Model assessment, selection and averaging

Part 1: cross-validation
Part 2: projection predictive inference

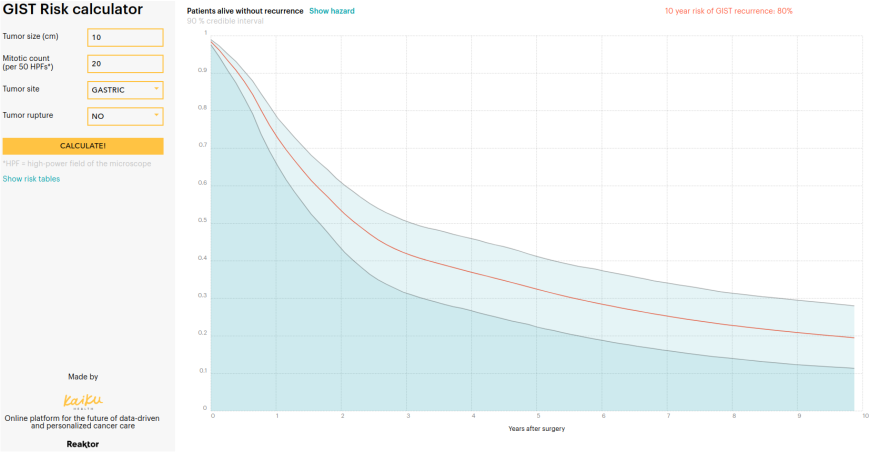
Aki Vehtari
Aalto University, Finland

Slides and extra material at avehtari.github.io/masterclass/

Predicting concrete quality

Slides and extra material at avehtari.github.io/masterclass/

Predicting cancer recurrence



Slides and extra material at avehtari.github.io/masterclass/

Model assessment, comparison, selection and averaging

- Modeling complex phenomena with models that are much simpler than the nature (M-open)

Model assessment, comparison, selection and averaging

- Modeling complex phenomena with models that are much simpler than the nature (M-open)
- Decision theoretical approch in spirit of
  - Lindley, Box, Rubin, Bernardo & Smith, etc.

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
```

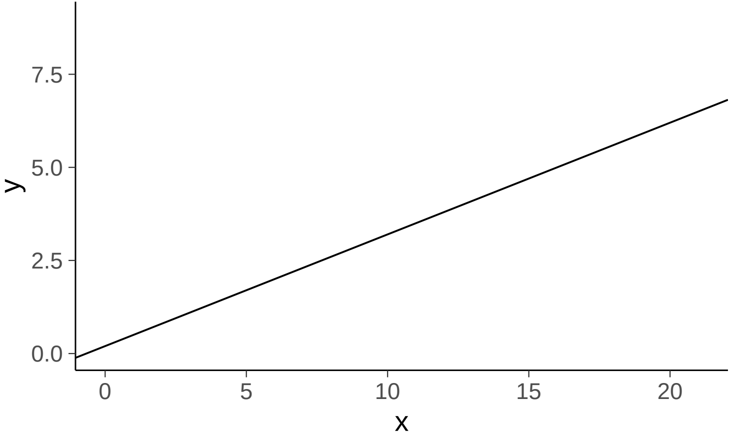
Outline

- What is cross-validation
  - Leave-one-out cross-validation (elpd_loo, p_loo)
  - Uncertainty in LOO (SE)
- When is cross-validation applicable?
  - data generating mechanisms and prediction tasks
  - leave-many-out cross-validation
- Fast cross-validation
  - PSIS and diagnostics in loo package (Pareto k, n_eff, Monte Carlo SE)
  - K-fold cross-validation
- Related methods (WAIC, *IC, BF)
- Model comparison and selection (elpd_diff, se)
- Model averaging with Bayesian stacking

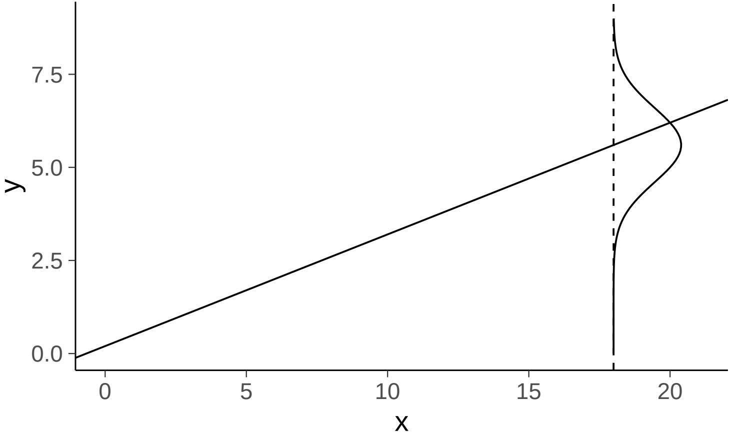
Outline

- What is cross-validation
  - Leave-one-out cross-validation (elpd_loo, p_loo)
  - Uncertainty in LOO (SE)
- When is cross-validation applicable?
  - data generating mechanisms and prediction tasks
  - leave-many-out cross-validation
- Fast cross-validation
  - PSIS and diagnostics in loo package (Pareto k, n_eff, Monte Carlo SE)
  - K-fold cross-validation
- Related methods (WAIC, *IC, BF)
- Model comparison and selection (elpd_diff, se)
- Model averaging with Bayesian stacking
- Part 2: Projective Inference in High-dimensional Problems: Prediction and Feature Selection

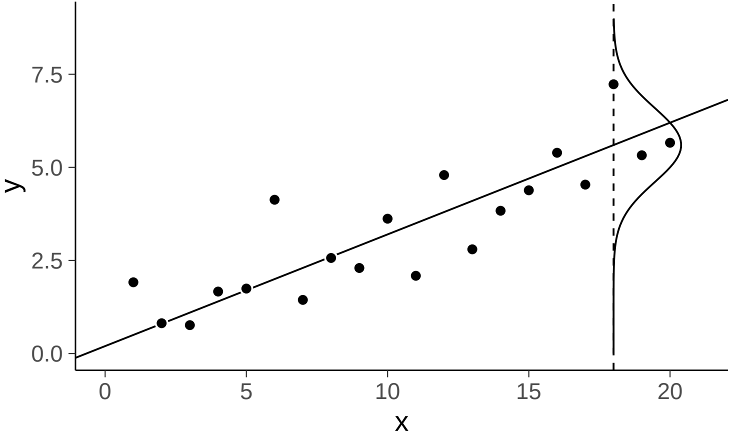
True mean y = a + bx



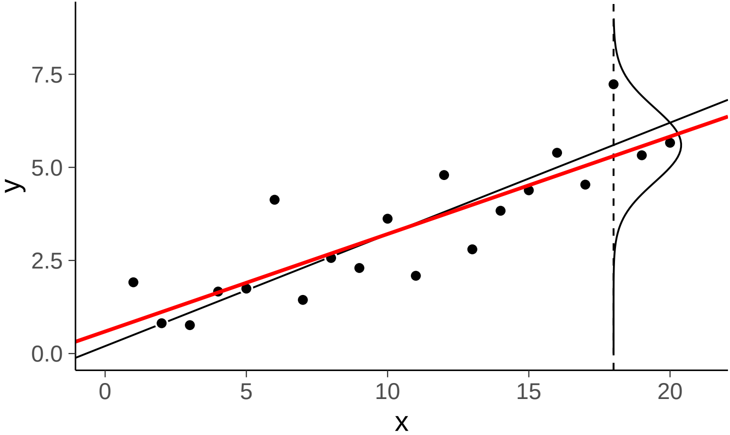
True mean and sigma



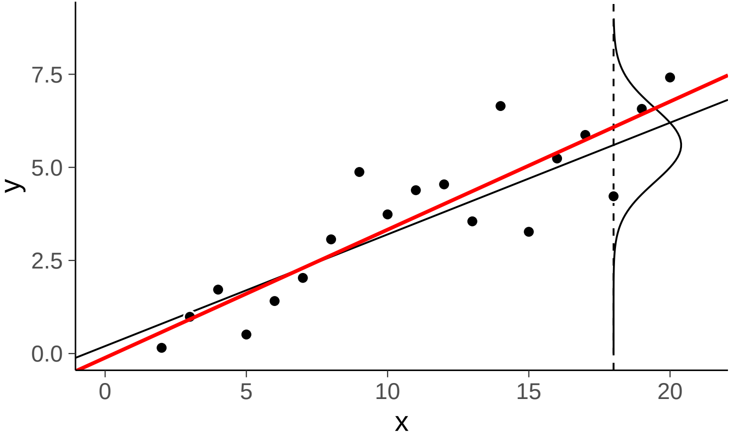
Data



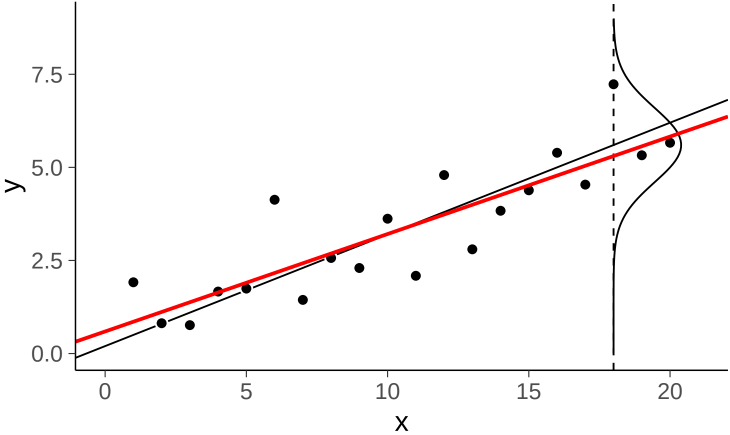
Posterior mean



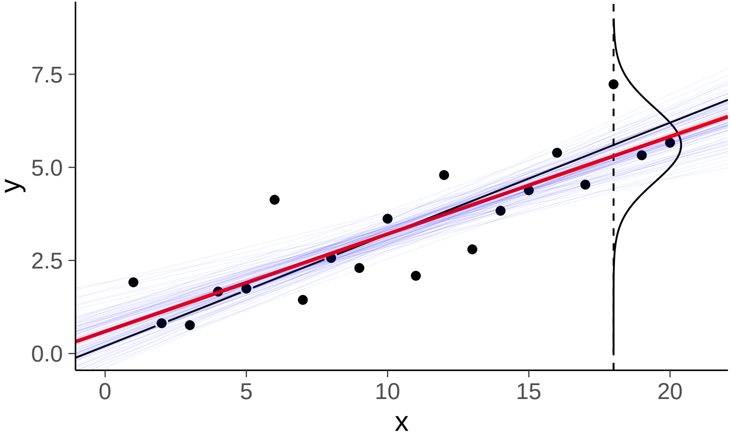
Posterior mean, alternative data realisation



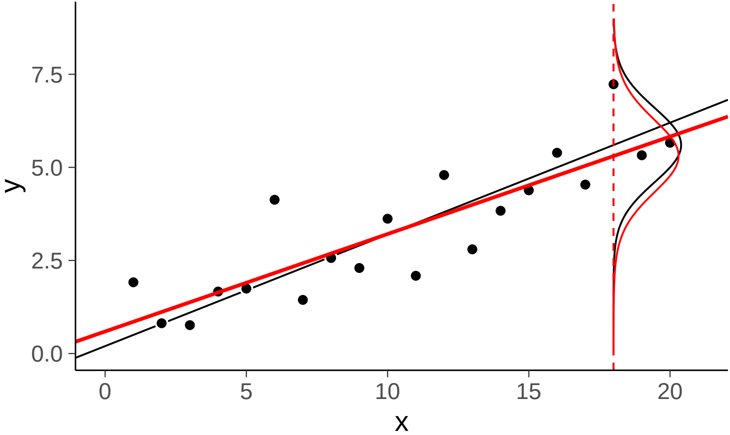
Posterior mean



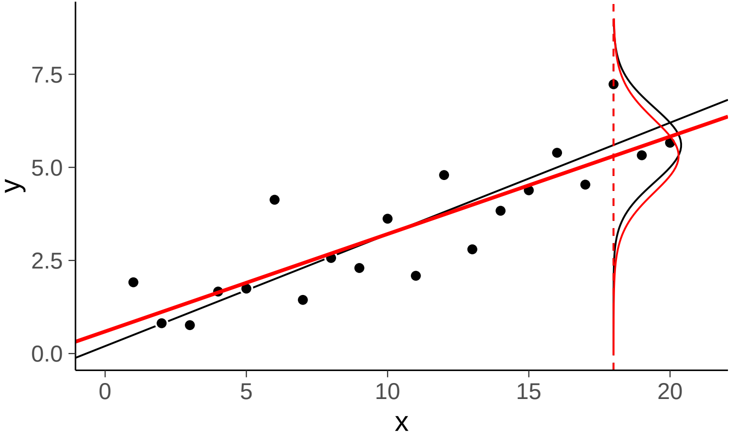
Posterior draws



Posterior predictive distribution

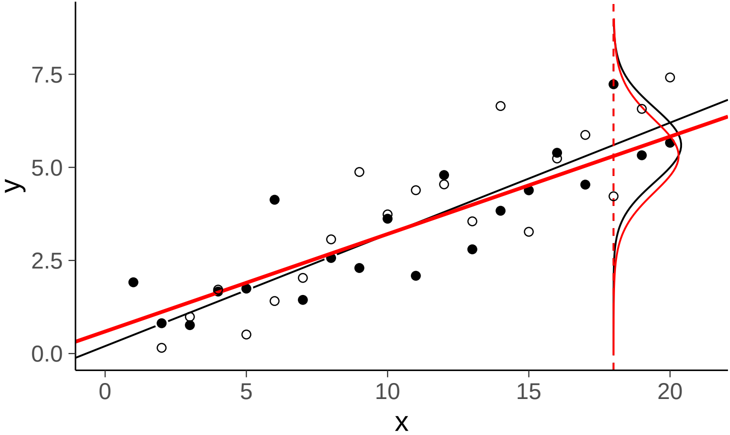


Posterior predictive distribution

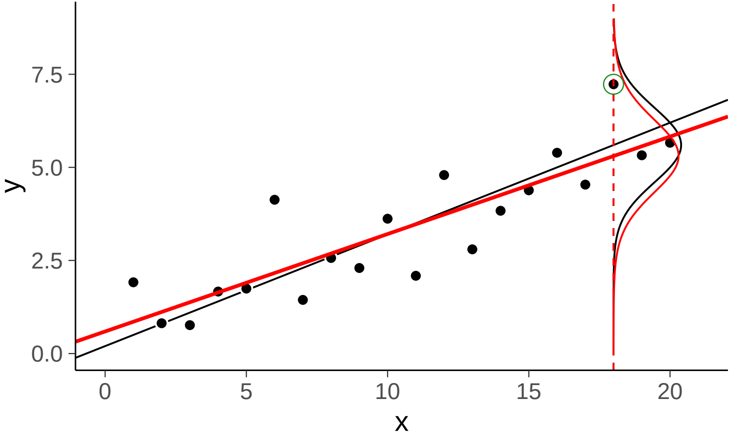


$$p(\tilde{y}|\tilde{x} = 18, x, y) = \int p(\tilde{y}|\tilde{x} = 18, \theta)p(\theta|x, y)d\theta$$

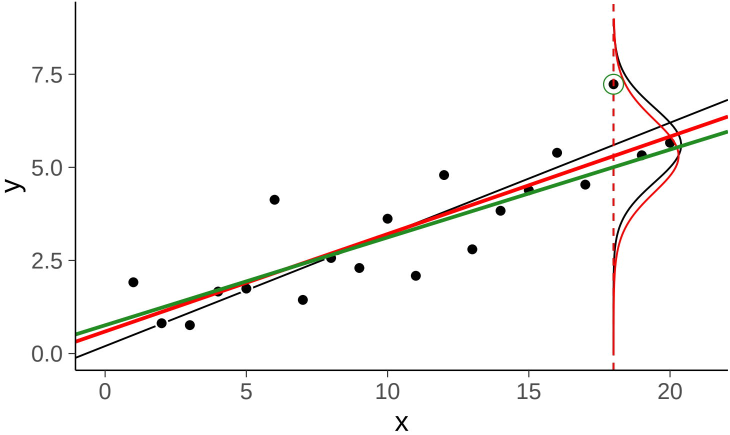
New data



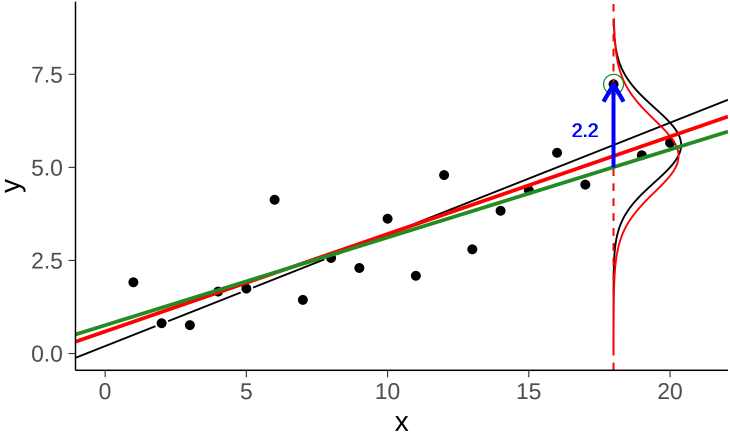
Posterior predictive distribution



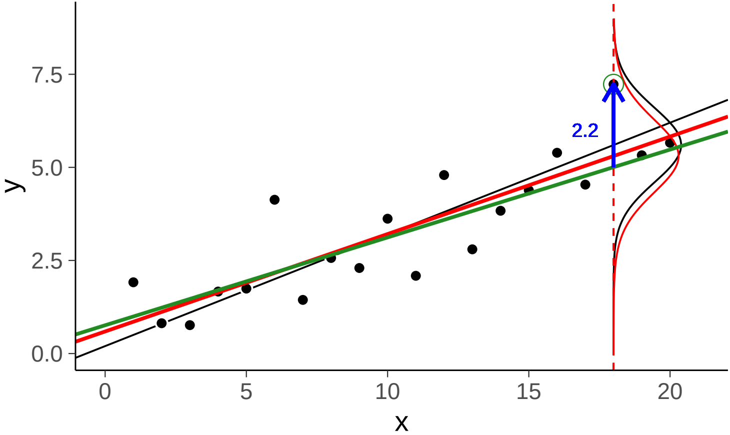
Leave−one−out mean



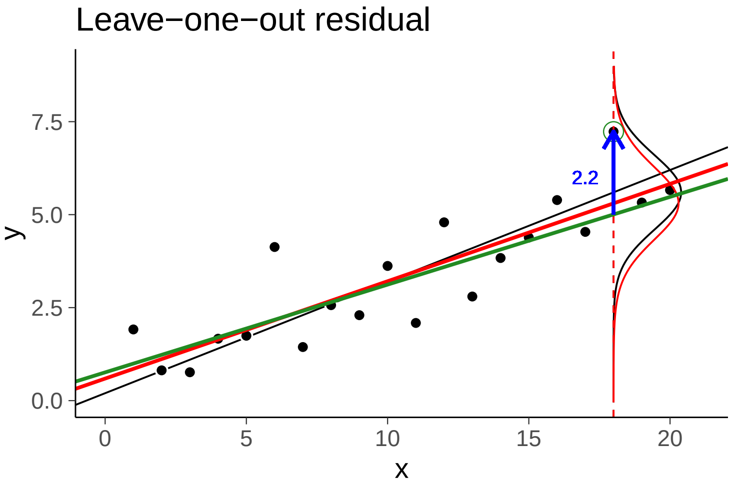
Leave−one−out residual



Leave−one−out residual

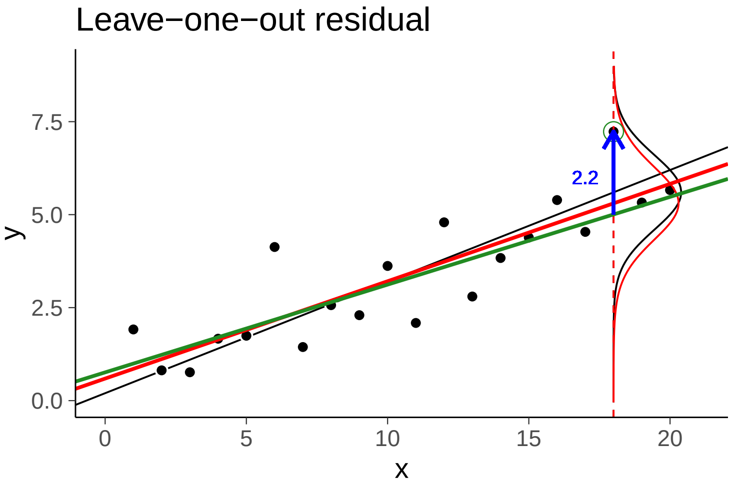


$$y_{18} - E[p(\tilde{y}|\tilde{x} = 18, x_{-18}, y_{-18})]$$



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

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

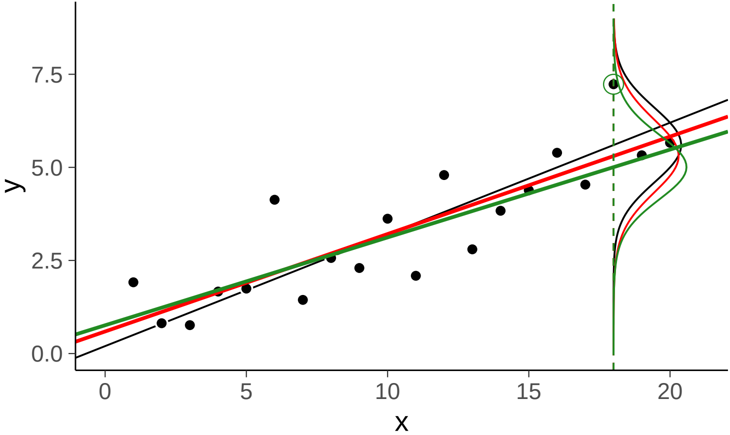


$y_{18} - E[p(\tilde{y}|\tilde{x} = 18, x_{-18}, y_{-18})]$

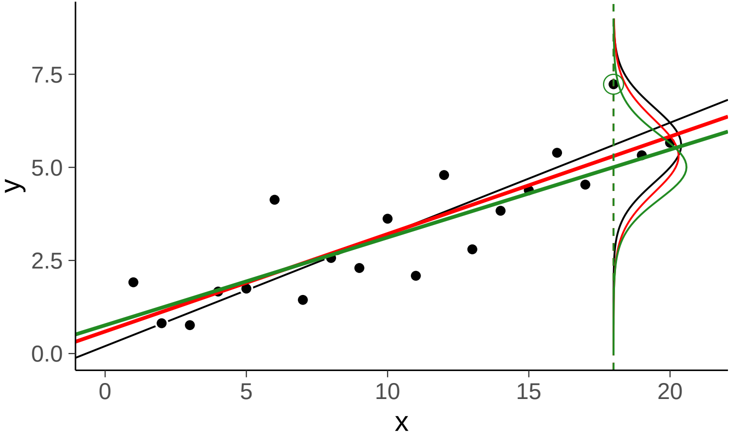
Can be use to compute, e.g., RMSE, R^2, 90% error

See LOO-R^2 at avehtari.github.io/bayes_R2/bayes_R2.html

Leave−one−out predictive distribution

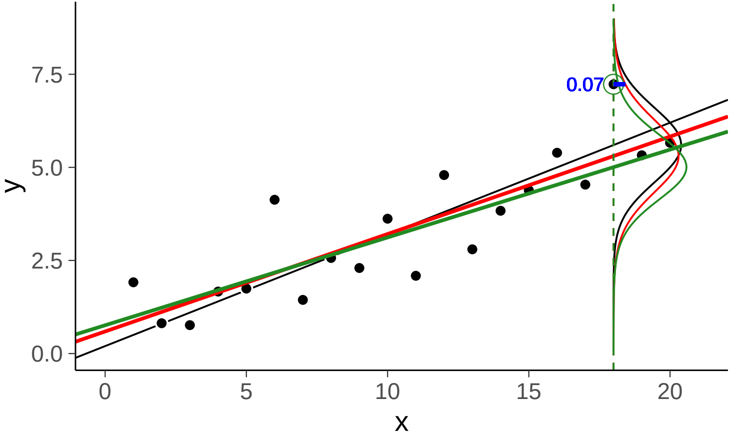


Leave−one−out predictive distribution

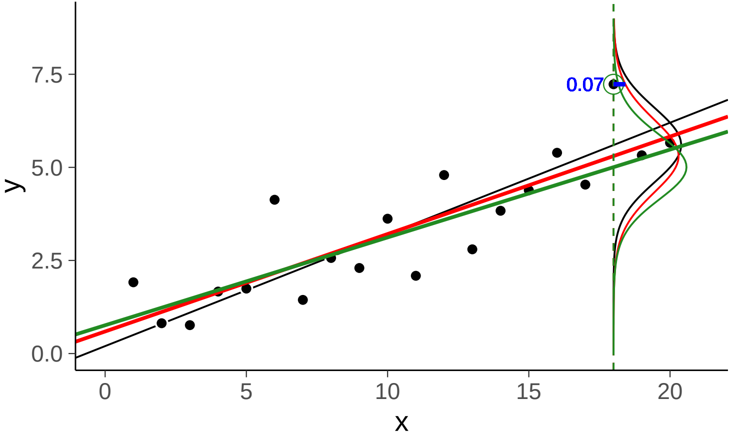


$$p(\tilde{y}|\tilde{x}=18, x_{-18}, y_{-18}) = \int p(\tilde{y}|\tilde{x}=18,\theta)p(\theta|x_{-18}, y_{-18})d\theta$$

Posterior predictive density

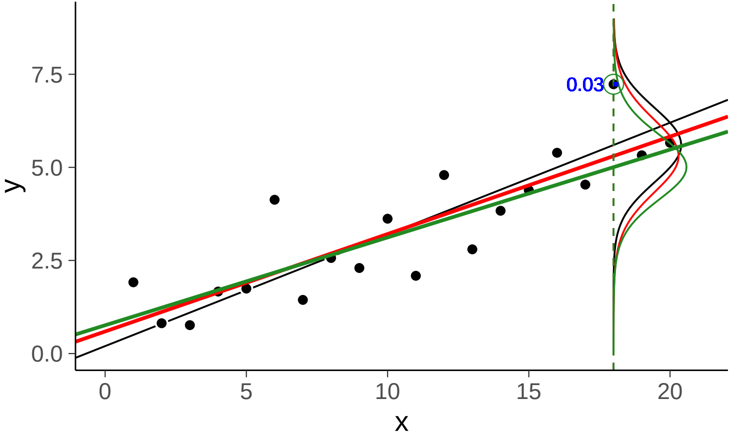


Posterior predictive density



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

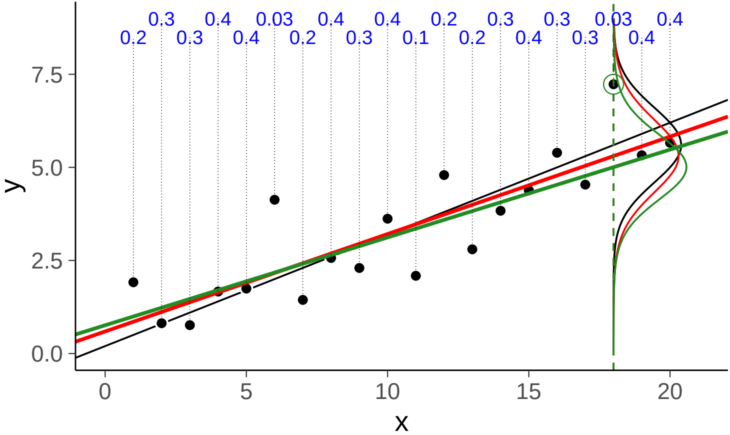
Leave−one−out predictive density



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

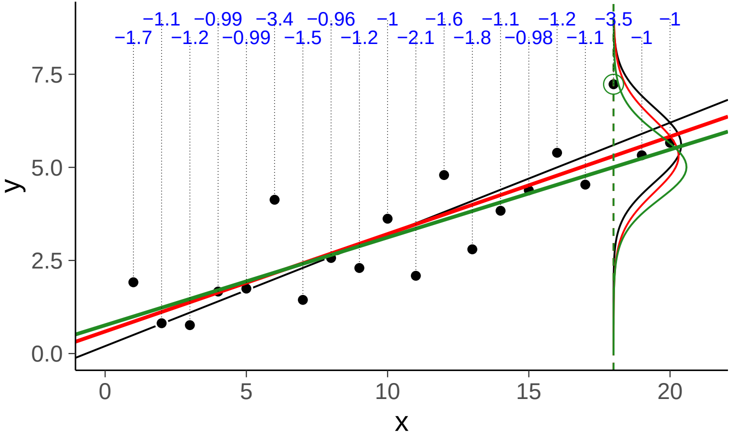
$p(\tilde{y} = y_{18} | \tilde{x} = 18, x_{-18}, y_{-18}) \approx 0.03$

Leave−one−out predictive densities



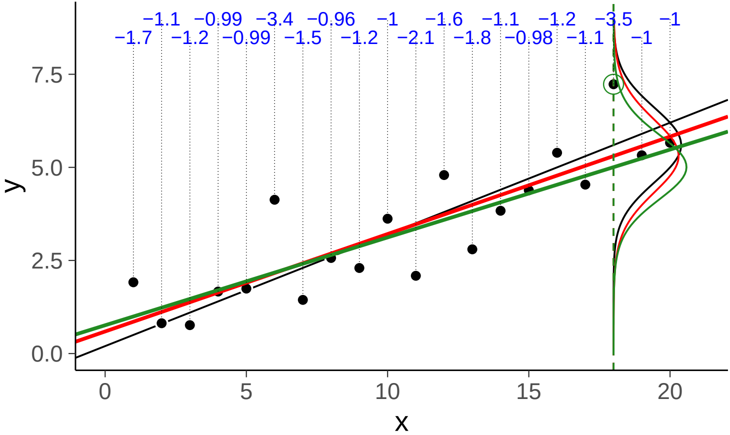
$p(y_i|x_i, x_{-i}, y_{-i}), \quad i = 1, \ldots, 20$

Leave−one−out log predictive densities



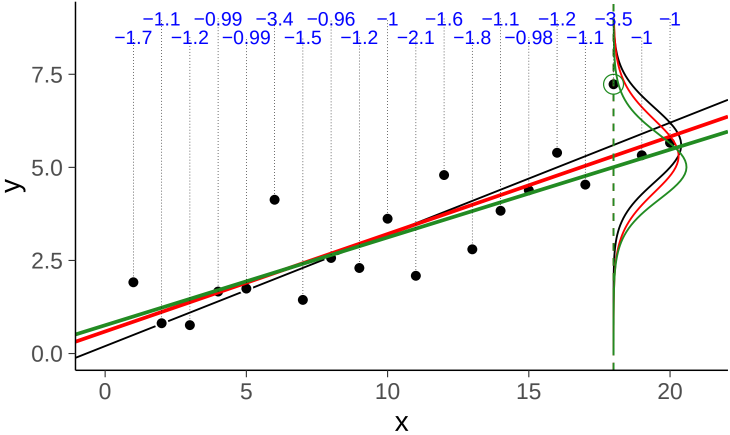
$\log p(y_i|x_i, x_{-i}, y_{-i}), \quad i = 1, \ldots, 20$

Leave−one−out log predictive densities

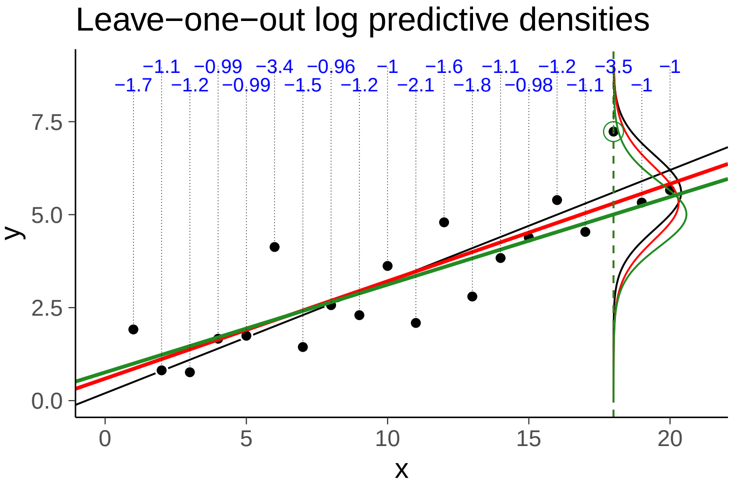


$\sum_{i=1}^{20} \log p(y_i|x_i, x_{-i}, y_{-i}) \approx -29.5$

Leave−one−out log predictive densities



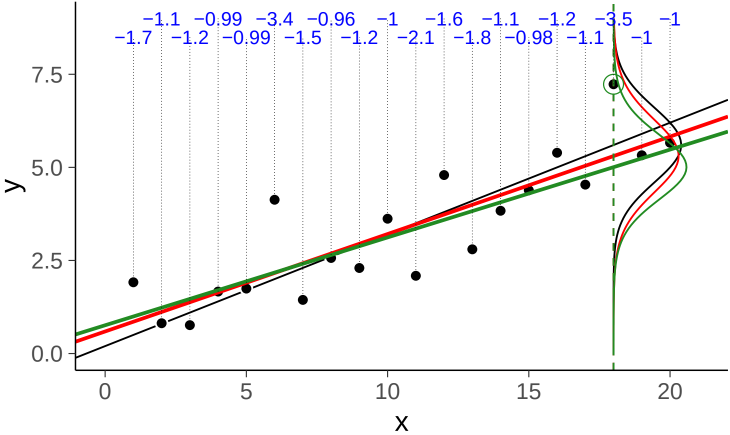
$\text{elpd\_loo} = \sum_{i=1}^{20} \log p(y_i | x_i, x_{-i}, y_{-i}) \approx -29.5$



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

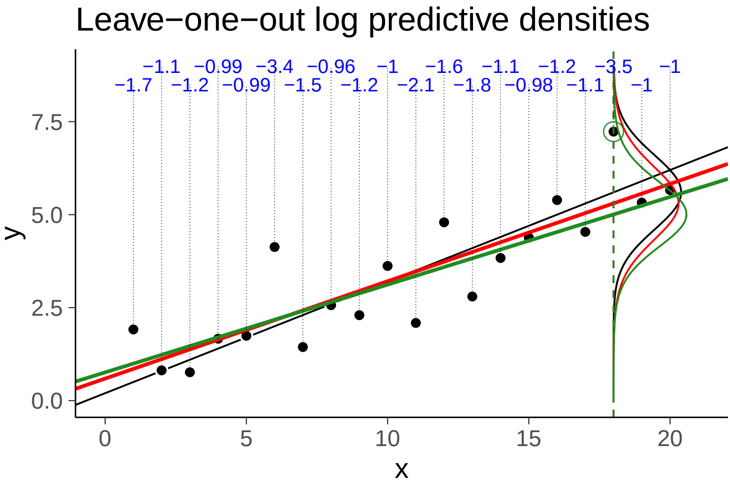
unbiased estimate of log posterior pred. density for new data

Leave−one−out log predictive densities



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

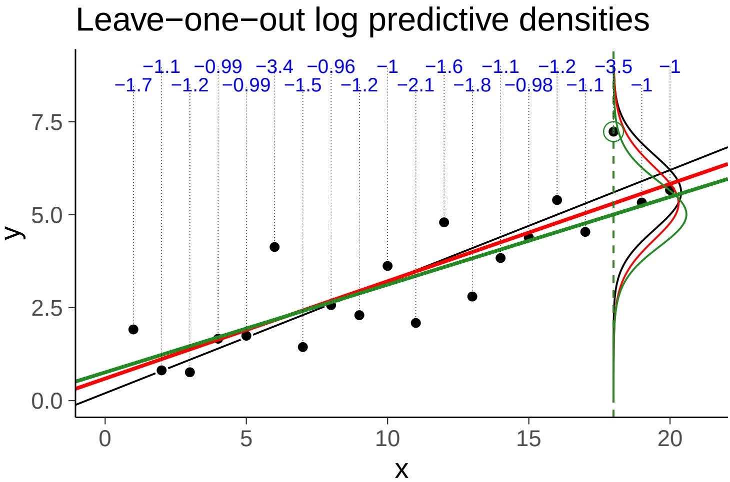
$$\text{lpd} = \sum_{i=1}^{20} \log p(y_i | x_i, x, y) \approx -26.8$$



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

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

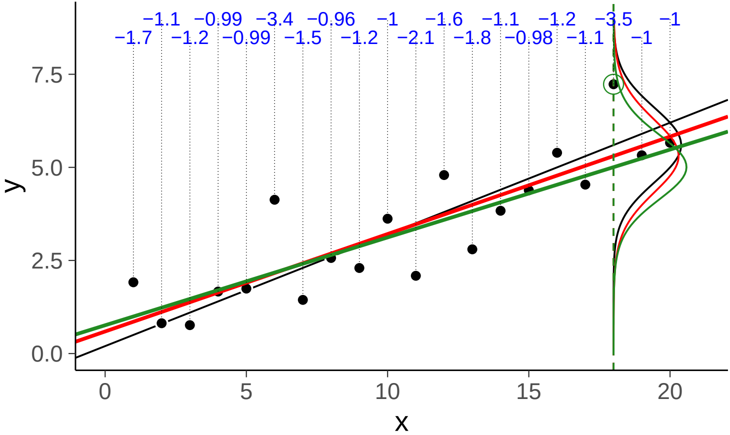
$\text{p\_loo} = \text{lpd} - \text{elpd\_loo} \approx 2.7$



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

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

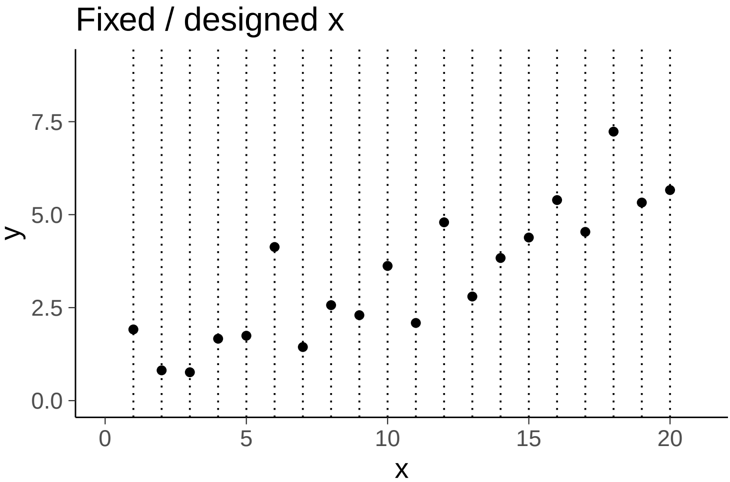
Leave−one−out log predictive densities



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

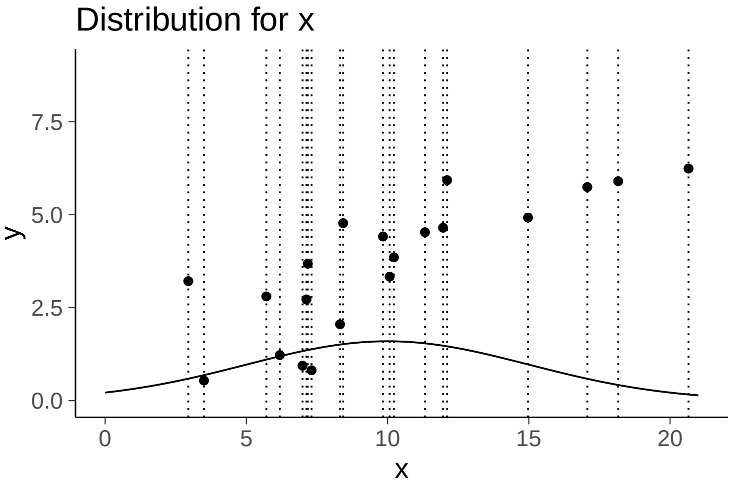
SE $= \mathrm{sd}(\log p(y_i|x_i, x_{-i}, y_{-i})) \cdot \sqrt{20} \approx 3.3$

see Vehtari, Gelman & Gabry (2017a) and Vehtari & Ojanen (2012) for more



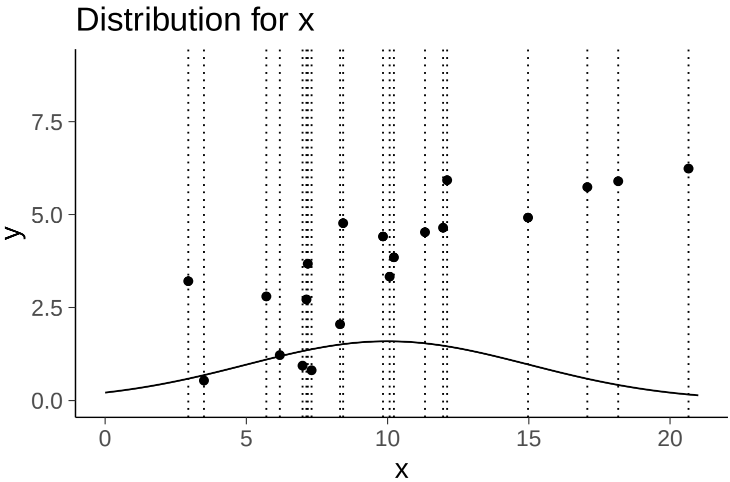
LOO is ok for fixed / designed x. SE is uncertainty about $y|x$.

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/



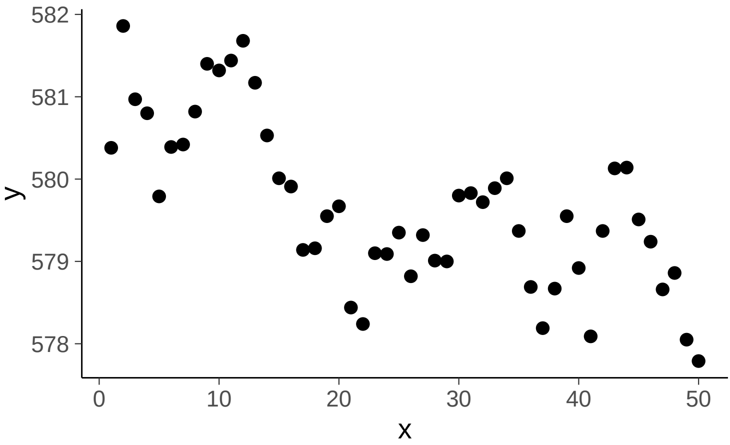
LOO is ok for random x. SE is uncertainty about $y|x$ and x.

Covariate shift can be handled with importance weighting or modelling

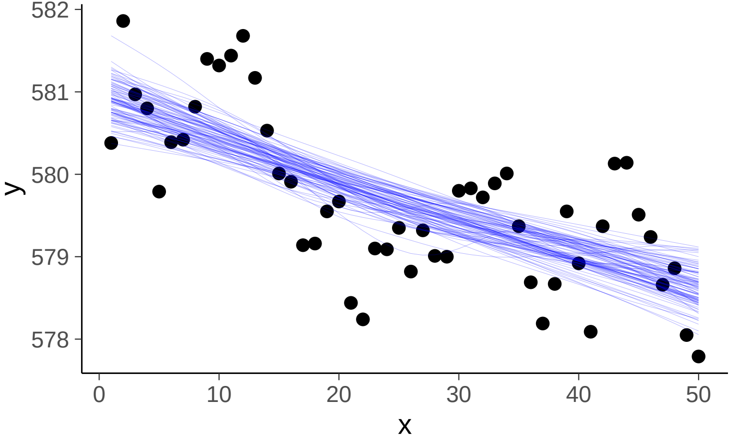
see Vehtari & Ojanen (2012) and andrewgelman.com/2018/08/03/loo-cross-validation-approaches-valid/

loo package

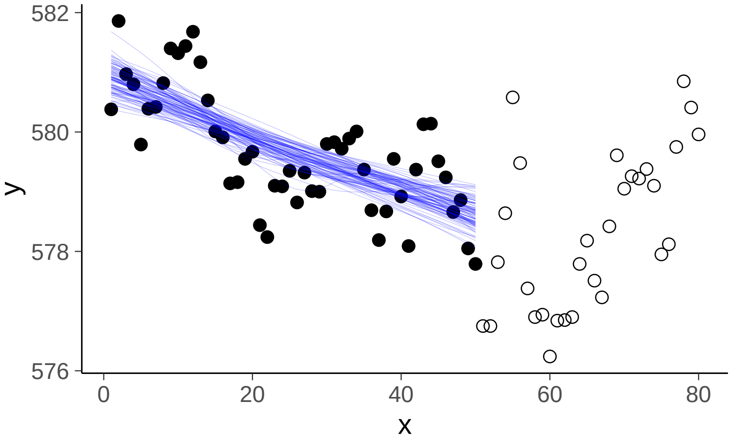
```
 Computed from 4000 by 20 log-likelihood matrix

         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.
```

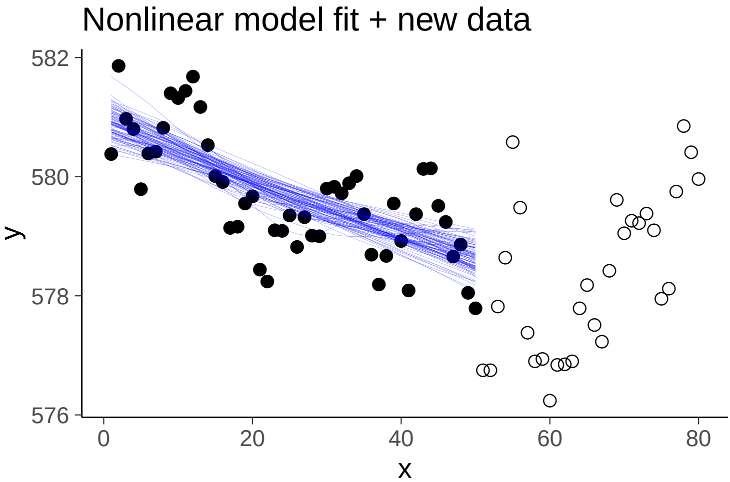

582
581
580
579
578
y
0
10
20
30
40
50
x

Nonlinear model fit

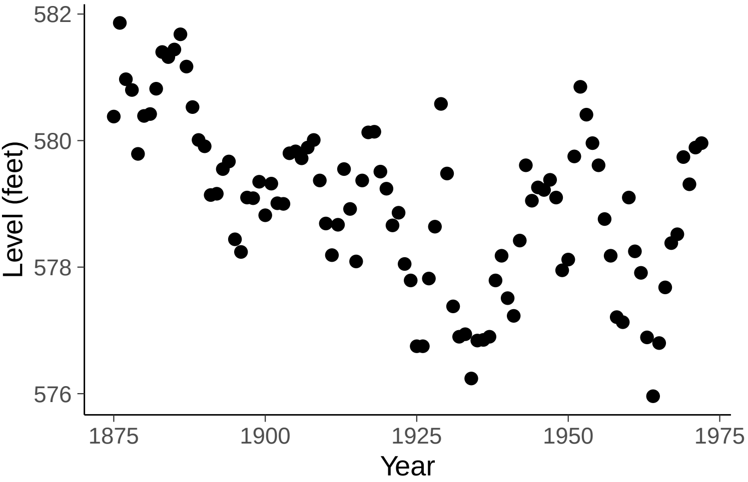


Nonlinear model fit + new data

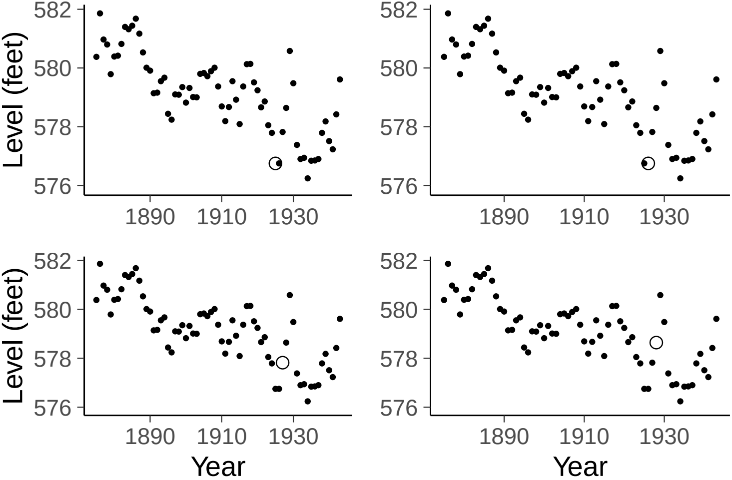




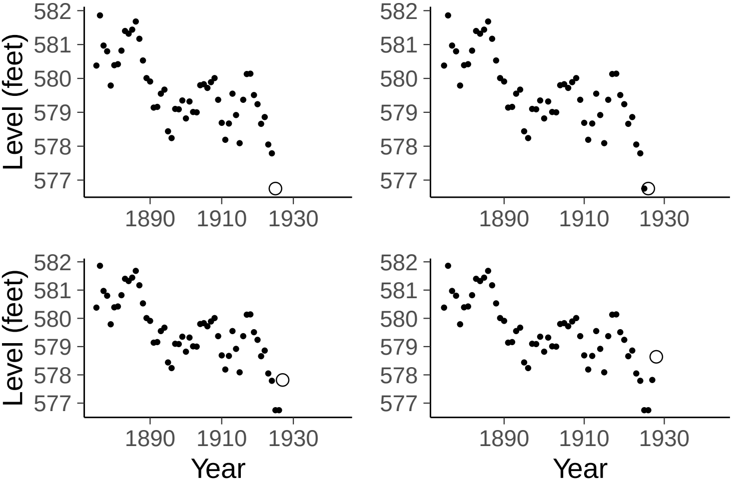
Extrapolation is more difficult



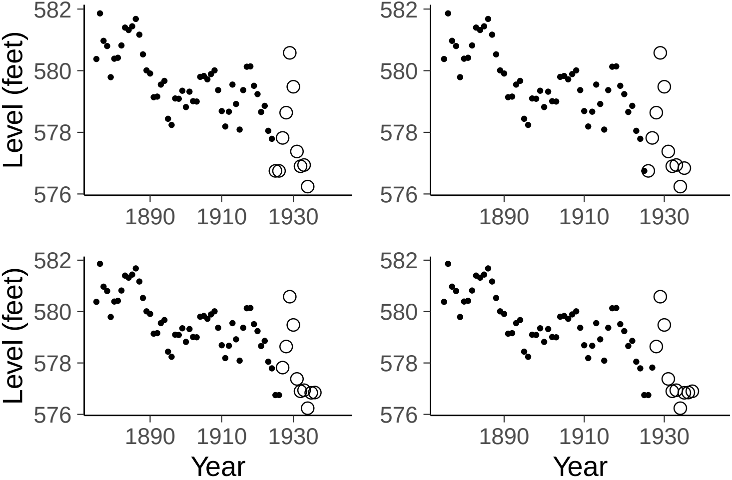
Can LOO or other cross-validation be used with time series?



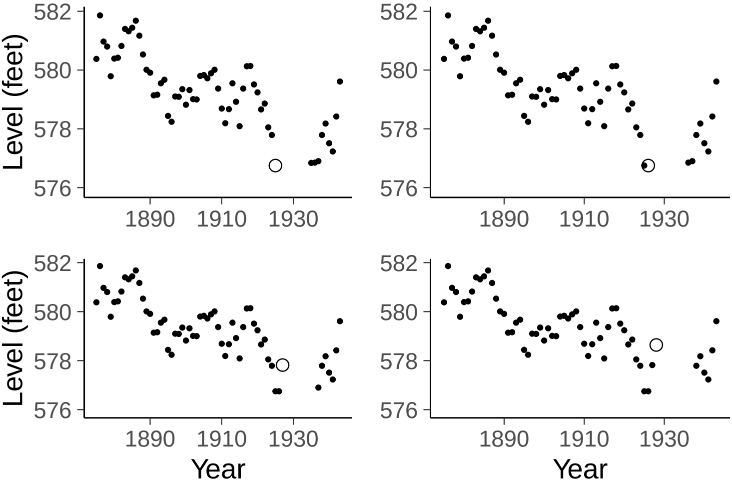
Leave-one-out cross-validation is ok for assessing conditional model



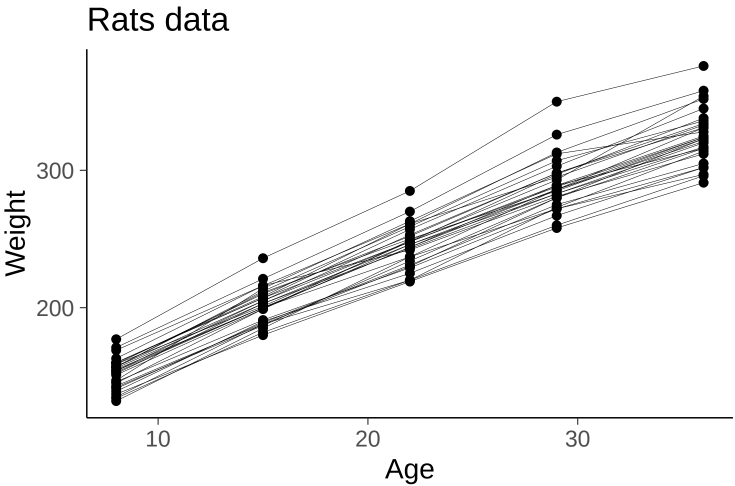
leave-future-out cross-validation is better for predicting future



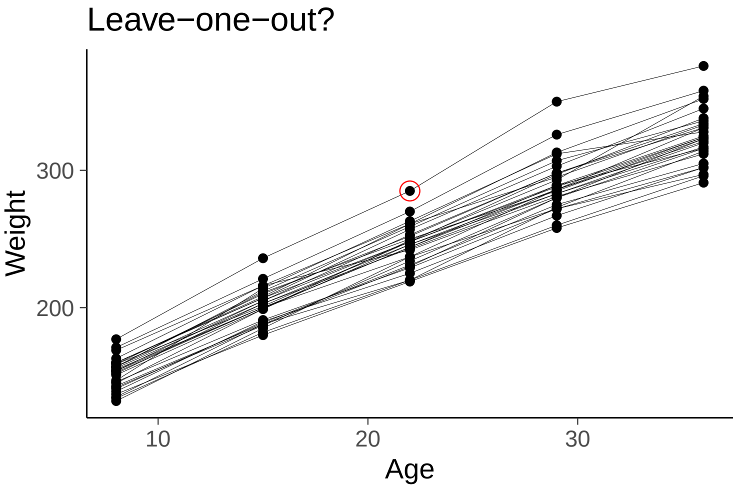
m-step-ahead cross-validation is better for predicting further future



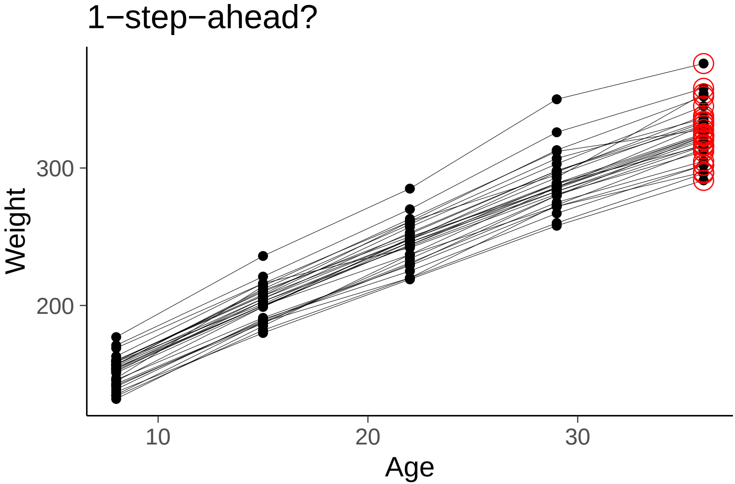
m-step-ahead leave-a-block-out cross-validation



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

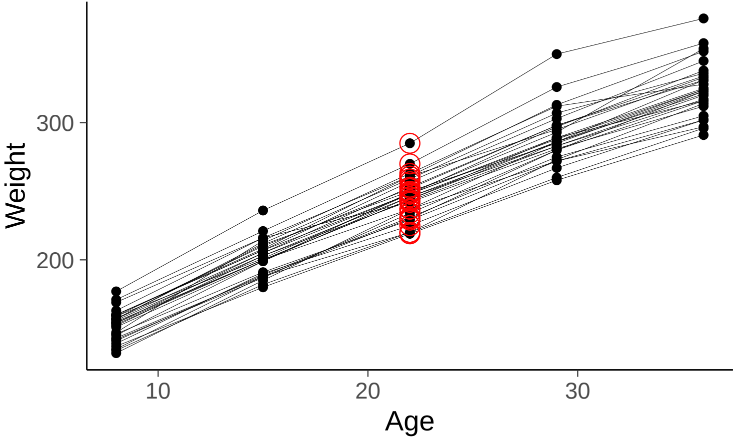


Yes!



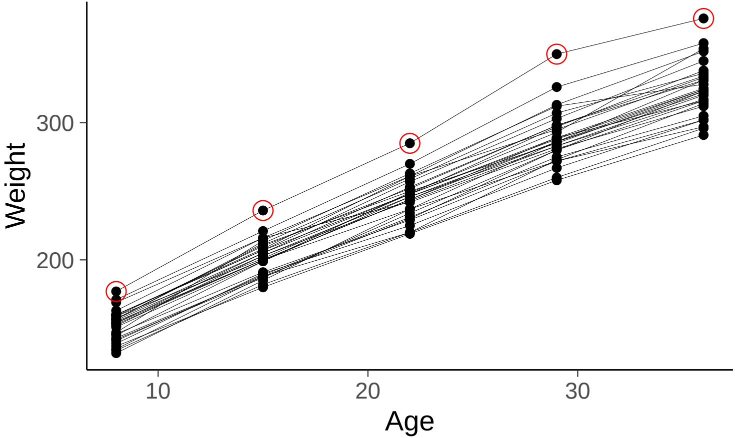
Yes!

Leave−one−time−point−out?



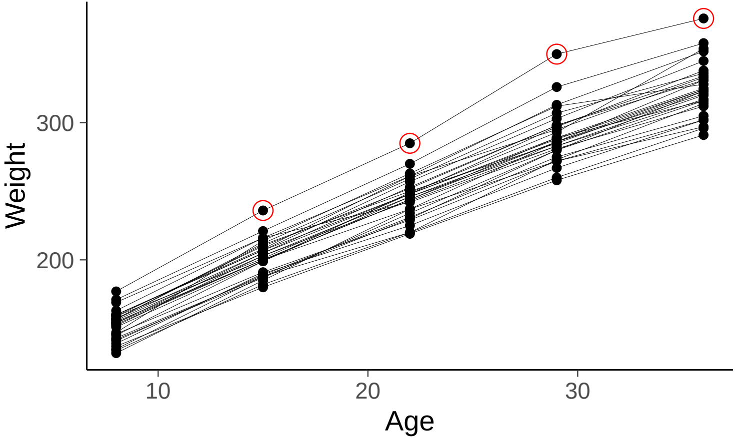
Yes!

Leave−one−rat−out?



Yes!

Predict given initial weight?



Yes!

Summary of data generating mechanisms and prediction tasks

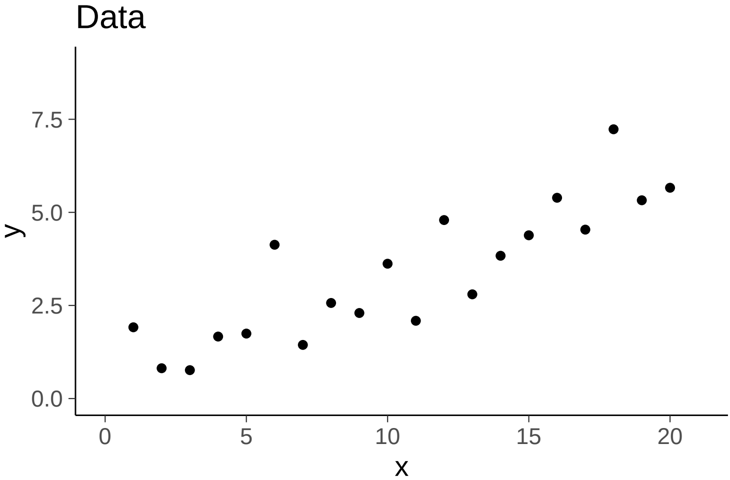
- You have to make some assumptions on data generating mechanism
- Use the knowledge of the prediction task if available
- Cross-validation can be used to analyse different parts, even if there is no clear prediction task

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

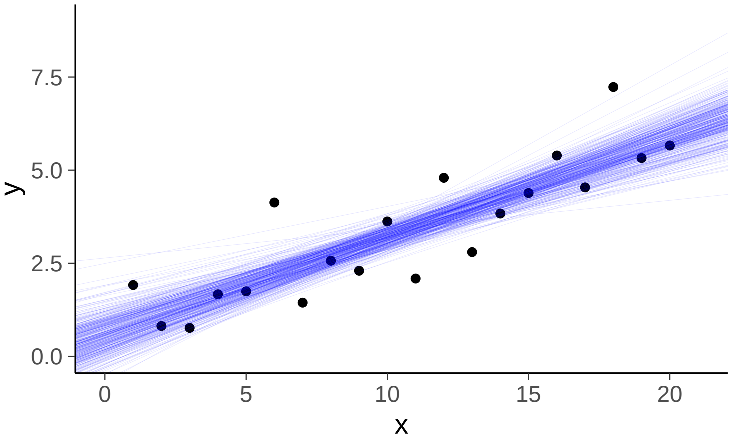
Fast cross-validation

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

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

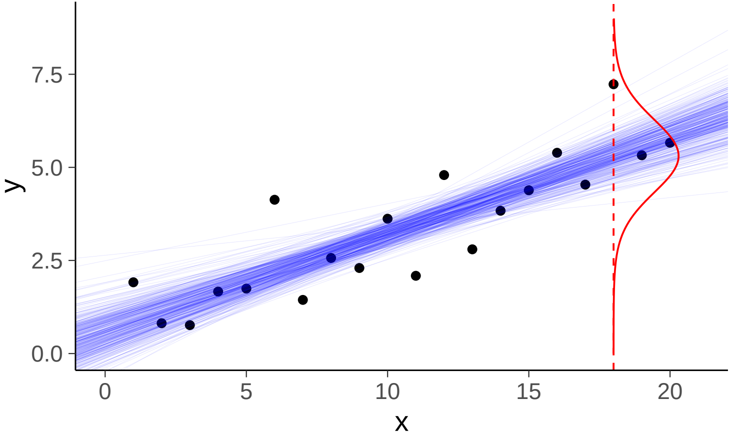

Data
y
7.5
5.0
2.5
0.0
0
5
10
15
20
x

Posterior draws



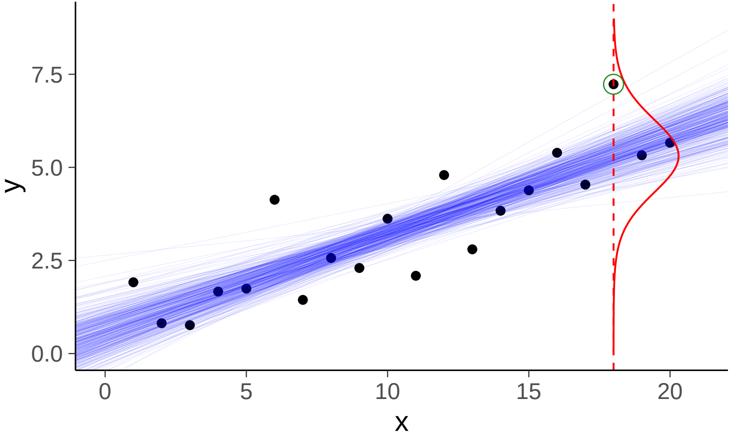
$\theta^{(s)} \sim p(\theta|x, y)$

Posterior predictive distribution



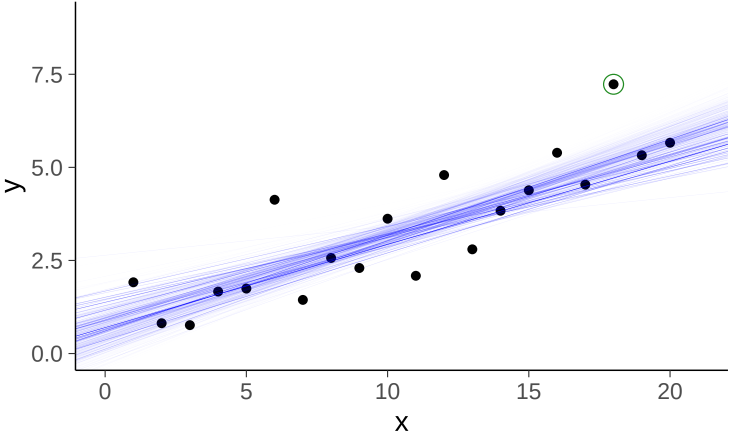
$$\theta^{(s)} \sim p(\theta|x,y), \quad p(\tilde{y}|\tilde{x},x,y) \approx \frac{1}{S}\sum_{s=1}^{S} p(\tilde{y}|\tilde{x},\theta^{(s)})$$

Posterior predictive distribution

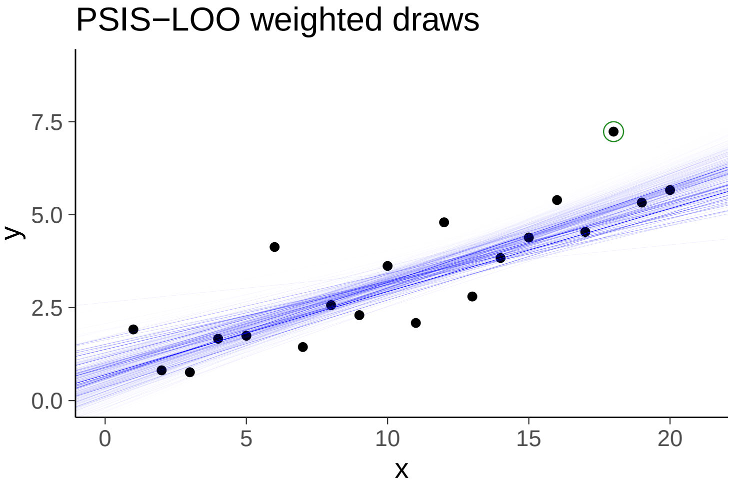


$$\theta^{(s)} \sim p(\theta|x,y), \quad p(\tilde{y}|\tilde{x},x,y) \approx \frac{1}{S}\sum_{s=1}^{S} p(\tilde{y}|\tilde{x},\theta^{(s)})$$

PSIS−LOO weighted draws

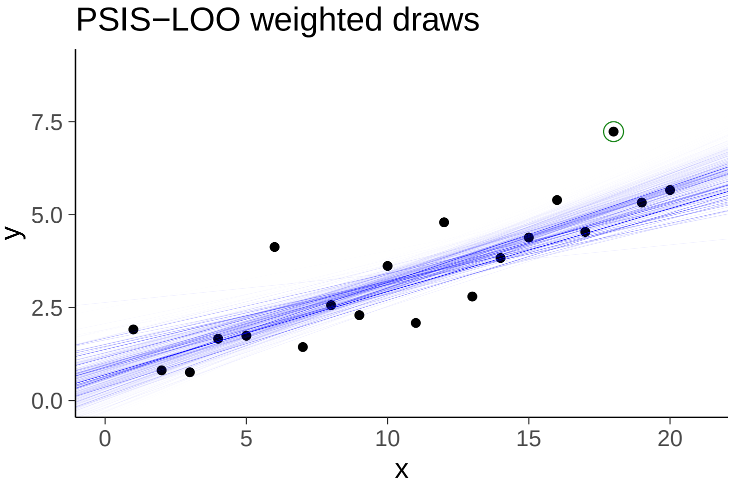


$$\theta^{(s)} \sim p(\theta|x, y)$$
$$r_i^{(s)} = p(\theta^{(s)}|x_{-i}, y_{-i})/p(\theta^{(s)}|x, y)$$



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

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

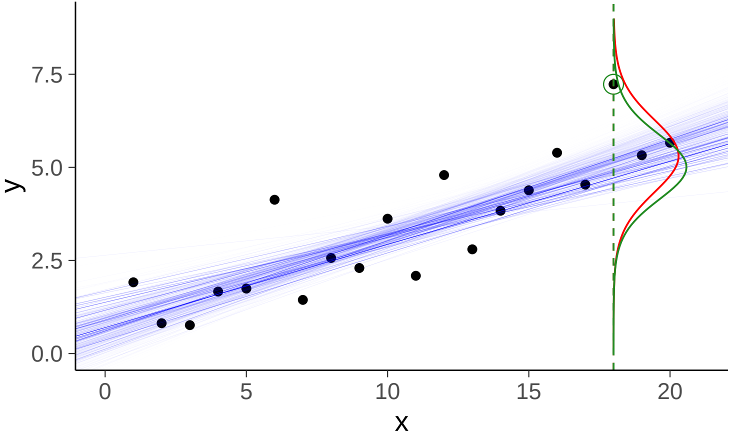


$$\theta^{(s)} \sim p(\theta|x,y)$$

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

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

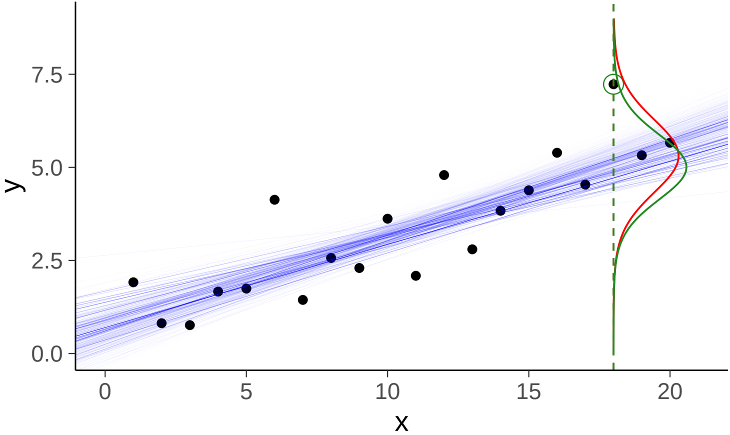
PSIS−LOO weighted predictive distribution



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

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

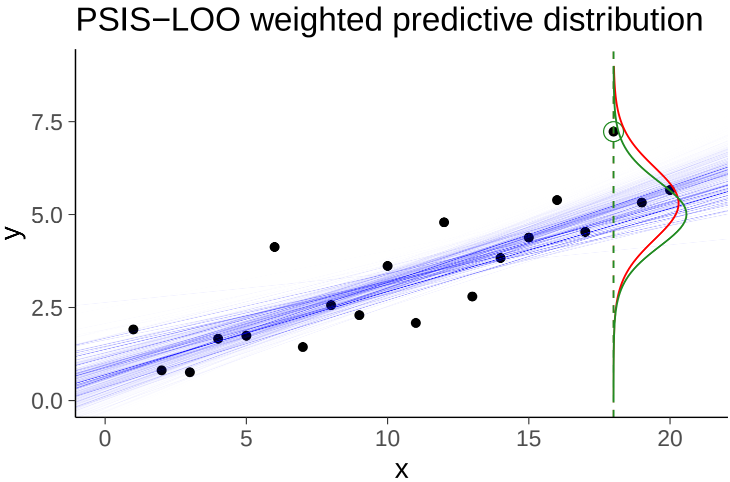
PSIS−LOO weighted predictive distribution



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

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

$$p(y_i|x_i,x_{-i},y_{-i}) \approx \textstyle\sum_{s=1}^{S}[w_i^{(s)} p(y_i|x_i,\theta^{(s)})]$$

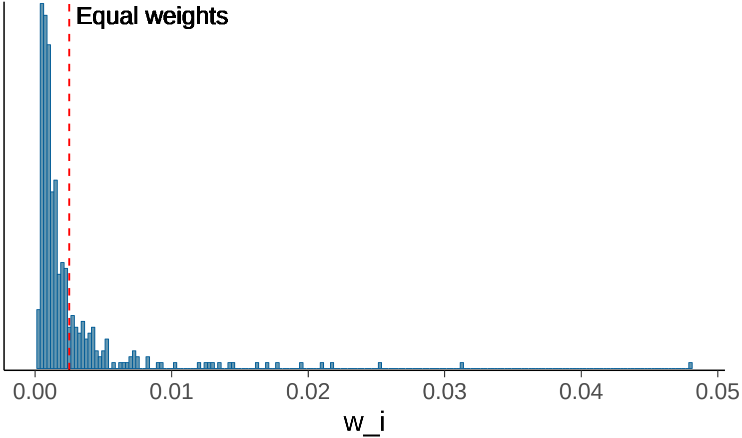


$\theta^{(s)} \sim p(\theta|x, y)$

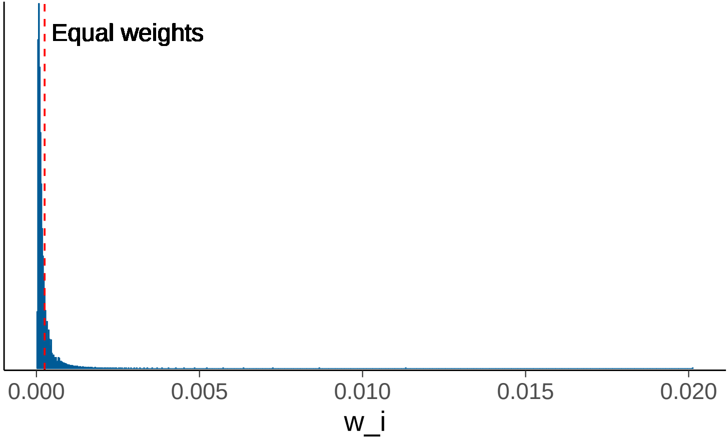
$r_i^{(s)} = p(\theta^{(s)}|x_{-i}, y_{-i})/p(\theta^{(s)}|x, y) \propto 1/p(y_i|x_i, \theta^{(s)})$

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

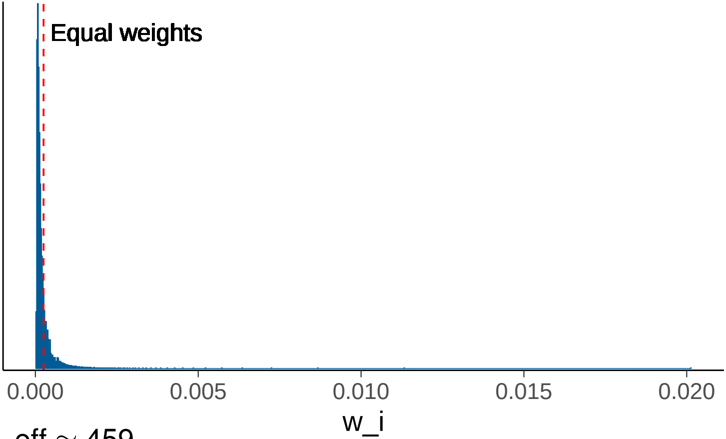
400 importance weights for leave−18th−out



4000 importance weights for leave−18th−out

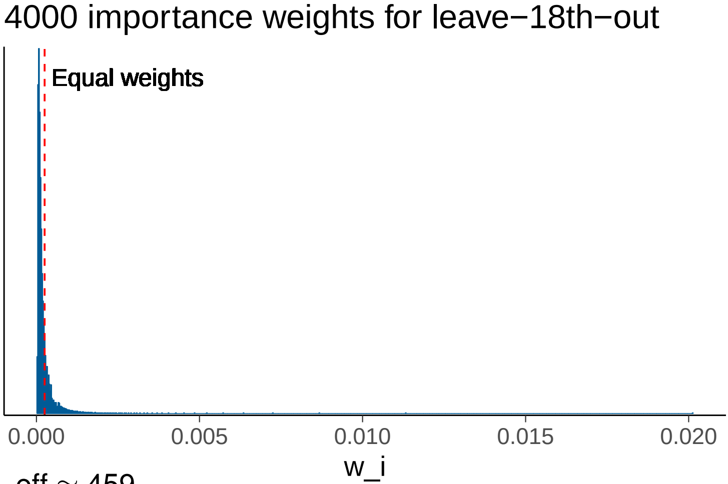


4000 importance weights for leave−18th−out



n_eff ≈ 459

see Vehtari, Gelman & Gabry (2017b)

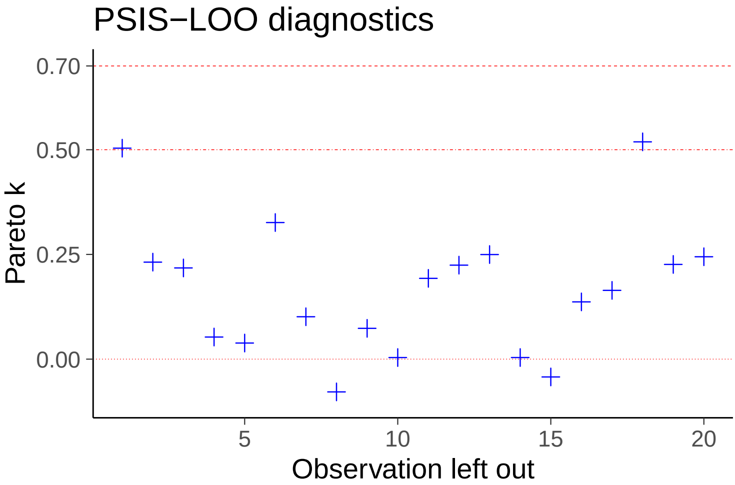


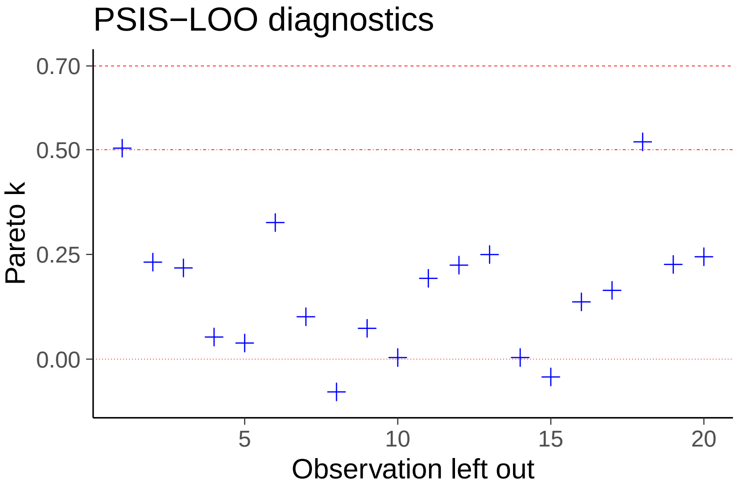
n_eff $\approx$ 459

Pareto $\hat{k} \approx 0.52$

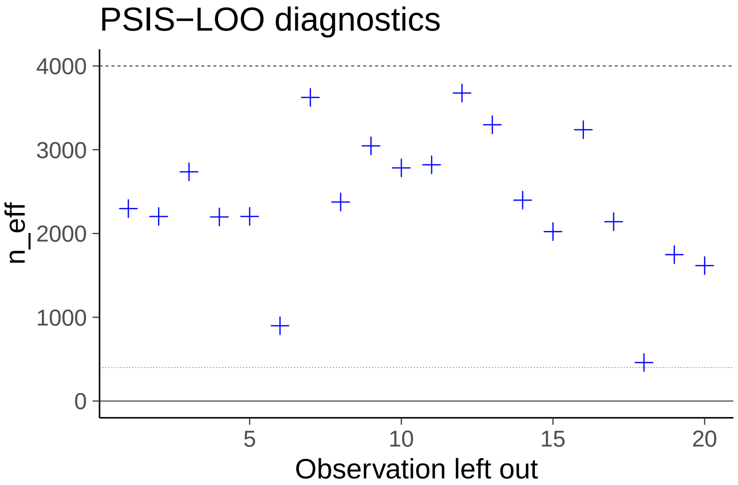
- Pareto $\hat{k}$ estimates the tail shape which determines the convergence rate of PSIS. Less than 0.7 is ok.

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




```
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>
```



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 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);
}
```

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);
}
```

- RStanARM and BRMS compute log_lik by default

Pareto smoothed importance sampling LOO

- PSIS-LOO for hierarchical models
    - 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

- PSIS-LOO for hierarchical models
  - leave-one-group out is challenging for PSIS-LOO

    see Merkel, Furr and Rabe-Hesketh (2018) for an approach using quadrature integration
- PSIS-LOO for non-factorizable models
  - mc-stan.org/loo/articles/loo2-non-factorizable.html

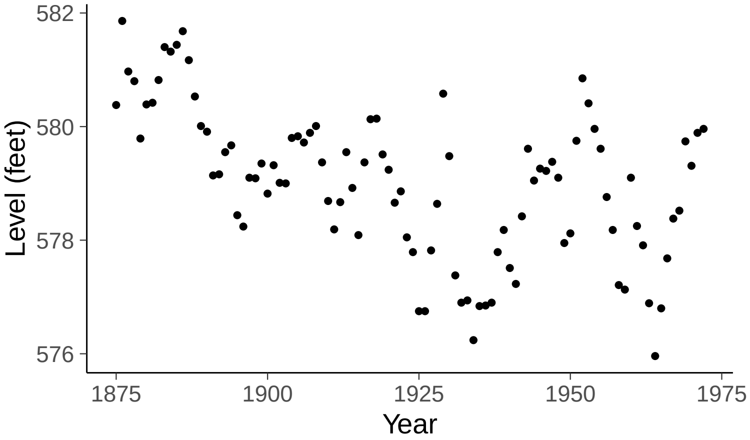
Pareto smoothed importance sampling LOO

- PSIS-LOO for hierarchical models
  - leave-one-group out is challenging for PSIS-LOO

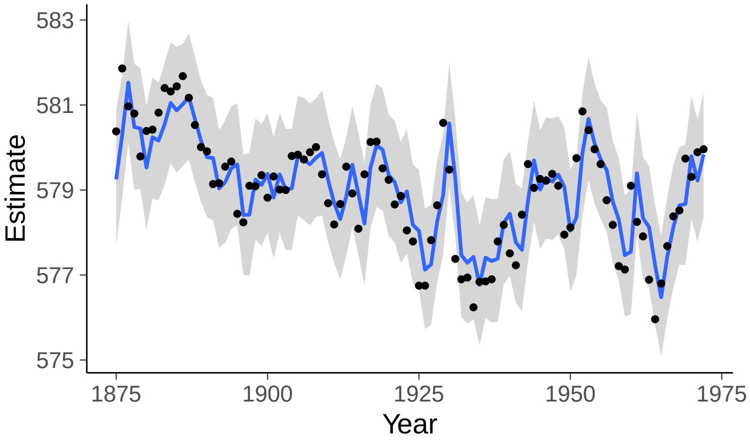
    see Merkel, Furr and Rabe-Hesketh (2018) for an approach using quadrature integration
- PSIS-LOO for non-factorizable models
  - mc-stan.org/loo/articles/loo2-non-factorizable.html
- PSIS-LOO for time series
  - Approximate leave-future-out cross-validation

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

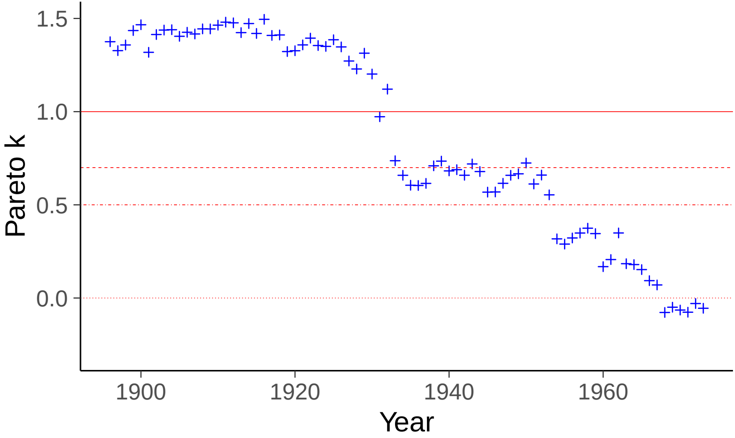
Data

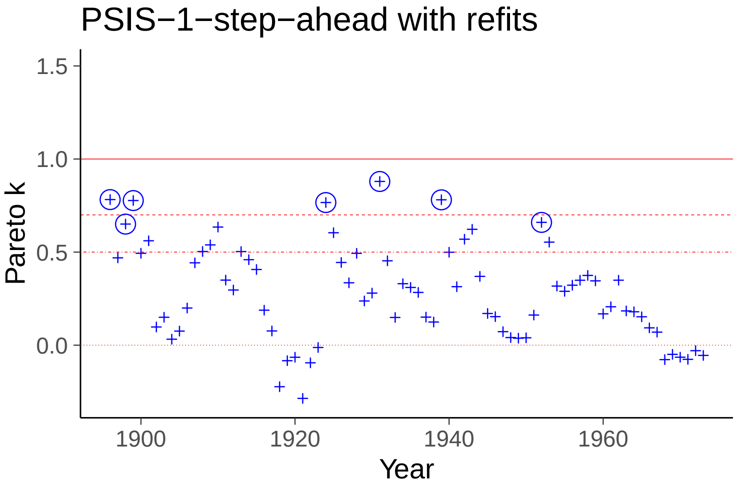


AR−2 prediction with 95% interval



PSIS−1−step−ahead



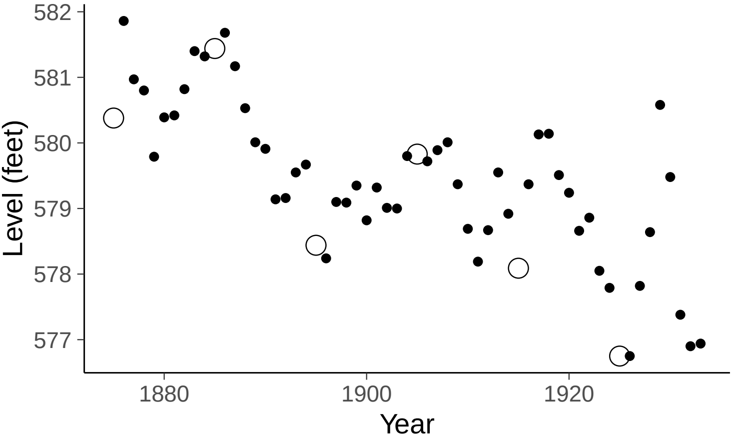


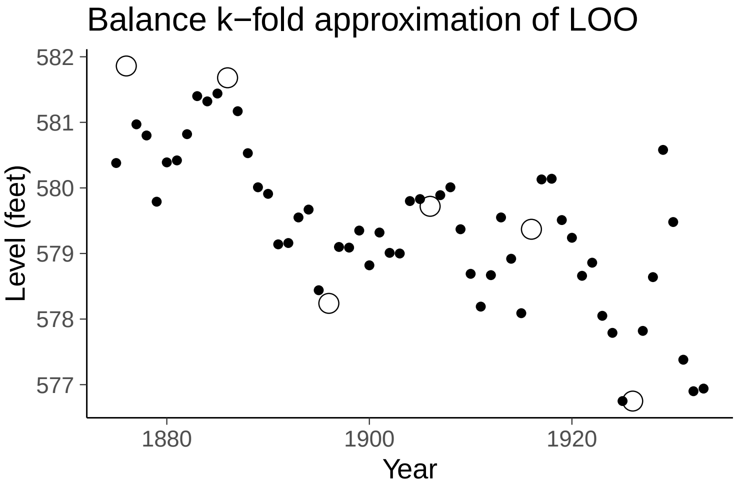
mc-stan.org/loo/articles/loo2-lfo.html

K-fold cross-validation

- K-fold cross-validation can approximate LOO
  - all uses for LOO
- K-fold cross-validation can be used for hierarchical models
  - good for leave-one-group-out
- K-fold cross-validation can be used for time series
  - with leave-block-out

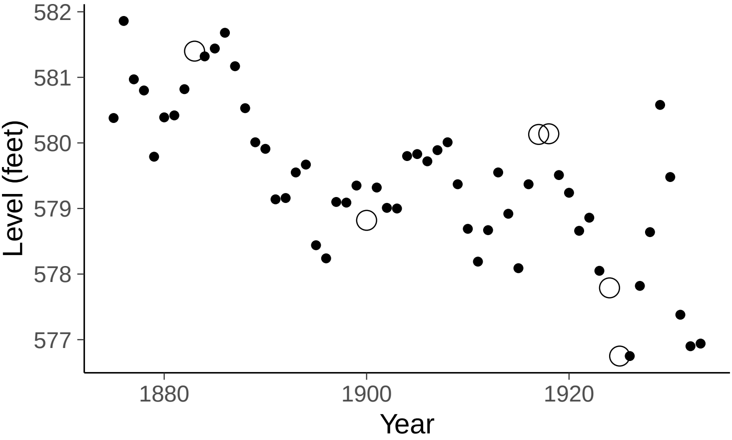
Balance k−fold approximation of LOO



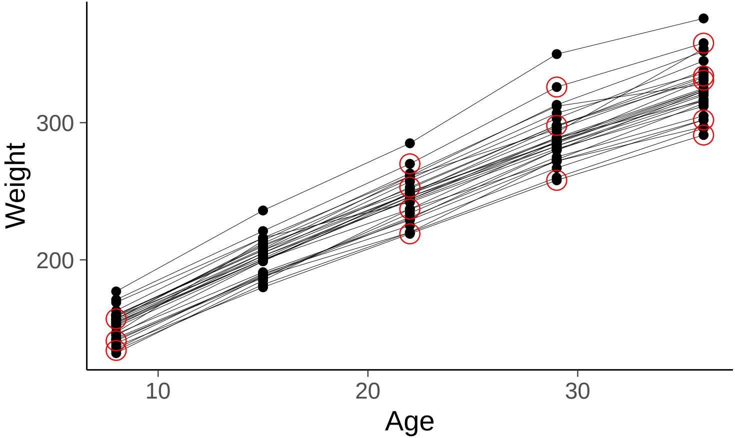


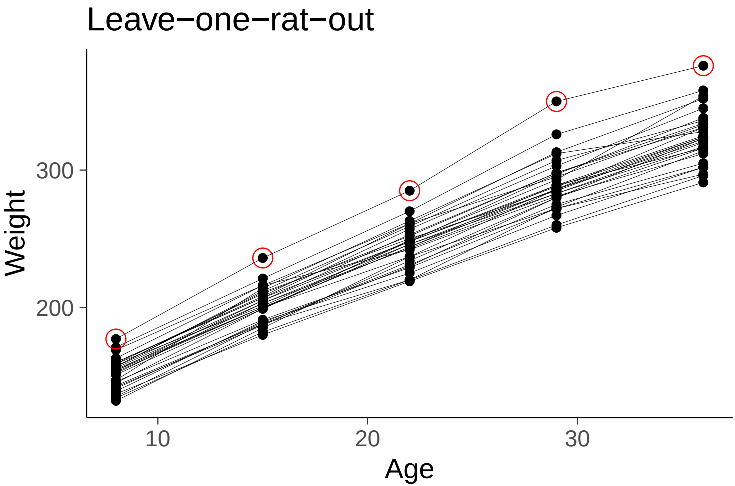
Balance k−fold approximation of LOO
582
581
580
579
578
577
Level (feet)
1880
1900
1920
Year

Random k−fold approximation of LOO



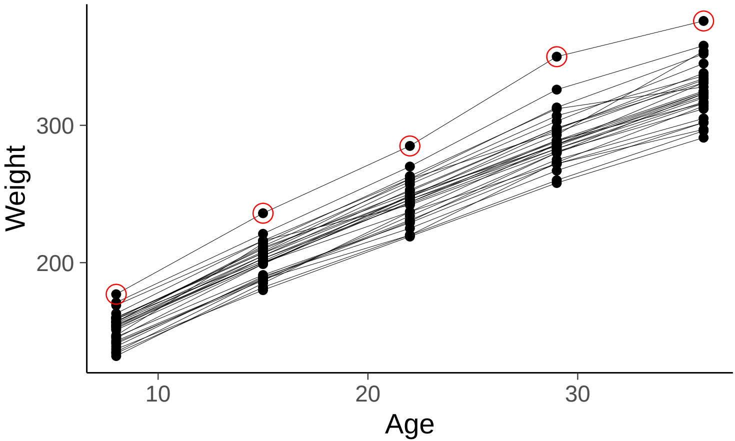
Random kfold approximation of LOO





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

Leave−one−rat−out



kfold_split_random()

kfold_split_balanced()

kfold_split_stratified()

WAIC vs PSIS-LOO

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

- WAIC has same assumptions as LOO

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

- WAIC has same assumptions as LOO
- PSIS-LOO is more accurate

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

- WAIC has same assumptions as LOO
- PSIS-LOO is more accurate
- PSIS-LOO has much better diagnostics

see Vehtari, Gelman & Gabry (2017a)

WAIC vs PSIS-LOO

- WAIC has same assumptions as LOO
- PSIS-LOO is more accurate
- PSIS-LOO has much better diagnostics
- 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

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

see Vehtari, Gelman & Gabry (2017a)

*IC

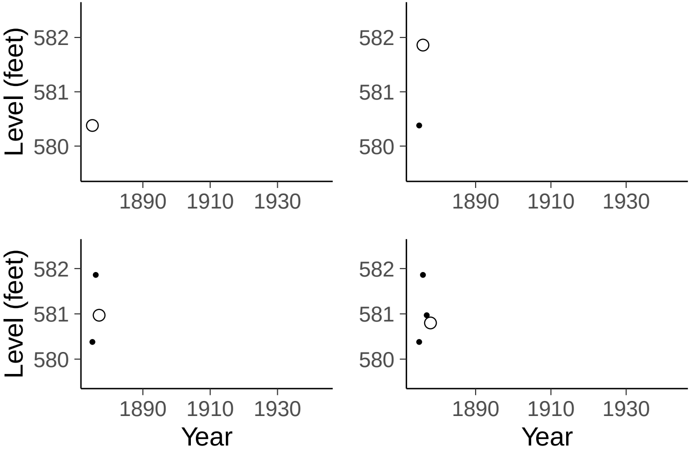
- AIC uses maximum likelihood estimate for prediction
- DIC uses posterior mean for prediction
- BIC is an approximation for marginal likelihood
- TIC, NIC, RIC, PIC, BPIC, QIC, AICc, ...

Marginal likelihood / Bayes factor

- Like leave-future-out 1-step-ahead corss-validation but starting with 0 observations

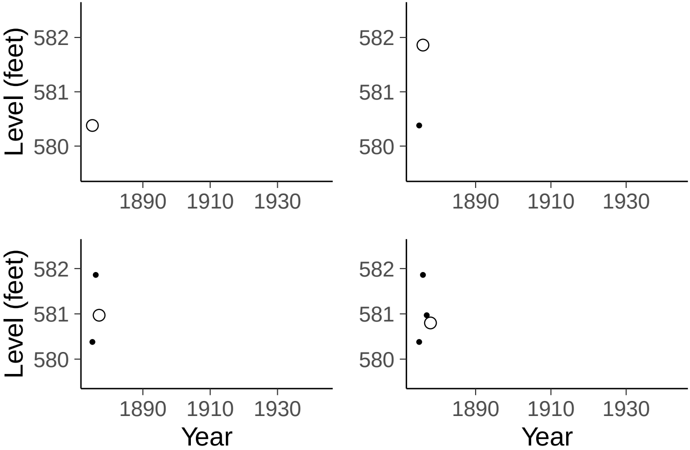
Marginal likelihood / Bayes factor

- Like leave-future-out 1-step-ahead corss-validation but starting with 0 observations



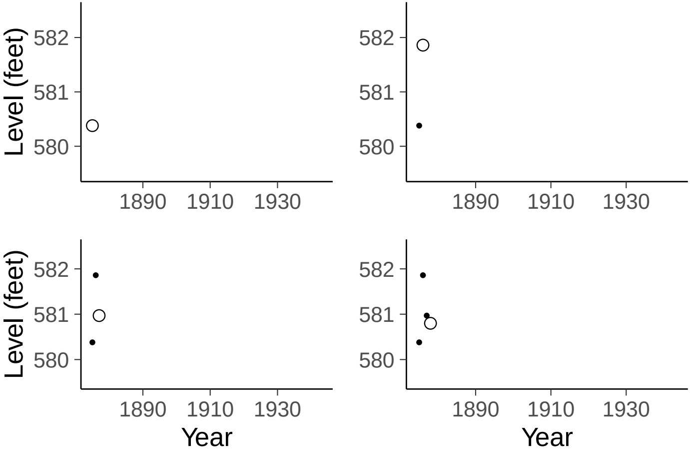
Marginal likelihood / Bayes factor

- Like leave-future-out 1-step-ahead corss-validation but starting with 0 observations
  - which makes it very sensitive to prior



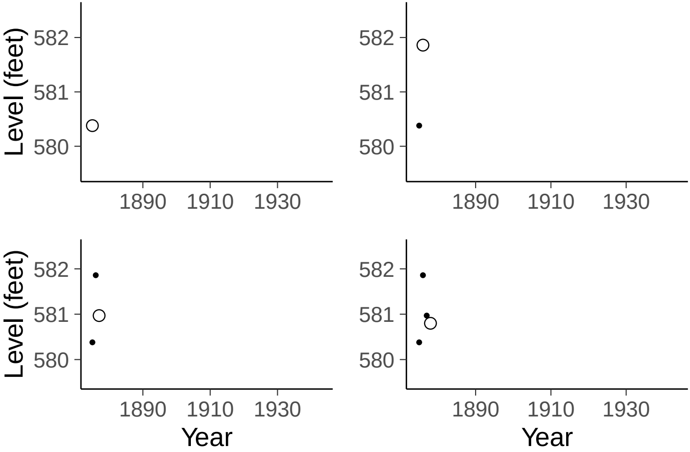
Marginal likelihood / Bayes factor

- Like leave-future-out 1-step-ahead corss-validation but starting with 0 observations
  - which makes it very sensitive to prior and
  - unstable in case of misspecified models



Marginal likelihood / Bayes factor

- Like leave-future-out 1-step-ahead corss-validation but starting with 0 observations
  - which makes it very sensitive to prior and
  - unstable in case of misspecified models also asymptotically



Cross-validation for model assessment

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

Cross-validation for model assessment

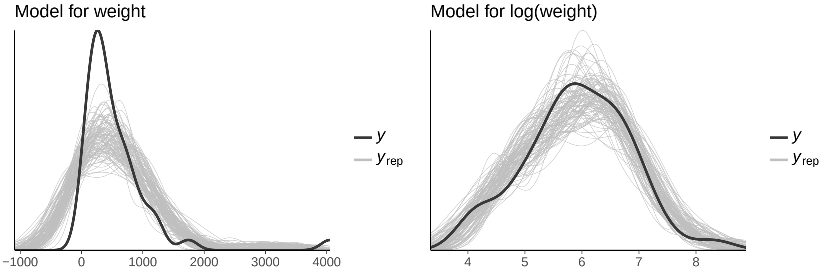
- CV is good for model assessment when application specific utility/cost functions are used
  - e.g. 90% absolute error
- Also useful in model checking in similar way as posterior predictive checking (PPC)
  - model misspecification diagnostics (e.g. Pareto-*k* and p_loo)
  - checking calibration of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

see demos avehtari.github.io/modelselection/

Sometimes cross-validation is not needed

Sometimes cross-validation is not needed

- Posterior predictive checking is often sufficient

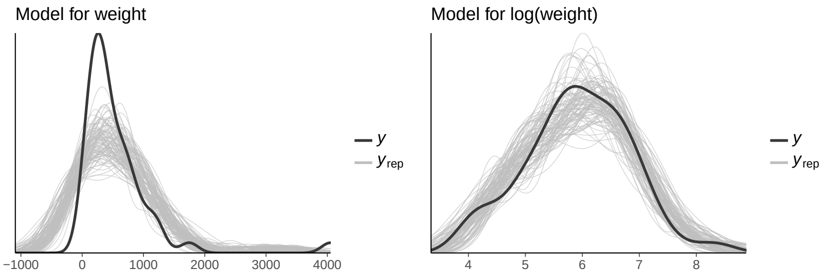


Predicting the yields of mesquite bushes.

Gelman, Hill & Vehtari (2019): Regression and Other Stories, Chapter 11.

Sometimes cross-validation is not needed

- Posterior predictive checking is often sufficient



Predicting the yields of mesquite bushes.

Gelman, Hill & Vehtari (2019): Regression and Other Stories, Chapter 11.

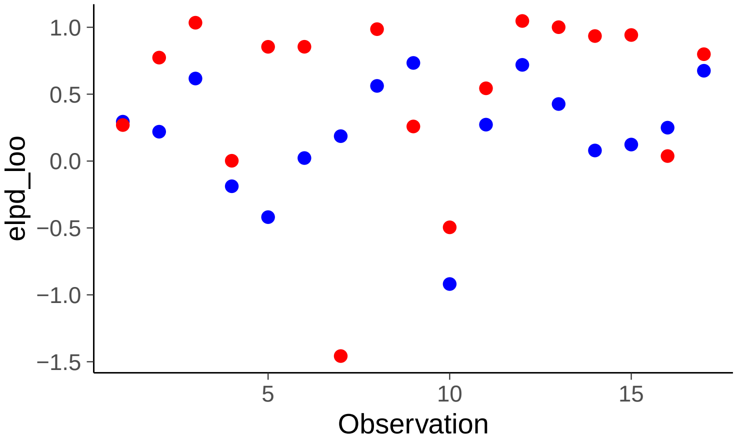
- BDA3, Chapter 6
- Gabry, Simpson, Vehtari, Betancourt, Gelman (2018). Visualization in Bayesian workflow. JRSS A, preprint arXiv:1709.01449
- mc-stan.org/bayesplot/articles/graphical-ppcs.html
- betanalpha.github.io/assets/case_studies/principled_bayesian_workflow.html

Model comparison

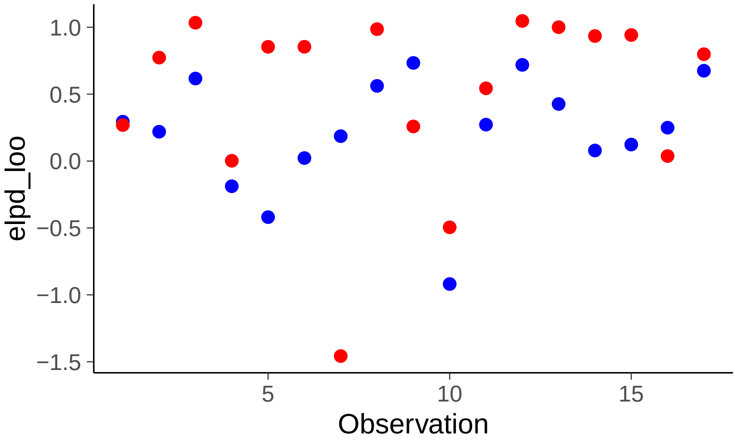
- "A popular hypothesis has it that primates with larger brains produce more energetic milk, so that brains can grow quickly" (from Statistical Rethinking)
  - Model 1: formula = kcal.per.g $\sim$ neocortex
  - Model 2: formula = kcal.per.g $\sim$ neocortex + log(mass)

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

Pointwise comparison LOO models: Model 1

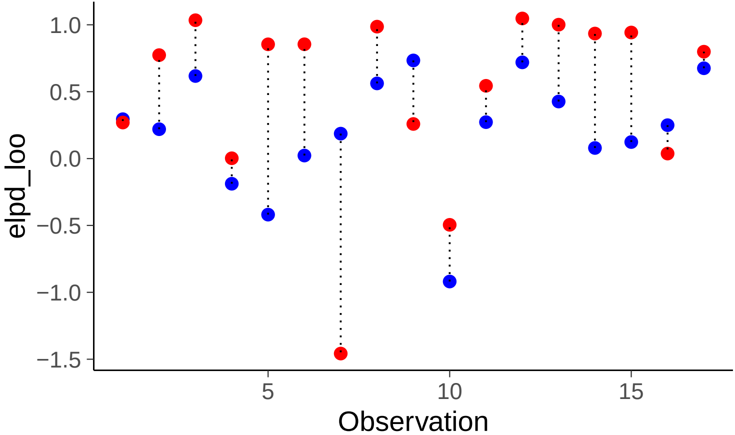


Pointwise comparison LOO models: Model 1

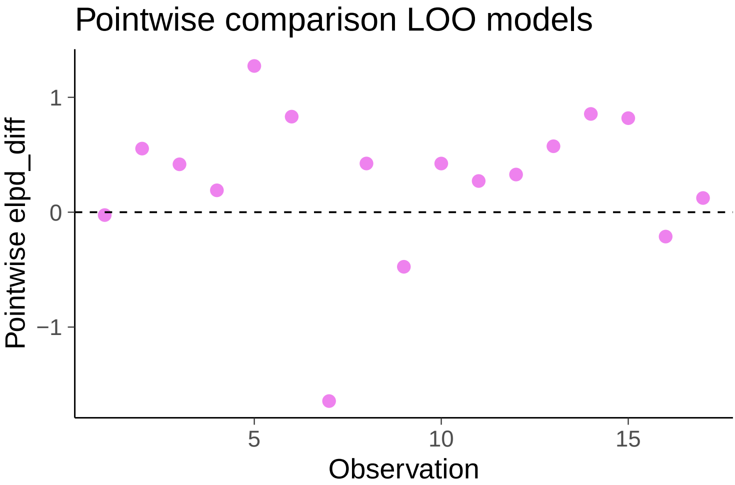


Model 1 elpd_loo $\approx$ 3.7, SE=1.8
Model 2 elpd_loo $\approx$ 8.4, SE=2.8

Pointwise comparison LOO models: Model 1



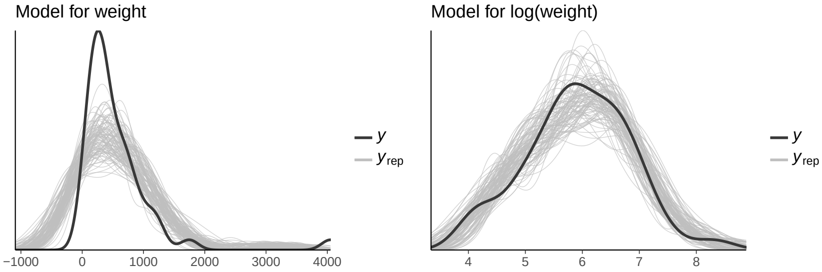
Model 1 elpd_loo $\approx$ 3.7, SE=1.8
Model 2 elpd_loo $\approx$ 8.4, SE=2.8




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

elpd_diff        se
      4.7       2.7
```

Sometimes cross-validation is not needed

- Posterior predictive checking is often sufficient



Predicting the yields of mesquite bushes.

Gelman, Hill & Vehtari (2019): Regression and Other Stories, Chapter 11.

- BDA3, Chapter 6
- Gabry, Simpson, Vehtari, Betancourt, Gelman (2018). Visualization in Bayesian workflow. JRSS A, preprint arXiv:1709.01449
- mc-stan.org/bayesplot/articles/graphical-ppcs.html
- betanalpha.github.io/assets/case_studies/principled_bayesian_workflow.html

Sometimes cross-validation is not needed

- For some very simple cases you may assume that true model is included in the list of models considered (M-closed)

Sometimes cross-validation is not needed

- For some very simple cases you may assume that true model is included in the list of models considered (*M*-closed)
  - see predictive model selection in *M*-closed case by San Martini and Spezzaferri (1984)

Sometimes cross-validation is not needed

- For some very simple cases you may assume that true model is included in the list of models considered (*M*-closed)
  - see predictive model selection in *M*-closed case by San Martini and Spezzaferri (1984)
  - but you should not force your design of experiment or analysis to stay in the simplified world

Sometimes cross-validation is not needed

- For some very simple cases you may assume that true model is included in the list of models considered (*M*-closed)
  - see predictive model selection in *M*-closed case by San Martini and Spezzaferri (1984)
  - but you should not force your design of experiment or analysis to stay in the simplified world
- For fully non-parametric models you may assume that true model is included in the list of models considered (*M*-closed)

Sometimes cross-validation is not needed

- For some very simple cases you may assume that true model is included in the list of models considered (*M*-closed)
  - see predictive model selection in *M*-closed case by San Martini and Spezzaferri (1984)
  - but you should not force your design of experiment or analysis to stay in the simplified world
- For fully non-parametric models you may assume that true model is included in the list of models considered (*M*-closed)
  - related to talk by Chris Holmes
  - see Vehtari & Ojanen (2012) for earlier references

Sometimes cross-validation is not needed

- For some very simple cases you may assume that true model is included in the list of models considered (M-closed)
  - see predictive model selection in M-closed case by San Martini and Spezzaferri (1984)
  - but you should not force your design of experiment or analysis to stay in the simplified world
- For fully non-parametric models you may assume that true model is included in the list of models considered (M-closed)
  - related to talk by Chris Holmes
  - see Vehtari & Ojanen (2012) for earlier references
  - posterior convergence rate can be slow for fully non-parametric models

Sometimes cross-validation is not needed

- For some very simple cases you may assume that true model is included in the list of models considered (M-closed)
  - see predictive model selection in M-closed case by San Martini and Spezzaferri (1984)
  - but you should not force your design of experiment or analysis to stay in the simplified world
- For fully non-parametric models you may assume that true model is included in the list of models considered (M-closed)
  - related to talk by Chris Holmes
  - see Vehtari & Ojanen (2012) for earlier references
  - posterior convergence rate can be slow for fully non-parametric models
- In nested case, often easier and more accurate to analyse posterior distribution of more complex model directly
  avehtari.github.io/modelselection/betablockers.html

What if one is not clearly better than others?

What if one is not clearly better than others?

- Continuous expansion including all models?
  - and then analyse the posterior distribution directly
    avehtari.github.io/modelselection/betablockers.html
  - 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?

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

What if one is not clearly better than others?

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

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

Bayesian stacking

- Consider the model averaging as a decision problem with aim of maximizing the predictive performance

Bayesian stacking

- Consider the model averaging as a decision problem with aim of maximizing the predictive performance
- Maximize the scoring rule of the predictive distribution for future $\tilde{y}$

$$\max_{w} S\Big(\sum_{k=1}^{K} w_k p(\tilde{y}|x, y, M_k), p_t(\tilde{y})\Big),$$

Bayesian stacking

- Consider the model averaging as a decision problem with aim of maximizing the predictive performance
- Maximize the scoring rule of the predictive distribution for future $\tilde{y}$

$$\max_w S\Big(\sum_{k=1}^{K} w_k p(\tilde{y}|x, y, M_k), p_t(\tilde{y})\Big),$$

- As we don't know $p_t(\tilde{y})$, we approximate with LOO

Bayesian stacking

- Consider the model averaging as a decision problem with aim of maximizing the predictive performance
- Maximize the scoring rule of the predictive distribution for future $\tilde{y}$

$$\max_w S\Big(\sum_{k=1}^{K} w_k p(\tilde{y}|x, y, M_k), p_t(\tilde{y})\Big),$$

- As we don't know $p_t(\tilde{y})$, we approximate with LOO
- We define the stacking weights as the solution to the following optimization problem:

$$\max_w \frac{1}{n}\sum_{i=1}^{n} S\Big(\sum_{k=1}^{K} w_k \hat{p}(y_i|x_{-i}, y_{-i}, M_k)\Big),$$

$$s.t. \quad w_k \geq 0, \quad \sum_{k=1}^{K} w_k = 1.$$

Bayesian stacking

- The combined estimation of the predictive density is

$$\hat{p}(\tilde{y}|x, y) = \sum_{k=1}^{K} \hat{w}_k p(\tilde{y}|x, y, M_k).$$

Bayesian stacking

- The combined estimation of the predictive density is

$$\hat{p}(\tilde{y}|x, y) = \sum_{k=1}^{K} \hat{w}_k p(\tilde{y}|x, y, M_k).$$

- When using log-score (corresponding to Kullback-Leibler divergence), we call this stacking of predictive distributions:

$$\max_{w} \frac{1}{n} \sum_{i=1}^{n} \log \sum_{k=1}^{K} w_k p(y_i|x_{-i}, y_{-i}, M_k),$$

$$s.t. \quad w_k \geq 0, \quad \sum_{k=1}^{K} w_k = 1.$$

Bayesian stacking

- The combined estimation of the predictive density is

$$\hat{p}(\tilde{y}|x, y) = \sum_{k=1}^{K} \hat{w}_k p(\tilde{y}|x, y, M_k).$$

- When using log-score (corresponding to Kullback-Leibler divergence), we call this stacking of predictive distributions:

$$\max_{w} \frac{1}{n} \sum_{i=1}^{n} \log \sum_{k=1}^{K} w_k p(y_i|x_{-i}, y_{-i}, M_k),$$

$$s.t. \quad w_k \geq 0, \quad \sum_{k=1}^{K} w_k = 1.$$

- We can approximate $p(y_i|x_{-i}, y_{-i}, M_k)$ with PSIS-LOO

Bayesian stacking

- The combined estimation of the predictive density is

$$\hat{p}(\tilde{y}|x, y) = \sum_{k=1}^{K} \hat{w}_k p(\tilde{y}|x, y, M_k).$$

- When using log-score (corresponding to Kullback-Leibler divergence), we call this stacking of predictive distributions:

$$\max_{w} \frac{1}{n} \sum_{i=1}^{n} \log \sum_{k=1}^{K} w_k p(y_i|x_{-i}, y_{-i}, M_k),$$

$$s.t. \quad w_k \geq 0, \quad \sum_{k=1}^{K} w_k = 1.$$

- We can approximate $p(y_i|x_{-i}, y_{-i}, M_k)$ with PSIS-LOO
- Other cross-validation structures can be used, too

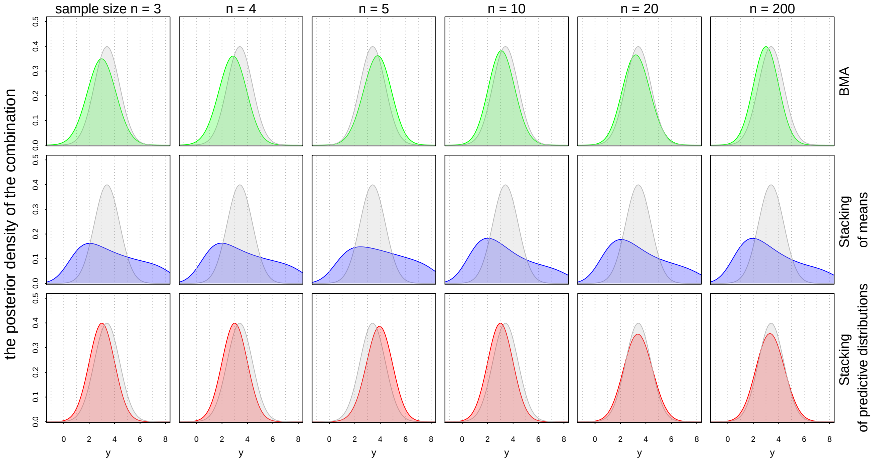
Gaussian mixture example

$y \sim \mathrm{N}(3.4, 1), \quad p_k = \mathrm{N}(k, 1)$ with $k = 1, \ldots, 8$

Gaussian mixture example

$y \sim \mathrm{N}(3.4, 1), \quad p_k = \mathrm{N}(k, 1)$ with $k = 1, \ldots, 8$

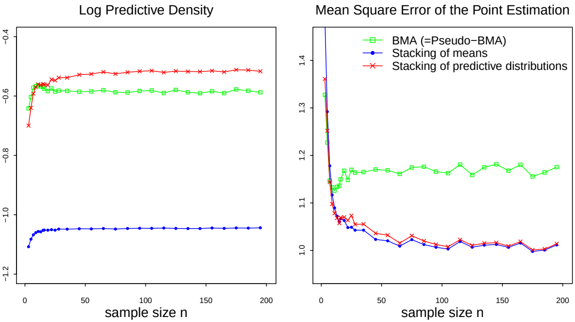
Model averaged predictive distributions



Gaussian mixture example

$y \sim \mathrm{N}(3.4, 1), \quad p_k = \mathrm{N}(k, 1)$ with $k = 1, \ldots, 8$

(a, b) Stacking of predictive distributions vs. BMA

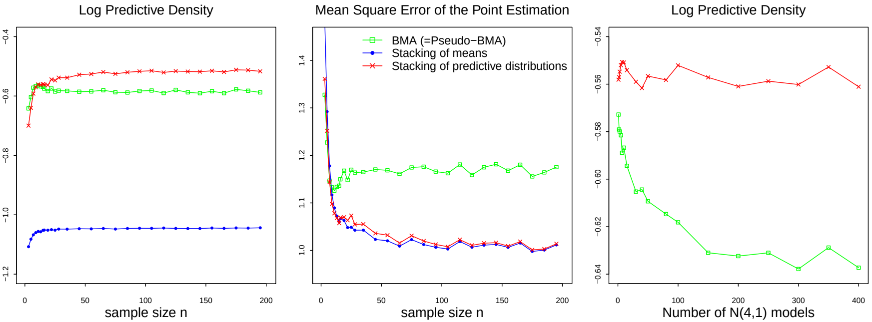


(a) (b)

Gaussian mixture example

$y \sim \mathrm{N}(3.4, 1), \quad p_k = \mathrm{N}(k, 1)$ with $k = 1, \ldots, 8$

(a, b) Stacking of predictive distributions vs. BMA



(a) (b) (c)

(c) Dilutation of prior by adding copies of $\mathrm{N}(4, 1)$ to the model space

Linear subset regression example k

$y \sim \mathrm{N}(\mu, 1), \quad \mu = \beta_1 X_1 + \ldots \beta_{15} X_{15}$

$\beta_j = \gamma\left((1_{|j-4|<h}(h - |j-4|)^2 + (1_{|j-8|<h})(h - |j-8|)^2 + (1_{|j-12|<h})(h - |j-12|)^2\right)$

Linear subset regression example k

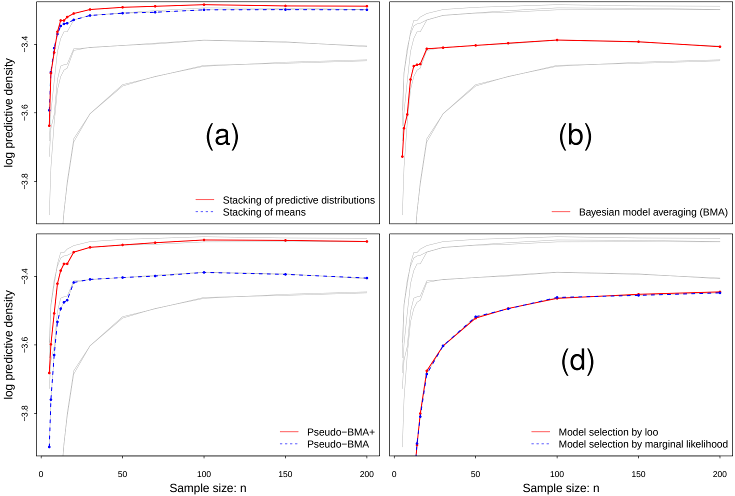
$y \sim \mathrm{N}(\mu, 1), \quad \mu = \beta_1 X_1 + \ldots \beta_{15} X_{15}$

Non-nested M-open case with $M_k : \mathrm{N}(\beta_k X_k, \sigma)$

Linear subset regression example *k*

$y \sim \mathrm{N}(\mu, 1), \quad \mu = \beta_1 X_1 + \ldots \beta_{15} X_{15}$

Non-nested M-open case with $M_k : \mathrm{N}(\beta_k X_k, \sigma)$



(a) Stacking, (b) BMA, (d) model selection by LOO and BF

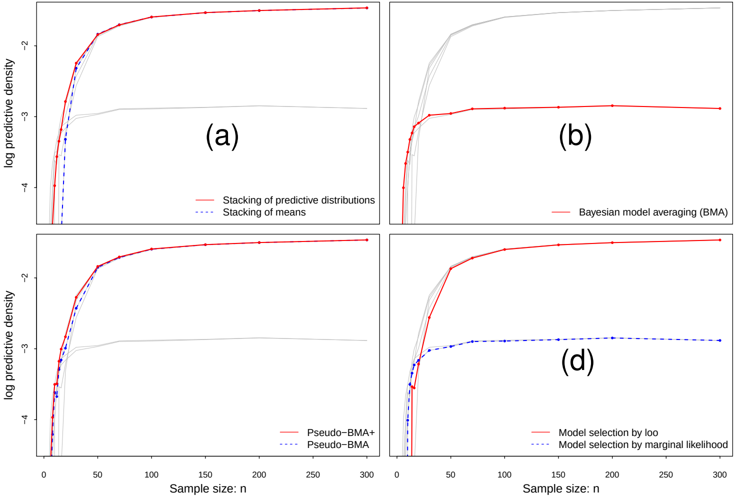
Linear subset regression example 1 : k

$y \sim \mathrm{N}(\mu, 1), \quad \mu = \beta_1 X_1 + \ldots \beta_{15} X_{15}$
Nested M-closed case with $M_k : \mathrm{N}(\sum_{j=1}^{k} \beta_j X_j, \sigma)$

Linear subset regression example 1 : k

$y \sim \mathrm{N}(\mu, 1), \quad \mu = \beta_1 X_1 + \ldots \beta_{15} X_{15}$

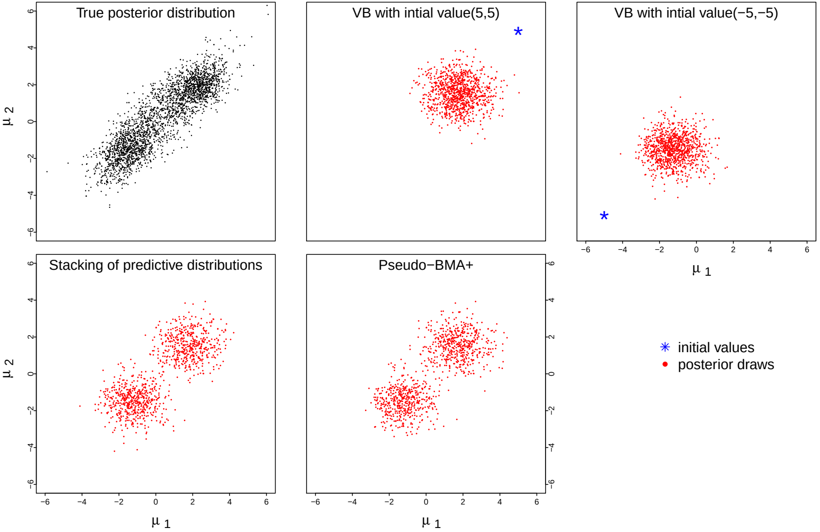
Nested M-closed case with M_k : $\mathrm{N}(\sum_{j=1}^{k} \beta_j X_j, \sigma)$



(a) Stacking, (b) BMA, (d) model selection by LOO and BF

Variational multimodal example

Stacking of predictive distributions can be helpful also in case of multimodal posteriors



Bayesian stacking

- In M-open case works better than BMA
- In M-closed case can have a better small sample performance than BMA

Bayesian stacking

- In M-open case works better than BMA
- In M-closed case can have a better small sample performance than BMA
- Should be used only for model averaging
  - you may drop models with 0 weights
  - you shouldn't choose the model with largest weight unless it's 1
- Yao, Vehtari, Simpson, & Gelman (2018)

Cross-validation and model selection

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

Cross-validation and model selection

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

Cross-validation and model selection

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

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
    - recognised already, e.g., by Stone (1974)

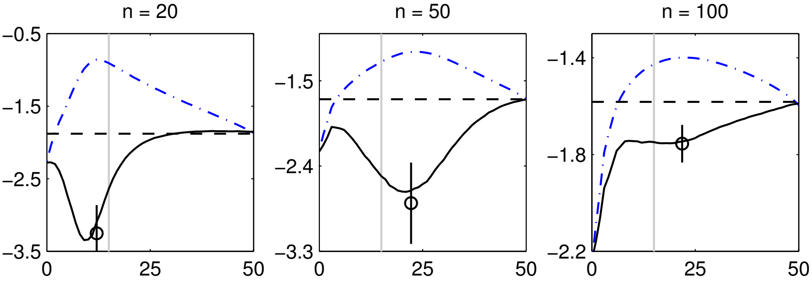
Selection induced bias and overfitting

- 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
  - recognised 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

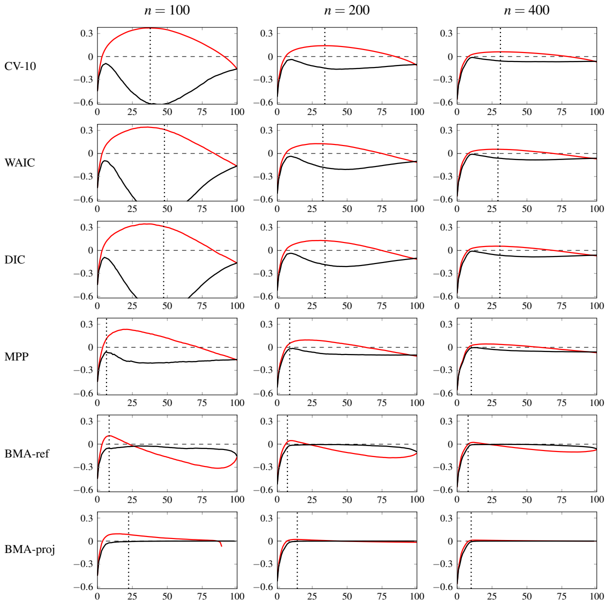
Selection induced bias and overfitting

- 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
  - recognised 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

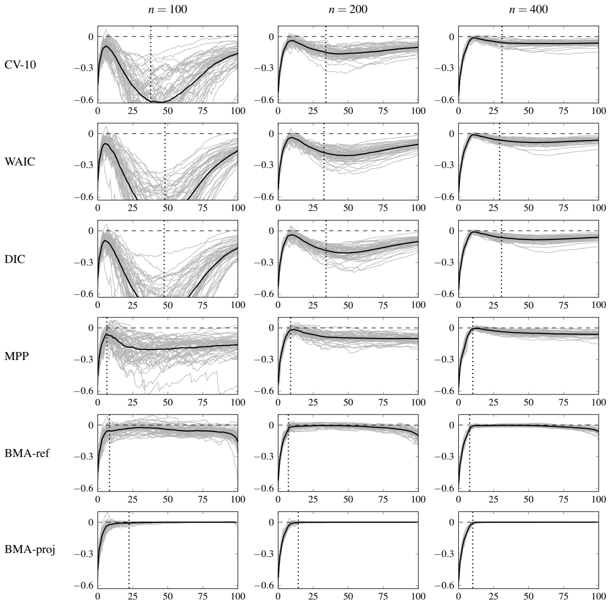


Selection induced bias in variable selection



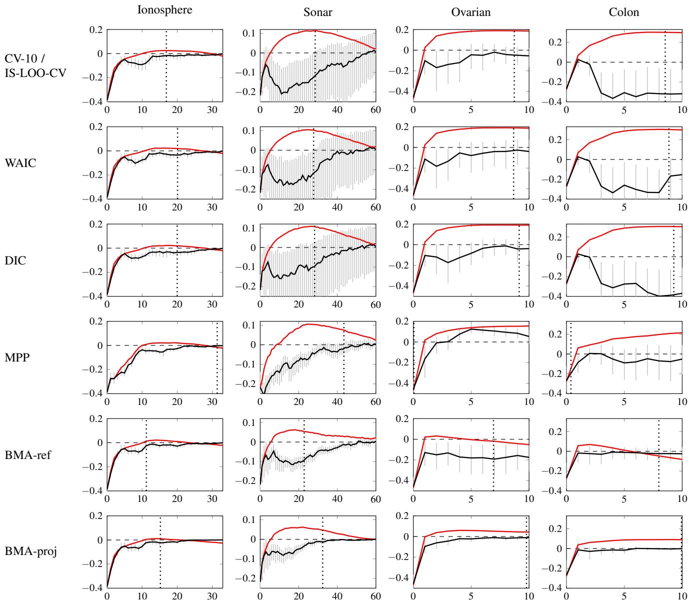
Piironen & Vehtari (2017)

Selection induced bias in variable selection



Piironen & Vehtari (2017)

Selection induced bias in variable selection



Piironen & Vehtari (2017)

Take-home messages (part 1)

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

Take-home messages (part 1)

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

Take-home messages (part 1)

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

Take-home messages (part 1)

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

Take-home messages (part 1)

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

Part 2: Projective Inference in High-dimensional Problems: Prediction and Feature Selection

High dimensional small data

- In the examples $n = 54...102$, $p = 1536...22283$
  - could scale to bigger n and bigger p

High dimensional small data

- In the examples $n = 54...102$, $p = 1536...22283$
  - could scale to bigger n and bigger p
- Priors necessary
  - shrinkage priors, hierarchical shrinkage priors
  - dimension reduction with factor models

High dimensional small data

- In the examples $n = 54...102$, $p = 1536...22283$
  - could scale to bigger n and bigger p
- Priors necessary
  - shrinkage priors, hierarchical shrinkage priors
  - dimension reduction with factor models

- The main content of this part: Two stage approach
  - Construct a best predictive model you can
    $\Rightarrow$ *reference model*
  - Feature selection and post-selection inference
    $\Rightarrow$ *projection*

Rich model vs feature selection?

- If we care only about the predictive performance
  - Include all available prior information
  - Integrate over all uncertainties
  - No need for feature selection

Rich model vs feature selection?

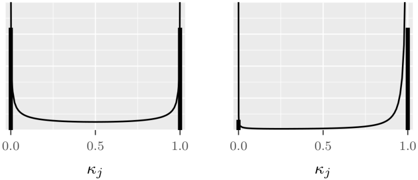
- If we care only about the predictive performance
  - Include all available prior information
  - Integrate over all uncertainties
  - No need for feature selection
- Variable selection can be useful if
  - need to reduce measurement or computation cost in the future
  - improve explainability

Rich model vs feature selection?

- If we care only about the predictive performance
  - Include all available prior information
  - Integrate over all uncertainties
  - No need for feature selection
- Variable selection can be useful if
  - need to reduce measurement or computation cost in the future
  - improve explainability
- Two options for variable selection
  - Find a minimal subset of features that yield a good predictive model
  - Identify all features that have predictive information

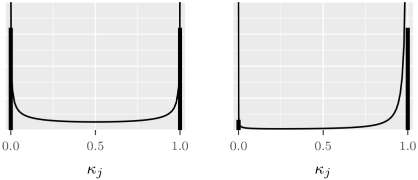
Regularized horseshoe prior

- Horseshoe: can be seen as continuos version of spike-and-slab with *infinite* width slab
  - no shrinkage ($\kappa_j \to 0$) allows complete separation in logistic model with $n \ll p$

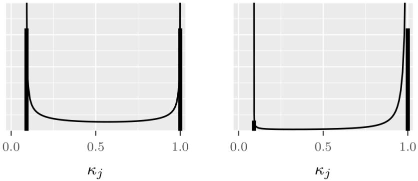


Regularized horseshoe prior

- Horseshoe: can be seen as continuos version of spike-and-slab with *infinite* width slab
  - no shrinkage ($\kappa_j \to 0$) allows complete separation in logistic model with $n \ll p$



- Regularized horseshoe: adds additional *finite* width slab
  - some minimal shrinkage ($\kappa_j > 0$) for relevant features, but maintains division to relevant and non-relevant features



Regularized horseshoe

- Piironen and Vehtari (2017). Sparsity information and regularization in the horseshoe and other shrinkage priors. In Electronic Journal of Statistics, 11(2):5018-5051. Online
    - regularized horseshoe
    - how to set the prior based on the sparsity assumption

Why shrinkage priors alone do not solve the variable selection problem

- A common strategy:
  - Fit model with a shrinkage prior
  - Select variables based on marginal posteriors (of the regression coefficients)

Why shrinkage priors alone do not solve the variable selection problem

- A common strategy:
  - Fit model with a shrinkage prior
  - Select variables based on marginal posteriors (of the regression coefficients)
- Problems
  - Marginal posteriors are difficult with correlated features
  - How to do post-selection inference correctly?

Example

Consider data

$$
\begin{aligned}
f &\sim \mathrm{N}(0,1), \\
y \mid f &\sim \mathrm{N}(f,1) \\
x_j \mid f &\sim \mathrm{N}(\sqrt{\rho} f,\, 1-\rho), \qquad && j = 1,\ldots,25\,, \\
x_j \mid f &\sim \mathrm{N}(0,1), && j = 26,\ldots,50\,.
\end{aligned}
$$

Example

Consider data

$$
\begin{aligned}
f &\sim \mathrm{N}(0, 1), \\
y \mid f &\sim \mathrm{N}(f, 1) \\
x_j \mid f &\sim \mathrm{N}(\sqrt{\rho} f,\ 1 - \rho), && j = 1, \ldots, 25\,, \\
x_j \mid f &\sim \mathrm{N}(0, 1), && j = 26, \ldots, 50\,.
\end{aligned}
$$

- y are noisy observations about latent f

Example

Consider data

$$
\begin{aligned}
f &\sim \mathrm{N}(0,1), \\
y \mid f &\sim \mathrm{N}(f,1) \\
x_j \mid f &\sim \mathrm{N}(\sqrt{\rho} f,\, 1-\rho), \qquad & j = 1,\ldots,25\,, \\
x_j \mid f &\sim \mathrm{N}(0,1), & j = 26,\ldots,50\,.
\end{aligned}
$$

- y are noisy observations about latent f
- First $p_{\mathrm{rel}} = 25$ features are correlated with ρ and predictive about y

Example

Consider data

$$
\begin{aligned}
f &\sim \mathrm{N}(0,1), \\
y \mid f &\sim \mathrm{N}(f,1) \\
x_j \mid f &\sim \mathrm{N}(\sqrt{\rho} f,\, 1-\rho), \qquad && j = 1,\ldots,25\,, \\
x_j \mid f &\sim \mathrm{N}(0,1), && j = 26,\ldots,50\,.
\end{aligned}
$$

- y are noisy observations about latent f
- First $p_{\mathrm{rel}} = 25$ features are correlated with ρ and predictive about y
- Remaining 25 features are irrelevant random noise

Example

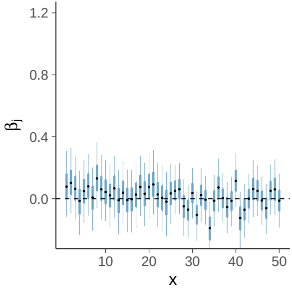
Consider data

$$
\begin{aligned}
f &\sim \mathrm{N}(0,1), \\
y \mid f &\sim \mathrm{N}(f,1) \\
x_j \mid f &\sim \mathrm{N}(\sqrt{\rho} f,\, 1-\rho), \qquad & j = 1,\ldots,25\,, \\
x_j \mid f &\sim \mathrm{N}(0,1), & j = 26,\ldots,50\,.
\end{aligned}
$$

- y are noisy observations about latent f
- First $p_{\mathrm{rel}} = 25$ features are correlated with ρ and predictive about y
- Remaining 25 features are irrelevant random noise

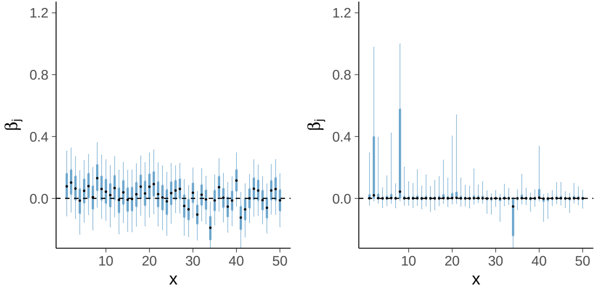
Generate one data set $\{x^{(i)}, y^{(i)}\}_{i=1}^{n}$ with $n = 50$ and $\rho = 0.8$ and assess the feature relevances

Example



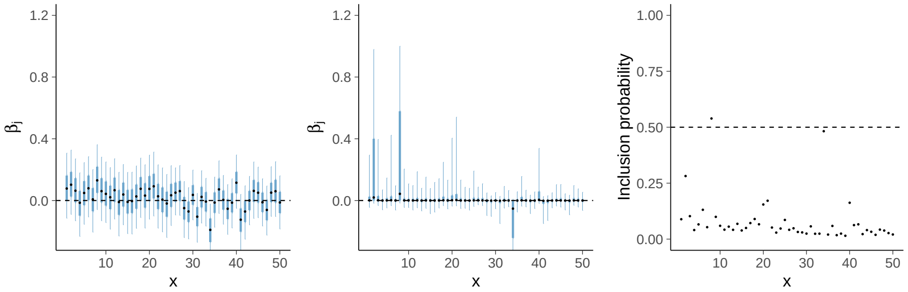
A) Gaussian prior, posterior median with 50% and 90% intervals

Example



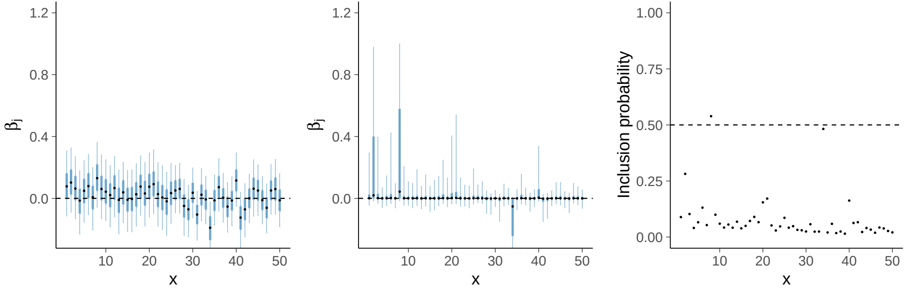
A) Gaussian prior, posterior median with 50% and 90% intervals
B) Horseshoe prior, same things

Example



A) Gaussian prior, posterior median with 50% and 90% intervals
B) Horseshoe prior, same things
C) Spike-and-slab prior, posterior inclusion probabilities

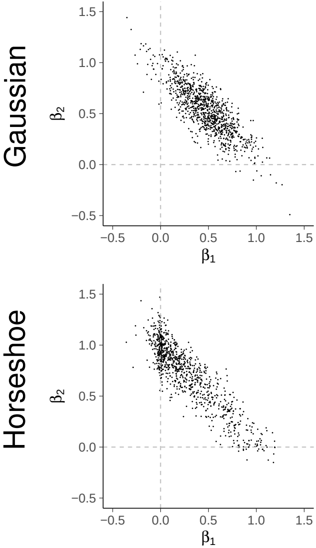
Example



A) Gaussian prior, posterior median with 50% and 90% intervals
B) Horseshoe prior, same things
C) Spike-and-slab prior, posterior inclusion probabilities

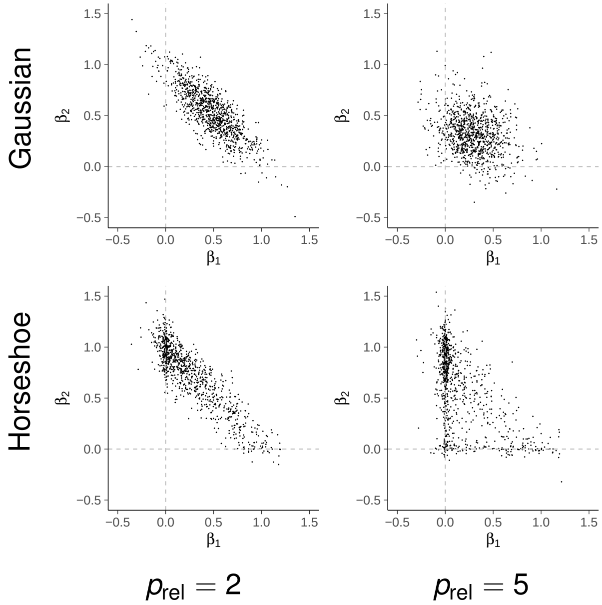
Half of the features relevant, but all marginals substantially overlapping with zero

What happens?

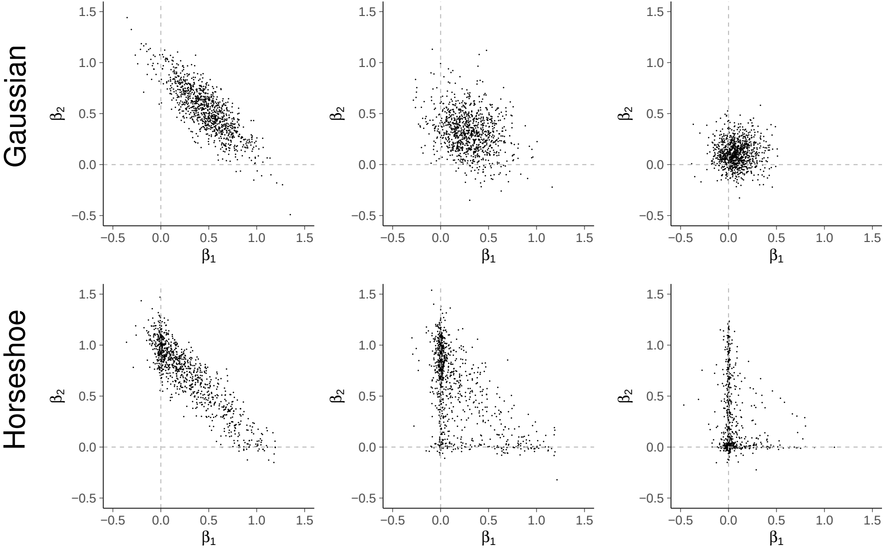


$p_{\text{rel}} = 2$

What happens?



What happens?



$p_{\mathrm{rel}} = 2$ $p_{\mathrm{rel}} = 5$ $p_{\mathrm{rel}} = 25$

Focus on predictive performance

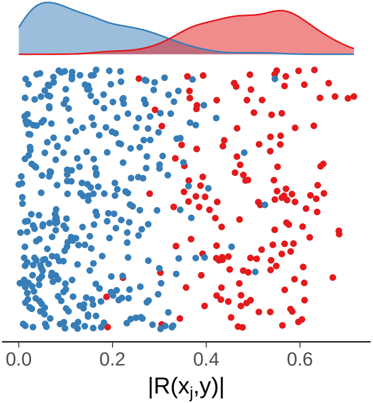
- Two stage approach
  - Construct a best predictive model you can
    $\Rightarrow$ *reference model*
  - Variable selection and post-selection inference
    $\Rightarrow$ *projection*

Focus on predictive performance

- Two stage approach
  - Construct a best predictive model you can
    $\Rightarrow$ *reference model*
  - Variable selection and post-selection inference
    $\Rightarrow$ *projection*
- Instead of looking at the marginals, find the minimal subset of features which have (almost) the same predictive performance as the reference model

Reference model improves variable selection

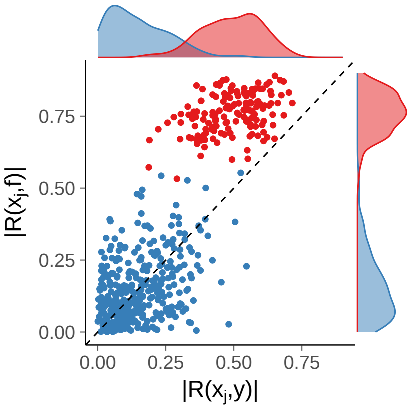
Same data generating mechanism, but
$n = 30$, $p = 500$, $p_{\text{rel}} = 150$, $\rho = 0.5$.



irrelevant x_j, relevant x_j

Sample correlation with y

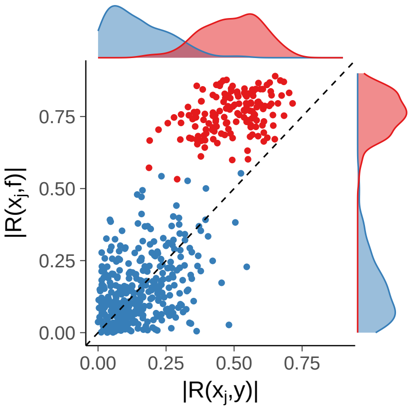
Reference model improves variable selection



irrelevant x_j, relevant x_j

A) Sample correlation with y vs. sample correlation with f

Reference model improves variable selection



irrelevant x_j, relevant x_j

A) Sample correlation with y vs. sample correlation with f

B) Sample correlation with y vs. sample correlation with f_*

$f_* =$ linear regression fit with 3 supervised principal components

(Iterative) Supervised Principal Components

- Dimension reduction for high dimensional small data with highly correlating features
  - dimension reduction helps to speed up later computation without discarding much information
  - supervised means that features correlating with the target are favored in construcing the principal components
- Piironen and Vehtari (2018). Iterative supervised principal components. 21st AISTATS, PMLR 84:106-114. Online.

Predictive projection, idea

- Model simplification technique

Predictive projection, idea

- Model simplification technique
- Replace full posterior $p(\theta \mid D)$ with some constrained $q(\theta)$ so that the predictive distribution changes as little as possible

Predictive projection, idea

- Model simplification technique
- Replace full posterior $p(\theta \mid D)$ with some constrained $q(\theta)$ so that the predictive distribution changes as little as possible
- Example constraints
  - $q(\theta)$ can have only point mass at some θ_0
    $\Rightarrow$ "Optimal point estimates"

Predictive projection, idea

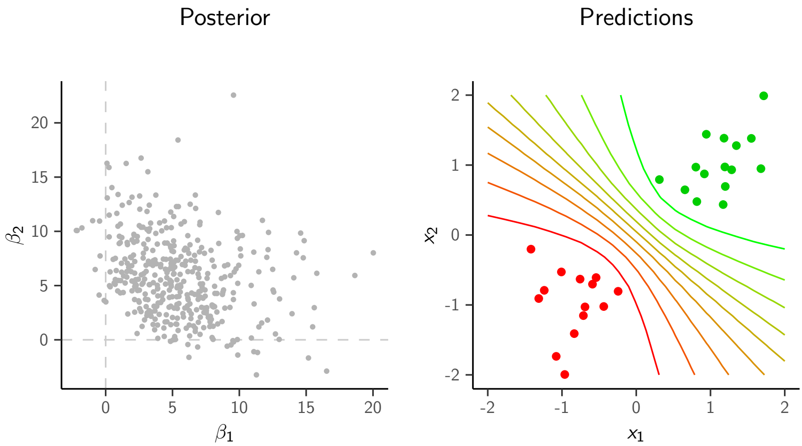
- Model simplification technique
- Replace full posterior $p(\theta \mid D)$ with some constrained $q(\theta)$ so that the predictive distribution changes as little as possible
- Example constraints
  - $q(\theta)$ can have only point mass at some θ_0
    $\Rightarrow$ "Optimal point estimates"
  - Some features must have exactly zero regression coefficient
    $\Rightarrow$ "Which features can be discarded"

Predictive projection, idea

- Model simplification technique
- Replace full posterior $p(\theta \mid D)$ with some constrained $q(\theta)$ so that the predictive distribution changes as little as possible
- Example constraints
  - $q(\theta)$ can have only point mass at some θ_0
    $\Rightarrow$ "Optimal point estimates"
  - Some features must have exactly zero regression coefficient
    $\Rightarrow$ "Which features can be discarded"

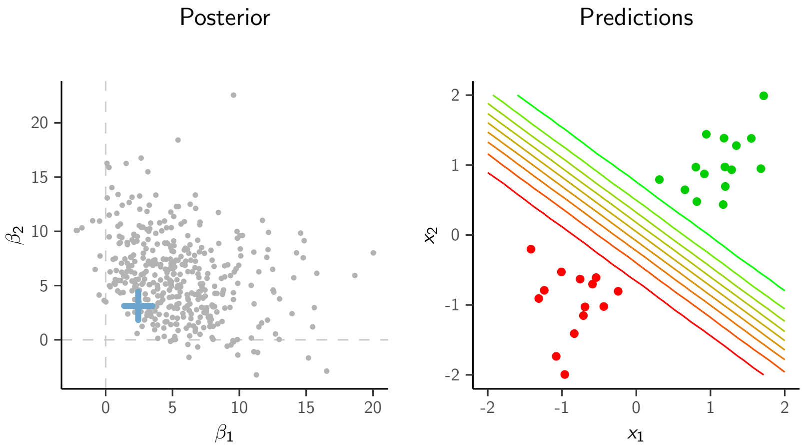
- The decision theoretic idea of conditioning the smaller model inference on the full model can be tracked to Lindley (1968)
  - draw by draw projection introduced by Goutis & Robert (1998), and Dupuis & Robert (2003)
  - see also many related references in a review by Vehtari & Ojanen (2012)

Logistic regression with two features



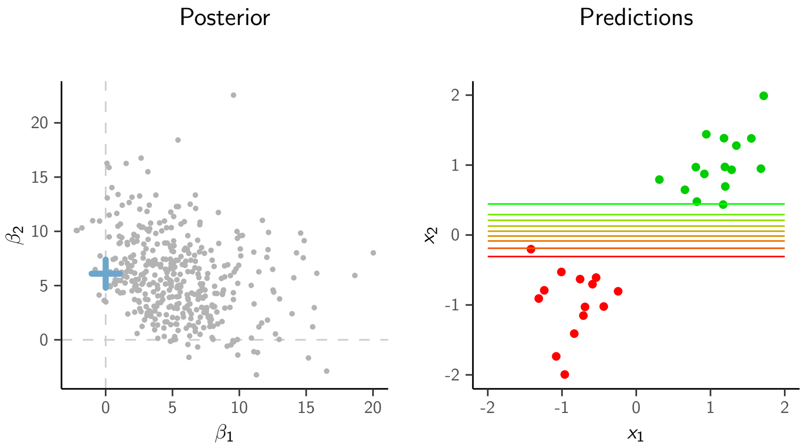
Full posterior for β_1 and β_2 and contours of predicted class probability

Logistic regression with two features



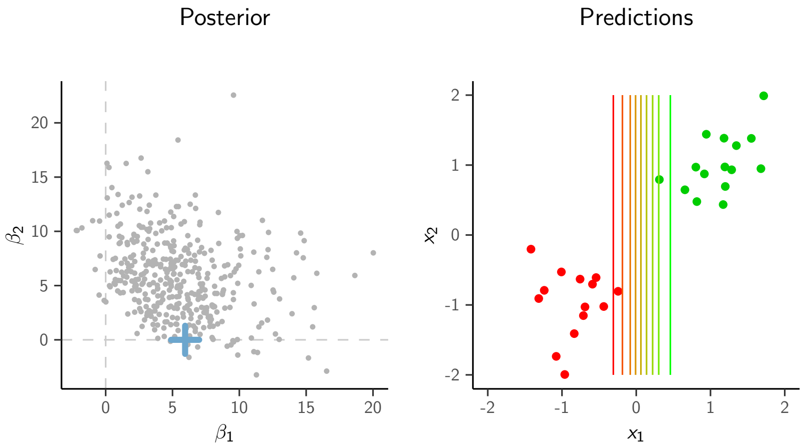
Projected point estimates for β_1 and β_2

Logistic regression with two features



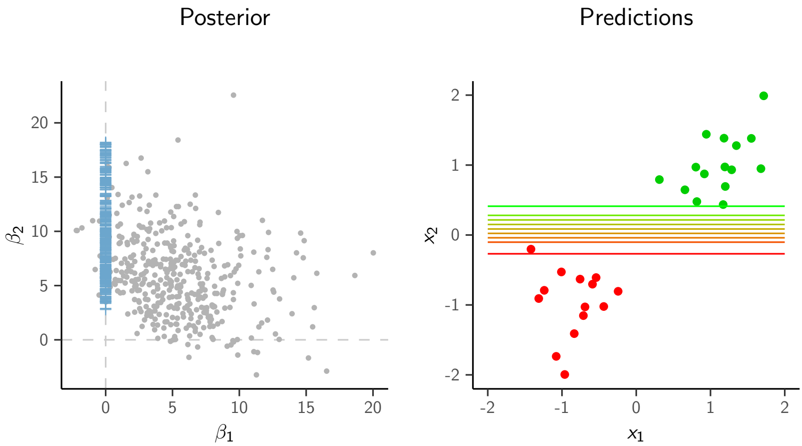
Projected point estimates, constraint $\beta_1 = 0$

Logistic regression with two features



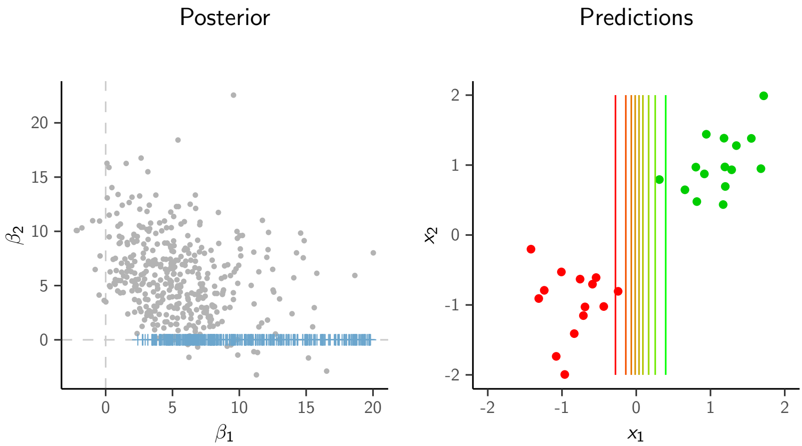
Projected point estimates, constraint $\beta_2 = 0$

Logistic regression with two features



Draw-by-draw projection, constraint $\beta_1 = 0$

Logistic regression with two features



Draw-by-draw projection, constraint $\beta_2 = 0$

Predictive projection

- Replace full posterior $p(\theta \mid D)$ with some constrained $q(\theta)$ so that the predictive distribution changes as little as possible

Predictive projection

- Replace full posterior $p(\theta \mid D)$ with some constrained $q(\theta)$ so that the predictive distribution changes as little as possible
- As the full posterior $p(\theta \mid D)$ is projected to $q(\theta)$
  - the prior is also projected and there is no need to define priors for submodels separately

Predictive projection

- Replace full posterior $p(\theta \mid D)$ with some constrained $q(\theta)$ so that the predictive distribution changes as little as possible
- As the full posterior $p(\theta \mid D)$ is projected to $q(\theta)$
  - the prior is also projected and there is no need to define priors for submodels separately
  - even if we constrain some coefficients to be 0, the predictive inference is conditoned on the information related features contributed to the reference model

Projective selection

- How to select a feature combination?

Projective selection

- How to select a feature combination?
- For a given model size, choose feature combination with minimal projective loss

Projective selection

- How to select a feature combination?
- For a given model size, choose feature combination with minimal projective loss
- Search heuristics, e.g.
  - Monte Carlo search
  - Forward search
  - L_1-penalization (as in Lasso)

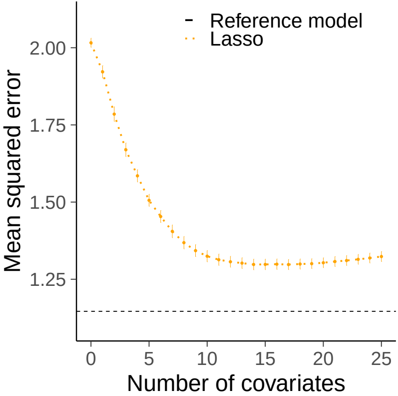
Projective selection

- How to select a feature combination?
- For a given model size, choose feature combination with minimal projective loss
- Search heuristics, e.g.
  - Monte Carlo search
  - Forward search
  - L_1-penalization (as in Lasso)
- Use cross-validation to select the appropriate model size
  - need to cross-validate over the search paths

Projective selection vs. Lasso

Same simulated regression data as before,

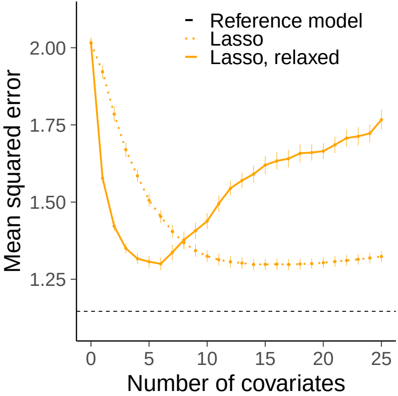
Â $n = 50$, $p = 500$, $p_{\text{rel}} = 150$, $\rho = 0.5$



Projective selection vs. Lasso

Same simulated regression data as before,

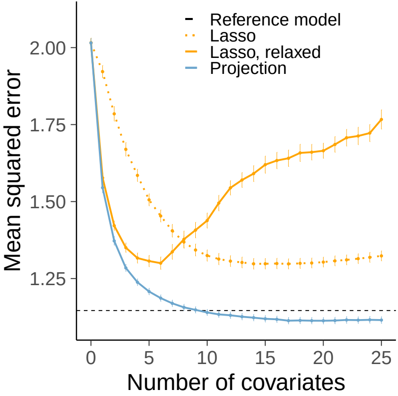
Â $n = 50$, $p = 500$, $p_{\mathrm{rel}} = 150$, $\rho = 0.5$



Projective selection vs. Lasso

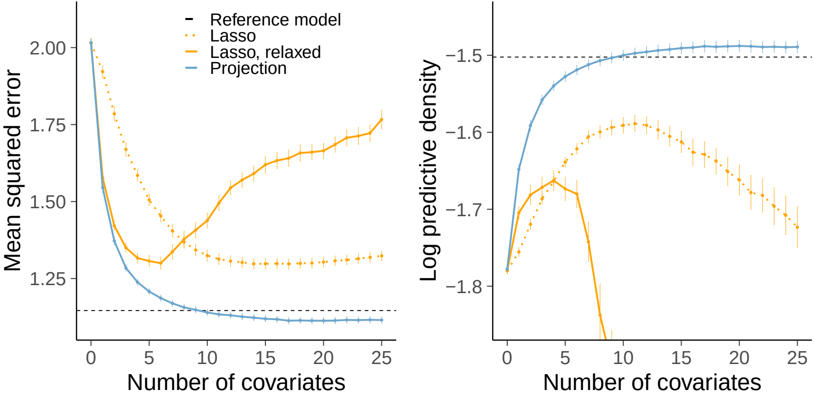
Same simulated regression data as before,

Â $n = 50$, $p = 500$, $p_{\mathrm{rel}} = 150$, $\rho = 0.5$



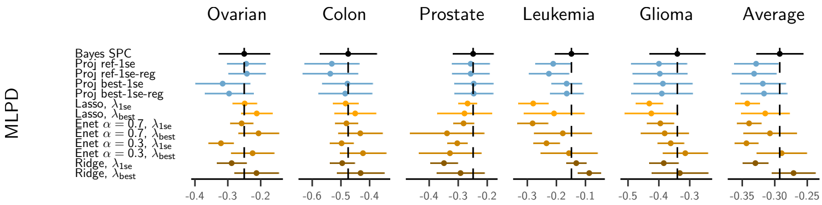
Projective selection vs. Lasso

Same simulated regression data as before,
Â $n = 50$, $p = 500$, $p_{\text{rel}} = 150$, $\rho = 0.5$



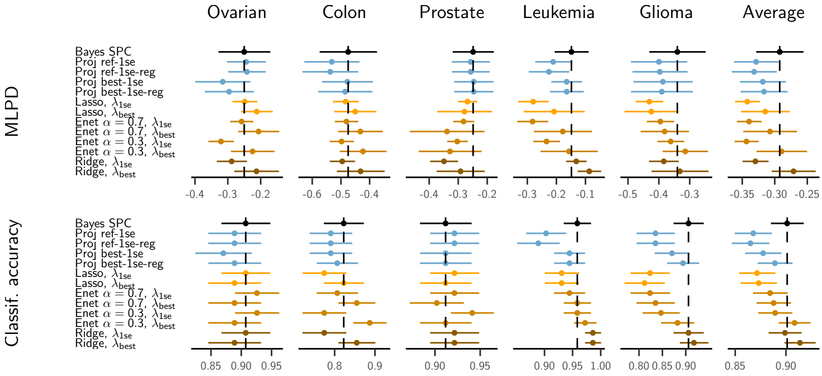
Real data benchmarks

$n = 54...102$, $p = 1536...22283$, Bayes SPC as the reference



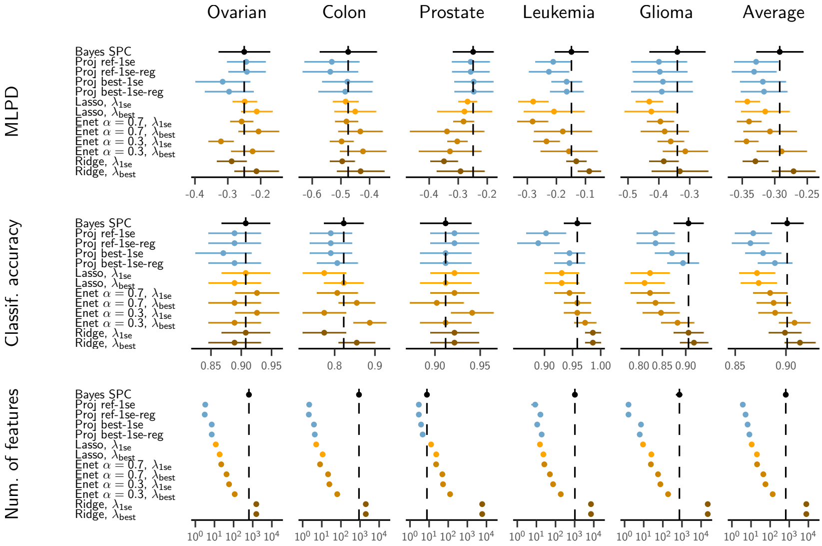
Real data benchmarks

$n = 54...102$, $p = 1536...22283$, Bayes SPC as the reference



Real data benchmarks

$n = 54...102$, $p = 1536...22283$, Bayes SPC as the reference

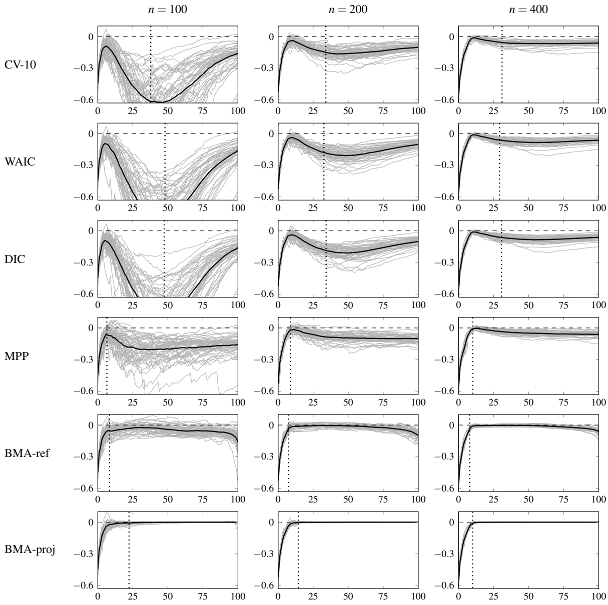


Computation time

Data set	n	p	Computation time			
			Bayes SPC	Projection	Lasso1	Lasso2
Ovarian	54	1536	30.4	3.6	1.3	0.2
Colon	62	2000	31.0	4.0	1.6	0.3
Prostate	102	5966	49.4	7.6	5.0	0.8
Leukemia	72	7129	47.0	6.3	5.6	0.7
Glioma	85	22283	95.8	14.2	15.6	2.6

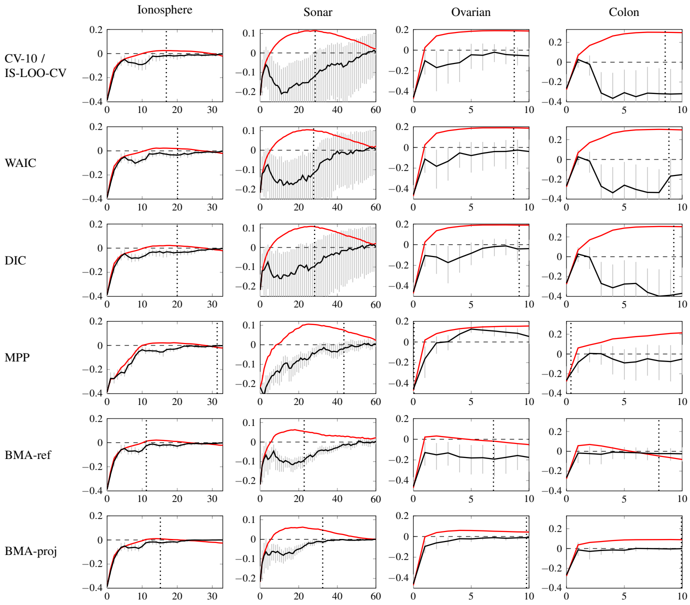
Table: *Computation times:* Average computation time (in seconds) over five repeated runs. In all cases the time contains the cross-validation of the tuning parameters and/or the model size. The first result for Lasso is computed using our software (`projpred`) whereas the second result (and that of ridge) is computed using the R-package `glmnet` which is more highly optimized.

Selection induced bias in variable selection



Piironen & Vehtari (2017)

Selection induced bias in variable selection



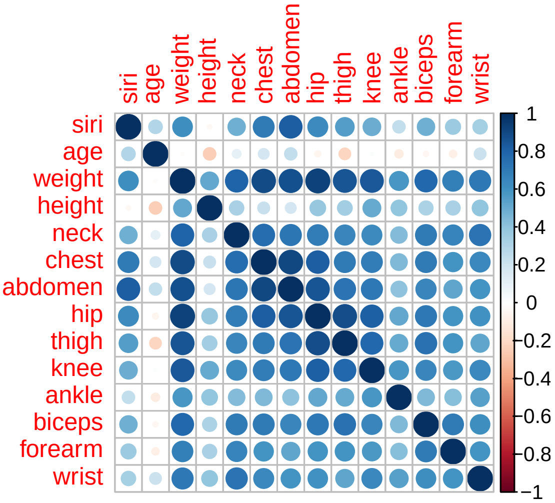
Piironen & Vehtari (2017)

Bodyfat: small p example of projection predictive

Predict bodyfat percentage. The reference value is obtained by immersing person in water. $n = 251$.

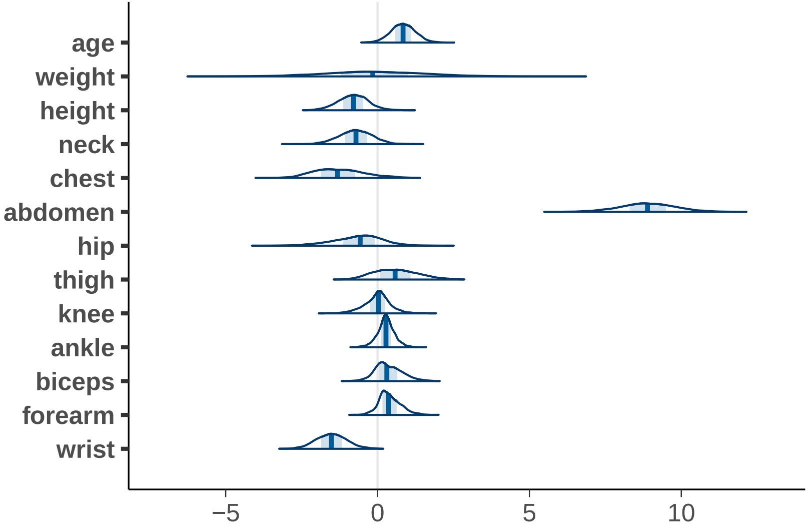
Bodyfat: small *p* example of projection predictive

Predict bodyfat percentage. The reference value is obtained by immersing person in water. $n = 251$.



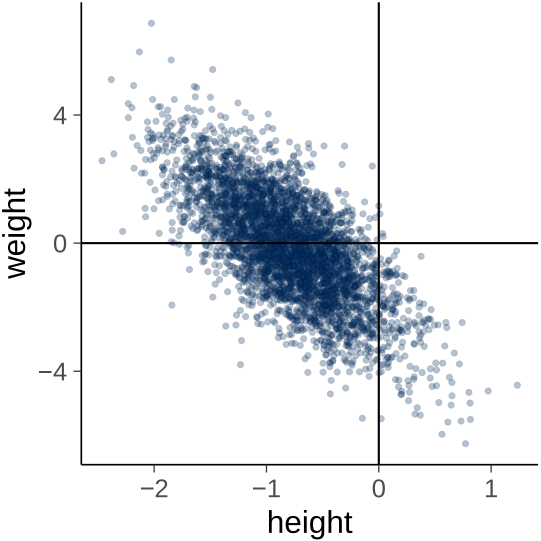
Bodyfat

Marginal posteriors of coefficients



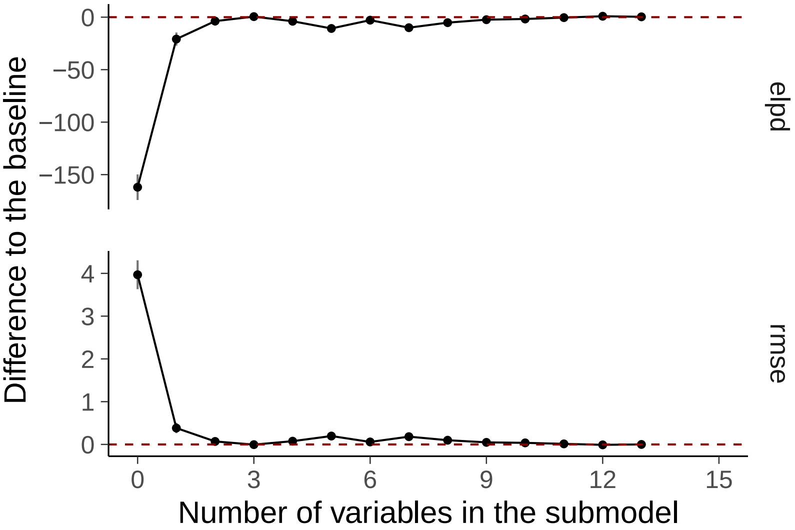
Bodyfat

Bivariate marginal of weight and height



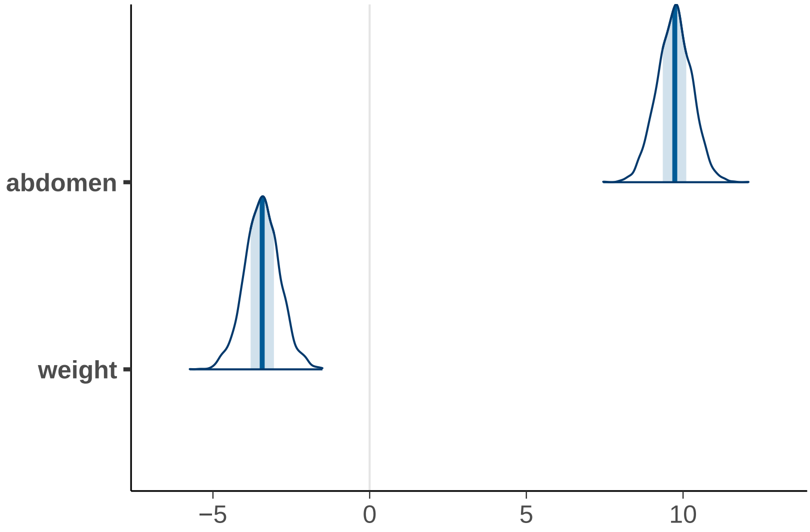
Bodyfat

The predictive performance of the full and submodels



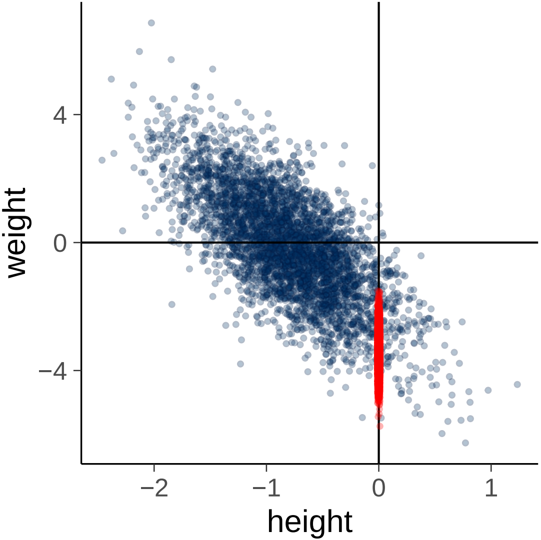
Bodyfat

Marginals of projected posterior



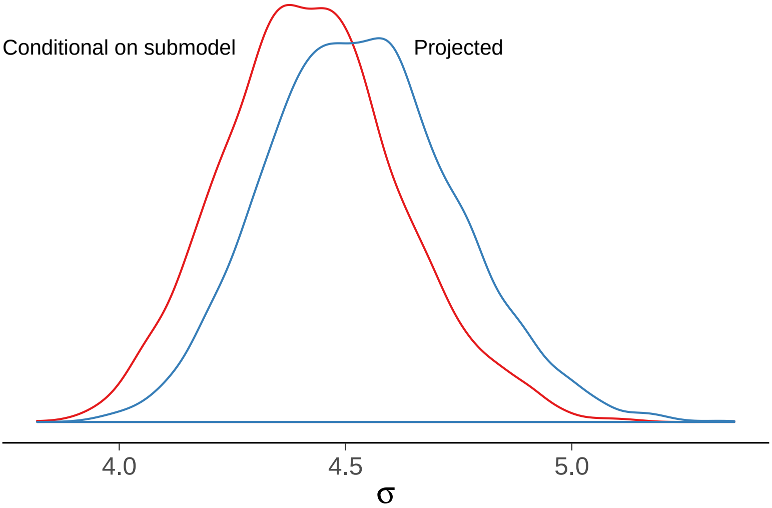
Bodyfat

Projected posterior is not just the conditional of joint



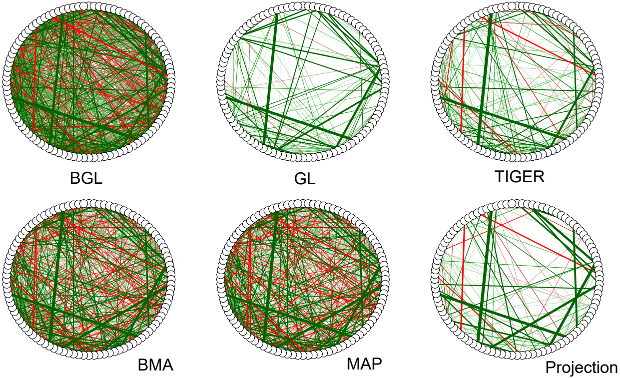
Bodyfat

Projected posterior is different than posterior conditioned only on selected features



Projection of Gaussian graphical models

- Williams, Piironen, Vehtari, Rast (2018). Bayesian estimation of Gaussian graphical models with projection predictive selection. arXiv:1801.05725



CEU genetic network. BGL: Bayesian glasso; GL: glasso; TIGER: tuning insensitive graph estimation and regression; BMA: Bayesian model averaging; MAP: Maximum a posteriori; Projection: projection predictive

More results

- More results projpred vs. Lasso and elastic net:
  Piironen, Paasiniemi, Vehtari (2018). Projective Inference in High-dimensional Problems: Prediction and Feature Selection. arXiv:1810.02406
- More results projpred vs. marginal posterior probabilities:
  Piironen and Vehtari (2017). Comparison of Bayesian predictive methods for model selection. Statistics and Computing, 27(3):711-735. doi:10.1007/s11222-016-9649-y.
- projpred for Gaussian graphical models:
  Williams, Piironen, Vehtari, Rast (2018). Bayesian estimation of Gaussian graphical models with projection predictive selection. arXiv:1801.05725
- More results for Bayes SPC:
  Piironen and Vehtari (2018). Iterative supervised principal components. 21st AISTATS, PMLR 84:106-114. Online.
- Several case studies for small to moderate dimensional ($p = 4 \ldots 100$) small data:
  Vehtari (2018). Model assesment, selection and inference after selection. https://avehtari.github.io/modelselection/

Take-home messages (part 2)

- Sparse priors do not automate variable selection
  - Don’t trust marginal posteriors

Take-home messages (part 2)

- Sparse priors do not automate variable selection
  - Don't trust marginal posteriors
- Reference model + projection can improve feature selection
  - Excellent tradeoff between accuracy and model complexity
  - Useful also for identifying all the relevant features

Take-home messages (part 2)

- Sparse priors do not automate variable selection
  - Don't trust marginal posteriors
- Reference model + projection can improve feature selection
  - Excellent tradeoff between accuracy and model complexity
  - Useful also for identifying all the relevant features
- Well developed for GLMs, but can be used also with other model families

Take-home messages (part 2)

- Sparse priors do not automate variable selection
  - Don't trust marginal posteriors
- Reference model + projection can improve feature selection
  - Excellent tradeoff between accuracy and model complexity
  - Useful also for identifying all the relevant features
- Well developed for GLMs, but can be used also with other model families
- More details and results (+ some theoretical discussion) in the paper
  - Piironen, Paasiniemi, Vehtari (2018). Projective Inference in High-dimensional Problems: Prediction and Feature Selection. arXiv:1810.02406

Take-home messages (part 2)

- Sparse priors do not automate variable selection
  - Don't trust marginal posteriors
- Reference model + projection can improve feature selection
  - Excellent tradeoff between accuracy and model complexity
  - Useful also for identifying all the relevant features
- Well developed for GLMs, but can be used also with other model families
- More details and results (+ some theoretical discussion) in the paper
  - Piironen, Paasiniemi, Vehtari (2018). Projective Inference in High-dimensional Problems: Prediction and Feature Selection. arXiv:1810.02406
- R-package `projpred` in CRAN and github
  https://github.com/stan-dev/projpred
  (easy to use, e.g. with RStan, RStanARM, brms)

References

References and more at avehtari.github.io/masterclass/ and avehtari.github.io/modelselection//

- Model selection tutorial at StanCon 2018 Asilomar
  - more about projection predictive variable selection
- Regularized horseshoe talk at StanCon 2018 Asilomar
- Several case studies
- References with links to open access pdfs