UPPSALA UNIVERSITET

Bayesian Statistics and Data Analysis
Lecture 8b

Måns Magnusson
Department of Statistics, Uppsala University
Thanks to Aki Vehtari, Aalto University

UPPSALA UNIVERSITET

Section 1

Model assessment and selection

Predictive performance

- Modeling complex phenomena with models that are simplified
  All models are wrong... but some are useful.

UPPSALA UNIVERSITET

Predictive performance

- Modeling complex phenomena with models that are simplified
  All models are wrong... but some are useful.
- True predictive performance is found out by using it to make predictions and comparing predictions to true observations
  - external validation

Predictive performance

- Modeling complex phenomena with models that are simplified
  All models are wrong... but some are useful.
- True predictive performance is found out by using it to make predictions and comparing predictions to true observations
  - external validation
- Expected predictive performance
  - approximates the external validation

Goal of model evaluation

- Model choice is a (model-)decision-theoretic problem

Goal of model evaluation

- Model choice is a (model-)decision-theoretic problem
- Evaluate the utility of a model M for new unseen data $\tilde{y}$:

$$U = \int u(\tilde{y}) p_{\text{true}}(\tilde{y}) d\tilde{y} ,$$

where $\tilde{y}$ is an unseen observation generated from the true data generating process $p_{\text{text}}(\tilde{y})$, and y are observed data and $u(\tilde{y})$ is a utility function.

Goal of model evaluation

- Model choice is a (model-)decision-theoretic problem
- Evaluate the utility of a model M for new unseen data $\tilde{y}$:

$$U = \int u(\tilde{y}) p_{\text{true}}(\tilde{y}) d\tilde{y} \,,$$

  where $\tilde{y}$ is an unseen observation generated from the true data generating process $p_{\text{text}}(\tilde{y})$, and y are observed data and $u(\tilde{y})$ is a utility function.
- The expectation is with respect to p_{true} (f in BDA3)

Goal of model evaluation

- Model choice is a (model-)decision-theoretic problem
- Evaluate the utility of a model M for new unseen data $\tilde{y}$:

$$U = \int u(\tilde{y}) p_{\text{true}}(\tilde{y}) d\tilde{y} \,,$$

  where $\tilde{y}$ is an unseen observation generated from the true data generating process $p_{\text{text}}(\tilde{y})$, and y are observed data and $u(\tilde{y})$ is a utility function.
- The expectation is with respect to p_{true} (f in BDA3)
- Choose the model function to maximize our utility

Model choice utility

- Application specific utility/cost functions are important
  - eg. money, life years, quality adjusted life years, etc.

Model choice utility

- Application specific utility/cost functions are important
  - eg. money, life years, quality adjusted life years, etc.
- General utility: overall in the goodness of the predictive distribution
  - we don't know (yet) the application specific utility then good information theoretically justified choice is log-score for model M

$$\log p_M(y^{\text{rep}}|y)$$

UPPSALA UNIVERSITET

Model choice utility

- Application specific utility/cost functions are important
  - eg. money, life years, quality adjusted life years, etc.
- General utility: overall in the goodness of the predictive distribution
  - we don't know (yet) the application specific utility then good information theoretically justified choice is log-score for model M

$$\log p_M(y^{\text{rep}}|y)$$

- We want the "best" model to explain the data

Subsection 1

Measures of predictive accuracy

UPPSALA UNIVERSITET

(Bayesian) Points Prediction Accuracy

- Point residuals

$$e_i = y_i - E(\tilde{y}_i|y)\,,$$

where

$$E(\tilde{y}|y) = \int \tilde{y}_i p(\tilde{y}_i|y) d\tilde{y}\,,$$

i.e. the expected predicted value

(Bayesian) Points Prediction Accuracy

- Point residuals

$$e_i = y_i - E(\tilde{y}_i|y),$$

where

$$E(\tilde{y}|y) = \int \tilde{y}_i p(\tilde{y}_i|y) d\tilde{y},$$

i.e. the expected predicted value

- Mean squared (prediction) error (MSE)

$$\mathrm{MSE} = \frac{1}{n}\sum_i^n [y_i - E(\tilde{y}_i|y)]^2 .$$

Probabilistic predictions

- The log score (a local and proper scoring rule)*

$$\log p(y|\theta)$$

Probabilistic predictions

- The log score (a local and proper scoring rule)*

$$\log p(y|\theta)$$

- The log predictive density (lpd)

$$\begin{aligned}\text{lpd} &= \log p(y|y) \\ &= \log \int p(y|\theta)p(\theta|y)d\theta\end{aligned}$$

Probabilistic predictions

UPPSALA UNIVERSITET

- The lpd is usually approximated with the log point predictive density (lppd or just lpd)

$$
\begin{aligned}
\text{lppd} &= \sum_{i}^{n} \log \, p(y_i | y) \\
&\approx \log p(y | y)
\end{aligned}
$$

UPPSALA UNIVERSITET

Probabilistic predictions

- The lpd is usually approximated with the log point predictive density (lppd or just lpd)

$$\text{lppd} = \sum_{i}^{n} \log\, p(y_i|y)$$
$$\approx \log p(y|y)$$

- Estimation using MCMC

$$\text{lppd} = \sum_{i}^{n} \log\left(\frac{1}{S}\sum_{s}^{S} p(y_i|\theta_s))\right)$$

Subsection 2

Model selection

UPPSALA UNIVERSITET

Model selection

- Evaluate how model M generalizes to unseen data $\tilde{y}$ (the *expected log predictive density*):

$$\text{elpd}_M = \int \log p_M(\tilde{y}|y) p_{\text{true}}(\tilde{y}) d\tilde{y} \,,$$

  where $\tilde{y}$ is an unseen observation generated from the true data generating process $p_{\text{true}}(\tilde{y})$, and y are observed data.
- $\log p_M(\tilde{y}|y)$ is the log score (the utility of the model)

Model selection

- Evaluate how model M generalizes to unseen data $\tilde{y}$ (the *expected log predictive density*):

$$\mathrm{elpd}_M = \int \log p_M(\tilde{y}|y) p_{\mathrm{true}}(\tilde{y}) d\tilde{y}\,,$$

  where $\tilde{y}$ is an unseen observation generated from the true data generating process $p_{\mathrm{true}}(\tilde{y})$, and y are observed data.
- $\log p_M(\tilde{y}|y)$ is the log score (the utility of the model)
- The expectation is with respect to p_{true}

Model selection

- Evaluate how model M generalizes to unseen data $\tilde{y}$ (the *expected log predictive density*):

$$\mathrm{elpd}_M = \int \log p_M(\tilde{y}|y) p_{\mathrm{true}}(\tilde{y}) d\tilde{y} \,,$$

  where $\tilde{y}$ is an unseen observation generated from the true data generating process $p_{\mathrm{true}}(\tilde{y})$, and y are observed data.
- $\log p_M(\tilde{y}|y)$ is the log score (the utility of the model)
- The expectation is with respect to p_{true}
- p_{true} is (almost always) unknown

UPPSALA UNIVERSITET

Model selection

- Evaluate how model M generalizes to unseen data $\tilde{y}$ (the *expected log predictive density*):

$$\text{elpd}_M = \int \log p_M(\tilde{y}|y) p_{\text{true}}(\tilde{y}) d\tilde{y}\,,$$

  where $\tilde{y}$ is an unseen observation generated from the true data generating process $p_{\text{true}}(\tilde{y})$, and y are observed data.
- $\log p_M(\tilde{y}|y)$ is the log score (the utility of the model)
- The expectation is with respect to p_{true}
- p_{true} is (almost always) unknown
- The utility function is the log scoring rule.

UPPSALA UNIVERSITET

Section 2

Cross-validation

Leave-one-out cross-validation (LOO-CV)

- Evaluate how model M *generalizes* to unseen data (the *expected log predictive density*):

$$\mathrm{elpd}_M = \int \log p_M(\tilde{y}_i|y)p_{\mathrm{true}}(\tilde{y}_i)d\tilde{y}_i \,,$$

where $\tilde{y}_i$ is an unseen observation generated from the true data generating process $p_{\mathrm{true}}(\tilde{y}_i)$, and y are observed data.

Leave-one-out cross-validation (LOO-CV)

- Evaluate how model M *generalizes* to unseen data (the *expected log predictive density*):

$$\mathrm{elpd}_M = \int \log p_M(\tilde{y}_i | y) p_{\mathrm{true}}(\tilde{y}_i) d\tilde{y}_i \,,$$

  where $\tilde{y}_i$ is an unseen observation generated from the true data generating process $p_{\mathrm{true}}(\tilde{y}_i)$, and y are observed data.
- Can we approximate $p_{\mathrm{true}}(\tilde{y}_i)$?

Leave-one-out cross-validation (LOO-CV)

- Approximate $p_{\text{true}}(\tilde{y}_i)$ with data y

Leave-one-out cross-validation (LOO-CV)

- Approximate $p_{\text{true}}(\tilde{y}_i)$ with data y
- Hold out observation i and try to predict y_i based on $\mathbf{y}_{-i}$
- Estimation of elpd_M using leave-one-out cross-validation

$$\begin{aligned}
\text{elpd}_{\text{loo}} &= \sum_{i=1}^{n} \log p_M(y_i|\mathbf{y}_{-i}) \\
&= \sum_{i=1}^{n} \log \int p_M(y_i|\theta)p(\theta|\mathbf{y}_{-i})d\theta
\end{aligned}$$

Leave-one-out cross-validation (LOO-CV)

- Approximate $p_{\text{true}}(\tilde{y}_i)$ with data y
- Hold out observation i and try to predict y_i based on $\mathbf{y}_{-i}$
- Estimation of elpd_M using leave-one-out cross-validation

$$\begin{aligned}\text{elpd}_{\text{loo}} &= \sum_{i=1}^{n} \log p_M(y_i|\mathbf{y}_{-i}) \\ &= \sum_{i=1}^{n} \log \int p_M(y_i|\theta)p(\theta|\mathbf{y}_{-i})d\theta\end{aligned}$$

- Analogy: Monte Carlo approximation using our data
- Similar to jack-knife resampling

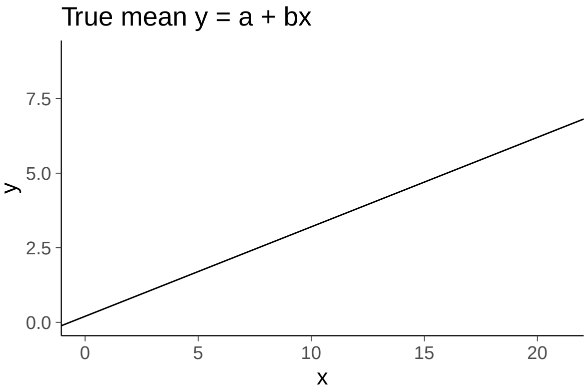
Leave-one-out cross-validation (LOO-CV)

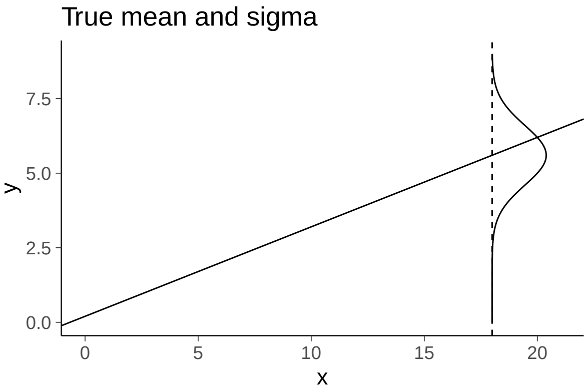
- Approximate $p_{\text{true}}(\tilde{y}_i)$ with data y
- Hold out observation i and try to predict y_i based on $\mathbf{y}_{-i}$
- Estimation of elpd_M using leave-one-out cross-validation

$$\begin{aligned}\text{elpd}_{\text{loo}} &= \sum_{i=1}^{n} \log p_M(y_i|\mathbf{y}_{-i}) \\ &= \sum_{i=1}^{n} \log \int p_M(y_i|\theta)p(\theta|\mathbf{y}_{-i})d\theta\end{aligned}$$

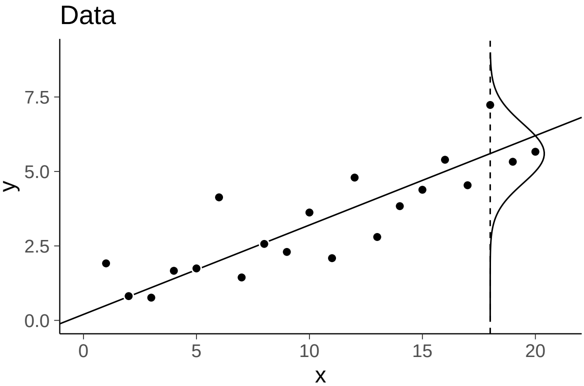
- Analogy: Monte Carlo approximation using our data
- Similar to jack-knife resampling
- The elpd, lpd and efficient number of parameters (p_{loo})

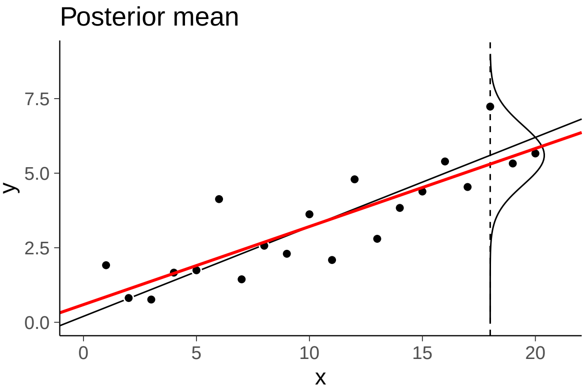
$$\text{elpd}_{\text{loo}} = \text{lpd} + p_{\text{loo}}$$


True mean y = a + bx
7.5
5.0
2.5
0.0
y
0
5
10
15
20
x

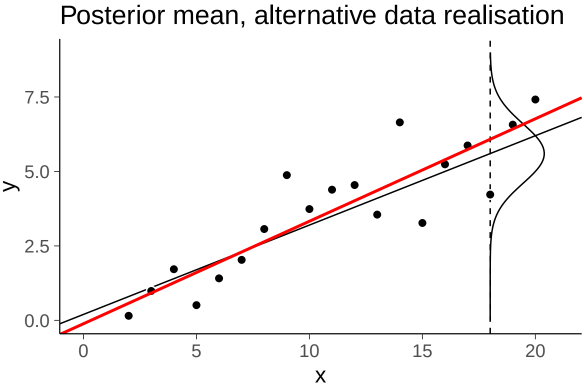

True mean and sigma
7.5
5.0
2.5
0.0
y
0
5
10
15
20
x

UPPSALA UNIVERSITET

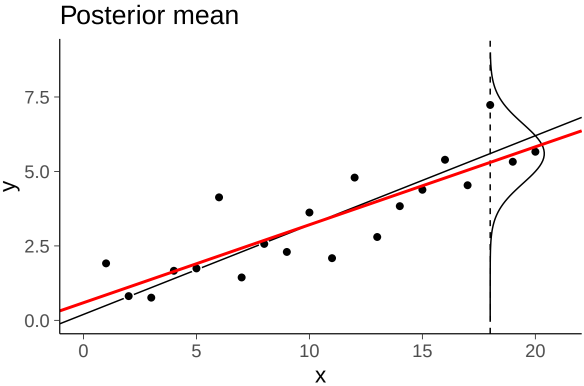



Posterior mean
7.5
5.0
2.5
0.0
y
0
5
10
15
20
x

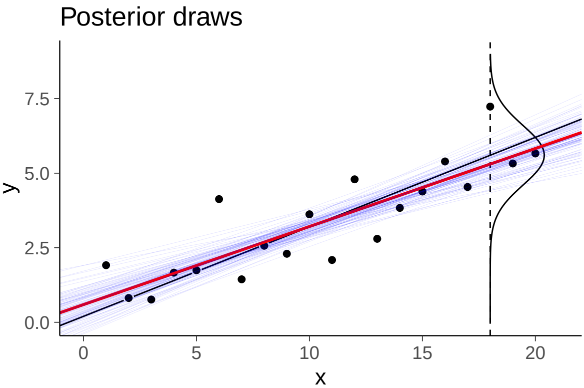
UPPSALA UNIVERSITET



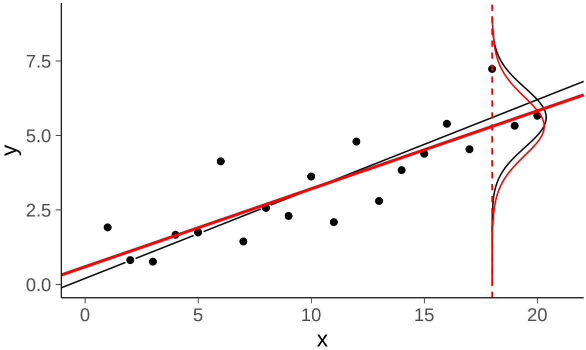
UPPSALA UNIVERSITET



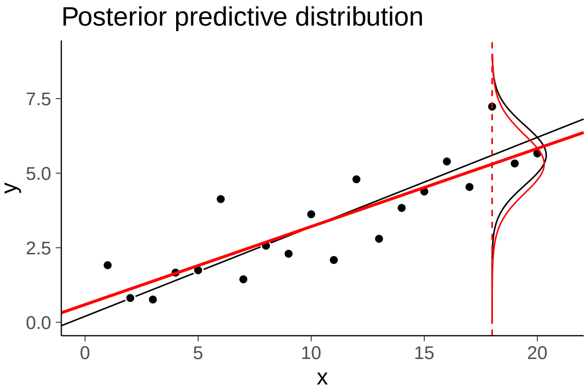
UPPSALA UNIVERSITET



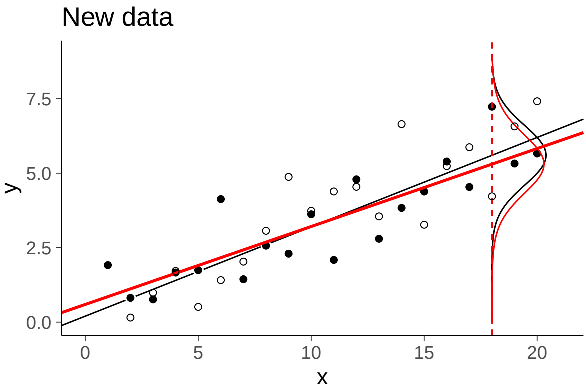
Posterior predictive distribution

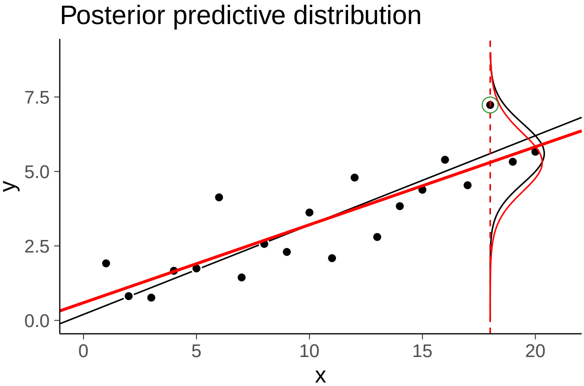


UPPSALA UNIVERSITET

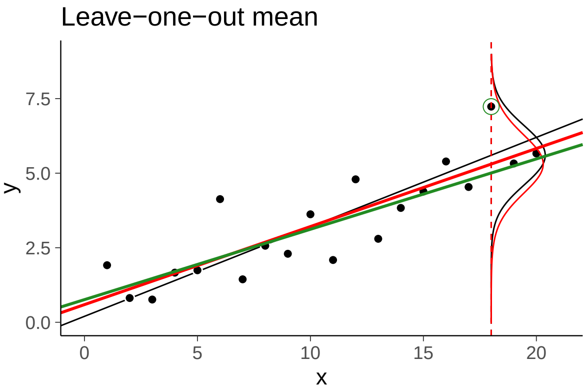


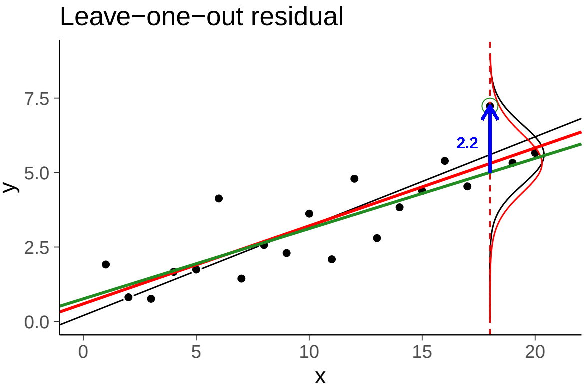
$$p(\tilde{y}|\tilde{x} = 18, x, y) = \int p(\tilde{y}|\tilde{x} = 18, \theta)p(\theta|x, y)d\theta$$


New data
y
7.5
5.0
2.5
0.0
0
5
10
15
20
x

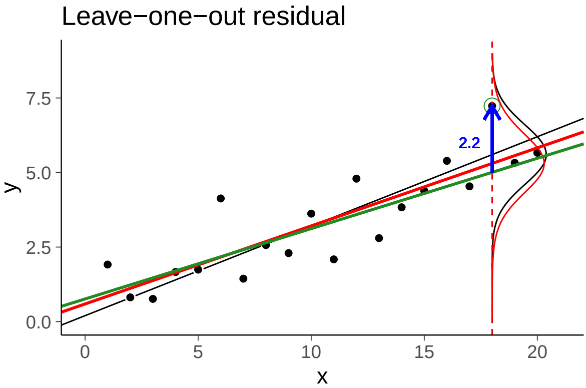

Posterior predictive distribution
7.5
5.0
2.5
0.0
y
0
5
10
15
20
x

UPPSALA UNIVERSITET

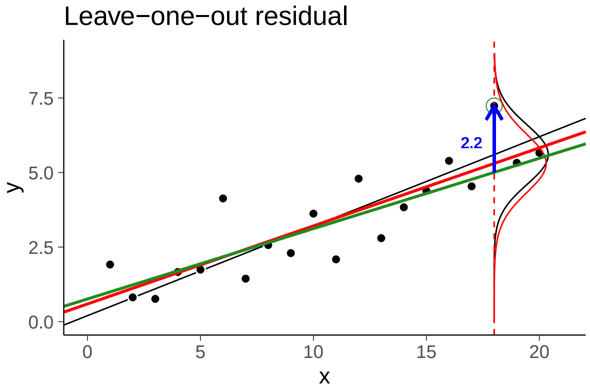



Leave−one−out residual
7.5
5.0
2.5
0.0
y
0
5
10
15
20
x
2.2

UPPSALA UNIVERSITET

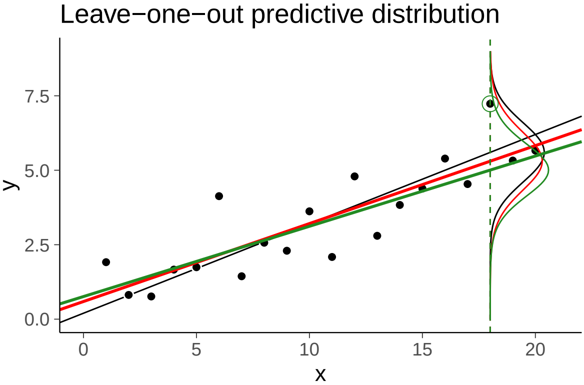


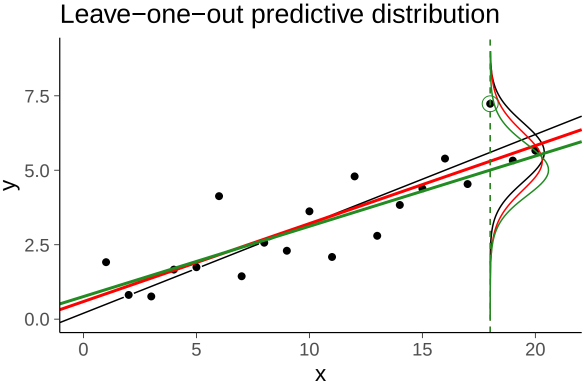
$$y_{18} - E[p(\tilde{y}|\tilde{x} = 18, x_{-18}, y_{-18})]$$



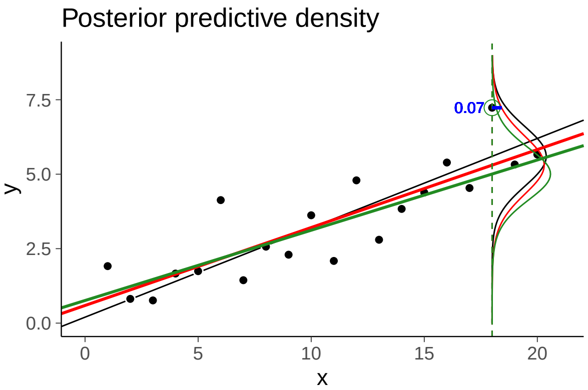
$y_{18} - E[p(\tilde{y}|\tilde{x} = 18, x_{-18}, y_{-18})]$

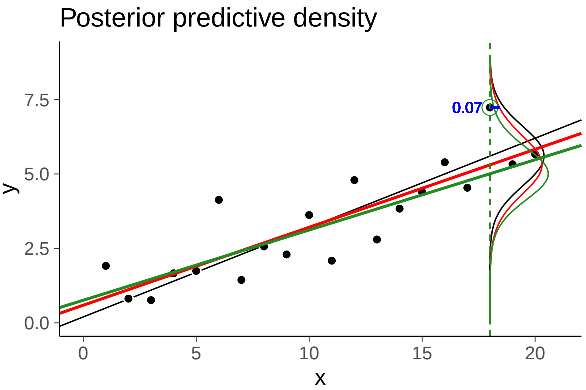
Can be use to compute, e.g., RMSE, R^2, 90% error


Leave−one−out predictive distribution
y
x
0.0
2.5
5.0
7.5
0
5
10
15
20



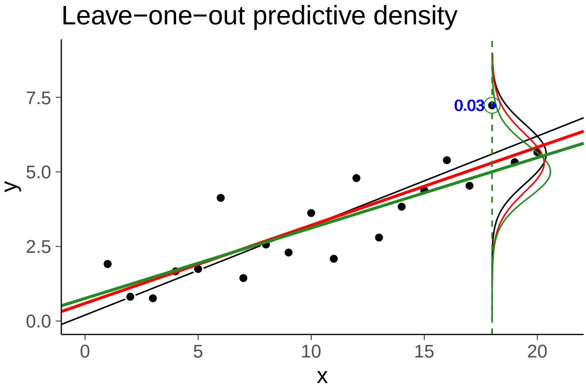
$$p(\tilde{y}|\tilde{x}=18, x_{-18}, y_{-18}) = \int p(\tilde{y}|\tilde{x}=18,\theta)p(\theta|x_{-18}, y_{-18})d\theta$$


Posterior predictive density
0.07
y
x
7.5
5.0
2.5
0.0
0
5
10
15
20



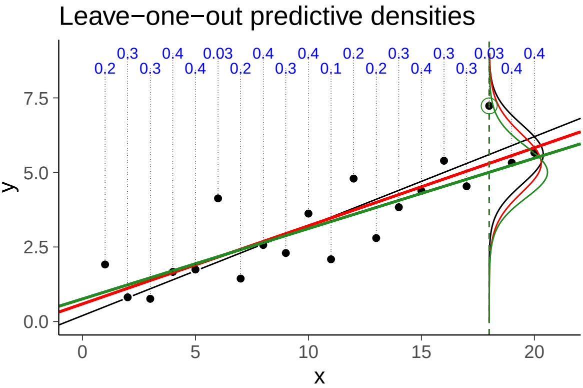
$$p(\tilde{y} = y_{18} | \tilde{x} = 18, x, y) \approx 0.07$$

UPPSALA UNIVERSITET

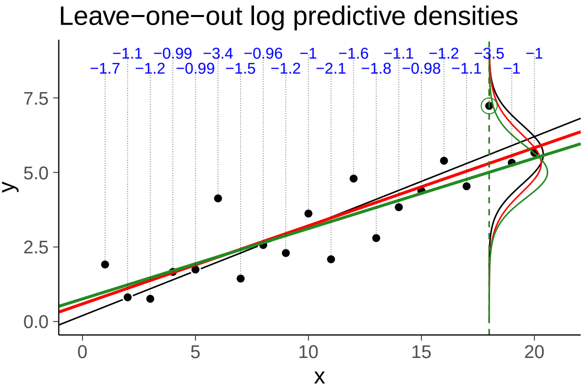


$$p(\tilde{y} = y_{18} | \tilde{x} = 18, x, y) \approx 0.07$$

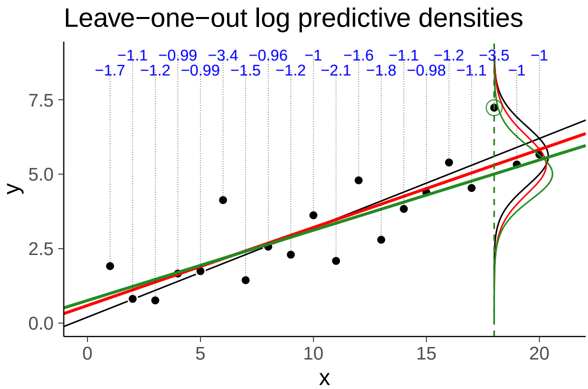
$$p(\tilde{y} = y_{18} | \tilde{x} = 18, x_{-18}, y_{-18}) \approx 0.03$$



$$p(y_i|x_i, x_{-i}, y_{-i}), \quad i = 1, \ldots, 20$$

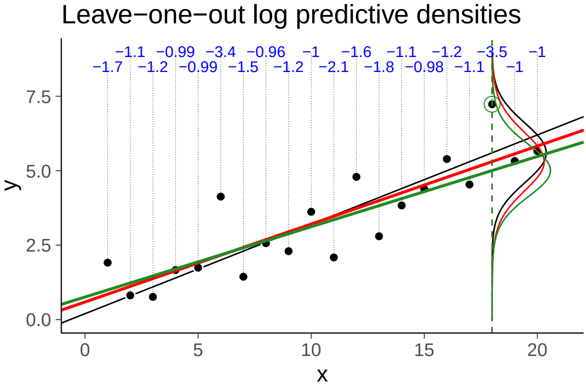


$$\log p(y_i | x_i, x_{-i}, y_{-i}), \quad i = 1, \ldots, 20$$

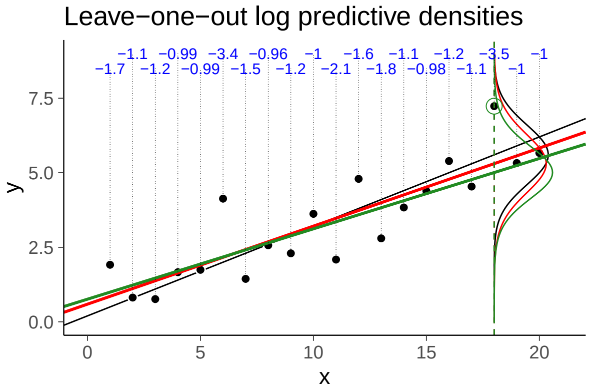


$$\sum_{i=1}^{20} \log p(y_i|x_i, x_{-i}, y_{-i}) \approx -29.5$$

UPPSALA UNIVERSITET



$$\text{elpd\_loo} = \sum_{i=1}^{20} \log p(y_i | x_i, x_{-i}, y_{-i}) \approx -29.5$$

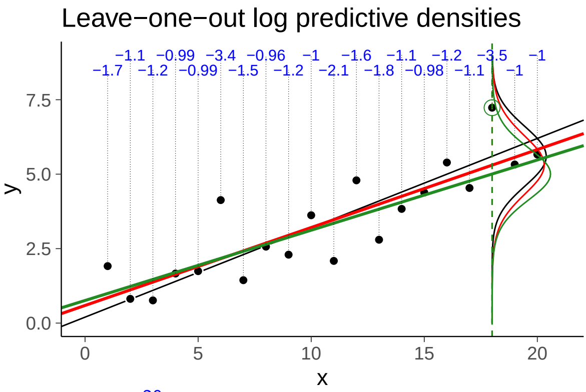


$\text{elpd\_loo} = \sum_{i=1}^{20} \log p(y_i|x_i, x_{-i}, y_{-i}) \approx -29.5$

unbiased estimate of log posterior pred. density for new data

UPPSALA UNIVERSITET

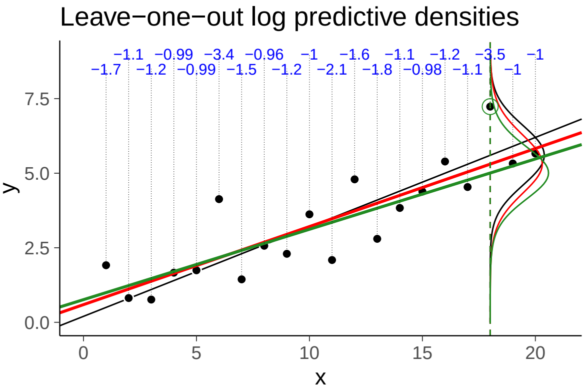
- Model assessment and selection
  - Measures of predictive accuracy
  - Model selection
- Cross-validation
  - When is LOO applicable
  - PSIS-LOO and `loo`
  - K-fold cross-validation
  - Comparison and selection
  - Additional reading
- Information criteria
- Model averaging
- Summary



$$\text{elpd\_loo} = \sum_{i=1}^{20} \log p(y_i|x_i, x_{-i}, y_{-i}) \approx -29.5$$

$$\text{lpd} = \sum_{i=1}^{20} \log p(y_i|x_i, x, y) \approx -26.8$$

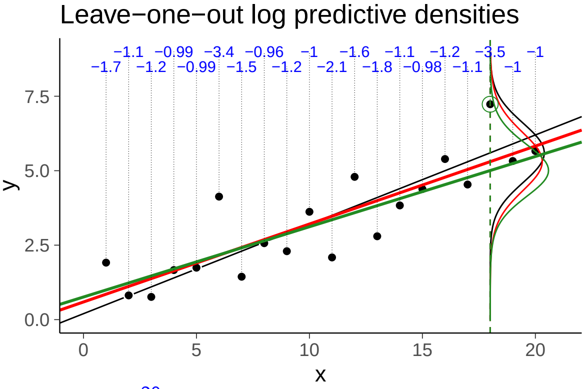
UPPSALA UNIVERSITET



$$\text{elpd\_loo} = \sum_{i=1}^{20} \log p(y_i|x_i, x_{-i}, y_{-i}) \approx -29.5$$

$$\text{lpd} = \sum_{i=1}^{20} \log p(y_i|x_i, x, y) \approx -26.8$$

$$\text{p\_loo} = \text{lpd} - \text{elpd\_loo} \approx 2.7$$



$$\text{elpd\_loo} = \sum_{i=1}^{20} \log p(y_i|x_i, x_{-i}, y_{-i}) \approx -29.5$$

$$\text{SE} = \text{sd}(\log p(y_i|x_i, x_{-i}, y_{-i})) \cdot \sqrt{20} \approx 3.3$$

Stan and `loo` package

UPPSALA UNIVERSITET


```
Computed from 4000 by 20 log-likelihood matrix

         Estimate  SE
elpd_loo    -29.5 3.3
p_loo         2.7 1.0
------
Monte Carlo SE of elpd_loo is 0.1.

Pareto k diagnostic values:
                         Count  Pct.    Min. n_eff
(-Inf, 0.5]   (good)     18     90.0%   899
 (0.5, 0.7]   (ok)        2     10.0%   459
   (0.7, 1]   (bad)       0      0.0%   <NA>
   (1, Inf)   (very bad)  0      0.0%   <NA>

All Pareto k estimates are ok (k < 0.7).
See help('pareto-k-diagnostic') for details.
```

Pro and cons with LOO-CV

+ Intuitive
+ Robust
+ Good theoretical properties

UPPSALA UNIVERSITET

Pro and cons with LOO-CV

\+ Intuitive

\+ Robust

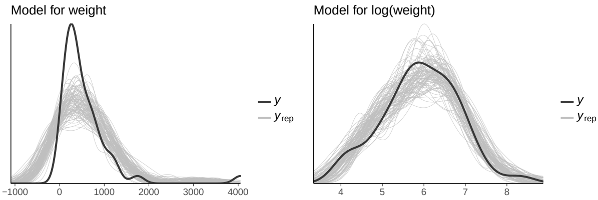
\+ Good theoretical properties

\- Can be costly (naive LOO-CV mean n posterior computations)

UPPSALA UNIVERSITET

Sometimes cross-validation is not needed

- Posterior predictive checking can be sufficient



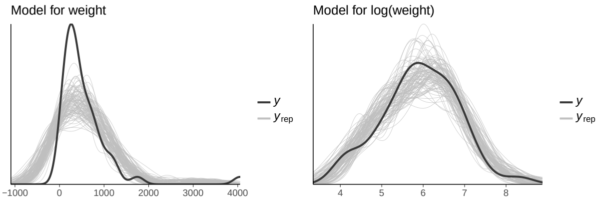
Predicting the yields of mesquite bushes.

Gelman, Hill & Vehtari (2020): Regression and Other Stories, Chapter 11.

UPPSALA UNIVERSITET

Sometimes cross-validation is not needed

- Posterior predictive checking can be sufficient



Predicting the yields of mesquite bushes.

Gelman, Hill & Vehtari (2020): Regression and Other Stories, Chapter 11.

- In nested case, often easier and more accurate to analyse posterior distribution of more complex model directly

UPPSALA UNIVERSITET

Subsection 1

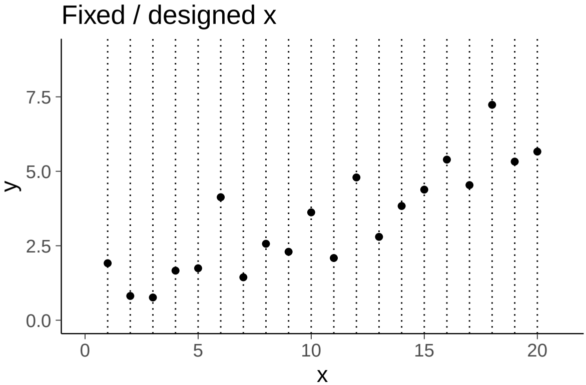
When is LOO applicable

Summary of data generating mechanisms and prediction tasks

1. You have to make some assumptions on data generating mechanism
2. Use the knowledge of the prediction task if available
3. Cross-validation can be used to analyse different parts, even if there is no clear prediction task

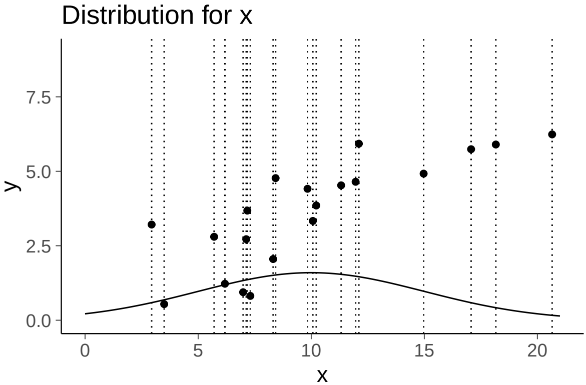
see Vehtari & Ojanen (2012) and andrewgelman.com/2018/08/03/loo-cross-validation-approaches-valid/

UPPSALA UNIVERSITET



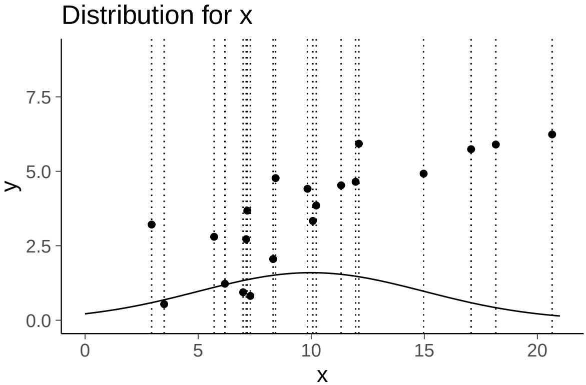
LOO is ok for fixed / designed x. SE is uncertainty about $y|x$.

see Vehtari & Ojanen (2012) and `andrewgelman.com/2018/08/03/loo-cross-validation-approaches-valid/`



LOO is ok for random x. SE is uncertainty about $y|x$ and x.

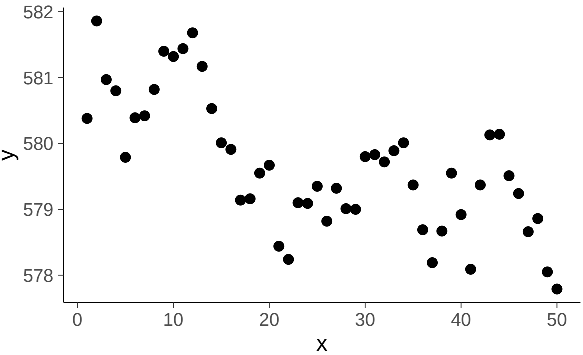
see Vehtari & Ojanen (2012) and andrewgelman.com/2018/08/03/loo-cross-validation-approaches-valid/

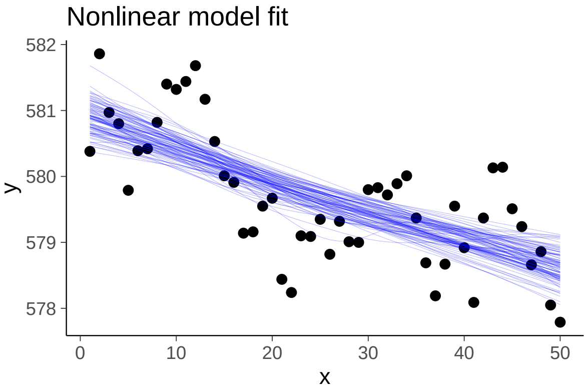


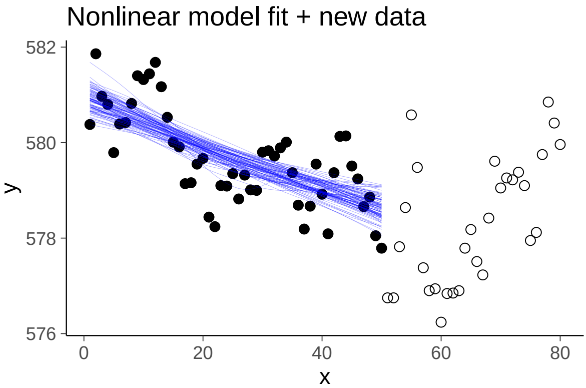
LOO is ok for random x. SE is uncertainty about $y|x$ and x.

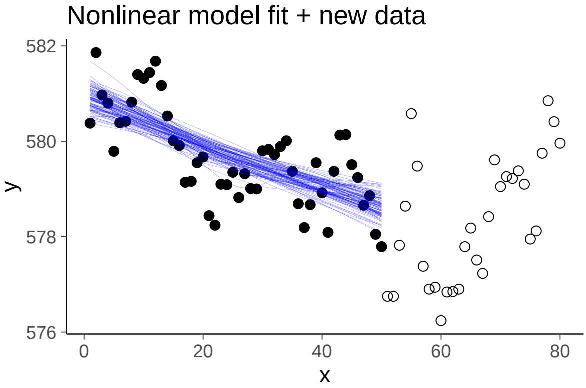
Covariate shift can be handled with importance weighting or modelling

see Vehtari & Ojanen (2012) and andrewgelman.com/2018/08/03/loo-cross-validation-approaches-valid/

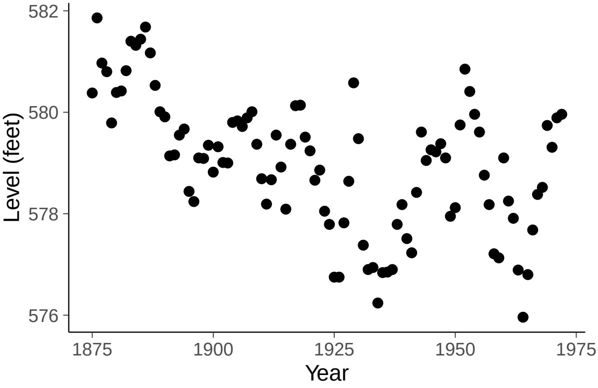

582
581
580
579
578
y
0
10
20
30
40
50
x


Nonlinear model fit
582
581
580
579
578
y
0
10
20
30
40
50
x

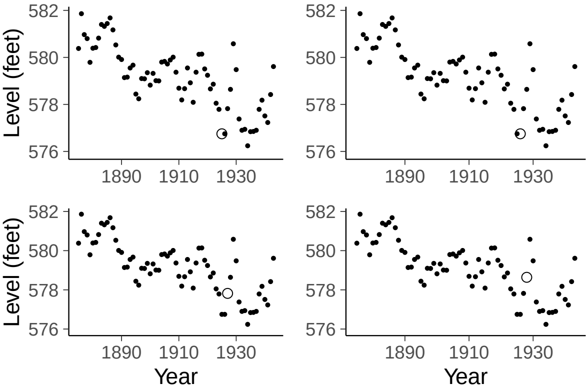

Nonlinear model fit + new data
582
580
578
576
y
0
20
40
60
80
x



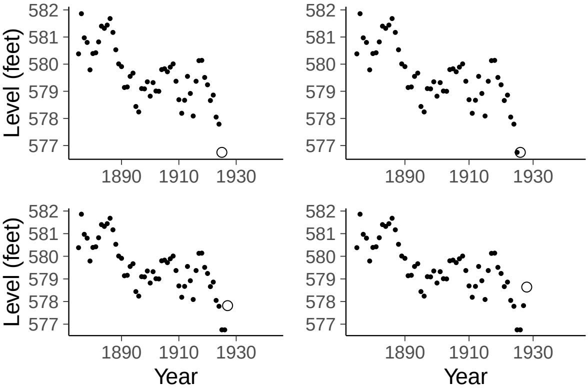
Extrapolation is more difficult



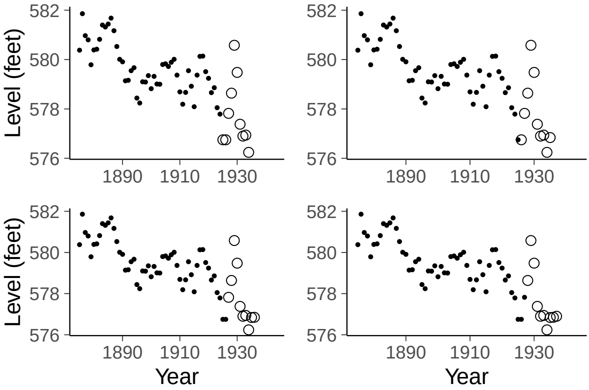
Can LOO or other cross-validation be used with time series?



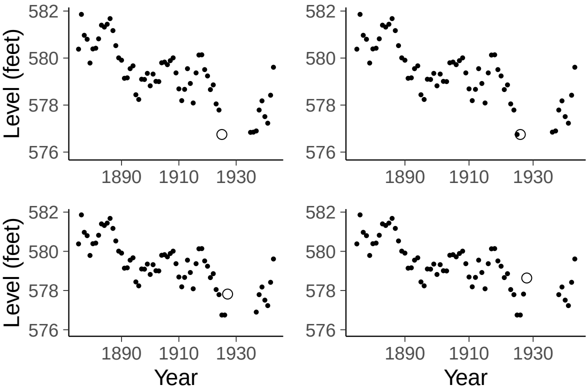
Leave-one-out cross-validation is ok for assessing conditional model



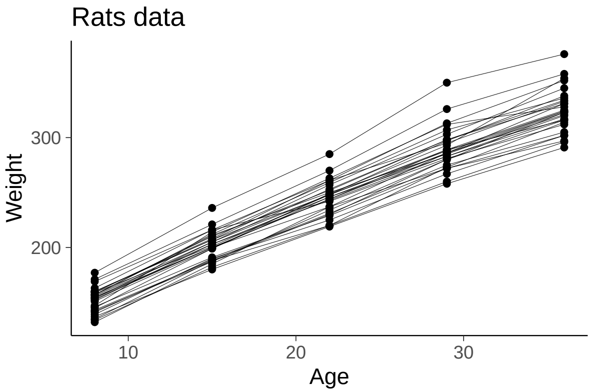
Leave-future-out cross-validation is better for predicting future



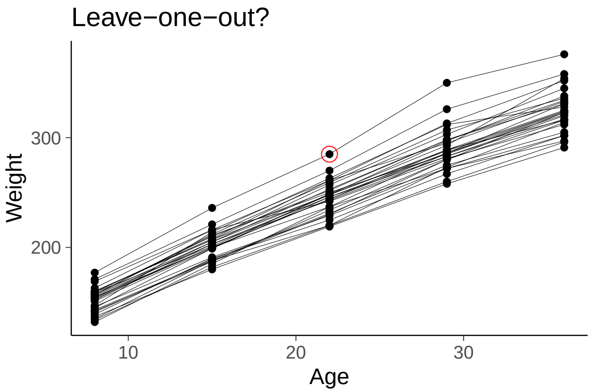
m-step-ahead cross-validation is better for predicting further future



m-step-ahead leave-a-block-out cross-validation

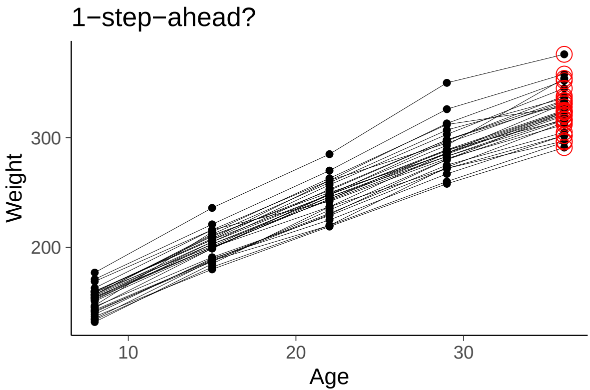


Can LOO or other cross-validation be used with hierarchical data?

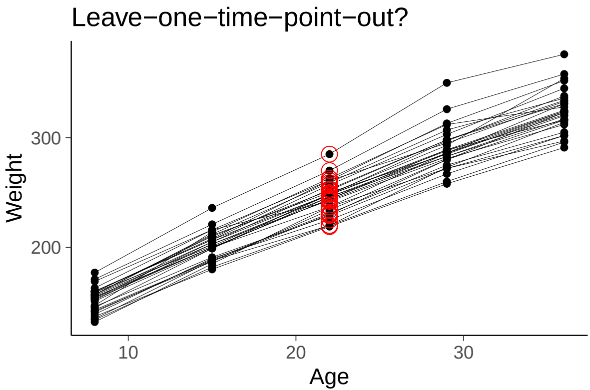


Yes!

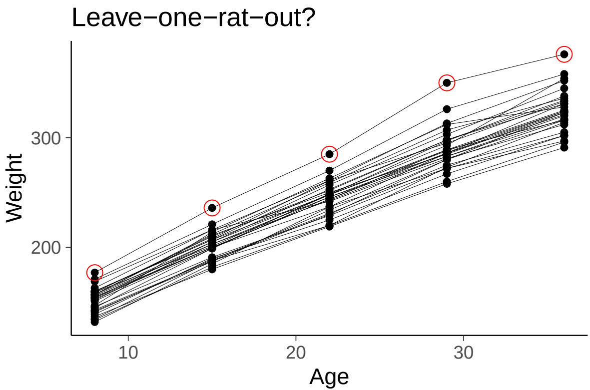
UPPSALA UNIVERSITET



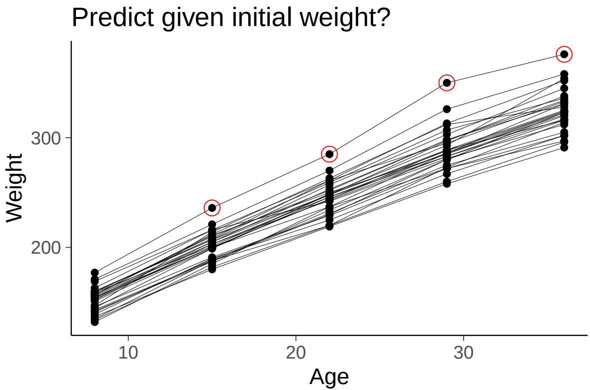
Yes!



Yes!



Yes!



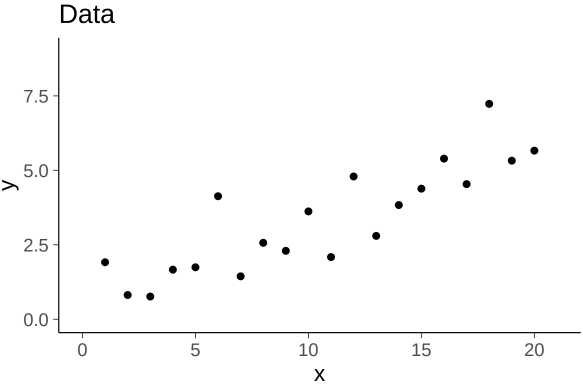
Yes!

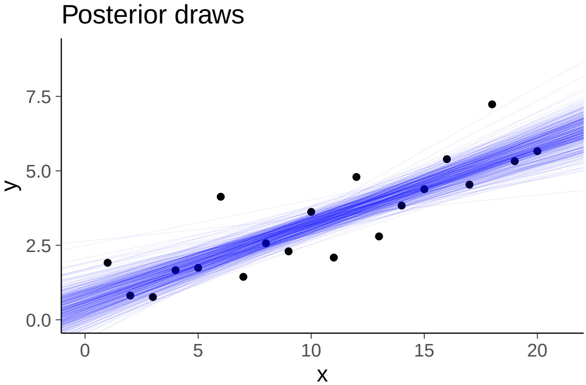
Fast cross-validation

1. Pareto smoothed importance sampling LOO (PSIS-LOO)
2. K-fold cross-validation

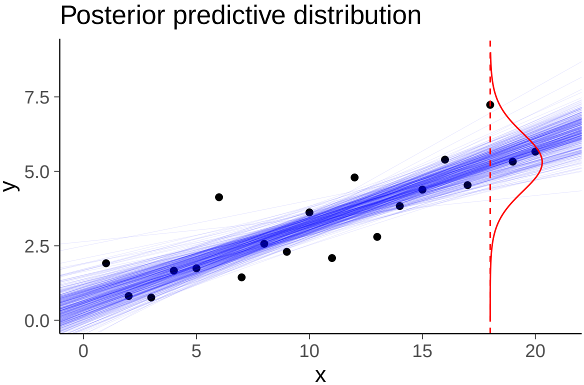
see Vehtari, Gelman & Gabry (2017a) and `mc-stan.org/loo/`

UPPSALA UNIVERSITET

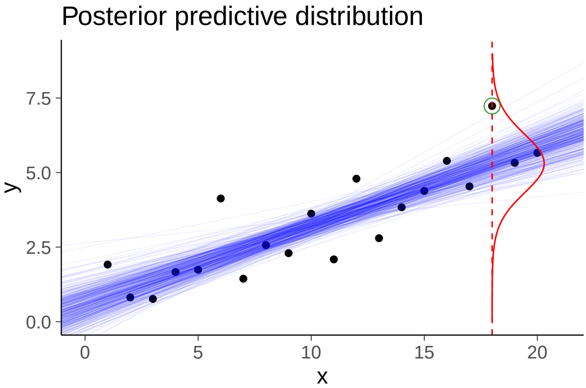




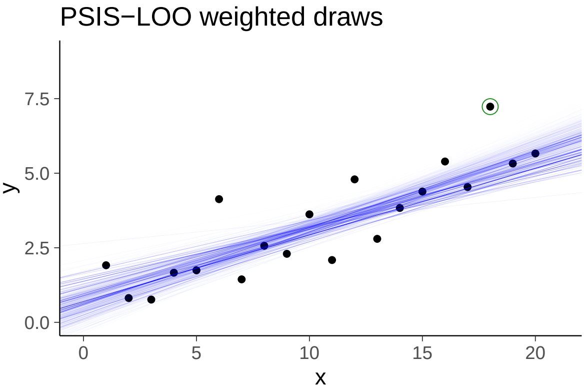
$$\theta^{(s)} \sim p(\theta|x, y)$$



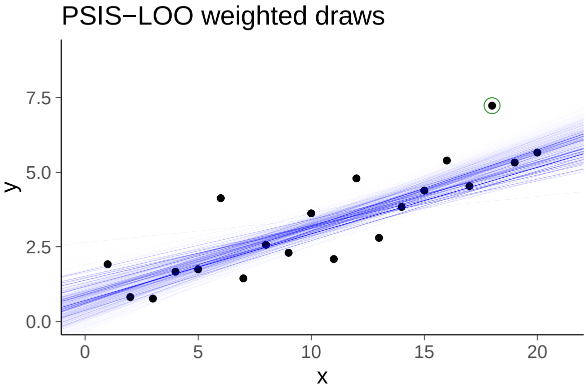
$$\theta^{(s)} \sim p(\theta|x,y), \quad p(\tilde{y}|\tilde{x},x,y) \approx \frac{1}{S}\sum_{s=1}^{S} p(\tilde{y}|\tilde{x},\theta^{(s)})$$



$$\theta^{(s)} \sim p(\theta|x,y), \quad p(\tilde{y}|\tilde{x},x,y) \approx \frac{1}{S}\sum_{s=1}^{S} p(\tilde{y}|\tilde{x},\theta^{(s)})$$

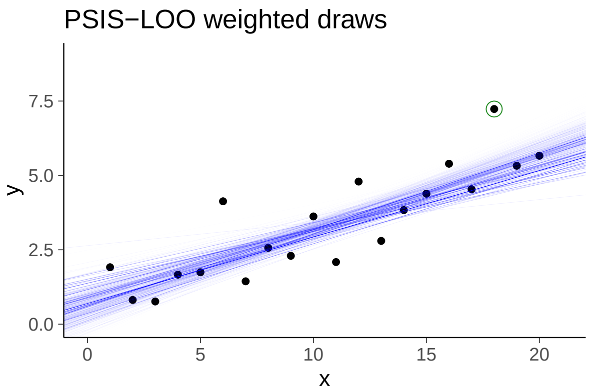


$$\theta^{(s)} \sim p(\theta|x,y)$$
$$r_i^{(s)} = p(\theta^{(s)}|x_{-i},y_{-i})/p(\theta^{(s)}|x,y)$$

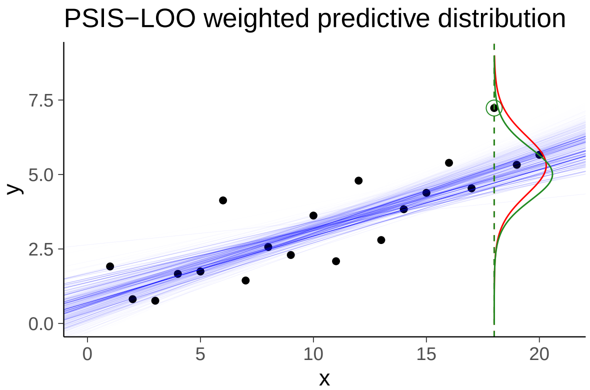


$$\theta^{(s)} \sim p(\theta|x, y)$$

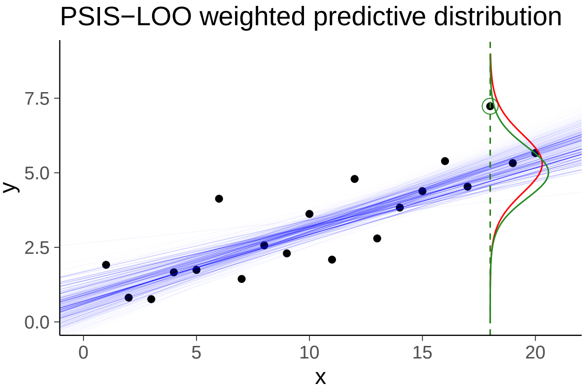
$$r_i^{(s)} = p(\theta^{(s)}|x_{-i}, y_{-i})/p(\theta^{(s)}|x, y) \propto 1/p(y_i|x_i, \theta^{(s)})$$



$$\theta^{(s)} \sim p(\theta|x, y)$$
$$r_i^{(s)} = p(\theta^{(s)}|x_{-i}, y_{-i})/p(\theta^{(s)}|x, y) \propto 1/p(y_i|x_i, \theta^{(s)})$$
$$\log(1/p(y_i|x_i, \theta^{(s)})) = -\text{log\_lik}[i]$$

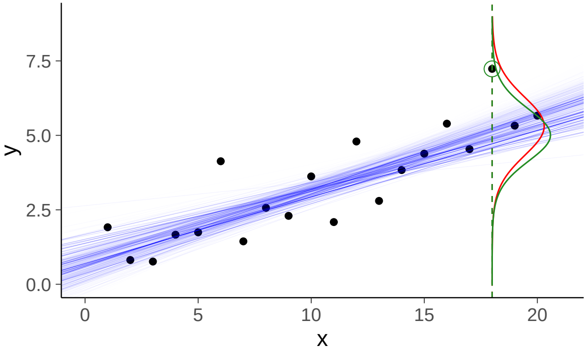


$$\theta^{(s)} \sim p(\theta|x, y)$$
$$r_i^{(s)} = p(\theta^{(s)}|x_{-i}, y_{-i})/p(\theta^{(s)}|x, y) \propto 1/p(y_i|x_i, \theta^{(s)})$$



$$\theta^{(s)} \sim p(\theta|x, y)$$
$$r_i^{(s)} = p(\theta^{(s)}|x_{-i}, y_{-i})/p(\theta^{(s)}|x, y) \propto 1/p(y_i|x_i, \theta^{(s)})$$
$$p(y_i|x_i, x_{-i}, y_{-i}) \approx \sum_{s=1}^{S}[w_i^{(s)} p(y_i|x_i, \theta^{(s)})]$$

PSIS−LOO weighted predictive distribution

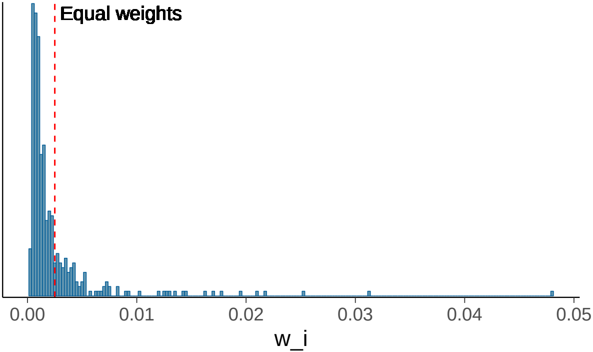


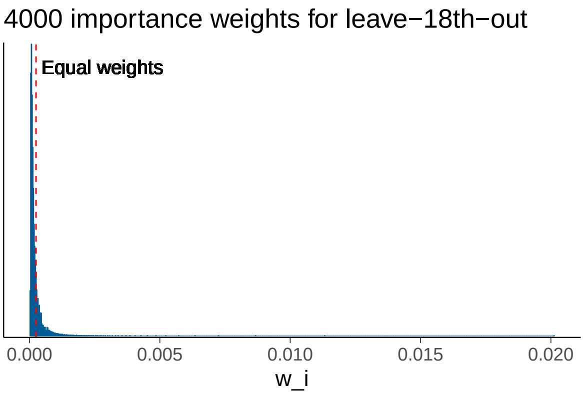
$$\theta^{(s)} \sim p(\theta|x, y)$$

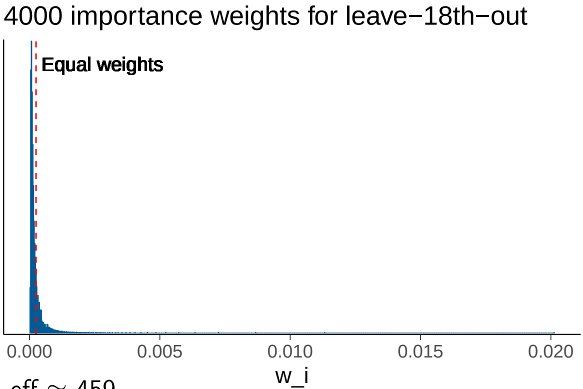
$$r_i^{(s)} = p(\theta^{(s)}|x_{-i}, y_{-i})/p(\theta^{(s)}|x, y) \propto 1/p(y_i|x_i, \theta^{(s)})$$

$p(y_i|x_i, x_{-i}, y_{-i}) \approx \sum_{s=1}^{S}[w_i^{(s)} p(y_i|x_i, \theta^{(s)})]$, where
$w \leftarrow \text{PSIS}(r)$

400 importance weights for leave−18th−out

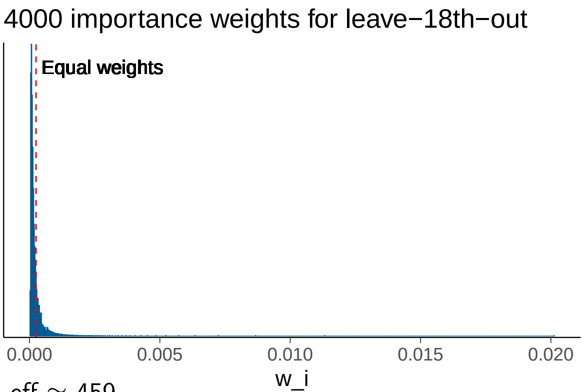



4000 importance weights for leave−18th−out
Equal weights
0.000
0.005
0.010
0.015
0.020
w_i



n_eff $\approx$ 459

see Vehtari, Gelman & Gabry (2017b)

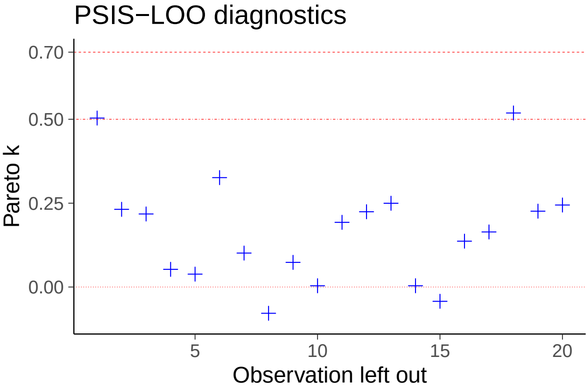


n_eff $\approx$ 459

Pareto $\hat{k} \approx 0.52$

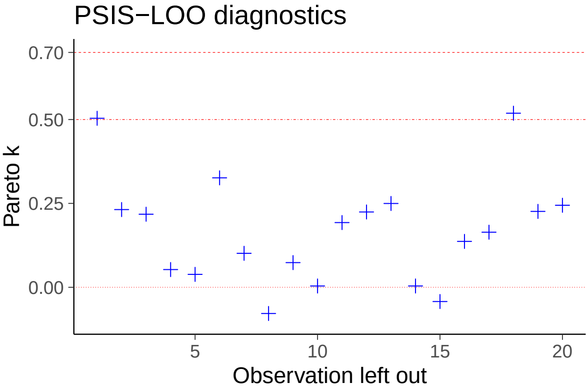
1. Pareto $\hat{k}$ estimates the tail shape which determines the convergence rate of PSIS. Less than 0.7 is ok.

see Vehtari, Gelman & Gabry (2017b)


PSIS−LOO diagnostics
0.70
0.50
0.25
0.00
Pareto k
5
10
15
20
Observation left out

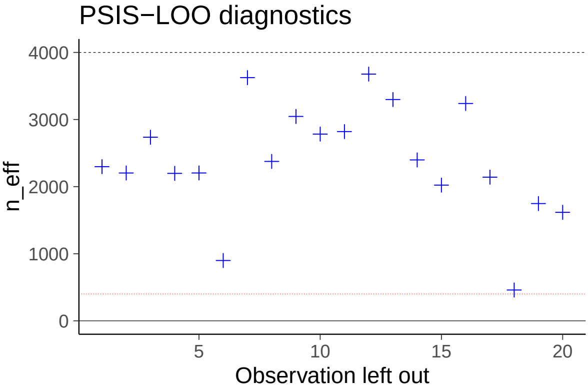
UPPSALA UNIVERSITET

- Model assessment and selection
  - Measures of predictive accuracy
  - Model selection
- Cross-validation
  - When is LOO applicable
  - PSIS-LOO and `loo`
  - K-fold cross-validation
  - Comparison and selection
  - Additional reading
- Information criteria
- Model averaging
- Summary




```
Pareto k diagnostic values:
                          Count  Pct.    Min. n_eff
(-Inf, 0.5]   (good)      18     90.0%   899
 (0.5, 0.7]   (ok)         2     10.0%   459
   (0.7, 1]   (bad)        0      0.0%   <NA>
   (1, Inf)   (very bad)   0      0.0%   <NA>
```

UPPSALA UNIVERSITET




```
Pareto k diagnostic values:
                         Count  Pct.    Min. n_eff
(-Inf, 0.5]   (good)     18     90.0%   899
 (0.5, 0.7]   (ok)        2     10.0%   459
   (0.7, 1]   (bad)       0      0.0%   <NA>
   (1, Inf)   (very bad)  0      0.0%   <NA>
```

loo package

```
Computed from 4000 by 20 log-likelihood matrix

         Estimate  SE
elpd_loo    -29.5 3.3
p_loo         2.7 1.0

------
Monte Carlo SE of elpd_loo is 0.1.

Pareto k diagnostic values:
                         Count Pct.    Min. n_eff
(-Inf, 0.5]   (good)     18    90.0%   899
 (0.5, 0.7]   (ok)        2    10.0%   459
   (0.7, 1]   (bad)       0     0.0%   <NA>
   (1, Inf)   (very bad)  0     0.0%   <NA>

All Pareto k estimates are ok (k < 0.7).
See help('pareto-k-diagnostic') for details.
```

see more in Vehtari, Gelman & Gabry (2017b)

Stan and `loo` package

```
 Computed from 4000 by 20 log-likelihood matrix

         Estimate  SE
elpd_loo    -29.5 3.3
p_loo         2.7 1.0
------
Monte Carlo SE of elpd_loo is 0.1.

Pareto k diagnostic values:
                         Count Pct.    Min. n_eff
(-Inf, 0.5]   (good)     18    90.0%   899
 (0.5, 0.7]   (ok)        2    10.0%   459
   (0.7, 1]   (bad)       0     0.0%   <NA>
   (1, Inf)   (very bad)  0     0.0%   <NA>

All Pareto k estimates are ok (k < 0.7).
See help('pareto-k-diagnostic') for details.

Model comparison:
(negative 'elpd_diff' favors 1st model, positive favors 2nd)

elpd_diff        se
     -0.2       0.1
```

loo package

```
Computed from 4000 by 20 log-likelihood matrix

         Estimate  SE
elpd_loo    -29.5 3.3
p_loo         2.7 1.0
------
Monte Carlo SE of elpd_loo is 0.1.

Pareto k diagnostic values:
                         Count Pct.    Min. n_eff
(-Inf, 0.5]   (good)     18    90.0%   899
 (0.5, 0.7]   (ok)        2    10.0%   459
   (0.7, 1]   (bad)       0     0.0%   <NA>
   (1, Inf)   (very bad)  0     0.0%   <NA>

All Pareto k estimates are ok (k < 0.7).
See help('pareto-k-diagnostic') for details.
```

Importance sampling

- Having samples θ^s from $p(\theta^s|D)$

$$p(\tilde{y}_i|x_i, D_{-i}) \approx \frac{\sum_{s=1}^{S} p(\tilde{y}_i|\theta^s) w_i^s}{\sum_{s=1}^{S} w_i^s},$$

where w_i^s are importance weights and

$$w_i^s = \frac{p(\theta^s|x_i, D_{-i})}{p(\theta^s|D)} \propto \frac{1}{p(y_i|\theta^s)}.$$

Importance sampling

- Having samples θ^s from $p(\theta^s|D)$

$$p(\tilde{y}_i|x_i, D_{-i}) \approx \frac{\sum_{s=1}^{S} p(\tilde{y}_i|\theta^s) w_i^s}{\sum_{s=1}^{S} w_i^s},$$

where w_i^s are importance weights and

$$w_i^s = \frac{p(\theta^s|x_i, D_{-i})}{p(\theta^s|D)} \propto \frac{1}{p(y_i|\theta^s)}.$$

- If evaluated with $\tilde{y}_i = y_i$

$$p(y_i|x_i, D_{-i}) \approx \frac{1}{\sum_{s=1}^{S} \frac{1}{p(y_i|\theta^s)}},$$

UPPSALA UNIVERSITET

Stan code

$$\log(r_i^{(s)}) = \log(1/p(y_i|x_i, \theta^{(s)})) = -\text{log\_lik}[i]$$

Stan code

$$\log(r_i^{(s)}) = \log(1/p(y_i|x_i, \theta^{(s)})) = -\text{log\_lik}[i]$$

```
...
model {
  alpha ~ normal(pmualpha, psalpha);
  beta ~ normal(pmubeta, psbeta);
  y ~ normal(mu, sigma);
}
generated quantities {
  vector[N] log_lik;
  for (i in 1:N)
    log_lik[i] = normal_lpdf(y[i] | mu[i], sigma);
}
```

Pareto smoothed importance sampling LOO

1. PSIS-LOO for hierarchical models
   1.1 leave-one-group out is challenging for PSIS-LOO
   see Merkel, Furr and Rabe-Hesketh (2018) for an approach using quadrature integration

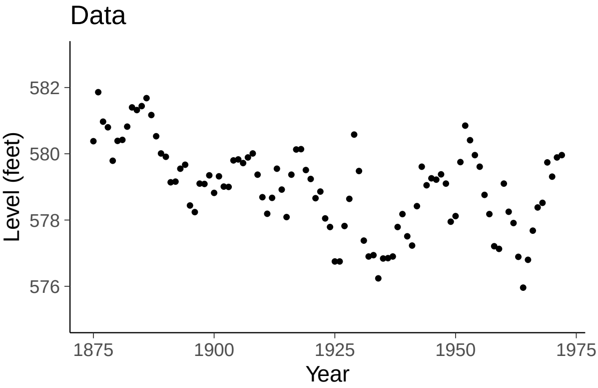
Pareto smoothed importance sampling LOO

1. PSIS-LOO for hierarchical models
   1.1 leave-one-group out is challenging for PSIS-LOO
       see Merkel, Furr and Rabe-Hesketh (2018) for an approach using quadrature integration
2. PSIS-LOO for non-factorizable models
   2.1 `mc-stan.org/loo/articles/loo2-non-factorizable.html`

Pareto smoothed importance sampling LOO

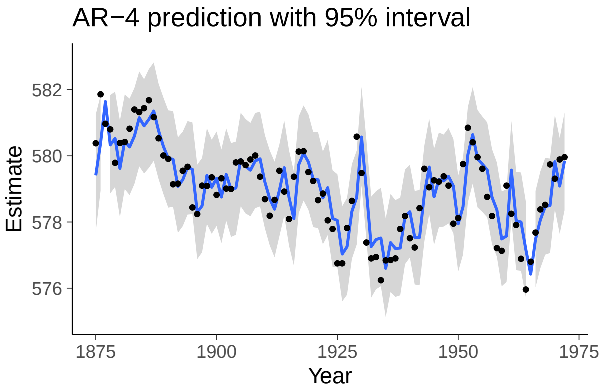
1. PSIS-LOO for hierarchical models
   1.1 leave-one-group out is challenging for PSIS-LOO
   see Merkel, Furr and Rabe-Hesketh (2018) for an approach using quadrature integration
2. PSIS-LOO for non-factorizable models
   2.1 mc-stan.org/loo/articles/loo2-non-factorizable.html
3. PSIS-LOO for time series
   3.1 Approximate leave-future-out cross-validation
   mc-stan.org/loo/articles/loo2-lfo.html

PSIS-LOO for time series

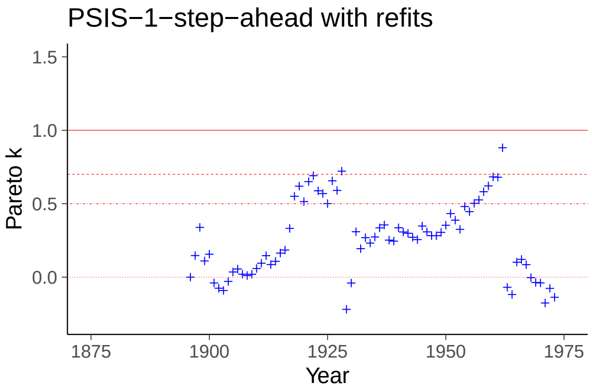


UPPSALA UNIVERSITET

PSIS-LOO for time series



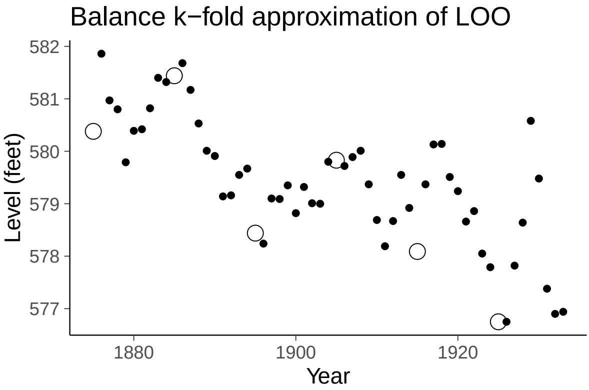
PSIS-LOO for time series

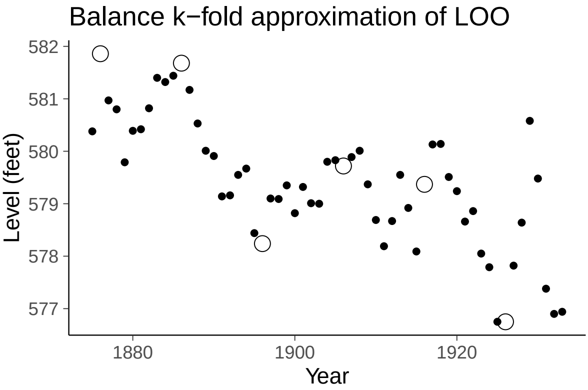


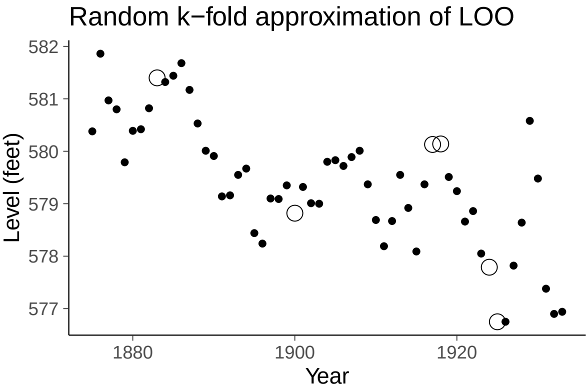
mc-stan.org/loo/articles/loo2-lfo.html

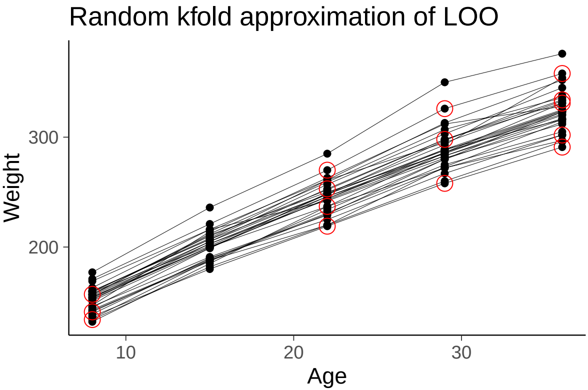
K-fold cross-validation

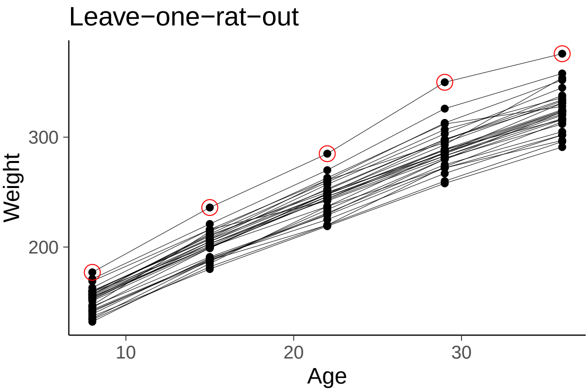
1. K-fold cross-validation can approximate LOO
   1.1 all uses for LOO
2. K-fold cross-validation can be used for hierarchical models
   2.1 good for leave-one-group-out
3. K-fold cross-validation can be used for time series
   3.1 with leave-block-out

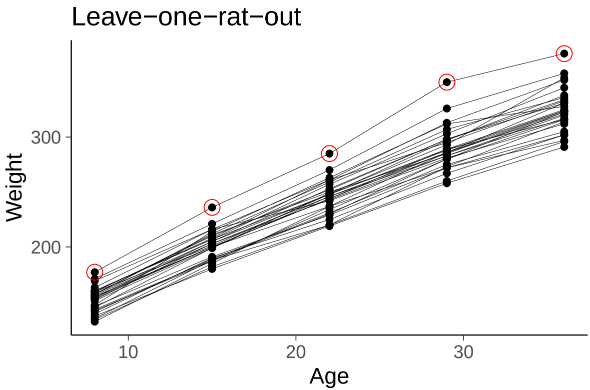

Balance k−fold approximation of LOO
Level (feet)
582
581
580
579
578
577
1880
1900
1920
Year


Balance k−fold approximation of LOO
Level (feet)
582
581
580
579
578
577
1880
1900
1920
Year


Random k−fold approximation of LOO
582
581
580
579
578
577
Level (feet)
1880
1900
1920
Year


Random kfold approximation of LOO
Weight
300
200
10
20
30
Age


Leave−one−rat−out
Weight
300
200
10
20
30
Age



kfold_split_random()

kfold_split_balanced()

kfold_split_stratified()

Cross-validation for model assessment

1. CV is good for model assessment when application specific utility/cost functions are used
   1.1 e.g. 90% absolute error

Cross-validation for model assessment

1. CV is good for model assessment when application specific utility/cost functions are used
   1.1 e.g. 90% absolute error
2. Also useful in model checking in similar way as posterior predictive checking (PPC)
   2.1 model misspecification diagnostics (e.g. Pareto-k and p_loo)
   2.2 checking calibration of leave-one-out predictive posteriors (ppc_loo_pit in bayesplot)

see demos `avehtari.github.io/modelselection/`

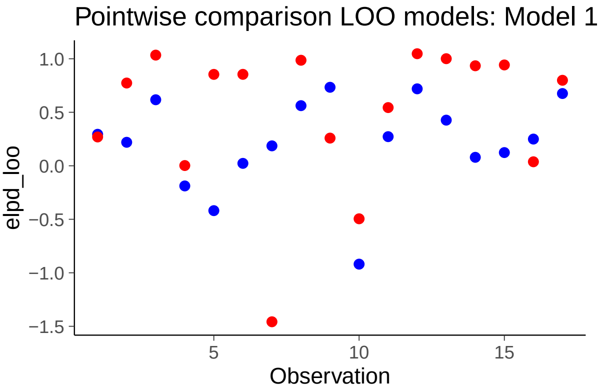
UPPSALA UNIVERSITET

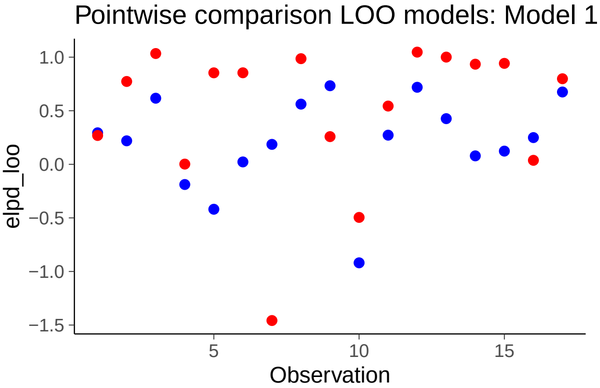
Model comparison

1. "A popular hypothesis has it that primates with larger brains produce more energetic milk, so that brains can grow quickly" (from Statistical Rethinking)
   1.1 Model 1: formula = kcal.per.g $\sim$ neocortex
   1.2 Model 2: formula = kcal.per.g $\sim$ neocortex + log(mass)

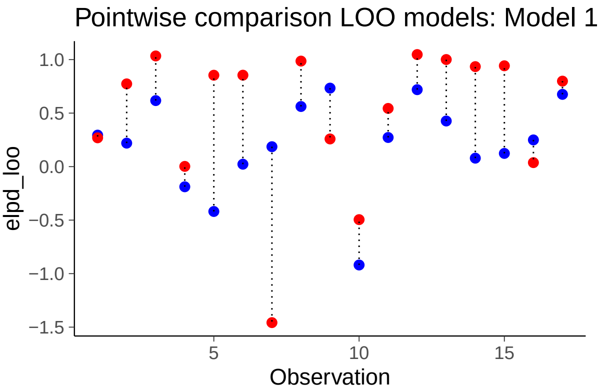
mc-stan.org/loo/articles/loo2-example.html

UPPSALA UNIVERSITET

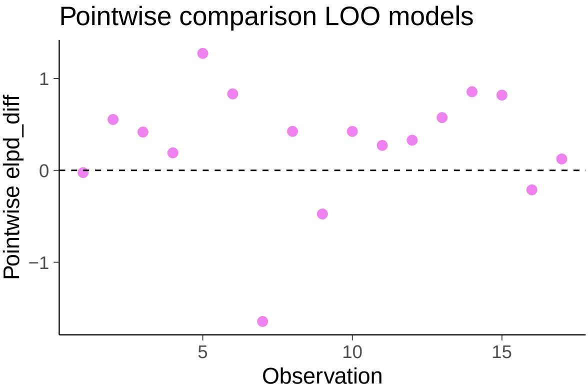




Model 1 elpd_loo $\approx$ 3.7, SE=1.8
Model 2 elpd_loo $\approx$ 8.4, SE=2.8



Model 1 elpd_loo $\approx$ 3.7, SE=1.8
Model 2 elpd_loo $\approx$ 8.4, SE=2.8

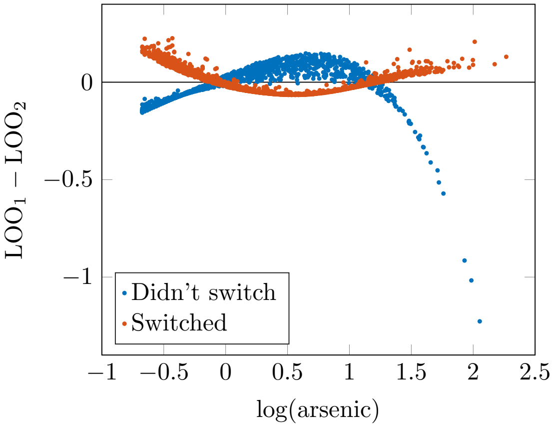



```
Model comparison:
(negative 'elpd_diff' favors 1st model, positive favors 2nd)

elpd_diff        se
      4.7       2.7
```

UPPSALA UNIVERSITET

Arsenic well example – Model comparison



An estimated difference in $\mathrm{elpd}_{\mathrm{loo}}$ of 16.4 with SE of 4.4.

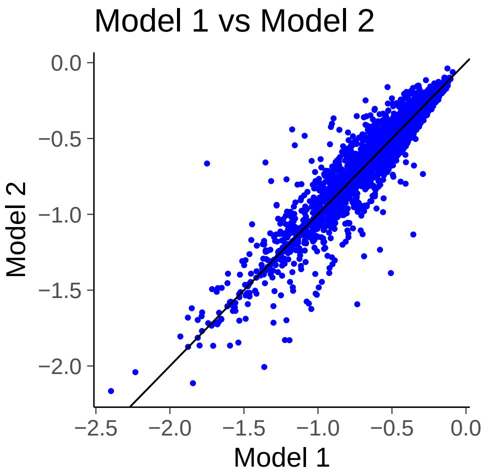
see Vehtari, Gelman & Gabry (2017a)

Arsenic well example – Model comparison

1. Probability of switching well with high arsenic level in rural Bangladesh
   1.1 Model 1 covariates: log(arsenic) and distance
   1.2 Model 2 covariates: log(arsenic), distance and education level

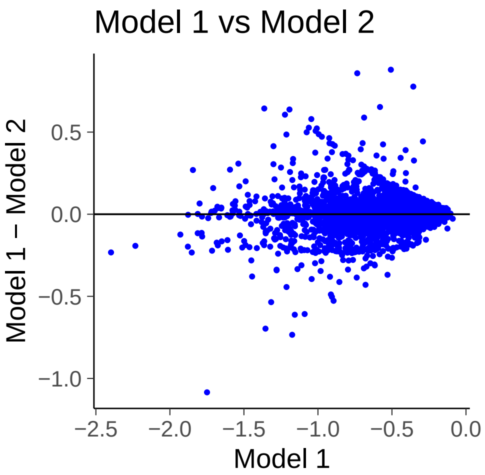
Gelman, Hill & Vehtari (2020): Regression and Other Stories, Chapter 13.

Arsenic well example – Model comparison



Model 1 elpd_loo ≈ -1952, SE=16
Model 2 elpd_loo ≈ -1938, SE=17

Arsenic well example – Model comparison




```
> loo_compare(model1, model2)
       elpd_diff se_diff
model2    0.0       0.0
model1  -14.4       6.1
```

see Vehtari, Gelman & Gabry (2017a)

Arsenic well example – Model comparison

```
> loo_compare(model1, model2)
       elpd_diff  se_diff
model2    0.0        0.0
model1  -14.4        6.1
```

`se_diff` and normal approximation for the uncertainty in the difference is good only if models are well specified and the number of observations is relatively big (more details in a forthcoming article).

What if one is not clearly better than others?

What if one is not clearly better than others?

1. Continuous expansion including all models?
   1.1 and then analyse the posterior distribution directly
   avehtari.github.io/modelselection/betablockers.html
   1.2 sparse priors like regularized horseshoe prior instead of variable selection
   video, refs and demos at
   avehtari.github.io/modelselection/

What if one is not clearly better than others?

1. Continuous expansion including all models?
   1.1 and then analyse the posterior distribution directly
   avehtari.github.io/modelselection/betablockers.html
   1.2 sparse priors like regularized horseshoe prior instead of variable selection
   video, refs and demos at
   avehtari.github.io/modelselection/
2. Model averaging with BMA or Bayesian stacking?
   mc-stan.org/loo/articles/loo2-example.html

What if one is not clearly better than others?

1. Continuous expansion including all models?
   1.1 and then analyse the posterior distribution directly
   avehtari.github.io/modelselection/betablockers.html
   1.2 sparse priors like regularized horseshoe prior instead of variable selection
   video, refs and demos at
   avehtari.github.io/modelselection/
2. Model averaging with BMA or Bayesian stacking?
   mc-stan.org/loo/articles/loo2-example.html
3. In a nested case choose simpler if assuming some cost for extra parts?
   andrewgelman.com/2018/07/26/parsimonious-principle-vs-integration-uncertainties/

What if one is not clearly better than others?

1. Continuous expansion including all models?
   1.1 and then analyse the posterior distribution directly
   avehtari.github.io/modelselection/betablockers.html
   1.2 sparse priors like regularized horseshoe prior instead of variable selection
   video, refs and demos at
   avehtari.github.io/modelselection/
2. Model averaging with BMA or Bayesian stacking?
   mc-stan.org/loo/articles/loo2-example.html
3. In a nested case choose simpler if assuming some cost for extra parts?
   andrewgelman.com/2018/07/26/parsimonious-principle-vs-integration-uncertainties/
4. In a nested case choose more complex if you want to take into account all the uncertainties.
   andrewgelman.com/2018/07/26/parsimonious-principle-vs-integration-uncertainties/

Cross-validation and model selection

1. Cross-validation can be used for model selection if
   1.1 small number of models
   1.2 the difference between models is clear

Cross-validation and model selection

1. Cross-validation can be used for model selection if
   1.1 small number of models
   1.2 the difference between models is clear
2. Do not use cross-validation to choose from a large set of models
   2.1 selection process leads to overfitting

Cross-validation and model selection

1. Cross-validation can be used for model selection if
   1.1 small number of models
   1.2 the difference between models is clear
2. Do not use cross-validation to choose from a large set of models
   2.1 selection process leads to overfitting
3. Overfitting in selection process is not unique for cross-validation

UPPSALA UNIVERSITET

Selection induced bias and overfitting

- Selection induced bias in cross-validation
  - same data is used to assess the performance and make the selection
  - the selected model fits more to the data
  - the CV estimate for the selected model is biased
  - recognized already, e.g., by Stone (1974)

UPPSALA UNIVERSITET

Selection induced bias and overfitting

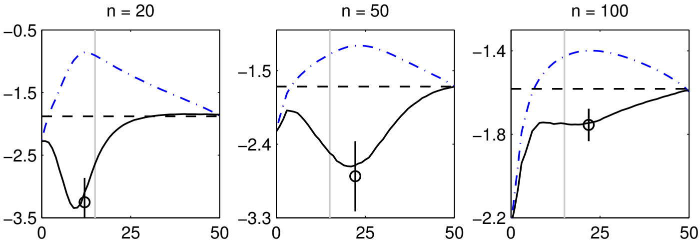
- Selection induced bias in cross-validation
  - same data is used to assess the performance and make the selection
  - the selected model fits more to the data
  - the CV estimate for the selected model is biased
  - recognized already, e.g., by Stone (1974)
- Performance of the selection process itself can be assessed using two level cross-validation, but it does not help choosing better models

UPPSALA UNIVERSITET

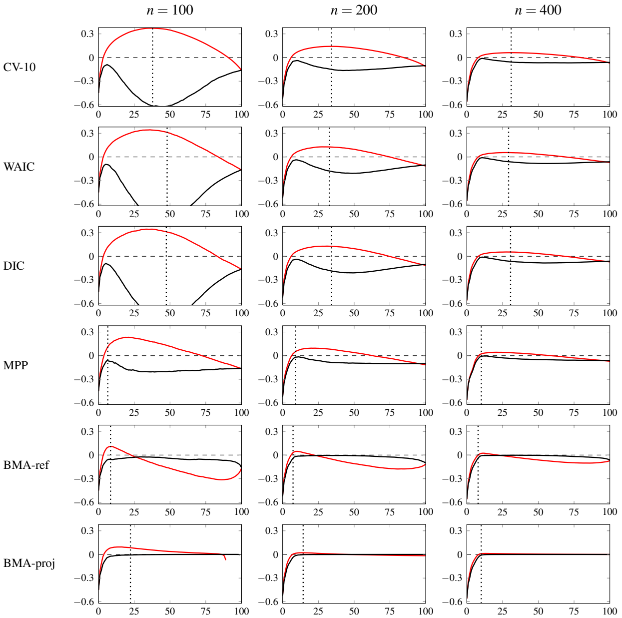
Selection induced bias and overfitting

- Selection induced bias in cross-validation
  - same data is used to assess the performance and make the selection
  - the selected model fits more to the data
  - the CV estimate for the selected model is biased
  - recognized already, e.g., by Stone (1974)
- Performance of the selection process itself can be assessed using two level cross-validation, but it does not help choosing better models
- Bigger problem if there is a large number of models as in covariate selection

Selection induced bias in variable selection



Selection induced bias in variable selection



Section 3

Information criteria

WAIC vs PSIS-LOO

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

1. WAIC has same assumptions as LOO

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

1. WAIC has same assumptions as LOO
2. PSIS-LOO is more accurate

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

1. WAIC has same assumptions as LOO
2. PSIS-LOO is more accurate
3. PSIS-LOO has much better diagnostics

see Vehtari, Gelman & Gabry (2017a)

UPPSALA
UNIVERSITET

WAIC vs PSIS-LOO

1. WAIC has same assumptions as LOO
2. PSIS-LOO is more accurate
3. PSIS-LOO has much better diagnostics
4. LOO makes the prediction assumption more clear, which helps if K-fold-CV is needed instead

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

1. WAIC has same assumptions as LOO
2. PSIS-LOO is more accurate
3. PSIS-LOO has much better diagnostics
4. LOO makes the prediction assumption more clear, which helps if K-fold-CV is needed instead
5. Multiplying by -2 doesn't give any benefit (Watanabe didn't multiply by -2)

see Vehtari, Gelman & Gabry (2017a)

UPPSALA UNIVERSITET

*IC

1. AIC uses maximum likelihood estimate for prediction
2. DIC uses posterior mean for prediction
3. BIC is an approximation for marginal likelihood
4. TIC, NIC, RIC, PIC, BPIC, QIC, AICc, ...

UPPSALA UNIVERSITET

Section 4

Model averaging

Section 5

Summary

UPPSALA UNIVERSITET

Take-home messages

1. It's good to think predictions of observables, because observables are the only ones we can observe
2. Cross-validation can simulate predicting and observing new data
3. Cross-validation is good if you don't trust your model
4. Different variants of cross-validation are useful in different scenarios
5. Cross-validation has high variance, and **if** you trust your model you can beat cross-validation in accuracy

Take-home messages

1. It's good to think predictions of observables, because observables are the only ones we can observe
2. Cross-validation can simulate predicting and observing new data
3. Cross-validation is good if you don't trust your model
4. Different variants of cross-validation are useful in different scenarios
5. Cross-validation has high variance, and **if** you trust your model you can beat cross-validation in accuracy

Take-home messages

UPPSALA UNIVERSITET

1. It's good to think predictions of observables, because observables are the only ones we can observe
2. Cross-validation can simulate predicting and observing new data
3. Cross-validation is good if you don't trust your model
4. Different variants of cross-validation are useful in different scenarios
5. Cross-validation has high variance, and **if** you trust your model you can beat cross-validation in accuracy

UPPSALA UNIVERSITET

Take-home messages

1. It's good to think predictions of observables, because observables are the only ones we can observe
2. Cross-validation can simulate predicting and observing new data
3. Cross-validation is good if you don't trust your model
4. Different variants of cross-validation are useful in different scenarios
5. Cross-validation has high variance, and **if** you trust your model you can beat cross-validation in accuracy

Take-home messages

1. It's good to think predictions of observables, because observables are the only ones we can observe
2. Cross-validation can simulate predicting and observing new data
3. Cross-validation is good if you don't trust your model
4. Different variants of cross-validation are useful in different scenarios
5. Cross-validation has high variance, and **if** you trust your model you can beat cross-validation in accuracy