Probabilistic models for neural data: From single neurons to population dynamics

NEUROBIO 316QC

Jan Drugowitsch

jan_drugowitsch@hms.harvard.edu

Session 1: Course overview & Bayesian recap

Couse outline

Aim: Understand modular structure of probabilistic models

- Framework for thinking about models
- Reveals assumptions
- Supports changing/refining models

This course is *only a starting point!*

Structure

- This/next week: recap of/intro to Bayesian statistics
- Future sessions:
  - discussion of paper uses techniques of previous sessions
  - introduction of new techniques
- Between sessions:
  - Reading (Perusall) & preparing discussion (Google Slides)
  - exercises & brief write-up
  - completing quiz by noon of day of session

What to expect …

You should **not** expect to

become an expert in Bayesian modeling;

understand all the details of the discussed papers;

design and implement new models from ground up.

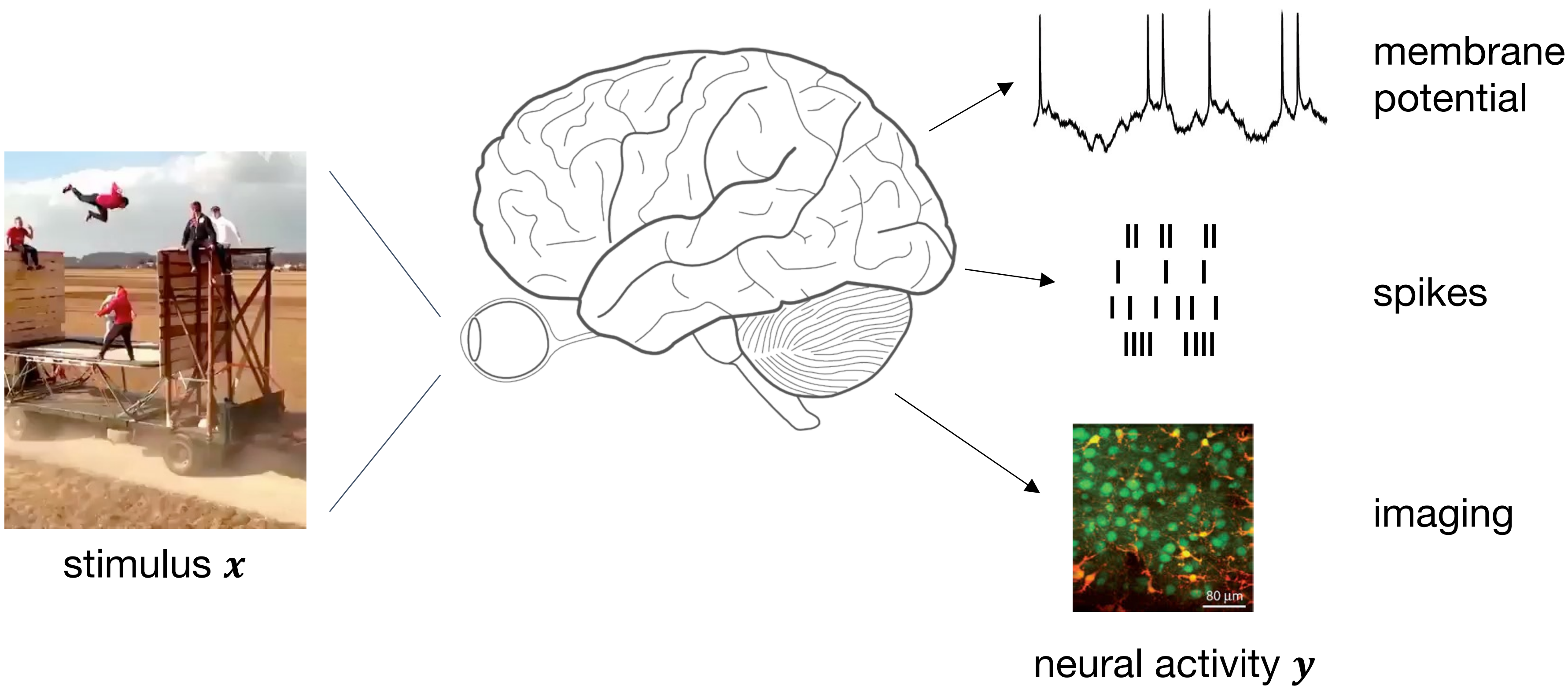
Ideally, you would learn to

understand structure of different Bayesian generative models for neural data, the associated graphical models and assumptions;

read up on new models and understand how they relate to existing models;

know what you would need to learn (and where you could find information) to design and implement new models.

Why build models?



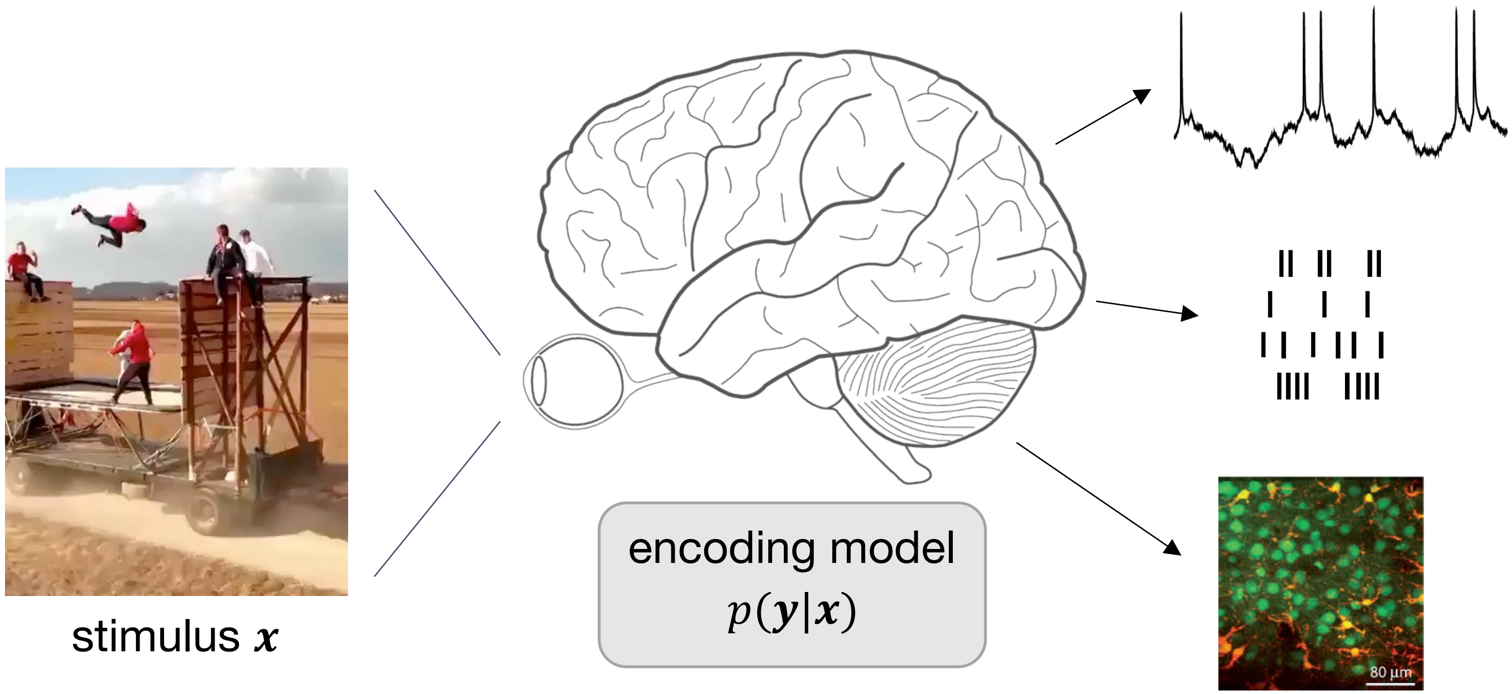
How are stimuli x encoded in neural activity y?

What aspects of neural activity carry information?

What can we/the brain say about the stimulus given neural activity?

(after Jonathan Pillow)

Why build models?



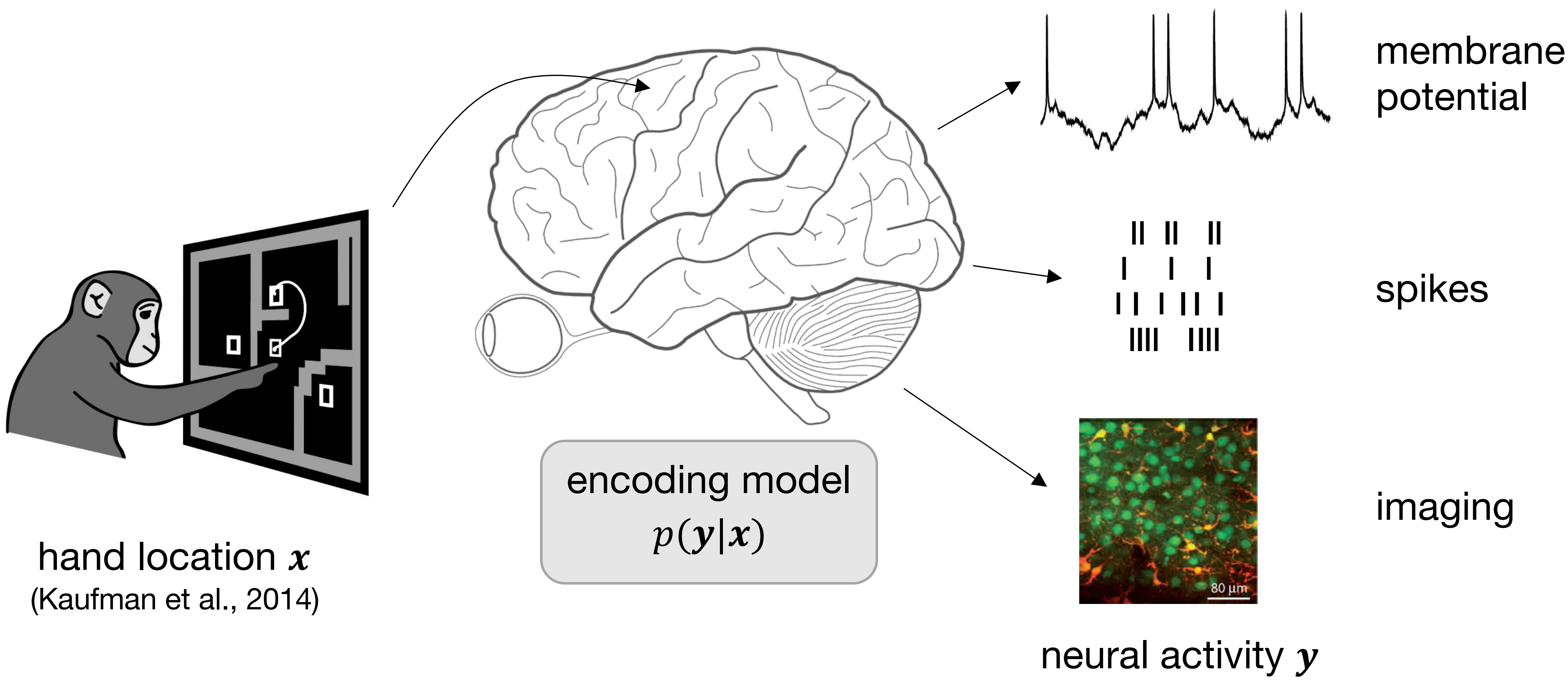
Build flexible statistical encoding model $p(\boldsymbol{y}|\boldsymbol{x})$

Quantify information carried in neural responses

Invert encoding model for decoding, $p(\boldsymbol{x}|\boldsymbol{y})$

(after Jonathan Pillow)

Why build models?

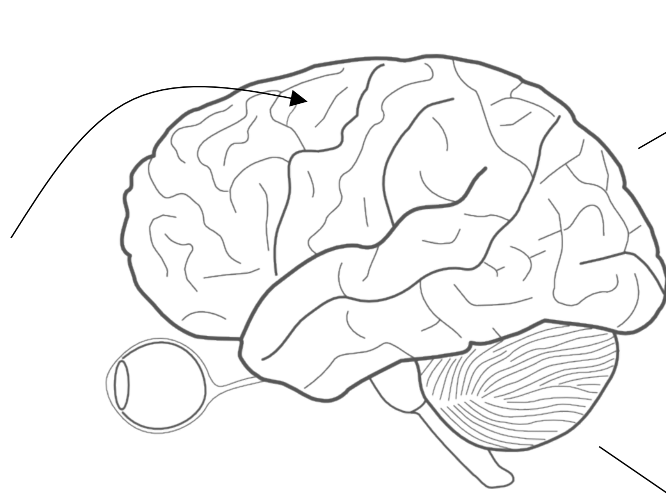


not restricted to sensory variables

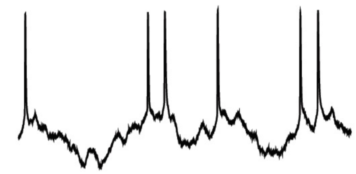
(after Jonathan Pillow)

Why build models?

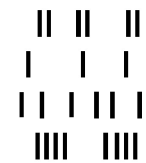
“external covariates” $\boldsymbol{x}$
(Hardcastle et al., 2015)



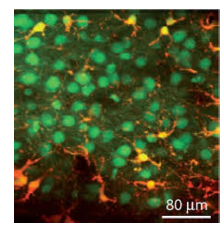
encoding model
$p(\boldsymbol{y}|\boldsymbol{x})$



membrane potential



spikes



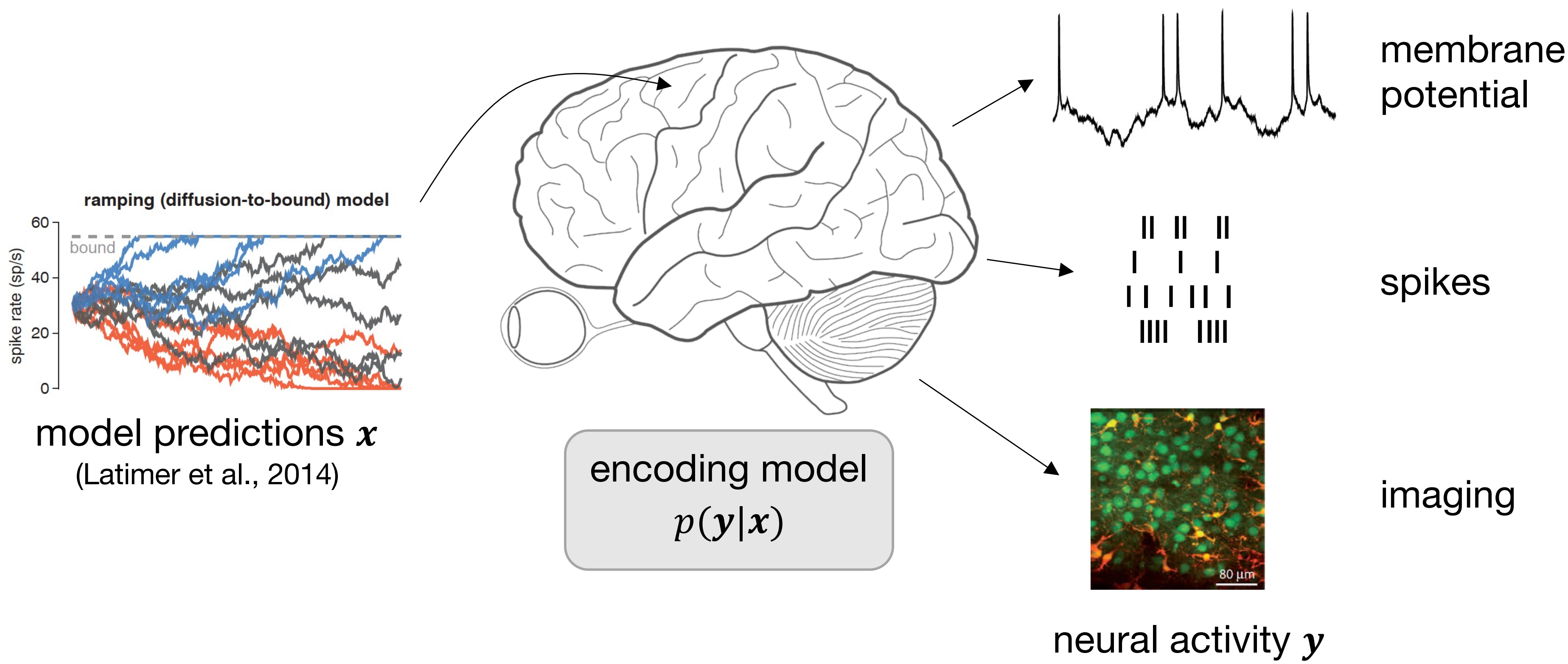
imaging

neural activity $\boldsymbol{y}$

not restricted to sensory variables

(after Jonathan Pillow)

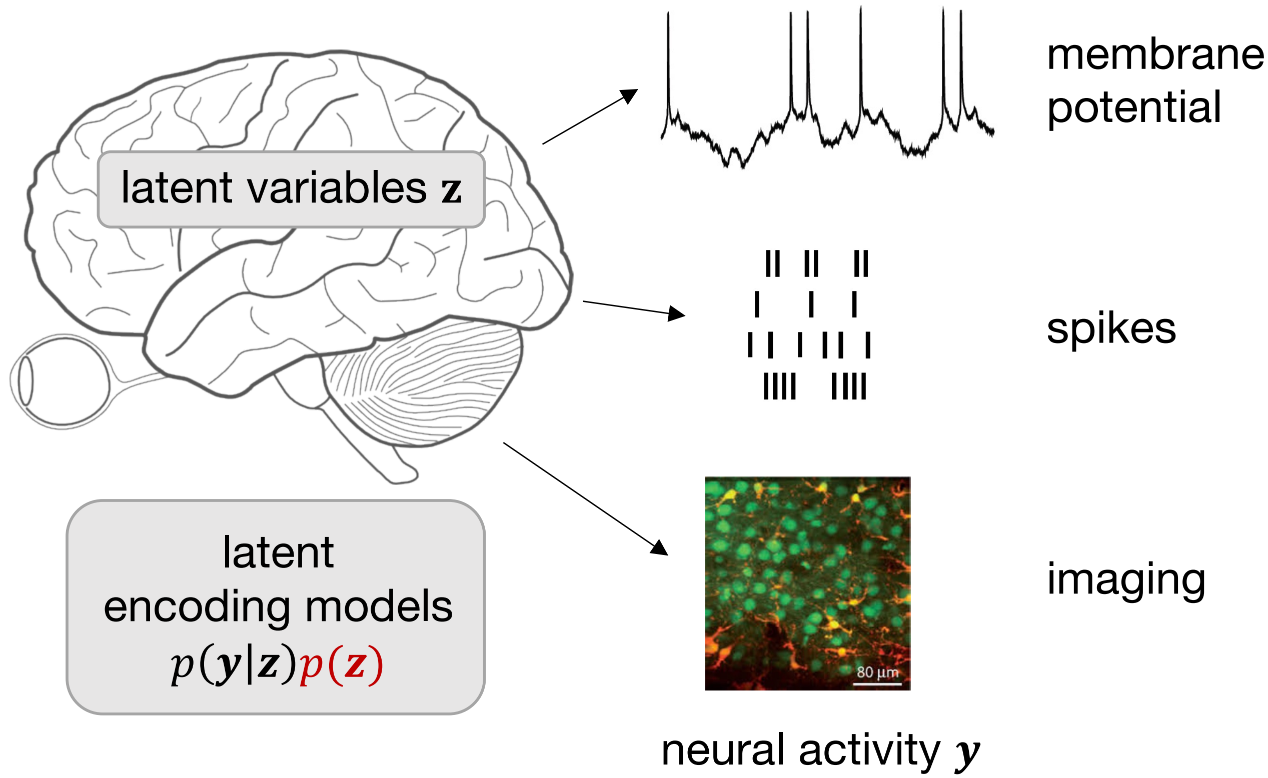
Why build models?



not restricted to sensory variables

(after Jonathan Pillow)

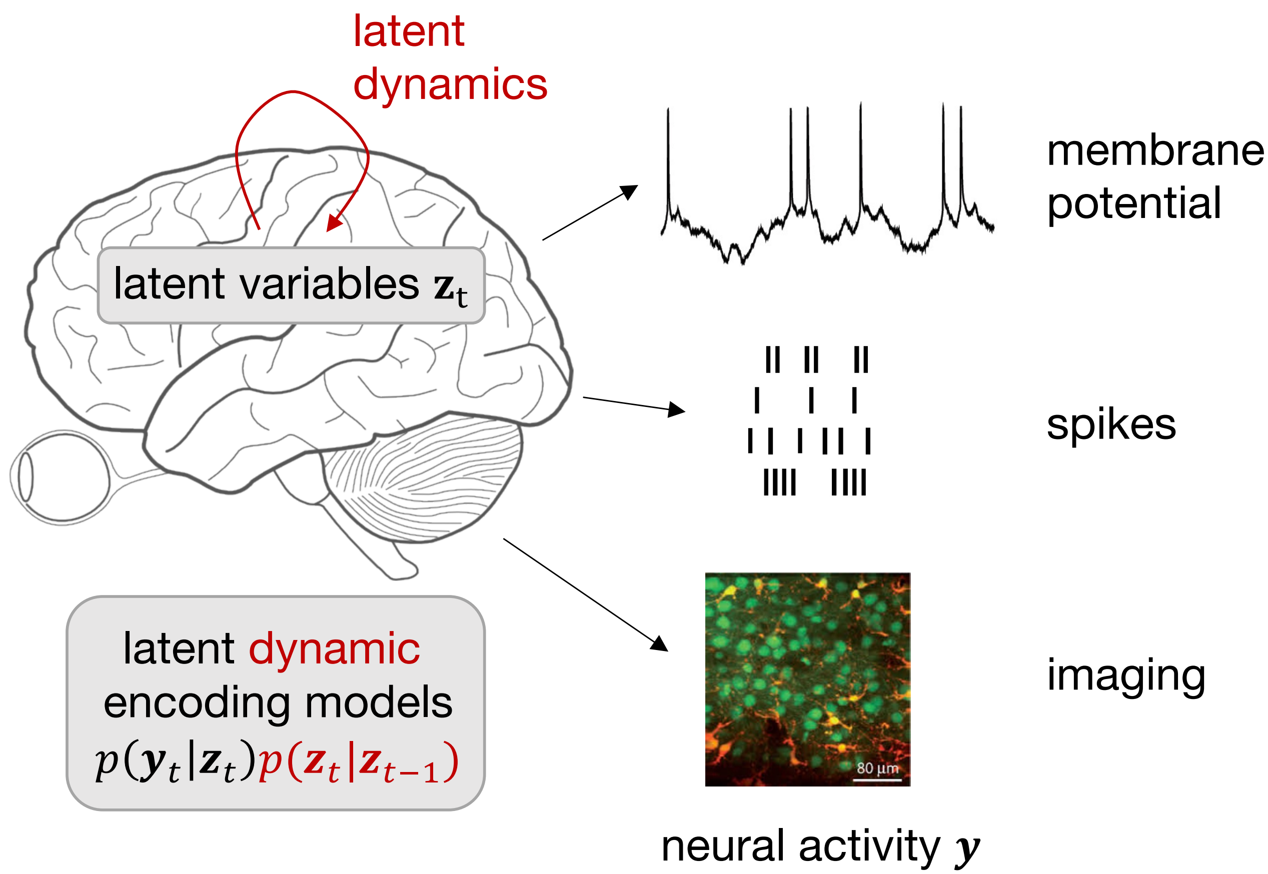
Why build models?



capture hidden structure underlying neural activity

(after Jonathan Pillow)

Why build models?



capture hidden dynamics underlying neural activity

The same can be done for behavior (but not in this course)

(after Jonathan Pillow)

Model desiderata

sweet spot

linear,
Gaussian

generalized
linear models

neural state
space models

multi-compartment,
Hodgkin-Huxley

fitability/
tractability

(can be fit to data)

richness/
flexibility

(capture realistic neural properties)

(after Jonathan Pillow)

Descriptive statistical models

normative theories
(e.g., efficient coding)

"Why does the code
take this form?"

descriptive statistical models

$p(\boldsymbol{y}|\boldsymbol{x})$

"What is the code?"

anatomy, biophysics

"How is it implemented?"

(after Jonathan Pillow)

Sessions

Session 1 (today): Bayesian recap

Exercise: Bayesian histogram tuning curve fits

Session 2: linear models
Topics: linear-Gaussian models, priors as regularizers

Session 3: generalized linear models #1
Topics: LNP neurons, single-neuron GLMs, IF neurons

Session 4: generalized linear models #2
Topics: GLMs for neural populations, decoding with GLMs

Exercise: GLMs

encoding & decoding

$p(\boldsymbol{y}|\boldsymbol{x})$
&
$p(\boldsymbol{x}|\boldsymbol{y})$

Session 5: Dimensionality reduction
Topics: PCA, probabilistic PCA & Factor Analysis, TCA

latent encoding
$p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$

Session 6: State space models #1
Topics: Laplace approx., Expectation Maximization, Variational Bayes

Session 7: State space models #2
Topics: Gaussian processes

Exercise: GPFA? (TBD)

Session 8: State space models #3
Topics: Artificial neural networks

latent dynamic encoding

$p(\boldsymbol{x}_t|\boldsymbol{z}_t)p(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})$

Session 9: Paper discussion & wrap-up

Session 1: Bayesian recap

Overview

Probabilities and probabilistic models

- Simple stimulus → response models
- Rules of probabilities
- Parametric models and their graphical representation
- Independent and identically distributed data

Inference with probabilistic models

- Maximum likelihood estimates
- Bayesian inference and its components, generative model inversion
- Maximum a-posteriori estimates
- Conjugacy and tractability

Model comparison

- Bayesian decision theory
- Posterior predictive checks
- Bayesian model comparison

Overview

Probabilities and probabilistic models

- Simple stimulus → response models
- Rules of probabilities
- Parametric models and their graphical representation
- Independent and identically distributed data

Inference with probabilistic models

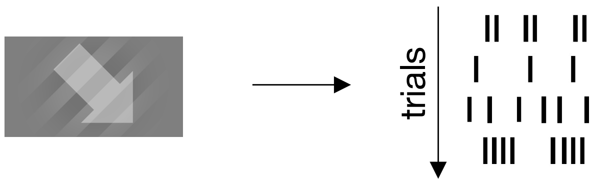
- Maximum likelihood estimates
- Bayesian inference and its components, generative model inversion
- Maximum a-posteriori estimates
- Conjugacy and tractability

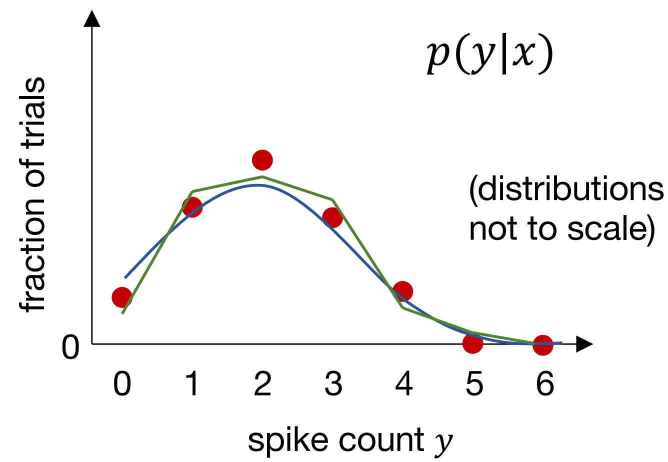
Model comparison

- Bayesian decision theory
- Posterior predictive checks
- Bayesian model comparison

Simple stimulus → response models

stimulus x response y





Directly measure $p(y|x)$?

Either x or y might be too large/continuous

Cannot extrapolate beyond seen data

Instead use (parametric) models, for example

	$p(y\|x)$	parameters
Poisson	$p(y\|x) = \mathrm{Pois}(y\|\lambda(x))$	rate $\lambda(x)$
Gaussian	$p(y\|x) = \mathrm{N}(y\|\mu(x), \sigma^2(x))$	mean $\mu(x)$, variance $\sigma^2(x)$

Fundamentals of probabilities

Probability distributions are *functions* that return probabilities

$p(X=x)$ (short: $p(x)$) returns probability that random variable X takes value $X=x$

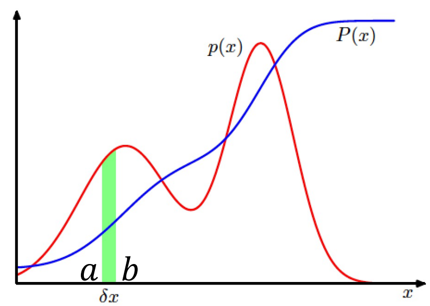
Probability *mass* differs from probability *density*

Discrete x (e.g., spike count, $x \in \{0,1,2,\dots\}$)

Continuous x (e.g., $\Delta F/F$, $x \in [0,\infty]$)

$p(x)$ returns *probability mass* ($p(x) \in [0,1]$)

$p(x)$ returns *probability density* ($p(x) \in [0,\infty]$)



(Bishop, 2006)

mass and density are related

$$\underbrace{p(a \le x \le b)}_{\text{mass}} = \int_a^b \underbrace{p(x)}_{\text{density}} \mathrm{d}x$$

Probabilities sum to one

Discrete x: $\sum_x p(x) = 1$

Continuous x: $\int p(x)\mathrm{d}x = 1$

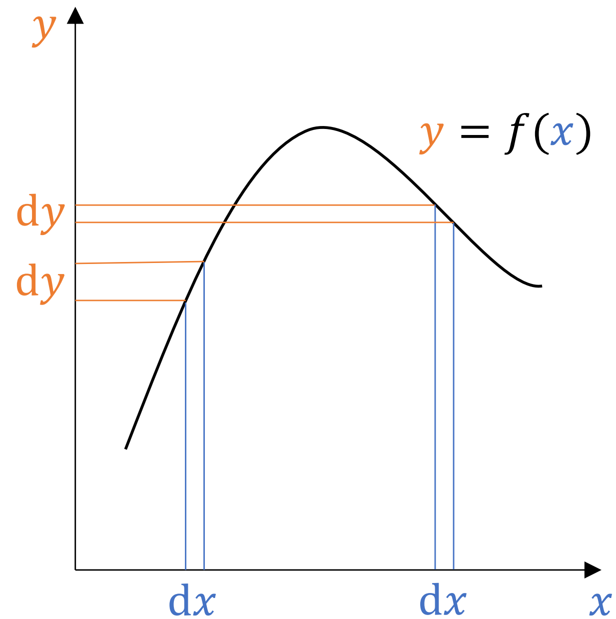
Probabilities can be defined across multiple random variables

$p(X=x, Y=y)$ (short: $p(x,y)$) returns *joint probability* that $X=x$ and $Y=y$

Transformations of random variables

We know $p_x(x)$ and $y = f(x)$

What is $p_y(y)$?



Matching probability mass $\quad p_x(x)\mathrm{d}x = p_y(y)\mathrm{d}y \quad$ s.t. $\quad p_y(y) = p_x(x)\dfrac{\mathrm{d}x}{\mathrm{d}y}$

…but ignore the sign of the derivative $\quad p_x(x) = p_y(y)\left|\dfrac{\mathrm{d}y}{\mathrm{d}x}\right|$

$$p_y(y) = p_x(X = f^{-1}(y))\left|\frac{\mathrm{d}x}{\mathrm{d}y}\right| = p_x(X = f^{-1}(y))\left|\frac{1}{f'(x)}\right|$$

For vector-valued x and y: $\left|\frac{\mathrm{d}x}{\mathrm{d}y}\right|$ becomes determinant of Jacobian

Rules of probabilities

Sum rule (also called *marginalization*) $p(y) = \sum_x p(x, y)$ (still a function!)

Product rule $p(x, y) = \underbrace{p(y|x)}p(x)$

conditional probability that $Y = y$ given that $X = x$

$$p(y|x) = \frac{p(x, y)}{p(x)}$$

Independence $p(x, y) = p(x)p(y)$

(what does sum/product rule simplify to?)

Preview: results in Bayes rule

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\sum_x p(y|x)p(x)}$$

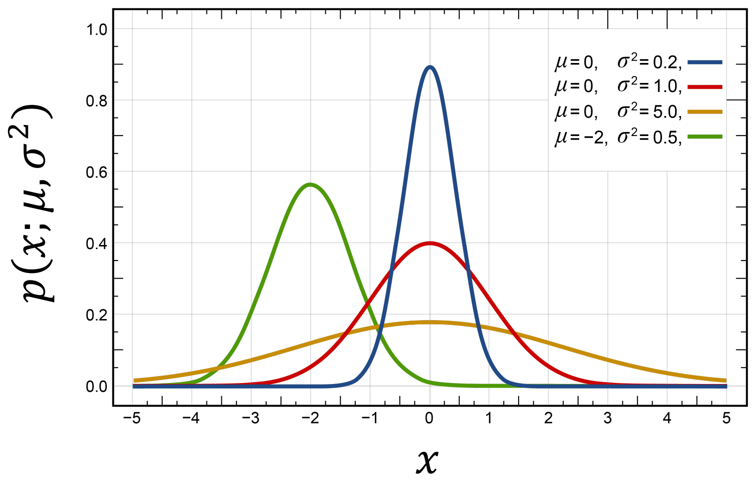
Parametric distributions

Gaussian distribution

$$p(x;\mu,\sigma^2) = \mathrm{N}(x|\mu,\sigma^2) = \overbrace{\frac{1}{\sqrt{2\pi\sigma^2}}\underbrace{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}_{x\text{ –dependent}}}^{\text{probability density function (pdf)}}$$

mean $\quad \mathrm{E}[x] = \mu$

variance $\quad \mathrm{var}[x] = \sigma^2$

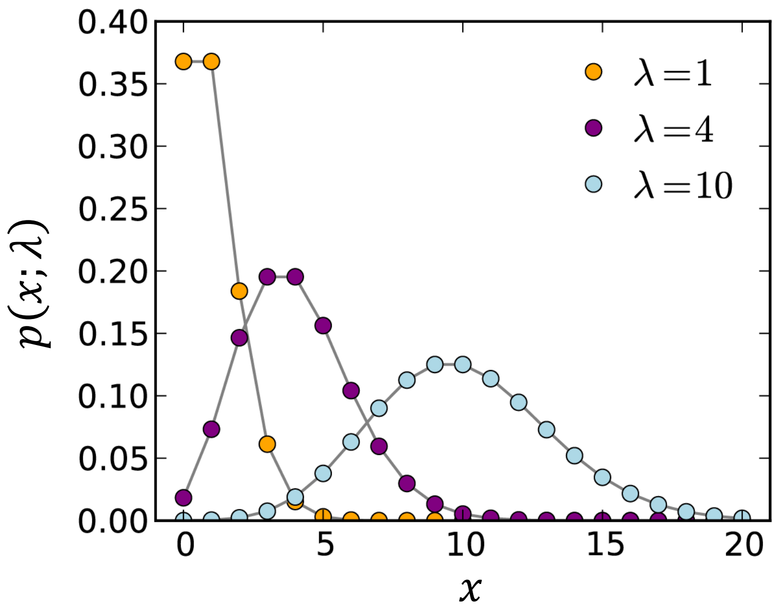


Poisson distribution

$$p(x;\lambda) = \mathrm{Pois}(x|\lambda) = \overbrace{e^{-\lambda}\underbrace{\frac{\lambda^x}{x!}}_{x\text{ –dependent}}}^{\text{probability mass function (pmf)}}$$

mean $\quad \mathrm{E}[x] = \lambda$

variance $\quad \mathrm{var}[x] = \lambda$



i.i.d. data & (directed) graphical models

Independent and _identically_ distributed (i.i.d.) data

e.g., data $y_{1:N} = y_1, y_2, \ldots, y_N$ from set of trials with same stimulus x

independent, conditional on stimulus x

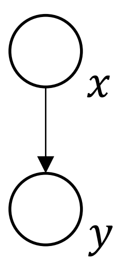
$$p(y_{1:N}|x) = \prod_{n=1}^{N} p(y_n|x) = \prod_{n=1}^{N} \mathrm{Pois}(y_n|\lambda(x))$$

assume *identical* Poisson "emissions"

(directed) graphical models

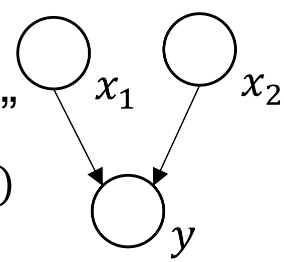
"x generates y"

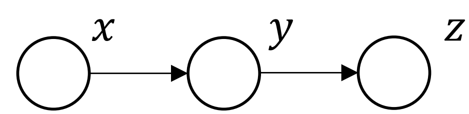
$p(y|x)$



"x_1 and x_2 generate y"

$p(y|x_1, x_2)$

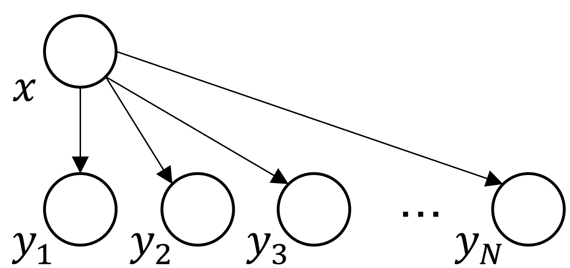




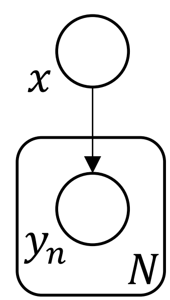
"x generates y, which generates z"

$p(z|x, y) = p(z|y)$

"x generates each y_n independently"



=



"plate"

Overview

Probabilities and probabilistic models

- Simple stimulus → response models
- Rules of probabilities
- Parametric models and their graphical representation
- Independent and identically distributed data

Inference with probabilistic models

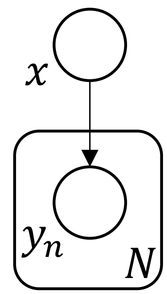
- Maximum likelihood estimates
- Bayesian inference and its components, generative model inversion
- Maximum a-posteriori estimates
- Conjugacy and tractability

Model comparison

- Bayesian decision theory
- Posterior predictive checks
- Bayesian model comparison

maximum likelihood estimates

Generative model of data $y_{1:N}$ in response to (fixed) stimulus x



$$p(y_{1:N}|x) = \prod_{n=1}^{N} p(y_n|x) = \prod_{n=1}^{N} \mathrm{Pois}(y_n|\lambda(x))$$

Assume: single rate λ for fixed stimulus x

Maximum likelihood: what is the rate λ that makes the observed data most likely?

(Which model parameters make the observed data most likely?)

$$\hat{\lambda}_{\mathrm{ML}} = \mathrm{argmax}_\lambda p(y_{1:N}|\lambda) = \mathrm{argmax}_\lambda \log p(y_{1:N}|\lambda) = \mathrm{argmax}_\lambda \sum_{n=1}^{N} \log \mathrm{Pois}(y_n|\lambda)$$

$$\ldots$$

$$= \frac{1}{N}\sum_{n=1}^{N} y_n$$

the average spike count!

Pros Consistent (converges to true λ)
Efficient (asymptotically no better estimator)
…

Cons Noisy for little data
No estimate of uncertainty
…

Proportionality, $\propto$

$$\underbrace{p(x)}^{\text{normalized}} \underbrace{\propto_x}_{\text{“proportional in } x \text{ to”}} \overbrace{f(x)}^{\text{unnormalized}} \longleftrightarrow p(x) = \frac{1}{\underbrace{Z_p}_{\text{independent of } x}} f(x)$$

This works because probability distributions sum/integrate to one!

$$\int p(x) \mathrm{d}x = 1 \qquad \frac{1}{Z_p} \int f(x) \mathrm{d}x = 1 \qquad Z_p = \int f(x) \mathrm{d}x \qquad p(x) = \frac{1}{\int f(x) \mathrm{d}x} f(x)$$

Examples

$$\mathrm{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \propto_x e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad \mathrm{Pois}(x; \lambda) = e^{-\lambda} \frac{\lambda^x}{x!} \propto_\lambda \lambda^x e^{-\lambda}$$

ML inference

$$\mathrm{argmax}_\lambda \sum_{n=1}^{N} \log \mathrm{Pois}(y_n | \lambda) = \mathrm{argmax}_\lambda \sum_{n=1}^{N} \left(\log(\lambda^{y_n} e^{-\lambda}) + \log \frac{1}{y_n!} \right)$$

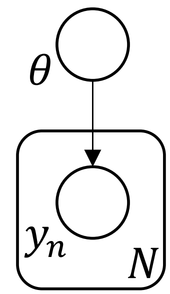
Bayesian inference

Most likely model parameters → *p*(model parameters | data)

Assume (again): single rate λ (model parameter $\theta = \lambda$) for fixed stimulus x

$$\underbrace{p(\theta|y_{1:N})}_{\text{posterior}} = \frac{\overbrace{p(y_{1:N}|\theta)}^{\text{likelihood}}\overbrace{p(\theta)}^{\text{prior}}}{\underbrace{p(y_{1:N})}_{\text{marginal likelihood}}} \propto_\theta p(y_{1:N}|\theta)p(\theta)$$

generative model $p(y_{1:N}|\theta)$ ↓



↑ generative model inversion $p(\theta|y_{1:N})$

prior	$p(\theta)$	belief about model parameter value(s) before observing data
likelihood	$p(y_{1:N}|\theta)$	likelihood of data given model parameter value(s) (function of θ)
posterior	$p(\theta|y_{1:N})$	belief about model parameter value(s) after observing data
marginal likelihood	$p(y_{1:N}) = \int p(y_{1:N}|\theta)p(\theta)\mathrm{d}\theta$	likelihood of data under model a.k.a. *model evidence*

Beliefs vs. probabilities

Probability	relative frequency of x across "trials"
Belief	belief that x is true value within "trial"

Maximum a-posteriori inference

Including prior information → *regularizes* the estimate

$$p(\theta|y_{1:N}) \propto_\theta p(y_{1:N}|\theta)p(\theta) \qquad \log p(\theta|y_{1:N}) = \log p(y_{1:N}|\theta) + \log p(\theta) + \text{const.}$$

MAP estimate: $\hat{\theta}_{\text{MAP}} = \text{argmax}_\theta(\log p(y_{1:N}|\theta) + \log p(\theta))$

Compare to ML estimate: $\hat{\theta}_{\text{ML}} = \text{argmax}_\theta \log \mathrm{p}(\mathrm{y}_{1:N}|\theta)$ (assumes $p(\theta) \propto 1$)

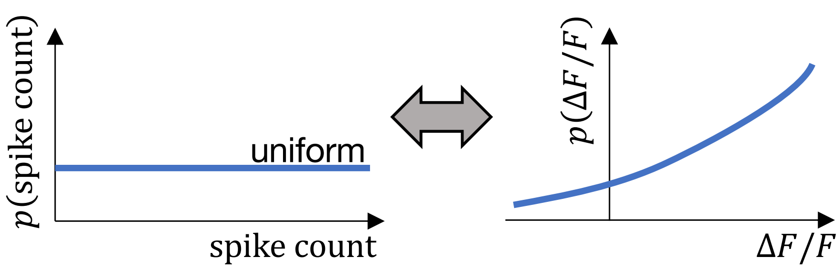
Benefits of MAP estimates

Little/uninformative data → minimize impact of noise

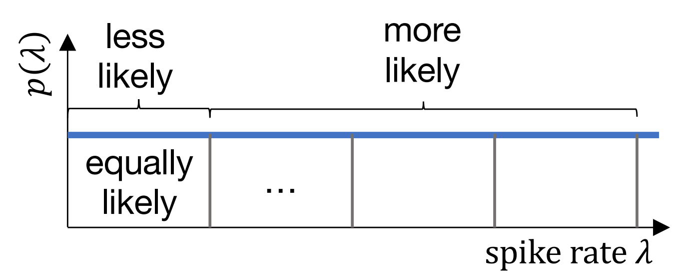
Includes prior information

The myth of "uninformative" priors (i.e., ML estimates also make assumptions)

e.g., $\Delta F/F = \log(\text{spike count})$



uniform priors, e.g. on spike rate λ?



Full posteriors & conjugate priors

Sequential inference with i.i.d. data

$$p(\theta|y_1, y_2) \propto p(y_1, y_2|\theta)p(\theta) = p(y_2|\theta)\underbrace{p(y_1|\theta)p(\theta)}_{\propto\, p(\theta|y_1)}$$

Split into two steps

$$p(\theta|y_1) \propto p(y_1|\theta)p(\theta)$$
$$p(\theta|y_1, y_2) \propto p(y_2|\theta)p(\theta|y_1)$$

Challenge: "prior" $p(\theta|y_{1:n})$ and "posterior" $p(\theta|y_{1:n+1})$ should have same distribution

Solution: use priors that are *conjugate* to likelihood

Examples	likelihood	parameter(s)	conjugate prior
	$\mathrm{N}(y_n|\mu, \sigma^2)$	μ	Gaussian
	$\mathrm{N}(y_n|\mu, \sigma^2)$	μ, σ^2	Normal-inverse-gamma (NIG)
	$\mathrm{Pois}(y_n|\lambda)$	λ	Gamma

(see "Conjugate prior" on Wikipedia)

Pros mathematical tractability
interpretable parameters

Cons inflexible
reflect undesired assumptions

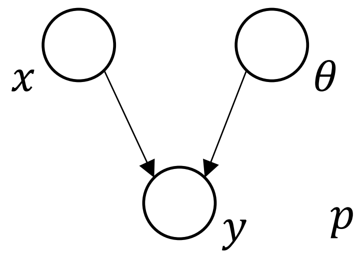
Inference summary

Generative model

models data generation, specifies assumptions

$p(y|x,\theta)$

$p(\theta)$



Inference

model inversion, identify best parameters

$p(\theta|x,y)$

Decoding

stimulus given response

$p(x|y,\hat{\theta})$

$p(x|y)$

Methods of inference

Full Bayesian	find posterior, (often) requires approximations, or conjugacy
MAP estimates	find most likely parameter posterior, $\hat{\theta}_{\mathrm{MAP}} = \mathrm{argmax}_\theta \, p(\theta|x,y)$
ML estimates	find most likely data likelihood, $\hat{\theta} = \mathrm{argmax}_\theta \, p(y|x,\theta)$ (beware implicit prior assumptions)

Overview

Probabilities and probabilistic models

- Simple stimulus → response models
- Rules of probabilities
- Parametric models and their graphical representation
- Independent and identically distributed data

Inference with probabilistic models

- Maximum likelihood estimates
- Bayesian inference and its components, generative model inversion
- Maximum a-posteriori estimates
- Conjugacy and tractability

Model comparison

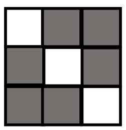
- Bayesian decision theory
- Posterior predictive checks
- Bayesian model comparison

Bayesian decision theory

Tuning posterior probability distributions into decisions/estimates

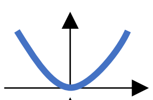
$$\underbrace{\hat{\theta}_L}_{\text{best estimate under loss } L} = \text{argmin}_{\hat{\theta}} \int \underbrace{L(\hat{\theta}, \theta)}_{\text{loss for choosing } \hat{\theta} \text{ when } \theta \text{ is correct}} p(\theta | x, y) \mathrm{d}\theta = \text{argmin}_{\hat{\theta}} \mathrm{E}\left[L(\hat{\theta}, \theta) | x, y\right]$$

Decision problems (e.g., θ is nominal, i.e., unordered and discrete)

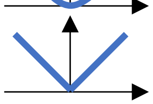


0-1 loss $\quad L(\hat{\theta}, \theta) = \begin{cases} 0, & \hat{\theta} = \theta, \\ 1, & \hat{\theta} \neq \theta. \end{cases}$ $\quad$ pick most likely θ

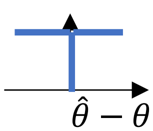
Estimation problems (e.g., θ is ordinal or continuous)



squared loss $\quad L(\hat{\theta}, \theta) = (\hat{\theta} - \theta)^2 \quad \hat{\theta}_L = \mathrm{E}[\theta | x, y]$



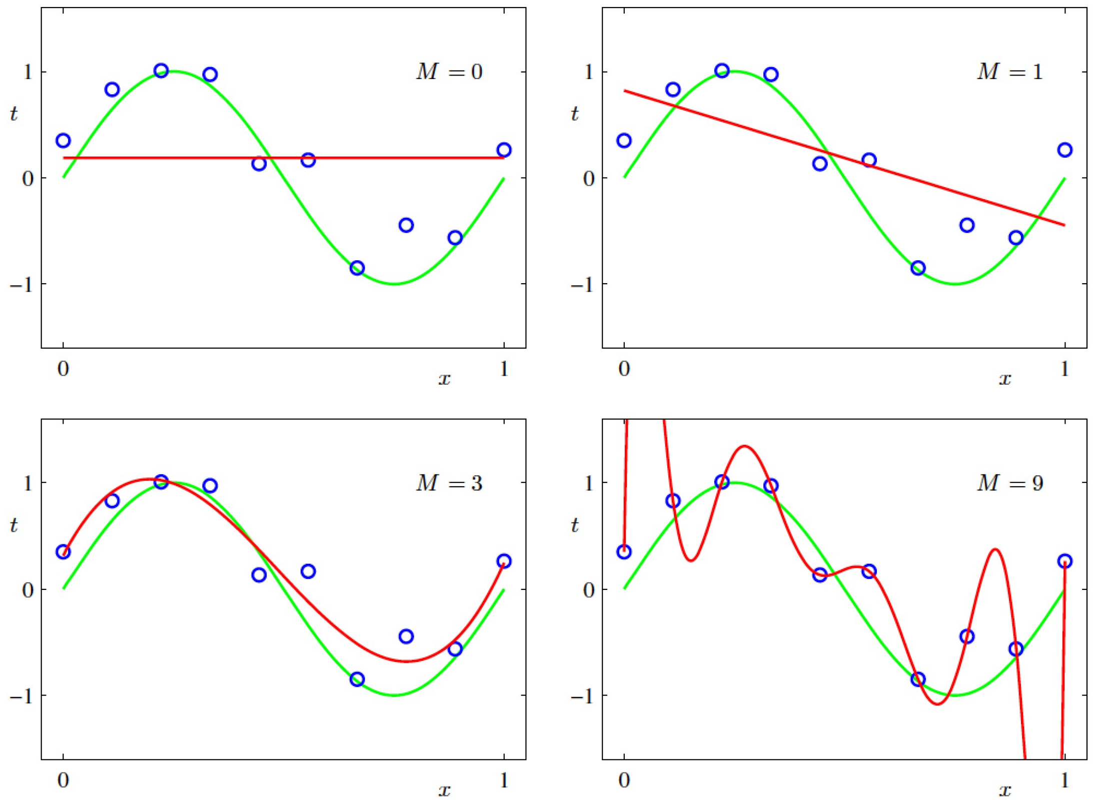
absolute loss $\quad L(\hat{\theta}, \theta) = |\hat{\theta} - \theta| \quad \hat{\theta}_L = \text{median of } \mathrm{p}(\theta | x, \mathrm{y})$

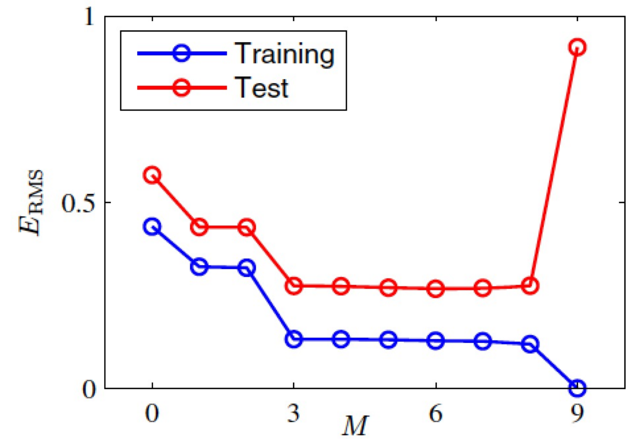


notch loss $\quad L(\hat{\theta}, \theta) = \lim_{c \to 0} \begin{cases} 0, & |\hat{\theta} - \theta| < c, \\ 1, & \text{otherwise.} \end{cases} \quad \hat{\theta}_L = \hat{\theta}_{\mathrm{MAP}} = \text{mode of } p(\theta | x, y)$

Why care? The chosen estimator determines the assumed loss function

Comparing models





(Bishop, 2006)

too simple ↔ too complex

unfitting ⟷ overfitting

Posterior predictive checks assess model performance on hold-out dataset (e.g., cross-validation)

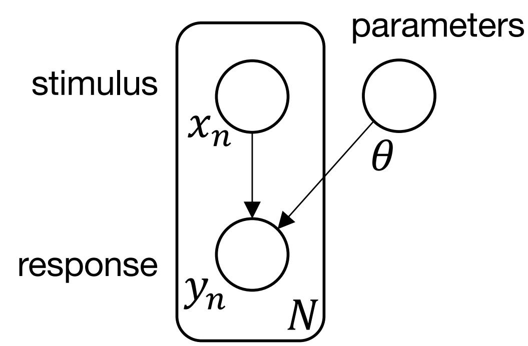
Bayesian model comparison use model evidence (marginal likelihood) to reward high data likelihood while penalizing model complexity

Posterior predictive checks

Estimate prediction quality by comparing model predictions to hold-out data

Posterior predictive distribution

Training data $x_{1:N}, y_{1:N}$; test set instance $\tilde{x}, \tilde{y}$



$$p(\tilde{y}|\tilde{x}, x_{1:N}, y_{1:N}) = \int \overbrace{p(\tilde{y}|\theta, \tilde{x})}^{\text{likelihood of test set instance}} \underbrace{p(\theta|x_{1:N}, y_{1:N})}_{\text{training set posterior}} \mathrm{d}\theta \overset{\text{ignoring posterior parameter uncertainty}}{\approx} p(\tilde{y}|\hat{\theta}, \tilde{x})$$

Assessing prediction quality (here across training instances $\tilde{x}_{1:M}, \tilde{y}_{1:M}$)

Choose loss function → measure average test set loss/error

e.g., absolute loss: $$\frac{1}{M}\sum_{m=1}^{M} |\tilde{y}_m - \underbrace{\mathrm{median}(\tilde{y}|\tilde{x}_m)}_{\text{median estimate compatible with absolute loss}}|$$

… or assess hold-out data log-likelihood $\log p(\tilde{y}_{1:M}|\tilde{x}_{1:M}) = \sum_{m=1}^{M} \log p(\tilde{y}_m|\tilde{x}_m)$

(requires comparable likelihoods across models)

Bayesian model comparison

Marginal likelihood (a.k.a. *model evidence*) captures model fit and complexity

$$\underbrace{p(y_{1:N} | x_{1:N}, M_j)}_{\text{model evidence for model } M_j} = \underbrace{\int p(y_{1:N} | x_{1:N}, \theta, M_j) \underbrace{p(\theta | M_j)}_{\text{parameter prior for model } M_j} \mathrm{d}\theta}_{\text{conjugacy might make integral tractable}}$$

Compare M_1 and M_2 by (log-)Bayes' factor

Bayes' rule,
assuming uniform model prior, $p(M_j) \propto 1$

$$\log \frac{p(M_1 | x_{1:N}, y_{1:N})}{p(M_2 | x_{1:N}, y_{1:N})} = \log \frac{p(y_{1:N} | x_{1:N}, M_1)}{p(y_{1:N} | x_{1:N}, M_2)} \overset{?}{\lesseqgtr} 0$$

See "Bayes factor" on Wikipedia
for guideline values of significant differences

Comparison to posterior predictive checks

Pros does not require hold-out data
(sometimes) computationally cheaper
(usually) more sensitive to model details

Cons spurious results for bad models
sensitive to choice of prior
(often) hard/impossible to compute

In general, posterior predictive checks are the safer choice!

Topics that we won’t discuss

Intractability

Bayesian inference is in most cases intractable, need to be approximated

Approximations: variational Bayes, Markov Chain Monte Carlo, etc.

Calibration

Bayesian inference is sensitive to model misspecification

How to ensure that posterior beliefs correspond to variability across datasets?

Combining Bayesian inference and deep learning

Deep neural networks are ‘just another function approximator’

Normative computational models

Bayesian inference as a model for how brain processes uncertain information

Generates predictions for neural dynamics through encoding/decoding models

…

Overview

Probabilities and probabilistic models

- Simple stimulus → response models
- Rules of probabilities
- Parametric models and their graphical representation
- Independent and identically distributed data

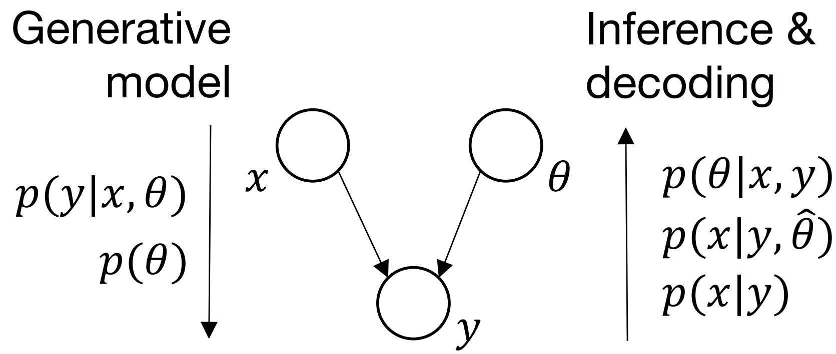
Inference with probabilistic models

- Maximum likelihood estimates
- Bayesian inference and its components, generative model inversion
- Maximum a-posteriori estimates
- Conjugacy and tractability

Model comparison

- Bayesian decision theory
- Posterior predictive checks
- Bayesian model comparison

Summary

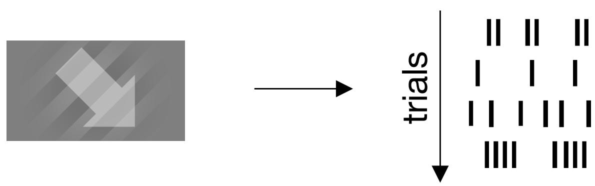


Handle uncertainty/noise	Probabilities (and associated rules)
Model structure / independence	Graphical models
Inference	Approximate (ML, MAP), or full posteriors
Assess/compare models	Posterior predictive checks & Bayesian model comp.

Exercise

Simple stimulus-response models $p(y|x,\theta)$

stimulus x response y



6 neurons
16 different drift directions x
40 trials/neuron and drift direction
bin drift directions into {1,2,4,8,16} bins
assume same response model per bin

Poisson likelihood $\mathrm{Pois}(y|\lambda(x))$, fit rate $\lambda(x)$ per stimulus bin
Gaussian likelihood $N(y|\mu(x),\sigma^2(x))$, fit mean $\mu(x)$ and variance $\sigma^2(x)$ per stimulus bin

Compare ML and MAP estimates, impact of prior

Compare models across different bin sizes, and different prior variances

Deliverable: brief write-up
See session notes for instructions

Until next week

Complete exercise and write-up

Read statistical methods sections (see notes for Session 2)

Next session

Discussing the exercise (~15min)

Theory of Gaussians & linear models (remaining time)