Chapter 1 - Literature Review

Michael Anderson

Robust Locally Weighted Regression and Smoothing Scatterplots (Cleveland 1979)

The author suggests a technique for fitting a smooth curve to a scatterplot using locally weighted regression, with a fitting procedure that makes the estimate more robust to outliers. The fitting procedure goes as follows: an initial polynomial is fit for each observed data point, weighted by a chosen weight function, from which the predicted value for each observed data point is obtained. Then, new weights are calculated based on the residuals from this initial fit, thus limiting the effect of distant, outlying observations. These new weights are used to refit local polynomials, and the process is repeated iteratively. The author demonstrates the method in a few examples, including lead intoxication data, and as a diagnostic tool when viewing residual plots in regression. The author discusses selection of the various tuning parameters for the method, including the degree of the local polynomial, the initial weight function, the number of iterations used in the fitting procedure, and the smoothing parameter. The recommendation is to use linear local polynomials, a tricube weight function, and 2 iterations, for practical and computational concerns. The author gives a reasonable range of smoothing parameter values as 0.2 to 0.8, recommending 0.5 as a starting point if no specific value is preferred. The author then discusses computational considerations, and estimation techniques for fitted value standard errors.

Self-Consistent Estimation of Mean Response Functions and Their Derivatives (Charnigo and Srinivasan 2011)

In this paper the authors describe a method for nonparametrically estimating a mean response function, and its derivatives, in such a way that this estimated function is self-consistent. Self-consistency is defined by the authors as a property of estimators in which the derivatives of the estimated mean response function are equal to the estimated derivatives of the mean response function, for an arbitrary number of derivatives. To clarify, this can be written notationally as $\frac{d^j}{dx^j}\hat{\mu}(x) = \widehat{\frac{d^j}{dx^j}\mu(x)}$, for any jth derivative of interest. The authors demonstrate a few practical examples where the self-consistency property is desirable; in particular, when one wishes to model physical systems and phenomena which can often be described using differential equations, or when one is interested in finding optima of a mean response function. In the latter case, estimators which lack the property of self-consistency can often give conflicting results based on the optimization technique used. As an example of this issue, the authors discuss the case of estimating the age at which girls experience the highest rate of growth, using data from the Berkeley Growth Study (Ramsay and Silverman 2002). If a self-consistent modeling technique is not used, one can obtain different estimates of this age, based on whether a first derivative or second derivative optimization technique is used. If a self-consistent estimator is used, however, you are guaranteed to get the same estimate from both optimization techniques, by the definition of self-consistency.

The authors then discuss the details of the compound estimator they have developed. The first step is to create a pointwise Taylor polynomial estimate of the mean response function, based on its estimated derivatives. Then these local polynomials are smoothed to ensure differentiability and statistical consistency. The authors discuss the use of different pointwise estimation techniques, including different weights and polynomial order for local regression, as well as the use of either smoothing splines or kernel smoothing. The authors do not evaluate the benefits of these various techniques, however, and use a relatively simple local regression estimator with rectangular weights for the applications and simulations.

The authors discuss some of the assumptions and conditions under which the pointwise estimators converge optimally, which affects the convergence rate of the compound estimator - see Stone (1980, 1982) for more on convergence properties of nonparametric estimators. The compound estimator is shown to have convergence rates that are very close to the convergence rate of the underlying pointwise estimator used. This means that utilizing the compound estimation technique instead of just a local regression technique does not meaningfully increase the rate of convergence. The authors then discuss techniques to filter out noise in the data to better estimate the mean response function and methods to improve estimation near the boundaries of the input space. Namely, the authors recommend filtering the data by fitting two low order polynomial (0 and 1) compound estimates of the mean response function first, and then taking a weighted average of the two fits to get a denoised version of the data on which they will train a higher order compound estimator. This technique will improve performance near the boundaries, but to get better results the authors also discuss generating extrapolated data outside of the boundaries of the input space using low order compound estimation, which will smooth out the estimation near the boundaries of the input space. This is similar to how natural smoothing splines extrapolate a straight line beyond the boundaries of the data to improve boundary performance - see Wasserman (2006) for more on smoothing splines.

Next, the authors go into the topic of tuning parameter selection for the compound estimator. Outside of the tuning parameters of the underlying pointwise estimators, there are two tuning parameters, one that controls the weight function used to smooth the local polynomials and one that controls the amount of grid points at which the local polynomials are estimated. The GC_p criterion was used to evaluate tuning parameter selection - see Charnigo et al. (2011) for more on GC_p - and the authors give some recommendations for tuning parameter selection in different circumstances. The authors ran some simulations to compare their compound estimation technique to other nonparametric estimators (local regression and spline smoothing). Their simulations showed that over all of the scenarios presented, their technique was better able to estimate higher derivatives of the response function accurately, compared to the other techniques, and the estimated response functions were also closer overall to their true form. The authors finished by applying the technique to human growth, using data from the Berkeley Growth Study (Ramsay and Silverman 2002). They were able to closely replicate the results found by Ramsay and Silverman - who used smoothing splines - while ensuring the desired self-consistency property that was laid out at the beginning of the paper.

Nonparametric and Semiparametric Compound Estimation in Multiple Covariates (Charnigo et al. 2015)

This article is an extension of Charnigo and Srinivasan (2011), in that it alters the self-consistent compound estimator the authors previously developed to allow it to estimate a mean response function with a multivariate input space of arbitrary dimension. The compound estimator is further extended in this article to fit a semiparametric model with random effects, to allow for specified parametric terms in the model, as well as repeated measures on subjects. The authors highlight an extensive list of possible applications for this new method, including the primary motivating example of estimating the progression of Parkinson's disease symptoms using patient voice recordings (Tsanas et al. 2010). This example includes 2 covariates whose effect will be measured nonparametrically and 7 covariates whose effect will be measured parametrically. Using this example the authors further highlight the inconsistency of inference from estimation techniques in which estimation and differentiation are not interchangeable. They show that one's estimate of a local minimum of the two-dimensional mean response function depends upon whether one uses the first derivatives of the estimated mean response function or the estimates of the first derivatives of the mean response function (i.e. $\frac{\partial}{\partial x_j}\hat{\mu}(x_1, x_2)$ or $\frac{\partial}{\partial x_j}\widehat{\mu(x_1, x_2)}$). Not only do the authors state that their method will resolve these sorts of issues, but they also argue that under certain conditions their method can actually outperform traditional local regression methods in terms of squared error.

The authors then provide the technical details for the implementation of their compound estimator for multiple variable nonparametric regression. The broad idea is similar to that of the previous, univariate method (Charnigo and Srinivasan 2011) but with the mathematical details being convoluted slightly by the multidimensionality of their input space. Roughly speaking, their method starts by estimating local Taylor polynomials of the mean response function at a series of centering points throughout the covariate space. These local polynomials can be estimated using regular local regression techniques, or other methods if desired. Once the series of local polynomials is obtained, each polynomial is given a weight function (called the convolution weight by the authors) based on the location of its centering point within the covariate space, so that the farther away you get from a given centering point, the less contribution its corresponding local polynomial will have on the resulting compound estimator - similar to kernel functions and other weight functions used in local smoothing operations. For more on the technical details of smoothing techniques, see Härdle (1990). This weight function depends on a tuning parameter that the authors refer to as the convolution matrix. The authors recommend using a scalar multiple of the identity matrix for the convolution matrix in most practical situations, though it could mathematically be any symmetric positive definite matrix. The final compound estimator of the mean response function is the sum of all of the local polynomial estimates, weighted by their corresponding convolution weight.

The authors then discuss the theoretical properties of their estimator, including a derivation of its convergence rate under some loose assumptions. They discuss how their estimator compares to the optimal convergence rates derived by Stone (1980, 1982), and the fact that their estimator only differs from the optimal rates by a logarithmic factor of n. This is even an improvement over the theoretical results discussed in Charnigo and Srinivasan (2011), in which they showed their estimator differed from optimality by small powers of n, rather than the logarithmic factors derived here. They also highlight that the convergence rates derived for their compound estimator do not assume the same degree of smoothing or convolution along all the axes of the input space, which allows for some flexibility in modeling, namely in the form of tuning parameter selection for both the local polynomials and the convolution matrix.

The next part of the article is an extension of their multivariate nonparametric compound estimator to allow for both fixed and random parametric effects in the model as well. An estimation technique is derived for fitting this model, and the consistency and convergence properties are derived. However, the assumptions of this fitting technique may be too strict for most observational applications, so the authors also discuss a different technique that uses a backfitting algorithm similar to that first introduced for use in generalized additive models (Breiman and Friedman 1985). This method starts with an initial fit of the nonparametric portion of the model, then regresses the residuals from this fit on the parametric portion of the model, from which residuals are again calculated, upon which the nonparametric portion is refit. This iteration continues, alternating between the parametric and nonparametric components of the model, until convergence is satisfied. The authors provide a convergence criterion that can be utilized to ascertain convergence according to a user-specified cutoff.

The authors then discuss a method of improving estimation near the boundaries of the covariate space, similar to that used in the one-dimensional case (Charnigo and Srinivasan 2011). Essentially, artificial data is created to extrapolate beyond the true input space, to ensure the smoothness of the compound estimate of the mean response function beyond those boundaries. As shown previously, this can significantly improve prediction and estimation near the boundary. Then, the authors discuss selection of tuning parameters for both the local polynomials and the compound estimator. They argue that the compound estimates tend to be more sensitive to the local polynomial estimator's tuning parameter selection than to that of the higher level compound estimator. For this reason, the authors recommend spending more computational time and effort on selecting the bandwidth and other tuning parameters for the local polynomial estimators than for the higher level tuning parameters. Toward this aim, the authors give a few recommendations for specific ranges of values and techniques for narrowing down the selection of the higher level parameters.

Next, the authors demonstrate their method using the Parkinson's disease dataset mentioned earlier. The outcome of interest is an estimate of disease symptom progression based on the Unified Parkinson's Disease Rating Scale (UPDRS). They fit a model with 2 covariates whose effects will be estimated nonparametrically and 7 covariates whose effects will be estimated parametrically, with random effects for the repeated subject measures. They fit this model with a traditional local regression fitting technique, as well as their compound estimation technique that utilizes a backfitting algorithm. They also used the boundary adjustment technique they previously discussed to get more stable estimates near the boundaries of the covariate space. Their compound estimator produced much smoother estimates of the mean response function and its derivatives, compared to the local regression technique. Also, the standard errors for the parametric coefficients are almost identical to those produced by the standard technique. The authors also demonstrate that their compound estimates do maintain the self-consistency property as desired, by which estimation and differentiation are interchangeable, while the local regression estimates do not.

Finally the authors briefly highlight some results from a simulation study comparing local regression to their compound estimator under two different simulated scenarios: the first with an additive mean response function, and the second with a multiplicative mean response function. The authors used the two techniques on 100 datasets with a fixed sample size of 225 each, and utilized the sums of squared errors for estimating the mean response function and its first and second derivatives to compare the two techniques. The authors found that their technique "reduced the average sum of squared errors by 26% (Scenario 1) or 27% (Scenario 2) versus local regression when estimating the mean response, by between 63% and 65% when estimating the first-order derivatives, by between 75% and 80% when estimating the homogeneous second-order derivatives, and by 27% or 25% when estimating the mixed second-order derivative." So, we can see that their compound estimator better estimates the mean response function than local regression. But, the most significant improvements came in the estimation of the derivatives of the mean response function, with higher order derivatives having even higher reductions in sums of squares than the lower order derivatives. This simulation study really demonstrates the advantages of using this technique when accurate estimation of derivatives of a mean response function is necessary, such as when estimation of local optima is desired.

A Generalized C_p Criterion for Derivative Estimation (Charnigo et al. 2011)

The GC_p criterion was utilized by Charnigo and Srinivasan (2011) with their compound estimator. This criterion was specifically developed for parameter selection in nonparametric estimation of derivatives, and can be used with any nonparametric estimation technique that estimates derivatives via a linear combination of the responses. The GC_p criterion is based on the C_p criterion (Mallows 1973) that was originally developed for variable selection in parametric regression settings. Mallow's C_p is defined as the residual sum of squares $\sum_i^n [Y_i - \hat{\mu}(x_i)]^2$, with an added penalty term that makes it equal in expectation to the desired, but unobservable, error sum of squares $\sum_i^n [\mu(x_i) - \hat{\mu}(x_i)]^2$ (note that the residual sum of squares depends on the observed responses, while the error sum of squares depends on the underlying mean response function). Many different criteria have been developed for tuning parameter selection in nonparametric regression settings, and the authors give an overview of some of these methods. But, many of them are not mathematically appropriate for derivative estimation tasks, and they often depend upon a specific fitting method. The proposed GC_p criterion is built for nonparametric derivative estimation, and is general enough to be used with many popular nonparametric regression techniques.

The authors go into the details of the development of GC_p as an analog to C_p, by similarly defining a residual sum of squares type function that depends on the estimated derivatives of the function, instead of the mean response: $\sum_i^n [Y_i^{(q)} - \widehat{\frac{d^q}{dx^q}\mu_\lambda(x_i)}]^2$, where $Y^{(q)}$ is a noise-contaminated version of the q-th derivative of the mean response function, and λ is the tuning parameter of interest. The authors note that the $Y_i^{(q)}$ can be defined many ways, and later will develop empirical derivatives that can be used for this. But, for now they just assume they are simple linear combinations of the responses. Just like in the original C_p, this residual sum of squares does not necessarily have the same expected value as its error sum of squares counterpart: $\sum_i^n [\frac{d^q}{dx^q}\mu(x_i) - \widehat{\frac{d^q}{dx^q}\mu_\lambda(x_i)}]^2$. The authors move to a more general framework, where the target function is a weighted error sum of squares, and then define:

$$GC_p(\mathbf{Y}, \hat{\mu}_\lambda) := \sum_i^n s_i \left(Y_i^{(q)} - \widehat{\frac{d^q}{dx^q}\mu_\lambda(x_i)} \right)^2 + \sigma^2 \sum_i^n s_i \sum_m^n \left(2c_{i,m} l_{m;\lambda}^{(q)}(x_i) - c_{i,m}^2 \right)$$

where the s_i are the observation weights, the $c_{i,m}$ are the elements of the linear transformation matrix used to create the $Y_i^{(q)}$ values, and the $l_{m;\lambda}^{(q)}(x_i)$ are the elements of the smoothing matrix used to create $\widehat{\frac{d^q}{dx^q}\mu_\lambda(x)}$. The authors note that this penalty term technically doesn't make GC_p exactly equal in expectation to its target weighted error sum of squares, but it is only off by a term that they will show to be asymptotically negligible. Another issue with this definition is that it depends on the often unknown response variance σ^2. So, the authors recommend using an empirical variance estimate $\hat{\sigma}^2 := \frac{\sum_i^n [Y_i - \hat{\mu}_\lambda(x_i)]^2}{n - \sum_m^n l_{m;\lambda}(x_m)}$, where the sum in the denominator behaves similarly to degrees of freedom. Note that this estimated variance depends on λ. To get around this fact, the authors give the recommendation to calculate $\hat{\sigma}^2$ based on the λ value that produces the least smooth fit. This recommendation comes from Loader (1999), who says this will help reduce the bias in the estimate of $\hat{\sigma}^2$.

The authors then go on to define a method of calculating the $Y_i^{(q)}$ that have lower variance and better convergence properties than traditional first order difference quotients, which they call empirical derivatives. The definition of the first-order empirical derivative $Y_i^{(1)}$ is essentially a weighted sum of symmetric difference quotients centered at i. The authors discuss the convergence properties of this noise-contaminated measure of the first derivative, and show how to calculate optimal quotient weights to use in the sum, so that the variance of the resulting empirical derivative is minimized. The higher order empirical derivatives $Y_i^{(q)}$ (for $q \geq 2$) are defined inductively, using a weighted sum of the difference quotients defined by $Y_i^{(q-1)}$. The authors argue that there aren't simple formulas for the weights that minimize the variances for higher order empirical derivatives, but there are recommendations for calculating weights that can at least provide reasonable control over the variances.

Next, a simulation study was performed, which compared GC_p for tuning parameter selection to several other criteria, including cross-validation, generalized cross-validation, and AIC. The estimation techniques used were kernel smoothing, local regression, spline smoothing, P-spline smoothing (for more on P-spline smoothing methods, see Eilers et al. (2015)), and the authors' own compound estimation technique. The simulated datasets all had the same mean response function, with varying levels of noise, controlled by the variance of the random error term. For all models, the mean response function and its first two derivatives were estimated. The methods were compared based on their inflation of the weighted residual sum of squares with respect to the estimated derivative of interest, in comparison to either the lowest weighted residual sum of squares across all tuning parameter values, or in comparison to the average weighted residual sum of squares across all possible tuning parameter values. Overall, the GC_p criterion performed well against its competitors. It very often produced the best results, particularly for kernel smoothing, local regression, and spline smoothing. The authors note that the most viable competitors of GC_p are probably AIC and generalized cross-validation. Though, there were scenarios where these two performed very poorly in relation to GC_p, which always performed fairly well, even when other methods produced better results in that scenario.

After displaying some of the theoretical properties of GC_p, including its asymptotic efficiency and its use in constructing approximate prediction intervals for the unknown weighted error sum of squares, the authors demonstrate a practical application of the GC_p criterion to inference of the chemical composition of a substance using Raman spectroscopy. They objective is to accurately classify a chemical sample by comparing it to two other samples of known composition, by comparing the first derivatives of the underlying intensity functions that produced the Raman spectrum for each sample. The authors used their compound estimation technique to estimate the first derivatives, utilizing both GC_p and AIC to select the tuning parameters, and the GC_p tuning parameters produced a much smoother estimate of the first derivative than the AIC tuning parameters. The authors point out that this makes sense since AIC was developed for mean response estimation, when derivative estimation probably requires more smoothing than mean response estimation normally would. More importantly, the GC_p model's first derivative estimate was able to correctly classify the "unknown" third sample to its appropriate chemical compound, while the AIC model's estimate was not. This demonstrates the impact that proper tuning parameter selection can have on inference, and specifically why GC_p is a potentially useful criterion for tuning parameter selection when derivative estimation is necessary.

A Multivariate Generalized C_p and Surface Estimation (Charnigo and Srinivasan 2015)

This article proposes a new, multivariate analog to the authors' previously developed GC_p criterion for tuning parameter selection in nonparametric estimation (Charnigo et al. 2011). The introduction of the paper begins by describing the motivation behind the development of this new technique, both in terms of a case study that deals with measuring liver function (more on that later), as well as in terms of the statistical shortcomings of the current methods used in this area. Most current techniques minimize a data-based proxy of the error sum of squares of the mean response function, while the proposed method minimizes that of the derivatives of the mean response function, which allows for more smooth estimation of a function's derivatives. Also, many techniques only work well for a specific regression technique (e.g. kernel or spline smoothing), while the proposed technique is generally applicable to many different estimation schemes. And lastly, most current techniques are derived for regression of the outcome on only one explanatory variable, while the proposed criteria is explicitly defined in terms of D-dimensional covariate spaces, where D is an arbitrary, positive integer.

The outline of the specification of MGC_p is very similar to that of GC_p, only in terms of multiple covariates instead of a single one. The goal is to derive a quantity that can be empirically measured, which is approximately equal in expectation to the error sum of squares for a given derivative of the mean response function. Just like in the original C_p statistic (Mallows 1973), the method of obtaining this desired behavior is to calculate a data based analog to the error sum of squares (in the case of MGC_p, the error sum of squares for the derivative) and then add a penalty term that makes the resulting sum approximate the target error sum of squares in expectation. The authors then discuss some practical matters with regards to the use of MGC_p, including selection of constants used in the calculation of the empirical partial derivatives needed to calculate MGC_p, as well as a way of estimating the unknown outcome variance, using Loader's (1999) advice to do so using tuning parameter values that produce under-smoothed estimates of the mean response function.

Next, the authors demonstrate their criterion using the liver function case study mentioned previously. The goal is to see how well two measures of liver function (ALT and AST) can predict a more invasive measure (bilirubin). ALT and AST are easier and cheaper to test for compared to the current standard, bilirubin, and together could provide a good early screening test for physician follow up based on liver function (Pollock et al. 2012). The authors used their compound estimation technique (Charnigo et al. 2015) to estimate the mean response function $E(Y|X_1, X_2) = f(X_1, X_2)$ where Y is the log-transformed bilirubin, and X_1 and X_2 are proportional to, respectively, the sum and difference of the log-transformed ALT and AST.

They also utilized their boundary adjustment technique mentioned in Charnigo et al. (2015) to get better performance near the boundaries of the two-dimensional covariate space. Using this estimation scheme, they used six different criteria to select the two tuning parameters used in the regression, one for the local polynomial nearest neighbor fraction, and one for the convolution matrix applied to the local polynomials by the compound estimation technique.

The selection criteria used were: MGC_p, C_p, $iAIC$ (Hurvich et al. 1998), GCV (Craven and Wahba 1979), and then altered versions of both $iAIC$ and GCV, using the empirical partial derivatives and smoothing matrix used to calculate MGC_p. The six different criteria produced 4 different combinations of the two tuning parameters: one chosen by C_p and GCV, one chosen by $iAIC$, one chosen by the altered $iAIC$, and one chosen by the altered GCV and MGC_p. Overall, it appears that the results produced by the first tuning parameter combination are definitely under-smoothed, in terms of the resulting mean response function as well as its derivatives. The other three tuning parameter selections performed similarly enough to one another that the choice of the "best" criterion is not clear in this specific example. So, while MGC_p can't be said to be the best of the criteria in this case, it does at least seem to perform just as well as any other criterion.

The authors then demonstrated the efficacy of their criterion in a simulation study, using the same six criteria as in the case study, plus an additional criterion for smoothing splines based on the REML criterion (Ruppert et al. 2003), which the present authors call the variance components criterion. Overall the MGC_p performs very well. It is often one of the most efficient methods for tuning parameter selection, in terms of its resulting sum of squared errors. The biggest competitor to the MGC_p criterion is the $iAIC$ criterion, though it doesn't perform quite as well in as many of the simulated scenarios. The authors note that in general, the estimates obtained from MGC_p tend to be smoother than other techniques, which makes sense given the method's specification in terms of derivative estimation, where smoother estimates are often required when compared to just estimating mean responses. The authors also point out that in terms of computation, the added burden needed to calculate MGC_p is often almost negligible compared to the computation required to actually perform the nonparametric smoothing itself.

Lastly, the authors derive some theoretical properties of MGC_p, the most important of which is that it is asymptotically equal in expectation to its intended target, the error sum of squares for estimation of a derivative, under some assumptions, which the authors discuss in more detail. The authors conclude the paper by discussing the curse of dimensionality; in particular, how nonparametric estimation could be difficult in high dimensional cases, regardless of the criterion used for tuning parameter selection. For more on the curse of dimensionality in general, see Chapter 2 of Hastie et al. (2009). In those cases, the authors recommend either using a semi-parametric model, where some covariates are treated parametrically, thus reducing the dimensionality of the portion to be estimated nonparametrically, or by introducing some simplifying assumptions on the form of the mean response function, such as assuming an additive model with no interaction between the nonparametrically estimated covariate effects.

Jump Regression, Image Processing, and Quality Control (Qiu 2018)

This article gives an overview of some of the latest research in the methods of jump regression analysis, with particular focus on the areas of image processing and statistical process control. This review will focus on the article's discussion of jump regression methods and their application to image processing, choosing to leave out the extended discussions of statistical process control applications for the sake of brevity. In general, jump regression analysis (JRA) is a nonparametric technique that allows one to model a regression function which is not continuous in either is mean response function or its derivatives. Many traditional nonparametric regression methods, such as kernel and spline smoothing, are only appropriate for estimating continuous functions. JRA is specifically designed to handle estimation of functions where there are discontinuities, or jumps, in the function or its derivatives. Not appropriately allowing for these discontinuities in your estimation can lead to very different estimates, as the author displays in the case of a weather station's measurement of sea-level pressures, from 1921 to 1992. In a previous paper (Qiu and Yandell 1998), the author showed that there was a definite jump in the pressure measurements around the year 1960. Had that jump not been accounted for, and instead a continuous regression function had been assumed, the resulting estimated regression function would have been quite poor in comparison.

Starting with the one-dimensional case, the author goes into the details of the specification of the JRA model. Namely, we first assume that the regression function has the generic nonparametric form of $y_i = f(x_i) + \epsilon_i$, where f is an unknown regression function and ϵ_i are random errors. However, JRA then places a further stipulation on the functional form of f by assuming that $f(x) = g(x) + \sum_{j=1}^{p} d_j I(x > s_j)$, where $g(x)$ is a continuous function of x in its domain, p is the number of jump points, s_j are the positions of the jump points, and d_j are the sizes of the jumps. This formulation of the mean response function f gives us the two crucial steps of JRA: the first is jump detection, where we estimate the number, location, and size of the jumps in our regression function, and the second is jump-preserving curve estimation, where we then estimate the continuous portions of the regression function, while maintaining the estimated jumps.

There are many different methods for jump detection (the author provides nine different references for some at one point in the article) but most of them rely upon a criteria that is the difference of two one-sided local weighted averages. If the number of jump points p is known, one can simply estimate their positions by maximizing this criteria for p different points in the input space. More recently, the problem of how to identify the number of jumps in a dataset has seen some development, including the introduction of a jump information criterion for their estimation (Xia and Qiu 2015). To estimate the regression function that preserves the estimated jumps, the simplest method is just to transform your data by subtracting off the estimated jumps, $y_i^* = y_i - \sum_{j=1}^{\hat{p}} \hat{d}_j I(x_i > \hat{s}_j)$, and then estimate a continuous regression function on this transformed data, using whatever method you wish. The author provides references for another method that estimates $f(x)$ directly, without explicit jump detection being performed first. The author also notes here that JRA can be formulated to handle jumps in the derivatives of the regression function as well. Then the case of two-dimensional JRA is discussed, with the major difference being that jump positions in two dimensions are described by curves, called jump location curves. While the estimation of jump location curves is mathematically more complicated than their one-dimensional counterparts, the overall process of JRA in two dimensions is similar to that of the one-dimensional case.

After a short section discussing the similarities and dissimilarities of JRA and traditional statistical process control tasks, the author begins to discuss the two-dimensional JRA model for gray-scale image processing: $Z_{ij} = f(x_i, y_j) + \epsilon_{ij}$, where (x_i, y_j) is a pixel in the image, $f(x_i, y_j)$ is the true image intensity level, ϵ_{ij} is the pointwise noise, and Z_{ij} is the observed image intensity level. Jump regression is an extremely important task in image processing, since objects in an image can often be identified by their outlines, which can be located via the detection of jumps in the intensity function. The author briefly compares JRA to two other popular tools in describing images: Markov random fields and diffusion equations. The author argues that JRA is the most flexible of the three, due to the other two methods being sensitive to assumption violations and excess noise, respectively.

Next, the author discusses two of the fundamental problems in image processing: edge detection and edge-preserving image denoising. These two tasks are almost synonymous to the two steps of JRA discussed earlier. For this reason, JRA methods are an obvious choice for both of these tasks. Many edge detectors are built on jumps in the first or second derivatives of the image intensity surface. The author briefly compares the use of the two different derivatives, and the pros and cons of each. A task that is different from, but closely related to, edge detection is image segmentation. This involves identifying the edges of specific objects in an image, rather than the outline of the entire image. The author discusses the subtleties of the two tasks, and their heavy dependence on the specific application of interest. Then the author discusses some non-JRA methods for image denoising, including the Markov random field and diffusion equation methods touched upon earlier, as well as wavelet transformations and adaptive weights smoothing.

In the cases where we don't suspect the pixel noise values are spatially independent, we instead have to focus on image deblurring, where spatial blur can be caused by many reasons, including a relative movement between the camera and the subject of the image. In these cases, the observed image is often modeled as a convolution between a function which describes the spatial contamination (blur) of an image, and the true image intensity function. Difficulties in this task often stem from the fact the problem is "ill-posed", as the author puts it; meaning that different combinations of intensity and contamination functions could produce the same observed image, even without considering the pointwise noise in the image as well. The author discusses some different techniques for this task as well, including some which use test images from the same camera to first estimate the contamination function, then apply the inverse convolution of that estimated contamination function to other images to deblur the image.

The author finishes his section discussing various image processing techniques by talking about image registration, which is basically the geometric alignment of objects in different images to provide better comparison. As an example, he discusses the task of looking at the pixelwise difference in two gray-scale satellite images of the San Francisco bay area in 1990 and 1999. Even though the two images look to be well aligned by the human eye, if the pixelwise difference is taken without image registration, there is a clear trend in the differences caused by minute differences in the location of the edges in the two images. By first registering the images, a better, more subtle comparison can be made about the true differences in the two images. He then briefly discusses the issues of 3-D imaging, and how most 2-D methods can't be generalized well to a 3-D setting. To end the article the author goes into detail about the application of these image processing methods to the specific area of statistical process control which, as mentioned earlier, is left out of this review.

Jump Information Criterion for Statistical Inference in Estimating Discontinuous Curves (Xia and Qiu 2015)

As was discussed in Qiu (2018), most methods for estimation of jump regression curves depend on prior specification of either the number of jumps to be estimated, or related quantities, such as the minimum jump size. The present article seeks to get around these assumptions by instead developing an information criterion for nonparametric estimation of a regression curve with an unknown jump structure. This criterion includes a penalty term for the complexity of the model's estimated jump structure, rather than the complexity of the entire model, thus allowing one to optimize the criterion to obtain a reliable estimate of the regression function's true jump structure without overcomplicating it. The assumed model for this method is similar to the general Jump Regression Analysis (JRA) model discussed previously. Assume that the Y_i are generated from the model $Y_i = f(x_i) + \epsilon_i$, where the ϵ_i are iid with zero mean and constant variance. Further assume that $f(x) = f_C(x) + f_J(x) = f_C(x) + \sum_{j=1}^{m_0} d_j I(x > s_j)$, where the number m_0, positions s_j, and sizes d_j of the jumps are all unknown. This assumption states that the mean response function $f(x)$ can be broken down into two parts: $f_C(x)$, which captures the continuous portion of the mean response functions, and $f_J(x)$, which captures the jump structure of the mean response function. For simplicity of theoretical derivation, the design points x_i are assumed to be equally spaced in the interval $[0, 1]$, and the ϵ_i are assumed to have equal variance. Their method does generalize to cases where these assumptions are not necessarily true, however.

The next section discusses the estimation of the jump regression model given above, but first under the assumption that there is some known upper bound m on the number of jump points. The continuous portion of the regression function is estimated using a local linear kernel smoother (Loader 1999). Under this framework, the jump locations and sizes are estimated using a similar approach to that described in our previous discussion of Qiu (2018), by looking at a two-sided local weighted mean to find discontinuities. After the locations and sizes are estimated, the data is transformed by removing the estimated jump portion of the regression function $f_J(x)$, and then the remaining continuous portion $f_C(x)$ is estimated using whatever method is desired. The authors point out that the degrees of freedom of the final regression model, measured by the trace of the resulting smoother matrix (Efron 2004), only depend on the jump points through the number of points to be estimated, which is constant in n. So, asymptotically, the effect of the jump structure on the complexity of the resulting full model is negligible, compared to the complexity of the continuous portion of the model, which tends to infinity. Finally, under the known number of jumps assumption, the authors show that this method can consistently estimate all m jumps. But, if there are more than m jumps in the true regression function, then the other jumps that weren't estimated would give an asymptotic bias to the estimates near the unmeasured jump points, that is a function of the true jump size. On the other hand, if the true number of jumps is actually less than the assumed number, then the estimation will artificially create some jumps, whose size will uniformly converge to 0. So, the takeaway for this method is to probably err on the side of overestimating the number of jumps, since the bias inflicted by overestimation tends to be less impactful than the bias of underestimation.

However, the authors argue that this issue is more easily avoided if the number of jump points is first estimated from the data, rather than assumed prior to estimation. Toward that goal, the authors have developed a jump information criterion, denoted $JIC(m)$, which takes into account the effects of over- and under-estimation of the number of jump points. Like many other information criteria, the authors' has two parts: one that depends on the sum of squared residuals $SSR(m) = \sum_i^n [Y_i - \hat{f}_m(x_i)]^2$, and another that is a penalty for the complexity of the model. The actual criterion is defined as $JIC(m) := nlog[SSR(m)/n] + P(n) \sum_j^m \frac{1}{|\hat{d}_j(m)|^\gamma}$, where γ is a tuning parameter, $\sum_j^m \frac{1}{|\hat{d}_j(m)|^\gamma}$ is a penalty based on both the number and size of the jumps in a model's jump structure. It is an increasing function in m, and it grows faster when $m > m_0$. The adjustment factor $P(n)$ is there to ensure that, asymptotically, the JIC is decreasing when $m \leq m_o$ and increasing when $m > m_0$. The authors provide a range of values for $P(n)$ that result in the desired behavior. These values are based on n, γ, and the bandwidth h_n of the local smoother. In practice, the authors recommend first estimating the optimal number of jump points m with $\hat{m} = \arg\min_{m \geq 0} JIC(m)$, and then continuing to estimate the jump locations and sizes as normal, based on the assumed $\hat{m}$. The authors provide a formula for $P(n)$ that will provide the optimal convergence rate of $\hat{m}$ to m_0. Finally, the authors also derive a Bayesian information criterion for estimating the number of jumps: $BIC(m) = n \log[SSR(m)/n] + m \log(nh_n)$. Note that this is merely a special case of their JIC, where $\gamma = 0$, $P(n) = \log(nh_n)$ (which is outside of the authors' given range for the desired behavior), and the complexity penalty depends on the jump structure only through the number of jumps, instead of both the number and size of jumps. Returning to their JIC, the authors then show that the estimates $\hat{s}_{\hat{m}}$, $\hat{d}_{\hat{m}}$, and $\hat{f}_{\hat{m}}(x)$ all converge to their true values as if the number of jumps were known *a priori.*

Next, the authors demonstrate the properties and practical use of their JIC method in numerical studies using both simulated and real data. Their simulated data come from a model with two jump points, where the two outer sections of the regression function are simple linear functions, and the interior section of the regression function is a sin curve that is very steep near the jump points, which according to the authors makes the jumps particularly difficult to detect. They consider three different cases: Case 1, where the model assumptions laid out earlier are all met, Case 2, where the design points are random instead of evenly spaced, and Case 3, where the error terms are not independent. They use 5 different methods to estimate the number of jump points: JIC with a small penalty, JIC with a moderate penalty (the one said to provide optimal convergence rates), JIC with a large penalty, the derived BIC criterion, and an additional wavelet method of Wang (1995). They used sample sizes of 200, 500, and 1000, for 1000 simulations each. Looking just at $\hat{m}$ across all the different scenarios, their JIC with the recommended moderate penalty performed better than all the other methods, in every single setting. It is also the only one that accurately detected 2 jump points in 100% of the simulations for both the 500 and 1000 sample size scenarios, for all three cases presented (for the sample size 200 simulations, it was correct 96.6%, 95.6%, and 93.4% of the time for the three cases, respectively). The authors also compared the methods based on the Hausdorff distance between the estimated jump locations and the true jump locations, and similar results were found, lending more evidence to the utility of their criterion. For the real data study, the authors returned to the Bombay weather station sea-level pressure data, which has measurements from 1921-1992, and was discussed in Qiu (2018) and originally in Qiu and Yandell (1998). In the original article, the authors were able to discover a jump point at around 1960, using a method that relies on a threshold determined by a pre-selected significance level. The authors of the current article were able to replicate that finding using their JIC method. However, they were also able to detect an additional jump point in 1938. The authors argue that their method is more reliable due to its relative objectivity, since the original article used a method that depended on a subjectively pre-selected significance level, whereas their method does not rely on such *a priori* specifications.

References

Breiman, L., and Friedman, J. H. (1985), "Estimating Optimal Transformations for Multiple Regression and Correlation," *Journal of the American Statistical Association*, 80, 580–598. https://doi.org/10.1080/01621459.1985.10478157.

Charnigo, R., and Srinivasan, C. (2011), "Self-consistent estimation of mean response functions and their derivatives," *Canadian Journal of Statistics*, 39, 280–299. https://doi.org/10.1002/cjs.10104.

Charnigo, R., and Srinivasan, C. (2015), "A multivariate generalized Cp and surface estimation," *Biostatistics*, 16, 311–325. https://doi.org/10.1093/biostatistics/kxu042.

Charnigo, R., Feng, L., and Srinivasan, C. (2015), "Nonparametric and semiparametric compound estimation in multiple covariates," *Journal of Multivariate Analysis*, 141, 179–196. https://doi.org/10.1016/j.jmva.2015.07.005.

Charnigo, R., Hall, B., and Srinivasan, C. (2011), "A Generalized C_p Criterion for Derivative Estimation," *Technometrics*, 53, 238–253. https://doi.org/10.1198/TECH.2011.09147.

Cleveland, W. S. (1979), "Robust Locally Weighted Regression and Smoothing Scatterplots," *Journal of the American Statistical Association*, 74, 829–836. https://doi.org/10.2307/2286407.

Craven, P., and Wahba, G. (1979), "Smoothing noisy data with spline functions," *Numerische Mathematik*, 31, 27. https://doi.org/https://doi-org.ezproxy.uky.edu/10.1007/BF01404567.

Efron, B. (2004), "Rejoinder," *Journal of the American Statistical Association*, 99, 640–642. https://doi.org/10.1198/016214504000000917.

Eilers, P., Marx, B., and Durbán, M. (2015), "Twenty years of P-splines," *SORT (Statistics and Operations Research Transactions)*, 39, 149–186.

Hastie, T., Tibshirani, R., and Friedman, J. (2009), *The Elements of Statistical Learning*, Springer Series in Statistics, New York: Springer.

Härdle, W. (1990), *Applied Nonparametric Regression*, Cambridge University Press.

Hurvich, C. M., Simonoff, J. S., and Tsai, C.-L. (1998), "Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion," *Journal of the Royal Statistical Society:*

Series B (Statistical Methodology), 60, 271–293. https://doi.org/10.1111/1467-9868.00125.

Loader, C. (1999), *Local regression and likelihood*, New York: Springer-Verlag.

Mallows, C. (1973), "Some Comments on C_p," *Technometrics*, 15, 661–675.

Pollock, N. R., Rolland, J. P., Kumar, S., Beattie, P. D., Jain, S., Noubary, F., Wong, V. L., Pohlmann, R. A., Ryan, U. S., and Whitesides, G. M. (2012), "A Paper-Based Multiplexed Transaminase Test for Low-Cost, Point-of-Care Liver Function Testing," *Science Translational Medicine*, 4, 152ra129–152ra129. https://doi.org/10.1126/scitranslmed.3003981.

Qiu, P. (2018), "Jump regression, image processing, and quality control," *Quality Engineering*, 30, 137–153. https://doi.org/10.1080/08982112.2017.1357077.

Qiu, P., and Yandell, B. (1998), "A Local Polynomial Jump-Detection Algorithm in Nonparametric Regression," *Technometrics*, 40, 141. https://doi.org/10.2307/1270648.

Ramsay, J. O., and Silverman, B. W. (2002), *Applied functional data analysis: Methods and case studies*, Springer series in statistics, New York: Springer.

Ruppert, D., Wand, M. P., and Carroll, R. J. (2003), *Semiparametric Regression*, Cambridge University Press.

Stone, C. J. (1980), "Optimal Rates of Convergence for Nonparametric Estimators," *The Annals of Statistics*, 8, 1348–1360. https://doi.org/10.1214/aos/1176345206.

Stone, C. J. (1982), "Optimal Global Rates of Convergence for Nonparametric Regression," *The Annals of Statistics*, 10, 1040–1053. https://doi.org/10.1214/aos/1176345969.

Tsanas, A., Little, M., McSharry, P., and Ramig, L. (2010), "Accurate Telemonitoring of Parkinson's Disease Progression by Noninvasive Speech Tests," *IEEE Transactions on Biomedical Engineering*, 57, 884–893. https://doi.org/10.1109/TBME.2009.2036000.

Wang, Y. (1995), "Jump and Sharp Cusp Detection by Wavelets," *Biometrika*, 82, 14.

Wasserman, L. (2006), *All of nonparametric statistics*, Springer texts in statistics, New York: Springer.

Xia, Z., and Qiu, P. (2015), "Jump information criterion for statistical inference in estimating discontinuous curves," *Biometrika*, 102, 397–408. https://doi.org/10.1093/biomet/asv018.