Information & Coding in the Brain

Prof. Yoram Burak

Nischal Mainali

Contents

§1 Introduction

In the modern dogma of Neuroscience, one of the central tenet is that there are representations of external and latent internal variables in the brain. Indeed, Neuroscientists often work with an operational belief that the information is represented in neural activities and that the brain can be fundamentally understood via the lens of representations, and computations over the representations.

These representations, as alluded earlier, come in two flavors: external and internal. The external representations are the ones that are directly related to the external world, and are often referred to as sensory representations. The internal representations are the ones that are related to the internal states of the organism, and are often referred to as cognitive representations. Representation of orientation in the scene by the visual area, and representation of frequencies of auditory signals are examples of external representations. Meanwhile, non-sensory information such as spatial locations in place and grid cells are examples of internal, cognitive representations.

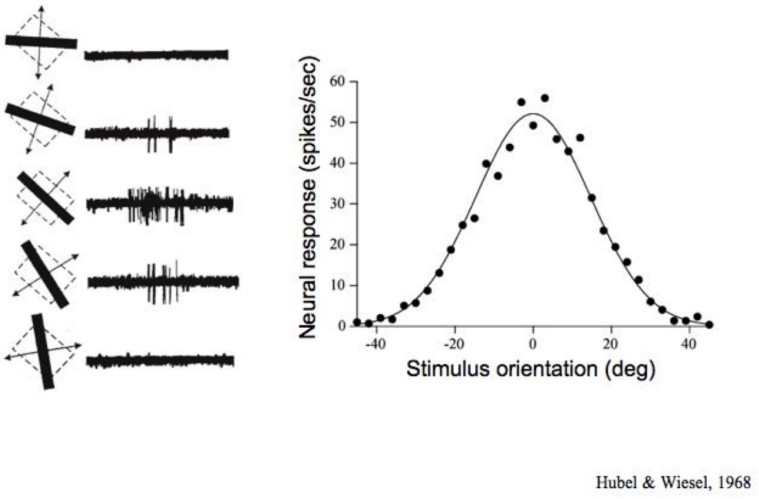

Figure 1: Orientation Tuning in V1

Above you can see a representative example of a neuron tuned to a particular orientation in the visual scene. These neurons form a pinwheel structure and lie in a column of neurons in the visual cortex and together the neural code charts the whole parameter space i.e, visual angle $\Theta \in [0, 2\pi]$.

Now as will be the theme in this course we are ready to ask an important question:

How well can the representation be readout either by the brain itself or by neuroscientists recording the neural activity?

Indeed, we will adopt some sort of normative notions to ask this question: Given some notion of precision, what determines the readout precision? This might be the inherent property of the (model of) neural activity, noise in the system, or the way the readout is done. We will look at the impact of these factors on the readout precision.

§1.1 Neural Variability and Decoding

It is a readily observable fact that neurons are noisy. The noise in the neural activity is often referred to as neural variability. The variability in the neural activity can be due to a number of factors. One of the most important factor is the intrinsic noise in the neuron. The intrinsic noise is the noise that is due to the stochastic nature of the ion channels and inherent biological processes in the neuron. The other important factor is the noise in the sensory input. The sensory input is often noisy and the noise in the sensory input is often referred to as sensory or input noise. Finally, the noise in the neural activity can also be due to the noise in the readout. The readout noise is the noise that is due to the fact that the readout, either by the brain itself or by the neuroscientists, is not perfect. More precisely, if we refer r_i as the neural activity of the i^{th} neuron, then the given some noise in the neural activity, we have a distribution over its value:

$$\mathbb{P}\left(r_i \mid \theta\right) \tag{1.1}$$

Later we will for example see an example of Poisson neurons whose mean $\Lambda(\theta)$ is parametrized by the stimulus and the noise is captured by the Poisson distribution. And the situation is that given some P observations of the neural activity, we have to estimate the stimulus θ. To set the notation, we will denote the estimate as $\hat{\theta}$.

More generally, both the brain and neuroscientists might want to infer the stimulus from observing more than one neuron.

Given N neurons: $\mathbf{r} = (r_1, \ldots, r_N)^\mathsf{T}$, we have a distribution of the neural activity vector: $\mathbb{P}\left(\mathbf{r} \mid \theta\right)$. And with P observations of the neural activity, we have to estimate the stimulus θ.

Alternatively, we might need to discriminate rather than estimate beween two possible stimuli. We will start our discussion from the simpler case of discrimination and then

move to estimation. To wit, we will start with a discussion of general statistical decision theory with the question of neural discrimination and estimation in the back of our mind.

§2 Statistical Decision Theory

We will start with a probabilistic framework for making decisions, and we will try to illuminate concepts and ideas by relating to neural example where neurons respond to stimulus from the world.

For example, neurons might be presented with a stimulus θ_1. Given the stimulus, neurons might respond probabilistically. Given the firing of the neurons, a downstream area might have to take an action: say press left or right. But because of the probabilistic nature of the neural response to the stimulus, the downstream area might not be able to take a deterministic action. Instead, it might have to take a probabilistic action. Statistical Decision Theory helps us formulate this problem of inference from noisy observation and taking an action based on the inference via the following objects:

1. **World States** The set of possible world states (stimulus) Ω.

   Example

   Discrete $\Omega = \{\theta_1, \theta_2\}$; Continuous $\Omega = [0, 2\pi]$.

   We assume that these span all the possible states of the world (i.e., in each possible world exactly one of these is the state).

2. **Observations** Some property that we get to observe R (neural firing).

   Example

   Could have continuous values (e.g., $R = \mathbb{R}$), multidimensional (e.g., $R = \mathbb{R}^n$) or discrete (e.g., $R = \{0, 1\}$).

3. **Possible actions** The set of actions we can take A.

   Example

   $A = \{\text{press left}, \text{press right}, \text{don't act}\}$.

4. **Risk function (or loss function)**

   Example

   Whether our action was good or not depends on what the real state of the world (stimulus) is (which we do not know before we carry out the action). In an experimental setting for an animal, pressing right when the stimulus is θ_1 might bt rewarding, but pressing right when the stimulus is θ_2 might not.

   Formally we define a function $\lambda(\alpha, \theta) \in \mathbb{R}$ which gives the price we pay for action α if the state of the world is θ.

So far we said nothing about probabilities. The first two items are actually random variables, and they have corresponding distributions.

A probabilistic model:

Assume we know the distributions $\mathbb{P}(\theta)$ and $\mathbb{P}(r \mid \theta)$ for $r \in R$. The distribution $\mathbb{P}(\theta)$ is called the prior over θ. In some cases it is natural to assume we know it while in others it might not be as sensible. Approaches which assume the priors are called Bayesian approaches.

The goal of decision theory is to decide how to act given a measurement $r \in R$. We are interested in a **decision function (or a strategy)** $\delta : R \to A$. There are several criteria for deciding what the best strategy is but here we focus on the Bayesian approach.

§2.1 Bayesian Decision Theory

How do we (or some readout in the brain) chooses a "good" decision function? Had we known what the state of the world θ is, there would be no problem. Clearly we would just choose the action α that minimized the cost $\lambda(\alpha, \theta)$. But we don't. The state of the world is *unobserved*.

If we take action α every time we see an observation r we will incur a different cost every time, since θ may have different values every such time (due to the probabilistic nature of the neural code). Instead, we can calculate **the expected cost** of taking action α for a given observation r.

$$\mathfrak{R}\left[\alpha \mid r\right] = \sum_{\theta} \lambda(\alpha, \theta)\mathbb{P}(\theta \mid r) \tag{2.1}$$

But how do we know $\mathbb{P}(\theta \mid r)$? We don't. We only know $\mathbb{P}(\theta)$ and $\mathbb{P}(r \mid \theta)$. So we can use the Bayes rule that states:

$$\mathbb{P}(\theta \mid r) = \frac{\mathbb{P}(r \mid \theta)\mathbb{P}(\theta)}{\mathbb{P}(r)} \tag{2.2}$$

$$\implies \mathfrak{R}\left[\alpha \mid r\right] = \sum_{\theta} \lambda(\alpha, \theta)\frac{\mathbb{P}(r \mid \theta)\mathbb{P}(\theta)}{\mathbb{P}(r)} \tag{2.3}$$

The risk in (2.1) and (2.3) is defined per value of observation. We would like a decision rule for all values of r, so we would like to consider the risk when averaged over all values of $r \in R$, namely the **overall risk**:

$$\mathfrak{R}\left[\delta\right] = \sum_{r} \mathfrak{R}\left[\delta \mid r\right]\mathbb{P}(r) = \sum_{r,\theta} \lambda(\delta(r), \theta)\mathbb{P}(r, \theta) \tag{2.4}$$

Then the optimal Bayes decision rule is defined as:

$$\delta^*(r) = \arg\min_{\delta(r)} \mathfrak{R}\left[\delta\right] \tag{2.5}$$

and the corresponding risk is $\mathfrak{R}\left[\delta^*\right]$ is the **Bayes risk**.

§3 Neural Discrimination

Suppose that our action is to predict the correct identity of the stimulus. This is clearly a special case of the situation we have considered above. Suppose we have K states that we want to discriminate between i.e, $\Omega = \{\theta_1, \theta_2, \ldots, \theta_K\}$, and we have a set of actions, which are nothing but our prediction for the stimulus. So, $\alpha = \{\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_K\}$. The risk function is defined as:

$$\lambda(\hat{\theta}, \theta) = \begin{cases} 0 & \text{if } \hat{\theta} = \theta \\ 1 & \text{otherwise} \end{cases} \tag{3.1}$$

Then, the conditional risk is:

$$\mathfrak{R}\left[\hat{\theta} \mid r\right] = \sum_{\theta} \lambda(\hat{\theta}, \theta)\mathbb{P}(\theta \mid r) = \sum_{\theta \neq \hat{\theta}} \mathbb{P}(\theta \mid r) = 1 - \mathbb{P}(\theta \mid r) \tag{3.2}$$

This implies that the optimal policy that minimizes risk would be to maximize the posterior probability of the correct stimulus. This is the **Bayes optimal policy**.

$$\delta^*(r) = \arg\max_{\theta} \mathbb{P}(\theta \mid r) \tag{3.3}$$

But often our observation is not based on a single observation but rather on a sequence of observations. A common situation is where we have n observations (of neural activity for example) $r_1, r_2, \ldots, r_n$ which we will refer as **samples** and we typically assume them to be i.i.d (independent and identically distributed). In this case, the posterior probability of the correct stimulus is:

$$\mathbb{P}(r_1, r_2, \ldots, r_n \mid \theta) = \prod_{i=1}^{n} \mathbb{P}(r_i \mid \theta) \tag{3.4}$$

We will be particularly interested in the role P in the precision of the decoder. Now, in this context, suppose we wanted to discrimate between two stimulus classes, θ_1 and θ_2. Then, the Bayes optimal policy would be to choose the stimulus class that maximizes the posterior probability of the correct stimulus:

$$\mathbb{P}(\theta_1 \mid r_1, r_2, \ldots, r_n) \gtrless \mathbb{P}(\theta_2 \mid r_1, r_2, \ldots, r_P) \tag{3.5}$$

$$\tag{3.6}$$

Equivalently, we can use the log-function and do some algebra to get:

$$\log\left[\frac{\mathbb{P}(\theta_1 \mid r_1, r_2, \ldots, r_P)}{\mathbb{P}(\theta_2 \mid r_1, r_2, \ldots, r_P)}\right] \gtrless 0 \tag{3.7}$$

$$\implies \frac{\log}{n}\left[\frac{\mathbb{P}(\theta_1)}{\mathbb{P}(\theta_2)}\right] + \frac{\log}{n}\left[\frac{\mathbb{P}(r_1, r_2, \ldots, r_P \mid \theta_1)}{\mathbb{P}(r_1, r_2, \ldots, r_P \mid \theta_2)}\right] \gtrless 0 \tag{3.8}$$

$$\implies \frac{\log}{n}\left[\frac{\mathbb{P}(\theta_1)}{\mathbb{P}(\theta_2)}\right] + \frac{1}{n}\sum_{i=1}^{n} \log\left[\frac{\mathbb{P}(r_i \mid \theta_1)}{\mathbb{P}(r_i \mid \theta_2)}\right] \gtrless 0 \tag{3.9}$$

$$\tag{3.10}$$

§3.1 The Log-Likelihood Ratio

Definition 3.1

The log-likelihood ratio is a random variable Z_i defined as:

$$Z_i = \log\left[\frac{\mathbb{P}(r_i \mid \theta_1)}{\mathbb{P}(r_i \mid \theta_2)}\right] \tag{3.11}$$

and

$$\hat{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i \tag{3.12}$$

which by the i.i.d property of z_i and law of large number converged to

$$\lim_{P\to\infty} \hat{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i = \mathbb{E}\left[Z\right] \tag{3.13}$$

Indeed, for a moment suppose that we were actually in the world state θ_1. Then:

$$\mathbb{E}\left[Z\right] = \sum_{r} \mathbb{P}(r \mid \theta_1) \log\left[\frac{\mathbb{P}(r \mid \theta_1)}{\mathbb{P}(r \mid \theta_2)}\right] \tag{3.14}$$

This is a specialy quantity called the **Kullback-Leibler divergence** between the two distributions $\mathbb{P}(r \mid \theta_1)$ and $\mathbb{P}(r \mid \theta_2)$. This is a measure of the difference between two probability distributions.

§3.2 Kullback-Leibler Divergence

Definition 3.2

The Kullback-Leibler divergence between two distributions $\mathbb{P}(r \mid \theta_1)$ and $\mathbb{P}(r \mid \theta_2)$ is defined as:

$$\mathfrak{D}_{KL}\left[\mathbb{P}(x) \mid\mid \mathbb{Q}(x)\right] = \sum_{x} \mathbb{P}(x) \log\left[\frac{\mathbb{P}(x)}{\mathbb{Q}(x)}\right] \tag{3.15}$$

As the number of samples become large $P \to \infty$, the log-likelihood ratio converges to the Kullback-Leibler divergence:

$$\lim_{P\to\infty} \hat{Z} = \mathbb{E}\left[Z\right] = \mathfrak{D}_{KL}\left[\mathbb{P}(r \mid \theta_1) \mid\mid \mathbb{P}(r \mid \theta_2)\right] \tag{3.16}$$

If we are in the world state θ_2, then as we will prove below the KL divergence is always positive, implying that in the limit $n \to \infty$ we will always make the right decision.

Theorem (Non-negativity of Kullback-Leibler Divergence)

$$\mathfrak{D}_{KL}\left[\mathbb{P}(z) \,||\, \mathbb{Q}(x)\right] \geq 0 \tag{3.17}$$

Proof. We will use the following identity:

$$\log x \leq x - 1 \tag{3.18}$$

Then,

$$-\mathfrak{D}_{KL}\left[\mathbb{P}(x) \,||\, \mathbb{Q}(x)\right] = -\sum_x \mathbb{P}(x) \log\left[\frac{\mathbb{P}(x)}{\mathbb{Q}(x)}\right] \tag{3.19}$$

$$\leq -\sum_x \mathbb{P}(x)\left[1 - \frac{\mathbb{Q}(x)}{\mathbb{P}(x)}\right] \tag{3.20}$$

$$\leq \sum_x \mathbb{P}(x) + \mathbb{Q}(x) = 0 \tag{3.21}$$

□

Another property of the KL divergence is that it is zero if and only if the two distributions are equal:

$$\mathfrak{D}_{KL}\left[\mathbb{P}(x) \,||\, \mathbb{Q}(x)\right] = 0 \iff \mathbb{P}(x) = \mathbb{Q}(x) \tag{3.22}$$

And finally that the KL divergence is generally not symmetric:

$$\mathfrak{D}_{KL}\left[\mathbb{P}(x) \,||\, \mathbb{Q}(x)\right] \neq \mathfrak{D}_{KL}\left[\mathbb{Q}(x) \,||\, \mathbb{P}(x)\right] \tag{3.23}$$

One quick and interesting way to observe is to check what happens when there is an x for which $\mathbb{P}(x) > 0, \mathbb{Q}(x) = 0$. Then we will have a divergence of ∞. This makes sense because it means we are looking at the likelihood ration between $\mathbb{P}(x)$ and $\mathbb{Q}(x)$ observe something that is infinitely more likely under $\mathbb{P}(x)$. But when $\mathbb{P}(x) = 0, \mathbb{Q}(x) > 0$ we will have a divergence of 0.

§4 Neural Estimation

If we think about it, we can realize that neural discrimination is a simplified take on a larger problem that our brains face: that of neural estimation. Neural Estimation becomes pertinent when there is some stimulus or variable that is represented by neural code and some downsetream readout area wants to "decode" or "estimate" that information. The cruicial difference here is that we go for a typically discrete case of discrimination to a case where some continuous variable is being represented by the neural code such as the orientations $\theta \in [0, 2\pi]$ or the position of an animal some 2D surface $\vec{p} \in \mathbb{R}^2$.

Continuiung the tradition, We shall consider the distribution by $\mathbb{P}(r \mid \theta)$. This is called a **parameterized family of distributions**, where the parameter is the world state θ, and r is the random variable denoting neural response. Generally, we are in a common situation where we have n observations (of neural activity for example) $r_1, r_2, \ldots, r_n$ which we will refer as **samples** that we typically assume them to be i.i.d, and our (or the brain readout area's) task is to estimate the world state θ. We will denote the estimate of θ as $\hat{\theta}$.

Example 4.1 (Non Neural Example: Coin Toss)

For every θ define x representing result of a coin toss such that:

$$\begin{aligned} \mathbb{P}(x = 1 \mid \theta) &= \theta \\ \mathbb{P}(x = 0 \mid \theta) &= 1 - \theta \end{aligned}$$

Now, given a sample of P coin tosses, we want to estimate the probability of heads, θ.

§4.1 Loss Function

Now, if we are to estimate θ we need to define a loss function. The loss function is a function of the estimate $\hat{\theta}$ and the true world state θ. The loss function is a measure of how bad our estimate is. The loss function is typically denoted as $\lambda(\hat{\theta}, \theta)$. The loss function is typically chosen to be a function of the difference between the estimate and the true world state.

A typical and popular choice for the loss function is the squared error loss function:

Definition 4.2

The squared error loss function is defined as:

$$\lambda(\hat{\theta}, \theta) = \left(\hat{\theta} - \theta\right)^2 \tag{4.1}$$

Claim

The optimal estimate given the loss function is the expected value of the posterior given a set of observation $r^n = r_1, r_2, \ldots, r_n$:

$$\hat{\theta} = \mathbb{E}\left[\theta \mid r^n\right] = \int \theta \mathbb{P}(\theta \mid r^n) d\theta \tag{4.2}$$

Proof. Bayesian decision theory tells us that the optimal estimate $\hat{(\theta)}$ is the one that minimizes the risk $\mathfrak{R}\left[\hat{\theta} \mid x\right]$. Since in our case α corresponds to $\hat{\theta}$ we write:

$$\mathfrak{R}\left[\hat{\theta} \mid r^n\right] = \int (\hat{\theta} - \theta)^2 \mathbb{P}(\theta \mid r^n) d\theta \tag{4.3}$$

Take derivative w.r.t. $\hat{\theta}$ to get the optimal estimate:

$$\begin{aligned} \int (\hat{\theta} - \theta) \mathbb{P}(\theta \mid r^n) d\theta &= 0 \\ \hat{\theta} \int \mathbb{P}(\theta \mid r^n) d\theta &= \int \theta \mathbb{P}(\theta \mid r^n) d\theta \\ \hat{\theta} &= \int \theta \mathbb{P}(\theta \mid r^n) d\theta \end{aligned}$$

□

Exercise. Check or argue that the critical point in the above proof is indeed a minimum.

§4.2 MAP and ML Estimator

What if we want our estimator $\hat{\theta}$ to equal θ exactly. In other words we want our loss to be zero if $\hat{\theta} = \theta$ and one otherwise. We already encountered this loss when talking about Bayesian decision theory,

$$\lambda(\hat{\theta}, \theta) = \begin{cases} 0 & \hat{\theta} = \theta \\ 1 & \hat{\theta} \neq \theta \end{cases} \tag{4.4}$$

Alternatively for the continuous case we can generalize to

$$\lambda(\hat{\theta}, \theta) = 1 - \delta(\hat{\theta} - \theta) \tag{4.5}$$

where δ is the Dirac delta function.

We already know that the optimal strategy in his case:

$$\hat{\theta}_{MAP}(x^n) = \arg\max_{\theta} \mathbb{P}(\theta | r^n) \tag{4.6}$$

Where MAP stands for maximum-aposteriori.

But intuitively, we'd expect the the prior to be increasingly less important as we get more and more data. In other words, we'd expect the posterior to be increasingly more

important as we get more and more data. This is the intuition behind the maximum-likelihood (ML) estimator. Concretely,

$$\begin{aligned} \mathbb{P}(\theta|r^n) &\propto \mathbb{P}(\theta)\mathbb{P}(r^n \mid \theta) \\ \log \mathbb{P}(\theta|r^n) &= \log \mathbb{P}(\theta) + \log \mathbb{P}(r^n \mid \theta) \\ \log \mathbb{P}(\theta|r^n) &= \log \mathbb{P}(\theta) + \sum_{i=1}^{n} \log \mathbb{P}(r_i \mid \theta) \end{aligned}$$

It is clear that as the number of samples grows the term $\log \mathbb{P}(\theta)$ has little effect on the posterior. This implies that the prior is negligible and we may think of considering maximizing only the second term. Note that it is non-Bayesian in that it does not assume a prior.

$$\hat{\theta}_{ML}(r^n) = \arg\max_{\theta} \mathbb{P}(r^n \mid \theta) \tag{4.7}$$

§4.3 Consistency

As we get more and more data, we would expect the random variable $\hat{\theta}(r^n)$ to get close to θ. If an estimator has this property it is said to be consistent.

Definition

An estimator is consistent if for every value of θ the following holds in probability for samples drawn from $\mathbb{P}(r^n \mid \theta)$:

$$\hat{\theta}(r^n) \to \theta \tag{4.8}$$

Claim

The ML estimator is consistent.

Proof.

$$\begin{aligned} \lim_{n\to\infty} \hat{\theta}(r_n) &= \lim_{n\to\infty} \arg\max_{\theta'} \frac{1}{n} \log \mathbb{P}(r^n \mid \theta') \\ &= \arg\max_{\theta'} \lim_{n\to\infty} \frac{1}{n} \log \mathbb{P}(r^n \mid \theta') \\ &= \arg\max_{\theta'} \lim_{n\to\infty} \frac{1}{n} \log \mathbb{P}(r^n \mid \theta') = \arg\max_{\theta'} \lim_{n\to\infty} \frac{1}{n} \sum_i \mathbb{P}(r_i \mid \theta') \\ &= \arg\max_{\theta'} \sum_r \mathbb{P}(r \mid \theta) \log \mathbb{P}(r \mid \theta') \\ &= \arg\max_{\theta'} \left[-\mathfrak{D}_{KL}\left[\mathbb{P}(r \mid \theta) \,||\, \mathbb{P}(r \mid \theta')\right] + \sum_x \mathbb{P}(r \mid \theta) \log \mathbb{P}(r \mid \theta) \right] \\ &= \arg\max_{\theta'} -\mathfrak{D}_{KL}\left[\mathbb{P}(r \mid \theta) \,||\, \mathbb{P}(r \mid \theta')\right] = \theta \end{aligned}$$

□

Indeed, if θ_0 is the actual value of the world state, then this leads us to an interesting observation that:

$$\begin{aligned}\frac{1}{n}\mathbb{E}_{\mathbb{P}(r^n|\theta_0)} \log \mathbb{P}(r^n \mid \theta) &= \frac{1}{n}\mathbb{E}\left(\sum_i \log \mathbb{P}(r^i \mid \theta)\right) = \mathbb{E}\left[\log \mathbb{P}(r \mid \theta)\right] \\ &= -\mathfrak{D}_{KL}\left[\mathbb{P}(r \mid \theta_0) \,||\, \mathbb{P}(r \mid \theta)\right] + \sum_r \mathbb{P}(r \mid \theta_0) \log \mathbb{P}(r \mid \theta_0)\end{aligned}$$

implying that maximum likelihood is the distribution in the family that is closely (in the KL sense) to the empirical distribution.

§4.4 The Bias Variance Decomposition

Zooming out a little, we started by really caring about $\left(\theta - \hat{\theta}\right)^2$ for an estimator.

$$\mathfrak{R}\left[\hat{\theta} \mid \theta\right] = \mathbb{E}\left[\left(\theta - \hat{\theta}\right)^2\right] \tag{4.9}$$

We can try to understand this term by decomposing it into parts that we can more intuitively understand:

$$\begin{aligned}&\mathbb{E}\left[\left(\hat{\theta} - \theta\right)^2\right] = \mathbb{E}\left[\left(\hat{\theta} - \mathbb{E}\left[\hat{\theta}\right] + \mathbb{E}\left[\hat{\theta}\right] - \theta\right)^2\right] \\ = \;& \mathbb{E}\left[\left(\hat{\theta} - \mathbb{E}\left[\hat{\theta}\right]\right)^2\right] - 2\mathbb{E}\left[\left(\hat{\theta} - E(\hat{\theta})\right)\left(\mathbb{E}\left[\hat{\theta}\right] - \theta\right)\right] + \mathbb{E}\left[\left(\mathbb{E}\left[\hat{\theta}\right] - \theta\right)^2\right] \\ = \;& \mathbb{E}\left[\left(\hat{\theta} - \mathbb{E}\left[\hat{\theta}\right]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}\left[\hat{\theta}\right] - \theta\right)^2\right] \\ = \;& \mathbb{E}\left[\left(\hat{\theta} - \mathbb{E}\left[\hat{\theta}\right]\right)^2\right] + \left(\mathbb{E}\left[\hat{\theta}\right] - \theta\right)^2 \\ = \;& \mathsf{Var}[\hat{\theta} \mid \theta] + \mathsf{Bias}[\hat{\theta} \mid \theta]^2\end{aligned}$$

where we have defined the bias and variance of an estimator as:

$$\mathsf{Var}[\hat{\theta} \mid \theta] = \mathbb{E}\left[\left(\hat{\theta} - \mathbb{E}\left[\hat{\theta}\right]\right)^2\right] \tag{4.10}$$

$$\mathsf{Bias}[\hat{\theta} \mid \theta] = \mathbb{E}\left[\hat{\theta}\right] - \theta \tag{4.11}$$

Bias, in this context, as is clear from the expression denotes how much our estimator deviates from the true value of the parameter. Meanwhile, variance captures that idea that our estimator is itself a random variable and might change with different set of samples. Of course, we want the bias to be small and the estimate to be similar even if the samples change implying that variance should also be small. Another (rather significant) observation is that an estimator has two routes to get to some small value

of the mean squared error: it can either have a small bias or a small variance but can't make both arbitrarily small. This is a fundamental trade off in estimation and some remarks are in order:

1. Generally restricting the size of the possible θ set will tend to increase variance but reduce bias.
2. Generally increasing the number of samples using the same estimator will reduce variance so that we can use a lower bias estimator.
3. Biased estimators may not be consistent asymptotically but can work better than consistent estimators for finite sample sizes.

So, it seems like if we constrain the values of θ that we consider, we will lower the variance but probably increase bias. If our constraint is met in reality we will have lowered both bias and variance.

§5 Fisher Information

Let's continuue our discussion with only estimators that are unbiased. There is no mathematical force ebhind thhis choihce, but simply our attempt at understanding estimation better and simplying the context to achieve that. An estimator is unbiased if:

$$\mathbb{E}\left[\hat{\theta}\right] = \theta \tag{5.1}$$

$$\text{In Particular, } \mathsf{Bias}[\hat{\theta} \mid \theta] = \mathbb{E}\left[\hat{\theta}\right] - \theta = 0 \tag{5.2}$$

Now, in these cases, we can try to quantify the idea of a robust estimator or an estimator wiuht small variance. Intuitively, the variance must deoend on how the parametric distribution $\mathbb{P}(r \mid \theta)$ changes with θ. Why? Because, if the distribution changes a lot with θ, then the samples will be very different even if we sample from the same distribution. This is because the distribution is very sensitive to θ. On the other hand, if the distribution is not very sensitive to θ, then the samples will be very similar even if we sample from the same distribution. This is because the distribution is not very sensitive to θ. If the samples change a lot with changes in the parameter, we will be able to more easily identify the parameter, so we'd expect some inverse relation between the variance and the sensitivity of the distribution to the parameter.

We can have a first go at capturing this idea by defining the score function:

$$S(\theta) = \frac{\partial \log \mathbb{P}(r \mid \theta)}{\partial \theta} = \nabla_\theta \log \mathbb{P}(r \mid \theta) \tag{5.3}$$

But turns out that score is not a very good estimate of the changes of the parametric distribution because in expectation its value is zero.

Claim

$$\mathbb{E}\left[S(\theta)\right] = 0 \tag{5.4}$$

Proof.

$$\begin{aligned}
\mathbb{E}\left[S(r,\theta)\right] &= \int \nabla_\theta \log \mathbb{P}(r \mid \theta) \cdot \mathbb{P}(r \mid \theta) dr \\
&= \int \frac{1}{\mathbb{P}(r \mid \theta)} \nabla_\theta \mathbb{P}(r \mid \theta) \cdot \mathbb{P}(r \mid \theta) dr \\
&= \int \nabla_\theta \mathbb{P}(r \mid \theta) dr \\
&= \nabla_\theta \int \mathbb{P}(r \mid \theta) dr \\
&= \nabla_\theta 1 = 0
\end{aligned}$$

□

So, we need to find a better way to quantify the sensitivity of the distribution to the parameter. We can do this by defining the Fisher Information, which is the variance of the score function:

Definition (Fisher Information)

$$J(\theta) = \int \left[\nabla_\theta \log \mathbb{P}(r \mid \theta)\right]^2 \mathbb{P}(r \mid \theta) dr = \mathsf{Var}[S(\theta)] \tag{5.5}$$

$$\text{Alternatively, } J(\theta) = \int -\nabla_\theta^2 \log \mathbb{P}(r \mid \theta) \cdot \mathbb{P}(r \mid \theta) dr \tag{5.6}$$

Indeed, fisher information captures non trivial information about the distribution and is closely related to estimation as we will shortly see. Before that, let's note some properties of the fisher information:

1. Fisher information is always positive: $\forall \theta \in \Theta, J(\theta) \geq 0$
2. Fisher information is additive for i.i.d samples: $J(\theta) = \sum_{i=1}^{n} J_{\mathcal{I}}(\theta) = n \cdot J_{\mathcal{I}}(\theta)$
3. Fisher information obeys a chain rule decomposition if the parametric distribution is a joint distribution of variable X and Y:

$$J_{X,Y}(\theta) = J_X(\theta) + J_{Y|X}(\theta) \tag{5.7}$$

$$J_{X,Y}(\theta) = J_X(\theta) + J_Y(\theta) \text{ If } Y \text{ is independent of } X \tag{5.8}$$

4. Data Processing Inequality: applying a function on the observations (process-

ing) can only decrease the fisher information

$$\begin{aligned} &\theta \to x \sim \mathbb{P}(x \mid \theta) \\ &x \to y \sim q(x) \\ \implies &J_Y(\theta) \leq J_X(\theta) \end{aligned} \tag{5.9}$$

Now, we are in the position to state a central theorem relating fisher information to the variance of the estimator.

Theorem (Cramér-Rao Theorem (1944))

The **Cramér Rao** theorem relates the Fisher information to the minimum possible variance of an estimator of θ. For any **unbiased** estimator $\hat{\theta}(X)$ it holds that:

$$\mathfrak{R}\left[\hat{\theta} \mid \theta\right] = \mathsf{Var}[\hat{\theta}] \leq \frac{1}{J(\theta)} \tag{5.10}$$

Before starting the proof, we need an important lemma:

Lemma (Cauchy-Schwarz Inequality)

For two random variables X, Y it holds that

$$\mathbb{E}^2[XY] \leq \mathbb{E}[X^2]\mathbb{E}[Y^2] \tag{5.11}$$

with equality iff $X = aY$ for some constant a. Equivalently

$$\begin{aligned} \mathbb{E}^2[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])] &\leq \mathbb{E}[(X - \mathbb{E}[X])^2]\mathbb{E}[(Y - \mathbb{E}[Y])^2] \\ \mathsf{Cov}^2(X, Y) &\leq \mathsf{Var}(X)V(Y) \end{aligned} \tag{5.12}$$

Remark. The ratio of the LHS to the RHS is called the squared correlation coefficient.

Proof. Throughout we assume we are given some estimator $\hat{\theta}(X)$. Consider the expected value of $\hat{\theta}(X)S(X, \theta)$:

$$\mathbb{E}^2\left[\hat{\theta}(X)S(X, \theta)\right] \tag{5.13}$$

On the one hand:

$$\begin{aligned} \mathbb{E}\left[\hat{\theta}(X)S(X, \theta)\right] &= \int \nabla_\theta \log \mathbb{P}(x \mid \theta)\hat{\theta}(x)\mathbb{P}(x \mid \theta)dx \\ &= \int \nabla_\theta \mathbb{P}(x \mid \theta)\hat{\theta}(x)dx \\ &= \nabla_\theta \int \mathbb{P}(x \mid \theta)\hat{\theta}(x)dx = \nabla_\theta \theta = 1 \end{aligned} \tag{5.14}$$

where we have used the fact that the estimator is unbiased for the last equality.

On the other hand, using the cuachy-schwarz inequality we have:

$$\begin{aligned}\mathbb{E}^2\left[\hat{\theta}(X)S(X,\theta)\right] &= \mathbb{E}\left[\hat{\theta}(X)S(X,\theta) - \mathbb{E}[\hat{\theta}]\mathbb{E}[S(X,\theta)]\right] \\ &= \mathsf{Cov}\left[\hat{\theta}(X), S(X,\theta)\right] \\ &\leq \mathsf{Var}\left[\hat{\theta} \mid \theta\right] \cdot J(\theta)\end{aligned}$$

Which combined with 5.14 yields the CR bound:

$$\mathsf{Var}\left[\hat{\theta} \mid \theta\right] \cdot J(\theta) \geq 1 \implies \mathsf{Var}\left[\hat{\theta} \mid \theta\right] \geq \frac{1}{J(\theta)} \tag{5.15}$$

Remark. If an unbiased estimator saturates the CR bound, then it is called **efficient**.

□

§5.1 Fisher Information and KL Divergence

The fisher information is closely related to the KL divergence. Indeed, the KL divergence is a measure of the distance between two distributions and the Fisher information ca be thought of as the distance when the distributions are close to each other. Let's see this in more detail.

$$\mathfrak{D}_{KL}\left[\mathbb{P}(\theta+\delta\theta) \mid\mid \mathbb{P}(\theta)\right] = \sum_x \mathbb{P}(\theta+\delta\theta)\log\frac{\mathbb{P}(\theta+\delta\theta)}{\mathbb{P}(\theta)} \approx \frac{1}{2}\delta\theta^2 J(\theta) \tag{5.16}$$

Theorem

For an exponential family distribution,

$$\mathbb{P}(x \mid \theta) = \exp A(x)B(\theta) + D(\theta) + C(x) \tag{5.17}$$

If $A(x)$ is an unbiased estimator of θ then $A(x)$ saturates the Cramer Rao bound and if $\hat{\theta}(x^n)$ is efficient for $\mathbb{P}(x^n \mid \theta)$ then $\mathbb{P}(x \mid \theta)$ is from the exponential family.

§5.2 Asymptotic Efficiency of Maximum Likelihood Estimator

We have already seen that the ML estimator is consistent (i.e., also asymptotically unbiased). Is it efficient?

Theorem (MLE is asymptotically both normal and efficient)
The ML estimator is asymptotically efficient as $n \to \infty$

$$\hat{\theta}_{ML} \sim \mathcal{N}\left(\theta, \frac{1}{nJ(\theta)}\right) \tag{5.18}$$

Proof. Write a Taylor expansion of $\ell'(\theta) = S(X, \theta)$ around $\hat{\theta}_{ML}$which by definition obeys $S(X, \hat{\theta}_{ML}) = 0$.

$$0 = S(X, \hat{\theta}_{ML}) \approx S(X, \theta) + (\hat{\theta}_{ML} - \theta)\frac{\partial}{\partial\theta}S(X, \theta) + \frac{1}{2}(\hat{\theta}_{ML} - \theta)^2\frac{\partial^2}{\partial\theta^2}S(X, \theta) \tag{5.19}$$

We first assume the last term is small and ignore it. More precisely, we assume that the variance is $O(1/n)$, but the bias squared is $O(1/n^2)$. Then, the bias contribution is negligible.

$$\begin{aligned} 0 &= S(\hat{\theta}_{ML}, X) \approx S(X, \theta) + (\hat{\theta}_{ML} - \theta)S'(X, \theta) \\ \hat{\theta}_{ML} - \theta &= -\frac{S(X, \theta)}{S'(X, \theta)} \\ \sqrt{n}(\hat{\theta}_{ML} - \theta) &= -\frac{\frac{1}{\sqrt{n}}S(X, \theta)}{\frac{1}{n}S'(X, \theta)} \end{aligned}$$

The top converges to a normal distribution because of CLT, and the bottom is just the Fisher Information.

$$\begin{aligned} \frac{1}{\sqrt{n}}\sum_i S(x_i, \theta) &\to \mathcal{N}(0, J(\theta)) \\ -\frac{1}{n}\sum_i \frac{\partial^2 \log \mathbb{P}(x \mid \theta)}{\partial\theta^2} &\to J(\theta) \end{aligned}$$

From which it follows that $\sqrt{n}(\hat{\theta}_{ML} - \theta)$ is approximately distributed as $\mathcal{N}\left(\theta, \frac{1}{nJ(\theta)}\right)$.

Let us return to the ignored term in 5.19.

$$(\hat{\theta}_{ML} - \theta) = \frac{-S(X, \theta) + \frac{1}{2}(\hat{\theta}_{ML} - \theta)^2 nJ'(\theta)}{nJ(\theta)} \tag{5.20}$$

Averaging this equation, we see that the leftover term goes to zero as $n \to \infty$.

$$b(\theta) = \frac{Var(\hat{\theta}_{ML})J'(\theta)}{2J(\theta)} = \frac{J'(\theta)}{2n} \tag{5.21}$$

□

§5.3 Fisher Information for Dsicrimination

Returing back to the ideas from the last chapter briefly, we can use the Fisher information to measure the discrimination of a classifier between two possible classes θ_1 and θ_2. Assuming w.l.o.g. that θ_1 is the positive class, we can write the log-likelihood ratio as

$$\Delta L = \log \mathbb{P}(\overline{x} \mid \theta_1) - \log \mathbb{P}(\overline{x} \mid \theta_2) \tag{5.22}$$

And note that error occurs when $\Delta L > 0$. Let's now look at the local behavior assuming that θ_1 is close to θ_2. By Taylor expansion, we have

$$\Delta L = \frac{\partial \log \mathbb{P}(\overline{x} \mid \theta_1)}{\partial \theta}(\theta_1 - \theta_2) - \frac{\partial^2 \log \mathbb{P}(\overline{x} \mid \theta_2)}{\partial \theta^2}(\theta_1 - \theta_2)^2 \tag{5.23}$$

$$= \frac{\partial \log \mathbb{P}(\overline{x} \mid \theta_1)}{\partial \theta}(\Delta\theta) - \frac{\partial^2 \log \mathbb{P}(\overline{x} \mid \theta_2)}{\partial \theta^2}(\Delta\theta)^2 \tag{5.24}$$

Now, by the central limit theorem, we can get that ΔL is distributed as

$$\Delta L \sim \mathcal{N}\left(-\frac{1}{2}n\Delta\theta^2 J, \sqrt{nJ}\Delta\theta\right) \tag{5.25}$$

Implying that the error rate i.e., the probability of error $\mathbb{P}(\Delta L > 0)$ is given by

$$\mathbb{P}(\Delta L > 0) = \frac{1}{2}\operatorname{erfc}\left(\frac{\Delta\theta\sqrt{nJ}}{2\sqrt{2}}\right) \tag{5.26}$$

§5.4 Fisher Information for Poisson Neurons

Consider a population of neurons firing at rate λ where λ be a function of some parameter θ that represents the stimulus. The motivation is that the neural firing rate is not deterministic and often noisy, and we attempt to capture this noise by modelling the firing rate as a Poisson process. The probability of observing r spikes in a time interval of length t is given by

$$\mathbb{P}(r \mid \lambda) = \frac{\lambda^r e^{-\lambda}}{r!} \tag{5.27}$$

Recall that the mean and variance of a Poisson random variable are equal to the parameter λ, and we assume that $\lambda = f(\theta)$, implying:

$$\mathbb{P}(r \mid \theta) = \frac{[f(\theta)]^r e^{-f(\theta)}}{r!} \tag{5.28}$$

Claim

The Fisher Information for a Poisson neuron is given by

$$J(\theta) = \frac{[f'(\theta)]^2}{f(\theta)} \tag{5.29}$$

Proof.

$$\begin{aligned}
J(\theta) &= -\mathbb{E}\left[\frac{\partial^2}{\partial\theta^2}\log \mathbb{P}(r \mid \theta)\right] \\
&= -\mathbb{E}\left[\frac{\partial^2}{\partial\theta^2}\log \frac{[f(\theta)]^r\, e^{-f(\theta)}}{r!}\right] \\
&= -\mathbb{E}\left[\frac{\partial^2}{\partial\theta^2}\left\{r\log f(\theta) - f(\theta) - \log r!\right\}\right] \\
&= -\mathbb{E}\left[\frac{\partial}{\partial\theta}\left\{\frac{r}{f(\theta)}f'(\theta) - f'(\theta)\right\}\right] \\
&= -\mathbb{E}\left[\frac{r}{f(\theta)}f''(\theta) - \frac{r}{f(\theta)^2}f'(\theta)^2 - f''(\theta)\right] \\
&= -\frac{f(\theta)}{f(\theta)}f''(\theta) + \frac{f(\theta)}{f(\theta)^2}f'(\theta)^2 + f''(\theta) \\
&= \frac{f'(\theta)^2}{f(\theta)}
\end{aligned}$$

Remark. So, far we can think of the fisher information as spikes per unit time, but more generally we can think of spikes in a given time period Δt, which will make the fisher information a function of Δt. Observing that the parameter λ now becomes $\lambda = f(\theta)\Delta t$, we can write the fisher information as:

$$J(\theta) = \frac{[f'(\theta)\Delta t]^2}{f(\theta)\Delta t} = \frac{[f'(\theta)]^2}{f(\theta)}\Delta t \tag{5.30}$$

□

§5.5 Multivariate version of Carmer Rao

Thus far we looked at a scalar parameter θ. What happens if we have multiple parameters $\theta = (\theta_1, \ldots, \theta_k)$. Say we have an unbiased estimator of these $\theta(x^n)$. What is the minimum variance? In this case we can actually talk about the co-variance. We can bound it via the Fisher information matrix.

$$J_{ij}(\theta) = E_{p(x|\theta)}\left[\frac{\partial \log \mathbb{P}(x \mid \theta)}{\partial\theta_i} \cdot \frac{\partial \log \mathbb{P}(x \mid \theta)}{\partial\theta_j}\right] \tag{5.31}$$

This is a matrix. For any unbiased estimator $\hat{\theta}(x)$ of θ it can be shown that:

$$Cov(\hat{\theta} \mid \theta) \succeq J^{-1}(\theta) \tag{5.32}$$

Note: for two matrices A,and B the inequality

$$A \succeq B \tag{5.33}$$

means that the matrix

$$A - B \text{ is semidefinite} \tag{5.34}$$

which in our case has the following meaning: Suppose we are interested in MSE loss for a scalar variable ϕ given as

$$\phi = u^T \theta \tag{5.35}$$

where u is a unit vector in θ space. Then, if the estimator is unbiased

$$V(\phi) \geq u^T J^{-1} u \tag{5.36}$$

Note that if u_λ is an eigenvector of J with eigenvalue $J_\lambda$then,

$$V(\phi) \geq \frac{1}{J_\lambda(\theta)} \tag{5.37}$$

In particular, we can look at the Cramer Rao bound for variances:

$$V(\hat{\theta}_i|\theta) \geq \left(J^{-1}\right)_{ii}. \tag{5.38}$$

A simpler bound is often used for the variances,

$$V(\hat{\theta}_i|\theta) \geq J_{ii}^{-1}. \tag{5.39}$$

Proof:

$$J_{ii}(J^{-1})_{ii} = \sum_{\lambda\lambda'} J_\lambda J_{\lambda'}^{-1} u_{i\lambda}^2 u_{i\lambda'}^2 = 1 + \sum_{\lambda \neq \lambda'} J_\lambda J_{\lambda'}^{-1} u_{i\lambda}^2 u_{i\lambda'}^2 \tag{5.40}$$

thus,

$$J_{ii}\left(J^{-1}\right)_{ii} \geq 1 \tag{5.41}$$

$$V(\hat{\theta}_i|\theta) \geq (J^{-1})_{ii} \geq J_{ii}^{-1} \tag{5.42}$$

This bound however can be rather loose.

§5.6 Narrow Vs. Wide Receptive Fields

Let's go back and consider a population of N Poisson Neurons that tile the input space say orientation uniformly $\theta \in [0, 2\pi]$. As before we assume that the mean of the Poisson Neuron is a function of the stimulus θ , that is $\lambda(\theta) = f(\theta)$. Every neuron i has a preferred orientation θ_i. The tuning curve is symmetric and the code is translation invariant, so that we can write the tuning curve for each neuron as $f(\theta) = f(\theta - \theta_i)$, and the way we discretize the space is given by the relation $\Delta\theta = \frac{2\pi}{N}$. Now, we are ready to write the fisher information, which sums over all neurons:

$$J(\theta) = \sum_{i=1}^{N} \frac{[f'(\theta - \theta_i)]^2}{f(\theta - \theta_i)} \tag{5.43}$$

Since we have uniform tiling we can try to approximate the fisher information sum as an integral:

$$J(\theta) \approx \frac{1}{\Delta\theta} \int_0^{2\pi} \frac{[f'(\theta - \theta_i)]^2}{f(\theta - \theta_i)} d\theta_i \approx \frac{N}{2\pi} \int_0^{2\pi} \frac{[f'(\theta - \theta_i)]^2}{f(\theta - \theta_i)} d\theta_i \tag{5.44}$$

The integral is a definitie integral which will simply come down to some number that we can call J_0.

$$J(\theta) \approx \frac{N}{2\pi} J_0 \tag{5.45}$$

Now, we are ready to ask: what would be the effect of a parameter s that could scale to make the receptive field narrow or wider, and to that end we define the following:

$$f(\theta) \stackrel{\text{def}}{=} g\left(\frac{\theta}{s}\right) \tag{5.46}$$

where s large implies broader tuning and small implies narrower tuning. How does it affect the fisher information then?

$$J(\theta) = \frac{N}{2\pi} \int_0^{2\pi} \frac{1}{s^2} \frac{g'(\bar{\theta})^2}{g(\bar{\theta})} s d\bar{\theta} = \frac{N}{2\pi} \frac{J_0}{s} \tag{5.47}$$

Ideally, we would then like to take the limit $s \to 0$ but we are limited by the fact that $\Delta\theta$ is lower bounded. But the broad lesson that we can indeed learn is that is preferable to have a narrower tuning curve.

What about dimension larger than 1?

We can start by noting that in the above calculation the dimensionality only explicity comes into play when we use the Jacobian for substitution of variable and instead of $sd\bar{\theta}$ we would get $s^d\bar{\theta}$, implying:

$$J(\theta) = \frac{N}{2\pi} J_0 s^{d-2} \tag{5.48}$$

This leads us to two interesting finding that in dimension 2 the width of the tuning curve doesn't really matter and for dimension greater than 3 we would instead prefer to have a wider tuning curve!

§5.7 Sufficient Statistics

We want to estimate θ from an observation x (can also be an IID sample). The estimate $\hat{\theta}(x)$ can be viewed as a compression of x that provides information about θ. We would like to develop a theory of what functions of x provide information about the parameter. For example, we saw that the estimate of the covariance is a function of the empirical first and second moments.

Definition 5.1

A function $T(x)$ is called a statistic of x and a statistic $T(x)$ is called sufficient for θ if estimators of the form $\hat{\theta}(T(x))$ are not worse than estimators $\hat{\theta}(x)$. Formally, $T(x)$ is sufficient if $p(x|T(x), \theta)$ is not dependent on θ. This is often written as:

$$f(x|T(x), \theta) = f(x|T(x)) \tag{5.49}$$

Claim

One-parameter members of the exponential family have density or mass function of the form

$$f(x \mid \theta) = \exp[c(\theta)T(x) + d(\theta) + S(x)]$$

Suppose that $X_1, \cdots, X_n$ are i.i.d. samples from a member of the exponential family, then the joint probability function is

$$\begin{aligned} f(\mathbf{x} \mid \theta) &= \prod_{i=1}^{n} \exp\left[c(\theta)T\left(x_i\right) + d(\theta) + S\left(x_i\right)\right] \\ &= \exp\left[c(\theta)\sum_{i=1}^{n} T\left(x_i\right) + nd(\theta)\right] \exp\left[\sum_{i=1}^{n} S\left(x_i\right)\right] \end{aligned}$$

and the function $\sum_{i=1}^{n} T(x_i)$ is a sufficient statistics.

In particular this is true for the Poisson distribution implying that a sufficient statistics for Poisson neurons is simply the number of spikes in a give time period, since $T(x_i) = x_i$. One way to read this is that the precise timing of the neural spike doesn't really convey much information!

§6 Linear Decoding

In this section, we ask a simplified question: what can we know from linear combination of spike observations? We can try to formalize this as follows:

$$\text{Linear Discrimination:} \quad \Theta\left(\sum_{i=1}^{N} w_i n_i + b\right) \tag{6.1}$$

$$\text{Linear Estimation:} \quad \hat{\Theta} = \sum_{i=1}^{N} w_i n_i + b \tag{6.2}$$

where w_i are the weights and b is the bias. We can then ask: How well does a linear decoding do, and what is the best choice of w_i and b?

Claim

Poisson Neurons have linear discriminators by non-linear ML estimators.

Proof. The log-likelihood ratio for Poisson neurons is:

$$\begin{aligned}\Delta \log L &= \log\left(\mathbb{P}\left(\mathbf{n} \mid \theta_2\right)\right) - \log\left(\mathbb{P}\left(\mathbf{n} \mid \theta_2\right)\right) \\ &= \sum_{j=1}^{N} n_i \log\left[\frac{f_i(\theta_2)}{f_i(\theta_1)}\right] - \sum_{j=1}^{N}\left[f_i(\theta_2) - f_i(\theta_1)\right]\end{aligned}$$

But if we were to calculate the ML estimator:

$$\hat{\theta} = \arg\max_{\theta} \log L(\theta)$$

$$\implies \frac{\partial \log L}{\partial \theta} = \sum_{j=1}^{N} n_i \frac{f_i'(\theta)}{f_i(\theta)} - \sum_{j=1}^{N} f_i'(\theta) = 0$$

We need to select θ such that the above equality is satisfied, and we can observe that it is a non-linear function of neural response. □

§6.1 General Theory for Linear Decoding

We can formalize the task of linear decoding by looking at the MSE as we have done before. So, given some neural observations $\mathbf{r} = [r_1, r_2, \cdots, r_N]$ sampled from $\mathbb{P}\left(\mathbf{r} \mid \theta\right)$ and some weights $\mathbf{w} = [w_1, w_2, \cdots, w_N]$ we can define the estimated parameter as:

$$\hat{\theta} = \mathbf{w} \cdot \mathbf{r} + b \tag{6.3}$$

and the MSE:

$$\text{MSE} = \mathbb{E}\left[\left(\hat{\theta} - \theta\right)^2\right]_{\mathbb{P}(\mathbf{r},\theta)} = \int d\theta \int d\mathbf{r}\, \mathbb{P}\left(\mathbf{r} \mid \theta\right)\left(\hat{\theta} - \theta\right)^2 \tag{6.4}$$

$$= \mathbb{E}\left[\left(\mathbf{w} \cdot \mathbf{r} + b - \theta\right)^2\right] \tag{6.5}$$

where the expectation is taken over the distribution of the neural response. We can then ask: What is the optimal choice of $\mathbf{w}$ and b that minimizes the MSE?

First, let's calculate the optimal b for a given $\mathbf{w}$. We can do this by setting the derivative of the MSE with respect to b to zero:

$$\frac{\partial \text{MSE}}{\partial b} = \mathbb{E}\left[(\mathbf{w}\cdot\mathbf{r} + b - \theta)\right] = 0 \tag{6.6}$$

$$\implies \hat{b} = \mathbb{E}\left[\theta - \mathbf{w}\cdot\mathbf{r}\right] = \mathbb{E}\left[\theta\right] - \mathbf{w}\cdot\mathbb{E}\left[\mathbf{r}\right] \tag{6.7}$$

Now, let's calculate the optimal $\mathbf{w}$ given $\hat{b}$. We can do this by setting the derivative of the MSE with respect to $\mathbf{w}$ to zero:

$$\begin{aligned}
\frac{\partial \text{MSE}}{\partial \mathbf{w}} &= \mathbb{E}\left[\left(\mathbf{w}\cdot\mathbf{r} + \hat{b} - \theta\right)^\top \cdot \mathbf{r}\right] = 0 \\
&\implies \mathbb{E}\left[\mathbf{r}\cdot\mathbf{r}^\top\right]\cdot\mathbf{w} + \hat{b}\cdot\mathbb{E}\left[\mathbf{r}\right] - \mathbb{E}\left[\theta\cdot\mathbf{r}\right] = 0 \\
&\implies \mathbb{E}\left[\mathbf{r}\cdot\mathbf{r}^\top\right] + \mathbb{E}\left[\theta\right]\,\mathbb{E}\left[\mathbf{r}\right] - \mathbb{E}\left[\mathbf{r}\right]\cdot\mathbb{E}\left[\mathbf{r}^\top\right]\cdot\mathbf{w} - \mathbb{E}\left[\theta\cdot\mathbf{r}\right] = 0 \\
&\implies \left(\mathbb{E}\left[\mathbf{r}\cdot\mathbf{r}^\top\right] - \mathbb{E}\left[r\right]\,\mathbb{E}\left[\mathbf{r}^\top\right]\right)\cdot\mathbf{w} = \mathbb{E}\left[\theta\cdot\mathbf{r}\right] - \mathbb{E}\left[\theta\right]\,\mathbb{E}\left[\mathbf{r}\right] \\
&\implies \mathbf{C}\cdot\mathbf{w} = \mathbf{u} \\
&\implies \mathbf{w} = \mathbf{C}^{-1}\mathbf{u}
\end{aligned}$$

Proposition

All in all, we get the following expression for optimal weight in terms of variable $\mathbf{C}$ and $\mathbf{u}$.

$$\hat{\mathbf{w}} = \mathbf{C}^{-1}\mathbf{u} \tag{6.8}$$

$$\mathbf{C} = \mathbb{E}\left[\mathbf{r}\cdot\mathbf{r}^\top\right] - \mathbb{E}\left[r\right]\,\mathbb{E}\left[\mathbf{r}^\top\right] \tag{6.9}$$

$$\mathbf{u} = \mathbb{E}\left[\theta\cdot\mathbf{r}\right] - \mathbb{E}\left[\theta\right]\,\mathbb{E}\left[\mathbf{r}\right] \tag{6.10}$$

In particular, if the neurons are independent, we get a simpler expression for $\hat{\mathbf{w}}$:

$$\hat{\mathbf{w}}_i = \frac{\mathbb{E}\left[\theta\cdot r_i\right] - \mathbb{E}\left[\theta\right]\,\mathbb{E}\left[r_i\right]}{\mathbb{E}\left[r_i^2\right] - \mathbb{E}\left[r_i\right]^2} = \frac{\mathbb{E}\left[\theta\cdot r_i\right] - \mathbb{E}\left[\theta\right]\,\mathbb{E}\left[r_i\right]}{\mathsf{Var}\left[r_i\right]^2} \tag{6.11}$$

Note that we have found the optimal weights and bias for a given distribution of the neural response, our estimator is unbiased over the joint distribution of the neural response and the parameter. However, conditioned on the stimulus, the estimator is not unbiased.

§6.2 Correlated Neurons

As usual consider a neural population whose firing rate given a stimulus θ is given by: $\langle r_i \rangle = f_{\mathcal{I}}(\theta)$. In this case consider a presence of gaussian noise (independent to the stimulus) given some covariance matrix C.

$$\mathbb{P}(\bar{r} \mid \theta) \propto \exp\left[-\frac{1}{2}\sum_{i,j}\left(r_i - f_{\mathcal{I}}(\theta)\right) C_{ij}^{-1}\left(r_j - f_j(\theta)\right)\right] \tag{6.12}$$

Consider a discrimination problem where we want to distinguish between 2 values of the stimulus θ_1 and θ_2. We can define the response of the neuron as:

$$\begin{aligned} f_i^- &\equiv f_i\left(\theta_1\right) = \langle r_i \mid \theta_1 \rangle \\ f_i^+ &\equiv f_i\left(\theta_2\right) = \langle r_i \mid \theta_2 \rangle \\ g_i &= f_i^+ - f_i^- \end{aligned}$$

The discriminator is a linear combination of the responses:

$$L_w(\bar{r}) = \sum_{i=1}^{N} w_i r_i \tag{6.13}$$

The discriminator, itself is also a random variable, so we get:

$$\mu_1 = \sum_i w_i f_i^- \quad ; \quad \mu_2 = \sum_i w_i f_i^+$$

$$\sigma^2 = \left\langle \left[\sum_i w_i r_i - \sum_i w_i f_{\mathcal{I}}(\theta)\right]^2 \right\rangle = \sum_{i,j} w_i \left\langle \left(r_i - f_i\right)\left(r_j - f_j\right)\right\rangle w_j = \sum_{i,j} w_i C_{ij} w_j$$

Note that: ($\sigma_1 = \sigma = \sigma_2$) because the noise is independent of θ. And now, we can calculate error as the area under the curve beyond the decision threshold supposing that the true stimulus is θ_1:

$$\mathbb{P}_{\text{error}} = \mathbb{P}\left(L_w(\bar{r}) \geq \frac{\mu_1 + \mu_2}{2}\right) = \mathbb{P}\left(\mathcal{Z} \geq \frac{1}{\sigma}\left(\frac{\mu_1 + \mu_2}{2} - \mu_1\right)\right) = \mathbb{P}\left(\mathcal{Z} \geq \frac{\mu_2 - \mu_1}{2\sigma}\right)$$

Now, we define a key quantity called the discriminability:

$$\mathrm{D}'^2 = \left(\frac{\mu_2 - \mu_1}{\sigma}\right)^2 = \frac{\left(\sum_i w_i g_i\right)^2}{\sum_{ij} w_i C_{ij} w_j} = \frac{\left(g^{\mathsf{T}} w\right)^2}{w^{\mathsf{T}} C w} \tag{6.14}$$

And as you will note that this quantity appears as a threshold for the error calculation, and for gaussians, the error is monotonically decreasing with respect to D'^2. So, we can find the optimal weights by maximizing the discriminability (and thereby the parameters of the linear discriminator) and you'll find that:

$$w^* \propto C^{-1} g \tag{6.15}$$

Example 6.1 (N Identical Uncorrelated Neurons with Gaussian Noise)

$$g_i = f_i^+ - f_i^- = g \quad ; \quad C_{ij} = a \cdot \delta_{ij} \implies w_i^* = w$$

$$\mathrm{D}'^2 = \frac{g^2}{a} \cdot (1, 1, \cdots, 1) \cdot \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} = N\frac{g^2}{a}$$

As we expect the larger the distance between the responses implies better discriminability, but larger noise variance implies worse discriminability.

Example 6.2 (N Identical semi-Correlated Neurons with Gaussian Noise)

$$g_i = g \quad ; \quad C_{ij} = a\left[\delta_{ij} + \alpha\left(1 - \delta_{ij}\right)\right] = a \begin{pmatrix} 1 & & \alpha \\ & \ddots & \\ \alpha & & 1 \end{pmatrix}$$

Implies,

$$C^{-1} = b \cdot \begin{pmatrix} 1 & & \beta \\ & \ddots & \\ \beta & & 1 \end{pmatrix} \quad ; \quad b = \frac{1}{a} \cdot \frac{N\alpha + 1 - 2\alpha}{(N\alpha + 1 - \alpha)} \quad ; \quad \beta = -\frac{\alpha}{N\alpha + 1 - 2\alpha}$$

Finally,

$$\mathrm{D}'^2_{\mathrm{opt}} = g^2 b \begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix} \begin{pmatrix} 1 & & \beta \\ & \ddots & \\ \beta & & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ \cdots \\ 1 \end{pmatrix} =$$

$$g^2[Nb + N(N-1) \cdot b\beta] = g^2 Nb(1 + (N-1)\beta) = \frac{g^2}{a} \cdot \frac{N}{(N-1)\alpha + 1}$$

Note that at $\lim_{N\to\infty} \mathrm{D}'^2_{\mathrm{opt}} \to \frac{g^2}{a\alpha}$ is a function of N and when $0 < \alpha < 1$ and N is small we get better discriminability than with single neuron, but at some point it saturates, and we stop benefiting from adding more neurons.

Example 6.3 (Two Neuronal Populations With Opposite Responses)

$$g_i = f_i^+ - f_i^- = \begin{cases} g & i = 1 \dots \frac{N}{2} \\ -g & i = \frac{N}{2} + 1 \dots N \end{cases}$$

$$C_{ij} = a\left[\delta_{ij} + \alpha\left(1 - \delta_{ij}\right)\right] = a \begin{pmatrix} 1 & & \alpha \\ & \ddots & \\ \alpha & & 1 \end{pmatrix}$$

As before,

$$C^{-1} = b \cdot \begin{pmatrix} 1 & & \beta \\ & \ddots & \\ \beta & & 1 \end{pmatrix} \quad ; \quad b = \frac{1}{a} \cdot \frac{N\alpha + 1 - 2\alpha}{(N\alpha + 1 - \alpha)} \quad ; \quad \beta = -\frac{\alpha}{N\alpha + 1 - 2\alpha}$$

Now, it's clear that the optimal weights have a structure that subtracts one population from the other, i.e. $w^* = \left(1, 1, \cdots, -1, -1\right)$. This implies that the optimal discriminability is:

$$\mathrm{D}'^2_{\mathrm{opt}} = 2g^2 b\left[\frac{N}{2} + \frac{N}{2}\left(\frac{N}{2} - 1\right)\beta\right] - 2\left(\frac{N}{2}\right)^2 \beta b = \frac{g^2 N}{(1-\alpha)a}$$

When the neuronal responses were identical the discriminability saturates when $N \to \infty$, but now grows linearly with N and doesn't saturate! Further, when α increases the discriminability improves, and when $\alpha \to 1$ it diverges becuase when the neurons are fully correlated, taking the difference eliminates the noise, and discriminability diverges.

Remark. Note that for general g_i and $C_{ij} = a\left[\delta_{ij} + \alpha\left(1 - \delta_{ij}\right)\right]$

$$\mathrm{D}'^2_{\mathrm{opt}} = N\frac{\langle g \rangle^2}{a}\left[1 + \alpha\left(-1 + N\frac{(1-\alpha)(1-\kappa)}{1-\alpha(1-N\kappa)}\right)\right]^{-1} \quad ; \quad \kappa = \frac{\langle g^2 \rangle - \langle g \rangle^2}{\langle g^2 \rangle} \tag{6.16}$$

You can verify that substituting the variables according to the setup from the above examples, we get the results from the examples.

The major takeaway is that both the structure of noise and the structure of the responses are important for the discrimination, and geometrically, it matters if the noise is along or orthogonal to the responses leading to very different discriminability.

§7 Markov Decoding

We consider a random process in discrete time that have states s_t that have a joint distribution $\mathbb{P}(s_1, \dots s_t, \dots)$ and a transition matrix $\mathbb{P}(s_{t+1} \mid s_t)$. In particular if we have a Markov process the joint distribution factorizes:

$$\mathbb{P}(s_1, \dots s_t, \dots) = \mathbb{P}(s_1) \prod_{t=1}^{\infty} \mathbb{P}(s_{t+1} \mid s_t) \tag{7.1}$$

where $\mathbb{P}(s_1)$ is the initial distribution (prior) and $\mathbb{P}(s_{t+1} \mid s_t)$ is the transition probability.

In our context the states s_t are the hidden states of the neural system, and we collect observations o_t are that have a conditioned distribution $\mathbb{P}(o_t \mid s_t)$. Echoing, what we have been doing so far, we want to find the posterior distribution of the hidden states given the observations. There are few different types of inference problems associated with Markov decoding:

1. **Filtering**: Given the observations up to time t, what is the posterior distribution of the hidden states at time t?
2. **Smoothing**: Given the observations up to time $T > t$, what is the posterior distribution of the hidden states at time t?
3. **ML sequence inference**: Given the observations up to time T, what is the most likely sequence of hidden states?
4. **Likelihood of Observations**: Given the observations up to time T, what is the likelihood of the observations?

While this calculation seems daunting the Markov property massively simplifies the calculation. In particular,

$$\mathbb{P}(s_{1\dots t} \mid o_{1\dots t}) = \frac{\mathbb{P}(o_{1\dots t} \mid s_{1\dots t})\,\mathbb{P}(s_{1\dots t})}{\mathbb{P}(o_{1\dots t})} \tag{7.2}$$

$$= \frac{1}{Z_t} \mathbb{P}(s_1) \prod_{i=1}^{t-1} \mathbb{P}(s_{i+1} \mid s_i) \prod_{i=1}^{t} \mathbb{P}(o_i \mid s_i) \tag{7.3}$$

where Z_t is the normalization constant. In particular, we can write the posterior distribution as a product of the prior and the likelihood of the observations. This is the key to the Markov property, and it is the reason why we can do inference in a tractable way. Let's see how this makes things easier in the case of filtering. We can marginalize the distribution over previous states to get the posterior distribution of the current state:

$$\mathbb{P}(s_t \mid o_{1\dots t}) = \sum_{s_{1\dots t-1}} \mathbb{P}(s_{1\dots t} \mid o_{1\dots t}) \tag{7.4}$$

$$= \frac{Z_{t-1}}{Z_t} \sum_{s_{1\dots s_{t-1}}} \mathbb{P}(s_{1\dots t-1} \mid o_{1\dots t-1}) \cdot \mathbb{P}(s_t \mid s_{t-1}) \cdot \mathbb{P}(o_t \mid s_t) \tag{7.5}$$

$$= \frac{1}{\tilde{Z}_t} \sum_{s_{1\dots s_{t-1}}} \mathbb{P}(s_{1\dots t-1} \mid o_{1\dots t-1}) \cdot \mathbb{P}(s_t \mid s_{t-1}) \cdot \mathbb{P}(o_t \mid s_t) \tag{7.6}$$

which provides an iterative algorithm for calculating the posterior distribution of the hidden states. In particular, when the change in the state is gaussian and the observations conditioned on the states are gaussian, we recover the Kalman filter. We get

a similarly simple algorithm for smoothing, and for the most likely sequence inference called the "forwards backward algorithm" and "Viterbi algorithm" respectively. Finally, the likelihood of the observations is given by (simply from the Bayes rule):

$$\mathbb{P}\left(o_{1\ldots t}\right) = Z_t = \prod_{i=1}^{t} \tilde{Z}_i \tag{7.7}$$

§7.1 Moving Animal in 1D

Consider N neurons that are recording the activity of a moving animal in 1D in the form of a dense localized place cell repceptive field. The animal is in some location $x \in X$. The neurons are as usual poission neuron and each neuron has a preferred location $x_i \in X$ such that the mean firing rate is given by: $\lambda_i = f(x - x_i)$. We derived earlier that the fisher information in a small interval Δt is given by:

$$J = \frac{N\Delta t}{L} \int_{-\infty}^{\infty} \frac{f'(x)^2}{f(x)} = \mathcal{J}\Delta t \tag{7.8}$$

where $\mathcal{J}$ is the fisher information per unit time. Now, what if the animal sis constantly moving? Let's consider a simpler case of a random walk. Consider:

$$x_{t+\Delta t} = x_t + \Delta x \quad ; \quad \Delta x \sim \mathcal{N}(0, 2\mathcal{D}\Delta t)) \tag{7.9}$$

where Δx is a gaussian random variable with mean 0 and variance $2\mathcal{D}\Delta t$. Doing a brief dimensionality analysis we see that the units of $\mathcal{D}$ is $\frac{\texttt{Length}^2}{\texttt{time}}$.

If we want to calculate the Mean Squared Error (MSE) of the estimate of the location of the animal using the markovian decoder, we can use the fact that units of $\mathcal{J}$ is $\frac{1}{\texttt{Length}^2 \cdot \texttt{time}}$. So, we can guess that since the units of MSE is $\texttt{Length}^2$, the MSE is given by:

$$\text{MSE} \propto \sqrt{\frac{\mathcal{D}}{\mathcal{J}}} \quad ; \text{Turns out, precisely this is the case, and : } \text{MSE} = \sqrt{\frac{2\mathcal{D}}{\mathcal{J}}}$$

Simlarly, we can also generate a timescale as $\sqrt{\frac{1}{\mathcal{D}\mathcal{J}}}$. This is the timescale for optimal discretization for decoder!

§8 Coding

The most well studied neural network in the brain is actually physically outside the brain: the retina. The retina is both convenient and interesting to study since it is part of the central nervous system but is outside the brain. Experimentalists can remove the retina from the animal and display images to it while measuring electrophysiologically. Additionally, the retina does not receive inputs from higher order areas in the brain, so we can disconnect it without damaging its processing. This has been true since the beginning of the 1960, when response of photoreceptors and ganglion cells to light was measured by electrophysiologists. If one looks at this response, it becomes very clear that it has some non-trivial structure such as center surround receptive fields and non-linearity.

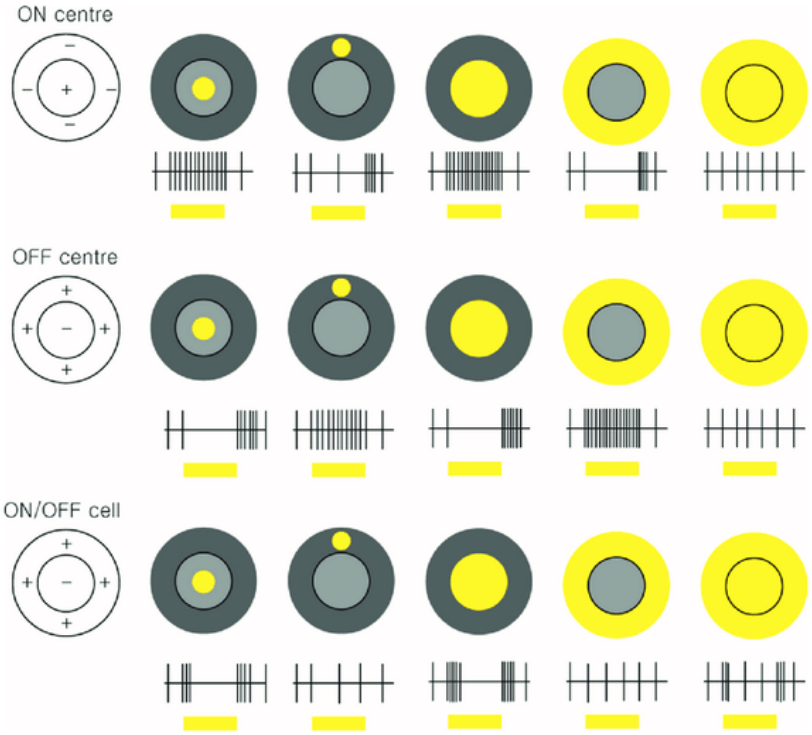

Figure 2: Retinal Ganglion Cells Receptive Field

In 1961 Horace Barlow a very influential paper by Horace Barlow, a British neuroscientist, started with the assertion that to understand the retina (as an example for the neural systems), we must have a hypothesis on the function of this neural network implying that there is some non-trivial yet well-defined function involved in transmitting the visual information to the brain. He made some hypotheses that by todays' viewpoint seem rather over-simplistic, but then simplified his assumptions to an idea which remained very influential until today:

(Barlow's Efficient Coding Hypothesis)

Sensory relays code sensory messages so that their redundancy is reduced but comparatively little information is lost.

There is a lot of redundancy in natural images - for example, nearby pixels usually have similar values. Sending the colors of two adjacent pixels is usually sending the

same information twice. Barlow then talked about the capacity of the optical nerve that connects the retina to the brain, i.e., it's ability to transmit a limited amount of information per unit time, and the notion that to fit the visual signal into the optical nerve it is necessary to remove redundancy that exists in it. We can think of Center-surround fields as a mean to remove redundancy. The computation they preform is similar to a derivative, activity is high when there is a change between the center and the periphery. We only encode areas where there are differences between nearby pixels.

Barlow suggested the efficient coding hypothesis: The retina is trying to maximize the information under constraints (for example maximal spike rates). This is not a trivial hypothesis. For example, an alternative hypothesis is that the retina maximizes certain types of information while discarding others. Additionally, maximal compression is not necessarily the best thing to do to implement behaviorally relevant function. It might be better to represent information in a way that facilitates downstream processing and encoding and decoding might introduce problematic constraints themselves. Nevertheless, this was a very powerful hypothesis, and we will see in this coming up classes how far it can take us. Barlow already used terms like information, capacity and redundancy.

We can formulate these terms, using the work of engineers like Claude Shannon. Shannon developed his theory of information trying to quantify how much information can pass through a communication channel. Shannon tried to answer two main questions:

1. How efficiently can one encode a stream of information (in bits), given the statistics of the information. For example, if we wanted to encode an English text, how many bits per letter do we need?
2. How many bits of information can be transmitted through a noisy channel in a given time? Electric cables always have noise: they can only pass certain frequencies and there is also general noise.

Shannon showed that these two question can be answered separately. We will mainly focus on the first question. In neuroscience, we think in terms of spikes instead of bits. We could in principle ask how many spikes can pass through an axon. There are of course constraints. There are electrical constraints but also metabolic constraints. However, we will not directly discuss this.

§8.1 Information Theory

§8.1.1 Entropy

Suppose we have N symbols in our language and we want to encode a stream of them with length n. The i th symbol appears with probability P_i $\left(\sum_{i=1}^{N} P_i = 1\right)$. How much information do we have in this stream? We can formalize this question by asking what the minimal number of bits we need to encode it is. We will assume that the symbols are independent. This assumption is not true for letters in a text. It might be a better approximation if we look at longer sequences of 10 or 100 letters. Shannon proved that under these conditions to encode n symbols one needs on average at least a certain number of bits larger than n. We are interested in the average since the stream itself is random, and different streams might require different number of bits to encode.

$$\langle n_{bits} \rangle \geq nH$$

$$\mathcal{H}(X) = \sum_{i=1}^{N} P_i \log_2 \left(\frac{1}{P_i}\right) = -\sum_{i=1}^{N} P_i \log_2 \left(P_i\right)$$

More generally, we can define the entropy of a random variable X as the average number of bits needed to encode it.

Definition 8.1 (Entropy)

Let X be a random variable. The entropy of X is defined as:

$$\mathcal{H}(X) = -\sum_{x \in X} p(x) \log p(x) \tag{8.1}$$

Alternatively, one can think of entropy of a random variable is a measure of the uncertainty of the random variable. The entropy is always non-negative and the entropy of a deterministic random variable is 0. The entropy of a uniform random variable is $\mathcal{H}(X) = \log n$ where n is the number of possible outcomes. The entropy of a random variable is maximized when all outcomes are equally likely, i.e. when the random variable is uniform. Let's look at some examples.

Example 8.2

Suppose that the j th symbol appears with $P_j = 1$ and all others appear with probability 0 . In this case, we don't need any bits to reconstruct the stream, since there is only one option.

$$\mathcal{H}(X) = -\sum_{i \neq j}^{N} 0 \log_2(0) - 1 \log_2(1) = 0$$

It is convenient to define $x \log(x) = 0$ since this will give us a function that is right-continuous. Without loss of generality we can prove it for a logarithm in the natural base using L'Hopital's rule (all other logarithms are multiplication of the natural log with some constant):

$$\lim_{x \to 0^+} x \log x = \lim_{x \to 0^+} \frac{\log x}{\frac{1}{x}} = \lim_{x \to 0^+} \frac{\frac{1}{x}}{-\frac{1}{x^2}} = \lim_{x \to 0^+} -x = 0$$

Example 8.3

In the case where the distribution of the symbols is uniform $P_i = \frac{1}{N} \forall_i$:

$$\mathcal{H}(X) = -\sum_{i=1}^{N} \frac{1}{N} \log_2 \left(\frac{1}{N} \right) = \sum_{i=1}^{N} \frac{1}{N} \log_2(N) = \log_2 N$$

Example 8.4

For $1 \leq i \leq M$ (where $M < N$) $P_i = \frac{1}{M}$ and for all others $P_i = 0$:

$$\mathcal{H}(X) = -\sum_{i=1}^{M} \frac{1}{M} \log_2 \left(\frac{1}{M} \right) - \sum_{i=M+1}^{N} 0 \log_2(0) = \log_2 M$$

This case equivalent to the case where there are only M symbol, since we do not

need to take the $N-M$ other symbol into consideration. We do not need to allocate representations to them if they never appear.

Example 8.5

Lets examine the case where $N = 5$ and

$$P_i = \begin{cases} \frac{1}{2} & i = 1 \\ \frac{1}{8} & 1 < i \leq 5 \end{cases}$$

$$\mathcal{H}(X) = \sum_{i=1}^{N} P_i \log_2 \left(\frac{1}{P_i}\right) = \frac{1}{2}\log_2(2) + 4 \cdot \frac{1}{8}\log_2(8) = \frac{1}{2} + 4 \cdot \frac{1}{8} \cdot 3 = 2$$

§8.1.2 Properties of Entropy

Shanon entropy satisfies the following properties:

1. **Non-negativity**: $\mathcal{H}(X) \geq 0$
2. **Bounded**: $\mathcal{H}(X) \leq \log n$
3. **Triviality**: $\mathcal{H}(X) = 0$ iff X is deterministic
4. **Additivity**: If X and Y are independent, then $\mathcal{H}(X, Y) = \mathcal{H}(X) + \mathcal{H}(Y)$
5. **Data Processing**: If X is a function of $Y = f(X)$, then $\mathcal{H}(X) \leq \mathcal{H}(Y)$
6. **Uniqueness**: When Shannon derived the entropy function he started with requirement of what entropy should behave like, its features. He wanted it to be increasing with N, the number of state and to be additive. This requirement uniquely defines the entropy function up to a factor (the base of the logarithm).

§8.2 Efficient Coding of a Single Variable

We will deal with the case of some non-linear mapping $x \mapsto r$ where x is the input (stimulus) and r is the response, and both are continuous variables. Formally, we assume $X \sim p_X$ and $R = f(X)$, where $X \in [-\infty, \infty]$ and we constrain R such that $R \in [0, r_{\max}]$. We ask what is the "optimal" f in this case, and our working hypothesis is that the "neuron" is trying to maximize the (differential) entropy of R. We have all the required tools to pose this as an optimization problem:

$$\begin{array}{ll} \max_f & \mathcal{H}(R) \\ \text{s.t.} & f(x) \in [0, \infty] \end{array}$$

To compute $\mathcal{H}(R)$ we first need to find p_R. To do this, we will use the change of variables technique. Assuming that f is (strictly) monotonic (so that f^{-1} exist) and $R = f(X)$ we know that

$$p_R(r) = \frac{1}{f'(x)} p_X(x)$$

Intuitively, this result is can be explained as follows. Given some value x, we know that the probability for $X \in [x, x + \Delta x]$ is $p_X(x) \cdot \Delta x$. Similarly, if $r = f(x)$ then the

probability for $R \in [r + \Delta r]$ is $p_R(r)\Delta r$. But because r is a deterministic function of x, this probability should be equal, because it describes the same event (this is in the limit of small Δr and Δx). Therefore, when $\Delta x, \Delta r$ become dx, dr respectively we can write:

$$p_X(x)dx = p_R(r)dr$$
$$\frac{dr}{dx}p_R(r) = p_X(x)$$

But because $r = f(x)$, the term $\frac{dr}{dx}$ is simply $f'(x)$ therefore

$$p_R(r) = \frac{1}{f'(x)}p_X(x)$$

Remark. To prove this formally, one could write the CDF of the new variable $F_R(r)$ (in terms of F_X), and then differentiate it to get $p_R(r)$.

Now that we know what p_R is, we can ask what is the optimal f so that $H(R)$ is maximized. This can be done by using calculus of variation (after adding the appropriate constrains), but we can actually find the solution in an easier way [1]. Recall that R is bounded on $[0, r_{\max}]$. We already saw in the previous lesson that the distribution maximizing the (differential) entropy of a variable with a bounded support is the uniform distribution. Therefore, we can immediately deduce that the optimal f satisfies $p_R(r) = C$ for some constant independent of r. We also know that $C = \frac{1}{r_{\max}}$ so that the PDF of R will integrate to 1 . Substituting in (1) we get:

$$\frac{1}{f'(x)}p_X(x) = \frac{1}{r_{\max}}$$
$$f'(x) = r_{\max}p_X(x)$$

And now we can find f by integrating with respect to x :

$$f(x) = \int_{-\infty}^{x} r_{\max}p_X(\tilde{x})d\tilde{x}$$
$$= r_{\max}F_X(x)$$

Why does this result make sense? we got that $f'(x) \propto p_X(x)$. This means that for x values with "high probability" the function f changes fast (see Figure 1), which is to say that f is more sensitive around/for them, hence preserving most of the uncertainty of X.

§8.2.1 The Fly Visual System

We turn to discuss a work by Simon Laughlin that analyzes the response of first-order interneurons in the fly's compound eye. These neurons are similar to the bipolar cells in the human retina. They have continuous input/output function (graded output, not spiking), responding to contrast (illumination) fluctuations. We will assume that the input to these cells is proportional to the illumination level, and will investigate their response pattern. In the experiment, Laughlin first habituated the fly to a fixed illumination level $\bar{I}$, and then presented a fluctuation/transient change in the illumination ΔI while recording the neuron's response. Independent of this experiment, Laughlin recorded contrast levels in many natural scenes such as "dry sclerophyll woodland and lakeside vegetation". He then calculated the empirical CDF of contrast/illumination

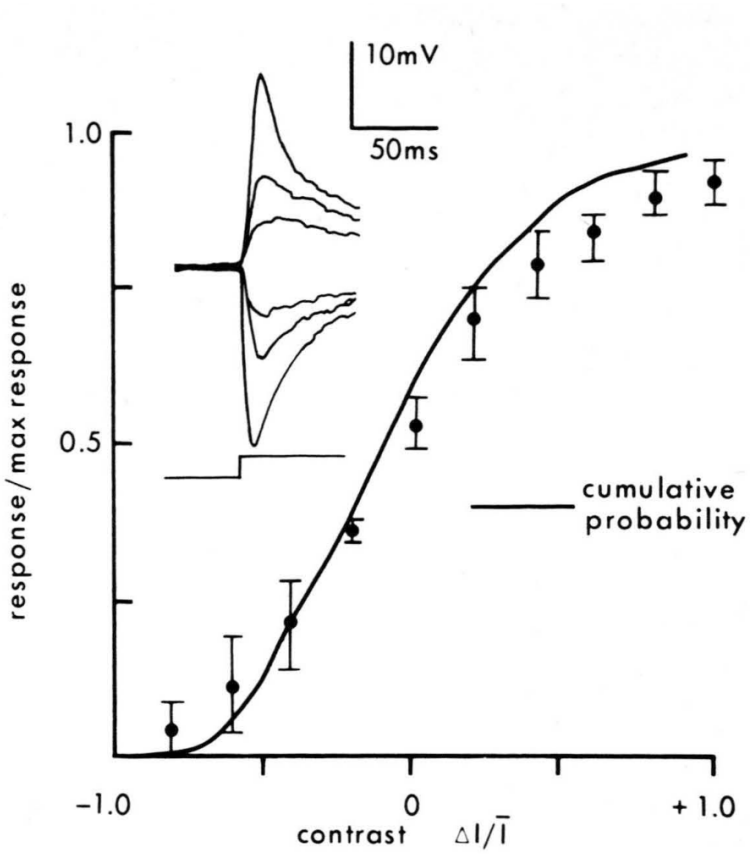

Figure 3: Fly Visual System Response

intensity in these natural scenes, to get an approximation of the distribution of the stimulus.

The results of the experiment are shown above in 8.2.1. As we can see, there is a strong correspondence between then neural response and the statistics of the natural stimulus, in a manner predicted by considerations of efficient coding. Importantly, note that this result involved no tuning of any parameter, the fit/correspondence was simply observed in the measured data.

Criticism of the Analysis

We started with the hypothesis that the neuron is trying to maximize the (differential) entropy of the response. At first, this seem natural, as we are interested in maximizing the information of the output. However, a closer examination reveals some problems. Note, that as long as f is a one-to-one function, then given R we know, with probability 1 , what X is. Therefore, no information is lost, and so as long as f is one-to-one (which it is in our case since we assume it to be monotonic) the shape of f doesn't matter at all, and our result about the "optimal" f become almost meaningless. What is crucially lacking so far from our analysis is noise. If we assume that the neural response is not a perfect deterministic function of the input, but some noisy version of it $R = f(X) + \xi$ then it would be natural to expect some specific form of an optimal f which will be more sensitive around values of X that occur with high probability. We would like to develop a formalism with which such revised model can be analyzed. To do so, we will present a new information-theoretic quantity, namely the mutual information.

§8.2.2 Mutual Information

Definition 8.6 (Mutual Information)

Let X and Y be two random variables. The mutual information between X and Y is defined as:

$$\mathcal{I}(X;Y) = \mathcal{H}(X) - \mathcal{H}(X \mid Y) \tag{8.2}$$

Equivalently the Mutual Information also admits the followinf equations

$$\mathcal{I}(X;Y) = \mathcal{H}(X) + \mathcal{H}(Y) - \mathcal{H}(X,Y) \tag{8.3}$$

$$\mathcal{I}(X;Y) = \left\langle \ln \frac{p\,(x,y)}{p(x)\cdot p(y)} \right\rangle_{p(x,y)} \tag{8.4}$$

Intuitively, $\mathcal{H}(Y)$ is the number of bits required to encode Y, and $\mathcal{H}(Y \mid X)$ is the average number of bits required to encode Y when X is known - that is, the information in Y which isn't included in X. Therefore, the difference $\mathcal{I}(X;Y)$ stands for the information in Y that is included in X.

§8.2.3 Properties of Mutual Information

The following properties hold for the mutual information of two variables X, Y:

1. **Symmetry**: $\mathcal{I}(X;Y) = \mathcal{I}(Y;X)$
2. **Non-negativity**: $\mathcal{I}(X;Y) \geq 0$
3. **Triviality**$\mathcal{I}(X;Y) = 0$ if and only if X and Y are independent
4. **Bounded** $\mathcal{I}(X;Y) \leq H(X)$ and $\mathcal{I}(X;Y) \leq H(Y)$

Example

Consider the following model: $X \sim \mathcal{N}\left(0, \sigma_X^2\right), \xi \sim \mathcal{N}\left(0, \sigma_\xi^2\right)$ (X and ξ are independent) and $Y = X + \xi$. Note that this implies $Y \sim \mathcal{N}\left(0, \sigma_X^2 + \sigma_\xi^2\right)$. We will treat X as the "signal" and ξ as the "noise" both of which compose Y. Finally, note that $(Y \mid X) \sim \mathcal{N}\left(0, \sigma_\xi^2\right)$. We are interested in $\mathcal{I}(X;Y)$, and we will develop it using the differential entropy of a Gaussian variable that we discussed in the previous lessons. We saw that if $Z \sim \mathcal{N}\left(\mu, \sigma^2\right)$ then $H(Z) = \frac{1}{2}\ln\left(2\pi e\sigma^2\right)$. In our case then:

$$\begin{aligned}
\mathcal{I}(X;Y) &= H(Y) - H(Y \mid X) \\
&= \frac{1}{2}\ln\left(2\pi e\left(\sigma_x^2 + \sigma_\xi^2\right)\right) - \frac{1}{2}\ln\left(2\pi e\sigma_\xi^2\right) \\
&= \frac{1}{2}\ln\left(\frac{2\pi e\left(\sigma_x^2 + \sigma_\xi^2\right)}{2\pi e\sigma_\xi^2}\right) \\
&= \frac{1}{2}\ln\left(1 + \left(\frac{\sigma_x}{\sigma_\xi}\right)^2\right)
\end{aligned}$$

Remark:Thinking about $\left(\frac{\sigma_x}{\sigma_\xi}\right)$ as the Signal to Noise Ratio (SNR), we can analyze two extreme cases:

- With high SNR, $\left(\frac{\sigma_x}{\sigma_\xi}\right) \to \infty$ we get $\mathcal{I}(X;Y) \to \ln \frac{\sigma_x}{\sigma_\xi}$. We can think of this result as if the noise is inducing effective "binning" of the variable of size σ_ξ, because with this noise we can only know X up to perturbations coming from ξ which are typically of $O(\xi)$.
- With low SNR $\left(\frac{\sigma_x}{\sigma_\xi}\right) \to 0$ we get $\mathcal{I}(X;Y) \to \frac{1}{2}\left(\frac{\sigma_x}{\sigma_\xi}\right)^2$ (because $\ln(1+x) \xrightarrow{x\to 0} x$). This means that the mutual information goes to 0 , and we can tell how fast. Again this should be intuitively clear - if Y is fully dominated by the noise then we cannot hope to retrieve any information about X from it.

§8.2.4 Revising the Model for Efficient Coding Using Mutual Information

We now have the tools to re-formulate the working model for efficient coding of a single variable. As before, we will have an input variable $X \sim p_X$. We will denote $Y = f(X)$, but this time we will assume that the response itself, R is $R = f(X) + \xi$ where $\xi \sim \mathcal{N}\left(0, \sigma^2(Y)\right)$. As before, our goal is to find an optimal f that maximizes $\mathcal{I}(X;R)$ under the constraint $f(x) \in [0, r_{\max}]$. In other words, we want to find the best mapping that maximizes the information that the response contains about the input (rather than simply maximizing the information in the response). Note that if we assume $\sigma^2(Y)$ to be independent of Y, that is, a constant/uniform noise for is added independent of the input, then we will get back the original version of the model, with maximizing the differential entropy of R. In light of the previous comments, this can be thought of as a working with uniform binning of R. However, if the noise does depend on $f(X)$, the results would be different.

We have variable x with probability p(x) :

$$x \to R = \underbrace{f(x)}_{y} + \xi$$

Where ξ is normally distributed noise with $\langle \xi \rangle = 0$ and variance that depends on the output y $\left(\text{var}(\xi) = \sigma^2(y)\right)$. Our goal is to maximize $\mathcal{I}(X;R)$ under the constraint that $f(x) \in [0, R_{\max}]$. 2

$$\begin{aligned}\mathcal{I}(X;R) &= S[R] - \langle S[R \mid X] \rangle_{p(x)} \\ &= S[R] - \frac{1}{2}\int dx p(x) \ln\left(2\pi e \sigma^2(f(x))\right)\end{aligned}$$

As the distribution $(R \mid x) \sim \mathcal{N}\left(f(x), \sigma^2(y)\right)$. In general, optimizing this quantity may be complicated. We will consider a particular case that we can understand analytically. 2.2 The low noise limit In this case $\sigma^2 \to 0$. We still allow σ^2 to reach zero in a non-uniform way for the different values of y, but we assume its sufficiently small. How it the mutual information effected by this assumption?

$$\mathcal{I}(X;R) = S[R] - \frac{1}{2}\int dx p(x) \ln\left(2\pi e \sigma^2(f(x))\right)$$

The first term: Note that $R = y + \mathcal{O}(\sigma)$, and this imply that as we take the limit of $\sigma^2 \to 0$:

$$p(R) \underset{\sigma^2\to 0}{\to} p(y) \Rightarrow S[R] \underset{\sigma^2\to 0}{\to} S[Y]$$

The second term: We can see that it is not possible to just ignore σ^2 as $\mathcal{I}(X;Y) \propto \ln\left(\sigma^2\right)$ and this diverges for small σ^2. We can change variables:

$$p_x(x)dx = p_y(y)dy$$

And combining this with the approximation for the first term we get:

$$\mathcal{I}(X;R) \underset{\sigma^2 \to 0}{\to} S[Y] - \frac{1}{2}\int dy p(y) \ln\left(2\pi e \sigma^2(y)\right)$$

This is a functional of $p(y)$ that we can optimize:

$$\begin{aligned}\mathcal{L} &= \mathcal{I}(X;R) - \lambda \underbrace{\left[\int p(y)dy - 1\right]}_{\text{constraint}} \\ &= -\int p(y)\ln(p(y))dy - \frac{1}{2}\int dy p(y) \ln\left(2\pi e \sigma^2(y)\right) - \lambda\left[\int p(y)dy - 1\right]\end{aligned}$$

We can do a functional differentiation:

$$\frac{\partial \mathcal{L}}{\partial p(y)} = 0 \Rightarrow -\ln(p(y)) - 1 - \frac{1}{2}\ln\left(2\pi e\sigma^2(y)\right) - \lambda = 0$$

We can find $p(y)$

$$\ln(p(y)) = -\ln(\sigma) + \text{ const}$$

$$p(y) = \frac{1}{z}\frac{1}{\sigma(y)}$$

Where $\frac{1}{z}$ is a normalization term (set such that $\int p(y)dy = 1$). We still need to find the input-output relation $f(x)$ that will generate this $p(y)$. This is similar to what we did for the optimization of the differential entropy. In the case of the differential entropy we got that $p(y)$ should be constant. In this case, the condition for uniform $p(y)$ is that σ is independent of Y. This makes sense because we discussed the fact that the noise has similar interpretation as the discretization of Y values. If the noise is uniform, when we measure R we know something about Y with equal error for the entire range of Y values. This is similar to uniform discretization. This is why the differential entropy which implicitly implies uniform discretization led us to the same result.

If the noise depend on Y what we get is that under the optimal transfer function, the uniformity is over $p(y)\sigma(y) = \text{const}$. Intuitively this makes sense. It tells us that the encoder assign higher probability for the bins with less noise as these bins are more informative.

§8.3 Efficient Coding of Multiple Input Output Variable

Let's consider a system that has multiple input variables $\bar{x} = (x_1, x_2, \ldots, x_N)^\mathsf{T}$ and multiple output variables $\bar{y} = (y_1, y_2, \ldots, y_M)^\mathsf{T}$. We wil assume that the inputs follow a Gaussian statistics with 0 mean and covariance matrix C. The model will have a linear input-output relation with output noise.

$$y_i = \sum_j W_{ij}x_j + z_i^{\text{out}} \tag{8.5}$$

$$\text{or,} \quad \bar{y} = W^{\mathsf{T}}\bar{x} + \bar{z}^{\text{out}} \tag{8.6}$$

where W is a $M \times N$ matrix and $\bar{z}^{\text{out}}$ is a $M \times 1$ vector of output noise. The output noise is gaussian with 0 mean and covariance matrix ΔI.

Our goal here is to optimze the mutual information between input and output. But the problem is not well formulated yet. For that we will need a constraint on the model. We will generally impose a constraint on the second moment of the output. Let's look at this with a simplified case of $M = 1$.

§8.3.1 Multiple Input Single Output

Example

Here, the output is a single variable y and the matrix W becomes a vector.

$$y = \sum_j w_j x_j + z^{\text{out}} = \bar{w}^{\mathsf{T}}\bar{x} + z^{\text{out}} \tag{8.7}$$

Our constraint is that

$$\langle y^2 \rangle = q \tag{8.8}$$

We can start by calculating the variance (also second moment because y has 0 mean).

$$\begin{aligned}
\langle y^2 \rangle &= \left\langle \left(\bar{w}^{\mathsf{T}}\bar{x} + z^{\text{out}} \right)^2 \right\rangle \\
&= \left\langle \bar{w}^{\mathsf{T}}\bar{x}\bar{x}^{\mathsf{T}}\bar{w} + 2\bar{w}^{\mathsf{T}}\bar{x}z^{\text{out}} + z^{\text{out}}z^{\text{out}} \right\rangle \\
&= \bar{w}^{\mathsf{T}} \left\langle \bar{x}\bar{x}^{\mathsf{T}} \right\rangle \bar{w} + 2\bar{w}^{\mathsf{T}} \left\langle \bar{x}z^{\text{out}} \right\rangle + \left\langle z^{\text{out}}z^{\text{out}} \right\rangle \\
&= \bar{w}^{\mathsf{T}} C \bar{w} + \Delta
\end{aligned}$$

Now, to calulate the mutual information we need to calculate the conditional entropy of y given x which we can do by first finding the variance of the conditional distribution of y given $\bar{x}$. But when $\bar{x}$ is fixed the variance only comes from the noise and will be equal to the noise variance. So, by using the formula for the entropy of gaussians we find the mutual information as:

$$\begin{aligned}
\mathcal{I}(\bar{x}; y) &= \frac{1}{2} \ln \left(\frac{\langle y^2 \rangle}{\langle z^{\text{out}} \rangle} \right) \\
&= \frac{1}{2} \ln \left(\frac{\bar{w}^{\mathsf{T}} C \bar{w} + \Delta}{\Delta} \right)
\end{aligned}$$

Recall that $\langle y^2 \rangle = \bar{w}^{\mathsf{T}} C \bar{w} + \Delta = q$ implying that

$$\mathcal{I}(\bar{x}; y) = \frac{1}{2} \ln \left(\frac{q}{\Delta} \right) \tag{8.9}$$

$$\tag{8.10}$$

So, we end up finding that the mutual information is independent of the weights. What went wrong? **The constraint only determines the magnitude of $\bar{w}$.** Let's look at the constraint in more detail. Suppose that one chooses a vector $\bar{w}$ with a certain direction $\hat{w}$, such that $\|\hat{w}\| = 1$($\hat{w}$ is normalized) and an undetermined magnitude:

$$\bar{w} = \sqrt{\alpha}\hat{w}$$

We substitute $\bar{w}$ in the constraint:

$$\alpha\hat{w}^{\mathsf{T}}C\hat{w} + \Delta = q \rightarrow \alpha = \frac{q - \Delta}{\hat{w}^{\mathsf{T}}C\hat{w}}$$

We can see that the constraint determines that magnitude of $\bar{w}$ (the prefactor $\sqrt{\alpha}$). In other words, we have the freedom to choose the direction of $\bar{w}$ as we wish, whereas the magnitude of $\bar{w}$ is determined by the constraint.

§8.3.2 Modified model

But we want the choice of $\bar{w}$ to affect the mutual information between the output unit y and the inputs $\bar{x}$. For this purpose, we should modify our model, such that the input would be also corrupted by noise. We assume that the noise in the input is isotropic (same noise in all input channels), such that it is distributed in a circular manner, as depicted by the green circle in the above figure. Then, the SNR at the input level (the ratio between the standard deviation of the input and the standard deviation of the input noise) is the largest for the largest radius of the ellipse. Moreover, when expanding the projection to satisfy the constraint, we expand the noise together with the signal, such that the smaller the radius of the ellipse at a certain direction, the more the input noise would be expanded. Thus, in this modified model the direction of $\bar{w}$ does affect the mutual information between y and $\bar{x}$. Accordingly, the best choice of $\bar{w}$ would be at the direction with the largest variance (the first principal component), since it has the largest SNR and its input noise is expanded the least.

Let's first formulate the model in the general case, for multiple inputs and multiple output units.

1. Inputs: $x_i + z_i^{\text{in}}$ for $i = 1, \ldots, N$
2. Outputs: $y_i = \left(W^{\mathsf{T}}\left(\bar{x} + \bar{z}^{\text{in}}\right)\right)_i + z_i^{\text{out}} \qquad \text{for } i = 1, \ldots, M$
3. Assumptions about the input noise:

$$\left\langle z_i^{\text{in}} \right\rangle = 0 \quad \left\langle z_i^{\text{in}}\, z_j^{\text{in}} \right\rangle = \delta_{ij}\varepsilon$$

4. Assumptions about the output noise:

$$\left\langle z_i^{\text{out}} \right\rangle = 0 \quad \left\langle z_i^{\text{out}}\, z_j^{\text{out}} \right\rangle = \delta_{ij}\Delta$$

As before, let's calulate the covariance of the output units We will calculate the covariance $M \times M$ matrix of the output units $\bar{y}$ (over realizations of the input noise, the output

noise and the input):

$$\begin{aligned}\left\langle \bar{y}\bar{y}^{\mathsf{T}} \right\rangle &= \left\langle \left(W^{\mathsf{T}} \left(\bar{x} + \bar{z}^{\text{in}} \right) + \bar{z}^{\text{out}} \right) \left(W^{\mathsf{T}} \left(\bar{x} + \bar{z}^{\text{in}} \right) + \bar{z}^{\text{out}} \right)^{\mathsf{T}} \right\rangle \\ &= \left\langle \left(W^{\mathsf{T}} \left(\bar{x} + \bar{z}^{\text{in}} \right) + \bar{z}^{\text{out}} \right) \left(\left(\bar{x} + \bar{z}^{\text{in}} \right)^{\mathsf{T}} W + \left(\bar{z}^{\text{out}} \right)^{\mathsf{T}} \right) \right\rangle \\ &= \left\langle \left(W^{\mathsf{T}} \bar{x} + W^{\mathsf{T}} \bar{z}^{\text{in}} + \bar{z}^{\text{out}} \right) \left(\bar{x}^{\mathsf{T}} W + \left(\bar{z}^{\text{in}} \right)^{\mathsf{T}} W + \left(\bar{z}^{\text{out}} \right)^{\mathsf{T}} \right) \right\rangle\end{aligned}$$

The cross terms are canceled due to the independence between variables: the input and the input noise are independent, the input and the output noise are independent, the input noise and the output noise are independent.

$$\begin{aligned}\left\langle \bar{y}\bar{y}^{\mathsf{T}} \right\rangle &= W^{\mathsf{T}} \left\langle \bar{x}\bar{x}^{\mathsf{T}} \right\rangle W + W^{\mathsf{T}} \left\langle \bar{z}^{\text{in}} \left(\bar{z}^{\text{in}} \right)^{\mathsf{T}} \right\rangle W + \left\langle \bar{z}^{\text{out}} \left(\bar{z}^{\text{out}} \right)^{\mathsf{T}} \right\rangle \\ &= W^{\mathsf{T}} C W + W^{\mathsf{T}} \varepsilon I W + \Delta I \\ &= W^{\mathsf{T}} C W + \varepsilon W^{\mathsf{T}} W + \Delta I\end{aligned}$$

§8.3.3 Modified model with a single output unit

Example

We can get the variance of the output unit from the general case that we solved earlier:

$$\left\langle y^2 \right\rangle = \bar{w}^{\mathsf{T}} C \bar{w} + \varepsilon \bar{w}^{\mathsf{T}} \bar{w} + \Delta$$

The constraint on the variance:

$$\left\langle y^2 \right\rangle = \bar{w}^{\mathsf{T}} C \bar{w} + \varepsilon \bar{w}^{\mathsf{T}} \bar{w} + \Delta = q$$

As we did before, we choose a direction for $\bar{w}$ to project the input vector $\bar{x}$ on this direction:

$$\bar{w} = \sqrt{\alpha} \hat{w}$$

We substitute $\bar{w}$ in the constraint and find the prefactor α :

$$\begin{aligned}\alpha \hat{w}^{\mathsf{T}} C \hat{w} + \alpha \varepsilon \underbrace{\hat{w}^{\mathsf{T}} \hat{w}}_{=1} &= q - \Delta \\ \rightarrow \alpha &= \frac{q - \Delta}{\hat{w}^{\mathsf{T}} C \hat{w} + \varepsilon}\end{aligned}$$

We calculate the mutual information between the inputs $\bar{x}$ and the output unit:

$$\begin{aligned}\mathcal{I}(y; \bar{x}) &= S(y) - S(y \mid \bar{x}) \\ &= \frac{1}{2} \ln \left(2\pi e \left\langle y^2 \right\rangle \right) - \frac{1}{2} \ln \left(2\pi e \left(\varepsilon \bar{w}^{\mathsf{T}} \bar{w} + \Delta \right) \right) \\ &= \frac{1}{2} \ln \left(\frac{\bar{w}^{\mathsf{T}} C \bar{w} + \varepsilon \bar{w}^{\mathsf{T}} \bar{w} + \Delta}{\varepsilon \bar{w}^{\mathsf{T}} \bar{w} + \Delta} \right)\end{aligned}$$

We note that the entropy of y conditioned on $\bar{x}$, $S(y \mid \bar{x})$, is affected by both types of noise; the input noise and the output noise. We substitute the constraint and the

chosen $\bar{w}$:

$$\begin{aligned}
\mathcal{I}(y;\bar{x}) &= \frac{1}{2}\ln\left(\frac{q}{q-\alpha\hat{w}^{\mathsf{T}}C\hat{w}}\right) \\
&= \frac{1}{2}\ln\left(\frac{q}{q-\left(\frac{q-\Delta}{\hat{w}^{\mathsf{T}}C\hat{w}+\varepsilon}\right)\hat{w}^{\mathsf{T}}C\hat{w}}\right) \\
&= \frac{1}{2}\ln\left(\frac{q-\left(\frac{q-\Delta}{\hat{w}^{\mathsf{T}}C\hat{w}+\varepsilon}\right)\hat{w}^{\mathsf{T}}C\hat{w}+\left(\frac{q-\Delta}{\hat{w}^{\mathsf{T}}C\hat{w}+\varepsilon}\right)\hat{w}^{\mathsf{T}}C\hat{w}}{q-\left(\frac{q-\Delta}{\hat{w}^{\mathsf{T}}C\hat{w}+\varepsilon}\right)\hat{w}^{\mathsf{T}}C\hat{w}}\right) \\
&= \frac{1}{2}\ln\left(1+\frac{\frac{(q-\Delta)\hat{\hat{w}}^{\mathsf{T}}C\hat{w}}{\hat{w}^{\mathsf{T}}C\hat{w}+\varepsilon}}{\frac{q\hat{w}^{\mathsf{T}}C\hat{w}+q-q\hat{w}^{\mathsf{T}}C\hat{w}+\Delta\hat{w}^{\mathsf{T}}C\hat{w}}{\hat{w}^{\mathsf{T}}C\hat{w}+\varepsilon}}\right) \\
&= \frac{1}{2}\ln\left(1+(q-\Delta)\frac{\hat{w}^{\mathsf{T}}C\hat{w}}{q\varepsilon+\Delta\hat{w}^{\mathsf{T}}C\hat{w}}\right)
\end{aligned}$$

We can see that the mutual information is only a function of $\hat{w}$ (the direction of $\bar{w}$). Moreover, the mutual information is a monotonically increasing function of $\hat{w}^{\mathsf{T}}C\hat{w}$ (since $q-\Delta>0$ and $q\varepsilon>0$). Thus, we can reduce the maximization problem of the mutual information to maximizing $\hat{w}^{\mathsf{T}}C\hat{w}$ only. In other words, we want to find the direction $\hat{w}$ that maximizes $\hat{w}^{\mathsf{T}}C\hat{w}$. We note that $\bar{w}^{\mathsf{T}}C\bar{w}$ is the variance of the projection $\left(\left\langle\left(\hat{w}^{\mathsf{T}}\bar{x}\right)^2\right\rangle=\left\langle\hat{w}^{\mathsf{T}}\bar{x}\bar{x}^{\mathsf{T}}\hat{w}\right\rangle=\hat{w}^{\mathsf{T}}\left\langle\bar{x}\bar{x}^{\mathsf{T}}\right\rangle\hat{w}=\hat{w}^{\mathsf{T}}C\hat{w}\right)$. Thus, in order to maximizes $\bar{w}^{\mathsf{T}}C\bar{w}$, we choose the direction with the largest variance (the longest radius- depicted in pink in the figure below), which provides the largest SNR. The $\bar{w}$ that maximize $\bar{w}^{\mathsf{T}}C\bar{w}$ is in the direction of the first principal component of $\bar{x}$ (the principal eigenvector of the covariance matrix C)

§8.4 Dimension Reduction

Neural Data is usually high dimensional either due to the number of neurons being recorder or due to the length of time they are recorded. Understanding the data as is is not always possible. Dimension reduction is a technique that allows us to reduce the dimensionality by projecting onto a low dimensional subspace of the data while preserving as much information as possible. There is a ontological reasoning behind it which says that neural data is actually low dimensional and we would like to uncover the underlying structure in it. Alternatively, one can take a weaker philosophical position and use it merely as a methodological tool to understand experimental data. Whichever disposition you might have, these techniques are bound to be useful.

§8.4.1 Principal Component Analysis (PCA)

We start with a N dimensional data where $N\gg 1$. We would ideally like to project the data linearly into some subspace $\mathcal{S}$ with dimension $D\ll N$. Such projections are characterized by an application of a $D\times N$ matrix to the N dimensional data.

How can we choose criterias that will let us find an "optimal" subspace for the projection.

1. We can try to minimize the reconstruction error. Consider data x from some 0 mean distribution. First, we project x into a low dimensional subspace by $\mathbf{W}x$ where $\mathbf{W}$ is a $D\times N$ matrix. We will choose $\mathbf{W}$ to be orthonormal matrix. Then

we can reconstruct the data by $\mathbf{W}^\intercal\mathbf{W}x$. The reconstruction error is then given by $\|x - \mathbf{W}^\intercal\mathbf{W}x\|^2$. So, we can now write the objective for P points as:

$$\min_W \frac{1}{P}\sum_{i=1}^{P}\|x_i - \mathbf{W}^\intercal\mathbf{W}x_i\|^2 \tag{8.11}$$

2. Alternately we can ask that the reconstruction will have as high a variance as possible so that we will be able to capture the variability of the original data. This will lead to the following objective:

$$\max_W \frac{1}{P}\sum_{i=1}^{P}\left\|\mathbf{W}^\intercal\mathbf{W}x_i\right\|_2^2 \tag{8.12}$$

It is a simple exercise to show that in fact the two objectives are identical and both lead to what we call the Principal Component Analysis (PCA). The PCA solution picks a set of orthonormal vectors that maximize the variance of the projected data. The first vector will be the direction of maximum variance, the second will be orthogonal to the first and will have the maximum variance among all vectors that are orthogonal to the first and so on. The number of vectors that we choose is the dimension of the subspace we want to project the data into. The projection matrix is then given by the first D unit eigenvectors of the covariance matrix Σ.

§8.4.2 $D = 1$ or a simple Autoencoder

For $D = 1$, we are fitting a unit vector $\mathbf{w}$, and the code is a scalar $z^{(i)} = \mathbf{w}^\top\left(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}\right)$. Let's maximize the projected variance. From observation 1, we have

$$\begin{aligned}
\frac{1}{P}\sum_i\left\|\tilde{\mathbf{x}}^{(i)} - \hat{\boldsymbol{\mu}}\right\|^2 = \frac{1}{P}\sum_i\left[z^{(i)}\right]^2 &= \frac{1}{P}\sum_i\left(\mathbf{w}^\top\left(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}\right)\right)^2 \\
&= \frac{1}{P}\sum_{i=1}^{P}\mathbf{w}^\top\left(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}\right)\left(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}\right)^\top\mathbf{w} \\
&= \mathbf{w}^\top\left[\frac{1}{P}\sum_{i=1}^{P}\left(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}\right)\left(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}\right)^\top\right]\mathbf{w} \\
&= \mathbf{w}^\top\hat{\boldsymbol{\Sigma}}\mathbf{w} \\
&= \mathbf{w}^\top\mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top\mathbf{w} \\
&= \mathbf{a}^\top\boldsymbol{\Lambda}\mathbf{a} \\
&= \sum_{j=1}^{D}\lambda_j a_j^2
\end{aligned}$$

where $\mathbf{a} = \mathbf{Q}^\top\mathbf{w}$. Now, assume that the eigenvalues are sorted in order such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_D$ and $\mathbf{a}$ is norm 1 So the choice that maximizes the above quantity is when $a_1 = 1$ and $a_i = 0 \quad \forall i \neq 1$. Hence we get that $\mathbf{w} = \mathbf{Q}\mathbf{a} = q_1$, the top eigen vector. A similar argument where we project data to the $N - 1$ dimensions orthogonal to the principal eigen vector and perform the above analysis will tell us that if we have D hidden units we will pick up the top D eigen vectors.

For $D > 1$ the dimensions of hiddent representation $\mathbf{z}$ are decorrelated.

$$\begin{aligned}\text{Cov}(\mathbf{z}) &= \text{Cov}\left(\mathbf{W}^\top(\mathbf{x}-\boldsymbol{\mu})\right)\\ &= \mathbf{W}^\top \text{Cov}(\mathbf{x})\mathbf{W}\\ &= \mathbf{W}^\top\boldsymbol{\Sigma}\mathbf{W}\\ &= \mathbf{W}^\top\mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top\mathbf{W}\\ &= \begin{pmatrix}\mathbf{I} & \mathbf{0}\end{pmatrix}\boldsymbol{\Lambda}\begin{pmatrix}\mathbf{I}\\ \mathbf{0}\end{pmatrix}\\ &= \text{ top left } D\times D \text{ block of } \boldsymbol{\Lambda}\end{aligned}$$

Furthermore, the value of minimum MSE is

$$\text{MSE}_{\text{min}} = \sum_{i=D+1}^{N}\lambda_i \tag{8.13}$$

i.e., the sum of the eigenvalues of the orthogonal subspace to the PCA subspace. Indeed an interesting and simple hebbian learning rule in the brain can give rise to PCA.

§8.4.3 Oja's rule

Suppose we have a neuron (y) that receives n inputs ($\bar{x}$), and there's a freedom to adjust the weights ($\bar{w}$) between this neuron and it's input. We will present a local plasticity rule (that depends on the presynaptic and postsynaptic activity) that will make the neuron learn to extract the first PC.

The learning rule: the neuron is presented with inputs sequentially and after a presentation we change the weights according to-

$$\Delta w_i = w_i^{(n+1)} - w_i^{(n)} = \eta y^{(n)} x_i^{(n)} \tag{8.14}$$

This is a Hebbian learning rule. The change in the synapse is proportional to the activity of the postsynaptic neuron and the presynaptic neuron. But as is the problem with this learning rule is that if we would apply it the weights will either explode or go to zero. We mostly care about the direction of $\bar{w}$ but it doesn't make sense for the weights to explode. We will rewrite the rule:

$$w_i^{(n+1)} = \frac{w_i^{(n)} + \eta y^{(n)} x_i^{(n)}}{\sqrt{\Sigma_j\left(w_j^{(n)} + \eta y^{(n)} x_j^{(n)}\right)^2}} \tag{8.15}$$

But this imply that there is some interaction between the different weights and the learning is not as local as before. We can handle this by taking a **small η limit.**

$$\begin{aligned}\frac{1}{\sqrt{\Sigma_j\left(w_j^{(n)} + \eta y^{(n)} x_j^{(n)}\right)^2}} &\simeq \frac{1}{\sqrt{\underbrace{\Sigma_j w_j^{(n)^2}}_{=1} + 2\eta\Sigma_j \underbrace{y^{(n)} w_j^{(n)} x_j^{(n)}}_{y^{(n)2}}}} =\\ &= \frac{1}{\sqrt{1+2\eta y^{(n)2}}} \simeq 1 - \eta y^{(n)2}\end{aligned}$$

Where we used the fact the due to the normalization $\Sigma_j w_j^{(n)^2} = 1$. The learning rule under this approximation:

$$\begin{aligned} w_i^{(n+1)} &\simeq \left(w_i^{(n)} + \eta y^{(n)} x_i^{(n)}\right)\left(1 - \eta y^{(n)2}\right) \\ &\simeq w_i^{(n)} + \eta y^{(n)} x_i^{(n)} - w_i^{(n)} \eta y^{(n)2} \end{aligned}$$

The average effect of this rule on the change in the weights:

$$\langle \Delta w_i \rangle = \eta \left[\langle y x_i \rangle - w_i \langle y^2 \rangle\right]$$

We are averaging over the possible realizations of x. We would like to average over time (over the presentations of the input). The problem is that as we present more examples the weights are constantly changing, and in the expression we assume that w_i is fixed. Thus, to average over time we need the learning rate to be sufficiently small. This way, the neuron has a chance to sample the distribution of the inputs over a timescale during which the weights are still not changing significantly. For slow learning we get the following synaptic dynamics:

$$\frac{dw_i}{dt} \propto \eta \left[\langle y x_i \rangle - w_i \langle y^2 \rangle\right] \tag{8.16}$$

Now, we can get to the steady state solution of this by:

$$\begin{aligned} \langle y x_i \rangle &= w_i \langle y^2 \rangle \\ \Sigma_j w_j \underbrace{\langle x_j x_i \rangle}_{C_{ij}} &= w_i \Sigma_{k,j} w_j w_k \underbrace{\langle x_j x_k \rangle}_{C_{jk}} \\ \Sigma_j w_j C_{ij} &= w_i \Sigma_j w_j C_{jk} w_k \\ C\bar{w} &= \left(\bar{w}^\mathsf{T} C \bar{w}\right) \bar{w} \end{aligned}$$

This implies that $\bar{w}$ is an eigen vector of the covariance matrix C. But which one? It turns out if one does a stability analysis or notes that the dynamics admits a lyapunov function implying that we will go to global minima, we will find that the eigen vector with the largest eigen value is the one we want. This is the first principal component.

§8.4.4 Linear Discriminant Analysis (LDA)

PCA is a linear transformation that maximizes the variance of the projected data. But there might be strcuture in the mean of the data in the sense that the data might be mixture of different types of object classes with labels $\mathcal{C} = \{c_1, \cdots, c_K\}$. In this case, we might want to project the data such that the projected data from different classes are as far apart as possible. This is the idea behind LDA. The goal is to find a linear transformation that maximizes the separation between the classes. The idea is to project the data into a subspace such that the projected data from different classes are as far apart as possible and we capture discriminative features. We have already come across similar idea when we studied bayesian discrimination in the case of exponential family.

Let's focus on the case where our data set is a mixture of two sets of labeled data i.e, $\mathcal{D}_1 = \{x_1 \dots x_{n_1}\}$ and $\mathcal{D}_2 = \{x_1 \dots x_{n_2}\}$. We can then define the class mean as:

$$m_i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_j^i \tag{8.17}$$

(8.18)

The scatter or the variance of the data as usual is given by :

$$\mathcal{S} = \sum_{x \in \mathcal{D}} (x - m)(x - m)^{\mathsf{T}} \tag{8.19}$$

The within class scatter is given by:

$$\mathcal{S}_w = \sum_{i=1}^{2} \sum_{j=1}^{n_i} (x_j^i - m_i)(x_j^i - m_i)^{\mathsf{T}} \tag{8.20}$$

And the between class scatter is given by:

$$\mathcal{S}_b = (m_2 - m_1)(m_2 - m_1)^{\mathsf{T}} \tag{8.21}$$

The Fisher Linear Discrimination criteria states that we want to maximize the ratio of the between class scatter to the within class scatter. That is we want to find a linear transformation W that creates a representation of the data by $\mathcal{Y}_i = W^{\mathsf{T}}\mathcal{D}_i$ that maximizes the following criterion:

$$\mathcal{J} = \frac{\mathcal{W}^{\mathsf{T}} S_b W}{W^{\mathsf{T}} \mathcal{S}_w W} \tag{8.22}$$

We find w by setting $dJ/dW = \mathbf{0}$:

$$\begin{aligned} \frac{dJ}{dW} = \mathbf{0} &\Leftrightarrow \left(W^{\top}\mathcal{S}_W W\right)\mathcal{S}_B W - \left(W^{\top}\mathcal{S}_B W\right)\mathcal{S}_W W = \mathbf{0} \\ &\Leftrightarrow \mathcal{S}_B W - J\mathcal{S}_W W = \mathbf{0} \\ &\Leftrightarrow \mathcal{S}_W^{-1}\mathcal{S}_B W - JW = \mathbf{0} \implies \mathcal{S}_W^{-1}\mathcal{S}_B W = JW \end{aligned}$$

This looks like a complicated expression but we can easily derive the direction of W by noting that $\mathcal{S}_b W \propto (m_2 - m_1)$ since it is a rank 1 matrix by construction. This implies that $W \propto \mathcal{S}_W \cdot (m_2 - m_1)$.

§8.5 Efficient Coding with Multiple Input and Output Neurons

Given what we have learned from a single output unite, it makes sense that similar to the 1D case, the optimal subspace is the first M Principle components (PCs) for M output units. In this subspace the SNR should be the highest. Firstly, We have the covariance matrix of the input

$$C\bar{u}^{(i)} = \lambda^{(i)}\bar{u}^{(i)}$$

where $\bar{u}^{(i)}$ are the principle axes and they are orthogonal and normalized: $\bar{u}^{(k)T}\bar{u}^{(l)} = \delta_{kl}$. Now, If we add the noise than the covariance is $C_{ij} + \epsilon\delta_{ij}$ with the same eigenvectors and eigenvalues: $\tilde{\lambda}^{(i)} = \lambda^{(i)} + \epsilon$. Then, to extract the coefficients of the first M PCs we can write the input as a linear combinations of the PC:

$$x_i + z_i^{in} = \sum_{k=1}^{N} c_k u_i^{(k)}$$

$$c_j = \bar{u}^{(j)T}\left(x_i + z_i^{in}\right) = \sum_{i=1}^{N} u_i^{(j)}\left(x_i + z_i^{in}\right)$$

We want to take the first M coefficients. The c_j we found are uncorrelated. In addition, multi-dimensional Gaussian under a linear transformation is still multi-dimensional Gaussian $\Rightarrow$ these coefficients are also Gaussian. Components of Gaussian that are uncorrelated are also independent. Thereforeit makes sense to encode each c_j separately:

$$W_{ij} = u_i^{(j)}\sqrt{\alpha_j} \text{ where } j = [1, \dots, M]$$

In 1-dim $\sqrt{\alpha}$ was set to match the constraint. Now we have M-dim output and the constraint is on the sum of the variances - so there is a question how many resources we invest in each input. Larger $\sqrt{\alpha_j}$ means more resources.

§8.5.1 Low Input Noise

Let's look at the limit of the input noise $\bar{z}^{\text{in}} \to 0$:

$$\bar{y} \simeq W^{\mathsf{T}}\bar{x} + \bar{z}^{\text{out}}$$

This is independent of the input noise, but $\bar{z}^{\text{in}}$ does effect the solution as it was the main reason that we choose to take the first M Principle component (for higher SNR). We can write the mutual information as:

$$\mathcal{I}(\bar{y}, \bar{x}) = \mathcal{H}(\bar{y}) - \mathcal{H}(\bar{y} \mid \bar{x}) = \mathcal{H}(\bar{y}) - \mathcal{H}\left(\bar{z}^{\text{out}}\right)$$

Thus, in this limit to maximize $\mathcal{I}(\bar{y}, \bar{x})$ we maximize $H(\bar{y})$. $\bar{y}$ is a multi-dimensional Gaussian variable, with constraint on the sum of variances of its inputs. To maximize the entropy its form should be spherically symmetric:

$$\langle y_j y_k \rangle = \delta_{jk} q = \begin{cases} 0 & j \neq k \\ \lambda_j \alpha_j + \Delta & j = k \end{cases} \quad ; \qquad \Rightarrow \alpha_j = \frac{q - \Delta}{\lambda_j} \equiv \frac{Q}{\lambda_j} \tag{8.23}$$

We can see that for smaller $\lambda \to$ larger α. We suppress the direction with larger variance, and by this we are equalizing the variance over the different directions. We do this since there is no input noise; we don't loose SNR by expanding directions with smaller variance.

§8.5.2 High Input Noise

Here the logic is opposite. We want to invest in the directions with larger variance:

1. Small $\lambda_i \to$ small SNR: less information about the input $\to$ suppress.
2. Large $\lambda_i \to$ large SNR: more information about the input $\to$ enhance.

Let's see this mathematically. We find:

$$\left\langle \overline{y}\overline{y}^{\mathsf{T}} \right\rangle = W^{\mathsf{T}} C W + \epsilon W^{\mathsf{T}} W + \Delta I$$

The output is a Gaussian so we want to maximize its entropy:

$$\mathcal{L} = \underbrace{\frac{1}{2}\ln\left(\det\left[W^{\mathsf{T}} C W + \epsilon W^{\mathsf{T}} W + \Delta I\right]\right)}_{H(\bar{y})} - \underbrace{\frac{1}{2}\ln\left(\det\left[\epsilon W^{\mathsf{T}} W + \Delta I\right]\right)}_{H(\bar{y}|\bar{x})}$$
$$\underbrace{-\frac{1}{2}\Lambda\left[\operatorname{Tr}\left[W^{\mathsf{T}} C W\right] + \epsilon \operatorname{Tr}\left[W^{\mathsf{T}} W\right] + M\Delta - MQ\right]}_{\text{constraint}}$$

This clealry is optimized by the PC solution but the solution space is degenrate. A unitary transformation is a rotation of the system the preserves the length of the vectors: $U^\mathsf{T}U = I$ and $\mathcal{L}$ is invariant to any unitary transformation of the output: $W^\mathsf{T} \to UW^\mathsf{T}$. Thus, the choice of the weights is not unique.

Theorem 8.7 (PC solution)

The **PC solution** for W has the following form:

$$W_{PC}^\mathsf{T} = \begin{pmatrix} \sqrt{\alpha_1} & & \\ & \ddots & \\ & & \sqrt{\alpha_M} \end{pmatrix} \underbrace{\begin{pmatrix} - & u^{(1)T} & - \\ \vdots & & \\ - & u^{(M)T} & - \end{pmatrix}}_{M \times N} \tag{8.24}$$

where: $\lambda_1 > .. > \lambda_M$ In this solution we encode each component separately and α_j depends on the parameters: $\lambda_1, .., \lambda_M, \epsilon, \Delta, q$ Furthermore, all other optimal solutions are of the form:

$$W^\mathsf{T} = UW_{PC}^\mathsf{T} \quad \text{where } U^\mathsf{T}U = I \tag{8.25}$$

Substituting the form of the solution to the equation we get a diagonal matrices:

$$\left(W_{PC}^\mathsf{T} C W_{PC}\right)_{jk} = \alpha_j \lambda_j \delta_{jk}$$
$$\left(W_{PC}^\mathsf{T} W_{PC}\right)_{jk} = \alpha_j \delta_{jk}$$

For diagonal matrices the determinant is the product of the elements on the diagonal. The Lagrangian:

$$\mathcal{L} = \frac{1}{2}\sum_{j=1}^{M} \ln\left(\alpha_j\lambda_j + \epsilon\alpha_j + \Delta\right) - \frac{1}{2}\ln\left(\epsilon\alpha_j + \Delta\right) - \frac{\Lambda}{2}\left(\sum_j \alpha_j\lambda_j + \epsilon\sum_j \alpha_j - MQ\right)$$

$$\text{When,} \quad \frac{\partial L}{\partial \alpha_i} = 0$$
$$\Downarrow$$
$$\frac{\lambda_i + \epsilon}{\left(\lambda_i + \epsilon\right)\alpha_i + \Delta} - \frac{\epsilon}{\epsilon\alpha_i + \Delta} - \Lambda\left(\lambda_i + \epsilon\right) = 0$$

For the PC solution we can see that each α_i depends on the corresponding λ_i. However, we need to remember that there is coupling between the different components that is coming from the constraint through Λ :

$$\sum_i \alpha_i\left(\lambda_i + \epsilon\right) = MQ$$

So we need to tune Λ to obey the constraint.