Title

1 Abstract

2 Introduction

In recent years, the use of various shrinkage and thresholding procedures has become ever more popular in a variety of applications where sparsity is a desirable notion. Here the term "sparsity" is loosely defined in the sense that it also includes the case where there are many near-zero (as opposed to exactly zero) terms, as shrinkage procedures often produce. The most prominent usage of such procedures appear in multiresolution denoising analyses such as wavelets, where sparsity in a transformed domain often corresponds to smoothness in the data space [references]. Other examples include estimation of covariance and precision matrices (though thresholding is used more often here since zero terms are usually desired) [references], accounting for different measurement precisions in FDR control for multiple testing [references], or more generally optimizing penalized likelihood problems such as LASSO [references]. In this paper we will look at a flexible and adaptive shrinkage method named ASH, and illustrate its usage in two key areas of univariate wavelet denoising.

Wavelet estimators have become extremely popular in nonparametric regression problems following the seminal paper by Donoho & Johnstone (1994). Not only are wavelet methods locally adaptive in the sense that they achieve the optimal minimax rate over a wide variety of functions, but they are also computationally faster than many other adaptive methods such as variable-bandwidth kernel methods [antoniadis 2001]. While classical thresholding estimators have been shown to possess optimal asymptotic properties, an extensive simulation study by Antoniadis et al. (2001) has shown that various Bayesian shrinkage methods can perform just as well, if not better than, thresholding methods in finite samples where performance is measured by mean squared error (MSE). However, many Bayesian wavelet methods are often computationally intensive, and may not be practical for large scale applications. Furthermore, it is a nontrivial task to extend the same thresholding or Bayesian shrinkage rules to other nonparametric regression problems, two of which we will now describe.

The first problem relates to the frequently encountered Gaussian case. Although the theory for mean estimation with homoskedastic Gaussian errors has been thoroughly

developed in the wavelet literature [references], the existing methods for variance estimation with heteroskedastic errors have been far and few between. Fan & Yao (1998) estimated the variance by smoothing the squared residuals using local polynomial smoothing, while Brown & Levine (2007) employed difference-based kernel estimators. More recently Cai & Wang (2008) also proposed a wavelet thresholding approach using first order differences. The problem of joint mean and variance estimation as described is generally a nontrivial one, and we will demonstrate how we can make use of ASH to tackle this problem.

The second task is to denoise a Poisson distributed signal. This often occurs in the experimental sciences such as gamma-ray burst signals in astronomy [references], and more recently in genetics where high throughput sequencing data is of interest. This problem is particularly interesting in the sense that there are many different ways to perform signal recovery. Variance stabilizing techniques together with normal approximations have been proposed by Donoho (1993) and Fryzlewicz & Nason (2001), by noting that the variance of a Poisson count is equal to its mean. Kolaczyk (1997, 1999a) derived thresholds achieving optimal asymptotic properties in the context of wavelet transformations, similar to the thresholds in the i.i.d. Gaussian case. Multiscale analysis using recursive dyadic partitions within a Bayesian framework has also been developed by Kolaczyk (1999) to make use of a particular form of likelihood factorization. We will discuss the pros and cons of these and some other methods in the simulation section, and further compare them to our method which uses ASH as the main shrinkage procedure in a Bayesian framework similar to that of Kolaczyk (1999b).

Both these problemss are generally harder to deal with than the classical problem of i.i.d. Gaussian errors, and usually requires some work in extending the ideas from methods aimed at the latter task. Before proceeding to describe the two problems in detail, we will first describe briefly the portion of ASH that performs shrinkage.

describe ash

The flexibility and computational efficiency of this shrinkage method is immediately obvious. Gaussian mixture priors can effectively approximate any unimodal distribution, allowing us to apply this single shrinkage procedure to the two problems we will be focusing on. At the same time, we can enhance the computational speed of the the entire signal denoising problem via likelihood approximations, which will be discussed in detail in the next section. For an in-depth understanding of the motivations and applications of ASH, see Stephens (??). With the main shrinkage method accounted for, we will now proceed to explore the two aforementiond problems in detail.

3 Method

Note: the notation should be consistent in the two applications

3.1 Gaussian denoising with heteroskedastic errors

We first consider the application of ASH in the context of the nonparametric regression model with heteroskedastic (and independent) Gaussian errors that are spatially structured. Specifically, consider the model

$$Y(t) = \mu(t) + \epsilon(t) \tag{1}$$

for $t = 1, ..., T$, where $\boldsymbol{Y}$ is the vector of observations, $\boldsymbol{\mu}$ is the mean curve sampled on an equally spaced grid, and $\epsilon(t)$ are independent $N(0, \sigma_t^2)$ noise. To better understand and motivate our method, we first give a brief introduction to classical Bayesian denoising techniques in the wavelet literature.

In the case of i.i.d Gaussian noise, given an observation vector $\boldsymbol{Y}$ of length T (where $T = 2^J$ for some positive integer J) and an orthogonal $T \times T$ matrix representing the orthonormal wavelet basis chosen, we have that the wavelet coefficients $\boldsymbol{d}$ are given by

$$\boldsymbol{d} = W\boldsymbol{Y} \tag{2}$$

where $\mathbb{V}(Y) = \sigma^2 I$. Hence, we can easily see that

$$\boldsymbol{d} \sim N_T(\boldsymbol{\alpha}, \sigma^2 I) \tag{3}$$

where $\boldsymbol{\alpha} = W\mathbb{E}(\boldsymbol{Y})$. This implies that the likelihood for $\boldsymbol{\alpha}$ factorizes into a product of likelihoods for $\alpha(t)$, $t = 1, ..., T$, since a diagonal covariance matrix for a multivariate Gaussian random vector implies independence of the components in the vector. As such, a natural and computationally convenient approach is to set independent priors on $\alpha(t)$ as a form of "shrinkage". Although we have adopted the priors from ASH in our method, the usual choice for the prior is a "spike and slab" prior, ie

$$\alpha_{jk} = \pi_0 \delta_0 + (1 - \pi_0) N(0, \sigma_j^2) \tag{4}$$

where π_0 and σ_j^2 are hyperparameters. Note that, for notational convenience, we have switched to a double index following standard wavelet convention for indices: here $j = 1, ..., log_2(T)$ is the resolution level, and $k = 0, ..., 2^j - 1$ is the location within each resolution level j. Given the independent prior and factorized likelihoods for $\boldsymbol{\alpha}$, one can then easily obtain the posterior distribution of each α_{jk} since they will also be independent. The posterior mean or median $\hat{\boldsymbol{\alpha}}$ is used to estimate the true $\boldsymbol{\mu}$ by applying the inverse wavelet transform on $\hat{\boldsymbol{\alpha}}$. One could also construct credible bands using the posterior variances.

This now sets the stage for our method. One key obstacle when dealing with heteroskedastic errors is that the likelihood for $\boldsymbol{\alpha}$ does not necessarily factorize. If the true variances were known, one could obviously write out the full likelihood and proceed to compute the posterior mean/median using some specified prior. However,

this would be computationally cumbersome, and may not be feasible when extended to multiple signals or higher dimensions. A suitable prior might also be difficult to find. As such, one key aspect of our approach is to treat the wavelet coefficients as if they were independent, so that the true likelihood is approximated by a composite likelihood (see Silverman (1999) for an example where this is done).

To describe our approach in more detail, first assume that the true variance function is known, and that the true mean function is given by $\boldsymbol{\mu} = (\mu(1), ..., \mu(T))$, where $T = 2^J$ for some positive integer J. Hence

$$\boldsymbol{Y} \sim N_T(\boldsymbol{\mu}, \Lambda) \tag{5}$$

where $\Lambda = diag(\lambda(1), ..., \lambda(T))$ is a diagonal matrix. Now let W be the matrix associated with the NDWT for the Haar wavelet basis, so that

$$\boldsymbol{d} \sim N_T(\boldsymbol{\alpha}, \tilde{\Lambda}) \tag{6}$$

where $\boldsymbol{d}$ is the vector of detail coefficients, $\boldsymbol{\alpha} = W\boldsymbol{\mu}$, and $\tilde{\Lambda} = W\Lambda W^T$. By treating the likelihood for $\boldsymbol{\alpha}$ as if it were independent, we can write the likelihood as follows (using the double index mentioned above):

$$L(\boldsymbol{\alpha}|\boldsymbol{d}) = \prod_{j=0}^{J} \prod_{k=0}^{T-1} P(d_{jk}|\alpha_{jk}) \tag{7}$$

where $P(d_{jk}|\alpha_{jk}) = \phi(d_{jk}; \alpha_{jk}, \tilde{\Lambda}_{(jk,jk)})$. Note that there are T coefficients at each resolution level instead of 2^j for resolution j because we are using the non-decimated wavelet transform (NDWT) instead of the standard wavelet transform. Here ϕ is the Gaussian density function. Since Λ is diagonal, it is easy to see that $\tilde{\Lambda}_{(jk,jk)} = \sum_{i=1}^{T} \Lambda_{ii} W_{jk,i}^2$. In our method, we use ASH to assign independent priors to α_{jk}:

$$\alpha_{jk} = \sum_{i=1}^{m} \pi_i^{(j)} N(0, (\sigma_i^{(j)})^2) \tag{8}$$

with $\sum_i \pi_i^{(j)} = 1$, for $0 = 1, ..., J$. Here $\pi_i^{(j)}$ and $\sigma_i^{(j)})^2$ are hyperparameters that are shared between coefficients in the same resolution level. Since the prior is a Gaussian mixture and the likelihoods are Gaussian, the posterior is also a Gaussian mixture, with closed form expressions for the posterior mean and variance of α_{jk}. Here we use ASH to perform posterior inference. We estimate α_{00} using the corresponding scaling coefficient, which seems intuitively appealing. Finally, we can obtain an estimate of μ by using the average basis inverse, which is essentially an average of the inverse wavelet transforms for every shift of the data (see Coifman & Donoho (1995)) since we are using the NDWT. Although the method described here uses a matrix formulation for easier conceptual understanding, the actual NDWT and inverse transform are done through Mallat's pyramid algorithm, which takes only $T \log(T)$ time.

While we assumed that the true variances are known for mean estimation, the problem of variance estimation itself is a non-trivial one. Here we make the reasonable assumption that the variance function is also spatially structured, in addition to the mean function. The approach we took is similar in spirit to that of Cai & Wang (2008), since we make use of wavelet decomposition as well. However, while they use first order differences, we look at the squared residuals. As such, we need a reasonably accurate estimator of the mean function in order to form sensible residuals. At the same time, estimating the mean function in the presence of heteroskedasticity requires that we know the variance function up to some degree of accuracy, and so this process is an iterative one. A natural initial estimate would be the square of first order differences between adjacent points, as discussed in previous variance estimation papers (eg Cai & Wang (2008)). This estimator has the property that it is approximately unbiased for the variance at the two corresponding points, provided that the mean and the variance function is smooth enough. Furthermore, since the estimator is approximately χ^2 distributed, it is right skewed and thus overestimates the true variance in general. This results in smoother estimates of the mean function, which is generally acceptable for an initial estimate. Following that, we estimate the mean function by treating the variance estimates as if they were the truth, and re-estimate the variance function using squared residuals. Although this iterative procedure (using squared residuals) could go on for a long time, and convergence is difficult to prove, we found via simulations that two cycles is usually sufficient to produce relatively accurate estimates. Hence, our approach can be described in three steps:

1. Using squared first order differences as our estimates of the unknown variance function, we estimate the mean function as if the variance function was known (see the above procedure for the case when the variance function is known)

2. Given the estimate of the mean function, take squared residuals and treat those as "observations" for the variance function. Next, we project them into wavelet space and apply some form of shrinkage to the wavelet coefficients, before projecting them back into data space to obtain an estimate of the variance function.

3. Repeat steps (1)-(2) once to obtain the final estimates of the mean and variance functions respectively.

In the first step, the initial variance estimates are defined by

$$\hat{\sigma}_i^2 = \frac{1}{2}\left(\sum_{i=1}^{T}(Y_i - Y_{i-1})^2 + \sum_{i=1}^{T}(Y_i - Y_{i+1})^2\right) \tag{9}$$

where $Y_0 \equiv Y_T$ and $Y_{T+1} \equiv Y_1$, since wavelet methods typically assume that the function is periodic. Hence, the missing element to complete the description of our method would be the shrinkage of the wavelet coefficients for the variance "observations" in step (2) (and similarly in step (3)) above. This is a difficult problem in

general, so we simplify the task by using Gaussian approximations to the likelihoods and set independent mixture Gaussian priors. Of course, better likelihoods and priors could be used that reflect the skewness in the distributions of the "observations", but we found that a normal approximation works reasonably well (especially for smoother variance functions), and has the key advantage of being easy and fast to implement.

To describe the shrinkages in steps (2) (and (3)) in more detail, we use

$$Z^2(t) = (Y(t) - \hat{\mu}(t))^2 \tag{10}$$

as our "observations". As long as the mean function estimate $\hat{\mu}(t)$ is reasonably accurate, we have that $\mathbb{E}(Z^2(t)) \approx \sigma^2(t)$, so that $Z(t)$ is approximately unbiased for the true variance at point t. At this point we essentially have a mean function estimation problem, with variances that are distributed more-or-less as a χ^2 distribution squared. We then approximate this likelihood with a Gaussian likelihood as previously mentioned, and set a Gaussian mixture prior as in ASH. The variance estimation process is then very similar to one for mean estimation as described above, with a few extra details. The full procedure is outlined in Appendix A.

3.2 Poisson denoising

Next we turn to another popular nonparametric regression problem and demonstrate how we could make use of ASH. Specifically, we assume an underlying signal λ_k, $k = 1, ..., n$ with $n = 2^J$ for some positive integer J. Further assume that each data point Y_k, $k = 1, ..., n$ is realized from a Poisson distribution with mean λ_k ie. $Y_k = Pois(\lambda_k)$. Our goal is to recover the true signal λ_k as accurately as possible, given Y_k. To make our approach easier to understand and relate it to wavelet-based methods, we first summarize the data in a recursive manner (see Kolaczyk (1999) and Nowak (1998) (how to reference technical report?)):

$$Y_{Jl} \equiv Y_l \tag{11}$$

for $l = 1, ..., n$, and

$$Y_{sl} = Y_{s+1,2l} + Y_{s+1,2l+1} \tag{12}$$

for scale $s = 0, ..., J-1$ and location $l = 0, ..., 2^s - 1$, so that we are looking at the sum of adjacent pairs of "points" as we move to coarser resolutions, where a point is already itself a sum of the true data points. Another way to define the $Y's$ is

$$Y_{sl} = \sum_{i=l2^{J-s}+1}^{(l+1)2^{J-s}} Y_i \tag{13}$$

for $s = 0, ..., J$ and $l = 0, ..., 2^s - 1$. Further define the following:

$$\lambda_{Jl} \equiv \lambda_l \tag{14}$$

for $l = 1, ..., n$, and

$$\lambda_{sl} = \lambda_{s+1,2l} + \lambda_{s+1,2l+1} \tag{15}$$

for $s = 0, ..., J-1$ and $l = 0, ..., 2^s - 1$. Furthermore, define

$$\alpha_{sl} = \log(\lambda_{s+1,2l}) - \log(\lambda_{s+1,2l+1}) \tag{16}$$
$$\tag{17}$$

for $s = 0, ..., J-1$ and $l = 0, ..., 2^s - 1$. The α's defined this way is extremely similar to the (true) Haar wavelet coefficients, which forms the basis of our approach. Using this recursive representation, we can see that the likelihood for $\boldsymbol{\alpha}$ factorizes into a product of likelihoods, where $\boldsymbol{\alpha}$ is the vector of all the α_{sl}'s. To be specific, we have

$$\begin{aligned} L(\boldsymbol{\alpha}|\mathbf{Y}) &= P(\mathbf{Y}|\boldsymbol{\alpha}) && (18)\\ &= P(Y_{0,0}|\lambda_{0,0}) \prod_{j=0}^{J-1} \prod_{k=0}^{2^j-1} P(Y_{j+1,2k}|Y_{j,k}, \alpha_{j,k}) && (19)\\ &= L(\lambda_{0,0}|Y_{0,0}) \prod_{j=0}^{J-1} \prod_{k=0}^{2^j-1} L(\alpha_{j,k}|Y_{j+1,2k}, Y_{j,k}) && (20) \end{aligned}$$

where the factorization is due to the recursive definition above. Note that $Y_{00}|\lambda_{00} \sim \text{Pois}(\lambda_{00})$. Now for any give s, l, Y_{sl} is a sum of two independent Poisson random variables, and is itself a Poisson random variable. Hence

$$Y_{s+1,2l}|Y_{sl}, \alpha_{sl} \sim \text{Bin}(Y_{sl}, \frac{1}{1+e^{-\alpha_{sl}}} \equiv \frac{\lambda_{s+1,2l}}{\lambda_{sl}})$$

Whereas Kolaczyk (1999) sets a conjugate prior (which is a mixture of a point mass at 0.5 and a uniform distribution, similar to the "spike and slab" prior for the Gaussian case) for the parameters $\frac{\lambda_{s+1,2l}}{\lambda_{sl}}$ and uses a Binomial likelihood, we felt that the two-component mixture prior is too inflexible. As such, we work with the parameters α_{sl} as defined in (16) and use ASH to set Gaussian mixture priors to the Gaussian-approximated logit-Binomial (is there such a thing?) likelihoods. Furthermore, these parameters can be more easily connected to wavelet coefficients since they consist of (log) differences rather than ratios. For more details regarding Gaussian approximations to the likelihoods for the α's, see Appendix B. Finally, to reconstruct the signal in the original space using the posterior distribution of the α's (which are in the logit space), we make use of the Delta method. For full details on the reconstruction process as well as the Delta method, see Appendix C.

4 Simulations

Note: after simulation results also describe the method to assess performance of methods that denoise Poisson distributed data without knowing the underlying truth

5 Application to real datasets

In this section we apply the Gaussian denoising technique to two example datasets from different fields, and run the Poisson denoising method on a sequencing dataset. Whereas the application to the sequencing dataset is straightforward, the other two datasets require some pre-processing due to the presence of repeated observations and the fact that the number of sample points is not a power of two. To deal with these complications, we first use the median of the repeated observations at their respective sample points (see eg Delouille et al. (2004)). Next, we mirror the data about the right edge and extract the first $2^{\lfloor \log_2(2n) \rfloor}$ sample points. This ensures that the number of data points in the new "dataset" is a power of two, and the mean curve would be continuous at the right edge. To further ensure that the input to the Gaussian denoising method is periodic, we then reflect the new dataset about the right edge and use this as the final input. Finally, we extract the first n points from the output of our denoising technique as an estimate of the true mean function. Although the data points are not evenly spaced (in contrast to the requirements of standard wavelet methods), we took the simplest approach and applied the method treating the observations as if they were evenly spaced. This approach is not only intuitively appealing, but can also be considered a formal treatment of unequally spaced data in traditional wavelet settings (see Sardy et al. (1999)).

5.1 Motorcycle Acceleration Data

Here we apply our Gaussian denoising method to a classical dataset from Silverman (1985). The observations are measurements of head acceleration simulated motorcycle accidents in order to test the effectiveness of crash helmets. The response variable here is acceleration in g, and the predictor variable is time in milliseconds (ms). Note that the observations are not evenly spaced here, and there may be multiple observations at one point. These issues were dealt with as per discussion above.

5.2 Three-month Treasury Bill Yields

The second dataset involves the yields of three-month Treasure Bill from secondary market rates. There are 1735 weekly observations, starting from 5 January 1962 and ending on 31 March 1995. This dataset has been analyzed extensively by others, including Andersen & Lund (1997), Gallant & Tauchen (1997) and Fan & Yao (1998). We preprocess the data following the procedure described in Fan & Yao (1998):

5.3 ChIP-Seq Data

Since the original goal of the Poisson denoising procedure is smoothing next generation sequencing data that is common in the field of genetics, we apply our method to an example dataset from the ENCODE (**Enc**yclopedia **O**f **D**NA **E**lements) project launched by the National Human Genome Research Institute. We specifically looked at reads from ChIP sequencing measuring transcription factor binding in two different cell types, with two samples for each cell type. Due to the massive size of the data, we selected a representative portion of the reads of length 2^{15} from chromosome 1. Since we are looking mainly at univariate denoising here, we run our method on each sample separately to produce the following plots.

6 Discussion

In this paper we have briefly introduced the adaptive shrinkage method ASH; while it was originally developed in the setting of FDR control for multiple comparisons, we have demonstrated its flexibility by utilizing it as part of two wavelet denoising techniques. Both the applications discussed here are not often seen in the wavelet literature because they are typically harder to deal with than the case for i.i.d. Gaussian noise. Nevertheless, they are interesting problems in their own right, as many real world datasets do not satisfy the standard assumptions of having i.i.d. Gaussian noise. In the Gaussian case, our software allows users to easily obtain point estimates for both the mean function and the variance function, the latter of which can be used to provide frequentist confidence intervals for other forms of mean estimation (to the best of our knowledge, there is no readily available software which implements both mean and variance estimation in the wavelet literature). Furthermore, approximate credible intervals are also computed for both the mean and the variance to provide uncertainty estimates. We have also tested our method and found that it is relatively robust to simple forms of autocorrelation between the errors so long as it is not too strong (eg AR? with autocorrelation ?). In the case of Poisson regression, we have improved upon the conjugate Beta priors used in conjunction with the binomial likelihoods by utilizing ASH as the shrinkage procedure, which allows for more flexibility and accuracy. Our method is also much faster and comparable in accuracy to the popular Haar-Fisz algorithm, due to the fact that we do not have to perform external cycle-spinning. In addition, we have proposed a new procedure to assess the performance of a given method on real datasets under the assumption that they follow a Poisson distribution, without knowing the underlying truth. In both of the applications, one major advantage of both our methods is that there is no tuning parameter other than the type of wavelet basis used, whereas the so-called primary resolution level in almost all of the other wavelet-based methods actually affects their performance substantially, depending on the underlying mean and/or variance function. Hence, our fully adaptive procedures allow users to easily apply them to any given dataset.

We have also demonstrated through numerical studies that our methods mostly outperform their respective counterparts from standard wavelet literature in terms of pointwise accuracy (MSE in this case). Furthermore, the simplicity of the approximated Gaussian likelihoods as well as the conjugacy of the mixture Gaussians used indicate that our methods are computationally fast, making them superior to many other Bayesian wavelet techniques. Note that we have not directly compared the computation times of our methods with all the methods listed in the simulation studies, due to the fact that most standard wavelet techniques are part of a package in Matlab whereas our method is coded entirely in R. However, the methods that we did compare against in R indicate that our methods are only slightly slower than the non-Bayesian methods and much faster than the Bayesian ones. On the other hand, since ASH is computationally efficient, we have chosen to sacrifice speed for accuracy in the wavelet transformation stage by using the NDWT (which is $O(n \log n)$), while most standard wavelet methods use the DWT(which is $O(\log n)$). For a more detailed summary of computation times see ??? (maybe?).

Although we have only focused on one-dimensional univariate denoising here (effectively regression problems with only one covariate), our methods can be extended to various scenarios. Even in one dimension, our methods could be extended to the case with multiple samples, otherwise known as regression analysis of functional data (See Morris ??). Instead of dealing with a vector of observations, we could perform regression analysis on a matrix of observations, each row of which would encapsulate a sample with observations that are temporally or spatially structured. While Morris (2006) proposed a way to solve a generic regression model, they implicitly assumed the same variance structure for each sample in the same group or category. Our work potentially allows for differing variance structures amongst all the samples, thereby imposing even less restrictions on the assumptions behind the regression model. In the simplest case, we could potentially obtain spatially structured differences between the samples by including a single covariate into the regression model that categorizes each sample. We have already developed methods for this scenario for both the Gaussian and the Poisson data discussed in this paper. The Poisson model is extremely useful for discovering regions in sequencing reads where differential expression is present between say, different tissues, as per our sequencing example in the previous section. The Gaussian model could also potentially be used in...(???).

With some work, our methods could also be extended to higher dimensions which have a wider range of applications. For example, we could attempt a straight extension to the two dimensional case for both the Gaussian and the Poisson cases as described in Nowak (1998) (again, reference technical report). However, recent advances in image denoising problems have shown that wavelet transformations might not be the most ideal due to the presence of smooth curves in many images such as photographs. We could thus attempt to use ASH as a shrinkage procedure in other types of transformations such as curvelets, and that could be a potential direction for future work.

7 Reference

Appendix A

Variance estimation for Gaussian denoising

With $\boldsymbol{Z}$ as defined in (10), we apply the wavelet transform W to $\boldsymbol{Z}^2$, and obtain the wavelet coefficients $\boldsymbol{\delta} = W\boldsymbol{Z}^2$. Note that $\mathbb{E}(\boldsymbol{\delta}) \approx (\boldsymbol{\gamma})$, where $\boldsymbol{\gamma} = W\boldsymbol{\sigma}^2$. As with (7), we treat the likelihood for $\boldsymbol{\gamma}$ as if it were independent, resulting in

$$L(\boldsymbol{\gamma}|\boldsymbol{\delta}) = \prod_{j=0}^{J} \prod_{k=0}^{T-1} P(\delta_{jk}|\gamma_{jk}) \tag{21}$$

However, the likelihoods $L(\gamma_{jk}|\delta_{jk})$ are not normal, and have no simple closed form expressions. As such, we approximate the likelihood by a normal likelihood through matching the moments of a normal distribution to the distribution $P(\delta_{jk}|\gamma_{jk})$ i.e.

$$P(\delta_{jk}|\gamma_{jk}) \approx N(\gamma_{jk}, \hat{\mathbb{V}}(\delta_{jk})) \tag{22}$$

so that

$$L(\gamma_{jk}|\delta_{jk}) \approx \phi(\delta_{jk}; \gamma_{jk}, \mathbb{V}(\delta_{jk})) \tag{23}$$

where ϕ is the normal density function, and $\mathbb{V}(\delta_{jk})$ is the variance of the detail coefficients. Since these variances are unknown, we estimate them from the data and then proceed to treat them as known. More specifically, since $Z(t) \approx N(0, \sigma^2(t))$, we have that

$$\begin{aligned} & \mathbb{E}(Z^4(t)) \approx 3\sigma^4 \\ \Rightarrow \quad & \mathbb{V}(Z^2(t)) \approx 2\sigma^4 \end{aligned} \tag{24}$$

and so we simply use $\frac{2}{3}Z^4(t)$ as an unbiased estimator for $\mathbb{V}(Z^2(t))$. It then follows that $\hat{\mathbb{V}}(\delta_{jk})$ is given by $\sum_{i=1}^{T} \frac{2}{3}Z^4(i)W_{jk,i}^2$, and is unbiased for $\mathbb{V}(\delta_{jk})$. Although this works well in most cases, there are variance functions for which the above procedure tends to overshrink the detail coefficients at the finer levels. This is likely due to the fact that the distribution of the wavelet coefficients are extremely skewed, especially when the true coefficients are large (at coarser levels the distributions are much less skewed since we are dealing a linear combination of a large number of data points). One way around this issue is to employ a procedure that jointly shrinks the coefficients $\boldsymbol{\gamma}$ and their variance estimates (see JASH).

Now that we have the likelihood, the specification of the prior is exactly the same as in (8): independent Gaussian mixtures for every detail coefficient. ASH then provides us with the posterior mean, which will be used to obtain the final estimate of the variance function via the average basis inverse across all the shifts .

Appendix B

Gaussian Approximation to logit-Binomial Likelihood for Poisson denoising

Dropping the subscripts, we want to use a Gaussian likelihood to approximate $L(\alpha)$. Since the true variance of $\hat{\alpha}$ is unknown, the most natural approximation would be $\phi(\alpha; \hat{\alpha}, se(\hat{\alpha})^2)$. This choice of parameters for the Gaussian approximation is also appropriate in the sense that it minimizes the Kullback-Leibler divergence between a Gaussian distribution and the likelihood for α (also known as moment matching).

We next consider consider appropriate estimators for the parameters ie. $\hat{\alpha}$ and $se(\hat{\alpha})$. The obvious choice for $\hat{\alpha}$ and $se(\hat{\alpha})$ would be the MLE of α and its asymptotic standard error respectively, given by

$$\hat{\alpha}_{MLE} = \log(S/F) \qquad S = 0, ..., n \tag{25}$$

$$se(\hat{\alpha}_{MLE}) = \sqrt{SF/n} \qquad S = 0, ..., n \tag{26}$$

where S is the number of successes in the binomial setup, and $F = n - S$ is the number of failures (in our case the even locations would be successes and odd locations failures). The MLE has one notable drawback however, in that it cannot deal well with extreme cases. This would in turn affect the normal approximation in a non-negligible way. To improve the accuracy of the approximation at the endpoints of the binomial distribution (ie. when we have $\hat{p} = 0$ or n, so that $\hat{\alpha} = -\infty$ or ∞ respectively), while at the same time ensuring that $\hat{\alpha}$ is approximately unbiased for α (so that moment matching still makes sense), we propose a mix of Berkson's estimator and Tukey's estimator (see Gart & Zweifel (1967)). Specifically, our final estimators for α and its standard error are given by

$$\hat{\alpha} = \begin{cases} \log\{(S + 0.5)/(F + 0.5)\} - 0.5 & S = 0 \\ \log\{S/F\} & S = 1, 2, ..., n-1 \\ \log\{(S + 0.5)/(F + 0.5)\} + 0.5 & S = n \end{cases} \tag{27}$$

$$se(\hat{\alpha}) = \sqrt{V^*(\hat{\alpha}) - \frac{1}{2}\{V_3(\hat{\alpha})\}^2 \left\{V_3(\hat{\alpha}) - \frac{4}{n}\right\}} \tag{28}$$

where

$$V_3(\hat{\alpha}) = \frac{n+1}{n}\left(\frac{1}{S+1} + \frac{1}{F+1}\right) \quad S = 0, ..., n \tag{29}$$

$$V^*(\hat{\alpha}) = V_3(\hat{\alpha})\left\{1 - \frac{2}{n} + \frac{V_3(\hat{\alpha})}{2}\right\} \tag{30}$$

This variance estimator is V^{**} from p. 182 of Gart & Zweifel (1967), and is chosen because it is less biased (when n is small) as compared to the asymptotic variance

of the MLE (see Gart & Zweifel (1967)). The other two variance estimators from Gart & Zweifel (1967), V_1^{++} and V^{++}, were also used in simulations and gave similar results, but V^{**} was chosen for its simplicity.

Appendix C

Signal reconstruction for Poisson denoising

Given the posterior means and variances of the α's from ASH, the first step to reconstructing the signal is to find the posterior means of $p_{sl} := \frac{\lambda_{s+1,2l}}{\lambda_{sl}}$ and $q_{sl} := \frac{\lambda_{s+1,2l+1}}{\lambda_{sl}}$ (for $s = 0, ..., J-1$ and $l = 0, ..., 2^s - 1$). Specifically, for each s and l, we wish to find

$$E(p_{sl}) \equiv E\left(\frac{e^{\alpha_{sl}}}{1+e^{\alpha_{sl}}}\right) \tag{31}$$

$$E(q_{sl}) \equiv E\left(\frac{e^{-\alpha_{sl}}}{1+e^{-\alpha_{sl}}}\right) \tag{32}$$

Given that we already have the posterior expectations and variances for α_{sl}, we can approximate (31)-(32) using the Delta method. First, define

$$ff(x) = \frac{e^x}{1+e^x} \tag{33}$$

and consider the Taylor expansion of $ff(x)$ about $ff(E(x))$:

$$ff(x) \approx ff(E(x)) + ff'(E(x))(x - E(x)) + \frac{ff''(E(x))}{2}(x - E(x))^2 \tag{34}$$

where

$$ff'(x) = \frac{e^x}{(1+e^x)^2} \tag{35}$$

$$ff''(x) = \frac{e^x(1-e^x)}{(1+e^x)^3} \tag{36}$$

It is easy to see that

$$E(p_{sl}) \approx ff(E(\alpha_{sl})) + \frac{ff''(E(\alpha_{sl}))}{2} Var(\alpha_{sl}) \tag{37}$$

$$E(q_{sl}) \approx ff(-E(\alpha_{sl})) + \frac{ff''(-E(\alpha_{sl}))}{2} Var(\alpha_{sl}) \tag{38}$$

noting that we have already computed $E(\alpha)$ and $Var(\alpha)$.

Finally, we can easily back-transform to construct an estimated signal, by noting that we can express $\hat{\lambda}_k$ as a product of the p's and q's for any $k = 1, 2, ..., n$. Specifically, let $\{c_1, ..., c_J\}$ be the binary representation of $k-1$, and $d_m = \sum_{i=1}^{m} c_i 2^{m-i}$ for $i = 1, ..., J-1$. We then have

$$\hat{\lambda}_k = \lambda_{00} p_{00}^{1-c_1} p_{1,d_1}^{1-c_2} ... p_{J-1,d_{J-1}}^{1-c_J} q_{00}^{c_1} q_{1,d_1}^{c_2} ... q_{J-1,d_{J-1}}^{c_J} \tag{39}$$

where we usually estimate λ_{00} by $\sum_l Y_l$ (see Kolaczyk (1999)). Using the independence of the p's and q's from different scales, we have:

$$\begin{aligned} E(\hat{\lambda}_k) = \lambda_{00} E(p_{00})^{1-c_1} E(p_{1,d_1})^{1-c_2} ... E(p_{J-1,d_{J-1}})^{1-c_J} \\ E(q_{00})^{c_1} E(q_{1,d_1})^{c_2} ... E(q_{J-1,d_{J-1}})^{c_J} \end{aligned} \tag{40}$$

As an additional step, we can also construct a credible band around the signal using the posterior variances for inference purposes. From (39) we have the following:

$$\begin{aligned} E(\hat{\lambda}_k^2) = \lambda_{00}^2 E(p_{00}^2)^{1-c_1} E(p_{1,d_1}^2)^{1-c_2} ... E(p_{J-1,d_{J-1}}^2)^{1-c_J} \\ E(q_{00}^2)^{c_1} E(q_{1,d_1}^2)^{c_2} ... E(q_{J-1,d_{J-1}}^2)^{c_J} \end{aligned} \tag{41}$$

To compute the terms in (41), we again make use of the Delta method (with $ff(x) = (\frac{e^x}{1+e^x})^2$) to obtain:

$$\begin{aligned} E(p_{sl}^2) \approx \left(ff(E(\alpha_{sl})) + \tfrac{ff''(E(\alpha_{sl}))}{2} Var(\alpha_{sl}) \right)^2 + \\ \{ff'(E(\alpha_{sl}))\}^2 Var(\alpha_{sl}) \end{aligned} \tag{42}$$

$$\begin{aligned} E(q_{sl}^2) \approx \left(ff(-E(\alpha_{sl})) + \tfrac{ff''(-E(\alpha_{sl}))}{2} Var(\alpha_{sl}) \right)^2 + \\ \{ff'(E(-\alpha_{sl}))\}^2 Var(\alpha_{sl}) \end{aligned} \tag{43}$$

Finally we combine (40) and (41) to find $Var(\hat{\lambda}_k)$, which allows us to construct credible intervals.

Note here that in order for the reconstructed signal to possess the property of shift invariance (see Coifman & Donoho (1995)), the α's are extracted from a so-called translation invariant (TI) table (see Coifman & Donoho (1995), and Kolaczyk (1999)) rather than as described above. The idea remains the same however, and we can simply think of the extra α's as being defined similarly as the original α's, albeit from a shifted version of the original data points. To be more specific, the TI table contains the α_{sl} for all circulant shifts of the signal. Here we define the t-th shift of the signal $\boldsymbol{Y}$, denoted by $\boldsymbol{Y}^{(t)}$, to be created from $\boldsymbol{Y}$ itself by moving the first $n-t$ elements of $\boldsymbol{Y}$ t positions to the right and then putting the last t elements of $\boldsymbol{Y}$ in the first t locations. Using this table, we are essentially computing the posterior expectations

in (40)-(41) by averaging over all posterior expectations for every shift of the original signal ie.

$$\frac{1}{n}\sum_{t=1}^{n} E(\hat{\lambda}_k^{(t)}) \tag{44}$$

which is an approximation to the true quantity we wish to compute, given by

$$E(\hat{\lambda}_k) = \sum_{t=1}^{n} E(\hat{\lambda}_k^{(t)}) P(t\text{-th shift}) \tag{45}$$