

Inverting nonlinear systems with approximately Bernoulli responses

Sean Bittner

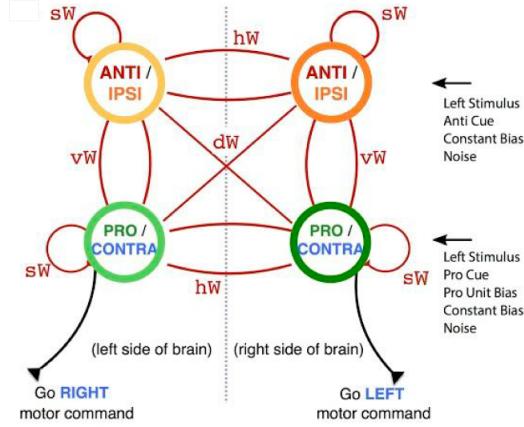
April 4, 2019

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

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

$\theta = 0.05$ and $\beta = 0.5$ control the position and shape of the nonlinearity, respectively, 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

$$v = \begin{bmatrix} V_{LP} \\ V_{LA} \\ V_{RA} \\ V_{RP} \end{bmatrix} \quad u = \begin{bmatrix} U_{LP} \\ U_{LA} \\ U_{RA} \\ U_{RP} \end{bmatrix} \quad (2)$$

The internal variables follow dynamics:

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

with time constant $\tau = 0.09s$ and gaussian noise $\sigma \partial W$ controlled by the magnitude of σ . 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

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

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

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

$$\begin{aligned} I_{\text{constant}}(t) &= E_{\text{constant}} [1 \ 1 \ 1 \ 1]^\top \\ I_{\text{P,bias}}(t) &= E_{\text{P,bias}} [1 \ 0 \ 0 \ 1]^\top \\ I_{\text{P,rule}}(t) &= \begin{cases} E_{\text{P,rule}} [1 \ 0 \ 0 \ 1]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \\ I_{\text{A,rule}}(t) &= \begin{cases} E_{\text{A,rule}} [0 \ 1 \ 1 \ 0]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \\ I_{\text{choice}}(t) &= \begin{cases} E_{\text{choice}} [1 \ 1 \ 1 \ 1]^\top, & \text{if } t > 1.2s \\ 0, & \text{otherwise} \end{cases} \\ I_{\text{light}}(t) &= \begin{cases} E_{\text{light}} [1 \ 1 \ 0 \ 0]^\top, & \text{if } t > 1.2s \text{ and Left} \\ E_{\text{light}} [0 \ 0 \ 1 \ 1]^\top, & \text{if } t > 1.2s \text{ and Right} \\ 0, & t \leq 1.2s \end{cases} \end{aligned}$$

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

Reduced Model

$$W = \begin{bmatrix} sW & vW & dW & hW \\ vW & sW & hW & dW \\ dW & hW & sW & vW \\ hW & dW & vW & sW \end{bmatrix} \quad (6)$$

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 :

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

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} :

$$E_{z \sim q_\phi} [E_{\sigma \partial W} [V_{LP,ss} | s = L, c = P, z]] = E_{z \sim q_\phi} [\hat{p}] = p_{LP} \quad (8)$$

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} .

$$Var_{\sigma \partial W} [V_{LP,ss} | s = L, c = P, z] = \hat{p}(1 - \hat{p}) \quad (9)$$

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

$$E_{z \sim q_\phi} [Var_{\sigma \partial W} [V_{LP,ss} | s = L, c = P, z] - \hat{p}(1 - \hat{p})] = 0 \quad (10)$$

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 Attempt 1: Two constraints

To begin, I wanted to see if I could train a DSN on the reduced parameterization of the DSN model (4 parameter connectivity matrix) to produce a Bernoulli rate of 0.5. In Duan et al. 2019, many other parameters are treated as free parameters:

- 8 connectivity params
- 6 input params – (E_{constant} , $E_{P,\text{bias}}$, $E_{P,\text{rule}}$, $E_{A,\text{rule}}$, E_{choice} , E_{light})
- 1 noise param – σ
- 1 opto param (we're not worried about this yet)

For this analysis, I set $E_{\text{constant}} = 0.0$, $E_{P,\text{bias}} = 0.1$, $E_{P,\text{rule}} = 0.5$, $E_{\text{choice}} = -0.2$, $E_{\text{light}} = 0.1$, and $\sigma = 0.3$ as they were the means of Figure S6 in Duan et. al. 2019. Just because these are the means of the marginalized distribution of satisfactory parameters, does **not** mean that the joint

distribution has samples here. That's something to be mindful of, considering we may be working in a region of parameter space with no satisfactory weight matrices.

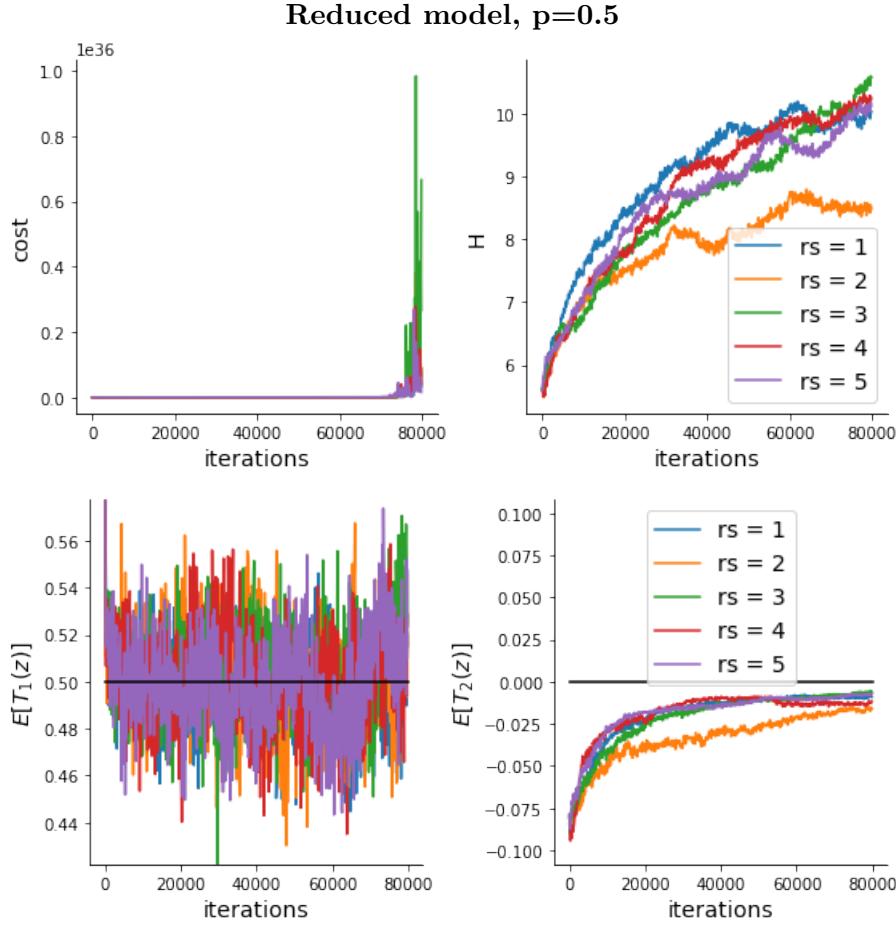
$E_{A,\text{rule}}$ is irrelevant for the LP condition, so it was set to zero.

I trained a DSN with batchsize ($M=1000$) on the reduced (4 weight param) SC model to produce the following behavior:

$$E_{z \sim q_\phi} [\hat{p}] = 0.5$$

$$E_{z \sim q_\phi} [Var_{\sigma \partial W} [V_{LP,ss} | s = L, c = P_i z] - \hat{p}(1 - \hat{p})] = 0$$

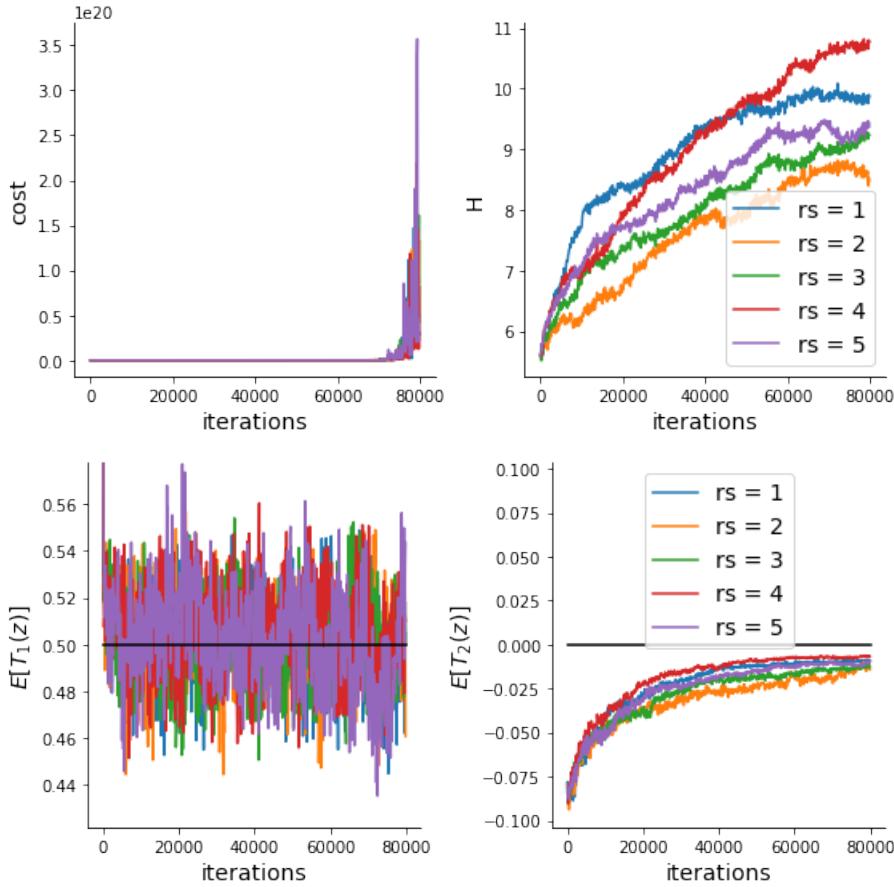
This was with an initialization of augmented lagrangian (AL) parameter $c_0 = 10^{15}$. All c initializations with exponent 20 or greater had numerical instabilities. Each of the 5 randomly initialized optimizations of 10 planar flows ran for 40 AL epochs.



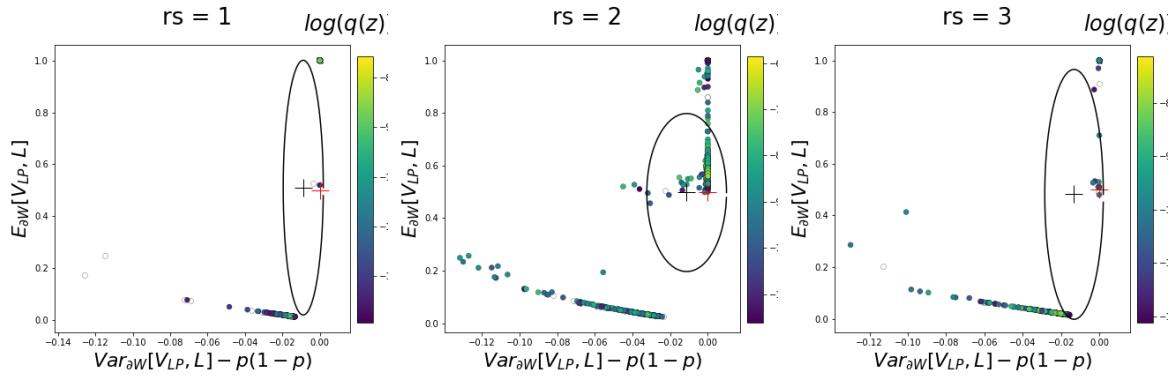
The rate of 0.5 is matched (bottom left), yet the Bernoulli variance constraint is undershot (bottom right). Again, there will always be some degree of empirical undershoot for such a constraint. Notably, after the constraint satisfaction has saturated, the entropy continues to increase substantially. An important question is whether this increase in entropy is solely driven by the entropy term in the optimization (which is weighted very little) or actually by the choice of constraints.

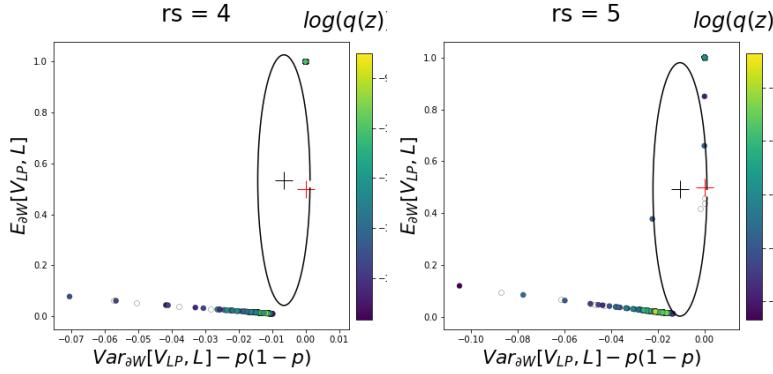
I ran the same optimization with $c_0 = 0$, dropping the entropy term from the cost function entirely. Again, we saw constraint adherence saturate, while entropy grows.

Reduced model, no entropy term, p=0.5

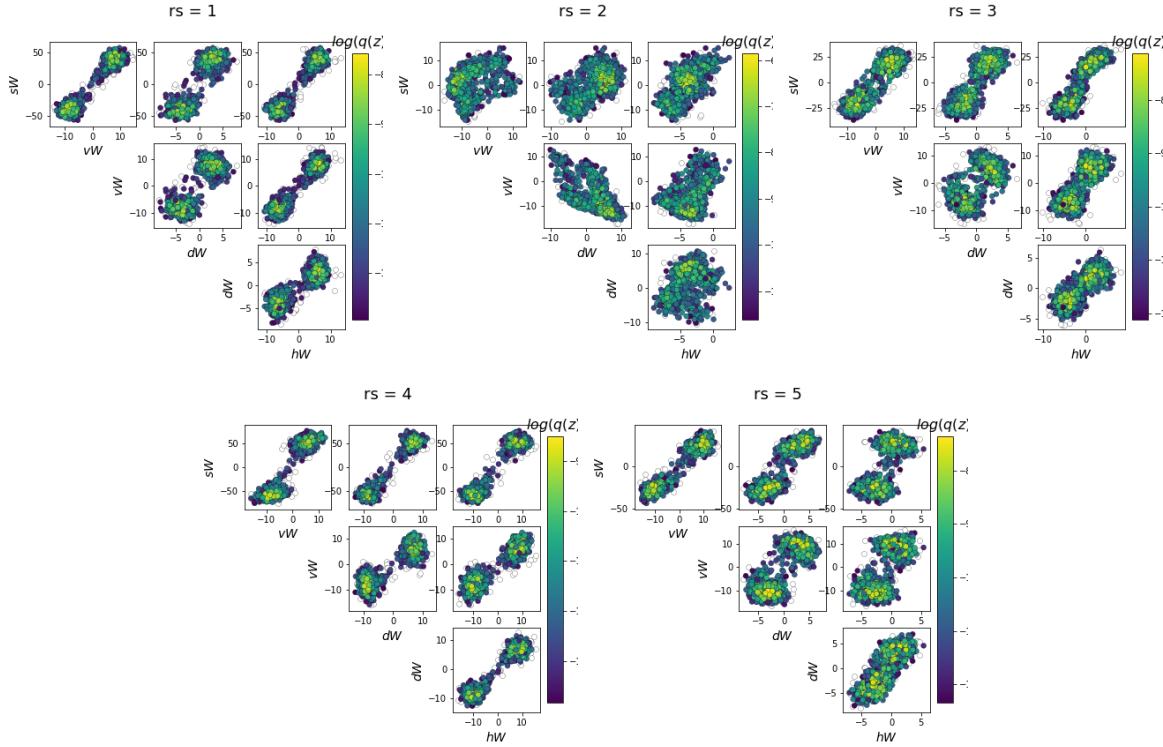


To understand why entropy is growing, it may be insightful to look at the distributions underlying the mean sufficient statistics of the behavior. Below, I've plotted the \hat{p} and Bernoulli variance error across DSN samples at the end of optimization for each initialization.



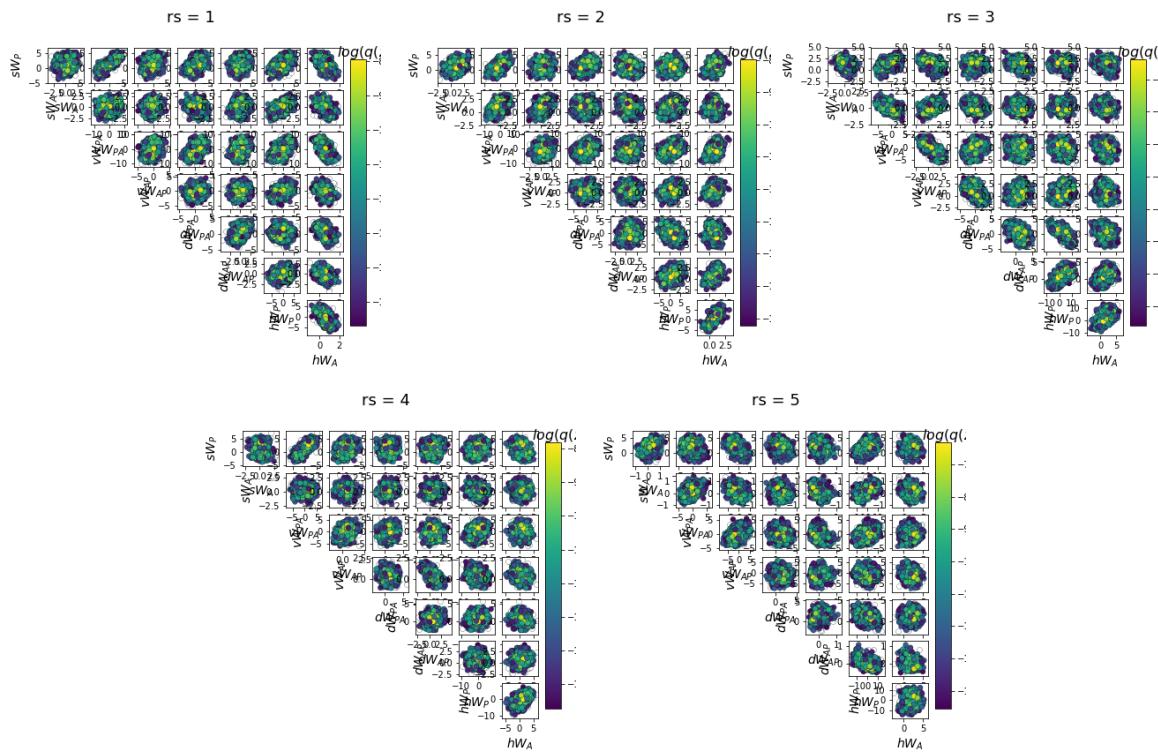
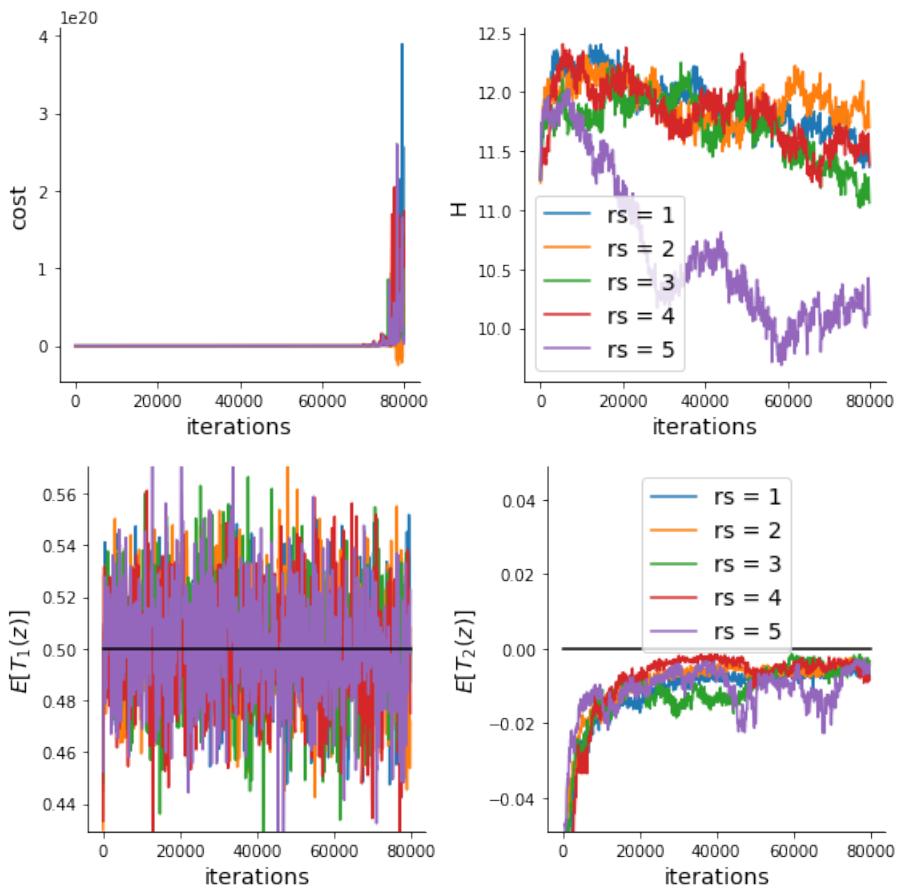


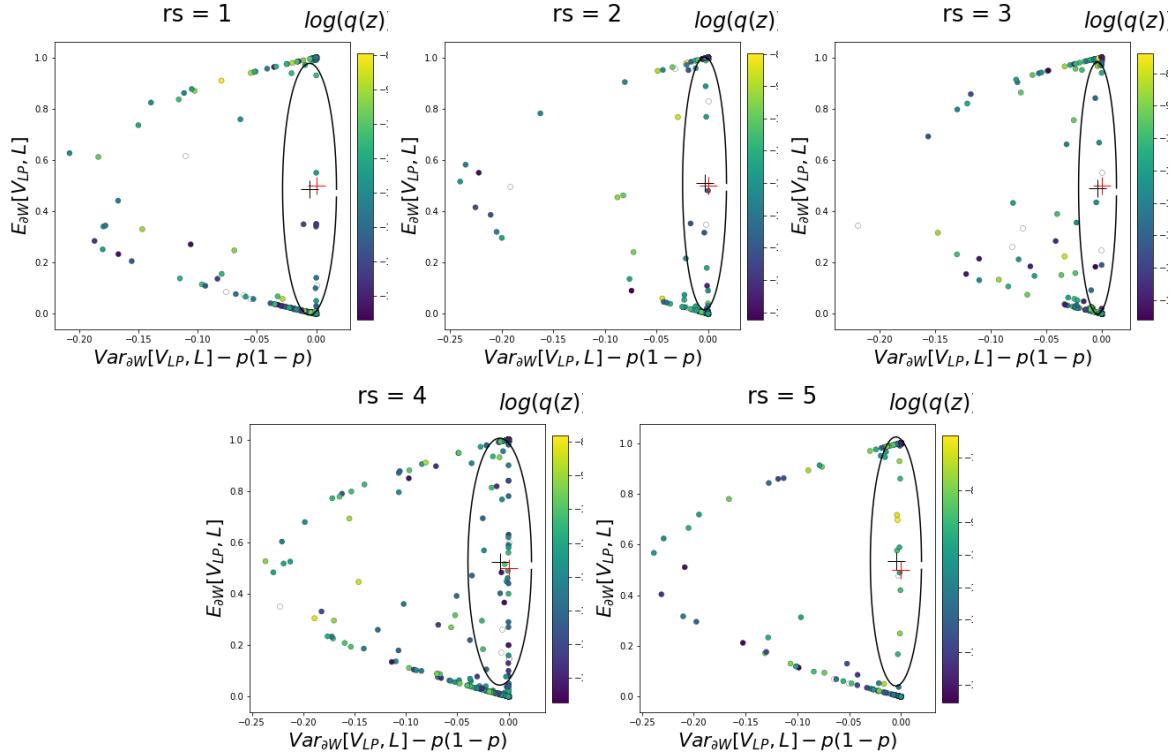
The DSN is producing a bimodal distribution of networks that grow to saturation in V_{LP} , and half that decay to zero every time (no variance). This makes sense in the DSN parameter distributions which have a connectivity mode in each of the all-positive and all-negative “quartants.”



I wondered if the full mode (8-param connectivity) would fair better in constraint satisfaction, but that did not happen.

Full model, no entropy term, $p=0.5$



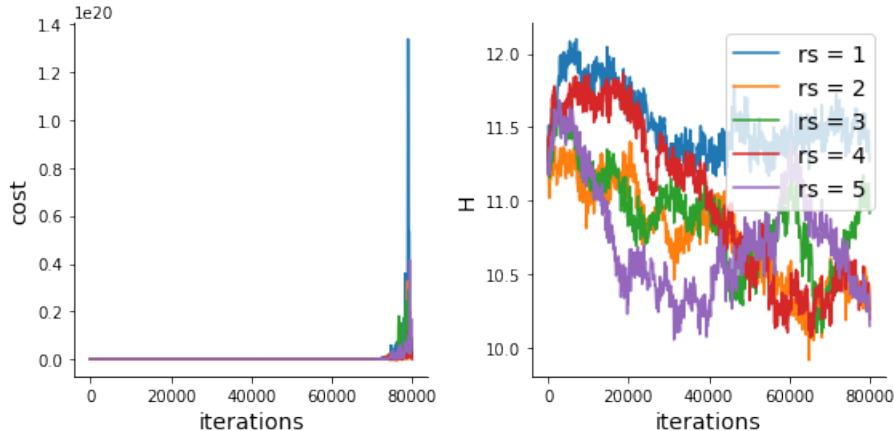


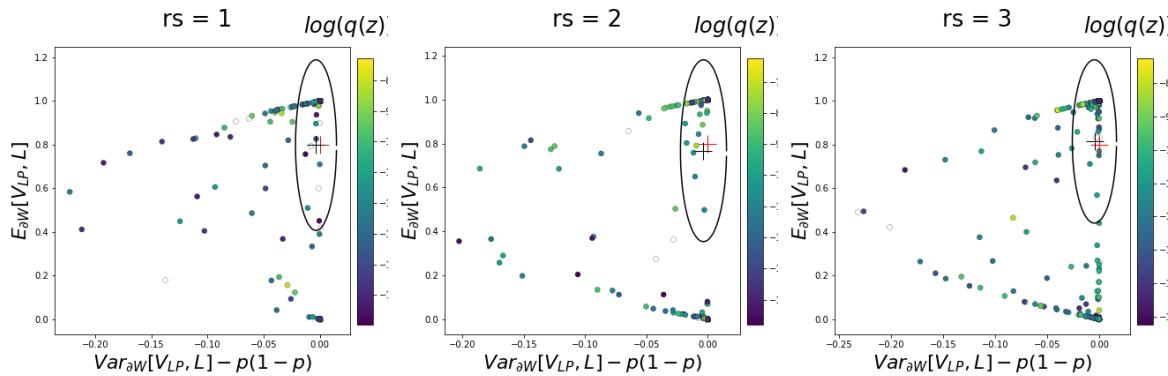
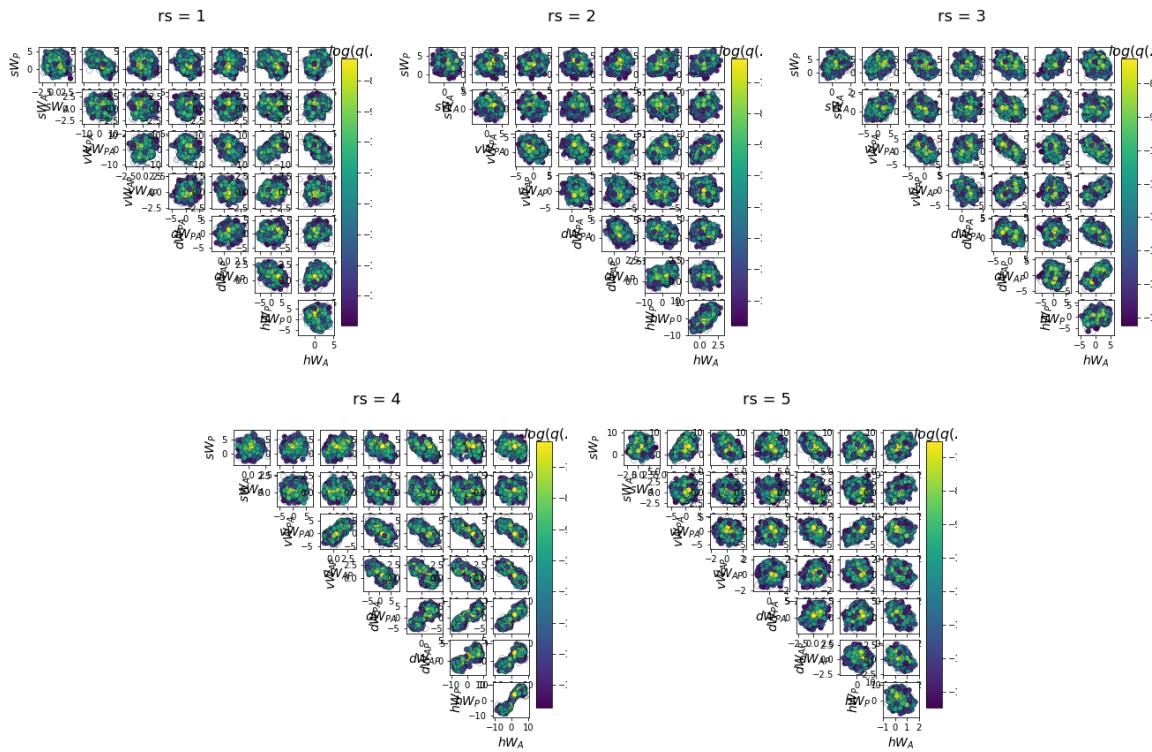
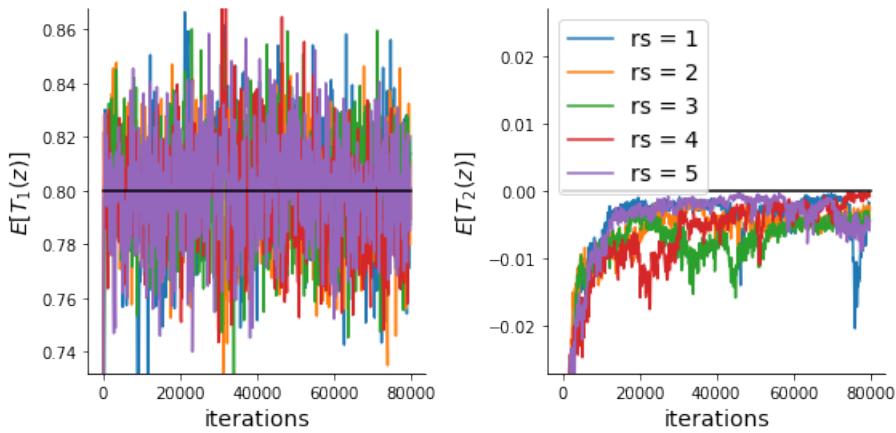
The apparent parabola in the behavior plots is the 0-variance line of DSN samples, while the y-axis from 0-1 is the Bernoulli-variance line of DSN samples. The network achieves the constraints by putting two modes on the extreme points of the parabola of 0-variance networks. Ironically, this is the exact opposite of what we wanted.

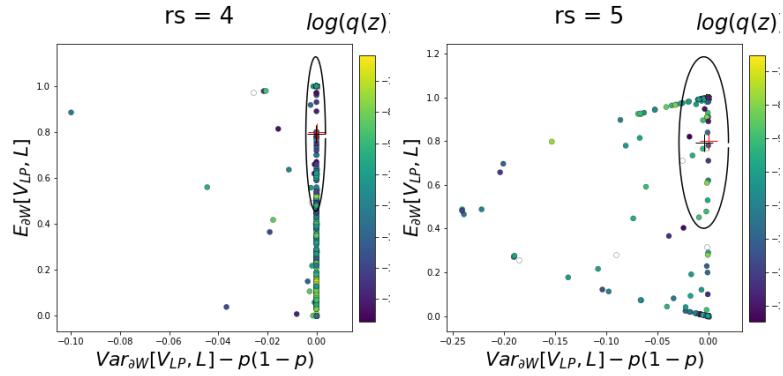
In case you were wondering what happens with a $p \neq 0.5$, here is the same optimization with $p = 0.8$. Same result, except the base rates of those two expansive and decay networks are shifted from $\pi = [0.5, 0.5]$ to $\pi = [0.8, 0.2]$

EXCEPT, for random seed 4 (red), which puts all its density on the Bernoulli-variance line, NOT predominantly the 0-variance line. At least, we know that placing all density on the Bernoulli variance in this model is learnable. Maybe, that indicates that I should run the optimizations for longer, or play with new architectures.

Full model, no entropy term, $p=0.8$





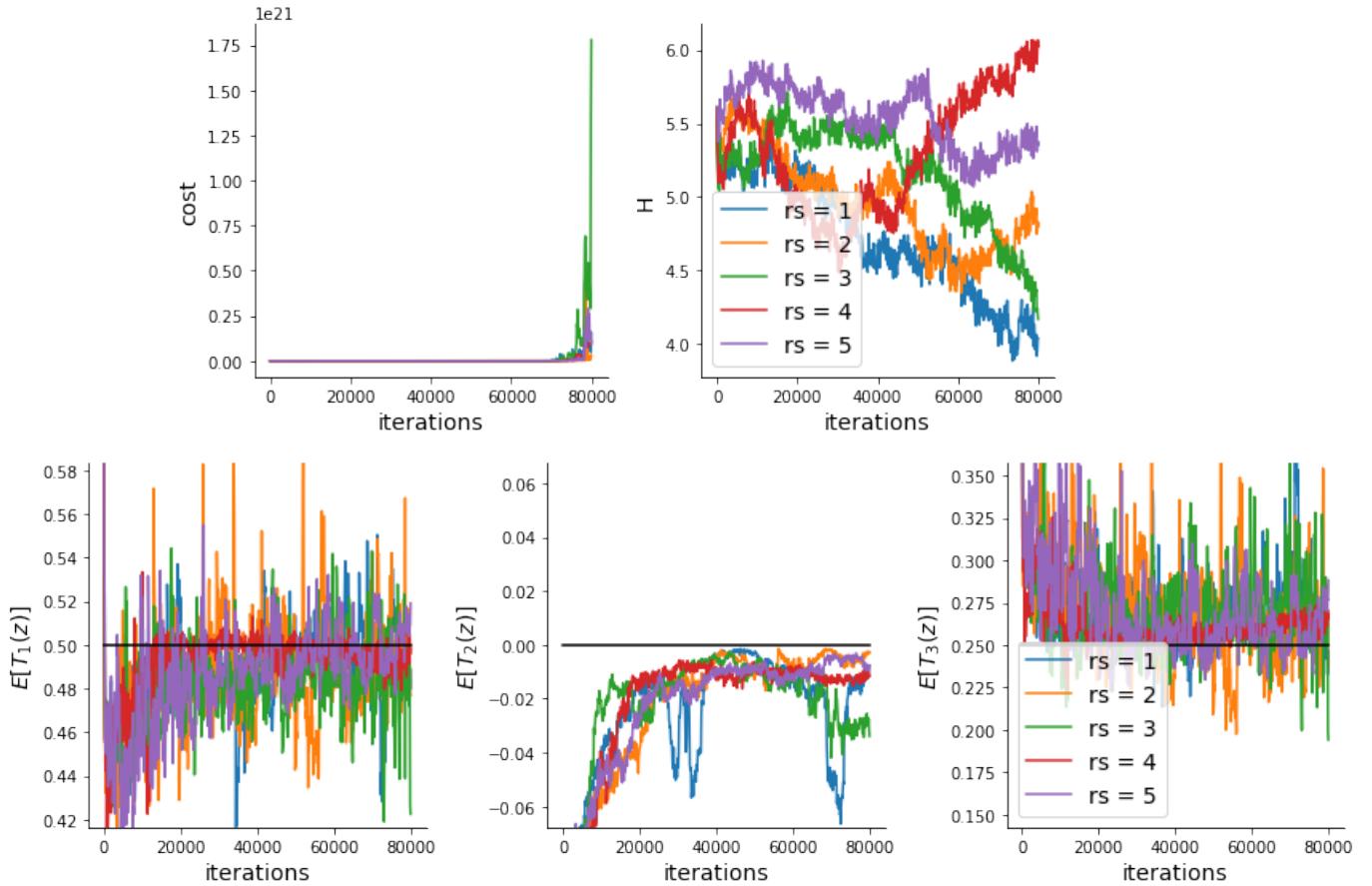


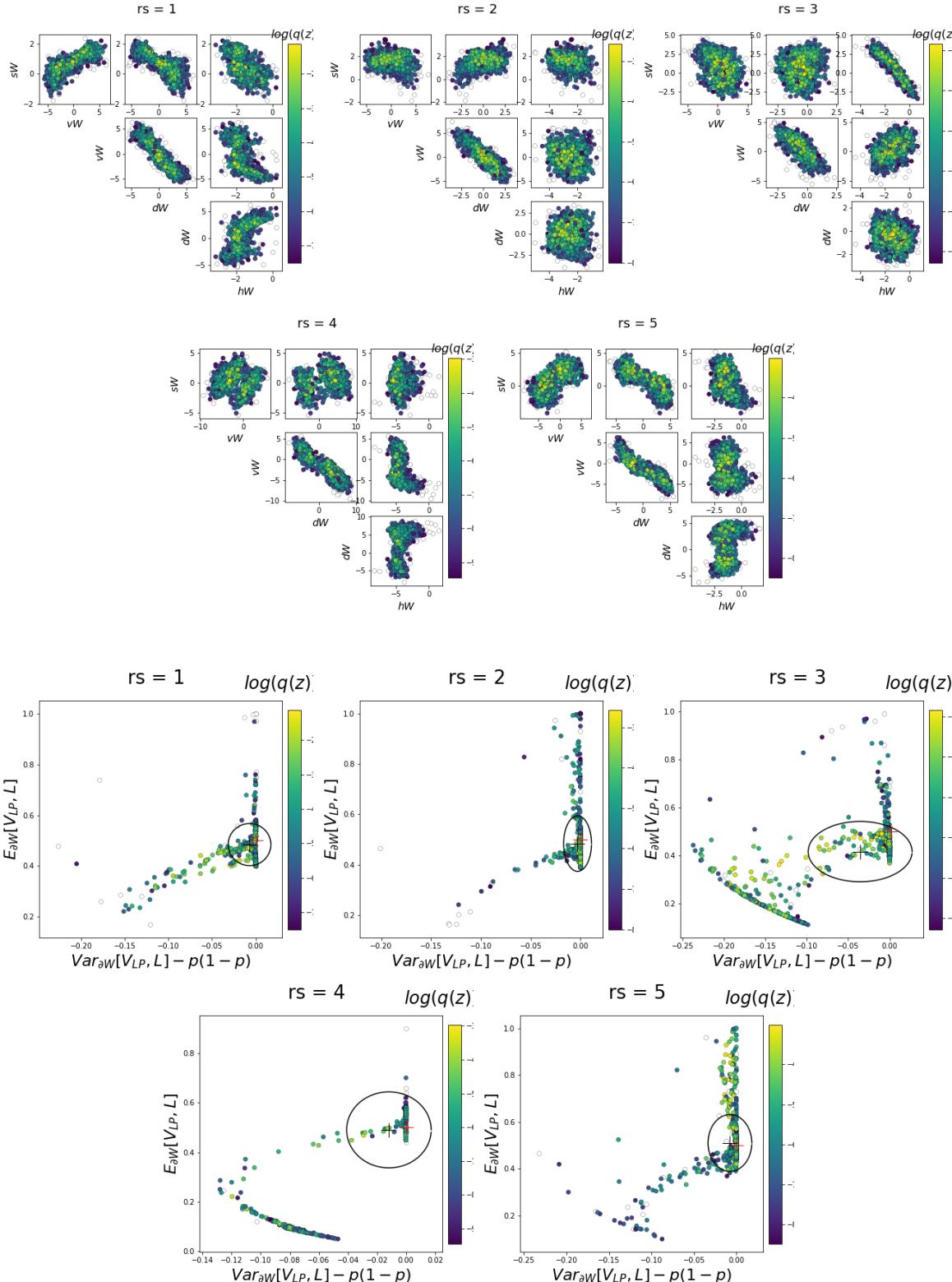
5 Attempt 2: Two constraints

To combat the bimodal solutions on the extremes of the 0-variance network parabola, we can introduce a variance constraint on the estimated \hat{p} of the networks. Now, we have three constraints:

$$\begin{aligned} E_{z \sim q_\phi} [\hat{p}] &= 0.5 \\ E_{z \sim q_\phi} [Var_{\sigma \partial W} [V_{LP,ss} \mid s = L, c = P_i z] - \hat{p}(1 - \hat{p})] &= 0 \\ Var_{z \sim q_\phi} [\hat{p}] &= 0.0001 \end{aligned}$$

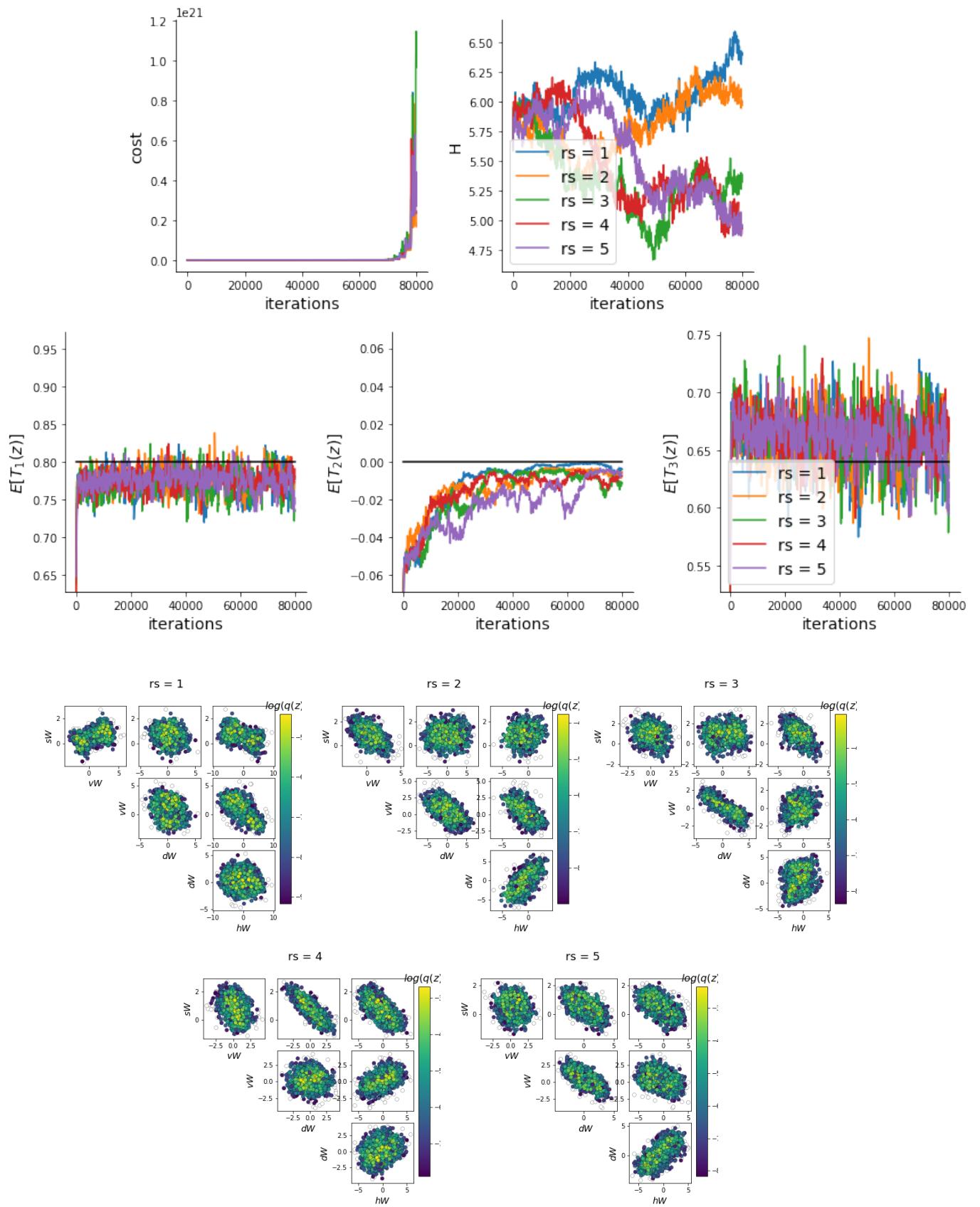
Reduced model, no entropy term, \hat{p} var constraint, $p=0.5$

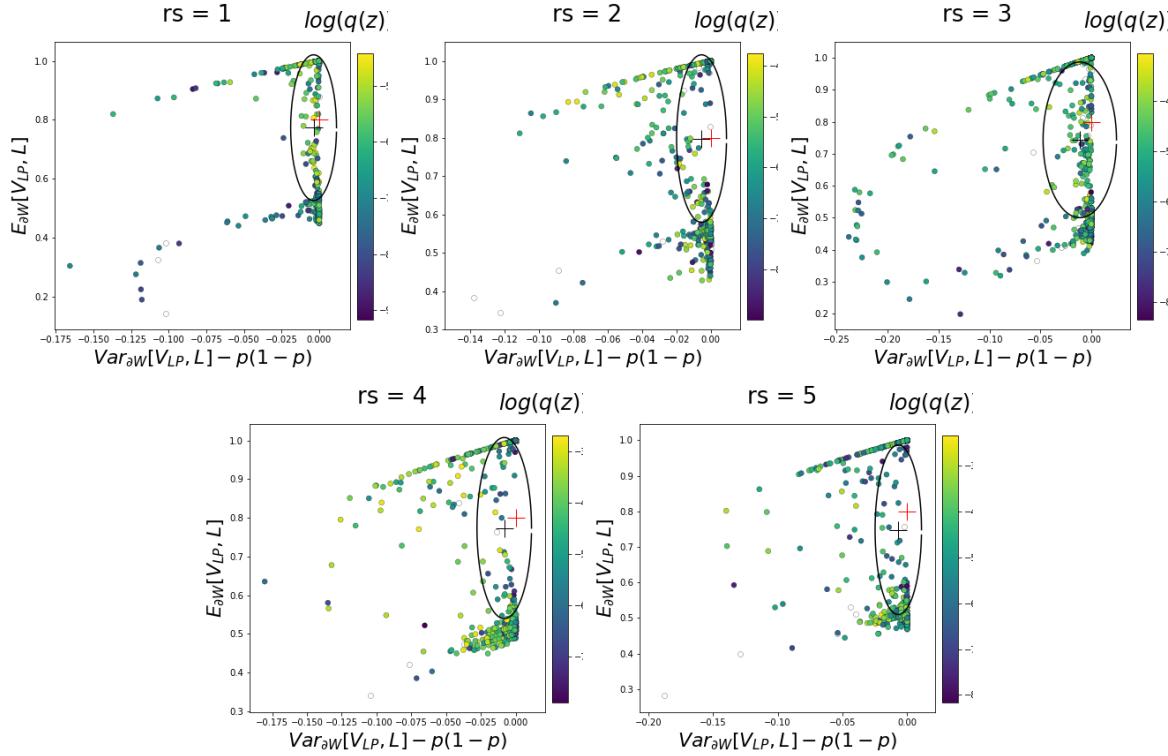




This more or less gives us the Bernoulli networks we were looking for. But, this is only at $p = 0.5$. What happens for other p , such as $p = 0.8$?

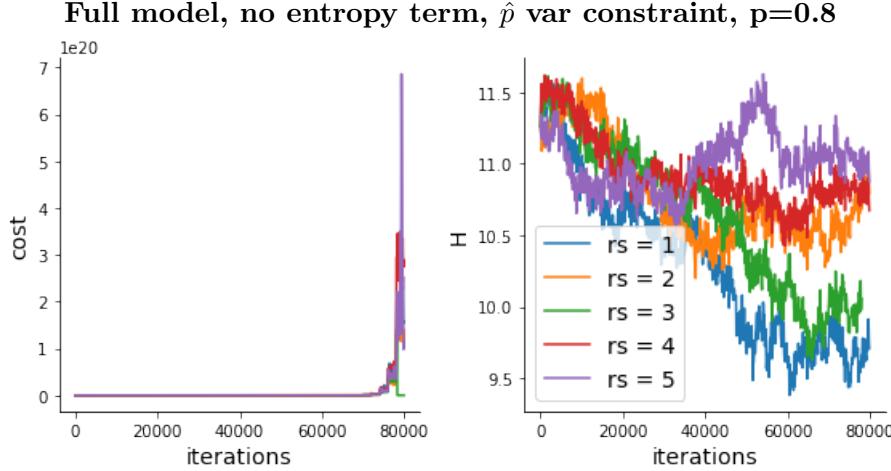
Reduced model, no entropy term, \hat{p} var constraint, $p=0.8$

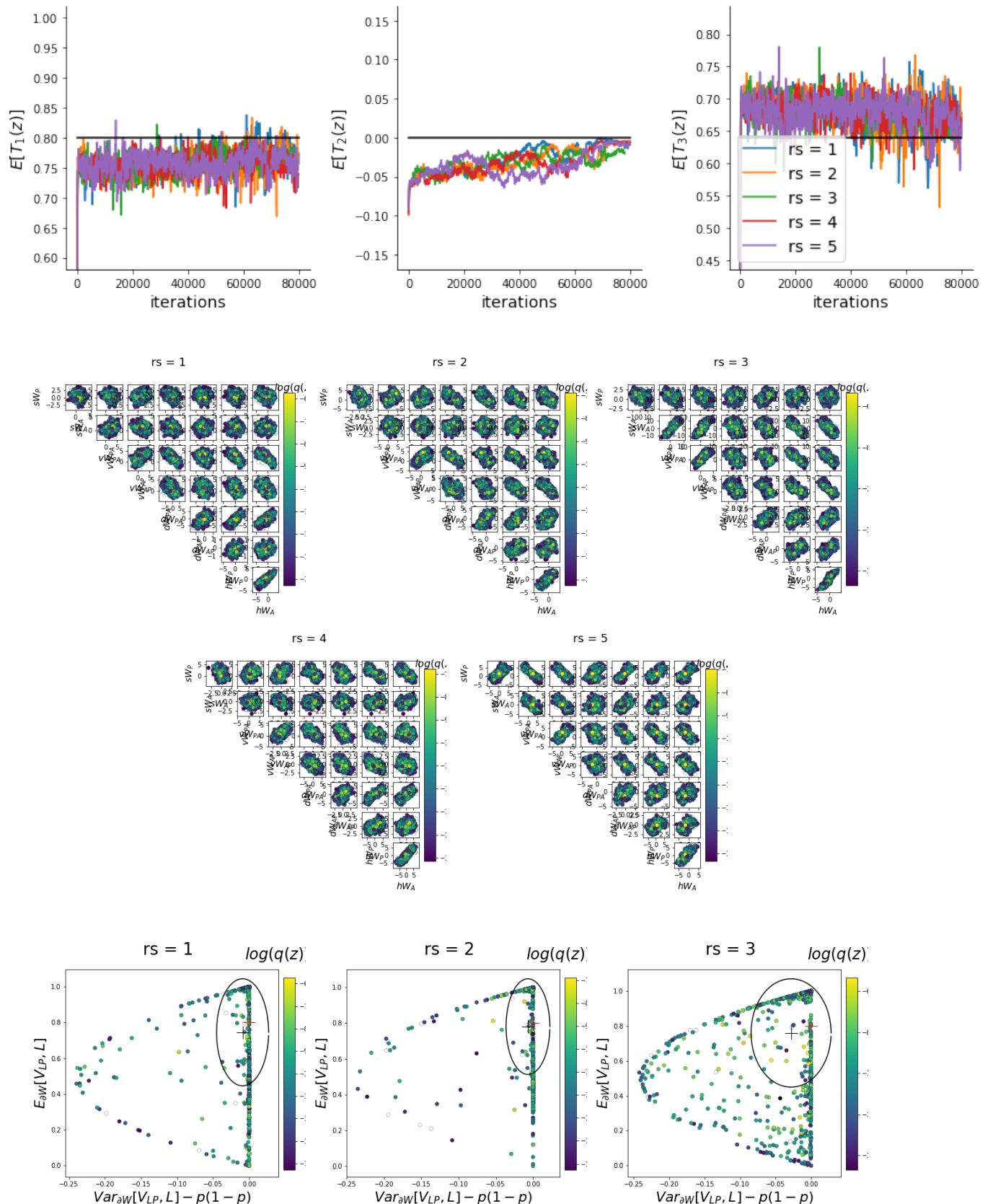


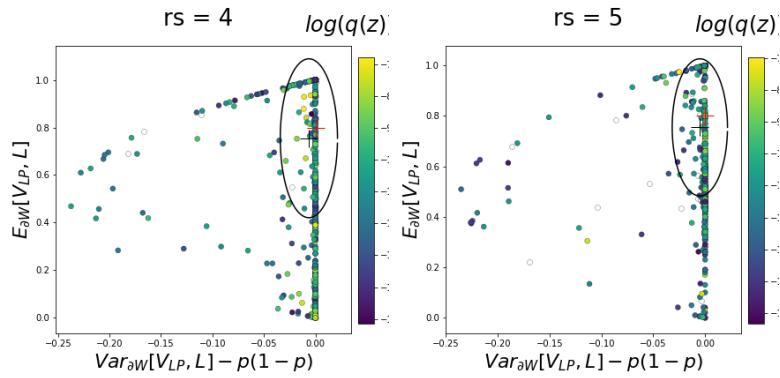


Without the third constraint (on the second moment of \hat{p}), we were able to fit the first constraint (on the first moment of \hat{p}) perfectly. We hoped the third constraint would simply control the variance, but it seems that low variance in \hat{p} is hard to learn. Thus, the first and third constraints trade-off their violations optimally with respect to the cost function (the first moment undershoots, while the third moment overshoots). As always the second moment is undershooting.

We again have a similar bimodality problem as before. There are two main modes in the behavioral distribution ($\hat{p} = 0.5$, approx bernoulli-variance) and ($\hat{p} = 1.0$, 0-variance). Let's try the full model, to see if an increased parameterization is able to satisfy the constraints.







Encouragingly, most of the density of the DSN is on the Bernoulli-networks line, although we still have the too much variance due to constraint violation.