

In our lab we've been exploring the BCM learning rule for years, and have found that it produces nice RF's in natural scene environment. The question I want to deal with today is: what do you need to get them? You can write down many learning rules which have the same basic form, with perhaps different motivation. To what extent do we get the same results with these rules, and to what extent do we get different results.

I will start with a description of some of the observed properties of the natural scene environment, which will outline the main concepts used in this work. I follow it with a brief description of projection pursuit, which is a method for motivating a whole class of learning rules which have been used to model the orientation selectivity in visual cortex. Next I give a presentation of the architecture and environment used in this study, as well as some of the learning rules used. Finally I present a new form of structure removal which will help us more quantifiably define the concept of sparse, and help us figure out what is necessary for the development of orientation selectivity.

it is generally believed that biological neuronal responses are sparse, but what does that actually mean? If we think of a neuron has coding information when it responds to a particular subset of patterns in its environment, and not responding to others, then a sparse coding would be when that subset of patterns which give high responses is a small part of the whole environment. For example, let's look at the response properties of this model neuron over the entire environment. What is shown here is the output distribution, or the probability density for finding a particular activity level. It has been shown by ruderman and others that the distribution that comes from natural scenes is very nearly exponential. This implies that most of the patterns have a very low activity, and a few have a much higher activity, which conforms to our idea of a sparse coding. sparse coding is a reasonable thing to expect, because we assume that neurons in the visual cortex are optimized for the particular environment, so a sparse coding would be a very efficient representation of the environment. How do we obtain a learning rule which achieves this sparse coding, and how can we more quantifiably describe the sparsity itself?

One method for obtaining these learning rules is a Projection Pursuit. We choose an energy function which measures deviation from a Gaussian distribution, usually in the form of polynomial moments. This energy function has a value for every direction chosen in the data space, so a gradient descent or ascent of this function guides the search for optimum projections. For example, we could

choose kurtosis, which is related to the fourth moment of the data, as a measure of deviation from gaussian distribution. We then search for directions where the projected data maximizes this measure, leading to one of these two directions. The same can be done if multi-modality is chosen as the deviation from gaussianity, leading to the x direction in this bi-modal distribution. I will now go on to explain our implementation of several rules which yield sparse coding, and try to explore the concept of sparsity a little more.

We choose to use a single cell implementation, with random patches taken from an input image, processed with retinal preprocessing, and presented to the neuron. The output of the neuron is given by the dot product of the inputs with the weights, and is passed through a sigmoid function. When I say “sigmoid” here, I am really referring to a “rectifying sigmoid”, or an “asymmetric sigmoid” pictured here. If we are taking  $c = 0$  as spontaneous, and we note that the spontaneous rate in the cortex is fairly low, then the value of the output cannot be negative as much as it can go positive.

The retinal preprocessing is difference of gaussians filter, applied monocularly. We are only dealing with one eye in this study. The filter is supposed to represent the response properties of retinal ganglion cells. The results of this filter is shown here. We have also explored a whitening filter, but there isn't enough time to go into that right now.

Some of the learning rules we used are derived from these energy functions. Kurtosis, which is related to the fourth moment of the data, we expect to be very sensitive to outliers. The energy function for Quadratic BCM (introduced by Intrator and Cooper in 1992) is a combination of the 3rd moment and 2nd moment. Skewness, also related to the third moment, we expect may not be quite as sensitive to the outliers, and hence may not yield as sparse of a representation.

A quick note that the sigmoid on the output, motivated by biology, is actually a necessity for odd moment learning rules when dealing with symmetric distributions. In these cases, a symmetric distribution would cause the odd moments to vanish. The sigmoid forces the symmetric distribution to be asymmetric. This is not a problem for even-powered moments, like kurtosis.

The learning rules are then gradients calculated by a gradient ascent of these energy functions. The equations themselves are pretty opaque, but one can see from a quick picture that the functions look quite similar. They have a crossing at zero and at some positive value, though their specific dependence on the data can be different. To what extent do these rules behave the same, and

to what extent do they behave differently?

Initially, we can see that they behave fairly similarly. They each give oriented receptive fields and similar looking output distributions. The RFs found by skewness do seem to be a slightly lower spatial frequency. In order to flush out some of the other possible differences, I introduce a new form of structure removal.

As I said before, polynomial moments tend to be sensitive to the outliers. It is instructive, then, to remove those outliers, or that structure in the environment to which the neuron is selective, and see how this affects the learning. Friedman introduced a form of structure removal in 1987, where he transforms the data such that the projected distribution is gaussian: eliminates the information along that projection. In this study, we use a different form of structure removal, where we eliminate patterns with high responses. this does not require the assumption of whitened data, as friedman's method does, and is allows for a partial amount of information to be removed.

The structure removal procedure goes as follows. Say we train a neuron with a particular learning rule, and obtain an oriented receptive field. We then can calculate the output distribution, as before, and, say, mark off where the 99th percentile is (for example). That means that only one percent of the patterns in the input environment elicit responses stronger than this point. We can then convolve the receptive field with the entire image environment, and mark those patterns which yield responses in this 1 percent regime and remove them. If we continue the training of the neuron in this modified environment, then it will never see those patterns: if it were to chose one of them randomly, then it would just choose another. We can now use this as a sensitivity measure. If we delete enough patterns, we should see a change in the RF when we continue training. If we delete just few enough patterns to not see a change in the RF upon retraining, then that percent can be a measure of the sensitivity of a particular learning rule to the outliers, and thus a measure of the sparsity of the code attained. Also, one can do successive structure removal and form a hierarchy of receptive fields, eliminating the most important structure first and less important structure later.

Let's look at the sensitivity measure first. In these pictures, we train a neuron and obtain an oriented RF. We then remove a certain percentage of patterns, and continue training. If the RF doesnt change, then that learning rule was not sensitive to that change. The change in RF can be in orientation or phase, so some of the changes are not as clear as others. We found that kurtosis

and bcm were sensitive to between a tenth to 1 percent of the inputs, whereas skewness was sensitive only to changes above 1 percent.

Repeated structure removal can form a hierarchy of receptive fields. Here is an example using BCM. The number above each figure is the cumulative percentage deleted. Note that we get all of the phases and orientations in the environment, phase changes generally preceding orientation changes. Also, further down the list, are some fairly local receptive fields. Finally We get garbage if too much is deleted.

So how sparse is sparse? Depending on the learning rule and some of the neuron parameters, a neuron can code as little as 1/1000 of the environment. Kurtosis and BCM are more sensitive to the outliers than skewness. There are indications that kurtosis does not perform as well as BCM for multi-modal distributions, but that is another matter. What is needed for orientation selectivity? We don't have a full answer yet. We can't say that only 1/1000 of the environment is needed to attain orientation selectivity, but we can say that changes on the order of 1/1000 of the environment can be detected by neurons and can be influential when the neuron is developing selectivity.

---

# How Sparse is Sparse?

What is required in the  
Natural Scene Environment for the  
Development of Orientation Selectivity

---

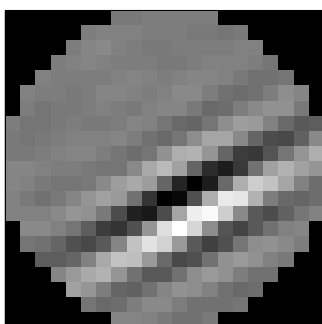
# Outline

---

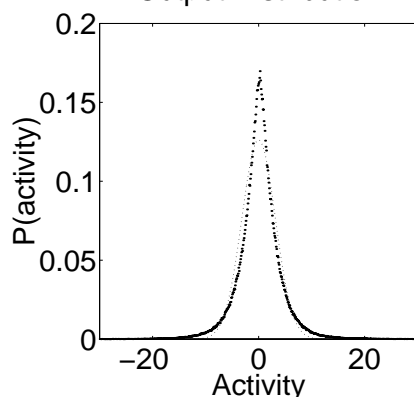
- Distribution produced by natural scenes
- Projection Pursuit
- Architecture and Environment
- Learning Rules
  - Definitions and Motivation
  - Initial Results
- Structure removal
  - Description
  - Differences between Learning rules
  - Effect of Sigmoid
- Conclusions

## Distribution from Natural Scenes

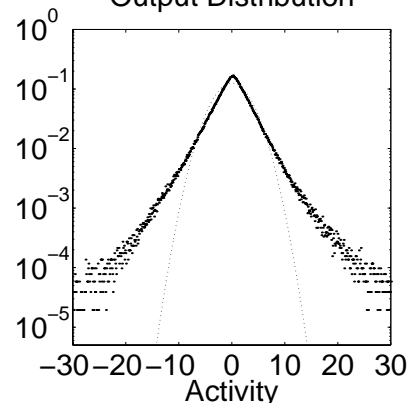
Example Receptive Field



Output Distribution



Output Distribution



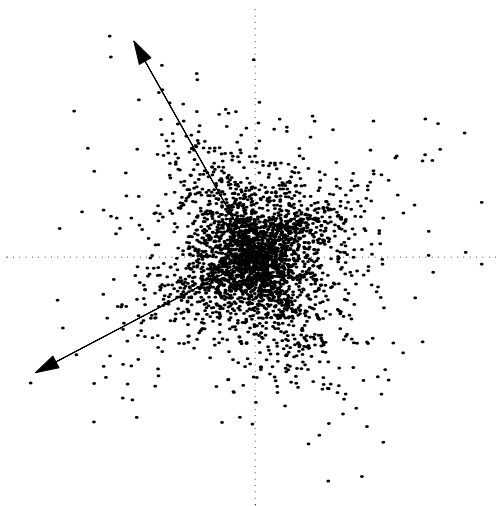
- sparse coding: the neuron responds strongly to the very few important patterns
  - necessity for the visual cortex to have efficient representation
  - observed sparsity of natural scenes (exponential distribution)
  - rules based on sparse coding obtain simple-cell like receptive fields

# Projection Pursuit

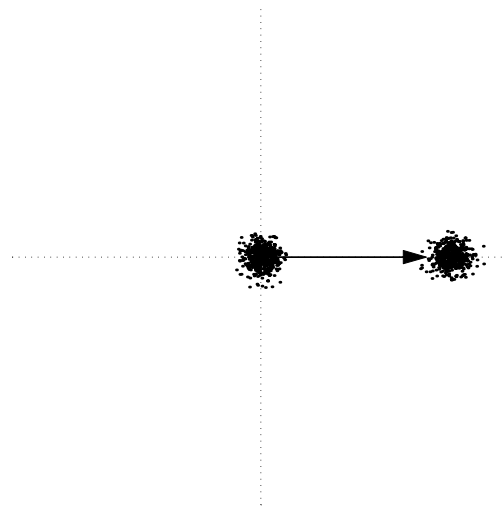
---

- an energy function is chosen which measures deviation from Gaussian distribution, usually in terms of polynomial moments
- gradient descent/ascent of the energy function guides the search for projections
- Examples:

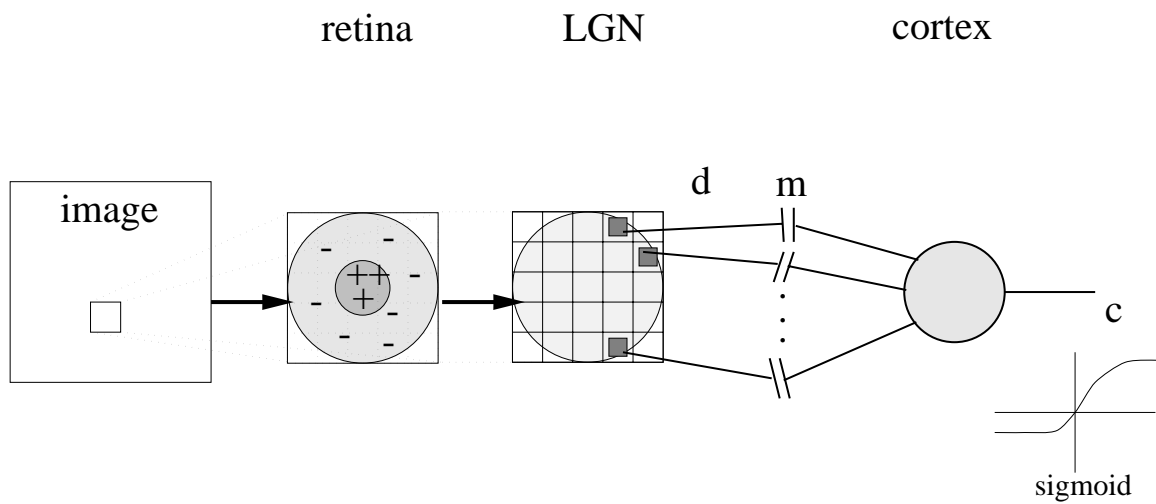
Maximizing Kurtosis



Maximizing Multi-Modality



# Neuron Architecture



- single cell
- retinal preprocessing
- sigmoid function on output:  $c = \sigma(\mathbf{m} \cdot \mathbf{d})$

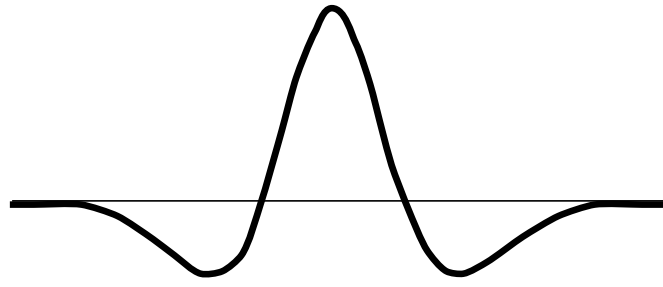
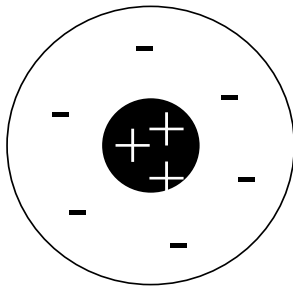
# Natural Scene Environment

---

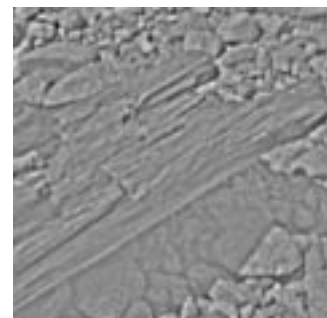
- before retinal preprocessing



- retinal preprocessing (Center-Surround DOG filter)



- after retinal preprocessing



## Energy Functions Used

---

- **Kurtosis 1**

$$K_1 = E[c^4]/E^2[c^2] - 3.$$

- **Quadratic BCM**

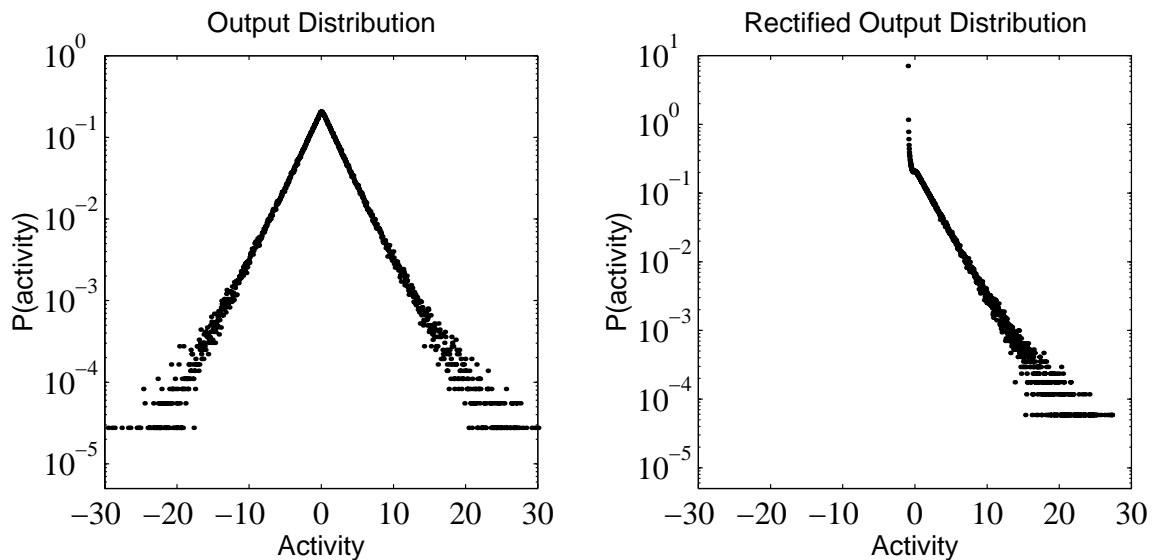
$$\text{QBCM} = \frac{1}{3}E[c^3] - \frac{1}{4}E^2[c^2].$$

- **Skewness 1**

$$S_1 = E[c^3]/E^{1.5}[c^2].$$

# Rectifying Sigmoid

---



- Learning rules which are based on *even* powered moments do not require the rectifying sigmoid
  - kurtosis
- Learning rules which are based on *odd* powered moments **do** require the rectifying sigmoid, if the distribution is symmetric
  - skewness, QBCM

## Gradients of Energy Functions

---

- **Kurtosis 1**

$$\nabla K_1 = \frac{1}{\Theta_M^2} E [c (c^2 - E[c^4]/\Theta_M) \sigma' \mathbf{d}] = \frac{d\mathbf{m}}{dt}.$$

where  $\Theta_m$  is defined as  $E[c^2]$ .

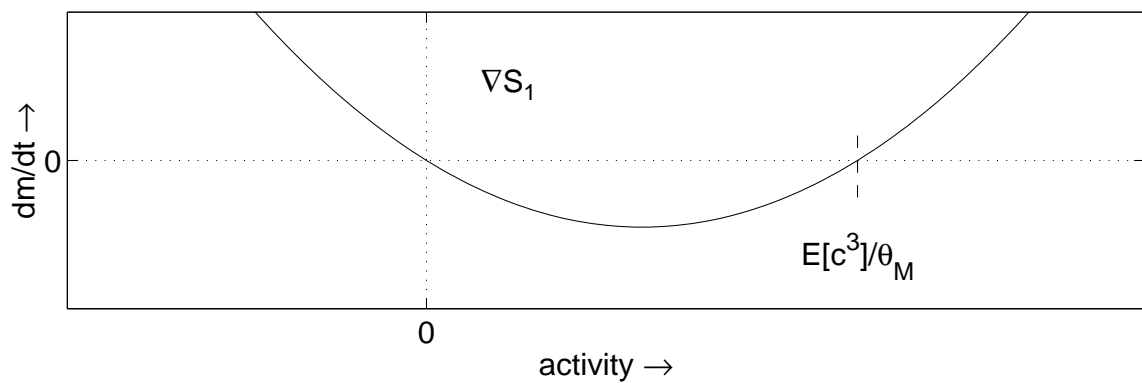
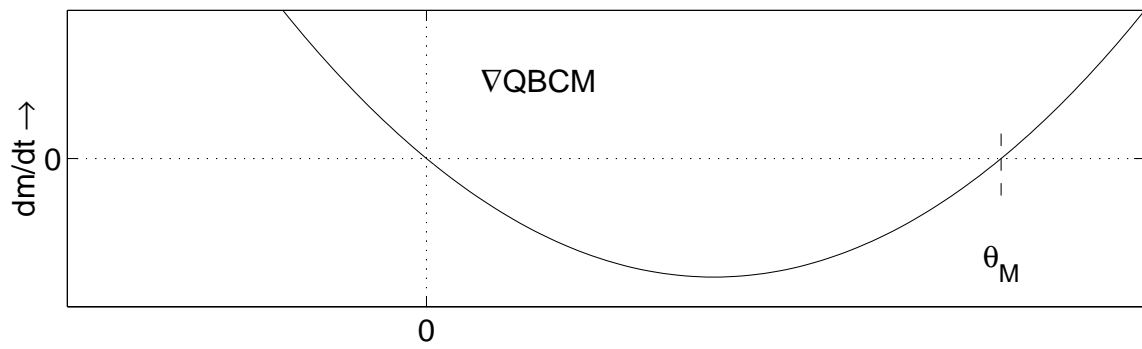
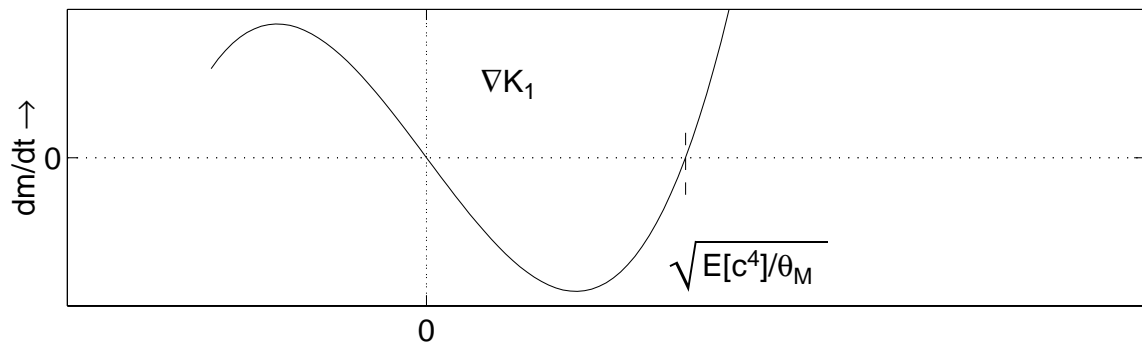
- **Quadratic BCM**

$$\nabla \text{QBCM} = E [c (c - \Theta_M) \sigma' \mathbf{d}] = \frac{d\mathbf{m}}{dt}.$$

- **Skewness 1**

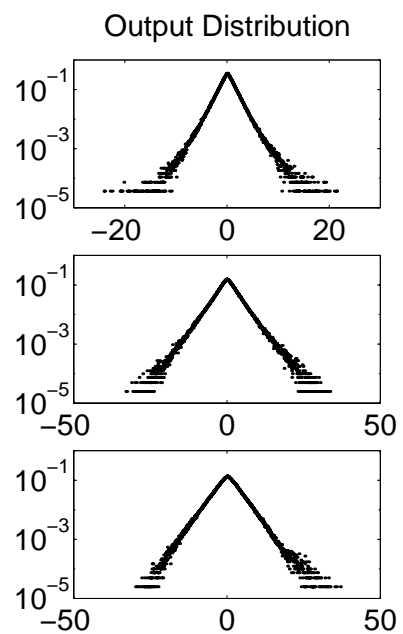
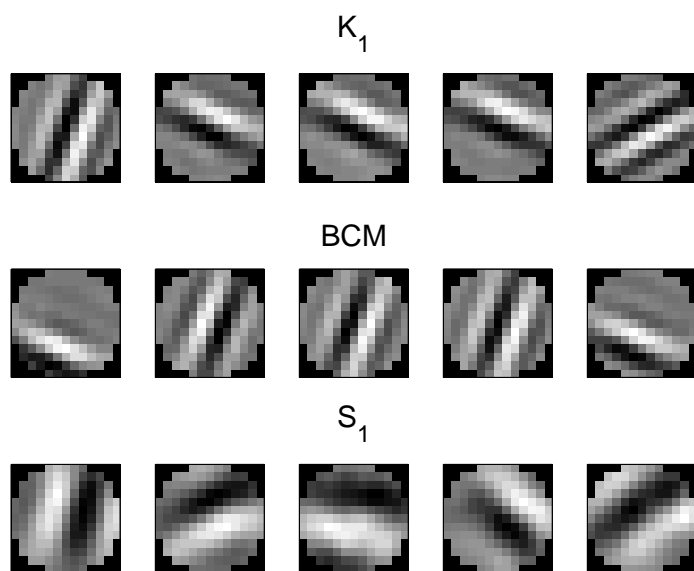
$$\nabla S_1 = \frac{1}{\Theta_M^{1.5}} E [c (c - E[c^3]/\Theta_M) \sigma' \mathbf{d}] = \frac{d\mathbf{m}}{dt}.$$

# Gradients of Energy Functions



# Receptive Fields from Natural Scene Input: DOGed

---

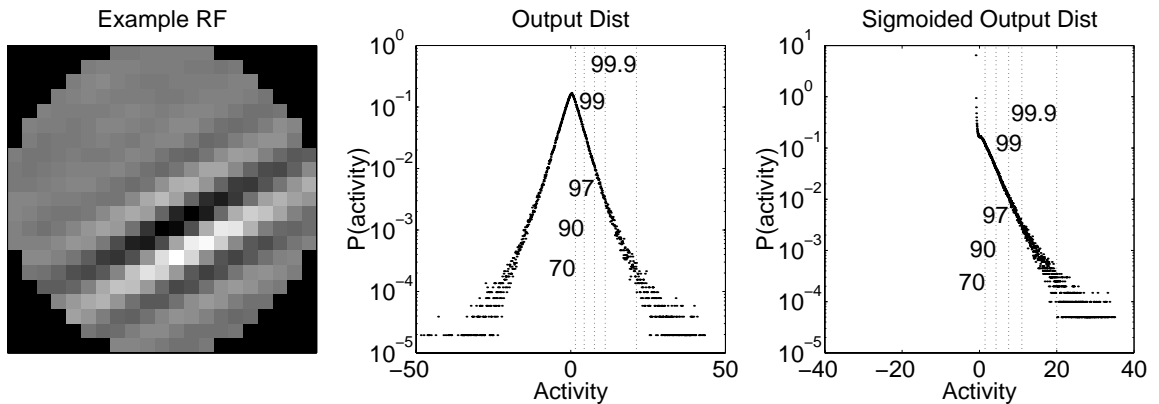


## Structure Removal

---

- Learning rules based on polynomial moments are sensitive to outliers
- Friedman's structure removal transforms the data such that the projected distribution is gaussian: eliminates information along that projection
- New form of structure removal eliminates those patterns which contain structure, defined as patterns yielding high responses
  - does not require whitened data
  - allows for partial amount of structure to be removed

# Structure Removal Example



## Example Patches Removed from Environment

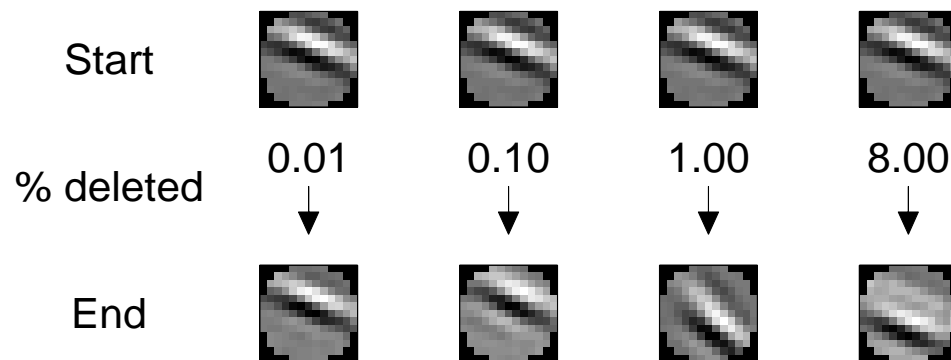


- removal is done such that the neuron *never sees* a forbidden patch: if such a patch is randomly chosen at any time, the neuron simply chooses another
- Sensitivity measure: change in RF vs. number of patterns deleted
- Hierarchy of receptive fields by successive structure removal

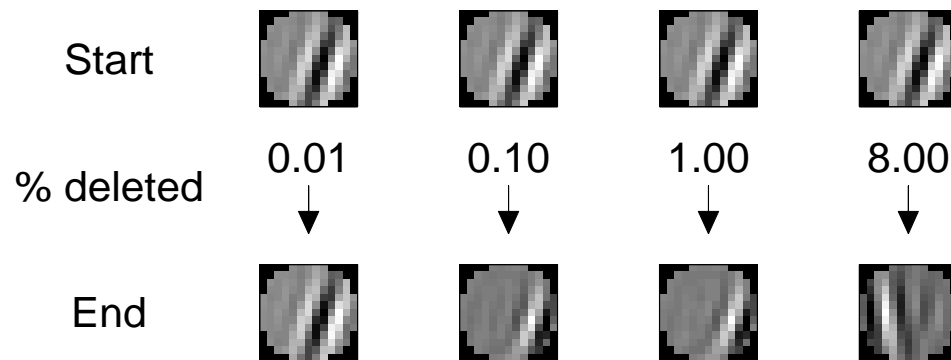
# Measuring Sensitivity using Structure Removal

---

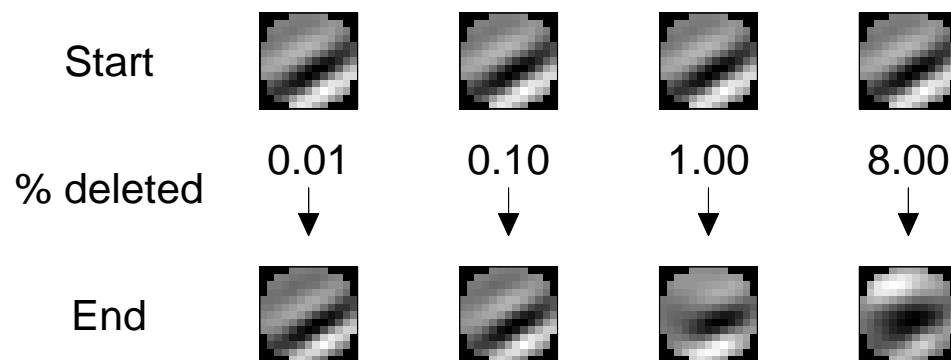
- Kurtosis ( $K_1$ )



- BCM

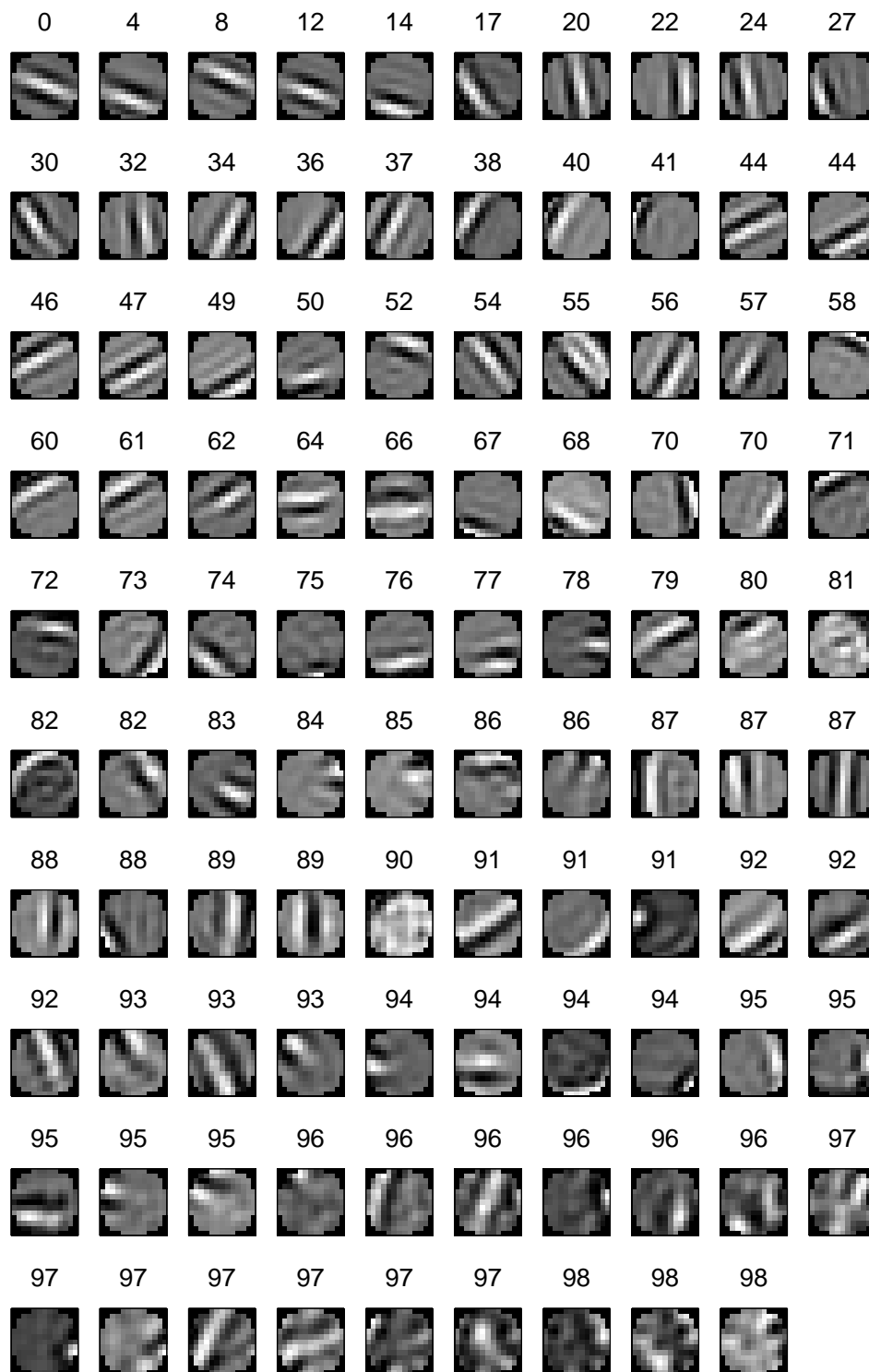


- Skewness ( $S_1$ )



# Repeated Structure Removal for BCM

---



## Conclusions

---

- How sparse is sparse?
  - Kurtosis and BCM sensitive  $\sim 1/10 - 1/2$  percent
  - Skewness sensitive  $\sim 1 - 2$  percent
- Structure removal can help us explore some of the information in the natural scene environment