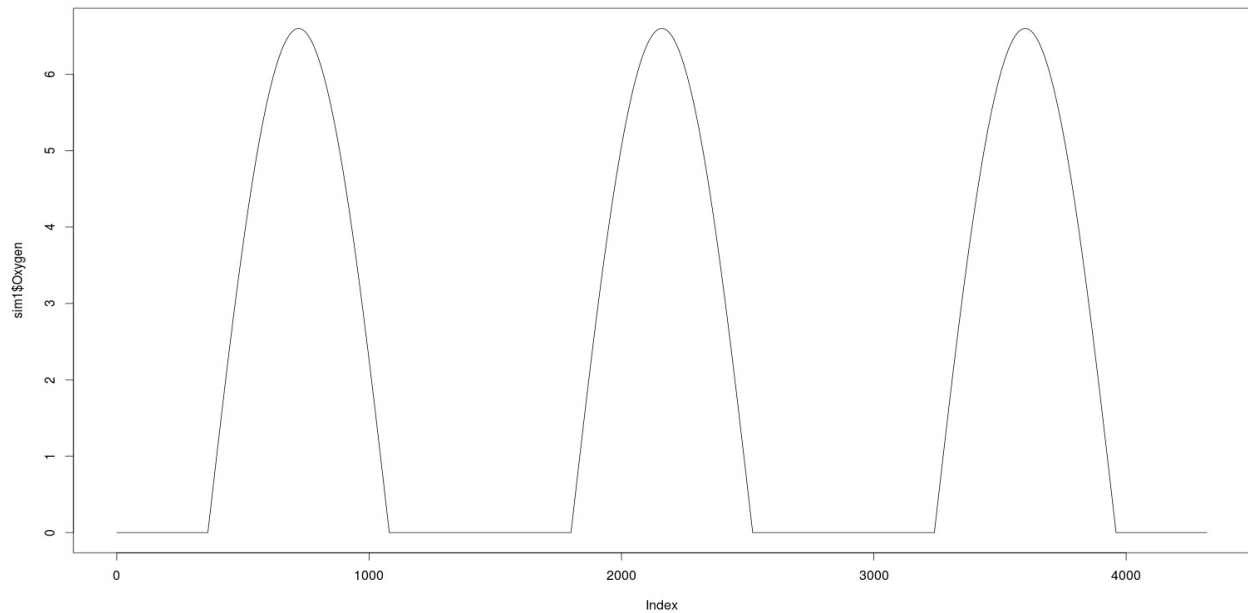


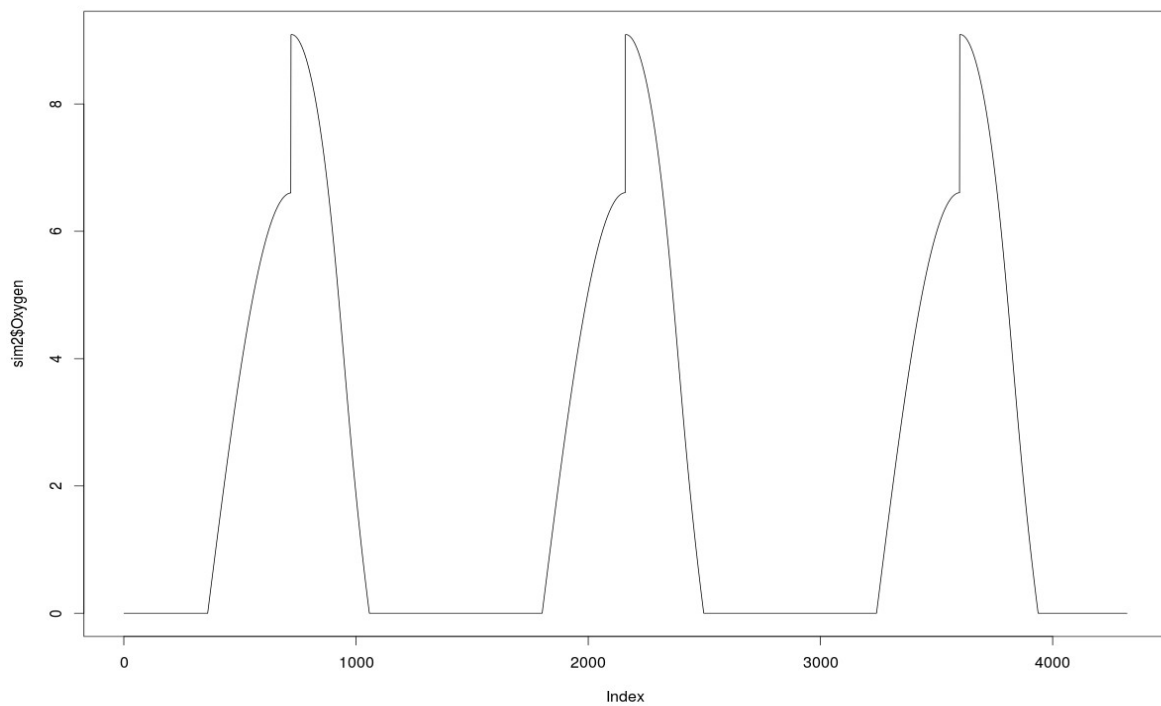
What do you think about this:

```
sim1 <- pitcherPlantSim(foodWeight=0) # baseline  
sim2 <- pitcherPlantSim(foodWeight=10) # with food
```

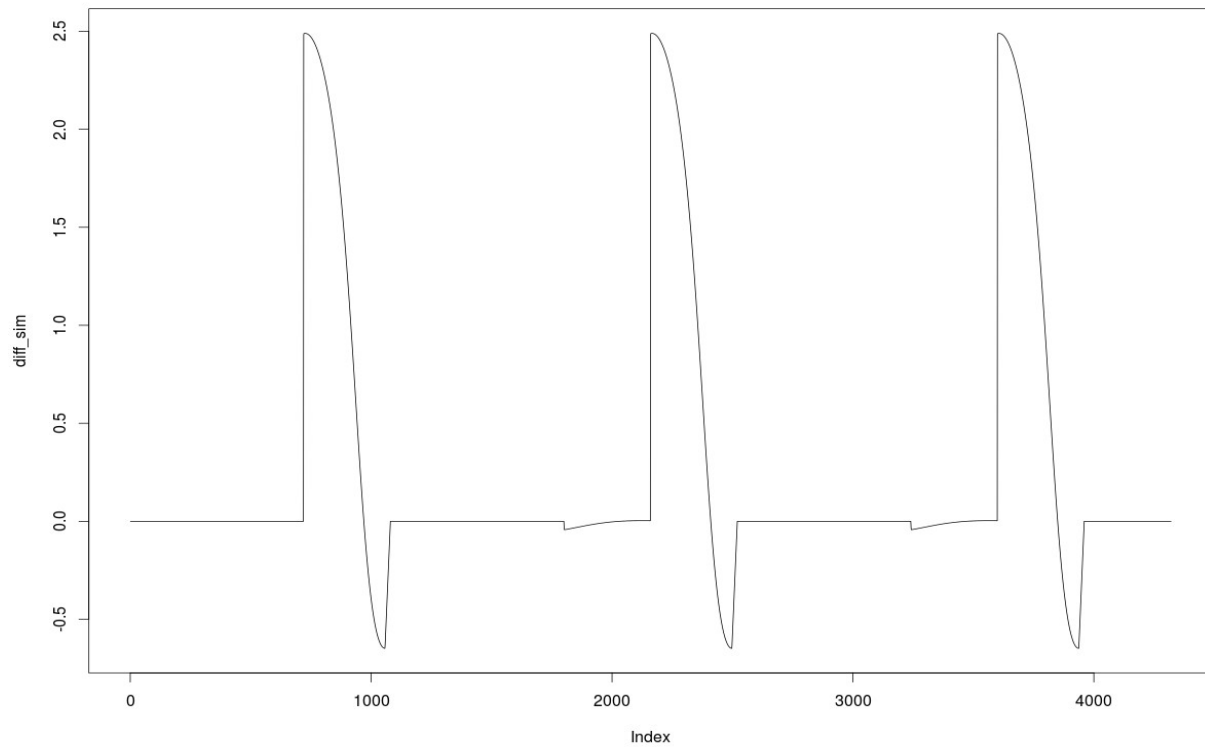
```
plot(sim1$Oxygen, type="l")
```



```
plot(sim2$Oxygen, type="l")
```



```
diff_sim <- sim2$Oxygen - sim1$Oxygen # difference between the two  
plot(diff_sim, type="l")
```



This looks interesting to me. Is this what you'd expect?

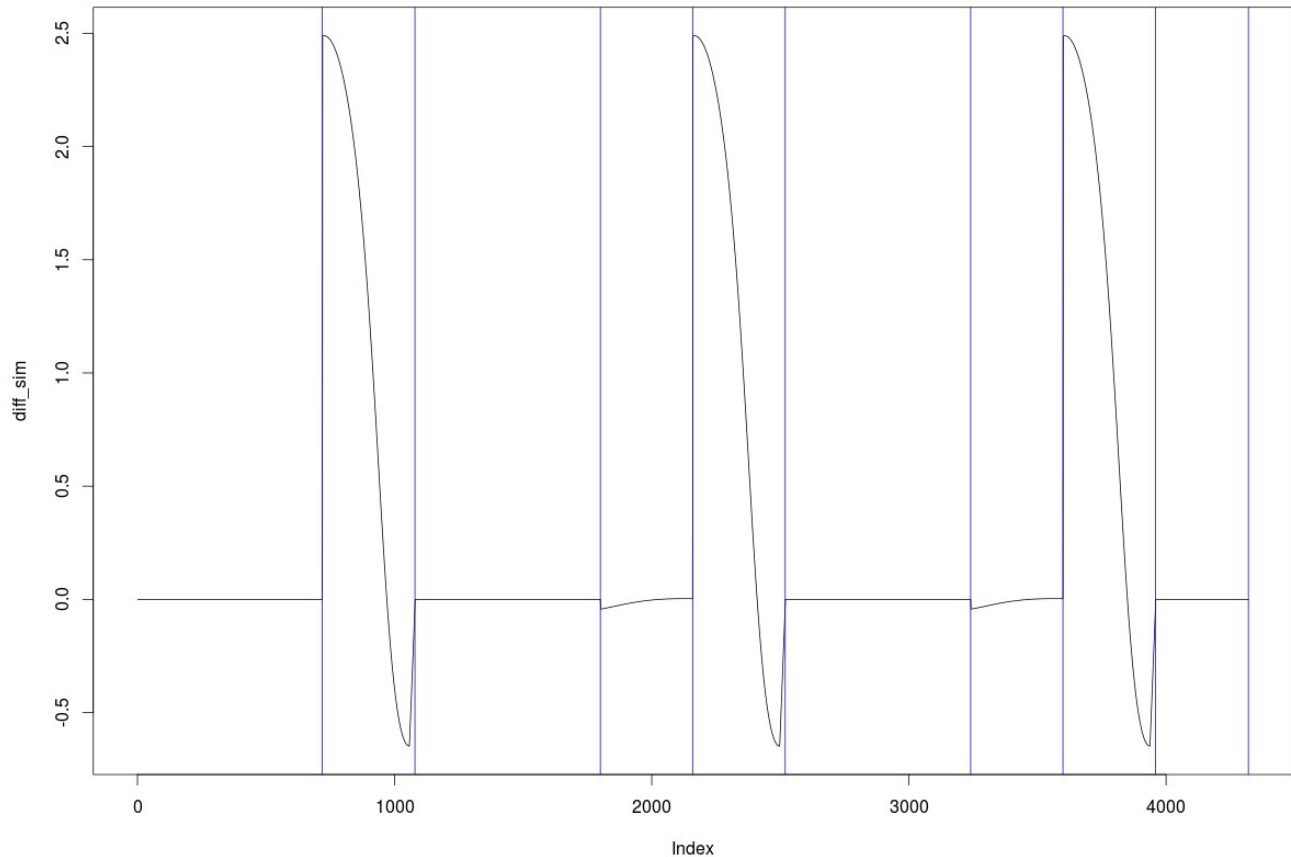
BUT, moving onto something more interesting...

This is the changepoint function called on the plot above (the difference between baseline and foodWeight=10):

```
bp1 <- cpt.meanvar(diff_sim, penalty="None", method="SegNeigh", Q=9)
```

```
plot(diff_sim, type="l")
```

```
abline(v=bp1@cpts, col="blue")
```



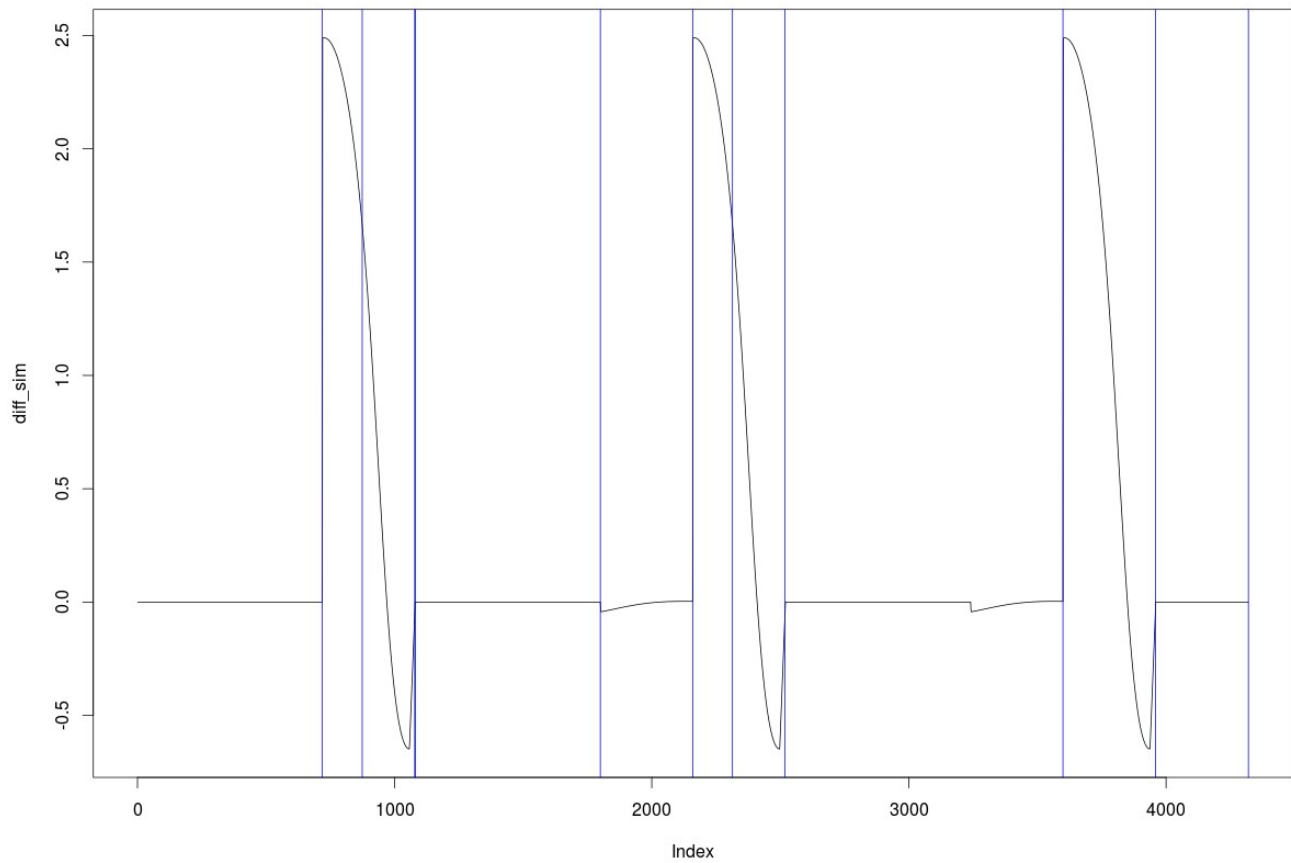
These breakpoint lines line up super well!

Here is the same function, but using a different method (BinSeg, not SegNeigh)

```
bp2 <- cpt.meanvar(diff_sim, penalty="None", method="BinSeg", Q=10)
```

```
plot(diff_sim, type="l")
```

```
abline(v=bp2@cpts, col="blue")
```



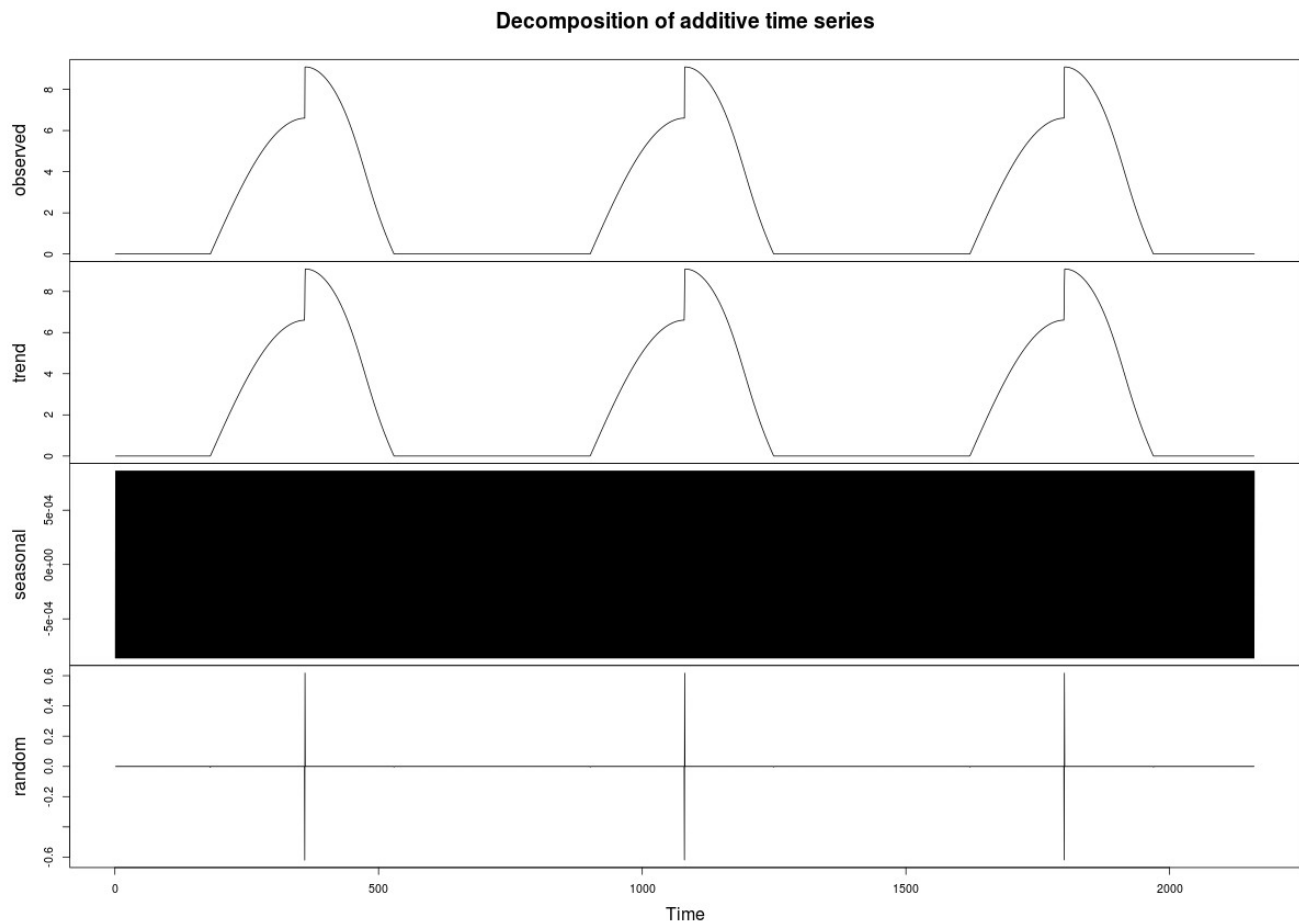
So this has all been run on the difference between the baseline model and one with foodWeight=10.

Check this out though...

```
ts_sim2 <- ts(sim2$Oxygen, frequency=2) #I make the second simulation  
(foodWeight=10) a time series with frequency of 2 (for no particular reason)
```

```
decomp_sim2 <- decompose(ts_sim2) #decomposition/detrending
```

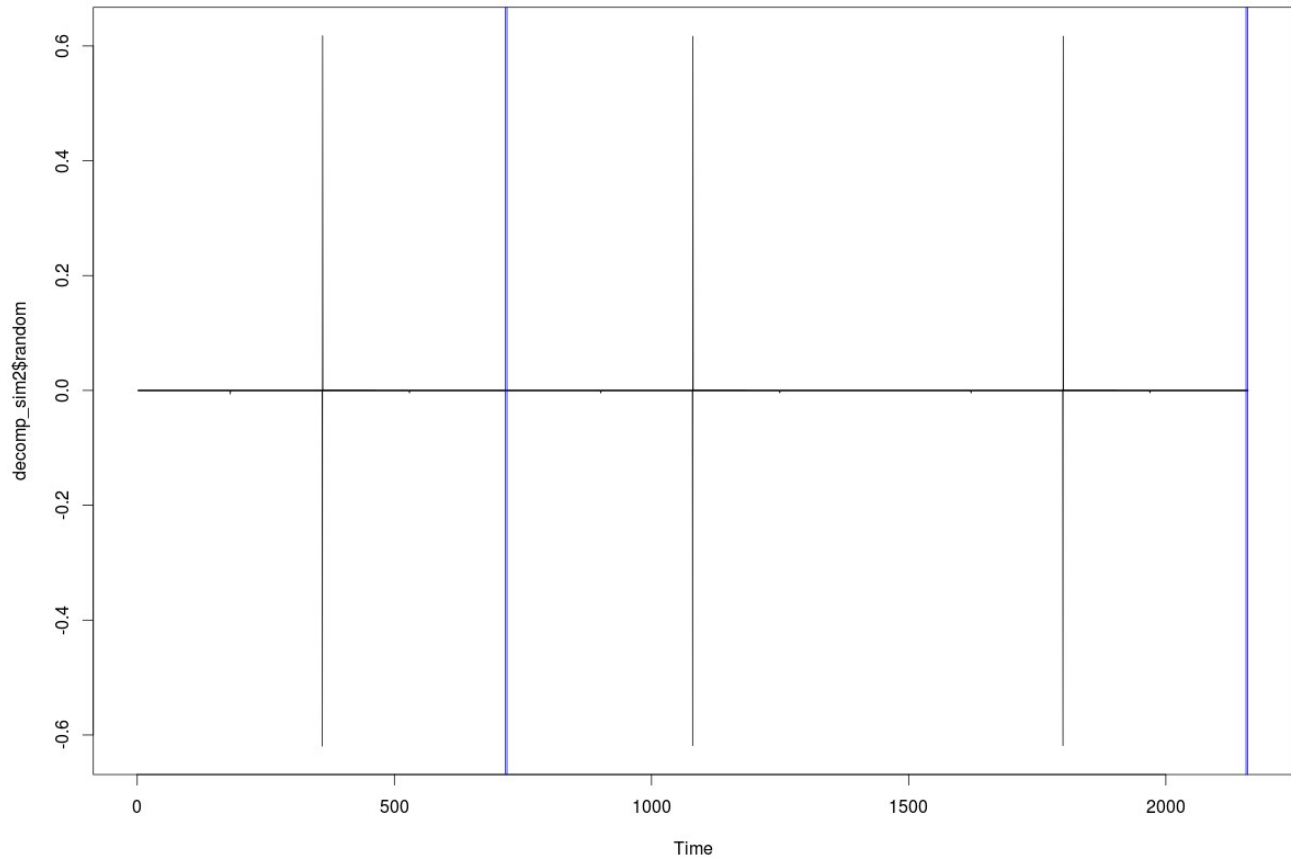
```
plot(decomp_sim2)
```



```
bp3 <- cpt.meanvar(decomp_sim2$random[3:4319], penalty="None",  
method="BinSeg") # run on the residuals from the plot above (index is offset  
because NA values pad the vector)
```

```
plot(decomp_sim2$random)
```

```
abline(v=bp3@cpts, col="blue")
```

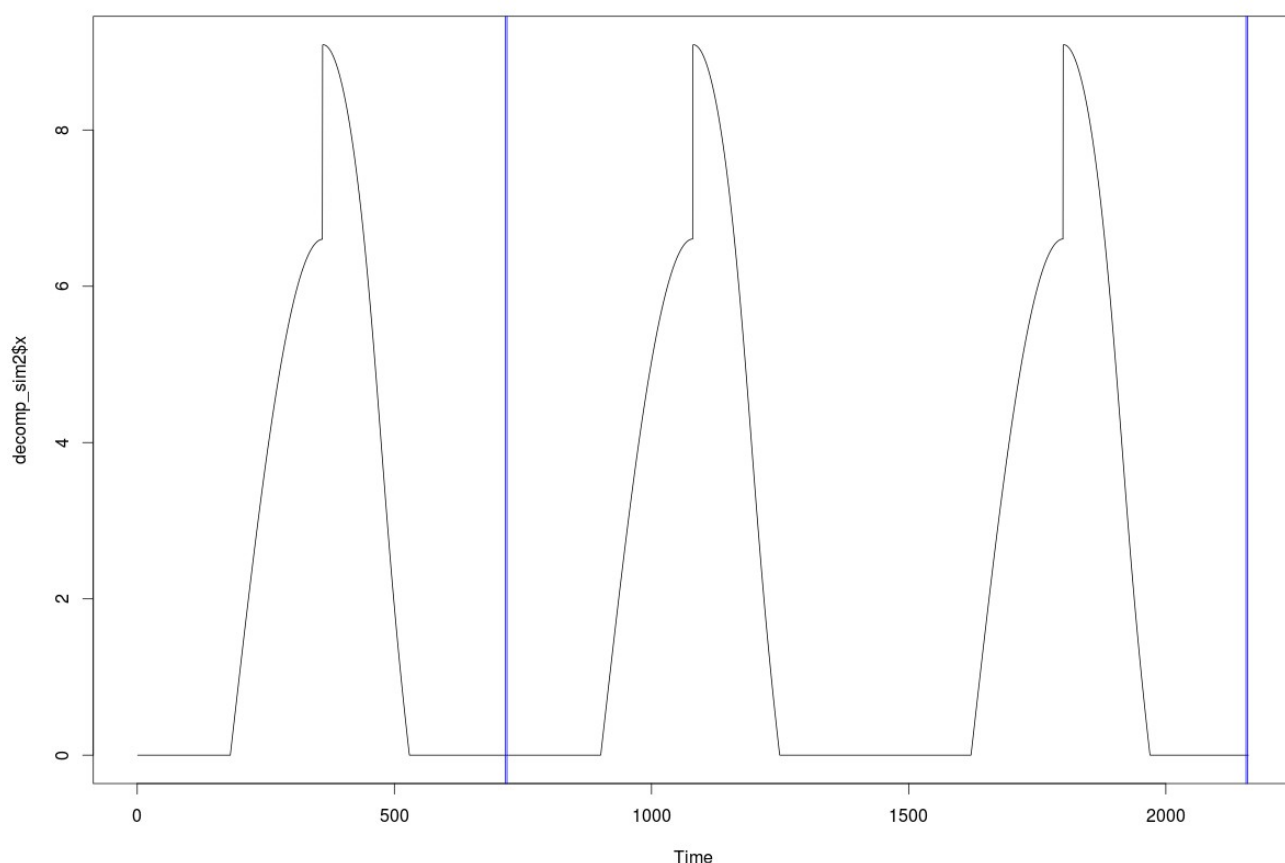


This is changepoints from the residuals, plotting against the residuals. It looks like crap. But....

```
bp3 <- cpt.meanvar(decomp_sim2$random[3:4319], penalty="None",  
method="BinSeg") # the same function call as above
```

```
plot(decomp_sim2$x) # this is the "observed" data, taken from the decompose  
function
```

```
abline(v=bp3@cpts, col="blue")
```



I had hope the changepoints derived from the residuals would line up, and when plotting the decomposed "observed" data they don't.

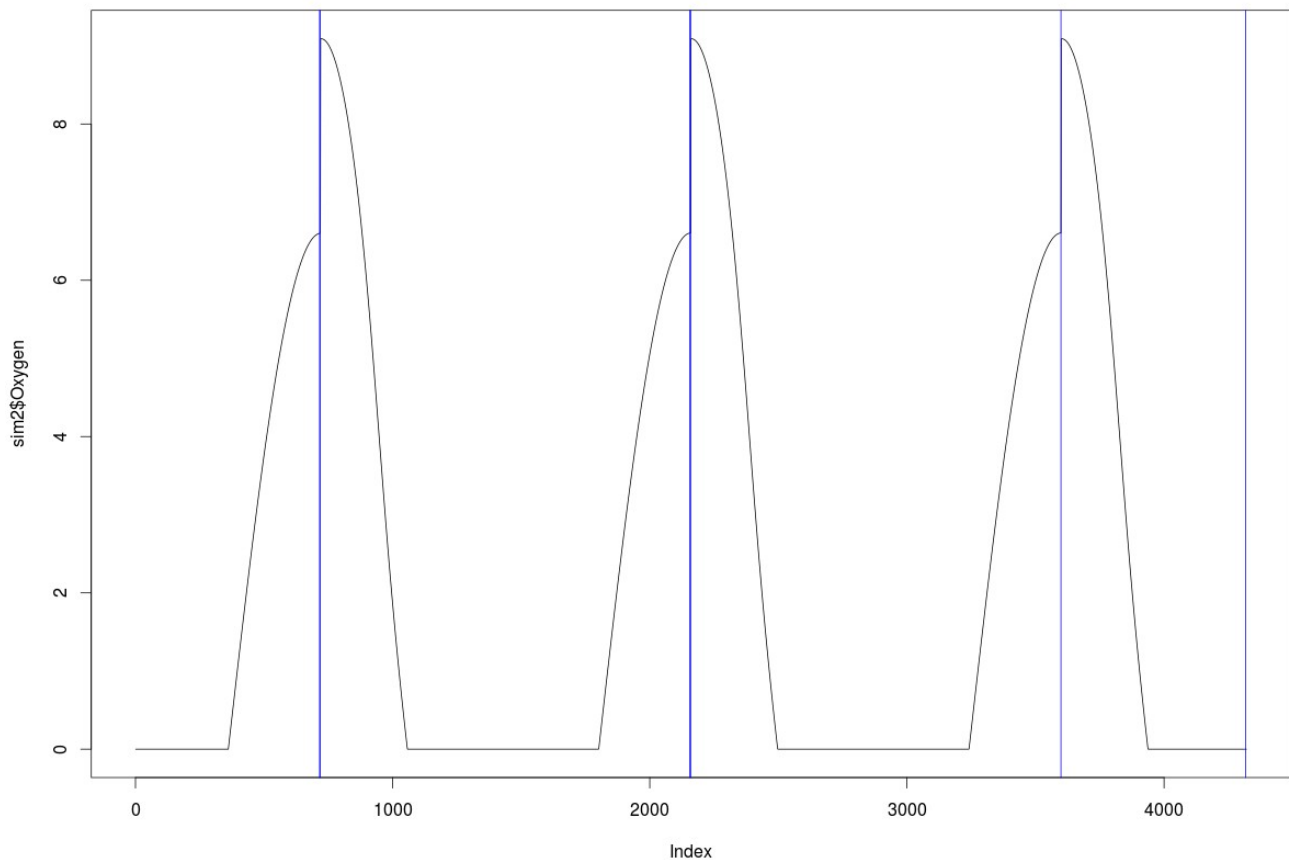
I noticed however that the plot above is missing some changepoints. Then I noticed the x axis is about half of what it should be. I realized that's because when I converted the original simulation data into a timeseries object, I used a frequency=2 - which halved the original length of the number of observations (which should be 4320 - 3 days). As the frequency argument goes up, the length of the "observed" data is reduced.

That may not have been a clear explanation, but it doesn't matter because when I plot the changepoints derived from the residuals against the original simulation data, I get this:

```
bp3 <- cpt.meanvar(decomp_sim2$random[3:4319], penalty="None",  
method="BinSeg")
```

```
plot(sim2$Oxygen, type="l") # simulation with foodWeight=10!!
```

```
abline(v=bp3@cpts, col="blue")
```



DID I DO IT? What do you think?