

# Documentation for MKCLASS v1.03

Richard O. Gray

June 24, 2014

## 1 Introduction

MKCLASS is a stellar spectral classification program designed for use on a range of platforms, including most types of UNIX and Linux. MKCLASS requires as input a stellar spectrum in a two-column ascii format with wavelength in ångströms as the first column and flux or rectified intensity as the second column. Currently MKCLASS can classify spectra only in the violet – green spectral range (3800 – 5600Å) although this will be extended in later versions. MKCLASS also requires a spectral standards library; two are provided in the distribution.

## 2 Installation

MKCLASS is distributed as two tarred, gzipped files, `mkclass.tar.gz` and `mklib.tar.gz`. The source code is contained in the first file, the spectral libraries in the second. To compile MKCLASS unpack `mkclass.tar.gz` in your home or source directory with the following command:

```
tar -xvzf mkclassv103.tar.gz
```

This will create the directory `mkclassv103`. Enter that directory. To compile the code, issue the following command:

```
make all
```

This assumes that you have both the `gcc` compiler and the `make` function installed on your machine. If you do not, they can be easily installed on most modern Linux/Unix systems with a software package manager such as SYNAPTICS or `apt-get`. The program `mkclass`, its auxiliary pre-processing program `mkprelim`, and two other auxiliary programs `srebin0` and `smooth2` may be installed in your `bin` directory with the command

```
make install
```

You may need to be root to issue this last command. By default, this will place the executables in `/usr/local/bin`. If you want to alter this behavior, you will have to edit the `Makefile` in the distribution.

By default, the program expects to find the standard libraries installed in `/usr/local`; you may need to be root to do this. Copy `mklib.tar.gz` to `/usr/local`, enter `/usr/local` and issue

the command

```
tar -xvzf mklib.tar.gz
```

This will create a new directory called `mkclass` which contains in subdirectories the two provided spectral libraries.

If for some reason you cannot access `/usr/local` (for instance, you do not have root privileges on a multi-user machine), then you may install the standard libraries in another directory, and set the environment variable `MKLIB` to the path to that library (only versions 1.03 and above). So, let us suppose that you install the libraries under `/home/username/spectrastuff`, then you should set the environment variable `MKLIB` to `"/home/username/spectrastuff/mkclass"`. Where that is done depends on your operating system. It might be in the file `.bashrc`, or the file `.profile`; ask your system administrator or look it up on google. In any case, at the end of that file you should put the line:

```
Export MKLIB="/home/username/spectrastuff/mkclass"
```

You will have to log out and then log back in to make that change effective.

### 3 Operation

`MKCLASS` is a complex program that can classify stellar spectra on the MK classification system. It is strongly recommended before use it you read the journal paper on `MKCLASS` (Gray & Corbally, 2014) and that you familiarize yourself with the MK system (Gray & Corbally, 2009). `MKCLASS` operates on the command line:

```
mkclass spectrum library output log R NI
```

where `spectrum` is the full path to the stellar spectrum you wish to classify, `library` is the name of the standards library you will be using; the two libraries that come with the distribution are `libnor36` and `libr18`, `output` is the full path to the output file which will contain the spectral type and quality evaluation, `log` is the full path to a log file, which contains detailed notes on the process `MKCLASS` took to arrive at the final spectral type, `R` is either 1 or 2, depending upon which method you choose to ask `MKCLASS` to use to derive a preliminary spectral type, and `NI` is the number of iterations you wish `MKCLASS` to use. `NI = 2` or `3` is recommended.

#### 3.1 Spectrum file format

The spectrum must be an ascii file with two columns, the first column is wavelength in ångstroms, the second the flux or rectified intensity. The delimiter between the columns should be a space or spaces, not tabs. If the spectrum file contains a header, the header lines must begin with a “#” character. No blank lines are allowed in the file, including at the end.

## 3.2 Standards Library

Two standards libraries are supplied with the distribution, `libnor36` and `libr18`. The `libnor36` library consists of MK standard star spectra in normalized flux format. These spectra were obtained with the GM spectrograph employing a  $600 \text{ g mm}^{-1}$  grating on the Dark Sky Observatory (DSO) 32-inch telescope. They have a spectral range  $3800 - 5600 \text{ \AA}$ , and a spectral resolution of  $3.6 \text{ \AA}/2$  pixels. The file `mkclass.lib` contains more information about this library: the spectra are binned with a  $1.0 \text{ \AA}$  wavelength interval and the library spans a spectral-type range from O6 to M5. The `libr18` library consists of MK standard spectra obtained at DSO with a  $1200 \text{ g mm}^{-1}$  grating. They have a spectral range of  $3800 - 4620 \text{ \AA}$ , and a resolution of  $1.8 \text{ \AA}/2$  pixels. The spectra are *rectified* which means that the continuum has been normalized to unity. These spectra are binned with a  $0.5 \text{ \AA}$  wavelength interval, and the library spans a spectral-type range from O9 to K3. A third library is also present, `libr18_225` which is derived from the `libr18` library, but smoothed to reduce the resolution to  $2.25 \text{ \AA}$ .

Users may add their own spectral libraries. How to do this will be covered below.

## 3.3 Output file

The contents of a typical output file for a single spectrum is shown below:

```
hd358.r18  B7 IV-V    vgood
```

MKCLASS appends output to the output file, and so if MKCLASS is run more than once with the same output file, the spectral-type results will simply be added to the end of the file. The “vgood” is a quality indicator which depends on the S/N of the spectrum and how well that spectrum can be matched with an interpolated spectrum from the spectral library. The quality indicators are poor, fair, good, vgood, and excellent.

## 3.4 Log file

The contents of a typical log file for a single spectrum is shown below:

```
MKCLASSv103: libr18 hd358.r18
Roughtype = 1
MKLIB not set: using default MKLIB path: /usr/local/mkclass
prelim = shift
t120150p00.rbn
Initial type = t120150p00.rbn
Star appears normal
spt = 13.778106 lum = 4.492126
Initial Spectral type estimate = B8 IV-V
Classifying as a B-type star
```

```

K-line type = 14.362466
D2 = 10.5
HeII type = 15.377060
Helium/metal spectral type = 13.473224
Luminosity type = 4.498579
Polished helium/metal spectral type = 13.459750
Helium strength = -0.018598
1: B7 IV-V    0.001697
Classifying as a B-type star
K-line type = 14.361362
D2 = 10.5
HeII type = 15.071873
Helium/metal spectral type = 13.459750
Luminosity type = 4.505434
Polished helium/metal spectral type = 13.459750
Helium strength = -0.018650
2: B7 IV-V    0.001704
Classifying as a B-type star
K-line type = 14.360222
D2 = 10.5
HeII type = 15.058386
Helium/metal spectral type = 13.459750
Luminosity type = 4.505434
Polished helium/metal spectral type = 13.459750
Helium strength = -0.018650
3: B7 IV-V    0.001704
=====

```

This file shows the results for 3 iterations with MKCLASS. MKCLASS uses a numerical spectral code for temperature types and luminosity classes which are then translated into a spectral type at the end. Those numerical codes can be understood by examining the contents of `code3spt.c`.

### 3.5 Preliminary spectral type algorithm

MKCLASS employs two algorithms to determine a preliminary or “starting” spectral type for the star. Roughtype 1 (invoked by specifying  $R = 1$ ) should be used only for rectified spectra. Roughtype 2 ( $R = 2$ ) can be used with both rectified and flux-calibrated spectra.

### 3.6 Iterations

MKCLASS can be programmed to iterate a number of times through the classification process. This can help to obtain more accurate spectral types, especially if the flux calibration of the program spectra is poor.

The above results for the spectrum `hd358.r18` were obtained with the command line:

```
mkclass hd358.r18 hd358.out hd358.log 1 3
```

for most stars, MKCLASS will also produce a “matched” spectrum which is its best fit to the program spectrum from the standards library. For the above example, this “matched” spectrum will have the name `hd358.mat`.

## 4 Preprocessing Spectra

If you are not using a standards library derived from the same spectrograph/grating combination as the program stars, then some preprocessing will be required. Let us suppose that your program spectra are approximately flux calibrated, have a resolution of  $3\text{\AA}/2$  pixels and a spectral range from  $3500 - 6000\text{\AA}$ . This means that you can preprocess your spectra so that they can be classified against the `libnor36` standards library. See above. The preprocessing steps will be as follows: 1) If necessary, extract your spectra from fits files and transform them into the required ASCII file format (see above). 2) Truncate your spectra to the spectral range  $3800 - 5600\text{\AA}$ . 3) Smooth your spectra by convolving with a gaussian of the appropriate width to change the effective resolution to  $3.6\text{\AA}$ . This may take some experimentation.

Step 1 is left to the user, as the way spectra are stored in fits files is not uniform from observatory to observatory. When the spectrum is extracted to an ASCII file, the format must again be with two columns, the first for the wavelength in angstroms, the second with the flux. Header lines must start with the character “#”. There should be no blank lines in the file. For step 2, the program `srebin0`, provided in the distribution, may be used to truncate the spectrum to the  $3800 - 5600\text{\AA}$  spectral region and to rebin the spectrum to a uniform spacing. This program is used as follows:

```
srebin0 input output 3800.0 5600.0 1.0
```

where `input` is the input spectrum prepared in step 1, and `output` is the truncated spectrum. The spectrum will be truncated to  $3800 - 5600\text{\AA}$ , and will be rebinned to a uniform spacing of  $1.0\text{\AA}$ . For step 3, the program `smooth2` may be used to convolve a gaussian function with your spectrum to give an output spectrum with a resolution of  $3.6\text{\AA}$ . Some experimentation will be required, but if the beginning resolution is estimated to be  $3.0\text{\AA}$ , then smoothing with a gaussian function with  $\text{FWHM} = 2.0\text{\AA}$  is a good first guess. This is derived from the consideration that:

$$2.0 = \sqrt{(3.6^2 - 3.0^2)}$$

The program `smooth2` may be used as follows:

```
smooth2 input output input_spacing resolution output_spacing
```

The program `smooth2` requires that the spectrum has a uniform wavelength spacing, and that has already been provided by `srebin0`. The output spacing must be an integer multiple of the input spacing. For this application, `input_spacing = output_spacing = 1.0Å`. The “resolution” is the FWHM for the gaussian function, in this case `2.0Å`. Deciding on the appropriate FWHM to use with `smooth2` may require some trial and error. Choose a relatively high S/N spectrum from your program list. That spectrum should be of star for which rotational broadening is either small or is not apparent. Go through the above process to transform your spectrum. Normalize the flux to unity at the wavelength `4503Å`, and then find a spectrum in the `libnor36` library of a similar spectral type, plot the two on top of each other, and compare the widths of the spectral lines. Adjust the resolution as necessary until you get a match in resolution between the program spectrum and the library spectrum.

These two programs may be combined in a Unix script to automate the above procedure. For instance, write the script (included in the distribution as `preprocess_example.sh`):

```
srebin0 "$1" temp.rbn 3800.0 5600.0 0.2
smooth2 temp.rbn "$2" 0.2 3.46 1.0
```

This script is designed to smooth the Indo-US library spectra (Valdes et al, 2004) to the resolution of the `libnor36` library. You will have to change it into an executable:

```
chmod a+x preprocess_example.sh
```

It is executed as follows:

```
./preprocess_example 47731.txt 47731.f36
```

where `47731.txt` is the spectrum of HD 47731 from the Indo-US library (edited by hand to remove the blank lines). The script first applies `srebin0` to rebin to a wavelength step of 0.2 (`0.4Å` is the native binning of the Indo-US spectra, but the final output spacing desired is 1.0: recall that the output spacing from `smooth2` must be an integer multiple of the input spacing. Thus we rebin to 0.2 in order not to lose information before the smoothing takes place.) and truncates to the `3800 – 5600Å` region. The script then applies `smooth2` to produce the final spectrum, smoothed to `3.6Å`, with a spacing of `1.0Å`.

If you want to write a script to do the whole thing – rebin, smooth, and classify at one fell swoop, then the following should do you (call the script `fell_swoop.sh`)

```
srebin0 "$1" temp.rbn 3800.0 5600.0 0.2
smooth2 temp.rbn "$2" 0.2 3.46 1.0
mkclass "$2" libnor36 results.out results.log 2 3
```

it is executed as

```
fell_swoop.sh 47731.txt 47731.f36
```

The results will be printed to `results.out`. If you want to preprocess and classify a large number of stars at one time, make a batch file:

```
fell_swoop.sh star1.flx star1.f36
```

```
fell_swoop.sh star2.flx star2.f36
```

```
.  
. .  
.
```

make the batch file executable, run it, sit back and think pleasant thoughts until it completes.

## 5 Making your own Spectral Library

Constructing a full spectral standards library is a formidable task, and will require many hours at the telescope. However, if you are classifying stars only within a certain restricted spectral type range (say F and G-type stars), then the task is not so bad. To construct a spectral standards library, you should observe with your telescope/spectrograph setup as many standards as possible within the spectral type range of interest. It will probably be wise to get standard spectra slightly earlier and slightly later than your desired range. So, for instance, if you are interested in classifying stars in the F0 – G5 range, get spectra of standards in the range A5 – K0. You should try to get examples at the luminosity types V, III, Ib and Ia, and intermediate luminosity types such as IV and II will also be useful. You will then use those standard star observations to fill in *completely* a subset of the following 2-dimensional grid:

06 V	06 III	06 Ib	06 Ia	3.0
07 V	07 III	07 Ib	07 Ia	4.0
08 V	08 III	08 Ib	08 Ia	5.0
09 V	09 III	09 Ib	09 Ia	6.0
B0 V	B0 III	B0 Ib	B0 Ia	7.0
B1 V	B1 III	B1 Ib	B1 Ia	8.0
B2 V	B2 III	B2 Ib	B2 Ia	9.0
B3 V	B3 III	B3 Ib	B3 Ia	10.0
B5 V	B5 III	B5 Ib	B5 Ia	12.0
B7 V	B7 III	B7 Ib	B7 Ia	13.0
B8 V	B8 III	B8 Ib	B8 Ia	14.0
B9 V	B9 III	B9 Ib	B9 Ia	15.0
A0 V	A0 III	A0 Ib	A0 Ia	16.0
A1 V	A1 III	A1 Ib	A1 Ia	17.0
A3 V	A3 III	A3 Ib	A3 Ia	19.0
A5 V	A5 III	A5 Ib	A5 Ia	20.0
A7 V	A7 III	A7 Ib	A7 Ia	21.0
F0 V	F0 III	F0 Ib	F0 Ia	23.0
F2 V	F2 III	F2 Ib	F2 Ia	24.0
F3 V	F3 III	F3 Ib	F3 Ia	25.0

F5 V	F5 III	F5 Ib	F5 Ia	26.0
F6 V	F6 III	F6 Ib	F6 Ia	27.0
F8 V	F8 III	F8 Ib	F8 Ia	28.0
F9 V	F9 III	F9 Ib	F9 Ia	29.0
G0 V	G0 III	G0 Ib	G0 Ia	30.0
G2 V	G2 III	G2 Ib	G2 Ia	31.0
G5 V	G5 III	G5 Ib	G5 Ia	32.0
G8 V	G8 III	G8 Ib	G8 Ia	33.0
K0 V	K0 III	K0 Ib	K0 Ia	34.0
K2 V	K2 III	K2 Ib	K2 Ia	36.0
K3 V	K3 III	K3 Ib	K3 Ia	37.0
K5 V	K5 III	K5 Ib	K5 Ia	39.0
K7 V	K7 III	K7 Ib	K7 Ia	40.0
M0 V	M0 III	M0 Ib	M0 Ia	40.7
M2 V	M2 III	M2 Ib	M2 Ia	42.5
M3.5 V	M3.5 III	M3.5 Ib	M3.5 Ia	44.0
M5 V	M5 III	M5 Ib	M5 Ia	45.5

The final column of the table above is the temperature type running number. Running numbers may also be assigned for luminosity classes; these are the ones used by MKCLASS: V = 5.0, III = 3.0, Ib = 1.0, Ia = 0.0. MKCLASS operates in this 2D-space of running numbers and only translates those codes into a traditional spectral type at the end.

Lists of spectral standards may be found in the appendices of Gray & Corbally (2009), but a glance at those lists will show that many of the above grid points are not represented by standards! Some gaps may be filled by consulting papers of the founders of the MK system (Morgan and Keenan) and by those who have been involved in extending and refining the system (Walborn, Garrison, Hiltner, Schild, Lesh [Rountree], Abt, Boeshaar, Kirkpatrick, Corbally, Gray, etc.). Don't trust the spectral types in libraries of spectra, such as the Jacoby atlas or the Indo-US because either those spectral types were assigned in non-orthodox ways, or the spectral types are from non-homogeneous sources. The same goes for the spectral types on Simbad. Even after using these sources, gaps in the above grid will remain, and these will need to be filled by interpolation or, in extremis, by extrapolation. For instance, if a F3 III standard is not available, interpolate between F2 III and F5 III using the running numbers in the interpolation process. Library spectra *must* have names that obey the following format: `tNNNlnnp00.rbn`, where the fifth character is the letter “l”, standing for “luminosity”, the NNN is the temperature-type running number (without the decimal point) and the `nn` is the luminosity-type running number, also without the decimal point. So, the library spectrum for an F5 III star will have the name `t260130p00.rbn`.

The library spectra *must* have exactly the same spectral range and the same wavelength spacing. For the libnor36 spectra, for instance, the first wavelength is 3800.000Å, and the last wavelength is 5599.000. The wavelength spacing is 1.0. Note that if an observed spectrum used for a library



spectrum does not cover exactly that range, the first and last few wavelength points can have zero fluxes. So, for the library spectrum `t390110p00.rbn` in `libnor36` the first 22 wavelengths have 0.000 for the flux, and the last 2 wavelengths also have zero fluxes.

## 6 Final Thoughts

Keep in mind that MKCLASS is a computer program, that it can make mistakes, that it is a work in progress, and that the human eye-brain combination is still better than any classification program. If your paper or research depends on a spectral type given by MKCLASS, look at the spectrum yourself and verify that spectral type.

Keep in mind that MKCLASS has certain important limitations – it doesn't work with red or infrared or UV spectra – extending to those spectral regions will be projects for the future. It will not classify L, T, and Y dwarfs. Its classification of O-type stars is still rudimentary. It is not capable of classifying white dwarfs, carbon stars, or other MK types other than informing you that it has encountered such a spectrum. Again, these are projects for the future.

Because MKCLASS *is* a work in progress, we still do not have a lot of experience using it. If you are interested in using MKCLASS for a large project (for instance, classifying hundreds of thousands of stellar spectra from the SDSS), it might be a good idea to contact us first for a consultation and help. Applying MKCLASS to different datasets will help to make it a robust program and so more useful to the astronomical community.

Contact information:

Richard Gray, [grayro@appstate.edu](mailto:grayro@appstate.edu)

Chris Corbally, [corbally@as.arizona.edu](mailto:corbally@as.arizona.edu)

## References

Gray, R.O., & Corbally, C.J. 2014, AJ 147, 80

Gray, R.O., & Corbally, C.J. 2009, “Stellar Spectral Classification” (Princeton: Princeton University Press)

Valdes, F., Gupta, R., Rose, J.A., Singh, H.P., & Bell, D.J. 2004, ApJS 152, 251