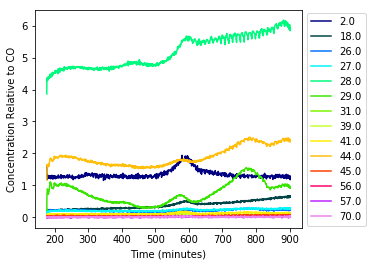
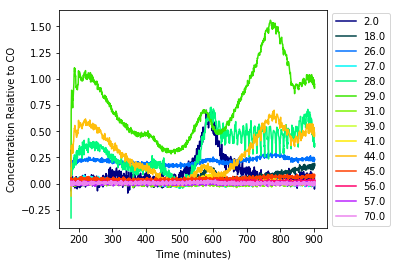
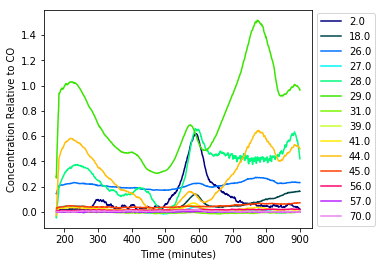
Raw --- the axis name needs to change. It’s counts/s.



Baseline corrected



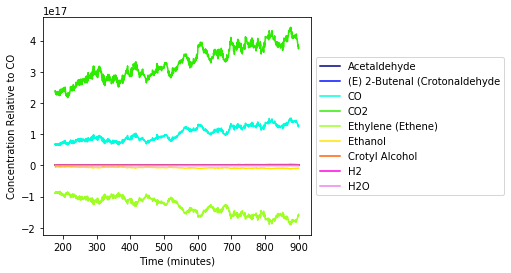
Smoothed



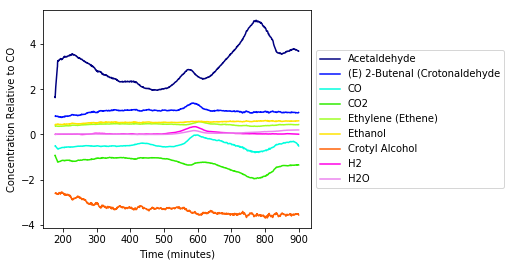
**Now below compares different analyses.**

If going to choose a square matrix, it matters a lot which one you choose.

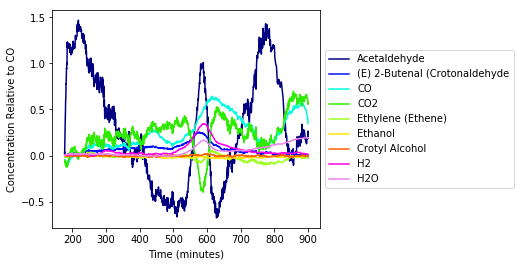
Average of Matrices (inverse w/o distinguished)



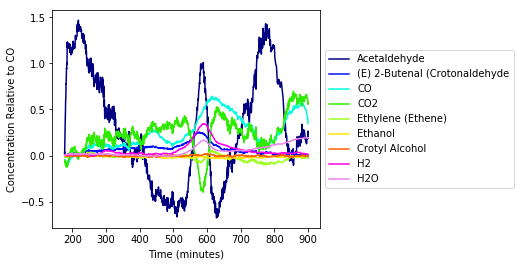
Distinguished of Matrices, distinguished = 'yes' . Now much more reasonable overall, but notice the negative signals for CO2 and things like that. Still not sufficiently reasonable.



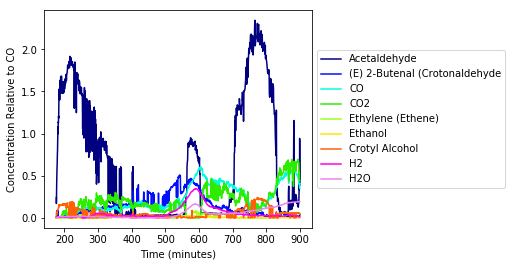
SLS Unique w/ inverse finisher (finisher wasn’t needed, analytically solvable w/ fragments used).



SLS Unique w/ brute finisher (same, b/c analytically solvable).



SLS Unique with Negative Analyzer (takes about 5 minutes)

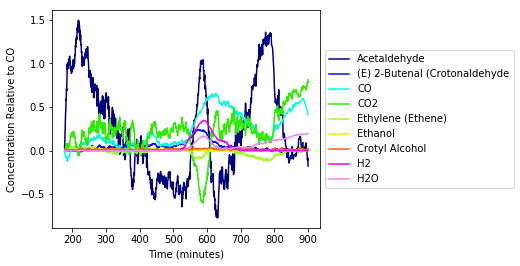


SLSUnique, brute finisher, below options.

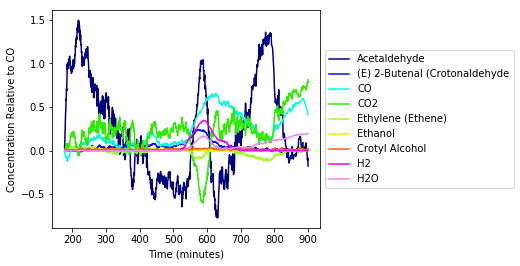
lowerBoundThresholdPercentage = [0.25] # 1.0 is max value. leave as [ ] to only use the absolute threshold. Always include a decimal.

lowerBoundThresholdAbsolute = [0.01] # leave as [ ] to only use the percentage threshold. Always include a decimal.

lowerBoundThresholdChooser = 'yes'



Same w/ inverse:



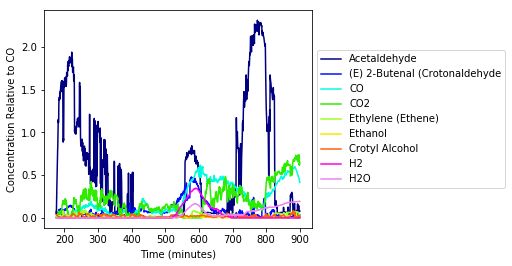
Now w/ Negative analyzer on again.

SLSUnique, brute finisher, below options.

lowerBoundThresholdPercentage = [0.25] # 1.0 is max value. leave as [ ] to only use the absolute threshold. Always include a decimal.

lowerBoundThresholdAbsolute = [0.01] # leave as [ ] to only use the percentage threshold. Always include a decimal.

lowerBoundThresholdChooser = 'yes'



**I think we need to apply the smoother on this too… as final step in paper for SoftwareX journal.**

**Need to add “FinalConcentrationSmoothing” or something.**