

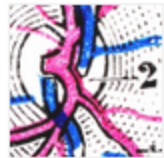


Lineare Regression: Gute Vorhersagen

Ziele: Vorhersage vs. Inferenz

- **Inferenz**: Hintergründe verstehen
 - Durchschnittliche Kassierzeit pro Produkt ?
 - Gibt es einen Zshg zw Dosis und Heilungswa. ?
 - Möglichst **einfaches Modell**, um interpretieren zu können
- **Vorhersage**: Hintergründe egal (Black box)
 - Gegeben die Blutwerte: Verträgt der Patient das Medikament ?
 - Wie viele Kunden wollen neues Produkt ?
 - Möglichst **komplexes Modell**, um alle Details zu erfassen
- **Lineare Regression**: Guter Kompromiss
 - werden uns weiter damit beschäftigen
- Fokus heute: **Vorhersage**

Reale Probleme: Kaggle



Diabetic Retinopathy Detection

Identify signs of diabetic retinopathy in eye images

\$100,000

661

15 days ago



Heritage Health Prize

Identify patients who will be admitted to a hospital within the next year using historical claims data. (Enter by 06:59:59 UTC Oct 4 2012)

\$500,000

1353

2 years ago



West Nile Virus Prediction

Predict West Nile virus in mosquitos across the city of Chicago

\$40,000

1306

55 days ago



Restaurant Revenue Prediction

Predict annual restaurant sales based on objective measurements

\$30,000

2257

3 months ago



Problem mit einfachem Setting

Completed • \$30,000 • 2,257 teams

Restaurant Revenue Prediction

Mon 23 Mar 2015 – Mon 4 May 2015 (3 months ago)

Competition Details » [Get the Data](#) » [Make a submission](#)

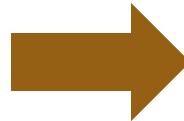
Predict annual restaurant sales based on objective measurements



Lineare Regression für Vorhersage

Data fields

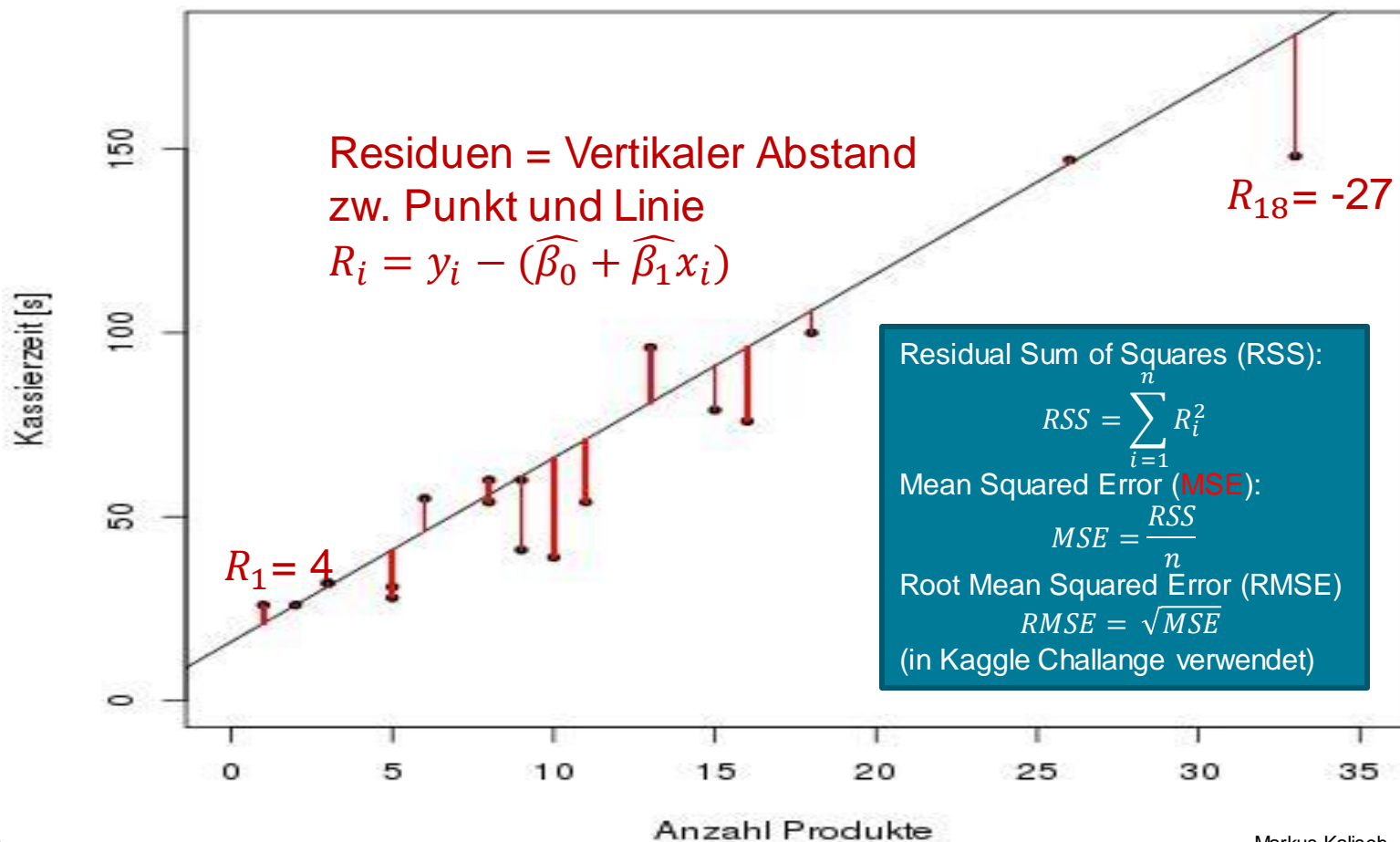
- **Id** : Restaurant id.
- **Open Date** : opening date for a restaurant
- **City** : City that the restaurant is in. Note that there are unicode in the names.
- **City Group**: Type of the city. Big cities, or Other.
- **Type**: Type of the restaurant. FC: Food Court, IL: Inline, DT: Drive Thru, MB: Mobile
- **P1, P2 - P37**: There are three categories of these obfuscated data. **Demographic data** are gathered from third party providers with GIS systems. These include population in any given area, age and gender distribution, development scales. **Real estate data** mainly relate to the m2 of the location, front facade of the location, car park availability. **Commercial data** mainly include the existence of points of interest including schools, banks, other QSR operators.



Revenue: The revenue column indicates a (transformed) revenue of the restaurant in a given year and is the target of predictive analysis. Please note that the values are transformed so they don't mean real dollar values.

Wdh: Güte vom Modell

Streudiagramm

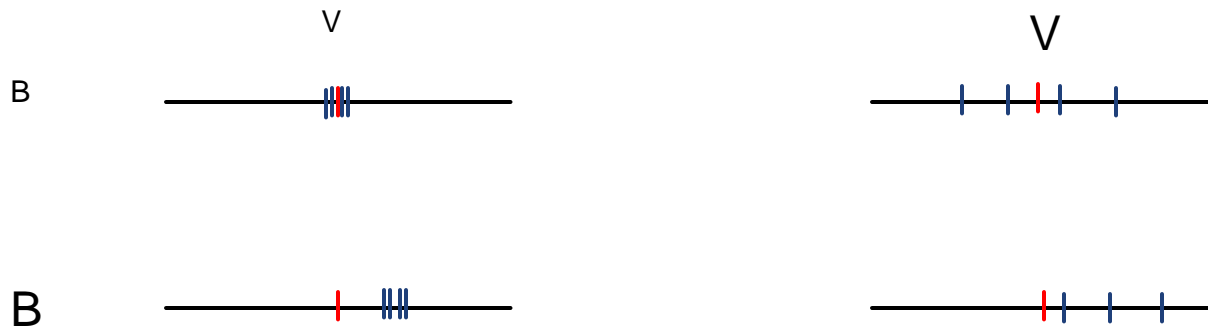


Training MSE vs. Test MSE

- Training Daten: Bisher gesehene Daten
Test Daten: Neue, zukünftige Daten
- Training MSE: Fehler auf bisher gesehenen Daten
Test MSE: Fehler auf zukünftigen Daten
- Bisher: “Gesehenes gut erklären”
 - Modellklasse wählen (z.B. Geradengleichung)
 - Parameter finden, so dass **Trainings MSE minimal**
- Neues Ziel: “Zukünftige Daten gut erklären”
Modell finden, so dass **Test MSE minimal**

Wdh: Bias und Varianz eines Schätzers

- Schätzer $\hat{\Theta}$, wahrer Parameter Θ
- Je nach beobachteten, zufälligen Daten: $\hat{\Theta}$ variiert
- Bias (B) von $\hat{\Theta}$: $E(\hat{\Theta} - \Theta)$
- Varianz (V) von $\hat{\Theta}$: $\text{Var}(\hat{\Theta}) = E\left((\hat{\Theta} - \Theta)^2\right)$



Bias-Variance Trade-Off

- **Training MSE**: Kann beliebig klein gemacht werden, wenn wir nur genügend Parameter verwenden
- **Test MSE**: Selbst wenn wir $f(x)$ perfekt schätzen, stört der Fehlerterm ε unsere Vorhersage
- (Eq. 2.7): Erwartetes Residuenquadrat an Stelle x_0 im Testset:

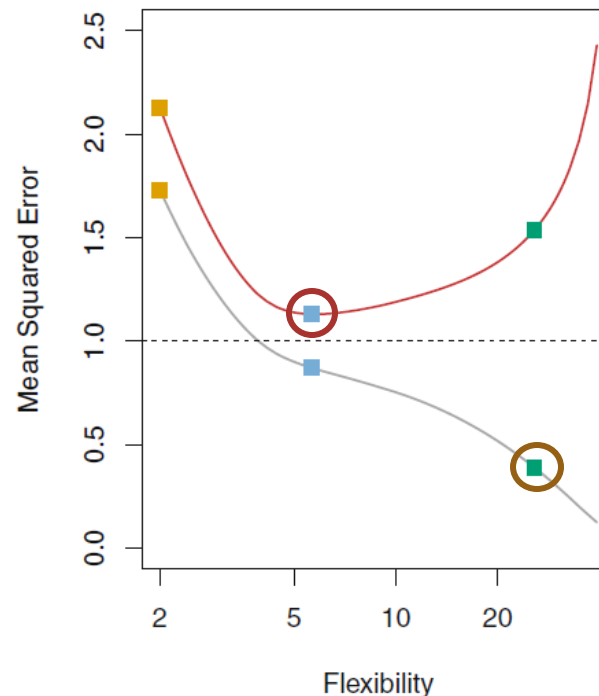
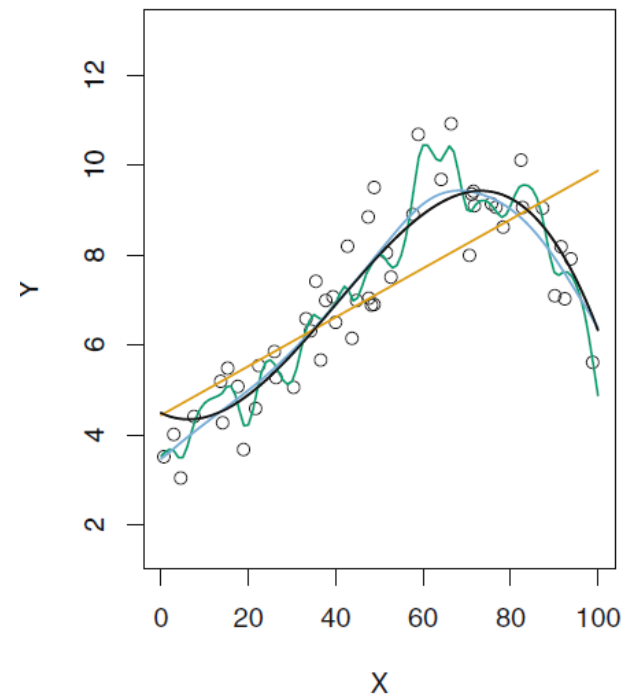
$$E \left(y_0 - \hat{f}(x_0) \right)^2 = \text{Var} \left(\hat{f}(x_0) \right) + \left[\text{Bias} \left(\hat{f}(x_0) \right) \right]^2 + \text{Var}(\varepsilon)$$

Test MSE: Mittelwert von $E \left(y_0 - \hat{f}(x_0) \right)^2$ über alle möglichen Werte von x_0 im Testset.

→ perfekter Fit von $f(x)$: $\text{Test MSE} = \text{Var}(\varepsilon) > 0$
- **Fazit: Training MSE \neq Test MSE**

Paradox: “Overfitting”

- Paradox: Modell das die bisherigen Daten am besten beschreibt (**minimaler Training MSE**) ist nicht unbedingt das beste für zukünftige Daten (**minimaler Test MSE**)!



$Y = f(x) + \varepsilon$
 $f(x)$ meist “glatt”

Perfekter Fit auf Trainings-Daten modelliert v.a. Fehlerterm, der bei zukünftigen Daten anders sein wird.

Fazit

- Um gute Vorhersagen zu machen, müssen wir den Test MSE minimieren
- Um den Test MSE zu minimieren reicht es **NICHT** den Training MSE zu minimieren

Wie schätzt man den **Test MSE** ?

- Direkte Methoden
 - Test Datensatz
 - Cross-validation (CV)
- Indirekte Methoden: C_p , AIC , **BIC** , *Adjusted R^2*
 - Korrektur vom Training MSE
 - Approximation von direkter Methode:
Je mehr Beobachtungen, desto genauer die Approx.
 - Schnell: Gut, wenn viele oder aufwändige Modelle zu schätzen sind



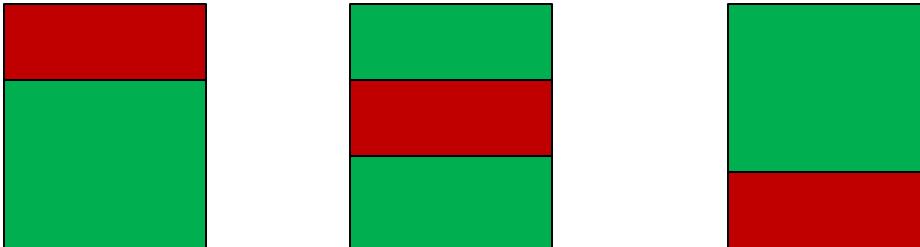
Direkte Methode 1: Expliziter Test Datensatz

- Teile Daten in Test- und Trainingsdatensatz



- Schätze Modell auf Trainingsdatensatz, evaluiere auf Testdatensatz
- Vorteil: Einfach, schnell
- Nachteil:
 - Je nach Wahl vom Testdatensatz: Unterschiedlicher MSE
 - Trainingsdatensatz kleiner als Originaldatensatz → Test MSE wird überschätzt

Direkte Methode 2: Cross-Validation (CV)



- Leave-one-out cross-validation (**LOOCV**):
Jede Zeile ist einmal Testset; Rest ist Trainingsset
Nachteil: Langsam, weil ein Fit pro Zeile
- **K-fold cross-validation**, z.B. 10-fold:
Teile Daten in 10 Blöcke; jeder Block ist einmal Testset
Nachteil: Je nach Unterteilung in K Blöcke unterschiedliches Test MSE



Cross-Validation in R

- Grundsätzlich funktioniert CV für alle erdenklichen Vorhersagemethoden
- Für Lineare Modelle (und sogar GLM's) gibt es eine besonders einfache Funktion: `cv.glm()` in package 'boot'
- Lineare Modelle sind eine Unterklasse der Generalized sog. Linear Models (GLMs) (siehe später)
- Damit kann man sowohl LOOCV als auch k-Fold CV durchführen

Indirekte Methoden: Hintergrund

- Kriterium K , das Güte vom Fit (RSS) und Anzahl Parameter (d) verbindet: $K = RSS + f(d)$
- Theorie: Unter gewissen Annahmen und bei sehr grossen Datensätzen (asymptotisch) gilt:
Modell mit bestem K ist optimal für Vorhersage
- Vorteil: Schnell zu rechnen
- Nachteil: Approximativ; macht Annahmen; Test MSE nicht berechnet
- Praxis: Verwenden, falls viele oder komplizierte Modelle geschätzt werden müssen

Bayesian Information Criterion (BIC)

- Je nach theo. Annahmen, leicht andere Form von K
(d : Anz. Parameter im Modell, n : Anz. Beobachtungen)
- Beliebte Wahl: $BIC = \frac{1}{n} (RSS + \log(n)d\hat{\sigma}^2)$
- Es gibt viele alternative Kriterien: C_p , AIC , Adjusted R^2 , ...
(s. ISLR Kap. 6.3)
- Praxis: Willkürlich für eine Methode entscheiden
(z.B. BIC)

Modellwahl

- Fokus 1: Finde Modell, das möglichst kleinen Vorhersagefehler hat
- Fokus 2: Finde Variablen, die für gute Vorhersage nötig sind
- Könnten CV verwenden (s. ISL 6.5.3); wir konzentrieren uns aber auf indirekte Methoden: Einfacher und schneller
- Faustregel für die Praxis:
 - Test MSE mit CV (direkte Methode)
 - Modellwahl mit BIC (indirekte Methode)

Techniken zur Modellwahl 1: Exakt

- Berechne eine Lineare Regression für alle möglichen Kombinationen von erklärenden Variablen; speichere BIC
- Vorteil: Findet bestes Subset bzgl. BIC
- Nachteil: Rechenaufwand !
 p Variablen $\rightarrow 2^p$ Subsets

p	2^p
10	10^3
20	10^6
30	10^9
40	10^{12}
...	...

In der Praxis kaum
mehr zu berechnen

Techniken zur Modellwahl 2: Heuristiken

- Wie Bergwanderung im Nebel: Gehe immer nach oben
→ man landet evtl. auf Zwischengipfel
- Verfehlen evtl. globales Optimum
- “Stepwise forward”: Starte mit dem leeren Modell; füge immer eine Variable hinzu
- “Stepwise backward”: Start mit dem vollen Modell; lasse immer eine Variable weg

Bsp: Stepwise forward selection

- Variablen: Y, X_1, X_2, X_3
 $M_1: Y \sim 1 \rightarrow BIC = 20$
- Bestes Modell mit einer Variable:
 $M_2: Y \sim X_2 \rightarrow BIC = 17$, also besser
- Bestes Modell mit X_2 und noch einer Variable:
 $M_3: Y \sim X_2 + X_1 \rightarrow BIC = 18$,
also schlechter als $M_2 \rightarrow$ Stop
- Ausgabe: Bestes Modell ist $Y \sim X_2$.

Bsp: Stepwise backward selection

- Variablen: Y, X_1, X_2, X_3
 $M_1: Y \sim X_1 + X_2 + X_3 \rightarrow BIC = 20$
- Bestes Modell mit einer Variable weniger:
 $M_2: Y \sim X_2 + X_3 \rightarrow BIC = 17$, also besser als M_1
- Bestes Modell mit einer Variable weniger als X_2, X_3 :
 $M_3: Y \sim X_2 \rightarrow BIC = 18$, also schlechter als $M_2 \rightarrow \text{Stop}$
- Ausgabe: Bestes Modell ist $Y \sim X_2 + X_3$.



Modellwahl in R

- Funktion “**regsubsets**” in Paket “**leaps**”;
berechnet sowohl exakt als auch mit Heuristiken
- Definition von BIC in “leaps”:

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

→ finde Modell mit **minimalem** BIC

- Vorgehen:

1) Für jede Anzahl erklärende Variablen: Finde bestes Subset bzgl. RSS

```
m1 <- regsubsets(revenue ~ ., data = dat, method = "exhaustive", nvmax = 19)
m1s <- summary(m1)
```

2) Vergleiche Modelle mit unterschiedlichen Variablenzahlen mit BIC

```
> which.min(m1s$bic)
```


```
[1] 2
```

```
> coef(m1, 2)
```

(Intercept)	P6	P8
5507746.1	500823.2	-530886.6

Kaggle Ergebnis



- Unsere Schätzungen von RMSE: Ungenau, weil “nur” gut 100 Beobachtungen
- Test-Datensatz hat 100.000 Beobachtungen: RMSE sehr genau bestimmt.
- Ergebnis:  Submitted an entry to [Restaurant Revenue Prediction](#), obtaining 1916571.42502
RMSE: 1.917 Mio
- Damit wären wir ca. im **Mittelfeld** aller submissions gelandet (s. Appendix)
- Der **Sieger** ist bzgl. RMSE **knapp 8% besser**
- Normalerweise muss man mehrere Mann-Wochen Arbeit reinstecken um diese paar Prozent Verbesserung zu erhalten

Fazit

- Pareto-Prinzip: “80% Nutzen mit 20% Aufwand”
- Haben ein sehr einfaches Modell verwendet (wenige Zeilen Code) und wichtige Variablen weggelassen (Datum, Ort, Typ)
- Trotzdem ist der Sieger nur ein paar Prozent besser

Appendix: Kaggle Ergebnis

- Da dieses Projekt schon abgeschlossen ist, kommen wir nicht mehr in das Leaderboard
- Im Leaderboard erscheint “Sample Submission Benchmark” mit $RMSE = 1883099.54122$ (Rang 1246/2257)
- Wenn man genau diese “Sample Submission” aber manuell nochmal ausführt, erscheint
 $RMSE = 1929245.11374$, also ein anderer Wert; vermutlich wurde bei Abschluss des Wettbewerbs der Testdatensatz leicht geändert
- Wann man unser bestes Modell submitted erhält man
 $RMSE = 1916571.42502$, also ca. 0.67% besser als die “Sample Submission”
- Daher hätten wir mit dem ursprünglichen Testdatensatz wohl etwa
 $RMSE = 1883099.5 * 0.9933 = 1870729$ erhalten
- Das wäre Rang 1103 / 2257, also Mittelfeld
- Der Sieger hat $RMSE = 1727811.48554$, also knapp 8% besser