Using the API

April 18, 2022

1   Introduction

Our REST API is the main tool you use to contribute features and algorithms to the ecosystem. This document walks you through how to do so, with specific attention to the alpha test that starts May 5th, 2022.

During the alpha test, everything, including this API, is brand new. If any part of using the API, or onboarding in general, is not intuitive after reading this document, please send us an email,[1] or send us a message on Telegram.[2] We will respond as soon as we see it (we are all on EST time) and try to smooth out whatever was the issue.

The goal is to make this easy, fun, and useful for you, but this is an alpha test: We surely will need your feedback to get things right.

2   The Basics

The basic workflow for contributing features to the ecosystem:

1. Get a private key from this form.

2. Format your data as a JSON object.

3. Submit the data that covers the appropriate in-sample time frame.

4. Schedule a cron job to send in real-time updates thereafter.

The following pages walk you through everything you need to know to submit features and begin claiming 'land' in the feature space.

---

[1] gina [at] judgeresearch [dot] co.

[2] [at] ckerman

3   Submitting Your Features and Algorithms

3.1   Submitting Algorithms

The goal of the main launch is to accept both algorithms and features. However, for the alpha test, we do not have means to test algorithms you submit without looking at your code. The alpha test will therefore be primarily limited to the submission of features.

We will include a basic library of models that will be genes responsible for turning other genes into forecasts. If you want to submit algorithms, get in touch at the email/telegram in page 1's footnotes. We'd love to have your help!

3.2   Submitting Features

For the alpha test, the endpoint to submit features, via a PUT request, is,

```
https://ct6gu86m3d.execute-api.us-east-2.amazonaws.com/dev
```

Anything larger than 2 megabytes will be rejected. If you are sending in multiple features, your first send might be larger than this - but you can easily loop over PUTs without an issue.

Your API key is required in the header, with the following format:

1. `x-api-key`: the API key we have sent you

No additional header information is required.

3.3   The API Key

Use this form to get your private key. You'll receive an email with a key soon after you submit the form. You can reply to that email if you have any questions and an actual human being will respond.

3.4   Formatting Your Data

The API accepts only JSON objects. In the first example, the user is sending in a new feature, from the start date of the training period of the alpha test, up to the most recent observation available:

```json
{
    "data" : [
        {
        "attribute": "2019-01-01T00:00:00Z",
        "DV": "BTC-USD",
        "MBS": "60",
        "featureName": "x1",
        "value": "17",
        "IPP": "last"
        },
        {
        "attribute": "2019-01-01T04:00:00Z",
        "DV": "ETH-USD",
        "MBS": "240",
        "featureName": "x1",
        "value": "19.2345337",
        "IPP": "last"
        },
        ...
    ]
}
```

And in this next example, the user is updating two features with a new observation:

```json
{
    "data" : [
        {
        "attribute": "2022-03-01T16:00:00Z",
        "DV": "BTC-USD",
        "MBS": "240",
        "featureName": "x1",
        "value": "-22.2",
        "IPP": "last"
        },
```

```
11                {
12                "attribute": "2022-03-01T16:00:00Z",
13                "DV": "BTC-USD",
14                "MBS": "240",
15                "featureName": "x1",
16                "value": "289899.2",
17                "IPP": "zero"
18                }
19          ]
20  }
```

Every element is listed in an array named `"data"`. Each object in the array should contain the following six key-value pairs:

- `"attribute"`: a string representing the time block of the observation, in coordinated universal time following the ISO 8601 standard, in UTC. The attribute should be structured as `"YYYY-MM-DDTHH:MM:SSZ"`. For the alpha test, if you mistakenly send in objects with a date prior to `"2019-01-01T00:00:00Z"`, you won't receive an error message but earlier observations will be discarded.

- `"DV"`: a string specifying the dependent variable. Choices during the alpha test are `"ETH-USD"`, `"BTC-USD"`, and `"V-BTC-USD"`, which denotes the volatility of BTC-USD.

- `"MBS"`: a string specifying the block size of the observation, in minutes. This can be any positive number. For the alpha test, we are currently set to test four hour series, so `240` should be the value you set. We will likely include another time frame for the alpha test, and will update this document when we do.

- `"featureName"`: an alphanumeric string for the name of the feature, of length no more than 50 characters. You can name the feature whatever you want.

- `"value"`: the most important part! A string with the value of the feature, also of length no more than 50 characters.

4

- **"IPP"**: *optional, defaults to* `"last"` - what type of interpolation procedure is appropriate for your feature. Values can be either `"last"` - fill in the missing value with the last observed value - and `"zero"` - fill in the last value with a 0.

3.5 A Few Extra Notes

3.5.1 Naming Your Feature

We assume that you will not want to reveal your features' substance by labeling it accurately. So just make sure you give it a unique identifier that you can remember. The downside to protecting your IP is that no one but you is keeping track of your features. Submitting a few observations under mixed-up names might doom your feature to statistical irrelevance in our tests!

Remember, the system by design does nothing to track your features. If you mix up your feature names or otherwise mis-organize your features, an error may not be thrown but the AI will eventually throw your features out as not helpful.

3.5.2 Specifying Your Dependent Variable (Target Feature)

When you submit a feature, you choose your target feature. Your feature will be given a small chance to be run in models with other target features. Thus, if it does exceptionally well, it will grow in the population and become prominent for target features other than the one you choose. However, more than 95% of the initial runs (chances at life) your feature gets will be related to the target feature.

If you want to submit the same feature for multiple target features, simply send the feature in multiple times with different `"DV"` values.

3.5.3 Additional Validation and Data Issues

- Additional elements in the JSON beyond the array `"data"` or any additional elements in each object in `"data"` will return a 400 error.

- Duplicate elements in `"data"` may not be included.

- Each object in `"data"` must contain the four, and only four, elements: `"attribute"`, `"DV"`, `"featureName"`, `"value"`, each as a string.

- Only a feature's current time block can be overwritten. If you try to overwrite any other time block, you will receive an error message.

4   Updating Your Data

After the initial submission of the training sample, each observation /textitmust be submitted *before* the turn of the time period. For example, the cut-off submission time at the turn of the day is 23:59:59, not midnight. Like OHLC data and most time series data, the timestamp - 'attribute' in your JSON - refers to the beginning of the UTC time period. For example, for an hourly time series, thw row labeled '20:00' refers to the time period 20:00-20:59. Updates may be submitted at any time during this period.

5   The Alpha Test

The alpha test begins at the end of Q1, 2022, but you or your fund can start submitting data now.

It is a data experiment - no live positions will be taken, as the fund itself is not yet set up.

While the alpha test is an experiment, users still 'claim their land' (see 'How It Works') by moving first. The massive, overwhelming first-mover advantages we've built into the incentive structure could, therefore, end up affecting your rewards for years to come.

After your first submission, you should schedule a cron job to send in new data every period. The contest will run for 45 days.

The judging criteria will be the one-step-ahead forecasts during the 45 days of live data you send in in real time. Users that submit features before others, with an identity being defined as as $\rho \geq .975$, will retain their 'land.'

The long-run emphasis of the project is for experts to send in high-value, hard-to-engineer features. However, in the beginning, easy-to-engineer features may prove invaluable land to hold in the feature space, while involving very little overhead for participants. For example, the simple logged differenced OHLC data with various lags and various pairs will likely be included in a large number of models in the future, once the fund goes live. Whoever submits those first, if they continue to do so reliably, will likely see their features enter into many models of many assets.