IPCA with RKHS Ridge Regression

Belvisi Andrea, D'Andrea Michelangelo, Ferrazzi Matteo
Machine Learning (CS-433) Project - Fall 2023
EPFL, Switzerland

***Abstract*—In this Machine Learning course project at EPFL we applied machine learning techniques to study factor models. We explore the IPCA method proposed by [1], which introduces a linear relationship between characteristics and returns, and we augment the analysis by introducing Kernel Ridge Regression method. The introduction of non-linearity yields the best results, and the choice of a Gaussian Kernel proves to be the optimal one**

I. Introduction

Our work revolves around the concept of factor model, one of the key tools for performing asset pricing. In a factor model the asset returns are assumed to be explained by a group of risk sources called factors by a linear dependence. The general setting of a static model is

$$r_{i,t} = \alpha_i + \boldsymbol{\beta}_i' \mathbf{f}_t + \epsilon_{i,t}$$
$$\forall t \in \{1, .., T\}, \forall i \in \{1, .., N\}$$

Where

- $r_{i,t}$: return of asset i at time t.
- $\mathbf{f}_t \in \mathcal{R}^K$: vector composed by K factor returns at time t.
- $\boldsymbol{\beta}_i \in \mathcal{R}^K$: vector of factor exposures of stock i.
- $\epsilon_{i,t}$: idiosyncratic risk having zero mean, no correlation with both the factor returns nor the other idiosyncratic risk components.
- α_i: intercept.

There are two main approaches to estimate this model:

- Manually selecting your factors and estimating the exposures through linear regressions.
- Estimate the historical covariance matrix of returns $\boldsymbol{\Sigma}$ and use PCA to extrapolate K principal components (factors) and weights matrix (factor exposure matrix).

Each one of these approaches has flaws. The first approach requires prior knowledge of the behaviour of the returns, which is at best partial and assumes that all the driving risk factors are observable on the market. The second approach does not give a straightforward interpretation of the factors and does not allow the incorporation of any additional information about the market.

Thanks to Instrumented Principal Component Analysis (IPCA), we try to overcome the shortcomings of the static Factor Model using dynamic loadings that are a linear function of stocks' characteristics. Then we employ kernel regression to have loadings that depend on stocks' characteristics in a non-linear way.

Finally, we show the results of all the models in terms of R^2 both in-sample and out-of-sample, highlighting the best models and the relative parameters. In conclusion, we present our final considerations.

II. IPCA Model Description

Instrumented PCA[1] generalizes the previous factor model. It allows dynamic factor exposures ($\boldsymbol{\beta}_{i,t}$) that are dependent on stock characteristics ($\boldsymbol{z}_{i,t}$), and factor extrapolation through PCA. These extensions overcome the problem of the first approach of manual selection of factors, that now don't have to be pre-specified nor completely identifiable. But also allows researchers to incorporate information through the instruments (asset characteristics).
In this framework the model for stock returns is

$$r_{i,t+1} = \alpha_{i,t} + \boldsymbol{\beta}_{i,t}' \mathbf{f}_{t+1} + \epsilon_{i,t+1}$$

Where we have

$$\alpha_{i,t} = \mathbf{z}_{i,t}' \boldsymbol{\Gamma}_\alpha + \nu_{\alpha,i,t}$$
$$\boldsymbol{\beta}_{i,t} = \mathbf{z}_{i,t}' \boldsymbol{\Gamma}_\beta + \nu_{\beta,i,t}$$

With this notation, we identify as $\mathbf{z}_{i,t} \in \mathcal{R}^L$ the vector with L characteristics of asset i at time t, as $\Gamma_\beta \in \mathcal{R}^{L \times K}$ and $\Gamma_\alpha \in \mathcal{R}^{L \times 1}$ the matrices that map the instruments to the factors.
From this moment on we will assume that $\Gamma_\alpha = 0$, since in [1] it is shown that with enough factors this parameter becomes statistically not significant, meaning that the factors explain most of the returns without significant anomalies.
In vector form the model is

$$\mathbf{r}_{t+1} = \mathbf{Z}_t \boldsymbol{\Gamma}_{\boldsymbol{\beta}} \mathbf{f}_{t+1} + \boldsymbol{\epsilon}_{t+1}^*$$

Where $\boldsymbol{\epsilon}_t^*$ is the vector of composite errors and $\mathbf{Z}_t$ the matrix with asset characteristics stack in the rows.
The objective is now to find the unknown parameters $\boldsymbol{\Gamma}_\beta$ and $\mathbf{f}_{t+1}$. We will achieve this by the minimization of the sum of squared errors. The model described in [1] has been augmented by both a regularization on the norm of the unknowns and a weighting

$$\min_{\boldsymbol{\Gamma}_\beta, F} \left\{ \sum_{t=1}^{T-1} \left(\mathbf{r}_{t+1} - \mathbf{Z}_t \boldsymbol{\Gamma}_\beta \mathbf{f}_{t+1}\right)' \boldsymbol{\Omega}_t \left(\mathbf{r}_{t+1} - \mathbf{Z}_t \boldsymbol{\Gamma}_\beta \mathbf{f}_{t+1}\right) + \frac{\lambda_1}{2} \|\mathbf{f}_{t+1}\|_2^2 + \frac{\lambda_2}{2} \|\boldsymbol{\Gamma}_\beta\|_F^2 \right\}$$

As proven in [1], the values of the parameters that satisfy the first-order conditions are

$$\hat{\mathbf{f}}_{t+1} = \left(\hat{\boldsymbol{\Gamma}}_\beta' \mathbf{Z}_t' \boldsymbol{\Omega}_t \mathbf{Z}_t \hat{\boldsymbol{\Gamma}}_\beta + \lambda_1 \mathbf{I}\right)^{-1} \hat{\boldsymbol{\Gamma}}_\beta' \mathbf{Z}_t' \boldsymbol{\Omega}_t \mathbf{r}_{t+1} \qquad \forall t$$

$$\text{vec}\left(\hat{\mathbf{\Gamma}}_{\beta}^{'}\right) = \left(\sum_{t=1}^{T-1} \mathbf{Z}_t' \mathbf{\Omega}_t \mathbf{Z}_t \otimes \hat{\mathbf{f}}_{t+1} \hat{\mathbf{f}}_{t+1}' + \lambda_2 \mathbf{I}\right)^{-1} \left(\sum_{t=1}^{T-1} \left[\mathbf{\Omega}_t^{\frac{1}{2}} \mathbf{Z}_t \otimes \hat{\mathbf{f}}_{t+1}' \mathbf{\Omega}_t^{\frac{1}{2}}\right]' \mathbf{r}_{t+1}\right)$$

Both the original version ($\lambda_1 = \lambda_2 = 0$ and $\mathbf{\Omega}_t = \mathbf{I}$) and the augmented one have been implemented.
From the first-order conditions it is not possible to get an explicit solution. To overcome this problem the standard procedure is to implement a simple numerical scheme of alternating least squares. To initialize the algorithm, it is chosen a starting guess for $\mathbf{\Gamma}_\beta$ as the left eigenvectors corresponding to the leading K eigenvalues of the quantity

$$\mathbf{x}_{t+1} = \frac{\mathbf{Z_t}' \mathbf{r}_{t+1}}{N}$$

The reason for this choice is that this is the solution of an approximation of our problem as extensively explained in [1].

III. RKHS Ridge Regression Model

One step further has been made to generalize the regularized IPCA described in the previous section, allowing for a non-linear relationship between the asset returns and their characteristics. For this purpose, we set up as in [2], a more general setup

$$r_{i,t+1} = \langle g(\mathbf{z}_{t,i}), \mathbf{f}_{t+1} \rangle_{\mathcal{C}} + \epsilon_{i,t+1}$$
$$\forall t \in \{1, .., T\}, \forall i \in \{1, .., N\}$$

Where

- $\mathcal{C}$ could be a generic Hilbert Space, but we used $\mathcal{C} = \mathcal{R}^K$ with K number of factors, as stated previously.
- g: $\mathcal{R}^L \to \mathcal{C}$, is our non-linear factor loading dependent on the asset characteristics. Moreover $g \in \mathcal{G}$ where $\mathcal{G}$ is the Reproducing Kernel Hilbert space spanned by the kernel function $k(\cdot, \cdot)$.

In vector form, with the scalar product of $\mathcal{C} = \mathcal{R}^K$

$$\mathbf{r}_t = g(\mathbf{Z}_t)' \mathbf{f}_t + \boldsymbol{\epsilon}_t$$

Where $g(\mathbf{Z}_t)$ is the matrix containing in the rows the K non-linear factor loadings of $r_{t,i}$ $\forall i \in \{1, .., N\}$. The objective is now to find the unknown factors $\mathbf{f}_{t+1}$ and the function $g(\cdot)$. We will achieve this by performing a Reproducing Kernel Hilbert Space (RKHS) Ridge Regression

$$\min_{g \in \mathcal{G}, F} \left\{ \sum_{t=0}^{T-1} (\mathbf{r}_{t+1} - g(\mathbf{Z}_t)' \mathbf{f}_{t+1})' \mathbf{\Omega}_t (\mathbf{r}_{t+1} - g(\mathbf{Z}_t)' \mathbf{f}_{t+1}) + \frac{\lambda_1}{2} \|\mathbf{f}_{t+1}\|_2^2 + \frac{\lambda_2}{2} \|g\|_{\mathcal{G}}^2 \right\}$$

To solve this minimization problem we used an alternating kernel ridge regression algorithm. The advantages of this approach overcome the non-convexity of the above minimization. Taken separately, and solved recursively both the minimization in the factors set F, given $g \in \mathcal{G}$ and the one in $g \in \mathcal{G}$ given F are conditionally convex problems.
Given as granted g, the minimization to find the factors becomes

$$\min_{\mathbf{f}_{t+1} \in \mathcal{R}^5} \left\{ (\mathbf{r}_{t+1} - g(\mathbf{Z}_t)' \mathbf{f}_{t+1})' \mathbf{\Omega}_t (\mathbf{r}_{t+1} - g(\mathbf{Z}_t)' \mathbf{f}_{t+1}) + \frac{\lambda_1}{2} \|\mathbf{f}_{t+1}\|_2^2 \right\}$$

For the Representer Theorem stated in [2], the solution $\hat{\mathbf{f}}_{t+1} \in span(g(\mathbf{Z}_t))$, and can be written as $\hat{\mathbf{f}}_{t+1} = g(\mathbf{Z}_t)' \mathbf{c}$ for some $\mathbf{c} \in \mathcal{R}^N$. The minimization becomes

$$\min_{\mathbf{c} \in \mathcal{R}^N} \left\{ (\mathbf{r}_{t+1} - \mathbf{G}\mathbf{c})' \mathbf{\Omega}_t (\mathbf{r}_{t+1} - \mathbf{G}\mathbf{c}) + \frac{\lambda_1}{2} \mathbf{c}' \mathbf{G} \mathbf{c} \right\}$$

Where $\mathbf{G} = g(\mathbf{Z}_t)' g(\mathbf{Z}_t)$.
The solution of the above Ridge Regression is

$$\mathbf{c}_{t+1} = (\mathbf{G} + \lambda_1 \mathbf{\Omega}_t^{-1})^{-1} \mathbf{r}_{t+1}$$

and hence the solution can be found using $\hat{\mathbf{f}}_{t+1} = g(\mathbf{Z}_t)' \mathbf{c}$.
On the other hand, given as granted the factors in F, the minimization to find $g \in \mathcal{G}$ becomes

$$\min_{g \in \mathcal{G}} \left\{ \sum_{t=0}^{T-1} (\mathbf{r}_{t+1} - g(\mathbf{Z}_t)' \mathbf{f}_{t+1})' \mathbf{\Omega}_t (\mathbf{r}_{t+1} - g(\mathbf{Z}_t)' \mathbf{f}_{t+1}) + \frac{\lambda_2}{2} \|g\|_{\mathcal{G}} \right\}$$

As proven in [2], the solution to the above problem is of the form:

$$g(\cdot) = \sum_{s=0}^{T-1} \left(k\left(\cdot, \boldsymbol{Z}_s'\right) \mathbf{v}_s\right) \mathbf{f}_{s+1}$$

for some coefficients $\mathbf{v}_s \in \mathcal{R}^N$. Using this theorem we can reformulate the minimization problem as

$$\min_{\mathbf{V} \in \mathcal{R}^{K * N}} \left\{ (\mathbf{R} - \mathbf{Q}\mathbf{V})' \mathbf{\Omega} (\mathbf{R} - \mathbf{Q}\mathbf{V}) + \frac{\lambda_2}{2} \mathbf{V}' \mathbf{Q} \mathbf{V} \right\}$$

and the unique solution for $\mathbf{V}$ is

$$\mathbf{V} = (\mathbf{Q} + \lambda_2 \mathbf{\Omega}^{-1})^{-1} \mathbf{R}$$

Where the variables are defined as

$$\begin{aligned} A_{st} &:= \mathbf{f}_{s+1}' \mathbf{f}_{t+1}, \\ \mathbf{Q}_{t,s} &:= A_{ts} k\left(\boldsymbol{Z}_t, \boldsymbol{Z}_s'\right) \\ \boldsymbol{Q} &:= \left[\boldsymbol{Q}_{t,s} : t, s = 0, \ldots, T-1\right] \\ \boldsymbol{R} &:= [\boldsymbol{r}_1; \ldots; \boldsymbol{r}_T] \\ \boldsymbol{V} &:= [\boldsymbol{v}_0; \ldots; \boldsymbol{v}_{T-1}] \end{aligned}$$

The solution to the RKHS Ridge Regression is found through this alternating least square algorithm run till convergence. To initialize the algorithm, we decided as an initial guess for $\mathbf{f}_t$ $\forall t$, the first iteration of regularized IPCA.
Furthermore, the metric used throughout the whole project is the R^2. In particular, we computed it differently based on the setting:

- IPCA: $R^2 = 1 - \frac{\sum_{i,t} (r_{i,t+1} - \mathbf{z}_{\mathbf{i,t}}' (\mathbf{\Gamma}_{\boldsymbol{\beta}} \boldsymbol{f}_{\boldsymbol{t+1}}))^2}{\sum_{i,t} r_{i,t+1}^2}$
- Kernel Regression: $R^2 = 1 - \frac{||\boldsymbol{R} - \boldsymbol{Q}\boldsymbol{V}||_2^2}{||\boldsymbol{R}||_2^2}$

Then, to evaluate the R^2 out-of-sample, we estimated the model in the following way:

- IPCA: we fixed $\mathbf{\Gamma}_\beta$ estimated in-sample, and we used it to compute factors out-of-sample
- Kernel Regression: we fixed $\boldsymbol{V}$ estimated in-sample, and used it to compute $g(\boldsymbol{Z_t})$, $\boldsymbol{G}$ and consequently $\boldsymbol{c}$ leading to the factors out-of-sample

Finally, we also performed a Low-Rank Approximation method: the algorithm in [3] called Pivoted Cholesky is a special decomposition algorithm applied on the Kernel matrix that reduces the dimension of $\boldsymbol{V}$ from *(number of periods x number of stocks)* to *(number of factors x number of characteristics)*. The method works with large datasets because it avoids the problem of the inversion of the big matrix $\boldsymbol{Q}$.

IV. Data

The data employed in this project comprises a list of datasets indexed by date. Each date corresponds to a month from January 2000 to December 2020, and each dataset contains 94 different characteristics on N stocks, i.e. a matrix of n rows and 94 columns. N is not fixed and varies from month to month, but this problem does not affect our solution since we decided to focus only on the 100 stocks with the largest market capitalization each month.
To process the data, we decided to perform the following operations on each monthly dataset:

- compute the market cap of all stocks.
- drop the stocks with more than 60% of *Nan* characteristic observations.
- drop the stocks with a return smaller than -1, that would imply a negative stock value.
- order the stocks in descending order by market cap.
- select the first 100 stocks.
- drop the 'return' column in the dataset and save it separately.
- standardize the data and fill the *Nan* values with zero.

At the end of the procedure, we return two lists:

- *data*: list of matrices, each of them containing the clean monthly data.
- *ret*: list of stock returns for each month.

In conclusion, throughout the whole project, we decided to keep 5 years of data (from 2000 to 2004) as the training set and one year (2005) as the validation set, for a total of 6 years of data. Then for a rolling window procedure, we used all the data from 2000 to 2020.

V. Results

The graph displayed in Fig. 1 shows the evolution of the in-sample R^2 as a function of the number of iterations of the regression procedure. First, we observe that all the methods reach a *plateaux* at around 10 iterations, i.e. the iterating procedure converges pretty quickly in all three cases. Then, the plot underlines how the Gaussian Kernel Ridge regression greatly outperforms IPCA and regularized IPCA at each step.

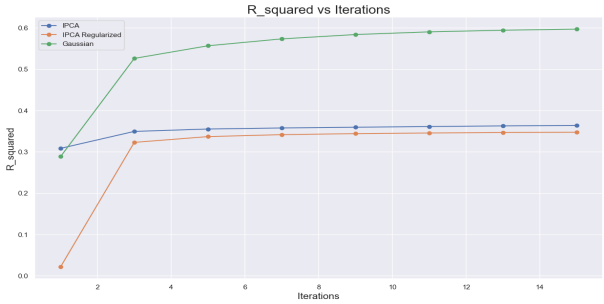

Fig. 1: R^2 as a function of the number of iterations

In addition, we performed a sanity check to confirm that regularized IPCA and the linear Kernel regression are actually the same method. The metric we used is the R^2, evaluated in-sample as well as out-of-sample. The values are displayed in the table below:

TABLE I

Method	**In-sample R^2**	**Out-of-sample R^2**
IPCA Regularised	0.3277	0.1412
Linear Kernel Regression	0.3227	0.1441

As we can see, the the results are very similar, proving the equivalence between the two methods at least in terms of performance.
Another interesting comparison is the difference between the in-sample and out-of-sample performance of IPCA and regularised IPCA. To achieve this goal, we can look at the R^2 and the predictive R^2 of the two methods:

TABLE II

Method	**In-sample R^2**	**Predictive R^2 [1]**
IPCA	0.3668	0.0145
IPCA Regularised	0.3277	0.0267

As we can observe, IPCA has a higher R^2 relative to IPCA in-sample, but it is due to overfitting:, as soon as we consider a forward-looking metric as the predictive R^2, regularised IPCA is significantly better.
Then, we decided to validate the models by selecting the parameters that yielded the best performance in terms of R^2 on the validation set. The sets of hyperparameters we tested our methods on are:

- λ_1: [10, 1, 0.1, 0.01, 0.001, 0.0001, 0.00001, 0.000001]
- λ_2: [1000, 100, 10, 1, 0.1, 0.01, 0.001, 0.0001]
- l: [5, 10, 20]
- α: [5, 10, 20]

TABLE III

Method	λ_1	λ_2	l	α	**Out-of-sample R^2**
IPCA Regularised	0.01	1000	X	X	0.1551
Gaussian KR*	0.01	1	20	X	0.2863
Rational Quadratic KR*	0.01	1	20	20	0.2858

*KR stands for Kernel Regression

Not every method uses all the hyperparameters: in the *Table III* we report the best values by indicating with X the absence of the parameter in the method. We also report the

corresponding out-of-sample R^2.
It is worth noticing that for $\alpha \to \infty$ the Rational Quadratic Kernel tends to the Gaussian Kernel. Observing that the validation on the RQKR always led to the choice of the biggest alpha and that the R^2 was always lower than in the GKR, we deduced that the RQKR was converging to the GKR, thus we decided to use only the latter.
The results of the validation of the hyperparameters can also be shown through 3D surfaces. In particular, the z axis is given by the out-of-sample R^2, while values of λ_1 and λ_2 build respectively the x and y axes.
The resulting plots for the three methods are shown below:

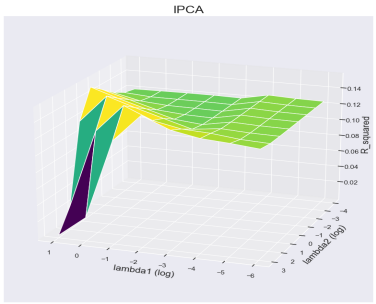

Fig. 2: IPCA surface

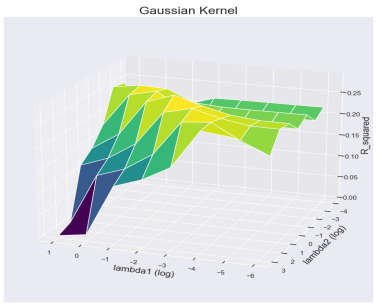

Fig. 3: Gaussian Kernel surface

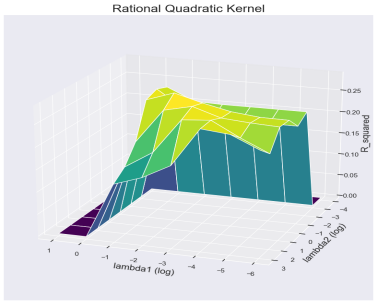

Fig. 4: Rational Quadratic Kernel surface

The observation that in the IPCA surface the R^2 peaks line up on a diagonal led us to an interesting deduction: the values of λ_1 and λ_2 do not matter separately, but to understand their impact one has to look at their product. The result is perfectly coherent with the definition of the iterating procedure used to perform the regression since the R^2 metric relies on the product of the factors and the loadings, which are the elements that we regularize via λ_1 and λ_2 respectively. Furthermore, this result remains true up to a certain level even in the Gaussian and Rational Quadratic Kernel regressions. The property is not as evident as in IPCA, equivalent to Linear Kernel, but the R^2 peaks still line up on a diagonal in Fig. 3 and Fig. 4.

Finally, we want to show the result of the rolling window procedure for the computation of the out-of-sample R^2:

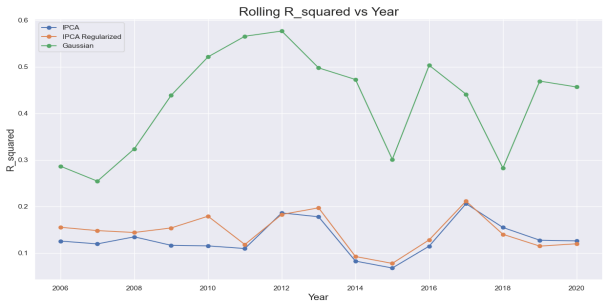

Fig. 5: Rolling window R squared estimation

It is worth noticing that the first time step is on the validation set while from the second onward it is *really* out-of-sample. Then, as we can see, the Gaussian Kernel Regression is consistently better than the IPCA and the regularised IPCA. In addition, we also observe that the estimation of the R^2 is strongly sensitive to data since its values vary significantly from one window to the next.
In conclusion, we report the result of the Low-Rank approximation method on Linear and Gaussian Kernel:

TABLE IV

Method	**In-sample R^2**	**Low Rank R^2**
Linear Kernel Regression	0.3227	0.1629
Gaussian Kernel Regression	0.5831	0.3496

As we can observe from the differences in R^2, the Low-Rank method does not perform well even though the approximation of the Kernel matrix is quite satisfactory (less than 1.5% error). The issue could be that the dimension of the data we are working with is too small, but we would need to perform further analyses to be certain.

VI. Conclusions

We managed to successfully replicate the IPCA procedure of [1], augmenting a factor model with the introduction of *betas* depending on the characteristics of the assets. The out-of-sample performance is improved by the introduction of a regularization parameter: in this setting, we also validate the hypothesis of equivalence with the Linear Kernel Regression. Finally, the Gaussian Kernel Ridge Regression model stands out as the best overall, showcasing superior performance in and out-of-sample, thus confirming the non-linear characteristics-returns relationship.

VII. Ethical Considerations

Throughout the development of our project, ethical considerations surfaced, necessitating a thorough examination of the impact of our factor models. The project's main objective, which is to explain returns using a designated set of factors, encountered a challenge. Despite the efficiency of the IPCA procedure in providing an easily implementable closed-form solution, it grapples with the trade-off of losing interpretability. This limitation implies that we cannot articulate the nature of the factors or identify tradable factors, thereby hindering users from fully comprehending the workings of the solution.

Moreover, the solution does not allow users to make decisions based on it. Since the factors are not tradable, an investor could not use the outcomes of the paper to implement a successful trading strategy.

Moreover, obtaining the characteristics of stocks proves challenging, with the added complication that this information is frequently private. Consequently, the findings of the paper lack reproducibility when altering the set of stocks. Additionally, the outcomes are significantly influenced by the availability and quality of data. Even in cases where the user manages to access all requisite data, the final results may exhibit considerable variability in comparison to those presented in the paper, despite the underlying core meaning remaining theoretically consistent.

On a positive note, our model offers a more comprehensive approach by allowing the consideration of additional characteristics associated with stocks. This expanded capability enables users to incorporate a broader range of elements into their analyses, that are not limited by the market. However, it comes with the aforementioned ethical caveat that, despite this enhanced comprehensiveness, the resulting factors remain foggy.

In conclusion, we are positive that the nature of our model allows us not to have concerns relative to privacy, sustainability, or non-maleficence.

VIII. Acknowledgements

We would like to extend our sincere thanks and heartfelt gratitude to Professor Damir Filipović and Postdoctoral Researcher Urban Ulrych for entrusting us with this project. Special appreciation goes to them for their continuous assistance and support. Their guidance has been instrumental, and we are grateful for their dedication and care throughout this academic journey.

References

[1] B. T. Kelly, S. Pruitt, and Y. Su, "Characteristics are covariances: A unified model of risk and return," *Journal of Financial Economics*, vol. 134, no. 3, pp. 501–524, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304405X19301151

[2] P. Collin-Dufresne, D. Filipovic, and U. Ulrych, "Smart kernel factors," *working paper*, 2023.

[3] D. Filipovic, M. Multerer, and P. Schneider, "Adaptive joint distribution learning," *working paper*, 2023.