# Kapitel 3

# Nicht-parametrische Regression

## 3.2 Splines

Fragen? https://www.menti.com/alneo5om9sq5.

# Probleme des Nadaraya-Watson-Schätzers

Erinnerung: Bei der **nicht-parametrischen Regression** schätzen wir das Modell

$$Y_i = m(x_{i1}) + \epsilon_i, \quad i = 1, \ldots, n$$

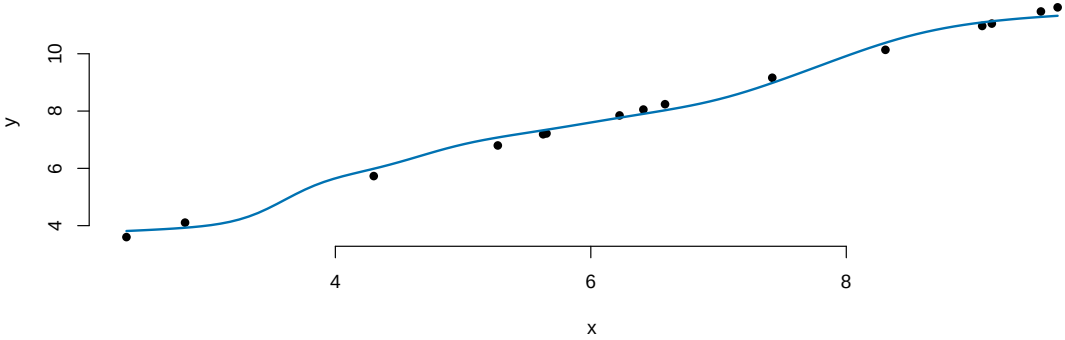
wobei $m$ eine nicht näher spezifizierte, aber möglichst glatte Funktion ist.

Bisher haben wir $m$ mit Hilfe von Kernfunktionen geschätzt, konkret mit dem Nadaraya-Watson-Schätzer (NWS). Dieser Schätzer hat jedoch einige Probleme:

1. Schätzergebnisse sind unzufriedenstellend, wenn $m$ tatsächlich linear ist
2. schlechte Eigenschaften im Randbereich der $x$-Werte (Bias)
3. schlechte Eigenschaften in Bereichen, in denen keine $x$-Werte liegen

# Illustration von 1.

Illustration von 1.: aus perfekt linearem Modell[1] generierte Daten und NWS.

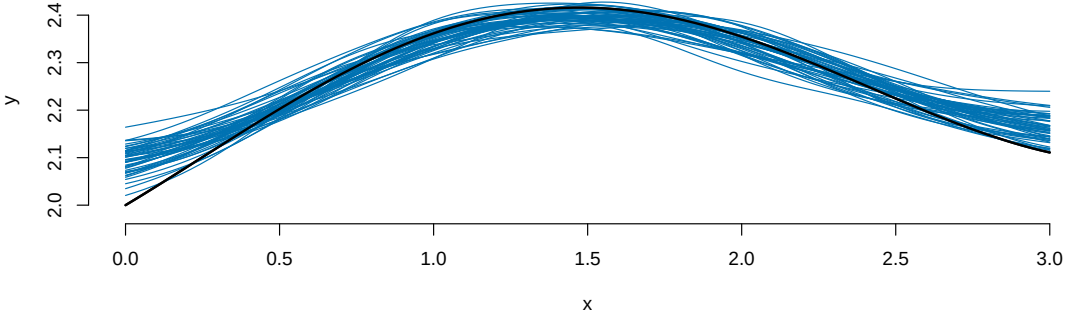



$\rightsquigarrow$ ein sinnvoller Schätzer sollte hier eine Gerade ergeben!

[1] d.h. $Y_i = \beta_0 + \beta_1 x_{i1}$, keine zufälligen Fehlerterme

# Illustration von 2.

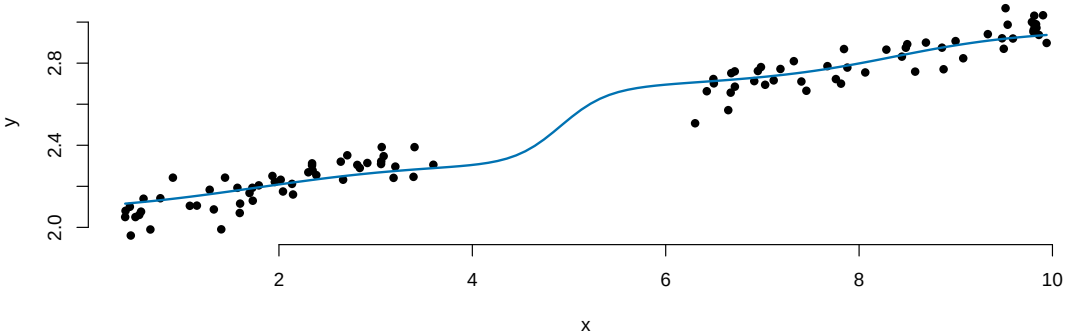
50 NWS (blaue Linien) für 50 simulierte Datensätze aus einem Regressionsmodell (schwarze Linie zeigt wahre Regressionsfunktion):




$\rightsquigarrow$ systematische Verzerrung ("Ausfransung") in den Randbereichen

# Illustration von 3.

Aus einem linearen Modell simulierte Daten und NWS:

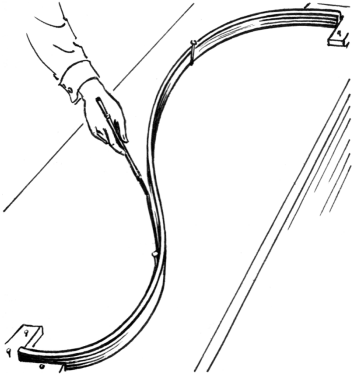



$\rightsquigarrow$ ein sinnvoller Schätzer sollte den Bereich fehlender $x$-Werte “logisch” füllen

Fragen? https://www.menti.com/alneo5om9sq5.

# B-Splines

Zur Herkunft von B-Splines:



Fragen? https://www.menti.com/alneo5om9sq5.

# B-Splines

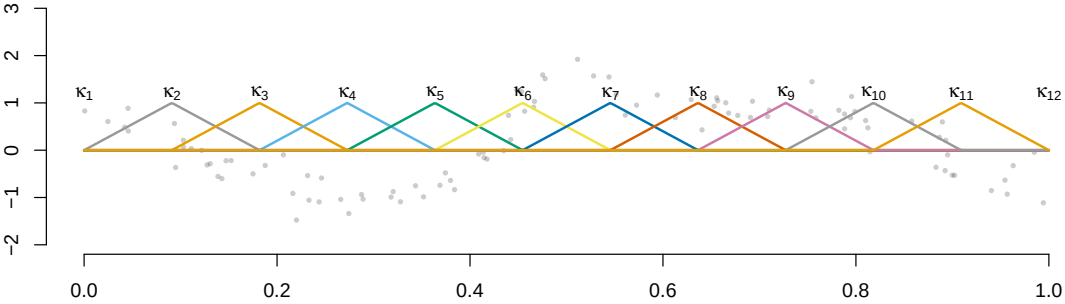
Grundidee: konstruiere $m(x)$ als gewichtete Summe fester **Basisfunktionen**:

$$\begin{aligned} Y_i &= m(x_{i1}) + \epsilon_i \\ &= \gamma_1 B_1(x_{i1}) + \gamma_2 B_2(x_{i1}) + \ldots + \gamma_k B_k(x_{i1}) + \epsilon_i \end{aligned}$$

Die $B_1, \ldots, B_k$ sind dabei B-Spline Basisfunktionen. Illustration (in der Vorlesung):

# B-Spline Basisfunktionen

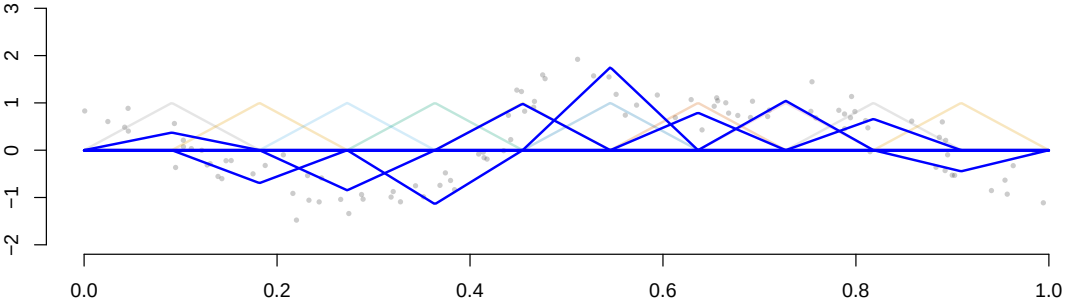
Die B-Spline Basisfunktionen “stellen” wir dann in unser Streudiagramm, wobei die Position für jede Basisfunktion durch drei Knoten $\kappa_j$ bestimmt wird:




Fragen? https://www.menti.com/alneo5om9sq5.

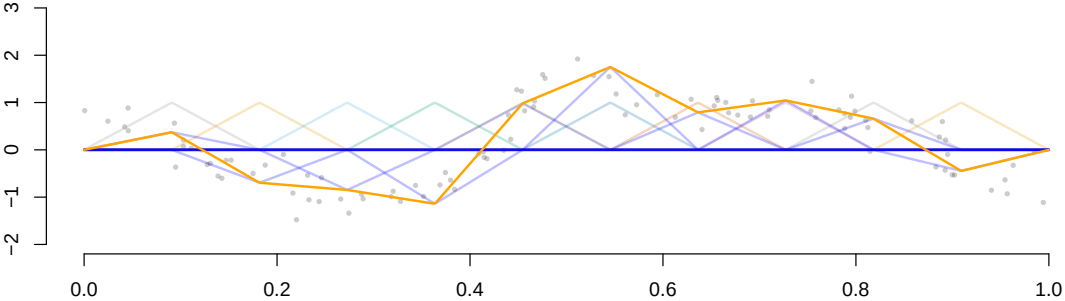
# Gewichtete B-Spline Basisfunktionen

Die Parameter $\gamma_j$ schätzen wir über Minimierung der Fehlerquadratsumme, wodurch sich die gewichteten Basisfunktionen ergeben (in blau):




Fragen? https://www.menti.com/alneo5om9sq5.

Die finale geschätzte Regressionsfunktion (in orange) wird gebildet, indem man die gewichteten Basisfunktionen aufsummiert:



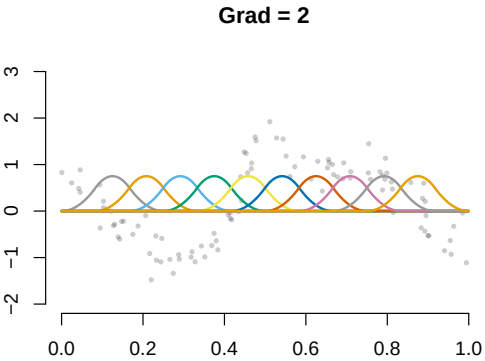

$\rightsquigarrow$ hierdurch ergibt sich eine plausible Regressionsfunktion

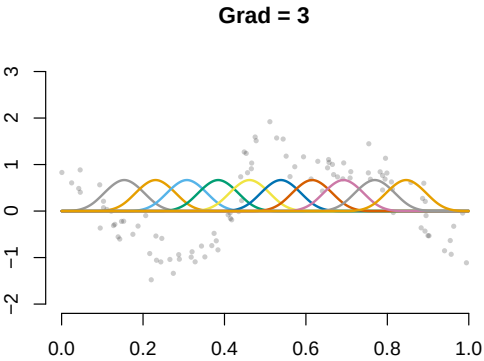
$\rightsquigarrow$ $\hat{m}(x)$ ist jedoch keine glatte Funktion!

Fragen? https://www.menti.com/alneo5om9sq5.

# Glatte Regressionsfunktionen mit B-Splines

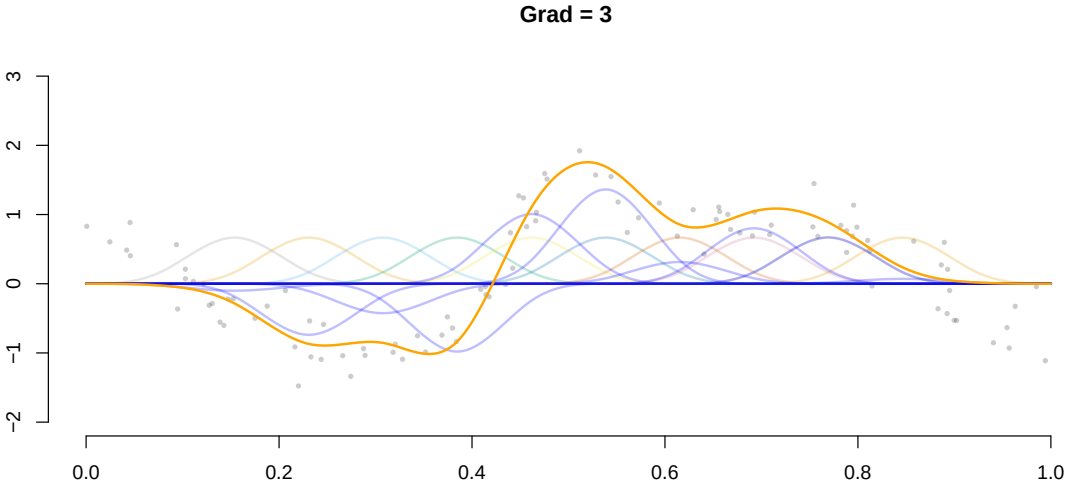
Um eine glatte Regressionsfunktion zu erhalten, verwenden wir Basisfunktionen mit einem höheren Grad:




Fragen? https://www.menti.com/alneo5om9sq5.

# Regressionsfunktion für B-Splines vom Grad 3




Fragen? https://www.menti.com/alneo5om9sq5.

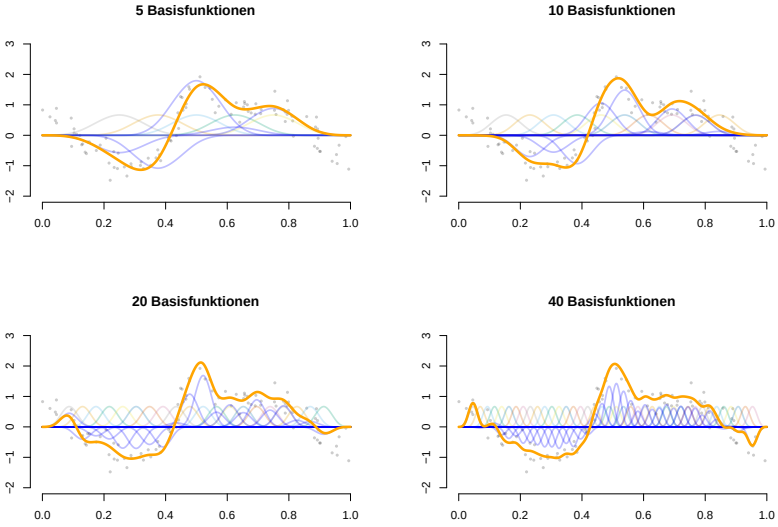
# B-Splines: Knotenwahl

Die geschätzte Regressionsfunktion hängt insbesondere ab von. . .

- . . . dem Grad der Splines (sollte mindestens drei sein)
- . . . der Anzahl der Basisfunktionen/Knoten
- . . . der Platzierung der Knoten

Der Einfluss des Grads und der Anzahl der Knoten lässt sich mithilfe folgender Shiny-App untersuchen:

https://lehre-rdusl.shinyapps.io/Splines_1D/.




Fragen? https://www.menti.com/alneo5om9sq5.

# Kontrolle der Flexibilität der konstruierten Funktion

Wie finden wir eine Regressionsfunktion mit Splines?

- Je mehr Basisfunktionen, desto mehr Flexibilität, sodass die Regressionsfunktion immer genauer den Daten folgen wird ($\rightsquigarrow$ Varianz hoch, Bias niedrig).
- Zu wenige Basisfunktionen: Regressionsfunktion kann Muster möglicherweise nicht genau abbilden ($\rightsquigarrow$ Bias hoch, Varianz niedrig).
- Wie finden wir den richtigen Mittelweg im Sinne des Bias-Varianz-Trade-offs? Wir wollen eine Funktion, die die Daten gut erklärt (d.h. nah an den Punkten liegt) und dabei möglichst glatt ist (d.h. nicht zu unruhig)

Wir nehmen viele Basisfunktionen (z.B. 40), sodass zunächst einmal fast beliebig viel Flexibilität vorhanden ist — wir penalisieren dann aber Krümmung der geschätzten Funktion.

# P-Splines

Der Ausdruck

$$\int_a^b m''(x)^2 dx$$

ist ein Maß dafür, wie unruhig die Regressionfunktion ist. Diese wird bei Regressionen mit B-Splines über Differenzen zweiter Ordnung approximiert:

> Nichtparametrische Regression durch penalisierte B-Splines (bzw. kurz: **P-Splines**) beinhaltet die Minimierung des folgenden Zielkriteriums:
>
> $$\sum_{i=1}^{n} \Big( y_i - \big(\gamma_1 B_1(x_{i1}) + \ldots + \gamma_k B_k(x_{i1})\big)\Big)^2 + \lambda \sum_{k=3}^{K} (\Delta^2 \gamma_k)^2$$

Hierbei bezeichnet $\Delta^2$ die Differenzen 2. Ordnung, d.h.

$$\Delta^2 \gamma_k = \Delta(\gamma_k - \gamma_{k-1}) = (\gamma_k - \gamma_{k-1}) - (\gamma_{k-1} - \gamma_{k-2})$$

$(\Delta^2 \gamma_k) = 0$ gilt genau dann, wenn $\gamma_{k-2}, \gamma_{k-1}$ und $\gamma_k$ auf einer Gerade liegen, die Krümmung in diesem Bereich also gleich 0 ist. Zusätzlich ist $\lambda$ ein (festzulegender) **Glättungsparameter**.

# P-Splines

Der Glättungsparameter $\lambda$ im Zielkriterium

$$\sum_{i=1}^{n} \Big( y_i - \big(\gamma_1 B_1(x_{i1}) + \ldots + \gamma_k B_k(x_{i1})\big) \Big)^2 + \lambda \sum_{k=3}^{K} (\Delta^2 \gamma_k)^2$$

bestimmt, wie wichtig uns “Biegsamkeit” der Funktion ist:

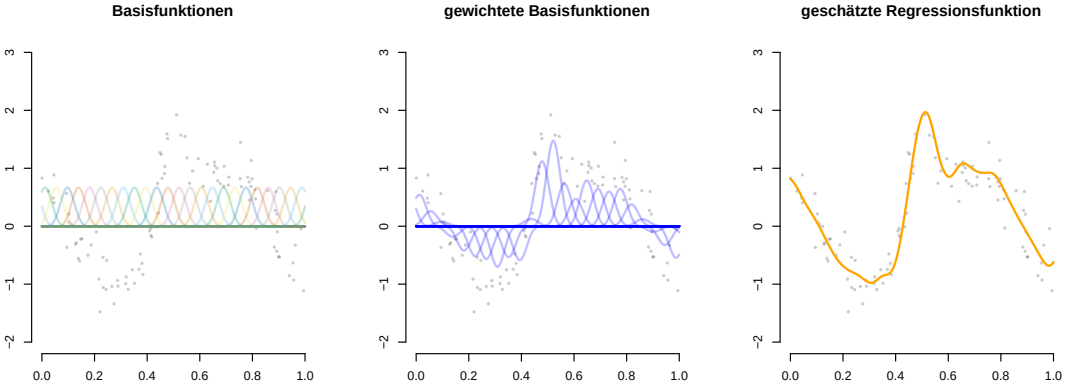
$$\begin{aligned} \lambda \to 0 \quad &\rightsquigarrow \quad m \text{ verläuft sehr unruhig} \\ \lambda \to \infty \quad &\rightsquigarrow \quad m \text{ wird zu einer Geraden} \end{aligned}$$

Insbesondere erhalten wir wieder den bekannten Bias-Varianz-Trade-off:

- niedriges $\lambda$ führt zu niedrigem Bias aber hoher Varianz
- hohes $\lambda$ führt zu niedriger Varianz aber hohem Bias

Der Glättungsparameter $\lambda$ kann wieder z.B. per Kreuzvalidierung ausgewählt werden, aber auch eine Wahl nach Augenmaß ist möglich.

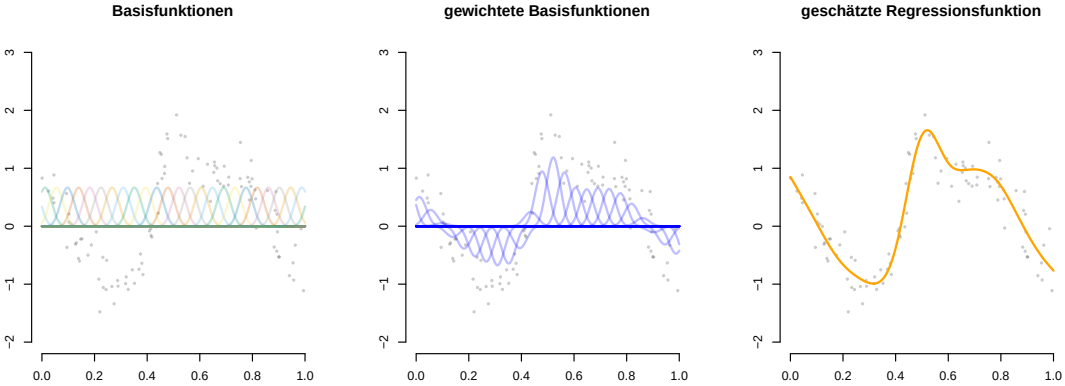
P-Spline-Schätzung mit $\lambda = 0.1$:




Niedriger Strafterm für Krümmung $\rightsquigarrow$ **Overfitting**!

Fragen? https://www.menti.com/alneo5om9sq5.

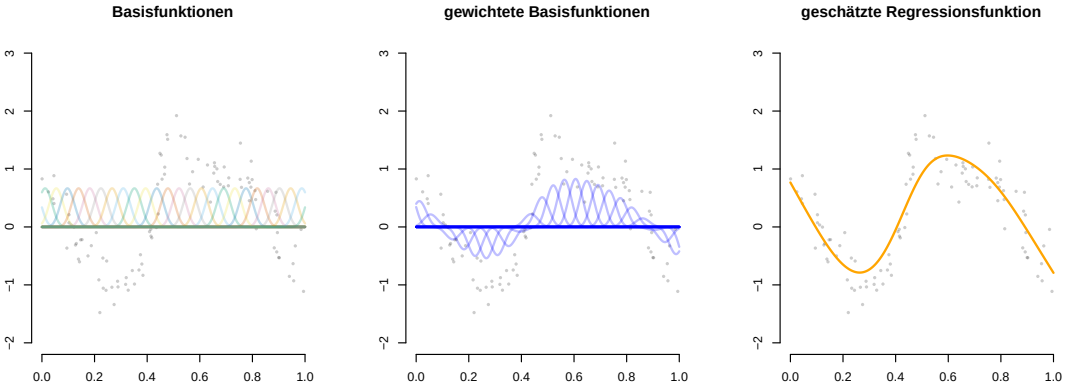
P-Spline-Schätzung mit $\lambda = 2$:




Moderater Strafterm für Krümmung $\rightsquigarrow$ sieht gut aus!

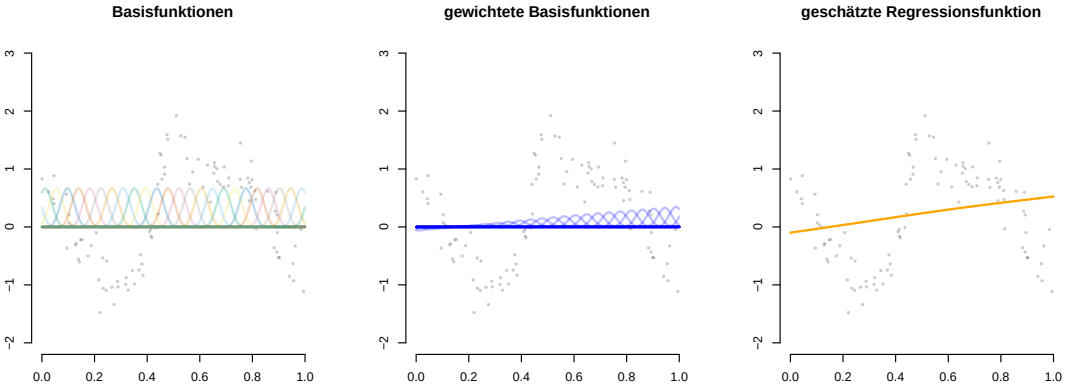
Fragen? https://www.menti.com/alneo5om9sq5.

P-Spline-Schätzung mit $\lambda = 200$:



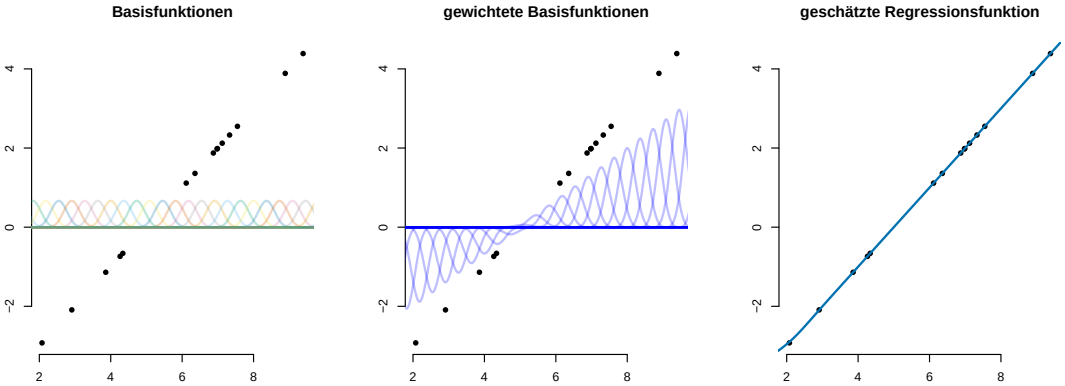

Zu hoher Strafterm für Krümmung $\rightsquigarrow$ **Underfitting**, d.h. zu starke Betonung auf Glattheit (Unterschätzung von Gipfeln & Überschätzung von Tälern).

P-Spline-Schätzung mit $\lambda = 500000$:




Sehr hoher Strafterm für Krümmung $\rightsquigarrow$ man erhält als Schätzer die am besten passende Funktion ohne jegliche Krümmung, also eine Gerade.

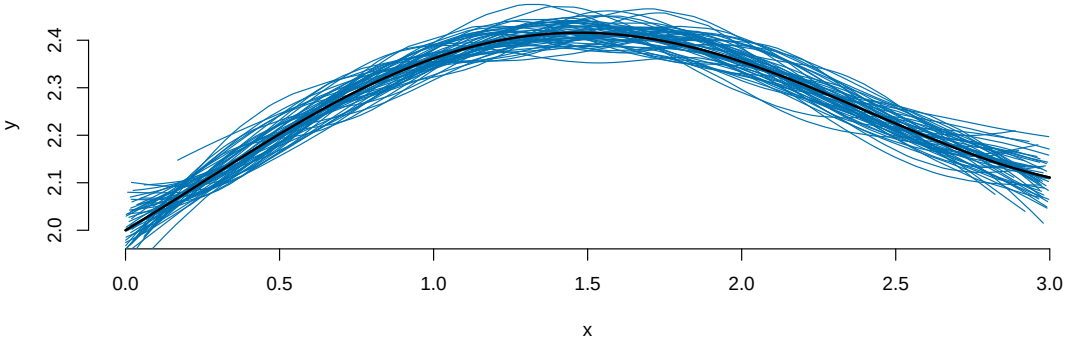
# Verhalten von P-Splines, wenn die wahre Funktion exakt linear ist




Wenn die Datenpunkte exakt auf einer Geraden liegen, dann ist die mit P-Splines geschätzte Regressionsfunktion auch exakt linear.

# Verhalten von P-Splines in den Randbereichen

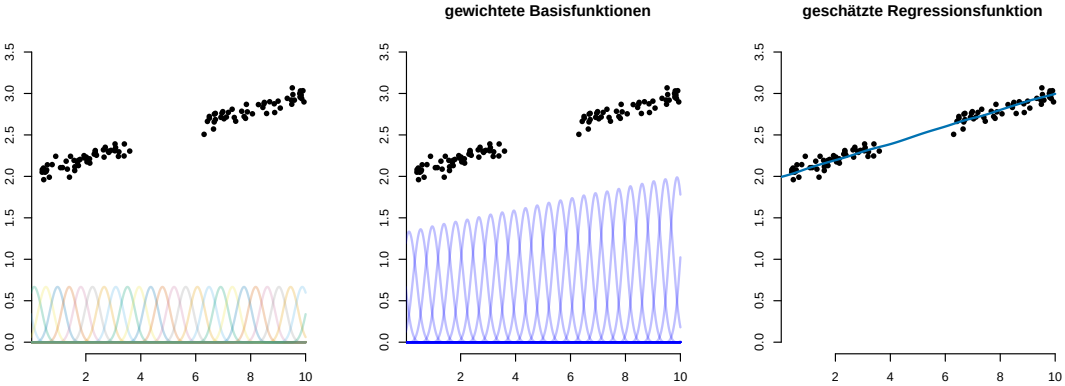
50 P-Spline-Schätzer (blaue Linien), erhalten für 50 simulierte Datensätze aus einem Regressionsmodell (schwarze Linie):




Sieht gut aus: keine “Ausfransung” in den Randbereichen

Fragen? https://www.menti.com/alneo5om9sq5.

# Verhalten von P–Splines in Bereichen fehlender $x$-Werte




Der P-Spline-Schätzer überbrückt sinnvoll den Bereich fehlender $x$-Werte.

Fragen? https://www.menti.com/alneo5om9sq5.

# Erinnerung: “Curse of dimensionality”

Kann man ein Modell der Form

$$Y_i = m(x_{i1}, \ldots, x_{ip}) + \epsilon_i, \quad i = 1, \ldots, n$$

auch mit $p > 2$ sinnvoll nichtparametrisch schätzen?

Antwort: ja, aber der Schätzer ist dann sehr “datenhungrig”: Um den 3D-Raum ähnlich gut abzudecken wie den 1D-Raum braucht man also 100 Mal so viele Daten (und nicht etwa nur 3 Mal so viele).

Kurzum: die nötige Stichprobengröße, um $m$ sinnvoll zu schätzen, steigt rasant mit steigendem $p$. Man spricht vom “curse of dimensionality”.

# Additive Zerlegung der Regressionsfunktion

In der Praxis wird meist eine **additive Struktur** angenommen, d.h.

$$m(x_{i1}, \ldots, x_{ip}) = m_1(x_{i1}) + \ldots + m_p(x_{ip})$$

Hiermit wird das Problem des “curse of dimensionality” umgangen:

- effektiv haben wir so ein univariates Schätzproblem pro Variable
- im univariaten Fall liegen mehr Datenpunkte pro Umgebung, die lokale Schätzung funktioniert hier!

$\rightsquigarrow$ restriktive Annahme, da Interaktionseffekte nicht mehr abgebildet werden

$\rightsquigarrow$ wegen des “curse of dimensionality” hat man oft aber keine andere Wahl

# Generalisierte additive Modelle (GAMs)

**Generalisierte additive Modelle** (GAMs) beinhalten drei Komponenten:

1. Verteilungsannahme für die Zielvariable (z.B. Poisson, Bernoulli)
2. additiver Prädiktor mit nichtlinearen Effekten der erklärenden Variablen:

$$\eta_i = \beta_0 + m_1(x_{i1}) + \ldots + m_p(x_{ip})$$

   (einzelne $m_j$ können aber auch lineare/quadratische Funktionen sein)

3. Transformation des linearen Prädiktors, damit $E(Y)$ den für die gegebene Verteilungsannahme richtigen Wertebereich hat (z.B. exp bei Poisson)

$\rightsquigarrow$ flexibler als parametrische Modelle aus Kapitel 2 (wegen nicht-linearer Effekte)

$\rightsquigarrow$ weniger flexibel als nicht-parametrische Modelle, in denen $m(x_{i1}, \ldots, x_{ip})$ ohne Annahme der additiven Zerlegung geschätzt wird

$\rightsquigarrow$ in der anwendungsorientierten Forschung inzwischen sehr populär

# GAM mit normalverteilter Zielgröße

Mögliches GAM für eine normalverteilte Zielvariable $Y_i$:

$$Y_i \sim N(\mu_i, \sigma^2),$$
$$\mu_i = E(Y_i) = \beta_0 + m_1(x_{i1}) + \ldots + m_p(x_{ip})$$

Im Vergleich zum linearen Modell werden also einfach die linearen Effekte $\beta_j x_{ij}$ durch (nicht-parametrisch zu schätzende) nicht-lineare Effekte $m_j(x_{ij})$ ersetzt.

Beispiel:

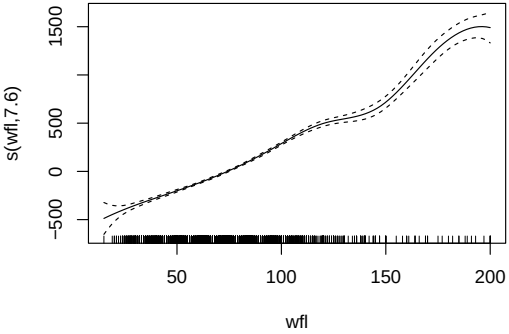
$$\text{Miete}_i \sim N(\mu_i, \sigma^2), \quad \mu_i = E(\text{Miete}_i) = \beta_0 + m_1(\text{Wfl}_i) + m_2(\text{Baujahr}_i)$$

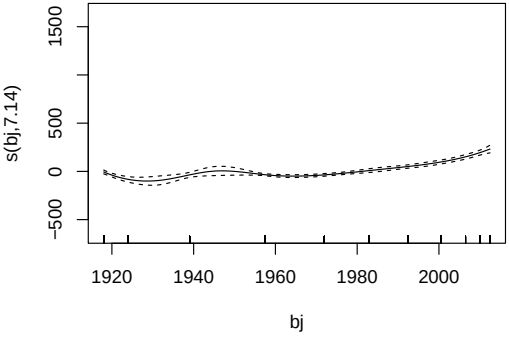
In R:

```
install.packages("mgcv")
library(mgcv)
mod1 = gam(nm ~ s(wfl, bs = "ps") + s(bj, bs = "ps"), data = miete)
plot(mod1)
```

# GAM im Mietspiegel-Beispiel

Geschätzte nicht-lineare Effekte $\hat{m}_1(\text{Wfl}_i)$ und $\hat{m}_2(\text{Baujahr}_i)$:




Fragen? https://www.menti.com/alneo5om9sq5.

# GAM mit Poisson- und Bernoulli-verteilter Zielgröße

Mögliches GAM für eine Zählvariable $Y_i$:

$$Y_i \sim Po(\lambda_i),$$
$$\lambda_i = E(Y_i) = e^{\beta_0 + m_1(x_{i1}) + \ldots + m_p(x_{ip})}$$

Mögliches GAM für eine binäre Zielvariable $Y_i$:

$$Y_i \sim Bern(\pi_i),$$
$$\pi_i = E(Y_i) = \frac{e^{\beta_0 + m_1(x_{i1}) + \ldots + m_p(x_{ip})}}{1 + e^{\beta_0 + m_1(x_{i1}) + \ldots + m_p(x_{ip})}}$$

$\rightsquigarrow$ analog zur Poisson- und Bernoulli-Regression aus Kapitel 2, nur mit nicht-parametrisch zu schätzenden Effekten.

# GAM im Donnerparty-Beispiel

Das Modell

$$\text{Überleben}_i \sim Bern(\pi_i), \quad \pi_i = \frac{e^{\beta_0+\beta_1\cdot\text{Geschlecht}_i+\beta_2\cdot\text{Famgroesse}_i+m_1(\text{Alter}_i)}}{1+e^{\beta_0+\beta_1\cdot\text{Geschlecht}_i+\beta_2\cdot\text{Famgroesse}_i+m_1(\text{Alter}_i)}}$$

kann in R wie folgt geschätzt werden:

```
mod2 = gam(Ueberleben ~ Geschlecht + Familiengroesse + s(Alter, bs = "ps"),
           family = "binomial", data = donner_party)
plot(mod2)
```

Zur Anwendung von `gam()` allgemein:

- Verteilungsannahme wird mit family-Option getroffen (wie auch bei `glm()`)
- Angabe der im Prädiktor auftauchenden erklärenden Variablen
- nichtparametrische Effektdurchschätzung durch `s(...)` (sonst linear)
- eine ggf. nötige Transformation (wie z.B. exp) wird automatisch genutzt
- `plot(gam(...))` visualisiert die geschätzten nichtlinearen Effekte