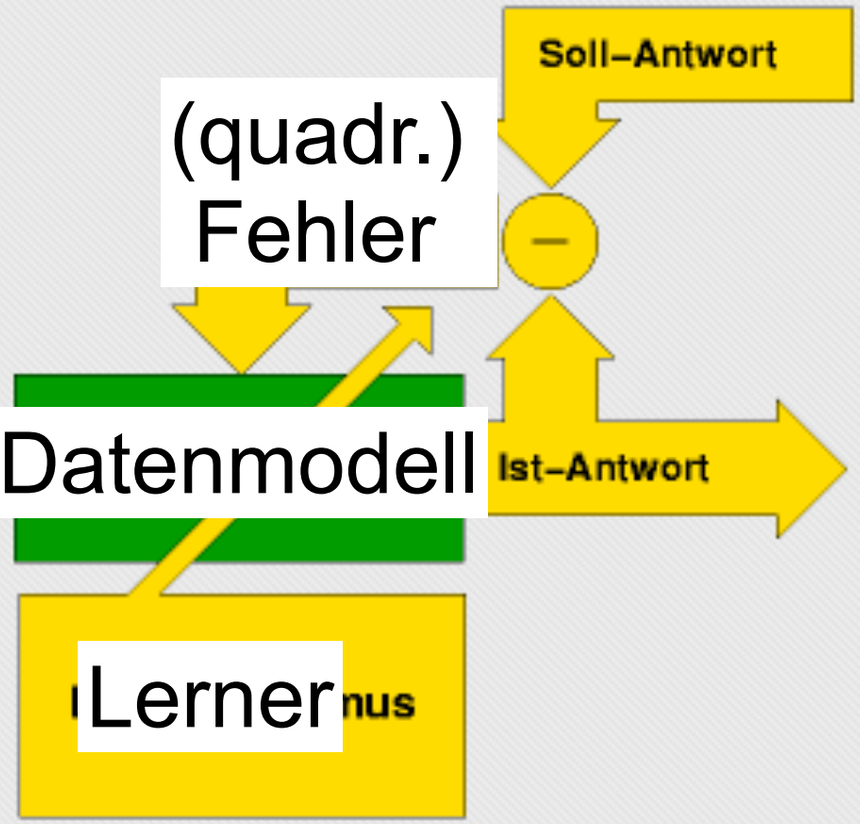
Wiederholung Grundbegriffe am Bsp. Regression

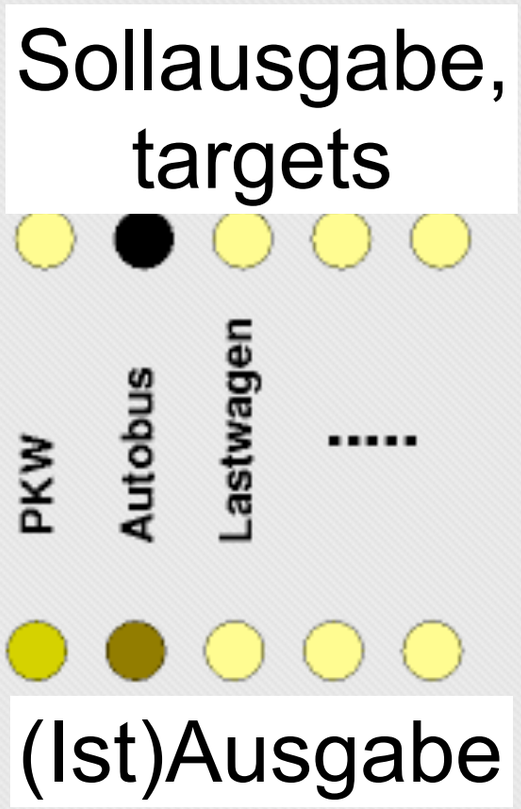
Grundbegriffe

- überwachtes Lernen (Eingabe, Ausgabe, Modell)
- Modellselektion via parametrisierte Funktion (Bsp: Polynom)
- Parameteroptimierung (Minimierung quadratischer Fehler)
- Overfitting, Regularisierung

=> Achtung Annahme !
(inductive bias)

Wiederholung Szenario: Überwachtes Lernen





(Trainings-)
Daten

Datenmodell =
parametrisierte Funktion
(Modellselektion)

Grundbegriffe der probabilistischen Modellierung

Lernziele:

- Modell von Unsicherheit: (normalverteilte) Störung/Rauschen
- Grundbegriffe prob. Modellierung (Likelihood, predictive distribution)
- Grundbegriffe: Bayes Ansatz, Modellierung der Parameterverteilung, inkrementelles Bayes'sches Lernen

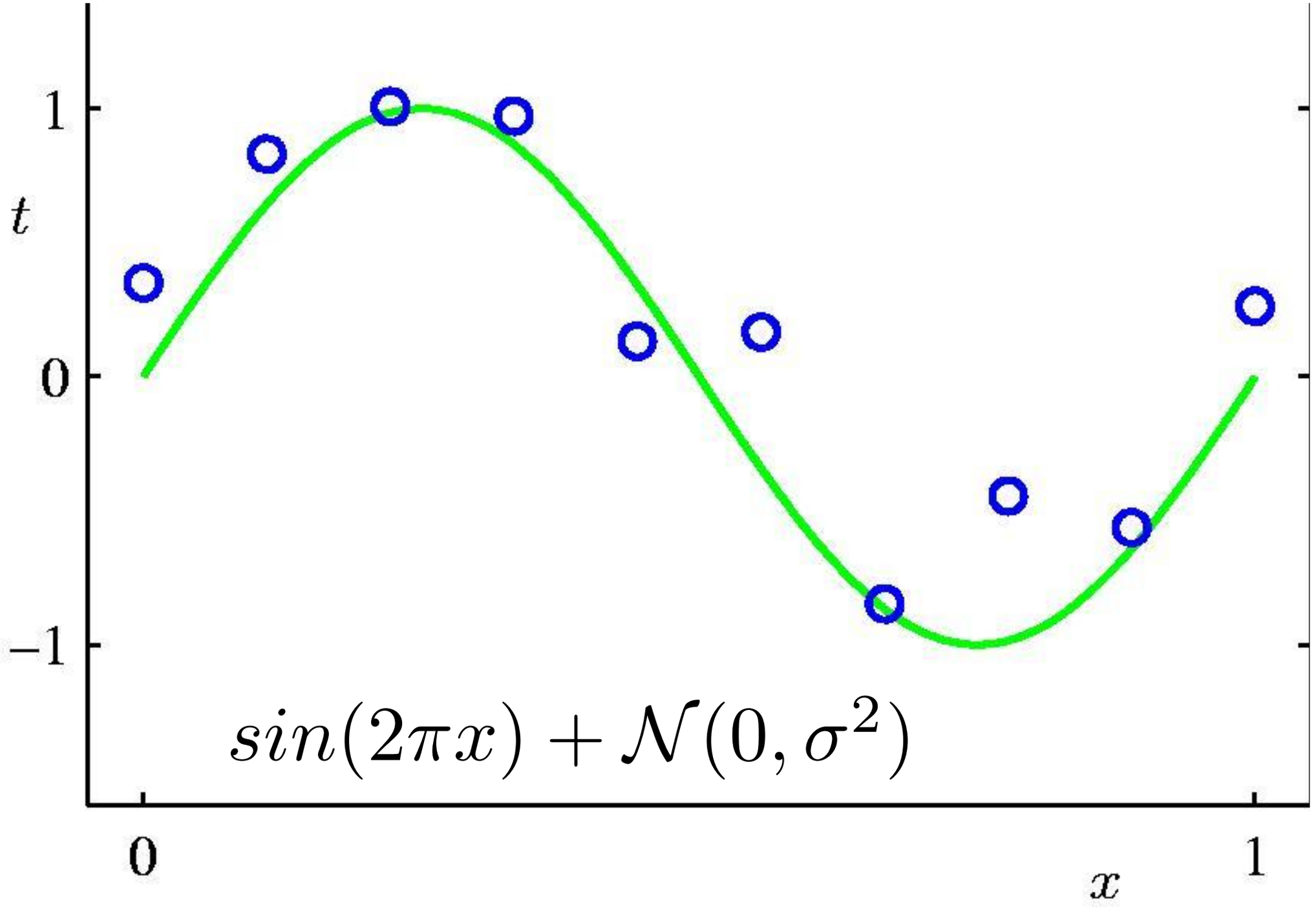
(Mathem.) Voraussetzungen:

- (multi-dim.) Normalverteilung, Ableitungen, bedingte W.-keiten

Vorgehen:

- "triviales" Funktionsbeispiel: sin( ) + Rauschen
- Ansatz (weiterhin): Summe von Polynomen
- nach: Bishop, Kap 1.

Probabilistic modeling



Slide from Bishop, machine learning

Probabilistic modeling

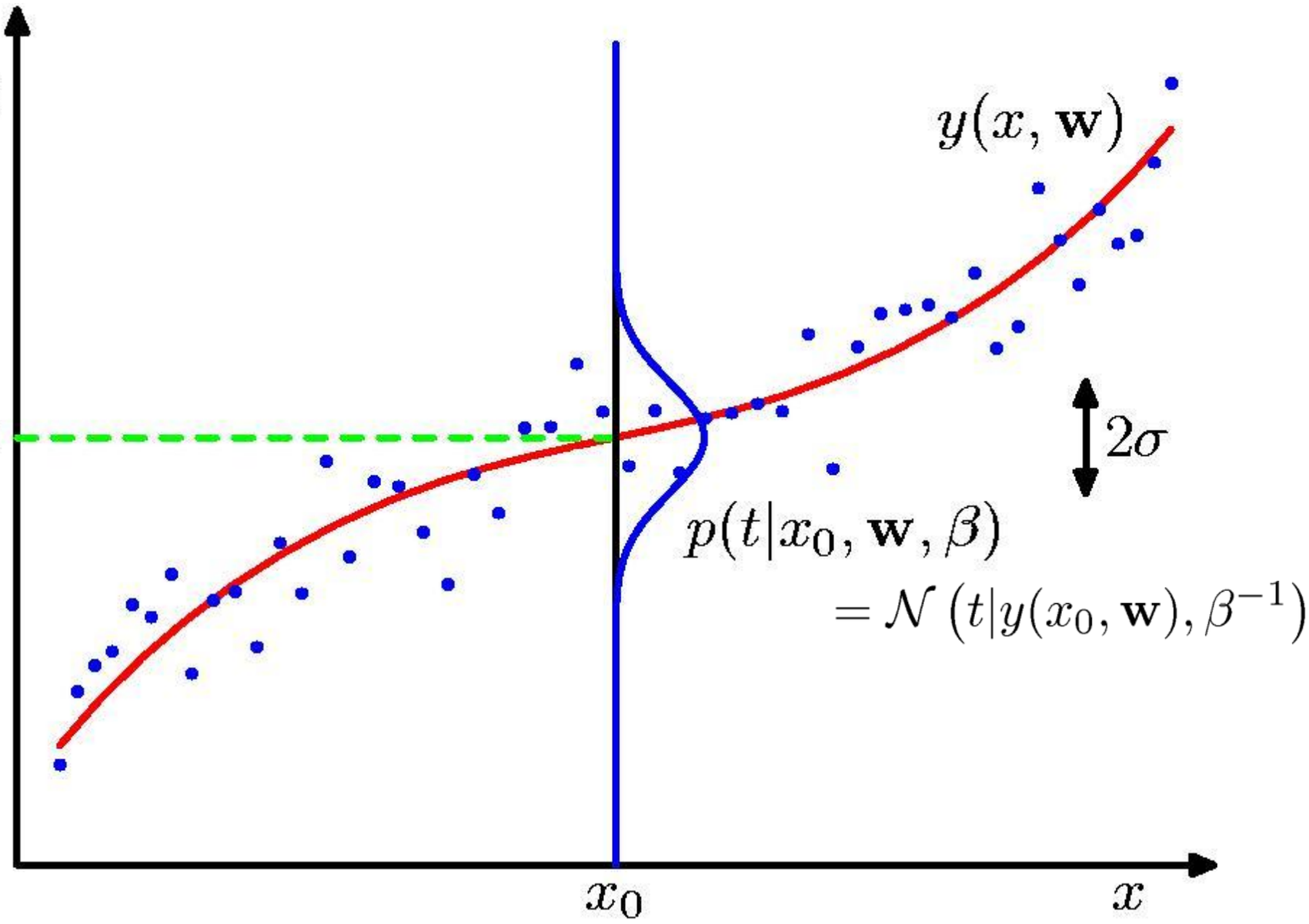
Im Beispiel:

- **Datenerzeugung: Funktionswert + Gaußsches Rauschen**

$$sin(2\pi x) + \mathcal{N}(0, \sigma^2)$$

- **Gaußsches Rauschen ist definiert durch Mittelwert + Varianz** σ^2
- **Ziel: finde Modell für die Daten und die Unsicherheit (das Rauschen)**
- ***konkrete Annahme*: Rauschen ist gaußverteilt, Mittelwert = 0**
  - **dann: wähle Datenmodell wie vorher**
  - **und addiere Gaußschen Term:** $t = y(\omega, x_n) + N(0, \sigma^2)$
  - **schätze Modellparameter und** σ^2
- **Generalisierung: mache wahrscheinlichkeitsbasierte Vorhersage für neue Eingaben**

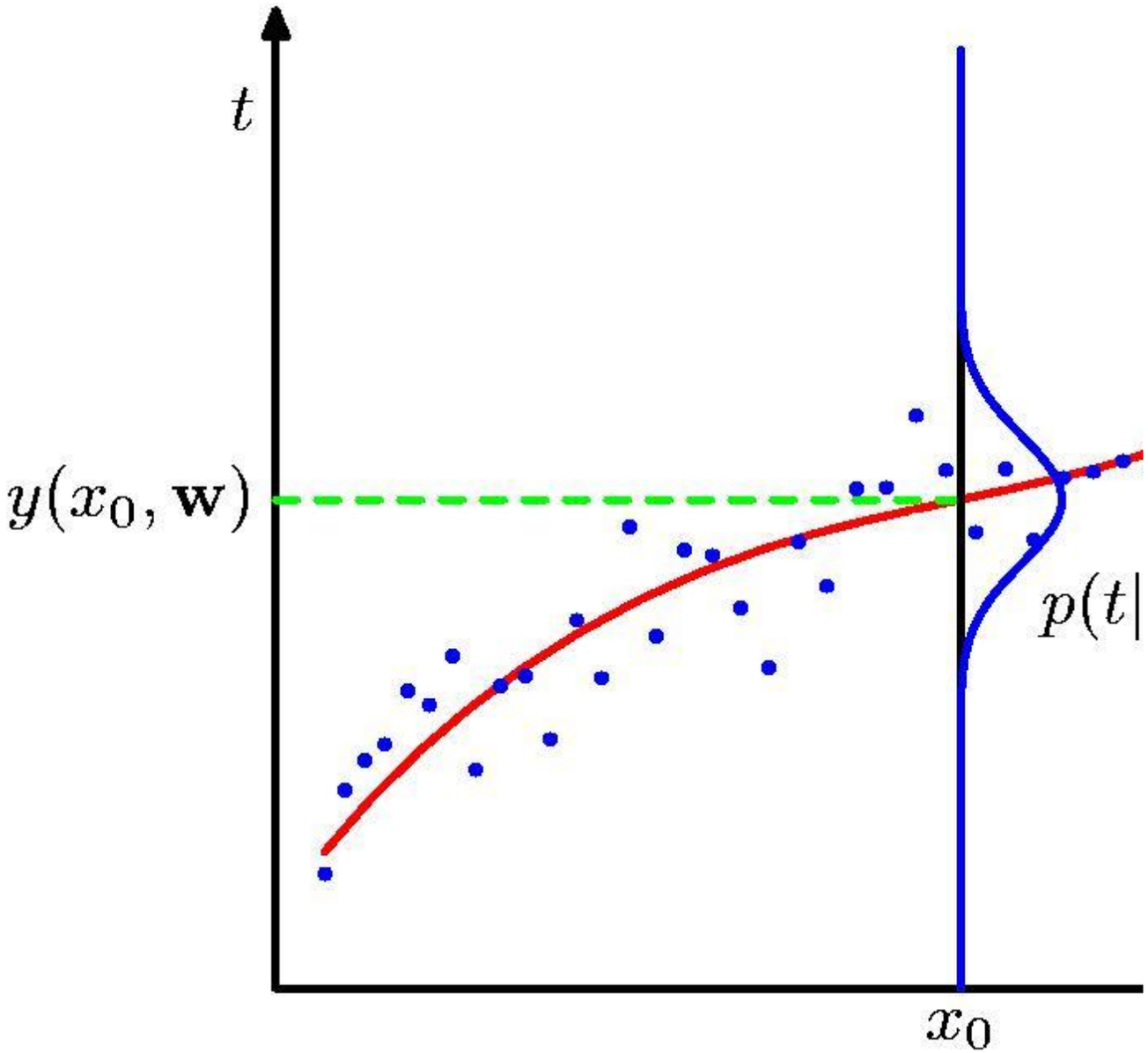
Daten mit Unsicherheit



modelliere Daten durch W.-keiten

Slide from Bishop, machine learning

Daten mit Unsicherheit



Slide from Bishop, machine learning

Probabilistische Modellierung: der Likelihood

Modellierung durch Likelihood

- für gegebenes parameterisches Datenmodell y(w,x)
- Likelihood = bedingte Wahrscheinlichkeit, eine Ausgabe t bei gegebenen Parametern w, β und Eingabe x_0 zu erhalten:

$$p(t|x_0, \omega, \beta)$$

- diese W.-keit ist nach Annahme graußverteilt mit Mittelwert im Modell und Varianz $\beta^{-1} = \sigma^2$ (1-dim. Fall):

$$p(t|x_0, \omega, \beta) = N\left(t|y(\omega, x_0), \sigma^2\right)$$

- β heißt auch Präzision
- multidimensional:

$$p(\vec{t}|\vec{x}_0, \omega, \Sigma^{-1}) = N\left(\vec{t}|\vec{y}(\omega, x_0), \Sigma^{-1}\right)$$

Stochastisches Datenmodell

von der Messung zur Wahrscheinlichkeitsverteilung

- für gegebenes parameterisches Datenmodell $y(x, \vec{\omega})$
- und gegebene Sollausgaben (*targets*) t

$$t = y(x, \vec{\omega}) + N(0, \beta^{-1})$$

$$\Leftrightarrow \quad t - y(x, \vec{\omega}) \sim N(0, \beta^{-1})$$

$$\Leftrightarrow \quad t \sim N(y(x, \vec{w}), \beta^{-1})$$

- Datenmodell: Normalverteilung um das Modell $y(x, \vec{\omega})$ mit Eingabe *x* und Parametern $\vec{\omega}$

Probabilistische Modellierung: der Data-Likelihood

Likelihood für alle Daten (Data-Likelihood)

- Annahme: Daten unabhängig voneinander erzeugt
- dann gemeinsame Verteilung = Produkt der einzelnen Likelihoods:

$$L(\mathbf{w}) = P(T|X,\mathbf{w})$$

$$= \prod_{n=1}^{N} N(t_n|y(x_n,\mathbf{w}),\beta^{-1})$$

$$= \prod_{n=1}^{N} \frac{1}{\mathcal{N}} e^{-\frac{(t_n - y(x_n,\mathbf{w}))^2}{2\sigma^2}}$$

where $L(\vec{\omega})$ denotes the likelihood and

$\mathcal{N}$ is a the normalization constant

Maximum Likelihood

Parameteroptimierung

- Data-likelihood ist ein “stochastisches Datenmodell”
- L(w) ist Funktion *aller* Parameter des Datenmodells und von β
- Parameteroptimierung = Maximierung des Likelihood
- maximiert Wahrscheinlichkeit, die gemessenen Ausgaben zu beobachten, gegeben die Modellparameter und gegebene Eingaben

Vorgehen

- bilde den negativen log-Likelihood $-\log L(\omega)$
- dadurch wird Produkt zur Summe
- dann finde argmin $-\log L(\omega)$
- führt wieder auf Minimierung des quadratischen Fehlers ! (Übung)
- wir erhalten die optimalen Parameter ω_{ML}, β_{ML}

Maximum Likelihood

$$\mathbf{maximise}\ L(\mathbf{w})$$
$$\Leftrightarrow\ \mathbf{minimize}\ \ -logL(\mathbf{w})$$

Then

$$\ln p(\mathbf{t}|\mathbf{x},\mathbf{w},\beta) = \underbrace{-\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w})-t_n\}^2}_{\beta E(\mathbf{w})} + \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi)$$

Determine $\mathbf{w}_{\mathrm{ML}}$ by minimizing sum-of-squares error, $E(\mathbf{w})$.

$$\frac{1}{\beta_{\mathrm{ML}}} = \frac{1}{N}\sum_{n=1}^{N}\{y(x_n,\mathbf{w}_{\mathrm{ML}})-t_n\}^2$$

Rechnung Tafel

Generalisierung durch Max-Likelihood Parameter

Anwendung auf “neue” Daten

- verwende die ω_{ML}, β_{ML} Parameter
- dann ist die optimale Output-Verteilung

$$p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}) = \mathcal{N}\left(t|y(x, \mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}\right)$$

- die wahrscheinlichste Ausgabe für neues x ?

  Mittelwert $y(\omega_{ML}, x)$

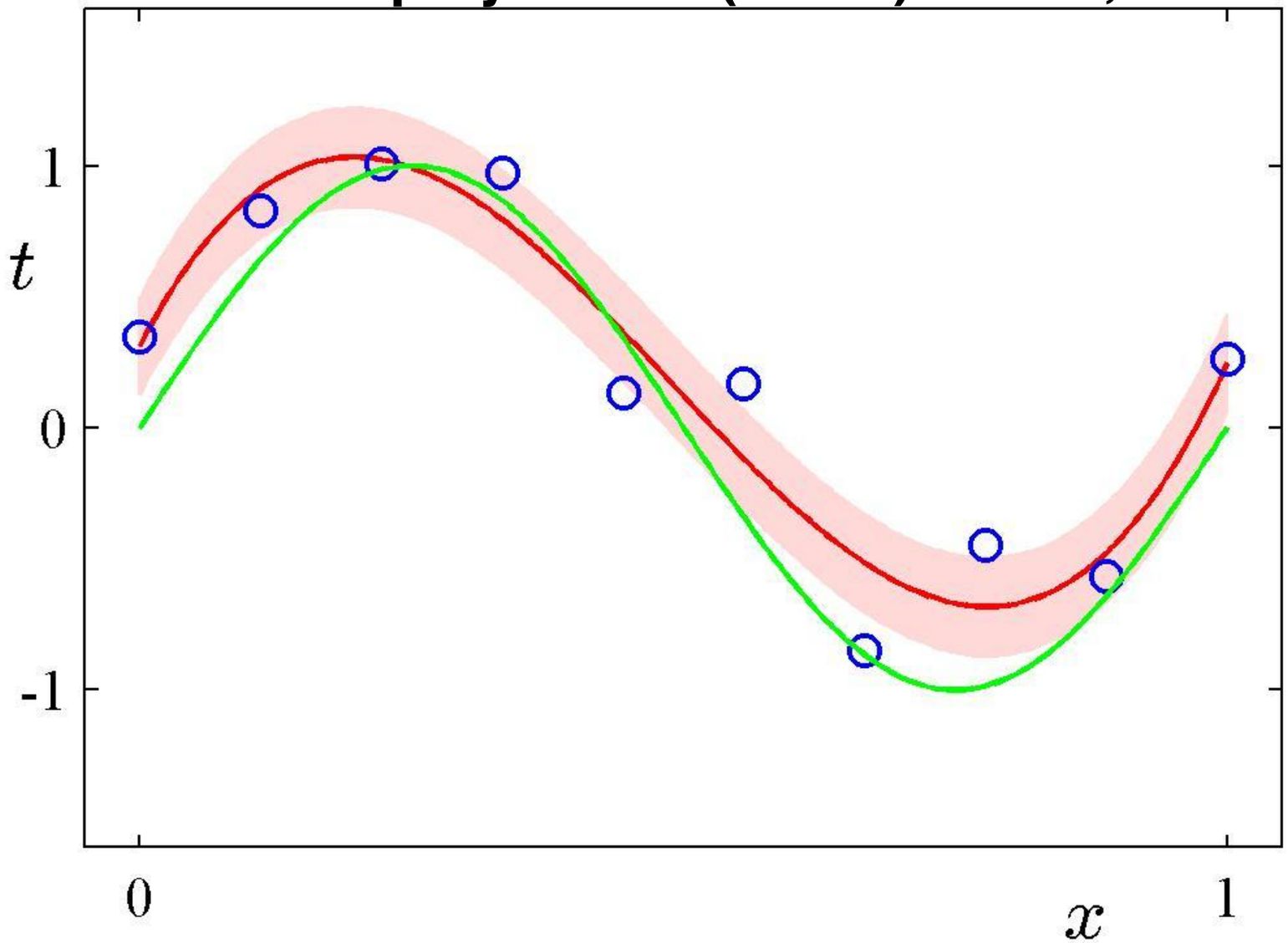
- aber: auch zufälliges Generieren von Ausgabe mit maximum Likelihood Verteilung möglich (sampling, generative Modell)

- β_{ML} gibt eine Konfidenz an

Generalisierung durch Max-Likelihood Parameter

$$p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}) = \mathcal{N}\left(t|y(x, \mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}\right)$$

wie vorher: polynomial (linear) model, M = 9

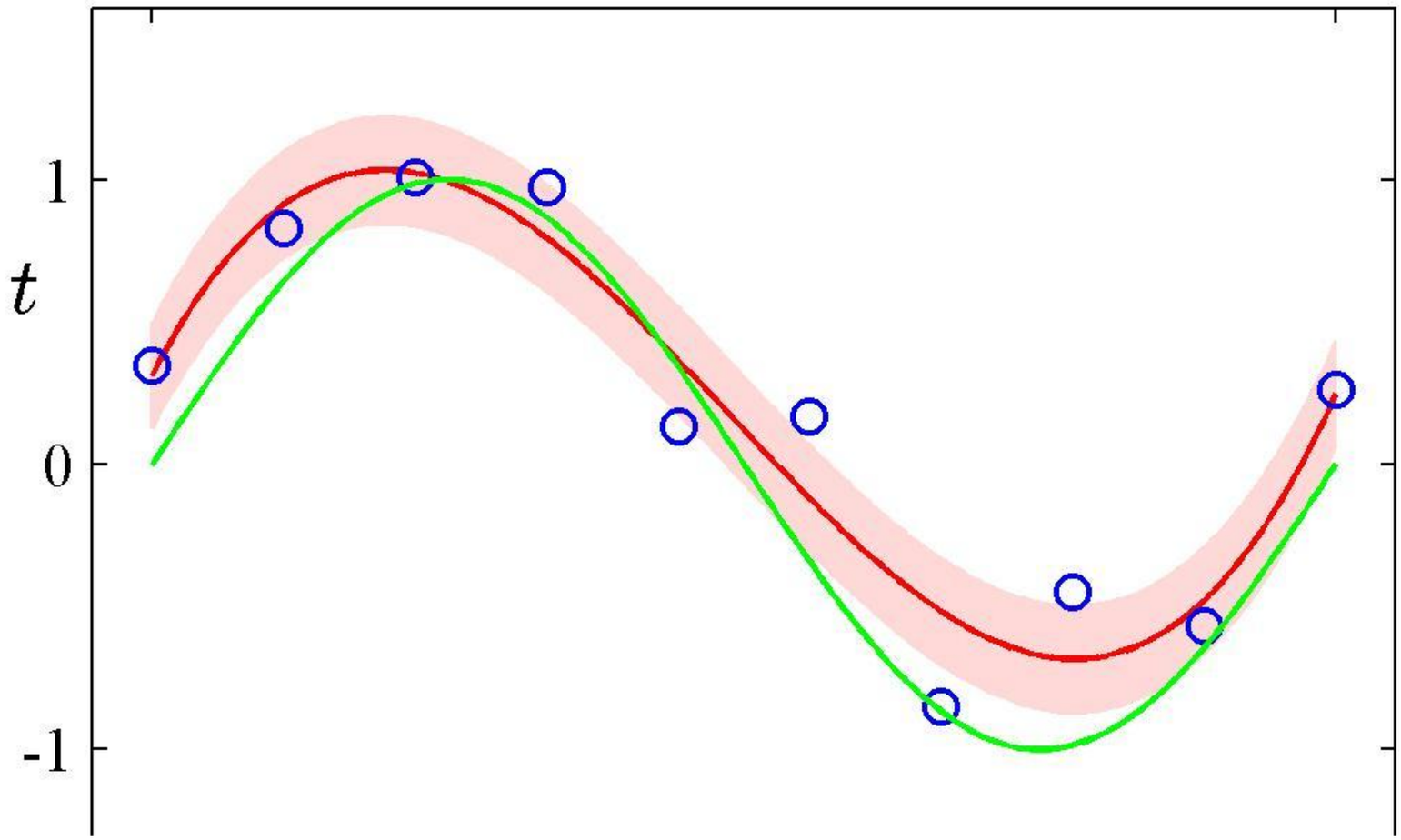


Slide from Bishop, machine learning

- **Was zeigt die Schattierung ? 1-sigma (std. deviation)**

Predictive Distribution

$$p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}) = \mathcal{N}\left(t|y(x, \mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}\right)$$



Zwischenfazit: Max-Likelihood als statistischer Ansatz

zugrundeliegende Annahmen:

- es gibt nur einen Datensatz
- es gibt einen “wahren, idealen” Satz Modellparameter
- ω_{ML}, β_{ML} sind “gute” Approximationen
- Modellierung der Unsicherheit durch Störung resultiert in Verteilung
- aber: Daten sind endlich und zufällig
- damit sind ω_{ML}, β_{ML} ebenfalls datenabhängig zufällig
- Unsicherheit in der Wahl des Datensatzes ist (noch) NICHT modelliert

[mögliche Verbesserung:

- Wiederholung des Experimentes => neuer Datensatz =>
  neue “beste” Parameter
- dann z.B. Mittelung über mehrere Experimente (später)]

Vollständiger Bayes'scher Ansatz

Interpretiere W-keiten als Wissen/Unsicherheit über Parameter

- Annahme: es gibt nur "einen" Datensatz, der unvollständig bekannt ist
- verwende wieder stochastische Datenmodell
- wenn neue/weitere Daten gemessen werden, dann ändert sich das Wissen/die Unsicherheit über die Parameter
- modelliere initiale Unsicherheit über die Parameter als $P(\omega)$
- $P(\omega)$ ist die a-priori W.-keit bevor Daten beobachtet werden
- modelliere den Likelihood wie vorher
- dann berechne die a-posteriori Wahrscheinlichkeit mit der Bayes-Formel:

$$P(w|D) = \frac{P(D|w)P(w)}{P(D)}$$

A-posteriori Wahrscheinlichkeit

$$p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta) \propto p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)p(\mathbf{w}|\alpha)$$

a-posteriori W.-keit **likelihood** **x** **prior**

(updated knowledge)

linke Seite:
Gaussfunktion,
(explizite Formel bekannt)

rechte Seite:
Produkt von
Gaussfunktionen

- **hier: initiale Unsicherheit abhängig von “Hyperparameter”** α

  (Hyperparameter: Parameter, der Verteilung von Parametern steuert)

Der Posterior

$P(\mathbf{w}|D)$ **heißt a-posteriori Verteilung oder einfach: “posterior”**

Die a-posteriori Verteilung drückt das Wissen/ die Unsicherheit über die Modellparameter aus, nachdem die Daten beobachtet wurden, welche selbst nur unter Unsicherheit beobachtet werden und damit durch stochastisches Datenmodell approximiert werden.

Generalisierung/Anwendung auf “neue” Daten

Maximum a-posteriori Parameter

- **bilde** $w_{MAP} = \text{argmax}_w \, P(\omega|D)$
- **dann generalisiere durch:**

$$t_{new} = y(x_{new}, \mathbf{w}_{MAP})$$

- **dies entspricht dem Mittelwert der posterior-Verteilung denn diese ist wiederum eine Gaussverteilung**

- **möglich wäre auch “Ziehen” eines Parameters w’ aus der posterior-Verteilung:**

$$w' \sim p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta)$$

und Generalisierung durch $y(x_{new}, w')$

°Anmerkung: x,t bezeichnen hier die gesamten Trainingsdaten

Maximum A-Posteriori Parameter

$$\begin{aligned} & \textbf{maximise } P(\mathbf{w}|D) \\ \Leftrightarrow\ & \textbf{minimize } -logP(\mathbf{w}|D) \end{aligned}$$

Then minimize:

$$\beta\widetilde{E}(\mathbf{w}) = \frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w}) - t_n\}^2 + \frac{\alpha}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

Determine $\mathbf{w}_{\mathrm{MAP}}$ by minimizing regularized sum-of-squares error, $\widetilde{E}(\mathbf{w})$.

- **wir sehen: Max-a-posteriori für gaußverteiltes stochastisches Datenmodell ist äquivalent zu Fehlerminimierung + Regularisierung**
- **Overfitting ist “automatisch” verhindert**

Max-Likelihood Berechnungen:

Woher kommt der Regularisierungsterm $\frac{\alpha}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$?

- **aus der Vornahme für den Prior (Achtung: induktiver Bias !)**

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I}) = \left(\frac{\alpha}{2\pi}\right)^{(M+1)/2} \exp\left\{-\frac{\alpha}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}\right\}$$

$$p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta) \propto p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)p(\mathbf{w}|\alpha)$$

- **dann logarithmieren …** (Rechnung Tafel)

Predictive Distribution

Vollständige Berücksichtigung von Unsicherheit

- **bekannt: Data-Likelihood und Parameter posterior**
- **integriere über alle mögliche Parameter gewichtet mit ihrer Wahrscheinlichkeit:**

$$p(t|x,\mathbf{x},\mathbf{t}) = \int p(t|x,\mathbf{w})p(\mathbf{w}|\mathbf{x},\mathbf{t})\,\mathrm{d}\mathbf{w} = \mathcal{N}\left(t|m(x),s^2(x)\right)$$

- **diese W.-keit kann für lineare Modelle ebenfalls explizit berechnet werden und ist gaußverteilt (d.h.** $m(x), s^2(x)$ **sind bekannt** => später**)**
- **dann generalisiere durch Ziehen aus°**

$$P(t_{new}|x_{new},X,T)\,(= p(t|x,\mathbf{x},\mathbf{t}) \text{ in Bishop notation})$$

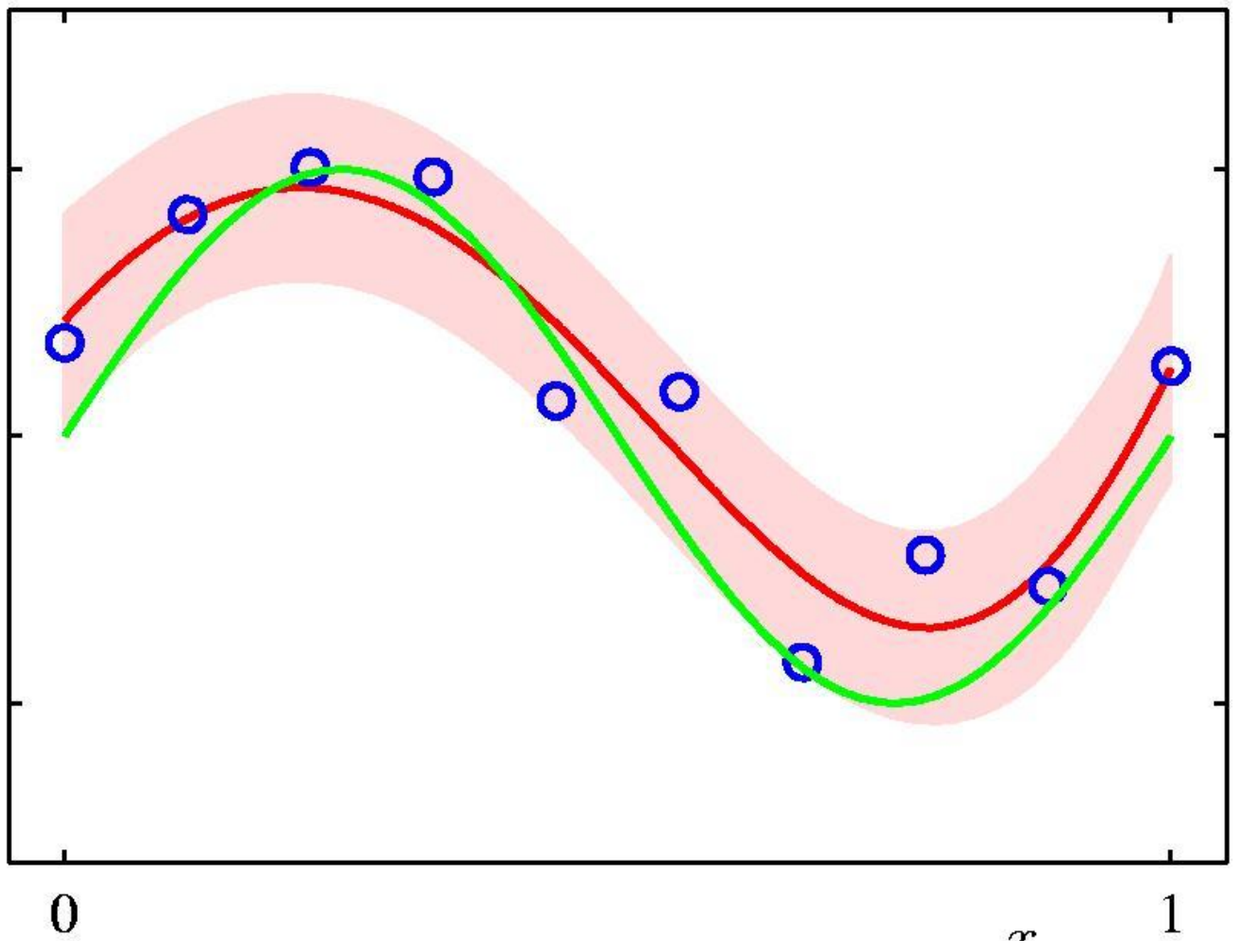
- **diese Verteilung heißt *"predictive distribution"***

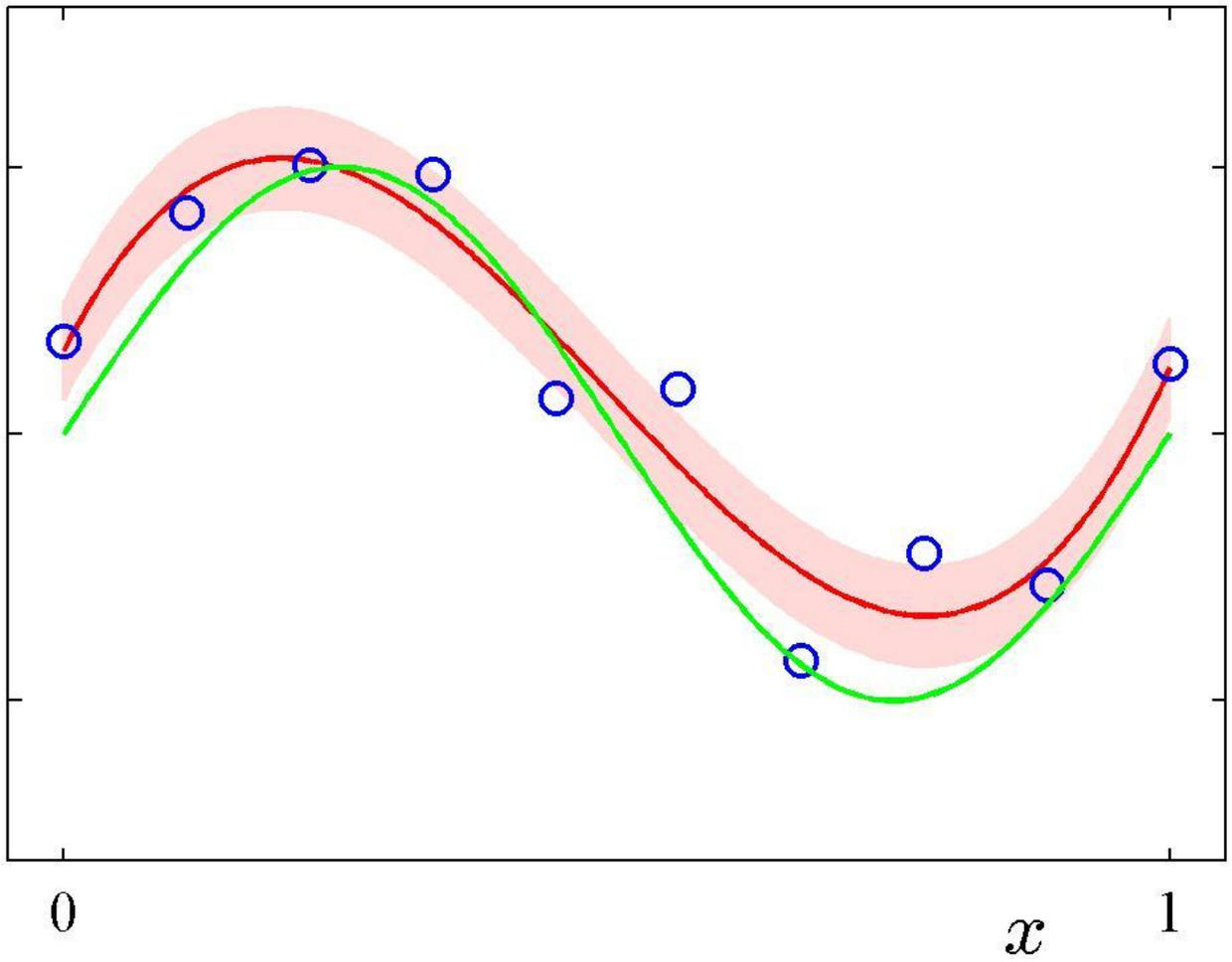
(° i.e. ziehe zufällig aus der Verteilung P(t|x,X,T))

Predictive Distribution vs. Maximum Likelihood

$$p(t|x, \mathbf{x}, \mathbf{t}) = \mathcal{N}\left(t|m(x), s^2(x)\right)$$

$$p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}) = \mathcal{N}\left(t|y(x, \mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}\right)$$





Illustrations from Bishop, machine learning

der volle Bayes'sche Ansatz zeigt mehr Unsicherheit, da die Unsicherheit in den Parametern auch modelliert ist !

Take home:

Unsicherheit

- Rauschen/Unsicherheit in Daten ~ Normalverteilung
- Bilde Likelihood und Data-Likelihood
- Minimierung von -log L(w) => maximum Likelihood Parameter
- Generalisierung durch Predictive Distribution:
  - wahrscheinlichster Wert (Mittelwert) $y(\omega_{ML}, x)$
  - Konfidenz gegeben durch die Präzision (Inverse der Varianz der Verteilung)

Bayes'scher Ansatz:

- interpretiere W-keiten als Unsicherheiten
- Modellierung wir vorher
- zusätzlich: a-priori Annahme über Parameter (prior)
- Inferenzschritt auf P(w|D) => max posterior