Gaussian processes & non-Gaussian likelihoods

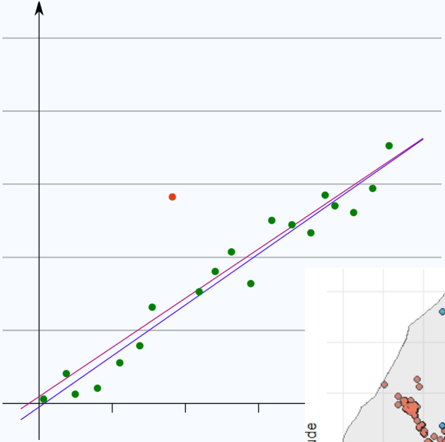
ST John

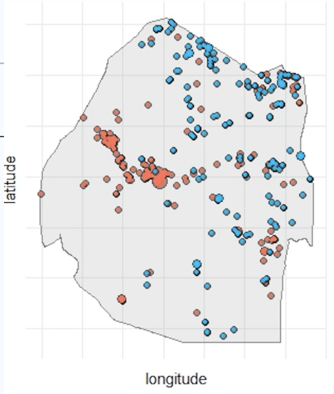
ti.john @ aalto.fi
infinitecuriosity.org

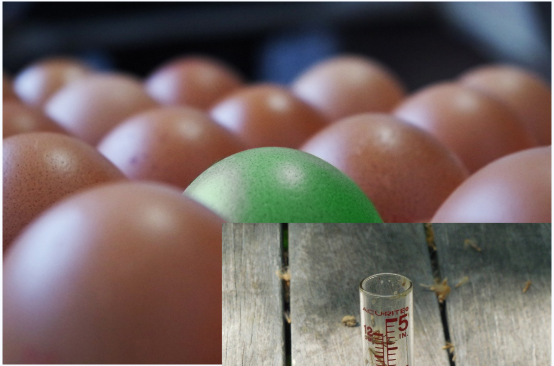
Finnish Center for Artificial Intelligence
& Aalto University

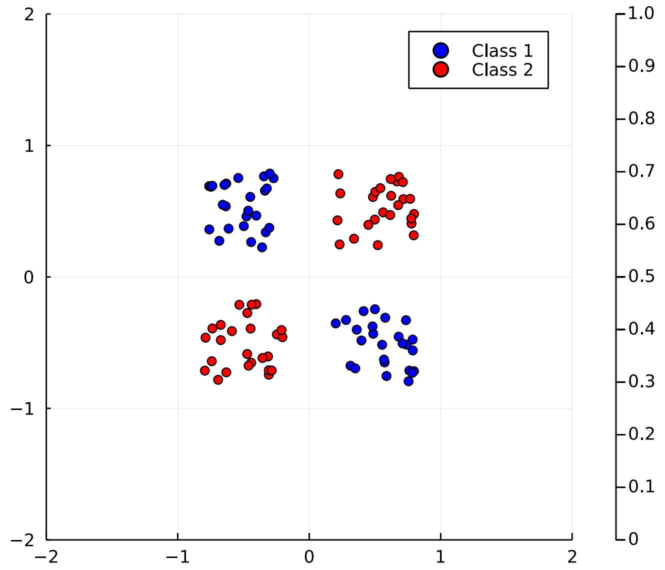
Gaussian Process Summer School 2024, 10 September 2024

Not Gaussian noise

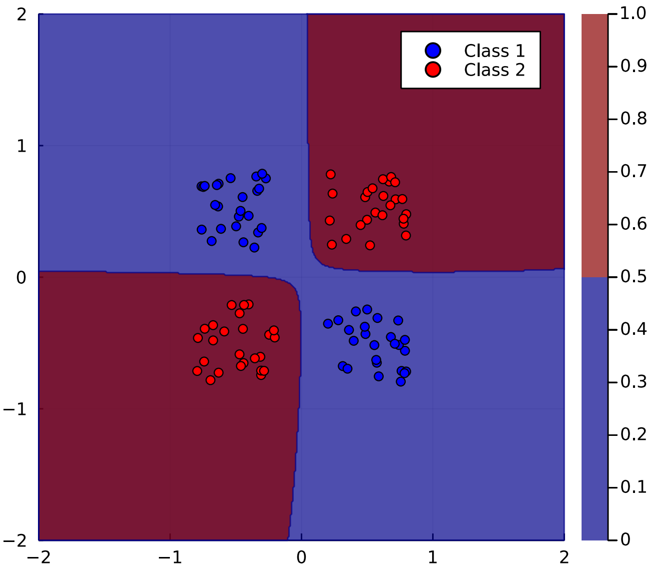




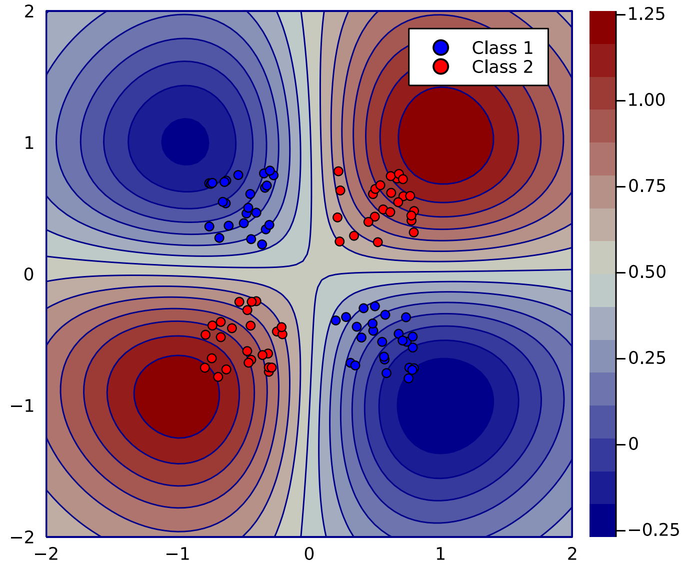




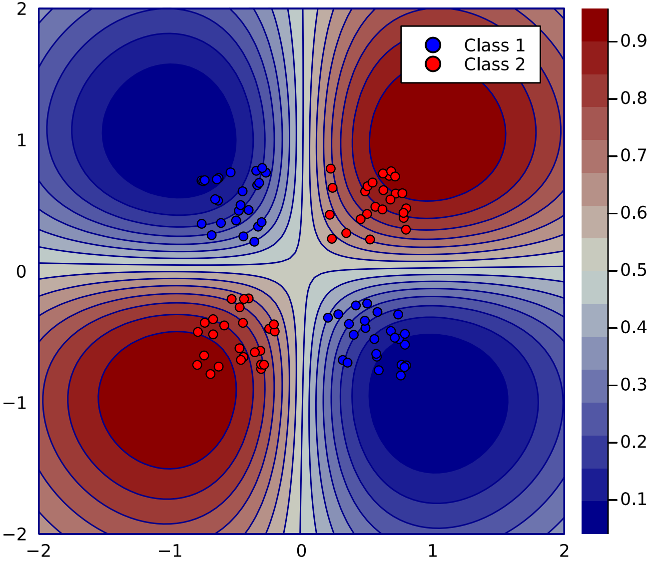
How can we model this?



SVM classification



Gaussian process regression

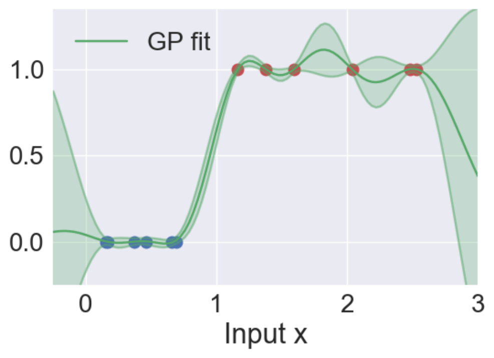


Gaussian process **classification**

Why don't we use regression models for classification?

- Binary classification: data set $\{x_n, y_n\}_{n=1}^N$ with $y_n \in \{0, 1\}$
- We want to model $p(y_n = +1 \mid x_n)$
- Why not simply use a GP regression model with labels: $y_n \in \{0, 1\}$:

$$p(y_n = +1 \mid x_n) = f(x_n)$$



Overview

Outline:

1. **Gaussian processes with Gaussian likelihood**
2. What is the likelihood? Connecting observations and Gaussian process prior
3. Non-Gaussian likelihoods: what happens to the posterior?
4. How to approximate the intractable
5. Comparison

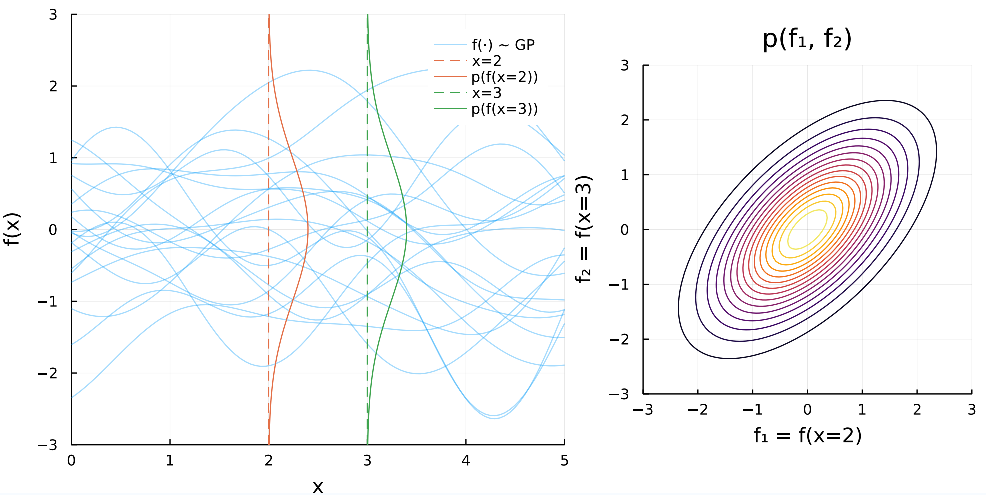
+ *Intuitive* understanding
+ Learning the language

– In-depth expertise
– Lots of maths

Setting the scene

Gaussian process $f(\cdot)$

Distribution over *functions*
Marginals are Gaussian (mean and covariance)

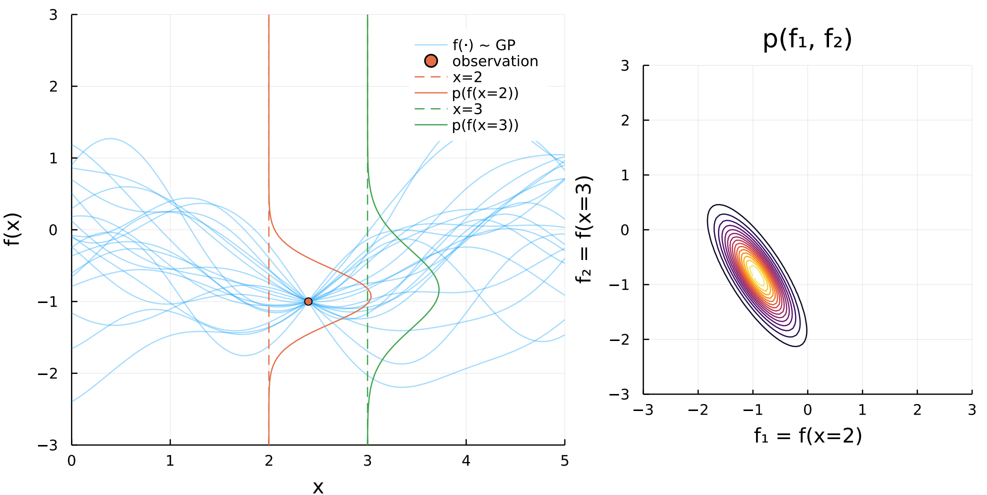


infinitecuriosity.org/vizgp

Gaussian process conditioned on observation

Distribution over *functions*

Marginals are Gaussian (mean and covariance)



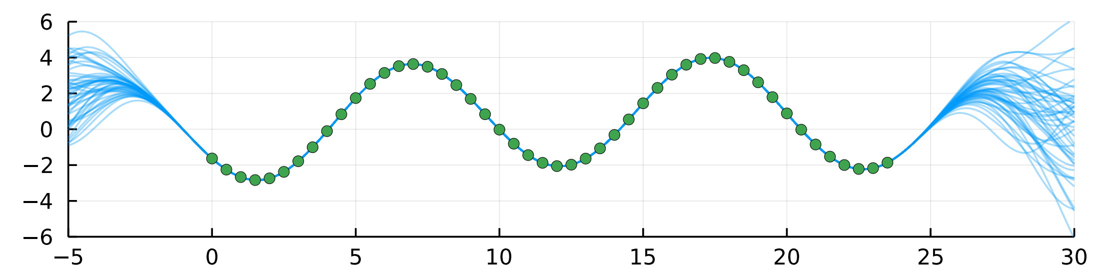
infinitecuriosity.org/vizgp

exact conditioning

Without noise model, we interpolate observations:

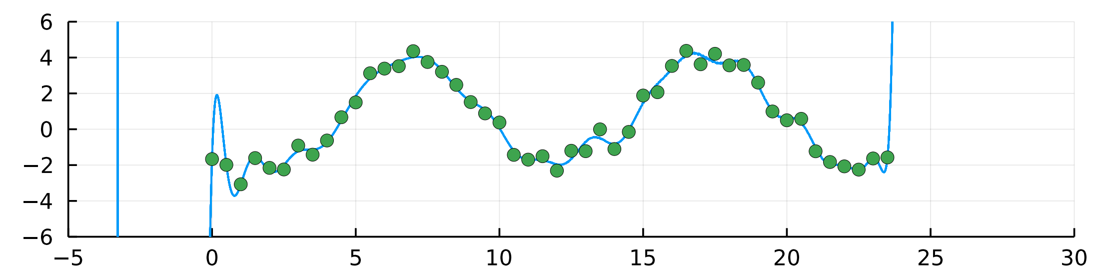
$$y(x) = f(x) + \epsilon, \qquad \epsilon \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2_{\text{noise}})$$

$$p(y \mid f) = \mathcal{N}(y \mid f, \sigma^2_{\text{noise}})$$



Without noise model, we interpolate observations:

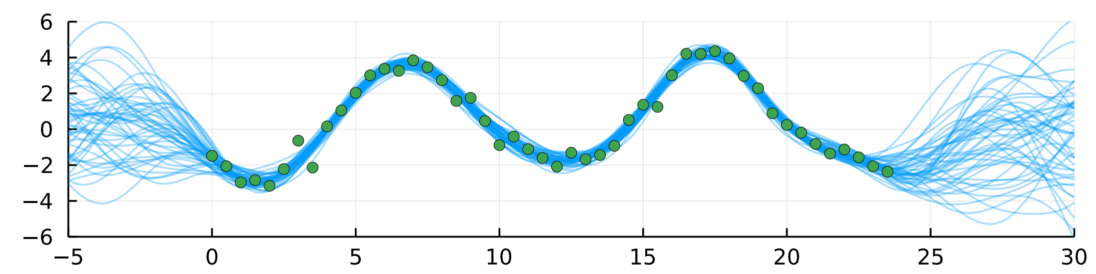
$$y(x) = f(x) + \epsilon, \qquad \epsilon \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2_{\text{noise}})$$
$$p(y \mid f) = \mathcal{N}(y \mid f, \sigma^2_{\text{noise}})$$



Gaussian noise model

Gaussian additive noise model, written two ways:

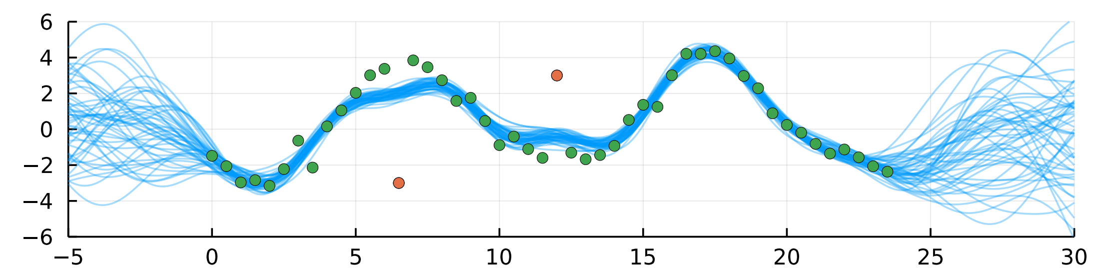
$$y(x) = f(x) + \epsilon, \qquad \epsilon \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2_{\text{noise}})$$
$$p(y \mid f) = \mathcal{N}(y \mid f, \sigma^2_{\text{noise}})$$



misspecified Gaussian noise model

Gaussian additive noise model, written two ways:

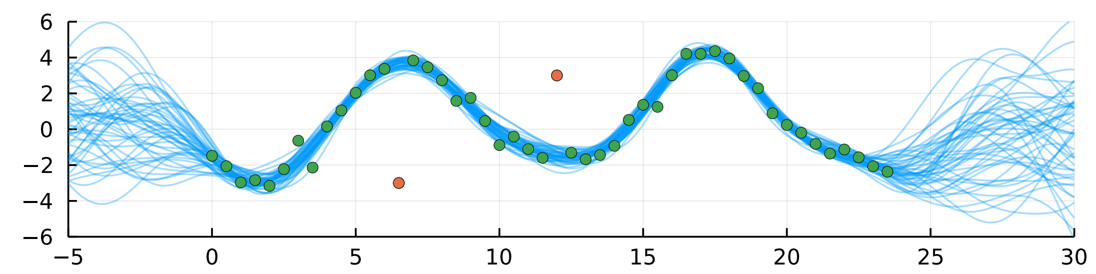
$$y(x) = f(x) + \epsilon, \qquad \epsilon \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2_{\text{noise}})$$
$$p(y \mid f) = \mathcal{N}(y \mid f, \sigma^2_{\text{noise}})$$



heavy-tailed noise model

$$y(x) = f(x) + \epsilon, \qquad \epsilon \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2_{\text{noise}})$$

$$p(y \mid f) = \mathcal{N}(y \mid f, \sigma^2_{\text{noise}})$$

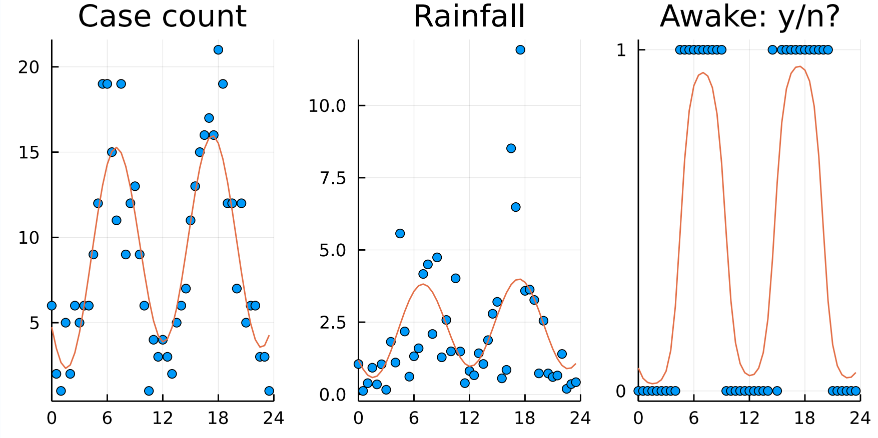


Outline

- ✓ Gaussian processes with Gaussian likelihood
2. **What is the likelihood? Connecting observations and Gaussian process prior**
3. Non-Gaussian likelihoods: what happens to the posterior?
4. How to approximate the intractable
5. Comparison

Likelihood

Non-Gaussian observations



latent functional relationship (correlations!)

$$p(y_n \mid f(x_n))$$

Likelihood

$$p(\mathbf{y} \mid \mathbf{f}) = \prod_{n=1}^{N} p(y_n \mid f_n); \qquad f_n = f(x_n)$$

factorizing

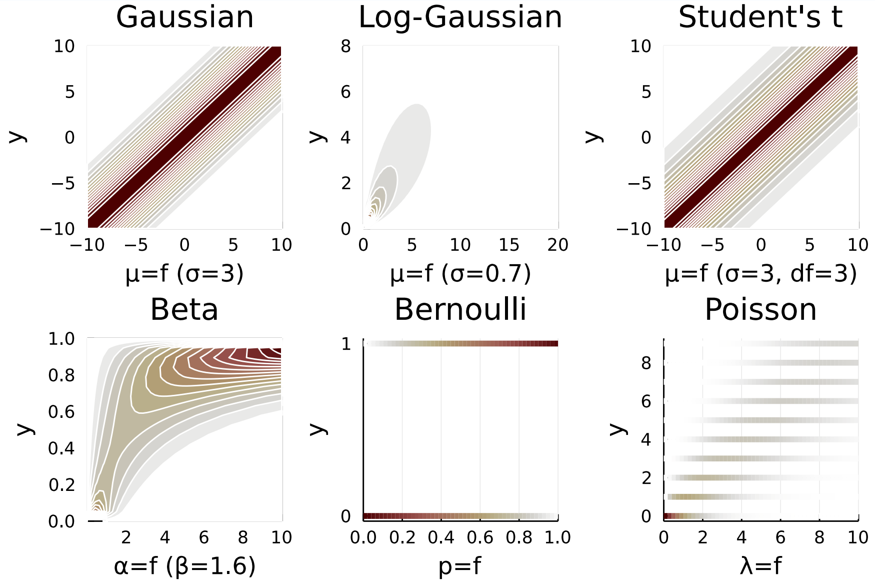
Let’s consider the individual (marginal, 1D) likelihood term:

$$p(y \mid f)$$

Function of two arguments:

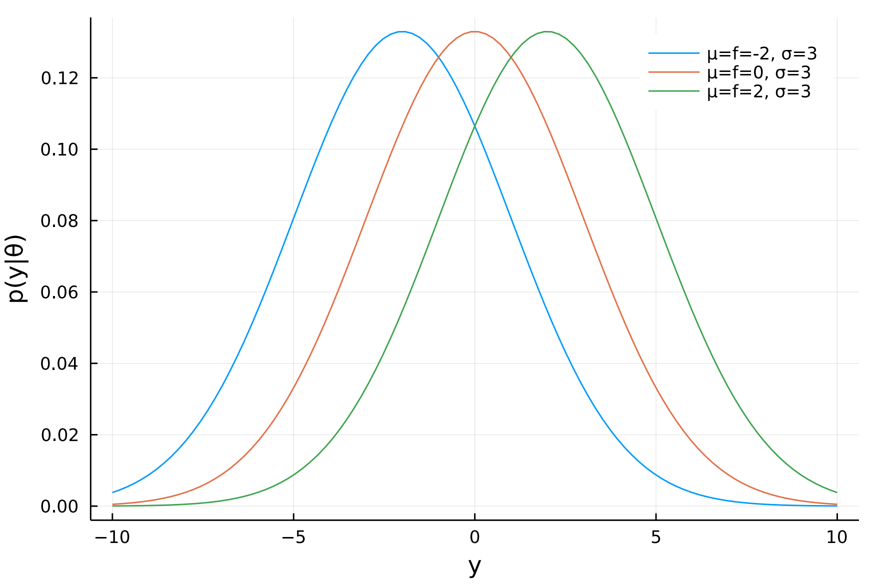
$$y \mapsto p(y \mid f), \qquad f \mapsto p(y \mid f)$$

$p(y \mid f)$

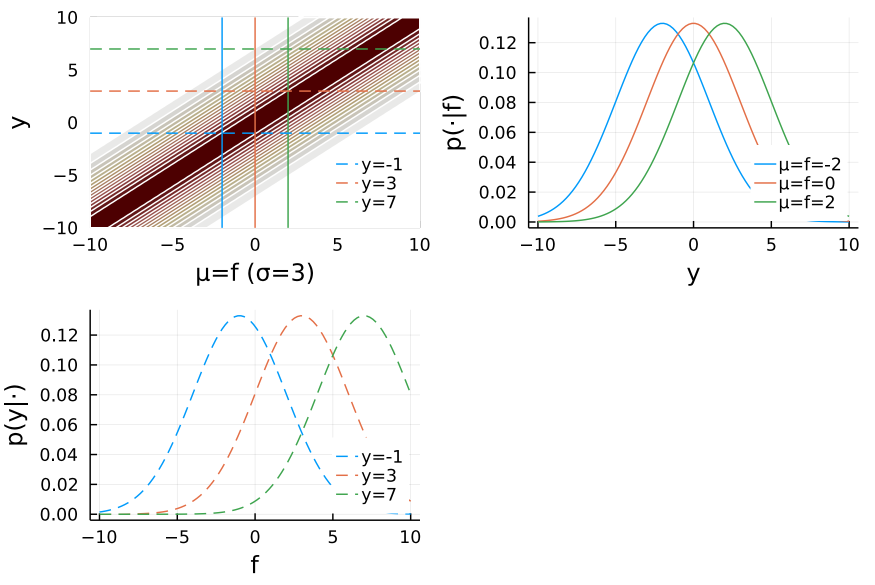


Gaussian
10
5
0
−5
−10
y
−10 −5 0 5 10
μ=f (σ=3)
Log-Gaussian
8
6
4
2
0
y
0 5 10 15 20
μ=f (σ=0.7)
Student's t
10
5
0
−5
−10
y
−10 −5 0 5 10
μ=f (σ=3, df=3)
Beta
1.0
0.8
0.6
0.4
0.2
0.0
y
2 4 6 8 10
α=f (β=1.6)
Bernoulli
1
0
y
0.0 0.2 0.4 0.6 0.8 1.0
p=f
Poisson
8
6
4
2
0
y
0 2 4 6 8 10
λ=f

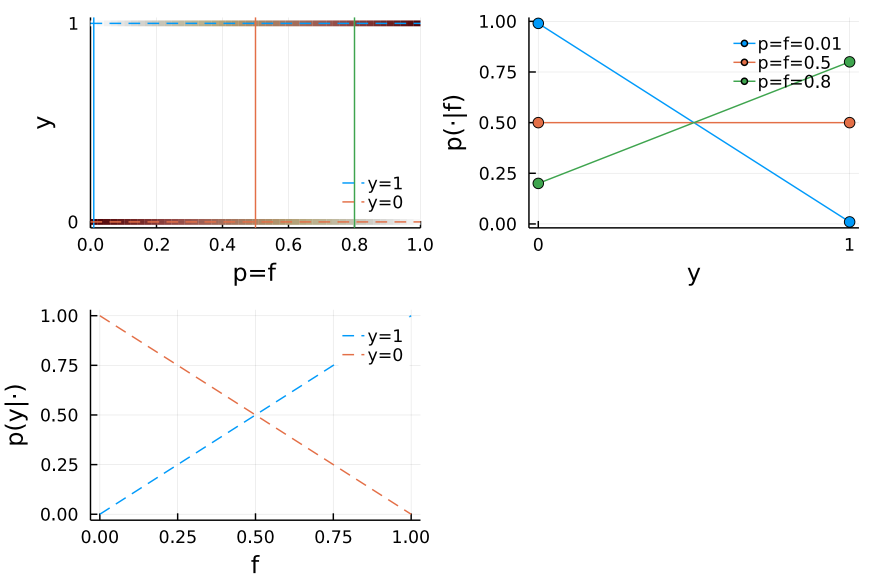
$p(y \mid f)$: Gaussian



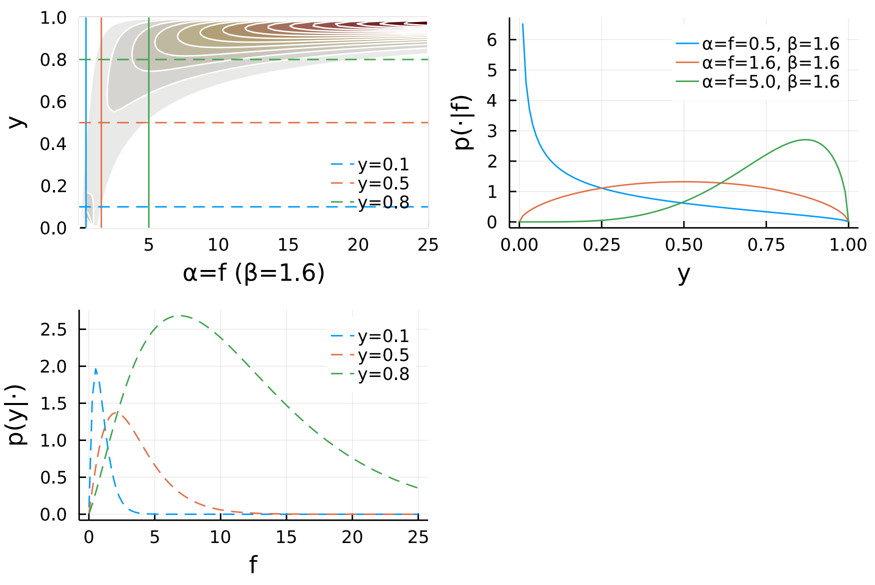
$p(y\,|\,f)$: Gaussian



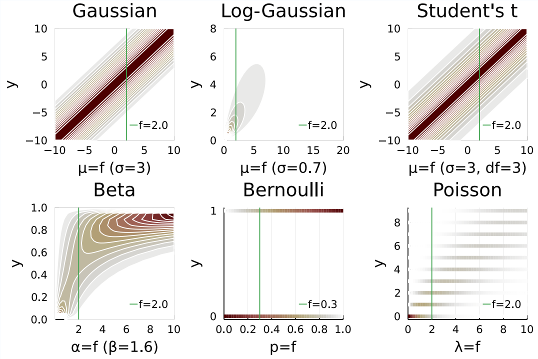
$p(y \mid f)$: Bernoulli

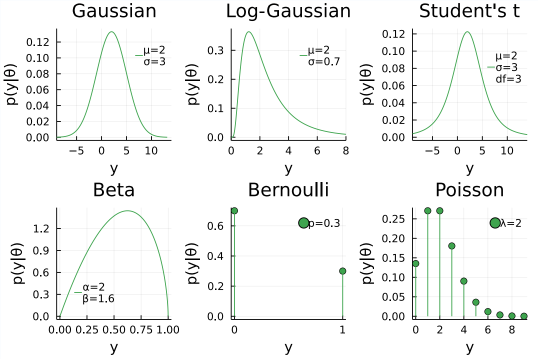


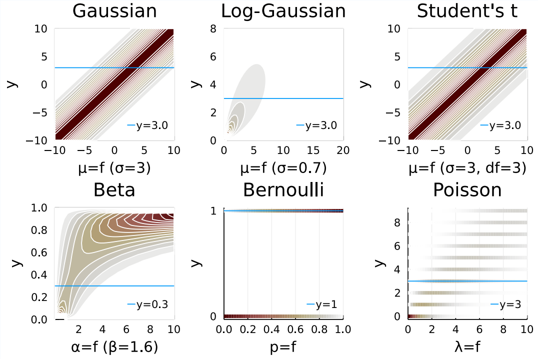
$p(y \mid f)$: Beta

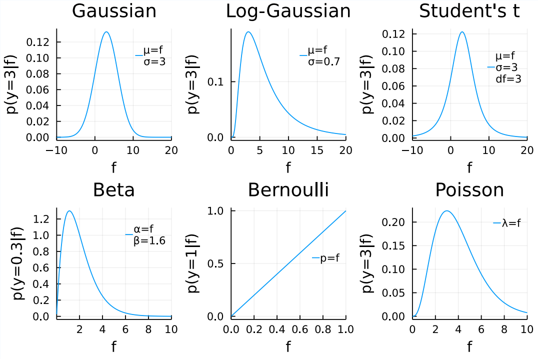


y
α=f (β=1.6)
y=0.1
y=0.5
y=0.8
p(·|f)
α=f=0.5, β=1.6
α=f=1.6, β=1.6
α=f=5.0, β=1.6
y
p(y|·)
f
y=0.1
y=0.5
y=0.8


Gaussian
Log-Gaussian
Student's t
Beta
Bernoulli
Poisson
μ=f (σ=3)
μ=f (σ=0.7)
μ=f (σ=3, df=3)
α=f (β=1.6)
p=f
λ=f
f=2.0
f=0.3
y


Gaussian
Log-Gaussian
Student's t
Beta
Bernoulli
Poisson
p(y|θ)
y
μ=2
σ=3
σ=0.7
df=3
α=2
β=1.6
p=0.3
λ=2


Gaussian
Log-Gaussian
Student's t
y=3.0
μ=f (σ=3)
μ=f (σ=0.7)
μ=f (σ=3, df=3)
Beta
Bernoulli
Poisson
y=0.3
y=1
y=3
α=f (β=1.6)
p=f
λ=f
y


Gaussian
Log-Gaussian
Student's t
μ=f σ=3
μ=f σ=0.7
μ=f σ=3 df=3
p(y=3|f)
f
Beta
Bernoulli
Poisson
α=f β=1.6
p=f
λ=f
p(y=0.3|f)
p(y=1|f)

$p(y \mid f)$: likelihood of parameter f — y fixed

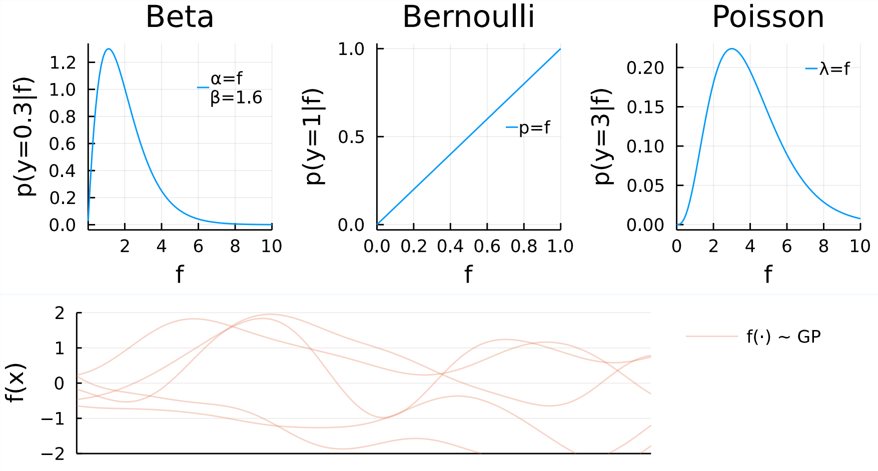
Two important aspects of likelihoods:

1. link functions
2. log-concavity

Link functions

$$\mathbb{E}[y] = \theta \in (0 \ldots \infty) \qquad \text{link}(\theta) = f$$

$$f \sim \mathcal{N} \qquad \in (-\infty \ldots \infty) \qquad \theta = \text{invlink}(f)$$



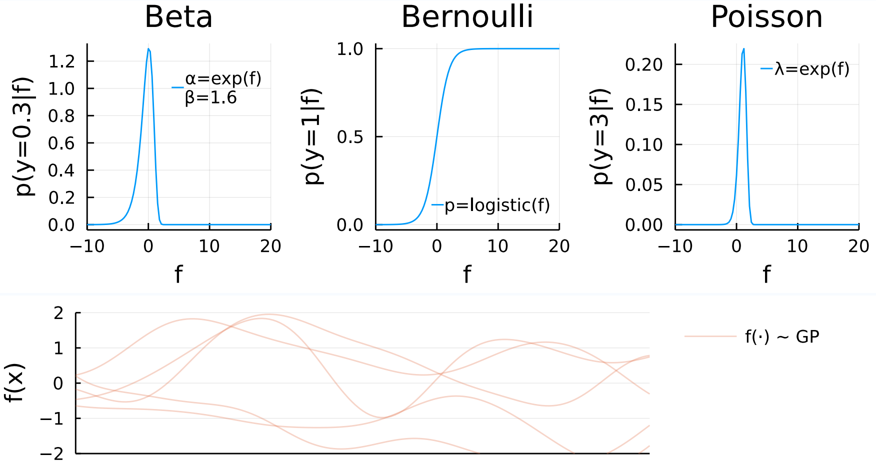
Link functions

$$\mathbb{E}[y] = \theta \in (0 \ldots \infty)$$

$$f \sim \mathcal{N} \qquad \in (-\infty \ldots \infty)$$

$$\text{link}(\theta) = f$$

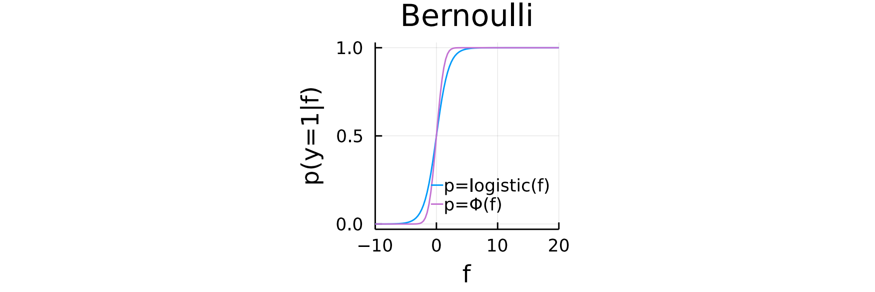
$$\theta = \text{invlink}(f)$$

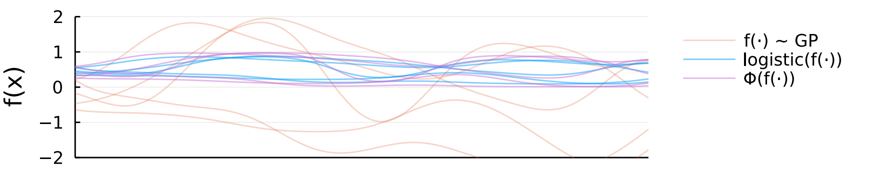


Link functions

$$\mathbb{E}[y] = \theta \in (0 \ldots \infty)$$
$$f \sim \mathcal{N} \qquad \in (-\infty \ldots \infty)$$

$$\text{link}(\theta) = f$$
$$\theta = \text{invlink}(f)$$

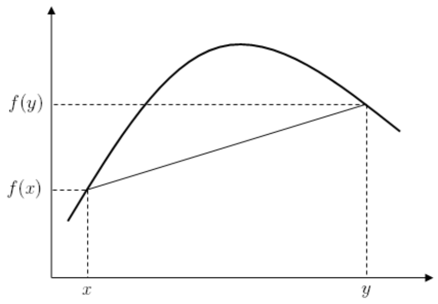




Link functions: key take-aways

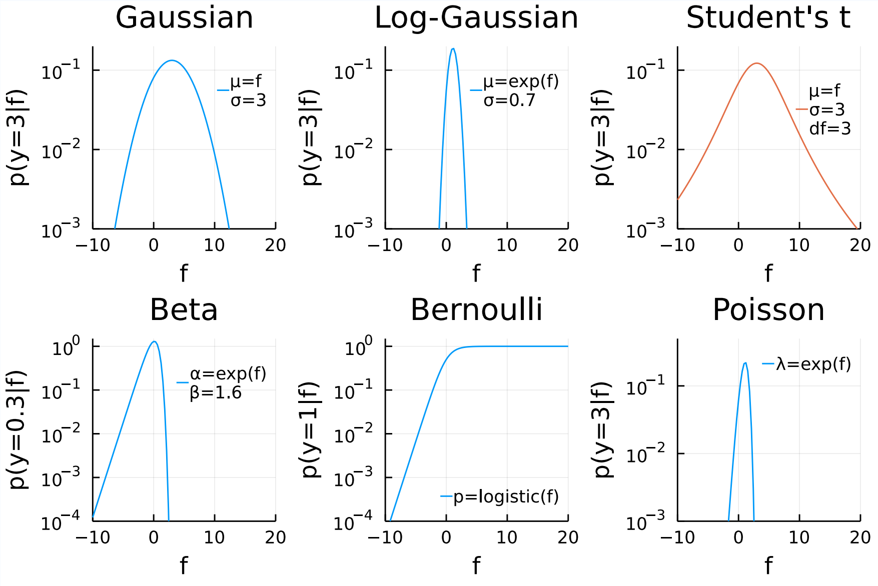
- link function is a bijector that **matches GP** (unbounded function values) **to domain of likelihood parameter** (e.g. positive rate for Poisson, Gamma)
- bijector is not unique, but a **modelling choice** (e.g. $\exp(f)$ vs. f^2 vs. $\mathrm{softplus}(f)$)
  - affects your model: **check your assumptions!**

(Log-)concavity



$$f\left(\alpha x + (1-\alpha)y\right) \geq \alpha f(x) + (1-\alpha)f(y)$$

Log-concavity of likelihoods

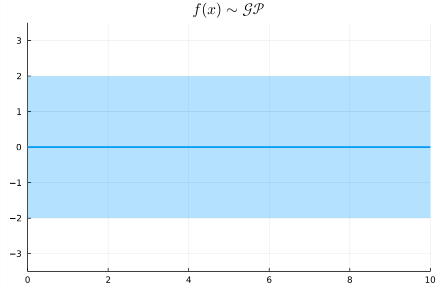


Log-concavity of likelihoods: key take-aways

- **log-concave likelihoods are "nice"**
  - related to convexity of optimization problem in approximate inference
- for non-log-concave likelihoods, **special implementations** may be needed (e.g. for Student's t likelihood)

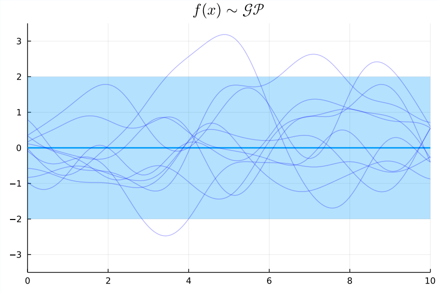
Back to GPs...

Functional prior $p(f)$



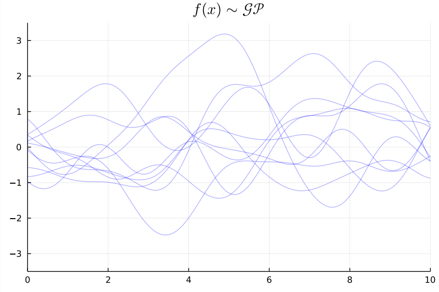
Back to GPs...

Functional prior $p(f)$



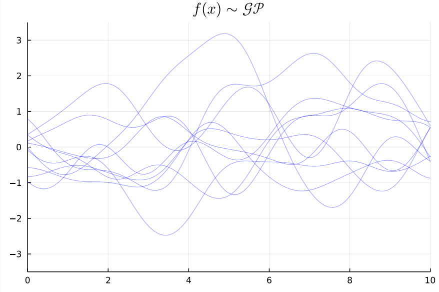
Back to GPs...

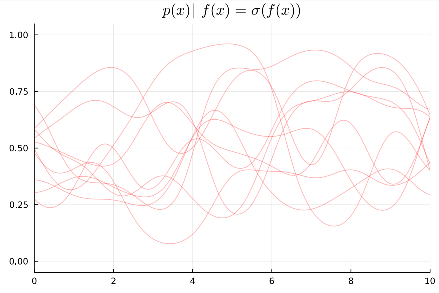
Functional prior $p(f)$



Back to GPs...

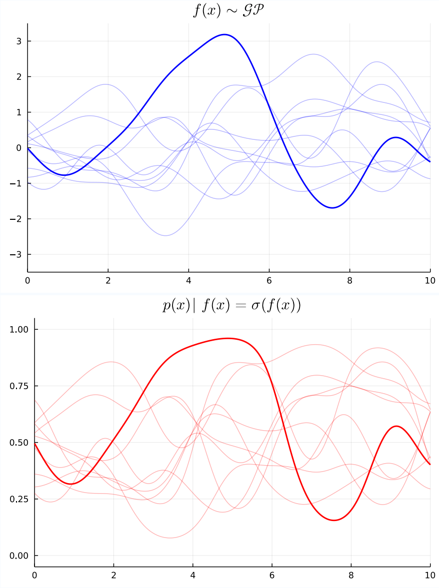
Joint (generative) model: $p(y,f) = p(y\,|\,f)p(f)$





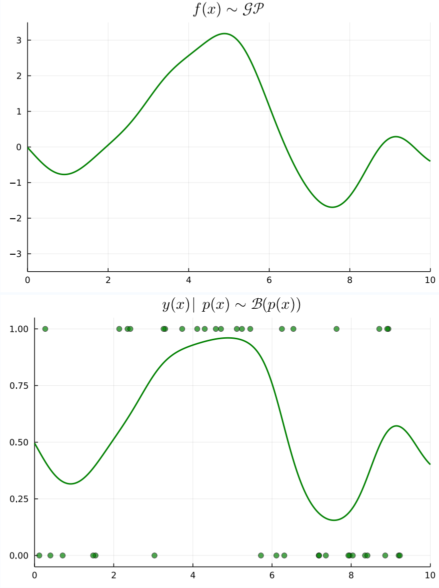
Back to GPs...

Joint (generative) model: $p(y, f) = p(y \mid f)p(f)$



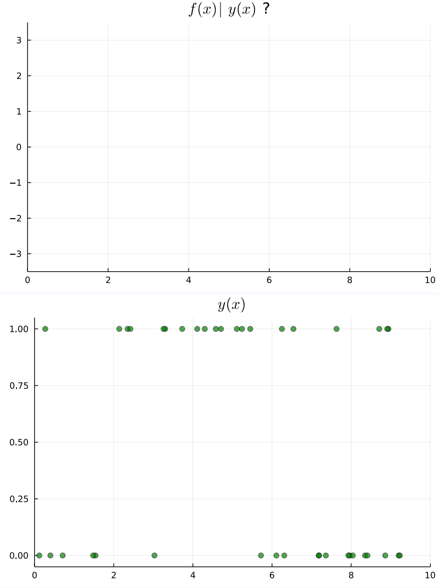
Back to GPs...

Joint (generative) model: $p(y, f) = p(y \mid f)p(f)$



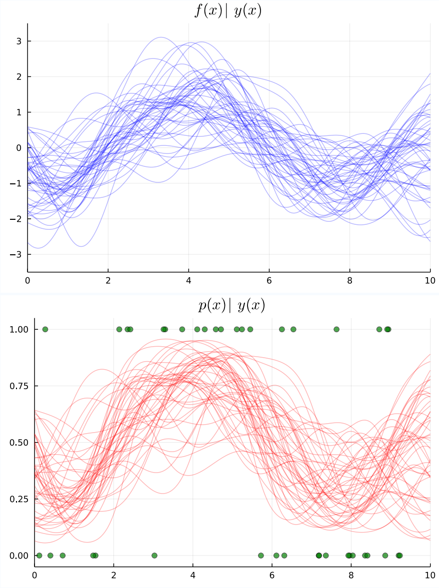
Back to GPs...

Posterior: $p(f \mid y) = p(y \mid f)p(f)/p(y)$



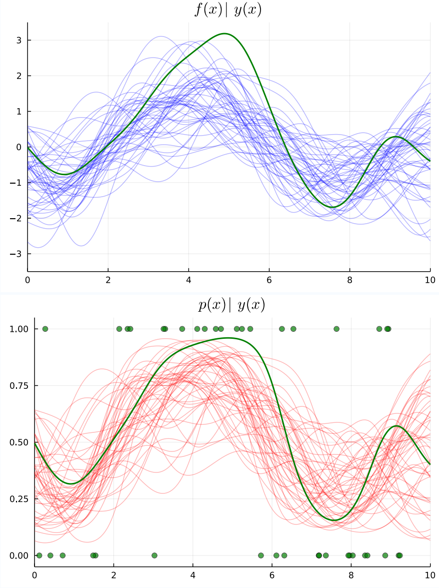
Back to GPs...

Posterior: $p(f \mid y) = p(y \mid f)p(f)\,/\,p(y)$



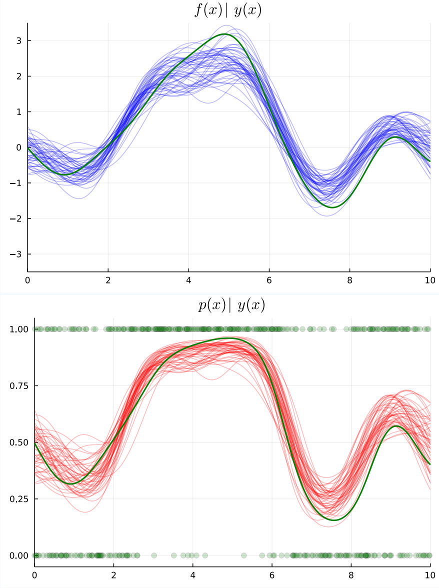
Back to GPs...

Posterior: $p(f \mid y) = p(y \mid f)p(f) / p(y)$



Back to GPs...

Posterior: $p(f \mid y) = p(y \mid f)p(f)/p(y)$ for more data



Outline

- ✓ Gaussian processes with Gaussian likelihood
- ✓ What is the likelihood? Connecting observations and Gaussian process prior

3. **Non-Gaussian likelihoods: what happens to the posterior?**
4. How to approximate the intractable
5. Comparison

Posterior

Likelihood

$$p(\mathbf{y} \mid \mathbf{f})$$

Joint distribution

$$p(\mathbf{y}, \mathbf{f}) = p(\mathbf{y} \mid \mathbf{f})p(\mathbf{f})$$

Posterior

$$\mathbf{f} \mapsto p(\mathbf{f} \mid \mathbf{y}) = \frac{p(\mathbf{y} \mid \mathbf{f})p(\mathbf{f})}{p(\mathbf{y})}$$

$$\mathbf{y} \mapsto \left(\mathbf{f} \mapsto p(\mathbf{f} \mid \mathbf{y})\right)$$

Posterior predictions

At new point x^*:

$$p(f^* \mid x^*, \mathbf{x}, \mathbf{y}) = \int p(f^* \mid x^*, \mathbf{x}, \mathbf{f})\, p(\mathbf{f} \mid \mathbf{x}, \mathbf{y})\, \mathrm{d}\mathbf{f}$$

At training data:

$$p(\mathbf{f} \mid \mathbf{x}, \mathbf{y}) = \frac{p(\mathbf{f} \mid \mathbf{x}) \prod_{n=1}^{N} p(y_n \mid f(x_n))}{\int p(\mathbf{f}' \mid \mathbf{x}) \prod_{n=1}^{N} p(y_n \mid f'(x_n))\, \mathrm{d}\mathbf{f}'}$$

$$p(\mathbf{f} \mid \mathbf{y}) = \frac{1}{Z} p(\mathbf{f}) \prod_{n=1}^{N} p(y_n \mid f_n)$$

$$Z = p(\mathbf{y} \mid \mathcal{M}) = \int p(\mathbf{f} \mid \mathcal{M}) \prod_{n=1}^{N} p(y_n \mid f_n, \mathcal{M})\, \mathrm{d}\mathbf{f}$$

“marginal likelihood” or “evidence” given model $\mathcal{M}$

Posterior

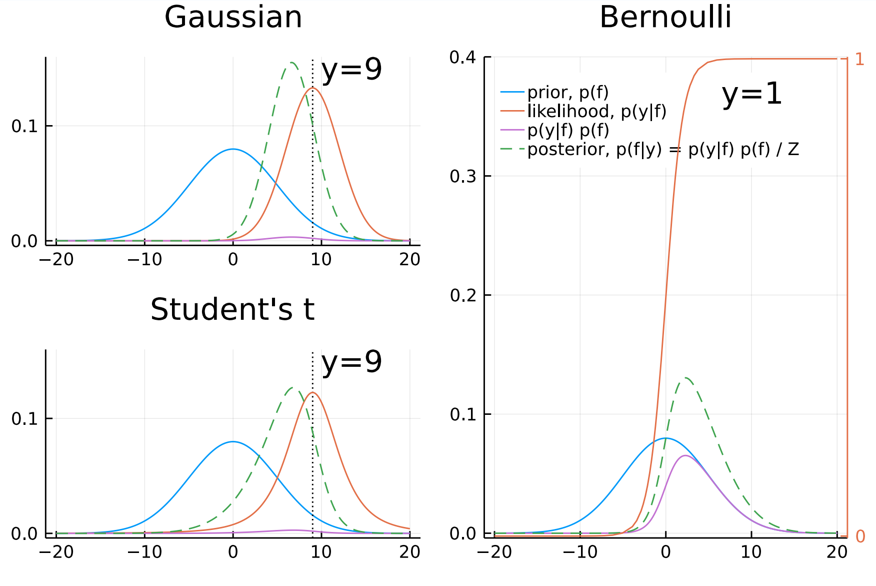
$$p(\mathbf{f}\,|\,\mathbf{y}) = \frac{1}{Z} p(\mathbf{f}) \prod_{n=1}^{N} p(y_n\,|\,f_n)$$

Gaussian (process) prior $p(f(\cdot))$ … $\qquad p(\mathbf{f}) = \mathcal{N}(\mathbf{f}\,|\,\mathbf{0}, \mathrm{K})$

& Gaussian likelihood: conjugate case $\rightarrow$ posterior Gaussian

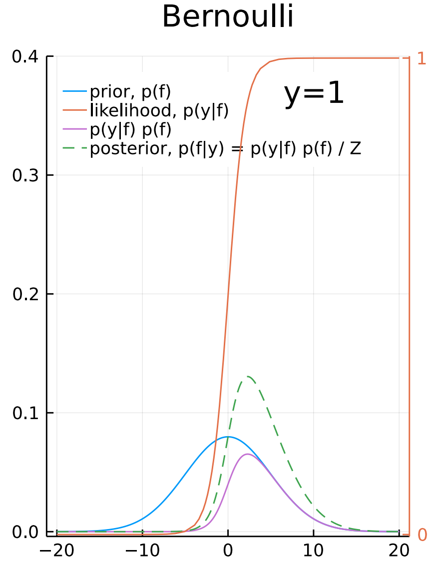
& non-Gaussian $p(y|f)$ $\rightarrow p(\mathbf{f}\,|\,\mathbf{y})$ also non-Gaussian, intractable

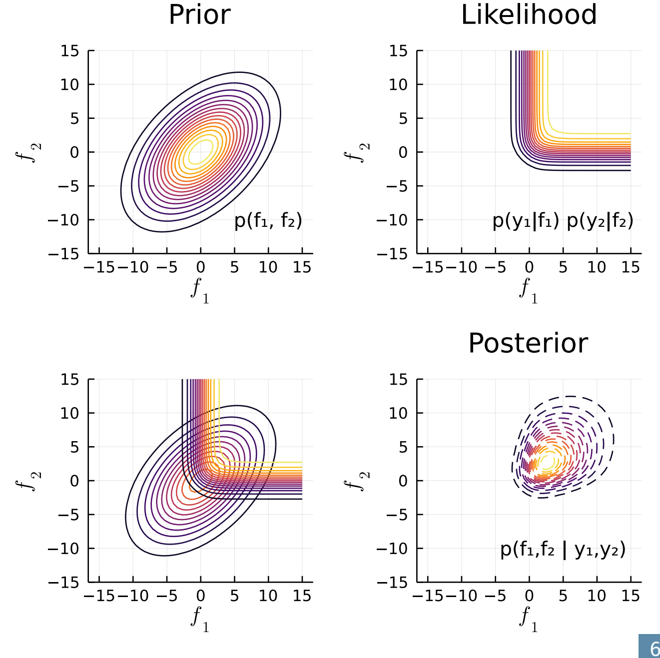
1D examples



Gaussian
y=9
0.1
0.0
−20
−10
0
10
20
Student's t
y=9
0.1
0.0
−20
−10
0
10
20
Bernoulli
y=1
prior, p(f)
likelihood, p(y|f)
p(y|f) p(f)
posterior, p(f|y) = p(y|f) p(f) / Z
0.4
0.3
0.2
0.1
0.0
1
0
−20
−10
0
10
20

Bernoulli example in 2D





Posterior for N observations

$$p(\mathbf{f} \mid \mathbf{y}) = \frac{p(\mathbf{f}) \prod_{n=1}^{N} p(y_n \mid f_n)}{\int p(\mathbf{f}') \prod_{n=1}^{N} p(y_n \mid f_n') \, \mathrm{d}\mathbf{f}'}$$

$$\begin{aligned} f_1 &= f(x_1) \\ f_2 &= f(x_2) \\ &\vdots \\ f_N &= f(x_N) \end{aligned}$$

Summary so far

- What is the likelihood $p(y \mid f)$?
- When is it non-Gaussian?
- Why does the posterior $p(f \mid y)$ become intractable?

Questions?! :)

Outline

✓ Gaussian processes with Gaussian likelihood
✓ What is the likelihood? Connecting observations and Gaussian process prior
✓ Non-Gaussian likelihoods: what happens to the posterior?
4. **How to approximate the intractable**
5. Comparison

Approximations

- Joint model:

$$p(\mathbf{y}, \mathbf{f}) = p(\mathbf{y} \mid \mathbf{f})\, p(\mathbf{f}) = \prod_{n=1}^{N} p(y_n | f_n) \mathcal{N}(\mathbf{f} \mid \mathbf{0}, \mathrm{K})$$

- Posterior distribution at training points:

$$p(\mathbf{f}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f})}{p(\mathbf{y})} \approx q(\mathbf{f})$$

- Posterior of f^* for new test point $\mathbf{x}^*$:

$$p(f^*|\mathbf{y}) = \int p(f^*|\mathbf{f})p(\mathbf{f}|\mathbf{y})\, \mathrm{d}\mathbf{f} \approx \int p(f^*|\mathbf{f})q(\mathbf{f})\, \mathrm{d}\mathbf{f} \equiv q(f^*)$$

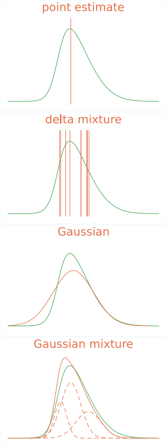
- Predictive distribution

$$p(y^*|\mathbf{y}) = \int p(y^* | f^*)p(f^*|\mathbf{y})\, \mathrm{d}f^* \approx \int p(y^* | f^*)q(f^*)\, \mathrm{d}f^*$$

Analytically intractable distributions!

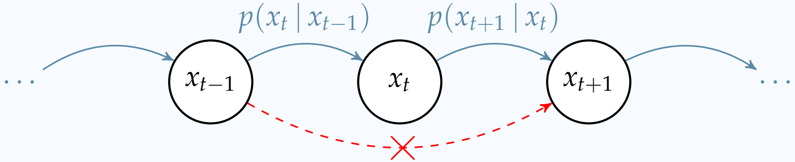
Approximating distributions

- Delta distribution
  - Point estimate
- Mixture of delta distributions
  - **Markov Chain Monte Carlo (MCMC)**
  - Neural network ensembles...
- Gaussian distribution
  - Laplace Approximation (LA)
  - Variational Bayes/Variational Inference (VB / VI)
  - Expectation Propagation (EP), PowerEP, ...
- Mixture of Gaussians
- ...



Markov Chain Monte Carlo

Markov Chain



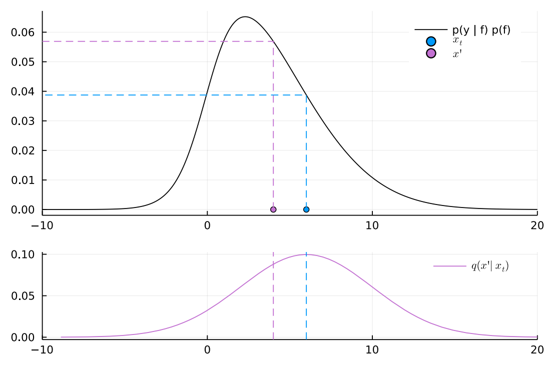
- Samples $x_1, \ldots, x_T$
- "Markov" = 1-step history
- $x_{t+1} \sim p(x_{t+1} \mid x_t)$, independent of $x_{t-1}, \ldots, x_1$

Markov Chain Monte Carlo (MCMC)

Generate samples $\{x_t\} \sim p(f \mid y)$

Requires:

- *unnormalized* posterior $h(f) = p(y \mid f)p(f)$
- Markov proposal $q(x' \mid x_t)$
- initial x_0



In each iteration t:

1. Random proposal $x' \sim q(x' \mid x_t)$
2. Acceptance probability $\frac{h(x')}{h(x_t)}$ → ensures sampling from $p(f \mid y)$

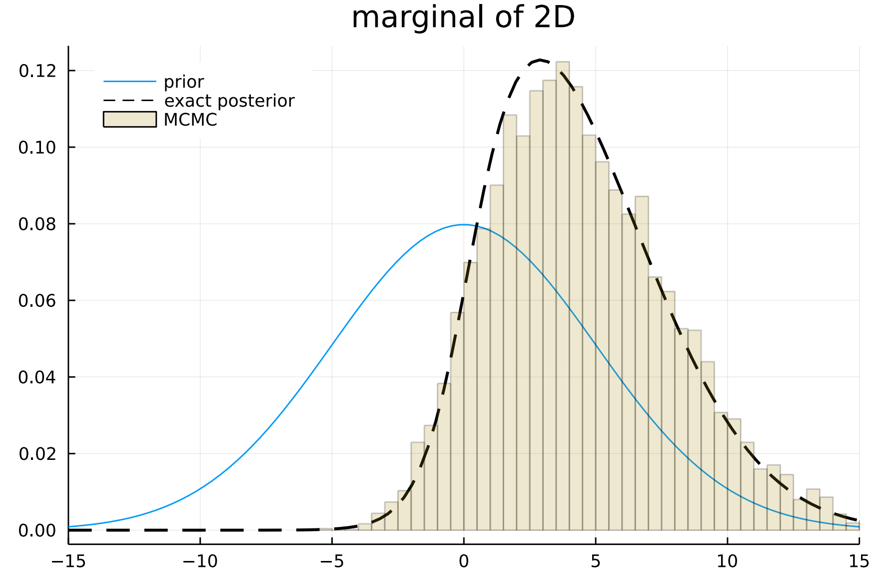
accept: $x_{t+1} = x'$ reject: copy $x_{t+1} = x_t$

$h(x') > h(x_t)$: always accepts → climbs uphill

Demo: MCMC in 2D

tinyurl.com/nongaussian-inference-viz-v1

Marginals



marginal of 2D
prior
exact posterior
MCMC
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−15
−10
−5
0
5
10
15

MCMC: important properties

- burn-in
- acceptance ratio
- auto-correlation, effective sample size (ESS); thinning to save memory
- mixing and multiple chains ($\hat{R}$)
- better proposals (HMC, NUTS) $\rightarrow$ use robust implementations

+ very accurate (gold-standard)

– very slow, predictions require keeping all (thinned) samples around

Michael Betancourt's `betanalpha.github.io/writing/`

MCMC: robust implementations

- Stan
- PyMC3
- Pyro & NumPyro
- TensorFlow Probability (GPflow)
- Turing.jl

Outline

- ✓ Gaussian processes with Gaussian likelihood
- ✓ What is the likelihood? Connecting observations and Gaussian process prior
- ✓ Non-Gaussian likelihoods: what happens to the posterior?

4. **How to approximate the intractable**
   - ✓ with samples: MCMC
   - 4.2 **with Gaussians**
     - Laplace Approximation
     - Variational Inference
     - Expectation Propagation
5. Comparison

Gaussian approximations

Approximating the exact posterior with Gaussian

Approximating the posterior at observations:

$$p(\mathbf{f} \mid \mathbf{y}) \approx q(\mathbf{f}) = \mathcal{N}(\mathbf{f} | \mu = ?, \Sigma = ?)$$

Predictions at new points:

$$p(f^* \mid x^*, \mathbf{y}) \approx q(f^*) = \underbrace{\int p(f^* \mid x^*, \mathbf{f})\, q(\mathbf{f})\, \mathrm{d}\mathbf{f}}_{\text{closed-form integral!}}$$

Demo: What does this mean for Gaussian processes?

tinyurl.com/nongaussian-inference-viz-v1

Choosing μ and Σ for $q(\mathbf{f})$

$$p(\mathbf{f} \mid \mathbf{y}) \approx q(\mathbf{f}) = \mathcal{N}(\mathbf{f} | \mu = ?, \Sigma = ?)$$

locally: match mean & variance at point

globally: minimise divergence

Laplace approximation

Variational Inference (VI)

Expectation Propagation (EP)

Laplace approximation

Laplace approximation

Idea: log of Gaussian pdf = quadratic polynomial

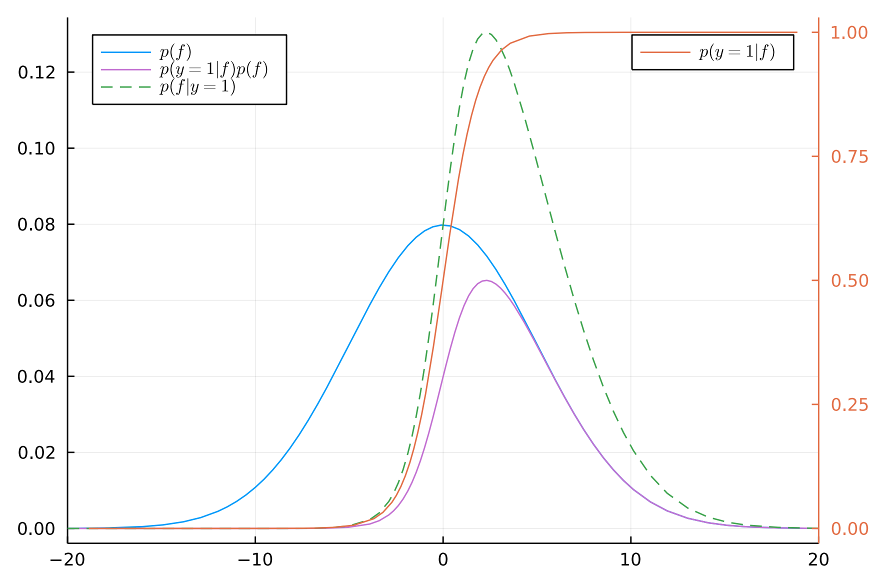
$$p_{\mathcal{N}}(\mathbf{f}) = \frac{1}{\sqrt{(2\pi)^k|\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{f}-\mu)^\top \Sigma^{-1}(\mathbf{f}-\mu)\right)$$

quadratic polynomial through approximation:
2nd-order Taylor expansion of log of $h(f) = p(y \mid f)p(f)$ at $\hat{f}$

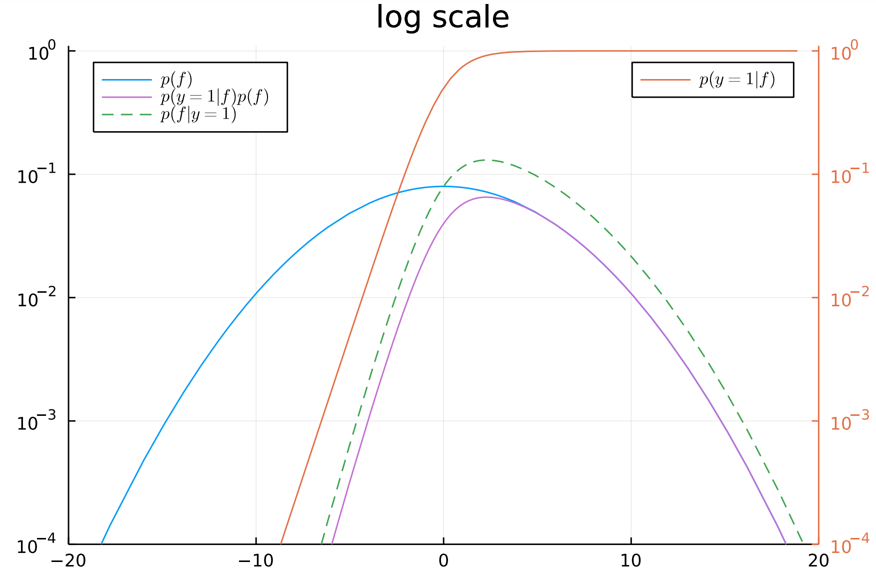
$$g(x+\delta) \approx g(x) + \left(\frac{\mathrm{d}g}{\mathrm{d}x}(x)\right)\delta + \frac{1}{2!}\left(\frac{\mathrm{d}^2 g}{\mathrm{d}x^2}(x)\right)\delta^2$$

1. Find **mode** of posterior
   2nd-order gradient optimisation (e.g. Newton's method)
2. Match **curvature** (Hessian) at mode

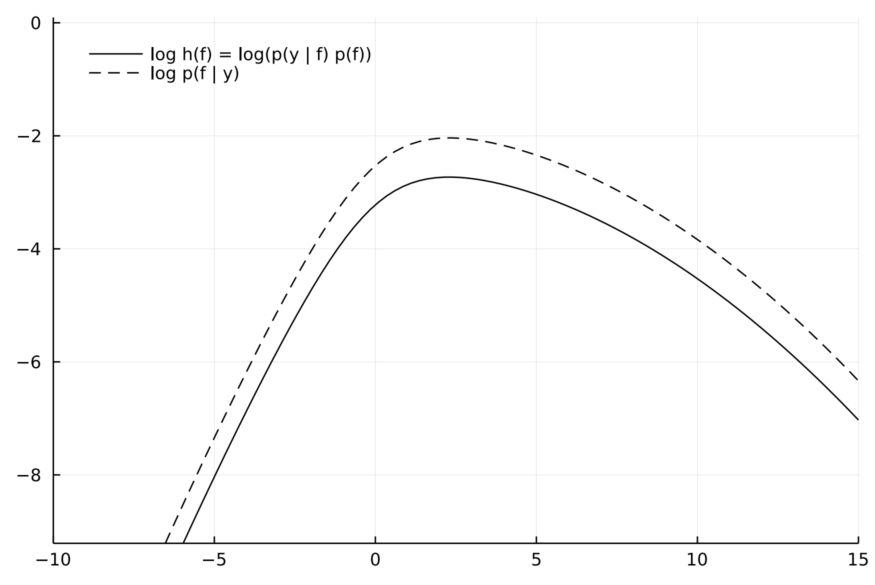
$p(f \mid y) = \frac{1}{Z} p(y \mid f) p(f)$



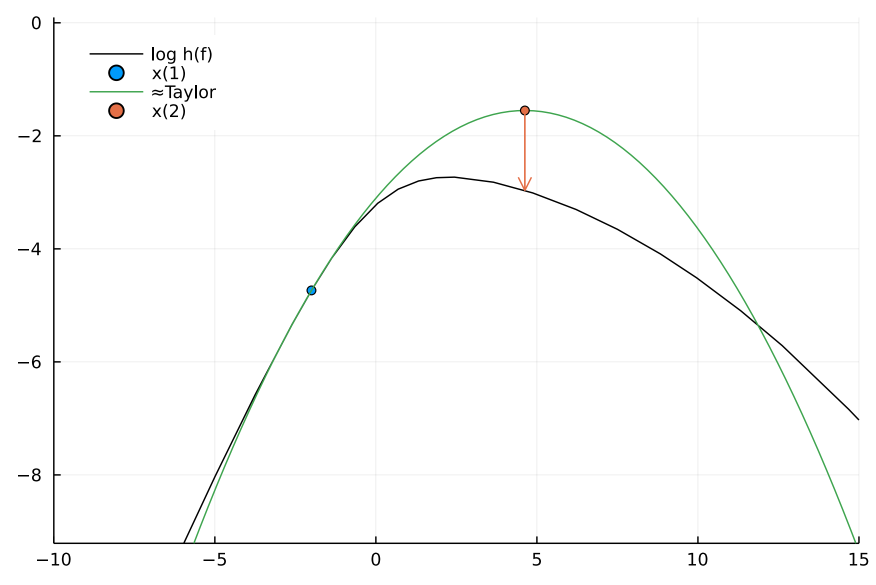
$$\log p(f \mid y) = -\log Z + \log p(y \mid f) + \log p(f)$$



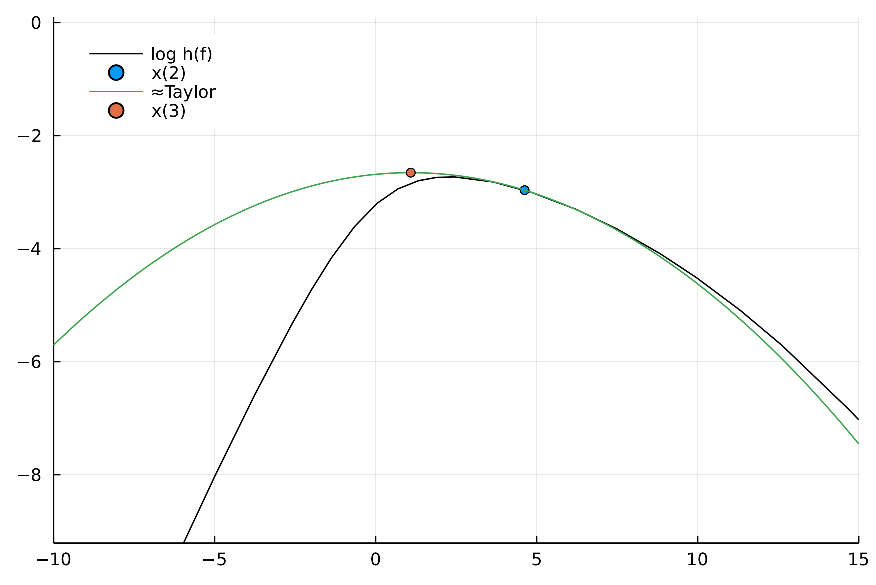
$\log p(f \mid y) = -\log Z + \log h(f)$



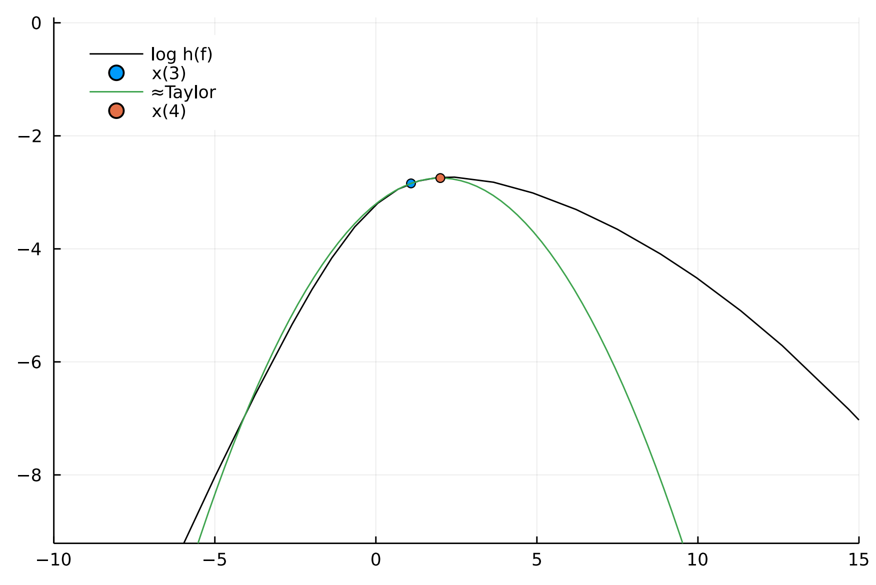
Newton's method



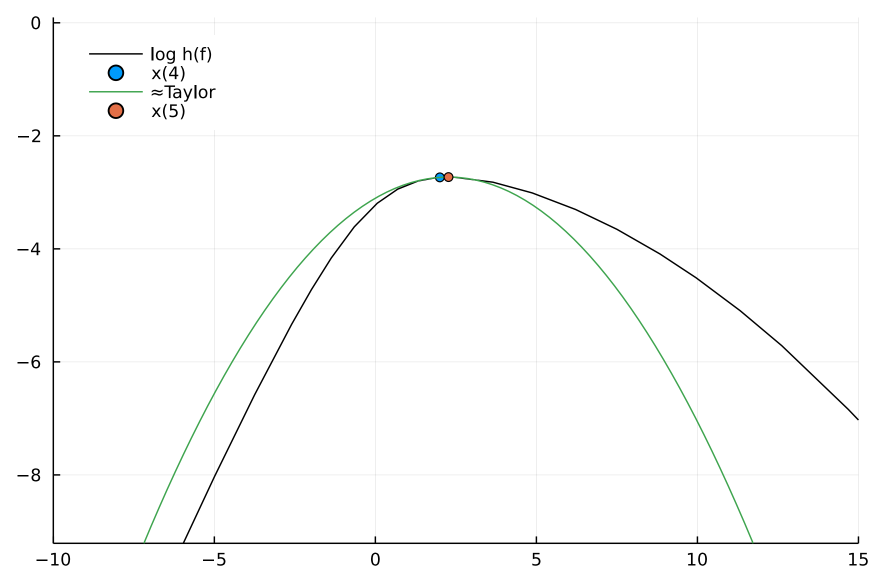
Newton's method



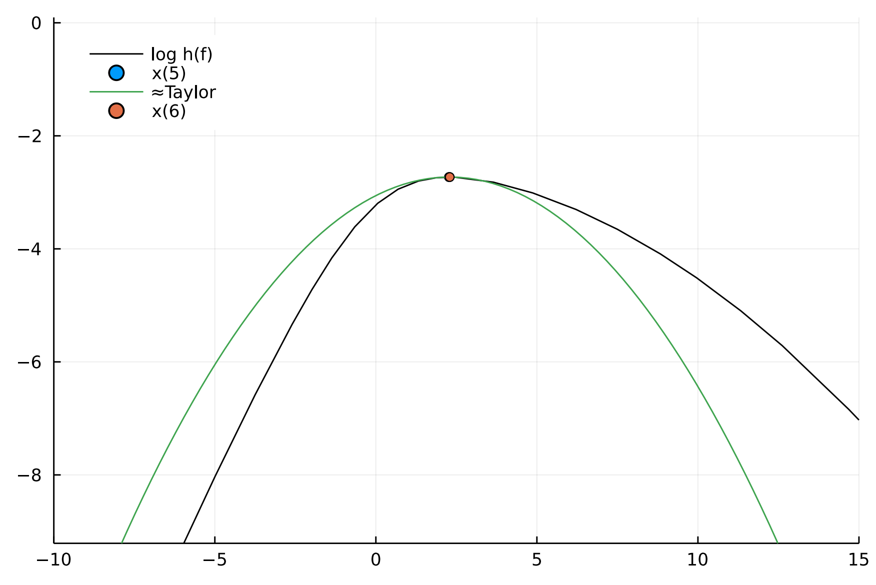
Newton's method



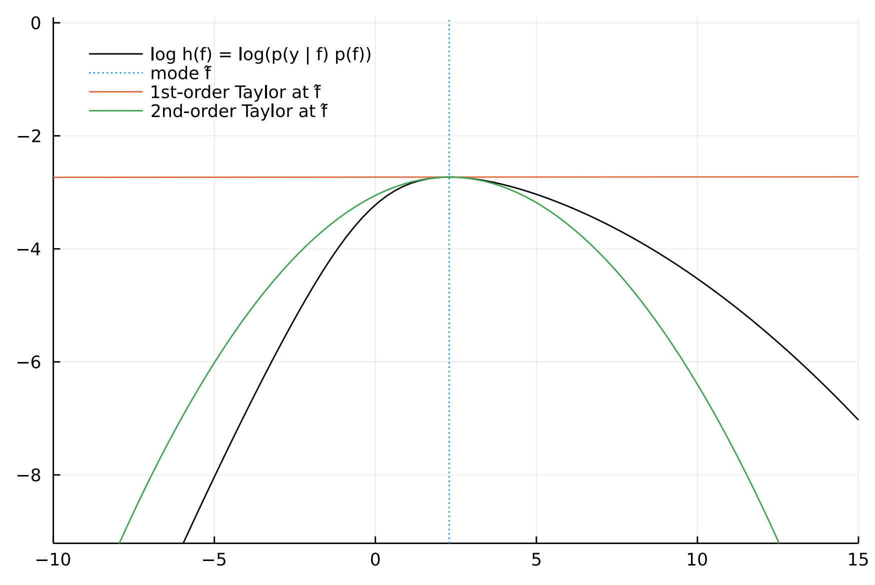
Newton's method



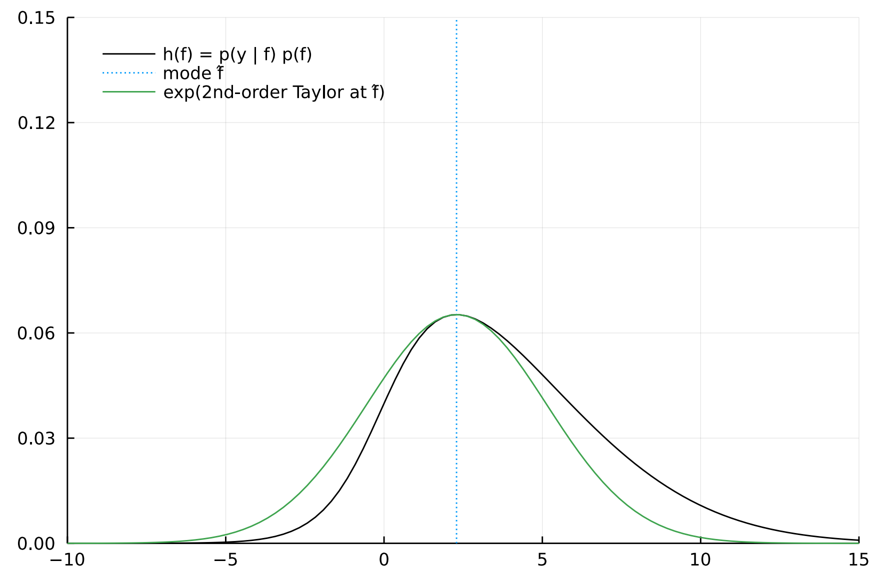
Newton's method



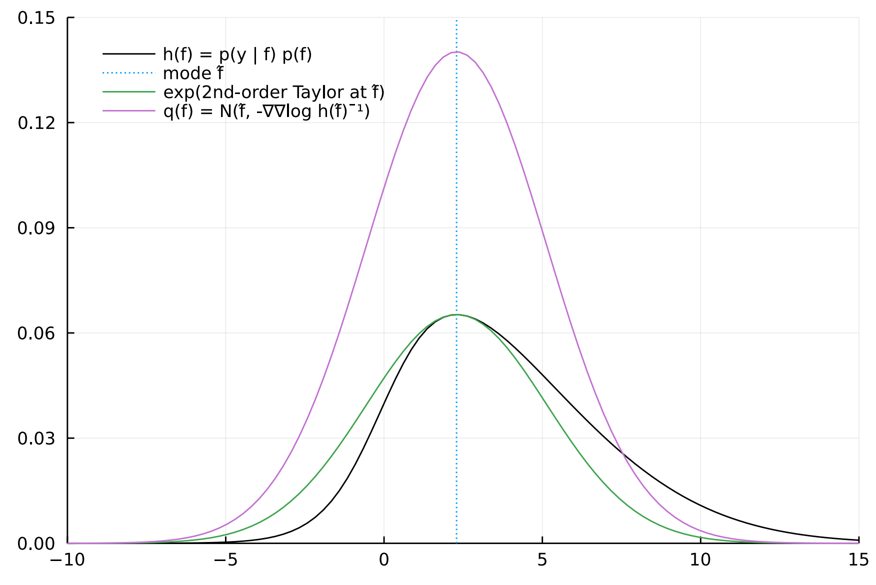
$\log p(f \mid y) + \log Z = \log h(f) \approx \mathcal{O}(f^2)$



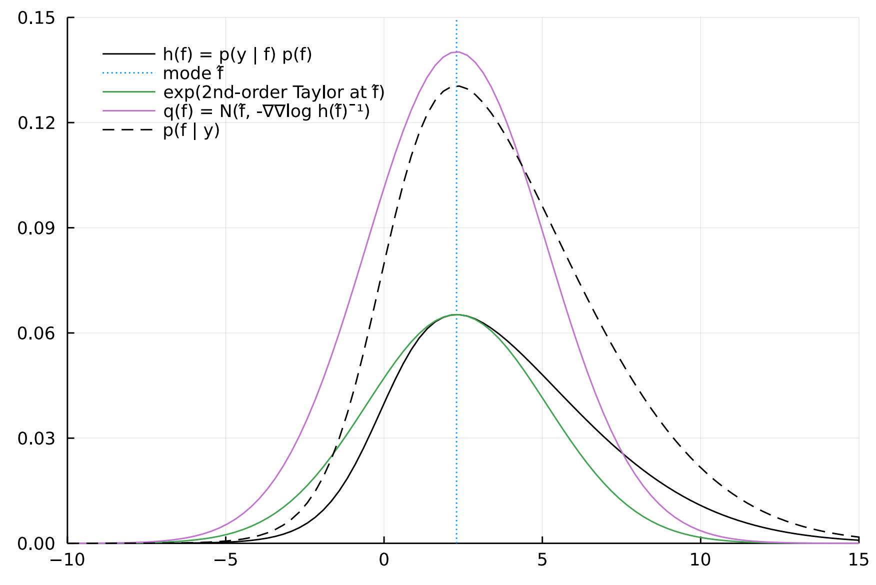
$p(f \mid y)\, Z \approx \exp(\mathcal{O}(f^2))$



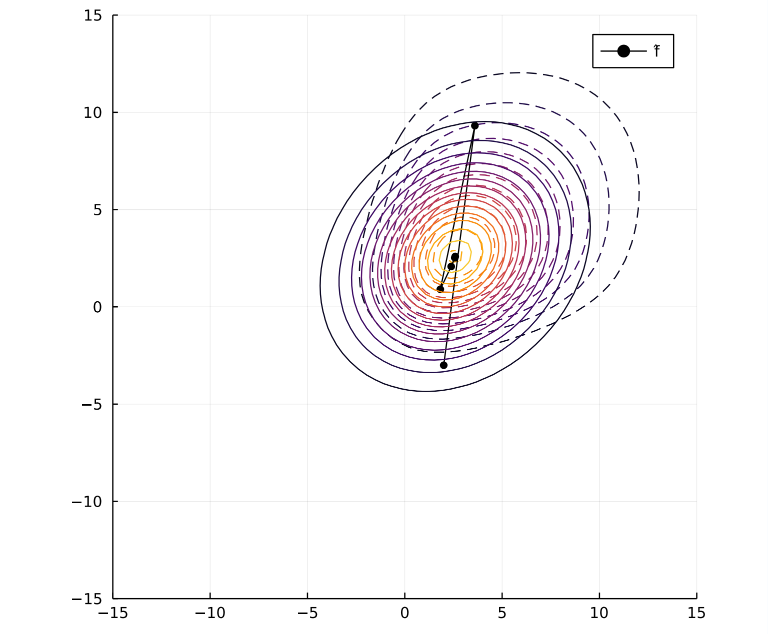
$$p(f \mid y) \approx \mathcal{N}(f \mid \hat{f}, -(\mathrm{d}^2 \log h / \mathrm{d}f^2)^{-1})$$



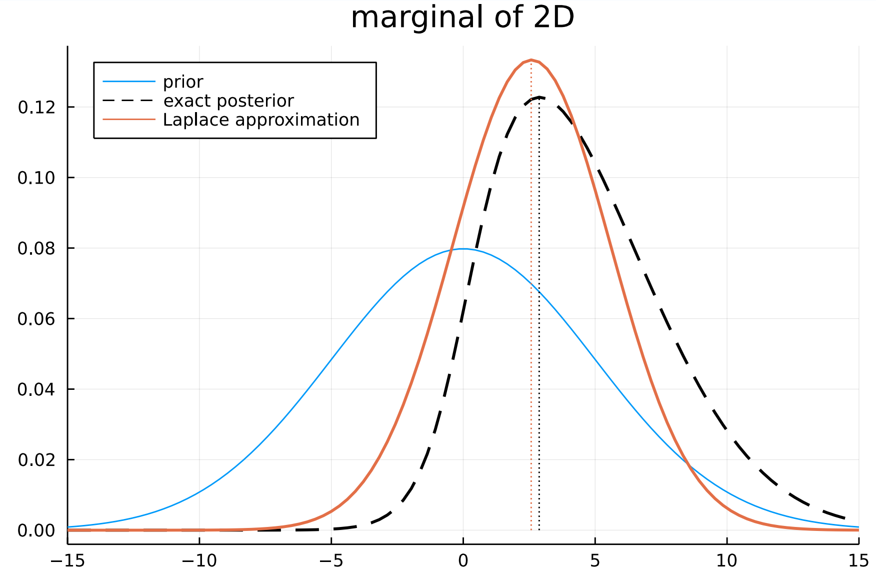
$p(f \mid y) \approx \mathcal{N}(f \mid \hat{f}, -(\mathrm{d}^2 \log h / \mathrm{d}f^2)^{-1}) = q(f)$



Laplace in 2D example



Laplace in 2D: marginals



Laplace approximation: important properties

- find mode: Newton's method
- match curvature (Hessian) at mode
- "point estimate++"

+ simple, fast

– poor approximation if mode is not representative (e.g. Bernoulli)

– may not converge for non-log-concave likelihoods [Hartmann and Vanhatalo, 2018]

Choosing μ and Σ for $q(\mathbf{f})$

$$p(\mathbf{f} \mid \mathbf{y}) \approx q(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\mu = ?, \Sigma = ?)$$

locally: match mean & variance at point

globally: minimise divergence

Laplace approximation

Variational Inference (VI)

Expectation Propagation (EP)

Variational Bayes (VB)
Variational Inference (VI)

Variational inference: the big picture

Recipe for approximating **intractable distribution** $p \in \mathcal{P}$

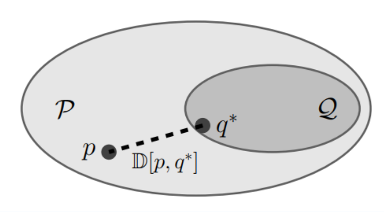
1. Define some "simple" family of distributions $\mathcal{Q}$.
2. Define some way to compute a "distance" $\mathbb{D}[p, q]$ between intractable distribution p and each distribution $q \in \mathcal{Q}$

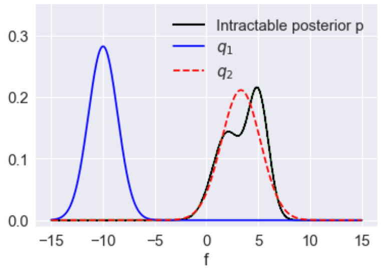
$$\mathbb{D}[p, q_1] > \mathbb{D}[p, q_2]$$

3. Search for $q \in \mathcal{Q}$ such that $\mathbb{D}[p, q]$ is minimized

$$q^* = \arg\min_{q \in \mathcal{Q}} \mathbb{D}[p, q]$$

4. Use q^* as an approximation of p





How to "measure distances" between distributions?

Here: *Kullback–Leibler (KL) divergence*

$$\mathbb{D}[p,q] := \mathrm{KL}[q \,\|\, p] = \int q(f) \log \frac{q(f)}{p(f)} \mathrm{d}f = \mathbb{E}_q \left[\log \frac{q(f)}{p(f)} \right]$$

Important properties:

1. Non-symmetric: $\mathrm{KL}[q \,\|\, p] \neq \mathrm{KL}[p \,\|\, q]$
2. Positive: $\mathrm{KL} \geq 0$ (Gibbs' inequality)
3. Minimum: $\mathrm{KL}[q \,\|\, p] = 0 \iff q \equiv p$.

Variational inference

$$q(\mathbf{f}) = \mathcal{N}(\mu, \Sigma)$$

$$\operatorname*{argmin}_{\mu,\Sigma} \mathrm{KL}\left[q(\mathbf{f}) \| p(\mathbf{f} \mid \mathbf{y})\right]$$

Variational inference: Minimizing $\mathrm{KL}[q(\mathbf{f}) \| p(\mathbf{f} \mid \mathbf{y})]$

$$
\begin{aligned}
\mathrm{KL}[q(\mathbf{f}) \| p(\mathbf{f} \mid \mathbf{y})] &= \int q(\mathbf{f}) \left[\log \frac{q(\mathbf{f})}{p(\mathbf{f} \mid \mathbf{y})} \right] \mathrm{d}\mathbf{f} \\
&= \int q(\mathbf{f}) \left[\log q(\mathbf{f}) - \log p(\mathbf{f} \mid \mathbf{y}) \right] \mathrm{d}\mathbf{f} \\
&= \int q(\mathbf{f}) \left[\underbrace{\log q(\mathbf{f}) - \overbrace{\log p(\mathbf{f})}} \overbrace{- \log p(\mathbf{y} \mid \mathbf{f}) + \log p(\mathbf{y})} \right] \mathrm{d}\mathbf{f} \\
&= \int q(\mathbf{f}) \left[\log \frac{q(\mathbf{f})}{p(\mathbf{f})} \right] \mathrm{d}\mathbf{f} - \int q(\mathbf{f}) \left[\log p(\mathbf{y} \mid \mathbf{f}) \right] \mathrm{d}\mathbf{f} + \log p(\mathbf{y}) \\
&= \mathrm{KL}[q(\mathbf{f}) \| p(\mathbf{f})] - \int q(\mathbf{f}) \left[\log p(\mathbf{y} \mid \mathbf{f}) \right] \mathrm{d}\mathbf{f} + \log p(\mathbf{y}) \\
\log p(\mathbf{y}) &= \int q(\mathbf{f}) \left[\log p(\mathbf{y} \mid \mathbf{f}) \right] \mathrm{d}\mathbf{f} - \mathrm{KL}[q(\mathbf{f}) \| p(\mathbf{f})] + \mathrm{KL}[q(\mathbf{f}) \| p(\mathbf{f} \mid \mathbf{y})]
\end{aligned}
$$

Variational inference: Minimizing $\mathrm{KL}[q(\mathbf{f}) \,\|\, p(\mathbf{f} \mid \mathbf{y})]$ by bounding

$$\log p(\mathbf{y}) = \underbrace{\int q(\mathbf{f}) \big[\log p(\mathbf{y} \mid \mathbf{f})\big] \mathrm{d}\mathbf{f} - \mathrm{KL}[q(\mathbf{f}) \,\|\, p(\mathbf{f})]}_{\mathcal{L}[q]} + \underbrace{\mathrm{KL}[q(\mathbf{f}) \,\|\, p(\mathbf{f} \mid \mathbf{y})]}_{\geq 0}$$

$$\geq \int q(\mathbf{f}) \big[\log p(\mathbf{y} \mid \mathbf{f})\big] \mathrm{d}\mathbf{f} - \mathrm{KL}[q(\mathbf{f}) \,\|\, p(\mathbf{f})] = \mathcal{L}[q]$$

- $\log p(\mathbf{y})$ is a constant
- $\mathcal{L}[q]$ does **not** depend on $p(\mathbf{f} \mid \mathbf{y})$
- $\mathcal{L}[q] \leq \log p(\mathbf{y})$, so $\mathcal{L}[q]$ is *lower bound* on marginal log likelihood $\log p(\mathbf{y})$
- Maximizing $\mathcal{L}[q]$ is equivalent to minimizing $\mathrm{KL}[q(\mathbf{f}) \,\|\, p(\mathbf{f} \mid \mathbf{y})]$

Key take-away: we can fit variational approximation q by optimizing $\mathcal{L}$

Variational inference: ELBO

$$\log p(\mathbf{y}) \geq \mathcal{L}[q] = \underbrace{\int q(\mathbf{f}) \big[\log p(\mathbf{y} \mid \mathbf{f})\big] \mathrm{d}\mathbf{f}}_{\text{data fit}} - \underbrace{\mathrm{KL}[q(\mathbf{f}) \parallel p(\mathbf{f})]}_{\text{regularization}}$$

$\mathcal{L}[q]$ often called the *Evidence Lower Bound* (ELBO)

- We approximate $p(\mathbf{f} \mid \mathbf{y}) \approx q(\mathbf{f}) = \mathcal{N}(\mathbf{f} | \mu = ?, \Sigma = ?)$
- Defining $\lambda = \{\mu, \Sigma\}$, we can write $\mathcal{L}[q] = \mathcal{L}(\lambda)$
- In practice, we optimize $\mathcal{L}(\lambda)$ using gradient-based methods

Likelihood term (data fit)

Integral separates for a factorizing likelihood:

$$\int q(\mathbf{f}) \left[\log p(\mathbf{y} \,|\, \mathbf{f}) \right] \mathrm{d}\mathbf{f}$$

$$= \sum_{n=1}^{N} \int q(f_n) \left[\log p(y_n \,|\, f_n) \right] \mathrm{d}f_n$$

Sum over 1D integrals:

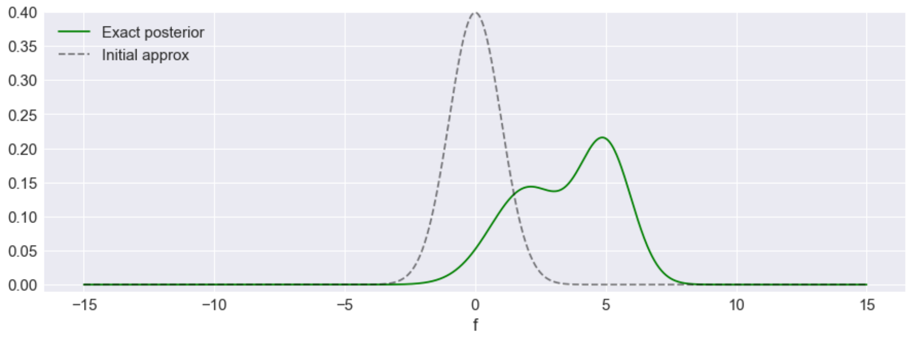
- analytic for some (e.g. Exponential, Gamma, Poisson)
- fast and accurate to approximate numerically (for example Gauss–Hermite quadrature)
- Monte Carlo (e.g. multi-class classification)

Take away #2: We can tractably optimize the bound for non-Gaussian likelihoods

1D Toy example

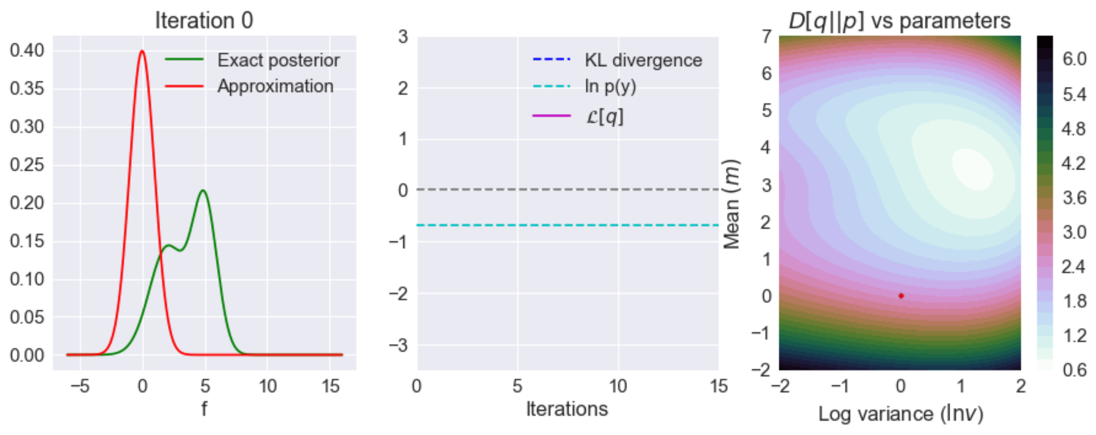
Assume model $p(y, f)$ with some intractable posterior $p(f \mid y)$

- In 1D: $\mathcal{Q}$ is the the set of univariate Gaussians, i.e. $q_\lambda(f) = \mathcal{N}(f \mid m, v)$, and $\boldsymbol{\lambda} = \{m, v\}$
- Initialization: $q(f) = \mathcal{N}(f \mid 0, 1)$



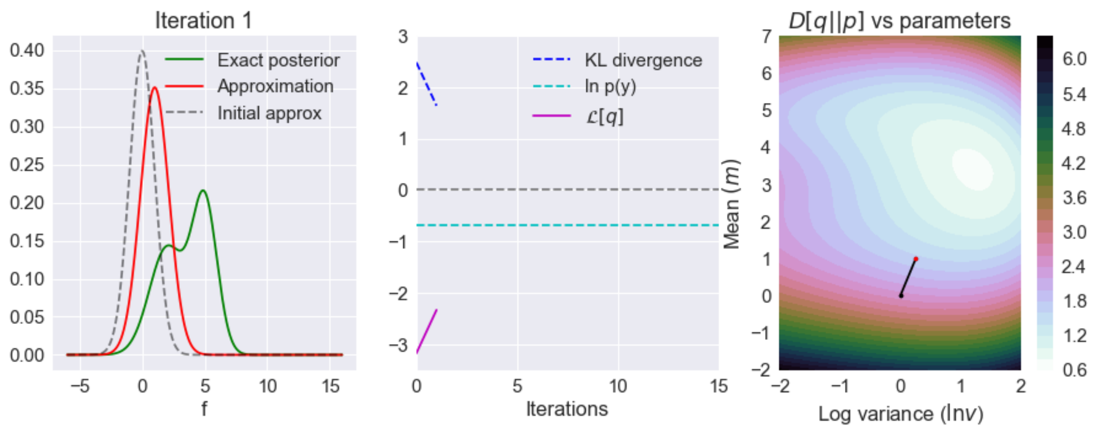
1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\boldsymbol{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \,\|\, p(\boldsymbol{f} \mid \boldsymbol{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



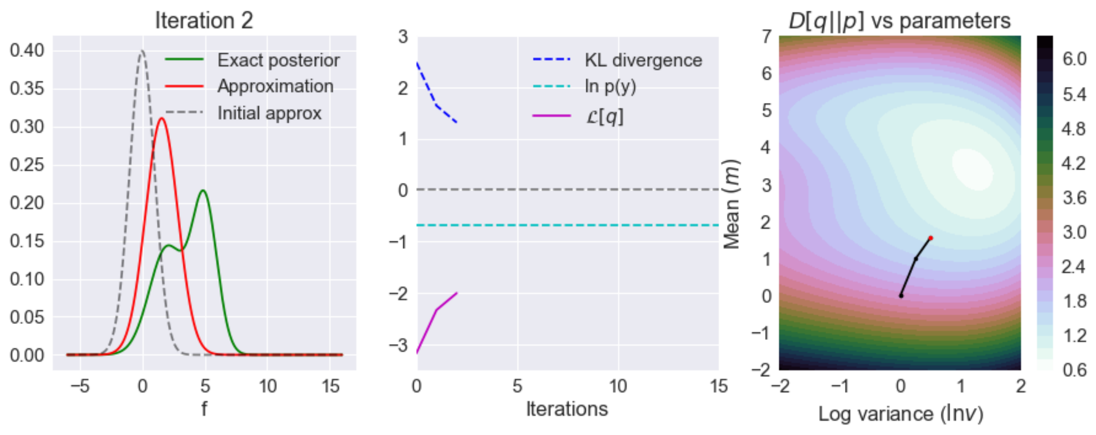
1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\mathbf{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \,\|\, p(\boldsymbol{f} \mid \mathbf{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



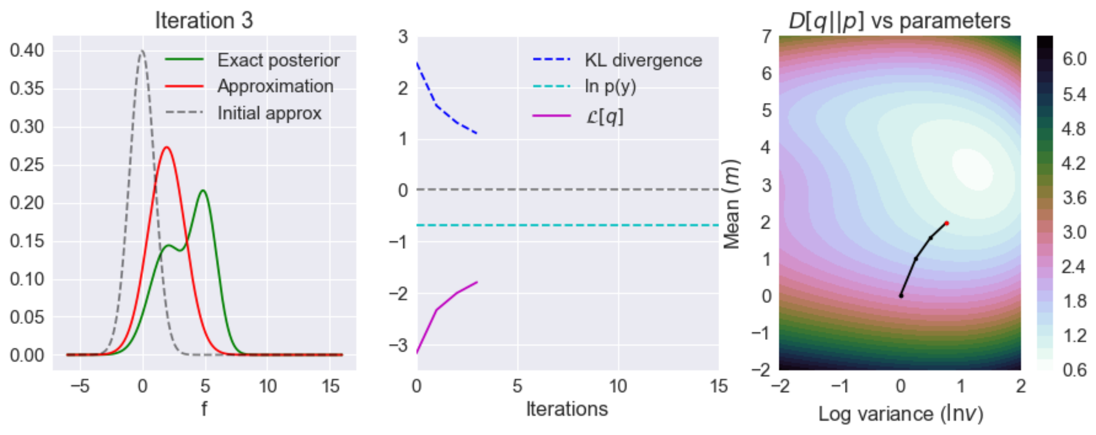
1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\mathbf{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \,\|\, p(\boldsymbol{f} \mid \mathbf{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



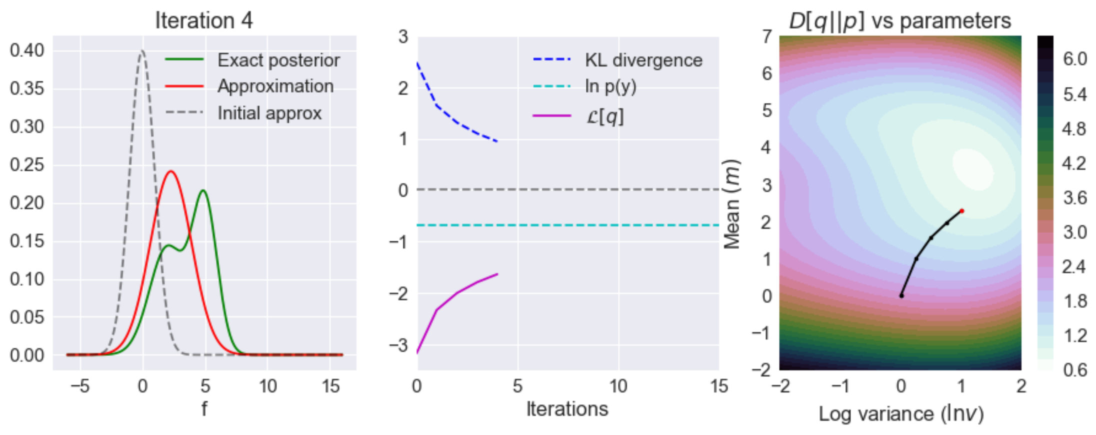
1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\boldsymbol{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \,\|\, p(\boldsymbol{f} \mid \boldsymbol{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



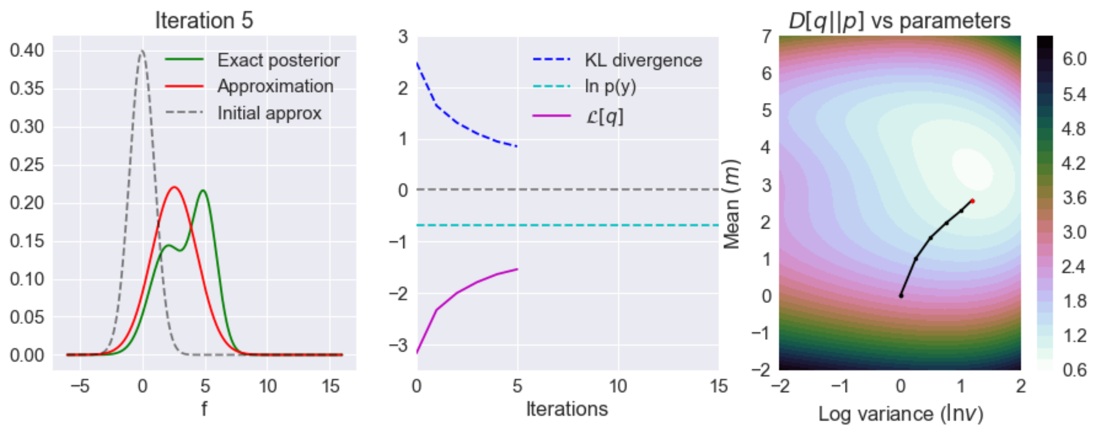
1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\boldsymbol{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \parallel p(\boldsymbol{f} \mid \boldsymbol{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



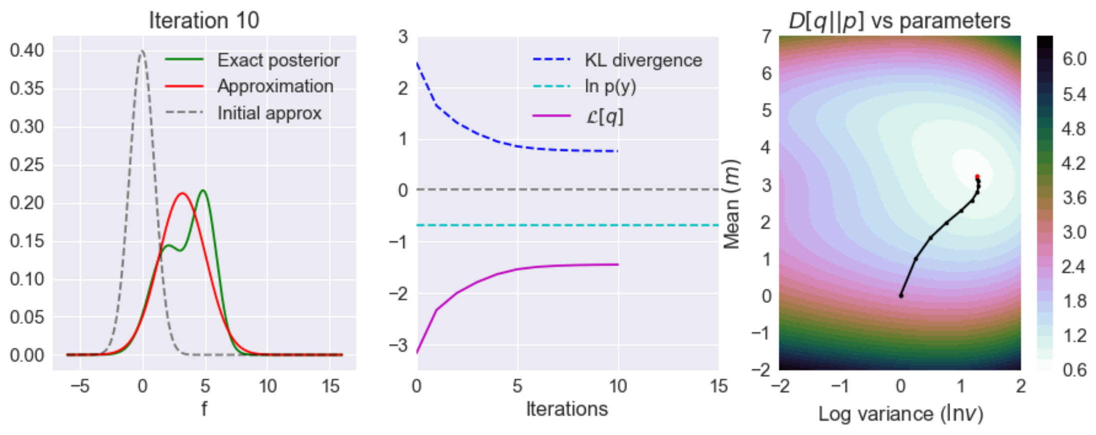
1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\boldsymbol{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \,\|\, p(\boldsymbol{f} \mid \boldsymbol{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



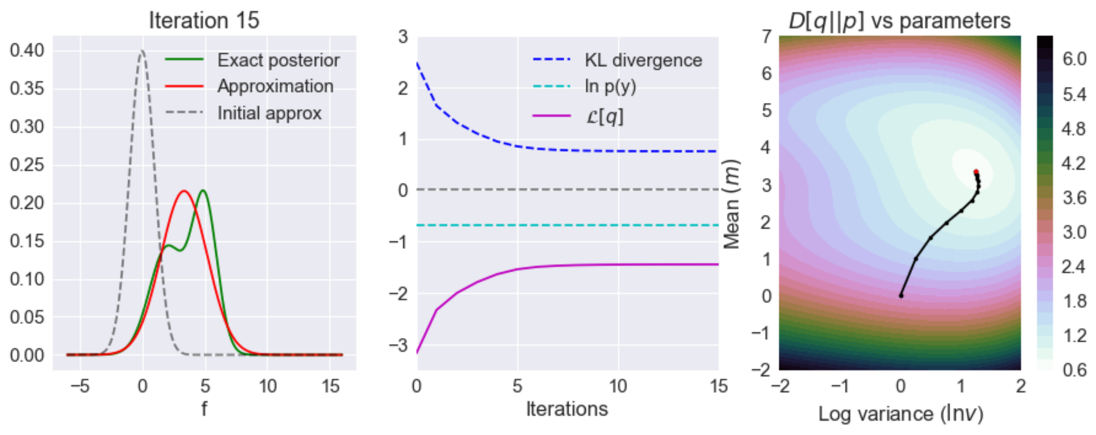
1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\boldsymbol{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \,\|\, p(\boldsymbol{f} \mid \boldsymbol{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



1D Toy example

- Gradient ascent: $\lambda_{i+1} = \lambda_i + \eta \nabla_\lambda \mathcal{L}(\lambda)$
- $\log p(\boldsymbol{y}) = \mathcal{L}(\boldsymbol{\lambda}) + \mathbb{D}\left[q_\lambda(\boldsymbol{f}) \,\|\, p(\boldsymbol{f} \mid \boldsymbol{y})\right] \geq \mathcal{L}(\boldsymbol{\lambda})$



Variational inference: important properties

- principled: directly minimising divergence from true posterior
- mode-seeking (e.g. multi-modal posterior: fits just one, if q is unimodal)

+ minimises a true lower bound $\rightarrow$ convergence

– underestimates variance

Minimising divergences

$$p(\mathbf{f} \mid \mathbf{y}) \approx q(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\mu = ?, \Sigma = ?)$$

✓ $\min \mathrm{KL}[q(\mathbf{f}) \| p(\mathbf{f} \mid \mathbf{y})]$: Variational Inference

2. $\min \mathrm{KL}[p(\mathbf{f} \mid \mathbf{y}) \| q(\mathbf{f})]$: **Expectation Propagation**

Expectation Propagation (EP)

Expectation Propagation

Can we minimise KL divergence in opposite direction?

$$q(\mathbf{f}) = \underset{\mu,\Sigma}{\operatorname{argmin}} \ \mathrm{KL}\left[p(\mathbf{f} \mid \mathbf{y}) \| q(\mathbf{f})\right] = \underset{\mu,\Sigma}{\operatorname{argmin}} \int p(\mathbf{f} \mid \mathbf{y}) \left[\log \frac{p(\mathbf{f} \mid \mathbf{y})}{q(\mathbf{f})}\right] \mathrm{d}\mathbf{f}$$

Exact posterior: $p(\mathbf{f} \mid \mathbf{y}) \propto p(\mathbf{f}) \prod_{n=1}^{N} p(y_n \mid f_n)$

Approximate posterior: $q(\mathbf{f}) \propto p(\mathbf{f}) \prod_{n=1}^{N} t_n(f_n)$

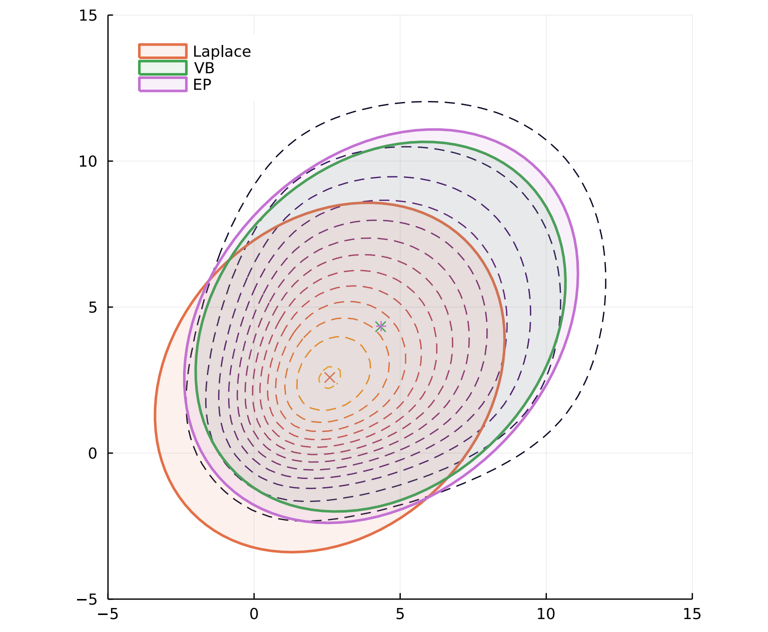
$$t_n = Z_n \mathcal{N}(f_n \mid \tilde{\mu}_n, \tilde{\sigma}_n^2)$$ "sites"

- Expectation propagation iteratively updates the sites for each data point
- minimizes KL of local approximations to the posterior:
  $\min \mathrm{KL}[q(f_n) \frac{p(y_n \mid f_n)}{t_n(f_n)} \| q(f_n) \frac{t'_n(f_n)}{t_n(f_n)}]$

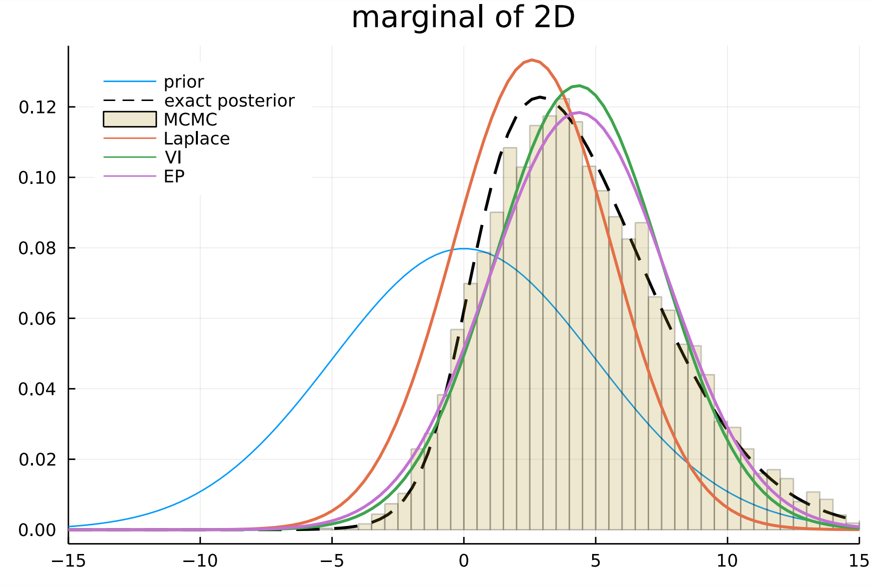
Expectation Propagation: important properties

- multiple passes required to converge
- moment-matching (e.g. covering multiple modes)
- \+ effective for classification
- – not guaranteed to converge
- – updates may be invalid (non-log-concave likelihoods) [Jylänki et al., 2011]

Comparison 2D



Marginals



marginal of 2D
prior
exact posterior
MCMC
Laplace
VI
EP
0.00
0.02
0.04
0.06
0.08
0.10
0.12
−15
−10
−5
0
5
10
15

Outline

- ✓ Gaussian processes with Gaussian likelihood
- ✓ What is the likelihood? Connecting observations and Gaussian process prior
- ✓ Non-Gaussian likelihoods: what happens to the posterior?
- ✓ How to approximate the intractable
  - ✓ with samples: MCMC
  - ✓ with Gaussians
    - Laplace
    - Variational Inference
    - Expectation Propagation

5. **Comparison**

Comparison

Comparison

MCMC

- samples
- gold standard
- slow

Laplace

- $\mathcal{N}$ = curvature at mode
- simple & fast
- often poor approximation

Variational Inference

- $\mathcal{N}$ minimises $\mathrm{KL}[q(\mathbf{f}) \| p(\mathbf{f} \mid \mathbf{y})]$
- principled, any likelihood
- underestimates variance

Expectation Propagation

- $\mathcal{N}$ matches marginal moments
- good calibration in classification
- may not converge

Learning: hyperparameters

What about hyperparameters
(kernel: lengthscales, function scale, etc.; likelihood: noise scale, ...)?

- **MCMC:** priors on hyperparameters, integrate out everything
- **Gaussian approximations:** approximations to marginal likelihood
  - (may be biased)

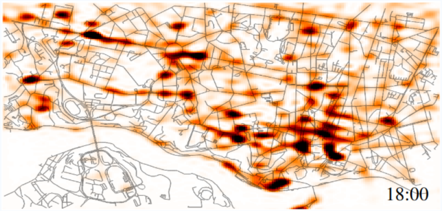
What we did not cover...

- More complex likelihoods (heteroskedastic, zero-inflated, multi-stage...)
- Marginal likelihood approximations for hyperparameter learning [Nickisch and Rasmussen, 2008, Li et al., 2023]
- How parametrisation affects Gaussianity of $p(\mathbf{f} \mid \mathbf{y})$
- Connections between EP and VI ("PowerEP", CVI dual parameterization) [Bui et al., 2017, Adam et al., 2021]
- Other divergences, generalised VI, ...
- Combinations of MCMC and variational methods
- Augmenting likelihood with auxiliary variable
  $\rightarrow$ conditionally conjugate model [Galy-Fajou et al., 2020]

Take-aways

We can...

- create **richer models** with likelihoods beyond the Gaussian
- **learn latent functions** that form the connection between data points
- handle the non-Gaussian posterior with **approximations**
- **trade off** speed, accuracy, and ease-of-use



References I

Adam, V., Chang, P., Khan, M. E. E., and Solin, A. (2021).
Dual parameterization of sparse variational gaussian processes.
NeurIPS.

Bui, T. D., Yan, J., and Turner, R. E. (2017).
A unifying framework for Gaussian process pseudo-point approximations using Power Expectation Propagation.
Journal of Machine Learning Research, 18(104):1–72.

Galy-Fajou, T., Wenzel, F., and Opper, M. (2020).
Automated augmented conjugate inference for non-conjugate Gaussian process models.
AISTATS.

Hartmann, M. and Vanhatalo, J. (2018).
Laplace approximation and natural gradient for Gaussian process regression with heteroscedastic student-t model.
Statistics and Computing, 29(4):753–773.

References II

Hensman, J., Fusi, N., and Lawrence, N. D. (2013).
Gaussian processes for big data.
UAI.

Jylänki, P., Vanhatalo, J., and Vehtari, A. (2011).
Robust Gaussian process regression with a student-t likelihood.
Journal of Machine Learning Research, 12(99):3227–3257.

Kuss, M. and Rasmussen, C. E. (2005).
Assessing approximate inference for binary Gaussian process classification.
Journal of Machine Learning Research, 6(57):1679–1704.

Li, R., John, S. T., and Solin, A. (2023).
Improving hyperparameter learning under approximate inference in Gaussian process models.
ICML.

Nickisch, H. and Rasmussen, C. E. (2008).
Approximations for binary Gaussian process classification.
Journal of Machine Learning Research, 9(67):2035–2078.

References III

Penny, W. (2013).
Bayesian inference course: Variational inference.

Saul, A. (2017).
Gaussian process based approaches for survival analysis.

Vehtari, A., Gelman, A., Sivula, T., Jylänki, P., Tran, D., Sahai, S., Blomstedt, P., Cunningham, J. P., Schiminovich, D., and Robert, C. P. (2020).
Expectation Propagation as a way of life: A framework for Bayesian inference on partitioned data.
Journal of Machine Learning Research, 21(17):1–53.