Explizite Implementierung von A-Priori Information

Jörg Lemm

Institut für Theoretische Physik I
Universität Münster

lemm@uni-muenster.de
http://pauli.uni-muenster.de/~lemm

Empirisches Lernen: Einige Beispiele

- NETtalk (Sejnowski, Rosenberg 1987)
- Postleitzahlenerkennung (LeCun 1989)
- Auto fahren (Pomerleau 1989)
- Buchstabenerkennung, Spracherkennung
- Gesichtserkennung
- Fehlerdiagnose, Medizinische Diagnose
- Zeitreihenprognose (Aktienkurse)
- Kreditrisikoprognose, $\cdots$

1. Klassifikation/Muster–/Objekterkennung
2. Regression
3. Dichteschätzung

Empirisches Lernen: Phasen

- 1960er: Perzeptron. Rosenblatt 1962 ("McCulloch–Pits–Neuron")
$$y = \mathrm{sign}(w \cdot x - b)$$
Minski–Papert 1969 ("Krise")

- 1960er und 1970er: Lernen als schlechtgestelltes Problem ("ill–posed" problems, Tikhonov 1963), Regularisierungsansätze

- 1980er: Neuronale Netze (= Mehrschichtperzeptrons) Backpropagation, Parallel Distributed Processing: Rumelhart et. al. 1986

- 1990er: Zurück zur Regularisierung
Bayes'sche Interpretation,
Gauß'sche Prozesse, Graphische Modelle,
strucural risk minimization & support vector machines, Vapnik 1995
("Perceptron in feature space")

Empirisches Lernen: A–Priori–Information

Trainingsdaten: e.g., $h(x_1) = 5$

Testdaten: $h(x_2) =$?

Abhängigkeiten (A–Priori-Information, Regeln, Energiefunktion), e.g., "Glattheit":
$h(x_2) - h(x_1) = 2$

Generalisierung/Vorhersage/Induktion:
$h(x_2) = 7$

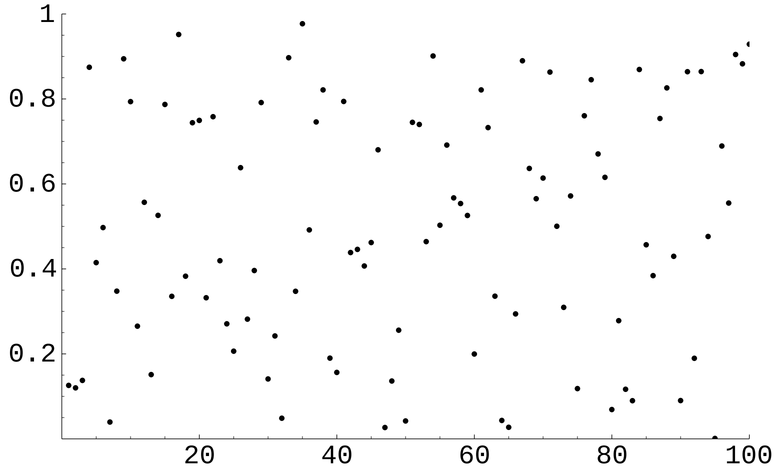
$$h(x_2) - h(x_3) = ?$$
$$h(x_3) - h(x_4) = ?$$
$$\cdots$$

Generalisierung erfordert

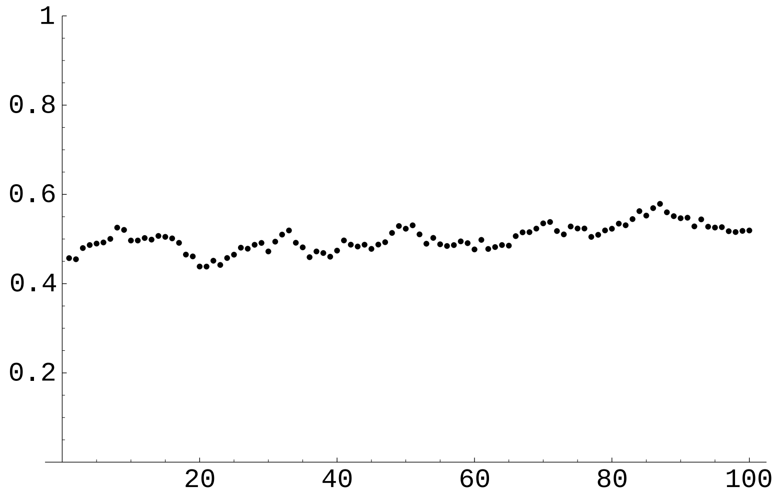
1. (Algorithmischer Aspekt) Umformulieren von Trainingsdaten und A–Priori–Information in Aussagen über Testdaten ("No–free–lunch")

2. (Kontrollaspekt) empirische Kontrolle/Verifikation/Messung von Abhängigkeiten zwischen Test– und Trainingsdaten.

Zufallsdaten: Keine Vorhersage möglich



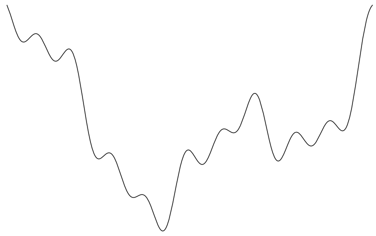
Messapparatur mit Eingaberauschen
oder Eingabemittelung:
Vorhersage möglich



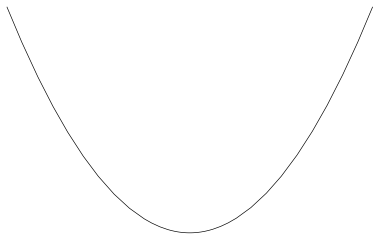
Fehlerlandschaften

Parametrische Methoden (z.B., Neuronale Netze)

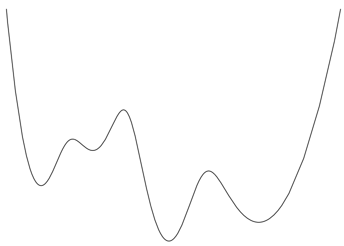
(viele lokale Minima, implizite A-Priori Information)



Gauß'sche Prozesses (explizite A-Priori Information (z.B., Glattheit), immer unimodal, effektive Dimension = Anzahl der Daten)



Mischung von Gauß–Prozessen (flexibel, explizite Implementierung von komplexer A-Priori Information)



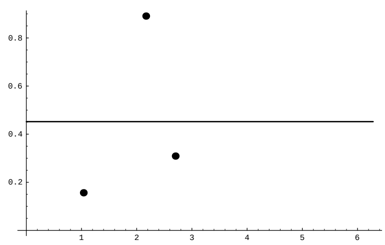
Fehler/Energieminimierung auf Trainingsdaten

$$E(\xi) = \frac{1}{2}\sum_i (y_i - h(x_i, \xi))^2$$

Lernen = Bestimmung des optimalen Parametervektors ξ.

Bsp.: Konstante

$$h(x) = b$$

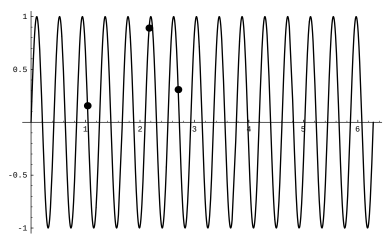


Bsp.: Lineare Regression

$$h(x) = ax + b$$

Bsp.: Nichtlineare Regression

$$h(x) = \sin(ax)$$



Bsp.: Nichtlineare Regression (Neuronales Netz)

$$h_i(x) = \sigma(Wh_{i-1}(x) - B), \;\; h_1 = \sigma(wx - b)$$

Glattheit

1. Fehler/Energieminimierung auf Trainingsdaten

$$E_D[h] = \frac{1}{2}\sum_i (y_i - h(x_i))^2$$

2. Addiere einen extra "Glattheitsterm" (Approximative Invarianz unter infinitesimalen Translationen)

$$\begin{aligned}
E_0[h] &= \frac{1}{2}\sum_k \left(\frac{h(x_{k+1}) - h(x_k)}{x_{k+1} - x_k}\right)^2 \\
&\to \frac{1}{2}\int dx \left(\frac{\partial h(x)}{\partial x}\right)^2 \qquad (x \text{ kontinuierlich}) \\
&= -\frac{1}{2}\int dx\, h(x)\frac{\partial^2 h(x)}{\partial x^2} \qquad (\text{partielle Int.}) \\
&= -\frac{1}{2}\langle h \,|\Delta|\, h\rangle \qquad (\text{Skalarprodukt})
\end{aligned}$$

$-\Delta$ = neg. Laplaceoperator (kinetische Energie)

(Höhere Ableitungen führen zu glatteren Funktionen, z.B. zu verschiedene Arten von Splines oder Radialen Basis Funktioen)

3. Minimiere $E_D + E_0$. (Regularisierter Fehler)

Minimiere regularisiertes Fehlerfunktional

$$
\begin{aligned}
E[h] &= \underbrace{E_D[h]}_{\text{Training}} + \underbrace{E_0[h]}_{\text{Regularisierer}} \\
&= \underbrace{\frac{1}{2}\sum_{i=1}^{N}(y_i - h(x_i))^2}_{\text{Training}} + \underbrace{\frac{1}{2}\int dx \left(\frac{\partial h}{\partial x}\right)^2}_{\text{Regularisierer}} \\
&= \underbrace{\frac{1}{2}\left\langle h - t_D \middle| K_D \middle| h - t_D \right\rangle + c}_{\text{Training}} \underbrace{- \frac{1}{2}\langle h \,|\Delta|\, h\rangle}_{\text{Regularisierer}} \\
&= \frac{1}{2}\sum_{i=0}^{N}\left\langle h - t_i \middle| K_i \middle| h - t_i \right\rangle \\
&= \frac{1}{2}\langle h - t \,|K|\, h - t\rangle + c' \qquad \Rightarrow h^* = t
\end{aligned}
$$

$t_D(x) =$ Mittelwert y_i für x

$K_D(x,x') = \delta(x-x')\sum_i \delta(x-x_i) =$ Anzahl y_i für x

$-\Delta = K_0 =$ neg. Laplaceop., K_i reel symm. pos. semidef.

$t_i(x) = y_i$ für $i > 0$, $t_0(x) \equiv 0$

$K = \sum_i K_i$, $t = K^{-1}\sum_i^N K_i t_i = h^*$

c gegeben durch die h–unabhängige Varianz der y_i bei x

c' gegeben durch die Varianz der t_i

"Kernel"–Methoden

$$E(h) = \frac{1}{2}\left\langle h - t_D \,\middle|\, K_D \,\middle|\, h - t_D \right\rangle + \frac{1}{2}\left\langle h - t_0 \,\middle|\, K_0 \,\middle|\, h - t_0 \right\rangle$$

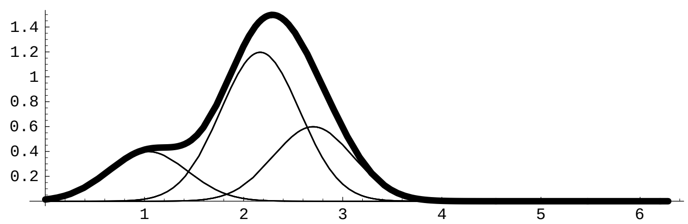
z.B.

$$E(h) = \frac{1}{2}\sum_i (y_i - h(x_i))^2 + \frac{\lambda}{2}\int dx \left(\frac{dh(x)}{dx}\right)^2$$

$$h = t_0 + \underbrace{K_0^{-1}}_{\text{Kovarianz}} \underbrace{(K_D^{-1} + K_{0,DD}^{-1})^{-1}(t_D - t_0)}_{a}$$

Für $t_0 = 0$

$$h = \sum_i \underbrace{K_0^{-1}(x, x_i)}_{\text{Kovarianz}} a_i$$



1. Definiere Modell–"Likelihoods" für Trainingsdaten (z.B., Regression, $h(x)$ = Regressionsfunktion)

$$p(y_i|x_i,h) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y_i - h(x_i))^2}{2\sigma^2}} \propto e^{-E_i[h]}$$

2. Definiere A–Priori–Wahrscheinlichkeit über Modelle

$$p(h) \propto e^{-E_0[h]}$$

3. Berechne A–Posteriori–Wahrscheinlichkeit der Modelle, gegeben Trainingsdaten

$$p(h|D) \propto p(D|h)p(h) \propto e^{-E_D[h] - E_0[h]}$$

4. Zur Vorhersage integriere über alle Modelle mit Gewichtung $p(h|D)$

$$p(y|x,D) = \int dh\, p(y|x,h)\, p(h|D)$$

5. In Maximum–A–Posteriori–Approximation

$$\text{argmax}_h p(h|D) = \text{argmin}_h (E_D[h] + E_0[h])$$

$\Rightarrow$ Posteriormaximierung = Fehlerminimierung

- Eindimensional:

$$p(h) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(h-t)^2}{2\sigma^2}}$$

- Mehrdimensional:

$$p(h) = \frac{e^{-\frac{1}{2}\sum_{i,j}(h_i - t_i)C^{-1}_{ij}(h_j - t_j)}}{(2\pi)^{\frac{d}{2}}(\det C)^{\frac{1}{2}}}$$

- Gauß'scher Prozess:

$$p(h) = \frac{e^{-\frac{1}{2}\left\langle h-t \middle| C^{-1} \middle| h-t \right\rangle}}{(2\pi)^{\frac{d}{2}}(\det C)^{\frac{1}{2}}}$$

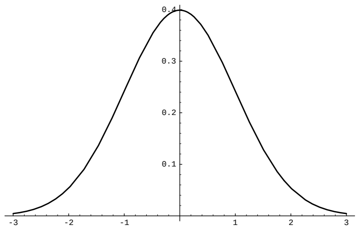
wobei

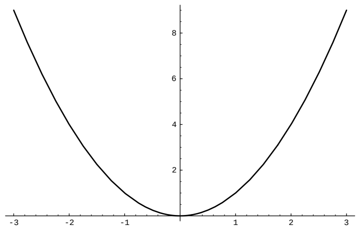
$$\left\langle h - t \middle| C^{-1} \middle| h - t \right\rangle =$$

$$\int dx\, dx' \Big(h(x) - t(x)\Big) C^{-1}(x, x') \Big(h(x' - t(x')\Big)$$

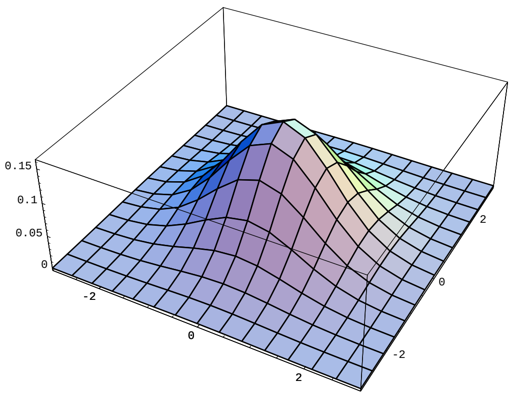
Gauß–Verteilung, quadratischer Fehler

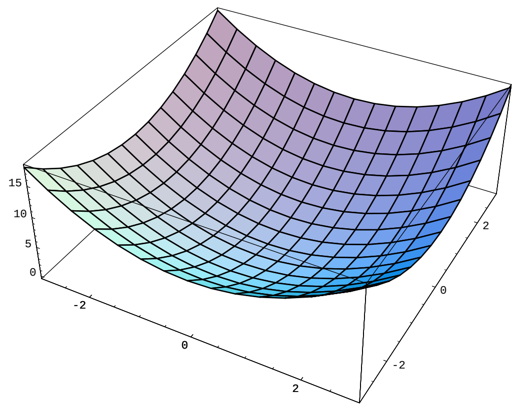
1–dimensional





2–dimensional





Gauß'sche Processe entsprechen quadratischen Fehlerfunktionalen. Wir benutzen nun quadratische Bausteine um allgemeine nichtkonvexe Fehlerfunktionale zu konstruieren

Quadratisches Konzept

$$\boxed{d^2[h] = \langle h - t\,|K|\,h - t\rangle}$$

$t = t(x)$ Template (Mittel, Prototyp, Referenz, Daten)

$K = W^T W = (I - S)^T (I - S)$, symm., positiv (semi–)definit,

(definiert Abstandsmaß , inverse Kovarianz $K = C^{-1}$)

Beispiel: Quadratischer Fehler

$$\langle h - t\,|K|\,h - t\rangle = (y_i - h(x_i))^2$$

wobei $t(x) = y_i$ und $K(x, x') = \delta(x - x')\delta(x - x_i)$.

Beispiel: Glattheit

$$\langle h - t\,|K|\,h - t\rangle = \langle h\,|K|\,h\rangle$$

mit $t(x) \equiv 0$ (Nulltemplate), und

$K = -\Delta$ (kinetische Energie)

$K = \sum_{m=0}^{\infty}(-1)^m \frac{\sigma^{2m}}{m!2^m}\Delta^m$ (Radiale Basis Funktionen)

Die Approximationsbedingungen

$$\boxed{t_1 \text{ UND } t_2 \text{ UND } \cdots \text{ UND } t_N}$$

werden approximiert durch das additives Fehler-funktional

$$\boxed{E = \sum_i^N \frac{1}{2}\, d_i^2[h] = \frac{1}{2}\, d^2[h] + E_{\text{Min}}}$$

wobei $\qquad d^2[h] = \langle h - t\,|K|\, h - t\rangle$

das quadratische Konzept bezeichnet mit

$$t = K^{-1}\tilde{t} \text{ mit } \tilde{t} = \sum_i^N K_i t_i \text{ und } K = \sum_i K_i$$

sowie h–unabhängiger minimaler Energie

$$E_{\text{Min}} = \frac{1}{2}\left(\sum_i^N \langle t_i\,|K_i|\, t_i\rangle - \langle t\,|K|\, t\rangle\right).$$

Die Stationaritätsbedingung ist linear

$$\boxed{0 = \tilde{t} - Kh \Rightarrow h = t.}$$

Alternative Approximationsbedingungen

$$\boxed{\text{(data UND } t_1\text{) ODER (data UND } t_2\text{)} \cdots}$$

können implementiert werden durch eine
Mischung von Gauß'schen Prozessen:

$$\boxed{p(h) = \sum_j^{n_j} p_j \, p(h|j)}$$

mit

$$p(h|j) = e^{-\beta E_j}/Z_j, \qquad Z_i = \int dh \, e^{-\beta E_i} \quad \text{(Norm)}$$

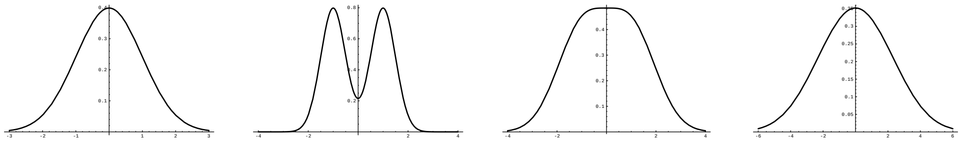
Dies entspricht einem Fehlerfunktional

$$\boxed{E^M[h] = -\ln\left(\sum_j^{n_j} p_j \, e^{-\beta \, d_j^2[h] - \ln Z_j}\right)}$$

Als (nichtlineare) Stationaritätsbedingung resultiert
$0 = t^M[h] - K^M[h]h$,
wobei $t^M = \sum_i p(h,i) K_i \bar{t}_i = \sum_i p(h,i) \tilde{t}_i$.

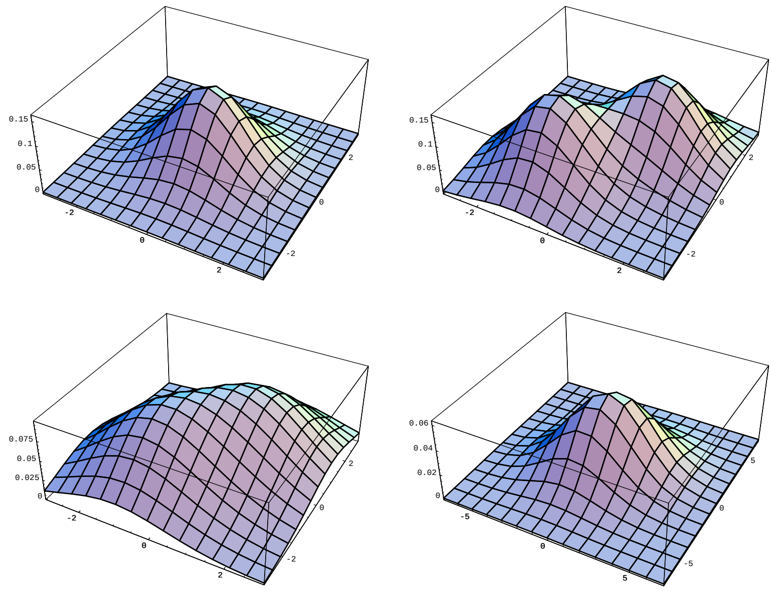
Mischung von Gauß–Verteilungen

1–dimensional



1D-Gauß $\sigma = 1$, Mischungen: $\sigma_i = 0.5$, $\sigma_i = 1$, $\sigma_i = 2$

2–dimensional



2D-Gauß $\sigma = 1$, Mischungen: $\sigma_i = 1$, $\sigma = 1.4$, $\sigma_i = 2$

Mittelwert der Vorhersageverteilung

$$\bar{y} = \int dy\, y\, p(y|x,f) = \sum_j b_j \,\bar{t}_j$$

mit Komponentenmittelwert

$$\bar{t}_j = t_j + \mathbf{K}_j^{-1} \mathbf{K}_{t_D}(j)(t_D - t_j)$$

Mischungskoeffizient

$$b_j = \frac{p(j)p(y_D|x_D, D_0, j)}{\sum_j p(j)p(y_D|x_D, D_0, j)}$$

und Vorhersageverteilung für Komponente j

$$p(y_D|x_D, D_0, j) =$$

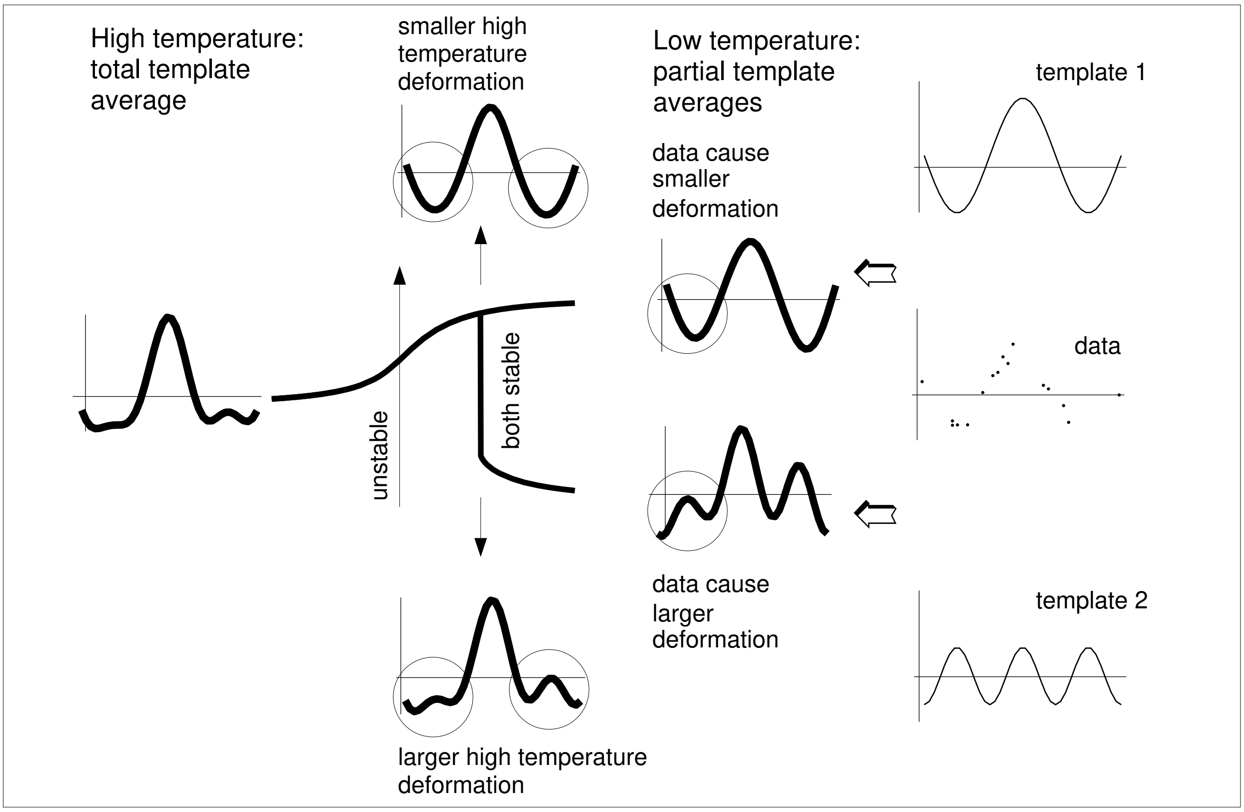
$$e^{-\frac{\beta}{2}\left\langle t_D - t_j \middle| \mathbf{K}_{t_D}(j) \middle| t_D - t_j \right\rangle + \frac{1}{2}\ln\det(\frac{\beta}{2\pi}\mathbf{K}_{t_D}(j))}.$$

Hierbei ist

$$\mathbf{K}_{t_D}(j) = (\mathbf{K}_D^{-1} + \mathbf{K}_{j,DD}^{-1})^{-1}$$

eine $\tilde{n} \times \tilde{n}$–Matrix im Raum der Daten und $\mathbf{K}_{j,DD}^{-1}$ die Projektion der Priorkovarianz der Komponente j in diesen Raum.

Zeitreihe mit zwei Templates



$$p^M = p_1^M + p_2^M \propto e^{-\frac{\beta}{2}(d_D^2 + d_1^2)} + e^{-\frac{\beta}{2}(d_D^2 + d_2^2)}$$

Die Lösungen der Stationaritätsbedingung

$$h = p_1^M[h]t_1 + p_2^M[h]t_2$$

liegen auf einer eindimensionalen Linie und werden parametrisiert durch β (Ferromagnet)

$$\boxed{h = t + \tanh\left[\frac{\beta}{2}(E_2 - E_1)\right]\frac{t_1 - t_2}{2}}$$

Praktisch sehr wichtig sind alternative, kontinuierliche Transformationen. Das Referenzbild eines Gesichtes, beispielsweise, mag eine ganze Klasse von Gesichtern repräsentieren die durch Verschiebungen, Rotationen, oder Reskalierung entstehen.

$$p^M \propto \int d\theta\, p(\theta) \sum_i e^{-\frac{\beta}{2}\sum_j \left\langle h - t_j(\theta) \middle| K_j \middle| h - t_j(\theta) \right\rangle + \beta F}$$

Die korrespondierende Bayes'sche Integration über alle Varianten kann in der Regel nicht analytisch ausgeführt werden und muß daher z.B. in Maximum A–Posteriori Approximation (MAP) ausgeführt werden.

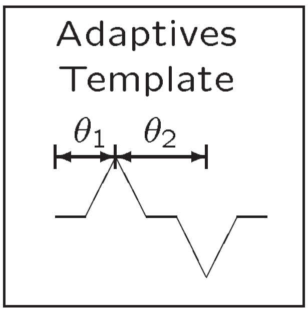
D.h. wir behandeln

☞ $\int d\theta$ in MAP

☞ $\sum_i$ exakt

Algorithmus:

1. Bestimme optimales θ^* auf Trainingsdaten
2. Bestimme h^* für festes θ^*
3. Bestimme θ^* für festes h^*
4. Iteriere bis zur Konvergenz.



Iterationsverfahren = Lernalgorithmen

$$\boxed{h^{k+1} = h^k + \eta A^{-1}(t^k[h^k] - K^k[h^k]h^k)}$$

Lernalgorithmus	inv. Iterationsmatrix
Gradient	$\mathbf{A} = \mathbf{I}$ (Identität)
Jacobi	$\mathbf{A} = \mathbf{D}$ (diagonal)
Gauß–Seidel	$\mathbf{A} = \mathbf{D} + \mathbf{L}$ (unt.Dreieck)
Newton	$\mathbf{A} = -\mathbf{H}$ (neg. Hessem.)
Prior–Relaxation	$\mathbf{A} = \mathbf{K}$ (für Misch.: EM)
Massive Relaxation	$\mathbf{A} = \mathbf{A}_0 + m^2\mathbf{I}$
Lineare Randbed.	$\mathbf{A} = \mathbf{A}_{II} + \mathbf{A}_{IB} + \mathbf{A}_{BB}$
Gauß	$\mathbf{A} = \sum_{k=0}^{\infty} \frac{1}{k!}\left(\frac{\mathrm{M}^2}{2\tilde{\sigma}^2}\right)^k = e^{\frac{\mathrm{M}^2}{2\tilde{\sigma}^2}}$

Hessematrix, Mischungsmodell

$$\begin{aligned} H^M \; = & -\beta \textstyle\sum_i p(h,i) K_i \\ & +\beta^2 \textstyle\sum_i p(h,i) | K_i h - t_i \rangle\langle K_i h - t_i | \\ & -\beta^2 \textstyle\sum_{ij} p(h,i) | K_i h - t_i \rangle\langle K_j h - t_j | p(h,j). \end{aligned}$$

Hessematrix, Landau–Ginzburg–Modell

$$\begin{aligned} H^L = & \textstyle\sum_i \sum_{j_i} M_{ij_i} K_{ij_i} \\ & + \textstyle\sum_i \sum_{j_i} \sum_{k_i} M_{ij_ik_i} | K_{ij_i} h - t_{ij_i} \rangle\langle K_{ij_i} h - t_{ij_i} | \end{aligned}$$

mit $M_{ij_ik_i} = \prod_{j_i \neq l_i \neq k_i} \beta_F^{n_{j_i}-1} d_{il_i}^2[h]$

Beispiel: Optische Flußfelder

Stationaritätsgleichung für Hyperparameter

Mittelwert der Vorhersageverteilung:

$$\bar{y} = \int dy\, y\, p(y|x, D, D_0) = \sum_j \int d\theta\; b_j(\theta)\, \bar{t}_j(\theta)$$

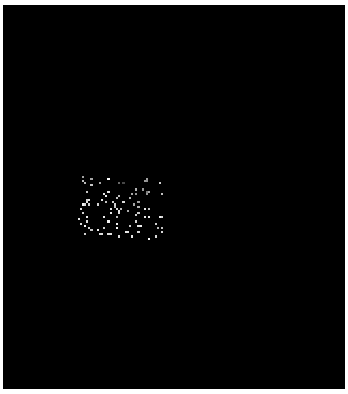
mit Mischungskoeffizienten

$$b_j(\theta) = \frac{p(\theta, j)\, p(y_D|x_D, D_0, \theta, j)}{\sum_j \int d\theta\, p(\theta, j)\, p(y_D|x_D, D_0, \theta, j)}$$

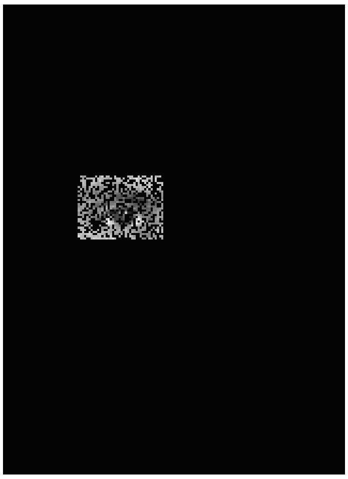
Die Stationaritätsbedingung für θ ist

$$\begin{aligned} 0 \;=\; & \sum_j \left(\left\langle \frac{\partial t_j(\theta)}{\partial \theta} \middle| \mathbf{K}_{t_D}(\theta, j) \middle| t_j(\theta) - t_D \right\rangle \right. \\ & + \frac{1}{2} \left\langle t_D - t_j(\theta) \middle| \frac{\partial \mathbf{K}_{t_D}(\theta, j)}{\partial \theta} \middle| t_D - t_j(\theta) \right\rangle \\ & \left. - \mathrm{Tr} \left(\mathbf{K}_{t_D}^{-1}(\theta, j) \frac{\partial \mathbf{K}_{t_D}(\theta, j)}{\partial \theta} \right) - \frac{1}{p(\theta, j)} \frac{\partial p(\theta, j)}{\partial \theta} \right) \end{aligned}$$

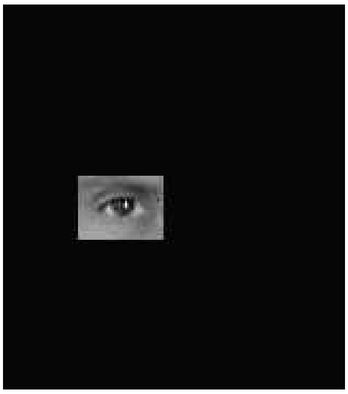
Einige Bilddaten



100



1000



10000

Lernen mit Mischungsmodellen 1

Zweikomponentiges Mischungsmodell bei tiefer Temperatur. Beispiel: Bildrekonstruktion. Templatefunktionen t_i representieren (alternative) A–Priori–Erwartungen betreffend das Bild h. Hyperparameter θ wurden einbezogen für Translationen, Rotationen und Skalierungen der Bilder.

Daten D und 2 Templatefunktionen t_1, t_2 (2-dim.):

Original und rekonstruiertes Bild $h(x)$

Lernen mit Mischungsmodellen 2

Zwei andere Templatefunktionen.

Daten und 2 Templatefunktionen t_1, t_2:

Original und rekonstruiertes Bild $h(x)$

Bei hohen Temperaturen tragen beide Templates zur Lösung bei.

Rekonstruierte Bilder bei tiefer (links und rechts) und bei hoher Temperatur (Mitte).

Fazit

1. Die adäquate Modellierung von A–Priori–Information ist besonders wichtig für komplexe Problemstellungen mit relative wenig Daten.

2. Mischungen von Gauß'schen Prozessen sind technisch oft verhältnismäßig leicht implementierbar und besitzen eine große Flexibilität, insbesondere bei Kombination mit Hyperparametern.

3. Die explizite Formulierung von A–Priori–Information liefert eine wohldefinierte Aufgabenstellung und damit einen Startpunkt für eventuelle weitere Näherungen, selbst wenn das Modell direkt nicht lösbar ist.