

### 3.3) Modellselektion & Regularisierung

202

Schon gesehen: Bei einer großen Anzahl an verfügbaren Prädiktoren  $X_1, \dots, X_p$  ist die Gefahr groß, auch unwichtige Größen in ein Modell aufzunehmen.

z.B.  $X_1 = X, X_2 = X^2, X_3 = X^3, \dots$  beliebig erweiterungsfähig:

$$Y = b_0 + b_1 X_1 + b_2 X_2 + b_3 X_3 + \dots = b_0 + b_1 X + b_2 X^2 + b_3 X^3 + \dots$$

Insbesondere: bei einer (zu) geringen Anzahl an Trainingsdaten ( $n \approx p$ ) werden die Parameterschätzungen instabil

Für  $n < p$  gibt es z.B. keine Möglichkeit mehr ein lineares

Modell

$$Y = b_0 + b_1 X_1 + \dots + b_p X_p + \varepsilon$$

zu fitten!

Dafür gibt's unendlich viele Möglichkeiten — 3 Beispiele

### SUBSET SELECTION

sinnvolle Teilmenge  
aus  $X_1, \dots, X_p$   
(möglichst automa-  
tisiert) finden

### SHRINKAGE REGULARI- SIERUNG

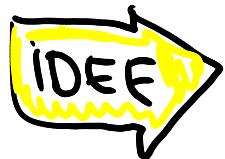
Modellparameter gegen  
 $0$  gehen lassen ...  
Neuro: Weight Decay

### DIMENSION REDUCTION

Projektion auf  
(günstig gewählten)  
 $M$ -dim. Unterraum  
z.B. PCA, Autoencoder

### 3.3.1) Subset Selection

204



Korre das Modell "entscheiden" (schon gesehen p-Werte bei linearem Modell aus Hypothesentests)

Man legt also zunächst die geeignete Modellklasse fest und untersucht dann den Einfluss verschiedener Prädiktoren auf die Modellgüte

Fehlermaße:

|             |  |
|-------------|--|
| RSS         | $RSS = \sum_{i=1}^n (y_i - \hat{y}_i)^2$                     |
| RMSE        | $RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$ |
| $\bar{R}^2$ | $\bar{R}^2 = 1 - (1-R^2) \frac{(n-1)}{n-(p+1)}$              |
| :           | :  |

Mit Hilfe dieser Fehlermaße  
⇒ Bewertung der Modellgüte für verschiedene Teilmengen von  $X_1, \dots, X_p$

↳ SUBSET SELECTION

### 3 STRATEGIEN: Faktoren: $X_1, \dots, X_p$

205

①

#### BEST SUBSET SELECTION

- ① Sei  $M_0$  das Modell ohne Prädiktoren
- ② für  $k=1, \dots, p$ :
  - fitte alle  $\binom{p}{k}$  Modelle mit genau  $k$  Prädiktoren
  - wähle das Beste davon aus und nenne es  $M_k$
- ③ Wähle das Beste Modell aus den gefundenen  $M_0, M_1, \dots, M_p$



Hier kann man den Trainingsfehler verwenden

Auch  $R^2$  (ohne Anpassung) möglich, da alle Modelle die gleiche Komplexität haben



Hier ist der Testfehler wichtig (Generalisierungsfähigkeit!)

Man braucht  $\bar{R}^2$  oder Cross-validation (Bootstrap)

## FORWARD STEPWISE SELECTION

① Sei  $M_0$  das Modell ohne Prädiktoren

② für  $k=0, 1, \dots, p-1$ :

- betrachte alle  $p-k$  Modelle, die zu den Prädiktoren im  $M_k$  noch einen zusätzlichen berücksichtigen
- Wähle das beste dieser  $p-k$  Modelle aus und nenne es  $M_{k+1}$

③ Suche ein besten Modell aus den gefundenen  $M_0, M_1, \dots, M_p$

 In jedem Schritt wird genau diejenige Variable ins Modell aufgenommen, die den größten Nutzen bringt (reduzierter Rechenaufwand)

 Auswahl des Top-Modells ist nicht sichergestellt

Ähnlich: Backward Selection (siehe Lecture 5, p.94) für Lineare Regressionsmodelle  
→ Metriken ändern sich – Strategie bleibt gleich

(3)

## BACKWARD STEPWISE SELECTION

 $(n \gg p)$ 

① Sei  $M_p$  das "volle" Modell, das alle  $p$  Prädiktoren enthält

② für  $k = p, p-1, \dots, 1$ :

- betrachte alle  $k$  Modelle, die alle bis auf einen der Prädiktoren im  $M_k$  enthalten (  $k-1$  Prädiktoren müssen untersucht werden! )

- Wähle das Beste dieser  $k$  Modelle und nenne es  $M_{k-1}$

③ Wähle ein bestes Modell aus  $M_0, \dots, M_p$

... auch hybride Lösungen sind möglich  
i.e.: ②  $\leftrightarrow$  ③

zur

hier wird das Modell auf dem kleinsten RSS, RMSE oder größten  $R^2$  gewählt

mit

hier braucht man die CrossValidation  $\Rightarrow C_p$  oder  $\bar{R}^2$



Modelle haben alle unterschiedliche Komplexität!



Auswahl des Top-Modells ist Nicht garantiert!

**FRAGE**

Wann ist welche Metrik geeignet um das "beste" Modell zu finden?

In den ersten Schritten werden jeweils Modelle mit gleicher Komplexität verglichen. Hier ist es nur wichtig, welche Features ins Modell eingehen

⇒ der Trainingsfehler reicht für eine qualitative Entscheidung aus!

geeignete Maße sind  $\text{MSE}$ ,  $\text{RMSE}$ ,  $R^2$  ← auch wenn klar ist, dass der Trainingsfehler keine gute Approximation für den Testfehler darstellt!

→ Wollen wir aus einer Menge von Modellen mit unterschiedlicher Komplexität das beste Modell auswählen ⇒

① weil das komplexeste Modell immer den kleinsten RMSE (größtes  $R^2$ ) hat brauchen wir eine Anpassung, die das rausrechnet

② wir sollten den Testfehler auswerten (Cross Validation, Bootstrapping)

Mögliche Anpassungen der Fehlermaße ( $X_1, \dots, X_p$  Features im Modell,  $n$  Datenpunkte)

209

sich gesehen:

① Angepasstes Bestimmtheitsmaß  $\bar{R}^2$ :

$$R^2 = \frac{SSE}{SQT} = \frac{\sum (\hat{y}_i - \bar{y})^2}{\sum (y_i - \bar{y})^2} \Rightarrow \boxed{\bar{R}^2 = 1 - (1 - R^2) \frac{(n-1)}{n-(p+1)}}$$

② AIC (Akaike Information Criterion) (adjustierter RSS)

$$AIC = \frac{1}{n \hat{\sigma}^2} (RSS + 2p \hat{\sigma}^2)$$

$$\text{mit } RSS = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$\hat{\sigma}$   $\hat{\sigma}$  Standardfehler der Schätzung  
für das Modell mit allen  $X_i$  ( $i=1 \dots p$ )

③ BIC (Bayesian Information Criterion) (adjustierter RSS)

$$BIC = \frac{1}{n \hat{\sigma}^2} (RSS + \log(n) p \hat{\sigma}^2)$$

← wegen  $n > 2 \Rightarrow BIC$  bestraft

Modelle mit vielen Variablen mehr

→ es werden "kleinere" Modelle gewählt!

Insgesamt gilt: die oben gezeigten Methoden der Subset-Selection  
filtern erst das Modell und wählen geeignete (optimale)  
Teilmenge aus  $X_1, \dots, X_p$  hinterher auf Basis  
geeigneter Metriken ...



Forciere die Entscheidung während der  
Anpassung der Modellparameter ganz gezielt !

... das führt zum Begriff der sogenannten

SHRINKAGE - METHODEN

### 3.3.2) Shrinkage - Methoden (Schrumpfen des Modells)

211

Überlegungen am linearen Modell

$$Y = b_0 + b_1 X_1 + \dots + b_p X_p + \varepsilon$$

Beobachtung: wird ein Parameter  $b_j = 0$ , dann wird die zugehörige Größe  $X_j$  (effektiv) nicht zur Beschreibung von  $Y$  genutzt!



Wann wir beim Anpassen der Parameter eine Technik benutzen, die erzwingt, dass ohnehin schon kleine  $b_j$  auf Null gesetzt werden, dann haben wir hinterher weniger Variablen im Modell



Stichworte :

Ridge Regression  
Lasso Regression

} lineare  
Modelle  
(KQ-Sch.)

weight decay  
Sparsity

} Neuronale  
Netze

## ① Ridge Regression

Erinnerung : beim KQ-Fit der linearen Regression bestimmen wir

$b_j$  ( $j = 0, \dots, p$ ) so, dass

$$E(b_0, b_1, \dots, b_p) = \sum_{i=1}^n (y_i - (b_0 + b_1 x_{i1} + \dots + b_p x_{ip}))^2 \rightarrow \min_{b_0, \dots, b_p}$$

also

$$E(b_0, \dots, b_p) = \sum_{i=1}^n (y_i - b_0 - \sum_{j=1}^p b_j x_{ij})^2 \rightarrow \min_{b_0, \dots, b_p}$$

KQ -  
LOSS

Modifikation : führe einen Shaffer (sterinkage penalty) in die zu minimierende Fehlerfunktion ein, der es bestraft, wenn viele Variablen im Modell sind.

⇒ modifizierte Fehlerfunktion:

$$\sum_{i=1}^n (y_i - b_0 - \sum_{j=1}^p b_j x_{ij})^2 + \lambda \sum_{j=1}^p b_j^2$$

$E(b_0, \dots, b_p)$

Loss-Funktion

$$+ \lambda \sum_{j=1}^p b_j^2$$

$\lambda \geq 0$   
tuning parameter

← zusätzlicher Term  
(Strafterm)

Def.: Schrinkage Penalty

Der Term

$$\lambda \sum_{j=1}^p b_j^2$$

heißt schrinkage penalty mit tuning  
Parameter  $\lambda \geq 0$

## Interpretation

$$\sum_{i=1}^n \left( y_i - b_0 - \sum_{j=1}^p b_j x_{ij} \right)^2$$

→ Summe von Quadraten  
 $\Rightarrow \geq 0$

$$\lambda \sum_{j=1}^p b_j^2$$

$$\lambda \geq 0$$

$$\text{und } \sum_{j=1}^p b_j^2 \geq 0$$

↓  
 !: wirkt nicht  
 auf  $b_0$   
 $\Rightarrow$  es bleibt mind.  $b_0$   
 im Modell !!

$\Rightarrow$  Shrinkage penalty  
 $\text{ist } \geq 0$

$\Rightarrow$  ① Weil beide Terme im  positiv sind, sind im gesuchten Minimum idealerweise beide Terme minimal

- $E(b_0, \dots, b_p)$  ist dann minimal, wenn  $b_j$  ( $j=0, \dots, p$ ) optimal zu den Daten passen

$\Rightarrow$  optimaler Satz von Schätzungen  $\hat{\beta} = \begin{pmatrix} \hat{b}_0 \\ \vdots \\ \hat{b}_p \end{pmatrix}$

②  $\lambda \cdot \sum_{j=1}^p b_j^2$  ist minimal, wenn möglichst viele der  $b_j$

ganz nah an Null oder sogar exakt gleich Null sind

⇒ der shrinkage penalty stellt eine Randbedingung für die KQ-Schätzung dar (möglichst viele  $b_j$  nahe Null)

③ Bedeutung des tuning parameters  $\lambda \geq 0$ :

→ steuert den Trade-off zwischen den beiden Effekten

- $\lambda = 0$ : "penalty = 0" ⇒ kein Einfluss ( $\hat{\equiv}$  Standard KQ wie vorher)

- $\lambda \rightarrow \infty$ : wachsender Einfluss ⇒ am Ende alle  $b_j$  nahe Null

Bemerkung:

→ das Ergebnis einer KQ-Schätzung ist ein eindeutiger Parametervektor  $\hat{b} = \begin{pmatrix} \hat{b}_0 \\ \vdots \\ \hat{b}_p \end{pmatrix} \rightsquigarrow$  analytische Lösung!

**ABER**

Der Tuning parameter  $\lambda$  ist ein Meta-Parameter der Ridge Regression und aufs erstmal geeignet gewählt

werden (u.U. nicht ganz einfach)

⇒ je nach Wahl von  $\lambda$  erhält man

$$\hat{b}_\lambda = \begin{pmatrix} \hat{b}_{0,\lambda} \\ \vdots \\ \hat{b}_{p,\lambda} \end{pmatrix}$$

Cross Validation für die opl. Wahl!

**FRAGE**

Worin ist das besser als KQ alleine?

Bias-Variance Dilemma:

- $\lambda=0 \Rightarrow$  Varianz hoch, Bias=0 (KQ)

- $\lambda \rightarrow \infty \Rightarrow$  Varianz sinkt schnell, Bias steigt leicht ⇒ **BESSER!!**



INSGES.

## Bemerkung:

217

### Nachteil der Ridge - Regression:

das gefundene Modell ist und bleibt eines mit allen p Inputs  $x_1, \dots, x_p$

 die  $b_j$  werden nur nahe an die Null gedrückt  
sind aber nie exakt Null (außer für  $\lambda = \infty$ )

... das ist vielleicht kein Problem für die Performance des finalen Modells (das ist ja optimiert), aber die Interpretierbarkeit leidet!

ALTERNATIVE

Lasso  $\rightsquigarrow$  ändere den Penalty!

② Lasso - Shrinkage

betrachte:

$$\sum_{i=1}^n (y_i - b_0 - \sum_{j=1}^p b_j x_{ij})^2 + \lambda \sum_{j=1}^p |b_j|$$

$\ell_1$ -Norm von  $\vec{b}$

KQ - Loss

Lasso-penalty

Beobachtung: indem man eine andere Norm benutzt erleichtert

$b_j = 0$  !

$\leftarrow$  warum ist das so ???

Überlegung: es lässt sich zeigen, dass das Minimierungsproblem sich durch in der Form

$$\min_b \left( \sum_{i=1}^n (y_i - b_0 - \sum_{j=1}^p b_j x_{ij})^2 \right)$$

unter der Bedingung

$$\sum_{j=1}^p |b_j| \leq s$$

Lasso  
 $\ell_{1-N}$

$$\min_b \left( \sum_{i=1}^n (y_i - b_0 - \sum_{j=1}^p b_j x_{ij})^2 \right)$$

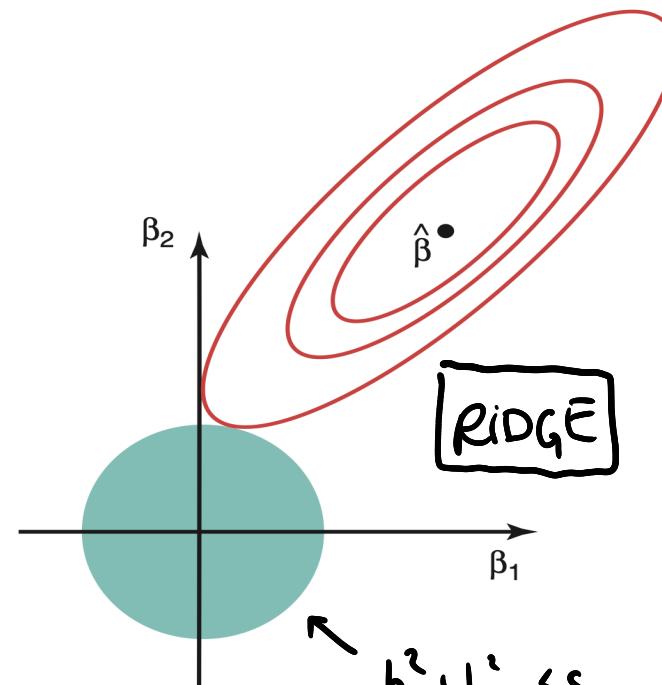
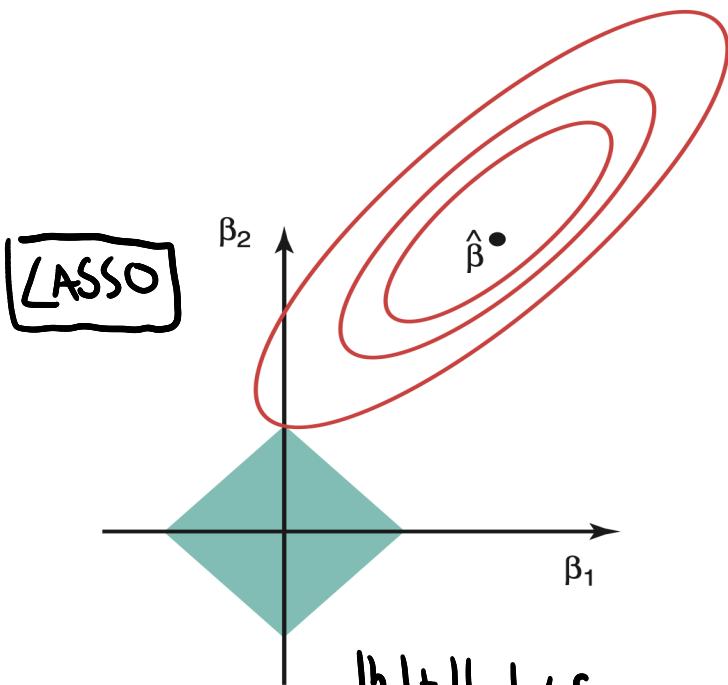
unter der Bedingung

$$\sum_{j=1}^p b_j^2 \leq s$$

Ridge  
 $\ell_{2-N}$   
 $s \in \mathbb{R}$

schreiben lässt.

Wie sehen jeweils die Randbedingungen aus?



Experiment für 2 Features

KQ-Lösung heißt hier  $\hat{\beta}$ , die blauen Flächen sind die Randbedingungen, die roten Linien sind die Isolinien des RSS

⇒ Schätzung liegt da, wo die roten Linien die blauen Flächen zuerst treffen

⇒ beim Kreis höchst unwahrscheinlich auf einer Achse, beim Rechteck höchst wahrscheinlich auf einer Achse  $\Rightarrow b_1$  oder  $b_2 = 0$ !



Und was ist am Ende insgesamt  
besser (in Bezug auf die Performance des  
finalen Modells) ??

Antwort: das hängt davon ab

①  $x_1, \dots, x_p$  haben tatsächlich alle einen  
(wenn auch winzigen) Bezug zu  $Y$

→ Ridge Regression liefert das  
bessere Modell

② nur ein paar der  $x_1, \dots, x_p$   
sind wirklich relevant für  $Y$

⇒ Lasso liefert das  
bessere Modell

**FAZIT** Wenn möglich sollte man beides versuchen ☺