Draft version July 11, 2017
Preprint typeset using LaTeX style AASTeX6 v. 1.0

# DATA ANALYSIS METHODS FOR THE DETECTION OF THE EPOCH OF REIONIZATION

Carina Cheng[1], et al.

[1]Astronomy 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. This important epoch of our cosmic dawn harbors a wealth of information regarding the environment during this transformative time, including insight into the nature of the first luminous sources and implications about cosmological parameters. 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).

## 1. INTRODUCTION

By about one billion years after the Big Bang, the very first stars and galaxies are thought to have ionized all the neutral hydrogen that dominated the baryonic matter content in the early 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. The 21 cm line therefore offers a window into the evolution of ionization, temperature, and density fluctuations on cosmic scales.

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 $23\sigma$ 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 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. In this paper, we discuss three themes essential to a 21 cm power spectrum analysis and how data analysis choices affect each. We approach these 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 into each one. Section 3 presents an overview of the PAPER-64 array and observations, highlighting key changes from Ali et al. (2015). Sections 4, 5, and 6 detail how the new PAPER-64 analysis quantifies signal loss, estimates error bars, and eliminates bias, respectively. We conclude in Section 7.

## 2. POWER SPECTRUM THEMES AND TECHNIQUES

We seek to measure the statistical 21 cm power spectrum. As such, we require robust methods that determine accurate confidence intervals and rigorous techniques to identify and suppress systematics. Additionally, because of the challenge in isolating a faint signal, we desire to maximize our sensitivity through filtering and weighting techniques.

There are many choices for a data analyst, such as how to optimally combine time-ordered measurements, how to best and most accurately estimate its variance, and how to weight data in a way that suppresses contaminated modes while not destroying an EoR signal. Common techniques such as averaging data, weighting, bootstrapping, and jack-knife testing each affect data in different ways and reveal different lessons.

There are tradeoffs between these power spectrum techniques which impact resulting limits. For example, an aggressive filtering method may succeed in eliminating interfering systematics but comes at the cost of losing the EoR signal. A chosen weighting scheme may maximize sensitivity but fail to suppress foregrounds.

In this paper, we focus on four power spectrum techniques and their effect on three overarching 21 cm power spectrum themes. We will give brief definitions now, and build intuition for each theme in the sections to follow.

Power Spectrum Themes

- **Signal Loss:** As explained in the next session, there are analysis techniques that can lead to the loss of an EoR signal. If not corrected for, it could lead to a false non-detection. Computing signal loss has subtle challenges but is a crucial component for confidence in any result.
- **Error Bar Estimation:** Errors on a 21 cm power spectrum result can make the difference between a detection and a noise-like measurement, which have two very different implications. Errors can be estimated in a variety of ways, and we will discuss a few of them.
- **Bias:** There are several possible sources of bias in a visibility measurement that can leak its way into a power spectrum. It is important to identity them in order to interpret results and develop techniques to suppress contamination.

Power Spectrum Techniques

- **Fringe-rate filtering:** The concept of a fringe-rate filter is similar to averaging data in time. We explain our filter in more detail in Section 3, but a broad description is that 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.
- **Weighting:** A dataset can be weighted to emphasize certain features and minimize others. One particular flavor of weighting is inverse covariance weighting, which weights a dataset by minimizing the covariance between frequency channels. This weighting has the effect of down-weighting correlated information (i.e. foregrounds) and up-weighting noise-like information (i.e. EoR).
- **Bootstrapping:** This 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.
- **Jack-Knife testing:** A resampling technique useful for estimating bias, jack-knives can be taken along different dimensions of a dataset to cross-validate results.

We will now discuss the three themes more in depth and discuss how certain power spectrum techniques affect each.

### 2.1. *Signal Loss*

Signal loss refers to the loss of a cosmological signal in a dataset. It can arise through a variety of ways in an analysis pipeline, such as fitting a polynomial during calibration or applying a delay-domain filter. Here we focus on signal loss associated with applying a weight matrix to data. Driven by sensitivity needs, we would like to use a weighting method that succeeds in down-weighting foregrounds. However, as we'll investigate, this can carry the risk of large amounts of signal loss.

Before we demonstrate how different weighting matrices affect signal loss, we first recap how weights are related to the power spectrum estimation technique of optimal quadratic estimators (OQE, Liu et al. 2014). A summary of the method is as follows.

We begin with our data vector $\mathbf{x}$, which contains our visibilities that have the shape (*times*, *frequencies*). We form the quantity $\hat{q}_\alpha$:

$$\hat{q}_\alpha = \frac{1}{2}\mathbf{x}^\dagger \mathbf{w}\mathbf{Q}_\alpha \mathbf{w}\mathbf{x} \tag{1}$$

where $\mathbf{w}$ is a weighting matrix (for example, inverse covariance weighting would set $\mathbf{w} = \mathbf{C}^{-1}$). The matrix $\mathbf{Q}$ is an operator that takes our frequency-domain visibilities and Fourier-transforms them into power spectrum space. The index $\alpha$ represents each $k_\parallel$ bin, where $k_\parallel$ is the Fourier-dual to frequency.

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

$$\hat{\mathbf{p}} = \mathbf{M}\hat{\mathbf{q}} \tag{2}$$

where $\hat{\mathbf{p}}$ is the normalized estimate of the true power spectrum $\mathbf{p}$. We compute $\mathbf{M}$ using the Fisher matrix $\mathbf{F}$, defined as:

$$\mathbf{F}_{\alpha\beta} = \frac{1}{2} tr(\mathbf{w}\mathbf{Q}_\alpha \mathbf{w}\mathbf{Q}_\beta). \tag{3}$$

We have a choice for $\mathbf{M}$, and for simplicity in this section we set $\mathbf{M} = \mathbf{I}$, the identity matrix. This is not the case for the analysis of PAPER-64 data, which we describe in Section 3.2. Together, $\mathbf{M}$ and $\mathbf{F}$ form our window function:

$$\mathbf{W} = \mathbf{M}\mathbf{F}. \tag{4}$$

Finally, our final power spectrum is:

$$\mathbf{p} = \mathbf{W}\hat{\mathbf{p}}. \tag{5}$$

In the next sections, we will experiment with different weighting matrices $\mathbf{w}$ and their impact on the resulting power spectra $\mathbf{p}$.

#### 2.1.1. *Toy Model: Inverse Covariance Weighting*

One choice for a weight matrix $\mathbf{w}$ 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. The covariance matrix $\mathbf{C}$ is empirically derived from a data vector $\mathbf{x}$ and defined as:

$$\mathbf{C} \equiv \langle \mathbf{x}\mathbf{x}^\dagger \rangle, \tag{6}$$

assuming $\langle \mathbf{x} \rangle = 0$. The inverse covariance matrix is therefore $\mathbf{C}^{-1}$.

Because $\mathbf{C}$ is computed from the data itself and its inverse is then applied to the data, it carries the risk of over-fitting information in the data and destroying signal. In contrast to $\mathbf{C}$, another choice for a weighting matrix is the identity matrix $\mathbf{I}$. We will refer to this choice as 'unweighted', since using the identity does not alter (doesn't upweight or downweight) the data at all, and therefore does not have signal loss.

We will now build our intuition into how inverse covariance weighting can lead to signal loss through the use of a toy model. Suppose we have a very simple dataset that contains 2-dimensional data (representing 100 time integrations and 20 frequency channels). This model represents realistic dimensions of an ∼hour of PAPER data which could be used for a power spectrum analysis. We create a fake bright foreground signal, $\mathbf{x}_{FG}$, as a complex sinusoid that varies in time and frequency, as well as a fake EoR signal, $\mathbf{x}_{EoR}$, as complex, random-noise. Adding the two together forms our combined dataset, $\mathbf{x}$, which we will be applying different weighting schemes to. The three data vectors are shown on the left in Figure 1, and our ultimate goal is to suppress the foregrounds perfectly without destroying EoR.

We compute the power spectrum of $\mathbf{x}$ using OQE formalism and a weighting matrix of $\mathbf{C}^{-1}$. 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).

Examining this case, our inverse covariance weighted result successfully suppresses foregrounds. This is in principle the reason why inverse covariance weighting is so attractive for 21 cm data. It is also evident that our result almost recovers the EoR signal — it exhibits the correct shape, but the amplitude level is slightly low. This is actually evidence of a small amount of signal loss. In order to understand how this occurs, we need to closely study our covariance matrix $\mathbf{C}$.

It turns out that the eigenspectrum of $\mathbf{C}$ offers a window into understanding how inverse covariance weighting affects data. An eigenspectrum ranks the eigenvalues of $\mathbf{C}$ from highest to lowest and can be thought of as a spectrum of weights that are given to each frequency mode in the data. The eigenspectrum of the identity matrix $\mathbf{I}$ is flat (all 1's) because it gives equal weighting to all modes.

For our toy model, the eigenspectra of $\mathbf{C}_{FG}$, $\mathbf{C}_{EoR}$, and $\mathbf{C}$ are shown in the right plot of Figure 1. The blue curve peaks only at the zeroth eigenmode, revealing that the foregrounds can be described by a single eigenmode. This is a direct consequence of how it was constructed (only one sinusoid). In fact, the blue peak in Figure 2 is also a direct result of our foreground's construction. The eigenspectrum of EoR, however, is fairly flat, signifying that its covariance matrix is similar to an identity matrix. This makes sense because the EoR signal consists of random noise, which should only have strong frequency-frequency correlations when two frequencies are the same.

The eigenmodes with the highest eigenvalues will be down-weighted the most (i.e. when the spectrum is inverted, they will be given the smallest weights). In contrast, the modes with the smallest eigenvalues will be up-weighted the most. In other words, if an eigenspectrum is not perfectly flat, all modes will be given different weights and it is possible to up-weight a few modes much more than others. The subtle slope of the eigenspectrum in this toy example gives rise to a small amount of signal loss because it is possible to overfit a few modes (the highest weighted ones) ever so slightly. We will see in the next section how this effect can be exaggerated as the 'steepness' of the spectrum changes.

Using what we've learned about the eigenspectrum,

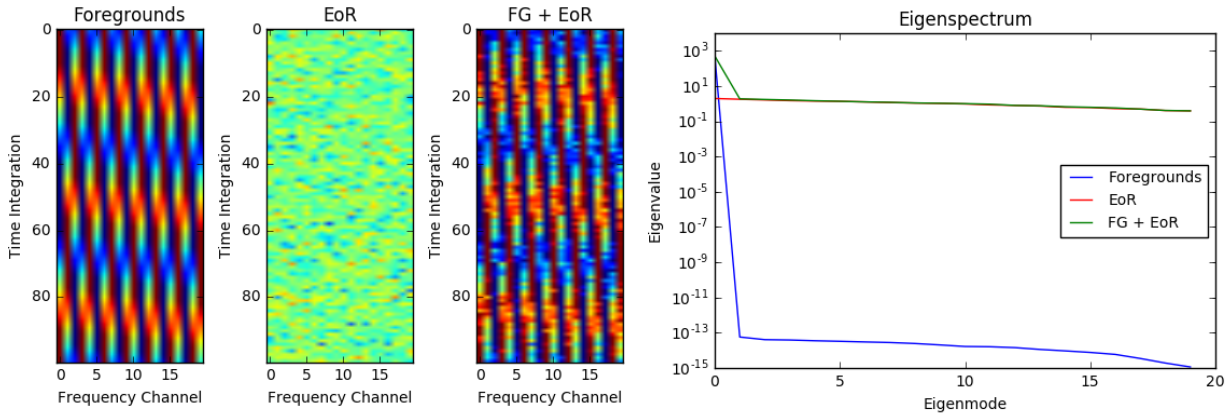


**Figure 1**. Left: Our toy model dataset contains a sine-wave foreground that varies in time and random noise as an EoR signal. Real parts are shown here. Right: Eigenspectrum of foregrounds only (blue), EoR only (red), and both (green)

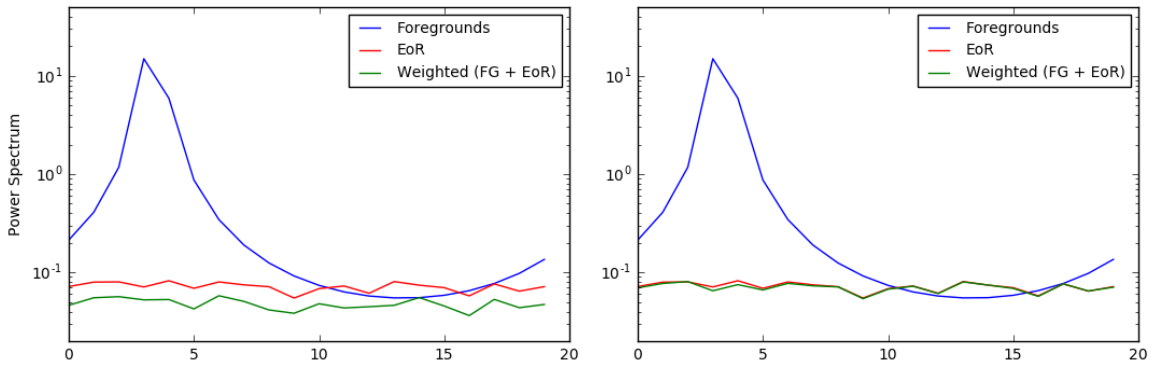


**Figure 2**. Power spectrum of foregrounds only (blue), EoR only (red), and the combined dataset (green). Contrasted are the effect of using inverse covariance weighting (left) and information from the first eigenmode only (right).

we can tweak it in a simple way to suppress foregrounds and yield zero signal loss. Namely, we can zero out all eigenmodes except the first one, thereby down-weighting the foregrounds perfectly and nothing else. The result is shown in the right plot of Figure 2. In this case, we perfectly recover EoR, demonstrating that if we can perfectly model our foregrounds, we can down-weight them without signal loss. However, this is not the case in reality since we aren't able to distinguish between foreground and EoR modes.

#### 2.1.2. *Toy Model: Fringe-Rate Filtering*

We have shown how signal loss can arise due to unintentionally up-weighting eigenmodes with small eigenvalues (overfitting the noise). We will next show how this effect is exaggerated with a reduction of 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). To mimic this filter, we average every 4 time integrations of our toy model dataset together, yielding 25 independent samples in time (Figure 3, left). We choose these numbers so that the total number of independent samples is similar to the number of frequency channels (our matrices will be full rank). The resulting eigenspectra (Figure 3, right), as compared to those in Figure 1, fall more steeply for last few eigenmodes. This shape has large consequences on the resulting power spectra.

Applying inverse covariance weighting to this dataset results in the power spectra shown in Figure 4. There is a much larger amount of signal loss for this time-averaged dataset. To reiterate, the drop-off in the eigenspectrum shape suggests that when the spectrum is inverted, a few modes are up-weighted much more strongly than the rest. This dramatic up-weighting drowns out the EoR signal, giving rise to signal loss. Additionally, most of our power spectrum information is coming from only the last few modes and as a result, is a noisier estimate. This is evident by noticing that the green curve in Figure 4 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. Longer integrations increase sensitivity but reduce the number of independent samples, resulting in steep drop-offs in eigenspectra. For PAPER-64 data, we see $\sim 3$ orders of magnitude of signal loss when combining the use of an optimal fringe-rate filter and inverse covariance weight-

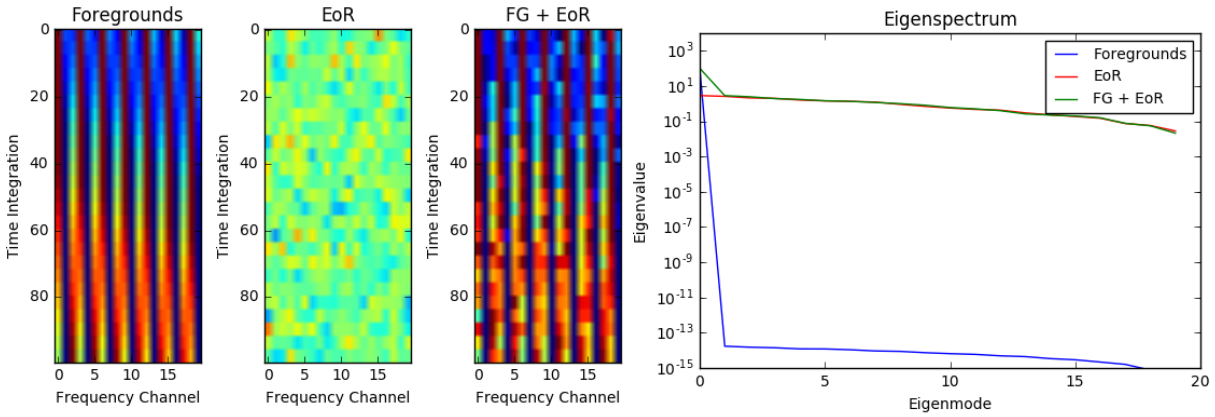


**Figure 3**. Left: Our 'fringe-rate filtered' toy model dataset. It contains 25 independent samples in time. Real parts are shown here. Right: Eigenspectrum of foregrounds only (blue), EoR only (red), and both (green).

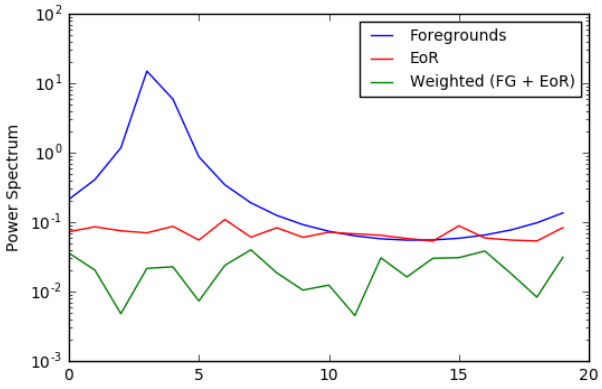


**Figure 4**. Power spectrum of foregrounds only (blue), EoR only (red), and the full inverse covariance weighted combined dataset (green) for a time-averaged dataset (fewer independent modes).

ing.

### 2.1.3. *Toy Model: Other Weighting*

In a previous section we showed how altering an eigenspectrum can make the difference between zero and some signal loss, if we can distinguish between foreground eigenmodes and EoR eigenmodes. We will now use our toy model to describe several other ways to alter a covariance matrix $\mathbf{C}$ in order to minimize signal loss.

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 $\mathbf{C}_{FG} = \mathbf{x}_{FG}\mathbf{x}_{FG}^{\dagger}$ to obtain a final $\mathbf{C}$ to use in weighting, we see that there is negligible signal loss (Figure 5, upper left).

The next three panels of Figure 5 each involve methods that regularize the matrix $\mathbf{C}$ after it is computed from the foreground + EoR dataset. By regularizing $\mathbf{C}$, we are flattening out its eigenspectrum. Therefore, some modes will be down-weighted, but the rest will be treated with similar weights. Hence, we won't be up-weighting the last few noisy modes.

Regularization schemes that we are highlighting here are adding an identity matrix to $\mathbf{C}$ (Figure 5, upper right), killing all but the first three eigenmodes of $\mathbf{C}$ (lower left), and multiplying an identity matrix to $\mathbf{C}$ (lower right). Each avoids signal loss, but it is clear that two of the methods down-weight foregrounds much better than the third. For this toy model, our foregrounds are spread out in frequency and therefore have non-negligible frequency-frequency correlations. Therefore, when multiplying $\mathbf{C}$ by an identity matrix, we are only preserving correlations between the same frequencies. Because we killed off information from all other frequency combinations, we do a poor job suppressing the foreground.

Although this 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. 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 $\mathbf{C}$ 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. Inverse covariance weighting is an optimal weighting for dealing with foregrounds, but its aggressiveness results in sloped eigenspectra shapes, the overfitting of noise, and thus signal loss. The application of a fringe-rate filter heightens this effect by reducing the number of independent samples and steepening eigenspectra further. Rather than use inverse covariance weighting, there are several alternate ways to regularize covariance matrices and flatten out eigenspectra to minimize signal loss. In Section 4, we apply the lessons we've learned about signal loss and its interaction with fringe-rate filtering and inverse covariance weighting to PAPER-64 data.

## 2.2. *Error Estimation*

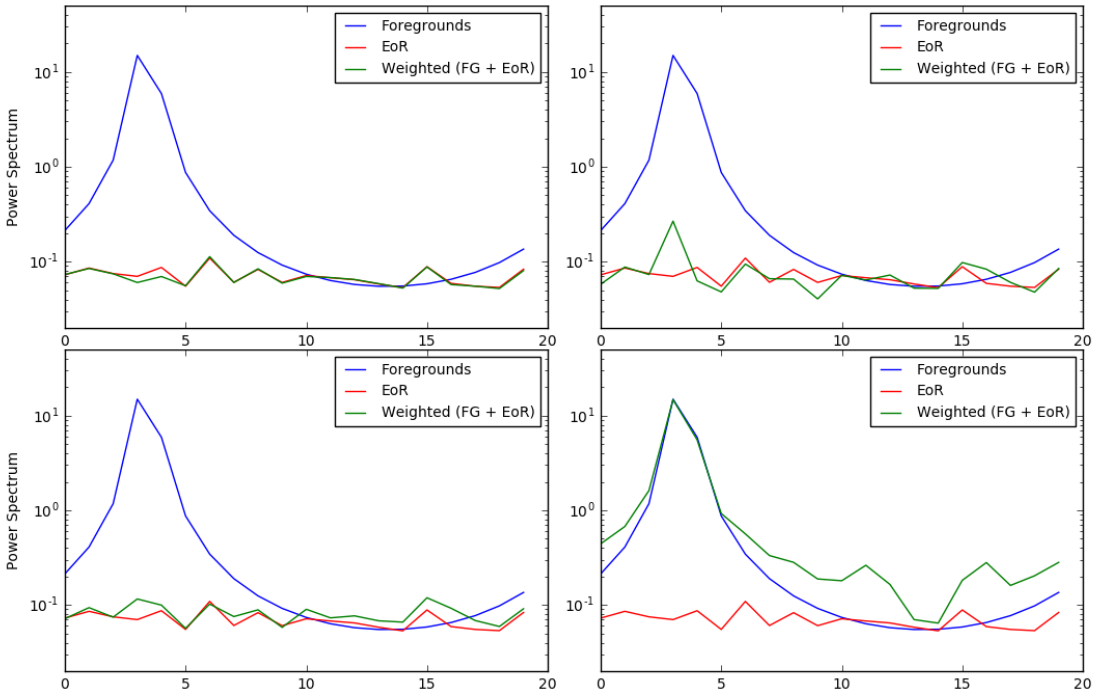


**Figure 5**. Power spectra of foregrounds only (blue), EoR only (red), and the weighted combined dataset (green). We show four alternate weighting options that each avoid signal loss, including modeling the covariance matrix of EoR (upper left), regularizing $\mathbf{C}$ by adding an identity matrix to it (upper right), using only the first few eigenmodes of $\mathbf{C}$ (lower left), and multiplying an identity matrix to $\mathbf{C}$ (lower right).

Two ways of estimating errors on a power spectrum measurement are exploring 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 contaminates in the data that prevent it from being perfectly noise-limited. 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 variance in a dataset is bootstrapping. Bootstrapping is a metric that relies on sampling with replacement and allows the estimation of a sampling distribution. For example, power spectrum measurements of 21 cm data can be made along many axes, including time and baselines. Through the process of bootstrapping, we can obtain error bars for our results which represent the underlying distribution of power spectra values that are allowed by our measurements.

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

We form 100 bootstraps, each comprised of an array of length 1000 that is created by random sampling the original data with replacement. The standard deviation over the 100 bootstraps gives an error estimation for our dataset. As shown in Figure 6 (grey star), the error computed from bootstrapping matches our theoretical prediction.

One major caveat of bootstrapping is that the method is no longer valid when working with correlated data. In other words, bootstrapping assumes completely independent samples. Therefore, the use of a fringe-rate filter, which averages data in time to increase sensitivity, creates a situation in which bootstrapping can underestimate errors.

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 exact method as described earlier (drawing the same number of samples as the length of the dataset), under-estimates the error by a factor of $\sim 3$. This occurs because we are drawing more samples than independent ones available. In fact, the error derived from bootstrapping is a strong function of the number of samples that are drawn (Figure 6, black points), and we can both under-estimate the error by drawing too many or over-estimate it by drawing too few. However, in this case we know that we have 100 independent samples, and the error associated with drawing 100 samples with replacement does match the theoretical prediction as expected.

This examples highlights the importance of understanding how analysis techniques (like a fringe-rate filter) can affect a common statistical procedure (like bootstrapping). Bootstrapping as a means of estimating power spectrum errors from real fringe-rate filtered data

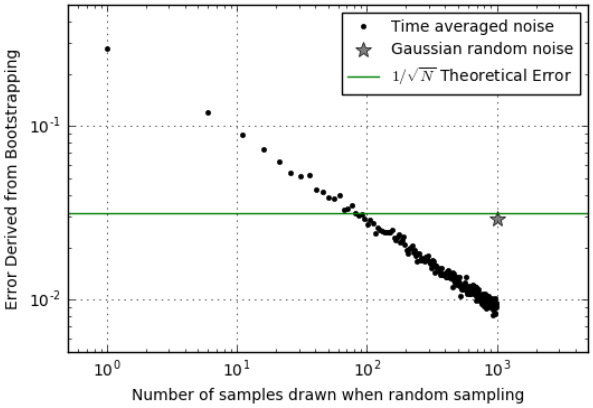


**Figure 6**. Error estimation produced by bootstrapping as a function of the number of samples drawn when sampling with replacement. The grey star represents the error associated with drawing 1000 samples from a length 1000 array of random noise. The black points correspond to time-averaged data and illustrate how errors can be underestimated if drawing more samples than independent ones in the data. There are 100 independent samples in the time-averaged data and using this number does indeed match the theoretical error prediction.

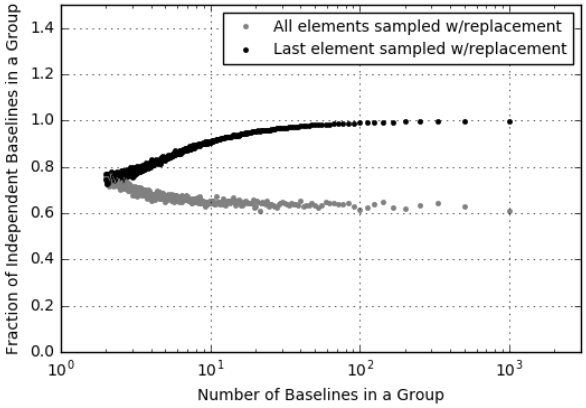


**Figure 7**. Fraction of independent baselines in a baseline group as a function of the number of baselines in a group. A fraction of 1.0 means that all baselines in a group are independent, and therefore sensitivity is maximized (alternatively, this axis can be treated as a power spectrum sensitivity in temperature-squared space). 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 by not maximizing baseline sensitivity to its full capacity.

requires precise knowledge of the number of independent samples, which is not a trivial task.

We will now discuss a second subtle feature of bootstrapping that can lead to an over-estimation of errors. Suppose we have 1000 baselines (another appropriate axis for bootstrapping power spectra), each an independent measurement of the sky. Bootstrap theory requires a bootstrap to comprise of some combination of the 1000 baselines. If all 1000 spaces are filled randomly, there is in fact a high probability that baselines will be repeated in a sample (sometimes many times). Thus, bootstrapping in this way negatively affects power spectrum sensitivity because the number of independent baselines in a sample can be significantly lower than the total number of baselines.

In order to maximize sensitivity, we use a slightly modified bootstrapping method. We first shuffle all the baselines for a bootstrap, take the first 999, and then fill the last slot randomly with replacement. There's a small chance it will yield 1000 independent samples, but even with 999 our sensitivity is nearly maximized. One may wonder whether this change is still a legitimate way of error estimating since the number of possible random samples for a single bootstrap has dramatically decreased. However, as long as this number is still much greater than the number of bootstraps we perform, it is a valid way to uncover the inherent variability in a dataset.

For our toy model, the fraction of independent baselines using this method (999/1000) is over 1.5 times greater than the aforementioned approach. These two methods converge for small numbers of samples, when filling a few spots randomly is nearly the same as filling only the last spot randomly. The evolution of both of these methods as a function of number of baselines is shown in Figure 7. In order to reduce the number of baselines, we divide our 1000 baselines into $N$ groups, where the number of baselines in a group is $1000/N$. No baselines are repeated within or between groups, until a bootstrapping method is applied to each (either sampling all elements with replacement, or only the last one).

It is clear that the fraction of independent baselines in a group is always greater using our new sampling approach, with the greatest discrepancy for small number of groups (large number of baselines per group). An analog for the y-axis in Figure 7 is the final power spectrum sensitivity in temperature-squared space, which decreases as the square root of the number of baselines that are averaged together for a measurement.

In summary, bootstrapping is still an effective and straightforward way to estimate errors of a dataset. However, one must be careful about violating independence assumptions, as it is possible to under-estimate errors by over-sampling data. Additionally, small changes in the way datasets are randomly sampled can ensure that the overall sensitivity is maximized.

We have discussed the interplay between two of our power spectrum techniques — fringe-rate filtering and bootstrapping — and how they affect the theme of error bar estimation. In addition to these, inverse covariance weighting can also affect error bars. Specifically, a power spectrum measurement and its errors are susceptible to blowing up when the number of non-zero eigenmodes are less than the number of independent samples in a dataset (i.e. the covariance matrix is not full rank). **[Not really sure how to explain this or how much detail to get into here...]**

Our fourth power spectrum technique, jack-knife testing, is also related to error estimation. Splitting data

along various axes and performing the same power spectrum analysis on each provides a test of robustness. If the same errors are obtained for each data subset, we can have confidence that they are unbiased.

### 2.3. *Bias*

As alluded to in the previous section, jack-knife testing is a powerful way to tease out biases by studying how power spectrum measurements of different subsets of data vary. Biases can arise due to...

## 3. CASE STUDY: PAPER-64

Now that we have developed our intuition regarding signal loss, error estimation, and bias, we will look at each in more detail as applied to data taken by PAPER. We begin with an overview of the telescope array, its observations, and analysis pipeline.

### 3.1. *Observations*

The Donald C. Backer Precision Array for Probing the Epoch of Reionization (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, each 2 m on a side. The antenna layout is formatted in a grid layout (Figure 8), with 8 antennas on a side, 30 m spacing between antennas along the East/West direction, and 4 m spacing between antennas along the North/South direction. For the rest of this paper, we focus on data from only the 30 m pure East/West baselines.

PAPER-64 observed for a total of 135 nights between 2012-2013. The correlator processes a bandwidth of 100-200 MHz, corresponding to a redshift range of 6-12. For more information about the backend system of PAPER-64 and its observations, we refer the reader to [??] and Ali et al. (2015).

Because there is a detailed discussion of the PAPER-64 data reduction pipeline in Ali et al. (2015), here we will only briefly summarize the data processing steps prior to the power spectrum analysis.

Beginning with compressed data products which have been cleaned of radio frequency interference (RFI) at the $6\sigma$ level and then down-sampled (to 42.9 s integrations, 203 frequency channels), we then employ the technique of redundant calibration as a means of calibrating for individual antenna gains without needing to use information about the sky. Because PAPER is laid out on a grid, antenna calibration parameters can be found through the use redundancy. Namely, baselines of the same length and orientation should measure the same sky signal, but in practice there are differences due to instrumental effects caused by antennas, cables, and receivers. Using the basis of redundancy, however, allows us to set up a system of equations to solve for a complex gain per antenna that brings all baseline of a particular type into alignment. We do this using the package OMNICAL.

We next solve for the array's overall phase and gain calibration parameters using a standard self calibration method. We used radio sources Pictor A, Fornax A, and the Crab Nebula to fit for an overall phase solution, and set our overall flux scale using Pictor A as a calibrator source. Although PAPER-64 data exists for all 4 polarizations, we only use the $xx$ and $yy$ polarization data to form Stokes I as $V_I = \frac{1}{2}(V_{XX} + V_{YY})$.

Eliminating bright foregrounds remains a challenging yet crucial component of any 21 cm data analysis. There are many techniques to go about foreground removal, including spectral polynomial fitting, principle component analysis, Fourier-mode filtering, non-parametric subtractions, and inverse covariance weighting. PAPER uses a delay-spectrum filtering method which was first explained and applied in Parsons et al. (2014). In short, the delay-filtering technique employs the spectrally smooth nature of foregrounds, which are consequently localized in delay-space, the Fourier dual to frequency. We subtract all components that fall within the horizon limit for a specific baseline type in delay-space, plus a 15 ns buffer, thus gaining about $\sim 4$ orders of magnitude in sensitivity.

After removing an additional layer of RFI (values $3\sigma$ above the median), we average together all our data into two datasets: even julian dates and odd julian dates. A total of 124 nights of data comprises this average.

Finally, our last analysis step before power spectrum estimation is the use of a fringe-rate filter. As shown in Parsons et al. (2016), a fringe-rate filter allows us to maximize our sensitivity even further by averaging visibilities in time. We apply this filter in the fringe-rate domain, the Fourier-dual to time, and it effectively amounts to increasing our data's integration time. Another way to describe the filter is that it weights different parts of the sky (which rotate at different "fringe-rates") by the antenna beam. This allows us to up-weight parts of the sky high in the primary beam, and down-weight those that are less sensitive. In practice, the optimal fringe-rate profile is computed for a fiducial 30 m baseline at 150 MHz, the center frequency in our band. The filter is implemented on a per baseline basis by convolving the time-domain visibilities with the Fourier transform of the fringe-rate filter, resulting in averaged visibilities yielding $\sim 2$ orders of magnitude in sensitivity.

### 3.2. *Power Spectrum Estimation*

To form power spectrum quantities using our two fringe-rate filtered, LST-binned datasets, we use OQE methods as summarized in Section 2.1. Here we will describe its application to PAPER-64 data.

We have a 2-dimensional data vector $\mathbf{x}$ for each baseline and data group ('even' or 'odd'). As part of our bootstrapping routine (and to speed things up), we divide up our total number of baselines into 5 random, equal-sized groups. Taking $\mathbf{x}$ to be averaged data from all baselines within a group, we now have 10 possible data vectors (for 5 groups and 2 datasets).

We use a weight matrix of $\mathbf{w} =$ [???] in our analysis. Notice that there are two quantities of $\mathbf{wx}$ in the ex-

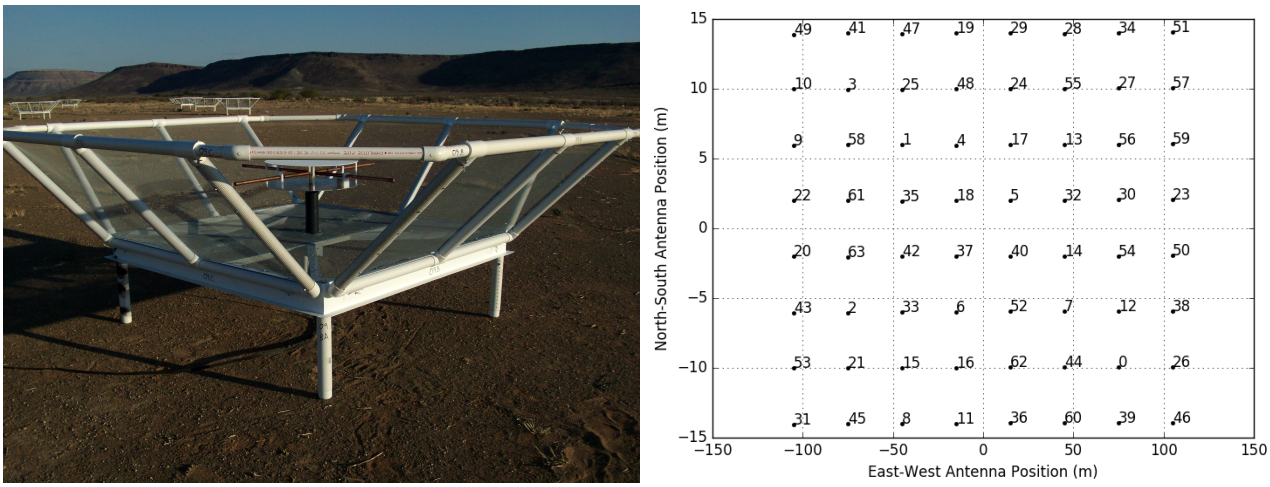


**Figure 8**. PAPER dipole in South Africa (left) and PAPER-64 antenna layout (right).

pression for $\hat{\mathbf{q}}$ in Equation 1, representing two copies of weighted data. We perform all possible cross-multiplications of this quantity, except between the same datasets ('even' with 'even', for example) and same groups ('baseline group 1' with 'baseline group 1', for example). We avoid these computations to prevent the introduction of biases from shared auto power.

As mentioned previously, we have a choice for our normalization matrix $\mathbf{M}$. For our analysis, we choose to compute it by taking the Cholesky decomposition of $\mathbf{F}$. Namely, we decompose the Fisher matrix such that $\mathbf{F} = \mathbf{L}\mathbf{L}^\dagger$, where $\mathbf{L}$ is a lower triangular matrix. Next, we construct $\mathbf{M} = \mathbf{D}\mathbf{L}^{-1}$, where $\mathbf{D}$ is a diagonal matrix. Hence, our window function $\mathbf{W} = \mathbf{M}\mathbf{F}$ becomes $\mathbf{W} = \mathbf{D}\mathbf{L}^\dagger$ and is an upper triangular matrix. This window function was constructed in a way to prevent 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.

In practice, our final power spectrum result is a 2-dimensional quantity (a function of both time and $k$). We compute 20 bootstraps of these quantities, where each bootstrap creates baseline groups randomly. We form both the quantity $\mathbf{P(k)}$ and the folded version $\Delta^2\mathbf{(k)}$, defined as:

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

## 4. APPLICATION: SIGNAL LOSS

### 4.1. *Noise and EoR*

One critical new component of our power spectrum pipeline is that we have many different power spectrum channels that simultaneously get processed at the same time and are especially useful for signal loss (Section 4) and sensitivity (Section **??**) verification. Two important channels, in addition to our PAPER-64 data, are a noise dataset and an EoR dataset (in addition to these, we do different combinations of all three). We will briefly describe each.

We create random noise with a scale determined by the sensitivity of our instrument and parameters of our dataset. We calculate our system temperature for our frequencies of interest as:

$$T_{sys} = 180\left(\frac{\nu}{0.18}\right)^{-2.55} + T_{rcvr} \tag{8}$$

where $\nu$ 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 **[why?]**. We convert this temperature into a variance statistic using:

$$T_{rms} = \frac{T_{sys}}{\sqrt{\Delta\nu\Delta t N_{days} N_{pols}}} \tag{9}$$

where $\Delta\nu$ is channel spacing, $\Delta t$ is integration time, $N_{days}$ is the number of data samples for a particular time and frequency that went into our LST binned set, and $N_{pols}$ is the number of polarizations (2 for Stokes I). Our simulated noise has the same shape as our data, and we fringe-rate filter it in the same way to best mimic the real noise floor of our data.

A second additional important channel in our pipeline is a simulated EoR signal. We again simulate random noise (with a default scale of 1) with the same shape as our data, and we fringe-rate filter this signal twice. The first filter transforms the unattached noise into a signal that's attached to the sky (what our instrument observes). The second filter represents the fringe-rate filtering step in our data analysis pipeline. As described in Section 4, we create an EoR signal at various amplitude levels.

Our multi-channel power spectrum pipeline has been essential in helping us understand the nuances in our pipeline. Because there are many different components and countless subtle effects affecting our final limits, it

has been imperative to carry all our channels through the analysis to validate each step.

### 4.2. *Signal Loss Methodology*

Based on our analysis pipeline, potential signal loss is a real and significant issue. More specifically, when applying inverse covariance weighting, $\mathbf{C}^{-1}$ is empirically estimated from the data itself, which has the consequence of over-fitting the noise in the data, producing power spectra values well below the thermal noise limit that is predicted based on observation parameters. This is especially prevalent when weighting fringe-rate-filtered data, which has so few independent time modes to begin with, leading to a noisier dataset. Being able to accurately quantify this loss is crucial in interpreting and providing credibility to any power spectrum limits.

New to the PAPER-64 analysis is a robust method to estimate signal loss associated with inverse covariance weighting. This method, explained below, is now a standard analysis step for all PAPER analyses and one that will be used for HERA moving forward.

As discussed previously, our power spectrum pipeline runs on a standardized set of channels (pure data, pure noise, pure EoR, and combinations of the three). As part of our signal loss routine, we also compute power spectra with various levels of the created EoR signal, dialing its amplitude 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:

$$\mathbf{r} = \mathbf{x} + \mathbf{e} \tag{10}$$

Using our OQE formalism, we are interested in the following two quantities: $P_{in}$ and $P_{out}$. The input power spectrum, $P_{in}$ represents the unweighted power spectrum of only $\mathbf{e}$, our simulated EoR signal. The output power spectrum, $P_{out}$, 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:

$$P_{in} \propto \mathbf{e}^\dagger \mathbf{I}^{-1}\mathbf{Q}\mathbf{I}^{-1}\mathbf{e} \tag{11}$$

$$\begin{aligned} P_{out} \equiv \mathbf{P}_e &= \mathbf{P}_r - \mathbf{P}_x \\ &\propto \mathbf{r}^\dagger \mathbf{C}_r^{-1}\mathbf{Q}\mathbf{C}_r^{-1}\mathbf{r} - \mathbf{x}^\dagger \mathbf{C}_x^{-1}\mathbf{Q}\mathbf{C}_x^{-1}\mathbf{x} \end{aligned} \tag{12}$$

It is noted that the output power spectrum is comprised of two terms: the covariance treated 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 $P_{out} \propto \mathbf{e}^\dagger \mathbf{C}_e^{-1}\mathbf{Q}\mathbf{C}_e^{-1}\mathbf{e}$. Expanding Equation 12:

$$\begin{aligned} P_{out} &\propto (\mathbf{x}+\mathbf{e})^\dagger \mathbf{C}_r^{-1}\mathbf{Q}\mathbf{C}_r^{-1}(\mathbf{x}+\mathbf{e}) - \mathbf{x}^\dagger \mathbf{C}_x^{-1}\mathbf{Q}\mathbf{C}_x^{-1}\mathbf{x} \\ &\propto \mathbf{x}^\dagger \mathbf{C}_r^{-1}\mathbf{Q}\mathbf{C}_r^{-1}\mathbf{x} + \mathbf{e}^\dagger \mathbf{C}_r^{-1}\mathbf{Q}\mathbf{C}_r^{-1}\mathbf{e} + \mathbf{x}^\dagger \mathbf{C}_r^{-1}\mathbf{Q}\mathbf{C}_r^{-1}\mathbf{e} \\ &\quad + \mathbf{e}^\dagger \mathbf{C}_r^{-1}\mathbf{Q}\mathbf{C}_r^{-1}\mathbf{x} - \mathbf{x}^\dagger \mathbf{C}_x^{-1}\mathbf{Q}\mathbf{C}_x^{-1}\mathbf{x} \end{aligned}$$

And taking the case of very large $\mathbf{e}$, so that $\mathbf{C}_r^{-1} \sim \mathbf{C}_e^{-1}$ and any terms involving only $\mathbf{x}$ are small:

$$\begin{aligned} P_{out,\mathbf{e}\gg\mathbf{x}} \propto \mathbf{e}^\dagger \mathbf{C}_e^{-1}\mathbf{Q}\mathbf{C}_e^{-1}\mathbf{e} + \mathbf{x}^\dagger \mathbf{C}_e^{-1}\mathbf{Q}\mathbf{C}_e^{-1}\mathbf{e} \\ + \mathbf{e}^\dagger \mathbf{C}_e^{-1}\mathbf{Q}\mathbf{C}_e^{-1}\mathbf{x} \end{aligned} \tag{13}$$

We see that our naive expression for $P_{out}$ is the first term, but there also two additional terms. An initial assumption would be that the cross-terms that involve both $\mathbf{e}$ and $\mathbf{x}$ should be zero, since the two quantities are un-correlated. However, **[need explanation about power in cross-terms here]**. Therefore, in our investigation of signal loss, we use the full quantity for $P_{out}$ as in Equation 12.

For the unweighted case ($\mathbf{C} \equiv \mathbf{I}$), we expect $P_{out}$ and $P_{in}$ to be equal, and hence the ratio of $P_{in}/P_{out}$ to be 1. For the weighted case, this is not true due to signal loss. In order to quantify the loss, we look at the ratio of $P_{in}$ to $P_{out}$ as the amplitude level of the injected signal $\mathbf{e}$ is increased. In the next section, we highlight two methods that yield similar results for the determination of signal loss using $P_{in}$ and $P_{out}$

### 4.3. *Signal Loss in Practice*

Recall that fringe-rate filtered noise, which mimics the level of noise in our actual PAPER-64 dataset, is a channel in our power spectrum pipeline. We can compute signal loss quantities of interest for the noise $\mathbf{n}$ similar to the expressions we featured previously for data $\mathbf{x}$.

$$P_{in} \propto \mathbf{e}^\dagger \mathbf{I}^{-1}\mathbf{Q}\mathbf{I}^{-1}\mathbf{e} \tag{14}$$

$$\mathbf{s} = \mathbf{n} + \mathbf{e} \tag{15}$$

$$P_{out} \propto \mathbf{s}^\dagger \mathbf{C}_s^{-1}\mathbf{Q}\mathbf{C}_s^{-1}\mathbf{s} - \mathbf{n}^\dagger \mathbf{C}_n^{-1}\mathbf{Q}\mathbf{C}_n^{-1}\mathbf{n} \tag{16}$$

Using the input and output power spectra for range of EoR amplitudes, we use two methods to determine signal loss associated with the over-fitting of noise during inverse covariance weighting. We first look at signal loss for the pure noise case (no data), to show that we can successfully inject EoR signals, determine signal loss, and then recover the signal.

The first method is very straightforward. Post-bootstrapping, our final power spectra 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 9. The shape of this function can be explained as follows. At small injection levels (small $\mathbf{e}$), $P_{out}$ 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 an unphysical feature, but rather illustrates that there is some non-negligible cross-term power between $\mathbf{n}$ and $\mathbf{e}$.

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 $2\sigma$ power spectrum upper limit of noise alone. In other words, we look at $P_{noise} \propto \mathbf{n}^\dagger \mathbf{C}_n^{-1} \mathbf{Q} \mathbf{C}_n^{-1} \mathbf{n}$, compute its $2\sigma$ upper limit (mean over bootstraps + 2×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 $k$.

Figure 10 shows the power spectrum of our noise simulation, using full inverse covariance weighting, both before and after signal loss correction. Prior to signal loss correction, it is obvious 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. This is an effect caused by the steep nature of the eigenspectrum of $\mathbf{C}$, and is explained more in Section 4.4.

**[Need a plot that shows our signal loss factors are CORRECT. How to do that??]**

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 11 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 9, but otherwise shows a familiar trend. Similarly, our weighted noise power spectra also has a defined spread due to bootstrapping.

Using these two distributions ($P_{in}/P_{out}$ and $P_{noise}$), we can create bins along the $P_{noise}$ axis to yield histograms of signal loss factors for each bin. We similarly sort the values of $P_{noise}$ into the same bins, and multiply the probability of $P_{noise}$ 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 $k$. As shown in Figure 12, the results 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.

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 $\mathbf{C}$.

### 4.4. *Data Weighting*

With our signal loss formalism established, we now have the capability of experimenting with different weighting options for $\mathbf{C}$. 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 **[?? finish sentence...]**.

We now turn our attention to power spectra using the 30 m East/West baselines of PAPER-64. Our dataset spans 8.5 hours of LST (.1-8.6 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 $z = 8.4$.

Using full inverse covariance weighting, our results are not too dissimilar to that of pure noise. Signal loss factors (Figure 13) are of similar order of magnitude, and our power spectrum blows up past the unweighted version after signal loss correction (Figure 14).

Looking into this behavior in more detail, we investigate the shape of the eigenspectrum of $\mathbf{C}$ for a typical baseline used in the analysis. Figure 15 shows this spectrum for baseline (1,4). Most obviously, the spectrum is steep, spanning 4 orders of magnitude. Not as obvious is the effect of this shape on our results. When the matrix $\mathbf{C}$ is inverted to form $\mathbf{C}^{-1}$, the effect of the steepness of the eigenspectrum is to up-weight very few modes of the sky while the rest are drastically down-weighted. More specifically, our fringe-rate filtered data contains a finite, small number of independent modes, thereby resulting in a covariance matrix that can be described by just a few modes. Beyond the first few modes, the eigenvalues of each additional mode falls of dramatically. When inverting, we end up not only down-weighting those initial modes but severely up-weighting a few insignificant ones. Because of our weighting choice, signal loss blows up as it thinks we only have a couple modes in our data.

Clearly the full inverse covariance treatment of our data is suboptimal to even the unweighted case, but 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 $\mathbf{C}$, but here we will illustrate [?] promising ones as applied to PAPER-64.

**[TO DO: decide on/explain/show different weightings]**

## 5. APPLICATION: ERROR ESTIMATION

### 5.1. *Thermal Noise*

**[Add info on sensitivity equation, how we calculate effective inttime and adjust beam factor because of FRF, etc.]**

### 5.2. *Bootstrapping*

**[Include info about baseline bootstrapping:]** For the Ali et al. (2015) method, each group is then

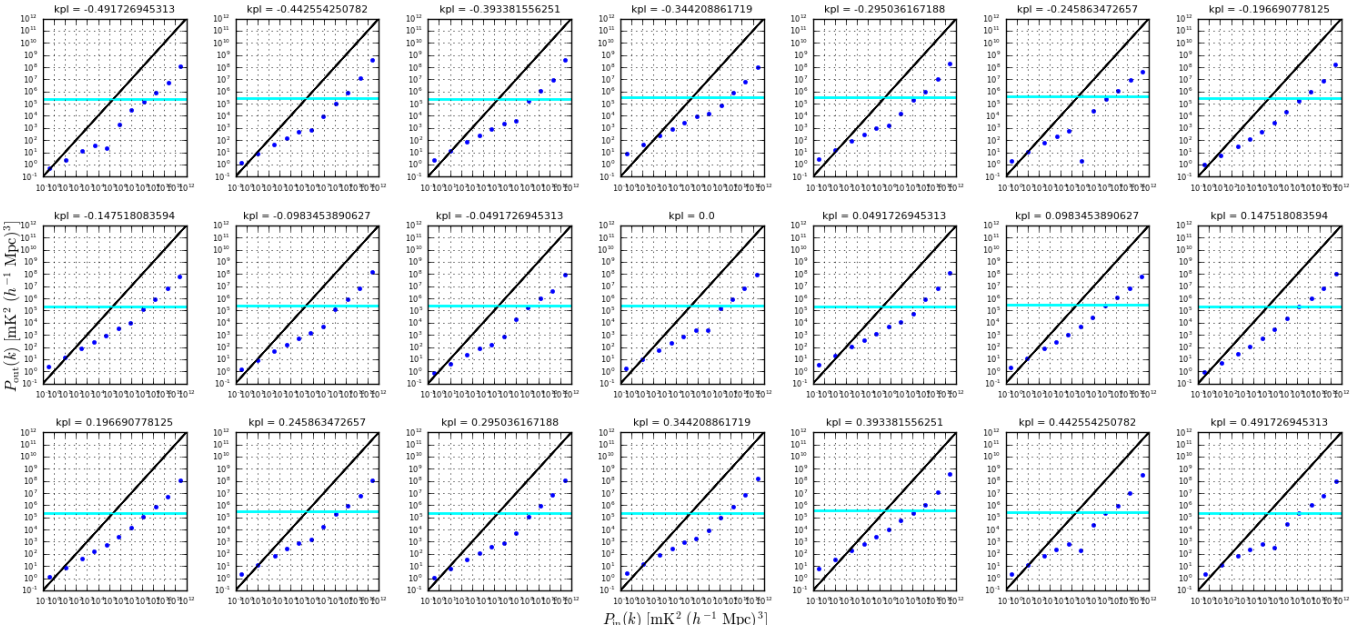


**Figure 9**. $P_{in}$ vs. $P_{out}$ (blue points) for 15 injection levels and 21 $k$'s. The solid cyan line is the $2\sigma$ upper limit for the weighted power spectrum of noise alone, and it is at this level where the signal loss factor $P_{in}/P_{out}$ is computed by interpolation. **[Maybe re-do this plot with closer-together points]**

sampled with replacement to create a new group, of the same size, that can have repeated baselines inside it. We discover that in doing so, we are sacrificing some of our sensitivity since this results in there being 3-4 repeated baselines per group. 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. For example, if we have 10 baselines in a group, we use the first 9 to guarantee at least 9 independent measurements, and then fill the last slot randomly out of the 10. We do this for all 5 groups. This is still a valid means of bootstrapping because there are many more possibilities of baseline groupings than the number of bootstraps we run for this analysis ($nboots = 20$).

**[Include info about second round of bootstrapping:]** In Ali et al. (2015), a second round of bootstrapping occurs over the 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.

However, we have found that this method greatly underestimates power spectrum errors, especially for fringe-rate filtered data. This can be explained by the fact that fringe-rate filtered data has a dramatically reduced number of independent modes. Hence, drawing 100 samples out of a length-100 dataset that only has 5 independent modes in it, for example, results in a narrower distribution of values that leads to a false error estimation.

To avoid this issue, we instead take a simple average along the time axis. Our final power spectrum limits are computed by taking the mean and standard deviation over our single bootstrap axis.

## 6. APPLICATION: BIAS

**[Include info about optimal FRF:]** In Ali et al. (2015), the filter applied was degraded (widened in fringe-rate space) from an optimal one that can be constructed based on the integration time of the data. This was chosen to decrease the resulting integration time, therefore increasing the number of independent modes and reducing signal loss associated with applying optimal quadratic estimators (OQE) on data with too few modes.

New to our updated PAPER-64 analysis is the use of the optimal fringe-rate filter, which maximizes our sensitivity. With the development of a robust method for assessing signal loss (see Section 4), we feel comfortable using a narrow filter (Figure 16), resulting in an effective integration time of 3857 s (see Section **??** for calculation) and 8 total independent modes for our 8.5 hours of LST.

## 7. CONCLUSION

## 8. ACKNOWLEDGEMENTS

**[NSF Graduate Research Fellowship Program (GRFP) Fellowship] [UC Berkeley Chancellor's Fellowship]**

## REFERENCES

Ali, Z. S., et al. 2015, ApJ, 809, 61

Bowman, J. D., & Rogers, A. E. E. 2010, Nature, 468, 796

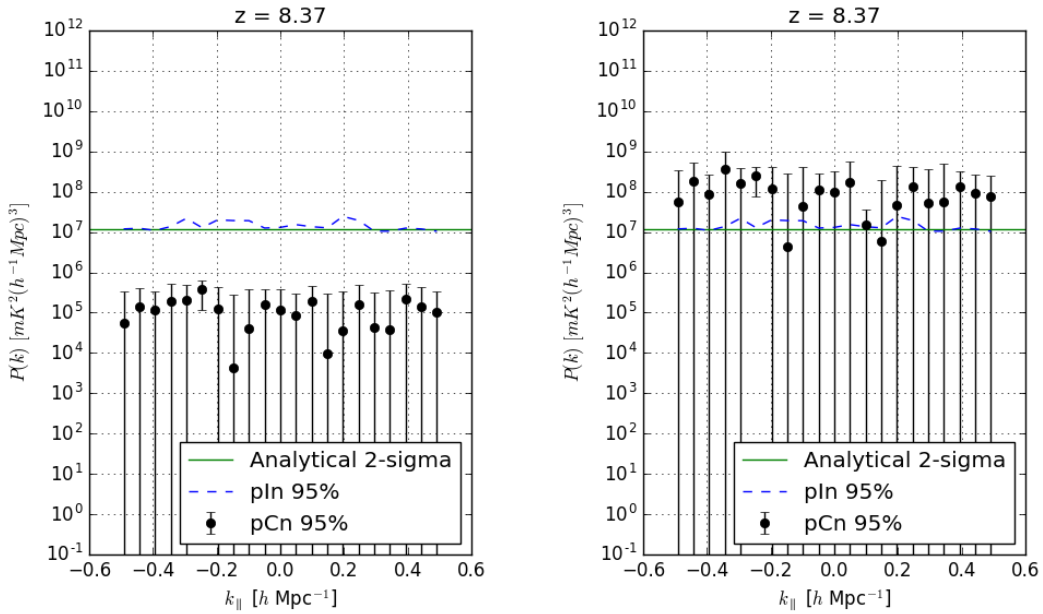


**Figure 10**. Full inverse covariance weighted power spectrum of pure noise (black points, with $2\sigma$ error bars) before signal loss correction (left) and after (right). The dashed blue line is the unweighted power spectrum ($2\sigma$ upper limit). The solid green line is the theoretical noise level prediction based on observational parameters. **[Color negative points grey]**

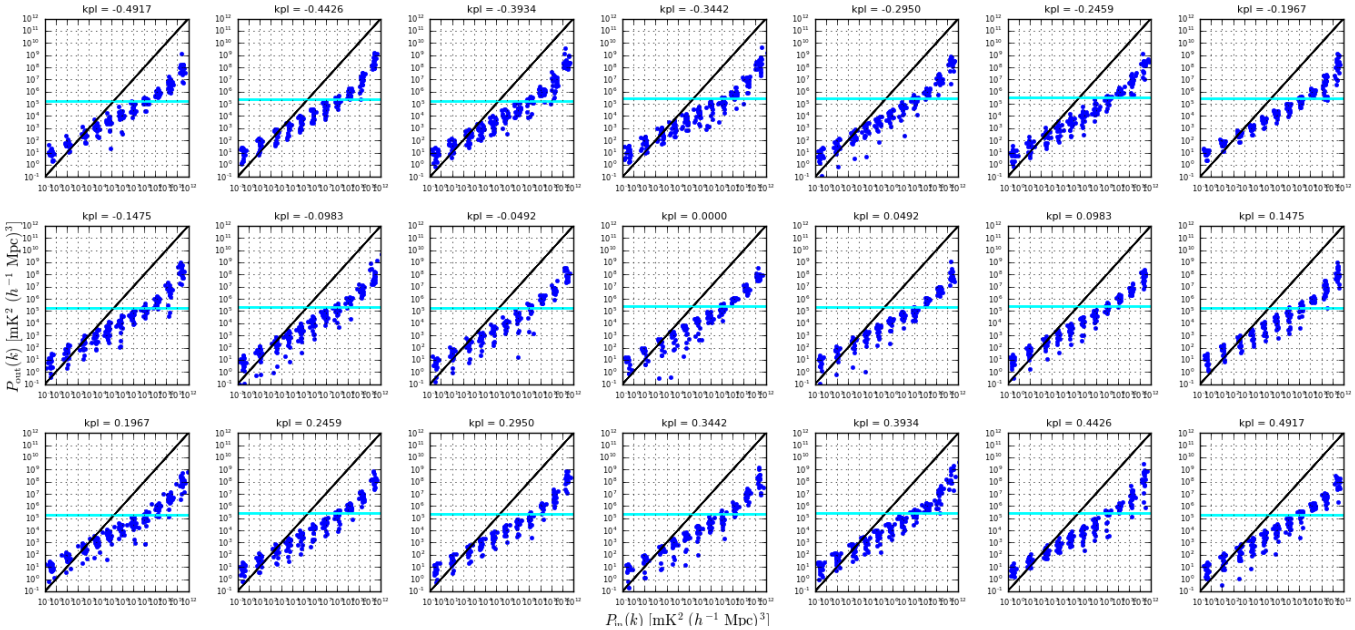


**Figure 11**. $P_{in}$ vs. $P_{out}$ (blue) for 15 injection levels, 20 bootstraps, and 21 $k$'s. **[Plot a semi-transparent cyan range of pCn values instead of just the max]**


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

Liu, A., Parsons, A. R., & Trott, C. M. 2014, PhRvD, 90, 023019

Paciga, G., et al. 2013, MNRAS

Parsons, A. R., Liu, A., Ali, Z. S., & Cheng, C. 2016, ApJ, 820, 51

Parsons, A. R., et al. 2010, AJ, 139, 1468

—. 2014, ApJ, 788, 106

Patra, N., Subrahmanyan, 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

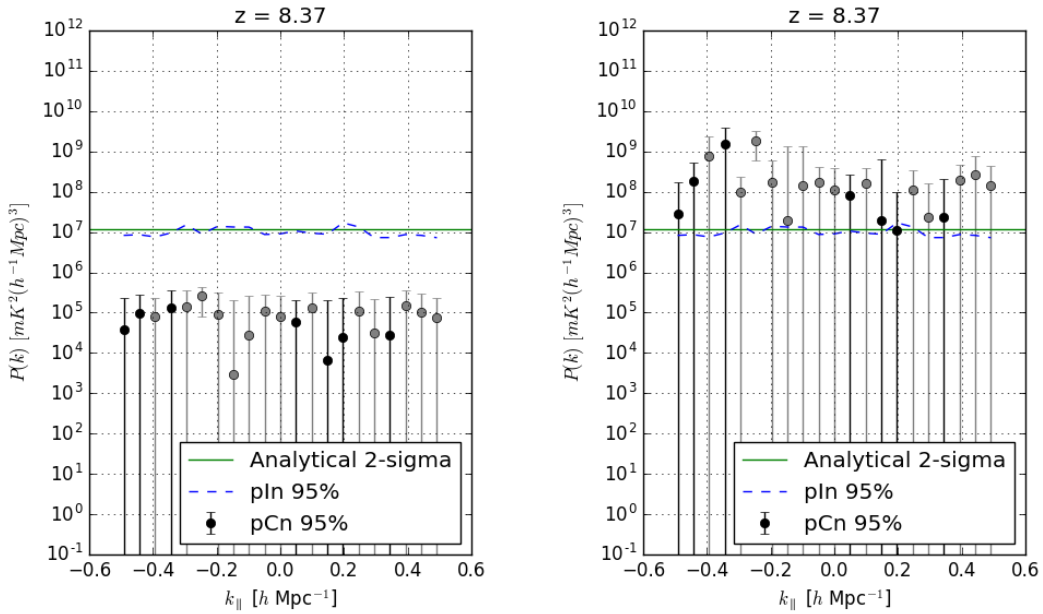


**Figure 12**. Full inverse covariance weighted power spectrum of pure noise (black and grey points, with $2\sigma$ error bars) before signal loss correction (left) and after (right). Black points correspond to positive values, while grey points correspond to originally negative values that have been made positive for plotting. The dashed blue line is the unweighted power spectrum ($2\sigma$ upper limit). The solid green line is the theoretical noise level prediction based on observational parameters.

Tingay, S. J., et al. 2013, PASA, 30, 7
van Haarlem, M. P., et al. 2013, A&A, 556, A2
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

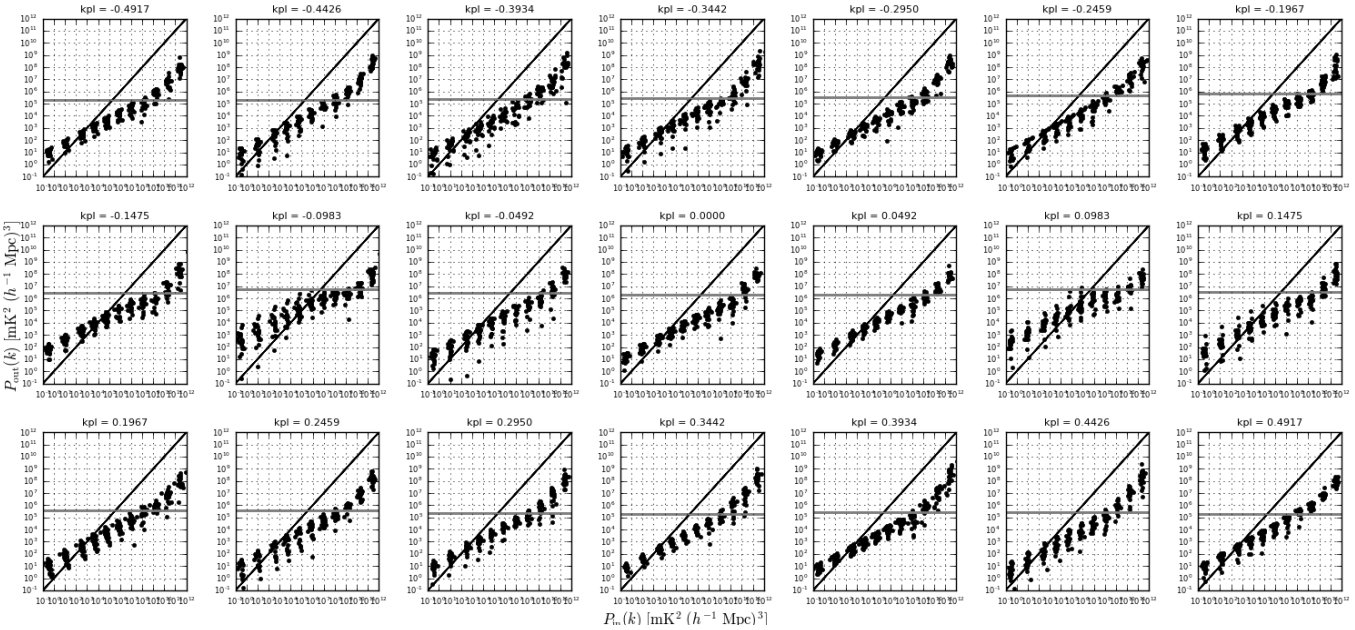


**Figure 13**. $P_{in}$ vs. $P_{out}$ (black) for 15 injection levels, 20 bootstraps, and 21 $k$'s. **[Plot a semi-transparent grey range of pCv values instead of just the max]**

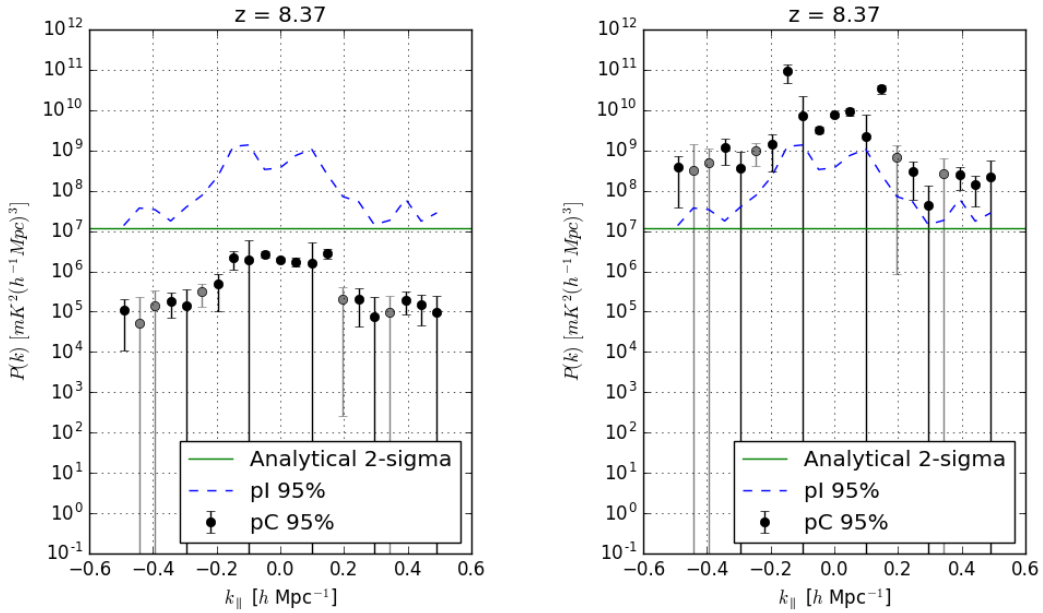


**Figure 14**. Full inverse covariance weighted power spectrum of PAPER-64 data (black and grey points, with $2\sigma$ error bars) before signal loss correction (left) and after (right). Black points correspond to positive values, while grey points correspond to originally negative values that have been made positive for plotting. The dashed blue line is the unweighted power spectrum ($2\sigma$ upper limit). The solid green line is the theoretical noise level prediction based on observational parameters.

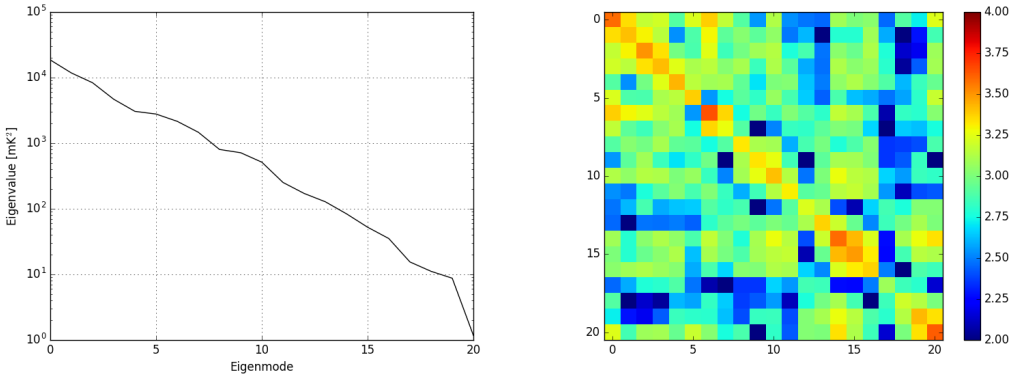


**Figure 15**. Eigenspectrum for **C** for baseline (1,4) for the 21 channels ofs interest (left) and covariance matrix **C** for the same baseline (right).

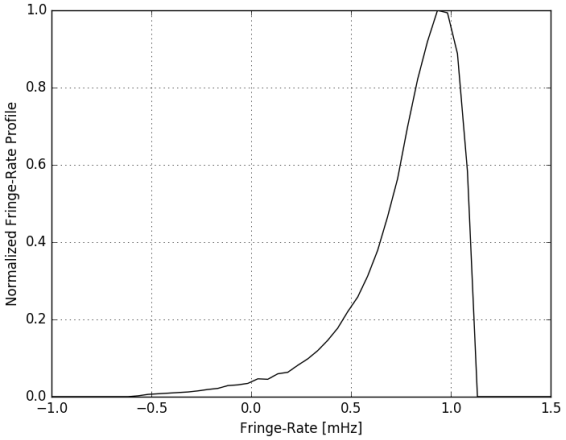


**Figure 16**. The optimal fringe-rate filter used in the analysis, normalized to 1.