Bovy's Net Analysis

February 17, 2020

We would like to create a GitHub project for the computation of Magallanic Cloud distances and comparison to Alvaro. The idea is to use the local GitHubs that allow the running of astronn and bovys paper routines but only considering small and specific codes, such as dr16 dataset download and prediction of distances

So, let us start with the creation of local GitHub tools in astronn-local. Later we can create a GitHub project with exercises that use these tools.

January 21, 2020

Computation of distances to LMC and SMC considering the trained models of Bovy

- First, download the dr16 for LMC and SMC datasets locally
- Secondly, Use different pre-trained models reproduce Alvaro's result
- check notebook astroNN_gaia_dr2_paper/distance-computation.ipynb

January 14, 2020

Training

1. read gaia_dr2_train.h5 and generate the model astroNN_no_offset_model/
2. read gaia_dr2_train.h5 and generate the model astroNN_constant_model/
3. read gaia_dr2_train.h5 and generate the model astroNN_multivariate_model/
    1. 20% in mean_absolute_percentage_error with 100 epochs

January 10, 2020

Datasets_Data_Reduction

1. download and generation of apogee normalized spectra, *contspec_dr14_140K.fits*. This file is generated from the allstar_data file from apogee. We have recorded 140K spectra in constpec file. However the apogee_dr14 allstar(dr=14) file contains 277371 elements
    - Notice that the first element of the column allstar_data['LOCATION_ID'], i.e for counter==0, is equal to 1, so the spec == 0 for all the components. This is saved in contspec file.
    - The first element (counter == 0) of the column apogee_id from catalog files is VESTA, so null spectrum correspond to unknown star, which is ok

    - On the other hand, the counter==1 non-null spectrum recorded in the spec array has apogee_id=2M00000002+7417074, which coincides with the counter==1 element of the catalogs, which is ok
    - Indeed, if the LOCATION_ID==1 or ap_path==False the saved spectrum is null, but the array element is anyway created.
    - Later, only the elements with non-null spectrum are used. It is IMPORTANT to notice that spec contains 1 element for each of the allstar elements, even if they contain LOCATION_ID==1 or false ap_path
2. generation of *apogeedr14_gaiadr2_xmatch_allcolumns.fits* file bases in allstar file and gaia_tools. The generation of this file is independent of the contspec file. It contains 275020 lines, quite close to the full apogee_dr14 files
3. generation of reduced file apogeedr14_gaiadr2_xmatch.fits, which does not contain allcolumns. This file contains 277371 elements, as the full apogee_dr14 allstar file. This is in order to keep the correspondence between constpec fits file and the elementes of gaia_dr2 xmatch elements. So, I guess that there are approximately 2000 apogee stars without gaia_dr2 associated data
4. for simplicity we copy *contspec_dr14_140K.fits to contspec_dr14.fits.* We read it and check that it contains 140K elements in .data. Indeed the shape of file[0] is (140000, 7514) but we still need to check the meaning of the number 7514
5. we get the first 140K elements of allstar_apogee and gaia_dr2_xmatch generated files. Also we consider the first 140K elements of an external file called *astroNN_apogee_dr14_catalog.fits file*, which seems to contain extra data, such as the effective temperature (check if this is included in apogee original data)
6. we have checked that the apogee_id of the catalog files indeed coincide. Also, we have checked that the spectrum list of files agree too, which is super good!!
7. generation of gaia_dr2_train.h5 (18K) and gaia_dr2_test.h5 (15K) files, which contain only the well behaved stars, i.e. with non-null apogee spectrum, parallax, SNR>200, etc. plots generation

Alvaro’s guidelines to get magallanic cloud star apogee information

Si tienes abierta la tabla allStar en python, por ejemplo haciendo:

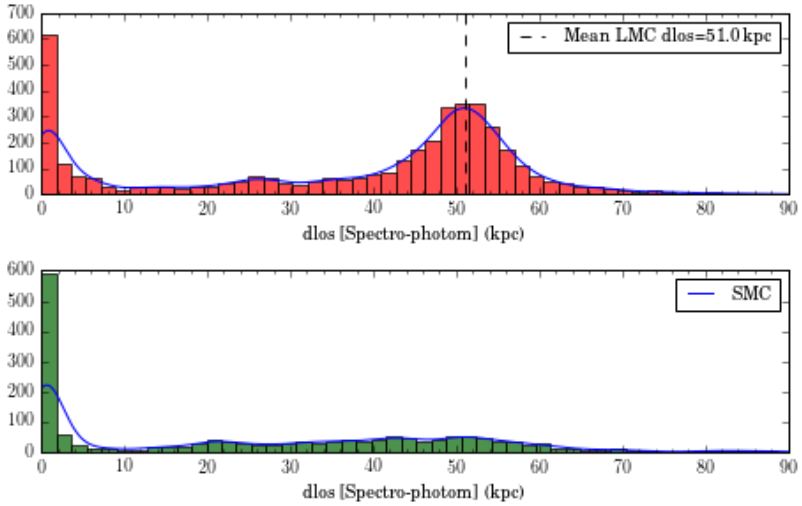
```
import astropy.io.fits as pf
import numpy as np
hh0 = pf.open("../tables/allStar-r13-l33-58672.fits")
dd0 = hh0[1].data
```

puedes seleccionar estrellas de las nubes usando estos boolean arrays:

```
i_lmc = np.array(["LMC" in x for x in dd0["field"]])
i_smc = np.array(["SMC" in x for x in dd0["field"]])
```

los hice así porque en la columna "field" aparece la palabra "LMC" o "SMC" pero generalmente con algo más.

Cuando hice el experimento con mis distancias spectro-photometric concluimos que APOGEE no llega a agarrar muchas estrellas de la SMC porque está muy lejos. Las distribuciones de distancia son estas:



Los valores de referencia son estos (tal vez para la SMC haya alguna publicación más reciente, habría que verificar):
LMC --> D=49.59+/-0.54 kpc (Pietrzyński+2019)
SMC --> D=62.1+/-1.9 kpc (Graczyk+2013).

December 15, 2019

Inference.ipynb
Offset_Gaia.ipynb

apogee_non_rc_ra.txt not found
Where is this file?

Jacobian.ipynb (done)
MW_Science.ipynb (done)

December 14, 2019

Notice that the notebooks in astroNN_gaia_dr2_paper assume that we are going to work with the full apogee dataset (227000 samples). This is costly and tedious so we can reduce the number of samples to work to be 40K but weed to change a couple of lines in

Datasets_Data_Reduction.ipynb (number of samples, done)
Training.ipynb (done)
Inference.ipynb (number of samples, done)
Offset_Gaia.ipynb (we modified the import keras part, done)

1) astronn package bugs

install astronn from GitHub

python setup.py install

try python, from astroNN.apogee import allstar, visit_spectra and you will find an error. comment line 11 of astroNN/astroNN/apogee/__init__.py and install again

python setup.py install

2) Requirements

conda create -n python37-astronn python=3.7

conda activate python37-astronn

pip install –upgrade tensorflow
pip install --upgrade tensorflow-probability
pip install astropy
pip install pydot
pip install galpy
conda install graphviz
conda install -c pytorch pytorch

install from github the **astronn** packages

- https://github.com/henrysky/astroNN or
  https://astronn.readthedocs.io/en/latest/quick_start.html (latest version)
  git clone --depth=1 https://github.com/henrysky/astroNN
- https://github.com/henrysky/astroNN_gaia_dr2_paper
  git clone https://github.com/henrysky/astroNN_gaia_dr2_paper.git
- https://github.com/jobovy/gaia_tools
- git clone https://github.com/jobovy/gaia_tools.git

cd gaia_tools
python setup.py install

cd astroNN
python setup.py install
if fails go to 1) and try again

setup environment variables for astronn datasets in .bashrc

```
export SDSS_LOCAL_SAS_MIRROR=/home/bapanes/Research-Now/astronn-Local/SDSS_LOCAL_SAS_MIRROR
export GAIA_TOOLS_DATA=/home/bapanes/Research-Now/astronn-Local/GAIA_TOOLS_DATA
export LASMOT_DR5_DATA=/home/bapanes/Research-Now/astronn-Local/LASMOT_DR5_DATA
```

conda install jupyter

git clone https://github.com/bapanes/Automatic-Spectral-Distances.git

jupyter-notebook
navigate to distance-computation.ipynb

just comments:

pip install –upgrade tensorflow indicates that we are not going to require gpu resources, which is detected by astronn programs automatically, behind the scenes. Instead, if we do pip install –upgrade tensorflow-gpu we could see the gpu version of astronn, but I have not tried that since I have a video card which is not compatible with tensorflow-gpu as far as I know. This is supported by the fact that Nvidia cards work on the fly with tensorflow-gpu, I tested, and because is included by design. Instead, AMD cards require some extra work as can be seen in MLC-Notes-GPU_AMD_Radeon_Tensorflow.doc.

conda list shows that astronn, gaia_tools, etc get installed in the tree of the environment, which is quite elegegant since we could try to export the full environment to get reproduced by other users. I have not tried this option in conda

link useful because it talks about versions and system requierements

https://astronn.readthedocs.io/en/latest/quick_start.html

December 12, 2019

check the paths included in .bashrc regarding astronn and gaia_tools folders

playing with the notebook Training.ipynb

1) after some iterations with conda environments we have found that this program is able to work in our old environment Python37_Class, which has tensorflow 1.13.1 With other new environments considering tensorflow 2.0 we did not have success. So, I am going to try an environment with tensorflow 1.13 but from scratch

pip install jupyter
pip install tensorflow==1.13.1
pip install tensorflow-probability==0.6.0
pip install astronn
pip install pydot
conda install graphviz
pip install torch (**yep, we can use torch without gpus**)

with these instructions the notebook Training.ipynb is working from end to end
also it is working Datasets_Data_Reductio.ipynb

2) maybe the problem with the newest version of tensorflow (2.0.0) appears because of the installation of astronn, let us try to install astronn from the source

pip install –upgrade tensorflow
pip install –upgrade tensorflow-probability
pip install jupyter
pip install pydot
conda install graphviz
pip install torch (**yep, we can use torch without gpus**)

astronn/python setup.py install
gaia_tools/python setup.py install

This also works. This is probably because the latest version of tensorflow is only compatible with the latest version of astronn. However we still have some problems during the installation of astronn, although the scripts seem to work

running inference.ipynb

December 9, 2019

- visit_spectra details
- Gaia ESO Survey: optical spectrum information, analogous to APOGEE
- Training and test file sizes depending on the distances

December 6, 2019

277000 files of 1MB = 277 GB. I have that space in my disk so I am dowoading all apogee continnum spectra

Meeting with Anell

- astronn installation does not work properly in windows anell's lapton. We have decided to move her laptop to ubuntu 18.04

December 5, 2019

Exploring the data with **bapanes_data_visualization.ipynb**

- Target: identify the key aspects of apogee and gaia data that allow the use of bovy approach
  - Study these aspects in the context of GES and 4MOST
- Variable **allstar_data** contains list of all apogee stars with huge amount of detail: 277371 elements, including the spectrum? Nop, this is just saved in the loop that run over the allstar_data variable in **Datasets_Data_Reduction.ipynb**
  - There are repeated stars in the table associated to allstar. We should try to remove these cases
- what is the difference between **astroNN_data_file** and allstar_file. They both seem to deal with apogee data
  - It is Teff field. What is this?
- The columns that we need to include in the training and test sets is explicitly shown during the creation of h5 files
  - continuum spectrum comes from apogee allstar
  - parallax, etc from gaia data
  - teff from astroNN_data_file
- We have to check that the continuum spectrum saved in **contspec_dr14.fits** indeed correspond, in terms of array indices, to the same objects in **gaia_data_file** and **astroNN_data_file,** specially since the first element of apogee data in allstar_data is VESTA con RA=0 and DEC=0.
  - **astroNN_data_file** has the same structure as **allstar_data**
  - **apogeedr14_gaiadr2_xmatch_allcolumns.fits** does not contain VESTA but **apogeedr14_gaiadr2_xmatch.fits** does and it is mostly builded from **allstar_data**. We use the latter to generate the train and test sets, so maybe for that reason we expect match between apogee and matched data
  - In bovy file we reads:
  - # But then we only interested in a handful of columns
  - # and we want the .fits to nicely match allstar from row to row

    - # USE 'ra' and 'dec' in all_columns !!!!!!! NOT 'RA' and 'DEC'
    - There, we can see that the rows that exist in gaia are used to fill the rows of apogee stars and when this is not possible, as for VESTA, we use -9999 to fill the common columns

December 3, 2019

Tools

- https://github.com/henrysky/astroNN
- https://github.com/henrysky
- https://github.com/henrysky/astroNN_gaia_dr2_paper
- https://github.com/jobovy/gaia_tools

Working with Datasets_Data_Reduction.ipynb from repo astroNN_gaia_dr2_paper

- After some minimal modifications implemented in order to reduce the size of data and the installation of necessary packages, such as gaia_tools or galpy, we have checked that all the steps are working
- In the first step, we notice that the full APOGEE data-set size, 200K 100MB files approximately, is out of reach, so we download only 1000 samples
- In the second step, we have to check if this affects the consistent selection of matched Gaia stars
    - Maybe all the APOGEE stars can be found in Gaia sample, in which case the matched Gaia sample is always of the same size as APOGEE shortened list

December 2, 2019

Installation of astroNN:

- pip plus GitHub
- Clone of astroNN_gaia_dr2_paper
- Starting with Dataset_Data_Reduction

November 11, 2019

Discussion about paper
https://ui.adsabs.harvard.edu/abs/2019MNRAS.489.2079L/abstract

“Simultaneous calibration of spectro-photometric distances and the Gaia DR2 parallax zero-point offset with deep learning” Bovy, Jo et al.

Básicamente, dado una interseccion entre estrellas de Gaia y Apogee para distancias menores a 2kpc es posible encontrar un modelo que permite predecir distancias solo en base al espectro medido por Apogee, para estrellas mucho más alejadas que 2kpc, incluso con mejor precisión que Gaia. Esto permite, por ejemplo, estudiar la estructura químico dinámica de las estrellas considerando su ubicación espacial, en este caso se enfocan en el disco pero supongo que uno puede escoger otras zonas, cómo el bulge, etc.

El espectro medido por APOGEE tiene un formato analogo a las time-series, que podriamos llamar 1D plots, donde el eje y contiene el flujo y el eje x la frecuencia. En este caso es posible aplicar layers convolucionales, siempre y cuando estas también sean en 1D, como se explica en el siguiente link

https://blog.goodaudience.com/introduction-to-1d-convolutional-neural-networks-in-keras-for-time-sequences-3a7ff801a2cf