Inverting nonlinear systems with approximately Bernoulli responses

Sean Bittner

April 4, 2019

# 1 Introduction

We want to have approximately Bernoulli responses of the SC network in the various conditions. Before modeling the full task, we should figure out the appropriate way to invert a nonlinear dynamical system with approx. Bernoulli responses in a single condition. The key question is how to model this "emergent property" as a moment constraint.

# 2 SC Model

There are four total units: two in each hemisphere corresponding to the PRO/CONTRA and ANTI/IPSI populations. Each unit had an external ($V_i$) and internal ($U_i$) variable related by

$$V_i(t) = \eta(t)\left(\frac{1}{2}\tanh\left(\frac{U_i(t) - \theta}{\beta}\right) + \frac{1}{2}\right) \tag{1}$$

$\theta = 0.05$ and $\beta = 0.5$ control the position and shape of the nonlinearity, repsectively, and $\eta(t)$ is the optogenetic inactivation function.

We can order the elements of $V_i$ and $U_i$ into vectors $v$ and $u$ with elements

$$v = \begin{bmatrix} V_{LP} \\ V_{LA} \\ V_{RA} \\ V_{RP} \end{bmatrix} \qquad\qquad u = \begin{bmatrix} U_{LP} \\ U_{LA} \\ U_{RA} \\ U_{RP} \end{bmatrix} \tag{2}$$

The internal variables follow dynamics:

$$\tau \frac{\partial u}{\partial t} = -u + Wv + I + \sigma \partial W \tag{3}$$

with time constant $\tau = 0.09s$ and gaussian noise $\sigma \partial W$ controlled by the magnitude of $\sigma$. The weight matrix has 8 parameters $sW_P$, $sW_A$, $vW_{PA}$, $vW_{AP}$, $hW_P$, $hW_A$, $dW_{PA}$, and $dW_{AP}$, related to the depiction in Fig. 2:

**Full Model**

$$W = \begin{bmatrix} sW_P & vW_{PA} & dW_{PA} & hW_P \\ vW_{AP} & sW_A & hW_A & dW_{AP} \\ dW_{AP} & hW_P & sW_A & vW_{AP} \\ hW_A & dW_{PA} & vW_{PA} & sW_P \end{bmatrix} \tag{4}$$

The input is a sum of five task-relalated inputs.

$$I = I_{\text{constant}} + I_{\text{pro-bias}} + I_{\text{rule}} + I_{\text{choice-period}} + I_{\text{light}} \tag{5}$$

We'll also consider a 4-parameter reduced model:

**Reduced Model**

$$W = \begin{bmatrix} sW & vW & dW & hW \\ vW & sW & hW & dW \\ dW & hW & sW & vW \\ hW & dW & vW & sW \end{bmatrix} \tag{6}$$

# 3 Setting up the DSN behavior constraints

Let's say that we want to learn the parameters that produce a Bernoulli rate of $p_{LP}$ in the Left, Pro condition. We'll let $\hat{p}_i$ be the empirical average steady state (ss) response (final $V_{LP}$ at end of task) over M=100 gaussian noise draws for a given dynamical system parameterization $z_i$:

$$\hat{p}_i = E_{\sigma \partial W} \left[ V_{LP,\text{ss}} \mid s = L, c = P, z_i \right] = \frac{1}{M} \sum_{j=1}^{M} V_{LP,\text{ss}}(s = L, c = P, z_i, \sigma \partial W_j) \tag{7}$$

The noise is fixed at $\sigma = 0.3$ (the average of satisfactory parameterizations from Duan et al.). For the 1st constraint, we certainly want the average over DSN samples to be $p_{LP}$:

$$E_{z \sim q_\phi} \left[ E_{\sigma \partial W} \left[ V_{LP,\text{ss}} \mid s = L, c = P, z \right] \right] = E_{z \sim q_\phi} \left[ \hat{p} \right] = p_{LP} \tag{8}$$

We can then ask that the variance of the steady state responses across gaussian draws, is the Bernoulli variance for the empirical rate $\hat{p}$.

$$Var_{\sigma \partial W} \left[ V_{LP,\text{ss}} \mid s = L, c = P,_i z \right] = \hat{p}(1 - \hat{p}) \tag{9}$$

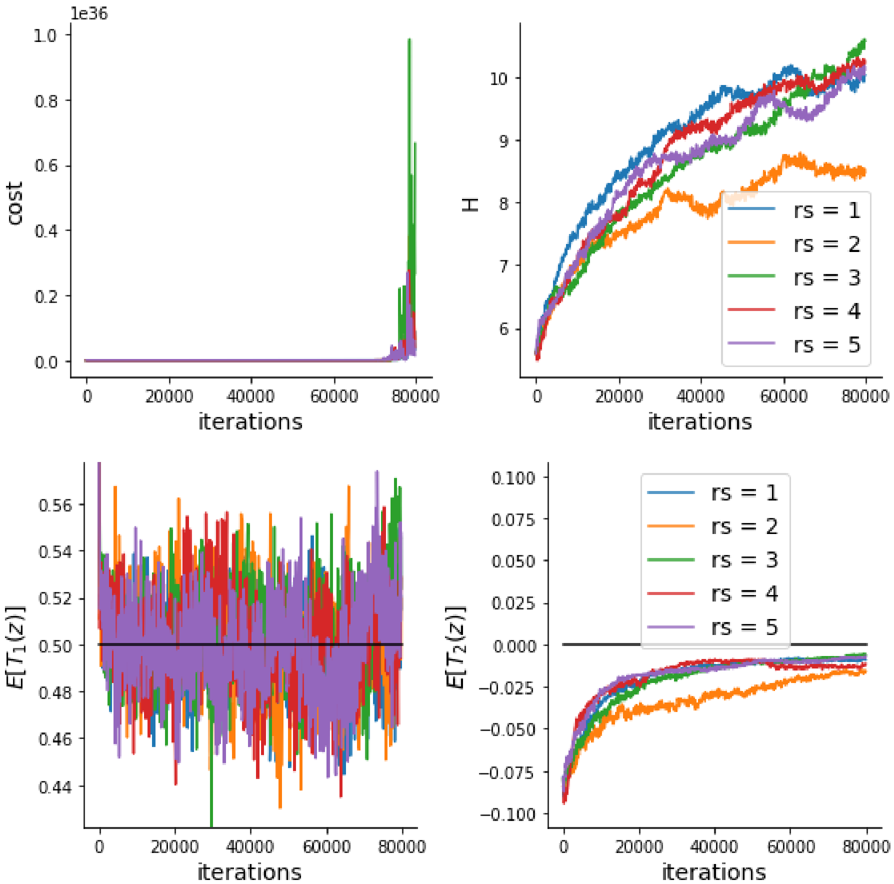
With DSNs, we enforce constraints in expectation over DSN samples, so we can force Bernoulli responses with this 2nd constraint:

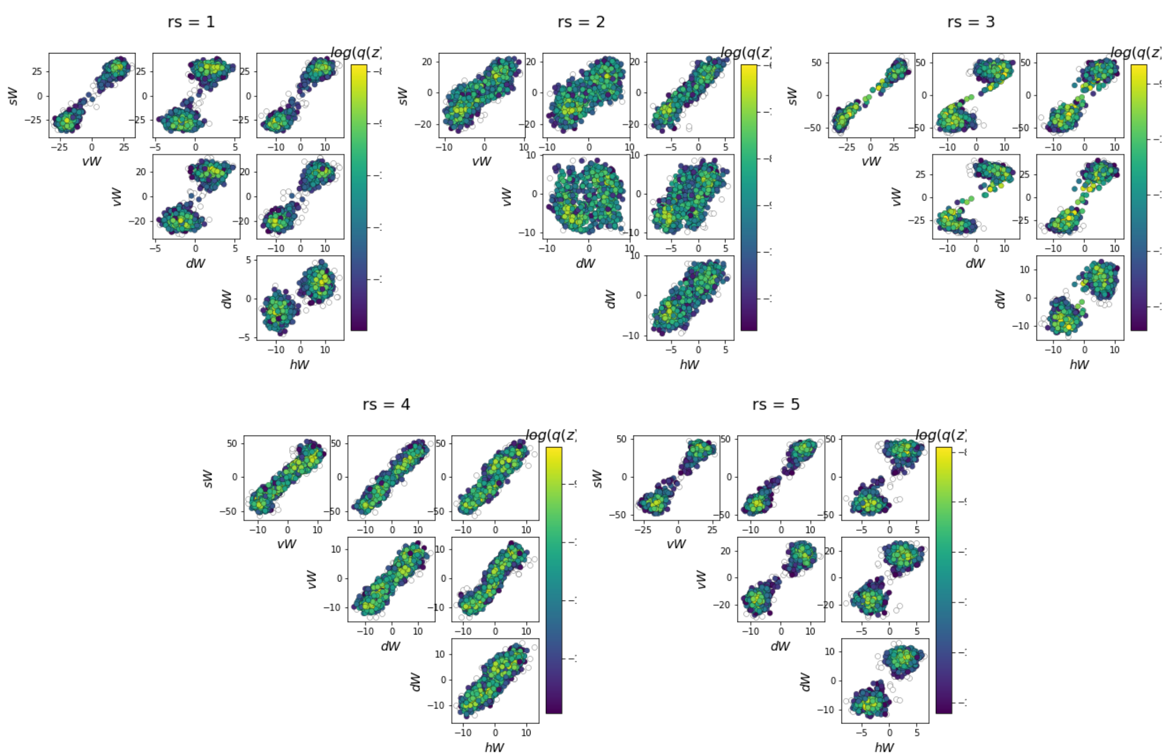
$$E_{z \sim q\phi} \left[ Var_{\sigma \partial W} \left[ V_{LP,\text{ss}} \mid s = L, c = P,_i z \right] - \hat{p}(1 - \hat{p}) \right] = 0 \tag{10}$$

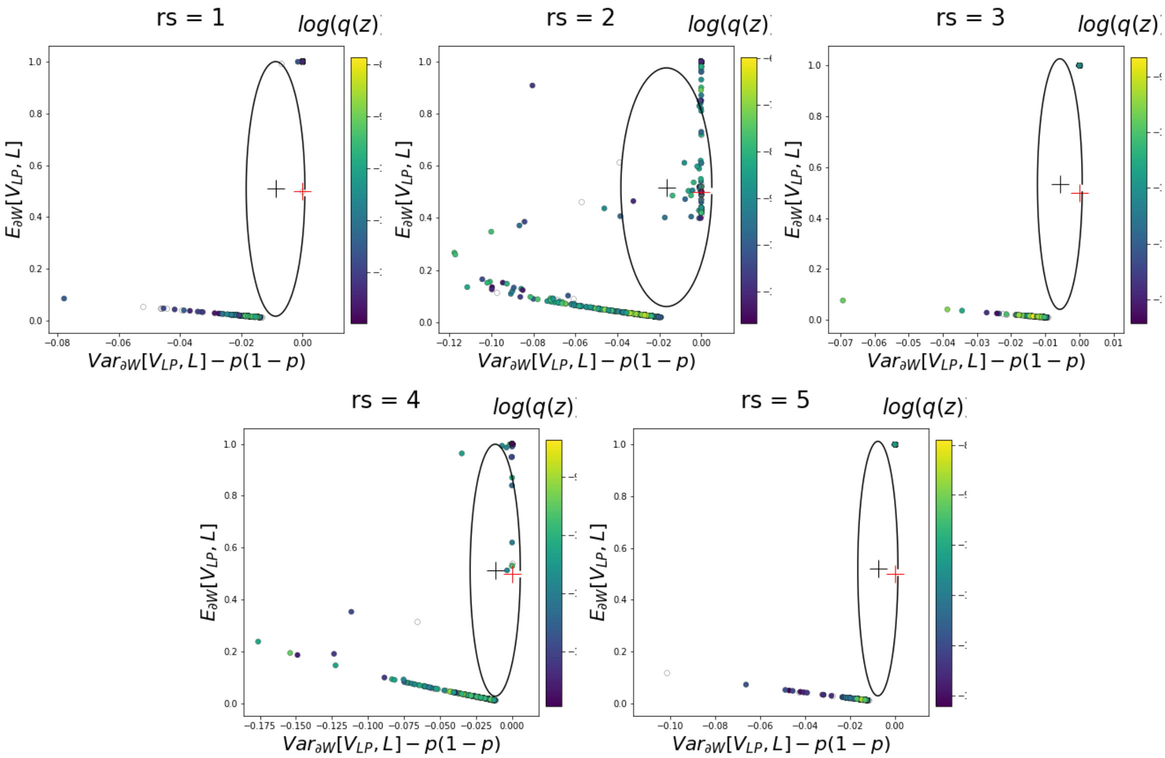
Since the maximum variance of a random variable bounded from 0 to 1 is the Bernoulli variance ($\hat{p}(1 - \hat{p})$), in principal, we do not need to control the second moment (over DSN samples) of this test-static (the variance over gaussian draws). In reality, these variables are dynamical system states and can only exponentially decay (or saturate) to 0 (or 1), so the Bernoulli variance constraint can only be undershot. This is important to be mindful of, when thinking about how to enforce Bernoulli responses in this fashion.

# 4 An attempt was made (to sample Bernoulli networks)

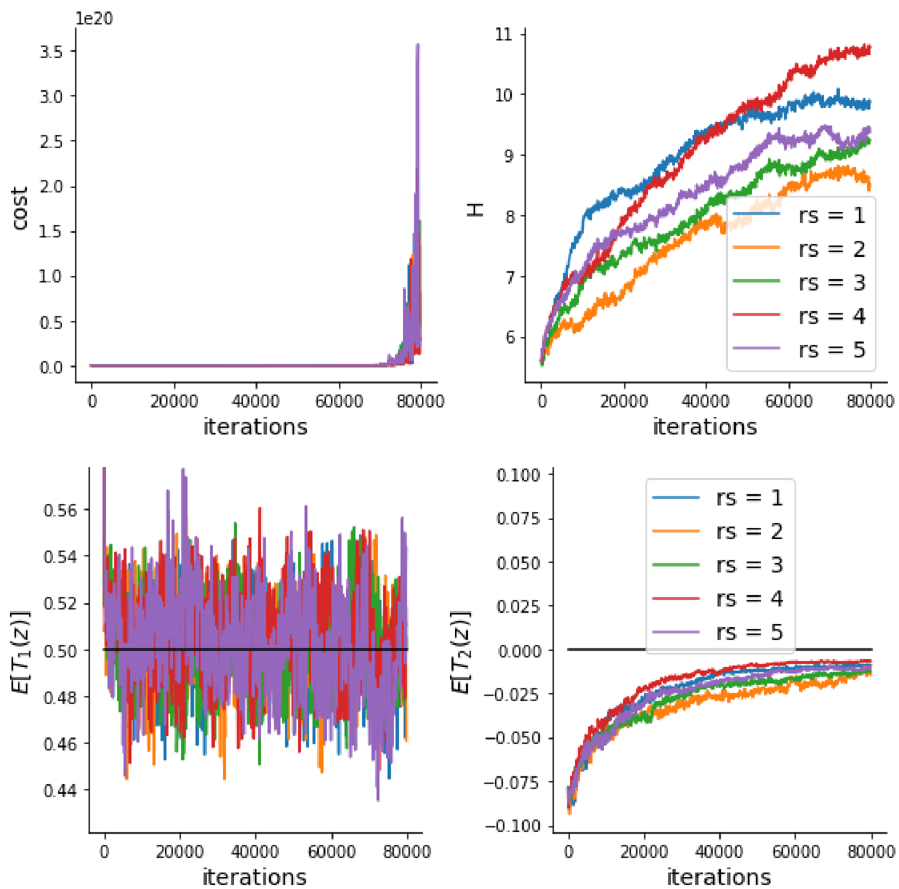
**reduced entropy p = 0.5**


1e36
cost
iterations
H
rs = 1
rs = 2
rs = 3
rs = 4
rs = 5
iterations
$E[T_1(z)]$
iterations
$E[T_2(z)]$
rs = 1
rs = 2
rs = 3
rs = 4
rs = 5
iterations

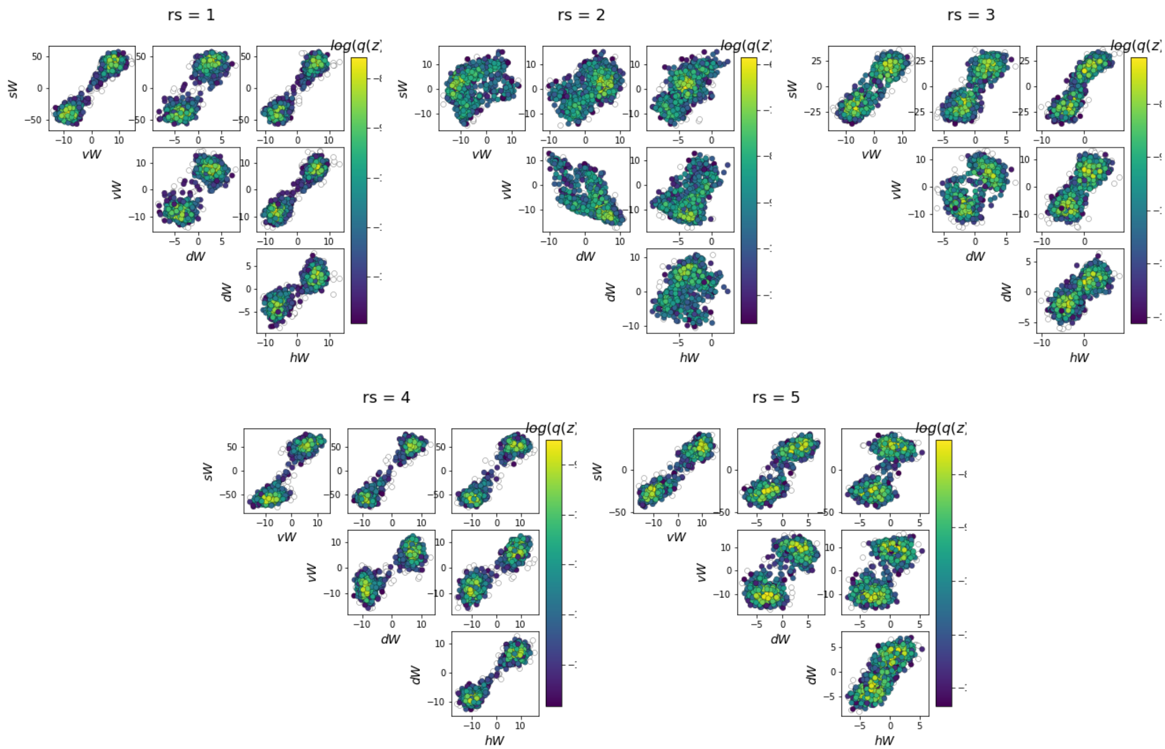

rs = 1
rs = 2
rs = 3
rs = 4
rs = 5
sW
vW
dW
hW
log(q(z)

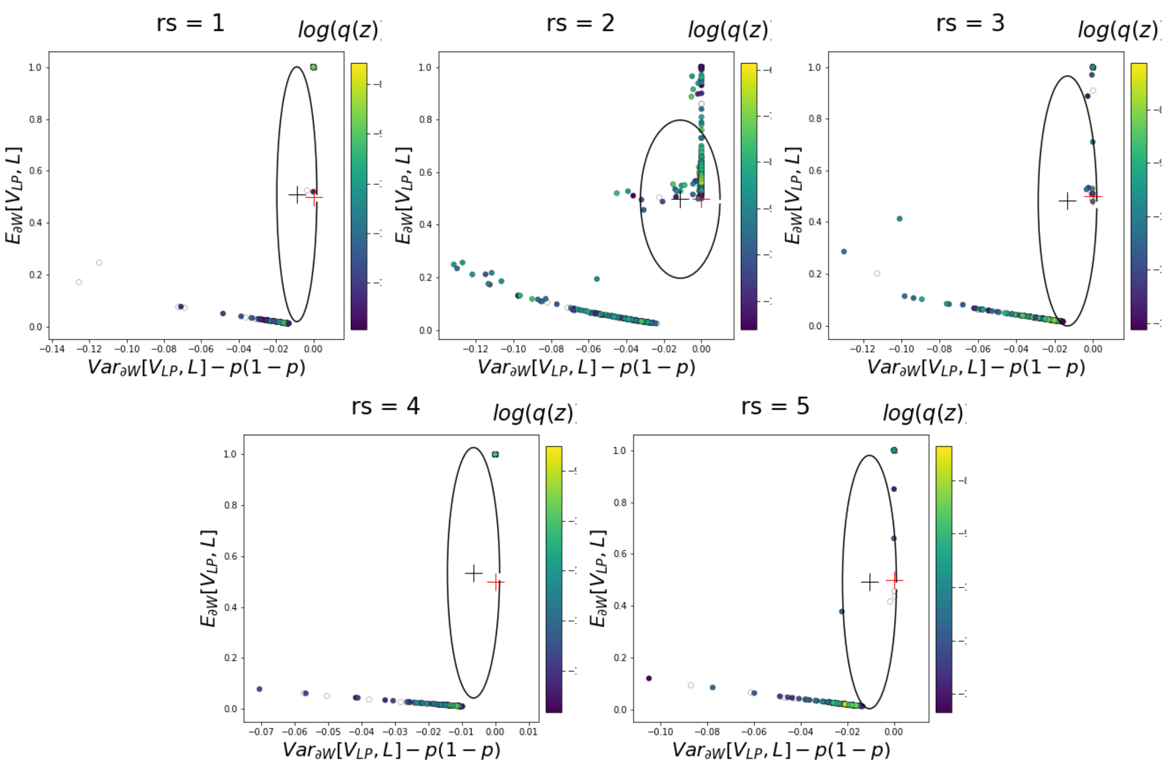



**reduced NO entropy p = 0.5**

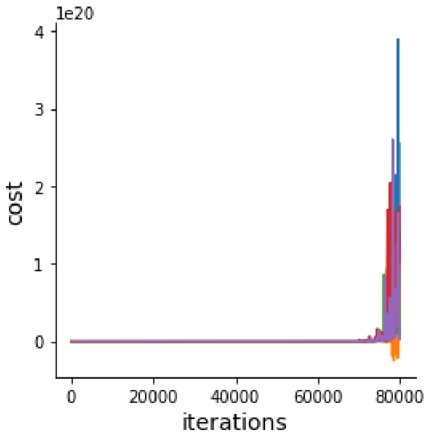


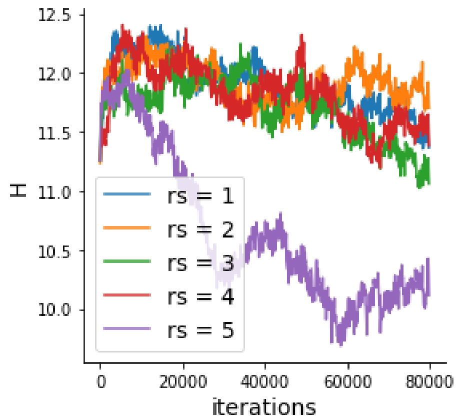
Sean Bittner
srb2201@columbia.edu

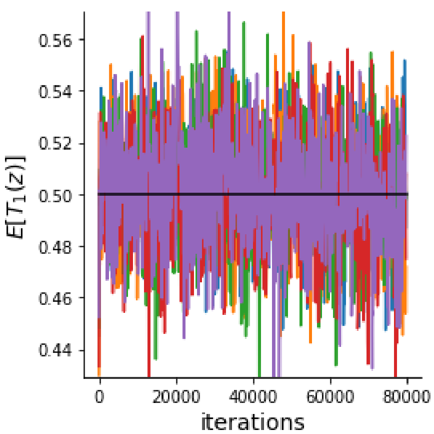


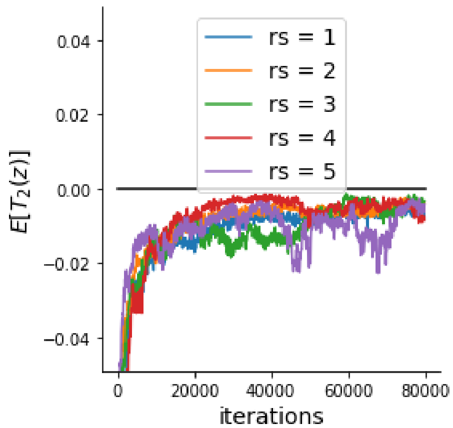



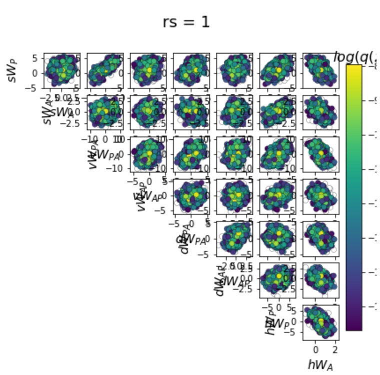

**full NO entropy p = 0.5**

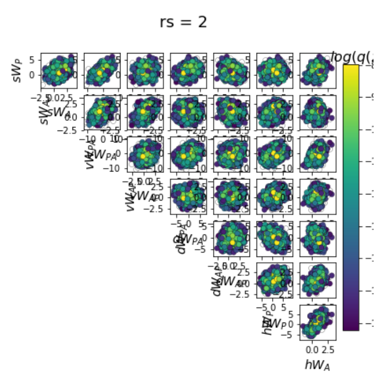

1e20
cost
iterations

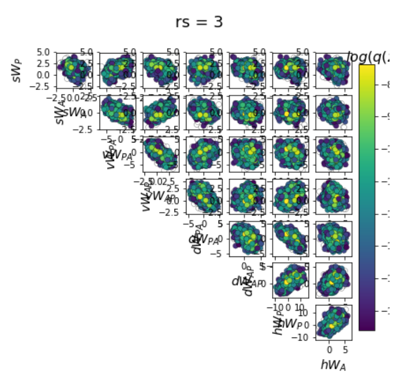

H
iterations
rs = 1
rs = 2
rs = 3
rs = 4
rs = 5

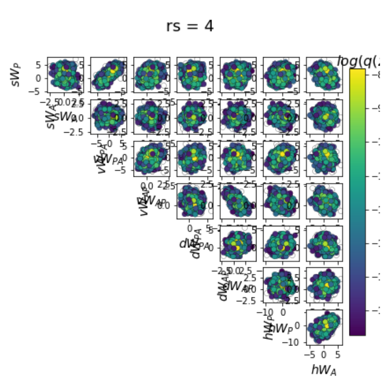

$E[T_1(z)]$
iterations

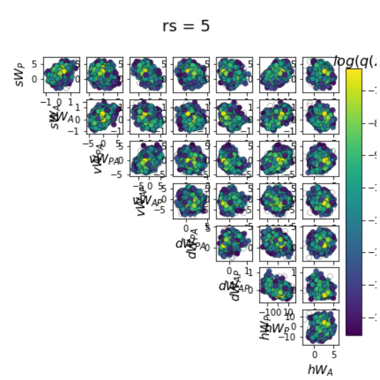

$E[T_2(z)]$
iterations
rs = 1
rs = 2
rs = 3
rs = 4
rs = 5

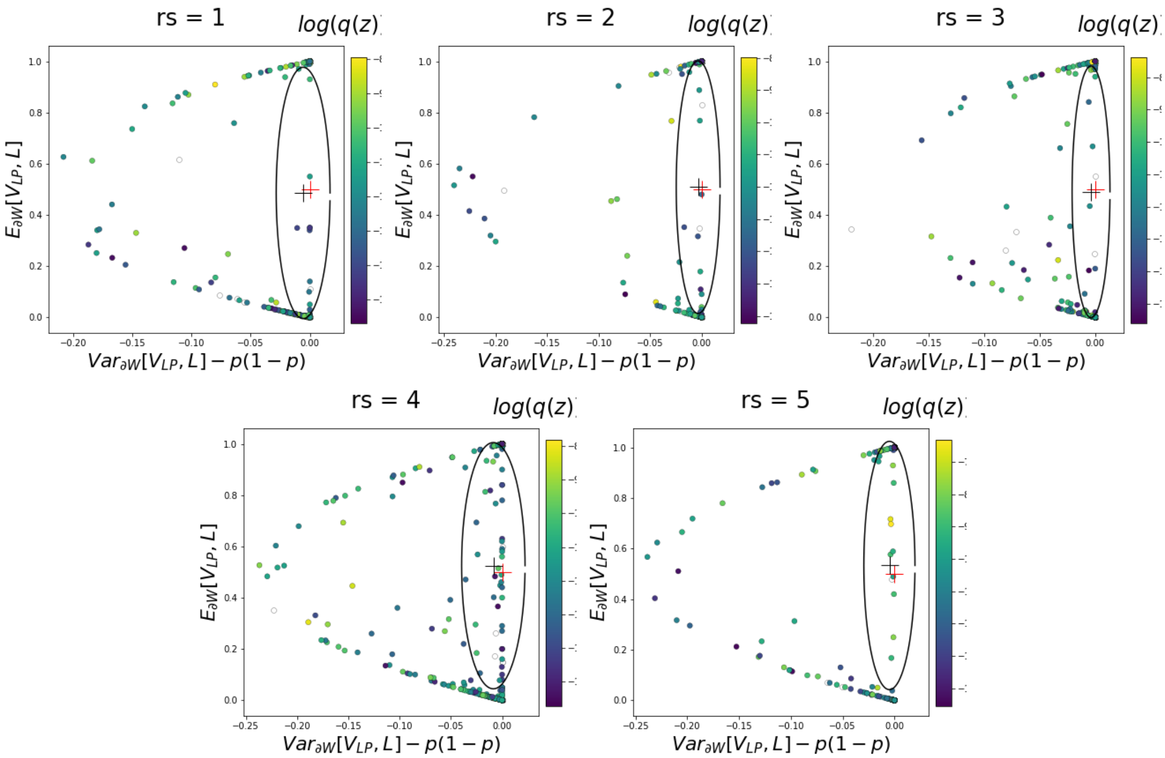

rs = 1
$sW_P$
$hW_A$


rs = 2
$sW_P$
$hW_A$

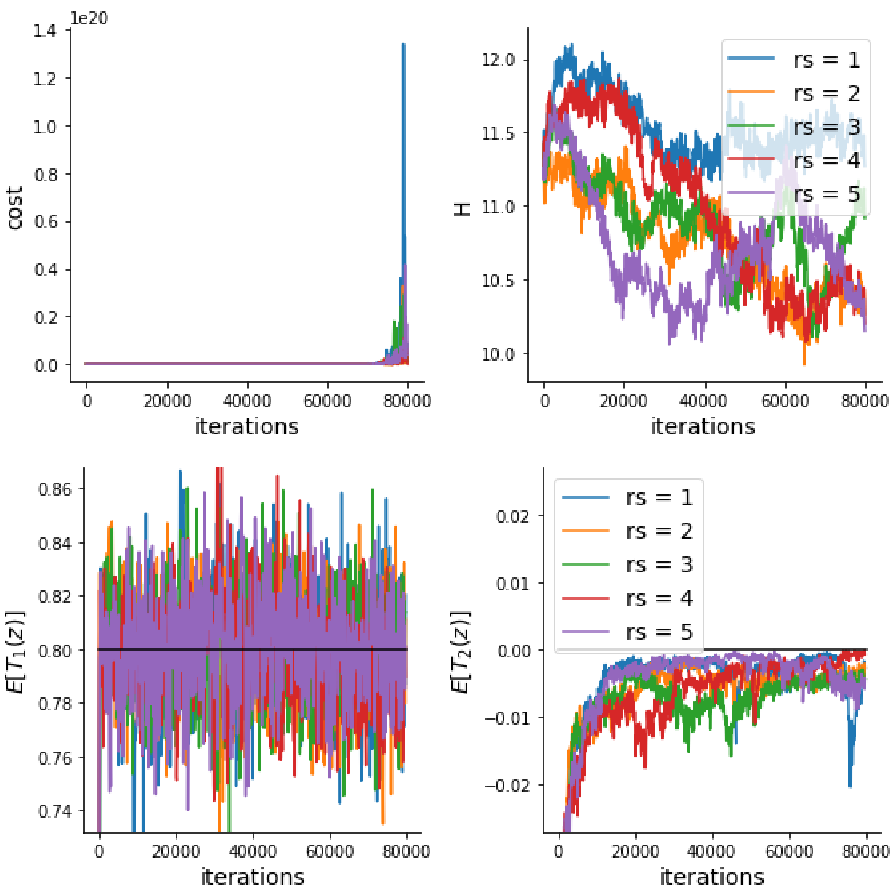

rs = 3
$sW_P$
$hW_A$

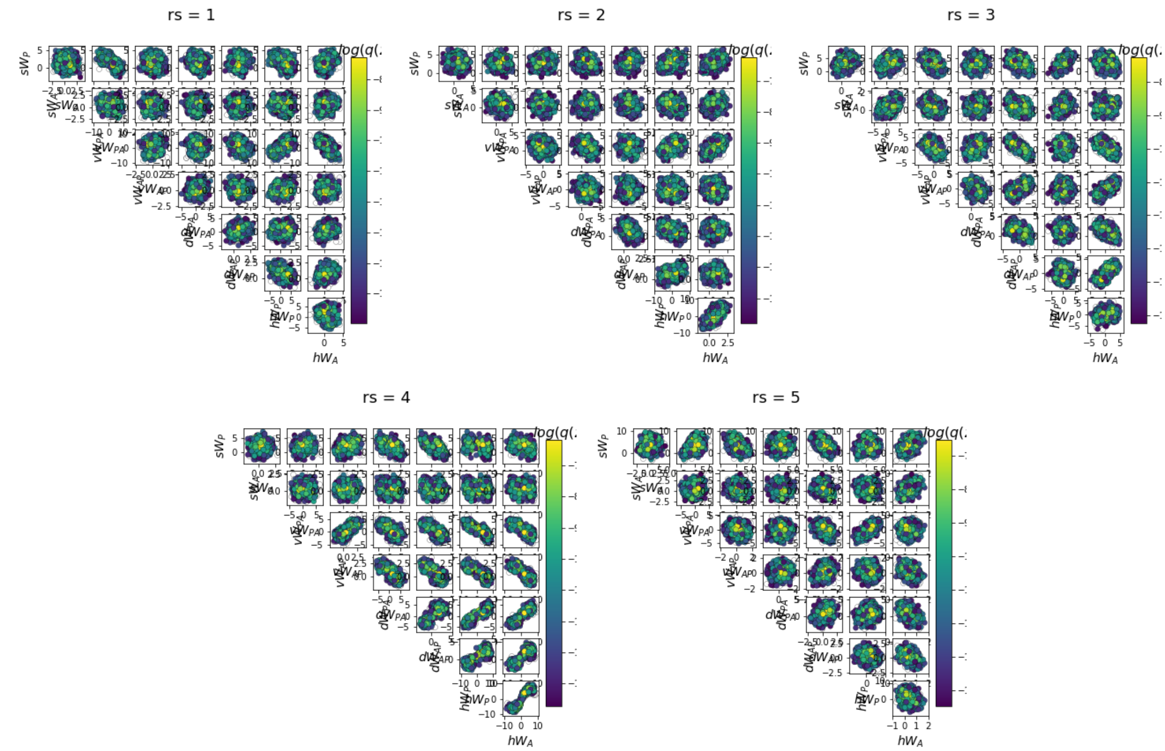

rs = 4
$sW_P$
$hW_A$

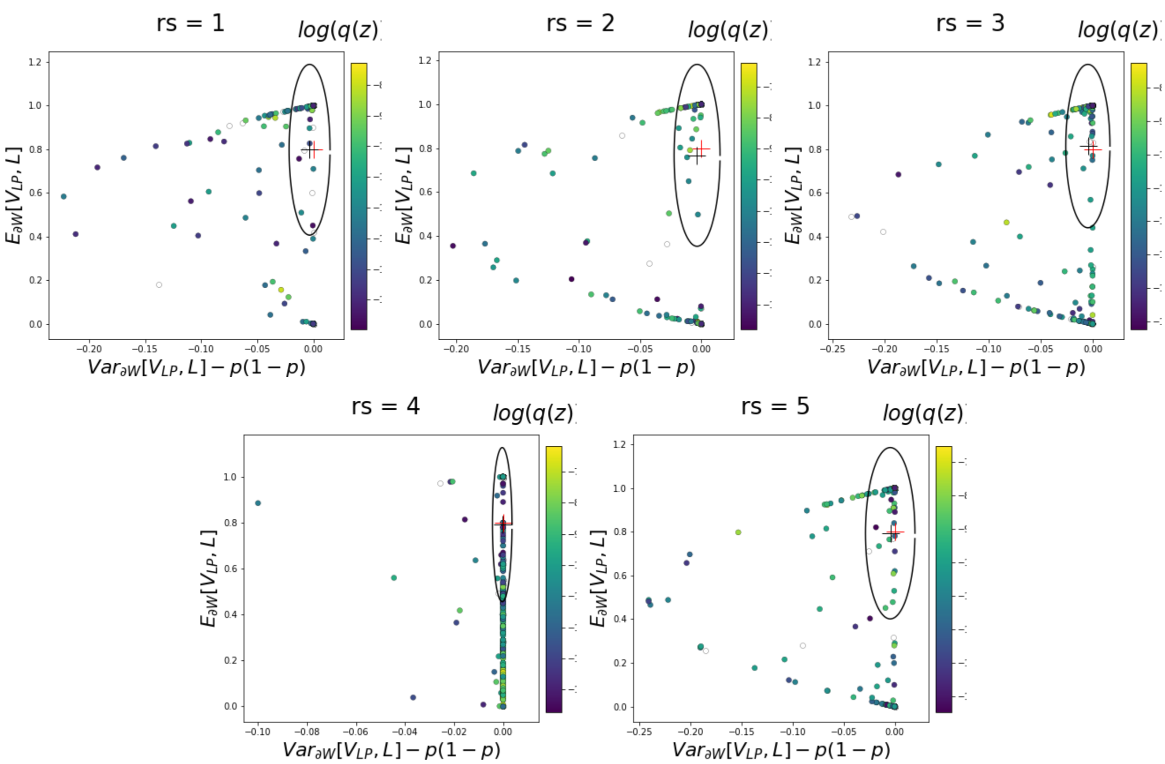

rs = 5
$sW_P$
$hW_A$




**full NO entropy p = 0.8**




rs = 1
rs = 2
rs = 3
rs = 4
rs = 5
$log(q(.$
$sW_P$
$hW_A$


rs = 1
rs = 2
rs = 3
rs = 4
rs = 5
$log(q(z)$
$E_{\partial W}[V_{LP}, L]$
$Var_{\partial W}[V_{LP}, L] - p(1-p)$