SF2930 - Regression analysis

KTH Royal Institute of Technology, Stockholm

Lecture 9 – Shrinkage methods (MPV 9.5.3, HTF 3.4, Iz 5.4)

February 14, 2022

Todays lecture

- Ridge regression
- Model validation techniques
  - Cross validation
  - (Unconditional bootstrap)
  - (Conditional bootstrap)

Assumptions

In this lecture, we assume that the vectors of observations of each regressor is centered and normalized, and that the vector of responses is also centered and normalized.
In other words, we assume that the data has been transformed so that

$$\bar{\mathbf{y}} = \overline{X_{\cdot 1}} = \ldots = \overline{X_{\cdot (k+1)}} = 0$$

and

$$\|\mathbf{y}\|_2^2 = \|X_{\cdot 1}\|_2^2 = \ldots = \|X_{\cdot (k+1)}\|_2^2 = 1.$$

Motivation for ridge regression

Problem

If $X^T X$ is ill-conditioned (if there is multicollinearity), then the variance of $\hat{\beta}$ becomes very large. Consequently, even though the model might have a good fit, it will probably not be very good for predictions.
PCR solves this by changing the regressors, but what if $\hat{\beta}$ is the problem, rather than the regressors...? However, by the Gauss Markov theorem, $\hat{\beta}$ is the unbiased estimator with smallest variance..

Idea

Find a biased estimator that performs better when there is multicollinearity.

What does it mean to be a "good" estimator

In general, we want estimators to

- be unbiased
- have small variance

By the Gauss-Markov theorem, we know that the LSE $\hat{\boldsymbol{\beta}}$ is the unbiased linear estimator with smallest variance, but since the variance is large when there is collinearity, we might be happy enough if there is approximate unbiasedness.

The mean square error

$$MSE(\hat{\boldsymbol{\beta}}^*) := \mathbb{E}\left[\|\hat{\boldsymbol{\beta}}^* - \boldsymbol{\beta}\|_2^2\right] = \underbrace{\operatorname{tr} \operatorname{Var}(\hat{\boldsymbol{\beta}}^*)}_{\text{variance}} + \underbrace{\left\|\mathbb{E}[\hat{\boldsymbol{\beta}}^*] - \boldsymbol{\beta}\right\|_2^2}_{\text{bias}}$$

Ridge regression estimates

Ridge regression estimates

The ridge regression estimate (RRE) $\hat{\boldsymbol{\beta}}_R$ of $\boldsymbol{\beta}$ is given by the solution to the equation

$$(X^T X + tI)\hat{\boldsymbol{\beta}}_R = X^T \mathbf{y}.$$

Here t is a constant, called the *selection constant* or *penalty/tuning/shrinkage/biasing/complexity parameter*.

Comments

- If $t = 0$, then $\hat{\boldsymbol{\beta}}_R = \hat{\boldsymbol{\beta}}$.
- The eigenvalues of $X^T X + tI$ are given by $\lambda_j + t$, and since $X^T X$ is positive definite, $\lambda_j \geq 0$. Hence, if t is not too small, then $X^T X + tI$ is not ill-conditioned.
- It is not at all clear that the mean square error $\mathbb{E}\left[\|\hat{\boldsymbol{\beta}}^* - \boldsymbol{\beta}\|_2^2\right]$ is small.

Properties of $\hat{\boldsymbol{\beta}}_R$

- $\hat{\beta}_R$ as a linear transform of $\hat{\beta}$.
- Expected value
- Variance
- Residual sum of squares
- The bias
- The mean squared error

Properties of $\hat{\boldsymbol{\beta}}_R$

- $\hat{\beta}_R$ as a linear transform of $\hat{\beta}$.
- Expected value
- Variance
- Residual sum of squares
- The bias
- The mean squared error

Properties of $\hat{\boldsymbol{\beta}}_R$

$\hat{\beta}_R$ as a linear transform of $\hat{\beta}$.

$$\begin{aligned}\hat{\boldsymbol{\beta}}_R &= (X^TX + tI)^{-1}X^T\mathbf{y} = (X^TX + tI)^{-1}X^TX(X^TX)^{-1}X^T\mathbf{y} \\ &= \underbrace{(X^TX + tI)^{-1}X^TX}_{=:A_t}\hat{\boldsymbol{\beta}}.\end{aligned}$$

Expected value

$$\begin{aligned}\mathbb{E}[\hat{\boldsymbol{\beta}}_R] &= \mathbb{E}\big[(X^TX + tI)^{-1}X^T\mathbf{y}\big] = (X^TX + tI)^{-1}X^T\mathbb{E}\big[\mathbf{y}\big] \\ &= (X^TX + tI)^{-1}X^TX\boldsymbol{\beta}\end{aligned}$$

Variance

$$\begin{aligned}\operatorname{Var}(\hat{\boldsymbol{\beta}}_R) &= \operatorname{Var}\big((X^TX + tI)^{-1}X^T\mathbf{y}\big) \\ &= (X^TX + tI)^{-1}X^T\operatorname{Var}\big(\mathbf{y}\big)\big((X^TX + tI)^{-1}X^T\big)^T \\ &= (X^TX + tI)^{-1}X^T\sigma^2IX(X^TX + tI)^{-1} \\ &= \sigma^2(X^TX + tI)^{-1}X^TX(X^TX + tI)^{-1}\end{aligned}$$

$$\operatorname{tr}\operatorname{Var}(\hat{\boldsymbol{\beta}}_R) = \sigma^2\operatorname{tr}(X^TX + tI)^{-1}X^TX(X^TX + tI)^{-1} = \sigma^2\operatorname{diag}\frac{\lambda_j}{(\lambda_j + t)^2}$$

Properties of $\hat{\boldsymbol{\beta}}_R$

The bias

$$\|\mathbb{E}[\hat{\boldsymbol{\beta}}_R] - \boldsymbol{\beta}\|_2^2 = \left\|\left((X^TX + tI)^{-1}X^TX - I\right)\boldsymbol{\beta}\right\|_2^2 = t^2\boldsymbol{\beta}^T(X^TX + tI)^{-2}\boldsymbol{\beta}$$

Residual sum of squares

$$\begin{aligned} SS_{Res}(\hat{\boldsymbol{\beta}}_R) &= \|\mathbf{y} - X\hat{\boldsymbol{\beta}}_R\|_2^2 = \|\mathbf{y} - X\hat{\boldsymbol{\beta}} + X(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_R)\|_2^2 \\ &= \|\mathbf{y} - X\hat{\boldsymbol{\beta}}\|_2^2 + \|\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_R\|_2^2 + 2(\mathbf{y} - X\hat{\boldsymbol{\beta}})^T(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_R) \\ &= \|\mathbf{y} - X\hat{\boldsymbol{\beta}}\|_2^2 + \|\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_R\|_2^2 + 0 = SS_{Res}(\hat{\boldsymbol{\beta}}) + \|\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_R\|_2^2 \end{aligned}$$

The mean squared error

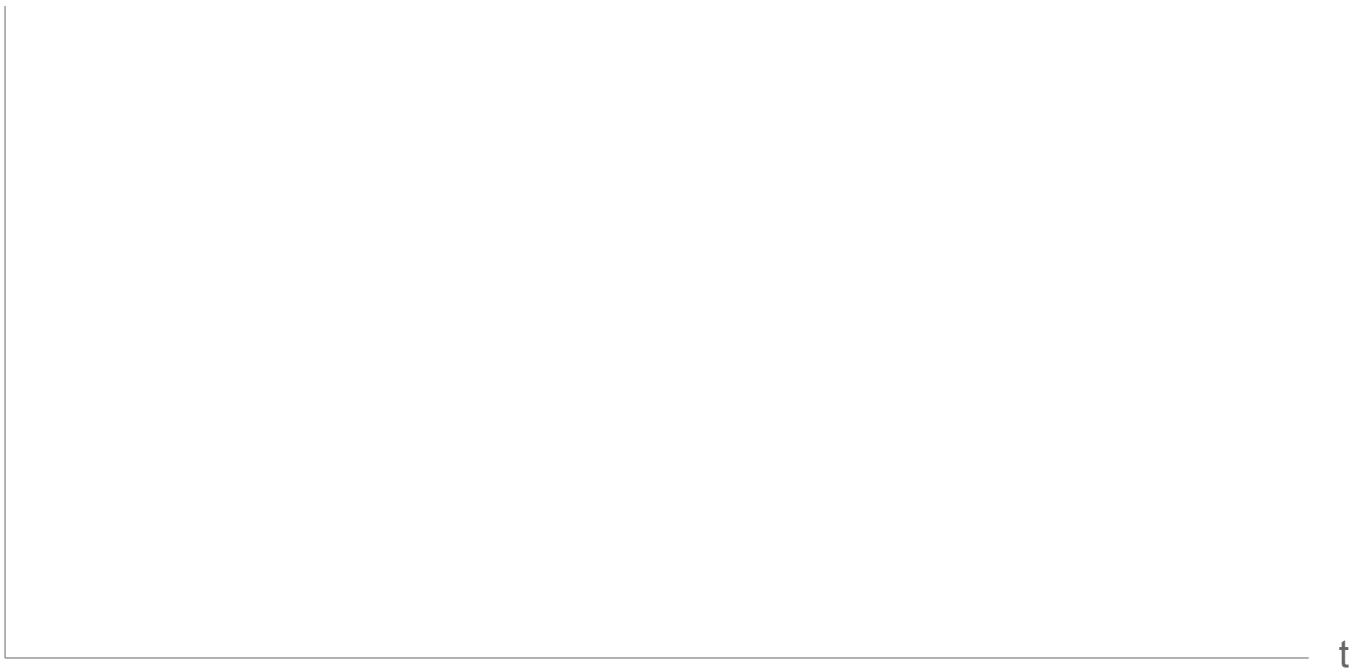
$$\begin{aligned} MSE(\hat{\boldsymbol{\beta}}_R) &= \mathbb{E}\left[\|\hat{\boldsymbol{\beta}}_R - \boldsymbol{\beta}\|_2^2\right] = \overbrace{\operatorname{tr}\operatorname{Var}(\hat{\boldsymbol{\beta}}_R)}^{\text{variance}} + \overbrace{\|\mathbb{E}[\hat{\boldsymbol{\beta}}_R] - \boldsymbol{\beta}\|_2^2}^{\text{bias}} \\ &= \sigma^2 \sum \frac{\lambda_j}{(\lambda_j + t)^2} + t^2\boldsymbol{\beta}^T(X^TX + tI)^{-2}\boldsymbol{\beta} \end{aligned}$$

Properties of $\hat{\boldsymbol{\beta}}_R$

The mean squared error

$$MSE(\hat{\boldsymbol{\beta}}_R) = \mathbb{E}\left[\|\hat{\boldsymbol{\beta}}_R - \boldsymbol{\beta}\|_2^2\right] = \overbrace{\operatorname{tr}\operatorname{Var}(\hat{\boldsymbol{\beta}}_R)}^{\text{variance}} + \overbrace{\|\mathbb{E}[\hat{\boldsymbol{\beta}}_R] - \boldsymbol{\beta}\|_2^2}^{\text{bias}}$$

$$= \sigma^2 \sum \frac{\lambda_j}{(\lambda_j + t)^2} + t^2 \boldsymbol{\beta}^T (X^T X + tI)^{-2} \boldsymbol{\beta}$$



Example

```
1 library("lmridge")
2
3 df01.model <- lmridge(people_fully_vaccinated_per_hundred~ gdp_
      per_capita + hospital_beds_per_thousand + total_cases_per_
      million + total_deaths_per_million + median_age + aged_65_
      older + cardiovasc_death_rate + diabetes_prevalence + male_
      smokers + life_expectancy, data = df01, K=.026)
4
5 summary(df01.model)
```

```
Coefficients: for Ridge parameter K= 0.026
                    Est. Est. (Sc)  SE (Sc)  t (Sc)  Pr(>|t|)
Intercept       -99.0190  -3271400  1468100 -2.2283    0.0278 *
gdp_per_capita    0.0002     48.182  20.124   2.3943    0.0183 *
hospital_beds_p   0.0388      1.066  19.271   0.0553    0.9560
total_cases_p     0.0000     33.383  20.887   1.5983    0.1127
total_deaths_p   -0.0040    -44.642  20.802  -2.1460    0.0340 *
median_age        1.0469    102.530  35.335   2.9015    0.0044 **
aged_65_older    -0.6642    -48.361  30.369  -1.5925    0.1140
cardiovasc_dea.  -0.0404    -54.985  18.826  -2.9207    0.0042 **
diabetes_prev     0.5462     23.381  16.170   1.4459    0.1509
male_smokers      0.0475     70.928  16.520   0.4293    0.6685
life_expectancy   1.5749    116.130  28.003   4.1473    0.0001 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

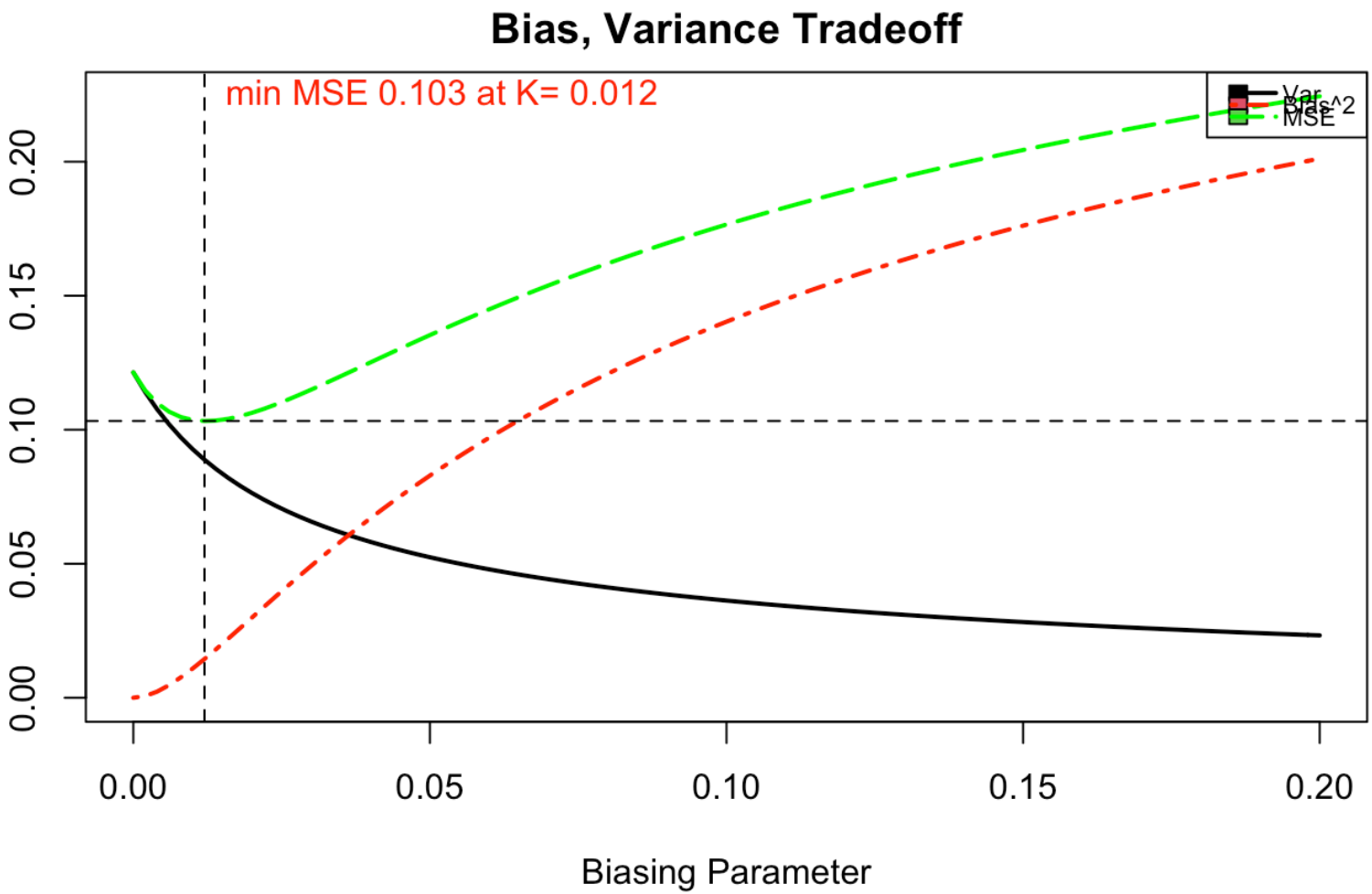
Example

```
library("lmridge")

df01.model00 <- lmridge(people_fully_vaccinated_per_hundred~ gdp
    _per_capita + hospital_beds_per_thousand + total_cases_per_
    million + total_deaths_per_million + median_age + aged_65_
    older + cardiovasc_death_rate + diabetes_prevalence + male_
    smokers + life_expectancy, data = df01, K=.026)

summary(df01.model00)
```

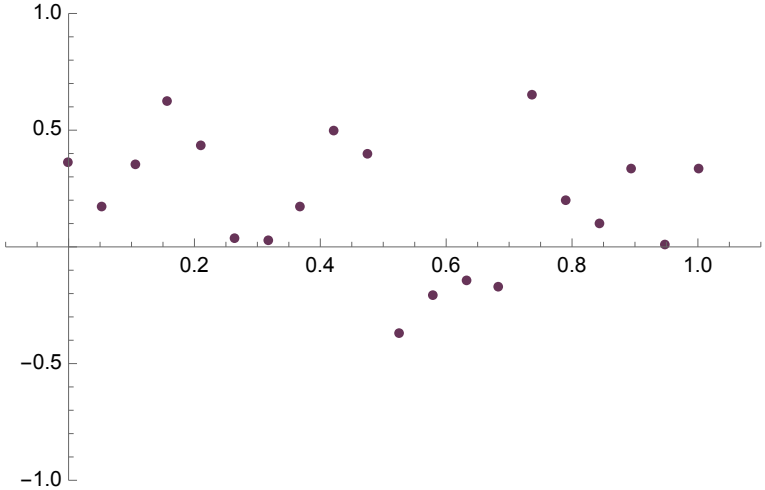
```
Ridge Summary
       R2     adj-R2  DF ridge          F        AIC        BIC
   0.7189     0.6971     9.0387    33.1875  671.4730 1306.4807

Ridge minimum MSE= 9189.994 at K= 0.026

P-value for F-test ( 9.03866 , 116.2246 ) = 3.200431e-28
```

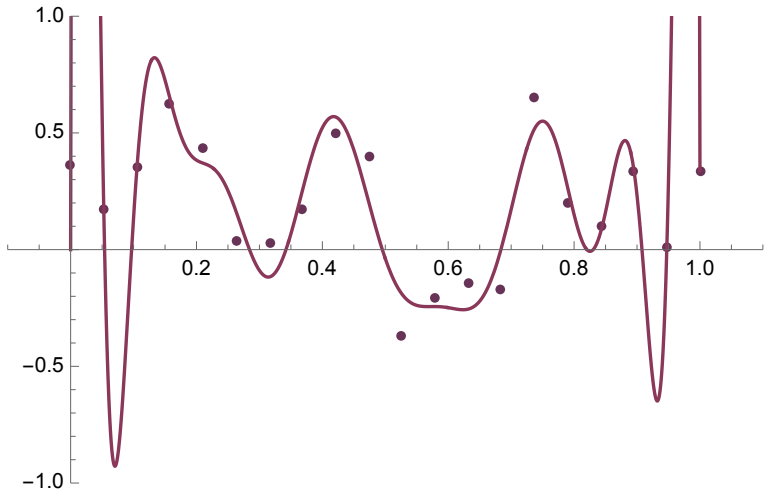
Example

```
1 bias.plot(df01.model00,abline=FALSE)
```

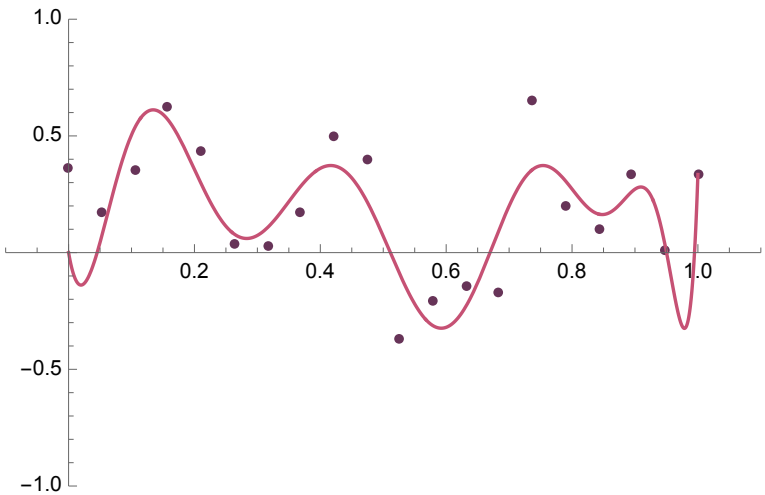


Example

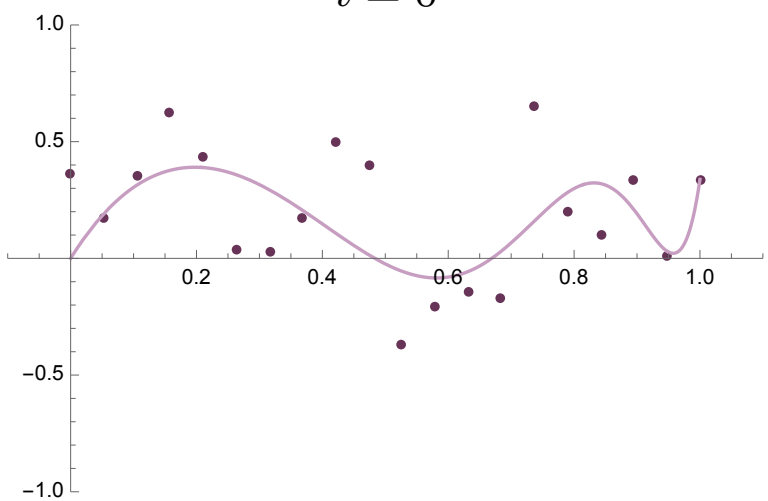




$t = 0$



$t = 10^{-10}$



$t = 10^{-5}$

How to choose the biasing parameter?

The ridge trace

The plot of the RREs $\hat{\boldsymbol{\beta}}_R(j)$ as a function of the biasing parameter t is called the *ridge trace*.

Idea

The RREs $\hat{\boldsymbol{\beta}}_R(j)$ will stabilize quite quickly as t increases.

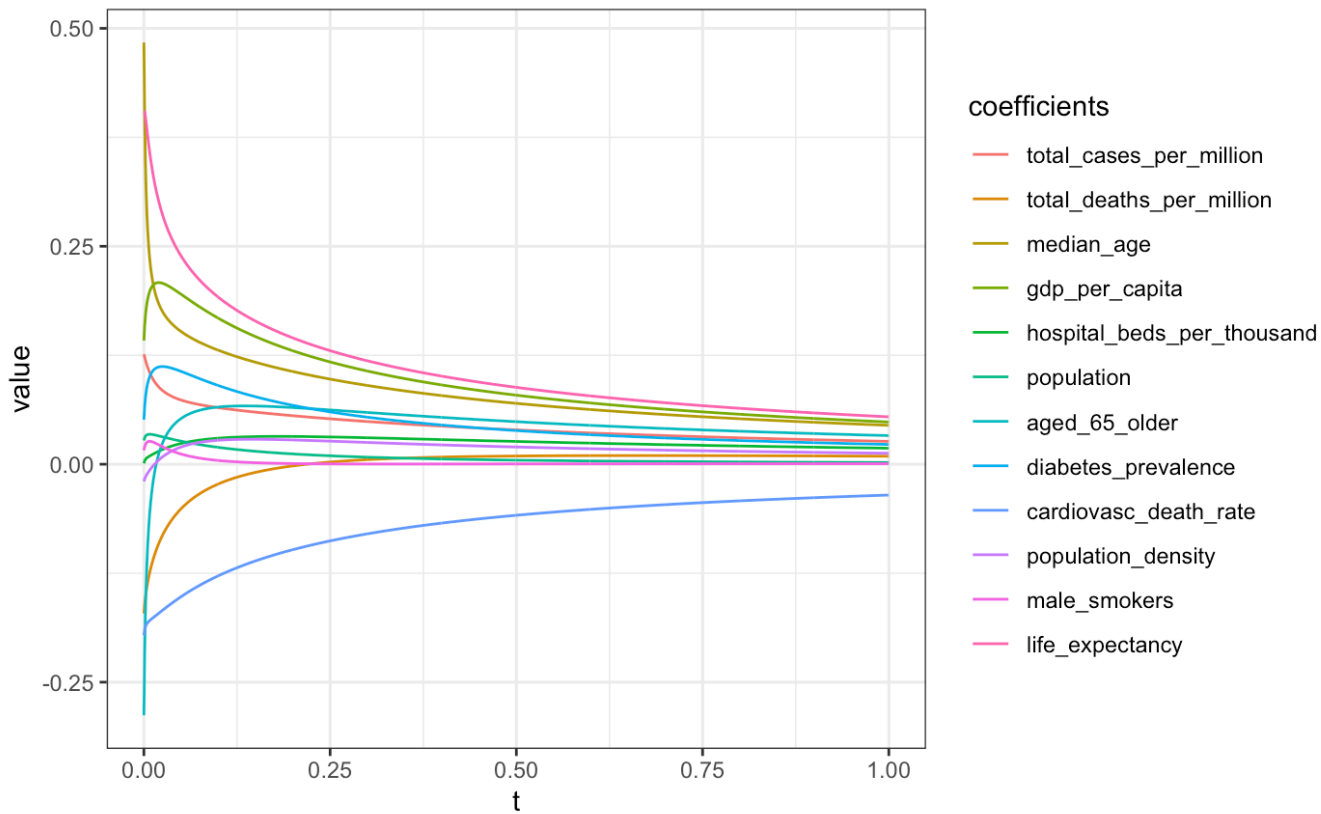
Method

1. Plot the RREs $\hat{\boldsymbol{\beta}}_R(j)$ as functions of the biasing parameter t for some finite set of values for t.
2. Pick (by inspection) a small t for which the traces seem to have stabilized.

How to choose the biasing parameter?

Method

1. Plot the RREs $\hat{\boldsymbol{\beta}}_R(j)$ as functions of the biasing parameter t for some finite set of values for t.
2. Pick (by inspection) a small t for which the traces seem to have stabilized.

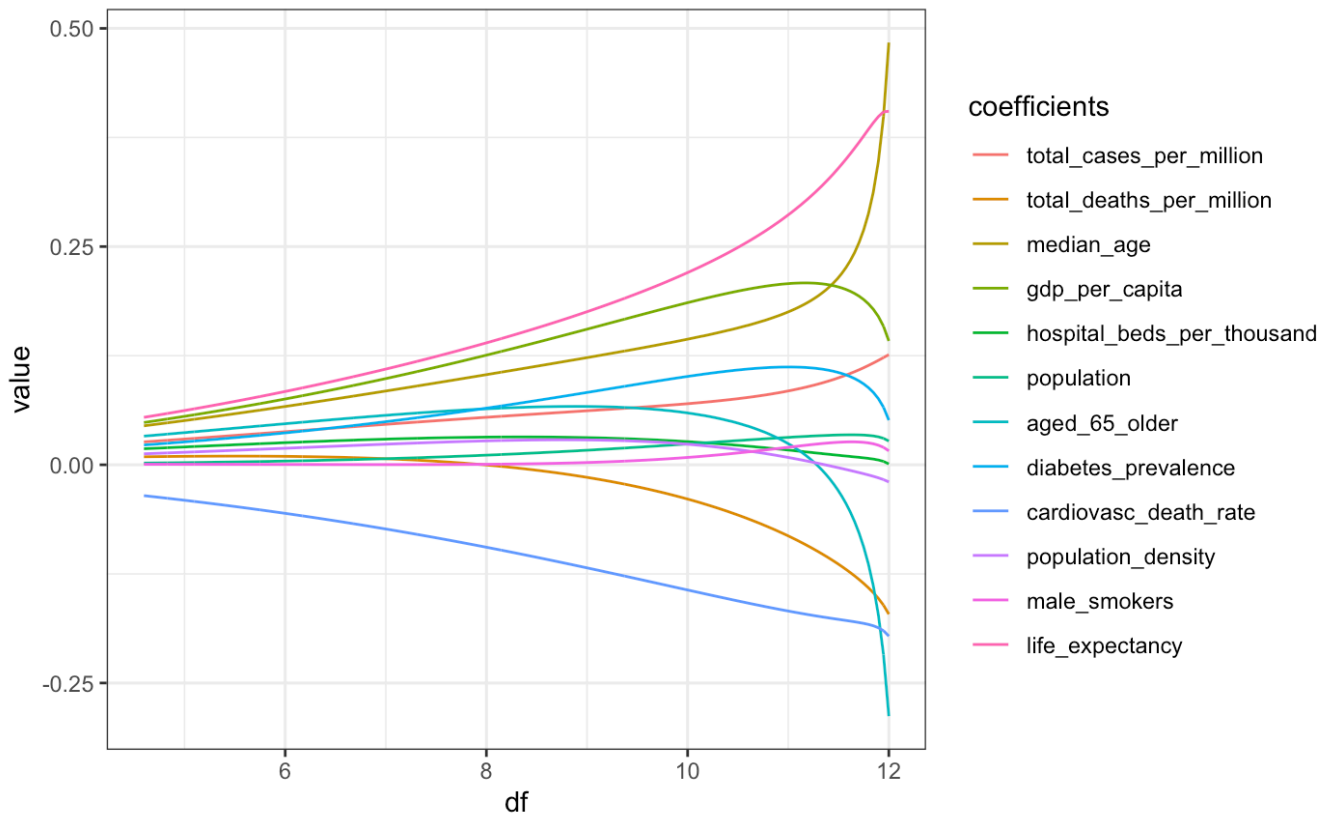


How to choose the biasing parameter?

The effective degrees of freedom

Iz suggests plotting $\hat{\boldsymbol{\beta}}_R$ against the *effective degrees of freedom*, defined by

$$df(t) := \operatorname{tr} H_t := \operatorname{tr} X(X^T X + tI)^{-1} X^T$$



Ridge regression as a shrinkage operator

Ridge regression as a shrinkage operator

Recall the decomposition $X^TX = PDP^T$, and assume that X^TX is invertible. Then

$$\begin{aligned}\hat{\boldsymbol{\beta}}_R &= (X^TX + tI)X^T\mathbf{y} = (X^TX + tI)X^TX(X^TX)^{-1}X^T\mathbf{y} \\ &= (X^TX + tI)(X^TX)\hat{\boldsymbol{\beta}}\end{aligned}$$

and

$$\begin{aligned}(X^TX + tI)^{-1}X^TX &= (PDP^T + tI)^{-1}PDP^T \\ &= (PDP^T + tPP^T)^{-1}PDP^T = (P(D + tI)P^T)^{-1}PDP^T \\ &= P(D + tI)^{-1}P^TPDP^T = P(D + tI)^{-1}DP^T\end{aligned}$$

Hence

$$\hat{\boldsymbol{\beta}}_R = P(D + tI)^{-1}DP^T\hat{\boldsymbol{\beta}} = P(D + tI)^{-1}DP^T\hat{\boldsymbol{\beta}}.$$

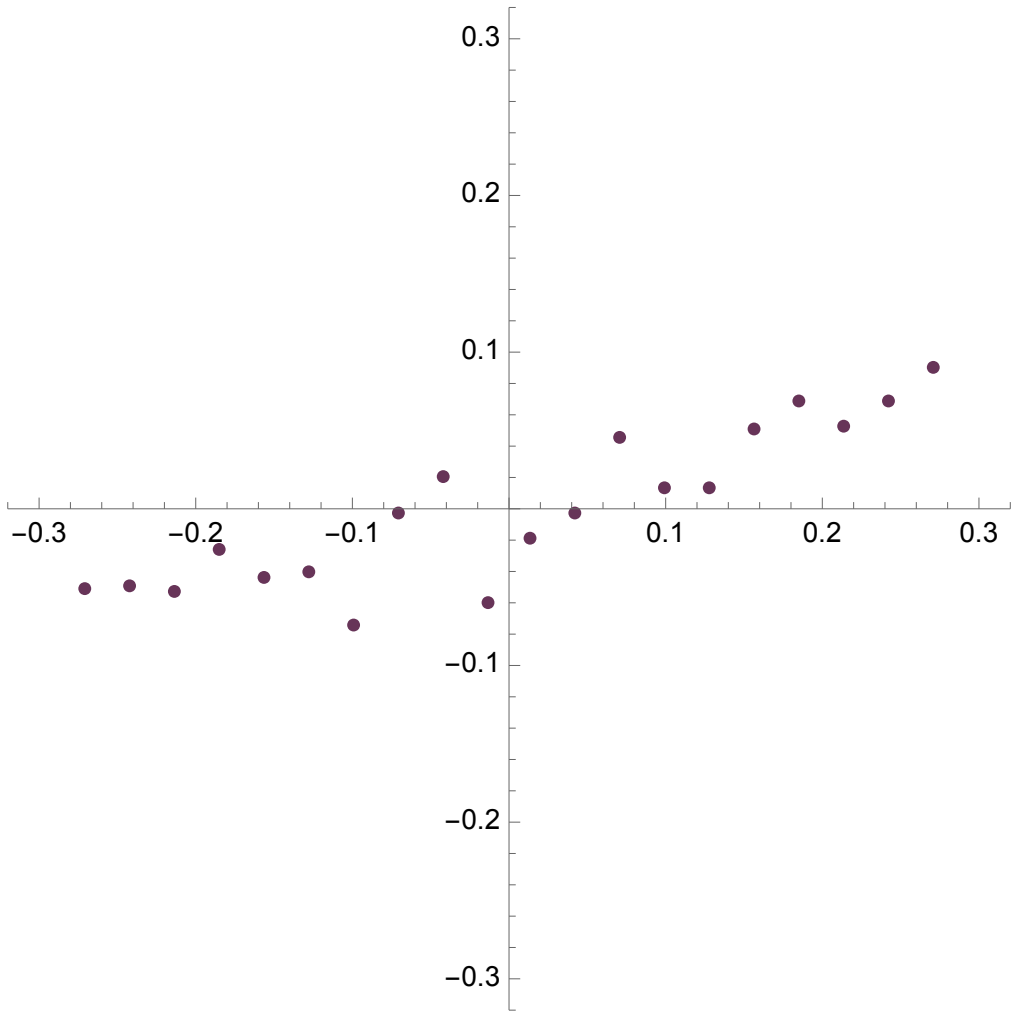
Ridge regression vs. PCR

Ridge regression

$$\hat{\boldsymbol{\beta}}_R = P(D + tI)^{-1} D P^T \hat{\boldsymbol{\beta}}$$

Principal components regression

$$\hat{\boldsymbol{\beta}}_{PCR} = P \operatorname{diag}\big(\mathbf{1}(\lambda_j > \delta)\big) P^T \hat{\boldsymbol{\beta}}$$



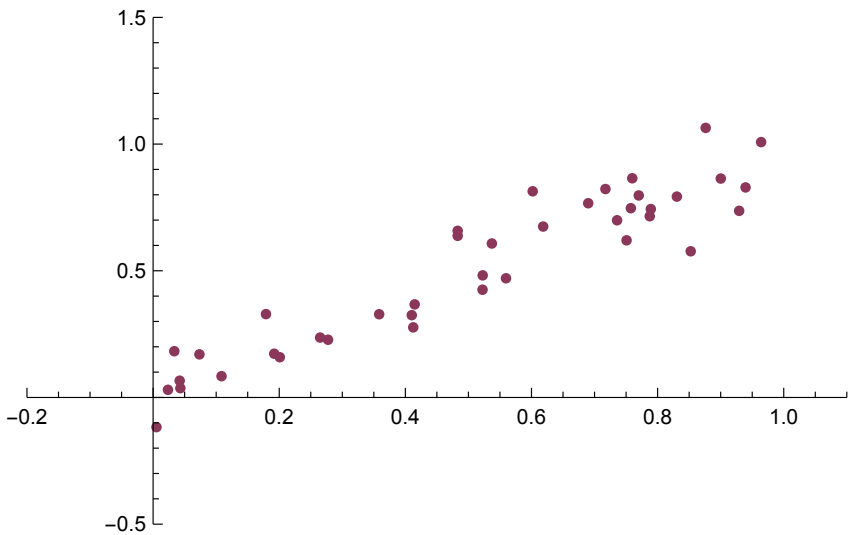
Resampling techniques for model assessment

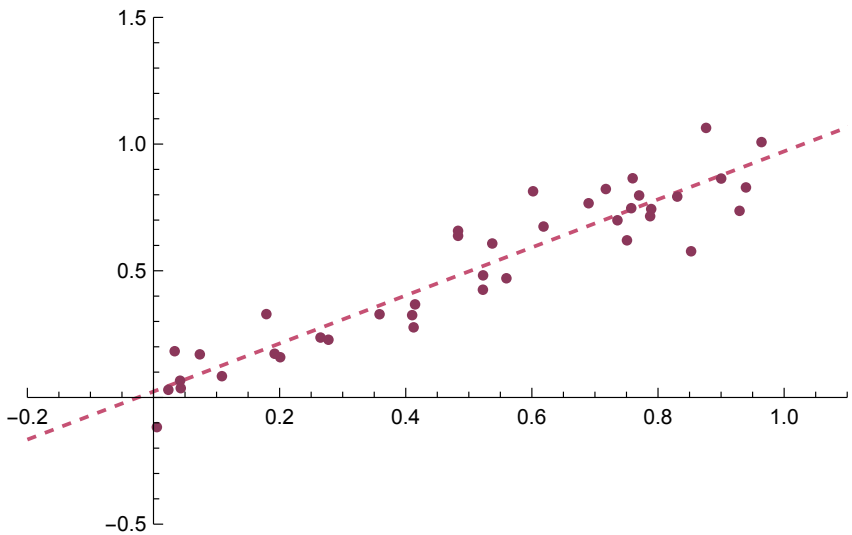
We will now start developing methods to validate the models we develop using linear regression. In this course, will discuss two such methods:

- **Cross validation (random regressors)**
- Bootstrap (random or non-random regressors)

Cross validation

Assume that $(\mathbf{x}_j, y_j)_{j\in\{1,2,\ldots,n\}}$ is a random sample with random regressors.





Cross validation

Assume that $(\mathbf{x}_j, y_j)_{j\in\{1,2,\ldots,n\}}$ is a random sample with random regressors.

Basic idea

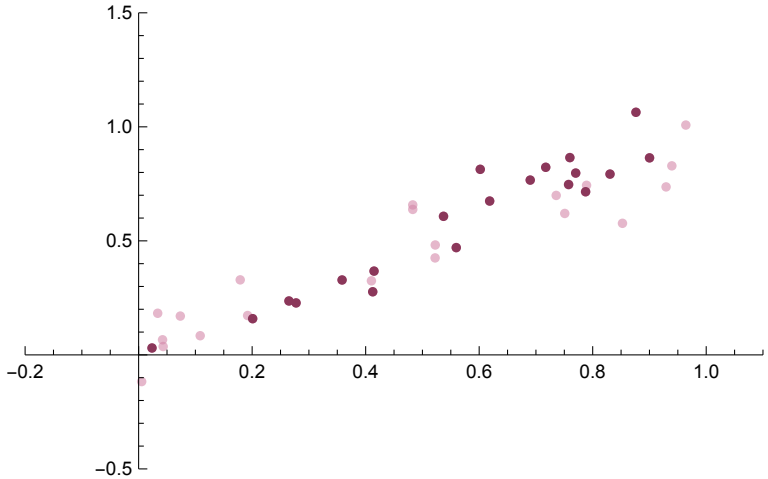
Assume that n is even.

1. Let $T \cup V$ be a random partition of $[n]$ into two sets of equal size. Then the sets of observations $(\mathbf{x}_j, y_j)_{j\in T}$ and $(\mathbf{x}_j, y_j)_{j\in V}$ can be though of as two separate random samples, though of as the *training set* and the *validation set*.
2. Apply linear regression to the training set to find en estimate of the regression coefficients $\hat{\boldsymbol{\beta}}$.
3. Use the validation set to estimate, e.g., the prediction error

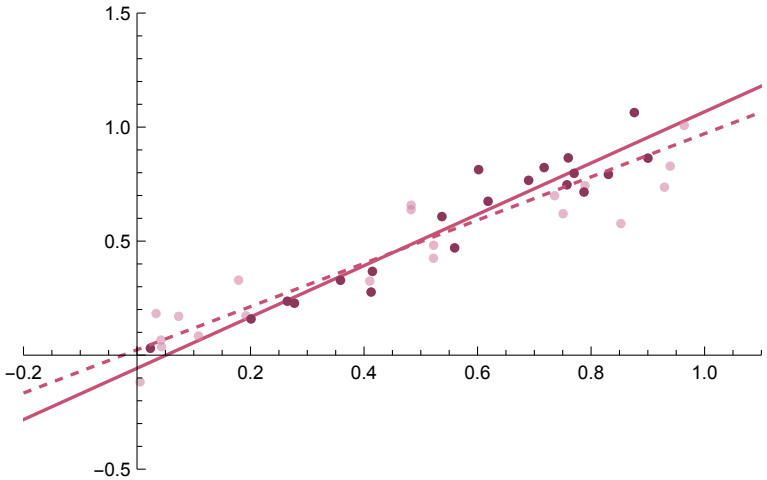
$$\widehat{PE} = \frac{1}{|V|}\sum_{j\in V}(y_j - X_{j,\cdot}\hat{\boldsymbol{\beta}}^{(T)})^2$$

4. Switch the roles of T and V to obtain another estimate of $\widehat{PE}$, and take the average of these for a better estimate.

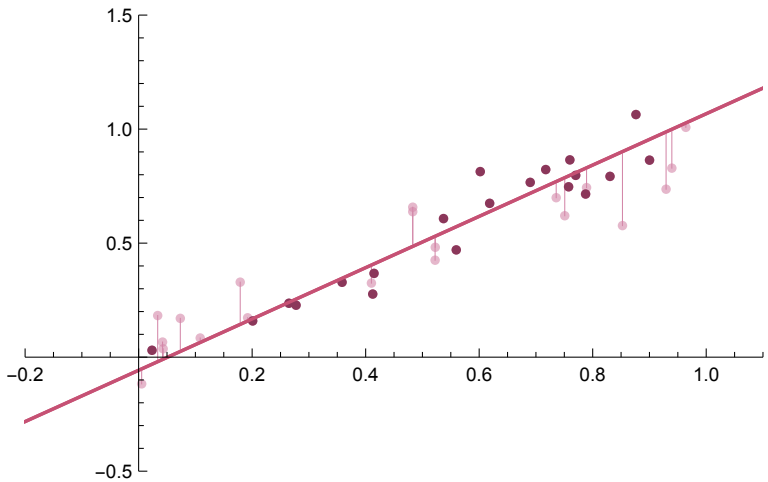
Cross validation



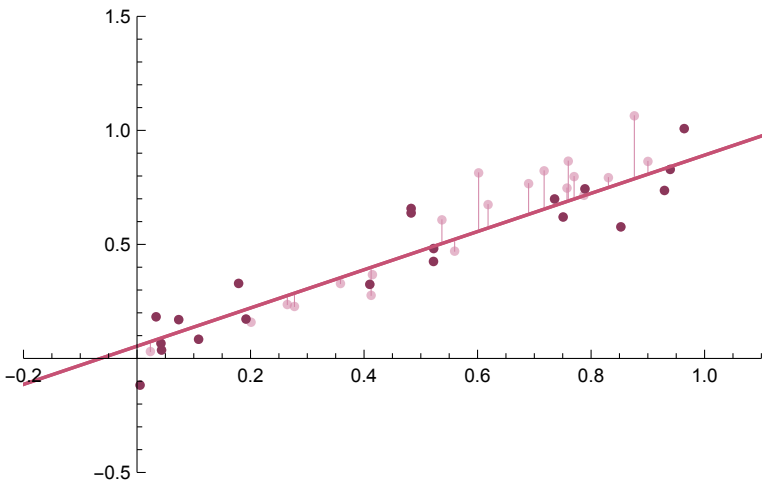
Divide the sample into a training set T and a validation set V.



Fit a model using the training set.

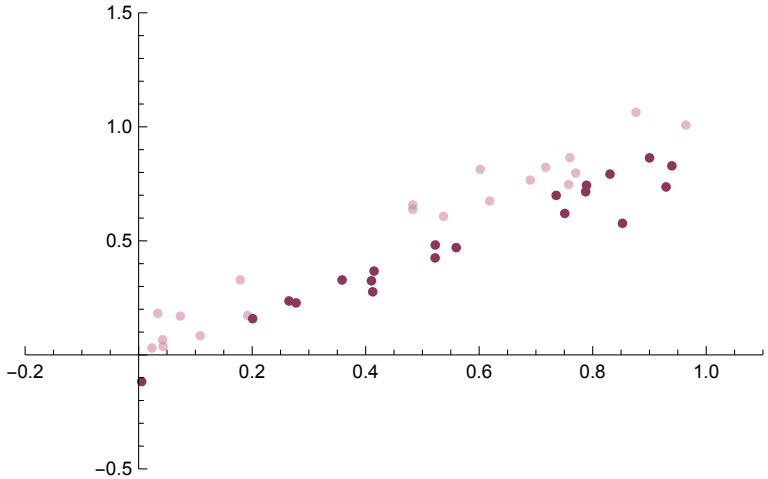


Calculate the mean of the squared residuals using the validation set.

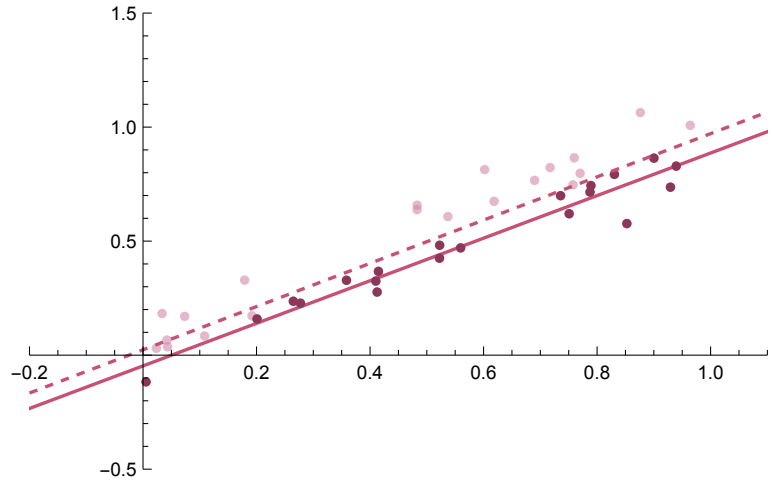


Switch the roles of T and V.

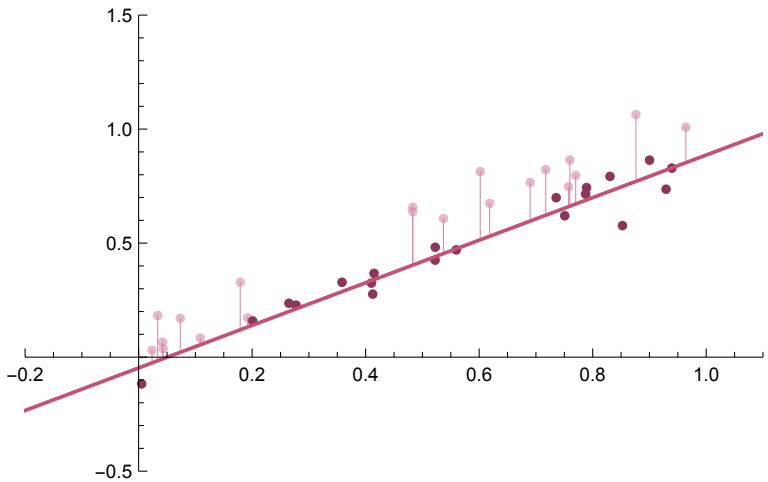
Cross validation



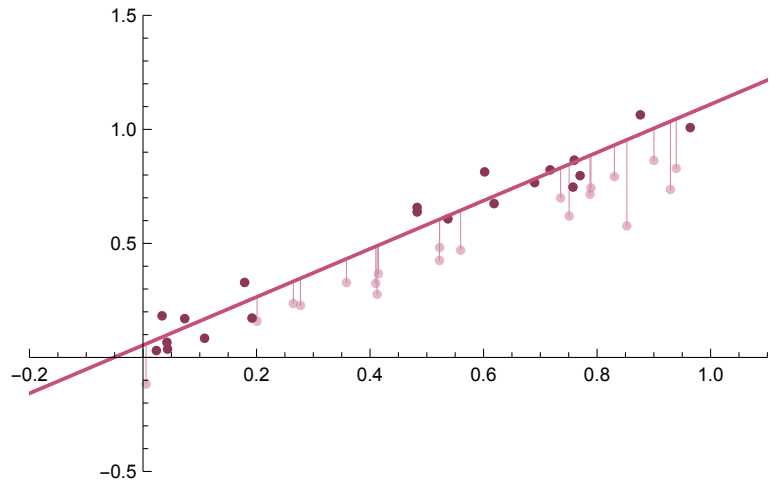
Divide the sample into a training set T and a validation set V.



Fit a model using the training set.

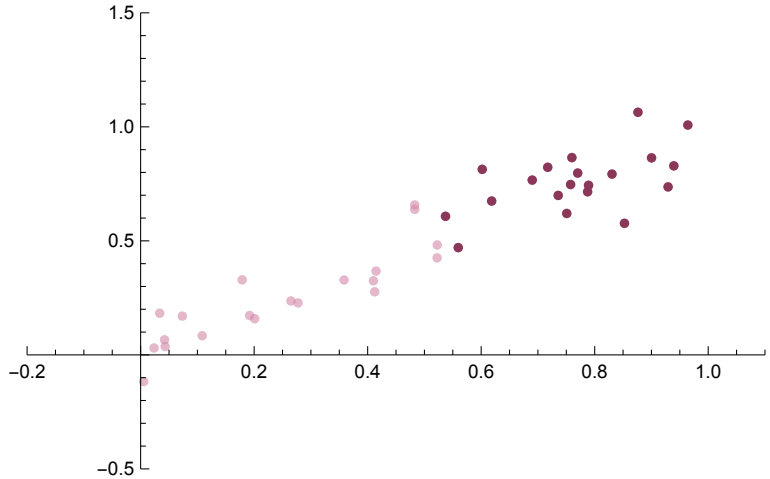


Calculate the mean of the squared residuals using the validation set.

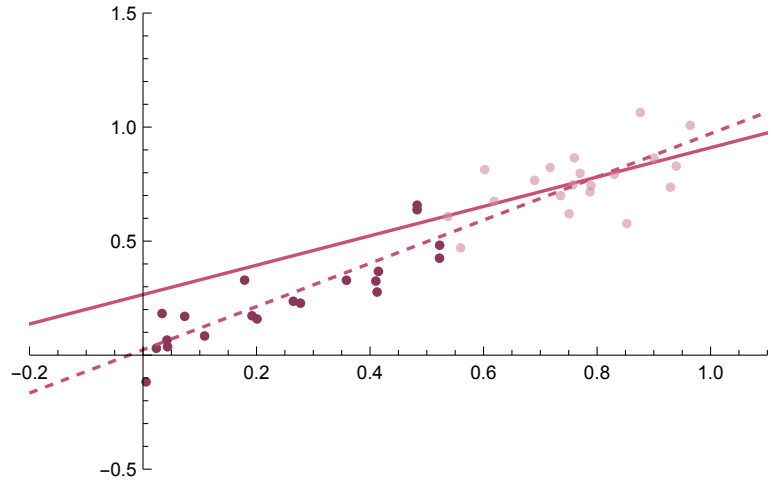


Switch the roles of T and V.

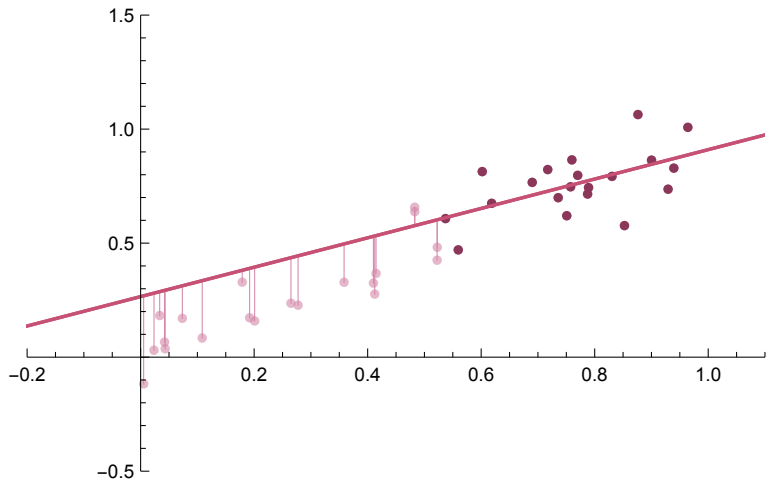
Cross validation



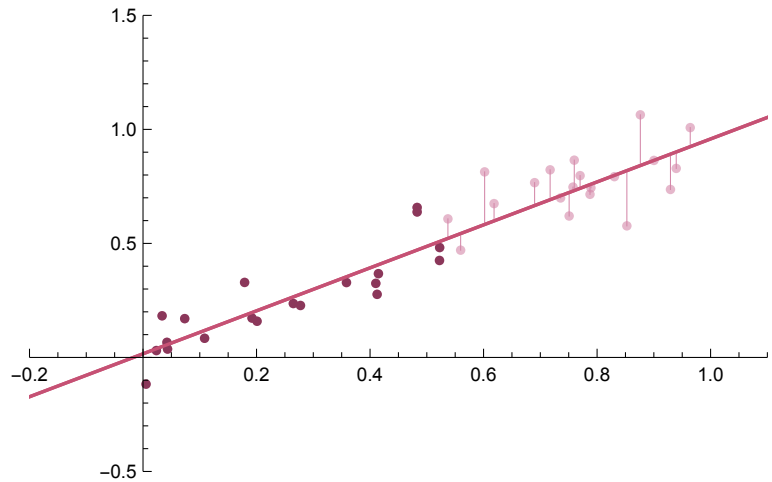
Divide the sample into a training set T and a validation set V.



Fit a model using the training set.



Calculate the mean of the squared residuals using the validation set.



Switch the roles of T and V.

Cross validation

Assume that $(\mathbf{x}_j, y_j)_{j\in\{1,2,\ldots,n\}}$ is a random sample with random regressors.

General idea (m-fold cross validation)

Assume that $m|n$.

1. Let $T_1 \cup T_2 \cup \ldots \cup T_{n/m}$ be a random partition of $[n]$ into sets of size m. For each $i \in \{1, 2, \ldots, n/m\}$, the sets $(\mathbf{x}_j, y_j)_{j\in\{1,2,\ldots,n\}\setminus T_i}$ and $(\mathbf{x}_j, y_j)_{j\in T_i}$ can be though of as separate random samples, though of as the *training set* and the *validation set*.
2. Apply linear regression to the training set to find en estimate of the regression coefficients $\hat{\boldsymbol{\beta}}$.
3. Use the validation set to estimate, e.g., the prediction error,

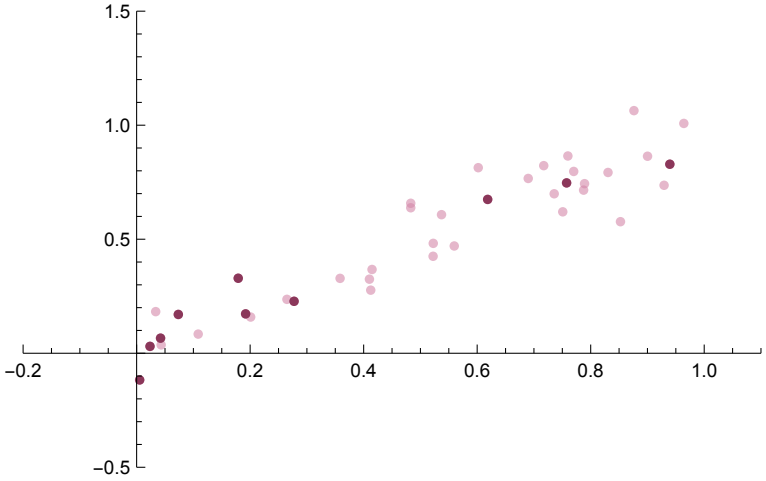
$$\widehat{PE}_i = \frac{1}{n - |T_i|} \sum_{j\in V_i} (y_j - \mathbf{x}_j^T \hat{\boldsymbol{\beta}}^{(T_i)})^2$$

4. Obtain a better estimate by averaging over each $i \in \{1, 2, \ldots, n - m\}$.

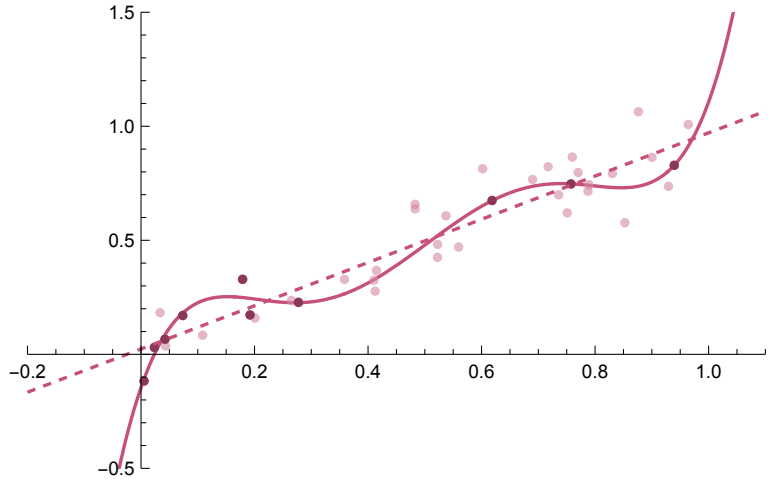
$$\widehat{PE}_{CV/m} = \frac{1}{n/m} \sum_{i=1}^{n/m} \widehat{PE}_i.$$

$\rightarrow$ Note that we have used this idea before with $m = 1$.

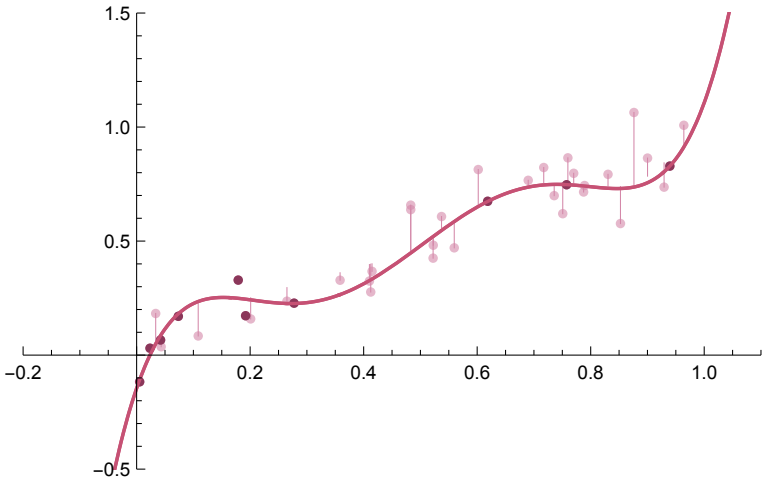
Cross validation



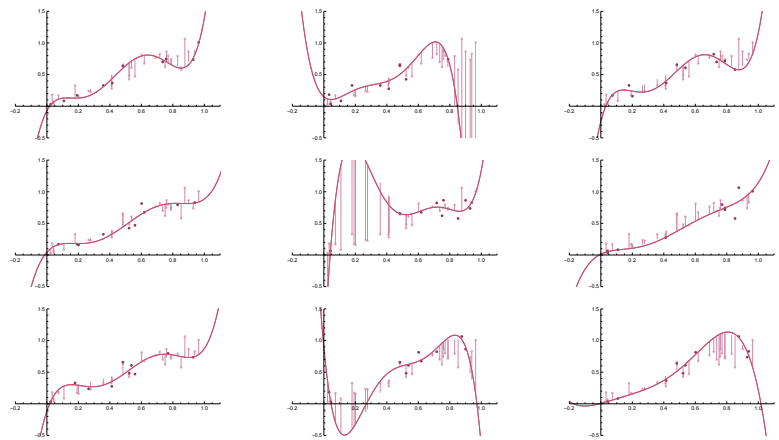
Divide the sample into a training set T_i and a validation set $V = \bigcup_{j \neq i} T_i$.



Fit a model using the training set.



Calculate the mean of the squared residuals using the validation set.



Repeat for each potential training set T_i.

Example: How to choose the biasing parameter?

Ridge regression estimates

The ridge regression estimate (RRE) $\hat{\boldsymbol{\beta}}_R$ of $\boldsymbol{\beta}$ is given by the solution to the equation

$$(X^T X + tI)\hat{\boldsymbol{\beta}}_R = X^T \mathbf{y}.$$

Method

Using cross-validation, we can pick the parameter t which minimizes the estimated prediction error $\widehat{PE}_{CV/m} = \frac{1}{n/m} \sum_{i=1}^{n/m} \widehat{PE}_i$.

Example

```
X <- data.matrix(df01[, c('total_cases_per_million', 'total_
    deaths_per_million', 'median_age', 'gdp_per_capita','
    hospital_beds_per_thousand', 'population', 'aged_65_
    older','diabetes_prevalence','cardiovasc_death_rate','
    population_density','male_smokers','life_expectancy' )])
y <- df01$people_fully_vaccinated_per_hundred
X <- scale(X)/sqrt(nrow(X)-1)
y <- (y-mean(y))/sqrt(sum((y-mean(y))^2))

library(glmnet)
cv_model <- cv.glmnet(X, y, alpha = 0, nfolds=10)
print(cv_model$lambda.min)
plot(cv_model)
```

```
[1] 0.03839179
```