Recommendations for visual predictive checks in Bayesian workflow

Teemu Säilynoja, Andrew R. Johnson, Osvaldo A. Martin, Aki Vehtari,
Aalto University, Espoo, Finland

February 28, 2025

This article is intended to be viewed as an HTML, and its latest version is hosted online here.

> **!** **Under Review**
>
> This paper is under review on the experimental track of the Journal of Visualization and Interaction. See the reviewing process.

Abstract

Introduction
A key step in the Bayesian workflow for model building is the graphical assessment of model predictions, whether these are drawn from the prior or posterior predictive distribution. The goal of these assessments is to identify whether the model is a reasonable (and ideally accurate) representation of the domain knowledge and/or observed data. There are many commonly used visual predictive checks which can be misleading if their implicit assumptions do not match the reality. Thus, there is a need for more guidance for selecting, interpreting, and diagnosing appropriate visualizations. As a visual predictive check itself can be viewed as a model fit to data, assessing when this model fails to represent the data is important for drawing well-informed conclusions.

Demonstration
We present recommendations for appropriate visual predictive checks for observations that are: continuous, discrete, or a mixture of the two. We also discuss diagnostics to aid in the selection of visual methods. Specifically, in the detection of an incorrect assumption of continuously-distributed data: identifying when data is likely to be discrete or contain discrete components, detecting and estimating possible bounds in data, and a diagnostic of the goodness-of-fit to data for density plots made through kernel density estimates.

Conclusion
We offer recommendations and diagnostic tools to mitigate ad-hoc decision-making in visual predictive checks. These contributions aim to improve the robustness and interpretability of Bayesian model criticism practices.

Materials, Authorship, License, Conflicts

Authorship
- **Teemu Säilynoja**: Conceptualization, Methodology, Software, Investigation, Writing - Review & Editing, Visualization
- **Andrew R. Johnson**: Conceptualization, Writing - Review & Editing

- **Osvaldo A. Martin**: Writing - Review & Editing, Visualization
- **Aki Vehtari**: Conceptualization, Methodology, Writing - Review & Editing, Supervision, Funding acquisition.

Conflicts of interest
The authors declare that there are no competing interests.

1 Introduction

Assessing the sensibility of model predictions and their fit to observations is a key part of most model building workflows. These assessments may reveal that the model predictions poorly represent (or replicate) the observed data, prompting the modeller to either improve their model, or adjust their confidence in the predictions accordingly. In this paper, we focus on examples rising from Bayesian workflows (Gabry et al. 2019; Gelman et al. 2020), such as posterior predictive checking [PPC; Box (1980);Rubin (1984); Gelman, Xiao-Li, and Stern (1996); Gelman et al. (2013)], and visualizations of data, posterior, and posterior predictive distributions. Visualization of data distribution is useful even if we do not do any modeling. In Bayesian inference, the posterior distribution presents the uncertainty in the parameter values after conditioning on data, and the posterior predictive distribution presents the uncertainty in the predictions for new observations. The posterior inference is often performed using Monte Carlo methods (Štrumbelj et al. 2024), and posterior and posterior predictive distributions are then represented with draws from these distributions. In the same way as we visualize data distributions, we can visualize posterior and posterior predictive distributions. In posterior predictive checking, we compare the data distribution to the posterior predictive distribution. If these distributions are not similar, the model is misspecified and we should consider improving the model. These visualizations are also applicable to prior predictive checking, where one might compare prior predictive samples to some reference values rising from domain knowledge (Wesner and Pomeranz 2021), and to cross-validation predictive checking where data are compared to cross-validation predictive distributions to avoid double use of data.

The purpose of this paper is to illustrate common issues in visual predictive checking, to provide better recommendations for different use cases, and to provide diagnostics to automatically warn if useless or misleading visualization is used. Together we have decades of experience in teaching visual predictive checking and helping modellers using popular visual predictive checking tools such as `bayesplot` (Gabry and Mahr (2024a)) and `ArviZ` (Kumar et al. (2019)). We have seen many times where students and modellers use commonly recommended visualizations without realizing that in their context, the specific visualization is the wrong one and either useless or even misleading. Backed up with arguments from recent studies in uncertainty visualization, we provide a summary of methods we recommend. These methods are aimed to offer an informed basis for decision-making, be it for the modeller to improve on the model or for the end user to set expectations on the performance of the model.

The visualizations we discuss should be considered as broadly-applicable tools aimed to give insight into the overall goodness of model fit to the observations, as well as to possible deviations in certain parts of the predictive density. Most modeling workflows benefit from additional visualizations tailored to their specific use cases, as there is no general visual or numeric check that would reveal every aspect of the predictive capabilities of any given model. Some commonly-used methods for visualizing model predictions have risen from comparisons of continuous variables, and as such have varying degrees of usability for assessing predictions with discrete values. Some may not work with all continuous cases, in case of sharp upper or lower bounds in the set of possible values the prediction can take.

In Section 2, we inspect the common use of kernel density estimates (KDE) in summarizing the predictive distribution and the observed data, and show comparisons between commonly used visualization approaches. We highlight common cases when these visualizations may hide useful information and propose an automated goodness-of-fit test to alert the modeller of the presence of conflict between the data and the visualization. In Section 3, we discuss the use of visualizations assuming continuous data when the data and predictions are in reality discrete, but have a high number of unique values. We also discuss an alternate way of visualizing count data, when the number of cases is large, but still small enough for the modeller to be interested in attempting to assess the predictions as individual counts. In Section 4, we focus on workflows involving binary predictions. We showcase tools that expand the visualizations beyond the typical bar graphs and binned calibration plots and instead allow for a more robust assess-

ment of the calibration of the predictive probabilities. In Section 5 and Section 6, we show how the visual predictive checks described for binary predictions can be extended to discrete predictions with a small or medium number of individual cases.

2 Visual predictive checks for continuous data

In this section, we consider visualizations for observations from a *continuous* or *almost everywhere continuous* distribution. We focus on three data visualizations; histograms, and KDE-based continuous density plots, as they are the two most commonly used visualizations for summarizing unidimensional distributions, and quantile dot plots (Kay et al. 2016) as we see that they offer a useful alternative that we recommend in many cases. Table 1 summarizes the main advantages and disadvantages of these three visualizations, and Figure 1 shows a side-by-side comparison of these visualizations. Histograms and density plots are implemented in various commonly used software packages for data visualization and are the two most common choices for initial visual PPCs (Wickham (2016); Gabry and Mahr (2024a); Kumar et al. (2019); Kay (2023)).

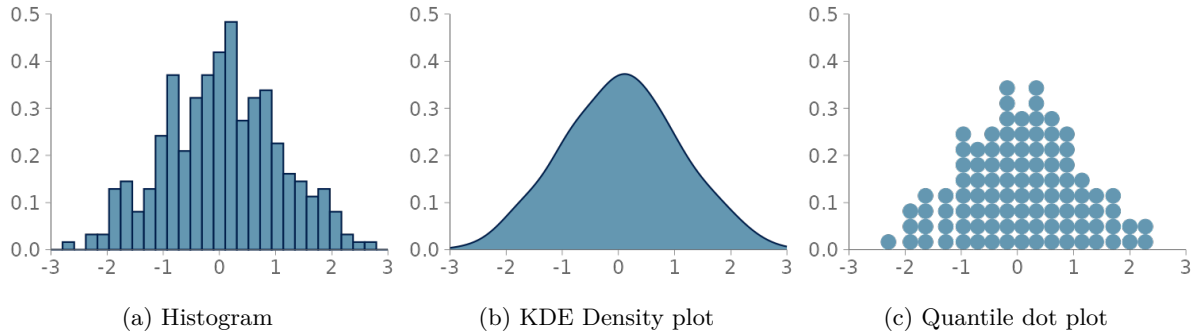| (a) Histogram | (b) KDE Density plot | (c) Quantile dot plot |

Figure 1: Three visualizations of the same sample from a smooth continuous distribution. a) Histogram with 30 equally spaced bins. b) Density plot using a Gaussian kernel and bandwidth 0.37 computed through the R implementation of the method by Sheather and Jones (1991). c) Quantile dot plot with 100 quantiles as implemented in GGDIST by Kay (2023).

Visualization	Advantages	Disadvantages
Histogram	• Familiar to most users. • Can show discontinuities with right binning choice for the data.	• Requires users to select bin width. • Challenges in comparing multiple histograms. • Artefacts with non-discrete valued distributions.
KDE plot	• Familiar to most users. • Clear visual summary when sample adequately smooth. • Easy to stack multiple plots for comparison.	• Tendency to over smooth. • Challenges with bounded data and discontinuities. • Selected kernel bandwidth can greatly impact the visualization.
Quantile dot plot	• Dot position adaptive to local smoothness of sample. • Easily quantifiable tail probability estimation.	• Not familiar to most users. • If area fixed and dots are restricted to circles, the aspect ratio is locked, unless vertical space is added between dots. • Exceptions for discrete distributions need to be implemented.

Table 1: Summary of advantages and disadvantages of the three distribution visualizations discussed in the article.

Histogram

Here, we consider the most commonly used histograms, which are centered over the data, and where all of the bins have an equal width (see an example in Figure 1a). These bin widths are either chosen by the modeller, or through a heuristic, such as the Freedman-Diaconis rule (Freedman and Diaconis 1981). Often, instead of choosing the bin width directly, the modeller inputs a desired amount of bins and an equal partitioning of the data range is computed with an adequate number of breaks.

KDE-based density plot

The most typical KDE-based density plot, often simply called a density plot, is a line or filled area representing a density approximation obtained through convolution of the data with some kernel function (see an example in Figure 1b). Most commonly, a Gaussian kernel is used for the approximation, and a kernel bandwidth is selected through one of the widely used algorithms (Sheather and Jones 1991; Scott 1992; Silverman 2018). As the default implementation of the KDE plot in most visualization software packages is usually enough to produce aesthetically pleasing smooth summaries of the data, KDE plots are a very popular method of visually summarizing data.

Quantile dot plot

The quantile dot plot (Kay et al. 2016), is a dot plot, where a set number of the observed quantiles—we use one hundred—are visualized instead of the observations themselves (see an example in Figure 1c). This results in a visual density estimator with a low bias and high variance similar to that of a histogram (Wilkinson 1999). Kay et al. (2016) show that a quantile dot plot with one hundred quantiles performs very similarly to a KDE-based density plot for tasks of estimating probabilistic predictions from visualizations. Compared to KDEs, quantile dot plots have the added benefit of allowing for fast visual probability estimation in the tails of the distribution. In our experience, and as shown in Section 2.6 and Section 2.4 in case of discontinuities and outliers, quantile dot plots also offer a better visualization than kernel density plots and histograms.

Advantages and disadvantages of the three visualizations

In Table 1 we have collected what we see as the main advantages and disadvantages of histograms, KDE density plots, and quantile dot plots for visualizing observations and assessing the quality of model predictions. Both the histogram and the KDE density plot have commonly recognized drawbacks, summarized by a trade-off between over and under smoothing.

The choice of bin width and breakpoints causes histograms to suffer from binning artefacts (Dimitriadis, Gneiting, and Jordan 2021). Additionally, comparing the histogram of observations to multiple histograms visualizing predictive samples may be difficult.

Density plots from KDEs, as shown in Section 2.4, Section 2.5, and Section 2.6, have a tendency to either hide details such as discontinuities in the observation distribution if the chosen bandwidth is too large, or over-fit to the sample when the bandwidth is too small. When over-fitting, comparing multiple density plots may be difficult, as there is more variation on the estimate of the underlying distribution. These drawbacks are typically especially common in the standard implementations of KDE density plots. More specialized software often implement measures, such as automated boundary detection and less conservative bandwidth selection algorithms, to mitigate these issues (Kumar et al. 2019; Kay 2023).

Based on our experience, we prefer quantile dot plots in many cases, as they offer a low overhead solution to plotting the data without defining a suitable binning and are flexible enough to represent distributions with long tails.

2.1 Assessing the goodness-of-fit of a density visualization

As KDE plots, histograms, and quantile dot plots all produce a visualization of a density, we can assess the representativeness of the visualization by assessing the goodness-of-fit of the implied density estimate to the data being visualized. To assess the goodness-of-fit of a density estimate $\widehat{f}$, we test for the uniformity of the probability integral transformed (PIT) data $\{x_1, ..., x_N\}$ when the density approximator is used for the transform. The PIT value, u_i, of x_i w.r.t. a density estimator $\widehat{f}$ is defined as the cumulative distribution value,

$$u_i = \int_{-\infty}^{x_i} \widehat{f}(x) \, dx. \tag{1}$$

The underlying principle of this goodness-of-fit test is that, when computed with regard to the true underlying density f s.t. $x_i \sim f$, the PIT values would satisfy $u_n \sim \mathbb{U}(0,1)$, where $\mathbb{U}(0,1)$ is the standard uniform distribution.

Our uniformity test of choice, due to its graphical representation that further enhances the explainability of the results, is the graphical uniformity test proposed by Säilynoja, Bürkner, and Vehtari (2022). This test provides simultaneous $1 - \alpha$ level confidence bands for the uniformity of the empirical cumulative distribution function (ECDF) of the PIT values. In the graphical test, the PIT-ECDF and the simultaneous confidence intervals are evaluated at a set of equidistant points along the unit interval. An example

of these simultaneous confidence intervals for the ECDF of the PIT values, F_{PIT} is provided in Figure 3b. Alternatively, Figure 3c shows the ECDF difference version of the same information, that is, the vertical axis of the plot is scaled to show the difference of the observed ECDF to the theoretical expectation, $F_{\text{PIT}}(x) - x$. Especially for large samples, this transformation allows for an easier assessment of the values near the ends of the unit interval.

As goodness-of-fit testing does not greatly increase the computational cost of constructing the visualizations, we propose implementing automated testing and recommendations for users to consider alternative visualization techniques when the fit of the chosen visualization isn't satisfactory.

Next, we introduce how PIT can be implemented for the three visualizations.

PIT for KDE density plots

In theory, a KDE defines a proper probability distribution, and the PIT could be directly computed as in Equation 1. However, in practice the KDE density visualizations are often showing a truncated version of the KDE, limited to an interval centered around the domain of the observed data. For PIT computation, we limit the integration to the displayed range and normalize the results to make the truncated KDE integrate to one.

Most software implementations evaluate the KDE on a dense grid spanning the domain of the observed data, thus allowing numerical PIT computation by extracting the density values on these evaluation points and carrying out the aforementioned normalization to obtain PIT values.

PIT for histograms

For a histogram with equal bins of width h, we define the PIT as

$$\text{PIT}(x) = h \sum_{j=1}^{J} f_j + (x - l_{J+1}) f_{J+1}, \tag{2}$$

where $J = \max_j \{r_j \leq x\}$, and l_j and r_j are the left and right ends of the jth bin.

Again, the relative densities and boundaries of these bins are usually readily available in the software used for constructing these visualizations, and thus implementing the transform for histograms is a relatively simple task.

PIT for quantile dot plots

As the quantile dot plots are discrete, the PIT values computed according to Equation 1 are not guaranteed to be uniformly distributed. As demonstrated by Früiiwirth-Schnatter (1996), for a discrete random variable X, one can employ a randomized interpolation method to obtain uniformly distributed values,

$$u(x) = \alpha P(X \leq x) + (1 - \alpha) P(X \leq (x - 1)), \tag{3}$$

where $\alpha \sim \mathbb{U}(0, 1)$.

For quantile dot plots, where n_q is the number of quantiles, c_k the horizontal position of the center of the kth dot and r the radius of the dots, we define a randomized PIT as

$$\text{PIT}(x) \sim \mathbb{U}\left(\frac{l(x)}{n_q}, \frac{u(x)}{n_q} \right), \tag{4}$$

§

where n_q is the amount of quantiles, that is, dots, in the plot, and $l(x)$ and $u(x)$ are the quantile indices satisfying

$$u(x) = \min\{k \in \{1, \ldots, n_q\} \mid x < c_k - r\}, \tag{5}$$

and

5

$$l(x) = \begin{cases} 0, & \text{if } u(x) = 1, \\ \min\{k \in \{1, \dots, n_q\} \mid |c_k - x| \le r\}, & \text{if } \exists\, k \text{ s.t. } |c_k - x| \le r, \\ \max\{k \in \{1, \dots, n_q\} \mid x \ge c_k - r\}, & \text{otherwise.} \end{cases} \quad (6)$$

That is, we consider the left and right edges of the dots, $c_k - r$, and $c_k + r$ respectively. Now the PIT value of x is limited from above by the smallest quantile dot fully to the right of x. The lower limit of the PIT value is determined in three cases. First, for those x that are to the left of all of the quantile dots, the lower limit of the PIT value is zero. Second, if x lies between the left and right edges of one or more quantile dots, the lower limit is determined by the smallest of these quantiles. Finally, when x is not between the edges of any single quantile dot, the largest dot fully to the left of x determines the lower limit of the PIT value.

Again, as the centre points and radius of the dots are required for constructing the visualizations in the first place, implementing the PIT computations is relatively straightforward.

2.2 Continuous valued observations

When the observation density is smooth and unbounded, and the practitioner uses visualizations aimed at continuous distributions, misrepresenting the data is less common. The main benefit of goodness-of-fit testing for visualizations arises, when the assumptions in the visualization do not meet the properties of the underlying observation distribution. Sometimes data, that may seem continuous at first glance, proves problematic for KDEs to visualize.

Below, we will first look at the ideal case of a continuously valued observation with a smooth and unbounded underlying distribution, and then step through three examples of continuous valued observations where the observation distribution is not smooth and visualizing the observation requires extra attention. The true underlying densities are visualized in Figure 2. In the three later cases, using the default KDE plot or histogram can hide important details of the observation, and looking at the issues detected through goodness-of-fit testing could affect future modeling choices.

(a) Density with steps   (b) Strictly bounded density   (c) Density with a point mass
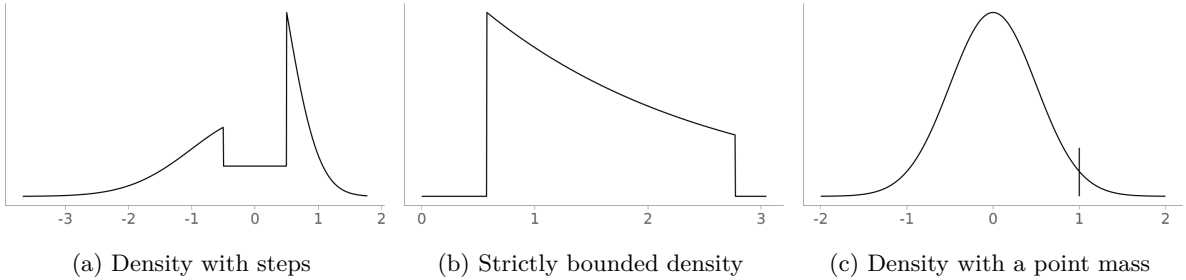
Figure 2: True densities for the examples used in Section 2.4, Section 2.5, and Section 2.6. Each of these examples poses a challenge for KDE density plots and histograms, and avoiding misrepresenting the data requires special attention.

2.3 Smooth and unbounded density

When the underlying distribution is smooth and unbounded, large issues in goodness-of-fit of the discussed three visualizations are rare. Figure 3a includes the KDE plot for an observation of 1000 standard normally distributed values. Figures Figure 3b and Figure 3c show the corresponding goodness-of-fit test with the PIT ECDF values well within the 95% simultaneous confidence intervals for uniformity. Figure 4 shows the histogram and the corresponding goodness-of-fit assessment of the same data. This time, we show only the ECDF difference version of the graphical test. Again, no issues are detected. Figure 5 in turn shows the quantile dot plot paired with the corresponding goodness-of-fit evaluation. Again, no issues are detected. Although one hundred evaluation points results in a quite smooth visualization, the discreteness of the PIT ECDF of the quantile dot plot is visible.

2.4 Density with steps

When the observation density has steps, that is, points with unequal one-sided limits, the continuous representation offered by KDEs—that assume smooth density—may struggle to accurately represent the discontinuity. In turn, the visualizations provided by histograms and quantile dot plots are more flexible,

(a) KDE

(b) PIT ECDF
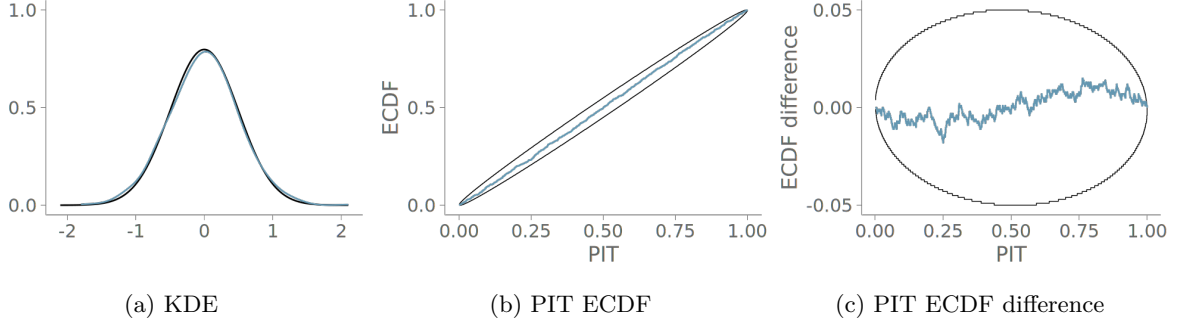
(c) PIT ECDF difference

Figure 3: Visualizing a sample from a smooth and unbounded density with a kernel density estimate. (a) the KDE of the sample in blue and the true density in black. (b) The corresponding PIT ECDF plot with 95% simultaneous confidence bands for uniformity. No goodness-of-fit issues are indicated by the plot as the PIT ECDF stays within the confidence bands. (c) The corresponding PIT ECDF difference plot, showing the deviation from the expected CDF when testing against the true distribution. The plot allows for a more dynamic use of the plotting area, making it easier to inspect the deviations from the expectation of the CDF.
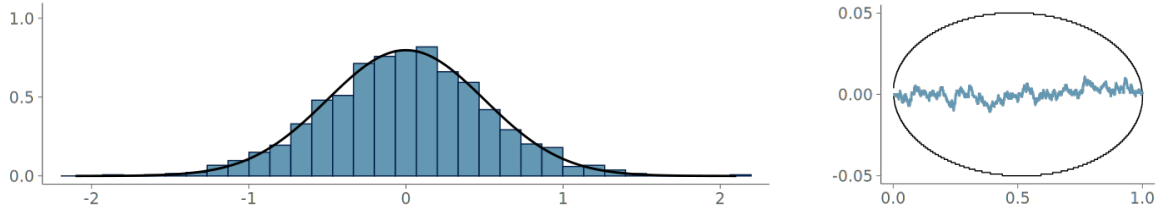
Figure 4: A histogram of the sample, with the smooth and unbounded true density overlaid in black. We use the rule by Freedman and Diaconis (1981) to determine the number of bins. The PIT ECDF difference plot indicates a good fit between the visualization and the sample.
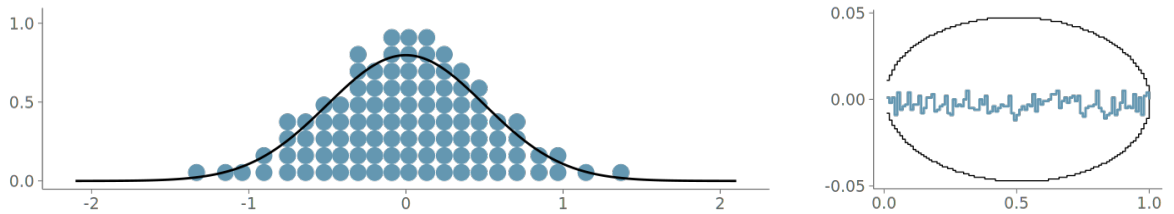
Figure 5: The quantile dot plot and the true density overlaid in black. As with all quantile dot plots in this article, we show 100 quantiles. The PIT ECDF difference plot indicates a good fit between the visualization and the sample.

but if the location of the step is not known, an unfortunate histogram binning may introduce large deviations to the true density values. If the location of the step was known, a good representation could also be obtained with KDE plots or histograms by using separate bounded KDEs for the two sides of the step (see Section 2.5), or by tailoring the histogram by aligning the discontinuity to the boundary of two adjacent bins.

To illustrate the challenges of visualizing observations from stepped densities, we use samples from the following true density f, shown in Figure 2a:

$$f(x) = \begin{cases} \frac{2}{5}\Phi(-\frac{1}{2})^{-1}\mathcal{N}(x \mid 0, 1), & x \leq -\frac{1}{2} \\ \frac{1}{5}, & -\frac{1}{2} < x \leq \frac{1}{2} \\ \frac{2}{5}\Phi(-\frac{1}{4})^{-1}\mathcal{N}(x \mid 0, \frac{1}{4}), & x > \frac{1}{2}, \end{cases} \tag{7}$$

this kind of bimodal skewed distribution with low density between the modes is present for example in the z-scores of published p-values (Van Zwet and Cator 2021).

Figure 6 shows the resulting density plot and the corresponding calibration for two common kernel bandwidth selection strategies; Silverman's rule of thumb (Silverman 2018), and the Sheather-Jones (SJ) method (Sheather and Jones 1991). Of these, SJ is expected to give a more robust bandwidth selection to data from non-Gaussian distributions (Sheather and Jones 1991). Despite this, Silverman's rule of thumb is the default strategy in many KDE density visualization implementations using Gaussian kernels. As seen in the figure, both strategies have difficulties representing the discontinuity in the observation density, and PIT ECDF for the KDE plot using Silverman's rule of thumb crosses the 95% simultaneous confidence interval and is recognized as having significant goodness-of-fit issues.

Figure 7 shows the visualization and goodness-of-fit assessment for the same data, when using a histogram. Although the discontinuity is strongly hinted in the histogram, it is not located close to a bin boundary and causes significant goodness-of-fit issues as too much density is placed on the last values of the low density region preceding the discontinuity.

Lastly, Figure 8 shows the same process for a quantile dot plot. Again, the discontinuities are visible in the plot. Here, as the visualization follows the ECDF quantiles, the discontinuities don't cause issues to the goodness-of-fit.

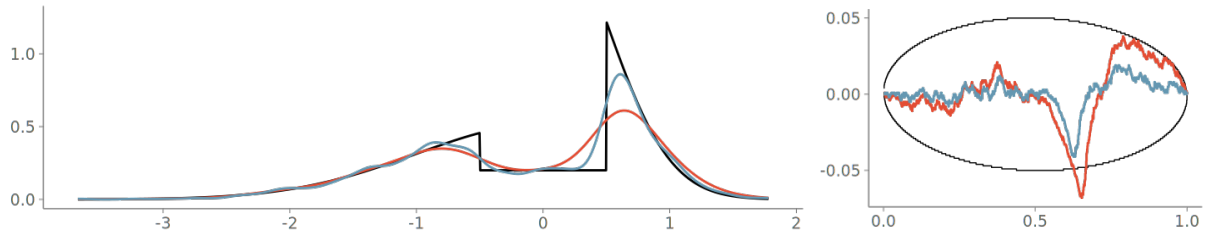

Figure 6: Two kernel density plots for a continuous valued sample from a density with steps. In red, a KDE with the bandwidth selected with Silverman's rule of thumb (Silverman 2018), and in blue a KDE using the Sheather-Jones bandwidth selection method (Sheather and Jones 1991), which results to a smaller bandwidth and a better fit to the sample.
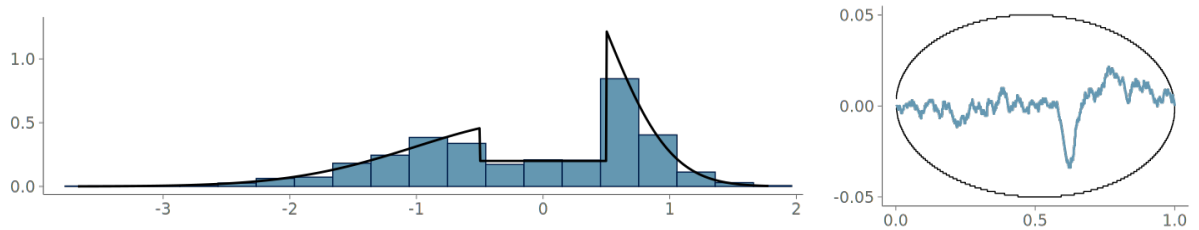

Figure 7: Visualizing the continuous valued sample from a density with steps with a histogram. Here, our bin width selection algorithm has resulted in an adequate fit to the data, although the local deviation from the expected CDF is clearly visible in the PIT ECDF plot.
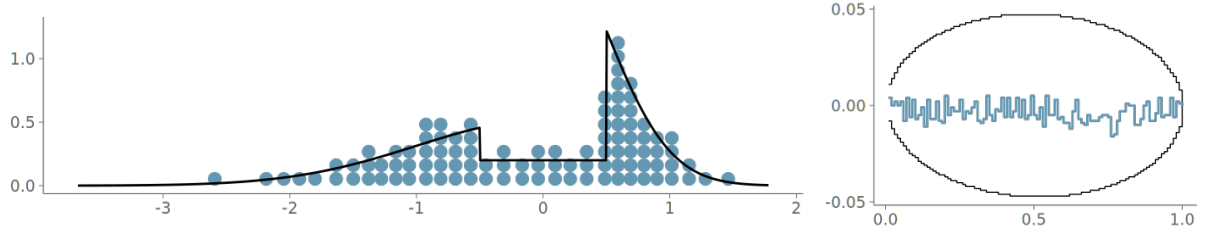
Figure 8: Visualizing the continuous valued sample from a density with steps with a quantile dot plot. The larger step is clearly visible, and PIT ECDF plot stays within the simultaneous confidence intervals. The 100 quantiles are able to adapt to the sample better than the histogram above with a relatively low number of bins obtained from the used binning algorithm.

2.5 Density with strict bounds

Bounded density functions are very commonplace, and a special case of the stepped densities discussed above. If the data is known to be bounded, the knowledge allows for specialized visualizations; for example, KDE plots with boundary corrections methods, such as boundary reflection, or limiting the histogram bins to the domain of the density.

A problem arises, when unknowingly visualizing bounded data. Below, we inspect our three visualizations of interest applied to data from a truncated exponential distribution with rate parameter $\lambda = 1$ (shown in Figure 2b). The truncation is to the central 80% interval of the untruncated distribution.

Figure 9 shows a comparison between two KDE plots; one without any information on the boundedness of the data, the other made using boundary reflection based on bounds estimated through an automated boundary detection algorithm implemented in the `ggdist` R package (Kay 2023). The goodness-of-fit test clearly indicates that the density is misrepresented near the boundaries by the unbounded KDE plot. In general, a ∩-shaped PIT ECDF plot indicates bias towards too small PIT values, and the strong upward trend at small PIT values indicates that the estimated density at the left tail is lower than expected if the data was sampled from the estimated density.

Figure 10 shows the data visualized with a histogram without limiting the bins to lie inside the bounds. Again, bins overlap the discontinuities, causing the goodness-of-fit test to indicate possible data misrepresentation.

Figure 11 shows the data visualized with quantile dot plot using 100 quantiles. As the quantile dot plot by design isn't placing any dots outside the data range, the goodness-of-fit is satisfactory.
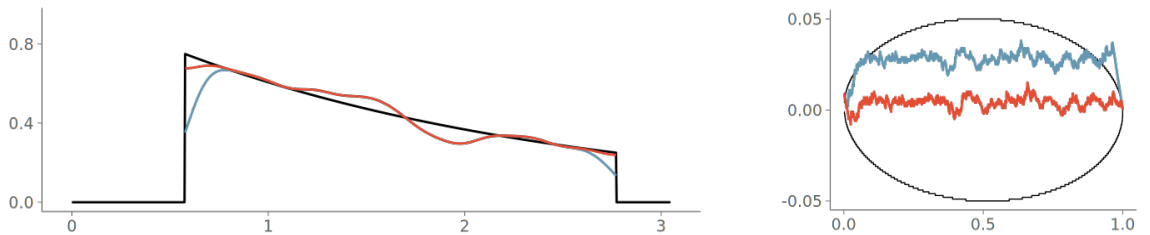

Figure 9: Two kernel density plots for a continuous valued sample from a bounded density. In red, a KDE using a boundary correction and automated boundary detection as implemented in the ggdist package for R (Kay 2023). The KDE in blue assumes unbounded data and uses no boundary corrections. Again, the misrepresentation of the sample close to the distribution boundaries is detected with the graphical goodness-of-fit test.

2.6 Density with point masses

The final example of continuous valued observations with discontinuities in the density is met when one or more point masses are present. These kinds of observation densities are often met in biomedical and economical studies, where zero-inflated, but otherwise continuous data is common (Liu et al. 2019). Another cause of point masses can be problematic treatment of missing values, where any missing fields in multidimensional observations are replaced with some predetermined value, be it zero or some summary statistic.
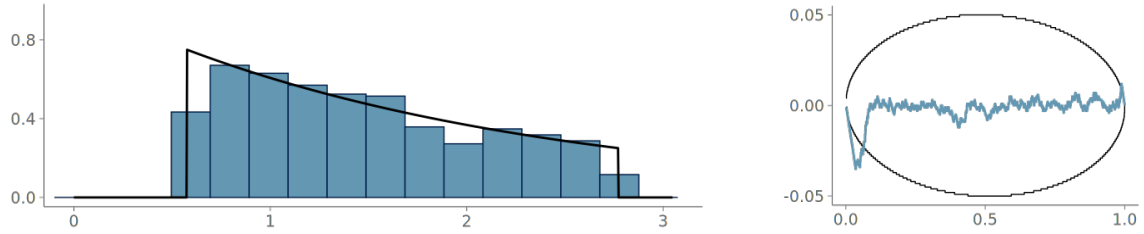
Figure 10: Visualizing the continuous valued sample from a bounded density with a histogram. The PIT ECDF line crosses the simultaneous confidence bands at the extreme PIT values, indicating issues in representing the boundaries of the data distribution. The drop in the PIT ECDF close to zero indicates underestimation of the left bound of the distribution, and respectively, the smaller upwards peak close to one indicates over estimation of the right bound.
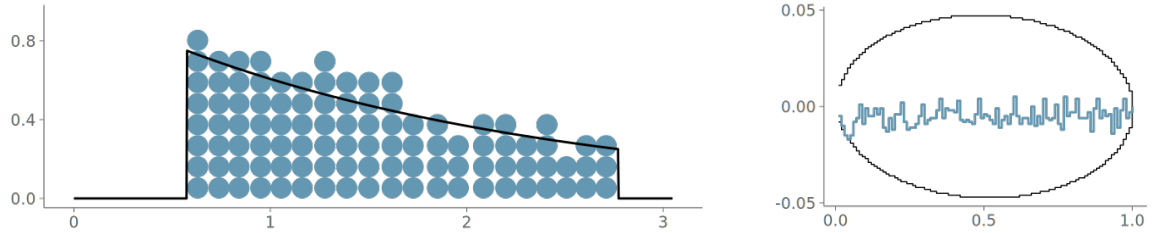

Figure 11: Visualizing the continuous valued sample from a bounded density with a quantile dot plot. The quantile dots are correctly placed within the bounds of the data distribution, making the quantile dot plot a good alternative for visualizing bounded data.

Below, we inspect an example following the density depicted in Figure 2c, where an observed value from the standard normal distribution is replaced with 1 with probability 0.2.

Figure 12 shows the KDE plot and the corresponding goodness-of-fit assessment, when the SJ method is used for bandwidth selection. Although the point mass does show as an additional bump in the density, the goodness-of-fit test notices that the KDE isn't flexible enough and alerts us of the underlying point mass with a sharp jump in the PIT ECDF.

Figure 13 shows the same data visualized using a histogram. Now the discontinuity is arguably more visible in the visualization, but, as all the bins are of equal width, the PIT ECDF shows a sharp discontinuity.

Figure 14 shows the data visualized with quantile dot plot using 100 quantiles. Again, the point mass is visible in the visualization, and the PIT ECDF indicates that the visualization is representative of the sample.
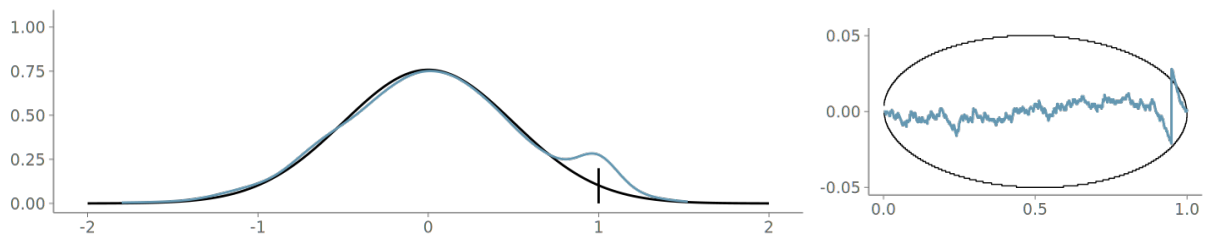

Figure 12: KDE for a continuous valued sample from a density with a point mass. The selected bandwidth is too large to adequately represent the point mass. This is detected by the goodness-of-fit test.

Detecting if observation is discrete or mixture of discrete and continuous

In addition to the aforementioned goodness-of-fit test, a fast and simple to implement method is to simply count unique values in the observed data. Relatively high counts of repeated values can inform the practitioner if the observations are discrete or contain point masses, for example, zero-inflation.

If at least one non-unique value, we move to check for relative frequencies of repeated values. If the relative frequency of any value is more than 2% of the full sample, we consider the possibility that the data might contain discrete values.
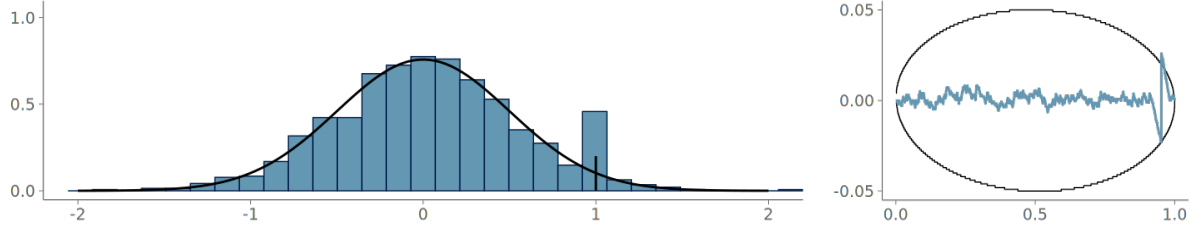
Figure 13: Histogram for the continuous valued sample from a density with a point mass. The selected bin width is too wide and the misrepresentation of the data is detected by the goodness-of-fit test.
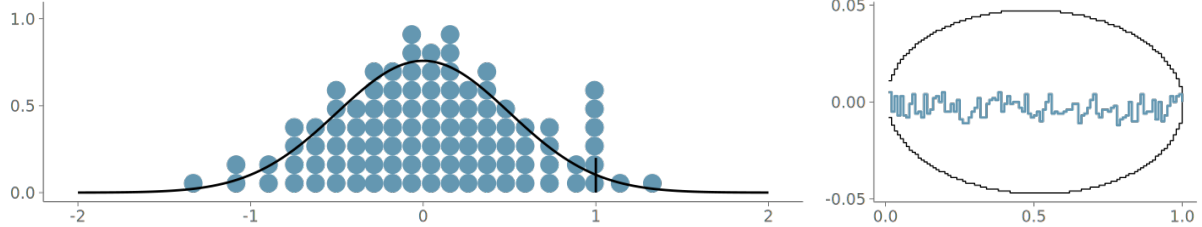

Figure 14: Quantile dot plot of the continuous valued sample from a density with a point mass. The quantiles dots are able to represent the point mass and the visualization passes the goodness-of-fit test.

2.7 Visual predictive checking with overlaid plots

When comparing data and model predictions, one approach is to show visualization of data and visualization of the draws from the predictive distribution overlaid in one plot. The most common approach is to plot overlaid KDE plots, as it is usually easy to distinguish several overlaid KDE lines from each other. The bandwidth selection plays an important role in the comparison. On one hand, if the chosen bandwidth is narrow, the KDEs can show large variation even when repeatedly drawing from the same distribution. On the other hand, with a wide bandwidth, the KDE may be too smooth and hide important details. With histograms, and quantile dot plots, overlaid plots are rarely used, as the bin boundaries and dot locations depend on the model prediction and the resulting overlaid figure becomes difficult to read. A common solution for histograms, shown later in Section 3, is to use a shared set of bins for the predictions and data, and to increase readability by only show the predictions through per bin summary statistics, such as mean and quantile based intervals. A drawback with this approach is, that the per bin summaries don't show dependency between the bins, making it harder to assess the global shape of the density. To compare multiple quantile dot plots to a reference plot, we propose overlaying the reference plot with just the top dots of each stack in the quantile dot plots of the predictive samples. Figure 15c shows the resulting plot, which illustrates the variation in the overall form of the predictive distribution. Again, when summarizing the quantile dot plots of the predictive draws, we lose information on the dependency between heights of the stacks and the overall shape of the predictive density.

Overlaid KDE plots are usually easy to interpret—assuming the smoothness assumptions match the underlying data and draws from the predictive distribution—although they lack a quantification for when the differences are significant. For better quantification of the difference and more robust behavior in the case of non-smooth underlying distributions, we recommend the use of the graphical PIT-ECDF test described in Section 2.1. In posterior predictive checks, we assess the fit between the observations and the predictive distribution of the model. The predictive distribution is not expected to give an exact match, for example the posterior predictive distribution of a Bayesian model usually has thicker tails than the true underlying observation distribution. To avoid using the same observations for both fitting the model and assessing the predictive performance, we use leave-one-out cross-validation (LOO-CV). Especially flexible models and small observations require LOO-CV for more reliable PIT values. If one were to not use cross-validation, the PIT values would tend to be too close to 0.5 and the PIT ECDF plot would be S-shaped, suggesting many observations are too close to their respective predictive means. For fast LOO-CV computation, we recommend using Pareto smoothed importance sampling (PSIS) (Vehtari et al. 2024), estimates the leave-one-out predictive distributions without the need for refitting the model for each left out observation. In our experience, the PIT ECDF plots of both the posterior predictive and LOO-predictive draws are useful for assessing that the predictive distribution of the model is well calibrated and the visualizations give insight to the nature of the possible issues in model fit.

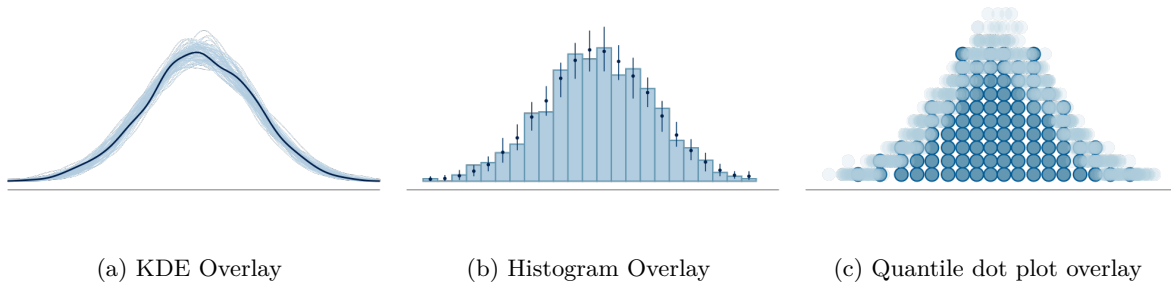| (a) KDE Overlay | (b) Histogram Overlay | (c) Quantile dot plot overlay |

Figure 15: Overlaid predictive checks. KDE plots can easily be overlaid and the resulting visualization is straightforward to read. The histogram for the data is overlaid with means and central 90% quantiles of the histogram bins of the posterior draws, when using the binning from the histogram of the observation. For the quantile dot plot, the top dot of each stack of dots in each posterior draw is overlaid to the full quantile dot plot of the observation, visualizing the variation in the shape of the predictive distributions.

3 Visual predictive checks for count data

In this section, we focus on count data, which although being discrete, can benefit from both continuous and discrete visualizations. Continuous visualizations are commonly used for count data and they can sometimes be a good choice, but there are cases where they can be misleading and visual predictive checks designed specifically for count data should be used.

In predictive checking for count data, we focus on two visualizations; the overlaid KDE plots introduced in Section 2.7, and our version of the rootogram (Kleiber and Zeileis 2016). In our experience, the higher the number of distinct counts being visualized, and the smaller the variance of probabilities of individual counts is, the more effective KDEs are for visualizing count data, although one should pay close attention to the effect of the boundedness on the KDE fit as count data is usually limited to non-negative values and sometimes also has maximum count limit. Rootograms offer a visualization which emphasizes the discrete nature of the predictive and observation distributions. In our experience, rootograms are good for predictive checking of count data, when the number of distinct counts is low or there is a sharp changes in the probabilities of consecutive counts, for example, when the data exhibits zero-inflation (the probability of 0 is much higher than the probability of 1).

In Section 3.1 we show how, similarly to the cases in Section 2, goodness-of-fit tests can be used to assess how well the KDE plot represents the data. In Section 3.2 we introduce our version of the rootogram, and demonstrate its use when the familiar KDE exhibits goodness-of-fit issues. The visual predictive checks in both Section 3.1 and Section 3.2 follow an example from a modeling workflow of estimating the effect of integrated pest management on reducing cockroach levels in urban apartments (Gelman, Hill, and Vehtari 2020). We focus on assessing the quality of the predictions of a generalized linear model with a negative binomial data model for the number of roaches caught in traps during the original experiment.

3.1 Visualizing count data with KDE plots

A common approach for count data, especially when the number of unique discrete values is high, is to assume that the values can be visualized as if they were from a continuous distribution. This enables the use of overlaid KDE plots for visual predictive checks. Figure 16 and Figure 17 illustrate how—after changing the default bandwidth algorithm—a KDE plot can have a satisfactory goodness-of-fit to discrete data, and how the PIT ECDF diagnostic can be used to assess the goodness-of-fit of the continuous visual representation of the discrete data.

The data in the roach dataset exhibits zero-inflation, and the default algorithm of the selected visualization software fits an unbounded KDE with a relatively large bandwidth, resulting in a bad fit shown in Figure 16. After using a left bounded KDE and the SJ bandwidth selection method, we see in Figure 17 that away from zero, the observation distribution is well summarized by the KDE plot.

A reasonable step in the visual predictive checking for the roaches model, is to separate the analysis of the long tail and the bulk of the distribution. One possible approach for analyzing the tail of the distribution, is to compare how the estimated Pareto shape parameter of the predictive draws compares to the tail shape of the data. When comparing the tail shapes, one shouldn't expect an exact fit, as the predictive distributions typically exhibit thicker tails due to uncertainty. When moving to analyze the bulk of the distribution, we see that the number of distinct counts below 90% quantile is relatively small, 48, and
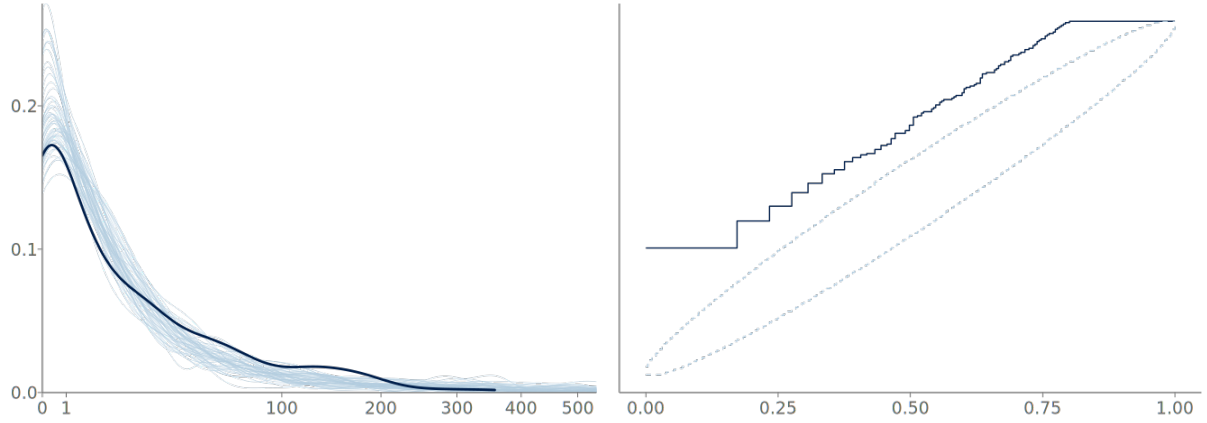
Figure 16: An overlaid KDE plot of the observed roach counts and the predictive samples. The KDE density uses the default predictive check implementation of `bayesplot`, and is not bounded and has a relatively large kernel bandwidth. The PIT ECDF shows that the visualization assigns too much probability mass to small counts, and between the discrete integer valued observations, resulting in the PIT ECDF staying above the simultaneous confidence bands.
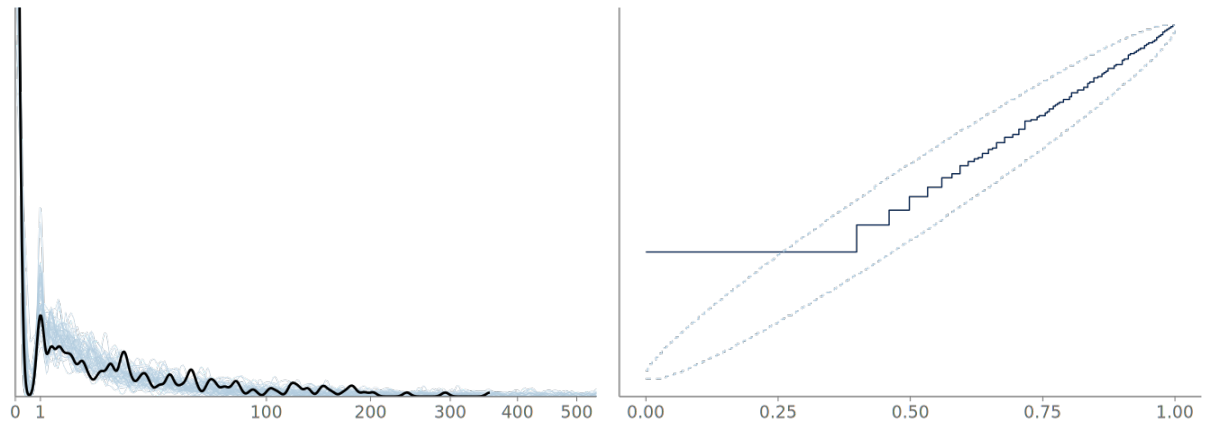

Figure 17: An overlaid KDE plot of the observed roach counts and the predictive samples. The KDE is bounded at zero and uses the SJ method for bandwidth selection, resulting in a better fit to the discrete data. Due to zero inflation, there are numerous ties in the PIT values and the PIT ECDF is outside the simultaneous confidence bands at zero. We see that for larger observed counts, the visualization represents the data well.

thus we may expect a discrete visual predictive check to perform well.

3.2 Rootograms emphasize discreteness of count data

(a) Standing Rootogram   (b) Hanging Rootogram   (c) Suspended Rootogram
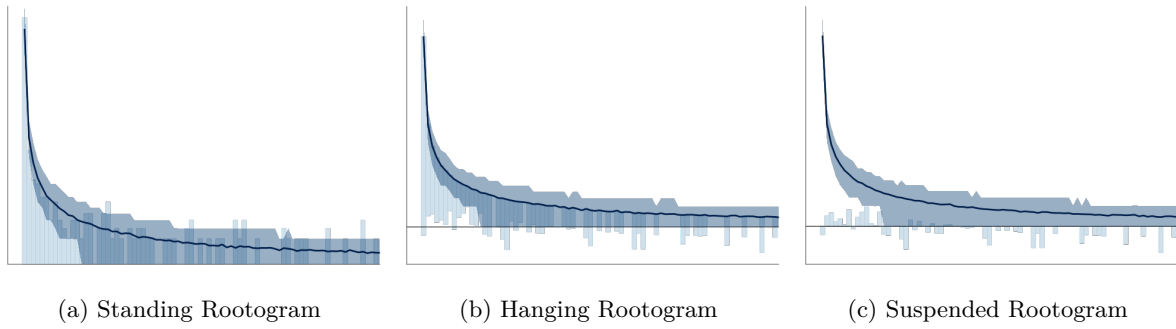
Figure 18: The three types of rootograms introduced by Kleiber and Zeileis (2016), here with an additional darker ribbon representing the predictive uncertainty. a) The standing rootogram displays a bar graph of the observed counts overlaid with the predictive expectations and their 90% credible intervals. b) In the hanging rootogram, the bar plot of the observed counts is hanging from the predictive expectation, visualizing the average predictive error as the empty space between the x-axis and the bars. c) In the suspended rootogram, only the difference between the prediction and the observation is displayed as bars.

Kleiber and Zeileis (2016) proposed rootograms to be particularly useful in diagnosing issues, such as over-dispersion and excess zeros, in count data models.

Traditionally, in a rootogram the frequencies of the observed counts are visualized with a bar plot, and predictive expected frequencies as a connected line plot. Importantly, the frequencies on the vertical axis are plotted on a square root scale to emphasize counts with low frequencies, occurring often in the tails of the distribution (see for example, Tukey 1972; Kleiber and Zeileis 2016). As shown in Figure 18a, one can include the visual representations of the predictive uncertainty without overly complicating the visualization. Aside from the *standing* rootogram in figure Figure 18a, two common variations are the hanging and suspended rootogram, shown in Figure 18b and Figure 18c respectively. These alternatives offer a more pattern focused approach to inspecting the possible differences between the observed and predicted values.

In the hanging rootogram, the observed frequencies are drawn hanging form the predictive mean, placing the lower end of the bar at the difference of the predicted mean and the observation, $y_{pred} - y_{obs}$. This end is then directly readable as over- or underestimation, depending if the bar is above or below the horizontal axis.

The suspended rootogram instead visualizes the residual, $y_{obs} - y_{pred}$, as bars with their other end at the horizontal axis of the visualization.

Figure 19b shows our proposed modification to the traditional rootogram. This version has two main differences to the previously used visualizations; first, it emphasizes the discreteness of the predictions by showing the predictive frequencies as points and point-wise credible intervals, instead of connecting these into lines and filled areas. For overlaying, we prefer to show the observations as points and use color to highlight the observations falling outside the credible intervals. Although some values are expected to fall outside the credible intervals, these are often of interest to the practitioner for identifying possible issues. The second important distinction is how we implement the square root scaling on the vertical axis—instead of the traditional method of having an axis with equally spaced tick marks displaying the square roots of the frequencies—we transform the axis to square root scale and display the untransformed frequencies (see comparison in Figure 19). We recommend the use of axis ticks or horizontal lines as visual cues of the non-linear scaling. This second change allows for a more natural reading of the frequencies shown on the vertical axis of the plot.

4 Visual predictive checks for binary data

Assessing the calibration of binary predictions is a common task made difficult by the discreteness of the observed outcome. Below, we walk through two usual approaches, and introduce an alternative offered by PAV-adjusted calibration plots (Dimitriadis, Gneiting, and Jordan 2021).

We continue using the roaches example introduced in Section 3, and consider three models for predicting if one or more roaches will be observed:
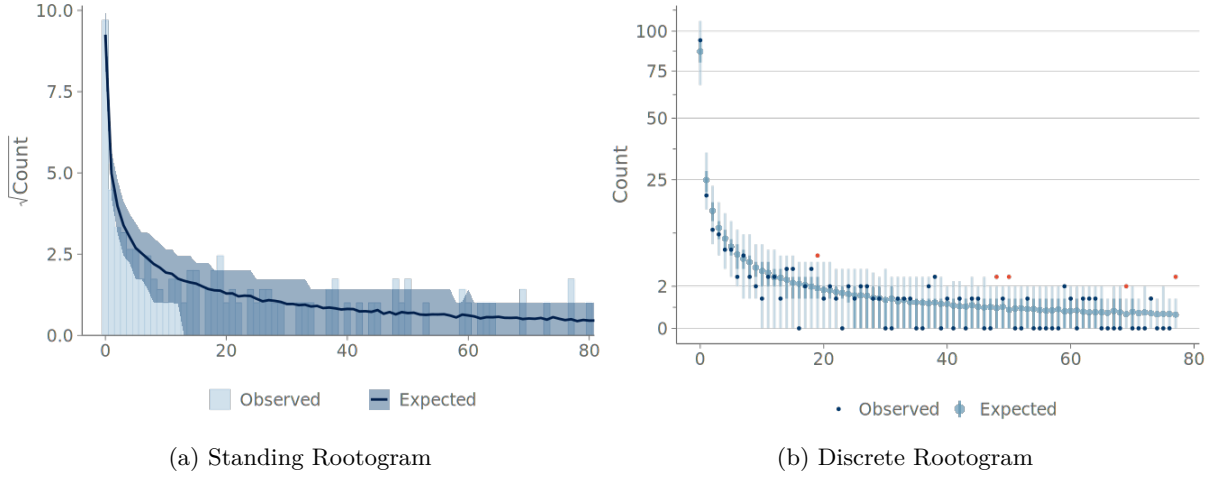
(a) Standing Rootogram

(b) Discrete Rootogram

Figure 19: A side-by-side comparison of the commonly used standing rootogram with credible intervals, and our suggested version which emphasizes discreteness of count data. a) Displays the square roots of the count frequencies, and visualizes the model predictions through a continuous line and ribbon. b) Displays the unmodified counts on the vertical axis, while using square root scaling for the axis, thus enabling direct reading of the count frequencies. The predictions are displayed as light blue points and interval lines. The observations are overlaid with darker points, except the observations outside the predictive credible intervals are highlighted in red. Again, some points are expected to fall outside the credible intervals.

(a) Bar graph

(b) Binned calibration plot

(c) PAV-adjusted calibration plot
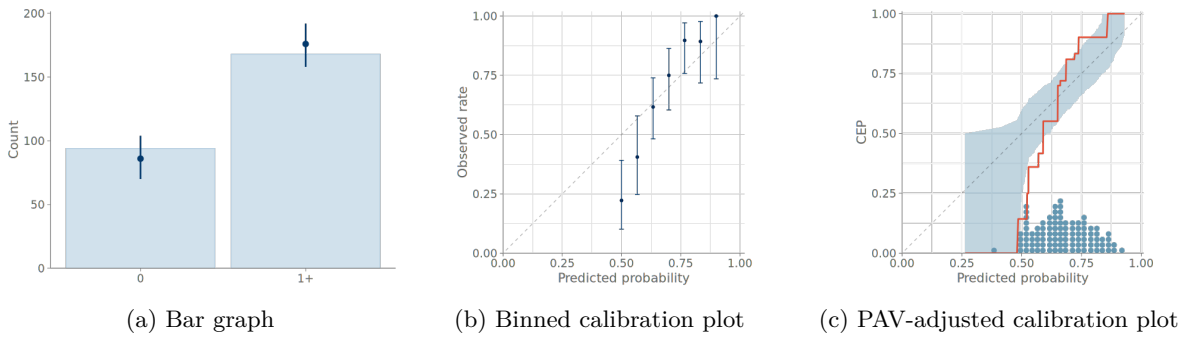
Figure 20: Calibration of estimated probability of observing a non-zero number of roaches. (a) & (b) commonly used bar and binned calibration plots, (c) the PAV-adjusted calibration plot, with 95% consistency bands computed from predictive samples.

1. an intercept-only model estimating the overall probability of observing one or more roaches,
2. the negative binomial regression model from Section 3.
3. a zero-inflated version of Model 2 using the covariates to additionally model the probability of not observing any roaches.

4.1 Bar graphs

A bar graph consists of bars representing the frequencies of the observed values, and point intervals of the predictive means and central quantiles. These elements are positioned over each state of the observation distribution. Bar graphs are a common choice when summarizing discrete binary, categorical, or ordinal data and predictions. Although bar graphs can work well for predictive checking as part of rootgrams for count data, they are less useful when there are a small number of discrete values. In the extreme case of binary data, even a model with one parameter corresponding to the proportion of one class, can perfectly model the proportion, and the overlaid posterior predictive checking bar plots are useless (and KDE plots are even worse with misleading smoothing). Additionally, as shown in Figure 21 and Figure 22 below, any assessment that is more advanced than inspecting the relative frequencies of the discrete states, is impossible with the bar graphs. For example, under- and overconfidence, and symmetric confusion between states, is not visible in a graph.

Figure 20a shows bar graphs summarizing the predictions of the intercept-only and negative binomial models for predicting the binary outcome of observing one or more roaches. Both bar graphs show the model predictions agreeing with the observations, and no clear difference in the predictions of the models can be seen.

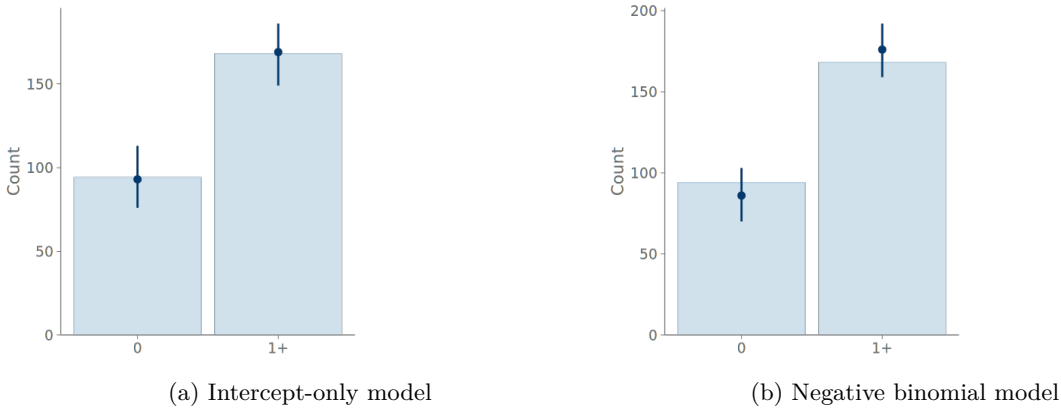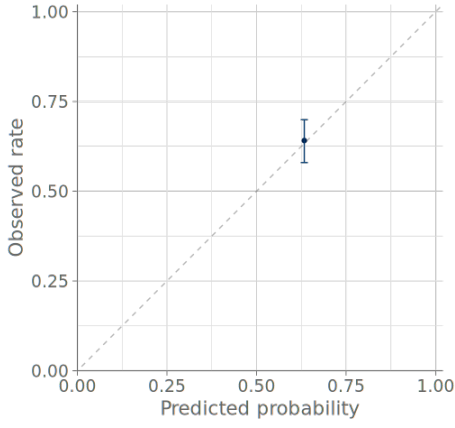(a) Intercept-only model   (b) Negative binomial model

Figure 21: We cannot see a difference in the predictive performance of the two models from the bar graphs. For both models, the observed event frequency falls within the 95% credible interval with the mean predicted frequency of the intercept only model aligning exactly with the observation. Bar plots are almost always useless for predictive checking in case of binary data.

4.2 Binned calibration plots

What could be argued to be the first step towards assessing calibration of the predictive probabilities in binary prediction tasks is the use of binned calibration plots (Niculescu-Mizil and Caruana 2005), also called calibration curves or reliability diagrams (DeGroot and Fienberg 1983). In these visualizations, the binary observations are divided into a predetermined number of uniform bins, based on the event probabilities predicted by the model. Comparing the mean predicted probability to the observed event rate within each bin allows assessing the calibration, or reliability, of the model. Typically, binned calibration plots include a binomial confidence intervals of the observed event rates. In practice, binned calibration plots suffer from similar drawbacks as histograms; the choice of binning is up to the user, and can cause the resulting plots to look drastically different, as shown by (Dimitriadis, Gneiting, and Jordan 2021).

Figure 20b shows the binned calibration plots for the predictive distributions visualized previously through bar graphs. It is immediately clear that even though the probabilities predicted by the intercept-only model are calibrated, the model has no ability to discriminate between the cases where one is more or less likely to observe roaches. We can also see, that although the negative binomial model is better at discriminating the cases, it suffers from calibration issues and is especially under-confident at predicting cases where no roaches will be observed.

(a) Intercept-only model
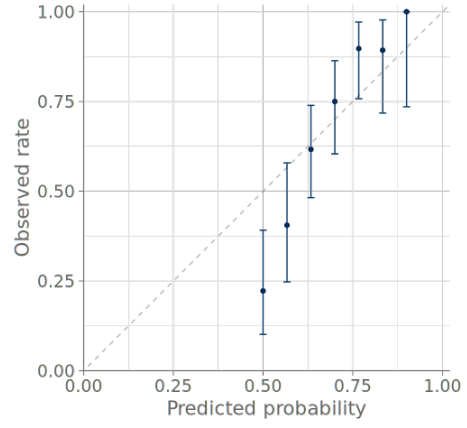
(b) Negative binomial model

Figure 22: The intercept only model has no ability to discriminate between the observations, while the negative binomial model can recognize cases where we are more likely to observe roaches. Still, the negative binomial model the model is under-confident in predicting cases of no roaches observed.

4.3 PAV-adjusted calibration plots

The ad-hoc binning decision has the potential to play a major role in the conclusions drawn from binned calibration plots. To address this, Dimitriadis, Gneiting, and Jordan (2021) propose a method where the observed binary events are replaced with conditional event probabilities (CEP), that is the probability that a certain event occurs given that the classifier has assigned a specific predicted probability. To compute the CEPs, the authors use the pool adjacent violators (PAV) algorithm (Ayer et al. 1955), which provides a non-parametric maximum likelihood solution to the problem of assigning CEPs that are monotonic with respect to the model predictions. This monotonicity assumption is reasonable for calibrated models, where higher predicted probabilities should correspond to higher actual event probabilities.

To aid in the calibration assessment, we employ the point-wise consistency bands introduced by (Dimitriadis, Gneiting, and Jordan 2021). These intervals show, how much variation should be expected from the calibration plot even if the model was perfectly calibrated.

Figure 20c shows this *PAV-adjusted calibration plot* with 95% point-wise consistency bands computed from the posterior predictive distribution. First, we compute the CEPs using the posterior predictive means for each observation. Then we form the consistency bands by using the corresponding posterior predictive draws to determine the 95% credible intervals for CEPs of draws from the predictive distribution.
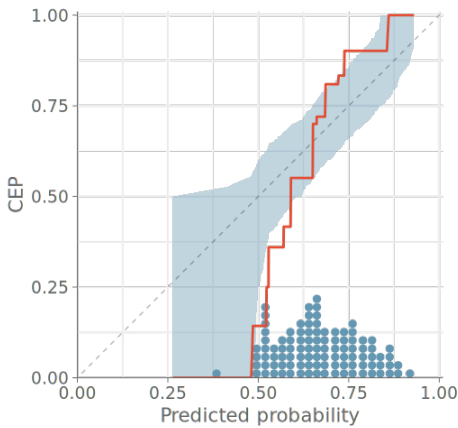
(a) Negative binomial model
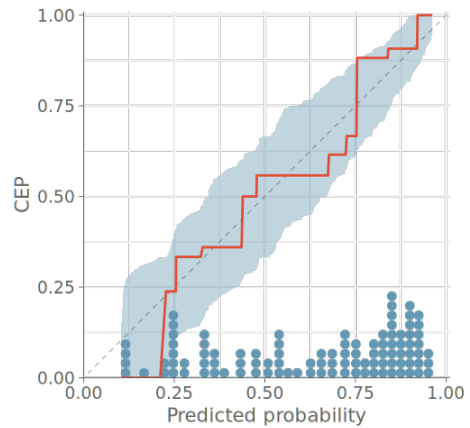
(b) Zero-inflated model

Figure 23: Comparison of two models for predicting if roaches will be observed. As noted before, the negative binomial model is under confident when predicting cases where no roaches are observed. In the PAV-adjusted calibration plot of the zero-inflated negative binomial model, we see the calibration curve staying within the consistency bands, and also observe a wider range of predicted event probabilities. Together these two qualities indicate the zero-inflated model to be better calibrated and additionally have higher ability to discriminate the observations with no roaches.

4.4 Residual plots

While binned and PAV-adjusted calibration plots are useful for assessing the overall predictive calibration of a model, the behavior conditional on individual predictors is often of interest to practitioners.

Residual plots are commonly deployed for assessing the deviation between observations and predictions (Gelman, Hill, and Vehtari 2020; Gelman et al. 2013; Agresti 2013). As with binned calibration plots, binning is used for transforming discrete outcomes into relative event frequencies.

Again, the ad-hoc choice of binning may result in vastly different plots, and thus we propose the use of PAV-adjusted CEPs, computed with regards to the predicted probabilities, in place of the discrete observations. Figure 24 shows an example of a PAV-adjusted residual plot for the roaches example.
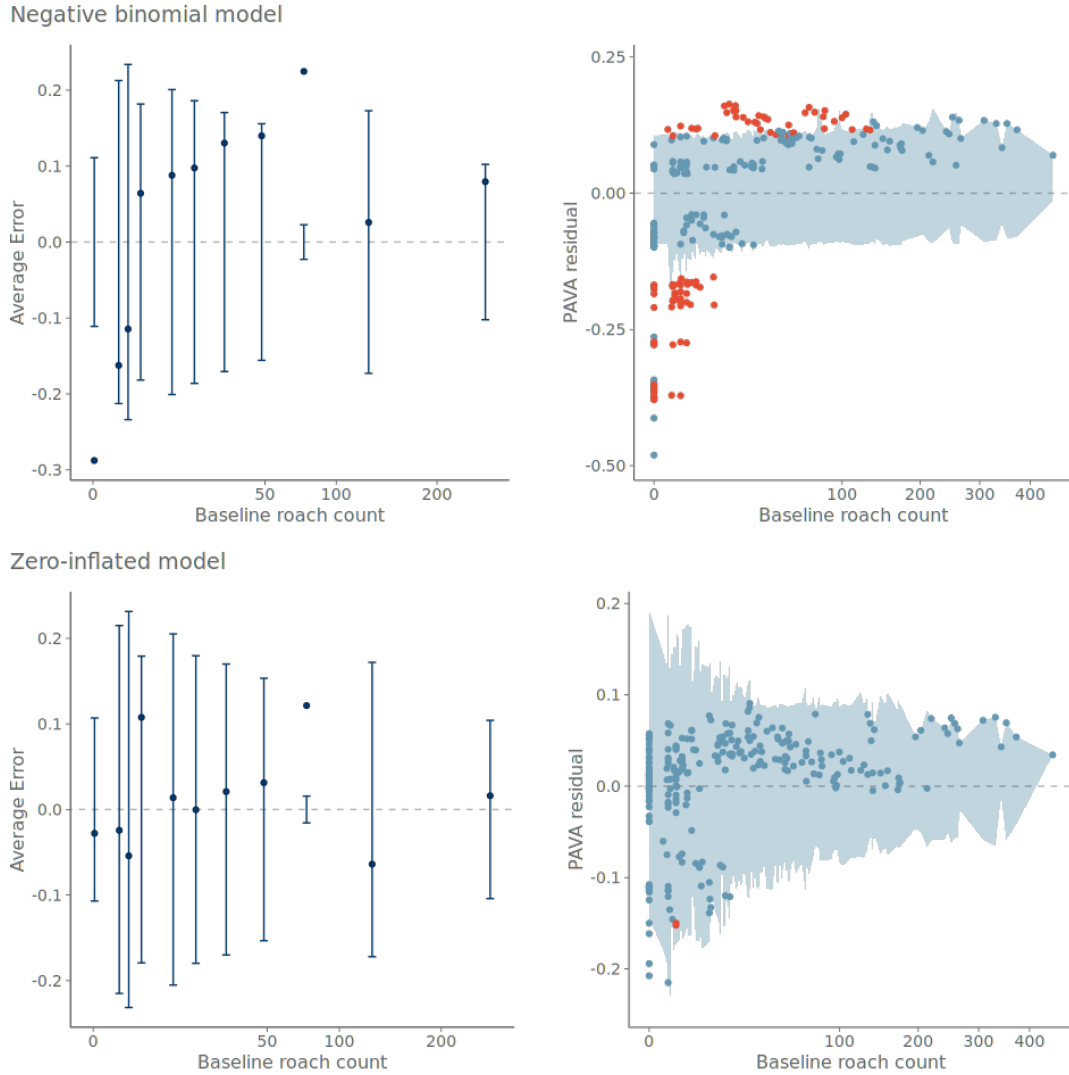

Figure 24: Residual plots of the two models against the baseline roach count observed at the start of the study. The binned calibration plots on the left are commonly used, while the PAV-adjusted residuals on the right offer a non-parametric alternative that doesn't require a binning choice. Here, we use the consistency bands of the PAV-adjusted calibration plot in Figure 23 and highlight the observations falling outside these intervals. From both visualizations, one can see the tendency of the simpler model without zero-inflation to overestimate the probability of observing roaches, when the baseline roach count is very low, as well as the tendency of overestimating the probability of observing roaches after the treatment in low to medium baseline counts.

5 Visual predictive checks for categorical data

When the number of categories is relatively low, the methods presented for the binary case in Section 4 extend to the categorical case. In the same way as for binary data, even the simplest model is likely to have such parameters that the predictive probabilities match the observed frequencies and bar plots are

useless. However, we can use the calibration of the binomial one-versus-others (OVO) distributions for each of the categorical cases. With a large number of categories, investigating one plot for each OVO calibration can be unwieldy, but scalar summaries of miscalibration for each category could be used to filter the set of individual cases before graphical inspection.

Figure 25 shows a predictive check using bar graphs for a predictive model with three distinct categories. Figure 26 shows both binned and PAV-adjusted calibration plots for the same model. No immediate calibration issues are visible for the model when predicting category A, but one can also see that cases, where the model would estimate the probability of any single category to be larger than 0.7 are rare – only one to three quantile dots are at values larger than 0.7. Additionally, both plots types show that the model confuses classes B and C, and is over confident in its predictions concerning these classes.
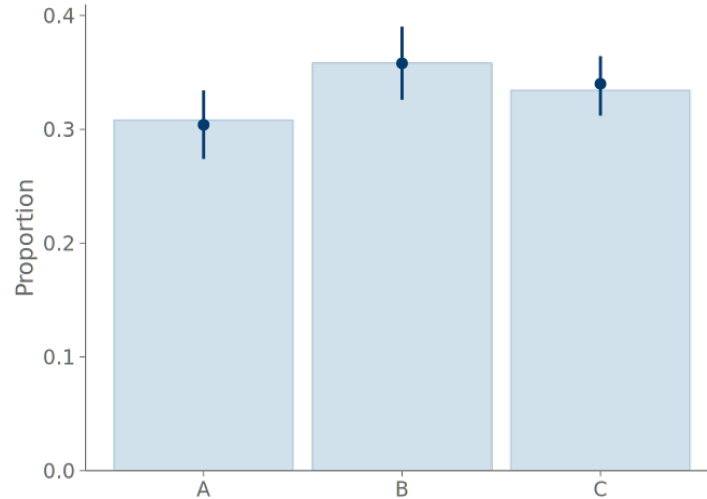

Figure 25: A bar graph of the observed counts overlaid with posterior predictive means and 95% credible intervals. As in the earlier examples of using bar charts, everything seems to be in order with the model predictions.

6   Visual predictive checks for ordinal data

Models for ordinal data often have such parameters that even the simplest model predictive probabilities match the observed frequencies and bar plots are useless (KDE plots are even worse with misleading smoothing). In the case of ordinal predictions, instead of looking at the OVO distributions, we can assess the calibration of the cumulative conditional event probabilities. This allows us to again use the toolset presented for binomial predictive checks in Section 4, now resulting in $M - 1$ plots, where M is the number of possible distinct discrete cases.

Figure 28 shows a PPC using bar graphs for two models without any sign of calibration issues in either predictive distribution. Figure 29 shows the PAV-adjusted calibration plots for each cumulative event probability. The diagrams for model 1 on the left show a calibration issue, with the calibration curves exhibiting an S-shape, indicative of under confidence in predicted probabilities.

7   Discussion

In this paper we discussed the current options for graphical predictive checking in Bayesian model checking workflows. As discussed, the type of data being modelled, and the type of model being applied are key considerations when choosing a particular approach. We strongly recommend that practitioners carefully consider the underlying assumptions being made by a chosen visualization and perform some form of diagnostic-check, such as the discussed PIT-ECDF, if unsure.

To summaries our recommendations, KDE plots are primarily only recommended for when a modeller is confident that the distribution is smooth (no point-masses or steps), or if there are many unique discrete values. If a KDE plot is going to be used for a bounded distribution, boundary corrections methods should be used, and we still recommend goodness-of-fit checking. Failing this, if the distribution has a continuous component, we recommend a quantile dot plot. For count distributions, we recommend a modified rootogram, plotting points and credible intervals instead of smoothed limits. For unordered discrete data, we recommend using a PAV-adjusted One-Versus-Others calibration plot. Finally, for
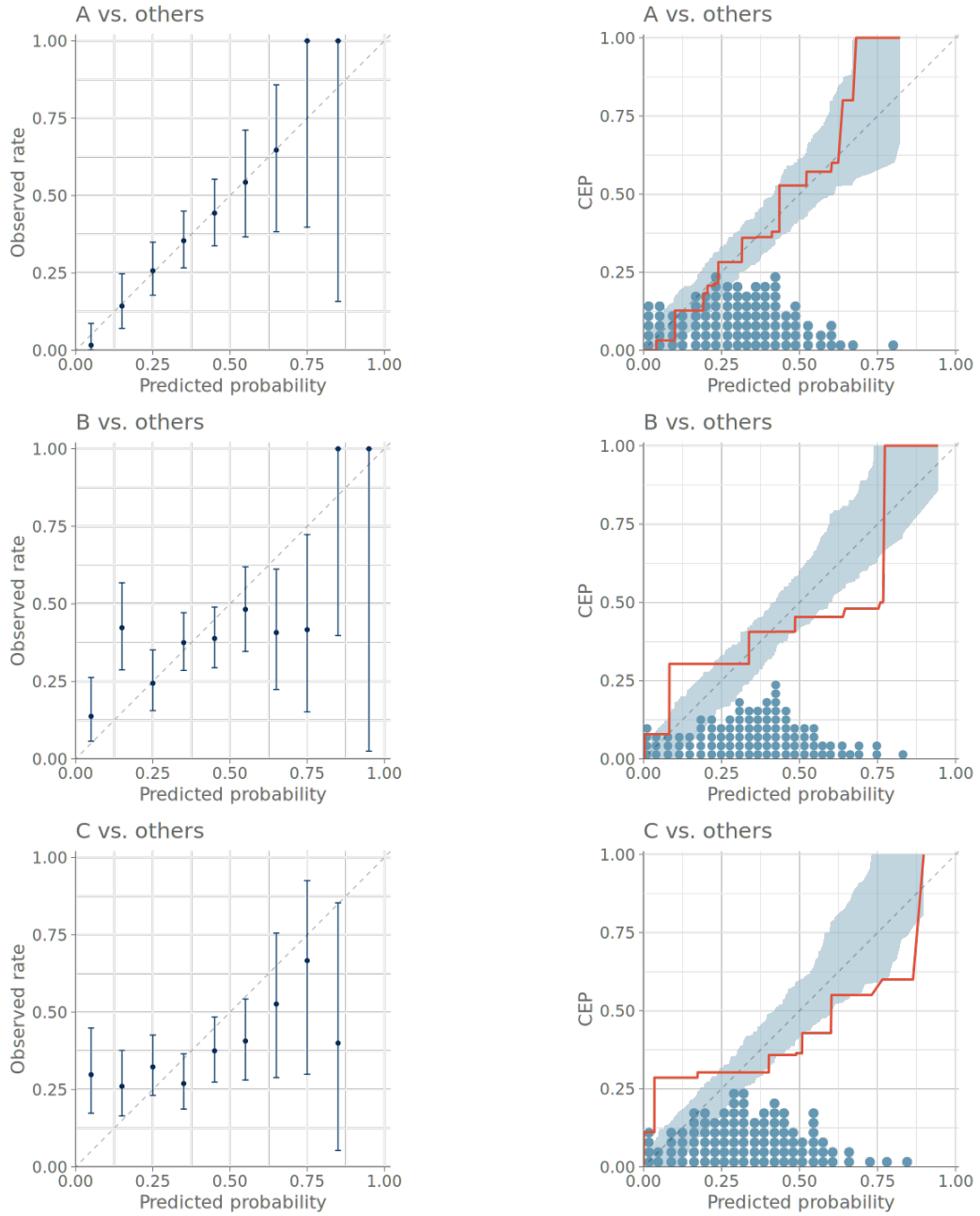
Figure 26: Comparison between binned calibration plots and PAV-adjusted calibration plots for one versus others comparisons. We see that the predictions are well calibrated for class A, but the model seems to be confusing observations of classes B and C. The sharp rises at the extreme predictions indicate that the model is over-confident in its predicted probabilities for these classes.
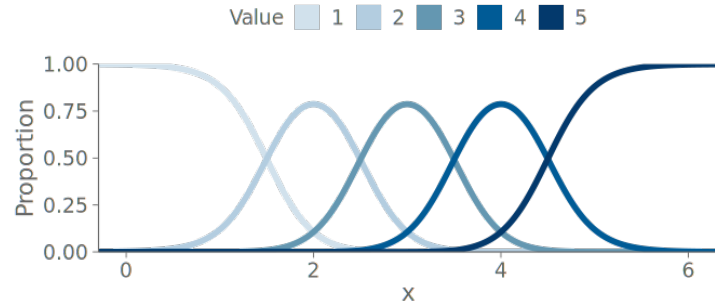
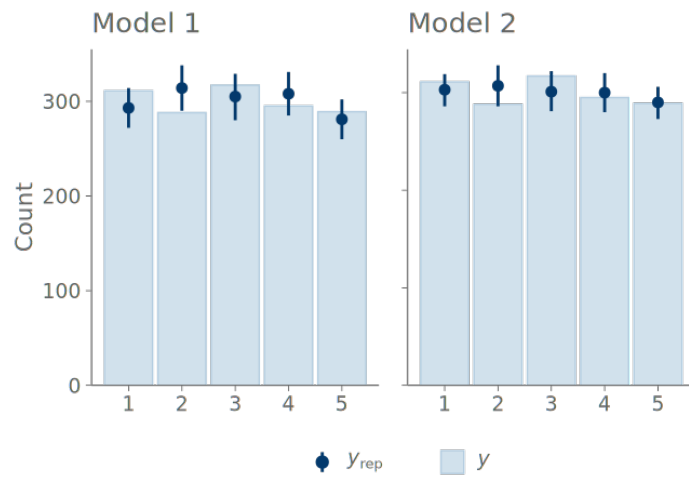Figure 27: True proportions of the five possible values given value of the covariate x.

Figure 28: Again, the bar graph doesn't indicate meaningful difference between the predictive distributions.
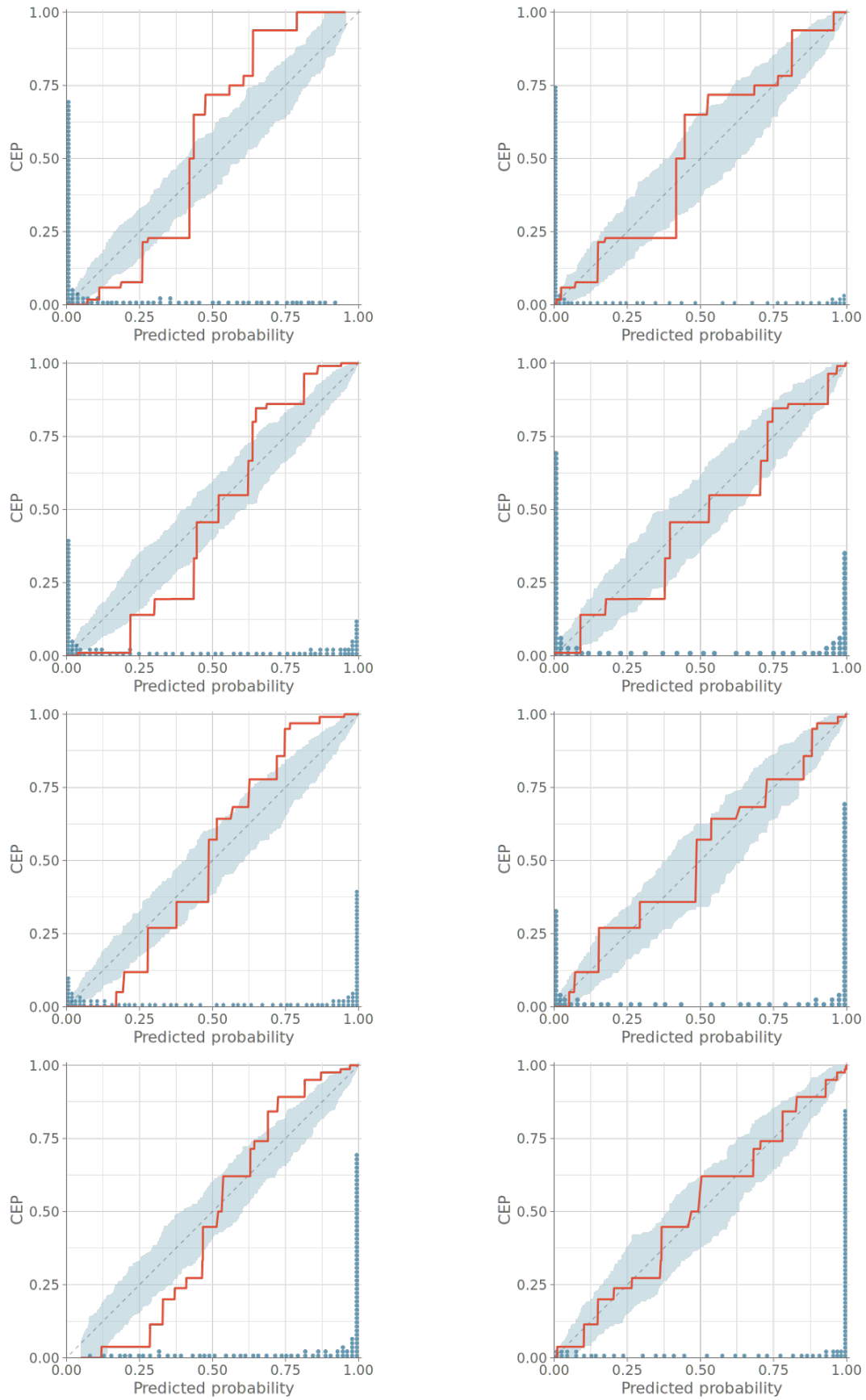
Figure 29: The side-by-side comparison of the PAV-adjusted calibration plots for the cumulative conditional event probabilities reveal that Model 1, containing an implementation error, is under confident in its predictions.

ordered discrete data, we also recommend a PAV-adjusted calibration, but evaluating the cumulative conditional event probabilities.

It must be emphasized that these are only general recommendations. As we have presented above, these recommended plots may still fail to identify miscalibration or other goodness-of-fit issues in some situations. Careful consideration of model assumptions and application of diagnostics should always form part of the modeller's standard practice. Many of our recommendations can be taken into account in the visual predictive checking software, for example, by giving a warning if a user tries to use overlaid KDE plot or bar plot for checking a model in case of binary data.

Acknowledgements

Agresti, Alan. 2013. *Categorical Data Analysis*. Third edition. Wiley Series in Probability and Statistics. Hoboken, New Jersey: Wiley-Interscience.

Ayer, Miriam, H. D. Brunk, G. M. Ewing, W. T. Reid, and Edward Silverman. 1955. "An Empirical Distribution Function for Sampling with Incomplete Information." *The Annals of Mathematical Statistics* 26 (4): 641–47. https://doi.org/10.1214/aoms/1177728423.

Box, George E. P. 1980. "Sampling and Bayes' Inference in Scientific Modelling and Robustness." *Journal of the Royal Statistical Society. Series A (General)* 143 (4): 383. https://doi.org/10.2307/2982063.

Bürkner, Paul-Christian. 2017. "brms: An R Package for Bayesian Multilevel Models Using Stan." *Journal of Statistical Software* 80 (1): 1–28. https://doi.org/10.18637/jss.v080.i01.

DeGroot, Morris H., and Stephen E. Fienberg. 1983. "The Comparison and Evaluation of Forecasters." *The Statistician* 32 (1/2): 12. https://doi.org/10.2307/2987588.

Dimitriadis, Timo, Tilmann Gneiting, and Alexander I. Jordan. 2021. "Stable Reliability Diagrams for Probabilistic Classifiers." *Proceedings of the National Academy of Sciences* 118 (8): e2016191118. https://doi.org/10.1073/pnas.2016191118.

Freedman, David, and Persi Diaconis. 1981. "On the Histogram as a Density Estimator:L 2 Theory." *Zeitschrift Für Wahrscheinlichkeitstheorie Und Verwandte Gebiete* 57 (4): 453–76. https://doi.org/10.1007/BF01025868.

Früiiwirth-Schnatter, Sylvia. 1996. "Recursive Residuals and Model Diagnostics for Normal and Non-Normal State Space Models." *Environmental and Ecological Statistics* 3 (4): 291–309. https://doi.org/10.1007/BF00539368.

Gabry, Jonah, Rok Češnovar, Andrew Johnson, and Steve Bronder. 2024. *cmdstanr: R Interface to 'CmdStan'*. https://mc-stan.org/cmdstanr/.

Gabry, Jonah, and Tristan Mahr. 2024a. "Bayesplot: Plotting for Bayesian Models." https://doi.org/10.32614/CRAN.package.bayesplot.

———. 2024b. "Bayesplot: Plotting for Bayesian Models." https://doi.org/10.32614/CRAN.package.bayesplot.

Gabry, Jonah, Daniel Simpson, Aki Vehtari, Michael Betancourt, and Andrew Gelman. 2019. "Visualization in Bayesian Workflow." *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 182 (2): 389–402. https://doi.org/10.1111/rssa.12378.

Gelman, Andrew, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. 2013. *Bayesian Data Analysis*. 0th ed. Chapman; Hall/CRC. https://doi.org/10.1201/b16018.

Gelman, Andrew, Jennifer Hill, and Aki Vehtari. 2020. *Regression and Other Stories*. 1st ed. Cambridge University Press. https://doi.org/10.1017/9781139161879.

Gelman, Andrew, Aki Vehtari, Daniel Simpson, Charles C. Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. 2020. "Bayesian Workflow." *arXiv:2011.01808 [Stat]*, November. http://arxiv.org/abs/2011.01808.

Gelman, Andrew, Meng Xiao-Li, and Hal S. Stern. 1996. "Posterior Predictive Assessment of Model Fitness via Realized Discrepancies." *Statistica Sinica* 6 (4): 733–60. https://www.jstor.org/stable/24306036.

Goodrich, Ben, Jonah Gabry, Imad Ali, and Sam Brilleman. 2024. "rstanarm: Bayesian Applied Regression Modeling via Stan." https://doi.org/10.32614/CRAN.package.rstanarm.

Kay, Matthew. 2023. "Ggdist: Visualizations of Distributions and Uncertainty." Zenodo. https://doi.org/10.5281/zenodo.3879620.

Kay, Matthew, Tara Kola, Jessica R. Hullman, and Sean A. Munson. 2016. "When (Ish) Is My Bus?: User-Centered Visualizations of Uncertainty in Everyday, Mobile Predictive Systems." In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 5092–5103. San Jose California USA: ACM. https://doi.org/10.1145/2858036.2858558.

Kleiber, Christian, and Achim Zeileis. 2016. "Visualizing Count Data Regressions Using Rootograms." *The American Statistician* 70 (3): 296–303. https://doi.org/10.1080/00031305.2016.1173590.

Kuhn, and Max. 2008. "Building Predictive Models in R Using the caret Package." *Journal of Statistical Software* 28 (5): 1–26. https://doi.org/10.18637/jss.v028.i05.

Kumar, Ravin, Colin Carroll, Ari Hartikainen, and Osvaldo Martin. 2019. "ArviZ a Unified Library for Exploratory Analysis of Bayesian Models in Python." *Journal of Open Source Software* 4 (33): 1143. https://doi.org/10.21105/joss.01143.

Liu, Lei, Ya-Chen Tina Shih, Robert L. Strawderman, Daowen Zhang, Bankole A. Johnson, and Haitao Chai. 2019. "Statistical Analysis of Zero-Inflated Nonnegative Continuous Data: A Review." *Statistical Science* 34 (2): 253–79. https://doi.org/10.1214/18-STS681.

Narasimhan, Balasubramanian, Steven G. Johnson, Thomas Hahn, Annie Bouvier, and Kiên Kiêu. 2024. *Cubature: Adaptive Multivariate Integration over Hypercubes.* https://doi.org/10.32614/CRAN.package.cubature.

Niculescu-Mizil, Alexandru, and Rich Caruana. 2005. "Predicting Good Probabilities with Supervised Learning." In *Proceedings of the 22nd International Conference on Machine Learning - ICML '05*, 625–32. Bonn, Germany: ACM Press. https://doi.org/10.1145/1102351.1102430.

R Core Team. 2021. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Rubin, Donald B. 1984. "Bayesianly Justifiable and Relevant Frequency Calculations for the Applied Statistician." *The Annals of Statistics* 12 (4). https://doi.org/10.1214/aos/1176346785.

Säilynoja, Teemu, Paul-Christian Bürkner, and Aki Vehtari. 2022. "Graphical Test for Discrete Uniformity and Its Applications in Goodness-of-Fit Evaluation and Multiple Sample Comparison." *Statistics and Computing* 32 (2): 32. https://doi.org/10.1007/s11222-022-10090-6.

Scott, David W. 1992. *Multivariate Density Estimation: Theory, Practice, and Visualization.* 1st ed. Wiley Series in Probability and Statistics. Wiley. https://doi.org/10.1002/9780470316849.

Sheather, S. J., and M. C. Jones. 1991. "A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation." *Journal of the Royal Statistical Society: Series B (Methodological)* 53 (3): 683–90. https://doi.org/10.1111/j.2517-6161.1991.tb01857.x.

Silverman, B. W. 2018. *Density Estimation for Statistics and Data Analysis.* Chapman and Hall/CRC Monographs on Statistics and Applied Probability v.26. Boca Raton: Routledge.

Stan Development Team. 2018. "The Stan Core Library." http://mc-stan.org/.

Štrumbelj, Erik, Alexandre Bouchard-Côté, Jukka Corander, Andrew Gelman, Håvard Rue, Lawrence Murray, Henri Pesonen, Martyn Plummer, and Aki Vehtari. 2024. "Past, Present and Future of Software for Bayesian Inference." *Statistical Science* 39 (1): 46–61. https://doi.org/10.1214/23-STS907.

Tukey, John Wilder. 1972. "Some Graphic and Semi-Graphic Displays." In *Statistical Papers in Honor of George W. Snedecor*, edited by T. A. Bancroft and S. A. Brown, 293–316. Ames, Iowa: Iowa State University Press.

Turner, Rolf. 2023. *Iso: Functions to Perform Isotonic Regression.* https://doi.org/10.32614/CRAN.package.Iso.

Van Zwet, Erik W., and Eric A. Cator. 2021. "The Significance Filter, the Winner's Curse and the Need to Shrink." *Statistica Neerlandica* 75 (4): 437–52. https://doi.org/10.1111/stan.12241.

Vehtari, Aki, Daniel Simpson, Andrew Gelman, Yuling Yao, and Jonah Gabry. 2024. "Pareto Smoothed Importance Sampling." *Journal of Machine Learning Research* 25 (72): 1–58. http://jmlr.org/papers/v25/19-556.html.

Wesner, Jeff S., and Justin P. F. Pomeranz. 2021. "Choosing Priors in Bayesian Ecological Models by Simulating from the Prior Predictive Distribution." *Ecosphere* 12 (9): e03739. https://doi.org/10.1002/ecs2.3739.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York. https://doi.org/10.1007/978-3-319-24277-4.

Wilkinson, Leland. 1999. "Dot Plots." *The American Statistician* 53 (3): 276. https://doi.org/10.2307/2686111.