Notebook

March 26, 2021

[1]:

```
dataset_name = 'mohr_smith'
```

[2]:

```
%reload_ext autoreload
%autoreload 2
default_figsize=(14,12)
```

[3]:

```
import datasets
import numpy as np
import pandas as pd
import seaborn as sn
import matplotlib.pyplot as plt
import matplotlib
matplotlib.rcParams['figure.figsize'] = (14, 12)


dataset_module = datasets.datasets_by_name_all[dataset_name]
x,y,metadata = dataset_module.load(dropna=True,verbose=True)
y = datasets.map_y_em(y,dataset_name)

# generate dataframe with both x and y
xy = pd.concat([x,y],axis=1)
xy.describe()
```

```
Warning loading data from Mohr-Smith_2017.csv:
Dropped 38 rows with missing values.
Rows (original):   5915
Rows (after drop): 5877
```

[3]:

```
              umag         gmag         rmag         imag        Hamag  \
count  5877.000000  5877.000000  5877.000000  5877.000000  5877.000000
mean     17.216993    16.980114    15.609820    14.803716    15.268086
std       2.328989     1.966562     1.659154     1.527776     1.620235
min      12.143000    13.004000    11.957000    11.081000    11.620000
25%      15.378000    15.410000    14.243000    13.561000    13.940000
50%      17.371000    17.121000    15.716000    14.847000    15.362000
75%      19.272000    18.684000    16.989000    16.049000    16.606000
max      21.260000    19.998000    18.669000    17.864000    18.737000
```

```
              Jmag         Hmag         Kmag           em
count  5877.00000  5877.000000  5877.000000  5877.000000
mean     13.37808    12.937877    12.713908     0.055470
std       1.39622     1.373603     1.387794     0.228916
min       8.69300     7.870000     7.414000     0.000000
25%      12.35800    11.977000    11.765000     0.000000
50%      13.34900    12.930000    12.710000     0.000000
75%      14.44600    13.976000    13.742000     0.000000
max      16.38600    15.896000    15.691000     1.000000
```

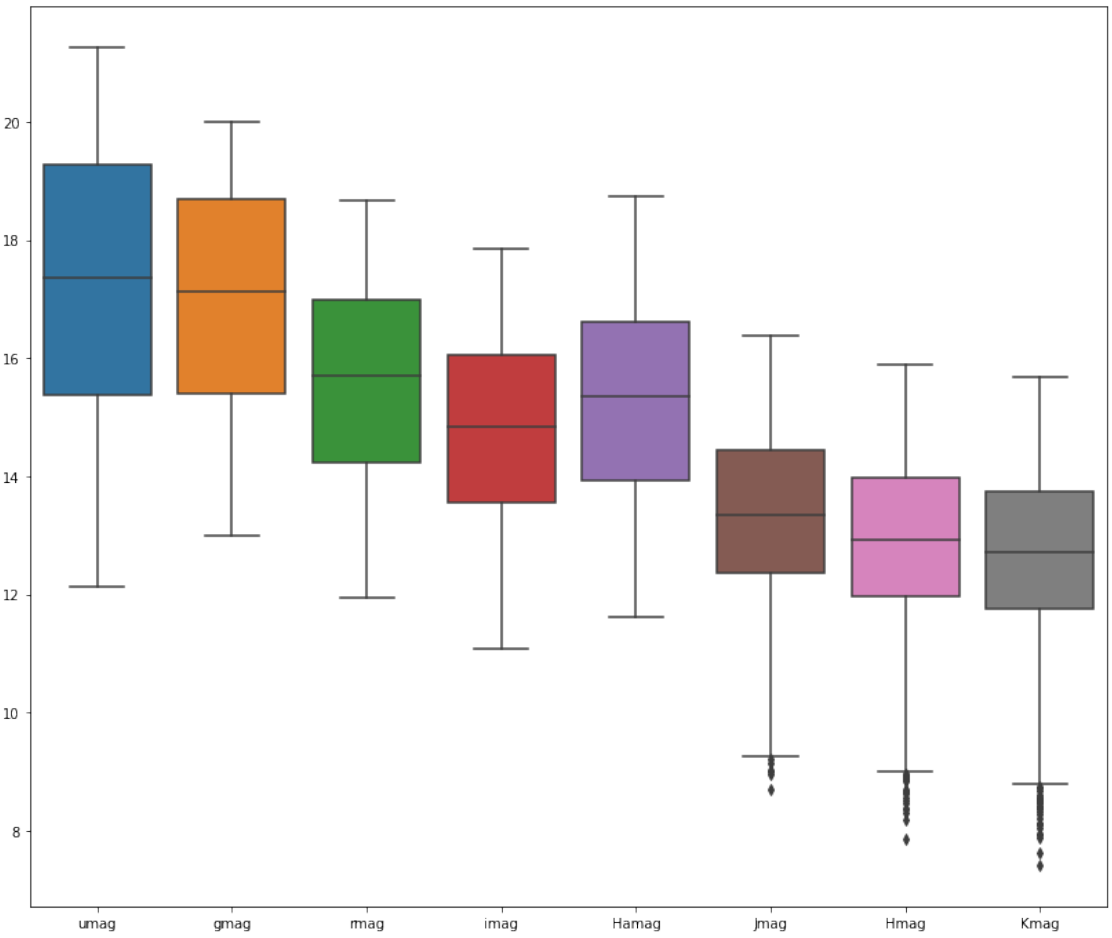
1 Variable visualization

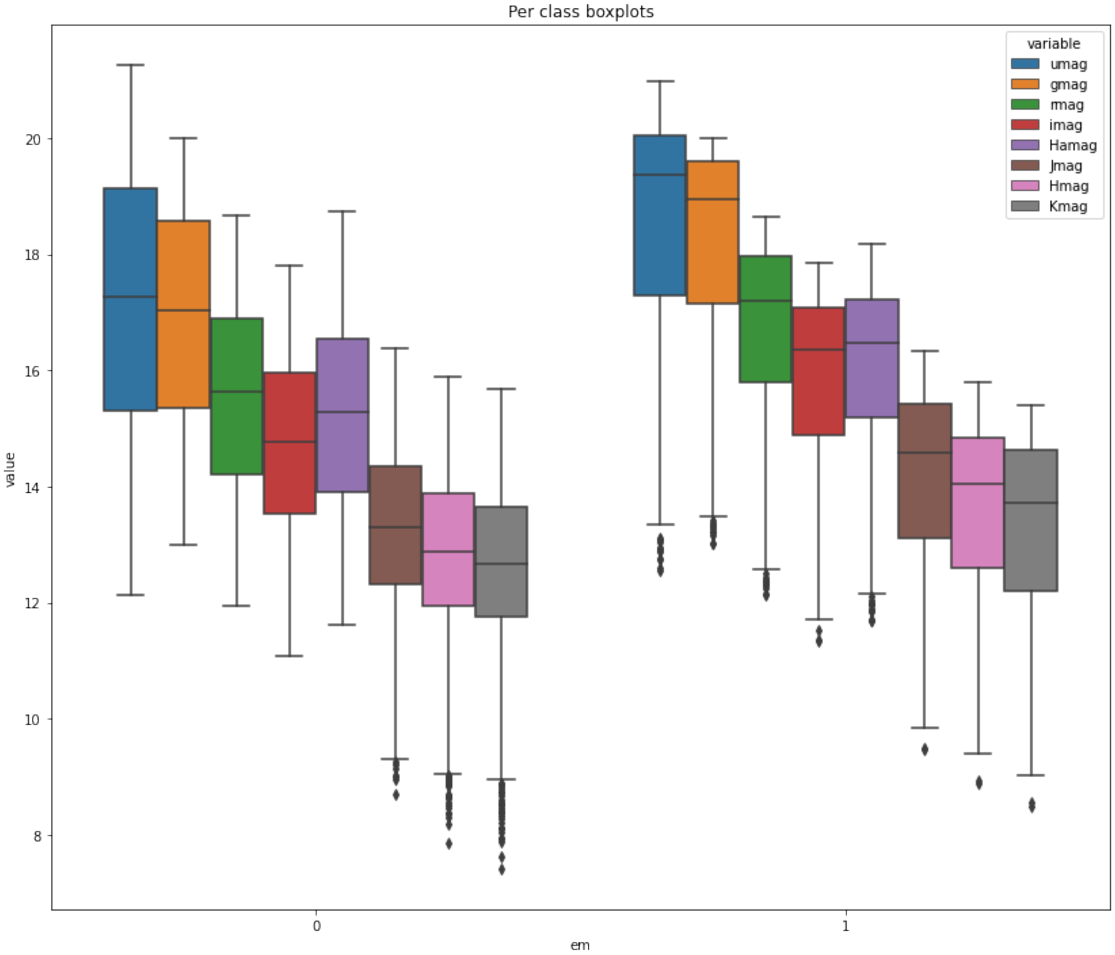
[4]:

```
sn.boxplot(data=x)

plt.figure()
xy_long = pd.melt(xy, id_vars='em')
sn.boxplot(x='em', y='value', hue='variable', data=xy_long)
plt.title("Per class boxplots")
```

[4]:

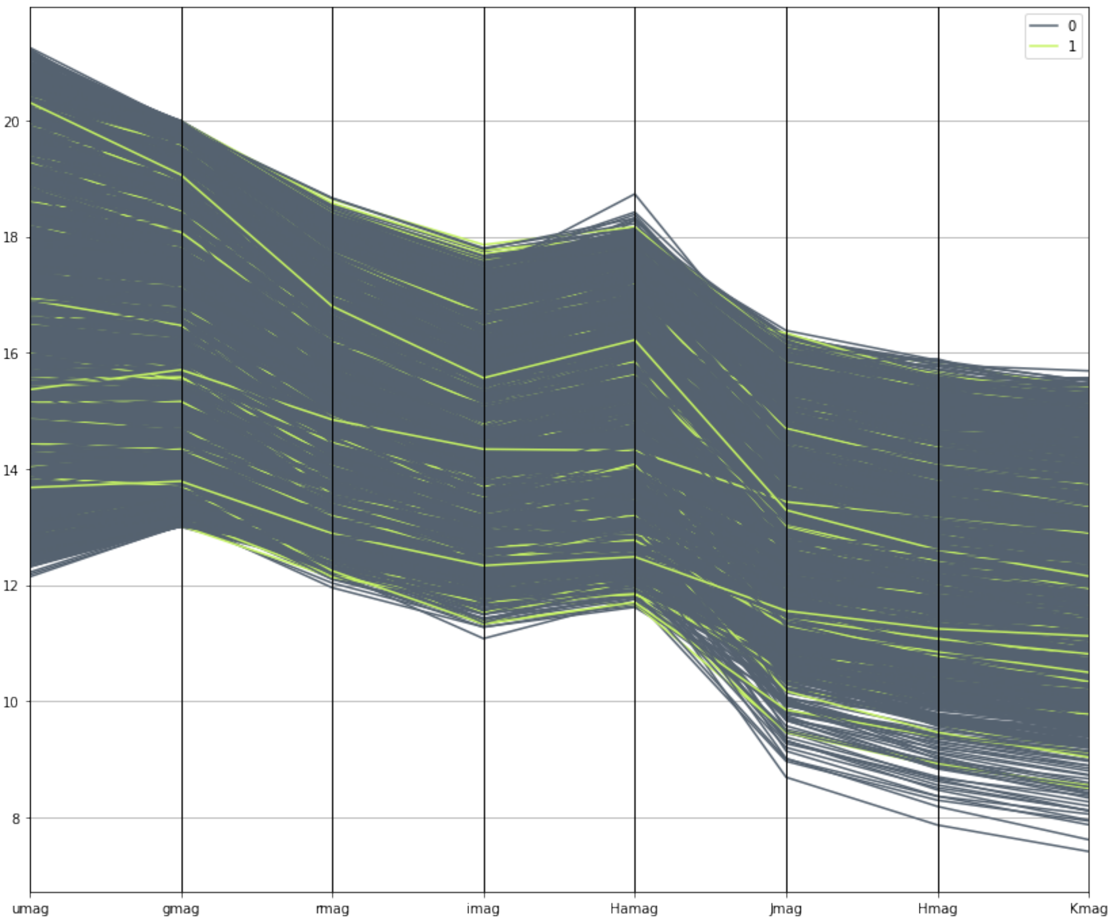
```
Text(0.5, 1.0, 'Per class boxplots')
```


20
18
16
14
12
10
8
umag
gmag
rmag
imag
Hamag
Jmag
Hmag
Kmag

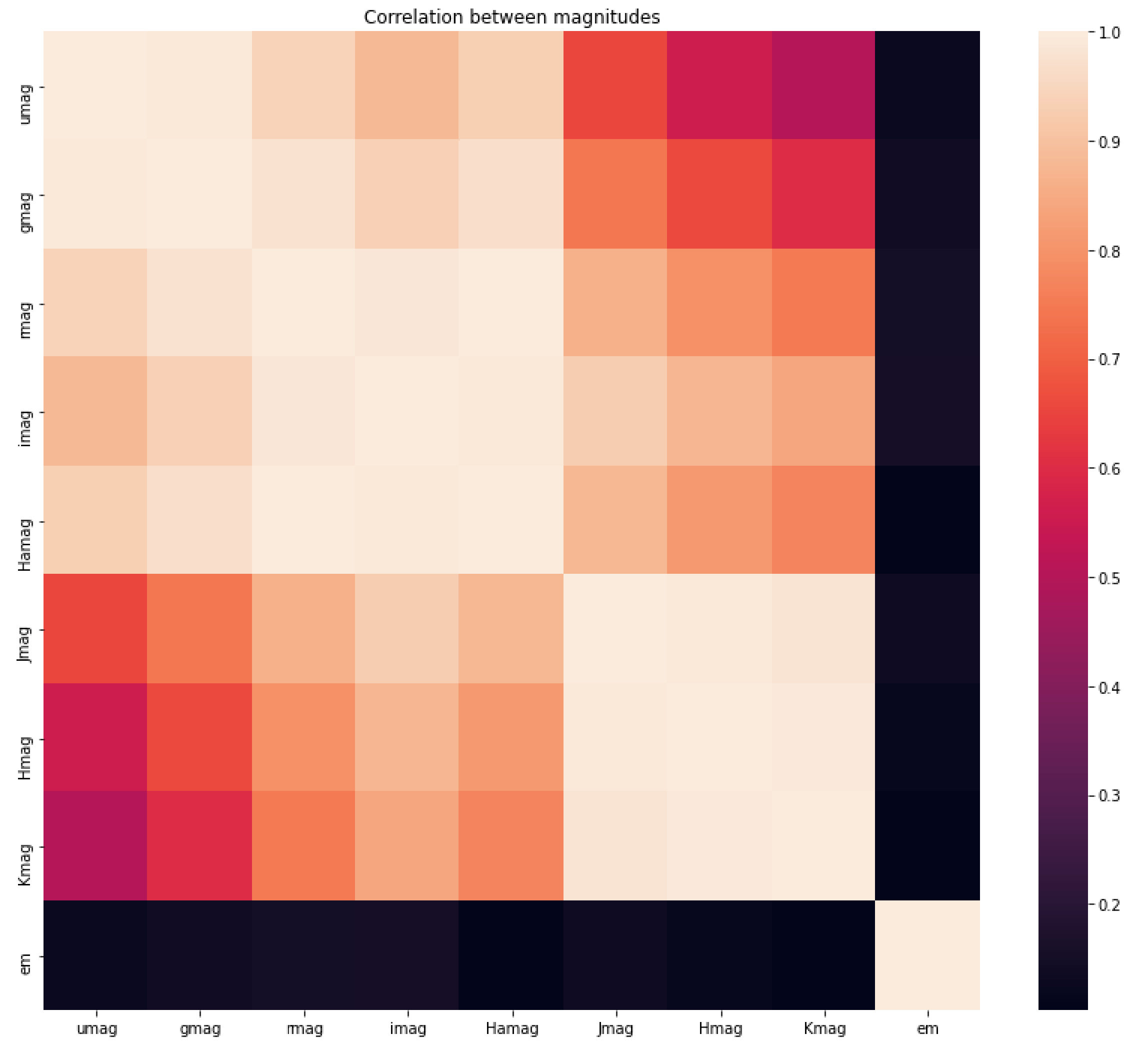


```
[5]: pd.plotting.parallel_coordinates(xy,"em",color=('#556270','#C7F464'))
```

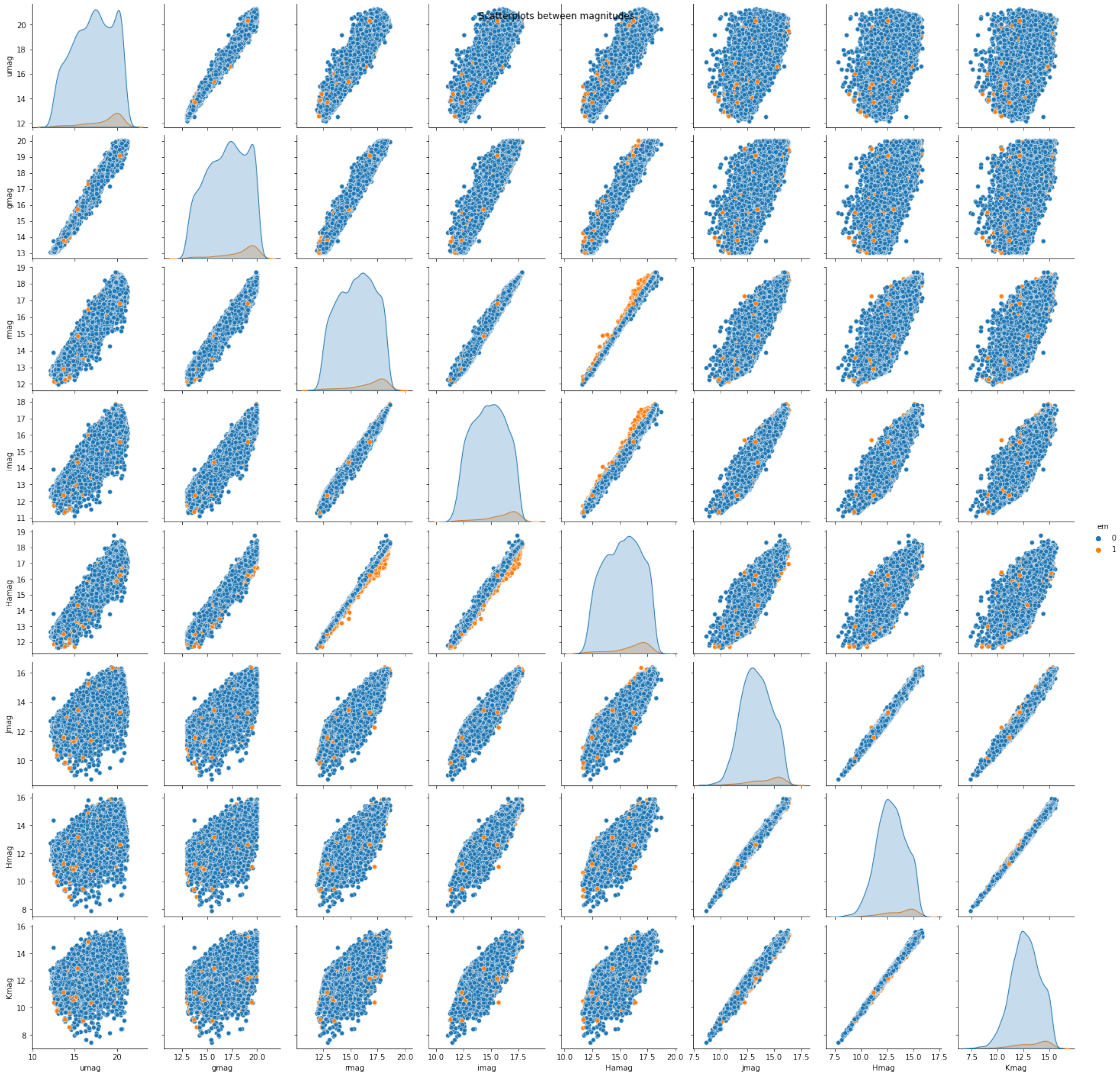
```
[5]: <AxesSubplot:>
```



[6]:
```
sn.heatmap(xy.corr().abs())
plt.title("Correlation between magnitudes")
plt.show()

sn.pairplot(xy,hue="em")
plt.suptitle("Scatterplots between magnitudes")
# axes=pd.plotting.scatter_matrix(x,c=y["em"],alpha=0.
↪9,grid=False,figsize=(14,12))
```



```
[6]: Text(0.5, 0.98, 'Scatterplots between magnitudes')
```



2 Outlier detection via confidence interval

```
[7]: from scipy import stats
     m = len(x.columns) # number of columns = number of hypothesis
     confidence= 0.99
     adjusted_confidence = 1- (1-confidence)/m  # bonferroni-adjusted confidence
     max_zscore = stats.norm.ppf(adjusted_confidence)
     print(f"Confidence  (desired): {confidence}")
     print(f"Confidence (adjusted): {adjusted_confidence}")
     print(f"Z-score    (adjusted): {max_zscore}")

     indices = (np.abs(stats.zscore(x-x.mean())) > max_zscore).any(axis=1)
     outliers_x = x[indices]
```

```
if dataset_name != "all_em":
    outliers_metadata = metadata[indices]
    outliers_x = pd.concat([outliers_x,outliers_metadata],axis=1)
outliers_x
```

```
Confidence  (desired): 0.99
Confidence (adjusted): 0.99875
Z-score    (adjusted): 3.023341439739154
```

```
[7]:         umag    gmag    rmag    imag   Hamag    Jmag   Hmag   Kmag    DEJ2000  \
     40    17.566  15.969  13.405  11.925  12.928   9.317  8.560  8.115 -57.545887
     46    17.929  16.214  13.487  11.917  12.970   9.144  8.367  7.875 -57.463467
     98    16.304  15.184  12.878  11.698  12.404   9.383  8.702  8.270 -57.016295
     131   16.936  15.480  12.953  11.486  12.341   8.693  7.870  7.414 -57.407289
     152   20.597  18.508  15.374  13.514  14.826  10.056  9.057  8.424 -57.090546
     241   17.558  16.161  13.570  12.095  13.049   9.327  8.474  7.953 -57.546570
     301   14.975  14.257  12.236  11.081  11.762   8.968  8.297  7.940 -57.917252
     437   19.141  17.146  14.245  12.551  13.746   9.479  8.656  8.113 -57.159384
     455   15.700  14.928  12.883  11.681  12.432   9.315  8.620  8.200 -58.052421
     549   14.600  14.320  12.575  11.566  12.225   9.273  8.673  8.371 -57.758620
     1120  14.979  14.243  12.437  11.366  11.680   9.488  8.925  8.498 -58.818031
     1713  15.168  14.395  12.444  11.318  11.885   9.005  8.365  8.059 -59.199053
     1937  17.744  16.371  13.813  12.309  13.302   9.674  8.837  8.406 -58.782316
     2245  16.736  16.161  13.962  12.592  13.489   9.756  8.880  8.334 -59.783881
     4753  16.710  15.442  13.140  11.778  12.642   9.577  8.856  8.469 -60.884853
     5359  16.351  15.423  13.115  11.700  12.603   9.018  8.190  7.617 -61.330502
     5605  15.297  14.487  12.594  11.352  12.067   9.205  8.511  8.126 -61.469052
     5722  20.484  18.317  15.149  13.372  14.604   9.970  9.007  8.402 -61.949044

              RAJ2000  logTeff      A0  chi2     Rv   r2mag     mu  VPHAS-OB1
     40    152.325021    4.324   7.995  1.57  3.538  13.394   8.41        109
     46    152.598693    4.366   8.524  2.09  3.528  13.506   8.49        130
     98    153.926987    4.352   7.143  3.60  3.638  12.880   8.89        261
     131   153.668178    4.341   8.350  1.83  3.794  12.942   7.76        331
     152   154.220685    4.417  10.406  4.29  3.769  15.327   9.29        376
     241   154.177204    4.404   8.544  0.54  3.787  13.571   8.89        559
     301   154.224633    4.488   6.844  0.28  3.769  12.251   9.91        708
     437   156.036587    4.315   9.147  4.96  3.589  14.247   8.19       1006
     455   155.131682    4.424   7.261  0.84  4.026  12.882   9.43       1041
     549   156.004990    4.530   6.772  4.69  4.400  12.582  10.69       1216
     1120  158.143987    4.316   6.007  6.16  3.785  12.286   8.94       2881
     1713  160.281809    4.363   6.855  6.44  4.013  12.356   8.68       4755
     1937  161.481177    4.470   8.375  2.12  3.647  13.814  10.03       5468
     2245  161.403107    4.624   8.616  3.46  4.526  13.954  11.66       6480
     4753  167.855369    4.365   7.189  4.68  3.479  13.127   9.22      12768
     5359  168.918911    4.548   8.415  3.48  4.098  13.109  10.10      13765
     5605  169.767548    4.360   6.790  7.18  3.954  12.545   8.81      14229
```

```
5722  170.078909    4.369  10.121  1.96  3.680  15.143   8.86      14506
```

3 Outlier detection via IQR

[8]:

```
iqr_factor=1.5
q25,q75=x.quantile(0.25),x.quantile(0.75)
iqr=q75-q25
min_values = q25-iqr_factor*iqr
max_values = q75+iqr_factor*iqr
# ou
indices = (np.logical_or(x<min_values,x>max_values)).any(axis=1)
outliers_x = x[indices]
if dataset_name != "all_em":
    outliers_metadata = metadata[indices]
    outliers_x = pd.concat([outliers_x,outliers_metadata],axis=1)
outliers_x
```

[8]:

```
        umag    gmag    rmag    imag   Hamag    Jmag   Hmag   Kmag    DEJ2000  \
33    17.230  15.975  13.523  12.100  13.031   9.670  8.930  8.536 -57.331160
40    17.566  15.969  13.405  11.925  12.928   9.317  8.560  8.115 -57.545887
46    17.929  16.214  13.487  11.917  12.970   9.144  8.367  7.875 -57.463467
98    16.304  15.184  12.878  11.698  12.404   9.383  8.702  8.270 -57.016295
131   16.936  15.480  12.953  11.486  12.341   8.693  7.870  7.414 -57.407289
152   20.597  18.508  15.374  13.514  14.826  10.056  9.057  8.424 -57.090546
185   16.588  15.619  13.410  12.131  12.934   9.722  9.025  8.567 -57.348754
241   17.558  16.161  13.570  12.095  13.049   9.327  8.474  7.953 -57.546570
301   14.975  14.257  12.236  11.081  11.762   8.968  8.297  7.940 -57.917252
306   14.794  14.270  12.475  11.404  12.034   9.516  8.969  8.667 -58.156718
437   19.141  17.146  14.245  12.551  13.746   9.479  8.656  8.113 -57.159384
455   15.700  14.928  12.883  11.681  12.432   9.315  8.620  8.200 -58.052421
469   17.844  16.488  14.034  12.574  13.513   9.870  9.100  8.600 -57.769781
549   14.600  14.320  12.575  11.566  12.225   9.273  8.673  8.371 -57.758620
558   14.593  14.293  12.563  11.542  12.169   9.450  8.950  8.520 -57.759795
1120  14.979  14.243  12.437  11.366  11.680   9.488  8.925  8.498 -58.818031
1713  15.168  14.395  12.444  11.318  11.885   9.005  8.365  8.059 -59.199053
1783  14.353  13.961  12.394  11.538  11.903   9.465  8.881  8.555 -59.454645
1937  17.744  16.371  13.813  12.309  13.302   9.674  8.837  8.406 -58.782316
2245  16.736  16.161  13.962  12.592  13.489   9.756  8.880  8.334 -59.783881
2361  15.297  14.605  12.680  11.556  12.232   9.689  9.120  8.725 -59.220889
4753  16.710  15.442  13.140  11.778  12.642   9.577  8.856  8.469 -60.884853
5170  16.262  15.748  13.700  12.411  13.154   9.975  9.167  8.738 -61.242734
5359  16.351  15.423  13.115  11.700  12.603   9.018  8.190  7.617 -61.330502
5563  14.949  14.260  12.473  11.405  12.035   9.485  8.892  8.531 -61.559307
5605  15.297  14.487  12.594  11.352  12.067   9.205  8.511  8.126 -61.469052
5722  20.484  18.317  15.149  13.372  14.604   9.970  9.007  8.402 -61.949044
```

```
         RAJ2000  logTeff      A0  chi2     Rv   r2mag     mu  VPHAS-OB1
33    152.448798    4.507   7.852  1.60  3.525  13.512  10.64         85
40    152.325021    4.324   7.995  1.57  3.538  13.394   8.41        109
46    152.598693    4.366   8.524  2.09  3.528  13.506   8.49        130
98    153.926987    4.352   7.143  3.60  3.638  12.880   8.89        261
131   153.668178    4.341   8.350  1.83  3.794  12.942   7.76        331
152   154.220685    4.417  10.406  4.29  3.769  15.327   9.29        376
185   154.072129    4.423   7.475  1.52  3.836  13.438   9.80        444
241   154.177204    4.404   8.544  0.54  3.787  13.571   8.89        559
301   154.224633    4.488   6.844  0.28  3.769  12.251   9.91        708
306   153.990057    4.509   6.190  1.75  3.722  12.432  10.92        724
437   156.036587    4.315   9.147  4.96  3.589  14.247   8.19       1006
455   155.131682    4.424   7.261  0.84  4.026  12.882   9.43       1041
469   155.582926    4.366   8.195  3.05  3.791  14.009   9.21       1069
549   156.004990    4.530   6.772  4.69  4.400  12.582  10.69       1216
558   156.009539    4.562   6.496  2.25  4.159  12.569  11.33       1226
1120  158.143987    4.316   6.007  6.16  3.785  12.286   8.94       2881
1713  160.281809    4.363   6.855  6.44  4.013  12.356   8.68       4755
1783  160.248764    4.300   5.901  6.84  4.484  12.401   8.76       4980
1937  161.481177    4.470   8.375  2.12  3.647  13.814  10.03       5468
2245  161.403107    4.624   8.616  3.46  4.526  13.954  11.66       6480
2361  162.343201    4.449   6.294  5.33  3.607  12.676  10.38       6880
4753  167.855369    4.365   7.189  4.68  3.479  13.127   9.22      12768
5170  168.747828    4.632   7.747  3.22  4.254  13.696  12.28      13501
5359  168.918911    4.548   8.415  3.48  4.098  13.109  10.10      13765
5563  169.582737    4.338   6.086  1.34  3.835  12.517   9.12      14153
5605  169.767548    4.360   6.790  7.18  3.954  12.545   8.81      14229
5722  170.078909    4.369  10.121  1.96  3.680  15.143   8.86      14506
```

4 Analysis of q-features (q_3) (all magnitudes)

[9]:

```
x_np=x.to_numpy()
import qfeatures
coefficients = dataset_module.coefficients
systems = dataset_module.systems
coefficients_np = np.array([coefficients[k] for k in x.columns])
systems = [systems[k] for k in x.columns]
q=qfeatures.calculate(x_np,coefficients_np,x.columns,systems,combination_size=3)
m = q.magnitudes

q_df = pd.DataFrame(m, columns = q.column_names)
q_df.describe()
```

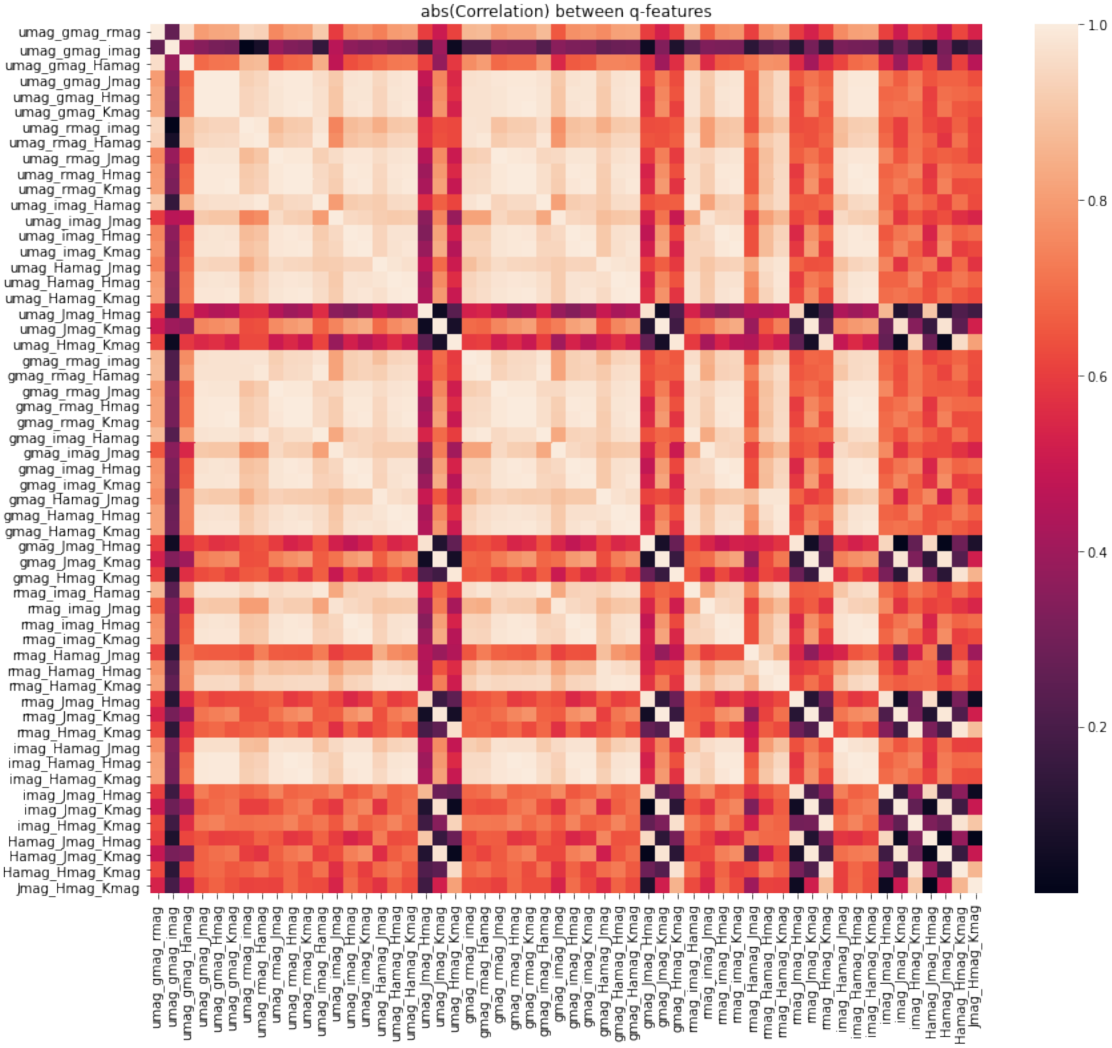
[9]:

```
       umag_gmag_rmag  umag_gmag_imag  umag_gmag_Hamag  umag_gmag_Jmag  \
count     5877.000000     5877.000000      5877.000000     5877.000000
mean        -0.409710       -1.150416        -0.635135       -5.216200
```

```
std         0.258130        0.130671         0.224254         1.549386
min        -1.183961       -2.091596        -1.406519       -10.758042
25%        -0.599502       -1.227567        -0.791967        -6.291792
50%        -0.426035       -1.124708        -0.651668        -5.218694
75%        -0.227082       -1.053386        -0.482411        -4.125556
max         0.672143       -0.832544         0.275799        -0.477486

       umag_gmag_Hmag  umag_gmag_Kmag  umag_rmag_imag  umag_rmag_Hamag  \
count     5877.000000     5877.000000     5877.000000      5877.000000
mean        -9.341466      -14.959737        0.626648         1.275020
std          3.078338        5.187979        0.614700         0.885778
min        -20.305761      -33.831131       -1.319994        -1.448589
25%        -11.489065      -18.563993        0.177035         0.620065
50%         -9.338935      -14.950752        0.605485         1.244047
75%         -7.134609      -11.265647        1.063117         1.900215
max          0.201065        1.158850        3.173503         4.805280

       umag_rmag_Jmag  umag_rmag_Hmag  …  imag_Hamag_Jmag  imag_Hamag_Hmag  \
count     5877.000000     5877.000000  …      5877.000000      5877.000000
mean        -4.840076      -10.474658  …         0.664383         1.713869
std          1.527176        3.667144  …         0.260581         0.676747
min        -11.826778      -25.465348  …        -0.156500        -0.411696
25%         -5.907778      -13.052304  …         0.482347         1.232391
50%         -4.782889      -10.416435  …         0.657583         1.705935
75%         -3.768000       -7.872522  …         0.844611         2.190152
max         -0.261444        0.832565  …         1.753681         4.289304

       imag_Hamag_Kmag  imag_Jmag_Hmag  imag_Jmag_Kmag  imag_Hmag_Kmag  \
count      5877.000000     5877.000000     5877.000000     5877.000000
mean          3.124834        0.478241       -0.723156        0.950934
std           1.254977        0.269327        0.442834        0.476000
min          -0.772131       -0.953239       -2.635706       -2.127608
25%           2.239379        0.305826       -0.988647        0.630033
50%           3.103575        0.482783       -0.677353        0.932301
75%           4.002680        0.661935       -0.413118        1.275706
max           7.736477        1.935283        0.788412        2.847046

       Hamag_Jmag_Hmag  Hamag_Jmag_Kmag  Hamag_Hmag_Kmag  Jmag_Hmag_Kmag
count      5877.000000      5877.000000      5877.000000     5877.000000
mean          0.531117        -1.192100         1.100577        0.249903
std           0.376616         0.683245         0.618342        0.141120
min          -1.730391        -4.544699        -2.820353       -0.617078
25%           0.302348        -1.585222         0.691098        0.154026
50%           0.538217        -1.109229         1.080824        0.240222
75%           0.786391        -0.722118         1.519471        0.338431
max           2.403304         0.906758         3.560882        1.061144
```

```
[8 rows x 56 columns]
```

[10]:
```
sn.heatmap(q_df.corr().abs())
plt.title("abs(Correlation) between q-features")
plt.show()
```



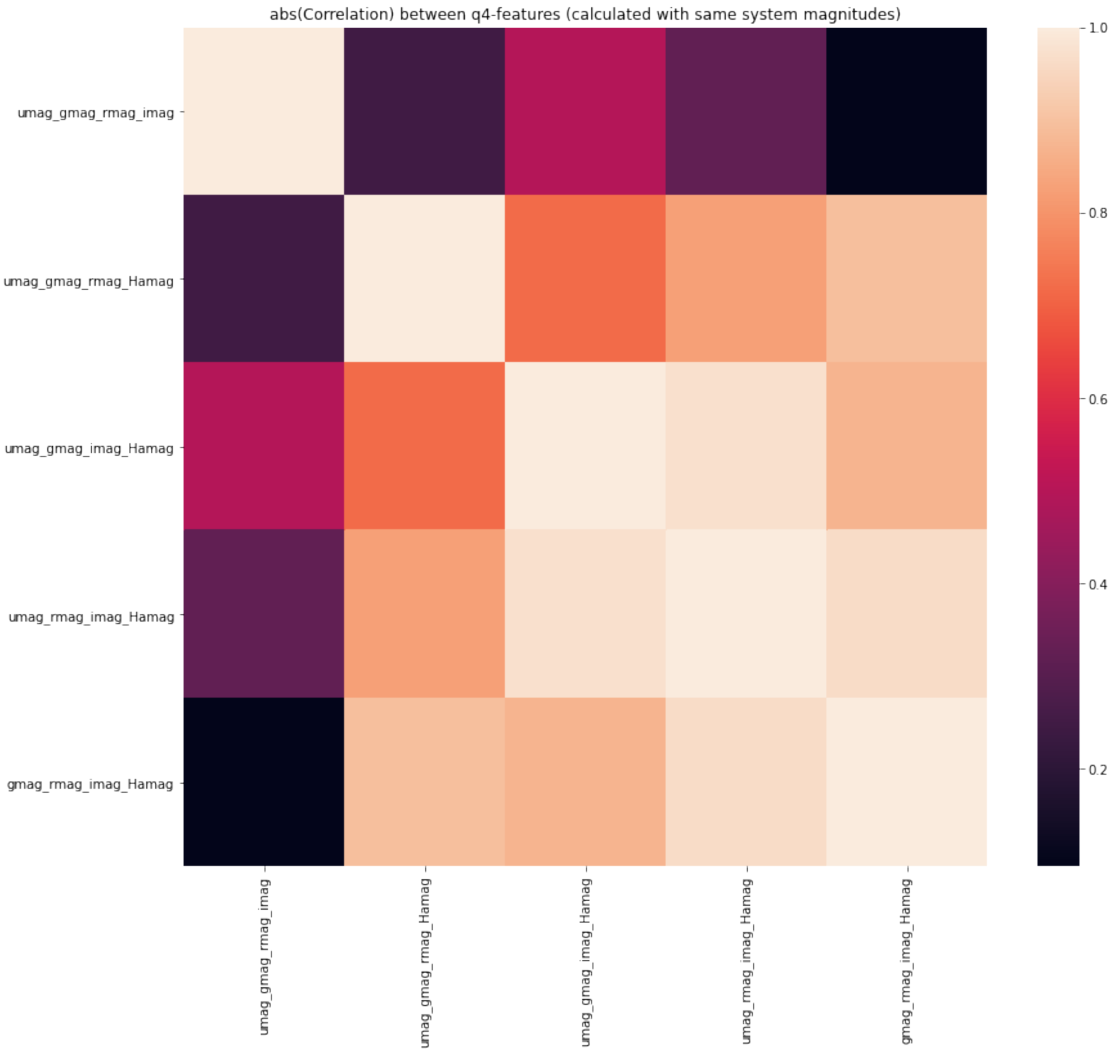
5 Analysis of q-features (q_4) (calculated by system to avoid combinatory explosion)

[11]:
```
x_np=x.to_numpy()
import qfeatures
coefficients = dataset_module.coefficients
systems = dataset_module.systems
```

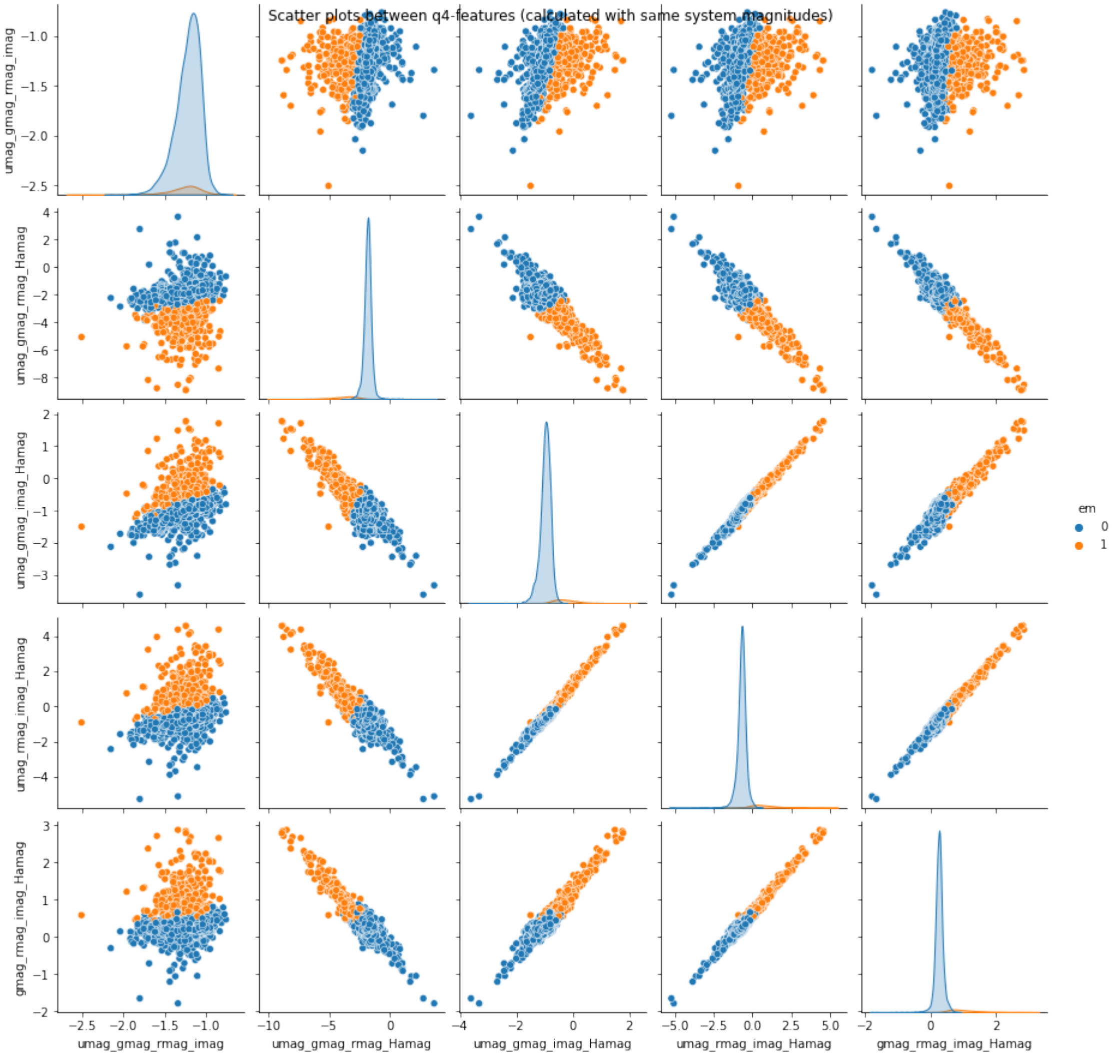
```
coefficients_np = np.array([coefficients[k] for k in x.columns])
systems = [systems[k] for k in x.columns]
q= qfeatures.calculate(x_np,coefficients_np,x.
 ↪columns,systems,combination_size=4,by_system=True)

m = q.magnitudes

q_df = pd.DataFrame(m, columns = q.column_names)
q_df.describe()
```

```
[11]:        umag_gmag_rmag_imag  umag_gmag_rmag_Hamag  umag_gmag_imag_Hamag  \
      count          5877.000000           5877.000000           5877.000000
      mean             -1.227544             -1.954241             -0.940245
      std               0.153862              0.677955              0.298900
      min              -2.504267             -8.901765             -3.602581
      25%              -1.312917             -2.053176             -1.072070
      50%              -1.205000             -1.851647             -0.952535
      75%              -1.118600             -1.685059             -0.836233
      max              -0.764783              3.644647              1.778744

             umag_rmag_imag_Hamag  gmag_rmag_imag_Hamag
      count           5877.000000           5877.000000
      mean              -0.639082              0.301163
      std                0.535721              0.254820
      min               -5.258953             -1.790860
      25%               -0.847581              0.199279
      50%               -0.690721              0.268465
      75%               -0.530558              0.336535
      max                4.594419              2.874535
```

```
[12]: sn.heatmap(q_df.corr().abs())
      _=plt.title("abs(Correlation) between q4-features (calculated with same system␣
       ↪magnitudes)")
```



```
[13]: q_dfy=pd.concat([q_df,y],axis=1)
      sn.pairplot(q_dfy,hue="em")
      _=plt.suptitle("Scatter plots between q4-features (calculated with same system␣
       ↪magnitudes)")
```



Scatter plots between q4-features (calculated with same system magnitudes)