Computational Methods for Statistics (VU) (706.026)

Elisabeth Lex

ISDS, TU Graz

Jan 24, 2023

Outline

Readings

Fahrmeir et. al, Statistik: der Weg zur Datenanalyse (Chapter 12)

Wasserman, All of Statistics (Chapter 13)

Chernick, Bootstrap Methods: A Guide for Practitioners and Researchers (Chapter 6)

Regression

- With regression we study the relationship between a **response variable** Y and a **feature** X
- Y also called *dependent* variable
- X also called *independent* variable, *predictor* variable or *covariate*
- The relationship can be summarized with the **regression function**:

$$r(x) = E(Y|X = x) = \int y f(y|x) dy$$

- The goal is to estimate $r(x)$ from a random sample of the form:

$$(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n) \sim F_{X,Y}$$

Linear Regression

- The simplest version of regression is **linear regression**
- $r(x)$ is assumed to be linear:

$$r(x) = \beta_0 + \beta_1 x$$

- This is the **simple linear regression model**
- Simplifying assumption: $Var(\epsilon_i|X = x) = \sigma^2$ does not depend on x

Definition 11 (The Simple Linear Regression Model)

The simple linear regression model is given by:

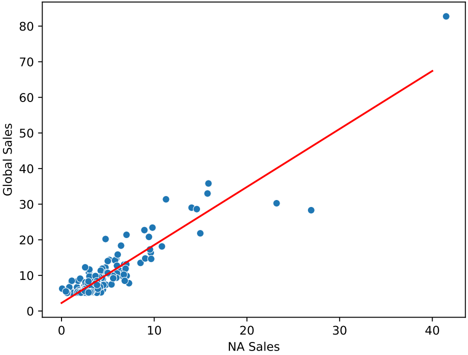
$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i,$$

where $E(\epsilon_i|X_i) = 0$ and $Var(\epsilon_i|X_i) = \sigma^2$.

Linear Regression

Example 26 (Linear relationship in game sales)

Figure shows a plot of global sales of top 160 video games Y versus sales of these games in North America X. The red line is an estimated linear regression line.



Fitted Line

- The unknown parameters are:
  1. **Intercept** β_0
  2. **Slope** β_1
- Let $\hat{\beta}_0$ and $\hat{\beta}_1$ denote estimates for β_0 and β_1
- The **fitted** line is:

$$\hat{r}(x) = \hat{\beta}_0 + \hat{\beta}_1 x$$

Sum of Squared Errors

- The **predicted values** or **fitted values** are $\hat{Y}_i = \hat{r}(X_i)$
- **Residuals** or **errors** are defined as:

$$\hat{\epsilon}_i = Y_i - \hat{Y}_i = Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)$$

- The **residual sums of squares** (RSS) or **sum of squared errors** (SSE) measures how well the line fits the data:

$$SSE = \sum_i \hat{\epsilon}_i^2 = \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right]^2$$

Least Square Estimates

Definition 12 (Least Square Estimation for Simple Linear Regression)

The *least square estimates* are the values of $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize the sum of squared errors:

$$SSE = \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i \right) \right]^2$$

- How can we compute $\hat{\beta}_0$ and $\hat{\beta}_1$?

Least Square Estimates

Definition 12 (Least Square Estimation for Simple Linear Regression)

The *least square estimates* are the values of $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize the sum of squared errors:

$$SSE = \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i \right) \right]^2$$

- How can we compute $\hat{\beta}_0$ and $\hat{\beta}_1$?
- By taking partial derivatives of the total error with respect to $\hat{\beta}_0$ and $\hat{\beta}_1$ and setting them to zero

Least Square Estimates

- Partial derivatives:

$$
\begin{aligned}
J &= \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right]^2 \\
\frac{\partial J}{\partial \hat{\beta}_0} &= -2 \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right] \\
\frac{\partial J}{\partial \hat{\beta}_1} &= -2 \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right] X_i
\end{aligned}
$$

- We now solve the equation system:

$$
\begin{aligned}
-2 \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right] &= 0 \\
-2 \sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right] X_i &= 0
\end{aligned}
$$

Least Square Estimates

- Solving for $\hat{\beta}_0$:

$$\begin{aligned} \cancel{-2}\sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right] &= 0 \\ \cancel{n}\overline{Y}_n - \cancel{n}\hat{\beta}_0 - \cancel{n}\hat{\beta}_1 \overline{X}_n &= 0 \end{aligned}$$

- We arrive at:

$$\hat{\beta}_0 = \overline{Y}_n - \hat{\beta}_1 \overline{X}_n$$

Least Square Estimates

- Solving for $\hat{\beta}_1$:

$$\begin{aligned}
\cancel{-2}\sum_i \left[Y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 X_i\right)\right] X_i &= 0 \\
\sum_i \left(X_i Y_i - \hat{\beta}_0 X_i - \hat{\beta}_1 X_i^2\right) &= 0
\end{aligned}$$

- We substitute $\overline{Y}_n - \hat{\beta}_1 \overline{X}_n$ for $\hat{\beta}_0$:

$$\begin{aligned}
\sum_i \left(X_i Y_i - X_i \overline{Y}_n + \hat{\beta}_1 X_i \overline{X}_n - \hat{\beta}_1 X_i^2\right) &= 0 \\
\sum_i \left[X_i Y_i - X_i \overline{Y}_n - \hat{\beta}_1 (X_i^2 - X_i \overline{X}_n)\right] &= 0 \\
\sum_i (X_i Y_i - X_i \overline{Y}_n) - \hat{\beta}_1 \sum_i (X_i^2 - X_i \overline{X}_n) &= 0
\end{aligned}$$

Least Square Estimates

- This gives for $\hat{\beta}_1$:

$$\hat{\beta}_1 = \frac{\sum_i (X_i Y_i - X_i \overline{Y}_n)}{\sum_i (X_i^2 - X_i \overline{X}_n)}$$

- Please note that:

$$\begin{aligned} \sum_i (\overline{X}_n^2 - X_i \overline{X}_n) &= 0 \\ \sum_i (\overline{X}_n \overline{Y}_n - \overline{X}_n Y_i) &= 0 \end{aligned}$$

Least Square Estimates

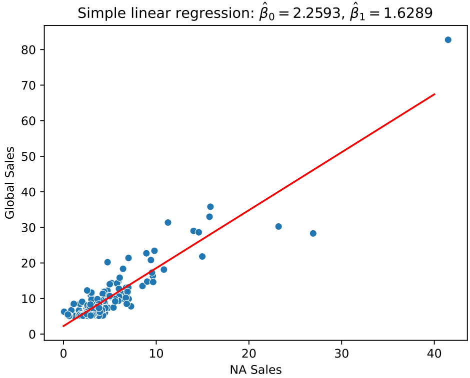
- We add these two zeros to the expression for $\hat{\beta}_1$:

$$\begin{aligned}
\hat{\beta}_1 &= \frac{\sum_i (X_i Y_i - X_i \overline{Y}_n + \overline{X}_n \overline{Y}_n - \overline{X}_n Y_i)}{\sum_i (X_i^2 - X_i \overline{X}_n + \overline{X}_n^2 - X_i \overline{X}_n)} \\
&= \frac{\sum_i \left[X_i (Y_i - \overline{Y}_n) - \overline{X}_n (Y_i - \overline{Y}_n) \right]}{\sum_i \left(X_i - \overline{X}_n \right)^2} \\
&= \frac{\sum_i (X_i - \overline{X}_n)(Y_i - \overline{Y}_n)}{\sum_i \left(X_i - \overline{X}_n \right)^2} \\
&= \frac{Cov(X, Y)}{Var(X)}
\end{aligned}$$

Least Square Estimates

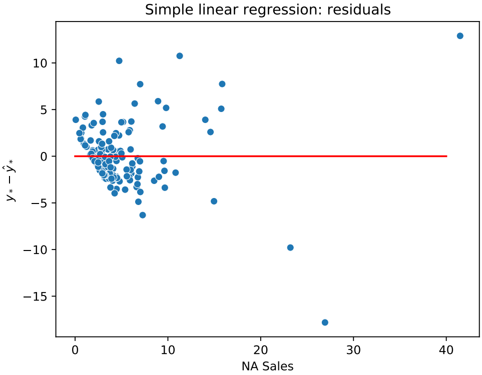
Example 26 (Linear relationship in game sales)

Figure shows a plot of global sales of top 160 video games Y versus sales of these games in North America X. The red line is an estimated linear regression line with $\hat{\beta}_0 = 2.2593$ and $\hat{\beta}_1 = 1.6289$.



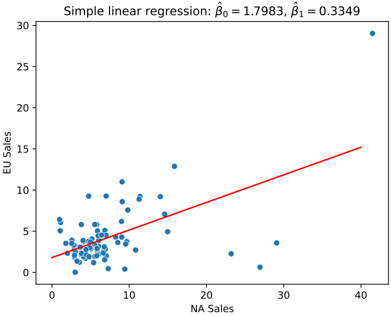
Residuals

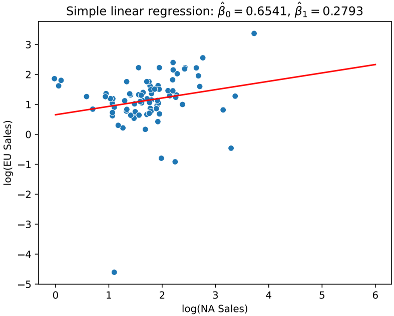
- Linear regression is the most accurate when the residuals behave like random normal numbers



Transforming Data

- Often, better results can be achieved by transforming the data, e.g. $log(X)$, $log(Y)$
- Here we regress Europe sales on NA sales





MLE under Normality Assumption

- Suppose we add assumption that $\epsilon_i | X_i \sim N(0, \sigma^2)$
- We have then: $Y_i | X_i \; N(\mu_i, \sigma^2)$, where $\mu_i = \beta_0 + \beta_1 X_i$
- The likelihood function is:

$$
\begin{aligned}
L(\beta_0, \beta_1, \sigma) &= \prod_{i=1}^{n} f(X_i, Y_i) = \prod_{i=1}^{n} f_X(X_i) f_{Y|X}(Y_i | X_i) \\
&= \prod_{i=1}^{n} f_X(X_i) \times \prod_{i=1}^{n} f_{Y|X}(Y_i | X_i)
\end{aligned}
$$

- The first term does not depend on β_0 and β_1 and we may omit it

MLE under Normality Assumption

- The second term is then proportional to:

$$L_2(\beta_0, \beta_1, \sigma) \propto \frac{1}{\sigma^n} e^{-\frac{1}{2\sigma^2}\sum_i (Y_i - \mu_i)^2}$$

- The log-likelihood is then given by:

$$\mathcal{L}(\beta_0, \beta_1, \sigma) = -nlog\sigma - \frac{1}{2\sigma^2}\sum_i (Y_i - (\beta_0 + \beta_1 X_i))^2$$

- Maximizing log-likelihood is then equivalent to minimizing the second term, i.e. minimizing SSE
- Under the assumption of normality MLE is the same as the least squares estimator

Prediction

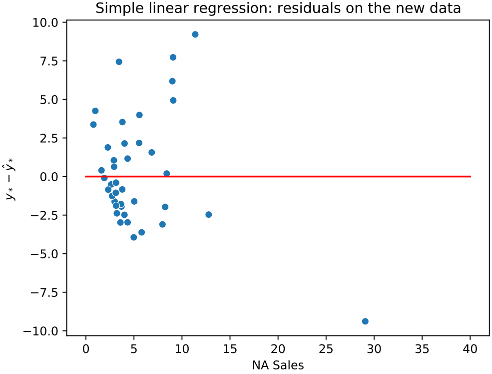
- We start with our data as before: $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$
- We estimate $\hat{r}(x) = \hat{\beta}_0 + \hat{\beta}_1 x$ from that data
- We observe a new value $X = x_*$
- Then an estimate or a prediction of Y_* is given by:

$$\hat{Y}_* = \hat{\beta}_0 + \hat{\beta}_1 x_*$$

Prediction with the Unseen Data

Example 26 (Linear relationship in game sales)

We estimate $\hat{\beta}_0 = 2.2593$ and $\hat{\beta}_1 = 1.6289$ from the 160 sold video games globally versus North America. With these estimates we predict global sales from the North America sales for 40 new games.



Vectors as Features

- Now suppose that our feature is a vector of length k
- We have data as before: $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$, where

$$X_i = (X_{i1}, \ldots, X_{ik})$$

- The linear regression model is now:

$$Y_i = \sum_{j=1}^{k} \beta_j X_{ij} + \epsilon_i, \text{ where } E(\epsilon_i | X_i) = 0$$

- To include intercept in the model we can set $X_{i1} = 1$
- Then each X_i is a vector of length $k + 1$

Model in Matrix Notation

- Typically we express models using **matrices**:

$$
\begin{aligned}
y &= \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} &
X &= \begin{pmatrix} X_{11} & X_{12} & \dots & X_{1k} \\ X_{21} & X_{22} & \dots & X_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n1} & X_{n2} & \dots & X_{nk} \end{pmatrix} \\
\beta &= \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix} &
\epsilon &= \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}
\end{aligned}
$$

- Then we can write:

$$y = X\beta + \epsilon$$

Least Squares for Multiple Regression

- Let $\hat{\beta}$ denote the estimate for β
- The predicted values are:

$$\hat{y} = X\hat{\beta}$$

- The errors are given by:

$$\hat{\epsilon} = y - \hat{y} = y - X\hat{\beta}$$

- The total sum of squared errors is given by the vector norm:

$$SSE = ||y - X\hat{\beta}||_2^2$$

Least Squares Estimates for Multiple Regression

- By taking all the partial derivatives and collecting all the terms in a matrix notation:

$$
\begin{aligned}
J &= ||y - X\hat{\beta}||_2^2 \\
\frac{\partial J}{\partial \hat{\beta}} &= -2X^T(y - X\hat{\beta})
\end{aligned}
$$

- We set then the partial derivatives to zero:

$$
\begin{aligned}
\cancel{-2}X^T(y - X\hat{\beta}) &= 0 \\
X^T X\hat{\beta} &= X^T y \\
\hat{\beta} &= (X^T X)^{-1} X^T y
\end{aligned}
$$

- We arrive at **normal equations**

Video Games Ratings

Example 27 (User Rating for Video Games)

We have a video games dataset containing games release dates, price, sales, average and median playtime from Steam, critics rating, and the user rating from from Metacritic reviewing site. We are interested in relationship between user rating and other features. We fit the model and obtain the following results:

Feature	$\hat{\beta}_j$	conf. intervals
Intercept	7.04948665	(6.999040, 7.102739)
Release Date	0.09082842	(0.032808, 0.145047)
Price	-0.1283695	(-0.197546, -0.057582)
Sales	0.03249976	(-0.015483, 0.085458)
Avg. Playtime	0.02901469	(-0.041831, 0.111011)
Md. Playtime	0.01683537	(-0.077188, 0.071164)
Critics Score	0.70902049	(0.647059, 0.775075)

Relations between Variables

- This example raises the question about the degree of relation
- We could formulate the question as a hypothesis test:

$$H_0 : \beta_j = 0 \text{ vs. } H_1 : \beta_j \neq 0$$

- This question has some important scientific and practical consequences:
  1. We learn something about relationships between variables
  2. Should we eliminate some variables for efficiency and a better prediction
  3. Correlation vs. causation

Underfitting vs. Overfitting

- The previous example illustrates a typical problem in multiple regression
- We have many features but do we want to include them all in the model?
- A smaller model with less features has two advantages:
  1. The prediction may be better
  2. Better understanding of the problem
- Generally, more features leads to less bias but a higher variance
- Too few features: **underfitting** results in high bias
- Too many features: **overfitting** results in high variance
- Good predictions result from achieving a good balance between bias and variance

Model Selection

- We apply **model selection** to achieve a good balance between bias and variance
- Model selection consists of two steps:
  1. Assign a *score* to each model, which measures how good model is
  2. Search trough (all) models to find the model with the best score
- It is a bad idea to score the model on the **training** data
- We will always underestimate the prediction error because we are using the data twice: to fit the model and to estimate the error
- A better estimation of the prediction error is on the **test** data
- Popular choices for scoring are SSE or R^2:

$$R^2 = 1 - \frac{\sum_i (Y_i - \hat{Y}_i)^2}{\sum_i (Y_i - \overline{Y}_n)^2}$$

Cross-Validation

- Cross-validation is a computational method for (among others) estimating prediction error
- A typical approach is **k-fold cross-validation**:
  1. We divide the data into k groups, e.g. $k = 10$
  2. We omit one group and fit the model on the remaining groups
  3. We use the fitted model to predict the data from the omitted group
  4. We estimate the error by e.g. SSE on the omitted group
  5. We repeat all the steps for all k groups and average the individual SSEs

Model Search

- Now that we know how to score the models we need a search strategy
- If there are k features how many possible models do we have?

Model Search

- Now that we know how to score the models we need a search strategy
- If there are k features how many possible models do we have?
- 2^k
- If k is not too large we can perform an exhaustive search
- Otherwise we need a heuristic and search only over a subset of all models

Greedy Model Search

- Two common methods are **forward and backward stepwise regression**
- In forward regression we start with no feature in the model
- We then greedily add the feature that gives the best score
- We continue adding one feature at a time until the score does not improve
- In backward regression we start with the full model and remove one feature at a time until we can not improve the score any more
- Both methods can not guarantee to reach the model with the globally best score
- In practice, they work quite well

Video Games Ratings: Model Search

Example 28 (User Rating for Video Games)

We have a video games dataset containing games release dates, price, sales, average and median playtime from Steam, critics rating, and the user rating from from Metacritic reviewing site. We are interested in relationship between user rating and other features. We fit the model and obtain the following results:

Notebook 15 (Video Games Ratings)

`model_selection.ipynb`

Video Games Ratings: Result Analysis

Example 29 (User Rating for Video Games)

We have a video games dataset containing games release dates, price, sales, average and median playtime from Steam, critics rating, and the user rating from from Metacritic reviewing site. We are interested in relationship between user rating and other features. We fit the model and obtain the following results:

Notebook 16 (Video Games Ratings)

`games_regression.ipynb`

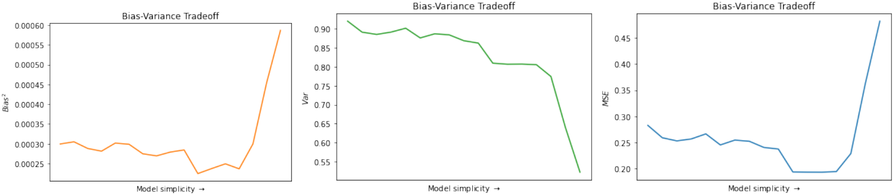
Bias-Variance Trade-off

- Too few features (simple model): **underfitting** $\rightarrow$ high bias
- Too many features (complex model): **overfitting** $\rightarrow$ high variance
- With overfitting prediction accuracy drops on new data

Example 30 (Player Skills in StarCraft)

Using measures of cognitive-motor, attentional, and perceptual processing extracted from game data from 3360 Real-Time Strategy players (StarCraft 2 players) at different levels of expertise, identify variables most relevant to expertise.

Bias-Variance Trade-off



Notebook 17 (Video Games Ratings)

`ridge.ipynb`

Approaching Bias-Variance Trade-off with Regularization

- OLS is unbiased w.r.t. feature coefficients
- Each feature is equally important as others
- However, in practice some features are more important than others
- We would like to give more weight to important features
- Also, less weight to less important features
- We can achieve this by shrinking the less important coefficients

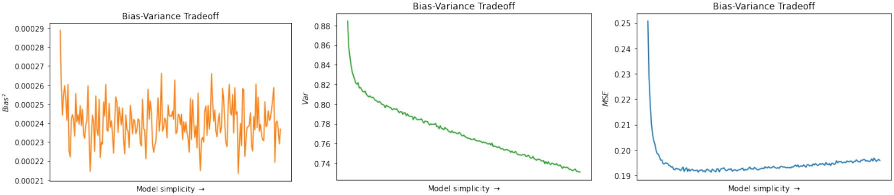
Approaching Bias-Variance Trade-off with Regularization

- We regularize the SSE and add a coefficient shrinking term:

$$J(\hat{\beta}) = ||y - X\hat{\beta}||_2^2 + \lambda||\hat{\beta}||_2^2$$

- The contribution of a less important feature to SSE term is low
- Its contribution to the regularization term should be also low
- Hence, we move its coefficient towards zero
- The parameter λ: SSE vs. regularization
- Higher λ more coefficients moving towards zero

Approaching Bias-Variance Trade-off with Regularization



Notebook 18 (Video Games Ratings)

`ridge.ipynb`