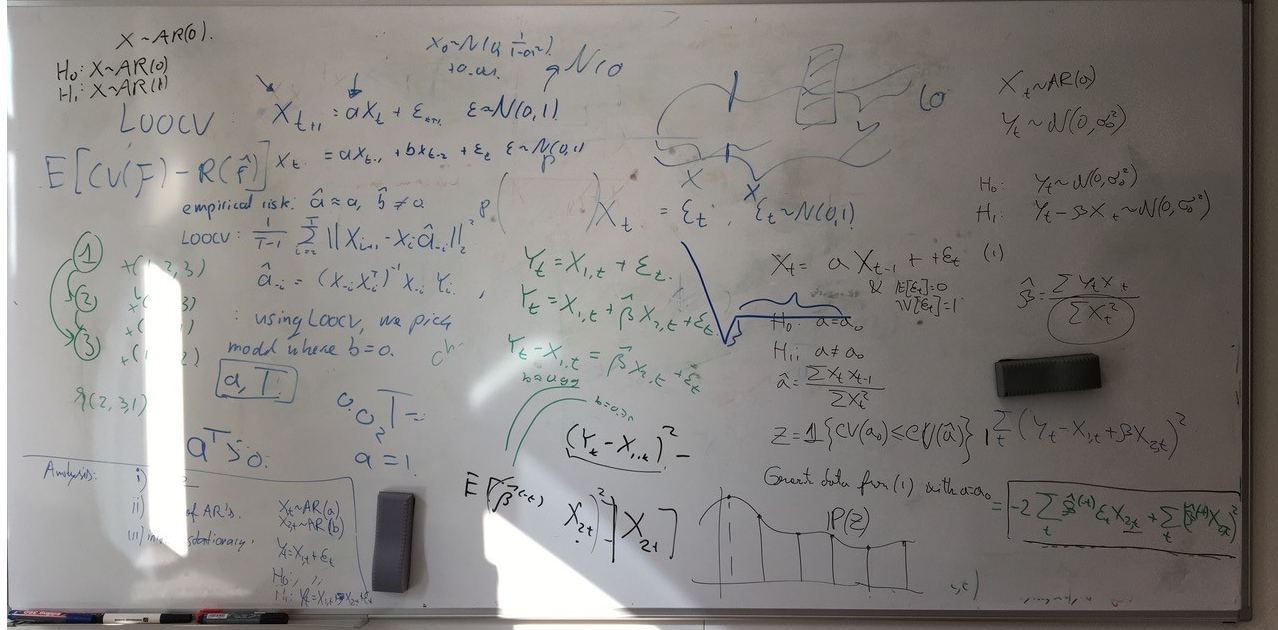
Week 27



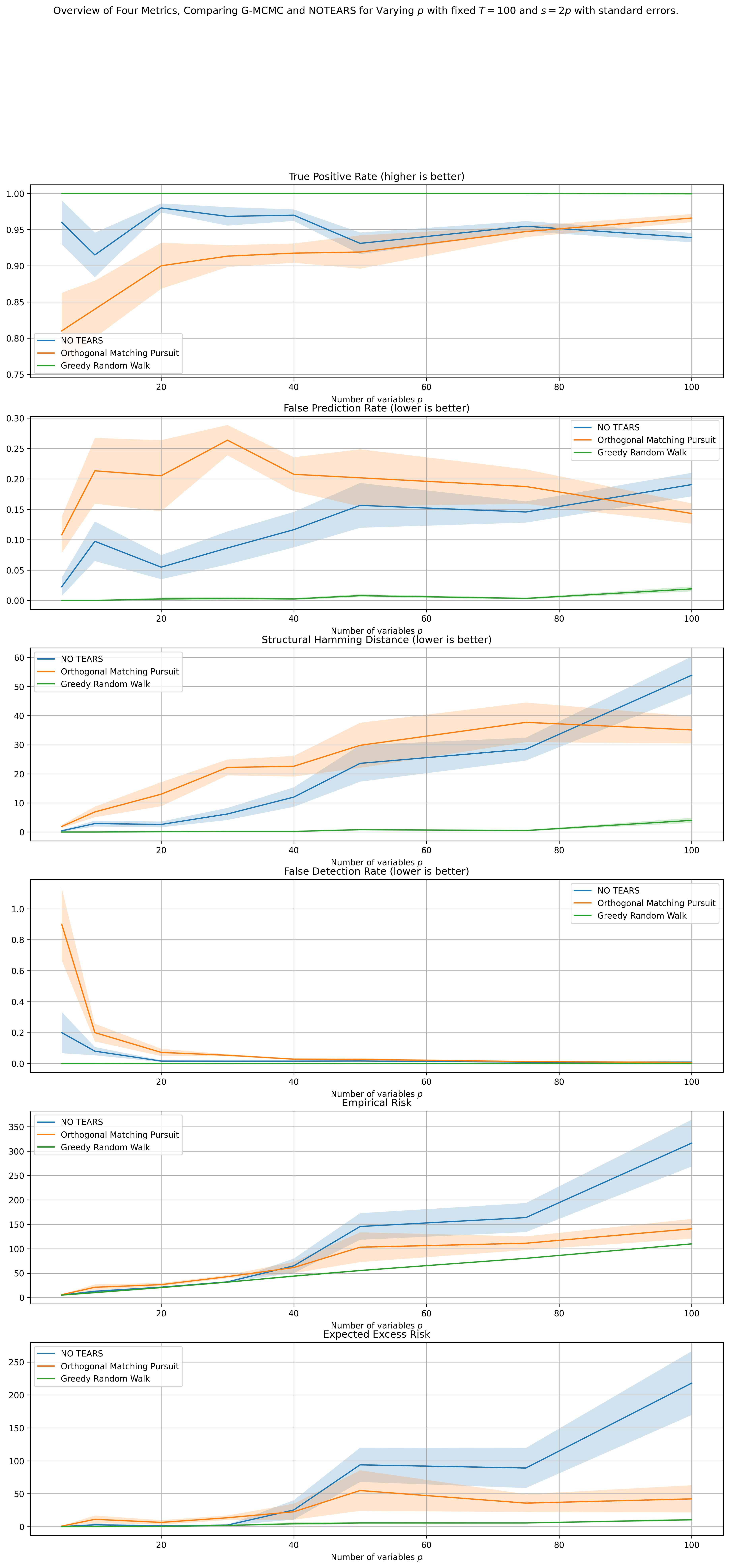
# SIOUX Presentation

Curious about Alex’ opinion, what went good what went bad? Took of course some time so that is why not that much content now.

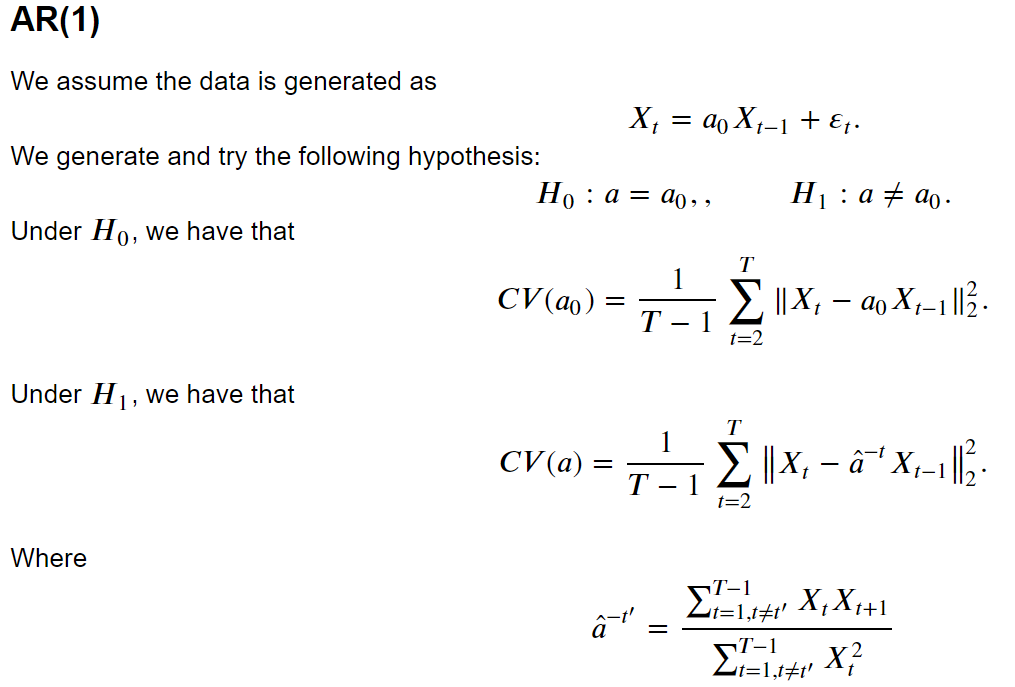
# Greedy Random Walk (MCMC Approach)

Redid the experiments for Sioux, did OMP, NO TEARS, and Greedy MCMC. Results were interesting and in the advantage of Greedy MCMC and OMP!

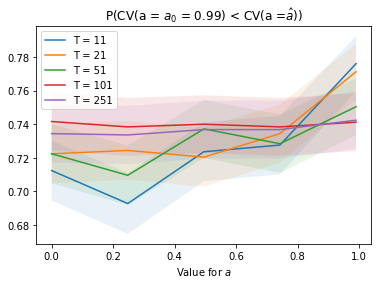
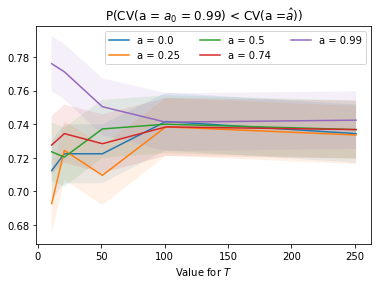
Talk with Jeroen Riels: Spoke with Jeroen about doing double Master, he was very interested so that was nice to see.



# AR(1) Estimate a

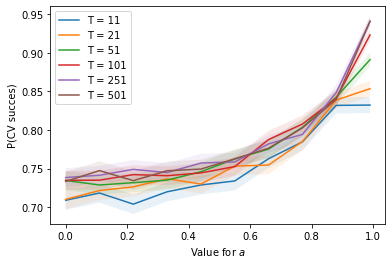
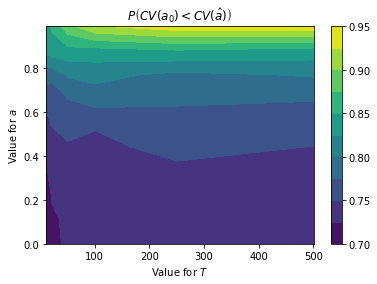
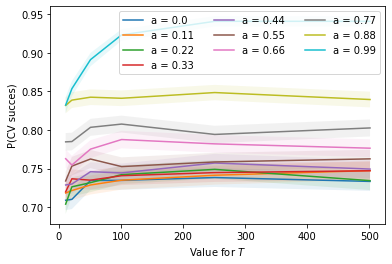


**Using correct stationary distribution N(0, 1 / (1 – a^2))**

****

It seems as if the probability of succes remains constant for large values of T. For smaller values of T, it seems that a large value for *a*, so more dependency, increases the probability of success.

**Using incorrect stationary distribution N(0, 1 / (1 – a^2)^2)**

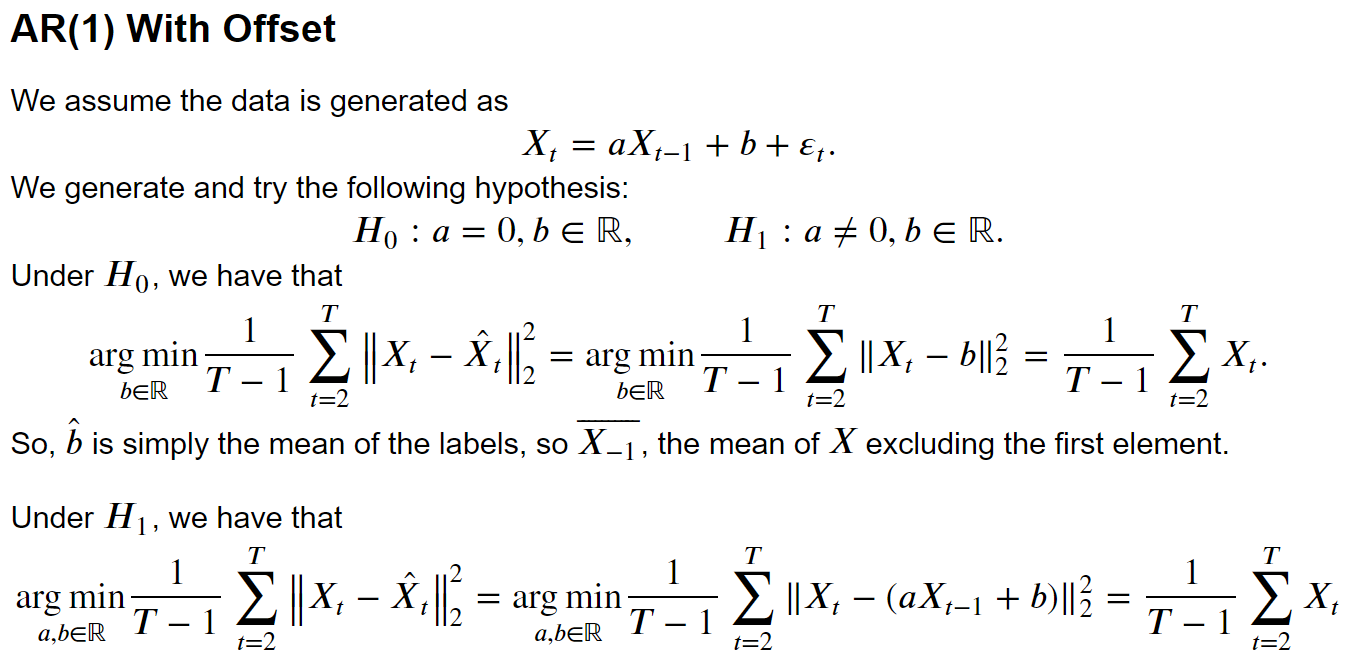
 

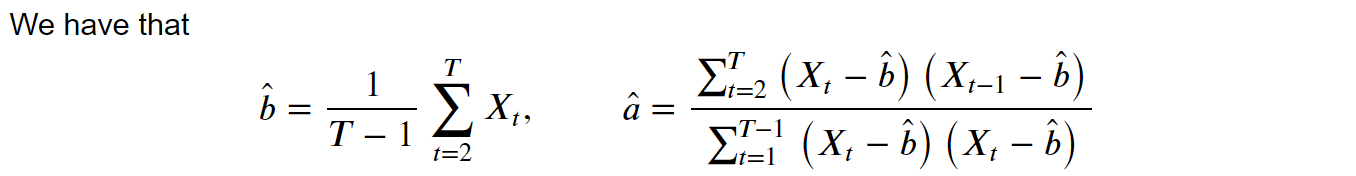
Variables that can influence this are the dependency in the AR(1) model, *a*, and the number of samples, T. You would think that the larger *a* is, the stronger the dependency is and this helps in determining between the true and OLS estimate.

Also, the larger T is, it seems as if it is slightly easier to distinguish this, albeit not by much.

# Offset

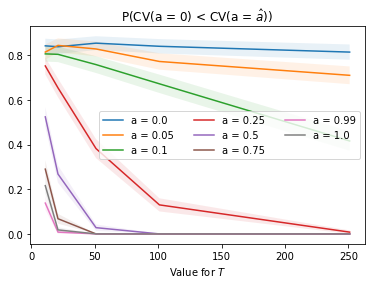
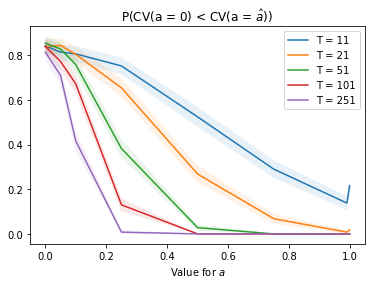
## Compare H0: a = 0, b in R versus H1: a =/= 0, b in R.





Variables: offset b, dependency a, number of time steps T.

Not really a good one. As soon as our *a* is non-zero, we easily determine this. For small *a*, we are more in the middle.

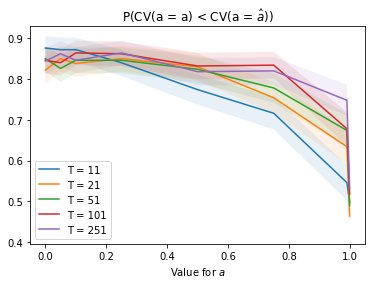
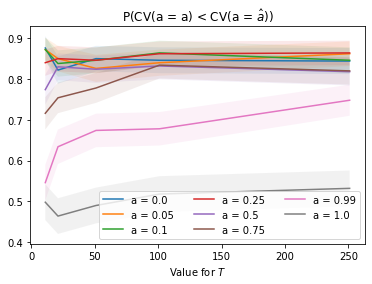
’

If a is close to zero, then doing cross validation helps in determining whether a is zero or a needs to be estimated. However, if a increases then the hypothesis that a is equal to zero will lose ground of course. And eventually go towards zero. In fact, if *T* is larger, then we are able to tell make this conclusion quicker.

## Compare H0: a = a0, b in R. H1: a =/= a0, b in R.

Value of b does not influence anything, as both are estimated in the same way.

It seems that the larger *T* is, the better we are in making the correct decision, i.e., determining whether *a* was indeed the data generating value. Nevertheless, as *a* gets larger, it gets significantly more difficult to make this decision correctly, up to the point where it is basically a coin toss.

# Dependent Noise

## Noise as an AR(1) model itself.

Interestingly, the cross validation breaks down when the noise becomes dependent. The noise is an AR(1) with coefficient a.

When we shuffle, we remove the dependency and the probabilities increase again.

**Different *a* as a function of *T***

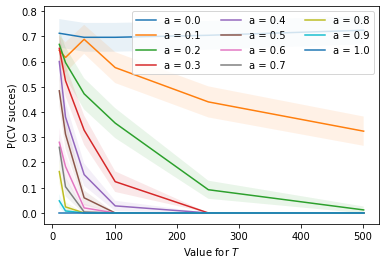
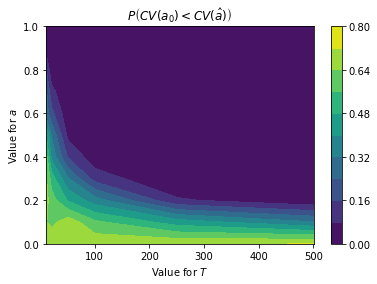
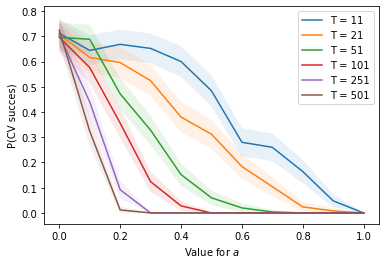
The probability of success diminishes to zero when the noise is dependent. Note that for *a = 0.0*, we do indeed have independent noise, and for larger values of T, this barely matters. For *a > 0.0*, we see that as *a* gets larger, the probability of success decreases. Furthermore, as *T* increases, the probability of success will also decrease.

**Different *T* as a function of *a***

The probability of success diminishes to zero when the noise becomes more dependent. Furthermore, this effect is strengthened for larger values of *T*.

The larger the dependency, the less effective the cross validation is. The more samples we have, the less effective the cross validation is.

P(CV success) = P(CV(a = a0) < CV(a = â)), where a0 = 0.50.

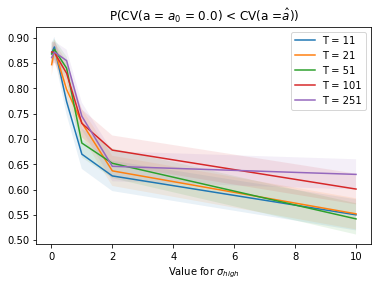
## Heterogeneous Noise

The noise is Gaussian, but not homogeneous. At t = 1, we have unit variance. At t = T, we have variance of 2 (or another value > 1). We interpolate at values in between.

We see that the probability of success decreases, but will not go down below 0.5.

Also, when we shuffle the noise, the dependence of course is gone, and the results get a little better.

### *a = 0.00, X is random noise.*



It seems that the value for *sigma\_high* matters greatly. For large values of *sigma\_high*, determining the correct value for *a* using cross validation becomes very difficult.

What is very interesting, is that there is no dependency for *sigma\_high* *= 1.0,* so we would expect the largest success rate there. However, values of *sigma\_high* smaller than 1.0m the probability of success increases, which I find very odd.

It seems as if the value for T barely matters. The more data we have available, does not result in a better cross-validation result.

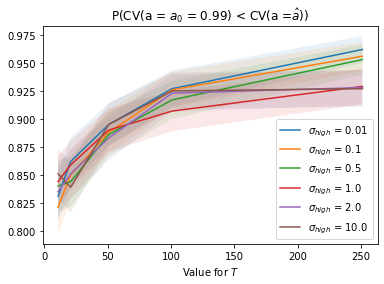
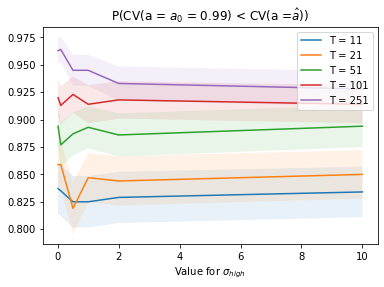
### *a = 0.50, X is dependent on its past.*

### 

Interestingly, the value for *T* seems to barely influence the probability of success.

The value for *sigma\_high* seems to decrease the probability of success. The larger the value for sigma\_high, the larger the dependency between the noises and hence, the less effective cross-validation is.

### *a = 0.99, X is heavily dependent on its past.*

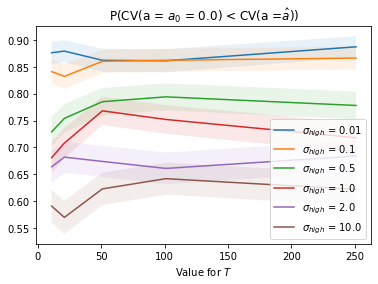


We now see something very strange here. Suddenly, the choice of *T* begins to matter, whereas the choice of *sigma\_high* does not matter anymore?!

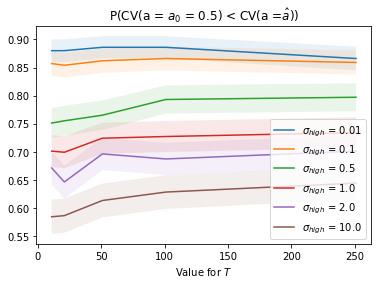
Important to check: Initialization! The AR(1) model should be initialized in a stationary distribution.

**Correct One Below**

### *a = 0.00*



### *a = 0.50*



### *a = 0.99*

