ASTR 288C – Lecture 4

Monday, 28 September 2009

Signal Detection and Statistics

Introduction

Source detection involves finding a signal in a sea of noise. In general instruments are designed to be as sensitive as possible, but this high sensitivity has the side effect of making it difficult to distinguish between noise and faint sources. Photon-counting detectors, such as *Swift*'s *X*-Ray Telescope, can detect individual photons, which means that very faint sources can be detected. Unfortunately, having a small number of photons makes it difficult to determine which photons are from the source of interest and which are simply background noise. The idea is to be able to detect a signal due to a small number of photons from a real source, even if it is buried in a large amount of noise, but this is not easy. Astronomical sources are usually intrinsically faint due to their large distances, so it is important to make every photon count.

The details of detecting a signal depend on several factors. Some of these are listed here.

- The wavelength band of the observations.
    - The type of data that can be collected is different in different wavelength bands. For example, it is not possible to focus radio waves onto an imaging detector, so direct images of the sky are not possible. Spectra, however, are possible at radio wavelengths.
- The instrument that is being used to detect the photons.
    - Different instruments use different physical processes to detect and measure photons, so the data needs to be handled in different ways.
- The intensity of the source relative to the background.
    - The number of photons affects how the data is handled.
    - For example, the *X*-ray sky background is very low, so bright *X*-ray sources are dominated by photon noise. At optical wavelengths, however, the sky background is usually bright, so noise from the background can be significant even for bright sources.
- Prior knowledge of the intensity and shape of the source.
    - Knowing what you are looking for makes it easier to find the sources.

Signal detection is a statistical process that depends on understanding the statistical properties of the data that you are interested in.

Astronomical images contain typically a large set of point-like sources (stars, asteroids), some quasi-point-like objects (faint galaxies, double stars), and some complex, diffuse structures (galaxies, nebulae, clusters, etc.). These objects are often hierarchically organized: a star in a small nebula, which in turn is embedded in a

galactic arm, which is part of a galaxy, and so on. This hierarchical nature of structure makes it difficult to define a single procedure to identify all types of sources in an image. A process that works well for point sources, for example, may not work well for extended sources such as galaxies.

We will concentrate on the basic principles behind detecting point sources in two-dimensional astronomical CCD images.

Statistics

Since source detection is a statistical process there are some basic statistical concepts that need to be understood.

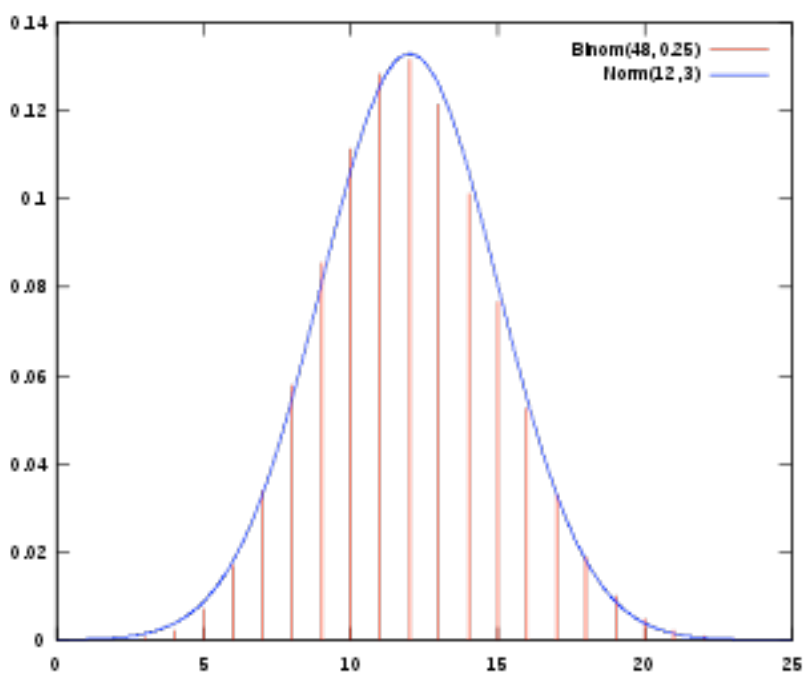
Mean

There are several statistical quantities that are often called the "mean". Here the term mean will refer to the arithmetic mean of a set of data. The arithmetic mean, $<x>$, is the sum of the values of the data, x_i, set divided by the number of data points in the set, N. The definition of $<x>$ is

$$<x> = \Sigma_{i=1,\mathrm{N}}\, x_i$$

where each data point, x_i, is independent.

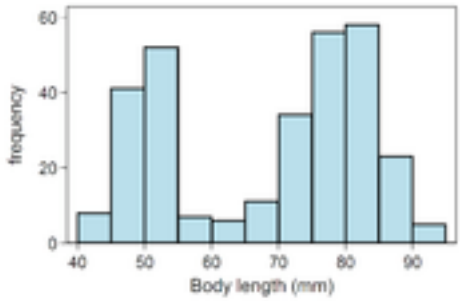
The mean works well for a symmetric distribution, or a nearly symmetric distribution such as a Gaussian.



The mean is often used as a typical value for data set, but it can be biased by outliers. For example, if a neighbourhood has 15 pet dogs, 12 pet cats, and an

elephant the mean weight of the pets in the neighbourhood will be about 350 lbs. However, none of the animals has a weight near this. A more typical weight would be about 20 lbs with the elephant (at about 9700 lbs) being a significant outlier. In this case the mean is not a good representative value for the set of pets.

The mean can also fail if the distribution has multiple peaks.



Median

This is the middle value in a sorted set. The median is strongly immune to most outlying values. For example, the median of {1,2,3,4,5,6,57} is 4, while the mean is 11.1. Medians are good for estimating typical values if the data contains outliers, or a highly asymmetric distribution of values. A disadvantage is that medians can be computationally intensive since the data needs to be sorted.

Mode

This is the most common number in a set. For example, the mode of {1,2,2,3,3,3,4,4,4,4} is 4 (while the mean and median are 3. The mode is not sensitive to outliers and can work well for a multi-peaked distribution. It tends to return a value that is fairly representative of the data set. The mode can be significantly different from the mean or the median if the distribution of data is strongly skewed, and not all data sets have a mode. For example, the mode of {1,2,3,4,5} is undefined. The mode can be computationally intensive to compute, but there are computational tricks, if the underlying distribution is know, that can make computing the mode very fast.

Standard Deviation

The standard deviation is a measure of how much variation there is in the values in a data set. It is defined as, σ, where

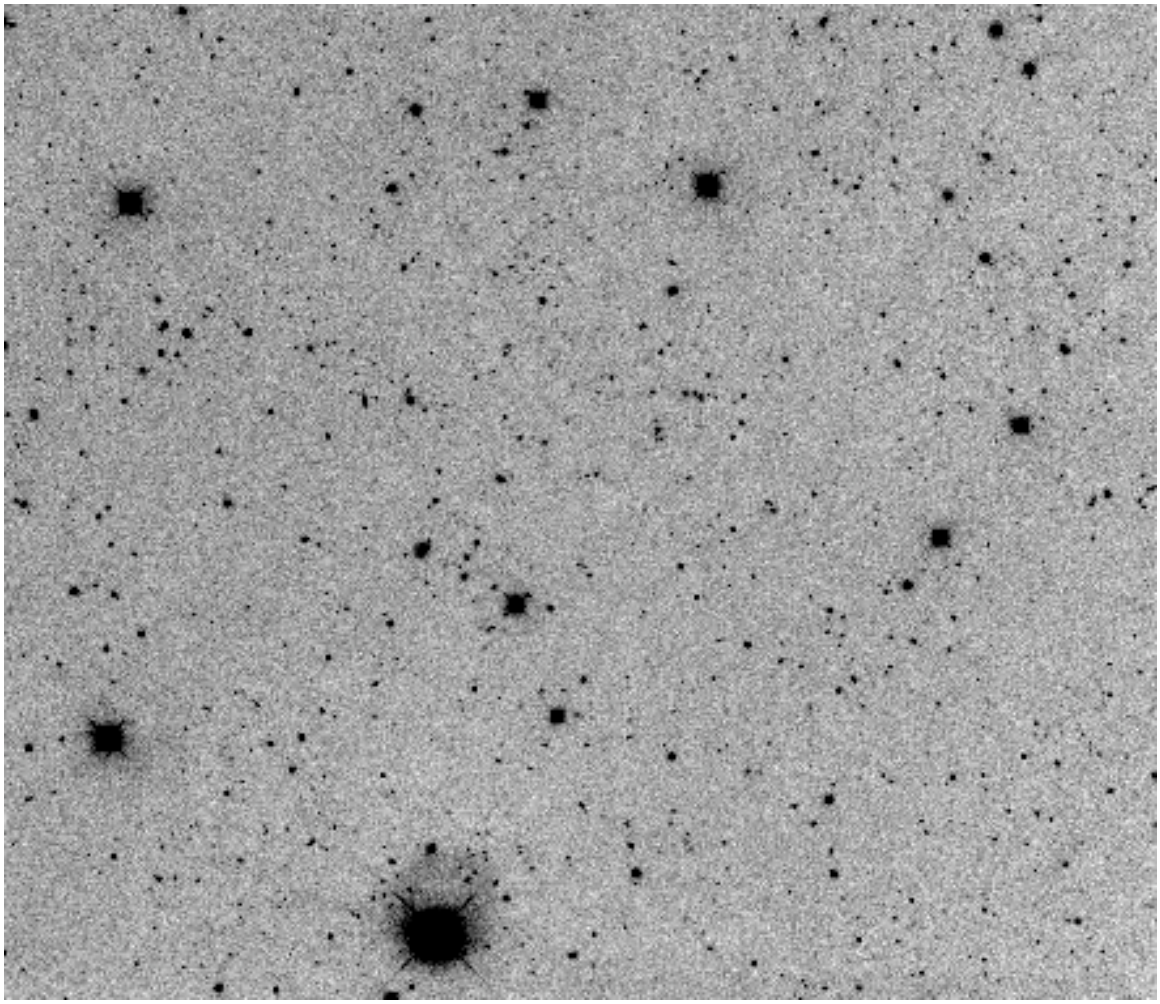
$$\sigma^2 = 1/(N-1)\ \Sigma_{i=1,N}\ (\ x_i^2 - \langle x \rangle^2\ )\ .$$

The standard deviation assumes that values are symmetrically distributed around the mean value. This is a reasonable assumption for CCD images since the noise has a Gaussian distribution. However, some detectors give data where the noise has a Poisson distribution, and in some cases the noise can have a binomial distribution. It is important to determine what the distribution of noise is for your detector in

order to do source detection in a statistically rigorous way. In general CCDs have a Gaussian distribution.

Source Detection

The techniques used for source detection are different for different types of signals and different types of detectors. However, the basic principles can be illustrated by using a CCD image of the sky. Let's assume that we have a deep image of a field of stars taken with the Nordic Optical Telescope. Let's imagine that it looks like this.



This image is a 300 s *I*-band exposure of the field containing the gamma-ray burst GRB 0590416A. It was taken with the Nordic Optical Telescope's DFOSC/FASU instrument on 20 Apr 2005 starting at 01:01:00 UTC. The data has been preprocessed to create a science-quality image. The read-out noise (the noise that is added to the data by the process of reading it off the detector) is 3.3 electrons per pixel and the gain (the number of incoming photons that correspond to one count on the detector) is 13.29 electrons per count.

This image contains a large number of sources, and a visual examination suggests that most of them are either stars, or some other point-like sources such as nucleated galaxies. Notice that there is a distribution of the brightness of the sources ranging from bright enough to saturate the detector (the bright star near the bottom of the image) to faint enough that it is not possible to visually distinguish real sources from the noise in the image.

This is not a highly crowded field. Most of the sources are isolated from each other and can be easily identified. There are even areas of the image that appear to contain no sources. A close examination of these areas shows that there even though there are no obvious sources there are still intensity fluctuations in image. These fluctuations are due to:

- **Unresolved sources**. These are sources that are fainter than the detection limit of the detector. They are too faint to be easily distinguished by eye, and source detection algorithms can not distinguish them from noise. The number of sources in an image increases as they get fainter, so there will be a large population of these unresolved sources that are too faint to detect, but still contribute to the background. Unresolved sources will add to the pixel-to-pixel fluctuations, and are a significant source of sky photons.
- **Noise**. There is statistical noise in each CCD pixel. There are several sources of noise in a CCD image

  1. **Poisson Noise**. This is the intrinsic noise due to photon counting being a Poisson process. The mean Poisson noise in each pixel is equal to the square root of the number of photons from the source in that pixel. This is a fundamental physical limitation on counting photons and represents the absolute minimum noise in an astronomical image. Because of noise the observed number of photons measured for a source is *not* the true number of photons from the source. The observed number is the sum of the true source photons plus (or minus) the Poisson noise photons. Therefore, one can not simply take the square root of the observed counts in a pixel to determine the noise in that pixel.
  2. **Sky Noise**. There are background photons coming from the sky. These are independent of the photons from the source, and have their own Poisson noise.
  3. **Instrument Noise**. This is noise that is introduced into the image by the CCD electronics, and is often called read-out noise. Modern CCD detectors have very small read-out noise, but it can be the dominant source of noise for very short exposures.
  4. **Flat-Field Variance**. This is noise that arises when the detector sensitivity varies slightly from pixel to pixel. It is usually about 1% or 2% of the total counts in a pixel.

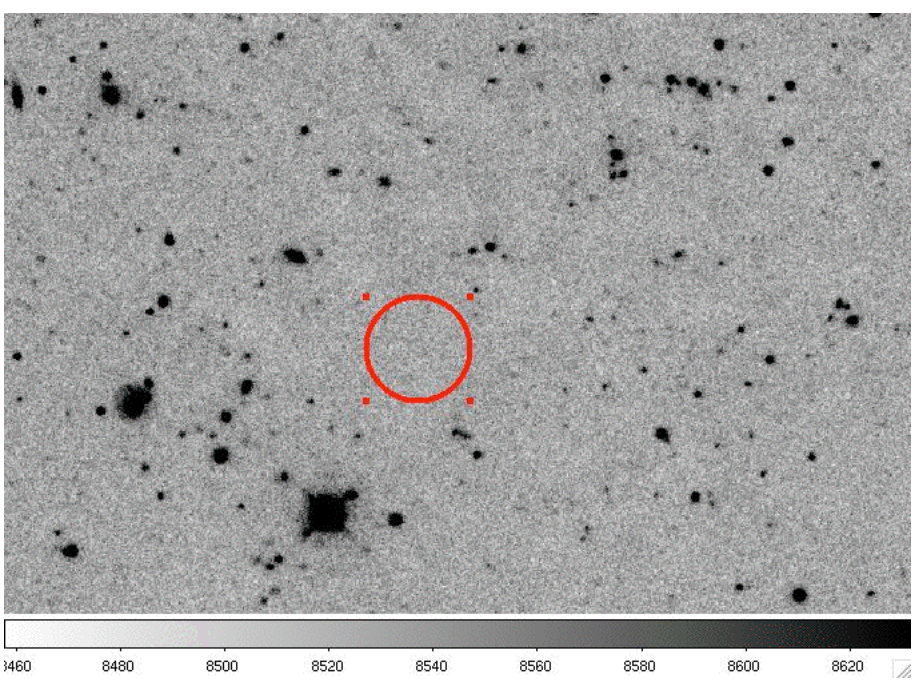
  The total noise in each pixel is the quadratic sum of the noise from each source of noise.

In order to find sources in an image one first needs to understand the noise properties of that image. In order to do that one must first estimate the background of the image.

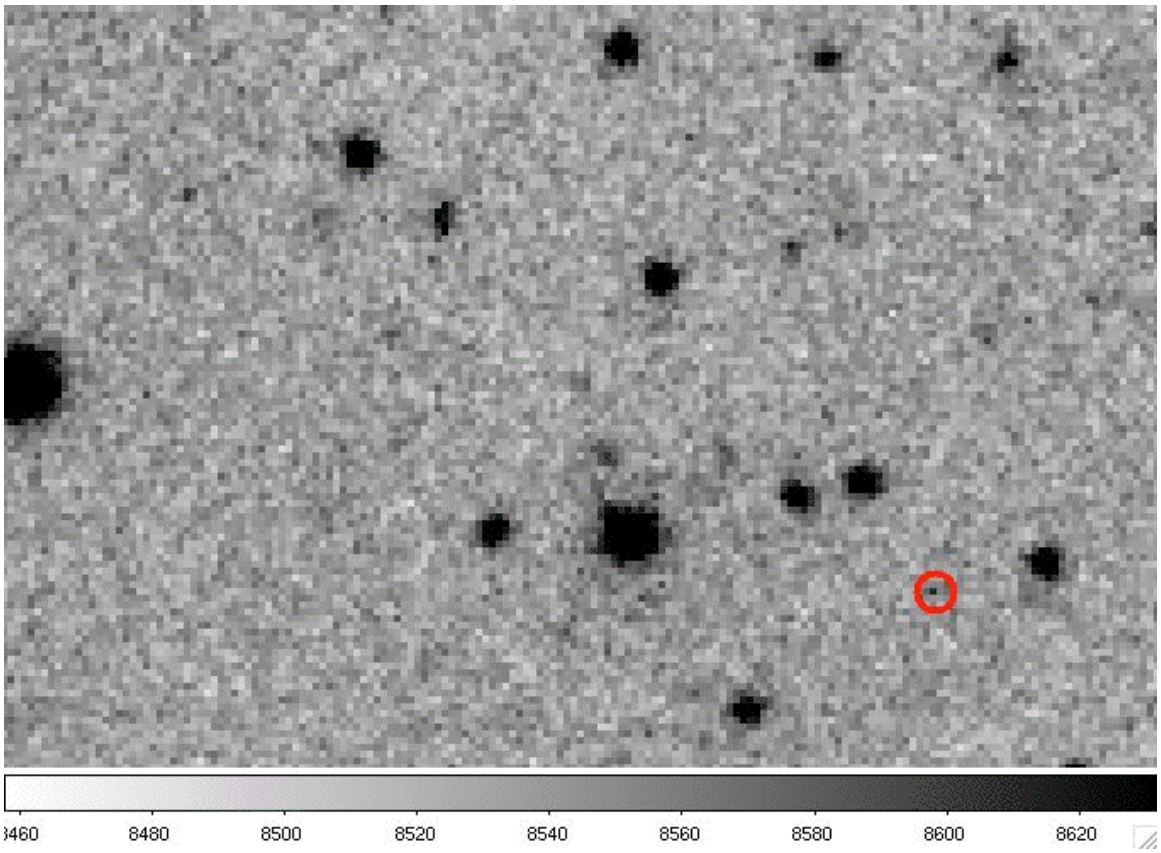
Measure the Background

Detecting sources involves finding all sources that are brighter than the background at some well-determined level of significance. The first step in doing this is to measure the value of the background. If the background is approximately constant across the image then one can simply take the typical pixel value in a region where there are no sources. If the background varies then more complex techniques need to be used, such as computing local background values. There are many different ways of computing a global background, and none of them is valid for all images. The simplest method is to examine the image and search of an area where there are no sources. In general the mean is not a good statistic to describe the value of the sky. It is easily biased by faint sources, which leads to the value of the sky being overestimated. A better sky statistic is often the mode, which provides an estimate of the most common value of a sky pixel. The mode is not biased by faint sources, since it automatically ignores the brightest pixels. Most modern source detection software uses information about the noise properties of the image to compute an adjusted mode that tries to compensate for the skew in the distribution of sky values caused by contamination from astrophysical sources and noise photons. In practice the sky background is often determined locally. This is important if the sky is not constant across the image. The process of determining the sky background can be automated by subdividing the image into a grid of regions and computing the modal value for the sky in each region.

Once you have a measurement of the sky background, measure the standard deviation of the pixel values at the location that the sky was determined.



Once the typical sky value and standard deviation have been determined the next step is to filter the image in order to smooth over the noise. Filtering is done so that individual noise spikes are not treated as detections. It is important that the size and shape of the filter is chosen so that filtering does not smooth out any of the sources in the image. The size and shape of the filter need to be picked with care, and usually matched to the individual image, and perhaps the type of source that one is interested in. Consider the sub-image below.

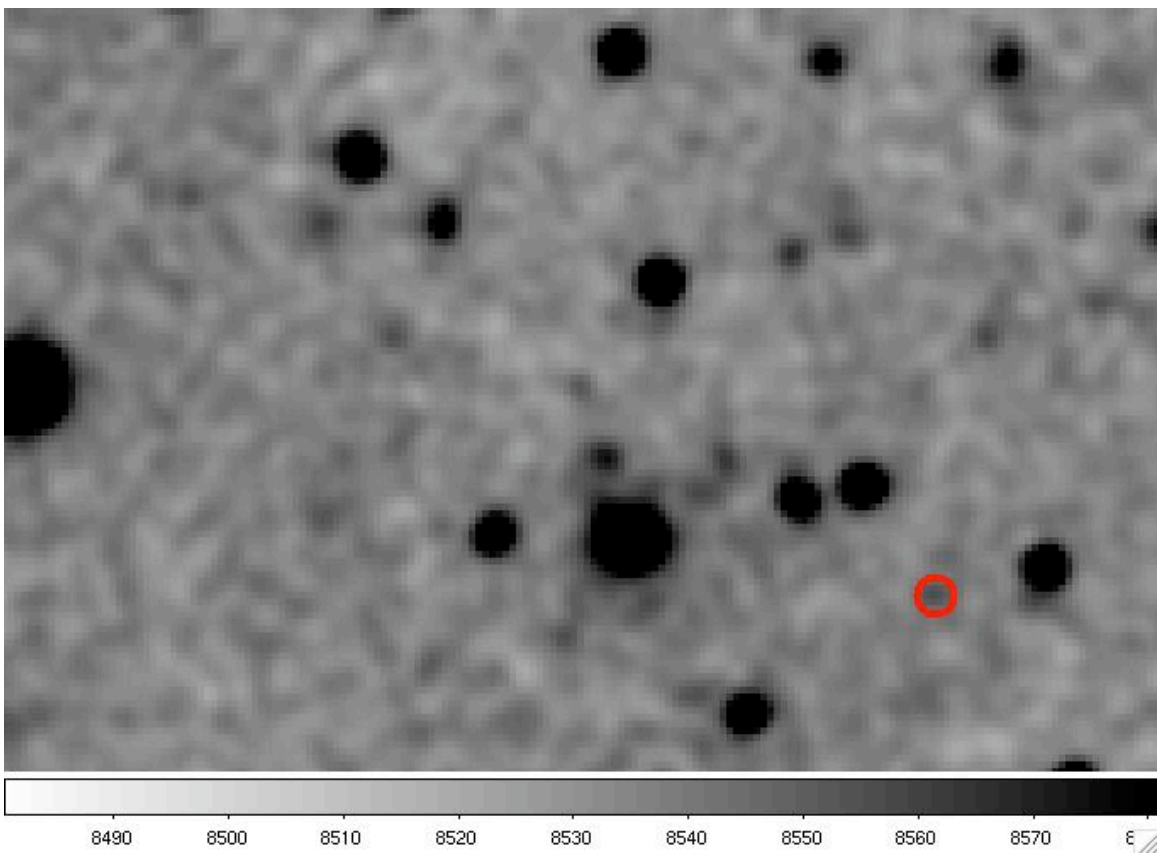


The pixel inside the circle in the lower right of the sub-image is a high positive noise pixel. It is sufficiently brighter than the background that it could be mistaken for a source by an automated source detection algorithm. However, we know that it is not an astrophysical source because none of the neighbouring pixels are enhanced relative to the background. We want our filter to smooth over this, but not any of the real sources in the image.

The size and shape of the filter depends on the nature of the sources that you are interested in detecting. The smallest source in an image will be a point source. Point sources are sources that are too small to be resolved by the telescope's optics and detector. Stars are a good example. A one Solar mass star has a typical diameter of 1.4×10^6 km. At a distance of 10 pc this corresponds to an angular diameter of approximately 0.001 arcsecond. The diffraction limit for the *I* band (with a central wavelength of about 8060 Å) is $\theta = 1.22\lambda/D$ rad, where λ is the wavelength of the light and D is the diameter of the mirror (2.66 m for the NOT). This gives a

diffraction limit of approximately 0.08 arcsecond. In addition, the pixel scale of the DFOSC CCD is 0.39 arcsecond/pixel, which effectively sets the minimum size that can be resolved on this CCD detector. Finally, atmospheric turbulence acts to spread out photons from a point source as they travel through the atmosphere. This spreading varies rapidly with atmospheric conditions and needs to be measured for each image. In the case of the data shown here the width of the point-spread function is 1.2 arcseconds. Any source that appears to be smaller than this in an image can not be an astrophysical source, but must be some form of noise.

To smooth the noise in an image run a filter over the image. To optimize for detecting point sources the filter should have the same size and shape as a point source. To a first approximation point sources on CCD images are approximately circular with a Gaussian profile. This is not true in detail, but it is a reasonable approximation for source detection. The width of the Gaussian filter should approximately match the width of the point-spread function. The following figure shows the CCD image after smoothing it with a Gaussian filter with a width that matches the seeing of the image.



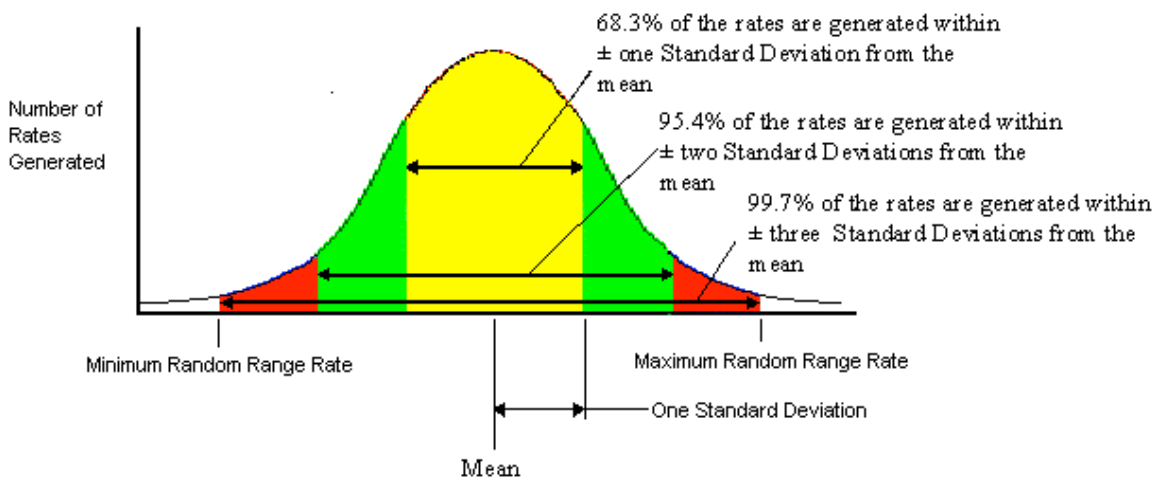
Notice that the high pixel (circled) has been smoothed over. There is still a local background enhancement, but it is significantly fainter than the true sources in this image.

Now that noise pixels have been effectively removed from consideration the image can be scanned for sources that are significantly brighter than the smoothed

local background. This is done by automated software such as SEXTRACTOR or DAOPHOT. The details of how individual source detection software packages do source detection vary from package to package, but in general the procedure is as follows:

- Filter the image to suppress individual noise spikes.
- Scan the image to find peaks relative to the local smoothed background.
- Record the height of each peak above the local background.
- If the peak is more than *N* sigma above the local background it is considered to be a source candidate.
- If there are more than *M* adjacent pixels about the threshold then it is considered to be a source.

For a Gaussian or Poissonian distribution of noise most of the pixel values will be the very similar to the typical sky value. The number of pixels that deviate from this value will decrease the further a pixel value gets from the sky value.



For Gaussian noise 68.3 percent of the sky photons will have values within 68.3% of the mean (1 standard deviation, or 1 sigma), 95.4% will have values within 2 sigma, and 99.7% will have values within 3 sigma. In other words, only 0.3% of the peaks that are more than 3-sigma brighter than the background will be due to noise. This means that there is a high probability (0.997) that any peak that more than 3 standard deviations brighter than the background is a real source. So, in order to identify real sources and discard noise spikes one can set the detection threshold to 3 sigma above the value of the sky background. This will ensure that approximately 99.7% of the detected sources are real. The probabilities of a false detection for different values of *N* are given in the Table below.

***N*-sigma**	**Chance of False Detection**
1	31.7%
2	4.6%
3	0.3%

In practice the fact that there are multiple sources of noise in an image, and that there will be a population of faint sources that are less than 3 sigma above the background, will bias the source detection towards finding more than 0.3% false positives for a 3-sigma detection threshold. One way to compensate for this is to require that there be at least *M* adjacent pixels that are brighter than *N*-sigma above the background for a source to be considered real. For example, in the following grid of pixels there is a candidate detection (X) in the central pixel. It is surrounded by five pixels that are brighter than *N*-sigma (O). If the adjacent pixel threshold is set to *M* = 5 then the first diagram below would be considered to be a real source, but the second diagram below would *not* be considered to be a detection.

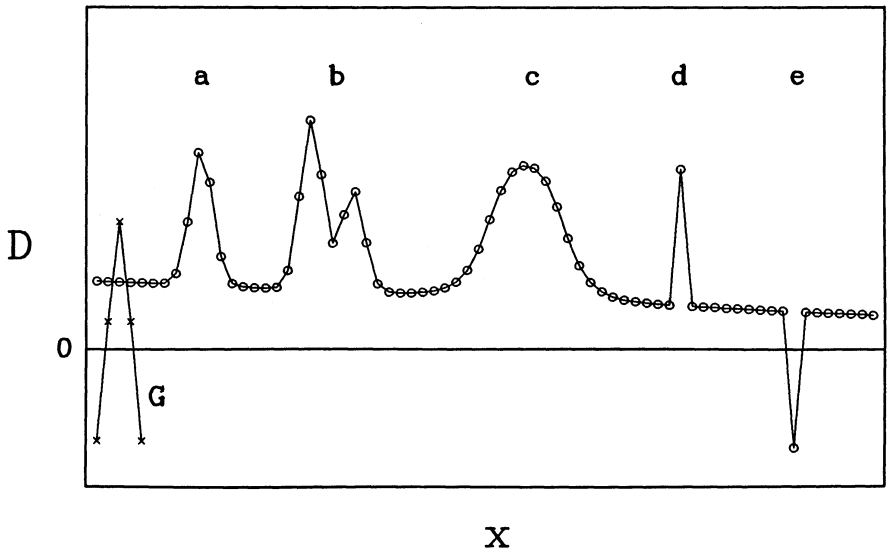
	O	O	O	
		X	O	
		O		

This is a detection.

	O			
		X	O	
		O		

This is not a detection.

Here is an example of several sources situated on a sloping background. The data has been reduced to one dimension for clarity



a. This is a star.
b. This is two stars that are blended. The point-spread functions of the two stars overlap.
c. This is a galaxy that is slightly larger than the point-spread function. It is only barely resolved.
d. This is either a noise spike, or a cosmic ray.
e. This is a bad (cold) pixel.

Lab Work

The purpose of this lab activity is to do source detection on an astronomical CCD image. You will need to log into your unix account and `cd` to your astr288c directory, or to whichever directory you prefer to work in. Next, download the data for this lab project and put it in your working directory.

1. Log into your unix account and start X Windows. Open a terminal window.
2. Use a Web browser to download the data for this lab from http://lheawww.gsfc.nasa.gov/~sholland/astr288c/autumn_2009/lectures/lecture4.tar.gz .
3. Move these files to your working directory.
4. Uncompress and untar this file using the following unix commands
   a. > `gunzip -v lecture4.tar.gz`
   b. > `tar xvf lecture4.tar`

   This should give you the following four files:
   - grb050416a.fits — the CCD image
   - grb050416a.sex — SEXTRACTOR's parameter file
   - catalogue.param — a file that describes how SEXTRACTOR writes its output, **do not edit this file**
   - filter.conv — the filter used to smooth the data, **do not edit this file**

   Check that these four files exist (use the `ls` command).

The purpose of this exercise is to determine how the source detection threshold affects the number of sources that are detected. The aim is to determine the optimal value for the source detection threshold parameter in SEXTRACTOR. To do this you need to edit the grb050416a.sex file, and run the SEXTRACTOR program for several values of the source detection parameter.

The source detection threshold parameter in SEXTRACTOR is called **DETECT_THRESH**, and it is set in the grb050416a.sex file. You will run SEXTRACTOR several times with various values of DETECT_THRESH and record the number of detections that are made for each value. Make a plot of detections as a function of threshold. Determine what value of DETECT_THRESH gives the best balance between finding the most faint sources and rejecting the most spurious noise detections. If DETECT_THRESH is set too high then SEXTRACTOR will miss the faint sources. If DETECT_THRESH is set too low then SEXTRACTOR will mistake noise spikes for faint sources. In practice there is no sharp boundary that guarantees that all the faint sources will be detected and all the noise spikes will be discarded. However, there will be a minimum value of DETECT_THRESH, below which the number of noise spikes that are detected starts to increase dramatically. This threshold is the optimum detection threshold. Picking it can be a bit of a dark art. In general the threshold is approximately the value of DETECT_THRESH where the curve changes from linear growth to exponential growth.

Procedure

Edit the grb050416a.sex file by opening it in a text editor. The contents of the file should look like this.

```
# Default configuration file for SExtractor 2.3.2
# EB 2004-08-02
# Set for Apr19_I_m.fits
#

#------------------------------- Catalog ----------------------------------

CATALOG_NAME    grb050416a.cat     # name of the output catalog
CATALOG_TYPE    ASCII_HEAD         # "NONE","ASCII_HEAD","ASCII","FITS_1.0"
                                   # or "FITS_LDAC"

PARAMETERS_NAME catalogue.param # name of the file containing catalog contents

#------------------------------ Extraction --------------------------------

DETECT_TYPE     CCD          # "CCD" or "PHOTO"
FLAG_IMAGE      flag.fits    # filename for an input FLAG-image
DETECT_MINAREA  5            # minimum number of pixels above threshold
DETECT_THRESH   8            # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
ANALYSIS_THRESH 1.5          # <sigmas> or <threshold>,<ZP> in mag.arcsec-2

FILTER          Y            # apply filter for detection ("Y" or "N")?
FILTER_NAME     filter.conv  # name of the file containing the filter

DEBLEND_NTHRESH 32           # Number of deblending sub-thresholds
DEBLEND_MINCONT 0.005        # Minimum contrast parameter for deblending

CLEAN           Y            # Clean spurious detections? (Y or N)?
CLEAN_PARAM     1.0          # Cleaning efficiency

MASK_TYPE       CORRECT      # type of detection MASKing: can be one of
                             # "NONE", "BLANK" or "CORRECT"
```

There are many parameters in this file (and many more that have been left out of this version of the file), but they can all be ignored for now. The only parameter that will be used in this project is **DETECT_THRESH**. This is the value of the detection threshold in units of sigma above the background. You will need to edit this value to change it from "8" to whatever value you want to use. You will need to try values between about 1 and 8.

Run SEXTRACTOR using the following command

```
> sex grb050416a.fits -c grb050416a.sex
```

The first argument tells SEXTRACTOR what image to use (grb050416a.fits). The −c flag tells SEXTRACTOR to read various run-time parameters from grb050416a.sex. SEXTRACTOR will run quickly and write a few lines to the screen. The output will look a bit like this

```
wcne-2-144-244 289> sex grb050416a.fits -c grb050416a.sex
----- SExtractor 2.5.0 started on 2009-02-20 at 12:00:44 with 1 thread

Measuring from: "grb050416aI"  / 1688 x 1447 / 32 bits FLOATING POINT data
(M+D) Background: 8530.37   RMS: 15.86     / Threshold: 79.3002
Objects: detected 538     / sextracted 516
> All done (in 1 s)
```

The program outputs to the screen some information about the noise properties of the image, and the number of sources that it has detected (highlighted in yellow in the example above). In this case SEXTRACTOR found (sextracted) 516 sources (you may get a different number). *This is the number that you need to record.*

Run SEXTRACTOR for several values of DETECT_THRESH between 1 and 8. You may wish to experiment with values outside of this range. DETECT_THRESH can be a decimal number, so values such as 2.5 or 6.67 are allowed. For each value of DETECT_THRESH record the number of sources detected (sextracted). Make a plot with DETECT_THRESH on the x-axis and the number of detections on the y-axis. Identify the value of DETECT_THRESH that will give the optimum balance between detecting faint sources and rejecting noise spikes. Explain why you picked this value.

Hand in your plot and your explanation with your homework.