Viet Nguyen
vin041@stanford.edu

Timing the Fama-French HML Factor

Abstract:

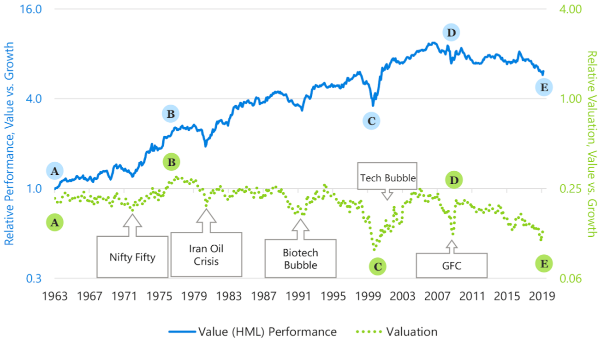
I demonstrate that profitability, R&D, SGA, and potentially valueSpread x log b/p are features which can reduce a model's correlation to HML. Furthermore, I show that FFNNs using the same predictors can decrease exposure to HML relative to linear models by generating additional alpha: an equal weighted long-short decile spread achieves a Sharpe of 3.23 compared to the linear model's 1.66. On the other hand, LSTM models seem to perform well until the start of the value drawdown, perhaps signalling that a regime shift in the market has indeed occurred.

Introduction:

In 1992, Fama and French built an asset pricing model that explains the contemporaneous returns of individual stocks, achieving an R^2 of ~90%. This model was founded on the empirical observation that stocks with high book-to-market ratios (B/P) tend to outperform stocks with low B/P ratios (the value effect), and that small-cap stocks tend to outperform large-cap stocks (the size effect).

Over the past 12 years, the HML factor, a portfolio that is long the stocks with the highest 30% of B/P ratios (growth) and short the lowest 30% of B/P ratios (value)[1] , suffered a prolonged drawdown: value stocks are unexpectedly underperforming, bringing HML into the spotlight. In 2018, Sheth and Lim showed that HML overperforms during and shortly after recessions, though using yield curve inversions to time the factor was ineffective. In 2019, Asness used the "value spread" to suggest that HML's recent underperformance was irrational, suggesting that investors should overweight portfolios towards value. In 2020, Arnott et al. addressed narratives announcing the "death of value". The goal of this project is to use features inspired by recent HML research in combination with deep learning in an attempt to time the HML Factor.

Figure 3. Value vs. Growth Performance and Relative Valuations, United States, Jul 1963–Sep 2019



Source: Research Affiliates, LLC, using data from CRSP/Compustat.

Value has underperformed for the last 12 years (Arnott et al., 2020)

I will use 2 different approaches to "time" the HML factor. In the first, following Gu, Kelly, and Xiu (2018), I will use a 3 layer feedforward neural network (FFNN) to predict future one month returns in excess of 3 month Treasury Bills for individual stocks. I will compare the metrics of this method to corresponding

[1] For their HML portfolio, Fama and French further categorize by size (smallest 50% vs largest 50%). HML return is calculated by .5(small value return + large value return) -.5(small growth return + large growth return), where each of the 4 portfolios is weighted by size.

linear models. To judge the effectiveness of each model at "timing" the HML factor, I will analyze the returns of each model's simulated trading strategy, and measure their correlation and exposure to HML.

For the second approach, I will use a single-layered LSTM that takes in features corresponding to the past 12 months and predicts next month's HML return.

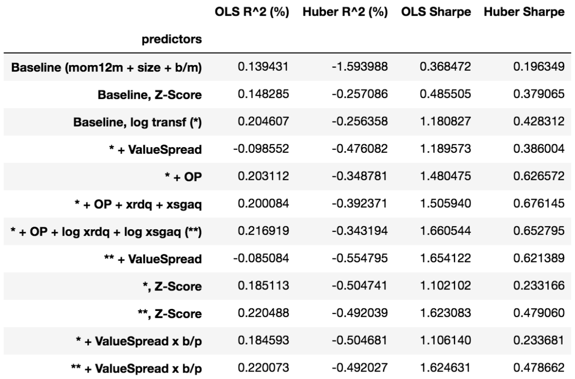
Dataset:

I obtain monthly price and quarterly fundamental data for individual stocks via the CRSP/Compustat merged database from WRDS. In addition, I obtain 3 month treasury data from FRED and Fama-French factor data from Ken French's website.

The processed dataset includes roughly 1.6 Million stock/month pairs from January 1982 to December 2019, with an average of 3000 stocks per time period. The test set spans the years 2000 - 2020, consisting of roughly 60% of the dataset.

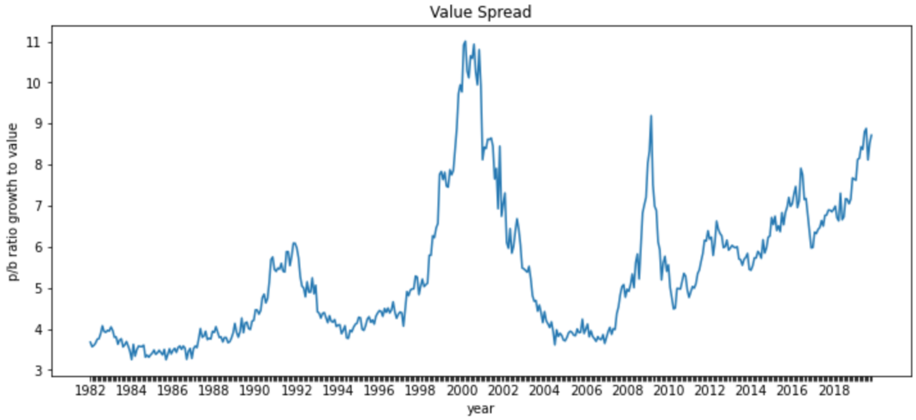
Features:

I train two FFNN models, using 6 or 7 lagged input features. I start with the features used by a linear model inspired by Lewellen (2014), which uses 3 predictors: log B/P, log size, and historical 2-12 month return. I add operating profitability as a feature based on Fama and French's (2015) observation that linear regression models using profitability can subsume models using B/P. I add 2 more features based on Arnott et al. (2020): log R&D, and log SGA (selling, general, and administrative), which were used to strengthen the HML factor. The goal of adding these additional features is to use possible confounding variables to reduce the model's reliance on HML as a predictor, removing the correlation of its returns to the HML drawdown. Finally, I test a 7th predictor, which multiplies the valueSpread by log B/P in an attempt to learn the current state of HML. Empirically, using the raw valueSpread feature fails.



	OLS R^2 (%)	Huber R^2 (%)	OLS Sharpe	Huber Sharpe
predictors				
Baseline (mom12m + size + b/m)	0.139431	-1.593988	0.368472	0.196349
Baseline, Z-Score	0.148285	-0.257086	0.485505	0.379065
Baseline, log transf (*)	0.204607	-0.256358	1.180827	0.428312
*** + ValueSpread**	-0.098552	-0.476082	1.189573	0.386004
*** + OP**	0.203112	-0.348781	1.480475	0.626572
*** + OP + xrdq + xsgaq**	0.200084	-0.392371	1.505940	0.676145
*** + OP + log xrdq + log xsgaq (**)**	0.216919	-0.343194	1.660544	0.652795
**** + ValueSpread**	-0.085084	-0.554795	1.654122	0.621389
***, Z-Score**	0.185113	-0.504741	1.102102	0.233166
****, Z-Score**	0.220488	-0.492039	1.623083	0.479060
*** + ValueSpread x b/p**	0.184593	-0.504681	1.106140	0.233681
**** + ValueSpread x b/p**	0.220073	-0.492027	1.624631	0.478662

Linear regression results of various features and transformations



Value Spread, a measure of the ratio of the p/b of growth to value stocks. Asness proposes that a high value spread could forecast larger future HML returns.

For the LSTM model, I use features described by Arnott et al. (2020), who decompose HML's returns into 3 components.

$$\log(1+r_{t^+}) =$$

$$= \underbrace{\underbrace{\left[\log\left(\frac{P_{t^+}}{B_{t^+}}\right) - \log\left(\frac{P_{t-1}}{B_{t-1}}\right)\right]}_{\boxed{\text{Revaluation}}}}_{\boxed{\text{Revaluation Component}}} + \underbrace{\underbrace{\left[\log\left(\frac{B_{t^-}}{B_{t-1}}\right) + \log\left(1+\frac{D_{t^-}}{P_{t^-}}\right)\right]}_{\boxed{\text{Profitability}}} + \underbrace{\left[\log\left(\frac{P_{t^-}}{B_{t^-}}\right) - \log\left(\frac{P_{t^+}}{B_{t^+}}\right)\right]}_{\boxed{\text{Migration}}}}_{\boxed{\text{Structural Component}}}$$

Accounting identity for decomposing HML returns (Arnott et al. 2020)

Methodology:

For the FFNN models, I use a methodology inspired by Gu, Kelly, and Xiu (2018). I use a FFNN with 3 layers, with 32, 16, and 8 hidden nodes per layer, following the geometric pyramid rule. I transform the 6 or 7 input features by taking the cross-sectional Z-Score of each input feature per time period. I use ReLU activation for each hidden layer, and batch transform after each activation, with no affine transformation. I use He initialization for the weights (He et al. 2015). I use ADAM optimization for gradient descent (Kingma et al. 2014).

For the sample-splitting scheme, I start with a train set from 1982 - 1998, a validation set of 1999, and a test set of 2000. Each year, I train a new model, increasing the size of the train set by 1 and pushing back the validation and test sets. The goal of this process is to simulate an actual trading strategy, while preventing look-ahead bias. All models are trained using this rolling structure.

I use the following hyperparameters: batch size = 10000, lr ∊ [.01, .001], and L1 ∊ [0, 1e-3, 1e-4, 1e-5], along with early stopping with patience = 5. Rather than choosing hyperparameters from the best score on the validation set, I use the best modal hyperparameters: the hyperparameter set that is optimal for the most validation years. I add 5 validation years with no test predictions to stabilize this process. The goal of using the modal hyperparameter set is to decrease the dependency of the selection on the specific validation year. For example, if the hyperparameter set [lr = .01, L1 = 0] is only optimal for the validation year 2007, whereas [lr = .01, L1 = 1e-3] was optimal for validation years 1994 - 2006, I have reason to believe that choosing the former set would overfit to the specific validation year.

The LSTM model includes one LSTM layer with sequence length 12 and 4 hidden nodes, with a linear layer connecting the last 4 hidden variables to the output. The rest of the details are identical to those used for the FFNN models.

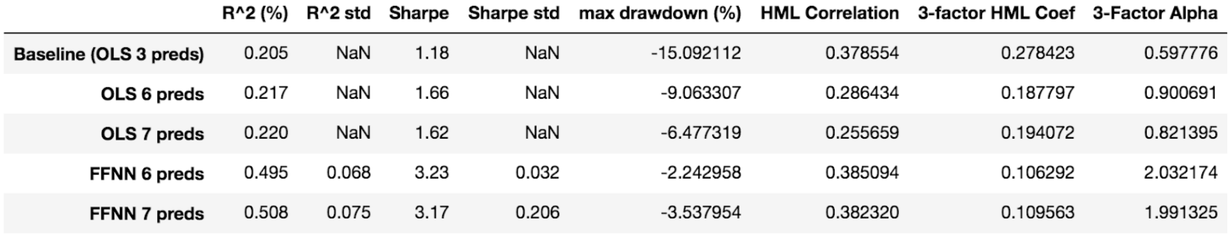
I will now discuss my choice of metrics. For the FFNN models, I measure de-mean R^2 rather than typical R^2: I compare the residuals of the models' predictions to the performance of predicting a return of 0 for each stock. Using the standard R^2 measure would understate the model's performance: if we had a model that predicted the exact average of future stock returns every time, we could easily time the market. In addition, as explained by Gu, Kelly, and Xiu (2018), making predictions based on historical averages tends to underperform relative to predicting an excess return of 0.

I also measure the annualized Sharpe ratio of a simulated strategy, the average excess monthly return over its standard deviation. I multiply this number by sqrt(12) to annualize it. This strategy buys the top decile of the model's predicted returns, and sells the bottom decile in equal weights[2].
For the LSTM model, I measure demeaned R^2, as well as R^2 values which predict the latest HML return and the rolling 12 month HML average.

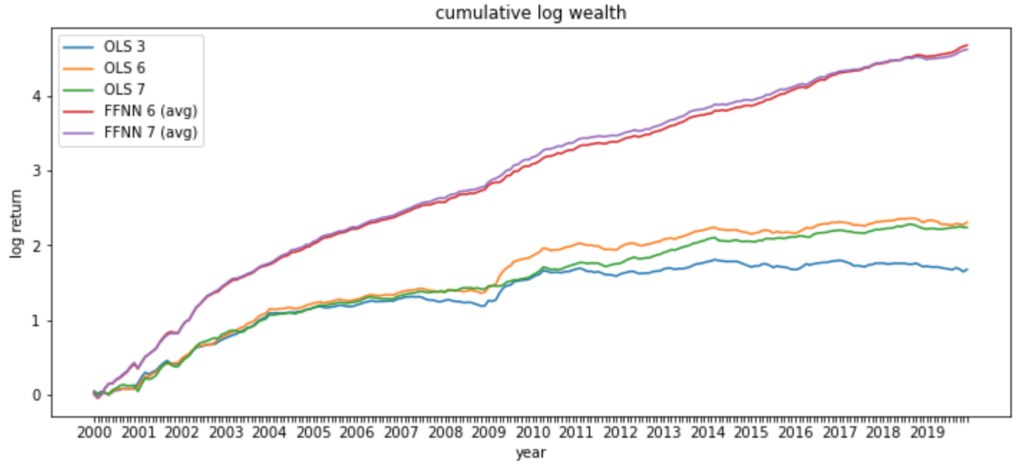
To account for model variance, I run each FFNN and LSTM model 5 times and report the mean and variance of the metrics.

FFNN Results:

I report various statistics of 5 models below. These models include the aforementioned baseline model from Lewellen (2014), and OLS/FFNN models with 6 or 7 input features.



	R^2 (%)	R^2 std	Sharpe	Sharpe std	max drawdown (%)	HML Correlation	3-factor HML Coef	3-Factor Alpha
Baseline (OLS 3 preds)	0.205	NaN	1.18	NaN	-15.092112	0.378554	0.278423	0.597776
OLS 6 preds	0.217	NaN	1.66	NaN	-9.063307	0.286434	0.187797	0.900691
OLS 7 preds	0.220	NaN	1.62	NaN	-6.477319	0.255659	0.194072	0.821395
FFNN 6 preds	0.495	0.068	3.23	0.032	-2.242958	0.385094	0.106292	2.032174
FFNN 7 preds	0.508	0.075	3.17	0.206	-3.537954	0.382320	0.109563	1.991325



One statistic that I use to measure HML exposure is calculated as follows: I use the Fama French 3 factor regression model (1992) to decompose each model's returns into a function of market return, HML return, and SMB return(size exposure). This decomposition has an R^2 of roughly 90% on a variety of test

[2] For comparative purposes, the Sharpe ratio of buying and holding the S&P 500 is 0.51. Gu, Kelly, and Xiu (2018)'s best FFNN model achieves an out-of-sample R^2 of .40% and a Sharpe of 2.45.

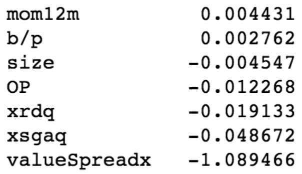
portfolios. The coefficient (b_v in the following equation) measures how much of a portfolio's return is due to market, value, and size exposure, as well as how much of the return is unexplained by the decomposition (alpha). I perform this decomposition on each model's yearly returns.

$$r = R_f + \beta(R_m - R_f) + b_s \cdot SMB + b_v \cdot HML + \alpha$$

Fama French 3 factor model

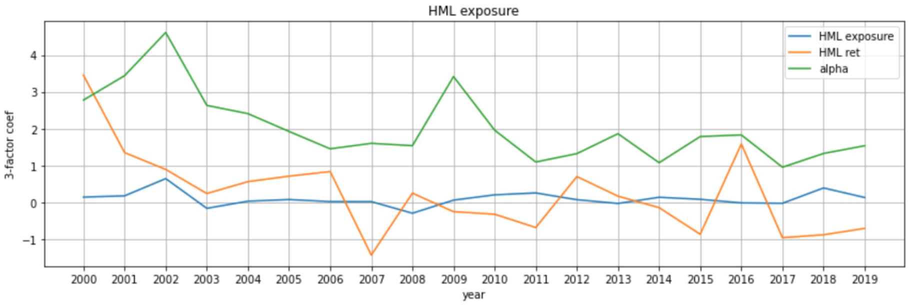
As expected, the 3 features added to the baseline OLS model decrease correlation with HML, as does the valueSpread times log b/p predictor. This suggests that these features are indeed confounding variables.

Comparing the two FFNN models, it seems like the last input feature only adds variance to the model. This seems to contradict the FFNN 7 ablation results(which measures the R^2 after zeroing out a predictor), which identify valueSpread x log b/p as the most important feature. More samples need to be collected to distinguish these 2 models.



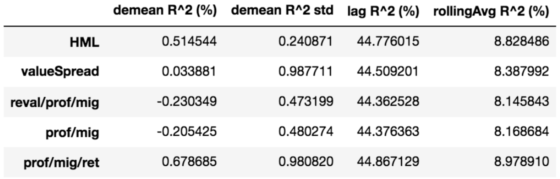
```
mom12m          0.004431
b/p             0.002762
size           -0.004547
OP             -0.012268
xrdq           -0.019133
xsgaq          -0.048672
valueSpreadx   -1.089466
```

Interestingly, the FFNN models have a higher correlation with HML than the linear models, but are less exposed to the factor. Below is a graph of FFNN 6's value exposure and 3-factor alpha over time, overlaid with HML's yearly return. The gap between the HML correlation and HML exposure seem to imply that the FFNN model generates more of its return from non-HML related profits (alpha), but the part of its returns that are related to HML are more heavily correlated than the linear models.

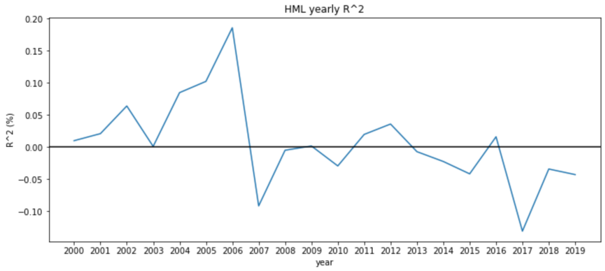


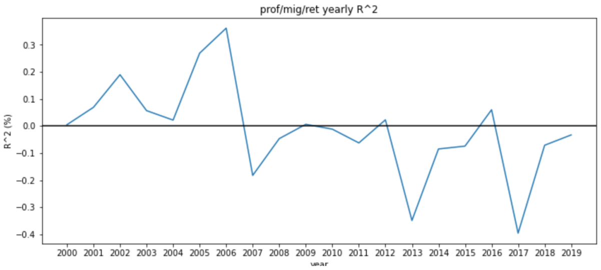
LSTM Results:

I also report various statistics for the LSTM model. Unfortunately, only one model does not contain 0 in its 95% CI. In addition, this model performs poorly over the last 12 years:



	demean R^2 (%)	demean R^2 std	lag R^2 (%)	rollingAvg R^2 (%)
HML	0.514544	0.240871	44.776015	8.828486
valueSpread	0.033881	0.987711	44.509201	8.387992
reval/prof/mig	-0.230349	0.473199	44.362528	8.145843
prof/mig	-0.205425	0.480274	44.376363	8.168684
prof/mig/ret	0.678685	0.980820	44.867129	8.978910





Conclusion:

These results suggest that profitability, R&D, SGA are features which can reduce a model's correlation to HML. Furthermore, I demonstrate the potential of FFNNs to substantially improve out of sample R^2 and Sharpe ratios relative to linear models, as well as their ability to decrease exposure to HML. Further research is required to determine the effectiveness of valueSpread x b/p as a predictor.

The failure of the LSTM model to explain the last 12 years of value drawdown is likely because the past 12 years of HML returns have deviated from any typical pattern. Rather than trying to time HML directly, one could take the indirect approach of reducing a strategy's correlation and/or exposure to the factor. That said, maybe a more stateful RNN approach that looks at more than 12 months of historical predictors could improve the model.

Viet Nguyen
vin041@stanford.edu

Citations:

- Fama, Eugene F., and Kenneth R. French. "The cross−section of expected stock returns." *the Journal of Finance* 47.2 (1992): 427-465.
- Lewellen, Jonathan. "The cross section of expected stock returns." *Forthcoming in Critical Finance Review* (2014).
- Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).
- Fama, Eugene F., and Kenneth R. French. "A five-factor asset pricing model." *Journal of financial economics* 116.1 (2015): 1-22.
- He, Kaiming, et al. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification." *Proceedings of the IEEE international conference on computer vision*. 2015.
- Gu, Shihao, Bryan Kelly, and Dacheng Xiu. *Empirical asset pricing via machine learning*. No. w25398. National Bureau of Economic Research, 2018.
- Asness, Cliff. “It's Time for a Venial Value-Timing Sin.” *AQR Capital Management*, 7 Nov. 2019, www.aqr.com/Insights/Perspectives/Its-Time-for-a-Venial-Value-Timing-Sin.
- Arnott, Robert D., et al. "Reports of Value’s Death May Be Greatly Exaggerated." *Available at SSRN 3488748* (2020).

Data:

- “CRSP/Compustat Merged Database.” *CRSP*, www.crsp.org/products/research-products/crspcompustat-merged-database.
- “3-Month Treasury Constant Maturity Rate.” *FRED*, fred.stlouisfed.org/series/DGS3MO.
- French, Kenneth. *Kenneth R. French - Data Library*, mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html.

Libraries:

- Hunter, John D. "Matplotlib: A 2D graphics environment." *Computing in science & engineering* 9.3 (2007): 90-95.
- McKinney, Wes. "Data structures for statistical computing in python." *Proceedings of the 9th Python in Science Conference*. Vol. 445. 2010.
- Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." *the Journal of machine Learning research* 12 (2011): 2825-2830.
- Paszke, Adam, et al. "Automatic differentiation in pytorch." (2017).
- Virtanen, Pauli, et al. "SciPy 1.0: fundamental algorithms for scientific computing in Python." *Nature methods* 17.3 (2020): 261-272.
- Harris, Charles R., et al. "Array programming with NumPy." *Nature* 585.7825 (2020): 357-362.