

How It Works & How to Participate

January 2022

1 Introduction

The complexity of the ecosystem¹ lies in the design of the AI that governs it and the OEMS that executes its orders. Leaving those aside for the moment, the system itself is straightforward:

1. Users in our decentralized ecosystem send in features and/or algorithms.
2. Our AI creates forecasts from the best combinations thereof.
3. Additional algorithms turn those forecasts into systematic strategies.
4. The OEMS executes those strategies.
5. Participants are rewarded in proportion to (1) the information they contributed to models that went live in systematic strategies, (2) those strategies' monthly profits, and (3) the amount they have staked.

1.1 What Does it Take to Participate?

Write a script that formats your data & sends it to our [API](#); then schedule a cron job to run it. That's it. It takes about an hour. The particulars are below.

1.2 Why Participate?

With one hour of set-up time, and without revealing your IP, you can test your features in millions of modeling contexts. If any of those contexts prove profitable, you will be well-rewarded with a share of the profit from capital you did not have to raise.

When we write 'millions of modeling contexts' we mean across assets, across time frames and in conjunction with an effectively-endless, constantly-evolving set of high-value features that other funds have spent years building.

Finally, for early participants, the reward structure is analogous to those that successfully 'aped into' other successful DeFi projects.

¹If you are not already familiar with this project, you can find a short overview [here](#).

1.3 Who Should Participate?

This ecosystem is built as a tool for advanced quant funds and developers who,

1. have already spent a lot of time engineering algorithms and features which they suspect can be monetized more thoroughly than their current use cases; and
2. would like a low time-cost way to test new algorithms, features or operationalizations extremely thoroughly.

That being said, especially at the beginning, there is a lot of ‘low hanging fruit’ contributions needed by the ecosystem, so if all you bring to the table is some basic coding skills, please get in touch and we can point you in some super-useful directions!

2 Purpose of this Document

This document is for people who are curious enough that they are considering participating in the ecosystem. It organizes the documents and resources we have created thus far and walks them through related considerations.

It is meant as a living document that makes onboarding super-easy. *We are in the alpha testing phase, so we surely won't get it right right away.* If you have questions or suggestions, please contact marc [at] judgeresearch [dot] co or nick [at] judgeresearch [dot] co. We'd love to help and to hear from you how we can improve the onboarding process.

If you want a more basic overview of what we are up to than this document, check out the link in the footnote on page 1.

3 Step 1: Submit Your Features and/or Algorithms

Text in **red** is particular to our MVP. Sections in red will be removed as various parts of our build go live.

Having a better idea how our AI works isn't necessary but might help develop your intuition. Our **technical 3-pager** is a light, quick read. For the rest of this paper, we will use ‘genes’ as shorthand for ‘features or algorithms.’

Your first step is to go to judgeresearch.co and decide for which time frames and assets you'd like to submit genes. (If the site doesn't load, you may have to turn your VPN off temporarily.)

3.1 Features

The AI will sporadically try your features out on all dependent variables. But most of the chances your features get to prosper will be in the ‘gene pool’² of

²see the below section, ‘Defining a Claim.’

the dependent variable you choose.

You can instantiate your own independent variable. For instance, we started with 4 hour forecasting windows for BTC and ETH. But perhaps you would like to begin training your features on 15 minute intervals; or a different asset altogether, say, the gold-usd pair.

The Alpha testing era will focus only on 4 hour forecasts of BTC and ETH.

See our [API documentation](#) for endpoints, data formatting, and other specifics of submitting your features. *coming soon:* Our API is about to go live. For the early part of the alpha testing, there is a convenient google form for you to drop your features into at [judgeresearch.co](#).

3.2 Algorithms

Features have an easy way to fully protect your IP: Just don't label your data. Unfortunately, the same is not true for algorithms. You can submit algorithms in Python, R, C++ and Julia. However, we have to read them first. Still, we will destroy them if you withdraw them, and we won't reproduce them.

We anticipate the algorithm space will be smaller than the feature space by orders of magnitude. So that does increase the chances of your algorithm getting picked up by our AI, and thus its eventual value to you as a gene.

Two additional rules dictate algorithm submission:³

1. It must take the form of $\hat{y} = f(X)$. That is, it must take in independent variables of any dimension and return forecasts that match the number of rows of X (with NAs for any missing rows)
2. It must not exceed 15,000x the processing costs of a simple regression. Scripts that do will result in some of your stake being pulled. For algorithms below this modeling cost, the fit - processing cost curve will be published at a later date.⁴

³Eventually, the lagged error terms generated by algorithms will be thrown into the mix as a feature created by your algorithm. 'Secondary genes' such as this are a more complex topic, however, and likely will not appear in the first year following the alpha test. If you're animated by questions like how to design a reward structure governing error terms as a gene created by a submitted algorithm, get in touch! We'd love to have you on the team.

⁴For now, all algorithms beneath 1500x will be treated the same, while those 1500-15,000x will be docked 1% of our various fit metrics. The eventual effect of the curve will be a bias towards low computational costs that will only be sacrificed in the face of considerable gains in predictive inference.

- For algorithms with certain types of highly stochastic structures - e.g. tree-based methods - consider limiting the number of trees/other stochastic substructures. The large number of trials your genes will enjoy can be considered an amplification of the number of stochastically built structures your algorithm will create.
- For numerical optimization, try stochastic optimization routines more efficient (so, ironically, the opposite of GAs) than full-blown MCMCs. Unfortunately, you will have a hard time specifying starting values, given that you can't specify columns in your X!
- Another tip is to start your algorithm by detecting the number of columns of X and adapting certain parameters to fit in under computational limits. After our alpha tests,

4 How the AI works

See our [technical 3-pager](#) for a review of the basic architecture.

5 How Do I Receive Rewards and How is the Reward Determined?

Once we end alpha testing Your reward will be dropped monthly in the staking vault associated with the ethereum wallet you submit via either our google form or the API. You may remove it the following month.

The one month delay is an important feature of the staking system, which is necessary to discipline the ecosystem, and keep it from being overwhelmed by ‘spray and pray’ submissions.

See our [tokenomics paper](#) for the details of the reward structure.

6 A High-Level Overview of Participating and Related Considerations

Fig. 1 represents the basic structure of the ecosystem. It may look complex, but don’t be overwhelmed. The system is simple: Users submit features and algorithms to our AI, which combines them in many ways, chooses the best combinations and creates forecasts. Users are rewarded in proportion to the informational value of their contributions, the amount the relevant strategies makes that month, and the amount users stake.

The OEMS has already processed more than \$10 billion of real-world trade volume, and is quite efficient at avoiding slippage.

The AI has been under active development for two years, and is based on many years of academic work before that.

6.1 Compensation

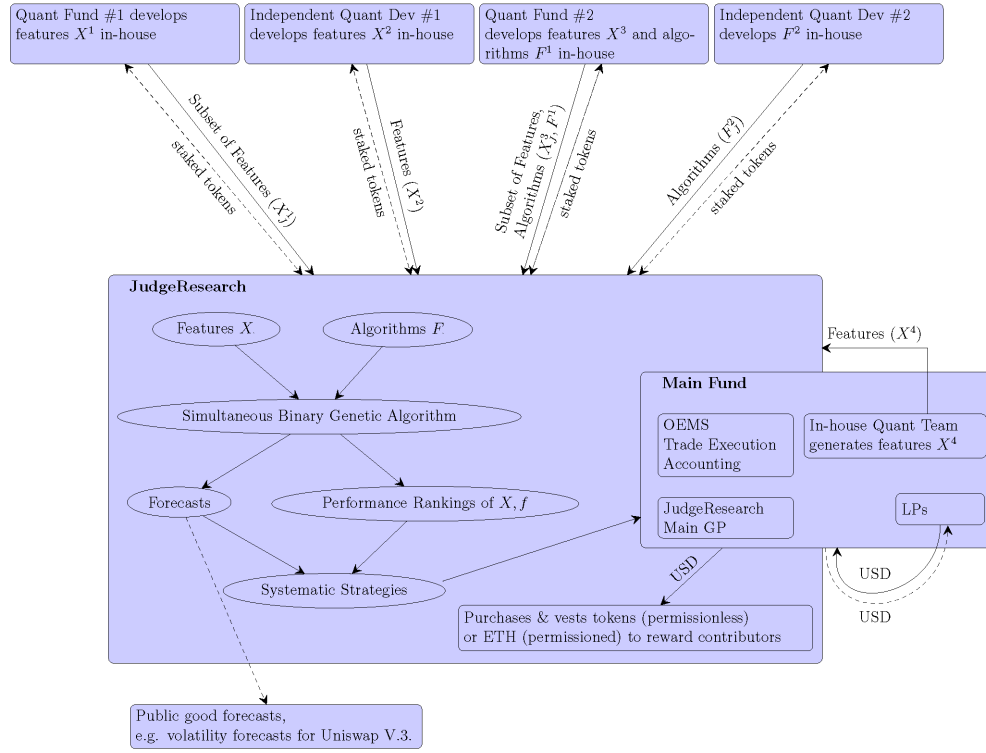
Payouts from the system occur on a monthly basis. The vast majority of the main fund’s ‘2 + 20’ flows to ecosystem participants, via token purchases/vesting, and the remainder to the GPs of the fund. More granular detail of the compensation formula can be found in our [tokenomics paper](#).

While we alpha test the AI, the system exists purely as a data experiment. The fund has not yet been set up, and we will not yet allocate actual capital. Alpha testers do, however, claim their subspaces in the feature space, along with any future claims that may relate to that space.

During Beta testing, when the system exists as a permissioned, tokenless ecosystem, the staking and payouts will be in ETH or a set of stablecoins.

we will begin publishing special characters that contain special information (such as the cost of running a regression on the current X) for you to customize your scripts.)

Figure 1: The Decentralized Ecosystem as a Wireframe



6.2 Claiming Subspaces

Users ‘claim subspaces’ of the feature space by being the first to submit features. For instance, the lagged differenced percentage value of BTC will likely belong in any number of - if not most - models, and thus be incredibly valuable ‘land’ to hold, in part because it takes almost no effort to create and submit. Because we’ll use this example throughout, we refer to this variable as x .

6.3 Defining a claim

We’ve built what we call a Simultaneous Binary Genetic Algorithm (SBGA). See our [technical 3-pager](#) for more details. If statistics isn’t your thing, it is enough to know that the system is a reasonably strict statistical representation of the process of natural selection. Every gene is a gene in an evolutionary pool, and ‘evolutionary fitness’ is defined as the out-of-sample fit of forecasts and other measures of forecast - real observation alignment. Forecasts are the product of a set of genes. Forecasts can therefore be called ‘beasts,’⁵ and the systematic strategies that marshal those forecasts, ‘packs.’ The fittest beasts mate and move on to the next generation. Even if a gene dies off, the occasional mutation brings it back to life - but the vast majority of randomness comes from the crossover, or ‘mating’ function, *not* random mutation.

Generations are constantly evolving in extremely stochastic ways.

As academics that probably would not make it in the wild: We can assure you that not every gene is fit for every environment. Some will prosper in some parts of our ecosystem, some in others. The vast majority will die off.

6.4 Defending Subspaces

If one fund submits a set of genes that does better than yours at whatever role your gene is playing, you may have been ‘attacked’ successfully, and pushed out of some or even all of the models. Life, like markets, is a brutal place.

Unless the attacker genes are vastly superior to yours, you will be given a chance to respond, and may be able to do so. You respond and defend your subspace by improving your variables further.

For example, you might want to represent x by quintile, since it is easy to see various models playing better with thresholded variables.

6.5 The Role of Staking in Keeping Things Fair - and Statistically Efficient

Our staking system penalizes people financially for a sort of statistical spamming of the system. If they submit variables in a ‘spray and pray’ fashion, they will be penalized. Continuing on with our example, x thresholded into five columns might be a good idea, but into 100 is likely not.

⁵The genetic algorithm literature confusingly uses ‘individual’ and ‘chromosome’ for the same concept. We think it’s cooler to call them ‘beasts.’

Submitting 100 variables that die off quickly will cost users a portion of their stake, and you will have retained your subspace x .

6.6 The System as a Testing Ground

Users also have their stake pulled for submitting variables for a brief period - say, one month - and then removing them from the system.

One of the more elegant aspects of this staking system is its creation of a dual-purpose ecosystem.

The main point of the ecosystem is to rely on a decentralized network of funds to rapidly grow a curated feature and algorithm space capable of surpassing that of the world's most advanced quant funds.

A secondary use, however, is to rapidly, extremely thoroughly test features and algorithms one only intends to use in-house.

For example, a user is perfectly welcome to submit 100 operationalizations of the same feature. They may leave them in the system, and see which prosper and which die off. They will effectively pay a fee for their use of the system when, for example, ninety-nine of them are replaced by the one superior feature, and their stake is reduced. They will further lose some of their stake when they pull that final, superior gene after a short amount of time.

They will have contributed to the ecosystem the funds they staked. But they would have gained, at a time cost of roughly one hour, the type of extremely robust testing that might take a full dev team years to build.

7 Conclusion

One can understand the power of this system: Unintentional overfitting is the aspect of financial modeling that even the most seasoned experts struggle to avoid. Avoiding it requires a profound amount of discipline and is often not possible. Evolution - life - can be thought of as a set of harshly-designed experiments whose information aggregation scales beyond our analytical limits. By combining evolution's statistical framework with the exogenous variance of many other funds' work, we have a system with so many millions of unpredictable tests that it seems - at the least - capable of besting our propensity to overfit.

It worked, in the end, for the dinosaurs - perhaps DeFi should start helping others join their ranks.