Problem Set: CCCN 2019

Ann Hermundstad
hermundstada@janelia.hhmi.org

August 2019

1 Refresher on random variables and information theory

1.1

The mean of a random variable X is given by its expected value, $E[X]$. Similarly, the variance of a random variable is given by the expected value of the squared deviation from the mean, $\text{Var}(X) = E[(X - E[X])^2]$. Show that the variance can be equivalently written as $\text{Var}(X) = E[X^2] - E[X]^2$.

1.2

Let X define stimuli drawn randomly from a Gaussian distribution, $X \sim \mathcal{N}(0, \sigma_X^2)$. Assume that these stimuli are corrupted by additive Gaussian noise $Z \sim \mathcal{N}(0, \sigma_Z^2)$. The resulting signal can be written as another random variable $Y_i = X_i + Z_i$. Find an expression for $\text{Var}(Y)$ in terms of σ_X^2 and σ_Z^2.

1.3

The mutual information between stimuli $s \in S$ and neural responses $r \in R$ is given by:

$$I(R; S) = \sum_{r \in R} \sum_{s \in S} p(r, s) \log_2 \left(\frac{p(r, s)}{p(r)p(s)} \right).$$

Show that this can be equivalently written as $I(R; S) = H(R) - H(R|S)$, where

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$$

is the entropy of X, and

$$H(X|Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 \left(\frac{p(x, y)}{p(y)} \right)$$

is the conditional entropy of X given Y. Convince yourself that if the response is completely determined by the stimulus, the mutual information is equivalent to the entropy of the response distribution.

1.4

Consider a deterministic encoder that can produce $N = 2$ discrete response values (i.e., this encoder takes an incoming stimulus s and produces one of two responses, r_1 or r_2). Show that this encoder maximizes information transmission when $p(r_1) = p(r_2) = 1/2$. How much information does the encoder transmit in this case?

1.5

The result you derived in the previous problem holds more generally; a deterministic encoder that produces N discrete responses r_i, $i \in [1, N]$, conveys a maximum of $\log_2(N)$ bits of information when $p(r_i) = 1/N$ for all i. Assign discrete spike counts k (with $k \in [0, N-1]$) to the discrete outputs of this optimal encoder. What assignment minimizes the average spike count of the output? Now imagine that you design a suboptimal encoder that produces response r_1 with probability $p_1 = 1/3$, response r_2 with probability $p_2 = 1/3$, and all other responses r_i with probability $p_i = 1/[3(N-2)]$. What assignment of spike counts minimizes the average spike count of the output?

2 Optimizing nonlinear encoders

*2.1 Construct a nonlinear encoder.

Now, consider an encoder that transforms incoming stimuli into discrete spike counts via a saturating nonlinearity. Write a function that performs this transformation. First, write a function that maps a stimulus onto a continuous value between 0 and 1 via a sigmoidal nonlinearity with slope k and offset x_0. Then, discretize the output of this nonlinearity into N evenly spaced values (try using $N = 8$ as a start; you can always change this later). Finally, map these output values onto discrete spike counts. The result should be a function that takes a continuously-valued stimulus as an input, and produces one of N discrete spike counts as an output, based on the parameters k and x_0.

*2.2 Generate a stimulus distribution.

Consider an environment that that generates stimuli from a Gaussian distribution with mean μ and variance σ^2. Write a function to generate stimuli from this distribution. For the time being, use $\mu = 0$ and $\sigma = 1$ (you'll vary these shortly).

*2.3 Visualize the output of your encoder.

Plot the output of your encoder for a range of stimuli drawn randomly from your stimulus distribution. Superimpose the sigmoidal nonlinearity that you generated above (note that you will have to rescale this nonlinearity based on the maximum spike count produced by your encoder).

*2.4 Maximize the output entropy of your encoder.

Determine the parameters of the nonlinearity that maximize the entropy of the distribution of spike counts, given this stimulus distribution as an input. Do this in multiple steps to better understand the results: (1) Generate two vectors of parameters, one for possible values of k, and another for possible values of x_0. (2) For each combination of k and x, determine the entropy of the response distribution. You'll need to write a function to compute this entropy from the output of the encoder. Store these values in a matrix $H(k, x_0)$. (3) Plot H (this should be an "entropy surface" computed over different values of k and x_0); do the results make sense? How many maxima are there? (4) Find the parameter values corresponding to each maximum, and determine the values of these maxima. Are these maximum entropy values consistent with what you would have expected given the discreteness of your encoder? (5) Compute the average "firing rate", or output spike count, of your encoder. Does this value make sense?

2.5 Minimize reconstruction error.

Maximizing the information that neural responses convey about the stimulus distribution says nothing about whether, or how, an estimate of the stimulus should be recovered from the neural responses. For this, we need to decode an estimate of the stimulus, $\hat{s}$, from the response. Design a simple linear decoder that minimizes the reconstruction error, $(s - \hat{s})^2$, when averaged over the stimulus distribution (we will denote this average with angle brackets, i.e. $\langle(s - \hat{s})^2\rangle$). Check that your decoder produces something sensible by plotting the reconstructed stimuli against their true values.

Now, use this decoder to find the parameters of the nonlinearity that minimize the average reconstruction error $\langle(s - \hat{s})^2\rangle$. As before, construct an "error surface" $E(k, x_0)$ that specifies the average reconstruction error as a function of different parameter combinations. Plot this error surface (try plotting the error on a log scale); how many minima are there? Find the parameter values corresponding to each minimum.

Finally, use these optimal parameters to encode your distribution of stimuli. What is the entropy of the response distribution? How does this compare to the value you obtained when you directly maximized the entropy? What is the average firing rate, and how does this compare the value you found earlier? Plot the reconstruction error $(s - \hat{s})^2$ as a function of s. Which stimuli are encoded with highest accuracy?

2.6 Vary the stimulus distribution.

The parameters of your optimal nonlinearity depend on the properties of your stimulus distribution (why does it make sense for these to depend on each other?). Intuitively, how do you expect these parameters to change as you vary the mean and variance of your Gaussian stimulus distribution? Sketch this out. To test your intuition, vary the mean μ of your stimulus distribution (keeping the variance fixed), and compute the optimal values of k and x_0 as a function of μ. Plot your results; from these, can you deduce expressions for k and x_0 in terms of μ?

Repeat this, now changing the variance σ^2 of your stimulus distribution while keeping the mean fixed. Plot the optimal values of k and x_0 as a function of σ^2. As before, can you deduce expressions for k and x_0 in terms of σ^2? To help with this, try plotting your results on a log-log scale.

2.7 Add input noise.

Consider your original Gaussian stimulus distribution with mean $\mu = 0$ and variance $\sigma = 1$. Now add Gaussian noise of varying power σ_{in}^2 to the stimulus. We will call this input noise, because it is added to the stimulus before passing it through the encoding nonlinearity. Based on the results of the previous problem, how do you expect the parameters of the optimal nonlinearity to scale with noise power? Check this intuition by finding the optimal nonlinearity parameters as a function of σ_{in}^2.

2.8 Add output noise.

Instead of adding input noise directly to the stimulus, add noise the output of your nonlinear encoder, before your decoding step. Use Gaussian noise of varying power σ_{out}^2. How do the optimal parameters of your nonlinearity depend on σ_{out}^2? Compare this to the results you found for adding the same amount of noise at the input.

2.9 Explore implications for adaptation.

You should have found earlier that the offset of the optimal nonlinearity is aligned with the mean of the Gaussian stimulus distribution (i.e., $x_0^* = \mu$), and the slope is inversely related to the standard deviation (i.e., $k^* = 1/\sigma$) (if you didn't find this, go back and check your earlier results).

The optimization that you used to you determine these parameters was performed under the assumption that the system "knows" the true input distribution and can tune the parameters of the encoder accordingly. Instead, assume that the system maintains a *belief*, or expectation, that incoming stimuli are drawn from a Gaussian distribution with mean $\hat{\mu}$ and variance $\hat{\sigma}^2$. Your earlier results still hold; in this case, the system should optimally choose the parameters of its nonlinearity based on its belief; i.e., $x_0 = \hat{\mu}$ and $k = 1/\hat{\sigma}$. In general, this belief might not be accurate; in other words, $\mu \neq \hat{\mu}$, $\sigma^2 \neq \hat{\sigma}^2$. This difference between true and expected properties of the stimulus distribution can lead to large differences in the output of the encoder.

To get a feel for this, consider a scenario in which the variance of the Gaussian stimulus distribution is fixed and known to the system (i.e., $\sigma^2 = \hat{\sigma}^2 = 1$), but the mean can change in time. Compute the average reconstruction error $\langle (s - \hat{s})^2 \rangle$ for different values of the true mean μ, assuming the system maintains a given estimate of the mean $\hat{\mu}$. Repeat this for different values of $\hat{\mu}$ to construct an error surface, $E(\mu, \hat{\mu})$. Compute two related matrices, one that measures the average spike counts, $K(\mu, \hat{\mu})$, and another that measures the entropy of the spike count distribution, $H(\mu, \hat{\mu})$. Are these matrices symmetric? These symmetries tell you how the system responds to overestimating versus underestimating the mean of incoming stimuli.

Repeat for the scenario in which the mean of the Gaussian stimulus distribution is fixed and known to the system (i.e., $\mu^2 = \hat{\mu}^2 = 1$), but the variance can change in time. Do these matrices share the same symmetries as those that you found in the variable-mean case? Why or why not?

Suggested reading

[1] Claude E Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, 1948.

[2] Fred Attneave. Some informational aspects of visual perception. *Psychological Review*, 61(3):183–193, 1954.

[3] Horace B Barlow. Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1:217–234, 1961.

[4] Simon Laughlin. A simple coding procedure enhances a neuron's information capacity. *Zeitschrift für Naturforschung c*, 36(9-10):910–912, 1981.

[5] Mandyam Veerambudi Srinivasan, Simon Barry Laughlin, and Andreas Dubs. Predictive coding: a fresh view of inhibition in the retina. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 216(1205):427–459, 1982.

[6] David J Field. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, 4(12):2379–2394, 1987.

[7] J H van Hateren. A theory of maximizing sensory information. *Biologial Cybernetics*, 68:23–29, 1992.

[8] Joseph J Atick and A Norman Redlich. What does the retina know about natural scenes? *Neural Computation*, 4:196–210, 1992.

[9] Dawei W Dong and Joseph J Atick. Statistics of natural time-varying images. *Network: Computation in Neural Systems*, 6(3):345–358, 1995.

[10] Yang Dan, Joseph J Atick, and R Clay Reid. Efficient coding of natural scenes in the lateral geniculate nucleus: Experimental test of a computational theory. *The Journal of Neuroscience*, 16(10):3351–3362, 1996.

[11] Anthony J Bell and Terrence J Sejnowski. An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1004–1004, 1995.

[12] Bruno A Olshausen and David J Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607, 1996.

[13] Anthony J Bell and Terrence J Sejnowski. The "independent components" of scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

[14] J H van Hateren and D L Ruderman. Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proc. R. Soc. Lond. B*, 265:2315–2320, 1998.

[15] Naama Brenner, William Bialek, and Rob de Ruyter Van Steveninck. Adaptive rescaling maximizes information transmission. *Neuron*, 26(3):695–702, 2000.

[16] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[17] Adrienne L Fairhall, Geoffrey D Lewen, William Bialek, and Robert R de Ruyter van Steveninck. Efficiency and ambiguity in an adaptive neural code. *Nature*, 412(6849):787, 2001.

[18] Eero P Simoncelli and Bruno A Olshausen. Natural image statistics and neural representation. *Annual Review of Neuroscience*, 24:1193–1216, 2001.

[19] Christian K Machens, Tim GOllisch, Olga Kolesnikova, and Andreas V M Herz. Testing the efficiency of sensory coding with optimal stimulus ensembles. *Neuron*, 47(6):447–456, 2005.

[20] William Bialek, Rob R De Ruyter Van Steveninck, and Naftali Tishby. Efficient representation as a design principle for neural coding and computation. In *2006 IEEE international symposium on information theory*, pages 659–663. IEEE, 2006.

[21] Evan C Smith and Michael S Lewicki. Efficient auditory coding. *Nature*, 439(23):978–982, 2006.

[22] Gasper Tkacik, Jason S Prentice, Vijay Balasubramanian, and Elad Schneidman. Optimal population coding by noisy spiking neurons. *Proceedings of the National Academy of Sciences*, 107(32):14419–14424, 2010.

[23] Neil C Rabinowitz, Ben DB Willmore, Jan WH Schnupp, and Andrew J King. Contrast gain control in auditory cortex. *Neuron*, 70(6):1178–1191, 2011.

[24] Yan Karklin and Eero Simoncelli. Efficient coding of natural images with a population of noisy linear-nonlinear neurons. *Adv Neural Inf Process Syst*, pages 999–1007, 2011.

[25] David B Kastner and Stephen A Baccus. Coordinated dynamic encoding in the retina using opposing forms of plasticity. *Nature neuroscience*, 14(10):1317, 2011.

[26] Josh H McDermott, Michael Schemitsch, and Eero P Simoncelli. Summary statistics in auditory perception. *Nature Neuroscience*, 16(4):493–498, 2013.

[27] Eizaburo Doi and Michael S Lewicki. A simple model of optimal population coding for sensory systems. *PLoS Computational Biology*, 10(8):e1003761, 2014.

[28] Ann M Hermundstad, John J Briguglio, Mary M Conte, Jonathan D Victor, Vijay Balasubramanian, and Gasper Tkacik. Variance predicts salience in central sensory processing. *eLife*, 3:e03722, 2014.

[29] Deep Ganguli and Eero P Simoncelli. Efficient sensory encoding and bayesian inference with heterogeneous neural populations. *Neural computation*, 26(10):2103–2134, 2014.

[30] Xue-Xin Wei and Alan A Stocker. A bayesian observer model constrained by efficient coding can explain'anti-bayesian'percepts. *Nature neuroscience*, 18(10):1509, 2015.

[31] Stephanie E Palmer, Olivier Marre, Michael J Berry, and William Bialek. Predictive information in a sensory population. *Proceedings of the National Academy of Sciences*, 112(22):6908–6913, 2015.

[32] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv:1703.00810*, 2017.

[33] Il Memming Park and Jonathan W Pillow. Bayesian efficient coding. *bioRxiv*, page 178418, 2017.

[34] Wiktor F Młynarski and Ann M Hermundstad. Adaptive coding for dynamic sensory inference. *Elife*, 7:e32055, 2018.

[35] Matthew Chalk, Olivier Marre, and Gašper Tkačik. Toward a unified theory of efficient, predictive, and sparse coding. *Proceedings of the National Academy of Sciences*, 115(1):186–191, 2018.

[36] Lajos Vago and Balazs B Ujfalussy. Robust and efficient coding with grid cells. *PLoS Computational Biology*, 14(1):e1005922, 2018.

[37] Christopher J Cueva and Xue-Xin Wei. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. *arXiv:1803.07770*, 2018.

[38] Banino et al. Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557:429–433, 1996.

[39] Julijana Gjorgjieva, Markus Meister, and Haim Sompolinsky. Functional diversity among sensory neurons from efficient coding principles. *bioRxiv*, page 131946, 2019.

[40] Wiktor F Młynarski and Ann M Hermundstad. Adaptability and efficiency in neural coding. *bioRxiv*, page 669200, 2019.