Table of content

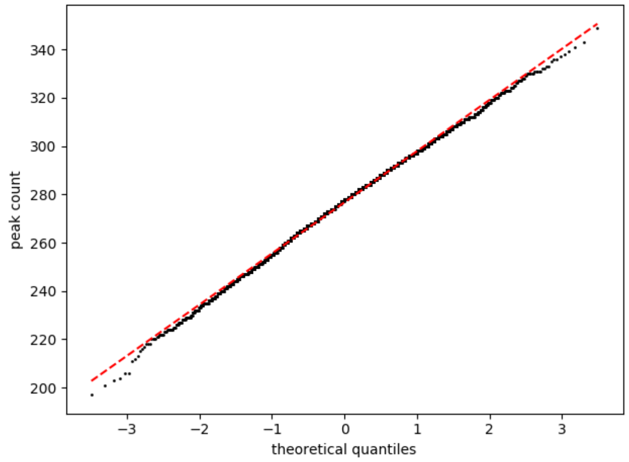
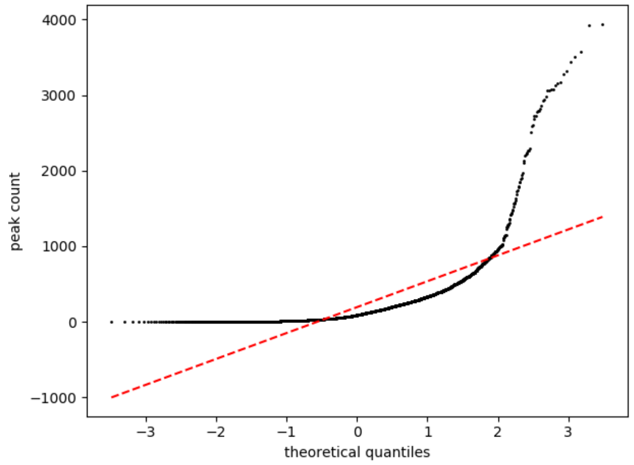
- **Neuron spike analysis (DIV25_spks)**
- **Dynamics (case 4: directed weighted random network)**
- **Dynamics (DIV25_PREmethod, synaptic coupling function)**
- **Dynamics (DIV25_PREmethod, diffusive coupling function)**
- **Spike analysis (peak histogram, raster plot)**
- **Spike analysis (peak distribution, peak vs strength plot)**
- **Autocorrelation analysis**
- **Spike analysis (Gaussian reference network)**
- **Reference network analysis**

<u>**Neuron spike analysis (DIV25_spks)**</u>

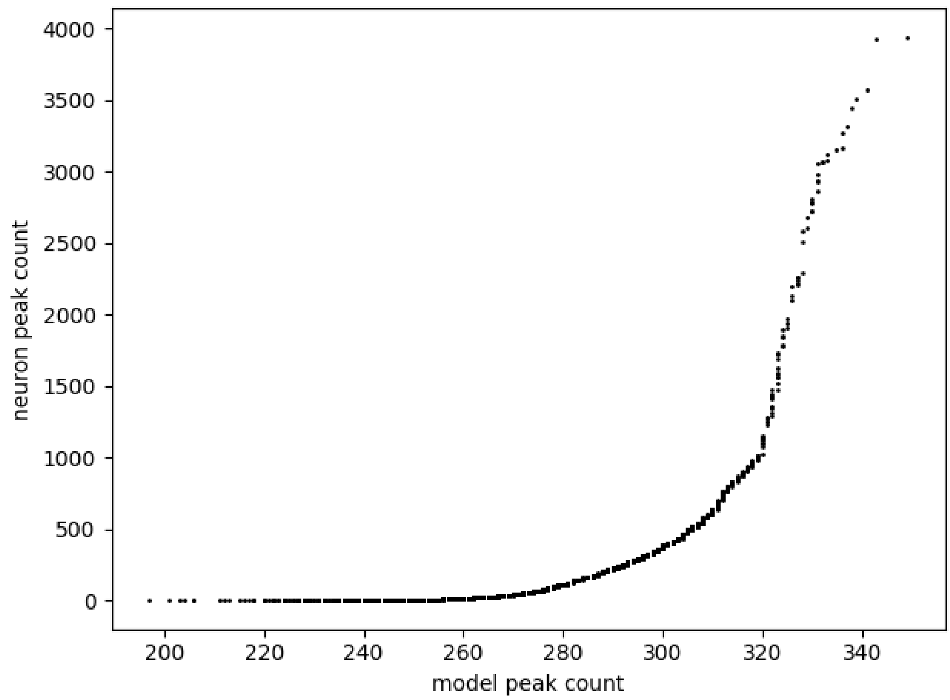
- **Analyze how neuron spikes distribute**
- **Here all peak count data are experimental**

- **(Continuing from last time) With-noise case: params** $(r_0, g_{ij} \times \text{multiplier}, \sigma_i) = (r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5)$
- **Question to address**: how peaks distribute?

- **How peak data contrast with normal quantiles**
  Quantile-quantile plot (QQ plot): plot percentiles of data against percentiles of another set of data (maybe from a specified distribution)
  Model peak data almost perfectly align with normal quantiles
  Neuron peak data have a very heavy tail compared with normal quantiles

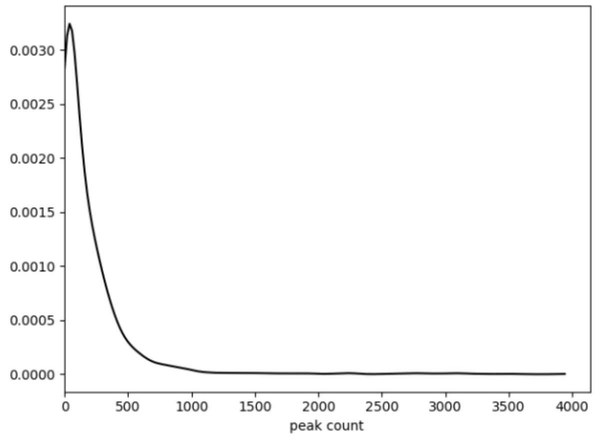
Model peak data	Neuron peak data
	

- **How peak data contrast with each other**
  Expected as model peak data are near normal

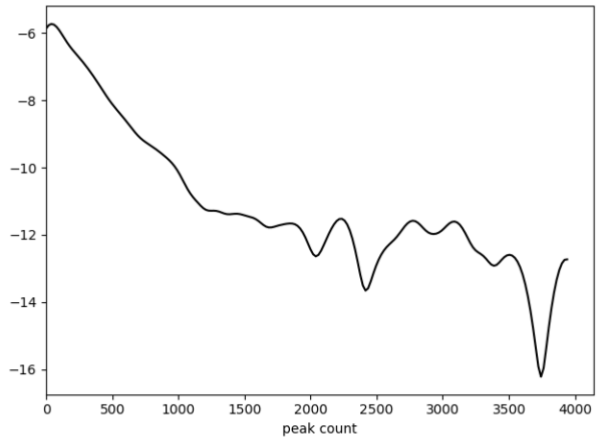


- **More analysis for neuron peak data**
  Distributions are smoothed

Prob distribution vs peak count

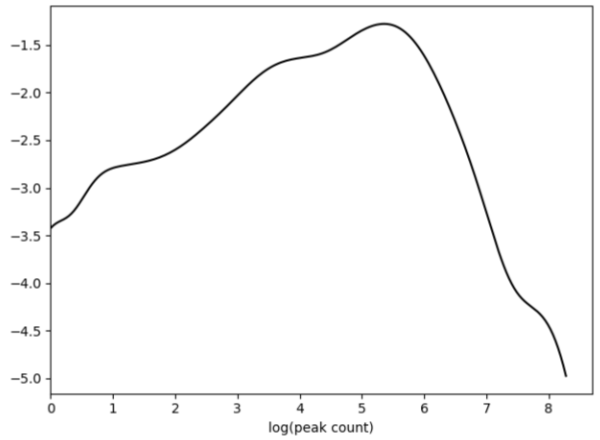


Log-prob distribution vs peak count



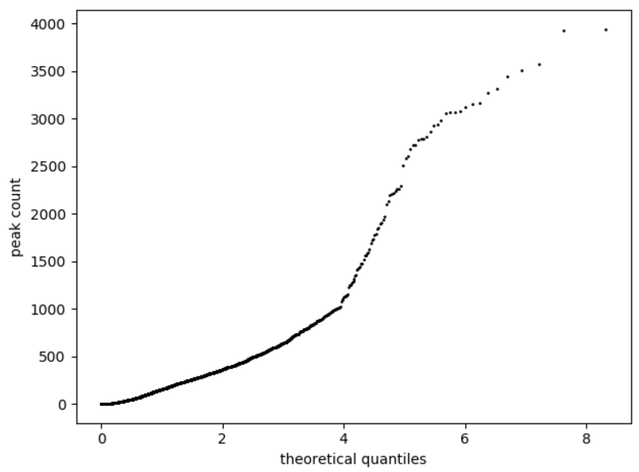
- A check for exponential distribution
- **Good fit for small peak counts** but large peak counts are heavier than tail of exponential distribution

Log-prob distribution vs log-peak count



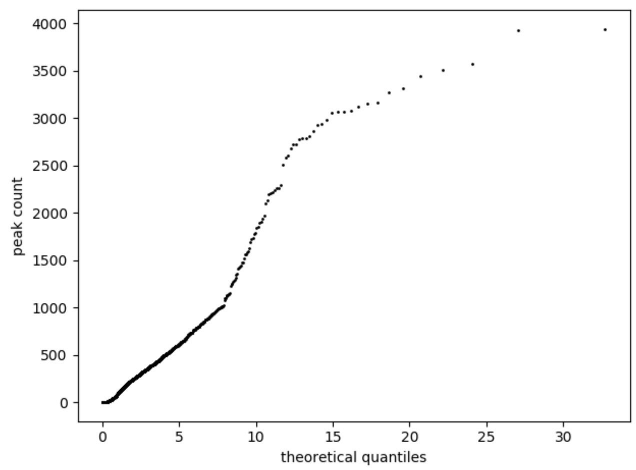
- A check for Pareto (power-law) distribution
- Pareto: $\boldsymbol{b/x^{b+1}}$
- Log-log plot has slope $-(b+1)$
- **Good fit for large peak counts** ($e^6 \sim 400$ to $e^8 \sim 3000$) but small peak counts are less prominent than what Pareto distribution "predicts"
- Rough: $\boldsymbol{b \sim 0.75}$

QQ plot (vs exponential distribution)



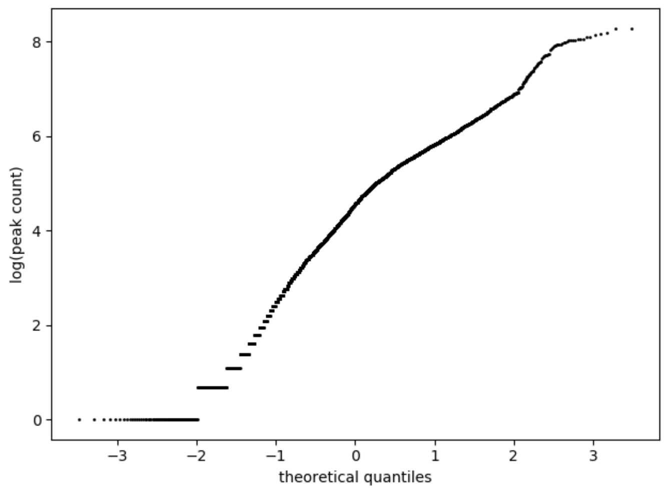
- Exponential distribution may be a good fit for small peak counts
- But QQ plot has some initial curvature

QQ plot (vs lognormal distribution)



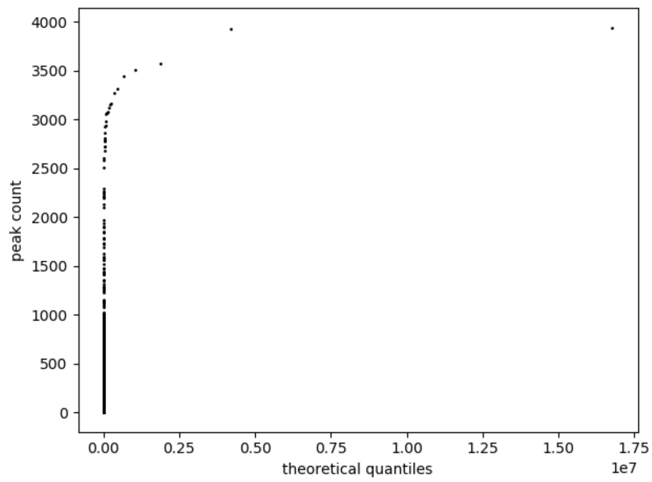
- Lognormal distribution may be a good fit for small peak counts
- QQ plot is almost straight for small peak counts

QQ plot (log peak count vs normal distribution)

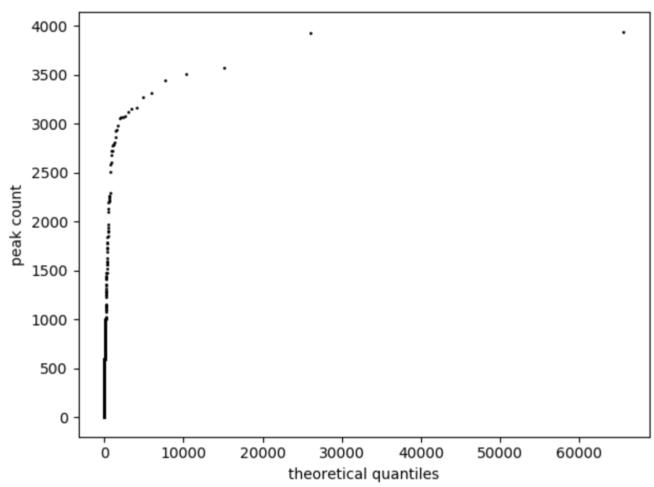


- To further check lognormal distribution
- Quite good fit for small peak counts

QQ plot (vs Pareto distribution, $b = 0.5$)

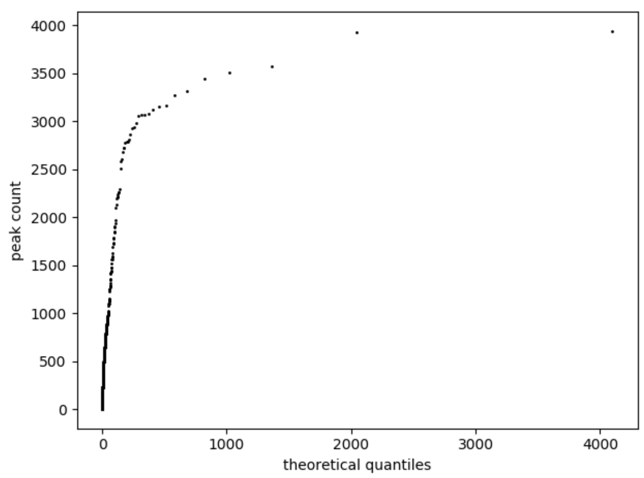


QQ plot (vs Pareto distribution, $b = 0.75$)



- Good fit for large peak counts (400-3000), as what previous result suggests
- But outliers (>3000) are not as heavy

QQ plot (vs Pareto distribution, $b = 1$)



- **Guess**

  Peak counts consist of 3 regimes:

  (1) **Less than 400**: lognormal distribution

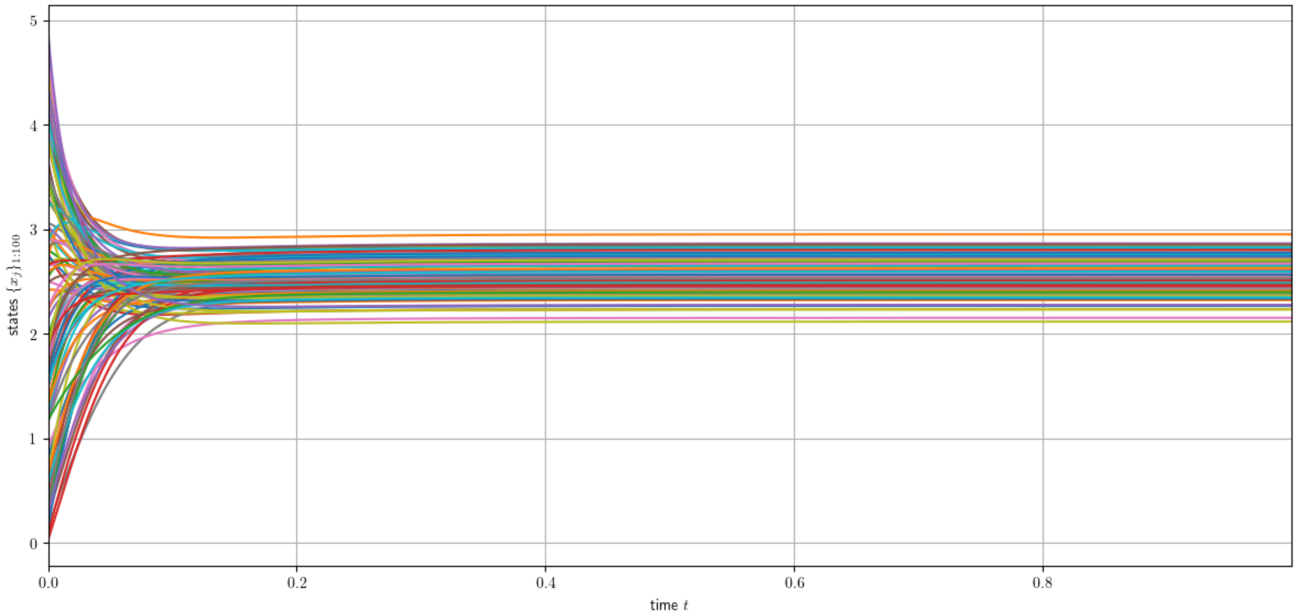
  (2) **Between 400 and 3000**: Pareto $b \sim 0.75$

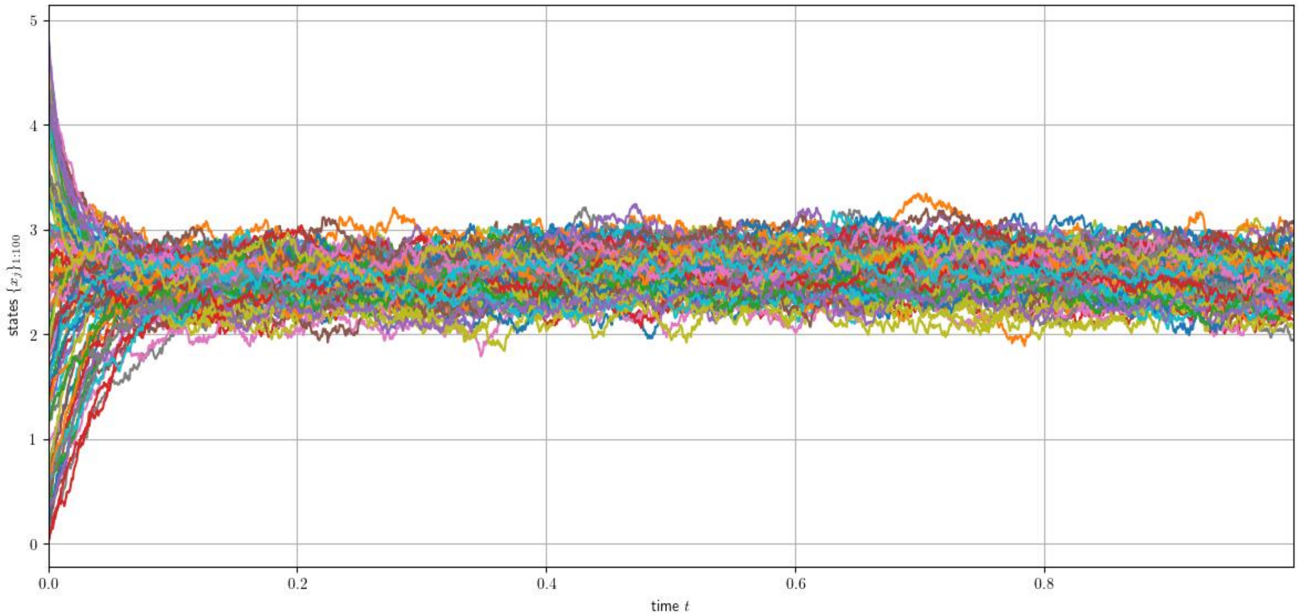
  (3) **Larger than 3000** (only 14 nodes fall in this extreme outlier range): another Pareto with larger b (so that tail is thinner), but may be meaningless to fit distribution $\because$ small number of extreme outliers

Dynamics (case 4: directed weighted random network)

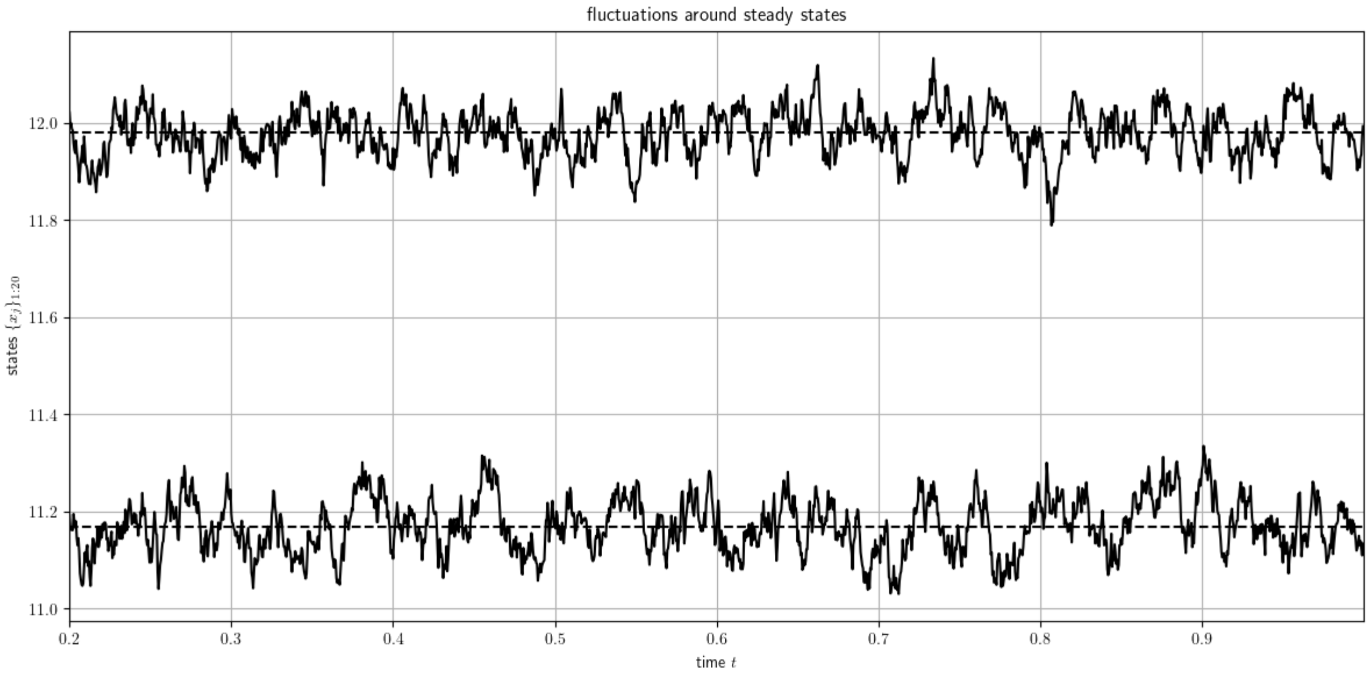
- **Examine linear in Q vs M plot**

- **Parameters**
  (a) Random directed weighted network
  (b) Size = 100, connection probability = 0.2, weights $\sim N(10,2)$, $\sigma_i = 1$
  (c) Intrinsic dynamics coefficient $r_i = 10$
  (d) Synaptic coupling function $(\beta_1, \beta_2, y_0) = (2, 0.5, 4)$
  (e) Initial conditions $\mathrm{uniform}[0,5]$
  (f) Step size $\delta t = 5 \times 10^{-4}$, time steps $N_{\mathrm{data}} = 2 \times 10^6$
  (g) Computation time: (run dynamics) 7 min for $N_{\mathrm{data}} = 2 \times 10^6$
- **Time series: noise-free & with-noise**
  $N_{\mathrm{data}} = 2 \times 10^3$
  Transient behavior before $t = 0.2$
  Steady values btw. 2 and 3

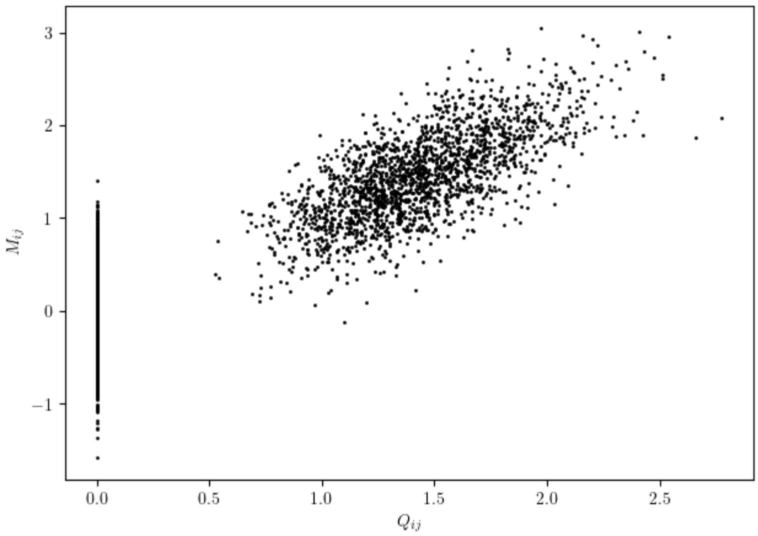




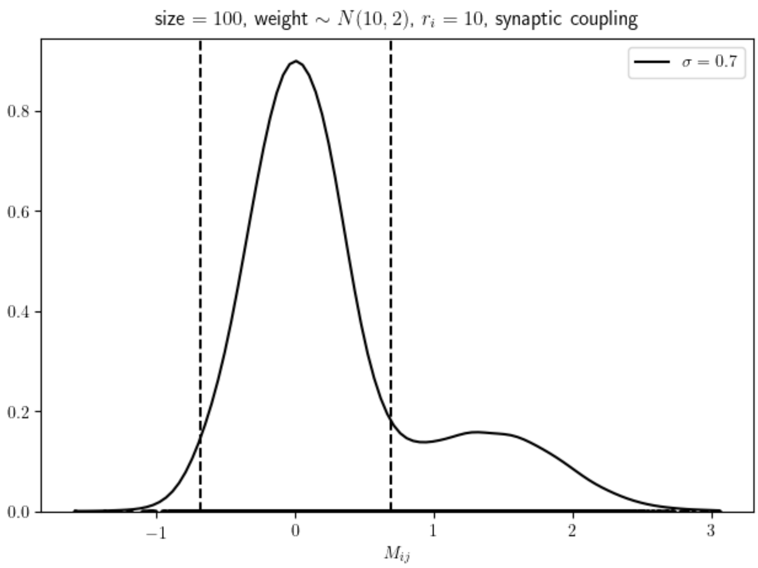
- (Same params but 20 nodes) fluctuations about steady values



- **Q vs M plot**

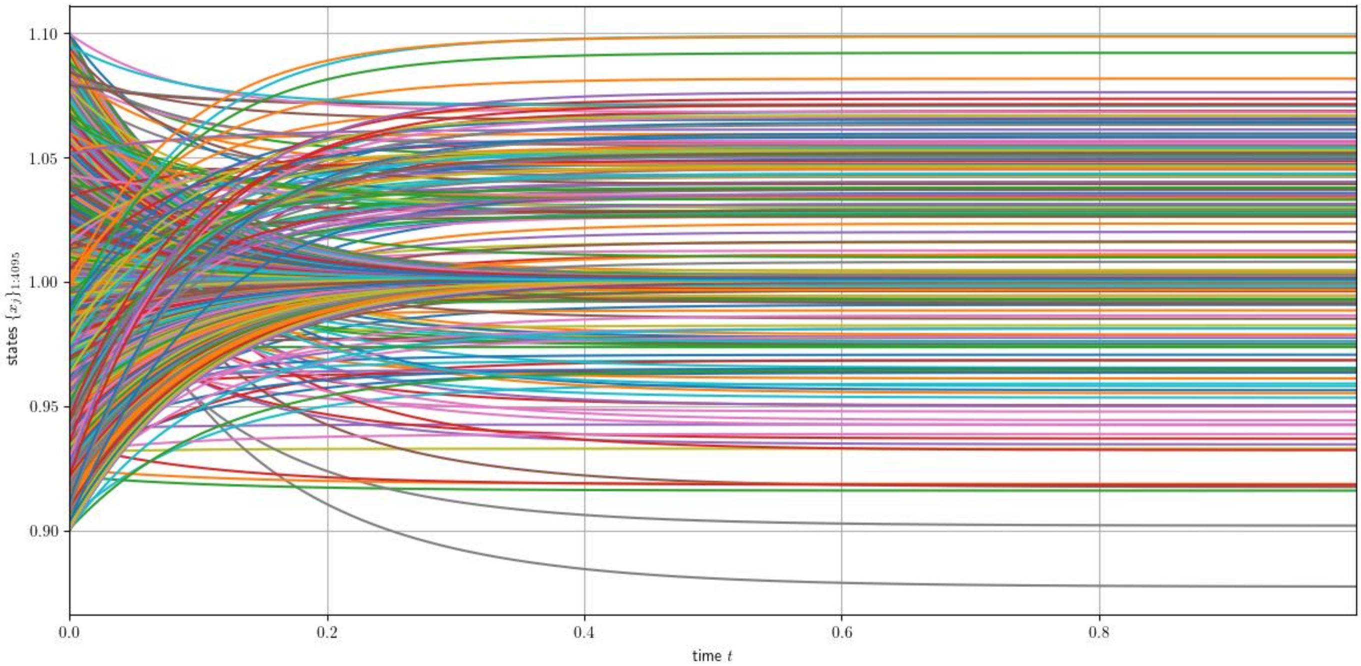


Two peaks: 0 & non-0



Dynamics (DIV25_PREmethod, synaptic coupling function)

- **Noise-free case: parameters**
  (a) (model-dependent) effective neuronal network
  (b) Size = 4095, from file "DIV25_PREmethod" (contains g_{ij})
  (c) Synaptic coupling h^{syn}, $(r_0 = 10, \beta_1 = 2, \beta_2 = 0.5, y_0 = 4, \sigma_i = 0.5)$
  (d) Initial conditions $\text{uniform}[0.9,1.1]$
  (e) Step size $\delta t = 2 \times 10^{-4}$, time steps $N_{\text{data}} = 1 \times 10^6$ (hence time series ends at $t = 200$)
  (f) Computation time: (run dynamics) 1 h 50 min for $N_{\text{data}} = 1 \times 10^6$
- **Noise-free case: time series**
  $N_{\text{data}} = 2 \times 10^3$
  Transient behavior before $t = 0.8$
  Steady values btw. 0.9 and 1.1 → initial conditions $\text{uniform}[0.9,1.1]$



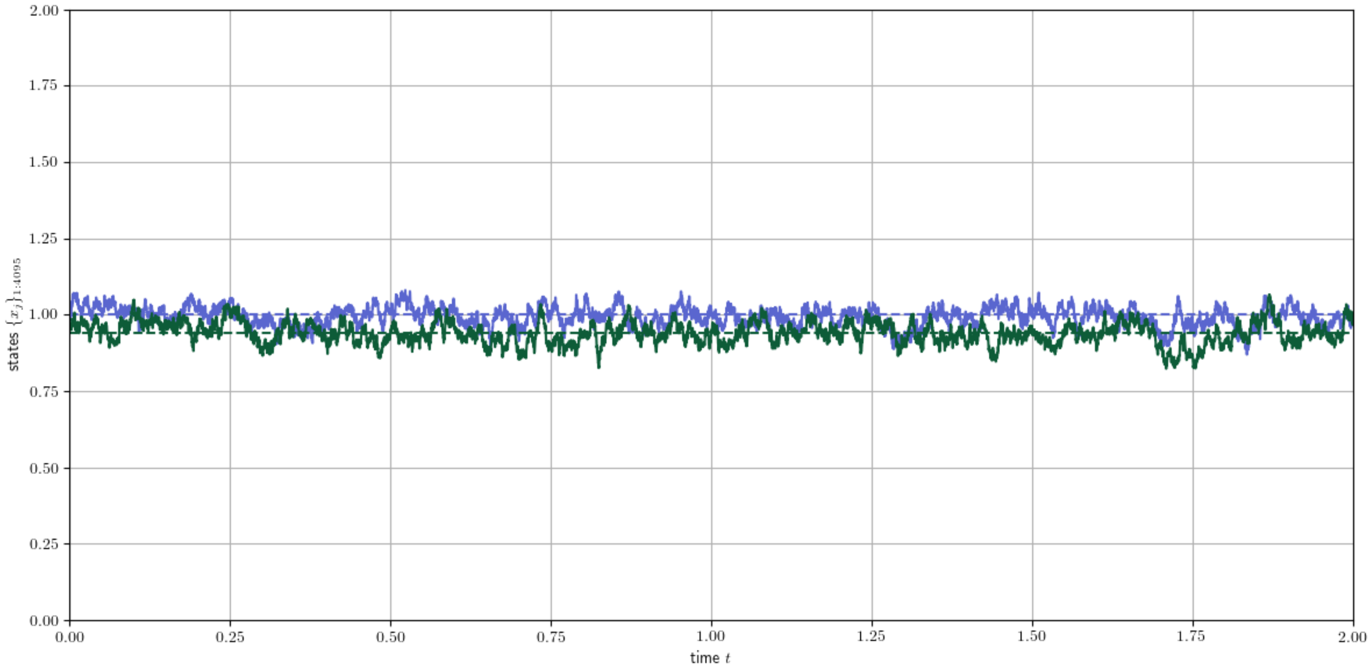
- **With-noise case**
  (h^{syn}) $y_0 \sim$ steady values X_i (X_i obtained under $y_0 = 4$)
  - finding the right set of params
  $(r_0, \beta_1, \beta_2, y_0 = 1, \sigma_i)$
  Fail (divergent dynamics)
  1. (10,2,0.5,1,0.5)
  2. (10,10,0.5,1,0.5)
  3. (10,50,0.5,1,0.5) [stable up to $N_{\text{data}} = 2 \times 10^5$]
  4. (10,2,0.1,1,0.5)
  5. (50,2,0.5,1,0.5)
  6. (10,20,1,1,0.5) [case 9ii]

  OK (stable dynamics, up to $N_{\text{data}} = 1 \times 10^6$)
  1. (100,2,0.5,1,0.5) [Rationale: stronger tendency to revert to 1]
  2. (10,20,1,1,0.25) [case 9ii, from Chris] [Rationale: weaker couplings + smaller noise]

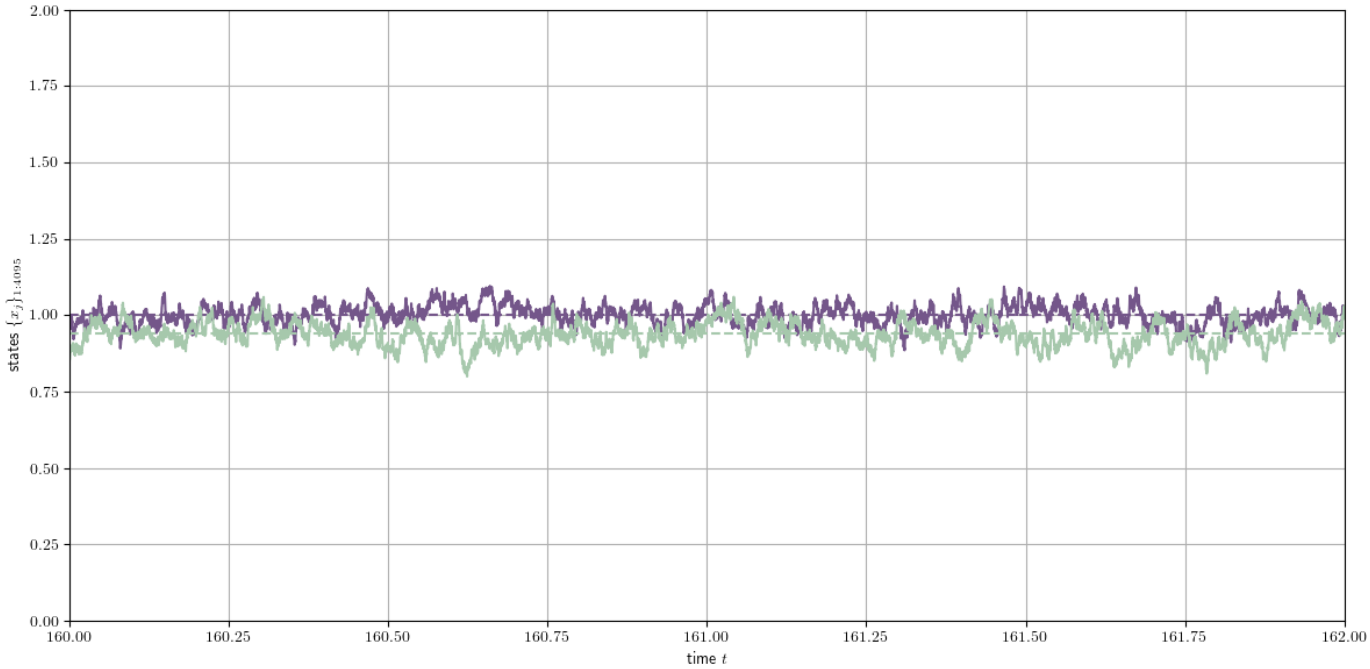
  Used (100,2,0.5,1,0.5) & (10,20,1,1,0.25) for the following analysis.

(a) With-noise case: (100,2,0.5,1,0.5)

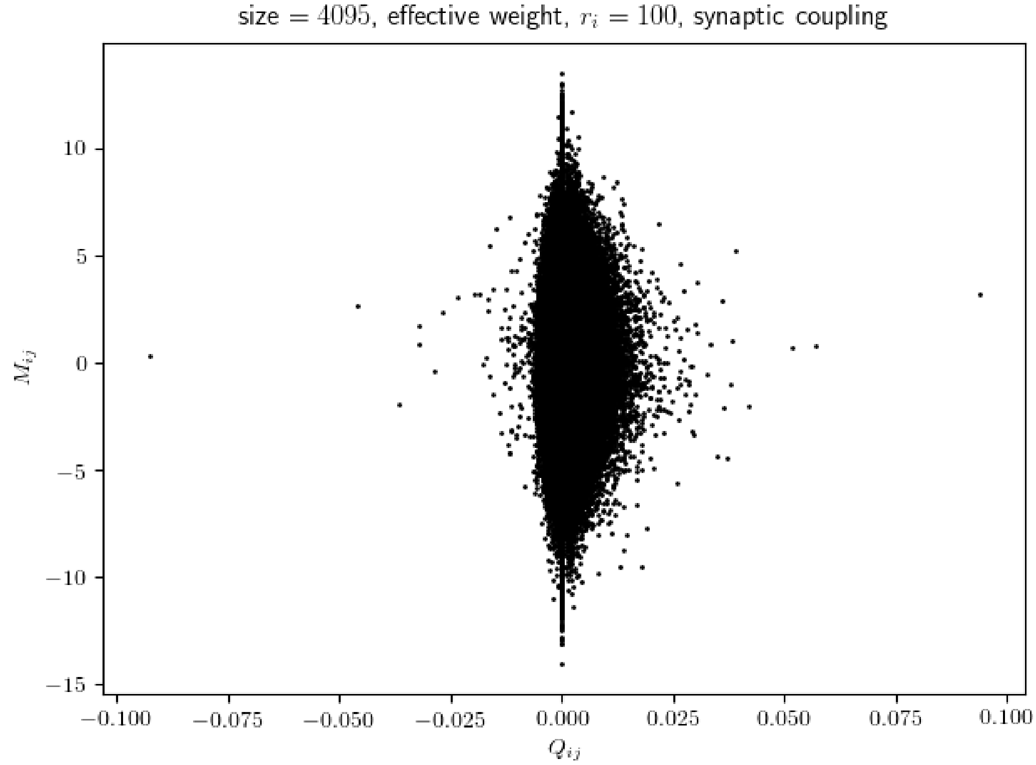
- **Fluctuations around steady values (note: node 0 & 4)**
  Time: 0 to 2
  Recall: transient behavior before $t = 0.8$



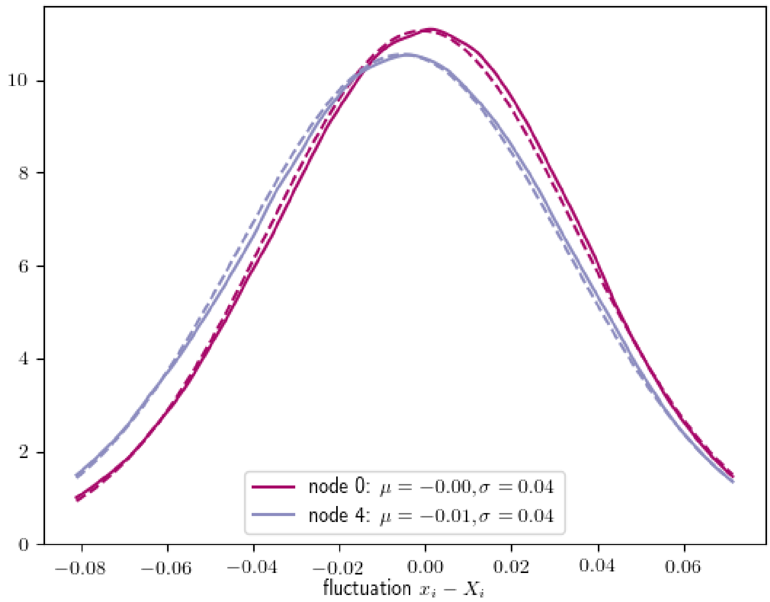
Time: 160 to 162
Still stable, as expected



- **Condition $\tau \max_i |\mathrm{Im}\,\lambda_i| < \pi$ is true → $\log \exp Q = Q$, i.e., OK to take log**
- **Q vs M plot ($N_{\mathbf{data}} = 4 \times 10^5$)**
  Linear pattern not obvious, need longer time series

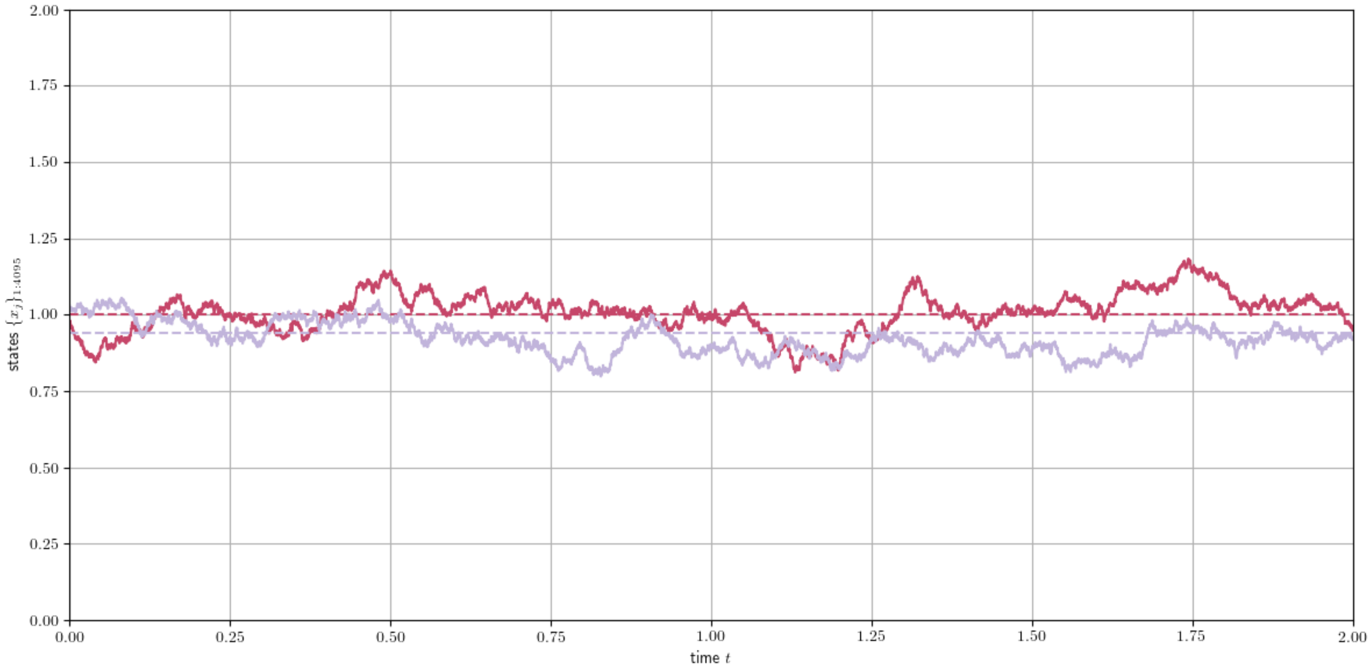


- **Distribution of fluctuations ($N_{\mathbf{data}} = 1 \times 10^6$)**
  Gaussian, $\mathrm{s.d.} < \sigma_i = 0.5$
  Network connectivity modifies s.d. of fluctuation but not shape (Gaussian noise → Gaussian fluc.)

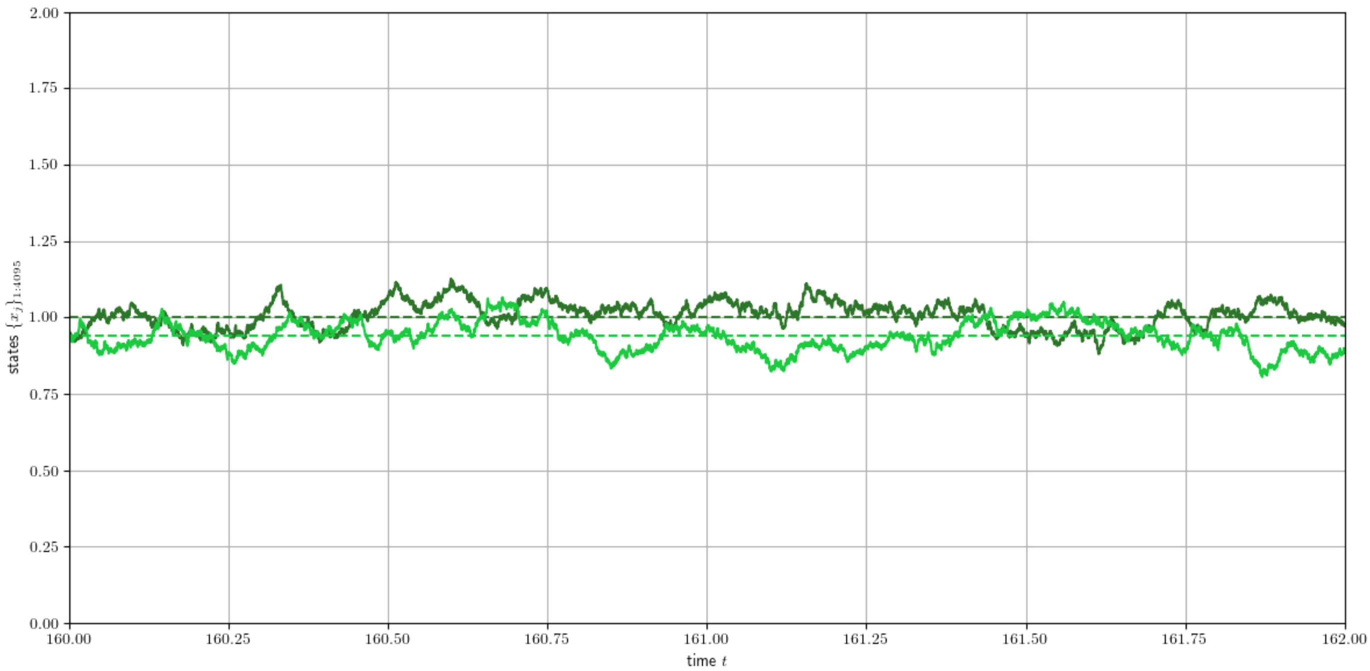


(b) With-noise case: (10,20,1,1,0.25)

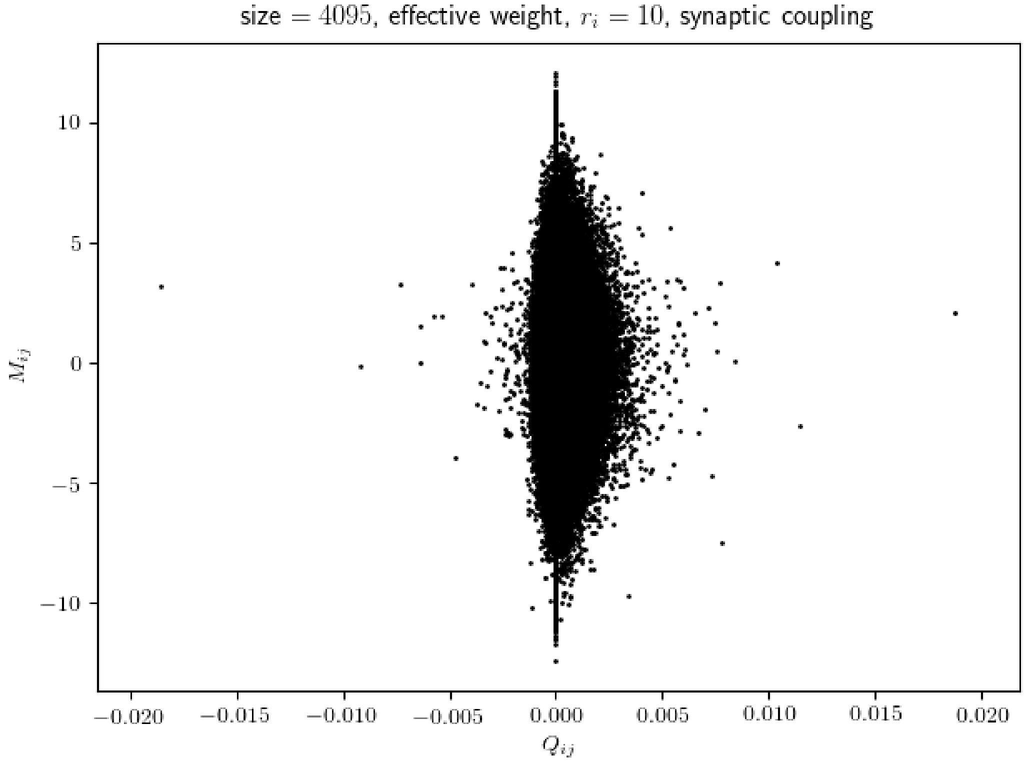
- **Fluctuations around steady values (note: node 0 & 4)**
  Time: 0 to 2
  Recall: transient behavior before $t = 0.8$



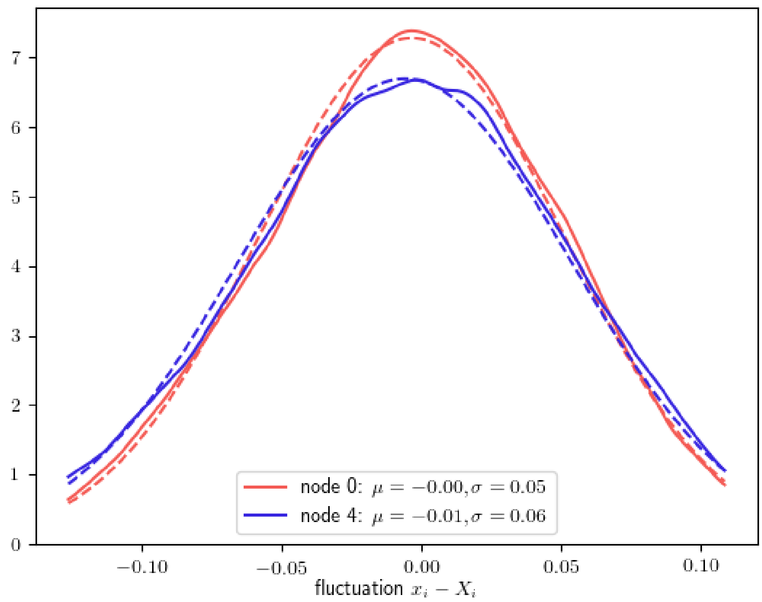
Time: 160 to 162
Still stable, as expected



- **Condition $\tau \max_i |\mathrm{Im}\, \lambda_i| < \pi$ is true → $\log \exp Q = Q$, i.e., OK to take log**
- **Q vs M plot ($N_{\mathrm{data}} = 4 \times 10^5$)**
  Linear pattern not obvious, need longer time series



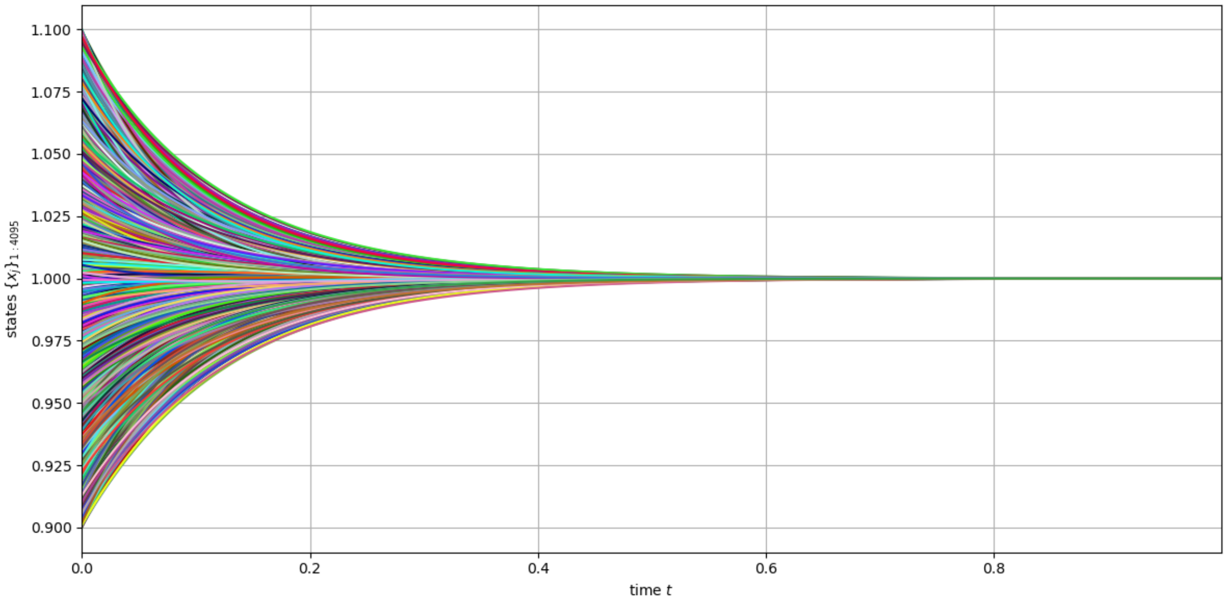
- **Distribution of fluctuations ($N_{\mathrm{data}} = 1 \times 10^6$)**
  Gaussian, s.d. $< \sigma_i = 0.25$
  Network connectivity modifies s.d. of fluctuation but not shape (Gaussian noise → Gaussian fluc.)



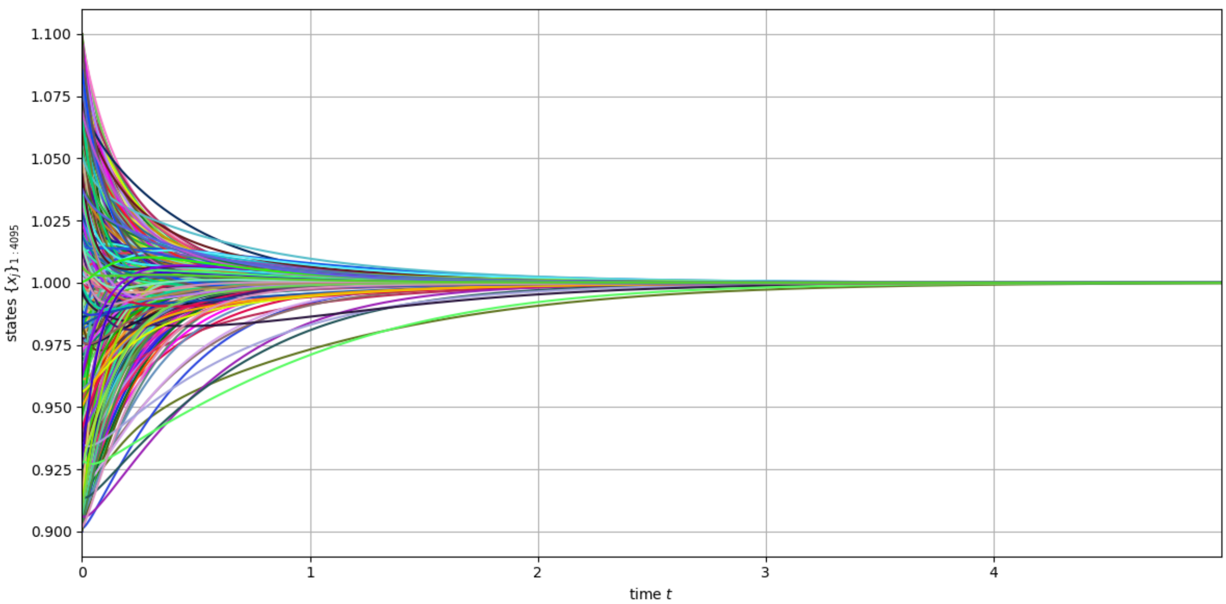
Dynamics (DIV25_PREmethod, diffusive coupling function)

- **Model params: diffusive, $\left(r_0 = 10, g_{ij} \times 10, \sigma_i = 0.25\right)$**

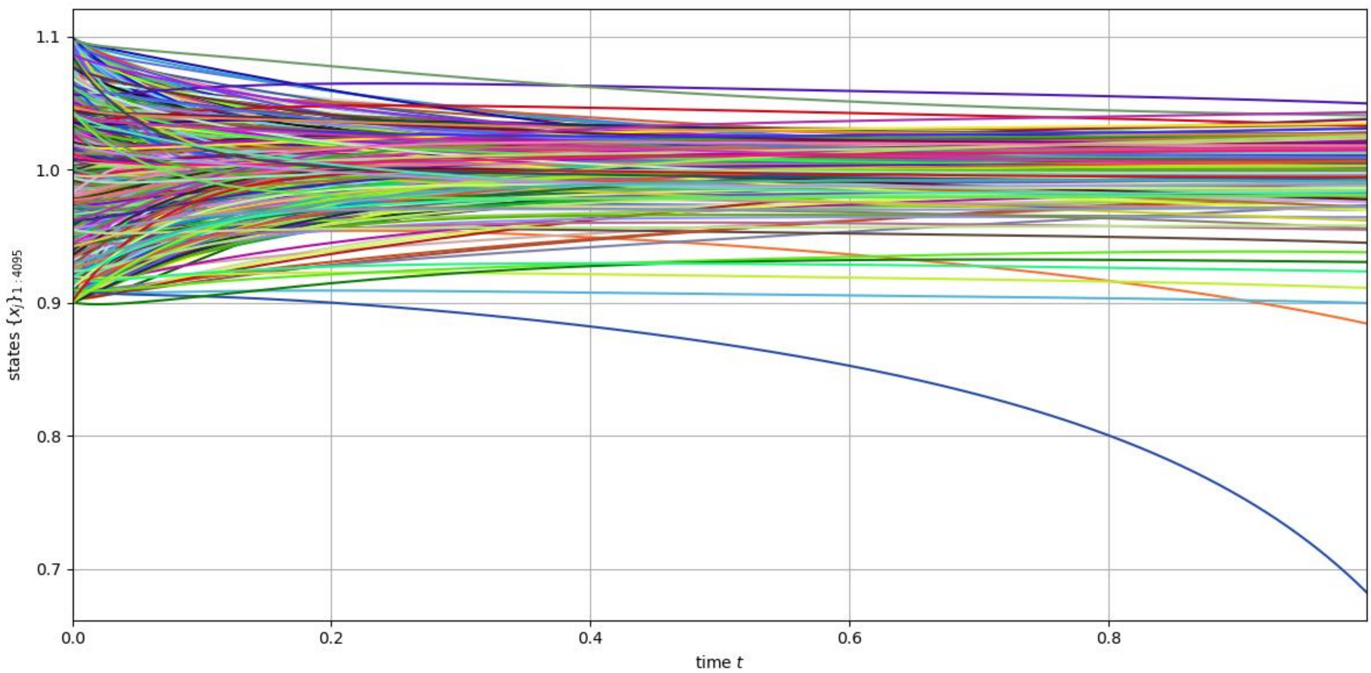
- **Noise-free case: parameters**
  (a) (model-dependent) effective neuronal network
  (b) Size = 4095, from file "DIV25_PREmethod" (contains g_{ij})
  (c) **Diffusive coupling h^{diff}, $\left(r_0 = 10, g_{ij} \times 10\right)$**
  (d) Initial conditions $\text{uniform}[0.9,1.1]$
  (e) Step size $\delta t = 5 \times 10^{-4}$, time steps $N_{\text{data}} = 2 \times 10^6$ (hence time series ends at $t = 1000$)
  (f) Computation time: (run dynamics) 3 h 30 min for $N_{\text{data}} = 2 \times 10^6$
- **Noise-free case: time series**
  1. $\left(r_0 = 10, g_{ij} \times 10\right)$: $g_{ij} \times 10 \Rightarrow \max g_{ij} \sim 0.3, \min g_{ij} \sim -0.3$
     $N_{\text{data}} = 2 \times 10^3$
     Transient behavior before $t = 0.8$
     Steady values all 1 → initial conditions $\text{uniform}[0.9,1.1]$



  2. $\left(r_0 = 10, g_{ij} \times 40\right)$: similar behaviour but longer transient behaviour
     Transient behavior before $t = 4$



3. $\left(r_0 = 10, g_{ij} \times 50\right)$: divergent
   Tend to be less stable when $g_{ij} \sim r_0$



- **With-noise case**
  Finding the right set of params
  $\left(r_0, g_{ij} \times \text{multiplier}, \sigma_i\right)$
  Fail (divergent dynamics)
  1. (10,10,0.5)

  OK (stable dynamics, up to $N_{\text{data}} = 2 \times 10^6$)
  1. (10,10,0.25)

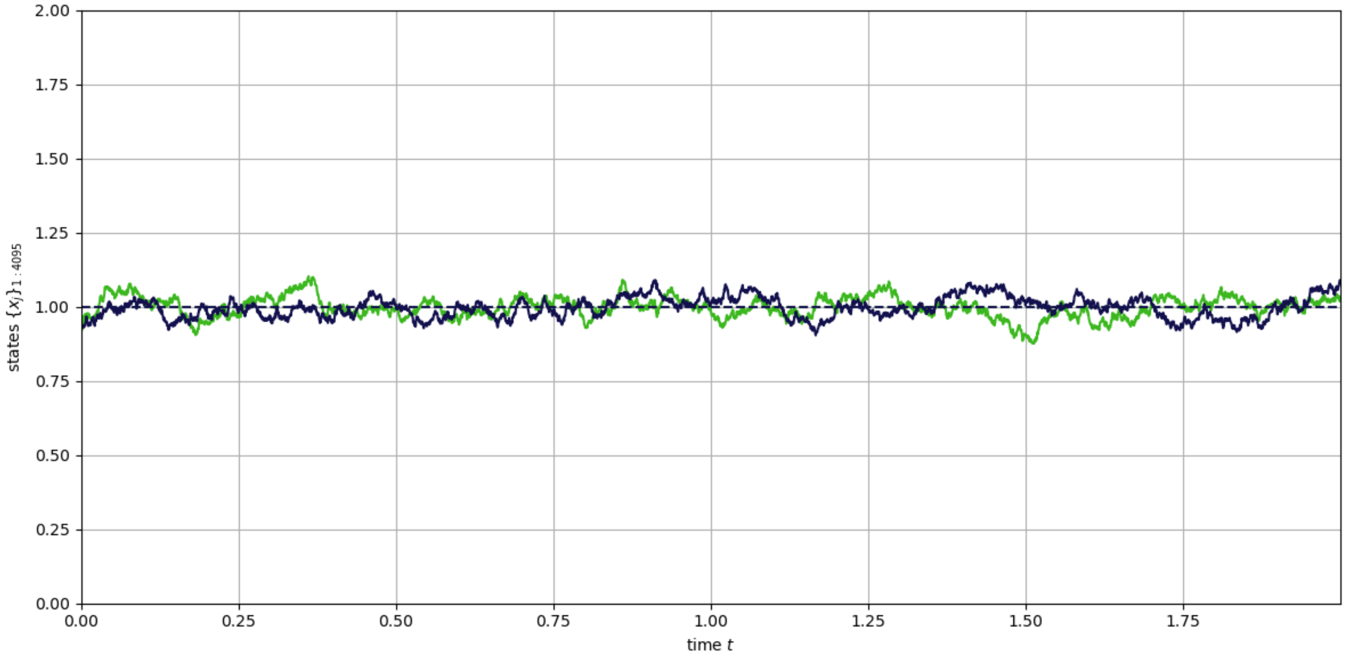
  Used (10,10,0.25) for the following analysis.

With-noise case: (10,10,0.25)

- **Fluctuations around steady values (note: node 0 & 1)**

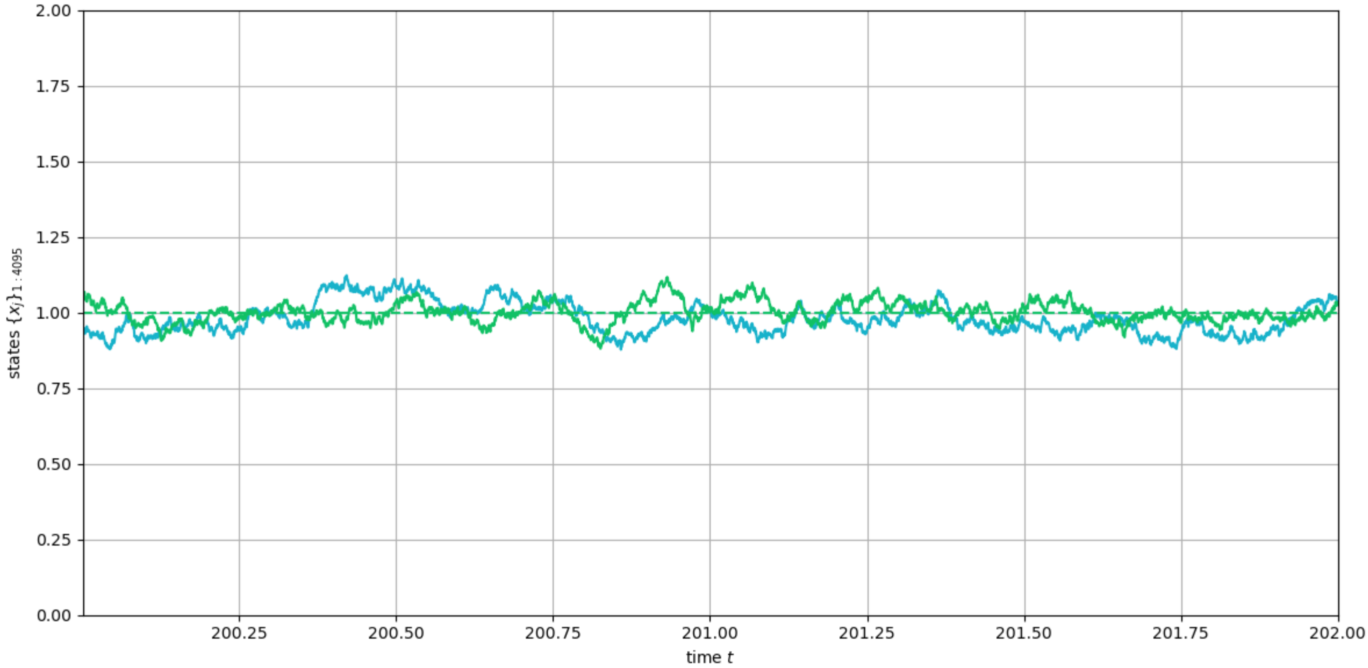
  Time: 0 to 2

  Recall: transient behavior before $t = 0.8$, steady values all 1

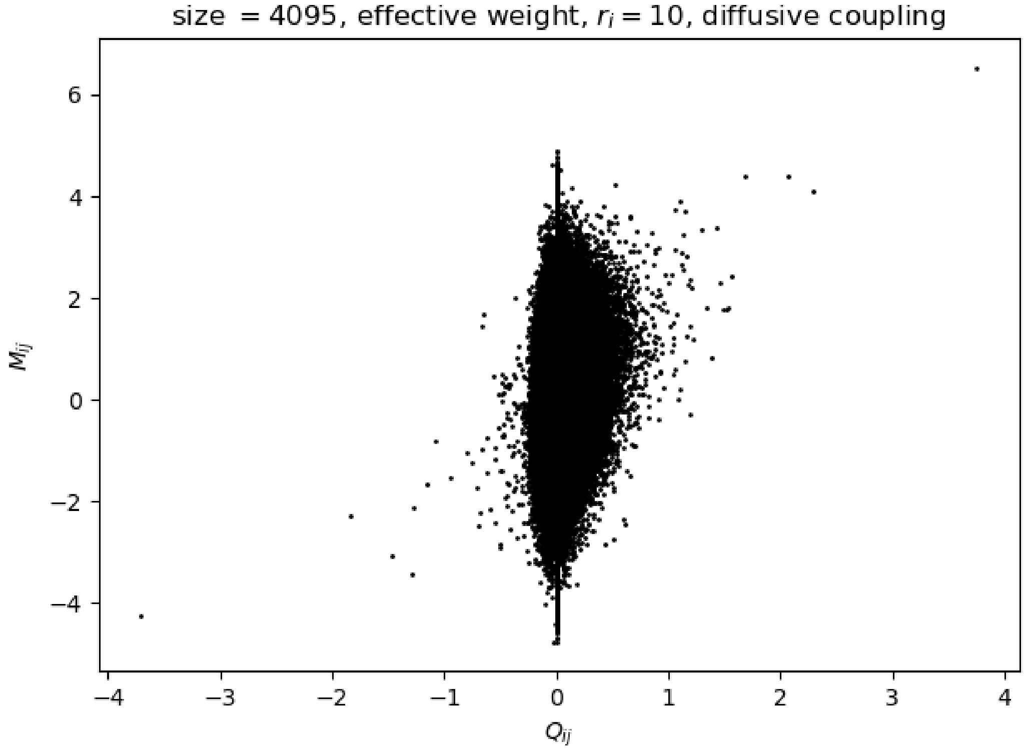


  Time: 200 to 202

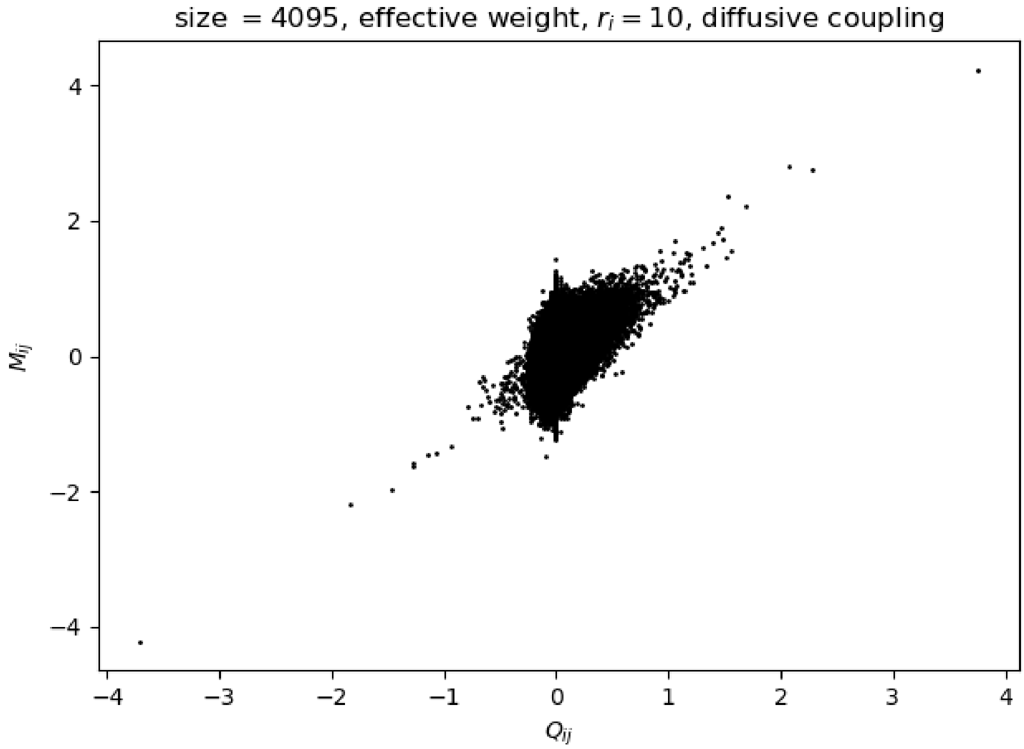
  Still stable, as expected



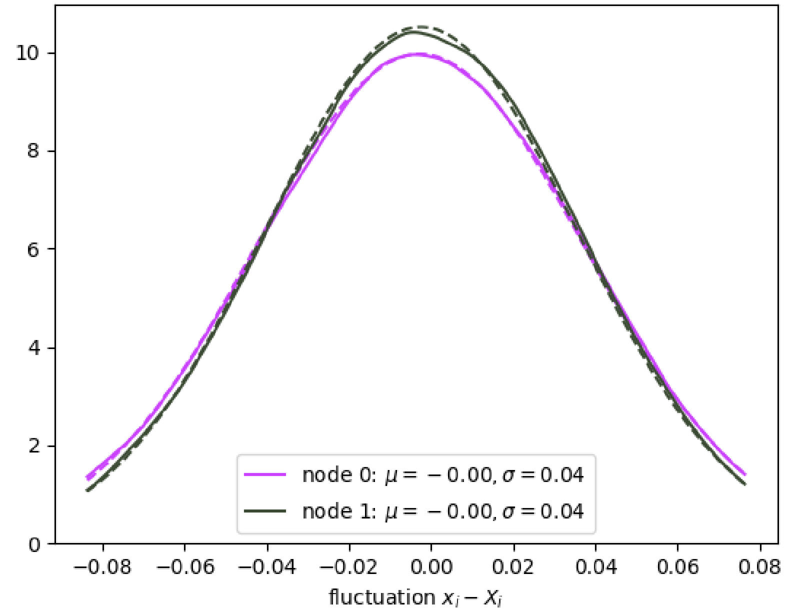
- **Condition $\tau \max_i |\mathrm{Im}\, \lambda_i| < \pi$ is true → $\log \exp Q = Q$, i.e., OK to take log**
- **Q vs M plot ($N_{\mathbf{data}} = 4 \times 10^5$)**
  Linear pattern starting to become obvious, M_{ij} & Q_{ij} have same order
  M converges faster to Q with diffusive coupling function



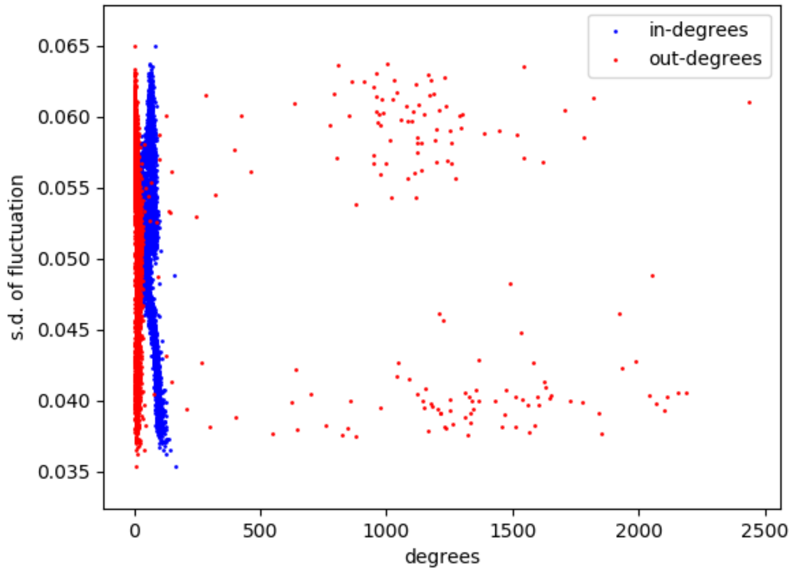
- **Q vs M plot ($N_{\mathbf{data}} = 2 \times 10^6$)**

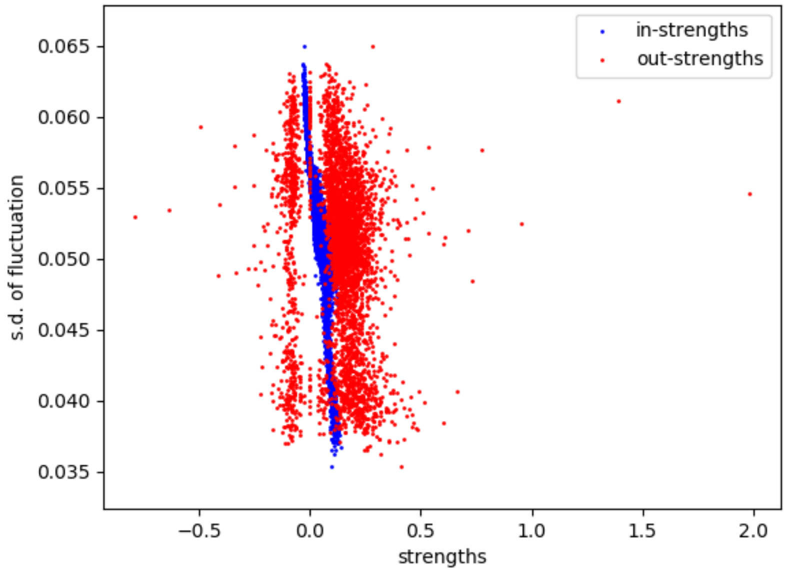


- **Distribution of fluctuations ($N_{\mathbf{data}} = \mathbf{1} \times \mathbf{10^6}$)**
  Gaussian, $\mathrm{s.d.} < \sigma_i = 0.25$
  Network connectivity modifies s.d. of fluctuation but not shape (Gaussian noise → Gaussian fluc.)



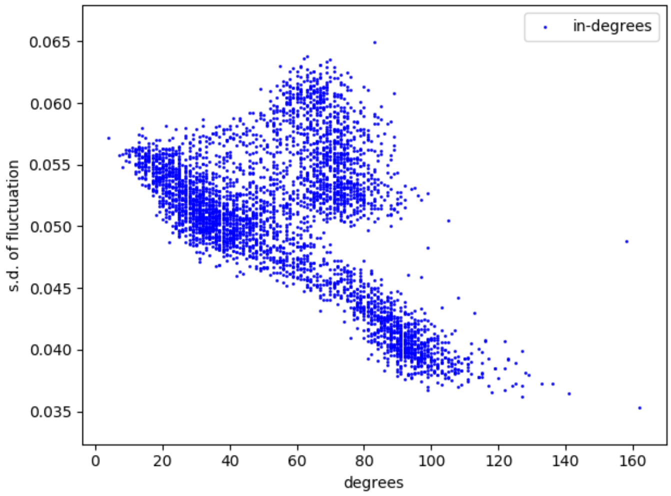
- **Dependence of s.d. of fluctuations on degrees and strengths ($N_{\mathbf{data}} = \mathbf{1} \times \mathbf{10^6}$)**

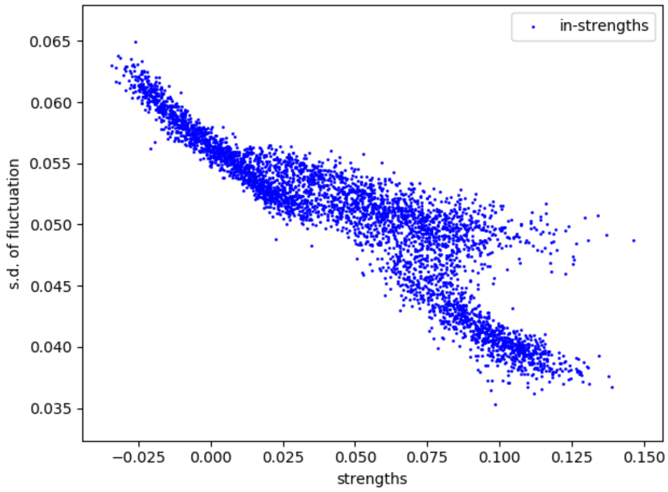




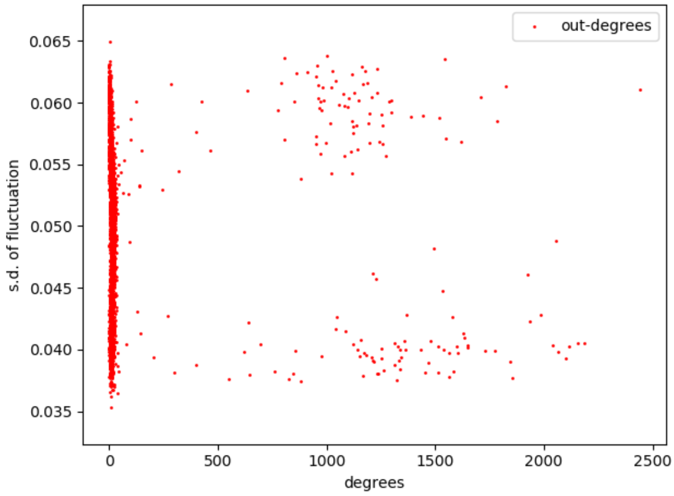
Linear pattern for in-strengths

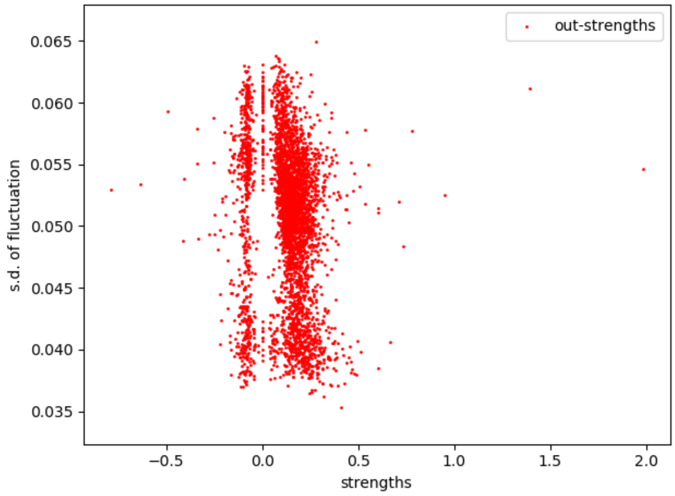
- **Dependence of s.d. of fluctuations on in-degrees and in-strengths ($N_{\text{data}} = 1 \times 10^6$)**





- **Dependence of s.d. of fluctuations on out-degrees and out-strengths ($N_{\text{data}} = 1 \times 10^6$)**





- **Cross-sectional distribution (distribution of all nodes at some time): time-evolution**
  Time: 0 to 1
  Evolution: uniform → normal
  https://drive.google.com/file/d/1LrP6pSeSgs6OIdGqB56v5eANH0lWatan/view?usp=sharing

- **Spiking detection**
  In first 1e5 steps, node 0 reaches its max at around $t = 35.55$
  Time series of node 0 & its adjacent nodes in neighborhood of $t = 35.55$:
  https://drive.google.com/file/d/1fQD313ho7DmrQsBmWhyej_O4P7Nm8QED/view?usp=sharing

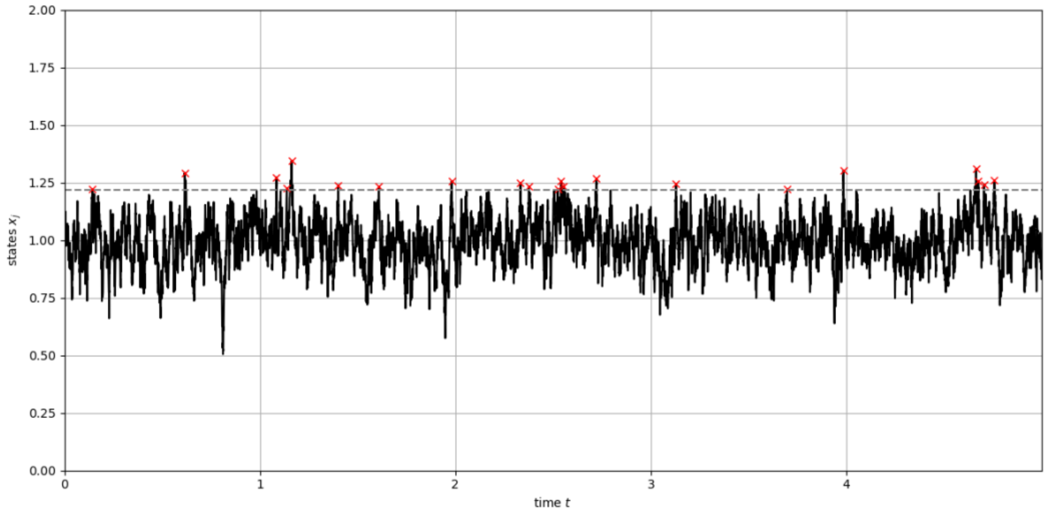
Spike analysis (peak histogram, raster plot)

- **Model params: diffusive, $\left(r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5\right)$**
- **$\sigma_i = 1.5$ to magnify spiking behavior**

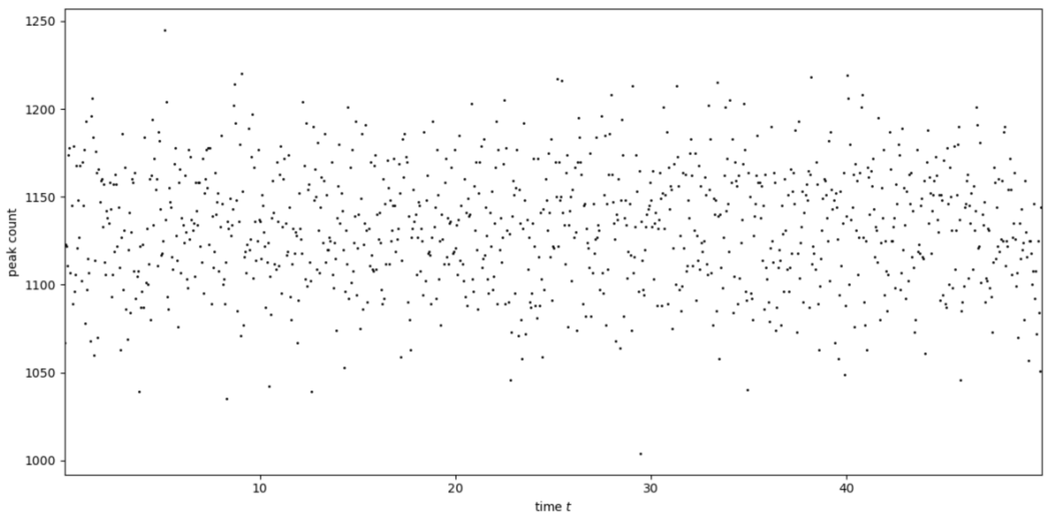
- **With-noise case: params**
  $\left(r_0, g_{ij} \times \text{multiplier}, \sigma_i\right) = \left(r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5\right)$
  ⇒ larger noise to look for more apparent spiking

- **[last time] (Visual) spiking detection**
  In first 1e5 steps, node 0 reaches its max at around $t = 9.9$
  Time series of node 0 & its adjacent nodes in neighborhood of $t = 9.9$:
  https://drive.google.com/file/d/18MMoLGlIHhWUJZ3vGw1XtBM-c7N_4vqJ/view?usp=sharing
  (Visual) correlation for node 14,162,223,329,406,4041

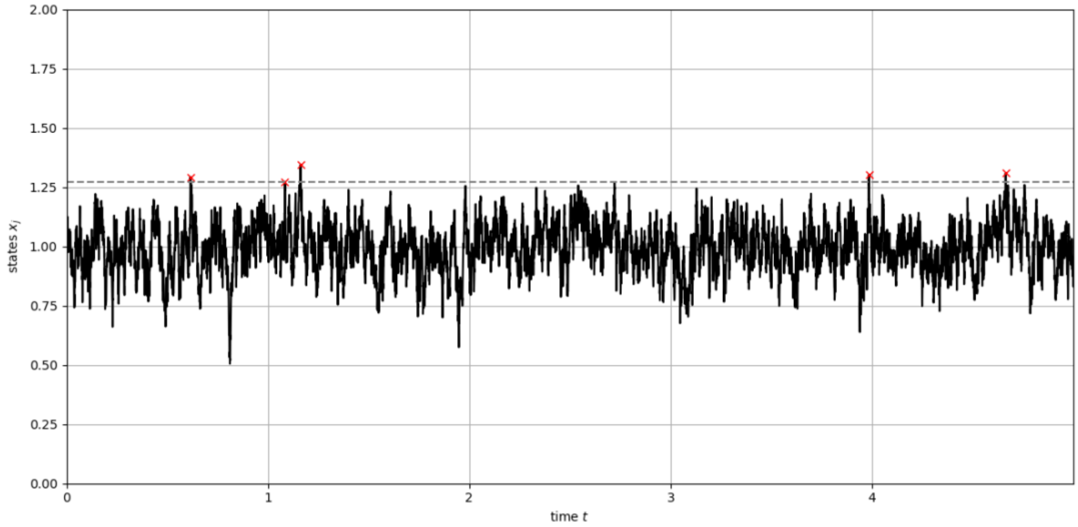
- **(Numerical) spiking detection**
  Criteria:
  (a) Peak value > steady value + ($\alpha = 2$ or 2.5) * s.d. of fluctuations (of all nodes) over $[0, T]$, s.d. of fluctuations ~ 0.1
  (b) Time steps btw. peak values > $d = 20$ (avoid multiple-counting for very close peaks)
- **$\alpha = 2$, $d = 20$, node 0 dynamics & peaks over $[0, 5]$**



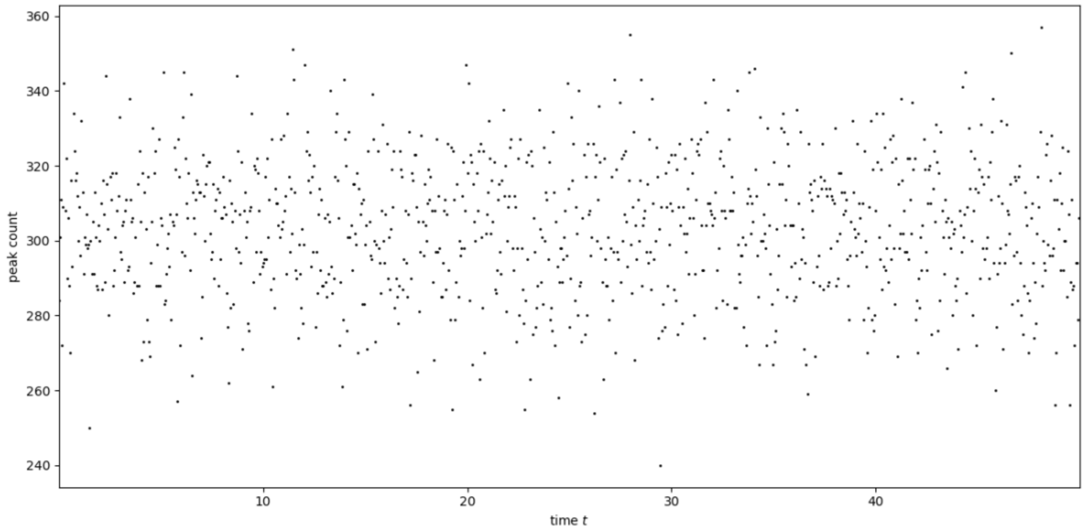
Peak histogram over $[0, 50]$ (i.e. 1e5 steps), 1000 bins
To look for concentrations of peaks but distribution appears to be uniform



- $\alpha = 2.5, d = 20$, **node 0 dynamics & peaks over** $[0, 5]$

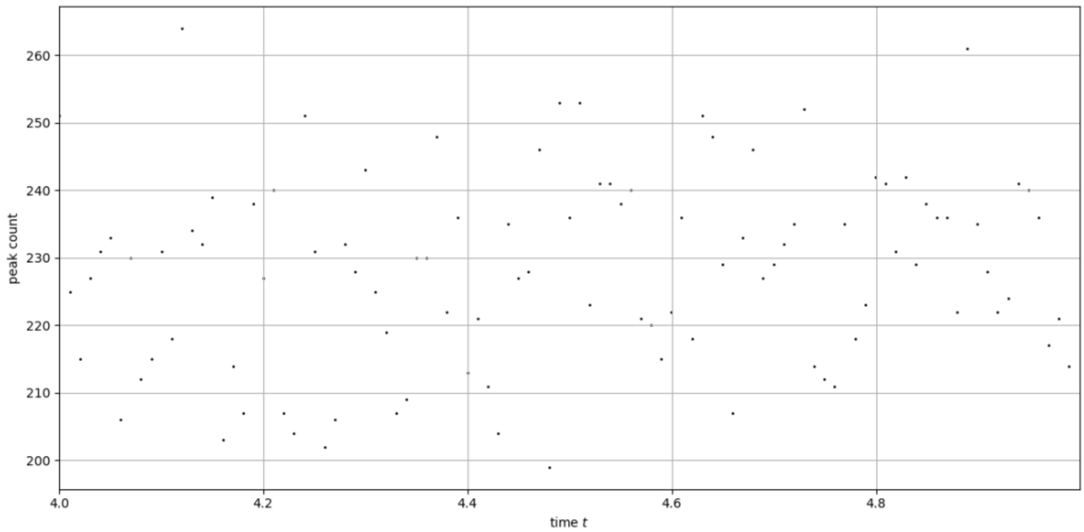


Peak histogram over $[0, 50]$ **(i.e. 1e5 steps), 1000 bins**
To look for concentrations of peaks but distribution appears to be uniform

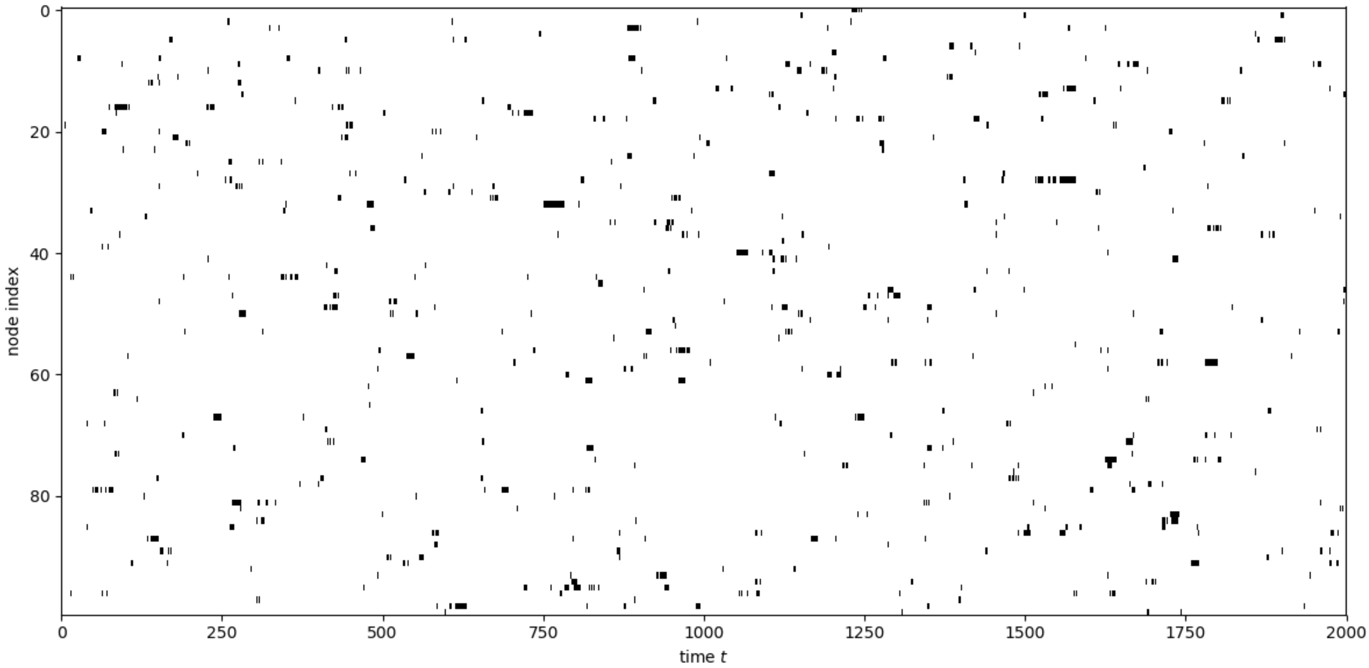


- $\alpha = 2, d = 20$, **node 0 to 100 over** $[4, 5]$
https://drive.google.com/file/d/1W20hD-8YxLiXhi6n7G5I08lmtS4nomNM/view?usp=sharing
To visually inspect correlated spiking activities

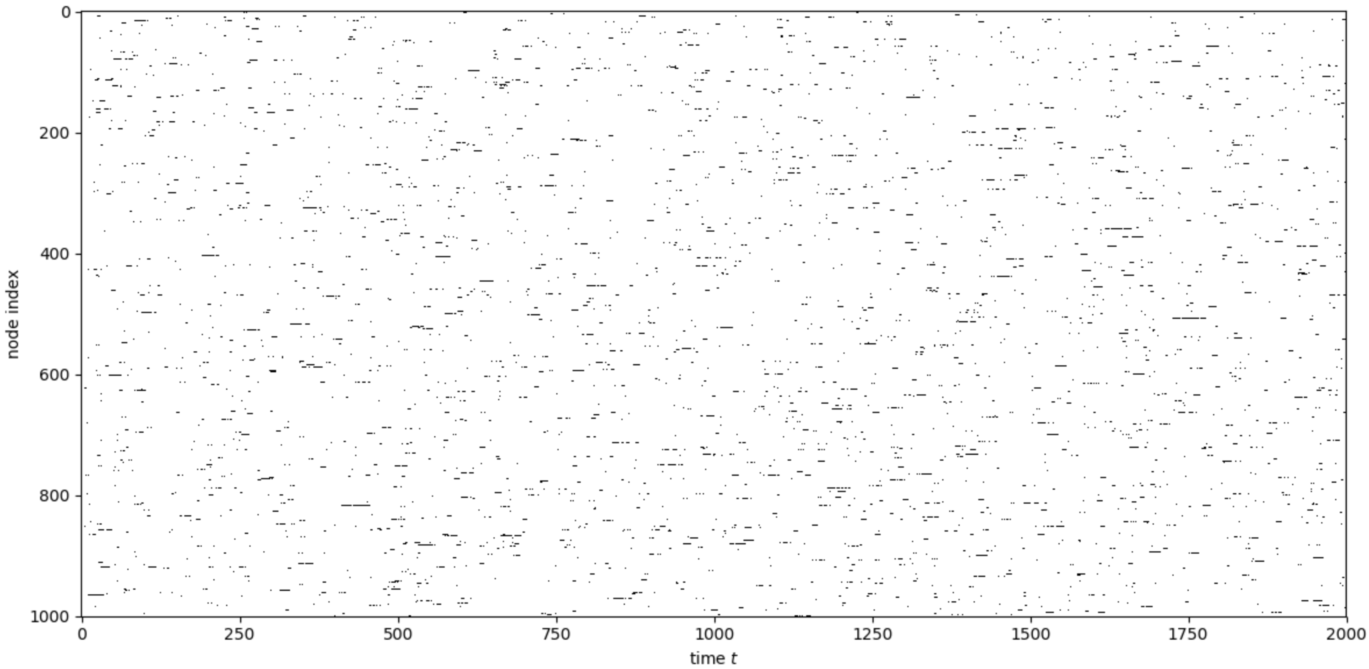
Peak histogram over $[4, 5]$ **(i.e. 2e3 steps), 100 bins**



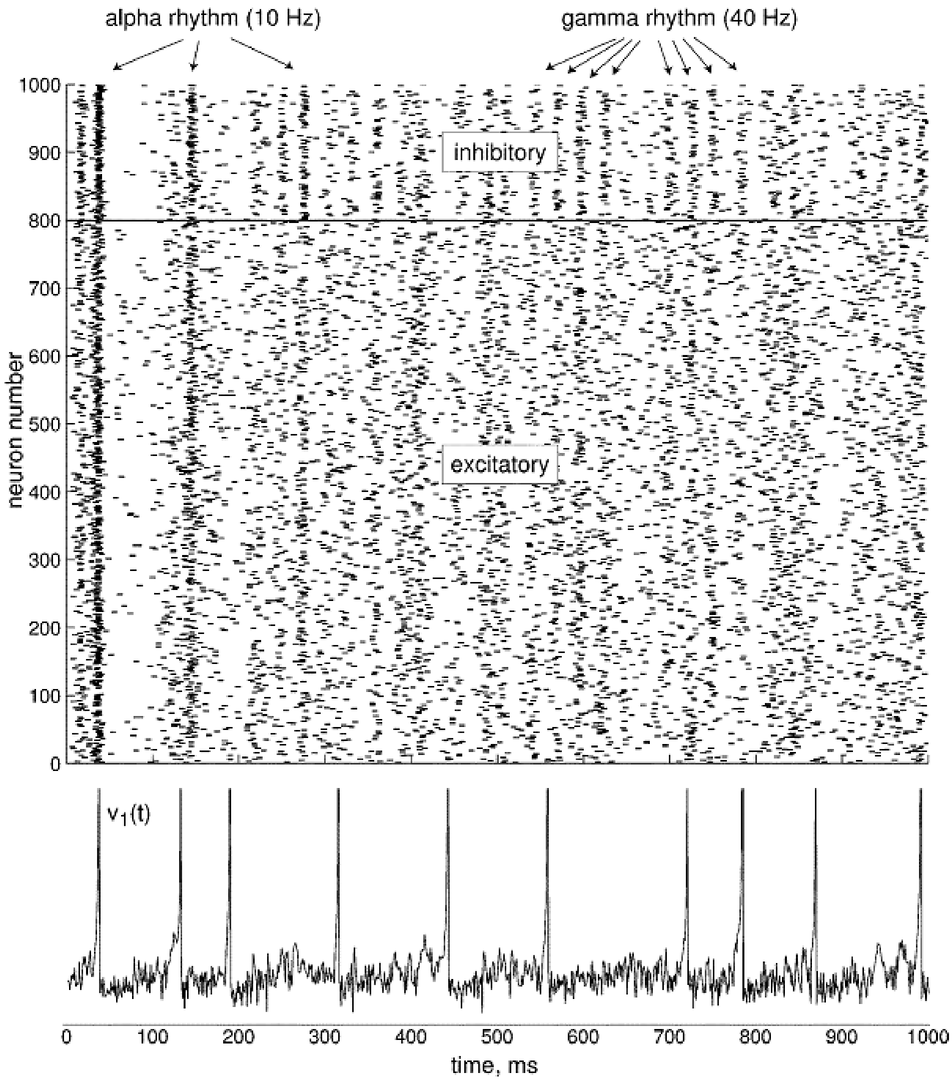
- **Raster plot: $\alpha = 2, d = 0$ (allow "continuous" peaks)**
  Node 0 to 100, 2e3 steps



Node 0 to 1000, 2e3 steps



- **Neuron raster plot found on internet:**
  Not similar …



<u>Spike analysis (peak distribution, peak vs strength plot)</u>

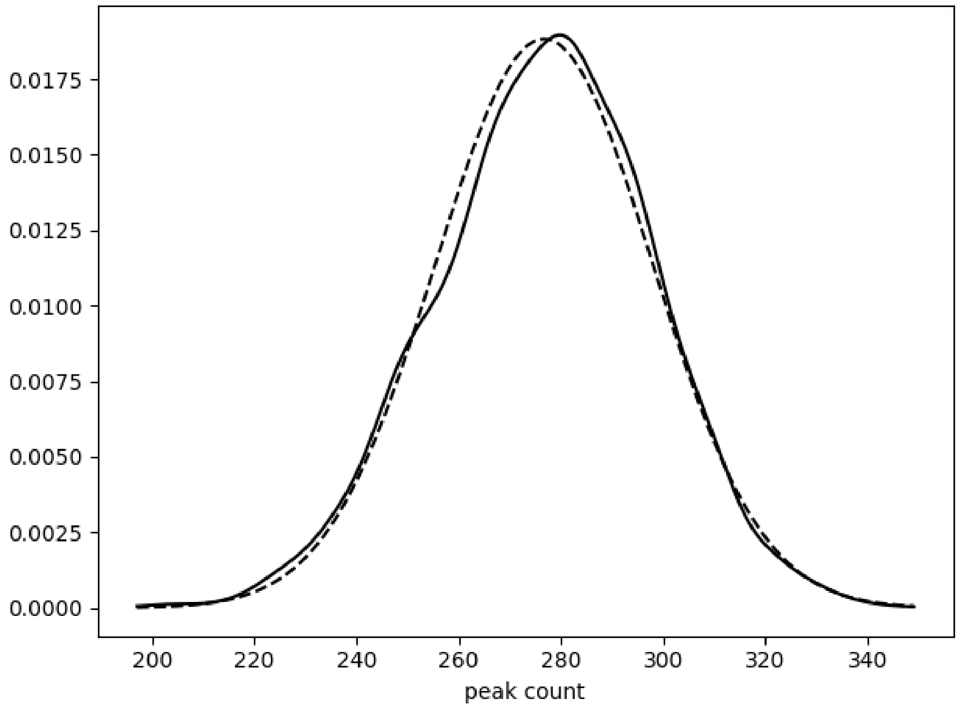
- **Model params: diffusive, $\left(\boldsymbol{r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5}\right)$**
- **Draw comparison btw. simulation data & experimental data**

- **(Continuing from last time)** **With-noise case**: **params**
  $\left(r_0, g_{ij} \times \text{multiplier}, \sigma_i\right) = \left(r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5\right)$
  ⇒ larger noise to look for more apparent spiking

- **(Continuing from last time)** **(Numerical) spiking detection**
  Criteria:
  (a) Peak value > steady value + ($\alpha = 2$ or 2.5) * s.d. of fluctuations (of all nodes) over $[0, T]$, s.d. of fluctuations ~ 0.1
  (b) Time steps btw. peak values > $d = 20$ (avoid multiple-counting for very close peaks)

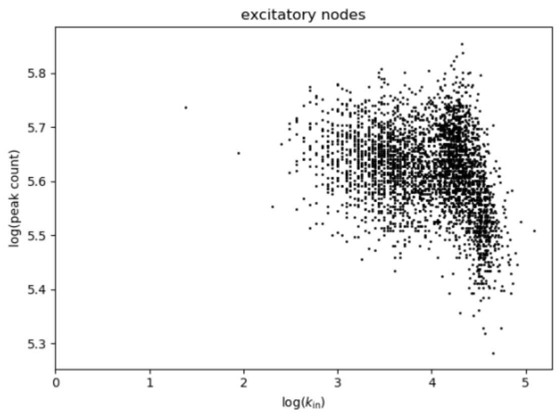
- Note for the following plots:
  (a) **Peak criteria: $\boldsymbol{\alpha = 2, d = 20}$**
  (b) Number of peaks range from 200 to 340 **(1e5 steps)**

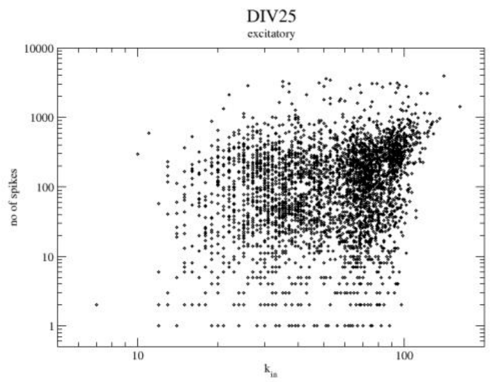
- **Distribution of number of peaks (near normal)**



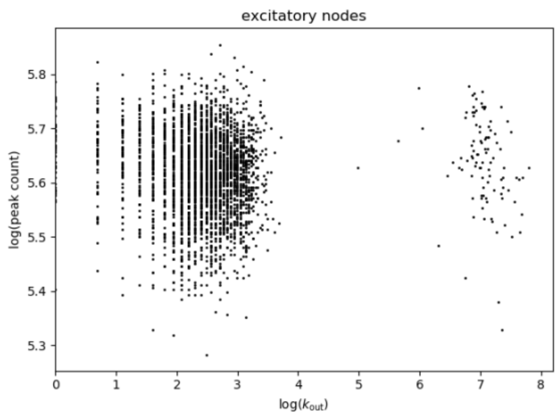
- **Excitatory nodes**

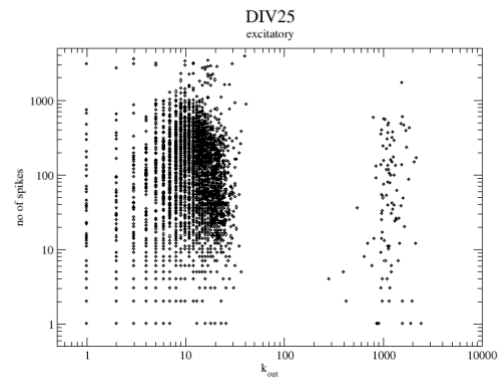
log peak count vs log k_in



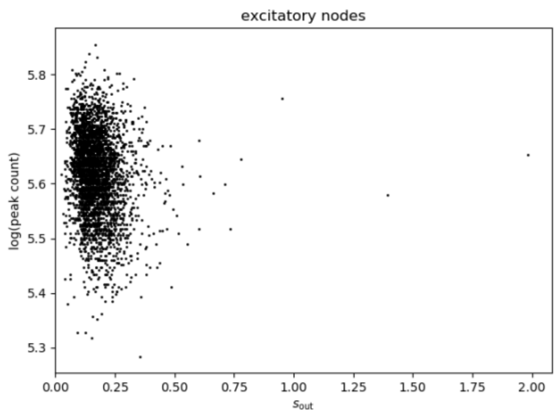


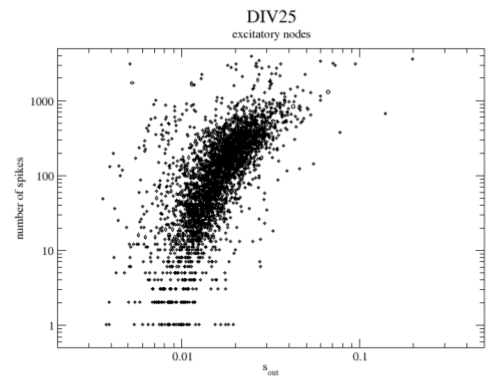
log peak count vs log k_out



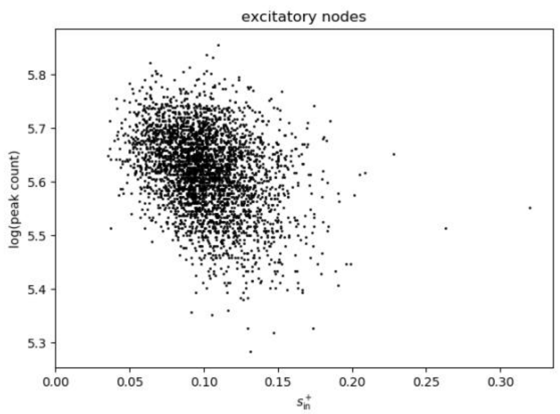


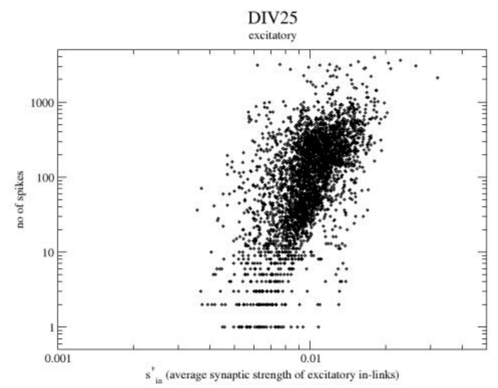
log peak count vs s_out





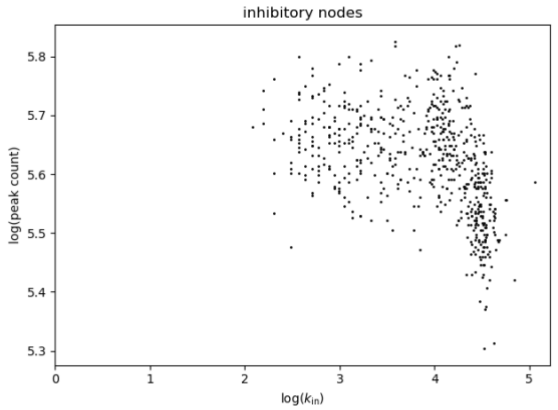
log peak count vs s^+_in

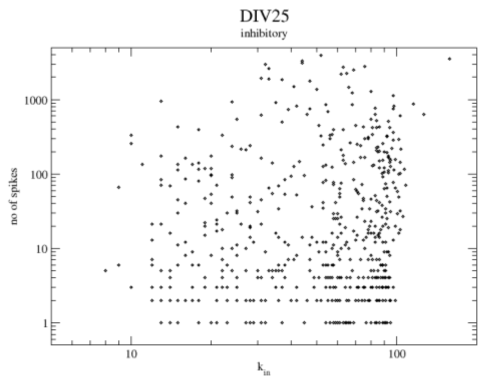




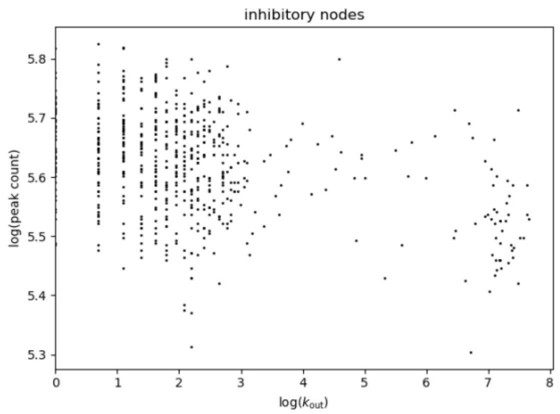
- **Inhibitory nodes**

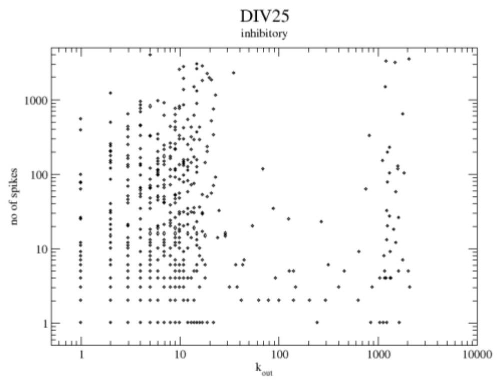
log peak count vs log k_in



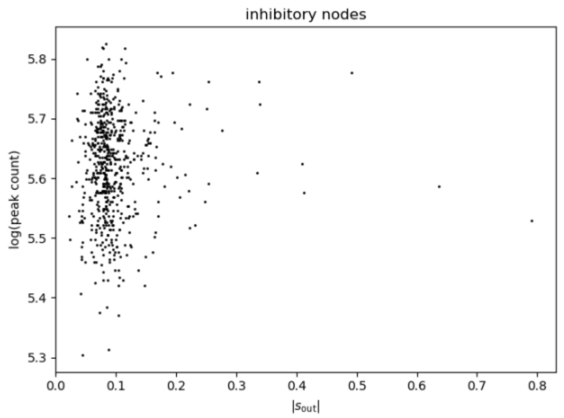


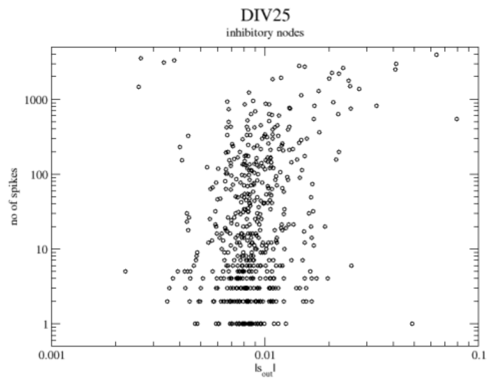
log peak count vs log k_out



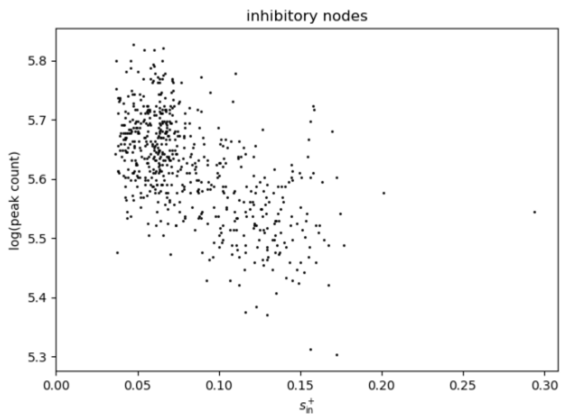


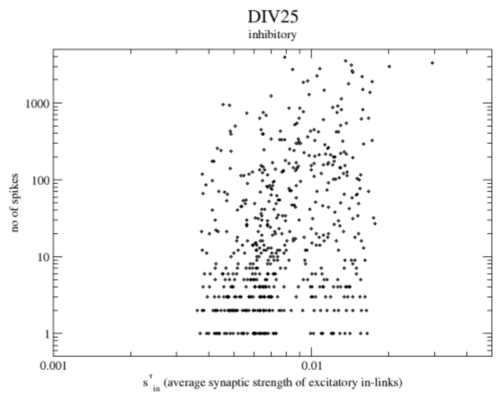
log peak count vs |s_out|





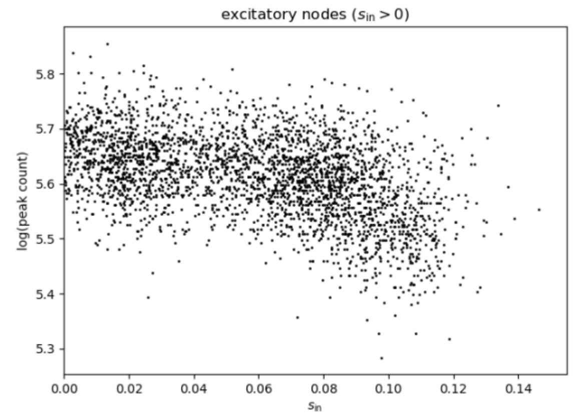
log peak count vs s^+_in

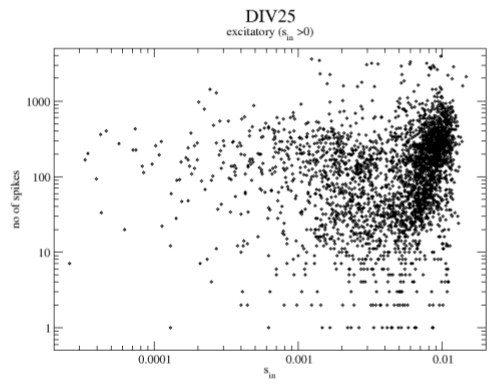




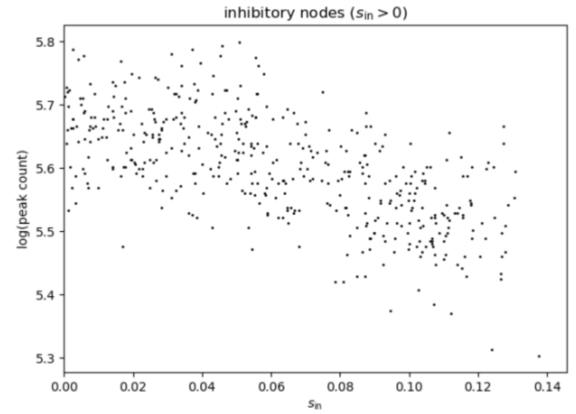
- **Mixed (x-axis: s_in or |s_in|)**

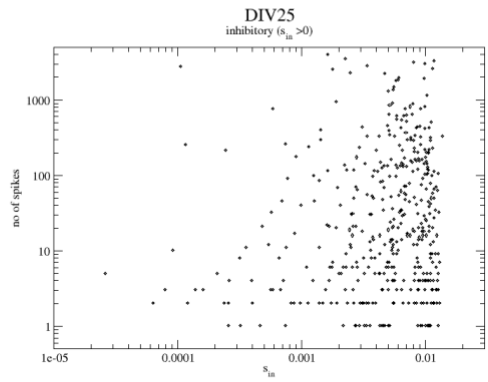
s_in > 0 & s_out > 0



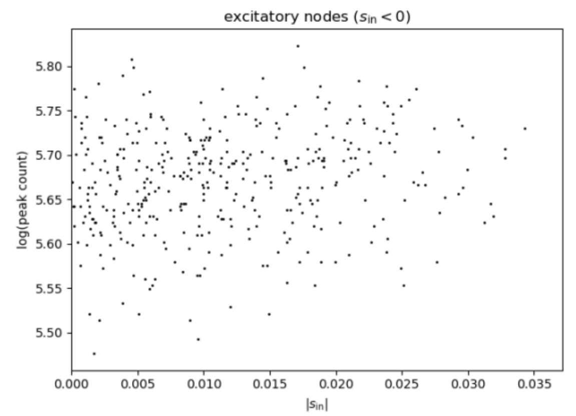


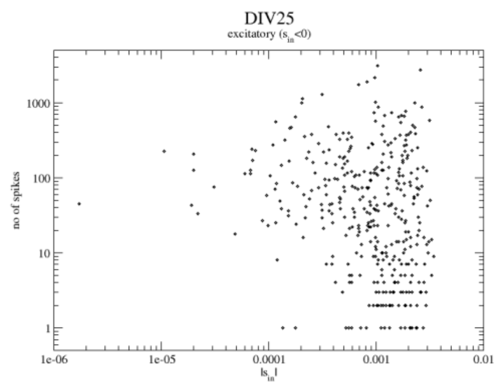
s_in > 0 & s_out < 0



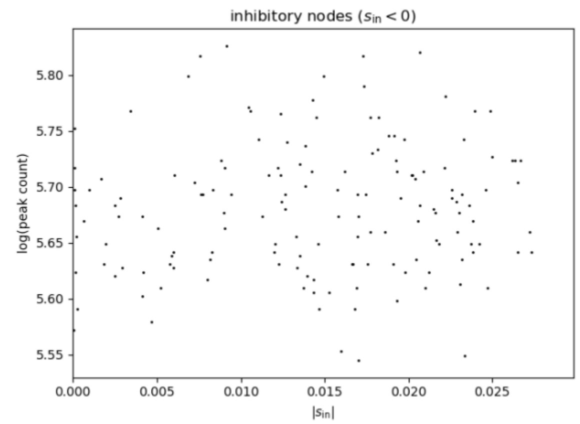


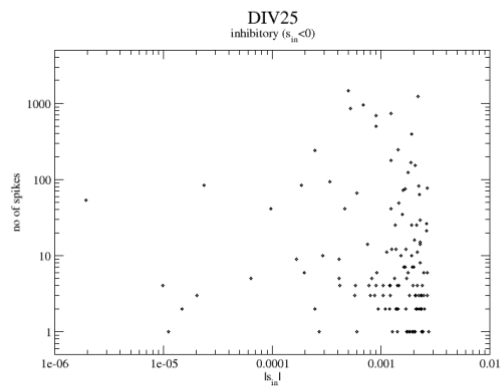
s_in < 0 & s_out > 0



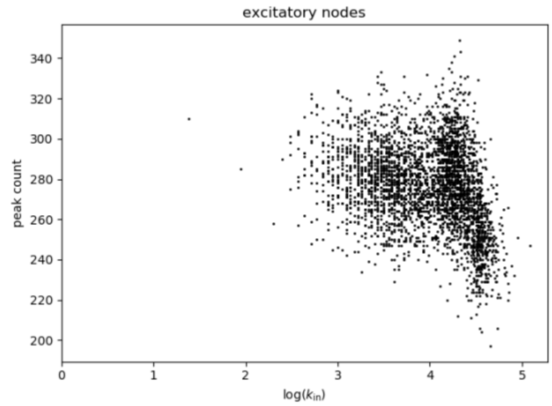
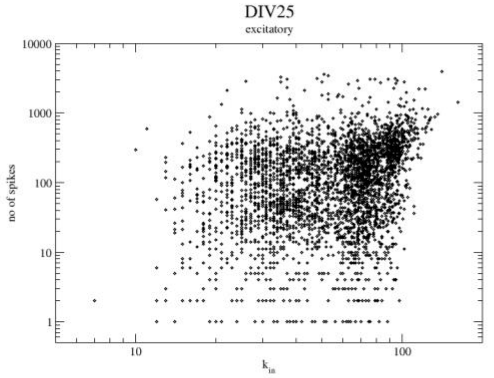
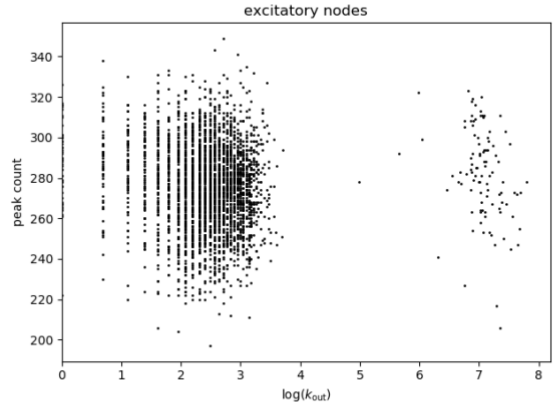
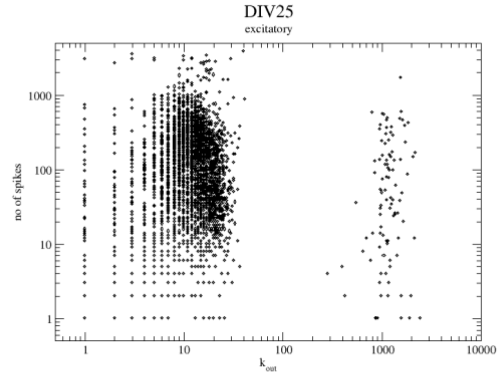
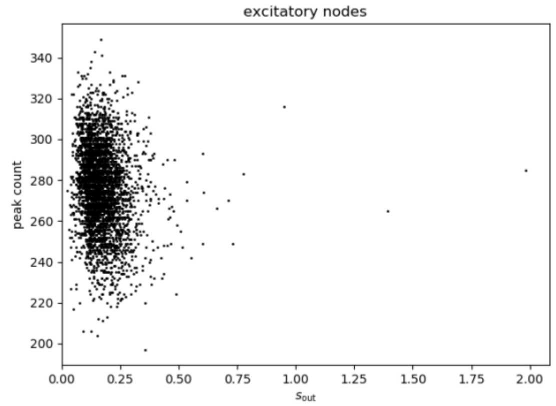
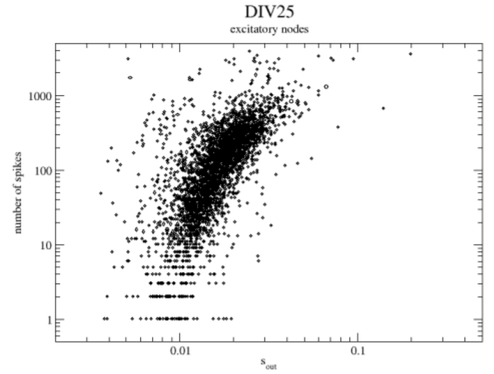
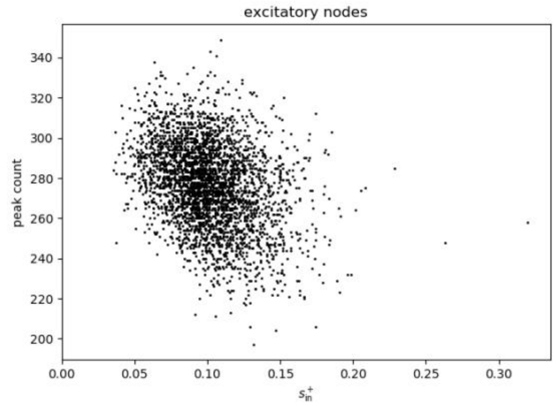
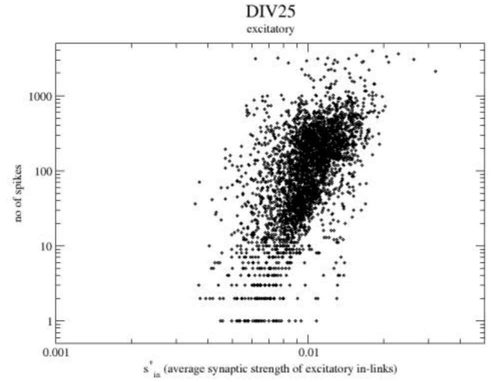


s_in < 0 & s_out < 0



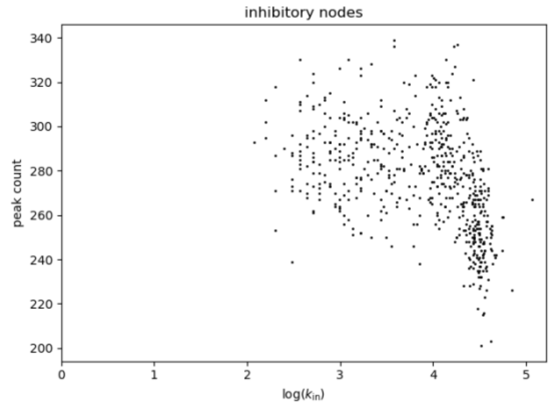


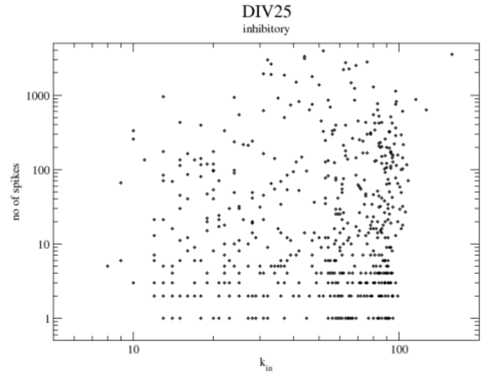
- **Excitatory nodes (log not taken for y-axis)**

peak count vs log k_in	
	
peak count vs log k_out	
	
peak count vs s_out	
	
peak count vs s^+_in	
	

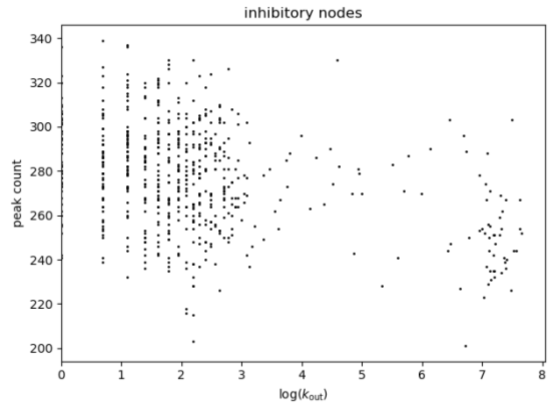
- Inhibitory nodes (log not taken for y-axis)

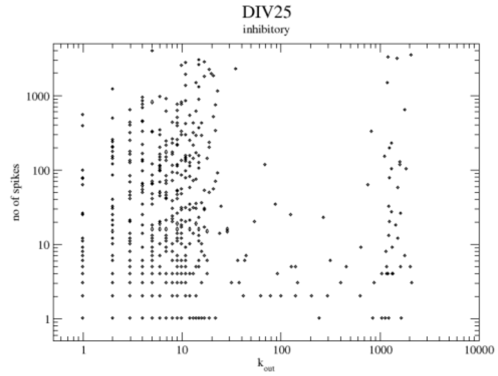
peak count vs log k_in



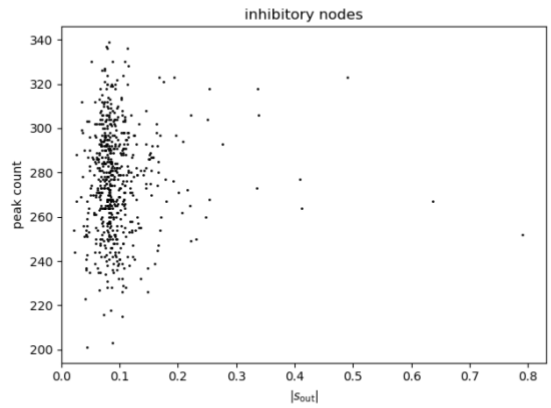


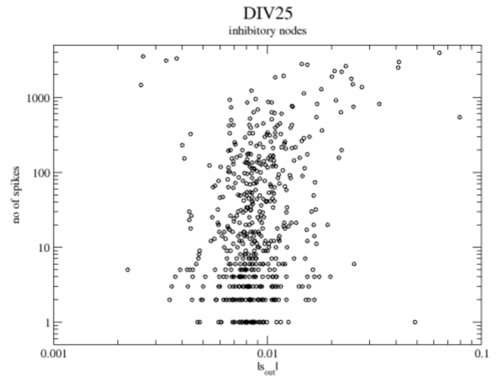
peak count vs log k_out



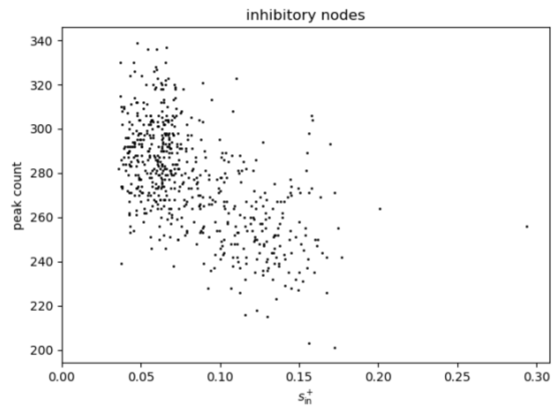


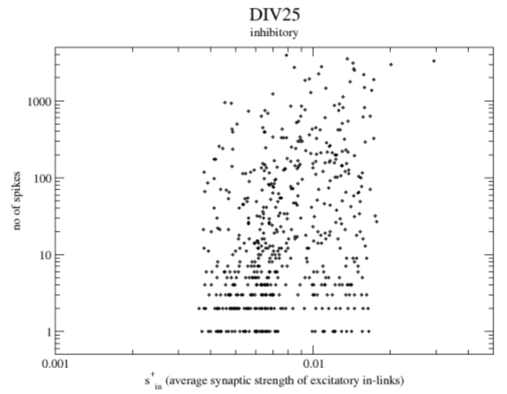
peak count vs |s_out|





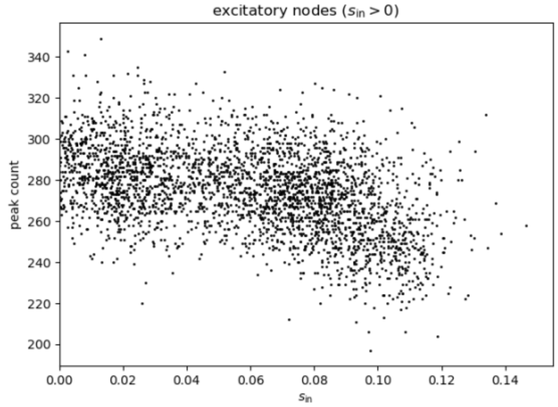
peak count vs s^+_in

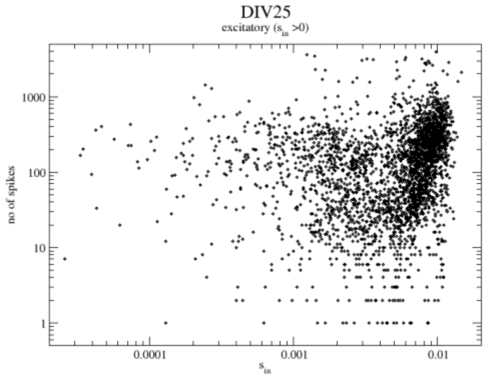




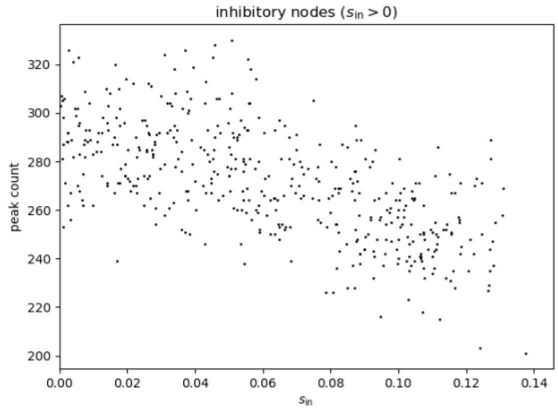
- **Mixed** (x-axis: s_in) (log not taken for y-axis)

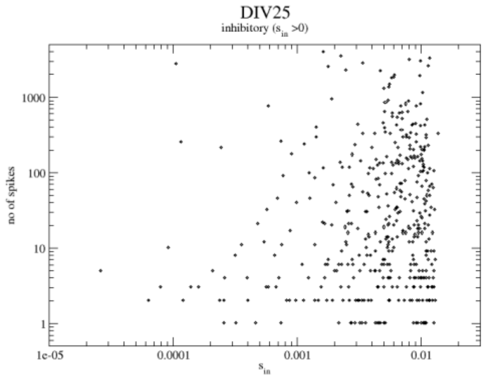
s_in > 0 & s_out > 0



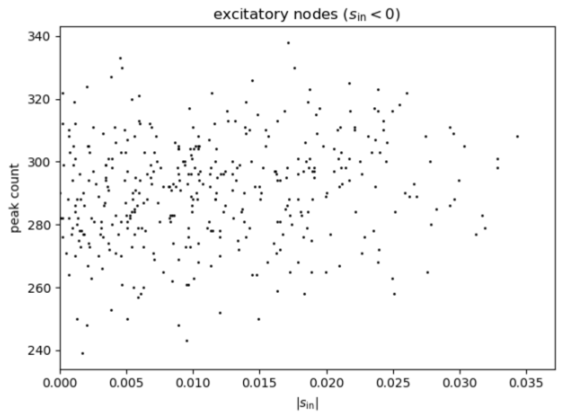


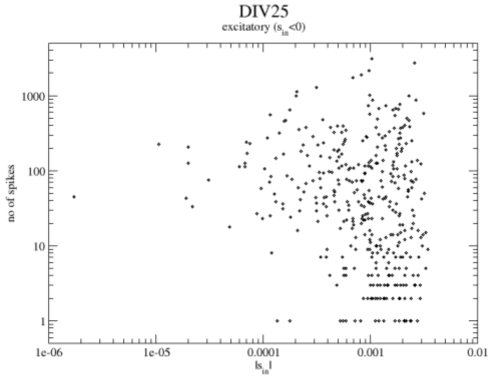
s_in > 0 & s_out < 0



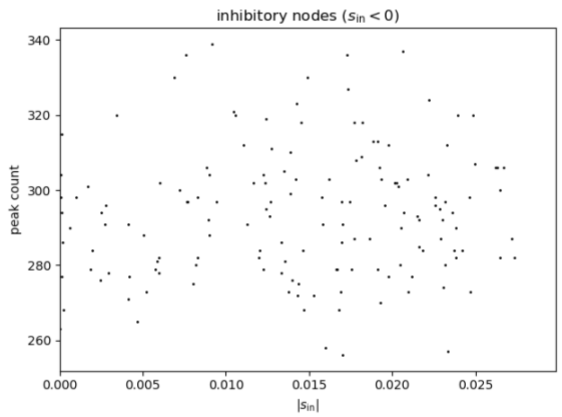


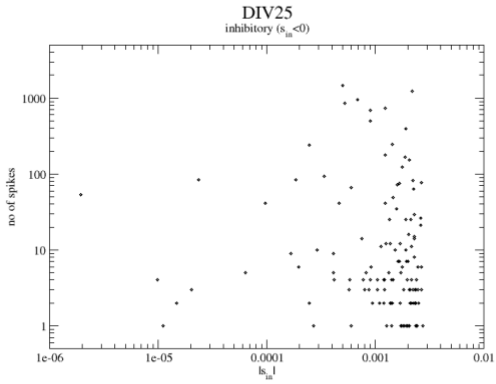
s_in < 0 & s_out > 0





s_in < 0 & s_out < 0



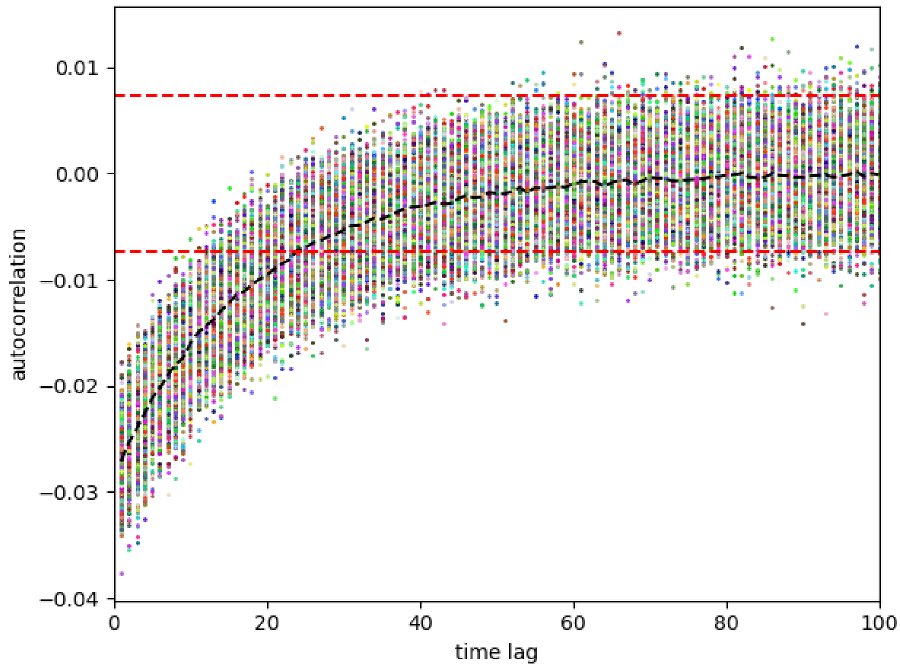


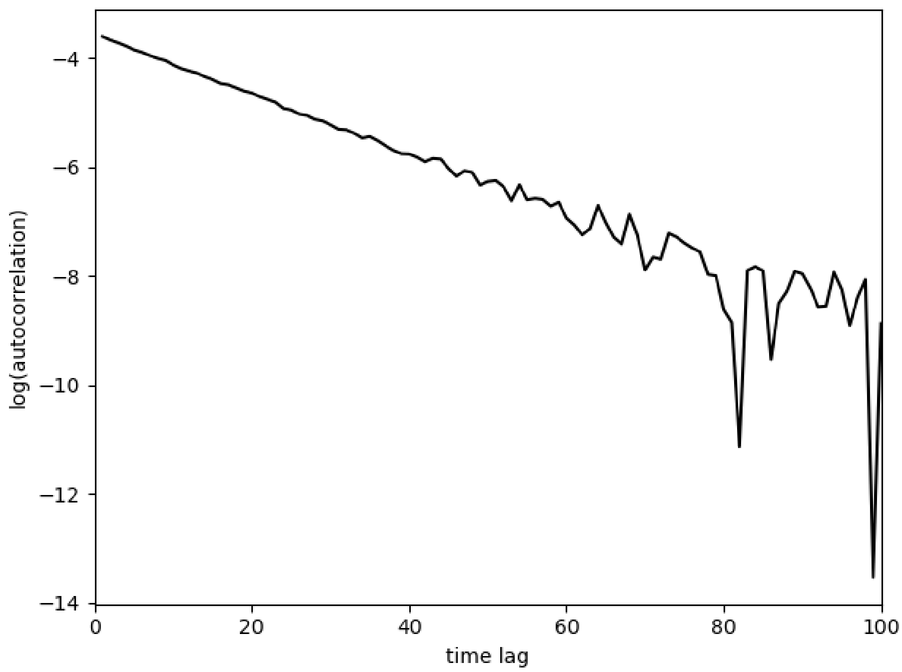
Autocorrelation analysis

- **This disproves that spiking are due entirely to white noise**

$(r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5)$

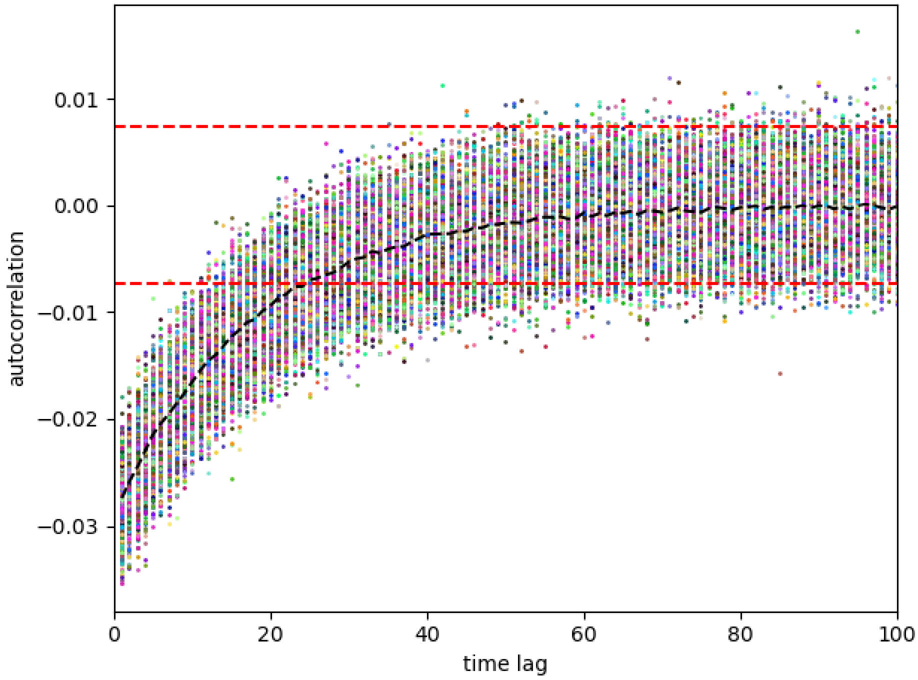
- **Autocorrelation of dx_i (node 0 to 500)**
  - Statistically significant but **not strong**
  - **Red dotted lines ($\pm 2.33/\sqrt{T}$)**: 98% likelihood bound (for a pure white noise process)
  - **Black dotted line**: mean of autocorrelations over all nodes
  - Autocorrelation < 0 $\because x(1-x)$ term makes time series mean-reverting
  - Autocorrelation $\sim 10^{-2}$, only a small multiple larger than the likelihood bound → autocorrelation not strong
  - Autocorrelation obeys an exponential decay → ARMA process, makes sense $\because dx_i$ is a linear combination of node values

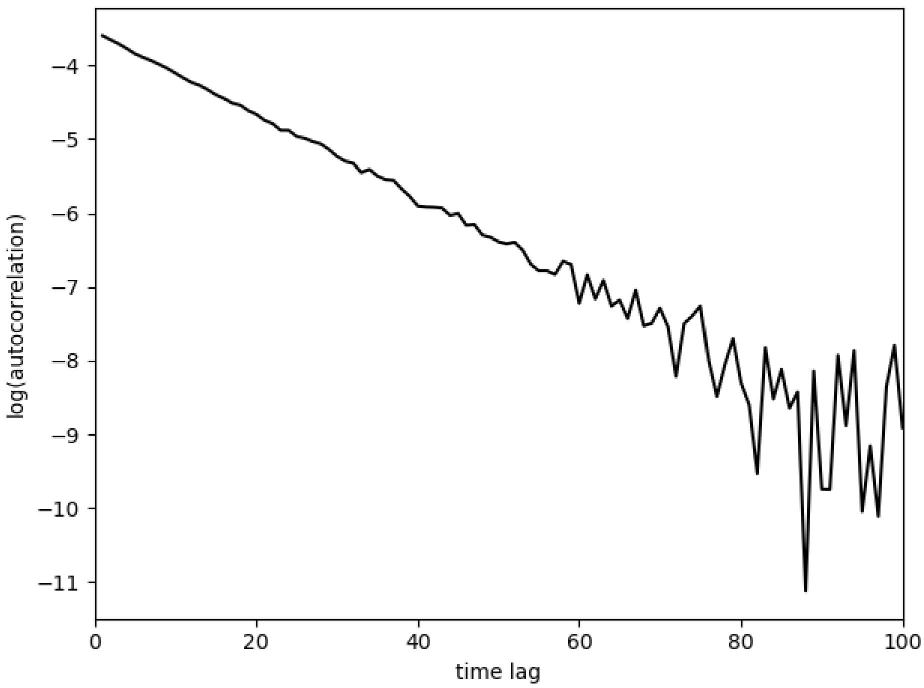




$(r_0 = 100, g_{ij} \times 10, \sigma_i = 0.5)$

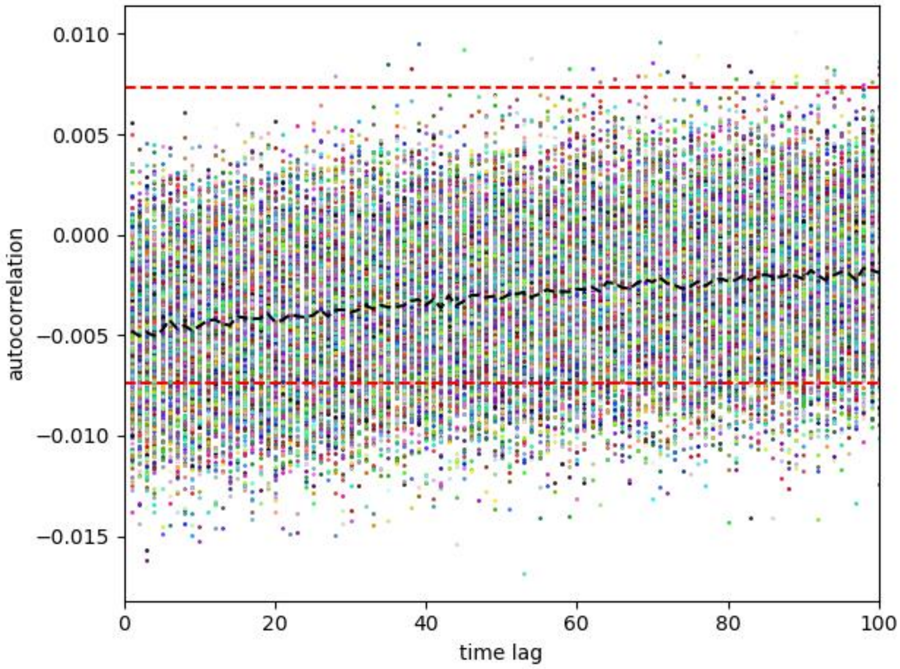
- **Autocorrelation of dx_i (node 0 to 500)**
  - Here smaller $\sigma_i = 0.5$ is used
  - Autocorrelation plot has similar shape & values compared to $\sigma_i = 1.5$ case
  - **Original guess**: large noise "fuzzes out" dependence of time series (and peaks) on g_{ij} hence making autocorrelation small; but it turns out noise doesn't affect autocorrelation at all (maybe because of the averaging when computing autocorrelation)
  - **Guess**: autocorrelation is small because g_{ij} is small

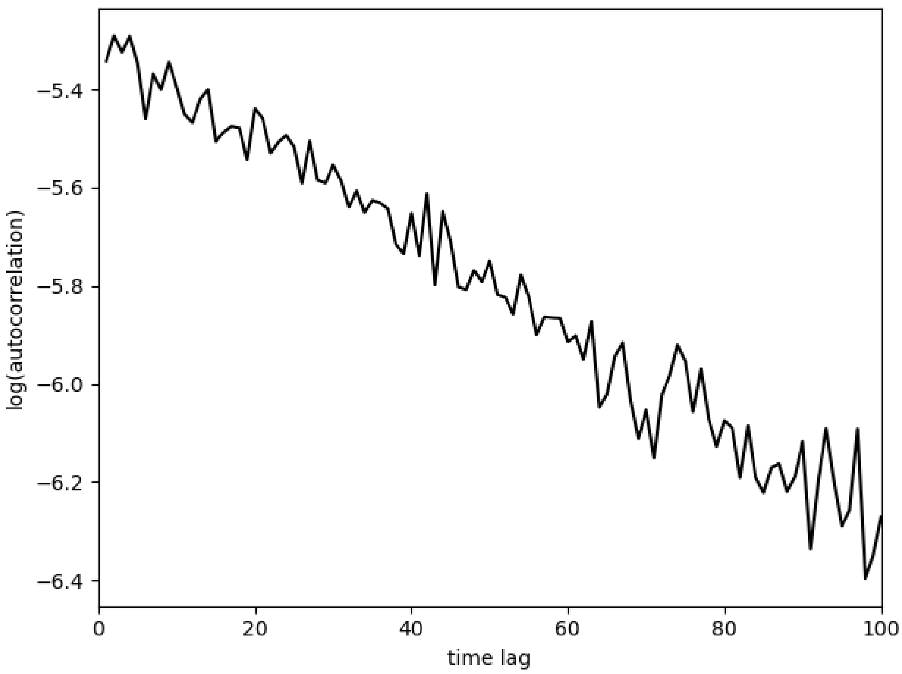




$(r_0 = 10, g_{ij} \times 10, \sigma_i = 0.25)$

- **Autocorrelation of dx_i (node 0 to 500)**
  - Here smaller $r_0 = 10$ is used
  - Autocorrelation becomes an order smaller (e.g. $-0.03 \to -0.005$)

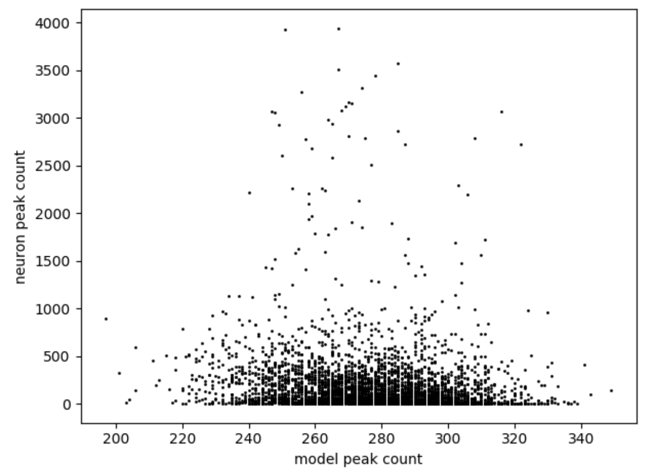
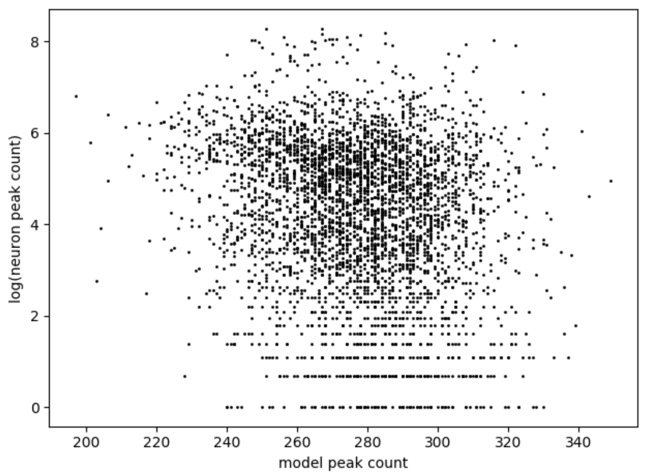




Spike analysis (Gaussian reference network)

- **Model params: diffusive, $\left(r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5\right)$**
- **Draw comparison btw. simulation data & reference data (reference Gaussian network)**

- **Model peaks vs experimentally measured peaks**

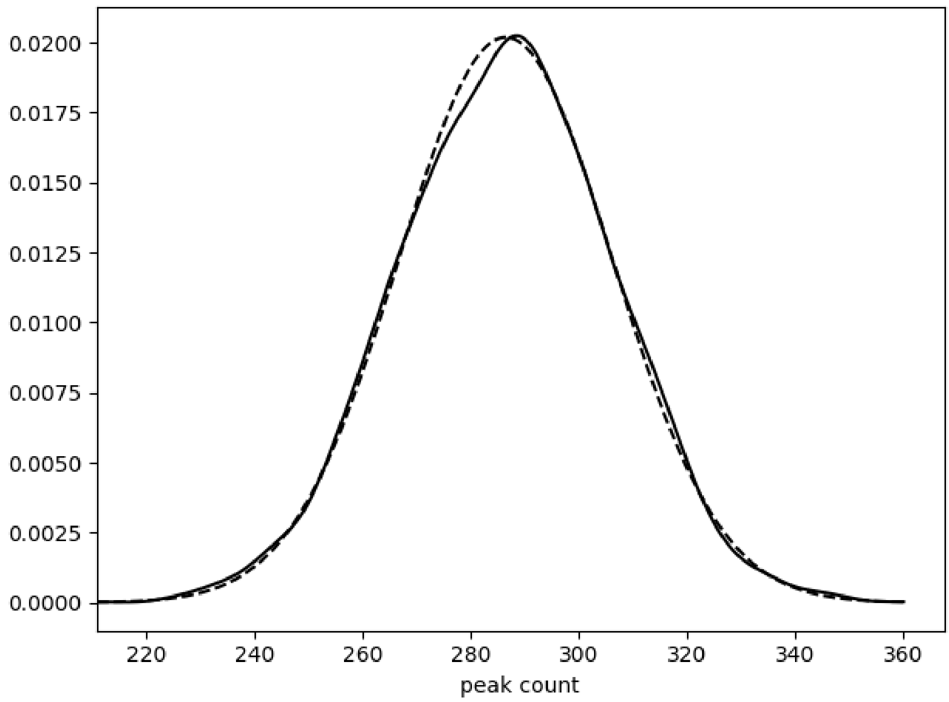
Linear y	Log-scale y
	

- **With-noise case: params**
  $(r_0, g_{ij} \times \text{multiplier}, \sigma_i) = (r_0 = 100, g_{ij} \times 10, \sigma_i = 1.5)$
- **Question to address**: do heavy-tailed strengths explain any behaviour of peaks?

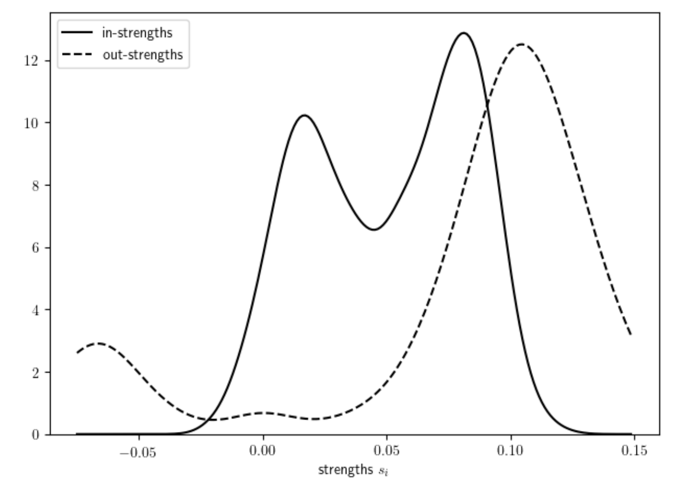
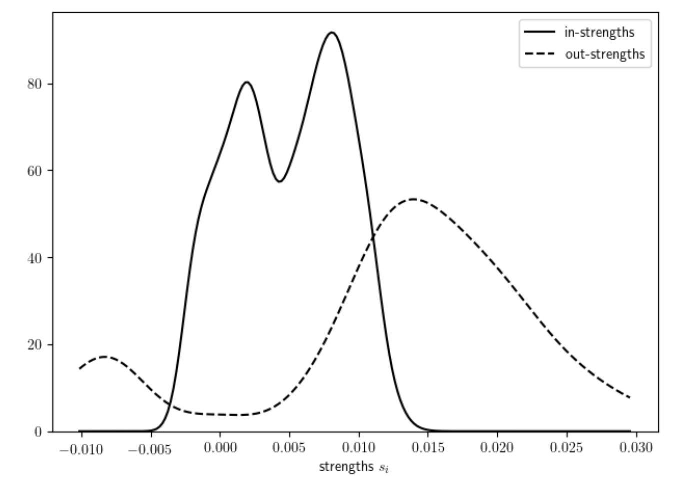
- **Reference network with Gaussian couplings**
  (1) Calculate the mean and standard deviation of all positive gij (mu_1, sigma_1) and all negative gij (mu_2, sigma_2).
  (2) For every link with a positive gij, replace it from a value randomly picked from a Gaussian distribution of mean mu_1 and standard deviation sigma_1 but with the additional requirement that it has to be positive.
  (3) Similarly, for every link with a negative gij, replace it from a value randomly picked from a Gaussian distribution of mean_2 and standard deviation sigma_2 but with the additional requirement that it has to be negative.

- **(Continuing from last time) (Numerical) spiking detection**
  Criteria:
  (a) Peak value > steady value + ($\alpha = 2$ or 2.5) * s.d. of fluctuations (of all nodes) over $[0, T]$, s.d. of fluctuations ~ 0.1
  (b) Time steps btw. peak values > $d = 20$ (avoid multiple-counting for very close peaks)

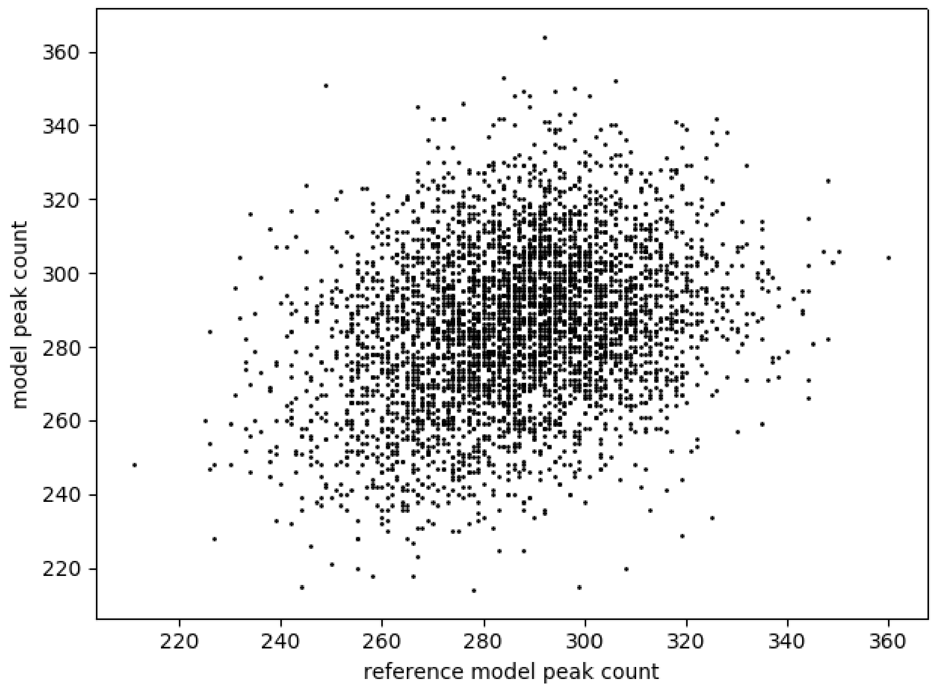
- **Distribution of number of peaks (near normal)**



- **s_in s_out distribution**

Reference network	Original network
	

- **How peaks in reference network correlate with original network (near linear, similar range)**

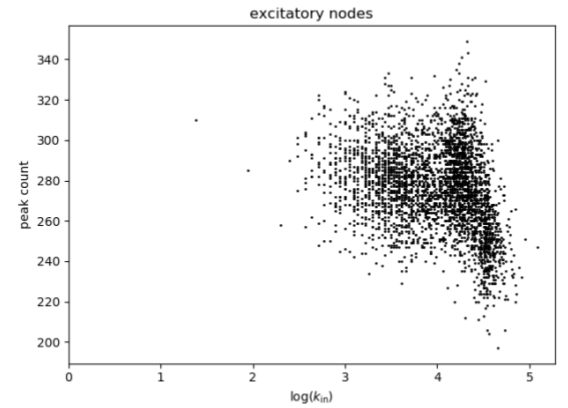


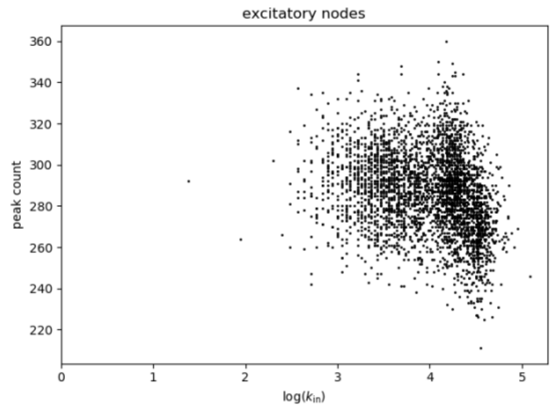
- Note for the following plots:
  (a) **Peak criteria: $\alpha = 2, d = 20$**
  (b) Number of peaks range from 200 to 360 **(1e5 steps)**
  (c) **Left** column: original network
  Right column: reference network

- **Excitatory nodes** **(log not taken for y-axis)**

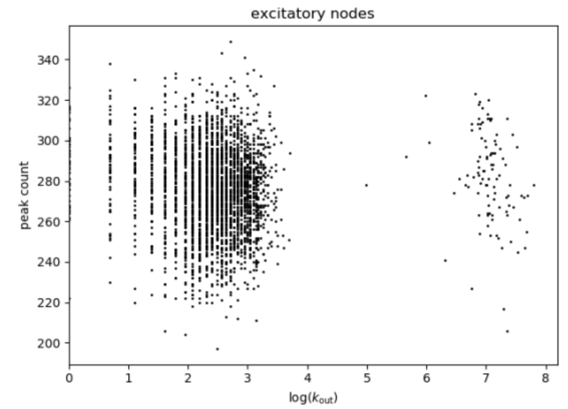
original network	reference network

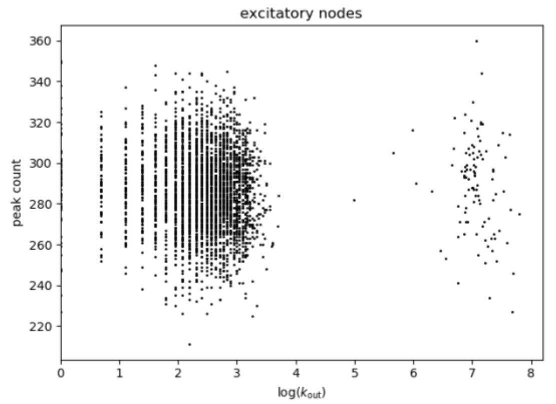
peak count vs log k_in



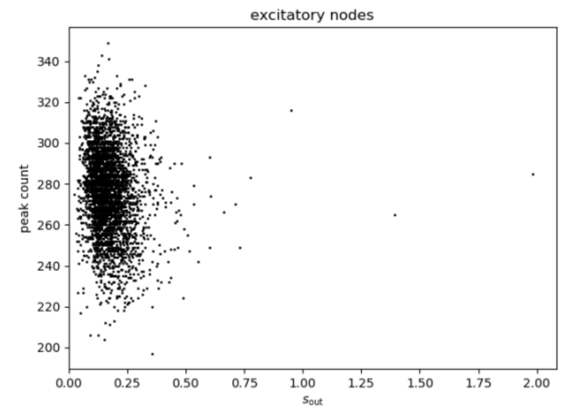


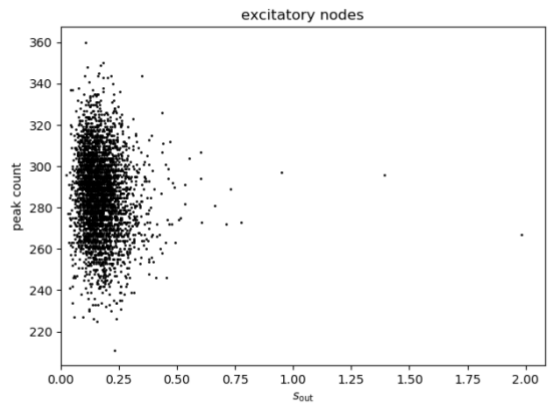
peak count vs log k_out



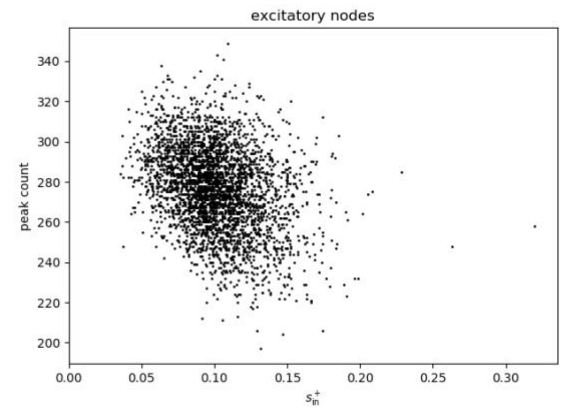


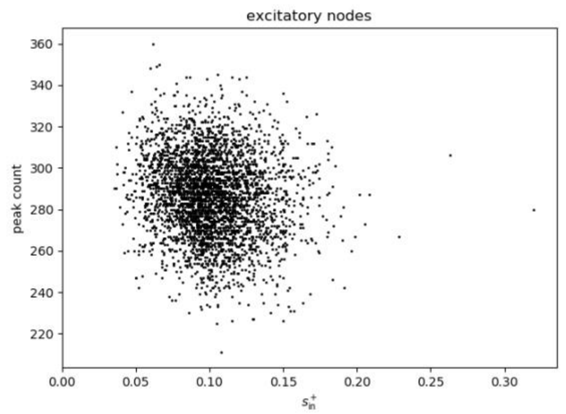
peak count vs s_out



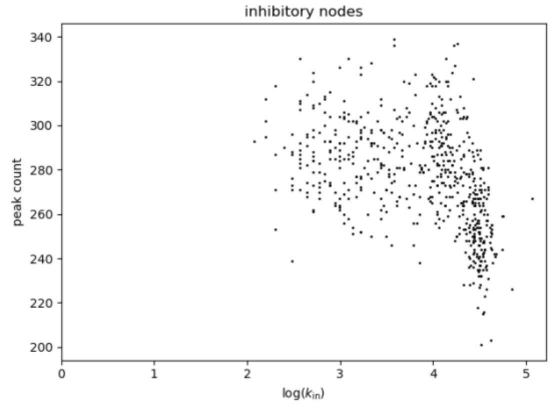
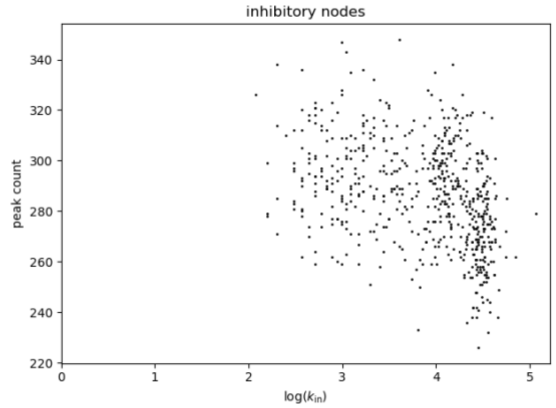
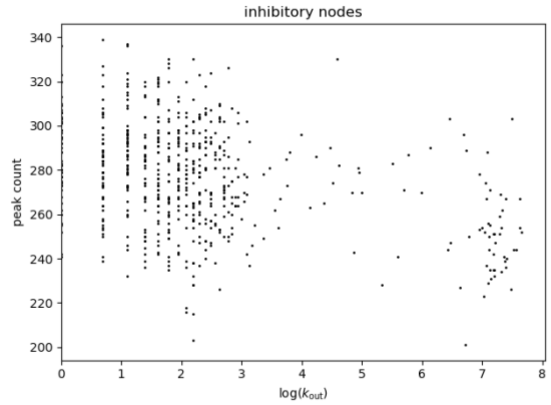
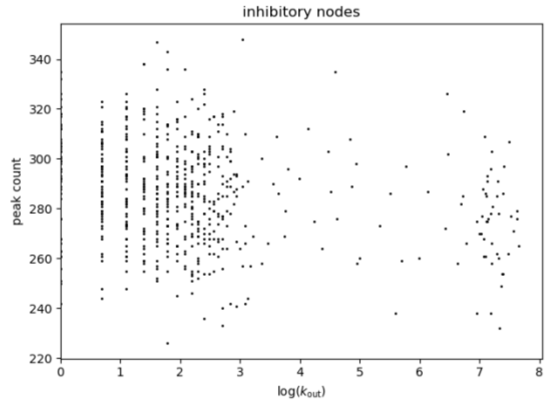
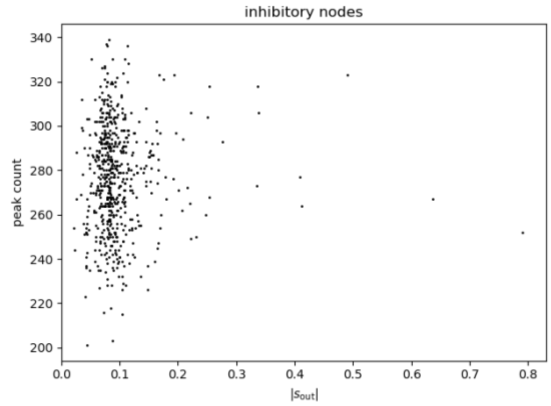
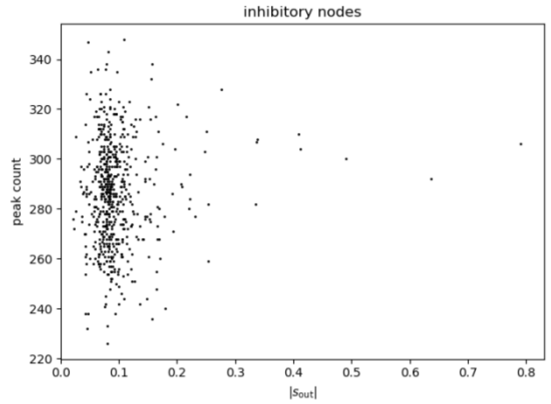
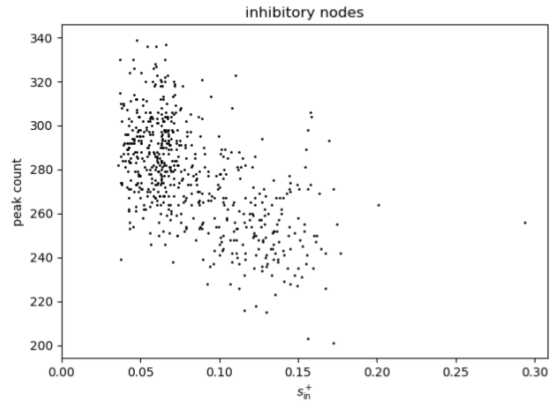
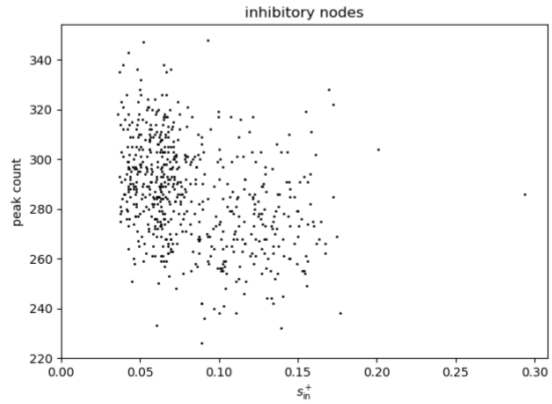


peak count vs s^+_in

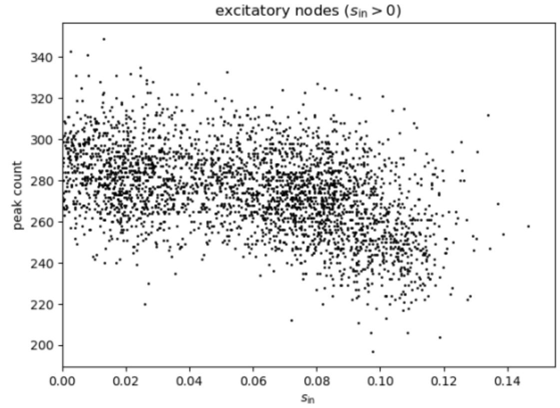
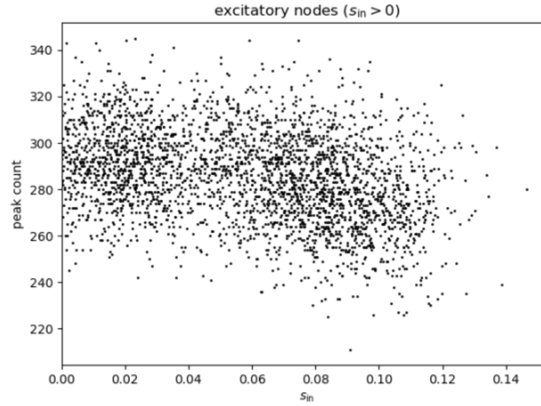
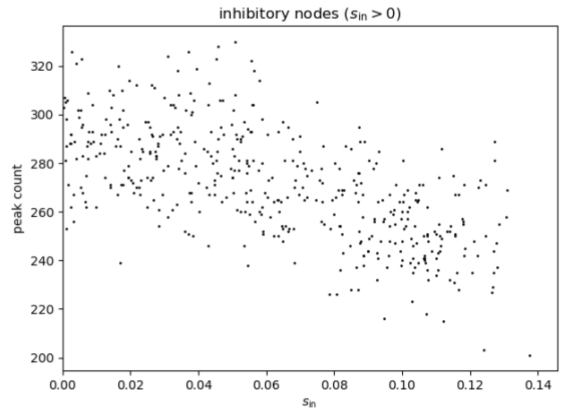
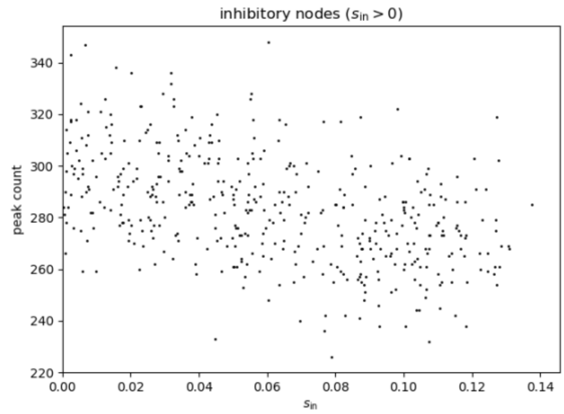
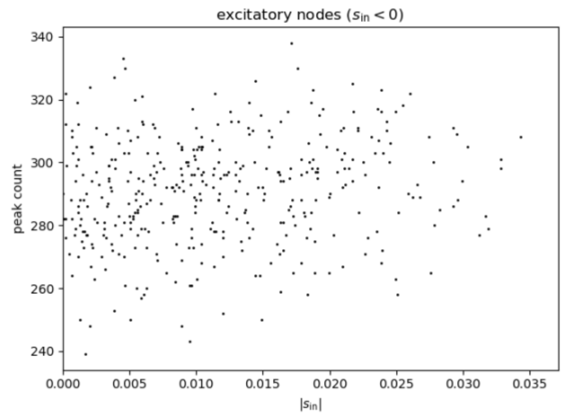
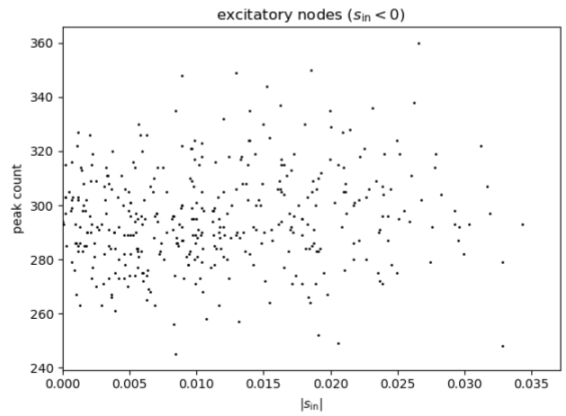
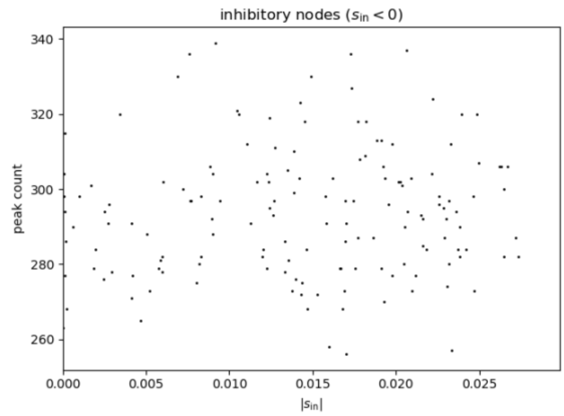
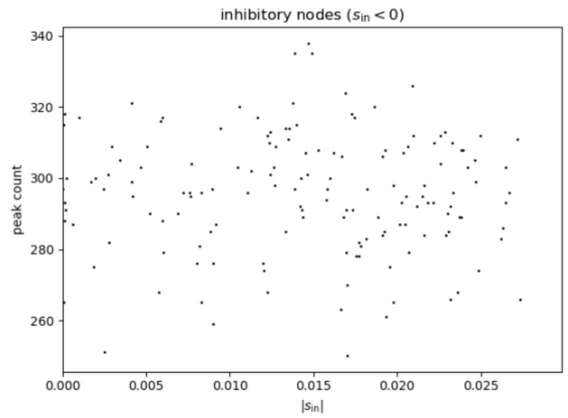




- **Inhibitory nodes** **(log not taken for y-axis)**

original network	reference network
peak count vs log k_in	
	
peak count vs log k_out	
	
peak count vs |s_out|	
	
peak count vs s^+_in	
	

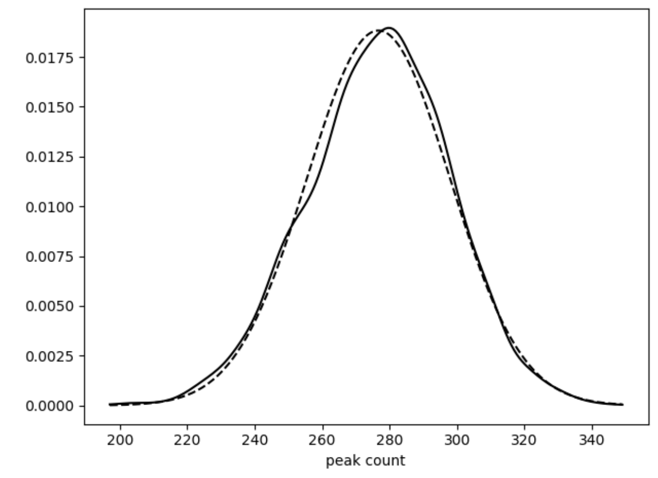
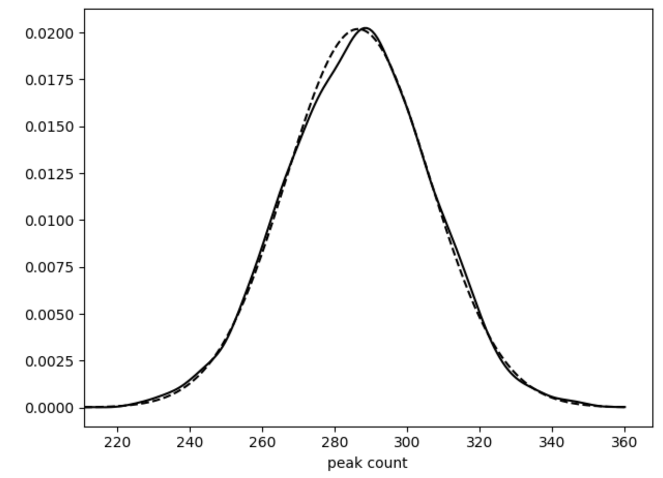
- **Mixed** (x-axis: s_in) **(log not taken for y-axis)**

original network	reference network
s_in > 0 & s_out > 0	
	
s_in > 0 & s_out < 0	
	
s_in < 0 & s_out > 0	
	
s_in < 0 & s_out < 0	
	

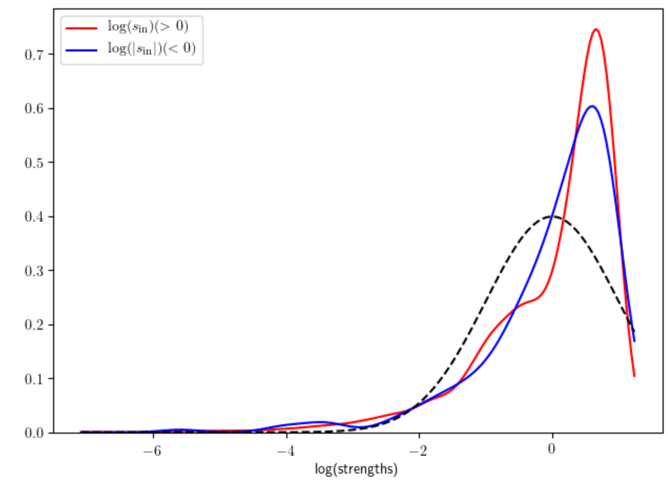
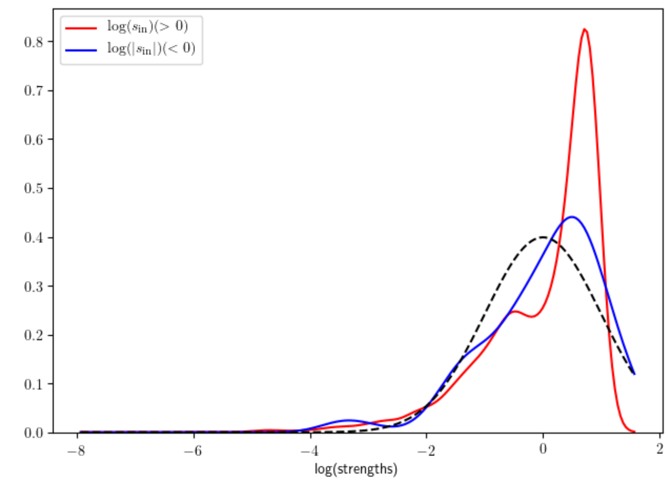
- **Guess**
  (1) Noise is too large ⇒ network structure (gij) has little to do with the noise & peaks are completely due to random Gaussian white noise, which explains why peak counts are Gaussian
  (2) But choosing smaller noise makes peaks less well-defined

Reference network analysis

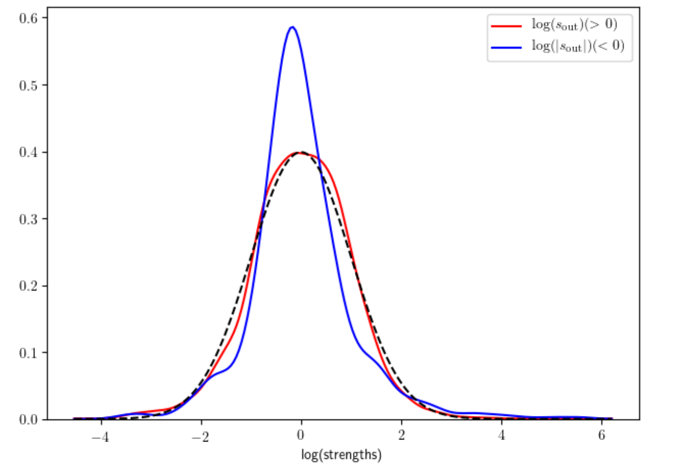
- **Distribution of number of peaks (both near normal)**

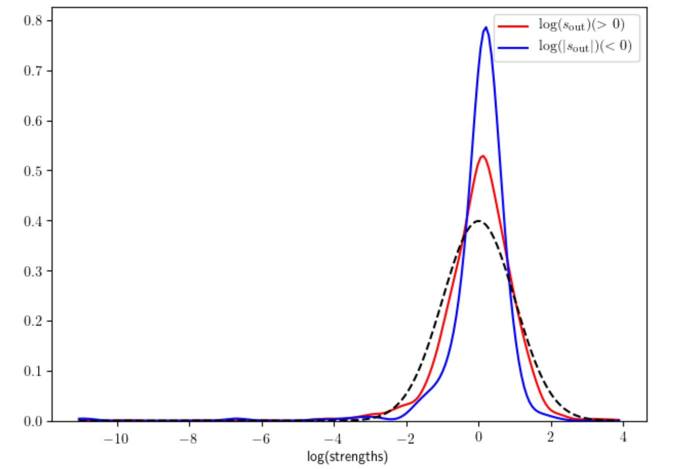
original network	**reference network**
	

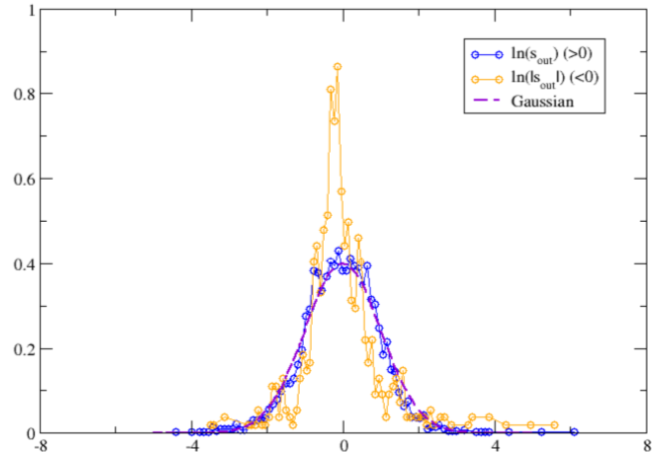
- **Standardized log(in-strength) distribution**

original network	**reference network**
	

- **Standardized log(out-strength) distribution**

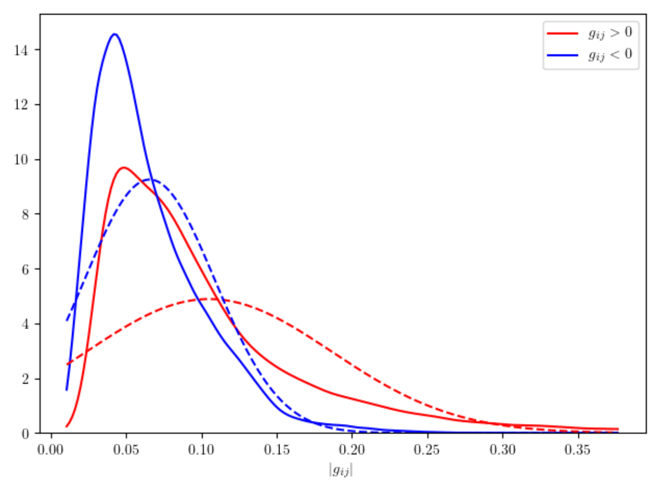
<table>
<tr><th>original network</th><th>reference network</th></tr>
<tr><td>


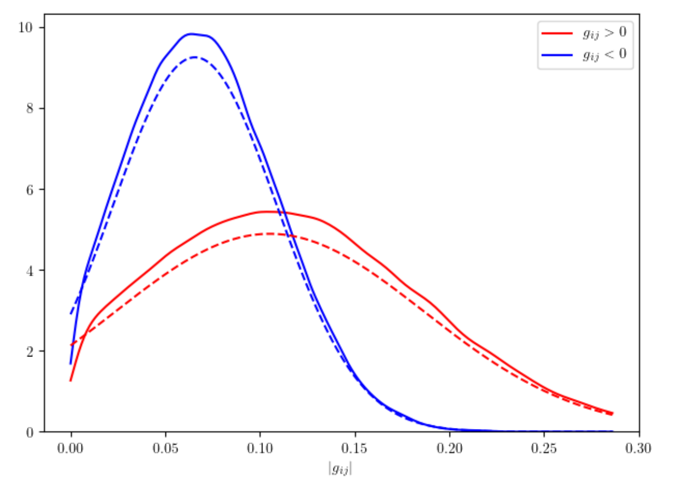
</td><td>


</td></tr>
<tr><td>


</td><td></td></tr>
</table>

- **|g_ij| distribution (2 cases: g_ij > 0 and g_ij < 0)**
  Classified as “heavy-tailed” if mean >> median

<table>
<tr><th>original network</th><th>reference network</th></tr>
<tr><td>


- **Dashed lines**: fitted Gaussian with same mean and s.d.
- **(>0) Mean = 0.1052; Median = 0.0822**
- **(<0) Mean = 0.0657; Median = 0.0560**
</td><td>


- **Dashed lines**: fitted Gaussian from original network (from which reference g_ij is generated)
- **(>0) Mean = 0.1210; Median = 0.1157**
- **(<0) Mean = 0.0712; Median = 0.0690**
</td></tr>
</table>