MockGals

Make mock stars and galaxies in a FITS image with noise.
Manual for mockgals version 0.1

Mohammad Akhlaghi

This manual is for `mockgals`, a program to make mock astronomical objects in a FITS image and add the appropriate noise.

Table of Contents

1 About this manual

To get started with MockGals please go to the next chapter.

The following sections are more general discussions. In the first section we will give a short review of the way modern science has become intertwined with software and how important it is for scientists to have very good understanding of the software they use. After that you rights as the reader of this document and user of MockGals are explained and in the end of this chapter we give a short review of how this manual is available for your use and a short summary of the manual.

1.1 Good science

We believe that if a software (a technique) is to be used in scientific research, it has to be open source and open to scrutiny. So any critical/interested user can easily follow all the steps and understand all their details and assumptions. No software is perfect and there are ultimately some flaws in any software that allow future generations to find them and advance our experimental and theoretical knowledge of the universe we live in. With the progress of technology, scietific results are increasing becoming more and more dependent on hard/software techniques. Therefore obscure software is arguably the biggest roadblock in our aim to acheive this goal and a modern scientists cannot claim to be interested in understanding the universe and our relation to it, what ever their specific field, without accurately understanding the software techniques used.

To reach this goal, namely to let the users of a software be able to understand its intricate details at their will, a software has to be accompanied by a good and up to date manual. The user of a software cannot claim to understand how it works only based on the experience they have gained in using it a lot or on mock simulations, that kind of subjective experience is prone to very serious mis-understandings that only help in producing dogmas. Unfortunately in the commonly used astronomical software, most are either closed source or don't have a good manual. The scientific results that are deduced from such software are thus not objective, because in the former case scientists have to look at it as a black box and trust its results on faith. In the latter, it is very hard to find enough time to understand the programming of an open source project with out a clearly written manual explaining all the ideas behind the major functions, data structures or objects. It is certainly time consuming for the programmer to make such a manual, but if the greater common experience is considered, that time put is very nicely invested. Because through sharing of these fundamental ideas that shape all our scientific results and keeping them open to criticism, we can make them more robust and thus take better steps in our endeavor to quench our curiosity in nature.

Bjarne Stroustrup (creator of the `C++` language) says: "Without understanding software, you are reduced to believing in magic". Ken Thomson (the designer or the Unix operating system) has said very nicely that "I abhor a system designed for the "user" if that word is a coded pejorative meaning "stupid and unsophisticated"." Both these cases naturally apply more seriously to systems written for scientific applications and as scientists we should be loyal to such principles if we want to remain critical and be objective. After all, all our data and models are produced, archived and processed using hard/software. We cannot and should not accept any result through faith in the programmer who wrote that particular program we are using to publish our papers.

1.2 Your rights

mockgals is "free software"; this means that everyone is free to use it and free to redistribute it on certain conditions. mockgals is not in the public domain; it is copyrighted and there are restrictions on its distribution, but these restrictions are designed to permit everything that a good cooperating citizen would want to do. What is not allowed is to try to prevent others from further sharing any version of Texinfo that they might get from you.

Specifically, we want to make sure that you have the right to give away copies of the programs that relate to mockgals, that you receive source code or else can get it if you want it, that you can change these programs or use pieces of them in new free programs, and that you know you can do these things.

To make sure that everyone has such rights, we have to forbid you to deprive anyone else of these rights. For example, if you distribute copies of the mockgals related programs, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

Also, for our own protection, we must make certain that everyone finds out that there is no warranty for the programs that relate to mockgals. If these programs are modified by someone else and passed on, we want their recipients to know that what they have is not what we distributed, so that any problems introduced by others will not reflect on our reputation.

The precise conditions of the licenses for the programs currently being distributed that relate to mockgals are found in the General Public Licenses that accompany them. This manual is covered by the GNU Free Documentation License (*note GNU Free Documentation License::).

The paragraphs above, in this section, belong to the GNU Texinfo manual and are not written by us! We have just changed the name “Texinfo” to “mockgals” because it and this manual are released under the same licenses and it is beautifully written to inform you of your rights.

1.3 How to read this manual

This manual is available in plain text, HTML, info, Docbook, PDF and finally the Texinfo format. The last one is the basis for creating all the other formats. As an initial introduction the PDF and HTML formats might be easier to begin using mockgals, but if you would like to understand the code, it is much easier to do it in the info format, where for example in emacs, you can have one window on the code and another on the info file explaining each function, the idea behind its usage and how it is used in the Emacs `*info*` window.

`info` is a very easy to use and particularly useful format for software documentation. We strongly recommend becoming familiar with it if you are not because it can significantly enhance your experience in programming and in GNU based operating systems (like GNU/Linux). It is very trivial made for reading manuals and getting facilitates navigating in long manuals. To learn more about it in any GNU/linux terminal you can run: `info info`. It does a fantastic job in explaining its self in practice interactively.

The first chapter is a quick start, explaining the generalities of Mockgals so the user can begin using it very fast. The following chapters give a much more detailed review of each step.

2 Quick start

Once `mockgals` is installed, if you simply run `./mockgals` in the installed directory, 45 mock galaxies and 5 stars will be randomly positioned in a `FITS` image of size $200 * 200$.

3 Introduction

Making mock galaxies is very important in the process of understanding our data. `mockgals` was initially made with this exact intent. Certain astronomical targets, for example elliptical galaxies, are very sharp in their central regions, this makes a simple calculation of the profile in the center of each pixel unrealistic for such cases. The main advantage of `mockgals` is that it integrates the central parts of profiles until a given accuracy. It does this without any sorting or ordering and in a very fast manner.

A summary of the advantages of `mockgals` includes:

1. Integration of the center of the profile.
2. Very efficient in CPU usage, resuling in a very fast processing.
3. Written in the C programming language, which is easy to understand and modify or contribute to by any interested user.
4. [To be added] Can make profiles in any dimentions.

4 Installation

`mockgals` relys on only 3 packages: `GSL` (for mathematical functions), `FFTW` (for convolution) and `cfitsio` (for reading to and from FITS files).

5 Configuring mockgals.

Currently configuration paramters into `mockgals` can only be given through single letter commandline options. The options are explained below.

6 Mock parameters

The parameters of the profiles you want to build into a FITS image are fed to it though a table. In this section we will review how `mockgals` reads that table and how best to prepare it.

7 How mockgals works

In this section we will give a complete review on how `mockgals` works and how this is implemented in its coding in order to facilitate your reading of the code and possibly modifying it and making it better.

Index

(Index is nonexistent)