Financial Econometrics

Machine Learning application to scoring procedure

Dakoli Isei
Dantoni Dario
Sallin Gabriel
Von Der Weid Olivier

Introduction

Our project is about scoring procedure. From a sample of funds in the durability theme, we want to predict the rating of new funds having the same factors. We will compare different predictive methods and different factors to assess which ones are the best. We will use the concepts seen in the course A. XII. and code it in Python.

Data (120 observations/funds)

Y dependent variable :

- Rate (grade over 5 of the fund given by Morningstar)

X's independent variables:

- Sharp ratio of the fund (3 years)
- Beta of the fund (3 years)
- Alpha of the fund (3 years)
- Durability score (over 50) of the fund
- Year (year of creation of the fund)
- Yearly management fees of the fund in %
- Day change of the fund in %
- Currency of the fund
- Country (domicile of the fund)

Method

We start with the platform Morningstar[1] that gather a list of sustainable investment funds produced by different banks in different countries. This website attributes a grade expressed in stars, from 0 to 5, according to a set of characteristics available for each fund. For each fund, we have access to graphics, risks, performance, durability, fees and many other metrics. We have chosen two kinds of metrics. First one : the ones we think are **the most important** to assess the quality of a fund : **sharp ratio, alpha, beta, durability score, year of creation.** Ratio ratio, alpha, beta describe the trade-off

[1] https://www.morningstar.fr/fr/funds/esg.aspx

between performance and risk of the fund, durability score is essential because the funds are listed around this theme and year of creation is an indicator of the lifespan of the fund.
Second ones are seen as **less important metrics** : **currency, day change, country, yearly management fees**. Currency and country should not impact any kind of performance, day change is a random percentage that changes every day and depends on the date we record it, yearly management fees are fixed by many more criteria than just the performance of the fund.

We have a limited number of metrics because we only have 120 different funds. Prediction needs a lot of data to increase correctness. The more factors we add, the more data we should have. We gathered 120 different individuals. It may not be enough to create a good model. However, we gathered manually everything in an excel sheet, which took already a lot of time. Moreover, we prefer not mixing data sources because the ratings procedure may differ among financial analyst companies. Finally we import the data on Python.

Before applying prediction methods, we need to clean the data. It implies treating the missing values and replacing them by the mean of the variable. We explore with graphs such as histograms or boxplots the repartition of each variable, to detect abnormal values like outliers. We have to create dummy variables for the categorical variables. It concerns Rate, Currency and Country. As we carefully recorded ourselves the data, the cleaning procedure is easy and quickly done.
The last step is to divide our dataset in two : one training set which will have 70% of the data, and the test set with the 30% remaining. Training set is used to create the algorithm. In our case, the β coefficients of the regression are estimated with the train set. Then we need to test the performance of the algorithm on the test set to have a correct view of the prediction power of our model on new data.

We are now ready to apply a prediction method. First, we will try naively a linear regression. Thus, the output rate is considered here as a numerical variable instead of a categorical. We use the coefficients estimated to predict the test set, and round the result to the closest integer.
Second method is more appropriate : the multinomial Logistic model, as stated in the course A.XII. We will run three different models. One with the “good” independent variables, another one with the “bads”, and the last one with all variables.

The metric used to assess the predicting power is the accuracy, which is the number of correct predictions within all categories divided by the total of predictions. We choose to add another metric: as we have 6 classes to predict (0 to 5) and not a classic 0-1, we want to weigh the error we make. For example, predicting a rate of 4 instead of a real rate of 5 is not as problematic as predicting a rate of 1 instead of 5. The accuracy does not give this information, so we compute this error as the standard deviation of the prediction (see equation (1)). This explains how far we are from the exact value on average.

$$Average\ error\ = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\widehat{y}_i)^2}{N}}\ = Standard\ deviation\ error \quad (1)$$

Models[2]

*Good predictors (**Model 1** on Python)*

$$rate_i = \beta_0 + \beta_1 x_{Sharpe\ ratio} + \beta_2 x_{Beta} + \beta_3 x_{Alpha} + \beta_4 x_{Durability\ Score} + \beta_5 x_{Year} + \varepsilon_i$$

*Bad predictors (**Model 2** on Python)*

$$rate_i = \beta_0 + \beta_1 x_{Yearly\ mana.\ fees} + \beta_2 x_{Day\ change} + \beta_3 x_{Currency_{EUR}} + \beta_4 x_{Currency_{GBP}} + \beta_5 x_{Currency_{JPY}} + \beta_6 x_{Currency_{SEK}} + \beta_7 x_{Currency_{SGD}} + \beta_8 x_{Currency_{USD}} + \beta_9 x_{Country_{Irlande}} + \beta_{10} x_{Country_{Luxembourg}} \beta_{11} x_{Country_{Norway}} + \beta_{12} x_{Country_{UK}} + \beta_{13} x_{Country_{Spain}} + \varepsilon_i$$

*Combination of both (**Model 3** on Python)*

$$rate_i = \beta_0 + \beta_1 x_{Sharpe\ ratio} + \beta_2 x_{Beta} + \beta_3 x_{Alpha} + \beta_4 x_{Durability\ Score} + \beta_5 x_{Year} + \beta_6 x_{Yearly\ mana.\ fees} + \beta_7 x_{Day\ change} + \beta_8 x_{Currency_{EUR}} + \beta_9 x_{Currency_{GBP}} + \beta_{10} x_{Currency_{JPY}} + \beta_{11} x_{Currency_{SEK}} + \beta_{12} x_{Currency_{SGD}} + \beta_{13} x_{Currency_{USD}} + \beta_{14} x_{Country_{Irlande}} + \beta_{15} x_{Country_{Luxembourg}} + \beta_{16} x_{Country_{Norway}} + \beta_{17} x_{Country_{UK}} + \beta_{18} x_{Country_{Spain}} + \varepsilon_i$$

Results

	Models					
	Good predictors		Weak predictors		All predictors	
	Accuracy	**Error**	**Accuracy**	**Error**	**Accuracy**	**Error**
Linear regression	0.06	3.48	0.33	1.62	0.42	1.63
Logistic Regression	0.44	1.39	0.22	1.67	0.36	1.50

From this table[3], we can see that the best model is the Logistic one with only the set of good predictors. It performs an accuracy of **44%** and it is wrong about **1.39** units on average.
As we have planned, the Logistic regression (with the "good" metrics) is the best since it is more appropriate to that kind of situation (multiclass classification). However, we can see that the more we add variables, better become the Linear model so we should pay attention to this and maybe think about other models feeded with richer inputs and still compare between both.

[2] Here we just decided to show the equation for the linear model regression to have a visual representation of the model. The logistic one is quite similar with some difference in the dependent variable side.
[3] The details of the prediction results are stored in the confusion matrix in the index.

Improvement

Regarding our tiny dataset, our quality of prediction is not so bad. Obviously, collecting more data in order to train better our algorithm could help getting a much better prediction power. Another alternative for improvement is to feed our model with richer and more valuable inputs/features. Indeed, the predictors we have chosen may not be the best for scoring rate prediction. Furthermore, we could play with the variables we already have. Here we have just modelised simple linear dependence between the rate and its features. However we could modelised quadratic or logarithm relationships between both in order to complexify the model and capture more patterns. Finally, we could try other machine learning algorithms known to have a better prediction. Since it is outside the scope of this course, we are restricted to the ones seen in class.

Conclusion

By constraint of time and knowledge in this area, we have just figured out how complex it is to model the score of a fund. We have been able to distinguish “good” from “bad” metrics but on average we are making wrong about more than 1 unit from the exact value. Our model can give a general idea of the score rating of a new fund, but should not be used as a definitive score. A mix between using the model and having professional advice may give the best representation of how worthy a fund is.

Index