**54. Python Generalized Additive Models for Financial Data Analysis**

In this project, I am applying Generalized Additive Models (GAMs) within the context of financial data analysis, particularly focusing on modeling complex relationships in financial datasets. Having previously explored regression models with splines and polynomial regressions, I am now advancing to GAMs to leverage their flexibility in capturing nonlinear relationships in financial variables. I will skip the basic example of modeling a financial metric based solely on a single feature, which is similar to what I have already done, and instead focus on the more intricate aspects of GAMs involving multiple financial features.

The first model I fit includes multiple financial variables, specifically using splines to model a financial outcome as a function of both time (e.g., year) and another continuous variable such as age or experience of investors. While this model isn't strictly a GAM, it utilizes natural splines with different degrees of freedom and employs ordinary least squares regression, akin to the natural spline and B-spline fits I have previously encountered. In this model, I include nonlinear terms for both time and experience, and I account for categorical variables like investment type or sector by fitting constants at each level. By incorporating these three variables—time, experience, and investment type—into my model, I am able to obtain fits for the two nonlinear functions as well as the constants for each investment type.

The key difference between this model and the models I have worked with before is the inclusion of more than one feature, which is a hallmark of additive models. In financial data analysis, it's crucial to examine the individual effects of each variable, and this is typically done through partial dependence plots. Although I won't detail every step of the code required to produce these plots, I demonstrate that each feature in the model yields a nonlinear function. For example, the estimated nonlinear effect of experience on investment returns displays a pattern similar to what I observed when modeling returns based solely on experience, though it differs slightly in certain areas and maintains a similar form.

The essential insight here is that, unlike standard linear models where each variable is associated with a single coefficient, in additive models, each variable is associated with a fitted function. I illustrate the fitted function for experience, but similar functions could be visualized for time, and a set of constants could be plotted for the different levels of investment type, illustrating how additive models provide a broader generalization of linear models.

I also explore the fitted function for time, noting that there are estimated coefficients for each level of investment type, although these are not plotted here. The model I fit here uses ordinary least squares regression, specifying flexible functions for experience and time, along with dummy variables for investment types. This setup does not strictly conform to a generalized additive model, as it lacks the smoothing penalty discussed in the lectures.

Moving forward, I proceed with a similar fit using the pygam library, specifically employing the LinearGAM function, which is tailored for regression problems in financial data analysis. Notably, pygam considers column indices rather than feature names when specifying a model. For instance, in the X\_GAM matrix, the first column might represent experience, and I use a spline for experience with s(0). The second column might represent time, with s(1) specifying the number of splines for time. It is important to understand that while the GAM function does not use variable names for its specifications, the model is fitted using the fit() method.

To visualize the model's effects, I use a function from the ISLP package to create partial dependence plots for the fitted model. Here, I show the partial dependence plots for experience on investment returns when no parameters are specified in the spline function for GAM. This visualization suggests the need for tuning. For instance, setting lam = 0.6 results in a less smooth plot compared to others. I experiment with estimating a lambda that provides a specific number of degrees of freedom, aiming for smoother plots for the time variable. After modifying the lambda values and refitting the model, the resulting partial dependence plots are smoother, utilizing around five degrees of freedom for each effect.

I also include an example of a partial dependence plot for a categorical variable, such as investment type, showing coefficients for each category. As I conclude this section, I provide an example of how to apply binary regression in GAMs to model binary financial outcomes, such as whether an investment is profitable or not. The primary adjustment here is the base estimator; instead of using LinearGAM, I switch to LogisticGAM. The specifications remain largely consistent, with some modifications. For instance, I use a linear effect for time rather than a spline, indicated by the L in the model. I fit the model using the same fit() method and analyze the partial dependence plots.

During this process, I observe unusual behavior in the estimated effects on investment returns for a particular group of investors, specifically those with minimal experience but high returns. This anomaly is likely due to the rarity of such cases, making it difficult to estimate this parameter accurately, which leads to highly variable estimates. To address this, I refit the model, excluding these outliers. The refitted model provides more reasonable estimates of the effects of experience on investment returns across different levels of investment types.

Finally, just as with linear GAMs, I produce partial dependence plots for each feature in this logistic model. Interestingly, the plot for experience does not show the same sharp decline observed when modeling based solely on experience, which is an encouraging result. This could be attributed to the inclusion of more comprehensive investment types in the model, allowing for a more nuanced understanding of the relationships between the financial variables.