

DATA ANALYSIS METHODS FOR THE DETECTION OF THE EPOCH OF REIONIZATION

CARINA CHENG¹, ET AL.

¹Astronomy Dept., U. California, Berkeley, CA

ABSTRACT

The Epoch of Reionization (EoR) is an uncharted era in our Universe’s history during which the birth of the first stars and galaxies led to the ionization of neutral hydrogen in the intergalactic medium. There are many experiments investigating the EoR by tracing the 21 cm line of neutral hydrogen, a signal which is very faint and difficult to isolate. With a new generation of instruments and a statistical power spectrum detection in our foreseeable future, it has become increasingly important to develop techniques that help maximize sensitivity and validate results. Additionally, it is imperative to understand the trade-offs between different methods and their effects on common power spectrum themes. In this paper, we focus on three major themes — signal loss, power spectrum error bar estimation, and bias in measurements. We describe techniques that affect these themes using both a toy model and data taken by the 64-element configuration of the Donald C. Backer Precision Array for Probing the Epoch of Reionization (PAPER). [CC: motivate a bit more once the paper is complete]

1. INTRODUCTION

By about one billion years after the Big Bang ($z \sim 6$), the very first stars and galaxies are thought to have ionized all the neutral hydrogen that dominated the baryonic matter content in the Universe. This transition period, during which the first luminous structures formed from gravitational collapse and began to emit intense radiation that ionized the cold neutral gas into a plasma, is known as the epoch of reionization (EoR). The EoR is a relatively unexplored era in our cosmic dawn. Its history encodes important information regarding the nature of the first galaxies and the processes of structure formation. Direct measurements of the EoR would unlock powerful information about the intergalactic medium, revealing connections between the smooth matter distribution exhibited via cosmic microwave background (CMB) studies and the highly structured web of galaxies we observe today.

One promising technique to probe the EoR is to target the 21 cm wavelength emission that is emitted by neutral hydrogen via its spin-flip transition. This technique is powerful because it can be observed as a function of redshift — that is, the wavelength of the signal reaching our telescopes can be directly mapped to a distance from where the emission originated before stretching out as it traveled through expanding space. Tracing the 21 cm line as a function of redshift offers a window into the evolution of ionization, temperature, and density fluctuations during this transformative era.

Although a detection of the EoR remains elusive, there are several radio telescope experiments that have succeeded in using the 21 cm signal from hydrogen to place constraints on the brightness of the EoR. Examples of

experiments investigating the mean brightness temperature of the EoR relative to the CMB are the Experiment to Detect the Global EoR Signature (EDGES; [Bowman & Rogers 2010](#)), the Long Wavelength Array (LWA; [Ellingson et al. 2009](#)), the Large Aperture Experiment to Detect the Dark Ages (LEDA; [Greenhill & Bernardi 2012](#)), the Dark Ages Radio Explorer (DARE; [Burns et al. 2012](#)), the Sonda Cosmológica de las Islas para la Detección de Hidrógeno NeutroSciHi (SCI-HI; [Voytek et al. 2014](#)), the Broadband Instrument for Global HydrOgen ReioNisation Signal (BIGHORNS; [Sokolowski et al. 2015](#)), and the Shaped Antenna measurement of the background RAdio Spectrum (SARAS; [Patra et al. 2015](#)). Radio interferometers which seek to measure statistical power spectra include the Giant Metre-wave Radio Telescope (GMRT; [Paciga et al. 2013](#)), the LOw Frequency ARray (LOFAR;[van Haarlem et al. 2013](#)), the Murchison Widefield Array (MWA; [Tingay et al. 2013](#)), the 21 Centimeter Array (21CMA; [Peterson 2004](#); [Wu 2009](#)), and PAPER ([Parsons et al. 2010](#)). The Hydrogen Epoch of Reionization Array (HERA), which is currently being built, is a next-generation instrument that aims to combine lessons learned from previous experiments and is forecasted to be able to make a successful high-significance power spectrum detection with an eventual 350 elements using current analysis techniques ([DeBoer et al. 2017](#); [Pober et al. 2014](#)).

The major challenge that faces all 21 cm experiments is isolating a small signal that is buried underneath foregrounds and instrumental systematics that are, when combined, 4–5 orders of magnitude brighter. A clean measurement therefore requires an intimate understanding of the instrument and a rigorous study of data analysis choices. With continual progress being made in

the field and HERA on the horizon, it is becoming increasingly important to understand how the methods we choose interact with each other to affect power spectrum results. More specifically, it is imperative to develop techniques and tests that ensure the accuracy and reliability of a potential EoR detection. In this paper, we discuss three themes that are essential to investigate for a robust 21 cm power spectrum analysis. We also highlight four power spectrum techniques and their trade-offs, traps, and connections to the themes. We first approach the themes from a broad perspective, and then perform a detailed case study using data from the 64-element configuration of PAPER.

This paper is organized as follows. In section 2 we introduce the three themes of our focus, using a toy model to develop intuition for each one. Section 3 presents a case study into each theme using data from the PAPER-64 array, highlighting key changes from [Ali et al. \(2015\)](#). We conclude in Section 7.

2. POWER SPECTRUM THEMES AND TECHNIQUES

There are many choices for a data analyst. How can time-ordered measurements be combined? How can the variance of the data be estimated? In what way(s) can the data be weighted to suppress contaminated modes while not destroying an EoR signal? How can the source of a detection be properly identified? Many common techniques, such as averaging data, weighting, bootstrapping, and jackknife testing, address these issues but harbor additional trade-offs. For example, an aggressive filtering method may succeed in eliminating interfering systematics but comes at the cost of losing some EoR signal. A chosen weighting scheme may maximize sensitivity but fail to suppress foregrounds.

Despite there being many data analysis choices, measuring the statistical 21 cm power spectrum ultimately requires robust methods for determining accurate confidence intervals and rigorous techniques to identify and suppress systematics. In this paper, we focus on three 21 cm power spectrum themes that encapsulate this goal and discuss four techniques that interplay with each other and impact the themes. We will give brief definitions now, and build intuition for each theme in the sections to follow.

Power Spectrum Themes

A deep understanding of the following three themes is essential for the accuracy and interpretation of a 21 cm power spectrum result. Stemmed from a re-analysis of PAPER-64 data, we believe these themes serve as an important check-list for a rigorous power spectrum analysis.

- **Signal Loss** (Section 2.1): As explained in the next section, certain analysis techniques can lead to the loss of the EoR signal. If not corrected for, it could lead to a false non-detection. Computing

signal loss has subtle challenges but is crucial to the accuracy of any result.

- **Error Bar Estimation** (Section 2.2): Confidence intervals on the 21 cm power spectrum result determine the difference between a detection and a null result, which have two very different implications. Errors can be estimated in a variety of ways, and we will discuss a few of them.
- **Bias** (Section 2.3): There are several possible sources of bias in a visibility measurement that can show up as a detection in a power spectrum, such as bias from noise and foregrounds. A successful EoR detection would also imitate a bias. Proving a bias is EoR may be the most difficult challenge for 21 cm analyses, as it is crucial to be able to distinguish a detection of foreground leakage, for example, from that of EoR. In this paper we will highlight some sources of bias and discuss ways to mitigate their effects.

Power Spectrum Techniques

The following techniques each have advantages when it comes to maximizing sensitivity and understanding systematics in data. We focus on these techniques because [CC: ??]. However, some have limitations, and we will discuss circumstances in which there are trade-offs.

- **Fringe-rate filtering:** Fringe-rate filtering is considered an averaging scheme for time-ordered data ([Parsons et al. 2016](#)), and we explain the trade-offs of filtering in more detail in Section 2.1. Broadly, a fringe-rate filter increases the sensitivity of a dataset and reduces the number of independent samples by an amount dependent on the width of the averaging window. However, it can also affect the presence of foregrounds and systematics.
- **Weighting:** A dataset can be weighted to emphasize certain spectral features and minimize others. One particular flavor of weighting is inverse covariance weighting, which is a generalized version of inverse variance weighting that also takes into account data correlations. This weighting has the effect of down-weighting correlated information (i.e. foregrounds) and up-weighting noise-like information (i.e. EoR). However, a challenge of inverse covariance weighting is in accurately describing a covariance matrix that best describes the data.
- **Bootstrapping:** Because theoretical models for covariance matrices are questionable and theoretical error estimation methods rely on assumptions that can be violated by data, applying theoretical errors to a 21 cm measurement is often avoided.

Hence, bootstrapping is a useful method for estimating errors of a dataset from itself. By randomly drawing many samples of the data, we get a sense of its inherent variance, though there are subtleties to consider such as the independence of values in a dataset.

- **Jackknife testing:** A resampling technique useful for estimating bias, jackknives can be taken along different dimensions of a dataset to cross-validate results. In particular, null tests can be used to verify whether results are free of systematics (Keating et al. 2016).

In the next three subsections, we go into each theme in depth, focusing on how power spectrum technique trade-offs affect each. We use a toy data model to develop intuition into why certain analysis choices may be appealing and discuss ways in which they are limited. We highlight problems that can arise regarding each theme and offer suggestions to mitigate the issues. Ultimately, [CC: some kind of summarizing sentence here].

2.1. Signal Loss

In this section we illustrate how signal loss arises in analysis. Signal loss refers to attenuation of the target cosmological signal in a power spectrum estimate. It can arise in a variety of ways in the analysis pipeline, such as fitting a polynomial during spectral calibration, applying a delay-domain filter, or by weighting data by itself. Here we focus on signal loss associated with applying a weighting matrix to data, a loss that can be significant depending on the choice of weighting and one that was previously underestimated in the PAPER-64 analysis.

Driven by the need to mitigate foreground bias, we use a weighting method that succeeds in down-weighting foregrounds. We apply this weighting to data and propagate it into a final estimator using the power spectrum estimation technique of optimal quadratic estimators (OQE; Liu & Tegmark 2011; Liu et al. 2014b). Before showing how signal loss can arise when using different weighting matrices, we first summarize OQE as follows.

We begin with our data vector, \mathbf{x} which contains our measured visibilities in Jy. It has length (N_t, N_f) , where N_t is the number of time integrations and N_f is the number of frequency channels. Visibilities are measurements of the Fourier transform of the sky along spatial dimensions, and since we are interested in 3-dimensional Fourier-modes we only need to take one Fourier transform along the line-of-sight dimension. We do this when forming the un-normalized power spectrum estimate \hat{q}_α :

$$\hat{q}_\alpha = \frac{1}{2} \mathbf{x}^\dagger \mathbf{R} \mathbf{Q}_\alpha \mathbf{R} \mathbf{x}, \quad (1)$$

where \mathbf{R} is a weighting matrix. As an example, inverse covariance weighting would set $\mathbf{R} = \mathbf{C}^{-1}$ and an unweighted case would use $\mathbf{R} = \mathbf{I}$, the identity matrix. The matrix \mathbf{Q} is an operator that takes our frequency-domain visibilities and Fourier-transforms them into

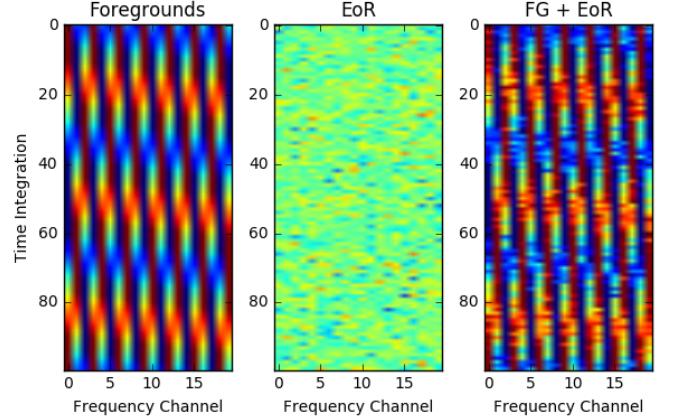


Figure 1. Our toy model dataset contains a sinusoid foreground that varies in time and frequency and a random Gaussian signal as a mock EoR signal. Real parts are shown here. [CC: colorbar?]

power spectrum space while also taking into account physical constants such as the Boltzmann constant used to convert Jy to Kelvin. The index α denotes a waveband in k_\parallel , where k_\parallel is the Fourier-dual to frequency under the delay approximation (Parsons et al. 2012).

We normalize our power spectrum estimates using the matrix \mathbf{M} :

$$\hat{\mathbf{p}} = \mathbf{M} \hat{\mathbf{q}}, \quad (2)$$

where $\hat{\mathbf{p}}$ is the normalized estimate of the true power spectrum. We have a choice for \mathbf{M} . For simplicity in this section we set $\mathbf{M} \propto \mathbf{I}$, although we explore other cases for the analysis of PAPER-64 data as explained in Section 3.3

In the next sections, we investigate the effects of weighting matrices on signal loss by experimenting with different matrices \mathbf{R} and examine their impact on the resulting power spectra estimate $\hat{\mathbf{p}}$. Our ultimate goal in experimenting with weighting is to suppress foregrounds perfectly without destroying an EoR signal.

2.1.1. Toy Model: Inverse Covariance Weighting

To build our intuition into how a particular choice of weighting, namely inverse covariance weighting, can lead to signal loss, we use a toy model. We construct a simple dataset that contains 2-dimensional data (representing visibility data with 100 time integrations and 20 frequency channels). This model represents realistic dimensions of about an hour of PAPER data which might be used for a power spectrum analysis. We create a mock bright foreground signal, \mathbf{x}_{FG} , as a complex sinusoid that varies in time and frequency, a simplistic but realistic representation of a single bright source. We also create a mock EoR signal, \mathbf{x}_{EoR} , as a complex, Gaussian random signal. Our combined data vector is $\mathbf{x} = \mathbf{x}_{FG} + \mathbf{x}_{EoR}$, to which we apply different weighting schemes. The three data vectors are shown in Figure 1.

One choice for a weight matrix \mathbf{R} is an inverse covariance matrix. This type of weighting is attractive for

power spectrum analyses because it is an aggressive way to down-weight foregrounds and yields the smallest possible error bars on a measurement (Tegmark 1997; Bond et al. 1998). The covariance matrix \mathbf{C} , which in our case describes covariances between frequency channels, can be estimated in a variety of ways. Given perfect foreground, instrumental, and EoR models, we could form \mathbf{C} in a way that accurately describes our measured data. However, if our foreground model is flawed for example, our estimate of \mathbf{C} would not be successful at down-weighting them in the data.

One attractive way to estimate \mathbf{C} is to empirically derive it from the data vector \mathbf{x} itself:

$$\hat{\mathbf{C}} \equiv \langle \mathbf{x} \mathbf{x}^\dagger \rangle_t, \quad (3)$$

assuming $\langle \mathbf{x} \rangle = 0$, where $\langle \rangle$ denotes a finite average over time t . The inverse covariance matrix is therefore $\mathbf{R} = \hat{\mathbf{C}}^{-1}$.

First, we compute the power spectrum of \mathbf{x} using the OQE formalism and a weighting matrix of $\hat{\mathbf{C}}^{-1}$. The result is shown in green in the left plot of Figure 2. Also plotted in the figure are the unweighted power spectrum of \mathbf{x}_{FG} (blue) and \mathbf{x}_{EoR} (red).

As shown, our inverse covariance-weighted result successfully suppresses foregrounds. It is also evident that our result fails to recover the EoR signal — it exhibits the correct shape, but the amplitude level is slightly low. This is evidence of signal loss. In order to understand the behavior of this result, we need to closely study our covariance matrix, $\hat{\mathbf{C}}$, which is shown in Figure 3.

If \mathbf{C} is computed from the data itself, it carries the risk of over-fitting information in the data and introducing a multiplicative bias to estimates of the signal. For a mathematical derivation of signal loss arising from a data-estimated covariance matrix, see Appendix A. Here we will describe the origin of this signal loss intuitively.

It turns out that because we estimated \mathbf{C} from our data, its eigenspectrum offers insight into understanding how inverse covariance weighting affects our result. An eigenspectrum ranks the eigenvalues of a matrix from highest to lowest and can be thought of as a spectrum of weights that are given to each frequency mode in the data. In other words, the eigenvalues encode the strength of different shapes in the dataset. The eigenspectrum of the identity matrix \mathbf{I} is flat (all 1's) because it gives equal weighting to all modes. This is usually not the case for a covariance matrix, in which a sloped eigenspectrum means that modes are given different weights. When weighting data by an inverse covariance matrix, the modes with the highest eigenvalues are down-weighted the most.

If the true covariance matrix \mathbf{C} of our data was known, then every single eigenvalue of \mathbf{C} would be representative of real fluctuations in the data. However, when using an estimated $\hat{\mathbf{C}}$ that is derived from one particular data realization, some of its eigenvalues may be describing random fluctuations that just happen to ex-

ist in the data realization. Said differently, shapes that may not exist (or have a weaker existence) in a true covariance may appear stronger in the estimated covariance. Hence, they will be down-weighted more than they should be.

In general, the strongest modes of $\hat{\mathbf{C}}$ (highest eigenvalues) are more trustworthy than the weakest modes. Bright foregrounds usually dominate and would therefore exist in both \mathbf{C} and $\hat{\mathbf{C}}$. This is demonstrated in the toy model by the successful suppression of the foreground mode, where our estimated covariance matrix identifies the sinusoid and assigns it the highest eigenvalue (the peak in Figure ??).

The danger of an empirically estimated covariance matrix comes from not being able to describe weak eigenmodes accurately. The weak eigenmodes of $\hat{\mathbf{C}}$ may characterize random noise fluctuations which will be down-weighted. This is what we call the ‘overfitting’ of noise, which leads to signal loss.

Using what we’ve learned about the eigenspectrum, we can tweak it in a simple way to suppress foregrounds and yield zero signal loss. Namely, we can project out the zeroth eigenmode (i.e. zero out all eigenmodes except the first one), thereby down-weighting the foregrounds perfectly and nothing else. Hence, we are not using information from all the weaker eigenmodes which carry the risk of describing fluctuations that may not exist in the true covariance. Altering $\hat{\mathbf{C}}$ in this way is one example of a regularization method, in which we are changing $\hat{\mathbf{C}}$ in a way that flattens its eigenspectrum, making it more identical to that of \mathbf{I} . The resulting power spectrum estimate for this case is shown in the right plot of Figure 2. In this case we recover EoR, demonstrating that if we can model our foregrounds, we can down-weight them without signal loss. There are several other ways to regularize $\hat{\mathbf{C}}$, and we will discuss some in Section 2.1.3.

2.1.2. Toy Model: Fringe-Rate Filtering

We have shown how signal loss can arise due to inaccurately characterizing weak eigenmodes with a data-estimated covariance. We will next show how this effect is exaggerated by reducing the total number of independent samples in a dataset.

A fringe-rate filter is an analysis technique designed to maximize sensitivity by integrating in time (Parsons et al. 2016). Rather than a traditional box-car average, a fringe-rate filter can be designed to up-weight parts of the sky that an instrument is most sensitive to, while down-weighting the opposite. We apply a fringe-rate filter to PAPER-64 data in order to optimally combine our time-ordered measurements. We weight data based on our primary beam sensitivity, yielding a net increase in sensitivity in our measurements.

Because fringe-rate filtering is analogous to averaging in time, it comes at the cost of reducing the total number of independent samples in the data. To mimic this filter, we average every four time integrations of our toy model dataset together, yielding 25 independent sam-

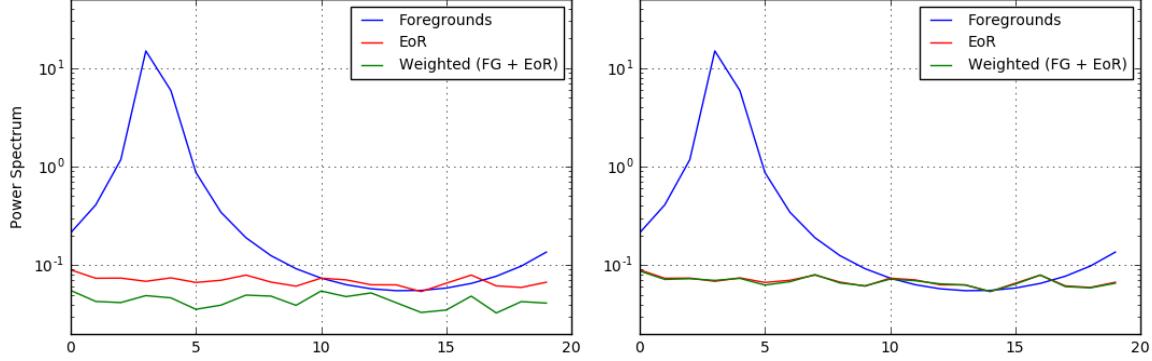


Figure 2. Resulting power spectrum estimates for the toy model simulation described in Section 2.1.1 — foregrounds only (blue), EoR only (red), and the weighted FG + EoR dataset (green). We use inverse covariance weighting where \mathbf{C} is derived from the data (left), contrasted with projecting out the first eigenmode only (right). In the former case, signal loss arises from using information from all eigenmodes of $\hat{\mathbf{C}}$. Because $\hat{\mathbf{C}}$ is empirically estimated, its eigenvalues pose the risk of describing random fluctuations that happen to exist in the data realization but may not exist in the true covariance. Consequently, these modes are over-fitted and down-weighted, leading to signal loss. There is no signal loss when using only the zeroth eigenmode, since we are not using information from the weaker eigenmodes which $\hat{\mathbf{C}}$ does not describe accurately.

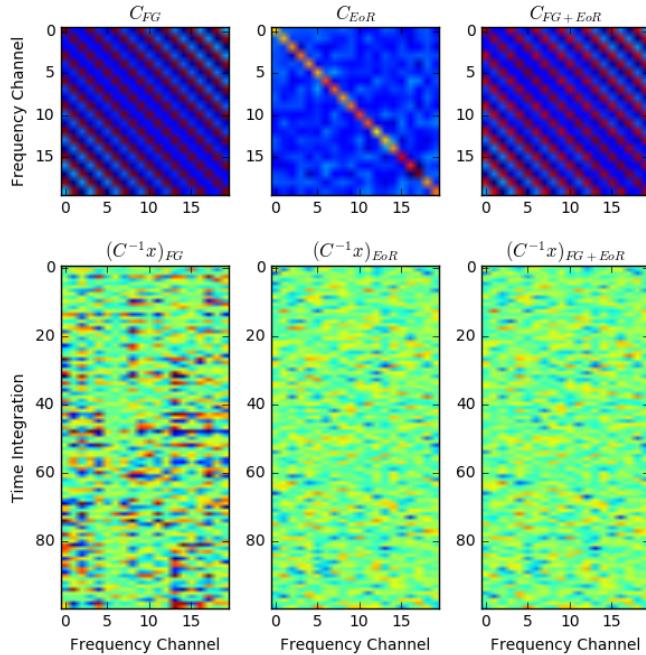


Figure 3. The covariance matrices (top row) and inverse covariance-weighted data (bottom row) for FG only, EoR only, and FG + EoR. Real parts are shown here. [CC: col-orbar?]

ples in time (Figure 5, left). We choose these numbers so that the total number of independent samples is similar to the number of frequency channels — therefore, our matrices will be full rank.

The resulting eigenspectra (Figure 5, right), as compared to those in Figure 4, fall more steeply, especially for the last few eigenmodes. This is because we have fewer independent modes — fewer shapes in the data — so we do a worse job characterizing the weak eigenmodes. Therefore, there is a more dramatic difference

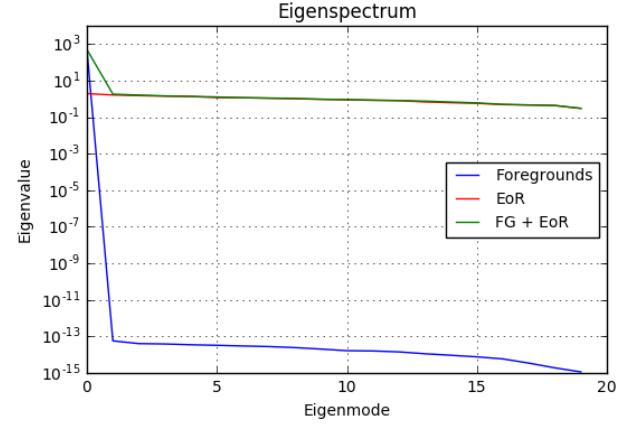


Figure 4. Eigenspectrum of $\hat{\mathbf{C}}_{FG}$ (blue), $\hat{\mathbf{C}}_{EoR}$ (red), and $\hat{\mathbf{C}}_{FG+EoR}$ (green). The eigenspectrum of $\hat{\mathbf{C}}_{FG}$ peaks at the zeroth eigenmode, due to the presence of only one sinusoid.

in weighting between the modes and we end up down-weighting random noise-like fluctuations more severely than for the un-fringe-rate filtered case.

The power spectrum results for inverse covariance weighted fringe-rate filtered data is shown in Figure 6. As expected, there is a much larger amount of signal loss for this time-averaged dataset. Additionally, because of the steepened eigenspectrum, most of our power spectrum information is coming from only the last few modes and as a result, it is a noisier estimate. This is evident by noticing that the green curve in Figure 6 fails to trace the shape of the unweighted EoR power spectrum.

Using our toy model, we have seen that a sensitivity-driven analysis technique like fringe-rate filtering has trade-offs of signal loss and noisier estimates when using data-estimated covariance matrices. Longer integrations increase sensitivity but reduce the number of independent samples, resulting in weakly characterized, steep eigenspectra that can overfit noise greatly.

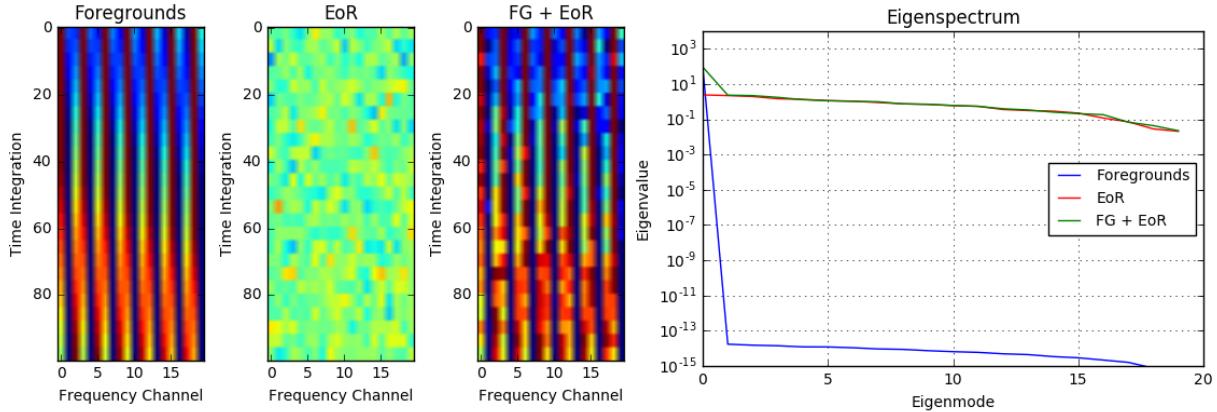


Figure 5. Left: Our ‘fringe-rate filtered’ (time-averaged) toy model dataset. We averaged every four samples together, yielding 25 independent samples in time. Real parts are shown here. Right: Eigenspectrum of the covariance matrices of foregrounds only (blue), EoR only (red), and FG + EoR (green). The eigenspectrum of foregrounds only peaks at the zeroth eigenmode, a consequence of the presence of only one sinusoid. This spectrum is steeper than that in Figure 4, resulting in more dramatic difference in weighting between eigenmodes. Consequently, if all eigenmodes are used in weighting it is possible to down-weight random noisy modes more severely than for the un-fringe-rate filtered case.

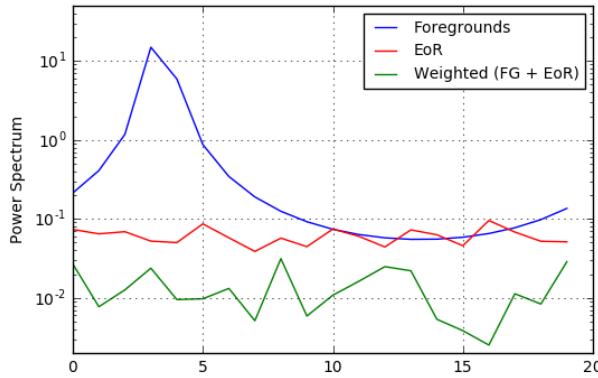


Figure 6. Resulting power spectrum estimate for the fringe-rate filtered toy model simulation — foregrounds only (blue), EoR only (red), and the weighted FG + EoR dataset (green). We use inverse covariance weighting where \mathbf{C} is derived from the data. There is a larger amount of signal loss than for un-fringe-rate filtered data, a consequence of a steepened, weakly characterized eigenspectrum from using fewer independent modes in the data. This example shows that it is possible to down-weight modes more severely for the un-fringe-rate filtered since there is a more dramatic difference in weighting between modes. The power spectrum estimate is also noisier, since most of the information used in constructing the estimate came from only a few eigenmodes.

2.1.3. Toy Model: Other Weighting Options

In Section 2.1.1 we showed one example of how altering $\hat{\mathbf{C}}$ can make the difference between zero and some signal loss, if we can distinguish between real eigenmodes in a true covariance matrix from random fluctuation-induced eigenmodes in an estimated one. We will now use our toy model to describe several other ways to tailor $\hat{\mathbf{C}}$ in order to minimize signal loss. We illustrate the resulting power spectra and eigenspectra for four different cases in Figures 7 and 8.

As a first test, we know that our simulated EoR should have a covariance matrix that mimics the identity matrix, with its variance encoded along the diagonal. If we

model \mathbf{C}_{EoR} as such, instead of computing it based on the dataset itself, and add it to $\hat{\mathbf{C}}_{FG} = \langle \mathbf{x}_{FG} \mathbf{x}_{FG}^\dagger \rangle$ to obtain a final $\hat{\mathbf{C}}$ to use in weighting, we see that there is negligible signal loss (Figure 7, upper left). This is because by modeling \mathbf{C}_{EoR} , we avoid over-fitting small fluctuations in the data that our model doesn’t know about (but an empirically derived $\hat{\mathbf{C}}$ would). This is evident when comparing the green and red curves in Figure 8. We don’t see perfect EoR recovery though, as the green and red curves differ at one k value in Figure 7. This deviation is a consequence of the difference between the true and modeled covariance matrices.

The second panel (top right) in Figure 7 uses a regularization method of setting $\hat{\mathbf{C}} = \hat{\mathbf{C}} + \mathbf{I}$. By adding the identity, we are weighting the diagonal elements of the matrix more heavily than those off-diagonal, thereby flattening out its eigenspectrum. If all modes are given similar weights, we avoid down-weighting artificial modes more than others.

The third panel (bottom left) in Figure 7 flattens out the eigenspectrum of $\hat{\mathbf{C}}$ a different way - by zeroing all but the first three eigenmodes. This means that the first three eigenmodes will be given weights, but the rest will be treated with similar weights. Again, flattening the eigenspectrum results in negligible signal loss. However, we do not perfectly recover the shape of EoR because we lost information when projecting out certain modes.

The last regularization scheme we are highlighting here is setting $\hat{\mathbf{C}} = \hat{\mathbf{C}} \cdot \mathbf{I}$. In the bottom right panel of Figure 7, we see that this method does a poor job down-weighting foregrounds. For this toy model, our foregrounds are spread out in frequency and therefore have non-negligible frequency-frequency correlations. Multiplying by the identity results in a diagonal matrix, meaning we are only left with correlation information between the same two frequencies. Because we disregard information from all other frequency combinations, we do a poor job suppressing the foreground. But because

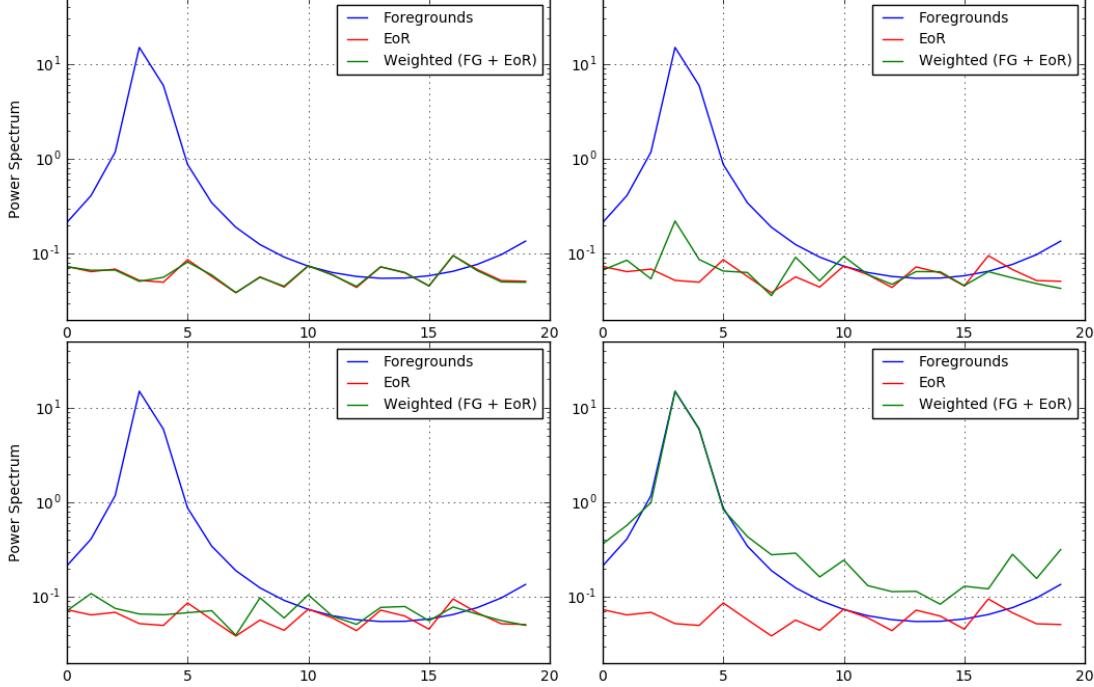


Figure 7. Resulting power spectra estimates for our fringe-rate filtered toy model simulation — foregrounds only (blue), EoR only (red), and the weighted FG + EoR dataset (green). We show four alternate weighting options that each avoid signal loss, including modeling the covariance matrix of EoR (upper left), regularizing \hat{C} by adding an identity matrix to it (upper right), using only the first three eigenmodes of \hat{C} (lower left), and multiplying an identity matrix to \hat{C} (lower right). There are many regularization options available to minimize signal loss, and some work better than others depending on the dataset.

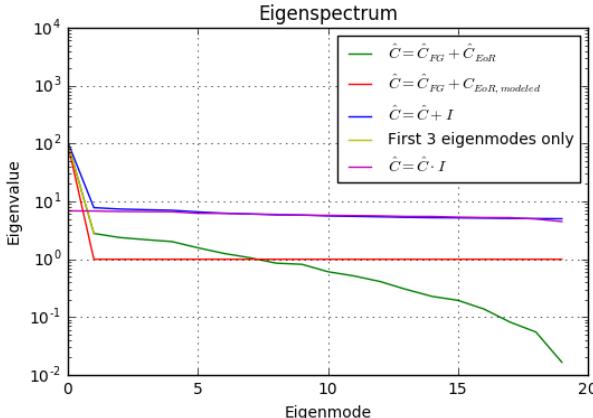


Figure 8. We compare the eigenspectrum of an empirically calculated \hat{C} (green) to that of four alternate weighting options, including modeling the covariance matrix of EoR (red), regularizing \hat{C} by adding an identity matrix to it (blue), using only the first three eigenmodes of \hat{C} (yellow), and multiplying an identity matrix to C (magenta).

we flattened the eigenspectrum, we also avoid signal loss.

Although the fourth method did not successfully recover EoR for this particular simulation, it is important that we show that there are many options for estimating a covariance matrix, and some may work better than others based on the dataset. **[CC: how to make this paragraph less hand-wavy?]** One may imagine a situation where a particular systematic is contained to an isolated frequency. In such a case, preserving only the

diagonal elements of \hat{C} would be an effective way of removing this contamination.

In summary, we have a choice of how to weight 21 cm data. Ideally, we want to down-weight bright foregrounds without removing the underlying cosmological signal. As we investigated however, there are trade-offs between the weighting method used, its foreground-removal effectiveness, and the amount of resulting signal loss.

2.2. Error Estimation

Our second major 21 cm power spectrum theme is error estimation, as we desire robust methods for determining accurate confidence intervals for our measurements. Two ways of estimating errors on a power spectrum measurement are calculating the variance of a dataset, and computing a theoretical error estimate based on an instrument's system temperature and observational parameters. In a perfect world, both methods would match up. However, in practice the two don't always agree due to a number of factors, including time, frequency, and antenna-dependent noise and non-uniform weightings. Therefore, it is important to place error bars on our measurements that have been derived from its inherent variance.

A common technique used to estimate the error in a measurement is bootstrapping. Bootstrapping uses sampling with replacement to estimate a posterior distribution. For example, power-spectral measurements of 21 cm data can be made along many axes, including

time and baselines. Through the process of resampling and averaging along these axes, we can estimate error bars for our results which represent the underlying distribution of power spectra values that are allowed by our measurements.

Again, we focus on toy models to highlight traps that one can fall into when bootstrapping power spectra. Suppose we have a Gaussian random signal dataset of length $N = 1000$ and unity variance. We are interested in the average of the dataset, and predict that the error on the mean should obey $1/\sqrt{N}$, where N is the number of samples.

We form 100 bootstraps, each comprised of an array of length 1000 that is created by a random resampling of the original data, with replacement. The standard deviation over the 100 bootstraps gives an error estimation for our dataset. As shown in Figure 9, the error computed from bootstrapping matches our theoretical prediction.

One major caveat of bootstrapping arises when working with correlated data. If, for example, a dataset has many repeated values inside it, this would be reflected in each bootstrap. The same value would be present multiple times within a bootstrap and also be present between bootstraps, purely because it has a more likely chance of being drawn if there are many of them. Therefore, bootstrapping correlated data results in a smaller variation between bootstraps, and hence, under-estimates errors. The use of a fringe-rate filter, which averages data in time to increase sensitivity, is one example which leads to a reduction in the number of independent samples, creating a situation in which errors can be under-estimated. We will now show this effect using our toy model.

Going back to our toy model, we apply a sliding boxcar average to 10 samples at a time, thus reducing the number of independent samples to $1000/10 = 100$. Bootstrapping this time-averaged noise, using the same method as described earlier (drawing the same number of samples as the length of the dataset), under-estimates the error by a factor of ~ 3 . This occurs because we are drawing more samples than independent ones available, and thus some samples are repeated multiple times in all bootstraps, leading to less variation between the bootstraps. In fact, the error derived from bootstrapping is a strong function of the number of samples that are drawn (Figure 9, black points), and we can both under-estimate the error by drawing too many or over-estimate it by drawing too few. However, if we know that we have 100 independent samples, the error associated with drawing 100 samples with replacement does match the theoretical prediction.

This examples highlights the importance of understanding how analysis techniques (e.g. fringe-rate filtering) can affect a common statistical procedure like bootstrapping. Bootstrapping as a means of estimating power spectrum errors from real fringe-rate filtered data requires knowledge of the number of independent sam-

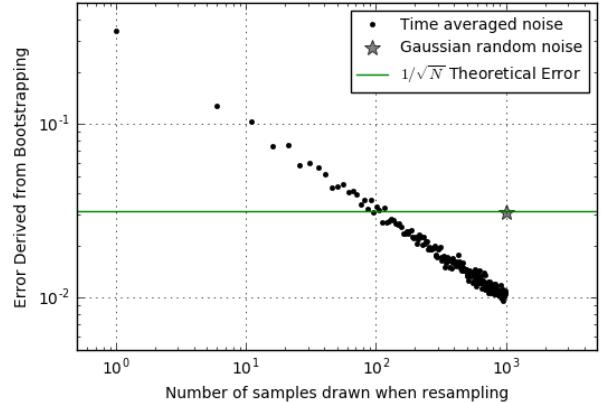


Figure 9. Error estimation produced by bootstrapping as a function of the number of samples drawn when sampling with replacement. The star represents the error associated with drawing 1000 samples from a length 1000 array of a Gaussian random signal. The black points correspond to time-averaged data which has 100 independent samples. They illustrate how errors can be underestimated if drawing more samples than there are independent samples in the data. The estimated errors match up with the theoretical prediction only at $N = 100$.

ples, which is not always a trivial task. For example, computing the effective number of independent samples of fringe-rate filtered data is not as simple as counting the number of averages performed. Additionally, we can have a non-integer effective number of samples due to the weighting.

We will now discuss a second subtle feature of bootstrapping that can lead to an over-estimation of errors. Suppose we have 5 independent measurements of the sky (from 5 different baselines, for example). A bootstrap then consists of 5 measurements that are drawn randomly with replacement from the original set. If all 5 spaces are filled randomly, there is a high probability that some measurements will be repeated in the bootstrap because they are drawn more than once. The bootstrap may therefore consist of only 3 or 4 independent measurements — a number smaller than the total number of samples. In fact, the probability of drawing 5 completely independent measurements is less than 4%.

In order to maximize sensitivity, we desire as many independent samples as possible. However, solely using all 5 independent measurements does not allow variation between bootstraps.

Therefore, we use a slightly modified bootstrapping method. For each bootstrap, we first shuffle the data. Next, we take the first 4 samples, and fill the last slot randomly with replacement. In doing so there is a small chance a bootstrap will consist of 5 independent samples, but even with 4 our sensitivity is nearly maximized. One may wonder whether this change is still a legitimate way of error estimating since random sampling only occurs for one value in a bootstrap. However, as long as the number of possible variations is greater than the number of bootstraps we perform, it is a valid way to uncover the inherent variability in a dataset [CC: ex-

plain why].

In Figure 10 we compare the two methods of bootstrapping: sampling all elements randomly (black points) versus sampling just the final element randomly (grey points). The two converge for datasets with small numbers of elements, when filling a few spots randomly is nearly the same as filling only the last spot randomly. However, there is also increased scatter in this regime due to small number statistics.

The difference between the two methods is most pronounced for datasets with large numbers of elements. As a dataset grows in size, it becomes increasingly rare to draw entirely independent samples, and thus the benefit of ensuring, say, 999 independent samples out of 1000 is most noticeable. The more elements in our dataset, the more sensitivity we can gain by mandating that most of our measurements are independent ones. An analog for the y-axis in Figure 10 is therefore the sensitivity of a final power spectrum measurement.

In summary, bootstrapping can be an effective and straightforward way to estimate errors of a dataset. However, we have illustrated two situations in which bootstrapping can lead to inaccurate error estimation. Trade-offs exist between averaging data and the validity of bootstrapping — working with correlated data gives rise to questions concerning the effective number of independent samples, how to best calculate it, and whether it is valid to simply use that number when bootstrapping. We have also shown a situation in which a naive bootstrapping method can over-estimate errors, thereby compromising measurement sensitivity. While bootstrapping is convenient because it provides a way to estimate errors from the data itself, one must assess whether certain analysis choices have compromised the method and whether a variation of traditional resampling could be preferred instead.

2.3. Bias

A power spectrum detection could be EoR, but it could also be attributed to other sources of bias. Connecting a detection to EoR as opposed to noise or foreground bias is a key challenge of 21 cm data analyses. In this section we will discuss possible sources of bias in a measurement, as well as techniques that can help mitigate their effects. We will also present a series of essential tests that a legitimate EoR detection must pass and highlight how these tests can be used to identify biases. These tests serve as a starting point for the necessary task of verifying a future EoR detection.

2.3.1. Foreground and Noise Bias

Foreground bias is perhaps one of the main limiting factors pushing up against 21 cm results. Foreground signals lie $\sim 4\text{-}5$ orders of magnitude above the cosmological signal, and there are many existing techniques to avoid or remove these strong signals. Despite current best efforts to do so, however, there remains some foreground leakage that can show up as detections in a power spectrum. [CC: how do we know that?]

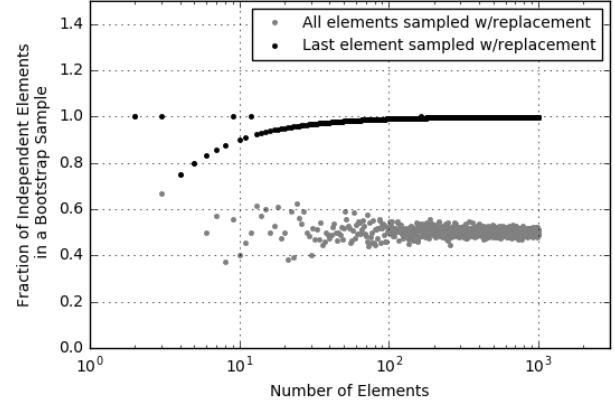


Figure 10. The fraction of independent elements in a bootstrap as a function of the total number of elements in a dataset. A fraction of 1.0 means that all elements in a bootstrap are independent, and therefore sensitivity is maximized (alternatively, this axis can be thought of as power spectrum sensitivity). Two bootstrapping methods are shown here — sampling all elements with replacement (grey) and sampling only the last element with replacement (black). The former over-estimates error bars since some elements end up being repeated due to random sampling. Therefore, sensitivity is not maximized to its full capacity.

Because foreground spectra are smooth in frequency, they preferentially show up at low delay, or k modes. For a particular baseline length, there is a maximum delay imposed on foregrounds, which corresponds to the light-crossing time between the two antennas in the baseline. For longer baselines, this value increases, producing what is known as “the wedge” (Parsons et al. 2012; Liu et al. 2014a; Liu et al. 2014b; Vedantham et al. 2012; Thyagarajan et al. 2013). The wedge describes a region in k -space contaminated by foregrounds, bounded by baseline length (which is proportional to k_{\perp}) and delay (which is proportional to k_{\parallel}). Properties of the wedge can be used to isolate and remove foregrounds, as done by Ali et al. (2015), Parsons et al. (2014), and Jacobs et al. (2015).

Despite making power spectrum measurements outside of the wedge in the “EoR window”, foreground detections are still common at low k values. This leakage can be attributed to convolution kernels associated with Fourier-transforming visibilities into delay-space. In other words, smooth-spectrum foregrounds appear as δ -functions in delay-space, convolved by the Fourier transform of the source spectrum and the antenna response, both of which could smear out the foregrounds and cause leakage outside the wedge.

There are analysis techniques to mitigate the effects of foreground leakage and prevent information from low k 's from spreading to high k values. For example, narrow window functions can be used to minimize the covariance of a particular k value with other ones (Liu et al. 2014b). [CC: math!] In other words, one can construct an estimator using OQE that forces a window function to have a minimum response to low k values. The PAPER-64 window function is constructed in such

a way, specifically to prevent foregrounds that live at low k 's from contaminating higher k -modes (see Section 3.3). Minimizing foreground leakage in this way however, comes with the trade-off of compromising power spectrum sensitivity, since narrow window functions increases errors for each k -mode [CC: develop more or cite?].

Confirming foreground detections at higher k 's is more difficult. In the next section, we will present some tests that can help distinguish these excesses from that of EoR.

In addition to foreground bias, noise bias can also be responsible for positive power spectrum detections. One example is noise bias arising if thermal noise is multiplied by itself. Every 21 cm visibility measurement contains thermal noise that is comprised of receiver and sky noise. We expect this noise to be independent between antennas and thus we can beat it down (increase sensitivity) by integrating longer, using more baselines, etc. However, the squaring of noise occurs when cross-multiplying visibilities, which is shown by the two copies of \mathbf{x} in Equation 1. If both copies of \mathbf{x} come from the same baseline and time, it can result in power spectrum measurements that are higher than those predicted by the thermal noise of the instrument. One way to avoid this type of noise bias is to avoid cross-multiplying data from the same baselines or days. This ensures that the two quantities that go into a measurement have separate noises that don't correlate with each other.

Another type of noise bias can stem from the spurious cross-coupling of signals between antennas. This excess is known as instrumental crosstalk and is an inadvertent correlation between two independent measurements via a coupled signal path. Crosstalk appears as a constant phase bias in time in visibilities, and it varies slowly compared to the typical fringe-rates of sources. Because it is slow-varying, crosstalk can be suppressed using time-averages or fringe-rate filters. However, there remains a possibility that power spectrum detections are caused by residual, low-level crosstalk which survived any suppression techniques.

In the next section, we approach the difficult task of tracing excesses to foreground, noise, and EoR biases through a discussion of useful jackknife tests.

2.3.2. Jackknife Tests

The jackknife is a resampling technique in which a statistic (i.e. power spectrum) is computed multiple times using subsets of the data. In this section we define two main tests — the null test and the traditional jackknife — and explain how a power spectrum detection must pass each. We then highlight how these tests can be used to help distinguish between different sources of bias.

- **Null Test:** A null test is a type of jackknife test that removes the astronomical signal from data in order to investigate underlying systematics. A clean measurement should be noise-like (a ‘null’ re-

sult) if it is not dominated by systematics. There are several ways to perform null tests on data, as implemented in Keating et al. (2016). For example, one can divide data into two subsets by separating odd and even julian dates, or the first half of the observing season from the second. Subtracting the two removes signal that is common to both subsets, including foregrounds and EoR. The resulting power spectrum should be consistent with thermal noise estimates; if it is not, it suggests the presence of a systematic that differs from one of the data subsets to the other (i.e. doesn't get subtracted perfectly).

- **Traditional Jackknife:** In a more broad sense, it is important to perform many jackknife tests in order to instill confidence in a final result, since a stable result should be consistent through all of them. A successful EoR detection must be steadfast throughout all jackknives no matter how the data is sliced, while other sources of bias may come and go depending on the jackknife performed. Jackknives can be taken along several different axes — for example, one could start with a full dataset, and compute a power spectrum each time as a day of data is removed, or a baseline is removed. This type of jackknife would reveal bias present only at certain LSTs (such as a foreground source), for example, or misbehaving baselines.

While the null test hunts for deviations from thermal noise and the jackknife tests for deviations in subsamples, they are both closely related. We can highlight the connection between the two using a toy model dataset.

Suppose we divide the toy model dataset into two subsets, \mathbf{x}_1 and \mathbf{x}_2 , which represent jackknives of the complete dataset \mathbf{x} . Both subsets have dimensions of 100 time integrations and 20 frequency channels, similar to the toy model used in Section 2.1.1. They also have the same level of thermal noise, constructed as a Gaussian random signal for each. Identical EoR signals, also constructed as a Gaussian random signal, are in both subsets.

The subsets differ in that only one of them contains a foreground signal (a single complex sinusoid) on top of the noise and EoR, representing a bias that only appears for half the data. Mathematically,

$$\mathbf{x}_1 = \mathbf{n}_1 + \mathbf{e} + \mathbf{fg} \quad (4)$$

$$\mathbf{x}_2 = \mathbf{n}_2 + \mathbf{e} \quad (5)$$

where \mathbf{n} is noise, \mathbf{e} is the EoR signal, and \mathbf{fg} is the foreground signal.

We do not perform a time-average or apply a fringe-rate filter to this toy model, since we are interested only in what jackknife tests can tell us about biases. For the same reason, we use a weighting matrix of \mathbf{I} for power spectrum estimation to avoid signal loss. We form 5 different power spectrum estimates: $\hat{\mathbf{p}}$ (full dataset),

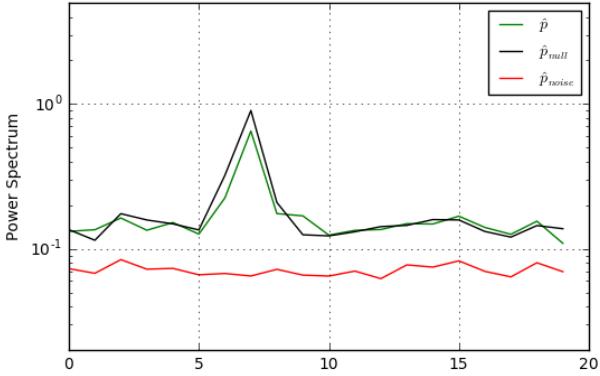


Figure 11. Power spectrum estimates for the full toy model dataset \mathbf{x} (green), the differenced dataset $\mathbf{x}_1 - \mathbf{x}_2$, which represents a null jackknife test (black), and noise alone (red). Because $\hat{\mathbf{p}}_{\text{null}}$ is not consistent with noise, it suggests the presence of a systematic in either \mathbf{x}_1 or \mathbf{x}_2 . Null tests of clean measurements should be consistent with thermal noise.

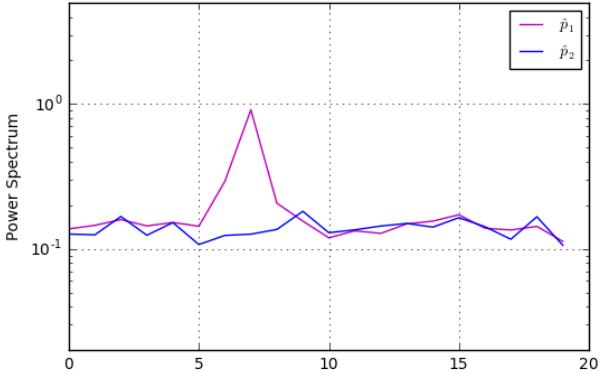


Figure 12. Power spectrum estimates for \mathbf{x}_1 and \mathbf{x}_2 , two jackknives of the toy model. They suggest the presence of a systematic in \mathbf{x}_1 only, illustrating how jackknives can be used to tease out excesses. Clean measurements should remain consistent despite the jackknife taken.

$\hat{\mathbf{p}}_1$ (subset 1), $\hat{\mathbf{p}}_2$ (subset 2), $\hat{\mathbf{p}}_{\text{noise}}$ (noise only), and $\hat{\mathbf{p}}_{\text{null}}$, which is the result of our differenced dataset \mathbf{x}_{null} , defined as:

$$\mathbf{x}_{\text{null}} = \mathbf{x}_1 - \mathbf{x}_2. \quad (6)$$

In Figure 11, we compare $\hat{\mathbf{p}}_{\text{null}}$, $\hat{\mathbf{p}}_{\text{noise}}$, and $\hat{\mathbf{p}}$. Our goal is to determine whether the excess we see in $\hat{\mathbf{p}}$ (the power spectrum peak) is due to the EoR signal, something we would not know for a real dataset. The null test reveals that there is also an excess in $\hat{\mathbf{p}}_{\text{null}}$ — hence, we can rule out EoR. Because this is a toy model, we know that the excess is attributable to the foreground signal that did not get subtracted off for the null test.

Additionally, we look at the jackknife results, namely $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$. The results are shown in Figure 12. Again, we see a clear difference between the two, signifying a non-EoR bias that is only present in one subset.

While the null test is useful for ruling out EoR, jack-

knives are useful in pinpointing which data subsets are contaminated by biases and which are not; in our toy model we see that the bias exists only in $\hat{\mathbf{p}}_1$. If foreground or noise biases exist in a dataset, jackknives can tease them out and provide insight into possible sources. For example, if jackknives along the time-axis reveal a bias present at a certain LST, a likely explanation would be excess foreground emission from a radio source in the sky at that time. A jackknife test involving data before and after the application of a fringe-rate filter can reveal whether cross-talk noise bias is successfully suppressed with the filter, or if similar-shaped detections in both power spectra suggest otherwise. There are many other jackknife axes of which we will not go into detail here, including baseline, frequency, and polarization. Ultimately, an EoR detection should persist through them all and a clean measurement should pass all null tests.

In this section we have highlighted how null tests and jackknife tests are key for determining the nature of a power spectrum detection. In Section 3.3 we perform some examples of these tests on PAPER-64 data.

3. CASE STUDY: PAPER-64

In the previous sections we have discussed three overarching 21 cm power spectrum themes — signal loss, error estimation, and bias. Understanding the subtleties and trade-offs involved in each is necessary for an accurate and robust understanding of a power spectrum result.

We now present a case study of these same three themes using data from the PAPER experiment. We use the intuition we’ve developed through our toy model simulations in order to make smartly-motivated data analysis choices for PAPER. In light of our new understandings, we also highlight major changes since [Ali et al. \(2015\)](#).

As a brief review, PAPER is a dedicated 21 cm experiment located in the Karoo Desert in South Africa. The PAPER-64 configuration consists of 64 dual-polarization drift-scan elements (Figure 13) that are arranged in a grid layout. For our case study, we focus solely on Stokes I data from PAPER’s 30 m East/West baselines, of which there are a total of 56. For information about the backend system of PAPER-64, its observations, and data reduction pipeline, we refer the reader to [Parsons et al. \(2010\)](#) and [Ali et al. \(2015\)](#).

The previously best published 21 cm upper limit result from [\(Ali et al. 2015\)](#) uses 124 nights of data to place a 2σ upper limit on $\Delta^2(k)$, defined as

$$\Delta^2(\mathbf{k}) = \frac{k^3}{2\pi^2} \hat{\mathbf{p}}(\mathbf{k}), \quad (7)$$

of $(22.4 \text{ mK})^2$ in the range $0.15 < k < 0.5 \text{ h Mpc}^{-1}$ at $z = 8.4$. The revision of this limit stems mostly from previously underestimated signal loss and underestimated error bars [CC: cite retraction paper], both of which we address in the case study.

We note that the PAPER-64 dataset that we use in



Figure 13. PAPER dipole in South Africa.

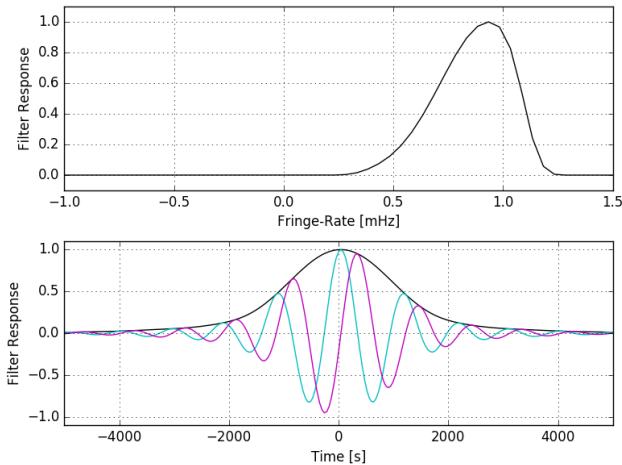


Figure 14. Top: the normalized optimal power-spectrum sensitivity weighting in fringe-rate space for our fiducial baseline and Stokes I polarization beam. Bottom: the time-domain convolution kernel corresponding to the top panel. Real and imaginary components are illustrated in cyan and magenta, respectively, with the absolute amplitude in black. The fringe-rate filter acts as an integration in time, increasing sensitivity but reducing the number of independent samples in the dataset.

this case study differs from that in Ali et al. (2015) only by the fringe-rate filter. In Ali et al. (2015), the applied filter was degraded by widening it in fringe-rate space. This was chosen in order to increase the number of independent modes and reduce signal loss. With the development of a robust method for assessing signal loss, we feel comfortable using a narrow filter — the optimal fringe-rate filter — in order to maximize sensitivity. This filter is computed for a fiducial 30 m baseline at 150 MHz, the center frequency in our band. The filter in both the fringe-rate domain and time domain is shown in Figure 14.

3.1. Case Study: Signal Loss

In Section 2.1, we showed how signal loss arises when weighting data using information from itself. Here we

describe a methodology that simulates the injection and recovery of a cosmological signal in order to quantify the amount of signal loss accompanying a weighting scheme. In particular, we highlight major differences from the signal loss computation used in Ali et al. (2015), which previously underestimated losses.

3.1.1. Signal Loss Methodology

We simulate an EoR signal by creating a random Gaussian signal (with a default variance of 1) with the same shape as our data, and we fringe-rate filter this signal twice using the optimal filter. The first filter transforms the white noise into a signal that's attached to the sky (i.e. what our instrument observes). The second filter represents the fringe-rate filtering step in our data analysis pipeline. This mock EoR signal is injected on top of fringe-rate filtered PAPER-64 data at a range of amplitude levels, ranging from well-below the data level, to well-above.

Suppose that \mathbf{e} is the injected EoR (at some amplitude level), and \mathbf{x} is our data vector. We define \mathbf{r} to be the data plus the EoR signal:

$$\mathbf{r} = \mathbf{x} + \mathbf{e}. \quad (8)$$

We are interested in quantifying how much \mathbf{e} is lost after weighting \mathbf{r} and applying OQE formalism. We investigate this by comparing two quantities we define as the input power spectrum and output power spectrum: P_{in} and P_{out} . P_{in} represents the unweighted power spectrum of only \mathbf{e} , our simulated EoR signal. P_{out} is the weighted power spectrum of \mathbf{e} that would result from our pipeline if the signal was mixed with our data. Comparing the two quantities yields insight into how much of \mathbf{e} is lost due to our choice of weighting. Ignoring normalization factors, these two quantities can be written as:

$$P_{in,\alpha} \propto \mathbf{e}^\dagger \mathbf{I} \mathbf{Q}_\alpha \mathbf{I} \mathbf{e} \quad (9)$$

and

$$P_{out,\alpha} \equiv \hat{\mathbf{p}}_{e,\alpha} = \hat{\mathbf{p}}_{r,\alpha} - \hat{\mathbf{p}}_{x,\alpha} \\ \propto \mathbf{r}^\dagger \mathbf{R}_r \mathbf{Q}_\alpha \mathbf{R}_r \mathbf{r} - \mathbf{x}^\dagger \mathbf{R}_x \mathbf{Q}_\alpha \mathbf{R}_x \mathbf{x}. \quad (10)$$

It is noted that the output power spectrum is comprised of two terms: the weighted power spectrum associated with \mathbf{r} , and that of data \mathbf{x} alone.

One may wonder why P_{out} cannot be computed simply as the weighted power spectrum of \mathbf{e} alone, namely $P_{out,\alpha} \propto \mathbf{e}^\dagger \mathbf{R}_e \mathbf{Q}_\alpha \mathbf{R}_e \mathbf{e}$. Expanding Equation 10, we see that P_{out} is comprised of multiple terms:

$$P_{out,\alpha} \propto (\mathbf{x} + \mathbf{e})^\dagger \mathbf{R}_r \mathbf{Q}_\alpha \mathbf{R}_r (\mathbf{x} + \mathbf{e}) - \mathbf{x}^\dagger \mathbf{R}_x \mathbf{Q}_\alpha \mathbf{R}_x \mathbf{x} \\ \propto \mathbf{x}^\dagger \mathbf{R}_r \mathbf{Q}_\alpha \mathbf{R}_r \mathbf{x} + \mathbf{e}^\dagger \mathbf{R}_r \mathbf{Q}_\alpha \mathbf{R}_r \mathbf{e} + \mathbf{x}^\dagger \mathbf{R}_r \mathbf{Q}_\alpha \mathbf{R}_r \mathbf{e} \\ + \mathbf{e}^\dagger \mathbf{R}_r \mathbf{Q}_\alpha \mathbf{R}_r \mathbf{x} - \mathbf{x}^\dagger \mathbf{R}_x \mathbf{Q}_\alpha \mathbf{R}_x \mathbf{x}. \quad (11)$$

Taking the case of very large \mathbf{e} , so that $\mathbf{R}_r \sim \mathbf{R}_e$ and any terms involving only \mathbf{x} are small, yields:

$$\begin{aligned} P_{out,\alpha,e \gg x} &\propto e^\dagger \mathbf{R}_e \mathbf{Q}_\alpha \mathbf{R}_e e + x^\dagger \mathbf{R}_e \mathbf{Q}_\alpha \mathbf{R}_e e \\ &+ e^\dagger \mathbf{R}_e \mathbf{Q}_\alpha \mathbf{R}_e x. \end{aligned} \quad (12)$$

We see that our naive expression for P_{out} is the first term in Equation 12, but there are also two additional terms. An initial assumption would be that the cross-terms that involve both e and x should be zero, since foregrounds and the cosmological signal are physically unrelated. In Ali et al. (2015), this assumption is used, and the first term in Equation 12 is directly compared to the input (unweighted) power spectrum to compute signal loss.

However, deeper investigations of these terms reveal that they contain non-negligible power. In fact, foreground modes and signal modes are anti-correlated on average, as derived and explained in Switzer et al. (2015). The spurious correlation of the two means that we cannot compute signal loss using a signal-only simulation, which would yield greater values for P_{out} and thereby underestimate signal loss. Therefore, in our revised signal loss computation we use the full quantity for P_{out} as defined in Equation 10, which subtracts the weighted power spectrum of the data from the weighted power spectrum of data plus EoR.

For the unweighted case ($\mathbf{R} \equiv \mathbf{I}$), we expect P_{out} and P_{in} to be equal, and hence the ratio $P_{in}/P_{out} = 1$. For the weighted case, this may not be true due to signal loss. To quantify the loss, we look at the ratio P_{in}/P_{out} as the amplitude level of the injected signal e is increased. In the next section, we describe our signal loss transfer function (P_{in} vs. P_{out}) for PAPER-64 data and detail the computation used to translate our power spectrum result into one viewed through a signal loss lens.

3.1.2. Signal Loss in Practice

We now shift our attention towards computing signal loss for our fringe-rate filtered PAPER-64 dataset. To recap, we use 30 m East/West baselines of PAPER-64. Our dataset spans 8 hours of LST (.5-8.6 hrs), includes a total of 51 baselines, and is fringe-rate filtered using an optimal fringe-rate filter. We have two datasets (even days and odd days), and only cross-multiply data from different days and different baselines. We are interested in 21 frequency channels (channels 95-115), which yields a power spectrum for a redshift of $z = 8.4$.

While our methodology outlined below is robust to any weighting scheme, here we demonstrate the computation using inverse covariance weighting ($\mathbf{R} = \mathbf{C}^{-1}$), which leads to substantial loss. With this weighting, our expressions for P_{in} and P_{out} become:

$$P_{in,\alpha} \propto e^\dagger \mathbf{I} \mathbf{Q}_\alpha \mathbf{I} e \quad (13)$$

$$P_{out,\alpha} \propto \mathbf{r}^\dagger \mathbf{C}_r^{-1} \mathbf{Q}_\alpha \mathbf{C}_r^{-1} \mathbf{r} - \mathbf{x}^\dagger \mathbf{C}_x^{-1} \mathbf{Q}_\alpha \mathbf{C}_x^{-1} \mathbf{x} \quad (14)$$

Using the input and output power spectrum estimates for a range of EoR amplitudes, we next describe two methods to determine signal loss associated with the

over-fitting of noise from inverse covariance weighting of PAPER-64 data. [CC: describing both methods may be overkill?]

The first method is very straightforward. Post-bootstrapping, our final power spectrum estimates are 1-dimensional and only a function of k . For each k , we simply look at P_{out} as a function of P_{in} , as shown in Figure ???. The shape of this function can be explained as follows. At small injection levels (small e), P_{out} and P_{in} are equal, and there is no signal loss. As the amplitude of EoR increases, we then move into a regime where the final output power spectrum is lower than the unweighted input one. This is dangerous, because without correcting for this effect one might be led to underestimate the EoR signal. The peculiar tail at very low injection levels (where $P_{out} > P_{in}$) is not a physical feature, but rather illustrates that there is some non-negligible power in the cross-terms of Equation 11 when e is very small.

For this method, we interpolate the signal loss factor (per k), computed as P_{in}/P_{out} , at a P_{out} value equal to the 2σ power spectrum upper limit of our data. In other words, we look at $\hat{\mathbf{p}}_\alpha \propto \mathbf{x}^\dagger \mathbf{C}_x^{-1} \mathbf{Q}_\alpha \mathbf{C}_x^{-1} \mathbf{x}$, compute its 2σ upper limit (mean over bootstraps + $2\times$ standard deviation over bootstraps), and interpolate the value of P_{in}/P_{out} at this value. We therefore end up with one signal loss correction factor per k .

Our second method for estimating signal loss is similar to the first, but more comprehensive in a statistical sense. Instead of looking at input and output power spectra after bootstrapping, we now look at their values for every bootstrap in order to get a sense of their distributions. Figure ?? plots P_{in} vs. P_{out} for 20 bootstraps, and as expected, the function now has a spread in the width-direction in comparison to what was plotted in Figure ??, but otherwise shows a familiar trend. Similarly, our weighted power spectra also has a defined spread due to bootstrapping.

Using these two distributions (P_{in}/P_{out} and $\hat{\mathbf{p}}$), we can create bins along the $\hat{\mathbf{p}}$ axis to yield histograms of signal loss factors for each bin. We similarly sort the values of $\hat{\mathbf{p}}$ into the same bins, and multiply the probability of $\hat{\mathbf{p}}$ per bin (the number of values falling into that bin, divided by the total) with the signal loss factors in that bin, essentially computing a weighted average across all bins to obtain a final signal loss factor per k . The signal loss results for this method are very similar to the previous method. For future power spectrum results, we choose to use the second method because it computes final signal loss values using our full distributions of measurements.

[CC: This next paragraph needs to be worked on. What I'm going for is a plot that 'proves' our signal loss method is correct, but it's hard to show since the combination of inverse covariance weighting + signal loss correction produces results that are HIGHER than the unweighted level. The plot shown below uses PS points in-

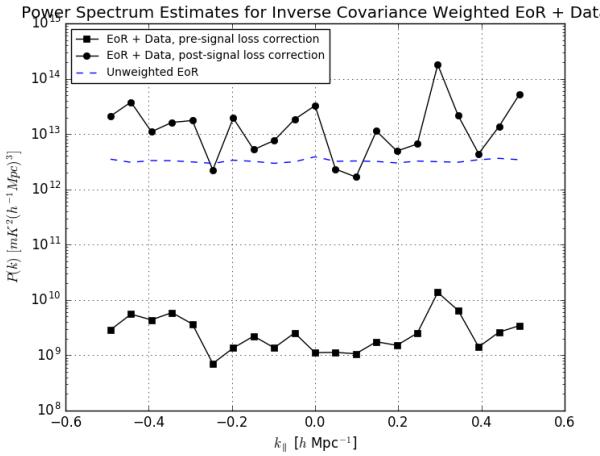


Figure 15. Power spectrum estimates of the combined EoR + PAPER-64 dataset, pre (squares) and post (circles) signal loss correction. The dashed blue line is the unweighted power spectrum estimate of EoR alone.

stead of error bars, which brings them into closer agreement...] To verify our signal loss computations, we compare the unweighted power spectrum estimate of EoR ('true' EoR) with the signal loss corrected weighted power spectrum estimate of EoR + data in a regime where the injected EoR is very large (since there is minimal signal loss for small EoR). We compare these two quantities because we must ensure that our weighted combined dataset, after signal loss correction, does not lie below the level of EoR. From Figure 15, we see over 3 orders of magnitude of signal loss associated with the inverse covariance weighting of fringe-rate filtered PAPER-64 data. The power spectrum, post-correction, lies just above the EoR-only power spectrum. However, it exhibits a different shape due to [CC: ??].

One thing to note is that for both methods, we have been careful to validate that the computations yield no signal loss (signal loss factors of 1) for the unweighted power spectrum case, as is expected. This is important in confirming that signal loss is a direct result of the choice for \mathbf{C} . Similarly, we can compute signal loss for intermediate cases as well, such as a regularized \mathbf{C} which exhibits some lossiness, but not as much as inverse covariance weighting.

3.1.3. Data Weighting

With our signal loss formalism established, we now have the capability of experimenting with different weighting options for \mathbf{R} . Our goal here is to choose a weighting method that successfully down-weights foregrounds and systematics in our data without generating large amounts of signal loss. We have found that the balance between the two is a delicate one and requires a careful understanding and altering of covariance matrices.

Using full inverse covariance weighting, our power spectrum results before and after signal loss correction is shown in Figure 16. Prior to signal loss correction, it is

clear that the power spectrum is unfeasible because it is well below the theoretical noise level prediction. Post-correction, the power spectrum values blow up much higher than both the theory and unweighted power spectrum.

Following our toy model investigation in Section 2.1, we investigate the shape of the eigenspectrum of \mathbf{C} for a typical baseline used in the analysis. Figure 17 shows this spectrum for baseline (1,4). The spectrum is very steep, spanning 4 orders of magnitude, implying that a couple modes are weighted much more than the rest. In fact, this behavior is much more extreme than we saw in the toy model, whose eigenspectrum (for the 'fringe-rate filtered' case) only started to vividly steepen for the last few eigenmodes. The consequence of assigning individual weights to many more modes than are present in the data is significant — to the point of resulting in power spectrum values that are higher than if no weighting scheme was used at all.

The full inverse covariance treatment of our data is suboptimal to the unweighted case, meaning that choosing such a weighting does not help suppress systematics as we intended. Now we would like to find a weighting method that does successfully down-weight contaminants in our data and make some improvement over the unweighted power spectrum. There are many choices for determining the covariance matrix \mathbf{C} , as described in Section 2.1, but here we will illustrate [CC: ?] promising ones as applied to PAPER-64.

[CC: Should I just show the identity multiplication scheme that seems to be the best? Show lots of options? Mention different options but not show the results?]

3.2. Case Study: Error Estimation

In this section we discuss the ways in which we estimate errors for PAPER-64 power spectra. We first walk through a derivation for a theoretical error estimation based on observational parameters. Although a theoretical model often differs from true errors as explained in Section 2.2, it is helpful to understand the ideal case and the factors that affect its sensitivity. We also highlight major changes in our calculation from Ali et al. (2015), which in total decrease the estimated power spectrum sensitivity of PAPER-64 by a factor of [CC: fill in when ready]. Finally, we build on the lessons learned about bootstrapping in Section 2.2 to revise our bootstrapping method as applied to PAPER-64 data in order to compute accurate errors from the data itself.

3.2.1. Theoretical Error Estimation

Theoretical errors, computed from the instrumental temperature and observational parameters, provide a useful cross-check for bootstrapped errors and is a helpful predictor of the maximum sensitivity that can be achieved by a particular observation and analysis. Here we walk through a detailed computation of a theoretical noise estimation as applied to PAPER-64 observations and highlight major changes from Ali et al. (2015). We

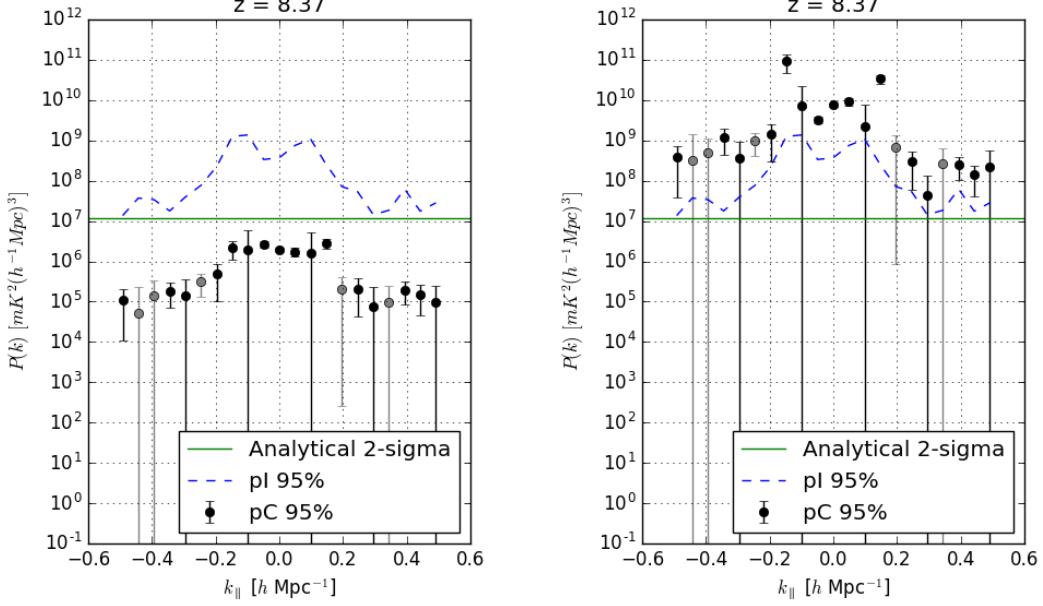


Figure 16. Full inverse covariance weighted power spectrum of PAPER-64 data (black points, with 2σ error bars) before signal loss correction (left) and after (right). The dashed blue line is the unweighted power spectrum (2σ upper limit). The solid green line is the theoretical noise level prediction based on observational parameters.

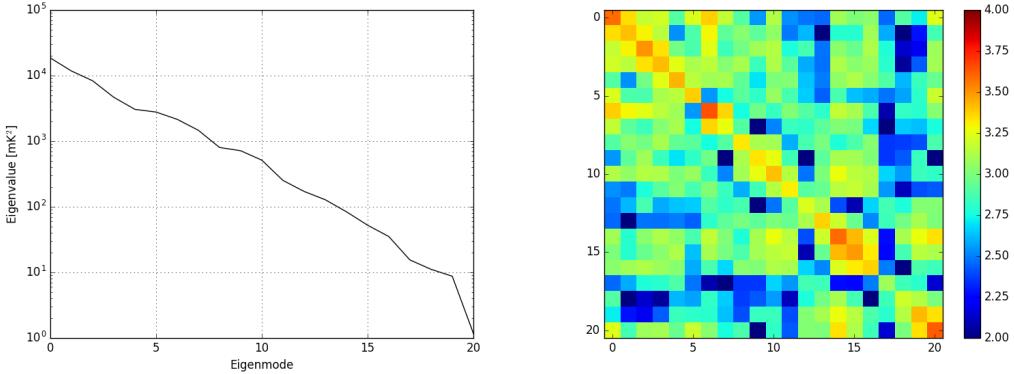


Figure 17. Eigenspectrum for \mathbf{C} for baseline (1,4) for the 21 channels of interest (left) and covariance matrix \mathbf{C} for the same baseline (right).

first give a general formula, and then go into detail about each factor.

The sensitivity prediction for a power spectral analysis of interferometric 21 cm data, in temperature-units, is:

$$p(k) = \frac{X^2 Y \Omega_{eff} T_{sys}^2}{\sqrt{2N_{lst}s N_{seps}} t_{int} N_{days} N_{bls} N_{pol}} \quad (15)$$

- $X^2 Y$: Conversion factors from observing coordinates (angles on the sky) to cosmological coordinates (co-moving distances). For $z = 8.4$, $X^2 Y = 5 \times 10^{11} h^{-3} Mpc^3 str^{-1} GHz^{-1}$.
- Ω_{eff} : The primary beam angular size. The effective beam area changes with the application of a fringe-rate filter, since parts of the beam are up-

weighted and down-weighted. Using numbers from Table 1 in [Parsons et al. \(2016\)](#), $\Omega_{eff} = 0.74^2 / 0.24$ for an optimal fringe-rate filter.

- T_{sys} : The system temperature is set by:

$$T_{sys} = 180 \left(\frac{\nu}{0.18} \right)^{-2.55} + T_{rcvr}, \quad (16)$$

where ν are frequencies in GHz. We use a receiver temperature of 200 K, yielding $T_{sys} = 487$ at 150 MHz. This is lower than in [Ali et al. \(2015\)](#) because [CC: why?].

- $\sqrt{2}$: This factor in the denominator of the sensitivity equation comes from only using the real part of power spectrum estimates when plotting

and quoting upper limits (imaginary component is discarded).

- N_{lsts} : The number of LST hours that go into a power spectrum estimation. The sensitivity scales as the square root because we integrate incoherently over time. For PAPER-64, $N_{lsts} = 8$ hours.
- N_{seps} : The number of baseline separation types averaged incoherently in a final power spectrum estimate. For the analysis in this paper, we only use one type of baseline, hence $N_{seps} = 1$.
- t_{int} : The integration time of the data. It is crucial to adapt this number if filtering is applied along the time axis (i.e. a fringe-rate filter). We compute the effective integration time of our fringe-rate filtered data by scaling the original integration time using the following:

$$t_{frf} = t_{int} \frac{\int 1 df}{\int w^2(f) df}, \quad (17)$$

where $t_{int} = 43$ seconds, t_{frf} is the fringe-rate filtered integration time, w is the fringe-rate profile, and the integral is taken over all fringe-rates. In other words, we calculate the effective integration time by computing how our beam is altered in fringe-rate space. For PAPER-64, this number is $t_{int} = 3857$ s.

- N_{days} : The total number of days of data analyzed. In [Ali et al. \(2015\)](#), this number was set to 135. However, because we divide our data in half (to form ‘even’ and ‘odd’ datasets), this number should be reduced by a factor of 2. Additionally, because our *LST* coverage is not 100% complete (it doesn’t overlap for every single day), we compute a realistic value of N_{days} as the RMS of all the daily counts in one dataset (‘even’, for example). For PAPER-64, our revised estimate of N_{days} is ~ 34 days.
- N_{bls} : The number of baselines contributing to the sensitivity of a power spectrum estimate. In [Ali et al. \(2015\)](#), this number was the total number of 30 m East/West baselines used in the analysis. However, using the total number of baselines neglects the fact that we divided baselines into 5 groups before cross-multiplying data. Our revised estimate for the parameter is $N_{bls} = \frac{N_{bls}}{N_{gps}} \sqrt{N_{gps}^2 - N_{gps}}$, where $N_{gps} = 5$. This expression arises because N_{bls} must be scaled by the total number of cross-multiplications going into the power spectrum estimate. For 5 groups, there are 25 total cross-products, minus 5 cross-products since we avoid multiplying the same group with itself. For our PAPER-64 analysis with only one baseline separation type, this becomes $N_{bls} \sim 46$ baselines.

- N_{pol} : The number of polarizations averaged together. For the case of Stokes I, $N_{pol} = 2$.

[Equation 15](#) represents the most sensitive estimate for $\hat{\mathbf{p}}(k)$ given our observing parameters. An additional factor of $\sqrt{2}$ is gained in sensitivity when folding our power spectra into $\Delta^2(k)$, due to averaging together positive and negative k ’s.

To verify our thermal noise prediction, we form power spectra estimates using a pure noise simulation. We create Gaussian random noise and use T_{sys} in order to best represent the thermal noise level of PAPER-64 data. We convert T_{sys} into a variance statistic using:

$$T_{rms} = \frac{T_{sys}}{\sqrt{\Delta\nu\Delta t N_{days} N_{pol}}}, \quad (18)$$

where $\Delta\nu$ is channel spacing, Δt is integration time, N_{days} is the number of daily counts for a particular time and frequency that went into our LST binned set, and N_{pol} is the number of polarizations (2 for Stokes I). This rms temperature sets the variance of the Gaussian random noise.

We fringe-rate filter the noise in the same way as PAPER-64 data, using an optimal fringe-rate filter. Power spectrum results for the noise simulation are shown in [Figure 18](#), where the black and grey points represent positive and negative power spectrum values, respectively (with 2σ error bars and $\hat{\mathbf{C}} = \hat{\mathbf{C}} * \mathbf{I}$), the dashed blue line represents the unweighted power spectrum, and the solid green line denotes our 2σ theoretical noise prediction as calculated by [Equation 15](#). All three are in agreement, validating our analytical thermal noise calculation.

3.2.2. Bootstrapping

We bootstrap PAPER-64 power spectra in order to determine confidence intervals for our results. In this section, we highlight two major changes in the way we estimate errors since [Ali et al. \(2015\)](#), using the lessons we’ve learned about sampling with replacement and bootstrapping independent samples.

As discussed in [Section 2.2](#), bootstrapping is only a valid way of estimating errors if a dataset is comprised of independent samples. The PAPER-64 pipeline outputs N bootstraps (over baselines), each a 2-dimensional power spectrum that is a function of k and time. In [Ali et al. \(2015\)](#), a second round of bootstrapping occurs over the baseline bootstrap and time axes simultaneously. Random values are sampled with replacement along both axes, drawing as many values as there are number of bootstraps and times. Final power spectrum limits are then computed by taking the mean and standard deviation over this second bootstrap axis (which is typically large, on the order of $N_{boot} \sim 400$).

Because our dataset is fringe-rate filtered, we have substantially increased our integration time, resulting in fewer independent samples (from ~ 700 samples to 8 samples). A random draw of 700 samples from this

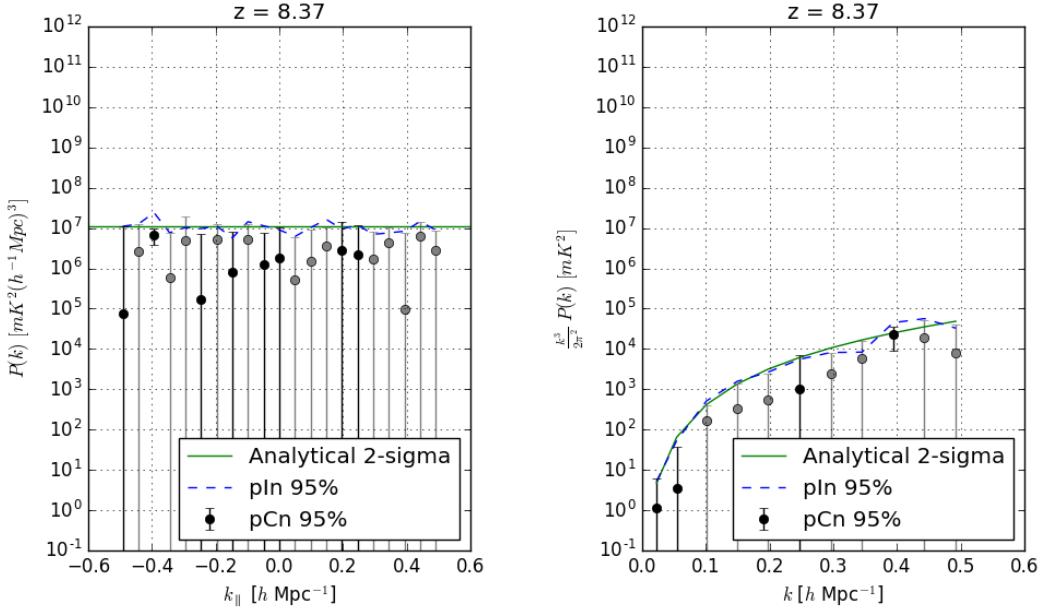


Figure 18. Power spectrum estimate for a noise simulation that mimics the noise level of PAPER-64 data. The weighted power spectrum points and 2σ errors are shown in black and grey (positive and negative values), where we regularize $\hat{\mathbf{C}}$ by setting $\hat{\mathbf{C}} = \hat{\mathbf{C}} * \mathbf{I}$ to minimize signal loss. The dashed blue line is the unweighted power spectrum (also 2σ upper limit). The solid green line is the theoretical noise level prediction as calculated by Equation 15. All three estimates agree.

dataset therefore has many repeated samples, and the variance between hundreds (N_{boot}) of these random samples is smaller than the true underlying variance of the data. Another way of describing this underestimation is that there are fewer ways to randomly sample data with 8 independent modes versus 700, so bootstrapping a fringe-rate filtered dataset yields a narrower distribution of values from which we calculate its standard deviation.

To prevent error underestimation, we simply take the average along the time axis in our revised pipeline rather than bootstrapping. Hence, our final power spectrum errors are calculated from the standard deviation over the baseline bootstrapping axis only, which is still a valid axis since each baseline makes an independent measurement of the sky.

The second aspect of bootstrapping discussed in Section 2.2 is the overestimation of errors which arises from repeated values in a randomly sampled bootstrap. In order to maximize sensitivity, we have revised the method by which we sample baselines when bootstrapping to ensure mostly independent samples.

First, we speed things up in our pipeline by separating our total number of baselines into 5 groups (yielding 10 baselines per group), where there are no repeated baselines within or between groups. In Ali et al. (2015), each group is then sampled with replacement to create a new group of the same size, which can have repeated baselines inside it. In doing so, we are sacrificing some of our sensitivity since there ends up being 3-4 repeated baselines per group (Figure 10 shows a fraction of independent samples of $\sim 55\%$ for 10 total independent

samples). In order to maximize our sensitivity but still apply random sampling for use in error estimation, we instead form new groups using all independent baselines except the very last one, which we fill randomly. We use this method for every bootstrap, creating new groups and sampling the last spot randomly each time.

Power spectrum estimates for PAPER-64 fringe-rate filtered data are shown in Figure 19. The estimates use a weighting matrix of \mathbf{C}_{eff}^{-1} , where $\mathbf{C}_{eff} = \hat{\mathbf{C}} * \mathbf{I}$, in order to minimize signal loss. We show upper limits (2σ error bars) for three different results. Our revised error estimation method is in black, where we only bootstrap along the baseline axis and we maximize sensitivity by random sampling only the last baseline in each group. The red curve uses a similar method except all baselines are randomly sampled, leading to a higher power spectrum estimate because some baselines are repeated per group. In blue, we bootstrap along both the baseline and time axes, drawing as many samples as there are in the data. We under-estimate errors in doing so because we are drawing many more samples than independent ones.

3.3. Case Study: Bias

In Section 2.3 we highlighted some common sources of bias that can show up as a power spectrum detection and imitate EoR. We discussed the importance of using jackknife and null tests for instilling confidence in an EoR detection, as well as identifying other sources of biases. Here we demonstrate methods used by PAPER-64 to mitigate foreground and noise bias and we perform jackknife and null tests in order to characterize the sta-

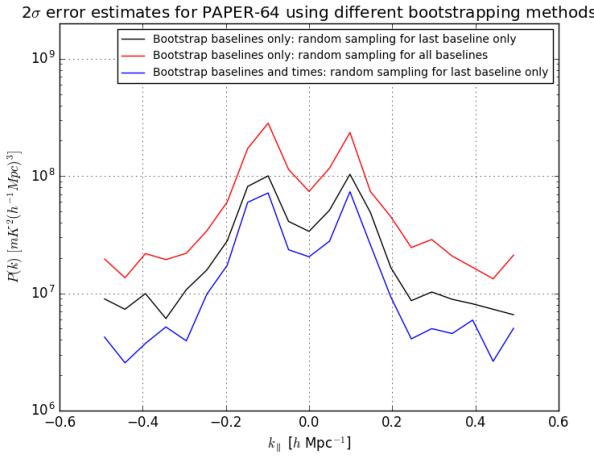


Figure 19. 2σ upper limit power spectrum estimates using PAPER-64 data and three different bootstrapping methods. The data is fringe-rate filtered and a weighting matrix of \mathbf{C}_{eff}^{-1} is used, where $\hat{\mathbf{C}}_{eff} = \mathbf{C} * \mathbf{I}$, in order to minimize signal loss. Three different bootstrapping methods are shown: the black and red use methods where bootstrapping occurs on the baseline axis only (not the time axis), and they differ by how the baselines are randomly sampled. It is evident that sensitivity is maximized when baselines are sampled in a way that ensures mostly independent samples (only the last baseline spot is sampled randomly). The blue result bootstraps over both the baseline and time axes, and illustrates how errors can be under-estimated if sampling more values than independent ones (fringe-rate filtering reduces the number of independent samples).

bility and implications of our results.

3.3.1. Mitigating Bias

We briefly discuss one way in which we mitigate foreground leakage in a power spectrum estimate, and two in which we suppress noise biases. These methods are not novel to this analysis but here we frame them in the context of minimizing false (non-EoR) detections.

Tailoring window functions is one way to suppress foreground biases. As alluded to in Section 2.1, we have a choice for the normalization matrix \mathbf{M} in Equation 2. For the analysis of PAPER-64 data, we compute \mathbf{M} using the Fisher matrix \mathbf{F} , defined as:

$$\mathbf{F}_{\alpha\beta} = \frac{1}{2} \text{tr}[\mathbf{RQ}_\alpha \mathbf{RQ}_\beta] \quad (19)$$

where \mathbf{R} is the data-weighting matrix and α and β are wavebands in k_\parallel . We take the Cholesky decomposition of \mathbf{F} , decomposing it into two lower triangular matrices:

$$\mathbf{F} = \mathbf{LL}^\dagger. \quad (20)$$

Next, we construct \mathbf{M} :

$$\mathbf{M} = \mathbf{DL}^{-1} \quad (21)$$

where \mathbf{D} is a diagonal matrix. In doing so, our window function, defined as $\mathbf{W} = \mathbf{MF}$, becomes:

$$\mathbf{W} = \mathbf{DL}^\dagger. \quad (22)$$

Because of the nature of the lower triangular matrix, this window function has the property of preventing the leakage of foreground power from low k to high k modes. Specifically, we order the elements in \mathbf{F} in such a way so that power can leak from high k modes to low k modes, but not vice versa. Since most foreground power shows up at low k 's, this method ensures a window function that retains clean, noise-dominated measurements while minimizing the contamination of foreground bias.

In addition to mitigating foreground bias at high k 's, two other sources of bias that we actively suppress in the PAPER-64 analysis are noise bias associated with the squaring of thermal noise and noise bias from crosstalk. In order to avoid the former, we filter out certain cross-multiplications when forming \hat{q} in Equation 1. Namely, the PAPER-64 dataset is divided into two halves: even julian dates and odd julian dates. Our data vectors are then $\mathbf{x}_{even,1}$ for the ‘even’ dataset and baseline group 1, $\mathbf{x}_{odd,1}$ for the ‘odd’ dataset and baseline group 1, etc. We only form \hat{q} when the two copies of \mathbf{x} come from different groups and baselines, never multiplying ‘even’ with ‘even’, for example, in order to prevent the squaring of the same thermal noise.

To mitigate crosstalk bias, which appears as a static bias in time, we apply a fringe-rate filter that suppresses fringe-rates of zero. Figure 14 shows that the filter response is zero for such static signals. The effect of filtering out zero fringe-rates on power spectrum results is shown in Ali et al. (2015). Most notably, power spectrum detections exist at all k 's without crosstalk removal and these are detections that, depending on the power spectrum level, could be mistaken for EoR.

3.3.2. Jack-Knife Tests

[CC: Describe some jackknife tests done on PAPER-64 data, such as random signs to baselines and even/odd summing and differencing]
[CC: Need to explain positive bias still... not sure where it comes from yet!]

4. CONCLUSION

5. ACKNOWLEDGEMENTS

[CC: NSF Graduate Research Fellowship Program (GRFP) Fellowship] [CC: UC Berkeley Chancellor’s Fellowship]

APPENDIX

A. A TOY MODEL FOR SIGNAL LOSS

In this Appendix, we examine a toy model for signal loss. Our goal is to build intuition by deriving an analytic formula for power spectrum signal loss. We will also show that in general, signal loss appears as a multiplicative bias on one's power spectrum estimate.

The minimum-variance quadratic estimator \hat{p}_α for the α th bandpower of the power spectrum is given by [ACL: We need to decide whether the \mathbf{Q} matrices have their Greek indices as upper or lower indices.] [ACL: Also need to state in a footnote that for this section only we'll assume without loss of generality that the data are real]

$$\hat{p}_\alpha = \frac{1}{2F_{\alpha\alpha}} \mathbf{x}^t \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{x}, \quad (\text{A1})$$

where

$$F_{\alpha\alpha} \equiv \frac{1}{2} \text{tr}(\mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{Q}^\alpha) \quad (\text{A2})$$

is the α th diagonal element of the Fisher matrix. In our case, however, we do not have *a priori* knowledge of the covariance matrix. Thus, we replace \mathbf{C} with $\hat{\mathbf{C}}$, its data-derived approximation. Our estimator then becomes

$$\hat{p}_\alpha^{\text{loss}} = \frac{1}{2F_{\alpha\alpha}} \mathbf{x}^t \hat{\mathbf{C}}^{-1} \mathbf{Q}^\alpha \hat{\mathbf{C}}^{-1} \mathbf{x}, \quad (\text{A3})$$

where

$$\hat{F}_{\alpha\alpha} \equiv \frac{1}{2} \text{tr}(\hat{\mathbf{C}}^{-1} \mathbf{Q}^\alpha \hat{\mathbf{C}}^{-1} \mathbf{Q}^\alpha), \quad (\text{A4})$$

with the label “loss” to foreshadow the fact that this will be an estimator with signal loss (i.e., a multiplicative bias of less than unity). We will now provide an explicit demonstration of this by modeling the estimated covariance as

$$\hat{\mathbf{C}} = (1 - \eta) \mathbf{C} + \eta \mathbf{x} \mathbf{x}^t, \quad (\text{A5})$$

where η is parameter quantifying our success at estimating the true covariance matrix. If $\eta = 0$, our covariance estimate has perfectly modeled the true covariance and $\hat{\mathbf{C}} = \mathbf{C}$. On the other hand, if $\eta = 1$, then our covariance estimate is based purely on the one realization of the covariance that is our actual data, and we would expect a high level of overfitting and signal loss.

Our strategy for computing the signal loss will be to insert Eq. (A5) [ACL: Make sure we're consistent w.r.t. “Equation” vs. “Eq.”] into Eq. (A3) and to express the resulting estimator $\hat{p}_\alpha^{\text{loss}}$ in terms of \hat{p}_α . We begin by expressing $\hat{\mathbf{C}}^{-1}$ in terms of \mathbf{C}^{-1} using the Woodbury identity so that

$$\hat{\mathbf{C}}^{-1} \mathbf{x} = \frac{\mathbf{C}^{-1} \mathbf{x}}{1 - \eta} \left[1 - \frac{\eta g}{1 + \eta(g-1)} \right], \quad (\text{A6})$$

where we have defined $g \equiv \mathbf{x}^t \mathbf{C}^{-1} \mathbf{x}$. Inserting this into our Fisher estimate we have

$$\hat{F}_{\alpha\alpha} = \frac{F_{\alpha\alpha}}{(1-\eta)^2} \left[1 - \frac{\eta g}{1 + \eta(g-1)} \frac{h_{\alpha\alpha}}{F_{\alpha\alpha}} + \frac{1}{2} \left(\frac{\eta g}{1 + \eta(g-1)} \right)^2 \frac{h_\alpha^2}{F_{\alpha\alpha}} \right], \quad (\text{A7})$$

where $h_\alpha \equiv \mathbf{x}^t \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{x}$ and $h_{\alpha\alpha} \equiv \mathbf{x}^t \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{x}$. Note that g , h_α , and $h_{\alpha\alpha}$ are all random variables, since they depend on \mathbf{x} . Inserting these expressions into our estimator gives

$$\begin{aligned} \hat{p}_\alpha^{\text{loss}} &= \frac{1}{2} \frac{h_\alpha}{F_{\alpha\alpha}} \left[1 - \frac{\eta g}{1 + \eta(g-1)} \right]^2 \left[1 - \frac{\eta g}{1 + \eta(g-1)} \frac{h_{\alpha\alpha}}{F_{\alpha\alpha}} + \frac{1}{2} \left(\frac{\eta g}{1 + \eta(g-1)} \right)^2 \frac{h_\alpha^2}{F_{\alpha\alpha}} \right]^{-1} \\ &= \frac{1}{2} \frac{h_\alpha}{F_{\alpha\alpha}} (1 - 2\beta + \beta^2) \sum_{n=0}^{\infty} \sum_{k=0}^n \binom{n}{k} \left(\frac{\beta h_{\alpha\alpha}}{F_{\alpha\alpha}} \right)^k \left(-\frac{\beta^2 h_\alpha^2}{F_{\alpha\alpha}} \right)^{n-k}, \end{aligned} \quad (\text{A8})$$

where $\beta \equiv \eta g / [1 + \eta(g-1)]$. To proceed, we take the ensemble average of $\hat{p}_\alpha^{\text{loss}}$ to understand its average statistical properties. Inspecting Eq. (A8), we see that its expectation value is a linear sum of terms of the form $\langle \beta^a h_\alpha^b h_{\alpha\alpha}^c \rangle$, where a , b , and c are integers. Expanding β in powers of g , we have

$$\langle \beta^a h_\alpha^b h_{\alpha\alpha}^c \rangle = \sum_{m_1=1}^{\infty} \cdots \sum_{m_a=1}^{\infty} \frac{(-\eta)^{m_1+\dots+m_a}}{(1-\eta)^{m_1+\dots+m_a}} \langle g^{m_1+\dots+m_a} h_\alpha^b h_{\alpha\alpha}^c \rangle. \quad (\text{A9})$$

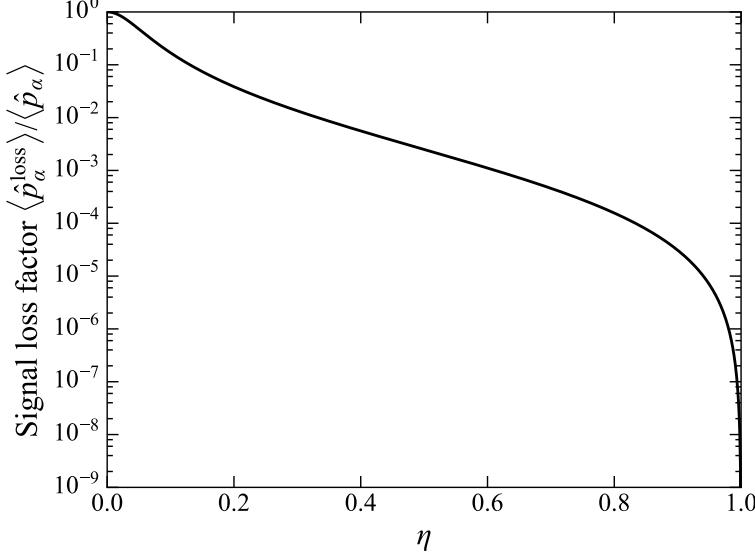


Figure A1. Power spectrum signal loss for $N = 20$ for a toy model where $\hat{\mathbf{C}} = (1 - \eta)\mathbf{C} + \eta \mathbf{x} \mathbf{x}^t$.

Writing out the ensemble averaged quantity on the right hand side and defining $M \equiv m_1 + \dots + m_a$ yields

$$\begin{aligned} \langle g^{m_1+\dots+m_a} h_\alpha^b h_{\alpha\alpha}^c \rangle &= \langle (\mathbf{x}^t \mathbf{C}^{-1} \mathbf{x})^M (\mathbf{x}^t \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{x})^b (\mathbf{x}^t \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{x})^c \rangle \\ &= \sum_{i_1}^N \sum_{j_1}^N \dots \sum_{k_1}^N \sum_{l_1}^N \dots \sum_{p_1}^N \sum_{q_1}^N \dots \dots (\mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1})_{p_1 q_1} \dots (\mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1})_{p_c q_c} \\ &\quad \times (\mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1})_{k_1 l_1} \dots (\mathbf{C}^{-1} \mathbf{Q}^\alpha \mathbf{C}^{-1})_{k_b l_b} \mathbf{C}_{i_1 j_1}^{-1} \dots \mathbf{C}_{i_M j_M}^{-1} \langle x_{i_1} x_{j_1} \dots x_{k_1} x_{l_1} \dots x_{p_1} x_{q_1} \dots \rangle, \end{aligned} \quad (\text{A10})$$

where N is the length of vector \mathbf{x} , i.e., the number of data points. To further simplify this expression, we assume that the data are Gaussian distributed. The expectation value on the right hand side can then be reduced to a sum of product of covariances, where each term in the sum is comprised of a particular way to partition the copies of x into pairs. However, in the limit of large N one particular partitioning dominates, and the result can be approximated as [ACL: Should probably explain this more fully]

$$\langle g^{m_1+\dots+m_a} h_\alpha^b h_{\alpha\alpha}^c \rangle \approx N^{m_1+\dots+m_a} [\text{tr}(\mathbf{C}^{-1} \mathbf{Q}^\alpha)]^b (2F_{\alpha\alpha})^c, \quad (\text{A11})$$

and re-inserting this into Eq. (A9) yields

$$\langle \beta^a h_\alpha^b h_{\alpha\alpha}^c \rangle \approx \left[\frac{\eta N}{1 + \eta(N-1)} \right]^a [\text{tr}(\mathbf{C}^{-1} \mathbf{Q}^\alpha)]^b (2F_{\alpha\alpha})^c. \quad (\text{A12})$$

With this result, taking the ensemble average of Eq. (A8) is relatively straightforward. One simply replaces every copy of β with $\beta_N \equiv \eta N / [1 + \eta(N-1)]$, every copy of h_α with $\text{tr}(\mathbf{C}^{-1} \mathbf{Q}^\alpha)$, and every copy of $h_{\alpha\alpha}$ with $2F_{\alpha\alpha}$. Performing these substitutions and recalling [ACL: make sure the reader will indeed be recalling at this point] that $\langle \hat{p}_\alpha \rangle = \langle h_\alpha \rangle / 2F_{\alpha\alpha}$, we obtain

$$\frac{\langle \hat{p}_\alpha^{\text{loss}} \rangle}{\langle \hat{p}_\alpha \rangle} = \frac{(1 - \beta_N)^2}{1 - 2\beta_N + \beta_N^2 [\text{tr}(\mathbf{C}^{-1} \mathbf{Q}^\alpha)]^2 / F_{\alpha\alpha}}. \quad (\text{A13})$$

This quantity is in general less than unity, and quantifies the signal loss relative to the ideal estimator \hat{p}_α where we know the true covariance \mathbf{C} . As an illustrative example, for cases where $\mathbf{C} \propto \mathbf{I}$, this reduces to

$$\left. \frac{\langle \hat{p}_\alpha^{\text{loss}} \rangle}{\langle \hat{p}_\alpha \rangle} \right|_{\mathbf{C} \propto \mathbf{I}} = \frac{(1 - \beta_N)^2}{(1 - \beta_N)^2 + \beta_N^2} = \frac{(1 - \eta)^2}{(1 - \eta)^2 + N^2 \eta^2}. \quad (\text{A14})$$

This is shown in Figure A1 for $N = 20$. As expected, the signal loss factor is unity for $\eta = 0$ (perfect estimation of \mathbf{C}) and goes to zero as η approaches 1.

In general, one's estimate of the covariance will not be expressible in the form given by Eq. (A5). However, the qualitative point here is that the signal loss can be expressed as a multiplicative factor. Even if $\hat{\mathbf{C}}$ is more complicated

than the form used for this toy model, the same mathematical technique of expressing the estimator as a series of higher-point functions, reducing these functions (assuming Gaussianity) to two-point functions, and re-summing any power series will show that the signal loss is a multiplicative factor. [ACL: I can express this last bit more eloquently, but it's late and I'm tired and need to eat dinner.]

REFERENCES

- Ali, Z. S., et al. 2015, ApJ, 809, 61
 Bond, J. R., Jaffe, A. H., & Knox, L. 1998, PhRvD, 57, 2117
 Bowman, J. D., & Rogers, A. E. E. 2010, Nature, 468, 796
 Burns, J. O., et al. 2012, Advances in Space Research, 49, 433
 DeBoer, D. R., et al. 2017, Publications of the Astronomical Society of the Pacific, 129, 045001
 Ellingson, S. W., Clarke, T. E., Cohen, A., Craig, J., Kassim, N. E., Pihlstrom, Y., Rickard, L. J., & Taylor, G. B. 2009, IEEE Proceedings, 97, 1421
 Greenhill, L. J., & Bernardi, G. 2012, ArXiv e-prints
 Jacobs, D. C., et al. 2015, ApJ, 801, 51
 Keating, G. K., Marrone, D. P., Bower, G. C., Leitch, E., Carlstrom, J. E., & DeBoer, D. R. 2016, The Astrophysical Journal, 830, 34
 Liu, A., Parsons, A. R., & Trott, C. M. 2014a, PhRvD, 90, 023018
 —. 2014b, PhRvD, 90, 023019
 Liu, A., & Tegmark, M. 2011, Phys. Rev. D, 83, 103006
 Paciga, G., et al. 2013, MNRAS
 Parsons, A. R., Liu, A., Ali, Z. S., & Cheng, C. 2016, ApJ, 820, 51
 Parsons, A. R., Pober, J. C., Aguirre, J. E., Carilli, C. L., Jacobs, D. C., & Moore, D. F. 2012, ApJ, 756, 165
 Parsons, A. R., et al. 2010, AJ, 139, 1468
 —. 2014, ApJ, 788, 106
 Patra, N., Subrahmanyam, R., Sethi, S., Udaya Shankar, N., & Raghunathan, A. 2015, ApJ, 801, 138
 Peterson, U.-L. P. X.-P. W. J. 2004, ArXiv Astrophysics e-prints
 Pober, J. C., et al. 2014, ApJ, 782, 66
 Sokolowski, M., et al. 2015, PASA, 32, e004
 Switzer, E. R., Chang, T.-C., Masui, K. W., Pen, U.-L., & Voytek, T. C. 2015, The Astrophysical Journal, 815, 51
 Tegmark, M. 1997, PhRvD, 55, 5895
 Thyagarajan, N., et al. 2013, ApJ, 776, 6
 Tingay, S. J., et al. 2013, PASA, 30, 7
 van Haarlem, M. P., et al. 2013, A&A, 556, A2
 Vedantham, H., Shankar, N. U., & Subrahmanyam, R. 2012, The Astrophysical Journal, 745, 176
 Voytek, T. C., Natarajan, A., Jáuregui García, J. M., Peterson, J. B., & López-Cruz, O. 2014, ApJL, 782, L9
 Wu, X. 2009, in Bulletin of the American Astronomical Society, Vol. 41, American Astronomical Society Meeting Abstracts #213, 474