PHSX815_Project3: Characterizing instrumental noise

Ryan Low

March 2021

1 Introduction

Modern astronomy relies on Charged Coupled Devices (CCDs) and other such imaging sensors for recording astronomical data. All of these technologies rely on photons exciting the electrons in some semiconducting material. Counting those electrons becomes a proxy for the number of photons detected. Because of this, recording astronomical data is a counting problem, and thus we can expect the number of photons recorded on a CCD to be distributed as a Poisson distribution. As with all electronic measurements, we must also be aware of sources of noise. Since CCD noise comes from electron counts, we can also expect it to be distributed as a Poisson distribution. The determination of λ_{noise} is an important problem that all astronomers face. It is often told that a good rule of thumb for determining λ_{noise} is to take the per-pixel median of the calibration images. We will investigate whether this rule of thumb holds in practice.

2 Problem Statement

We would like to determine the best-fitting λ_{noise} given a set of data. Because we are dealing with a semiconducting system, how the electrons are distributed in energy depends on the Fermi-Dirac distribution (Equation 1).

$$P\left(E\right) = \frac{1}{1 + \exp\left(\left(E - E_F\right)/k_B T\right)} \tag{1}$$

For silicon, the band gap is about $1.12\,eV$ and the Fermi energy is approximately half of the band gap energy. Using this distribution, we can model the number of noise electrons that we count, which in turn gives us a distribution for λ_{noise}. Using this, we will generate sample data for our detected counts (see Section 3).

Using the generated data, we must be able to infer λ_{noise}. The relationship between the probability distribution of the data given λ is related to probability distribution of λ given the data by Bayes' theorem. The posterior probability distribution is proportional to Equation 2.

$$P\left(\lambda|x\right) \propto \frac{\lambda^{x}e^{-\lambda}}{x!} \tag{2}$$

Here, we do not know λ. Instead, we can input the data into Equation 2 and calculate the Log-Likelihood (Equation 3) as a function of λ.

$$\mathrm{LL}\left(\lambda\right) = \sum_{i} \log\left(P\left(\lambda|x_i\right)\right) \tag{3}$$

The value of λ that maximizes the LL will be our best estimate of λ. When viewed as a function of λ, terms in the LL that don't explicitly depend on λ are just normalizations and therefore don't need to be considered in the maximization process. The LL can therefore be written as

$$\mathrm{LL}\left(\lambda\right) = \sum_{i} x \log\lambda - \lambda$$

We numerically maximize this function to find $\lambda_{\hat{noise}}$, the best estimate of λ_{noise}.

3 Algorithm Analysis

To generate the sample data, we perform Gibbs sampling. In Gibbs sampling, this integration is approximated by taking intermediate samples of these nuisance parameters according to their own distributions. Then, using those samples, we generate samples of our rate parameters and feed those rate parameters into a Poisson distribution. This final set of Poisson-distributed samples is a simulated set of data with these nuisance parameters integrated out. We use this when constructing the simulated data for the Log-Likelihood Ratio (LLR) of each model.

So that our computation time remains reasonable, we use `numpy` methods wherever possible. This includes both array operations and random number generation. `numpy` methods are faster than their pure-Python counterparts because they pass execution to an underlying `C` implementation. Since compiled `C` code is much faster than interpreted Python code, using `numpy` affords greater computational speed, which allows us to get away with some less efficient methods.

In our noise model, a noise electron is counted if it is excited into the conduction band. For our purposes, this occurs when the electron's energy is above the Fermi level. The probability that an electron has this energy is

$$P_{detected} = \int_{E_f}^{\infty} P\left(E\right) dE$$

We can easily perform this integral numerically, and do so using Monte Carlo integration. Since the tail probability in Equation 1 is extremely small, it is sufficient to just integrate up to a reasonable upper bound, in our case $\left(E - E_f\right)/k_BT = 1$. Once this probability is calculated, we can produce a uniformly-distributed number from 0 to 1 for each free electron in the pixel and decide whether the electron is excited or not. The total number of excited electrons is our noise rate parameter, λ_{noise}. For our purposes, we will assume that the number of free electrons is fixed.

Although the LL is a one-parameter function, it is a complicated one. Therefore, we will numerically optimize the LL to find λ. Equivalent to maximizing the LL is minimizing the negative LL. Therefore, we can take advantage of all the neat numerical minimization technology that exists. We use the `scipy` optimization package to perform numerical minimization and obtain uncertainties. `scipy.optimize.minimize` gives the optimal value of λ_{noise}, and the inverse Hessian matrix of the fit. Since this is a one-parameter fit, the inverse Hessian matrix will only hold one value and will be the estimate of the variance.

4 Results

A typical observing run can afford about 30 calibration frames for characterizing noise. Simulating 30 measurements per experiment for 100 experiments, we estimate the noise rate parameter. To see how well the median characterizes the error, we can plot the value of $\hat{\lambda_{noise}}$ against the median for each experiment. If the median is a good estimate of $\hat{\lambda_{noise}}$, then the resulting scatter should align along the line $y = x$. Equivalently, plotting the difference between $\hat{\lambda_{noise}}$ and the mean should result in a scatter about the line $y = 0$. With $T = 300\,K$, we present plots of both cases in Figure 1. From either of those figures, we see that the median generally underestimates $\hat{\lambda_{noise}}$. The mean difference is 0.405 counts per second. Compared to the mean error on $\hat{\lambda_{noise}}$ of 0.729 counts per second, we see that the median also gives a reasonable estimate of the noise. We repeat the analysis with $T = 77\,K$ to simulate the detector being cooled to liquid nitrogen temperature. The result is presented in Figure 2. Again, the median underestimates $\hat{\lambda_{noise}}$, this time with a mean difference of 0.411 counts per second. Compared with the mean error on $\hat{\lambda_{noise}}$ of 0.430 counts per second, the median still is a good estimate of the error.

5 Conclusions

By simulating the noise detected by a CCD, we were able to estimate the Poisson rate parameter of the noise distribution by maximizing the likelihood. With this, we were able to show that the median of the data estimated $\hat{\lambda_{noise}}$ reasonably well. This means the astronomer's rule of thumb of using the median of the data to estimate the error is a reliable rule of thumb and may not significantly affect the results of astronomical data analysis if the detected noise is Poisson distributed.

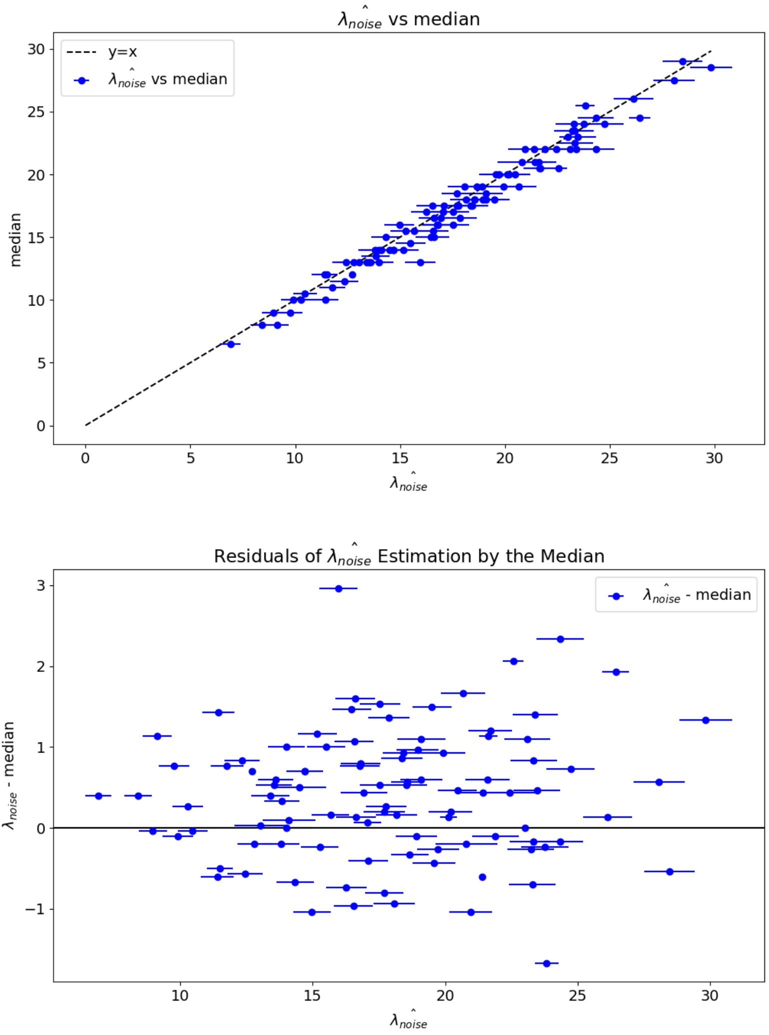

Figure 1: Top: Plotting $\hat{\lambda_{noise}}$ vs the median for $T = 300\,K$. Bottom: Plotting the difference between $\hat{\lambda_{noise}}$ and the median against $\hat{\lambda_{noise}}$.

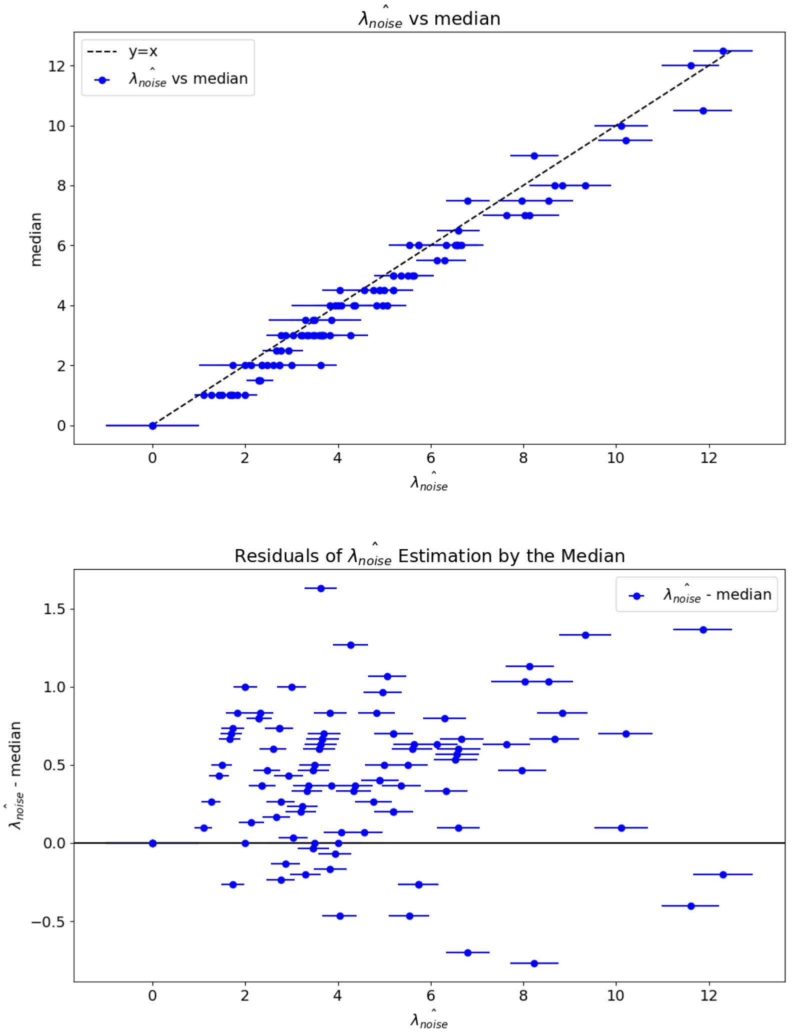

Figure 2: Left: Plotting $\hat{\lambda_{noise}}$ vs the median for $T = 77\,K$. Right: Plotting the difference between $\hat{\lambda_{noise}}$ and the median against $\hat{\lambda_{noise}}$.