# Using Neighbourhoods to Classify Functional Brain Data

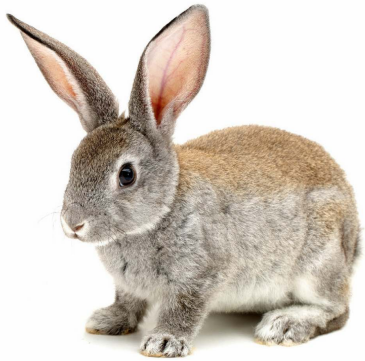
Jason P. Smith

Nottingham Trent University

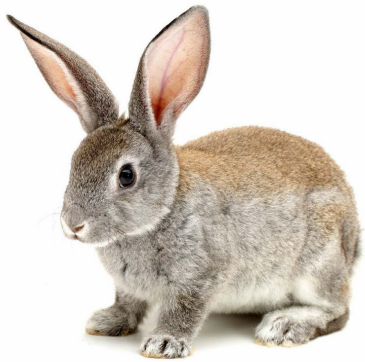
24th May 2022

# The Problem

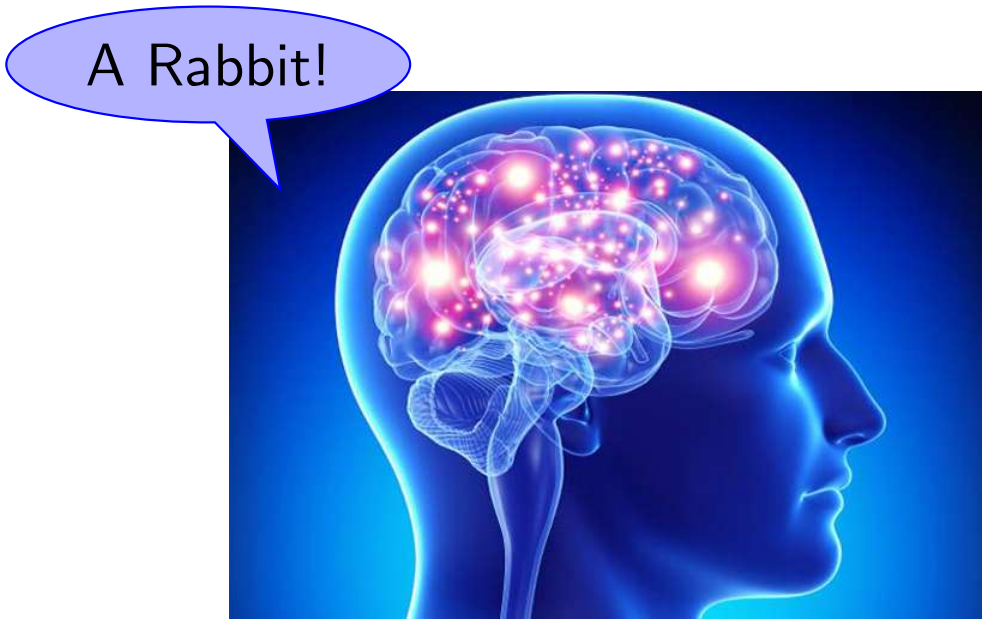
# The Problem

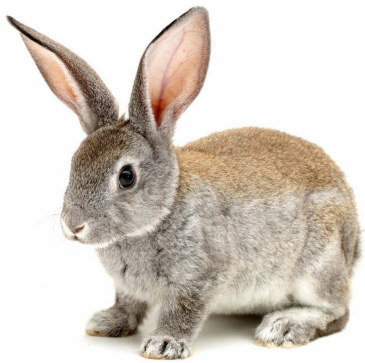


# The Problem

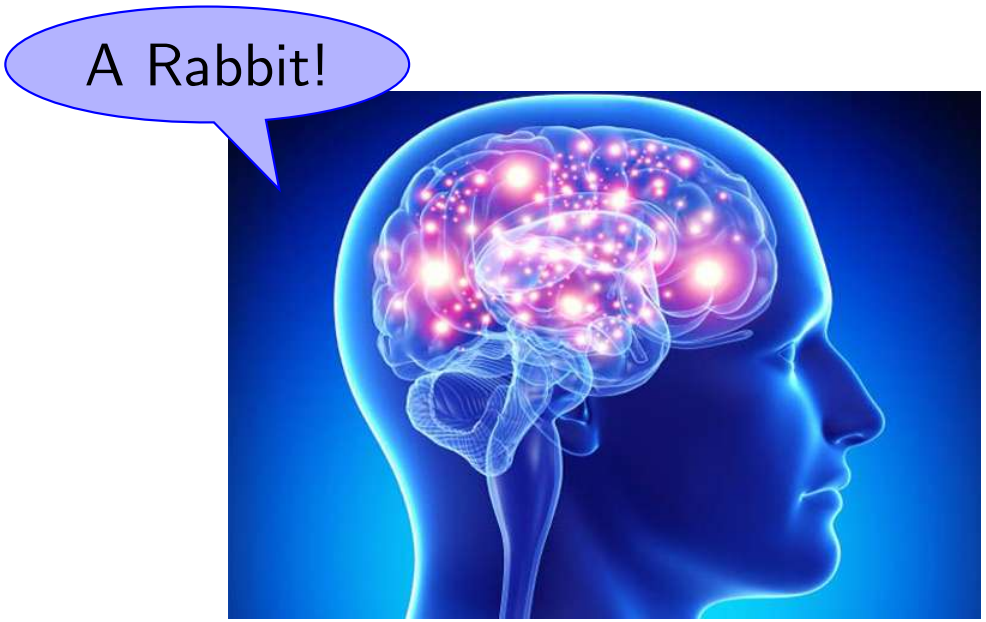


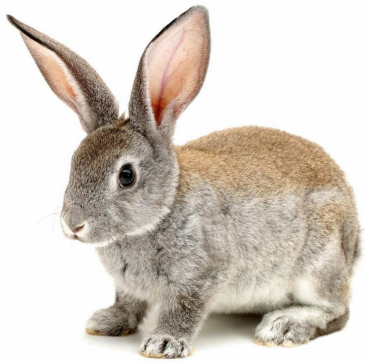
# The Problem



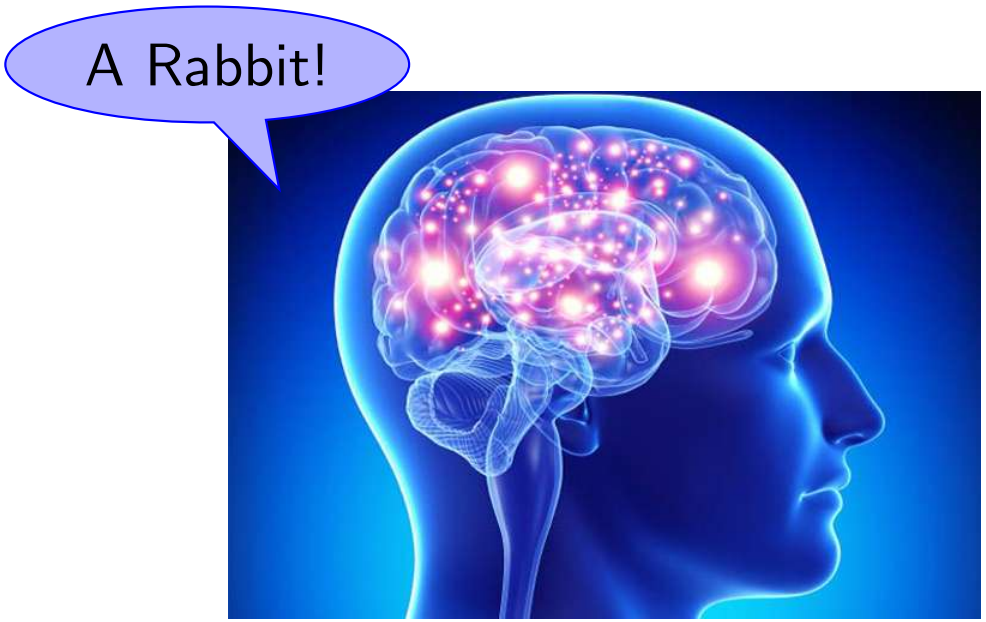
# The Problem

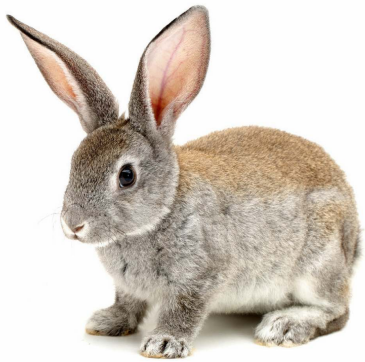


How does the brain determine it has seen a rabbit from this sequence of neurons firing?

# The Problem

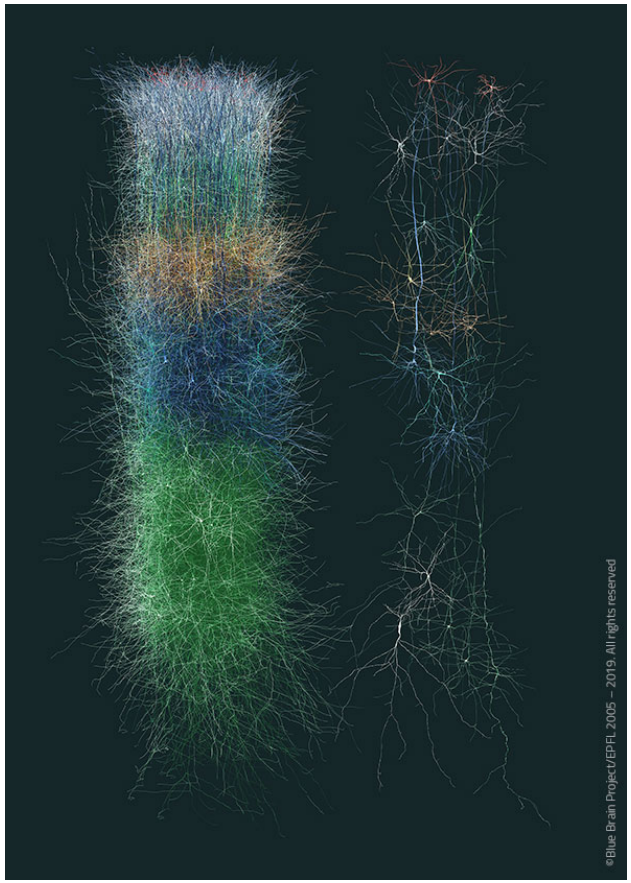


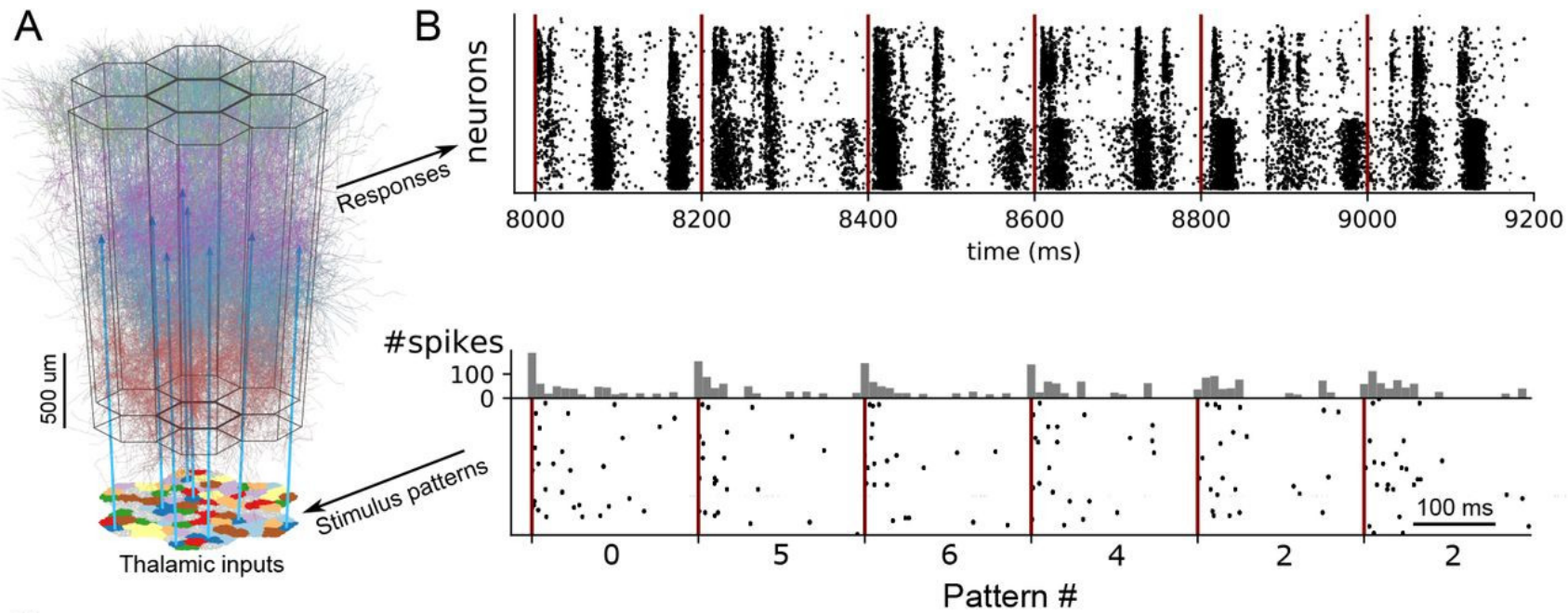
How does the brain determine it has seen a rabbit from this sequence of neurons firing? Can we make a computer do it?

# The Blue Brain Microcircuit


- Rat Somatosensory Cortex
- $\sim$ 30 000 Neurons
- $\sim$ 8 000 000 Connections
- 6 Layers
- 55 Neuronal Types
- 11 Electrical Types
- Functional Model



# The Stimuli


A
B
neurons
Responses
8000
8200
8400
8600
8800
9000
9200
time (ms)
#spikes
100
0
500 um
Stimulus patterns
100 ms
Thalamic inputs
0
5
6
4
2
2
Pattern #

# The Plan

8 distinct stimuli
550 repetitions of each

→ 4400 spike trains →

Classifier:
Maps spike trains X to stimulus Y

For example:

Stimuli 0:
Thalamic inputs 1 — 40
Stimuli 1:
Thalamic inputs 41 — 80
Stimuli 2:
Thalamic inputs 81 — 120

etc

→

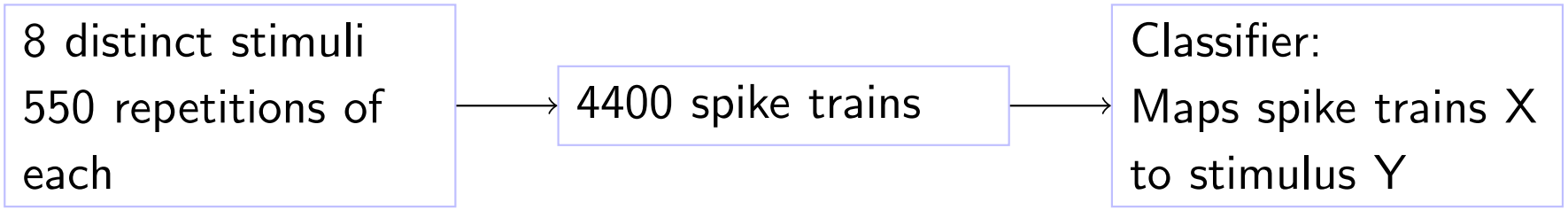
(time, neuron)
(0.0, 1003)
(5.5, 21403)
(12.1, 13)
(15.9, 31002)

etc

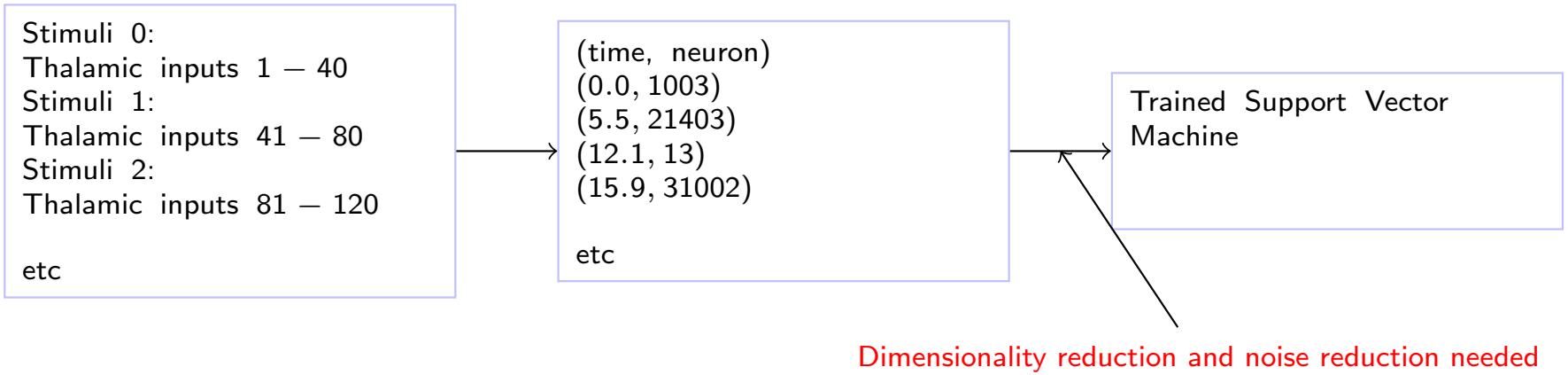
→

Trained Support Vector Machine

# The Plan

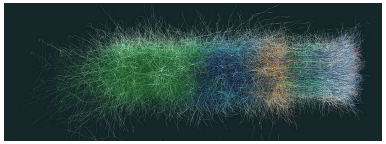



For example:



# The Approach

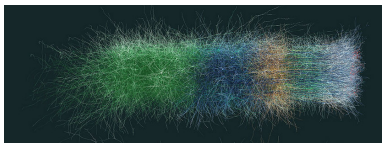
## The Blue Brain Circuit
& Spike Activity

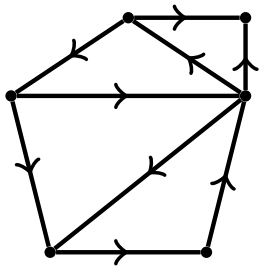


# The Approach

The Blue Brain Circuit $\longrightarrow$ A Directed Graph

& Spike Activity

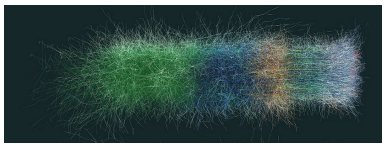


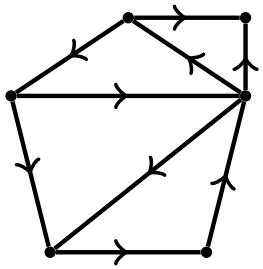


# The Approach

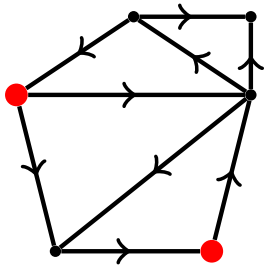
The Blue Brain Circuit ⟶ A Directed Graph → Select Champion Vertices

& Spike Activity



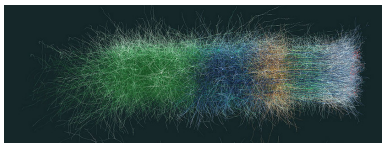


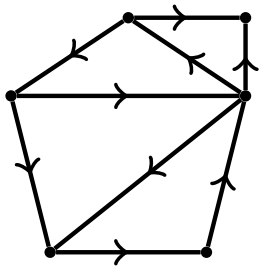
Using some "Selection Parameter"

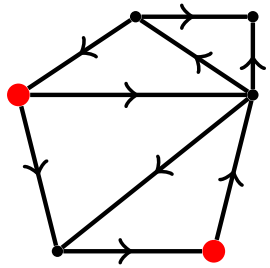


# The Approach

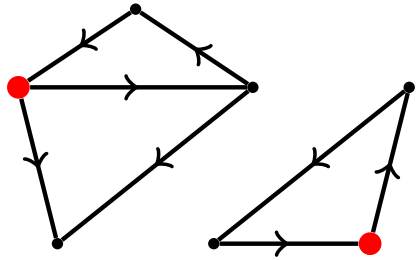
The Blue Brain Circuit & Spike Activity $\longrightarrow$ A Directed Graph $\rightarrow$ Select Champion Vertices Using some "Selection Parameter"





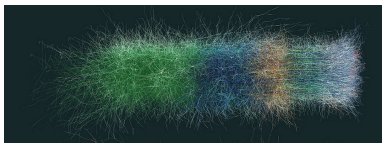


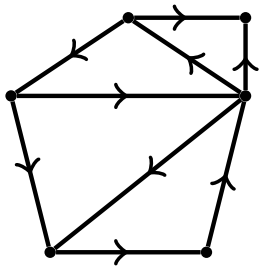
Take closed neighbourhood of each champion

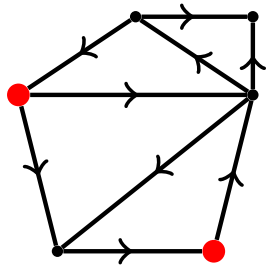


# The Approach

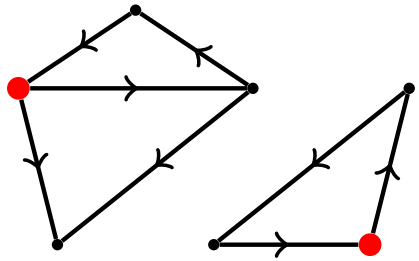
The Blue Brain Circuit & Spike Activity ⟶ A Directed Graph → Select Champion Vertices Using some "Selection Parameter"





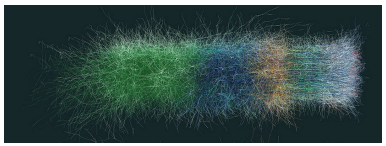


Take closed neighbourhood of each champion ← Apply "featurisation function" to spike activity on each neighbourhood

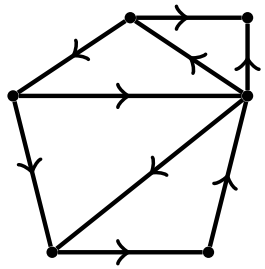


# The Approach

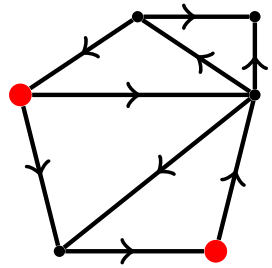
The Blue Brain Circuit & Spike Activity



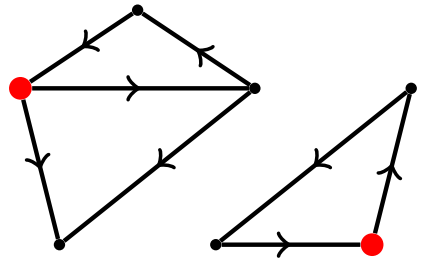
$\longrightarrow$ A Directed Graph



$\rightarrow$ Select Champion Vertices Using some "Selection Parameter"



Take closed neighbourhood of each champion



$\leftarrow$ Apply "featurisation function" to spike activity on each neighbourhood

$\longleftarrow$ Feed feature vectors into support vector machine (SVM)

# The Champions

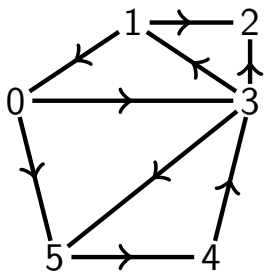
For every vertex ($\sim 30k$) we computed the following on their closed neighbourhood:

| | |
|---|---|
| **size** | Number of vertices in neighbourhood |
| **ec:** | Euler characteristic of directed flag complex |
| **nbc:** | Normalised Betti coefficient of directed flag complex |
| **dck** | k-th Density Coefficient ($\frac{\#k+1 \text{ cliques}}{\#k \text{ cliques}}$ normalised by dimension) |
| **fcc** | Clustering Coefficient (ratio of triangles to possible triangles) |
| **tcc** | Transitive clustering coefficient (directed version of Clustering Coefficient) |
| **rc** | Bidirectional edges |
| **asg** | Adjacency spectral gap and radius |
| ***lsg** | Laplacian spectral gap and radius |
| **tpsg** | Transition probability spectral gap and radius |

We then selected the 50 highest values neighbourhoods.

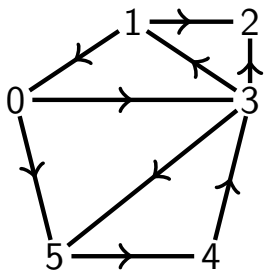
# The Champions (Example)

| vertex | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| size | 4 | 4 | 3 | 6 | 3 | 4 |
| ec | 0 | 0 | 1 | -1 | 0 | 0 |
| nbc | 0.65 | 0.65 | 0.333 | 0.611 | 1.00 | 0.65 |
| tcc | 0.25 | 0.25 | 0.5 | 0.143 | 0.0 | 0.25 |
| dc2 | 0.111 | 0.111 | 0.333 | 0.067 | 0 | 0.111 |
| fcc | 0.333 | 0.333 | 0.5 | 0.2 | 0.5 | 0.333 |
| rc | 0 | 0 | 0 | 0 | 0 | 0 |
| asg | 1.0 | 1.0 | 0 | 0.123 | 0 | 1.0 |
| blsg | 0.424 | 0.424 | 0 | 0.507 | 0 | 0.732 |
| tpsg | 1.0 | 1.0 | 0 | 0.097 | 0 | 0.63 |



# The Champions (Example)

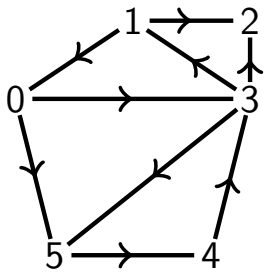
| vertex | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| size | 4 | 4 | 3 | 6 | 3 | 4 |
| ec | 0 | 0 | 1 | -1 | 0 | 0 |
| nbc | 0.65 | 0.65 | 0.333 | 0.611 | 1.00 | 0.65 |
| tcc | 0.25 | 0.25 | 0.5 | 0.143 | 0.0 | 0.25 |
| dc2 | 0.111 | 0.111 | 0.333 | 0.067 | 0 | 0.111 |
| fcc | 0.333 | 0.333 | 0.5 | 0.2 | 0.5 | 0.333 |
| rc | 0 | 0 | 0 | 0 | 0 | 0 |
| asg | 1.0 | 1.0 | 0 | 0.123 | 0 | 1.0 |
| blsg | 0.424 | 0.424 | 0 | 0.507 | 0 | 0.732 |
| tpsg | 1.0 | 1.0 | 0 | 0.097 | 0 | 0.63 |




Parameter=size, $k = 1$ champions

# The Champions (Example)

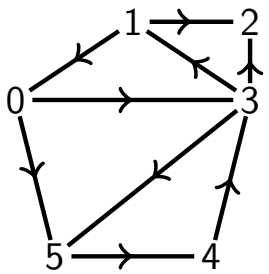
| vertex | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| size | 4 | 4 | 3 | 6 | 3 | 4 |
| ec | 0 | 0 | 1 | -1 | 0 | 0 |
| nbc | 0.65 | 0.65 | 0.333 | 0.611 | 1.00 | 0.65 |
| tcc | 0.25 | 0.25 | 0.5 | 0.143 | 0.0 | 0.25 |
| dc2 | 0.111 | 0.111 | 0.333 | 0.067 | 0 | 0.111 |
| fcc | 0.333 | 0.333 | 0.5 | 0.2 | 0.5 | 0.333 |
| rc | 0 | 0 | 0 | 0 | 0 | 0 |
| asg | 1.0 | 1.0 | 0 | 0.123 | 0 | 1.0 |
| blsg | 0.424 | 0.424 | 0 | 0.507 | 0 | 0.732 |
| tpsg | 1.0 | 1.0 | 0 | 0.097 | 0 | 0.63 |




Parameter=fcc, $k = 2$ champions

# The Champions (Example)

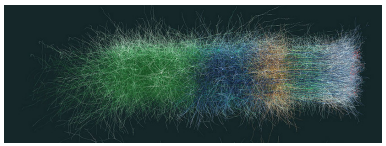
| vertex | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| size | 4 | 4 | 3 | 6 | 3 | 4 |
| ec | 0 | 0 | 1 | -1 | 0 | 0 |
| nbc | 0.65 | 0.65 | 0.333 | 0.611 | 1.00 | 0.65 |
| tcc | 0.25 | 0.25 | 0.5 | 0.143 | 0.0 | 0.25 |
| dc2 | 0.111 | 0.111 | 0.333 | 0.067 | 0 | 0.111 |
| fcc | 0.333 | 0.333 | 0.5 | 0.2 | 0.5 | 0.333 |
| rc | 0 | 0 | 0 | 0 | 0 | 0 |
| asg | 1.0 | 1.0 | 0 | 0.123 | 0 | 1.0 |
| blsg | 0.424 | 0.424 | 0 | 0.507 | 0 | 0.732 |
| tpsg | 1.0 | 1.0 | 0 | 0.097 | 0 | 0.63 |

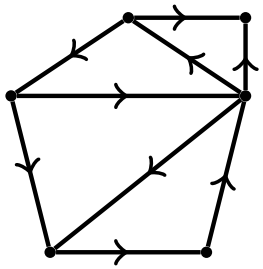


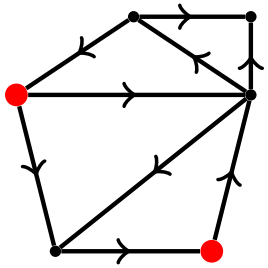

Parameter=nbc (smallest), $k = 2$ champions

# The Approach

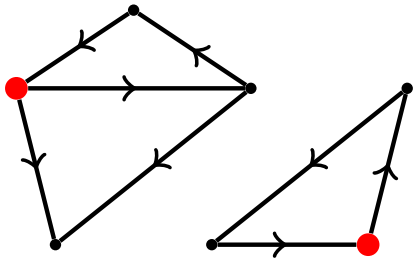
The Blue Brain Circuit & Spike Activity ⟶ A Directed Graph → Select Champion Vertices Using some "Selection Parameter"







Take closed neighbourhood of each champion

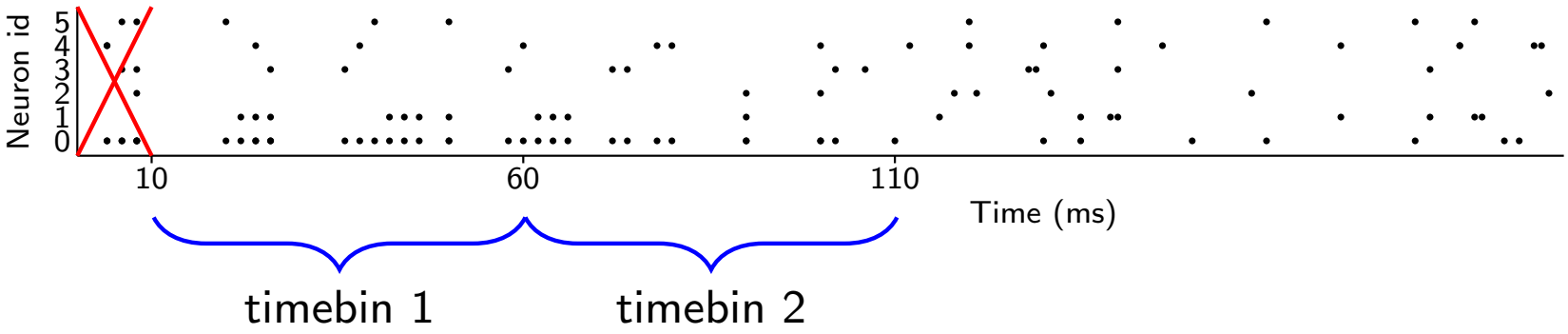


Apply "featurisation function" to spike activity on each neighbourhood

Feed feature vectors into support vector machine (SVM)

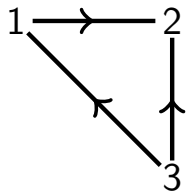
# The Featurisation Function

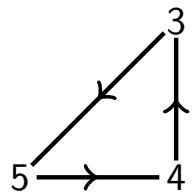
For each timebin and each neighbourhood selected, take subgraph induced by spiking neurons and compute parameter.



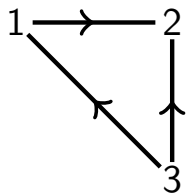

Selection Parameter = fcc, $k = 2$, champions = $\{2,4\}$

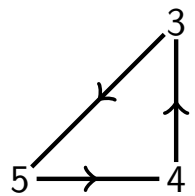
timebin 1



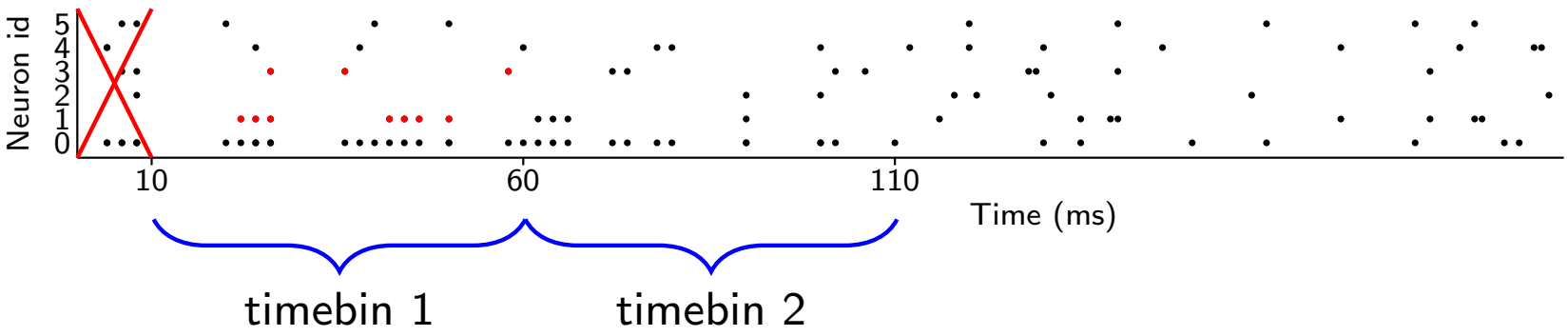

timebin 2



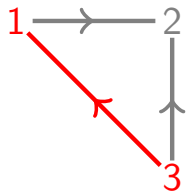
# The Featurisation Function

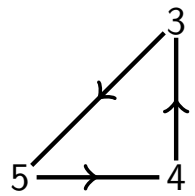
For each timebin and each neighbourhood selected, take subgraph induced by spiking neurons and compute parameter.



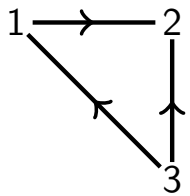

Selection Parameter = fcc, $k = 2$, champions = $\{2,4\}$

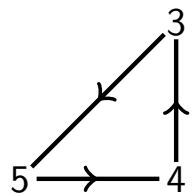
timebin 1



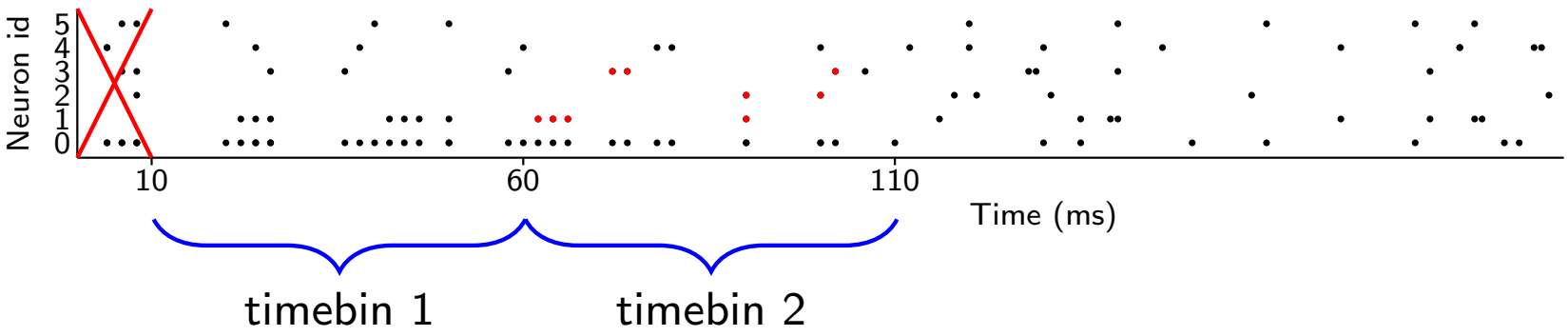

timebin 2



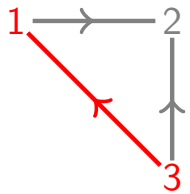
# The Featurisation Function

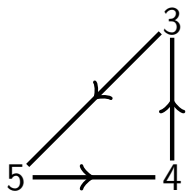
For each timebin and each neighbourhood selected, take subgraph induced by spiking neurons and compute parameter.



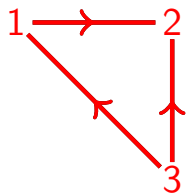

Selection Parameter = fcc, $k = 2$, champions = $\{2,4\}$

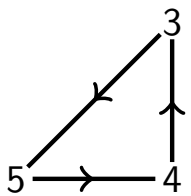
timebin 1



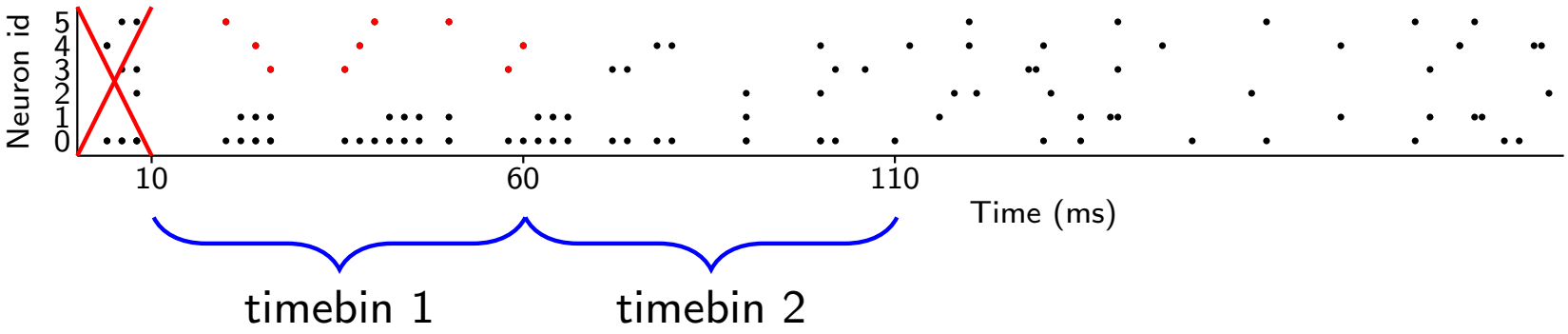

timebin 2



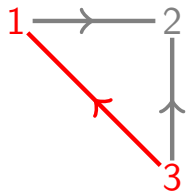
# The Featurisation Function

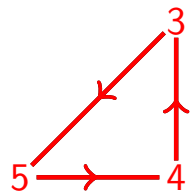
For each timebin and each neighbourhood selected, take subgraph induced by spiking neurons and compute parameter.



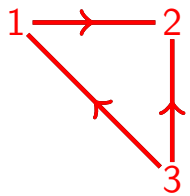

Selection Parameter = fcc, $k = 2$, champions = {2,4}

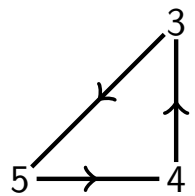
timebin 1



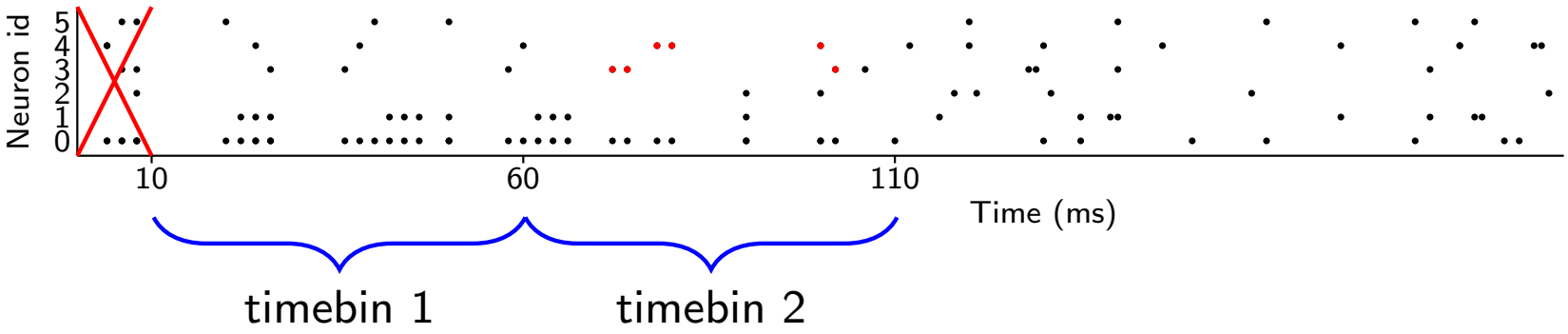

timebin 2



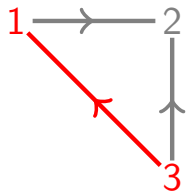
# The Featurisation Function

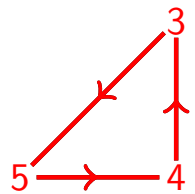
For each timebin and each neighbourhood selected, take subgraph induced by spiking neurons and compute parameter.



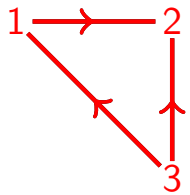

Selection Parameter = fcc, $k = 2$, champions = {2,4}

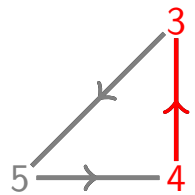
timebin 1



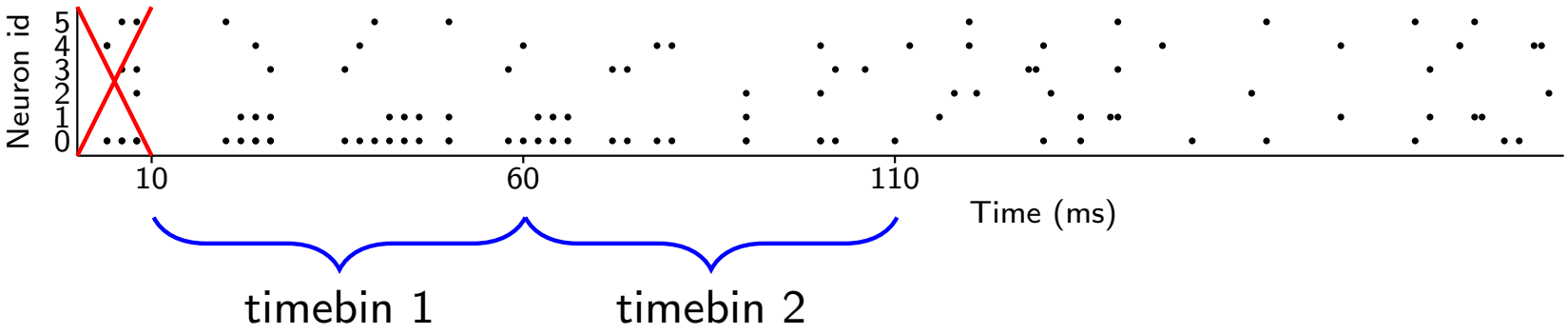

timebin 2



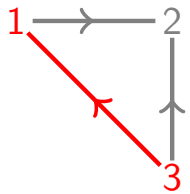
# The Featurisation Function

For each timebin and each neighbourhood selected, take subgraph induced by spiking neurons and compute parameter.

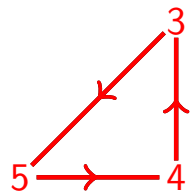



Selection Parameter = fcc, $k = 2$, champions = {2,4}

timebin 1

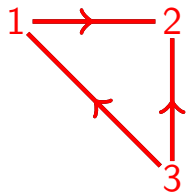



2

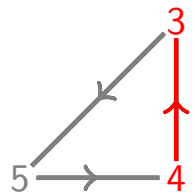



3

timebin 2




3

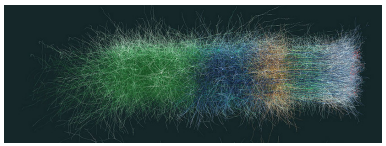



2

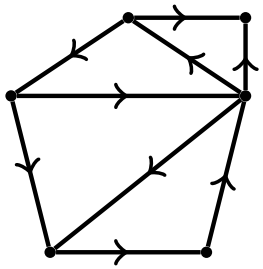
Feature Parameter = size, Feature vector = [2,3,3,2]

# The Approach

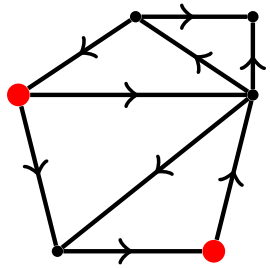
The Blue Brain Circuit & Spike Activity



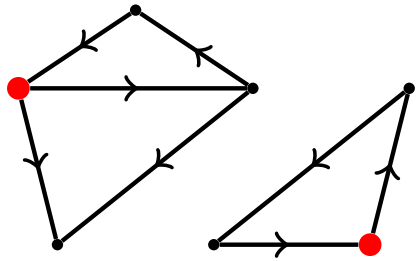
⟶ A Directed Graph



→ Select Champion Vertices
Using some "Selection Parameter"



Take closed neighbourhood of each champion



← Apply "featurisation function" to spike activity on each neighbourhood

⟵ Feed feature vectors into support vector machine (SVM)

# Some Machine Learning

For each timebin (2) and each neighbourhood selected (50) compute featurisation value and place all into feature vector of length 100.

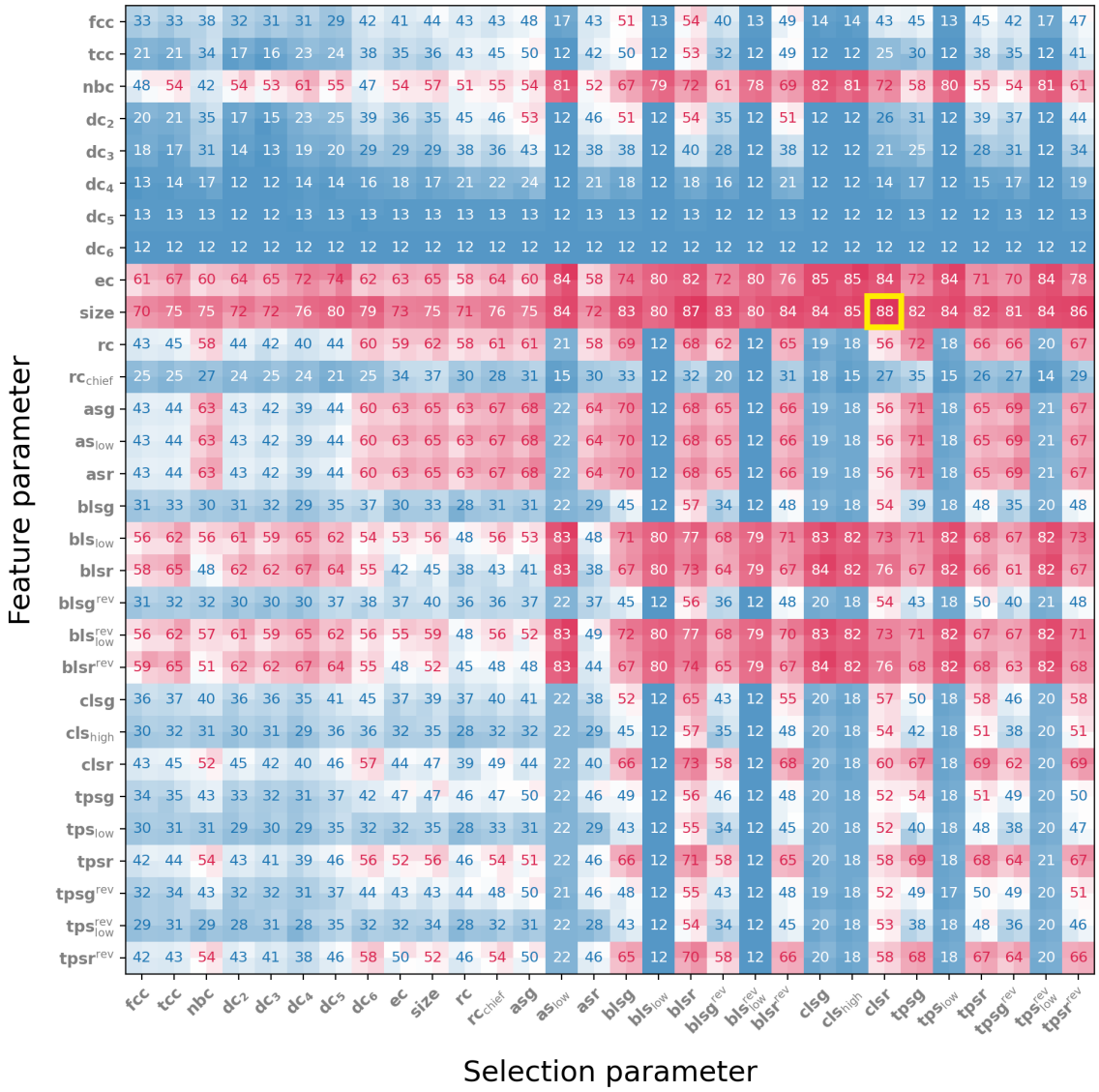
4400 feature vectors, 1 for each input (550) of each stimulus (8)

Feed 60% of vectors into SVM to train

Test classsifying ablity of SVM on remaining 40% of vectors.

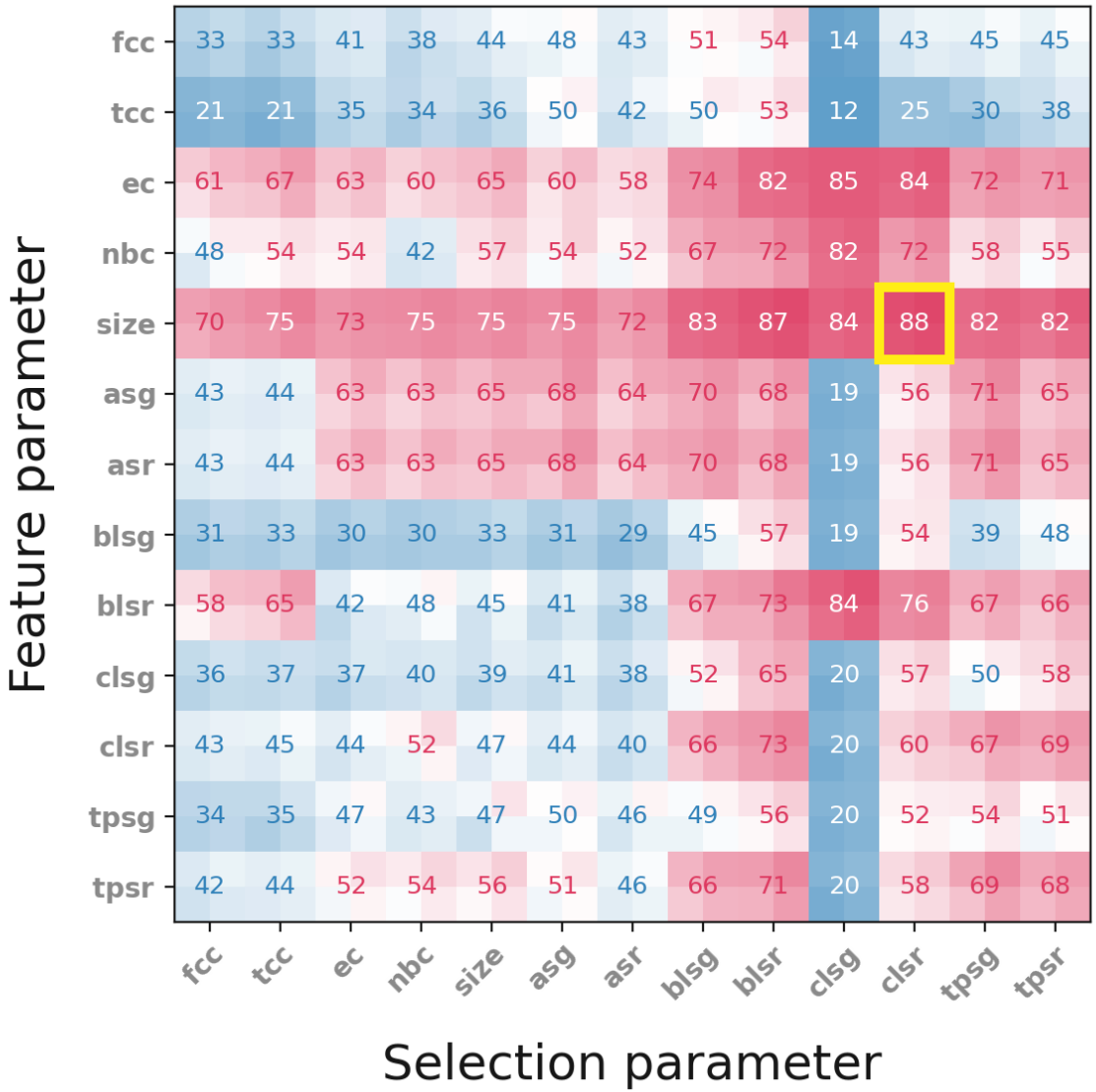
# The Results

## Top Champions Classification Accuracy



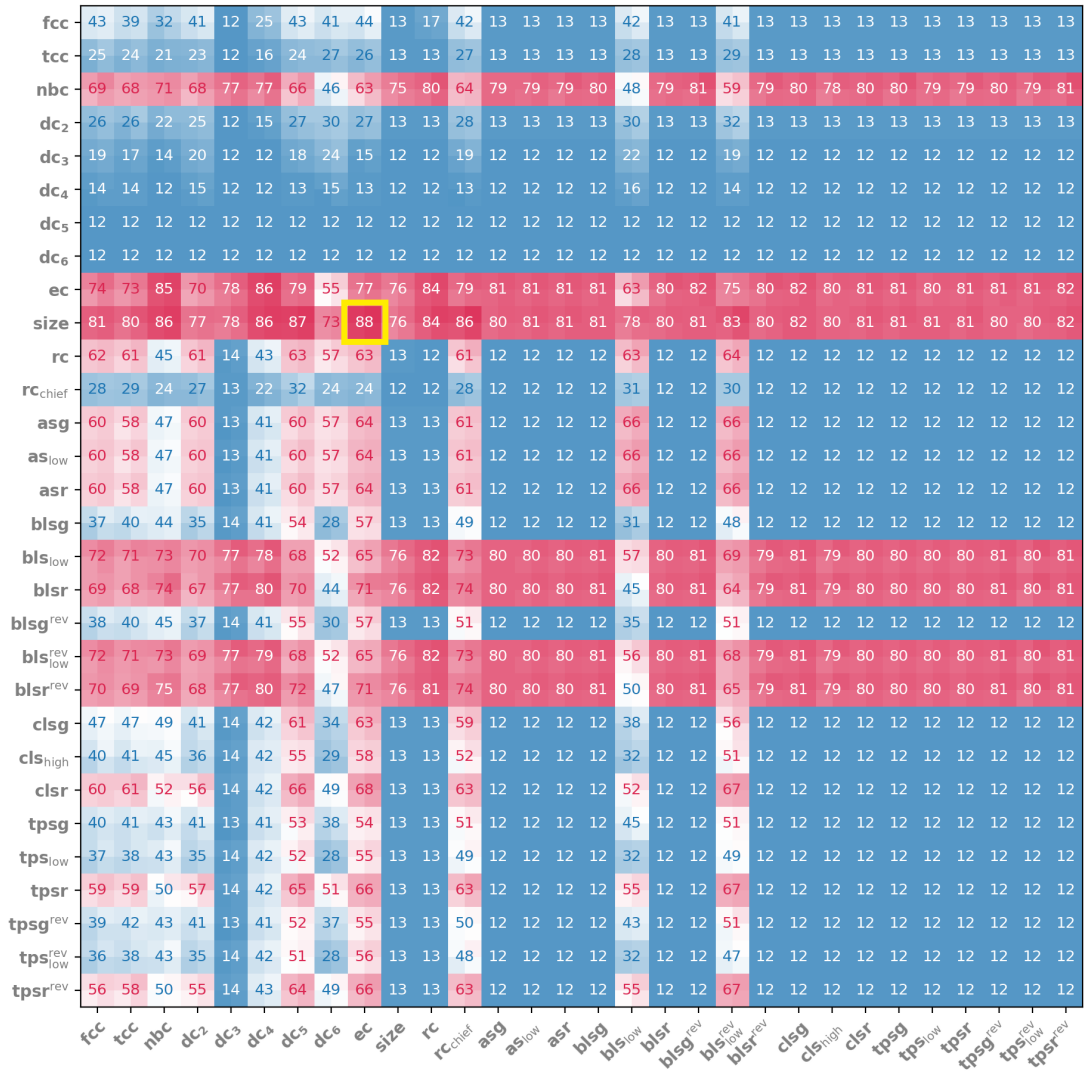
# The Results

## Top Champions Classification Accuracy



# The Results

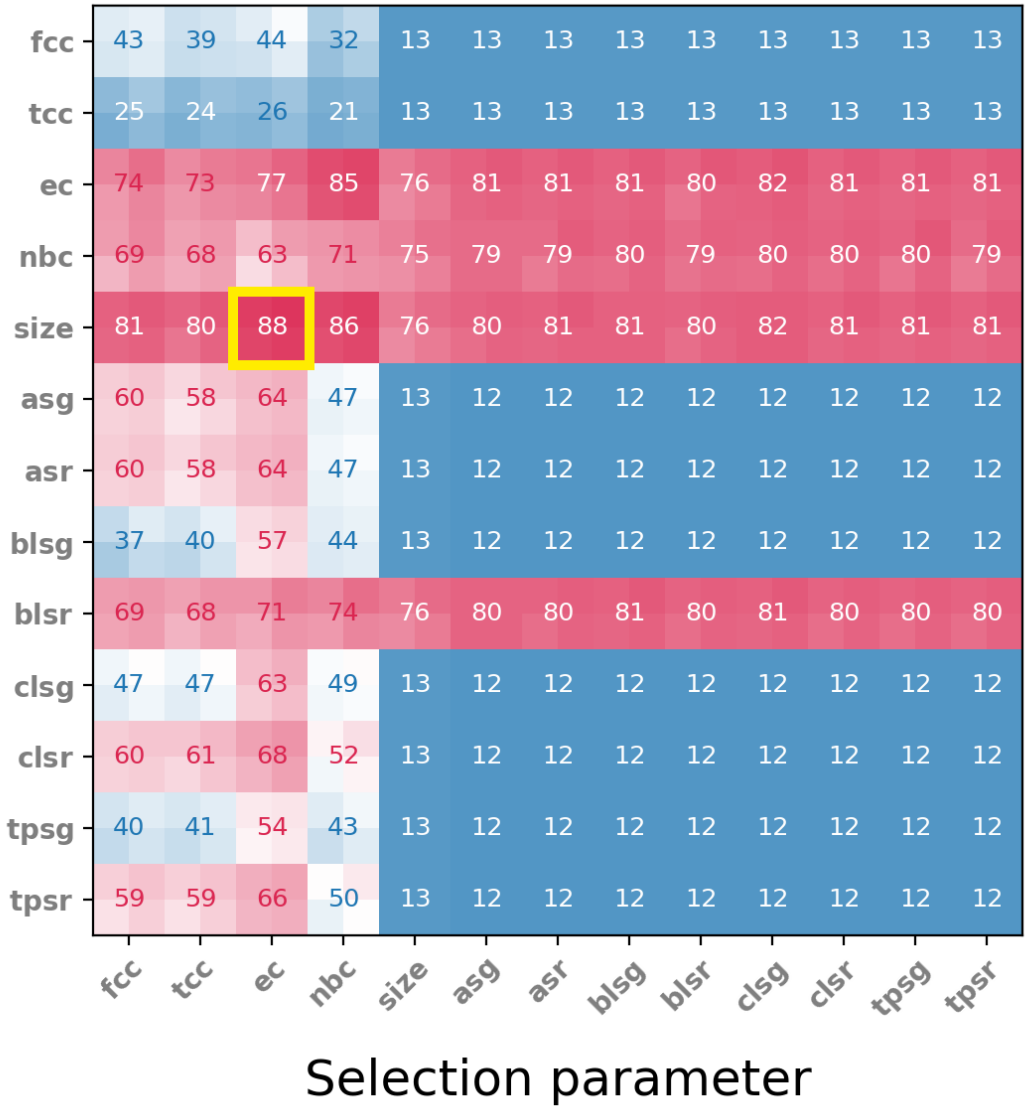
## Bottom Champions Classification Accuracy




Selection parameter

# The Results

## Bottom Champions Classification Accuracy




Selection parameter

# The Future

- Apply to other data sets
- Use other graph metrics
- Improve accuracy with different ML techniques
- Certain parameters better for certain situations?
- Why does selecting by these parameters work?

# Thanks for Listening!

Many Thanks to the Blue Brain Project

And my collaborators:
Ran Levi (Aberdeen), Jānis Lazovskis (Riga), Henri Riihimäki (KTH),
Dejan Govc (Ljubljana), and Pedro Conceição (Aberdeen)

github.com/JasonPSmith/tridy

An application of neighbourhoods in digraphs to the classification of binary dynamics, *Network Neuroscience (2022)*

Topology of synaptic connectivity constrains neuronal stimulus representation, predicting two complementary coding strategies, *PLOS ONE (2022)*