for the clinician to critically consider the algorithm's analysis of the waveform.

The task of analysing physiological data both comprehensively and transparently seems a perfect fit for generalized additive models (GAMs). A recent paper by Wyffels et al. demonstrates how GAMs can be used to isolate the respiratory component of PPV in subjects with atrial fibrillation [4]. An elegant solution that may be used to guide fluid therapy in this patient group.

The aim of this paper is to demonstrate how GAMs can be used to decompose waveforms or time series recorded in mechanically ventilated patients into separate, physiologically relevant, components. This allows analysts to focus on each component individually. We give a short introduction to splines and GAMs, and then demonstrate the method using two examples. First, we use a time series of pulse pressure measurements to give a robust estimate of PPV in mechanically ventilated patients with sinus rhythm (a simplified version of the model presented by Wyffels et al. [4]). Second, we decompose the CVP waveform into separate, physiologically relevant, effects. Finally, we summarise and discuss how GAMs might be used in future research and in clinical monitoring.

1.1 What is a GAM?

Generalized additive models are both flexible and interpretable. In the space of statistical models, they reside somewhere between simple but rigid methods like linear regression and flexible but complex methods like neural networks. With GAMs, we can build transparent models, with components that represent known physiology.

Hastie and Tibshirani introduced GAMs in 1986, as extensions of generalized linear models [5]. Instead of fitting straight lines, GAMs can fit any smooth function. In the basic form of a GAM, a smooth function is fitted for each independent variable in the model. These functions are added together to give the model's prediction of the dependent variable:

$$Y_{predicted} = \alpha + f(X_1) + f(X_2),$$

where α is a constant value and f can be any smooth function (continuous and with no kinks). In this paper, we do not introduce link functions, and we mainly use models with a Gaussian conditional distribution.

1.1.1 Cubic splines

Several types of smooth functions can be used to fit data. In this paper, we use one type: the *cubic spline*. A cubic spline is built by combining a number of third-order polynomials. Each polynomial fits its individual section of the data (e.g., a period of time if time is the independent variable) and is constrained to join smoothly to the adjacent polynomial(s). The intersections between adjacent polynomials are called *knots*. Smoothness at the knots is ensured by constraining adjacent polynomials to align at the knots. Specifically, the values of adjacent cubic polynomials' 0th, 1st and 2nd derivatives must be equal at the knots. The knots can be placed at will, but a common choice is to position knots at the quantiles (including at minimum and maximum) of the independent variable, giving the same number of observations in each segment (see Fig. 1a). Cubic splines are often additionally constrained by fixing the second and third derivative at the

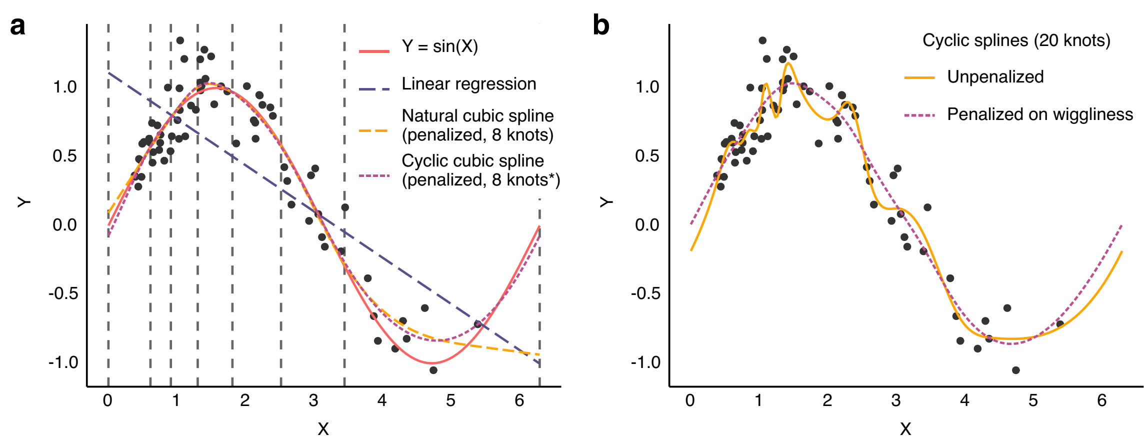

Fig. 1 Splines fitted to simulated data (n = 70). The data-generating function is Y = sin(X) with added normally distributed noise. **a** Vertical dashed lines show the position of the 8 knots. *In the cyclic spline there are effectively 7 knots, since the first and last line represent a single knot, joining the ends of the spline. **b** Comparison of a penalised and an unpenalised spline fitted to the same data. The unpenalised spline with 20 knots is clearly too wiggly and overfits the data. Penalising the spline on wiggliness reduces the risk of overfitting, but keeps the model flexible in case the data demand it