

# Documentation for `compile_fits.py`

## Overview

This script processes a directory of FITS files using Gaussian fitting and chi-square analysis. It filters files based on specific criteria, handles background files, and attempts to merge contiguous files when necessary.

## Dependencies

The script relies on the following Python libraries:

- **os**: For file and directory operations.
- **shutil**: For file copying.
- **astropy.io.fits**: For reading and manipulating FITS files.
- **scipy.optimize.curve\_fit**: For performing Gaussian curve fitting.
- **scipy.stats.chisquare**: For chi-square analysis.
- **numpy**: For numerical operations.
- **subprocess**: For executing external commands.

## Functions

`three_gaussians(x, a1, b1, c1, a2, b2, c2, a3, b3, c3)`

This function models the sum of three Gaussian distributions.

- **Parameters:**
  - **x**: Independent variable.
  - **a1, b1, c1**: Amplitude, mean, and standard deviation of the first Gaussian.
  - **a2, b2, c2**: Parameters for the second Gaussian.
  - **a3, b3, c3**: Parameters for the third Gaussian.
- **Returns:** The sum of three Gaussian distributions evaluated at **x**.

### `gauss_fit_chi2(channel, count)`

Fits the data within the 1-2 keV energy range using the `three_gaussians` model and computes the chi-square value.

- **Parameters:**
  - `channel`: Array of energy values.
  - `count`: Array of count values.
- **Returns:** Chi-square value if fitting is successful; `None` otherwise.

### `process_fits_file(fits_file, compiled_folder)`

Processes a single FITS file, evaluates it using Gaussian fitting, and categorizes it based on chi-square results.

- **Parameters:**
  - `fits_file`: Path to the FITS file.
  - `compiled_folder`: Directory to store processed files.
- **Returns:** Tuple (`success`, `chi2`), where `success` indicates if the file met criteria, and `chi2` is the computed chi-square value.

### `isBG(fits_file)`

Checks if the FITS file corresponds to a background (BG) file based on the `SOLARANG` header value.

- **Parameters:** `fits_file`: Path to the FITS file.
- **Returns:** `True` if the file is a background file, `False` otherwise.

### `add_fits_files(file_list, output_dir, fits_directory)`

Merges contiguous FITS files using the external `gd1` command and generates a combined output file.

- **Parameters:**
  - `file_list`: List of FITS files to merge.
  - `output_dir`: Directory to store the combined file.
  - `fits_directory`: Directory containing the original FITS files.
- **Returns:** Path to the combined output file.

```
main(fits_directory, compiled_folder)
```

Main logic to process all FITS files in the specified directory:

- Filters out background files.
- Processes individual files.
- Merges and processes contiguous files when chi-square values are outside the desired range.

## Execution

### Command-line Arguments

- `--fits_directory` or `-d`: Path to the directory containing FITS files.
- `--compiled_folder` or `-c`: Path to the directory where processed files will be stored.

### How to Run

Example command:

```
python compile_fits.py -d /path/to/fits/files -c /path/to/compiled/files
```

## Output

- **Processed FITS files:** Files with acceptable chi-square values are moved to categorized subfolders based on their timestamp.
- **Merged Files:** Contiguous files are combined and stored in the compiled folder when required.

## Error Handling

- Catches exceptions during Gaussian fitting.
- Logs errors and skips files that fail processing.
- Removes intermediate files if merging fails.

## Notes

- Adjust the `initial_guess` in `gauss_fit_chi2` for different datasets if necessary.
- Ensure `gdl` is installed and configured for the `add_fits_files` function to work.
- Review chi-square thresholds ( $0.8 \leq \chi^2 \leq 2$ ) and modify them as needed for specific use cases.