HT2015: SC4
Statistical Data Mining and Machine Learning

Dino Sejdinovic
Department of Statistics
Oxford

http://www.stats.ox.ac.uk/~sejdinov/sdmml.html

Parametric vs Nonparametric models

- **Parametric models** have a fixed finite number of parameters, regardless of the dataset size. In the Bayesian setting, given the parameter vector θ, the predictions are independent of the data $\mathcal{D}$.

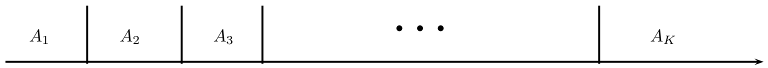
$$p(\tilde{x}, \theta|\mathcal{D}) = p(\theta|\mathcal{D})p(\tilde{x}|\theta)$$

  Parameters can be thought of as a data summary: communication channel flows from data to the predictions through the parameters. Model-based learning (e.g., mixture of K multivariate normals)

- **Nonparametric models** allow the number of “parameters” to grow with the dataset size. Alternatively, predictions depend on the data (and the hyperparameters).
  Memory-based learning (e.g., kernel density estimation)

Dirichlet Process

- We have seen that a conjugate prior over a probability mass function $(\pi_1, \ldots, \pi_K)$ is a Dirichlet distribution $\text{Dir}(\alpha_1, \ldots, \alpha_K)$. Can we create a **prior over probability distributions** on $\mathbb{R}$?
- **Dirichlet process** $\text{DP}(\alpha, h)$, $\alpha > 0$ and H a probability distribution on $\mathbb{R}$
  A random probability distribution F is said to follow a Dirichlet process if when restricted to any finite partition it has a Dirichlet distribution, i.e., for any partition $A_1, \ldots, A_K$ of $\mathbb{R}$,
  $(F(A_1), \ldots, F(A_K)) \sim \text{Dir}(\alpha h(A_1), \ldots, \alpha h(A_K))$



- **Stick-breaking construction** allows us to draw from a Dirichlet process:
  1. Draw $s_1, s_2, \ldots \overset{i.i.d.}{\sim} h$
  2. Draw $v_1, v_2, \ldots \overset{i.i.d.}{\sim} \text{Beta}(1, \alpha)$
  3. Set $w_1 = v_1, w_2 = v_2(1 - v_1), \ldots, w_j = v_j \prod_{\ell=1}^{j-1}(1 - v_\ell) \ldots$

  Then $\sum_{\ell=1}^{\infty} w_\ell \delta_{s_\ell} \sim \text{DP}(\alpha, h)$

Dirichlet Process and a Posterior over Distributions

- Given data $\mathcal{D} = \{x_i\}_{i=1}^n \overset{i.i.d.}{\sim} F$, $x_i \in \mathbb{R}^p$, we put a prior $\text{DP}(\alpha, h)$ on F
- Posterior $p(F|\mathcal{D})$ is $\text{DP}(\alpha + n, \bar{h})$, where $\bar{h} = \frac{n}{n+\alpha}\hat{F} + \frac{\alpha}{n+\alpha}h$ and $\hat{F} = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$ is the empirical distribution.
- But how to reason about this posterior? Answer: sample from it!

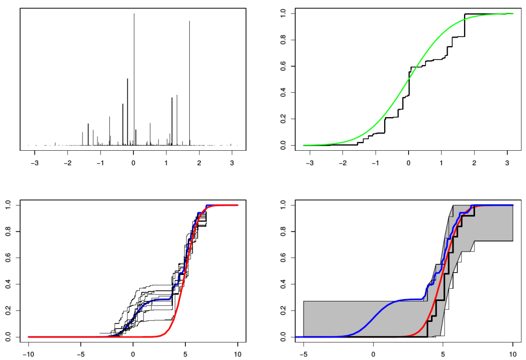


Figure : top left: a draw from $\text{DP}(10, \mathcal{N}(0, 1))$; top right: resulting cdf; bottom left: draws from a posterior based on $n = 25$ observations from a $\mathcal{N}(5, 1)$ distribution (red); bottom right: Bayesian posterior mean (blue), empirical cdf (black).

Example and Figure by L. Wasserman

Dirichlet Process Mixture Models

- In mixture models for clustering, we had to pick the number of clusters K. Can we automatically infer K from data?
- Just use an infinite mixture model

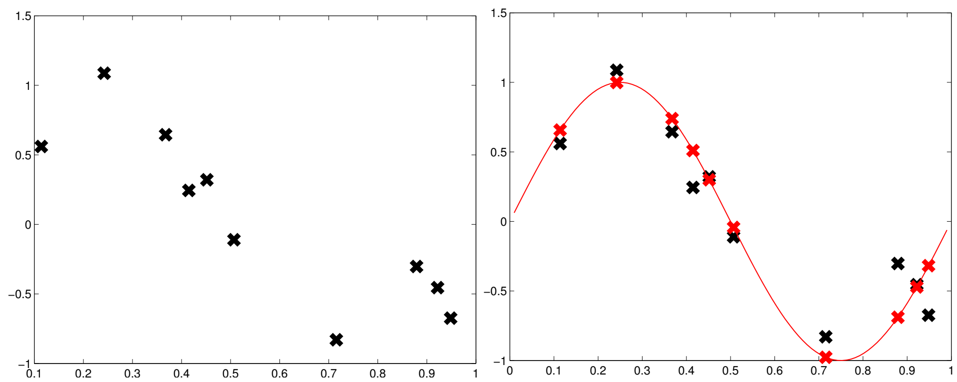
$$g(x) = \sum_{k=1}^{\infty} \pi_k p(x|\theta_k)$$

  The following generative process defines an implicit prior on g:
  1. Draw $F \sim \mathrm{DP}(\alpha, h)$
  2. Draw $\theta_1, \ldots, \theta_n | F \overset{i.i.d.}{\sim} F$
  3. Draw $x_i|\theta_i \sim p(\cdot|\theta_i)$
- F is discrete and will get ties - ties form clusters.
- Posterior distribution is more involved but can be sampled from[1].

[1] Radford Neal, 2000: Markov Chain Sampling Methods for Dirichlet Process Mixture Models

Gaussian Processes

Regression



- We are given a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, $x_i \in \mathbb{R}^p$, $y_i \in \mathbb{R}$.
- Regression: learn the underlying real-valued function $f(x)$.

Different Flavours of Regression

- We can model response y_i as a noisy version of the underlying function f evaluated at input x_i:

$$y_i|f, x_i \sim \mathcal{N}(f(x_i), \sigma^2)$$

  Appropriate loss: $L(y, f(x)) = (y - f(x))^2$
- **Frequentist Parametric** approach: model f as f_θ for some parameter vector θ. Fit θ by ML / ERM with squared loss (linear regression).
- **Frequentist Nonparametric** approach: model f as the unknown parameter taking values in an infinite-dimensional space of functions. Fit f by regularized ML / ERM with squared loss (kernel ridge regression)
- **Bayesian Parametric** approach: model f as f_θ for some parameter vector θ. Put a prior on θ and compute a posterior $p(\theta|\mathcal{D})$ (Bayesian linear regression).
- **Bayesian Nonparametric** approach: treat f as the random variable taking values in an infinite-dimensional space of functions. Put a prior over functions $f \in \mathcal{F}$, and compute a posterior $p(f|\mathcal{D})$ (Gaussian Process regression).

- Just work with the function values at the inputs $\mathbf{f} = (f(x_1), \dots, f(x_n))^\top$
- What properties of the function can we incorporate?
  - Multivariate normal prior on $\mathbf{f}$:

$$\mathbf{f} \sim \mathcal{N}(0, \mathbf{K})$$

  - Use a kernel function k to define $\mathbf{K}$:

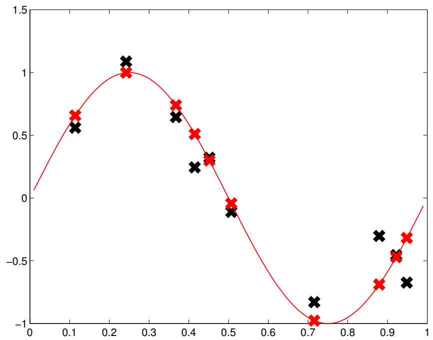
$$\mathbf{K}_{ij} = k(x_i, x_j)$$

  - Expect regression functions to be smooth: If x and x' are close by, then $f(x)$ and $f(x')$ have similar values, i.e. strongly correlated.

$$\begin{pmatrix} f(x) \\ f(x') \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} k(x,x) & k(x,x') \\ k(x',x) & k(x',x') \end{pmatrix} \right)$$

In particular, want $k(x,x') \approx k(x,x) = k(x',x')$.

The prior $p(\mathbf{f})$ encodes our prior knowledge about the function.



- Model:

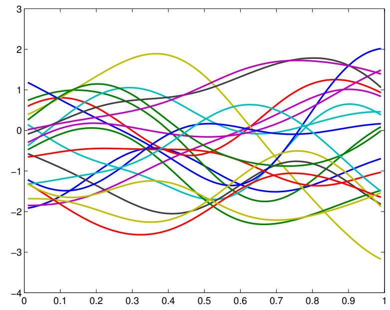
$$\mathbf{f} \sim \mathcal{N}(0, \mathbf{K})$$
$$y_i | f_i \sim \mathcal{N}(f_i, \sigma^2)$$

Gaussian Processes

- What does a multivariate normal prior mean?
- Imagine $\mathbf{x}$ forms an infinitesimally dense grid of data space. Simulate prior draws

$$\mathbf{f} \sim \mathcal{N}(0, \mathbf{K})$$

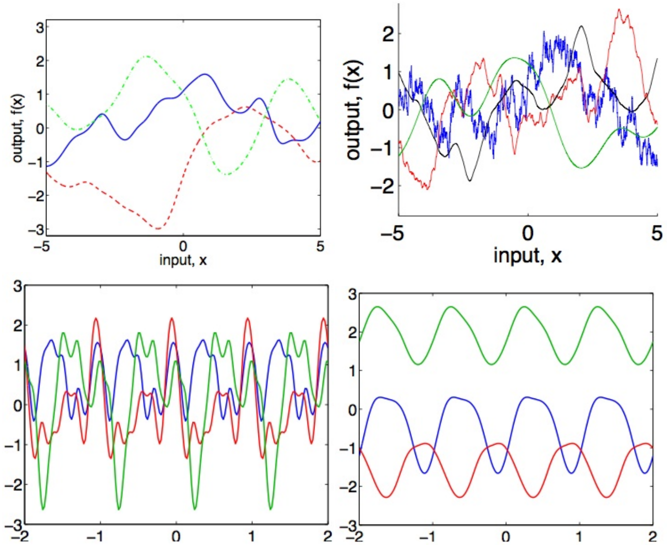
Plot f_i vs x_i for $i = 1, \dots, n$.
- The corresponding prior over functions is called a **Gaussian Process** (GP): any finite number of evaluations of which follow a Gaussian distribution.



http://www.gaussianprocess.org/

Gaussian Processes

- Different kernels lead to different function characteristics.



Carl Rasmussen. Tutorial on Gaussian Processes at NIPS 2006.

Gaussian Processes

$$\mathbf{f}|\mathbf{x} \sim \mathcal{N}(0, \mathbf{K})$$
$$\mathbf{y}|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma^2 I)$$

- Posterior distribution:

$$\mathbf{f}|\mathbf{y} \sim \mathcal{N}(\mathbf{K}(\mathbf{K} + \sigma^2 I)^{-1}\mathbf{y}, \mathbf{K} - \mathbf{K}(\mathbf{K} + \sigma^2 I)^{-1}\mathbf{K})$$

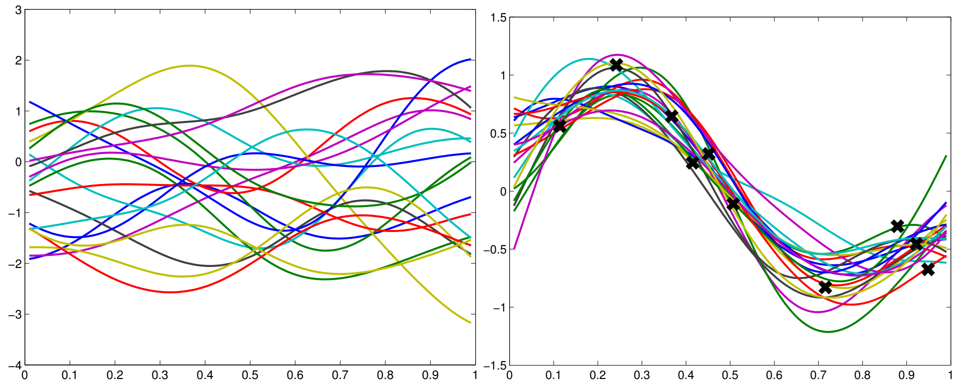
- Posterior predictive distribution: Suppose $\mathbf{x}'$ is a test set. We can extend our model to include the function values $\mathbf{f}'$ at the test set:

$$\begin{pmatrix} \mathbf{f} \\ \mathbf{f}' \end{pmatrix} | \mathbf{x}, \mathbf{x}' \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{K_{xx}} & \mathbf{K_{xx'}} \\ \mathbf{K_{x'x}} & \mathbf{K_{x'x'}} \end{pmatrix} \right)$$
$$\mathbf{y}|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma^2 I)$$

where $\mathbf{K_{xx'}}$ is matrix with (i,j)-th entry $k(x_i, x'_j)$.
- Some manipulation of multivariate normals gives:

$$\mathbf{f}'|\mathbf{y} \sim \mathcal{N}\left(\mathbf{K_{x'x}}(\mathbf{K_{xx}} + \sigma^2 I)^{-1}\mathbf{y}, \mathbf{K_{x'x'}} - \mathbf{K_{x'x}}(\mathbf{K_{xx}} + \sigma^2 I)^{-1}\mathbf{K_{xx'}}\right)$$

Gaussian Processes



GP regression demo

http://www.tmpl.fi/gp/

- A whirlwind journey through data mining and machine learning techniques:
  - **Unsupervised learning**: PCA, MDS, Isomap, Hierarchical clustering, K-means, mixture modelling, EM algorithm, Dirichlet process mixtures.
  - **Supervised learning**: LDA, QDA, naïve Bayes, logistic regression, SVMs, kernel methods, kNN, deep neural networks, Gaussian processes, decision trees, ensemble methods: random forests, bagging, stacking, dropout and boosting.
  - **Conceptual frameworks**: prediction, performance evaluation, generalization, overfitting, regularization, model complexity, validation and cross-validation, bias-variance tradeoff.
  - **Theory**: decision theory, statistical learning theory, convex optimization, Bayesian vs. frequentist learning, parametric vs non-parametric learning.
- **Further resources**:
  - Machine Learning Summer Schools, videolectures.net.
  - Conferences: NIPS, ICML, UAI, AISTATS.
  - Mailing list: ml-news.

Thank You!