STATG019 – Selected Topics in Statistics 2015

Lecture 3

Learning with Gaussian Processes

Dr Franz J. Király

Course organization

Lecture schedule

Simon will start next week with point processes

Likely plan: 2 or 3 lectures on point processes, then kernels again

Are you fine with an ICA on the content of the first three kernels lectures?

(so you will not have to worry strategically about which topics to choose)

Voting about further kernels topics in the coming weeks.

Feel free to suggest topics that interest you!

Tutorials and/or practical sessions

Thursday, 11am - 1pm, January 29

Gordon Square 16-18, room 101

Tutorial will be mainly on kernlab with R and feature maps

Please install R and kernlab on your laptops (cluster rooms are full)

Informal schedule: no frontal teaching, questions are very welcome

Are kernels for the frequentist only?

Support vector machines, ridge regession etc allow non-linear regression and classification

Input: independent data points $x_1, \ldots, x_N \in \mathbb{R}^n$
dependent data points/labels $y_1, \ldots, y_N \in \mathbb{R}$

Kernel ridge regression:

$$f(x) = y^\top (K + \lambda I)^{-1} \cdot \kappa(x) = \widehat{\alpha}^\top \kappa(x)$$

where $\widehat{\alpha}$ minimizes $R(\alpha) = \|y - K \cdot \alpha\|^2_{F,\mathcal{H}} + \lambda \cdot \alpha^\top K \alpha$

$$K = (k(x_i, x_j))_{ij} \qquad \kappa(x) = (k(x_i, x))_i$$

Kernel SVM:

$$f(x) = \text{sgn}\left(b + \sum_{i=1}^{N} \alpha_i k(x_i, x)\right) = \text{sgn}\left(\alpha^\top \kappa(x)\right)$$

$$\max_\alpha W(\alpha) = \|\alpha\|_1 - \frac{1}{2} \cdot \alpha^\top \tilde{K} \alpha \qquad \text{s.t.} \quad \alpha \geq 0, \quad 0 = \sum_{i=1}^{N} \alpha_i y_i$$

Both formulations are discriminative and non/probabilistic

Is there a way to include prior information on data samples?

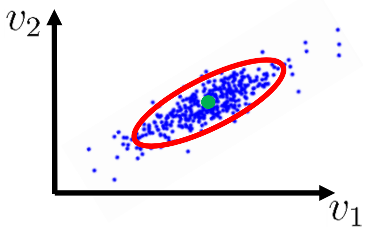
Bayesian inference? Confidence intervals? Estimation of kernel parameters?

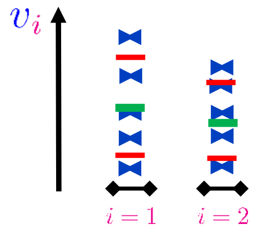
Gaussian Processes

(multivariate) Gaussian random variable X

outcomes/realizations are vectors

$$v \in \mathbb{R}^n, \quad v_1 \sim X_1, \dots, v_n \sim X_n,$$





possible **definition**:

$$p_X(v) = \frac{\det(\Sigma^{-1})}{\sqrt{(2\pi)^n}} \exp\left(-\frac{1}{2}(v-\mu)^\top \Sigma^{-1}(v-\mu)\right)$$

for some $\mu \in \mathbb{R}^n$ and positive semi-definite $\Sigma \in \mathbb{R}^{n\times n}$

another possible **definition**:

For any $a \in \mathbb{R}^n$, the marginal $\langle a, X\rangle = \sum_{i=1}^n a_i X_i$ is univariate Gaussian

classical result (immediate from definition 1):

X is uniquely determined by all

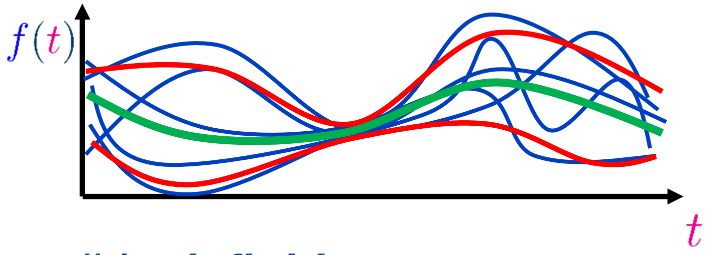
$\mu_i = \mathbb{E}(X_i)$ "mean vector" $\quad \Sigma_{ij} = \text{Cov}(X_i, X_j)$ "covariance matrix"

$$X \sim \mathcal{N}(\mu, \Sigma)$$

Gaussian process X

outcomes/realizations are functions

$$f : \mathbb{R}^n \to \mathbb{R}, \quad f(t) \sim X_t \text{ for all } t \in \mathbb{R}^n$$



possible **definition**:

For any $t_1, \dots, t_m \in \mathbb{R}^n$, the $X_{t_1}, \dots, X_{t_m}$ are joint Gaussian

another possible **definition**:

For any $t_1, \dots, t_m \in \mathbb{R}^n, a \in \mathbb{R}^m$, the marginal $\sum_{i=1}^n a_i X_{t_i}$ is univariate Gaussian

classical result (by using properties of mgf):

X is uniquely determined by all

$\mu(t) = \mathbb{E}(X_t)$ "mean function" $\quad k(s,t) = \text{Cov}(X_s, X_t)$ "covariance function"

$$X \sim \mathcal{GP}(\mu, k)$$

A (non-exhaustive) list of popular covariance functions

$$k(x,y) = C$$

constant covariance

$$k(x,y) = \sigma^2 \cdot \delta_{xy}$$

Gaussian noise

$$k(x,y) = x^\top A y$$

linear covariance function

$$k(x,y) = \rho\left(\|x-y\|\right)$$

Distance covariance functions

$$k(x,y) = f\left(\langle x,y\rangle\right)$$

Dot-product covariance functions

$$k(x,y) = \exp\left(-\frac{2}{\sigma^2}\sin^2\left(\frac{1}{2}\|x-y\|^2\right)\right)$$

Periodic covariance function

$$k(x,y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$$

Squared exponential covariance function

$$k(x,y) = \exp\left(-\frac{\|x-y\|}{\sigma}\right)$$

Ornstein-Uhlbeck covariance function

$$k(x,y) = \left(1 + \frac{\|x-y\|^2}{\sigma^2}\right)^{-1}$$

Rational quadratic covariance function

Proposition:

For covariance functions $k_1, k_2, \ldots$, and $\alpha_i \in \mathbb{R}_{\geq 0}$,

a k defined by $k(x,y) =$ as any below is a covariance function:

$$k_3(x,y) + \alpha_{42}$$

$$\alpha_1 \cdot k_1(x,y) + \alpha_2 \cdot k_2(x,y)$$

$$k_1(x,y) \cdot k_2(x,y)$$

$$\sum_{\nu=1}^{\infty} \alpha_\nu \langle x,y\rangle^\nu$$

(in case of convergence)

$$\lim_{\nu\to\infty} k_\nu(x,y)$$

(in case of existence)

$$\frac{k_1(x,y)}{\sqrt{k_1(x,x)\cdot k_1(y,y)}}$$

An exhaustive list of popular mean functions

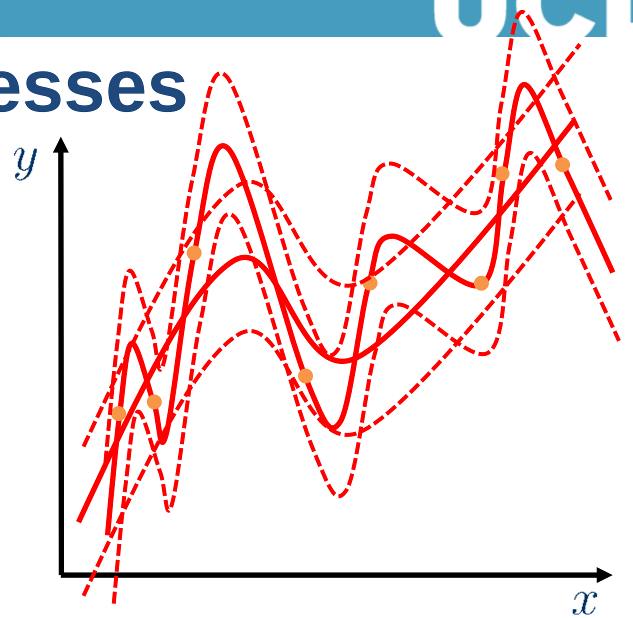
$$\mu(x) = 0$$

Non-zero mean is usually application-specific when used,
it encodes prior knowledge on the possible outcome functions.

In application where there is only *one* sample of a process
- as in the following-
the mean function cannot be reliably estimated,
thus setting it to zero is scientifically parsimonious
in the absence of further prior knowledge.

Gaussian process regression

Interpolation ~~Regression~~ with Gaussian processes



Input: independent data points $x_1, \dots, x_N \in \mathbb{R}^n$
dependent data points/labels $y_1, \dots, y_N \in \mathbb{R}$

Output: Regressor $f : \mathbb{R}^n \to \mathbb{R}$ such that $y_i \approx f(x_i)$
and which predicts well unseen labels $f(x)$

Main assumption:

The "true" f is outcome of Gaussian process

Mathematical/statistical idea:

do Bayesian inference to obtain posterior for $f(x)$

$$\begin{pmatrix} f(x) \\ f(x_1) \\ \vdots \\ f(x_N) \end{pmatrix} \underset{\text{by main ass.}}{\sim} \mathcal{N}\left(0, \begin{pmatrix} k(x,x) & k(x,x_1) & \dots & k(x,x_n) \\ k(x_1,x) & k(x_1,x_1) & \dots & k(x_1,x_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(x,x) & k(x_n,x) & \dots & k(x_n,x_n) \end{pmatrix}\right) = \mathcal{N}\left(0, \begin{pmatrix} k(x,x) & \kappa(x)^\top \\ \kappa(x) & K \end{pmatrix}\right)$$

$K = (k(x_i, x_j))_{ij}$
"covariance matrix"

$\kappa(x) = (k(x_i, x))_i$
"cross-covariance vector"

Bayes' theorem: $p_{f(x)|f(X)}(z|Z) = p_{f(x),f(X)}(z,Z)/p_{f(X)}(Z)$

where $p_{f(X)}(Z) = \int_{\mathbb{R}^n} p_{f(x),f(X)}(z,Z)\, dz$

An elementary though tedious computation **yields:** $f(x)|f(x_1), \dots, f(x_N) \sim \mathcal{N}\left(\kappa(x)^\top K^{-1} y,\ k(x,x) - \kappa(x)^\top K^{-1} \kappa(x)\right)$ if $y_i = f(x_i)$

so we can estimate $\mathbb{E} f(x) = \kappa(x)^\top K^{-1} y$

and we obtain error bars $\mathrm{SE} f(x) \approx \sqrt{k(x,x) - \kappa(x)^\top K^{-1} \kappa(x)}$

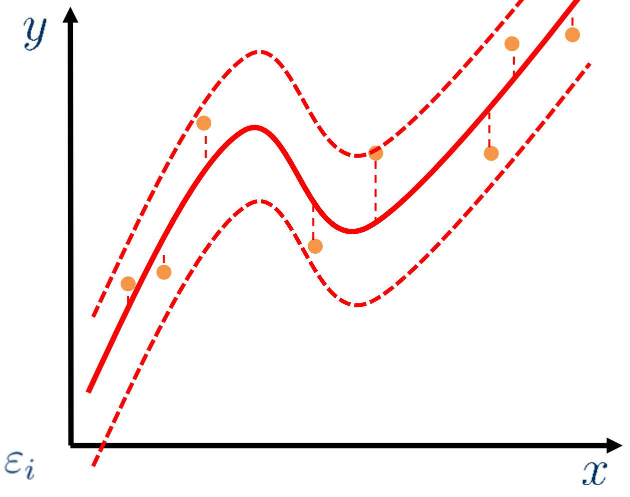
Bad news:

overfits horribly

Regression with Gaussian processes

Input: independent data points $x_1, \dots, x_N \in \mathbb{R}^n$
dependent data points/labels $y_1, \dots, y_N \in \mathbb{R}$

Output: Regressor $f : \mathbb{R}^n \to \mathbb{R}$ such that $y_i \approx f(x_i)$
and which predicts well unseen labels $f(x)$



Main assumptions:

The "true" f is outcome of Gaussian process
(jointly) Gaussian noise on observations: $y_i = f(x_i) + \varepsilon_i$

Equivalently: $y_i = g(x_i)$ with $g \sim \mathcal{GP}(0,k) + \mathcal{GP}(0,k_\varepsilon)$ s.t. $\epsilon_1, \dots, \epsilon_N \sim \mathcal{N}(0, K_\varepsilon)$

$K_\varepsilon = (k_\varepsilon(x_i, x_j))_{ij}$

By additivity of variance: $g \sim \mathcal{GP}(0, k + k_\varepsilon)$

So regression with noise can be reduced to interpolation!

Substitution into previous equations yields, for posterior:

$$f(x)|y_1, \dots, y_N \sim \mathcal{N}\left(\tilde{\kappa}(x)^\top (K + K_\varepsilon)^{-1} y,\ k(x,x) - \tilde{\kappa}(x)^\top (K + K_\varepsilon)^{-1} \tilde{\kappa}(x)\right)$$

where $\tilde{\kappa}(x) = (k(x, x_i) + k_\varepsilon(x, x_i))_i$

for i.i.d. Gaussian noise of variance λ:

$k_\varepsilon(x,y) = \lambda \cdot \delta_{xy}$ thus posterior is (in the case $x \neq x_1, \dots, x_N$)

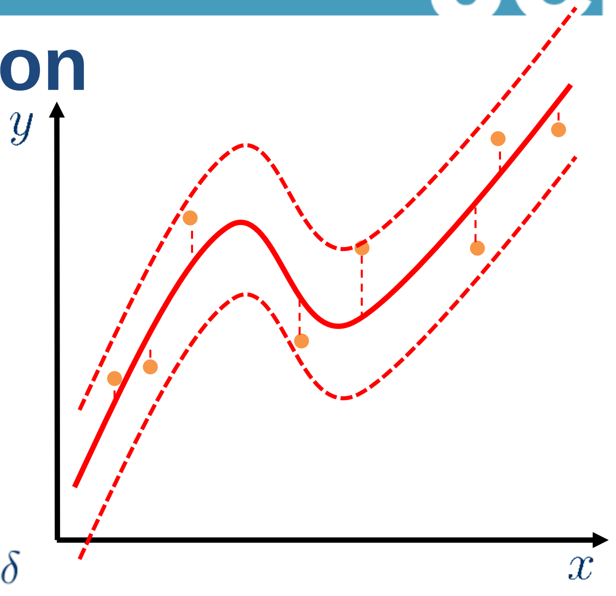
$$f(x)|y_1, \dots, y_N \sim \mathcal{N}\left(\kappa(x)^\top (K + \lambda \cdot I)^{-1} y,\ k(x,x) - \kappa(x)^\top (K + \lambda \cdot I)^{-1} \kappa(x)\right)$$

so we can estimate $\mathbb{E} f(x) = \kappa(x)^\top (K + \lambda \cdot I)^{-1} y$

with error bars SE $f(x) \approx \sqrt{k(x,x) - \kappa(x)^\top (K + \lambda \cdot I)^{-1} \kappa(x)}$

where $K = (k(x_i, x_j))_{ij}$
$\kappa(x) = (k(x, x_i))_i$

On error and parameter estimation



Input: independent data points $x_1, \dots, x_N \in \mathbb{R}^n$
dependent data points/labels $y_1, \dots, y_N \in \mathbb{R}$

Output: mean estimate $f(x) = \kappa(x)^\top (K + \lambda \cdot I)^{-1} y$
with variance $k(x,x) - \kappa(x)^\top (K + \lambda \cdot I)^{-1} \kappa(x)$

Remark 1: posterior $f(x)$ is Gaussian RV
so α-confidence intervals are obtained as usual

Remark 2: let ϕ be the feature map of the kernel $k + \lambda \cdot \delta$
Then $\mathrm{Var} f(x) = k(x,x) - \kappa(x)^\top (K + \lambda \cdot I)^{-1} \kappa(x) = \|\phi(x) - P_X \phi(x)\|^2$
where P_X denotes the projector onto span$\{\phi(x_1), \dots, \phi(x_N)\}$ in feature space

Remark 3: prediction depends on parameter λ and all parameters θ occurring in K, κ
The probabilistic viewpoint allows to find best parameters via the following

Methods: Maximum likelihood, maximum-a-posteriori, posterior expectation

Log-likelihood: $2 \cdot \ell(\theta | y_1, \dots, y_N, x_1, \dots, x_N) = \log \chi_K(-\lambda) - y^\top (K + \lambda \cdot I)^{-1} y - n \log 2\pi$
$= 2 \log p(y_1, \dots, y_N | \theta, x_1, \dots, x_N)$

with $\chi_K(t) = \det(t \cdot I - K)$
"characteristic polynomial"

Bayes posterior: $p(\theta | X, y) = p(X | \theta, y) \cdot \pi(\theta)$ "prior"

Bayesian prediction: $p(x | X, y) = \int_\theta p(x|\theta) \cdot p(\theta | X, y) \, d\theta$

Optimization and integrals require numerical methods…

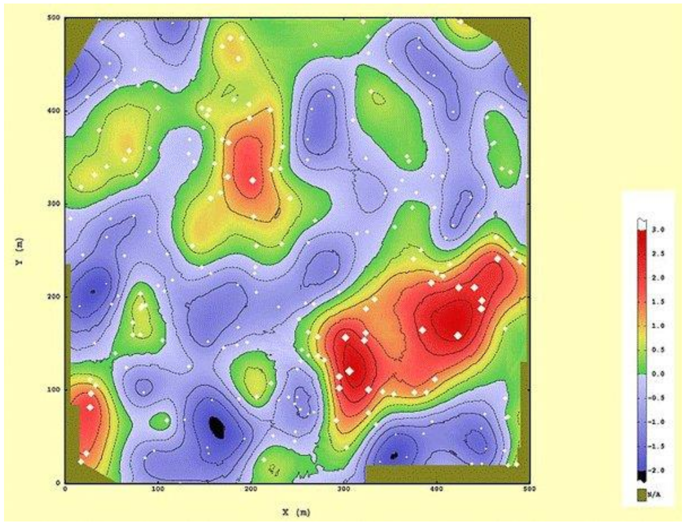
Kriging

Kriging (after D.G. Krige's master's thesis, 1951)

Input: independent data points $x_1, \dots, x_N \in \mathbb{R}^n$
dependent data points/labels $y_1, \dots, y_N \in \mathbb{R}$

Output: Regressor $f : \mathbb{R}^n \to \mathbb{R}$ such that $y_i \approx f(x_i)$
and which predicts well unseen labels $f(x)$

Mathematically identical to GP/ridge regression and Gaussian process prediction, with a ***few differences:***



Application predominantly in geostatistics to find ore, oil, gold, etc $n = 2, 3$

Interpretation as error minimizing prediction for specific point processes

Mean functions $\mu(x) = c$ "ordinary kriging" $\mu(x) = P(x)$ polynomial "universal kriging" known, or unknown

The variogram $$\gamma(x,y) = \frac{1}{2}\mathrm{Var}\left(f(x) - f(y)\right) = \frac{1}{2}k(x,x) + \frac{1}{2}k(y,y) - k(x,y) + \frac{1}{2}\left(\mu(x) - \mu(y)\right)^2$$

"empirical variogram":

$$\widehat{\gamma}(h) = \frac{1}{\#N(h)} \sum_{(i,j)\in N(h)} \|y_i - y_j\|^2 \quad \text{where} \quad N(h) = \{(i,j) \;:\; \|x_i - x_j\| = h\}$$

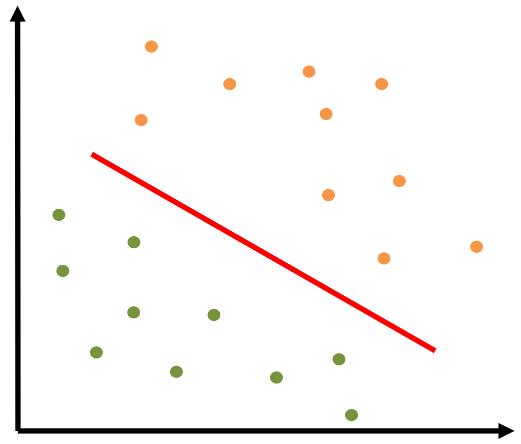
is used to obtain empirical estimate for k *then* regression is applied.

Gaussian process classification

Classification with Gaussian processes

Regression: For data $x_1, \ldots, x_N \in \mathbb{R}^n$ and labels $y_1, \ldots, y_N \in \mathbb{R}$
learn a regressor f such that $f(x_i) \approx y_i$

Classification: For data $x_1, \ldots, x_N \in \mathbb{R}^n$ and labels $y_1, \ldots, y_N \in \{-1, 1\}$
learn a classifier g such that $g(x_i) \approx y_i$



Idea 1: apply sigmoid link function $\varphi : \mathbb{R} \to [0, 1]$
to a regressor f to get a classifier $g = \varphi \circ f$
where hopefully $f(x)|y_1, \ldots, y_N \sim \mathcal{N}$(something)
estimate $\mathbb{E}g(x) = \int \varphi(f(x)) \cdot p(f(x)|y_1, \ldots, y_N)\, d[f(x)]$

Idea 2: p is Gaussian, so choose φ where integral is analytic

Good news: idea 2 works, for example with $\varphi = \Phi$ a standard normal cdf
in this case, $\mathbb{E}g(x) = \Phi\left(\mathbb{E}f(x) \cdot (1 + \text{Var}f(x))^{-1/2}\right)$

Bad news: $f(x)|y_1, \ldots, y_N$ will never be Gaussian except in uninteresting cases

Solution idea: first-order approximation to $f(x_1), \ldots, f(x_N)|y_1, \ldots, y_N \;\sim \mathcal{N}(\widehat{f}, \widehat{K})$

an elementary computation yields $\widehat{f} = \operatorname*{argmax}_f p(f|y) \qquad \widehat{K}^{-1} = \frac{\partial^2}{\partial^2 f} p(f|y)\big|_{f=\widehat{f}}$

so approximately $f(x)|y \sim \mathcal{N}(\kappa(x)^\top K^{-1}\widehat{f}, k(x,x) - \kappa(x)^\top \left(K + \widehat{K}\right)^{-1} \kappa(x))$

The Relevance Vector Machine

= Gaussian process regression/classification

with covariance function: $k(x,y) = \lambda \cdot \delta_{xy} + \sum_{i=1}^{N} k(x,x_i)\alpha_i^{-1}k(x_i,y)$

with i.i.d. hyperpriors: $p(\alpha_i) = \text{Gamma}(a,b)$ $\quad p(\lambda^{-1}) = \text{Gamma}(c,d)$

that ensure sparsity of α

usually $a = b = c = d = 0$

with specific optimization updates including expectation maximization

and a US patent (US 6633857) granted to

Microsoft

… so it is not available in kernlab.

Implementing the covariance function above and
invoking `gausspr` with the above parameters may constitute,
under certain circumstances, patent infringement as defined by US patent law

Using Gaussian processes in kernlab

Gaussian processes in kernlab

`gausspr` offers regression and classification with Gaussian processes

usage for training:

```
gprmodel <- gausspr(vartopredict~.,data=traindata,…)
```

trains a predictor, stored in the output variable `gprmodel` of type `gausspr`

important parameters for `gausspr`:

`type`	`"classification"` or `"regression"`
`kernel`	determines the kernel used, e.g. `"rbf-dot"`
`kpar`	a list of kernel parameters, e.g. `list(sigma=1)`
`var`	noise variance for i.i.d. Gaussian noise

output contains model parameters as `alpha` $= (K + \lambda \cdot I)^{-1} y$

`"regression"` is equivalent to ridge regression with regularizer `var`

usage for prediction:

```
predicted <- predict(gprmodel, testdata)
```

yields a vector `predicted` of predictions for `testdata`

Documentation: http://cran.r-project.org/web/packages/kernlab/kernlab.pdf

Type `help(gausspr)` or `example(gausspr)` for more details and examples

Outlook

Lecture 1: Introduction to kernels

- Main concepts and theoretical results, learning guarantees
- Kernel PCA and kernel ridge regression
- Some notes on R and kernlab

Lecture 2: the kernel support vector machine

- The linear support vector machine, duality
- Hard- and soft-margin two-class SVM
- The one-class SVM
- Support vector regression

Lecture 3: Gaussian processes and kernel learning

Potential further lecture topics:

- Algorithms: kernel discriminants, kernel k-means, kernel quantile regression
  kernel CCA, kernel MMD, kernel relevance vector machine
- Large-scale learning with kernels, subset methods and Nyström-approximation
- Combinatorial kernels: string kernels, graph kernels
- Invariance kernels
- Vapnik-Chervonenkis learning theory
- Outlier detection, novelty detection
- On-line kernel learning

Next week: Point Processes (Simon)