The Capstone 1 project

Prediction of the core financial ratios of the Austrian banking system

Project Proposal

Aim

The aim is to build a basic prediction model for 4 main factors in the banking system - capital, asset quality, liquidity and profitability.

It would enable the user to be ahead of the trend and can adjust its lending policies in advance

For this exercise I want to test it on Austria. For comparison purposes I also chose Germany, Italy, Spain and Netherlands.

Client

The client could be a risk management department of any bank or insurance company that wants to predict the future levels of capital and bad loans in the country. This would enable them to have a forward-looking approach and to adjust its risk approach much quickier and thus gain a competitive advantage.

Data

The macro-economic data are freely available from international organisations like IMF or Eurostat. The banking system data is available at the Central bank datasets.

Deliverable

A code that is scalable to include different countries.

Data Wrangling

The aim of the exercise was to download and wrangle the data needed for the Capstone project.
The data was downloaded through API from the IMF database and consisted of quarterly and annual Non-performing loans values of 5 Eurozone countries.
The steps to clean the data were as follows

1. Transforming nested JSON object into a dataframe
2. EDA showed missing data on the quarterly level and completely missing data for one country - DE, no index and shifted time to beginning of the period instead of the end

For the yearly data:

3. First step is to rename date column to 'Date'
4. Changing string to Datetime
5. Setting it as an index
6. Shifting the index by 1 month forward

7. Resampling yearly data to quarterly and filling the missing data with 0 so i could join the DE missing column to the quarterly
8. Extracting the DE column

For the quarterly data:

9. Repeating steps 3-7
10. Joining the DE column
11. Replacing the 0 resampled values for DE for NaN
12. Resampling the NaN values for all columns with linear function
13. Dropping the NaN values for the starting period
14. EDA analysis for the tidy data

For the EDA i used Seaborn therefore:

15. Melting the datasets so the Seaborn can create a line graph

This needs to be repeated for all the data sets

Statistical inference

Are there variables that are particularly significant in terms of explaining the answer to your project question?

All the dependent variables are significant as they are the core financial ratios for the various banking systems. In the next chapter will investigate the significance of the independent variables in the linear regression.

Are there significant differences between subgroups in your data that may be relevant to your project aim?

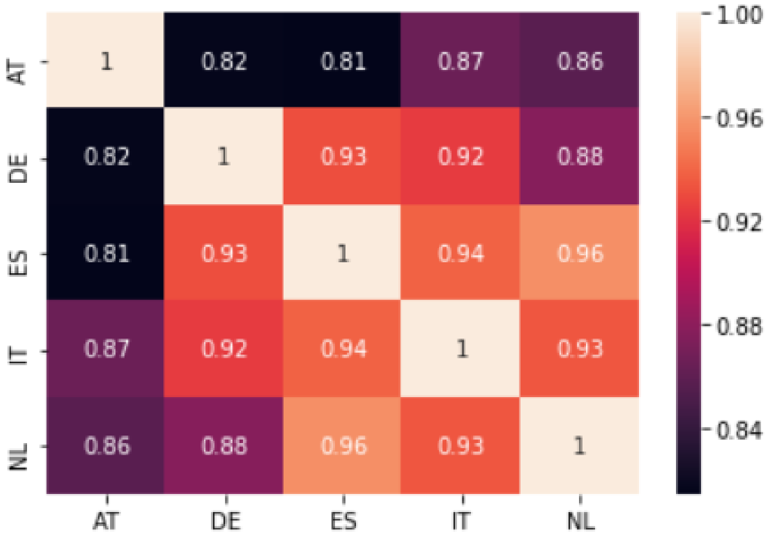
In the EDA analysis it was obvious that there are differentiations between various countries. Please refer to the presentation “European Banks Core Financials”. There is a trend of certain countries that are consistently outperforming and underperforming in different categories. Having said that the volatility is more pronounced in the profitability category.

Are there strong correlations between pairs of independent variables or between an independent and a dependent variable?

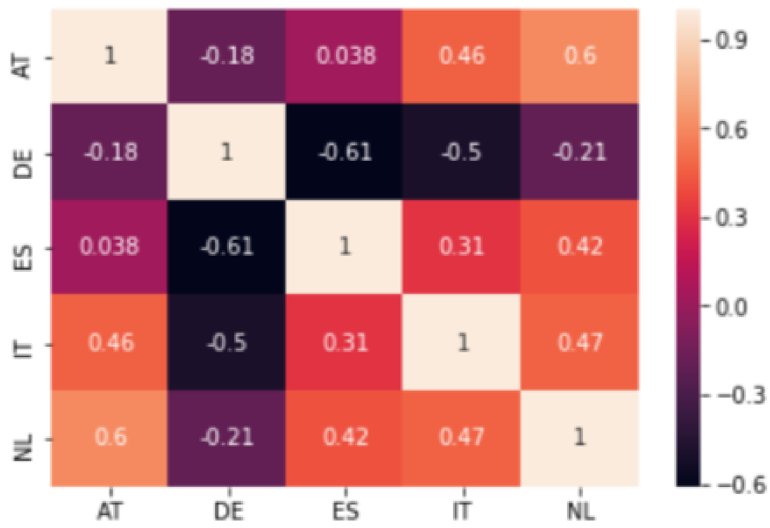
There are strong correlations mainly in the capitalisation ratios. This might be due to a fact , that this ratios are regulated by the European central bank, unlike other variables (profitability)

CAR

T1



Return on Assets



<u>What are the most appropriate tests to use to analyze these relationships?</u>

The most appropriate test would be a linear regression.

<u>**Regression**</u>

For regression i used:

Linear regression

Lasso regression using statsmodels

Lasso regression using Scikit - Learn

The data table

	INFL	DEP	LOA	UNE	GDP	NUM	CAR	T1	NPL	NIM	ROA	ROE	LA	LATA
count	40.000000	40.000000	40.000000	40.00000	40.000000	40.000000	40.000000	40.000000	40.000000	40.000000	40.000000	40.00000	40.00000	40.00000
mean	109.751638	28.675000	27.900000	5.22550	106.622698	42.587500	16.966000	13.453250	2.756000	61.102000	0.276000	4.13100	70.43450	24.74050
std	5.494802	0.729858	3.287895	0.46138	4.620427	3.349584	1.034933	1.418514	0.625836	2.928023	0.223639	3.20757	3.39787	1.07947
min	99.048490	28.000000	24.000000	4.51000	98.140928	38.000000	15.300000	11.360000	1.630000	56.210000	-0.200000	-3.16000	63.93000	22.53000
25%	105.970648	28.000000	25.000000	4.80250	103.492810	39.875000	16.147500	12.332500	2.400000	59.057500	0.127500	1.96000	67.90250	24.15750
50%	110.113449	29.000000	27.000000	5.19500	104.708923	42.000000	16.685000	13.165000	2.725000	60.680000	0.275000	4.31000	69.68000	24.64500
75%	113.907504	29.000000	31.000000	5.65250	110.395567	45.312500	17.990000	14.812500	3.117500	62.427500	0.432500	6.93000	73.01000	25.32500
max	119.054094	30.000000	33.000000	6.03000	115.073540	50.000000	18.840000	15.930000	4.100000	68.360000	0.770000	9.98000	77.00000	27.41000

Linear regression

The overall results were weak. The only meaningful adjusted R was for NPL and T1

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    NPL   R-squared:                       0.860
Model:                            OLS   Adj. R-squared:                  0.834
Method:                 Least Squares   F-statistic:                     33.66
Date:                Sun, 14 Jun 2020   Prob (F-statistic):           1.05e-12
Time:                        14:22:39   Log-Likelihood:                 1.7516
No. Observations:                  40   AIC:                             10.50
Df Residuals:                      33   BIC:                             22.32
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                     T1   R-squared:                       0.910
Model:                            OLS   Adj. R-squared:                  0.894
Method:                 Least Squares   F-statistic:                     55.85
Date:                Wed, 10 Jun 2020   Prob (F-statistic):           7.06e-16
Time:                        15:32:48   Log-Likelihood:                -21.997
No. Observations:                  40   AIC:                             57.99
Df Residuals:                      33   BIC:                             69.82
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
```

The statsmodel Lasso regression did not show any fit metrics, the results are however:

NPL
```
Parameters:  Intercept     0.000000
UNE          0.000000
GDP          0.016392
INFL         0.002163
DEP          0.000000
LOA          0.011115
NUM          0.010612
dtype: float64
```

CAR
```
Parameters:  Intercept     0.000000
UNE          0.797361
GDP          0.120488
INFL         0.000000
DEP          0.000000
LOA         -0.000945
NUM          0.000000
dtype: float64
```

T1
```
Parameters:  Intercept     0.000000
UNE          0.499931
GDP          0.126073
INFL         0.000000
DEP         -0.024334
LOA         -0.068296
NUM          0.000000
dtype: float64
```

NIM
```
Parameters:  Intercept     4.524967
UNE          6.066688
GDP          0.151470
INFL        -0.004990
DEP          0.005705
LOA          0.332400
NUM          0.000000
dtype: float64
```

ROA
```
Parameters:  Intercept     0.000000
UNE          0.000000
GDP          0.002543
INFL         0.000000
DEP          0.000000
LOA          0.000000
NUM          0.000000
dtype: float64
```

LA
```
Parameters:  Intercept     10.736264
UNE          5.935717
GDP          0.136117
INFL         0.000000
DEP          0.061476
LOA          0.526426
NUM         -0.049144
dtype: float64
```

ROE
```
Parameters:  Intercept     0.000000
UNE          0.000000
GDP          0.064190
INFL         0.000000
DEP         -0.035042
LOA         -0.061389
NUM          0.000000
dtype: float64
```

LATA
```
Parameters:  Intercept     0.000000
UNE          1.547753
GDP          0.119395
INFL         0.000000
DEP          0.000000
LOA          0.142126
NUM          0.000000
dtype: float64
```

The scikit-learn

The Lasso regression through scikit learn was not very successful as the highest training score was 0.88 and test score 0.83 for T1.