Module 3: Bayesian Nonparametrics

Gaussian Processes for Regression

STAT/BIOSTAT 527, University of Washington

Emily Fox

April 21st-22nd, 2014

Recap of regression so far

- Recall our regression setting

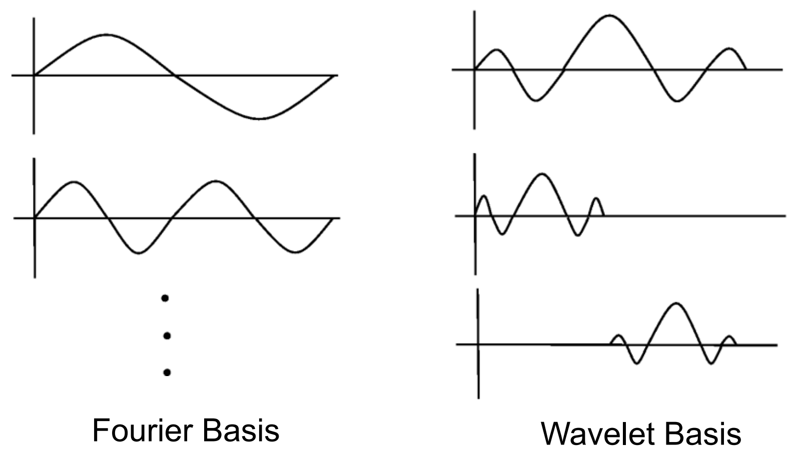
$$f(x) = E[Y \mid x]$$

- How to estimate from finite training set?

Restrict to model class

- Example = linear basis expansion
  - Standard linear
  - Polynomial
  - Splines
  - …

Other Important Basis Expansions



Recap of regression so far

- Recall our regression setting

$$f(x) = E[Y \mid x]$$

- How to estimate from finite training set?

Restrict to model class

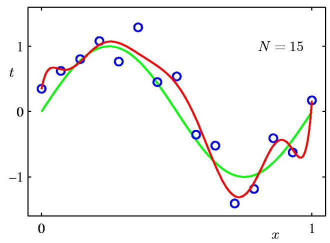
- Example = linear basis expansion

Overfitting as model complexity grows

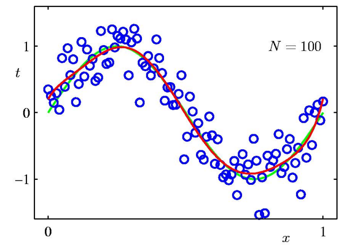
- Penalized linear basis expansions
  - Ridge
  - Lasso
  - Smoothing splines
  - Penalized regression splines

Overfitting

9th Order Polynomial



$n = 15$



$n = 100$

Recap of regression so far

- Recall our regression setting

$$f(x) = E[Y \mid x]$$

- How to estimate from finite training set?

Restrict to model class

Local nbhd methods

- Example = linear basis expansion

Overfitting as model complexity grows

- Penalized linear basis expansions

- Example = kernel regression

Again: Linear Basis Expansion

- Instead of just considering input variables *x* (potentially mult.), augment/replace with transformations = "input features"
- ***Linear basis expansions*** maintain linear form in terms of these transformations

trans.

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x)$$

- What transformations should we use?
  - $h_m(x) = x_m$ → linear model
  - $h_m(x) = x_j^2, \quad h_m(x) = x_j x_k$ → polynomial reg.
  - $h_m(x) = I(L_m \le x_k \le U_m)$ → piecewise constant
  - …

Making Predictions

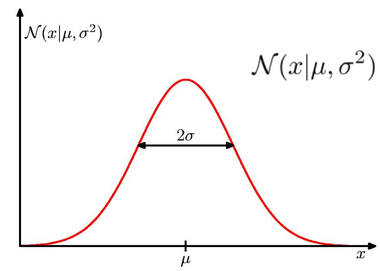
- So far, our focus has been on L_2 loss:

$$\min_{\beta} \ \mathrm{RSS}(\beta) + \lambda ||\beta||$$

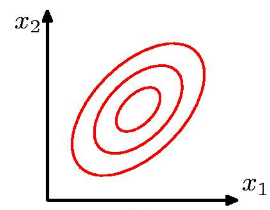
- Here, we assumed $y = f(x) + \epsilon$ with
- Now, let's assume a distributional form and log-likelihood loss

Quick Review of Gaussians

- Univariate and multivariate Gaussians

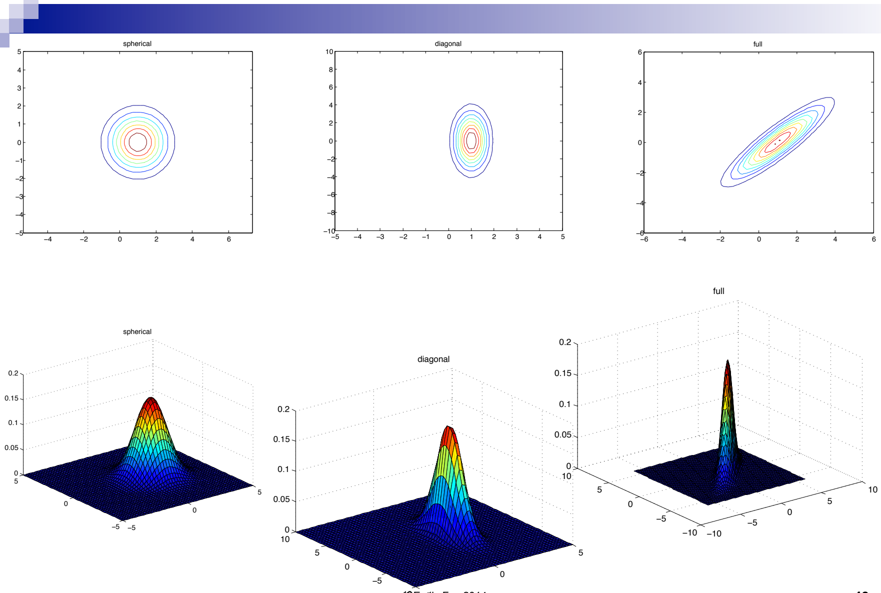


$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$

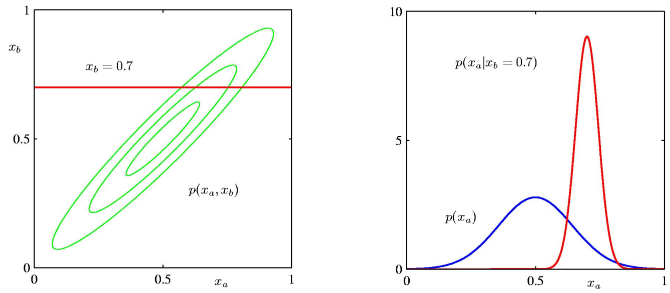


$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

Two-Dimensional Gaussians



Conditional & Marginal Distributions



Maximum Likelihood Estimation

- Model:

$$y = f(x) + \epsilon \quad \text{where} \quad \epsilon \sim N(0, \sigma^2)$$

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x)$$

- Equivalently,

$$p(y \mid x, \beta, \sigma^2) = N(y \mid f(x), \sigma^2)$$

- For our training data (independent obs)

$$p(y \mid X, \beta, \sigma^2) =$$

Maximum Likelihood Estimation

$$p(y \mid X, \beta, \sigma^2) = \prod_i N(y_i \mid \beta^T h(x_i), \sigma^2)$$

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

- Taking the log

- Equivalent objective to RSS *(Gaussian log-like loss = L_2 loss)*

- Taking the gradient and setting to zero, we have already shown

$$\hat{\beta}^{ML} = (H^T H)^{-1} H^T y$$

A Bayesian Formulation

- Consider a model with likelihood

if $\epsilon \sim N(0, \sigma^2)$

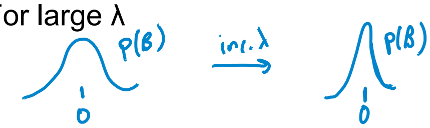
$$y_i \mid \beta \sim N(\underline{\beta_0} + x_i^T \underline{\beta}, \sigma^2)$$

and prior

$$\beta \sim N\left(0, \frac{\sigma^2}{\lambda} I_p\right)$$

$\beta_j \sim N(0, \frac{\sigma^2}{\lambda})$

- For large λ



prior peaked around $\beta = 0$
$\Leftrightarrow$ penalizing β far from 0

- The posterior is

$$\beta \mid y \sim N\left(\hat{\beta}^{ridge}, \sigma^2 (X^T X + \lambda I)^{-1} X^T X \sigma^2 (X^T X + \lambda I)^{-1}\right)$$

$\boxed{\hat{\beta}^{MAP} = \hat{\beta}^{ridge}}$

easy to show $var(\hat{\beta}^{ridge})$

Bayesian Linear Regression

- More generally, consider a conjugate prior on the basis expansion coefficients:

$$p(\beta) = N(\beta \mid \mu_0, \Sigma_0)$$

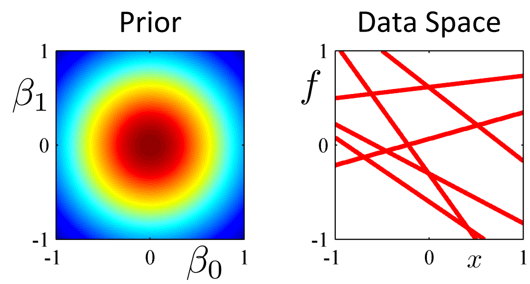
- Combining this with the Gaussian likelihood function, and using standard Gaussian identities, gives posterior

$$p(\beta \mid y) = N(\beta \mid \mu_n, \Sigma_n)$$

where

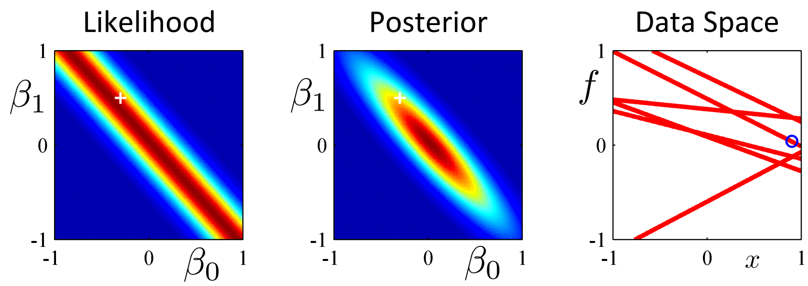
Example: Standard Linear Basis

0 data points observed



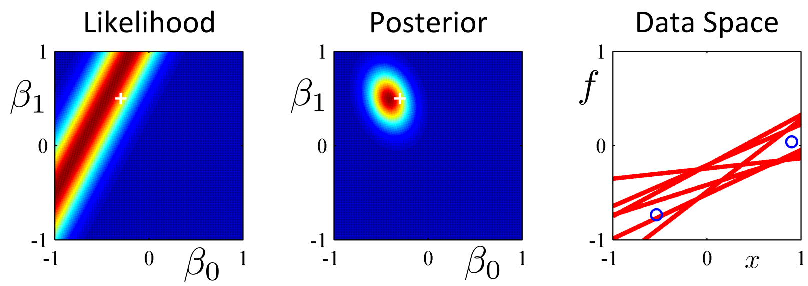
Example: Standard Linear Basis

1 data point observed



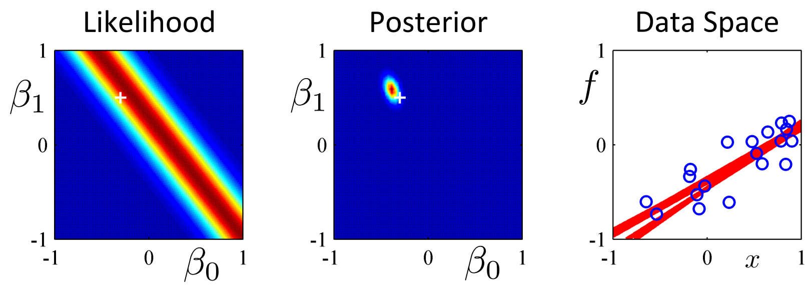
Example: Standard Linear Basis

2 data points observed



Example: Standard Linear Basis

20 data points observed



Predictive Distribution

- Predict y^* at new locations x^* by integrating over parameters β

$$p(y^* \mid y) = \int p(y^* \mid \beta) p(\beta \mid y) d\beta$$

$$p(\beta \mid y) = N(\beta \mid \mu_n, \Sigma_n)$$

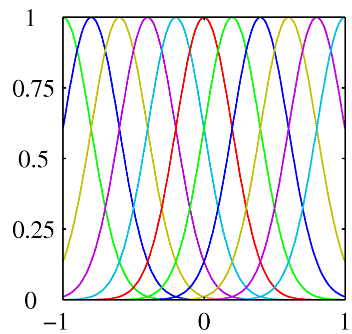
$$p(y \mid x, \beta, \sigma^2) = N(y \mid f(x), \sigma^2)$$

Example: Gaussian Basis Expansion

- Gaussian basis functions:

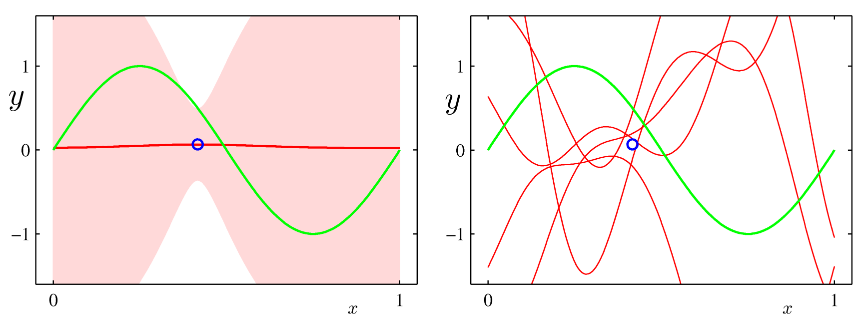
$$h_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$

- These are local;
  a small change in x
  only affects nearby
  basis functions.
  Parameters control
  location and scale (width)



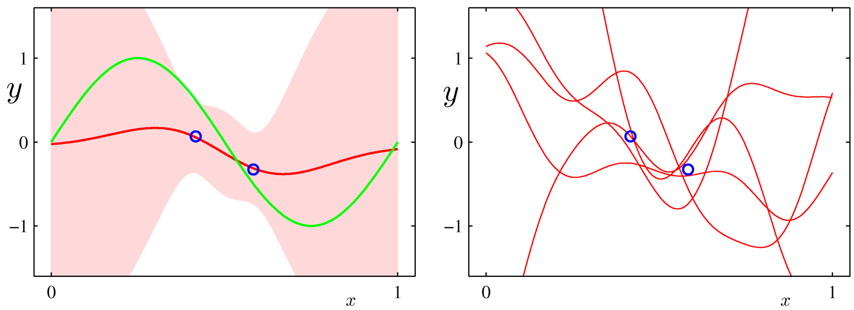
Example: Gaussian Basis Expansion

- Example: Sinusoidal data, 9 Gaussian basis functions,
  1 data point



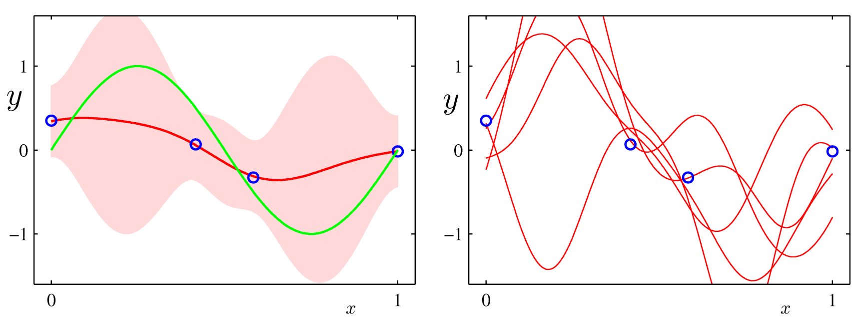
Example: Gaussian Basis Expansion

- Example: Sinusoidal data, 9 Gaussian basis functions, 2 data points



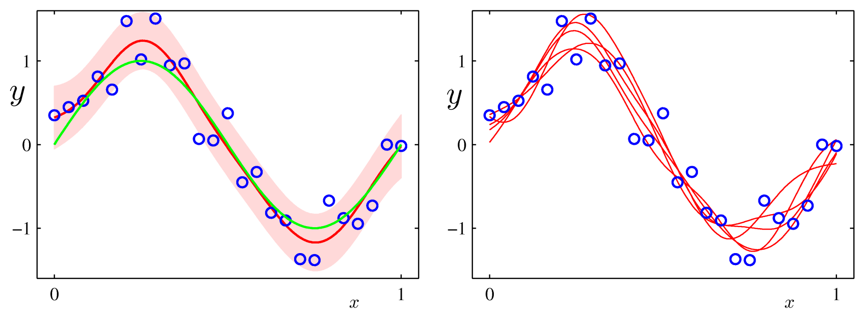
Example: Gaussian Basis Expansion

- Example: Sinusoidal data, 9 Gaussian basis functions, 4 data points

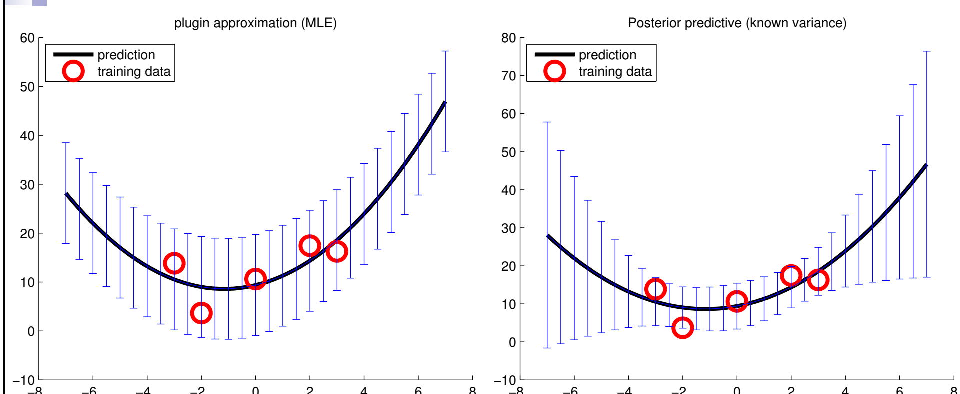


Example: Gaussian Basis Expansion

- Example: Sinusoidal data, 9 Gaussian basis functions, 25 data points



Estimation vs. Predictive Distributions



Bayesian Model Selection

- Assume some M possible models
  - Model M_m $m=1,\ldots,M$ has parameters θ_m and prior $p(\theta_m \mid M_m)$
  - Prior over models $p(M_m)$

- Model posterior

training data

$$p(M_m \mid Z) \propto p(M_m)p(Z \mid M_m)$$

$$\propto p(M_m) \int p(Z \mid \theta_m, M_m)p(\theta_m \mid M_m)d\theta_m$$

eg mean + cov of Gauss or $\{\beta_j\}$

- Compare models:

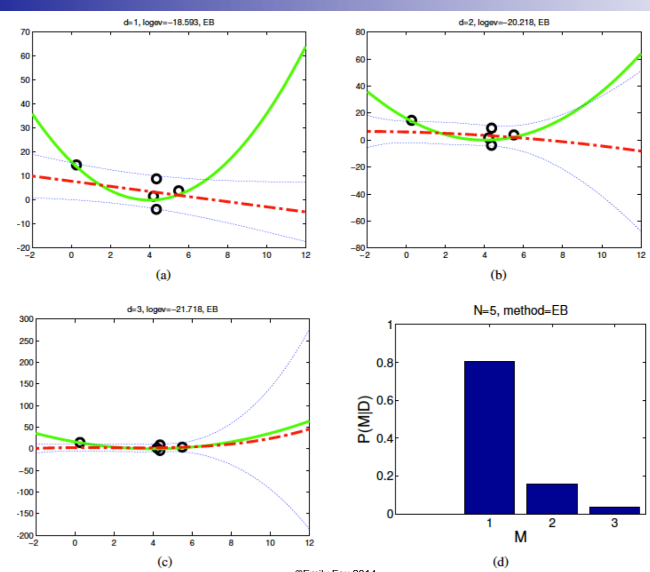
posterior odds

$$\frac{p(M_m \mid Z)}{p(M_\ell \mid Z)} = \frac{p(M_m)}{p(M_\ell)} \boxed{\frac{p(Z \mid M_m)}{p(Z \mid M_\ell)}} \gtrless 1$$

often, uniform prior

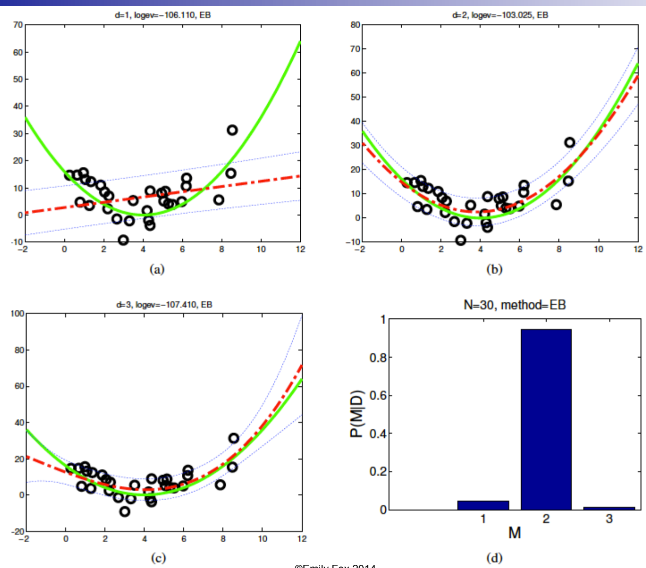
Bayes factor

BMS Example (n=5)

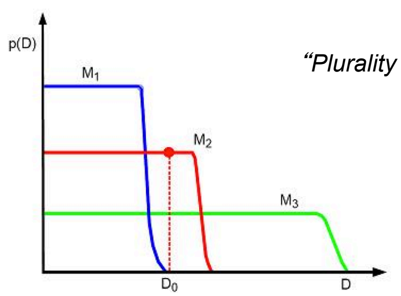


(a) (b) (c) (d)

BMS Example (n=30)



Bayesian Ockham's Razor



"Plurality must never be posited without necessity."

William of Ockham

- **Parametric Bayes:** Consider a finite list of possible models, average according to posterior probability (or in practice, just select the most probable)
- **Nonparametric Bayes:** Consider a single infinite model, integrate over parameters when making predictions or infer which finite subset is exhibited in your dataset

Going Infinite…

Change of notation:
$h(x) \to \phi(x)$

- Nonparametric Gaussian regression:
  Would like to let the number of basis functions $M \to \infty$
- *Prior:* $p(\beta \mid 0, \alpha^{-1} I_M)$
- *Distribution on f:* $f = \Phi\beta$
- Gaussian process models replace explicit basis function representation with a direct specification in terms of a ***positive definite kernel function***

Mercer Kernel Functions

- Distributions are of the form

  $$p(f) = N(f \mid 0, \alpha^{-1}\Phi\Phi^T)$$

  where the ***Gram matrix*** *K* is defined as

  $$K_{ij} =$$

- *K* is a ***Mercer kernel*** if the Gram matrix is positive definite for any *n* and any $x_1, \ldots, x_n$

Mercer's Theorem

- If *K* is positive definite, we can compute the eigendecomp:
- Then $K_{ij} =$
- Define $\phi(x) = \Lambda^{\frac{1}{2}} U_{\cdot i}$ so that

  $K_{ij} =$
- If a kernel is Mercer, there exists a function $\phi : \mathcal{X} \to \mathbb{R}^d$ s.t.

Example Mercer Kernels

- Example #1: (non-stationary) ***polynomial kernel***

$$\kappa(x, x') = (\gamma x^T x' + r)^M$$

- For *M*=2, *γ* = *r* = 1,

$$(1 + x^T x')^2 = (1 + x_1 x_1' + x_2 x_2')^2$$

- This can be written as $\phi(x)^T \phi(x')$, with

  $\phi(x) =$
  - Equivalent to working in a 6-dimensional feature space
  - For general *M,* basis contains all terms up to degree *M*
- Example #2: ***Gaussian kernel***

$$\kappa(x, x') = \exp\left(-\frac{1}{2}(x - x')^T \Sigma^{-1}(x - x')\right)$$

  - Feature map lives in an infinite-dimensional space

Gaussian Processes

- Dispense of parametric view (prior on β) and consider prior on functions themselves (prior on *f)*
- Seems hard, but we have shown that it is feasible when we look at a finite set of values $x_1, \ldots, x_n$

$$p(f) = N(f \mid 0, K)$$

- Defined by a *Mercer kernel*
- More generally, a ***Gaussian process*** provides a distribution over functions

Gaussian Processes

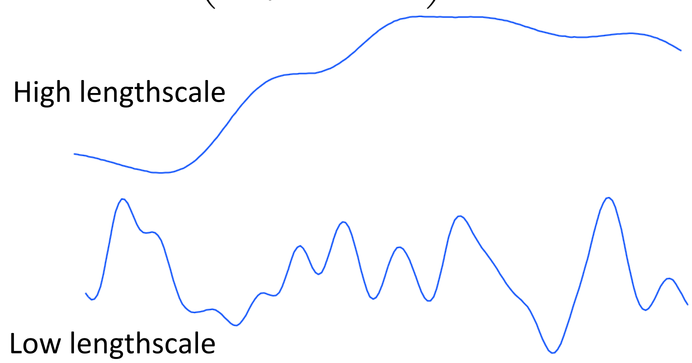
- Distribution on functions
  - $f \sim \mathrm{GP}(m,\kappa)$
    - m: mean function
    - κ: covariance function

$$\Updownarrow$$

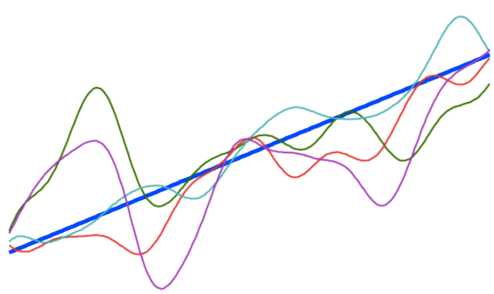
  - $p(f(x_1), \ldots, f(x_n)) \sim N_n(\mu, K)$
    - $\mu = [m(x_1),\ldots,m(x_n)]$
    - $K_{ij} = \kappa(x_i,x_j)$
- Idea: If x_i, x_j are similar according to the kernel, then $f(x_i)$ is similar to $f(x_j)$

κ: covariance function

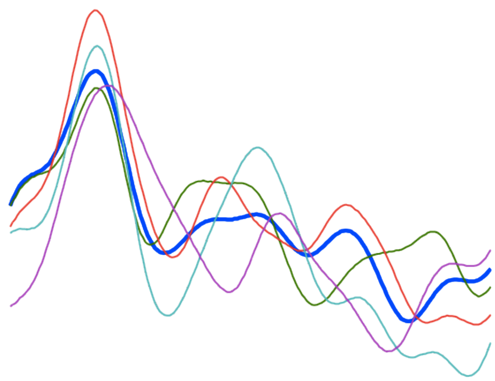
$$\kappa(x, x') = \sigma_f^2 \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$



m: mean function

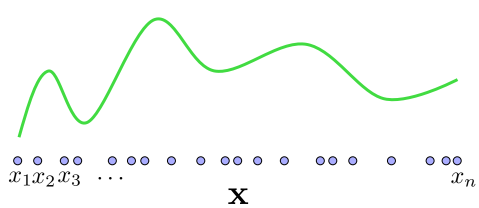


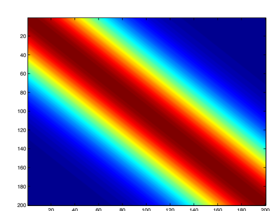
m: mean function



Induced Multivariate Gaussian

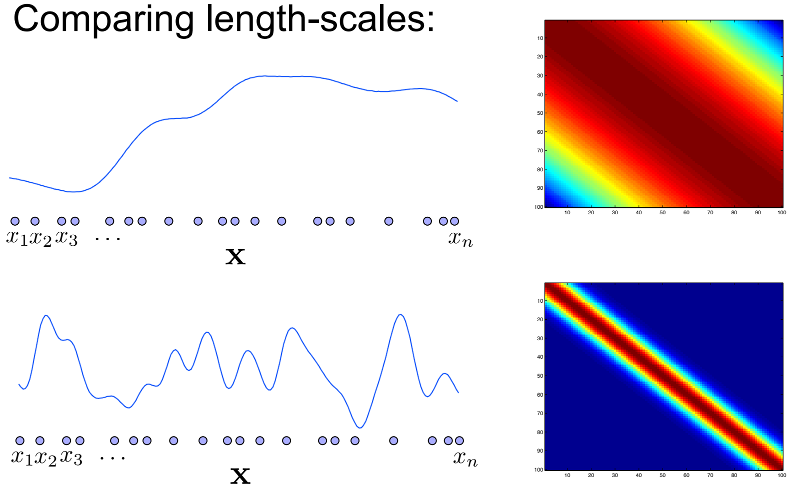
- Evaluating the GP-distributed function at any set of locations, we have





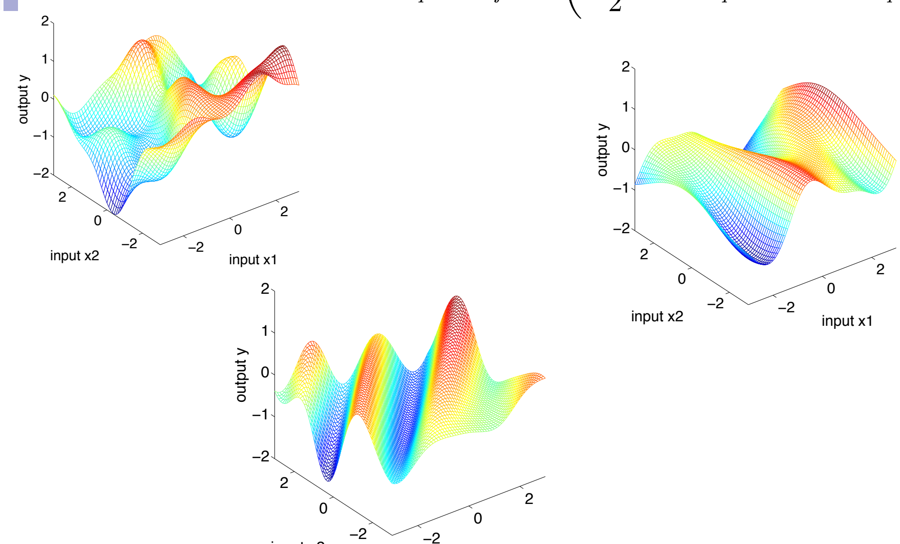
Induced Multivariate Gaussian

- Comparing length-scales:



2D Gaussian Processes

$$\kappa(x_p, x_q') = \sigma_f^2 \exp\left(-\frac{1}{2}(x_p - x_q')^T M (x_p - x_q')\right)$$



GPs for Regression

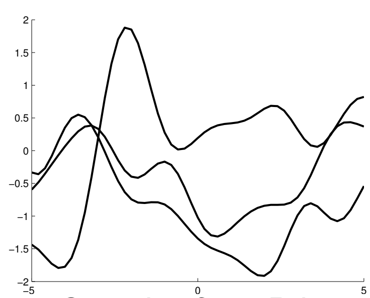
- Start with noise-free scenario: directly observe the function
- Training data $\mathcal{D} = \{(x_i, f_i), i = 1, \ldots, n\}$
- Test data locations X^* → predict *f**
- Jointly, we have

$$\begin{pmatrix} f \\ f^* \end{pmatrix} \sim N\left(\begin{pmatrix} \mu \\ \mu_* \end{pmatrix}, \begin{pmatrix} K & K_* \\ K_*^T & K_{**} \end{pmatrix} \right)$$

- Therefore,

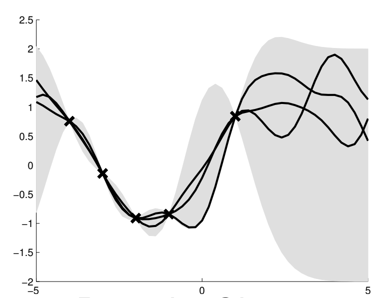
$$p(f^* \mid X^*, X, f) =$$

1D Noise-Free Example



Samples from Prior

$$\kappa(x, x') = \sigma_f^2 \exp(-\frac{1}{2\ell^2}(x - x')^2)$$



Posterior Given 5 Noise-Free Observations

- Interpolator, where uncertainty increases with distance
- Useful as a computationally cheap proxy for a complex simulator
    - Examine effect of simulator params on GP predictions instead of doing expensive runs of the simulator

GPs for Regression

- Noisy scenario: observe a noisy version of underlying function

$$y = f(x) + \epsilon \quad \epsilon \sim N(0, \sigma_y^2)$$

  - Not required to interpolate, just come "close" to observed data

$$\text{cov}(y|X) =$$

- Training data $\mathcal{D} = \{(x_i, y_i), i = 1, \ldots, n\}$
- Test data locations X^* → predict *f**
- Jointly, we have $\begin{pmatrix} y \\ f^* \end{pmatrix} \sim N\left(0, \begin{pmatrix} K_y & K_* \\ K_*^T & K_{**} \end{pmatrix}\right)$
- Therefore, $p(f^* \mid X^*, X, y) =$

GPs for Regression

$$p(f^* \mid X^*, X, y) = N(K_*^T K_y^{-1} y, K_{**} - K_*^T K_y^{-1} K_*)$$

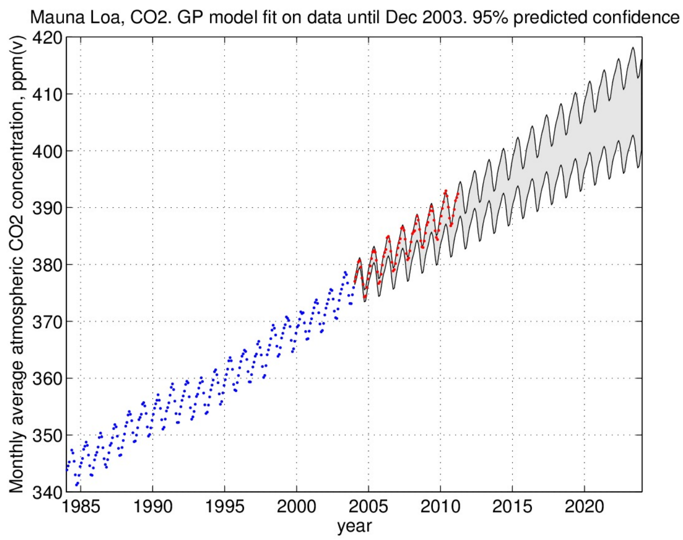
- For a single point *x**

$$p(f^* \mid X^*, X, y) = N(k_*^T K_y^{-1} y, k_{**} - k_*^T K_y^{-1} k_*)$$

so

$$\bar{f}^* = k_*^T K_y^{-1} y =$$

CO2 Concentration Over Time



Mauna Loa Observatory in Hawaii, analyzed by Rasmussen & Williams 2006

Mixing Kernels for CO2 GP Analysis

Smooth global trend

$$\kappa_1(x, x') = \theta_1^2 \exp\left(-\frac{(x-x')^2}{2\theta_2^2}\right)$$

Seasonal periodicity

$$\kappa_2(x, x') = \theta_3^2 \exp\left(-\frac{(x-x')^2}{2\theta_4^2} - \frac{2\sin^2(\pi(x-x'))}{\theta_5^2}\right)$$

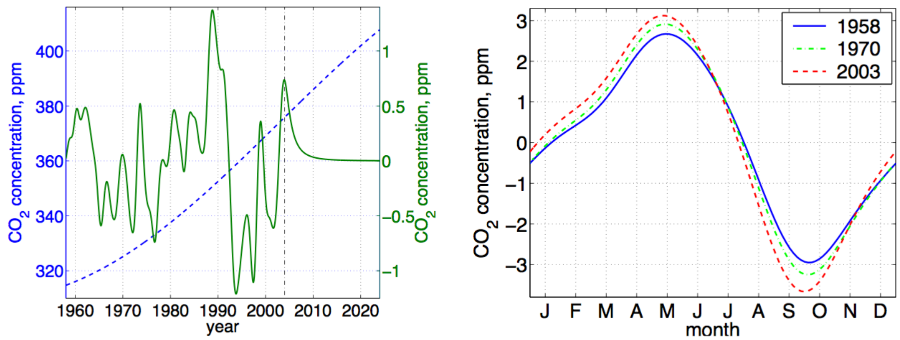
Medium term irregularities

$$\kappa_3(x, x') = \theta_6^2 \left(1 + \frac{(x-x')^2}{2\theta_8\theta_7^2}\right)^{-\theta_8}$$

Correlated Observation Noise

$$\kappa_4(x_p, x_q) = \theta_9^2 \exp\left(-\frac{(x_p-x_q)^2}{2\theta_{10}^2}\right) + \theta_{11}^2\delta_{pq}$$

CO2 Concentration Over Time



(a) (b)

Mauna Loa Observatory in Hawaii, analyzed by Rasmussen & Williams 2006

Estimating Hyperparameters

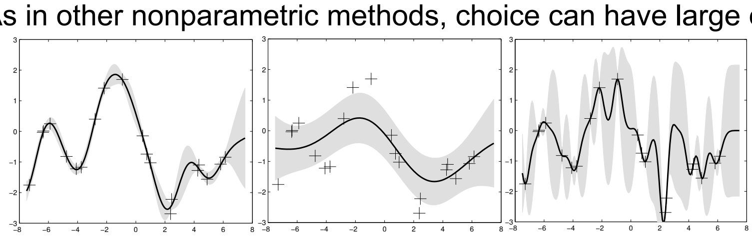
- How should we choose the kernel parameters?
  - Example: squared exponential kernel parameterization

$$\kappa(x, x') = \sigma_f^2 \exp\left(\frac{-1}{2}(x_p - x_q)^T M (x'_p - x'_q)\right) + \sigma_y^2 \delta_{pq}$$

  - Hyperparameters
  - As we saw before, can choose

$$M = \ell^{-2} I \quad M = \mathrm{diag}(\ell_1^{-2}, \ldots, \ell_d^{-2}) \quad M = \Lambda\Lambda' + \mathrm{diag}(\ell_1^{-2}, \ldots, \ell_d^{-2}) \ldots$$

- As in other nonparametric methods, choice can have large effect



Estimating Hyperparameters

- Options:
  - #1: Define a grid of possible values and use cross validation
  - #2: Full Bayesian analysis: Place prior on hyperparameters and integrate over these as well in making predictions
  - #3: Maximize the marginal likelihood

$$p(y \mid X, \theta) = \int p(y \mid f, X) p(f \mid X, \theta) df$$

$$\log p(y \mid X, \theta) =$$

Estimating Hyperparameters

$$\log p(y \mid X, \theta) = -\frac{1}{2} y^T K_y^{-1} y - \frac{1}{2} \log |K_y| - \frac{n}{2} \log 2\pi$$

  - For short length-scale, the fit is good, but *K* is nearly diagonal
  - For large length-scale, the fit is bad, but *K* is almost all 1's

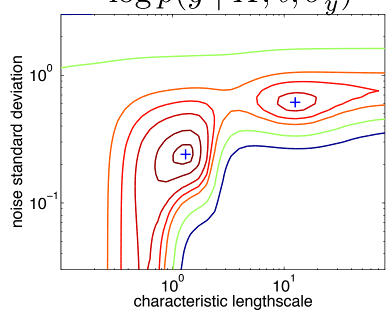
- Can show:

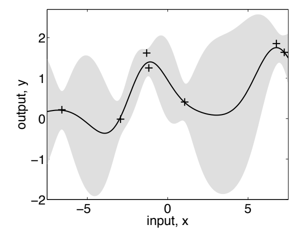
$$\frac{\partial}{\partial \theta_j} \log p(y \mid X, \theta) = \frac{1}{2} y^T K_y^{-1} \frac{\partial K_y}{\partial \theta_j} K_y^{-1} y - \frac{1}{2} \mathrm{tr} \left(K_y^{-1} \frac{\partial K_y}{\partial \theta_j} \right)$$
$$= \frac{1}{2} \mathrm{tr} \left((\alpha \alpha^T - K_y^{-1}) \frac{\partial K_y}{\partial \theta_j} \right)$$

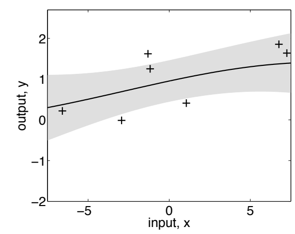
  - Optimize to choose hyperparameters
  - Complexity is
  - Objective is non-convex, so local minima are a problem

Example of Estimating Hypers

$\log p(y \mid X, \ell, \sigma_y^2) \qquad \sigma_f^2 = 1$







Relating GPs to Kernel Methods

- GPs as linear smoothers
  - Recall that the predictive posterior mean of a GP is

$$\bar{f}(x^*) = k_*^T (K + \sigma_y^2 I_n)^{-1} y$$

- In kernel regression, the weight function was derived from a smoothing kernel instead of a Mercer kernel
  - Clear that smoothing kernels have local support
  - Less clear for GPs since the weight function depends on the inverse of *K*

- For some GP kernels, can analytically derive ***equivalent kernel***
  - As with smoothing kernels,
  - Computing a linear combination, but not a convex combination of y_i's
  - Interestingly, the weight function is local even when the GP kernel is not
  - Furthermore, the effective bandwidth of the GP equivalent kernel automatically decreases with *n*, where as in kernel smoothing such tuning must be done by hand

Effective Degrees of Freedom

- For the training set, the fit is given by

$$\hat{f} = K(K + \sigma_y^2 I_n)^{-1} y$$

- Since *K* is a positive definite Gram matrix, it has eigendecomp

$$K = \sum_{i=1}^{n} \lambda_i u_i u_i^T$$

- Using this, one can show that $K(K + \sigma_y^2 I_n)^{-1}$ has eigenvals
- Therefore, the effective degrees of freedom is
- Remember that this specifies how "wiggly" the curve is

Relating GPs to Splines

- Recall smoothing spline objective

$$\min_f \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \int f''(x)^2 dx$$

- Consider the following model

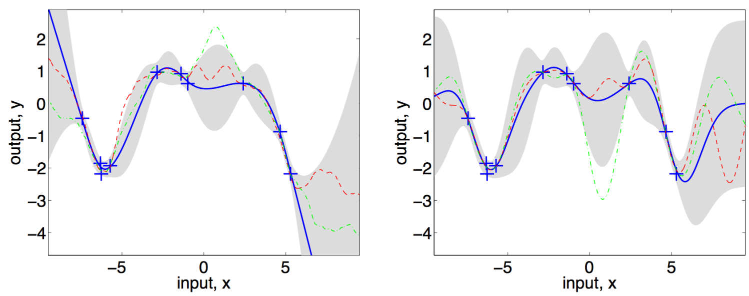
$$f(x) = \beta_0 + \beta_1 x + r(x)$$

where

- One can show that the MAP estimate of *f*(*x*) is a ***cubic smoothing spline*** when $p(\beta_j) \propto 1$
- Penalty parameter λ is now given by σ_y^2 / σ_f^2

Relating GPs to Splines

- The spline kernel leads to a smooth posterior mode/mean, but posterior samples are not smooth.
  - Again, as in lasso, regularizers do not always make good priors



(a), spline covariance (b), squared exponential cov.

Figure from Rasmussen and Williams 2006

- See Rasmussen and Williams 2006 for more details

GP Regression Recap

	Linear Basis Expansion	Gaussian Process
Prior	$\beta \sim N(0, \alpha^{-1} I_M)$ $f(x) = \sum_{m=1}^{M} \beta_m \phi_m(x)$	$f \sim \mathrm{GP}(0, \kappa(x, x'))$
Distribution on $x_1, \ldots, x_n$	$f \sim N(0, \alpha^{-1} \Phi \Phi^T)$	$f \sim N(0, K)$
Choices	• *Choose M* • *Choose bases*	• *Choose* $\kappa(x, x')$ • *Choose covariance hyperparameters*

GP Regression Recap

Linear Basis Expansion	GP regression	Splines	Kernels
$\{\phi_m(x)\}$	$\kappa(x, x')$		
$\downarrow$	$\downarrow$		
f	f		
$\downarrow$	$\downarrow$		
y	y		

Choice of Covariance Function

- Definitions
  - ***Stationary*** kernel – only depends on $x - x'$
  - ***Isotropic*** kernel – furthermore only depends on $||x - x'||$
- Examples
  - ***Squared exponential*** – $\kappa_{SE}(r) = e^{-\frac{r}{2\ell^2}}$
    - Kernel is infinitely differentiable → GP has mean square derivatives of all orders → resulting functions are very smooth
  - ***Matern*** – $\kappa_{Matern}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu r}}{\ell}\right)^{\nu} K_v\left(\frac{\sqrt{2\nu r}}{\ell}\right)$
    - When $\nu \to \infty$: squared exponential
    - When $\nu = \frac{1}{2}$: exponential kernel $\kappa_{exp}(r) = e^{-\frac{r}{\ell}}$
      ** equal to Brownian motion in 1D **

Sample Paths using Matern Kernel

- Can produce very rough sample paths

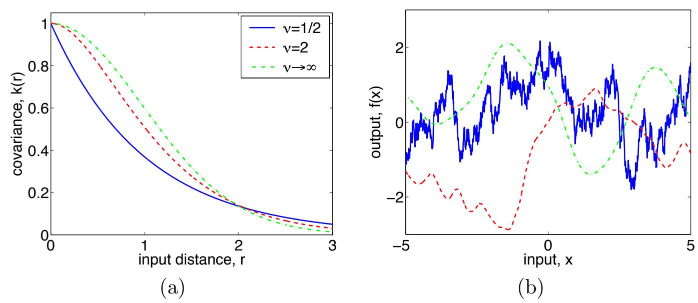
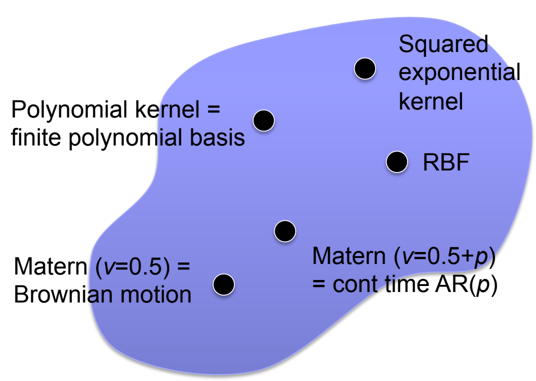


Figure from Rasmussen and Williams 2006

Family of Gaussian Processes



Acknowledgements

Many figures courtesy Kevin Murphy's textbook
Machine Learning: A Probabilistic Perspective,
and Chris Bishop's textbook
Pattern Recognition and Machine Learning

Slides based on parts of the lecture notes of Erik Sudderth for "Applied Bayesian Nonparametrics" at Brown University