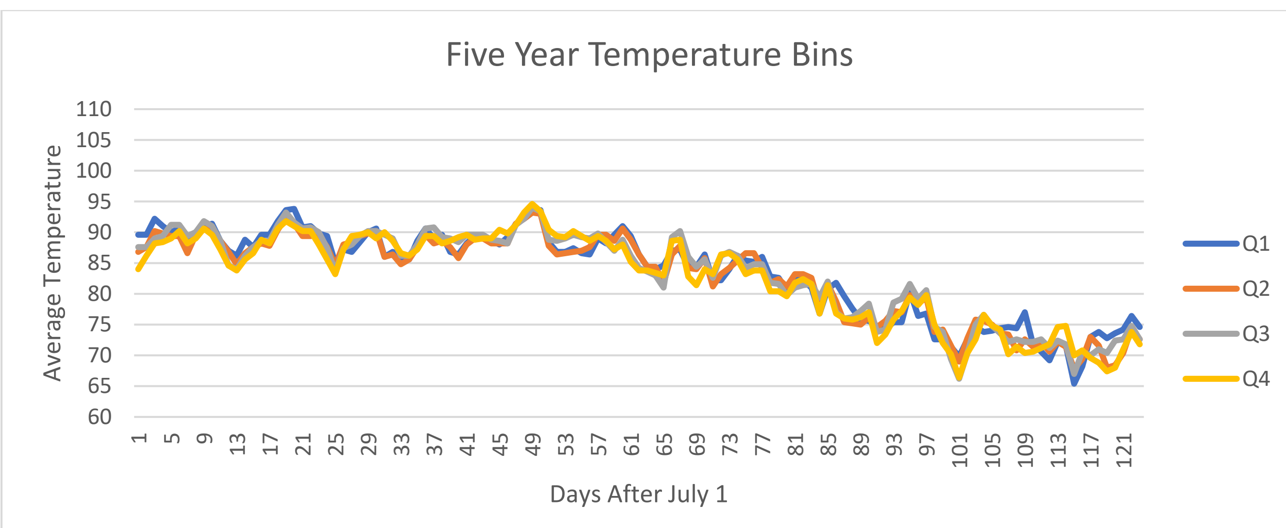
On-site, in-situ wind speed, direction, and temperature measurements are crucial to designing and optimizing a utility-scale wind energy plant. A fleet of 300 meteorological towers and ground-based remote sensing SODARS and LIDARS deployed all across the country are used to take measurements up to 200m into the atmosphere on a 10-minute time series basis. In order to do a climatological analysis, the collected data must be cleaned, validated, and correlated. with satellite data. Possible sensor data quality issues include failure, degradation, icing, tower shading, and waking. Because we have such a high volume of data coming in every day, we've implemented rudimentary change detection to identify sensor failures. This is relatively easy because flatlines are easier to detect. However I hope to utilize what I've learned in this class in order to detect more subtle changes like sensor degradation where a sensor hasn't flatlined, but is nevertheless reading lower than it should. Historical data has been manually flagged, so CUSUM parameters can be tailored based on the handmade training dataset. Wind speeds and directions change rapidly – especially at elevation – so they inherently have a high degree of noise. Due to this high degree of randomness, I would choose an alpha value closer to zero. This would prioritize the $S_{t-1}$ term – the previous baseline. Wind speeds are also highly seasonal, so the gamma term would help to identify patterns. Additionally, the beta term would help identify any long-term, non-cyclical trends such as the effect of climate change on wind speeds.

# Question 7.2

CUSUM and Excel were used to determine the end date of summer for the years in the period 1996 – 2015. Please refer to the excel spreadsheet for the detailed analysis. Formulas for CUSUM parameters were as follows:

$$\mu = AVERAGE(temps!B\$2:B\$40)$$

The critical period of $\mu$ was defined by qualitatively assessing the following graph to identify the period in the data that we don't see substantial change. The period from July 1st to August 8th was selected.



$$S_t = MAX(0, C9 + (C\$6 - temps!B3 - \$C\$2))$$

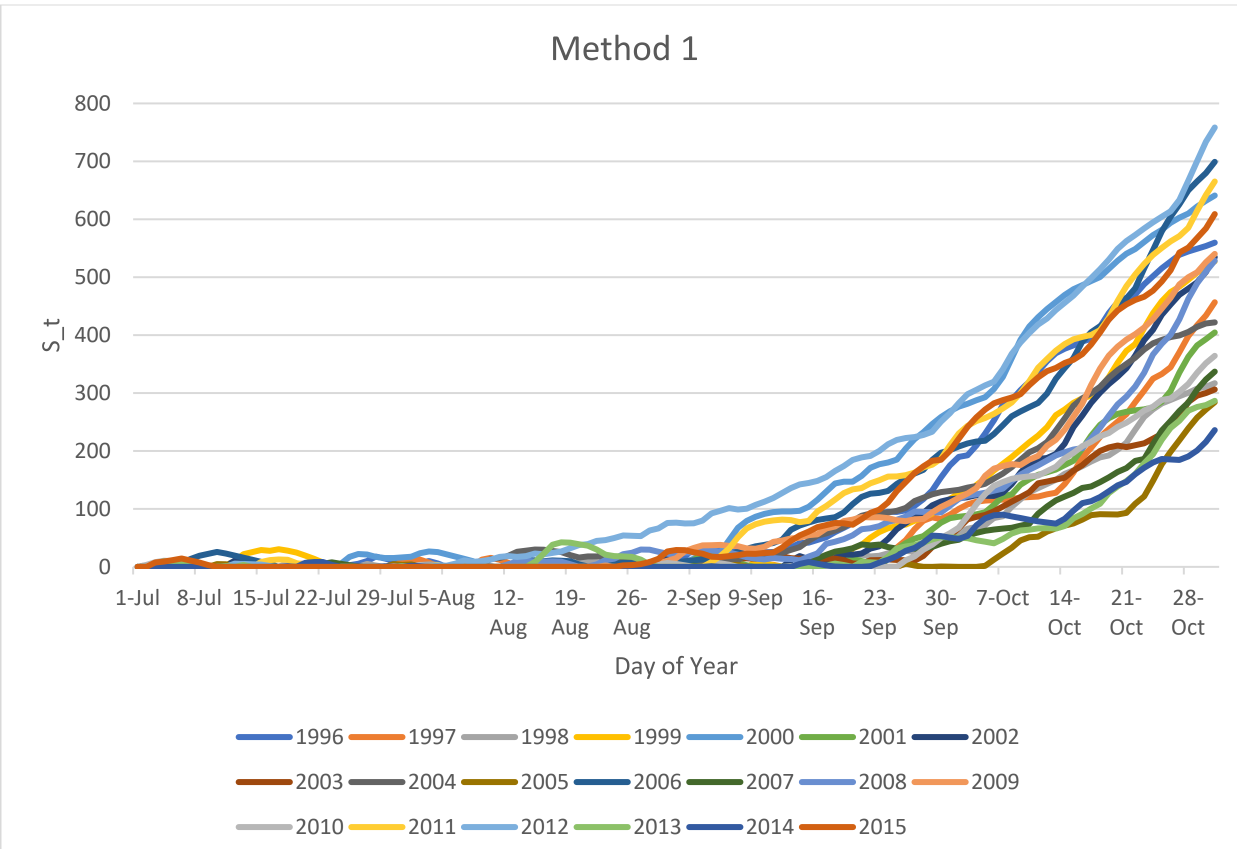
Three different methods were applied to determine C and T parameters.

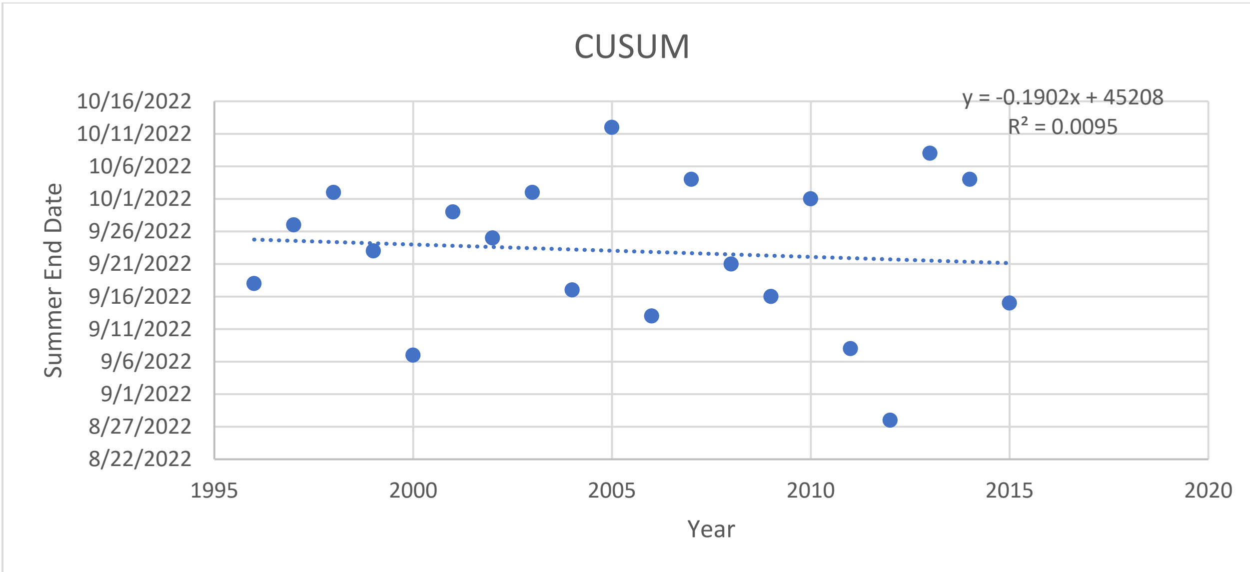
**Method 1 - Overfit**

The first was to calculate the end date of summer for each year with the following formula:

=INDIRECT("b"& MATCH(C132,C9:C131,0)+8)

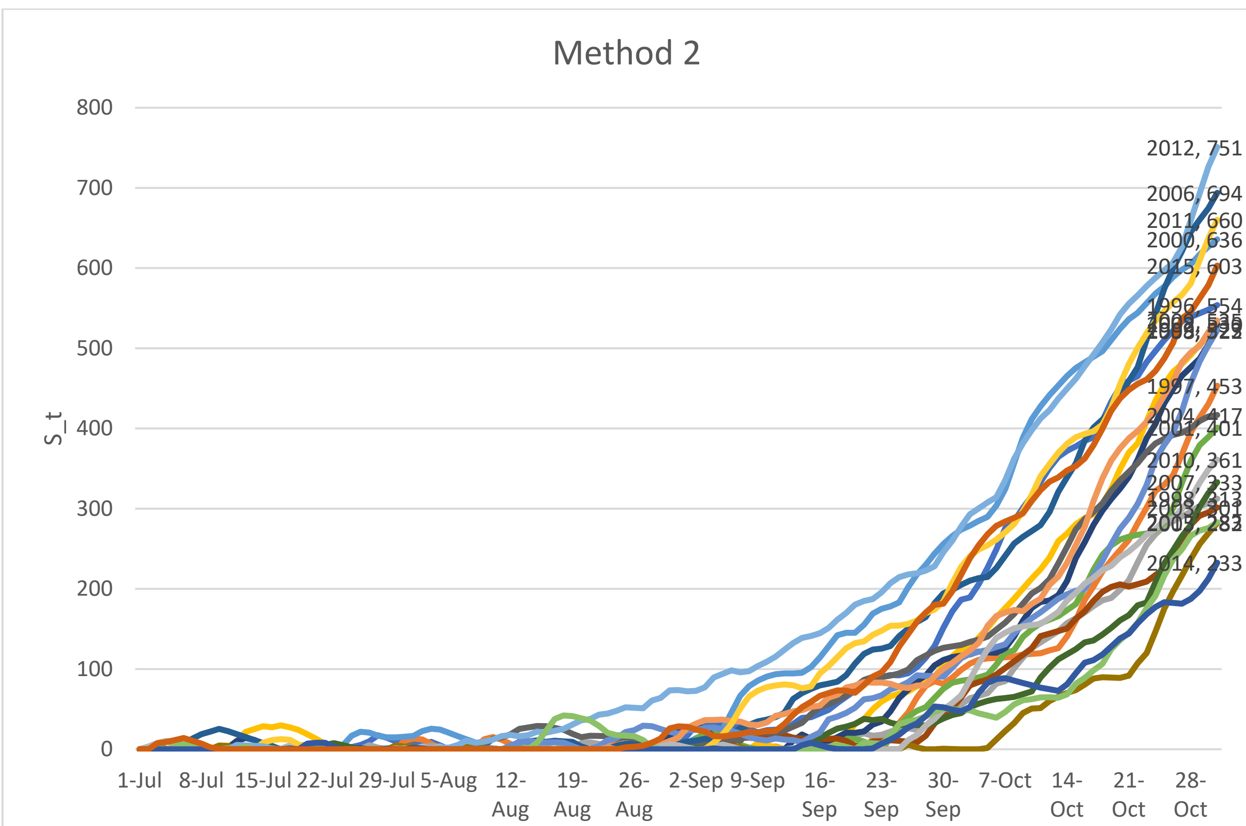
Then, the average end date over the 15 year period was calculated. Next, C and T were scaled using the manual goal-seek method until the average end date was the official end date of summer: September 22. This method fits expected seasonal trends, however it may be considered overfitting the model.

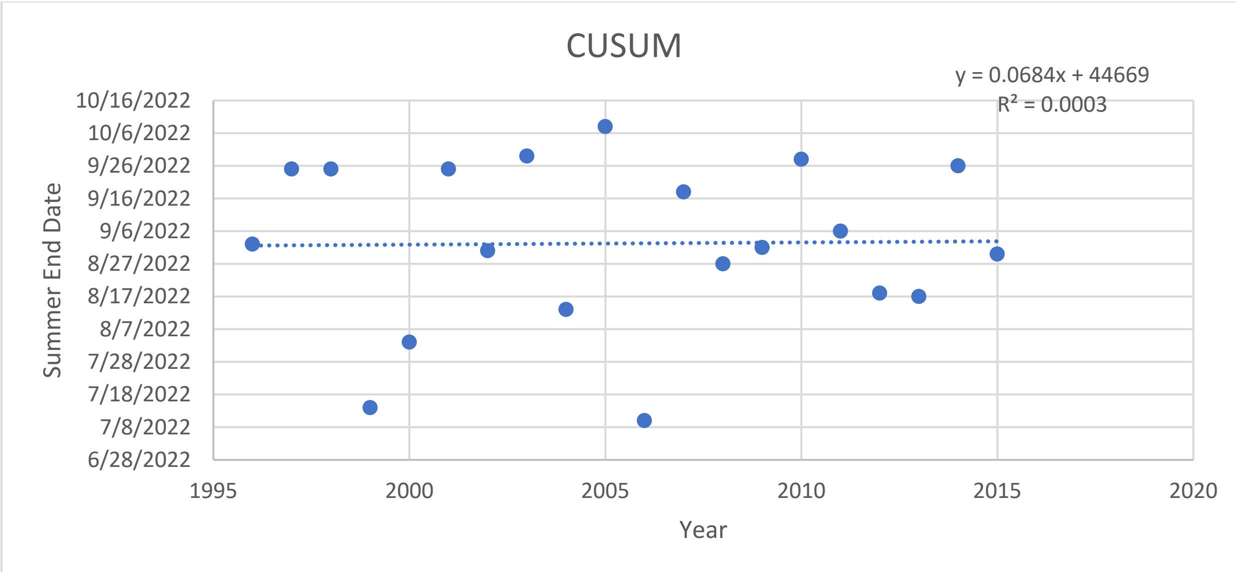





**Method 2 – Static Standard Deviation**

Method 2 uses C = 1 standard deviation and T equals 6 standard deviations.

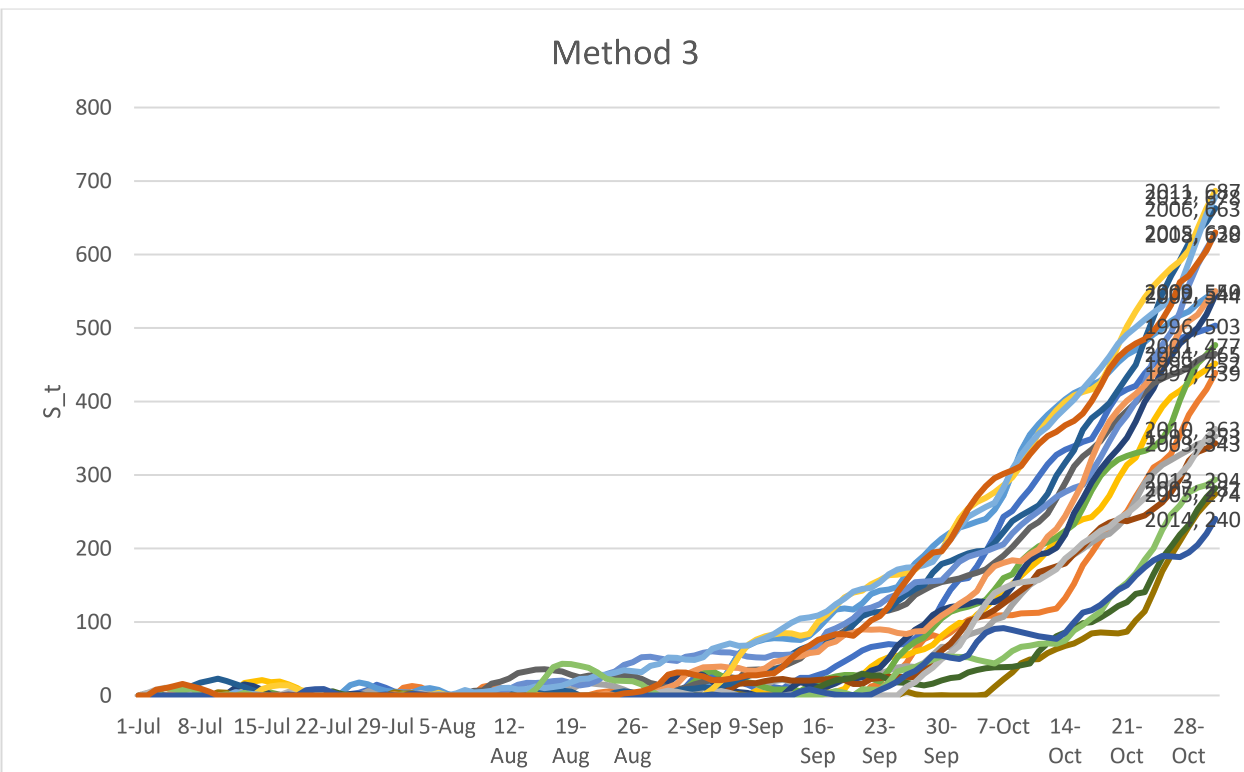


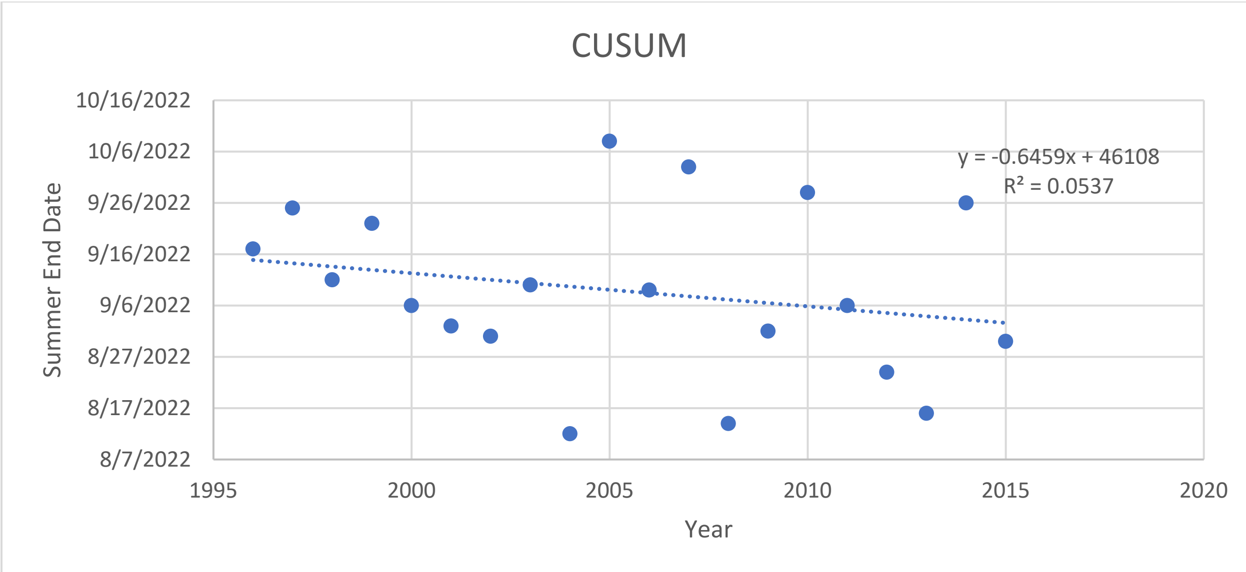



This method yielded an average end of summer date of September 2, about three weeks prior to the official end.

**Method 3 – Dynamic Standard Deviation**

Method three calculates a separate standard deviation, C and T value for each discreate year. C and T are still sized at 1x and 6x the annual standard deviation respectively.






Method 3 yielded an average end of summer on September 8th, 2 weeks before the official end date.

The time series temperature data is somewhat noisy. Exponential smoothing may be able to reduce random variations such that trends or cycles are more apparent. Holt-Winters filtering in R allows the user to calculate alpha, beta, and/or gamma parameters in an additive or multiplicative seasonal model.

---

Import data and ensure that it is as expected with the head() function.

```r
temps <- read.table("temps.txt", stringsAsFactors = FALSE, header = TRUE)
head(temps)
```

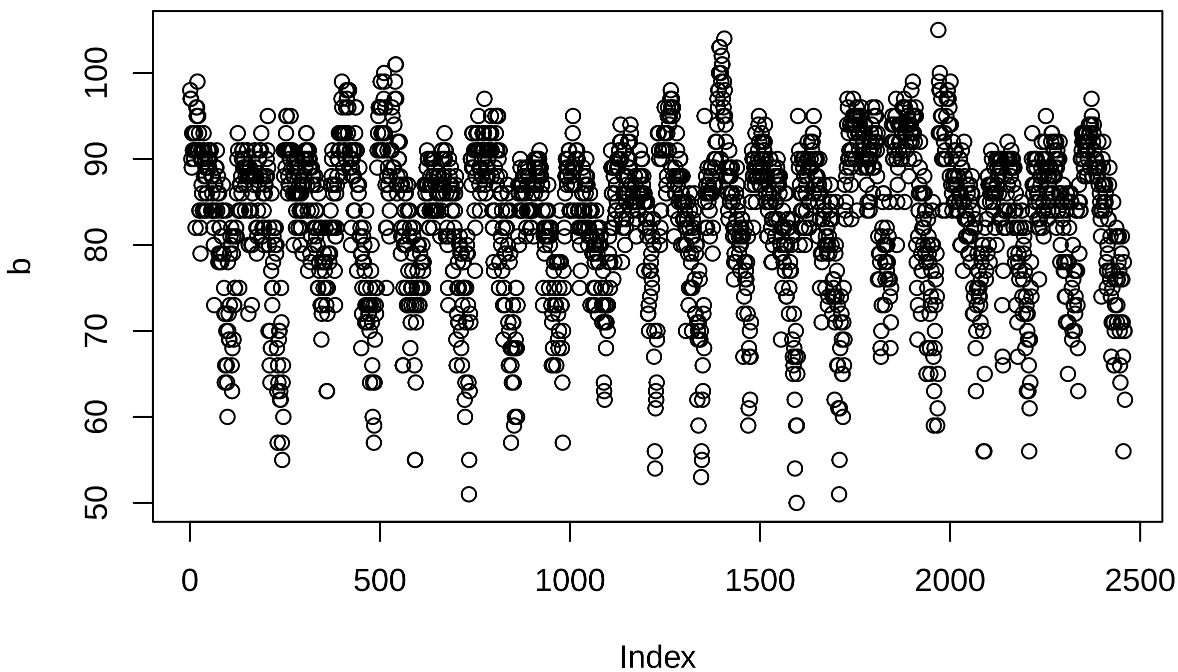
```
##     DAY X1996 X1997 X1998 X1999 X2000 X2001 X2002 X2003 X2004 X2005 X2006 X2007
## 1 1-Jul    98    86    91    84    89    84    90    73    82    91    93    95
## 2 2-Jul    97    90    88    82    91    87    90    81    81    89    93    85
## 3 3-Jul    97    93    91    87    93    87    87    87    86    86    93    82
## 4 4-Jul    90    91    91    88    95    84    89    86    88    86    91    86
## 5 5-Jul    89    84    91    90    96    86    93    80    90    89    90    88
## 6 6-Jul    93    84    89    91    96    87    93    84    90    82    81    87
##   X2008 X2009 X2010 X2011 X2012 X2013 X2014 X2015
## 1    85    95    87    92   105    82    90    85
## 2    87    90    84    94    93    85    93    87
## 3    91    89    83    95    99    76    87    79
## 4    90    91    85    92    98    77    84    85
## 5    88    80    88    90   100    83    86    84
## 6    82    87    89    90    98    83    87    84
```

Consolidate all data in a single vector. This is necessary in order to use the time series object.

```
b <- unlist(temps[,2:21])
head(b)
```

```
## X19961 X19962 X19963 X19964 X19965 X19966
##     98     97     97     90     89     93
```

```
plot(b)
```



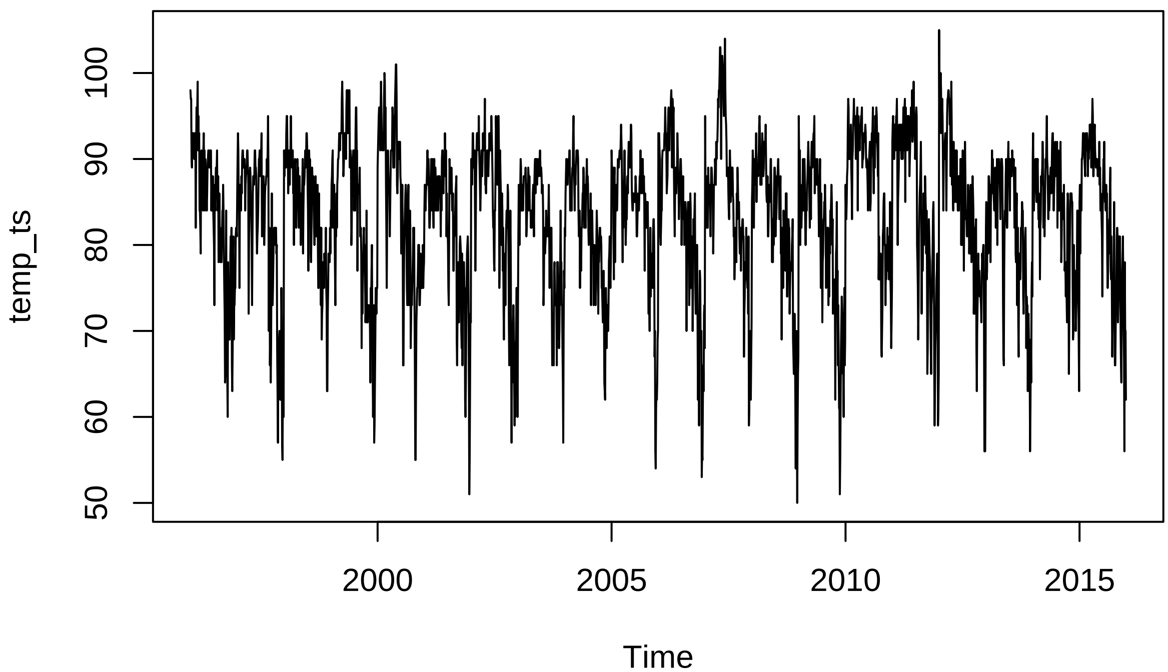

Sanity checks as standard practice. Length of b should be equal to 123 observation times 20 columns = 2460 elements

```
length(b)
```

```
## [1] 2460
```

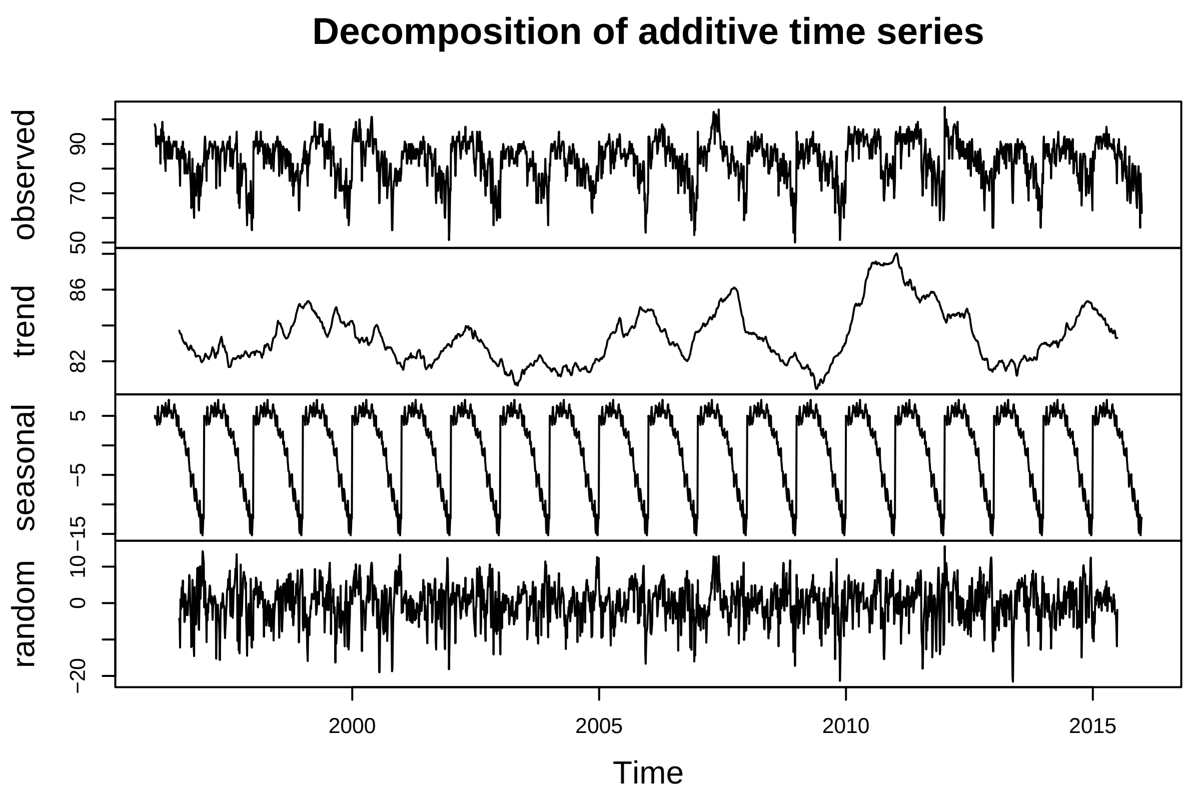
The time series object accepts data, a starting point as a reference, and a frequency argument that determines the repeat point.

```
temp_ts <- ts(b, start = 1996, frequency = 123)
plot(temp_ts)
```



The decompose() function is a high-level moving average way of assessing trend and seasonality by averaging over all periods. These plots indicate a distinct seasonal, cyclical occurrence but no trend.

```
plot(decompose(temp_ts))
```




We now apply the Holt-Winters function to the temperature vector. We can use both the additive and multiplicative models.

```
temps_HWA <- HoltWinters(temp_ts, alpha = NULL, beta = NULL, gamma = NULL, seasonal = "additive")
temps_HW <- HoltWinters(temp_ts, alpha = NULL, beta = NULL, gamma = NULL, seasonal = "multiplicative")
```

Now we calculate the sum of squares error for both models.

```
temps_HWA$SSE
```

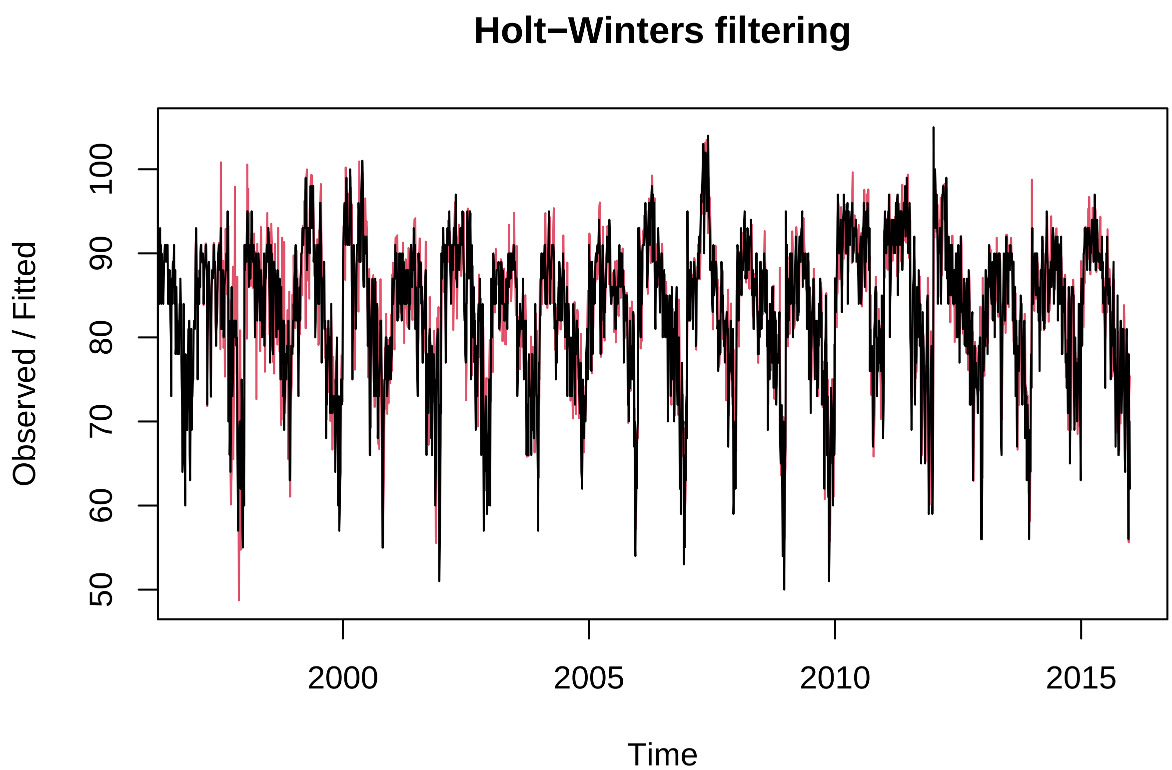
```
## [1] 66244.25
```

```
temps_HW$SSE
```

```
## [1] 68904.57
```

We select the additive model because it has lower error. When comparing the predictive model to the measured data, we can see that prediction error appears to decrease as the time series progresses.

```
plot(temps_HWA)
```



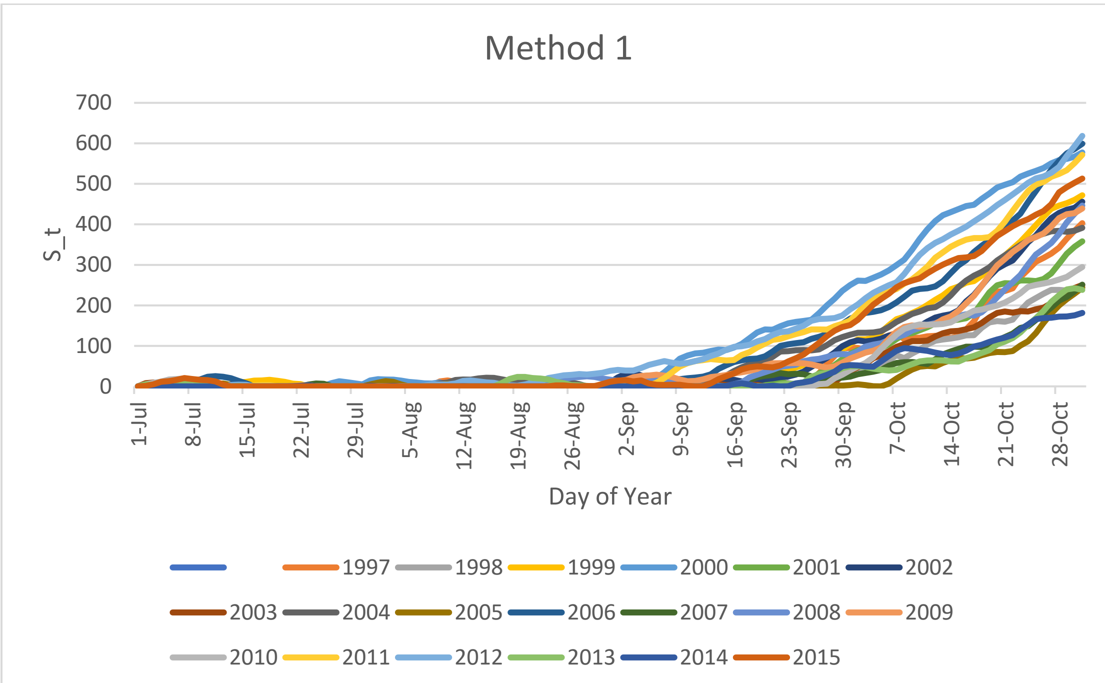

We take the additive model and convert the vector into a matrix before outputing both the fitted xhat and seasonal values to a tab-delimited txt file.

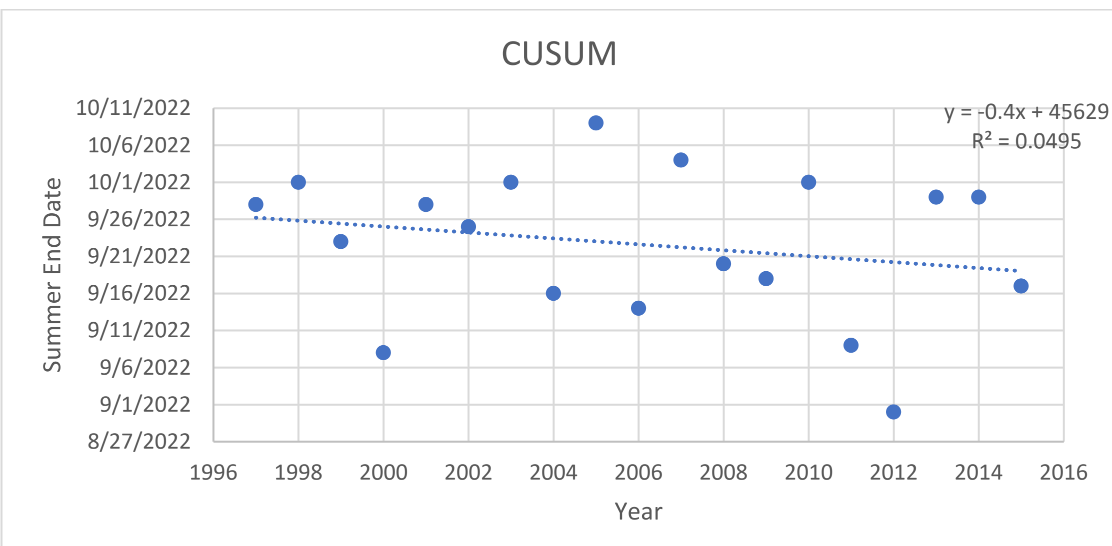
```
temps_HWA_smoothed <- matrix(temps_HWA$fitted[,1], nrow = 123)
write.table(temps_HWA_smoothed, file = "temps_HWA_smoothed.txt",  sep = "\t")

temps_HWA_smoothed_season <- matrix(temps_HWA$fitted[,4], nrow = 123)
write.table(temps_HWA_smoothed_season, file = "temps_HWA_smoothed_season.txt",  sep = "\t")
```

We now apply the above three CUSUM methods to the exponential smoothed data set. We also evaluate the effect of using xhat versus the seasonal data.

**Method 1 – Overfit (xhat)**

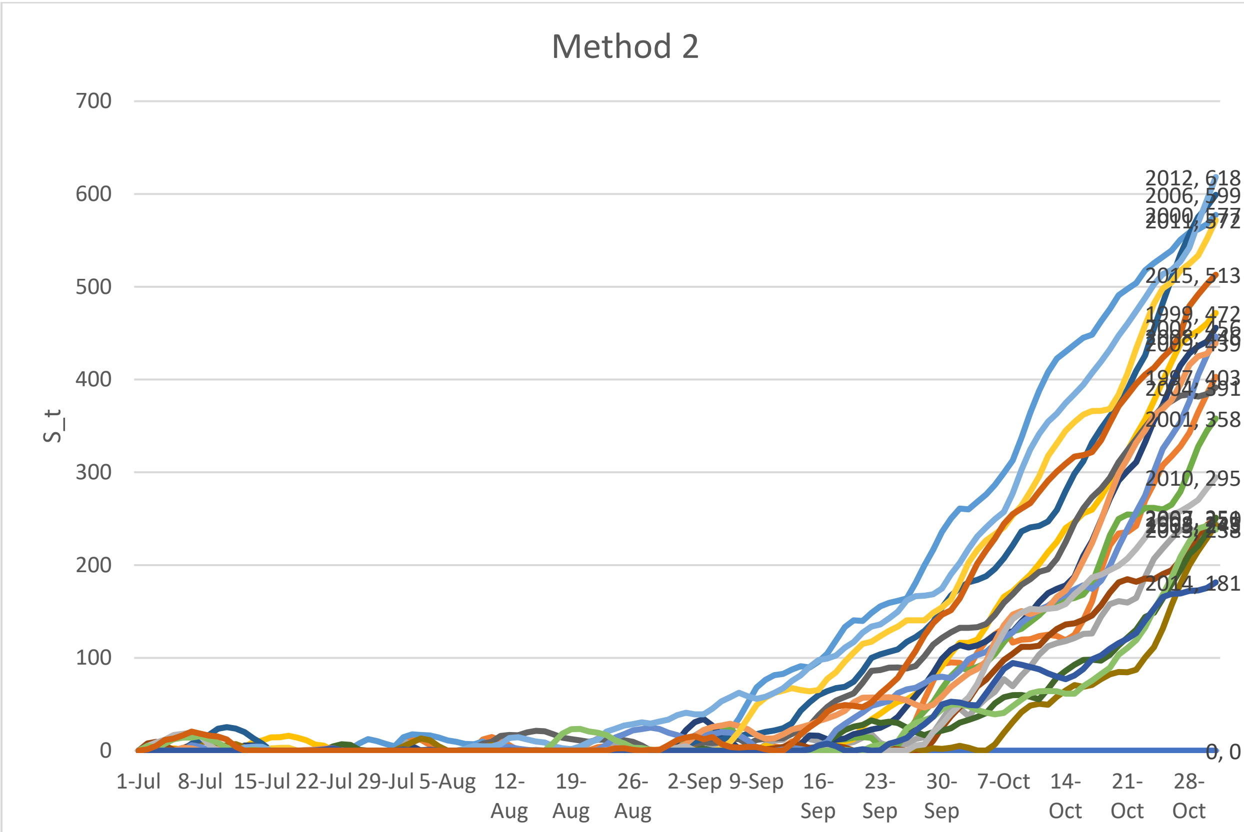
Average Summer End Date: September 22

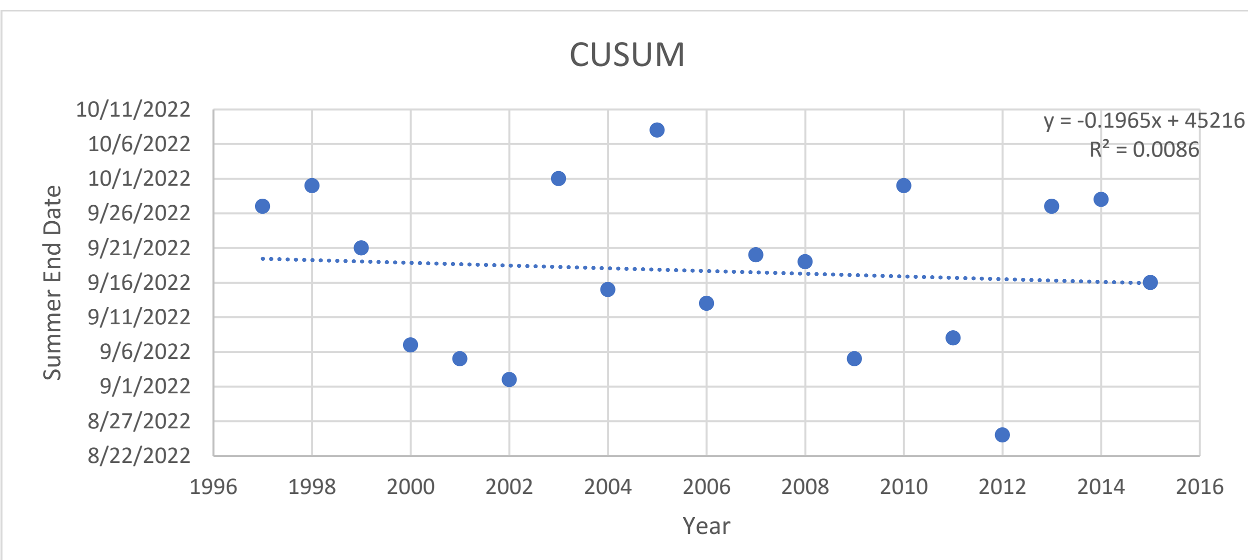


**Method 2 – Static Standard Deviation (xhat)**

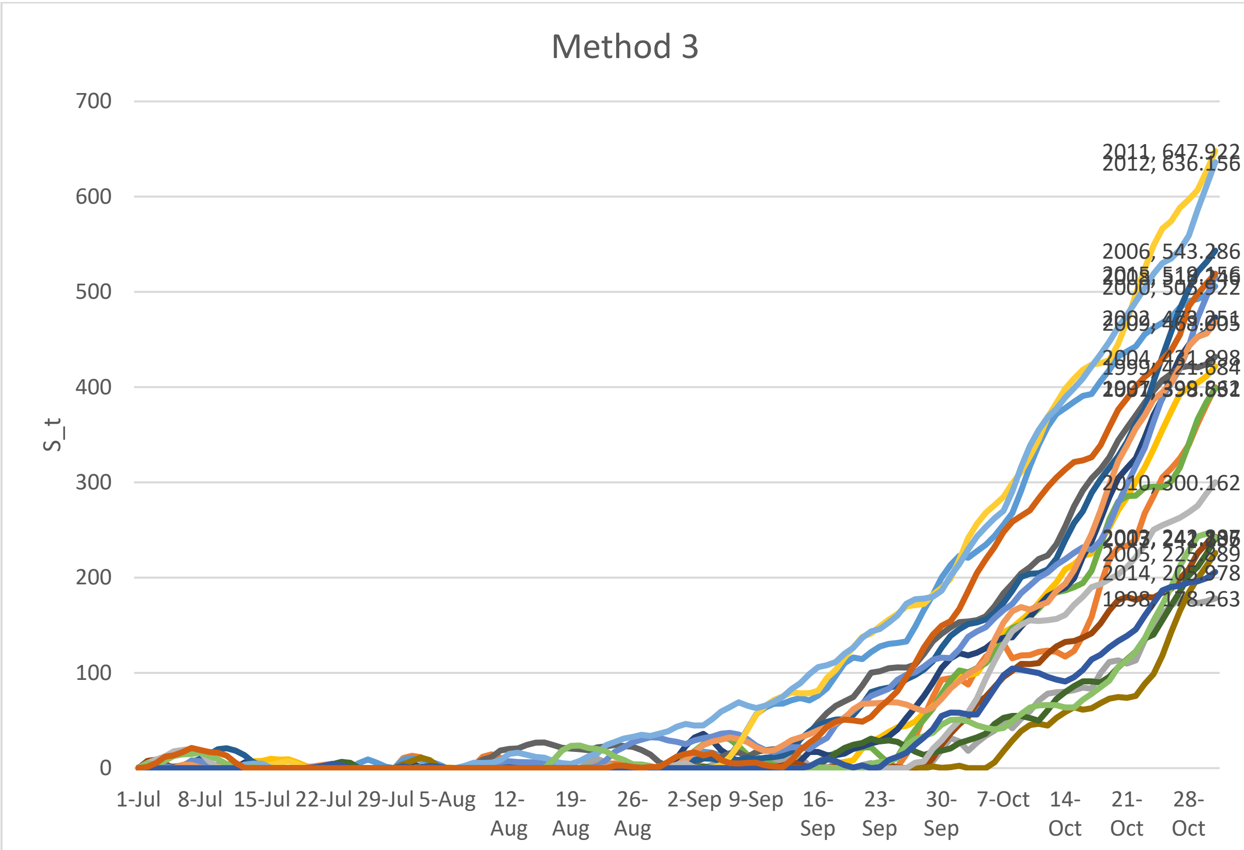
Average Summer End Date: September 17

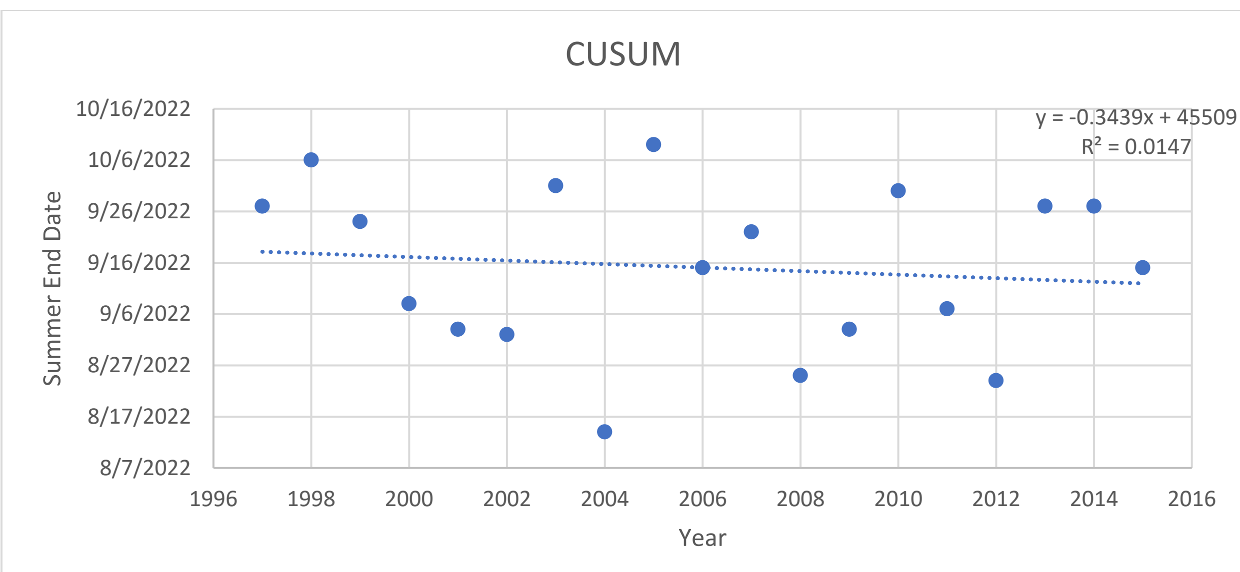



**Method 3 – Dynamic Standard Deviation (xhat)**

Average Summer End Date: September 15



The same process was repeated for the additive seasonal parameters, however their graphs are omitted in the interest of space. Instead, we present the results in table format:

| | Parameter | Method | Summer End Date | r2 |
|---|---|---|---|---|
| Raw | n/a | 1 | 22-Sep | 0.0095 |
| | | 2 | 2-Sep | 0.0003 |
| | | 3 | 8-Sep | 0.0537 |
| Smoothed | xhat | 1 | 22-Sep | 0.0495 |
| | | 2 | 17-Sep | 0.0086 |
| | | 3 | 15-Sep | 0.0147 |
| | seasonal | 1 | 22-Sep | 0.6679 |
| | | 2 | 15-Sep | 0.4534 |
| | | 3 | 9-Sep | 0.3981 |

We can see that the seasonal parameters result in a higher r-squared value compared to the xhat, however it is insufficient to be confident in any type of trend. While the exponential smoothing technique did reduce noise, it failed to identify any further trends. Therefore, we still do not have enough data to support the hypothesis that Summer is ending later in Atlanta.