Computerbasiertes Statistik und stochastische Simulation_Seil_Hong_108016263063

July 14, 2020

[35]:
```
import pandas as pds
import pandas_ods_reader as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
from sklearn.cluster import KMeans
import math
from sklearn.preprocessing import StandardScaler

df = pd.read_ods(r"C:
 ↪\Users\sel20\Desktop\computerbasiertes\daten_compstat_sose_aktualisiert.
 ↪ods",1)#read a ods file
df_0 = df.fillna(0) #replace NaN with 0
df_typ0 = df_0[df_0['Typ Klausur'] < 1.0] #split the dataframe
df_typ1 = df_0[df_0['Typ Klausur'] >=  1.0]

pds.set_option('mode.chained_assignment', None)

del df_typ0['ID']
del df_typ1['ID']
del df_typ0['Typ Klausur']# dont need the factor 'Typ Klausur' because of the␣
 ↪splitting the dataframe.
del df_typ1['Typ Klausur']
del df_typ0['Punkte Aufgabe Typ 1']#The factor 'Punkte Aufgabe Typ 1' is not␣
 ↪necessary
print(df_typ0.head(10)) #print 10 datas in front
print(df_typ1.head(10))

#partial correlation
#show all correlationscoefficients in the dataframe 'typ0' and 'typ1'
corr0=df_typ0.corr(method = 'pearson')
corr1=df_typ1.corr(method = 'pearson')
print(corr0)
print(corr1)

#calculate partial correlation in the dataframe 'typ0'
```

```
pBPnote = -0.736717
pPunkte_Rechenteil_note = -0.855729
pBP_Punkte_Rechenteil = 0.604798
partialcorr1 = (pPunkte_Rechenteil_note-pBP_Punkte_Rechenteil*pBPnote) / math.
 ↪sqrt((1-pBPnote*pBPnote)*(1-pBP_Punkte_Rechenteil*pBP_Punkte_Rechenteil))
print('Tpy0 : Partial correlation between BP and Note without Punkte Rechenteil␣
 ↪: ',partialcorr1) # partial correlation

#calculate partial correlation in the dataframe 'typ1'
pBPnote = -0.740851 # the value is negative because Note '1.0' sehr gut, 5.0␣
 ↪'nicht bestehend'
pPunkte_Rechenteil_note = -0.724988
pBP_Punkte_Rechenteil = 0.573876
partialcorr2 = (pPunkte_Rechenteil_note-pBP_Punkte_Rechenteil*pBPnote) / math.
 ↪sqrt((1-pBPnote*pBPnote)*(1-pBP_Punkte_Rechenteil*pBP_Punkte_Rechenteil))
print('Tpy1 : Partial correlation between BP and Note without Punkte Rechenteil␣
 ↪: ',partialcorr2)

#clusteranalyse
data_points = df_typ0.values # convert dataframe to numpy array
kmeans = KMeans(n_clusters = 5).fit(data_points) #kmean ++
kmeans.labels_
kmeans.cluster_centers_
df_typ0['cluster_id'] = kmeans.labels_
sns.lmplot('Punkte Rechenteil','Note', data = df_typ0, fit_reg= False,␣
 ↪scatter_kws={"s" : 100},hue = "cluster_id")
plt.title('after kmean clustering of Typ 0')

data_points = df_typ1.values # convert dataframe to numpy array
kmeans = KMeans(n_clusters = 5).fit(data_points) #kmean ++
kmeans.labels_
kmeans.cluster_centers_
df_typ1['cluster_id'] = kmeans.labels_
sns.lmplot('Punkte Aufgabe Typ 1','Note', data = df_typ1, fit_reg= False,␣
 ↪scatter_kws={"s" : 100},hue = "cluster_id")
plt.title('after kmean clustering of Typ 1')

#maincomponent analysis
Y = df_typ1['Note']

df_std = StandardScaler().fit_transform(df_typ1)#rescaling feature vectors to␣
 ↪all have the same scale

features = df_std.T

covariance_matrix = np.cov(features)#covariancematrix
```

```
eig_vals, eig_vecs = np.linalg.eig(covariance_matrix)
print('\nEigenvalues \n%s' %eig_vals)

IR=eig_vals[0] / sum(eig_vals)
print('First eingevector has',IR,'enough variances')

projected_df = df_std.dot(eig_vecs.T[0])
print(projected_df)

result=pds.DataFrame(projected_df,columns=['PC1'])
result['y-axis'] = 0.0
Y_index = Y.index+1
ID = pds.DataFrame(Y_index, columns =['ID'])
index = [i for i in range(55)]
Y.index = index
for i in range(55) :
    if Y[i]<=1.3 :
        Y[i] = "sehr gut"
    elif Y[i]>1.3 and Y[i]<=2.3 :
            Y[i] = "gut"
    elif Y[i]>2.3 and Y[i]<=3.3 :
            Y[i] = "befriedigend"
    elif Y[i]>3.3 and Y[i]<=4.0 :
                Y[i] = "ausreichend"
    else : Y[i] = "nicht ausreichend"

result['Note'] = Y
result2=pds.merge(result,ID,left_on = result.index, right_on = ID.index)
print('PCA result with ID')
print(result2.head(10))

result2 =sns.lmplot('PC1','y-axis', data = result, fit_reg = False, scatter_kws␣
↪= {"s" : 10}, hue = "Note")
plt.title('PCA result of typ1')

Z = df_typ0['Note']


df_typ0_std = StandardScaler().fit_transform(df_typ0)#rescaling feature vectors␣
↪to all have the same scale

features_typ0 = df_typ0_std.T

covariance_matrix_typ0 = np.cov(features_typ0)#covariancematrix
eig_vals_typ0, eig_vecs_typ0 = np.linalg.eig(covariance_matrix_typ0)
print('\nEigenvalues \n%s' %eig_vals_typ0)
```

```
IR_typ0=eig_vals_typ0[0] / sum(eig_vals_typ0)
print('First eingevector has',IR_typ0,'enough variances')

projected_df_typ0 = df_typ0_std.dot(eig_vecs_typ0.T[0])
print(projected_df_typ0)

result_typ0=pds.DataFrame(projected_df_typ0,columns=['PC1'])
result_typ0['y-axis'] = 0.0
Z_index = Z.index+1
ID_typ0 = pds.DataFrame(Z_index, columns =['ID'])
index_typ0 = [i for i in range(79)]
Z.index = index_typ0
for i in range(79) :
    if Z[i]<=1.3 :
        Z[i] = "sehr gut"
    elif Z[i]>1.3 and Z[i]<=2.3 :
            Z[i] = "gut"
    elif Z[i]>2.3 and Z[i]<=3.3 :
            Z[i] = "befriedigend"
    elif Z[i]>3.3 and Z[i]<=4.0 :
                Z[i] = "ausreichend"
    else : Z[i] = "nicht ausreichend"
result_typ0['Note'] = Z
result2_typ0=pds.merge(result_typ0,ID_typ0,left_on = result_typ0.index,␣
 ↪right_on = ID_typ0.index)
print('PCA result with ID')
print(result2_typ0.head(10))

a=sns.lmplot('PC1','y-axis', data = result_typ0, fit_reg = False, scatter_kws =␣
 ↪{"s" : 10}, hue = "Note")
plt.title('PCA result of typ0')
```

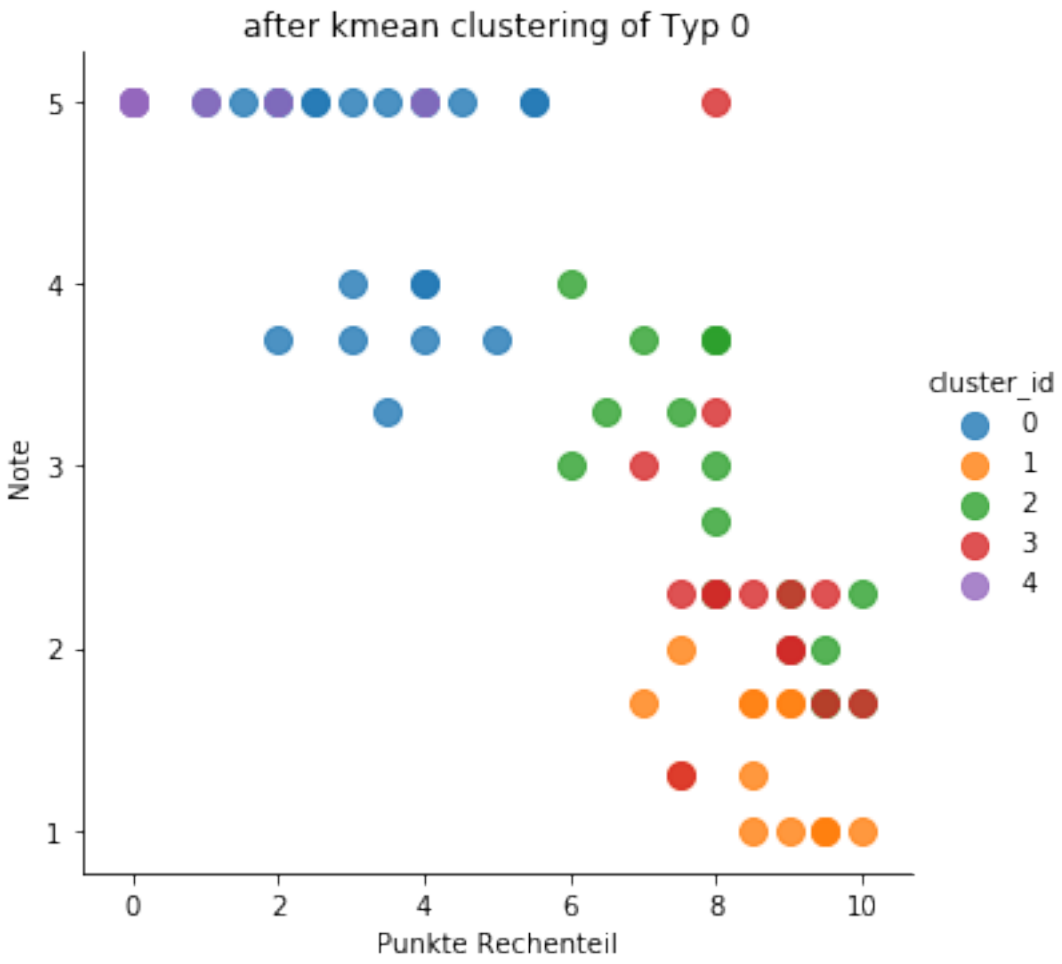
```
     BP  Punkte Rechenteil  Punkte Multiple-Choice-Teil  Punkte gesamt  Note
1   0.0                0.0                          0.0            0.0   5.0
2   0.0                0.0                          0.0            0.0   5.0
7   0.0                0.0                          0.0            0.0   5.0
9   0.0                0.0                          0.0            0.0   5.0
10  0.0                0.0                          0.0            0.0   5.0
16  0.0                0.0                          2.0            0.0   5.0
17  0.0                0.0                          3.0            0.0   5.0
20  0.0                4.0                          0.0            4.0   5.0
21  3.0                5.5                          6.0            5.5   5.0
24  2.0                0.0                          5.0            0.0   5.0
     BP  Punkte Rechenteil  Punkte Multiple-Choice-Teil  Punkte Aufgabe Typ 1  \
0   0.0                0.0                          0.0                   0.0
3   0.0                0.0                          0.0                   0.0
4   0.0                0.0                          0.0                   0.0
```

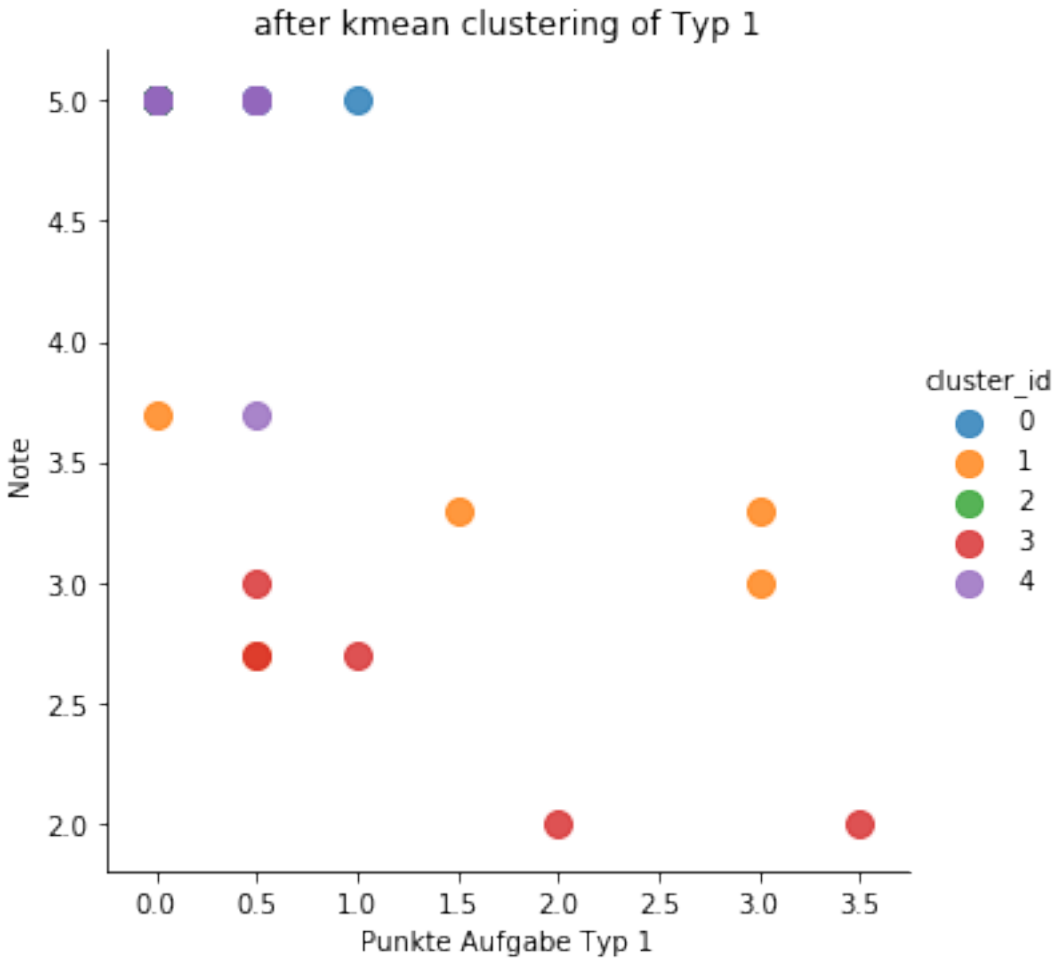
```
5   0.0                0.0                          0.0                    0.0
6   0.0                0.0                          0.0                    0.0
8   0.0                0.0                          0.0                    0.0
11  0.0                0.0                          0.0                    0.0
12  0.0                0.0                          0.0                    0.0
13  0.0                0.0                          0.0                    0.0
14  0.0                0.0                          0.0                    0.0

    Punkte gesamt  Note
0             0.0   5.0
3             0.0   5.0
4             0.0   5.0
5             0.0   5.0
6             0.0   5.0
8             0.0   5.0
11            0.0   5.0
12            0.0   5.0
13            0.0   5.0
14            0.0   5.0
                                    BP  Punkte Rechenteil  \
BP                            1.000000           0.604798
Punkte Rechenteil             0.604798           1.000000
Punkte Multiple-Choice-Teil   0.383799           0.603076
Punkte gesamt                 0.602967           0.999853
Note                         -0.736717          -0.855729

                             Punkte Multiple-Choice-Teil  Punkte gesamt  \
BP                                              0.383799       0.602967
Punkte Rechenteil                               0.603076       0.999853
Punkte Multiple-Choice-Teil                     1.000000       0.604879
Punkte gesamt                                   0.604879       1.000000
Note                                           -0.700483      -0.855908

                                 Note
BP                           -0.736717
Punkte Rechenteil            -0.855729
Punkte Multiple-Choice-Teil  -0.700483
Punkte gesamt                -0.855908
Note                          1.000000
                                    BP  Punkte Rechenteil  \
BP                            1.000000           0.573876
Punkte Rechenteil             0.573876           1.000000
Punkte Multiple-Choice-Teil   0.290509           0.535631
Punkte Aufgabe Typ 1          0.534877           0.609322
Punkte gesamt                 0.606264           0.984061
Note                         -0.826639          -0.724988

                             Punkte Multiple-Choice-Teil  \
```

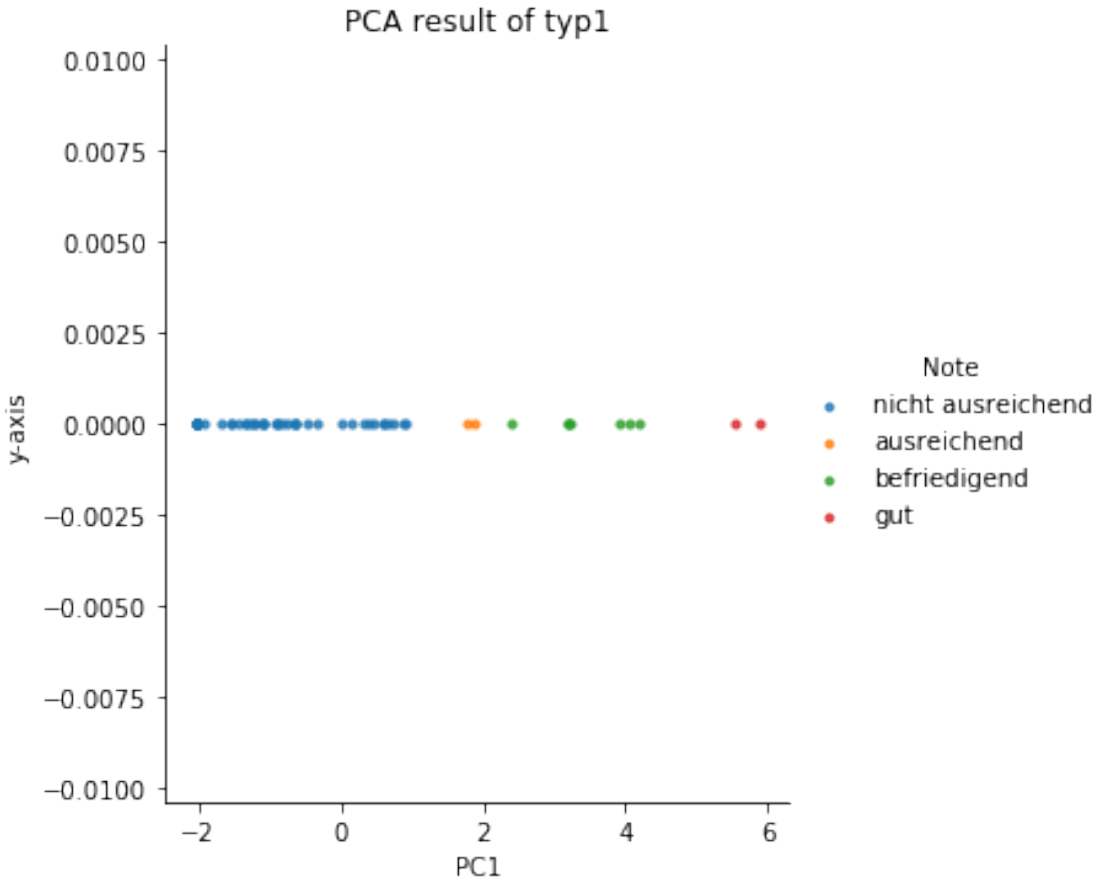
```
BP                                          0.290509
Punkte Rechenteil                           0.535631
Punkte Multiple-Choice-Teil                 1.000000
Punkte Aufgabe Typ 1                        0.273784
Punkte gesamt                               0.515300
Note                                       -0.370945

                            Punkte Aufgabe Typ 1  Punkte gesamt      Note
BP                                      0.534877       0.606264 -0.826639
Punkte Rechenteil                       0.609322       0.984061 -0.724988
Punkte Multiple-Choice-Teil             0.273784       0.515300 -0.370945
Punkte Aufgabe Typ 1                    1.000000       0.740615 -0.741978
Punkte gesamt                           0.740615       1.000000 -0.780765
Note                                   -0.741978      -0.780765  1.000000
Tpy0 : Partial correlation between BP and Note without Punkte Rechenteil :
-0.7616612878223321
Tpy1 : Partial correlation between BP and Note without Punkte Rechenteil :
-0.5450900782375377

Eigenvalues
[ 4.30054011e+00  1.15374263e+00  8.05201840e-01  5.40449873e-01
  8.83972772e-02  2.41297901e-01 -1.18110656e-16]
First eingevector has 0.6031926390978181 enough variances
[-2.01878847e+00 -2.01878847e+00 -2.01878847e+00 -2.01878847e+00
 -2.01878847e+00 -2.01878847e+00 -2.01878847e+00 -2.01878847e+00
 -2.01878847e+00 -2.01878847e+00 -1.93613656e+00 -1.08221560e+00
 -1.20155209e+00 -1.54367237e+00 -1.54367237e+00 -1.24520697e+00
 -1.66300886e+00 -1.25903205e+00 -6.34361644e-01 -1.31870030e+00
 -1.31870030e+00 -9.14723494e-01 -1.43803678e+00  4.53979908e-01
 -1.09372822e+00 -1.09372822e+00 -4.68968363e-01 -8.09087906e-01
 -8.68756149e-01  9.06151613e-01  5.86753453e-01 -7.58364621e-01
 -8.07375743e-04  7.33864286e-01 -6.43784074e-01  4.07748724e-01
 -6.43784074e-01  8.62448813e-01  1.46092079e-01 -3.31404144e-01
  3.20641696e-01  6.60460411e-01  6.00792167e-01 -8.91739821e-01
  1.87327013e+00  1.77301147e+00  3.18331842e+00  4.18683680e+00
  3.20882342e+00  3.92978900e+00  2.40264945e+00  5.86857608e+00
  5.54656009e+00  3.20155453e+00  4.03773454e+00]
PCA result with ID
   key_0       PC1  y-axis               Note  ID
0      0 -2.018788     0.0  nicht ausreichend   1
1      1 -2.018788     0.0  nicht ausreichend   4
2      2 -2.018788     0.0  nicht ausreichend   5
3      3 -2.018788     0.0  nicht ausreichend   6
4      4 -2.018788     0.0  nicht ausreichend   7
5      5 -2.018788     0.0  nicht ausreichend   9
6      6 -2.018788     0.0  nicht ausreichend  12
7      7 -2.018788     0.0  nicht ausreichend  13
8      8 -2.018788     0.0  nicht ausreichend  14
```

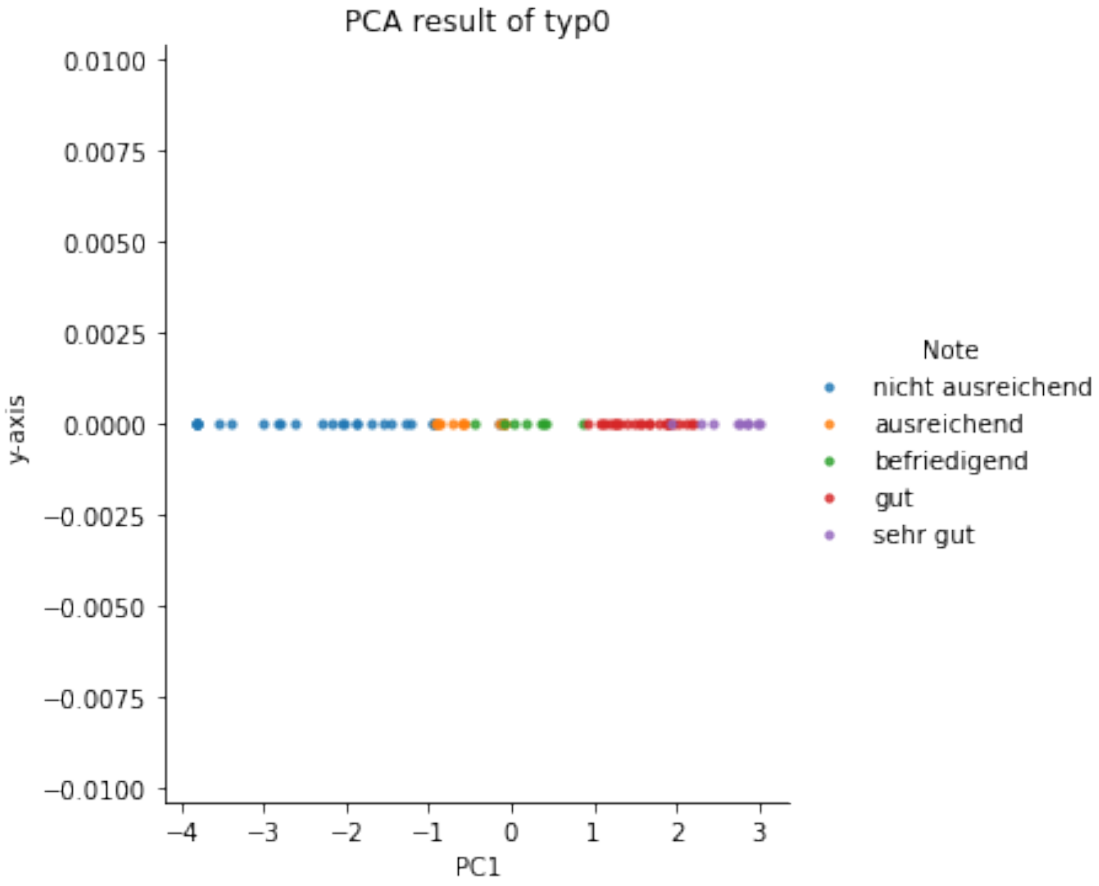
```
9      9 -2.018788     0.0  nicht ausreichend  15

Eigenvalues
[3.89296664e+00 1.27669341e+00 5.49247056e-01 3.03411461e-01
 5.44633806e-02 1.41125422e-04]
First eingevector has 0.6406147638363616 enough variances
[-3.80849769 -3.80849769 -3.80849769 -3.80849769 -3.80849769 -3.53839324
 -3.40334101 -2.63147269 -0.96066012 -3.02198623 -2.8148285  -0.96066012
 -2.8389803  -2.06998407 -2.30456061 -2.02238026 -1.86317624 -2.18158428
 -0.71233764 -1.70837665 -1.55684409 -1.87525213 -1.46974556 -1.21136711
 -1.26699227 -0.15185373 -0.89935424  0.38780294 -0.85140054 -0.603487
 -0.56754522 -0.88595326 -0.15275618 -0.08599138 -0.43748379  0.34923954
 -0.08100058 -0.91010506 -0.08100058 -0.08100058 -0.59169701  0.04138939
  1.27625241  1.67776843  0.85901951  0.17957046  2.1962003   0.42441914
  1.26417652  1.18073877  1.40363318  1.14085029  2.1841244   1.96162375
  1.30811954  1.79034383  1.54803184  2.10875188  1.07291244  1.65928217
  2.29443847  1.90725969  2.45364249  1.24824691  0.92983886  1.08904288
  2.84874814  1.94180748  1.50389589  1.57525774  2.99587627  2.02256456
  1.91874891  2.98380037  2.76129972  2.74922383  1.87484975  1.87484975
  2.84839825]
PCA result with ID
   key_0       PC1  y-axis               Note  ID
0      0 -3.808498     0.0  nicht ausreichend   2
1      1 -3.808498     0.0  nicht ausreichend   3
2      2 -3.808498     0.0  nicht ausreichend   8
3      3 -3.808498     0.0  nicht ausreichend  10
4      4 -3.808498     0.0  nicht ausreichend  11
5      5 -3.538393     0.0  nicht ausreichend  17
6      6 -3.403341     0.0  nicht ausreichend  18
7      7 -2.631473     0.0  nicht ausreichend  21
8      8 -0.960660     0.0  nicht ausreichend  22
9      9 -3.021986     0.0  nicht ausreichend  25
```

[35]: Text(0.5, 1, 'PCA result of typ0')


after kmean clustering of Typ 0
Note
Punkte Rechenteil
cluster_id
0
1
2
3
4
5
4
3
2
1
0
2
4
6
8
10


after kmean clustering of Typ 1
Note
5.0
4.5
4.0
3.5
3.0
2.5
2.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
Punkte Aufgabe Typ 1
cluster_id
0
1
2
3
4


PCA result of typ1
0.0100
0.0075
0.0050
0.0025
0.0000
−0.0025
−0.0050
−0.0075
−0.0100
y-axis
−2
0
2
4
6
PC1
Note
nicht ausreichend
ausreichend
befriedigend
gut


PCA result of typ0
0.0100
0.0075
0.0050
0.0025
0.0000
−0.0025
−0.0050
−0.0075
−0.0100
y-axis
−4
−3
−2
−1
0
1
2
3
PC1
Note
nicht ausreichend
ausreichend
befriedigend
gut
sehr gut