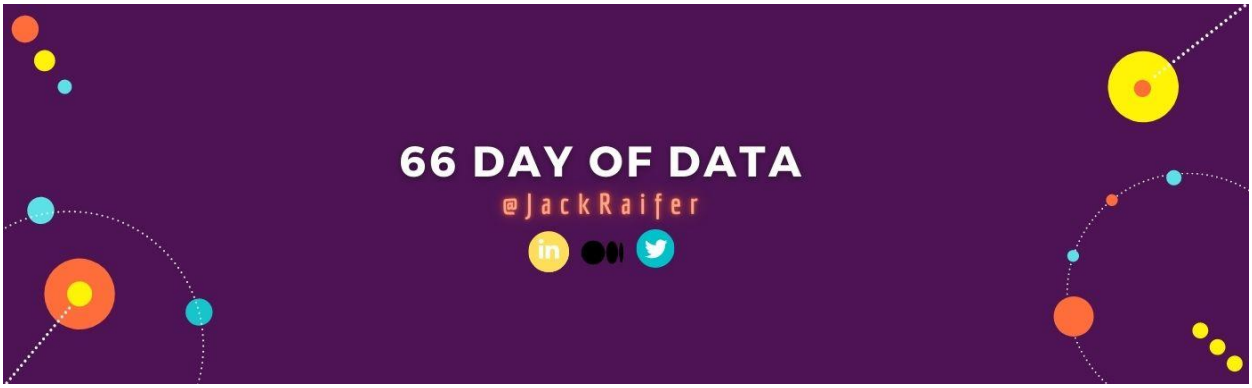


#66DaysOfData – Day 8: Cracking open some more data

A couple of days back I hit a wall, and to overcome it, I went on the lookout for even more data. Today, we crack open some of it, in particular, I will be checking out these 2 datasets:

Basic HR Data Set, also on Kaggle

IBM HR Analytics Employee Attrition & Performance on Kaggle

Once again, we set up the notebook with all the libraries I like for EDA:

```
[1]: import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
     %matplotlib inline
```

Here is what the basic dataframe for the basic HR Dataset looks like:

HR Dataset

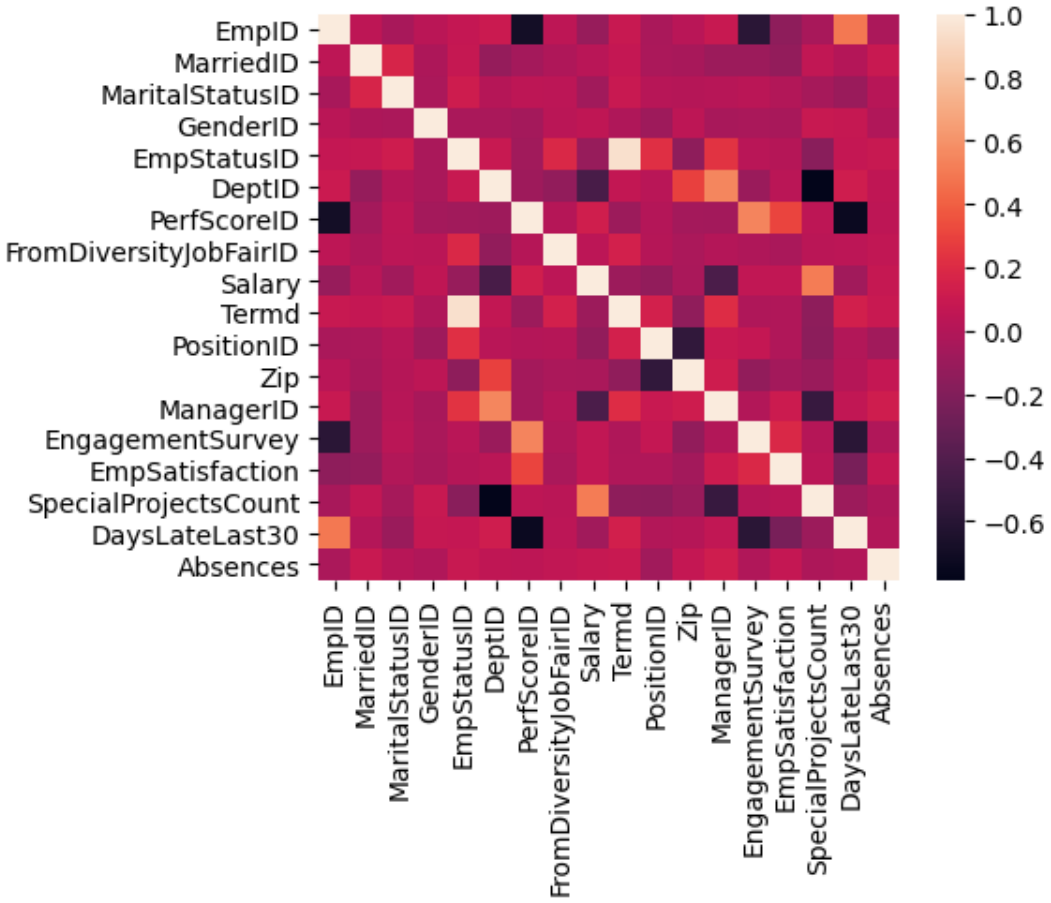
```
[2]: #We begin with the HR Dataset
     df1 = pd.read_csv('DATA_ATTRITION_AND_HR/HRDataset_v14.csv')

[3]: df1.head()
```

[3]:

ID	GenderID	EmpStatusID	DeptID	PerfScoreID	FromDiversityJobFairID	Salary	...	ManagerName	ManagerID	RecruitmentSource
0	1	1	5	4	0	62506	...	Michael Albert	22.0	LinkedIn
1	1	5	3	3	0	104437	...	Simon Roup	4.0	Indeed
1	0	5	5	3	0	64955	...	Kissy Sullivan	20.0	LinkedIn
1	0	1	5	3	0	64991	...	Elijiah Gray	16.0	Indeed
2	0	5	5	3	0	50825	...	Webster Butler	39.0	Google Search

There is some interesting information here, so let us see if we can find some correlations that we could use in the future:

```
[84]: fig=plt.figure(figsize=(5,4), dpi= 100, facecolor='w', edgecolor='k')
      sns.heatmap(df1.corr())
```



Here are the most interesting correlations I found on this dataset:

- A negative correlation between lateness the last 30 days and the engagement survey score. Meaning that the more engaged people are at work, the less likely they are to be late for work.
- Another negative correlation between performance score and lateness, the worse a person performs, the more common it is for them to be late.
- A positive correlation between performance score and engagement, which is expected.
- Finally, and interestingly, salary and special projects count have positive correlation, meaning the higher your salary the more extra projects people take on.

Even though there is a lot of interesting data on here and there is a lot we could do with it, we are missing an integral part for what we want at this moment, attrition data. So let us see if we can find some on the IBM dataset:

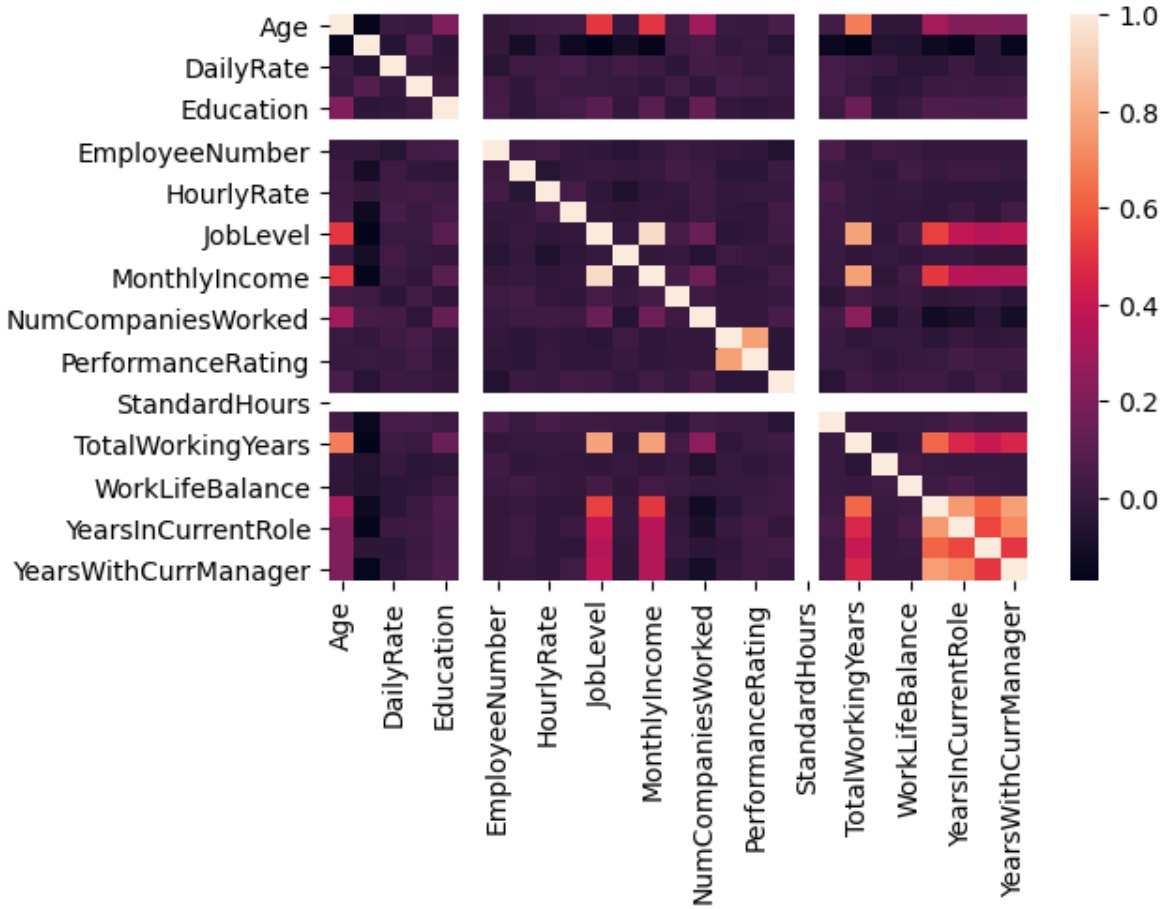
IBM HR Dataset

```
df2 = pd.read_csv('DATA_ATTRITION_AND_HR/IBM-HR-Employee-Attrition.csv')
```

```
df2.head()
```

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7

And let us dive right away into the correlations:



Here we can see few interesting information, but we are missing one VERY important feature: attrition. What happened? Simple, careless me forgot that in this dataset attrition is recorded as string variable with “Yes” or “No” options. So, we need to transform those into numbers, since there are only 2 options, a binary 1 for yes and 0 for no will work:

```
58]: # Changing yes / no in attrition into a 1, 0 variable
     df2['Attrition'].replace(('Yes', 'No'), (1, 0), inplace=True)
```

```
59]: df2.head()
```

59]:

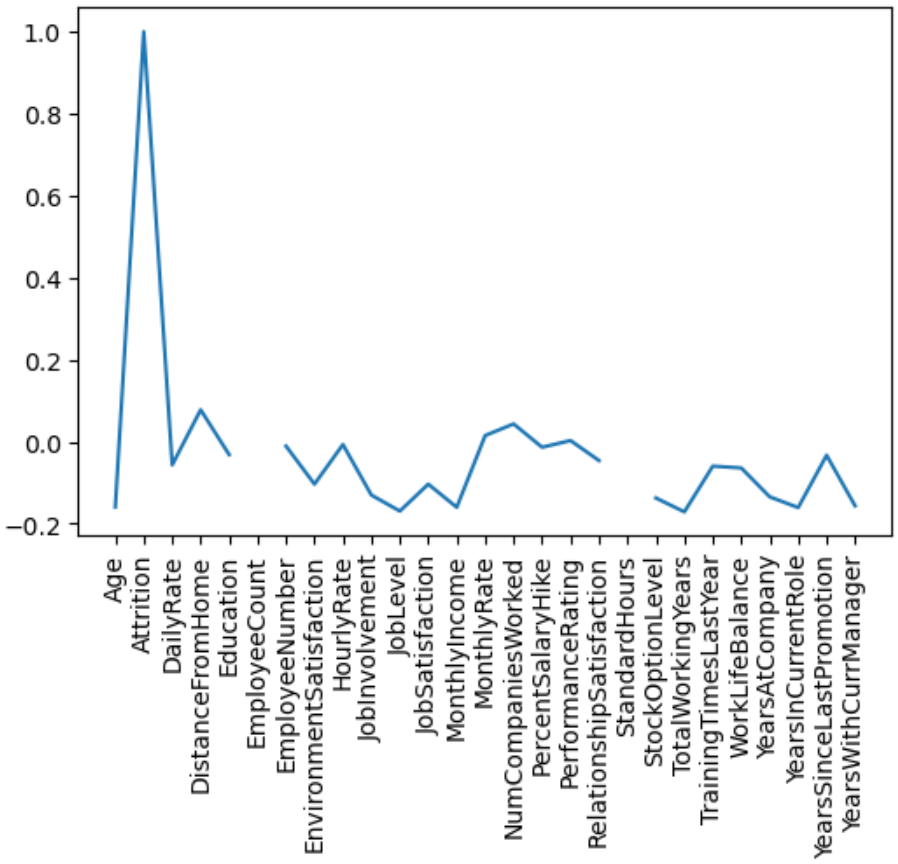
	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome
0	41	1	Travel_Rarely	1102	Sales	1
1	49	0	Travel_Frequently	279	Research & Development	8
2	37	1	Travel_Rarely	1373	Research & Development	2
3	33	0	Travel_Frequently	1392	Research & Development	3
4	27	0	Travel_Rarely	591	Research & Development	2

And, since we already ran a heatmap on everything else, let us this time just run the numbers and a simple line plot on the attrition column:

```
[73]: df2.corrwith(df2['Attrition'], axis=0, drop=False, method='pearson')
```

```
[73]: Age                       -0.159205
      Attrition                  1.000000
      DailyRate                 -0.056652
      DistanceFromHome           0.077924
      Education                 -0.031373
      EmployeeCount                   NaN
      EmployeeNumber            -0.010577
      EnvironmentSatisfaction   -0.103369
      HourlyRate                -0.006846
      JobInvolvement            -0.130016
      JobLevel                  -0.169105
      JobSatisfaction           -0.103481
      MonthlyIncome             -0.159840
      MonthlyRate                0.015170
      NumCompaniesWorked         0.043494
      PercentSalaryHike         -0.013478
      PerformanceRating          0.002889
      RelationshipSatisfaction  -0.045872
      StandardHours                   NaN
      StockOptionLevel          -0.137145
      TotalWorkingYears         -0.171063
      TrainingTimesLastYear     -0.059478
      WorkLifeBalance           -0.063939
      YearsAtCompany            -0.134392
      YearsInCurrentRole        -0.160545
      YearsSinceLastPromotion   -0.033019
      YearsWithCurrManager      -0.156199
      dtype: float64
```

```
[82]: attrition_corr = df2.corrwith(df2['Attrition'], axis=0, drop=False, method='pearson')
      fig3=plt.figure(figsize=(7,5), dpi= 100, facecolor='w', edgecolor='k')
      plt.xticks(rotation=90)
      plt.plot(attrition_corr)
```



Looking at this plot we get a sense that attrition is not very highly correlated with anything else on this dataset, the biggest one we have is a negative correlation between attrition and TotalWorkingYears -0.171063, which is not incredibly exciting.

This does not mean we cannot build a predictive model, we can, the only issues are that when there are not clear correlations, simple and easy to explain models will probably not work. Also, we still do not have any actual data that combines attrition and the OCEAN personality model.

Well, we still have one dataset to check out and some important decisions to make, but that is a problem for tomorrow me.

You can check out all the datasets, notebooks and more on my GitHub Repository here.

Next Time – One more dataset and making decisions to move forward.

Jack Raifer Baruch

Follow me on Twitter: @JackRaifer

Follow me on LinkedIN: jackraifer

About the Road to Data Science - #66DaysOfData Series

Road to Data Science series began after I experienced the first round of Ken Jee´s #66DaysOfData challenge back in 2020. Since we are starting the second round of the challenge, I thought it would be a good idea to add small articles every day where I can comment my progress.

I will be sharing all the notebooks, articles and data I can on GitHub: https://github.com/jackraifer/66DaysOfData-Road-to-Data-Science

Please do understand I might have to withhold some information, including code, data, visualizations and/or models, because of confidentiality regards. But I will try to share as much as possible.

Want to follow the #66DaysOfDataChallenge?

Just follow Ken Jee on twitter **@KenJee_DS** and join the #66DaysOfData challenge.

You can also reach out to me at any time through **LinkedIN** or **Twitter.**