

---

# Edgar Documentation

*Release 0.1.1*

**Brandon**

December 09, 2013



## CONTENTS

|          |                              |           |
|----------|------------------------------|-----------|
| <b>1</b> | <b>Module: Form10DB</b>      | <b>3</b>  |
| <b>2</b> | <b>Module: forecaster</b>    | <b>5</b>  |
| <b>3</b> | <b>Module: utils</b>         | <b>7</b>  |
| <b>4</b> | <b>Module: DataGenerator</b> | <b>9</b>  |
| <b>5</b> | <b>Indices and tables</b>    | <b>11</b> |
|          | <b>Python Module Index</b>   | <b>13</b> |
|          | <b>Index</b>                 | <b>15</b> |



Contents:



## MODULE: FORM10DB

edgar.form10DB includes functions to download 10 forms from dataset.

It includes ticker to cik transformation, quarter master file download, and 10 form download functions.

**class** form10DB.**CIKTranslator**

CIKTranslator manages the mapping between cik and ticker

**Users can get the dictionary from two members:** CIKtranslator.tickerGetter: dictionary with cik:ticker

CIKtranslator.cikGetter: dictionary with ticker:cik

**updateCIKtable** (*tickers*)

generate a txt file that map ticker to cik

**Args:** sequence of symbol(str): symbols to add to the dictionary

**Return:** emptycik(list): list of ticker without cik

**Output:** DATA\_ADD + '/Edgar/cik2ticker.txt'

**class** form10DB.**Form10Manager**

MasterFileManager download and maintains quarterly master file of EDGAR database

**buildIndex** ()

build index file for each ticker directory

**listQuarterMasterFile** ()

List all available quarterly master file

**updateAndDownload** (*year, qtr, range='all'*)

update master file and download new forms

**updateForms** (*year, qtr, range='all'*)

download all 10 forms listed in the master file to corresponding ticker directory

**Args:** year,qtr(int): year and quarter of master file range(str): specify date range to download

'all': download all forms in the master file 'today': download forms with date == today

return:

**updateQuarterMasterFile** (*beginyear, beginqtr, endyear, endqtr*)

Download quarterly master file from edgar database Args:

beginyear, ..., endqtr (int): specify required date range





## MODULE: FORECASTER

**class** `forecaster.DictionaryGenerator`

Generate a dictionary from the files indicated in the keyfile

**GenerateDictionary** (*keyfile*, *outdictname*)

**Args:** *keyfile*(str): path to look for the keyfile *outdictname*: name of the output dictionary

**class** `forecaster.Forecaster`

Forecaster is wrapper class for different classifiers

**Forecaster can:**

1. train and test on historical edgar data
2. make predictions on new incoming data

**Constructor Args:** *indexPath*(str): path to the index file of all training data

**GenerateTrainTestDataSet** (*traincut*, *devcut*)

cut training set and testing set based on cut date(cut date include in training set)

**setClassifier** (*classifier\_*)

set the classifier of the forecaster.

The classifier should have a *fit(X,y)*, *score(X,y)* and *predict(X)* functions

**Args:** *classifier*: instance of a classifier

`forecaster.doMostCommonClassify` (*keyfile*, *outfilename*)

guess all label to be 1, 2,3 Most Common Baseline

`forecaster.doWordListClassify` (*keyfile*, *outfilename*, *sue\_weight=0.0*, *k=0*)

count pos/neg words to make classification



## MODULE: UTILS

edgar.form10DB includes functions to download 10 forms from dataset.

It includes ticker to cik transformation, quarter master file download, and 10 form download functions.

**class** form10DB.**CIKTranslator**

CIKTranslator manages the mapping between cik and ticker

**Users can get the dictionary from two members:** CIKtranslator.tickerGetter: dictionary with cik:ticker

CIKtranslator.cikGetter: dictionary with ticker:cik

**updateCIKtable** (*tickers*)

generate a txt file that map ticker to cik

**Args:** sequence of symbol(str): symbols to add to the dictionary

**Return:** emptycik(list): list of ticker without cik

**Output:** DATA\_ADD + '/Edgar/cik2ticker.txt'

**class** form10DB.**Form10Manager**

MasterFileManager download and maintains quarterly master file of EDGAR database

**buildIndex** ()

build index file for each ticker directory

**listQuarterMasterFile** ()

List all available quarterly master file

**updateAndDownload** (*year, qtr, range='all'*)

update master file and download new forms

**updateForms** (*year, qtr, range='all'*)

download all 10 forms listed in the master file to corresponding ticker directory

**Args:** year,qtr(int): year and quarter of master file range(str): specify date range to download

'all': download all forms in the master file 'today': download forms with date == today

return:

**updateQuarterMasterFile** (*beginyear, beginqtr, endyear, endqtr*)

Download quarterly master file from edgar database Args:

beginyear, ..., endqtr (int): specify required date range



## MODULE: DATAGENERATOR

`DataGenerator.DatabaseIndex` (*sector*)

generate a key file indexing all data points for a sector file format:

date symbol label path

**TODO:** reset path

`DataGenerator.GenerateDataset` (*sector, data, begin, end*)

Generate data set Args:

sector(str): currently only 'Industrials' available data: data returned by loadData begin: begin index of the symbol list end: end index of the symbol list

`DataGenerator.checkEmpty` (*symbolList*)

return ticker list with empty index file

**Args:** symbolList: list of symbols

**Return:**

**result:**{'NoSymbol': symbols not in the cik-ticker dict 'NoIndex': symbol in the cik-ticker dict, but does not have index file}

`DataGenerator.cleanPrice` (*price*)

check same ticker used by two companies

`DataGenerator.loadData` (*sector*)

read the price, portfolio and sue data into memory

`DataGenerator.updateCoreIndex` ()

Core index file stores 10QK filing information of all ticker Update core index file based on the indexfile of each stock



## INDICES AND TABLES

- *genindex*
- *modindex*
- *search*





**d**

DataGenerator, 9

**f**

forecaster, 5

form10DB, 7



## B

buildIndex() (form10DB.Form10Manager method), 3, 7

## C

checkEmpty() (in module DataGenerator), 9

CIKTranslator (class in form10DB), 3, 7

cleanPrice() (in module DataGenerator), 9

## D

DatabaseIndex() (in module DataGenerator), 9

DataGenerator (module), 9

DictionaryGenerator (class in forecaster), 5

doMostCommonClassify() (in module forecaster), 5

doWordListClassify() (in module forecaster), 5

## F

Forecaster (class in forecaster), 5

forecaster (module), 5

form10DB (module), 3, 7

Form10Manager (class in form10DB), 3, 7

## G

GenerateDataset() (in module DataGenerator), 9

GenerateDictionary() (forecaster.DictionaryGenerator method), 5

GenerateTrainTestDateSet() (forecaster.Forecaster method), 5

## L

listQuarterMasterFile() (form10DB.Form10Manager method), 3, 7

loadData() (in module DataGenerator), 9

## S

setClassifier() (forecaster.Forecaster method), 5

## U

updateAndDownload() (form10DB.Form10Manager method), 3, 7

updateCIKtable() (form10DB.CIKTranslator method), 3, 7

updateCoreIndex() (in module DataGenerator), 9

updateForms() (form10DB.Form10Manager method), 3, 7

updateQuarterMasterFile() (form10DB.Form10Manager method), 3, 7