CSC 412/2506:
Probabilistic Learning and Reasoning

Week 11: Bayesian Regression & Gaussian Processes

Ohad Shamir

University of Toronto

Overview of the first hour

- Continuing in our theme of probabilistic models for continuous variables.
- We give a probabilistic interpretation of linear regression.
- Chapter 3.3 in Bishop's book.

Completing the Square for Gaussians

Useful technique to find moments of Gaussian random variables.

- It is a multivariate generalization of completing the square.
- The density of $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ satifies:

$$\begin{aligned}\log p(\mathbf{x}) &= -\tfrac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) + \text{const} \\ &= -\tfrac{1}{2}\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x} + \mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \text{const}\end{aligned}$$

- Thus, if we know $\mathbf{w}$ is Gaussian with *unknown* mean and covariance, and we also know that

$$\log p(\mathbf{w}) = -\tfrac{1}{2}\mathbf{w}^\top \mathbf{A}\mathbf{w} + \mathbf{w}^\top \mathbf{b} + \text{const}$$

for $\mathbf{A}$ positive definite, then we know that

$$\mathbf{w} \sim \mathcal{N}(\mathbf{A}^{-1}\mathbf{b}, \mathbf{A}^{-1}).$$

Bayesian Linear Regression

- We take the Bayesian approach to linear regression.
  - This is in contrast with the standard regression.
  - By inferring a posterior distribution over the *parameters*, the model can know what it doesn't know.
- How can uncertainty in the predictions help us?
  - Smooth out the predictions by averaging over lots of plausible explanations
  - Assign confidences to predictions
  - Make more robust decisions

Recap: Linear Regression

- Given a training set of inputs and targets $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$
- Linear model:

$$y = \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}) + \epsilon$$

  - Note: Here we employ a *feature mapping* $\boldsymbol{\psi}$ on $\mathbf{x}$
- Vectorized, we have the design matrix $\mathbf{X}$ in input space and

$$\boldsymbol{\Psi} = \begin{bmatrix} - & \boldsymbol{\psi}(\mathbf{x}^{(1)}) & - \\ - & \boldsymbol{\psi}(\mathbf{x}^{(2)}) & - \\ & \vdots & \\ - & \boldsymbol{\psi}(\mathbf{x}^{(N)}) & - \end{bmatrix}, \qquad \mathbf{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(N)} \end{bmatrix}$$

and predictions

$$\hat{\mathbf{y}} = \boldsymbol{\Psi}\mathbf{w}$$

Recap: Ridge Regression from 311

- No statistical model.
- Penalized sum of squares (ridge regression):

$$\text{minimize} \quad \frac{1}{2}\|\mathbf{y} - \mathbf{\Psi}\mathbf{w}\|^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

- The gradient: $(\mathbf{\Psi}^\top\mathbf{\Psi} + \lambda\mathbf{I})\mathbf{w} - \mathbf{\Psi}^\top\mathbf{y}$.
- Solution 1: solve analytically by setting the gradient to 0

$$\mathbf{w} = (\mathbf{\Psi}^\top\mathbf{\Psi} + \lambda\mathbf{I})^{-1}\mathbf{\Psi}^\top\mathbf{y}$$

- Solution 2: solve approximately using gradient descent

$$\mathbf{w} \leftarrow (1 - \alpha\lambda)\mathbf{w} - \alpha\mathbf{\Psi}^\top(\mathbf{\Psi}\mathbf{w} - \mathbf{y})$$

Linear Regression as Maximum Likelihood

- We can give linear regression a probabilistic interpretation by assuming a Gaussian noise model:

$$y \,|\, \mathbf{x} \sim \mathcal{N}(\mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}),\ \sigma^2)$$

- Linear regression is just maximum log-likelihood under this model:

$$\begin{aligned}
\sum_{i=1}^{N} \log p(y^{(i)} \,|\, \mathbf{x}^{(i)}; \mathbf{w}) &= \sum_{i=1}^{N} \log \mathcal{N}(y^{(i)}; \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}^{(i)}), \sigma^2) \\
&= \sum_{i=1}^{N} \log \left[\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}^{(i)}))^2}{2\sigma^2} \right) \right] \\
&= \text{const} - \frac{1}{2\sigma^2} \sum_{i=1}^{N} (y^{(i)} - \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}^{(i)}))^2 \\
&= \text{const} - \frac{1}{2\sigma^2} \|\mathbf{y} - \boldsymbol{\Psi}\mathbf{w}\|^2
\end{aligned}$$

Regularized Linear Regression as MAP Estimation

- Can get L_2 regularization via MAP inference with Gaussian prior.
- Assume $\mathbf{w} \sim \mathcal{N}(0, \eta I)$, so that

$$\log p(\mathbf{w}) = -\frac{1}{2\eta}\|\mathbf{w}\|^2$$

- Also, we just showed

$$\log p(\mathcal{D} \mid \mathbf{w}) = \text{const} - \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{\Psi}\mathbf{w}\|^2$$

- Using Bayes Rule $p(\mathbf{w} \mid \mathcal{D}) = p(\mathbf{w})p(\mathcal{D} \mid \mathbf{w})/p(\mathcal{D})$, we get that MAP inference is

$$\begin{aligned}\arg\max_{\mathbf{w}} \log p(\mathbf{w} \mid \mathcal{D}) &= \arg\max_{\mathbf{w}} \left[\log p(\mathbf{w}) + \log p(\mathcal{D} \mid \mathbf{w})\right] \\ &= \arg\min_{\mathbf{w}} \left[\frac{1}{2\eta}\|\mathbf{w}\|^2 + \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{\Psi}\mathbf{w}\|^2\right]\end{aligned}$$

This is just L_2 regularization!

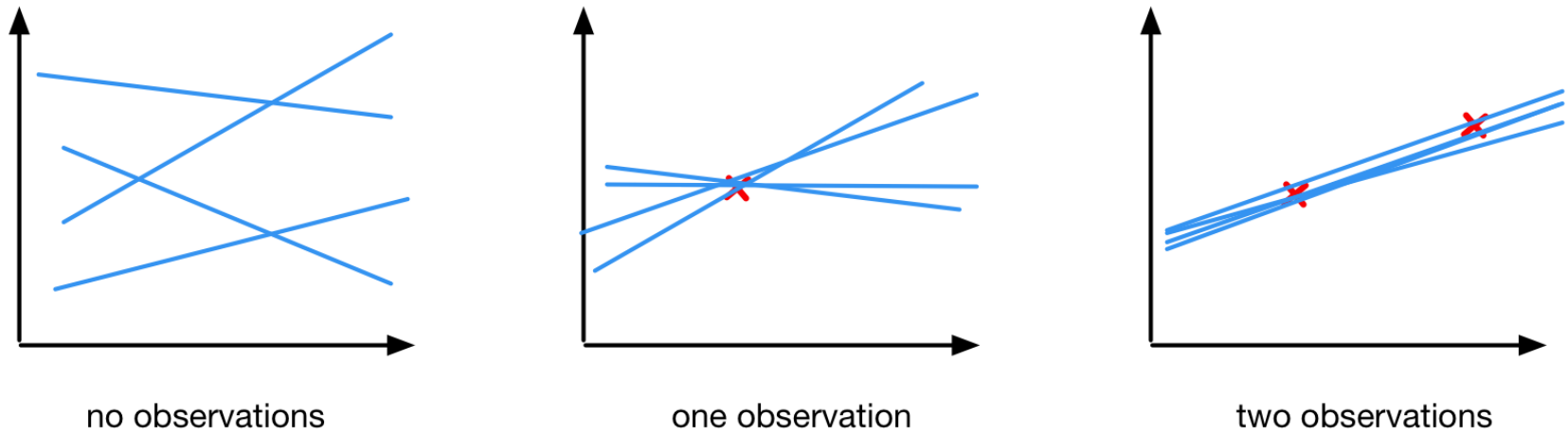
Full Bayesian Inference

- Can use MAP $\mathbf{w}$ for predictions.
- Alternatively, *full* Bayesian inference makes predictions by averaging over all likely $\mathbf{w}$ under the posterior distribution:

$$p(y \,|\, \mathbf{x}, \mathcal{D}) = \int p(\mathbf{w}, y \,|\, \mathbf{x}, \mathcal{D}) \mathrm{d}\mathbf{w}$$

$$= \int p(y \,|\, \mathbf{x}, \mathbf{w})\, p(\mathbf{w} \,|\, \mathcal{D})\, \mathrm{d}\mathbf{w}$$

- Doing this lets us quantify our uncertainty.
- Here are samples from the prior $p(\mathbf{w})$ and posteriors $p(\mathbf{w} \,|\, \mathcal{D})$



Bayesian Linear Regression: Posterior

- **Prior distribution:** $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{S})$
- **Likelihood:** $y \,|\, \mathbf{x}, \mathbf{w} \sim \mathcal{N}(\mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}),\ \sigma^2)$
  - Note: We assume fixed/known $\mathbf{S}$ and σ^2 for now
- **Deriving the posterior distribution:**

$$\begin{aligned}
\log p(\mathbf{w} \,|\, \mathcal{D}) &= \log p(\mathbf{w}) + \log p(\mathcal{D} \,|\, \mathbf{w}) + \text{const} \\
&= -\tfrac{1}{2}\mathbf{w}^\top \mathbf{S}^{-1}\mathbf{w} - \frac{1}{2\sigma^2}\|\boldsymbol{\Psi}\mathbf{w} - \mathbf{y}\|^2 + \text{const} \\
&= -\tfrac{1}{2}\mathbf{w}^\top \mathbf{S}^{-1}\mathbf{w} - \frac{1}{2\sigma^2}\left(\mathbf{w}^\top \boldsymbol{\Psi}^\top \boldsymbol{\Psi}\mathbf{w} - 2\mathbf{y}^\top \boldsymbol{\Psi}\mathbf{w} + \mathbf{y}^\top \mathbf{y}\right) + \text{const} \\
&= -\tfrac{1}{2}\mathbf{w}^\top \left(\sigma^{-2}\boldsymbol{\Psi}^\top \boldsymbol{\Psi} + \mathbf{S}^{-1}\right)\mathbf{w} + \frac{1}{\sigma^2}\mathbf{y}^\top \boldsymbol{\Psi}\mathbf{w} + \text{const} \ \text{(complete the } \square\text{!)}
\end{aligned}$$

- Thus $\mathbf{w} \,|\, \mathcal{D} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ where

$$\boldsymbol{\mu} = \left(\boldsymbol{\Psi}^\top \boldsymbol{\Psi} + \sigma^2 \mathbf{S}^{-1}\right)^{-1} \boldsymbol{\Psi}^\top \mathbf{y}$$

$$\boldsymbol{\Sigma} = \sigma^2 \left(\boldsymbol{\Psi}^\top \boldsymbol{\Psi} + \sigma^2 \mathbf{S}^{-1}\right)^{-1}$$

Bayesian Linear Regression: Posterior

- Compare $\boldsymbol{\mu}$ to the closed-form solution for linear regression:

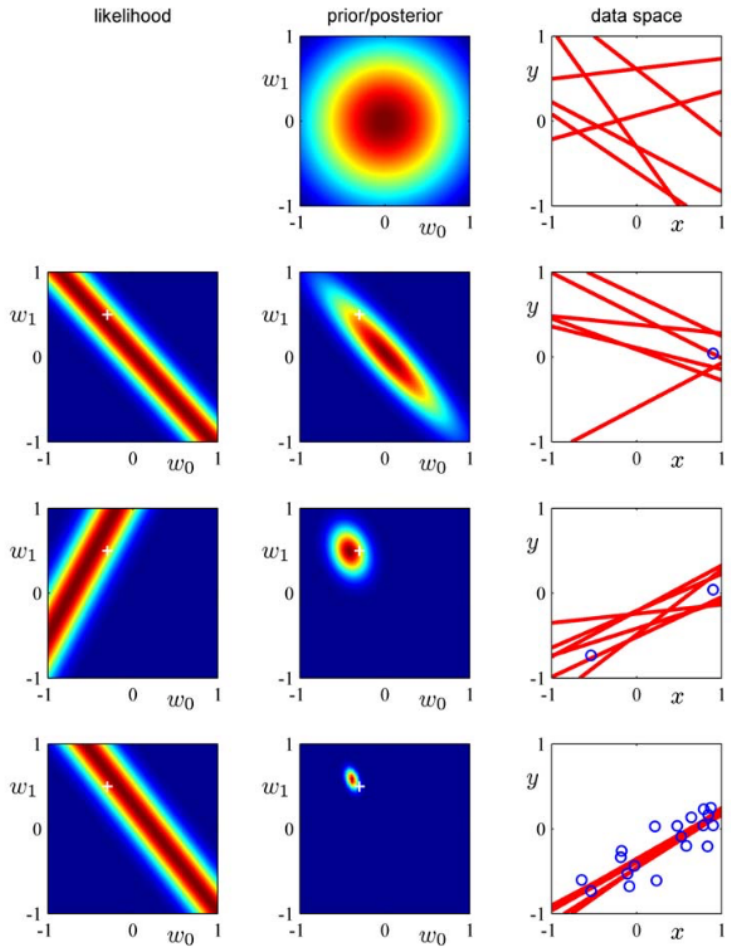
$$\mathbf{w} = (\boldsymbol{\Psi}^\top \boldsymbol{\Psi} + \lambda \mathbf{I})^{-1} \boldsymbol{\Psi}^\top \mathbf{y}$$

  This is the mean of the posterior for $\mathbf{S} = \frac{\sigma^2}{\lambda}\mathbf{I}$.

- As $\lambda \to 0$, the standard deviation of the prior goes to ∞ (uninformative prior), and the mean of the posterior converges to the solution of least squares (ridge regression with no regularization).

Bayesian Linear Regression: Posterior

Illustration of sequential Bayesian learning for $y = w_0 + w_1 x$, $w_0 = -0.3$, $w_1 = 0.5$.



Left column:

- Likelihood of a single data point.
- Single point does not identify a line.
- Fix (x, y) then $w_0 = y - w_1 x$.

Middle column:

- Prior/posterior.

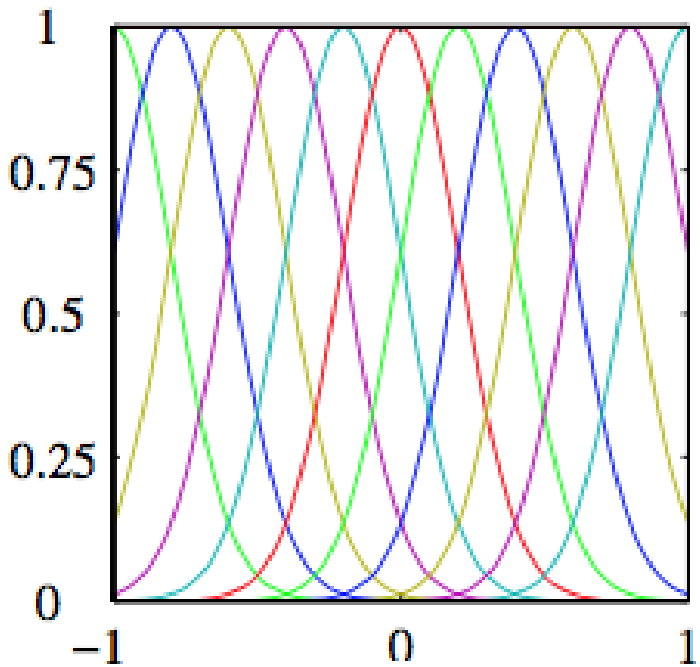
Right column:

- Lines: samples from the posterior.
- Dots: data points.

Radial bases example

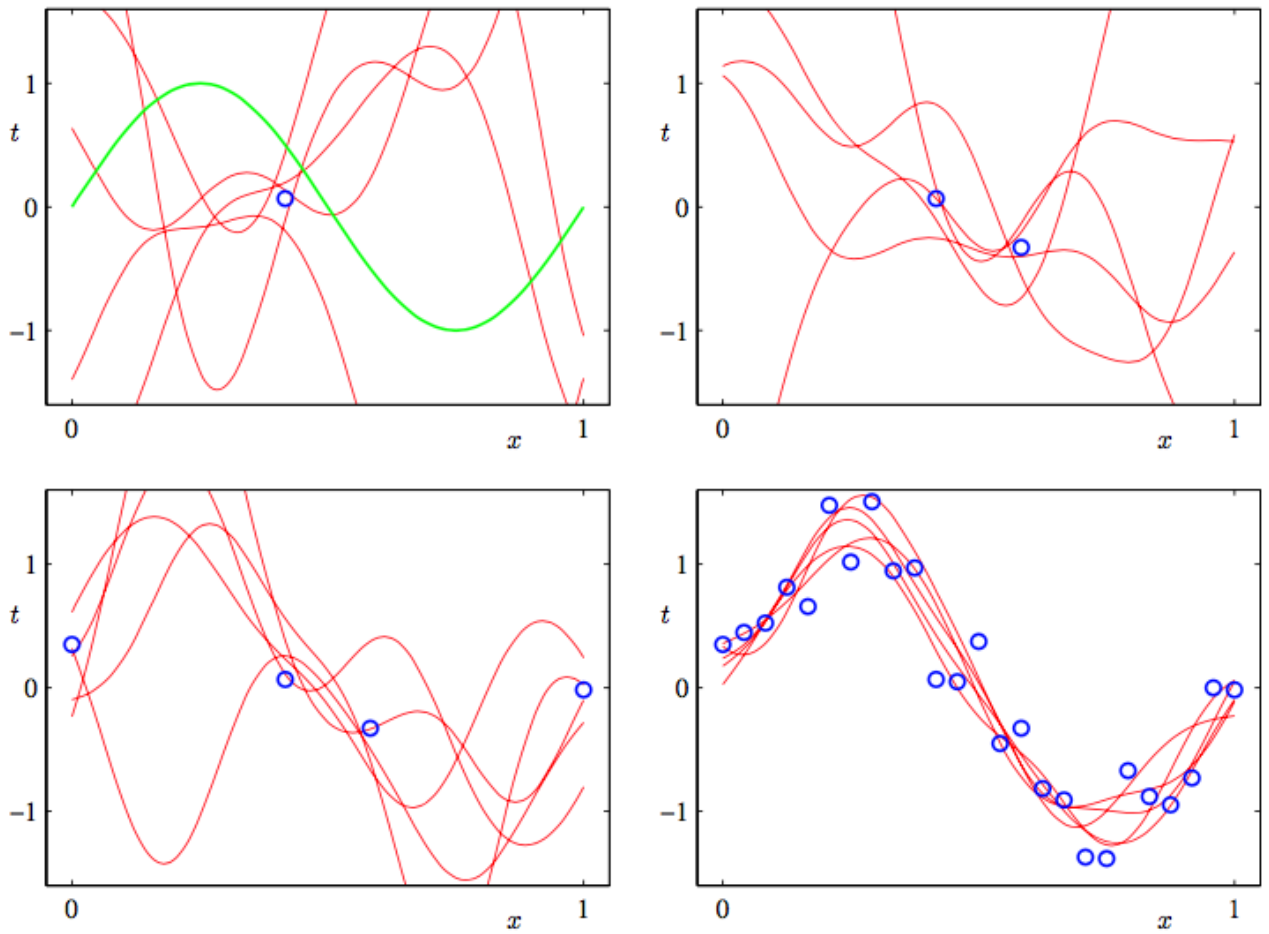
- By using a non-linear $\boldsymbol{\psi}$, the mapping $\mathbf{x} \rightarrow \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x})$ can encode non-linear functions of $\mathbf{x}$
- Example with radial basis function (RBF) features

$$\psi_j(x) = \exp\left(-\frac{(x-\mu_j)^2}{2s^2}\right)$$



Radial bases example

Functions sampled from the posterior:



Posterior predictive distribution

- The posterior just gives us distribution over the $\mathbf{w}$. To make predictions, we can use the *posterior predictive distribution*:

$$p(y \,|\, \mathbf{x}, \mathcal{D}) = \int \underbrace{p(y \,|\, \mathbf{x}, \mathbf{w})}_{\mathcal{N}(y\,;\, \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}), \sigma^2)} \underbrace{p(\mathbf{w} \,|\, \mathcal{D})}_{\mathcal{N}(\mathbf{w}\,;\, \boldsymbol{\mu}, \boldsymbol{\Sigma})} \, \mathrm{d}\mathbf{w}$$

Posterior predictive distribution

- Another interpretation: $y = \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}) + \varepsilon$, where $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ is independent of $\mathbf{w} \,|\, \mathcal{D} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

  Recall

$$\boldsymbol{\mu} = \left(\boldsymbol{\Psi}^\top \boldsymbol{\Psi} + \sigma^2 \mathbf{S}^{-1}\right)^{-1} \boldsymbol{\Psi}^\top \mathbf{y}$$

$$\boldsymbol{\Sigma} = \sigma^2 \left(\boldsymbol{\Psi}^\top \boldsymbol{\Psi} + \sigma^2 \mathbf{S}^{-1}\right)^{-1}$$

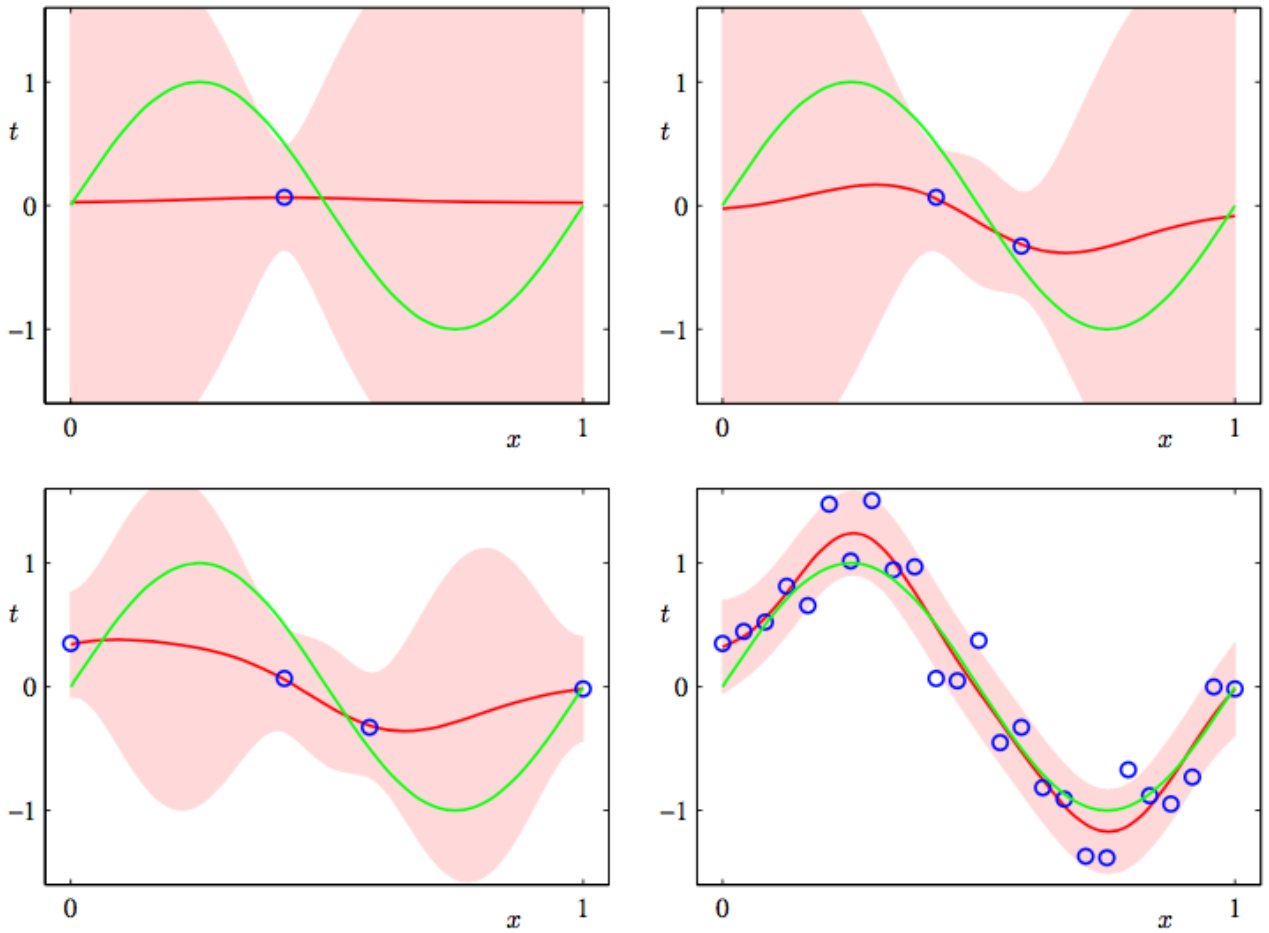
- By the fact that affine transformations of Gaussian vectors are Gaussian, y is a Gaussian distribution with parameters

$$\mu_{\text{pred}} = \boldsymbol{\mu}^\top \boldsymbol{\psi}(\mathbf{x})$$

$$\sigma^2_{\text{pred}} = \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma} \boldsymbol{\psi}(\mathbf{x}) + \sigma^2$$

Bayesian Linear Regression

Here we visualize the posterior predictive mean (red line) and variance (red region) at each possible input, given data (blue circles)



Summary

- This lecture covered the basics of Bayesian regression.

Key points:

- Posterior can be computed by completing the square.
- Posterior predictive distribution.
- Uncertainty quantification.

From Bayesian Regression to Gaussian Processes

- Consider Bayesian linear regression with isotropic Gaussian prior:

$$y(\mathbf{x}) = \hat{y}(\mathbf{x}) + \mathcal{N}(0, \sigma^2), \qquad \hat{y}(\mathbf{x}) = \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}), \qquad \mathbf{w} \sim \mathcal{N}(0, \tfrac{1}{\alpha}\mathbf{I}).$$

- Prior induces a covariance between function values:

$$\begin{aligned}\mathrm{Cov}(\hat{y}(\mathbf{x}), \hat{y}(\mathbf{x}')) &= \mathbb{E}_{\mathbf{w}}[\hat{y}(\mathbf{x})\hat{y}(\mathbf{x}')] \\ &= \boldsymbol{\psi}(\mathbf{x})^\top \mathbb{E}_{\mathbf{w}}[\mathbf{w}\mathbf{w}^\top]\boldsymbol{\psi}(\mathbf{x}') \\ &= \tfrac{1}{\alpha}\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\psi}(\mathbf{x}').\end{aligned}$$

- Write $k(\mathbf{x}, \mathbf{x}') = \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\psi}(\mathbf{x}')$, denoted as the kernel function. For any inputs $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$,

$$\hat{\mathbf{y}} \sim \mathcal{N}(0, \boldsymbol{K}), \qquad K_{ij} = \tfrac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}).$$

- Can we work with a covariance directly without choosing a feature map $\boldsymbol{\psi}$? This motivates Gaussian processes.

Gaussian process (GP)

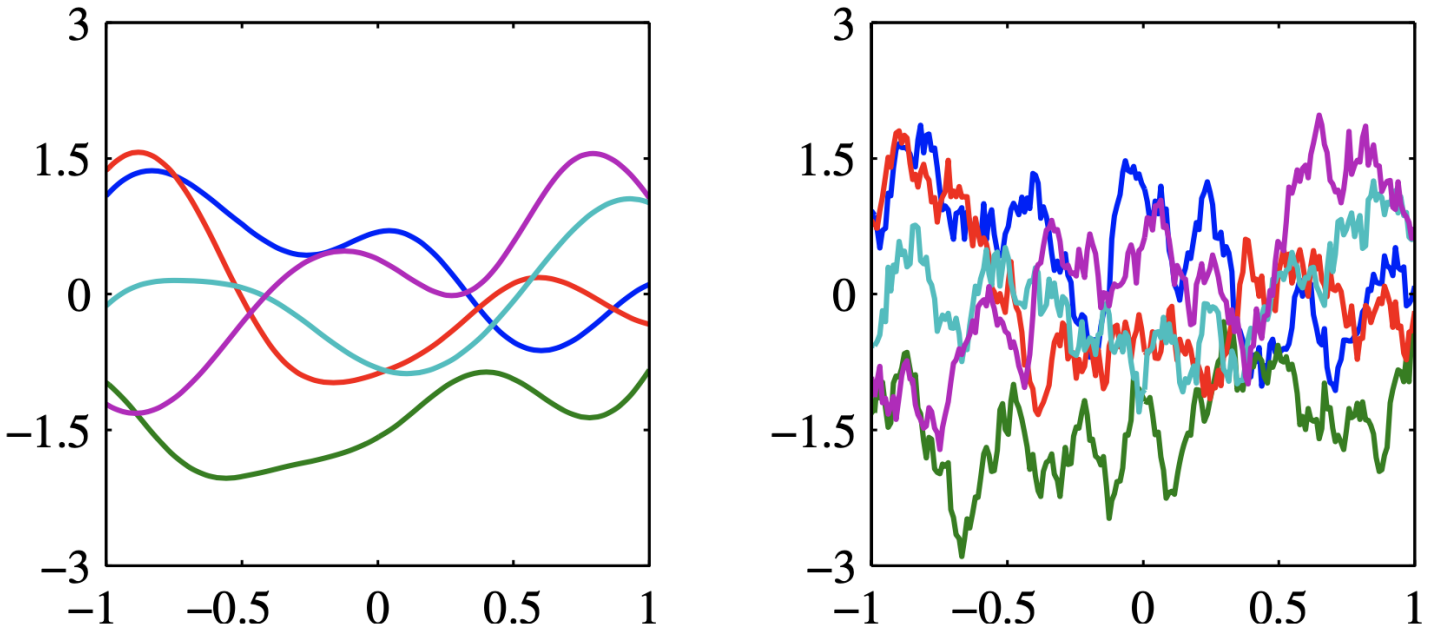
- **Definition:** A **Gaussian process** is a probability distribution over *functions* $\hat{y}(\mathbf{x})$, such that for any $M \geq 1$ and any set of M points $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(M)}$ in $\mathbb{R}^D$, the vector $(\hat{y}(\mathbf{x}^{(1)}), \ldots, \hat{y}(\mathbf{x}^{(M)}))$ is jointly Gaussian.

- The joint distribution is specified completely by the second-order statistics, i.e. the mean and the covariance functions.
- In most applications, the mean function of $\hat{y}(\mathbf{x})$ can be set to zero (e.g., because data is centered), and then the Gaussian process is completely specified by its covariance function

$$\mathbb{E}[\hat{y}(\mathbf{x})\hat{y}(\mathbf{x}')] = \frac{1}{\alpha}k(\mathbf{x}, \mathbf{x}')$$

Gaussian process (GP)

- We can specify the GP kernel function directly, without explicitly constructing a feature map.
  - E.g. $k(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|^2/\ell)$ (Gaussian kernel), $k(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|/\ell)$ (exponential kernel)
  - Technical condition on $k()$: Induced covariance matrix on any finite set of inputs must be positive semidefinite.



Samples from Gaussian processes for a Gaussian kernel (left) and an exponential kernel (right).

Gaussian processes for regression

- As before, use a Gaussian observation model

$$y \,|\, \mathbf{x} \sim \mathcal{N}(\hat{y}(\mathbf{x}), \sigma^2),$$

- For fixed inputs $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$, we saw that vector of values satisfy

$$\hat{\mathbf{y}} \sim \mathcal{N}(0, \boldsymbol{K}), \qquad \mathbf{y} \,|\, \hat{\mathbf{y}} \sim \mathcal{N}(\hat{\mathbf{y}}, \sigma^2 \mathbf{I}),$$

and hence

$$\mathbf{y} \sim \mathcal{N}(0, \boldsymbol{C}), \qquad \boldsymbol{C} = \boldsymbol{K} + \sigma^2 \mathbf{I},$$

where we recall $K_{i,j} = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$

Gaussian processes for regression

- Suppose we got observations $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$ and associated vector of values $\mathbf{y}_N = (y^{(1)}, \ldots, y^{(N)})$. How do we predict value $y^{(N+1)}$ of some new point $\mathbf{x}^{(N+1)}$?
- Idea: Consider joint distribution of value vector $(\mathbf{y}_N, y^{(N+1)})$
- By previous slide,

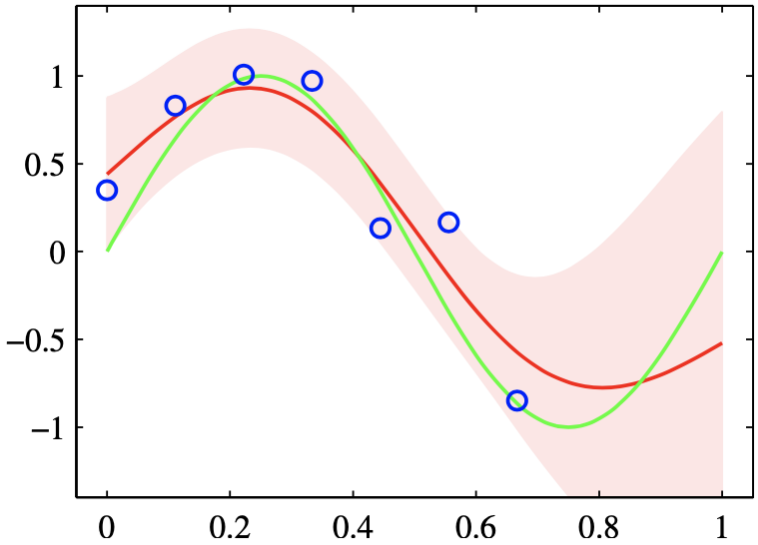
$$\begin{bmatrix} \mathbf{y}_N \\ y^{(N+1)} \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \boldsymbol{C} & \mathbf{k} \\ \mathbf{k}^\top & c \end{bmatrix}\right).$$

  where vector $\mathbf{k}$ satisfies $k_i = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(N+1)})$, and $c = \frac{1}{\alpha} k(\mathbf{x}^{(N+1)}, \mathbf{x}^{(N+1)}) + \sigma^2$.
- By Gaussian conditioning formulas, can show this implies that

$$y^{(N+1)} \,|\, \mathbf{y}_N \sim \mathcal{N}\left(\mathbf{k}^\top \boldsymbol{C}^{-1} \mathbf{y}_N,\ c - \mathbf{k}^\top \boldsymbol{C}^{-1} \mathbf{k}\right).$$

  This is the key result for GP regression.
- Can show that similar to Bayesian linear regression, but defined only via $k()$: No need to explicitly define/use feature map $\boldsymbol{\psi}$.

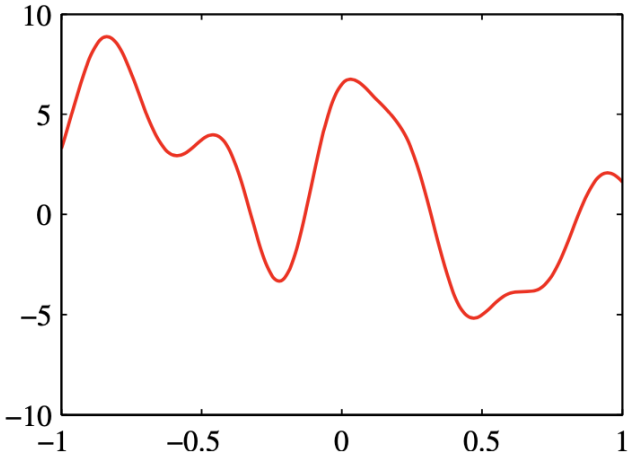
GPs for regression



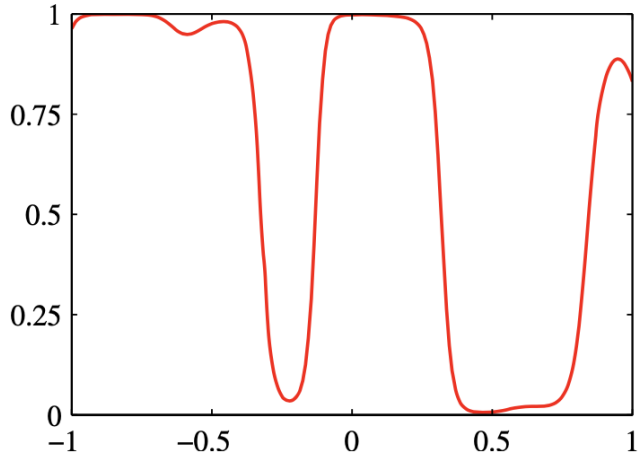
- The green curve is the true sinusoidal function from which the data points, shown in blue, are obtained.
- The red line shows the **mean** of the Gaussian process predictive distribution.
- The shaded region corresponds to plus and minus two standard deviations.

GPs for classification

- Consider a classification problem with target variables $y \in \{0, 1\}$
- We define a Gaussian process over a function $a(\mathbf{x})$ and then transform this to binary labels using a sigmoid $\hat{y}(\mathbf{x}) = \sigma(a(\mathbf{x}))$.
- We obtain a non-Gaussian stochastic process over functions $\hat{y}(\mathbf{x}) \in (0, 1)$.



$a(\mathbf{x})$



$\hat{y}(\mathbf{x})$

GPs for classification

- The probability distribution given $a(\mathbf{x})$ can be written as

$$p(y|a) = \sigma(a)^y(1-\sigma(a))^{1-y}, \quad y \in \{0,1\}.$$

- We need to compute

$$p(y^{(N+1)} \,|\, \mathbf{y}_N)$$

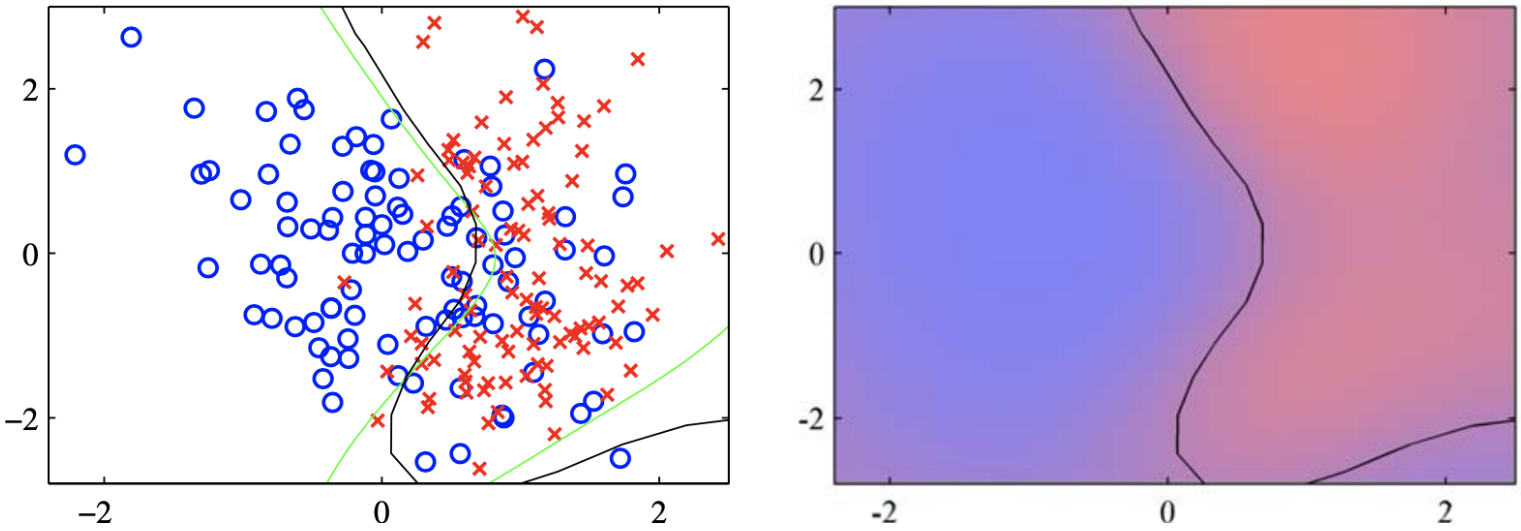
  and notice that $a(\mathbf{x})$ is a Gaussian process but $\hat{y}(\mathbf{x})$ is not.
- Similar to before, we have $\boldsymbol{a}_{N+1} \sim \mathcal{N}(0, \boldsymbol{K}_{N+1})$ for appropriate matrix $\boldsymbol{K}_{N+1}$.
- However, $\boldsymbol{a}_{N+1}$ is not observed, so we write

$$p(y^{(N+1)} \,|\, \mathbf{y}_N) = \int p(y^{(N+1)} \,|\, \boldsymbol{a}_{N+1}) p(\boldsymbol{a}_{N+1} \,|\, \mathbf{y}_N) d\boldsymbol{a}_{N+1}$$

- This is intractable. We need MCMC based methods, or numerical integration to approximate this integral.

GPs for classification: Illustration

- Illustration of GPs for classification:



- Left: optimal decision boundary from the true distribution in green, and the decision boundary from the Gaussian process classifier in black.
- Right: predicted posterior for the blue and red classes together with the Gaussian process decision boundary.

Learning the hyperparameters

- So far, we have assumed a fixed kernel function $k()$.
- Instead, we may prefer to use a parametric family of functions $k_\theta()$ (parameterized by hyperparameters θ), and then infer the θ from the data.
- Let $\boldsymbol{C}_\theta$ denote the covariance matrix induced by $k_\theta()$ on dataset. We can simply maximize data likelihood over θ
- E.g. for GP regression with model $\mathbf{y} \sim \mathcal{N}(0, \boldsymbol{C}_\theta)$,

$$\log p(\mathbf{y} \,|\, \theta) = -\frac{1}{2} \log |\boldsymbol{C}_\theta| - \frac{1}{2} \mathbf{y}^\top \boldsymbol{C}_\theta^{-1} \mathbf{y} + \text{const.}$$

- Maximize over θ using your favorite approach (gradient methods, grid search,...)

Summary of the second hour

- Gaussian processes are flexible tools that can be used in regression and classification tasks.
- One can simply choose a kernel and find the predictive distribution!
  - For regression with Gaussian noise: In closed form. For classification: Need approximate inference.
- Hyperparameters can be learned from data.