Today it’s my pleasure to introduce, Professor Ali Shojaie. Professor Shojaie holds master’s degrees in industrial engineering, statistics, applied math, and human genetics. He earned his PhD in statistics from the University of Michigan. His research focuses on the high dimensional data, longitudinal data, computational biology, network analysis, and neuroimaging. Professor Shojaie is a 2022 fellow of the American Statistical Association and 2022 winner of their Leo Breiman Award. He’s a full professor of biostatistics, adjunct professor of statistics, and the associate chair for strategic research affairs in the department of biostatistics in the University of Washington. Let’s welcome Professor Shojaie. Sometimes I get moved by the volume of my voice. You guys, can you hear me at the back, okay? Since I’m not gonna use the microphone yet, but I’d rather not use the microphone at all. Well, it’s a pleasure to be here and to talk to you about some work that I’ve been doing for the past couple of years.
I’m using machine learning tools for different types of data.

The question really is how do we process information on our brains?

What is the processing information?

The brain through neurons, we know that neurons interact with each other.

This is of course related to my broader interests on network and understanding how things interact.

Naturally I was drawn into this part here, but when I talk to scientist colleagues, then a lot of times I’m asked, what is the goal of understanding that network?

How do we use it?

How do we take advantage of that network that we learned?

Here’s an example of some recent work that we’ve been doing that indicates that learning something about these networks is actually important.

Should say that this is joint work.

I have colleagues at the University of Washington that has done that is biomedical engineering.

And then I’m collaborating with E Shea-Brown...
who’s in computational scientist, and Z Harchaoui, computer scientist slash statistician, and she’s been working on this project. This project, the lab is interested. And what they do is neurostimulation. What they wanna do is to see if they could stimulate in different regions of the brain to make in this case monkey do certain things or to restore function that the monkey might have lost. And it’s a really interesting platform that they’ve developed. It’s basically small implants that they put in a region of the brain on these monkeys. And the implant has two areas when the lasers beam shine in about 96 in this case, electrodes that collect data in that small region of the brain. This is made possible by optogenetics meaning that it made the neurons sensitive to these lasers. When neurons receive the laser, then they basically get excited, get activate. The goal in this research eventually is to see how the activation of neurons, which plasticity would change the connectivity of the neurons, would result in later on in changing function.
That’s the eventual goal of this.

This research work at the very beginning of that.

We are not there yet in terms of understanding function,
understanding the link, the connectivity and contact.

The collaboration with this lab started when they wanted to predict how the connectivity changes.

The way that the experiment is set up had these times where they have activation and then the latency period and then followed by observation.

They basically observe the activity of these brain regions.

That sort of 96.

Electrodes in this main region over time.

That’s the data that they’re correct.

Here’s a look at this functional connectivity a and that’s what they were trying to predict.

Basically the heat map shows the links between the various brain lesions,
but 96 of them, you don’t wanna.
And if that connectivity is defined based on coherence, which is basically correlation measure frequency domain, and we have coherence in four different frequency bands. These are the standard bands that signal instructive and they think that they measure activity and different spatial resolution. We have theta band, the beta band, the gamma band, and the high gamma band. And we wanna see how the connectivity in these different bands changes as the effect of these type neurons. This is not working.

What basically we have is that we have the baseline connectome and we have these experimental protocols, and we’re trying to predict how the connectivity changes. What the lab was doing before was that they were looking at trying to predict connectivity based on experimental protocols. And what they were getting was actually really bad prediction.
These are test R squares. And what they were getting was about 5% test R square when they were using these protocol features to predict how to connect with these genes. And the first thing that we understood and so you see it that sort of really bad is that that’s the prediction. If that’s the prediction that you’re getting, then really bad prediction. The first thing that we noticed in this research was that it’s actually important to incorporate the features of the current state of connectivity in order to predict how it’s gonna change. What we did was that in addition to those protocol features, we added some network features, the current state of the network in order to predict how it’s gonna change. And this is, to me, this is really interesting because it basically says that our prediction has to be subject specific depending on the current state of each month these connectivity, how their connectivity is going to change will be different. And what we saw was that when we incorporated these network features, we were able to improve quite a bit in terms of prediction. We’re still not doing hugely good,
we’re only getting like test R squared of what, 25%.

But what you see that sort of the connectivity is now, the prediction is now much more.

How the connectivity.

And also in terms of the pictures, you see that going from,

so say this is the true, the first part in d is the true change in connectivity,

e is what you would get from just the protocol features,

and you see that prediction is really bad,

and f is what you get when you combine protocol features

and the network features.

That prediction is closer to the true change in connectivity than just using the protocol feature.

This was the first thing that we learned from this research.

The second part of what we learned is that it also matters which approach you used the prediction.

What they had done was that they were using some simple linear model for prediction.

And then we realized that we need to use something more expressive and then we sort of ended up using these non-linear additive models

that we had previously developed,

partly because while they have a lot of expressive power,
they're still easy to interpret.

Interpretation for these additive models is still easy and particularly we see what the shapes basically these functions are. For example, with the distance we see how the function changes and that helps with the design of these models.

I'm not gonna spend too much time talking about the details of this given that we only have 50 minutes and I wanna get to the main topic, but basically these additive models are built by combining these features. Think of tailor expansion in a very simple sense that you have a linear term, you have a quadratic term, you have a cubic term. And the way that we form these additive models is that we automatically select the degree of complexity of each additive feature, whether it’s says linear, or quadratic, or cubic, etcetera. We also allow some features to be present in the models, features not to be present. What we end up with are these patterns where some features are real complex and other features, and that’s automatically decided from data.
This model is good in this prediction and it allows us to come up with these sets of predictions. We see now that for example, for coherence difference, which is the network feature, that’s the coherence difference. Network distance, that’s the distance between the two portals. The two laser points. We get these two patterns estimated and then when we combine them, we get this surface basically that determines how the connectivity, changing connectivity could be predicted based on these two features. And all of this is done automatically based on data. This approach, again, sort of the key feature of it is that it combines the network features of the current state of connectivity with protocol features in order to do a better job of prediction. This is a research that we just started and we will continue this research for the next at least five years. But the goal of it is eventually to see if we could predict the function if we could build a controller that we could determine how to change function based on various features of the experiment.
I mentioned all of this to say that knowing and learning the network matters. We need to learn the current state of connectivity, for example, in this work in order to be able to design experiments that would hopefully help and restore function. Now in this particular work, what we did was that we used a very simple notion of connectivity. We used coherence, which is basically correlation, but we know that that’s not always the best way to define connectivity between ranges. And so what I wanna talk about for the remaining 40 minutes or so is how do we learn connectivity between neurons? And this is using a different type of data that I had thought about before, and I’m hoping that so I could show you this clip, which is that shows the actual raw data. The data is actually a video. And this is activity of individual neurons in a small region of the brain. These dots that you see popping up, these are individual neurons firing over time. And you see that sort of neuron fires and other neuron fires, et cetera, et cetera. That’s the raw data that we’re getting.
And the goal is to understand based on this pattern of activation of neurons, how neurons talk to each other basically.

Now I’m gonna go back here.

And so the data of that video that I showed you, basically, here’s some snapshot of that data.

Here’s one frame. And there’s a lot of steps in getting this data to place it a bit more quick.

Were not gonna talk about this, but sort of we need to first identify where the neurons are.

No one tells us where the neurons are in that video.

We need to first identify where the neurons are.

We need to identify when they swipe, when they fire.

No one tells us that either.

There’s a lot of pre-processing step that happens.

The first task is called segmentation,

identifying where the neurons are,

then spike detection, when the nuance fire over time,

when which individual neuron fires over time.

And that none of these is a trivial task.

And then a lot of smart people are working on these,

including some of my colleagues.

After a lot of pre-processing,

so you end up with each individual neuron,
you end up with a data point, like data set like this.

that it basically has these takes whenever the neuron has fired.

A given neuron you have over time that the neuron fire

like this.

These are the time points the neuron apply.

Now, you can do something fancier, you can look at the magnitude,

the signal that you’re detecting at neuron. You could deal with that, but for now we’re ignoring that.

We’re just looking at when they fire.

This is called the spike train for each neuron.

That’s the data that we’re using.

These are neurons firing times.

And if we combine them, this is the cartoon.

we get something like this.

We get a sequence of activation pattern.

This is color coded based on that sort of five neuron

sort of cartoon network.

And you see that different neurons activate at different times.

And what I’ll talk about is a notion of connectivity.

that tries to predict the activation pattern of one neuron

from a network, basically.

That sort of maybe neuron one tells us something

about sort of activation patterns in neuro two,
that if we knew when neuron one activated or fired,
we could predict when neuron on two fires, and maybe neuron two will tell us something about activations of neurons three and four, et cetera.
And that’s the notion of connectivity at that time after, since we’re trying to estimate those edges in this time.
Now, please.
Could you say just a few words informally about the direction of connectivity?
Yeah.
Maybe drawing arrow forward in time.
Yes.
I’ll get to this, maybe in the next two slides.
The framework that we’re gonna work with is called the Hawkes process.
Just go back to seminal more by Alan Hawkes.
In ’70s where he looked at spectral properties of point processes.
What are point processing that basically is like activation over time.
Zeros and ones over time.
It could Poisson processes.
What the Hawkes process does in particular is that it uses the past history of one neuron to predict the future.
And this goes back to Forest’s question that sort of what is that edge in this case? This is the notion that is related closely in a special case. What makes this Hawkes process the convenient for this is that sort of it’s already set up to do this. I’m gonna present the Hawkes process. Its simplest form, this is the linear Hawkes process. And what it is, is that sort of, it’s a counting process. It’s just counting the events. And so that’s the event process N. And that event process has an intensity lambda j for each neuron is standard i, which is combination of two terms, a new I, that’s the baseline intensity of that neuron. That means that if you had nothing else, this neuron would fire at this rate, but basically random that would fire at random rate plus the effect that that neuron gets from the other neurons. Every time that there’s an activation in neuron,
any neuron \( j \) from one to \( p \) including neuron \( i \) itself, depending on how long it’s been since that activation. The time it’s been, the current time \( t \) and the time of activation of the previous neuron acquiring or the previous neuron, some weight function determines how much influence that neuron \( p_i \) gets. This has a flavor of causality, which is why econometricians call it danger causality. This is worked by the ranger, but it’s really not causality. We know that there’s beyond, and so there’s a lot of work on this that’s sort, it’s only causality on the day-to-day restrictive assumptions, talk about in general, but nonetheless it predicts in the future. It’s a prediction in the future. And again, sort of in this case this \( d \) and \( i \) is our point process, \( \lambda_i \) is our intensity process. It started itself. \( U_i \) is the background intensity and \( t_jk \) are the times when the other neurons acquired in the past. And this \( \omega_{ij} \) is the transfer function. It determines how much information is passed from firing your one neuron
to firing of other neurons in the future. And usually you think that sort of the further you go in the past, the less information is carrying over. Usually the types of functions that you consider, these transfer functions are decay and how to decay form. Usually the types of functions that you consider, these transfer functions are decay and how to decay form. If you go too far in the past, there’s no useful information. Any question on the basic of this linear Hawkes process because I’m not gonna present the more complicated version. I wanna make sure that we’re all good with this simple version. Okay, so no question on this. But if we agree with this and then this actually gives us a very convenient way of defining that connectivity. What it meant by connectivity now basically means that this function omega ij, if it’s non zero, then that means that there’s an edge between neuron j and neuron I. And that’s basically what I was showing you in that bigger module. It all comes down to estimating whether omega ij is zero or not for this Hawkes process.
Okay.

Let me show you a zero simple example with two neurons.

In this case, neuron one has no other influence. It’s only it’s past history and baseline intensity.

Neuron two has an edge on neuron one.

Let’s see what we would expect for the intensity of neuron one.

If we think about neuron one, then it’s basically a baseline intensity, that new one.

And it’s gonna fire at random times for some process.

It’s gonna fire at random times with the same intensity.

The intensity is not gonna change because fixed.

we could allow that intensity to be time varying, et cetera,

make it more complicated but in it simplest form

that neuron is just gonna fire randomly,

every time that they sort of it wants.

Now, neuron two would have a difference story because neuron two depends on activation of neuron one.

Any time that neural one fires, the intensity of neuron two

goes from, let’s say the baseline is zero for neuron two,

but every time that neuron one fires,

the intensity of neuron two becomes non zero
because it got excitement from neuron one. It responds to that. Neuron two would require to, and then when you have

like three activations, you can get the convolution of effects that would make neuron two more likely to activate as well or to spike as well.

And then so this is a pattern that sort of basically what we are doing here is that we’re taking this to be on omega this to be on omega to one, that sort of this you see there’s the K form and these get involved if you have more activation on neuron one, that sort of increases the intensity of neuron two, meaning that we have more of a chance for neuron two to fire and this.

Say this simple example, this could be the intensity of neuron two. And in fact this all we observe in this case are these two spike trains for neuron one and neuron two. We don’t observe the network, in this case there are four possible edges. One of them is the right edge. We don’t observe the intensity processes. All we observe is just the point process, the spike.
And the goal is to estimate the network based on that spike train. In fact, as part of that, we also need to estimate that process. That estimation problem is not actually that complicated. If you think of it, it’s trying to predict now based on past. We could do prediction. We could use basically penalized regression. It’s a penalized Poisson regression. Something along those lines. A little bit more complicated, but basically it’s a penalized Poisson regression and we could use the approach similar to what is known as neighborhood selection. We basically meaning that we regress each neuron on the past of all other neurons, including that neuron itself. It’s a simple regression problems. And then we use regularization to select a subset of them that are more informative, et cetera. And there’s been quite a bit of work on this, including some work that we’ve done. The work that we’ve done was focused more on extending the theory of these Hawkes processes to a setting that is more useful for neuroscience applications.
In particular, the theory that existed was focused mostly on the simple linear functions, but also on the case where we had non-negative transfer functions. And this was purely an artifact that the theoretical analysis approach that Hawkes had taken and using these what are known as cluster representation.

What Hawkes and Oakes had done was that they were representing each neuron as a sum of, sorry, homogeneous Poisson processes, activation pattern of each neuron as some of homogeneous Poisson process. And because there was a sum that could not allow for omega ijs to be negative, 'cause they would cancel throughout and we would get less. What we did, and this was the work of my former student, Chen Chang who’s Davis, was to come up with an alternative framework, theoretical framework motivated by the fact that neuroscience activations are not just positive, they’re not all excitement, they’re also inhibitions happening. Neuroscience and in any other biological system really, we can’t have biological systems being stable.
without negative feedback.
These negative feedback groups are critical.
We wanted to allow for negative effects or the effects of inhibition.
And so we came up with a different representation based on what is known as thinning process representation.
that then allowed us to get a concentration for general.
I won’t go into details of this, that basically we get something that we can show.
that for any sort of function, we get a concentration around its need in a sense.
And so using this as an application, you could show that sort of with high probability,
we get to estimate the network correctly using this name of selection type approach.
This is estimation but we don’t really have any sense of whether...
Let’s skip over this for the sake of time.
You don’t really have any sense of whether the edges that we estimate are true edges or not.
We don’t have a measure of uncertainty.
We have theory that shows that sort of the pi should be correct.
but we wanna maybe get a sense of uncertainty about this.
And so the work that we’ve been doing more recently focused on trying to quantify the uncertainty of these estimates. And so there’s been a lot of work over the past almost 10 years on trying to develop inference for these regularized estimation procedures. And so we’re building on these work, existing work in particular, we’re building on work on inferences for vector risk processes. However, there’s some differences most importantly that vector risk processes capture a fixed and pre-specified lag, whereas in the Hawkes process case, we have each basically dependence over the entire history. We don’t have a fixed lag and it’s all pre-specified. And also another difference is that vector auto-aggressive processes needs pardoning. Its’ observed over this free time, whereas the Hawkes process is observed over a continuous time. It’s a continuous time process and that adds a little bit of challenge, but nonetheless, so we use this de-correlated score testing work which is based on the work of Ning and Liu.
And what I’m gonna talk about in the next couple of slides is an inference framework for these Hawkes processes. Again, what I showed you before, the simple form of linear Hawkes process and motivated by your neuroscience applications, what we can consider is something quite simple, although, we could generalize that. And that generalization is in the paper but the simple case is to consider something like omega ij as beta ij times some function pathway j where that function is simply decay function over time.

It’s like exponentially decaying function. It’s class decay function. That’s called a transition for neuroscience applications. And so if we go with this framework then that beta ij coefficient determines the connectivity for us, that this beta ij, if it’s positive, that means that sort of there’s an excitement effect. If it’s negative, there’s an inhibition effect, and if it’s zero, there’s no influence from one or data. All we need to do really is to develop inference for this beta ij.

And so that is our goal.
And to do that, I’ll go into a little bit of technicalities and detail of not enough too much. Please stop me if there are any questions. The first thing we do is that we realize that we can represent that linear Hawkes process as a form of basically a regression almost.

The first thing we do is we turn it into this integrated stochastic process. We integrate all the past that form that sort of seemed ugly, we integrate it so that it becomes a little bit more compact.

And then once we do that, we then write it pretty similar to regression. Now this is something that’s easy and we’re able to deal with. The main complication is that sort of this a regression that depends on the beta lambda. It depends on the beta lambda.
Okay, so once we do this then to develop a test for $\beta_{ij}$, we could develop a test for $\beta_{ij}$ and then this also could extended to testing multiple betas and sort of allowing for ground expansions et cetera.

And even nonstationary the baseline, but the test is basically now based on this de-correlated score test.

Once we write in this regression form, we can take this de-correlated score test and I’ll skip over the details here but basically we form this set of octagonal columns and define a score test based on this that looks something like this, that you’re looking at the effect of the correlated $j$ with basically noise term, epsilon $i$.

Both of these are driven from data based on some parameters, but once you have this, this $S_{ij}$ then you could actually now define a test that basically looks at the magnitude of that $S_{ij}$.

And that’s the support that we could use. And under the no, we can show that this test SUT converges to a pi square distribution and we could use that for testing.

In practice, you need to estimate these parameters.
We estimate them, we ensure that things still work with the estimated parameters and still so that you have can register \( \pi \) squared. And you can also do confidence and all this sector. Maybe I’ll just briefly mention that this also has the usual power that we expect that you can study power of this as a local alternative. And this gives us basically how that we would expect. And simulation also behaves very close to the oracle procedure that knows which neurons acting with other. What we’ve done here is that we’ve looked at increasing sample size or own length of the sequence from 200 to 2,000 and then we see that sort of type one error becomes pretty well controlled as time increases. The pink here is oracle. The blue is our procedure. The power also increases as the sample size increases. And also look at the coverage of the confidence involved. Both for the zeros and non zeros, the coverage also seems to be well behaved. This is simple setting of simulation but that looks like
It's not too far actually in application that we've also looked at. And in particular we've looked at some data paper that was published in 2018 in nature when they had looked at activation patterns of neurons and how they would change with and without laser. At the time this was like the largest, so they had multiple device that they had looked at, and this was the largest region that they had looked at had 25 neurons. The technology has improved quite a bit. Now there's a couple of hundred neurons that they could measure, but this was 25 neurons. And then what I'm showing you are the activation patterns without laser and with laser and not showing the edges that are common between the two networks. I'm just showing the edges are different between these networks. And we see that these betas, some of them are clearly different. In one condition the coefficient covers zero and the other conditions not cover. And that's why you're seeing these difference in networks. And that's similar to what they had observed based on basically correlation that as you activate...
there's more connectivity among these neurons.

Now in the actual experiments, and this is maybe the last 15 minutes or so by top,
in the actual experiments, they don't do just a simple
one shot experiment because they have to implant
this device.

This is data of a mouse.

They have to implant this device on mouse's brain.

And so what they do is that they actually,
and sort of now with that camera,
they just measure activities of neurons.

But once they do that, they actually run
a sequence of experiments.

It's never just a single experiment or two experiments.

What they do is that they, for example,
they show different images, the mouse
and they see the activation patterns of neurons
as the mouse processes different images.

And what they usually do is that sort they show an image
with one orientation and then they have a washout period.

They show an image with different orientation,
they have a washout period.

They show an image with a different orientation
and then they might use laser
in combination of these different images et cetera.

What they ended up doing is that they have many, many experiments. And what we expect is that the networks in these different experiments to be different from each other but maybe share some commonalities as well. We don’t expect completely different networks but we expect somewhat related networks. And over different time segments the network might change. In one segment it might be that and the next segment it might change to something different but maybe some parts of the network structure are like. What this does is that it sort of motivates us to think about join the estimate in these networks because each one of these time segments might not have enough observation to estimate accurately. And this goes back to the simulation results that I showed you, that in order to get to good control of type one error and good power, we need to have decent number of observations. And in each one of these time segments might not have enough observations. In order to make sure that we get high quality estimates.
and valid inference, we need to maybe join the estimations in order to get better quality estimates and influence.

That’s the idea of the second part of what I wanna talk about going beyond the single experiment and trying to do estimation and inference, and multiple experiments of similar.

And in fact in the case of this paper by and Franks they had, for every single mouse, they had 80 different experimental setups with laser and different durations with laser and different strengths.

It’s not a single experiment for each mouse. It’s 80 different experiments for each mouse.

And you would expect that many of these experiments are similar to each other and they might have different degrees of similarities with each other that might need to take into account.

Then the goal of the second part is do joint estimation of inference for settings where we have multiple experiments and not just a single experiment. To do this, we went back to basically that destination that we had previously what we had was the sparsity type penalty.
What we do is that sort of now we added a fusion type penalty. Now we combine the estimates in different experiments. And this is based on past work that I had done with the post but the main difference in this board is that now we wanna allow these estimates to be similar to each other based on a data-driven notion of similarity. We don’t know which experiments are more similar to each other. And we basically want the data to tell us which experiments should be more similar to each other, should be combined and not necessarily find that a priority person usually don’t have that information. These data-driven weights are critical here, and we drive these data-driven weights based on just simple correlations. We calculate simple correlations. The first step we look to see which one of these conditions, the correlations are more correlated with each other, more similar to each other based on these correlations. And we use these cost correlations to then define ways for which experiments should be more closely used.
with each other. And estimates on which experiments should be more closely used. And I leave that in terms of details but in this similar setting as what I had explained before in terms of experimental setup for this, I’m sorry, in terms of simulation setup, there are 50 neurons in network from three different experiments in this case of three different lengths, and we use different estimators. And what we see is that sort of when we do this fusion, we do better in terms of the number of two positives for any given number of estimated edges compared to separately estimating or compared to sort of other types of fusions that what one might consider. Now, estimation is somewhat easy. The main challenge was to come up with these data-driven weights. The main issue is that if you wanted to come up with valid infants in these settings, when we have many, many experiments, then we would have very low power if we’re adjusting, for example, from all comparison using FDR, FWER, false discovery rate or family-wise error rate,
we have $p^2 \times MS$.
And so we have a low power.
To deal with this setting, what we have done is that we’ve come up with a hierarchical testing procedure
that avoids testing all these $p^2 \times M$ coefficient.
And the idea is this, the idea is that if you have a sense of which conditions are more similar to each other,
we construct a very specific type of binary tree,
which basically always has a single node on the left side in this case.
And then we start on the top of that tree and test for each coefficient.
We first test Albany experiments. If you don’t reject, then you stop there.
If you reject then we test one, and two, three, and four separately.
If you reject one, then we’ve identified the non zero edge.
If you reject two, three, four, then we go down.
If you don’t reject two, three, four, we stop there.
This way we stop at the level that is appropriate based on data.
And this this ends up especially in sparse networks,
this ends up saving us a lot of tests.
and gives us significant improvement in power. And that’s shown in the simulation that you end up, if you don’t do this, your power decreases as the number of experiments increases. And in this case you’ve gone up to 50 experiments as I mentioned. The golden and facts paper has about 80. Whereas if you don’t do that and if your network sparse actually power, you see that by combining experiments, you actually gain power you’re incorporating more data. And this is more controlling the family-wise error rate. And both methods control the family-wise error rate. We haven’t developed anything for FDR. We haven’t developed theory for FDR but the method also seems to be controlling FDR in a very stringent way actually. But we just don’t have theory for FDR control ’cause that becomes more complicated. I’m going very fast because of time but I’ll pause for a minute. Any questions. Please. What do you think about stationary of the Hawkes process in the context?
Whether it’s the exogenous experimental forcing and like over what timescale did that happen in the stationary, the reasonable?

Yeah, that’s a really good question.

To be honest, I think these hard processes are most likely non stationary. The two mechanisms of non stationary that could happen. One, we try to account for it. I skipped over it but we tried to account for one aspect of it by allowing the baseline rate to be time varying. Basically we allow this this new is to be a function of time. Baseline rate for each neuron is varying over time. And the hope is that, that would capture some of the exogenous factors that might influence overall. It could also be that the data are changing over time. That sort of we haven’t done or it could in fact be that we have abrupt changes in patterns of either activation or the baseline over time, but sort all of a sudden something completely changes. We have piecewise stationary, not monotone sort of, not continuous, not stationary. We have piecewise.
We have experimental that’s happening, something happening and then all of a sudden something else is happening. This eventually would capture maybe plasticity in these neurons to neuroplasticity to some extent.

There’s actually one paper that has looked at piece stationary for these hard processes neuron.

It becomes a competition, very, very difficult problem, especially the person becomes very difficult problem.

But I think it’s a very good question. Aside from that one paper much else that has done.

I have a question regarding the segmentation ‘cause on the video you showed us, the image is generally very shaky.

In the computer vision perspective, it’s very hard to isolate which neuron actually fired and make sure that it’s that same neuron fires over time.

And also the second question is that the mouse factory, the model you’ve mentioned is like 20 neurons, but in the picture you show us there’s probably
thousands of neurons. How do you identify which 20 neurons to look at?

Very good questions. First of all, before they even get to segmentation, they need to do what is known as, and this is actually common in time series and sort of (indistinct). In registration.

What this means is that you first need to register the images so that they’re basically aligning correct. Then you can do segmentation. If you remember first five, but if you remember had a couple of dots before getting to segmentation. There are a couple of steps that need to happen and some background correction and sort of getting noise correctly and every-thing.

And then there’s registration.
And then after that you could do segmentation, identifying neurons. Now, the data that they showed you was a data from actually cats video that showed it’s different, this holding and banks data that they showed you here. This one had 25 neurons that they had. This is an older technology. It’s an older paper that they only had 25 neurons, that they had smaller regions that they were capturing. The newer technologies, they were capturing the larger region a couple hundred. I think the most I’ve seen was about a thousand or so neurons. I haven’t seen more than a thousand neurons. I think the most I’ve seen was about a thousand or so neurons. I haven’t seen more than a thousand neurons. Thank you. Okay, so I’m close to the end of my time. Maybe I’ll have the remaining minutes or so I’ll basically mention that sort of give by this saying we have joint estimation to the data from holding advance. And then we also see that something that is not surprising perhaps that the no laser condition, the net yield is more different than the two different magnitudes of laser, maybe 10, 20 sort of meters and so square.
You see that so least two are more similar other
than the no laser condition.
And I'm probably gonna stop here
and sort of leave a couple of minutes for questions,
additional questions, but I'll mention that
so the last part I didn't talk about was to see if we could
go beyond prediction.
Could we use this and mention that sort major causality
is not really causality prediction.
It could we go beyond prediction,
get a sense of which neurons are impacting other neurons.
And I'll briefly mention that sort of there are two issues
in general going beyond prediction causality.
We have a review paper that talks about this one,
issue is subsampling.
And that you don't have enough resolution.
And the other issue is where you might have
limited processes that make it difficult
to answer all the questions.
Fortunately the issue of self sampling,
which is a difficult issue in general is not present,
but is not very prominent thinking these classroom
and imaging data
because you have continuous time videos.
And subsampling should not be a big deal in this case. However, we observe a tiny faction of the connection of the brain. The question is, can we somehow account for all the other neurons that we don’t see? The last part of this work is about that. And I’ll sort of jump to the end because I’ll put a reference to that work. That one is published in a paper that sort of looks at whether we could go beyond prediction, whether they actually identify causal links particularly neurons. And I think I’m gonna stop here and thank you guys. 

Biologically, what is a network connection here? Because they’re not, I’m assuming they’re not growing synapses or not based on the laser.

(group chattering)