Advances in Financial Machine Learning: Numerai's Tournament

Marcos López de Prado

Advances in Financial Machine Learning
ORIE 5256

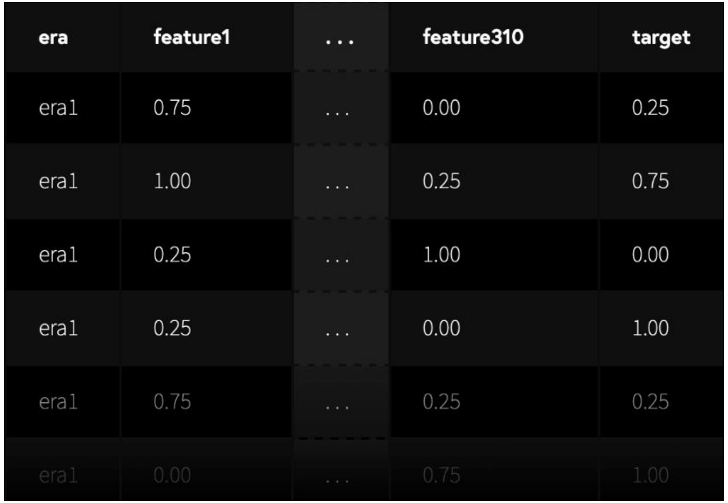
Key Points

- In this course, we have learned how to turn financial data into insight. It is time to put what we have learned into practice.
- Using Numerai's dataset, we will build a ML-based investment algorithm.
- The goal of this project is:
  - Learn how to solve a well-defined financial forecasting problem, using anonymized data
  - Become contestants in Numerai's tournaments
- Deliverables:
  - Paper: Write a description of your forecasting algorithm
  - Submission: Submit your forecasts to Numerai's tournament

How the Tournament Works

Data Structure

- Investment universe of 5,000 names across countries, sectors and sizes
- Cross-sectional real-valued **X** matrix
  - one matrix per month (monthly Eras)
  - approx. size of $(5000x310)$
  - values are adjusted for country-sector-size biases
- Categorical **y** array
  - one array per month
  - labels follow the fixed-horizon method, where the horizon is one month (no time-overlap across Eras)
  - {0,0.25,0.5,0.75,1}, indicating outperformance / neutral / underperformance against peers (targets adjusted for biases)
- New features are added from time to time



era	feature1	...	feature310	target
era1	0.75	...	0.00	0.25
era1	1.00	...	0.25	0.75
era1	0.25	...	1.00	0.00
era1	0.25	...	0.00	1.00
era1	0.75	...	0.25	0.25
era1	0.00	...	0.75	1.00

Numerai data sample (https://numer.ai)

Data Obfuscation

A common problem in tournaments and backtesting platforms is that users may incorporate information from the *test* set into the *train* set.

To prevent guided searches, (X, y) are obfuscated:

- A random distance-preserving transformation is applied to all X values
- Rows are shuffled within each Era
- Row names are encrypted
- Column names are encrypted

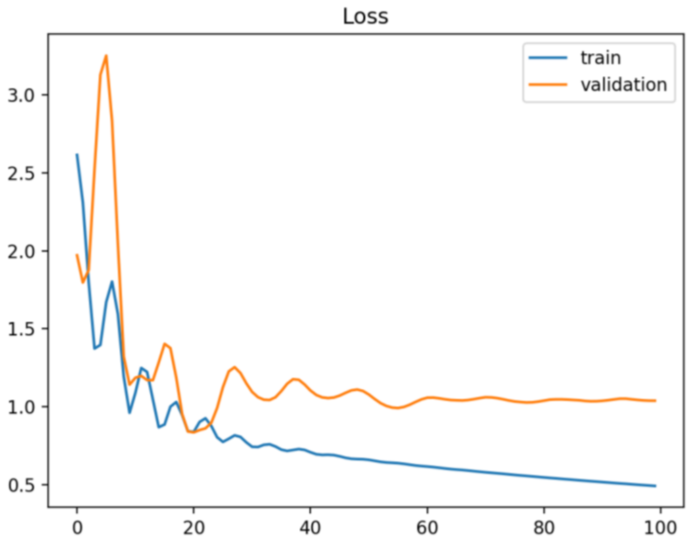
Obfuscation prevents users from applying their own biases to forecasting. They can only model the problem posed, using only the data provided.

Model Training

Data splits:

- Train set: 120 months (eras)
  - y-array is shared
  - useful for K-fold cross-validation
- Validation set: 12 months
  - y-array is shared
  - useful for hyper-parameter tuning
- Test set: 63 months
  - y-array is not shared. Users must forecast it
- Live set: recent data for live trading

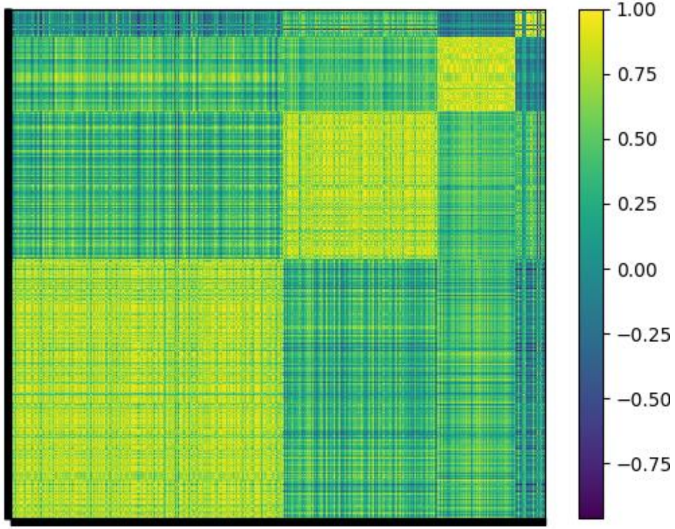
No p-hacking: Selection bias by the users is prevented with the obfuscation of the data, and by hiding the test-set's y array.



To avoid train-set overfitting, researchers must monitor at what point the forecast error *on the validation set* ceases to fall

Model Scoring & Selection

- There is one tournament per week
- Among the contributed models for a given week, the best-performing ones are selected based on a combination of factors:
  - The correlation of the forests on the test set with the actual targets: the ability to discriminate between outperforming and underperforming stocks
  - Novelty: models with uncorrelated forecasts are preferred
- In addition to preventing users' selection bias, Numerai also corrects for its own selection bias



A clustering algorithm can be used to identify subsets of similar / redundant models, and enforce diversification

User Staking

- Users are incentivized to contribute good models, not to overfit
- Users put skin in the game by staking cryptocurrency on their live performance
- A user’s reward is a combination of
    - Numerai’s payout for that week
    - Performance of contributed signals over the next month on the live data
    - Amount staked by the user
    - Long-term consistency
    - Confidence in the model expressed by the user in an auction mechanism
    - There are no rewards based on backtests or the test set.

Sections in Project Paper

1. Feature Engineering

Using the train set only, engineer the features that exhibit useful properties.

- Stationarity:
  - We can only learn from stationary features
  - Apply stationarity tests, and transform data if needed
- Low mutual information
  - Avoid substitution effects that prevent the correct selection of features, cause overfitting, and increase the correlation between predictors
- Dimensionality reduction
  - Avoid curse of dimensionality
  - Reinforce the signal by partially cancelling the noise
    - Clustering: Agglomerative, partitioning. The distance matrix could be based on correlations, normalized mutual information, etc.
    - Change of basis: PCA, Kernel-PCA, UMAP, etc.

2. Feature Selection

Using the train set only, apply an MDA analysis.

- Baseline classifier: Random forest (all default arguments, except `max_features`)
  - Masking effects: Make sure you set argument `max_features=int(1)`. In this way, only one random feature is considered per level
- K-fold cross-validation: Given the number of engineered features, choose an appropriate number of folds, *K*
- Score: Compute AUC-ROC on multilabel classification (macro or micro)
  - Even if the consensus forecast is weak, the probability may still be informative
  - Alternatively, you could use as score Fisher's transform on Spearman's correlation
- Criterion: Choose all features with positive mean Score improvement

As a starting point, you may want to adapt the code in chapter 8 of AFML.

3. Modelling (1/2)

Using the train set only, choose the appropriate approximating function.

The objective is to beat the baseline classifier (Random Forest).

- Hyper-parameter tuning
- Some options include
  - Modified Random Forest
  - Logistic regression
  - Naïve Bayes
  - AdaBoost
  - GradientBoost
  - SVC

3. Modelling (2/2)

Some advice when developing models:

- **Balance across Eras**: Find Eras where model performs poorly, exclude them from model, and develop a model specifically for those (a kind of boosting). The objective function should evaluate whether the combined model does well across all Eras. Potentially, also balance performance across countries, sectors, sizes, …
- **Balance across Features**: Avoid models that rely heavily on a few features. If those features cease to work, the model will perform poorly
- **Balance targets**: Avoid models that perform well on one level but not on other levels. A robust model relies on features that are informative for different levels
- **Variance reduction**: When bagging, form small bags while controlling that draws come from different Eras. That reduces the correlation across bags, hence reducing the variance of the error more effectively

4. Cross-Validation

Compute score on the validation set (point estimates).

Alternatively, apply a *K-fold* cross-validation on the validation set.

- For every fold forecasted, fit the function on the complementary dataset
  - Full train set
  - K-1 folds in validation set
- For example:
  - For predicting fold 1, fit the model on: train set + folds 2,…,K
  - For predicting fold 2, fit the model on: train set + folds 1, 3, …, K
  - …
  - For predicting fold K, fit the model on: train set + folds 1, 2, …, K-1
- This allows us to compute the mean and standard deviation on the scores

5. Submission

Create an account in Numerai. Report account name in the paper.

Compute forecasts on the test set.

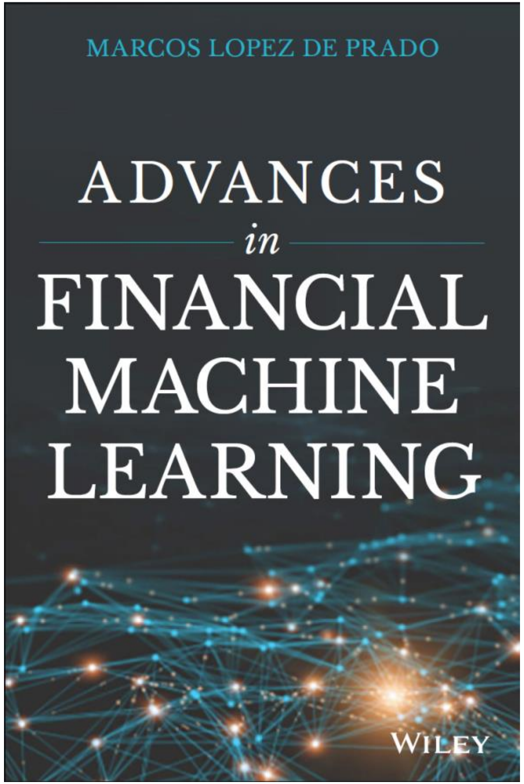
Multi–class forecast probabilities must be condensed into a single score

- For example, $[0,.1,.2,.7,0.] \rightarrow .6+.7x(.8-.6)=.74$

Upload your forecasts.

Send me paper + source code.

For Additional Details



The first wave of quantitative innovation in finance was led by Markowitz optimization. Machine Learning is the second wave and it will touch every aspect of finance. López de Prado's Advances in Financial Machine Learning is essential for readers who want to be ahead of the technology rather than being replaced by it.
— Prof. **Campbell Harvey**, Duke University. Former President of the American Finance Association.

Financial problems require very distinct machine learning solutions. Dr. López de Prado's book is the first one to characterize what makes standard machine learning tools fail when applied to the field of finance, and the first one to provide practical solutions to unique challenges faced by asset managers. Everyone who wants to understand the future of finance should read this book.
— Prof. **Frank Fabozzi**, EDHEC Business School. Editor of The Journal of Portfolio Management.

THANKS FOR YOUR ATTENTION!

Disclaimer

- The views expressed in this document are the authors’ and do not necessarily reflect those of the organizations he is affiliated with.
- No investment decision or particular course of action is recommended by this presentation.
-

The Past and Future of Quantitative Research

Marcos López de Prado

Professor of Practice, Operations Research & Information Engineering

Key Points

- Traditionally, the development of investment strategies has required:
  - domain-specific knowledge
  - access to restricted datasets
- The great majority of data scientists lack either or both of these requirements.
- These two barriers exist by design
  - Financial knowledge is hoarded by firms, and protected as trade secrets
  - Financial data is expensive, making it inaccessible to the broad scientific community
- Challenge: Overcome these two barriers, so that all data scientists can identify market inefficiencies
- Opportunity: Increase market efficiency by democratizing finance

For a more detailed discussion, read the paper: https://ssrn.com/abstract=3454234

The Handicaps to
Modern Quantitative Research

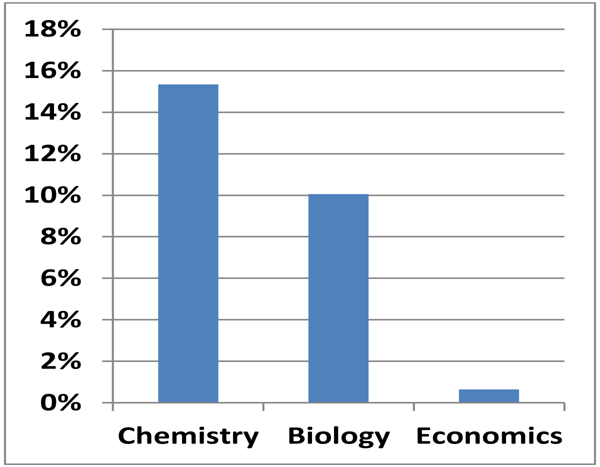
The Old Paradigm

- Investment managers routinely hire teams of quantitative researchers to develop investment strategies
- These teams compete with each other for asset allocations
  - They do not share IP with each other
  - They may not share IP with the hiring firm
- In order to derive these strategies, researchers analyze large amounts of historical data, searching for patterns that lead to repeatable outcomes
- Those repeatable patterns are demonstrated through historical simulations (known as backtests)

Traditionally, quant teams have worked and operated as silos. A firm's ability to cover multiple investment strategies is limited by how many teams it can hire

Handicap #1: The experience barrier

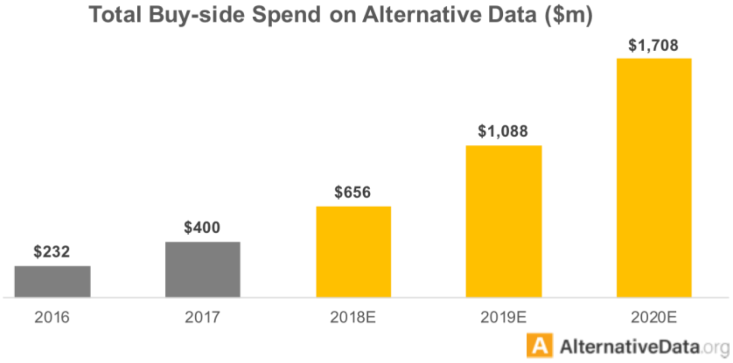
- **Developing an investment strategy typically requires domain-specific knowledge**
- This prevents the participation of the community of data scientists, as only a small fraction of them may understand how to translate the problem of developing an investment strategy into an abstract forecasting problem
- Consequently, many complex investment opportunities are not arbitraged or exploited, because those with the technical ability needed to solve the forecasting problem may not have the domain knowledge needed to pose the problem in the first place



According to the Web of Science, **only 89 articles (0.65%)** articles in economics journals contained any of the following terms: *classifier, clustering, neural network, machine learning*

Handicap #2: The data/resources barrier

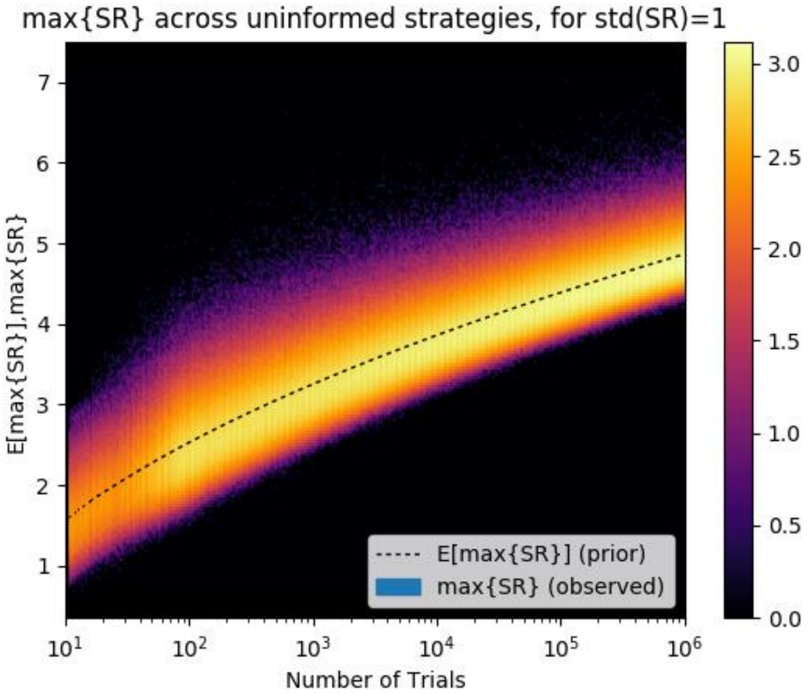
- **Developing an investment opportunity often requires access to expensive datasets, software and hardware**
- Long gone are the days when opportunities could be spotted through the mass media
- Identifying mispriced securities requires sophisticated analyses, which in turn consume complex datasets
- These datasets are expensive to purchase, curate and analyze
  - Add to that the cost of supercomputers and specialized software



The total amount of money spent by buy-side firms on alternative data has multiplied by 4.7 times over the past 3 years

Handicap #3: Selection bias

- **Researchers are incentivized to overfit their models, because they are paid to produce backtests with promising performance**
- Companies typically have no means to prevent backtest overfitting
- In some cases, companies may even encourage overfitting, because they can raise funds from investors through the promotion of those overfit backtests
- One solution is to implement the "assembly line paradigm"

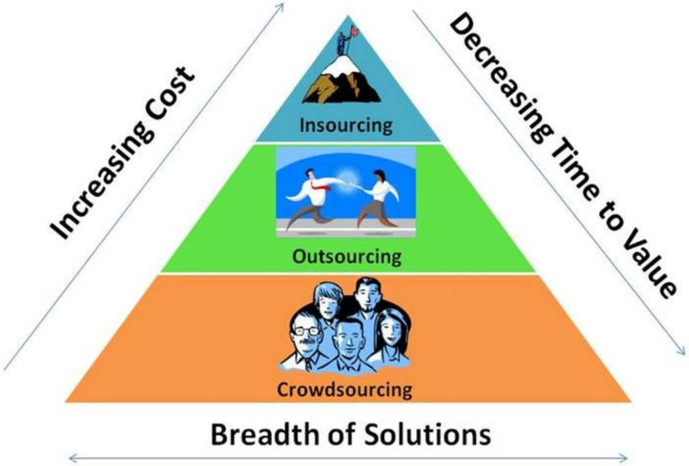


It is trivial to obtain high backtested Sharpe ratios, when firms do not control for the number of trials

Crowdsourced Quantitative Research

What is Crowdsourced Research?

- In crowdsourced research, quants share the following resources:
  - Data
  - Software
  - Hardware
- In additions to sharing resources, crowdsourcing must be **decentralized**
- Central planning defeats the purpose of crowdsourcing:
  - Planning narrows the range of method and ideas
  - It is expensive to micro-manage a crowd
- **The key is to coordinate the work of a crowd through a system of incentives, not directives**

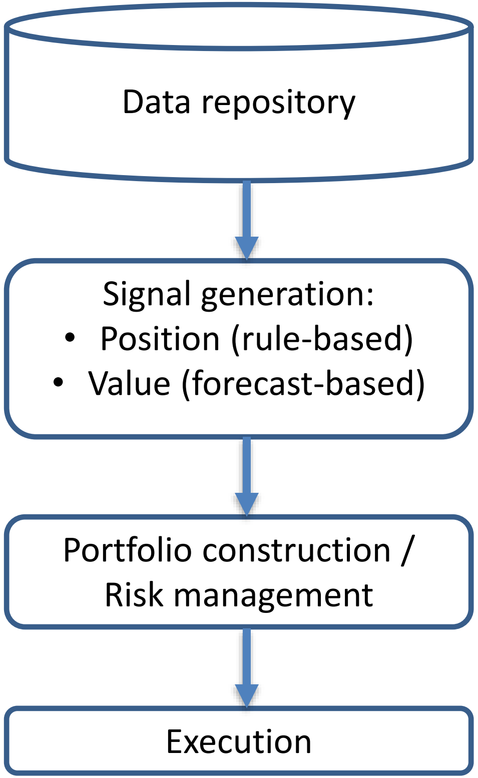


Source: www.blog.econocom.com

Traditional quant firms insource research. Modern quant managers increasingly outsource or crowdsource their research efforts

The Backtesting Platform Approach

- In the platform approach, research is undirected, and the deliverable is *trades*, not forecasts.
- Researchers develop or use all strategy modules:
  - Data analysis
  - Trading rules (or, less often, forecasting models)
  - Risk/Portfolio management
  - Execution
  - Backtesting
- Backtesting platforms can effectively address Handicap #2 by providing data and resources to their users, however:
  - **Handicap #1** (experience barrier) is left to the user rather than solved by design
  - **Handicap #3** (selection bias) is hard to overcome, due to unsupervised backtesting and selected reporting of outcomes



Data repository

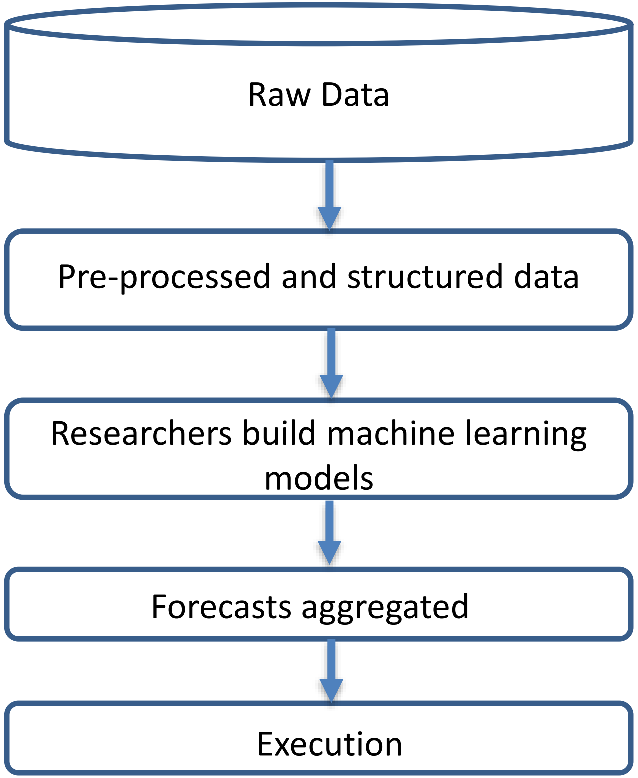
Signal generation:
- Position (rule-based)
- Value (forecast-based)

Portfolio construction / Risk management

Execution

The Tournament Approach

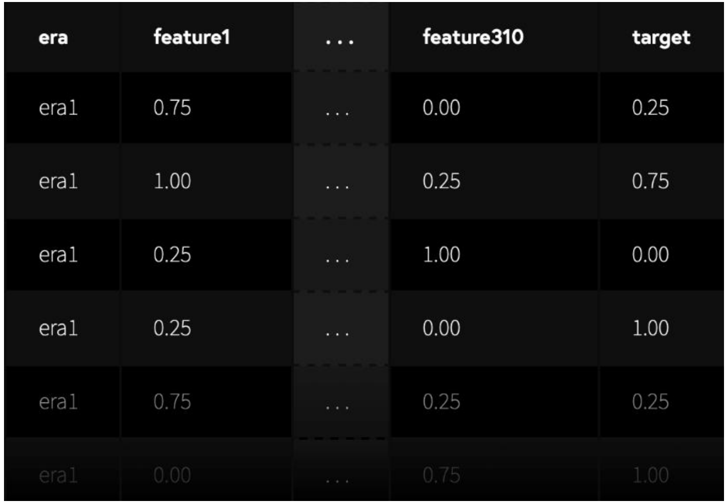
- In the tournament approach, research is directed, and the deliverable is *forecasts*, not trades
- Tournaments address:
  - **Handicap #1**: The data can be pre-processed and structured by experienced researchers, and a relevant problem (useful to an organization) is cast as a forecasting challenge
  - **Handicap #2**: Data is made available. Less commonly, remote access to computing cluster can be provided
  - **Handicap #3**: The tournament can be designed in such way that selection bias does not take place or, if it takes place, it is accounted for
- Tournament design is a complex subject, that can be best illustrated with an actual example


Raw Data
Pre-processed and structured data
Researchers build machine learning models
Forecasts aggregated
Execution

Case Study:
The Numerai Tournament

Data Structure

- Investment universe of 5,000 names across countries, sectors and sizes
- Cross-sectional real-valued **X** matrix
  - one matrix per month (monthly Eras)
  - approx. size of $(5000x310)$
  - values are adjusted for country-sector-size biases
- Categorical **y** array
  - one array per month
  - labels follow the fixed-horizon method, where the horizon is one month (no time-overlap across Eras)
  - {0,0.25,0.5,0.75,1}, indicating outperformance / neutral / underperformance against peers (targets adjusted for biases)
- New features are added from time to time



era	feature1	. . .	feature310	target
era1	0.75	. . .	0.00	0.25
era1	1.00	. . .	0.25	0.75
era1	0.25	. . .	1.00	0.00
era1	0.25	. . .	0.00	1.00
era1	0.75	. . .	0.25	0.25
era1	0.00	. . .	0.75	1.00

Numerai data sample (https://numer.ai)

Data Obfuscation

A common problem in tournaments and backtesting platforms is that users may incorporate information from the *test* set into the *train* set.

To prevent guided searches, (X, y) are obfuscated:

- A random distance-preserving transformation is applied to all X values
- Rows are shuffled within each Era
- Row names are encrypted
- Column names are encrypted

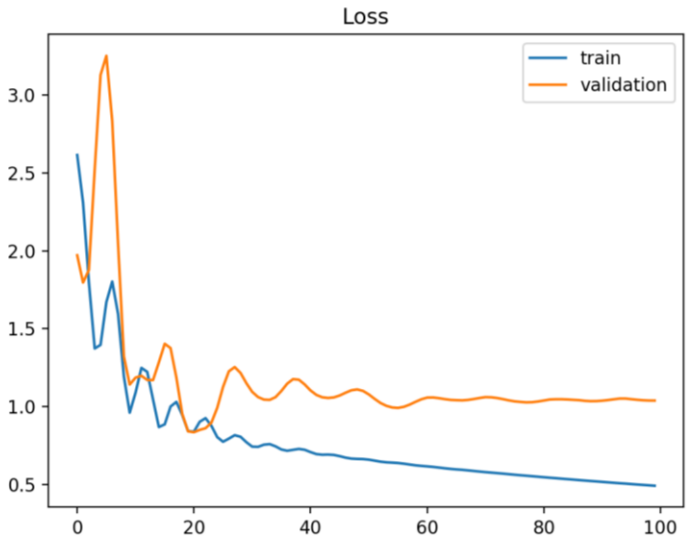
Obfuscation prevents users from applying their own biases to forecasting. They can only model the problem posed, using only the data provided.

Model Training

Data splits:

- Train set: 120 months (eras)
  - y-array is shared
  - useful for K-fold cross-validation
- Validation set: 12 months
  - y-array is shared
  - useful for hyper-parameter tuning
- Test set: 63 months
  - y-array is not shared. Users must forecast it
- Live set: recent data for live trading

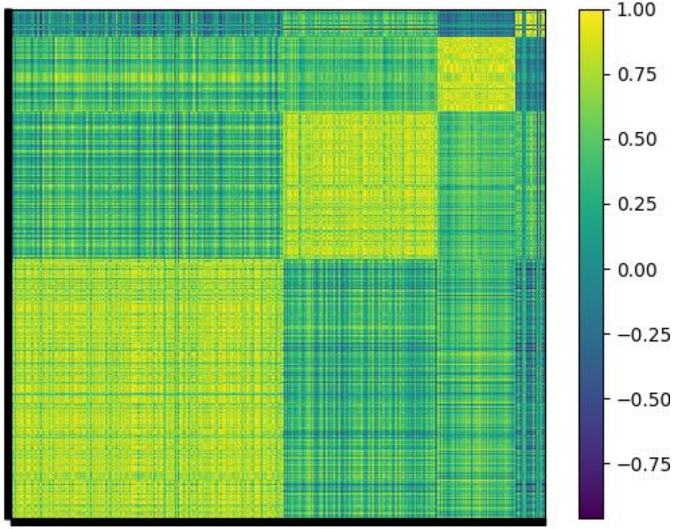
No p-hacking: Selection bias by the users is prevented with the obfuscation of the data, and by hiding the test-set's y array.



To avoid train-set overfitting, researchers must monitor at what point the forecast error *on the validation set* ceases to fall

Model Scoring & Selection

- There is one tournament per week
- Multi–class forecast probabilities can be condensed into a single score
  - For example, $[0, .1, .2, .7, 0.] \rightarrow .6 + .7x(.8 - .6) = .74$
- Among the contributed models for a given week, the best-performing ones are selected based on a combination of factors:
  - The Spearman correlation of the forests on the test set with the actual targets: the ability to discriminate between outperforming and underperforming stocks
  - Novelty: models with uncorrelated forecasts are preferred
- In addition to preventing users' selection bias, Numerai also corrects for its own selection bias



A clustering algorithm can be used to identify subsets of similar / redundant models, and enforce diversification

User Staking

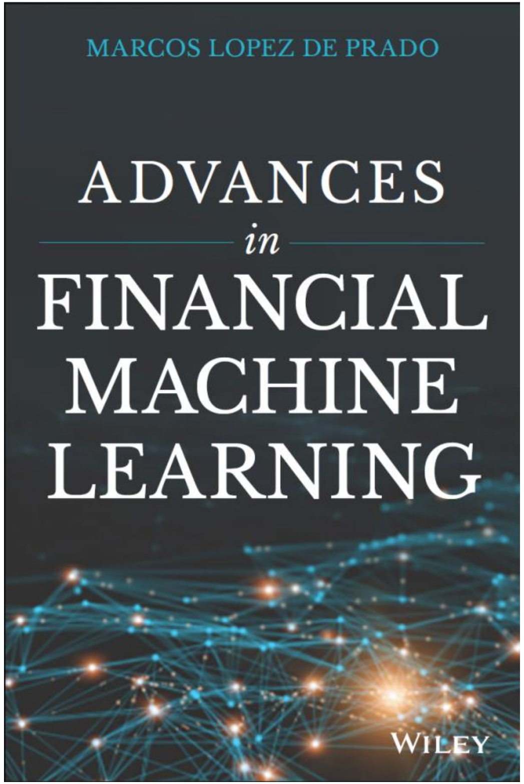
- Users are incentivized to contribute good models, not to overfit
- Users put skin in the game by staking cryptocurrency on their live performance
- A user’s reward is a combination of
  - Numerai’s payout for that week
  - Performance of contributed signals over the next month on the live data
  - Amount staked by the user
  - Long-term consistency
  - Confidence in the model expressed by the user in an auction mechanism
  - There are no rewards based on backtests, or the test set.

Scorecard

Comparing Three Research Paradigms

	Silo	Platform	Tournament
Experience barrier	Each team must include financial experts	Each participant is required to be a financial expert	Financial experts frame the problem, so that non-experts can solve it (misspecification risk)
Data barrier	Data distributed to a small team	Data must be distributed to a large community (many licenses needed)	Data obfuscation permits the redistribution of large amounts of data per $1 spent
Data monetization	Low	High	Extremely high
Selection bias	Preventable ex-post, through the collection of metadata	Possible to prevent ex-post, but difficult and costly	Prevented ex-ante, by design
Output	Trades, portfolios, forecasts	Trades, portfolios	Forecasts
IP ownership	IP typically owned by the firm	IP belongs to individual, but it may have to be shared	Owned by the individual (no code is ever shared)
Hardware / Computing	Paid by Firm	Paid by Platform	Paid by Contestants
Breadth	Low	High (undirected crowd)	Medium (directed crowd)
Depth	Low (due to small team)	Low (due to undirected research)	High (thanks to directed research)
Scalability	Low	Medium	High

For Additional Details



The first wave of quantitative innovation in finance was led by Markowitz optimization. Machine Learning is the second wave and it will touch every aspect of finance. López de Prado's Advances in Financial Machine Learning is essential for readers who want to be ahead of the technology rather than being replaced by it.
— Prof. **Campbell Harvey**, Duke University. Former President of the American Finance Association.

Financial problems require very distinct machine learning solutions. Dr. López de Prado's book is the first one to characterize what makes standard machine learning tools fail when applied to the field of finance, and the first one to provide practical solutions to unique challenges faced by asset managers. Everyone who wants to understand the future of finance should read this book.
— Prof. **Frank Fabozzi**, EDHEC Business School. Editor of The Journal of Portfolio Management.

THANKS FOR YOUR ATTENTION!

Disclaimer

- The author of this presentation advises and/or has advised asset managers, including but not limited to asset managers that develop backtesting platforms and crowdsourcing tournaments.

- The views expressed in this document are the author’s and do not necessarily reflect those of the organizations he is affiliated with.

- No investment decision or particular course of action is recommended by this presentation.

- All Rights Reserved. © 2019 by Marcos Lopez de Prado