Approximate leave-future-out cross-validation for time series models

Paul-Christian Bürkner, Jonah Gabry, Aki Vehtari

Abstract

One of the most common goals of a time series analysis is to use the observed series to inform predictions for future observations. In the absence of any actual new data to predict, cross-validation can be used to measure a model's predictive accuracy for instance for the purpose of model comparison or selection. As exact cross-validation is often practically infeasible for Bayesian models because it requires too much time, approximate cross-validation methods have been developed; most notably methods for leave-one-out cross-validation (LOO-CV). However, for time-series models, it does not make sense to leave out observations one at a time because then we are allowing information from the future to influence predictions of the past. To apply the idea of cross-validation to time-series models, we thus need some form of leave-future-out cross-validation (LFO-CV). Like exact LOO-CV, exact LFO-CV requires refitting the model many times to different subsets of the data, which is computationally very costly for most nontrivial examples, in particular for Bayesian models. Using Pareto-smoothed importance sampling, we propose a method for approximating exact LFO-CV that drastically reduces the computational burden while also providing informative diagnostics about the quality of the approximation.

1 Introduction

TODO: general introduction to time-series

One of the most common goals of a time series analysis is to use the observed series to inform predictions for future observations. We will refer to this task of predicting a sequence of M future observations as M-step-ahead prediction (M-SAP). Fortunately, once we have fit a Bayesian model and can sample from the posterior predictive distribution, it is straightforward to generate predictions as far into the future as we want. It is also straightforward to evaluate the M-SAP performance of a time series model by comparing the predictions to the observed sequence of M future data points once they become available.

Unfortunately, we are often in the position of having to use a model to inform decisions *before* we can collect the future observations required for assessing the predictive performance. If we have many competing models we may also need to first decide which of the models (or which combination of the models) we should rely on for predictions. In these situations the best we can do is to use methods for approximating the expected predictive performance of our models using only the observations of the time series we already have.

TODO: introduce cross-validation

If there were no time dependence in the data or if the focus is to assess the non-time-dependent part of the model, we could use methods like leave-one-out cross-validation (LOO-CV). For a data set with N observations, we refit the model N times, each time leaving out one of the N observations and assessing how well the model predicts the left-out observation. LOO-CV is very expensive computationally in most realistic settings, but the Pareto smoothed importance sampling (PSIS; Vehtari et al., 2017a,b) algorithm allows for

approximating exact LOO-CV with PSIS-LOO-CV. PSIS-LOO-CV requires only a single fit of the full model and comes with diagnostics for assessing the validity of the approximation.

With a time series we can do something similar to LOO-CV but, except in a few cases, it does not make sense to leave out observations one at a time because then we are allowing information from the future to influence predictions of the past (i.e., times $t+1, t+2, \ldots$ should not be used to predict for time t). To apply the idea of cross-validation to the M-SAP case, instead of leave-*one*-out cross-validation we need some form of leave-*future*-out cross-validation (LFO-CV). As we will demonstrate in this case study, LFO-CV does not refer to one particular prediction task but rather to various possible cross-validation approaches that all involve some form of prediction for new time series data. Like exact LOO-CV, exact LFO-CV requires refitting the model many times to different subsets of the data, which is computationally very costly for most nontrivial examples, in particular for Bayesian analyses where refitting the model means estimating a new posterior distribution rather than a point estimate.

Although PSIS-LOO-CV provides an efficient approximation to exact LOO-CV, until now there has not been an analogous approximation to exact LFO-CV that drastically reduces the computational burden while also providing informative diagnostics about the quality of the approximation. In this paper we present PSIS-LFO-CV, an algorithm that typically only requires refitting the time-series model a small number times and will make LFO-CV tractable for many more realistic applications than previously possible.

TODO: structure of the paper

2 M-step-ahead predictions

Assume we have a time series of observations $y = (y_1, y_2, \ldots, y_N)$ and let L be the *minimum* number of observations from the series that we will require before making predictions for future data. Depending on the application and how informative the data is, it may not be possible to make reasonable predictions for y_i based on $(y_1, \ldots, y_{i-1})$ until i is large enough so that we can learn enough about the time series to predict future observations. Setting $L = 10$, for example, means that we will only assess predictive performance starting with observation y_{11}, so that we always have at least 10 previous observations to condition on.

In order to assess M-SAP performance we would like to compute the predictive densities

$$p(y_{i<M} \,|\, y_{<i}) = p(y_i, \ldots, y_{i+M-1} \,|\, y_1, ..., y_{i-1}) \tag{1}$$

for each $i \in \{L+1, \ldots, N-M+1\}$, where we use $y_{i<M} = (y_i, \ldots, y_{i+M-1})$ and $y_{<i} = (y_1, \ldots, y_{i-1})$ to shorten the notation. The quantities $p(y_{i<M} \,|\, y_{<i})$ can be computed with the help of the posterior distribution $p(\theta \,|\, y_{<i})$ of the parameters θ conditional on only the first $i-1$ observations of the time-series:

$$p(y_{i<M} \,|\, y_{<i}) = \int p(y_{i<M} \,|\, y_{<i}, \theta)\, p(\theta \,|\, y_{<i})\, d\theta. \tag{2}$$

Having obtained S draws $(\theta^{(1)}_{<i}, \ldots, \theta^{(S)}_{<i})$ from the posterior distribution $p(\theta \,|\, y_{<i})$, we can estimate $p(y_{i<M}|y_{<i})$ as

$$p(y_{i<M} \mid y_{<i}) \approx \frac{1}{S} \sum_{s=1}^{S} p(y_{i<M} \mid y_{<i}, \theta_{<i}^{(s)}). \tag{3}$$

In the following, we consider factorizable models in which the response values are conditionally independent given the parameters and the likelihood can be written in the familiar form

$$p(y \mid \theta) = \prod_{n=1}^{N} p(y_j \mid \theta). \tag{4}$$

In this case, $p(y_{i<M} \mid y_{<i}, \theta_{<i})$ reduces to

$$p(y_{i<M} \mid \theta_{<i}) = \prod_{n=i}^{i+M-1} p(y_j \mid \theta_{<i}), \tag{5}$$

due to the assumption of conditional independence between $y_{i<M}$ and $y_{<i}$ given $\theta_{<i}$. Non-factorizable models, which do not make this assumption, are discussed in Bürkner et al. (in review).

2.1 Approximate M-step-ahead predictions

Unfortunately, the math above makes use of the posterior distributions from many different fits of the model to different subsets of the data. That is, to obtain the predictive density $p(y_{i<M} \mid y_{<i})$ requires fitting a model to only the first $i-1$ data points, and we will need to do this for every value of i under consideration (all $i \in \{L+1, \ldots, N-M+1\}$).

To reduce the number of models that need to be fit for the purpose of obtaining each of the densities $p(y_{i<M} \mid y_{<i})$, we propose the following algorithm. Starting with $i = N - M + 1$, we approximate each $p(y_{i<M} \mid y_{<i})$ using Pareto smoothed importance sampling (PSIS; Vehtari et al., 2017a,b):

$$p(y_{i<M} \mid y_{<i}) \approx \frac{\sum_{s=1}^{S} w_i^{(s)} \, p(y_{i<M} \mid \theta^{(s)})}{\sum_{s=1}^{S} w_i^{(s)}}, \tag{6}$$

where $w_i^{(s)}$ are importance weights and $\theta^{(s)}$ are draws from the posterior distribution based on *all* observations. To obtain $w_i^{(s)}$, we first compute the raw importance ratios

$$r_i^{(s)} \propto \frac{1}{\prod_{j \in J_i} p(y_j \mid \theta^{(s)})}, \tag{7}$$

and then stabilize them using PSIS as described in Vehtari et al. (2017b). The index set J_i contains all the indices of observations which are part of the actually fitted model but not of the model whose predictive performance we are trying to approximate. That is, for the starting value $i = N - M + 1$, we have $J_i = \{i, \ldots, N\}$. This approach to computing importance ratios is a generalization of the approach used in PSIS-LOO-CV, where only a single observation is left out at a time and thus $J_i = i$ for all i.

Starting from $i = N - M + 1$, we gradually *decrease* i by 1 (i.e., we move backwards in time) and repeat the

process. At some observation i, the variability of importance ratios $r_i^{(s)}$ will become too large and importance sampling fails. We will refer to this particular value of i as $i_1^\star$. To identify the value of $i_1^\star$, we check for which value of i does the estimated shape parameter k of the generalized Pareto distribution first cross a certain threshold τ (Vehtari et al., 2017b). Only then do we refit the model using only observations before $i_1^\star$ and then restart the process. Until the next refit, we have $J_i = \{i, \ldots, i_1^\star - 1\}$ for $i < i_1^\star$, as the refitted model only contains the observations up to index $N_1^\star = i_1^\star - 1$.

In some cases we may only need to refit once and in other cases we will find a value $i_2^\star$ that requires a second refitting, maybe an $i_3^\star$ that requires a third refitting, and so on. We repeat the refitting as few times as is required (only if $k > \tau$) until we arrive at $i = L + 1$. Recall that L is the minimum number of observations we have deemed acceptable for making predictions (setting $L = 0$ means predictions of all observations should be computed).

TODO: visualisation

The threshold τ is crucial to the accuracy and speed of the proposed algorithm. If τ is too large, we need fewer refits and thus achieve higher speed, but accuracy is likely to suffer. If τ is too small, we get high accuracy but a lot of refits to that speed will drop noticably. When performing exact CV of Bayesian models, almost all of the computational time is spend fitting models, while the time needed to do predictions is negligeble in comparison. That is, a reduction of the number of refits basically implies a proportional reduction in the overall time necessary for CV of Bayesian models.

A mathematical analysis of the Pareto distribution reveals that approximate CV via PSIS is very likely to be highly accuracte as long as $k < 0.5$ (Vehtari et al., 2017b). In practice, PSIS-LOO-CV turned out to be robust for $k < 0.7$ (Vehtari et al., 2017a). That is, for PSIS-LFO-CV introduced in the present paper, we can expect an appropriate threshold to be somewhere between $0.5 \leq \tau \leq 0.7$. It is unlikely to be as high as $\tau = 0.7$, as the error made in the prediction of a certain observation i will propagate to the predictions of observations $i - 1, i - 2, \ldots$ until a refit is performed. That is, problematic observations with high k are likely to have stronger effects in LFO-CV than LOO-CV. We will come back to the issue of setting approriate thresholds in Section 3.

2.2 Block M-step-ahead predictions

Depending on the particular time-series data and model, the Pareto k estimates may exceed τ rather quickly (i.e., after only few observations) and so a lot of refits may be required even when carrying out the PSIS approximation to LFO-CV. In this case, another option is to exclude only the block of B future values that directly follow the observations to be predicted while retaining all of the more distant values $y_{i>B} = (y_{i+B}, \ldots, y_N)$. This will usually result in lower Pareto k estimates and thus less refitting, but crucially alters the underlying prediction task, to which we will refer to as block-M-SAP.

The block-M-SAP version closely resembles the basic M-SAP only if values in the distant future, $y_{>B}$, contain little information about the current observations being predicted, apart from just increasing precision of the estimated parameters. Whether this assumption is justified will depend on the data and model. That is, if the time-series is non-stationary, distant future value will inform overall trends in the data and thus clearly inform predictions of the current observations being left-out. As a result, block-LFO-CV is only recommended for stationary time-series and corrsponding models.

There are more complexities that arise in block-M-SAP that we did not have to care about in standard M-SAP. One is that, by just removing the block, the time-series effectively gets split into two parts, one before and one after the block. This poses no problem for conditionally independent time-series models, where predictions just depend on the parameters and not on the former values of the time-series itself. However, if the model's predictions are *not* conditionally independent as is the case, for instance, in autoregressive models (see Section X), the observations of the left-out block have to be modeled as missing values in order to retain the integreity of the time-series' predictions after the block. A related example from spatial statistics, in which the modeling of missing values is required for valid inference, can be found in Bürkner et al. (in review).

Another complexity concerns the PSIS approximation of block-LFO-CV: Not only does the approximating model contain more observations than the current model whose predictions we are approximating, but it also may *not* contain observations that are present in the actual model. The latter observations are those right after the currently left-out block, which are included in the current model, but not in the approximating model as they were part of the block at the time the approximating model was (re-)fit. A visualitation of this situation is provided in Figure X. More formally, let $\overline{J}_i$ be the index set of observations that are missing in the approximating model at the time of predicting observation i. We find

$$\overline{J}_i = \{\max(i + B, N^\star + 1), \ldots, \min(N^\star + B, N)\} \tag{8}$$

if $\max(i+B, N^\star+1) \leq \min(N^\star+B, N)$ and $\overline{J}_i = \emptyset$ otherwise. As above, $N^\star$ refers to the largest observation included in the model fitting, that is $N^\star = i^\star - 1$ where i^* is the index of the latest refit. The raw importance ratios $r_i^{(s)}$ for each posterior draw s are then computed as

$$r_i^{(s)} \propto \frac{\prod_{j \in \overline{J}_i} p(y_j \mid \theta^{(s)})}{\prod_{j \in J_i} p(y_j \mid \theta^{(s)})} \tag{9}$$

before they are stabilized and further processed using PSIS (see Section 2.1).

3 Simulations

To evaluate the goodness of the approximation of PSIS-LFO-CV, we performed a simulation study by systematically varying the following conditions: The number M of future observations to be predicted took on values of $M = 1$ and $M = 4$. The number of future values to be excluded in the model fitting took on values of $B = \infty$ (i.e., leaving out the whole future), or $B = 10$ (i.e. leaving out only a block of 10 observations). The threshold τ of the Pareto k estimates was varied between $k = 0.5$ to $k = 0.7$ in steps of 0.1. In addition, we evaluated six different data generating models with linear and/or quadatric terms and/or autoregressive terms of order 2 (see Table X for an overview). These models are also illustrated graphically in Figure **??**.

Autoregressive (AR) models are some of the most commonly used time-series models. An AR(p) model – an autoregressive model of order p – can be defined as

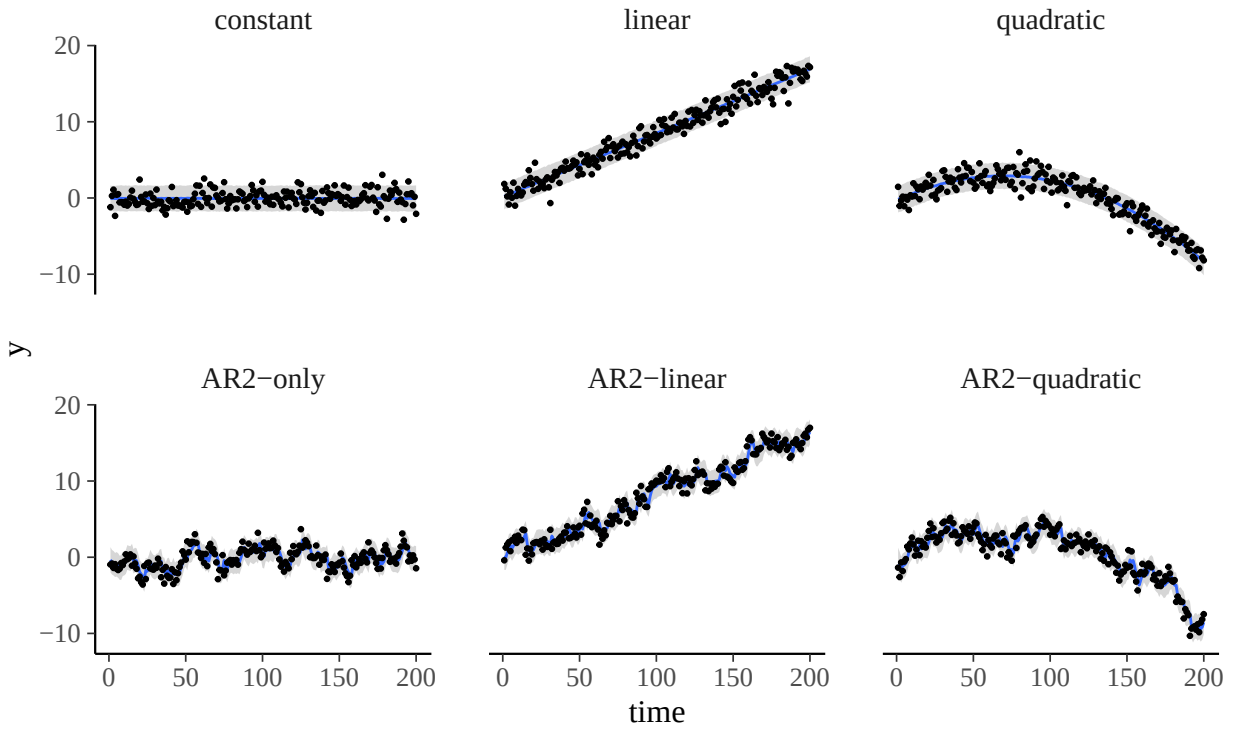

Figure 1: Illustration of the models used in the simulations.

$$y_i = \eta_i + \sum_{k=1}^{p} \varphi_k y_{i-k} + \varepsilon_i, \tag{10}$$

where η_i is the linear predictor for the ith observation, ϕ_k are the autoregressive parameters and ε_i are pairwise independent errors, which are usually assumed to be normally distributed with equal variance σ^2. The model implies a recursive formula that allows for computing the right-hand side of the above equation for observation i based on the values of the equations for previous observations. Thus, by definition, responses of AR-models are not conditionally independent. However they are still factorizable, that is we may write down a separate likelihood contribution per observation (see Bürkner et al., in review, for more discussion on factorizability of statistical models).

3.1 Results

4 Case Studies

4.1 Annual measurements of the level of Lake Huron

To illustrate the application of PSIS-LFO-CV for estimating expected M-SAP performance, we will fit a model for 98 annual measurements of the water level (in feet) of Lake Huron from the years 1875–1972. This

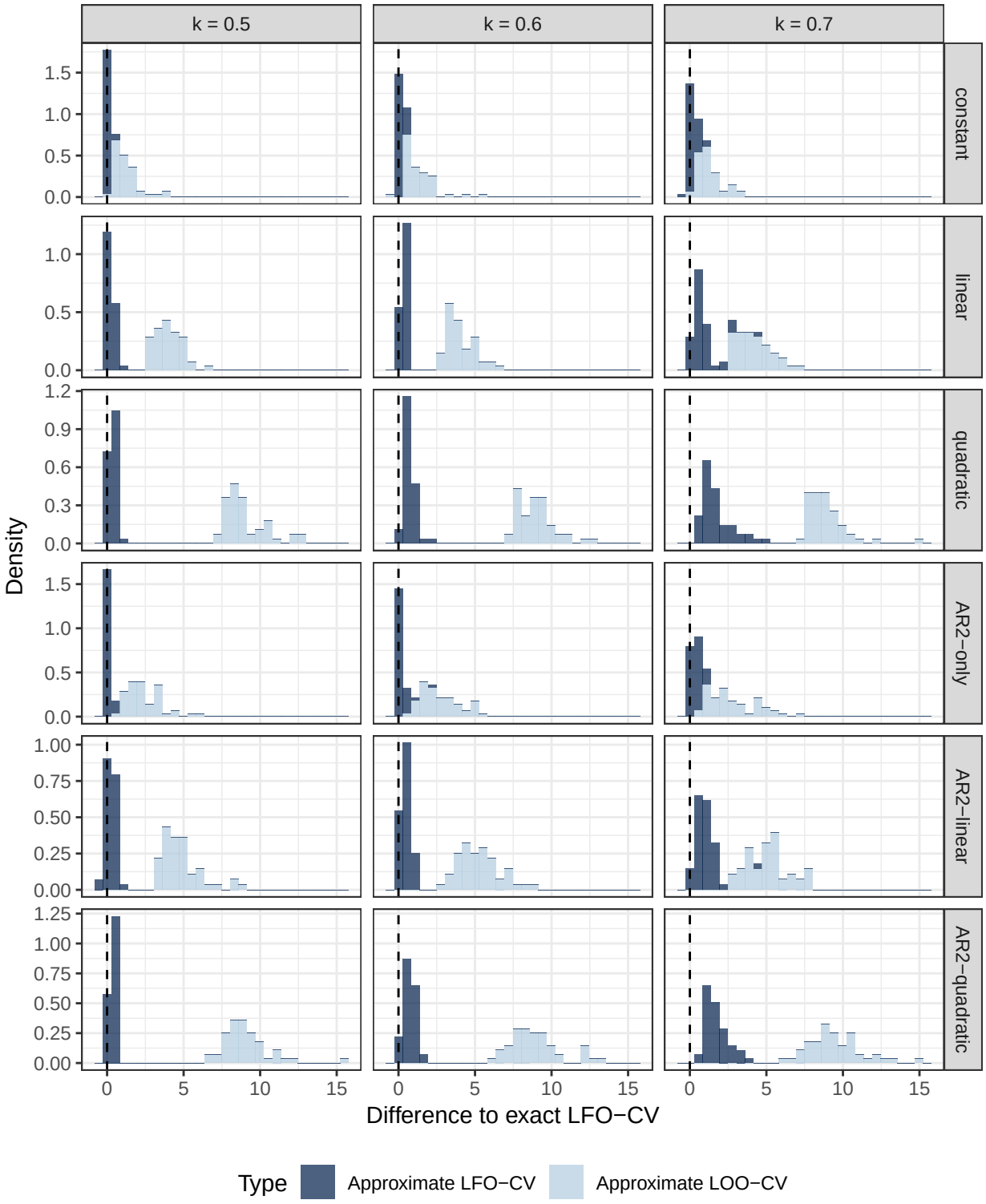

Figure 2: Simulation results of 1-step-ahead predictions.

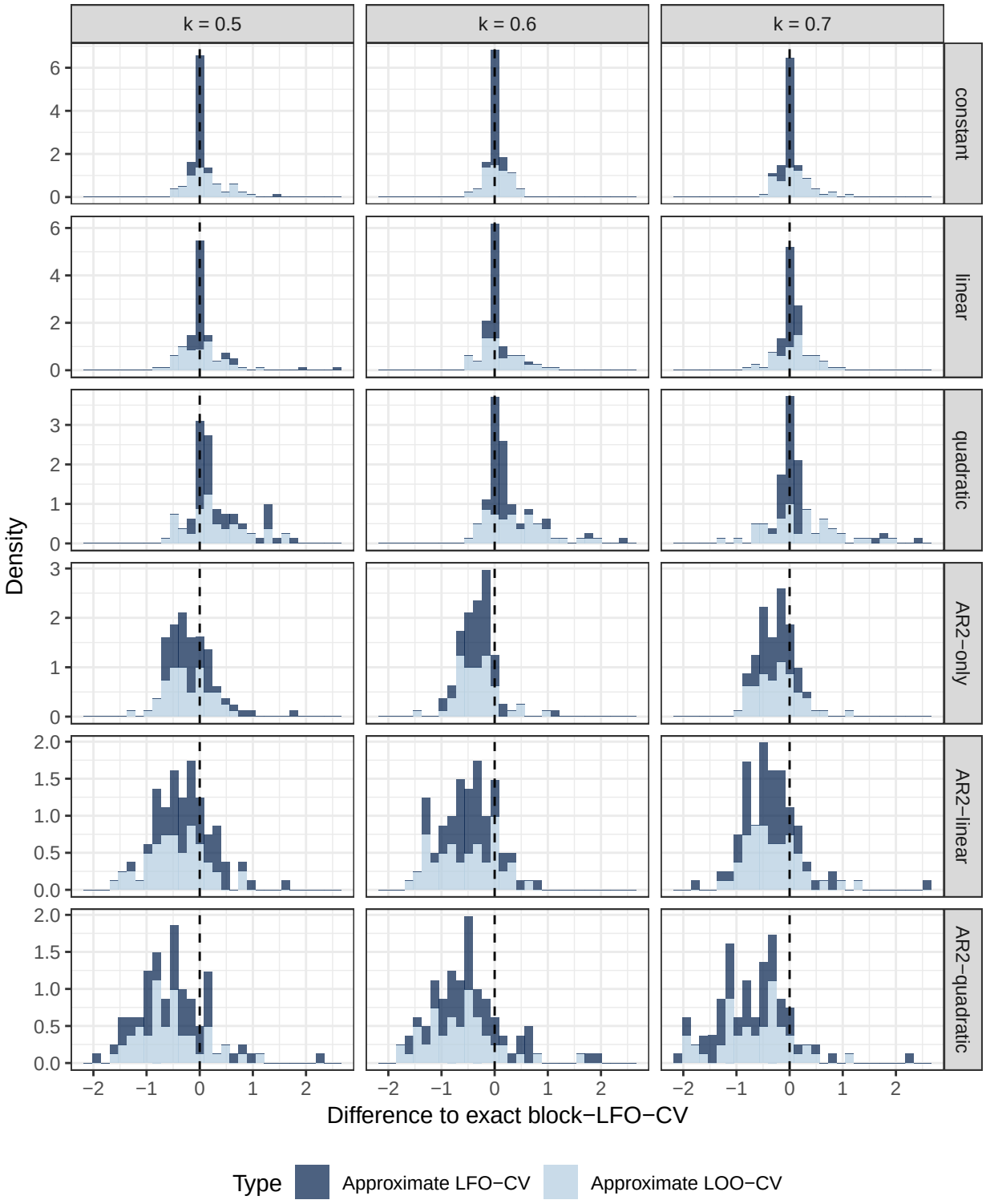

Figure 3: Simulation results of block 1-step-ahead predictions.

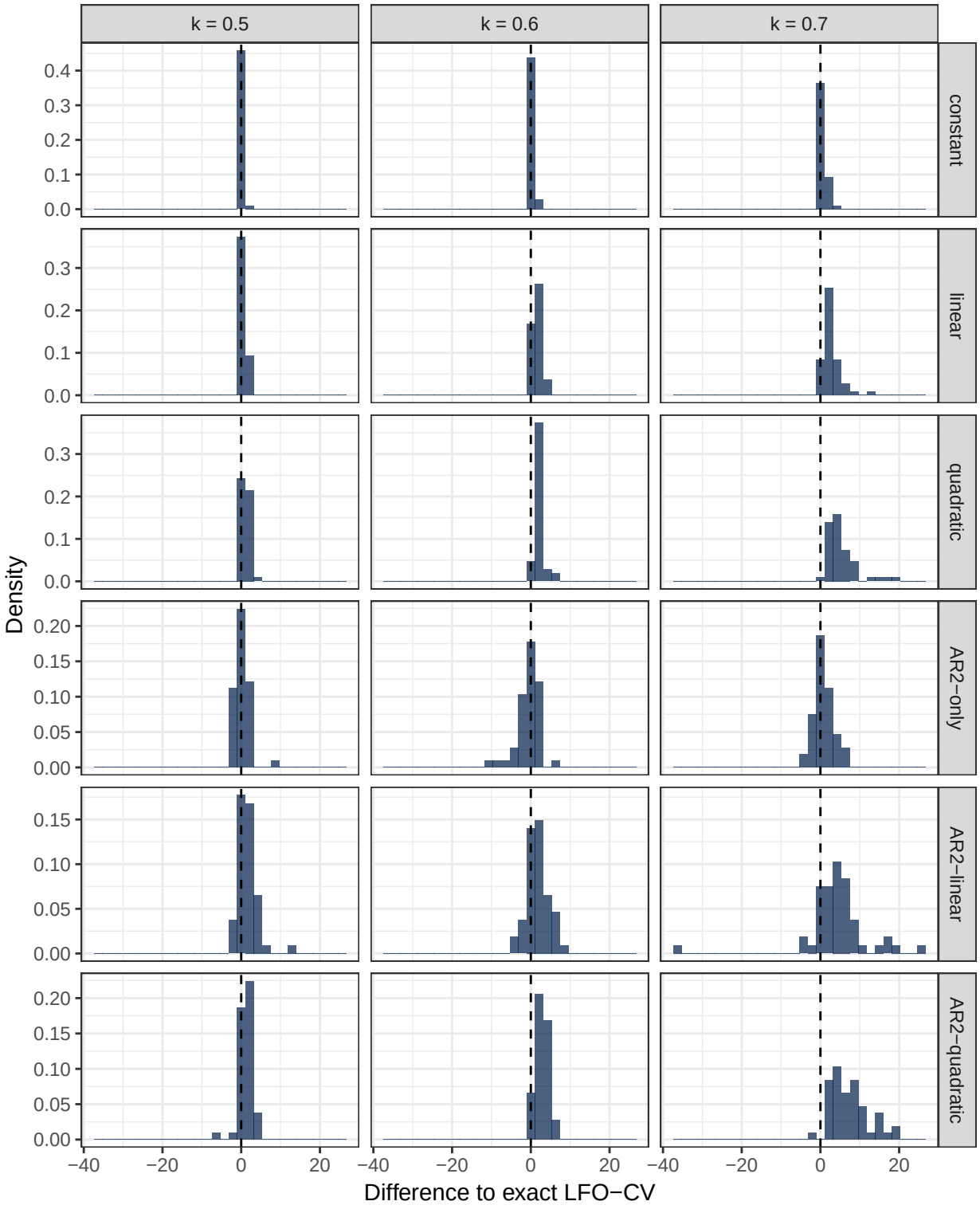

Figure 4: Simulation results of 4-step-ahead predictions.

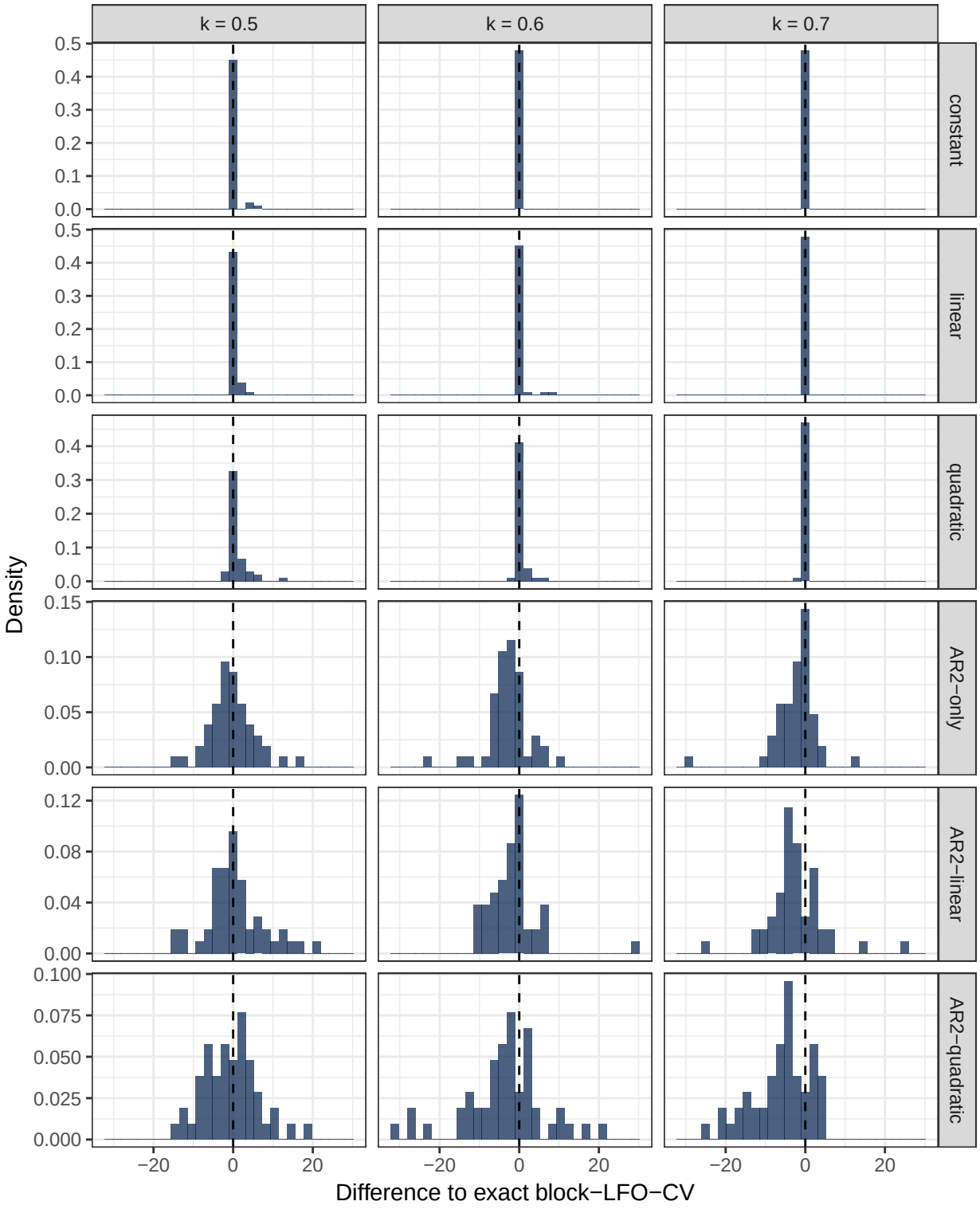

Figure 5: Simulation results of block 4-step-ahead predictions.

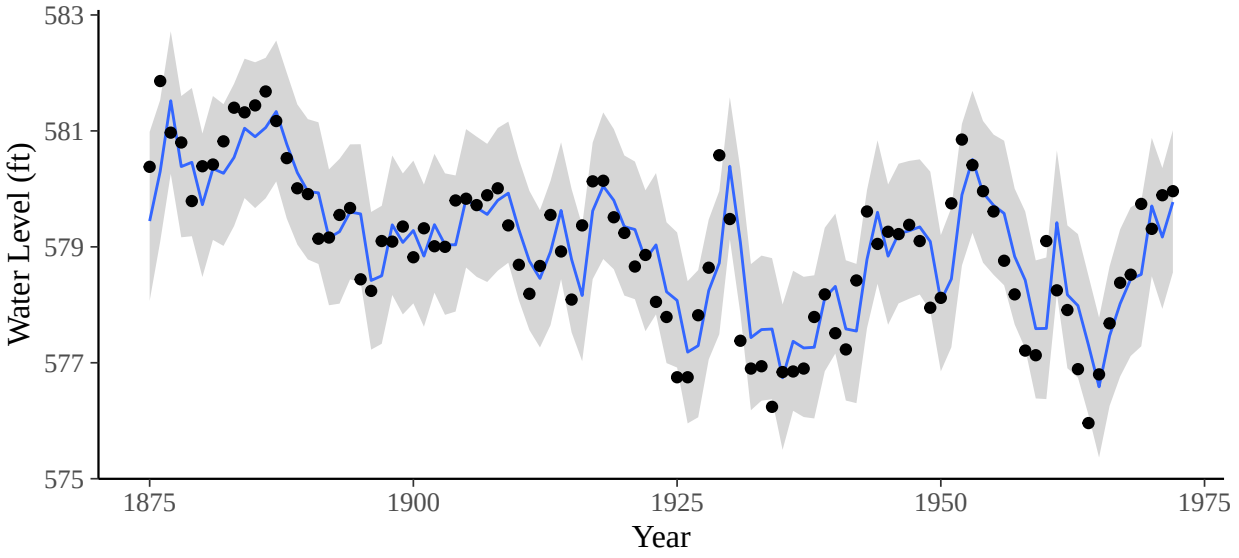

Figure 6: Water Level in Lake Huron (1875-1972). Black points are observed data. The blue line represents mean predictions of an AR(4) model with 90% prediction intervals shown in gray.

data set is found in the *datasets* R package, which is installed automatically with R (R Core Team, 2018). The time-series shows rather strong autocorrelation of the well as some trend towards lower levels for later points in time. We fit an AR(4) model and display the model implied predictions along with the observed values in Figure **??**.

Based on this data and model, we will illustrate the use of PSIS-LFO-CV to provide estimates of 1-SAP and 4-SAP leaving out all future values as well as leaving out only a block of future values. To allow for reasonable predictions of future values, we will require at least $L = 20$ historical observations (20 years) to make predictions. Further, we set a threshold of $\tau = 0.6$ for the Pareto k value at which define that refitting becomes necessary.

We start by computing exact and PSIS-approximated LFO-CV of 1-SAP where we leave out all future values. We compute $\text{elpd}_{\text{exact}}$ = -93.38 and $\text{elpd}_{\text{approx}}$ = -91.73, which are highly similar. Plotting the Pareto k estimates reveals that the model had to be refit 4 times, out of a total of $N - L = 78$ predicted observations (see Figure **??**). On average, this means one refit every 19.5 observations, which implies a drastic speed increase as compared to exact LFO-CV. Performing LFO-CV of 4-SAP leaving out all future values, we compute $\text{elpd}_{\text{exact}}$ = -538.68 and $\text{elpd}_{\text{approx}}$ = -539.58, which are again similar. In general, for increasing M, the approximation tends to become less accurate in absolute elpd units, as the elpd increment of each observation will be based on more and more observations. Since, for constant threshold τ, the importance weights are the same independent of M, Pareto k estimates are also the same in 4-SAP as in 1-SAP.

It is not entirely clear how stationary the time-series is as it may have a slight negative trend across time. However, the AR(4) model we are using assumes stationarity and it is appropriate to also use block-LFO-CV for this example, at least for illustrationary purposes. We choose to leave out a block of $B = 10$ future values as the dependency of an AR(4) model will not reach that far into the future. That is, we will include all observations after this block when re-fitting the model.

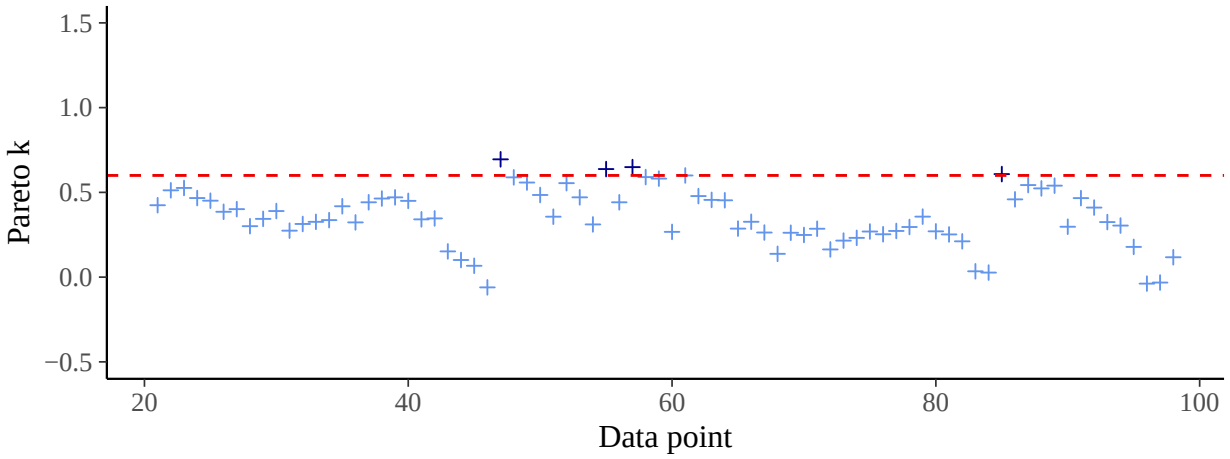

Figure 7: Pareto k estimates for PSIS-LFO-CV of 1 and 4 step-ahead predictions leaving out all future values. The dotted red line indicates the threshold at which the refitting was necessary.

Approximate LFO-CV of block-1-SAP reveals $\text{elpd}_{\text{exact}}$ = -88.55 and $\text{elpd}_{\text{approx}}$ = -87.99, which are highly similar. Plotting the Pareto k estimates reveals that the model had to be refit 2 times, out of a total of $N - L = 78$ predicted observations (see Figure **??**). On average, this means one refit every 39 observations, which again implies a drastic speed increase as compared to exact LFO-CV. What is more, we needed even fewer refits than in non-block LFO-CV, an observation we already made in our simulation in Section 3. Performing LFO-CV of block-4-SAP, we compute $\text{elpd}_{\text{exact}}$ = -484.25 and $\text{elpd}_{\text{approx}}$ = -488.81, which are again similar but not quite a close as in the 1-SAP case. Since AR-models fall in the class of conditionally dependent models, predicting observations right after the left-out block may be quite difficult as shown in Section 3. However, for the present data set, the PSIS apprixmations of block-LFO-CV seem to have worked out just fine.

4.2 Annual date of the cherry blossom in Japan

5 Discussion

TODO: discuss the possibility to compute bayes factors.

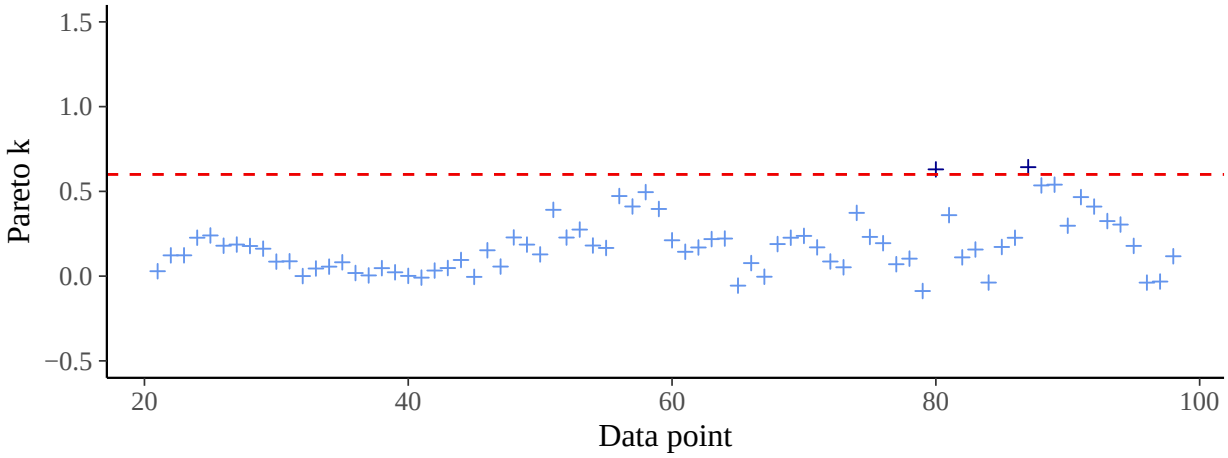

Figure 8: Pareto k estimates for PSIS-LFO-CV of 1 and 4 step-ahead predictions leaving out a block of 10 future values. The dotted red line indicates the threshold at which the refitting was necessary.

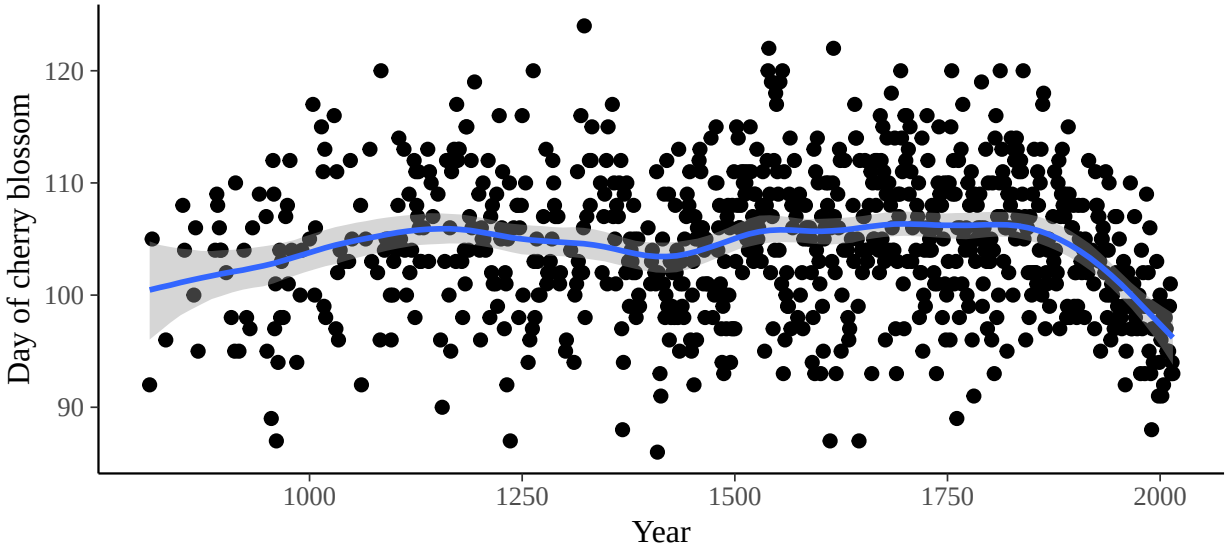

Figure 9: Day of the cherry blossom in Japan (812-2015). Black points are observed data. The blue line represents mean predictions of a thin-plate spline model with 90% regression intervals shown in gray.

References

Paul-Christian Bürkner, Jonah Gabry, and Aki Vehtari. Leave-one-out cross-validation for non-factorizable normal models. in review. URL https://arxiv.org/abs/1810.10559.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018. URL https://www.R-project.org/.

Aki Vehtari, Andrew Gelman, and Jonah Gabry. Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432, 2017a. URL http://link.springer.com/article/10.1007/s11222-016-9696-4.

Aki Vehtari, Andrew Gelman, and Jonah Gabry. Pareto smoothed importance sampling. *arXiv preprint*, 2017b. URL https://arxiv.org/abs/1507.02646.