Outlier Detection Code Documentation

1 main.py

1.1 Initial execution

This is the file that is executed with `python3` to run the program. On initial execution, it will prompt the user for their postgres info, specifically the host name/address, database name, user name, and password. The user is then prompted for their email address. If any outliers are flagged, the program will send an email to this address. The email will come from ourfish.outliers@gmail.com. The program will ask the user again for their postgres info if it is not able to make a simple query to the database for info from the `fishdata_catch` table. After that, the program will check for outliers from yesterday's data.

1.2 Checking outliers

The program will perform some data extraction; pulling in data from the postgres server, cleaning the data, then performing the outlier test described in the methodology report. If no samples are flagged, the program will send an email to ourfish.outliers@gmail.com as a way of monitoring its status. Otherwise, the program will send an email to the address provided notifying the user of flagged data. The email will include plots as attachments and a .csv containing the flagged data. The plots and .csv are also saved locally, in `\plots\yesterdays_date` and `\flagged_data`.

1.3 Continuity

After the initial check, the program will continue running but won't do anything until midnight. The program will always perform its outlier check at midnight.

1.4 Crash reporting

If the program crashes at any point, a log file is created and emailed to Angel. The logs are saved in the `logs` folder. The program may crash if the names of things change. For example, if the database name is changed, the program will crash. Or if names of folders change. These are quick fixes. If the program crashes and a warning pops up that *doesn't* mention that Angel will receive an email, then it's a really bad crash and no email will be sent. If that happens, please notify angel.d.umana@gmail.com and send any crash report info available (perhaps in the command prompt window).

2 algorithm.py

This script contains the routine for checking outliers. It will iterate through yesterday's data and compare it to all the data on record. If it flags any samples, plots will be saved to `plots\yesterdays_date`. The script includes helper functions for finding outliers and a plotting function.

3 clean_fish.py

Currently unused, but was used to clean data from `\data\fishdata_buyingunit.csv`, a dump from the postgres server. The clean data is in `\data\fishdata_buyingunit_clean.csv`. This script may be employed later to continuously update the fish data.

4 data_clean.py

Filters out samples from the `clean_postgres_data()` function from `postgres.py` that have inconsistent data. Inconsistent data means data where `unit_price`*quantity does not equal the designated `total_price`. Quantity is any one of `weight_kg`, `weight_lbs`, or `count`, since different samples use different units. The resulting data for all existing records are saved to `\data\clean_catch_data.csv`.

5 emailing.py

Handles emailing results of flagged data. Also includes code for pinging ourfish.outliers@gmail.com and asking the user for an email address at startup.

6 exception_handling.py

If the program crashes, this script handles sending out the crash report and displaying a warning that the program crashed.

7 postgres.py

This script handles querying the postgres server for data as well as cleaning it. Also handles asking the user for their postgres info. Each day's new samples queried are saved into `\data\postgres_dump.csv`. Clean data of all existing records are saved in `\data\pg_data_clean.csv`.

8 Miscellaneous

There is some existing data in `\flagged_data` and `\plots\2019-01-1`. These are flagged data starting from Jan. 1 2019 and ending on Mar. 2 2021.