

```
In [1]: ### This script is for the course Adv BA's assignment 1, from Team 3
from pandas import Series, DataFrame
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [2]: ## 1.Load data
data = pd.read_excel('student survey 06.xlsx')
```

```
In [3]: ## 2.Exploration
# check features(variables)
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 100 entries, 0 to 99
Data columns (total 50 columns):
Nationality          99 non-null float64
Gender               99 non-null float64
Age                 99 non-null float64
Have_offer          97 non-null float64
Teaching_Core       98 non-null float64
Teaching_Elective   92 non-null float64
Teaching_FacQual    97 non-null float64
Teaching_FacKnow    98 non-null float64
Teaching_FacAvail   97 non-null float64
Teaching_FacAware   97 non-null float64
Teaching_Integrated 97 non-null float64
CC_Current          92 non-null float64
CC_Useful           82 non-null float64
CC_Worklevel        93 non-null float64
CC_Interpersonal    93 non-null float64
CC_Prepate          83 non-null float64
CC_Framing          89 non-null float64
CC_Analyze          92 non-null float64
CC_Communicate      91 non-null float64
CC_Leadership       91 non-null float64
CC_Team             92 non-null float64
ADC_Overall         91 non-null float64
ADC_Avail           90 non-null float64
ADC_Faculty         91 non-null float64
ADC_Club            90 non-null float64
Acad_Longterm       83 non-null float64
Acad_Team           86 non-null float64
Acad_Responsive     84 non-null float64
Acad_Elective       79 non-null float64
Acad_Leadership     81 non-null float64
Acad_Diversity      84 non-null float64
Acad_Communicate    83 non-null float64
```

```

Acad_Ethics      84 non-null float64
Other_Speakers   82 non-null float64
Other_Orientation 82 non-null float64
Other_Interyear  76 non-null float64
Other_International 82 non-null float64
CMC_Contacts     80 non-null float64
CMC_Networking   77 non-null float64
CMC_Internshiphelp 72 non-null float64
CMC_Intership_overall 78 non-null float64
CMC_Firms        79 non-null float64
CMC_Independent  78 non-null float64
CMC_Counseling   79 non-null float64
CMC_Alumni       78 non-null float64
CMC_Library      80 non-null float64
CMC_Info         81 non-null float64
Overall_Satisfaction 81 non-null float64
Overall_NPV      80 non-null float64
Overall_Recommend 81 non-null float64
dtypes: float64(50)
memory usage: 39.8 KB

```

```

In [4]: #check stats summary
data.describe()

```

```

Out[4]:

```

	Nationality	Gender	Age	Have_offer	Teaching_Core	Teaching_Elect
count	99.000000	99.000000	99.000000	97.000000	98.000000	92.000000
mean	1.424242	1.323232	28.565657	1.556701	8.000000	8.250000
std	0.496743	0.470091	4.734075	0.499355	1.624554	1.58027
min	1.000000	1.000000	21.000000	1.000000	2.000000	3.000000
25%	1.000000	1.000000	25.000000	1.000000	7.000000	7.000000
50%	1.000000	1.000000	27.000000	2.000000	8.000000	8.000000
75%	2.000000	2.000000	30.500000	2.000000	9.000000	9.000000
max	2.000000	2.000000	43.000000	2.000000	11.000000	11.000000

8 rows × 50 columns

```

In [5]: # check missing data
data.shape[0] # number of observations

```

```

Out[5]: 100

```

```

In [6]: data.isnull().sum() / data.shape[0] # missing data rate

```

```

Out[6]: Nationality      0.01
        Gender          0.01
        Age             0.01
        Have_offer      0.03
        Teaching_Core    0.02
        Teaching_Elective 0.08
        Teaching_FacQual 0.03
        Teaching_FacKnow 0.02
        Teaching_FacAvail 0.03
        Teaching_FacAware 0.03
        Teaching_Integrated 0.03
        CC_Current      0.08
        CC_Useful        0.18
        CC_Worklevel     0.07
        CC_Interpersonal 0.07
        CC_Prepare       0.17
        CC_Framing       0.11
        CC_Analyze       0.08
        CC_Communicate   0.09
        CC_Leadership    0.09
        CC_Team          0.08
        ADC_Overall      0.09
        ADC_Avail        0.10
        ADC_Faculty      0.09
        ADC_Club         0.10
        Acad_Longterm    0.17
        Acad_Team        0.14
        Acad_Responsive  0.16
        Acad_Elective    0.21
        Acad_Leadership  0.19
        Acad_Diversity   0.16
        Acad_Communicate 0.17
        Acad_Ethics      0.16
        Other_Speakers   0.18
        Other_Orientation 0.18
        Other_Interyear  0.24
        Other_International 0.18
        CMC_Contacts     0.20
        CMC_Networking   0.23
        CMC_Internshiphelp 0.28
        CMC_Intership_overall 0.22
        CMC_Firms        0.21
        CMC_Independent  0.22
        CMC_Counseling   0.21
        CMC_Alumni       0.22
        CMC_Library      0.20
        CMC_Info         0.19
        Overall_Satisfaction 0.19
        Overall_NPV      0.20
        Overall_Recommend 0.19
        dtype: float64

```

```
In [7]: ## 3. Plot the means  
# get means of each feature  
dataMean = np.mean(data)
```

```
In [8]: # change column name into 'rmean'  
dataMean = pd.DataFrame(dataMean, columns=['rmean'])
```

```
In [9]: # drop 'Nationality', 'Gender', 'Age', 'Have_offer'  
dataMean = dataMean.drop(['Nationality', 'Gender', 'Age', 'Have_off  
er'])
```

```
In [10]: # add the category column  
dataMean['category'] = 'Teaching' # Create a column called 'catego  
ry' with all values equals 'Teaching', then adjust  
dataMean['category'][7:17] = 'cc'  
dataMean['category'][17:21] = 'ADC'  
dataMean['category'][21:29] = 'Acad'  
dataMean['category'][29:33] = 'Other'  
dataMean['category'][33:43] = 'CMC'  
dataMean['category'][43:46] = 'Overall'  
dataMean
```

```

/anaconda/lib/python2.7/site-packages/ipykernel/__main__.py:3: Sett
ingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pand
as-docs/stable/indexing.html#indexing-view-versus-copy
  app.launch_new_instance()
/anaconda/lib/python2.7/site-packages/ipykernel/__main__.py:4: Sett
ingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pand
as-docs/stable/indexing.html#indexing-view-versus-copy
/anaconda/lib/python2.7/site-packages/ipykernel/__main__.py:5: Sett
ingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pand
as-docs/stable/indexing.html#indexing-view-versus-copy
/anaconda/lib/python2.7/site-packages/ipykernel/__main__.py:6: Sett
ingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pand
as-docs/stable/indexing.html#indexing-view-versus-copy
/anaconda/lib/python2.7/site-packages/ipykernel/__main__.py:7: Sett
ingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pand
as-docs/stable/indexing.html#indexing-view-versus-copy
/anaconda/lib/python2.7/site-packages/ipykernel/__main__.py:8: Sett
ingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pand
as-docs/stable/indexing.html#indexing-view-versus-copy

```

Out[10]:

	rmean	category
Teaching_Core	8.000000	Teaching
Teaching_Elective	8.250000	Teaching
Teaching_FacQual	8.113402	Teaching
Teaching_FacKnow	9.285714	Teaching
Teaching_FacAvail	8.195876	Teaching
Teaching_FacAware	7.969072	Teaching
Teaching_Integrated	7.639175	Teaching
CC_Current	7.489130	cc

CC_Useful	7.195122	cc
CC_Worklevel	7.612903	cc
CC_Interpersonal	5.763441	cc
CC_Prepape	8.783133	cc
CC_Framing	7.651685	cc
CC_Analyze	7.619565	cc
CC_Communicate	7.307692	cc
CC_Leadership	6.901099	cc
CC_Team	8.043478	cc
ADC_Overall	7.384615	ADC
ADC_Avail	9.033333	ADC
ADC_Faculty	8.439560	ADC
ADC_Club	7.255556	ADC
Acad_Longterm	8.108434	Acad
Acad_Team	6.906977	Acad
Acad_Responsive	7.500000	Acad
Acad_Elective	7.430380	Acad
Acad_Leadership	6.975309	Acad
Acad_Diversity	6.809524	Acad
Acad_Communicate	6.722892	Acad
Acad_Ethics	6.523810	Acad
Other_Speakers	7.292683	Other
Other_Orientation	6.475610	Other
Other_Interyear	5.526316	Other
Other_International	5.378049	Other
CMC_Contacts	6.937500	CMC
CMC_Networking	6.142857	CMC
CMC_Internshiphelp	5.416667	CMC
CMC_Intership_overall	7.076923	CMC
CMC_Firms	6.379747	CMC
CMC_Independent	7.333333	CMC

CMC_Counseling	7.645570	CMC
CMC_Alumni	7.423077	CMC
CMC_Library	7.412500	CMC
CMC_Info	7.308642	CMC
Overall_Satisfaction	7.827160	Overall
Overall_NPV	7.762500	Overall
Overall_Recommend	7.876543	Overall

```
In [11]: # convert type of 'category'
dataMean['category'] = dataMean['category'].astype('category')
dataMean.dtypes
```

```
Out[11]: rmean          float64
category      category
dtype: object
```

```
In [12]: # creat color dictionary
printColor = {
    'Teaching': 'g',
    'cc': 'navy',
    'ADC': 'lightgreen',
    'Acad': 'lightblue',
    'Other': 'darkred',
    'CMC': 'lightcoral',
    'Overall': 'k'
}
printColor
```

```
Out[12]: {'ADC': 'lightgreen',
'Acad': 'lightblue',
'CMC': 'lightcoral',
'Other': 'darkred',
'Overall': 'k',
'Teaching': 'g',
'cc': 'navy'}
```

```
In [13]: dataMean['color'] = 'placeholder'
for cat in dataMean.category.unique():
    dataMean.ix[dataMean.category == cat, 'color'] = printColor[cat]
dataMean
```

```
Out[13]:
```

	rmean	category	color
Teaching_Core	8.000000	Teaching	g
Teaching_Elective	8.250000	Teaching	g
Teaching_FacQual	8.113402	Teaching	g

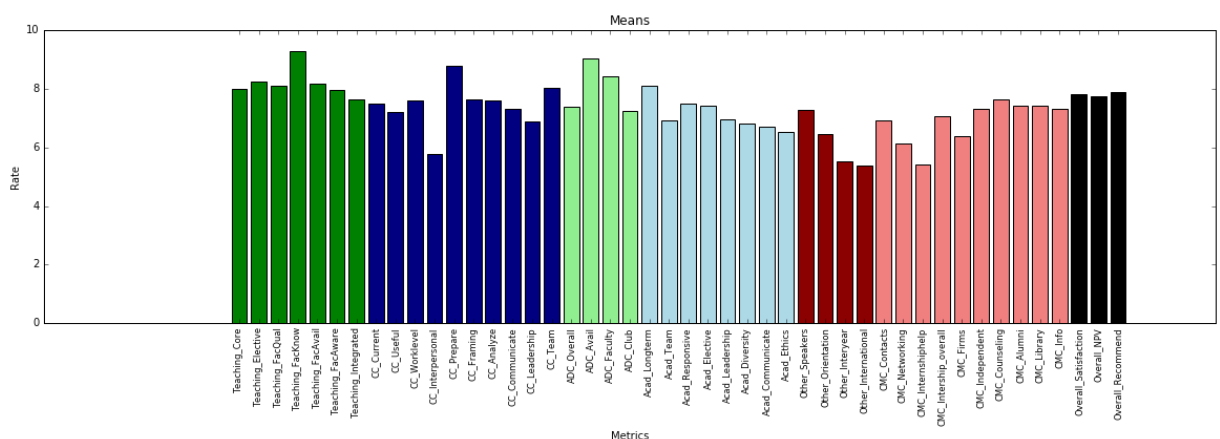
Teaching_FacKnow	9.285714	Teaching	g
Teaching_FacAvail	8.195876	Teaching	g
Teaching_FacAware	7.969072	Teaching	g
Teaching_Integrated	7.639175	Teaching	g
CC_Current	7.489130	cc	navy
CC_Useful	7.195122	cc	navy
CC_Worklevel	7.612903	cc	navy
CC_Interpersonal	5.763441	cc	navy
CC_Prepare	8.783133	cc	navy
CC_Framing	7.651685	cc	navy
CC_Analyze	7.619565	cc	navy
CC_Communicate	7.307692	cc	navy
CC_Leadership	6.901099	cc	navy
CC_Team	8.043478	cc	navy
ADC_Overall	7.384615	ADC	lightgreen
ADC_Avail	9.033333	ADC	lightgreen
ADC_Faculty	8.439560	ADC	lightgreen
ADC_Club	7.255556	ADC	lightgreen
Acad_Longterm	8.108434	Acad	lightblue
Acad_Team	6.906977	Acad	lightblue
Acad_Responsive	7.500000	Acad	lightblue
Acad_Elective	7.430380	Acad	lightblue
Acad_Leadership	6.975309	Acad	lightblue
Acad_Diversity	6.809524	Acad	lightblue
Acad_Communicate	6.722892	Acad	lightblue
Acad_Ethics	6.523810	Acad	lightblue
Other_Speakers	7.292683	Other	darkred
Other_Orientation	6.475610	Other	darkred
Other_Interyear	5.526316	Other	darkred
Other_International	5.378049	Other	darkred
CMC_Contacts	6.937500	CMC	lightcoral

CMC_Networking	6.142857	CMC	lightcoral
CMC_Internshiphelp	5.416667	CMC	lightcoral
CMC_Intership_overall	7.076923	CMC	lightcoral
CMC_Firms	6.379747	CMC	lightcoral
CMC_Independent	7.333333	CMC	lightcoral
CMC_Counseling	7.645570	CMC	lightcoral
CMC_Alumni	7.423077	CMC	lightcoral
CMC_Library	7.412500	CMC	lightcoral
CMC_Info	7.308642	CMC	lightcoral
Overall_Satisfaction	7.827160	Overall	k
Overall_NPV	7.762500	Overall	k
Overall_Recommend	7.876543	Overall	k

```
In [14]: # plot
plt.figure(figsize=(20, 5))

y = list(dataMean.rmean)
x = np.arange(len(y))
xlabel = list(dataMean.index)
plt.bar(x,y,0.8, align='center', color = dataMean.color)
plt.xticks(x, xlabel, rotation=90, fontsize='small')
plt.title("Means")
plt.xlabel("Metrics")
plt.ylabel("Rate")

plt.show()
```



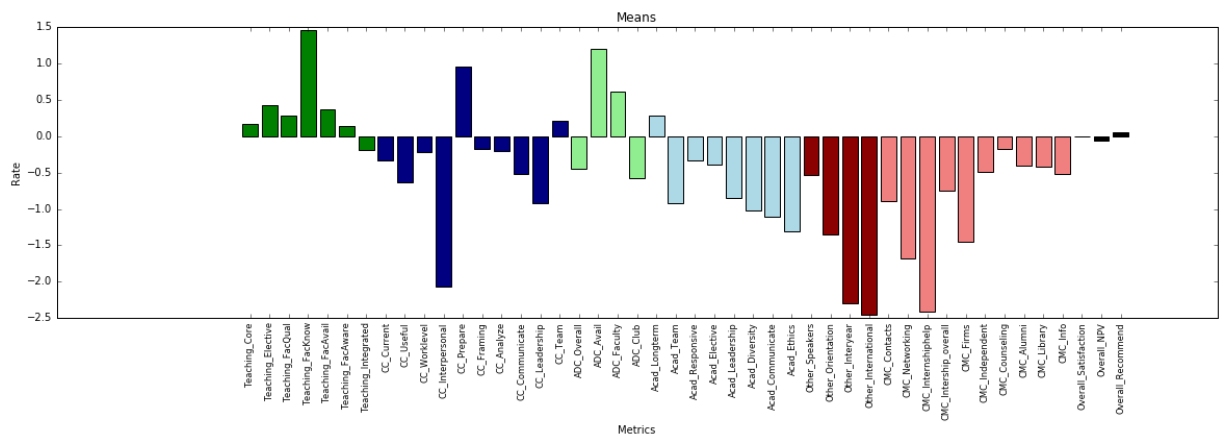
```
In [15]: ## 4. Plot differences in means from the mean overall satisfaction.
# calculation
mean_differences = dataMean.rmean - dataMean.rmean['Overall_Satisfaction']
mean_differences
```

```
Out[15]: Teaching_Core          0.172840
Teaching_Elective          0.422840
Teaching_FacQual           0.286242
Teaching_FacKnow           1.458554
Teaching_FacAvail          0.368716
Teaching_FacAware          0.141912
Teaching_Integrated        -0.187985
CC_Current                 -0.338030
CC_Useful                  -0.632039
CC_Worklevel               -0.214257
CC_Interpersonal           -2.063720
CC_Prepave                 0.955972
CC_Framing                 -0.175475
CC_Analyze                 -0.207595
CC_Communicate             -0.519468
CC_Leadership              -0.926062
CC_Team                   0.216318
ADC_Overall                -0.442545
ADC_Avail                  1.206173
ADC_Faculty                0.612400
ADC_Club                   -0.571605
Acad_Longterm              0.281273
Acad_Team                  -0.920184
Acad_Responsive            -0.327160
Acad_Elective              -0.396781
Acad_Leadership            -0.851852
Acad_Diversity             -1.017637
Acad_Communicate           -1.104269
Acad_Ethics                -1.303351
Other_Speakers             -0.534478
Other_Orientation          -1.351551
Other_Interyear            -2.300845
Other_International        -2.449112
CMC_Contacts               -0.889660
CMC_Networking             -1.684303
CMC_Internshiphelp         -2.410494
CMC_Intership_overall      -0.750237
CMC_Firms                  -1.447414
CMC_Independent            -0.493827
CMC_Counseling             -0.181591
CMC_Alumni                 -0.404084
CMC_Library                -0.414660
CMC_Info                   -0.518519
Overall_Satisfaction        0.000000
Overall_NPV                -0.064660
Overall_Recommend          0.049383
Name: rmean, dtype: float64
```

```
In [16]: # plot
plt.figure(figsize=(20, 5))

y = list(mean_differences)
x = np.arange(len(y))
xlabel = list(dataMean.index)
plt.bar(x,y,0.8, align='center', color = dataMean.color)
plt.xticks(x, xlabel, rotation=90, fontsize='small')
plt.title("Means")
plt.xlabel("Metrics")
plt.ylabel("Rate")

plt.show()
```



```
In [17]: ## 5. Plot correlation
# calculation
del data['Nationality']
del data['Gender']
del data['Age']
dataCorr = data.corr()
dataCorr
```

Out[17]:

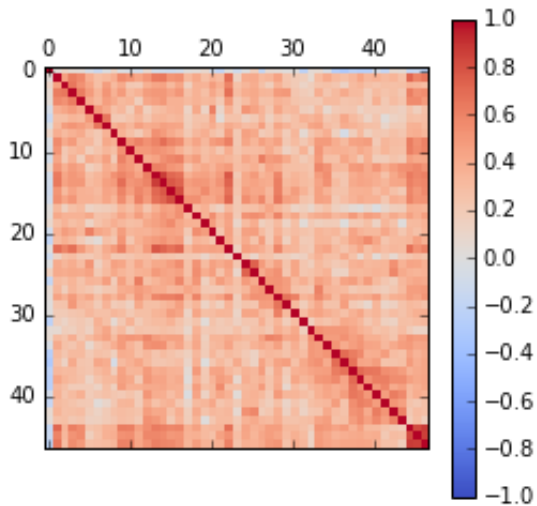
	Have_offer	Teaching_Core	Teaching_Elective	Teaching_FacQual
Have_offer	1.000000	0.039303	0.122248	0.054889
Teaching_Core	0.039303	1.000000	0.467311	0.628315
Teaching_Elective	0.122248	0.467311	1.000000	0.652265
Teaching_FacQual	0.054889	0.628315	0.652265	1.000000
Teaching_FacKnow	0.042691	0.574099	0.534709	0.630655
Teaching_FacAvail	-0.057710	0.384968	0.437430	0.487152
Teaching_FacAware	-0.133702	0.284815	0.283371	0.349658
Teaching_Integrated	-0.020906	0.324632	0.478376	0.521673
CC_Current	0.122591	0.386595	0.320000	0.374205

CC_Useful	-0.023172	0.564842	0.413705	0.491797
CC_Worklevel	-0.041807	0.517878	0.220836	0.310270
CC_Interpersonal	-0.068923	0.431355	0.204909	0.363438
CC_Prepere	0.075023	0.511879	0.380326	0.481185
CC_Framing	-0.067640	0.634187	0.433761	0.500860
CC_Analyze	-0.091108	0.647505	0.446519	0.503245
CC_Communicate	-0.105807	0.522393	0.321413	0.440689
CC_Leadership	-0.077271	0.545942	0.437225	0.515506
CC_Team	0.022728	0.324989	0.157742	0.278466
ADC_Overall	0.038517	0.416463	0.366610	0.365357
ADC_Avail	0.087755	0.282140	0.359190	0.288087
ADC_Faculty	-0.091821	0.365433	0.479452	0.427241
ADC_Club	-0.183487	0.308041	0.376415	0.322222
Acad_Longterm	0.004282	0.693535	0.593879	0.685918
Acad_Team	-0.106475	0.147184	0.268007	0.299129
Acad_Responsive	-0.004957	0.403369	0.342417	0.392115
Acad_Elective	0.159402	0.406722	0.396492	0.415613
Acad_Leadership	-0.178214	0.309467	0.278801	0.350691
Acad_Diversity	0.056145	0.340271	0.179351	0.338988
Acad_Communicate	0.046585	0.449457	0.361230	0.485337
Acad_Ethics	0.008548	0.249808	0.253341	0.506361
Other_Speakers	0.008217	0.200191	0.202342	0.235540
Other_Orientation	-0.221343	0.178028	0.124580	0.194368
Other_Interyear	0.050478	0.067439	0.175132	0.163637
Other_International	0.066783	0.305892	0.256174	0.383803
CMC_Contacts	-0.040581	0.412729	0.266404	0.239334
CMC_Networking	-0.261408	0.317525	0.236177	0.314917
CMC_Internshiphelp	-0.310977	0.280376	0.136098	0.408517
CMC_Intership_overall	-0.219044	0.329897	0.309194	0.314616
CMC_Firms	-0.247417	0.436672	0.345721	0.309422
CMC_Independent	-0.244094	0.331632	0.273542	0.191055

CMC_Counseling	-0.075731	0.361933	0.242549	0.310035
CMC_Alumni	-0.164211	0.314100	0.284429	0.256703
CMC_Library	0.065373	0.349949	0.279139	0.283140
CMC_Info	0.044528	0.273392	0.288621	0.221262
Overall_Satisfaction	-0.160207	0.723505	0.356173	0.621097
Overall_NPV	-0.143437	0.597732	0.330145	0.524172
Overall_Recommend	-0.205359	0.652103	0.300405	0.558554

47 rows × 47 columns

```
In [18]: # plot
plt.matshow(dataCorr, vmin=-1, vmax=1, cmap = 'coolwarm')
plt.colorbar()
plt.show()
```



In []: