Lecture 2: From Linear Regression to Kalman Filter and Beyond

Simo Särkkä

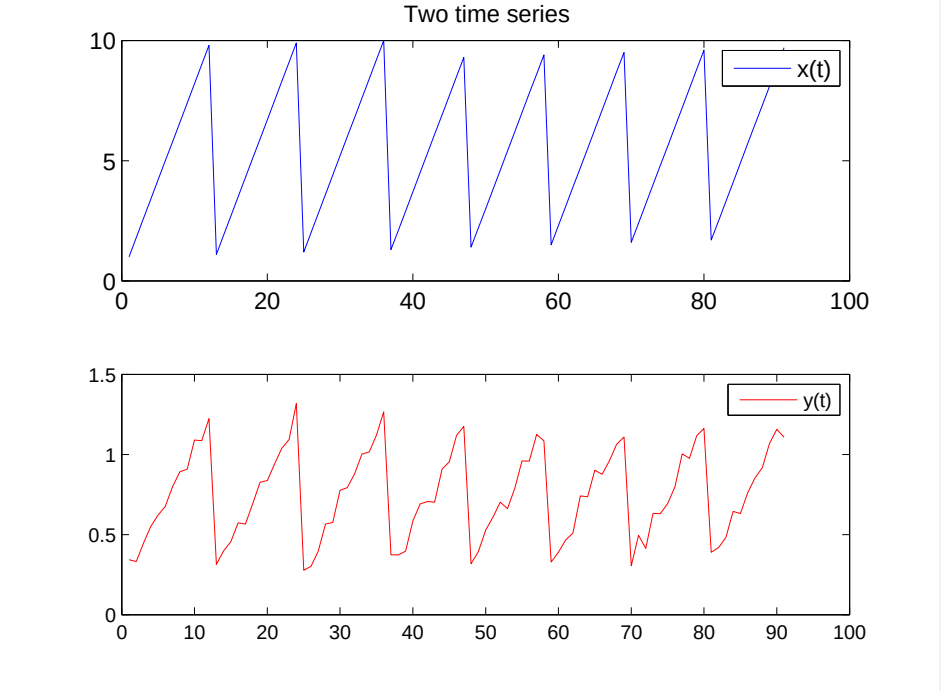
Department of Biomedical Engineering and Computational Science
Helsinki University of Technology

March 24, 2009

Contents

Introduction: Correlation of Time Series

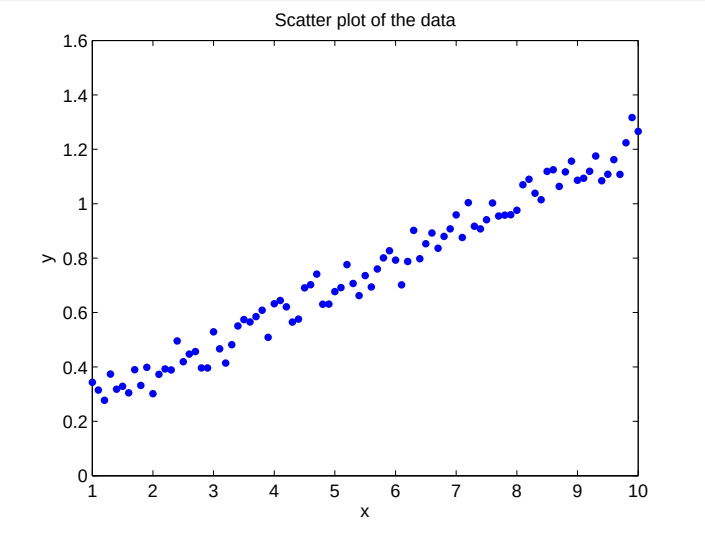
- Assume that we have two signals $x(t)$ and $y(t)$:

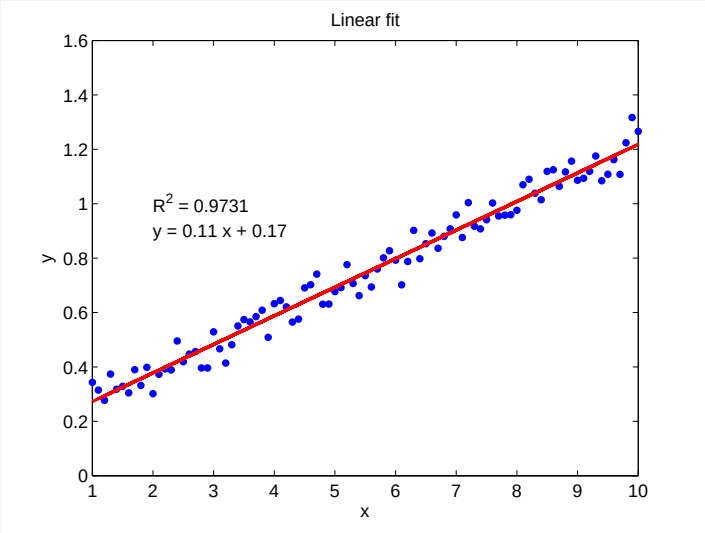


- We can now measure that the squared correlation R^2 of these is 0.97.
- What does this really mean?

Introduction: Correlation of Time Series (cont.)

- Plot x and y on separate axes (scatter plot) and fit a line $y = a\,x + b$ to it:

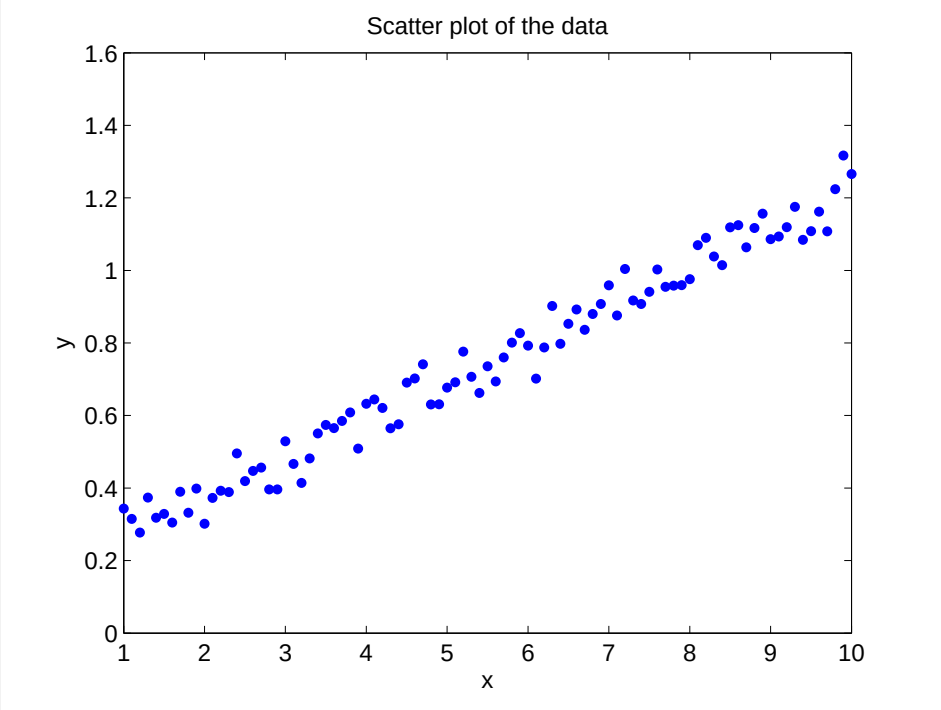




- The correlation coefficient measures how well the regression line fits to the data.

Least Squared Solution to Linear Regression [1/3]

- We have observed pairs $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$:



- Assume linear relationship

$$y^{(i)} = a\,x^{(i)} + b, \qquad i = 1, \ldots, n.$$

- We want to estimate the parameters a and b.

Least Squared Solution to Linear Regression [2/3]

- In the least squares method, we minimize the mean squared error:

$$S(a,b) = \frac{1}{n}\sum_i (y^{(i)} - ax^{(i)} - b)^2$$
$$= \mathrm{E}[(y - ax - b)^2],$$

  where $\mathrm{E}[\cdot]$ denotes the expectation of the argument.
- In the minimum, derivatives must vanish:

$$\frac{\partial S}{\partial a} = \frac{2}{n}\sum_i x^{(i)}\,(y^{(i)} - ax^{(i)} - b) = 0$$
$$\frac{\partial S}{\partial b} = \frac{2}{n}\sum_i (y^{(i)} - ax^{(i)} - b) = 0.$$

Least Squared Solution to Linear Regression [3/3]

- The final solution is

$$a = \frac{\mathrm{E}[x\,y] - \mathrm{E}[x]\;\mathrm{E}[y]}{\mathrm{E}[x^2] - \mathrm{E}[x]^2}$$
$$b = \mathrm{E}[y] - a\,E[x],$$

where

$$E[xy] = \frac{1}{n}\sum_i x^{(i)} y^{(i)}, \qquad E[x^2] = \frac{1}{n}\sum_i x^{(i)} x^{(i)}$$
$$E[x] = \frac{1}{n}\sum_i x^{(i)}, \qquad E[y] = \frac{1}{n}\sum_i y^{(i)}.$$

Correlation coefficient R

- The regression line equation has the form

$$y - \mathrm{E}[y] = a\,(x - \mathrm{E}[x]),$$

  and we could use parameter a as measure of the linear relationship between x and y.
- The problem is that if we scale x or y by constant, the coefficient a changes also.
- Thus instead, we should consider the relationship between the normalized and centered variables:

$$\tilde{y} = \frac{y - \mathrm{E}[y]}{\sqrt{\mathrm{E}[y^2] - \mathrm{E}[y]^2}}$$
$$\tilde{x} = \frac{x - \mathrm{E}[x]}{\sqrt{\mathrm{E}[x^2] - \mathrm{E}[x]^2}}.$$

Correlation coefficient R (cont)

- The equation reduces to form

$$\tilde{y} = \underbrace{\frac{\mathrm{E}[x\,y] - \mathrm{E}[x]\,E[y]}{\sqrt{\mathrm{E}[x^2] - \mathrm{E}[x]^2}\sqrt{\mathrm{E}[y^2] - \mathrm{E}[y]^2}}}_{R}\,\tilde{x}.$$

- The proportionality coefficient R is the correlation coefficient:

$$R = \frac{\mathrm{E}[x\,y] - \mathrm{E}[x]\,E[y]}{\sqrt{\mathrm{E}[x^2] - \mathrm{E}[x]^2}\sqrt{\mathrm{E}[y^2] - \mathrm{E}[y]^2}}.$$

Coefficient of determination R^2

- The correlation coefficient can be also derived in another way, which generalizes to non-linear models.
- The remaining mean squared error or the residual variance using the fitted parameters a and b can be written as

$$\begin{aligned} S(a,b) &= \frac{1}{n}\sum_i (y^{(i)} - ax^{(i)} - b)^2 \\ &= \mathrm{E}[y^2] - \mathrm{E}[y]^2 - \frac{(\mathrm{E}[x\,y] - \mathrm{E}[x]\;\mathrm{E}[y])^2}{\mathrm{E}[x^2] - \mathrm{E}[x]^2}. \end{aligned}$$

- One way of measuring the goodness of the fit is to compare the residual variance to the original data variance $\mathrm{Var}[y]$.

Coefficient of determination R^2 (cont)

- The proportion of variance (MSE) in y that the fit has explained, that is, the coefficient of determination can be computed as:

$$\frac{\mathrm{Var}[y] - S(a,b)}{\mathrm{Var}[y]} = 1 - \frac{S(a,b)}{\mathrm{E}[y^2] - \mathrm{E}[y]^2}$$
$$= \frac{(\mathrm{E}[x\,y] - \mathrm{E}[x]\,\mathrm{E}[y])^2}{(\mathrm{E}[x^2] - \mathrm{E}[x]^2)\,(\mathrm{E}[y^2] - \mathrm{E}[y]^2)}.$$

- Comparing to the correlation coefficient expression reveals that

$$\frac{\mathrm{Var}[y] - S(a,b)}{\mathrm{Var}[y]} = R^2.$$

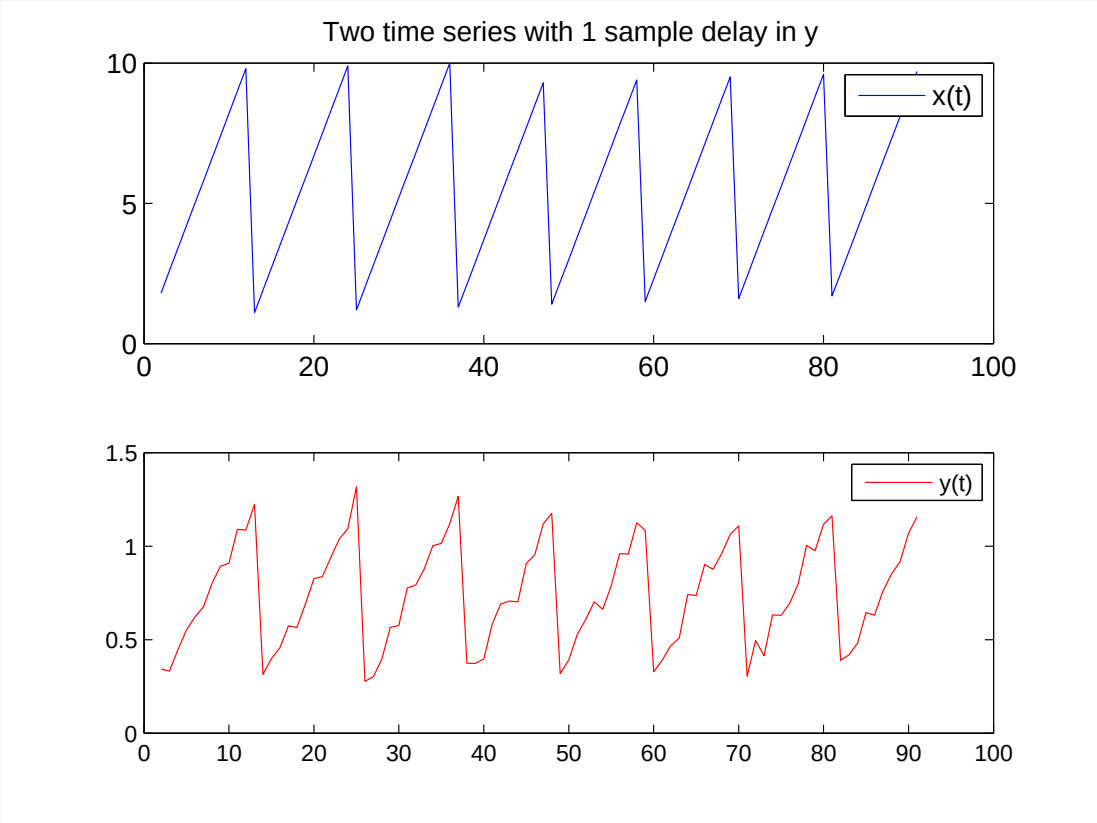
- That is, the coefficient of determination is the square of the correlation coefficient.
- This definition of correlation coefficient also works with non-linear and multidimensional models.

Cautions on Interpretation of Correlations

- Correlation does not imply causality!
- False conclusions: Using swimming suit correlates with drowning accidents $\Rightarrow$ using swimming suit causes drowning accidents.
- Correlation can be caused by latent (hidden) factor, which makes both variables change to the same direction.
- One must also be careful when making conclusions from correlations between physical phenomena.
- In addition, correlation only measures linear relationship between two variables.
- Variables can correlate much even if the variables are essentially constant - the scatter plot should always be checked visually.

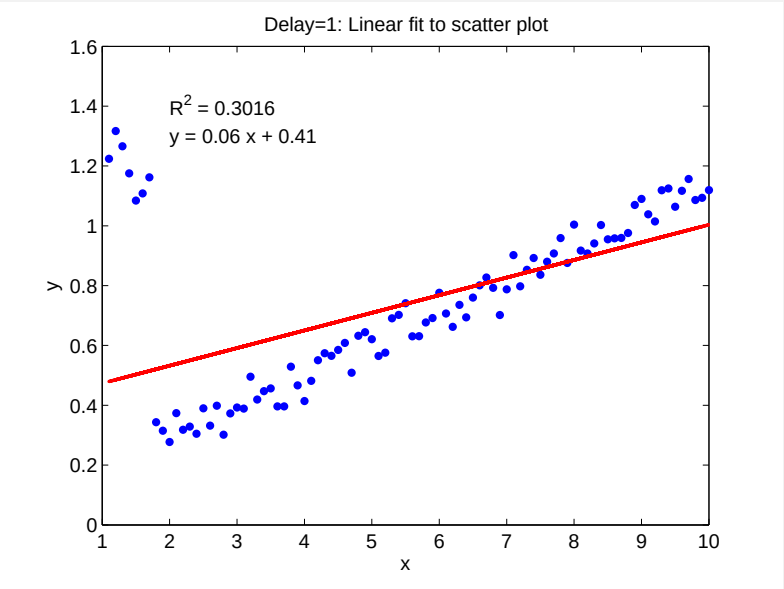
Effect of Delay to Correlations [1/2]

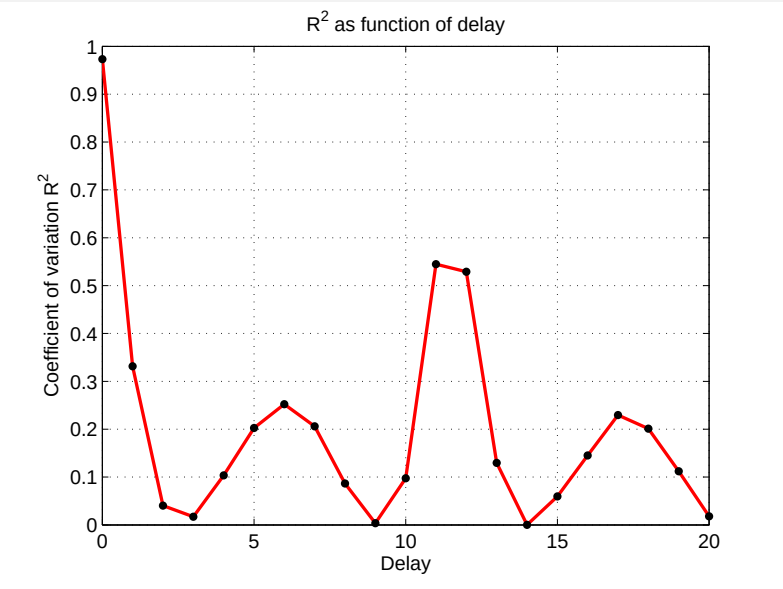
- Even small delay in inputs or outputs can destroy the correlation completely:



Effect of Delay to Correlations [2/2]

- With only 1 sample delay, the original correlation 0.97 decreases to 0.30!





Multidimensional linear models [1/3]

- Multidimensional generalization of linear regression model is

$$y = a_1 x_1 + a_2 x_2 + \ldots + a_d x_d + b$$

  where we measure $\{(x_1^{(i)}, \ldots, x_d^{(i)}, y^{(i)}) \ : \ i = 1, \ldots, n\}$.
- The mean squared error is now given as

$$S(a, b) = \frac{1}{n} \sum_i (y^{(i)} - a_1 x_1^{(i)} - a_2 x_2^{(i)} - \ldots - a_d x_d^{(i)} - b)^2.$$

- We could now solve the parameters by setting the derivatives to zero as in scalar case.
- However, more intuitive way is to first rewrite the model in a suitable matrix form:

Multidimensional linear models [2/3]

- We define the following:

$$\mathbf{H} = \begin{pmatrix} x_1^{(1)} & x_2^{(1)} & \dots & x_d^{(1)} & 1 \\ x_1^{(2)} & x_2^{(2)} & \dots & x_d^{(2)} & 1 \\ & & \vdots & & \\ x_1^{(n)} & x_2^{(n)} & \dots & x_d^{(n)} & 1 \end{pmatrix}, \mathbf{Y} = \begin{pmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{pmatrix}, \boldsymbol{\theta} = \begin{pmatrix} a_1 \\ \vdots \\ a_d \\ b \end{pmatrix}.$$

- The model and the mean squared error can be now written as:

$$\mathbf{Y} = \mathbf{H}\boldsymbol{\theta}$$

$$S(\boldsymbol{\theta}) = \frac{1}{n}(\mathbf{Y} - \mathbf{H}\boldsymbol{\theta})^T (\mathbf{Y} - \mathbf{H}\boldsymbol{\theta}).$$

Multidimensional linear models [3/3]

- The gradient of error should vanish at minimum:

$$\nabla S(\theta) = \frac{1}{n}[-2\mathbf{H}^T\mathbf{Y} + 2\mathbf{H}^T\mathbf{H}\theta] = 0.$$

- The resulting least squares estimate is

$$\theta = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{Y}.$$

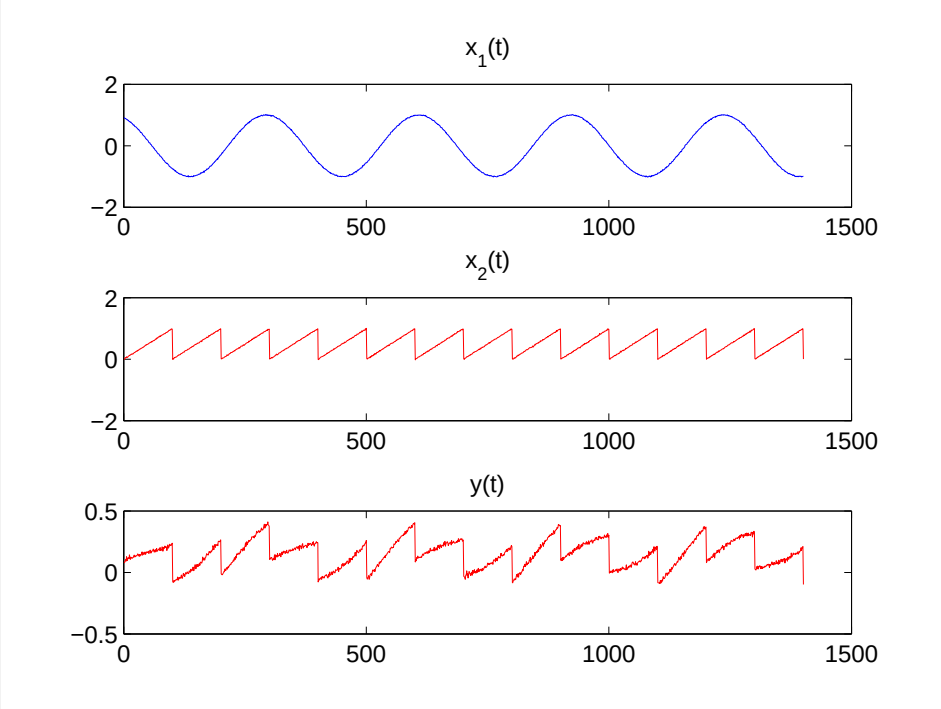
- Correlation coefficient is not applicable, but the coefficient of determination R^2 can still be computed (if y is scalar):

$$R^2 = \frac{\mathrm{Var}[y] - S(\theta)}{\mathrm{Var}[y]}.$$

- If y is a vector, R^2 can be computed for each component separately.

Example: Dependence of Three Signals [1/3]

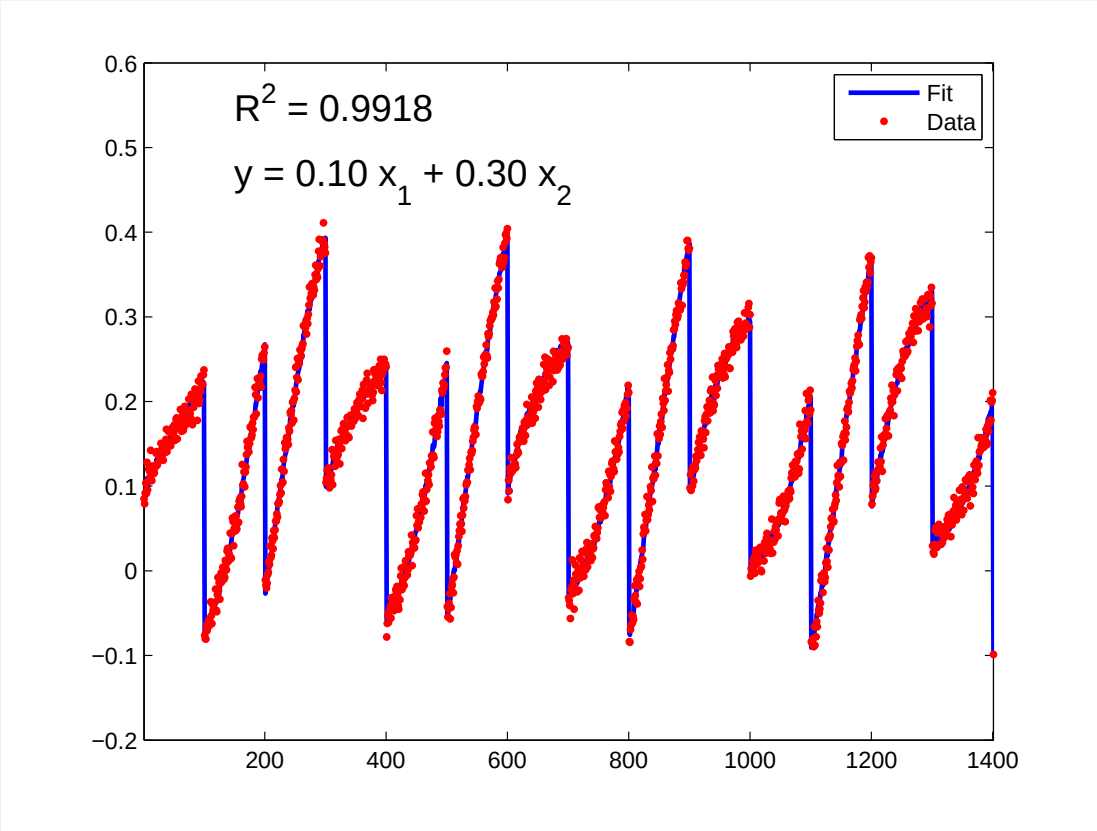
- We want to find out the relationship between the following signals:



- The 1d-correlations are: $R^2(x_1, x_2) = 0.00$, $R^2(x_1, y) = 0.36$ and $R^2(x_2, y) = 0.57$.

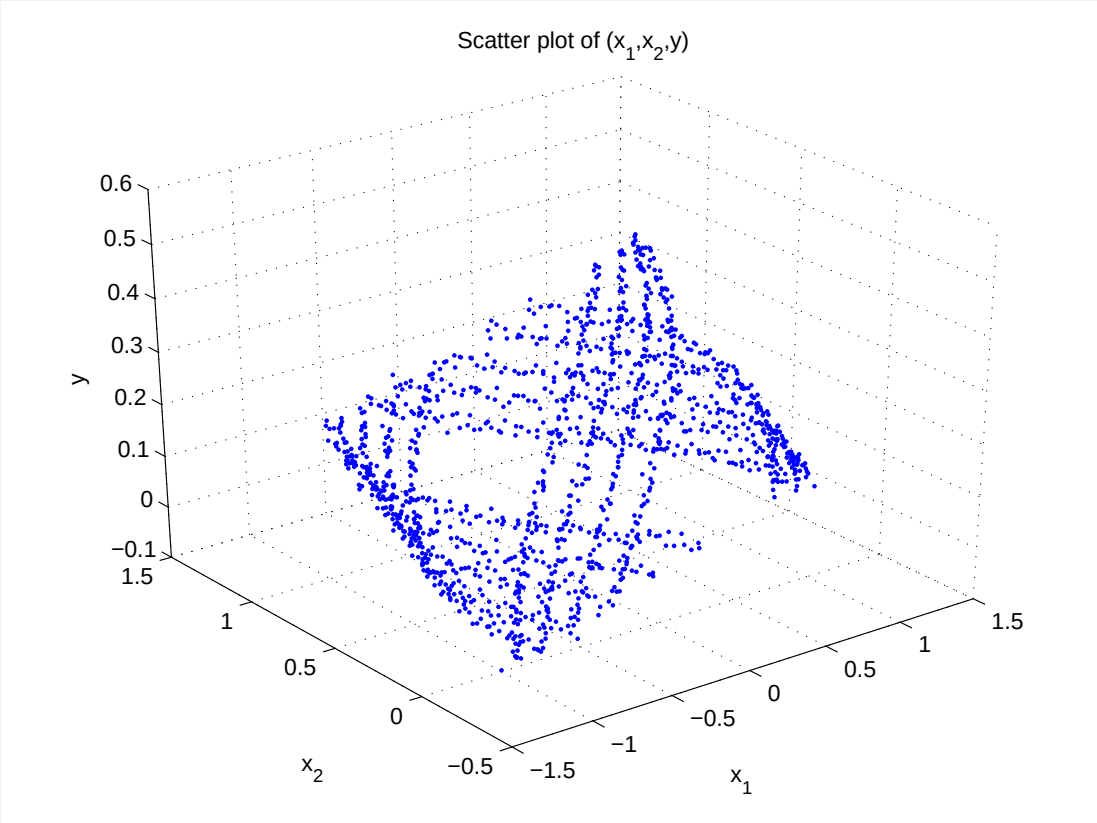
Example: Dependence of Three Signals [2/3]

- With 2-dimensional linear model $y = a_1 x_1 + a_2 x_2$ we get the following fit:



Example: Dependence of Three Signals [3/3]

- The scatter plot of (x_1, x_2, y) reveals that the time series lie on the same plane in 3d:



Linear-in-Parameters Models

- The multidimensional linear model framework also applies to linear-in-parameters models such as polynomial models

$$y = a_1 x + a_2 x^2 + \ldots + a_d x^d + b,$$

- or other models of the form

$$y = a_1 f_1(x) + a_2 f_2(x) + \ldots + a_d f_d(x) + b.$$

- For example, Fourier series can be derived from the least squares framework.
- Although the approximating function is non-linear, these are still called linear models because the parameters appear linearly.

Nonlinear Models [1/3]

- The advantage of linearity is that it yields to easy mathematics.
- But, linearity can be a restriction in the modeling point of view.
- We can also use general non-linear models of the form

$$y = f(\mathbf{x}; \boldsymbol{\theta}),$$

  where $\boldsymbol{\theta} \in \mathbb{R}^n$ is a vector of parameters and $\mathbf{x} = (x_1, ..., x_d)$.
- The non-linearity can, for example, result from modeling a non-linearity in the physical system under consideration.
- Non-physically based non-linear regression models are also used, e.g., multi layer perceptron (MLP) neural network:

$$y = \mathbf{W} \tanh(\mathbf{A}\,\mathbf{x} + \mathbf{b}) + c.$$

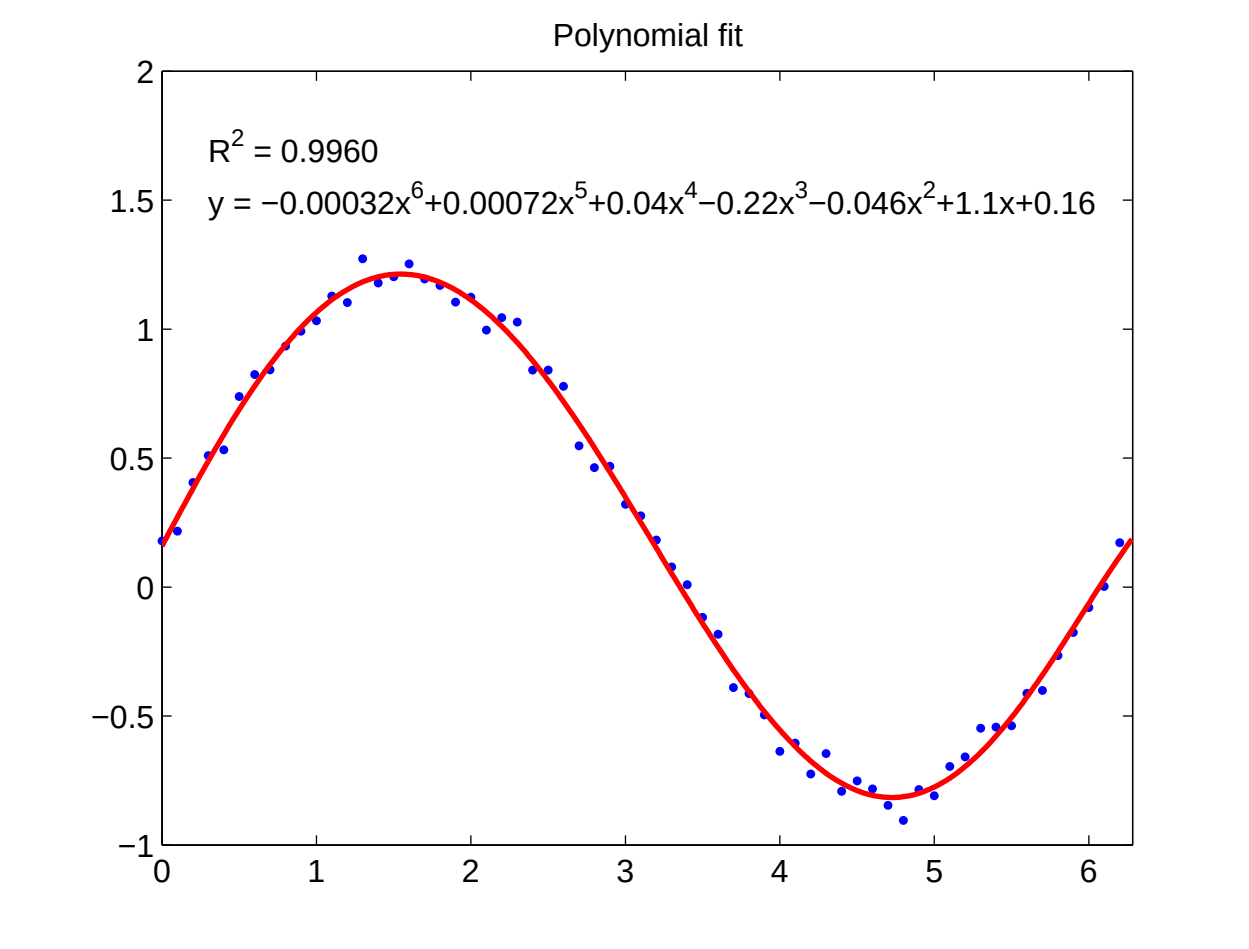
Nonlinear Models [2/3]

- In principle, least squares can be used for fitting parameters of non-linear models ("training" the model).
- The error function needs to be minimized using iterative methods, because closed form minimization is not possible.
- For example, simple gradient descend method can be used (called back-propagation training in case of MLP's).
- More advanced optimization methods such as conjugate gradient, Levenberg-Marquardt, simulated annealing, and genetic algorithms can be used as well.
- In practice, non-linear models easily suffer from over-fitting, and regularization and Bayesian methods are needed.

Nonlinear Models [3/3]

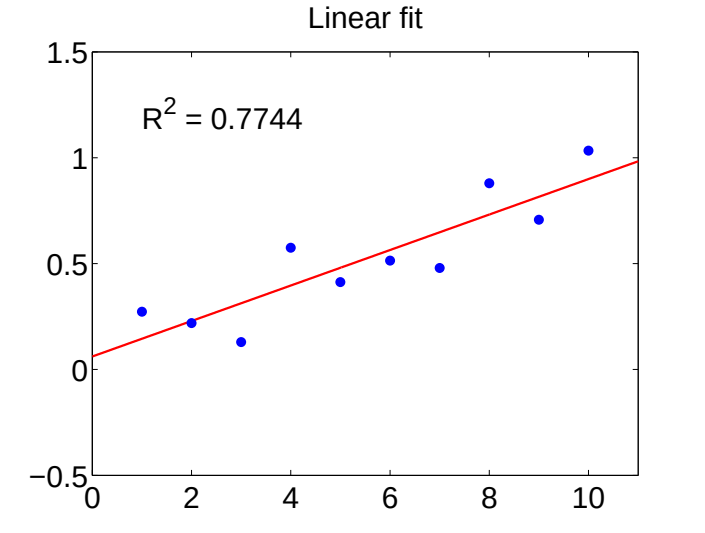
- Often non-linear problem can be converted into a linear one by using so called kernel trick.
- The idea is to map the input space to a feature space (reproducing kernel Hilbert space, RKHS), which is linear with respect to outputs.
- This method is often used in machine leaning and leads to support vector machines (SVM) and other kernel methods.
- Gaussian process (GP) regression models, which nowadays are often used instead of MLP's are also kinds of kernel methods.

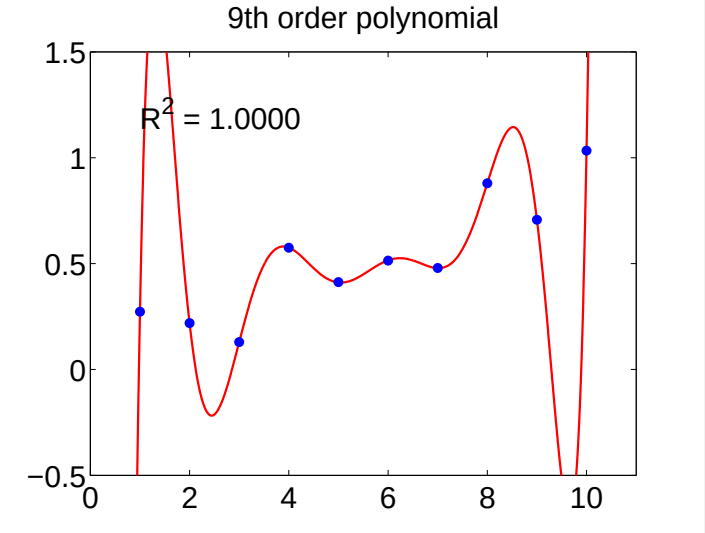
Example: Approximation of Sine



Over-fitting and Regularization [1/3]

- Consider the following alternative fits to the same data:





- The polynomial on the right is said to be over-fitted to the data.
- If there are many free parameters compared to effective number of data points the over-fitting behavior occurs.
- With 1d-models over-fitting is easy to “see”, but with high-dimensional models much harder to detect.

Over-fitting and Regularization [2/3]

- Over-fitting problem is the worst with multi-parameter nonlinear models such as neural network models.
- Even linear models can be ill-conditioned, which indicated that there are few linearly independent data points.
- A common solution to the problem is regularization, where an additional cost function $C(\theta)$ is added to the least squares error:

$$S_r(\theta) = S(\theta) + C(\theta).$$

- For example, the L^2 norm of the parameters can be penalized:

$$S_r(\theta) = \frac{1}{n}(\mathbf{Y} - \mathbf{H}\,\theta)^T(\mathbf{Y} - \mathbf{H}\,\theta) + \lambda\,|\theta|^2.$$

Over-fitting and Regularization [3/3]

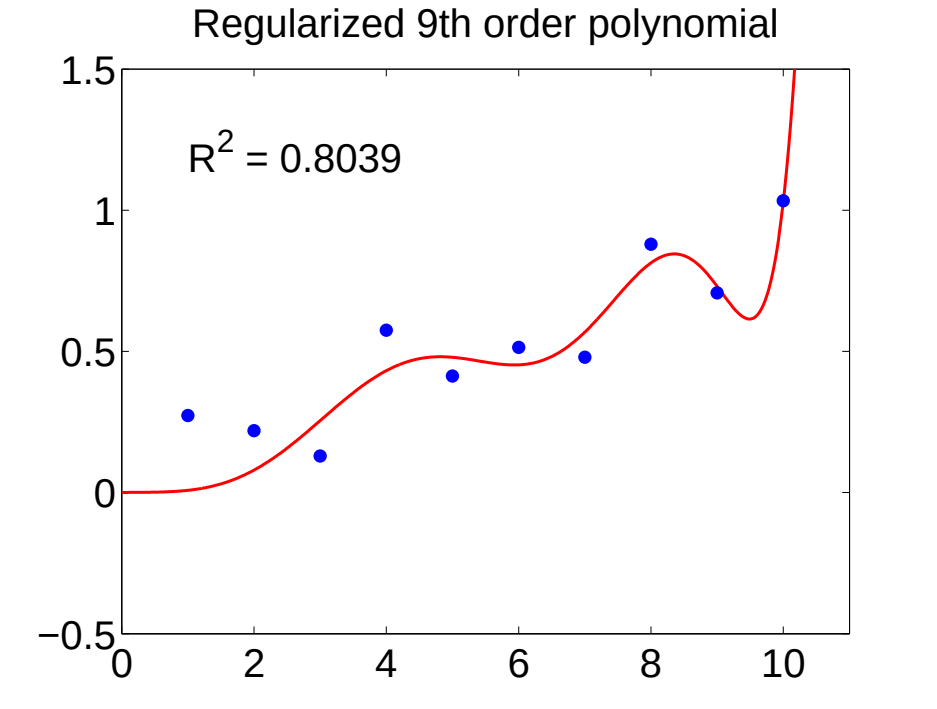
- The parameter λ can be used for tuning the effective order of the polynomial from say 10 to 0 ($\lambda = 0, \ldots, \infty$).
- It is also possible to optimize the parameter λ using information criteria (AIC, BIC, DIC, ...) or by using cross-validation.
- The polynomial order can be also used as a regularization parameter as such and estimated by information criteria or cross-validation.
- In the case of MLP's the number of hidden units (and parameters) can be similarly used as a regularization parameter.
- A general class of cost terms are the Tikhonov regularizers:

$$C(\boldsymbol{\theta}) = \int |\mathcal{L} f(\mathbf{x}; \boldsymbol{\theta})|^2 d\mathbf{x},$$

where $\mathcal{L}$ is a linear operator (e.g. differential or integral operator).

Example: Regularization of Polynomial Fit

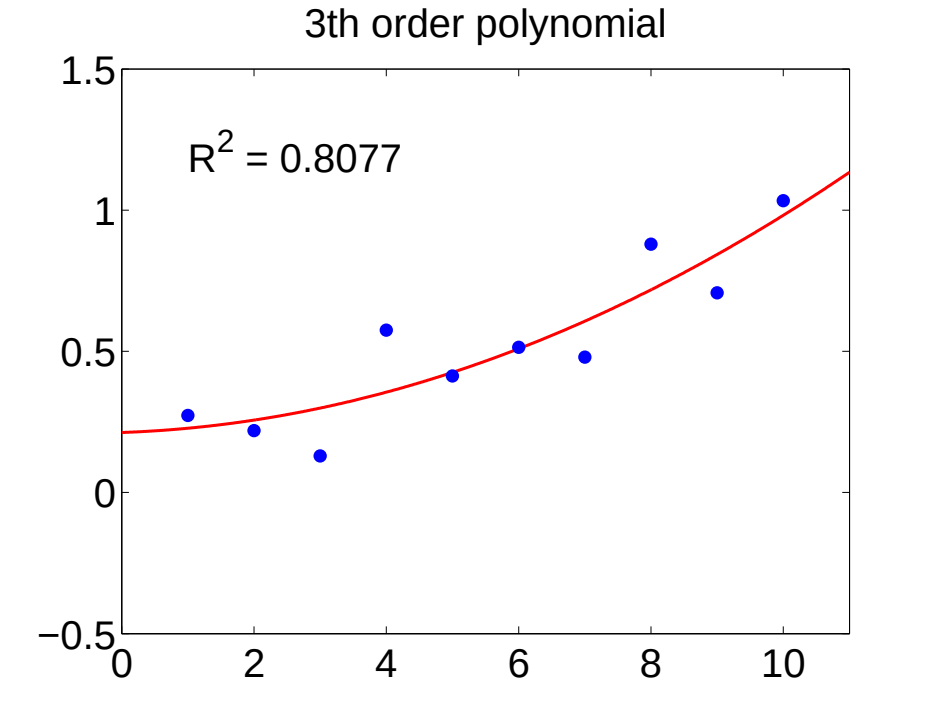
- Returning to the previous example, if we add the regularization term $\lambda|\theta|^2$ to the cost function of 9th order polynomial we get:



- In this case this kind of regularization does not help much.

Example: Regularization of Polynomial Fit (cont)

- Using cross-validation with respect to polynomial order would indicate that the following 3rd order polynomial is the best:



- Actually, the data is generated from linear model, but it is impossible to deduce it from the data.

Cautions on Practical Use of Non-Linear Models

- No advanced regression model or model class (e.g. NN) can be used for every regression task and no fancy computational method can replace good physics (or chemistry) based model.
- Neural Network models such as Multi Layer Perceptron (MLP) sound much more advanced than they really are.
- Including non-linear terms to regression models without physical or other a priori knowledge basis is often a bad idea - almost always linear models can be used instead.
- If the physics of the phenomenon are known, one must not assume that the method (such as neural network) would "learn" the phenomenon from the data.
- Instead, if anything about the mathematical form of the phenomenon is known a priori e.g. from physics, the knowledge should be explicitly included into the model.

Input Selection in Multi-Linear Models

- In input selection we try to find the subset inputs $\mathbf{x}$ that are most relevant in predicting the target values $\mathbf{y}$.
- That is, we try to find out, e.g., which of the following models is the best:

$$y = a_1 x_1 + a_2 x_2 + b$$
$$y = a_1 x_1 + a_3 x_3 + b$$
$$y = a_2 x_2 + a_3 x_3 + b$$
$$y = a_1 x_1 + a_2 x_2 + a_3 x_3 + b.$$

- In case of multi-linear models there exists good methods for input selection and it is even possible to take the noise in $\mathbf{x}$ into account.
- However, including more inputs always improves the fit and a separate cost term needs to be used for penalizing the number of inputs.

PCA and PLS

- If inputs are not independent, it is possible to try to find lower dimensional latent factors that explain the variations.
- Principal component analysis (PCA) finds the latent factors that explain the most variance in inputs.
- Partial least squares (or Projection to latent structures) (PLS) finds the latent factors that best explain the outputs.
- PCA or PLS give no answer to how many latent factors we should select.
- PCA and PLS apply to linear models, because the latent factors are linear combinations of the inputs.
- The methodogy can be extended non-linear models by using the kernel trick.
- Certain other generalizations to non-linear models exist, but they typically only work in restricted special cases.

Model Selection [1/2]

- If we do not restrict model class to linear models, we have the general problem of model selection of which input selection is a special case.
- As mentioned, one way of measuring goodness of fit of a linear or non-linear model is the coefficient of determination:

$$R^2 = \frac{\text{Var}[y] - S(\theta)}{\text{Var}[y]}$$

- With fixed data set this is equivalent to comparing models with respect to the error function $S(\theta)$.
- The problem is that increasing the number of parameters always reduces the error and increases correlation.
- Due to this over-fitting dilemma, goodness of fit does not imply that the fit would make any sense.

Model Selection [2/2]

- To overcome this problem, several regularized goodness fit measures have been developed:
  - Information criteria such as AIC, BIC and DIC include additional cost to $S(\theta)$, which penalizes high number of parameters
  - Training & validation set based methods the error function minimization or other estimation is done with respect to training set and the "generalization ability" is tested using the validation or test set.
  - Cross-validation methods, where the validation procedure is included as part of the training procedure.
- None of the methods work in all the cases - the best way is to use knowledge of the physics behind as much as possible.

Maximum Likelihood

- The linear regression can be equivalently formulated as stochastic model

$$y^{(i)} = ax^{(i)} + b + e^{(i)}, \quad e^{(i)} \sim \mathrm{N}(0, \sigma^2).$$

- In probabilistic notation this is:

$$p(y^{(i)} \mid a, b) = \mathrm{N}(y^{(i)} \mid ax^{(i)} + b, \sigma^2),$$

where

$$\mathrm{N}(y \mid m, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(y-m)^2/(2\sigma^2)}.$$

- In maximum likelihood (ML) method we maximize the likelihood function

$$L(a, b) = \prod_i p(y^{(i)} | a, b).$$

Maximum Likelihood (cont)

- In linear regression ML leads to exactly the same result as least squares.
- In principle, non-linear and multidimensional models such as

$$p(y^{(i)} \mid \boldsymbol{\theta}) = \mathrm{N}(y^{(i)} \mid f(\mathbf{x}^{(i)}; \boldsymbol{\theta}), \sigma^2),$$

  can be handled similarly.
- Also in principle, we could treat the noise variance as an unknown variable:

$$p(y^{(i)} \mid \boldsymbol{\theta}, \sigma^2) = \mathrm{N}(y^{(i)} \mid f(\mathbf{x}^{(i)}; \boldsymbol{\theta}), \sigma^2),$$

  and estimate it by maximum likelihood.
- But over-fitting will cause problems and there is no way of including regularization to the ML-framework - for that we need the Bayesian data analysis.

Bayesian Data Analysis

- In Bayesian analysis, also the parameters are considered as random variables with a prior distribution:

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta}).$$

- This prior distribution can be, for example, multidimensional Gaussian:

$$p(\boldsymbol{\theta}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\boldsymbol{\theta}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}-\boldsymbol{\mu})\right).$$

- The measurements are modeled in the same manner as in ML-estimation, e.g.:

$$p(y^{(i)} \mid \boldsymbol{\theta}) = \mathrm{N}(y^{(i)} \mid f(\mathbf{x}^{(i)}; \boldsymbol{\theta}), \sigma^2).$$

- The joint distribution of all the measurements is now

$$p(\mathbf{Y} \mid \boldsymbol{\theta}) = \prod_i p(y^{(i)} \mid \boldsymbol{\theta}).$$

where $\mathbf{Y} = \{y^{(1)}, \ldots, y^{(n)}\}$.

Posterior Distribution [1/3]

- Instead of maximizing the likelihood $L(\boldsymbol{\theta}) = p(\mathbf{Y} \mid \boldsymbol{\theta})$, we compute the posterior distribution of parameters using the Bayes' rule:

$$p(\boldsymbol{\theta} \mid \mathbf{Y}) = \frac{p(\mathbf{Y} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int p(\mathbf{Y} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})\, d\boldsymbol{\theta}}.$$

- The normalization factor in denominator does not depend on $\boldsymbol{\theta}$ and it is common to just write

$$p(\boldsymbol{\theta} \mid \mathbf{Y}) \propto p(\mathbf{Y} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta}).$$

- We may now e.g. compute the expected value (MMSE estimate) of the parameter $\boldsymbol{\theta}$:

$$\mathrm{E}[\boldsymbol{\theta} \mid \mathbf{Y}] = \int_{\mathbb{R}^n} \boldsymbol{\theta}\, p(\boldsymbol{\theta} \mid \mathbf{Y})\, d\boldsymbol{\theta}.$$

- Other possible estimates are e.g. the maximum a posteriori (MAP) estimate or the median of the posterior distribution.

Posterior Distribution [2/3]

- The Bayesian version of linear regression model is:

$$p(y^{(i)} \mid a, b) = \mathrm{N}(y^{(i)} \mid ax^{(i)} + b, \sigma^2)$$
$$p(a) = \mathrm{N}(a \mid 0, \sigma_a^2)$$
$$p(b) = \mathrm{N}(b \mid 0, \sigma_b^2).$$

- The posterior distribution is now of the form

$$p(a, b \mid y^{(1)}, \ldots, y^{(n)}) = \mathrm{N}\left(\begin{pmatrix} a \\ b \end{pmatrix} \,\middle|\, \mathbf{m}, \mathbf{\Sigma} \right),$$

where

  - The posterior mean $\mathbf{m}$ is the L^2-regularized least squares solution.
  - The posterior covariance $\mathbf{\Sigma}$ is the covariance of the error in the mean.

Posterior Distribution [3/3]

- The Bayesian solution also automatically produces estimate of the error in the used estimator (e.g., posterior mean).
- In Bayesian framework, regularization can be implemented by using suitable prior distributions for the parameters.
- The maximum likelihood (ML) method is obtained as a limiting special case when prior distributions become flat (i.e., non-informative).
- The non-regularized and regularized least squares (LS) solutions are also special cases of the framework.
- However, the true advantage of the framework is in models, which cannot be formulated in LS- and ML-frameworks at all.
- Examples of such models are hierarchical models.

Hierarchical Models

- In Hierarchical models prior distributions (regularizers) are specified for unknown noise variance parameters.
- The linear regression model was of the following form

$$p(y^{(i)} \mid a, b) = \mathrm{N}(y^{(i)} \mid ax^{(i)} + b, \sigma^2)$$
$$p(a) = \mathrm{N}(a \mid 0, \sigma_a^2)$$
$$p(b) = \mathrm{N}(b \mid 0, \sigma_b^2).$$

- In this model we assume that the variances σ^2, σ_a^2 and σ_b^2 are exactly known - but if they are not, we need hierarchical models.

Hierarchical Models (cont)

- The parameters σ^2, σ_a^2 and σ_b^2 can be included into model:

$$
\begin{aligned}
p(y^{(i)} \mid a, b, \sigma^2) &= \mathrm{N}(y^{(i)} \mid ax^{(i)} + b, \sigma^2) \\
p(a) &= \mathrm{N}(a \mid 0, \sigma_a^2) \\
p(b) &= \mathrm{N}(b \mid 0, \sigma_b^2) \\
p(\sigma^2) &= \text{Inv-Gamma}(\sigma^2 \mid \alpha, \beta) \\
p(\sigma_a^2) &= \text{Inv-Gamma}(\sigma_a^2 \mid \alpha_a, \beta_a) \\
p(\sigma_b^2) &= \text{Inv-Gamma}(\sigma_b^2 \mid \alpha_b, \beta_b).
\end{aligned}
$$

- The variance parameters are called hyper-parameters and the model is now a hierarchical model.
- The Inv-Gamma distributions are typically used here, because then the posterior distribution is analytically computable.

Marginalization of Hyper-parameters

- The Bayesian Data Analysis would proceed to analyzing the joint distribution of all the parameters:

$$p(a, b, \sigma^2, \sigma_a^2, \sigma_b^2 \,|\, \mathbf{Y}).$$

- If we are not interested in actual values of the noise variances, we may integrate them out:

$$p(a, b \,|\, \mathbf{Y}) = \int p(a, b, \sigma^2, \sigma_a^2, \sigma_b^2 \,|\, \mathbf{Y})\, d(\sigma^2)\, d(\sigma_a^2)\, d(\sigma_b^2).$$

- Watch out for the notation - this looks the same as the posterior with fixed variances, but it is not the same!
- In the linear regression case, this marginal posterior distribution is a Student's T-distribution.

Typically Used Distribution Models

- Gaussian prior and noise models with linear or non-linear regression functions.
- Mixture Gaussian models for modeling non-Gaussian phenomena.
- Inverse-Gamma, Inv-χ^2, and other such distributions for noise variances.
- Inverse-Wishart distributions for covariance matrices.
- Multinomial and Poisson distributions for discrete variables.
- Dirichlet and Gamma distributions for parameters of multinomial and Poisson models.
- Gaussian processes and random fields as function space priors in spatial analysis and signal processing.

Gaussian, Monte Carlo and Other Approximations

- The computation of the expected value required evaluation of multidimensional integrals:

$$\begin{aligned} \mathrm{E}[\boldsymbol{\theta} \mid \mathbf{Y}] &= \int_{\mathbb{R}^n} \boldsymbol{\theta}\, p(\boldsymbol{\theta} \mid \mathbf{Y})\, d\boldsymbol{\theta} \\ &= \frac{\int_{\mathbb{R}^n} \boldsymbol{\theta}\, p(\mathbf{Y} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})\, d\boldsymbol{\theta}}{\int_{\mathbb{R}^n} p(\mathbf{Y} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})\, d\boldsymbol{\theta}}. \end{aligned}$$

- Computation of other statistics is even harder than computation of the expectation.
- The closed form solution exists only for special cases, namely linear Gaussian models.
- The integrals are particularly problematic, when $\boldsymbol{\theta}$ is very high dimensional (tens or hundreds of parameters), as typically is the case.

Gaussian, Monte Carlo and Other Approximations (cont)

- In Gaussian or Laplace approximations, the posterior distribution is approximated with a multidimensional Gaussian distribution.
- Gaussian mixtures can be used for approximating the posterior as a weighted sum of Gaussian distributions.
- Gaussian mixtures can be fitted by the expectation maximization (EM) algorithm.
- Monte Carlo methods use a finite set of samples for approximating the posterior distribution.
- The most common types of Monte Carlo methods are Markov chain Monte Carlo (MCMC) methods (Metropolis-Hastings methods) and importance sampling (IS) based methods.
- Variational and minimum free energy approximations.

What is Different in Estimation of Dynamic Processes?

- In conventional offline or "batch" estimation we first get the measurements, then we estimate.
- In online or "recursive" estimation we get measurements continuously and compute new estimates continuously.
- The parameters do not need to be constant, but they can be modeled as stochastic process $\theta(t)$, which changes in time, e.g.:
  - Discrete-time random walk:

$$\theta_k = \theta_{k-1} + w_{k-1}.$$

  - Continuous-time random walk (Brownian motion):

$$d\theta/dt = w(t).$$

  - General (Hidden) Markov models:

$$\theta_k \sim p(\theta_k \,|\, \theta_{k-1}).$$

Batch Bayesian Estimation

1. Collect the data:

$$\mathbf{Y} = \{(x_1, \mathbf{y}_1), \ldots, (x_T, \mathbf{y}_T)\}.$$

2. Specify the likelihood model:

$$p(\mathbf{y}_k \,|\, \theta)$$

3. Specify the prior distribution:

$$p(\theta).$$

4. Compute the posterior distribution:

$$p(\theta|\mathbf{Y}) = \frac{1}{Z} p(\theta) \prod_k p(\mathbf{y}_k \,|\, \theta).$$

5. Use numerical methods for approximating the distribution and its statistics.

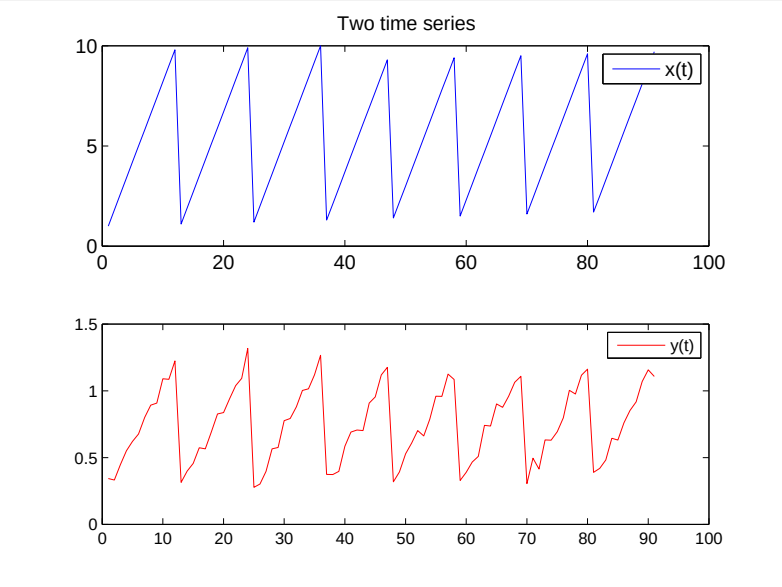
Recursive Bayesian Estimation

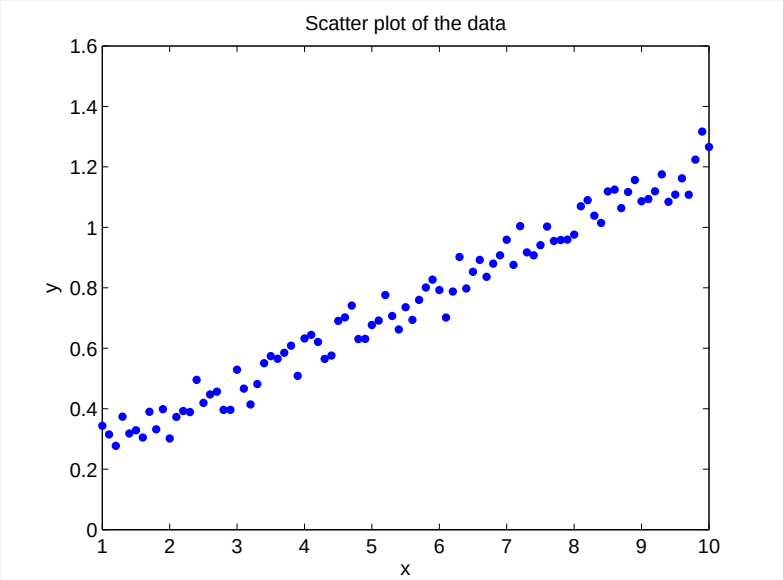
- In recursive estimation the likelihood, and prior are specified in the same way, but the data is obtained incrementally $\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \ldots$
- The posterior distribution is computed recursively as follows:

$$
\begin{aligned}
p(\theta|\mathbf{y}_1) &= \frac{1}{Z_1} p(\mathbf{y}_1 \,|\, \theta) p(\theta) \\
p(\theta|\mathbf{y}_{1:2}) &= \frac{1}{Z_2} p(\mathbf{y}_2 \,|\, \theta) p(\theta \,|\, \mathbf{y}_1) \\
&\vdots \\
p(\theta|\mathbf{y}_{1:T}) &= \frac{1}{Z_T} p(\mathbf{y}_T \,|\, \theta) p(\theta \,|\, \mathbf{y}_{1:T-1}).
\end{aligned}
$$

- At each stage, the result is the same as what would be obtained by computing the corresponding batch solution.

Bayesian Batch Linear Regression [1/4]





Bayesian Batch Linear Regression [2/4]

- Measurement data:

$$\mathbf{Y} = \{(x_1, y_1), \ldots, (x_T, y_T)\}.$$

- Likelihood (σ^2 given):

$$p(y_k \mid \theta) = \mathrm{N}(y_k \mid ax_k + b, \sigma^2).$$

- Prior (known $\mathbf{m}_0$ and $\mathbf{P}_0$):

$$p(\theta) = \mathrm{N}(\theta \mid \mathbf{m}_0, \mathbf{P}_0).$$

Bayesian Batch Linear Regression [3/4]

- Because $p(\theta)$ and $p(\theta \mid \mathbf{Y})$ are Gaussian, we get

$$p(\theta \mid \mathbf{Y}) = \mathrm{N}(\theta \mid \mathbf{m}_T, \mathbf{P}_T).$$

- Posterior mean and covariance:

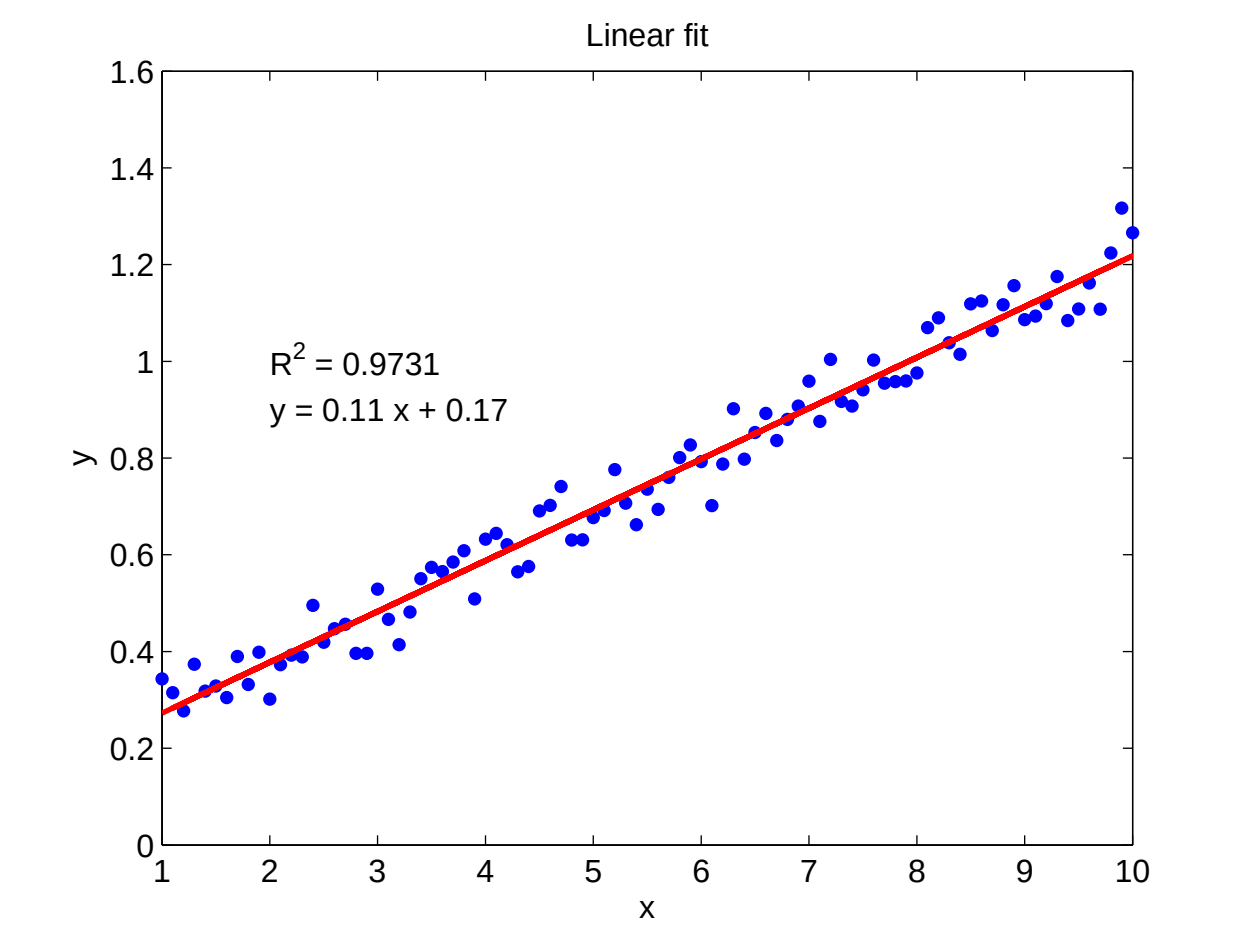
$$\mathbf{m}_T = \left[\mathbf{P}_0^{-1} + \frac{1}{\sigma^2}\mathbf{H}^T\mathbf{H}\right]^{-1}\left[\frac{1}{\sigma^2}\mathbf{H}^T\mathbf{y} + \mathbf{P}_0^{-1}\mathbf{m}_0\right]$$

$$\mathbf{P}_T = \left[\mathbf{P}_0^{-1} + \frac{1}{\sigma^2}\mathbf{H}^T\mathbf{H}\right]^{-1}.$$

- Here $\mathbf{H}$ and $\mathbf{y}$ are

$$\mathbf{H} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_T \end{pmatrix} \qquad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_T \end{pmatrix}.$$

Bayesian Batch Linear Regression [4/4]



Bayesian Recursive Linear Regression [1/4]

- Distribution given measurements $1, \ldots, t-1$:

$$p(\boldsymbol{\theta} \mid y_{1:k-1}) = \mathrm{N}(\boldsymbol{\theta} \mid \mathbf{m}_{k-1}, \mathbf{P}_{k-1}).$$

- Given another measurement y_k the posterior distribution is

$$p(\boldsymbol{\theta} \mid y_{1:k}) \propto p(y_k \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid y_{1:k-1}).$$

- Posterior distribution is Gaussian

$$p(\boldsymbol{\theta} \mid y_{1:k}) = \mathrm{N}(\boldsymbol{\theta} \mid \mathbf{m}_k, \mathbf{P}_k).$$

Bayesian Recursive Linear Regression [2/4]

- Gaussian distribution parameters are

$$\mathbf{m}_k = \left[\mathbf{P}_{k-1}^{-1} + \frac{1}{\sigma^2}\mathbf{H}_k^T\mathbf{H}_k\right]^{-1}\left[\frac{1}{\sigma^2}\mathbf{H}_k^T y_k + \mathbf{P}_{k-1}^{-1}\mathbf{m}_{k-1}\right]$$

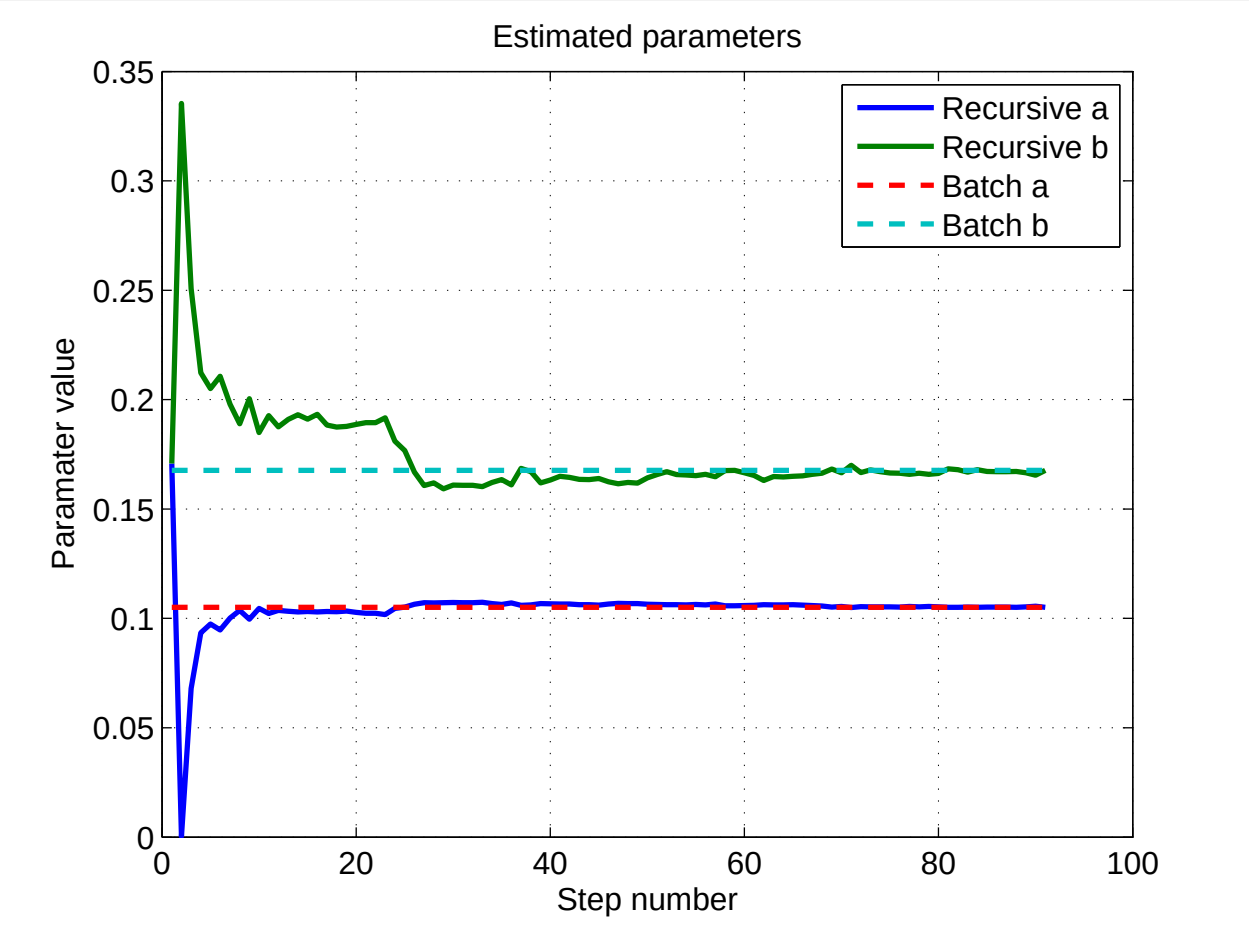
$$\mathbf{P}_k = \left[\mathbf{P}_{k-1}^{-1} + \frac{1}{\sigma^2}\mathbf{H}_k^T\mathbf{H}_k\right]^{-1}.$$

  where $\mathbf{H}_k = (x_k \; 1)$.

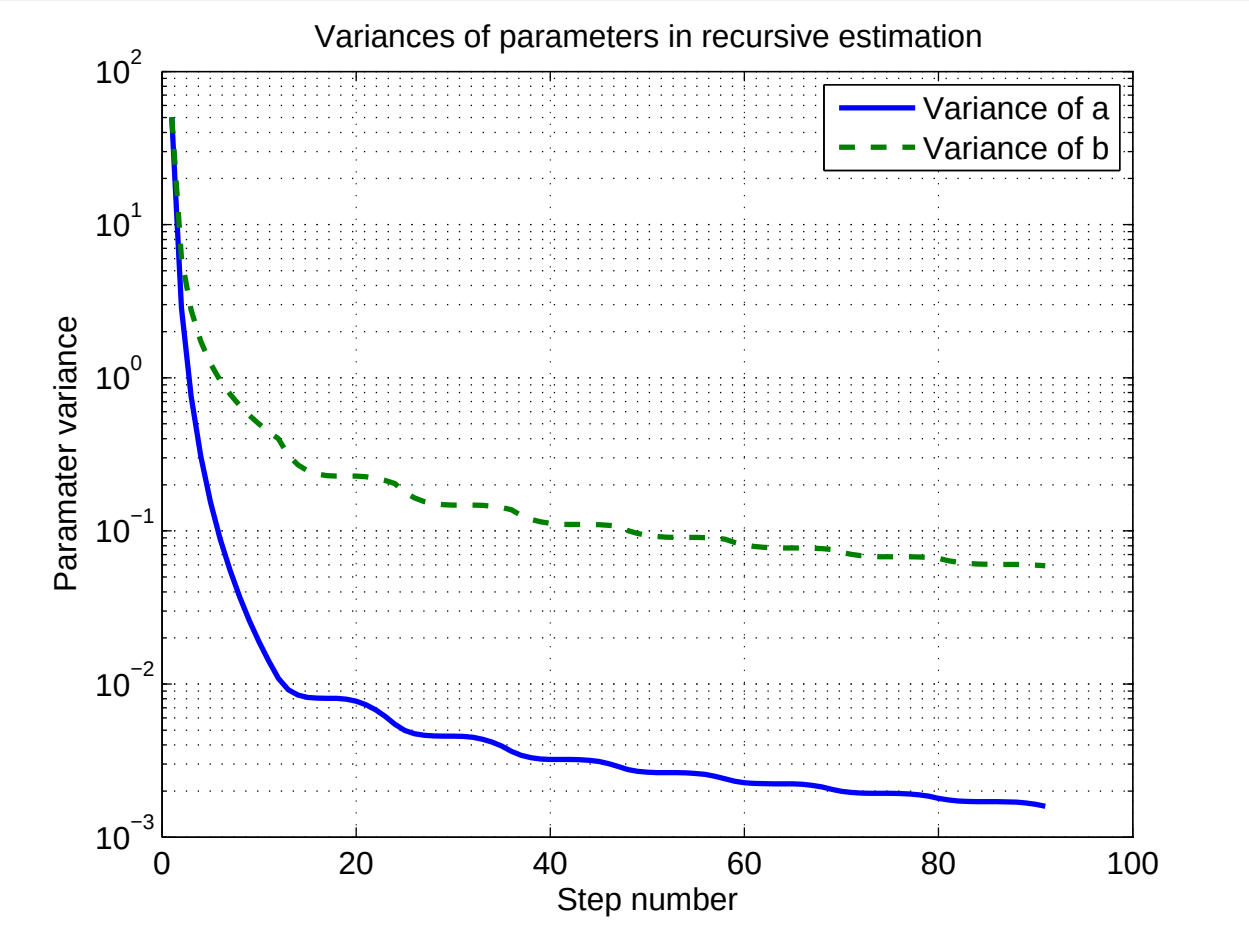
- Solution can be simplified by Matrix inversion lemma

$$\mathbf{P}_k = \mathbf{P}_{k-1} - \mathbf{P}_{k-1}\mathbf{H}_k^T\left[\mathbf{H}_k\mathbf{P}_{k-1}\mathbf{H}_k^T + \sigma^2\right]^{-1}\mathbf{H}_k\mathbf{P}_{k-1}.$$

Bayesian Recursive Linear Regression [3/4]



Bayesian Recursive Linear Regression [4/4]



Kalman Filtering [1/7]

- If the phenomenon is not constant in time, we may assume that parameters θ are not constant, but stochastic processes.
- Assume additional random walk model for θ_k:

$$\theta_k = \theta_{k-1} + \mathbf{w}_k$$

where $\mathbf{w}_k$ is 2d Gaussian random variable.

- The whole model is now of the form

$$\begin{aligned} p(y_k \,|\, \theta_k) &= N(y_k \,|\, a_k\, x_k + b_k, \sigma^2) \\ p(\theta_k \,|\, \theta_{k-1}) &= N(\theta_k \,|\, \theta_{k-1}, \mathbf{Q}) \\ p(\theta_0) &= N(\theta_0 \,|\, \mathbf{m}_0, \mathbf{P}_0). \end{aligned}$$

- $\mathbf{Q}$ is the covariance of random walk.

Kalman Filtering [2/7]

- Assume that we know distribution

$$p(\theta_{k-1} \mid y_{1:k-1}) = N(\theta_{k-1} \mid \mathbf{m}_{k-1}, \mathbf{P}_{k-1}).$$

- The posterior distribution of θ_k is Gaussian

$$p(\theta_k \mid y_{1:k}) = N(\theta_k \mid \mathbf{m}_k, \mathbf{P}_k).$$

- The posterior computation can be now composed into two steps:
  1. On prediction step, we predict the mean and covariance from step $k-1$ to k.
  2. On update step, we condition the mean and covariance to the measurement y_k.

Kalman Filtering [3/7]

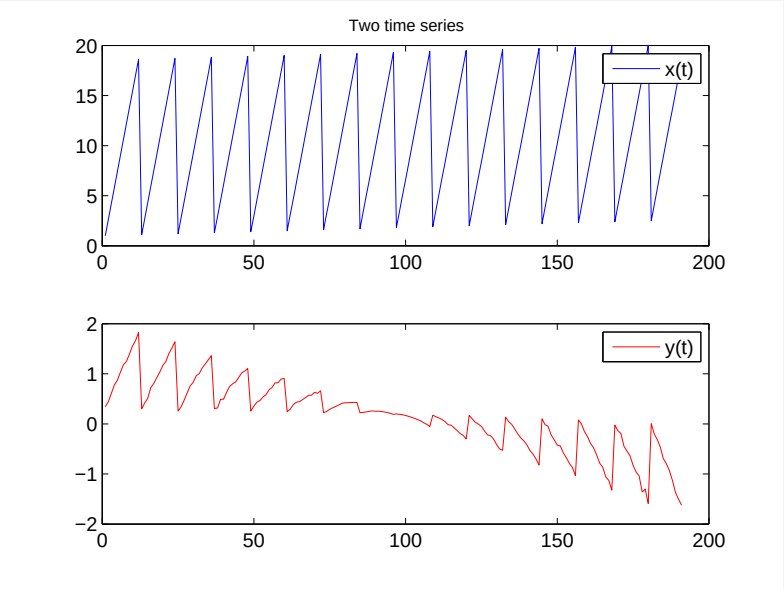
- Mean and covariance are given by
  - Prediction:

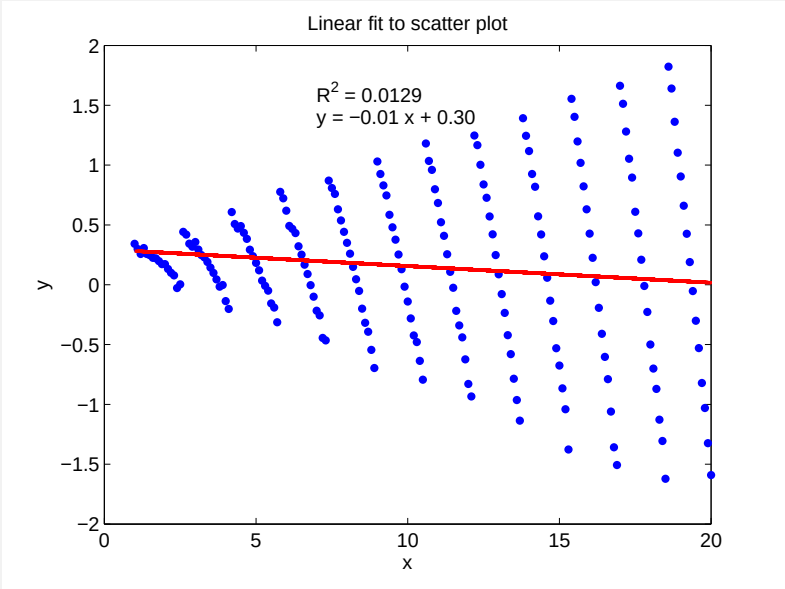
$$\begin{aligned} \mathbf{m}_k^- &= \mathbf{m}_{k-1} \\ \mathbf{P}_k^- &= \mathbf{P}_{k-1} + \mathbf{Q}. \end{aligned}$$

  - Update:

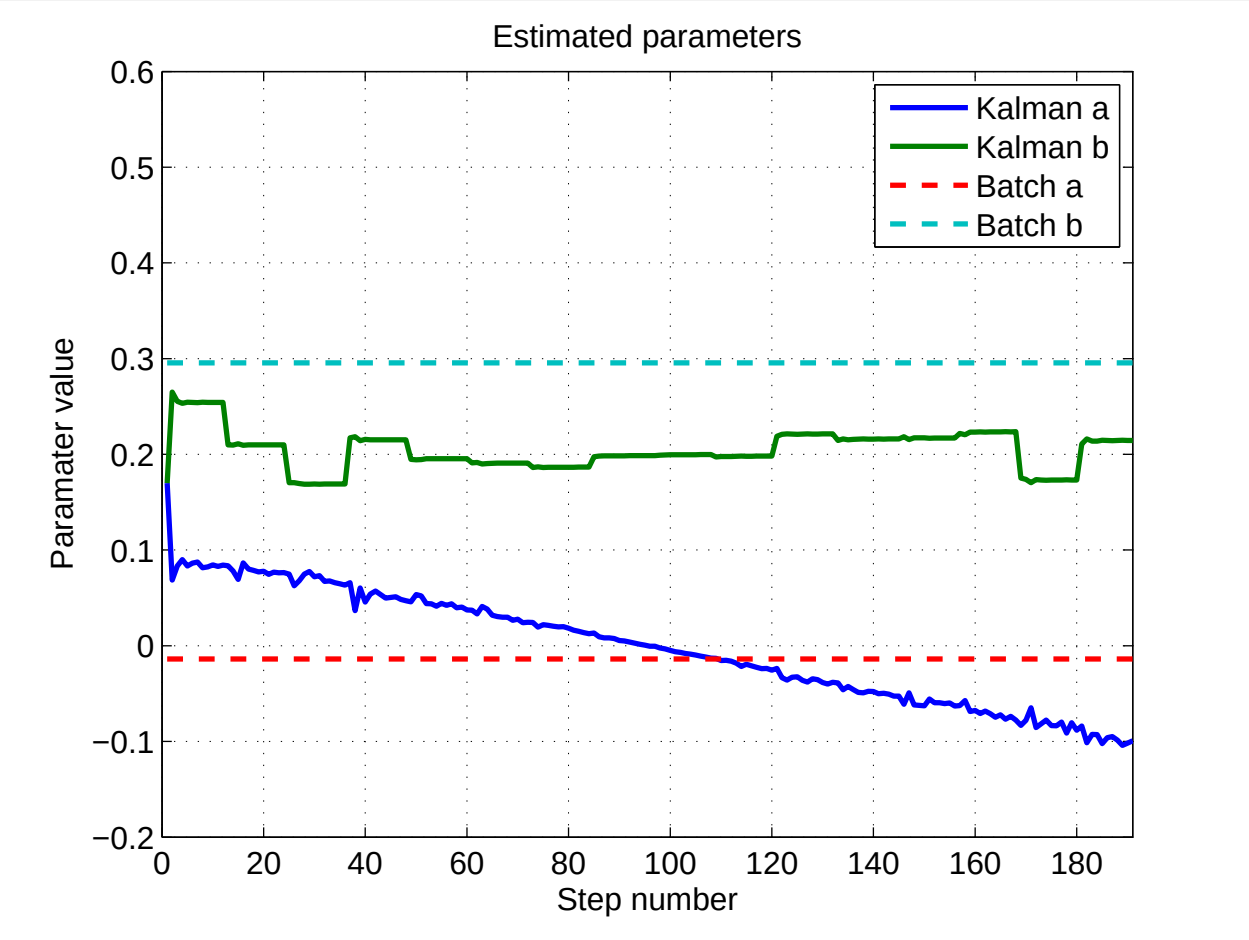
$$\begin{aligned} S_k &= \mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \sigma^2 \\ \mathbf{K}_k &= \mathbf{P}_k^- \mathbf{H}_k^T S_k^{-1} \\ \mathbf{m}_k &= \mathbf{m}_k^- + \mathbf{K}_k [y_k - \mathbf{H}_k \mathbf{m}_k^-] \\ \mathbf{P}_k &= \mathbf{P}_k^- - \mathbf{K}_k S_k \mathbf{K}_k^T. \end{aligned}$$

Kalman Filtering [4/7]

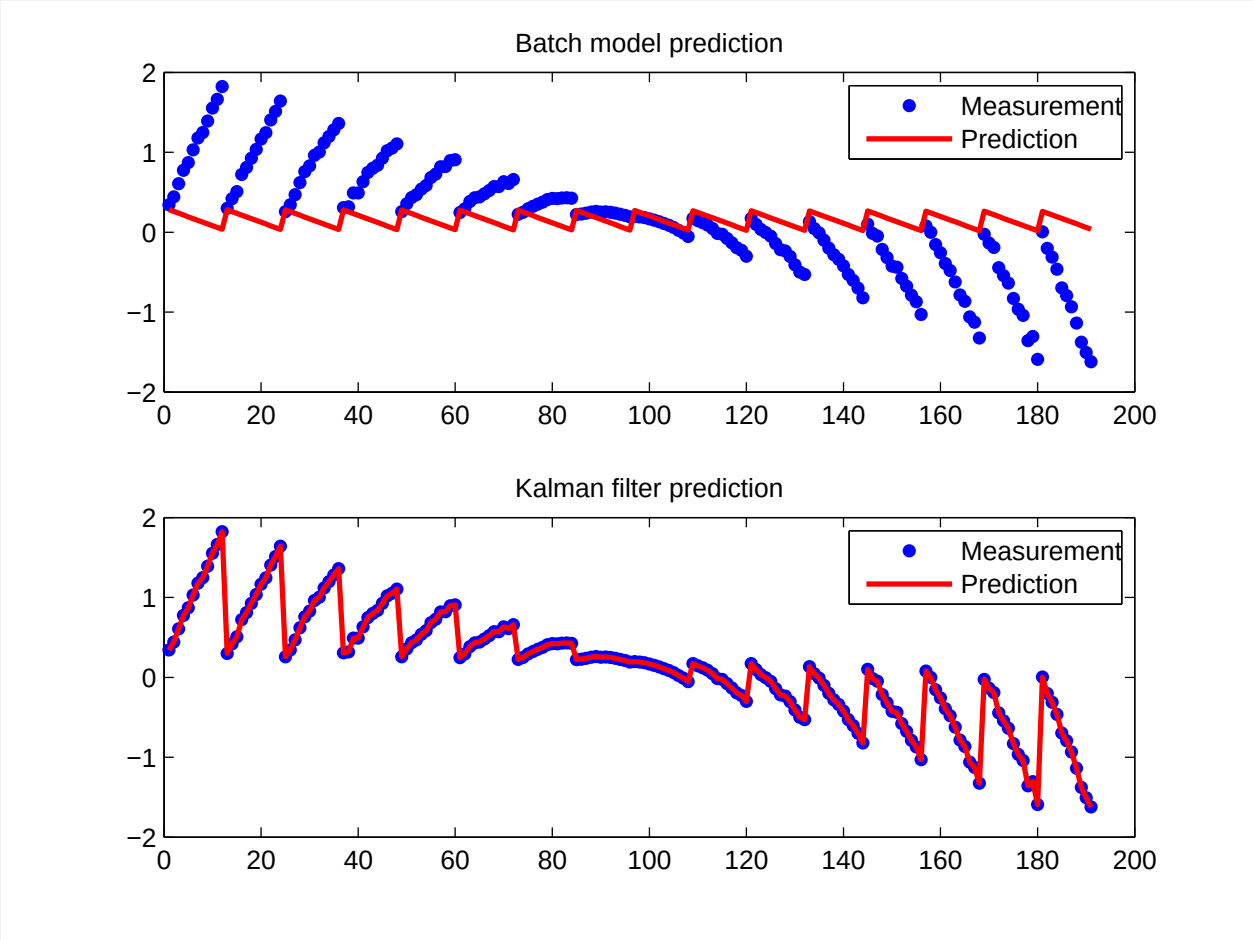




Kalman Filtering [5/7]



Kalman Filtering [6/7]



Kalman Filtering [7/7]

- In Kalman filtering literature the stochastic process $\theta(t)$ (or θ_k) is often denoted with $\mathbf{x}(t)$ (or $\mathbf{x}_k$).
- Kalman filter gives the solution to generic linear state space models of the form

$$\begin{aligned} \mathbf{x}_k &= \mathbf{A}\,\mathbf{x}_{k-1} + \mathbf{q}_k, & \mathbf{q}_k &\sim \mathrm{N}(0, \mathbf{Q}) \\ \mathbf{y}_k &= \mathbf{H}\,\mathbf{x}_k + \mathbf{r}_k, & \mathbf{r}_k &\sim \mathrm{N}(0, \mathbf{R}) \\ \mathbf{x}_0 &\sim \mathrm{N}(\mathbf{m}_0, \mathbf{P}_0). \end{aligned}$$

- Vector $\mathbf{x}_k$ is the state and vector $\mathbf{y}_k$ is the measurement.
- In probabilistic notation the model is:

$$\begin{aligned} p(\mathbf{y}_k \,|\, \mathbf{x}_k) &= \mathrm{N}(\mathbf{y}_k \,|\, \mathbf{H}\,\mathbf{x}_k, \mathbf{R}) \\ p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) &= \mathrm{N}(\mathbf{x}_k \,|\, \mathbf{A}\,\mathbf{x}_{k-1}, \mathbf{Q}). \end{aligned}$$

Probabilistic Non-Linear Filtering [1/2]

- Generic discrete-time state space models

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{q}_k)$$
$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{r}_k).$$

- Generic Markov models

$$\mathbf{y}_k \sim p(\mathbf{y}_k \,|\, \mathbf{x}_k)$$
$$\mathbf{x}_k \sim p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}).$$

- Approximation methods: Extended Kalman filters (EKF), Unscented Kalman filters (UKF), sequential Monte Carlo (SMC) filters a'ka particle filters.

Probabilistic Non-Linear Filtering [2/2]

- In continuous-discrete filtering models, dynamics are modeled in continuous time, measurements at discrete time steps.
- The continuous time versions of Markov models are called as stochastic differential equations:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t) + \mathbf{w}(t)$$

  where $\mathbf{w}(t)$ is a continuous time Gaussian white noise process.
- Approximation methods: Extended Kalman filters, Unscented Kalman filters, sequential Monte Carlo, particle filters.

Summary

- Correlation coefficient R measures goodness of one-dimensional linear regression fit to scatter plot.
- Multidimensional models and non-linear linear-in-parameter models are direct generalizations of the one-dimensional linear regression.
- The coefficient of determination R^2, which is the square of correlation coefficient can be generalized to multidimensional and non-linear models.
- With non-linear models over-fitting is a huge problem.
- Regularization diminishes the over-fitting problem by including additional cost function to the error function.
- Bayesian data analysis provides unified framework of statistical regression and regularization.
- Statistical estimation in dynamic context leads to Kalman filters and sequential Monte Carlo filters.