USER MANUAL

Apply machine learning predictions to your data without any programming knowledge.

PredictNow.ai

TABLE OF CONTENTS

INTRODUCTION

PredictNow.ai helps you apply machine learning predictions to your data without any prior programming knowledge. All you have to do is create an Excel (.xlsx) or (.csv) file (.csv) with columns of “Predictor” variables and 1column of “Target” variables. Two example data files titled "example_input_train.csv" and "example_input_live.csv" are available for download. Our program learns from your predictor-target pairs and makes predictions on unseen live data that you supply. We will also show you the performance metrics of the model (e.g. How accurate the predictions are).

PredictNow.ai can predict either discrete target variables, such as the sign of returns, or continuous variables, such as the returns themselves.

NOTE: The program allows both (.csv) files and (.xlsx) files to be processed. However, when modifying or creating a (.csv) file via Excel, some characters in the current file may not be correctly converted to (.csv). We encourage Excel users to save their files to (.xlsx), rather than (.csv).

Homepage: https://predictnow.ai

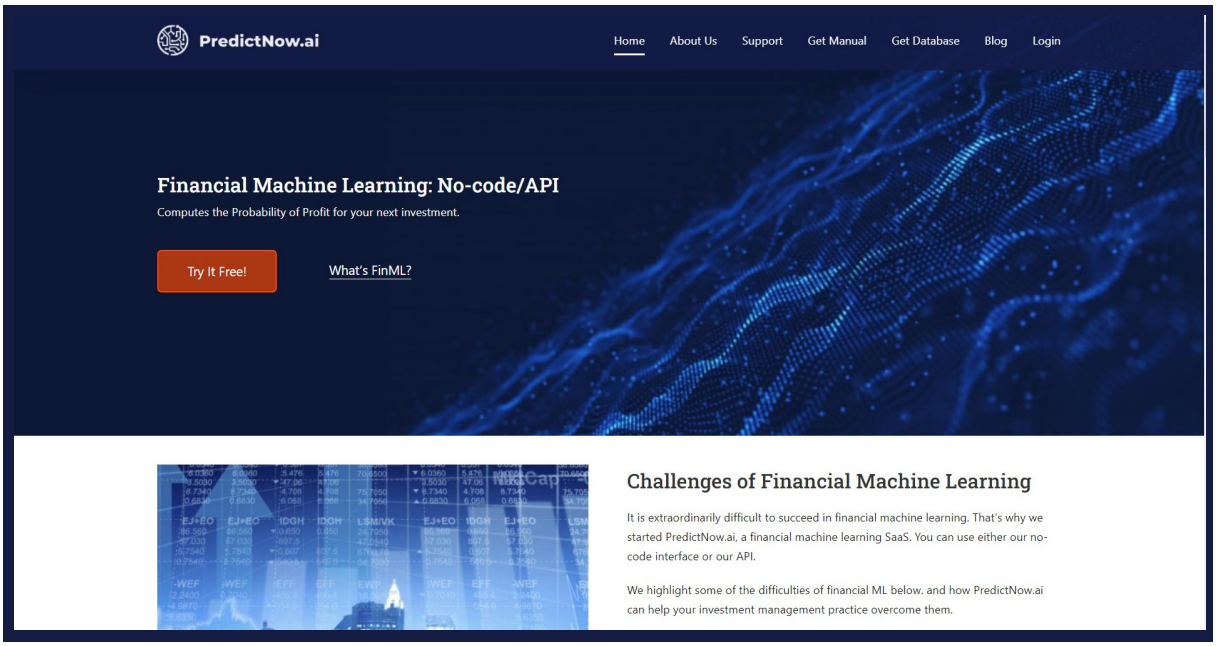


Figure 1: Website Homepage

HOW TO USE

1. Login / Register to the website.

NOTE: Password plaintext is not stored anywhere on the server, only as an encrypted version.

2. Choose between "Train Option" and "Live Option", then click Submit.

Train:

- "Training" means you supply historical data with known outcome (target) for our program to learn how to make predictions.
- Specifies the necessary parameters to perform data preprocessing, splitting between train and test sets, hyperparameter optimization, feature selection, train model. Many parameters have default values that are specified already. (Details in "Hyperparameters for Training" section below).
- Upload the data file on which to perform analysis. Note that there must be a column which will be used as the target variable. This target variable will be specified by the user.
- The program will automatically process this target (also called "Label") column by internally assigning "0" if target is negative or zero, and "1" if target is strictly positive. For example, in a metalabeling application, the target can be returns. In other classification tasks, the labels must have values "-1" or "1".

  NOTE: Other details regarding the data specification in the "Data Preprocessing" section below.
- Outputs from training will be available for download. See "Outputs From Training" section.

Live:

- "Live" means you use your previously trained model to make new (live) predictions.
- Upload a data file made of predictors (also called "features").

  NOTE: if there is a target column, it will be ignored.
- The data provided will be appended to a training dataset used for fitting the ML model, in order to apply data preprocessing (i.e. if input is a time series, there must be no gaps between last date in train data and first date of live data).
- The prediction (.csv) files (with probabilities and predictions) can be downloaded via download links.

HYPERPARAMETERS FOR TRAINING

1. Target: Column name for the target data variable.

This column should contain real numbers, which the program will convert into 2 classes based on their signs (if Classification task) as described above, OR will automatically detect the number of classes in it if the labels are already in the form of classes (e.g. 1, 2, 3, etc.) OR will be kept same if task is Regression.

IMPORTANT: In a metalabel application, if a trade wasn't made for a certain sample, the target should be "NaN" or "Null" instead of "0".

2. Timeseries (Yes/No): Whether the data is in timeseries format. Default: "No".

- If "Yes", then the program will automatically search for a "date/Date/DATE" column. If the program cannot find it, then it will default to "No".
- If "No", then the program will perform default indexing of data (0,1,2,. ..).

3. Target Type/Task (classification/regression): Whether to perform regression or classification.

Classification: Classification can be used in 2 cases.

- If the labels are continuous but the objective is to predict the sign of the label, the program will convert the label into two labels "-1" and "1" and run classification on it.
- If the labels are already in the form of labels with multiple labels e.g. 1, 2, 3, etc. (multiclass), the program will detect the number of classes and run classification on it.

Regression:

- If you want to predict the value of label. The label needs to be continuous values. Output will also be continuous.

4. Feature_selection (SHAP/Cluster MDA/None): Perform feature selection.

This will display a ranking of all your features based on feature importance scores. By choosing feature selection, **PredictNow.ai** will display and select those features which contribute most to your prediction of the target. Some features in your data can decrease the accuracy of the models and make your model learn based on irrelevant/unhelpful features.

4. Feature_selection (SHAP/Cluster MDA/None) *Continued*:

If SHAP, then perform SHAP feature selection.

- The SHAP Tree Explainer feature selection algorithm is an explanation method used for ensemble learning that computes optimal local explanations, using topics from game theory.
- More details can be found at: https://www.nature.com/articles/s42256-019- 0138-9.
- Refer to the following paper by Ernest P. Chan and Xin Man on feature selection: https://jfds.pm-research.com/content/3/1/127.

If Cluster MDA, then clustered feature importance is used for feature selection.

Cluster together features are that are similar and receive the same importance rankings. This promises to be a great way to remove the substitution effect. In our new paper by Xin Man and Ernest Chan: https://py.predictnow.ai/request_cmda_paper, we applied a hierarchical clustering methodology and compare it with MDA feature selection method.

If none, then perform no feature selection.

- Default: "SHAP".

5. Hyperparameter Optimization Analysis (Small/None/CPCV):

Modify the level of hyperparameter optimization. Default: "Small". The parameter grid used for the randomized search is the following:

- 'num_leaves' : [10,20, 30,60, 80, 100, 120, 150, 200, 300, 400, 500,600,700,1000]
- 'n_estimators' : [int(x) for x in np.linspace(start = 50, stop = 4250, num = 200)]
- 'bagging_fraction' : [0.1,0.2,0.3,0.4,0.5,0.8, 0.9]
- 'feature_fraction' : [0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8, 0.9,1]
- 'learning_rate' : [0.03,0.05, 0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9]
- 'max_depth' : [int(x) for x in np.linspace(10, 510, num = 24)]
- 'reg_alpha' : [0,0.02, 0.2, 0.5, 0.6, 0.8]
- 'reg_lambda' : [0, 0.02,0.4, 0.5, 0.6, 0.8
- 'drop_rate' : [0.01,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0.95]
- 'max_drop' : [-1,10,20,30,40,50,60,70,80,90,100,120,140]
- 'xgboost_dart_mode' : [True, False]

CPCV: The optimization is performed using multiple paths and division to simulate multiple trials. This method is extremely useful to prevent overfitting while optimizing. The same hyperparameter grid is used.

5. Hyperparameter Optimization Analysis (Small/None) *Continued*:

If you don't want hyperparameter optimization, the model will be trained using the following default parameters:

- 'num_leaves' : 100
- 'n_estimators' : 100
- 'feature_fraction' : 1
- 'bagging_fraction' : 1
- 'learning_rate' : 0.1
- 'reg_alpha' : 0
- 'reg_lambda' : 0
- 'xgboost_dart_mode : False

6. Testsize: Data to be used for test set.

This parameter will only be used if mode = "train". The testsize value will be used for splitting the feature matrix used for train/test, as well as corresponding labels. Test set is assumed to follow the train set. If "0" <= testsize < "1", we treat this as a fraction of the total number of rows. If "1" < testsize, then we treat this as the exact number of test rows. If testsize = "1" then only 1 row of data will be used as testsize.

NOTE: Testsize is ignored when mode = "live", since all data provided will be used for live predictions.

7. Boost (GBDT/DART): Perform ensemble learning boosting.

- Available options: gradient boosted decision trees (GBDT) or DART technique.
- Default: "GBDT".

8. Weights (Yes/No/Custom): Whether sample weights are used in the model (LGBMClassifier).

- Default: "No".
- Custom: If selected, a "Custom Weights" input box will pop up, here you can give your own weights columns. If kept as NA it will not have any weights selected.

9. Prob_calib (Yes/No): Whether the user wants to perform the probability calibration method.

- As seen on https://scikit-learn.org/stable/modules/calibration.html
- Default: "No".

10. Exploratory Data Analysis (EDA)

- Performs basic statistics about the input data, such as mean, standard deviation, counting number of Null values, etc. It will warn user if there are columns with too many null values, as well as the respective column names and how many nulls those columns have.

  NOTE: Currently, the threshold for raising this warning is "0.1". I.e., if there are columns with more than 10% of elements are NaNs. It will also warn the user if the dataset has too few rows. Current minimum number of rows before a warning is 100.
- Default: "No".

11. Suffix (String): File suffix used for renaming the output files.

INTERNAL PROCESSING STEPS

(This section contains technical details and can be skipped on first reading.)

Our current backend program achieves the following pipeline:

1. Data preprocessing

- Check whether dataset file is (.xlsx) or (.csv). Otherwise, an error will be raised
- If the mode = "train", remove rows that have no label. If mode=="live", then any label will be ignored. The user does not need to provide any labels column in livemodel.
- "Null", "in"', "N/A", "nan", (in any combination) will be interpreted as "NaNs" (null values)
- Remove all special character signs such as comma (,), dollar ($), euro (€), yen/yuan (¥), etc. from a specific cell
- Program will try the following:
    - After removing special characters found in non numerical columns, it will try to convert all elements of those non-numerical columns to numeric datatype.
    - If this fails, all of the elements from that column will be interpreted as string datatype and implicitly will be treated as a categorical column.
    - Look for "date/Date/DATE/Time/TIME/time" column name in the dataset. If one finds such a column, it will automatically set the index of the dataset to that column and will not be used as feature. All date (Python/Pandas) formats are acceptable, including dates and time in the same column. For example, '2013/02/01 12:00 AM' is an acceptable format.

2. EDA analysis:

Saves output from panda describe() method in 'filename.csv', where filename is Save output from panda profile report in 'profile_report_filename.html'.

3. Perform one-hot-encoding of features that are categorical.

Note that Categorical features must not have pure numbers! Users are warned that all cells in the data frame that have numbers only will be treated as numerical.

4. Perform fractional differentiation (if user indicates numerical features are time series data).

If the numerical features are not under timeseries format, then the concept of stationary features is inapplicable.

5. Perform features selection (SHAP) w.r.t Nancy's rank averaging score.

We use purged CV whenever timeseries = "yes".

6. Perform hyperparameter optimization using cross-validation, with accuracy or F1 score as objective.

- If the ratio of (nr. of class 0 elements)/ (nr.of class 1 elements) is in [0.6,...1.6], then we use accuracy objective for RandomizedSearchCV. The intuition is that in this case classes are 'balanced', and accuracy might be appropriate.
- If otherwise, i.e., classes are "Imbalanced", then F1 score (precision + recall) is necessary. If the user chooses "None", no hyperparameter optimization is performed.

7. Compute AUC/Accuracy/F1 score on both CV and test set and output indication of predictability (whether >0.5)

8. Compute probabilities of class 1 on both CV sets (test folds) and testset.

9. Predict labels on both CV sets (test folds) and test set, assuming probability threshold of 0.5.

OUTPUTS FROM TRAINING

After training is done, website will display the performance metrics, for both CV and test sets, such as for classification : accuracy, F1 score, AUC score and for regression: RMSE, MAE, MAPE, t-statistic, p-value. Other output can be downloaded via links provided in the Results page.

Output files:

performance_metrics_<<suffix>>.txt=same as the metrics displayed on Results page.

predicted_prob_cv_<<suffix>>.csv=predicted probability of class 1 (i.e. positive target) for CV (test folds)

predicted_targets_cv_<<suffix>>.csv=predicted classes (based on probability threshold = 0.5) for CV (test folds)

predicted_prob_test_<<suffix>>.csv=predicted probability of class 1 for test set

predicted_labels_test_<<suffix>>.csv= predicted classes (based on probability threshold = 0.5) for test set

summary_plot_all_test_folds_<<suffix>>.png= plot most relevant features across all folds using SHAP feature selection saved_model_<<suffix>>.pkl= ML model saved into a pickle file using joblib dump function

EXAMPLE

1. To download this instruction manual, click on "Get Datasets" at the top of the page.

2. Download example files.

- To download the "example_input_train.csv" click on "Finance Train Data".
- To download "example_input_live.csv" fileclick on "Finance live data" at the top of the page.
- The "example_input_train.csv" file contains a feature matrix with several technical indicators along with 10-day (buy and hold) SPY future return as target variable (also known as label).

  NOTE: The window length for technical indicators are arbitrarily chosen, since this is used for illustrative purposes only. We would like to train a ML model based on this feature matrix and target variable.

3. Register by clicking on "Try for Free"

4. Enter account details.

- In the form, input the username, password, retype password and click Register.
- You will be redirected to a Paypal window.
- You will need to sign the subscription agreement, otherwise the sign up is incomplete and will not be able to login!

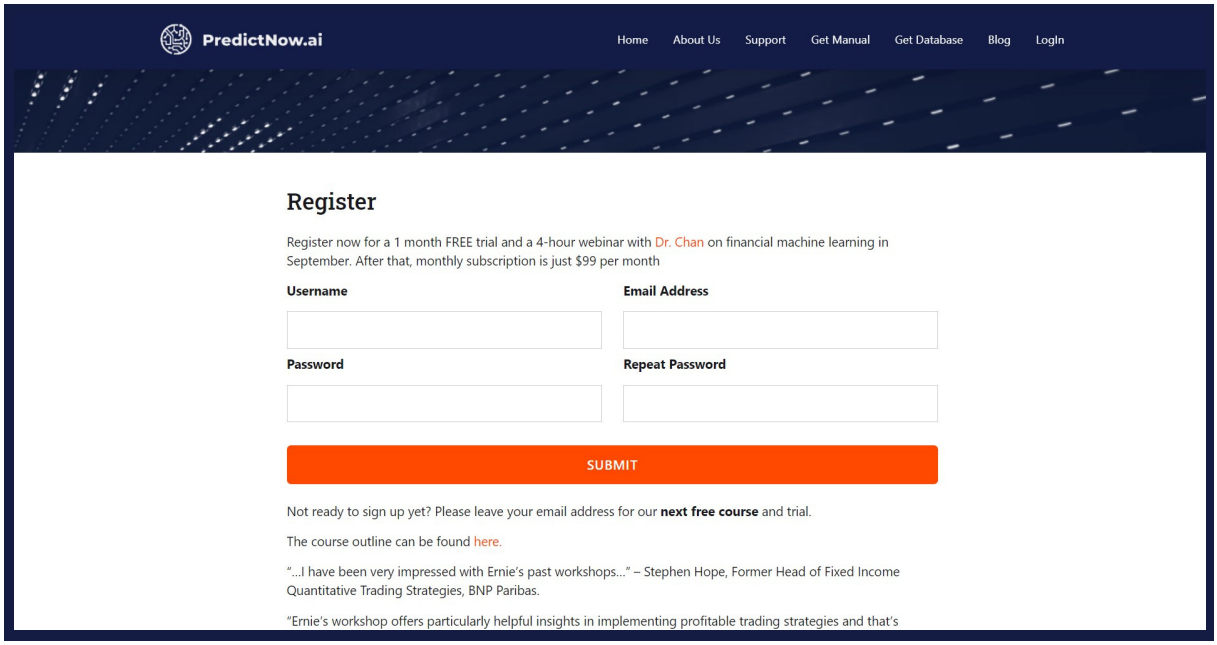


Figure 2: Registration Form

5. If you want to Login, click the "Login" button on the top right page. Input username and password and click on "Sign In".

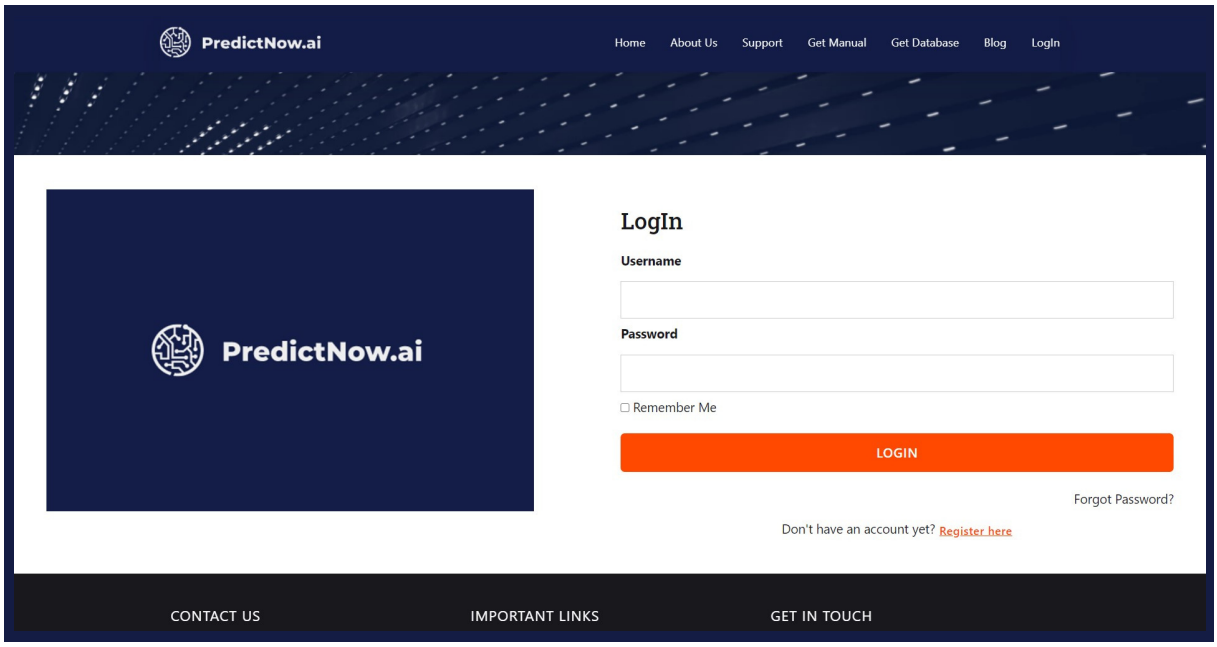


Figure 3: Login Form

6. User dashboard (Train/Live mode).

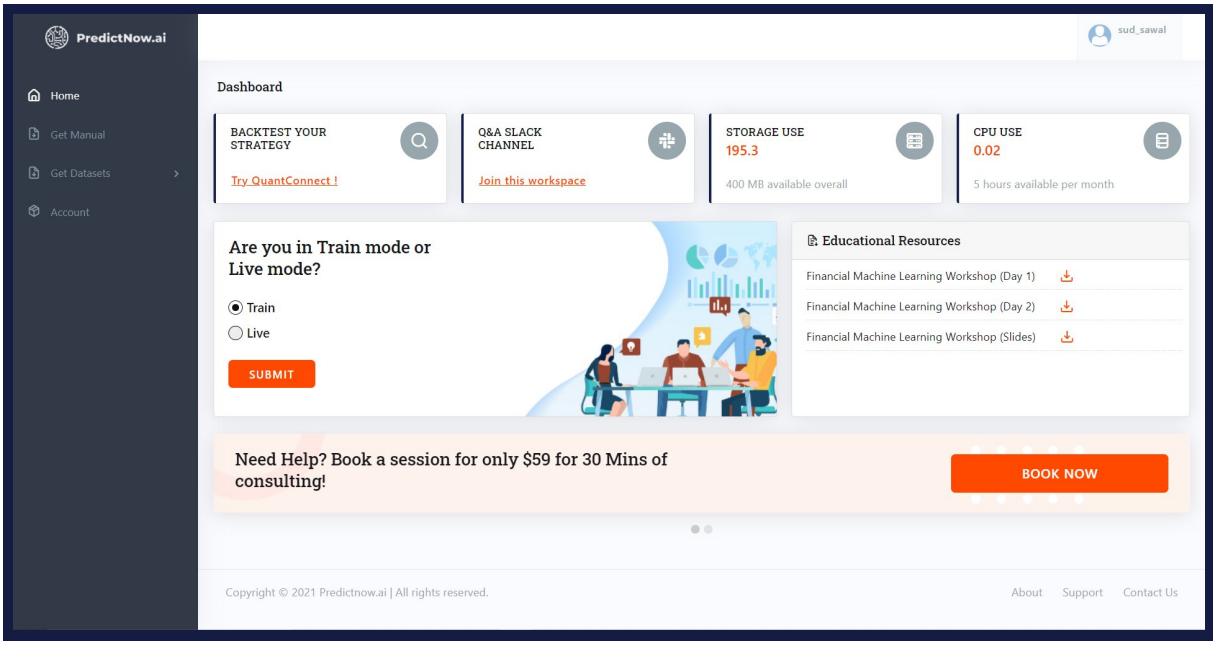


Figure 4: User Dashboard (Choose Between "Train/Live Mode")

Users have a choice between choosing "Train Mode" or "Live Mode". "Train Mode" involves splitting the data between train and test, building the model, then outputting results for CV and test set, where test set is specified as a parameter. "Live Mode" is used for live prediction only by using a pre-built machine learning model. Since we do not have existing machine learning model, choose "Train" and click "Submit" to open the range of options used to build such a model.

7. Assuming you have downloaded "example_input_train.csv" from the "Download Train example data" link at the top of the page, specify the data file on which the model is trained.

8. Upload your dataset by clicking on the box where it says "Drop files here to upload" or drag & drop it there. Please wait until the file has been uploaded.

You will notice the green progress bar which will display your progress. From the folder where "example_input_train.csv" has been saved, select it, and click Open. See Figure 5a). After the file has been successfully uploaded, click on "Next".

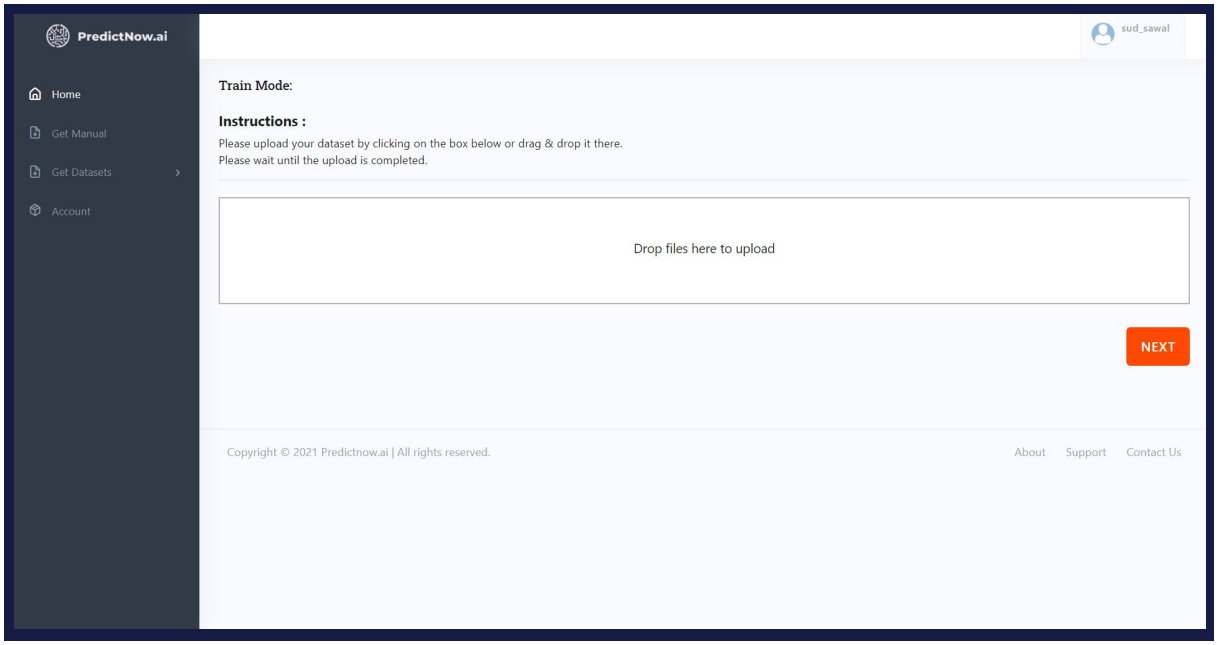


Figure 5a: Upload File Request

Figure 5b: File Successfully Uploaded

10. After clicking "Next", specify the parameters that will be used in order to train your machine learning model.

See Figure 4c. If you open this file in Excel, you will notice the first column is called "Date", and the last column is called "futreturn". Set the name of the target variable as "futreturn", denoting 10-day future return for (buy and hold) SPY index. The data is under timeseries format since the features are consecutive in time so require feature selection. When prompted with "Is your data file under timeseries format?" answer "Yes".

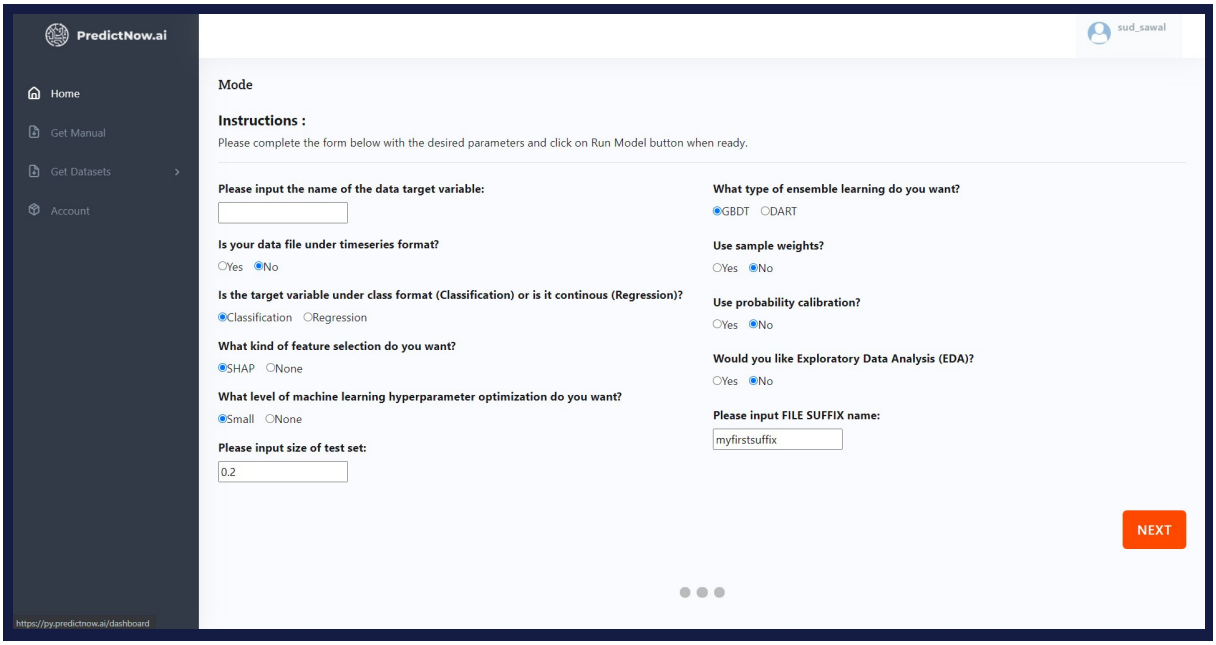


Figure 5c: Training Parameters Form

11. Decide whether the target variable is under class format or continuous format.

- If it is under class format, select "Classification" to the question "Is the target variable under class format (Classification) or is it continuous (Regression)?".
- For this particular example, we choose Classification because we want to predict whether the returns will be positive or negative. If one wants to predict actual values, one needs to choose "Regression".
- A suitable example for regression would be when one would like to predict house prices, actual returns, etc.

12. Keep level of machine learning optimization to "Small" for this example, to make computations faster.

NOTE: High machine learning optimization analysis could lead to better performance metrics. If you don't want hyperparameter optimization, choose "None" – the model will train using its default parameters as defined in "paragraph e" under the section "Hyperparameters for Training" above).

13. Make the size of the test set 200.

Recall that if this value >= 1, the last 200 rows are selected for test set. If the size of test set <1, we treat it as a fraction of the total number of rows.

14. Select "Ensemble Learning to be GBDT (default value, using Random Forrest), and let sample weights, prob calibration be set to 'no'. Set EDA analysis to 'yes'.

- After clicking on the "Submit" button, you are asked to review your inputs – if you are satisfied with the inputs provided, click on "Run Model" and wait for the ML calculation to take place.
- You will be informed about the status of the computations via a percentage progress bar, along with some extra information. It is advised to leave the web browser OPEN.
- If the computation has been completed successfully, you will receive a link in the browser (and also sent via an email) to access the results. If at any point an error occurs, you will be informed about the nature of the error and no CPU time will be added in your account.

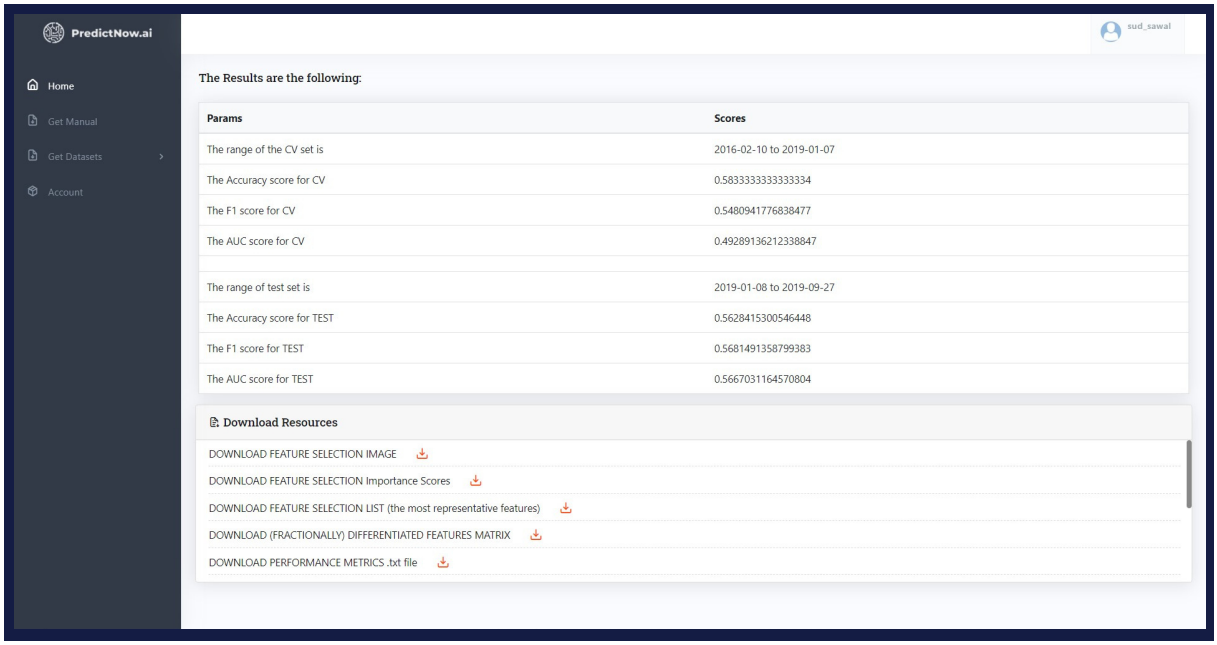


Figure 6: "Train Mode" Results

15. In the large yellow box, you can see the performance metrics for both CV and test set, along with their date range.

Recall that if the data has a column 'Date/date/Time/time', it will automatically set it as dates. If not, default indexing 0,1,2,… is performed.

16. The following download links follow the Outputs from training section.

NOTE: if EDA = "Yes", one can download basic statistics about the data inputted using "Download your "datasetdescribe().csv'", along with a more complicated analysis, in a (.html) file using "Download your dataset profile report". For our example, when downloaded, those files will have the names "example_input_train_describe().csv" and "profile_report_example_input_train.html".

WARNING: If you would like to later perform "Live "Mode", remember to click on "Download model file" because this will be used later. In our example, this file is called "saved_model_example.pkl". Also, please do not modify the name of the model file, because in "Live Mode", this name will be used to link it with the data used for training.

17. Navigate to your dashboard by clicking on "Home", either on the top page, or on the link on the bottom page.

18. Select "Live Mode" then click "Submit".

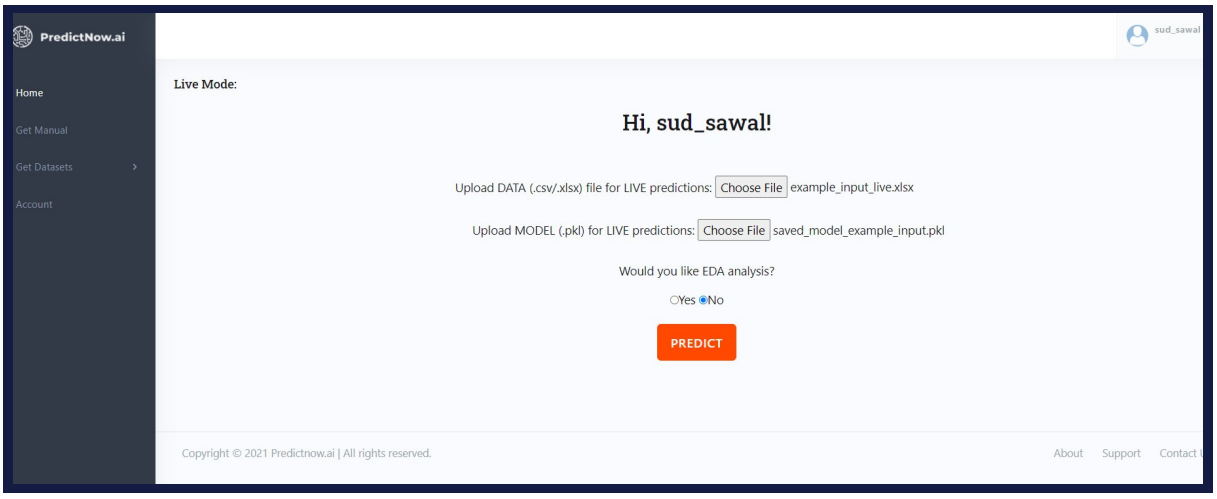


Figure 7: "Live Mode"

19. Assuming you have downloaded "example_input_live.csv" and an available model, such as "saved_model_example.pkl" upload these files and click "Predict".

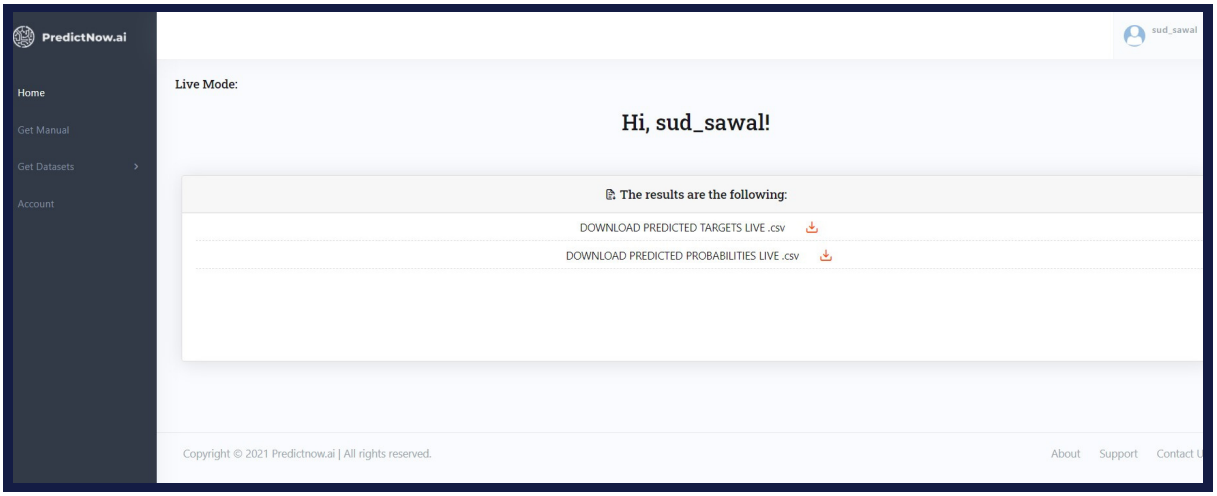


Figure 8: "Live Mode" Results

20. By using "Download PREDICTED TARGETS LIVE .csv", users can download the predicted targets for the feature matrix provided.

21. By using "Download PREDICTED PROBABILITIES LIVE .csv", users can download the predicted targets for the feature matrix provided.

SHARADAR FEATURES

NOTE: All data is updated every trading day by 5am New York Time

Balance Sheet

assets - total assets - sum of the carrying amounts as of the balance sheet date of all assets that are recognized. Major components are Cash and equivalents, investments, intangible assets, property plant & equipment net and trade and Non-Trade Receivables.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

deferredrev - deferred revenue - represents the carrying amount of consideration received or receivable on potential earnings that were not recognized as revenue; including sales; license fees; and royalties; but excluding interest income. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

deposits - deposit liabilities - represents the total of all deposit liabilities held; including foreign and domestic; interest and noninterest bearing. May include demand deposits; saving deposits; negotiable order of withdrawal and time deposits among others. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

equity - shareholders equity- a principal component of the balance sheet that represents the total of all stockholders' equity (deficit) items; net of receivables from officers; directors; owners; and affiliates of the entity which are attributable to the parent. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

debt - total debt - represents the total amount of current and non-current debt owed. Includes secured and unsecured bonds issued; commercial paper; notes payable; credit facilities; lines of credit; capital lease obligations; and convertible notes. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

inventory - represents the amount after valuation and reserves of inventory expected to be sold; or consumed within one year or operating cycle; if longer. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

investments - represents the total amount of marketable and non-marketable securities; loans receivable and other invested assets.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

payables - represents trade and non-trade payables.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

taxassets - tax assets - a component of assets representing tax assets and receivables.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

taxliabilities - tax liabilities - representing outstanding tax liabilities. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

cashneq - cash and equivalents - represent the amount of currency on hand as well as demand deposits with banks or financial institutions.This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is already a percentage. This reporting period could be quarterly, yearly or trailing twelve months.

Cash Flow Statement

capex - capital expenditure - A component of net cash flow from investing representing the net cash inflow (outflow) associated with the acquisition & disposal of long-lived; physical & intangible assets that are used in the normal conduct of business to produce goods and services and are not intended for resale. Includes cash inflows/outflows to pay for construction of self-constructed assets & software. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

depamor- depreciation amortization & accretion - a component of operating cash flow representing the aggregate net amount of depreciation; amortization; and accretion recognized during an accounting period. As a non-cash item; the net amount is added back to net income when calculating cash provided by or used in operations using the indirect method. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it.This reporting period could be quarterly, yearly or trailing twelve months.

ncf - net cash flow- principal component of the cash flow statement representing the amount of increase (decrease) in cash and cash equivalents. Includes net cash flow from operations; investing net cash flow from investing and financing for continuing and discontinued operations; and the effect of exchange rate changes on cash.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

Income Statement

cor - cost of revenue - The aggregate cost of goods produced and sold and services rendered during the reporting period.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

consolinc - consolidated income - The portion of profit or loss for the period; net of income taxes; which is attributable to the consolidated entity; before the deduction of Net Income to Non-Controlling Interests. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

ebit - earning before interest & taxes - earnings before interest and tax is calculated by adding income tax expense and interest expense back net income.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

gp - gross profit - aggregate revenue less cost of revenue directly attributable to the revenue generation activity. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

netinc - the portion of profit or loss for the period; net of income taxes; which is attributable to the parent after the deduction of net income to non-controlling interests from consolidated income; and before the deduction of prefered dividends income statement.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

opex - operating expenses represents the total expenditure on selling and general administrative expense; R&D and other operating expense items; it excludes cost of revenue. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

opinc - operating income is a measure of financial performance before the deduction of interest expense; tax expense and other non-operating items. It is calculated as gross profit minus operating expenditure.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

revenue - amount of revenue recognized from goods sold; services rendered; insurance premiums; or other activities that constitute an earning process. Interest income for financial institutions is reported net of interest expense and provision for credit losses.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

taxexp - income tax expense - Amount of current income tax expense (benefit) and deferred income tax expense (benefit) pertaining to continuing operations.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

netmargin - profit margin - measures the ratio between a company's net income and revenue.This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is already a percentage. This reporting period could be quarterly, yearly or trailing twelve months.

eps - earnings per basic share - earnings per share as calculated and reported by the company. Approximates to the amount of net income for common stock for the period per each weighted average shares. This feature was made stationary by calculating the difference between earnings normalised by enterprise value corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

dps - dividends per basic common share - aggregate dividends declared during the period for each split-adjusted share of common stock outstanding. This feature was made stationary by calculating the difference between total dividends by the firm normalised by enterprise value of the firm corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

tbvps - tangible assets book value per share - measures the ratio between tangibles and weighted average shares. This feature was made stationary by calculating the difference between tangible assets book value of the firm normalised by enterprise value of the firm corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

Metrics

fcf - free cash flow - free cash flow is a measure of financial performance calculated as net cash flow from operations minus capital expenditure.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

invcap - invested capital is an input into the calculation of return on invested capital; and is calculated as: Debt plus assets minus goodwill and intangible assets minus cash equivalents minus current liabilities. Please note this calculation method is subject to change. This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

tangibles - The value of tangibles assets calculated as the difference between assets and intangibles.This feature was made stationary by calculating the percentage change between the entries corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

grossmargin - Gross Margin - Gross Margin measures the ratio between a company's gross profit and revenue. This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is already a percentage. This reporting period could be quarterly, yearly or trailing twelve months.

payoutratio - payout ratio - the percentage of earnings paid as dividends to common stockholders. Calculated by dividing dividends per basic common share by earnings per basic share. This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is already a percentage. This reporting period could be quarterly, yearly or trailing twelve months.

de - debt to equity ratio - Measures the ratio between debt and equity. This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is already a ratio. This reporting period could be quarterly, yearly or trailing twelve months.

divyield - dividend yield - dividend yield measures the ratio between a company's dividend per common stock and its price. This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is a ratio. This reporting period could be quarterly, yearly or trailing twelve months.

pb - price to book value - measures the ratio between market capitalisation and shareholder equity in USD. This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is a ratio. This reporting period could be quarterly, yearly or trailing twelve months.

pe - price earnings (Damodaran Method) - measures the ratio between market capitalisation and net income for common stock. This feature was made stationary by calculating the difference between the entries corresponding to a given reporting period and the reporting period immediately before it. This is because this feature is a ratio. This reporting period could be quarterly, yearly or trailing twelve months.

bvps - book value per share - measures the ratio between shareholder equity and weighted average shares - This feature was made stationary by calculating the difference between book value normalised by enterprise value corresponding to a given reporting period and the reporting period immediately before it. This reporting period could be quarterly, yearly or trailing twelve months.

price - share price - The price per common share adjusted for stock splits but not adjusted for dividends

Daily Price Features

open - the price at market open

close - the price at market close

high - the highest trading price in the day

low - the lowest trading price in the day

volume - number of shares traded in the day

closeadj - the closing price adjusted for stock split and dividends

closeunadj -unadjusted close price. Same as Close.

sharebas - The number of shares or other units outstanding of the entity's capital or common stock or other ownership interests; as stated on the cover of related periodic report (10-K/10-Q); after adjustment for stock splits.

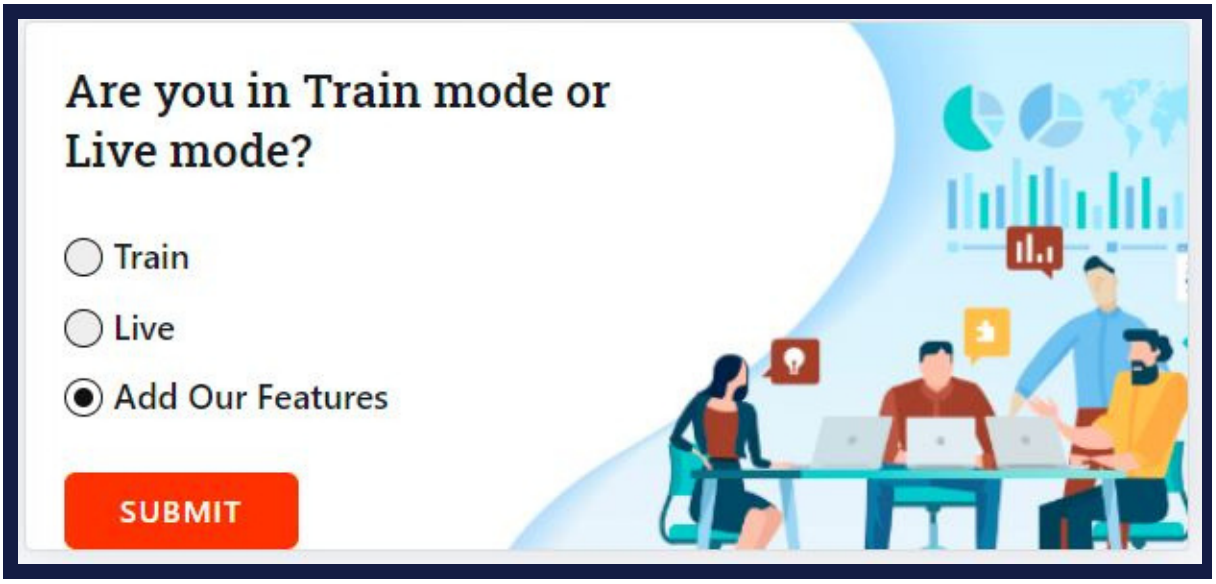
share_normalised_vol - volume normalized by total number of shares

Share_normalised_ev - enterprise value normalized by total number of shares

TRAINING YOUR DATA

PredictNow.ai is integrated with fundamental and price data from **Sharadar**. This includes most features from the document given here and also OHLCV features among others. The fundamental data has been stationarised and checked for structural breaks and survivorship biases. Both the fundamental and price data is updated on a daily basis. It includes all listed and delisted stocks. Let's look at how you can use data from **Sharadar** in your model training process.

Step 1: First select the "Add Our Features" mode to go to the **Sharadar** data page.



Step 2: On this page, select the tickers we might want to include in our data, the fundamental and daily price features corresponding to those tickers. For the fundamental data select the dimension of the data. The dimensions tell you the frequency of the reporting period. It can be either quarterly (ARQ), yearly (ARY) or trailing twelve month (ARY). AR here stands for "as reported" which is a point in time view of data, time-indexed to the date the form 10 regulatory filing was submitted to the SEC. This excludes changes due to any restatements on a later date. There might be zero or more entries for a reporting period.

Step 3: Next, choose the date range for which you'd like to view this data. This will display a view with the top 10 and the bottom 10 rows in the database filtered for the selected features, tickers, and dimensions. This is shown in the image below:

This table displays the top 10 and the bottom 10 rows of the data only.

While merging the last valid data will be forward filled.

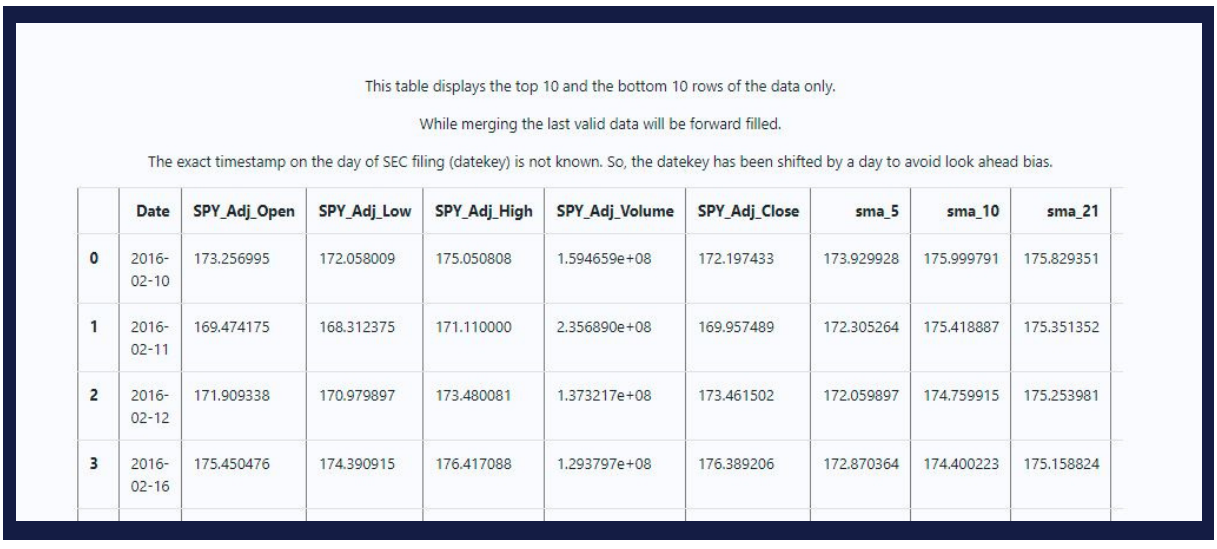
The exact timestamp on the day of SEC filing (datekey) is not known. So, the datekey has been shifted by a day to avoid look ahead bias.

	datekey	capex_AAPL_ARQ	assets_AAPL_ARQ	capex_TSLA_ARQ	assets_TSLA_ARQ	low_AAPL	low_TSLA
0	2012-01-25	0.687377	0.191714	NaN	NaN	15.848	5.410
1	2012-02-27	NaN	NaN	0.211812	0.018848	18.439	6.600
2	2012-04-25	-0.055983	0.088354	NaN	NaN	21.643	6.414
3	2012-05-10	NaN	NaN	-0.252939	0.066791	20.302	6.480
4	2012-07-25	-0.963552	0.079253	NaN	NaN	20.357	5.750
5	2012-08-02	NaN	NaN	0.098563	0.020719	21.438	5.104
6	2012-10-31	-0.181573	0.080837	NaN	NaN	20.989	5.474
7	2012-11-07	NaN	NaN	-0.117254	0.041587	19.848	6.162
8	2013-01-24	0.298772	0.113731	NaN	NaN	16.080	7.168

Step 4: Next, upload your input file. This is the file you want to merge the selected data to for model training.

12	2018-02-02	0.259000	0.083862	NaN	NaN	40.025	68.102
13	2018-02-23	NaN	NaN	0.272015	0.019507	43.385	69.420
14	2018-05-02	-0.363360	-0.096589	NaN	NaN	43.450	59.557
15	2018-05-07	NaN	NaN	0.195892	-0.048296	46.188	59.034
16	2018-08-01	0.191537	-0.049809	NaN	NaN	49.328	58.600
17	2018-08-06	NaN	NaN	0.070576	0.023415	51.767	68.364
18	2018-11-02	NaN	NaN	0.173428	0.048467	51.358	68.182
19	2018-11-05	0.069177	0.047331	NaN	NaN	49.542	66.028

Step 5: Once the data is uploaded you'll click on merge. This will merge the fundamental data and price data for the selected tickers, features, and for the date range as the input data. The features in the input data file come from the data price data for the SPY ticker as we can tell from the feature names.



This table displays the top 10 and the bottom 10 rows of the data only.

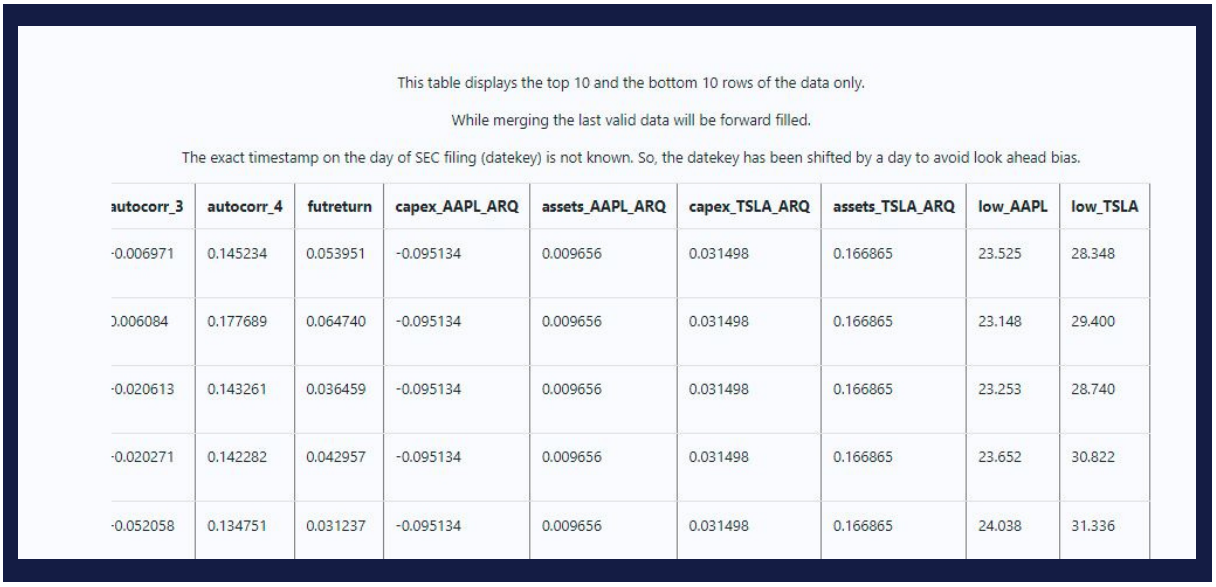
While merging the last valid data will be forward filled.

The exact timestamp on the day of SEC filing (datekey) is not known. So, the datekey has been shifted by a day to avoid look ahead bias.

	Date	SPY_Adj_Open	SPY_Adj_Low	SPY_Adj_High	SPY_Adj_Volume	SPY_Adj_Close	sma_5	sma_10	sma_21
0	2016-02-10	173.256995	172.058009	175.050808	1.594659e+08	172.197433	173.929928	175.999791	175.829351
1	2016-02-11	169.474175	168.312375	171.110000	2.356890e+08	169.957489	172.305264	175.418887	175.351352
2	2016-02-12	171.909338	170.979897	173.480081	1.373217e+08	173.461502	172.059897	174.759915	175.253981
3	2016-02-16	175.450476	174.390915	176.417088	1.293797e+08	176.389206	172.870364	174.400223	175.158824

Note: If you scroll to the very right, you will see the **Sharadar** data rows appended after the uploaded data.

In the example, the fundamental features we append are assets (total assets) and caxpex (capital expenditure) for tesla (TSLA) and apple (AAPL) at a quarterly frequency (ARQ). The price feature appended is the day low for both the tickers. For fundamental data, you don't see whole numbers because as mentioned above, the data has been made stationary.



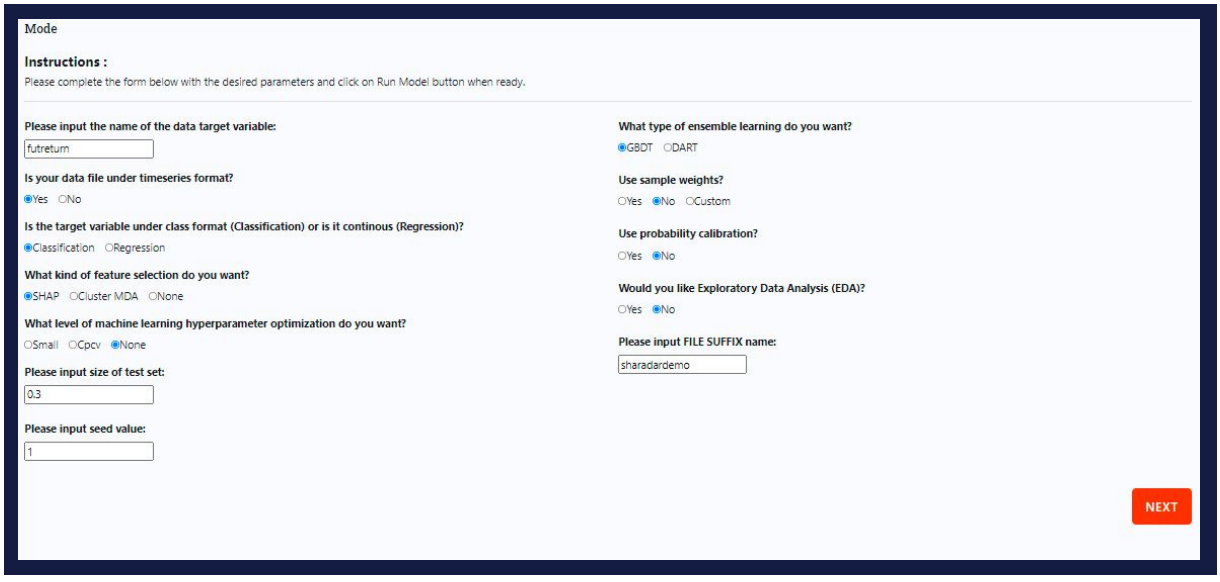
This table displays the top 10 and the bottom 10 rows of the data only.

While merging the last valid data will be forward filled.

The exact timestamp on the day of SEC filing (datekey) is not known. So, the datekey has been shifted by a day to avoid look ahead bias.

autocorr_3	autocorr_4	futreturn	capex_AAPL_ARQ	assets_AAPL_ARQ	capex_TSLA_ARQ	assets_TSLA_ARQ	low_AAPL	low_TSLA
-0.006971	0.145234	0.053951	-0.095134	0.009656	0.031498	0.166865	23.525	28.348
0.006084	0.177689	0.064740	-0.095134	0.009656	0.031498	0.166865	23.148	29.400
-0.020613	0.143261	0.036459	-0.095134	0.009656	0.031498	0.166865	23.253	28.740
-0.020271	0.142282	0.042957	-0.095134	0.009656	0.031498	0.166865	23.652	30.822
-0.052058	0.134751	0.031237	-0.095134	0.009656	0.031498	0.166865	24.038	31.336

Step 6: Click on "Next" and select model parameters as shown below. And then click next to go to the model train panel.



Note: In the panel, you can rectify the parameters selected in the previous page and start model training. You can choose to make a classification or regression model depending on your need.

Params	Value
Filename	example_input_train_sharadar.xlsx
Label	futreturn
Timeseries	yes
Type of target variable	classification
Feature selection	shap
Analysis	none
Boost	gbdt
Testsize	0.3
Seed	1
Weights	no | Custom: NA
Probability calibration	no
Suffix	sharadardemo
Exploratory data analysis (EDA)	no

100% Prediction completed! Experiment complete.

Success! Results Link

After training, you can see the performance metrics and also download files containing the model, train and test probabilities, features importance etc. This is shown below:

The Accuracy score for TEST	0.5127272727272727
The F1 score for TEST	0.45381649092318166
The AUC score for TEST	0.4514392441221709
The count of class 1.0 for Test in TrueTarget is	164
The count of class 0.0 for Test in TrueTarget is	111
The count of class 1 for Test in PredictedTarget is	228
The count of class 0 for Test in PredictedTarget is	47

Download Resources

DOWNLOAD PREDICTED TARGETS CV .csv

DOWNLOAD PREDICTED PROBABILITIES CV .csv

DOWNLOAD PREDICTED TARGETS TEST .csv

DOWNLOAD PREDICTED PROBABILITIES TEST .csv

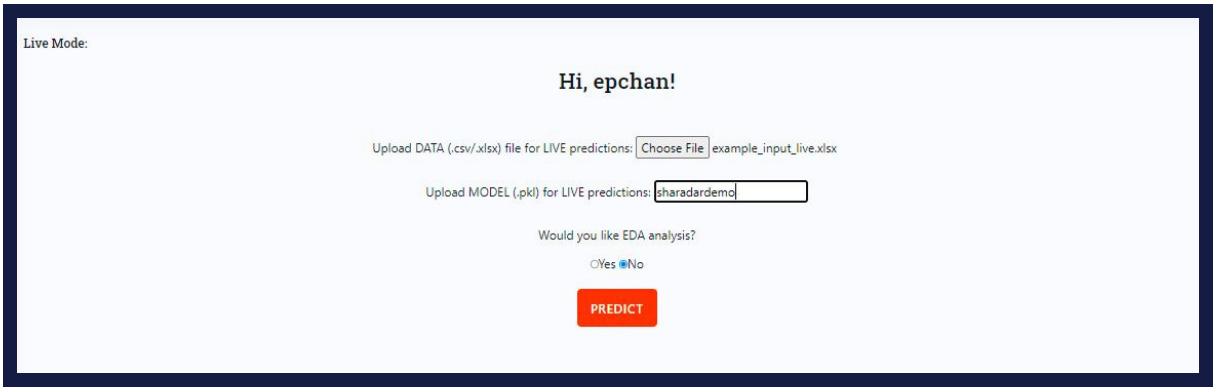
DOWNLOAD MODEL file

LIVE MODE

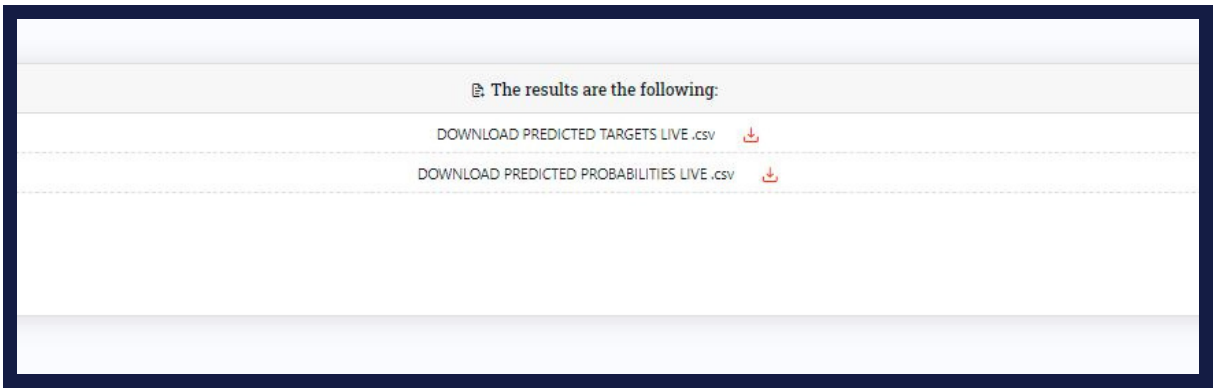
Once you have trained the model, you can try out the model trained with the **Sharadar** data in the live mode.

Step 1: Return to the dashboard and select "Live" mode.

Step 2: After entering the "Live" prediction panel, upload the input file and fill in the name of the suffix that you used for the "Train" mode. Then click on "predict".



Note: Once, the prediction is made you will be able to download the predicted label and probability file.



A sample probability file is shown below. This tells you the probability allocated for each label in the prediction by the model.

	Date	0	1
2	9/30/2019	0.061504	0.938496
3	10/1/2019	0.015925	0.984075
4	10/2/2019	0.006276	0.993724
5	10/3/2019	0.007071	0.992929
6	10/4/2019	0.010987	0.989013
7	10/7/2019	0.008403	0.991597
8	10/8/2019	0.007296	0.992704
9	10/9/2019	0.015719	0.984281
10	10/10/2019	0.016367	0.983633
11	10/11/2019	0.01639	0.98361
12	10/14/2019	0.014623	0.985377
13	10/15/2019	0.045575	0.954425
14	10/16/2019	0.015947	0.984053
15	10/17/2019	0.069443	0.930557
16	10/18/2019	0.026634	0.973366
17			

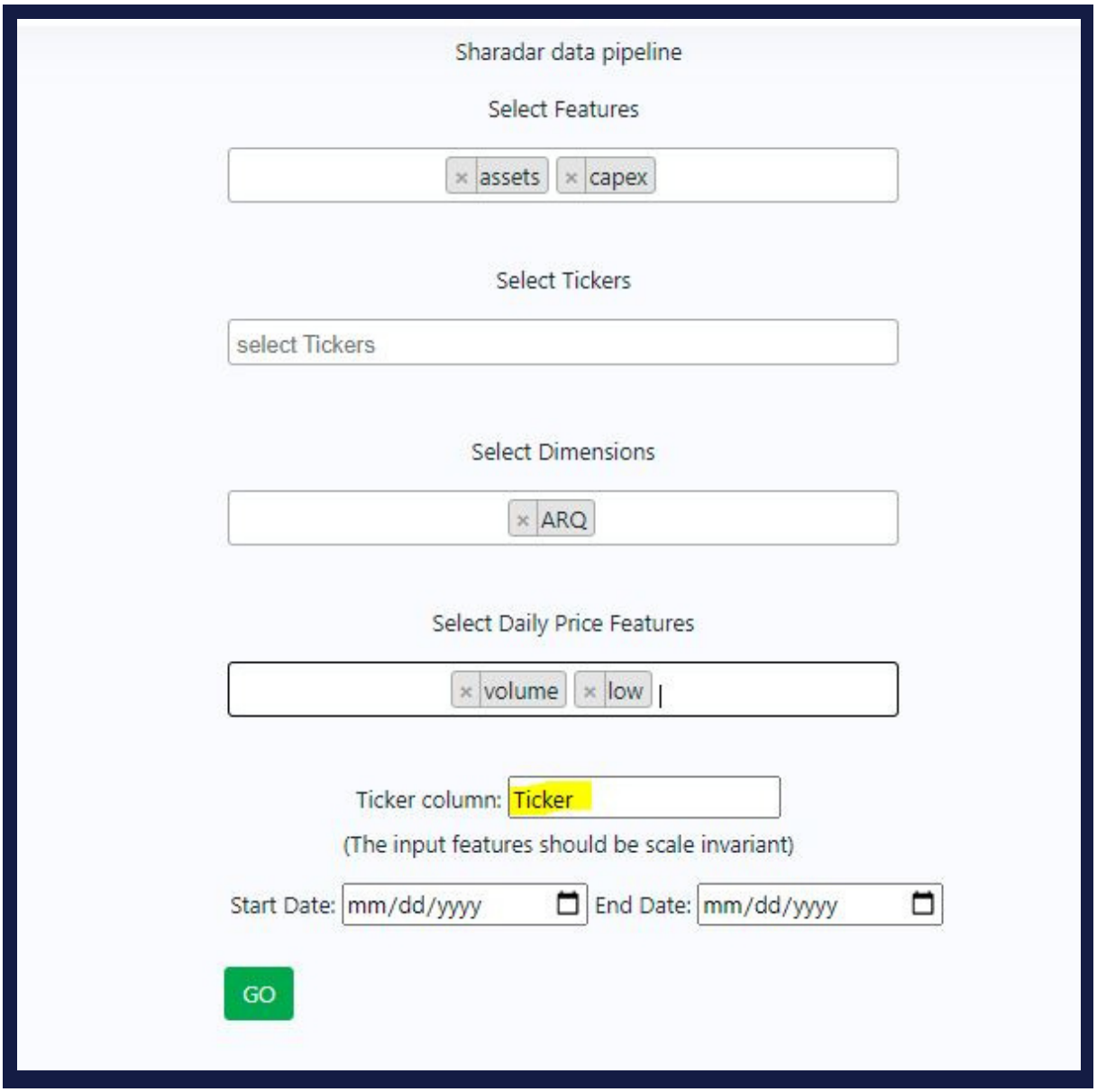
SHARADAR TRAINING FOR MULTI-TICKER DATA

The above example was for an input dataset where the features belonged to just one ticker. Namely SPY. But, we also provide the user with the option to train data belonging to multiple tickers. A snippet from a sample input data file is given below:

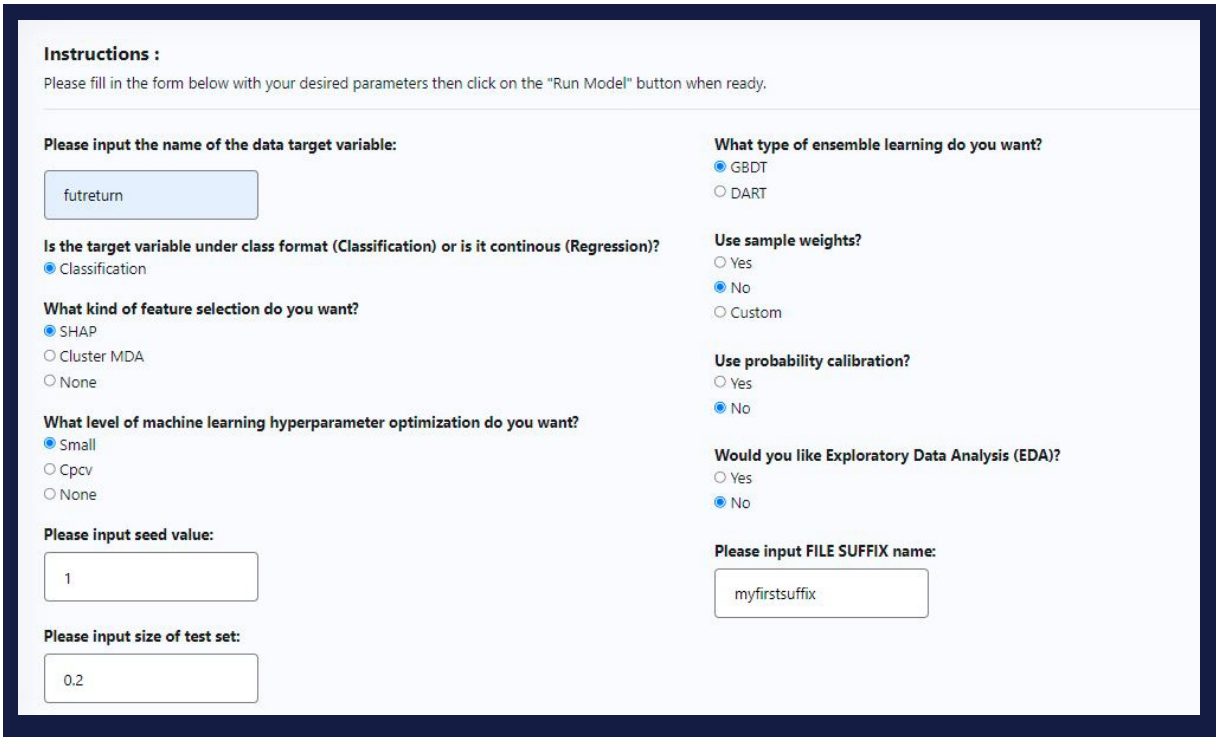
Date	Ticker	Adj_Open	Adj_Low	Adj_High	Adj_Volume	Adj_Close	sma_5	sma_10
2/10/2016	AAPL	173.2569951	172.0580086	175.0508077	159465948	172.197433	173.9299284	175.9997909
2/11/2016	AAPL	169.4741753	168.3123747	171.1099999	235689000.3	169.957489	172.305264	175.4188873
2/12/2016	AAPL	171.9093381	170.9798975	173.4800806	137321741	173.461502	172.0598966	174.7599151

A	B	C	D	E	F	G	H
Date	Ticker	Adj_Open	Adj_Low	Adj_High	Adj_Volume	Adj_Close	sma_5
9/6/2019	TSLA	296.7994053	296.0528528	297.3866902	49813278.3	296.679932	293.2657044
9/9/2019	TSLA	297.764936	295.7940267	297.8644515	51497020.15	296.829254	294.4104188
9/10/2019	TSLA	295.9931026	294.6095079	296.8292682	58214697.02	296.759583	295.8816286
9/11/2019	TSLA	297.0979953	296.381304	298.9593943	69138917.5	298.869812	297.117914
9/12/2019	TSLA	299.8652349	299.0291001	301.0696639	73245389.32	299.90506	297.8087282

The above screenshots are from two different parts of the input file. We can see that in the input dataset we have features for both TSLA and AAPL. They can be identified based on the ticker mentioned in the column named, Ticker. Before uploading this multi ticker input file we need to mention the name of the column name which identifies the ticker the corresponding row belongs to. This is shown in the screenshot below:



Once the ticker name is mentioned and the other inputs selected, we follow the same process as the one mentioned for the single ticker input file training. One difference, however, is that the merged data will not be treated as a time series and the model can only be a classification model. This reflects in the training panel in the screenshot below:


Instructions :
Please fill in the form below with your desired parameters then click on the "Run Model" button when ready.
Please input the name of the data target variable:
futreturn
Is the target variable under class format (Classification) or is it continous (Regression)?
Classification
What kind of feature selection do you want?
SHAP
Cluster MDA
None
What level of machine learning hyperparameter optimization do you want?
Small
Cpcv
None
Please input seed value:
1
Please input size of test set:
0.2
What type of ensemble learning do you want?
GBDT
DART
Use sample weights?
Yes
No
Custom
Use probability calibration?
Yes
No
Would you like Exploratory Data Analysis (EDA)?
Yes
No
Please input FILE SUFFIX name:
myfirstsuffix

FREQUENTLY ASKED QUESTIONS

1.What type of file should I provide?

You can use an Excel file (.xlsx) or a (.csv) file. Do not use whitespaces in the name of the file.

2. How should my dataset look like?

Your dataset should have rows and columns, but not charts. It is advised to use a dataset with at least 100 rows. It's important to name your columns and avoid using whitespaces in naming columns.

3. What is "Train Mode" and "Live Mode"?

These are the 2 modes used for HAL machine learning. In "Train Mode" you train a model by splitting your dataset in train and test sets, construct a machine learning model using train set data, and make predictions and performance metrics based on both train and test set data. In "Live Mode" you perform predictions on out-of-sample data.

NOTE: You cannot use :Live Mode" if you have not previously used "Train Mode".

4. What is a target (a.k.a label)?

A target is a column that contains real numbers which the program will convert into 2 classes based on their signs, as described above. If the target is positive, the program will convert this to class 1. If the target is "0" or negative, the program will convert this to class 0.

5. How many target/label columns can I have?

Only one label (column) should be present in the dataset.

6. I'm using PredictNow.ai for metalabelling but my strategy does not trade frequently. What should I do?

In a metalabel application, if a trade was not made for a certain sample, the target should be "NaN" or "Null" instead of "0".

7.I get an error when I attach a (.csv) file, what should I do?

Most likely, when a user opens a (.csv) file in Excel and saves it, it will pop up a warning that says "some characters may not be well converted". If this occurs, you can try to upload a (.xlsx) file in it's place. We recommend that if you are constructing your dataset via Excel, you should save documents to a (.xlsx) file rather than a (.csv) file.

8. What are 'Classification' and 'Regression' parameters?

If you want to predict actual numbers/values, you need to choose Regression. If you want to predict whether your target variable belongs to a certain class, such as "0" or "1", choose "Classification".

9. What size for test set should I provide?

You may choose any value between "0.01" and "0.999", to represent the percentage of data used as test set. We recommend to use "0.3" (i.e. 70% of data will be used for training, whereas the remaining 30% will be used for testing). If you input an integer number greater than "1", then we treat this as the exact number of test rows. If you input "1" as the size of the test set, then only 1 row of data will be used as test size.

10. What exactly is the 'suffix'?

It is a name for your model file used to keep track of machine learning logic internally.

11. Can I have 2 dates column?

No, the dataset must have only one date column. Make sure this is named as "Date/date/DATE/Time/TIME/time".

12. Is it necessary to always provide a date column?

It is not mandatory, but any strings/non-integer values will be used for one-hot encoding. We recommend, at least for the prototype version, to use a date column for simplicity.

13. Can I have column names on different rows?

No, column names can only occur on the horizontal axis. Look at the example (.xlsx)/(.csv) files from the download links for comparison.

14. How much is the waiting time to get results?

The waiting time varies depending on the size of your dataset. For comparison, a dataset with approx. 6000 rows and 90 columns will get around of 10 mins of waiting time. Smaller datasets will require much less waiting time.

15. Now I can view the Results page. What should I do?

You can look at the performance metrics of your trained machine learning model. You can also download the results via download links information about predicted probabilities (of achieving a class 1), as well as predicted targets (either "1" or "0"), for both train (via CV) and test data.

16. There is a "DOWNLOAD MODEL" link at the bottom of the Results page. What should I do?

It is mandatory to download this model if you intend to do live predictions. Please do not rename this file.

17. I can't get the "Live Mode" working. It outputs 'Internal server error'. What should I do?

Make sure you have the same column names in your dataset (check for typos as well!). It is not necessary to include the target/label column for Live mode.

NOTE: Please do not modify the name of the model file that you will upload when being asked in the "Live Mode", as this name will be used to link it with the data used for training.

18. I keep getting failure for cMDA option, but it works when I select SHAP option. What's wrong?

Since cMDA is null sensitive, try to minimize the null values in your Dataset. Our program does take care of null values with cMDA but if "Failure" persists, try minimizing the number of null values.

19. I obtained Results for Live mode. What are these?

You can download predicted probabilities as well as predicted targets for your live data that you uploaded.