Machine Learning

Cross-validation

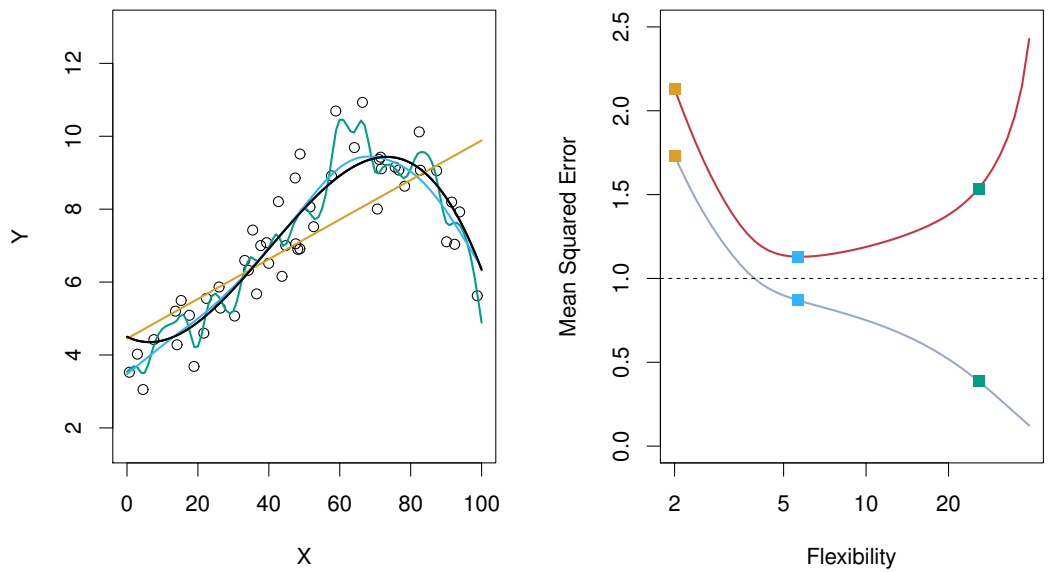
Sebastian Engelke

Master in Business Analytics

UNIVERSITÉ
DE GENÈVE

Model selection

- The training error $\mathrm{MSE}_{\mathrm{Tr}}$ is not suited for `model selection` since a method that overfits the data may have a small $\mathrm{MSE}_{\mathrm{Tr}}$ but perform badly on new test data.
- A good model should be able to predict well on new data, thus we choose the model $\widehat{f}$ that minimizes the `expected prediction error` $\mathrm{Err}_{\widehat{f}}$.
- We only have one data set, how can be obtain an estimate of $\mathrm{Err}_{\widehat{f}}$?

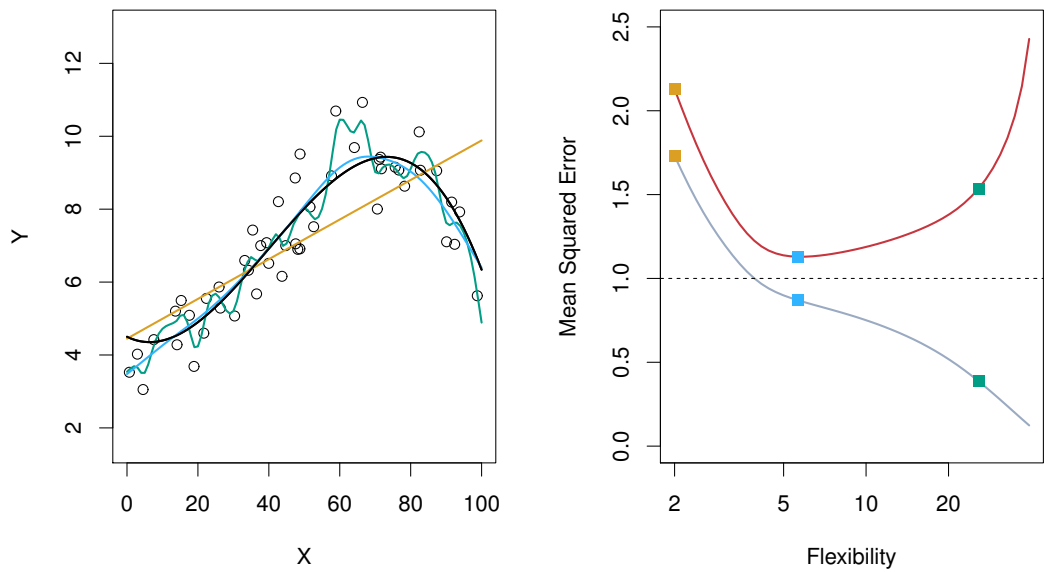


Model selection

- The training error $\mathrm{MSE}_{\mathrm{Tr}}$ is not suited for `model selection` since a method that overfits the data may have a small $\mathrm{MSE}_{\mathrm{Tr}}$ but perform badly on new test data.
- A good model should be able to predict well on new data, thus we choose the model $\widehat{f}$ that minimizes the `expected prediction error` $\mathrm{Err}_{\widehat{f}}$.
- We only have one data set, how can be obtain an estimate of $\mathrm{Err}_{\widehat{f}}$?

Two classes of approaches

- Theoretical approximations of $\mathrm{Err}_{\widehat{f}}$: `Akaike Information Criterion` (AIC), `Bayesian Information Criterion` (BIC), etc. (only works for certain model classes)
- Estimation of $\mathrm{Err}_{\widehat{f}}$ based on the `test error` $\mathrm{MSE}_{\mathrm{Te}}$: `Bootstrap`, `Cross-Validation`, etc.

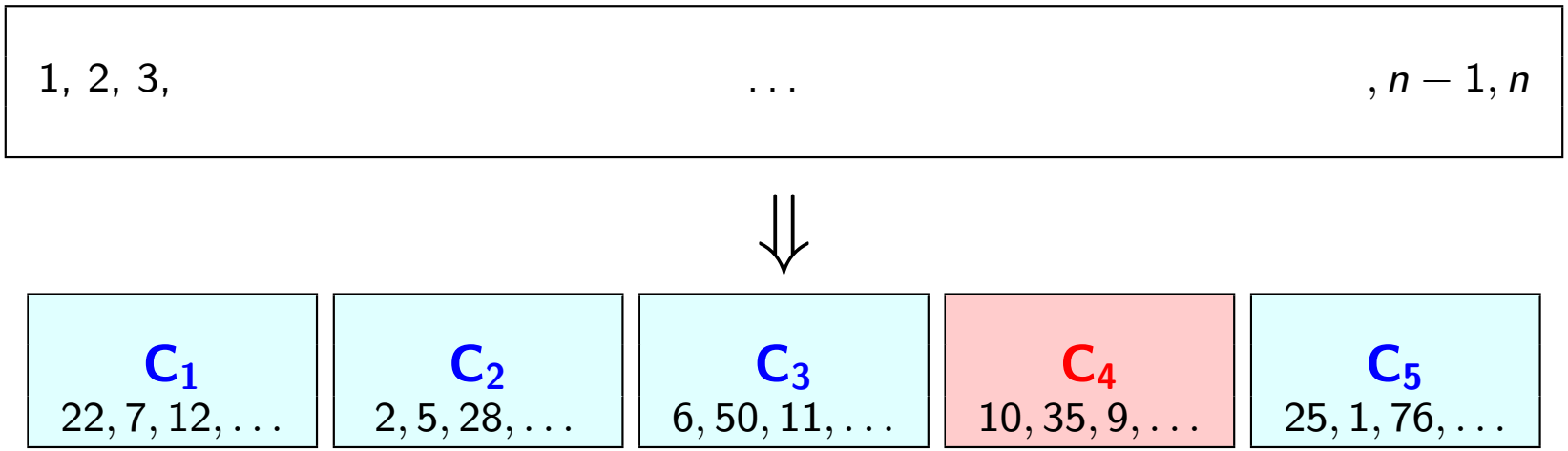


K-Fold Cross-Validation

1, 2, 3,	...	$, n-1, n$

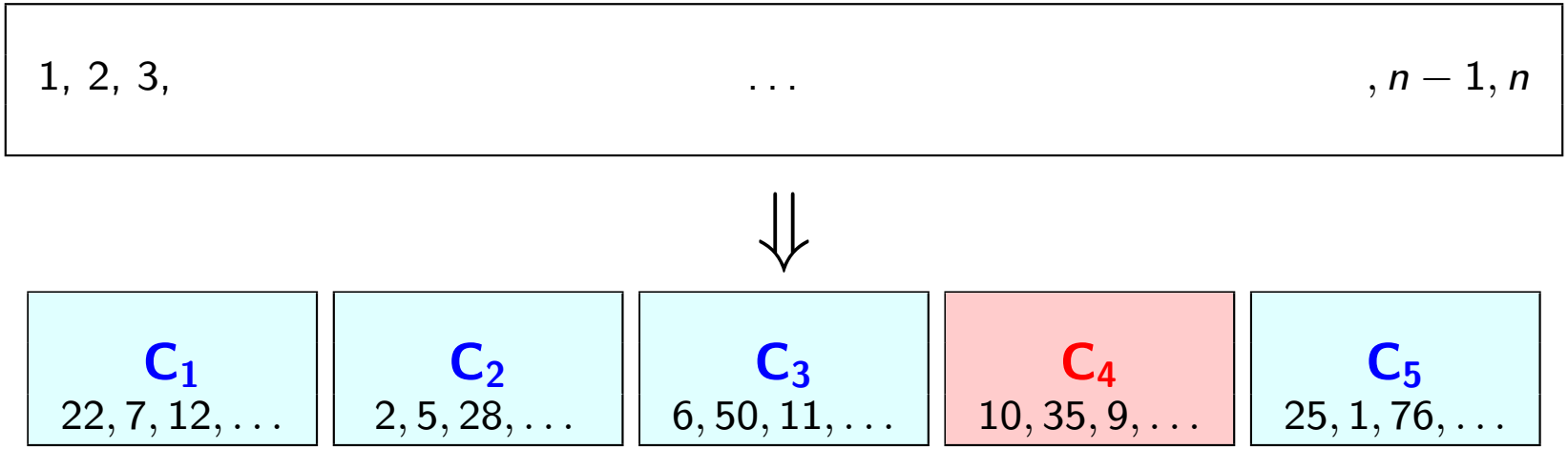
- We only have `one data set` $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, how do we obtain a new test sample?

K-Fold Cross-Validation



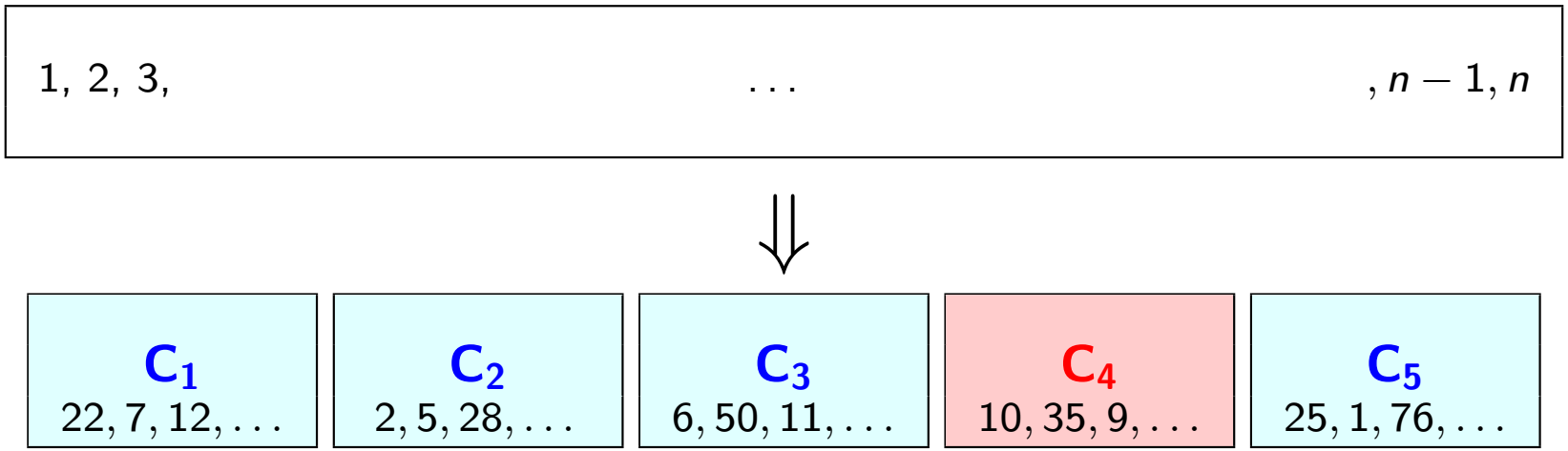
- We only have `one data set` $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, how do we obtain a new test sample?
- For K`-Fold Cross-Validation` we randomly divide the data into K groups $C_1, \ldots, C_K$, also called folds, of approximately equal size n/K.

K-Fold Cross-Validation



- We only have one data set $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, how do we obtain a new test sample?
- For K-Fold Cross-Validation we randomly divide the data into K groups $C_1, \ldots, C_K$, also called folds, of approximately equal size n/K.
- We treat the kth group C_k as validation/test set and fit a model $\widehat{f}_{-k}$ on the remaining $K-1$ groups. We then compute the $\mathrm{MSE}_k = (K/n)\sum_{i \in C_k}\{y_i - \widehat{f}_{-k}(x_i)\}^2$ on the held-out group. This is done successively for all groups C_k, $k = 1, \ldots, K$.

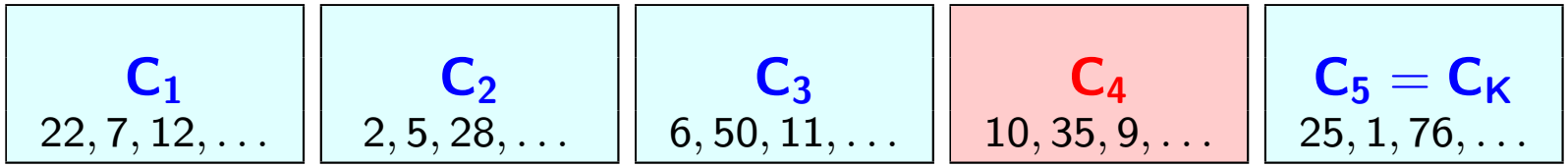
K-Fold Cross-Validation



- We only have one data set $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, how do we obtain a new test sample?
- For K-Fold Cross-Validation we randomly divide the data into K groups $C_1, \ldots, C_K$, also called folds, of approximately equal size n/K.
- We treat the kth group C_k as validation/test set and fit a model $\widehat{f}_{-k}$ on the remaining $K-1$ groups. We then compute the $\mathrm{MSE}_k = (K/n)\sum_{i \in C_k}\{y_i - \widehat{f}_{-k}(x_i)\}^2$ on the held-out group. This is done successively for all groups C_k, $k = 1, \ldots, K$.
- The K-fold CV estimate is the average of these values

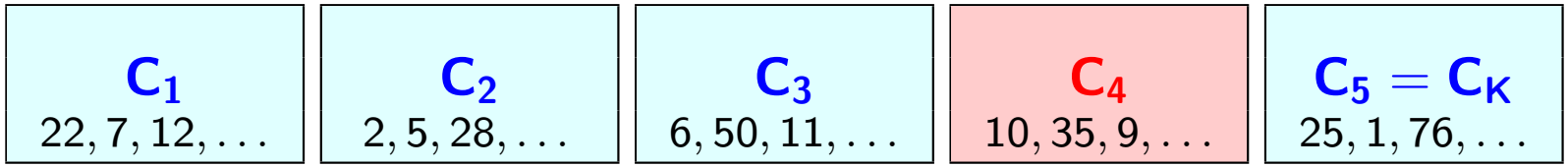
$$\mathrm{CV}_{(K)} = \frac{1}{K}\sum_{k=1}^{K}\mathrm{MSE}_k.$$

K-Fold Cross-Validation



- The K-fold CV estimate is widely used and generic approach that can be applied in most cases.
- This estimate can be used to select the best model, and to give an idea of the test error $\mathrm{Err}_{\widehat{f}}$ of the final model.
- A good choice for K depends on the size n of the data set, but typically $K = 5$ or $K = 10$ is used.

K-Fold Cross-Validation



- The *K-fold CV estimate* is widely used and generic approach that can be applied in most cases.
- This estimate can be used to *select the best model*, and to give an idea of the test error $\mathrm{Err}_{\widehat{f}}$ of the final model.
- A good choice for K depends on the size n of the data set, but typically $K = 5$ or $K = 10$ is used.

Important:

- We automatically get an idea of the *uncertainty* of the K-fold CV estimate $\mathrm{CV}_{(K)}$, by

$$\widehat{\mathrm{SE}}(\mathrm{CV}_{(K)}) = \frac{1}{\sqrt{K}} \sqrt{\sum_{k=1}^{K} \left(\mathrm{MSE}_k - \mathrm{CV}_{(K)}\right)^2 / (K-1)}.$$

Leave-One-Out Cross-Validation

- One drawback of K-fold CV is that it requires to fit K `different models`, which can be computationally costly.

Leave-One-Out Cross-Validation

- One drawback of K-fold CV is that it requires to fit K `different models`, which can be computationally costly.
- A nice special case of K-fold CV is `Leave-One-Out CV (LOOCV)`, namely each of the observations is treated as a separate fold, that is, $K = n$ and $C_k = \{k\}$, $k = 1, \ldots, n$.
- The `Leave-One-Out CV (LOOCV) estimate` is then

$$\mathrm{CV}_{(n)} = \frac{1}{n} \sum_{k=1}^{n} \{y_k - \widehat{f}_{-\{k\}}(x_k)\}^2.$$

- In general, this requires even more models to be fit, but in `linear models` there is a convenient approximation

$$\mathrm{CV}_{(n)} \approx \frac{1}{n} \sum_{k=1}^{n} \left[\frac{y_k - \widehat{f}(x_k)}{1 - S_{kk}} \right]^2,$$

where S_{kk} are constants automatically available from the linear model fit.

Leave-One-Out Cross-Validation

- One drawback of K-fold CV is that it requires to fit K different models, which can be computationally costly.
- A nice special case of K-fold CV is Leave-One-Out CV (LOOCV), namely each of the observations is treated as a separate fold, that is, $K = n$ and $C_k = \{k\}$, $k = 1, \ldots, n$.
- The Leave-One-Out CV (LOOCV) estimate is then

$$\mathrm{CV}_{(n)} = \frac{1}{n} \sum_{k=1}^{n} \{y_k - \widehat{f}_{-\{k\}}(x_k)\}^2.$$

- In general, this requires even more models to be fit, but in linear models there is a convenient approximation

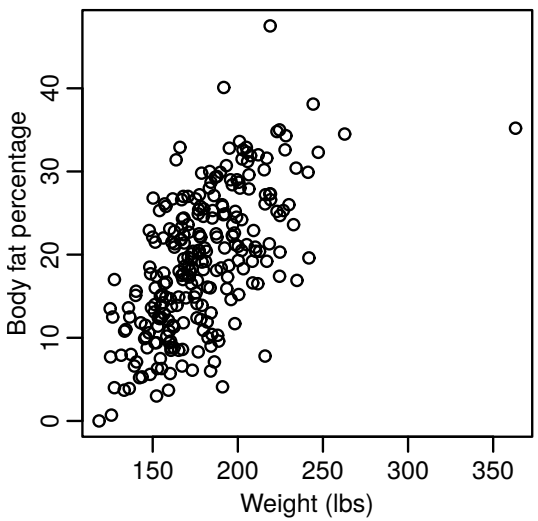
$$\mathrm{CV}_{(n)} \approx \frac{1}{n} \sum_{k=1}^{n} \left[\frac{y_k - \widehat{f}(x_k)}{1 - S_{kk}} \right]^2,$$

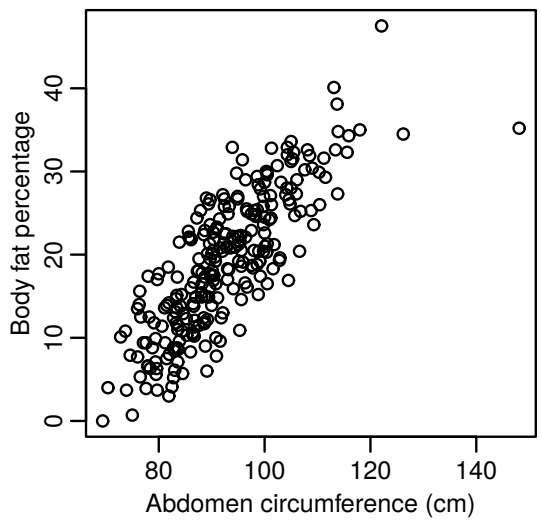
  where S_{kk} are constants automatically available from the linear model fit.
- To summarize: $\mathrm{CV}_{(K)}$ is a good estimate of the expected prediction error $\mathrm{Err}_{\widehat{f}}$, and the choice of K is a bias-variance trade off (large K: small bias, large variance; small K: large bias, small variance).

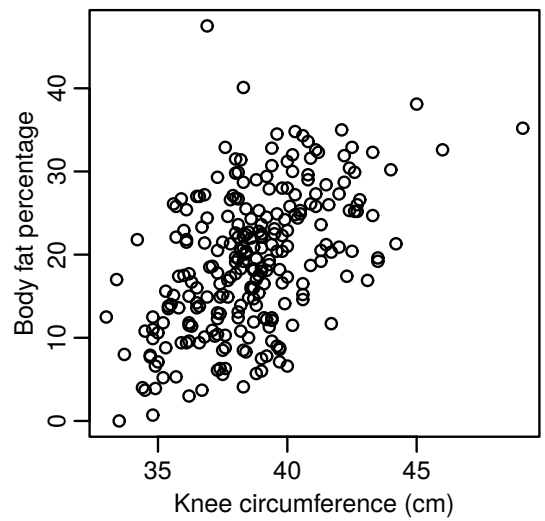
Medical application: body fat

Data set `bodyfat` (library `mfp`). The data set contains body fat estimates (`siri`) for 252 men with measurements of different body attributes, in total $p = 13$ predictors. The first 10 measurements and a plot of the responses versus some of the predictors:

siri	age	weight	height	neck	chest	abdo	hip	thigh	knee	ankle	biceps	forearm	wrist
12.3	23	154.25	67.75	36.2	93.1	85.2	94.5	59.0	37.3	21.9	32.0	27.4	17.1
6.1	22	173.25	72.25	38.5	93.6	83.0	98.7	58.7	37.3	23.4	30.5	28.9	18.2
25.3	22	154.00	66.25	34.0	95.8	87.9	99.2	59.6	38.9	24.0	28.8	25.2	16.6
10.4	26	184.75	72.25	37.4	101.8	86.4	101.2	60.1	37.3	22.8	32.4	29.4	18.2
28.7	24	184.25	71.25	34.4	97.3	100.0	101.9	63.2	42.2	24.0	32.2	27.7	17.7
20.9	24	210.25	74.75	39.0	104.5	94.4	107.8	66.0	42.0	25.6	35.7	30.6	18.8
19.2	26	181.00	69.75	36.4	105.1	90.7	100.3	58.4	38.3	22.9	31.9	27.8	17.7
12.4	25	176.00	72.50	37.8	99.6	88.5	97.1	60.0	39.4	23.2	30.5	29.0	18.8
4.1	25	191.00	74.00	38.1	100.9	82.5	99.9	62.9	38.3	23.8	35.9	31.1	18.2
11.7	23	198.25	73.50	42.1	99.6	88.6	104.1	63.1	41.7	25.0	35.6	30.0	19.2







Medical application: body fat

Data set `bodyfat` (library `mfp`). The data set contains body fat estimates (`siri`) for 252 men with measurements of different body attributes, in total $p = 13$ predictors.

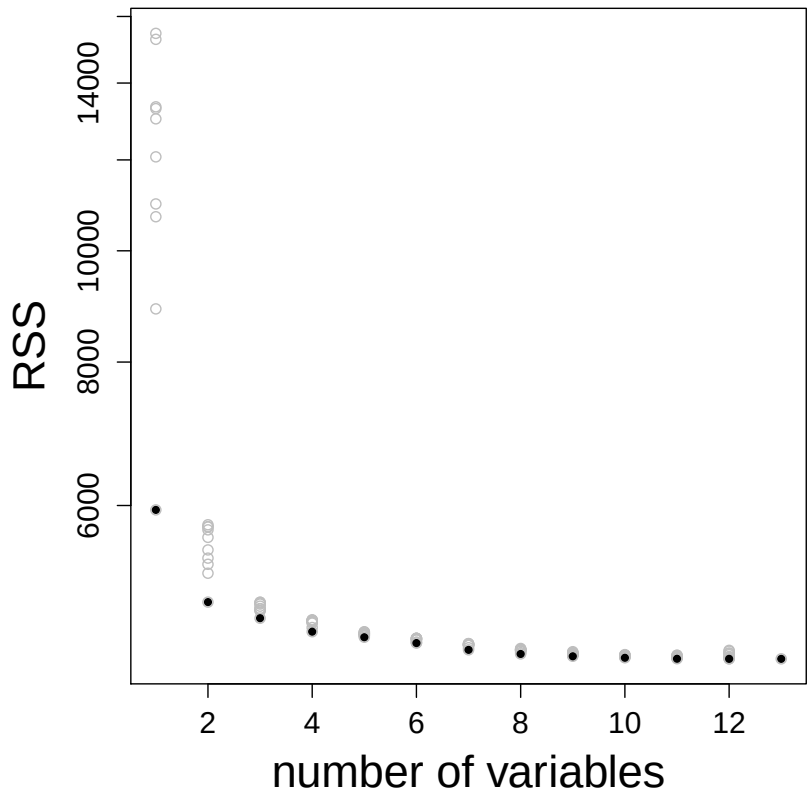


Figure: Best model fit (multiple linear regression) for a fixed number of predictors in terms of RSS; search is over all subsets with this number of predictors.

Medical application: body fat

Data set `bodyfat` (library `mfp`). The data set contains body fat estimates (`siri`) for 252 men with measurements of different body attributes, in total $p = 13$ predictors.

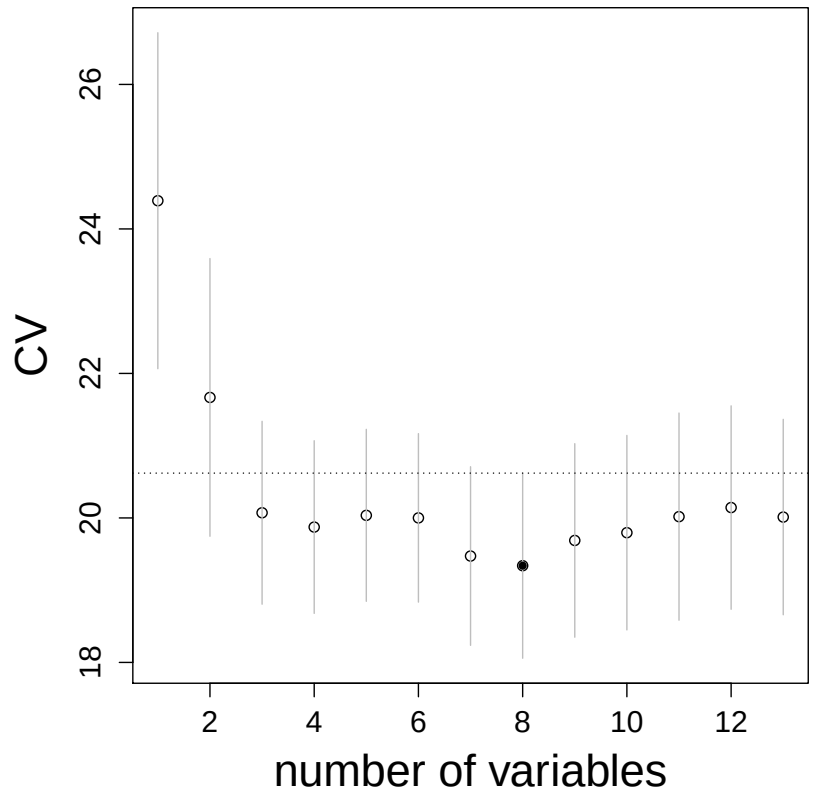


Figure: Results of 10 fold cross-validation for best-subset selection on the body fat dataset. The folds were randomly selected. The grey vertical lines are one-standard-error bars.

Medical application: body fat

Data set `bodyfat` (library `mfp`). The data set contains body fat estimates (`siri`) for 252 men with measurements of different body attributes, in total $p = 13$ predictors.

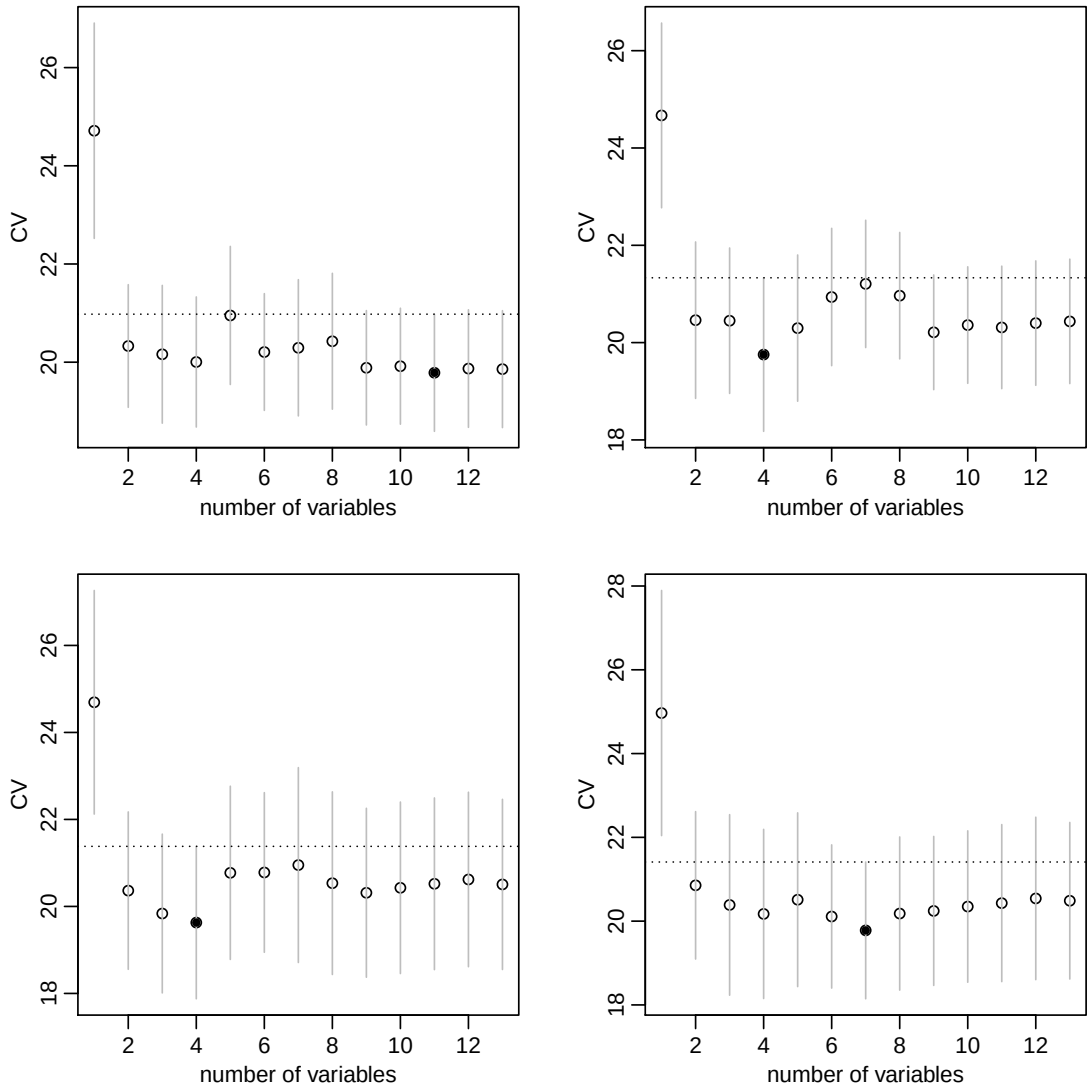


Figure: Results for 4 repetitions of 10-fold CV using different folds (randomly selected).

Medical application: body fat

- The previous slides show that the results of CV depend on the randomness of the folds.
- Model selection based on the so-called one-standard-error rule should be more stable: Choose the most parsimonious model whose error is no more than one standard error above the error of the best model.
- To investigate this, we can repeat 10-fold CV 100 times and identify the number of variables k minimizing CV and the optimal values of k chosen using the one-standard-error rule. The table below gives the number of times $\widehat{k} = k$ was selected by each method.

k	1	2	3	4	5	6	7	8	9	10	11	12	13
arg min of CV	0	5	8	11	0	1	11	20	11	12	18	3	0
one-sd-err rule	0	83	15	2	0	0	0	0	0	0	0	0	0

Medical application: body fat

- The previous slides show that the results of CV depend on the randomness of the folds.
- Model selection based on the so-called one-standard-error rule should be more stable: Choose the most parsimonious model whose error is no more than one standard error above the error of the best model.
- To investigate this, we can repeat 10-fold CV 100 times and identify the number of variables k minimizing CV and the optimal values of k chosen using the one-standard-error rule. The table below gives the number of times $\widehat{k} = k$ was selected by each method.

k	1	2	3	4	5	6	7	8	9	10	11	12	13
arg min of CV	0	5	8	11	0	1	11	20	11	12	18	3	0
one-sd-err rule	0	83	15	2	0	0	0	0	0	0	0	0	0

$\Rightarrow$ Use one-standard-error rule!

Medical application: body fat

To conclude on the body fat example:

- Using cross-validation, we may choose $\widehat{k} = 3$.
- We then estimate our final model $\widehat{f}_{\widehat{k}} = \widehat{f}_3$ on all 252 observations: this means we identify the best-subset model with 3 predictors and take this as our final model.
- Our final model uses the features `weight`, `abdomen` and `wrist`.
- We easily get the parameter estimates for this model: simply fit the multiple linear model with the features `weight`, `abdomen` and `wrist`.