**Implementation Techniques and Toolkits**

**Extracting SHAP Values for Factor Models on Stocks/ETFs**

**SHAP (SHapley Additive exPlanations)** is a popular Python library for interpreting ML models. It can explain linear models, tree ensembles (XGBoost, LightGBM, Random Forest), or neural nets by assigning each feature a “fair” contribution to a prediction​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=year%20in%20our%20sample%20period,This)

. In a factor investing context, we train a model (e.g. an ensemble or regression) to predict asset returns or risk using factor exposures (e.g. value, momentum, size) or technical/macro features. We then use SHAP to compute feature importance for each prediction. For example, Robeco researchers forecasting stock crash risk used a tree model and calculated SHAP values to see which features (volatility, valuation ratios, etc.) drove the predicted crash probability​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=year%20in%20our%20sample%20period,This)

​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=Figure%207%20,Values%20to%20the%20right%20indicate)

. A **SHAP summary plot** (beeswarm) can highlight the top factors influencing returns in a given year, with dot colors showing feature magnitude (e.g. high vs low value)​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=The%20importance%20of%20the%20anonymized,The)

​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=Figure%207%20,Values%20to%20the%20right%20indicate)

. In practice, one would use shap.TreeExplainer(model).shap\_values(data) for tree-based models or shap.LinearExplainer for linear factor models. The SHAP documentation and tutorials provide examples of computing and visualizing feature attributions​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=The%20goal%20of%20this%20section,With)

​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=the%20returns%20of%20the%20variables,returns%20are%20positively%20correlated%20with)

. Notably, Amundi researchers in 2022 applied SHAP to a gradient boosting model predicting credit excess returns and averaged the absolute SHAP values over time to gauge each factor’s impact​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=The%20goal%20of%20this%20section,With)

​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=the%20returns%20of%20the%20variables,returns%20are%20positively%20correlated%20with)

. They found, for instance, that **value** and **duration (DTS)** factors had consistently high SHAP contributions (positive correlation between factor returns and their SHAP values), whereas the **size** factor had negative correlation – indicating size mattered inversely in their model​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=The%20absolute%20average%20SHAP%20values,the%20pink%20bars%20indicate%20whether)

​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=match%20at%20L1453%20their%20SHAP,DTS%20and%20value%20factors%20on)

. Such analysis requires simply computing SHAP on rolling windows or different periods and aggregating. Python toolkits: **SHAP** library (which integrates with scikit-learn, XGBoost, etc.), along with data handling libraries (pandas) to align factor data with predictions. Many finance ML papers (and blogs) provide code: e.g. Unigestion (2020) used SHAP to explain which hedge fund characteristics drove predicted Sharpe ratios, grouping features into categories (returns, qualitative, macro)​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=to%20gauge%20which%20input%20factors,explanatory.%20The%20qualitative%20features)

​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=Sortino%20ratio%2C%20alpha%20and%20t,for%20performance%20forecasting%2C%20under%20the)

. They confirmed SHAP results aligned with intuition (e.g. certain “Nowcaster” macro regime indicators consistently had high importance)​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=Sortino%20ratio%2C%20alpha%20and%20t,for%20performance%20forecasting%2C%20under%20the)

​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=regularised%20linear%20specification%2C%20Random%20Forest,the%20SHAP%20variable%20importance%20to)

. This shows that extracting SHAP values is feasible with Python and helps interpret factor models on ETFs or stocks; one can find similar examples in Jupyter notebooks from industry whitepapers (for instance, Macrosynergy’s public notebook on classifying credit markets uses a random forest – one could easily add SHAP analysis to see which macro factors drive its predictions​

[macrosynergy.com](https://macrosynergy.com/research/classifying-credit-markets-with-macro-factors/#:~:text=performance%20and%20create%20a%20chance,of%20returns%20and%20produce%20good)

​

[macrosynergy.com](https://macrosynergy.com/research/classifying-credit-markets-with-macro-factors/#:~:text=A%20Jupyter%20notebook%20for%20audit,data%20sets%20for%20research%20projects)

).

**SHAP-Based Dynamic Metrics (Factor Dominance, Stability, Interaction)**

Once SHAP values are computed over time, we can derive **dynamic metrics** to quantify regime changes in factor importance:

* **Factor Dominance Momentum:** This idea extends the concept of *factor momentum* (the tendency of factor returns to continue, as in Ehsani & Linnainmaa 2019). Instead of returns, we look at the dominant explanatory factor. For each period (say each month or quarter), identify which factor had the highest average SHAP contribution to asset returns. Factor Dominance Momentum measures the persistence of that rank – i.e. does the same factor remain dominant in subsequent periods? If a particular factor has been the top driver recently, and continues to dominate next period, that indicates a stable regime centered on that factor. This metric can be quantified by the autocorrelation of the top factor’s dominance or by constructing a “factor leadership index” that is +1 when the previously dominant factor stays on top and 0 otherwise. While not explicitly named in literature, the concept relates to timing factors based on recent performance or importance. For example, Fan *et al.* (2021) found only 6 out of 20 factors exhibit strong return continuation and “dominate” a factor momentum portfolio​

[quantpedia.com](https://quantpedia.com/a-deeper-look-into-factor-momentum/#:~:text=momentum%20effect,weighted%20portfolios)

. By analogy, a SHAP-based dominance momentum would focus on continuation of *explanatory power*. If the **value** factor’s SHAP importance has been highest for several months, one might overweight value tilts going forward, assuming that regime persists (this aligns with factor timing strategies being a “type of factor momentum strategy”​

[quantpedia.com](https://quantpedia.com/a-deeper-look-into-factor-momentum/#:~:text=%E2%80%9CA%20factor%20momentum%20strategy%20is,2019%2C%20Ehsani%20and%20Linnainmaa%2C%202021)

).

* **Explanation Stability Index:** This index measures how consistent the model’s explanations are over time. One approach is to calculate the similarity of feature importance rank orders or vectors between periods. For instance, we can compute the Spearman rank correlation of the SHAP value vector at time *t* vs. *t-1*. The average correlation (or 1 minus variance) can serve as a *stability index*. A high value means the set of important factors remains roughly the same – indicating a stable regime – whereas a low value signals a regime shift (different factors driving returns). In practice, one could also introduce perturbations to data and check stability: an open-source project proposed a **“stability index for local interpretability”** that quantifies consistency of feature rankings under data perturbation​

[github.com](https://github.com/edo1691/ad_shap_stability#:~:text=,predictions%20even%20under%20varying%20conditions)

. That idea, introduced for model trust, can be adapted to time-series: treat each time window as a “perturbation” and see if the same factor remains top-3 in SHAP importance. If explanations fluctuate greatly, the Stability Index is low, flagging a potential regime change or model unreliability. This metric has roots in XAI research – e.g. studies on SHAP and LIME explanation variance​

[medium.com](https://medium.com/towards-data-science/instability-of-lime-explanations-3e0efc00a7de#:~:text=Instability%20of%20LIME%20explanations%20,explanations%20in%20the%20graphs%2C)

​

[arxiv.org](https://arxiv.org/html/2312.12115v1#:~:text=Shaping%20Up%20SHAP%3A%20Enhancing%20Stability,approach%20that%20achieves%20full)

– but applied to temporal consistency. A concrete example: if over 2020–2021 the model’s top features were consistently **momentum and quality**, but in 2022 **volatility** jumped to the top, the index would drop, indicating a new regime with different drivers.

* **Factor Interaction Matrix:** SHAP can quantify not only individual feature effects but also **feature interactions**. For tree models, shap.TreeExplainer can compute SHAP interaction values, producing an M×M matrix for M features​

[christophm.github.io](https://christophm.github.io/interpretable-ml-book/shap.html#:~:text=18%20SHAP%20%E2%80%93%20Interpretable%20Machine,How%20can%20we)

. Entry (i,j) in this matrix represents the contribution of the interaction between factor *i* and *j* to the prediction. By averaging these interaction matrices over a time window, we can obtain a **Factor Interaction Matrix** for that period. This matrix reveals how pairs of factors jointly affect returns in that regime. For example, Amundi’s study noted an interesting interaction: the importance of the **DTS (duration)** factor increased linearly with its own value *and* in conjunction with the **value** factor during certain periods​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=their%20SHAP%20values%2C%20while%20the,1%20show%20the%20relationship)

. By examining SHAP interaction values, they observed that in their nonlinear model the combination of low duration and high value had an outsized effect on credit returns. In implementation, one would compute interaction SHAP values for each date (which yields a matrix per observation) and then aggregate (mean or median) across a window to get a representative interaction matrix for that regime. Visualizing this as a heatmap can highlight, for instance, that **value and size** factors together contributed strongly (perhaps indicating their effects reinforce each other in a bull market), whereas in a different regime the **growth–momentum** interaction lights up. This kind of analysis goes beyond standard factor correlation by capturing nonlinear effects. Tooling: the SHAP library’s interaction outputs​

[christophm.github.io](https://christophm.github.io/interpretable-ml-book/shap.html#:~:text=18%20SHAP%20%E2%80%93%20Interpretable%20Machine,How%20can%20we)

and Python visualization (seaborn heatmaps) suffice. Academically, the interpretation of SHAP interaction is still emerging, but it aligns with the goal of **explainable factor models** – identifying not just which factors matter but how they interplay.

Together, these SHAP-driven metrics (dominance momentum, stability, interaction) form a toolkit to **quantify regime dynamics** in factor investing. They are custom measures, so one might not find them by name in textbooks, but they draw on concepts found in literature: e.g. factor momentum timing​

[quantpedia.com](https://quantpedia.com/a-deeper-look-into-factor-momentum/#:~:text=%E2%80%9CA%20factor%20momentum%20strategy%20is,2019%2C%20Ehsani%20and%20Linnainmaa%2C%202021)

, explanation stability in XAI​

[github.com](https://github.com/edo1691/ad_shap_stability#:~:text=,predictions%20even%20under%20varying%20conditions)

, and interaction effects in tree-based models​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=their%20SHAP%20values%2C%20while%20the,1%20show%20the%20relationship)

. By monitoring these, a practitioner can declare, for example, “the Explanation Stability Index dropped sharply and a new factor took dominance – suggesting we’ve entered a different regime driving equity returns.”

**Hybrid Clustering for Regime Detection (t-SNE + DTW)**

To detect regimes, we often need to cluster time periods with similar characteristics. A **hybrid clustering** approach can marry dimensionality reduction techniques like t-SNE with time-series distance measures like DTW (Dynamic Time Warping). Here’s how such an approach can be implemented:

* **Feature Construction:** First decide what features represent a “market state.” This could be the vector of factor SHAP values at a given time, or the recent history (trajectory) of certain variables (e.g. last 3 months of returns or SHAP outputs). For example, one might create a vector of the last 6 months of each factor’s average SHAP contribution – effectively a short time-series per factor, representing recent dominance patterns.
* **Distance Measure – DTW:** Because different regimes might unfold at different speeds, using Euclidean distance on raw time-series can be misleading. Dynamic Time Warping is a robust method to measure similarity between time-series that may be misaligned or vary in length. DTW finds the optimal nonlinear alignment between two sequences and computes a distance that accounts for shifts in time​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=match%20at%20L152%20DTW%20calculates,for%20time%20shifting%2C%20stretching%2C%20and)

​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=DTW%20calculates%20the%20minimum%20distance,for%20time%20shifting%2C%20stretching%2C%20and)

. In practice, the tslearn library in Python provides convenient DTW-based clustering. For instance, TimeSeriesKMeans(metric="dtw") can cluster sequences using DTW as the distance measure​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=match%20at%20L178%20model%20%3D,max_iter%3D200%29%20model.fit_predict%28data%29%20ssd.append%28model.inertia)

. This has been used to cluster stocks by price trend regardless of phase shifts​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=Dynamic%20Time%20Warping%20,different%20lengths%20and%20different%20speeds)

​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=analysis,similarity%20between%20time%20series%20data)

, and similarly can cluster market periods by factor trend. DTW ensures that, say, a slow-building bull regime in 2017 could be recognized as analogous to a faster V-shaped recovery in 2020 if their factor importance patterns are similar but stretched/compressed in time.

* **Dimensionality Reduction – t-SNE:** t-SNE (t-distributed Stochastic Neighbor Embedding) is a non-linear technique to embed high-dimensional data into 2D or 3D for visualization of clusters. One strategy is to compute pairwise DTW distances between all pairs of time windows (e.g. between every month’s last-6-month pattern) and then use t-SNE on that distance matrix to project each period into 2D. Periods that cluster together in the t-SNE plot are likely part of the same regime. Yale researchers Greengard *et al.* (2020) demonstrated the power of t-SNE in clustering financial strategies: they embedded historical return profiles of 20+ anomaly factors into 2D, which **endogenously grouped** them into six clusters corresponding to known styles (value, momentum, etc.)​

[economics.yale.edu](https://economics.yale.edu/sites/default/files/2022-10/tSNE_draft15.pdf#:~:text=%28t,by%20their%20corresponding%20first%20principal)

. This suggests t-SNE is effective at revealing structure in high-dimensional finance data. In our use-case, each “point” is a time period characterized by multiple factor signals; t-SNE can reveal distinct clouds for, say, **“low-vol, growth-led regime”** vs **“high-volatility, momentum-crash regime.”** We might perform clustering on the t-SNE output (e.g. DBSCAN or k-means in the embedded space) to formally assign regime labels.

* **Hybrid Workflow:** A hybrid regime detection could be: (1) Compute a matrix of DTW distances between rolling windows of factor SHAP vectors (to compare shapes of factor importance trajectories). (2) Apply t-SNE on this distance matrix to get a 2D map. (3) Visually identify clusters or use a clustering algorithm to separate them. The result is groups of time windows that are similar in factor influence patterns. Each cluster corresponds to a **dynamic factor regime**. This approach is model-free and unsupervised. It contrasts with parametric Hidden Markov Models (HMMs) or Markov-switching, which assume specific distributions. Instead, it leverages the actual shapes of data. Academic support for such methods is growing: Horváth, Issa & Muguruza (2024) introduced a *Wasserstein k-means* algorithm (distance between return distributions) to cluster market regimes, outperforming classical methods especially when return distributions are non-normal​

[risk.net](https://www.risk.net/journal-of-computational-finance/7959937/clustering-market-regimes-using-the-wasserstein-distance#:~:text=,discrepancy%20approach%2C%20validating%20qualitative%20results)

​

[risk.net](https://www.risk.net/journal-of-computational-finance/7959937/clustering-market-regimes-using-the-wasserstein-distance#:~:text=The%20problem%20of%20rapid%20and,Wasserstein)

. Their approach is conceptually similar – using a specialized distance (Wasserstein) and clustering to detect regimes without assuming a prior model. Likewise, *DTW + t-SNE* uses a specialized distance for time-series shapes. While I am not aware of a published paper that exactly combines t-SNE and DTW for financial regimes, the components are well-known: DTW clustering is used in quant research (e.g. clustering stocks by similar price movements​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=model%20%3D%20TimeSeriesKMeans%28n_clusters%3Di%2C%20metric%3D,inertia)

​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=The%20following%20code%20merges%20the,labels_%29%20into)

), and t-SNE has been used to cluster both strategies​

[economics.yale.edu](https://economics.yale.edu/sites/default/files/2022-10/tSNE_draft15.pdf#:~:text=%28t,by%20their%20corresponding%20first%20principal)

and market conditions (for example, one could embed volatility term structure shapes to find regimes). Implementing this in Python involves tslearn (for DTW distance and clustering) and sklearn.manifold.TSNE. There are code examples on forums and Medium; for instance, Xiaoguang Li (2023) provides a step-by-step DTW clustering of stock price series​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=Clustering%20Stocks%20Based%20on%20Close,Price)

​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=model%20%3D%20TimeSeriesKMeans%28n_clusters%3Di%2C%20metric%3D,inertia)

, which can be adapted to our use-case by treating “time periods” as the objects to cluster.

**SHAP-Informed Portfolio Optimization (Custom Risk/Return Definitions)**

Once we have SHAP-derived insights (which factors are driving returns, current regime classification, etc.), we can incorporate them into portfolio construction. This means defining custom risk/return measures based on the explanatory features rather than just historical returns. Some practical techniques and tools for this include:

* **Custom Return Estimates:** Instead of using plain expected returns (historical means or analyst forecasts), one can use a model’s predictions or factor-based forecasts. For example, if a gradient boosting model predicts each stock’s alpha and we trust it more in certain regimes, we might set the “expected return” of each asset equal to its model-predicted return (which is informed by current factor conditions). A SHAP-informed twist is to adjust these estimates by emphasizing stable factor contributions. Suppose SHAP analysis shows that a stock’s positive outlook is 80% driven by **Factor A** which we deem reliable in the current regime, and 20% by **Factor B** which is very volatile. We might haircut the portion from Factor B. In effect, the **return input** to optimization could be a weighted sum of SHAP values for “robust” factors. This is a subjective but defensible approach: it’s akin to tilting expected returns toward factors with high *Explanatory Stability*. Researchers are beginning to integrate ML “explanations” into allocation – e.g. a 2024 study by Shu & Mulvey constructs a multi-factor portfolio by first identifying bull/bear regimes for each factor and then feeding those views into a Black–Litterman model​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=and%20growth,interpretability%20compared%20to%20traditional%20methods)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=A%20hypothetical%20single,the%20EW%20benchmark%20itself%2C%20the)

. In their case, if their regime model signals **momentum** is in a bull phase and **size** in a bear phase, they adjust expected returns for those factor indices accordingly in the optimization. This is analogous to using SHAP signals: if SHAP indicates momentum is currently a key positive driver, a SHAP-informed portfolio might overweight momentum-heavy assets or long a momentum ETF. Tools like **Black–Litterman** (available via packages like cvxportfolio or custom code) allow blending such views with a prior.

* **Custom Risk Measures:** Traditional mean-variance optimization uses covariance for risk. SHAP-informed risk definitions could include penalties for explanation instability or unwanted factor exposures. For instance, one might define a risk penalty = portfolio variance + λ \* (volatility of SHAP importance over last N periods). This means the optimizer prefers portfolios whose performance is driven by consistent factors, all else equal. Another idea is to limit exposure to certain factor regimes: e.g. constrain that no more than 50% of portfolio risk comes from a single factor’s contribution. In a factor model context, one can compute the risk contribution of factor *i* to the portfolio as w’βiσiw’ \beta\_i \sigma\_iw’βi​σi​ (for beta exposures and factor vol). If SHAP indicates Factor *i* is highly dominant in current returns, one might set a constraint to not let the portfolio lean entirely on that factor (to avoid regime concentration risk). These kinds of constraints can be implemented in **CVXPY**, a Python convex optimization toolkit, by adding linear or quadratic constraints. For example, one can constrain factor exposure: the MOSEK Portfolio Cookbook shows how factor models decompose portfolio variance and how adding factor exposure limits can control risk​

[docs.mosek.com](https://docs.mosek.com/portfolio-cookbook/factormodels.html#:~:text=The%20purpose%20of%20factor%20models,6)

​

[docs.mosek.com](https://docs.mosek.com/portfolio-cookbook/factormodels.html#:~:text=Let%20%5C%28Z_t%5C%29%20be%20an%20%5C%28N%5C%29,linear%20function%20of%20components%20as)

. Using those formulas, a SHAP-informed investor could say “during a regime where **volatility** factor SHAP is spiking, keep the portfolio’s exposure to volatility below X.” Technically, if βvol\beta\_{vol}βvol​ is the vector of vol factor loadings for assets, we can require w’βvol<Xw’ \beta\_{vol} < Xw’βvol​<X. This becomes a linear constraint that CVXPY can handle.

* **Multi-Objective Optimization:** Often we want to balance returns, risk, and perhaps a third objective (like explanation stability). Modern solvers like MOSEK (which CVXPY can use as a backend) are efficient at quadratic programming, so one can solve: maximize (expected return – λ₁ \* variance – λ₂ \* instability). Such custom criteria were not traditionally used, but with ML insights they become viable. If needed, one can linearize some terms or do iterative optimization (first find max return at given stability, etc.). There is active exploration in this area: for instance, **Kaiju’s “WK-means” regime paper** noted that once they can estimate mean/variance/correlation of assets by regime, those feed into a **Markowitz optimization** to form trading strategies​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=method%20for%20multidimensional%20data,these%20values%20can%20be%20used)

​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=in%20the%20context%20of%20Modern,strategies%20when%20using%20two%20assets)

. In a similar spirit, once we have SHAP-based estimates of “which factor will drive returns,” we can optimize a portfolio to align with that factor (enhancing return) but hedge others (reducing risk). Practically, libraries like **cvxportfolio** (an open-source library from the Stanford/Kaiju group for portfolio optimization) allow one to plug in custom alpha forecasts and risk models​

[github.com](https://github.com/cvxgrp/cvxportfolio#:~:text=cvxgrp%2Fcvxportfolio%3A%20Portfolio%20optimization%20and%20back,described%20in%20the%20accompanying%20paper)

​

[github.com](https://github.com/cvxgrp/cvxportfolio#:~:text=Cvxportfolio%20is%20an%20object,described%20in%20the%20accompanying%20paper)

. One could integrate a SHAP-informed alpha as the forecast and a factor covariance for risk, then solve for optimal weights. Tools like **PyPortfolioOpt** (for mean-variance and risk budgeting) or even simpler, **PuLP/CVXOPT**, can solve these with custom inputs. For example, an optimization could target maximizing the portfolio’s total SHAP attribution to “good” factors per unit of risk. While this exact approach isn’t standard, it builds on the idea of **performance attribution via Shapley values**. Moehle *et al.* (2021) used Shapley values to attribute portfolio P&L to strategy components and noted it provides a fair allocation of performance​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=We%20consider%20an%20investment%20process,1%20Introduction)

​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=be%20active%20,more%20than%20just%20three%20features)

. In reverse, one could allocate capital in proportion to the positive Shapley attributions of certain factors.

* **Available Code & Data:** Many building blocks are available. **CVXPY** documentation and example notebooks (e.g. basic portfolio optimization problems) illustrate how to impose custom constraints​

[tirthajyoti.github.io](https://tirthajyoti.github.io/Notebooks/Portfolio_optimization.html#:~:text=Therefore%2C%20the%20central%20optimization%20problem,risk%20below%20a%20certain%20threshold)

​

[tirthajyoti.github.io](https://tirthajyoti.github.io/Notebooks/Portfolio_optimization.html#:~:text=In%20this%20article%2C%20we%20will,efficiently%20using%20simple%20Python%20scripting)

. **MOSEK’s cookbook** demonstrates factor model usage in risk calculations, which can be translated to CVXPY code​

[docs.mosek.com](https://docs.mosek.com/portfolio-cookbook/factormodels.html#:~:text=The%20purpose%20of%20factor%20models,6)

​

[docs.mosek.com](https://docs.mosek.com/portfolio-cookbook/factormodels.html#:~:text=Let%20%5C%28Z_t%5C%29%20be%20an%20%5C%28N%5C%29,linear%20function%20of%20components%20as)

. Data-wise, classic factor returns (Fama-French factors, momentum, etc.) are public (Kenneth French’s library), and one can compute factor loadings for ETFs/stocks via regressions. SHAP can use those as features in a model to get time-varying importance. There are also datasets like Kaggle’s “Numerai” or “Two Sigma” competitions where participants have shared notebooks using SHAP on stock features. While not directly about optimization, those notebooks show how to integrate SHAP analysis in a workflow. As a concrete starting point, one might use **Fama-French 5 factors** and train an XGBoost to predict monthly returns of S&P500 stocks, then use SHAP (with shap.TreeExplainer) to get factor attributions per stock per month. From there, design a portfolio that, for example, **overweights stocks with positive SHAP for momentum and value** in a regime where those are dominant, and solve for weights via CVXPY under usual constraints (budget=1, no shorting or bounded short, etc.). The end result is a SHAP-informed allocation that differs from a naive cap-weight or equal-weight by tilting toward the explanatory signals the model identifies.

**Relevant Literature and Methodology Comparison**

**Explainable ML (SHAP and XAI) in Financial Market Analysis**

**Using SHAP and ML explanations in finance** has gained traction since 2020, as practitioners seek to open the “black box” of complex models. A few notable papers/whitepapers in recent years:

* *Cherief et al. (2022, Amundi)* – “**Credit Factor Investing with Machine Learning Techniques**”. They compare a tree-based model to a linear factor model for corporate bond returns​

[papers.ssrn.com](https://papers.ssrn.com/sol3/Delivery.cfm/SSRN_ID4155247_code4709150.pdf?abstractid=4155247&mirid=1#:~:text=The%20most%20common%20models%20to,alternative%20factors%20to%20a%20traditional)

. Importantly, they apply TreeSHAP to measure each factor’s contribution by period​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=model,and%20during%20market%20crisis%20periods)

​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=The%20goal%20of%20this%20section,With)

. They report which factors **prevail in different time horizons and crisis periods**​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=model,and%20during%20market%20crisis%20periods)

. For example, over 2015–2021, their SHAP analysis revealed that alternative factors (like liquidity and ESG) added explanatory power on top of traditional factors, especially during stress periods​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=The%20goal%20of%20this%20section,With)

​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=the%20returns%20of%20the%20variables,returns%20are%20positively%20correlated%20with)

. They even visualized interactions: in their Appendix, plots show how the SHAP value of the **DTS factor** increases with the factor’s own return and in combination with the **value factor**, indicating a non-linear interaction in explaining excess returns​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=their%20SHAP%20values%2C%20while%20the,1%20show%20the%20relationship)

. This paper is a prime example of using ML explanations to **analyze factor importance dynamically**, bridging the gap between pure quant models and economic intuition.

* *de Wit et al. (2022, Robeco)* – “**Forecasting Stock Crash Risk with Machine Learning**”. This whitepaper builds a random forest model to predict the probability of a stock crashing. They use SHAP values to interpret the model’s output​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=year%20in%20our%20sample%20period,This)

​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=this%20is%20indicated%20by%20it,31st%20Conference%20on%20Neural%20Information)

. Figure 7 of their paper shows a beeswarm plot of the 20 most important features for distress prediction​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=Figure%206%20,of%20each%20dot%20indicates%20the)

​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=Figure%207%20,Values%20to%20the%20right%20indicate)

. They found that features like **volatility**, **market beta**, and **distance-to-default** had the highest SHAP contributions to crash risk​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=The%20average%20marginal%20contribution%20of,increase)

. The SHAP analysis also illustrated feature effects (e.g. high volatility pushes crash probability up – lots of red points on the right side of the SHAP plot). They then translate these insights into a portfolio application: by excluding the 5% of stocks with worst predicted crash risk (highest ML distress probability), they improved portfolio returns by 33–66 bps compared to the market​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=and%20the%20returns%20are%20annualized,distress%20probability%2C%20for%20the%20period)

​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=10%2C2,13)

. This demonstrates the use of ML explanations in **risk management** – identifying which risk factors matter and acting on that information.

* *Unigestion (2020)* – “**Machine Learning for Fund Selection**”. In this industry whitepaper, Unigestion applied ML models to select hedge funds, predicting performance metrics like Sharpe ratio. They explicitly employed SHAP at two levels: (1) **Global importance** – to confirm which input features are most predictive overall (they grouped predictors into qualitative, returns-based, and macro regime features, finding consistent signals like certain **“Nowcaster” macro regime indices** among top drivers)​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=to%20gauge%20which%20input%20factors,explanatory.%20The%20qualitative%20features)

​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=Sortino%20ratio%2C%20alpha%20and%20t,for%20performance%20forecasting%2C%20under%20the)

. (2) **Individual explanations** – for a given fund, SHAP values show which characteristics make it stand out (e.g. a fund might have a positive SHAP due to low downside deviation and strong past alpha)​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=Sortino%20ratio%2C%20alpha%20and%20t,for%20performance%20forecasting%2C%20under%20the)

​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=regularised%20linear%20specification%2C%20Random%20Forest,the%20SHAP%20variable%20importance%20to)

. They present a chart (Figure 2) of the average SHAP values by predictor category across investment styles, showing, for example, that in *Macro Directional* funds, **“Prime Broker Score”** (a qualitative proxy) and **volatility** were consistently influential​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=we%20employ%20the%20SHAP%20value,attributing%20a%20rank%20to%20the)

​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=Sortino%20ratio%2C%20alpha%20and%20t,for%20performance%20forecasting%2C%20under%20the)

. This work is notable for integrating domain knowledge (40 years of experience) with ML and using SHAP to ensure the model’s picks align with economic intuition (they note the SHAP analysis revealed intuitive patterns, lending trust to the model)​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=regularised%20linear%20specification%2C%20Random%20Forest,the%20SHAP%20variable%20importance%20to)

​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=manager%20in%20top%20quintile%20for,when%20a%20high%20degree%20of)

.

* *Moehle, Boyd, Ang (2021)* – “**Portfolio Performance Attribution via Shapley Value**”. While not about SHAP for features per se, this academic article applies the Shapley value concept to portfolio strategy components​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=We%20consider%20an%20investment%20process,1%20Introduction)

. They attribute the performance of an investment strategy to various “features” (which could be sub-strategies, constraints, or signals) in a way that fairly allocates marginal contributions​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=via%20Shapley%20Value%20Nicholas%20Moehle,attribution%20method%20due%20to%20Shapley)

​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=be%20active%20,more%20than%20just%20three%20features)

. This is relevant because it’s another use of explainability in finance: instead of asking “which feature of the market drove returns?”, they ask “which feature of our process drove P&L?”. The Shapley attribution method they propose can be computed exactly or approximately for portfolio returns​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=any%20specific%20case,We%20are)

​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=process%20that%20relies%20on%20daily,more%20than%20just%20three%20features)

. It’s an example of how ideas from cooperative game theory (which underlie SHAP) are being used in finance to dissect outcomes. This complements SHAP analysis of ML models by showing the versatility of Shapley values (features need not be input variables – they can be strategies or decisions).

In summary, recent literature shows a clear trend: **ML models are used to forecast returns or risks, and SHAP/explainable AI tools are used to interpret those models for insight.** These insights are then fed back into strategy design. The above works span credit, equities, hedge funds, and portfolio processes, indicating broad adoption. We also see a mix of academic and industry contributions. Academic finance has embraced explainable ML especially since 2020, often inspired by the success of ML in stock selection (e.g. the well-cited *Gu, Kelly, and Xiu (2020)* paper on empirical asset pricing with ML, which, while focusing on predictive accuracy, also opened the door to examining which predictors (firm characteristics) matter – some studies like *Chen, Pelger, Zhu (2021)* follow-up by interpreting ML asset pricing models’ learned factors). In industry, where decision accountability is crucial, SHAP is used to build investor trust. For example, **JPMorgan AI research** has published blog posts using SHAP to explain credit modeling or macro forecasts (one example: an AI blog showing SHAP to explain a NLP model’s interest rate predictions, albeit not directly portfolio-focused). The literature is converging on the view that **transparency** via tools like SHAP can enhance financial modeling by providing *economic narratives* for data-driven signals.

To tie this back to “SHAP-Informed Dynamic Factor Regimes”: the idea of tracking factor importance over time and adjusting portfolios is essentially an intersection of these explainable ML works (which tell us how to get factor importance) and the regime-switching works (which tell us how to act when drivers change).

**Dynamic Regime-Switching Models Driven by Explanatory Features**

Traditional regime-switching models (e.g. Hamilton’s Markov switching model for GDP/markets​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=market%20crashes,have%20been%20developed%20and%20applied)

) typically use **return series** or volatility as the input to identify regimes (bull vs bear, low-vol vs high-vol). The user’s query emphasizes *causal or explanatory features* – meaning regimes defined by *what drives the market* rather than just how returns behave. Several recent works align with this philosophy:

* **Macro Factor–Driven Regimes:** Gholkar & Sueppel (2025, Macrosynergy) classify credit market regimes using **point-in-time macro indicators**​

[macrosynergy.com](https://macrosynergy.com/research/classifying-credit-markets-with-macro-factors/#:~:text=performance%20and%20create%20a%20chance,of%20returns%20and%20produce%20good)

. They use features like lending surveys, credit growth, sentiment, etc., and feed them into classifiers (naive Bayes, logistic, random forest) to predict whether credit returns will be bullish or bearish. Essentially, the regime (positive or negative bias of credit spreads) is **forecasted by causal features** (macro conditions) rather than discovered from returns alone. They report decent out-of-sample accuracy and economic value, with random forests performing best​

[macrosynergy.com](https://macrosynergy.com/research/classifying-credit-markets-with-macro-factors/#:~:text=statistical%20learning%20processes%20that%20sequentially,power%20and%20economic%20value%20generation)

. This work is an example of *feature-based regime identification*: a recessionary macro environment vs an expansionary one will put credit markets in different states. They even provide a **Jupyter notebook** with the workflow​

[macrosynergy.com](https://macrosynergy.com/research/classifying-credit-markets-with-macro-factors/#:~:text=A%20Jupyter%20notebook%20for%20audit,data%20sets%20for%20research%20projects)

, which is helpful for implementation. The concept here is similar to using SHAP factors – one could imagine using SHAP of macro features to say “which macro factor is dominating now?” and thereby label regimes (“liquidity-driven market” vs “growth-driven market”). The Macrosynergy piece doesn’t explicitly use SHAP, but it embodies the idea of regimes defined by **exogenous drivers**.

* **Clustering/Jump Models for Regimes:** McGreevy *et al.* (2024) – “**Detecting Multivariate Market Regimes via Clustering**” – propose an unsupervised method to find regimes by clustering the distribution of returns​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=In%20this%20paper%20we%20study,step%20approach%20to%20clustering)

​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=method%20for%20multidimensional%20data,these%20values%20can%20be%20used)

. They introduce a two-step algorithm using either **Wasserstein distance** or Maximum Mean Discrepancy (MMD) to measure difference between return distributions, then k-means to cluster them​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=portfolio%20design,new%20approach%20can%20be%20used)

. This is model-free and captures regimes characterized by changes in mean/volatility/correlation of asset returns​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=method%20for%20multidimensional%20data,variance%20and%20correlation%20between%20two)

. While their features are still return-based (they feed in asset pair return series), the technique is novel in that it doesn’t assume Gaussian returns or a parametric form – regimes emerge from differences in *empirical distributions*. They demonstrate it on both synthetic data and S&P500 pairs trading, and show it can **estimate time-varying mean, variance, correlation** for the pair at each regime​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=performance%20of%20the%20algorithm%20endowed,strategies%20when%20using%20two%20assets)

, which then informs portfolio decisions. This method is complementary to feature-driven regimes: it finds regimes in a purely statistical way, but because it yields distribution characteristics, one could attach **explanations** (e.g. “Regime 1 has high vol and correlation – likely a crisis regime”). Indeed, they mention using the results in context of Modern Portfolio Theory to form strategies​

[theaifinancefrontier.beehiiv.com](https://theaifinancefrontier.beehiiv.com/p/detecting-multivariate-market-regimes-via-clustering-algorithms#:~:text=%E2%9E%A1%20The%20paper%20introduces%20a,distances%20or%20Maximum%20Mean%20Discrepancies)

​

[theaifinancefrontier.beehiiv.com](https://theaifinancefrontier.beehiiv.com/p/detecting-multivariate-market-regimes-via-clustering-algorithms#:~:text=%E2%9E%A1%20Utilizing%20the%20derived%20statistical,applications%20of%20these%20theoretical%20findings)

. This paper doesn’t use SHAP or macro features, but it’s relevant as a **new regime detection approach** in the 2020s (published in Journal of Computational Finance 2024 as Horváth et al.). It also highlights that regimes can be **persistent** – they discuss adding a “jump penalty” to avoid too-frequent switching, something also advocated in Shu (2024) below.

* **Sparse Jump Models (SJM) for Persistent Regimes:** Shu & Mulvey (2024) – “**Regime-Aware Asset Allocation: a Statistical Jump Model Approach**” – compare Markov switching with a **jump penalization model**​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=This%20article%20investigates%20the%20impact,aware%20asset%20allocation%20strategy)

. The jump model, by design, prefers to stay in the same regime unless evidence strongly suggests a switch (it adds a cost to switching). They use features of price series (returns and multiple volatility measures with different half-lives) as input to the regime model​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=The%20quality%20of%20the%20feature,is%20to%20develop%20a%20feature)

​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=concept%20is%20directly%20relevant%20to,of%20noise%20in%20daily%20index)

. Notably, they find that using longer-memory features (e.g. longer half-life vol) yields more stable regimes​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=match%20at%20L410%20concept%20is,of%20noise%20in%20daily%20index)

​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=match%20at%20L420%20Consequently%2C%20our,lives%20for%20individual%20indices)

, which makes sense (less reactive to noise). Their focus is still on *price-derived features*, but the methodology is driven by **explanatory variables** (volatility at various horizons) rather than hidden states on returns directly. They show that this approach yields regimes that improve asset allocation performance out-of-sample, beating both static allocation and traditional Markov-switching​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=This%20article%20investigates%20the%20impact,aware%20asset%20allocation%20strategy)

​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=analysis%20using%20daily%20return%20series,jump%20models%2C%20offering%20insights%20for)

. Key points: regimes identified with *features that emphasize persistent conditions* (like sustained high vol) are more robust, and tuning the “jump penalty” via cross-validation directly on portfolio performance gave the best results​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=models,US%20equity%20indices%2C%20we%20highlight)

​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=In%20our%20analysis%2C%20we%20focus,in%20both%20industry%20and%20academia)

. This underscores a practical tip: if one is developing SHAP-informed regimes, one might similarly tune how sensitively we declare a new regime (perhaps through the Explanation Stability Index threshold) by optimizing portfolio Sharpe in backtests.

* **Dynamic Factor Allocation with Regime Signals:** Shu & Mulvey didn’t stop there. In a follow-on (2024) paper, “**Dynamic Factor Allocation Leveraging Regime-Switching Signals**”, they applied SJM to **factor indices**​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=This%20article%20explores%20dynamic%20factor,set%20based%20on%20risk%20and)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=Litterman%20model%20to%20construct%20a,resulting%20in%20a%20positive%20Sharpe)

. They identify bull/bear regimes for each style factor (value, momentum, etc.) relative to the market, using a feature set that includes the factor’s recent active returns and some broad market variables​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=Litterman%20model%20to%20construct%20a,resulting%20in%20a%20positive%20Sharpe)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=market%20regimes%20for%20individual%20factors%2C,indices%2C%20with%20an%20equally%20weighted)

. By doing so, they get a timing signal for each factor (e.g. momentum is in a favorable regime or not). Then they feed those into a Black-Litterman optimization to allocate dynamically among the factors​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=and%20growth,interpretability%20compared%20to%20traditional%20methods)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=A%20hypothetical%20single,the%20EW%20benchmark%20itself%2C%20the)

. The result was a multi-factor portfolio that achieved a much higher Information Ratio than an equal-weighted static factor mix​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=ratio%20of%20this%20strategy%20across,metrics%20such%20as%20the%20Sharpe)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=dynamically%20adjust%20allocations%20among%20the,enhance%20factor%20allocation%20by%20capitalizing)

. This is very much in line with the “SHAP-informed factor regime” idea, though they used their SJM model instead of SHAP. The **regimes are driven by factor performance features and market environment features**, not simply by ex-post returns. They emphasize the stability and interpretability of the identified regimes​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=Litterman%20model%20to%20construct%20a,resulting%20in%20a%20positive%20Sharpe)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=market%20regimes%20for%20individual%20factors%2C,indices%2C%20with%20an%20equally%20weighted)

– an advantage we also seek with SHAP (since SHAP can be interpreted in economic terms). This work essentially validates that *dynamic factor weighting based on regime signals can add value*, which supports the case for the methodology in question.

To highlight contrast: a **return-based regime model** (like a pure Markov switching on index returns) might tell us “we are in state 1 vs state 2” but not *why*. An **explanatory feature-driven model** can label regimes with meaning (e.g. “tight monetary policy regime” or “high inflation regime” if one uses macro features, or “momentum-led regime” if one uses SHAP and sees momentum factor dominating). There are also causal discovery approaches (e.g. using Bayesian networks on macro data to find causal regime shifts), but those are less common in portfolio practice.

In comparing literature: Many regime-switching papers in finance pre-2020 used either macro indicators (often yield curve shape, recession dummy) or purely statistical latent variables. The newer trend (as seen above) is **machine learning classification or clustering** for regimes:

* Some use supervised learning to predict regime (binary or multi-class) based on features (as Macrosynergy did, essentially labeling months by subsequent return sign).
* Others use unsupervised learning to cluster periods (Horváth’s Wasserstein k-means, or even simpler k-means on PCA of returns). There’s an Imperial College thesis (2021 by J. Caldwell) that used k-means on PCA of asset returns to define regimes and showed it improved portfolio performance – again focusing on distribution differences rather than cause, but yielding distinct return profiles for regimes​

[imperial.ac.uk](https://www.imperial.ac.uk/media/imperial-college/faculty-of-natural-sciences/department-of-mathematics/math-finance/212236006---James-Mc-Greevy---MCGREEVY_JAMES_01075416.pdf#:~:text=,with%20a%20high%20degree)

.

* The **common theme** is acknowledging that markets go through **regimes** and trying to identify them in a **data-driven way** that is more flexible than classic 2-state HMMs.

Now, the methodology “SHAP-Informed Dynamic Factor Regimes” can be seen as an instance of this trend: we use an ML model’s explanations (SHAP of factors) as the “features” to identify regimes, rather than using returns or simple vol. This is quite novel – I haven’t seen a paper that explicitly does that – but it’s a logical extension of the above. It aligns with the idea that **when the drivers of returns change, the regime has changed**.

**Comparison to Traditional Factor Allocation Frameworks**

It’s helpful to compare this SHAP-informed regime approach with other factor-based allocation frameworks:

| **Framework** | **Drivers of Allocation** | **Regime Awareness?** | **Comments (Methods & Sources)** |
| --- | --- | --- | --- |
| **Static Factor Models** | Historical premia of factors (e.g. long-term averages of value, size returns as in Fama-French)​  [research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=Credit%20Factor%20Investing%20with%20Machine,sample%20performance%20and%20better%20scores)  ​  [research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=excess%20returns,in%20the%20credit%20pricing%20model)  . Often equal-weight or fixed-weight allocations to factors. | **No explicit** – assumes factors earn premia in all conditions (though could rebalance periodically, the model itself doesn’t switch regimes). | Example: Fama-French 5-factor model which is static. Practitioners using smart beta ETFs often stick to static weights (or market-cap weight within each factor index). These approaches can suffer when factor leadership changes (e.g. a static value portfolio underperformed in the 2010s when growth dominated). |
| **Risk Parity / Risk-Based** | Allocations based on inverse risk (volatility or VaR) of assets or factors. Goal is to equalize risk contributions, not to target return directly. | **Indirect** – Risk parity will allocate less to a high-vol regime asset, so it reduces exposure in turmoil, but it doesn’t *identify* regimes; it just continuously adapts to realized risk. | Example: Bridgewater’s All Weather or generic risk parity. If volatility spikes, risk parity shifts weight to more stable assets. However, it might miss return opportunities because it ignores return drivers. Some research attempts **Regime-Switching Risk Parity** (e.g. using Markov regimes to adjust risk targets​  [risk.net](https://www.risk.net/journal-of-operational-risk/7535001/a-regime-switching-factor-model-for-mean-variance-optimization#:~:text=A%20regime,returns%20in%20modern%20financial%20markets)  ), showing improved outcomes. In general, risk-based frameworks focus on stability, whereas SHAP-informed focuses on *explanations*, which could complement each other (one could impose risk parity within each regime). |
| **Machine Learning–Enhanced** | a) **Latent factor models**: use ML (PCA, autoencoders, cluster) to find new factors that better explain covariance or returns. b) **ML forecasts**: use predictive models to forecast returns or macro variables. | **Sometimes** – Latent factor approaches (e.g. autoencoder factors) usually do not explicitly model regimes; they assume a stable structure but more flexible form than linear. Predictive ML can be trained to recognize regimes if provided with regime-indicative features, but the model’s output typically is continuous (returns) rather than discrete regime labels. | Examples: *Gu, Kelly, Xiu (2020)* used tree and network models on firm characteristics to predict stock returns, implicitly capturing non-linear interactions (they found ML models picked up momentum and volatility effects that vary over time). *Iason & Kynigakis (2022)* used autoencoders to build a covariance matrix for portfolio optimization​  wp.lancs.ac.uk  ​  wp.lancs.ac.uk  – this improved minimum variance portfolios, but the factors are latent and not tied to economic regimes. In practice, ML-enhanced strategies often include **macro indicators or regime flags as features** (e.g. include a recession dummy in a return forecast model). If the model is explainable (via SHAP), one could derive regime insight (“the model is currently using the inflation feature heavily – so we are in an inflation-driven regime”). However, without explicit separation, ML models may struggle if regimes shift (hence why adding an explicit regime layer or adaptive component is beneficial). |
| **Regime-Switching Factors** | Identify regimes first (via some method: macro triggers, clustering, HMM) and then apply different factor weights or models in each regime. | **Yes (by design)** – The core of this approach is to acknowledge regimes. Different states have different expected factor returns or optimal weights. | Examples: *Ang & Bekaert (2004)* did an early stock/bond asset allocation with regime switching (bull vs bear equity regimes affecting bond yields). *Jakub Polec (2024)* illustrates using HMM to adjust factor exposures (e.g. momentum factor weight reduced in volatile regimes)​  [medium.com](https://medium.com/@jpolec_72972/factor-investing-with-hidden-markov-models-hmms-786c30cb3706#:~:text=adjustments%2C%20which%20can%20potentially%20optimise,our%20returns)  ​  [medium.com](https://medium.com/@jpolec_72972/factor-investing-with-hidden-markov-models-hmms-786c30cb3706#:~:text=For%20example%2C%20momentum%20factors%20might,and%20resilience%20of%20our%20portfolio)  . More recently, *Shu & Mulvey (2024)* on dynamic factor allocation uses regime signals for each factor (bull or bear) before allocation​  [arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=and%20growth,interpretability%20compared%20to%20traditional%20methods)  ​  [arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=A%20hypothetical%20single,the%20EW%20benchmark%20itself%2C%20the)  . Also, the *Horváth (2024) Wasserstein k-means* regime clustering could be applied: once regimes of returns are identified, one could measure factor performance in each regime and allocate accordingly (similar to how one might invest differently in “crisis” vs “calm” regimes). The SHAP-informed regime method falls here: it determines regimes by changing factor importance and then reallocates factors or assets. The difference is it defines regimes by **explanatory variables** (which is somewhat between unsupervised and supervised – the model’s explanations guide it). |

In essence, **SHAP-Informed Dynamic Factor Regimes** can be seen as an advanced form of *Regime-Switching Factors* with elements of *ML-enhanced* strategy. It stands out in that it uses the **explanation of an ML model as the regime signal**. Traditional frameworks either ignore regimes or use simpler regime identifiers. This method leverages rich information from a model that could incorporate dozens of features. It is particularly apt for the modern market where drivers rotate quickly (e.g. pandemic regime: health data mattered, then inflation regime: macro mattered, etc.). By continuously monitoring SHAP values, the approach can potentially catch regime shifts faster than waiting for a series of bad returns to infer “oops, momentum stopped working.”

To compare outcomes: A static factor portfolio (like a constant mix of value, momentum, quality) will lag when the market regime favors one factor over others for an extended period. A SHAP-informed dynamic strategy aims to overweight the factor(s) currently most **causally relevant**. This is somewhat analogous to **adaptive risk premia** strategies that some asset managers (e.g. Goldman Sachs ABS indices) employ – those often use momentum of factor *returns*, whereas here we use momentum of factor *importance*. There isn’t a long track record published since it’s a cutting-edge idea, but we can hypothesize: it would have shifted away from value in 2020 when growth/tech became dominant (SHAP of tech earnings growth or low rates would spike), then perhaps back toward value in late 2021 when inflation factors surged. Other frameworks might not respond until actual returns of value improved or volatility regimes were clearly flagged.

One more comparison: **Causal vs Correlative Regimes**. SHAP is closer to *causal drivers* (to the extent the model captures causal features) than pure return correlation-based regimes. If, say, a new factor (like retail trading activity) becomes important, SHAP can flag it even if returns haven’t yet manifested a separate regime in mean/vol. This could give a forward-looking edge. It aligns with the idea of **“early warning”**: indeed, a recent iScience paper (2023) integrated spillover networks and ML to get early warnings of regime shifts​

[sciencedirect.com](https://www.sciencedirect.com/science/article/pii/S2589004225001841#:~:text=,and%20a%20machine%20learning%20model)

. They found that certain network metrics spiked before a regime switch. Similarly, a jump in SHAP for a macro factor (like oil prices) could precede a market regime shift to “inflationary”. This is speculative but grounded in the notion that models picking up new signals can be a canary in the coal mine.

In conclusion, the SHAP-informed regime methodology is highly synergistic with current trends: it uses **XAI to illuminate ML models**, and uses that illumination to perform **dynamic allocation**, a bit like driving with high-beam headlights – seeing not just the road (returns) but what’s around the corners (the factors causing those returns). It compares favorably with static or purely return-driven approaches in that it can both *explain* and *adapt*, two qualities essential in today’s complex market environment. The key challenge (noted in literature as well) is **robustness** – one must ensure the model itself is reliable, to trust its SHAP signals. That’s why techniques to improve stability (like the jump models, or ensemble averaging of explanations) are important. If done carefully, this methodology could provide a more nuanced and effective framework for portfolio optimization, as hinted by the growing number of papers combining regime analysis with factor investing​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=This%20article%20explores%20dynamic%20factor,set%20based%20on%20risk%20and)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=A%20hypothetical%20single,the%20EW%20benchmark%20itself%2C%20the)

and the positive results they are finding.

**Sources:**

* Cherief *et al.*, *Credit Factor Investing with ML*, Amundi WP 2022 – used SHAP to quantify factor contributions by period​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=The%20goal%20of%20this%20section,With)

​

[research-center.amundi.com](https://research-center.amundi.com/files/nuxeo/dl/688f72c7-88b6-47d9-9599-63fb8308d062?inline=#:~:text=the%20returns%20of%20the%20variables,returns%20are%20positively%20correlated%20with)

.

* de Wit *et al.*, *Forecasting Stock Crash Risk with ML*, Robeco 2022 – used SHAP to explain crash risk model​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=year%20in%20our%20sample%20period,This)

​

[robeco.com](https://www.robeco.com/files/docm/docu-202206-forecasting-stock-crash-risk-with-machine-learning-hksg.pdf#:~:text=Figure%207%20,Values%20to%20the%20right%20indicate)

.

* Unigestion 2020, *Machine Learning for Fund Selection* – SHAP for feature importance in fund performance forecasts​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=to%20gauge%20which%20input%20factors,explanatory.%20The%20qualitative%20features)

​

[unigestion.com](https://www.unigestion.com/wp-content/uploads/2020/12/20201008-White-Paper-Machine-Learning-FINAL.pdf#:~:text=Sortino%20ratio%2C%20alpha%20and%20t,for%20performance%20forecasting%2C%20under%20the)

.

* Moehle, Boyd, Ang 2021 – *Portfolio Performance Attribution via Shapley Value*​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=We%20consider%20an%20investment%20process,1%20Introduction)

​

[arxiv.org](https://arxiv.org/pdf/2102.05799#:~:text=be%20active%20,more%20than%20just%20three%20features)

.

* Greengard *et al.* 2020 – *Factor Clustering with t-SNE* (Yale University) – t-SNE clustered 6 factor regimes resembling known styles​

[economics.yale.edu](https://economics.yale.edu/sites/default/files/2022-10/tSNE_draft15.pdf#:~:text=%28t,by%20their%20corresponding%20first%20principal)

.

* Horváth, Issa, Muguruza 2024 – *Wasserstein k-means for Market Regimes*, J. Computational Finance – clustering regimes via distribution distance​

[risk.net](https://www.risk.net/journal-of-computational-finance/7959937/clustering-market-regimes-using-the-wasserstein-distance#:~:text=,discrepancy%20approach%2C%20validating%20qualitative%20results)

​

[risk.net](https://www.risk.net/journal-of-computational-finance/7959937/clustering-market-regimes-using-the-wasserstein-distance#:~:text=The%20problem%20of%20rapid%20and,Wasserstein)

.

* McGreevy *et al.* 2024 – *Detecting Multivariate Market Regimes via Clustering*, SSRN – k-means with Wasserstein/MMD distances, two-step clustering​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=portfolio%20design,new%20approach%20can%20be%20used)

​

[papers.ssrn.com](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4758243#:~:text=method%20for%20multidimensional%20data,variance%20and%20correlation%20between%20two)

.

* Gholkar & Sueppel 2025 – *Classifying Credit Markets with Macro Factors*, Macrosynergy – ML classification of regimes using macro features​

[macrosynergy.com](https://macrosynergy.com/research/classifying-credit-markets-with-macro-factors/#:~:text=performance%20and%20create%20a%20chance,of%20returns%20and%20produce%20good)

​

[macrosynergy.com](https://macrosynergy.com/research/classifying-credit-markets-with-macro-factors/#:~:text=statistical%20learning%20processes%20that%20sequentially,power%20and%20economic%20value%20generation)

.

* Shu & Mulvey 2024 – *Regime-Aware Asset Allocation via Jump Model* – feature-based regimes with penalized switching​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=The%20quality%20of%20the%20feature,is%20to%20develop%20a%20feature)

​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=concept%20is%20directly%20relevant%20to,of%20noise%20in%20daily%20index)

; showed jump model outperforms Markov for asset alloc.​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=This%20article%20investigates%20the%20impact,aware%20asset%20allocation%20strategy)

​

[arxiv.org](https://arxiv.org/html/2402.05272v1#:~:text=analysis%20using%20daily%20return%20series,jump%20models%2C%20offering%20insights%20for)

.

* Shu & Mulvey 2024 – *Dynamic Factor Allocation with Regime Signals* – identified factor regimes with features and improved multi-factor portfolio IR​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=Litterman%20model%20to%20construct%20a,resulting%20in%20a%20positive%20Sharpe)

​

[arxiv.org](https://arxiv.org/html/2410.14841v1#:~:text=ratio%20of%20this%20strategy%20across,metrics%20such%20as%20the%20Sharpe)

.

* Jakub Polec 2024 (QuantJourney) – *Factor Investing with HMMs* (Medium blog) – discusses adjusting factor exposures in different hidden regimes​

[medium.com](https://medium.com/@jpolec_72972/factor-investing-with-hidden-markov-models-hmms-786c30cb3706#:~:text=adjustments%2C%20which%20can%20potentially%20optimise,our%20returns)

​

[medium.com](https://medium.com/@jpolec_72972/factor-investing-with-hidden-markov-models-hmms-786c30cb3706#:~:text=For%20example%2C%20momentum%20factors%20might,and%20resilience%20of%20our%20portfolio)

.

* Fan *et al.* 2021 – *A Reexamination of Factor Momentum* – found a few factors dominate momentum gains​

[quantpedia.com](https://quantpedia.com/a-deeper-look-into-factor-momentum/#:~:text=momentum%20effect,weighted%20portfolios)

(relevant to factor dominance concept).

* Xiaoguang Li 2023 – *Using DTW to Cluster Stocks* (Medium) – demo of tslearn DTW clustering and pattern recognition in stock time-series​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=match%20at%20L152%20DTW%20calculates,for%20time%20shifting%2C%20stretching%2C%20and)

​

[lixiaoguang.medium.com](https://lixiaoguang.medium.com/using-dynamic-time-warping-dtw-to-cluster-stocks-a2e50ad43480#:~:text=match%20at%20L178%20model%20%3D,max_iter%3D200%29%20model.fit_predict%28data%29%20ssd.append%28model.inertia)

.

* MOSEK Portfolio Optimization Cookbook – Sec.5 Factor Models – explains factor model benefits for portfolio optimization​

[docs.mosek.com](https://docs.mosek.com/portfolio-cookbook/factormodels.html#:~:text=The%20purpose%20of%20factor%20models,6)

​

[docs.mosek.com](https://docs.mosek.com/portfolio-cookbook/factormodels.html#:~:text=Let%20%5C%28Z_t%5C%29%20be%20an%20%5C%28N%5C%29,linear%20function%20of%20components%20as)

.

* CVXPortfolio Documentation – Python library for portfolio optimization with custom objectives​

[github.com](https://github.com/cvxgrp/cvxportfolio#:~:text=cvxgrp%2Fcvxportfolio%3A%20Portfolio%20optimization%20and%20back,described%20in%20the%20accompanying%20paper)

.